Die approbierte Originalversion dieser Dissertation ist in der Hauptbibliothek der Technischen Universität Wien aufgestellt und zugänglich.
http://www.ub.tuwien.ac.at

The approved original version of this thesis is available at the main library of the Vienna University of Technology.
http://www.ub.tuwien.ac.at/eng

INSTITUT FÜR
MECHANIK UND
MECHATRONIK
Mechanics & Mechatronics

DISSERTATION

Parameter Identification in Multibody System Dynamics using the Adjoint Sensitivity Analysis

ausgeführt zum Zwecke der Erlangung des akademischen Grades
eines Doktors der technischen Wissenschaften (Dr. techn.)
eingereicht an der TU Wien, Fakultät für Maschinenwesen und Betriebswissenschaften von

Dipl.-Ing. Stefan Oberpeilsteiner, BSc.

Mat.Nr.: 1328814

unter der Leitung von

Priv. Doz. Dipl.-Ing. Dr. Wolfgang Steiner
Institut für Mechanik und Mechatronik
Abteilung für Mechanik fester Körper

begutachtet von

Ao.Univ.Prof. Dipl.-Ing. Dr. Alois Steindl
Technische Universtät Wien
Institut für Mechanik und Mechatronik
Getreidemarkt 9, 1060 Wien

Prof. Dr.-Ing. Robert Seifried
Technische Universität Hamburg
Institut für Mechanik und Meerestechnik
Eißendorfer Straße 42, 21073 Hamburg

Diese Arbeit wurde von der Österreichischen Forschungsförderungsgesellschaft mbH (FFG) im Rahmen des Projekts ProtoFrame (Projektnr. 839074) unterstützt.

Ich nehme zur Kenntnis, dass ich zur Drucklegung meiner Arbeit unter der Bezeichnung Dissertation nur mit Bewilligung der Prüfungskommission berechtigt bin.

Eidesstaatliche Erklärung

Ich erkläre an Eides statt, dass die vorliegende Arbeit nach den anerkannten Grundsätzen für wissenschaftliche Abhandlungen von mir selbstständig erstellt wurde. Alle verwendeten Hilfsmittel, insbesondere die zugrunde gelegte Literatur, sind in dieser Arbeit genannt und aufgelistet. Die aus den Quellen wörtlich entnommenen Stellen, sind als solche kenntlich gemacht.
Das Thema dieser Arbeit wurde von mir bisher weder im In- noch Ausland einer Beurteilerin/einem Beurteiler zur Begutachtung in irgendeiner Form als Prüfungsarbeit vorgelegt. Diese Arbeit stimmt mit der von den Begutachterinnen/Begutachtern beurteilten Arbeit überein.

Linz, 20. Juli 2018

Stefan Oberpeilsteiner

Danksagung

An dieser Stelle möchte ich mich bei all jenen Menschen bedanken, die mich in den letzten Jahren unterstützt haben und so zum Gelingen dieser Arbeit beigetragen haben.
Mein besonderer Dank gilt meinem Betreuer und Vorgesetzten Dr. Wolfgang Steiner. Lieber Wolfgang, ich möchte mich für all deine Mühen und deine Unterstützung bei meiner Forschungstätigkeit und der Ausarbeitung meiner Dissertationsschrift bedanken. Ich habe mich sehr über dein Angebot zur Mitarbeit in deiner Forschungsgruppe gefreut.
Für die Bereitschaft meine Arbeit zu begutachten möchte ich Prof. Dr. Alois Steindl und Prof. Dr. Robert Seifried danken. Vielen Dank für die hilfreichen Anregungen und die wertvollen Diskussionen.
In der Zeit an der Fachhochschule Wels durfte ich mit ausgezeichneten Kollegen zusammenarbeiten. Auch bei diesen möchte ich mich ganz herzlich für das tolle Arbeitsklima bedanken. Allen voran gilt mein Dank Thomas Lauß. Unsere zahlreichen fachlichen Diskussionen haben enorm dazu beigetragen, dass die Arbeit in dieser Form entstehen konnte.
Ein herzliches Dankeschön möchte ich meiner Frau Judith aussprechen. Du hast mich immer wieder zum Vervollständigen meiner Arbeit motiviert und mir auch den notwendigen Rückhalt gegeben. Ebenso möchte ich meiner Familie, ganz besonders meinen Eltern, für die jahrelange Unterstützung und das ehrliche Interesse an meiner Arbeit danken.

Kurzfassung

Der verstärkte Einsatz von Software zur Mehrkörpersimulation ermöglicht es die Anzahl von Prototypen, welche während der Entwicklung eines Produkts benötigt werden, zu reduzieren. Häufig verursachen ungenaue oder unbekannte Parameterwerte große Abweichungen der Simulationsergebnisse von den in der Realität gemessenen Werten. Die Qualität eines virtuellen Prototyps kann durch Anpassung der Parameterwerte an die realen Verhältnisse erhöht werden. Durch die Komplexität von Mehrkörpersystemen ist es bei einer großen Anzahl von Parametern in einem Simulationsmodell meist nicht mehr möglich die Parameter manuell anzupassen. Aus diesem Grund stellt ein automatisierter und effizienter Abgleich der Systemausgänge den einzig möglichen Ansatz zur Verbesserung von Simulationsergebnissen dar. Darüber hinaus kann durch eine schlecht gewählte Anregung der Fall eintreten, dass der Einfluss der Parameter auf die Systemausgänge zu gering ist. In diesem Fall stützt sich die Identifikation auf unzureichende Daten und führt zu einer nicht zufriedenstellenden Qualität des virtuellen Prototyps.

Zur Automatisierung des Parameteridentifikationsprozesses wird ein sinnvolles Gütemaß benötigt, welches erlaubt die Abweichung der Simulation vom Experiment zu quantifizieren. Erst durch Einführen dieser sogenannten Kostenfunktion wird die Verwendung eines iterativen Ansatzes zum Lösen des Optimierungsproblems, welches die skalare Kostenfunktion minimiert, ermöglicht. Der Gradient, welcher die Konvergenz des Optimierungsverfahrens deutlich verbessert, kann mit Hilfe der adjungierten Sensitivitätsanalyse berechnet werden. Um den Informationsgehalt in den zur nachfolgenden Parameteridentifikation verwendeten Messungen zu steigern, wird ein Ansatz verfolgt, der durch Modifikation der Systemanregung auf eine Optimierung der Sensitivität der Kostenfunktion in Bezug auf Parameteränderungen abzielt.

Für beide Spezialfälle, Gradientenberechnung und Optimierung der Systemanregung, werden detaillierte Herleitungen durchgeführt. Neben der Beschreibung der entwickelten Ansätze werden nachvollziehbare Beispiele gezeigt, welche die Performance der jeweiligen Methode unterstreichen.

Abstract

The usage of state-of-the-art software for analyzing the dynamics of multibody systems allows to reduce the number of prototypes in an product development process. Often, unknown parameters cause notable deviations of simulation results compared to measurements taken during experiments. Improving the quality of the virtual prototype may be achieved by matching the parameters used in the simulation model with the real ones. With an increasing number of parameters, adjusting their values manually in order to improve the accordance is hardly possible due to the complexity of the multibody system. Therefore, an automated and efficient strategy for parameter identification represents the only reasonable approach for gaining better simulation results. Another problem arises due to the fact, that the chosen excitations may not cause a sufficient reaction of the components under consideration. In such a case the result of the identification relies on insufficient data, and therefore the accuracy of the virtual prototype is not satisfactory.

Automating the process of parameter identification requires a meaningful performance measure in order to quantify the deviation of experiment and simulation. This allows for using an iterative approach that aims at solving the optimization problem that minimizes a scalar performance measure. The gradient required by the optimization algorithm is computed by using the adjoint sensitivity analysis. Addressing the problem raised by insufficient information contained in the measurements is done by adjusting the system inputs in order to maximize the performance measure's sensitivity onto parameter changes, usually denoted as *optimal input design*.

For both special issues, the computation of the gradient and the optimization of system inputs, detailed derivations are done. Besides the description of procedures developed, comprehensible examples are presented for emphasizing the performance of the respective method.

Contents

Kurzfassung **iv**

Abstract **v**

Contents **vi**

1 Introduction **1**
- 1.1 Literature Review 2
  - 1.1.1 Multibody dynamics 2
  - 1.1.2 Adjoint method 2
- 1.2 Thesis Objectives 4
- 1.3 Thesis Structure 4

2 Dynamic Analysis of Multibody Systems **6**
- 2.1 Lagrangian mechanics 6
- 2.2 Kinematics of rigid multibody systems 10
  - 2.2.1 Kinematics of a single rigid body 10
  - 2.2.2 Kinematics of a system of several rigid bodies 16
- 2.3 Kinetics of rigid multibody systems 17
  - 2.3.1 Kinetic energy of a single rigid body 17
  - 2.3.2 Generalized forces for a single rigid body 18
  - 2.3.3 Equations of motion for a single rigid body 19
  - 2.3.4 Equations of motion for multibody systems 21
- 2.4 Numerics 22
  - 2.4.1 Newmark discretization 22
  - 2.4.2 Hilber-Hughes-Taylor method 23

3 Adjoint Sensitivity Analysis of Multibody Systems **25**
- 3.1 System Sensitivity Analysis 25
  - 3.1.1 Illustrative example: Single degree of freedom oscillator . . 26
  - 3.1.2 Application to multibody system dynamics 29

3.2 Performance Measures 30
3.2.1 Illustrative example: Single degree of freedom oscillator . . 31
3.3 Adjoint Sensitivity Analysis 33
3.3.1 Basic considerations on the adjoint approach 33
3.3.2 An interpretation of the adjoint variables 34
3.3.3 Illustrative example: Single degree of freedom oscillator . . 35
3.3.4 Application to multibody system dynamics 37

4 Identification of Multibody System Parameters 45
4.1 Identification of Parameters using Measurements in Time Domain 47
4.2 Identification of Parameters using Measured Spectra 48
4.2.1 Remark on window functions 49
4.2.2 The adjoint gradient computation 50
4.2.3 Application to multibody systems 52
4.2.4 Summary: The Computation of the Gradient 54
4.3 Iterative Methods for Optimization 55
4.3.1 Determination of the search direction 57
4.3.2 Step size 60
4.4 Numerical Examples 65
4.4.1 Triple pendulum 66
4.4.2 Identification of torsional vibration damper parameters . . 72

5 Design of Optimal Inputs for Multibody Systems 81
5.1 Optimal Control Approach for Input Optimization 82
5.1.1 System sensitivity analysis 83
5.1.2 Maximization of the information content 84
5.1.3 Adoption of optimal control approach 85
5.1.4 Considering model input constraints 86
5.2 Parameter Identification 87
5.3 Illustrative examples 88
5.3.1 Optimal input for a linear system 89
5.3.2 Single mass oscillator 91
5.3.3 Two mass oscillator 92
5.3.4 Cart pendulum system 98

6 Conclusions and Outlook 102

Bibliography 104

CHAPTER 1

Introduction

Modern product development processes crave for efficient methods allowing to predict a system's performance prior to its initial operation. In the field of the design of mechanical components and systems understanding their dynamical behavior is vital and is typically analyzed using *multibody simulation* (MBS) software. These analyses include the chronology of system states for a given time period, or loads the system components are exposed to during operation. The practical effect of investing in the development of a so called *virtual prototype* strongly depends on the level of detail featured by the simulation results gathered. Often, unknown or imprecise parameters cause a deviation between simulation results and test bench samples. Hence, optimizing the quality of virtual prototypes is only possible by balancing the parameters of the simulation model with the real prototype. With an increasing number of parameters an automatized balancing method represents the only possibility for reaching improved simulation results.

This work therefore aims for the derivation of approaches enabling an automated identification process. Due to reasons concerning the performance and the applicability to multibody systems, the adjoint method is proposed as comprehensive approach. Throughout the work tailored adaptations of the adjoint concept will be made in order to find efficient solutions ready for the implementation in modern MBS software.

In practice, the achieved quality of parameters depends on the measurements taken during the operation of the real prototype. In optimal input design this issue is addressed by maximizing the information about a system's sensitivity to parameter changes.

1.1 Literature Review

1.1.1 Multibody dynamics

Multibody dynamics represents one of the younger domains in natural sciences. However, the concepts and theory behind rely on a solid base. The dynamics of a system, consisting out of several bodies, basically is described by equations of motion based on *Lagrangian mechanics*, that some sort of include *Newtonian mechanics*. A good overview of classical mechanics is given in the work of Goldstein [26]. The historical development of multibody dynamics itself is given in the work of Schiehlen [58]. As the importance of multibody dynamics has grown steadily, many textbooks were published in order to allow the integration into academic courses. The first textbook available was written by Nikravesh [52]. Other well-known textbooks originate from Shabana [60], Bauchau [5] and Bremer [10], where also flexible bodies are considered.

1.1.2 Adjoint method

The adjoint method is probably the most efficient way to solve a variety of optimization problems in engineering sciences. Much attention to this approach has been paid recently in the context of continuous systems for sensitivity analysis (see, e.g., [30, 29, 8, 19]). The class of dynamic programming methods for the computation of gradients in an optimization problem, includes the adjoint method which has a long history in optimal control theory [40]. The adjoint method is applied in, e.g., aerodynamic shape optimization by Jameson [33], in which the gradient of an objective function is determined indirectly by solving an adjoint equation which has coefficients determined by the solution of the multibody dynamics equations. This directly corresponds to the gradient technique for trajectory optimization pioneered by Bryson and Ho [12]. Once the gradient has been calculated, a descent method can be used to determine the optimal parameters or controls. The fast calculation of the gradients makes optimization computationally feasible even for designs in complex three-dimensional multibody systems. For this purpose, the equations of motion of the multibody system and adjoint equations may either be separately discretized from their representations as differential-algebraic equations, or, alternatively, the equations of motion of the multibody system may be discretized first, and the discrete adjoint equations are then derived directly from the discrete multibody equations [12]. In the field of aerodynamic design optimization using the adjoint formulation, the works by Anderson and Venkatakrishnan [2] and Nadarajah and Jameson [49] have to be mentioned. The work of Oberai et al. [53] shows the solution of inverse problems in elasticity imaging using the adjoint method. There, a straightforward calculation of the gradient requires N forward solutions of the elasticity problem. This cost is computationally prohibitive for typical values of N ($>= 10^3$). To circumvent this difficulty, [53] presents a new algorithm based on the adjoint elasticity operator which requires two solutions only (independent of N) to compute the gradient. Adjoint methods have also been the subject of studies in fluid dynamics research, as e.g., in the work by McNamara et al. [41], in which the adjoint method is used for controlling fluid simulations through gradient-based nonlinear optimization. Taylor et al. [65] present a hybrid adjoint approach applied to turbulent flow simulations.

Previous work on the adjoint method in multibody dynamics can be found in the work of Bottasso et al. [9], where the solution of inverse dynamics and trajectory optimization problems for multibody systems is reflected by an indirect approach combining optimal control theory with control and adjoint equations and transversality conditions. The inverse solution methodology presented in [9] has been implemented in the general purpose multibody code ADAMS. Within this implementation, some simple problems that provide a reasonable test and proof of concept of the presented methodology have been investigated. The design of the indirect method for solving optimal control problems for multibody systems presented by Bertolazzi et al. [6] seems to be familiar with the idea of the adjoint method. The work by Schaffer [57] presents a numerical algorithm, the piecewise adjoint method, which formulates the coordinate partitioning underlying ordinary differential equations as a boundary value problem, which is solved by multiple shooting methods. Additionally, convergence analysis of backward differentiation formulas is performed for stabilized differential-algebraic equations of motion in index 1 form and as well for the adjoint differential-algebraic equations for Cartesian non-centroidal multibody systems. Numerical studies in [57] compare the direct differentiation method, the adjoint method and the piecewise adjoint method for a slider-crank mechanism and a high mobility wheeled vehicle which revealed the speed-up of multibody systems with a small number of degrees of freedom and the potential speed-up for larger problems are discussed as well.

The group around Petzold, Cao, Li and Serban [54, 13, 14] describes forward and adjoint methods for sensitivity analysis for differential-algebraic equations and partial differential equations and state that the results of sensitivity analysis have wide-ranging applications in science and engineering, including model development, optimization, parameter estimation, model simplification, data assimilation, optimal control, uncertainty analysis and experimental design [54]. In the work of Eberhard [19], the adjoint method is used for sensitivity analysis in multibody systems interpreted as a continuous, hybrid form of automatic differentiation.

In case of an orientation parametrization of a body in multibody dynamics without using angles, as e.g. described in the absolute nodal coordinate formulation (ANCF) [59], a gradient-based optimization approach using adjoint equations has been presented by Held and Seifried [31]. Here, two different objective functions are defined to optimize a flexible slider-crank mechanism. One criterion accounts for the deformation energy of the flexible body and the second criterion accumulates the squared deviation of the actual and desired position of the slider block. The adjoint method is then derived for the sensitivity analyses of the different objective functions [31]. Due to the structure of the objective functions and the fact that the ANCF includes a constant mass matrix with vanishing derivative, the equations reduce to a simpler form [31, Eq. (19)]. The framework of the ANCF is as well used in the sensitivity analysis using the adjoint method by Pi et al. [55] within a first order approach, while in [17] a second order adjoint sensitivity analysis of multibody systems has been presented within the classical multibody formulation. The optimization strategies employing second order sensitivity information shows higher accuracy, with the drawback of its complex structure. However, the adjoint method is utilized in [17] and shows a substantial reduction of computational costs in case of a large number of design variables. The comparision of the adjoint method to the direct differentiation is given in Table 4 in [17] method.

1.2 Thesis Objectives

Despite of the great potential of the adjoint sensitivity analysis, it is rarely applied in multibody dynamics, since the sophisticated structure of the equations of motion, and the effort to obtain the set of adjoint equations seems tremendously high. Hence, dealing with the adjoint approach is obviously unattractive for most developers of multibody simulation software. Therefore, the main goal of this thesis is to illustrate how the adjoint sensitivity analysis may be embedded in state of the art multibody system descriptions. First, the sensitivity with respect to system parameters is computed in order to identify them based on measurements. Second, a performance measure quantifying the information about parameters contained in measurements is developed and its variation is derived by using the adjoint sensitivity analysis. In order to verify the methods, several examples are covered within the particular sections.

1.3 Thesis Structure

The thesis is organized as follows: In Chapter 2 the equations necessary to describe the dynamical behavior of a multibody system are derived. Although these equations are well known and documented in the literature, this chapter helps in understanding the subsequent derivations and introduces the nomenclature adopted. Analyzing a system's sensitivity with respect to its parameters is treated in Chapter 3. First, the general approach using the direct differentiation method is applied on the multibody system. As already mentioned in the review of the existing literature, the adjoint approach is known to reduce the computational effort for sensitivity analysis. Therefore in Section 3.3.4 this approach is adopted for the use in multibody system dynamics. The derivation of an identification method for parameters is carried out in Chapter 4. In Section 4.2 special emphasis is put on performance measures including frequency domain information in form of measured spectra, which represents a novel approach in the field of multibody system dynamics. The topic of optimal input design discussed in Chapter 5 thematically completes the thesis by investigating the planning of experiments used for parameter identification.

CHAPTER 2

Dynamic Analysis of Multibody Systems

2.1 Lagrangian mechanics

In this section we start with a brief review of the principles of Lagrangian mechanics which are used in the subsequent sections to derive the equations of motion of multibody systems. In the Lagrangian approach the configuration of a mechanical system is described by a set of generalized coordinates $\mathbf{q} = (q_1, \dots, q_n)^\mathsf{T}$. The basic problem is to find a set of equations which describe the time history $\mathbf{q}(t)$ of this generalized coordinates.

Following [26], we start from Newton's second law for a material point m_K

$$m_K \ddot{\vec{r}}_K = \vec{f}_K, \tag{2.1}$$

where $\vec{r}_K(t)$ denotes the position of m_K in an inertial frame. In Lagrangian mechanics $\vec{r}_K$ is described by a function of the generalized coordinates and eventually explicitly of time

$$\vec{r}_K = \vec{r}_K(q_1, \dots, q_n; t). \tag{2.2}$$

Moreover in Eq. (2.1) $\vec{f}_K$ denotes the total force acting on m_K. Forces on m_K which can be expressed explicitly from the generalized coordinates are denoted as impressed forces $\vec{f}_K^{(e)}$, whereas forces which are associated with geometric restrictions of the particle's motion and which cannot be described by an explicit law are denoted as constraint forces $\vec{f}_K^{(c)}$. These forces can be eliminated with D'Alembert's principle which states that the virtual work of all constraint forces in a mechanical system resulting from a virtual variation

$$\delta\vec{r}_K = \sum_{i=1}^{n} \frac{\partial \vec{r}_K}{\partial q_i}\delta q_i \tag{2.3}$$

is zero for arbitrary variations δq_i of the generalized coordinates. Hence, multiplying Eq. (2.1) with $\delta\vec{r}_K$ and summing up over all particles yields

$$\sum_{K=1}^{N} m_K \ddot{\vec{r}}_K \delta\vec{r}_K = \sum_{K=1}^{N} \left(\vec{f}_K^{(e)} + \vec{f}_K^{(c)}\right)\delta\vec{r}_K = \sum_{K=1}^{N} \vec{f}_K^{(e)}\delta\vec{r}_K \tag{2.4}$$

since the virtual work of the constraint forces $\delta W^{(c)} = \sum_{K=1}^{N} \vec{f}_K^{(c)}\delta\vec{r}_K$ is zero due to D'Alembert's principle. The term

$$\delta W^{(e)} = \sum_{K=1}^{N} \vec{f}_K^{(e)}\delta\vec{r}_K = \sum_{K=1}^{N}\left(\vec{f}_K^{(e)} \sum_{i=1}^{n} \frac{\partial \vec{r}_K}{\partial q_i}\delta q_i\right)$$

is the virtual work $\delta W^{(e)}$ of the impressed forces. After rearranging the sums we obtain

$$\delta W^{(e)} = \sum_{i=1}^{n}\left\{\sum_{K=1}^{N}\left(\vec{f}_K^{(e)}\frac{\partial \vec{r}_K}{\partial q_i}\right)\right\}\delta q_i := \sum_{i=1}^{n} Q_i\delta q_i,$$

where the generalized force

$$Q_i := \sum_{K=1}^{N}\left(\vec{f}_K^{(e)}\frac{\partial \vec{r}_K}{\partial q_i}\right) \tag{2.5}$$

associated to the coordinate q_i is introduced.

In order to relate Q_i with the inertia part in Eq. (2.4), the left side of Equation (2.4) must also be written in the form

$$\sum_{K=1}^{N} m_K \ddot{\vec{r}}_K \delta\vec{r}_K = \sum_{i=1}^{n} X_i\delta q_i.$$

Substituting $\delta\vec{r}_K$ from Eq. (2.3) yields

$$\sum_{K=1}^{N} m_K \ddot{\vec{r}}_K \delta\vec{r}_K = \sum_{K=1}^{N} m_K \ddot{\vec{r}}_K \sum_{i=1}^{n}\frac{\partial \vec{r}_K}{\partial q_i}\delta q_i = \sum_{i=1}^{n}\left\{\sum_{K=1}^{N} m_K \ddot{\vec{r}}_K \frac{\partial \vec{r}_K}{\partial q_i}\right\}\delta q_i$$

and obviously the quantity X_i is found to be

$$X_i = \sum_{K=1}^{N} m_K \ddot{\vec{r}}_K \frac{\partial \vec{r}_K}{\partial q_i}.$$

As X_i cannot be computed in this form the product rule is applied, which results in

$$\begin{aligned} X_i &= \sum_{K=1}^{N} m_K \ddot{\vec{r}}_K \frac{\partial \vec{r}_K}{\partial q_i} \\ &= \sum_{K=1}^{N} \left\{ \frac{\mathrm{d}}{\mathrm{d}t} \left(m_K \dot{\vec{r}}_K \frac{\partial \vec{r}_K}{\partial q_i} \right) - m_K \dot{\vec{r}}_K \frac{\mathrm{d}}{\mathrm{d}t} \frac{\partial \vec{r}_K}{\partial q_i} \right\} \\ &= \frac{\mathrm{d}}{\mathrm{d}t} \sum_{K=1}^{N} \left(m_K \vec{v}_K \frac{\partial \vec{r}_K}{\partial q_i} \right) - \sum_{K=1}^{N} \left(m_K \vec{v}_K \frac{\mathrm{d}}{\mathrm{d}t} \frac{\partial \vec{r}_K}{\partial q_i} \right). \end{aligned} \tag{2.6}$$

The velocity $\vec{v}_K$ can be derived from Eq. (2.2) by performing the derivation with respect to time, reading

$$\vec{v}_K = \dot{\vec{r}}_K = \sum_{j=1}^{n} \frac{\partial \vec{r}_K}{\partial q_j} \dot{q}_j + \frac{\partial \vec{r}_K}{\partial t}. \tag{2.7}$$

Having a closer look on Eq. (2.7) allows to find

$$\frac{\partial \vec{v}_K}{\partial \dot{q}_i} = \frac{\partial \vec{r}_K}{\partial q_i}. \tag{2.8}$$

Inserting this in Eq. (2.6) and using

$$\frac{\mathrm{d}}{\mathrm{d}t} \frac{\partial \vec{v}_K}{\partial q_i} = \sum_{j=1}^{n} \frac{\partial^2 \vec{r}_K}{\partial q_j \partial q_i} \dot{q}_i + \frac{\partial^2 \vec{r}_K}{\partial q_j \partial t}$$

which is equal to

$$\frac{\partial \vec{r}_K}{\partial q_i} = \sum_{j=1}^{n} \frac{\partial^2 \vec{r}_K}{\partial q_i \partial q_j} \dot{q}_j + \frac{\partial^2 \vec{r}_K}{\partial t \partial q_j}$$

results in

$$\begin{aligned} X_i &= \frac{\mathrm{d}}{\mathrm{d}t} \sum_{K=1}^{N} \left(m_K \vec{v}_K \frac{\partial \vec{v}_K}{\partial \dot{q}_i} \right) - \sum_{K=1}^{N} \left(m_K \vec{v}_K \frac{\partial \vec{v}_K}{\partial q_i} \right) \\ &= \frac{\mathrm{d}}{\mathrm{d}t} \left\{ \frac{\partial}{\partial \dot{q}_i} \sum_{K=1}^{N} \left(\frac{m_K}{2} v_K^2 \right) \right\} - \frac{\partial}{\partial q_i} \left(\sum_{K=1}^{N} \frac{m_K}{2} v_K^2 \right). \end{aligned} \tag{2.9}$$

In both terms of Eq. (2.9) the kinetic energy

$$T = \frac{1}{2} \sum_{K=1}^{N} m_K v_K^2 \tag{2.10}$$

can be substituted and finally Eq. (2.4) reads

$$\sum_{i=1}^{n}\left\{\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial T}{\partial \dot{q}_i}\right)-\frac{\partial T}{\partial q_i}\right\}\delta q_i=\sum_{i=1}^{n}Q_i\delta q_i, \tag{2.11}$$

which can be written also in matrix notation in the form

$$\delta\mathbf{q}^{\mathsf{T}}\left\{\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial T}{\partial \dot{\mathbf{q}}}\right)-\frac{\partial T}{\partial \mathbf{q}}\right\}=\delta\mathbf{q}^{\mathsf{T}}\mathbf{Q}. \tag{2.12}$$

We may now distinguish between minimal sets of generalized coordinates, which are not constrained by additional geometric relations and redundant sets of generalized coordinates, which are not independent but constrained by equations of the form $\mathbf{C}(\mathbf{q})=(C_1(\mathbf{q}),\ldots,C_m(\mathbf{q}))^{\mathsf{T}}=\mathbf{0}$ (only holonomic constraint equations are considered in this work).
In the first case Eq. (2.12) holds for arbitrary variations $\delta\mathbf{q}$ and hence, the equations of motion for $\mathbf{q}(t)$ are given by

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial T}{\partial \dot{\mathbf{q}}}\right)-\frac{\partial T}{\partial \mathbf{q}}=\mathbf{Q}.$$

However, in the latter case the variations $\delta\mathbf{q}$ are also constrained by the relation

$$\frac{\partial \mathbf{C}}{\partial \mathbf{q}}\delta\mathbf{q}=\mathbf{C}_{\mathbf{q}}\delta\mathbf{q}=\mathbf{0}. \tag{2.13}$$

But now we can add in Eq. (2.12) the zero term

$$\lambda^{\mathsf{T}}\mathbf{C}_{\mathbf{q}}\delta\mathbf{q}=\delta\mathbf{q}^{\mathsf{T}}\mathbf{C}_{\mathbf{q}}^{\mathsf{T}}\lambda=0$$

where $\lambda=(\lambda_1,\ldots,\lambda_m)^{\mathsf{T}}$ is a vector of arbitrary Lagrange multipliers, yielding

$$\delta\mathbf{q}^{\mathsf{T}}\left\{\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial T}{\partial \dot{\mathbf{q}}}\right)-\frac{\partial T}{\partial \mathbf{q}}+\mathbf{C}_{\mathbf{q}}^{\mathsf{T}}\lambda\right\}=\delta\mathbf{q}^{\mathsf{T}}\mathbf{Q}.$$

As we have $n-m$ independent variations in the vector $\delta\mathbf{q}$ and m arbitrary Lagrange multipliers in λ we may now equate the expressions multiplied with $\delta\mathbf{q}^{\mathsf{T}}$ on both sides. Extending these equations of motion with the constraint equations gives a set of second-order differential-algebraic equations (second-order DAE)

$$\begin{gathered}\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial T}{\partial \dot{\mathbf{q}}}\right)-\frac{\partial T}{\partial \mathbf{q}}=\mathbf{Q}-\mathbf{C}_{\mathbf{q}}^{\mathsf{T}}\lambda\\ \mathbf{C}(\mathbf{q})=\mathbf{0}\end{gathered} \tag{2.14}$$

which have to be solved for $\mathbf{q}(t)$ and $\lambda(t)$ in the case of a system described by a redundant generalized coordinates. This formulation is used in many multibody simulation packages and will serve as the basis for the subsequent discussions, too.

2.2 Kinematics of rigid multibody systems

2.2.1 Kinematics of a single rigid body

In multibody systems rigid bodies represent the smallest unit and therefore their kinematics has to be described in detail. In kinematics the body's motion can be expressed by geometric relations by just using the quantities time, position, velocity and acceleration. This section discusses the approaches typically used for describing the motion of an unconstrained rigid body in three-dimensional space with six degrees of freedom. The notation used here is taken from [3].

2.2.1.1 Body-fixed coordinate system and generalized coordinates

In rigid body kinematics the motion of any material point P may be expressed by the use of a body-fixed coordinate system as shown in Fig. 2.1. The relation

$$\vec{r} = \vec{u} + \mathbf{B}\vec{R}$$

gives the transformation from local coordinates $\vec{R}$ with respect to the body-fixed coordinate system to global coordinates $\vec{r}$ with respect to the inertial frame. The rotation matrix

$$\mathbf{B} = \left(\vec{b}_x, \vec{b}_y, \vec{b}_z\right) \tag{2.15}$$

assembles all three orientation vectors describing the axes of the body-fixed coordinate system. As the unit-vectors $\vec{b}_x$, $\vec{b}_y$ and $\vec{b}_z$ are perpendicular to each other, i.e.

$$\begin{aligned} \vec{b}_x \cdot \vec{b}_x = 1, \quad \vec{b}_y \cdot \vec{b}_y = 1, \quad \vec{b}_z \cdot \vec{b}_z = 1 \\ \vec{b}_x \cdot \vec{b}_y = 0, \quad \vec{b}_y \cdot \vec{b}_z = 0, \quad \vec{b}_x \cdot \vec{b}_z = 0 \end{aligned} \tag{2.16}$$

the rotation matrix is an orthogonal matrix satisfying the identities

$$\mathbf{B}^{\mathsf{T}}\mathbf{B} = \mathbf{I} \text{ or } \mathbf{B}^{-1} = \mathbf{B}^{\mathsf{T}}. \tag{2.17}$$

Here $\mathbf{I}$ denotes the identity matrix. For the definition of the body-fixed coordinate system the vectors $\vec{u}$, $\vec{b}_x$, $\vec{b}_y$ and $\vec{b}_z$ defined in an inertial frame are required. Hence, for a single rigid body we introduce a vector of translational generalized coordinates $\mathbf{q}^t = \vec{u}$ and a set of rotational generalized coordinates $\mathbf{q}^r$ which describe the vectors $\vec{b}_x$, $\vec{b}_y$, $\vec{b}_z$ forming the columns of the rotation matrix $\mathbf{B}$.

2.2.1.2 Parametrization of the rotation matrix

The nine components of $\mathbf{B}$ are constrained by six orthogonality relations Eq. (2.16). Therefore, the vector of rotational generalized coordinates $\mathbf{q}^r$ must contain at

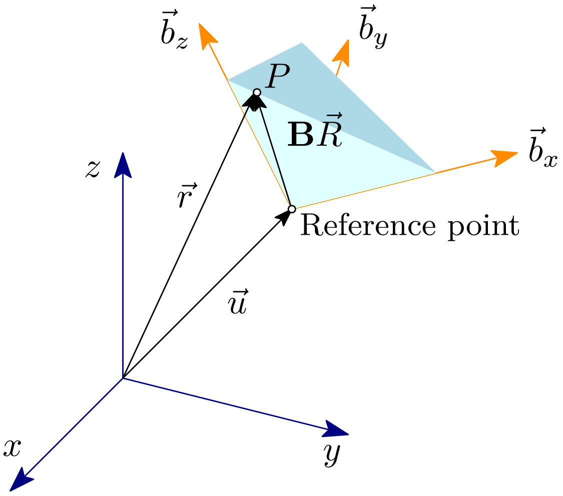


Figure 2.1: Position $\vec{r}$ of a material point P in the inertial frame given by $\vec{u} + \mathbf{B}\vec{R}$, where $\mathbf{B}$ is the rotation matrix of the body fixed reference frame $\vec{b}_x, \vec{b}_y, \vec{b}_z$ and $\vec{R}$ contains the coordinates of P with respect to the body fixed frame.

least three entries. There are different approaches for this parametrization given in the literature (see e.g., [60]). When considering *Euler angles* three successive rotations, first about the z-axis, then about the x-axis and again about the z-axis, are performed. In contrast to that, *Tait-Bryant angles* use rotations about the x-axis, the y-axis and the z-axis. Both parametrizations lead to singularities in some special configurations. This can be avoided, e.g. if a redundant set of four parameters, denoted as *Euler parameters*, is used.

For the subsequent discussion we focus on Euler parameters, since the equations of motion for a single rigid body can be obtained rather simply in this case. Before discussing Euler parameters, the rotation vector has to be introduced. In Fig. 2.2 the vector $\vec{r}_0$ is rotated about the axis defined by the unit vector $\vec{a}$. In order to compute the rotated vector $\vec{r}_1$, the vectors $\vec{b}_1$ and $\vec{b}_2$ are expressed as function of the circle radius ρ and the rotation angle θ. Inserting the circle radius

$$\rho = |\vec{a} \times \vec{r}_0|$$

in the expressions for the vectors $\vec{b}_1$ and $\vec{b}_2$

$$\begin{aligned} \vec{b}_1 &= \rho \sin\theta \, \vec{e}_1 \\ \vec{b}_2 &= (\rho - \rho\cos\theta) \, \vec{e}_2 \end{aligned}$$

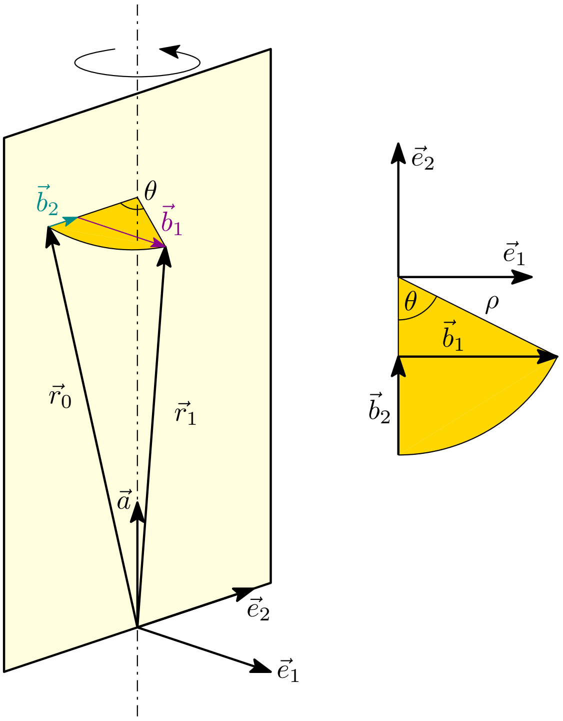


Figure 2.2: Rotation of vector $\vec{r}_0$ about angle θ

the unit vectors $\vec{e}_1$ and $\vec{e}_2$ form an orthogonal basis with the rotation axis $\vec{a}$ and therefore

$$\begin{aligned}\vec{r}_1 &= \vec{r}_0 + \vec{b}_1 + \vec{b}_2 \\ &= \vec{r}_0 + \rho \sin\theta \, \vec{e}_1 + (\rho - \rho\cos\theta)\, \vec{e}_2 \\ &= \vec{r}_0 + \sin\theta\,(\vec{a} \times \vec{r}_0) + (1 - \cos\theta)\,[\vec{a} \times (\vec{a} \times \vec{r}_0)]\,.\end{aligned} \tag{2.18}$$

In order to end up with a matrix equation for the rotation of $\vec{r}_0$, the skew symmetric matrix

$$\hat{a} = \begin{pmatrix} 0 & -a_z & a_y \\ a_z & 0 & -a_x \\ -a_y & a_x & 0 \end{pmatrix}$$

is introduced. By using this matrix instead of $\vec{a}$ the cross product can be replaced by a matrix product. Equation (2.18) therefore reads

$$\vec{r}_1 = \left[\mathbf{I} + \sin\theta\hat{a} + (1 - \cos\theta)\,\hat{a}^2\right]\vec{r}_0 = \mathbf{B}\vec{r}_0$$

with using the rotation matrix

$$\mathbf{B} = \mathbf{I} + \sin\theta\hat{a} + (1 - \cos\theta)\,\hat{a}^2. \tag{2.19}$$

When making use of Euler parameters a special parametrization of Eq. (2.19) is used. First, the trigonometric relations

$$\sin\theta = 2\sin\frac{\theta}{2}\cos\frac{\theta}{2} \quad \text{and} \quad 1-\cos\theta = 2\sin^2\frac{\theta}{2}$$

are inserted in Eq. (2.19). This leads to

$$\mathbf{B} = \mathbf{I} + 2\hat{a}\sin\frac{\theta}{2}\cos\frac{\theta}{2} + 2\hat{a}^2\ \sin^2\frac{\theta}{2}.$$

By using a different skew symmetric matrix $\hat{e}$

$$\hat{e} = \hat{a}\sin\frac{\theta}{2} = \begin{pmatrix} 0 & -a_z & a_y \\ a_z & 0 & -a_x \\ -a_y & a_x & 0 \end{pmatrix} \sin\frac{\theta}{2} = \begin{pmatrix} 0 & -e_3 & e_2 \\ e_1 & 0 & -e_1 \\ -e_2 & e_1 & 0 \end{pmatrix}$$

and

$$e_0 = \cos\frac{\theta}{2}$$

the rotation matrix becomes

$$\mathbf{B} = \mathbf{I} + 2\hat{e}\ e_0 + 2\hat{e}^2 = \begin{pmatrix} 1-2e_2^2-2e_3^2 & 2e_1e_2-2e_0e_3 & 2e_1e_3+2e_0e_2 \\ 2e_1e_2+2e_0e_3 & 1-2e_1^2-2e_3^2 & 2e_2e_3-2e_0e_1 \\ 2e_1e_3-2e_0e_2 & 2e_2e_3+2e_0e_1 & 1-2e_1^2-2e_2^2 \end{pmatrix}. \tag{2.20}$$

The four parameters

$$e_0 = \cos\frac{\theta}{2}, \quad e_1 = a_x\sin\frac{\theta}{2}, \quad e_2 = a_y\sin\frac{\theta}{2}, \quad e_3 = a_z\sin\frac{\theta}{2}, \tag{2.21}$$

are called Euler parameters. In contrast to Euler angles or Tait-Bryant angles, **B** can be computed without evaluating trigonometric functions. Another advantage of using Euler parameters is the absence of singularities and therefore the numeric stability. One disadvantage is that four parameters are used to described three degrees of freedom and therefore an additional constraint equation has to be taken into account. Due to the unit vector $\vec{a}$ and the identity $\sin^2\alpha + cos^2\alpha = 1$ the constraint equation

$$e_0^2 + e_1^2 + e_2^2 + e_3^2 = 1$$

must hold in all configurations of **B**. Inserting this in Eq. (2.20), the rotation matrix reads

$$\mathbf{B} = \begin{pmatrix} e_0^2+e_1^2-e_2^2-e_3^2 & 2e_1e_2-2e_0e_3 & 2e_1e_3+2e_0e_2 \\ 2e_1e_2+2e_0e_3 & e_0^2+e_2^2-e_1^2-e_3^2 & 2e_2e_3-2e_0e_1 \\ 2e_1e_3-2e_0e_2 & 2e_2e_3+2e_0e_1 & e_0^2+e_3^2-e_1^2-e_2^2 \end{pmatrix}.$$

2.2.1.3 Velocities

For computing the kinetic energy required for Lagrange's equations, the velocity of material points must be considered in kinetics. Differentiation of $\vec{r}$ with respect to the time

$$\vec{v} = \dot{\vec{u}} + \dot{\mathbf{B}}\vec{R} \tag{2.22}$$

gives the velocity $\dot{\vec{r}} = \vec{v}$. The local coordinate $\vec{R}$ has no time derivative since only rigid bodies are investigated. It is very common to use the origin of the body-fixed coordinate system for describing the translational degree of freedom of a rigid body. Due to this the velocity $\dot{\vec{u}} = \dot{\mathbf{q}}^t$ will directly appear in the equations of motion and thus is not considered in detail. However, the time derivative of the rotation matrix $\mathbf{B}$ needs to be investigated. First, the differentiation is performed by applying the chain rule

$$\dot{\mathbf{B}} = \sum_{i=1}^{n_r} \frac{\partial \mathbf{B}}{\partial q_i^r} \dot{q}_i^r$$

where n_r is the number of generalized rotational coordinates $\mathbf{q}^r$. Due to Eq. (2.17) the time derivative

$$\mathbf{B}^{\mathsf{T}}\dot{\mathbf{B}} + \dot{\mathbf{B}}^{\mathsf{T}}\mathbf{B} = \mathbf{0}$$

must hold. This gives

$$\mathbf{B}^{\mathsf{T}}\dot{\mathbf{B}} = -\left(\mathbf{B}^{\mathsf{T}}\dot{\mathbf{B}}\right)^{\mathsf{T}}$$

and allows to introduce the skew symmetric matrix $\hat{\Omega}$ in the form

$$\hat{\Omega} = \mathbf{B}^{\mathsf{T}}\dot{\mathbf{B}}. \tag{2.23}$$

Using $\dot{\mathbf{B}} = \mathbf{B}\vec{\Omega}$ in Eq. (2.22) and bearing in mind that the product of a skew symmetric matrix and a vector corresponds with the cross product of two vectors leads to

$$\vec{v} = \dot{\mathbf{q}}^t + \mathbf{B}\hat{\Omega}\vec{R} = \dot{\mathbf{q}}^t + \mathbf{B}\left(\vec{\Omega} \times \vec{R}\right). \tag{2.24}$$

This relates $\hat{\Omega}$ with the vector $\vec{\Omega}$ commonly denoted as angular velocity vector. For the usage in the equations of motion the relation between the time derivatives of the rotational parameters $\dot{\mathbf{q}}^r$ and the angular velocity vector is desirable. For this purpose the matrix $\mathbf{G}^i$, which allows to rewrite Eq. (2.23) into

$$\hat{\Omega} = \mathbf{B}^{\mathsf{T}}\dot{\mathbf{B}} = \sum_{i=1}^{n_r} \mathbf{B}^{\mathsf{T}} \frac{\partial \mathbf{B}}{\partial q_i^r} \dot{q}_i^r = \sum_{i=1}^{n_r} \mathbf{G}^i \dot{q}_i^r, \tag{2.25}$$

is introduced. Due to the skew symmetric property of $\hat{\Omega}$, only the elements of $\mathbf{G}^i$ with positive sign $\mathbf{G}^i_{3,2}$, $\mathbf{G}^i_{1,3}$ and $\mathbf{G}^i_{2,1}$ are necessary to compute $\vec{\Omega}$. Introducing the vectors

$$\vec{G}^i = \left(G^i_{3,2}, G^i_{1,3}, G^i_{2,1}\right)^{\mathsf{T}}$$

and joining them allows to write Eq. (2.25) in vectorial form

$$\vec{\Omega} = \sum_{i=1}^{n_r} \vec{G}^i \dot{q}_i^r = \left(\vec{G}^1 \dots \vec{G}^{n_r} \right) \begin{pmatrix} \dot{q}_1^r \\ \vdots \\ \dot{q}_{n_r}^r \end{pmatrix} = \mathbf{G}\dot{\mathbf{q}}^r \tag{2.26}$$

where $\mathbf{G}$ is the mapping matrix between the time derivatives of the rotational parameters and the angular velocity vector $\vec{\Omega}$.

Depending on the parametrization of the rotation matrix chosen, the matrix $\mathbf{G}$ is generated by computing the term $\mathbf{G}^i$, picking the three components for $\vec{G}^i$ and finally assembling them in the matrix $\mathbf{G}$. In case of Euler parameters the first column of $\mathbf{G}$ is made up of the elements of

$$\mathbf{G}^i = \mathbf{B}^{\mathsf{T}} \frac{\partial \mathbf{B}}{\partial e_0} = 2 \begin{pmatrix} e_0 & e_3 & -e_2 \\ -e_3 & e_0 & e_1 \\ e_2 & -e_1 & e_0 \end{pmatrix}$$

and similarly for the remaining columns. Finally the matrix $\mathbf{G}$ can be presented as

$$\mathbf{G} = 2 \begin{pmatrix} -e_1 & e_0 & e_3 & -e_2 \\ -e_2 & -e_3 & e_0 & e_1 \\ -e_3 & e_2 & -e_1 & e_0 \end{pmatrix}.$$

Substituting Eq. (2.26) in Eq. (2.24) and permuting the order of the cross product gives the handy expression

$$\vec{v} = \dot{\mathbf{q}}^t - \mathbf{B} \left(\vec{R} \times \mathbf{G}\dot{\mathbf{q}}^r \right)$$

which can be used for the efficient computation of a material point's velocity vector by matrix calculus. Furthermore the expression can be rearranged to

$$\vec{v} = \dot{\mathbf{q}}^t - \mathbf{B}\hat{R}\mathbf{G}\dot{\mathbf{q}}^r \tag{2.27}$$

where the skew-symmetric matrix

$$\hat{R} = \begin{pmatrix} 0 & -R_z & R_y \\ R_z & 0 & -R_x \\ -R_y & R_x & 0 \end{pmatrix}$$

is introduced. Due to this the cross product can be replaced with a matrix multiplication. The components of $\hat{R}$ are given by the vector elements $\vec{R} = (R_x, R_y, R_z)^{\mathsf{T}}$.

2.2.2 Kinematics of a system of several rigid bodies

Multibody systems are made up of several rigid bodies, which can be interconnected by different types of joints, such as hinges, linear guides, radial bearings, etc. Describing the kinematics for the entire system is generally speaking a non-trivial task. Doing so would involve finding a minimal set of coordinates, which is not possible for a computer program in a systematic manner. One possibility to circumvent seeking for those coordinates is to consider the kinematics of every rigid body with all it's degrees of freedom. The constraints resulting from joining the system's bodies are introduced by a set of algebraic equations $\mathbf{C}$.
Another practicable method is trying to find a quasi-minimal set of coordinates by applying recursive kinematics. This means, that only such degrees of freedom are introduced, that allow the system to fulfill all constraints. Basically this is only possible for systems with chain and tree topology. In case of other topologies additional constraint equations must be introduced. Due to the chosen set of coordinates these equations are of very complex structure and may involve coordinates which are not directly part of the physical constraint.
In order to illustrate how such constraints may be described in terms of multibody systems, in the following the revolute joint will be presented as an example. The physical realization thereof can be a combined radial and axial bearing of a shaft or a hinge used for mounting car doors, etc. Other representatives would be the planar, cylindrical or the spherical joint.
Before looking after the equations representing the joint's geometrical specifications, first the degrees of freedom have to be clarified. As it can be seen in Fig. 2.3 the two bodies $\mathcal{B}_1$ and $\mathcal{B}_2$ are interconnected with a hinge. This only allows the two bodies to rotate about $\vec{A}$ as the translation along the axis is locked too. Due to this the number of degrees of freedom is six for one free body plus one for the connected second body. As there are twelve degrees of freedom in the unconstrained case and seven in the constrained one, this means the joint constrains five degree of freedom. Due to this, five equations have to be found. Three equations can be formed by comparing the position of point P in global coordinates. Application of the according coordinate transformation on each body and setting them equal reads

$$\vec{u}_1 + \mathbf{B}_1 \vec{R}_P^{(1)} - \left(\vec{u}_2 + \mathbf{B}_2 \vec{R}_P^{(2)}\right) = \mathbf{0}. \tag{2.28}$$

Here, the vectors to the particular coordinate system origin $\vec{u}_1$ and $\vec{u}_2$, the rotation matrices $\mathbf{B}_1$ and $\mathbf{B}_2$ and the position P in local coordinates $\vec{R}_P^{(1)}$ and $\vec{R}_P^{(2)}$ are used.
The remaining two equations are formed by demanding

$$\begin{aligned} \left(\mathbf{B}_2 \vec{G}^{\,(2)}\right) \cdot \left(\mathbf{B}_1 \vec{A}^{\,(1)}\right) &= 0 \\ \left(\mathbf{B}_2 \vec{H}^{\,(2)}\right) \cdot \left(\mathbf{B}_1 \vec{A}^{\,(1)}\right) &= 0 \end{aligned} \tag{2.29}$$

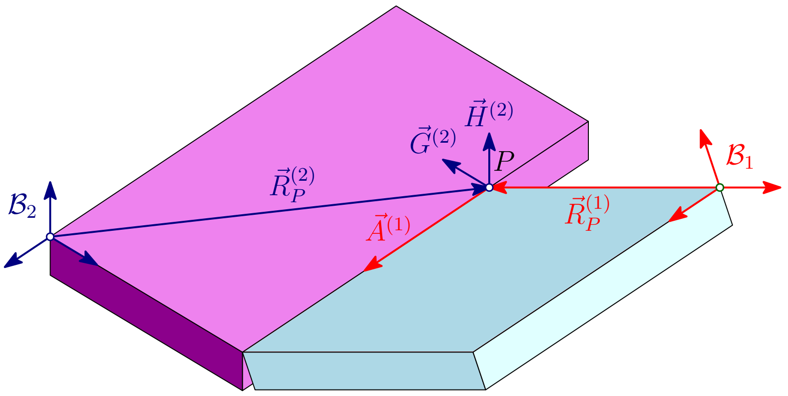


Figure 2.3: Sketch of revolute joint

which means both vectors $\vec{G}^{(2)}$ and $\vec{H}^{(2)}$, described in the body fixed coordinate system $\mathcal{B}_2$, are perpendicular to the rotation axis $\vec{A}^{(1)}$, described in $\mathcal{B}_1$. By using those five algebraic equations, Eq. (2.28) and Eq. (2.29) respectively, both bodies are constrained to the desired rotational degree of freedom.

2.3 Kinetics of rigid multibody systems

2.3.1 Kinetic energy of a single rigid body

For a solid body the sum in Eq. (2.10) can be replaced by an integral. The kinetic energy then results in

$$T = \frac{1}{2}\int_m v^2 \,\mathrm{d}m \tag{2.30}$$

for the body's mass m. Due to the orthogonality of the rotation matrix $\mathbf{B}^\mathsf{T}\mathbf{B} = \mathbf{I}$ the squared velocity

$$\begin{aligned} v^2 &= \left(\dot{\vec{u}} - \mathbf{B}\hat{R}\vec{\Omega}\right)^\mathsf{T}\left(\dot{\vec{u}} - \mathbf{B}\hat{R}\vec{\Omega}\right) \\ &= \dot{\vec{u}}^\mathsf{T}\dot{\vec{u}} - 2\dot{\vec{u}}\,\mathbf{B}\hat{R}\vec{\Omega} + \vec{\Omega}^\mathsf{T}\hat{R}^\mathsf{T}\mathbf{B}^\mathsf{T}\mathbf{B}\hat{R}\vec{\Omega} \end{aligned}$$

can be written as

$$v^2 = \dot{\vec{u}}^\mathsf{T}\dot{\vec{u}} - 2\dot{\vec{u}}\,\mathbf{B}\hat{R}\vec{\Omega} + \vec{\Omega}^\mathsf{T}\hat{R}^\mathsf{T}\hat{R}\vec{\Omega}.$$

Now the kinetic energy can be computed by inserting v^2 in Eq. (2.30)

$$T = \frac{1}{2}\dot{\vec{u}}^\mathsf{T}\dot{\vec{u}}\int_m \mathrm{d}m - 2\dot{\vec{u}}\,\mathbf{B}\left(\int_m \hat{R}\mathrm{d}m\right)\vec{\Omega} + \vec{\Omega}^\mathsf{T}\left(\int_m \hat{R}^\mathsf{T}\hat{R}\,\mathrm{d}m\right)\vec{\Omega} \tag{2.31}$$

where here all terms, constant for the entire volume, are extracted from the integral. Obviously the integral in the first term equates to the mass m itself. If the center of mass is used for the body-fixed coordinate system's origin, the second term vanishes. The integral

$$\mathbf{I}_M = \int_m \hat{R}^\mathsf{T} \hat{R} \, \mathrm{d}m$$

is denoted as the tensor of inertia and can be evaluated in a straight forward way for rigid bodies. Using this, Eq. (2.31) reads

$$T = \frac{1}{2} m \vec{u}^\mathsf{T} \vec{u} + \frac{1}{2} \vec{\Omega}^\mathsf{T} \mathbf{I}_M \vec{\Omega}.$$

Applying the findings of section 2.2.1.3, the kinetic energy for one rigid body reads

$$T = \frac{1}{2} m \left(\dot{\mathbf{q}}^t\right)^\mathsf{T} \dot{\mathbf{q}}^t + \frac{1}{2} \left(\dot{\mathbf{q}}^r\right)^\mathsf{T} \mathbf{G}^\mathsf{T} \mathbf{I}_M \mathbf{G} \dot{\mathbf{q}}^r.$$

Commonly the kinetic energy is written in terms of the mass matrix,

$$\mathbf{M} = \begin{pmatrix} m\mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{G}^\mathsf{T} \mathbf{I}_M \mathbf{G} \end{pmatrix}$$

which allows T to be computed as

$$T = \frac{1}{2} \left(\dot{\mathbf{q}}\right)^\mathsf{T} \mathbf{M} \dot{\mathbf{q}} \tag{2.32}$$

where the generalized coordinates are given as $\mathbf{q} = \left(\mathbf{q}^t, \mathbf{q}^r\right)^\mathsf{T}$.

2.3.2 Generalized forces for a single rigid body

The equations of motion introduced in section 2.1 employ generalized forces Q_i. Equation (2.5) incorporates the partial derivatives $\partial \vec{r}_K / \partial q_i$, which can be replaced by $\partial \dot{\vec{r}}_K / \partial \dot{q}_i$ due to the equality shown in Eq. (2.8). Performing the differentiation of Eq. (2.27) gives

$$\frac{\partial \vec{v}}{\partial \dot{\mathbf{q}}^t} = \frac{\partial}{\partial \dot{\mathbf{q}}^t} \left(\dot{\mathbf{q}}^t - \mathbf{B} \hat{R} \mathbf{G} \dot{\mathbf{q}}^r\right) = \mathbf{I}$$

for the translational generalized coordinates $\mathbf{q}^t$ and

$$\frac{\partial \vec{v}}{\partial \dot{\mathbf{q}}^r} = \frac{\partial}{\partial \dot{\mathbf{q}}^r} \left(\dot{\mathbf{q}}^t - \mathbf{B} \hat{R} \mathbf{G} \dot{\mathbf{q}}^r\right) = -\mathbf{B} \hat{R} \mathbf{G}$$

for the rotational generalized coordinates $\mathbf{q}^r$. In order to find an expression for Q_i in matrix notation, the scalar product from Eq. (2.5) is written as matrix multiplication

$$Q_i = \sum_{K=1}^{N} \left(\vec{f}_K\right)^\mathsf{T} \frac{\partial \vec{r}_K}{\partial q_i}$$

and the generalized forces therefore read

$$\mathbf{Q} = \begin{pmatrix} Q_1 \\ \vdots \\ Q_n \end{pmatrix} = \sum_{K=1}^{N} \left(\frac{\partial \vec{r}_K}{\partial \mathbf{q}} \right)^{\mathsf{T}} \vec{f}_K.$$

Hence, the generalized forces for one rigid body can be assembled as external forces:

$$\mathbf{Q}^{(e)} = \begin{pmatrix} \mathbf{Q}^t \\ \mathbf{Q}^r \end{pmatrix} = \sum_{K=1}^{N} \begin{pmatrix} \vec{f}_K \\ \mathbf{G}^{\mathsf{T}} \hat{R}_K \mathbf{B}^{\mathsf{T}} \vec{f}_K \end{pmatrix} = \sum_{K=1}^{N} \begin{pmatrix} \mathbf{I} \\ \mathbf{G}^{\mathsf{T}} \hat{R}_K \mathbf{B}^{\mathsf{T}} \end{pmatrix} \vec{f}_K. \tag{2.33}$$

2.3.3 Equations of motion for a single rigid body

At this point all equations necessary for performing dynamical analyses of multibody systems are developed. In this section they are assembled to one differential algebraic system. Before inserting Eq. (2.32) and Eq. (2.33) in Eq. (2.14) some partial derivatives of the kinetic energy are computed. The translational part of the Lagrange equations is simply given by

$$m\mathbf{I}\ddot{\mathbf{q}}^t = \sum_{i=1}^{n_f} \vec{f}_i, \tag{2.34}$$

with $\mathbf{I}$ being the identity matrix. More attention must be paid to the rotational parts of the equations of motion. First the partial derivative of T with respect to the generalized velocities is specified:

$$\frac{\partial T}{\partial \dot{\mathbf{q}}^r} = \mathbf{G}^{\mathsf{T}} \mathbf{I}_M \mathbf{G} \dot{\mathbf{q}}^r.$$

This is followed by the differentiation with respect to time, which results in

$$\frac{\mathrm{d}}{\mathrm{d}t} \left(\frac{\partial T}{\partial \dot{\mathbf{q}}^r} \right) = \mathbf{G}^{\mathsf{T}} \mathbf{I}_M \mathbf{G} \ddot{\mathbf{q}}^r + \left(\dot{\mathbf{G}}^{\mathsf{T}} \mathbf{I}_M \mathbf{G} + \mathbf{G}^{\mathsf{T}} \mathbf{I}_M \dot{\mathbf{G}} \right) \dot{\mathbf{q}}^r. \tag{2.35}$$

The second term involving a partial derivative of T is given by

$$\frac{\partial T}{\partial \mathbf{q}^r} = \frac{\partial}{\partial \mathbf{q}^r} \left(\frac{1}{2} \left(\dot{\mathbf{q}}^r \right)^{\mathsf{T}} \mathbf{G}^{\mathsf{T}} \mathbf{I}_M \mathbf{G} \dot{\mathbf{q}}^r \right). \tag{2.36}$$

Up to this point no specific rotation parametrization has been chosen. Hence, no further improvements of the equation structure can be performed. In case of using Euler parameters the equations may be written in an elegant manner. Primarily the identity

$$\mathbf{G}\mathbf{q}^r = 2 \begin{pmatrix} -e_1 e_0 + e_0 e_1 + e_3 e_2 - e_2 e_3 \\ -e_2 e_0 - e_3 e_1 + e_0 e_2 + e_1 e_3 \\ -e_3 e_0 + e_2 e_1 - e_1 e_2 + e_0 e_3 \end{pmatrix} = \mathbf{0}$$

is adopted, which holds for Euler parameters only. Because of the structure of $\mathbf{G}$ not only the relation $\mathbf{G}\mathbf{q}^r = \mathbf{0}$ but also $\dot{\mathbf{G}}\dot{\mathbf{q}}^r = \mathbf{0}$ is valid. Furthermore, the derivative with respect to time

$$\dot{\mathbf{G}}\mathbf{q}^r + \mathbf{G}\dot{\mathbf{q}}^r = \mathbf{0}$$

can be used in the form

$$\dot{\mathbf{G}}\mathbf{q}^r = -\mathbf{G}\dot{\mathbf{q}}^r. \tag{2.37}$$

Due to this Eq. (2.35) simplifies to

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial T}{\partial \dot{\mathbf{q}}^r}\right) = \mathbf{G}^\mathsf{T}\mathbf{I}_M\mathbf{G}\ddot{\mathbf{q}}^r - \dot{\mathbf{G}}^\mathsf{T}\mathbf{I}_M\dot{\mathbf{G}}\mathbf{q}^r \tag{2.38}$$

By making use of Eq. (2.37), Equation (2.36) can also be simplified and reads:

$$\begin{aligned}\frac{\partial T}{\partial \mathbf{q}^r} &= \frac{\partial}{\partial \mathbf{q}^r}\left(\frac{1}{2}\left(\mathbf{q}^r\right)^\mathsf{T}\dot{\mathbf{G}}^\mathsf{T}\mathbf{I}_M\dot{\mathbf{G}}\mathbf{q}^r\right)\\ &= \dot{\mathbf{G}}^\mathsf{T}\mathbf{I}_M\dot{\mathbf{G}}\mathbf{q}^r.\end{aligned} \tag{2.39}$$

Assembling the terms Eq. (2.34), Eq. (2.38), Eq. (2.39) and using the external forces $\mathbf{Q}^{(e)}$ from Eq. (2.33) gives

$$\begin{pmatrix} m\mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{G}^\mathsf{T}\mathbf{I}_M\mathbf{G}\end{pmatrix}\begin{pmatrix}\ddot{\mathbf{q}}^t \\ \ddot{\mathbf{q}}^r\end{pmatrix} = \sum_{i=1}^{n_f}\begin{pmatrix}\mathbf{I} \\ \mathbf{G}^\mathsf{T}\hat{R}_i\mathbf{B}^\mathsf{T}\end{pmatrix}\vec{f}_i - \begin{pmatrix}\mathbf{0} \\ 2\dot{\mathbf{G}}^\mathsf{T}\mathbf{I}_M\dot{\mathbf{G}}\mathbf{q}^r\end{pmatrix} - \mathbf{C}_\mathbf{q}^\mathsf{T}\lambda. \tag{2.40}$$

By introducing the force vector $\mathbf{f}(\mathbf{q}, \dot{\mathbf{q}}, t)$ combining external forces and gyroscopic forces,

$$\mathbf{f}(\mathbf{q}, \dot{\mathbf{q}}, t) = \sum_{i=1}^{n_f}\begin{pmatrix}\mathbf{I} \\ \mathbf{G}^\mathsf{T}\hat{R}_i\mathbf{B}^\mathsf{T}\end{pmatrix}\vec{f}_i - \begin{pmatrix}\mathbf{0} \\ 2\dot{\mathbf{G}}^\mathsf{T}\mathbf{I}_M\dot{\mathbf{G}}\mathbf{q}^r\end{pmatrix},$$

and the mass matrix $\mathbf{M}$,

$$\mathbf{M} = \begin{pmatrix} m\mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{G}^\mathsf{T}\mathbf{I}_M\mathbf{G}\end{pmatrix}, \tag{2.41}$$

the equations of motion for one rigid body subject to constraints $\mathbf{C}$ can be written in the form

$$\begin{aligned}\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} &= \mathbf{f}(\mathbf{q}, \dot{\mathbf{q}}, t) - \mathbf{C}_\mathbf{q}^\mathsf{T}\lambda\\ \mathbf{C}(\mathbf{q}) &= \mathbf{0}\end{aligned} \tag{2.42}$$

where Eq. (2.40) is completed with $\mathbf{C}(\mathbf{q}) = \mathbf{0}$ in order to satisfy the constraint equations.

2.3.4 Equations of motion for multibody systems

The main goal of this chapter is to develop the equations of motion for an entire multibody system in the form of Eq. (2.42). Therefore a mass matrix in the form

$$\mathbf{M} = \begin{pmatrix} \mathbf{M}_1 & \mathbf{0} & & \\ \mathbf{0} & \mathbf{M}_2 & \mathbf{0} & \\ & \mathbf{0} & \ddots & \mathbf{0} \\ & & \mathbf{0} & \mathbf{M}_N \end{pmatrix} \tag{2.43}$$

is introduced. Herein $\mathbf{M}_1, \dots, \mathbf{M}_N$ denotes the N rigid body mass matrices in the form of Eq. (2.41). The generalized coordinates therefore read

$$\mathbf{q} = \begin{pmatrix} \mathbf{q}_1^t \\ \mathbf{q}_1^r \\ \vdots \\ \mathbf{q}_N^t \\ \mathbf{q}_N^r \end{pmatrix} \tag{2.44}$$

and assemble all translational and rotational generalized coordinates related to all N bodies. The same applies for the forces $\mathbf{f} = (\mathbf{f}_1, \dots, \mathbf{f}_N)^{\mathsf{T}}$.

In addition to the equations of inner constraints $\mathbf{C}^{(i)}$, resulting from a redundant set of rotational generalized coordinates, the equations representing the joints interconnecting the bodies, also called outer constraints $\mathbf{C}^{(o)}$, have to be observed. They are assembled in the form

$$\mathbf{C}(\mathbf{q}) = \begin{pmatrix} \mathbf{C}^{(1)}(\mathbf{q}_1) \\ \mathbf{C}^{(2)}(\mathbf{q}_2) \\ \vdots \\ \mathbf{C}^{(N)}(\mathbf{q}_N) \\ \mathbf{C}^{(o)}(\mathbf{q}) \end{pmatrix} \tag{2.45}$$

which result in the constraint Jacobian

$$\mathbf{C}_{\mathbf{q}} = \begin{pmatrix} \mathbf{C}^{(1)}_{\mathbf{q}_1} & \mathbf{0} & & \\ \mathbf{0} & \mathbf{C}^{(2)}_{\mathbf{q}_2} & \mathbf{0} & \\ & \mathbf{0} & \ddots & \mathbf{0} \\ & & \mathbf{0} & \mathbf{C}^{(N)}_{\mathbf{q}_N} \\ \mathbf{C}^{(o)}_{\mathbf{q}_1} & \mathbf{C}^{(o)}_{\mathbf{q}_2} & \dots & \mathbf{C}^{(o)}_{\mathbf{q}_N} \end{pmatrix}. \tag{2.46}$$

Similar to Eq. (2.42) the equations of motion for a multibody system read

$$\begin{aligned} \mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} &= \mathbf{f}(\mathbf{q}, \dot{\mathbf{q}}, t) - \mathbf{C}_{\mathbf{q}}^{\mathsf{T}}\lambda \\ \mathbf{C}(\mathbf{q}) &= \mathbf{0} \end{aligned} \tag{2.47}$$

using the definitions of Eq. (2.44), Eq. (2.43), Eq. (2.45) and Eq. (2.46) respectively.

2.4 Numerics

As mentioned at the beginning of this chapter the only approach practicable for analyzing multibody systems modeled by equations in the form of Eq. (2.47) is to apply numerical solution strategies. Finding solutions by analytical methods is hardly possible even for small non-linear systems and therefore not applicable for multibody systems.

The very first step before applying numerical solution methods is to discretize the equations of interest. Therefore the state at $t = t_n$ is abbreviated with $\mathbf{q}(t_n) =: \mathbf{q}_n$ and similarly this applies to additional Lagrangian multipliers $\lambda(t_n) =: \lambda_n$. Doing so and using the time t_n, the states $\mathbf{q}_n$, $\dot{\mathbf{q}}_n$ and the Lagrangian multiplier λ_n in Eq. (2.47) lead to the discretized version of the differential-algebraic equations reading

$$\begin{aligned} \mathbf{M}(\mathbf{q}_{n+1})\ddot{\mathbf{q}}_{n+1} - \mathbf{f}(\mathbf{q}_{n+1}, \dot{\mathbf{q}}_{n+1}, t_{n+1}) + \mathbf{C}_{\mathbf{q}}(\mathbf{q}_{n+1})^{\mathsf{T}}\lambda_{n+1} &= \mathbf{0} \\ \mathbf{C}(\mathbf{q}_{n+1}) &= \mathbf{0} \end{aligned}$$

which is below written in the simpler form

$$\begin{aligned} (\mathbf{M}\ddot{\mathbf{q}})_{n+1} - \left(\mathbf{f} - \mathbf{C}_{\mathbf{q}}^{\mathsf{T}}\lambda\right)_{n+1} &= \mathbf{0} \\ \mathbf{C}_{n+1} &= \mathbf{0}. \end{aligned} \tag{2.48}$$

In the following the Hilber-Hughes-Taylor (HHT) implicit method [32] is chosen to be investigated in detail as one representation of numeric integrators. In the field of structural dynamics this method is used for the integration of a linear set of second order ordinary differential equations (ODE). Due to the artificial damping introduced by the algorithm, this allows for efficient time integration without the need for resolving frequencies out of the range of interest. Beyond that in further derivations the good conditioning of the Jacobian matrix, associated with the implicit integrator, can be found to be beneficial.

2.4.1 Newmark discretization

Before going into detail in terms of the HHT method the underlying integration formulas are discussed in advance. These are the Newmark formulas [51]

$$\begin{aligned} \mathbf{q}_{n+1} &= \mathbf{q}_n + h\dot{\mathbf{q}}_n + \frac{h^2}{2}\left[(1-2\beta)\ddot{\mathbf{q}}_n + 2\beta\ddot{\mathbf{q}}_{n+1}\right] \\ \dot{\mathbf{q}}_{n+1} &= \dot{\mathbf{q}}_n + h\left[(1-\gamma)\ddot{\mathbf{q}}_n + \gamma\ddot{\mathbf{q}}_{n+1}\right] \end{aligned} \tag{2.49}$$

depending on two parameters β and γ and utilizing the integration step size $h = t_{n+1} - t_n$. Inserting these formulas in Eq. (2.48) the acceleration $\ddot{\mathbf{q}}_{n+1}$ remains the only unknown quantity. However, the resulting equations are implicit. When selecting $\beta = 1/2$ and $\gamma = 1/4$, this ends up with the trapezoidal rule, which is known to be A-stable. Expressed in simple words this means the integration formula provides correct results for the test equation $y'(x) = ky$ for $k \in \mathbb{C}$ with the initial condition $y(0) = 1$. The major drawback of the trapezoidal rule is that it does not induce any numerical damping and therefore is impractical when thinking about undamped high-frequency oscillations potentially arising in multibody systems. This is why the HHT-method was suggested for finding parameters β and γ and the suitable discrete equations of motion leading to both, A-stability and numerical damping.

2.4.2 Hilber-Hughes-Taylor method

Following the idea presented in [32] instead of modifying the Newmark formulas themselves the equations of motion are manipulated and thus reading

$$\frac{1}{1+\alpha}\left(\mathbf{M}\ddot{\mathbf{q}}\right)_{n+1} - \left(\mathbf{f} - \mathbf{C}_{\mathbf{q}}^{\mathsf{T}}\lambda\right)_{n+1} + \frac{\alpha}{1+\alpha}\left(\mathbf{f} - \mathbf{C}_{\mathbf{q}}^{\mathsf{T}}\lambda\right)_{n} = \mathbf{0}$$

where the forces of the previous time step, scaled with the factor α, are applied again. The method is stable for $\alpha \in [-1/3, 0]$ and if the parameters of the Newmark integration formulas are

$$\gamma = \frac{1-2\alpha}{2} \qquad \beta = \frac{(1-\alpha)^2}{4}.$$

As proposed by Hilber, Hughes and Taylor in every discrete time step an error

$$\begin{aligned}
\mathbf{e}_1 &= \frac{1}{1+\alpha}\left(\mathbf{M}\ddot{\mathbf{q}}\right)_{n+1} - \left(\mathbf{f} - \mathbf{C}_{\mathbf{q}}^{\mathsf{T}}\lambda\right)_{n+1} + \frac{\alpha}{1+\alpha}\left(\mathbf{f} - \mathbf{C}_{\mathbf{q}}^{\mathsf{T}}\lambda\right)_{n} \\
\mathbf{e}_2 &= \frac{1}{\beta h^2}\mathbf{C}_{n+1}
\end{aligned}$$

is computed. In order to trim these errors to equate to zero, an iterative update process for the accelerations and the Lagrange multipliers can be introduced. The equation system for the updates $\Delta\ddot{\mathbf{q}}$ and $\Delta\lambda$ reads

$$\begin{pmatrix} \hat{\mathbf{M}} & \mathbf{C}_{\mathbf{q}}^{\mathsf{T}} \\ \mathbf{C}_{\mathbf{q}} & \mathbf{0} \end{pmatrix} \begin{pmatrix} \Delta\ddot{\mathbf{q}} \\ \Delta\lambda \end{pmatrix} = \begin{pmatrix} -\mathbf{e}_1 \\ -\mathbf{e}_2 \end{pmatrix} \tag{2.50}$$

where the Jacobian $\hat{\mathbf{M}}$ is computed by building the total derivative of $\mathbf{e}_1$ with respect to the acceleration $\ddot{\mathbf{q}}_{n+1}$:

$$\frac{\mathrm{d}\mathbf{e}_1}{\mathrm{d}\ddot{\mathbf{q}}_{n+1}} = \frac{\partial \mathbf{e}_1}{\partial \ddot{\mathbf{q}}_{n+1}} + \frac{\partial \mathbf{e}_1}{\partial \dot{\mathbf{q}}_{n+1}}\frac{\partial \dot{\mathbf{q}}_{n+1}}{\partial \ddot{\mathbf{q}}_{n+1}} + \frac{\partial \mathbf{e}_1}{\partial \mathbf{q}_{n+1}}\frac{\partial \mathbf{q}_{n+1}}{\partial \ddot{\mathbf{q}}_{n+1}}.$$

Inserting the partial derivatives of the Newmark integration formulas Eq. (2.49)

$$\frac{\partial \mathbf{q}_{n+1}}{\partial \ddot{\mathbf{q}}_{n+1}} = \beta h^2 \quad \text{and} \quad \frac{\partial \dot{\mathbf{q}}_{n+1}}{\partial \ddot{\mathbf{q}}_{n+1}} = \gamma h$$

the Jacobian $\hat{\mathbf{M}}$ results in

$$\hat{\mathbf{M}} = \frac{\partial \mathbf{e}_1}{\partial \ddot{\mathbf{q}}_{n+1}} = \left(\frac{1}{1+\alpha} \mathbf{M} - \gamma h \frac{\partial \mathbf{f}}{\partial \dot{\mathbf{q}}} + \beta h^2 \left[\frac{1}{1+\alpha} (\mathbf{M}\ddot{\mathbf{q}})_{\mathbf{q}} - \frac{\partial \mathbf{f}}{\partial \mathbf{q}} + \left(\mathbf{C}_{\mathbf{q}}^{\mathsf{T}} \lambda \right)_{\mathbf{q}} \right] \right)_{n+1} .$$

Herein, index $\mathbf{q}$ is used to indicate a partial derivative with respect to the generalized coordinates. By setting $\ddot{\mathbf{q}}_{n+1} \rightarrow \ddot{\mathbf{q}}_{n+1} + \Delta\ddot{\mathbf{q}}$ and $\lambda_{n+1} \rightarrow \lambda_{n+1} + \Delta\lambda$ improved accelerations and Lagrange multipliers are obtained. Repeating this process should reduce the residua $\mathbf{e}_1$, $\mathbf{e}_2$ to approximately zero after several iterations. As initial guess for $\ddot{\mathbf{q}}_{n+1}$ and λ_{n+1} the values from the previous time step $\ddot{\mathbf{q}}_n$, λ_n can be used. The HHT-method is known to be applicable to a wide range of multibody systems. Using $\alpha = 0$ the method again coincides with the trapezoidal rule. For the use in general purpose multibody simulation software Negrut [50] proposed extensions, such as additional differential equations, user-defined variables, error estimation and integration step-size control.

CHAPTER 3

Adjoint Sensitivity Analysis of Multibody Systems

In the preceding chapter the analysis of multibody systems has been discussed. Based on a set of parameters ξ the evolution of multibody system states can be given as time series. Basically these parameters can be included in the mass matrix $\mathbf{M} = \mathbf{M}(\mathbf{q}, \xi)$, the forces $\mathbf{f} = \mathbf{f}(\mathbf{q}, \dot{\mathbf{q}}, \xi, t)$ and in the constraints $\mathbf{C} = \mathbf{C}(\mathbf{q}, \xi)$. During the design process of systems subject to dynamics the influence of such parameters on the performance of the system is crucial. Therefore not the actual response of the system is of particular interest, but the sensitivity of the system with respect to the set of parameters. Here is where sensitivity analysis comes into play.

3.1 System Sensitivity Analysis

The starting point of sensitivity analysis is to introduce new states which represent the sensitivity of every system state at time t. For this reason the differential equations describing the system to analyze are differentiated with respect to each parameter of interest. The resulting set of additional differential equation systems is not coupled and can be solved separately, provided a solution for the system states is present.

For the sake of simplicity, before operating on the second-order DAE from Eq. (2.47), a system using minimal coordinates is treated. Furthermore an order reduction is performed which allows to investigate the ordinary differential equation system

(ODE)

$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}, \xi, t) \quad \mathbf{x}(0) = \mathbf{x}_0 \\ \mathbf{y} &= \mathbf{y}(\mathbf{x})\end{aligned} \tag{3.1}$$

where $\mathbf{x}(t) \in \mathbb{R}^n$ denotes the vector of state variables, $\mathbf{y} \in \mathbb{R}^m$ the vector of system outputs, $\xi \in \mathbb{R}^l$ the system parameters, and the vector $\mathbf{x}_0$ gives the vector of initial conditions. Differentiating this equation system with respect to the parameters $\xi = (\xi_1, \xi_2, \ldots, \xi_l)^\mathsf{T}$ results in

$$\begin{aligned}\left(\frac{\partial \dot{\mathbf{x}}}{\partial \xi_1}, \ldots, \frac{\partial \dot{\mathbf{x}}}{\partial \xi_l}\right) &= \left(\frac{\partial \mathbf{f}}{\partial \mathbf{x}}\frac{\partial \mathbf{x}}{\partial \xi_1} + \frac{\partial \mathbf{f}}{\partial \xi_1}, \ldots, \frac{\partial \mathbf{f}}{\partial \mathbf{x}}\frac{\partial \mathbf{x}}{\partial \xi_l} + \frac{\partial \mathbf{f}}{\partial \xi_l}\right) \\ \left(\frac{\partial \mathbf{y}}{\partial \xi_1}, \ldots, \frac{\partial \mathbf{y}}{\partial \xi_l}\right) &= \left(\frac{\partial \mathbf{y}}{\partial \mathbf{x}}\frac{\partial \mathbf{x}}{\partial \xi_1}, \ldots, \frac{\partial \mathbf{x}}{\partial \xi_l}\right) = \mathbf{S}\end{aligned} \tag{3.2}$$

where $\mathbf{S}$ denotes the $m \times l$ sensitivity matrix. Introducing the sensitivity states $\mathbf{x}_{\xi_i} := \partial\mathbf{x}/\partial\xi_i$ and assembling them in the matrix $\mathbf{x}_\xi := (\mathbf{x}_{\xi_1}, \mathbf{x}_{\xi_2}, \ldots, \mathbf{x}_{\xi_l})$ Eq. (3.2) reads

$$\begin{aligned}\dot{\mathbf{x}}_\xi &= \mathbf{f}_\mathbf{x}\mathbf{x}_\xi + \mathbf{f}_\xi \\ \mathbf{y}_\xi &= \mathbf{y}_\mathbf{x}\mathbf{x}_\xi = \mathbf{S}\end{aligned} \tag{3.3}$$

where the Jacobians $\mathbf{f}_\mathbf{x} := \partial\mathbf{f}/\partial\mathbf{x}$, $\mathbf{f}_\xi := \partial\mathbf{f}/\partial\xi$, and $\mathbf{y}_\mathbf{x} := \partial\mathbf{y}/\partial\mathbf{x}$ are used. Restricting oneself to initial conditions $\mathbf{x}_0$ that do not depend on parameters, the initial conditions of the sensitivity differential equations read

$$\mathbf{x}_\xi(0) = \mathbf{0}.$$

Equation (3.3) is a linear time-variant matrix differential equation and may be solved with common numerical integration methods. The time-variant Jacobians $\mathbf{f}_\mathbf{x}$, $\mathbf{f}_\xi$, and $\mathbf{y}_\mathbf{x}$ must be evaluated for a solution of Eq. (3.1). In regards to computation effort, this means that a differential equation system of the same size as the equations of motion has to be solved for each parameter ξ_i of the system.

3.1.1 Illustrative example: Single degree of freedom oscillator

In order to demonstrate the sensitivity analysis, equations (3.2) are applied to a mechanical problem, namely the single degree of freedom oscillator (see Fig. 3.1). After order reduction the equations of motion thereof read

$$\begin{aligned}\begin{pmatrix}\dot{x} \\ \dot{v}\end{pmatrix} = \dot{\mathbf{x}} &= \begin{pmatrix} v \\ -cx - 2dv + \sin(t)\end{pmatrix} \\ y &= \begin{pmatrix}1 & 0\end{pmatrix}\mathbf{x} = x,\end{aligned} \tag{3.4}$$

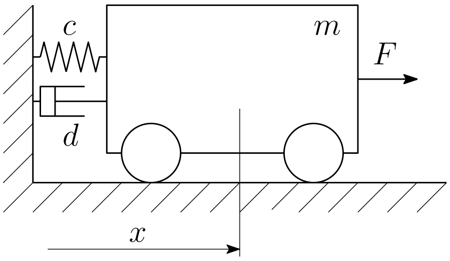


Figure 3.1: Representation of a single degree of freedom oscillator with mass m, linear stiffness c and linear damping d

where the mass is already set to $m = 1$. In the following the sensitivity of the position $x(t)$ with respect to the two parameters c and d, representing the stiffness and the damping coefficient, will be analyzed. The system is excited with a harmonic force having the amplitude $F = 1$ and the frequency $\Omega = 1$. Assuming a weakly damped system with $d^2 < c$ and using the abbreviation $\omega = \sqrt{c - d^2}$, the analytical solution for $x(t)$ is given by

$$\begin{aligned} x(t) = \frac{1}{\omega\left((c-1)^2 + 4d^2\right)} \Bigg\{ & e^{-dt}\Big[\Big(-c + 2d^2 + 1\Big)\sin(\omega t) \\ & + 2d\omega\cos(\omega t)\Big] - \omega((1-c)\sin(t) + 2d\cos(t))\Bigg\}, \end{aligned} \tag{3.5}$$

where the initial conditions $x(0) = 0$ and $v(0) = 0$ were used. Using the abbreviations $\mathbf{x}_c$ and $\mathbf{x}_d$ instead of the partial derivatives $\partial\mathbf{x}/\partial c$ and $\partial\mathbf{x}/\partial d$ respectively, the system sensitivities may be computed by differentiation of the analytic solution for $x(t)$. Hence, the sensitivity with respect to the parameter c is given by

$$\begin{aligned} x_c(t) &= \frac{1}{2((c-1)^2 + 4d^2)^2\omega^3}\Big(e^{-dt}a_h(t) + a_p(t)\Big) \\ a_h(t) &= 2\omega^2\Big(-(c-1)^2 - 4d^2\Big)\sin(\omega t) \\ &\quad - 4(1-c)\omega^2\Big[\Big(c - 2d^2 - 1\Big)\sin(\omega t) - 2d\omega\cos(\omega t)\Big] \\ &\quad - \Big((c-1)^2 + 4d^2\Big)\Big[t\omega\Big(c - 2d^2 - 1\Big)\cos(\omega t) \\ &\quad + \sin(\omega t)\Big(-c + 2d\Big(d + t\omega^2\Big) + 1\Big)\Big] \\ a_p(t) &= 2\omega^3\Big(4(c-1)d\cos(t) - \Big(c^2 - 2c - 4d^2 + 1\Big)\sin(t)\Big) \end{aligned}$$

and the sensitivity with respect to the parameter d reads

$$
\begin{aligned}
x_d(t) =\ & \frac{1}{((c-1)^2+4d^2)^2\omega]^3}\left(e^{-dt}b_h(t)+b_p(t)\right)\\
b_h(t) =\ & \sin(\omega t)\Big[(c-1)\omega^2\left(4(c+1)d+(c-1)^2t+4d^2t\right)\\
& +d\left(-c+2d^2+1\right)\left((c-1)^2+4d^2\right)\Big]\\
& -\omega\cos(\omega t)\Big\{2\omega^2\left[-c^2+td\left((c-1)^2+4d^2\right)+2c+4d^2-1\right]\\
& +td\left(-c+2d^2+1\right)\left((c-1)^2+4d^2\right)\Big\}\\
b_p(t) =\ & 2\omega^3\left(-\left((c-1)^2-4d^2\right)\cos(t)-4(c-1)d\sin(t)\right).
\end{aligned}
$$

Here, both solutions are split up into the decaying terms $a_h(t)$ and $b_h(t)$ resulting from the homogeneous equation and the terms oscillating with the excitation frequency $a_p(t)$ and $b_p(t)$ resulting from the particular solution.
For the actual choice of parameters $c = 1$ and $d = 1/2$ inserting in the analytical solution gives

$$
\begin{aligned}
x(t) &= e^{-t/2}\left(\frac{1}{\sqrt{3}}\sin\left(\frac{\sqrt{3}t}{2}\right)+\cos\left(\frac{\sqrt{3}t}{2}\right)\right)-\cos(t)\\
x_c(t) &= \sin(t)+\frac{1}{9}e^{-t/2}\left(3t\cos\left(\frac{\sqrt{3}t}{2}\right)-\sqrt{3}(3t+8)\sin\left(\frac{\sqrt{3}t}{2}\right)\right)\\
x_d(t) &= 2\cos(t)+\frac{2}{9}e^{-t/2}\left(\sqrt{3}\sin\left(\frac{\sqrt{3}t}{2}\right)-3(2t+3)\cos\left(\frac{\sqrt{3}t}{2}\right)\right).
\end{aligned}
\tag{3.6}
$$

Figure 3.2 shows the evaluation of Eq. (3.6) in the time range of $t \in [0, 6\pi]$s. Note, that after the homogeneous solution has decayed, the sensitivity for the damping parameter d oscillates with a pure cosine with double the amplitude of the sensitivity with respect to the stiffness parameter c oscillating with a pure sine. In other words, this means that there are points with zero sensitivity for one parameter, while the sensitivity with respect to the other parameter becomes maximal or minimal respectively. Drawing conclusions for improving the design of systems and machines is hardly possible and therefore a more handy description of a system's performance has to be found.

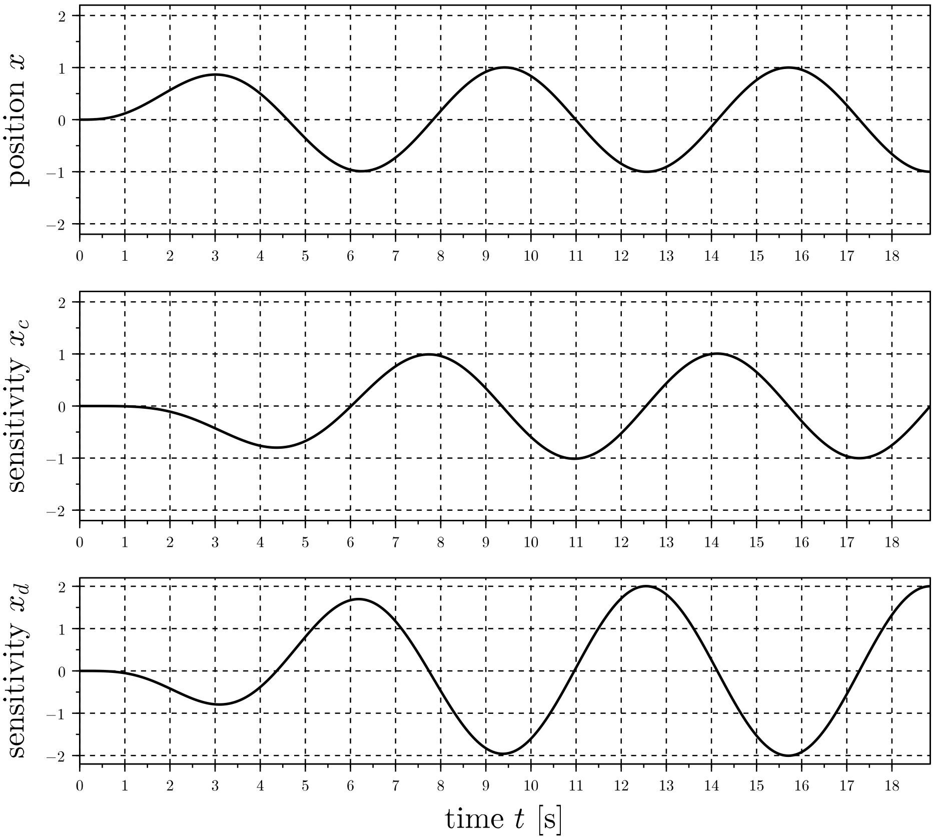


Figure 3.2: Result of sensitivity analysis for single degree of freedom oscillator

3.1.2 Application to multibody system dynamics

Recalling the equations of motion for general multibody systems given in Eq. (2.47) and expanding the set of equations by the system outputs $\mathbf{y}$ yields

$$\begin{aligned}\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} &= \mathbf{f}(\mathbf{q},\dot{\mathbf{q}},t) - \mathbf{C}_{\mathbf{q}}^{\mathsf{T}}\lambda \\ \mathbf{C}(\mathbf{q}) &= \mathbf{0} \\ \mathbf{y} &= \mathbf{y}(\mathbf{q},\dot{\mathbf{q}}).\end{aligned}$$

After performing an order reduction through introducing $\mathbf{v} \coloneqq \dot{\mathbf{q}}$, this equations read

$$\begin{aligned}\dot{\mathbf{q}} &= \mathbf{v} \\ \mathbf{M}(\mathbf{q})\dot{\mathbf{v}} &= \mathbf{f}(\mathbf{q},\mathbf{v},t) - \mathbf{C}_{\mathbf{q}}^{\mathsf{T}}\lambda \\ \mathbf{C}(\mathbf{q}) &= \mathbf{0} \\ \mathbf{y} &= \mathbf{y}(\mathbf{q},\mathbf{v}).\end{aligned} \tag{3.7}$$

Similar to the procedure done in Eq. (3.2), the direct differentiation method is applied to the equations of Eq. (3.7). The sensitivity differential algebraic equations therefore read

$$\begin{aligned}\dot{\mathbf{q}}_\xi &= \mathbf{v}_\xi \\ \mathbf{M}\dot{\mathbf{v}}_\xi + \left(\mathbf{M}\dot{\mathbf{v}}\right)_\mathbf{q}\mathbf{q}_\xi + \left(\mathbf{M}\dot{\mathbf{v}}\right)_\xi &= \mathbf{f}_\mathbf{q}\mathbf{q}_\xi + \mathbf{f}_\mathbf{v}\mathbf{v}_\xi + \mathbf{f}_\xi \\ &\quad - \mathbf{C}_\mathbf{q}^\mathsf{T}\lambda_\xi - \left(\mathbf{C}_\mathbf{q}^\mathsf{T}\lambda\right)_\mathbf{q}\mathbf{q}_\xi - \left(\mathbf{C}_\mathbf{q}^\mathsf{T}\lambda\right)_\xi \\ \mathbf{C}_\mathbf{q}\mathbf{q}_\xi + \mathbf{C}_\xi &= \mathbf{0}.\end{aligned} \tag{3.8}$$

which again can be evaluated for a given time range via numerical integration methods.

In order to initialize a numerical integration method, the initial conditions for Eq. (3.8) have to be found. The initial states of the multibody system $\mathbf{q}(0) = \mathbf{q}_0$ and $\mathbf{v}(0) = \mathbf{v}_0$ are prescribed and therefore the initial conditions for $\mathbf{q}_\xi$ and $\mathbf{v}_\xi$ are readily given by $\mathbf{q}_\xi(0) = \mathbf{0}$ and $\mathbf{v}_\xi(0) = \mathbf{0}$.

3.2 Performance Measures

In order to quantify the results of dynamical and sensitivity analysis, performance measures are introduced. Basically three different types of performance measures or cost functions can be distinguished. In the most general case the costs are computed by

$$J(\xi) = \phi(t_0, \mathbf{y}(t_0), T, \mathbf{y}(T)) + \int_0^T h(t, \mathbf{y})\mathrm{d}t, \tag{3.9}$$

which is usually denoted as the *Bolza form.* If only contributions at t_0 and T are present in J, the cost function is in *Mayer form* and reads

$$J(\xi) = \phi(t_0, \mathbf{y}(t_0), T, \mathbf{y}(T)).$$

In the opposite case, if J only includes the integral component of Eq. (3.9), the cost function is in *Lagrange form* and therefore reads

$$J(\xi) = \int_0^T h(t, \mathbf{y})\mathrm{d}t. \tag{3.10}$$

For example one design goal could require one, or several system outputs to follow a desired trajectory. In this case a root mean square (RMS) error, using the squared differences of current and desired system outputs, could be utilized in $h(t, \mathbf{y}, \xi)$. The performance of this system may then be described by

$$J(\xi) = \frac{1}{2}\int_0^T \left(\mathbf{y} - \bar{\mathbf{y}}\right)^\mathsf{T}\left(\mathbf{y} - \bar{\mathbf{y}}\right)\mathrm{d}t, \tag{3.11}$$

with $\bar{\mathbf{y}}$ being the vector of desired system outputs.
In order to provide information about the sensitivity upon parameter changes, the expression of J can be used for the computation. Forming the derivative

$$\frac{\partial J(\xi)}{\partial \xi} = (\nabla J)^\mathsf{T} = \int_0^T (\mathbf{y} - \bar{\mathbf{y}})^\mathsf{T}\, \mathbf{S}\, \mathrm{d}t \tag{3.12}$$

leads to the cost function's gradient ∇J, using the sensitivity matrix $\mathbf{S}$.
Similar to the RMS error in time domain given by Eq. (3.11), an RMS type cost function may be defined in the frequency domain using Fourier coefficients. For the sake of simplicity a system featuring a single output is assumed in the following. This output depends on the system states in the form $y(t) := g(\mathbf{x})$. By applying classical Fourier analysis, $y(t)$ can be approximated by

$$y(t) \approx \frac{1}{2} A_0 + \sum_{k=1}^{N} \Big(A_k \cos(\omega_k t) + B_k \sin(\omega_k t) \Big)$$

in which ω_k represents the angular frequency of the kth harmonics, each of which is assigned to the appropriate value of its amplitude $\sqrt{A_k^2 + B_k^2}$. The corresponding Fourier coefficients A_k and B_k are defined by

$$\begin{aligned} A_k &= \frac{2}{T} \int_0^T y(t) \cos(\omega_k t) \mathrm{d}t \\ B_k &= \frac{2}{T} \int_0^T y(t) \sin(\omega_k t) \mathrm{d}t. \end{aligned}$$

Relating the simulation result with measured amplitudes $\sqrt{\bar{A}_k^2 + \bar{B}_k^2},\ k = 1, \ldots, N$ similar to Eq. (3.11), they may be arranged in a performance measure of the form

$$J = \frac{1}{4} \sum_{k=1}^{N} \Big[A_k^2 + B_k^2 - \Big(\bar{A}_k^2 + \bar{B}_k^2 \Big) \Big]^2 .$$

Again, this may be interpreted as a cost function in generalized Lagrange form. However, in some applications it seems promising to take advantage of a cost function in Mayer form (see Section 4.2).

3.2.1 Illustrative example: Single degree of freedom oscillator

Again the single degree of freedom oscillator, introduced in Section 3.1.1, is used for displaying the significance of performance measures. Using the system output

$y = x$ and the desired trajectory $\bar{y} = \bar{x}$, the performance measure given as RMS error may read

$$J = \int_0^T \frac{1}{2} (x - \bar{x})^2 \, \mathrm{d}t.$$

Forming the derivatives

$$\frac{\partial J}{\partial c} = \int_0^T (x - \bar{x}) \frac{\partial x}{\partial c} \mathrm{d}t$$

and

$$\frac{\partial J}{\partial d} = \int_0^T (x - \bar{x}) \frac{\partial x}{\partial d} \mathrm{d}t$$

provides the components of the cost function's gradient ∇J:

$$\nabla J = \int_0^T \begin{pmatrix} x_c \\ x_d \end{pmatrix} (x - \bar{x}) \, \mathrm{d}t = \int_0^T \mathbf{S}^\mathsf{T} (x - \bar{x}) \, \mathrm{d}t. \tag{3.13}$$

Choosing $\bar{x} = 0$, the current design is compared with a design without any motion. Therefore the gradient in Eq. (3.13) gives the sensitivity of the system with respect to changes in the parameters c and d towards less motion. Inserting $x(t)$ and the sensitivities $x_c(t)$, $x_d(t)$ from Eq. (3.6) and therefore again using the parameters $c = 1$, $d = 1/2$ and the end time $T = 6\pi$ results in

$$\begin{aligned} J = \frac{1}{6}\Big\{ & 9\pi - 3 - e^{-6\pi} \left(2 + \cos\left(6\sqrt{3}\pi\right)\right) + 2e^{-3\pi} \left(\sqrt{3} \sin\left(3\sqrt{3}\pi\right) + 3\cos\left(3\sqrt{3}\pi\right)\right) \Big\} \\ \approx {} & 4.2123 \end{aligned}$$

and the gradient $\nabla J = (J_c, J_d)^\mathsf{T}$ with

$$\begin{aligned} J_c = {} & \frac{1}{36} e^{-6\pi} \Big\{ 16 - 27e^{6\pi} + 3\sqrt{3}(3 + 8\pi) \sin\left(6\sqrt{3}\pi\right) + 11 \cos\left(6\sqrt{3}\pi\right) \\ & + 8e^{3\pi} \left(9\pi \cos\left(3\sqrt{3}\pi\right) - \sqrt{3}(4 + 9\pi) \sin\left(3\sqrt{3}\pi\right)\right) \Big\} \approx -0.7498 \\ J_d = {} & \frac{1}{9} e^{-6\pi} \Big\{ 14 + e^{6\pi}(36 - 54\pi) + 36\pi - 3\sqrt{3}(1 + 2\pi) \sin\left(6\sqrt{3}\pi\right) \\ & + 2(2 + 9\pi) \cos\left(6\sqrt{3}\pi\right) \\ & + 2e^{3\pi} \left(-5\sqrt{3} \sin\left(3\sqrt{3}\pi\right) - 9(3 + 4\pi) \cos\left(3\sqrt{3}\pi\right)\right) \Big\} \approx -14.8474 \end{aligned} \tag{3.14}$$

The interpretation of this result is more promising than the interpretation of the system sensitivities. Increasing the damping coefficient d about Δd results in a increase of J about $\Delta J = J_d \Delta d$. Of course ∇J represents a linearization of J at the current parametrization $\xi = (c, d)^\mathsf{T}$ only and therefore ΔJ is just valid for small changes $\Delta\xi$.

3.3 Adjoint Sensitivity Analysis

In the preceding section the applicability of performance measures is shown. As the computation of ∇J incorporates the sensitivity matrix $\mathbf{S}$ and therefore the solution of the sensitivity differential equations Eq. (3.2). This requires expensive computations of the additional sensitivity differential equations. More precisely l differential equation systems, where l is the number of system parameters, have to be solved in addition to the original one. Now, in order to circumvent these computations the adjoint sensitivity analysis can be adopted. Here one cannot access the sensitivities itself. However, the sensitivity of the performance measure ∇J can still be computed.

3.3.1 Basic considerations on the adjoint approach

In brief, the key idea of the adjoint sensitivity analysis (see [54, 13, 14, 19] for example) may be summarized as follows. Again, the ordinary differential equation (ODE) system

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}, \xi, t) \quad \mathbf{x}(0) = \mathbf{x}_0 \\ \mathbf{y} &= \mathbf{y}(\mathbf{x}) \end{aligned} \tag{3.15}$$

is studied and a performance measure or cost function of the form

$$J(\xi) = \phi(\mathbf{y}(T)) + \int_0^T h(\mathbf{y}, t)\mathrm{d}t \tag{3.16}$$

is investigated, where $h(\mathbf{y}, t)$ can be any function depending on the system outputs $\mathbf{y}$ and on the time explicitly. The term ϕ introduced in Section 3.2 depends on the outputs $\mathbf{y}(T)$ only, as the initial system output is given by prescribed initial conditions $\mathbf{x}_0$. As presented previously, the goal is to compute the partial derivative of J with respect to the parameters ξ. The adjoint sensitivity analysis is a powerful alternative to compute the gradient of $J(\xi)$. First, $J(\xi)$ does not change if Eq. (3.15) is added to the integrand

$$J(\xi) = \phi(\mathbf{y}(T)) + \int_0^T \left[h + \mathbf{p}^\mathsf{T}(\mathbf{f} - \dot{\mathbf{x}})\right] \mathrm{d}t. \tag{3.17}$$

no matter how the multipliers $\mathbf{p}(t)$ are chosen. This is because the equations of motion Eq. (3.15) must hold and evaluate to zero.
Let us now consider the solution for $\mathbf{y}(t)$ according to the system equations from Eq. (3.15) for a set of parameters ξ. The derivation of J from Eq. (3.17) with respect to the set of parameters ξ leads to

$$\frac{\partial J}{\partial \xi} = (\nabla J)^\mathsf{T} = \int_0^T \left[h_\mathbf{y}\mathbf{y}_\mathbf{x}\mathbf{x}_\xi + \mathbf{p}^\mathsf{T}\left(\mathbf{f}_\mathbf{x}\mathbf{x}_\xi + \mathbf{f}_\xi\right) - \mathbf{p}^\mathsf{T}\dot{\mathbf{x}}_\xi\right] \mathrm{d}t + \phi_\mathbf{y}\mathbf{y}_\mathbf{x}\mathbf{x}_\xi\Big|_{t=T} \tag{3.18}$$

where subscript $\mathbf{x}$ represents the derivatives with respect to the states $^{\partial}/_{\partial\mathbf{x}}$ and subscript ξ denotes the derivatives with respect to the set of parameters $^{\partial}/_{\partial\xi}$. Applying integration by parts in the last term of Eq. (3.18) yields

$$\begin{aligned}(\nabla J)^{\mathsf{T}} = & \int_0^T \left[h_{\mathbf{y}}\mathbf{y}_{\mathbf{x}}\mathbf{x}_\xi + \mathbf{p}^{\mathsf{T}}\left(\mathbf{f}_{\mathbf{x}}\mathbf{x}_\xi + \mathbf{f}_\xi\right) + \dot{\mathbf{p}}^{\mathsf{T}}\mathbf{x}_\xi\right] \mathrm{d}t \\ & - \mathbf{p}^{\mathsf{T}}\mathbf{x}_\xi\Big|_{t=0}^{t=T} + \phi_{\mathbf{y}}\mathbf{y}_{\mathbf{x}}\mathbf{x}_\xi\Big|_{t=T} \\ = & \int_0^T \left[\left(h_{\mathbf{y}}\mathbf{y}_{\mathbf{x}} + \mathbf{p}^{\mathsf{T}}\mathbf{f}_{\mathbf{x}} + \dot{\mathbf{p}}^{\mathsf{T}}\right)\mathbf{x}_\xi + \mathbf{p}^{\mathsf{T}}\mathbf{f}_\xi\right] \mathrm{d}t + \left(\phi_{\mathbf{y}}\mathbf{y}_{\mathbf{x}} - \mathbf{p}^{\mathsf{T}}\right)\mathbf{x}_\xi\Big|_{t=T}\end{aligned} \tag{3.19}$$

since $\mathbf{x}_\xi(0) = 0$ as the initial state is assumed to be prescribed and does not depend on parameters ξ. Key idea of the adjoint sensitivity analysis is to circumvent the computation of the sensitivities $\mathbf{x}_\xi$ by choosing the adjoint variables $\mathbf{p}(t)$ such that

$$\dot{\mathbf{p}} = -\mathbf{y}_{\mathbf{x}}^{\mathsf{T}}h_{\mathbf{y}}^{\mathsf{T}} - \mathbf{f}_{\mathbf{x}}^{\mathsf{T}}\mathbf{p} \quad \text{and} \quad \mathbf{p}(T) = \mathbf{y}_{\mathbf{x}}^{\mathsf{T}}\phi_{\mathbf{y}}^{\mathsf{T}} \tag{3.20}$$

holds. This set of ordinary differential equations is called the adjoint system of Eq. (3.15). It may be solved backwards in time starting at time $t = T$, once the original equations have been solved forward for $t \in [0, T]$. Using the solutions $\mathbf{x}(t)$ and $\mathbf{p}(t)$ from Eqs. (3.15) and (3.20), the gradient of J according to Eq. (3.19) is already given by

$$\nabla J = \int_0^T \mathbf{f}_\xi^{\mathsf{T}}\mathbf{p}\; \mathrm{d}t. \tag{3.21}$$

Note, that only two systems of ODEs must be solved for evaluating the gradient ∇J by Eq. (3.21). Compared to the computation of ∇J by using the system sensitivities and therefore solving Eq. (3.2), this leads to increased efficiency. Unlike as in the case of computing sensitivities, the number of equations to be solved does not depend on the number of parameters.

3.3.2 An interpretation of the adjoint variables

Let $\mathbf{x}(\tau) =: \mathbf{x}_\tau$ be the state at $t = \tau \in [0, T]$. The part of the cost function, which can be still influenced after $t = \tau$ is given by

$$J_\tau = \int_\tau^T h(\mathbf{x}, t)\mathrm{d}t$$

where $\mathbf{y} = \mathbf{x}$ is used for simplification. If $\mathbf{x}_\tau$ is now disturbed by $\delta\mathbf{x}(\tau) = \delta\mathbf{x}_\tau$, then the influence of $\delta\mathbf{x}_\tau$ onto J_τ may be studied.
Considering a small variation of the equations of motion $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, t)$ leads to

$$\delta\dot{\mathbf{x}} = \frac{\partial\mathbf{f}}{\partial\mathbf{x}}\delta\mathbf{x} = \mathbf{A}\delta\mathbf{x}. \tag{3.22}$$

As derived in Eq. (3.20) the adjoint equations are given by

$$\dot{\mathbf{p}} = -\mathbf{A}^{\mathsf{T}}\mathbf{p} - h_{\mathbf{x}}^{\mathsf{T}} \quad \text{and} \quad \mathbf{p}(T) = \mathbf{0} \tag{3.23}$$

where $\phi(\mathbf{x})$ is assumed to be zero and $\mathbf{A}$ gives the Jacobian $\partial\mathbf{f}/\partial\mathbf{x}$. Adding the product of Eq. (3.22) with $\mathbf{p}^{\mathsf{T}}$ to the product of Eq. (3.23) with $\delta\mathbf{x}^{\mathsf{T}}$ results in the expression

$$\mathbf{p}^{\mathsf{T}}\delta\dot{\mathbf{x}} + \delta\mathbf{x}^{\mathsf{T}}\dot{\mathbf{p}} = \mathbf{p}^{\mathsf{T}}\mathbf{A}\delta\mathbf{x} - \delta\mathbf{x}^{\mathsf{T}}\mathbf{A}^{\mathsf{T}}\mathbf{p} - \delta\mathbf{x}^{\mathsf{T}}h_{\mathbf{x}}^{\mathsf{T}}.$$

By applying the product rule and using the equality

$$\mathbf{p}^{\mathsf{T}}\mathbf{A}\delta\mathbf{x} = \left(\delta\mathbf{x}^{\mathsf{T}}\mathbf{A}^{\mathsf{T}}\mathbf{p}\right)^{\mathsf{T}} = \delta\mathbf{x}^{\mathsf{T}}\mathbf{A}^{\mathsf{T}}\mathbf{p}$$

this results in

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\mathbf{p}^{\mathsf{T}}\delta\mathbf{x}\right) = -\delta\mathbf{x}^{\mathsf{T}}h_{\mathbf{x}}^{\mathsf{T}} = -h_{\mathbf{x}}\delta\mathbf{x}.$$

Using this relation and inserting in the variation δJ_τ

$$\delta J_\tau = \int_\tau^T h_{\mathbf{x}}\delta\mathbf{x}\mathrm{d}t$$

gives

$$\delta J_\tau = -\int_\tau^T \frac{\mathrm{d}}{\mathrm{d}t}\left(\mathbf{p}^{\mathsf{T}}\delta\mathbf{x}\right)\mathrm{d}t = \mathbf{p}^{\mathsf{T}}(\tau)\delta\mathbf{x}_\tau - \mathbf{p}^{\mathsf{T}}(T)\delta\mathbf{x}(T).$$

Finally, after inserting the boundary condition $\mathbf{p}(T) = \mathbf{0}$ the adjoint variables at $t = \tau$ may be interpreted as the sensitivity of the cost function with respect to the system states at $t = \tau$:

$$\frac{\partial J_\tau}{\partial \mathbf{x}_\tau} = \mathbf{p}^{\mathsf{T}}(\tau).$$

3.3.3 Illustrative example: Single degree of freedom oscillator

Based on the system equations

$$\begin{aligned} \dot{x} &= v \\ \dot{v} &= -cx - 2dv + \sin(t) \\ y &= x \end{aligned}$$

describing the single degree of freedom oscillator presented in Section 3.1.1 and the performance measure introduced in Section 3.2.1

$$J = \int_0^T \frac{1}{2}\left(x - \bar{x}\right)^2 \mathrm{d}t \tag{3.24}$$

the according adjoint system is derived. Further on, the gradient of the performance measure is computed as explained in the preceding section.
First, according to Eq. (3.20) the adjoint system can be written the following:

$$\begin{aligned} \dot{p_1} &= -(x(t) - \bar{x}) + c\,p_2 \\ \dot{p_2} &= -p_1 + 2dp_2. \end{aligned} \tag{3.25}$$

Due to $\phi = 0$ in Eq. (3.24) the boundary terms describing the final state of the adjoint variables evaluate to zero, $p_1(6\pi) = p_2(6\pi) = 0$. Inserting $x(t)$ from the solution of Eq. (3.6) and the according parameter values $c = 1$, $d = 1/2$, the expressions for the adjoint variables read

$$\begin{aligned} p_1(t) = \frac{1}{3}e^{-\frac{t}{2}-6\pi}\Bigg\{ & \left(3e^{6\pi} - 2e^t\right)\cos\left(\frac{\sqrt{3}t}{2}\right) + e^t\bigg[e^{3\pi}\bigg(\sqrt{3}\sin\left(\frac{1}{2}\sqrt{3}(6\pi - t)\right) \\ & + 3\cos\left(\frac{1}{2}\sqrt{3}(6\pi - t)\right)\bigg) - \cos\left(\frac{1}{2}\sqrt{3}(12\pi - t)\right)\bigg]\Bigg\} + \sin(t) - \cos(t), \\ p_2(t) = \frac{1}{6}e^{-\frac{t}{2}-6\pi}\Bigg\{ & -\sqrt{3}\left(e^{6\pi} - 2e^t\right)\sin\left(\frac{\sqrt{3}t}{2}\right) + \left(3e^{6\pi} - 2e^t\right)\cos\left(\frac{\sqrt{3}t}{2}\right) \\ & - e^t\bigg[\sqrt{3}\left(\sin\left(\frac{1}{2}\sqrt{3}(12\pi - t)\right) - 4e^{3\pi}\sin\left(\frac{1}{2}\sqrt{3}(6\pi - t)\right)\right) \\ & + \cos\left(\frac{1}{2}\sqrt{3}(12\pi - t)\right)\bigg]\Bigg\} + \sin(t). \end{aligned} \tag{3.26}$$

The plot in Fig. 3.3 shows both adjoint variables in the range $t = [0, 6\pi]$ s.
In order to obtain the gradient of J, Eq. (3.21) has to be evaluated. In case of the single mass oscillator this reads

$$\nabla J = \int_0^{6\pi} \left(-p_2(t)x(t), -p_2(t)2v(t)\right)^{\mathsf{T}} \, \mathrm{d}t.$$

By inserting $x(t)$, $v(t) = \dot{x}(t)$ from Eq. (3.6) and $p_1(t)$, $p_2(t)$ from Eq. (3.26) this results in the expressions for $\nabla J = (J_c, J_d)^{\mathsf{T}}$ given in Eq. (3.14) as they may be found by using the system sensitivities from Eq. (3.6).

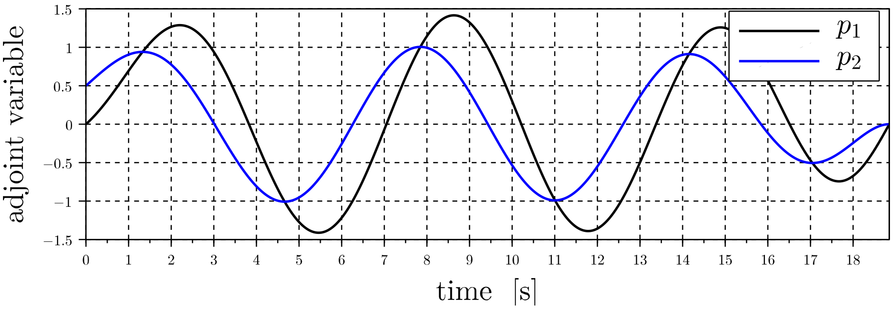


Figure 3.3: Solution of the adjoint system for single degree of freedom oscillator

3.3.4 Application to multibody system dynamics

Due to the complexity of a multibody system, many authors focused on two-dimensional examples or rather general aspects (e.g. [18], [62], [63]). However, based on highly redundant formulations the adjoint equations Eq. (3.20) for a multibody system are relatively simple. In this section, the derivation of necessary equation is shown for general multibody systems. First, the adjoint equations for a multibody system are derived. Below, the structure of the boundary conditions is discussed. Secondly, a time discretization method for solving the adjoint equations numerically is presented. This is done due to the fact that the equations can hardly be solved analytically and in the case of using Euler parameters for the parametrization of rotations, singular mass matrices may arise.

3.3.4.1 Adjoint equations

A mechanical system consisting of rigid bodies, forces, and constraints acting between these bodies can be described by equations of motion in the following form already given in Eq. (2.47):

$$\begin{aligned} \mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} &= \mathbf{f}(\mathbf{q}, \dot{\mathbf{q}}, t) - \mathbf{C}_{\mathbf{q}}^{\mathsf{T}}\lambda \\ \mathbf{C}(\mathbf{q}) &= \mathbf{0} \\ \mathbf{y} &= \mathbf{y}(\mathbf{q}, \dot{\mathbf{q}}). \end{aligned} \tag{3.27}$$

Here, $\mathbf{q}$ denotes a vector of redundant generalized coordinates. They are subject to the holonomic constraints $\mathbf{C}(\mathbf{q}) = \mathbf{0}$, which enter the equations of motion via the Jacobian $\mathbf{C}_{\mathbf{q}}$ and the vector of Lagrange multipliers λ representing the constraint forces in the system. Moreover, the system may incorporate a vector of parameters ξ. For simplicity, we suppose that ξ appears only in the vector $\mathbf{f}$. However, the subsequent derivations may be extended to the case, where ξ appears in the mass

matrix and the constraint equations, too.
Using the additional variables $\mathbf{v} = \dot{\mathbf{q}}$, the equations of motion can be reformulated as a first order system of equations reading

$$\begin{aligned} \dot{\mathbf{q}} &= \mathbf{v} \\ \mathbf{M}(\mathbf{q})\dot{\mathbf{v}} &= \mathbf{f}(\mathbf{q}, \mathbf{v}, \xi, t) - \mathbf{C}_{\mathbf{q}}^{\mathsf{T}}\lambda \\ \mathbf{C}(\mathbf{q}) &= \mathbf{0} \\ \mathbf{y} &= \mathbf{y}(\mathbf{q}, \mathbf{v}). \end{aligned} \tag{3.28}$$

The state vector $\mathbf{x}$ introduced in the previous section now consists of $\mathbf{q}$ and $\mathbf{v}$. We want to derive a gradient formula like Eq. (3.21) for the multibody system. Without changing the function we may therefore augment J by the system equations in the following way:

$$J = \phi(\mathbf{y}(T)) + \int_0^T \Big[h(\mathbf{y}, t) + \mathbf{p}^{\mathsf{T}}(\dot{\mathbf{q}} - \mathbf{v}) + \mathbf{w}^{\mathsf{T}}(\mathbf{M}\dot{\mathbf{v}} - \mathbf{f} + \mathbf{C}_{\mathbf{q}}^{\mathsf{T}}\lambda) + \mu^{\mathsf{T}}\mathbf{C}\Big]\,\mathrm{d}t.$$

At this point, the variables $\mathbf{p}(t)$, $\mathbf{w}(t)$ and $\mu(t)$ may be chosen arbitrarily. The derivative of the cost function J with respect to the set of parameters ξ is given by

$$\begin{aligned} \frac{\partial J}{\partial \xi} = \int_0^T \Big\{ & h_{\mathbf{y}}\mathbf{y}_{\mathbf{q}}\mathbf{q}_\xi + h_{\mathbf{y}}\mathbf{y}_{\mathbf{v}}\mathbf{v}_\xi + \mathbf{p}^{\mathsf{T}}(\dot{\mathbf{q}}_\xi - \mathbf{v}_\xi) \\ & + \mathbf{w}^{\mathsf{T}}\Big[(\mathbf{M}\dot{\mathbf{v}})_{\mathbf{q}}\mathbf{q}_\xi + \mathbf{M}\dot{\mathbf{v}}_\xi - \mathbf{f}_{\mathbf{q}}\mathbf{q}_\xi - \mathbf{f}_{\mathbf{v}}\mathbf{v}_\xi - \mathbf{f}_\xi + \Big(\mathbf{C}_{\mathbf{q}}^{\mathsf{T}}\lambda\Big)_{\mathbf{q}}\mathbf{q}_\xi + \mathbf{C}_{\mathbf{q}}^{\mathsf{T}}\lambda_\xi\Big] \\ & + \mu^{\mathsf{T}}\mathbf{C}_{\mathbf{q}}\delta\mathbf{q}\Big\}\,\mathrm{d}t + \phi_{\mathbf{y}}\left(\mathbf{y}_{\mathbf{q}}\mathbf{q}_\xi + \mathbf{y}_{\mathbf{v}}\mathbf{v}_\xi\right)\Big|_{t=T}. \end{aligned} \tag{3.29}$$

Integration by parts of the terms including $\dot{\mathbf{q}}_\xi$ and $\dot{\mathbf{v}}_\xi$ is computed by

$$\begin{aligned} \int_0^T \mathbf{p}^{\mathsf{T}}\,\dot{\mathbf{q}}_\xi\,\mathrm{d}t &= -\int_0^T \dot{\mathbf{p}}^{\mathsf{T}}\mathbf{q}_\xi\,\mathrm{d}t + \mathbf{p}^{\mathsf{T}}\mathbf{q}_\xi\Big|_{t=T} \\ \int_0^T \mathbf{w}^{\mathsf{T}}\mathbf{M}\dot{\mathbf{v}}_\xi\,\mathrm{d}t &= -\int_0^T \frac{\mathrm{d}}{\mathrm{d}t}\Big(\mathbf{w}^{\mathsf{T}}\mathbf{M}\Big)\mathbf{v}_\xi\,\mathrm{d}t + \mathbf{w}^{\mathsf{T}}\mathbf{M}\mathbf{v}_\xi\Big|_{t=T} \end{aligned} \tag{3.30}$$

with $\mathbf{q}_\xi(0) = \mathbf{v}_\xi(0) = \mathbf{0}$ since initial conditions for the state variables are prescribed as $\mathbf{q}(0) = \mathbf{q}_0$ and $\mathbf{v}(0) = \mathbf{v}_0$. Using Eq. (3.30) and collecting the terms multiplied with $\mathbf{q}_\xi$, $\mathbf{v}_\xi$ and λ_ξ, the derivative of the cost function J given by Eq. (3.29) can be rewritten as

$$\begin{aligned} \frac{\partial J}{\partial \xi} = \int_0^T \Big\{ & \Big[h_{\mathbf{y}}\mathbf{y}_{\mathbf{q}} - \dot{\mathbf{p}}^{\mathsf{T}} + \mathbf{w}^{\mathsf{T}}\Big((\mathbf{M}\dot{\mathbf{v}})_{\mathbf{q}} - \mathbf{f}_{\mathbf{q}} + \Big(\mathbf{C}_{\mathbf{q}}^{\mathsf{T}}\lambda\Big)_{\mathbf{q}}\Big) + \mu^{\mathsf{T}}\mathbf{C}_{\mathbf{q}}\Big]\mathbf{q}_\xi \\ & + \Big[h_{\mathbf{y}}\mathbf{y}_{\mathbf{v}} - \mathbf{p}^{\mathsf{T}} - \mathbf{w}^{\mathsf{T}}\mathbf{f}_{\mathbf{v}} - \frac{\mathrm{d}}{\mathrm{d}t}\Big(\mathbf{w}^{\mathsf{T}}\mathbf{M}\Big)\Big]\mathbf{v}_\xi + \Big[\mathbf{w}^{\mathsf{T}}\mathbf{C}_{\mathbf{q}}^{\mathsf{T}}\Big]\lambda_\xi - \mathbf{w}^{\mathsf{T}}\mathbf{f}_\xi\Big\}\,\mathrm{d}t \\ & + \Big(\mathbf{p}^{\mathsf{T}} + \phi_{\mathbf{y}}\mathbf{y}_{\mathbf{q}}\Big)\mathbf{q}_\xi\Big|_{t=T} + \Big(\mathbf{w}^{\mathsf{T}}\mathbf{M} + \phi_{\mathbf{y}}\mathbf{y}_{\mathbf{v}}\Big)\mathbf{v}_\xi\Big|_{t=T} \end{aligned} \tag{3.31}$$

To eliminate the terms involving $\mathbf{q}_\xi$ and $\mathbf{v}_\xi$, the adjoint variables $\mathbf{p}$, $\mathbf{w}$, and μ may now be defined by equating the respective expressions in square brackets to zero. After transposing these expressions one obtains the following system of adjoint equations:

$$\begin{aligned}
\frac{\mathrm{d}\mathbf{p}}{\mathrm{d}t} &= \mathbf{y}_\mathbf{q}^\mathsf{T} h_\mathbf{y}^\mathsf{T} + \mathbf{A}\mathbf{w} + \mathbf{C}_\mathbf{q}^\mathsf{T}\mu && (3.32)\\
\frac{\mathrm{d}}{\mathrm{d}t}(\mathbf{M}\mathbf{w}) &= \mathbf{y}_\mathbf{v}^\mathsf{T} h_\mathbf{y}^\mathsf{T} - \mathbf{p} - \mathbf{B}\mathbf{w} && (3.33)\\
\mathbf{0} &= \mathbf{C}_\mathbf{q}\mathbf{w} && (3.34)\\
\mathbf{p}(T) &= -\mathbf{y}_\mathbf{q}^\mathsf{T}\phi_\mathbf{y}^\mathsf{T} && (3.35)\\
\mathbf{M}(T)\mathbf{w}(T) &= -\mathbf{y}_\mathbf{v}^\mathsf{T}\phi_\mathbf{y}^\mathsf{T} && (3.36)
\end{aligned}$$

where the abbreviations

$$\begin{aligned}
\mathbf{A} &= (\mathbf{M}\dot{\mathbf{v}})_\mathbf{q}^\mathsf{T} - \mathbf{f}_\mathbf{q}^\mathsf{T} + (\mathbf{C}_\mathbf{q}^\mathsf{T}\lambda)_\mathbf{q}^\mathsf{T} && (3.37)\\
\mathbf{B} &= \mathbf{f}_\mathbf{v}^\mathsf{T} && (3.38)
\end{aligned}$$

and the symmetry of the mass matrix $\mathbf{M} = \mathbf{M}^\mathsf{T}$ have been used. If Eqs. (3.32)–(3.36) are satisfied Eq. (3.31) reduces to

$$\nabla J = \left(\frac{\partial J}{\partial \xi}\right)^\mathsf{T} = -\int_0^T \mathbf{f}_\xi^\mathsf{T}\mathbf{w}\,\mathrm{d}t. \tag{3.39}$$

3.3.4.2 Consistent boundary conditions for the adjoint system

In general, the boundary condition Eq. (3.36) for the adjoint variable $\mathbf{w}$ is incompatible with the constraint equation Eq. (3.34) at $t = T$. Only when $\mathbf{y}_\mathbf{v}^\mathsf{T}\phi_\mathbf{y}^\mathsf{T} = \mathbf{0}$, i. e. when the Mayer term does not depend on $\mathbf{v}$, all equations are satisfied by setting $\mathbf{p}(T) = -\mathbf{y}_\mathbf{q}^\mathsf{T}\phi_\mathbf{y}^\mathsf{T}$ and $\mathbf{w}(T) = \mathbf{0}$. However, troubles occur if $\mathbf{y}_\mathbf{v}^\mathsf{T}\phi_\mathbf{y}^\mathsf{T} \neq \mathbf{0}$. To circumvent this problem an approach similar to the idea of Gear-Gupta-Leimkuhler [23] is applied. Using the constraint equations $\mathbf{C}(\mathbf{q}) = \mathbf{0}$ of the original system Eq. (3.27) at velocity level

$$\mathbf{C}_\mathbf{q}\dot{\mathbf{q}} = \mathbf{C}_\mathbf{q}(\mathbf{q})\mathbf{v} = \mathbf{0}$$

and forming the derivative with respect to the parameters ξ reads

$$\mathbf{C}_\mathbf{q}\mathbf{v}_\xi + (\mathbf{C}_\mathbf{q}\mathbf{v})_\mathbf{q}\,\mathbf{q}_\xi = \mathbf{0}.$$

Considering this relation at time $t = T$ and multiplying it with an arbitrary vector of numbers ξ results in

$$\xi^\mathsf{T}\left(\mathbf{C}_\mathbf{q}\mathbf{v}_\xi + (\mathbf{C}_\mathbf{q}\mathbf{v})_\mathbf{q}\,\mathbf{q}_\xi\right)\Big|_T = \mathbf{0}.$$

Since the expression on the left side is always zero it may be added to Eq. (3.31) without modifying the actual value of the cost function. Hereby, two additional boundary terms are generated in Eq. (3.31) transforming Eqs. (3.35) and (3.36) into

$$\begin{aligned} \mathbf{y}_{\mathbf{q}}^{\mathsf{T}}\phi_{\mathbf{y}}^{\mathsf{T}} + \mathbf{p} + (\mathbf{C}_{\mathbf{q}}\mathbf{v})_{\mathbf{q}}^{\mathsf{T}}\xi &= \mathbf{0} \quad \text{...at } t = T \qquad (3.40) \\ \mathbf{y}_{\mathbf{v}}^{\mathsf{T}}\phi_{\mathbf{y}}^{\mathsf{T}} + \mathbf{M}\mathbf{w} + \mathbf{C}_{\mathbf{q}}^{\mathsf{T}}\xi &= \mathbf{0} \quad \text{...at } t = T \qquad (3.41) \end{aligned}$$

By involving the still undetermined variable ξ, the adjoint variables $\mathbf{p}(T)$ and $\mathbf{w}(T)$ can be computed such that the constraint equation (3.34) is satisfied at $t = T$. For that purpose, the following system of equations has to be solved for $\mathbf{p}(T)$, $\mathbf{w}(T)$ and ξ:

$$\begin{bmatrix} \mathbf{I} & \mathbf{0} & (\mathbf{C}_{\mathbf{q}}\mathbf{v})_{\mathbf{q}}^{\mathsf{T}} \\ \mathbf{0} & \mathbf{M} & \mathbf{C}_{\mathbf{q}}^{\mathsf{T}} \\ \mathbf{0} & \mathbf{C}_{\mathbf{q}} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{p} \\ \mathbf{w} \\ \xi \end{bmatrix} = \begin{bmatrix} -\mathbf{y}_{\mathbf{q}}^{\mathsf{T}}\phi_{\mathbf{y}}^{\mathsf{T}} \\ -\mathbf{y}_{\mathbf{v}}^{\mathsf{T}}\phi_{\mathbf{y}}^{\mathsf{T}} \\ \mathbf{0} \end{bmatrix} \quad \text{...at } t = T \qquad (3.42)$$

In practice, one may first solve

$$\begin{bmatrix} \mathbf{M} & \mathbf{C}_{\mathbf{q}}^{\mathsf{T}} \\ \mathbf{C}_{\mathbf{q}} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{w} \\ \xi \end{bmatrix} = \begin{bmatrix} -\mathbf{y}_{\mathbf{v}}^{\mathsf{T}}\phi_{\mathbf{y}}^{\mathsf{T}} \\ \mathbf{0} \end{bmatrix} \quad \text{...at } t = T \qquad (3.43)$$

yielding ξ and $\mathbf{w}(T)$, and subsequently compute $\mathbf{p}(T)$ from

$$\mathbf{p} = -\left(\mathbf{C}_{\mathbf{q}}\mathbf{v}\right)_{\mathbf{q}}^{\mathsf{T}}\xi - \mathbf{y}_{\mathbf{q}}^{\mathsf{T}}\phi_{\mathbf{y}}^{\mathsf{T}} \quad \text{...at } t = T. \qquad (3.44)$$

Once $\mathbf{w}(T)$ and $\mathbf{p}(T)$ has been determined in this way, the differential-algebraic set of adjoint equations (3.32)–(3.34) may be solved to obtain $\mathbf{w}$, $\mathbf{p}$ and μ at every time instant in the interval $[0, T]$.

3.3.4.3 Remarks on the computation of Jacobian matrices A and B

For the adjoint equations of a multibody system the matrices $\mathbf{M}$, $\mathbf{C}_{\mathbf{q}}$, $\mathbf{A}$ and $\mathbf{B}$ from Eqs. (3.37) and (3.38) are required along a forward simulation of the equations of motion. Whereas $\mathbf{M}$ and $\mathbf{C}_{\mathbf{q}}$ also appear in the equations of motion and are therefore available, the determination of the Jacobian matrices $\mathbf{A}$ and $\mathbf{B}$ requires additional computational effort.

Basically, the Jacobians can be computed in three ways: First, the derivatives are computed exactly by implementing explicit formulas in the MBS software. Second, the computation of the derivatives may be done numerically by substituting the derivatives by finite difference quotients. Finally, the derivatives could also be determined by the technique of automatic differentiation, see, e. g., [19]. For complex multibody systems, the first way seems expensive and susceptible to programming errors. However, if a highly redundant formulation of the equations of motion is used, this way becomes attractive. Therefore it is advisable to assign full rotational and translational degrees of freedom to every body of the system and to describe the kinematic coupling between the bodies by constraint equations. Moreover, the use of Euler parameters for the rotational motions simplifies the system matrices such that programming explicit formulas for $\mathbf{A}$ and $\mathbf{B}$ becomes the most efficient strategy. It should also be noted, that the Jacobian matrices $(\mathbf{M}\dot{\mathbf{v}})_{\mathbf{q}}$, $\mathbf{f}_{\mathbf{q}}$, $\mathbf{f}_{\mathbf{v}}$ and $(\mathbf{C}_{\mathbf{q}}^{\mathsf{T}}\lambda)_{\mathbf{q}}$ may be required already for the simulation of the multibody system, if an implicit integration scheme such as, e. g., the HHT-algorithm [32, 50] is applied, see Section 2.4.2. Hence, an efficient computation of these matrices is crucial also for solving the equations of motion in forward direction.

3.3.4.4 A backward differentiation scheme for the adjoint system

Since Eqs. (3.32)–(3.34) must be solved backward in the physical time $t \in [0, T]$ it is advantageous to introduce a new time coordinate τ running also from $\tau = 0$ to $\tau = T$, before a time discretization scheme is developed. The transformation

$$\tau = T - t, \quad \tau \in [0, T], \quad \frac{\mathrm{d}}{\mathrm{d}t} = \frac{\mathrm{d}}{\mathrm{d}\tau}\frac{\mathrm{d}\tau}{\mathrm{d}t} = -\frac{\mathrm{d}}{\mathrm{d}\tau}$$

converts the adjoint Eqs. (3.32–3.34) into

$$\begin{aligned} \frac{\mathrm{d}\mathbf{p}}{\mathrm{d}\tau} &= -\mathbf{y}_{\mathbf{q}}^{\mathsf{T}}(\tau)\, h_{\mathbf{y}}^{\mathsf{T}}(\tau) - \mathbf{A}(\tau)\mathbf{w} - \mathbf{C}_{\mathbf{q}}^{\mathsf{T}}(\tau)\mu \\ \frac{\mathrm{d}}{\mathrm{d}\tau}\left(\mathbf{M}(\tau)\mathbf{w}\right) &= -\mathbf{y}_{\mathbf{v}}^{\mathsf{T}}(\tau)\, h_{\mathbf{y}}^{\mathsf{T}}(\tau) + \mathbf{p} + \mathbf{B}(\tau)\mathbf{w} \\ \mathbf{0} &= \mathbf{C}_{\mathbf{q}}(\tau)\mathbf{w} \end{aligned} \tag{3.45}$$

where $\mathbf{M}(\tau)$, $\mathbf{A}(\tau)$, $\mathbf{B}(\tau)$, $\mathbf{C}_{\mathbf{q}}(\tau)$, $\mathbf{y}_{\mathbf{q}}(\tau)$, $\mathbf{y}_{\mathbf{v}}(\tau)$, and $h_{\mathbf{y}}(\tau)$ have to be computed for $\mathbf{q}(\tau) = \mathbf{q}(T - t)$, $\mathbf{v}(\tau) = \mathbf{v}(T - t)$ and $\lambda(\tau) = \lambda(T - t)$ resulting from a forward simulation of the equations of motion Eq. (3.28).
For the numerical solution of Eq. (3.45) at the time instances $\tau_n = n\gamma$, $n = 1 \ldots N_t$, $\gamma = T/N_t$, a backward differentiation scheme approximating the derivative of a function $F(\tau)$ at a time instance τ_n by using the function values at $\tau_n, \tau_{n-1}, \ldots, \tau_{n-k}$ is proposed. The backward differentiation formula (BDF) reads

$$\left.\frac{\mathrm{d}F}{\mathrm{d}\tau}\right|_{\tau_n} \approx \frac{1}{\gamma}\sum_{i=0}^{k} \alpha_i F(\tau_{n-i}). \tag{3.46}$$

The coefficients α_i result from differentiating an interpolation polynomial through $F(\tau_n), \ldots, F(\tau_{n-k})$ and are chosen as the standard coefficients presented, e.g., in

[64, p.349]. Considering Eqs. (3.45) at $\tau = \tau_n$ and inserting the BDF-approximation for $\mathrm{d}\mathbf{p}/\mathrm{d}\tau$ and $\mathrm{d}(\mathbf{M}\mathbf{w})/\mathrm{d}\tau$, the following set of algebraic equations for $\mathbf{p}(\tau_n)$, $\mathbf{w}(\tau_n)$ and $\mu(\tau_n)$ is obtained:

$$\begin{aligned}\frac{1}{\gamma}\Big(\alpha_0\mathbf{p}(\tau_n) + \sum_{i=1}^{k}\alpha_i\mathbf{p}(\tau_{n-i})\Big) &= \\ -\,\mathbf{y}_{\mathbf{q}}^{\mathsf{T}}h_{\mathbf{y}}^{\mathsf{T}}(\tau_n) - \mathbf{A}(\tau_n)\mathbf{w}(\tau_n) &- \mathbf{C}_{\mathbf{q}}^{\mathsf{T}}(\tau_n)\mu(\tau_n)\end{aligned} \tag{3.47}$$

$$\begin{aligned}\frac{1}{\gamma}\left(\alpha_0\mathbf{M}(\tau_n)\mathbf{w}(\tau_n) + \sum_{i=1}^{k}\alpha_i\mathbf{M}(\tau_{n-i})\mathbf{w}(\tau_{n-i})\right) &= \\ -\,\mathbf{y}_{\mathbf{v}}^{\mathsf{T}}h_{\mathbf{y}}^{\mathsf{T}}(\tau_n) + \mathbf{p}(\tau_n) &+ \mathbf{B}(\tau_n)\mathbf{w}(\tau_n)\end{aligned} \tag{3.48}$$

$$\mathbf{C}_{\mathbf{q}}(\tau_n)\mathbf{w}(\tau_n) = \mathbf{0} \tag{3.49}$$

where $\mathbf{p}(\tau_{n-i})$, $\mathbf{w}(\tau_{n-i})$, and $\mu(\tau_{n-i})$ is supposed to be known for $i > 0$ from previous integration steps. Notice that it is not necessary to differentiate the term $d(\mathbf{M}\mathbf{w})/d\tau$ by using the product rule, which would require the additional term $\mathrm{d}\mathbf{M}/\mathrm{d}\tau$ to be computed from the forward simulation.

Since the adjoint system in Eq. (3.45) is linear in $\mathbf{p}$, $\mathbf{w}$, and μ the discretized system given by Eqs. (3.47)–(3.49) is also linear in $\mathbf{p}(\tau_n)$, $\mathbf{w}(\tau_n)$, and $\mu(\tau_n)$. Moreover, $\mathbf{p}(\tau_n)$ may be eliminated by solving Eq. (3.47) for

$$\begin{aligned}\mathbf{p}(\tau_n) = &-\frac{\gamma}{\alpha_0}\Big(\mathbf{y}_{\mathbf{q}}^{\mathsf{T}}h_{\mathbf{y}}^{\mathsf{T}}(\tau_n) + \mathbf{A}(\tau_n)\mathbf{w}(\tau_n) + \mathbf{C}_{\mathbf{q}}^{\mathsf{T}}(\tau_n)\mu(\tau_n)\Big) \\ &-\frac{1}{\alpha_0}\sum_{i=1}^{k}\alpha_i\mathbf{p}(\tau_{n-i}).\end{aligned} \tag{3.50}$$

Inserting into Eq. (3.48) yields

$$\begin{aligned}&\frac{1}{\gamma}\left(\alpha_0\mathbf{M}(\tau_n)\mathbf{w}(\tau_n) + \sum_{i=1}^{k}\alpha_i\mathbf{M}(\tau_{n-i})\mathbf{w}(\tau_{n-i})\right) = \\ &\quad -\,\mathbf{y}_{\mathbf{v}}^{\mathsf{T}}h_{\mathbf{y}}^{\mathsf{T}}(\tau_n) - \frac{\gamma}{\alpha_0}\Big(\mathbf{y}_{\mathbf{q}}^{\mathsf{T}}h_{\mathbf{y}}^{\mathsf{T}}(\tau_n) + \mathbf{A}(\tau_n)\mathbf{w}(\tau_n) + \mathbf{C}_{\mathbf{q}}^{\mathsf{T}}(\tau_n)\mu(\tau_n)\Big) \\ &\quad -\frac{1}{\alpha_0}\sum_{i=1}^{k}\alpha_i\mathbf{p}(\tau_{n-i}) + \mathbf{B}(\tau_n)\mathbf{w}(\tau_n)\end{aligned}$$

or after rearranging and multiplying with $\alpha_0\gamma$

$$\begin{aligned}&\Big(\alpha_0^2\mathbf{M}(\tau_n) + \gamma^2\mathbf{A}(\tau_n) - \alpha_0\gamma\mathbf{B}(\tau_n)\Big)\mathbf{w}(\tau_n) + \gamma^2\mathbf{C}_{\mathbf{q}}^{\mathsf{T}}(\tau_n)\mu(\tau_n) = \\ &\quad -\,\alpha_0\gamma\mathbf{y}_{\mathbf{v}}^{\mathsf{T}}h_{\mathbf{y}}^{\mathsf{T}}(\tau_n) - \gamma^2\mathbf{y}_{\mathbf{y}}^{\mathsf{T}}h_{\mathbf{y}}^{\mathsf{T}}(\tau_n) \\ &\quad -\sum_{i=1}^{k}\alpha_i\left(\gamma\mathbf{p}(\tau_{n-i}) + \alpha_0\mathbf{M}(\tau_{n-i})\mathbf{w}(\tau_{n-i})\right).\end{aligned} \tag{3.51}$$

Using the abbreviations

$$\mathbf{W}(\tau_n) = \alpha_0^2\mathbf{M}(\tau_n) + \gamma^2\mathbf{A}(\tau_n) - \alpha_0\gamma\mathbf{B}(\tau_n)$$

$$\mathbf{r}(\tau_n) = -\alpha_0\gamma\mathbf{y}_{\mathbf{v}}^{\mathsf{T}}h_{\mathbf{y}}^{\mathsf{T}}(\tau_n) - \gamma^2\mathbf{y}_{\mathbf{q}}^{\mathsf{T}}h_{\mathbf{y}}^{\mathsf{T}}(\tau_n) - \sum_{i=1}^{k}\alpha_i\left(\gamma\mathbf{p}(\tau_{n-i}) + \alpha_0\mathbf{M}(\tau_{n-i})\mathbf{w}(\tau_{n-i})\right)$$

Eqs. (3.51) and (3.49) may be summarized in the following matrix equation for $\mathbf{w}(\tau_n)$ and $\mu(\tau_n)$:

$$\begin{bmatrix}\mathbf{W}(\tau_n) & \gamma^2\mathbf{C}_{\mathbf{q}}^{\mathsf{T}}(\tau_n)\\ \mathbf{C}_{\mathbf{q}}(\tau_n) & \mathbf{0}\end{bmatrix}\begin{bmatrix}\mathbf{w}(\tau_n)\\ \mu(\tau_n)\end{bmatrix} = \begin{bmatrix}\mathbf{r}(\tau_n)\\ \mathbf{0}\end{bmatrix} \tag{3.52}$$

After solving this equation, the second adjoint variable $\mathbf{p}(\tau_n)$ may be computed from Eq. (3.50). The integration process can be started by choosing the integration order $k = 1$. The according initial conditions may be computed by invoking the equation system derived in Section 3.3.4.2. $\mathbf{p}(\tau_0) = \mathbf{p}(T)$ and $\mathbf{w}(\tau_0) = \mathbf{w}(T)$.

3.3.4.5 Summary: The adjoint gradient computation

Finally, the steps necessary to compute the gradient of the cost function J using the adjoint sensitivity analysis are summarized. Again ξ denotes the given vector of parameters of a multibody system.

1. Solve the equations of motion Eq. (3.28) forward in time $t \in [0, T]$ yielding $\mathbf{q}(t)$, $\mathbf{v}(t)$, $\lambda(t)$. This may be done, e.g. by choosing the Hilbert-Hughes-Taylor (HHT) integration scheme, as proposed in [32] and its application for a differential algebraic system given in an index three formulation in [50].

2. Just for information, the objective function J may already be computed by inserting $\mathbf{q}(t)$ and $\mathbf{v}(t)$ into Eq. (3.16). Note, that the integration must be done numerically.

3. Along the forward simulation of the equations of motion compute the mass matrix $\mathbf{M}(t)$, the constraint Jacobian $\mathbf{C}_{\mathbf{q}}(t)$ and the Jacobian matrices $\mathbf{A}(t)$, $\mathbf{B}(t)$ from Eqs. (3.37) and (3.38). For lack of computer memory it might be impossible to store these matrices at a sufficiently high number of time instances. In this case, one must provide formulas to compute these matrices from $\mathbf{q}$, $\mathbf{v}$ and ξ on demand.

4. Solve the adjoint system Eq. (3.45) for $\mathbf{p}(\tau)$, $\mathbf{w}(\tau)$ and $\mu(\tau)$, where $\tau = T - t$. A numerical solution at time instances τ_n can be computed from Eqs. (3.52) and (3.50), where the integration order k must be chosen equal to one in the first step and may be increased afterwards. The initial conditions for the adjoint system may be computed according to Eq. (3.42).

5. Compute the adjoint variables as functions of the original time by setting $\mathbf{p}(t) = \mathbf{p}(\tau = T - t)$ and $\mathbf{w}(t) = \mathbf{w}(\tau = T - t)$. Moreover, determine $\mathbf{f}_\xi(t)$ along the forward simulation.
6. From Eq. (3.39) the gradient of J may finally be derived.

CHAPTER 4

Identification of Multibody System Parameters

Up to this point the theory for analyzing the sensitivities of dynamical systems with respect to changes of parameters was treated and derived using different approaches. As given in the title of the thesis the focus lies in utilizing adjoint methods for the identification of multibody system parameters. Basically the term *parameter identification* indicates that there are correct parameters in terms of a real system, which should be found. As they have to be identified they are not directly known, although their influence onto a system's behavior can be monitored. Comparing system outputs of the real and the virtual system allows to adjust the parameters of the virtual system such that they match as good as possible. For solving this sort of problem many approaches exist. In this chapter the focus will lie on iterative approaches using the result of the adjoint sensitivity analysis derived in the preceding chapter. Nevertheless other methods are introduced briefly, too.
In the field of control engineering *Recursive least squares filter* and *Extended Kalman filter* are quite common. Here the main goal is to enable online identification, such that parameters can be identified during the operation of a controlled machine, etc. A good review of techniques used in control engineering is given in the work of Rao and Unbehauen [56]. Although these methods feature fast performance, their practicability for non-linear differential algebraic equations is limited. Due to this, these methods hardly occur in the field of multibody dynamics.
Before going into detail with methods typically used for parameter identification in multibody dynamics, the results derived above are summarized. Using a suitable performance measure J, the output of the multibody system can be validated with a single scalar value. Recalling the general definitions from Section 3.2 the

performance measure may read

$$J = \int_0^T h\left(t, \mathbf{y}(\mathbf{x})\right) \mathrm{d}t,$$

where h is formulated such that it measures the agreement between real and virtual system outputs, $\bar{\mathbf{y}}$ and $\mathbf{y}$ respectively. By doing so, the set of parameters evaluating J to a minimum is the identification result in terms of the measurements taken into account. As the gradient ∇J, computed by use of the adjoint sensitivity analysis, is demanded to be zero in a local minimum, the optimization problem reads

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}, \xi, t) & \mathbf{x}(0) &= \mathbf{x}_0 \\ \dot{\mathbf{p}} &= -y_{\mathbf{x}}^{\mathsf{T}} h_{\mathbf{y}}^{\mathsf{T}} - \mathbf{f}_{\mathbf{p}}\mathbf{x}^{\mathsf{T}}\mathbf{p} & \mathbf{p}(T) &= \mathbf{0} \\ \nabla J &= \int_0^T \mathbf{f}_\xi^{\mathsf{T}} \mathbf{p} \, \mathrm{d}t = \mathbf{0} \end{aligned} \tag{4.1}$$

in case of a dynamical system given in minimal coordinate description. Usually this type of problem is denoted as two-point boundary value problem. There are solution strategies aiming at directly solving this kind of problem by using single, or multiple shooting methods. However, the integral condition $\mathbf{0} = \int_0^T \mathbf{f}_\xi^{\mathsf{T}} \mathbf{p} \, \mathrm{d}t$ contained in Eq. (4.1) does not allow the application of standard solvers. By introducing the set of parameters ξ as a time-dependent function $\xi(t)$, the equations of motion read

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}, \xi, t) \quad \mathbf{x}(0) = \mathbf{x}_0 \\ \dot{\xi} &= \mathbf{0}. \end{aligned}$$

Following the idea presented in Section 3.3.4, the extended cost functional reads

$$J = \int_0^T \left\{ h(t, \mathbf{y}) + \mathbf{p}^{\mathsf{T}} \left(\dot{\mathbf{x}} - \mathbf{f} \right) + \mathbf{w}^{\mathsf{T}} \dot{\xi} \right\} \mathrm{d}t,$$

where the arbitrary multipliers $\mathbf{p}(t)$ and $\mathbf{w}(t)$ are introduced. As J represents a functional depending on the functions $\xi(t)$, the optimality condition can be derived by using the Gâteaux derivative (see, e.g. [44, p. 407])

$$\delta J(\xi; \eta) := \lim_{\varepsilon \to 0} \frac{J(\xi + \varepsilon\eta) - J(\xi)}{\varepsilon} = \frac{\partial}{\partial \varepsilon} J(\xi + \varepsilon\eta) \bigg|_{\varepsilon = 0},$$

where η denotes an admissible test-function. As the optimal set of parameters results in a variation $\delta J = 0$ and therefore in

$$\begin{aligned} \delta J &= \int_0^T \left\{ h_{\mathbf{y}} \mathbf{y}_{\mathbf{x}} \delta\mathbf{x} + \mathbf{p}^{\mathsf{T}} \left(\delta\dot{\mathbf{x}} - \mathbf{f}_{\mathbf{x}} \delta\mathbf{x} - \mathbf{f}_\xi \delta\xi \right) + \mathbf{w}^{\mathsf{T}} \delta\dot{\xi} \right\} \mathrm{d}t \\ &= \int_0^T \left\{ \left[h_{\mathbf{y}} \mathbf{y}_{\mathbf{x}} - \dot{\mathbf{p}}^{\mathsf{T}} - \mathbf{p}^{\mathsf{T}} \mathbf{f}_{\mathbf{x}} \right] \delta\mathbf{x} + \left[-\mathbf{p}^{\mathsf{T}} \mathbf{f}_\xi - \dot{\mathbf{w}}^{\mathsf{T}} \right] \delta\xi \right\} \mathrm{d}t \\ &\quad + \mathbf{p}^{\mathsf{T}} \delta\mathbf{x} \Big|_0^T + \mathbf{w}^{\mathsf{T}} \delta\xi \Big|_0^T = 0 \end{aligned}$$

the two-point boundary value problem reads:

$$\begin{aligned} \dot{\mathbf{p}} &= -\mathbf{f}_{\mathbf{x}}^{\mathsf{T}}\mathbf{p} + \mathbf{y}_{\mathbf{x}}^{\mathsf{T}} h_{\mathbf{y}}^{\mathsf{T}} & \qquad \mathbf{p}(T) &= \mathbf{0} \\ \dot{\mathbf{w}} &= -\mathbf{f}_{\xi}^{\mathsf{T}}\mathbf{p} & \mathbf{w}(0) &= \mathbf{0} \\ \dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}, \xi, t) & \mathbf{x}(0) &= \mathbf{x}_0 \\ \dot{\xi} &= \mathbf{0} & \mathbf{w}(T) &= \mathbf{0}. \end{aligned}$$

Solving this kind of problem may be done by using common solver packages (Matlab bvp4c, Boundsco). In general, the convergence of such solvers strongly depends on the initial conditions supplied for $\mathbf{x}(t)$, $\mathbf{p}(t)$, $\mathbf{w}(t)$, and $\xi(t)$. Using homotopy approaches may help to improve the convergence, nevertheless finding a valid solution remains tricky. Therefore it seems more promising to adopt an optimization algorithm for finding the minimum of J iteratively by using the gradient information ∇J.

4.1 Identification of Parameters using Measurements in Time Domain

Since both simulated and experimental data are present at discrete time points, it seems obvious to directly account for deviations between them. In order to quantify the difference, typically the RMS error, already given in Eq. (3.11), is used. In this case the cost function used for optimization reads

$$J(\xi) = \frac{1}{2}\int_0^T (\mathbf{y} - \bar{\mathbf{y}})^{\mathsf{T}} (\mathbf{y} - \bar{\mathbf{y}})\,\mathrm{d}t, \tag{4.2}$$

where $\mathbf{y}$ is the vector of system outputs generated by simulation and $\bar{\mathbf{y}}$ denotes the measured ones. Transforming the integral into a sum of $N_T = {}^{T}\!/_{\Delta t}$ finite time steps yields

$$J(\xi) \approx \frac{1}{2}\Delta t \sum_{i=1}^{N_T} \left[(\mathbf{y}_i - \bar{\mathbf{y}}_i)^{\mathsf{T}} (\mathbf{y}_i - \bar{\mathbf{y}}_i) \right],$$

with $\mathbf{y}_i := \mathbf{y}(t_i)$ and $\bar{\mathbf{y}}_i := \bar{\mathbf{y}}(t_i)$ being the evaluations of $\mathbf{y}$ and $\bar{\mathbf{y}}$ at a time point t_i. The discrete time step size Δt is constant, as measurement systems typically use fixed sampling rates.

Similarly a transformation of the gradient in Eq. (3.39), derived in Section 3.3.4 by the use of the adjoint sensitivity analysis, may be performed. The sum representing an approximation of the cost function's gradient reads

$$\nabla J \approx \Delta t \sum_{i=1}^{N_T} \left(-\mathbf{f}_{\xi,i}^{\mathsf{T}} \mathbf{w}_i \right),$$

where $\mathbf{w}_i$ denotes the evaluation of the adjoint variables at $t = t_i$ and $\mathbf{f}_{\xi,i}$ being the Jacobian with respect to the parameters evaluated at this time point. Using this gradient formula within an iterative optimization algorithm and hence locating a minimum of J results in an identification result ξ^*.

4.2 Identification of Parameters using Measured Spectra

In many industrial applications, the identification of system parameters by matching measurements with simulated system outputs in the time domain does not lead to satisfying results. Especially, this is the case when some low-frequency content in measured signals cannot be reached by the simulation model. During the identification process, parameters are modified such that the low-frequency content reaches a best-fit solution, but other frequencies are not represented well, because of their negligible impact on to the cost function. Hence, the goal of the identification is in general to fit a special frequency range. In [7], a system identification for vehicle dynamic applications has been presented based on impulse–momentum equations using a transfer function written as a frequency response function in order to take into account low and high frequency ranges. Spectral element techniques for parameter identification can also be found in the field of layered media in structural dynamics [1]. Therein, the characteristic function of the system, combining the response and impulse force function of the system, is represented in the frequency domain. The transfer function which characterizes the system in the frequency domain is then given as a Fourier transformation. The wavelet transform is used in [39] as a time–frequency representation for the determination of modal parameters of a vibrating system. Therein, natural frequencies, damping ratios, and mode shapes are estimated in the time domain from output data only. A wavelet-based approach for parameter identification is as well presented in [24]. Systems with cubic nonlinearities and systems undergoing both continuous and stick/slip motion have been addressed therein.

The latter mentioned works emphasize the importance of the spectral analysis of the system in order to understand the behavior of the system and consequently be capable of efficient parameter identification. Considering this situation, a redefinition of the cost-function, by means of using frequency domain information, seems meaningful. Within the following section an identification method using frequency information only is introduced for the use in multibody dynamics. A combination of the adjoint sensitivity computation and classical Fourier analysis for the identification of the amplitude response is presented as a novel approach.

Let us consider the system equations of motion in first order form

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}, \xi, t) \quad \mathbf{x}(0) = \mathbf{x}_0 \tag{4.3}$$

where ξ may describe the unknown parameters of the system. For the sake of simplicity we assume the system has only one output which depends on the states $y(t) := g(\mathbf{x})$. By applying classical Fourier analysis, $y(t)$ can be approximated by

$$y(t) \approx \frac{1}{2}A_0 + \sum_{k=1}^{N} \Big(A_k \cos(\omega_k t) + B_k \sin(\omega_k t) \Big)$$

in which ω_k represents the angular frequency of the kth harmonics, each of which is assigned to the appropriate value of its amplitude $\sqrt{A_k^2 + B_k^2}$. The corresponding Fourier coefficients A_k and B_k are defined by

$$A_k = \frac{2}{T} \int_0^T y(t) \cos(\omega_k t) \mathrm{d}t \tag{4.4}$$

$$B_k = \frac{2}{T} \int_0^T y(t) \sin(\omega_k t) \mathrm{d}t \tag{4.5}$$

For example, it might be of interest to determine a set of parameters ξ in such a way that the measured amplitudes $\sqrt{\bar{A}_k^2 + \bar{B}_k^2}$, $k = 1, \ldots, N$ are best approximated by the amplitudes of the system. Therefore, the goal is to find ξ such that an error function of the form

$$J = \frac{1}{4} \sum_{k=1}^{N} \left[A_k^2 + B_k^2 - \left(\bar{A}_k^2 + \bar{B}_k^2 \right) \right]^2 \tag{4.6}$$

is minimized. However, the problem may as well be specified in Mayer form (see Section 3.2). For that purpose, the Fourier coefficients are introduced by the differential equations

$$\dot{a}_k(t) = \frac{2}{T} y(t) \cos(\omega_k t) = \frac{2}{T} g(\mathbf{x}) \cos(\omega_k t) \tag{4.7}$$

$$\dot{b}_k(t) = \frac{2}{T} y(t) \sin(\omega_k t) = \frac{2}{T} g(\mathbf{x}) \sin(\omega_k t) \tag{4.8}$$

with the corresponding initial conditions $a_k(0) = b_k(0) = 0$, yielding $A_k = a_k(T)$ and $B_k = b_k(T)$ defined in Eqs. (4.4) and (4.5). Hence, the cost function is considered as a function of the final values of a_k and b_k, i.e., $J = J(A_k, B_k)$.

4.2.1 Remark on window functions

Describing a signal $y(t)$ by means of a Fourier series requires the function to be periodically continuable. Window functions therefore allow to transform non-periodic into periodic functions. In place of several window functions given in

the literature [28], the *Hanning window* and it's characteristics is presented in the following.
Let us consider a signal of the form

$$g(t) = \frac{a_0}{2} + \sum_{k=1}^{N} \left[a_k \cos(\omega_k t) + b_k \sin(\omega_k t)\right]$$

with $\omega_k = k\omega = k\frac{2\pi}{T}$ and the period T. Using the Hanning window function given in [28]

$$\eta(t) = \frac{1}{2}\left[1 - \cos\left(\frac{2\pi}{nT}\, t\right)\right],$$

the kth Fourier coefficient ($k \geq 1$) can be computed by

$$\tilde{a}_k = \frac{2}{n\,T}\int_0^{n\,T} g(t)\eta(t)\cos(k\,\omega\,t)\mathrm{d}t$$

where n is the number of periods to be considered. For $n = 1$ the Fourier coefficient $\tilde{a}_k$ is

$$\tilde{a}_k = \lim_{n\to 1}\frac{2}{n\,T}\int_0^{n\,T} g(t)\eta(t)\cos(k\,\omega\,t)\mathrm{d}t = \frac{1}{4}\left(-a_{k-1} + 2a_k - a_{k+1}\right) \tag{4.9}$$

and for $n > 1$ the Fourier coefficient $\tilde{a}_k$ is

$$\tilde{a}_k = \frac{2}{n\,T}\int_0^{n\,T} g(t)\eta(t)\cos(k\,\omega\,t)\mathrm{d}t = \frac{1}{2}a_k \quad n > 1. \tag{4.10}$$

Equations (4.9) and (4.10) show that more than one period T is required when using the Hanning window function. Otherwise, the amplitude $\tilde{a}_i$ includes information from other harmonics, which leads to wrong results. Furthermore, the amplitude $\tilde{a}_k$ has to be corrected by a factor of 2 to be comparable with true amplitudes.

4.2.2 The adjoint gradient computation in terms of Fourier coefficients

Following the basic idea presented in Chapter 3, the adjoint sensitivity analysis is applied to the cost function in Eq. (4.6). In a first step, the cost function is enhanced by the system equations in Eqs. (4.3), (4.7), and (4.8) leading to

$$\begin{aligned}\overline{J} = J(A_k, B_k) &+ \int_0^T \mathbf{p}^\mathsf{T}\left(\mathbf{f}(\mathbf{x}, \xi, t) - \dot{\mathbf{x}}\right)\,\mathrm{d}t \\ &+ \sum_{k=1}^{N}\int_0^T \alpha_k\left(\frac{2}{T}g(\mathbf{x})\cos(\omega_k t) - \dot{a}_k(t)\right)\,\mathrm{d}t \\ &+ \sum_{k=1}^{N}\int_0^T \beta_k\left(\frac{2}{T}g(\mathbf{x})\sin(\omega_k t) - \dot{b}_k(t)\right)\,\mathrm{d}t.\end{aligned} \tag{4.11}$$

Herein $\mathbf{p}$ represents the vector of adjoint variables corresponding to the state vector. Moreover, α_k and β_k with $k = 1, \ldots, N$ are the adjoint variables related to the Fourier coefficients. Note that $\mathbf{p}(t)$, $\alpha_k(t)$, and $\beta_k(t)$ can be arbitrary time functions at this point, since $\overline{J} = J$, if Eqs. (4.3), (4.7), and (4.8) are satisfied.

Let us now consider a forward solution $\mathbf{x}(t)$ of the system equations in Eq. (4.3) for a set of parameters ξ. Applying direct differentiation with respect to the parameters ξ onto Eq. (4.11) results in

$$\begin{aligned}\frac{\partial \overline{J}}{\partial \xi} = \left(\nabla \overline{J}\right)^{\mathsf{T}} = \sum_{k=1}^{N} \frac{\partial J}{\partial A_k} A_{k,\xi} + \sum_{k=1}^{N} \frac{\partial J}{\partial B_k} B_{k,\xi} &+ \int_0^T \mathbf{p}^{\mathsf{T}} \left(\mathbf{f}_{\mathbf{x}} \mathbf{x}_\xi + \mathbf{f}_\xi - \dot{\mathbf{x}}_\xi\right) \mathrm{d}t \\ &+ \sum_{k=1}^{N} \int_0^T \left(\frac{2}{T} \alpha_k g_{\mathbf{x}} \mathbf{x}_\xi \cos(\omega_k t) - \alpha_k \dot{a}_{k,\xi}\right) \mathrm{d}t \\ &+ \sum_{k=1}^{N} \int_0^T \left(\frac{2}{T} \beta_k g_{\mathbf{x}} \mathbf{x}_\xi \sin(\omega_k t) - \beta_k \dot{b}_{k,\xi}\right) \mathrm{d}t\end{aligned} \tag{4.12}$$

where the sensitivities $A_{k,\xi}$, $B_{k,\xi}$, $\mathbf{x}_\xi$, $a_{k,\xi}$, and $b_{k,\xi}$ are introduced. After applying integration by parts to the terms including $\dot{\mathbf{x}}_\xi$, $\dot{a}_{k,\xi}$, and $\dot{b}_{k,\xi}$, the gradient can be written in the form

$$\begin{aligned}\left(\nabla \overline{J}\right)^{\mathsf{T}} = &\sum_{k=1}^{N} \frac{\partial J}{\partial A_k} A_{k,\xi} + \sum_{k=1}^{N} \frac{\partial J}{\partial B_k} B_{k,\xi} \\ &+ \int_0^T \mathbf{p}^{\mathsf{T}} \left(\mathbf{f}_{\mathbf{x}} \mathbf{x}_\xi + \mathbf{f}_\xi\right) \mathrm{d}t + \int_0^T \dot{\mathbf{p}}^{\mathsf{T}} \mathbf{x}_\xi \, \mathrm{d}t - \mathbf{p}^{\mathsf{T}}(T) \mathbf{x}_\xi(T) \\ &+ \sum_{k=1}^{N} \left(\int_0^T \frac{2}{T} \alpha_k g_{\mathbf{x}} \mathbf{x}_\xi \cos(\omega_k t) \, \mathrm{d}t + \int_0^T \dot{\alpha}_k a_{k,\xi} \, \mathrm{d}t - \alpha_k(T) A_{k,\xi}\right) \\ &+ \sum_{k=1}^{N} \left(\int_0^T \frac{2}{T} \beta_k g_{\mathbf{x}} \mathbf{x}_\xi \sin(\omega_k t) \, \mathrm{d}t + \int_0^T \dot{\beta}_k b_{k,\xi} \, \mathrm{d}t - \beta_k(T) B_{k,\xi}\right)\end{aligned} \tag{4.13}$$

where the initial conditions $\mathbf{x}_\xi(0) = \mathbf{0}$, $a_{k,\xi}(0) = b_{k,\xi}(0) = 0$ and end conditions $a_{k,\xi}(T) = A_{k,\xi}$, $b_{k,\xi}(T) = B_{k,\xi}$ are used. The computation of $\mathbf{x}_\xi$, $a_{k,\xi}$, and $b_{k,\xi}$ can be circumvented, if the factors multiplied vanish. First, the terms including $a_{k,\xi}$ and $b_{k,\xi}$ disappear, if $\dot{\alpha}_k = \dot{\beta}_k = 0$, i.e., if $\alpha_k = \text{const.} = \alpha_k(T)$ and $\beta_k = \text{const.} = \beta_k(T)$. Second, the terms including $\mathbf{x}_\xi$ vanish, if the adjoint variables $\mathbf{p}$ are defined by

$$\dot{\mathbf{p}} = -\mathbf{f}_{\mathbf{x}}^{\mathsf{T}} \mathbf{p} - \frac{2}{T} g_{\mathbf{x}}^{\mathsf{T}} \sum_{k=1}^{N} \Big(\alpha_k \cos(\omega_k t) + \beta_k \sin(\omega_k t)\Big). \tag{4.14}$$

The boundary conditions are chosen such that $\mathbf{p}(T) = \mathbf{0}$ in order to eliminate the coefficients of $\mathbf{x}_\xi(T)$ in Eq. (4.13). Finally, the terms multiplied with $A_{k,\xi}$ and $B_{k,\xi}$

can be eliminated by defining $\alpha_k(T)$ and $\beta_k(T)$ from

$$\begin{aligned}\alpha_k &= \alpha_k(T) = \frac{\partial J}{\partial A_k}\\ \beta_k &= \beta_k(T) = \frac{\partial J}{\partial B_k}\end{aligned}$$

With $\mathbf{x}(t)$ from the forward solution of the system equations in Eq. (4.3) and the backward solution for $\mathbf{p}(t)$ of the adjoint system in Eq. (4.14), the gradient $\nabla \overline{J}$ is readily given by

$$\nabla \overline{J} = \int_0^T \mathbf{f}_\xi^\mathsf{T} \mathbf{p} \, \mathrm{d}t. \tag{4.15}$$

4.2.3 Application to multibody systems

The approach developed above may also be applied onto multibody systems. For this purpose the system output $y(t) = g(\mathbf{q}, \mathbf{v})$, incorporating the system states $\mathbf{q}$ and $\mathbf{v}$ of the first order system given in Eq. (3.28), is used. The differential equations for the Fourier coefficients can be formulated analogously to Eqs. (4.7) and (4.8) by

$$\dot{a}_k(t) = \frac{2}{T} g(\mathbf{q}, \mathbf{v}) \cos(\omega_k t) \qquad \text{with } a_k(0) = 0 \tag{4.16}$$

$$\dot{b}_k(t) = \frac{2}{T} g(\mathbf{q}, \mathbf{v}) \sin(\omega_k t) \qquad \text{with } b_k(0) = 0. \tag{4.17}$$

Again, the goal is to find a set of parameters ξ such that the cost function in Eq. (4.6) is minimized. The enhancement of the cost function by the system equations in first order form in Eq. (3.28) and by the additional differential equations for the Fourier coefficients in Eqs. (4.16) and (4.17) leads to

$$\begin{aligned}\overline{J} = {} & J(A_k, B_k)\\ & + \int_0^T \left\{ \mathbf{p}^\mathsf{T} (\dot{\mathbf{q}} - \mathbf{v}) + \mathbf{w}^\mathsf{T} \left(\mathbf{M}\dot{\mathbf{v}} - \mathbf{f}(\mathbf{q}, \mathbf{v}, \xi, t) + \mathbf{C}_\mathbf{q}^\mathsf{T} \lambda \right) + \mu^\mathsf{T} \mathbf{C}(\mathbf{q}) \right\} \mathrm{d}t\\ & + \sum_{k=1}^N \int_0^T \alpha_k \left(\frac{2}{T} g(\mathbf{q}, \mathbf{v}) \cos(\omega_k t) - \dot{a}_k(t) \right) \mathrm{d}t\\ & + \sum_{k=1}^N \int_0^T \beta_k \left(\frac{2}{T} g(\mathbf{q}, \mathbf{v}) \sin(\omega_k t) - \dot{b}_k(t) \right) \mathrm{d}t\end{aligned} \tag{4.18}$$

in which $\mathbf{p}$ and $\mathbf{w}$ represent the adjoint variables corresponding to the multibody system, μ pertains to the constraint equation, and α_k and β_k with $k = 1, \dots, N$ are the adjoints corresponding to the Fourier coefficients a_k and b_k, respectively.

At this point, the variables $\mathbf{p}(t)$, $\mathbf{w}(t)$, $\mu(t)$, α_k and β_k may be chosen arbitrarily. The gradient of the function $\overline{J}$ is given by

$$\begin{aligned}\frac{\partial \overline{J}}{\partial \xi} = \left(\nabla \overline{J}\right)^{\mathsf{T}} = &\sum_{k=1}^{N} \frac{\partial J}{\partial A_k} A_{k,\xi} + \sum_{k=1}^{N} \frac{\partial J}{\partial B_k} B_{k,\xi} \\ &+ \int_0^T \Big\{ \mathbf{p}^{\mathsf{T}}(\dot{\mathbf{q}}_\xi - \mathbf{v}_\xi) + \mu^{\mathsf{T}} \mathbf{C}_{\mathbf{q}} \mathbf{q}_\xi + \mathbf{w}^{\mathsf{T}} \Big[\left(\mathbf{M}\dot{\mathbf{v}}\right)_{\mathbf{q}} \mathbf{q}_\xi + \mathbf{M}\dot{\mathbf{v}}_\xi \\ &- \mathbf{f}_{\mathbf{q}} \mathbf{q}_\xi - \mathbf{f}_{\mathbf{v}} \mathbf{v}_\xi - \mathbf{f}_\xi + \left(\mathbf{C}_{\mathbf{q}}^{\mathsf{T}} \lambda\right)_{\mathbf{q}} \mathbf{q}_\xi + \mathbf{C}_{\mathbf{q}}^{\mathsf{T}} \lambda_\xi \Big] \Big\} \, \mathrm{d}t \\ &+ \sum_{k=1}^{N} \int_0^T \left\{ \frac{2}{T} \alpha_k \Big(g_{\mathbf{q}} \cos(\omega_k t) \mathbf{q}_\xi + g_{\mathbf{v}} \cos(\omega_k t) \mathbf{v}_\xi \Big) - \alpha_k \dot{a}_{k,\xi} \right\} \, \mathrm{d}t \\ &+ \sum_{k=1}^{N} \int_0^T \left\{ \frac{2}{T} \beta_k \Big(g_{\mathbf{q}} \sin(\omega_k t) \mathbf{q}_\xi + g_{\mathbf{v}} \sin(\omega_k t) \mathbf{v}_\xi \Big) - \beta_k \dot{b}_{k,\xi} \right\} \, \mathrm{d}t.\end{aligned}$$

Integrating by parts of the terms with $\dot{\mathbf{q}}_\xi$, $\dot{\mathbf{v}}_\xi$, $\dot{a}_{k,\xi}$, $\dot{b}_{k,\xi}$ and assuming that $\mathbf{q}_\xi(0) = \mathbf{0}$, $\mathbf{v}_\xi(0) = \mathbf{0}$, $a_{k,\xi}(0) = 0$ and $b_{k,\xi}(0) = 0$, as a consequence of prescribed initial conditions for $\mathbf{q}$, $\mathbf{v}$, a_k and b_k, yields

$$\begin{aligned}\left(\nabla \overline{J}\right)^{\mathsf{T}} = &\sum_{k=1}^{N} \frac{\partial J}{\partial A_k} A_{k,\xi} + \sum_{k=1}^{N} \frac{\partial J}{\partial B_k} B_{k,\xi} \\ &+ \int_0^T \Big\{ - \dot{\mathbf{p}}^{\mathsf{T}} \mathbf{q}_\xi - \mathbf{p}^{\mathsf{T}} \mathbf{v}_\xi + \mathbf{w}^{\mathsf{T}} \left(\mathbf{M}\dot{\mathbf{v}}\right)_{\mathbf{q}} \mathbf{q}_\xi - \frac{\mathrm{d}}{\mathrm{d}t}(\mathbf{w}^{\mathsf{T}} \mathbf{M}) \mathbf{v}_\xi - \mathbf{w}^{\mathsf{T}} \mathbf{f}_{\mathbf{q}} \mathbf{q}_\xi \\ &- \mathbf{w}^{\mathsf{T}} \mathbf{f}_{\mathbf{v}} \mathbf{v}_\xi - \mathbf{w}^{\mathsf{T}} \mathbf{f}_\xi + \mathbf{w}^{\mathsf{T}} \left(\mathbf{C}_{\mathbf{q}}^{\mathsf{T}} \lambda\right)_{\mathbf{q}} \mathbf{q}_\xi + \mathbf{w}^{\mathsf{T}} \mathbf{C}_{\mathbf{q}}^{\mathsf{T}} \lambda_\xi + \mu^{\mathsf{T}} \mathbf{C}_{\mathbf{q}} \mathbf{q}_\xi \Big\} \, \mathrm{d}t \\ &+ \sum_{k=1}^{N} \int_0^T \left\{ \frac{2}{T} \alpha_k \Big(g_{\mathbf{q}} \cos(\omega_k t) \mathbf{q}_\xi + g_{\mathbf{v}} \cos(\omega_k t) \mathbf{v}_\xi \Big) + \dot{\alpha}_k a_{k,\xi} \right\} \, \mathrm{d}t \\ &+ \sum_{k=1}^{N} \int_0^T \left\{ \frac{2}{T} \beta_k \Big(g_{\mathbf{q}} \sin(\omega_k t) \mathbf{q}_\xi + g_{\mathbf{v}} \sin(\omega_k t) \mathbf{v}_\xi \Big) + \dot{\beta}_k b_{k,\xi} \right\} \, \mathrm{d}t \\ &+ \Big(\mathbf{p}^{\mathsf{T}} \mathbf{q}_\xi + \mathbf{w}^{\mathsf{T}} \mathbf{M} \mathbf{v}_\xi \Big) \Big|_{t=T} - \sum_{k=1}^{N} \Big(\alpha_k(T) A_{k,\xi} + \beta_k(T) B_{k,\xi} \Big).\end{aligned}$$

The computation of the sensitivities $a_{k,\xi}$, $b_{k,\xi}$, $\mathbf{q}_\xi$, $\mathbf{v}_\xi$, and λ_ξ can be circumvented, if the factors multiplied vanish. In case of the adjoints α_k and β_k, constant values

$$\alpha_k = \alpha_k(T) = \frac{\partial J}{\partial A_k} \quad \text{and} \quad \beta_k = \beta_k(T) = \frac{\partial J}{\partial B_k} \tag{4.19}$$

are used to fulfill $\dot{\alpha}_k = \dot{\beta}_k = 0$. The adjoint variables $\mathbf{p}(t)$, $\mathbf{w}(t)$, and $\mu(t)$ have to be chosen such that the adjoint equations

$$\begin{aligned} \dot{\mathbf{p}} &= \mathbf{A}\mathbf{w} + \mathbf{C}_{\mathbf{q}}^{\mathsf{T}}\mu + g_{\mathbf{q}}\mathcal{G}(t) \\ \frac{\mathrm{d}}{\mathrm{d}t}(\mathbf{M}\mathbf{w}) &= -\mathbf{p} - \mathbf{f}_{\mathbf{v}}^{\mathsf{T}}\mathbf{w} + g_{\mathbf{v}}\mathcal{G}(t) \\ \mathbf{C}_{\mathbf{q}}\mathbf{w} &= \mathbf{0} \end{aligned} \tag{4.20}$$

hold. Here the terms

$$\mathbf{A} = (\mathbf{M}\dot{\mathbf{v}})_{\mathbf{q}}^{\mathsf{T}} - \mathbf{f}_{\mathbf{q}}^{\mathsf{T}} + \left(\mathbf{C}_{\mathbf{q}}^{\mathsf{T}}\lambda\right)_{\mathbf{q}}^{\mathsf{T}}$$

and

$$\mathcal{G}(t) = \sum_{k=1}^{N} \frac{2}{T}\Big(\alpha_k \cos(\omega_k t) + \beta_k \sin(\omega_k t)\Big) \tag{4.21}$$

are used. At this point the adjoints can be chosen arbitrarily at $t = T$. For the sake of simplicity, they are set to zero, $\mathbf{p}(T) = \mathbf{w}(T) = \mu(T) = \mathbf{0}$, in order to eliminate the corresponding boundary terms.

It has to be mentioned here that the symmetry of the mass matrix $\mathbf{M} = \mathbf{M}^{\mathsf{T}}$ has been used. If Eqs. (4.20) are satisfied, the gradient $\nabla\overline{J}$ reduces to

$$\nabla\overline{J} = \int_0^T \left(-\mathbf{w}^{\mathsf{T}}\mathbf{f}_{\xi}\right) \mathrm{d}t. \tag{4.22}$$

4.2.4 Summary: The Computation of the Gradient

1. Solve the equations of motion Eq. (3.28) forward in time $t \in [0, T]$ yielding $\mathbf{q}(t)$, $\mathbf{v}(t)$, $\lambda(t)$. This may be done, e.g. by choosing the Hilbert-Hughes-Taylor (HHT) integration scheme, as proposed in [32] and its application for a differential algebraic system given in an index three formulation in [50].
2. By inserting $g(\mathbf{q}(t), \mathbf{v}(t))$ in Eqs. (4.4) and (4.5) the requested Fourier coefficients can be computed. At this point the actual value of the performance measure may be evaluated by utilizing a_k and b_k in Eq. (4.6).
3. Compute the adjoint Fourier coefficients by raising Eq. (4.19).
4. Generate the function $\mathcal{G}(t)$ from Eq. (4.21) by using the resulting adjoint Fourier coefficients.
5. Solve the adjoint system Eq. (4.20) for $\mathbf{p}(\tau)$, $\mathbf{w}(\tau)$ and $\mu(\tau)$, where $\tau = T - t$. A numerical solution at time instances τ_n can be computed by using the backward differentiation scheme presented in Section 3.3.4.4.
6. Compute the adjoint variables as function of the original time by setting $\mathbf{p}(t) = \mathbf{p}(\tau = T - t)$ and $\mathbf{w}(t) = \mathbf{w}(\tau = T - t)$. Moreover, determine $\mathbf{f}_{\xi}$ along the forward simulation.

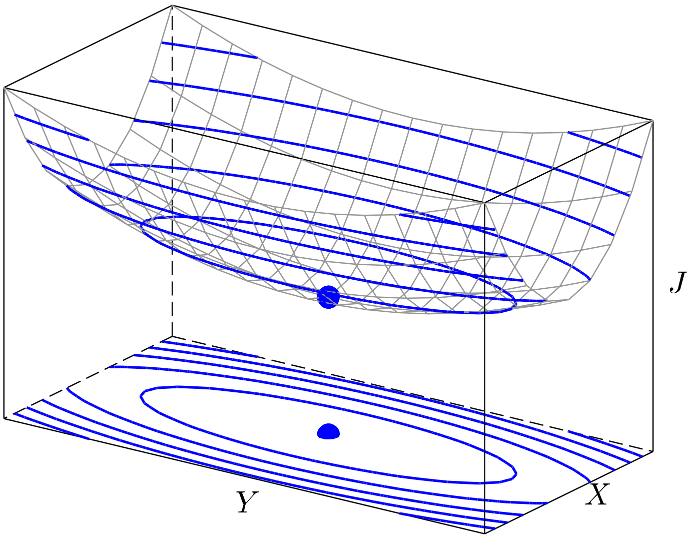


Figure 4.1: Example of an optimization surface – elliptic paraboloid

7. From Eq. (4.22) the gradient of J may finally be derived.

4.3 Iterative Methods for Optimization

Assuming l unknown parameters and therefore l optimization variables, the cost function J can be seen as representation of a surface in the $(l+1)$-dimensional space. Figures 4.1 and 4.2 show two generic surfaces, an elliptic parabolid and the Styblinski-Tang function, which both may be seen as result of a performance measure J depending on the parameters X and Y. Both functions have only one distinct global minimum, which is marked with a blue dot in each case. Moreover, the plots show contours of the surfaces, which allow finding the gradient direction intuitively by forming the normal vector onto a contour. In the following section both functions depicted are raised for explaining different concepts of iterative optimization. The point ξ^* with lowest elevation J corresponds with the optimal set of parameters. The quality of an identification result given by such an approach depends on the level of detail the virtual model uses to emulate the real experiment. Provided that the real behavior is matched well, the parameters found will be equivalent to those of the physical model. Most of the existing iterative approaches do not find the optimal point ξ^* in a finite number of computation steps, but rather

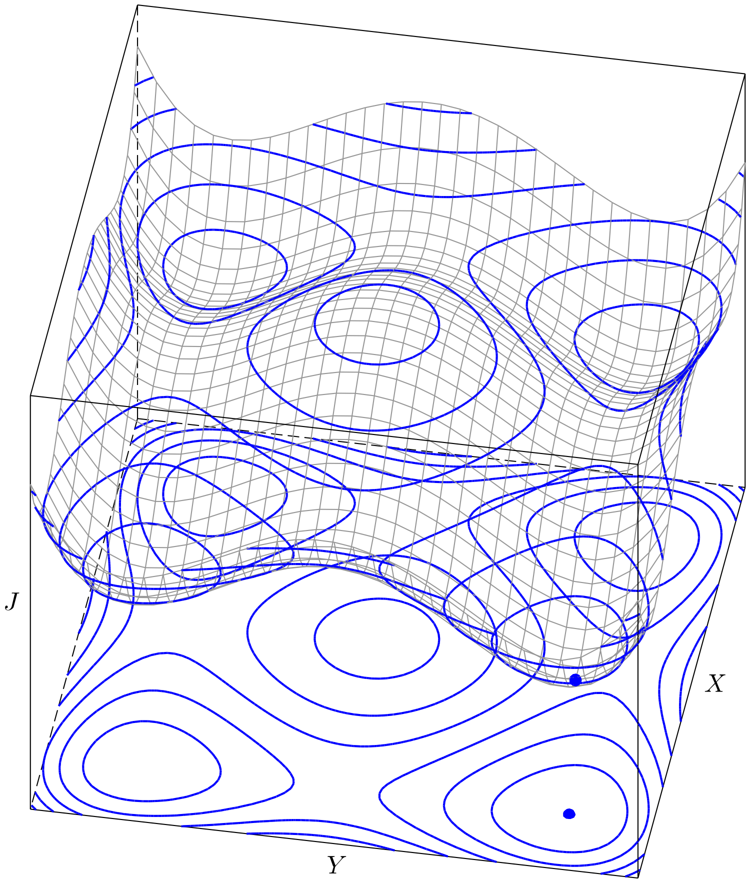


Figure 4.2: Example of an optimization surface – Styblinski-Tang function

generate a progression reducing the cost function $J(\xi)$, i.e.

$$J(\xi_{k+1}) < J(\xi_k), \quad k = 0, 1, 2, \ldots$$

and which should converge to ξ^* for $k \to \infty$, i.e.

$$\lim_{k\to\infty} \xi_k = \xi^*.$$

As the author's background and focus lies in modeling of mechanical systems, optimization methods are understood as mathematical tools and therefore in the following section no proofs are performed. Proofs, further details and additional methods can be found in the literature, e.g. the work of Kelley [37]. Nevertheless, the overview given here allows obtaining an overall picture of the *line search* approach, chosen for the identification of multibody system parameters. Here, in order to find an admissible update $\Delta\xi_k = \xi_{k+1} - \xi_k$ resulting in $J(\xi_{k+1}) < J(\xi_k)$, two different tasks are performed. First, a direction $\mathbf{p}_k$ is computed such that the update is given by $\Delta\xi_k = \mathbf{p}_k\alpha_k$, with $\alpha_k > 0$. Second, a valid step size α_k is determined in order to minimize J along the search direction $\mathbf{p}_k$.

4.3.1 Determination of the search direction

The computation of the search direction is limited by the information available, which is the value of the performance measure to minimize and the gradient thereof. In the following two approaches for determining $\mathbf{p}_k$ are presented.

4.3.1.1 Method of steepest descent

The method of steepest descent employs the fact, that the gradient information can be used for generating a first order approximation of the cost function J in the form

$$J(\xi_{k+1}) = J(\xi_k) + \mathbf{p}_k^{\mathsf{T}} \nabla J(\xi_k) + \mathcal{O}(\mathbf{p}_k^2). \tag{4.23}$$

Using this linearization of J, the maximum possible change is gained using the direction parallel to $\nabla J(\xi_k)$. As the goal is to minimize J the opposite direction has to be chosen and therefore $\mathbf{p}(\xi_k) = -\nabla J(\xi_k)$.

The plot in Fig. 4.3 shows the search direction according to the method of steepest descent for two different starting points. Due to linearization of J the minimum cannot be reached searching along the direction $\mathbf{p}_k$ in general. As shown in the plot this can happen by chance but is not the general case. Taking a step in direction of the steepest descent leads to maximal reduction of the cost function in the vicinity of the starting point. However, beyond that point the gradient direction does not lead to maximal minimization, as the higher order terms $\mathcal{O}(\mathbf{p}_k^2)$ are not considered by the approximation. Practically speaking, this means that a high number of search direction updates have to be performed in case of complex shaped cost functions J. A term arising in this context is the so called effect of *zigzagging.* If the valley, which contains the minimum, is curved and narrow, the optimization is performed in a zigzag manner and therefore many iterations are demanded before ending up in the minimum. The linearization of J given in Eq. (4.23) can be interpreted as a tangential plane fitted onto the surface. In Fig. 4.4 the tangential plane at a given point $(X_0, Y_0)^{\mathsf{T}}$ is fitted onto the Styblinski-Tang function.

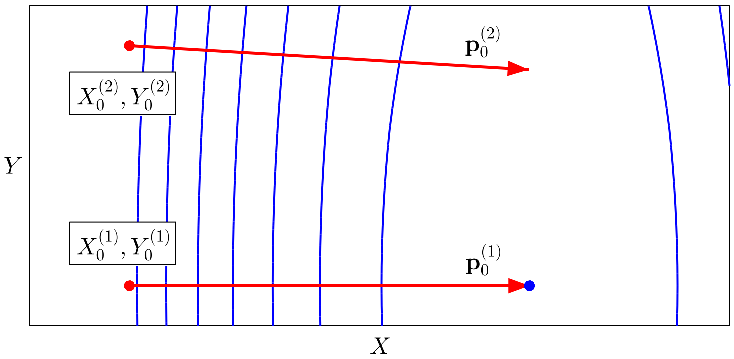


Figure 4.3: Two search directions $\mathbf{p}_0^{(1)}$, $\mathbf{p}_0^{(2)}$ are computed, starting at two different points $(X_0^{(1)}, Y_0^{(1)})^\mathsf{T}$, $(X_0^{(2)}, Y_0^{(2)})^\mathsf{T}$ using the method of steepest descent.

4.3.1.2 Quasi-Newton method

Following the idea of Eq. (4.23) the approximation of J can be extended by a quadratic term

$$J(\xi_{k+1}) = J(\xi_k) + \mathbf{p}_k^\mathsf{T} \nabla J(\xi_k) + \frac{1}{2} \mathbf{p}_k^\mathsf{T} \nabla^2 J(\xi_k) \mathbf{p}_k + \mathcal{O}(\mathbf{p}_k^3), \tag{4.24}$$

where $\nabla^2 J$ denotes the Hessian matrix. The so called *Newton direction* results directly from minimizing the approximating function Eq. (4.24) with respect to $\mathbf{p}_p$ and reads

$$\mathbf{p}_k = -\left(\nabla^2 J(\xi_k)\right)^{-1} \nabla J(\xi_k).$$

Due to the higher order terms given in Eq. (4.24), the actual update of parameters is computed by $\Delta\xi_k = \mathbf{p}_k \alpha_k$ and an appropriate method for computing a feasible step size α_k has to be found.

The advantage of using the curvature information of J by including the Hessian in the update formulation, can easily be seen by means of the example shown in Fig. 4.5. As the elliptical paraboloid function represented by its contour lines is quadratic, the global optimum can be reached independently from the starting point chosen. Figure 4.6 shows the surface of the Styblinski-Tang function and a second order approximation at the given point $(X_0, Y_0)^\mathsf{T}$. Although the approximation shows good accordance with the surface, the minimum thereof does not coincide with the minimum of the actual function J. However, the minimum, marked with a blue sphere, can nearly be hit by tracking the projection of the search direction onto the surface, represented by the green curve. Up to this point there is no possibility to find the Hessian $\nabla^2 J$, neither with invoking system sensitivity analysis nor with the adjoint approach presented in the preceding chapters. Nevertheless it

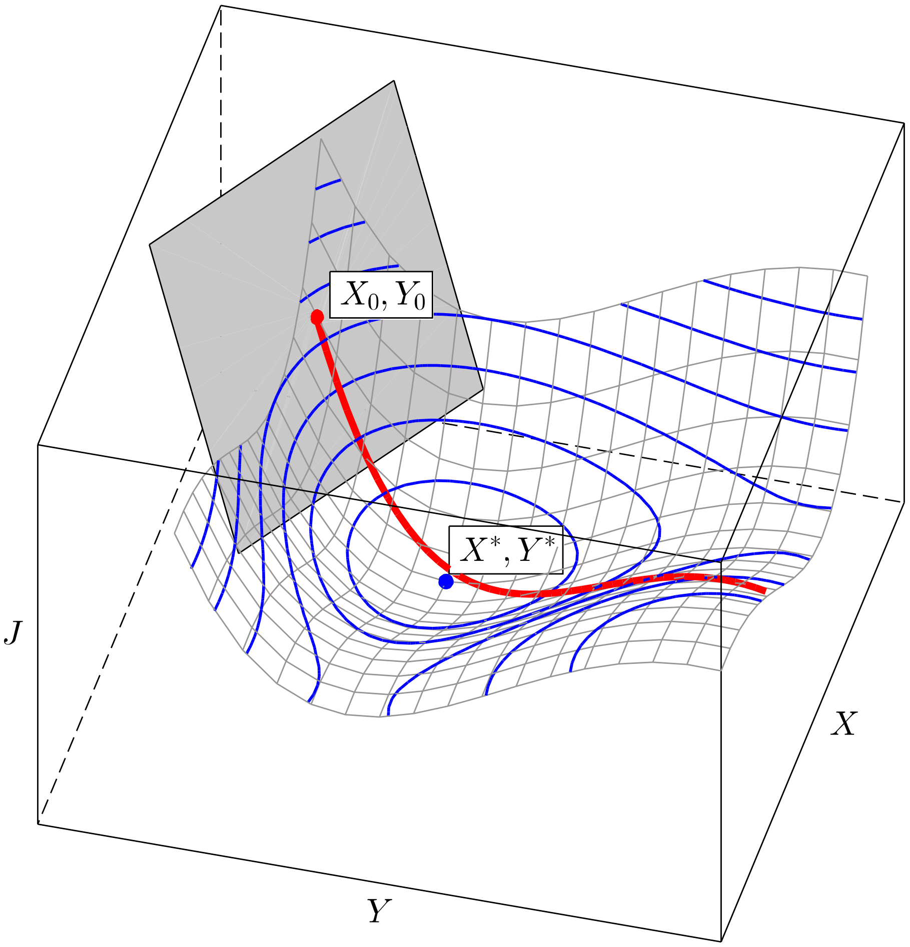


Figure 4.4: First order approximation for Styblinski-Tang function

is possible to compute a search direction by utilizing an algorithm that provides a guess for the Hessian. A very common representative is the so called BFGS algorithm introduced by Broyden [11], Fletcher [21], Goldfarb [25] and Shanno [61]. Using the approximation of the Hessian in combination with the Newton update step is usually denoted as Quasi-Newton method.

Starting with an initial guess for the Hessian $\mathbf{B}_0$, which could be the identity matrix, the algorithm uses consecutive values of the gradient ∇J_{k+1} and ∇J_k, the last update step $\mathbf{s}_k = \xi_{k+1} - \xi_k$ and the previous guess of the Hessian $\mathbf{B}_k$ for computing

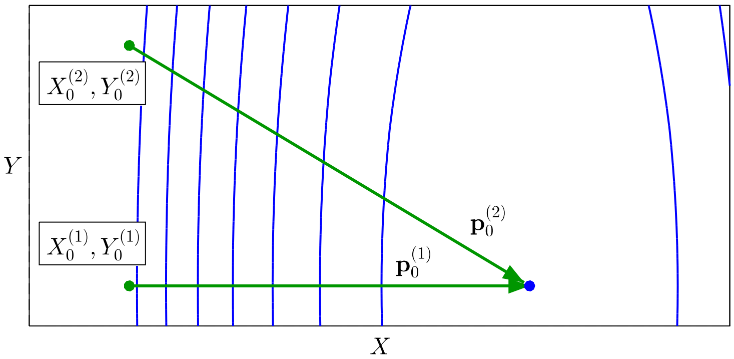


Figure 4.5: Two search directions $\mathbf{p}_0^{(1)}$, $\mathbf{p}_0^{(2)}$ are computed, starting at two different points $(X_0^{(1)}, Y_0^{(1)})^\intercal$, $(X_0^{(2)}, Y_0^{(2)})^\intercal$ using a Newton step.

$\mathbf{B}_{k+1}$. The Hessian therefore is given by

$$\mathbf{B}_{k+1} = \mathbf{B}_k + \frac{\mathbf{r}_k \mathbf{r}_k^\intercal}{\mathbf{r}_k^\intercal \mathbf{s}_k} - \frac{\mathbf{B}_k \mathbf{s}_k \mathbf{s}_k^\intercal \mathbf{B}_k}{\mathbf{s}_k^\intercal \mathbf{B}_k \mathbf{s}_k},$$

where $\mathbf{r}_k$ describes the difference of gradients

$$\mathbf{r}_k = \nabla J(\xi_{k+1}) - \nabla J(\xi_k).$$

For computing the search direction $\mathbf{p}_k$ the inverse of $\mathbf{B}_k$ is required. Instead of developing a guess for the Hessian itself, a modification of the BFGS algorithm intends to find a guess for the inverse $\mathbf{B}_{k+1}^{-1}$. Making use of the formula

$$\mathbf{B}_{k+1}^{-1} = \mathbf{B}_k^{-1} + \frac{\left(\mathbf{s}_k^\intercal \mathbf{r}_k + \mathbf{r}_k^\intercal \mathbf{B}_k^{-1} \mathbf{r}_k\right)\left(\mathbf{s}_k \mathbf{s}_k^\intercal\right)}{\left(\mathbf{s}_k^\intercal \mathbf{r}_k\right)^2} - \frac{\mathbf{B}_k^{-1} \mathbf{r}_k \mathbf{s}_k^\intercal + \mathbf{s}_k \mathbf{r}_k^\intercal \mathbf{B}_k^{-1}}{\mathbf{s}_k^\intercal \mathbf{r}_k}$$

therefore allows to directly compute the demanded inverse and eliminates the costly matrix inversion.

4.3.2 Step size

Once a search direction $\mathbf{p}_k$ is found, the remaining task is to solve the scalar optimization problem

$$J(\xi_k + \alpha_k \mathbf{p}_k) = J(\alpha_k) \rightarrow \min.$$

for α_k. The mapping between the multidimensional and the one-dimensional problem is illustrated in Fig. 4.7, where the search direction computed by the

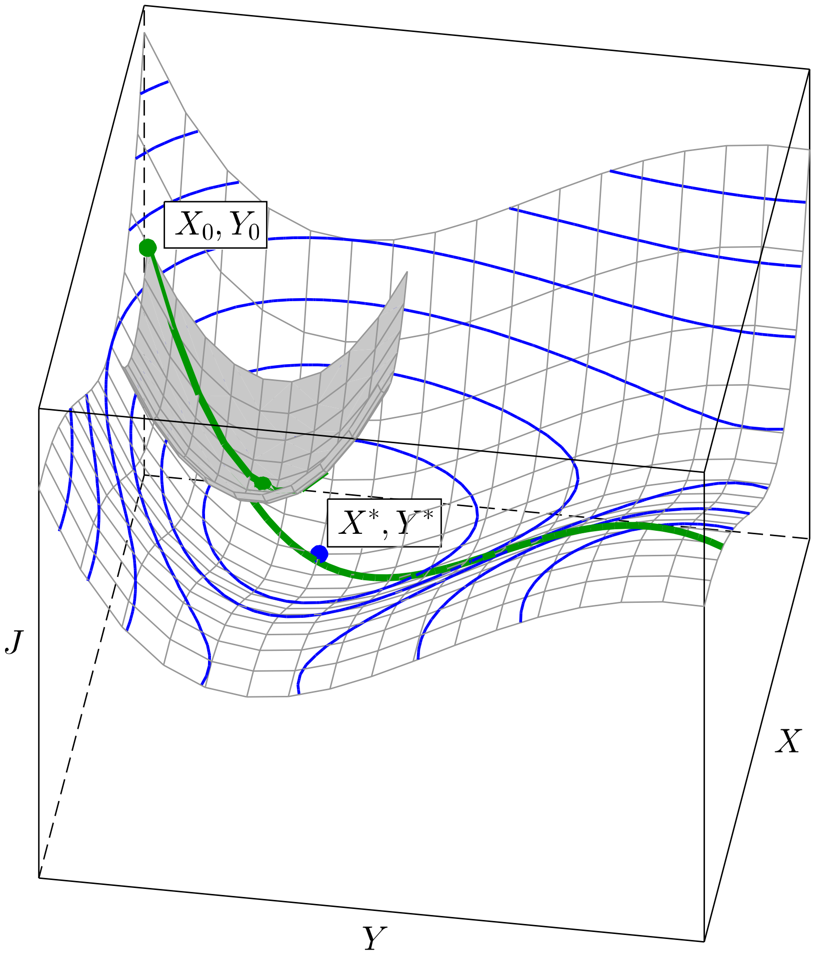


Figure 4.6: Second order approximation for Styblinsky-Tang function

Newton method is not only plotted as projection onto the surface of J, but also projected onto a separate plane. Line search algorithms address this issue by different approaches. In the following two common representatives, the *Golden Section* and the *Quadratic Interpolation*, are introduced.

4.3.2.1 Nested intervals - golden section

The method of nested intervals helps generating a converging progression of nested intervals in order to reach the minimum of $J(\alpha_k)$. After finding an interval $[l_0, r_0]$ for α_k including a minimum of J the method can be started at those boundary points. At iteration step j the interval $[l_j, r_j]$, which still contains the optimal step

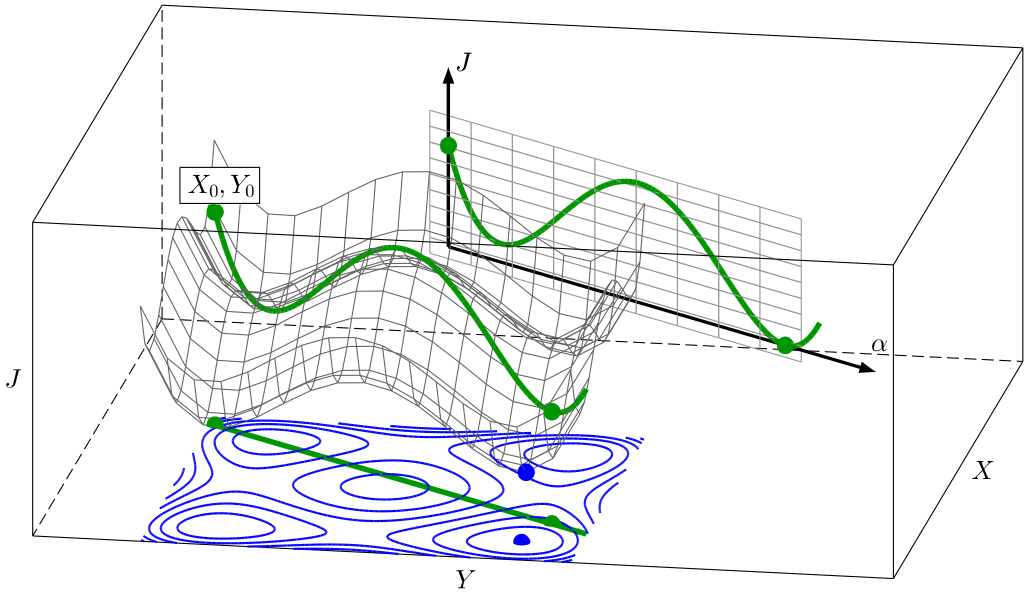


Figure 4.7: Reduction of optimization surface to one-dimensional problem using the example of the Styblinski-Tang function

size α_k^*, is treated. By evaluating

$$\begin{aligned}\hat{l}_j &= l_j + (1-a)(r_j - l_j)\\ \hat{r}_j &= l_j + a(r_j - l_j)\end{aligned} \tag{4.25}$$

two new points satisfying $l_j < \hat{l}_j < \hat{r}_j < r_j$ are generated. The parameter a is demanded to be chosen in the interval $[0.5, 1]$.
Once the iteration is started, two different cases may occur. If $J(\hat{l}_j \leq \hat{r}_j)$ the point r_j is discarded. The interval for the next iteration is then given by $[l_j, \hat{r}_j]$. In case of $J(\hat{l}_j > \hat{r}_j)$ the point l_j is ignored and the next interval is $[\hat{l}_j, r_j]$.
The plot in Fig. 4.8 shows the performance measure $J(\alpha_k)$ of the Styblinski-Tang function along the direction found by the Newton method which is also given in Fig. 4.7. Here two steps of the presented method are displayed. First, in iteration j the point r_j is discarded due to the fact, that $\hat{l}_j < \hat{r}_j$. Different from that in iteration $j+1$ the point l_{j+1} is neglected as $\hat{l}_{j+1} > \hat{r}_{j+1}$. With a special choice of a only one evaluation per iteration has to be performed in order to obtain the

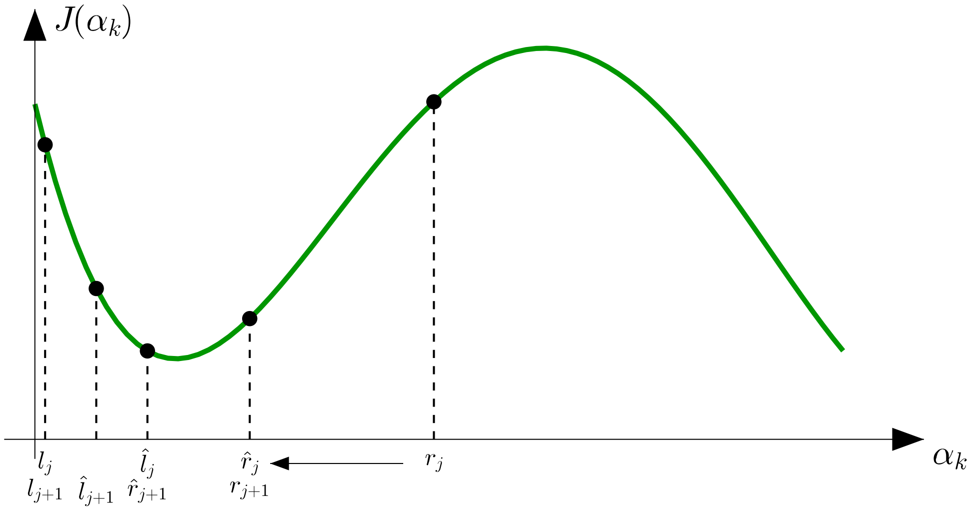


Figure 4.8: Method of nested intervalls - line search along the search direction

intermediate points $\hat{l}_j$ and $\hat{r}_j$. Looking at the example depicted in Fig. 4.8 and employing Eq. (4.25) the relations necessary can be developed. The equations for iteration $j+1$ read

$$\begin{aligned}\hat{l}_{j+1} &= l_{j+1} + (1-a)(r_{j+1} - l_{j+1}) = l_j + (1-a)a(r_j - l_j)\\ \hat{r}_{j+1} &= l_{j+1} + a(r_{j+1} - l_{j+1}) = l_j + a^2(r_j - l_j),\end{aligned}$$

where here the result of iteration j, $l_{j+1} = l_j$ and $r_{j+1} = \hat{r}_j$, is already inserted. Demanding $\hat{r}_{j+1} = \hat{l}_j$ leads to the equation $a^2 + a - 1 = 0$ and therefore to $a = (\sqrt{5}-1)/2$. This directly relates the number a with the so called *golden section* $\Phi = 1 + a = 1/a$.

4.3.2.2 Quadratic interpolation

Another approach for finding the optimal step size is to compute an approximation of the performance measure along the current search direction $\mathbf{p}_k$. Typically a quadratic interpolation polynomial is built up by three points $J(\alpha_1)$, $J(\alpha_2)$ and $J(\alpha_3)$. Another approach generates a quadratic polynomial

$$J(\alpha) \approx \tilde{J}(\alpha) = a\alpha^2 + b\alpha + c \tag{4.26}$$

from the following conditions at two sampling points α_l and α_r:

$$\begin{aligned} J(\alpha_l) &= a\ \alpha_l^2 + b\ \alpha_l + c = J_l \\ \left.\frac{\mathrm{d}J}{\mathrm{d}\alpha}\right|_{\alpha_l} &= 2\ a\ \alpha_l + b = J_l' \\ J(\alpha_r) &= a\ \alpha_r^2 + b\ \alpha_r + c = J_r \end{aligned}$$

where the derivative $\mathrm{d}J/\mathrm{d}\alpha$ is computed by the projection of ∇J onto $\mathbf{p}$:

$$\left.\frac{\mathrm{d}J}{\mathrm{d}\alpha}\right|_{\alpha_k} = \nabla J(\xi_k + \mathbf{p}_k\alpha_k)\mathbf{p}_k.$$

Solving for a, b and c results in

$$\begin{aligned} a &= \frac{J_l - J_r - J_l'(\alpha_l - \alpha_r)}{(\alpha_l - \alpha_r)^2} \\ b &= \frac{2J_l\alpha_l - \alpha_l\left(2J_r + J_l'\alpha_l\right) + J_l'\alpha_r^2}{(\alpha_l - \alpha_r)^2} \\ c &= \frac{J_r\alpha_l^2 + \alpha_r\left(J_l'\alpha_l\left(\alpha_l - \alpha_r\right) + J_l\left(-2\alpha_l + \alpha_r\right)\right)}{(\alpha_l - \alpha_r)^2}. \end{aligned}$$

Hence, the optimal point α^* can be found by demanding

$$\left.\frac{\mathrm{d}J}{\mathrm{d}\alpha}\right|_{\alpha^*} = 0$$

which results in the expression for α^*:

$$\alpha^* = \frac{2J_l\alpha_l - \alpha_l\left(2J_r + J_l'\alpha_l\right) + J_l'\alpha_r^2}{2\left(-J_l + J_r + J_l'\left(\alpha_l - \alpha_r\right)\right)}.$$

The plot in Fig. 4.9 shows the interpolating polynomial constructed by using the points α_l and α_r. Due to the higher order terms contained in $J(\alpha)$ the optimal solution of the interpolation polynomial and the actual evaluation of $J(\alpha_k^*)$ differ from each other. Depending on the actual strategy chosen, different actions may follow. First, α_k^* may be used as optimized step size and the next iteration starts straight away by computing a new search direction $\mathbf{p}_{k+1}$ followed by the next step size determination. Another possibility is to improve the result by increasing the order of the interpolating polynomial. As given in Fig. 4.10 the interpolating polynomial may therefore be extended by the point α_k^* and used for computing the improved point α_k^{**}.

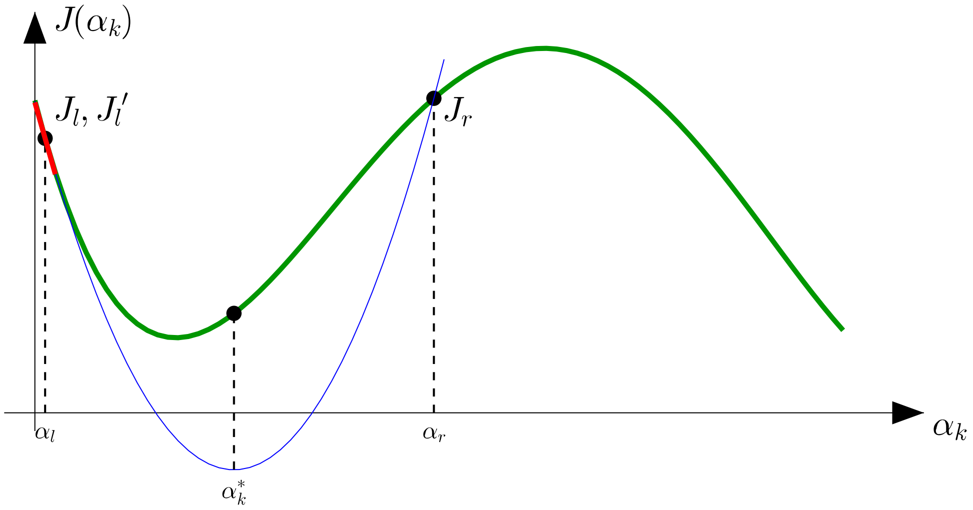


Figure 4.9: Quadratic interpolation method - line search along the search direction

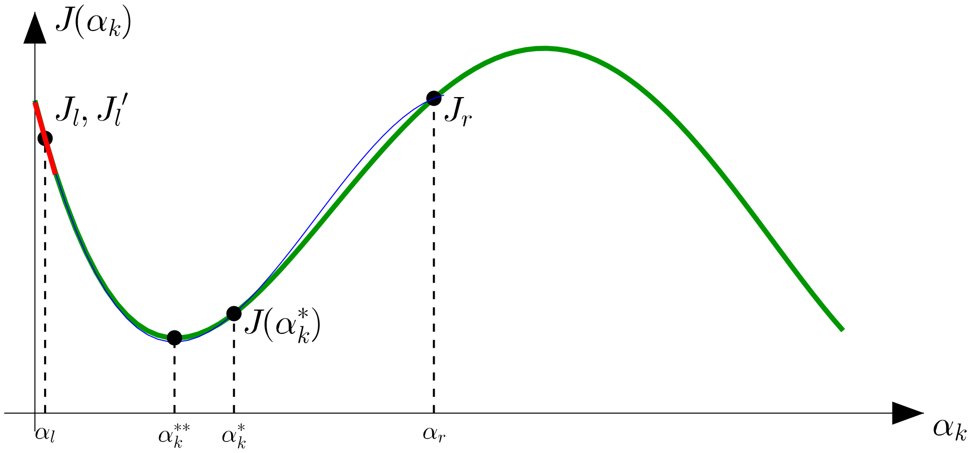
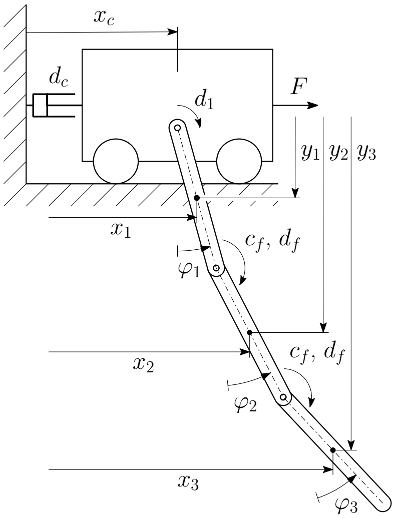


Figure 4.10: Cubic interpolation - line search along the search direction

4.4 Numerical Examples

In this section two examples forming a parameter identification problem for multibody systems are studied. First, the triple pendulum system is investigated and both approaches, in time and frequency domain, are applied. In the second example the crank drive of a four-cylinder combustion engine is treated. Here, the rotational oscillations of the crankshaft are investigated in detail. As this oscillations cannot



(a)

parameter	value	unit
mass of cart	1.0	kg
mass of pendulums	0.5	kg
length of pendulums	1.0	m
d_c	0.01	Ns/m
d_1	0.02	Nms
c_f	10.0	Nm
d_f	0.02	Nms

(b)

Figure 4.11: A system consisting of a cart and three rigid pendula is studied, where the parameters of the flexible pendulum are identified for a given excitation F. (a) Geometric description of the cart–pendulum system. (b) Definition of the parameters used for simulating the system.

be measured directly, they are analyzed by means of order analysis. Hence, only the Fourier coefficient approach is examined. Both examples have been implemented in the inhouse toolbox for inverse dynamics, which was presented at the ECCOMAS 2017 [47].

4.4.1 Triple pendulum

In order to present the performance of the identification method, we study a system of pendula connected to a cart performing a one-dimensional motion. As it can be seen in Fig. 4.11(a), each pendulum is represented by the redundant coordinates (x_i, y_i, φ_i), their interconnection is modelled with rotational springs and therefore this configuration represents a discretization of a rotating flexible beam. In this example we assume the parameters of the flexible beam, the stiffness c_f and damping coefficient d_f, to be unknown. The parameters d_c and d_1 remain untouched during the identification process at a prescribed value.

4.4.1.1 Time domain approach

First, the performance of the virtual system is measured by using the time domain approach introduced in Section 4.1. Therefore a numerical simulation utilizing the parameters listed in the table in Fig. 4.11(b) is performed in order to obtain some kind of virtual measurement. The system output used within the cost function Eq. (4.2) is given by $y = \varphi_1(t)$. In order to include some kind of measurement noise, a zero-mean gaussian noise is added to the simulation result. The plot in Fig. 4.12 shows the according signal of this virtual measurement. Moreover, the solution using the initial parameters $c_f = 8.5$ Nm and $d_f = 0.15$ Nms is depicted. After performing a sufficient number of iterations according to an optimization strategy presented in Section 4.3, the simulation result is given by the plot in Fig. 4.13 using the final set of parameters. As given by the convergence history in Fig. 4.14, the optimization process requires 12 iterations to converge to the

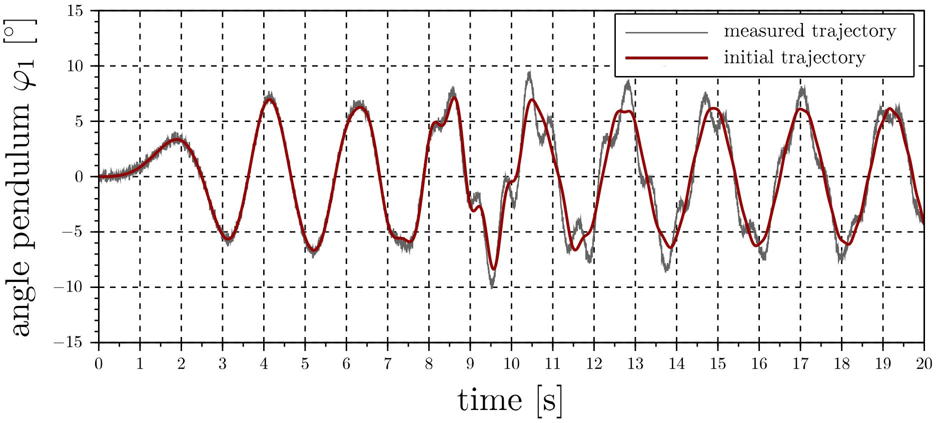


Figure 4.12: Virtual measurement and initial solution for pendulum angle φ_1

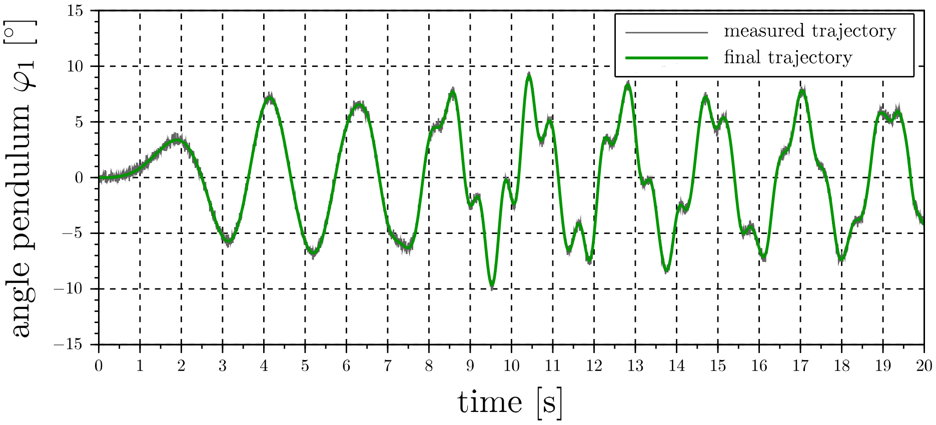


Figure 4.13: Virtual measurement and final solution for pendulum angle φ_1

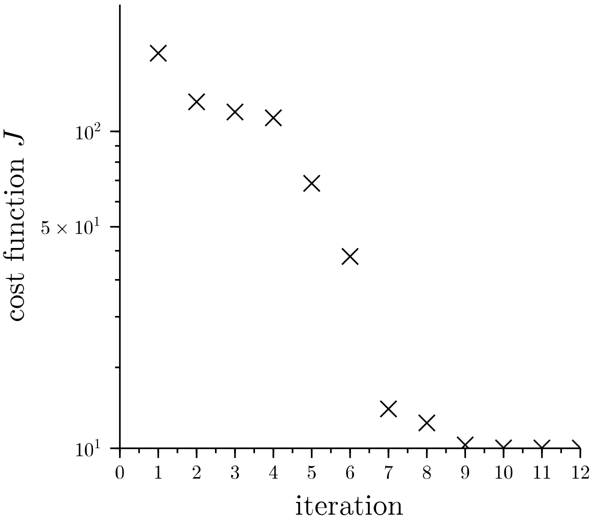


Figure 4.14: Convergence history for optimization of parameters c_f and d_f

minimum at $c_f = 9.9947205$ and $d_f = 0.0200257$. The contour plot in Fig. 4.15 gives an impression of the path taken during the optimization of J. Obviously a Quasi-Newton method was used for computing the update direction, as the search direction does not point into the direction of steepest descent.

4.4.1.2 Frequency domain approach

In order to apply the frequency domain approach using Fourier coefficients, the desired spectrum is computed via Fourier transform of the virtual measurement given in Fig. 4.12. The resulting amplitude spectrum with the given *real* parameters is shown in Fig. 4.16.

Again, the actual parameter identification is initiated with the parameters $c_f =$ 8.5 Nm and $d_f = 0.15$ Nms, and therefore the initial spectrum presented in Fig. 4.17 differs from the desired one. The main advantage of using Fourier coefficients for the identification of system parameters is the possibility to filter data and perform the identification in frequency intervals of interest only. In this special case the parameters of interest c_f and d_f mainly affect the amplitudes occurring around the eigenfrequency of the first bending mode, which is located in the interval [1.4,1.9] Hz. Due to this fact, just the amplitudes in this interval are considered as desired spectrum. When thinking about real applications, this consideration may help to perform a parameter identification even though some subsystems are not

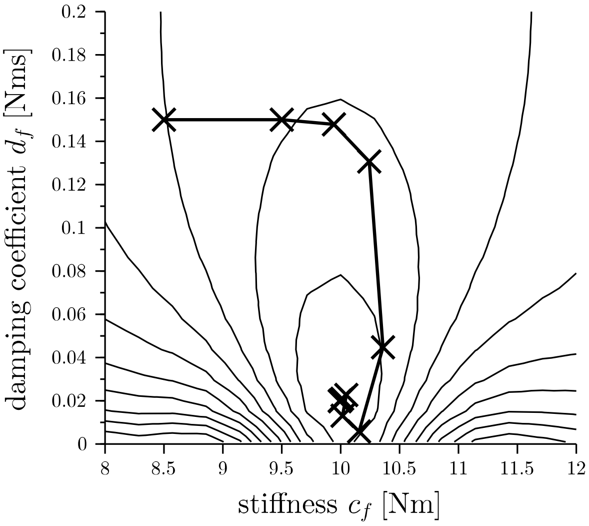


Figure 4.15: Contour plot of cost function J for parameters c_f and d_f using RMS error

known to full extent or to exclude the effects of measurement noise.

By using a quasi-Newton method like the BFGS algorithm for finding a minimum of J and incorporating the computed gradient of Eq. (4.22), the solution can be found within 10 iterations. In Fig. 4.18 the convergence history for the optimization process is shown. The contour plot in Fig. 4.19 gives an impression of the complex shape of $J(c_f, d_f)$ and displays the optimization path taken by the BFGS algorithm. The final parameters gained by using the presented method are $c_f = 9.9999998$ Nm and $d_f = 0.02$ Nms compared to the values used for generating the measure $c_f = 10.0$ Nm and $d_f = 0.02$ Nms.

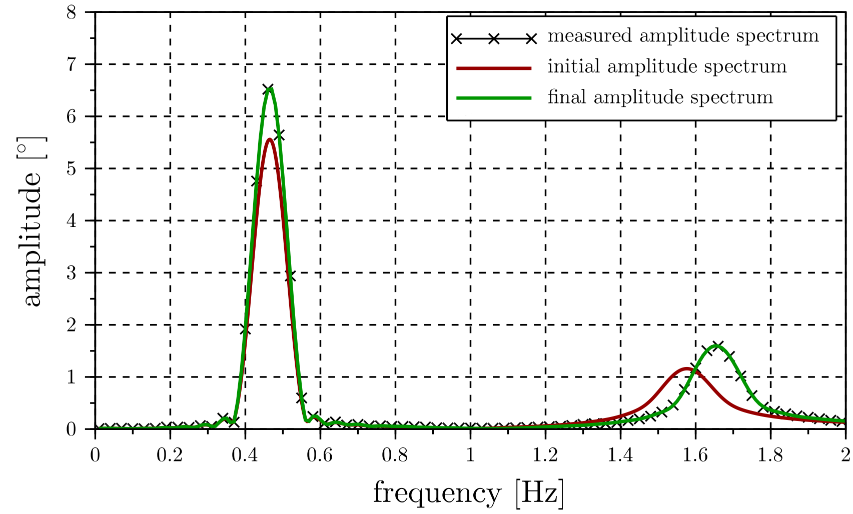


Figure 4.16: Measured, initial and final amplitude spectrum of pendulum angle φ_1

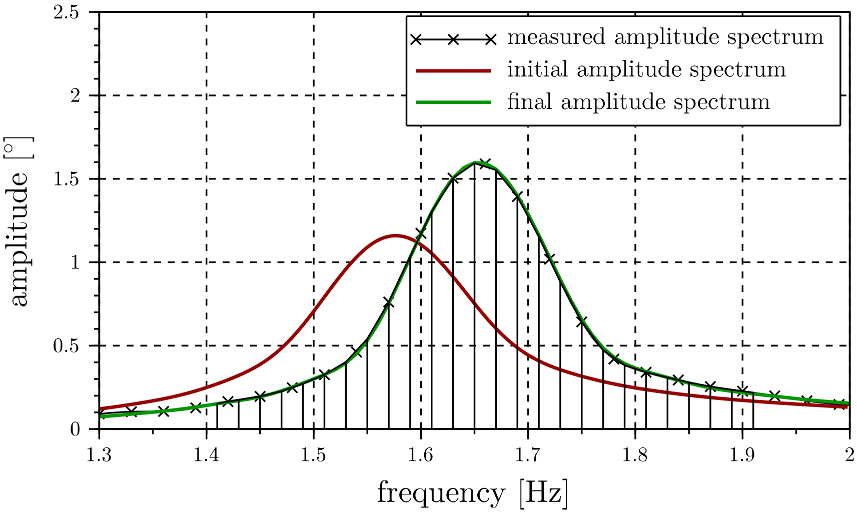


Figure 4.17: Spectrum of pendulum angle φ_1 used for identification (interval highlighted)

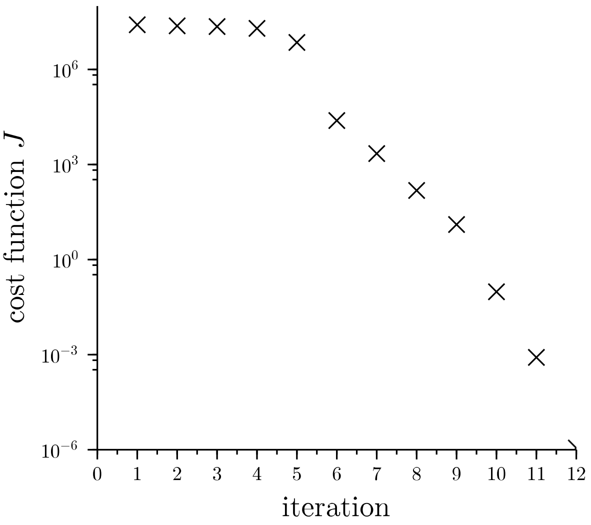


Figure 4.18: Convergence history for optimization of parameters c_f and d_f

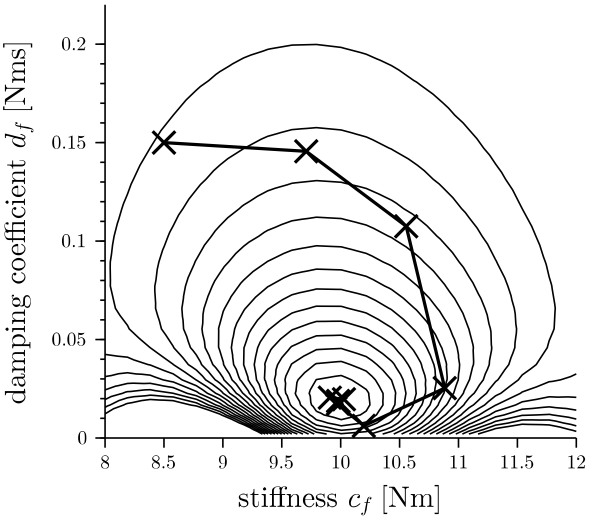


Figure 4.19: Contour plot of cost function J for parameters c_f and d_f

4.4.2 Identification of torsional vibration damper parameters

In this section the presented method is applied to a model of a four–cylinder engine, schematically shown in Fig. 4.20. The goal is to identify the parameters of the engine's torsional vibration damper (TVD), which is described by two Maxwell elements. The TVD is installed in order to reduce torsional oscillations of the crankshaft, which show large amplitudes at the 6th engine order.

4.4.2.1 Model structure

Crankshaft The torsional vibration modes of the crankshaft are crucial for the the parameter identification process. Therefore, six lumped masses resulting in six degrees of freedom $q_1, \ldots, q_6$ (see Fig. 4.21) are introduced in order to model the structural flexibility of the crankshaft. The masses are interconnected with linear springs and linear damping elements $c_1, \ldots, c_6$ and $d_1, \ldots, d_5$ respectively. The inertia parameters are given by the moments of inertia $J_1, \ldots, J_6$.

Conrod For describing the in-plane motion of an engine's connecting rod (conrod) three degrees of freedom are introduced. According to Fig. 4.22 two degrees of freedom are used for the translational motion and one for the rotation about the rotation axis. The mass of each conrod is given by m_{cr} and the moment of inertia by J_{cr}.

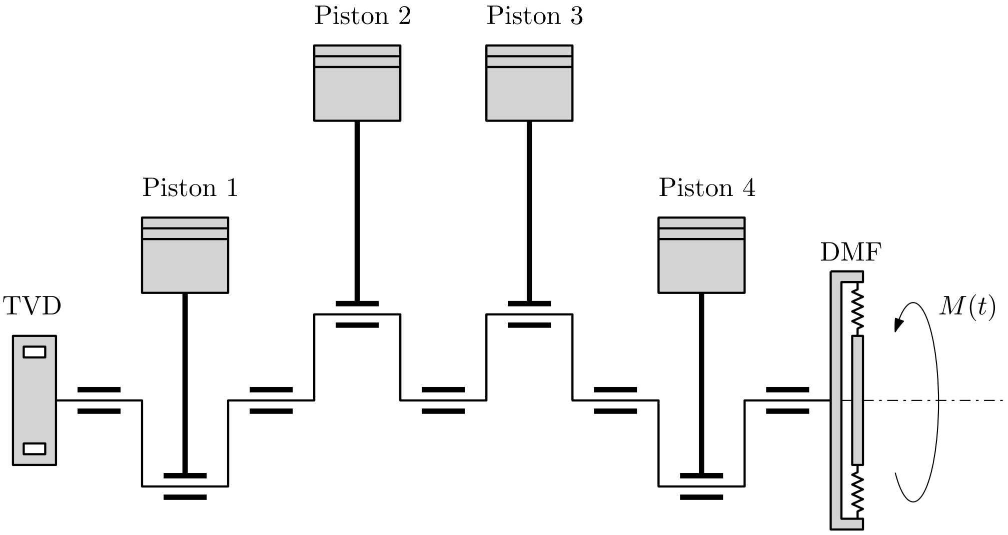


Figure 4.20: Schematics of a four-cylinder engine

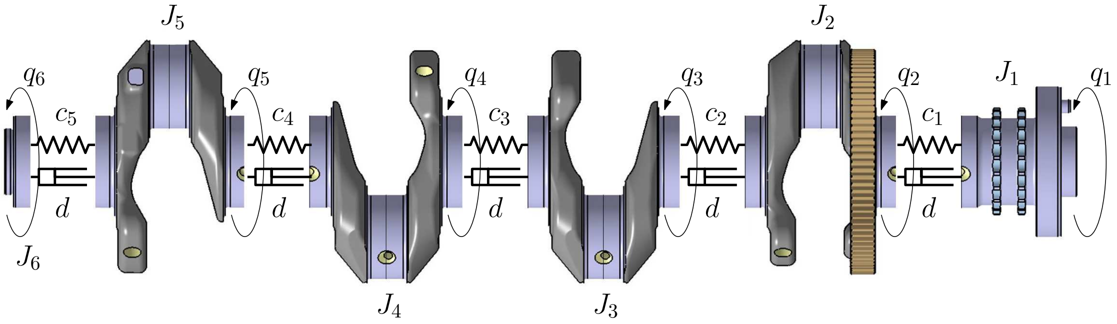


Figure 4.21: Discretization of the crankshaft

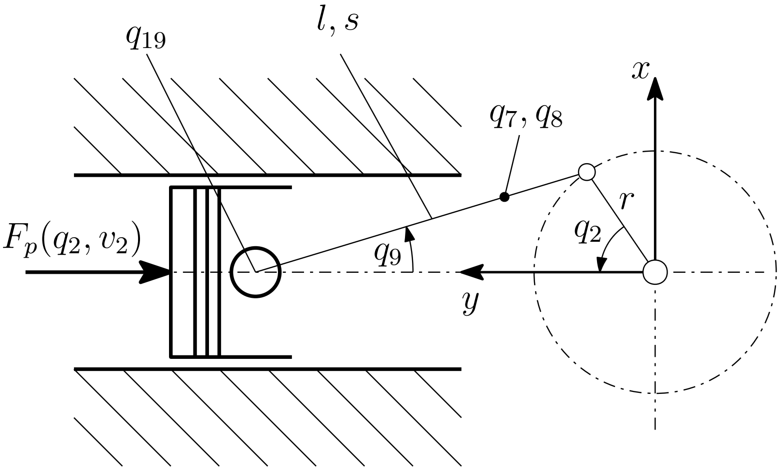


Figure 4.22: Slider–crank mechanism (for cylinder 1)

Piston In the used model each piston features only one translational degree of freedom. The mass of each piston is denoted as m_p and the piston's effective area as A_P.

Dual mass flywheel The primary side of the dual mass flywheel (DMF) is mounted on the right end of the crankshaft (see Fig. 4.21). Hence, its moment of inertia is assigned to q_1. Instead of introducing a degree of freedom for the secondary side of the DMF, the prescribed angle $q_{runup}(t)$ is used. A nonlinear torsional spring and a linear damping element is used for connecting the primary with the secondary side.

Torsional vibration damper The torsional vibration damper (TVD) is installed for reducing the internal torsional vibrations of the crankshaft. Within the housing

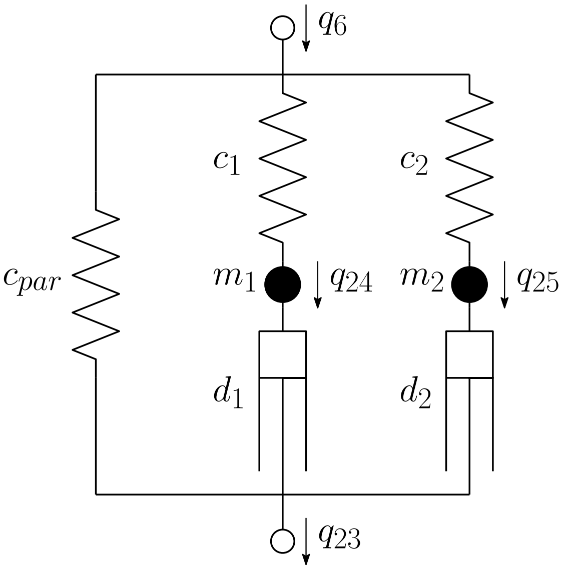


Figure 4.23: Model of the torsional vibration damper

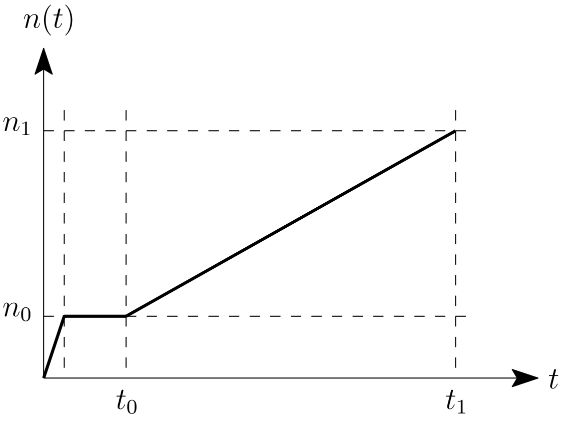


Figure 4.24: Rotational speed of the engine run-up

of the TVD a flywheel ring is gliding in a viscous fluid. Usually, the mathematical model of the TVD is approximated by Maxwell elements. Here, two Maxwell elements and one parallel spring are used. In Fig. 4.23 the schematic description of the model and the degrees of freedom used are presented. The accuracy of the parameters c_1^*, c_2^*, d_1^*, d_2^* and c_{par}^* supplied by the manufacturer may not be satisfactory, and therefore the values of the four parameters c_1^*, c_2^*, d_1^* and d_2^* are adjusted by using parameter identification in the frequency domain. The masses m_1^* and m_2^* are set to zero.

Pulley wheel The pulley wheel used for driving additional aggregates introduces another degree of freedom (q_{26}), which is connected to the TVD hub using a linear spring/damper with parameters c_{PW} and d_{PW}.

Cylinder pressures The cylinder pressure is given by a two-dimensional map depending on the rotational speed and the crankshaft angle. The pressure is applied on each piston in accordance with the firing order.

Run-up of the engine In order to simulate the run-up performed on the real test bench, the rotational speed of the secondary side of the DMF is increased up to the final rotational speed. The ramp used for the simulation is given in Fig. 4.24. The run-up to the final rotational speed itself is done within the time interval $[t_0, t_1]$.

4.4.2.2 Model equations

In the following the equations of motion for the given four-cylinder engine are derived. The vector of redundant degrees of freedom $\mathbf{q}$ are given by

$$\mathbf{q} = [\underbrace{q_1, q_2, ..., q_6}_{\text{crankshaft}}, \underbrace{q_7, q_8, ..., q_{18}}_{\text{conrod}}, \underbrace{q_{19}, q_{20}, ..., q_{22}}_{\text{piston}}, \underbrace{q_{23}, q_{24}, q_{25}}_{\text{TVD}}, \underbrace{q_{26}}_{\text{pulley wheel}}]^{\mathsf{T}}.$$

Hence, the resulting global mass matrix is constant and reads

$$\begin{aligned}\mathbf{M} = \operatorname{diag}(&m_1, m_2, m_3, m_4, m_5, m_6, m_{cr}, m_{cr}, J_{cr},\\ &m_{cr}, m_{cr}, J_{cr}, m_{cr}, m_{cr}, J_{cr}, m_{cr}, m_{cr}, J_{cr}\\ &m_p, m_p, m_p, m_p, J_{\text{TVD}}, 0, 0, J_{\text{PW}}).\end{aligned}$$

The corresponding force vector is

$$
\mathbf{f} = \begin{pmatrix}
-F_{\mathrm{DMF}}(\mathbf{q}, q_0(t)) + d_{\mathrm{DMF}}(\dot{q}_0(t) - v_1) + c_1(q_2 - q_1 - 2\pi) + d(v_2 - v_1) \\
c_1(q_1 - q_2 + 2\pi) + d(v_1 - v_2) + c_2(q_3 - q_2 + \pi) + d(v_3 - v_2) \\
c_2(q_2 - q_3 + \pi) + d(v_2 - v_3) + c_3(q_4 - q_3 - 2\pi) + d(v_4 - v_3) \\
c_3(q_3 - q_4 - 2\pi) + d(v_3 - v_4) + c_4(q_5 - q_4 + 3\pi) + d(v_5 - v_4) \\
c_4(q_4 - q_5 + 3\pi) + d(v_4 - v_5) + c_5(q_6 - q_5) + d(v_6 - v_5) \\
F_6(\mathbf{q}, \dot{\mathbf{q}}) \\
0 \\
0 \\
0 \\
0 \\
0 \\
0 \\
0 \\
0 \\
0 \\
0 \\
0 \\
0 \\
-p_1(\mathbf{q}, \dot{\mathbf{q}})A_P \\
-p_2(\mathbf{q}, \dot{\mathbf{q}})A_P \\
-p_3(\mathbf{q}, \dot{\mathbf{q}})A_P \\
-p_4(\mathbf{q}, \dot{\mathbf{q}})A_P \\
c^*_{\mathrm{par}}(q_6 - q_{23}) + c^*_1(q_{24} - q_{23}) + c^*_2(q_{25} - q_{23}) \\
c^*_1(q_{23} - q_{24}) + d^*_1(v_{24} - v_6) \\
c^*_2(q_{23} - q_{25}) + d^*_2(v_{25} - v_6) \\
c_{\mathrm{PW}}(q_6 - q_{26}) + d_{\mathrm{PW}}(v_6 - v_{26})
\end{pmatrix},
$$

where the 6th entry is

$$
\begin{aligned}
F_6(\mathbf{q}, \dot{\mathbf{q}}) = {} & c_5(q_5 - q_6) + d(v_5 - v_6) + c^*_{\mathrm{par}}(q_{23} - q_6) \\
& + d^*_1(v_6 - v_{24}) + d^*_2(v_6 - v_{25}) \\
& + c_{\mathrm{PW}}(q_{26} - q_6) + d_{\mathrm{PW}}(v_{26} - v_6).
\end{aligned}
$$

The first term in the force vector $F_{\mathrm{DMF}}(\mathbf{q}, q_0(t))$ is given by a nonlinear stiffness characteristic. The crankshaft, conrods, and the pistons are described with redundant coordinates. Hence, four algebraic constraint equations are introduced for each cylinder, where r_{cs} is the radius of the crankshaft, l_{cr} is the length of the conrod, and s_{cr} is the distance to the center of mass. Finally, the vector of constraint equations $\mathbf{C}(\mathbf{q})$ is given by

$$\mathbf{C}(\mathbf{q}) = \begin{pmatrix} r_{cs}\sin(q_2) - l_{cr}\sin(q_9) \\ q_7 + r_{cs}\sin(q_2) - s_{cr}\sin(q_9) \\ q_8 - r_{cs}\cos(q_2) - s_{cr}\cos(q_9) \\ q_{19} - r_{cs}\cos(q_2) - l_{cr}\cos(q_9) \\ r_{cs}\sin(q_3) - l_{cr}\sin(q_{12}) \\ q_{10} + r_{cs}\sin(q_3) - s_{cr}\sin(q_{12}) \\ q_{11} - r_{cs}\cos(q_3) - s_{cr}\cos(q_{12}) \\ q_{20} - r_{cs}\cos(q_3) - l_{cr}\cos(q_{12}) \\ r_{cs}\sin(q_4) - l_{cr}\sin(q_{15}) \\ q_{13} + r_{cs}\sin(q_4) - s_{cr}\sin(q_{15}) \\ q_{14} - r_{cs}\cos(q_4) - s_{cr}\cos(q_{15}) \\ q_{21} - r_{cs}\cos(q_4) - l_{cr}\cos(q_{15}) \\ r_{cs}\sin(q_5) - l_{cr}\sin(q_{18}) \\ q_{16} + r_{cs}\sin(q_5) - s_{cr}\sin(q_{18}) \\ q_{17} - r_{cs}\cos(q_5) - s_{cr}\cos(q_{18}) \\ q_{22} - r_{cs}\cos(q_5) - l_{cr}\cos(q_{18}) \end{pmatrix} = \mathbf{0}.$$

From Fig. 4.22 the constraint equations can be assembled. The first constraint equation describes the x-component of the big end bearing. The second and third constraint equation links the center of mass of the conrod with the redundant generalized coordinates while the fourth constraint equation describes the piston's center of mass.

4.4.2.3 Results of the parameter identification

In this section the actual identification of the four parameters c_1^*, c_2^*, d_1^* and d_2^* of the torsional vibration damper is presented. The main purpose of the TVD is to reduce torsional vibrations of the flexible crankshaft caused by the periodic and dynamic loads (e.g. cylinder pressures). Hence, the twist angle of the crankshaft $y(t) = q_6(t) - q_1(t)$ is chosen to be transformed into the frequency domain. As the measurements performed on a test bench commonly result in spectra for different engine orders, they are used in this investigation, too. Basically, an engine order relates the Fourier coefficients with the rotating frequencies of the engine's crankshaft. In case of the four cylinder engine, the amplitude of the 6th order is dominated by the parameter of the torsional vibration damper. In contrast to Eqs. (4.7), (4.8), the differential equations for computing the Fourier coefficients are

$$\dot{a}_k = \frac{2}{T}\eta(t)y(t)\cos(\omega_k t)$$
$$\dot{b}_k = \frac{2}{T}\eta(t)y(t)\sin(\omega_k t)$$

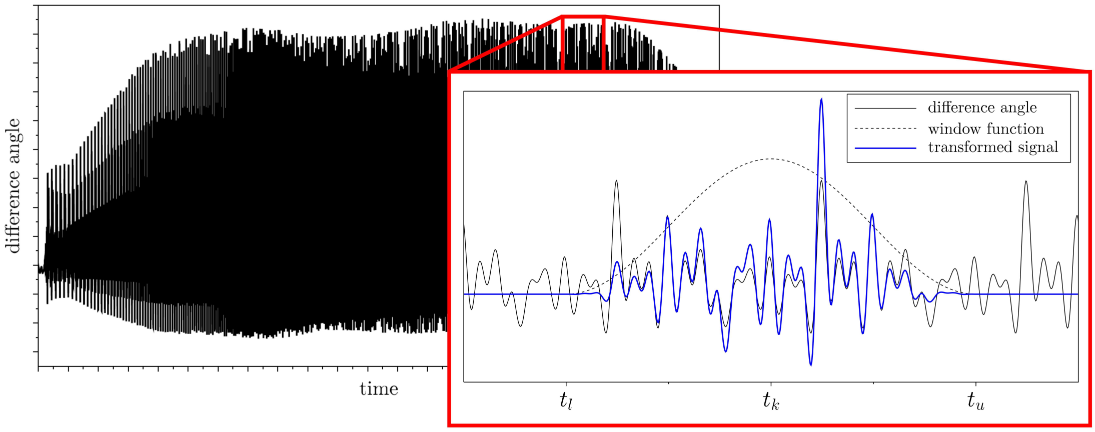


Figure 4.25: Window function and twist angle

where $\eta(t)$ represents a window function and ω_k the kth frequency of interest. It has been shown that the Hanning window given by

$$\eta(t) = \frac{g_\mathrm{c}}{2}\left[1 - \cos\left(\frac{\pi}{t_\mathrm{u} - t_\mathrm{l}}(t - t_\mathrm{l})\right)\right] \tag{4.27}$$

is a good choice for filtering the system output. The upper time limit t_u and the lower time limit t_l determine the borders of the window function. Note, that an amplitude correction factor $g_\mathrm{c} = 2$ is required (for further details see Section 4.2.1). In Fig. 4.25 the system output $y(t)$ is shown for the entire time interval. Moreover, a small interval $t \in [t_l, t_u]$ of $y(t)$ is depicted in detail. The black line shows the original system output $y(t)$, while the dashed line shows the Hanning window function $\eta(t)$. The blue line is the multiplication of $y(t)$ with $\eta(t)$, which is used for the Fourier transformation. Due to the slowly increasing ramp shown in Fig. 4.24 assuming a steady state with constant angular velocity is valid. For the rotational speed n the time interval $[t_l, t_u]$ is given by

$$t_l = t_k - \frac{2}{n} \quad \text{and} \quad t_u = t_k + \frac{2}{n}$$

with

$$t_k = \frac{n_0\, t_1 - n_1\, t_0 + n\,(t_0 - t_1)}{n_0 - n_1}.$$

Here two periods of the base frequency, which is $2/n$ for a four stroke engine, are contained in the Hanning window (for further details see Section 4.2.1).
In Fig. 4.26 the dashed line shows the vibration angle corresponding to the 6th engine order using initial parameters of the TVD. The dotted–dashed line (green) represents the measured vibration angle of the 6th engine order on a test bench.

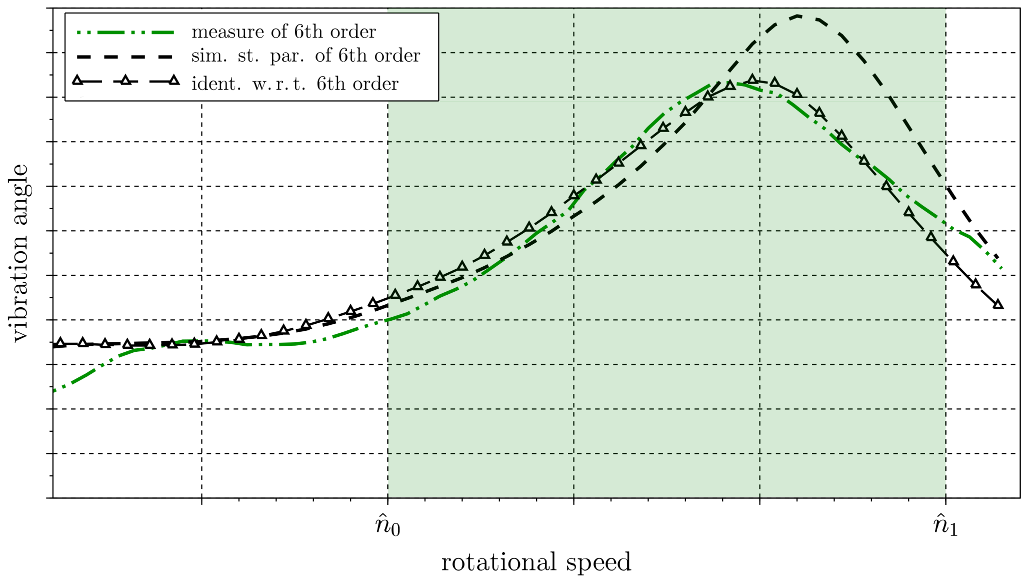


Figure 4.26: Vibration angle of the 6th engine order

The black line (with the triangle symbol) shows the vibration angle of the 6th engine order with the identified TVD parameters. In the interval $n \in [\hat{n}_0, \hat{n}_1]$, highlighted in Fig. 4.26, the deviation between the measured amplitude and the simulated amplitude is included in the cost function of Eq. (4.6). Hence, a significant improvement can be seen in this range compared to the simulation utilizing the initial parameters of the torsional vibration damper. In Fig. 4.27 the engine orders 2, 4, and 8 are depicted, too. On the one hand, the engine orders of the simulation with the initial parameters and, on the other hand, the engine orders of a simulation with the identified parameters of the TVD are shown. As expected the spectra of the orders other than the 6th are only slightly affected.

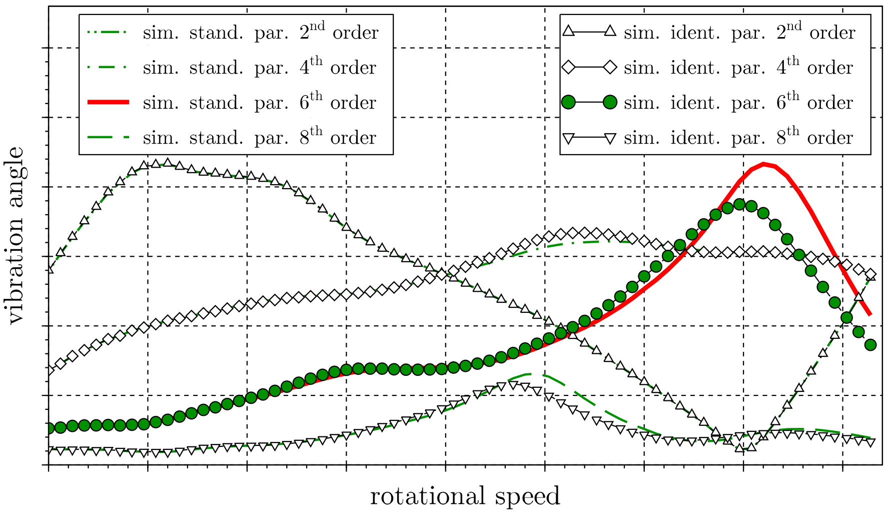


Figure 4.27: Comparison of the engine orders

CHAPTER 5

Design of Optimal Inputs for Multibody Systems

The problem of optimal input design plays a key role when considering an experiment in order to perform a parameter identification. Poorly planned experiments can cause a waste of time and resources and yield little useful information. The linkage between the experiment and modeling world is called *design of experiment* (DOE). If the model knowledge is used for designing the experiments often the term *model-based DOE* can be found in the literature [22].

One of the first authors dealing with the topic of designing experiments was R. A. Fisher in his substantial work *The Design of Experiments* [20]. Although the importance and applicability onto the problem of optimal input design was known only to some extent, many more recent papers refer to his work. Fisher stated that the basic problem of DOE is to decide which pattern of factor combination will best reveal the properties of the response and how this response is influenced by these factors. The term *optimal input design* emerges from the work of Mehra [42, 43] who worked on linear discrete-time systems. There, the most important requirement for designing an input was to generate a system output allowing to determine system parameters featuring a minimum of variance.

In [45], Morelli develops a method for generating optimal input signals utilizing basic statistics including the theory of maximum likelihood estimates for parameters. On this basis Morelli shows the practicability of the method in [46] where system inputs for flight tests with a F-18 High Alpha Research Vehicle are determined.

Recent publications include different fields of application, e.g. Jauberthie et al. [34, 35] have considered a model of an aerodynamic problem. Therein the ideas of Morelli [45] are used to generate an optimal input for identifying aerodynamic parameters of an aircraft. An extensive work on optimal input design in the field of chemistry but also on DOE in general was done by Franceschini [22]. According to this work *model-based DOE* is characterized by:

- The explicit use of the model equations (including any constraint) and current parameters to predict the "information content" of the next experiment (through the evaluation of some suitable objective function), and
- the application of an optimization framework to find the numerical solution of the resulting problem.

Another application dealing with optimal input design is that of process control. Chianeh [16] investigates models of tank systems fed by a pump. The goal is to determine the flow exponent used in Bernoulli's law. Keesman [36] considers optimal input design for choosing between model structures or *model discrimination*. A further topic related to optimal input design is that of optimal sensor placement. In the work by Castro-Triguero [15], several methodologies for computing a minimal set of sensor locations are investigated in order to get the required information for health monitoring of bridge structures.
The latter mentioned approaches can not be applied directly to the model of mechanical systems. Therefore, within this chapter it is shown how the process of optimal input design can systematically be applied for mechanical systems. As the adjoint method provides outstanding performance in the field of optimal control this method is used for computing the update direction during the optimal input iteration process. First a proper performance measure or cost functional for determining optimal inputs is defined via statistical context. For further analysis, the system of sensitivity differential equations is derived and also the adjoint system of the original system is extended by these new terms.

5.1 Optimal Control Approach for Input Optimization

For simplicity, the model equations investigated are first order ordinary differential equations (ODE), and therefore a set of minimal coordinates is used. Nevertheless, it is possible to formulate the entire process for more general differential algebraic equations, and therefore for models with redundant coordinates. The model equations can be written as

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}, \mathbf{u}, \xi, t) \quad \mathbf{x}(0) = \mathbf{x}_0 \\ \mathbf{y} &= \mathbf{y}(\mathbf{x}), \end{aligned} \tag{5.1}$$

where $\mathbf{x}(t) \in \mathbb{R}^n$ is the vector of state variables, $\mathbf{u}(t) \in \mathbb{R}^{N_u}$ the vector of model inputs and $\xi \in \mathbb{R}^l$ the vector of model parameters. In order to compare the simulation result with measurements, a vector of model outputs $\mathbf{y}(\mathbf{x})$ is defined such that it matches with measured outputs $\bar{\mathbf{y}}(t) \in \mathbb{R}^m$.

5.1.1 System sensitivity analysis

Analyzing a system's reaction to small changes in the system parameters at a special timepoint t_i results in the sensitivity matrix

$$\mathbf{S}(t_i) = \begin{bmatrix}\mathbf{y}_{\xi_1}(t_i) & \mathbf{y}_{\xi_2}(t_i) & \dots & \mathbf{y}_{\xi_l}(t_i)\end{bmatrix}. \tag{5.2}$$

As given in Section 3.1, this analysis can be performed by forming the derivatives of the output vector $\mathbf{y}$ with respect to the system parameters ξ. Therefore, Eq. (5.1) is differentiated with respect to each parameter. In the following, the abbreviations $\mathbf{x}_{\xi_j}$, $\mathbf{y}_{\mathbf{x}}$, $\mathbf{f}_{\xi_j}$ and $\mathbf{f}_{\mathbf{x}}$ are used instead of $\frac{\partial \mathbf{x}}{\partial \xi_j}$, $\frac{\partial \mathbf{y}}{\partial \mathbf{x}}$, $\frac{\partial \mathbf{f}}{\partial \xi_j}$ and $\frac{\partial \mathbf{f}}{\partial \mathbf{x}}$ respectively. Hence, the sensitivity equations read

$$\begin{aligned}\dot{\mathbf{x}}_{\xi_j} &= \mathbf{f}_{\mathbf{x}}\mathbf{x}_{\xi_j} + \mathbf{f}_{\xi_j}\\ \mathbf{y}_{\xi_j} &= \mathbf{y}_{\mathbf{x}}\mathbf{x}_{\xi_j},\end{aligned} \tag{5.3}$$

where $\mathbf{y}_{\xi_j}(t_i)$ equals the jth column in the output sensitivity matrix $\mathbf{S}(t_i)$. In case of more than one unknown parameter, the system of differential equations for the state variables and for the sensitivities can be written by

$$\begin{bmatrix}\dot{\mathbf{x}}\\ \dot{\mathbf{x}}_{\xi_1}\\ \dot{\mathbf{x}}_{\xi_2}\\ \vdots\\ \dot{\mathbf{x}}_{\xi_n}\end{bmatrix} = \dot{\mathbf{z}} = \begin{bmatrix}\mathbf{f}\\ \mathbf{f}_{\mathbf{x}}\mathbf{x}_{\xi_1} + \mathbf{f}_{\xi_1}\\ \mathbf{f}_{\mathbf{x}}\mathbf{x}_{\xi_2} + \mathbf{f}_{\xi_2}\\ \vdots\\ \mathbf{f}_{\mathbf{x}}\mathbf{x}_{\xi_n} + \mathbf{f}_{\xi_n}\end{bmatrix} = \tilde{\mathbf{f}}. \tag{5.4}$$

The Jacobian $\tilde{\mathbf{f}}_{\mathbf{z}}$ of the extended system, which is required for further computations, reads

$$\tilde{\mathbf{f}}_{\mathbf{z}} = \begin{bmatrix}\mathbf{f}_{\mathbf{x}} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0}\\ (\mathbf{f}_{\mathbf{x}}\mathbf{x}_{\xi_1} + \mathbf{f}_{\xi_1})_{\mathbf{x}} & \mathbf{f}_{\mathbf{x}} & \mathbf{0} & \dots & \mathbf{0}\\ (\mathbf{f}_{\mathbf{x}}\mathbf{x}_{\xi_2} + \mathbf{f}_{\xi_2})_{\mathbf{x}} & \mathbf{0} & \mathbf{f}_{\mathbf{x}} & \dots & \mathbf{0}\\ \vdots & & & & \\ (\mathbf{f}_{\mathbf{x}}\mathbf{x}_{\xi_n} + \mathbf{f}_{\xi_n})_{\mathbf{x}} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{f}_{\mathbf{x}}\end{bmatrix}. \tag{5.5}$$

5.1.2 Maximization of the information content in experimental measurement data

When dealing with optimal input design, first of all the term optimality has to be clarified. As Morelli defined in [46] optimal inputs minimize the parameter standard errors during model parameter estimation with a maximum likelihood estimator. In other words, the information contained in experimental measurement data has to be maximized. Hence, a proper measure - or cost functional - can be constructed by using a norm of the Fisher information matrix $\mathcal{M}$, see [45].

When estimating model parameter values from measured data, the minimum achievable parameter standard errors using an asymptotically unbiased and efficient estimator (such as maximum likelihood) are called the Cramer-Rao lower bounds [45]. These quantities are a function of the excitation of the system and the noise levels, and collectively measure the information content in the data. For a fixed instrumentation system, the Cramer-Rao lower bounds are influenced by the excitation of the system, which is determined by the input. The input implicitly includes the length of the maneuver.

The Cramer-Rao lower bounds for the parameter standard errors are given by the square root of the diagonal elements of the dispersion matrix $\mathcal{D}$. The dispersion matrix is defined as the inverse of the information matrix $\mathcal{M}$, the latter being a measure of the information content of the data from an experiment. The expressions for these matrices are

$$\mathcal{M} = \sum_{i=1}^{N_s} \mathbf{S}(t_i)^\mathsf{T}\mathbf{R}^{-1}\mathbf{S}(t_i), \tag{5.6}$$

$$\mathcal{D} = \mathcal{M}^{-1}. \tag{5.7}$$

Here, N_s is the number of samples taken during the measurement and $\mathbf{R}$ the discrete noise covariance matrix which is unknown prior to the optimization process. A very common way is to assume no correlation among the system outputs and moreover that the variance of all system outputs is equal. Using this assumption $\mathbf{R}$ reduces to the identity matrix $\mathbf{I}$.

In [4] several norms or optimality metrics are suggested for optimal input design. Investigating the determinant (*D-optimality*) or eigenvalues (*E-optimality*) of $\mathcal{M}$ in a cost functional does not allow to apply straightforward variational calculus. Hence, the so called *A-optimality* is chosen, which incorporates the trace of $\mathcal{M}$. Moreover, common optimization algorithms search for the minimum of a cost functional $J(\mathbf{u})$. Therefore the maximization of the information content leads to a cost functional using the negative trace of $\mathcal{M}$.

For further derivations the cost functional is defined as a continuous function. Instead of forming the sum in Eq. (5.6) the inner product of columns of $\mathbf{S}$ are integrated over time. The resulting cost functional to be minimized then reads

$$J = -\int_0^T \sum_{j=1}^{l} \mathbf{y}_{\xi_j}(t)^{\mathsf{T}} \mathbf{y}_{\xi_j}(t) \mathrm{d}t. \tag{5.8}$$

Different from the common approach of searching directly for bang-bang solutions via dynamic programming methods, here a variational approach is chosen.

5.1.3 Adoption of optimal control approach

Calling in mind that the cost functional in Eq. (5.8) uses the system sensitivities, and hence outputs of the extended system of Eq. (5.4), optimal input design can be seen as the standard problem of optimal control for the extended system. Similar to the computation of the gradient $\nabla J(\xi)$ presented in Chapter 3, the variation of the cost functional Eq. (5.8) may be performed using an adjoint approach.
As the cost functional in Eq. (5.8) depends on the system sensitivities and therefore on the states $\mathbf{z}$ of the extended system in Eq. (5.4), the cost functional can be defined as follows:

$$J(\mathbf{u}) = \int_0^T h(\mathbf{z}, \mathbf{u}, t) \mathrm{d}t. \tag{5.9}$$

The problem is to find control variables $\mathbf{u}(t)$ which minimize this function. In order to provide a search direction for the optimization process the variation of the cost functional with respect to the parameters has to be evaluated. Again, the starting point of the adjoint method is to add the system equations in Eq. (5.4) to the integrand in Eq. (5.9). Hence, the extended cost functional reads

$$J(\mathbf{u}) = \int_0^T \left[h + \mathbf{p}^{\mathsf{T}} (\tilde{\mathbf{f}} - \dot{\mathbf{z}}) \right] \mathrm{d}t. \tag{5.10}$$

Since the system equations are satisfied, the actual value of J does not depend on the selection of the functions $\mathbf{p}(t)$. Introducing the Hamiltonian

$$\mathcal{H}(\mathbf{z}, \mathbf{p}, \mathbf{u}, t) = h(\mathbf{z}, \mathbf{u}, t) + \mathbf{p}^{\mathsf{T}} \tilde{\mathbf{f}}(\mathbf{z}, \mathbf{u}, t) \tag{5.11}$$

Eq. (5.10) becomes

$$J(\mathbf{u}) = \int_0^T \left[\mathcal{H} - \mathbf{p}^{\mathsf{T}} \dot{\mathbf{z}} \right] \mathrm{d}t.$$

For a given forward solution $\mathbf{z}(t)$ of the system equations Eq. (5.4) with control variables $\mathbf{u}(t)$ and fixed parameters ξ, the variation of $\delta\mathbf{u}$ will result in variations of $\delta\mathbf{z}(t)$. This again will result in a variation δJ of the functional. Considering first order terms only, δJ is given by

$$\delta J = \int_0^T \left[\mathcal{H}_{\mathbf{u}} \delta\mathbf{u} + \mathcal{H}_{\mathbf{z}} \delta\mathbf{z} - \mathbf{p}^{\mathsf{T}} \delta\dot{\mathbf{z}} \right] \mathrm{d}t \tag{5.12}$$

where $\mathcal{H}_{\mathbf{u}}$ and $\mathcal{H}_{\mathbf{z}}$ denote the partial derivatives of $\mathcal{H}$ with respect to the vector of system inputs $\mathbf{u}$ and the states of the extended system $\mathbf{z}$ respectively. In order to avoid the computation of the variations of $\mathbf{z}$, the last term of Eq. (5.12) is integrated by parts, and therefore the variation of the cost functional reads

$$\begin{aligned}\delta J &= \int_0^T \left[\mathcal{H}_{\mathbf{u}}\delta\mathbf{u} + \mathcal{H}_{\mathbf{z}}\delta\mathbf{z} + \dot{\mathbf{p}}^{\mathsf{T}}\delta\mathbf{z}\right] \mathrm{d}t - \mathbf{p}^{\mathsf{T}}\delta\mathbf{z}\Big|_0^T \\ &= \int_0^T \left[\mathcal{H}_{\mathbf{u}}\delta\mathbf{u} + \left(\mathcal{H}_{\mathbf{z}} + \dot{\mathbf{p}}^{\mathsf{T}}\right)\delta\mathbf{z}\right] \mathrm{d}t - \mathbf{p}(T)\delta\mathbf{z}(T).\end{aligned} \tag{5.13}$$

Herein, the variation $\delta\mathbf{z}(0) = \mathbf{0}$ is already neglected as the initial conditions are prescribed independently from the actual choice of inputs. Now, in order to eliminate the term multiplied with $\delta\mathbf{z}$, a system of adjoint equations for the adjoint variables $\mathbf{p}(t)$ can be formed. The adjoint system reads

$$\dot{\mathbf{p}} = -\mathcal{H}_{\mathbf{z}}^{\mathsf{T}} \quad \text{and} \quad \mathbf{p}(T) = \mathbf{p}_f. \tag{5.14}$$

This set of equations may be solved backwards in time, since there is only an initial condition at time $t = T$. At this point there is no constraint on the adjoint states at $t = T$, and therefore they can be chosen arbitrarily. An option presented in [48] is to add a further term to the cost function which allows to consider end conditions for the system states $\mathbf{z}$. This term is commonly denoted as *scrap-function*. Using Eq. (5.14) the variation of J is given by Eq. (5.13)

$$\delta J = \int_0^T \mathcal{H}_{\mathbf{u}}\delta\mathbf{u}\mathrm{d}t.$$

In order to achieve the largest possible decrease of δJ, $\delta\mathbf{u}(t)$ is chosen in the direction of $\mathcal{H}_{\mathbf{u}}^{\mathsf{T}}$. Due to nonlinearities in the cost functional this direction is only valid near the current system input $\mathbf{u}(t)$. Therefore the update has to be done incrementally by using

$$\delta\mathbf{u}(t) = -\kappa\mathcal{H}_{\mathbf{u}}^{\mathsf{T}} \tag{5.15}$$

with small numbers κ. Finding a value for κ that minimizes J may be done by applying one of the optimization schemes presented in Section 4.3.

5.1.4 Considering model input constraints

In most cases, maximizing the information content in measurements leads to a maximization of the energy put into the system under consideration. Therefore, the system inputs to be optimized have to be constrained in a way that applicable optimization results are generated. One main difficulty is the direct influence of such constraints onto the optimization process. They insert further nonlinearities,

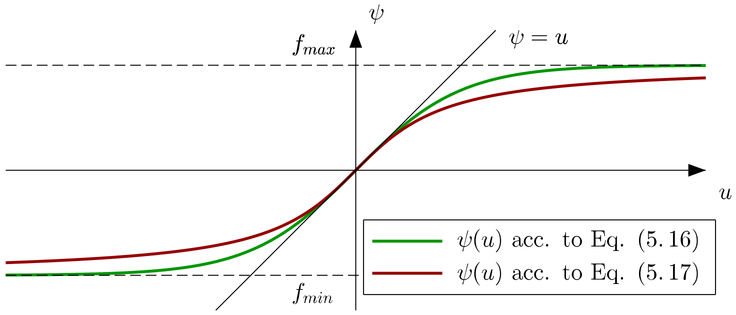


Figure 5.1: Comparison of constraining functions

and therefore affect the convergence negatively. The approach chosen here is to transform the input in such a way that the transition from unconstrained input u_i to constrained input $\psi_i(u_i)$ is smooth. In order to satisfy $\psi_i(u_i) \in [u_i^-, u_i^+]$ the authors in [27] propose to use the function

$$\psi_i(u_i) = u_i^+ - \frac{u_i^+ - u_i^-}{1 + \exp(s u_i)}, \quad s = \frac{4}{u_i^+ - u_i^-} \tag{5.16}$$

for constraining the input u_i. The term s is introduced in order to correct the slope at $u_i = 0$ to $\psi_i'(0) = 1$. Another possibility for constraining inputs is to use the arctangent function

$$\psi_i(u_i) = \frac{u_i^+ + u_i^-}{2} + s \ \arctan\left(\frac{1}{s} u_i\right), \quad s = \frac{u_i^+ - u_i^-}{\pi} \tag{5.17}$$

with again s being the function to correct the slope at $u_i = 0$. In Fig. 5.1 these two functions are displayed. As a function with distinct slopes in a wide range is preferred the arctangent function is suggested as constraining function, although the direct relation between $\psi_i \approx u_i$ is only valid in a very small region.

5.2 Parameter Identification

The purpose of optimal input design is to generate the excitation for a subsequent parameter estimation. Application of the computed optimal input onto the real system leads to an optimal desired trajectory. In the following, an approach utilizing the system sensitivities derived in Section 5.1.1 is presented therefore.

In direct comparison to the optimization of inputs in the previous section optimizing the parameters is less expensive. Choosing the root mean square error

$$E_{\mathrm{RMS}}(\xi) = \sum_{i=1}^{N_s} \frac{1}{2} \Delta\mathbf{y}_i^{\mathsf{T}} \Delta\mathbf{y}_i, \quad \Delta\mathbf{y}_i = \begin{bmatrix} \bar{y}_{1,i} - y_1(t_i) \\ \bar{y}_{2,i} - y_2(t_i) \\ \vdots \\ \bar{y}_{m,i} - y_m(t_i) \end{bmatrix}$$

as performance measure simplifies further derivations for the optimization procedure. Here, N_s is the number of sampling points in the measurement and $\Delta\mathbf{y}$ the deviation from simulation data $\mathbf{y}$ to measured data $\bar{\mathbf{y}}$ at a time point t_i. When arranging the output sensitivities at this timepoint in the form of the sensitivity matrix Eq. (5.2) the gradient of $E_{\mathrm{RMS}}(\xi)$ can be written as

$$\nabla E_{\mathrm{RMS}}(\xi) = -\sum_{i=1}^{N_s} \Delta\mathbf{y}_i^{\mathsf{T}} \mathbf{S}(t_i).$$

Performing another differentiation on ∇E_{RMS} and neglecting higher order terms, the Hessian reads

$$\nabla^2 E_{\mathrm{RMS}} = \sum_{i=1}^{N_s} \left(\mathbf{S}(t_i)^{\mathsf{T}} \mathbf{S}(t_i) - \frac{\partial}{\partial \xi} \left(\Delta\mathbf{y}^{\mathsf{T}}(t_i) \mathbf{S}(t_i) \right) \right) \approx \sum_{i=1}^{N_s} \mathbf{S}(t_i)^{\mathsf{T}} \mathbf{S}(t_i)$$

in which $\Delta\mathbf{y}^{\mathsf{T}}(t_i)$ is considered as constant in the case of the second term. With the exact solution for ∇E_{RMS} and an approximation for $\nabla^2 E_{\mathrm{RMS}}$ the Newton-Method may be applied. Performing a sufficient number of iterations

$$\Delta\xi_{k+1} = -(\nabla^2 E_{\mathrm{RMS}}(\xi_k))^{-1} \nabla E_{\mathrm{RMS}}(\xi_k)$$

the optimal set of parameters can be determined.

5.3 Illustrative examples

In the preceding chapter a method for finding the optimal excitation of a system with parameters that are to be identified has been presented. Similar to the methods for parameter identification shown in Chapter 4 the approach chosen aims at finding the optimum in an iterative way by using the gradient of a cost function. Nevertheless, according to *Pontryagin's minimum principle* it is possible to find an optimal solution for the system inputs $\mathbf{u}$ that minimize the according cost function without the need for a numerical iteration scheme. In the following the general solution for linear systems is derived in order to show how the optimal input may be computed in such a way. As this is quite hard for systems with more than one degree of freedom and several parameters, the single mass oscillator is analyzed in detail. In both examples presented thereafter, the two mass oscillator and the cart pendulum system, the iterative method is applied.

5.3.1 Optimal input for a linear system

The general solution of a linear system with a single constant input u is derived in the following. First, the system equations may be written in the form

$$\begin{pmatrix} \dot{\mathbf{z}} \\ \dot{\mathbf{p}} \end{pmatrix} = \mathbf{A} \begin{pmatrix} \mathbf{z} \\ \mathbf{p} \end{pmatrix} + \mathbf{b}u \tag{5.18}$$

which results in the general solution

$$\begin{pmatrix} \mathbf{z}(t) \\ \mathbf{p}(t) \end{pmatrix} = \underbrace{e^{\mathbf{A}t}}_{\mathbf{\Phi}(t)} \mathbf{C} - \underbrace{\mathbf{A}^{-1}\mathbf{b}}_{\mathbf{\Psi}} u. \tag{5.19}$$

Herein, $\mathbf{C}$ denotes the vector of integration constants that may be found by inserting in boundary conditions.
The system input may be chosen such that the cost function J given in Eq. (5.9) is minimized. Typically the system input has to be chosen in an interval $u \in [u^-, u^+]$ with the lower bound u^- and upper bound u^+ respectively. Therefore, according to *Pontryagin's minimum principle* [38] the optimal input u^* must fulfill the condition

$$\mathcal{H}(\mathbf{z}^*, \mathbf{p}^*, u^*) \leq \mathcal{H}(\mathbf{z}^*, \mathbf{p}^*, u) \quad \forall \quad t \tag{5.20}$$

where $\mathbf{z}^*$ and $\mathbf{p}^*$ denote the system states and the according adjoint variables generated by the optimal input u^*. Using the system equations Eq. (5.18) and inserting in the general definition of the Hamiltonian in Eq. (5.11) results in

$$\mathcal{H} = \mathbf{p}^{\mathsf{T}} \left(\mathbf{A}\mathbf{z} + \mathbf{b}u\right) + h(\mathbf{z})$$

where h is assumed to depend on the states $\mathbf{z}$ only. Hence, the inequality in Eq. (5.20) may further be simplified, reading

$$\begin{aligned} \mathcal{H}(\mathbf{z}^*, \mathbf{p}^*, u^*) &\leq \mathcal{H}(\mathbf{z}^*, \mathbf{p}^*, u) \\ \left(\mathbf{p}^*\right)^{\mathsf{T}} \mathbf{b}u^* &\leq \left(\mathbf{p}^*\right)^{\mathsf{T}} \mathbf{b}u \\ \left(\mathbf{p}^*\right)^{\mathsf{T}} \mathbf{b}\left(u^* - u\right) &\leq 0 \end{aligned} \tag{5.21}$$

and finally leading to the switching condition

$$\left(\mathbf{p}^*\right)^{\mathsf{T}} \mathbf{b} = 0 \tag{5.22}$$

and the optimal input

$$u^* = \begin{cases} u^-, & \text{if } \left(\mathbf{p}^*\right)^{\mathsf{T}} \mathbf{b} > 0 \\ u^+, & \text{if } \left(\mathbf{p}^*\right)^{\mathsf{T}} \mathbf{b} < 0 \end{cases} .$$

Assuming n switching points, the interval $[0,T]$ may be divided into $n+1$ subintervals, with a solution $\mathbf{z}^{(i)}(t)$ and $\mathbf{p}^{(i)}(t)$. At all switching points t_i the states and adjoint variables must fulfill the transition condition

$$\begin{pmatrix}\mathbf{z}^{(i)}(t_i)\\ \mathbf{p}^{(i)}(t_i)\end{pmatrix} = \begin{pmatrix}\mathbf{z}^{(i+1)}(t_i)\\ \mathbf{p}^{(i+1)}(t_i)\end{pmatrix}.$$

Inserting the solution from Eq. (5.19) the equation system for the integration constants $\mathbf{C}$ reads

$$\begin{aligned}\mathbf{\Phi}(t_i)\mathbf{C}^{(i)} + (-1)^i\mathbf{\Psi}u^{(i)} &= \mathbf{\Phi}(t_i)\mathbf{C}^{(i+1)} + (-1)^{i+1}\mathbf{\Psi}u^{(i+1)}\\ \mathbf{\Phi}(t_i)\left[\mathbf{C}^{(i)} - \mathbf{C}^{(i+1)}\right] &= -2(-1)^i\mathbf{\Psi}u^{(i)}\end{aligned}$$

where $u^{(i+1)}$ is assumed to be $-u^{(i)}$ and therefore $u^- = -u^+$. After rearranging the equation system the integration constants for successive intervals can be computed by

$$\mathbf{C}^{(i+1)} = \mathbf{C}^{(i)} + 2(-1)^i\mathbf{\Phi}^{-1}(t_i)\mathbf{\Psi}u^{(i)}.$$

Additionally, the states at $t=0$ must satisfy the initial condition

$$\mathbf{z}^{(1)}(0) = \mathbf{0}$$

and therefore

$$\mathbf{z}^{(1)}(0) = \underbrace{\begin{pmatrix}\mathbf{I} & \mathbf{0}\end{pmatrix}}_{\mathbf{L}}\begin{pmatrix}\mathbf{z}^{(1)}(0)\\ \mathbf{p}^{(1)}(0)\end{pmatrix} = \mathbf{L}\left[\mathbf{\Phi}(0)\mathbf{C}^{(i)} - \mathbf{\Psi}u^{(1)}\right] = \mathbf{L}\left[\mathbf{C}^{(i)} - \mathbf{\Psi}u^{(1)}\right] = \mathbf{0}$$

must hold. As the adjoint variables at the final time T may be chosen arbitrarily, they can be set to zero

$$\mathbf{p}^{(n+1)}(T) = \mathbf{0},$$

leading to additional equations for the unknown constants $\mathbf{C}^{(i+1)}$:

$$\mathbf{p}^{(n+1)}(T) = \underbrace{\begin{pmatrix}\mathbf{0} & \mathbf{I}\end{pmatrix}}_{\mathbf{R}}\begin{pmatrix}\mathbf{z}^{(n+1)}(T)\\ \mathbf{p}^{(n+1)}(T)\end{pmatrix} = \mathbf{R}\left[\mathbf{\Phi}(T)\mathbf{C}^{(i+1)} - \mathbf{\Psi}u^{(n+1)}\right] = \mathbf{0}.$$

Solving the resulting equations for $\mathbf{C}^{(1)}\dots\mathbf{C}^{(n+1)}$ gives expressions for $\mathbf{z}^{(1)}\dots\mathbf{z}^{(n+1)}$ and $\mathbf{p}^{(1)}\dots\mathbf{p}^{(n+1)}$ in the still undetermined switching points $t_1\dots t_n$ and the time t. By starting with an initial guess for the switching points, the Newton-Raphson method may be used to compute the switching points by inserting into the switching condition Eq. (5.22).

5.3.2 Single mass oscillator

In the following the procedure given above will be applied onto the single mass oscillator system (see Section 3.1.1) without damping $d = 0$.
First, the system states and the system sensitivities may be combined in the extended state vector

$$\mathbf{z} = \begin{pmatrix} x \\ v \\ x_c \\ v_c \end{pmatrix}$$

and therefore the equations of motion for the extended system read

$$\dot{\mathbf{z}} = \begin{pmatrix} z_2 \\ -\omega^2 z_1 + u \\ z_4 \\ -\omega^2 z_3 - z_1 \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ -\omega^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ -1 & 0 & -\omega^2 & 0 \end{pmatrix} \mathbf{z} + \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} u. \tag{5.23}$$

The system input $u \in [u^-, u^+]$ may be chosen such that the sensitivity with respect to the parameter $c = \omega^2$ is maximized, leading to the cost function

$$J = -\int_0^T \frac{1}{2} z_3^2 \, \mathrm{d}t$$

to be minimized. Inserting in Eq. (5.11) the Hamiltonian is given by

$$\mathcal{H} = -\frac{1}{2} z_3^2 - p_1 z_2 + p_2(-\omega^2 z_1 + u) + p_3 z_4 - p_4(\omega^2 z_3 + z_1). \tag{5.24}$$

According to Eq. (5.14) the adjoint system reads:

$$\begin{aligned} \dot{p_1} &= -\frac{\partial \mathcal{H}}{\partial z_1} = \omega^2 p_2 + p_4 \\ \dot{p_2} &= -\frac{\partial \mathcal{H}}{\partial z_2} = -p_1 \\ \dot{p_3} &= -\frac{\partial \mathcal{H}}{\partial z_3} = z_3 + \omega^2 p_4 \\ \dot{p_4} &= -\frac{\partial \mathcal{H}}{\partial z_4} = -p_3. \end{aligned} \tag{5.25}$$

For constant u the general solution of Eq. (5.23) and Eq. (5.25) can be determined in the form of Eq. (5.19).
As given in Eq. (5.21) the optimal input u^* must fulfill

$$p_2^*(u^* - u) < 0 \quad \forall \quad t.$$

As the inequality must hold for any admissible $u \in [u^-, u^+]$, this directly leads to a law for choosing between u^- and u^+:

$$u^* = \begin{cases} u^-, & \text{if } p_2^* > 0 \\ u^+, & \text{if } p_2^* < 0 \end{cases} . \tag{5.26}$$

The general solution for the adjoint variable $p_2(t)$ can be found to be

$$\begin{aligned} p_2(t, \mathbf{C}, u) = &- \frac{1}{48\omega^8}\Bigg\{\bigg[t\omega^4 \Big(C_6 t^2 - 3C_5 t + 18C_8\Big) + 6t\omega^6 \left(C_7 t + 4C_3\right) - 15C_6 t\omega^2 \\ &- 48C_2\omega^8\bigg] \cos(\omega t) + \omega\bigg[- \omega^4 \Big(C_5 t^3 - 6C_8 t^2 + 6C_7 t + 24C_3\Big) \\ &+ 24\omega^6 \left(C_4 t + 2C_1\right) + 3\omega^2 \left(t \left(C_5 - 2C_6 t\right) - 6C_8\right) + 15C_6\bigg] \sin(\omega t) + 48u\Bigg\}. \end{aligned}$$

Note, that the integration constants $\mathbf{C} = [C_1, \ldots, C_8]^\mathsf{T}$ are different for each time intervall between two switching points and still depend on the switching points. Assuming a single switching point t_1 in the interval $t \in [0, T]$ and $u^{(1)} = u^+$ for $t \in [0, t_1]$, the function

$$f(t_1) := p_2(t_1, \mathbf{C}^{(1)}(t_1), u^+) \tag{5.27}$$

may be used for finding the switching point t_1^*. The roots of $f(t_1) = 0$ are equivalent to points where $p_2(t)$ changes its sign and therefore, according to Eq. (5.26), the input $u^*(t)$ is determined. Using the numerical values $\omega = 1$, $T = 2\pi$ and $u^+ = -u^- = 1$ the function $f(t_1)$ can be evaluated. The graph given in Fig. 5.2 may then be used to get an appropriate initial value for finding the root of $f(t_1)$ numerically. The adjoint variable $p_2(t)$ may be defined as a piecewise function:

$$p_2^*(t) = \begin{cases} p_2(t, \mathbf{C}^{(1)}, u^+), & \text{if } t \leq t_1^* \\ p_2(t, \mathbf{C}^{(2)}, u^-), & \text{if } t > t_1^* \end{cases} .$$

In order to fulfill the optimality condition, $p_2^*(t)$ must not have further roots in $[0, T]$ and u^* must have the opposite sign of p_2^*. This can be checked by analyzing the plot in Fig. 5.3. By numerical evaluation of $J(t_1)$ the cost function may also be plotted for $t_1 \in [0, T]$ and used for verification of the optimal input u^*. As it can be seen in Fig. 5.4 the switching point t_1^* is a local minimum of J.

5.3.3 Two mass oscillator

Different from the previous example, the optimal excitation of the two mass oscillator shown in Fig. 5.5(a) should be computed by the iterative approach proposed in this chapter. The mass m_1 is excited by the force $F(t)$, whereas this force is constrained to a maximum amplitude of F_{max} by means of a constraining

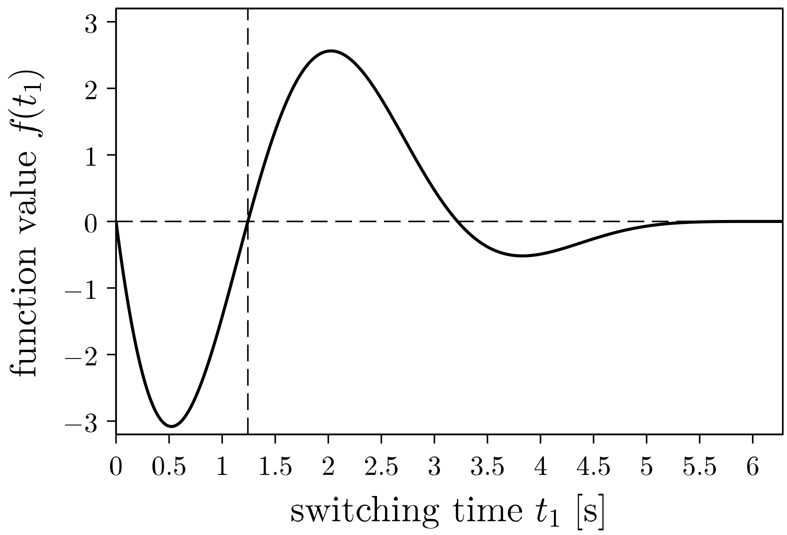


Figure 5.2: Function $f(t_1)$ used for estimating t_1^*

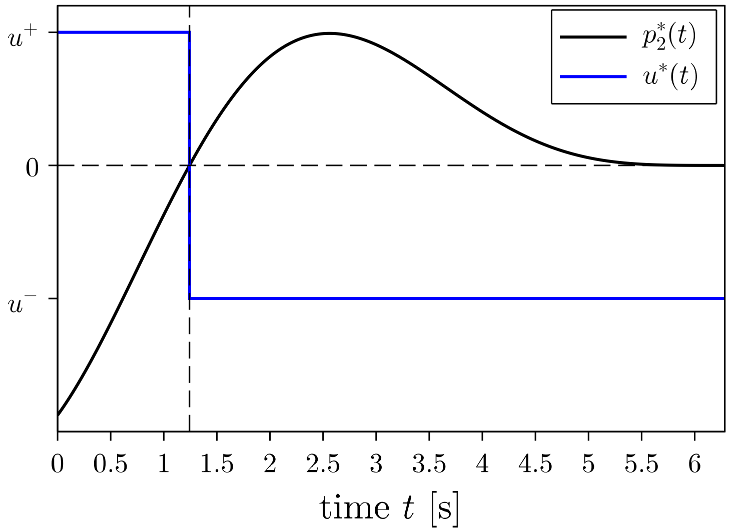


Figure 5.3: Adjoint variable $p_2^*(t)$ and optimal input $u^*(t)$ for one switching point

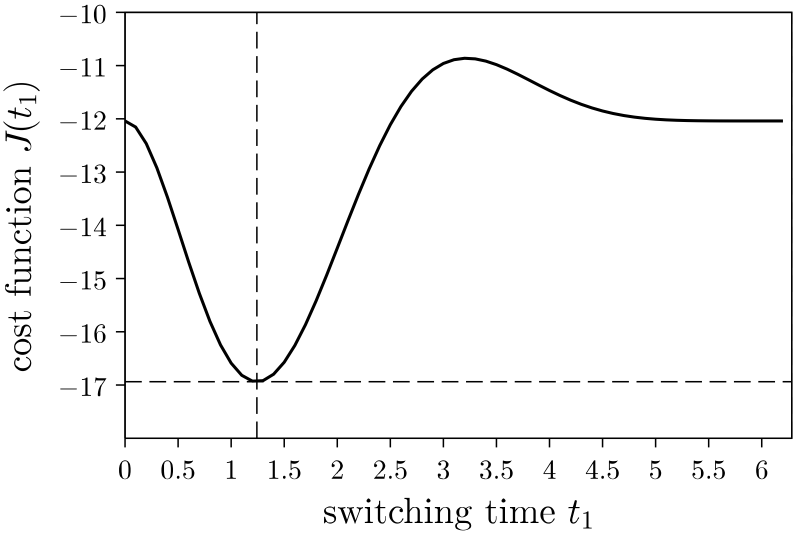
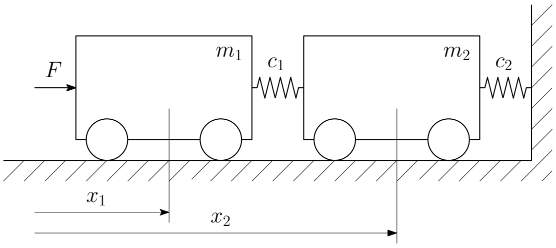


Figure 5.4: Cost function $J(t_1)$



(a)

parameter	value	unit
m_1	1.0	kg
m_2	0.1	kg
c_1	2000.0	N/m
F_{max}	10.0	N

(b)

Figure 5.5: An excitation signal F for the two mass oscillator in (a) is searched. In (b) parameters necessary for the numerical simulation are specified.

function presented in Section 5.1.4. Assuming that the position x_2 is measured during an experiment ($y = x_2$), a time history $F(t_i)$ maximizing the information content with respect to the stiffness c_2 and therefore using a cost functional in the form of Eq. (5.8) should be computed at discrete timepoints $t_i = \{t_0, t_1, \ldots, T\}$, with $T = 1\,\mathrm{s}$. For starting the iterative optimization process the initial input is set to a constant value $F_0(t_i) = 1/2\ F_{max}$. The parameter setting used can be found in Fig. 5.5(b) and c_2 is set to 1000 N/m initially. In order to avoid a motion of the bodies at $t > T$ a scrap function mentioned in Section 5.1.3 is specified such that the end velocities are set to $v_1(T) = v_2(T) = 0$. In Fig. 5.6 the convergence history for the input optimization and in Fig. 5.7 the resulting constrained input signal is

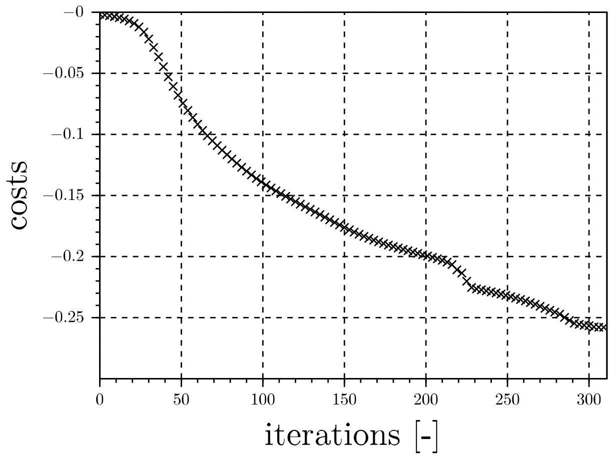


Figure 5.6: Convergence history for input optimization of two mass oscillator

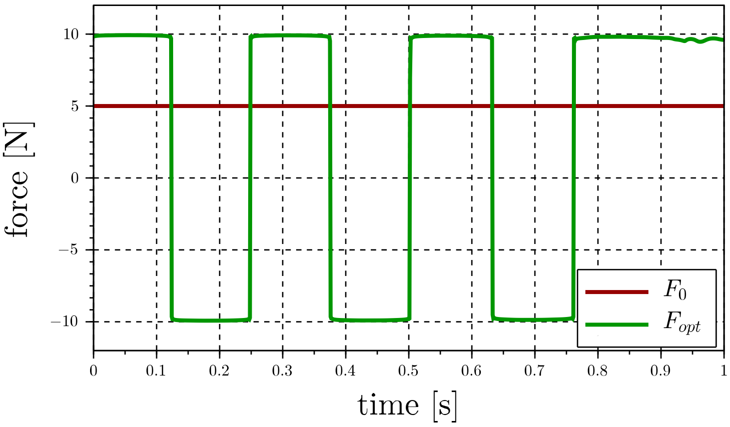


Figure 5.7: Optimized system input for two mass oscillator

depicted. Due to the linear convergence rate of the gradient method the convergence history shows quite poor but stable behavior. As this is only an example without a physical realization the measurements necessary for the parameter identification are generated by simulation using the optimized excitation force F_{opt} and an assumed stiffness coefficient $c_2 = 1000\,\mathrm{N/m}$. In order to include some kind of measurement noise the computed system outputs are superimposed with zero-mean Gaussian noise. In Fig. 5.9, Fig. 5.10 and Fig. 5.11 these measurements are presented for different standard deviations. For comparison, also the measurement using the initial excitation signal F_0 is displayed, which features smaller amplitudes and

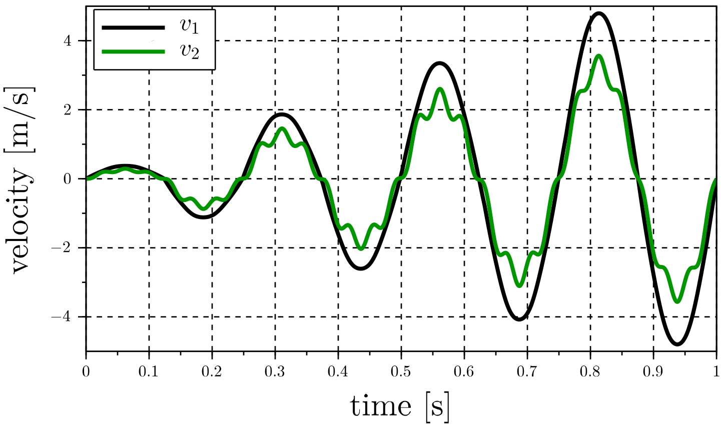


Figure 5.8: Velocity v_1 and v_2 for F_{opt} observing end condition

obviously leads to uncertainties of the parameter identification process. The velocity plot in Fig. 5.8 shows that the desired end conditions for v_1 and v_2 are fulfilled in the case of using F_{opt}.

Now, the main aim of optimal input design is to improve the quality of the parameter identification result and as a side benefit some constraints on the system states and inputs can be regarded. In order to show the advantage of the input generation for the actual example in Fig. 5.9, Fig. 5.10 and Fig. 5.11 the RMS error evaluated for parameter values c_2 near $\bar{c_2}$ is shown. According to the plot of RMS errors in Fig. 5.9, utilizing F_{opt} does not improve the shape of the cost functional in comparison to F_0 when using the unbiased measurements. The more noisy the sensor recording used for computing E_{RMS}, the more advantageous the optimal input F_{opt} influences the shape of the RMS error used for parameter identification (see Fig. 5.10 and Fig. 5.11). As this is only an one-dimensional optimization problem, no real benefit with regard to a speed up of the optimization process can be gained. Advantages may be the increased curvature of E_{RMS} and the possibility to include end conditions on the system outputs.

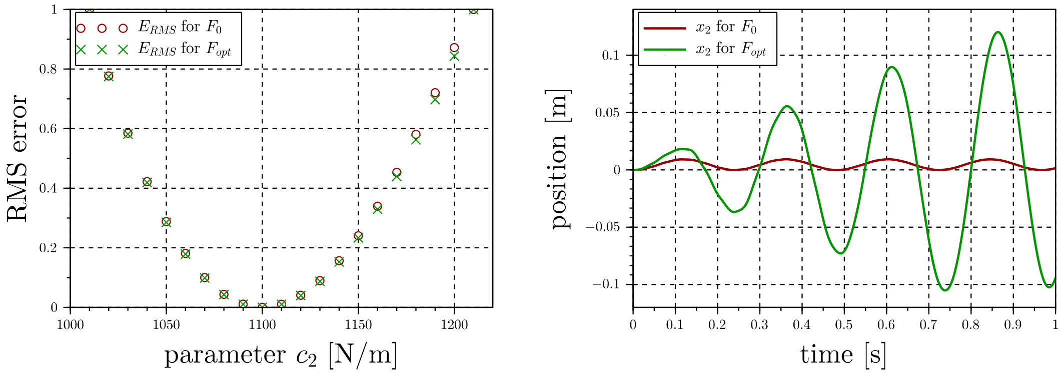


Figure 5.9: Evaluation of E_{RMS} and $x_2(t)$ using F_0 and F_{opt} for $\sigma = 0$

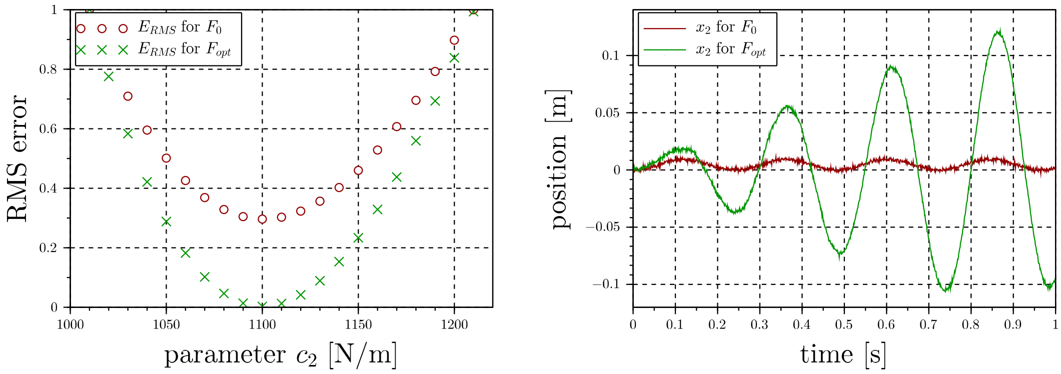


Figure 5.10: Evaluation of E_{RMS} and $x_2(t)$ using F_0 and F_{opt} for $\sigma = 10^{-6}$

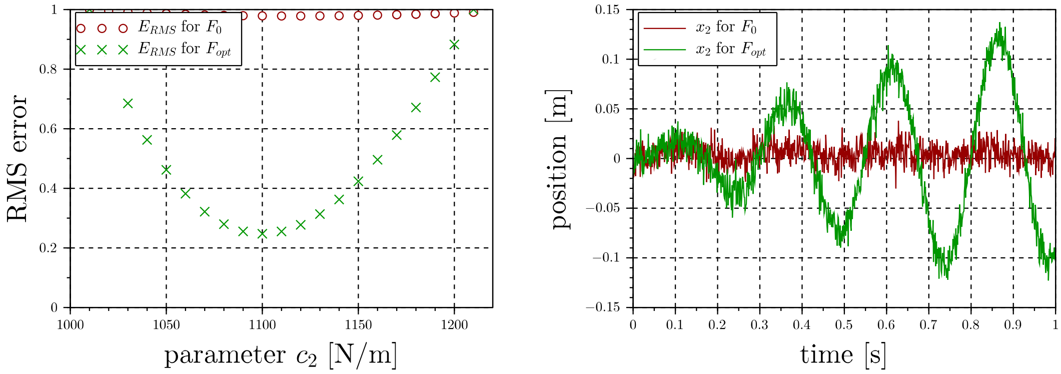
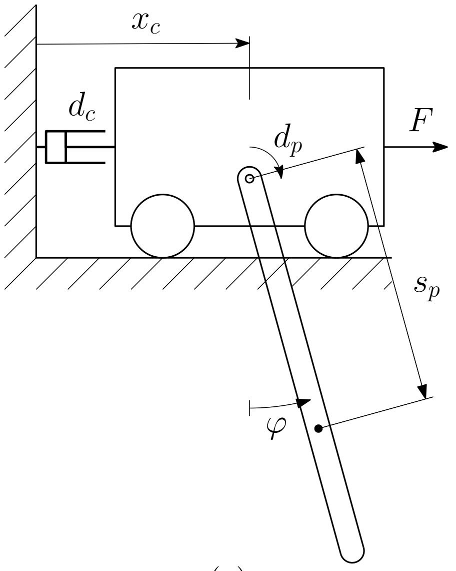


Figure 5.11: Evaluation of E_{RMS} and $x_2(t)$ using F_0 and F_{opt} for $\sigma = 10^{-4}$

5.3.4 Cart pendulum system

A system consisting of a translational moving cart and a pendulum mounted at its center of mass is studied next. Figure 5.12(a) shows the geometric description of the cart pendulum system. The cart is only allowed to move along the x-axis leading to a two-dimensional motion of the pendulum, see again Fig. 5.12(a). The coordinates chosen to describe the system's motion are the cart position x_c and the absolute pendulum angle φ resulting in the vector of generalized coordinates $\mathbf{q} = [x_c, \varphi]^\mathsf{T}$. Linear friction torque/force is considered for the revolute joint between cart and pendulum and also between ground and the cart, defined by friction coefficients d_p and d_c. The distance from the revolute joint to the pendulum's center of gravity is abbreviated by s_p. The numeric values used for simulation are defined in the table in Fig. 5.12(b). The goal of the optimization is to find the excitation force $F(t)$ that is best suited to generate measurements $\varphi(t)$ allowing the identification of s_p and d_p. Again the force $F(t)$ is constrained to the intervall $[-F_{max}, F_{max}]$. Incorporating a scrap function using $\varphi(T) = v_c(T) = \dot{\varphi}(T) = 0$ prevents from movements at $t > T$.

In Fig. 5.13 the convergence history for the input optimization is depicted. Figure 5.14 shows the resulting optimized input signal F_{opt}, a signal F_{comp} used for comparison purposes and the initial signal F_0. In Fig. 5.14 the signal F_{comp} is chosen as a sine wave with the period T and the amplitude equals the force constraint $|F_{comp}| = F_{max}$. The plot of costs over iterations in Fig. 5.13 shows stable behavior and convergence at $n = 200$ iterations. At about $n = 180$ iterations one can observe a jump, which may result from to the nonlinearity of the model structure. Unlike the previous example with only one parameter to identify, two parameters are now searched for. Comparing the error functions thus leads to threedimensional plots or contour plots, where one can study differences in shapes for different excitation signals. Although the signals of the considered system output φ (see Fig. 5.15) are comparable in amplitudes for F_{opt} and F_{comp} the RMS errors differ significantly. In Fig. 5.16 and Fig. 5.17 the contour plots for F_{opt} and F_{comp} are depicted, where both parameters d_p and s_p are varied in range of $\pm 10\%$ of the nominal value. The contours represent values of E_{RMS} at levels, which are chosen equally for both plots in Fig. 5.16 and Fig. 5.17. Analyzing the plots leads to three main differences. First, a small rotation of the functional can be detected, where the elliptical contours are more aligned with the coordinate axes in the case of using F_{opt}. Further, a slight compression of the functional can be noticed, which leads to a worse condition of the optimization problem, and therefore to poorer convergence for F_{opt}. Finally, the main difference of both functionals is the curvature, and hence the decreasing distance of level curves. The RMS error depicted in Fig. 5.17 therefore is flatter and more sensitive on biased signals of the measurements.



(a)

parameter	value	unit
mass of cart	2.0	kg
mass of pendulum	0.876	kg
s_p	0.162	m
d_c	0.1	Ns/m
d_p	0.1	Nms
F_{max}	10.0	N

(b)

Figure 5.12: An optimal input F is searched for the cart pendulum system in (a). In (b) parameters necessary for the numerical simulation are specified.

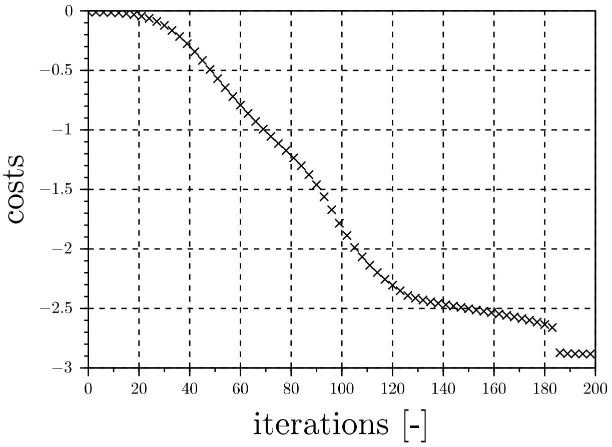


Figure 5.13: Convergence history for input optimization of cart pendulum system

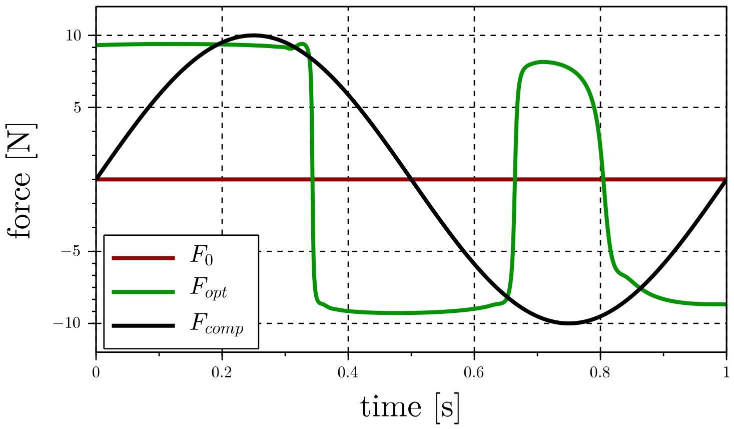


Figure 5.14: Optimized excitation signal for cart pendulum system

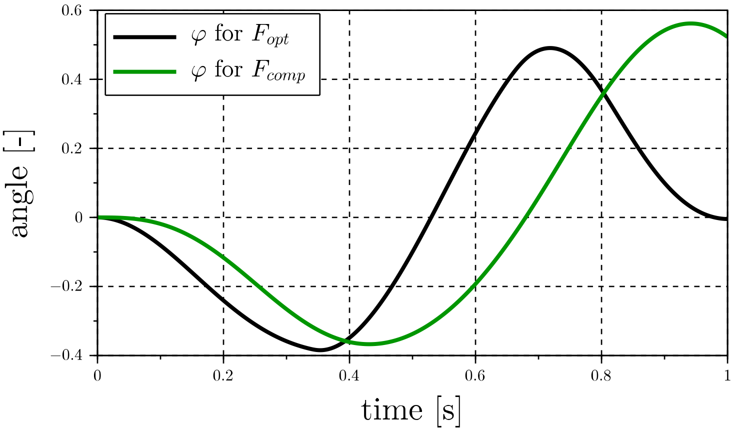


Figure 5.15: Pendulum angle $\varphi(t)$ for F_{opt} and F_{comp}

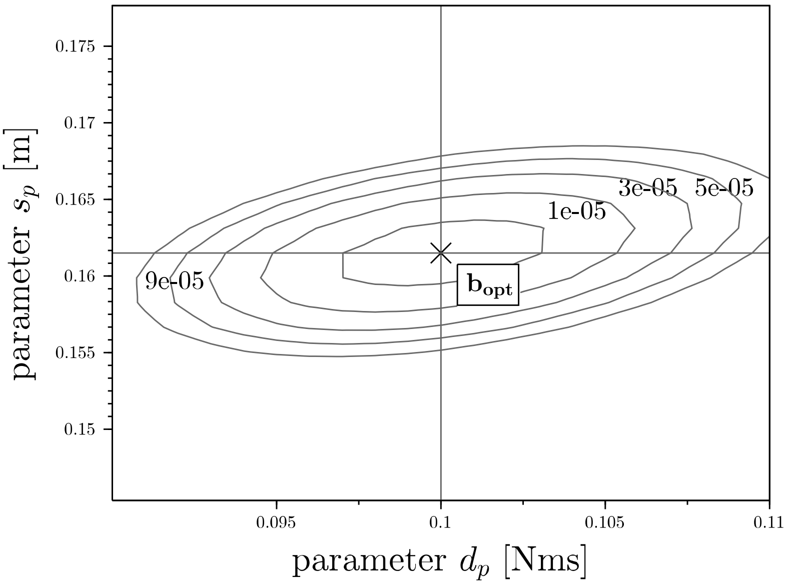


Figure 5.16: Contour plot of RMS error $E_{\text{RMS}}(\xi)$ for F_{opt}

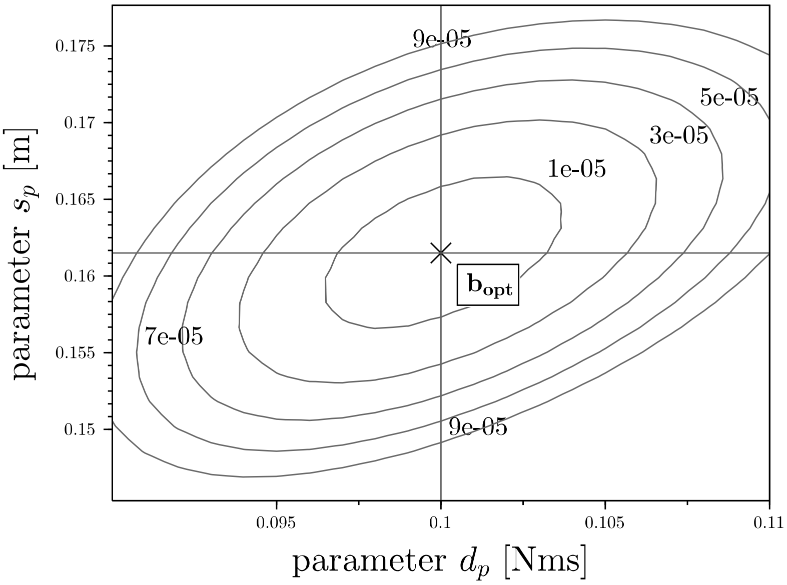


Figure 5.17: Contour plot of RMS error $E_{\text{RMS}}(\xi)$ for F_{comp}

CHAPTER 6

Conclusions and Outlook

State-of-the-art software tools allow for analyzing the behavior of complex multibody systems, e.g. crank drives, entire vehicles, or machine tools. The quality thereof mainly depends on the modeling chosen and the parameters used to quantify system components. Uncertainties regarding parameters are commonly reduced by the use of identification methods, that use data from real world experiments for trimming the virtual system to behave like the real one. In the present thesis an identification method, using the adjoint sensitivity analysis and an iterative optimization approach, is derived. First, in order to clarify the multibody background a short overview of the description of multibody systems and suitable analyzing techniques are given. Before addressing the identification problem, the prerequisites, which are the system sensitivity analysis, the introduction of performance measures and the adjoint sensitivity analysis, are outlined. The derivation of algorithms is accompanied by an illustrative example that allows to find symbolic solutions.

Basically, the identification method can use any gradient based optimization tool available. The results shown in Chapter 4 are gained by employing the quasi-Newton method in combination with the BFGS algorithm for approximating the Hessian.

As the identification result mainly depends on the measurements taken from experiments, the precision of parameters may be increased by making use of optimized experiments. Therefore, within this theses a method for designing optimal inputs is proposed that uses the definition of optimality by means of a minimum standard deviation of identified parameters. Furthermore, it allows to set end conditions which prescribe states at the end of an experiment. Looking at the example of the cart pendulum this results in more robust measurement signals. Even when dealing with biased signals, more accurate parameter estimates are

generated.
The thesis shows the great potential of using the adjoint sensitivity analysis in the field of multibody dynamics. However, some points worth thinking about in future investigations are outlined in the following.

1. Extension of the presented derivations for the case, when parameters appear in the constraint equations $\mathbf{C}(\mathbf{q}, \xi)$, the mass matrix $\mathbf{M}(\mathbf{q}, \xi)$, or in the initial conditions $\mathbf{q}_0(\xi)$, $\mathbf{v}_0(\xi)$. Especially the last two points seem to be of great significance in the field of identifying inertia parameters.

2. The approach for optimizing system inputs in order to maximize the information about parameters is carried out for systems in minimal coordinate formulation only. In order to allow the inputs of complex multibody systems to be optimized, further derivations have to be done. Moreover, the proposed method may also handle different norms of the Fisher matrix $\mathcal{M}$ in order to not only optimize the information content but also the condition of the optimization problem, e.g. the spectral radius of the linearized Hessian computed from the sensitivities.

3. Currently, all optimization packages available are performing the evaluations of the performance measure under consideration in a sequential manner. In particular, the step size computation requires many evaluations of the performance measure, and therefore many forward simulations of the multibody system. This causes a lot of time and may decides whether an engineer makes use of an identification software or not. Therefore, the implementation of an optimization algorithm that makes use of distributed computing resources is suggested as a considerable improvement and may lead to an increased applicability of the identification method.

Bibliography

[1] R. Al-Khoury, C. Kasbergern, A. Scarapas, and J. Blaauwendraad. Spectral element technique for efficient parameter identification of layered media - part ii: Inverse calculation. *International Journal of Solids and Structures*, 38:8753–8772, 2001.

[2] W. Anderson and V. Venkatakrishnan. Aerodynamic design optimization on unstructured grid with a continuous adjoint formulation. *AIA 97-0643, 35th aerospace science meeting and exhibition*, -:–, 1997.

[3] V. I. Arnold. *Mathematical Methods of Classical Mechanics -*. Springer Science & Business Media, Berlin Heidelberg, 2013.

[4] S. P. Asprey and S. Macchietto. Designing robust optimal dynamic experiments. *Journal of Process Control*, 12:545–556, 2002.

[5] O. A. Bauchau. *Flexible Multibody Dynamics*. Springer Netherlands, 2011.

[6] E. Bertolazzi, F. Biral, and M. Da Lio. Symbolic-numeric indirect method for solving optimal control problems for large multibody systems. *Multibody System Dynamics*, 13:233–252, 2005.

[7] M. Best and T. Gordon. Suspension system identification based on impulse-momentum equations. *Vehicle System Dynamics*, 29(S1):598–618, 1998.

[8] D. Bestle and P. Eberhard. Analyzing and optimizing multibody systems. *Mech. Struct. & Mach.*, 20:67–92, 1992.

[9] C. Bottasso, A. Croce, L. Ghezzi, and P. Faure. On the solution of inverse dynamics and trajectory optimization problems for multibody systems. *Multibody System Dynamics*, 11:1–22, 2004.

[10] H. Bremer and F. Pfeiffer. *Elastische Mehrkörpersysteme*. Teubner, Stuttgart, 1992.

[11] C. G. Broyden. The convergence of a class of double-rank minimization algorithms. *Journal of the Institute of Mathematics and Its Applications*, 6:76–90, 1970.

[12] A. E. Bryson and Y. C. Ho. *Applied Optimal Control.* Hemisphere, Washington, DC, 1975.

[13] Y. Cao, S. Li, and L. Petzold. Adjoint sensitivity analysis for differential-algebraic equations: algorithms and software. *Journal of Computational and Applied Mathematics*, 149:171–191, 2002.

[14] Y. Cao, S. Li, L. Petzold, and R. Serban. Adjoint sensitivity analysis for differential-algebraic equations: the adjoint dae system and its numerical solution. *SIAM Journal on Scientific Computing*, 24:1076–1089, 2003.

[15] R. Castro-Triguero, S. Murugan, R. Gallego, and M. I. Friswell. Robustness of optimal sensor placement under parametric uncertainty. *Mechanical Systems and Signal Processing*, 41:268–287, 2013.

[16] H. A. Chianeh, J. D. Stigter, and K. J. Keesman. Optimal input design for parameter estimation in a single and double tank system through direct control of parametric output sensitivities. *Journal of Process Control*, 21:111–118, 2011.

[17] J. Ding, Z. Pan, and L. Chen. Second order adjoint sensitivity analysis of multibody systems described by differential-algebraic equations. *Multibody System Dynamics*, 18:599–617, 2007.

[18] J. Ding, Z. Pan, and L. Chen. Parameter identification of multibody systems based on second order sensitivity analysis. *International Journal of Non-Linear Mechanics*, 47:1105–1110, 2012.

[19] P. Eberhard. Adjoint variable method for sensitivity analysis of multibody systems interpreted as a continuous hybrid form of automatic differentiation. *Proceedings of the 2nd Int. Workshop on Computational Differentiation*, -:319–328, 1996.

[20] R.A. Fisher. *The Design of Experiments.* Oliver and Boyd, 1935.

[21] R. Fletcher. A new approach to variable metric algorithms. *Computer Journal*, 13/3:317–322, 1970.

[22] G. Franceschini and S. Macchietto. Model-based design of experiments for parameter precision: State of the art. *Chemical Engineering Science*, 63:4846–4872, 2008.

[23] C.W. Gear, G.K. Gupta, and B. Leimkuhler. Automatic integration of euler-lagrange equations with constraints. *Journal of Computational and Applied Mathematics*, 12&13:77–90, 1985.

[24] R. Ghanem and F. Romeo. A wavelet-based approach for model and parameter identification of non-linear systems. *International Journal of Non-Linear Mechanics*, 36:835–859, 2001.

[25] D. Goldfarb. A family of variable metric updates derived by variational means. *Mathematics of Computation*, 24/109:23–26, 1970.

[26] H. Goldstein, J. L. Safko, and C. P. Poole. *Classical Mechanics*. Pearson, 2001.

[27] K. Graichen, A. Kugi, N. Petit, and F. Chaplais. Handling constraints in optimal control with saturation functions and system extension. *Systems & Control Letters*, 59:671–679, 2010.

[28] Fredric J Harris. On the use of windows for harmonic analysis with the discrete fourier transform. *Proceedings of the IEEE*, 66(1):51–83, 1978.

[29] E. Haug, R. Wehage, and N. Mani. Design sensitivity analysis of large-scaled constrained dynamic mechanical systems. *J. Mech., Trans., and Automation*, 106:156–162, 1984.

[30] E. J. Haug and P. E. Ehle. Second-order design sensitivity analysis of mechanical system dynamics. *International Journal for Numerical Methods in Engineering*, 18:1699–1717, 1982.

[31] A. Held and R. Seifried. Gradient-based optimization of flexible multibody systems using the absolute nodal coordinate formulation. *In Proceedings of the ECCOMAS Thematic Conference Multibody Dynamics 2013, Zagreb, Croatia, 1-4 July 2013*, -:1–8, 2013.

[32] H. M. Hilber, T. J. R. Hughes, and R. L. Taylor. Improved numerical dissipation for time integration algorithms in structural dynamics. *Earthquake Eng. and Strut. Dynamics*, 5:283–292, 1977.

[33] A. Jameson. Aerodynamic shape optimization using the adjoint method. *VKI Lecture Series on Aerodynamic Drag Prediction and Reduction, von Karman Institute of Fluid Dynamics, Rhode St Genese*, -:3–7, 2003.

[34] C. Jauberthie, F. Bournonville, P. Coton, and F. Rendell. Optimal input design for aircraft parameter estimation. *Aerospace Science and Technology*, 10:331–337, 2006.

[35] C. Jauberthie, L. Denis-Vidal, P. Coton, and G. Joly-Blanchard. An optimal design procedure. *Automatica*, 42:881–884, 2006.

[36] K. J. Keesman and E. Walter. Optimal input design for model discrimination using Pontryagin's maximum principle: Application to kinetic model structures. *Automatica*, 50:1535–1538, 2014.

[37] C. Kelley. *Iterative Methods for Optimization.* SIAM - Society for Industrial and Applied Mathematics, Philadelphia, 1995.

[38] D. E. Kirk. *Optimal Control Theory.* Dover Publications, Inc., New York, 2004.

[39] J. Lardies and S. Gouttebroze. Identification of modal parameters using the wavelet transform. *International Journal of Mechanical Sciences*, 44:2263–2283, 2002.

[40] J. Lions. *Optimal Control of Systems Governed by Partial Differential Equations.* Springer Berlin Heidelberg New York, 1971.

[41] A. McNamara, A. Treuille, Z. Popovic, and J. Stam. Fluid control using the adjoint method. *ACM Transactions on Graphics (TOG) - Proceedings of ACM SIGGRAPH 2004*, 23:449–456, 2004.

[42] R. K. Mehra. Optimal input signals for parameter estimation in dynamic systems - survey and new results. *IEEE Transactions on automatic control*, 19:753–768, 1974.

[43] R. K. Mehra. Optimal inputs for linear system identification. *IEEE transactions on automatic control*, 19:192–200, 1974.

[44] K. Meyberg and P. Vachenauer. *Hoehere Mathematik 2.* Springer-Verlag Berlin Heidelberg, 2001.

[45] E. Morelli and V. Klein. Determining the accuracy of maximum likelihood parameter estimates with colored residuals. *NASA Contractor Report*, 194893:1–46, 1994.

[46] E. A. Morelli. Flight test of optimal inputs and comparison with conventional inputs. *Journal of Aircraft*, 36:389–397, 1999.

[47] K. Nachbagauer, T. Lauß, S. Oberpeilsteiner, and W. Steiner. Inverse dynamics toolbox for multibody system simulations. *Proceedings of ECCOMAS Thematic Conference on Multibody Dynamics, Prague, Tschechische Republik*, -:1–2, 2017.

[48] K. Nachbagauer, S. Oberpeilsteiner, K. Sherif, and W. Steiner. The use of the adjoint method for solving typical optimization problems in multibody dynamics. *Journal of computational nonlinear dynamics*, 10:1–19, 2014.

[49] S. Nadarajah and A. Jameson. A comparison of the continuous and discrete adjoint approach to automatic aerodynamic optimisation. *AIAA*, 0667:1–20, 2000.

[50] D. Negrut, G. Ottarson, R. Rampalli, and A. Sajdak. On an implementation of the hilber-hughes-taylor method in the context of index 3 differential-algebraic equations of multibody dynamics. *ASME Journal of Computational and Nonlinear Dynamics*, 2:73–85, 2007.

[51] N. M. Newmark. A method of computation for structural dynamics. *Journal of the Engineering Mechanics Devision*, pages 67–94, 1959.

[52] P. E. Nikravesh. *Computer-Aided Analysis of Mechanical Systems.* Prentice-Hall, Engelwood Cliffs, NJ, 1988.

[53] A. Oberai, N. Gokhale, and G. Feijoo. Solution of inverse problems in elasticity imaging using the adjoint method. *Inverse Problems*, 19:297–313, 2003.

[54] L. Petzold, S. Li, Y. Cao, and R. Serban. Sensitivity analysis of differential-algebraic equations and partial differential equations. *Computers and Chemical Engineering*, 30:1553–1559, 2006.

[55] T. Pi, Y. Zhang, and L. Chen. First order sensitivity analysis of flexible multibody systems using the absolute nodal coordinate formulation. *Multibody System Dynamics*, 27(2):153–171, 2012.

[56] G. P. Rao and H. Unbehauen. Identification of continuous-time systems. *IEE Proceedings – Control Theory and Applications*, 153:185–220, 2006.

[57] A. S. Schaffer. *On the adjoint formulation of design sensitivity analysis of multibody dynamics.* PhD thesis, University of Iowa, 2005.

[58] W. Schiehlen. Multibody system dynamics: Roots and perspectives. *Multibody System Dynamics*, 1:149–188, 1997.

[59] A. Shabana. Definition of the slopes and the finite element absolute nodal coordinate formulation. *Multibody System Dynamics*, 1(3):339–348, 1997.

[60] A. Shabana. *Dynamics of Multibody Systems.* Cambridge University Press, 2010.

[61] D. F. Shanno. Conditioning of quasi-newton methods for function minimization. *Mathematics of Computation*, 24/111:647–656, 1970.

[62] W. Steiner and S. Reichl. A contribution to inverse dynamic problems in multibody systems. In *Multibody Dynamics 2011, ECCOMAS Thematic Conference*, 2011.

[63] W. Steiner and S. Reichl. The Optimal control approach to dynamical inverse problems. *Journal of Dynamical Systems, Measurement and Control*, 134(2):1–13, 2012. 2011.

[64] E. Sueli and F. Mayers. *An Introduction to Numerical Analysis.* Cambridge University Press, 2003.

[65] T. Taylor, F. Palacios, K. Duraisamy, and J. Alonso. A hybrid adjoint approach applied to turbulent flow simulations. *In 21st AIAA Computational Fluid Dynamics Conference, San Diego, 24-27 June 2013*, -:1–26, 2013.

Stefan Oberpeilsteiner

Linke Brückenstraße 16
4040 Linz
+43 (0) 664-5440795
stefan.oberpeilsteiner@fh-wels.at

Personal data

Date of birth	13.05.1989
Place of birth	Rohrbach, Austria
Nationality	Austria
Work adress	University of Applied Sciences Upper Austria, Stelzhamerstrasse 23, 4600 Wels, Austria
Phone	+43 (0) 50804-44499
Fax	+43 (0) 50804-944499

Work experience

Current position

Since 06.2016 **Research fellow**, *Upper Austrian University of Applied Sciences - Research and Development GmbH*, Wels, Austria.
Engaged in research towards the adjoint method in multibody dynamics in the multibody system research group

Previous position

07.2013–05.2015 **Research fellow**, *Upper Austrian University of Applied Sciences - Research and Development GmbH*, Wels, Austria.
Engaged in research project *"Framework und Frontend für teilautomatisierten Abgleich von realen und virtuellen Prototypen - Protoframe"*

02.2013–06.2013 **Research fellow**, *Upper Austrian University of Applied Sciences - Research and Development GmbH*, Wels, Austria.
Engaged in research project *"3D Simulation der Schadensakkumulation richtungsabhängiger faserverstärkter Kunststoffe mittels Computertomografie - 3D-SFC"*

07.2012–09.2012 **Internship as design engineer**, *BMW Motoren GmbH*, Steyr, Austria.
Engaged in development of serial diesel combustion engines

04.2011–02.2012 **Internship as simulation engineer**, *Magna Powertrain - Engineering Center Steyr*, St. Valentin, Austria.
Engaged in various multibody simulation projects

Education

2013–today **Doctoral Programme in Technical Sciences (Dr. techn.)**, *Vienna University of Technology*, Institute of Mechanics and Mechatronics, Vienna, Austria.
PhD-thesis on parameter identification and optimal input in multibody dynamics

2011–2013 **Master's Degree Program (Dipl.-Ing.)**, *Mechanical Engineering*, University of Applied Sciences Upper Austria, Campus Wels.

Title of Diploma Thesis: Mikromechanische Modellierung kurzfaserverstärkter Materialien mittels CT-basierender RVEs

2008–2011 **Bachelors's Degree Program (BSc.)**, *Mechanical Engineering*, University of Applied Sciences Upper Austria, Campus Wels.

Title of Bachelor Thesis 1: Engine Test Bench

Title of Bachelor Thesis 2: Gesamtfahrzeugsimulation mittels MKS: Abgleich eines Simulationsmodells mit Messdaten aus dem Fahrversuch

2003–2008 **Automation Engineering**, Höhere Technische Bundeslehranstalt, Neufelden.

Awards

2018 **Innovation Award**, *Research Fellow of the Year*.

2014 **MEC 2014**, *Best Diploma Thesis*, *Mikromechanische Modellierung kurzfaserverstärkter Materialien mittels CT-basierender RVEs*.

2014 **Innovation Award**, *Diploma Thesis*, *Mikromechanische Modellierung kurzfaserverstärkter Materialien mittels CT-basierender RVEs*.

2009 **Jugend Innovativ**, *Diploma Thesis (HTL)*, Inspection Robot for Paper Machines.

Knowledge of programming

LaTeX, C++, Matlab, Scilab, Mathematica

Teaching activities

Since 2014 **Exercise**, *Mathematics, Dynamics, Multibody Simulation*, Upper Austrian University of Applied Sciences - Campus Wels.

List of publication

Journal publication

[1] **Oberpeilsteiner**, S., Lauss, T., Steiner, W., und Nachbagauer, K. A Frequency Domain Approach for Parameter Identification in Multibody Dynamics. *Multibody System Dynamics*, 1–18, 2017.

[2] **Oberpeilsteiner**, S., Lauss, T., Nachbagauer, K., und Steiner, W. Optimal Input Design for Multibody Systems by Using an Extended Adjoint Approach. *Multibody System Dynamics*, 1–12, 2016.

[3] Lauss, T., **Oberpeilsteiner**, S., Steiner, W., und Nachbagauer, K. The Discrete Adjoint Gradient Computation for Optimization Problems in Multibody Dynamics. *Journal of Computational and Nonlinear Dynamics*, 12(3):1–16, 2016.

[4] Nachbagauer, K., **Oberpeilsteiner**, S., und Steiner, W. Enhancement of the Adjoint Method by Error Control of Accelerations for Parameter Identification in Multibody Dynamics. *Universal Journal of Control and Automation*, 3:47–52, 2015.

[5] Nachbagauer, K., **Oberpeilsteiner**, S., und Steiner, W. The Use of the Adjoint Method for Solving Typical Optimization Problems in Multibody Dynamics. *Journal of Computational and Nonlinear Dynamics*, 10:1–19, 2014.

Conference contribution

[1] T. Lauss, P. Leitner, S. **Oberpeilsteiner**, and W. Steiner. Energy Optimal Manipulation of an Industrial Robot. In *ECCOMAS*, Barcelona, Spain, July 2015.

[2] T. Lauss, P. Leitner, S. **Oberpeilsteiner**, and W. Steiner. Energy Optimal Control of an Industrial Robot by using the Adjoint Method. In *OAGM & ARW Joint Workshop on Computer, Vision and Robotics*, Wels, Austria, May 2016.

[3] T. Lauss, S. **Oberpeilsteiner**, W. Steiner, and K. Nachbagauer. The Discrete Adjoint Method for Multibody Systems. In *IMSD*, Montréal, Canada, June 2016.

[4] T. Lauss, S. **Oberpeilsteiner**, W. Steiner, and W. Witteveen. Parameteridentifikation zur Bedatung von Modellen und optimale Steuerung mit der adjungierten Methode. In *Tagungsband NAFEMS DACH Regionalkonferenz*, 2016.

[5] K. Nachbagauer, S. **Oberpeilsteiner**, and W. Steiner. The Use of the Adjoint Method for Solving Typical Optimization Problems in Multibody Dynamics. In *IMSD*, 2014.

[6] S. **Oberpeilsteiner**, T. Lauss, W. Steiner, and K. Nachbagauer. Optimal Input Design for Multibody Systems. In *IMSD*, Montréal, Canada, June 2016.

[7] S. **Oberpeilsteiner**, T. Lauss, W. Steiner, K. Nachbagauer, and S. Reichl. Parameter Identification of a Torsional Vibration Damper in Frequenccy Domain Using Adjoint Fourier Coefficients. In *Conference on Dynamic Simulation in Vehicle Engineering*, 2018.

[8] S. **Oberpeilsteiner** and W. Steiner. Evaluation of the adjoint sensitivity analysis for the identification of multibody system parameters. In *GAMM 2014*, 2014.

[9] S. **Oberpeilsteiner**, W. Steiner, T. Lauss, and S. Reichl. Optimization in Multibody Dynamics. In *Conference on Dynamic Simulation in Vehicle Engineering*, 2016.

Linz, 20. Juli 2018