ÓBUDA UNIVERSITY

A Thesis submitted for the degree of Doctor of Philosophy

A STUDY ON USING FIXED POINT TRANSFORMATION IN ADAPTIVE TECHNIQUES IN ROBOTICS AND NON–LINEAR CONTROL

Hamza Khan

Supervisor
Prof. Dr. habil. József K. Tar D.Sc.

Doctoral School of Applied Informatics and Applied Mathematics

August 19, 2020

Members of the Comprehensive Examination Committee:

Members of the Defense Committee:
Chairman:
Prof. Dr. Péter Nagy D.Sc., professor emeritus, Óbuda University

Secretary:
Dr. Adrienn Dineva Ph.D. , Óbuda Universiy

Opponents:

Prof. Dr. János Abonyi D.Sc., University of Pannonia

Dr. Krisztián Kósi Ph.D., Óbuda Universiy

Further members of the Defense Committee:

Prof. Dr. László Szeidl D.Sc. professor emeritus, Óbuda University

Prof. Dr. Róbert Fullér D.Sc., Széchenyi István University

Dr. György Cserey Ph.D., Pázmány Péter Catholic University

Date of the defense:

Declaration of Authorship

I, *Hamza Khan*, declare that the thesis entitled, *"A Study on Using Fixed Point Transformation in Adaptive Techniques in Robotics and Non-linear Control"* and the content presented in it, is entirely my own research work in the able supervision of *Prof. Dr. habil. József K. Tar. D.Sc.* It is hereby assured that:

- This work has been completed as a candidate for Ph.D. degree at *Óbuda University*.
- Wherever in the thesis, the published work of other authors is consulted, this work is clearly referenced.
- Wherever in the thesis, quotations from other authors are used, then this fact is clearly mentioned. The thesis is absolutely my own work with the exception of such quotations.
- All the main sources of help have been properly acknowledged.

Budapest, 4 May, 2020.

Hamza Khan (Ph.D Candidate)

Dedication

To my dearest Father who is my mentor, leader and counselor, guided me in my childhood, encouraged and assured to be alongside of me every time. To my beloved Mother, her support, all time love and prayers provided me enough potentials in every step of life. To my Spouse, she sacrificed in the period of my Ph.D. studies and patiently faced all hardships with courage and look after my beloved Muhammad Muzammil Hamza and Muhammad Hashir Hamza.

Acknowledgement

First of all, I am very thankful to *Allah Almighty* who helped me and provided the potentials to bear the hardships patiently and to be determinant in the complexities of life.

Secondly, my warmest gratitude goes to my respected supervisor *Prof. Dr. habil. József K. Tar D.Sc* his all–time help, guidance, encouragement, and full support with his immense research skills pushed me up to this position where I am today and able to write this thesis. I am forever indebted to my supervisor, without his kind support it was impossible. He used to stand with me every–time and provided tremendous knowledge using his all kind of expertise and research skills to make me able to complete my Doctorate Degree. Thank you, Sir.

I am also very thankful to the *Doctoral School of Applied Informatics and Applied Mathematics, Óbuda University Budapest, Hungary*, and all the faculty members on their full support in the past 4 years' study period. Also, my warmest gratitude goes to the *Tempus Public Foundation* on providing all kinds of facilities during these years through the *Stipendium Hungaricum Scholarship & Dissertation Scholarship Programs*.

My Special Thanks goes to my parents, teachers, friends, siblings, and all relatives who supported by any means. Especially I express my thanks to the *International Office Mobility Department* and its members, *Obuda Student Hotel's Authority* to assisting all possible facilities during my time in Hungary, Europe.

Last but not least, my gratitude goes to the *Hungarian Communities*, during my stay they cooperated where I faced difficulties, their all–time cooperation will always be remembered.

List of Abbreviations

AF Auxiliary Function.

AMPC Adaptive Model Predictive Control.

ARHC Adaptive Receding Horizon Controller.

ASLRC Adaptive Slotine–Li Robot Controller.

CFAS Closed Form Analytical Solution.

CTC Computed Torque Control.

DoF Degree of Freedom.

DP Dynamic Programming.

FPI Fixed Point Iteration.

FPT Fixed Point Transformation.

GRG Generalized Reduced Gradient.

HOSVD Higher Order Singular Value Decomposition.

LHS Left Hand Side.

LMI Linear Matrix Inequality.

LPV Linear Parameter–varying.

LQR Linear Quadratic Regulator.

LTI Linear Time–invariant.

MPC Model Prdeictive Controller.

MRAC Model Reference Adaptive Control.

NFPT New Fix Point Transformation.

NP Non–linear Programming.

OC Optimal Controller.

PD Proportional and Derivative.

PID Proportional, Integral and Derivative.

PLC Programmable Logical Controller.

RFPT Robust Fixed Point Transformation.

RGM Reduced Gradient Method.

RHC Receding Horizon Controller.

RHS Right Hand Side.

SLARC Slotine–Li Adaptive Robot Controller.

T1DM Type 1 Diabetes Mellitus.

TP Tensor Product.

VBA Visual basic for Applications.

VS/SM Variable Structure – Sliding Mode Control.

Contents

Chapter 1

Introduction

The theory of controlling non–linear systems extensively was developed and used in the mid of the 20^{th} century and got more popularity with the passage of time. A revolutionary change was seen after the invention of computer, that made this field easier especially when Rudolf Kálmán placed into the center of attention the state–space model formulated in the time domain instead of the frequency picture that was prevailing before the early sixties of the 20^{th} century [1, 2].

In the most of the application areas within the frames of "Model Predictive Control" (MPC) [3, 4] the controlled system's dynamics (as a rigorous condition) mathematically is expressed as a "constraint" that has to be met while a "cost function" often representing contradictory requirements can be minimized with compromises. Among these compromises the limitation of the control force (caused by either the saturation of the drives or other reasons) can be taken into account. A possibility for tackling this problem is the method of "Dynamic Programming" (DP) that is based on the variation calculus and the resulting Hamilton–Jacobi–Bellman equation [5, 6].

To evade the huge computational needs of the dynamic programming approach, in tackling the problems in the field of control the so–called non–linear programming approach can be applied. The heuristic "Receding Horizon Controllers" (RHC) that were introduced for industrial use in the seventies of the past century [7] approximate the (MPC) over a finite time–horizon by the use of an available approximate dynamic model only, and for the compensation of the consequences of the modeling imprecisions and unknown external disturbances, the controlled system's state is directly observed or estimated by the use of observable data in the last point of the horizon that can be used as a starting point of the next one. Normally, the finite horizon is approximated by a discrete time–grid, and nonlinear programming is used for the calculation of the solution in which the solution is computed by the use of Lagrange's "Generalized Reduced Gradient" (GRG) method published in 1811 [8].

Usually, replacing a non–linear system by a simple linear one is considered as a simple and effortless way to find the approximate solutions whenever it is satisfactory to consider the system's operation in the vicinity of a "working point". In this narrow vicinity the system can be approximated as a "Linear Time–Invariant" (LTI) one. Sometimes this approximation well captures the system's dynamics e.g. in modeling the behavior of the air path of Diesel engines [9]. In the close vicinity of a "working point", the traditional control design approaches that were based on the properties of the "Linear Time–Invariant" (LTI) systems by using the frequency domain, linear integral transformations as the Fourier, Z, or Laplace transforms [10], can lead to satisfactory results.

However, in many cases linearization (called "affine approximation") cannot be a relevant and satisfactory way in approximating the controlled system's dynamical model. Often a "system switching" has to be tackled between the neighboring cells of the state space that contain the local LTI models [11]. This approach has been extended in the tracking control for switched linear systems with time–varying delays [12, 13].

In other approaches the so called "Tensor Product Control Models" (TP) can be used based on polytopic dynamic models, "Higher Order Singular Value Decomposition" (HOSVD), and "Linear Matrix Inequalities" (LMI) as e.g. in [14, 15, 16]. This branch of research typically is based on the possession of a reliable complex system model that in the first phase is transformed into a TP form "offline" by the use of a computer. In the second step the "unnecessary complexity" of this transformed model is reduced by the application of HOSVD. Finally, according to the program announced by Boyd et al. in 1994 in [17], the so obtained model's control can be systematically tackled by existing software algorithms based on the use of LMIs.

It has to be emphasized that though the "heuristic RHC" provides a quite wide framework for control approaches, it suffers from some limitations that seem to be crucially significant in engineering applications. In the case of using general forms for the cost functions and allowing the use of arbitrary nonlinear models no rigorous statements can be done on the nature of the obtained solutions. Due to the fact that the number of the Lagrange multipliers equals with that of the state variables in the constraint terms, these state variables and their associated Lagrange multipliers behave like the canonical variable pairs of Classical Mechanics [18]. Consequently an "artificial Hamiltonian" can be associated with the realized “optimal motion”, and this Hamiltonian is kept constant. Furthermore, in the tangent space of these “canonical state variables” the rules of the Symplectic Geometry are valid that means that the “volume” of the phase cells remains constant (Liouville’s Theorem in Classical Mechanics), i.e. the motion is similar to the flow of some incompressible fluid (that also satisfies complementary conditions regarding the partial derivatives according to the control forces). In general, “incompressibility” does not promise very nice numerical behavior for the solutions. If in certain direction the cells are shrunk, in other directions they have to be extended to save their volume. This concerns stability issues: it cannot be expected that the solution can be settled in an attractive point of the state space. On this reason in the practice the otherwise quite wide frames of the RHC controllers are applied under strictly “narrow” conditions as follows:

a) Normally the cost functions are quadratic terms constructed of constant symmetric positive definite matrices and the state tracking errors. Consequently, their derivatives in the reduced gradient method will be well behaving linear functions of the tracking errors.

b) Generally similar quadratic terms are in use for the limitation of the control forces that provides similar advantages.

c) In the case of LTI system models the Lagrange multipliers can be constructed as the product of some symmetric matrices and the state variables. The equations of motion for these matrices can be decoupled from that of the state variables, that is a great advantage.

d) The equations of motion obtained for these matrices satisfy some Riccati equation with some “terminal condition”. In the 18$^{\text{th}}$ century Riccati realized that special first order quadratic differential equations can be solved by obtaining the solution for linear, second order differential equations [19]. Therefore, under these special conditions certain “general view” of the solution became available. The matrix versions of the Riccati equations obtained wide scale use in control technology (e.g. [20, 21]).

e) Regarding certain constraints, Schur’s matrix complement [22] can be applied to transform quadratic constraints into linear ones that can be efficiently tackled by the LMI techniques as it was recommended by Boyd et al. in [17].

One of my main aims was to “liberate” the researchers from the above restrictions in the RHC control by the application of non–quadratic cost functions that for instance can "better tolerate" smaller errors and "more drastically punish" greater ones than a simple quadratic structure.

It was considered an open question since a few years ago that the combination of the MPC within the frame of "Optimal Controllers" with some adaptive techniques can be possible in the area of control theory.

In the field of control theory to design non–linear adaptive controllers, generally Lyapunov's well known 2^{nd} or "Direct Method" [23, 24] is used. It is widely applicable in recent days, too. But its typically complex design process is considered as burden and difficult, therefore alternative simple methods were adopted. It has the great advantage that it provides a basis to design traditional and classical non–linear controllers. The purpose is to concentrate on the problem of the stability of the motion of the controlled non–linear system. Though, this has great advantages in the related field but despite those excellent advantages, the Lyapunov function–based approach suffers from certain disadvantages, too. Beside the fact that this method needs tricky and complex mathematical designer's skills, it typically prescribes rather "satisfactory" than "necessary and satisfactory" conditions in the proofs. Consequently it allows the application of a huge set of possible control parameters that maintain the stability and in the same time crucially influence the transients of the controlled motion without any optimization.

To get rid of the very complicated Lyapunov function–based adaptive solutions in 2009 an alternative approach "Fixed Point Transformation" (FPT) was proposed [25] where, the problem at first was transformed into the fixed point task and then the idea of iteratively finding the fixed–point of a contractive map was used. This idea is based on Banach's well known Fixed Point Theorem [26]. In my research the said idea was further extended regarding new aspects where the approach was combined with other methods and also was applied to get rid of the complicated burden of the precise calculation of the Jacobian in the inverse kinematics of robots. The purpose, also, is to give some contributions and achievements in this new line of problem tackling in adaptive control. I am especially interested in the possible combination of the traditional and the novel approaches.

1.1 Research Aims in the Mirror of the State of the Art

This is the era of modern sciences and technologies. Things and technologies continuously keep changing due to new ideas and up–to–date technological instruments. Such ideas and the advanced technological revolution bring severe changes in several natural systems in the Universe. Abundant of systems are there in the Universe, based on non–linear functional dependencies. Their non–linearity was always considered a great challenging subject for the researchers in view of their stability and efficient control. The control of such systems, by numerous techniques, fall in the area of study named "Control Theory".

To deal with such systems only a few methods were used before the last decade of the 19^{th} century but later in 1892 Alexander Lyapunov elaborated his way of solution in his doctoral dissertation to deal with the stability of the systems giving his theory with the approach named as "Lyapunov's Direct or Second Method" to determine the stability of a non–linear system without solving its equations of motion.

It is evident that, a control designer tries to bring about better and efficient methods to maintain the stability of the controlled systems. In the beginning, getting the solutions of the problems, based on non–linearity, were very hard due to the fact that only "manual working system" (consisting of crank driven mechanical calculators, slide–slip, metric paper and the tabulated form of certain special functions) was available, but later the invention of the computers provided an easy way to proceed in this area of study to extend it and widen its view from different aspects.

To understand the working process and criteria of the systems, consideration of their modeling and controlling process, measuring or estimating the states of the systems efficiently have become the prime need of the time. For the purpose of controlling the systems and to understand their stability, different varieties of methods and terminologies have been used in different times to enrich the stabilities.

Adaptive control is one of the methods where a system uses the techniques and approaches to change itself according to the behavior in new or varying circumstances. The motivation to consider this area of study was gotten in early fifties of the previous century when an autopilot high performance based aircraft for high altitudes and wide range of speed was designed. After this approach the study area got more attention in all aspects of life. Examples for a quite rich variety of problems in practical life can be mentioned in this context: the glucose–insulin metabolism [27, 28, 29, 30, 31, 32], the pharmaco–kinetics of various drugs in anaesthesia [33, 34, 35, 36], modeling the operation of the neurons and the nervous system [37, 38, 39, 40, 41] in life sciences, dynamic models of robots [42, 43], chemical processes like crystallization [44], efficient control of freeway traffic [45, 46, 47] including the limitation of the emission of polluting materials [48, 49], etc. can emphasize its importance and applicability.

The study of adaptive techniques for non–linear systems has considerable mathematical difficulties. Analyzing them theoretically is, in fact, a very complex and hard task. Therefore, the modern techniques and approaches in view of approximations in control design and signal processing include a various class of mathematical tools.

In the last decade of the 20$^{\text{th}}$ century the idea of MPC was vastly investigated (e.g. [50, 51]), and its novel developments (e.g. [4]) were successfully used in different fields of the life as e.g. in chemistry [52, 53, 54, 55], life sciences–related problems [56], economy [57], etc. Another use of advanced control solutions is, to get attention in today's medical practice regarding the control of physiological processes [58]. Many control solutions are under development which can be used for various kinds of control problems. It has been observed that there are many advanced control methods that have been successfully applied for physiological regulation problems, for example control of anaesthesia [59, 60], antiangiogenic inhibition of cancer [61, 62], immune response in presence of human immunodeficiency virus[63] and regulation of blood glucose (BG) level [64, 65, 66, 67] as well.

In the applications the non–linear nature of the advanced control techniques have high importance. Beside the non–linearities in the control problems the researchers on the field are facing with many challenges such as model and parameter uncertainties and even time–delay effects, too.

It is well known that in designing the adaptive controllers, based on the non–linear systems mostly Lyapunov's "direct" or "second method" is applied as a traditional approach [23, 24]. Essentially the same approach is extended to tackling time–delay problems by the use of the Lyapunov-Krasowskii functional [68]. The complexity of this method diverted the attention of researchers to propose the alternative simple approaches (e.g. [69, 70, 71, 72]). According to the basic facts the work of Lyapunov's method can be summarized as follows [73]:

a) it can be used to create the satisfactory conditions to guarantee the stability,

b) it does not focus on the tracking error relaxation in the initial phase of the controlled motion, but provides the opportunity to prove the global stability that is very necessary in common cases,

c) in the case of certain adaptive approaches for the identification of the parameters of the model of the controlled system, it provides significant methods,

d) it works with a large number of arbitrary adaptive control parameters because it contains certain components of the particular Lyapunov function in use, and may require further parameter optimization (e.g. [74]).

It is realized that the mathematical framework of the traditional MPC can hardly be combined with the Lyapunov function–based adaptive control. Certain approaches combining MPC and Lyapunov's stability theorem can be found in the literature (e.g. [75, 76]).

Concentrating on the primary design intent the "Robust Fixed Point Transformation" (RFPT)–based technique was suggested in which the non–linearly optimized trajectory can be adaptively tracked iteratively by the adaptive controller that converges to the appropriate point, based on Banach's Fixed Point Theorem [26]. Furthermore, the suggested "adaptive, iterative inverse kinematic approach" [77] – based on [78, 79] – can be convergent and useful even if the Jacobian of the robot arm is only approximately known. The application of an "abstract" rotational transformation in the state space can improve the convergence properties of the iteration without the need for obtaining complete information on the actual (i.e. the "exact") Jacobian. It is just enough to utilize the simple motion steps generated by the iteration that produces a smooth motion.

Similarly, a possible recent improvement of the RHC approach was reported in [80] that corresponds to the adaptive tracking of the optimized trajectory instead of exerting the forces calculated by the optimization algorithm on the basis of an available, approximate dynamic model.

All the above discussed results are introduced in our papers published recently (e.g. [80, 81, 82, 83], [84], [85], and [86], [87]). The main directions for this research will be outlined in the next section.

1.2 The Structure of the Thesis

The possible application and possible improvement of the Classical "Receding Horizon Controller" (RHC) and its application has been discussed in Chapter 2 in details.

In the initial stage the possibilities and useful applications of the design of the Classical RHC by using Non–linear Programming (NP) was investigated. After that, using the Generalized Reduced Gradient method, to overcome the complexities of non–linear optimization task the study was further extended to deal with the Type 1 Diabetes Mellitus, one of the dangerous disease for humanity, that needs treatment in the form of control. To check the performance of the solution "soft" disturbance and "unfavorable" disturbance signals were applied and studied.

To improve and overcome the disadvantages and difficulties in the field of non–linear control, a novel "Fixed Point Transformation" (FPT) recently invented by A. Dineva in [88, 89] was applied by combining it with the classical RHC. The alternative FPT approach was so introduced in adaptive control that, instead of Lyapunov's "2$^{\text{nd}}$ Method", it was based on Stefan Banach's "Fixed Point Theorem". This method was applied in various aspects to achieve precise and meaningful results. The combination of this method with the RHC is based on the idea that in the classical optimal control both the state variables and the control signals are estimated for the horizon based on the approximate dynamic system model. However, instead of exerting the so obtained control signals, the system adaptively tries to track this "optimized trajectory". For making simulation examples a simple 1$^{\text{st}}$ order LTI system was investigated in the adaptive control.

In Chapter 3 the adaptive RHC is explained for some special problem classes that are directly treatable by the use of the "Auxiliary Function" (AF).

In Chapter 4 the "Inverse Kinematic Task of Robots" is adaptively solved using the FPT method. It has been observed that in a wide class of robots of open kinematic chain the inverse kinematic task cannot be solved by the use of closed–form analytical formulae. On this reason the traditional approaches apply differential approximation in which the Jacobian of the "normally redundant" robot arm is "inverted" by the use of some "generalized inverse". These pseudo–inverses behave well whenever the robot arm is far from a singular configuration. However, in the singularities and nearby the singular configurations they suffer from a singular or ill–conditioned pseudo–inverse. For tackling the problem of singularities normally complementary "tricks" have to be used that so "deform" the original problem that the deformed version leads to the inversion of a well–conditioned matrix. Similarly, in the inverse kinematic tasks in robots the burden of the typical computations of Jacobian were removed by using the FPT via JULIA and MATLAB simulations.

Chapter 5 is the "conclusion chapter" where all the related novelties of this thesis will be discussed briefly. Finally, in Chapter 6 the study is discussed in future prospects that what results and novelties are being expected? Also, how we will be able to achieve reliable results according to the topic of the thesis?

1.3 Research Methodology

It is obvious that the computational mathematical problems and engineering topics need simulation–based studies to understand and further extend the existing solutions. The wide range of practical problems results in differential equations that cannot be solved analytically, can be studied via numerical methods using simulations and programming. Such problems can be applied and understood after the validation of the simulation investigations. For this purpose a lot of mathematical packages can help to find the clear results.

In the period of my study, I used the JULIA Programming Language for the programming purposes to find the required results of my research. It provides an easy, simply coded and fast running programming possibility. Similarly, the simple VISUAL BASIC of MS–EXCEL 2010 was also used that helped to easily compile the results by simple programming. In some cases I used Matlab 2018, too, to find the results easily. Matlab is a heavy but precise programming language which helped in rare cases to easily go through from my work.

The JULIA Package developed under the *MIT* and *GPLv2* license works by providing a modeling and computational interface for solving the dynamical problems. Julia is a high–level, high–performance dynamic programming language developed specifically for scientific computing [90]. This dynamic language ensures a very fast evaluation for technical computing. The applied scientific methods ensure the precision and thoroughness of the simulation results. It has an ability for distributed parallel task execution. It also has numerous developer communities who made some "extra" packages with the use of the build–in package manager. At its official website [90], the main features of the language are mentioned as

- Multiple dispatch: providing ability to define function behavior across many combinations of argument types
- Dynamic type system: types for documentation, optimization, and dispatch
- Good performance, approaching that of statically–typed languages like C
- A built–in package manager

- Lisp–like macros and other meta–programming facilities
- Call Python functions: use the PyCall package
- Call C functions directly: no wrappers or special APIs
- Powerful shell–like abilities to manage other processes
- Designed for parallel and distributed computing
- Coroutines: lightweight green threading
- User–defined types are as fast and compact as built–ins
- Automatic generation of efficient, specialized code for different argument types
- Elegant and extensible conversions and promotions for numeric and other types
- Efficient support for Unicode, including but not limited to UTF–8

For the research concerning optimization programs, a developed simple program in “Visual Basic for Applications” (VBA) in the background of the MS EXCEL serves as an excellent solution provided that the size of the problem is not too big. The MS EXCEL’s “Solver” module is provided by an external firm (Front–line Systems, Inc.). There are various solutions implemented into the Solver module, including the “Generalized Reduced Gradient” (GRG) method which can be used in the given case as well. The GRG is based on [91, 92] and its usability has been proved in various fields of research as e.g. in [83]. The problem conveniently can be formulated by functional relationships between the contents of the various cells of the worksheets. For this purpose User Defined Functions can be created in VBA. Then for the Solver a “model” can be specified by giving the cell that contains the cost to be minimized, the location of the independent variables and the constraints in the worksheets, and the parameter settings of this optimization package. The so defined “model” can be saved somewhere in one of the worksheets. Following that a small program can be written in VBA that declares the model parameters as global variables, reads their actual values from the worksheets, loads the “model” for the Solver, and for the horizons under consideration cyclically:

a) fills in the cells with the data of the nominal trajectory to be tracked, the initial values of the variables to be optimized, and the control forces,

b) calls the Solver with the options that it must stop optimization if the prescribed limits in the time or step numbers have been achieved, keeps the so obtained results, and

c) writes the optimized results in certain cells of a worksheet that are used for making various graphical representations of the results.

Regarding the "reliability" of the computed data, because we investigated "convergent iterations" in the simulation programs, the key factor was the time resolution of discretization. After obtaining certain numerical results with a given fixed time–step Δt, the same computations were repeated with $0.5 \times \Delta t$. If no observable differences in the results were observed, the computation was regarded as "reliable result". Roughly speaking, this practice corresponds to stopping an iteration when the subsequent variations in a convergent Cauchy sequence become small enough. In the practice such solutions are prevailing.

In the Julia realizations I applied the simple Euler integration in sequential programs. This structure was treatable in this manner. In the case of optimization programs, the role of Δt corresponded to the discrete time resolution in the case of Non–linear Programming. These problems were also treatable in this manner. I have to note that the problems in the convergence were influenced by other parameters in the Solver's setting as e.g. the maximum number of numerical steps or the maximal time allowed for finding the optimum. The effects of these parameters were separately considered in the programs realized by the EXCEL–VBA–Solver combination.

Chapter 2

Improvement of the Classical Receding Horizon Controller and Its Applications

In industries, for more advanced controlling purposes, the vastly applicable heuristic Receding Horizon Controller (RHC) invented in 1978 [7] can be used. By using the best possibilities of the RHC we are able to predict the possible future behavior of our system, moreover, we are able to intervene in its operation as well. I have investigated the possibilities of the design of a Receding Horizon Controller by using Non–linear Programming to simulate a possible treating for patients suffering from Type 1 Diabetes Mellitus [A. 1]. The non–linear optimization task was solved by the Generalized Reduced Gradient (GRG) method.

Diabetes Mellitus (DM) is the collective name of several chronic diseases connected to the metabolic system of the human body. There are many types of DM. The most dangerous is the Type 1 DM (T1DM) where the metabolic system is not able to function normally due to the lack of insulin. Type 2 DM (T2DM) which is the most widespread kind of DM and it occurs mostly because of the lifestyle. In this case the usual is that the blood glucose and insulin levels are continuously increasing over a long period of time. Due to the extreme glucose and insulin load, the cells are becoming resistant to the insulin over time. In order to compensate this condition the body produces more and more insulin – which leads to the "burnout" of the pancreatic β–cells which produce the hormone. At this point the T2DM turns into T1DM. Other frequently occurring type is the Gestational DM (GDM) from which women may suffering during pregnancy. Usually, this condition is temporary, however, sometimes it turns into T2DM and becomes permanent [93, 94, 95].

The investigations were done on the basis of a particular system model, the "Modified Minimal Model" [96] which originates from the model of Bergman [97]. Two practically important scenarios were investigated. In the first one, I applied "soft" disturbance – namely, smaller amount of external carbohydrate – in order to be sure that the proposed method operates well through the optimization process. In the second scenario, "unfavorable" disturbance signals were applied – a highly oscillating, peak kind external excitation with cyclic nature.

I have found that the performance of the realized controller was satisfactory and it was able to keep the blood glucose level in the desired healthy range – by considering the restrictions against the applicable control action.

In the 2nd step, as a preliminary investigation, I studied the possibility for the adaptive extension of the RHC method in combination with the Fixed Point Transformation–based approach [A. 2]. In this case the simplest 1st order 1 DoF LTI system model was used. The combination of this method with the Receding Horizon Control is based on the idea that in the classical optimal control both the state variables and the control signals are estimated for the horizon based on the approximate model using Lagrange's "Reduced Gradient Method" (RGM).

It provides the "estimated optimal control signals" and the "estimated optimal state variables" over this horizon. The controller exerts the estimated control signals but the state variables develop according to the exact dynamics of the system.

I have used the suggested alternative approach in which, instead of exerting the estimated control signals, the estimated optimized trajectory is adaptively tracked within the given horizon. Simulation investigations are presented for a simple LTI model with strongly non–linear cost and terminal cost functions. In this investigation I found that the transients of the adaptive controller that appear at the boundaries of the finite–length horizons reduce the available improvement in the tracking precision. In contrast to the traditional RHC, in which decreasing horizon length improves the tracking precision, in my case some increase in the horizon length improves the precision by giving the controller more time to compensate the effects of these transients.

2.1 Scientific Antecedents

Regardless of the type of DM a few common control goal can be defined: keep the glycemia (the BG level) in the healthy range; total avoidance of hypoglycaemic periods; and avoid the high BG variability as far as possible [98, 99, 100].

In case of T1DM many solutions are available, however, all of them have their own limitations, simplifications and restrictions – thus, none of them are general [67]. In these days from control point of view the most beneficial approach is the Artificial Pancreas (AP) concept. This idea aims to imitate the regular working of the pancreas from the insulin production point of view, namely, administering insulin demands on the needs determined by the BG level [101]. Thus, we have to face with contradictory requirements: the generalization and personalization as well.

One of the mostly used algorithms is the modified proportional–integral–derivative (PID) based solutions due to their simplicity and flexibility. Moreover, several clinical trials have been done by using these methodologies and investigated their effectivity [102, 103, 104]. Linear Parameter Varying (LPV) based solutions have high importance, since, the uncertainties can be handled with high efficiency by them [66, 64, 105]. The Tensor Product (TP) based techniques also represent interesting directions, since, they can be combined by Linear Matrix Inequality (LMI) based control and LPV methodology as well [106, 107, 65]. The most frequently used method is the Model Predictive Control (MPC) with regard to the control of DM [108, 101, 109].

By its applicability and persistency MPC approach is widely used in various research fields as well [44, 110, 57]. In general the goal of the MPC applications is the tracking and stabilization [111].

The RHC framework can be hardly combined with the Lyapunov function based control. However, certain approaches can be found in the literature where the Lyapunov stability and RHC was successfully combined for specific cases [75, 112]. Alternative solutions also exist which can be used instead of Lyapunov's stability theorem. The "Robust Fixed Point Transformation" (RFPT) based control [25, 113] uses Banach's fixed point theorem [26] to transform the control task into a fixed point problem which can be solved iteratively. This method allows to design a robust iterative adaptive controller which can avoid the main limitations of RHC if these are combined.

2.2 Investigation of the Applicability of RHC in Treating the Illness Type 1 Diabetes Mellitus

During my research I have applied a modified Minimal Model [96] which originates from the model of Bergman [97]. This model has several beneficial properties, such as simplicity, good transformability, flexibility and it is based on simpler biological considerations. The main goal of the model is to describe the glucose–insulin dynamics, namely, to define the connection between the blood glucose and insulin levels. Although, in order to characterize the daily life of a T1DM patient by the model, it is needed to extend it with additional sub–models.

These sub–models are the absorption of the external glucose and insulin intake. During the daily routine these substances are not directly injected to the blood stream – however, this can occur in case of persistent hospitalization – instead the carbohydrate is consumed via food intake and the insulin is entered through the extracellular tissue matrix under the skin [93]. Thus, the characteristic of their appearance in blood has rather peak kind than elongated dynamics. The glucose and insulin absorption is described by (2.1) – (2.4), respectively. These sub–models are coming from the Cambridge model [114], but I applied them in appropriate dimensions to insert it to the core model. The core model is described by (2.5) – (2.7).

$$\dot{D}_1(t) = -\frac{1}{\tau_D} D_1(t) + \frac{1000 A_g}{M_{wG} V_G} C \cdot d(t), \tag{2.1}$$

$$\dot{D}_2(t) = -\frac{1}{\tau_D} D_2(t) + \frac{1}{\tau_D} D_1(t), \tag{2.2}$$

$$\dot{S}_1(t) = -\frac{1}{\tau_S} S_1(t) + \frac{1}{V_I} u(t), \tag{2.3}$$

$$\dot{S}_2(t) = -\frac{1}{\tau_S} S_2(t) + \frac{1}{\tau_S} S_1(t), \tag{2.4}$$

$$\dot{G}(t) = -(p_1 + X(t))G(t) + p_1 G_B + \frac{1}{\tau_D} D_2(t), \tag{2.5}$$

$$\dot{X}(t) = -p_2 X(t) + p_3 (I(t) - I_B), \tag{2.6}$$

$$\dot{I}(t) = -n(I(t) - I_B) + \frac{1}{\tau_S} S_2(t), \tag{2.7}$$

The meaning and purpose of the state variables in (2.1) - (2.7) are as follows: $D_1(t)$ mg/dL and $D_2(t)$ mg/dL are the primary and secondary compartments belonging to glucose, where τ_D time constant determines how long it takes the meal is absorbed after consumption in time. $S_1(t)$ mU/L and $S_2(t)$ mU/L are the primary and secondary compartments belonging to insulin, where τ_S time constant determines how long it takes the insulin absorbed after injection (to the extracellular space) in time. The $G(t)$ mg/dL is the blood glucose (BG) concentration – the so–called glycemia –, $I(t)$ mU/L is the blood insulin concentration and $X(t)$ 1/min is the insulin-excitable tissue glucose uptake activity – which describes the connection between the blood's glucose and insulin levels, respectively.

From system engineering point of view the external glucose, namely, the food intake can be handled as disturbance. In this case $d(t)$ g/min is the disturbance input. It can be inserted to the $D_1(t)$ via the $\big((1000A_g)/(M_{wG}V_G)\big)C$ complex which describes the bio-availability of the glucose from complex carbohydrates. The $u(t)$ mU/min control signal – the injected insulin – is directly connected to $S_1(t)$. More detailed description of the used model parameters can be found in Table 2.1 and in [96, 114].

Table 2.1: The applied parameters of the models in this study [96, 114, 115].

Notation	**Value**	**Unit**	**Description**
G_B	110	mg/dL	Basal glucose level
I_B	1.5	mU/L	Basal insulin level
p_1	0.028	$1/min$	Transfer rate
p_2	0.025	$1/min$	Transfer rate
p_3	0.00013	$L/(mU\,min)$	Transfer rate
n	0.23	$1/min$	Time constant for insulin disappearance
BW	75	kg	Body weight
V_I	$0.12BW$	L	Distribution volume of insulin
V_G	$0.16BW$	L	Distribution volume of glucose
M_{wG}	180.1558	g/mol	Molecular weight of glucose
A_g	0.8	$-$	Glucose utilization
C	18.018	$mmol/L$	Conversion rate between $mmol/L$ and mg/dL
τ_D	40	min	Carbohydrate (CHO) to glucose absorption constant
τ_S	55	min	Insulin absorption constant

2.2.1 Simulation Results for Treating T1DM with RHC

During the development of the appropriate cost function – which fits to the given problem – the specificities of the model (2.1) – (2.7) should be taken into account. The main limitation coming from the model is the amount of injectable insulin and the fact that I do have only one control signal. Due to the applied control signal instances from the given horizon are independent variables in the optimization problem applying a specific form of limitation on them – a "bias" – is reasonable. Thus, the control signal in this construction should be limited in accordance with the phenomenon to be controlled. Further, not the control signal itself, but another variable should be selected as independent variable to avoid the initial value problems causing the rough numerical approximation at the beginning of the optimization. This is caused by the high non-linearity in the model.

Another property of the model that only the $G(t)$ blood glucose level can be measured. Thus, only this state variable can be embedded into the cost function to be developed. I do not have internal information about other state variables of the process to be controlled during action – which is the internal operations of a human being –, namely, this limitation has to be taken into account. Due to such reasons I have applied a more specific cost function defined in (2.8):

$$\begin{aligned} &\min_{\substack{\{\mathbf{x}_1,\dots,\mathbf{x}_F\}\\ \{\mathbf{u}_0\dots,\mathbf{u}_{F-1}\}}} \sum_{i=0}^{F-1} J(\mathbf{x}_i,\mathbf{u}_i) + \Phi(\mathbf{x}_F) \\ &\text{subject to } \frac{\mathbf{x}_{i+1}-\mathbf{x}_i}{\Delta t} - f(\mathbf{x}_i,\mathbf{u}_i) = \mathbf{0}, \end{aligned} \tag{2.8}$$

and $\{\lambda_0,\dots,\lambda_{F-1}\}$ are the Lagrangian multipliers – which are used in accordance with the optimization task to be solved by the reduced gradients method.

$$\begin{aligned} &\min_{\substack{\{G_1,\dots,G_F\}\\ \{v_0\dots,v_{F-1}\}}} \sum_{i=0}^{F-1} J(G_i,u_i) + \Phi(G_F) \\ &\text{subject to } \frac{\mathbf{x}_{i+1}-\mathbf{x}_i}{\Delta t} - f(\mathbf{x}_i,\mathbf{u}_i) = \mathbf{0} \end{aligned} , \tag{2.9}$$

where $u_i = u_{bias} + \tanh(v_i)$. We have developed a strongly non-linear cost function in which all requirements can be embedded against the control action to be reached during control.

$$J(G,u) \stackrel{def}{=} \left|\frac{G^N - G}{A_1}\right|^{\alpha_1} + B\left|\frac{u}{A_2}\right|^{\alpha_2} , \tag{2.10}$$

$$\Phi(G_F) \stackrel{def}{=} \left| \frac{G^N_{final} - G_{final}}{A_3} \right|^{\alpha_3} . \tag{2.11}$$

The tracking error in (2.10)–(2.11), namely, the deviation of the realized blood glucose level $G(t)$ from the nominal blood glucose level $G^N(t)$ can be calculated as $G_i^N - G_i$ at the grid points. The absolute value of this difference can be determined by A_1 and A_3 parameters which contribute the belonging level of "penalty" prevails in the cost function. In that case if $\alpha_1 > 1$ and $\alpha_3 > 1$ beside $|G^N - G| < A_1$ and $|G^N_{final} - G_{final}| < A_3$, then the contribution to the cost function is low. However, if $|G^N - G| > A_1$ and $|G^N_{final} - G_{final}| > A_3$ then due to the power terms the contribution to the cost of these terms are drastically increasing. The α_1 and α_3 constants can be used for different purposes. $\alpha_1 = \alpha_3 = 1$ provides proportional contribution, namely, the pure deviation will be better prohibited. Although, if $\alpha_1, \alpha_3 < 1$, then the smaller deviations will be prohibited better than the bigger ones. The role of the A_2 and α_2 parameters are similar, namely, the applied control input can be prohibited by applying them. The B parameter allows to modify the enforcement of the effect of the control signal in the cost function.

In order to test the realized control framework I investigated two scenarios. In the first scenario 25 g CHO was considered – 5 g over 5 minutes in each 240$^{\text{th}}$ time instance from the 60$^{\text{th}}$ time instance. In the second scenario we considered 50 g CHO intake – 10 g over 5 minutes in each 240$^{\text{th}}$ time instance from the 60$^{\text{th}}$ time instance. In both cases 200 time horizons have been considered within 10 grid points, thus the total simulated time domain was 2000 minutes. The resolution Δt was 1 minute in accordance to the properties of the model.

The applied cost function parameters (which represent the control parameters in this regard) can be found in Table 2.2. It should be noted that I have used permanent reference trajectories in both cases denoted by G^N = 90 mg/dL in (2.9)–(2.11).

Therefore, in accordance with the aforementioned details, the goal of the control becomes to keep the G^N = 90 beside respecting the predefined u_{bias}. In this manner – via the cost function – the deviations from these predefined values have been "punished", namely, the value of the cost function became higher.

Table 2.2: The parameters of the applied cost function (2.10).

	Scenario 1	Scenario 2
A_1	5 mg/dL	1 mg/dL
α_1	4	4
A_2	2 mU/min	15 mU/min
α_2	4	6
A_3	5 mg/dL	5 mg/dL
α_3	4	4
u_{bias}	10 mU/min	50 mU/min
G^N	90 mg/dL	90 mg/dL

2.2.1.1 Results of Scenario 1

In the followings the results of Scenario 1 are presented. First, the disturbance signal is shown by Left Hand Side (LHS) of Fig. 2.1. The applied – calculated – control signal can be seen on Right Hand Side (RHS) of Fig. 2.1. It is clear that the controller is able to administer the insulin in accordance to (2.9)–(2.11) where the $u_i = u_{bias} + \tanh(v_i)$ is prevailed in the control action.

Figures 2.2 show the absorption of glucose and its appearance in the blood with the dynamics determined by the model. Figures 2.3 show the absorption of insulin from the interstitium and its appearance in blood.

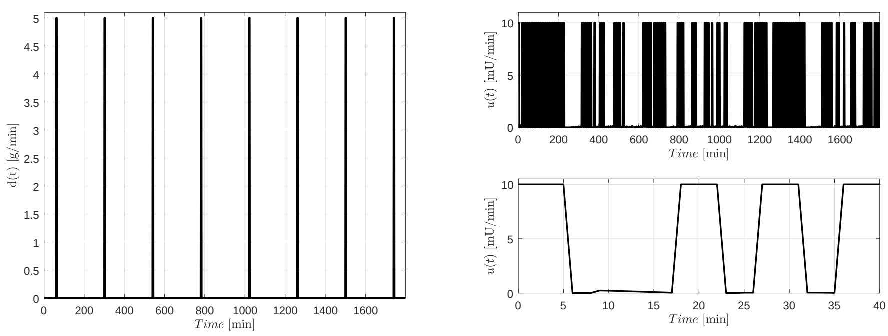

Figure 2.1: Applied disturbance (CHO intake) – 5 g over 5 minutes at each 240$^{\text{th}}$ time instance (LHS) and the calculated control signals (RHS). The upper figure represents the whole time horizon. The lower figure shows a piece of the whole time horizon between 0 and 40 minutes.

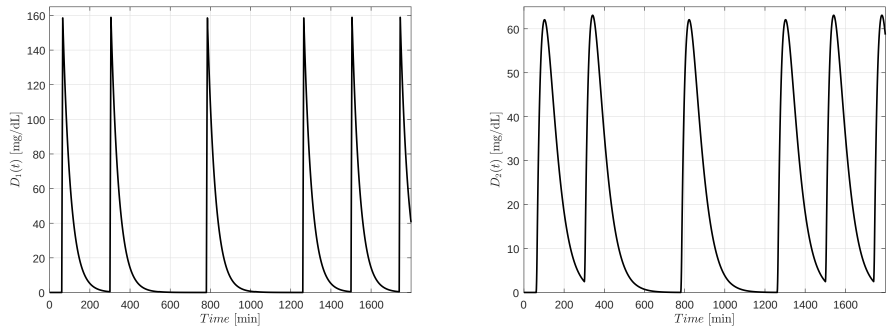

Figure 2.2: Variation of the first (LHS) and second (RHS) states of the glucose absorption subsystem.

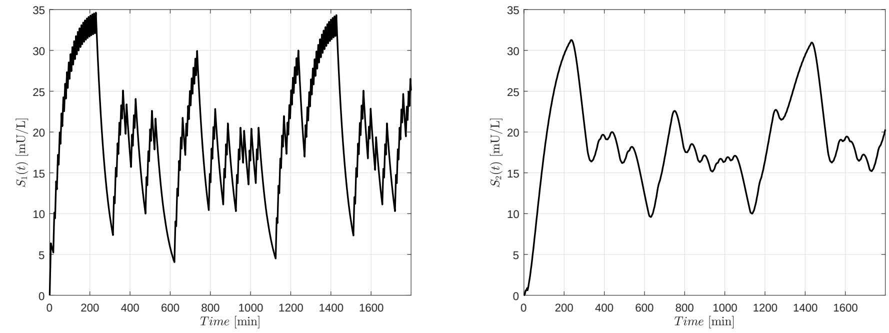

Figure 2.3: Variation of the first (LHS) and second (RHS) states of the insulin absorption subsystem.

The main point can be seen on Fig. 2.4. The controller is able to satisfy the determined conditions and the BG level ($G(t)$) is inside the predefined range – no hypo- and hyper-glycemia occurred. The BG level approaches the selected reference trajectory (G^N) as it is expected.

On Figs. 2.5 the the variation of blood insulin level and intermediate state variable can be seen. $X(t)$ determines how the blood insulin level affects the blood glucose level, namely, the connection between them.

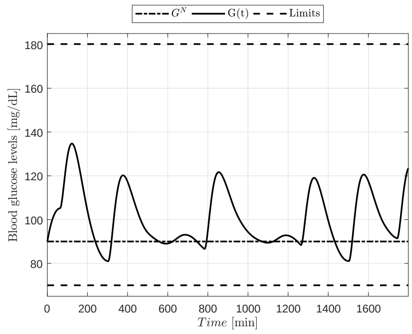

Figure 2.4: Variation of the blood glucose level over time

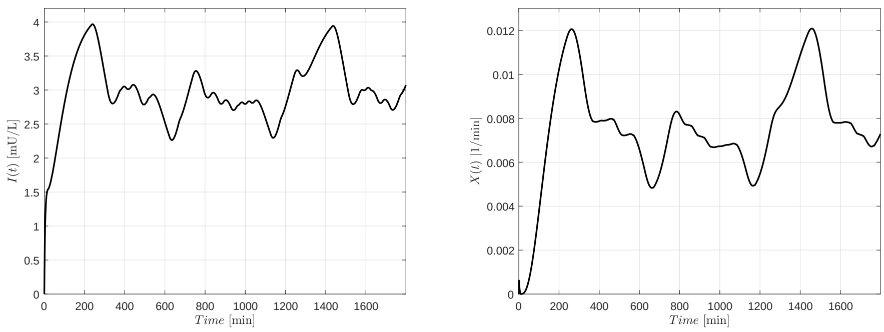

Figure 2.5: Variation of insulin levels and variation of the insulin-excitable tissue glucose uptake activity over time

2.2.1.2 Results of Scenario 2

The applied disturbance input in accordance with the detailed protocol can be seen on LHS of Figure 2.6. In this case, we have applied higher inputs in order to be sure that the developed control framework is able to deal with unfavourable external excitation.

Figure 2.6. (RHS) is the calculated and administered control signal. As it is visible, the dynamics of it is significantly different than the previous case due to the different settings in the applied cost function.

Figures 2.7. represent the absorption of the glucose and its appearance in the blood with the dynamics determined by the model. Figures 2.8. show the absorption of insulin from the interstitium to the blood.

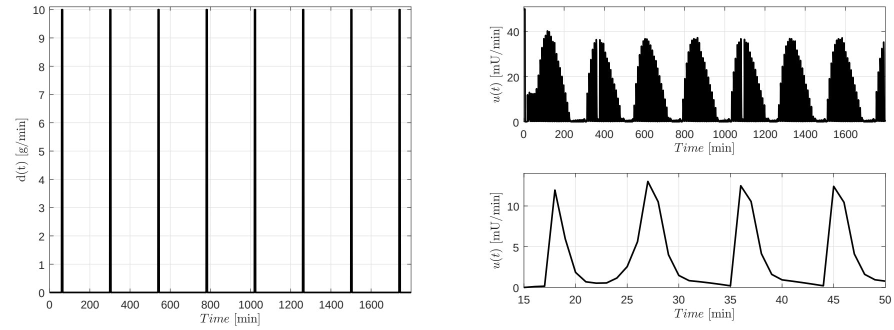

Figure 2.6: Applied disturbance (CHO intake) – 10 g over 5 minutes at each 240$^{\text{th}}$ time instance (LHS) and the calculated control signals (RHS). The upper figure represents the whole time horizon. The lower figure shows a piece of the whole time horizon between 0 and 40 minutes.

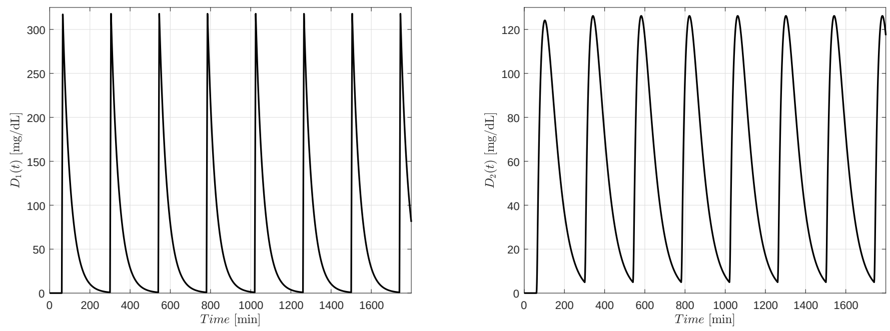

Figure 2.7: Variation of the first (LHS) and second (RHS) states of the glucose absorption subsystem.

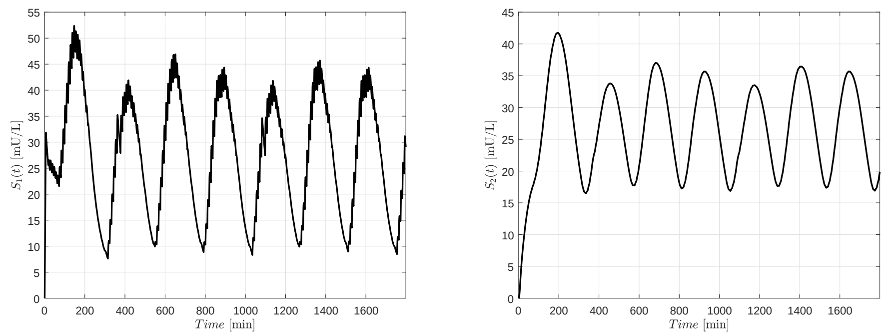

Figure 2.8: Variation of the first (LHS) and second (RHS) state of the insulin absorption subsystem.

The main result can be seen on Fig. 2.9. Though I drastically increased the disturbance input signal the controller was able to deal with the situation and realized appropriate control action – without domain violation from the determined u_{bias} point of view. The blood glucose level was inside the selected healthy range without any hypo- or hyper-glycemia. Moreover, the BG level oscillated around the reference trajectory – G^N – as it was expected.

Figs. 2.10 represent the variation of the blood insulin level and the intermediate state $X(t)$. Due to the higher frequency of the control signal these are oscillating with a higher frequency as well – which is directly reflected in the blood glucose level as well, since the $X(t)$ mediates the insulin's effect on $G(t)$.

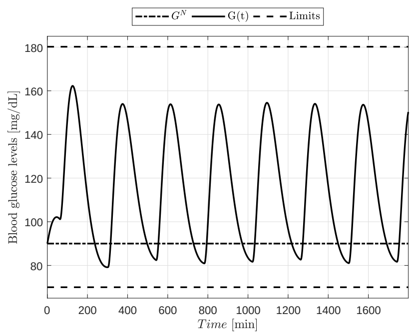

Figure 2.9: Variation of the blood glucose level over time

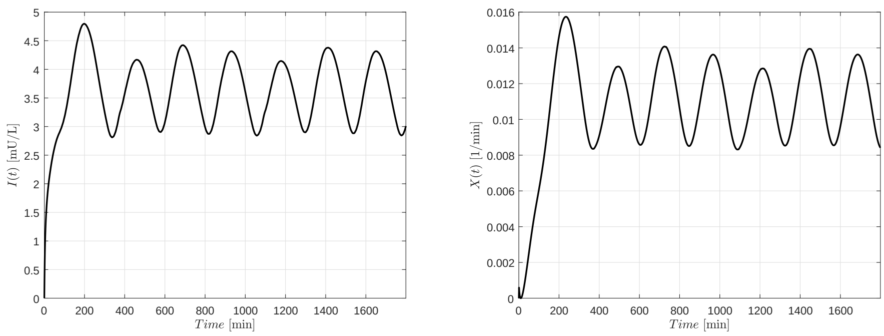

Figure 2.10: Variation of blood insulin and variation of the insulin-excitable tissue glucose uptake activity over time

2.2.2 Brief Conclusions

In this research my main goal was to design a RHC which is able to control the given patient model. For this purpose special non-quadratic cost functions have been suggested, and by the use of a specific model NP-based simulations were executed for two special scenarios by the use of the services of the MS EXCEL – VBA – Solver combination.

It is clearly visible in Figures 2.4. and 2.9. that the main requirement has been satisfied, since, the BG level was kept by the controller in the healthy range.

2.3 Novel Adaptive Extension of the RHC by Fixed Point Transformation–based Approach

"Model Predictive Control" (MPC) is a widely used approach in technical (e.g. [44]) and economic (e.g. [57]) application areas. The traditional MPC is formulated within the framework of a cost function–based optimal control in which the dynamics of the controlled system (i.e. the set of its equations of motion) mathematically is taken into account as a "constraint". The minimized cost function normally contains terms that depend on the tracking error, the control signal itself, and optionally, a separate term that gives "extra contribution" to the tracking error at the terminal point of the horizon. In the NP–based approximation the system's state variables and the control signals are considered over a discrete time–grid in each point of which Lagrangian multipliers determine the "reduced gradient" that is driven to zero numerically in order to find the solution. This solution consists of the estimated control signals and the estimated state variables. Whenever the available dynamic model is imprecise, this optimal design can be applied only for consecutive finite horizons, because the actual state of the controlled system propagates according to its exact dynamics. To reduce the effects of modeling imprecisions, for the next horizon–length design, the actually measured state variable at the last point of the previous horizon is used as starting point for the next one (e.g.[7]). In some special cases, this numerical calculation can be simplified. For instance, when the system's dynamics corresponds to an LTI model, and the cost functions have quadratic structure, the classical "Linear Quadratic Regulator" (LQR) is obtained [1], in which Riccati differential equation [19] is obtained for a symmetric matrix with a terminal condition. This matrix appears in a "driving term" in the separately obtained differential equation for the state variable with the given initial condition. In many applications state–dependent Riccati equations are in use (e.g. a survey paper in [116]). The mathematical framework of this traditional MPC can hardly be combined with the Lyapunov function–based adaptive control. Certain approaches combining MPC and Lyapunov's stability theorem can be found in the literature (e.g. [76], [112]). In 2009 in [25] an alternative approach was introduced in adaptive control that, instead of Lyapunov's "2nd Method", was based on Stefan Banach's "Fixed Point Theorem" [26].

The idea of transforming our task into a fixed point problem and solving it via iterations, has very early roots in the 17th century as the Newton-Raphson Algorithm, that has many applications even in our days (e.g. [117]). In 1922 Banach extended this way of thinking to quite wide problem classes [26]. According to his theorem, in a linear, normed, complete metric space (i.e. the "Banach Space") the sequence created by the contractive map $\psi : \mathbb{R}^m \mapsto \mathbb{R}^m$, $m \in \mathbb{N}$ as $x_{s+1} = \psi(x_s)$ is a Cauchy Sequence that converges to the unique fixed point of ψ defined as $\psi(x_\star) = x_\star$. (A map is contractive if $\exists 0 \leq H < 1$ so that $\forall x, y$ elements of the space $\|\psi(x) - \psi(y)\| \leq H\|x - y\|$.) In [118] the following transformation was used for this purpose: a real differentiable function $\varphi(\xi) : \mathbb{R} \mapsto \mathbb{R}$ was taken with an attractive fixed point $\varphi(\xi_\star) = \xi_\star$. It was used for the generation of a sequence of iterative signals as

$$q(i+1) = \left[\varphi(A\|x(q(i)) - x^{Des}\| + \xi_\star) - \xi_\star\right] \frac{x(q(i)) - x^{Des}}{\|x(q(i)) - x^{Des}\|} + q(i) \quad , \qquad (2.12)$$

in which the Frobenius norm was used. In (2.12) $A \in \mathbb{R}$ is an adaptive parameter. For $q(k) = q_\star$ that provides $x(q_\star) = x^{Des}$, (2.12) yields that $q(k+1) = q(k)$, that means that $q_\star$, i.e. the solution of our task, is the fixed point of this function. The convergence of this sequence was investigated in [119] by making the first order Taylor series approximation of $\varphi(\xi)$ in the vicinity of $\xi_\star$ and that of $x(q)$ around $q_\star$. It was found that if the real part of each eigenvalue of the Jacobian $\frac{\partial x}{\partial q}$ is simultaneously positive or negative, an appropriate parameter A can be so chosen that it guarantees the convergence. This result for the redundant robot arms of non–quadratic Jacobians in [78] was so applied that instead of the original problem $x^{Des} = x(q)$ the modified one $J^T(q)x^{Des} = J^T(q)x(q)$ was solved. By the Taylor series approximation of $x(q)$ around $q_\star$ it can be shown that the convergence will be determined by the positive semi–definite matrix $J^T(q)J(q)$ that has non–negative eigenvalues. (The zeros eigenvalues cause "stagnation" instead of infinite velocities in the singularities.) For adaptively tracking the "optimized trajectory" a similar transformation into a fixed point problem was applied.

In my research I have combined this newly proposed method with the RHC on the basis of the following simple idea. As in the classical optimal control, on the basis of the approximate model, both the state variables and the control signals are estimated for the horizon. However, instead of exerting the so obtained control signals, the system adaptively tries to track this "optimized trajectory". Though, this approach cannot guarantee the global asymptotic stability because it works with a bounded basin of convergence, its applicability was studied in case of hard non–linear control tasks as in anaesthesia control (e.g. [120], [121]), control of dynamically singular under-actuated mechanical systems (e.g. [122]), treatment for "Type 1 Diabetes Mellitus" (e.g. [123]), control of non–linear neuron models (e.g. [124], [125]), solution of the inverse kinematic task of robots [78], etc. On this reason the main novelty of my research consists in the combination of the "Fixed Point Transformation–based Adaptive Control" (FPTBAC) with the traditional RHC controllers. Because the restrictions in the allowed structure of the cost functions in the classic LQR evidently means a serious limitation.

2.3.1 Simulation Examples for Adaptive RHC

For the sake of simplicity an LTI–type 1st order model

$$\dot{x} = f(x,u) \stackrel{def}{=} -cx + du \quad , \quad c, d > 0 \tag{2.13}$$

given in (2.13) was considered. Its homogeneous version physically corresponds to the motion of a small mass–point connected to a linear spring in a viscous fluid. In this approach the acceleration's phase is neglected since the mass–point quickly achieves the velocity at which the viscous drag force compensates the spring's force that is proportional to its dilatation or contraction $\pm x$. On this reason, in this example the measurement unit of x is assumed to be m, and the control signal u is assumed to be measured in N units, while the dimension of $\dot{x}$ is ms^{-1}.

The exact model parameters were $c_E = 2\,s^{-1}$ and $d_E = 5\,m \cdot s^{-1} \cdot N^{-1}$. In the first set the approximate model values were $c_A = 1\,s^{-1}$ and $d_A = 4\,m \cdot s^{-1} \cdot N^{-1}$ that corresponds to underestimated values.

In the second set the approximate model values were $c_A = 3\,s^{-1}$ and $d_A = 6\,m \cdot s^{-1} \cdot N^{-1}$ that corresponds to overestimated values.

In the investigation 20 step horizon length was applied for the time-resolution $\Delta t = 10^{-3}\,s$. In the adaptive case, in the role of the trajectory to be tracked the trajectory optimized by the use of the approximate dynamic model was placed. The control parameters are given in Table 2.3.

Table 2.3: The control parameters

Parameter	**Numerical value**
Λ	$12.0\,s^{-1}$
K_c	$10^4\,m\cdot s^{-1}$
A_c	$1.3\times 10^{-4}\,s\cdot m^{-1}$
B_c	-1
A_1	$1.1\,m$
α_1	1
A_2	$5.0\times 10^{-3}\,N$
α_2	2
A_3	$2.5\times 10^{-2}\,m$
α_3	8

In the case of the underestimated parameter values, Fig. 2.11 reveals that the restricted control force caused corrupted tracking of the "Reference Trajectory" "`xRef`", but the "Optimized Trajectory" "`xOpt`" and the "Realized Trajectory" "`xReal`" in both cases were very close to each other. Subtle details in the tracking errors are revealed in Fig. 2.12. It can be clearly seen that due to the modeling errors the classic RHC works with increasing tracking error within a bounded horizon, while the adaptive version, after producing little transients, well tracks the optimized trajectory. The control forces are given in Fig. 2.13 according to which it can be stated that the adaptive deformation caused considerable modifications in the control forces "`uOpt`" and "`uAd`". The operation of the adaptive controller is illustrated by Fig. 2.13 (bottom RHS), revealing that the "realized" values well track the "desired" ones and they considerably differ from the "deformed" values. It can be seen that at the starting point of the horizon the adaptive controller generates some transients that in the first step a little bit corrupt the tracking error but this effect later is compensated by the adaptivity.

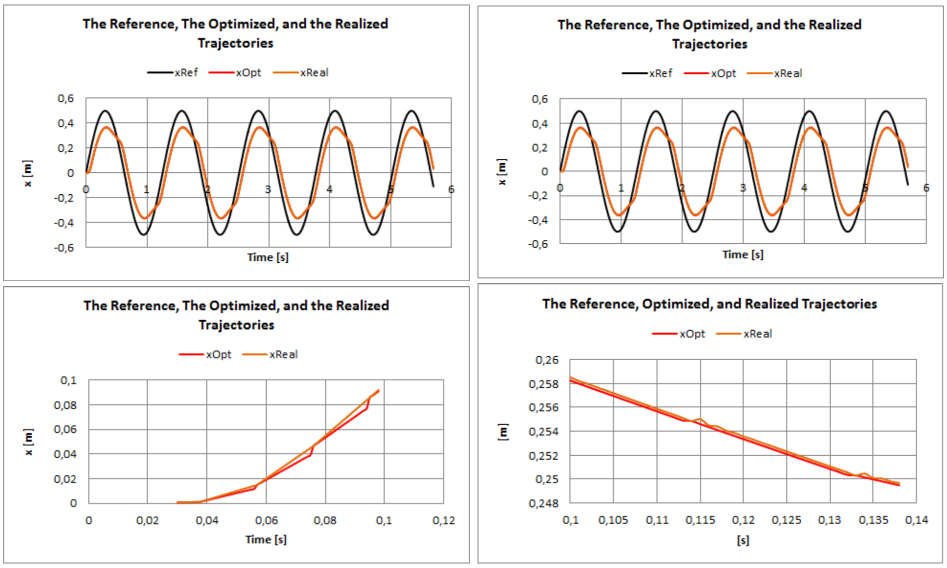

Figure 2.11: Trajectory tracking of the classic RHC (top LHS), its novel adaptive variant (top RHS) and their zoomed in excerpts (bottom LHS & RHS) for simulation series 1

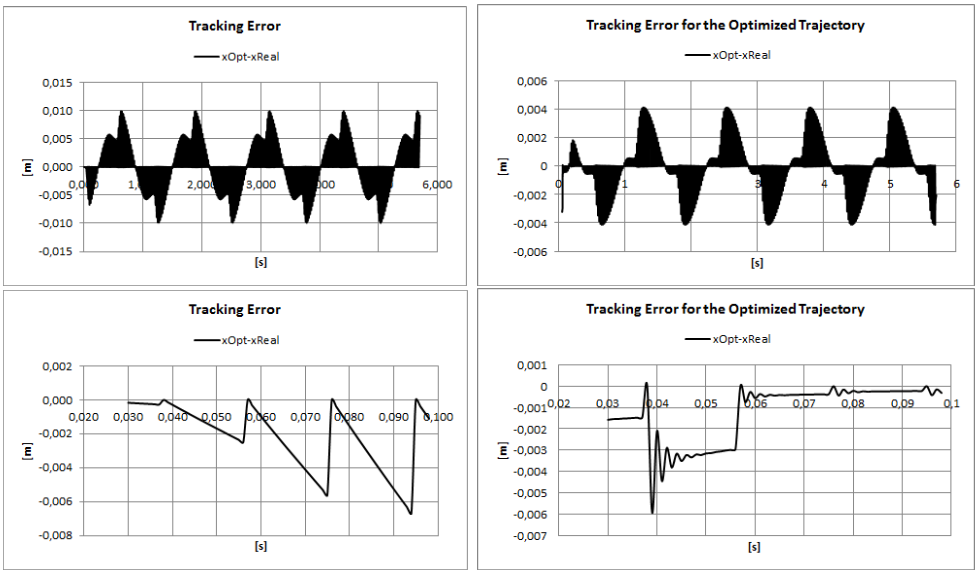

Figure 2.12: Trajectory tracking errors of the classic RHC (top LHS), its novel adaptive variant (top RHS) and its zoomed excerpts (bottom LHS & RHS) for simulation series 1

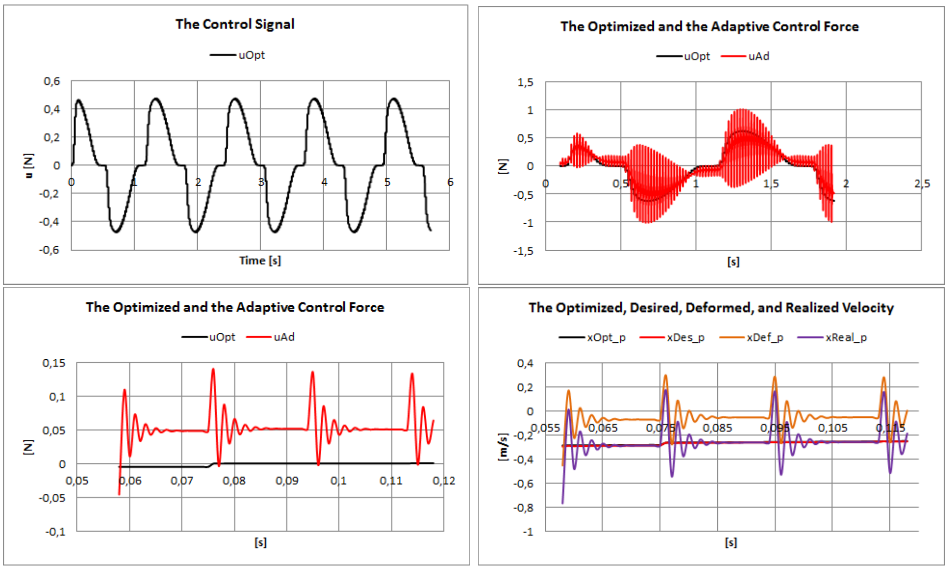

Figure 2.13: The control forces exerted by the classic RHC (top LHS) and by its novel adaptive variant (top RHS and bottom LHS) and zoomed in excerpts of the time-derivatives of the optimized, desired, deformed and the realized values of the novel adaptive RHC (bottom RHS) for simulation series 1

In the second set the approximate model values were $c_A = 3s^{-1}$ and $d_A = 6m \cdot s^{-1} \cdot N^{-1}$ that corresponds to overestimated values. The results are displayed in Figs. 2.14 and 2.15 that substantiate the same observations that were done in connection with that of series 1.

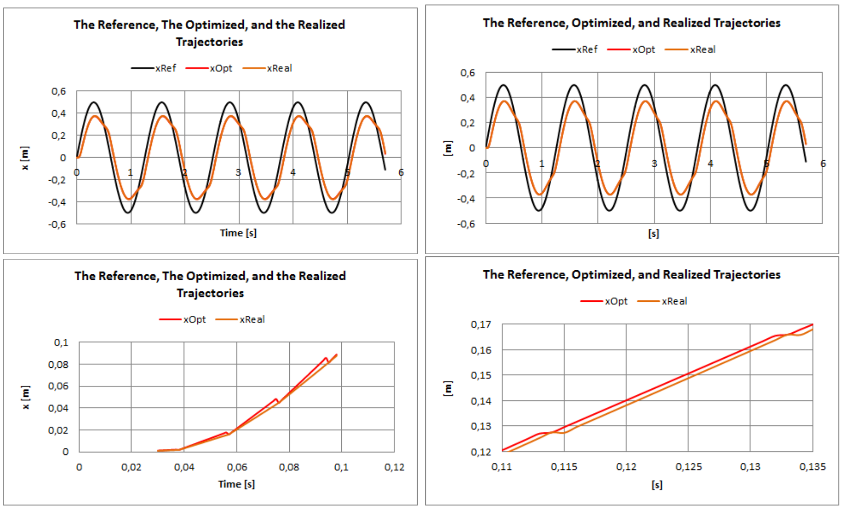

Figure 2.14: Trajectory tracking of the classic RHC (top LHS), its novel adaptive variant (top RHS) and their zoomed in excerpts (bottom LHS & RHS) for simulation series 2

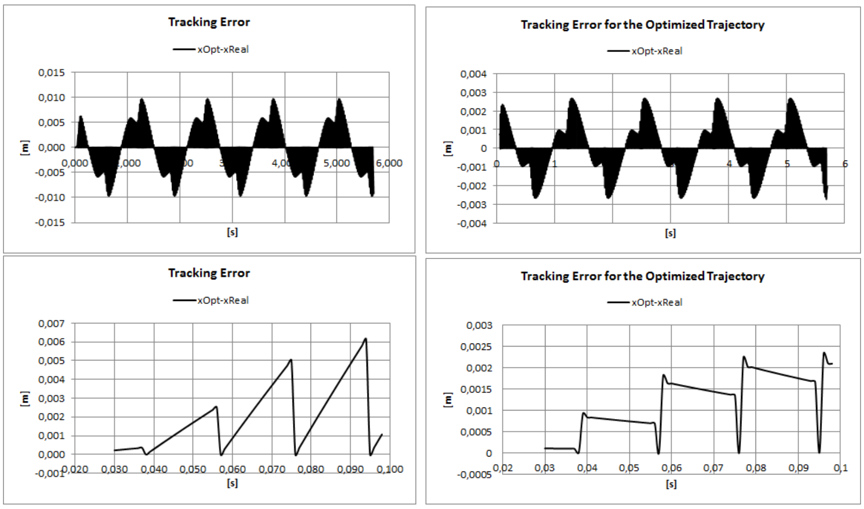

Figure 2.15: Trajectory tracking errors of the classic RHC (top LHS), its novel adaptive variant (top RHS) and its zoomed excerpts (bottom LHS & RHS) for simulation series 2

For the following alternative adaptive parameter settings values quite similar results were obtained $A_c = 1.3 \times 10^{-2}\, s \cdot m^{-1}$, $K_c = 10^2\, m \cdot s^{-1}$, and $A_c = 1.3 \times 10^{-3}\, s \cdot m^{-1}$, $K_c = 10^3\, m \cdot s^{-1}$ (with $B_c = -1$ in each case) that explains the name "Robust Fixed Point Transformation": for achieving good convergence not very precise estimation is needed for the control parameters.

2.4 Thesis Statement I.

I have proposed significant improvement of the traditional Nonlinear Programming–based Receding Horizon Controllers from two points of view: a) instead of the usual quadratic cost functions I suggested nonquadratic ones applying various, qualitatively interpreted format parameters; b) I invented the idea of the adaptive RHC controller by combining its original concept with the Fixed Point Transformation–based adaptive controllers: the trajectory computed by the traditional optimization was adaptively tracked instead using the traditionally estimated control forces. Based on the concept a) a solution was elaborated and simulated to treat patients suffering from Illness Type 1 Diabetes Mellitus (T1DM) to maintain the Blood Glucose (BG) level in the proposed range. The applicability of concept b) was illustrated by simulations for a simple first order paradigm. In both cases the MS EXCEL's embedded Solver solution was used to achieve the targeted results.

2.4.1 Substatement I.1

In order to control Type 1 Diabetes Mellitus a special dynamic system model was taken from the literature (the "Minimal Model") and subsequently it was modified. The essence of the modification was an extension with a sub-model to describe the absorption of the external glucose and insulin intake because during the daily routine these substances are not directly injected to the blood stream, therefore the characteristic of their appearance in blood has elongated dynamics that is better treatable than a "peak kind" ingress. Two different scenarios have been investigated to test my approach. In the first scenario, I applied "soft" disturbance and smaller penalties via the developed cost function in order to make sure that the controller design is possible at all and appropriate control action can be achieved by using the continuous optimization. In the second test scenario I used unfavourable, cyclic disturbance signal with high amplitude to test the "robustness" of the proposed controller. The developed RHC controller was able to handle the load and provided satisfactory control action. Furthermore, in both cases the BG level was kept in the predefined healthy range. In its structure the suggested approach can be further improved by the combination with a Fixed Point Transformation–based adaptive solution.

2.4.2 Substatement I.2

In case of the combination of the RHC and Fixed Point Transformation (FPT) a novel adaptive RHC controller was suggested in which the available approximate dynamical model of the controlled system is used as a constraint

for the calculation of the estimated optimized trajectory and the control signals over a finite time–grid in a Nonlinear Programming (NP) approach. In contrast to the traditional RHC that exerts the so estimated control signals and consecutively redesigns the tracking horizon, in my approach the so estimated optimized trajectory is adaptively tracked by a Robust Fixed Point Transformation–based Adaptive controller. The applicability of this approach is demonstrated by a comparative analysis of the operation of the traditional and the novel adaptive RHC controllers for a simple LTI system and strongly non–linear cost functions that exclude the use of the usual LQR approach. These investigations serve as the first step towards developing the adaptive RHC based on NP and FPT–based design.

Related own publications: [A. 1], [A. 2]

Chapter 3

Adaptive RHC for Special Problem Classes Treatable by the Auxiliary Function Approach

Non–linear Programming provides a practical, reduced–complexity solution for the realization of Model Predictive Controllers in which a cost function representing contradictory limitations is minimized under the constraints that express the dynamical properties of the system under control. For non–linear system models and non–quadratic cost functions the solution over a finite time–grid can be obtained by the use of Lagrange's Reduced Gradient Method that needs complicated numerical calculations. In this research it is shown that under not too limiting conditions this procedure can be replaced by a simple fixed point seeking iteration based on Banach's Fixed Point Theorem. The simplicity of the proposed algorithm widens the possibility for the practical applications of the Receding Horizon Control method. The same algorithm is used for adaptively and precisely tracking the "optimized trajectory" that can be constructed by the use of a dynamic model of "overestimated" parameters in order to evade dynamical overloads in the control process. To illustrate the efficiency of the method the Receding Horizon Control of a strongly non–linear, oscillating system, the van der Pol oscillator [126, 127] is presented. In the simulations three different parameter settings are considered: one of them produces the trajectory to be tracked, the second one is used for the optimization, and the third one serves as the model of the controlled system.

3.1 Scientific Antecedents

The realization and simple applicability of the well known "Model Predictive Controllers" (MPC) is famous for the use in different fields of life in case of controlling the systems, as discussed with details in Chapter 2.3. It was pointed out that in a "general case", in which the cost functions do not have quadratic structure and the daynamic model under consideration is not of LTI–type, the Newton–Raphson method [128] can be used for finding some starting point on the hyper–surface that represents the constraints, then Lagrange's Reduced Gradient Method [8] can be applied for finding the local optimum. It was mentioned, too, that for not very big problem the *MS EXCEL's Solver Package* (provided by an external firm Front–line Systems, Inc.) in combination with a little programming efforts in *Visual Basic for Applications (VBA)* in the background serves as an excellent tool for finding the solution.

It is a reasonable expectation that this complicated procedure can be evaded in the control of a system class in which a) the cost functions contain separate differentiable contributions for penalizing the tracking error and the too big control effort, and b) the mathematical form of the system's model under control is ab ovo known. In this case the appropriate gradients can be analytically calculated, and the EXCEL – VBA programming background does not offer further convenience, especially if the GRG algorithm can be replaced by a simpler one. This program is briefed in the next section.

3.2 Introduction of the "Auxiliary Function"

The NP approach of MPC uses a discrete time–grid over which the cost function in (3.1) has to be minimized

$$\sum_{i=0}^{N-1} J(x_i, u_i) + F(x_N) \quad , \tag{3.1}$$

where $F(x_N)$ gives an "extra weight" to the last point of the horizon under consideration. The optimization must be done under various constraints. The main constraint expresses the physical properties (i.e. dynamic model) of the controlled system $\dot{x} = f(x,u)$ by approximating it over the time–grid as $\frac{x_{i+1}-x_i}{\Delta t} \approx f(x_i, u_i)$, in which x denotes the state variable, and u is the control signal. By the use of Lagrange's RGM method he originally invented for use in the formulation of Classical Mechanics in [8] the "Auxiliary Function" (AF) can be introduced as $\Phi = \Phi(\{x\}, \{u\}, \{\lambda\})$

$$\Phi = \sum_{i=0}^{N-1} \left[J(x_i, u_i) + \lambda_i^T \left(\frac{x_{i+1} - x_i}{\Delta t} - f(x_i, u_i) \right) \right] + F(x_N) \qquad (3.2)$$

that has neither maximum nor minimum in the space of its variables, however, in the local optimum its partial derivative according to its each independent variable must be 0. The $\frac{\partial \Phi}{\partial \lambda} = 0$ conditions, i.e. the 0 derivatives according to the Lagrange multipliers express the requirement that the solution must be located on the hyper–surface that is determined by the constraint equations in which the constraint terms are set to 0. The $\frac{\partial \Phi}{\partial x} = 0$ equations mean that the reduced gradient must be 0 in the local optimum, while the partial derivatives according to the control signal components have to satisfy the extra conditions $\frac{\partial \Phi}{\partial u} = 0$.

In general, when no idea on the solution we have, all the complex numerical procedure described in Chapter 2.3 must be executed.

In simpler cases the above numerical procedure can be evaded: e.g. the optimization problem in the case of the calculation of the Moore–Penrose pseudoinverse can be solved "manually" simply by inverting a quadratic matrix if it is not singular. In the case of quadratic cost functions and LTI dynamic models certain "analytical manipulations" of the equations can be done "by hand" and we arrive at the solution of the Riccati equation. (Traditionally this step is important because Riccati elaborated a trick by the use of which the solution of this non–linear differential equation can be obtained by solving linear equation.) In [81] it was realized, that even if the cost functions are not quadratic, the problem has similar qualitative features than in the case of the quadratic cost functions, though in this case no manipulation by hand is possible to arrive to the Riccati equation. Instead of that an FPI–based numerical procedure was suggested for solving the problem even for not LTI dynamic models.

3.3 Analogy of the RHC with The Solution of The Inverse Kinematic Task of Robots

The task is to find appropriate joint coordinates q for a given x^{Des} "desired position" expressed in Cartesian frame fixed to the workshop. In the case of redundant robots, in which kinematic redundancies make the arm structure quite dexterous, this task has normally ambiguous solution. Furthermore, closed form solution of the inverse kinematic task exists only for special arm constructions, e.g. in the case of a PUMA–type robot [129]. As a general possibility, the differential solution based on the use of the Jacobian $\frac{\partial x}{\partial q}$ in a function of a *scalar variable* $\xi \in \mathbb{R}$

as $x(\xi) = x(q(\xi))$ is considered in the equation;

$$\frac{\mathrm{d}x_j}{\mathrm{d}\xi} = \sum_i \frac{\partial x_j}{\partial q_i}\frac{\mathrm{d}q_i}{\mathrm{d}\xi} \equiv \sum_i J_{ji}\frac{\mathrm{d}q_i}{\mathrm{d}\xi} \quad , \tag{3.3}$$

where the *initial conditions* as $x(\xi_{ini}) = x_{ini}$ and $q(\xi_{ini})$ are known. The traditional solutions contain some *generalized inverse* as e.g. the *Moore–Penrose Pseudoinverse* [130, 131] in which a quadratic matrix to be inverted is singular in, and ill–conditioned in the vicinity of the kinematic singularities of the robot arm. The general problem is that such a solution generates huge joint coordinate time–derivatives therefore it is expedient to "tame" the original task to evade the numerical inconveniences, as e.g. in the method of *Damped Least Squares* [132].

As an alternative of the traditional approach in [78] the original task was transformed into a fixed point problem that subsequently was solved by simple iteration, discussed in detail in section 2.3 and equation (2.12). Its special advantage is that it automatically shows stable solution in and in the vicinity of the kinematic singularities without the use of any "complementary trick", and automatically selects one of the ambiguous solutions. On this reason the use of this algorithm for driving the gradient of the auxiliary function to zero in the novel RHC controller was suggested.

3.4 Simulation Investigations for the van der Pol Oscillator

The investigated strongly non–linear 2nd order physical system was the van der Pol oscillator invented in 1927 [126]. Its equation of motion is given in (3.4)

$$\ddot{x} = \frac{-kx - b(x^2 - d)\dot{x} - cx^3 + eu}{m} \equiv f(x, \dot{x}, u) \quad , \tag{3.4}$$

in which u is the control force, x and $\dot{x}$ are the state variables. Parameters $k > 0$ and $c > 0$ describe a spring that "strengthens" with increasing extension x, parameter $b > 0$ describes viscous damping if $x^2 > d$, and excitation for $x^2 < d$. Due to it the state $x \equiv 0$ is an unstable equilibrium point: the smallest disturbance brings about excitation and drives the system into non–linear oscillation that is bounded by the dissipative nature of the term $-b(x^2 - d)\dot{x}$ for $x^2 > d$. Parameter e describes the system's sensitivity for the control force u. The appropriate model parameters are given in Table 3.1. For the *dynamic control* $\Lambda = 2.0\,s^{-1}$ was used, parameter A in equation (2.12) was $A_{dc} = -0.5$. For the purpose of the optimization various values were studied for A_{opt}. The time resolution of the grid was $\Delta t = 10^{-3}\,s$, the

Table 3.1: The applied model parameters

Param.	Exact	Approx.	Traj. generator
m	1.0	2.0	3.0
k	100.0	130.0	140.0
b	1.2	1.5	2.0
d	1.0	1.3	3.0
c	0.5	0.8	0.6
e	2.0	1.5	1.0

horizons consisted of $G = 10$ grid points, that, in the case of a 2^{nd} order system corresponds to 8 independent state variables (the initial conditions correspond to two independent grid points at the beginning of the horizon), and on the same reason we have 8 independent Lagrange multipliers and 8 independent control signals that determine the system's motion over the grid. No special terminal cost was applied, and the "auxiliary function" had the structure as follows:

$$\begin{aligned}\Psi = &\sum_{j=3}^{G} \left| \frac{x_j^N - x_j}{A_x} \right|^{\alpha_x} + B_u \sum_{j=1}^{G-2} \left| \frac{u_j}{A_u} \right|^{\alpha_u} + \sum_{j=1}^{G-2} \lambda_j \left[x_{j+2} - 2x_{j+1} + x_j \right] + \\ &+ \sum_{j=1}^{G-2} \lambda_j \left[-\Delta t^2 f(x_j, \dot{x}_j, u_j) \right] \quad ,\end{aligned} \tag{3.5}$$

in which for $\alpha_x > 1$ and $\alpha_u > 1$ the tracking error and the control force are well tolerated if $|x^N - x| < A_x$ and $|u| < A_u$, respectively, but they are strongly penalized over these limits. In (3.5) the term $f(x_j, \dot{x}_j, u_j)$ can be approximated as $f(x_j, \frac{x_{j+1}-x_j}{\Delta t}, u_j)$, and the terms in $\nabla\Psi$ and the Jacobian of the problem can be calculated in closed form for optimization. (For sparing room these derivatives are not detailed here.)

To highlight the role of the "auxiliary function", it has to be noted that the original numerical procedure based on the calculation of Lagrange's reduced gradient stops at the local optimum at which $\frac{\partial\Psi}{\partial\lambda} = 0$ guarantees that the solution is located on the hyper–surface determined by the constraints, while the $\frac{\partial\Psi}{\partial x} = 0$ equation means that the "reduced gradient" is zero, therefore no better point in the vicinity of the found solution exists. In very special cases, e.g. in the case of the Moore–Penrose pseudoinverse [130, 131], the whole numerical procedure can be evaded since the $\nabla\Psi = 0$ equations can be so utilized that the solution can be immediately obtained by a conventional matrix inversion. In other cases, as e.g. in the case of the Jacobi iteration [133], certain equations that appear in the set $\nabla\Psi = 0$ can be utilized for expressing some independent variables of the problem as the function of other ones, and in this manner at least the dimensions of the numerical procedure can be reduced. In our case $\nabla\Psi$ was directly driven to 0 with the fixed point iteration–based procedure.

In equation (2.12) the function $\varphi(x) = \frac{x}{2} + 0.3$ was in use. In the investigations the *trajectory generator* was excited with a constant force $F_\gamma = 300.0N$ that makes it settling down at the damped region. The control parameters were set as follows: $A_x = 0.5m$, $\alpha_x = 6.1$, $A_u = 100.0N$, $\alpha_u = 8.0$, $B_u = 100$. The RHC algorithm contained 100 internal iterations. The $A_{opt} = -1 \times 10^{-5}$ setting represents too slow iteration. Figure 3.1 reveals that the "optimized" trajectory is very far from the "nominal" one, and that the internal iteration did not result in good improvement of $\nabla\Psi$. The counterpart of Fig. 3.1 for $A_{opt} = -1 \times 10^{-2}$ is displayed in Fig. 3.2. It reveals that the tracking error is practically kept under $0.5m$ that is compatible with the setting $A_x = 0.5m$, $\alpha_x = 6.1$. It is also clear that the $\|\nabla\Psi\|$ went down from the value 15 to ≈ 0.13, i.e. the inner iteration was really responsible for driving $\nabla\Psi$ towards zero.

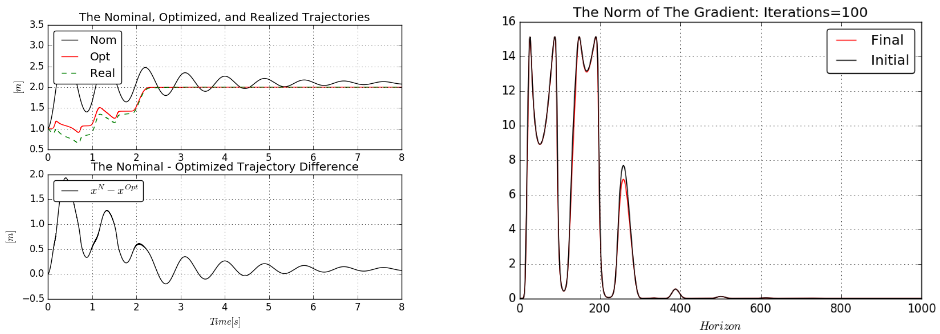

Figure 3.1: Trajectory tracking for too small adaptive parameter in the optimization ($A_{opt} = -1 \times 10^{-5}$)

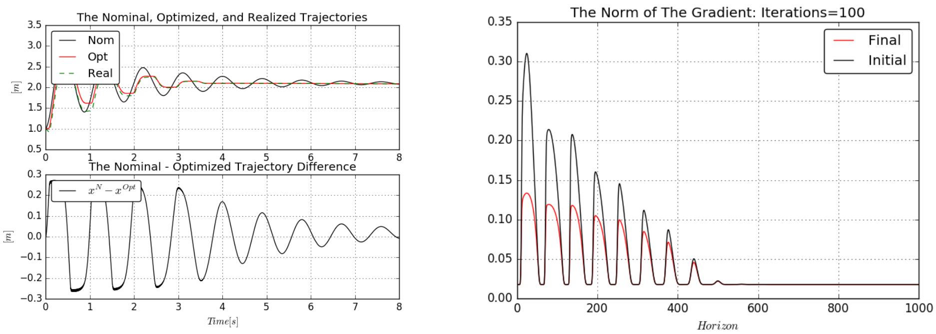

Figure 3.2: Trajectory tracking for appropriate adaptive parameter in the optimization ($A_{opt} = -1 \times 10^{-2}$)

Regarding the adaptive tracking of the optimized trajectory it can be seen that in both cases the adaptivity that was switched on in the beginning of the 2$^{\text{nd}}$ horizon, produced good results. Figure 3.3 explains its reason: the "desired" 2$^{\text{nd}}$ time–derivatives are well approximated by the realized ones while they considerably differ from the "adaptively deformed" values. The significance of the dynamic adaptivity in trajectory tracking is also substantiated by Fig. 3.4, that describes the case in which this dynamic adaptivity was switched off: the realized trajectory even does not approach the optimized one.

In the LHS of figure 3.5 explains the reason for the remnant part of $\nabla\Psi$: the minimal eigenvalue is 0, therefore the theoretically expected occurrence of "stagnation" was substantiated by the computations. The RHS of figure 3.5 reveals great fluctuations in the Lagrange multipliers and the control forces. It worths noting that in [80] similar fluctuations were observed in connection with a similar problem solved by the use of the EXCEL–Solver–Visual Basic apparatus.

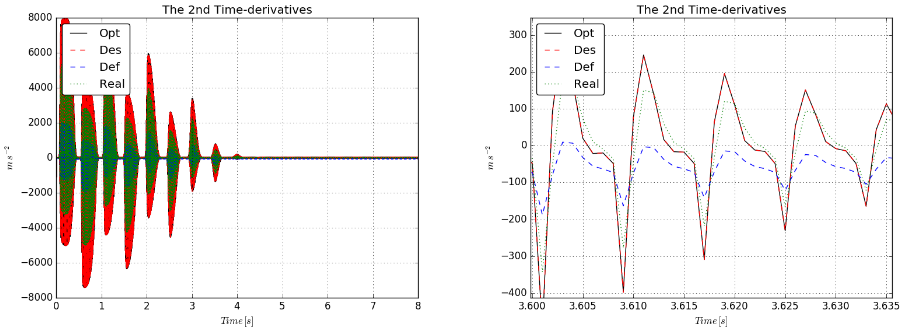

Figure 3.3: The second time-derivatives in the adaptive dynamic tracking of the optimized trajectory for appropriate adaptive parameter in the optimization ($A_{opt} = -1 \times 10^{-2}$)

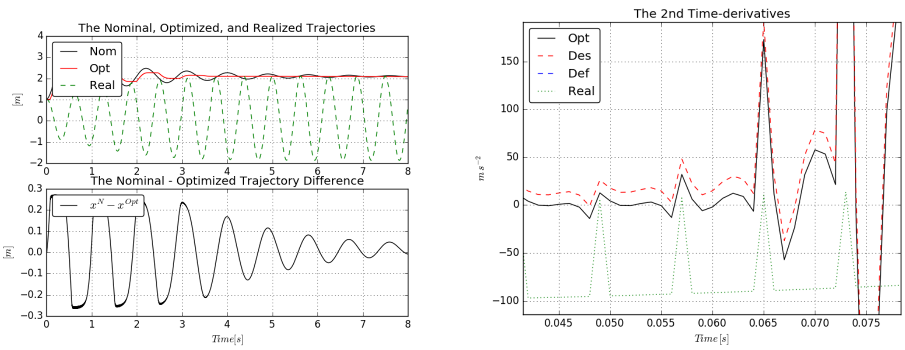

Figure 3.4: Trajectory tracking without dynamic adaptivity for appropriate adaptive parameter in the optimization ($A_{opt} = -1 \times 10^{-2}$)

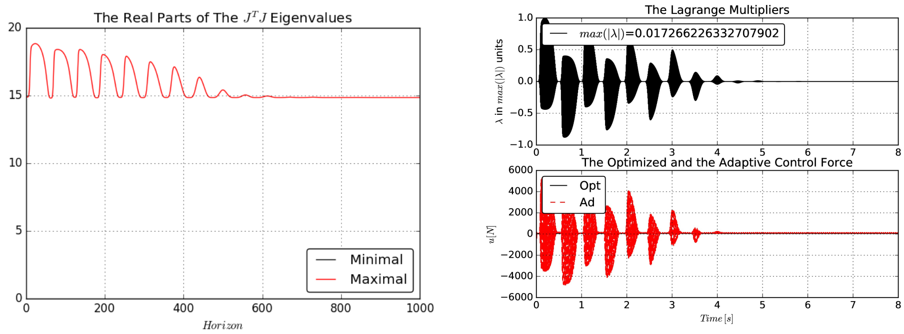

Figure 3.5: The maximal and minimal eigenvalues of $J^T J$ in the internal iterations and the Lagrange multipliers and the control signals for appropriate adaptive parameter in the optimization ($A_{opt} = -1 \times 10^{-2}$)

3.5 Simulations for the Duffing Oscillator

In this example the introduced simple iterative solution is investigated for the Adaptive Model Predictive Control (AMPC) for another strongly nonlinear dynamic paradigm, the Duffing oscillator. The main idea is again the replacement of the numerically much more complex Reduced Gradient method in the optimization task under constraints when the cost function has relatively simple structure.

The considered extension of the general equation of motion of the Duffing oscillator is explained in (3.6).

$$m\ddot{q} = -kq - lq^3 - b\dot{q} + u \ , \tag{3.6a}$$

$$\ddot{q} = -\frac{k}{m}q - \frac{l}{m}q^3 - \frac{b}{m}\dot{q} + \frac{1}{m}F \ , \tag{3.6b}$$

$$\dot{q}_i \approx \frac{q_{i+1} - q_i}{\Delta t} \ , \tag{3.6c}$$

$$\ddot{q}_i \approx \frac{\dot{q}_{i+1} - \dot{q}_i}{\Delta t} = \frac{q_{i+2} - q_{1+1} - q_{i+1} + q_i}{\Delta t^2} \approx \frac{q_{i+2} - 2q_{i+1} + q_i}{\Delta t^2} \tag{3.6d}$$

At the “auxiliary function” can be formulated as follows:

$$\Phi = \sum_{i=3}^{N-1} \left\| \frac{q_i^N - q_i^h}{A_q} \right\|^{\alpha_q} + \left\| \frac{q_N^N - q_N^O}{A_F} \right\|^{\alpha_T} + \sum_{j=1}^{N-2} \left\| \frac{F_j}{A_F} \right\|^{\alpha_F} + \sum_{i=1}^{N-2} \lambda_i \left[-\frac{k}{m}\dot{q}_i - \right. \\ \left. -\frac{l}{m}\dot{q}_i^3 - \frac{b}{m}\left(\frac{\dot{q}_{i+1} - \dot{q}_i}{\Delta t} \right) + \frac{F_i}{m} - \frac{\dot{q}_{i+2} - 2\dot{q}_{i+1} + \dot{q}_i}{\Delta t^2} \right] \ . \tag{3.7}$$

The appropriate partial derivatives are so complicated that they are not detailed in the Thesis. The details were published in [A. 3].

Since in the calculations a horizon consists of 6 grid points, the first two ones correspond to the "initial state" of the system, i.e. the "initial coordinate" and the "initial velocity", therefore the free variables of the optimum problem are the coordinate values x in grid points $3, 4, 5$ and 6. In similar manner the control force values u in the grid points $1, 2, 3$ and 4 are the independent variables of the problem. Since the system response is the second time–derivative of x that numerically can be estimated from minimum 3 grid points we have 4 Lagrange multipliers.

3.5.1 Simulation Results

The appropriate model parameters of the Duffing oscillator considered are given in Table 3.2.

Table 3.2: The control parameters

Parameter	Exact Approximate	Traj. Generators
$m[kg]$	1.0 1.2	0.9
$k[Nm^{-1}]$	100 120	90
$l[Nm^{-3}]$	0.5 0.6	0.4
$b[Nms^{-1}]$	1.2 1.5	0.7

In the cost function the *"original form"* of the tracking error contribution was $h_q(x^N - x) = \left(\frac{|x^N - x|}{A_x}\right)^{\alpha_x}$, and for the prohibition of the too big control effort $h_u(u) = \left(\frac{|h|}{A_u}\right)^{\alpha_u}$ were used. For very big α_x the penalty has very fast increase in the region $\left|x^N - x\right| > A_x$ and it is very small for $\left|x^N - x\right| < A_x$. Since for great α_x this may result in numerical difficulties, these power functions were "tamed" in the following manner: at first the cost functions were modified as $h = ln\left(1 + M\left[\frac{\mathrm{sgn}(x)x}{A}\right]^{\alpha}\right)$, and in its 1st and 2nd time–derivative the function $\mathrm{sgn}(x)$ was considered as a classical function that cannot be differentiated in a single point $x = 0$. In the next steps, in the derivatives, this function was "softened" as $\mathrm{sgn}(x) \approx \tanh\left(\frac{x}{w}\right)$. The 1st and 2nd order derivatives of the cost function had the structure:

$$x' = \frac{\alpha}{A_x} M \frac{1}{1+\left(\frac{\tanh(\frac{x}{w})x}{A_x}\right)^{\alpha} M} \left(\frac{\tanh(\frac{x}{w})x}{A_x}\right) \tanh\left(\frac{x}{w}\right)^{\alpha-1} \quad , \tag{3.8a}$$

$$x'' = \frac{\alpha}{A_x^2} M \frac{\left[\frac{-M\alpha\left(\frac{\tanh(\frac{x}{w})x}{A_x}\right)^{2(\alpha-1)}}{1+M\left(\frac{\tanh(\frac{x}{w})x}{A_x}\right)^{\alpha}}\right]}{1+\left(\frac{\tanh(\frac{x}{w})x}{A_x}\right)^{\alpha} M} + (\alpha-1)\left(\frac{\tanh\left(\frac{x}{w}\right)x}{A_x}\right)^{\alpha-2} \quad . \tag{3.8b}$$

The other control parameters we used are: $A_u = 1.0, \alpha_u = 3.0, B_u = 1 \times 10^0$, where A_u used for basis, and α_u is an exponent for punishing the control force. B_u is the weighting coefficient or the control force restriction in the cost function,and for the starting value for tuning u variable was used is $u_{ini} = 10.0$ was taken. The control input used for dynamic tracking is $\Lambda = 1.0s^{-1}, w = 1.0 \times 10^{-6}$ as used for softening of the cost function for large costs.

In the sequel the following essential parameters were varied: $\alpha_x = 2$ contribution was corresponds to “soft tracking”, $\alpha_x = 6$ corresponds to “sharp tracking”. Parameter $A_x = 5 \times 10^{-2}$ means “strict tracking”, while $A_x = 1 \times 10^{-1}$ was used for “loose tracking”. The parameter in the scheme in Fig. 3.6, $A = -3 \times 10^{-1}$ corresponds to “slow dynamic tracking”, and $A = -3 \times 10^0$ means “fast dynamic tracking”. In the figures depicted below, results for the “nominal”, “optimized”, and “realized” trajectories have been clarified where the correlations between them can be observed. For the varied parameters, similarly, the difference between the nominal and optimized trajectories, and the real part of the eigenvalues also show dissimilar situations.

In Fig. 3.6 the adjustment of “fast, loose, and sharp” parameters,for the Nominal, Optimized, Realized Trajectory & Difference,The Norm of the Gradient: Iterations=100 and The Real Part of the $J^T J$ Eigenvalues, in Fig. 3.7 parameters for “fast, loose, and soft”, Fig. 3.8 “fast, strict, and sharp” adjusted parameters, in Fig. 3.9 “fast, strict, and soft”, in Fig 3.10, “slow, loose, and sharp”, whereas in Fig. 3.11 “slow, loose, and soft” parameters adjusted for tracking were chosen. In Fig. 3.12 “slow, strict, and sharp” whereas in 3.13 “slow, strict, and soft” parameters were explained. The conditions of trajectories illustrate an assorted scenario where the “optimized” and “realized” trajectories gradually track and meet the “nominal” one after exhibiting an initial jump, and the difference between “optimized” and “nominal” state variables is declining with an irregular shape. Similarly, identical shapes with minor error occurred while subtracting "optimized" state from "nominal" state. The gradient trajectories are decreasing with a consistent form, whereas in some cases such gradients decrease with an irregular and not consistent form. The figures are depicted below:

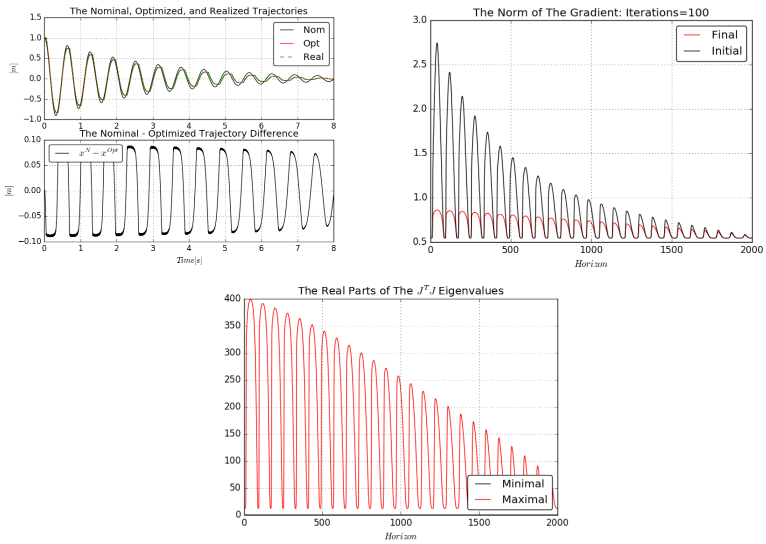

Figure 3.6: The Nominal, Optimized, Realized Trajectory & Difference,The Norm of the Gradient: Iterations=100 and The Real Part of the $J^T J$ Eigenvalues for fast, loose, and sharp tracking

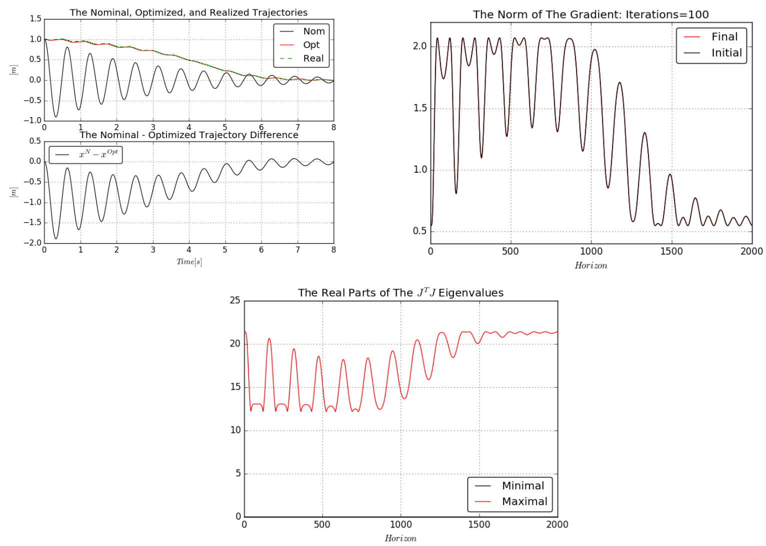

Figure 3.7: The Nominal, Optimized, Realized Trajectory & Difference, the Norm of the Gradient: Iterations=100 and The Real Part of the $J^T J$ Eigenvalues for fast, loose, and soft tracking

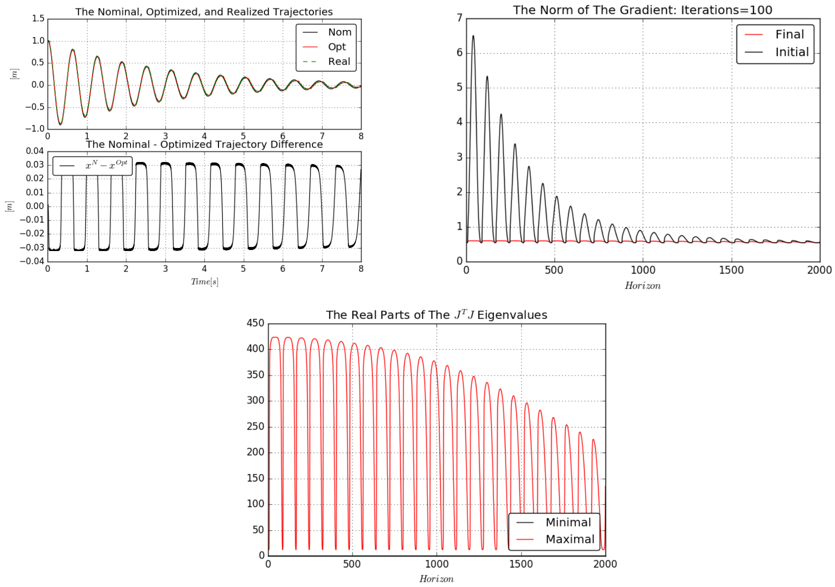

Figure 3.8: The Nominal, Optimized, Realized Trajectory & Difference, the Norm of the Gradient: Iterations=100, the Real Part of the $J^T J$ Eigenvalues for fast, strict, and sharp tracking

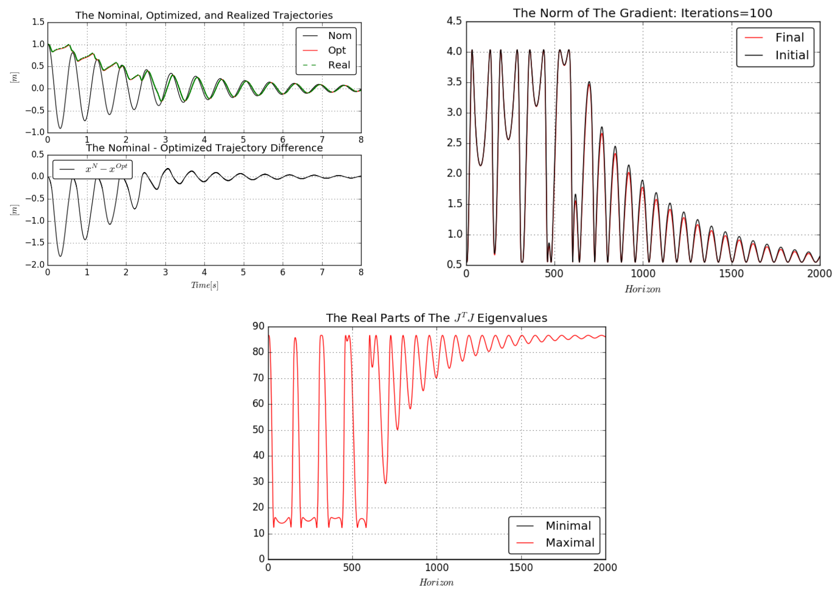

Figure 3.9: The Nominal, Optimized, Realized Trajectory & Difference, the Norm of the Gradient: Iterations=100, the Real Part of the $J^T J$ Eigenvalues for fast, strict, and soft tracking

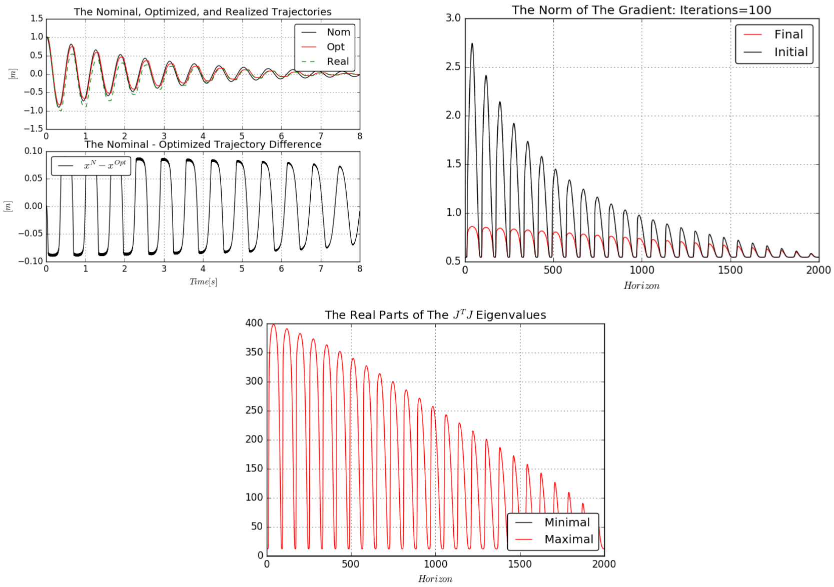

Figure 3.10: The Nominal, Optimized, Realized Trajectory & Difference, the Norm of the Gradient: Iterations=100, the Real Part of the $J^T J$ Eigenvalues for slow, loose, and sharp tracking

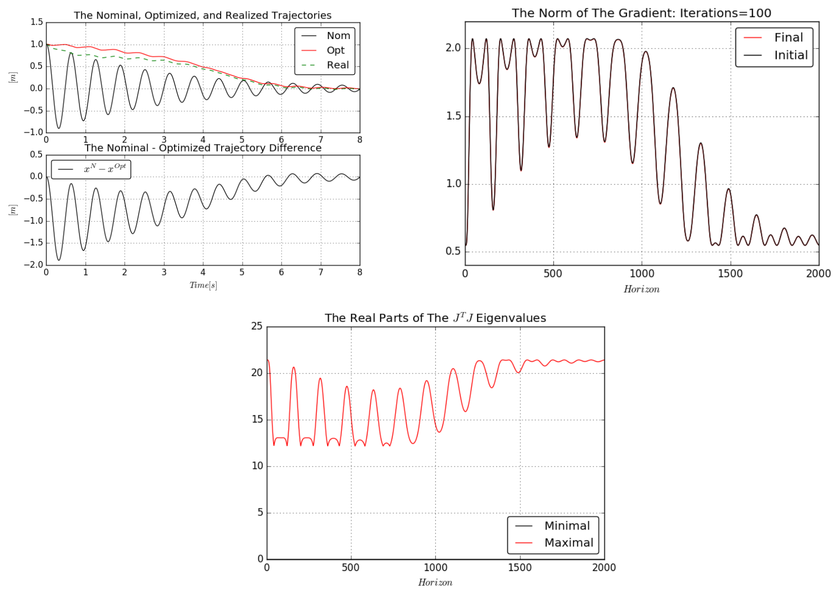

Figure 3.11: The Nominal, Optimized, Realized Trajectory & Difference, the Norm of the Gradient: Iterations=100, the Real Part of the $J^T J$ Eigenvalues for slow, loose, and soft tracking

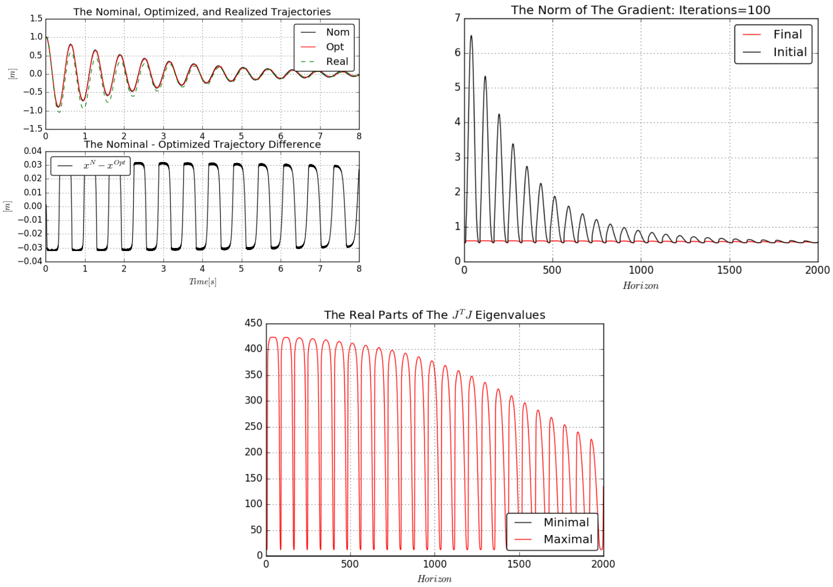

Figure 3.12: The Nominal, Optimized, Realized Trajectory & Difference, the Norm of the Gradient: Iterations=100, the Real Part of the $J^T J$ Eigenvalues for slow, strict, and sharp tracking

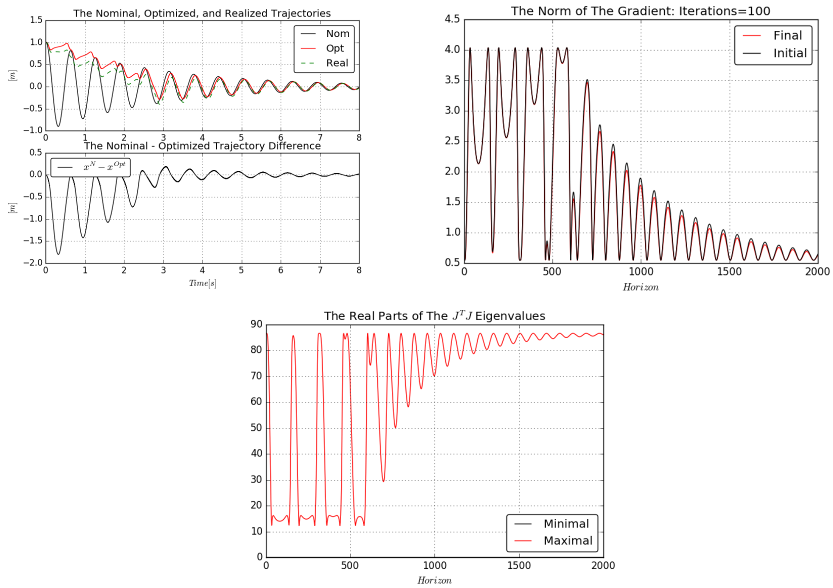

Figure 3.13: The Nominal, Optimized, Realized Trajectory & Difference, the Norm of the Gradient: Iterations=100, the Real Part of the $J^T J$ Eigenvalues for slow, strict, and soft tracking

3.6 Investigations Aiming at Further Possible Simplifications in the Application of the Fixed Point Iteration

It was found that in control problems just the calculation of the Jacobian means considerable programming and computational burden. To release it a recent solution was proposed for solving the inverse kinematic task by evading not only the inversion, but even the calculation of the Jacobian [134]. In this section it is shown by the use of a non-linear single degree of freedom paradigm that this simplification may be a viable route in solving "Adaptive Receding Horizon Control" (ARHC) problems. The idea was further extended in [A. 6] for Multiple Degree of Freedom, Higher Order Dynamical Systems (two coupled van der Pol oscillators) and found to be less successful. This fact can be explained by the very rich spectrum of the matrix that determines the possible convergence and divergence: in this case for a satisfactorily long horizon very big matrices are obtained the spectrum of which "cannot be kept under control". The convergence issues can be understood on the basis of the argumentation given in the sequel.

To achieve convergence in the case of the fixed point transformation suggested by Dineva in (2.12) the behavior of the sequence was investigated in the vicinity of this fixed point in [119] by the use of the 1st order Taylor series approximation of the function $F(\xi)$ around $\xi_\star$ and $f(q)$ around $q_\star$. She arrived at the conclusion that the expression in (3.10) well approximates the computations in the vicinity of the fixed point.

$$q_{i+1} = [F(A\|f(q_i)-\mathfrak{x}\| + \xi_\star) - \xi_\star] \frac{f(q_i)-\mathfrak{x}}{\|f(q_i)-\mathfrak{x}\|} + q_i \quad , \tag{3.9}$$

$$q_{i+1} - q_\star \approx \left[I + \left.\frac{\mathrm{d}F}{\mathrm{d}\xi}\right|_{\xi_\star} A \left.\frac{\partial f}{\partial q}\right|_{q_\star} \right] (q_i - q_\star) = [I + AM](q_i - q_\star) \quad , \tag{3.10}$$

in which $M \in \mathbb{R}^{n\times n}$, and $A \in \mathbb{R}$. This means the the error of the approximation of the solution varies according to the powers of the matrix $[I+AM]$. By the use of the Jordan canonical form (e.g. [135]) of this matrix she had shown that for satisfying the requirement $q_i \to q_\star$ as $i \to \infty$, it is satisfactory if the real parts of all eigenvalues of this matrix are simultaneously positive or simultaneously negative. In this case a small parameter A with appropriate sign can make the iteration convergent for each Jordan block. Of course, the speed of convergence depends on $|A|$: too big value may make the sequence divergent, and the smaller the value of $|A|$, the slower the convergence.

The use of this simple iteration for solving inverse kinematic tasks for non–redundant robots was investigated in [78]. For a simple, planar, 2 degree of freedom robot arm it was shown that the appropriate Jacobian $J(q) \stackrel{def}{=} \frac{\partial f}{\partial q}$ does not satisfy this restriction. However, this problem was simply eliminated by considering the modified problem $\tilde{x} \stackrel{def}{=} J^T(q)x = J^T(q)f(q)$ since in the Taylor series expansion of $f(q)$ around q makes the quadratic matrix $J^T(q)J(q)$ appear that normally is *positive semidefinite*:

$$\begin{aligned} &J^T(q)f(q) \approx J^T(q)f(q_\star) + J^T(q)J(q_\star)(q-q_\star) \text{ so} \\ &J^T(q)[f(q)-f(q_\star)] \approx J^T(q)J(q)(q-q_\star)\ , \end{aligned} \tag{3.11}$$

in which in the vicinity of $q_\star$ the $J(q_\star) \approx J(q)$ approximation was also used. This task modification automatically solved the problem for the *redundant robots* in which the number of the components of q is greater than that of x. The 0 "eigenvalues" of $J^T(q)J(q)$ correspond to *kinematic singularities*, and it was shown that the fixed point iteration–based solution behaved definitely nicely in the singularities and their vicinity and provided useful solutions to the inverse kinematic task without "complementary tricks" that always have to be applied in the approaches that somehow wish to use certain generalized inverse of $J(q)$.

In [A. 8], on the basis of simple geometric considerations it was shown that the condition for convergence introduced by Dineva requires too much: the matrix that satisfies it has to produce "contraction" in any direction. However, in the fixed point iteration not arbitrary directions occur, so it was reasonable to make an attempt to release this restriction. If we remain in the set of differentiable $\mathbb{R}^n \mapsto \mathbb{R}^n$ functions that have quadratic Jacobians, a "qualitative" property of the function can be introduced that corresponds to the generalization of the single variable "decreasing" and "increasing" functions in the case of multiple variable ones. If $\Delta f = f(x+\Delta x) - f(x)$, then it can be stated that if $\Delta x^T \Delta f > 0$ then the function f *varies approximately in the same direction* as the independent variable x does. If $\Delta x^T \Delta f < 0$, then f *varies approximately in the opposite direction*. Such qualitative properties of certain physical systems make it easy to realize their "iterative" or even fuzzy rules-based control (e.g. the use of the steering wheel, the brake, and the accelerator pedals of cars, etc.). Various cars can be driven by various chauffeurs on the basis of the qualitative knowledge that a small modification of the actual position of the steering wheel, the brake, and the accelerator will result in definite modification of the turning angle, deceleration, and acceleration. Accordingly, since $F'(\xi_\star)$ is a fixed value, in (3.10) the sign of the parameter A determines if for a given $(q_i - q_\star)$ the matrix $A \left.\frac{\partial f}{\partial q}\right|_{q_\star}$ will produce a modification "approximately in the same", or "approximately in the opposite direction". Since each matrix can be decomposed as the sum of its *symmetric* and skew symmetric parts as $M = \frac{1}{2}\left(M+M^T\right)+\frac{1}{2}\left(M-M^T\right)$, and in the product $\Delta q^T M \Delta q$ the skew symmetric part gives 0 contribution, only the symmetric part of the matrix is of interest for us. The positive semi–definite matrix $\left(\frac{\partial f}{\partial q}\right)^T \frac{\partial f}{\partial q}$ can work with an adaptive parameter of fixed sign. The basic idea in [A. 8] was the introduction of scalar parameter $\sigma \in \{+1,-1\}$ and using the iteration for the modified problem $\sigma x = \sigma f(q)$ in which in each control step the parameter σ was set according to the rule: $\sigma_1 = 1$, and for $i > 1$

$$\sigma_{i+1} = \begin{cases} 1 \text{ if } \Delta x_i^T \Delta f_i \geq 0 \ , \\ -1 \text{ otherwise } \ . \end{cases} \tag{3.12}$$

It was expected that in this manner the approximate direction keeping feature of the matrix used in the iterations was possibly maintained and by the use of a fixed adaptive parameter A the convergence could be guaranteed. This property was well illustrated by simulation results in inverse kinematics in which only relatively small matrices occur. However, in the RHC control long horizons produce large matrices with "rich" spectrum in which the "converging" and "diverging" contributions may have commensurate effects that may lead to less successful application.

In the present investigations the same idea is utilized in driving the gradient of the AF $\nabla\Phi(x,u,\lambda) \in \mathbb{R}^m$ towards zero. It is worth noting that the gradient of the gradient of the AF, i.e. $\nabla\nabla\Phi(x,u,\lambda) \in \mathbb{R}^{m\times m}$ by definition is a symmetric matrix. As is well known, the *real symmetric matrices* ($M \in \mathbb{R}^{n\times n}$ for which $M = M^T$) are special Hermitian matrices ($H \in \mathbb{C}^{n\times n}$) that by definition satisfy the restriction $H^{T*} = H$ that a) have *real eigenvalues* as $\{\mu_1,\dots,\mu_n \in \mathbb{R}\}$ and b) can be diagonalized by *orthogonal transformations* ($O^T O = I$) as $OHO^T = \langle \mu_1,\dots,\mu_n\rangle$. Therefore in this case the efficiency of the suggested algorithm can be expected. This expectation was only partly confirmed by the simulation results presented in Section 3.7.

3.7 New Simulations for the van der Pol Oscillator

In these investigations the same van der Pol oscillator was considered as in Section 3.4, with the same parameters determining the nominal trajectory, the exact, and the approximate models as in Table 3.1, and the auxiliary function of the problem was the same as in (3.5).

The adaptive tracking of the optimized trajectory had to be realized by the same FPI–based method that was detailed in [81], i.e. by a *Proportional, Derivative Tracking* with the *"Desired 2^{nd} Time-derivative"* $\ddot{x}^{Des} = \ddot{x}^{Opt} + 2\Lambda(\dot{x}^{Opt} - \dot{x}) + \Lambda^2(x^{Opt} - x)$ with $\Lambda = 2\,s^{-1}$.

In the program $A_x = 0.6$, $\alpha_x = 6$, $A_u = 100$, $\alpha_u = 8$, $B_u = 100$. To achieve better numerical stability the cost function was "bounded" by considering $|x| = x\mathrm{sign}(x)$ in $\hat{h}(x, A_x, \alpha_x) \stackrel{def}{=} \log\left(1 + \mathscr{M}\left[x\mathrm{sign}(x) A_x^{-1}\right]^{\alpha_x}\right)$ and its derivative

$$\frac{\mathrm{d}\hat{h}}{\mathrm{d}x} = \frac{\mathscr{M}\alpha_x \left[x\mathrm{sign}(x) A_x^{-1}\right]^{\alpha_x - 1} A_x^{-1}\mathrm{sign}(x)}{1 + \mathscr{M}\left[x\mathrm{sign}(x) A_x^{-1}\right]^{\alpha_x}} \tag{3.13}$$

in which the function $\mathrm{sign}(x)$ was replaced by its "softened variant" $\tanh\left(\frac{x}{\mathscr{W}}\right)$:

$$h(x, A_x, \alpha_x) \stackrel{def}{=} \log\left(1 + \mathscr{M}\left[x\tanh\left(\frac{x}{\mathscr{W}}\right) A_x^{-1}\right]^{\alpha_x}\right) \quad , \tag{3.14}$$

and its derivative was approximated as

$$\xi \stackrel{def}{=} x\tanh\left(\frac{x}{\mathscr{W}}\right) A_x^{-1}, \frac{\mathrm{d}h}{\mathrm{d}x} \approx \tanh\left(\frac{x}{\mathscr{W}}\right) \frac{\mathscr{M}\alpha_x \xi^{\alpha_x - 1} A_x^{-1}}{1 + \mathscr{M}\xi^{\alpha}} \tag{3.15}$$

In (3.15) the choice $0 < \mathscr{W} \ll A_x$ makes the function numerically treatable near $x = 0$, and for $0 < \mathscr{M}$ it remains numerically treatable for large x values.

In the simulations the initial value of the control signal was $u_{ini} = 50$. In (2.12) the adaptive function was $F(\xi) = \frac{\xi}{2} + 0.3$. The adaptive parameter was $A = -2.5 \times 10^{-2}$ in the optimization, and $A = -5 \times 10^{-1}$ in the adaptive tracking of the optimized trajectory. For driving $\nabla\Phi$ to 0 the number of the iterative steps was $N_I = 12$. The parameters $\mathscr{M} = 0.6$ and $\mathscr{W} = 0.1$ were experimentally chosen. The horizon length was $G = 12$ step (its minimal possible length is 6). The $\nabla\nabla\Phi$ matrix and its eigenvalues were computed only for demonstrating the direction–sensitivity of the FPI-based solution in which a fixed adaptive parameter A could drive the iteration into divergence, too. For the algorithm it was enough to compute the components of $\nabla\Phi$.

In Fig. 3.14 the results for the "linear cost terms" are given. It can be seen that for higher amplitudes this term cannot exert enough force to well approximate the nominal trajectory by the optimized one. The "optimized" trajectory is precisely tracked by the adaptive tracking control. In the region of "optimization failure" the no significant reduction in $|\nabla\Phi|$ was achieved. The eigenvalues of $\nabla\nabla\Phi$ reveal the strong "direction–dependence" of the problem. To improve the optimized tracking α_x was increased to 0.8. Figure 3.15 shows considerable improvement in the optimal trajectory under considerable direction-sensitivity. In the results displayed in figure 3.16 the "traditional quadratic cost function" was applied. This parameter still alows considerable tracking error max. ± 1.1 that is greater than $A_x = 0.6$. Figure 3.17 testifies that for $\alpha_x = 6.0$ the controller is quite tolerant for the small tracking error in the range ± 0.3 that –according to the expectations– it does not allow the occurrence of higher tracking errors. In spite of the direction sensitivity of $J \stackrel{def}{=} \nabla\nabla\Phi$ no direction change was needed in the reduction of $|\nabla\Phi|$ that worked efficiently.

Finally, for the case of $\alpha_x = 6.0$ the grid length has been doubled to $G = 12$. According to Fig. 3.18 it can be stated that the controller worked well, and the higher grid–length resulted in more considerable reduction in $|\nabla\Phi|$. It is easy to understand that the doubled horizon length results about doubled number of the components of approximately the same values in $\nabla\Phi$. For comparison with the operation of the *original algorithm* solving the equation $J^T(q)x = J^T(q)f(q)$ for $\alpha_x = 2.0$ the counterpart of Fig. 3.16 was provided in Fig. 3.19 for the same grid length. The original algorithm provided more precise results that is easy to understand: in Fig. 3.16 the squares of the eigenvalues of $J = \nabla\nabla\Phi$ were in the range $16 \sim 36$, while in Fig. 3.19 the maximal eigenvalue of $J^T J$ was about 35, therefore the original algorithm had the possibility to work with faster convergence in the case of the same parameter settings than the novel one.

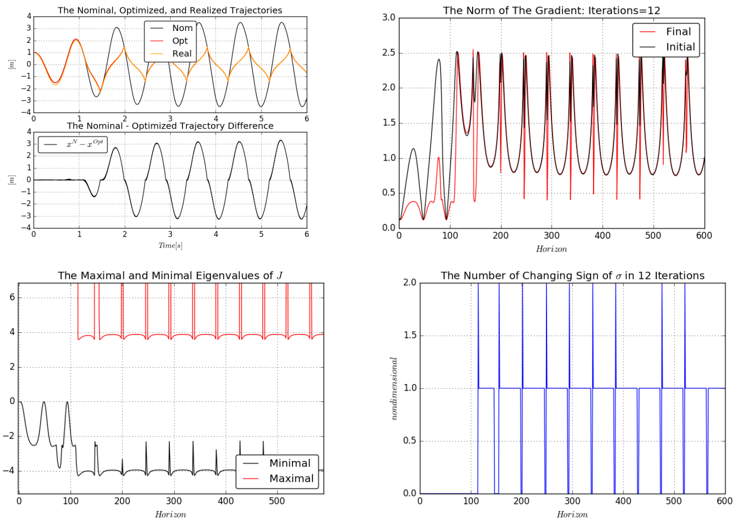

Figure 3.14: The results for $\alpha_x = 1.01$: tracking of the optimized trajectory, reduction of $|\nabla\Phi|$ (on top), the minimal and maximal eigenvalue of $\nabla\nabla\Phi$, and the number of changes in σ (on bottom)

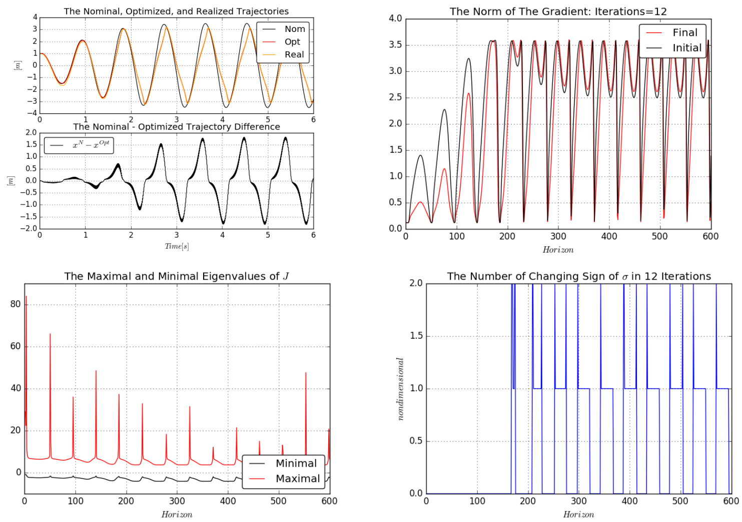

Figure 3.15: The results for $\alpha_x = 1.8$: tracking of the optimized trajectory, reduction of $|\nabla \Phi|$ (on top), the minimal and maximal eigenvalue of $\nabla\nabla\Phi$, and the number of changes in σ (on bottom)

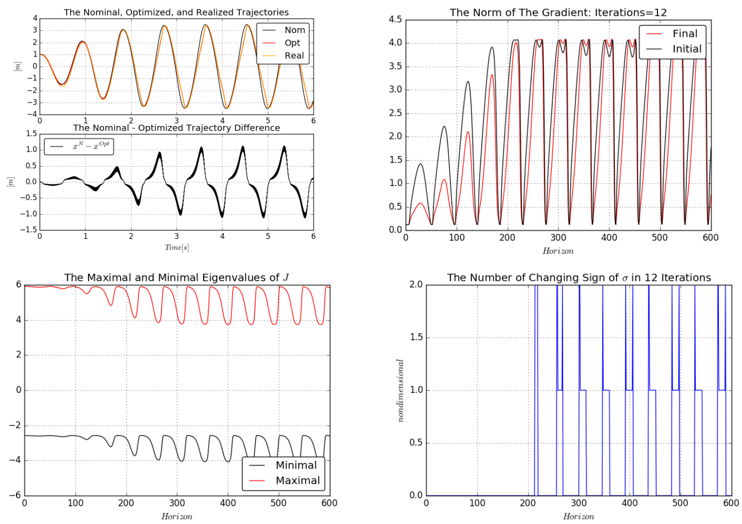

Figure 3.16: The results for $\alpha_x = 2.0$: tracking of the optimized trajectory, reduction of $|\nabla \Phi|$ (on top), the minimal and maximal eigenvalue of $\nabla\nabla\Phi$, and the number of changes in σ (on bottom)

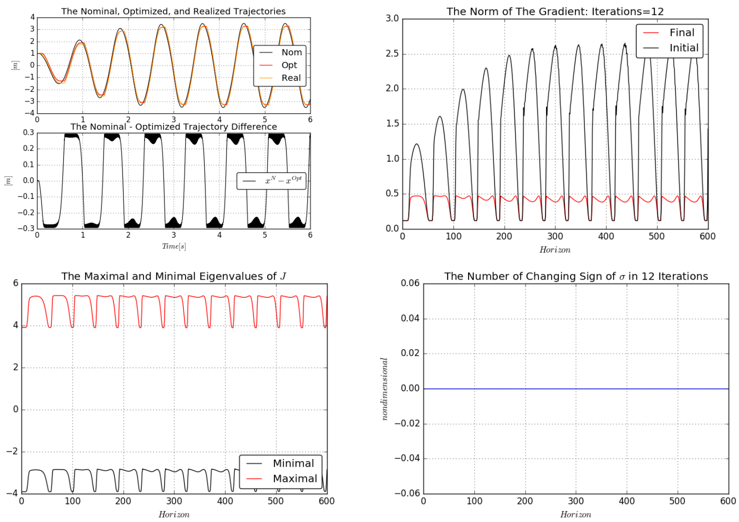

Figure 3.17: The results for $\alpha_x = 6.0$: tracking of the optimized trajectory, reduction of $|\nabla\Phi|$ (on top), the minimal and maximal eigenvalue of $\nabla\nabla\Phi$, and the number of changes in σ (on bottom)

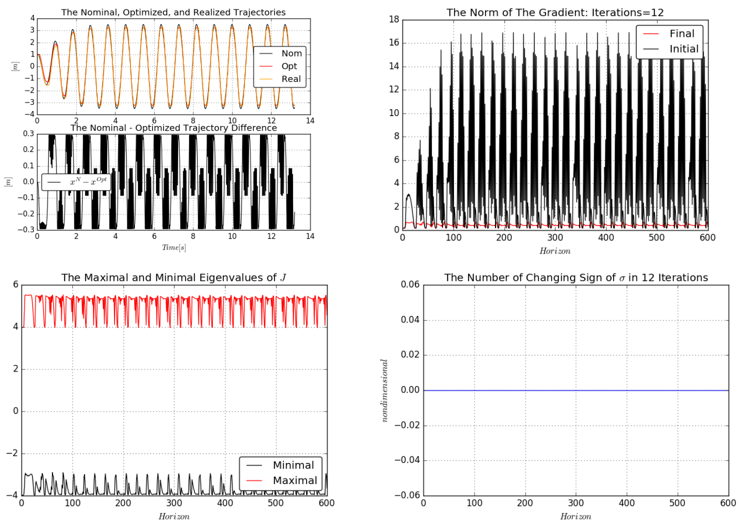

Figure 3.18: The results in the case of doubled grid length ($G = 24$) for $\alpha_x = 6.0$: tracking of the optimized trajectory, reduction of $|\nabla\Phi|$ (on top), the minimal and maximal eigenvalue of $\nabla\nabla\Phi$, and the number of changes in σ (on bottom)

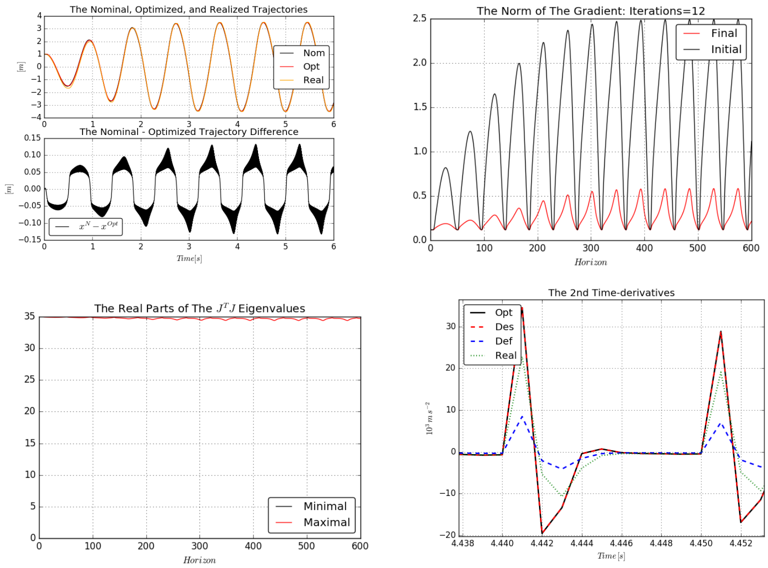

Figure 3.19: The results of the *original algorithm* solving the equation $J^T(q)x = J^T(q)f(q)$ for $\alpha_x = 2.0$: tracking of the optimized trajectory, reduction of $|\nabla\Phi|$ (on top), the minimal and maximal eigenvalue of $\nabla\nabla\Phi$, and excerpt of the variation of the 2$^{\text{nd}}$ time-derivatives (on bottom)

It worths noting that in the case of the really good results (i.e. that depicted in Figs. 3.17 and 3.18) the value of the parameter σ was constant in the internal iteration though the Jacobian of the problem had positive maximal and negative minimal eigenvalues that were comparable in their absolute values. Normally, increasing “sharpness” of the appropriate term in the cost function leads to the domination of one of the directions and good convergence can be achieved.

3.8 Thesis Statement II.

I have realized that there is a strict analogy between driving the gradient of the Auxiliary Function to zero in the Receding Horizon Controllers, and the novel, Fixed Point Transformation–based solution of the inverse kinematic task of robots. On this basis I suggested the replacement of the original Reduced Gradient Algorithm with the application of the Fixed Point Transformation–based approach to drive this gradient to zero. In this novel adaptive RHC the FPT–based solution is applied in two different levels: in finding the optimum, and in adaptively tracking the optimized trajectory calculated by the use of the available approximate dynamic model of the controlled system. The method has the “difficulty” that the constraint equations must be analytically expressed

before using the approximation over a discrete time–grid, and the Jacobian of the problem has to be computed, too.

3.8.1 Substatement II.1.

In this part I have introduced a new way based on the idea of driving Lagrange's Reduced Gradient (LRG) to zero where the numerically much more complex GRG method was replaced by a simple fixed–point transformation–based adaptive solution. It was also justified that it can easily be implemented in an arbitrary software environment for a wide class of problems in which the gradient of the "auxiliary function" as well as the gradient of this gradient can be determined in closed form formulation. The same type of fixed–point transformation was applied for driving the gradient of the auxiliary function and adaptively tracking of the optimized trajectory by the actual system. The applicability of the method was illustrated by presenting an example of a van der Pol oscillator and nonlinear dynamic paradigm, the Duffing oscillator. The method has the "difficulty" that the constraint equations must be analytically expressed before using the approximation over a discrete time–grid, and the Jacobian of the problem has to be computed, too. The simulations were made by a simple sequential code written in Julia language. It definitely can be stated that the theoretical expectations were verified by the simulations.

3.8.2 Substatement II.2

In the research concerned in this part I have further developed the main idea of the replacement of the original Reduced Gradient Algorithm with FPI procedure that directly drives the gradient of the Auxiliary Function of the optimization problem to zero. To investigate and validate the method a recent solution of the inverse kinematic task evading the calculation of the Jacobian was used. To make this procedure convergent, in the proposed solution for the calculation of the Jacobian only a rough numerical estimation was applied. Furthermore, it was realized that the convergence properties of the new algorithm can be improved by varying its presently established parameters that were experimentally set for the simulations. The method was presented and studied using numerical simulations for a strongly nonlinear, one degree of freedom, 2^{nd} order dynamical system, the van der Pol Oscillator, and 2 DoF 2^{nd} order nonlinear system that consists of two, nonlinearly coupled van der Pol oscillators. To guarantee lucid calculations simple functions were introduced that map the active parts of the horizon under consideration to the elements of the gradient of the auxiliary function that are calculated analytically. In general it can be concluded that the calculation or at least some good estimation

of the Jacobian can be spared only in very special cases.

Related own publications: [A. 3] [A. 4] [A. 5] [A. 6]

Chapter 4

FPT–based Adaptive Solution of the Inverse Kinematic Task of Robots

In a wide class of robots of open kinematic chain the inverse kinematic task cannot be solved by the use of closed–form analytical formulae. On this reason the traditional approaches apply differential approximation in which the Jacobian of the – normally redundant – robot arm is "inverted" by the use of some "generalized inverse". These pseudo-inverses behave well whenever the robot arm is far from a singular configuration, however, in the singularities and nearby the singular configurations they need the traditional inversion of singular or ill–conditioned quadratic matrices. For tackling the problem of singularities normally complementary "tricks" have to be used that so "deform" the original problem that the deformed version leads to the inversion of a well–conditioned matrix. Though the so obtained solution does not exactly solve the original problem, it is accepted as practical "substitute" of the not existing solution in the singularities, and an acceptable approximation of the exact solution outside the singular points.

Recently, in [78], an alternative, quasi–differential approach was suggested that was absolutely free of any matrix inversion. It was shown that it converged to one of the – normally ambiguous – exact solutions at the nonsingular configurations, and showed stable convergence in the singular points when a "substitute" of the not existing solution was created. This convenient convergence was guaranteed by the use of the "exact Jacobian" of the robot arm. The interesting question, i.e. what happens if only an "approximate Jacobian" is available, and the motion of the robot arm is precisely measurable with respect to a Cartesian "workshop"–based system of reference, was left open.

The scientific contents and novelty of the present investigations is based on the research I made in this interesting subject area.

4.1 Scientific Antecedents

The strict antecedents of the problem were considered in one of my papers [A. 7]. In general the inverse kinematic task of robots of open kinematic chain has only "differential solution" that is based on the use of some *"generalized inverse"* or *"pseudoinverse"* of the Jacobian of the arm. Let $q \in \mathbb{R}^n$, $n \in \mathbb{N}$ denote the joint coordinates of an n DoF open kinematic chain, and let $x \in \mathbb{R}^m$, $m \in \mathbb{N}$ be the array made of the Cartesian coordinates of certain points extended with the information on the pose of certain components with respect to the *"workshop frame"*. For the prescription of the motion of the arm the function $x(s)$, $s \in [s_i, s_f] \subset \mathbb{R}$ can be used in which s is a scalar parameter. In this manner a "line" is prescribed in $\mathbb{R}^m$ with the *initial* and *final* points at s_i and s_f, respectively. If $m > n$ no exact solution can be expected, but when $m < n$ the existence of ambiguous solutions is expected for a redundant arm.

The differential solution is provided by the Jacobian $J_{ij}(q) \overset{def}{=} \frac{\partial x_i}{\partial q_j}$ in equation (3.3) and the *initial condition* $x(s_i) = x_{ini}$ that normally is known. In the redundant case, when $m < n$, some "additional idea" is needed to choose one of the possible solutions. In the case of the *"Moore-Penrose Pseudoinverse"* [130, 131, 136] a "cost function" $\sum_k \left(\frac{\mathrm{d}q_k}{\mathrm{d}s}\right)^2$ is minimized under the constraint determined by equation (3.3) leading to equation (4.1) provided that JJ^T is invertible

$$\frac{\mathrm{d}q}{\mathrm{d}s} = J^T \left(JJ^T\right)^{-1} \frac{\mathrm{d}x(q)}{\mathrm{d}s} \quad . \tag{4.1}$$

Other generalized inverses based on the *Singular Value Decomposition (SVD)* [137] are not related to cost function minimization. The application of the *"Gram-Schmidt Algorithm"* (e.g. [138, 139]), originally invented by Laplace [140] in [141] also evaded the minimization of any cost function.

If JJ^T is not invertible, i.e. when J^T has non-empty *null space*, the problem is singular. In this case with the introduction of a small parameter $\mu > 0$ the substitute / approximate solution can be obtained as

$$\frac{\mathrm{d}q}{\mathrm{d}s} = J^T \left(JJ^T + \mu I\right)^{-1} \frac{\mathrm{d}x}{\mathrm{d}s} \quad , \tag{4.2}$$

since the appropriate matrix in it always has an inverse (I denotes the identity matrix) (e.g. [132]). If μ is much smaller than the smallest non–zero eigenvalue of JJ^T this means only little deformation far from the singularities, and provides finite $\frac{dq}{ds}$ values in the singularities (the so–called *"Damped Least Squares"* invented by Levenberg in 1944 [142]).

As alternative tackling of the problem of the singularities, Pohl suggested a set of 2nd order equations instead of the linear ones near the singularities [143]. Pohl and Lipkin in 1993 suggested complex extension of the generalized coordinates [144] for task deformation. Both methods have some drawbacks related to mathematical difficulties, and the latter has the additional problem of the physical interpretation of the complex joint coordinate values.

4.2 Adaptive Inverse Kinematics in the Possession of an Approximate Jacobian

We recapitulate 3.11 that was applied for the exactly known Jacobian with its counterpart using the available approximate Jacobian in 4.3 according to [77], in which in each step $n+1 \in \mathbb{N}$ the abstract rotation the rotational matrix $\mathcal{N}$ rotates back the vector into the direction of vector n. This abstract rotation has the role as follows: while Dineva's proof for convergence to the case considered in 3.9 and 3.10 required contractivity *for arbitrary direction*, in our case the convergence is needed only for the *initial direction* for which an appropriate adaptive parameter $A \in \mathbb{R}$ can be chosen.

$$\mathcal{N}(n+1)\check{J}^T(q)x^N(s) = \mathcal{N}(n+1)\check{J}^T(q)f(q(s)) \ \ , x^N(s_{ini}) = f(q_{ini}) \ \ . \tag{4.3}$$

It is easy to construct such an orthogonal transformation by the generalization of the Rodrigues formula published in 1840 [146]. In [147], for the purposes of adaptive dynamic control, a novel task transformation was suggested that used abstract rotations constructed as in (4.13). Assume, that we wish to transform the array $b \in \mathbb{R}^n$ into the array $a \in \mathbb{R}^n$, ($\|a\| \neq \|b\|$). A possible solution is augmenting the dimension n of the vectors to $n+1$ by adding to them a new orthogonal dimension $a \mapsto A \in \mathbb{R}^{n+1}$, $b \mapsto B \in \mathbb{R}^{n+1}$ so that $\|A\| = \|B\| = R_a$ is a common absolute value. Then, according to Fig. 4.1, the rotation that rotates the array B into A can be constructed. As a consequence, the projection of the rotated vector in the original space will behave accordingly, i.e. b will be moved into a with simultaneous rotation and shrink or dilatation.

Consider the vectors $a, b \in \mathbb{R}^m$, $m \in \mathbb{N}$. At first remove the component parallel to b from a with parameter λ in the form: $a^{Mod} = a + \lambda b$ so that a^{Mod} must be orthogonal to b, that means for the scalar product that $b^T a^{Mod} = b^T a + \lambda b^T b = 0$.

This leads to $\lambda = \frac{-b^T a}{b^T b}$. Then consider the *pairwisely orthogonal* unit vectors $e_a = \frac{a^{Mod}}{\|a^{Mod}\|}$, and $e_b = \frac{b}{\|b\|}$. The *skew symmetric matrix* $G \stackrel{def}{=} e_a e_b^T - e_b e_a^T$ generates rotations that mix the components of the vectors only in the two dimensional hyperplane spanned by these unit vectors. With a parameter $\xi \in \mathbb{R}$ these rotations have the form $\mathcal{O} = \exp(\xi G) \stackrel{def}{=} \sum_{s=0}^{\infty} \frac{\xi^s G^s}{s!}$. This matrix can be expressed in closed analytical form in similar manner as the Rodrigues formula [146] can be used for expressing 3D rotations around a given axis. Consider the various powers of G by taking into account that $e_a^T e_b = 0$, $e_a^T e_a = 1$, and $e_b^T e_b = 1$:

$$\begin{gathered} G^2 = \left(e_a e_b^T - e_b e_a^T\right)\left(e_a e_b^T - e_b e_a^T\right) = -e_a e_a^T - e_b e_b^T \quad, G^3 = -\left(e_a e_a^T + e_b e_b^T\right) \\ \left(e_a e_b^T - e_b e_a^T\right) = -\left(e_a e_b^T - e_b e_a^T\right) = -G, G^4 = -G^2, G^5 = -G^3 = G \quad \text{,etc.} \end{gathered} \tag{4.4}$$

By selecting the *even* and the *odd powers* of G it is obtained that

$$\mathcal{O} = I + \sin\xi G + (1 - \cos\xi) G^2 \quad . \tag{4.5}$$

The direction of the rotation can be found from the scalar product $(\mathcal{O}a)^T b = a^T \mathcal{O}^T b = \|a\| \cdot \|b\| \cos\xi$. Due to the symmetry of the $\cos x$ function it can be stated that $\xi_{1,2} = \pm \text{acos}\left(\frac{a^T \mathcal{O}^T b}{\|a\| \cdot \|b\|}\right)$. For finding the appropriate solution the greater $a^T \mathcal{O}(\xi_{1,2})^T b$ value must be selected for rotations of small $|\xi|$.

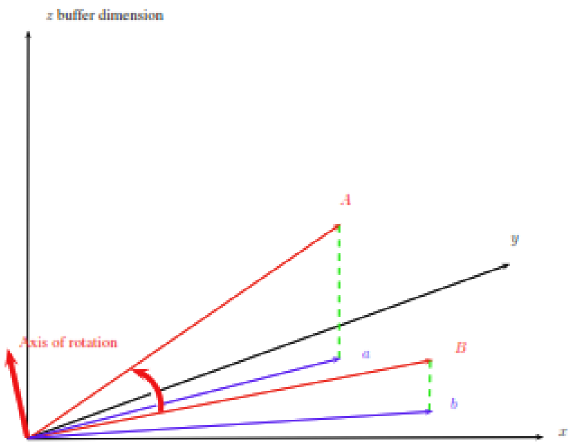

Figure 4.1: Schematic visualization of 2D rotations with a complementary buffer dimension (cited from [147])

The use of this scheme for adaptive control is quite simple: in the iteration in the previous step we observed that we need a rotation that transforms vector B into

A, and this rotation has to be applied for a new vector $C \in \mathbb{R}^{n+1}$ where C is the augmented version of vector $c \in \mathbb{R}^n$. The angle of this new rotation can $\lambda_a \in \mathbb{R}$ times of the original rotation. It is very easy to understand the operation of this method.

4.2.1 Discussion of the Novelties and Simulation Results

In contrast to the solution used earlier, in the present investigation I applied the abstract rotations–based adaptive transformation in combination with the rotations $\mathscr{N}$ in (4.6).

$$J^T(q)x^N(s) = J^T(q)f(q(s)) \quad , x^N(s_{ini}) = f(q_{ini}) \quad , \tag{4.6}$$

The kinematic construction of the 8 DoF redundant robot arm was modified, too, as follows: The open kinematic chain under consideration was described by the product of 8 homogeneous matrices as

$$\begin{bmatrix} r \\ 1 \end{bmatrix} = H^{(1)}\left[\xi_1, e^{(1)}, L^{(1)}\right] \cdots H^{(n-1)}\left[\xi_{n-1}, e^{(n-1)}, L^{(n-1)}\right] H^{(n)}\left[q_n, e^{(n)}, L^{(n)}\right] \begin{bmatrix} \tilde{r} \\ 1 \end{bmatrix}$$

$$\begin{pmatrix} r \\ 1 \end{pmatrix} = H^{(1)}(q_1)H^{(2)}(q_2)\cdots H^{(8)}(q_8) \begin{bmatrix} \tilde{r} \\ 1 \end{bmatrix} = H(q_1, \ldots, q_8) \begin{bmatrix} \tilde{r} \\ 1 \end{bmatrix} \quad , \tag{4.7}$$

in which $\tilde{r} \in \mathbb{R}^3$ is vector of the last segment in the *"home position"* with respect to the last local system of coordinates (its graphical representation can be seen in figure 4.2), i.e. $\tilde{r}$ is constant, $H^{(i)}(q_i) \in \mathbb{R}^{4\times4}$ is the homogeneous matrix of the i^{th} segment, the upper left block of $H^{(i)}$ of size $\mathbb{R}^{3\times3}$, $O(e^{(i)}, q_i)$ is a rotational matrix that rotates around the unit vector $e^{(i)}$ with angle q_i (it is expressed by the use of the Rodrigues formula [146]), and its 4^{th} column is a shift parameter in the form $(r^{(i)T}, 1)^T \in \mathbb{R}^4$. Since the homogeneous matrices form a Lie group, $H(q_1, \ldots, q_8)$ is a homogeneous matrix, too. Its upper left block of size $\mathbb{R}^{3\times3}$ is a rotational matrix that describes the "pose" of the last segment, and $r \in \mathbb{R}^3$ is the location of the endpoint with respect to the workshop reference frame.

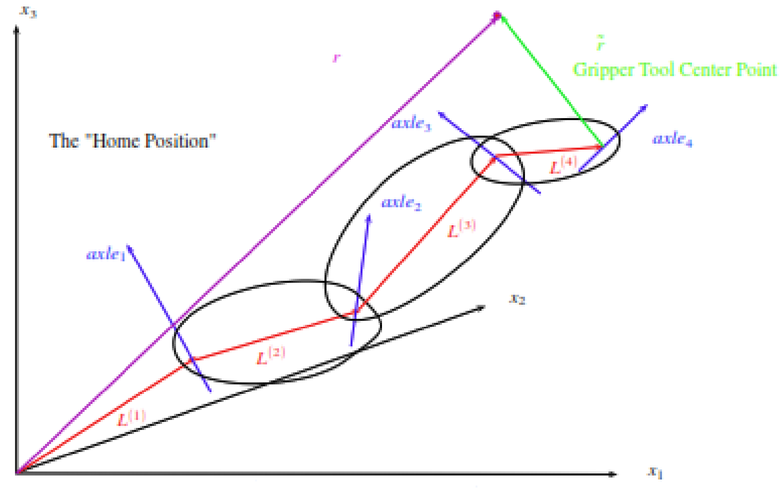

Figure 4.2: Cartesian System of Coordinates fixed at the Workshop

The unit vectors of the home position of *the approximate ("canonical") model* as well as the shift parameters can be placed in the columns of size 3×8 matrices in which each column belongs to an arm segment (link) as follows:

$$\check{E} \stackrel{def}{=} \begin{pmatrix} 0 & 0 & 0 & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{3}} \\ 0 & 1 & 1 & 0 & 0 & 1 & \frac{1}{\sqrt{3}} & \frac{-1}{\sqrt{3}} \\ 1 & 0 & 0 & \frac{-1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{3}} \end{pmatrix}, \tag{4.8}$$

while the shift parameters were

$$\check{R} \stackrel{def}{=} \begin{pmatrix} 0 & \check{L}_1 & \check{L}_1 & \check{L}_1 & \check{L}_3 & \check{L}_3 & \check{L}_1 & \check{L}_3 \\ 0 & 0 & 0 & 0 & 0 & \check{L}_3 & -\check{L}_2 & \check{L}_2 \\ \check{L}_1 & 0 & 0 & \check{L}_2 & \check{L}_3 & 0 & \check{L}_3 & -\check{L}_1 \end{pmatrix} \tag{4.9}$$

with $\check{L}_1 = 1.0\,[m]$, $\check{L}_2 = 1.5\,[m]$, and $\check{L}_3 = 2.0\,[m]$. The exact unit vectors are the rotated versions of the available *approximate ones* in the columns of (4.8) in a matrix $\check{E}$. The rotational angles of the units vectors around the workshop axles (φ_1 around X_1, φ_2 around X_2, and φ_3 around X_3) are given in Table 4.1.

The counterpart of the approximate matrix $\check{R}$ in (4.10) is the *exact one* as

$$R \stackrel{def}{=} \begin{pmatrix} 0 & L_1 & L_1 & L_1 & L_3 & L_3 & L_1 & L_3 \\ 0 & 0 & 0 & 0 & 0 & L_3 & -L_2 & L_2 \\ L_1 & 0 & 0 & L_2 & L_3 & 0 & L_3 & -L_1 \end{pmatrix} \tag{4.10}$$

with $L_1 = 1.2\,[m]$, $L_2 = 1.8\,[m]$, and $L_3 = 2.2\,[m]$. The *approximate value of the last segment* was the "canonical" $\check{\check{r}} = [2.5, 2.5, 2.5]\,[m]$ vector of equal components,

Table 4.1: The rotations of the unit vectors of the rotational axles for the exact model correspond to the rotated version of the approximate ones as $[O = O_1(\varphi_1)O_2(\varphi_2)O_3(\varphi_3)]$, $e^{(i)} = O^{(i)}\breve{e}^{(i)}$

Rotational angle	$\varphi_1\,[rad]$	$\varphi_2\,[rad]$	$\varphi_3\,[rad]$
For $e^{(1)}$:	2×0.10	2×0.09	2×0.08
For $e^{(2)}$:	2×0.02	2×0.02	2×0.0
For $e^{(3)}$:	2×0.02	2×0.03	2×0.07
For $e^{(4)}$:	2×0.08	2×0.06	2×0.04
For $e^{(5)}$:	2×0.05	2×0.01	2×0.07
For $e^{(6)}$:	2×0.03	2×0.01	2×0.06
For $e^{(7)}$:	2×0.06	2×0.06	2×0.08
For $e^{(8)}$:	2×0.04	2×0.01	2×0.10

while the *exact one* was $\tilde{r} = [2.6, 2.4, 2.3]\,[m]$ that *inevitably causes tracking error in the initial position* that later relaxes. For better relaxation in the first 10 discrete time–point 60 steps of the numerical iteration was applied, and later only 10 steps.

Furthermore, regarding the problem solution, (4.3) was further modified in (4.11) as

$$\mathscr{W}\mathscr{N}(n+1)\breve{J}^T(q)\mathscr{F}x^N(s) = \mathscr{W}\mathscr{N}(n+1)\breve{J}^T(q)\mathscr{F}f(q(s)) \quad , \tag{4.11}$$

in which $\mathscr{F}$ and $\mathscr{W}$ are *diagonal matrices of positive,* $0 < \mathscr{F}_{ii}, \mathscr{W}_{ii} \leq 1$ *elements*. The role of $\mathscr{F}$ is weighting the relative significance of the rotational pose and the location of the end–point in the solution. (We remind that f has 9 redundant components for the pose, and only 3 ones for the location.) The role of $\mathscr{W}$ is weighting the relative activities of the redundant joint coordinates in the disambiguation of the generally ambiguous solution. The *generator of the rotation operator rotating around an axle the direction of which is described by the unit vector e* expressed in a *right handed system of coordinates* is:

$$G = e_1 \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix} + e_2 \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{bmatrix} + e_3 \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} =$$
$$= \begin{bmatrix} 0 & -e_3 & e_2 \\ e_3 & 0 & -e_1 \\ -e_2 & e_1 & 0 \end{bmatrix} \quad , e_1^2 + e_2^2 + e_3^2 = 1 \quad .$$

The Rodrigues formula for a constant given unit vector of rotary axle $e = [e_1, e_2, e_3]^T$ and a variable rotational angle ξ has the simple analytical form (4.13):

$$O(\xi, e) = \exp(\xi G(e)) = I + \begin{bmatrix} 0 & -e_3 & e_2 \\ e_3 & 0 & -e_1 \\ -e_2 & e_1 & 0 \end{bmatrix} \left(\xi - \frac{\xi^3}{3!} + \frac{\xi^5}{5!} \mp \ldots \right) +$$
$$+ \begin{bmatrix} e_1^2 - 1 & e_1 e_2 & e_1 e_3 \\ e_2 e_1 & e_2^2 - 1 & e_2 e_3 \\ e_3 e_1 & e_3 e_2 & e_3^2 - 1 \end{bmatrix} \left(\frac{\xi^2}{2!} - \frac{\xi^4}{4!} + \frac{\xi^6}{6!} \mp \ldots \right) = I +$$
$$+ \begin{bmatrix} 0 & -e_3 & e_2 \\ e_3 & 0 & -e_1 \\ -e_2 & e_1 & 0 \end{bmatrix} \sin \xi + \begin{bmatrix} e_1^2 - 1 & e_1 e_2 & e_1 e_3 \\ e_2 e_1 & e_2^2 - 1 & e_2 e_3 \\ e_3 e_1 & e_3 e_2 & e_3^2 - 1 \end{bmatrix} (1 - \cos \xi) \tag{4.13}$$

The appropriate *homogeneous matrix representation* is:

$$\exp\left(\xi \left[\begin{array}{c|c} G & 0 \\ 0^T & 0 \end{array} \right] \right) = \left(\left[\begin{array}{c|c} \exp(\xi G) & 0 \\ 0^T & \exp(0) \end{array} \right] \right) = \left[\begin{array}{c|c} O & 0 \\ 0^T & 1 \end{array} \right]$$

The derivatives of the appropriate *homogeneous matrix* according to their rotational angles and inverses in (4.7) can be constructed in block form built up of the rotational matrices constructed according to (4.13) and the constant shift components of the home position denoted by L as in (4.14):

$$H = \left[\begin{array}{c|c} O(\xi, e) & L \\ 0^T & 1 \end{array} \right] \quad , \frac{\mathrm{d}H}{\mathrm{d}\xi} = \left[\begin{array}{c|c} \frac{\mathrm{d}O(\xi,e)}{\mathrm{d}\xi} & 0 \\ 0^T & 0 \end{array} \right] \quad , H^{-1} = \left[\begin{array}{c|c} O^{-1}(\xi, e) & -O^{-1}(\xi, e) L \\ 0^T & 1 \end{array} \right] \quad ,$$
$$\frac{\mathrm{d}H}{\mathrm{d}\xi} H^{-1} = \left[\begin{array}{c|c} \frac{\mathrm{d}O(\xi,e)}{\mathrm{d}\xi} O^{-1}(\xi, e) & -\frac{\mathrm{d}O(\xi,e)}{\mathrm{d}\xi} O^{-1}(\xi, e) L \\ 0^T & 0 \end{array} \right] = \left[\begin{array}{c|c} \Omega(\xi, e) & -\Omega(\xi, e) L \\ 0^T & 0 \end{array} \right] \quad , \tag{4.14}$$

in which $\Omega = \frac{\mathrm{d}O(\xi,e)}{\mathrm{d}\xi} O^{-1}(\xi, e)$ is a skew–symmetric matrix, i.e. a generator of the rotational matrices (an element of the tangent space of the rotational group at the identity element). Finally the Jacobian of the inverse kinematic task can be formulated by finding the coefficients in the linear combination of the actual tangent vectors of the SE(3) Lie group at its identity element that must be identical with the tangent vector determined by the desired motion:

Development of the differential formulae by using the *chain rule, and the inverse matrix*:

$$\begin{bmatrix} \dot{r} \\ 0 \end{bmatrix} = \left(\dot{\xi}_1 \frac{\mathrm{d}H^{(1)}}{\mathrm{d}\xi_1} H^{(2)} \cdots H^{(n)} + \dot{\xi}_2 H^{(1)} \frac{\mathrm{d}H^{(2)}}{\mathrm{d}\xi_2} H^{(3)} \cdots H^{(n)} + \ldots + \right. \\ \left. + \dot{\xi}_n H^{(1)} \cdots H^{(n-1)} \frac{\mathrm{d}H^{(n)}}{\mathrm{d}\xi_n} \right) \begin{bmatrix} \tilde{r} \\ 1 \end{bmatrix} , \tag{4.15}$$

$$\begin{bmatrix} \dot{r}(t) \\ 0 \end{bmatrix} = \left(\dot{\xi}_1 \frac{\mathrm{d}H^{(1)}}{\mathrm{d}\xi_1} H^{(1)^{-1}} + \dot{\xi}_2 H^{(1)} \frac{\mathrm{d}H^{(2)}}{\mathrm{d}\xi_2} H^{(2)^{-1}} H^{(1)^{-1}} + \ldots \right. \\ \left. + \dot{\xi}_n H^{(1)} H^{(2)} \cdots H^{(n-1)} \frac{\mathrm{d}H^{(n)}}{\mathrm{d}\xi_n} H^{(n)^{-1}} H^{(n-1)^{-1}} \cdots H^{(2)^{-1}} H^{(1)^{-1}} \right) \begin{bmatrix} r(t) \\ 1 \end{bmatrix} , \tag{4.16}$$

in which the $\frac{\mathrm{d}H^{(i)}}{\mathrm{d}\xi_1} H^{(i)^{-1}}$ expressions are the tangents at the identity element, $H^{(1)} \left(\frac{\mathrm{d}H^{(2)}}{\mathrm{d}\xi_2} H^{(2)^{-1}} \right) H^{(1)^{-1}}$ is the 2^{nd} tangent vector transformed by the group element $H^{(1)}$, therefore it is also a tangent at the identity element of the group SE(3), etc.

The physical interpretation of the tangent of the Lie group of the homogeneous matrices at the identity element in the differential equations can be expressed as;

$$\frac{\mathrm{d}H^{(i)}}{\mathrm{d}\xi_i} H^{(i)^{-1}} \quad , \left(\left[\begin{array}{c|c} O & L \\ 0^T & 1 \end{array} \right]^{-1} = \left[\begin{array}{c|c} O^{-1} & -O^{-1}L \\ 0^T & 1 \end{array} \right] \right) \; .$$

For an arbitrary homogeneous matrix H

$$H \left(\frac{\mathrm{d}H^{(i)}}{\mathrm{d}\xi_i} H^{(i)^{-1}} \right) H^{-1}$$

is a tangent, too! Since the tangent space of a Lie group at the identity element is *a linear space* we obtain a simple set of linear equations to be solved $\forall r(t)$: *the element of a linear space $G(t)$ has to be found as the linear combination of certain elements $G^{(i)}(\xi(t))$ of the same linear space:*

$$\begin{bmatrix} \dot{r}(t) \\ 0 \end{bmatrix} = \sum_{i=1}^{n} \dot{\xi}_i G^{(i)}(\xi(t)) \begin{bmatrix} r(t) \\ 1 \end{bmatrix} = G(t) \begin{bmatrix} r(t) \\ 1 \end{bmatrix} \; .$$

In the Julia program the above matrices can be calculated in closed form, and the elements of the upper 3×3 block, and the $(4, 1)$, $(4, 2)$, $(4, 3)$ elements can be arranged in the 12 rows of the Jacobian having 8 columns. From this point on the traditional matrix operations (e.g. SVD or calculation of the Moore–Penrose pseudoinverse) can be applied for solving the redundant set of linear equations. In my case, the adaptive function detailed above can be called *within an internal cycle for each discrete point of the trajectory*. in which the program variable `W` stands for $\mathscr{W}$, and `F` corresponds to $\mathscr{F}$.

4.2.1.1 Initial Tests

In the first step initial tests were made to check the operation of the algorithm. In these tests one had trivial expectations for the nominal trajectory and its tracking. To check the operation of the algorithm in the first step the *approximate, canonical model* was used for the generation of the nominal trajectory to be tracked. According to the canonical model X_3^N in the Cartesian coordinates must be constant since the rotation happens around an axis parallel to the vertical one of the workshop frame. Furthermore, the last link's pose suffers rotation around an axis parallel to X_3 of the workshop's frame of reference. In this case $\mathscr{F}$ and $\mathscr{W}$ were the identity matrices, i.e. no any weighting was applied. The results are given in Figs. 4.3 and 4.4 that correspond to the expectations. The inevitable initial tracking error rapidly decreases and the orientation error is small, too. The solution in the joint coordinates of the robot are given in Fig. 4.5. The significance of the stabilizing "counter–rotation" and that of the abstract rotations applied in the FPI–based iteration are given in Fig. 4.6 for $R_a = 100$ "abstract radius" and $\lambda_a = 5 \times 10^{-4}$ extrapolation parameter. The resolution of the scalar parameter s was 10^{-3}.

In the next run for $i = 1 : 9$ the $\mathscr{F}_{ii}$ elements were reduced from 1 to 0.5. The fine details of the trajectory tracking in Fig. 4.7 can be compared with that in Fig. 4.4. The orientation precision really was degraded, and this effect shows some coupling with the tracking error of the position of the endpoint. Also, in Fig. 4.8 subtle differences appear in the joint coordinated of solution in comparison with Fig. 4.5.

In the next run $\mathscr{F} = I$ was restored and the last two diagonal elements in $\mathscr{W}$ were decreased to 0.01 to reduce the motion of the last two redundant joint coordinates q_7 and q_8. According to Fig. 4.9 the tracking precision remained good, and in Fig. 4.10 it can be seen that q_7 and q_8 were really "blocked".

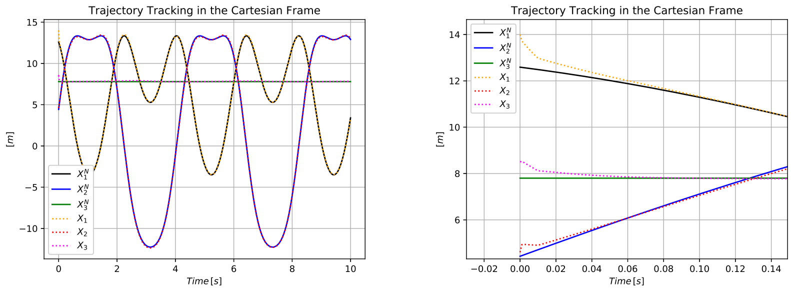

Figure 4.3: Tracking of a nominal trajectory generated by using only q_1 in the canonical approximate model

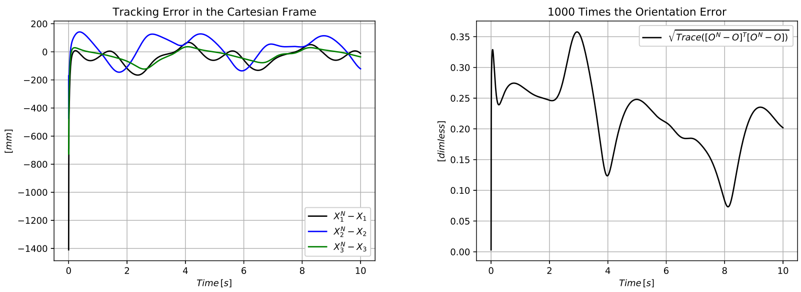

Figure 4.4: The tracking error of the end-point and the orientation for the nominal trajectory generated by using only q_1 in the canonical approximate model

4.3 Solution of Inverse Kinematic Problems without the Calculation of Jacobian

In the application of technical problems sometimes very typical issues arise regarding their solutions which can be "differentially" treated by the calculation of Jacobian and subsequent inversion of Jacobi matrices. On the other side, in many cases, the calculation of the inverse of the Jacobian gives a simple meaning which is not necessarily a computation based solution. In such cases the solution is interesting only for a provided and well elaborated input array. However, for an arbitrary input, it can be obtained by the use of the (generalized) inverse. In this case a recently defined quasi-differential solution of the inverse kinematic task of robots provides a very nice and fully stable behavior in and in the vicinity of the kinematic singularities where the classical matrix-inversion-based approaches have many difficulties and need complementary tricks to remain stable. It can be stated that in case of the redundant robots the said approach required the tricky calculations of the Jacobian. To avoid such tricky tasks, in this part, I have suggested and tried to give the solution that in the special case of quadratic

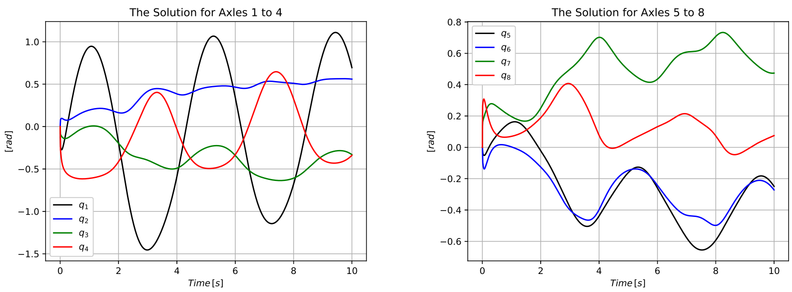

Figure 4.5: The solution in the space of the joint coordinates for the nominal trajectory generated by using only q_1 in the canonical approximate model

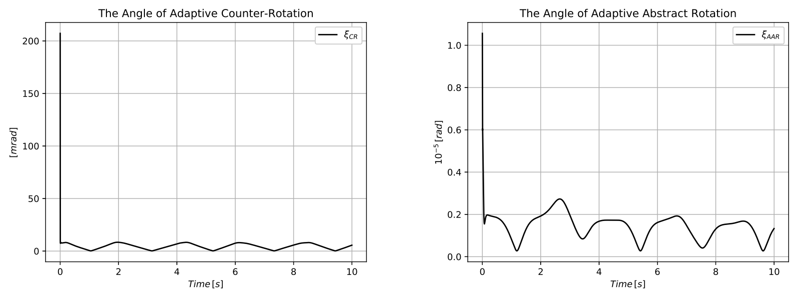

Figure 4.6: The angle of the “stabilizing rotation” $\mathscr{N}$ and the “abstract rotations” of the FPI-based algorithm for the nominal trajectory generated by using only q_1 in the canonical approximate model

Jacobian its calculation can be omitted.

This approach was extended by the motivation of section 3 where the replacement of the Lagrange’s original Reduced Gradient Method was discussed by suggesting with a simple Fixed Point Iteration for the gradient of the auxiliary function of the optimization under constraints. To elaborate and justify the suggested method a simple example 2 degree of freedom arm is considered via simulations to reveal the behaviour of this approach. This approach was applied for a simple 2 degree of freedom robot and enhanced the results in section 4.5. It has been shown that by the use of a simple complementary norm reduction built in into the solution this approach is promising.

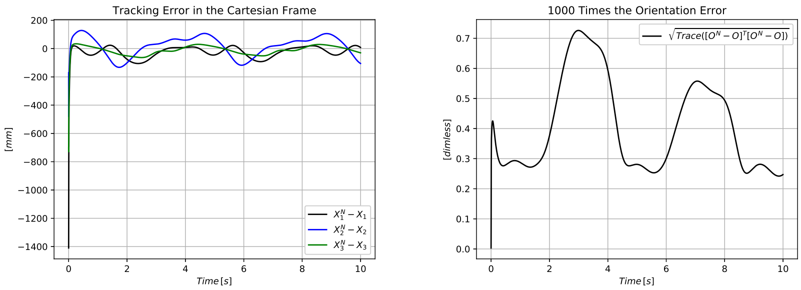

Figure 4.7: The tracking error of the end-point and the orientation for the nominal trajectory generated by using only q_1 in the canonical approximate model with reduced precision of the orientation

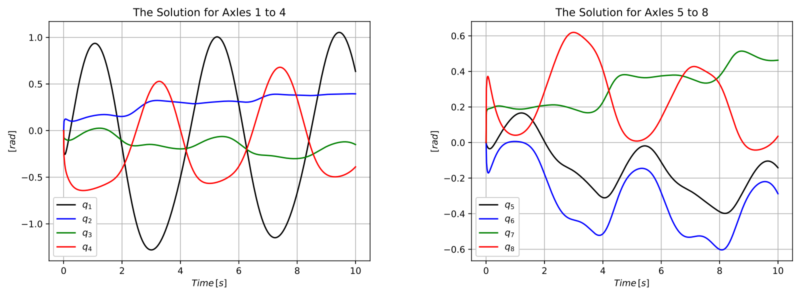

Figure 4.8: The solution in the space of the joint coordinates for the nominal trajectory generated by using only q_1 in the canonical approximate model with reduced precision of the orientation

4.4 Critical Observations Concerning The Original Fixed Point Transformation-based Approach

The above considerations can suffer from certain critics as follows:

1. The requirement that each $\Re(\lambda_i)$ must have the same sign is *"too rigorous"*, i.e. it is *satisfactory* but not *necessary* for the convergence. They can guarantee that the norm of $q_{i+1} - q_\star$ for an *arbitrary direction* of the previous $q_i - q_\star$ will be smaller than $\|q_i - q_\star\|$. However, this is not necessary: it is just enough to guarantee the contractivity for those directions that *actually occur* in the given task. So intuitively it can be expected that less rigorous conditions can guarantee the convergence, too.

2. Though it is a great advantage that the *generalized inverse* of the given Jacobian need not be computed, it would be more convenient to get rid of the

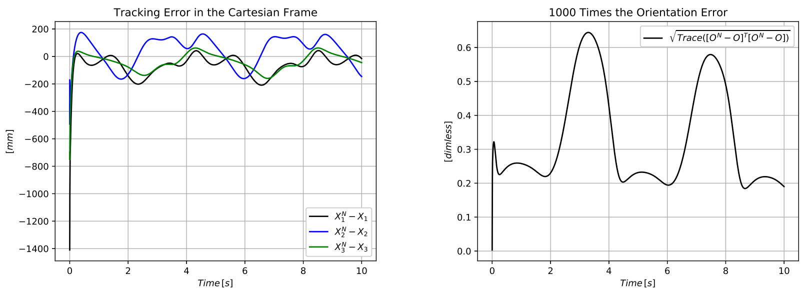

Figure 4.9: The tracking error of the end-point and the orientation for the nominal trajectory generated by using only q_1 in the canonical approximate model with reduced motion of q_7 and q_8

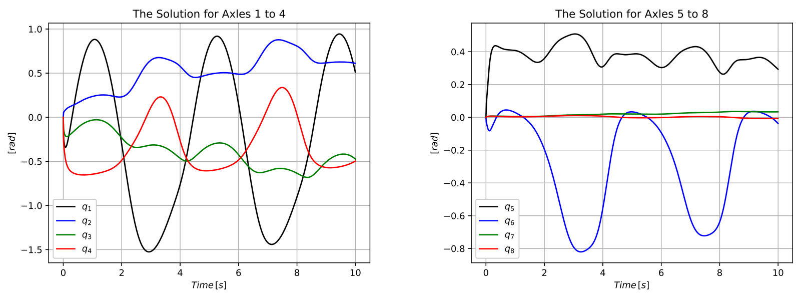

Figure 4.10: The solution in the space of the joint coordinates for the nominal trajectory generated by using only q_1 in the canonical approximate model with reduced motion of q_7 and q_8

computation of the Jacobian itself. This motivation is originated not only from the side of the inverse kinematic problems in robotics. In [81, 82] the same Fixed Point Iteration was used for replacing Lagrange's original Reduced Gradient Method in a non-linear programming-based solution of Receding Horizon Controllers (e.g. [7]) in which neither quadratic cost functions, nor Linear Time-invariant (LTI) system models are assumed as in the case of the Linear Quadratic Regulator (e.g. [148]). The idea was based on the observation that the *"Auxiliary Function"* of the optimization problem

$$\mathbb{R} \ni \Phi(x,\lambda) \stackrel{def}{=} f(x) + \sum_{s=1}^{k} \lambda_s g^{(s)}(x) \quad , \tag{4.17}$$

in which $f : \mathbb{R}^n \mapsto \mathbb{R}$ is the differentiable cost function to be minimized under the constraints expressed by the functions $g^{(s)} : \mathbb{R}^n \mapsto \mathbb{R}$ and equations $g^{(s)}(x) = 0$. It is well known that the gradient of Φ is zero at the solution of the optimization task. The idea was that the vector function $\nabla\Phi(x,\lambda) : \mathbb{R}^{n+k} \mapsto \mathbb{R}^{n+k}$ was driven towards 0 by a simple Fixed Point Iteration-based algorithm. In this case the computation of the Jacobian of $\nabla\Phi$ was very laborious and it is an interesting question whether this step can be eliminated.

In this research the suggested method is based on the simple idea which has already been discussed under the section 3.6 and the basic equation we used was also explained in equation 3.12. In the simulations presented in the next section a simple 2 DoF robot arm model will be considered with a simple arm structure that realizes a motion within a plain. Since the simulations revealed great jumps in the joint coordinates when the solution was ambiguous, a parameter $\Delta q_{max} = 8.0 \times 10^{-3}\, rad$ was introduced in a "smoothed solution" to limit the allowed joint speeds in the solutions in the form $\Delta q_{smoothed} = \Delta q_{max} \tanh(\Delta q/\Delta q_{max})$.

4.5 Simulations

In the simulations equal link lengths of $L_1 = L_2 = 1.0\,m$ were set. The trajectory to be tracked were limited in their amplitudes as $\pi/2$ for q_1 and π for q_2. The nominal motion $q_1^N(t)$ was simple sinusoidal one, for $q_2^N(t)$ a third power of the sinusoidal motion was applied. The parameter of the so obtained curve was closed in the interval $s \in [0,1]$ that was divided into 10^4 grid points. The adaptive function in (2.12) was $F(x) = \mathrm{atanh}(\tanh(x+D)/2)$ with $D = 0.3$, and the adaptive parameter was $A = -3.0$. In the internal loop 15 steps of iteration was done for making a step between two grid-points. Figure 4.11 reveals that in each case the "generating joint coordinates" were not identical to the "computed" ones that is a natural consequence of the fact that the solution is ambiguous. It is also revealed that the single drastic jump in the joint coordinates was well "distributed" or softened due to the smoothing. Figures 4.12 and 4.13 confirm that in the Cartesian coordinates the nominal trajectory was well tracked by the simplified method with the exception of the "critical point" in which a great jump occurred in the joint coordinates. It is also shown that by "smoothing" this tracking error is reduced and distributed over a wider range.

According to Fig. 4.14 (top) it can be stated that for achieving convergence approximately at one half of the number of internal iterations the parameter σ changed sign if no "smoothing" was applied (maximum 8 times of the 15 steps). Figure 4.14 (bottom) testifies that in the case of "smoothing" the limitation set by the parameter Δq_{max} made "serious limitations" only at a few points. While without smoothing the "region" in which many changes was necessary in the sign of σ was left after max. 8 steps, the applied "smoothing" kept the solution within the zone in which frequent changes (maximum 15) were necessary. In Figs. 4.15 (top) and 4.15 (bottom) "the smoothed" solutions are displayed in which $L_1 = 1.0m$ was fixed but the second link was longer as $L_2 = 1.5m$ and shorter as $L_2 = 0.5m$ than in the former examples. The other control and simulation parameters remained invariant. The figures show an operation similar to the symmetric $L_1 = L_2 = 1.0m$ case.

Figure 4.16 reveals that drastic increase in the number of the internal steps of iteration slightly can improve the solution.

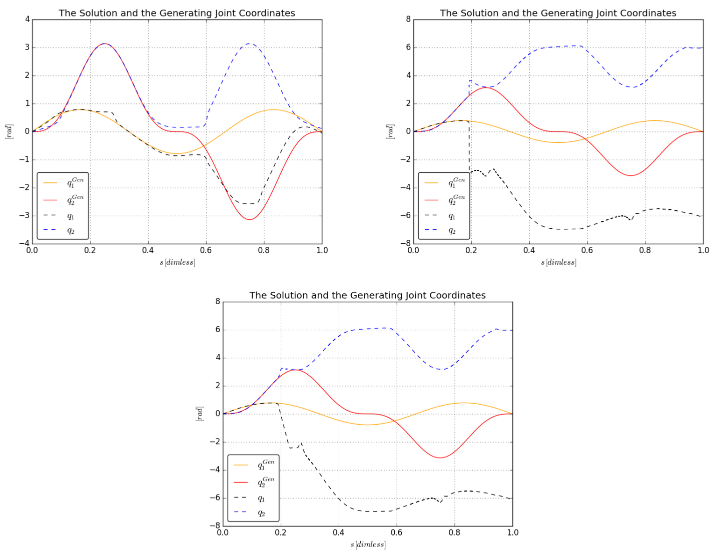

Figure 4.11: The joint coordinates of the trajectory generation and that of the numerical solutions (at the top LHS, the original solution that utilizes J,on the RHS the present solution without "smoothing", on the bottom with "smoothing")

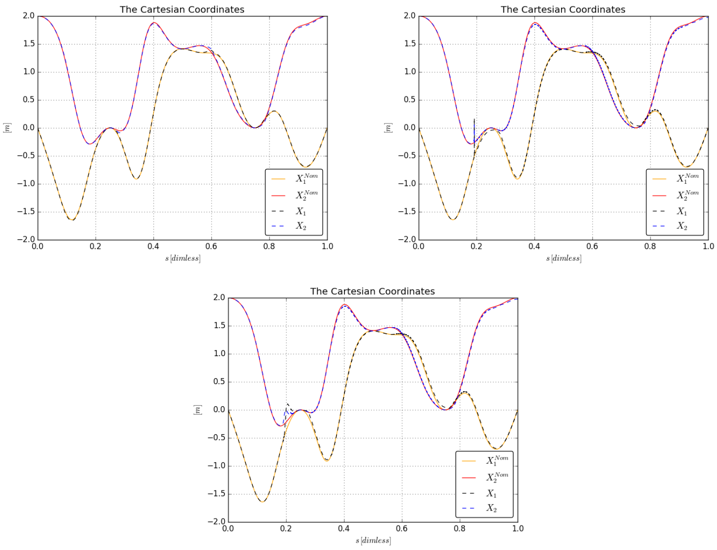

Figure 4.12: The Cartesian coordinates of the trajectory generation and that of the numerical solutions (at the top LHS the original solution that utilizes J, on the RHS the present solution without “smoothing”, and on the bottom with “smoothing”)

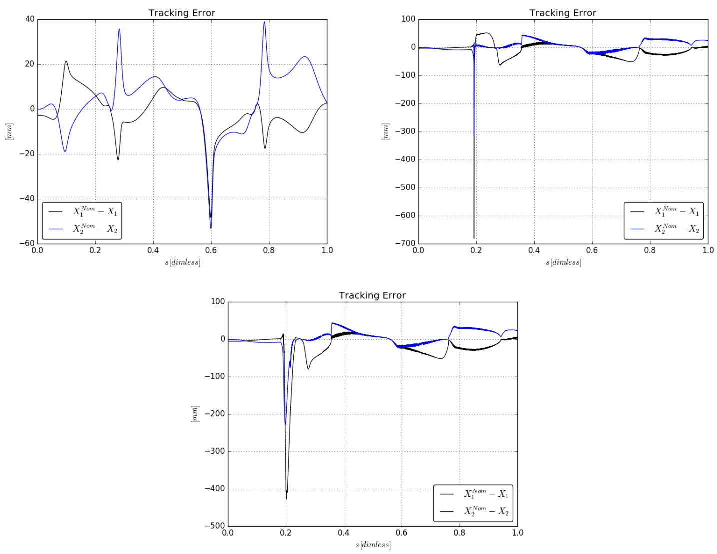

Figure 4.13: The tracking errors in the Cartesian coordinates of the numerical solutions (at the top LHS the original solution that utilizes J, on the RHS the present solution without “smoothing”, and on the bottom with “smoothing”)

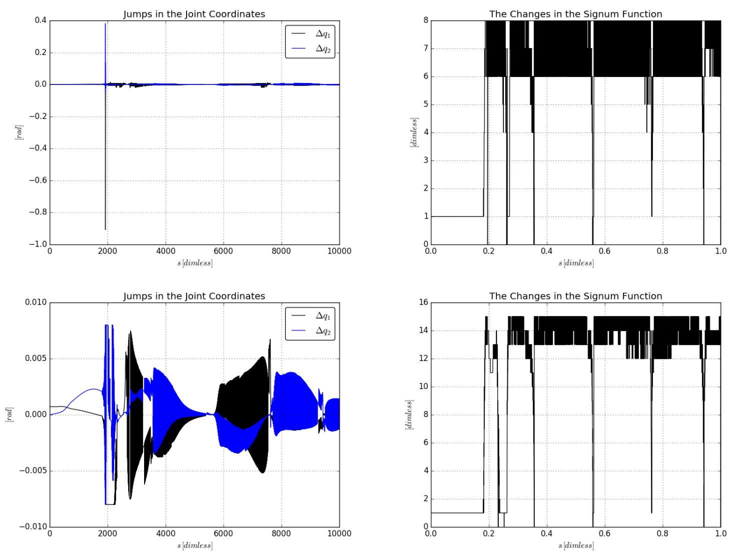

Figure 4.14: The jumps in the generalized coordinates and the number of changing the sign of σ within the internal iteration in the solution without "smoothing" (Top) and "smoothed" (Bottom)

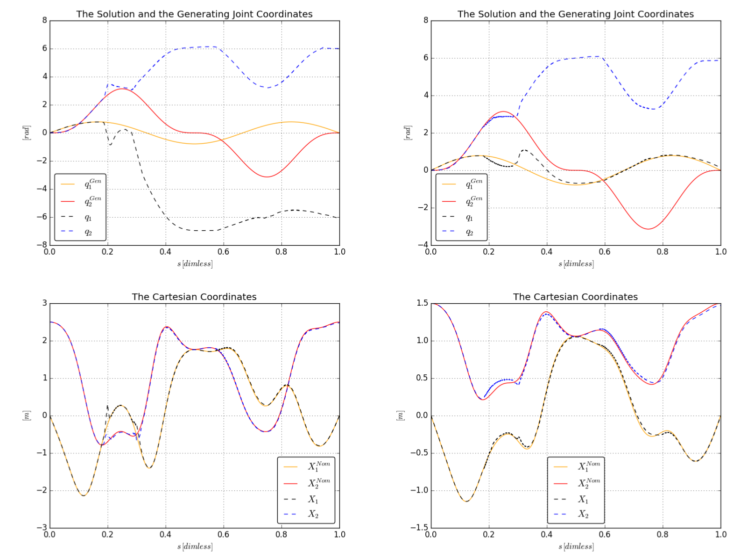

Figure 4.15: The joint coordinates of the trajectory generation (top LHS), the "smoothed" numerical solution (top RHS), the Cartesian coordinates of the trajectory tracking (bottom LHS) for $L_2 = 1.5\,m$ and the "smoothed" numerical solution (bottom RHS) for $L_2 = 0.5\,m$

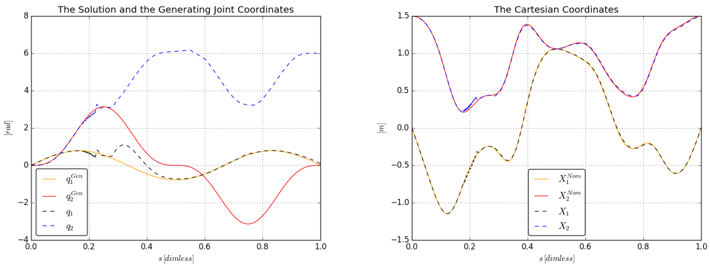

Figure 4.16: The joint coordinates of the trajectory generation and that of the "smoothed" numerical solution and trajectory tracking for $L_2 = 0.5\,m$ for 200 internal steps of iteration

4.6 Thesis Statement III.

I developed a novel method for adaptively solving the differential inverse kinematic task of redundant robot arms in the possession of an approximately known Jacobian. I is a generalization of the method by B. Csanádi that was based on the assumption of exactly known Jacobian. The novelty is the introduction of abstract rotations by the use of which the convergence can be guaranteed without knowing the exact Jacobian. The applicability of the method was investigated via simulatons for an 8 DoF robot arm.
Furthermore, I made investigations – at least for special cases – aiming at evading the computation of the Jacobian.

4.7 Thesis Substatement III.1

On the basis of the quasi–differential approach (in which only the computation of the Jacobian is needed without the calculation of its "generalized inverse") a possible alternative adaptive iterative approach was introduced as "Adaptive Inverse Kinematics" based on the application of the Fixed–Point Transformation (FPT). In this approach no complete information is needed on the Jacobian at a given point. The scientific novelty in this part consists of the fact that the here suggested procedure can be convergent and useful even if the Jacobian of the robot arm is only approximately known. The key factor is a rotational transformation, the application of which can improve the convergence properties of the iteration. It is content with the observable system behavior only along with the realized motion, so it seems to be easily implementable. Its operation is demonstrated for an irregularly extended 6 Degree of Freedom (DoF) PUMA–type robot arm, that has 8 rotary axles. From

the simulation–based results it can clearly be stated that the "tendency for divergence" is very small and only very tiny abstract rotations occurred in the simulations. Another outlining possibility seems to be the modification of the fixed–point transformation–based adaptive controllers in order to extend the set of physical systems for which this method can be convergent and practically useful.

4.8 Thesis Substatement III.2

In the inversion–free quasi–differential solution of the inverse kinematic of a redundant robot the calculation of the Jacobian is required. The computation of Jacobian generally was found to be very laborious, so conducting research to avoid this computational burden was important. I have tried to answer the question whether it is possible to avoid the calculation of the Jacobian in case of non–redundant robot arms of quadratic Jacobian. As a simple example, a 2 DoF arm was considered via simulations. It was shown that by the use of a simple complementary norm reduction built into the solution this approach was promising. However, for higher degree of freedom systems more investigations seem to be expedient.

Related own publication: [A. 7] [A. 8] [A. 9]

Chapter 5

Conclusions

The control of nonlinear systems with mathematical and simulation based techniques is an emerging area of studies in the field of engineering. Beside the inevitable presence of modeling imprecisions and errors, unknown external disturbances, measurement noises, observability problems as well as underactuation often arise in the engineering practice. These problems can be tackled by either robust or adaptive approaches that –depending on their internal mathematical construction– either need the development and use of complicated state observers or apply some simpler technique that is satisfied with the use of the directly observable signals without needing the estimation of the full internal system's state. The application of the technical realizations of the "universal approximators" elaborated for modeling continuous multiple variable functions under the name of "soft computing" is widely accepted, too. The subject area is kept developing with close connection with the development of the hardware and software applications. Theoretically well established "classical methods" that in the past were not applicable due to the general shortage of computing power nowadays become realizable options.

As an example, the realization of Lagrange's reduced gradient method in optimization problems nowadays is available if a discrete time–grid used as the approximation of a finite horizon is applied under the name of "Nonlinear Programming". One of my research area was the improvement of the nonlinear programing–based heuristic "Receding Horizon Controller" by releasing the historically prevailing restrictions, namely the application of quadratic terms in the cost function of the problem. I have shown via simulations that this approach is a promising possibility in treating type 1 diabetes mellitus. Further, with an additional RFPT framework the developed RHC controller can be extended in order to empower it with adaptive property. The solutions can be investigated from robustness, adaptivity, and other aspects' points of view.

In the 20$^{\text{th}}$ century the prevailing approach for designing adaptive controllers for strongly non–linear problems was based on Lypunov's "2$^{\text{nd}}$" or "Direct" method. Besides the mathematical difficulties that have to be coped with by the control designer, this method has certain practical shortcomings that it concentrates on the global (often asymptotic) stability of the controller while pays little attention to the "transient phase" of the controlled motion, normally it needs complete state estimation, and works on the basis of "satisfactory" instead of "necessary and satisfactory" conditions. These difficulties made the researchers elaborate mathematically simpler techniques that definitely concentrate on the transient part of the controlled motion, and do not require complete state observation or estimation. Instead of that it needs the observation of the "response" of the controlled system to the "actual control signal" applied. This approach mathematically was based on Banach's fixed point theorem proved in 1922.

In my research I have realized that while the Lyapunov function–based technique does not seem to be the one that easily can be combined with the idea of optimal controllers, the optimal control easily can be integrated with the adaptive control mathematically based on Banach's theorem. On this basis I invented the idea of the "Adaptive Receding Horizon Controller" and via simulation–based investigations I have shown that this idea deserves further attention.

Furthermore, I have observed that there is a formal possibility for the application of Banach's fixed point iteration–based method in the replacement of the computationally greedy Reduced Gradient method proposed by Lagrange in 1811, in the receding horizon controllers. I tried to develop techniques for further reduction of the computational needs of this approach and by the use of simulations highlighted the limitations of these seemingly plausible formal possibilities.

In the literature I have found a method that used Banach's fixed point iteration–based technique for the matrix inversion–free solution of the inverse kinematic task of redundant open kinematic chains. This method assumed that the designer has precise information on the Jacobian of the arm structure. Via investigating the convergence properties of this approach in the case in which the designer has only approximate information on this Jacobian, I elaborated an adaptive inverse kinematic approach that does not need the use of some generalized inverse of redundant robot arms. The idea is based on the application of abstract rotations by the use of which the method's convergence was made practically acceptable. The applicability of this method was illustrated by extensive simulation investigations. I also made attempt to evade the computation of the approximate Jacobian but it was found that in the case of a higher degree of freedom problems the expectations of the successful results using this approach seem not possible. However, further struggles to make it possible was taken into consideration.

Chapter 6

Possible Targets of Future Research

In my research the application of Banach's fixed point theorem in adaptive and adaptive optimal control played a role of key importance. The most attractive property of this approach is its mathematical simplicity and the fact that it does not need complete state estimation for the control. Since the practical lack of possibilities to realize complete state observation in the life sciences, communications sciences is a hard fact, this method may have widespread practical applications.

Though there are certain limitations regarding convergence properties, but its applicability was studied in the case of hard nonlinear control tasks as in anaesthesia control (e.g. [120, 121]), control of dynamically singular underactuated mechanical systems (e.g. [149]), treatment for "Type 1 Diabetes Mellitus" (e.g. [123]), control of nonlinear neuron models (e.g. [124, 125]), solution of the inverse kinematic task of robots [78], etc. On this reason it can be enough to say that the applicability of the method can be extended in many fields. The application of the approach can be fit for tackling adaptive RHC control realizations in biomedical applications.

However, the method has two practical limitations: its expected noise sensitivity if the relative order of the control task is high, and that during one digital control step only one step of Banach's iteration can be executed. Therefore the question generally arises: is the convergence of this method fast enough for keeping pace with the dynamics of the not precisely modeled phenomena taking part in the controlled system? Such a question cannot be generally answered, and abundant simulation studies have to be executed in different possible application areas to obtain satisfactory answer to it. Furthermore, the method offers formal possibilities to take into account time–delay effects that open an interesting research field to it, too.

Chapter 7

References

Own Publications Strictly Related to the Thesis

[A. 1] Hamza Khan, József K. Tar, Imre Rudas, Levente Kovács, and György Eigner: *"Receding Horizon Control of Type 1 Diabetes Mellitus by Using Non–linear Programming"*, Complexity, Article ID 4670159, `https://doi.org/10.1155/2018/4670159`, **2018**

[A. 2] Hamza Khan, Ágnes Szeghegyi, and József K. Tar: *"Fixed Point Transformation–based Adaptive Optimal Control Using NP"*, In Proc. of the 2017 IEEE 30th Jubilee Neumann Colloquium, November 24–25, 2017, Budapest, Hungary, pp. 35–40, **2017**

[A. 3] Hamza Khan, József K. Tar, Imre J. Rudas, and György Eigner: *"Iterative Solution in Adaptive Model Predictive Control by Using Fixed–Point Transformation Method"*, International Journal of Mathematical Models and Methods in Applied Sciences, Vol. 12, pp. 7–15, **2018**

[A. 4] Hamza Khan, József K. Tar, Imre J. Rudas, György Eigner: *"Adaptive Model Predictive Control Based on Fixed Point Iteration"*, WSEAS Transactions on Systems and Control, Vol. 12, pp. 347–354, **2017**

[A. 5] Hamza Khan, J.K. Tar,Károly Sźell: *"On Replacing Lagrange's "Reduced Gradient Algorithm" by Simplified Fixed Point Iteration in Adaptive Model Predictive Control"*, INES 2019 IEEE 23rd International Conference on Intelligent Engineering Systems April 25–27, 2019 Gödöllő, Hungary, **2019**

[A. 6] Hamza Khan, J.K. Tar *"On the Implementation of Fixed Point Iteration-based Adaptive Receding Horizon Control for Multiple Degree of Freedom, Higher Order Dynamical Systems"*, Acta Polytechnica Hungarica, Vol. 16, no. 9, pp. 135–154, DOI. 10.12700/APH.16.9.2019.9.8, **2019**

[A. 7] Hamza Khan, Aurél Galántai and József K. Tar: *"Adaptive Solution of the Inverse Kinematic Task by Fixed–Point Transformation"*, In Proc. of the SAMI 2017 IEEE 15th International Symposium on Applied Machine Intelligence and Informatics (SAMI 2017), January 26–28, 2017, Herl'any, Slovakia, pp. 247–252, **2017**

[A. 8] Hamza Khan and József K. Tar: *"Fixed Point Iteration–based Problem Solution without the Calculation of the Jacobian"*, In Proc. of the SAMI 2019 IEEE 17th International Symposium on Applied Machine Intelligence and Informatics (SAMI 2019), January 25–27, 2019, Herl'any, Slovakia, pp. 187–192, **2017**

[A. 9] Hamza Khan, J.K. Tar, *"Fine Tuning of the Fixed Point Iteration–Based Matrix Inversion–Free Adaptive Inverse Kinematics Using Abstract Rotations"*, Punjab University Journal of Mathematics, Vol. 52(3)(2020) pp. 111–134, **2020**

[A. 10] Hamza Khan, Tamás Faitli, Tamás Szili, and József K. Tar, *"Preliminary Investigation on the Possible Adaptive Control of an Inverted Pendulum–type Electric Cart"*, In Proc. of the IEEE 18th International Symposium on Computational Intelligence and Informatics (CINTI), 21–22 Nov. 2018, Budapest, Hungary, DOI: 10.1109/CINTI.2018.8928229, **2018**

[A. 11] Hamza Khan, József K. Tar, *"Novel Contradiction Resolution in Fixed Point Transformation-based Adaptive Control"*, In Proc. of the IEEE 18th International Symposium on Computational Intelligence and Informatics (CINTI), 21-22 Nov. 2018, Budapest, Hungary, DOI: 10.1109/CINTI.2018.8928235, **2018**

[A. 12] Hamza Khan, J.K. Tar, and Imre J. Rudas, *"On The Alternatives of Lyapunov's Direct Method in Adaptive Control Design"*, Robot Autom Eng J, vol. 3, no. 5, DOI:10.19080/RAEJ.2018.03.555623 **2020**

[A. 13] Hamza Khan, Hazem Issa, and J.K. Tar, *"Comparison of the Operation of Fixed Point Iteration–based Adaptive and Robust VS/SM–type Solutions for Controlling Two Coupled Fluid Tanks"*, Submitted for Publication in IEEE 20th

International Symposium on Computational Intelligence and Informatics (CINTI 2020), to be held in November 5–7, Budapest, Hungary, **2020**,

[A. 14] Hamza Khan, Hazem Issa, and J.K. Tar, *"Improved Simple Noise Filtering for Fixed Point Iteration–based Adaptive Controllers"*, Submitted for Publication in IEEE 20th International Symposium on Computational Intelligence and Informatics (CINTI 2020), to be held in November 5–7, Budapest, Hungary, **2020**,

Bibliography

[1] R.E. Kalman. Contribution to the theory of optimal control. *Boletin Sociedad Matematica Mexicana*, 5(1):102–119, 1960.

[2] V. Jurdjevic. *Geometric Control Theory*. Cambridge University Press, Cambrigde, New York, Melbourne, Madrid, Cape Town, Singapore, São Paulo, 1997.

[3] L. Grüne and J. Pannek. *Nonlinear Model Predictive Control*. Springer, 2011.

[4] A. Grancharova and T.A. Johansen. *Explicit Nonlinear Model Predictive Control*. Springer, 2012.

[5] R.E. Bellman. Dynamic programming and a new formalism in the calculus of variations. *Proc. Natl. Acad. Sci.*, 40(4):231–235, 1954.

[6] R.E. Bellman. *Dynamic Programming*. Princeton Univ. Press, Princeton, N. J., 1957.

[7] J. Richalet, A. Rault, J.L. Testud, and J. Papon. Model predictive heuristic control: Applications to industrial processes. *Automatica*, 14(5):413–428, 1978.

[8] J.L. Lagrange, J.P.M. Binet, and J.G. Garnier. *Mécanique analytique (Eds. J.P.M. Binet and J.G. Garnier)*. Ve Courcier, Paris, 1811.

[9] Zs. Horváth and A. Edelmayer. Robust model-based detection of faults in the air path of Diesel engines. *Acta Universitatis Sapientiae, Electrical and Mechanical Engineering*, 7:5–22, 2015.

[10] B. Davies. *Integral Transforms and Their Applications*. Springer Science & Business Media, 2002.

[11] P. Colaneri. *Analysis and control of linear switched systems (Lecture notes)*. Politecnico Di Milano, 2009.

[12] Qing-Kui Li, Jun Zhao, and Georgi M. Dimirovski. Robust tracking control for switched linear systems with time-varying delays. *IET Control Theory and Applications*, 2(6):449–457, 2008.

[13] Qing-Kui Li, Jun Zhao, and Georgi M. Dimirovski. Tracking control for switched time-varying delay systems with stabilizable and unstabilizable subsystems. *Nonlinear Analysis: Hybrid Systems*, 3(2):133–142, 2009.

[14] D. Tikk, P. Baranyi, R.J. Patton, I. Rudas, and J.K. Tar. Design methodology of tensor product based control models via HOSVD LMIs. *In Proc. of the IEEE Intl. Conf. on Industrial Technologies 2002*, pages 1290–1295, 2002.

[15] P. Baranyi, L. Szeidl, P. Várlaki, and Y. Yam. Definition of the HOSVD-based canonical form of polytopic dynamic models. *In Proc. of the 3^{rd} International Conference on Mechatronics (ICM 2006), Budapest, Hungary, July 3-5 2006*, pages 660–665, 2006.

[16] P. Baranyi, L. Szeidl, P. Várlaki, and Y. Yam. Numerical reconstruction of the HOSVD-based canonical form of polytopic dynamic models. *In Proc. of the 10^{th} International Conference on Intelligent Engineering Systems, London, UK, June 26-28 2006*, pages 196–201, 2006.

[17] S. Boyd, L.E. Ghaoui, E. Feron, and V. Balakrishnan. *Linear Matrix Inequalities in Systems and Control Theory*. SIAM books, Philadelphia, 1994.

[18] V.I. Arnold. *Mathematical Methods of Classical Mechanics*. Springer - Verlag, 1989.

[19] J. Riccati. Animadversiones in aequationes differentiales secundi gradus (observations regarding differential equations of the second order). *Actorum Eruditorum, quae Lipsiae publicantur, Supplementa,*, 8:66–73, 1724.

[20] W.M. Wonham. On a matrix Riccati equation of stochastic control. *SIAM Journal on Control and Optimization*, 6(1):681–697, 1968.

[21] A.J. Laub. *A Schur Method for Solving Algebraic Riccati Equations (LIDS-P 859 Research Report)*. MIT Libraries, Document Services, After 1979.

[22] E.V. Haynsworth. On the Schur complement. *Basel Mathematical Notes*, BMN 20:17, 1968.

[23] A.M. Lyapunov. *A General Task about the Stability of Motion. (in Russian)*. Ph.D. Thesis, University of Kazan, Tatarstan (Russia), 1892.

[24] A.M. Lyapunov. *Stability of Motion*. Academic Press, New-York and London, 1966.

[25] J.K. Tar, J.F. Bitó, L. Nádai, and J.A. Tenreiro Machado. Robust Fixed Point Transformations in adaptive control using local basin of attraction. *Acta Polytechnica Hungarica*, 6(1):21–37, 2009.

[26] S. Banach. Sur les opérations dans les ensembles abstraits et leur application aux équations intégrales (About the Operations in the Abstract Sets and Their Application to Integral Equations). *Fund. Math.*, 3:133–181, 1922.

[27] J.T. Sörensen. *A Physiologic Model of Glucose Metabolism in Man and Its use to Design and Assess Improved Insulin Therapies for Diabetes*. Massachusetts Institute of Technology, 1985.

[28] C. Cobelli and G. Pacini. Insulin secretion and hepatic extraction in humans by minimal modeling of C-peptide and insulin kinetics. *Diabetes*, 37:223–231, 1988.

[29] T.C. Ni, M. Ader, and E.N. Bergman. Reassessment of glucose effectiveness and insulin sensitivity from minimal model analysis: a theoretical evaluation of the single-compartment glucose distribution assumption. *Diabetes*, 46:1813–1821, 1997.

[30] R. Hovorka, F. Shojaee-Moradie, P.V. Carroll, L.J. Chassin, I.J. Gowrie, N.C. Jackson, R.S. Tudor, A.M. Umpleby, and R.H. Jones. Partitioning glucose distribution/transport, disposal, and endogenous production during ivgtt. *Am J Physiol Endocrinol Metab*, 282:E992–E1007, 2002.

[31] E. Friis-Jensen. *Modeling and Simulation of Glucose-Insulin Metabolism*. PhD Thesis, Technical University of Denmark, Kongens Lyngby, Denmark, 2007.

[32] L. Magni, D.M. Raimondo, L. Bossi, C. Dalla Man, G. De Nicolao, B. Kovatchev, and C. Cobelli. Model Predictive Control of Type 1 Diabetes: An in silico trial. *J Diab Sci Techn*, 1:804–812, 2007.

[33] I. Naşcu, R. Oberdieck, and E.N. Pistikopoulos. Offset-free explicit hybrid model predictive control of intravenous anaesthesia. *In: Proc. of the 2015 IEEE International Conference on Systems, Man, and Cybernetics, October 9-13, 2015, Hong Kong*, pages 2475–2480, 2015.

[34] M.M.R.F. Struys, H. Vereecke, A. Moerman, E.W. Jensen, D. Verhaegen, N. De Neve, F.J.E. Dumortier, and E.P. Mortier. Ability of the Bispectral Index, autoregressive modelling with exogenous input-derived auditory evoked potentials, and predicted Propofol concentrations to measure patient responsiveness during anesthesia with Propofol and Remifentanil. *Anesthesiology*, 99(4):802–812, 2003.

[35] T.W. Schnider, C.F. Minto, P.L. Gambus, C. Andersen, D.B. Goodale, S.L. Shafer, and E. Youngs. The influence of method of administration and covariates on the pharmacokinetics of propofol in adult volunteers [clinical investigations]. *Anesthesiology*, 88(5):1170–1182, 1998.

[36] T.W. Schnider, C.F. Minto, S.L. Shafer, P.L. Gambus, C. Andresen, D.B. Goodale, and E.J. Youngs. The influence of age on propofol pharmacodynamics. *Anesthesiology*, 90(6):1502–1516, 1999.

[37] L. Lapicque. Recherches quantitatives sur l'excitation électrique des nerfs traitée comme une polarisation. *J. Physiol. Pathol.*, 9:620–635, 1907.

[38] P. Dayan and L. F. Abbott. *Theoretical Neuroscience - Computational and Mathematical Modeling of Neural Systems*. MIT Press, 2001.

[39] A.L. Hodgkin and A.F. Huxley. A quantitative description of membrane current and its application to conduction and excitation in nerve. *The Journal of Physiology*, 117(4):500–544, 1952.

[40] T. Matsumoto. A chaotic attractor from Chua's circuit. *IEEE Transactions on Circuits and Systems*, CAS-31(12):1055–1058, 1984.

[41] L. Glass. *Chaos in neural systems. In: Arbib, M. (ed.) The Handbook of Brain Theory and Neural Networks*. MIT, Cambridge, 1995.

[42] B. Armstrong, O. Khatib, and J. Burdick. The explicit dynamic model and internal parameters of the PUMA 560 arm. *Proc. IEEE Conf. On Robotics and Automation 1986*, pages 510–518, 1986.

[43] P.I. Corke and B. Armstrong-Helouvry. A search for consensus among model parameters reported for the PUMA 560 robot. *Proc. IEEE Conf. Robotics and Automation, 1994*, pages 1608–1613, 1994.

[44] N. Moldoványi. *Model Predictive Control of Crystallisers*. PhD Thesis, Department of Process Engineering, University of Pannonia, Veszprém, Hungary, 2012.

[45] T. Bellemans, B. De Schutter, and B. De Moor. Anticipative model predictive control for ramp metering in freeway networks. *Proceedings of the 2003 American Control Conference, Denver, Colorado*, pages 4007–4082, 2003.

[46] T. Luspay, I. Varga, B. Kulcsár, and J. Bokor. *Modeling freeway traffic flow: an LPV approach*. EUROSIM, 2007.

[47] T. Luspay. *Advanced Freeway Traffic Modeling and Control - Linear Parameter Varying Concepts*. PhD Thesis, Budapest University of Technology and Economics, Dept. of Control and Traffic Automation, 2011.

[48] A. Csikós. *Modeling and control methods for the reduction of traffic pollution and traffic stabilization*. PhD Thesis, Budapest University of Technology and Economics, 2015.

[49] A. Csikós, I. Varga, and K.M. Hangos. Modeling of the dispersion of motorway traffic emission for control purposes. *Transportation Research Part C: Emerging Technologies*, 58:598–616, 2015.

[50] D.W. Clarke, C. Mohtadi, and P.S. Tuffs. Generalized predictive control – I. The basic algorithm. *Automatica*, 23:137–148, 1987.

[51] D.W. Clarke, C. Mohtadi, and P.S. Tuffs. Generalized predictive control – II. Extensions and interpretations. *Automatica*, 23:149–160, 1987.

[52] B.W. Bequette. Non-linear control of chemical processes: A review. *Ind. Engng. Chem. Res.*, 30:1391–1413, 1991.

[53] S. Rohani, M. Haeri, and H.C. Wood. Modeling and control of a continuous crystallization process Part 1. Linear and non-linear modeling. *Computers & Chem. Eng.*, 23:263, 1999.

[54] S. Rohani, M. Haeri, and H.C. Wood. Modeling and control of a continuous crystallization process Part 2. Model predictive control. *Computers & Chem. Eng.*, 23:279, 1999.

[55] J.W. Eaton and J.B. Rawlings. Feedback control of chemical processes using on-line optimization techniques. *Computers & Chem. Eng.*, 14:469–479, 1990.

[56] R. Hovorka, V. Canonico, L.J. Chassin, U. Haueter, M. Massi-Benedetti, M. Orsini-Federici, T.R. Pieber, H.C. Schaller, L. Schaupp, T. Vering, and M.E. Wilinska. Nonlinear model predictive control of glucose concentration in subjects with type 1 diabetes. *Physiol Meas*, 25(4):905–920, 2004.

[57] N. Muthukumar, Seshadhri Srinivasan, K. Ramkumar, K. Kannan, and V.E. Balas. Adaptive model predictive controller for web transport systems. *Acta Polytechnica Hungarica*, 13(3):181–194, 2016.

[58] Joseph Bronzino and Donald Peterson. *The Biomedical Engineering Handbook*. CRC Press, Boca Raton, Florida, USA, 4 edition, 2015.

[59] F. Padula, C. Ionescu, N. Latronico, M. Paltenghi, A. Visioli, and G. Vivacqua. A gain-scheduled PID controller for Propofol dosing in anesthesia. *In: Preprints of the 9th IFAC Symposium on Biological and Medical Systems, The International Federation of Automatic Control Berlin, Germany, Aug. 31 - Sept. 2, 2015*, pages 545–550, 2015.

[60] I. Nascu, A. Krieger, C.M. Ionescu, and E.N. Pistikopoulos. Advanced model-based control studies for the induction and maintenance of intravenous anaesthesia. *IEEE Tran Biomed Eng*, 62(3):832–841, 2015.

[61] D. Drexler, J. Sápi, and L. Kovács. Potential Benefits of Discrete-Time Controller-based Treatments over Protocol-based Cancer Therapies. *ACTA Pol Hung*, 14(1):11–23, 2017.

[62] J. Sápi, L. Kovács, D.A. Drexler, P. Kocsis, D. Gaári, and Z. Sápi. Tumor volume estimation and quasi-continuous administration for most effective bevacizumab therapy. *PLoS ONE*, 10(11), 2015.

[63] Ryan Zurakowski and Andrew R Teel. A model predictive control based scheduling method for hiv therapy. *Journal of Theoretical Biology*, 238(2):368–382, 2006.

[64] P Colmegna, RS Sánchez-Peña, and R Gondhalekar. Linear parameter-varying model to design control laws for an artificial pancreas. *Biomedical Signal Processing and Control*, 40:204–213, 2018.

[65] Gy. Eigner, J.K. Tar, I.J. Rudas, and L. Kovács. LPV-based quality interpretations on modeling and control of diabetes. *Acta Polytechnica Hungarica*, 13(1):171–190, 2016.

[66] Levente Kovács. Linear parameter varying (LPV) based robust control of type-I diabetes driven for real patient data. *Knowledge-Based Systems*, 122:199–213, 2017.

[67] Ahmad Haidar. The Artificial Pancreas: How Closed-Loop Control Is Revolutionizing Diabetes. *IEEE Contr Syst Mag*, 36:28–47, 2016.

[68] V.B. Kolmanovskii, S.-I. Niculescu, and D. Richard. On the Lyapunov-Krasovskii functionals for stability analysis of linear delay systems. *International Journal of Control*, 72(4):374–384.

[69] J.K. Tar, I.J. Rudas, J.F.Bitó, and M.O. Kaynak. Adaptive robot control gained by partial identification using the advantages of sympletic geometry. *Proc. of the 1995 IEEE 21st International Conference on Industrial Electronics, Control, and Instrumentation. November, 1995, Orlando, USA*, pages 75–80, 1995.

[70] J.K. Tar, A. Szakál, I.J. Rudas, and J.F. Bitó. Selection of different abstract groups for developing uniform structures to be used in adaptive control of robots. *Proc. of the 2000 IEEE International Symposium on Industrial Electronics, UDLA, December 4-8, 2000, Universidad de Las Américas-Puebla, Cholula, Puebla, México*, pages 559–564, 2000.

[71] J.K. Tar, I.J. Rudas, J.F. Bitó, and K. Jezernik. A generalized Lorentz group-based adaptive control for DC drives driving mechanical components. *In Proc. of the* 10^{th} *Intl. Conf. on Advanced Robotics (ICAR 2001) Budapest*, pages 299–305, 2001.

[72] J.K. Tar, J.A. Tenreiro Machado, I.J. Rudas, and J.F. Bitó. An adaptive robot control for technological operations based on uniform structures and reduced number of free parameters. *In Proc. of the 8th International Workshop on Robotics in Alpe-Adria-Danube Region (RAAD'99), 17-19 June, 1999, Munich, Germany*, pages 106–1011, 1999.

[73] J.K. Tar, J.F. Bitó, and I.J. Rudas. Replacement of Lyapunov's Direct Method in Model Reference Adaptive Control with Robust Fixed Point Transformations. *In Proc. of the* 14^{th} *IEEE Intl. Conf. on Intelligent Engineering Systems, Las Palmas of Gran Canaria, Spain*, pages 231–235, 2010.

[74] I. Sekaj and V. Veselý. Robust output feedback controller design: Genetic Algorithm approach. *IMA J Math Control Info*, 22(3):257–265, 2005.

[75] A. Jadbabaie, J. Yu, and J. Hauser. Receding horizon control of the Caltech ducted fan: A control Lyapunov function approach. *In the Proc. of the IEEE Conference on Control Applications, 1999*, pages 51–56, 1999.

[76] A. Jadbabaie. *Receding Horizon Control of Nonlinear Systems: A Control Lyapunov Function Approach (PhD Thesis, California Institute of Technology, Pasadena, California, USA)*. "http://www.seas.

upenn.edu/jadbabai/papers/Phdthesis.pdf", 2000, Visited: 2014. Jan. 15.

[77] H. Khan, A. Galántai, and J.K. Tar. Adaptive solution of the inverse kinematic task by fixed point transformation. *In Proc. of the 15th IEEE International Symposium on Applied Machine Intelligence and Informatics, 26-28 January, 2017, Herl'any, Slovakia*, pages 247–252, 2017.

[78] B. Csanádi, J.K. Tar, and J.F. Bitó. Matrix inversion-free quasi-differential approach in solving the inverse kinematic task. *In Proc. of the* 17^{th} *IEEE International Symposium on Computational Intelligence and Informatics (CINTI 2016), 17-19 November 2016, Budapest, Hungary*, pages 61–66, 2016.

[79] B. Csanádi, J.F. Bitó, I.J. Rudas, and J.K. Tar. Comparative analysis of quasi-differential approaches in inverse kinematics. *In Proc. of the 26th International Conference on Robotics in Alpe-Adria-Danube Region (RAAD 2017), Turin, Italy, 21-23 June 2017, Eds. Carlo Ferraresi & Giuseppe Quaglia*, pages 3–10, 2017.

[80] H. Khan, Á. Szeghegyi, and J.K. Tar. Fixed point transformation-based adaptive optimal control using NP. *In Proc. of the 2017 IEEE 30th Jubilee Neumann Colloquium, November 24-25, 2017, Budapest, Hungary*, pages 35–40, 2017.

[81] H. Khan, J.K. Tar, I.J. Rudas, and Gy. Eigner. Adaptive model predictive control based on fixed point iteration. *WSEAS Transactions on Systems and Control*, 12:347–354, 2017.

[82] H. Khan, J.K. Tar, I.J. Rudas, and Gy. Eigner. Iterative solution in Adaptive Model Predictive Control by using Fixed-Point Transformation method. *International Journal of Mathematical Models and Methods in Applied Sciences*, 12:7–15, 2018.

[83] H. Khan, J.K. Tar, I. Rudas, L. Kovács, and Gy. Eigner. Receding Horizon Control of Type 1 Diabetes Mellitus by using Nonlinear Programming. *Complexity*, https://doi.org/10.1155/2018/4670159, 2018.

[84] H. Khan and J.K. Tar. Fixed point iteration-based problem solution without the calculation of the jacobian. *In Proc. of the IEEE 17th International Symposium on Applied Machine Intelligence and Informatics (SAMI 2019), January 25-27, 2019, Herl'any, Slovakia,*, pages 187–192, 2019.

[85] H. Khan and J.K. Tar. On the implementation of fixed point iteration-based adaptive receding horizon control for multiple degree of freedom, higher order dynamical systems. *Acta Polytechnica Hungarica*, 16(9):135–154, 2019.

[86] Hamza Khan and József K. Tar. On replacing Lagrange's "Reduced Gradient Algorithm" by simplified fixed point iteration in adaptive model predictive control. *In Proc. of the 23^{rd} IEEE International Symposium on Intelligent Engineering Systems (INES 2019), Gödöllő, Hungary, April 25-27, 2019*, pages 201–206, 2019.

[87] H. Khan and J.K. Tar. Fine tuning of the fixed point iteration-based matrix inversion-free adaptive inverse kinematics using abstract rotations. *Punjab University Journal of Mathematics*, 52(3):1–15, 2020.

[88] A. Dineva, J.K. Tar, and A.R. Várkonyi-Kóczy. Novel generation of Fixed Point Transformation for the adaptive control of a nonlinear neuron model. *In proc. of the IEEE International Conference on Systems, Man, and Cybernetics, October 10-13, 2015, Hong Kong (SMC 2015)*, pages 987–992, 2015.

[89] A. Dineva, J.K. Tar, A.R. Várkonyi-Kóczy, and V. Piuri. Generalization of a Sigmoid Generated Fixed Point Transformation from SISO to MIMO systems. *In Proc. of the IEEE 19^{th} International Conference on Intelligent Engineering Systems, September 3-5, 2015, Bratislava, Slovakia (INES 2015)*, pages 135–140, 2015.

[90] J. Bezanson, A. Edelman, S. Karpinski, and V.B. Shah. Julia. `https://julialang.org`, Last time visited: 5 May 2019.

[91] Leon S Lasdon, Allan D Waren, Arvind Jain, and Margery Ratner. Design and testing of a generalized reduced gradient code for nonlinear programming. *ACM Transactions on Mathematical Software (TOMS)*, 4(1):34–50, 1978.

[92] Daniel Fylstra, Leon Lasdon, John Watson, and Allan Waren. Design and use of the microsoft excel solver. *Interfaces*, 28(5):29–55, 1998.

[93] World Health Organization. *Global Report on Diabetes*. WHO Press, Geneva, Switzerland, 2016.

[94] American Diabetes Association et al. Diagnosis and classification of diabetes mellitus. *Diabetes care*, 37(Supplement 1):S81–S90, 2014.

[95] World Health Organization et al. Definition and diagnosis of diabetes mellitus and intermediate hyperglycemia: report of a WHO/IDF consultation. *World Hearth Org*, 2006.

[96] F. Chee and T. Fernando. *Closed-Loop Control of Blood Glucose*. Springer, 2007.

[97] R.N. Bergman, Y.Z. Ider, C.R. Bowden, and C. Cobelli. Quantitative estimation of insulin sensitivity. *Am. J. Physiol. Endocrinol. Metab.*, 236:E667–E677, 1979.

[98] T. Ferenci, A. Körner, and L. Kovács. The interrelationship of HbA1c and real-time continuous glucose monitoring in children with type 1 diabetes. *Diabetes Research and Clinical Practice*, 108(1):38–44, 2015.

[99] Tamás Ferenci, Johanna Sápi, and Levente Kovács. Modelling Tumor Growth Under Angiogenesis In-hibition with Mixed-effects Models. *Acta Polytechnica Hungarica*, 14(1), 2017.

[100] C. Cobelli, E. Renard, and B. Kovatchev. Artificial pancreas: Past, present and future. *Diabetes*, 60(11):2672 – 2682, 2011.

[101] R. Nimri, A. Murray, N. abd Ochs, J.E. Pinsker, and E. Dassau. Closing the Loop. *Diabetes Technol The*, 19(1):s27 – s41, 2017.

[102] Joon Bok Lee, Eyal Dassau, Dale E Seborg, and Francis J Doyle. Model-based personalization scheme of an artificial pancreas for type 1 diabetes applications. In *American Control Conference (ACC), 2013*, pages 2911–2916. IEEE, 2013.

[103] F.J. Doyle, E. Dassau, D.E. Seborg, and J.B. Lee. Model-Based Personalization Scheme of an Artificial Pancreas for Type I Diabetes Applications, October 29 2015. US Patent App. 14/792,524.

[104] Jordan E Pinsker, Joon Bok Lee, Eyal Dassau, Dale E Seborg, Paige K Bradley, Ravi Gondhalekar, Wendy C Bevier, Lauren Huyett, Howard C Zisser, and Francis J Doyle. Randomized Crossover Comparison of Personalized MPC and PID Control Algorithms for the Artificial Pancreas. *Diabetes care*, 39:1135 – 1142, 2016.

[105] Marwa Jouini, Slim Dhahri, and Anis Sellami. Design of robust super-twisting algorithm based second-order sliding mode controller for nonlinear systems with both matched and unmatched uncertainty. *Complexity*, 2017, 2017.

[106] J. Kuti, P. Galambos, and P. Baranyi. Minimal volume simplex (mvs) convex hull generation and manipulation methodology for tp model transformation. *Asian J Control*, 19(1):289 – 301, 2017.

[107] L. Kovács and Gy. Eigner. Convex Polytopic Modeling of Diabetes Mellitus: A Tensor Product based approach. In *IEEE International Conference on Systems, Man, and Cybernetics 2016: IEEE SMC 2016*, pages 3393–3398. IEEE SMC.

[108] Levente Kovács, Csaba Maszlag, Miklós Mezei, and György Eigner. Robust nonlinear model predictive control of diabetes mellitus. In *Applied Machine Intelligence and Informatics (SAMI), 2017 IEEE 15th International Symposium on*, pages 000055–000060. IEEE, 2017.

[109] V.N. Shah, A. Shoskes, B. Tawfik, and S.K. Garg. Closed-loop system in the management of diabetes: Past, present, and future. *Diabetes Technol The*, 16(8):477 – 490, 2014.

[110] David Q Mayne, James B Rawlings, Christopher V Rao, and Pierre OM Scokaert. Constrained model predictive control: Stability and optimality. *Automatica*, 36(6):789–814, 2000.

[111] L. Grüne and J. Pannek. *Nonlinear Model Predictive Control*. Springer, London, 2011.

[112] A. Jadbabaie, J. Yu, and J. Hauser. Stabilizing Receding Horizon Control of Nonlinear Systems: A Control Lyapunov Function Approach. *In the Proc. of the American Control Conference, 1999*, pages 1535–1539, 1999.

[113] J.K. Tar, L. Nádai, and I.J. Rudas. *System and Control Theory with Especial Emphasis on Nonlinear Systems*. Typotex, Budapest, Hungary, 1st edition, 2012.

[114] R. Hovorka, V. Canonico, L.J. Chassin, U. Haueter, M. Massi-Benedetti, M. Orsini-Federici, T.R. Pieber, H.C. Schaller, L. Schaupp, T. Vering, and Wilinska M.E. Nonlinear model predictive control of glucose concentration in subjects with type 1 diabetes. *Physiol Meas*, 25(4):905 – 920, 2004.

[115] R. E. Mickens. Model predictive control for insulin administration in people with type 1 diabetes. Technical report, Technical University of Denmark, 2010.

[116] Tayfun Çimen. State-dependent Riccati equation in nonlinear optimal control synthesis. *In the Proc. of the Special International Conference on Complex Systems: Synergy of Control, Communications and Computing - COSY 2011, Hotel Metropol Resort, Ohrid, Republic of Macedonia, September, 16 – 20, 2011*, pages 321–332, 2011.

[117] C.T. Kelley. *Solving Nonlinear Equations with Newton's Method, no 1 in Fundamentals of Algorithms*. SIAM, 2003.

[118] A. Dineva, J.K. Tar, A. Várkonyi-Kóczy, and V. Piuri. Adaptive control of underactuated mechanical systems using improved "Sigmoid Generated Fixed Point Transformation" and scheduling strategy. *In Proc. of the* 14^{th} *IEEE International Symposium on Applied Machine Intelligence and Informatics, January 21-23, 2016, Herl'any, Slovakia*, pages 193–197, 2016.

[119] A. Dineva. *Non-conventional Data Representation and Control (PhD Thesis, Supervisors: A.R. Várkonyi-Kóczy & J.K. Tar)*. Óbuda University, Budapest, Hungary, 2016.

[120] B. Csanádi and J.K. Tar. Selection of kinematic requirements for RFPT-based adaptive anaesthesia control. *In Proc. of the* 11^{th} *IEEE International Symposium on Applied Computational Intelligence and Informatics (SACI 2016), 12-14 May 2016, Timisoara, Romania*, pages 181–186, 2016.

[121] A. Dineva, J.K. Tar, A.R. Vákonyi-Kóczi, and V. Piuri. Adaptive controller using Fixed Point Transformation for regulating Propofol administration through wavelet-based anesthetic value. *In Proc. of the IEEE International Symposium on Medical Measurements and Applications (MeMeA 2016), 05.15-05.18, Benevento, Italy*, pages 650–655, 2016.

[122] J.K. Tar, J.F. Bitó, and I. Rudas. Contradiction resolution in the adaptive control of underactuated mechanical systems evading the framework of optimal controllers. *Acta Polytechnica Hungarica*, 13(1):97–121, 2016.

[123] Gy. Eigner, P. Horváth, J. Tar, I. Rudas, and L. Kovács. Application of Robust Fixed Point control in case of T1DM. *In Proc. of the IEEE International Conference on Systems, Man, and Cybernetics 2015 (IEEE SMC 2015), Hong Kong, China*, pages 2459–2464, 2015.

[124] I.J. Rudas, J.K. Tar, and T.A. Várkonyi. Novel adaptive synchronization of different chaotic Chua circuits. *Proc. of the Special International Conference on Complex Systems: Synergy, of Control, Communications and Computing (ETAI COSY 2011), Ohrid, Macedonia, Sept. 16 – 20 2011*, pages 109–114, 2011.

[125] J.F. Bitó and J.K. Tar. Fixed point transformation-based adaptive control of the Hodgkin-Huxley neuron. *Proc. of the Workshop on Information Technology and Bionics – Symposium in Memory of Tamás Roska, 23-24 June, 2015, Pázmány University* e*Press, Budapest, Hungary*, pages 69–72, 2015.

[126] B. van der Pol. Forced oscillations in a circuit with non-linear resistance (reception with reactive triode). *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 7(3):65–80, 1927.

[127] I. Kovacic and R.E. Mickens. A generalized van der Pol type oscillator: Investigation of its limit cycle. *Mathematical and Computer Modelling*, 55(3-4):645–653, 2012.

[128] Tjalling J. Ypma. Historical development of the Newton-Raphson method. *SIAM Review*, 37(4):531–551, 1995.

[129] C.S.G. Lee and M. Ziegler. *RSD-TR-1-83 A geometric approach is solving the inverse kinematics of PUMA robots*. The University of Michigan, Ann Arbor, Michigan 48109-1109, 1983.

[130] E.H. Moore. On the reciprocal of the general algebraic matrix. *Bulletin of the American Mathematical Society*, 26(9):394–395, 1920.

[131] R. Penrose. A generalized inverse for matrices. *Proceedings of the Cambridge Philosophical Society*, 51:406–413, 1955.

[132] S. Chiaverini, O. Egeland, and R.K. Kanestrom. Achieving user-defined accuracy with damped least squares inverse kinematics. *1991 IEEE International Conference on Robotics and Automation, June 19-22, 1991, Pisa, Italy*, page 672, 1991.

[133] R.T. Rockafellar. *Convex Analysis*. Princeton University Press, 1970.

[134] H. Khan and J.K. Tar. Fixed point iteration-based problem solution without the calculation of the Jacobian. *In Proc. of the IEEE 17th World Symposium on Applied Machine Intelligence and Informatics (SAMI 2019), January 24-26, 2019, Herl'any, Slovakia*, pages 187–192, 2019.

[135] G.E. Shilov. *Linear Algebra (Dover Books on Mathematics)*. Dover Publications, 1977.

[136] A. Ben-Israel and T.N.E. Greville. *Generalized Inverses*. Springer-Verlag, 2003.

[137] G.H. Golub and W. Kahan. Calculating the singular values and pseudoinverse of a matrix. *SIAM Journal on Numerical Analysis*, 2:205–224, 1965.

[138] J.P. Gram. Über die Entwickelung reeler Funktionen in Reihen mittelst der Methode der kleinsten Quadrate. *Journal für die reine und angewandte Mathematik*, 94:71–73, 1883.

[139] E. Schmidt. Zur Theorie der linearen und nichtlinearen Integralgleichungen. I. Teil: Entwicklung willkürlicher Funktionen nach Systemen vorgeschriebener. *Matematische Annalen*, 63:442, 1907.

[140] P.-S. Laplace. *Théorie Analytique des Probabilités, in: Oeuvres completes de Laplace / publiés sous les auspices de l'Académie des sciences, par MM. les secrétaires perpétuel*. Gauthier-Villars, Paris, 1886.

[141] J.K. Tar, L. Nádai, I. Felde, and I.J. Rudas. *Cost Function-free Optimization in Inverse Kinematics of Open Kinematic Chains, In Proc. of the 24^{th} International Workshop on Robotics in Alpe Adria Danube Region (RAAD 2015), May 27-29 2015, Bucharest, Romania, In T. Borangiu (ed.) Advances in Robot Design and Intelligent Control (within the series "Advances in Systems and Computing" 371) pp. 137-145*. Springer International Publishing AG, Switzerland, 2015.

[142] K. Levenberg. A method for the solution of certain non-linear problems in least squares. *Quarterly of Applied Mathematics*, 2:164–168, 1944.

[143] E.D. Pohl. A new method of robotic rate control near singularities. *1991 IEEE International Conference on Robotics and Automation, June 19-22, 1991, Pisa, Italy*, page 405, 1991.

[144] E.D. Pohl and H. Lipkin. Complex robotic inverse kinematic solutions. *J. Mech. Des.*, 115(3):509–514, 1993.

[145] H. Khan, J.K. Tar, I.J. Rudas, and Gy. Eigner. Adaptive model predictive control based on fixed point iteration. *WSEAS TRANSACTIONS on SYSTEMS and CONTROL*, 12:347–354, 2017.

[146] O. Rodrigues. Des lois géometriques qui regissent les déplacements d' un systéme solide dans l' espace, et de la variation des coordonnées provenant de ces déplacement considérées indépendent des causes qui peuvent les produire. *J. Math. Pures Appl.*, 5:380–440, 1840.

[147] B. Csanádi, P. Galambos, J.K. Tar, Gy. Györök, and A. Serester. A novel, abstract rotation-based fixed point transformation in adaptive control. *In*

the Proc. of the 2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC2018), October 7-10, 2018, Miyazaki, Japan, pages 2577–2582, 2018.

[148] Brian D.O. Anderson and John B. Moore. *Optimal Control: Linear Quadratic Methods*. Prentice – Hall International, Inc., A Division of Simon & Schuster, Englewood Cliffs, NJ 07632, 1989.

[149] J.K. Tar, J.F. Bitó, and I.J. Rudas. Contradiction resolution in the adaptive control of underactuated mechanical systems evading the framework of optimal controllers. *Acta Polytechnica Hungarica*, 13(1):97–121, 2016.