UNIVERSIDAD POLITÉCNICA DE CARTAGENA

DEPARTAMENTO DE MATEMÁTICA APLICADA Y ESTADÍSTICA

APROXIMACIÓN DE ECUACIONES DIFERENCIALES MEDIANTE UNA NUEVA TÉCNICA VARIACIONAL Y APLICACIONES.

María José Legaz Almansa

2012

UNIVERSIDAD POLITÉCNICA DE CARTAGENA

DEPARTAMENTO DE MATEMÁTICA APLICADA Y ESTADÍSTICA

APROXIMACIÓN DE ECUACIONES DIFERENCIALES MEDIANTE UNA NUEVA TÉCNICA VARIACIONAL Y APLICACIONES.

Directores
Sergio Amat Plata
Pablo Pedregal Tercero

Memoria presentada por
María José Legaz Almansa
para optar al grado de
Doctor

Cartagena, 2012

Universidad Politécnica de Cartagena

DT-16

CONFORMIDAD DE SOLICITUD DEAUTORIZACIÓN DE DEPÓSITO DE TESIS DOCTORAL POR EL/LA DIRECTOR/A DE LA TESIS

D. Sergio Amat Plata y Pablo Pedregal Tercero Directores de la Tesis doctoral Aproximación de ecuaciones diferenciales mediante una nueva técnica variacional y aplicaciones.

INFORMA:

Que la referida Tesis Doctoral, ha sido realizada por Dª. Mª José Legaz Almansa, dando mi conformidad para que sea presentada ante la Comisión de Doctorado, para ser autorizado su depósito.

La rama de conocimiento por la que esta tesis ha sido desarrollada es: **Ciencias**

En Cartagena, a 3 de diciembre de 2012

DIRECTORES DE LA TESIS

Fdo.: Sergio Amat Plata Fdo.:Pablo Pedregal Tercero

COMISIÓN DE DOCTORADO

Universidad Politécnica de Cartagena

DT-17

CONFORMIDAD DE DEPÓSITO DE TESIS DOCTORAL POR LA COMISIÓN ACADÉMICA DEL PROGRAMA

D. Francisco Alhama López, Presidente/a de la Comisión Académica del Programa Tecnologías Industriales.

INFORMA:

Que la Tesis Doctoral titulada, "Aproximación de ecuaciones diferenciales mediante una nueva técnica variacional y aplicaciones.", ha sido realizada por Dª. Mª José Legaz Almansa, bajo la dirección y supervisión de los doctores Sergio Amat Plata y Pablo Pedregal Tercero.

En reunión de la Comisión Académica de fecha 4 de diciembre de 2012, visto que la mencionada tesis doctoral tiene acreditados los indicios de calidad, requeridos para el depósito de tesis doctorales, regulados en el artículo 32 del Reglamento de Estudios Oficiales de Máster y Doctorado de la UPCT, y la autorización del Director de la misma, se acordó dar la conformidad para que a dicha tesis le sea autorizado, por la Comisión de Doctorado, su depósito.

La Rama de conocimiento por la que esta tesis ha sido desarrollada es: **Ciencias**

En Cartagena, a 4 de diciembre de 2012

EL PRESIDENTE DE LA COMISIÓN ACADÉMICA DEL PROGRAMA

Fdo: Francisco Alhama López

COMISIÓN DE DOCTORADO

En primer lugar quiero dedicar esta tesis doctoral a toda mi familia y muy especialmente a mi madre Pepita, por su apoyo incondicional en mi decisión de prolongar mis estudios.

Extiendo mi agradecimiento a mi novio, Celedonio, quien ha sido el sufridor silencioso de todos mis desvelos y mi asesor informático más preciado.

Dar las gracias a la universidad politécnica de Cartagena por concederme la beca de iniciación a la investigación.

Y dejo para el final pero no por ser menos importantes a mis directores de tesis, el doctor Pablo Pedregal Tercero de la universidad de Castilla la Mancha, gran intelectual a quien agradezco su colaboración e ideas. Y el doctor Sergio Amat Plata de la universidad politécnica de Cartagena, para quien no es fácil expresar con palabras mi más sincera y profunda gratitud, tanto por su calidad humana como por haber conseguido trasmitirme su entusiasmo por la investigación y el trabajo bien hecho. Una mente brillante con la que he tenido la suerte de poder contar.

No puedo concluir mis agradecimientos sin dar las gracias a la doctora Sonia Busquier Sáez con quien he tenido la fortuna de poder contar para cualquier cosa que necesitara.

GRACIAS A TODOS VOSOTROS POR HABER FORMADO PARTE DE MI VIDA Y HABERME AYUDADO A MEJORARLA.

In the first place, I would like to dedicate this thesis to my whole family and especially to my mother Pepita, for their unconditional support in my decision to continue my studies.

I extend my thanks to my boyfriend, Celedonio, who has been the silent sufferer of all my concern and my most precious computer consultant.

Thanks to the Polytechnic University of Cartagena for giving me the grant for initiation to research.

Last but not least, to my thesis directors, Dr. Pablo Pedregal Third from University of Castilla La Mancha, great intellectual to whom appreciate his collaboration and many ideas. And, to Dr. Sergio Amat Plata of the Polytechnic University of Cartagena, for whom it is not easy to put into words my sincere and deep gratitude, both for his human quality as for having gotten to transmit his enthusiasm for research and well done job. A brilliant mind with who I have been fortunate to can count.

I can not conclude my acknowledgments without giving thanks to Dr. Sonia Busquier Sáez with whom I have had lucky to count for anything I needed it.

THANK EVERY ONE FOR HAVING BEEN PART OF MY LIFE, HELPING ME TO IMPROVE IT.

Resumen

En esta Tesis presentamos el estudio teórico y numérico de sistemas de ecuaciones diferenciales basado en el análisis de un funcional asociado de forma natural al problema original. Probamos que cuando se utiliza métodos del descenso para minimizar dicho funcional, el algoritmo decrece el error hasta obtener la convergencia dada la no existencia de mínimos locales diferentes a la solución original. En cierto sentido el algoritmo puede considerarse un método tipo Newton globalmente convergente al estar basado en una linearización del problema. Se han estudiado la aproximación de ecuaciones diferenciales rígidas, de ecuaciones rígidas con retardo, de ecuaciones algebraico-diferenciales y de problemas hamiltonianos. Esperamos que esta nueva técnica variacional pueda usarse en otro tipo de problemas diferenciales.

Abstract

This thesis is devoted to the study and approximation of systems of differential equations based on an analysis of a certain error functional associated, in a natural way, with the original problem. We prove that in seeking to minimize the error by using standard descent schemes, the procedure can never get stuck in local minima, but will always and steadily decrease the error until getting to the original solution. One main step in the procedure relies on a very particular linearization of the problem, in some sense it is like a globally convergent Newton type method. We concentrate on the approximation of stiff systems of ODEs, DDEs, DAEs and Hamiltonian systems. In all these problems we need to use implicit schemes. We believe that this approach can be used in a systematic way to examine other situations and other types of equations.

Contents

Chapter 1

Introduction

To achieve higher order of accuracy in the approximation of Cauchy problems, multistep methods, such as Adams-Bashforth methods, were developed. These methods use other methods to generate enough data to start time marching. Runge-Kutta (RK) methods offer an alternative to multistep methods for higher order of accuracy in time. In RK methods, the value of the dependent variable at the end of any time step is calculated from its value at the beginning of the time step. For a desired order of accuracy, RK methods are more stable when compared with multi-point methods of same accuracy [51]. The classical explicit RK methods can be used to achieve high order accuracy, but they are restricted by stability constraints on time-step size. Especially for the approximation of stiff ODEs, explicit RK methods are not suitable [66].

The stiffness property of a system of ODE's cannot be defined in precise mathematical terms. Following Lambert [78], stiffness occurs when stability requirements, rather than those of accuracy, constrain the step-length, or when some components of the solution decay much more rapidly than others. For these and others related statements, a classical recommendation is to use, in general, implicit schemes when we are interested in stiff problems.

A desired property of any numerical scheme is A-stability. As pointed out by Dahlquist [38], the order of convergence of an A-stable linear multistep method cannot exceed two (*the second Dahlquist barrier*). It is much easier to find implicit RK methods with desired stability properties as A-stability. Some classical examples are Gauss' family, Randau's family and Lobatto's family that are based on some quadrature formulas [66]. These examples are collocation methods.

A number of convergence results have been derived for the discretization of nonlinear stiff initial problems. For RK methods the concept of B-stability was essential.

Further concepts were introduced and led to the so-called B-convergence theory. Most of these results are valid for stiff problems satisfying some one-sided Lipschitz condition. In [22]-[23] the authors extend the B-convergence theory to be valid for a class of nonautonomous weakly nonlinear stiff systems; reference to the (potentially large) one-sided Lipschitz constant is avoided, in particular, including **the linear case**. Unique solvability of the system of algebraic equations is shown, and global error bounds are derived. As point out by the same authors, it is not clear if it is possible to cover in a satisfactorily way highly nonlinear stiff problems, i.e., problems where also the nonlinear terms are affected by large parameters. Moreover, any result should assume that, in each step, the associated nonlinear system is well approximated. In particular, that we are able to start with a good initial guess for the iterative scheme. This might be very restrictive for many stiff problems.

The aim of this thesis is to present and study a new alternative approach for the analysis and numerical simulation of differential equations. We believe that this approach can be used in a systematic way to examine many situations and many types of equations due to its flexibility and its simplicity.

The approach has a variational nature. For a given problem, we associate to it, in a natural way, an error functional. We prove that in seeking to minimize the error by using standard descent schemes, the procedure can never get stuck in local minima, but will always and steadily decrease the error until getting to solution of the original problem. This particular variational linearization can be seen as a global convergent Newton type method.

Due to the importance of solving, in the same way demanding problems, we should be up to date with all the high-order methods currently available. A chapter about a new family of high order resolution methods can be found (Chapter 6) in order to make comparisons between our variational technique and the classical implementations of implicit methods.

A very remarkable feature of the thesis is that all the chapters are self-independent and autonomous, so that they can be self-studied.

The thesis has 7 chapters and 2 annexes that are more or less self-contained.

1. Chapter 1. This chapter is the starting point in the thesis and we include it with all the details in order to have a self-contained manuscript. It studies regular ODEs both theoretically and numerically. Starting with an initial approximation to the solution, we improve it, adding the solution of some associated linear problems, in such a way that the error is significantly decreased.

2. Chapter 2. This chapter generalizes the previous one in order to approximate stiff delay differential equations.

3. Chapter 3. In this chapter we present a general existence theorem for DAEs. We analyze some applications and numerical experiments using a step descent strategy.

4. Chapter 4. This chapter deals with the approximation of systems of differential-algebraic equations based on an analysis of a certain error functional.

5. Chapter 5. We present some ideas of our current work related with two topics. A step-variable implementation of our approach and the approximation of Hamiltonian Systems.

6. Chapter 6. In this chapter we study a family of high order iterative methods under Kantorovich conditions.

7. Chapter 7. This chapter deals with the final conclusions.

8. Annexe I. This annexe presents a collection of problems where we can test the real behavior of a new differential solver like our new approach.

9. Annexe II. This annex deals with Reciprocal Polynomial Extrapolation vs Richardson Extrapolation.

Chapter 2

The starting point: On an alternative approach for the analysis and numerical simulation of ODEs

Abstract 2.0.1 *This chapter is devoted to the study and approximation of systems of ordinary differential equations based on an analysis of a certain error functional associated, in a natural way, with the original problem. It is the starting point of the thesis but it is not a original part. It is based on the previous paper [15], but some of the numerical experiments have been obtained during the preparation of this thesis. We prove that in seeking to minimize the error by using standard descent schemes, the procedure can never get stuck in local minima, but will always and steadily decrease the error until getting to the solution of the original problem. One main step in the procedure relies on a very particular linearization of the problem: in some sense, it is like a globally convergent Newton type method. Although our objective here is not to perform a rigorous numerical study of the method, we illustrate its potential for approximation by considering some stiff systems of equations. The performance is astonishingly very good due to the fact that we can use very robust methods to approximate linear stiff problems like implicit schemes. We also include a couple of typical test models for the Lorentz system and the Kepler problem, again confirming a very good performance (see the Chapter 6). We believe that this approach can be used in a systematic way to examine other situations and other types of equations due to its flexibility and its simplicity.*

2.1 Motivation

The ideas we would like to introduce for the treatment of ODEs are based on an analysis of a certain error functional of the form

$$E(x) = \frac{1}{2}\int_0^T |x'(t) - f(x(t))|^2\, dt,$$

to be minimized among the absolutely continuous paths $x : (0,T) \to \mathbb{R}^N$ with $x(0) = x_0$. Note that if $E(x)$ is finite for one such path x, then automatically x' is square integrable. This error functional is associated, in a natural way, with the Cauchy problem

$$x'(t) = f(x(t)) \text{ in } (0,T), \quad x(0) = x_0. \tag{2.1.1}$$

We will focus on this paradigmatic problem for explicitness, though our ideas can be used and extended for more general situations, as will be pointed out later.

One main assumption on $f : \mathbb{R}^N \to \mathbb{R}^N$ is its smoothness, so that $\nabla f : \mathbb{R}^N \to \mathbb{R}^{N\times N}$ is continuous, and its global Lipschitzianity with Lipschitz constant $M > 0$ ($|\nabla f| \le M$).

The main reason why we believe this is an interesting point of view to study and approximate ODEs is the following statement. We put, for $0 \le t \le s \le T$,

$$E_{t,s}(z) = \frac{1}{2}\int_t^s |z'(\tau) - f(z(\tau))|^2\, d\tau, \quad E_t(z) = E_{0,t}(z), \quad E(z) = E_T(z).$$

Claim 2.1.1 *Let x be the unique solution of (2.1.1), and let z be absolutely continuous in $[0,T]$, but otherwise arbitrary. Then for every $t \in [0,T]$, we have*

$$|x(t) - z(t)|^2 \le 2e^{2M}\left(|x_0 - z(0)|^2 + 2tE_t(z)\right),$$

where M is the Lipschitz constant for f. As a consequence, one also has

$$\|x' - z'\|^2_{L^2(0,T)} \le 2\left(3 + M^2T^2e^{2M}\right)E(z) + 2Me^{2M}T|x_0 - z(0)|^2.$$

The proof is completely elementary. We will prove a more general statement later.

We clearly see that if z is feasible so that $z(0) = x_0$, then the **global** error of z over the full interval $(0,T)$, as an approximation of the solution x, is measured, through the error functional E, by the departure of z from being a solution of the Cauchy problem, namely

$$|x(t)-z(t)|^2 \le 4te^{2M}E_t(z) \text{ for } t \in (0,T), \quad \|x'-z'\|^2_{L^2(0,T)} \le 2\left(3 + M^2T^2e^{2M}\right)E(z).$$

It, therefore, looks like a promising strategy **to search for approximations of the solution x by minimizing the error E**. We believe this perspective can be helpful both for the analysis and the numerical simulation of typical Cauchy problems like (2.1.1). As a matter of fact, we will place ourselves in a situation where we pretend not to know anything about problem (2.1.1), and build and recover existence results and numerical procedures from scratch by following this minimization strategy.

We do not obtain, from this perspective, any finer existence results than the classical ones. Indeed, the analytical part of this contribution requires a further property on the map f: for every positive $C > 0$ and small $\epsilon > 0$, there is $D_{C,\epsilon} > 0$ so that

$$|f(x+y) - f(x) - \nabla f(x)y| \leq D_{C,\epsilon}|y|^2, \quad |x| \leq C, |y| \leq \epsilon.$$

This extra regularity is somehow not surprising as our approach here is based on regularity and optimality. On the other hand, that regularity holds for most of the important problems in applications. It certainly does in all numerical tests performed in this work. See however [13] for a similar analysis under more general assumptions. Our emphasis here is placed on the fact that this optimization strategy may be utilized to set up approximation schemes based on the minimization of the error functional. Indeed, the analytical part is oriented towards providing a solid basis for this approximation procedure. In this regard, the following proposition is also crucial as it states that typical minimization schemes like (steepest) descent methods will work fine as they can never get stuck in local minima, and converge steadily to the solution of the problem, no matter what the initialization is. This is also a fundamental fact for our approach.

Claim 2.1.2 *Let $\overline{x}$ be a critical point for the error E. Then $\overline{x}$ is the solution of the Cauchy problem (2.1.1).*

Proof 2.1.3 *The proof is elementary. Based on the smoothness and bounds assumed on the mapping f, we conclude that if $x \equiv \overline{x}$ is a critical point for the error E, then x ought to be a solution of the problem*

$$-\frac{d}{dt}\left(x'(t) - f(x(t))\right) - \left(x'(t) - f(x(t))\right)\nabla f(x(t)) = 0 \text{ in } (0,T),$$
$$x(0) = x_0, x'(T) - f(x(T)) = 0.$$

The vector-valued map $y(t) = x'(t) - f(x(t))$ is then a solution of the problem

$$-y'(t) - y(t)\nabla f(x(t)) = 0 \text{ in } (0,T), \quad y(T) = 0.$$

The only solution of this problem is $y \equiv 0$, and so x is the solution of our Cauchy problem.

We will focus on two main descent strategies that are introduced in Section 2. One is a kind of a globally convergent Newton method since it relies on our ability to solve or approximate linear problems; the other one is a steepest descent method with respect to a suitable norm. We will show in theory (Sections 3 and 4), and in practice (the rest of the paper), that both methods are valid strategies to approximate the solution of the initial Cauchy problem. We have selected several academic and typical test problems to show the good performance of the procedure, though our numerical comments do not pretend to go any further than here. A good feature of this point of view is that, due to Claim 2.1.1, the error is always a sure indication of whether we are getting close to the solution: if the error is small, the simulation is going right; if it is not, there is no way we can be close to the solution. This is even so regardless of whether there are theoretical results to support our computations. Section 6 is devoted to some numerical tests where we perform a comparison with the standard implementation of implicit schemes. We concentrate on the approximation of stiff systems of equations since we can use very robust methods to approximate linear stiff problems like implicit methods. We believe that this approach can be used in a systematic way to examine other situations or problems in which very efficient methods in the linear case are available.

We should mention that we have already explored in some previous papers this point of view. Since the initial contribution [13], we have also treated the reverse mechanism of using first discretization and then optimality ([12]). The perspective of going through optimality and then discretization has already been indicated and studied in [94], though only for the steepest descent method, and without going through any further analytical foundation for the numerical procedure. We also plan to address, from this viewpoint, other problems, some of them in the next chapters.

Variational methods have been used before in the context of ODEs. See ([85, 91]), where numerical integration algorithms for finite-dimensional mechanical systems that are based on discrete variational principles are proposed and studied. This is one approach to deriving and studying symplectic integrators, and indeed one that yields much insight into the geometry of the continuous and the discrete problem. The starting point is Hamilton's principle and its direct discretization. In those references, some fundamental numerical methods are presented from that variational viewpoint where the model plays a prominent role.

2.2 Two main descent procedures

Suppose we start with an initial crude approximation $x_{(0)}(\equiv x)$ to the solution of our basic problem (2.1.1). We could take $x_{(0)} = x_0$ for all t, or $x_{(0)}(t) = x_0 +$

$tf(x_0)$. We would like to improve this approximation in such a way that the error is significantly decreased. We have already pointed out in the above section that descent methods can never get stuck on anything but the solution of the problem, under global lipschitzianity hypotheses.

It is straightforward to find the Gâteaux derivative of E at a given feasible x in the direction y with $y(0) = 0$. Namely

$$E'(x)y = \int_0^T (x'(t) - f(x(t))) \cdot (y'(t) - \nabla f(x(t))y(t))\, dt.$$

This expression suggests two main possibilities to select y from:

1. Choose y such that

$$y'(t) - \nabla f(x(t))y(t) = f(x(t)) - x'(t) \text{ in } (0,T), \quad y(0) = 0.$$

In this way, it is clear that $E'(x)y = -2E(x)$, and so the (local) decrease of the error is of the size $E(x)$. Finding y requires solving the above linear problem. In some sense, this is like a Newton method.

2. We can select the steepest descent direction y, with respect to various norms, in the form

$$\min_{\|y\|=1} E'(x)y,$$

or equivalently

$$\min_y \left(\frac{1}{2}\|y\|^2 + E'(x)y \right).$$

Typical choices are

$$\|y\|^2 = \int_0^T |y'(t)|^2\, dt, \quad \|y\|^2 = \int_0^T \left(|y'(t)|^2 + |y(t)|^2 \right)\, dt,$$

but they can also be dependent on the current iteration x, like

$$\|y\|^2 = \int_0^T \left(|y'(t)|^2 + |\nabla f(x(t))y(t)|^2 \right)\, dt.$$

One very attractive feature of one of these choices over the others is if steepest descent directions can be given explicitly. For instance, the solution of the regular variational problem of minimizing over y the functional

$$\int_0^T \left(\frac{1}{2}|y'(t)|^2 + \frac{1}{2}|y(t)|^2 + (x'(t) - f(x(t))) \cdot (y'(t) - \nabla f(x(t))y(t)) \right)\, dt$$

yields the steepest descent direction for the norm

$$\|y\|^2 = \int_0^T \left(|y'(t)|^2 + |y(t)|^2\right)\, dt.$$

However, the minimizer y has to be calculated as the solution of the problem

$$-\frac{d}{dt}\left((x'(t) - f(x(t))) + y'(t)\right) + (y(t) - \nabla f(x(t)))^T \left((x'(t) - f(x(t)))\right) = 0 \text{ in } (0,T),$$

together with $y(0) = 0$, and the transversality condition

$$x'(T) - f(x(T)) + y'(T) = 0.$$

This is a linear, second-order, boundary value problem whose solution would have to be approximated. If, instead, we focus on finding the steepest descent direction with respect to the norm (recall that $y(0) = 0$)

$$\|y\|^2 = \int_0^T |y'(t)|^2\, dt,$$

then such direction is given as the minimizer for the functional (subject to $y(0) = 0$)

$$\int_0^T \left(\frac{1}{2}|y'(t)|^2 + (x'(t) - f(x(t))) \cdot (y'(t) - \nabla f(x(t))y(t))\right)\, dt.$$

This time the minimizer is given as the solution of the problem

$$-\frac{d}{dt}\left((x'(t) - f(x(t))) + y'(t)\right) - \left(\nabla f(x(t))^T((x'(t) - f(x(t))\right) = 0 \text{ in } (0,T),$$

together with $y(0) = 0$, and the transversality condition

$$x'(T) - f(x(T)) + y'(T) = 0.$$

It can be written in a fully explicit form

$$\begin{aligned} y(t) = &- \int_0^t \left(\mathbf{1} - s\nabla f(x(s))^T\right)(x'(s) - f(x(s)))\, ds \\ &+ t\int_t^T \nabla f(x(s))^T (x'(s) - f(x(s)))\, ds. \end{aligned} \tag{2.2.1}$$

Here $\mathbf{1}$ is the identity matrix of size N. It is straightforward to check that the local decrease of the error, when we select this direction y, is given by

$$-\int_0^T |y'(s)|^2\, ds.$$

We therefore explore these two (apparently different) variants of our iterative approach to the numerical approximation of ODEs. For a given approximation x, compute y by

1. solving (approximating)

$$y'(t) - \nabla f(x(t))y(t) = f(x(t)) - x'(t) \text{ in } (0,T), \quad y(0) = 0,$$

and update x to $x + y$ until convergence;

2. putting

$$\begin{aligned} y(t) = &- \int_0^t \left(\mathbf{1} - s\nabla f(x(s))^T\right)(x'(s) - f(x(s)))\, ds \\ &+ t\int_t^T \nabla f(x(s))^T(x'(s) - f(x(s)))\, ds, \end{aligned}$$

and update x to $x + \epsilon y$ for small ϵ. We will come back to this.

However the procedure we use to update the current approximation x to some new approximation, the global error over the full interval $(0,T)$ is always bound from above by some fixed constant (depending on f and T) times the error E according to Claim 2.1.1.

We would like to explicitly explore the advantages and disadvantages of both schemes.

2.3 Some analysis

Let us focus on the first possibility, pretending not to know anything about the solution of the Cauchy problem (2.1.1). Suppose x is a feasible path in the interval $(0,T)$ so that $x(0) = x_0$, x' is square integrable, and the quantity

$$E(x) = \frac{1}{2}\int_0^T |x'(t) - f(x(t))|^2\, dt$$

measures how far such x is from being a solution of our problem. We also assume that $|x(t)| \le C$ for a fixed constant C, and all $t \in (0,T)$. Choose $\epsilon > 0$ and $0 < \alpha < 1$ so that

$$\frac{\epsilon}{1 - \sqrt{\alpha}} \le C,$$

and

$$|f(z+y) - f(z) - \nabla f(z)y| \leq D|y|^2, \quad |y| \leq \epsilon, |z| \leq 2C,$$

for some constant $D > 0$. We then solve for y as the solution of the non-autonomous linear Cauchy problem

$$y'(t) - \nabla f(x(t))y(t) = f(x(t)) - x'(t) \text{ in } (0,T), \quad y(0) = 0,$$

and pretend to update x to $x + y$ in such a way that the error for $x + y$ be less than the error for the current iteration x. Note that

$$E(x+y) = \frac{1}{2}\int_0^T |x'(t) + y'(t) - f(x(t)+y(t))|^2\, dt \tag{2.3.1}$$

$$= \frac{1}{2}\int_0^T |f(x(t)+y(t)) - f(x(t)) - \nabla f(x(t))y(t)|^2\, dt,$$

where we have used the differential equation satisfied by y. By our assumption on f above,

$$|f(x(t)+y(t)) - f(x(t)) - \nabla f(x(t))y(t)| \leq D|y(t)|^2, \quad t \in (0,T), \tag{2.3.2}$$

provided that $|y(t)| \leq \epsilon$. Since we know that y is the solution of a certain linear problem, we have the upper bound

$$|y(t)|^2 \leq Te^{2M}E(x) \text{ for all } t \in [0,T]. \tag{2.3.3}$$

Assume we select $T > 0$ so small that

$$E_{0,T}(x) \equiv E(x) \leq \frac{\epsilon^2}{e^{2M}T}, \tag{2.3.4}$$

and then $|y(t)| \leq \epsilon$ for all $t \in [0,T]$. By (2.3.1), (2.3.2), and (2.3.3), we can write

$$E(x+y) \leq \frac{D^2}{2}\int_0^T |y(t)|^4\, dt \leq \frac{D^2}{2}e^{4M}T^3E(x)^2. \tag{2.3.5}$$

If, in addition, we demand, by making T smaller if necessary,

$$E(x) \leq \frac{2\alpha}{D^2T^3e^{4M}}, \tag{2.3.6}$$

then $E(x+y) \leq \alpha E(x)$. Moreover, for all $t \in (0,T)$,

$$|x(t) + y(t)| \leq C + \epsilon \leq 2C.$$

All these calculations form the basis of a typical induction argument. Write $y_0 \equiv x$, and suppose, by induction, that we have

$$\left|\sum_{i=0}^{j-1} y_i(t)\right| \leq C + \epsilon\left(\sum_{i=0}^{j-2}\sqrt{\alpha}^i\right)(\leq 2C), \quad |y_{j-1}(t)| \leq \epsilon\sqrt{\alpha}^{j-2} \text{ for all } t \in [0,T],$$

$$E\left(\sum_{i=0}^{j-1} y_i\right) \leq \alpha^{j-1}E(x)(\leq E(x)).$$

Determine y_j as the unique solution of the linear problem

$$y_j'(t) - \nabla f\left(\sum_{i=0}^{j-1} y_i(t)\right) y_j(t) = f\left(\sum_{i=0}^{j-1} y_i(t)\right) - \sum_{i=0}^{j-1} y_i'(t) \text{ in } (0,T), \quad y_j(0) = 0.$$

Then, by the estimates above and the induction hypothesis, we can write, bearing in mind (2.3.4),

$$|y_j(t)|^2 \leq Te^{2M}E\left(\sum_{i=0}^{j-1} y_i\right) \leq Te^{2M}E(x)\alpha^{j-1} \leq \epsilon^2\alpha^{j-1}.$$

From here, we certainly have

$$\left|\sum_{i=0}^{j} y_i(t)\right| \leq C + \epsilon\left(\sum_{i=0}^{j-1}\sqrt{\alpha}^i\right)(\leq 2C).$$

Similarly, by (2.3.5),

$$E\left(\sum_{i=0}^{j} y_i\right) \leq \frac{D^2}{2}e^{4M}T^3E\left(\sum_{i=0}^{j-1} y_i\right)^2 \leq \frac{D^2}{2}e^{4M}T^3E\left(\sum_{i=0}^{j-1} y_i\right)\alpha^{j-1}E(x) \leq \alpha^j E(x).$$

It is therefore clear that the sum

$$\sum_{i=0}^{\infty} y_i(t)$$

converges strongly in $L^\infty(0,T)$ to a solution of our initial Cauchy problem in a small interval $(0,T)$. Since the various ingredients of the problem do not depend on T, we can proceed to have a global solution in a big interval by successively performing this analysis in intervals of appropriate small size. For instance, we can always divide

a global interval $(0,T)$ into a certain number n of subintervals of small length h ($T = nh$) with

$$\frac{E_{0,T}(x)D^2e^{4M}}{2\alpha} \leq \frac{1}{h^3},$$

according to (2.3.6).

The uniqueness of the solution is a direct consequence of a straightforward generalization of Claim 2.1.1.

Proposition 2.3.1 *Let y and z be two admissible paths for our error functional. Then for every $t \in (0,T)$*

$$|y(t) - z(t)|^2 \leq 4e^{2M}t\left(E_t(y) + E_t(z)\right),$$

where M is the Lipschitz constant for f. As a consequence, one also has

$$\|y' - z'\|^2_{L^2(0,T)} \leq 2\left(3 + M^2T^2e^{2M}\right)\left(E(y) + E(z)\right).$$

Proof

Let y and z be feasible for the error functional, and set $x = y - z$ so that $x(0) = 0$. Then it is elementary to have for each $t \in (0,T)$

$$|x(t)| \leq \int_0^t |x'(s)|\,ds \leq \int_0^t \left(|y'(s) - f(y(s))| + |z'(s) - f(z(s))| + |f(y(s)) - f(z(s))|\right)\,ds,$$

and so, by using Holder's inequality in the first two contributions,

$$|x(t)| \leq 2\sqrt{t}\left(E_t(y) + E_t(z)\right) + M\int_0^t |x(s)|\,ds.$$

By Gronwall's lemma, we conclude

$$|x(t)| \leq 2e^M\sqrt{t}\left(E_t(y) + E_t(z)\right).$$

For the inequality for the L^2-norm of the derivative of the difference, go back to the first inequality above, and write

$$\int_0^T |x'(s)|^2\,ds \leq 3\int_0^T \left(|y'(s) - f(y(s))|^2 + |z'(s) - f(z(s))|^2 + |f(y(s)) - f(z(s))|^2\right)\,ds.$$

Then use the inequality we have just obtained for $|x(t)|$, to have

$$\int_0^T |x'(s)|^2\,ds \leq 6\left(E(y) + E(z)\right) + M^2\int_0^T 4e^{2M}s\left(E(y) + E(z)\right)\,ds.$$

By direct integration in the second term, we get the second inequality in the statement.

□

We can sum up our work in this section in the following statement.

Theorem 2.3.2 *For T sufficiently small, the iterative procedure $x^{(j)} = x^{(j-1)} + y^{(j)}$, starting from arbitrary feasible $x^{(0)}$, where*

$$(y^{(j)})'(t) - \nabla f(x^{(j-1)}(t))y^{(j)}(t) = f(x^{(j-1)}(t)) - (x^{(j-1)})'(t) \text{ in } (0,T), \quad y^{(j)}(0) = 0,$$

converges strongly in $L^\infty(0,T)$ and in $H^1(0,T)$ to the unique solution of the Cauchy problem (2.1.1).

This is a local existence result. If our hypotheses on f are global, then, as before, nothing can prevent us from applying this theorem to successive small intervals so that global existence of a unique solution is thus obtained.

2.4 The steepest descent strategy

We have proved in the preceding section that the initial Cauchy problem (2.1.1) has a unique solution provided that the mapping f is $\mathcal{C}^2$, and has a uniformly bounded derivative. There is nothing new about this result, except for the variational perspective of the proof that also allows for a very clear approximation strategy. One main step in the procedure relies on our ability to solve or approximate linear problems of the form

$$y'(t) - \nabla f(x(t))y(t) = f(x(t)) - x'(t) \text{ in } (0,T), \quad y(0) = 0, \tag{2.4.1}$$

for x given. Suppose we were to find an approximation of this linear problem by using the steepest descent strategy, the second possibility of Section 2.

Proposition 2.4.1 *Formula (2.2.1) is exactly the steepest descent direction of problem (2.4.1) at $y \equiv 0$.*

This fact is hardly in need of proof. It is just a matter of going through the algebra. It represents a very clear way of bringing together the two possibilities described in Section 2 as one main underlying optimization strategy.

What happens then if we use the direction y given in (2.2.1), instead of the solution of (2.4.1), to update a given approximation x to $x + y$, taking $\epsilon = 1$?

Theorem 2.4.2 *Under our assumptions on the map f, if T is sufficiently small, the iterative procedure $x^{(j)}(t) = x^{(j-1)}(t) + y^{(j)}(t)$ where*

$$y^{(j)}(t) = -\int_0^t \left(\mathbf{1} - s\nabla f(x^{(j-1)}(s))^T\right)\left((x^{(j-1)})'(s) - f(x^{(j-1)}(s))\right) ds + t\int_t^T \nabla f(x^{(j-1)}(s))^T((x^{(j-1)})'(s) - f(x^{(j-1)}(s)))\, ds,$$

converges strongly in $H^1(0,T;\mathbb{R}^N)$ and in $L^\infty(0,T;\mathbb{R}^N)$ to the unique solution of problem (2.1.1).

Proof

The proof consists in the realization that we can redo all of Section 3 with y given by (2.2.1) instead of being the solution of (2.4.1) because it is easy to check that, directly from (2.2.1), we also have for every $t \in (0,T)$

$$|y(t)|^2 \leq T\left(1 + \frac{M^2T^2}{3} + \frac{T^4M^2}{3}\right)E(x).$$

As a matter of fact, the computation in Section 3 are always valid for every choice of the update direction y as long as

$$|y(t)|^2 \leq C(M,T)E(x),$$

for every $t \in (0,T)$.

□

Note that the steepest descent method does not rely on the solution of the auxiliary linear problem, or of any other ODE.

2.5 Numerical procedure

Since our optimization approach is really constructive, iterative numerical procedures are easily implementable.

For the Newton-like method:

1. Start with an initial approximation $x^0(t)$ compatible with the initial conditions (ex. $x^0(t) = x_0 + tf(x_0)$).

2. Assume we know the approximation $x^{(j)}(t)$ in $[0,T]$.

3. Compute its derivative $(x^{(j)})'(t)$.

4. Compute the auxiliar function $y^{(j+1)}(t)$ as the numerical solution of the problem

$$y'(t) - \nabla f(x^{(j)}(t))y(t) = f(x^{(j)}(t)) - (x^{(j)})'(t) \text{ in } (0,T), \quad y(0) = 0,$$

by making use of a numerical scheme for ODEs with dense output (like collocation methods, see next section).

5. Change $x^{(j)}$ to $x^{(j+1)}$ by using the update formula

$$x^{(j+1)}(t) = x^{(j)}(t) + y^{(j+1)}(t).$$

6. Iterate (3), (4) and (5), until numerical convergence.

In particular, one can implement, in a very easy way, this numerical procedure using a problem-solving environment like MATLAB [119].

The steepest descent strategy has already been indicated and tested in [94], though using a small parameter ϵ, determined experimentally, to update each iteration. We now know that we can always take $\epsilon = 1$ at the expense of taking T small enough. The iterative procedure has the same structure as the one just indicated.

1. Initialization. Take any simple initial guess $x^{(0)}(t)$ complying with the initial condition $x^{(0)}(0) = x_0$. For instance,

$$x^{(0)}(t) \equiv x_0, \text{ or } x^{(0)}(t) = x_0 + tf(x_0).$$

2. Update. If iterate $x^{(j)}(t)$ is at our disposal, generate $x^{(j+1)}(t)$ until convergence through the formula

$$\begin{aligned} x^{(j+1)}(t) = x^{(j)}(t) - \int_0^t \left(\mathbf{1} - s\nabla f(x^{(j)}(s))^T\right) \left((x^{(j)})'(s) - f(x^{(j)}(s))\right) ds \\ + t\int_t^T \nabla f(x^{(j)}(s))^T ((x^{(j)})'(s) - f(x^{(j)}(s))) \, ds. \end{aligned}$$

In practice, we should perform a discretization of this procedure, and describe in precise terms how to go from one iteration to the next. In our numerical tests below, we have implemented a typical multistep, mid-point quadrature rule.

2.6 Some experiments and comparisons

We pretend to provide some first experimental evidence that our iterative schemes are competitive with respect to other methods, without pretending at this stage to provide any further numerical analysis for our approach. In this section, we focus on the iterative procedure that uses the auxiliary linear problems at each iteration, solving these with typical implicit methods.

High order accuracy and stability are major areas of interest in simulation. We perform here a comparison with the standard implementation of implicit methods that solve the associated nonlinear equations by using Newton type schemes. We consider some stiff problems well known in the literature.

2.6.1 A nonlinear test problem

Certain types of problems can be characterized as stiff. A clear example is the linear test problem $x'(t) = \lambda x(t)$, where λ is a complex number with Re $\lambda << 0$. The approximation of this problem imposes severe restrictions on the step size of explicit methods, and it is used to test A-stability. We would like to start this section by exploring specifically our strategy with a simple nonlinear generalization of this linear test problem.

We are interested in approximating the value $x(1)$ of the solution $x(t)$ of the problem

$$\begin{aligned} x'(t) &= \lambda x(t) + x^2(t), \\ x(0) &= 1, \end{aligned}$$

whose explicit solution is

$$x(t) = \frac{\lambda e^{\lambda t}}{1 + \lambda - e^{\lambda t}}.$$

We consider the trapezoid method

$$x_{n+1} = x_n + \frac{h}{2}(f(x_n) + f(x_{n+1})). \tag{2.6.1}$$

This method averages the Euler and backward Euler methods, advancing the approximate solution at each step along a line whose slope is the arithmetic mean of the derivatives at its endpoints.

In Figures 2.1 and 2.2, we plot the approximation using a classical implementation of the (implicit) trapezoid method, that is, solving the associated nonlinear equation using Newton's method. The method gives a bad approximation when the stiffness

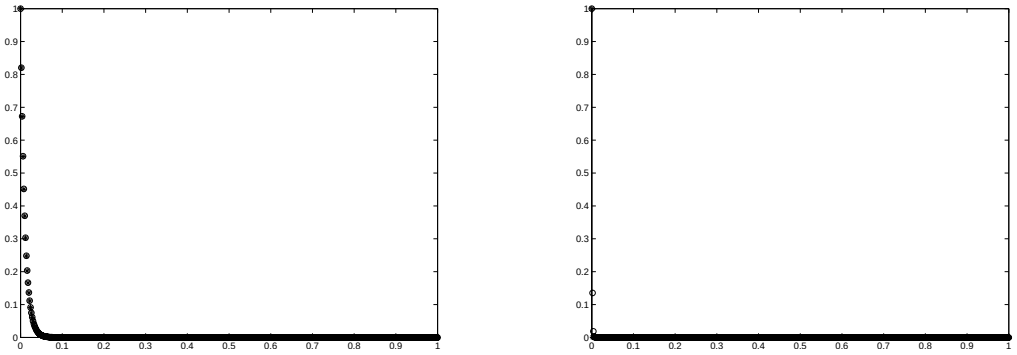

Figure 2.1: Classical implementation of the trapezoid method, nonlinear test problem, 'o'-original, '*'-approximation, left $\lambda = -100$ and right $\lambda = -1000$.

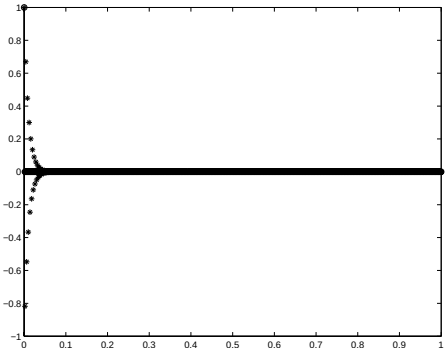

Figure 2.2: Classical implementation of the trapezoid method, nonlinear test problem, 'o'-original, '*'-approximation, $\lambda = -10000$.

of the problem increases (Figure 2.2). This fact indicates that we are outside of the basin of attraction of Newton's method [22]. However, looking at Figures 2.3 and

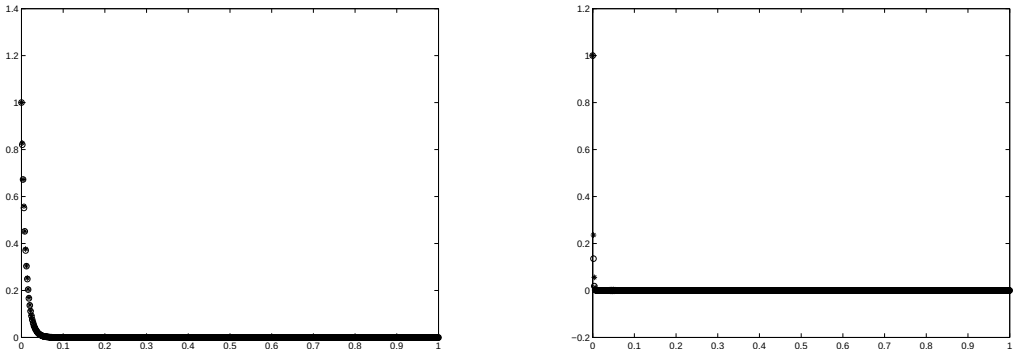

Figure 2.3: New implementation of the trapezoid method, nonlinear test problem, 'o'-original, '*'-approximation, left $\lambda = -100$ and right $\lambda = -1000$.

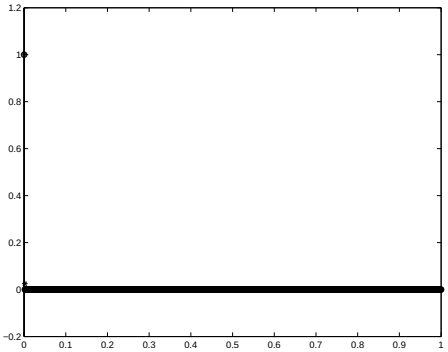

Figure 2.4: New implementation of the trapezoid method, nonlinear test problem, 'o'-original, '*'-approximation, $\lambda = -10000$.

2.4, we see that the linearization method

$$x'_{j+1}(t) - (\lambda + 2x_j(t))x_{j+1}(t) = \lambda x_j(t) + x_j(t)^2 - x'_j(t), \quad x_{j+1}(0) = 0,$$

with $x_0(t) = 1 + t(1 + \lambda)$, converges in all cases.

2.6.2 Linear vs nonlinear stiff problems

In this section, we analyze the numerical behavior, for linear and nonlinear problems, of the stiff solver *ode15s* (variable order solver based on the numerical differentiation formulas), that we can find in the *odeset* of MATLAB [119].

We consider the linear problem [116]

$$\begin{aligned} y_1'(t) &= \frac{\lambda_1 + \lambda_2}{2} y_1(t) + \frac{\lambda_1 - \lambda_2}{2} y_2(t), \\ y_2'(t) &= \frac{\lambda_1 - \lambda_2}{2} y_1(t) + \frac{\lambda_1 + \lambda_2}{2} y_2(t), \end{aligned}$$

with $\lambda_i < 0$, and the following (associated) nonlinear problem

$$\begin{aligned} y_1'(t) &= \frac{\lambda_1 + \lambda_2}{2} y_1(t) + \frac{\lambda_1 - \lambda_2}{2}\ y_2(t)\ + \lambda_3 y_1(t) y_2(t), \\ y_2'(t) &= \frac{\lambda_1 - \lambda_2}{2} y_1(t) + \frac{\lambda_1 + \lambda_2}{2}\ y_2(t)\ + \lambda_3 y_1(t) y_2(t). \end{aligned}$$

As we can see, in Figure 2.5, for the linear case, the method gives good results even for very large parameters. However, some serious problems occur in the nonlinear case Figure 2.6.

2.6.3 Collocation Polynomials

Let $h > 0$. Given different coefficients c_i, $1 \le i \le s$ there is a (unique for h sufficiently small) polynomial of collocation $q(t)$ of degree less than or equal to s such that

$$q(t_0) = y_0, \quad q'(t_0 + c_i\, h) = f(t_0 + c_i\, h, q(t_0 + c_i\, h)) \quad \text{if } 1 \le i \le s. \tag{2.6.2}$$

The collocation methods are defined by an approximation $y(t) \simeq q(t)$, and are equivalent to implicit RK methods of s stages

$$\begin{aligned} k_i &= f(t_0 + c_i\, h, y_0 + h \sum_{j=1}^{s} a_{i,j}\, k_j), \\ y_1 &= y_0 + h \sum_{i=1}^{s} b_i\, k_i, \end{aligned} \tag{2.6.3}$$

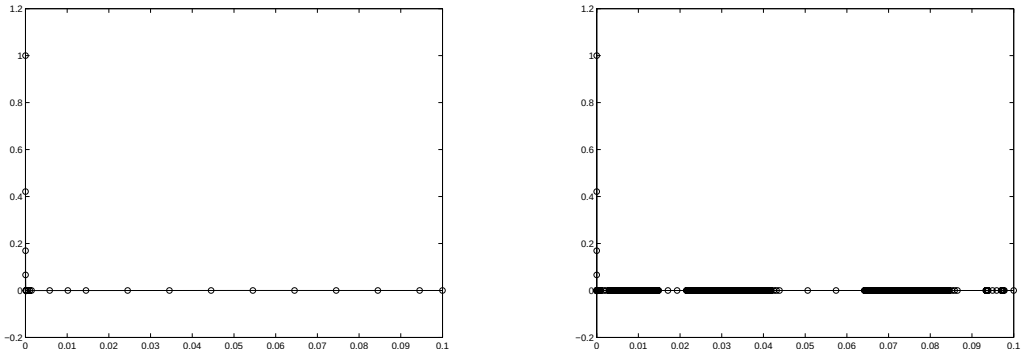

Figure 2.5: Second component of the linear problem. Left $\lambda_1 = -10^5$, $\lambda_2 = -10^{-5}$. Right $\lambda_1 = -10^{25}$, $\lambda_2 = -10^{-25}$.

for the coefficients

$$\begin{aligned} a_{i,j} &= \int_0^{c_i} \prod_{l \neq j} \frac{u - c_l}{c_j - c_l}\, du, \\ b_i &= \int_0^{1} \prod_{l \neq i} \frac{u - c_l}{c_i - c_l}\, du. \end{aligned} \tag{2.6.4}$$

The coefficients c_i play the role of the nodes of the quadrature formula, and the associated coefficients b_i are analogous to the weights. From (2.6.4) we can find implicit RK methods called Gauss of order $2s$, Radau IA and Radau IIA of order $2s-1$ and Lobatto IIIA of order $2s-2$. See [66] for more details.

A number of convergence results have been derived for the discretization of nonlinear stiff initial problems. For RK methods the concept of B-stability was essential. Further concepts were introduced and led to the so-called B-convergence theory. Most of these results are valid for stiff problems satisfying some one-sided Lipschitz condition. In [22]-[23] the authors extend the B-convergence theory to be valid for a class of nonautonomous weakly nonlinear stiff systems; reference to the (potentially large) one-sided Lipschitz constant is avoided, in particular, including **the linear case**. Unique solvability of the system of algebraic equations is shown, and global

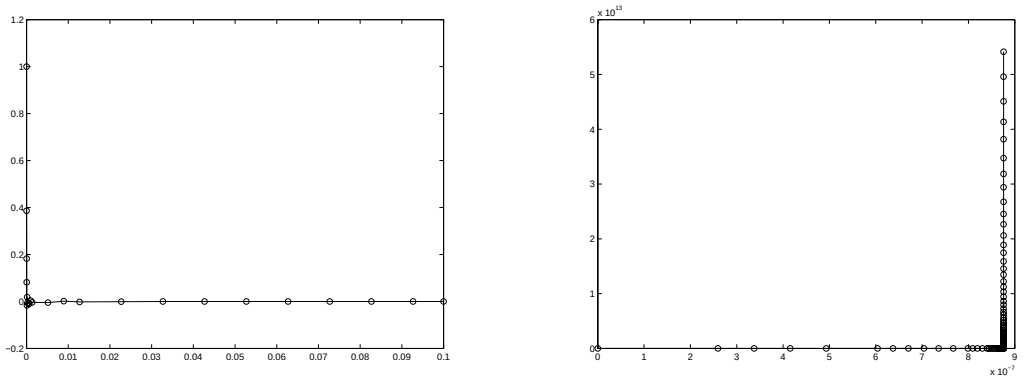

Figure 2.6: Second component of the nonlinear problem. Left $\lambda_1 = -10^5$, $\lambda_2 = -10^{-5}$, $\lambda_3 = 10^4$. Right $\lambda_1 = -10^5$, $\lambda_2 = -10^{-5}$, $\lambda_3 = 10^6$.

error bounds are derived. As point out by the same authors, it is not clear if it is possible to cover in a satisfactorily way highly nonlinear stiff problems, i.e., problems where also the nonlinear terms are affected by large parameters (see the above numerical example). Moreover, any result should assume that, in each step, the associated nonlinear system is well approximated. In particular, that we are able to start with a good initial guess for the iterative scheme. This might be very restrictive for many stiff problems (see Section 3.6.2).

A stiff problem: Chapman atmosphere

This model represents the Chapman mechanism for the generation of the ozone and the oxygen singlet. In this example, the concentration of the oxygen $y_3 = [O_2]$ will be held constant. It is a severe test for a stiff ODE package [34] governed by the following equations:

$$\begin{aligned} y_1'(t) &= 2k_3(t)y_3 + k_4(t)y_2(t) - (k_1y_3 + k_2y_2(t))y_1(t), \\ y_2'(t) &= k_1y_1(t)y_3 - (k_2y_1(t) + k_4(t))y_2(t), \end{aligned}$$

with $y_3 = 3.7 \times 10^{16}$, $k_1 = 1.63 \times 10^{-16}$, $k_2 = 4.66 \times 10^{-16}$,

$$k_i(t) = \begin{cases} \exp(\frac{-a_i}{\sin(\omega t)}), & \text{if } \sin(\omega t) > 0 \\ 0, & \text{otherwise} \end{cases}$$

for $i = 3, 4$, with $a_3 = 22.62$, $a_4 = 7.601$ and $\omega = \frac{\pi}{43200}$. The constant 43200 is 12 h measured in seconds. The initial conditions are $y_1(0) = 10^6$ and $y_2(0) = 10^{12}$.

This problem has important features like:

- The Jacobian matrix is not a constant.
- The diurnal effect is present.
- The oscillations are fast.
- The time interval used is fairly long, $0 \leq t \leq 8.64\ 10^5$, or 10 days.

We consider our approach with the implicit fourth order Gauss method ($s = 2$ as collocation method). We obtain a good approximation, see Figure 2.7. Note that $y_2 = [0_3]$ looks like a staircase with a rise at midday every day and $y_1 = [O]$ looks like a spike with its amplitude increases each day.

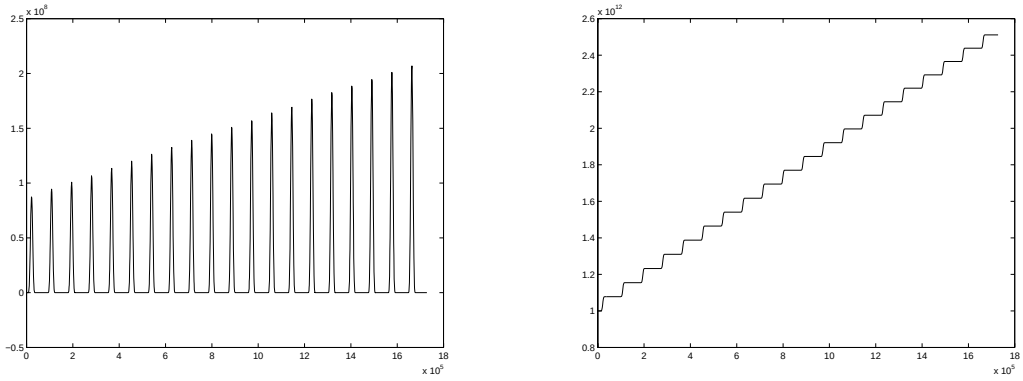

Figure 2.7: Chapman atmosphere approximated via our variational approximation. Left: first component. Right: second component.

Chapter 3

Linearizing Stiff Delay Differential Equations

Abstract 3.0.1 *This paper deals with the study and approximation of stiff delay differential equations based on an analysis of a certain error functional. In seeking to minimize the error by using standard descent schemes, the procedure can never get stuck in local minima, but will always and steadily decrease the error until getting to the solution sought. Starting with an initial approximation to the solution, we improve it, adding the solution of some associated linear problems, in such a way that the error is decreased. The performance is expected very good due to the fact that we can use very robust methods to approximate* **linear** *stiff delay differential equations.*

3.1 Introduction

Ordinary differential equations (ODEs) and delay differential equations (DDEs) are used to describe many physical models. While ODEs contain derivatives which depend on the solution at the present value of the independent variable, DDEs contain in addition derivatives which depend on the solution at previous times. For DDEs we must provide not just the value of the solution at the initial point, but also the solution at times prior to the initial point. Despite the obvious similarities between ODEs and DDEs, solutions of DDE problems can differ from solutions for ODE problems in several ways. One important thing is the presence of discontinuities in low-order derivatives. Generally there is a discontinuity in the first derivative of the solution at the initial point. Moreover, if the solution has a discontinuity in a derivative somewhere, there are discontinuities in the rest of the interval at a spacing

given by the delays.

A popular approach to solving DDEs is to extend one of the methods used to solve ODEs (see [24, 25, 46] and their references). Most of the codes are based on Runge-Kutta methods. The code dde23 [113] takes this approach by extending the method of the Matlab explicit ODE solver ode23. The code RADAR5 is developed in FORTRAN-90 and is based on an adaptation of the 3-stage Radau IIA method to stiff delay differential equations [57]. Stiff systems are prevalent in the study of damped oscillators, chemical reactions and electrical circuits. Although there have been numerous attempts to define stiffness, none seem quite satisfactory. One of them is the following "Stiff equations are problems for which explicit methods don't work" [66]. On the other hand, implicit schemes need to solve an auxiliary nonlinear system of equations in each step. These systems are approximated via Newton-type iterative methods. In particular, we have to be able to find a good initial guess inside the ball of convergence of the iterative scheme [18, 19, 20].

Recently ([13, 14, 12]), a variational approach for the analysis and approximation of Cauchy problems has been introduced. One main step in the procedure relies on a very particular linearization of the problem: in some sense, it is like a globally convergent Newton type method. The performance is astonishingly very good due to the fact that we can use very robust methods to approximate **linear** stiff problems like implicit collocation schemes. As point out in [22, 23], it is not clear if it is possible to cover in a satisfactorily way highly **nonlinear** stiff problems, i.e., problems where also the nonlinear terms are affected by large parameters. Moreover, any result should assume that, in each step, the associated nonlinear system is well approximated. In particular, that we are able to start with a good initial guess for the iterative scheme. This might be very restrictive for many stiff problems, however our variational approach gives good results in these cases, see the numerical section in [14]. In this paper, we extend this procedure to the case of DDEs.

The rest of the paper is divided in three sections. In Section 2 we introduce our variational approach for the linearization of DDEs. Section 3 introduces the numerical procedure and present a convergence analysis. Finally, we conclude with a small conclusion section including some further research directions.

3.1.1 A particular linearization of stiff DDEs via an error minimization problem

Let $C := C^1([-\tau, 0], \mathbb{R}^n)$ be the vector space of continuous differentiable functions mapping the interval $[-\tau, 0]$ into $\mathbb{R}^n$. The stiff problem we like to analyze can be

written as

$$\begin{aligned} x^{'}(t) &= f(x(t), x(t-\tau)), \quad t \in (0,T), && (3.1.1)\\ x(\theta) &= \phi(\theta), \quad \theta \in [-\tau, 0], && (3.1.2) \end{aligned}$$

where $\phi \in C$ specifies the initial condition and f is sufficiently smooth in both variables.

We consider the error functional

$$E(x) = \frac{1}{2}\int_0^T |x'(t) - f(x(t), x(t-\tau))|^2 \, dt,$$

to be minimized among the absolutely continuous paths $x : (0,T) \to \mathbb{R}^N$ with square-integrable derivative and such that $x(\theta) = \phi(\theta)$, $\theta \in [-\tau, 0]$.

It is straightforward to find the Gâteaux derivative of E at a given feasible x in the direction y with $y(\theta) = 0$, $\theta \in [-\tau, 0]$. Namely

$$E'(x)y = \int_0^T (x'(t) - f(x(t), x(t-\tau))) \cdot (y'(t) - \nabla_1 f(x(t), x(t-\tau))y(t) - \nabla_2 f(x(t), x(t-\tau))y(t-\tau)) \, dt,$$

where ∇_1 and ∇_2 denote the partial derivative with respect $x(t)$ and $x(t-\tau)$ respectively.

This expression suggests a nice possibility to select y from: Choose y such that

$$y'(t) - \nabla_1 f(x(t), x(t-\tau))y(t) - \nabla_2 f(x(t), x(t-\tau))y(t-\tau) = f(x(t), x(t-\tau)) - x'(t) \text{ in } (0,T),$$

with $y(\theta) = 0$, $\theta \in [-\tau, 0]$.

We have already pointed out that descent methods can never get stuck on anything but the solution of the problem, under global lipschitzianity hypotheses. The following proposition states that a minimization scheme will work fine as they can never get stuck in local minima, and converge steadily to the solution of the problem, no matter what the initialization is. This is also a fundamental fact for our approach.

Theorem 3.1.1 *Let $\bar{x}$ be a critical point for the error E. Then $\bar{x}$ is the solution of the problem (3.1.1).*

3.1.2 Numerical procedure

Our approach is really constructive and an iterative numerical procedure is easily implementable based. Mainly:

1. Start with an initial approximation $x^0(t)$ compatible with the initial conditions.
2. Assume we know the approximation $x^{(j)}(t)$ in $[0,T]$.
3. Compute its derivative $(x^{(j)})'(t)$.
4. Compute the auxiliary function $y^{(j+1)}(t)$ as the numerical solution of the problem (by making use of a numerical scheme for DDEs with dense output as RADAR5 [57])

$$y'(t)-\nabla_1 f(x^{(j)}(t),x^{(j)}(t-\tau))y(t)-\nabla_2 f(x^{(j)}(t),x^{(j)}(t-\tau))y(t-\tau) = f(x^{(j)}(t),x^{(j)}(t-\tau))-x'(t), \tag{3.1.3}$$

   in $(0,T)$, with $y(\theta)=0$, $\theta\in[-\tau,0]$.
5. Change $x^{(j)}$ to $x^{(j+1)}$ by using the update formula

$$x^{(j+1)}(t) = x^{(j)}(t) + y^{(j+1)}(t).$$

6. Iterate (3), (4) and (5), until numerical convergence ($||y^{(j)}|| \leq TOL$).

Assuming that the problem (3.1.1) has a unique solution and following [14], we can derive the convergence of this procedure:

Theorem 3.1.2 *The iterative procedure $x^{(j+1)} = x^{(j)}+y^{(j+1)}$, starting from arbitrary feasible $x^{(0)}$ compatible with the initial conditions, converges strongly in $L^\infty(0,T)$ and in $H^1(0,T)$ to the solution of (3.1.1) assuming that f is smooth enough.*

A main difference in the solution of delay equations compared to ordinary differential equations is the appearance of breaking points (jump discontinuities in the solution or in its derivatives) even in the presence of smooth functions. If the breaking points are not included in the mesh and a variable step size integration is used, the step sizes may be severely restricted near the low order jump discontinuities. Some algorithms are proposed for the detection and computation of breaking points in [58]. This paper includes theoretical results with regard to errors in the approximation of these important points. By construction, both the original problem (3.1.1)

and the auxiliary linear equation (3.1.3) have the same number of breaking points and in the same position.

If we use the algorithm proposed in [58] for the approximation of the linear equation (3.1.3) (without including the application of the Newton method since in our case the associated system of equations is linear) and combine the theoretical results of this paper with Theorem 7.3.2 we obtain the convergence of our full discretized algorithm:

Theorem 3.1.3 *With the notation and hypotheses of Theorem 7.3.2, if $\tilde{y}^{(j)}$ is the approximation of the sequence $y^{(j)}$ via RADAR5 with breaking point detection then for all $TOL > O(h^5)$ exists $j \in \mathbb{N}$ such that*

$$||y^{(j)}|| \leq TOL.$$

On the other hand, for the approximation of stiff problems implicit schemes are used [66]. A number of convergence results have been derived for the discretization of nonlinear stiff initial problems. In [22]-[23] the authors extend the B-convergence theory to be valid for a class of nonautonomous weakly nonlinear stiff systems; reference to the (potentially large) one-sided Lipschitz constant is avoided, in particular, including **the linear case**. Unique solvability of the system of algebraic equations is shown, and global error bounds are derived. As point out by the same authors, it is not clear if it is possible to cover in a satisfactorily way highly nonlinear stiff problems, i.e., problems where also the nonlinear terms are affected by large parameters. Moreover, any result should assume that, in each step, the associated nonlinear system is well approximated. In particular, that we are able to start with a good initial guess for the iterative scheme. This might be very restrictive for many stiff problems (see Section 6.2 of our recent work in [15]).

The results, as in the case of stiff ODEs [15], would be very satisfactory. For problems verifying the hypotheses of our theorems we obtain always the convergence to the true solution. Moreover, taking small tolerances (TOL) as stopping criterium, the exact and computed solutions should be indistinguishable in a first look [15]. The computational cost of the direct approximation of the stiff nonlinear DDE with an implicit scheme and with variational approach is similar. In each step of the implicit scheme we use a Newton iterative method to approximate the nonlinear system of equations. In our approach we use an iterative scheme to solve the minimization problem but in each iteration we only approximate linear system of equations.

3.1.3 Conclusions

In this paper we have presented a new variational approach of DDEs. The main step in the procedure relies on a very particular linearization of the problem. Therefore, the performance is expected to be very good due to the fact that we can use very robust methods to approximate **linear** stiff problems like implicit collocation schemes [58].

The main advantage of our approach is that we only need to approximate linear problems. We believe that this procedure can be used in a systematic way to examine other types of DDEs due to its flexibility and its simplicity. In particular, we are interesting in DDEs with multiple lags, in DDEs with non-constant lags and in neutral DDEs with lags in the derivatives. This is one of our current work (see Chapter 6 for others topics we are interesting in).

Chapter 4

A steepest descent strategy for the approximation of DAEs

Abstract 4.0.4 *This chapter deals with the study and approximation of systems of differential-algebraic equations based on an analysis of a certain error functional. In seeking to minimize the error by using standard descent schemes, the procedure can never get stuck in local minima, but will always and steadily decrease the error until getting to the solution sought. Starting with an initial approximation to the solution, we improve it, in such a way, that the error is significantly decreased. Some numerical examples are presented to illustrate the main theoretical conclusions.*

4.1 Motivation

Recently ([13]), a variational approach for a typical Cauchy problem for the differential system

$$F(t, x(t), x'(t)) = 0 \text{ in } (0, T), \quad x(0) = x_0,$$

has been proposed, based on the minimization of an error functional that vanishes precisely over solutions of such a problem. Namely, the error functional

$$E(x) = \int_0^T |F(t, x(t), x'(t))| \, dt,$$

measures the departure of feasible paths $x : (0, T) \to W^{1,1}(0, T; \mathbb{R}^n)$, $x(0) = x_0$, from being a solution of the problem. At the outset, no further requirements seem necessary so that implicit equations, differential inclusions, and differential-algebraic equations (DAEs) can be treated under this same framework. After all, solutions of

the initial-value problem above need to have zero error regardless of any other consideration. However, the main existence theorem in [13] asks for two main structural requirements that essentially rule out the application to singular problems when the matrix

$$\frac{\partial F}{\partial \xi}, \quad F = F(t, \lambda, \xi) : (0, T) \times \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^n,$$

may become singular. Those two essential requirements are:

1. coercivity in the form

$$|F(t, \lambda, \xi)| \geq C|\xi| - M_1|\lambda| - M_0,$$

   for positive constants C, M_1, and M_0;

2. regularity in the form

$$\frac{1}{h} \min_{\{z:F(s,y,z)=0\}} \int_s^{s+h} |F(t, y + (t-s)z, z)|\, dt \to 0, \text{ as } h \to 0,$$

   for every s, and y.

In a typical DAE problem, both requirements may fail to hold. Yet, we would like to stress that our approach based on minimizing an error functional is so flexible that allows for a general approach.

To emphasize our point, consider the smooth, quadratic error functional

$$E(x) = \frac{1}{2} \int_0^T |F(t, x(t), x'(t))|^2\, dt$$

defined for feasible paths $x : (0, T) \to H^1(0, T; \mathbb{R}^n)$ with $x(0) = x_0$. The reason to use power 2 instead of 1 is that we would like a smooth error functional, as our procedure is based on regularity and smoothness.

What is remarkable is that, from a practical point of view, if we are interested in numerical approximations of the potential solution, we can easily devise an iterative procedure based on a steepest descent (or conjugate gradient) strategy for the error functional. The successive iterations of such a procedure and the evolution of the error will tell us if we are getting close to a solution of the problem when the error decreases steadily to zero, as the iterations proceed. In this way, one can launch the approximation, and from the resulting iterations judge *a posteriori* if we are getting close to a solution, even if there are no analytical results available, or if, presumably, such a problem may lack solutions.

Remark 4.1.1 *A main ingredient in the proof makes use of a generalization of the typical compactness property for non-linear operators introduced recently in [95]. It has been called the non-finer oscillation (NFO) property, as it makes an attempt to convey the idea that the images of a sequence of iterates* $\{\mathbf{T}u_j\}$ *cannot produce finer oscillations than the ones already contanined in* $\{u_j\}$*, when such an operator enjoys this property. Indeed, one main application in such a contribution was the analysis of a regular Cauchy problem for a dynamical system (Section 6 in [95]). We have tried to expand that analysis to cover the singular case of DAEs. See [14] for all the details.*

One final important point to be emphasized is the fact that the theoretical and numerical analysis of DAEs are typically linked in a very fundamental way to its index. See, among others, [28], [35], [70], [72], [73], [98], [99], [102], [112].

4.2 Some applications

DAEs play nowadays an important role in mathematical modeling. Several types of DAEs are found in many applications. In this section, we select and review some of those typical examples.

4.2.1 Mechanical systems

The general form of a constrained mechanical system is given by ([98])

$$\begin{aligned} q^{'} &= u, \\ M(q)u^{'} &= f(q,u) - G^T(q)\lambda, \\ 0 &= g(q), \end{aligned}$$

where M is a positive definite matrix, $G(q) = \partial g/\partial q$ is a $m \times n$-matrix so that g takes values on $\mathbb{R}^m$, $q = (q_1, \ldots, q_n)^T$, $u = (q_1^{'}, \ldots, q_n^{'})^T$, $\lambda = (\lambda_1, \ldots, \lambda_m)^T$. This is a DAE on the variables $(q, u, \lambda) \in \mathbb{R}^{2n+m}$, $m < n$. Note that if the matrix $GM^{-1}G^T$ is invertible, the system is a Hessenberg index 3 problem.

4.2.2 Optimal control problems

Consider a linear optimal problem with quadratic cost ([66])

$$\begin{aligned} y^{'} &= Ay + Bu \text{ in } (0,1), \quad y(0) = y_0, \\ J(u) &= \frac{1}{2}\int_0^1 (y^T C y + u^T D u)dt, \end{aligned}$$

where C and D are symmetric and positive semi-definite. The optimal control u can be found by solving the following system

$$\begin{aligned} y' &= Ay + Bu, \\ v' &= -A^T v - Cy, \\ 0 &= B^T v + Du, \end{aligned}$$

in the interval $(0, 1)$, together with the initial condition $y(0) = y_0$, and the transversality condition $v(1) = 0$. The variable v is the associated costate.

Notice that the whole system is linear. If D vanishes and $B^T CB$ is positive definite, then it has index 3.

4.2.3 Chemical reactions

Chemical processes is another field where DAEs are prominent [115]. In this context, DAEs turn out to be quite often of a very high index with nasty nonlinearities. Chemical processes are modeled by describing the reaction rate, mass flow, and thermal dynamics, as well as the energy balance.

A general model can be formulated by

$$A(x(t), t)(D(t)x(t))' = b(x(t), t).$$

4.2.4 Electrical circuits

The simulation of electrical circuits is of great interest today. Circuits consist of a large number of elements, and the equations have to be generated automatically. There are two modern modeling techniques making an automatic generation: the classical approach, and the charge-oriented approach of the modified nodal analysis [49].

The classical modified nodal analysis furnishes systems of the form

$$D(x)x' + f(x) = r(t),$$

where the vector of unknowns x consists of

- the nodal potentials u, and
- the currents I of the voltage-controlled elements.

The system contains the equations derived by Kirchhoff's nodal law for each node. Additionally, the characteristic equations of the voltage-controlled elements belong to the system. The equations of the current-controlled elements are set into the system directly.

The charge-oriented modified nodal analysis leads to systems of the form

$$\begin{aligned} Aq' + f(x) &= r(t), \\ q - g(x) &= 0, \end{aligned}$$

Here the vector of unknowns (x, q) contains

- the nodal potentials u,
- the currents I of the voltage-controlled elements,
- the charge Q of the capacitors and
- the flux Φ of the inductors.

The last relation represents the characteristic equations for charge and flux.

In both situations, we have a linear-in-the-derivative, possibly-degenerate problem.

4.2.5 Fluid dynamics

Let us consider the Navier-Stokes system in the form

$$\begin{aligned} u_t + uu_x + vu_y + p_x - \nu(u_{xx} + u_{yy}) &= 0, \\ v_t + uv_x + vv_y + p_y - \nu(v_{xx} + v_{yy}) &= 0, \\ u_x + v_y &= 0. \end{aligned}$$

After a usual semi-discretization in space ([110]), we arrive at the following system

$$\begin{aligned} Mu' + (K + N(u))u + Cp &= f, \\ C^T u &= 0. \end{aligned}$$

If $C^T M^{-1} C$ is a nonsingular matrix with a bounded inverse, then the above system has index 2. It is again a system of Hessenberg type.

4.3 Some examples of DAEs with different index

An iterative numerical procedure is easily implementable based on the strategy of the steepest descent direction y above. Mainly:

1. Start with an initial approximation $x^0(t)$ compatible with the DAE.
2. Assume we know the nodal values of $x^{(j)}(t)$ in a given mesh t_i, $i = 1, 2, \ldots, N$ distributed in $[0, T]$.
3. Compute the nodal values of the steepest descent direction $y^{(j)}$ given by the formula
$$y(t) = -\int_0^t \left[s\overline{F}(s)\overline{F}_\lambda(s) + \overline{F}(s)\overline{F}_\xi(s)\right]\, ds - t\int_t^T \overline{F}(s)\overline{F}_\lambda(s)\, ds,$$
$$y'(t) = -\overline{F}(t)\overline{F}_\xi(t) - \int_t^T \overline{F}(s)\overline{F}_\lambda(s)\, ds.$$
by making use only of the known values $x_i^{(j)} = x^{(j)}(t_i)$. This can be done through the use of quadrature formulae for integrals and finite differences for derivatives. Let $y_i^{(j)}$ be such optimal nodal values. Notice that in this formula we have not taken the projected steepest direction. As pointed out earlier, if the successive iterations (regardless of how they are obtained) drive the error to zero, then we are getting close to a solution.
4. Change $x^{(j)}$ to $x^{(j+1)}$ by using the update formula
$$x_i^{(j+1)} = x_i^{(j)} + \alpha_j y_i^{(j)}$$
for some small α_j so that the error decreases.
5. Iterate (3) and (4), until numerical convergence.

In the following three experiments, we have used adaptive Gauss quadrature with $N = 30$ (with 3 subintervals), see [97] for more details.

We have taken $\alpha_j = 0.1$, and used stop criteria $||x^{(j+1)} - x^{(j)}|| \leq 10^{-10}$. For a survey of nonlinear conjugate gradient methods, see [63].

- Index 1 [101]
$$\begin{aligned} y^{'}(t) &= z(t), \\ y(t)^2 + z(t)^2 &= 1, \\ y(0) = z(0) &= \frac{\sqrt{2}}{2}. \end{aligned}$$

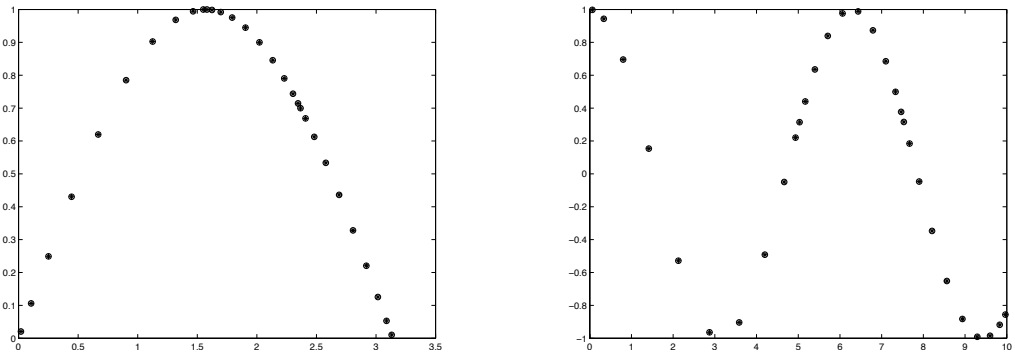

Figure 4.1: Index 1, $T = \pi$ left the y-coordinate, right the z-coordinate, 'o'-original, '+'-approximation

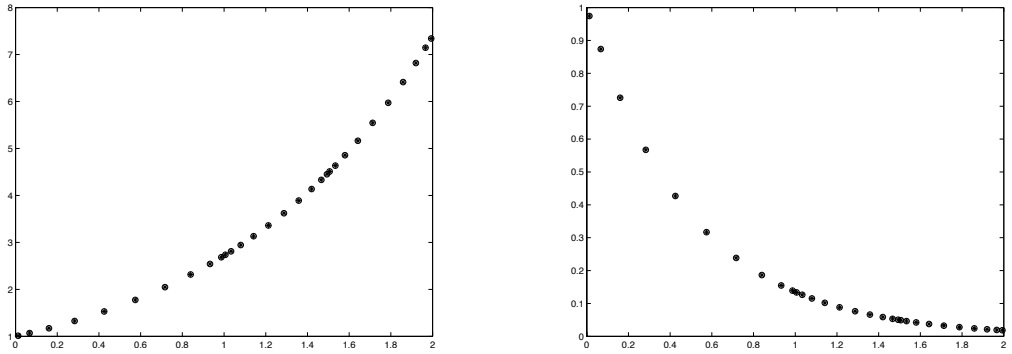

Figure 4.2: Index 2, $T = 2$, left the y_1-coordinate, right the y_2-coordinate, 'o'-original, '+'-approximation

- Index 2 [72]

$$\begin{aligned} y_1'(t) &= \sum_{i=1}^{5} f_i(y_1(t), y_2(t), z(t)), \\ y_2'(t) &= \sum_{i=1}^{5} g_i(y_1(t), y_2(t), z(t)), \\ y_1(t)^2 y_2(t) &= 1, \\ y_1(0) = y_2(0) &= 1, \\ z(0) &= 1, \end{aligned}$$

where

$$\begin{aligned}
f_1(y_1(t), y_2(t), z(t)) &= y_2(t) - 2y_1(t)^2 y_2(t) + y_1(t)y_2(t)^2 z(t)^2 + 2y_1(t)y_2(t)^2 - 2e^{-2t} y_1(t)y_2(t) \\
f_2(y_2(t), z(t)) &= -y_2(t)^2 z(t) + 2y_2(t)^2 z(t)^2, \\
g_1(y_1(t), y_2(t)) &= -y_1(t)^2 + y_1(t)^2 y_2(t)^2, \\
g_2(y_1(t), y_2(t), z(t)) &= -y_1(t) + e^{-t} z(t) - 3y_2(t)^2 z(t) + z(t).
\end{aligned}$$

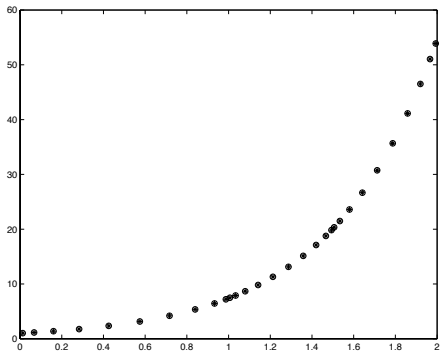

Figure 4.3: Index 2, $T = 2$, the z-coordinate, 'o'-original, '+'-approximation

- Index 3 [73]

$$\begin{aligned}
y_1'(t) &= 2y_1(t)y_2(t)z_1(t)z_2(t), \\
y_2'(t) &= -y_1(t)y_2(t)z_2(t)^2, \\
z_1'(t) &= (y_1(t)y_2(t) + z_1(t)z_2(t))u(t), \\
z_2'(t) &= -y_1(t)y_2(t)^2 z_2(t)^2 u(t), \\
y_1(t)y_2(t)^2 &= 1, \\
y_1(0) = y_2(0) &= 1, \\
z_1(0) = z_2(0) &= 1, \\
u(0) &= 1.
\end{aligned}$$

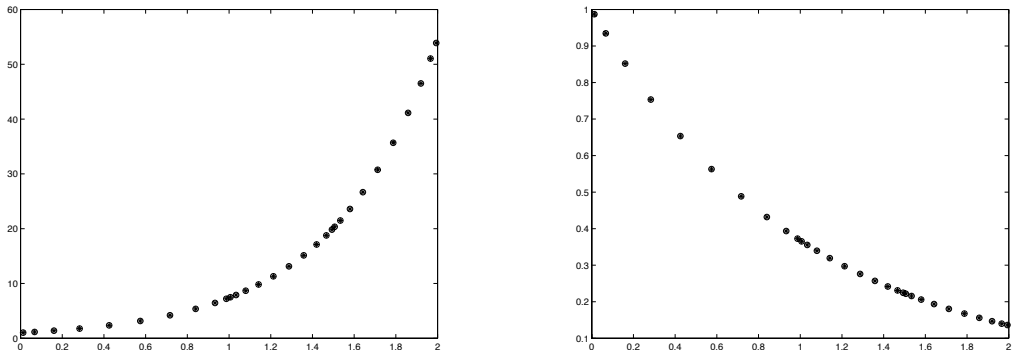

Figure 4.4: Index 3, $T = 2$, left the y_1-coordinate, right the y_2-coordinate, 'o'-original, '+'-approximation

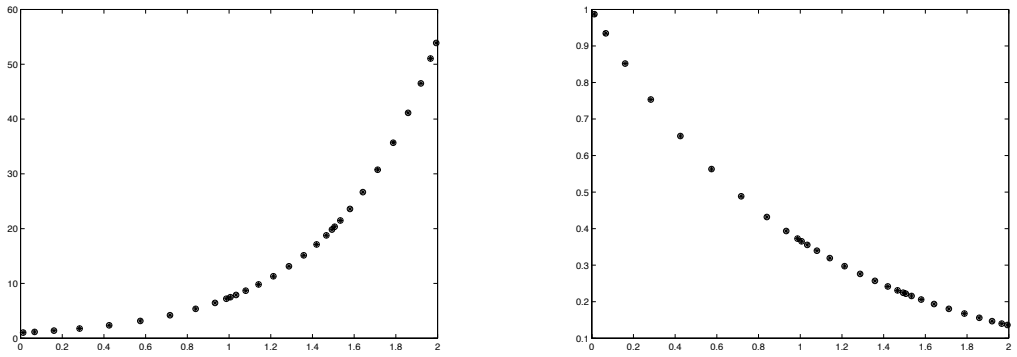

Figure 4.5: Index 3, $T = 2$, left the z_1-coordinate, right the z_2-coordinate, 'o'-original, '+'-approximation

In Figures 4.1, 4.2, 4.3, 4.4, 4.5, and 4.6, we compare the solution of the corresponding three problems with the approximations given by our approach. The results

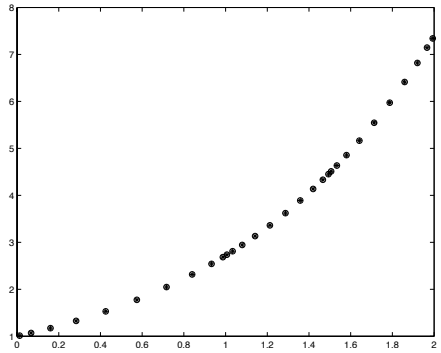

Figure 4.6: Index 3, $T = 2$, the u-coordinate, 'o'-original, +'-approximation

are very satisfactory in all cases, obtaining always the convergence to the true solution. In a first look, the exact and computed solutions are indistinguishable. A more systematic and careful analysis of the numerical possibilities of the method will be pursued in the future.

Chapter 5

A particular variational linearization of DAEs

Abstract 5.0.1 *This paper deals with the approximation of systems of differential-algebraic equations based on a certain error functional naturally associated with the system. In seeking to minimize the error, by using standard descent schemes, the procedure can never get stuck in local minima, but will always and steadily decrease the error until getting to the solution sought. Starting with an initial approximation to the solution, we improve it by adding the solution of some associated linear problems, in such a way that the error is significantly decreased. Some numerical examples are presented to illustrate the main theoretical conclusions. We should mention that we have already explored, in some previous papers [12, 13, 94], this point of view for regular problems. However, the main hypotheses in these papers ask for some requirements that essentially rule out the application to singular problems.*

5.1 Introduction

Differential-algebraic equations are becoming increasingly important in a lot of technical areas. They are currently the standard modeling concept in many applications such as circuit simulation, multibody dynamics, and chemical process engineering, see for instance [27, 28, 35, 66, 101] with no attempt to be exhaustive.

A basic concept in the analysis of differential-algebraic equations is the index. The notion of index is used in the theory of DAEs for measuring the distance from a DAE to its related ODE. The higher the index of a DAE, the more difficulties one may find in its numerical solution. There are different index definitions, but for simple problems they are identical. On more complicated nonlinear and fully implicit

systems they can be different (see [101] and his references).

For simplicity, we focus our attention on problems of the form

$$Mx'(t) = f(x(t)) \text{ in } (0,T), \quad x(0) = x_0, \tag{5.1.1}$$

where M is a given, eventually singular, matrix depending on t. More general situations can be allowed. This type of equations arises, for instance, in the functional analytic formulation of the initial value problem for the Stokes as well as for the linearized Navier-Stokes or Oseen equations [42].

For the approximation of these equations collocation-type methods are usually used. These methods are implicit, and we need to solve a nonlinear system of equations in each iteration using a Newton's type method. Given different coefficients c_i, $1 \leq i \leq s$, there is a (unique for h sufficiently small) polynomial of collocation $q(t)$ of degree less than or equal to s such that

$$q(t_0) = y_0, \quad q'(t_0 + c_i\, h) = f(t_0 + c_i\, h, q(t_0 + c_i\, h)) \quad \text{if } 1 \leq i \leq s. \tag{5.1.2}$$

The collocation methods are defined by an approximation $y(t) \simeq q(t)$, and are equivalent to implicit RK methods of s stages

$$\begin{aligned} k_i &= f(t_0 + c_i\, h, y_0 + h \sum_{j=1}^{s} a_{i,j}\, k_j), \\ y_1 &= y_0 + h \sum_{i=1}^{s} b_i\, k_i, \end{aligned} \tag{5.1.3}$$

for the coefficients

$$\begin{aligned} a_{i,j} &= \int_0^{c_i} \prod_{l \neq j} \frac{u - c_l}{c_j - c_l}\, du, \\ b_i &= \int_0^{1} \prod_{l \neq i} \frac{u - c_l}{c_i - c_l}\, du. \end{aligned} \tag{5.1.4}$$

The coefficients c_i play the role of the nodes of the quadrature formula, and the associated coefficients b_i are analogous to the weights. From (5.1.4), we can find implicit RK methods called Gauss of order $2s$, Radau IA and Radau IIA of order $2s-1$ and Lobatto IIIA of order $2s-2$. Also we can consider perturbed collocation methods like Lobato IIIC. See [66] for more details.

A number of convergence results have been derived for these methods introducing the so-called B-convergence theory. In [22, 23] the authors extend the B-convergence theory to be valid for a class of non-autonomous weakly nonlinear stiff systems, in

particular, including **the linear case**. As pointed out by the same authors, it is not clear if it is possible to cover, in a satisfactory way, highly nonlinear stiff problems, i.e., problems where also the nonlinear terms are affected by large parameters. Moreover, any result should assume that, in each step, the associated nonlinear system is well approximated [101]. In particular, we should be able to start with a good initial guess for the iterative scheme. This might be very restrictive for many stiff problems.

On the other hand, iterative methods are the typical tool to solve nonlinear systems of equations. In these schemes we compute a sequence of approximations solving associated linear problems. In this paper, we would like to introduce a new variational approach for the treatment of DAEs where we **linearize** the original equations obtaining an iterative scheme. Our ideas are based on the analysis of a certain error functional of the form

$$E(x) = \frac{1}{2}\int_0^T |Mx'(t) - f(x(t))|^2\, dt,$$

to be minimized among the absolutely continuous paths $x : (0,T) \to \mathbb{R}^N$ with $x(0) = x_0$. Note that if $E(x)$ is finite for one such path x, then automatically Mx' is square integrable. This error functional is associated, in a natural way, with the Cauchy problem (5.1.1). Indeed, the existence of solutions for (5.1.1) is equivalent to the existence of minimizers for E with vanishing minimum value. This is elementary.

We want to concentrate on the approximation issue through this perspective. We will place ourselves under the appropriate hypotheses so that there are indeed solutions for (5.1.1), i.e. there are minimizers for the error with vanishing minimum value. In addition, we would like to guarantee that the main ingredients for the iterative approximating scheme to work are valid. More explicitly, our approach for the numerical approximation of such problems relies on three main analytical hypotheses that we take for granted here:

1. The Cauchy problem (5.1.1) admits a unique solution for every feasible initial condition x_0 (the definition of feasible path should depend on the index of the equation).

2. The linearization around any feasible, absolutely continuous, path $x(t)$ with $x(0) = x_0$,

   $$My'(t) - \nabla f(x(t))y(t) = f(x(t)) - Mx'(t) \text{ in } (0,T), \quad y(0) = 0$$

   always has a unique solution, and moreover, for some constant $L > 0$ depending on M, f, x and its derivatives,

   $$\|y\|^2_{L^\infty(0,T)} \leq TL\|f(x(t)) - Mx'(t)\|_{L^2(0,T)}.$$

3. The only solution of the problem

$$\frac{d}{dt}(M^T z(t)) + \nabla f(x(t))^T z(t) = 0 \text{ in } (0,T), \quad M^T z(T) = 0,$$

is $z \equiv 0$, for every feasible, absolutely continuous, path $x(t)$ with $x(0) = x_0$.

Here the superscript ^{T} indicates transpose.

These requirements depend on the index of the equation and on some regularity on the pair $(M, \nabla f(x(t)))$. They should be more restrictive for equations with high index. More details can be found for example in [26] Theorem 3.9, where the authors consider DAEs transferable into standard canonical form. More precise information are outside of the scope of this paper. In any case, the equations verifying our hypotheses are, in general, a subclass of all analytically solvable systems.

In addition to the basic facts just stated on existence and uniqueness of solutions for our problems, the analysis of the approximation scheme, based on a minimization of the error functional E, requires one main basic assumption on the non-linearity $f : \mathbb{R}^N \to \mathbb{R}^N$. It must be smooth, so that $\nabla f : \mathbb{R}^N \to \mathbb{R}^{N\times N}$ is continuous, and globally Lipschitz with constant $K > 0$ ($|\nabla f| \leq K$). Moreover, the main result of this paper demands a further special property on the map f: for every positive $C > 0$ and small $\epsilon > 0$, there is $D_{C,\epsilon} > 0$ so that

$$|f(x+y) - f(x) - \nabla f(x)y| \leq D_{C,\epsilon}|y|^2, \quad |x| \leq C, |y| \leq \epsilon.$$

This regularity is somehow not surprising as our approach here is based on regularity and optimality. On the other hand, that regularity holds for most of the important problems in applications. It certainly does in all numerical tests performed in this work. Our goal here is placed on the fact that this optimization strategy may be utilized to set up approximation schemes based on the minimization of the error functional. Indeed, we provide a solid basis for this approximation procedure. One very important and appealing property of our approach states that typical minimization schemes like (steepest) descent methods will work fine as they can never get stuck in local minima, and converge steadily to the solution of the problem, no matter what the initialization is.

We should mention that we have already explored, in some previous papers, this point of view. Since the initial contribution [13], we have also treated the reverse mechanism of using first discretization, and then optimality ([12]). The perspective of going through optimality and then discretization has already been indicated and studied in [94], though only for the steepest descent method, and without going through any further analytical foundation for the numerical procedure. However,

the main hypotheses in these papers ask for some requirements that essentially rule out the application to singular problems. We will however address shortly ([14]) a complete treatment of DAEs with no a priori assumptions on existence and uniqueness. Rather, we will be interested in showing existence and uniqueness from scratch by examining the fundamental properties of the error functional E.

On the other hand, variational methods have been used also before in the context of ODEs. See ([85, 91]), where numerical integration algorithms for finite-dimensional mechanical systems that are based on discrete variational principles are proposed and studied. This is one approach to deriving and studying symplectic integrators. The starting point is Hamilton's principle and its direct discretization. In those references, some fundamental numerical methods are presented from that variational viewpoint where the model plays a prominent role.

The rest of the paper is divided in three sections. In Section 2 we introduce our variational approach, and present a convergence analysis. Section 3 introduces the numerical procedure. Finally, we present some numerical results in Section 4.

5.2 A main descent procedure

We start with a fundamental fact for our approach.

Proposition 5.2.1 *Let $\overline{x}$ be a critical point for the error E. Then $\overline{x}$ is the solution of the Cauchy problem (5.1.1).*

Proof

The proof is elementary. Based on the smoothness and bounds assumed on the mapping f, we conclude that if $x \equiv \overline{x}$ is a critical point for the error E, then x ought to be a solution of the problem

$$-\frac{d}{dt}\left(M^T(Mx^{'}(t) - f(x(t))\right) - \nabla f(x(t))^T\left((Mx^{'}(t) - f(x(t))\right) = 0 \text{ in } (0,T),$$
$$x(0) = x_0, M^T(Mx^{'}(T) - f(x(T))) = 0.$$

The vector-valued map $y(t) = Mx'(t) - f(x(t))$ is then a solution of the linear, non-degenerate problem

$$M^T y'(t) + \nabla f(x(t))^T y(t) = 0 \text{ in } (0,T), \quad M^T y(T) = 0.$$

The only solution of this problem, by our initial conditions on uniqueness of linearizations, is $y \equiv 0$, and so x is the solution of our Cauchy problem.

□

On the other hand, suppose we start with an initial crude approximation $x^{(0)}$ to the solution of our basic problem (5.1.1). We could take $x^{(0)} = x_0$ for all t, or $x^{(0)}(t) = x_0 + tf(x_0)$. We would like to improve this approximation in such a way that the error is significantly decreased. We have already pointed out that descent methods can never get stuck on anything but the solution of the problem, under global lipschitzianity hypotheses.

It is straightforward to find the Gâteaux derivative of E at a given feasible x in the direction y with $y(0) = 0$. Namely

$$E'(x)y = \int_0^T ((Mx'(t) - f(x(t))) \cdot (My'(t) - \nabla f(x(t))y(t)))\, dt.$$

This expression suggests a main possibility to select y from. Choose y such that

$$My'(t) - \nabla f(x(t))y(t) = f(x(t)) - Mx'(t) \text{ in } (0,T), \quad y(0) = 0.$$

In this way, it is clear that $E'(x)y = -2E(x)$, and so the (local) decrease of the error is of the size $E(x)$. Finding y requires solving the above linear problem which is assumed to have a unique solution by our main hypotheses in the introduction. In some sense, this is like a Newton method with global convergence.

Suppose $x^{(0)}$ is a feasible path in the interval $(0,T)$ so that $x^{(0)}(0) = x_0$, $M(x^{(0)})'$ is square integrable, $|x^0(t)| \leq C$ for a fixed constant C, and all $t \in (0,T)$ and the quantity

$$E(x^{(0)}) = \frac{1}{2}\int_0^T |M(x^{(0)})'(t) - f(x^{(0)}(t))|^2\, dt$$

measures how far such $x^{(0)}$ is from being a solution of our problem.

Theorem 5.2.2 *For T sufficiently small, the iterative procedure $x^{(j)} = x^{(j-1)} + y^{(j)}$, starting from the above feasible $x^{(0)}$, and defining $y^{(j)}$ as the solution of the linear problem*

$$M(y^{(j)})'(t) - \nabla f(x^{(j-1)}(t))y^{(j)}(t) = f(x^{(j-1)}(t)) - M(x^{(j-1)})'(t) \text{ in } (0,T), \quad y^{(j)}(0) = 0,$$

converges strongly in $L^\infty(0,T)$ to the solution of (5.1.1).

Proof

Choose $\epsilon > 0$ and $0 < \alpha < 1$ so that

$$\frac{\epsilon}{1 - \sqrt{\alpha}} \leq C,$$

and

$$|f(z+y)-f(z)-\nabla f(z)y| \le D|y|^2, \quad |y| \le \epsilon, |z| \le 2C,$$

for some constant $D > 0$ (see the main hypotheses in the introduction). We then solve for $y^{(0)}$ as the solution of the non-autonomous linear problem

$$My'(t) - \nabla f(x(t))y(t) = f(x(t)) - Mx'(t) \text{ in } (0,T), \quad y(0) = 0,$$

and pretend to update $x^{(0)}$ to $x^{(0)} + y^{(0)}$ in such a way that the error for $x^{(0)} + y^{(0)}$ be less than the error for the current iteration $x^{(0)}$. Note that

$$E(x^{(0)}+y^{(0)}) = \frac{1}{2}\int_0^T |f(x^{(0)}(t)+y^{(0)}(t))-f(x^{(0)}(t))-\nabla f(x^{(0)}(t))y^{(0)}(t)|^2\,dt, \quad (5.2.1)$$

where we have used the differential equation satisfied by $y^{(0)}$ and the definition of $E(x)$. By our assumption on f above,

$$|f(x^{(0)}(t)+y^{(0)}(t)) - f(x^{(0)}(t)) - \nabla f(x^{(0)}(t))y^{(0)}(t)| \le D|y^{(0)}(t)|^2, \quad t \in (0,T), \quad (5.2.2)$$

provided that $|y^{(0)}(t)| \le \epsilon$. Since we know that $y^{(0)}$ is the solution of a certain linear problem, by the upper bound assumed in the introduction on the size of these solutions,

$$|y^{(0)}(t)|^2 \le TLE(x^{(0)}) \text{ for all } t \in [0,T], \quad L \in \mathbb{R}^+. \quad (5.2.3)$$

Assume we select $T > 0$ so small that

$$E_{0,T}(x^{(0)}) \equiv E(x^{(0)}) \le \frac{\epsilon^2}{TL}, \quad (5.2.4)$$

and then $|y^{(0)}(t)| \le \epsilon$ for all $t \in [0,T]$. By (5.2.1), (5.2.2), and (5.2.3), we can write

$$E(x^{(0)}+y^{(0)}) \le \frac{D^2}{2}\int_0^T |y^{(0)}(t)|^4\,dt \le \frac{D^2}{2}L^2T^3E(x^0)^2. \quad (5.2.5)$$

If, in addition, we demand, by making T smaller if necessary,

$$E(x^{(0)}) \le \frac{2\alpha}{D^2T^3L^2}, \quad (5.2.6)$$

then $E(x^{(0)} + y^{(0)}) \le \alpha E(x^{(0)})$. Moreover, for all $t \in (0,T)$,

$$|x^{(0)}(t) + y^{(0)}(t)| \le C + \epsilon \le 2C.$$

All these calculations form the basis of a typical induction argument, verifying

$$\left|\sum_{i=0}^{j-1} x^{(i)}(t)\right| \leq C + \epsilon\left(\sum_{i=0}^{j-2}\sqrt{\alpha}^{i}\right) (\leq 2C),$$
$$|x^{(j-1)}(t)| \leq \epsilon\sqrt{\alpha}^{j-2} \text{ for all } t \in [0,T],$$
$$E\left(\sum_{i=0}^{j-1} x^{(i)}\right) \leq \alpha^{j-1}E(x^{(0)}) (\leq E(x^{(0)})).$$

It is therefore clear that the sum

$$\sum_{i=0}^{\infty} x^{(i)}(t)$$

converges strongly in $L^{\infty}(0,T)$ to the solution of our initial Cauchy problem in a small interval $(0,T)$.

□

Since the various ingredients of the problem do not depend on T, we can proceed to have a global approximation in a big interval by successively performing this analysis in intervals of appropriate small size. For instance, we can always divide a global interval $(0,T)$ into a certain number n of subintervals of small length h $(T = nh)$ with

$$\frac{E_{0,T}(x^{(0)})D^2L^2}{2\alpha} \leq \frac{1}{h^3},$$

according to (5.2.6).

5.3 Numerical procedure

Since our optimization approach is really constructive, iterative numerical procedures are easily implementable.

1. Start with an initial approximation $x^{(0)}(t)$ compatible with the initial conditions (ex. $x^{(0)}(t) = x_0 + tf(x_0)$).

2. Assume we know the approximation $x^{(j)}(t)$ in $[0,T]$.

3. Compute its derivative $M(x^{(j)})'(t)$.

4. Compute the auxiliar function $y^{(j)}(t)$ as the numerical solution of the problem

$$My'(t) - \nabla f(x^{(j)}(t))y(t) = f(x^{(j)}(t)) - M(x^{(j)})'(t) \text{ in } (0,T), \quad y(0) = 0,$$

by making use of a numerical scheme for DAEs with dense output (like collocation methods, see above subsection).

5. Change $x^{(j)}$ to $x^{(j+1)}$ by using the update formula

$$x^{(j+1)}(t) = x^{(j)}(t) + y^{(j)}(t).$$

6. Iterate (3), (4) and (5), until numerical convergence.

In practice, we use the stopping criterium

$$\max\{||y^{(j)}||_\infty, \sqrt{2E(x^{(j)})}\} \leq TOL.$$

In particular, one can implement, in a very easy way, this numerical procedure using a problem-solving environment like MATLAB [119].

5.4 Some experiments

In this section, we approximate some problems well known in the literature for different index [72, 73, 101]. High-order accuracy and stability are major areas of interest in this type of simulations. We don't perform an analysis of the convergence conditions imposed in the above section. We only are interested to test numerically our approach.

In our approach we only need to approximate, with at least order one, the associated linear system for $y^{(j)}$, in order to obtain the convergence of our scheme (see Theorem 7.3.2). The stability can be ensured by the fact that we approximate a **linear** problem using specific implicit methods [66]. This is not the case with a general non-linear problem [78], where we need to approximate well (with a Newton-type iterative method) the non-linear system related to the implicitness of the scheme. This approximation should be a difficult task due to the local (non-global) convergence of any iterative scheme for non-linear problems.

In this section, we consider the convergent Lobatto IIIC method [73] valid for index 1-3, in order to approximate the associated linear problem for $y^{(j)}$ in each iteration. This method can be considered as a perturbation collocation method.

The final error depends only on the stopping criterium. In the following examples, we stop the algorithm when

$$\max\{||y^{(j)}||_\infty, \sqrt{2E(x^{(j)})}\} \leq 10^{-6},$$

and plot the solution and the approximation given by our approach.

- Index 1 [101]

$$\begin{aligned} y^{'}(t) &= z(t), \\ y(t)^2 + z(t)^2 &= 1, \\ y(0) = z(0) &= \frac{\sqrt{2}}{2}. \end{aligned}$$

The solution of this problem is $(\sin(x + \pi/4), \cos(x + \pi/4))$.

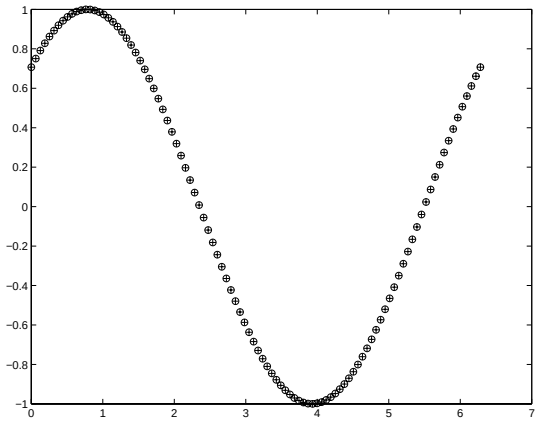

Figure 5.1: Index 1, $T = 2\pi$, the y-coordinate, 'o'-original, '+'-approximation

- Index 2 [72]

$$\begin{aligned} y_1^{'}(t) &= \sum_{i=1}^{5} f_i(y_1(t), y_2(t), z(t)), \\ y_2^{'}(t) &= \sum_{i=1}^{5} g_i(y_1(t), y_2(t), z(t)), \\ y_1(t)^2 y_2(t) &= 1, \\ y_1(0) = y_2(0) &= 1, \\ z(0) &= 1, \end{aligned}$$

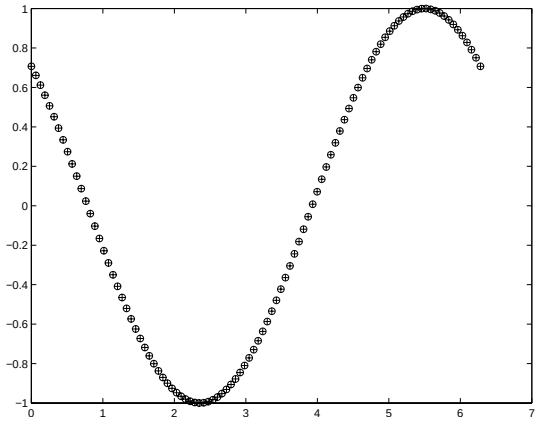

Figure 5.2: Index 1, $T = 2\pi$, the z-coordinate, 'o'-original, '+'-approximation

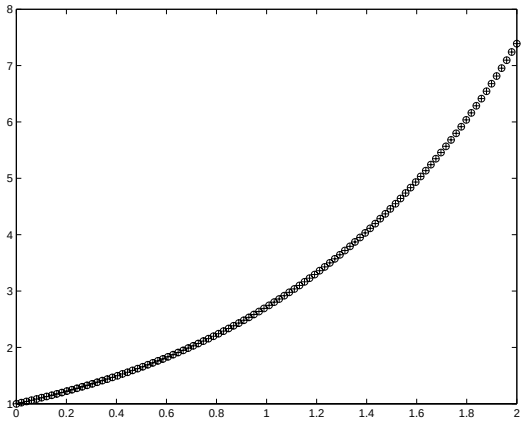

Figure 5.3: Index 2, $T = 2$, the y_1-coordinate, 'o'-original, '+'-approximation

where

$$\begin{aligned}
f_1(y_1(t), y_2(t), z(t)) &= y_2(t) - 2y_1(t)^2 y_2(t) + y_1(t)y_2(t)^2 z(t)^2 + 2y_1(t)y_2(t)^2 - 2e^{-2t} y_1(t)y_2(t), \\
f_2(y_2(t), z(t)) &= -y_2(t)^2 z(t) + 2y_2(t)^2 z(t)^2, \\
g_1(y_1(t), y_2(t)) &= -y_1(t)^2 + y_1(t)^2 y_2(t)^2, \\
g_2(y_1(t), y_2(t), z(t)) &= -y_1(t) + e^{-t} z(t) - 3y_2(t)^2 z(t) + z(t).
\end{aligned}$$

The solution of this problem is (e^t, e^{-2t}, e^{2t}).

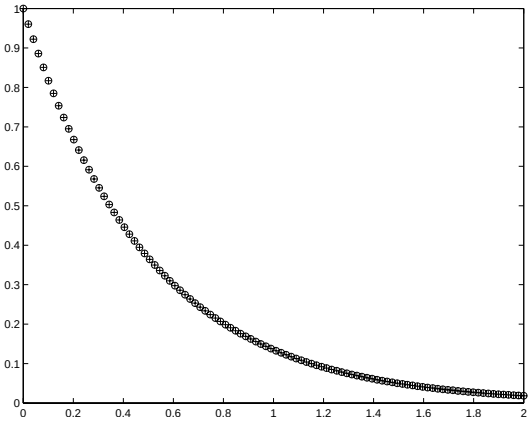

Figure 5.4: Index 2, $T = 2$, the y_2-coordinate, 'o'-original, '+'-approximation

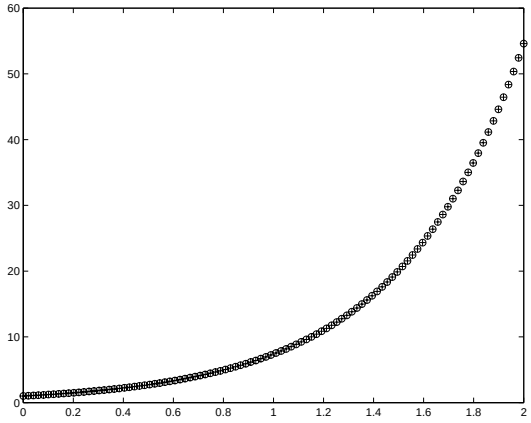

Figure 5.5: Index 2, $T = 2$, the z-coordinate, 'o'-original, '+'-approximation

- Index 3 [73]

$$\begin{aligned}
y_1'(t) &= 2y_1(t)y_2(t)z_1(t)z_2(t),\\
y_2'(t) &= -y_1(t)y_2(t)z_2(t)^2,\\
z_1'(t) &= (y_1(t)y_2(t) + z_1(t)z_2(t))u(t),\\
z_2'(t) &= -y_1(t)y_2(t)^2z_2(t)^2u(t),\\
y_1(t)y_2(t)^2 &= 1,\\
y_1(0) = y_2(0) &= 1,\\
z_1(0) = z_2(0) &= 1,\\
u(0) &= 1.
\end{aligned}$$

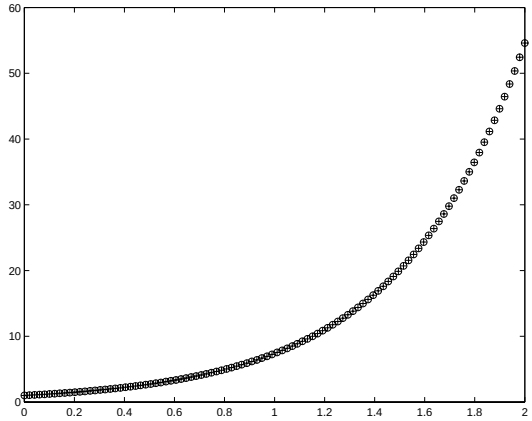

Figure 5.6: Index 3, $T = 2$, the y_1-coordinate, 'o'-original, '+'-approximation

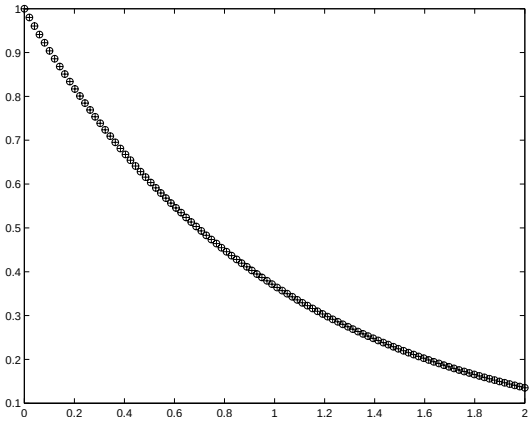

Figure 5.7: Index 3, $T = 2$, the y_2-coordinate, 'o'-original, '+'-approximation

The solution of this problem is $(e^{2t}, e^{-t}, e^{2t}, e^{-t}, e^{t})$.

In Figures 5.1-5.10, we compare the solution of the corresponding three problems with the approximations given by our approach. The results are very satisfactory in all cases, obtaining always the convergence to the true solution. In a first look, the exact and computed solutions are indistinguishable, since after convergence the error is smaller than the tolerance ($= 10^{-6}$) used in the stopping criterium. A more systematic and careful analysis of the numerical possibilities of the method will be pursued in the future.

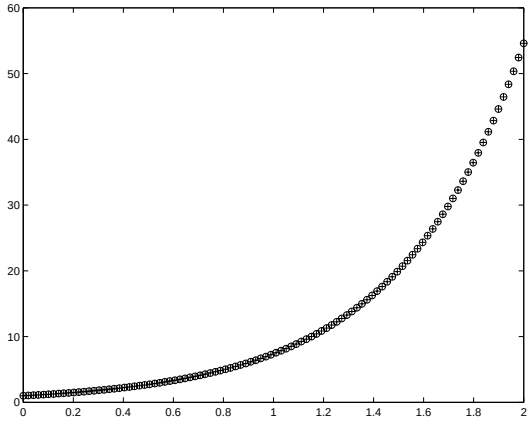

Figure 5.8: Index 3, $T = 2$, the z_1-coordinate, 'o'-original, '+'-approximation

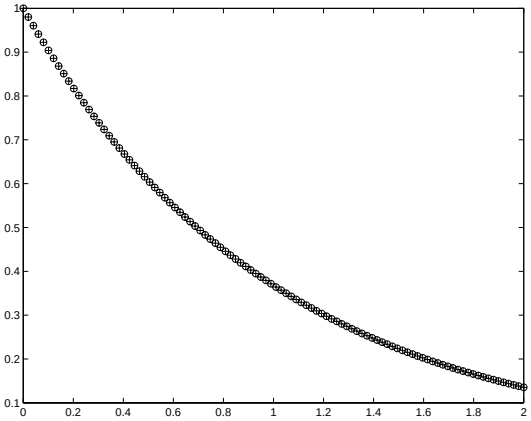

Figure 5.9: Index 3, $T = 2$, the z_2-coordinate, 'o'-original, '+'-approximation

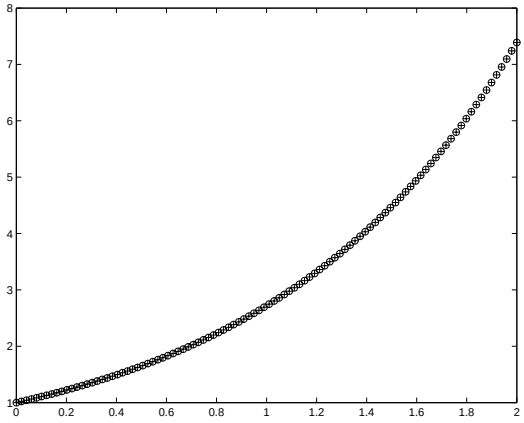

Figure 5.10: Index 3, $T = 2$, the u-coordinate, 'o'-original, +'-approximation

Chapter 6

Current work

Abstract 6.0.1 *This chapter deals with some ideas of our current work.*

6.1 A step variable implementation

An implicit fixed-step solver computes the state at the next time step as an implicit function of the state at the current time step and the state derivative at the next time step. The variable-step solvers dynamically vary the step size during the simulation. These solvers increase or reduce the step size using its local error control to achieve the tolerances that you specify. Computing the step size at each time step adds to the computational overhead but can reduce the total number of steps, and the simulation time required to maintain a specified level of accuracy.

For a stiff problem, solutions can change on a time scale that is very small as compared to the interval of integration, while the solution of interest changes on a much longer time scale. Methods that are not designed for stiff problems are ineffective on intervals where the solution changes slowly because these methods use time steps small enough to resolve the fastest possible change.

If we denote by $y(t_n, t_{n-1}, y_{n-1})$ the solution of a given differential equation $F(t, y, y^{'}) = 0$, then a method of order p will verify that locally its error has the form

$$||e_n|| = ||y(t_n, t_{n-1}, y_{n-1}) - y_n|| = C\ h^{p+1} + O(h^{p+2}).$$

In order to estimate C we can use extrapolations techniques like the presented in the following section. Other possibility is to consider other scheme $\tilde{y}_n$ with bigger order q starting from y_{n-1}. In this case we have

$$||\tilde{e}_n|| = ||y(t_n, t_{n-1}, y_{n-1}) - \tilde{y}_n|| = \tilde{C}\ h^{q+1} + O(h^{q+2}),$$

then

$$||e_n|| = ||y_n - \tilde{y}_n|| + O(h^{p+2}).$$

With an estimation of the error the variable step codes select the new step such that the error is smaller than a prescribed tolerance TOL.

The desired error associated to a h_* will be

$$||e_n(h')|| = Ch_*^{p+1} = TOL$$

and the real error

$$||e_n(h)|| = Ch^{p+1} = ||y_n - \tilde{y}_n||,$$

dividing both equations we obtain

$$h_* = h \sqrt[p+1]{\frac{TOL}{||y_n - \tilde{y}_n||}}.$$

6.1.1 Extrapolation techniques

When we are approximating the exact solution a_0 of a given differential problem by means of a Runge-Kutta method, the error can be expressed as a Taylor series:

$$S(h) = a_0 + a_1 h^{p_1} + a_2 h^{p_2} + \cdots . \tag{6.1.1}$$

In order to improve the accuracy we can use extrapolation procedures. Richardson's (or polynomial) extrapolation consists on successive elimination of the terms $a_i h^{p_i}$ by linear combinations of approximations $S(h)$ for different h. It can be viewed as the value at $h = 0$ of the only polynomial $P(x)$ interpolating the data $S(h)$ for the considered h's.

In practice, we can use the following algorithm for the implementation of Richardson extrapolation:

Algorithm 6.1.1 *Richardson Extrapolation*

For $l = 1, 2, 3, \ldots, n$

$$b_{(0,l)} = S(2^{l-1}h)$$

For $j = 1$ *up to* $r = n - 1$ *and* **for** $l = 1$ *up to* $k = r - j + 1$

$$b_{(j,l)} = \frac{2^p b_{(j-1,l)} - b_{(j-1,l+1)}}{2^p - 1}$$

where p is the first power of the error in each step.

It is well known that Richardson's extrapolation is equivalent to extrapolate by couples with functions of type $ax^p + b$ (p =order of the method in each step). The reciprocal polynomial extrapolation was introduced in [2] as an alternative of Richardson's extrapolation. It is based on the extrapolator function $R(x) = \frac{1}{P(x)}$ which is equivalent, as before, to considering extrapolations by couples with functions of the type $\frac{1}{dx^p+e}$.

The order of accuracy is the same in both cases [2].

A new implementation of the Reciprocal Polynomial Extrapolation

In this section, we introduce a new step in the implementation of the reciprocal polynomial extrapolation. We are interested in obtaining at least the same performance as that achieved by the Richardson extrapolation when this extrapolation works well and improving its robustness in other cases.

In the original reciprocal polynomial extrapolation technique [2], first we compute the inverse of the data, then we compute the Richardson extrapolation and finally we compute the inverse of the result.

One improvement proposed in the present paper consists on a specific translation of the original data. As in the case of the Richardson extrapolation scheme, we build the reciprocal polynomial extrapolation scheme by pairs; thus we start with two known values of S at different resolutions:

$$\begin{aligned} S(h) &= a_0 + a_1 h^{\gamma_1} + a_2 h^{\gamma_2} + \cdots \\ S(2h) &= a_0 + a_1 2^{\gamma_1} h^{\gamma_1} + a_2 2^{\gamma_2} h^{\gamma_2} + \cdots \end{aligned}$$

Let us consider the translation

$$T_h = sign(M)(1 + |m|),$$

where

$$\begin{cases} M = S(h),\ m = S(2h), & \text{if } |S(2h)| \leq |S(h)|, \\ M = S(2h),\ m = S(h), & \text{otherwise.} \end{cases}$$

The proposed translation T_h has the same sign as the maximum (in absolute value) of the data. The size of T_h is equal to one plus the absolute value of the minimum (in absolute value) of the data.

Taking the above translation T_h into account, we compute

$$\begin{aligned} S_1 &= S(h) + T_h, \\ S_2 &= S(2h) + T_h. \end{aligned}$$

We now consider the rational function $r(x;h) = \dfrac{1}{dx^{\gamma_1} + c}$ that is able to interpolate S_1 and S_2,

$$\begin{aligned} \frac{1}{dh^{\gamma_1} + c} &= S_1, \\ \frac{1}{d2^{\gamma_1}h^{\gamma_1} + c} &= S_2. \end{aligned}$$

This system is equivalent to

$$\begin{aligned} \frac{1}{S_1} &= dh^{\gamma_1} + c, \\ \frac{1}{S_2} &= d2^{\gamma_1}h^{\gamma_1} + c; \end{aligned}$$

that is, the linear system of the Richardson extrapolation scheme for the new data $\frac{1}{S_1}$ and $\frac{1}{S_2}$.

Therefore, using the formula of the Richardson extrapolation,

$$\frac{1}{r(0;h)} = \frac{2^{\gamma_1}\frac{1}{S_1} - \frac{1}{S_2}}{2^{\gamma_1} - 1};$$

and the approximation obtained by the new reciprocal polynomial extrapolation scheme is then given by

$$r(0;h) - T_h = \frac{(2^{\gamma_1} - 1)S_1S_2}{2^{\gamma_1}S_2 - S_1} - T_h.$$

With this new step (translation procedure) any zero division in the inversion of each pair is avoided ($S_i \neq 0$, $i = 1, 2$). Moreover, since $|S_i| > 1$ and therefore $\frac{1}{|S_i|} \in (0, 1)$, $i = 1, 2$, the data interval will be changed from $\mathbb{R} = (-\infty, +\infty)$ in the direct implementation of the Richardson extrapolation to $(0, 1)$ (or $(-1, 0)$) when we use the Richardson extrapolation within the reciprocal polynomial extrapolation mechanism.

The Richardson extrapolation has problems for improving a given approximation when the step discretizations are not small enough, this depends of the characteristics of the problem (stiffness or perturbation parameters). In these cases the points $S_i \neq 0$, $i = 1, 2$ should be not close enough. Changing the data interval through the proposed translation ensures that the extrapolated points when using the Richardson extrapolation procedure within the reciprocal polynomial extrapolation scheme are close. In the numerical experiments of the annexe II we analyze the improvements obtained by this fact.

6.1.2 Our particular situation

In our case, we can use the size of the direction $y^{(j)}$. These numbers are related with the local error. At this moment we are testing this approach to several interesting problems that we include in the annexe I. We would like also to perform a free MATLAB code for the scientific community.

6.1.3 Practical Implementation

The estimation of the error and the new h_* has been introduced from a mathematical point of view. However, in order to be effective in practice we need some control strategies.

Let $h_{n+1} = t_{n+1} - t_n$ be the discretization parameters. We introduce the following equation

$$h_{n+1} = \sigma h_n \sqrt[p+1]{\frac{TOL}{||y_n - \tilde{y}_n||}},$$

where σ is a security factor smaller than 1.

If in a step $||y_n - \tilde{y}_n|| > TOL$ we reject y_n and compute a new iteration with

$$h_n = \sigma h_n \sqrt[p+1]{\frac{TOL}{||y_n - \tilde{y}_n||}}.$$

Finally, it is important to add some computational restrictions. Namely

$$h_{\min} \leq h_n \leq h_{\max}$$

and

$$\omega \leq \frac{h_{n+1}}{h_n} \leq \Omega$$

for some given positive constants $h_{\min}$, $h_{\max}$, ω and Ω.

Some classical examples are:

$$\begin{aligned} h_{\min} &= 10^{-6}, \\ h_{\max} &= 1, \\ \omega &= \frac{1}{5}, \\ \Omega &= 5. \end{aligned}$$

6.2 Approximation of Hamiltonian systems

We introduce a new variational approach for models which are formulated naturally as conservatives systems of ODEs, most importantly Hamiltonian systems. As a general rule, Hamiltonian systems are related to numerous areas of mathematics and have a lot of application branches, such as classical and quantum mechanics, statistics, optical, astronomy, molecular dynamic, plasma physics, etc.

6.2.1 A short view of the state of the art

Regarding approximating approaches for Hamiltonian systems, it is well known that numerical methods such as the ordinary Runge-Kutta methods are not valid for integrating Hamiltonian systems, because Hamiltonian systems are not generic in the set of all dynamic systems. They are not structurally stable against non-Hamiltonian perturbations. Numerical solution of Hamiltonian systems is frequently carried out by symplectic integrator due to their good performance in moderate and long-time integration, see [65, 71, 84, 92, 106]. Symplectic numerical methods belong to the family of Geometric Numerical Integrators methods, which preserve important qualitative and geometric properties of the underlying differential system, and are arguably the most popular methods in this class. Certain qualitative properties of the evolution, like symplecticity, are preserved and, in general they exhibit smaller error growth along the numerical trajectory.

Some pioneering works on symplectic integrations is due to Vogelaere [122], Ruth [104], and Feng Kang [50]. The derivation of higher-order methods is covered by several approaches such as composition methods, classical Runge-Kutta methods (RK) as well as partitioned Runge-Kutta (PRK) methods, and methods based on generating functions.

The systematic study of symplectic Runge-Kutta (RK) methods started around 1988, and a complete characterization has been found independently by Lasagni [80](using the approach of generating functions), and by Sanz-Serna [105] and Suris[117] (using the ideas of the classical papers of Burrage and Butcher [33] and Crouzeix [36] on algebraic stability). Nowadays, it is well-known that certain implicit RK methods of Radau type (generalizing the implicit Euler method) are useful in the context of systems with strong dissipation, like electronic circuits or chemical reaction dynamics.

Partitioned Runge- Kutta (PRK) methods are another approach to approximating the solution trajectory which it is based on using different approximation formulas for different components of the solution. PRK methods use different sets of quadra-

ture rules for each subset of the variables. High-order symplectic PRK methods are implicit when applied to general Hamiltonian systems and can be solved by a fixed-point iteration similar to implicit RK methods, or by Newton iteration. The situation changes for systems with a separable Hamiltonian, while symplectic RK methods are still necessarily implicit, explicit PRK methods can be found. However this class of explicit PRK methods is equivalent to the class of composition methods [93, 106].

The starting point of generating function (GF) theory was the discovery of Hamilton that the motion of the system is completely described by a *characteristic* function S, and that S is the solution of a partial differential equation, now called the Hamilton-Jacobi differential equation. It was notice later, especially by Siegel (Siegel and Moser 1971), that such a function S is directly connected to any symplectic map. It was called generating function. See [65, 84].

Another important point should be taken into account regarding Hamiltonian systems, even with symplectic maps, and that is the lack of energy conservation in the map. It would seem to be an obvious goal for Hamiltonian integration methods both to preserve the symplectic structure and to conserve the energy, but it was shown that this was in general impossible. Thus a symplectic map which only approximates a Hamiltonian cannot conserve energy [125].

Recently, some research has been carried out about energy-preserving symplectic methods based on the key tool line integral associated with conservative vector fields, as well as its discrete version, the so called discrete line integral. Interestingly, the line integral provides a means to check the energy conservation property. See,[30, 31].

6.2.2 A new variational approach

The new variational method for Hamiltonian systems, which is proposed here, is both symplectic and energy preserving.

We consider Hamiltonian system of the form

$$\begin{aligned} x' &= \Omega \nabla H(x), \\ x(0) &= x_0, \end{aligned}$$

where

$$\Omega = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}. \tag{6.2.1}$$

This system can be studied under a variational approach based on analysis of certain error functional. We seek to minimize the error by using standard descent

schemes. From this approach a process, in which the error always and steadily decrease until getting to the original solution, is obtained. The error functional which is associated, in a natural way, with a Hamiltonian system

$$E(x) = \int_0^T \frac{1}{2}|x'(t) - \Omega\nabla H(x(t))|^2 + \frac{1}{2}|H(x(t)) - H(x(0))|^2\, dt,$$

The proposed functional has the characteristic of energy persevering. The steepest descent direction can be found as the solution of a variational problem of the form

Minimize y:

$$\frac{1}{2}\int_0^T |y'(t)|^2 + (x'(t) - \Omega\nabla H(x(t))(y'(t) + \Omega^T\nabla^2 H(x(t))y(t)) + (H(x(t)) - H(x(0)))\nabla H(x(t))y(t)\, dt,$$

under $y(0) = 0$.

The optimal solution is given by

$$\begin{aligned}
&-\frac{d}{dt}\left[y'(t) + x'(t) + \Omega^T\nabla H(x(t))\right] \\
&+\nabla^2 H(x(t))\Omega\left(x'(t) + \Omega^T\nabla H(x(t))\right) + \left(H(x(t)) - H(x_0)\right)\nabla H(x(t)) = 0 \text{ in } (0,T), \\
&y(0) = 0, y'(T) + x'(T) + \Omega^T\nabla H(x(T)) = 0.
\end{aligned}$$

The solution of the variational problem can be given in an explicit form as

$$y(t) = -\int_0^t \left[sG(s) + F(s)\right]\, ds - t\int_t^T G(s)\, ds, \tag{6.2.2}$$

where

$$\begin{aligned}
F(t) &= x'(t) + \Omega^T\nabla H(x(t)) \\
G(t) &= \nabla^2 H(x(t))\Omega(x'(t) + \Omega^T\nabla H(x(t))) + (H(x(t)) - H(x(0)))\nabla H(x(t))
\end{aligned}$$

Everything can be written as an iterative numerical process.

6.2.3 Numerical procedure

The iterative numerical procedure is easily implementable.

1. Start with an initial approximation $x^0(t)$ compatible with the initial conditions, for instance $x^{(0)}(t) = x_0 + t\Omega\nabla H(x_0)$.

2. Assume we know the approximation $(x^{(j)})(t)$ in $[0,T]$.

3. Compute its derivative $(x^{(j)})'(t)$.
4. Compute F and G functions:

$$F^{(j)}(t) = (x^{(j)})'(t) + \Omega^T \nabla H(x^{(j)}(t)),$$

$$G^{(j)}(t) = \nabla^2 H(x^{(j)}(t))\Omega((x^{(j)})'(t)+\Omega^T \nabla H(x^{(j)}(t)))+(H(x^{(j)}(t))-H(x(0)))\nabla H(x^{(j)}(t)).$$

5. Compute the function

$$y^{(j)}(t) = -\int_0^t \left[sG^{(j)}(s) + F^{(j)}(s)\right]\, ds - t\int_t^T G^{(j)}(s)\, ds, \tag{6.2.3}$$

using quadrature formulas.
6. Change $x^{(j)}$ to $x^{(j+1)}$ by using the update formula

$$x^{(j+1)}(t) = x^{(j)}(t) + y^{(j)}(t).$$

7. Iterate (3), (4), (5) and (6) until numerical convergence.

6.2.4 Approximation using the non symplectic trapezoidal rule

We start with three classical problems in order to put out the necessity of use symplectic rules for the approximation of $y^{(j)}$. The first example is a typical system; the other two are a Lotka-Volterra problem and the Kepler problem. It is well known that the Lotka-Volterra problem is defined as two species problem: one, a predator, the other one, its prey. It is frequently use to describe the dynamics of biological systems, in which two species interact. The Kepler problem, in classical mechanic is about a special case of the two-body problem, in which the two bodies interact by a central force that varies in strength as the inverse square of the distance between them.

We are interested in approximating the value of the solution of the problem 1

$$\begin{aligned} p' &= q, \\ q' &= p + p^2. \end{aligned} \tag{6.2.4}$$

We have considered the initial conditions $p = 0$ and $q = 1$.
The solution of this problem is

$$q = \pm\sqrt{2(\frac{1}{2} + \frac{p^2}{2} + \frac{p^3}{3})}.$$

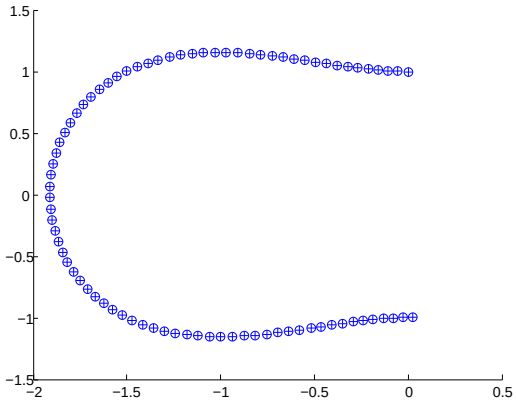

Figure 6.1: The x-coordinate versus y-coordinate,'o'-original,'+'-approximation. Problem 6.2.4

The Lotka-Volterra problem can be written as

$$\begin{aligned} p' &= e^q - 2, \\ q' &= 1 - e^p. \end{aligned} \tag{6.2.5}$$

The initial conditions considered are $p = 2.3$ and $q = 0.7$.
The Kepler problem can be found as

$$\begin{aligned} p_i' &= -\frac{q_1}{(q_1^2 + q_2^2)^{\frac{2}{3}}}, \\ q_i' &= p_i. \end{aligned} \tag{6.2.6}$$

for $i = 1, 2$.
The initial conditions have been $p_1 = 0.4$, $p_2 = 0$, $q_1 = 0$ and $q_2 = 2$.

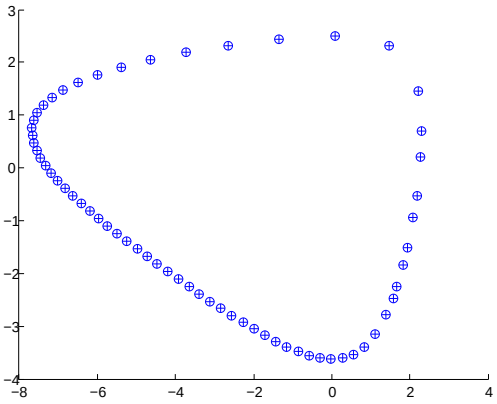

Figure 6.2: The x-coordinate versus y-coordinate,'o'-original,'+'-approximation. Problem 6.2.5

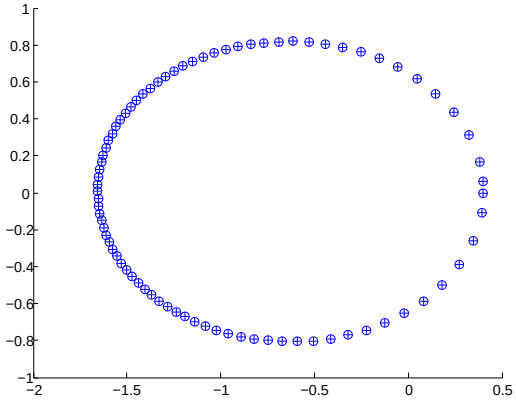

Figure 6.3: The x-coordinate versus y-coordinate,'o'-original,'+'-approximation. Problem 6.2.6

6.2.5 Approximation using symplectic rules

In this final section, we are going to explore specifically the steepest descent strategy with the Lorentz system $x' = Ax + F(x)$ with $x \in \mathbb{R}^3$ and

$$A = \begin{pmatrix} -10 & 10 & 0 \\ 28 & -1 & 0 \\ 0 & 0 & -8/3 \end{pmatrix}, \quad F(x) = (0, -x_1x_3, x_1x_2), \quad f(x) = Ax + F(x),$$

and for the Kepler model for elliptical orbits under Newton gravitational law $x' = F(x)$ where this time $x \in \mathbb{R}^4$, and

$$F(x) = \left(x_2, -\mu x_1/(x_1^2 + x_3^2)^{3/2}, x_4, -\mu x_3/(x_1^2 + x_3^2)^{3/2}\right).$$

These two particular models were also tested in [94], though the practical implementation was a bit different, and no particular attention was paid to the propagation of error with time. Initial conditions have been taken to be $(-10, 10, 25)$, and $(0.4, 0., 0., 2.)$, respectively.

The results have been obtained by using mid-point quadrature rule. The parameter h is the small time interval where the updated scheme $x_j + y_j$ for y_j the steepest descent direction at x_j converges. The simulations are then carried out by successive steps of length h. It especially strikes the result for the Kepler system where one can hardly distinguish the various turns around the elliptic orbit ([65]).

It is somewhat remarkable that our numerical tests for both problems are virtually exact to the degree of accuracy used, so numerical error do not propagate, or do so in such a small rate that errors do not spoil the approximation as time proceeds.

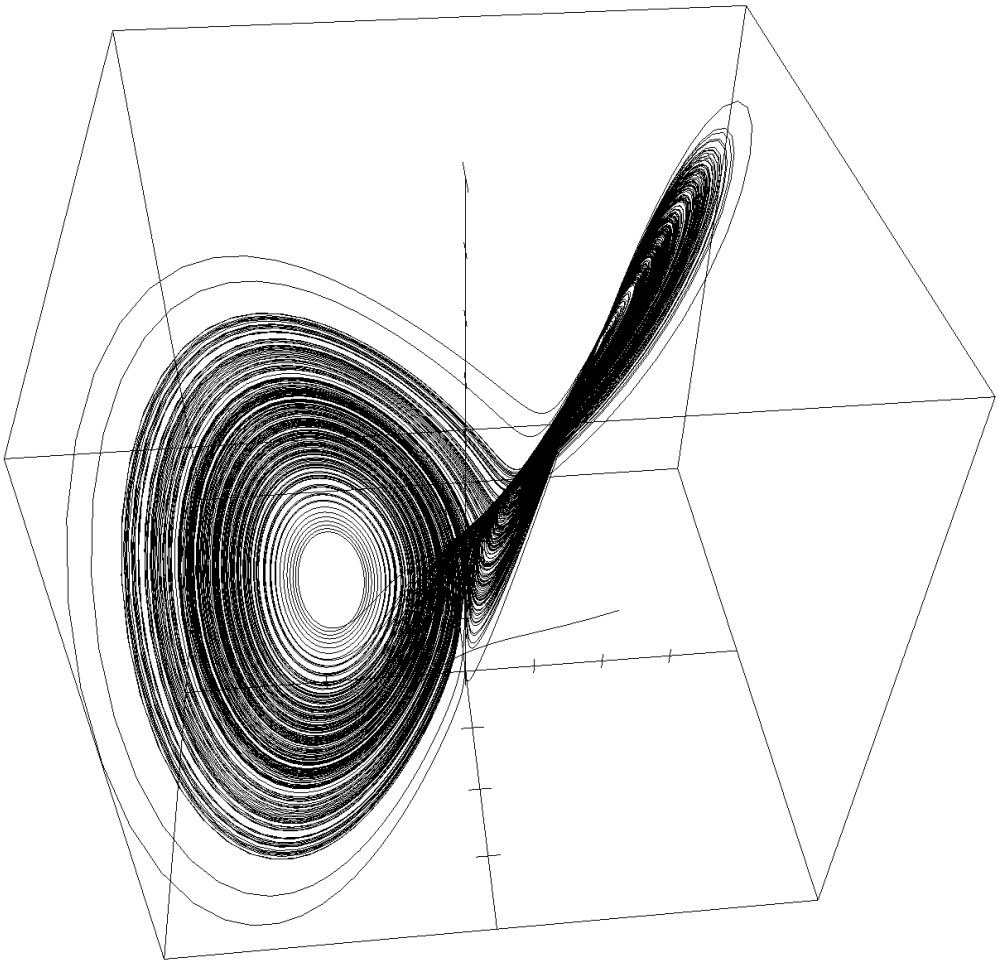

Figure 6.4: The Lorentz system for $h = 0.01$ and 30000 steps.

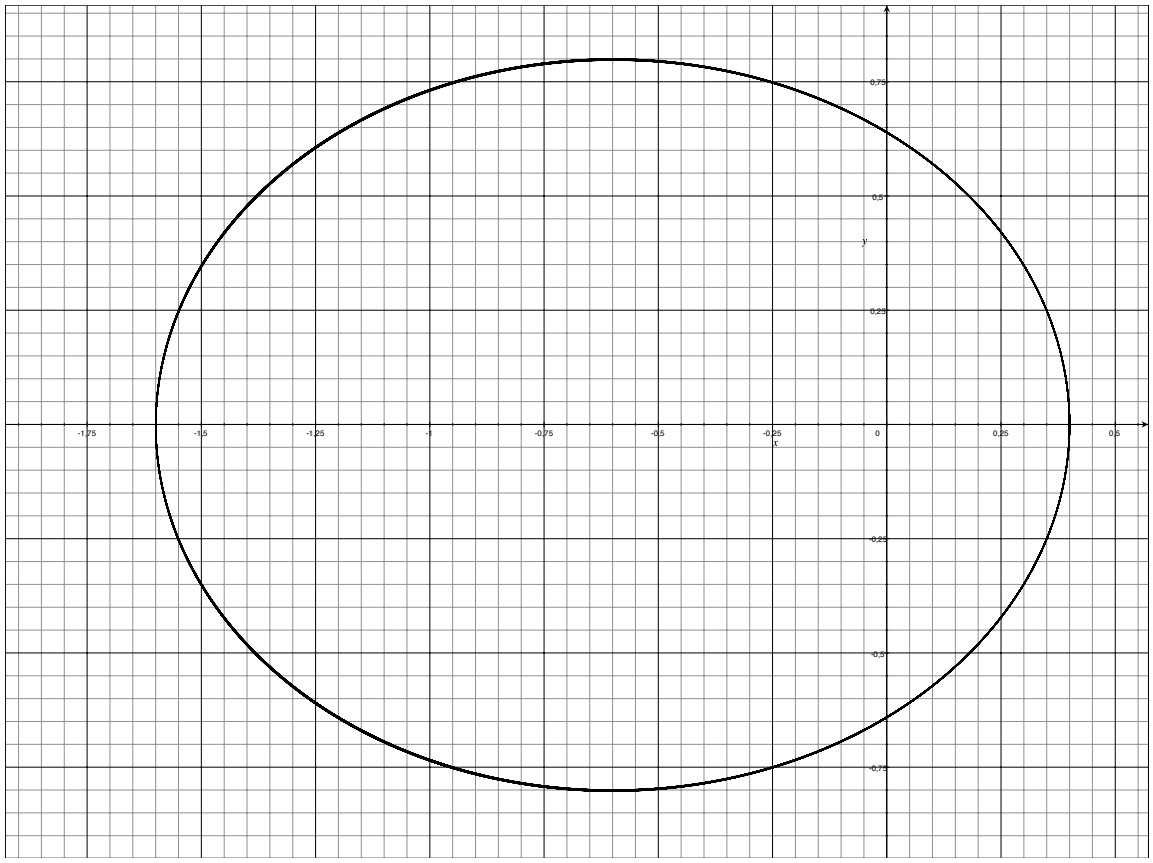

Figure 6.5: The Kepler system with $h = 0.01$ and 20000 steps.

Chapter 7

On a family of high order iterative methods under Kantorovich conditions

Abstract 7.0.1 *This chapter is devoted to the study of a class of high order iterative methods for nonlinear equations on Banach spaces. The motivation is to study these methods since they are used in the classical implementation of implicit methods. An analysis of the convergence under Kantorovich type conditions is proposed. Some numerical experiments, where the analyzed methods present better behavior than some classical schemes, are presented. These applications include the approximation of some quadratic and integral equations.*

7.1 Introduction

This paper deals with the approximation of nonlinear equations

$$F(x) = 0,$$

where $F : \Omega \subseteq X \to Y$ is a nonlinear operator between Banach spaces, using the following family of high order iterative methods:

$$\begin{cases} y_n &= x_n - F'(x_n)^{-1}F(x_n), \\ x_{n+1} &= y_n - (I + L_F(x_n) + L_F(x_n)^2 G_F(x_n))\, F'(x_n)^{-1}F(y_n), \end{cases} \tag{7.1.1}$$

where I is the identity operator on X and for each $x \in X$, $L_F(x)$ is the linear operator on $\Omega \subseteq X$ defined by

$$L_F(x) = F'(x)^{-1}F''(x)F'(x)^{-1}F(x),$$

assuming that $F'(x)^{-1}$ exists and $G_F : \Omega \subseteq X \to \mathcal{L}(X,X)$ is a given nonlinear operator (usually depending on the operator F and its derivatives). Here $\mathcal{L}(X,X)$ denotes the space of bounded linear operators from X to X.

The second step can be interpreted as an acceleration of the initial one (in our case Newton's method). Indeed, this family was introduced for scalar equations $f(t) = 0$ in [120], for any initial scheme, Traub's theorem reads:

Theorem 7.1.1 *For all sufficiently smooth function $g_f(x)$, the iterative method*

$$\left\{ \begin{array}{rcl} y_n & = & \Phi(x_n), \\ x_{n+1} & = & y_n - (1 + L_f(x_n) + L_f(x_n)^2 g_f(x_n)) \frac{f(y_n)}{f'(x_n)}, \end{array} \right. \tag{7.1.2}$$

has order of convergence $\min\{p+2, 2p\}$*, where p is the order of $\Phi(x)$.*

In this paper, we consider as the function $\Phi(x)$ the classical Newton method. We have mainly three reasons. First, because we can recover many well known high-order iterative methods. Second, because the domain of convergence of Newton's method is bigger than high order schemes [46]. Finally, since in practice is a good strategy to start with a simple method when we are not sufficiently close to the solution [8].

On the other hand, conditions are imposed on x_0 and on F in order to ensure the convergence of $\{x_n\}_n$ to a solution x^* of $F(x) = 0$. This analysis, usually known as Kantorovich type, is based on a relationship between the problem in a Banach space and a single nonlinear scalar equation which leads the behavior of the problem. *A priori* error estimates, depending only on the initial conditions, and, hence, the order of convergence can be obtained by using Kantorovich type theorems.

A review to the amount of literature on high order iterative methods in the two last decades (see for instance [1] and its references, or this incomplete list of recent papers [39, 41, 44, 47, 54, 55, 76, 86, 96, 111, 124, 126]) may reveal the importance of high order schemes. The main practical difficulty related to the classical third order iterative methods is the evaluation of the second order derivative. For a nonlinear system of m equations and m unknowns, the first Fréchet derivative is a matrix with m^2 entries, while the second Fréchet derivative has m^3 entries. This implies a huge amount of operations in order to evaluate every iteration. However, in some cases, the second derivative is easy to evaluate. Some clear examples of this case are the approximation of Hammerstein equations where the second Fréchet derivative is diagonal by blocks or quadratic equations where it is constant.

The structure of this paper is as follows: in Section 2 we present some particular examples of methods included in the family, in Section 3, we assert convergence and uniqueness theorems (Kantorovich type). Finally, some numerical experiments are

presented in Section 4. These applications include: quadratic (Riccati) equations and integral (Hammerstein) equations. In all these problems the proposed methods seem more efficient than second order methods.

7.2 A family of high order iterative methods

As was indicated in the introduction, we are interested in the study of the family of iterative methods

$$\left\{\begin{array}{rcl} y_n & = & x_n - \frac{f(x_n)}{f'(x_n)}, \\ x_{n+1} & = & y_n - (1 + L_f(x_n) + L_f(x_n)^2 g_f(x_n)) \frac{f(y_n)}{f'(x_n)}. \end{array}\right. \tag{7.2.1}$$

Note that the method (7.2.1) is equivalent to iterating the function M_f given by

$$M_f(x) = x - \frac{f(x)}{f'(x)} - (1 + L_f(x) + L_f(x)^2 g_f(x)) \frac{f(x - f(x)/f'(x))}{f'(x)},$$

that is,

$$x_{n+1} = M_f(x_n).$$

Particular examples of schemes included in the family with *non-smooth* functions $g_f(x)$ are:

- *Halley*
$$x_{n+1} = x_n - \left(\frac{1}{1 + \frac{1}{2} L_f(x_n)}\right) \frac{f(x_n)}{f'(x_n)},$$
- *Super-Halley*
$$x_{n+1} = x_n - \left(1 + \frac{L_f(x_n)}{2(1 - L_f(x_n))}\right) \frac{f(x_n)}{f'(x_n)},$$
- *Chebyshev*
$$x_{n+1} = x_n - \left(1 + \frac{1}{2} L_f(x_n)\right) \frac{f(x_n)}{f'(x_n)},$$
- *Chebyshev like methods.* For $0 \leq \alpha \leq 2$, we consider the α-methods
$$x_{n+1} = x_n - \left(1 + \frac{1}{2} L_f(x_n) + \alpha L_f(x_n)^2\right) \frac{f(x_n)}{f'(x_n)},$$

- *Two-step*

$$\left\{\begin{array}{rcl} y_n & = & x_n - \frac{f(x_n)}{f'(x_n)}, \\ x_{n+1} & = & y_n - \frac{f(y_n)}{f'(x_n)}. \end{array}\right.$$

These methods have order of convergence three that is smaller than the estimate $4 = \min\{2+2, 2\cdot 2\}$ in Traub's theorem (since $g_f(x)$ is non-smooth). For instance the above two-step method admits $g_f(x) = -\frac{1}{L_f(x)}$. Indeed, all these methods have the function f in the denominator.

On the other hand, considering different *smooth* functions $g_f(x)$, the following schemes are also particular examples in the family.

- The two step method ($g_f(x) = 0$)

$$M4: \qquad \left\{\begin{array}{rcl} y_n & = & x_n - \frac{f(x_n)}{f'(x_n)}, \\ x_{n+1} & = & y_n - (1 + L_f(x_n))\frac{f(y_n)}{f'(x_n)}, \end{array}\right.$$

  has order four.

- The two step method ($g_f(x) = \frac{1}{2}(\frac{5}{2} - L_{f'}(x))$)

$$M5: \left\{\begin{array}{rcl} y_n & = & x_n - \frac{f(x_n)}{f'(x_n)}, \\ x_{n+1} & = & y_n - \left(1 + L_f(x_n) + \frac{1}{2}(\frac{5}{2} - L_{f'}(x_n))L_f(x_n)^2\right)\frac{f(y_n)}{f'(x_n)}. \end{array}\right.$$

  has order five.

- We should start with other iterative functions $\Phi(x)$ and develop a similar analysis. For instance, starting with Chebyshev's method we can consider the method ($g_f(x) = \frac{1}{2}(3 - L_{f'}(x_n))$)

$$M6: \left\{\begin{array}{rcl} y_n & = & x_n - \left(1 + \frac{1}{2}L_f(x_n)\right)\frac{f(x_n)}{f'(x_n)}, \\ x_{n+1} & = & y_n - \left(1 + L_f(x_n) + \frac{1}{2}(3 - L_{f'}(x_n))L_f(x_n)^2\right)\frac{f(y_n)}{f'(x_n)}. \end{array}\right.$$

  that has order six [8]. We use this scheme only in the numerical section.

7.3 Semilocal convergence

Several techniques are usually considered to study the convergence of iterative methods, as we can see in the following papers [1, 16, 17, 45, 69]. Among these, the two most common are the based on majorant principle and on recurrence relations.

In this section, we analyze the semilocal convergence of the introduced family (7.1.1) under a generalization of Kantorovich conditions.

Namely, we assume that:

(C1) Let $x_0 \in \Omega$ such that $\Gamma_0 = F^{'}(x_0)^{-1}$ exists and $||\Gamma_0|| \leq \beta$.
(C2) $||\Gamma_0 F(x_0)|| \leq \eta$.
(C3) $||F^{''}(x)|| \leq M$ for all $x \in \Omega$.
(C4) $||F^{''}(x) - F^{''}(y)|| \leq K||x - y||$, $K > 0$, $x, y \in \Omega$.

Under these hypotheses it is possible to find a cubic polynomial in an interval $[a, b]$ such that $p(a) > 0 > p(b)$, $p^{'}(t) < 0$, $p^{''}(t) > 0$ and $p^{'''}(t) > 0$ in $[a, t^*]$, with t^* the unique simple solution of $p(t) = 0$, and verifying the following hypotheses:

For $t_0 \in [a, b]$ and $p(t_0) > 0$.
(H1) $||\Gamma_0|| \leq -\frac{1}{p^{'}(t_0)}$,
(H2) $||\Gamma_0 F(x_0)|| \leq -\frac{p(t_0)}{p^{'}(t_0)}$,
(H3) $||F^{''}(x)|| \leq p^{''}(t)$ for all $x \in \Omega$, $||x - x_0|| \leq t - t_0 \leq t^* - t_0$,
(H4) $||F^{''}(x) - F^{''}(y)|| \leq |p^{''}(u) - p^{''}(v)|$, with $||x - y|| \leq |u - v|$, $x, y \in \Omega$ and $u, v \in [a, t^*]$.

Some immediate properties of the polynomial may be obtained from the conditions above imposed:

1. $p(t)$ is decreasing in the interval $[a, t^*]$, since $p'(t) < 0$ in that interval.

2. $p(t) > 0$ in $[a, t^*[$.

3. $p'(t)$ is increasing and $p(t)$ is convex in $[a, t^*]$, since we have $p''(t) > 0$ in $[a, t^*]$.

4. $p''(t)$ is increasing in $[a, t^*]$, since $p'''(t) > 0$ in that interval.

From these properties it follows the next:

(a) The Newton map associate to $p(t)$, $N_p(t) = t - \frac{p(t)}{p'(t)}$, is increasing in $[a, t^*[$, $N_p(t^*) = t^*$ and $N_p'(t^*) = 0$.

(b) The function $L_p(t) = \frac{p(t)p''(t)}{p'(t)^2} > 0$ in $[a, t^*[$, since $p(t)$ and $p''(t)$ are strictly positive in that interval. Furthermore, $L_p(t^*) = 0$, since $p(t^*) = 0$ and $p'(t^*) \neq 0$.

In this paper, as in [103] (p. 43), we consider as the function $p(t)$ the following polynomial:

$$p\,(t) := \frac{K}{6}t^3 + \frac{M}{2}t^2 - \frac{1}{\beta}t + \frac{\eta}{\beta},$$

assuming

$$\eta \leq \frac{4K + M^2\beta - M\beta\sqrt{M^2 + 2K\beta}}{3\beta K(M + \sqrt{M^2 + 2K\beta})}.$$

If this last condition holds, then the cubic polynomial $p(t)$ has two roots t^* and t^{**} ($t^* \leq t^{**}$). We can choose a and b such that $0 < a < t^*$ and $b > \frac{2}{M\,\beta + \sqrt{M^2\,\beta^2 + 2\,K\,\beta}}$.

Moreover, we need some extra conditions associated with the operator G_F and the function g_p. We assume:

(Hg1) $||L_F(x)^2 G_F(x)|| \leq L_p(t)^2 G_p(t)$, for $||x - x_0|| \leq t - t_0 \leq t^* - t_0$,
(Hg2) $1 + L_p(t) + L_p(t)^2 g_p(t) \geq 0$,
(Hg3) $m^{'}(t) > 0$ in $[a, t^*[$, where

$$m(t) = t - \frac{p(t)}{p^{'}(t)} - (1 + L_p(t) + L_p(t)^2 g_p(t))\frac{p(t - \frac{p(t)}{p^{'}(t)})}{p^{'}(t)}.$$

All the methods considered in the above section have associated functions g_p that verify the three last conditions. With the two last hypotheses on g_p and the definition of p, following [103] (Corollary 2.2.2 in p. 31), the next result holds:

Proposition 7.3.1 *The sequence*

$$\left\{ \begin{array}{rcl} s_n & = & t_n - \frac{p(t_n)}{p'(t_n)}, \\ t_{n+1} & = & s_n - (1 + L_p(t_n) + L_p(t_n)^2 g_p(t_n))\frac{p(s_n)}{p^{'}(t_n)}, \end{array} \right.$$

starting from the above t_0 converges monotonically to t^* the real simple solution of $p(t) = 0$ in $[a, b]$.

We are now ready to prove the desired semilocal convergence.

Theorem 7.3.2 *Let us assume $x_0 \in \Omega$ and $t_0 \in [a, t^*]$ verifying the hypotheses (H1)-(H4) and (Hg1)-(Hg3) with*

$$\eta \leq \frac{4K + M^2\beta - M\beta\sqrt{M^2 + 2K\beta}}{3\beta K(M + \sqrt{M^2 + 2K\beta})}. \tag{7.3.1}$$

If $B(x_0, t^* - t_0) \subset \Omega$ then the sequence (7.1.1) is well defined and converges to x^* the unique solution of $F(x) = 0$ in $\overline{B(x_0, t^* - t_0)}$.

Moreover,

$$||x^* - x_n|| \leq t^* - t_n, \quad n \geq 0,$$

where

$$\begin{aligned} s_n &= t_n - \frac{p(t_n)}{p'(t_n)}, \\ t_{n+1} &= s_n - (1 + L_p(t_n) + L_p(t_n)^2 g_p(t_n))\frac{p(s_n)}{p'(t_n)}. \end{aligned}$$

Proof

By an induction process, it is possible to verify that

(i) $\left\|F'(x_n)^{-1}\right\| \leq -\frac{1}{p'(t_n)}$

(ii) $\|F(x_n)\| \leq p(t_n)$

and then,

(iv) $\|L_F(x_n)\| \leq L_p(t_n)$

and

(v) $\|x_{n+1} - x_n\| \leq t_{n+1} - t_n$.

The case $n = 0$ follows from the initial conditions on x_0 and t_0.

We now assume that the conditions are valid for n and we check them for $n + 1$.

(i)

$$F'(x_{n+1}) = F'(x_n)\left\{I - F'(x_n)^{-1}(F'(x_n) - F'(x_{n+1}))\right\}$$

Applying Taylor's theorem:

$$\begin{aligned}\left\|F'(x_n)^{-1}\left(F'(x_n)-F'(x_{n+1})\right)\right\| &\leq ||F^{'}(x_n)^{-1}||\left(\|F''(x_n)\|+\frac{1}{2}K\|x_n-x_{n+1}\|\right)\\ &\quad\cdot\|x_n-x_{n+1}\|\\ &\leq -\frac{1}{p'(t_n)}\left(p''(t_n)+\frac{1}{2}K(t_{n+1}-t_n)\right)\\ &\quad\cdot(t_{n+1}-t_n)\\ &= -\frac{1}{p'(t_n)}\left(p'(t_{n+1})-p'(t_n)\right)\\ &= 1-\frac{p'(t_{n+1})}{p'(t_n)}<1,\end{aligned}$$

because $p'(t)$ is increasing.

By applying the general invertibility criterion, $F'(x_{n+1})$ is invertible, and

$$\begin{aligned}\left\|F'(x_{n+1})^{-1}\right\| &\leq \left\|\left(I-F'(x_n)^{-1}\left(F'(x_n)-F'(x_{n+1})\right)\right)^{-1}\right\|\left\|F'(x_n)^{-1}\right\|\\ &\leq \frac{\left\|F'(x_n)^{-1}\right\|}{1-\left\|F'(x_n)^{-1}\left(F'(x_n)-F'(x_{n+1})\right)\right\|}\\ &\leq -\frac{1}{p'(t_n)\left[1-\left[1-\frac{p'(t_{n+1})}{p'(t_n)}\right]\right]}\\ &= -\frac{1}{p'(t_{n+1})}.\end{aligned}$$

(ii) Using the following Taylor expansion

$$\begin{aligned}F(y_n) &= F(x_n)+F^{'}(x_n)(y_n-x_n)+\frac{1}{2!}F^{''}(x_n)(y_n-x_n)^2\\ &+ \int_{x_n}^{y_n}\Big(F^{''}(x)-F^{''}(x_n)\Big)(y_n-x)dx,\end{aligned}$$

and by the definition of the method

$$F^{'}(x_n)(y_n-x_n) = -F^{'}(x_n)(F^{'}(x_n)^{-1}F(x_n)),$$

we obtain that

$$\begin{aligned}F(y_n) &= \frac{1}{2!}F^{''}(x_n)(F^{'}(x_n)^{-1}F(x_n))^2\\ &+ \int_{x_n}^{y_n}\Big(F^{''}(x)-F^{''}(x_n)\Big)(y_n-x)dx,\end{aligned}$$

and since

$$\begin{aligned} p(s_n) &= \frac{1}{2!}p^{''}(t_n)(p^{'}(t_n)^{-1}p(t_n))^2 \\ &+ \int_{t_n}^{s_n}\Big(p^{''}(x)-p^{''}(t_n)\Big)(s_n-t)dt, \end{aligned}$$

we conclude that

$$||F(y_n)|| \leq p(s_n).$$

Similarly from the following expansion

$$\begin{aligned} F(x_{n+1}) &= F(x_n)+F^{'}(x_n)(x_{n+1}-x_n)+\frac{1}{2!}F^{''}(x_n)(x_{n+1}-x_n)^2 \\ &+ \int_{x_n}^{x_{n+1}}\Big(F^{''}(x)-F^{''}(x_n)\Big)(x_{n+1}-x)dx, \end{aligned}$$

the definition of the method, the main hypotheses on G_F and the induction process, we obtain, using that $||F(y_n)|| \leq p(s_n)$ and that

$$\begin{aligned} F^{''}(x_n)(x_{n+1}-x_n)^2 &= F^{''}(x_n)F^{'}(x_n)^{-1}F(x_n)F^{'}(x_n)^{-1}F(x_n) \\ &+ F^{''}(x_n)F^{'}(x_n)^{-1}F(x_n)(I+L_F(x_n)+L_F(x_n)^2G_F(x_n))F^{'}(x_n)^{-1}F(y_n) \\ &+ F^{''}(x_n)F^{'}(x_n)^{-1}F(x_n)(I+L_F(x_n)+L_F(x_n)^2G_F(x_n))F^{'}(x_n)^{-1}F(y_n) \\ &+ F^{''}(x_n)((I+L_F(x_n)+L_F(x_n)^2G_F(x_n))F^{'}(x_n)^{-1}F(y_n))^2, \end{aligned}$$

the desired inequality:

$$||F(x_{n+1})|| \leq p(t_{n+1}).$$

In this situation, the theorem holds by applying the previous estimates directly to the formulas that describe the methods, we refer [103] (p. 41-42) for more details.
□

The estimates given in the present paper are optimal in the sense that the sequence associated with p verifies the inequalities with equalities.

7.4 Numerical experiments

We consider several problems where the presented high order methods can be considered as a good alternative to second order methods.

7.4.1 Approximation of Riccati's equations

In this first example, we consider quadratic equations, therefore the second Fréchet derivative is constant. Particular cases of this type of equations, that appear in many applications, are Riccati's equations [3, 61, 62]. For instance, if we consider the problem of calculating feedback controls for systems modeled by partial differential or delay differential equations, a classical controller design objective will be to find a control $u(t)$ for the state $x(t)$ such that the objective function

$$\int_0^\infty < Cx(t), Cx(t) > +u^* Ru(t)\ dt$$

is minimized, where R is a positive defined matrix and the observation $C \in \mathcal{L}(X, \mathbb{R}^d)$. In practice, the control is calculated through approximation. This leads to solving an algebraic Riccati equation

$$A^*P + PA - PBR^{-1}B^*P = -C^*C$$

for a feedback operator

$$K = -R^{-1}B^*P,$$

see [81, 82] for more details.

In the general case, an algebraic Riccati's equation is given by [79]

$$R(X) = XDX - XA - A^TX - C = 0, \tag{7.4.1}$$

where $D, A, C \in \mathbb{R}^{n\times n}$ are given matrices, D symmetric and $X \in \mathbb{R}^{n\times n}$ is the unknown.

In this case,

$$\begin{aligned} R^{'}(X)Y &= (XD - A^T)Y + Y(DX - A), \\ R^{''}(X)YZ &= YDZ + ZDY. \end{aligned}$$

In particular, the second derivative is constant. In this case, the Kantorovich conditions for Newton's methods have the compact form

$$\|R^{-1}(X_0)R^{''}(X_0)\|\|R^{-1}(X_0)R(X_0)\| \leq \frac{1}{2}. \tag{7.4.2}$$

Moreover, this hypothesis also gives the convergence for the high order methods [3].

Then, using a matricial norm

$$\|R^{''}(X)YZ\| \leq 2\|D\|\|Y\|\|Z\|$$

and

$$\|R^{''}(X)\| \leq 2\|D\|.$$

Given a symmetric initial guess $X_0 \in \mathbb{R}^{n\times n}$, to obtain $R^{'}(X_0)^{-1}$ we solve the equation

$$R^{'}(X_0)Y = (X_0D - A^T)Y + Y(DX_0 - A) = Z.$$

This equation has solution if $DX_0 - A$ is stable [79], that is, all its eigenvalues have negative real part. In this case

$$Y = R^{'}(X_0)^{-1}Z = -\int_0^{\infty} \exp((DX_0 - A)^T t)Z\exp((DX_0 - A)t)dt.$$

Next, to illustrate the previous results, we consider the algebraic Riccati equation (7.4.1) with matrix

$$D = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} = C, \qquad A = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix},$$

and the starting point

$$X_0 = \begin{pmatrix} -3/2 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & -5/4 \end{pmatrix}.$$

In this case, the algebraic Riccati equation has exact solution

$$X^* = \begin{pmatrix} -\sqrt{2} & 0 & 1 \\ 0 & 1-\sqrt{2} & 0 \\ 1 & 0 & -\sqrt{2} \end{pmatrix}. \tag{7.4.3}$$

Besides, from the aforesaid starting point it follows that $DX_0 - A$ is a stable matrix.

Now, considering the stopping criterion $\|X_n - X^*\| < 10^{-50}$ in Table 7.1, we obtain the errors $\|X_n - X^*\|$. If we now analyze the following computational order of convergence [56]:

$$\rho \approx \ln \frac{\|X_{n+1} - X^*\|}{\|X_n - X^*\|} / \ln \frac{\|X_n - X^*\|}{\|X_{n-1} - X^*\|}, \qquad n \in \mathbb{N}, \tag{7.4.4}$$

we observe that method M6 has computationally the order of convergence at least six. See Table 7.2, where ρ_{N}, ρ_{CH} and ρ_{M6} denote respectively the computational order of convergence of the three last methods.

n	Newton	Chebyshev	M6
1	$8.57864\ldots\cdot 10^{-2}$	$3.92135\ldots\cdot 10^{-2}$	$8.63800\ldots\cdot 10^{-3}$
2	$2.45310\ldots\cdot 10^{-3}$	$1.60604\ldots\cdot 10^{-5}$	$7.95704\ldots\cdot 10^{-14}$
3	$2.12390\ldots\cdot 10^{-6}$	$1.03568\ldots\cdot 10^{-15}$	
4	$1.59486\ldots\cdot 10^{-12}$	$2.77730\ldots\cdot 10^{-46}$	
5	$8.99292\ldots\cdot 10^{-25}$		
6	$2.85928\ldots\cdot 10^{-49}$		

Table 7.1: Errors for the Newton, Chebyshev methods and M6

n	ρ_{N}	ρ_{CH}	ρ_{M6}
1	$2.25751\ldots$	$3.30896\ldots$	$6.56567\ldots$
2	$1.98391\ldots$	$3.00811\ldots$	
3	$1.99975\ldots$	$3.00000\ldots$	
4	$1.99999\ldots$		
5	$1.99999\ldots$		

Table 7.2: The computational order of convergence for the Newton, Chebyshev methods and M6

In comparison with the classical Newton's method, the extra computational cost per iteration of method M6, is only two new evaluations of the operator F, and two extra matrix-vector multiplications. Moreover, the same as Newton's method only a LU decomposition is necessary. Thus, M6 is more efficient.

See [9] for more details.

7.4.2 Approximation of Hammerstein equations

We shall consider an important special case of integral equation, the Hammerstein equation

$$u(s) = \psi(s) + \int_0^1 H(s,t) f(t, u(t)) dt. \tag{7.4.5}$$

These equations are related with boundary value problems for differential equations. For some of them, high order methods using second derivatives are useful for their effective (discretized) solution.

The discrete version of (7.4.5) is

$$x^i = \psi(t_i) + \sum_{j=0}^{m} \gamma_j H(t_i, t_j) f(t_j, x^j), \quad i = 0, 1, \ldots, m, \tag{7.4.6}$$

where $0 \leq t_0 < t_1 < \ldots < t_m \leq 1$ are the grid points of some quadrature formula $\int_0^1 f(t)dt \approx \sum_{j=0}^m \gamma_j f(t_j)$, and $x^i = x(t_i)$.

The second Fréchet derivative of the associated discrete system is diagonal by blocks.

Let the Hammerstein equation

$$x(s) = 1 - \frac{1}{4} \int_0^1 \frac{s}{t+s} \frac{1}{x(t)} dt, \quad s \in [0, 1]. \tag{7.4.7}$$

The discretization of this equation verifies the Lipschitz condition of our Kantorovich theorem [1].

We consider $m = 20$ in the quadrature trapezoidal formula and as exact solution the one obtained numerically by Newton method. In table 7.3, we summarize the numerical results for different methods in the family: Newton, Halley and M4. We consider as initial guess $x_0(s) = 1.5$.

Since the second derivative is diagonal by blocks, its application has a computational cost of order $O(m^2)$. Thus, the computational cost in each iteration of the three schemes is, for m sufficiently big, of the same order ($O(m^3)$ due to the LU decomposition). Note that we only have to do a factorization in each iteration of the three schemes. As conclusion, the scheme M4 (order four) is the most efficient for m sufficiently big.

n	Newton	Halley	M4
1	$2.35786\ldots \cdot 10^{-2}$	$4.23125\ldots \cdot 10^{-3}$	$5.00638\ldots \cdot 10^{-4}$
2	$1.60604\ldots \cdot 10^{-4}$	$1.06034\ldots \cdot 10^{-6}$	$6.55602\ldots \cdot 10^{-15}$
3	$3.30548\ldots \cdot 10^{-8}$	$1.00158\ldots \cdot 10^{-17}$	$8.23560\ldots \cdot 10^{-60}$
4	$3.11276\ldots \cdot 10^{-16}$	$2.13492\ldots \cdot 10^{-49}$	
5	$1.12645\ldots \cdot 10^{-32}$		
6	$2.89613\ldots \cdot 10^{-65}$		

Table 7.3: Errors for the Newton, Halley and M4 methods

See [4] for other related problems.

Conclusions

Summing up, in this paper we have studied a family of high order iterative methods. Mainly, the theoretical analysis we did allows to ensure convergence conditions for all these schemes. We established priori error bounds for them and consequently their order. We have presented different applications where we may add that in these cases the analyzed high order methods are more efficient than simpler second order methods.

Chapter 8

A concluding remark

A new variational approach to the analysis and numerical implementation of regular ODEs has been recently introduced in ([13, 15]). Because of its flexibility and simplicity, it can easily be extended to treat other types of ODEs like differential-algebraic equations (DAEs), and delay-differential equations (DDEs). This has been precisely the main motivation for this thesis: to explore how well those ideas can be adapted to other frameworks. In particular, a lot of attention has been paid to DAEs, extending to this context some of the analytical results, and performing various numerical tests that confirm that indeed the variational perspective is worth pursuing. One remarkable feature is that this point of view only requires to count on good numerical schemes for linear problems, and this is the reason why it fits so well in other scenarios. After an initial, promising incursion for DDEs, we have also investigated resolution with variable step, and the extension to Hamiltonian systems. Because of the many good qualities of this viewpoint, it can be considered and implemented in essentially all fields where differential equations are relevant. There is, then, a long way to go.

Chapter 9

ANNEXE I: Model problems

Abstract 9.0.1 *This chapter presents a collection of problems where we can test the real behavior of a new differential solver.*

9.1 Example 1: Chemical Akzo Nobel problem

9.1.1 Origin of the problem

The problem originates from Akzo Nobel Central Research in Arnhem, The Netherlands. It describes a chemical process, in which two species, MBT and CHA, are mixed, while oxygen is continuously added. The resulting species of importance is CBS. The reaction equations, as given by Akzo Novel, are

$$2MBT + \frac{1}{2}O_2 \xrightarrow{k_1} MBTS + H_2O,$$

$$CBS + MBT \underset{k_2}{\overset{k_2/K}{\rightleftharpoons}} MBTS + CHA,$$

$$MBT + 2CHA + O_2 \xrightarrow{k_3} BT + \text{sulfate},$$

$$MBT \cdot CHA + \frac{1}{2}O_2 \xrightarrow{k_4} CBS + H_2O,$$

$$MBT + CHA \rightleftarrows MBT \cdot CHA.$$

The last equation describes an equilibrium

$$K_s^1 = \frac{[MBT \cdot CHA]}{[MBT] \cdot [CHA]},$$

while the others describe reactions, whose velocities are given by

$$\begin{aligned} r_1 &= k_1 \cdot [MBT]^4 \cdot [O_2]^{\frac{1}{2}}, \\ r_2 &= k_2 \cdot [MBTS] \cdot [CHA], \\ r_3 &= \frac{k_2}{K} \cdot [MBT] \cdot [CBS], \\ r_4 &= k_3 \cdot [MBT] \cdot [CHA]^2, \\ r_5 &= k_4 \cdot [MBT \cdot CHA]^2 \cdot [O_2]^{\frac{1}{2}}, \end{aligned}$$

respectively. Here the square brackets '[ ]' denote concentrations.

The inflow of oxygen per volume unit is denoted by F_{in}, and satisfies $F_{in} = klA \cdot (\frac{p(O_2)}{H} - [O_2])$, where klA is the mass transfer coefficient, H is the Henry constant and $p(O_2)$ is the partial oxygen pressure. $p(O_2)$ is assumed to be independent of $[O_2]$. The parameters $k_1, k_2, k_3, k_4, K, klA, H$ and $p(O_2)$ are constants. The process is started by mixing 0.437 mol/liter $[MBT]$ with 0.367 mol/liter $[MBT \cdot CHA]$. The concentration of oxygen at the beginning is 0.00123 mol/liter. Initially, no other species are present. The simulation is performed on the time interval $[0, 180]$ minutes. Identifying the concentrations $[MBT]$, $[O_2]$, $[MTBS]$, $[CHA]$, $[CBS]$, $[MBT \cdot CHA]$ with $y_1, \ldots, y_6$, respectively, one easily arrives at the mathematical formulation of the problem.

9.1.2 Mathematical description of the problem

The problem is of the form

$$\frac{dy}{dt} = f(y), \quad y(0) = y_0,$$

with $y \in \mathbf{R}^6, 0 \leq t \leq 180$.

The function f defined by

$$f(y) = \begin{pmatrix} -2r_1 & +r_2 & -r_3 & -r_4 & & \\ \frac{-1}{2}r_1 & & & -r_4 & \frac{-1}{2}r_5 & +F_in \\ r_1 & -r_2 & +r_3 & & & \\ & -r_2 & +r_3 & -2r_4 & & \\ & +r_2 & -r_3 & & +r_5 & \\ & & & & -r_5 & \end{pmatrix},$$

where the r_i and F_{in} are auxiliary variables, given by

$$\begin{aligned} r_1 &= k_1 \cdot y_1^4 \cdot y_2^{\frac{1}{2}}, \\ r_2 &= k_2 \cdot y_3 \cdot y_4, \\ r_3 &= \frac{k_2}{K} \cdot y_1 \cdot y_5, \\ r_4 &= k_3 \cdot y_1 \cdot y_4^2, \\ r_5 &= k_4 \cdot y_6^2 \cdot y_2^{\frac{1}{2}}, \\ F_{in} &= klA \cdot (\frac{p(O_2)}{H} - y_2). \end{aligned}$$

The values of the parameters $k_1, k_2, k_3, k_4, K, klA, p(O_2)$ and H are

$$\begin{aligned} k_1 &= 18.7, \\ k_2 &= 0.58, \\ k_3 &= 0.09, \\ k_4 &= 0.42, \\ k_5 &= 34.4, \\ klA &= 3.3, \\ p(O_2) &= 0.9, \\ H &= 737. \end{aligned}$$

Finally, the initial vector y_0 is given by

$$y_0 = \begin{pmatrix} 0.437 \\ 0.00123 \\ 0 \\ 0 \\ 0 \\ 0.367 \end{pmatrix}.$$

9.1.3 General information

This is a stiff system of 6 non-linear differential equations. It has taken from [66, 87].

9.2 Example 2: Problem HIRES

9.2.1 Origin of the problem

The HIRES problem originates from plant physiology and describes how light is involved in morphogenesis. To be precise, it explains the High Irradiance Responses (HIRES) of photomorphogenesis on the basis of phytochrome, by means of a chemical reaction involving eight reactants. It has been promoted as a test problem by Gottwald in [53, 87]. The reaction scheme is given in Figures 9.1 and 9.2 . P_r and P_{fr} refer to the red and far-red absorbing form of phytochrome, respectively. They can be bound by two receptors X and X', partially influenced by the enzyme E. The values of the parameters were taken from [66].

$k_1 = 1.71$	$k_+ = 280$
$k_2 = 0.43$	$k_- = 0.69$
$k_3 = 8.32$	$k^* = 0.69$
$k_4 = 0.69$	$O_{k_s} = 0.0007$
$k_5 = 0.0354$	
$k_6 = 8.32$	

For more details, we refer to [107].

Identifying the concentrations of $P_r, P_{fr}, P_rX, P_{fr}X, P_rX', P_{fr}X', P_{fr}X'E$ and E with y_i, $i \in \{1, \cdots, 8\}$, respectively, the differential equations can be obtained easily. See [118]for a more detailed description of this modeling process. The end point of the integration interval, 321.8122, was chosen arbitrarily [87].

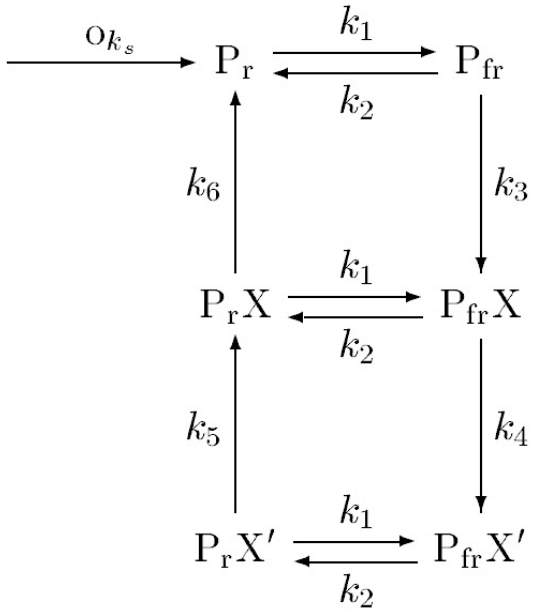

Figure 9.1: Reaction scheme for HIRES

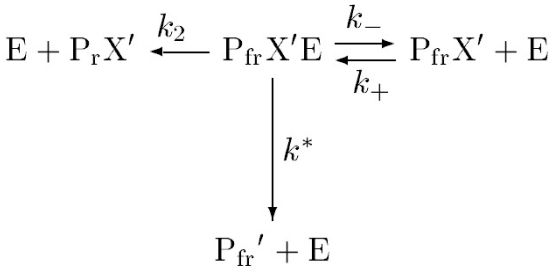

Figure 9.2: Reaction scheme for HIRES

9.2.2 Mathematical description of the problem

The problem is of the form

$$\frac{dy}{dt} = f(y), \quad y(0) = y_0$$

with $y \in \mathbf{R}^8, 0 \leq t \leq 321.8122$.

The function f is defined by

$$f(y) = \begin{pmatrix} -1.71y_1 & +0.43y_2 & +8.32y_3 & +0.0007 & \\ +1.71y_1 & -8.75y_2 & & & \\ -10.03y_3 & +0.43y_4 & +0.035y_5 & & \\ +8.32y_2 & +1.71y_3 & -1.12y_4 & & \\ -1.745y_5 & +0.43y_6 & +0.43y_7 & & \\ -280y_6y_8 & +0.69y_4 & +1.71y_5 & -0.43y_6 & +0.69y_7 \\ 280y_6y_8 & -1.81y_7 & & & \\ -280y_6y_8 & +1.81y_7 & & & \end{pmatrix}.$$

The initial vector y_0 is given by $(1, 0, 0, 0, 0, 0, 0, 0.0057)^T$.

9.2.3 General information

This IVP is a stiff system of eight non-linear ordinary differential equations. It was proposed by Schäfer in 1975 [107]. The name HIRES was given by Hairer & Wanner [66]. It refers to "High Irradiance Response", which is described by this IVP.

9.3 Example 3: Andrew's squeezing mechanism

9.3.1 Origin of the problem

The problem describes the motion of seven rigid bodies connected by joints without friction. It was promoted by [52] and [90] as a test problem for numerical codes. In [66] we can find a description of the problem and of the modeling process in full detail.

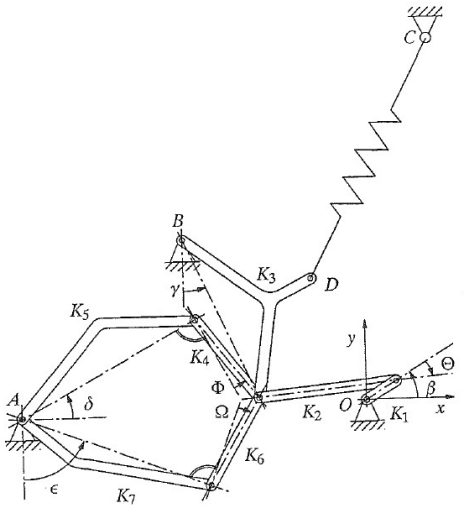

Figure 9.3: Andrews mechanism

9.3.2 Mathematical description of the problem

The problem is of the form

$$\begin{aligned} M(q)\ddot{q} &= f(q,\dot{q}) - G^T(q)\lambda, \\ 0 &= g(q), \end{aligned}$$

with initial conditions

$$q(0) = q_0, \dot{q}(0) = \dot{q}_0, \ddot{q}(0) = \ddot{q}_0, \lambda(0) = \lambda_0.$$

Here,

$$0 \leq t \leq 0.03,$$
$$q \in \mathbf{R}^7,$$
$$\lambda \in \mathbf{R}^6,$$
$$M : \mathbf{R}^7 \to \mathbf{R}^{7\times 7},$$
$$f : \mathbf{R}^{14} \to \mathbf{R}^7,$$
$$g : \mathbf{R}^7 \to \mathbf{R}^6,$$
$$G = \frac{dg}{dq}.$$

The function $M = (M_{ij}(q))$ is given by:

$$\begin{aligned}
M_{11}(q) &= m_1 \cdot ra^2 + m_2(rr^2 - 2da \cdot rr \cdot \cos q_2 + da^2) + I_1 + I_2,\\
M_{21}(q) &= M_{12}(q) = m_2(da^2 - 2da \cdot rr \cdot \cos q_2) + I_2,\\
M_{22}(q) &= m_2 \cdot da^2 + I_2,\\
M_{33}(q) &= m_3(sa^2 + sb^2) + I_3,\\
M_{44}(q) &= m_4(e - ea)^2 + I_4,\\
M_{54}(q) &= M_{45}(q) = m_4((e - ea)^2 + zt(e - ea)\sin q_4) + I_4,\\
M_{55}(q) &= m_4(zt^2 + 2zt(e - ea)\sin q_4 + (e - ea)^2) + m_5(ta^2 + tb^2) + I_4 + I_5,\\
M_{66}(q) &= m_6(zf - fa)^2 + I_6,\\
M_{76}(q) &= M_{67}(q) = m_6((zf - fa)^2 - u(zf - fa)\sin q_6) + I_6,\\
M_{77}(q) &= m_6((zf - fa)^2 - 2u(zf - fa)\sin q_6 + u^2) + m_7(ua^2 + ub^2) + I_6 + I_7,\\
M_{ij}(q) &= 0 \quad \text{for all other cases.}
\end{aligned}$$

The function $f = f_i(q, \dot{q})$ reads:

$$\begin{aligned}
f_1(q, \dot{q}) &= mom - m_2 \cdot da \cdot rr \cdot \dot{q}_2(\dot{q}_2 + 2\dot{q}_1)\sin(q_2),\\
f_2(q, \dot{q}) &= m_2 \cdot da \cdot rr \cdot \dot{q}_1^2 \cdot \sin q_2,\\
f_3(q, \dot{q}) &= F_x(sc \cdot \cos q_3 - sd \cdot \sin q_3) + F_y(sd \cdot \cos q_3 + sc \cdot \sin q_3),\\
f_4(q, \dot{q}) &= m_4 \cdot zt \cdot (e - ea)\dot{q}_5^2 \cdot \cos q_4,\\
f_5(q, \dot{q}) &= -m_4 \cdot zt(e - ea)\dot{q}_4(\dot{q}_4 + 2\dot{q}_5)\cos q_4,\\
f_6(q, \dot{q}) &= -m_6 \cdot u(zf - fa)\dot{q}_7^2 \cos q_6,\\
f_7(q, \dot{q}) &= m_6 \cdot u(zf - fa)\dot{q}_6(\dot{q}_6 + 2\dot{q}_7)\cos q_6.
\end{aligned}$$

F_x and F_y are defined by:

$$\begin{aligned}
F_x &= F(xd - xc),\\
F_y &= F(yd - yc),\\
F &= -c_0\frac{(L - l_0)}{L},\\
L &= \sqrt{(xd - xc)^2 + (yd - yc)^2},\\
xd &= sd \cdot \cos q_3 + sc \cdot \sin q_3 + xb,\\
yd &= sd \cdot \sin q_3 - sc \cdot \cos q_3 + yb.
\end{aligned}$$

The function $g = (g_i(q))$ is given by:

$$\begin{aligned}
g_1(q) &= rr \cdot \cos q_1 - d \cdot \cos(q_1 + q_2) - ss \cdot \sin q_3 - xb, \\
g_2(q) &= rr \cdot \sin q_1 - d \cdot \sin(q_1 + q_2) + ss \cdot \cos q_3 - yb, \\
g_3(q) &= rr \cdot \cos q_1 - d \cdot \cos(q_1 + q_2) - e \cdot \sin(q_4 + q_5) - zt \cdot \cos q_5 - xa, \\
g_4(q) &= rr \cdot \sin q_1 - d \cdot \sin(q_1 + q_2) + e \cdot \cos(q_4 + q_5) - zt \cdot \sin q_5 - ya, \\
g_5(q) &= rr \cdot \cos q_1 - d \cdot \cos(q_1 + q_2) - zf \cdot \cos(q_6 + q_7) - u \cdot \sin q_7 - xa, \\
g_6(q) &= rr \cdot \sin q_1 - d \cdot \sin(q_1 + q_2) - zf \cdot \sin(q_6 + q_7) - u \cdot \cos q_7 - ya.
\end{aligned}$$

The constants arising in these formulas are given by:

$m_1 = 0.04325$	$I_1 = 2.194.10^{-6}$	$ss = 0.035$
$m_2 = 0.00365$	$I_2 = 4.410.10^{-7}$	$sa = 0.01874$
$m_3 = 0.02373$	$I_3 = 5.255.10^{-6}$	$sb = 0.01043$
$m_4 = 0.00706$	$I_4 = 5.667.10^{-7}$	$sc = 0.018$
$m_5 = 0.07050$	$I_5 = 1.169.10^{-5}$	$sd = 0.02$
$m_6 = 0.00706$	$I_6 = 5.667.10^{-7}$	$ta = 0.02308$
$m_7 = 0.05498$	$I_7 = 1.912.10^{-5}$	$tb = 0.00916$
$xa = -0.06934$	$d = 0.028$	$u = 0.04$
$ya = -0.00227$	$d_a = 0.0115$	$ua = 0.01228$
$xb = -0.03635$	$e = 0.02$	$ub = 0.00449$
$yb = 0.03273$	$ea = 0.01421$	$zf = 0.02$
$xc = 0.014$	$rr = 0.007$	$zt = 0.04$
$yc = 0.072$	$ra = 0.00092$	$fa = 0.01421$
$c_0 = 4530$	$l_0 = 0.07785$	$mom = 0.033$

The initial values are $y_0 = (q_0, \dot{q}_0, \ddot{q}_0, \lambda_0)^T$ where

$$q_0 = \begin{pmatrix}
-0.061713890014276449635894845 8001 \\
0 \\
0.455279819163070380255912382449 \\
0.222668390165885884674473185609 \\
0.487364979543842550225598953530 \\
-0.222668390165885884674473185609 \\
1.230547444549821192497350155 68
\end{pmatrix},$$

$$\dot{q}_0 = (0, 0, 0, 0, 0, 0, 0)^T,$$

$$\ddot{q}_0 = \begin{pmatrix} 14222.443919954113870591162588 7 \\ -10666.832939965585402943371941 5 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix},$$

$$\lambda_0 = \begin{pmatrix} 98.566870396241089605765498217 0 \\ -6.122688344255662655031143931 22 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}.$$

9.3.3 General information

The problem is a non-stiff second order DAE of index 3, consisting of 7 differential and 6 algebraic equations. The problem is transformed into the form

$$\tilde{M}\frac{dy}{dt} = \tilde{f}(y), \quad y(0) = y_0,$$

with

$$y = \begin{pmatrix} q \\ \dot{q} \\ \ddot{q} \\ \lambda \end{pmatrix},$$

$$\tilde{M} = \begin{pmatrix} I & 0 & 0 & 0 \\ 0 & I & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix},$$

$$\tilde{f} = \begin{pmatrix} \dot{q} \\ \ddot{q} \\ M(q)\ddot{q} - f(q,\dot{q}) + G^T(q)\lambda \\ g(q) \end{pmatrix}.$$

9.4 Example 4: Charge pump

9.4.1 Origin of the problem

The Charge-pump circuit shown in Figure 9.4 consists of two capacitors and an n-channel MOS-transistor.The nodes gate, source, gate, and drain of the MOS-transistor are connected with the nodes 1, 2, 3, and Ground, respectively. In formulating the circuit equations, the transistor is replaced by four non-linear current sources in each of the connecting branches. They model the transistor.

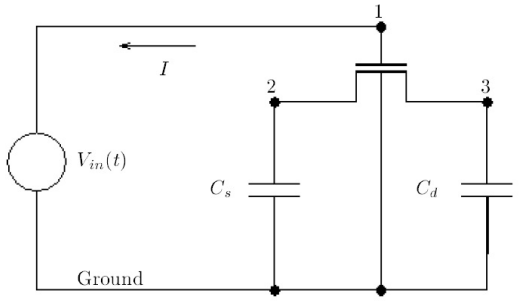

Figure 9.4: Circuit diagram of Charge-pump circuit

After inserting the transistor model in the circuit, we get the final circuit, which can be obtained from the circuit in Figure 9.4 by applying the following changes:

- Remove the transistor and replace it by a solid line between the nodes 2 and 3. The point where the lines 2-3 and 1-Ground cross each other becomes a node, which will be denoted by T.

- There are current sources between nodes 1 and T, between 2 and T and between 3 and T. There is also current source between the ground and node T, but as the node Ground does not enter the circuit equations, it will not be discussed. The currents produced by these sources are written as the derivatives of charges: current from 1 to T: Q_G', from T to 2: Q_S' and from T to 3: Q_D'. Here, the functions Q_G, Q_S and Q_D depend on the voltage drops U_1, $U_1 - U_2$ and $U_1 - U_3$, where U_i denotes the potential in node i.

The unknowns in the circuit are given by:

- The charges produced by the current sources: Y_{T1}, Y_{T2}, Y_{T3}. They are aliases for respectively Q_G, Q_S and Q_D. Consequently, $Y_{Ti}^{'}$ is the current between node T and node i.
- The charges Y_S and Y_D in the capacitors C_S and C_D.
- Potentials in nodes 1 to 3: U_1, U_2, U_3.
- The current through the voltage source $V_{in}(t) : I$.

In terms of these physical variables, the vector y introduced earlier reads

$$y = (Y_{T1}, Y_s, Y_{T2}, Y_D, Y_{T3}, U_1, U_2, U_3, I)^T.$$

Now, the following equations hold:

$$\begin{aligned} Y_{T1}^{'} &= -I, \\ Y_S^{'} + Y_{T2}^{'} &= 0, \\ Y_D^{'} + Y_{T3}^{'} &= 0, \\ U_1 &= V_{in}(t). \end{aligned}$$

The charges depend on the potentials and are given by

$$\begin{aligned} Y_{T1} &= Q_G(U_1, U_1 - U_2, U_1 - U_3), \\ Y_S &= C_S\ U_2, \\ Y_{T2} &= Q_S(U_1, U_1 - U_2, U_1 - U_3), \\ Y_D &= C_D\ U_3, \\ Y_{T3} &= Q_D(U_1, U_1 - U_2, U_1 - U_3). \end{aligned}$$

The functions Q_G, Q_S, and Q_D are given in the next section.

9.4.2 Mathematical description of the problem

The problem is of the form

$$M\frac{dy}{dt} - f(t, y(t)) = 0, \qquad y(0) = 0,$$

with

$$y \in \mathbf{R}^9, \qquad 0 \le t \le 1.2 \cdot 10^{-6}.$$

The matrix M is the zero matrix except for the minor $M_{1\cdots3,1\cdots5}$ that is given by

$$M_{1\cdots3,1\cdots5} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 \end{pmatrix}.$$

The function f is defined by

$$f(t,y) = \begin{pmatrix} -y_9 \\ 0 \\ 0 \\ -y_6 + Vin(t) \\ y_1 - Q_G(\upsilon) \\ y_2 - C_S \cdot y_7 \\ y_3 - Q_S(\upsilon) \\ y_4 - C_D \cdot y_8 \\ y_5 - Q_D(\upsilon) \end{pmatrix},$$

with $\upsilon := (\upsilon_1, \upsilon_2, \upsilon_3) = (y_6, y_6 - y_7, y_6 - y_8)$, $\quad C_D = 0.4 \cdot 10^{-12}$. The functions Q_G, Q_S and Q_D are given by:

1. if $V_1 \leq V_{FB} := U_{T0} - \gamma\sqrt{\Phi} - \Phi$ then

$$\begin{aligned} Q_G(\upsilon) &= C_{ox}(\upsilon_1 - V_{FB}), \\ Q_S(\upsilon) &= Q_D(\upsilon) = 0, \end{aligned}$$

   with $C_{ox} = 4 \cdot 10^{-12}$ and $U_{T0} = 0.2$, $\gamma = 0.035$ and $\Phi = 1.01$,

2. if $\upsilon_1 > V_{FB}$ and $\upsilon_2 \leq U_{TE} := U_{T0} + \gamma(\sqrt{\Phi + \upsilon_1 - \upsilon_2} - \sqrt{\Phi})$ then

$$\begin{aligned} Q_G(\upsilon) &= C_{ox}\gamma(\sqrt{(\gamma/2)^2 + \upsilon_1 - V_{FD}} - \gamma/2), \\ Q_S(\upsilon) &= Q_D(\upsilon) = 0, \end{aligned}$$

3. if $\upsilon_1 > V_{FB}$ and $\upsilon_2 > U_{TE}$ then

$$\begin{aligned} Q_G(\upsilon) &= C_{ox}[\frac{2}{3}(U_{GDT} + U_{GST} - \frac{U_{GDT} \quad U_{GST}}{U_{GDT} + U_{GST}}) + \gamma\sqrt{\Phi - U_{BS}}], \\ Q_S(\upsilon) &= -\frac{1}{2}(Q_G - C_{ox}\gamma\sqrt{\Phi - U_{BS}}). \end{aligned}$$

Here, U_{BS}, U_{GST} and U_{GDT} are given by

$$\begin{aligned} U_{BS} &= \upsilon_2 - \upsilon_1, \\ U_{GST} &= \upsilon_2 - U_{TE}, \end{aligned}$$

$$U_{GDT} = \begin{cases} \upsilon_3 - U_{TE} & \text{for} \quad \upsilon_3 > U_{TE}, \\ 0 & \text{for} \quad \upsilon_3 \leq U_{TE}. \end{cases} \tag{9.4.1}$$

The function $V_{in}(t)$ is defined using $\tau = (10^8 \cdot t) mod 120$ by

$$V_{in}(t) = \begin{cases} 0 & \text{if} \quad \tau < 50, \\ 20(\tau - 50) & \text{if} \quad 50 \leq \tau < 60, \\ 20 & \text{if} \quad 60 \leq \tau < 110, \\ 20(120 - \tau) & \text{if} \quad r \geq 110. \end{cases}$$

Finally, the initial value y_0 reads

$$y_0 = (0, 0, 0, 0, 0, 0, 0, 0, 0)^T.$$

9.4.3 General information

The problem is a stiff DAE of index 2, consisting of 3 differential and 6 algebraic equations. It has been contributed by Michael Günther, Georg Denk and Uwe Feldmann [59].

9.5 Example 5: Transistor amplifier

9.5.1 Origin of the problem

The problem originates from electrical circuit analysis. It is a model for the transistor amplifier. The diagram of the circuit is given in the figure 9.5.

Here U_e is the input signal and U_8 is the amplified exit voltage. To formulate the governing equations, Kirchoffs Current Law is used in each numbered node. This law states that the total sum of all currents entering a node must be zero. All currents passing through the circuit components can the expressed in terms of the unknown voltages $U_1, \cdots, U_8$. Consider for instance node 1. The current I_{C1}, passing through capacitor C_1 is given by

$$I_{C_1} = \frac{d}{dt}(C_1(U_2 - U_1)),$$

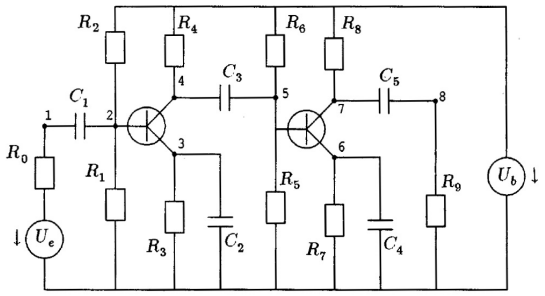

Figure 9.5: Circuit diagram of transistor amplifier

and the current I_{R_0} passing trough the resistor R_0 by

$$I_{R_0} = \frac{U_e - U_1}{R_0}.$$

Here, the currents are directed towards node 1 if the current is positive. A similar derivation for the other nodes gives the system:

$$\begin{aligned}
node\ 1 &: \frac{d}{dt}(C_1(U_2 - U_1)) + \frac{U_e(t)}{R_0} - \frac{U_1}{R_0} = 0,\\
node\ 2 &: \frac{d}{dt}(C_1(U_1 - U_2)) + \frac{U_b}{R_2} - U_2(\frac{1}{R_1} + \frac{1}{R_2}) + (\alpha - 1)g(U_2 - U_3) = 0,\\
node\ 3 &: -\frac{d}{dt}(C_2U_3) + g(U_2 - U_3) - \frac{U_3}{R_3} = 0,\\
node\ 4 &: -\frac{d}{dt}(C_3(U_4 - U_5)) + \frac{U_b}{R_4} - \frac{U_4}{R_4} - \alpha g(U_2 - U_3) = 0,\\
node\ 5 &: \frac{d}{dt}(C_3(U_4 - U_5)) + \frac{U_b}{R_6} - U_5(\frac{1}{R_5} + \frac{1}{R_6}) + (\alpha - 1)g(U_5 - U_6) = 0,\\
node\ 6 &: -\frac{d}{dt}(C_4U_6) + g(U_5 - U_6) - \frac{U_6}{R_7} = 0,\\
node\ 7 &: -\frac{d}{dt}(C_5(U_7 - U_8)) + \frac{U_b}{R_8} - \frac{U_7}{R_8} - \alpha g(U_5 - U_6) = 0,\\
node\ 8 &: -\frac{d}{dt}(C_5(U_7 - U_8)) + \frac{U_8}{R_9} = 0,
\end{aligned}$$

where

$$g(U_i - U_j) = \beta(e^{\frac{U_i - U_j}{U_F}} - 1)$$

is a simple model of the transistors. The initial signal $U_e(t)$ is

$$U_e(t) = 0.1 \sin(200\pi t).$$

To arrive at the mathematical formulation, one just has to identify U_i with y_i.

9.5.2 Mathematical description of the problem

The problem is of the form

$$G(t, y, y') = 0, \quad y(0) = y_0, y'(0) = y_0',$$

with

$$y \in \mathbf{R}^8, \quad 0 \le t \le 0.2.$$

The function G is defined by

$$G(t, y, y') = My' - f(y),$$

where the matrix M is given by

$$M = \begin{pmatrix} -C_1 & C_1 & 0 & 0 & 0 & 0 & 0 & 0 \\ C_1 & -C_1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -C_2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -C_3 & C_3 & 0 & 0 & 0 \\ 0 & 0 & 0 & C_3 & -C_3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -C_4 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -C_5 & C_5 \\ 0 & 0 & 0 & 0 & 0 & 0 & C_5 & -C_5 \end{pmatrix},$$

and the function f by

$$f(y) = \begin{pmatrix} -\frac{U_e(t)}{R_0} + \frac{y_1}{R_0} \\ -\frac{U_b}{R_2} + y_2(\frac{1}{R_1} + \frac{1}{R_2}) - (\alpha - 1)g(y_2 - y_3) \\ -g(y_2 - y_3) + \frac{y_3}{R_3} \\ -\frac{U_b}{R_4} + \frac{y_4}{R_4} + \alpha g(y_2 - y_3) \\ -\frac{U_b}{R_6} + y_5(\frac{1}{R_5} + \frac{1}{R_6}) - (\alpha - 1)g(y_5 - y_6) \\ -g(y_5 - y_6) + \frac{y_6}{R_7} \\ -\frac{U_b}{R_8} + \frac{y_7}{R_8} + \alpha g(y_5 - y_6) \\ \frac{y_8}{R_9} \end{pmatrix},$$

where g and U_e are auxiliary functions given by $g(x) = \beta(e^{\frac{x}{U_F}} - 1)$ and $U_e(t) = 0.1\sin(200\pi t)$. The values of the technical parameters are:

$$\begin{aligned} U_b &= 6, \\ U_F &= 0.026, \\ \alpha &= 0.99, \\ \beta &= 10^{-6}, \\ R_0 &= 1000, \\ R_k &= 9000 \quad for \quad k = 1, \ldots, 9, \\ C_k &= k \cdot 10^{-6} \quad for \quad k = 1, \ldots, 5. \end{aligned}$$

Consistent initial values at $t = 0$ are

$$\begin{aligned} y_1(0) &= 0, & y_1'(0) &= 51.338775, \\ y_2(0) &= U_b/(\tfrac{R_2}{R_1} + 1), & y_2'(0) &= y_1'(0), \\ y_3(0) &= y_2(0), & y'_3(0) &= -y_2(0)/(C_2 \cdot R_3), \\ y_4(0) &= U_b, & y_4'(0) &= -24.9757667, \\ y_5(0) &= U_b/(\tfrac{R_6}{R_5} + 1), & y_5'(0) &= y_4'(0), \\ y_6(0) &= y_5(0), & y_6'(0) &= y_5(0)/(C_4 \cdot R_7), \\ y_7(0) &= U_b, & y_7'(0) &= -10.00564453, \\ y_8(0) &= 0, & y_8'(0) &= y_7'(0)\ . \end{aligned} \tag{9.5.1}$$

The initial values $y_1'(0), y_4'(0)$ and $y_7'(0)$ were determined numerically.

9.5.3 General information

The problem is a stiff DAE of index 1 consisting of 8 equations and is of the form $My' = f(y)$ with M a matrix of rank 5. P. Rentrop has received it from K. Glashoff and H.J. Oberle and has documented it in [100]. The formulation presented here has been taken from [64].

9.6 Example 6: Car axis problem

9.6.1 Origin of the problem

The problem models the axis of a car as depicted in Figure 9.6. In this model, the left wheel (at the origin $(0,0)$) rolls on a flat surface and the right wheel moves up and down in a sinusoidal way. Denoting the coordinates of the right wheel by $(p(t), q(t))$, we suppose that

$$\begin{aligned} q(t) &= r\sin(\omega t), \\ p(t) &= \sqrt{l^2 - q^2(t)}. \end{aligned}$$

Note that in Figure 9.6 the coordinates $(p(t), q(t))$ are denoted by $(p_1(t), q_1(t))$. This parameterization describes the situation where the right wheel rolls over equidistant hills of height r. The movement of the lower axis between $(0,0)$ and $(p(t), q(t))$ is carried over to the upper axis between (x_2, y_2) and (x_1, y_1) by two massless stiff springs with Hooke's constant $\frac{1}{\varepsilon^2}$ and length l_0. The movement of the mechanism has two constraints. First, the distance between (x_1, y_1) and (x_2, y_2) are obtained by using Lagrangian mechanics. Scaling the Lagrange multipliers by ε^2 yields the 10 equations given in the next section.

9.6.2 Mathematical description of the problem

The problem is of the form

$$\begin{aligned} u' &= v, \\ Kv' &= f(u, \lambda), \\ 0 &= g(u). \end{aligned}$$

The equations are given by

$$\begin{aligned}
\frac{dx_1}{dt} &= \upsilon_1,\\
\frac{dy_1}{dt} &= \upsilon_2,\\
\frac{dx_2}{dt} &= \upsilon_3,\\
\frac{dy_2}{dt} &= \upsilon_4,\\
\varepsilon^2\frac{M}{2}\frac{d\upsilon_1}{dt} &= (l_0-\sqrt{(x_1-p(t))^2+(y_1-q(t))^2})\frac{(x_1-p(t))}{\sqrt{(x_1-p(t))^2+(y_1-q(t))^2}}-2\lambda_2(x_1-x_2),\\
\varepsilon^2\frac{M}{2}\frac{d\upsilon_2}{dt} &= (l_0-\sqrt{(x_1-p(t))^2+(y_1-q(t))^2})\frac{(y_1-q(t))}{\sqrt{(x_1-p(t))^2+(y_1-q(t))^2}}-\varepsilon^2\frac{M}{2}\lambda_2(y_1-y_2),\\
\varepsilon^2\frac{M}{2}\frac{d\upsilon_3}{dt} &= (l_0-\sqrt{x_2^2+y_2^2})\frac{x_2}{\sqrt{x_2^2+y_2^2}}-p(t)\lambda_1+2\lambda_2(x_1-x_2),\\
\varepsilon^2\frac{M}{2}\frac{d\upsilon_4}{dt} &= (l_0-\sqrt{(x_2^2+y_2^2)^2})\frac{y_2}{\sqrt{x_2^2+y_2^2}}-\varepsilon^2\frac{M}{2}-q(t)\lambda_1+2\lambda_2(y_1-y_2),\\
0 &= p(t)x_2+q(t)y_2,\\
0 &= (x_1-x_2)^2+(y_1-y_2)^2-l^2.
\end{aligned}$$

The constants read

$$M=10,\quad \varepsilon=10^{-2},\quad l=1,\quad l_0=0.5.$$

The functions $p(t)$ and $q(t)$ are defined by

$$\begin{aligned}
q(t) &= r\sin(\omega t),\\
p(t) &= \sqrt{l^2-q^2(t)},
\end{aligned}$$

where the constants are given by $r=0.1$ and $\omega=10$. The initial conditions were chosen to be

$x_1=1$	$x_1'=-0.5$
$x_2=0$	$x_2'=-0.5$
$y_1=0.5$	$y_1'=0$
$y_2=0.5$	$y_2'=0$
$\lambda_1=0$	$\lambda_2=0$

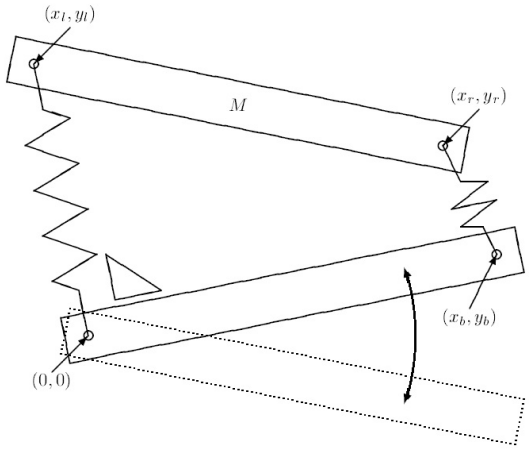

Figure 9.6: Model of the car axis

9.6.3 General information

The problem is a stiff DAE of index 3, consisting of 8 differential and 2 algebraic equations. It has been taken from [109]. Since not all initial conditions were given, a consistent set of initial conditions have been chosen.

9.7 Example 7: NAND gate

9.7.1 Origin of the problem

The NAND gate in Figure 9.7 consists of two n-channel enhancement MOSFETs (ME), one n-channel depletion MOSFET (MD) and two load capacitances C_5 and C_{10}. MOSFETs are special transistors. They have four terminals: the drain, the bulk, the source and the gate. The gate voltages of both enhancement transistors are controlled by two voltage sources V_1 and V_2.

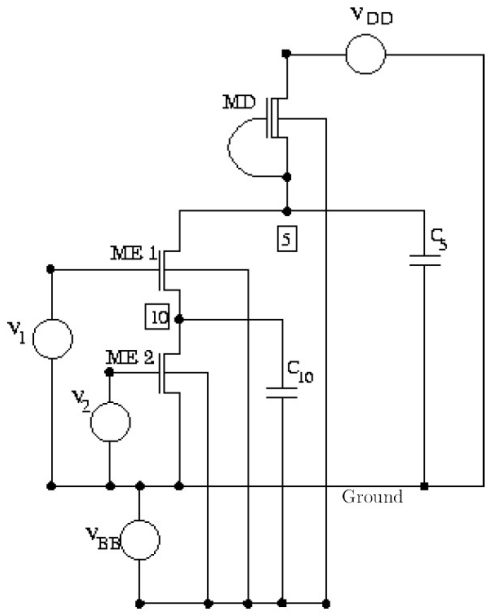

Figure 9.7: Circuit diagram of the NAND gate

		V2	
		LOW	HIGH
V1	LOW	HIGH	HIGH
	HIGH	HIGH	LOW

Figure 9.8: Response of the NAND gate

Depending on the input voltages, the NAND gate generates a response at node 5 as shown in Figure 9.8. If we represent the logical values 1 and 0 by high respectively low voltage levels, we see that the NAND gate executes the Not AND operation. This behavior is easily explained: if V_1 respectively V_2 is low, then the corresponding enhancement transistor locks; the voltage at node 5 is high at $V_{DD} = 5V$ due to MD. If both V_1 and V_2 exceed a given threshold voltage U_T, then a drain current through

both enhancement transistor occurs. The MOSFETs open and the voltage at node 5 breaks down. The response is low.

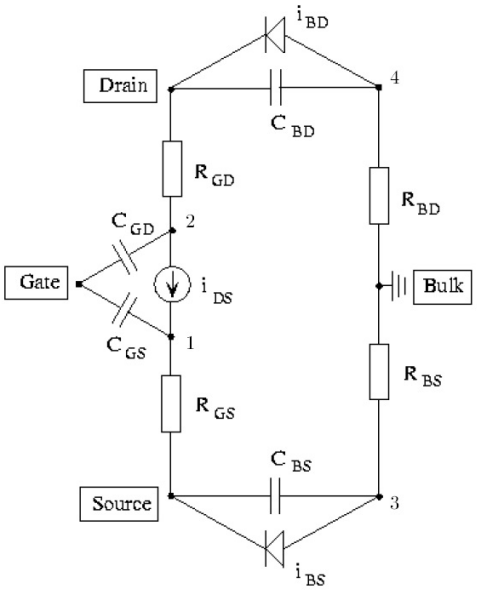

Figure 9.9: Companion model of a MOSFET

In the circuit analysis the three MOSFETs are replaced by the circuit shown in Figure 9.9. Here, the well-known companion model of Shichmann and Hodges [108] is used. The characteristics of the circuit elements can differ depending on the MD or ME case. This circuit has four internal nodes corresponding to the drain, the bulk, the source and the gate. The static behaviour of the transistor is described by the drain current i_{DS}. To include secondary effects, load capacitances like R_{GS}, R_{GD}, R_{BS}, and R_{BD} are introduced. The so-called *pn*-junction between source and bulk is modeled by the diode i_{BS} and the non-linear capacitance C_{BS}. Analogously, i_{BD} and C_{BD} model the *pn*-junction between bulk and diode. Linear gate capacitances C_{GS} and C_{GD} are used to describe the intrinsic charge flow effects roughly. To formulate the circuit equations, we note that the circuit consists of 14 nodes. These 14 nodes are the nodes 5 and 10 and the 12 internal nodes of the three transistors. For every node a variable is introduced that represents the voltage in that node. In terms of these voltages the circuit equations are formulated by using the Kirchoff Current Law (KCL) along with the transistor model shown in Figure

9.9. The differential equations given in the next section result from applying KCL to the following nodes:

equations	nodes
1-4	internal nodes MD-transistor
5	node 5
6-9	internal nodes ME1-transistor
10	node 10
11 14	internal nodes ME2-transistor

9.7.2 Mathematical description of the problem

The problem is of the form:

$$C(y(t))\frac{dy}{dt} = f(t, y(t)), \quad y(0) = y_0,$$

with

$$y \in \mathbf{R}^{14}, \quad 0 \le t \le 80.$$

The equations are given by:

$$\begin{aligned}
C_{GS} \cdot (\dot{y}_5 - \dot{y}_1) &= i^D_{DS}(y_2 - y_1, y_5 - y_1, y_3 - y_5, y_5 - y_2, y_4 - V_{DD}) + \frac{y_1 - y_5}{R_{GS}}, \\
C_{GD} \cdot (\dot{y}_5 - \dot{y}_2) &= -i^D_{DS}(y_2 - y_1, y_5 - y_1, y_3 - y_5, y_5 - y_2, y_4 - V_{DD}) + \frac{y_2 - V_{DD}}{R_{GD}}, \\
C_{BS}(y_3 - y_5) \cdot (\dot{y}_5 - \dot{y}_3) &= -\frac{y_3 - V_{BB}}{R_{BS}} - i^D_{BS}(y_3 - y_5), \\
C_{BD}(y_4 - V_{DD}) \cdot (-\dot{y}_4) &= -\frac{y_4 - V_{BB}}{R_{BD}} + i^D_{BD}(y_4 - V_{DD}),
\end{aligned}$$

$$\begin{aligned}
C_{GS} \cdot \dot{y}_1 + C_{GD} \cdot \dot{y}_2 + C_{BS}(y_3 - y_5) \cdot \dot{y}_3 \\
-(C_{GS} + C_{GD} + C_{BS}(y_3 - y_5) + C_5) \cdot \dot{y}_5 - C_{BD}(y_9 - y_5) \cdot (\dot{y}_5 - \dot{y}_9) = \\
\frac{y_5 - y_1}{R_{GS}} + i^D_{DS}(y_3 - y_5) + \frac{y_5 - y_7}{R_{GD}} + i^E_{BD}(y_9 - y_5),
\end{aligned}$$

$$\begin{aligned}
C_{GS} \cdot \dot{y}_6 &= -i^E_{DS}(y_7 - y_6, V_1(t) - y_6, y_8 - y_{10}, V_1(t) - y_7, y_9 - y_5) + C_{GS} \cdot \dot{V}_1(t) - \frac{y_6 - y_{10}}{R_{GS}}, \\
C_{GD}\dot{y}_7 &= i^E_{DS}(y_7 - y_6, V_1(t) - y_6, y_8 - y_{10}, V_1(t) - y_7, y_9 - y_5) + C_{GD} \cdot \dot{V}_1(t) - \frac{y_7 - y_5}{R_{GD}},
\end{aligned}$$

$$
\begin{aligned}
C_{BS}(y_8 - y_{10}) \cdot (\dot{y}_8 - \dot{y}_{10}) &= -\frac{y_8 - V_{BB}}{R_{BS}} + i^E_{BS}(y_8 - y_{10}), \\
C_{BD}(y_9 - y_5) \cdot (\dot{y}_9 - \dot{y}_5) &= -\frac{y_9 - V_{BB}}{R_{BD}} + i^E_{BD}(y_9 - y_5),
\end{aligned}
$$

$$
\begin{aligned}
C_{BS}(y_8 - y_{10}) \cdot (\dot{y}_8 - \dot{y}_{10}) - C_{BD}(y_{14} - y_{10}) \cdot (\dot{y}_{10} - \dot{y}_{14}) + C_{10} \cdot \dot{y}_{10} = \\
\frac{y_{10} - y_6}{R_{GS}} + i^E_{BS}(y_8 - y_{10}) + \frac{y_{10} - y_{12}}{R_{GD}} + i^E_{BD}(y_{14} - y_{10}),
\end{aligned}
$$

$$
\begin{aligned}
C_{GS} \cdot \dot{y}_{11} = &-i^E_{DS}(y_{12} - y_{11}, V_2(t) - y_{11}, y_{13}, V_2(t) - y_{12}, y_{14} - y_{10}) \\
&+ C_{GS} \cdot \dot{V}_2(t) - \frac{y_{11}}{R_{GS}},
\end{aligned}
$$

$$
\begin{aligned}
C_{GD} \cdot \dot{y}_{12} = &-i^E_{DS}(y_{12} - y_{11}, V_2(t) - y_{11}, y_{13}, V_2(t) - y_{12}, y_{14} - y_{10}) \\
&+ C_{GD} \cdot \dot{V}_2(t) - \frac{y_{12} - y_{10}}{R_{GD}},
\end{aligned}
$$

$$
\begin{aligned}
C_{GS}(y_{13}) \cdot \dot{y}_{13} &= -\frac{y_{13} - V_{BB}}{R_{BS}} + i^E_{BS}(y_{13}), \\
C_{BD}(y_{14} - y_{10})(\dot{y}_{14} - \dot{y}_{10}) &= -\frac{y_{14} - V_{BB}}{R_{BS}} + i^E_{BD}(y_{14} - y_{10}).
\end{aligned}
$$

The functions C_{BD} and C_{BS} read

$$
C_{BD}(U) = C_{BS}(U) = \begin{cases} C_0 \cdot (1 - \frac{U}{\phi_B})^{-\frac{1}{2}} & \text{for} \quad U \leq 0, \\ C_0 \cdot (1 + \frac{U}{2 \cdot \phi_B}) & \text{for} \quad U > 0, \end{cases}
$$

with $C_0 = 0.24 \cdot 10^{-4}$ and $\phi_B = 0.87$.

The functions i^D_{BS} and i^E_{BS} have the same form denoted by i_{BS}. The only difference between them is that the constants used in i_{BS} depend on the superscript D and E. The same holds for the functions $i^{D/E}_{BD}$, $i^{D/E}_{BS}$. The functions i_{BS}, i_{BD} and i_{DS} are defined by

$$
i_{BS}(U_{BS}) = \begin{cases} -i_S \cdot (\exp(\frac{U_{BS}}{U_T}) - 1) & \text{for} \quad U_{BS} \leq 0, \\ 0 & \text{for} \quad U_{BS} > 0, \end{cases}
$$

$$i_{BD}(U_{BD}) = \begin{cases} -i_S \cdot (\exp(\frac{U_{BD}}{U_T}) - 1) & \text{for} \quad U_{BD} \leq 0, \\ 0 & \text{for} \quad U_{BD} > 0, \end{cases}$$

$$i_{DS}(U_{DS}, U_{GS}, U_{BS}, U_{GD}, U_{BD}) = \begin{cases} GSD_+(U_{DS}, U_{GS}, U_{BS}) & \text{for} \quad U_{DS} > 0, \\ 0 & \text{for} \quad U_{DS} = 0, \\ GSD_-(U_{DS}, U_{GD}, U_{BD}) & \text{for} \quad U_{DS} < 0, \end{cases}$$

where $GDS_+(U_{DS}, U_{GS}, U_{BS}) =$

$$\begin{cases} 0 & \text{for} \quad U_{GS} - U_{TE} \leq 0, \\ -\beta \cdot (1 + \delta \cdot U_{DS}) & \text{for} \quad 0 < U_{GS} - U_{TE} \leq U_{DS}, \\ \cdot (U_{GS} - U_{TE})^2 & \\ -\beta \cdot U_{DS} \cdot (1 + \delta \cdot U_{DS}) & \text{for} \quad 0 < U_{DS} < U_{GS} - U_{TE}, \\ \cdot [2 \cdot (U_{GS} - U_{TE}) - U_{DS}]) & \end{cases}$$

with $U_{TE} = U_{T0} + \gamma \cdot (\sqrt{\Phi - U_{BS}} - \sqrt{\Phi})$

$GDS_-(U_{DS}, U_{Gd}, U_{BD}) =$

$$\begin{cases} 0 & \text{for} \quad U_{GD} - U_{TE} \leq 0, \\ \beta \cdot (1 - \delta \cdot U_{DS}) \cdot & \text{for} \quad 0 < U_{GD} - U_{TE} \leq U_{DS}, \\ \cdot (U_{GD} - U_{TE})^2 & \\ -\beta \cdot U_{DS} \cdot (1 - \delta \cdot U_{DS}) \cdot & \text{for} \quad 0 < -U_{DS} < U_{GD} - U_{TE}, \\ \cdot [2 \cdot (U_{GD} - U_{TE}) + U_{DS}] & \end{cases}$$

with $U_{TE} = U_{T0} + \gamma \cdot (\sqrt{\Phi - U_{BD}} - \sqrt{\Phi}$

The constants used in the definition of i_{BS}, i_{BD} and i_{DS} carry a superscript D or E. Using for example the constants with superscript E in the functions i_{BS} yields the function i_{BS}^E. These constants are shown in the following table.

	E	D
i_s	10^{-14}	10^{-14}
U_T	25.85	25.85
U_{T0}	0.2	-2.43
β	$1.748 \cdot 10^{-3}$	$5.35 \cdot 10^{-4}$
γ	0.035	0.2
δ	0.02	0.02
Φ	1.01	1.28

The other are given by

$$\begin{aligned} V_{BB} &= -2.5, \\ V_{DD} &= 5, \\ C_5 = C_{10} &= 0.5 \cdot 10^{-4}, \\ R_{GS} = R_{GD} &= 4, \\ R_{BS} = R_{BD} &= 10, \\ C_{GS} = C_{GD} &= 0.6 \cdot 10^{-4}. \end{aligned}$$

The functions $V_1(t)$ and $V_2(t)$ are

$$V_1(t) = \begin{cases} 20 - tm & \text{if} \quad 15 < tm \leq 20, \\ 5 & \text{if} \quad 10 < tm \leq 15, \\ tm - 5 & \text{if} \quad 5 < tm \leq 10, \\ 0 & \text{if} \quad tm \leq 5, \end{cases}$$

with $tm = t \cdot mod\ 20$ and

$$V_2(t) = \begin{cases} 40 - tm & \text{if} \quad 35 < tm \leq 40, \\ 5 & \text{if} \quad 20 < tm \leq 35, \\ tm - 15 & \text{if} \quad 15 < tm \leq 20, \\ 0 & \text{if} \quad tm \leq 15, \end{cases}$$

with $tm = t \cdot mod\ 40$.

The initial values are given by

$$\begin{aligned} &y_1 = y_2 = y_5 = Y_7 = 5.0, \\ &y_3 = y_4 = y_8 = y_9 = y_{13} = y_{14} = V_{BB} = -2.5, \\ &y_6 = y_{10} = y_{12} = y_7 = 3.62385, \\ &y_{11} = 0. \end{aligned}$$

9.7.3 General information

The problem is a system of 14 stiff implicit ordinary differential equations. It has been contributed by Michael Günther and Peter Rentrop [60] .

Chapter 10

ANNEXE II: Reciprocal Polynomial Extrapolation vs Richardson Extrapolation

Abstract 10.0.1 *The reciprocal polynomial extrapolation was introduced in [2], where its accuracy and stability were studied and a linear scalar test problem was analyzed numerically. In the present work, a new step in the implementation of the reciprocal polynomial extrapolation, ensuring at least the same behavior as the Richardson extrapolation, is proposed. Looking at the reciprocal extrapolation as a Richardson extrapolation where the original data is nonlinearly modified, the improvements that we will obtain should be justified. Several theoretical analysis of the new extrapolation, including local error estimates and stability properties, are presented. A comparison between the two extrapolation techniques is performed for solving some boundary problems with perturbation controlled by a small parameter ϵ. Using two specific boundary problems, the error and the robustness of the new technique using centered divided differences in a uniform mesh are investigated numerically. They turn out to be better than those presented by the Richardson extrapolation. Finally, investigations on the accuracy when using a special non-uniform discretization mesh are presented. A numerical comparison with the Richardson extrapolation for this particular case, where we present some improvements, is also performed.*

10.1 Introduction

In this paper we are interested in the approximation of singularly perturbed boundary value problems of the type:

$$\begin{aligned} \epsilon y'' &= f(t, y, y'), \\ y(a) &= \alpha, \\ y(b) &= \beta. \end{aligned}$$

When we are approximating the exact solution a_0 of this boundary problem by means of a numerical solution $S(h)$ obtained by some finite difference method, the error can be expressed as a Taylor series:

$$a_0 - S(h) = a_1 h^{\gamma_1} + a_2 h^{\gamma_2} + \cdots , \tag{10.1.1}$$

where $\gamma_i \in \mathbb{N}$ and h denotes the discretization step.

For second order central finite differences (see Sections 10.3.1 and 10.3.2) the exponents are well known; in fact $\gamma_i = 2 \cdot i$.

In order to improve the accuracy, we can use extrapolation procedures. The Richardson (or polynomial) extrapolation scheme consists of successive eliminations of the terms of order h^{γ_i}, $i = 1, \ldots, n$ by linear combinations of approximations $S(h)$ for different values of h. It can be viewed as the value at $h = 0$ of the only polynomial

$$P(x; h) = p_0 + p_1 x^{\gamma_1} + p_2 x^{\gamma_2} + \cdots + p_n x^{\gamma_n},$$

interpolating the data $S(h)$ for the parameters considered. A typical choice is

$$\{h, 2h, 2^2 h, \ldots, 2^n h\}.$$

It is well known that the Richardson extrapolation procedure is equivalent to extrapolating by pairs with functions of the type $p(x; h) = a x^{\gamma_i} + b$ (where γ_i is the order of the method at each step). For instance, associated to $S(h)$ and $S(2h)$, we consider the system

$$\begin{aligned} a h^{\gamma_1} + b &= S(h), \\ a (2h)^{\gamma_1} + b &= S(2h). \end{aligned}$$

This system determines the values of a and b, and the approximation obtained by the extrapolation is given by $p(0; h) = b$. It is easy to see that $p(0; h)$ is given by the following linear combination of $S(h)$ and $S(2h)$:

$$p(0; h) = \frac{2^{\gamma_1} S(h) - S(2h)}{2^{\gamma_1} - 1}.$$

By this process, we have achieved a better approximation of a_0 by subtracting the largest term in the error. The process can be repeated to remove more error terms to get even better approximations.

Considering known $S(h)$, $S(2h)$ and $S(4h)$, we can extrapolate by pairs; that is, from $S(h)$ and $S(2h)$ we obtain

$$p(0;h) = \frac{2^{\gamma_1} S(h) - S(2h)}{2^{\gamma_1} - 1},$$

and from $S(2h)$ and $S(4h)$ we obtain

$$p(0;2h) = \frac{2^{\gamma_1} S(2h) - S(4h)}{2^{\gamma_1} - 1}.$$

These two approximations have γ_2 as the exponent in the first term of the error; thus their extrapolation is given by

$$\frac{2^{\gamma_2} p(0;h) - p(0;2h)}{2^{\gamma_2} - 1}.$$

In this way, denoting the initial data by $p_{0,l} = S(2^{l-1}h)$ for $l = 1, 2, 3, \ldots, n$, the algorithm for the implementation of the Richardson extrapolation algorithm is given by:

Algorithm 10.1.1 (The Richardson Extrapolation Algorithm [32])

For $l = 1, 2, \ldots, n$

$$p_{0,l} = S(2^{l-1}h)$$

For $j = 1$ *up to* $r = n - 1$ *and* **for** $l = 1$ *up to* $k = r - j + 1$

$$p_{j,l} = \frac{2^{\gamma_j} p_{j-1,l} - p_{j-1,l+1}}{2^{\gamma_j} - 1}$$

where γ_j *is the first power of the error in each step.*

The Richardson extrapolation algorithm has been used in many contexts; see for instance [29, 75, 77, 88, 89, 121], to mention just some of the contributions.

On the other hand, the reciprocal polynomial extrapolation procedure was introduced in [2] as an alternative to the Richardson extrapolation. The concept is similar to the Richardson extrapolation strategy but considers extrapolations by pairs with functions of the type $\frac{1}{dx^{\gamma_i}+e}$. The order of accuracy is the same in both cases but the reciprocal polynomial extrapolation gives better stability behavior in stiff problems. See [2] and [5] for more details.

In this paper, we propose a new step in the implementation of the reciprocal polynomial extrapolation, in order to obtain at least the same behavior as the Richardson extrapolation. Our aim is to present a numerical comparison with this modified reciprocal polynomial extrapolation. It is important to point out the difficulties with the Richardson extrapolation for improving the original approximation when the discretizations are not small enough. The improvements in robustness, that we will obtain in theses cases, should be understand looking at the reciprocal extrapolation as a Richardson extrapolation in which the original data has been nonlinearly modified. We analyze some local error and stability properties of this extrapolation scheme. We compare the numerical behavior of both techniques, when they are used to increase the order of a given method in the approximation of singularly perturbed boundary value problems. For simplicity we use the central finite difference scheme. Other more sophisticated approximations can be found in the literature; see for instance [67] and the references therein. We first apply the numerical analysis to two particular problems; Viscous Shock [21] and Turning Point [83], extrapolating the second order divided difference scheme associated with a uniform mesh. Finally, uniformity in ϵ is ensured on a specific non-uniform mesh introduced in [123]. We analyze several problems and perform a comparison with the Richardson extrapolation, pointing out some advantages of the new extrapolation procedure. We analyze the error produced by our approach in both cases.

10.2 A new implementation of the RPE

In this section, we introduce a new step in the implementation of the reciprocal polynomial extrapolation. We are interested to obtain at least the same performance as that achieved by the Richardson extrapolation when this extrapolation works well and improve its robustness in other cases.

In the original reciprocal polynomial extrapolation technique [2], first we compute the inverse of the data, then we compute the Richardson extrapolation and finally we compute the inverse of the result.

One improvement proposed in the present section consists on a specific translation of the original data. As in the case of the Richardson extrapolation scheme, we build the reciprocal polynomial extrapolation scheme by pairs; thus we start with two known values of S at different resolutions:

$$\begin{aligned} S(h) &= a_0 + a_1 h^{\gamma_1} + a_2 h^{\gamma_2} + \cdots \\ S(2h) &= a_0 + a_1 2^{\gamma_1} h^{\gamma_1} + a_2 2^{\gamma_2} h^{\gamma_2} + \cdots \end{aligned}$$

Let us consider the translation

$$T_h = sign(M)(1 + |m|),$$

where

$$\begin{cases} M = S(h),\ m = S(2h), & \text{if } |S(2h)| \leq |S(h)|, \\ M = S(2h),\ m = S(h), & \text{otherwise.} \end{cases}$$

The proposed translation T_h has the same sign as the maximum (in absolute value) of the data. The size of T_h is equal to one plus the absolute value of the minimum (in absolute value) of the data.

Taking the above translation T_h into account, we compute

$$\begin{aligned} S_1 &= S(h) + T_h, \\ S_2 &= S(2h) + T_h. \end{aligned}$$

We now consider the rational function $r(x;h) = \dfrac{1}{dx^{\gamma_1} + c}$ that is able to interpolate S_1 and S_2,

$$\begin{aligned} \frac{1}{dh^{\gamma_1} + c} &= S_1, \\ \frac{1}{d2^{\gamma_1}h^{\gamma_1} + c} &= S_2. \end{aligned}$$

This system is equivalent to

$$\begin{aligned} \frac{1}{S_1} &= dh^{\gamma_1} + c, \\ \frac{1}{S_2} &= d2^{\gamma_1}h^{\gamma_1} + c; \end{aligned}$$

that is, the linear system of the Richardson extrapolation scheme for the new data $\frac{1}{S_1}$ and $\frac{1}{S_2}$.

Therefore, using the formula of the Richardson extrapolation,

$$\frac{1}{r(0;h)} = \frac{2^{\gamma_1}\frac{1}{S_1} - \frac{1}{S_2}}{2^{\gamma_1} - 1};$$

and the approximation obtained by the new reciprocal polynomial extrapolation scheme is then given by

$$r(0;h) - T_h = \frac{(2^{\gamma_1} - 1)S_1S_2}{2^{\gamma_1}S_2 - S_1} - T_h.$$

With this new step (translation procedure) any zero division in the inversion of each pair is avoided ($S_i \neq 0$, $i = 1, 2$). Moreover, since $|S_i| > 1$ and therefore $\frac{1}{|S_i|} \in (0, 1)$, $i = 1, 2$, the data interval will be changed from $\mathbb{R} = (-\infty, +\infty)$ in the direct implementation of the Richardson extrapolation to $(0, 1)$ (or $(-1, 0)$) when we use the Richardson extrapolation within the reciprocal polynomial extrapolation mechanism.

The Richardson extrapolation has problems for improving a given approximation when the step discretizations are not small enough, this depends of the characteristics of the problem (stiffness or perturbation parameters). In these cases the points $S_i \neq 0$, $i = 1, 2$ should be not close enough. Changing the data interval through the proposed translation ensures that the extrapolated points when using the Richardson extrapolation procedure within the reciprocal polynomial extrapolation scheme are close. In the numerical experiments we analyze the improvements obtained by this fact. Now, we include an artificial situation as a motivation of the new extrapolation scheme.

Rich. Ext.	**Rec. Poly. Ext.**
5.1×10^{-3}	5.0×10^{-3}

Table 10.1: $\gamma_1 = 2$, $S(h) = 10^{-2}$, $S(2h) = 2.5\ 10^{-2}$ and $h = \frac{1}{8}$

Rich. Ext.	**Rec. Poly. Ext.**
-2.0×10^{-2}	1.0×10^{-4}

Table 10.2: $\gamma_1 = 2$, $S(h) = 10^{-2}$, $S(2h) = 10^{-1}$ and $h = \frac{1}{8}$

In tables 10.1 and 10.2, we analyze two possible situations (they are not associated to any particular problem). In both cases, we consider two approximations $S(h)$ and $S(2h)$ and we compute both the Richardson and the reciprocal polynomial extrapolations. In table 10.1 the points to be extrapolated are close together and in table 10.2 the points are farther apart. In both cases the values of $S(h)$ and $S(2h)$ are positive and we expect to obtain, via the extrapolation techniques, an approximation to the exact solution, which is also positive. In the first case, the two extrapolations obtain similar (reasonable) results but in the second case the Richardson extrapolation predicts negative values which are nonsense.

In the next subsection we analyze the error produced by this process. We will achieve a better approximation of a_0 by eliminating the largest term of the error in

the original method. As in the case of the Richardson extrapolation, this process can be repeated to remove more error terms to get even better approximations. Other theoretical properties, including stability remarks, are also presented.

10.2.1 Theoretical properties for the new reciprocal polynomial extrapolation

In this section we analyze theoretically the new extrapolation technique.

Local error

In this first result we analyze the error produced by the new reciprocal polynomial extrapolation.

Proposition 10.2.1 *The first term of the error*

$$(r(0;h) - T_h) - a_0,$$

produced by the new reciprocal polynomial extrapolation has the following expression:

$$\frac{{a_1}^2(1-2^{2\gamma_1})h^{2\gamma_1} + (2^{\gamma_2} - 2^{\gamma_1})\tilde{a}_0 a_2 h^{\gamma_2}}{\tilde{a}_0(1-2^{\gamma_1})}, \tag{10.2.1}$$

where $\tilde{a}_0 = a_0 + T_h$.

Proof 10.2.2 *Since*

$$(r(0;h) - T_h) - a_0 = r(0;h) - (a_0 + T_h) = r(0;h) - \tilde{a}_0, \tag{10.2.2}$$

$r(0;h)$ *will be the approximation of* $\tilde{a}_0$.

From

$$\begin{aligned} \frac{1}{dh^{\gamma_1} + c} &= \tilde{a}_0 + a_1 h^{\gamma_1} + a_2 h^{\gamma_2} + \cdots \\ \frac{\frac{1}{c}}{\frac{d}{c}h^{\gamma_1} + 1} &= \tilde{a}_0 + a_1 h^{\gamma_1} + a_2 h^{\gamma_2} + \cdots, \end{aligned}$$

we have that

$$r(0;h) - \tilde{a}_0 = \frac{d}{c}h^{\gamma_1}\tilde{a}_0 + a_1 h^{\gamma_1} + \frac{d}{c}a_1 h^{2\gamma_1} + a_2 h^{\gamma_2} + \cdots$$

Taking into account that $h \to 0$ *we should compute:*

$$\frac{d}{c}h^{\gamma_1}\tilde{a}_0 + a_1h^{\gamma_1} + \frac{d}{c}a_1h^{2\gamma_1} + a_2h^{\gamma_2}.$$

Indeed,

$$\begin{aligned}
&\frac{d}{c}h^{\gamma_1}\tilde{a}_0 + a_1h^{\gamma_1} + \frac{d}{c}a_1h^{2\gamma_1} + a_2h^{\gamma_2} = h^{\gamma_1}\left(\frac{d}{c}(\tilde{a}_0 + a_1h^{\gamma_1})\right) + a_1h^{\gamma_1} + a_2h^{\gamma_2} \\
= \quad & h^{\gamma_1}\left(\frac{\dfrac{S_2 - S_1}{h^{\gamma_1}S_1S_2(1-2^{\gamma_1})}}{\dfrac{S_1 - 2^{\gamma_1}S_2}{S_1S_2(1-2^{\gamma_1})}}(\tilde{a}_0 + a_1h^{\gamma_1})\right) + a_1h^{\gamma_1} + a_2h^{\gamma_2} \\
= \quad & \frac{S_2 - S_1}{S_1 - 2^{\gamma_1}S_2}(\tilde{a}_0 + a_1h^{\gamma_1}) + a_1h^{\gamma_1} + a_2h^{\gamma_2} \\
= \quad & \frac{S_2\left(\tilde{a}_0 + (1-2^{\gamma_1})\,a_1h^{\gamma_1}\right) - S_1\tilde{a}_0}{S_1 - 2^{\gamma_1}S_2} + a_2h^{\gamma_2}.
\end{aligned}$$

Since

$$\begin{aligned}
S_1 &\simeq \tilde{a}_0 + a_1h^{\gamma_1} + a_2h^{\gamma_2}, \\
S_2 &\simeq \tilde{a}_0 + a_12^{\gamma_1}h^{\gamma_1} + a_22^{\gamma_2}h^{\gamma_2},
\end{aligned}$$

expanding the numerator,

$$\begin{aligned}
&S_2\left(\tilde{a}_0 + (1-2^{\gamma_1})\,a_1h^{\gamma_1}\right) - S_1\tilde{a}_0 = \\
&\quad = \left[\tilde{a}_0 + a_12^{\gamma_1}h^{\gamma_1} + a_22^{\gamma_2}h^{\gamma_2}\right]\left(\tilde{a}_0 + (1-2^{\gamma_1})\,a_1h^{\gamma_1}\right) - \left[\tilde{a}_0 + a_1h^{\gamma_1} + a_2h^{\gamma_2}\right]\tilde{a}_0 \\
&\quad = a_1^2(2^{\gamma_1} - 2^{2\gamma_1})h^{2\gamma_1} + \tilde{a}_0a_22^{\gamma_2}h^{\gamma_2} + a_1a_22^{\gamma_2}\,(1-2^{\gamma_1})\,h^{\gamma_1+\gamma_2} - \tilde{a}_0a_2h^{\gamma_2} \\
&\quad \simeq a_1^2(2^{\gamma_1} - 2^{2\gamma_1})h^{2\gamma_1} + \tilde{a}_0a_2(2^{\gamma_2} - 1)h^{\gamma_2}.
\end{aligned}$$

Now expanding the denominator,

$$\begin{aligned}
S_1 - 2^{\gamma_1}S_2 &= \tilde{a}_0 + a_1h^{\gamma_1} - 2^{\gamma_1}(\tilde{a}_0 + a_12^{\gamma_1}h^{\gamma_1}) \\
&= \tilde{a}_0(1-2^{\gamma_1}) + a_1(1-2^{2\gamma_1})h^{\gamma_1} \\
&\simeq \tilde{a}_0(1-2^{\gamma_1}).
\end{aligned}$$

Simplifying, we obtain the error term in (10.2.1):

$$\begin{aligned}
&\frac{S_2\left(\tilde{a}_0 + (1-2^{\gamma_1})\,a_1h^{\gamma_1}\right) - S_1\tilde{a}_0}{S_1 - 2^{\gamma_1}S_2} + a_2h^{\gamma_2} \simeq \\
&\frac{{a_1}^2(1-2^{2\gamma_1})h^{2\gamma_1} + \tilde{a}_0a_2(2^{\gamma_2} - 1)h^{\gamma_2}}{\tilde{a}_0(1-2^{\gamma_1}) + a_1(1-2^{2\gamma_1})h^{\gamma_1}} + a_2h^{\gamma_2} \simeq \\
&\frac{{a_1}^2(1-2^{2\gamma_1})h^{2\gamma_1} + (2^{\gamma_2} - 2^{\gamma_1})\tilde{a}_0a_2h^{\gamma_2}}{\tilde{a}_0(1-2^{\gamma_1})}.
\end{aligned}$$

Notice that in the state of the art methods, $2\gamma_1 \geq \gamma_2$. For the second order divided difference scheme, as we see in next section, $\gamma_{i+1} = 2\gamma_i$.

Reciprocal polynomial extrapolation as a modification of Richardson extrapolation

Let us consider $S(h)$ and $S(2h)$ approximations of a_0. With the notations used in the definition of the reciprocal polynomial extrapolation procedure the polynomial $\frac{1}{r(0;h)}$ is the approximation obtained by the Richardson extrapolation in order to approximate $\frac{1}{a_0+T_h}$.

In particular, we can use the theoretical properties derived first for the Richardson extrapolation. Indeed, from

$$|r(0;h) - T_h - a_0| = |r(0;h)(a_0 + T_h)| \; |\frac{1}{a_0 + T_h} - \frac{1}{r(0;h)}|,$$

and since the last factor is the error of the Richardson extrapolation for the data $\frac{1}{S_1}$ and $\frac{1}{S_2}$, then we can bound the error of the reciprocal polynomial extrapolation.

On the other hand, for the extrapolation of more than two points, we propose the use of the introduced procedure in a recursive way as the original Richardson extrapolation works. Using the result for two points we can prove the error bounds for this general case. In particular, each step of the reciprocal polynomial extrapolation achieves a better approximation of the solution a_0 by eliminating the largest term in the error of the extrapolated scheme S.

Stability properties

Let us consider two approximations $S(h)$ and $S(2h)$, of order γ, to the solution a_0 of a given problem. Without loss of generality, we suppose that $|S(2h)| > |S(h)|$. Let us denote some perturbation of them by $\tilde{S}(h)$ and $\tilde{S}(2h)$. Assume that the approximations have been perturbed maintaining their relative size and the sign of the biggest one, that is, $|\tilde{S}(2h)| > |\tilde{S}(h)|$ and $S(2h)\tilde{S}(2h) \geq 0$.

In this subsection, we are interesting to bound the different of the values obtained by the reciprocal polynomial extrapolation for both pairs $(S(h), S(2h))$ and $(\tilde{S}(h), \tilde{S}(2h))$.

First,

$$|(1 + T_h) - (1 + \tilde{T_h})| := |(1 + |S(h)|) - (1 - \tilde{S}(h))| \leq |S(h) - \tilde{S}(h)|.$$

Secondly, let us consider the function

$$F(x,y) = \frac{(2^\gamma - 1)xy}{2^\gamma y - x},$$

where $|y| > |x|$ and $xy \geq 0$.

After some calculus we obtain

$$\begin{aligned} \frac{\partial F}{\partial x} &= \frac{(2^\gamma - 1)2^\gamma y^2}{(2^\gamma y - x)^2}, \\ \frac{\partial F}{\partial y} &= -\frac{(2^\gamma - 1)^2 x^2}{(2^\gamma y - x)^2}, \end{aligned}$$

and then

$$||\nabla F(x,y)||_\infty \leq 2\frac{2^\gamma + 1}{2^\gamma - 1}.$$

Finally, using the above estimates and the definition of the reciprocal polynomial extrapolation scheme we obtain

$$\begin{aligned} |(\frac{(2^\gamma - 1)S_1S_2}{2^\gamma S_2 - S_1} - T_h) - (\frac{(2^\gamma - 1)\tilde{S}_1\tilde{S}_2}{2^\gamma \tilde{S}_2 - \tilde{S}_1} - \tilde{T}_h)| &\leq 2\frac{2^\gamma + 1}{2^\gamma - 1}\max\{|S_1 - \tilde{S}_1|, |S_2 - \tilde{S}_2|\} \\ &\quad + |S(h) - \tilde{S}(h)| \\ &\leq C\max\{|S(h) - \tilde{S}(h)|, |S(2h) - \tilde{S}(2h)|\}, \end{aligned}$$

for $C > 0$ and the stability is derived.

Analytical interpretation of the reciprocal polynomial extrapolation scheme

In connection with the theoretical studies of the extrapolation schemes, we have found it is very useful to relate the given approximations to some suitable function. This kind of an approach is obviously of greater generality than that dealing with sequences alone.

In the definition of the reciprocal polynomial extrapolation procedure, we have considered rational functions of the type

$$\frac{1}{dx^\gamma + c} - e,$$

where e is given by the translation value and (c,d) are such that $dx^\gamma + c$ interpolates the points $(h, 1/S_1)$ and $(2h, 1/S_2)$.

In particular, we can use the error bounds of rational interpolations [32].

10.3 Singular perturbed boundary problems

In this section, we perform a comparison between both extrapolations when they are used in the approximation of singular perturbed boundary problems using finite difference schemes. We consider several problems that appear in the literature. We discretize the problems using uniform and nonuniform meshes.

10.3.1 Uniform mesh: local error and numerical experiments

We start with the problem

$$\begin{aligned} \epsilon y'' &= f(t, y, y'), \\ y(a) &= \alpha, \\ y(b) &= \beta. \end{aligned}$$

To solve numerically this problem by some finite difference method, one has to overcome difficulties caused by the small parameter ϵ. If one is interested in the numerical solution around all the interval, then the uniform mesh will become the most efficient discretization.

We can approximate $y''(x_i)$ using the Taylor expansion

$$y(x_{i+1}) - 2y(x_i) + y(x_{i-1}) = y''(x_i)h^2 + \sum_{k=2}^{+\infty} C_k h^{2k},$$

so that we obtain the well known second order approximation of $y''(x_i)$

$$\left| \frac{y(x_{i+1}) - 2y(x_i) + y(x_{i-1})}{h^2} - y''(x_i) \right| \leq \sum_{k=2}^{+\infty} |C_k| h^{2k-2}.$$

Similarly we can approximate $y'(x_i)$ again using Taylor's expansion, obtaining as an approximation of $y'(x_i)$ the following expression:

$$\left| \frac{y(x_{i+1}) - y(x_{i-1})}{2h} - y'(x_i) \right| \leq \sum_{k=2}^{+\infty} |D_k| h^{2k-2}.$$

Using the proposed approximations, and denoting $y_i \approx y(x_i)$, we obtain the following nonlinear system of equations:

$$\begin{aligned} y_0 = y(a) = \alpha, \\ \epsilon \frac{y_{i+1} - 2y_i + y_{i-1}}{h^2} = f(x_i, y_i, \frac{y_{i+1} - y_{i-1}}{2h}), \\ y_{m+1} = y(b) = \beta, \end{aligned}$$

for $i = 1, 2, \ldots, m$, with an error of the type

$$\sum_{k=2}^{+\infty} |E_k| h^{2k-2}.$$

Thus, from Proposition 10.2.1, each step of the reciprocal polynomial extrapolation eliminates the largest term in the error. In particular, we obtain the following corollary

Corollary 10.3.1 *The error obtained combining the above numerical difference scheme and n steps of both Richardson and reciprocal polynomial extrapolation schemes admit developments of the form*

$$\sum_{k=n+2}^{+\infty} |\tilde{E}_k| h^{2k-2},$$

assuming that the discretization step is small enough.

Of course, the coefficients $\tilde{E}_k$ are different for both extrapolations.

As we see in next section, there are singular problems with small parameters ϵ where we should consider theoretically also very small values of h. This should be a problem in finite precision arithmetic since the difference schemes should be dominated by roundoff. In this type of situations our approach improves the classical one.

Numerical experiments

In this section we consider two particular problems called Viscous Shock [21] and Turning Point [83]. For a given h, we consider the extrapolation of the two approximations obtained using the second order divided differences scheme for h and $2h$. We present the error (we know the exact solution of both problems) for different discretization and perturbation parameters (h and ϵ).

Viscous Shock Problem: We start with the problem

$$\begin{aligned} \epsilon y'' + 2xy' &= 0, \\ y(-1) &= -1, \\ y(1) &= 1, \end{aligned}$$

which has the exact solution

$$y(x) = \frac{1}{erf(\frac{1}{\sqrt{\epsilon}})} erf(\frac{x}{\sqrt{\epsilon}}),$$

where $errf(z) = \frac{2}{\sqrt{\pi}} \int_0^z e^{t^2} dt$.

Parameters	**Finite Differences**	**Rich. Ext.**	**Rec. Poly. Ext.**
$h = 0.02$, $\epsilon = 10^{-2}$	5.029×10^{-3}	1.937×10^{-4}	1.320×10^{-4}
$h = 0.01$, $\epsilon = 10^{-2}$	1.249×10^{-3}	1.216×10^{-5}	8.918×10^{-6}
$h = 0.01$, $\epsilon = 10^{-4}$	1.572×10^{-1}	2.145×10^{-1}	1.477×10^{-1}
$h = 0.001$, $\epsilon = 10^{-4}$	3.229×10^{-2}	9.367×10^{-3}	5.966×10^{-3}

Table 10.3: Errors for the Viscous Shock Problem.

In this first example, both extrapolation schemes improve the initial approximation of the finite difference method. The final error is smaller in our nonlinear case.

Turning Point Problem We consider the problem

$$\begin{aligned} \epsilon y'' - xy &= 0, \\ y(-1) &= 1, \\ y(1) &= 1. \end{aligned}$$

The real solution of this problem is the linear combination of the Airy functions[1]

$$y(x) = c_1 A_i(\frac{x}{\sqrt[3]{\epsilon}}) + c_2 B_i(\frac{x}{\sqrt[3]{\epsilon}}). \tag{10.3.1}$$

Parameters	**Finite Differences**	**Rich. Ext.**	**Rec. Poly. Ext.**
$h = 0.02$, $\epsilon = 10^{-2}$	7.355×10^{-3}	1.049×10^{-4}	1.114×10^{-4}
$h = 0.01$, $\epsilon = 10^{-2}$	1.843×10^{-3}	6.644×10^{-6}	6.751×10^{-6}
$h = 0.01$, $\epsilon = 10^{-3}$	9.380×10^{0}	2.300×10^{1}	6.591×10^{-1}
$h = 0.005$, $\epsilon = 10^{-3}$	1.513×10^{0}	1.109×10^{0}	3.804×10^{-2}

Table 10.4: Errors for the Turning Point Problem.

In this second example, the reciprocal polynomial extrapolation gives the best results (see Table 10.4). Moreover, for this stiff problem (taking ϵ sufficiently small)

[1]The Airy functions are the two solutions of the Stoker differential equation $y'' - xy = 0$ which are related to the Bessel functions.

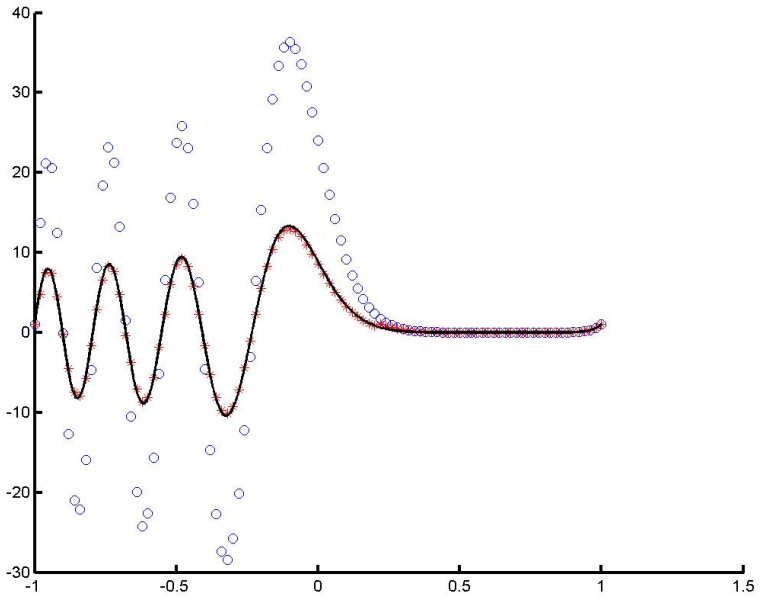

Figure 10.1: Approximations using the Richardson extrapolation 'o' and the reciprocal polynomial extrapolation '*' for the Turning Point Problem. The solid line represents the exact solution. Parameters: $h = 0.01$ and $\epsilon = 10^{-3}$.

the Richardson extrapolation presents some problems when the discretization is not small enough (in comparison with the stiffness). This fact is clearer in Figure 10.1 ($h = 0.01$ and $\epsilon = 10^{-3}$) where only the reciprocal polynomial extrapolation gives a good approximation. In particular, we improve the robustness of the Richardson extrapolation.

10.3.2 Non-uniform mesh and accuracy uniform in ϵ

In [123], the Richardson extrapolation was applied to the central finite difference scheme. The authors point out problems where the need for numerical solution within the boundary layers is important. This implies the use of a non-uniform mesh which is dense in the boundary layer regions. High accuracy uniform in ϵ was proved for the central finite difference scheme on a particular non-uniform mesh.

In this approach, the mesh points for the generic interval $I = [0, 1]$ were given by

$$x_i = \lambda(t_i),\ t_i = ih,\ i = 0, \dots m,\ h = 1/m,\ m \in \mathbb{N},$$

where

$$\lambda(t) = \begin{cases} \theta\epsilon \frac{t}{(1/2 + \sqrt[\tau]{\theta\epsilon} - t)^\tau}, & t \in [0, 1/2] \\ 1 - \lambda(1 - t), & t \in [1/2, 1]. \end{cases}$$

Here, $\theta > 0$ and $\tau > 0$ are some constants independent of ϵ. The mesh points are taken to be symmetric to the middle point of I. The mesh is dense in the boundary layers. The density is increased when θ is decreased. The same is true for τ as long as $\theta 2^\tau \epsilon < 1$. The location of the m mesh points $x_i = \lambda(t_i)$ depends on ϵ through the nonlinear function $\lambda(t)$.

Associated to this type of mesh, the error of the central finite difference scheme at each point x_i admits the following development

$$r_i(u_\epsilon) = \sum_{j=1}^{K} a_j h^{2j} + O(h^k), \quad i = 1, 2, \dots, m-1,$$

where $K = [\frac{k-1}{2}]$ and some smoothness in the solution is assumed, namely $u_\epsilon \in C^{k+4}(I)$.

Note that h is independent of ϵ; the dependence is in the non-uniform mesh computed from $\lambda(t)$. See [123] for more details.

Combining this result and the in Proposition 10.2.1, proves that each step of the reciprocal polynomial extrapolation subtracts the largest term in the error.

Numerical experiments

In this section, we consider two examples. The first one is the example analyzed in [123] and the second one a modification increasing the oscillation of the exact solution. As in Section 10.3.1, for a given h we consider the extrapolation of the two approximations obtained by using the second order divided difference scheme for h and $2h$. We present the error (we know the exact solution of both problems) for different discretization and perturbation parameters (h and ϵ). In this section, we use the non-uniform mesh defined by the function $\lambda(t)$, considering different values of the new parameters θ and τ.

We start with the following problem introduced in [40] and analyzed in [123]

$$\begin{aligned} -\epsilon^2 y''(x) + y(x) &= -\cos^2(\pi x) - 2(\epsilon\pi)^2 \cos(2\pi x), \\ y(0) &= y(1) = 0, \end{aligned}$$

with exact solution

$$y(x) = (\exp(-x/\epsilon) + \exp(-(1-x)/\epsilon))/(1+\exp(-1/\epsilon)) - \cos^2(\pi x).$$

In Tables 10.5 and 10.6 we observe that both extrapolations provide good results and improve the original divided difference scheme. This example was studied for Vulanović et al. in [123] to point out the good behavior of the Richardson extrapolation for this type of problems.

Parameters	**Finite Differences**	**Rich. Ext.**	**Rec. Poly. Ext.**
$\epsilon = 10^{-3}$	3.75×10^{-3}	1.38×10^{-4}	1.40×10^{-4}
$\epsilon = 10^{-6}$	5.09×10^{-3}	2.46×10^{-4}	2.48×10^{-4}
$\epsilon = 10^{-9}$	5.34×10^{-3}	2.25×10^{-4}	2.26×10^{-4}
$\epsilon = 10^{-12}$	5.36×10^{-3}	2.23×10^{-4}	2.23×10^{-4}

Table 10.5: Error for Vulanović et al. problem, $m = 40$, $\theta = 1$ and $\tau = 3$.

Finally, we analyze a modification of the last example with more oscillations in the exact solution. Indeed, we consider

$$\begin{aligned} -\epsilon^2 y''(x) + y(x) &= -\frac{1}{2} + (-\frac{1}{2} - 50(\epsilon\pi)^2) \cos(10\pi x), \\ y(0) &= y(1) = 0, \end{aligned}$$

with exact solution

$$y(x) = (\exp(-x/\epsilon) + \exp(-(1-x)/\epsilon))/(1+\exp(-1/\epsilon)) - \cos^2(5\pi x).$$

Parameters	Finite Differences	Rich. Ext.	Rec. Poly. Ext.
$\epsilon = 10^{-3}$	6.92×10^{-3}	2.96×10^{-4}	2.97×10^{-4}
$\epsilon = 10^{-6}$	7.71×10^{-3}	7.63×10^{-4}	7.70×10^{-4}
$\epsilon = 10^{-9}$	7.76×10^{-3}	7.11×10^{-4}	7.15×10^{-4}
$\epsilon = 10^{-12}$	7.77×10^{-3}	7.04×10^{-4}	7.08×10^{-4}

Table 10.6: Error for the Vulanović et al. problem, $m = 40$, $\theta = 0.1$ and $\tau = 3$.

In Figure 10.2 we observe the improvements obtained by the reciprocal polynomial extrapolation. In this problem, the discretization should increase its density throughout the interval (θ big) in order to approximate well the boundary layers when ϵ decreases. However, as in the second example for the uniform case, taking ϵ sufficiently small, we can improve the results given by the Richardson extrapolation when the discretization is not small enough (in comparison with the stiffness).

10.4 Conclusion

In this paper we have studied a nonlinear extrapolation technique for singularly perturbed boundary value problems. We have introduced a new step in the implementation of the reciprocal polynomial extrapolation obtaining the same behavior as the Richardson extrapolation when this extrapolation works well. Moreover, this nonlinear treatment of the data introduces advantages when the discretization step is not small enough in comparison with the stiffness of the problem. In particular, we improve the robustness of the Richardson extrapolation.

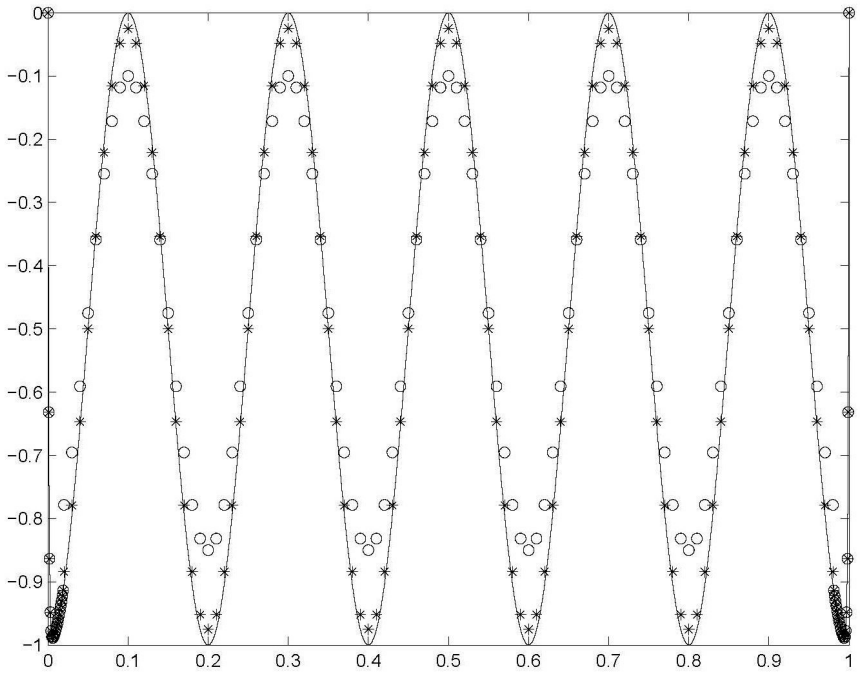

Figure 10.2: Approximations via the Richardson extrapolation 'o' and the reciprocal polynomial extrapolation '*' for the modification of the Vulanović et al. Problem. The solid line represents the exact solution. Parameters: $\theta = 100$, $\tau = 1$, $m = 500$ and $\epsilon = 10^{-3}$.

Bibliography

[1] Amat S. and Busquier S., Third-order iterative methods under Kantorovich conditions. J. Math. Anal. Appl., 336(1), (2007), 243–261.

[2] Amat S., Busquier S. and Candela V., Reciprocal Polynomial Extrapolation, Journal Computational Mathematics, 22(1), (2004), 1-10.

[3] Amat S., Busquier S. and Gutiérrez J.M., An adaptive version of a fourth order iterative method for quadratic equations. J. Comput. Appl. Math., 191(2) (2006), 259–268.

[4] Amat S., Busquier S. and Gutiérrez J.M., Geometric constructions of iterative functions to solve nonlinear equations. J. Comput. Appl. Math. 157(1) (2003), 197–205.

[5] Amat S., Busquier S., Legaz M.J. and Manzano F., Reciprocal Polynomial Extrapolation vs Richardson Extrapolation for Singular Perturbed Boundary Problems, to appear in Numerical Algorithms.

[6] Amat S., Busquier S., Bermúdez C., Legaz M.J. and Plaza S., On a family of high order iterative methods under Kantorovich conditions and some applications, to appear in Abstract and Applied Analysis.

[7] Amat S., Ezquerro J.A. and Hernández M.A., Approximation of inverse operators by a new family of high-order iterative methods, submitted 2011.

[8] Amat S., Hernández M.A. and Romero N., A modified Chebyshev's iterative method with at least sixth order of convergence. Appl. Math. Comput., 206(1) (2008), 164–174.

[9] Amat S., Hernández M.A. and Romero N., Semilocal convergence of a sixth order iterative method for Riccati's equations, to appear in Applied Numerical Mathematics.

[10] Amat S., Legaz M.J. and Pedregal P., On a Newton-type method for Differential-Algebraic Equations, to appear Journal of Applied Mathematics.

[11] Amat S., Legaz M.J. and Pedregal P., Linearizing Stiff Delay Differential Equations, to appear Applied Mathematics and Information Sciences.

[12] Amat S., López D.J. and Pedregal P., Numerical approximation to ODEs using a variational approach I: The basic framework, to appear in Optimization.

[13] Amat S. and Pedregal P., A variational approach to implicit ODEs and differential inclusions, ESAIM-COCV, 15(1), (2009), 139-148.

[14] Amat S. and Pedregal P., A constructive existence theorem for DAEs, in preparation.

[15] Amat S. and Pedregal P., On an alternative approach for the analysis and numerical simulation of stiff ODEs, to appear in DCDS serie A.

[16] Argyros I.K. Improving the order and rates of convergence for the super-Halley method in Banach spaces. Korean J. Comput. Appl. Math. 5(2) (1998), 465–474.

[17] Argyros I.K. The convergence of a Halley-Chebyshev-type method under Newton-Kantorovich hypotheses. Appl. Math. Lett. 6(5) (1993), 71–74.

[18] Argyros I.K. and Hilout S., Weaker conditions for the convergence of Newton's method. J. Complexity 28(3), (2012), 364-387.

[19] Argyros I.K. and Hilout S., Improved local convergence of Newton's method under weak majorant condition. J. Comput. Appl. Math. 236(7), (2012), 1892-1902.

[20] Argyros I.K. and Hilout S., Weak convergence conditions for inexact Newton-type methods. Appl. Math. Comput. 218(6), (2011), 2800-2809.

[21] Ascher U.M., Mattheij R.M.M. and Russell R.D., Numerical Solution of Boundary Value Problems for Ordinary Differential Equations. SIAM Classics in Applied Mathematics 1995.

[22] Auzinger W., Frank R. and Kirlinger G. An extension of B-convergence for Runge-Kutta methods. Appl. Num. Math. 9, (1992), 91-109.

[23] Auzinger W., Frank R. and Kirlinger G. Modern convergence theory for stiff initial value problems. J. Comput. Appl. Math. 45(1-2), (1993), 5-16.

[24] Baker C.T.H., Paul C.A.H. and Willé D.R.: Issues in the numerical solution of evolutionary delay differential equations. Adv. Comput. Math. 3, (1995), 171-196.

[25] Bellen A. and Zennaro M.: Numerical Methods for Delay Differential Equations. Oxford University Press, Oxford (2003).

[26] Berger T., Ilchmann A., On the standard canonical form of time-varying linear DAEs, to appear in Quarterly of Applied Mathematics (2012).

[27] Brenan K.E., Campbell S.L. and Petzold L.R., Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations, SIAM, Philadelphia, 1996.

[28] Brenan K.E. and Petzold L.R., The numerical solution of higher index differential/algebraic equations by implicit methods. SIAM J. Numer. Anal. 26(4), (1989), 976-996.

[29] Bruder J., Strehmel K. and Weiner R., Partitioned adaptive Runge-Kutta methods for the solution of nonstiff and stiff systems. Numer. Math. 52(6), (1988), 621–638.

[30] Brugnano L., Iavernaro F. and Trigiante D., On the existence of energy preserving symplectic integrators based upon Gauss collocation formulae. Math. N.A., 2010.

[31] Brugnano L., Iavernaro F. and Trigiante D., A two step, fourth order, nearly-linear method with energy preserving properties. Math. N. A., 2011.

[32] Bulirsch R. and Stoer J., Introduction to Numerical Analysis, Springer, 2002.

[33] Burrage K. and Butcher J.C., Stability criteria for implicit Runge-Kutta methods, SIAM J. Numer. Anal. 16, (1979), 46-57.

[34] Byrne G.D. and Hindmarsh A.C., Stift ODE Solvers: A Review of Current and Coming Attractions. J. Comput. Phys. 70, (1987), 1-62.

[35] Campbell S.L. and Griepentrog E., Solvability of general differential algebraic equations. SIAM J. Sci. Comput. 16(2), (1995), 257-270.

[36] Crouzeix M., Sur la B-stabilit des methods de Runge-Kutta, Numer. Math. 32, (1979), 75-82.

[37] Dacorogna B. Direct methods in the Calculus of Variations, Springer, 2008 (second edition).

[38] Dahlquist G., A special stability problem for linear multistep methods. BIT, 3, (1963), 27-43.

[39] Dehghan M. and Hajarian M., On derivative free cubic convergence iterative methods for solving nonlinear equations. Comput. Math. Math. Phys. 51(4), (2011), 513-519

[40] Doolan E.P., Miller J. and Schilders W., Uniform Numerical Methods for Problems with Initial and Boundary Layers, Boole Press, Dublin, 1980.

[41] Džunic J., Petkovic M.S. and Petkovic L.D., Three-point methods with and without memory for solving nonlinear equations. Appl. Math. Comput. 218(9), (2012), 4917-4927.

[42] Emmrich E. and Mehrmann W., Analysis of operator differential-algebraic equations arising in fluid dynamics. Part I. The finite dimensional case. (2010).

[43] Enright W.H. and Hayashi H., A delay differential equation solver based on a continuous Runge-Kutta method with defect control. Numer. Algorithms 16, (1997), 349364.

[44] Ezquerro J.A., Grau-Sánchez M., Grau A., Hernández M.A., Noguera M. and Romero N. On iterative methods with accelerated convergence for solving systems of nonlinear equations. J. Optim. Theory Appl. 151(1), (2011), 163-174.

[45] Ezquerro J.A. and Hernández M.A., New Kantorovich-type conditions for Halley's method. Appl. Numer. Anal. Comput. Math. 2(1), (2005), 70–77.

[46] Ezquerro J.A. and Hernández M.A. An improvement of the region of accessibility of Chebyshev's method from Newton's method. Math. Comp., 78(267) (2009), 1613–1627.

[47] Fang L., A cubically convergent iterative method for solving nonlinear equations. Adv. Appl. Math. Sci. 10(2), (2011), 117-119.

[48] Farhat C., Tezaur R. and Djellouli R., On the solution of three-dimensional inverse obstacle acoustic scattering problems by a regularized Newton method. Inverse Problems, 18(5) (2002), 1229–1246.

[49] Feldmann U. and Gunther M., The dae index in electric circuit simulation in Proc. IMACS Symposium on Mathematical Modelling, I. Troch and F. Breitenecker Eds., 4, (1994), 695-702.

[50] Feng K., On difference schemes and symplectic geometry, Proceedings of the 5-th Intern. Symposium on differential geometry and differential equations, Beijing, (1985), 42-58.

[51] Ferziger J.H. and Peric M., Computational Methods for Fluid Dynamics, Springer- Verlag, Berlin, Germany, 1980.

[52] Giles D.R.A., An algebraic approach to A-stable linear multistep-multiderivative integration formulas. BIT 14, (1978), 382-406.

[53] Gottwald B.A. MISS - Ein Einfaches Simulations-System für Biologische und Chemische Prozesse. EDV in Medizin und Biologie, 3, (1997), 85-90.

[54] Grau-Sánchez M., Grau À. and Noguera M., Frozen divided difference scheme for solving systems of nonlinear equations. J. Comput. Appl. Math. 235(6) (2011), 1739-1743.

[55] Grau-Sánchez M., Grau À. and Noguera M., On the computational efficiency index and some iterative methods for solving systems of nonlinear equations. J. Comput. Appl. Math. 236(6) (2011), 1259-1266.

[56] Grau M. and Noguera M., A variant of Cauchy's method with accelerated fifth-order convergence. Appl. Math. Lett., 17(5) (2004), 509–517.

[57] Guglielmi N. and Hairer E., Implementing Radau IIa methods for stiff delay differential equations, Computing, 67, (2001), 1-12.

[58] Guglielmi N. and Hairer E., Computing breaking points in implicit delay differential equations. Adv. Comput. Math. 29(3), (2008), 229-247.

[59] Günther M., Denk G. and Feldmann U., How models for MOS transistors reflect charge distribution effects ? Technical Report 1745, Technische Hochschule Darmstadt, Fachbereich Mathematik, Darmstadt, 1995.

[60] Günther M. and Rentrop P., The NAND-gate-a benchmark for the numerical simulation of digital circuits. In W. Mathis and P. Noll, editors, 2.ITG-Diskussionssitzung "Neue Anwendungen Theoretischer Konzepte in der Elektrotechnik" - mit Gedenksitzung zum 50. Todestag von Wilhelm Cauer, pages 27-33, Berlin, 1996. VDE-Verlag.

[61] Guo C-H. and Higham N.H., Iterative solution of a nonsymmetric algebraic Riccati equation. SIAM J. Matrix Anal. Appl., 29(2) (2007), 396-412.

[62] Guo C-H. and Laub A.J., On the iterative solution of a class of nonsymmetric algebraic Riccati equations. SIAM J. Matrix Anal. Appl., 22(2) (2000), 376-391.

[63] Hager W. and Zhang H., A survey of nonlinear conjugate gradient methods. Pac. J. Optim. 2(1), (2006), 35-58.

[64] Hairer E., Lubich C. and Roche M., The Numerical Solution of Differential-Algebraic Systems by Runge-Kutta Methods. Lecture Notes in Mathematics 1409. Springer-Verlag, 1989.

[65] Hairer E., Lubich C. and Wanner G., Geometric Numerical Integration: Structure-Preserving Algorithms for Ordinary Differential Equations, Springer Series in Computational Mathematics, Springer, 2006.

[66] Hairer E. and Wanner G., Solving Ordinary Differential Equations II: Stiff and Differential Algebraic Problems, Springer-Verlag, Berlin, Germany, 1991.

[67] Herceg D., Vulanović R. and Petrović N., Higher order schemes and Richardson extrapolation for singular perturbation problems. Bull. Austral. Math. Soc. 39, (1989), 129–139.

[68] Hernández M.A. and Romero N. On a characterization of some Newton-like methods of R-order at least three. J. Comput. Appl. Math., 183(1) (2005), 53–66.

[69] Hernández M.A. and Salanova M.A., Modification of the Kantorovich assumptions for semilocal convergence of the Chebyshev method. J. Comput. Appl. Math. 126(1-2), (2000), 131-143.

[70] Higueras I. and Garca-Celayeta B., Runge-Kutta methods for DAEs. A new approach. Numerical methods for differential equations, J. Comput. Appl. Math. 111(1-2), (1999), 49-61.

[71] Iserles A., A First Course in the Numerical Analysis of Differential Equations. 2nd edn. Cambridge University Press, Cambridge, 2008.

[72] Jay L.O., Solution of index 2 implicit differential-algebraic equations by Lobatto Runge-Kutta methods. BIT 43(1), (2003), 93-106.

[73] Jay L.O., Convergence of Runge-Kutta methods for differential-algebraic systems of index 3. Appl. Numer. Math. 17(2), (1995), 97-118.

[74] Kantorovich L.V. and Akilov G.P., Functional analysis. Pergamon Press, Oxford, 1982.

[75] Kaps P. and Wanner G., A study of Rosenbrock-type methods of high order. Numer. Math. 38(2), (1981/82), 279-298.

[76] Kim Y.I. and Geum Y.H., A cubic-order variant of Newton's method for finding multiple roots of nonlinear equations. Comput. Math. Appl. 62(4), (2011), 1634-1640.

[77] Kulkarni R.P. and Grammont L., Extrapolation using a modified projection method. Numer. Funct. Anal. Optim. 30(11-12), (2009), 1339–1359.

[78] Lambert J.D., Numerical Methods for Ordinary Differntial Systems: The initial value problem. John Wiley and Sons Ltd. 1991.

[79] Lancaster P. and Rodman L., Algebraic Riccati equations, Oxford University Press, 1995.

[80] Lasagni F.M., Canonical Runge-Kutta methods, ZAMP 39, (1988), 952-953.

[81] Lasiecka I. and Triggiani R., Control Theory for Partial Differential Equations: Continuous and Approximation Theories, Part 1, Cambridge University Press, 2000.

[82] Lasiecka I. and Triggiani R., Control Theory for Partial Differential Equations: Continuous and Approximation Theories, Part 2, Cambridge University Press, 2000.

[83] Lee J.Y. and Greengard L., A fast adaptative numerical method for stiff two-point boundary value problems. SIAM J. Sci. Comput. 18(2), (1997), 403-429.

[84] Leimkuhler B. and Reich S., Simulating Hamiltonian Dynamics. Cambridge University Press, Cambridge, 2004.

[85] Lew A., Marsden J.E., Ortiz M. and West M., Variational time integrators. Internat. J. Numer. Methods Engrg., 60(1), (2004), 153-212.

[86] Li W. and Chen H., A unified framework for the construction of higher-order methods for nonlinear equations. Open Numer. Methods J. 2, (2010), 6-11.

[87] Lioen W.M. and de Swart J.J.B., Test Set for Initial Value Problem Solvers. Modelling, Analysis and Simulation (MAS). MAS-R9832, December 1998

[88] Liu T., Yann N. and Zhang S., Richardson extrapolation and defect correction of finite element methods for optimal control problems. J. Comput. Math. 28(1), (2010), 55-71.

[89] Lubich Ch. and Ostermann A., Linearly implicit time discretization of nonlinear parabolic equations. IMA J. Numer. Anal. 15(4), (1995), 555–583.

[90] Manning D.W., A computer technique for simulating dynamic multibody systems based on dynamic formalism. PhD thesis, Univ. Waterloo, Ontario, 1981.

[91] Marsden J.E. and West M., Discrete mechanics and variational integrators. Acta Numer. 10, (2001), 357-514.

[92] McLachlan R.I. and Quispel R.G.W., Splitting methods. Acta Numerica 11, (2002), 341-434.

[93] Okunbor D. and Skeel R.D., Explicit canonical methods for Hamiltonian systems. Math. Comp. 59, (1992), 439-455.

[94] Pedregal P., A variational approach to dynamical systems, and its numerical simulation, Numer. Funct. Anal. Opt., 31(7), (2010), 1532-2467.

[95] Pedregal P., On a generalization of compact operators, and its application to the existence of critical points without convexity, Arch. Rat. Mech. Anal., 197, (2010), 965-983.

[96] Petkovic M.S., Džunic J. and Neta B., Interpolatory multipoint methods with memory for solving nonlinear equations. Appl. Math. Comput. 218(6), (2011), 2533-2541.

[97] Press W.H., Teukolsky S.A., Vetterling W.T., Flannery B.P., Numerical recipes in C : The art of scientific computing, 2nd ed., Cambridge University Press, 1995.

[98] Rabier P.J., Rheinboldt W.C., A general existence and uniqueness theory for implicit differential-algebraic equations. Differential Integral Equations 4(3), (1991), 563-582

[99] Reich S., On an existence and uniqueness theory for nonlinear differential-algebraic equations. Circuits Systems Signal Process. 10(3), (1991), 343-359.

[100] Rentrop P., Roche M. and Steinebach G., The application of Rosenbrock-Wanner type methods with stepsize control in differential-algebraic equations. Numer. Math., 55, (1989), 545-563.

[101] Riaza R., Differential-Algebraic Systems, Analytical Aspects and Circuit Applications, World Scientific, Singapore, 2008.

[102] Riaza R. and März R., Linear index-1 DAEs: regular and singular problems. Acta Appl. Math. 84(1), (2004), 29-53.

[103] Romero N., PhD Thesis. Familias paramétricas de procesos iterativos de alto orden de convergencia. http://dialnet.unirioja.es/

[104] Ruth R.D., A canonical integration technique, IEEE Trans. Nuclear Science NS 30, (1983), 2669-2671.

[105] Sanz-Serna J.M., Runge-Kutta schemes for Hamiltonian systems, BIT 28, (1988), 877-883.

[106] Sanz-Serna J.M. and Calvo M.P., Numerical Hamiltonian Problems. Chapman and Hall, London, 1994.

[107] Schäfer E., A new approach to explain the high irradiance responses of photomorphogenesis on the basis of phytochrome. J. of Math. Biology, 2, (1975), 41-56, .

[108] Schichman H. and Hodges D.A., Insolated-gate field-effect transistor switching circuits. IEEE J. Solid State Circuits, **3**, (1968), 285-289.

[109] Schneider S., Intégration de systèmes d' équations différentielles raides et différentielles algébriques par des méthodes de collocations et méthodes générales linéaires. PhD thesis, Université de Genève, 1994.

[110] Series of lectures on DAEs, URL: http://www.win.tue.nl/casa/meetings/seminar/previous/

[111] Sha F. and Tan X., A class of iterative methods of third order for solving nonlinear equations. Int. J. Nonlinear Sci. 11(2), (2011), 165-167.

[112] Shampine L., Reichelt M.W. and Kierzenka J.A., Solving index-1 DAEs in MATLAB and Simulink. SIAM Rev. 41(3), (1999), 538-552.

[113] Shampine L.F. and Thompson S., Solving DDEs in MATLAB. Appl. Numer. Math., 37(4), (2001), 441-458.

[114] Smirnova A., Renaut A.R. and Khan T., Convergence and application of a modified iteratively regularized Gauss-Newton algorithm. Inverse Problems, 23(4), (2007), 1547–1563.

[115] Stephanopoulos G., Chemical Process Control: An Introduction to Theory and Practice. Prentice-Hall, Englewood Cliffs, N.J., 1984.

[116] Stoer J. and Bulirsch R., Introduction to Numerical Analysis. Springer-Verlag. Second edition, 1993.

[117] Suris Y.B., On the conservation of the symplectic structure in the numerical solution of Hamiltonian systems (in Russian), In: Numerical Solution of Ordinary Differential Equations, ed. S.S. Filippov, Keldysh Institute of Applied Mathematics, USSR Academy of Sciences, Moscow, (1988), 148-160.

[118] Swart J.J.B. and Lioen W.M., Collecting real-life problems to test solvers for implicit differential equations. CWI Quarterly, 11(1), (1998), 83-100.

[119] The MathWorks, Inc. *MATLAB* and *SIMULINK*, Natick, MA.

[120] Traub J.F., Iterative methods for the solution of equations, Prentice Hall, Englewwod Clifss, New Jersey, 1964.

[121] Urbani A.M. and Marfurt M., Improvements of the linearly implicit Euler method. Int. J. Appl. Math. 7(4), (2001), 383–388.

[122] Vogelaere R., Methods of integration which preserve the contact transformation property of the Hamiltonian equations, Report No. 4, Dept. Math., Univ. Of Notre Dame, Notre Dame, Ind., 1956.

[123] Vulanović R., Herceg, D. and Petrović, N., On the extrapolation for a singularly perturbed boundary value problem. Computing 36, (1986), 69–79.

[124] Wang P., A third-order family of Newton-like iteration methods for solving nonlinear equations. J. Numer. Math. Stoch. 3(1), (2011), 13-19.

[125] Zhong G. and Marsden J., Lie-Poisson Hamilton-Jacobi theory and Lie-Poisson integrators. Phys. Lett. 133, (1988), 134.

[126] Zhou X., Chen X. and Song Y., Constructing higher-order methods for obtaining the multiple roots of nonlinear equations. J. Comput. Appl. Math. 235(14), (2011), 4199-4206.