Aspects of solving non-linear boundary value problems numerically

Aspects of solving non-linear boundary value problems numerically

M.E. Kramer

Aspects of solving non-linear boundary value problems numerically

Proefschrift

ter verkrijging van de graad van doctor aan de Technische Universiteit Eindhoven, op gezag van de Rector Magnificus, prof.dr. J.H. van Lint, voor een commissie aangewezen door het College van Dekanen in het openbaar te verdedigen op dinsdag 25 februari 1992 om 16.00 uur.

door

Martina Elijda Kramer

geboren te Leidschendam

Dit proefschrift is goedgekeurd door de promotoren

prof.dr. R.M.M. Mattheij

prof.dr. G.W. Veltkamp

Aan mijn ouders

Contents

Preface

In this thesis we study solution methods for well-conditioned boundary value problems (BVP's) of the form

(P.1a) $\dot{y}(x) = h(x,y)$, $a<x<b$, $y : [a,b] \to \mathbb{R}^n$ and $h : [a,b]\times\mathbb{R}^n \to \mathbb{R}^n$,

(P.1b) $g(y(a),y(b)) = 0$, $g : \mathbb{R}^n\times\mathbb{R}^n \to \mathbb{R}^n$.

Let $y^*(x)$ denote an isolated solution of (P.1). The BVP is considered to be well-conditioned at $y^*(x)$, if small changes in the functions h and g induce a small change in the solution only. A more precise definition is given in chapter 1. An important property of a well-conditioned BVP's is that its linearization at $y^*(x)$ is dichotomic, i.e. the solution space of the linearization can be split into a subspace of non-decaying modes and one of non-growing modes and the angle between both subspaces is bounded away from zero. Moreover, well-conditioning implies that both solution modes are well controlled by the boundary conditions (BC). In particular, if the boundary conditions are separated (see §1.1), the growing modes are controlled at the end point $x = b$ and the decaying modes at the initial point $x = a$.

A well-known solution method for BVP's is multiple shooting. For this method the interval $[a,b]$ is split into N subintervals $[x_k , x_{k+1}]$, with

(P.2) $a = x_1 < x_2 < \ldots < x_{N+1} = b$

and on every subinterval, an initial value problem (IVP) is defined :

(P.3a) $\dot{y}(x) = h(x,y)$, $x_k < x < x_{k+1}$,

(P.3b) $y(x_k) = s_k$, $s_k \in \mathbb{R}^n$.

The shooting vectors s_k have to be determined such that the solutions of the local IVP's form a continuous function on $[a,b]$ that satisfies the global BC (P.1b). The vector of these nN non-linear equations will be denoted by $f(s) = 0$, where $s^T = (s_1^T, s_2^T, \ldots, s_N^T)$ and $f : \mathbb{R}^{nN} \to \mathbb{R}^{nN}$.

The local IVP's (P.3) will not be well-conditioned if the ODE (P.1a) contains exponentially growing modes (indeed, there are no end-point conditions to control them). This may cause some problems, like error amplification, non-existence of the local solutions and high sensitivity of $f(s)$ for changes of s in some directions. Generally one tries to reduce these problems by choosing small subintervals, thus diminishing the effect of the exponentially growing modes. In particular for non-linear BVP's the high sensitivity of $f(s)$ is important, because most non-linear solvers, including Newton's method and its variants,

are not really equipped to handle this.

These features of the multiple shooting method have been the starting point for the research presented in this thesis. We will describe two solution methods, based on multiple shooting, that do more justice to both the growing and decaying solution modes of the BVP. The first method, *(preconditioned) time stepping*, is an alternative solver for $f(s) = 0$; the second method, *unbiased multiple shooting*, tackles the problem at an earlier stage in defining BVP's locally instead of IVP's.

The first method is based on the time stepping idea, i.e. $f(s)$ is embedded in an IVP

$$\text{(P.4a)} \qquad \frac{ds}{dt} = M(s)f(s) \qquad , t>0 \ ,$$

$$\text{(P.4b)} \qquad s(0) = s^0 \qquad , s^0 \in \mathbb{R}^{nN} \ ,$$

where the preconditioner $M(s) : \mathbb{R}^{nN} \rightarrow \mathbb{R}^{nN \times nN}$ is such that the requested zero s^* of $f(s)$ is a stable steady state of (P.4); i.e. it can be reached by integration of the IVP through time.

An important point is to find a suitable preconditioner. Hereto we construct a transformation that fully decouples the growing and decaying modes of the discretization of the linearized BVP. The preconditioner we construct applies these transformations and inverts the increment of the growing modes, thus reversing the integration direction for those modes.

Another point of research concerning the time stepping method to solve $f(s)$, is the integration method for (P.4). The method does not necessarily have to give a good approximation of the solution $s(t)$ of (P.4); we only want to reach the limiting state as quickly as possible, i.e. take larger steps. Since for most explicit integration methods, stability imposes a bound on the step size, we like to use an implicit method. On the other hand the preconditioner can be obtained, only after the Jacobian $J(s)$ of $f(s)$ is computed. Since computation of $M(s)$ is (relatively) expensive, we would prefer to use a method that is not implicit in $M(s)$. We found a compromise between these two requirements in following method:

$$\text{(P.5)} \qquad s^{j+1} = s^j + h_j M(s^j) f(s^{j+1}) \qquad , j \geq 0 \ ,$$

with h_j the step size. This method, named *mixed Euler*, is explicit in $M(s)$, thus reducing the computational costs, and implicit in $f(s)$, thus allowing for larger time steps once s^j is close to the stationary point, as we will show in chapter 3. This integration method to find the solution of the non-linear equation $f(s) = 0$, requires at every step the solution of the non-linear equation (P.5). However, these equations can be solved by Newton's method

for sufficiently small h_j, if s^j is far from the stationary point s^*, and for all h_j, once s^j is sufficiently close to s^*. If the preconditioner is chosen equal to $-J^{-1}(s)$, then the IVP (P.4) can be considered as the closure of Newton's method. Indeed, integration with the explicit Euler method and step size 1, yields Newton's iteration. And integration of (P.4) with mixed Euler converges to Newton's method if the step size approaches infinity.

The second method to improve the performance of multiple shooting on non-linear BVP's, allows for the character of growing modes before discretization. This leads us to consider the use of boundary conditions on the subintervals, i.e. try to solve on every subinterval a BVP :

(P.6a) $\dot{y}(x) = h(x,y)$, $x_k < x < x_{k+1}$,

(P.6b) $A_k y(x_k) + B_k y(x_{k+1}) = s_k$, $s_k \in \mathbb{R}^n$.

If the local BC are such that (P.6) is well-conditioned for every $k \in \{1, \ldots, N\}$, the resulting set of non-linear equations for s, i.e. $f(s)$, will not be overly sensitive for changes in s in any direction. In fact both the Jacobian of $f(s)$ and the Lipschitz constant of the Jacobian can be bounded in terms of the conditioning constant of the local BVP's (P.6). Hence we expect Newton's method to perform satisfactorily. However, on every subinterval we have to solve again a non-linear boundary value problem (which is our original problem). Nevertheless, an implementation of this idea, using collocation or finite differences for the local BVP's, yields a stable algorithm for non-linear BVP's, that can easily be implemented on a parallel computer. Moreover, in a sequential setting the memory use of this algorithm will be considerably less than for collocation or finite differences on the entire interval $[a,b]$.

The structure of the thesis is as follows. In chapter 1 we review the conditioning of both linear and non-linear BVP's and the relation between conditioning, dichotomy and boundary conditions. In chapter 2 we briefly describe existing solution methods for BVP's and derive an estimate for the Lipschitz constant of the Jacobian of $f(s)$. The two subsequent chapters deal with preconditioned time stepping. Convergence and implementation of the mixed Euler integration method are described in chapter 3. The formulation and properties of the preconditioner are the subject of chapter 4. Finally, chapter 5 considers the generalization (P.6) of multiple shooting.

1 Conditioning and dichotomy of boundary value problems

This thesis deals with numerical solution methods for non-linear boundary value problems (BVP's). Since these methods always introduce errors (e.g. rounding, discretization), it is important to be able to assess the influence of small perturbations on the solution of the problem. Therefore we dedicate this first chapter to the description of the conditioning of BVP's.

Over the past decade the conditioning of linear BVP's has been studied by many authors, see e.g. [Ma82,dHMa85,dHMa87]. Especially the link between well-conditioning, the boundary conditions (BC) and dichotomy, i.e. the splitting of the solution space into non-increasing and non-decreasing modes, has turned out to be a very useful tool in understanding the nature of BVP's and in the development of solution methods. In the first three sections we will state the commonly used definitions of well conditioning of linear BVP's and of dichotomy. Additionally we give an account of some relevant relations between these concepts and the influence of slight perturbations of the BVP on conditioning and dichotomy.

The conditioning of non-linear BVP's has not received much attention in literature thus far. In §4 we will give a definition of well-conditioning of non-linear BVP's that slightly differs from the one given in [Ma89] and investigate the link between conditioning of a non-linear BVP and its linearization.

§1.1 Conditioning of linear BVP's

Consider the linear boundary value problem

(1.1.1a) $\mathcal{L}y = q \quad , \quad y \in C^1([a,b] \to \mathbb{R}^n)$,
(1.1.1b) $\mathcal{B}y = \beta$,

with $q \in C([a,b] \to \mathbb{R}^n)$ and $\beta \in \mathbb{R}^n$, where the operators $\mathcal{L} : C^1([a,b] \to \mathbb{R}^n) \to C([a,b] \to \mathbb{R}^n)$ and $\mathcal{B} : C([a,b] \to \mathbb{R}^n) \to \mathbb{R}^n$ are defined by

(1.1.1c) $\forall_{y \in C^1([a,b] \to \mathbb{R}^n)} \quad \forall_{x \in [a,b]} \quad : \quad (\mathcal{L}y)(x) := \dot{y}(x) - A(x)y(x)$

and

(1.1.1d) $\forall_{y \in C^1([a,b] \to \mathbb{R}^n)} \quad : \quad \mathcal{B}y := B_a y(a) + B_b y(b)$,

with $A \in C([a,b] \to \mathbb{R}^{n \times n})$ and $B_a, B_b \in \mathbb{R}^{n \times n}$.

A well-known concept regarding linear ordinary differential equations (ODE's) with initial conditions only, is the *fundamental solution*, i.e. a matrix function $\Phi \in C^1([a,b] \to \mathbb{R}^{n \times n})$ that satisfies

(1.1.2) $\dot{\Phi}(x) = A(x)\Phi(x) \quad , \quad a < x < b$

and has n independent columns at every $x \in [a,b]$. The existence of such a matrix is based on the fact that if we start out at $x = a$ in n independent directions and integrate the ODE (1.1.1a) with $q \equiv 0$, the solutions will remain independent (see e.g. [Be53]). The fundamental solution is not uniquely determined; the matrix function ΦH is also a fundamental solution, for any non-singular $H \in \mathbb{R}^{n \times n}$.

Any solution of (1.1.1) can be expressed in terms of a fundamental solution Φ by

(1.1.3) $y(x) = \Phi(x)c + \int_a^x \Phi(x)\Phi^{-1}(t)q(t)\,dt,$

where the vector c is determined by the boundary conditions, viz.

(1.1.4) $[\mathcal{B}\Phi]c = \beta - B_b \int_a^b \Phi(b)\Phi^{-1}(t)q(t)\,dt.$

Hence we see that (1.1.1) has a unique solution for every β and q iff $\mathcal{B}\Phi$ is non-singular. In this thesis we assume that (1.1.1a) has a fundamental solution Φ such that this is true (in which case this property holds for all fundamental solutions). Consequently this property holds for all fundamental solutions of (1.1.1a) and we assume without loss of generality that Φ is scaled such that

(1.1.5) $\mathcal{B}\Phi = B_a\Phi(a) + B_b\Phi(b) = I_n$,

with I_n the $n \times n$ identity matrix. Consequently rank($(B_a \mid B_b)$) = n and rank(B_a) + rank(B_b) $\geq n$.
The relation (1.1.3) gives an expression of the solution of the linear BVP (1.1.1) in terms of the fundamental solution, the inhomogeneity q and the boundary value β. However, this form is not suitable for analyzing the conditioning of the BVP, because, as we see from (1.1.4), the vector c depends on both β and q. If the formula (1.1.4) is substituted into (1.1.3) we obtain the *Green's function*

$$G(x,t) = \begin{cases} \Phi(x)B_a\Phi(a)\Phi^{-1}(t) & , \ t \leq x \ , \\ -\Phi(x)B_b\Phi(b)\Phi^{-1}(t) & , \ t > x \end{cases} \tag{1.1.6}$$

and the solution of (1.1.1) now reads

$$y(x) = \Phi(x)\beta + \int_a^b G(x,t)q(t)\,dt \ . \tag{1.1.7}$$

The form of the Green's function for BVP is in fact a generalization of the one for initial value problems, where $B_b = 0$ and $B_a\Phi(a) = I_n$.

The representation (1.1.7) for $y(x)$ can now be used to estimate the influence of the boundary value and the inhomogeneity on $y(x)$. But first we have to introduce some notational conventions concerning norms.

1.1.8 Notation

The single lines $|\ .\ |_p$ *denote the p-Hölder norm of a vector or a matrix, i.e.*

$$\forall_{x \in \mathbb{R}^n} \quad : \quad |x|_p := \begin{cases} \left(\sum_{j=1}^{n} |x_j|^p \right)^{1/p} & , \ 1 \leq p < \infty \ , \\ \max_j |x_j| & , \ p = \infty \end{cases} \tag{1.1.8a}$$

and

$$\forall_{A \in \mathbb{R}^n} \quad : \quad |A|_p := \max_{x \neq 0} \frac{|Ax|_p}{|x|_p} \ . \tag{1.1.8b}$$

The double lines $\|\ .\ \|_{r,p}$ *will denote the r-Hölder function norm with respect to the p-vector norm, i.e. with* $f \in L_r([a,b] \to \mathbb{R}^n)$ or $f \in L_r([a,b] \to \mathbb{R}^{n \times n})$

(1.1.8c) $$\|f\|_{r,p} := \begin{cases} \left(\int_a^b |f(t)|_p^r dt\right)^{1/r} & , \ 1 \le r < \infty , \\ \max\limits_{a \le t \le b} |f(t)|_p & , \ r = \infty . \end{cases}$$

♦

The somewhat unusual choice of vector norm notation is due to the fact that we want to distinguish explicitly between function and vector norms. In this thesis we generally use the Euclidian vector norm; therefore the subscript p will be omitted if $p = 2$.

1.1.9 Definition

The conditioning constant κ_{lin} *of the BVP* (1.1.1) *is* $\max(\kappa_1,\kappa_2)$, *where*

(1.1.9a) $$\kappa_1 = \|\Phi\|_\infty ,$$

(1.1.9b) $$\kappa_2 = \max_{a \le x,t \le b} |G(x,t)| .$$

♦

Hence

(1.1.10) $$\|y\|_\infty \le \kappa_1 |\beta| + \kappa_2 \|q\|_1 \le \kappa_{lin}(|\beta| + \|q\|_1) .$$

Often one refers to a condition number as a quantity that measures the maximum ratio of the *relative* error due to *relative* perturbations; hence it is invariant under scaling (cf. the condition number of a matrix). Here, we use the terminology *conditioning constant* on purpose (cf. [AsMaRu]) as it refers to *absolute* errors instead. So the conditioning constant is uniquely defined only, if the scaling of the problem is standardized.
By default we will use the natural scaling of the ODE-part as given in (1.1.1a), where all the derivatives have coefficient 1, unless stated differently; this might for instance occur for singularly perturbed problems where a more desirable scaling is instead

(1.1.11) $$(\mathfrak{L}y)(x) := \varepsilon \dot{y}(x) - A(x)y(x) \quad , \ a < x < b .$$

The scaling of the BC requires somewhat more care. As was pointed out in [dHMa85, AsMaRu] the straightforward scaling

(1.1.12) $$\max(|B_a|,|B_b|) = 1$$

may lead to a rather unbalanced situation, like

(1.1.13) $$B_a = \begin{pmatrix} 1 & 0 \\ 0 & 10^{-6} \end{pmatrix} , \ B_b = \begin{pmatrix} 1 & 0 \\ 0 & 10^{-5} \end{pmatrix} .$$

A more satisfactory scaling is obtained if we consider both matrices together. Consider the QR-factorization

$$\left(\frac{B_a^\top}{B_b^\top}\right) = QR \ , \tag{1.1.14}$$

with $Q \in \mathbb{R}^{2n\times n}$ orthogonal and $R \in \mathbb{R}^{n\times n}$ upper triangular. Then (1.1.1b) reads

$$R^\top Q^\top \left(\frac{y(a)}{y(b)}\right) = \beta \tag{1.1.15}$$

and balanced BC emerge if the equation is premultiplied by $(R^\top)^{-1}$. Note that R must be invertible if we require the BVP to have a solution for every vector $\beta \in \mathbb{R}^n$. Hence we make the following assumption.

1.1.16 Assumption
The matrix $(B_a \mid B_b)$ *has orthonormal rows.*
♦

Boundary conditions that satisfy this assumption can be written in a special form, see e.g. [dHMa85, AsMaRu].

1.1.17 Lemma
There are orthogonal matrices $Q_1, Q_2, V \in \mathbb{R}^{n\times n}$ *and non-negative diagonal matrices* $\Sigma_1, \Sigma_2 \in \mathbb{R}^{n\times n}$ *such that*

$$B_a = V\Sigma_1 Q_1^\top \quad \text{and} \quad B_b = V\Sigma_2 Q_2^\top \ . \tag{1.1.17a}$$

Moreover

$$\Sigma_1^{\,2} + \Sigma_2^{\,2} = I_n \ . \tag{1.1.17b}$$

♦

Due to this decomposition of the matrices B_a and B_b, it is meaningful to review the notion of *separated* boundary conditions. Generally the BC are called separated if some, say r, conditions only involve $y(a)$ and the remaining $(n\text{-}r)$ only involve $y(b)$. However, consider the BC

$$B_a = \begin{pmatrix} \cos\alpha & 0 \\ \sin\alpha & 0 \end{pmatrix} \ ; \ B_b = \begin{pmatrix} -\sin\alpha & 0 \\ \cos\alpha & 0 \end{pmatrix} \ , \ \alpha \in \mathbb{R}\setminus\{\frac{\pi}{2}k \mid k \in \mathbb{N}_0 \cup -\mathbb{N}_0\}.$$

They appear to be non-separated, but the decomposition (1.1.17a) reads

$$B_a = \begin{pmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \; ; \; B_b = \begin{pmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{pmatrix} \cdot \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}.$$

Hence premultiplication of the equation $B_a\, y(a) + B_b\, y(b) = \beta$ by $V^\top$ yields separated BC and we give the following definition of separated and partially separated boundary conditions.

1.1.18 Definition

The boundary conditions are called separated *if* $\operatorname{rank}(B_a) + \operatorname{rank}(B_b) = n$.
The boundary conditions are called partially separated *if* $\operatorname{rank}(B_a) < n$ *or* $\operatorname{rank}(B_b) < n$.

♦

1.1.19 Remark

The decomposition (1.1.17a) has a special form for (partially) separated BC. For separated BC we have

(1.1.19a) $\qquad \forall_j \;:\; (\Sigma_1)_{jj} = 0 \quad \vee \quad (\Sigma_2)_{jj} = 0$

and without loss of generality we can assume that

(1.1.19b) $\qquad \Sigma_1 = \begin{pmatrix} 0 & 0 \\ 0 & I_r \end{pmatrix}$ and $\Sigma_2 = \begin{pmatrix} I_{n-r} & 0 \\ 0 & 0 \end{pmatrix}$,

with r the rank of Σ_1.
Partially separated BC satisfy the relation

(1.1.19c) $\qquad \exists_j \;:\; (\Sigma_1)_{jj} = 0 \quad \vee \quad (\Sigma_2)_{jj} = 0$

♦

From definition (1.1.9) it seems to follow that two quantities have to be evaluated to determine the conditioning constant. However, it was pointed out in [dHMa85,AsMaRu] that there is the following relation between the fundamental solution Φ and the Green's function G :

$$\Phi(x)\Phi(x)^\top = G(x,a)G(x,a)^\top + G(x,b)G(x,b)^\top$$

Since we use the Euclidian vector norm this yields the following lemma.

1.1.20 Lemma

(1.1.20a) $\qquad \kappa_1 \le \sqrt{2}\,\kappa_2$

♦

Hence κ_2 gives sufficient qualitative insight in the conditioning constant of a boundary value problem.

Generally one is mainly interested in an upper bound for the conditioning constant. However, it is also intriguing to ask for a lower bound. For instance the (relative) condition number of a matrix can never be less than 1. For BVP we prove that the lower bound is at least $\frac{1}{2}\sqrt{2}$ and for partially separated BC even 1.

1.1.21 Lemma

(1.1.21a) $\kappa_1 \geq \frac{1}{2}\sqrt{2}$.

If the boundary conditions are (partially) separated, then

(1.1.21b) $\kappa_1 \geq 1$.

♦

Proof

Let $V,Q_1,Q_2,\Sigma_1,\Sigma_2$ be as in lemma 1.1.17 and let v_j and e_j denote the j^{th} column of V and I_n resp.. Then

$$
\begin{aligned}
& B_a\Phi(a) + B_b\Phi(b) = I_n \\
& \quad \Leftrightarrow \Sigma_1 Q_1^T\Phi(a) + \Sigma_2 Q_2^T\Phi(b) = V^T \\
& \quad \Leftrightarrow \forall_{i,j} \;:\; (\Sigma_1)_{jj} e_j^T Q_1^T\Phi(a)v_i + (\Sigma_2)_{jj} e_j^T Q_2^T\Phi(b)v_i = \delta_{i,j} \\
& \quad \Rightarrow \forall_j \;:\; 1 \leq (\Sigma_1)_{jj}|e_j^T Q_1^T\Phi(a)v_j| + (\Sigma_2)_{jj}|e_j^T Q_2^T\Phi(b)v_j| \\
(*) \qquad & \qquad\qquad \leq (\Sigma_1)_{jj}|\Phi(a)| + (\Sigma_2)_{jj}|\Phi(b)| \\
& \qquad\qquad \leq \sqrt{(\Sigma_1)_{jj}^2 + (\Sigma_2)_{jj}^2}\ \sqrt{|\Phi(a)|^2 + |\Phi(b)|^2} \\
& \qquad\qquad = \sqrt{|\Phi(a)|^2 + |\Phi(b)|^2} \ .
\end{aligned}
$$

Hence $|\Phi(a)| \geq \frac{1}{2}\sqrt{2}$ or $|\Phi(b)| \geq \frac{1}{2}\sqrt{2}$. For (partially) separated BC a more precise result can be obtained from (*), viz.

$$
\begin{aligned}
& \exists_j \;:\; (\Sigma_1)_{jj} = 0 \;\Rightarrow\; |\Phi(b)| \geq 1 \ , \\
& \exists_j \;:\; (\Sigma_2)_{jj} = 0 \;\Rightarrow\; |\Phi(a)| \geq 1 \ .
\end{aligned}
$$

♦

§1.2 Dichotomy and conditioning

When considering well conditioning of an initial value problem (often called stability in this context), it is important that there are no (rapidly) growing solutions. The situation for a BVP is quite different: there is no bias towards forward integration with respect to the independent variable x ; indeed if the BVP is rewritten in terms of the variable $\xi = -x$, we again obtain a BVP, but all decaying solutions of the original BVP are transformed into growing ones and vice versa. So it is natural to assume that the fundamental solution of a well conditioned BVP has both growing and decaying modes and that the conditioning depends on the ability of the boundary conditions to control them properly.

The first descriptions of the solution space of linear ODE's in terms of growing and decaying modes (i.e. dichotomy) were made in papers about the existence of solutions on $[0,\infty)$ or $\mathbb{R}$ of linear ODE's for certain classes of inhomogeneities, cf. [Pe,MaSc58,MaSc66, Co78]. Later on, it was noted by several authors, cf. [Ma85], that dichotomy is closely related to well conditioning of BVP's and the choice of its boundary conditions.

1.2.1 Definition

The ODE (1.1.1.a) *is* dichotomic *if there is a fundamental solution* $Y(x)$, *a projection* $P \in \mathbb{R}^{n\times n}$ *and non-negative constants* K,λ *and* μ, K *of moderate size, such that*

$$\text{(1.2.1a)} \quad \forall_{a\le t\le x\le b} \quad : \quad |Y(x)PY^{-1}(t)| \le Ke^{-\lambda(x-t)} ,$$

$$\text{(1.2.1b)} \quad \forall_{a\le x\le t\le b} \quad : \quad |Y(x)(I_n-P)Y^{-1}(t)| \le Ke^{-\mu(t-x)} .$$

The ODE is exponentially dichotomic *if* λ *and* μ *can both be chosen positive. We say that* $Y(x)$ *is dichotomic with projection* P *and constants* (K,λ,μ).

♦

On an infinite interval, an inappropriate choice of P,λ or μ would make it impossible to satisfy (1.2.1a,b) for a finite value of K. However, on a finite interval the inequalities can be satisfied for any projection and any constant λ,μ at the expense of enlarging K. Hence on a finite interval the dichotomy concept can be meaningful only if K is of moderate size. Note that K is always at least 1, because

$$K \ge |Y(a)PY^{-1}(a)| = \max_{y\neq 0} \frac{|Y(a)Py|}{|Y(a)y|} \ge 1 .$$

Some authors, see e.g. [AsMaRu], require the projection P to be orthogonal. However, this is a superfluous requirement. Indeed, if $Y(x)$ is a fundamental solution satisfying (1.2.1a,b) with a nonorthogonal projection P of rank p, then there is an invertible matrix $C \in \mathbb{R}^{n\times n}$

such that $P = C\cdot\begin{pmatrix} 0 & 0 \\ 0 & I_p \end{pmatrix}\cdot C^{-1}$. Now let $C = U\Sigma V^\top$ be the singular value decomposition of C and define $Z(x) := Y(x)U\Sigma$. Then $Z(x)$ is also a fundamental solution of the ODE, $\tilde{P} := V^\top\cdot\begin{pmatrix} 0 & 0 \\ 0 & I_p \end{pmatrix}\cdot V$ is an orthogonal projection and $Y(x)PY^{-1}(t) = Z(x)\tilde{P}Z^{-1}(t)$. Hence the ODE has a fundamental solution which satisfies the dichotomy conditions with an orthogonal projection.

The definition of dichotomy states that no solution of the ODE can switch from strongly increasing to strongly decreasing or vice versa. For (1.2.1a) implies that

$$(1.2.2) \qquad \forall_{a\le t\le x\le b}\ \forall_{c\in\mathbb{R}^n}\ :\ |Y(x)Pc| \le Ke^{-\lambda(x-t)}\,|Y(t)Pc|\ ,$$

i.e. the set $S_1 := \{\ Y(.)Pc \mid c\in\mathbb{R}^n\ \}$ consists of the solutions of the homogeneous ODE that do not grow very strongly (for moderately sized K). And analogously does the set $S_2 := \{\ Y(.)(I_n\text{-}P)c \mid c\in\mathbb{R}^n\ \}$ consist of solutions that do not decrease rapidly. The dichotomy of an ODE involves more than just this splitting into non-increasing and nondecreasing solution modes. If the angle ϑ between the two subspaces S_1 and S_2 is defined by

$$(1.2.3) \qquad \vartheta := \min\{\min\{\measuredangle(u(x),w(x)) \mid x\in[a,b]\} \mid u\in S_1\ ,\ w\in S_2\}\ ,$$

then $\cot\vartheta \le K$ (for a proof see e.g.[dHMa87]). This means that the angle is bounded away from zero and that solutions of a different type cannot get arbitrarily close to one another. The best result that can be obtained from this estimate is $\vartheta\in(0,\frac{\pi}{4}]$, because $K \ge 1$ ($\vartheta\in(\frac{\pi}{4},\frac{\pi}{2}]$ cannot be concluded).

In [dHMa87] a proof was given that well-conditioned BVP's with separated BC are dichotomic. We will give an account of their proof (adapted to our notation), because it gives some insight into the structure of the problem.

1.2.4 Lemma ([dHMa87] Th.3.2)
If the BVP (1.1.1) *has separated BC, then* $B_a\Phi(a)$ *is an orthogonal projection and* $\Phi(x)$ *satisfies* (1.2.1) *with projection* $P = B_a\Phi(a)$ *and constants* $(\kappa_2,0,0)$.
Proof
From (1.1.5) and decomposition (1.1.17a) it follows that

$$(*) \qquad V\Sigma_1 Q_1^\top\Phi(a) + V\Sigma_2 Q_2^\top\Phi(b) = I_n\ .$$

Since the boundary conditions are separated, we may assume without loss of generality that

$$\Sigma_1 = \begin{pmatrix} 0 & 0 \\ 0 & I_r \end{pmatrix} \text{ and } \Sigma_2 = \begin{pmatrix} I_{n-r} & 0 \\ 0 & 0 \end{pmatrix}.$$

Premultiplication of (*) with $\Sigma_1 V^\top$ yields

$$\Sigma_1 Q_1^\top \Phi(a) = \Sigma_1 V^\top ,$$
$$\text{i.e. } B_a \Phi(a) = V \Sigma_1 Q_1^\top \Phi(a) = V \Sigma_1 V^\top$$

is an orthogonal projection and with $P := B_a \Phi(a)$:

$$\forall_{x \geq t} \; : \; |\Phi(x) P \Phi^{-1}(t)| = |\Phi(x) B_a \Phi(a) \Phi^{-1}(t)| = |G(x,t)| \leq \kappa_2 ,$$
$$\forall_{x < t} \; : \; |\Phi(x)(I_n - P)\Phi^{-1}(t)| = |\Phi(x) B_b \Phi(b) \Phi^{-1}(t)| = |G(x,t)| \leq \kappa_2 .$$

♦

Similarly exponential dichotomy follows from an exponential bound on the Green's function. In [dHMa87] dichotomy for BVP's with not (fully) separated BC was proven as well. The proof is based on constructing separated BC for the same ODE. We will just state the result.

1.2.5 Lemma
The BVP (1.1.1) *has a fundamental solution* $Y(x)$ *for which the dichotomy relations hold with* $K = \kappa_2 + 4\kappa_2^2$ *and* $\lambda = \mu = 0$.
♦

The dichotomy lemma 1.2.4 for separated BC already sheds some light on the relationship between well-conditioning and boundary conditions. In [AsMaRu] Ch.3 a somewhat more explicit relation is given.

1.2.6 Lemma
Suppose (1.1.1) *is well-conditioned and has separated BC. If* $Y(x)$ *is a fundamental solution with dichotomy projection* P, *then*

(1.2.6a) $\quad \ker(B_a) \cap \text{range}(Y(a)P) = \{0\}$,

(1.2.6b) $\quad \ker(B_b) \cap \text{range}(Y(b)(I_n - P)) = \{0\}$.

♦

This lemma states that the homogeneous solutions that end in a non-decreasing direction are controlled by the end point conditions and that those that start in a non-increasing direction are controlled by the initial point conditions.

Not only does well-conditioning imply dichotomy, if an ODE is dichotomic and the boundary conditions are such that κ_1 is bounded, then the norm of the Green's functions has to be moderately sized as well.

1.2.7 Lemma ([AsMaRu] Th.3.103)
If the ODE (1.1.1) *is dichotomic, then* $\kappa_2 \le K(2\kappa_1 + 1)$.
♦

However, if the boundary conditions do not fit with the dichotomic behaviour of the ODE, κ_1 will be large and so will κ_2. It has been noted in several papers that for any dichotomic ODE appropriate BC can always be found. Therefore we mention a lemma which is only a slight modification of [dHMa85] Lemma 2.3.

1.2.8. Lemma ([dHMa85])
Let $E \in \mathbb{R}^{n\times n}$ *and* $Y(x)$ *be a fundamental solution of* (1.1.1a). *For any* $c,d \in [a,b]$ *with* $c < d$ *define*

$$G(x,t) = \begin{cases} Y(x)EY^{-1}(t) & , \ c \le t \le x \le d \ , \\ Y(x)(E-I_n)Y^{-1}(t) & , \ c \le x < t \le d \ . \end{cases} \tag{1.2.8a}$$

Then there exist boundary conditions B_c *and* B_d *scaled as in assumption* 1.1.16 *such that* $G(x,t)$ *is the Green's function of*

$$\begin{cases} \dot{y} = A(x)y(x) & , \ c \le x \le d \\ B_c y(c) + B_d y(d) = \beta \ . \end{cases} \tag{1.2.8b}$$

♦

1.2.9 Corollary
(*i*) *If* (1.1.1a) *is dichotomic, then there are boundary conditions such that the conditioning constants of the resulting BVP satisfy*

$$\kappa_2 \le K \quad \text{and} \quad \kappa_1 \le \sqrt{2}K . \tag{1.2.9a}$$

(*ii*) *If* E *is a projection, then the boundary conditions* B_c *and* B_d *are separated.*
Proof of (*ii*), due to [Ve].

$$\forall_{t\in(c,d)} \ : \ B_c G(c,t) + B_d G(d,t) = 0 \ \Rightarrow \ B_c Y(c)(I_n - E) = B_d Y(d) E \ .$$

Hence $\operatorname{rank}(B_c) + \operatorname{rank}(B_d) = \operatorname{rank}(B_c\,\Phi(c)) + \operatorname{rank}(B_d\Phi(d)) \le \operatorname{rank}(E) + \operatorname{rank}(I_n - E) = n$, if E is a projector.
♦

The fact that well-conditioned BVP's may contain growing modes, has given rise to stability problems for some solution methods. If we want to tackle those problems, it would be very helpful if the growing modes can be isolated. Let Y be the fundamental solution of the BVP (1.1.1), that satisfies the dichotomy relations for $P = \begin{pmatrix} 0 & 0 \\ 0 & I_p \end{pmatrix}$, then with $Y(x)$ partitioned as $Y(x) = \underset{(n-p)}{\left(Y^1(x)\right.} \mid \underset{p}{\left.Y^2(x)\right)}$,

we have $S_2 = \text{range}(Y^1(.))$ and $S_1 = \text{range}(Y^2(.))$, i.e. the first columns of Y monitor the behaviour of the most dominant modes. However, these modes are uniquely defined when considered on a (half) infinite interval only. When integrated forward over a finite subinterval, as is done in some solution techniques for BVP's (see Ch.2), the subspace of increasing modes is not uniquely determined.
In practice any linear combination of increasing and decreasing solution modes will eventually show up as a growing solution and the influence of the decreasing modes will diminish rapidly. In order to isolate the growing solutions it would be most preferable to know the starting matrix $Y(a)$, but based on the foregoing one can also do with a fundamental solution whose first $(n-p)$ columns contain at least components of the $(n-p)$ growing modes. Therefore we define, analogously to [AsMaRu], the notion of *consistency*.

1.2.10 Definition
A fundamental solution Z of (1.1.1a) *is* consistent *with Y if*

(1.2.10a) $\qquad \text{range}(Z^1(a)) \cap \text{range}(Y^2(a)) = \{0\}$

♦

If the non-singular matrix H, such that $Z = YH$, is partitioned into

$$H = \begin{pmatrix} H^{11} & H^{12} \\ H^{21} & H^{22} \end{pmatrix},$$

with $H^{22} \in \mathbb{R}^{p \times p}$, then we can give the following relations.

1.2.11 Lemma ([AsMaRu] Lemma 6.12)
(i) Z is consistent with Y iff H^{11} is non-singular.

(ii) Z is consistent with Y iff $\quad \forall_{x \in [a,b]} \; : \; \text{range}(Z^1(x)) \cap \text{range}(Y^2(x)) = \{0\}$.

♦

The criteria mentioned in this lemma of course uniquely determine consistency, however, numerically it is difficult to distinguish between a singular and a nearly singular matrix. Since the basic thought behind it, is that the influence if $Y^2(x)H^{21}$ on an arbitrary solution should not exceed the influence of $Y^1(x)H^{11}$, a better criterion for consistency would be that the *consistency constant L*, defined by

$$L := \frac{|Y^2(a)H^{21}|}{\mathrm{glb}(Y^1(a)H^{11})}, \tag{1.2.12}$$

is of moderate size.

The considerations above give us a handle to split the solution modes into a directional part and a part describing the growth behaviour. Suppose that Z is a consistent fundamental solution. Since $Z(x)$ is a continuous function there exist two continuous matrix functions $Q(x)$ and $U(x)$ that together form the QU-decomposition $Z(x) = Q(x)U(x)$ for every $x\in[a,b]$, with $Q(x)$ an orthogonal matrix and $U(x)$ an upper triangular matrix with positive diagonal elements. Now we expect the left upper block of $U(x)$ to contain information on the growth behaviour of the most dominant modes. Henceforth we consider the following splitting of $U(x)$:

$$U(x) = \begin{pmatrix} B(x) & C(x) \\ 0 & E(x) \end{pmatrix}, \tag{1.2.13}$$

with $B(x)\in\mathbb{R}^{(n-p)\times(n-p)}$ and $E(x)\in\mathbb{R}^{p\times p}$. Then we can derive the following lemma, which is a slightly stronger result than the one derived in [AsMaRu] Ch.6.

1.2.14 Lemma

$$\forall_{a\le x\le t\le b} \quad : \quad |B(x)B^{-1}(t)| \le \tilde{K}\exp(-\mu(t-x)),$$
$$\forall_{a\le t\le x\le b} \quad : \quad |E(x)E^{-1}(t)| \le \tilde{K}\exp(-\lambda(x-t)),$$

$$\textit{with} \quad \tilde{K} = K\cdot\frac{1+LK^2}{\sqrt{\sin^2\vartheta + [\max(\cos\vartheta - LK^2, 0)]^2}} \quad \textit{and } \vartheta \textit{ as defined in } (1.2.3).$$

Proof see Appendix A.

Note that if the consistency constant $L = 0$, then $\tilde{K} = K$. Fortunately, the structure of BVP's with separated boundary conditions is such that a consistent fundamental solution can easily be found.

1.2.15 Lemma ([AsMaRu] Th.6.33)
Suppose that (1.1.1a) *is well-conditioned and that the BC* (1.1.1b) *are separated with* $B_a = \begin{pmatrix} 0 \\ B_{a2} \end{pmatrix} \begin{matrix} \updownarrow n-p \\ \updownarrow p \end{matrix}$. *Let Z be a fundamental solution of* (1.1.1a) , *then Z is consistent if it satisfies*

(1.2.15a) $\quad B_{a2} Z^1(a) = 0 .$

Proof
Let Y be the fundamental solution of (1.1.1a) with $BY = I_n$, then Y is dichotomic with projection $\begin{pmatrix} 0 & 0 \\ 0 & I_p \end{pmatrix}$, according to lemma 1.2.4. Now let H be the invertible matrix such that $ZH = Y$. Then

$$\begin{pmatrix} 0 & 0 \\ 0 & I_p \end{pmatrix} H = B_a Y(a) H = B_a Z(a) = \begin{pmatrix} 0 & 0 \\ 0 & B_{a2} Z^2(a) \end{pmatrix},$$

i.e. $H^{21} = 0$. Thus the H^{11} is invertible, because H is invertible.
♦

Finally we want to consider the inverse of a fundamental solution.

1.2.16 Lemma
Let Y(x) be a fundamental solution of (1.1.1a). *Then* $Y^{-\top}(x)$ *is a fundamental solution of the ODE*

(1.2.16a) $\quad \dot{y} = -A^{\top}(x) y \qquad , a < x < b .$

Moreover, if Y(x) is dichotomic with projection P and constants (K,λ,μ), *then* $Y^{-\top}(x)$ *is dichotomic with projection* $I_n - P^{\top}$ *and constants* (K,μ,λ).
Proof

$$\begin{aligned} \forall_{x \in [a,b]} \quad : \quad Y(x)Y^{-1}(x) = I_n \quad &\Rightarrow \quad \dot{Y}(x)Y^{-1}(x) + Y(x)\dot{Y}^{-1}(x) = 0 \\ &\Leftrightarrow \quad \dot{Y}^{-1}(x) = -Y^{-1}(x)\dot{Y}(x)Y^{-1}(x) = -Y^{-1}(x)A(x) \\ &\Leftrightarrow \quad \dot{Y}^{-\top}(x) = -A^{\top}(x)Y^{-\top}(x) . \end{aligned}$$

And

$$\forall_{a \le t \le x \le b} \quad : \quad |Y^{-\top}(x)(I_n - P^{\top})(Y^{-\top}(t))^{-1}|_2 = |Y(t)(I_n - P)Y^{-1}(x)|_2 \le K e^{-\mu(x-t)} .$$

$$\forall_{a \le x < t \le b} \quad : \quad |Y^{-\top}(x)P^{\top}(Y^{-\top}(t))^{-1}|_2 = |Y(t)PY^{-1}(x)|_2 \le K e^{-\lambda(t-x)} .$$

♦

§1.3 The influence of perturbations on conditioning and dichotomy

At several points in this thesis we will consider linear BVP's that are slight perturbations of each other. Therefore it is useful to summarize the relationship between their fundamental solutions, Green's functions, conditioning constants and dichotomy behaviour. The first three items follow straightforwardly from the definitions and the possibility to regard the difference in the homogeneous term as an inhomogeneity of the original BVP. Since this has already been presented by various authors, e.g [dHMa85,AsMa], we omit proofs.

Consider the boundary value problems

(1.3.1a) $\dot{y} = A(x)y \quad , \quad a<x<b \ ,$

(1.3.1b) $\mathcal{B}y = \beta$

and

(1.3.2a) $\dot{z} = \tilde{A}(x)z \quad , \quad a<x<b \ ,$

(1.3.2b) $\tilde{\mathcal{B}}z = \tilde{\beta} \ .$

Define the matrices $\delta A(x) := \tilde{A}(x) - A(x)$, $\delta B_a := \tilde{B}_a - B_a$ and $\delta B_b := \tilde{B}_b - B_b$ and the scalar quantities $\varepsilon_B = |\delta B_a| + |\delta B_b|$ and $\varepsilon_A = \|\delta A\|$.

Let $\Phi(x)$ and $\tilde{\Phi}(x)$ denote the fundamental solutions of (1.3.1), (1.3.2), respectively. Then

(1.3.3) $$\tilde{\Phi}(x) = \Phi(x)(\mathcal{B}\tilde{\Phi}) + \int_a^b G(x,s)\delta A(s)\tilde{\Phi}(s)\mathrm{d}s \ .$$

If $\delta A(x) \equiv 0$ then the relation simplifies into, see e.g. [dHMa85],

(1.3.4) $$\tilde{\Phi}(x) = \Phi(x)(\mathcal{B}\tilde{\Phi})$$

or equivalently

(1.3.5) $$\tilde{\Phi}(x) = \Phi(x)(\tilde{\mathcal{B}}\Phi)^{-1} \ .$$

Let $G(x,t)$ and $\tilde{G}(x,t)$ denote the Green's functions of (1.3.1), (1.3.2), respectively, then

(1.3.6) $$\tilde{G}(x,t) = G(x,t) + \Phi(x)(\mathcal{B}\,\tilde{G}(.,t)) - \int_a^b G(x,s)\delta A(s)\tilde{G}(s,t)\mathrm{d}s \ .$$

The previous relations yield the possibility to compare the conditioning constants of two neighbouring BVP's.

1.3.7 Lemma

Let κ_1,κ_2 *denote the conditioning constants of* (1.3.1) *and let* $\tilde{\kappa}_1$ *and* $\tilde{\kappa}_2$ *denote the conditioning constants of* (1.3.2).

If $\kappa_1 \varepsilon_B + \kappa_2 \varepsilon_A < 1$, *then*

$$\tilde{\kappa}_1 \leq \frac{\kappa_1}{1-\kappa_1\varepsilon_B-\kappa_2\varepsilon_A} \tag{1.3.7a}$$

and

$$\tilde{\kappa}_2 \leq \frac{\kappa_2}{1-\kappa_1\varepsilon_B-\kappa_2\varepsilon_A} . \tag{1.3.7b}$$

♦

This lemma indicates that a well-conditioned BVP remains reasonably conditioned if $A(x)$ and the boundary conditions are only slightly perturbed. A disadvantage, inherent to this type of error estimates, is that no upper bound is obtained for perturbations with $\varepsilon_A,\varepsilon_B = O(1)$ and that the estimates for the conditioning constants grow quite rapidly if either ε_A or ε_B increase. Also, the upper bounds for $\tilde{\kappa}_1$ and $\tilde{\kappa}_2$ are not always sharp estimates, as the following example shows.

1.3.8 Example

Consider the BVP

$$\dot{y}(x) = \begin{pmatrix} \mu & 0 \\ 0 & -\lambda \end{pmatrix} y(x) \qquad , \ 0 < x < 1 ,$$

$$\begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} y(0) + \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} y(1) = \beta ,$$

with $\lambda,\mu \in \mathbb{R}$. Then

$$\Phi(x) = \begin{pmatrix} e^{\mu(x-1)} & 0 \\ 0 & e^{-\lambda x} \end{pmatrix} \quad \text{and} \quad G(x,t) = \begin{cases} \begin{pmatrix} 0 & 0 \\ 0 & e^{-\lambda(x-t)} \end{pmatrix} & , \ x \geq t \\ \begin{pmatrix} e^{\mu(x-t)} & 0 \\ 0 & 0 \end{pmatrix} & , \ x < t . \end{cases}$$

If we take positive λ and μ for the original (unperturbed) problem, then for all $\tilde{\lambda},\tilde{\mu} \geq 0$, we have $\tilde{\kappa}_1 = \kappa_1 = 1$ and $\tilde{\kappa}_2 = \kappa_2 = 1$, even if $\kappa_2 \varepsilon_A > 1$. On the other hand if we set out from $\lambda = \mu = 0$ and take $\tilde{\lambda} = \varepsilon$ and $\tilde{\mu} = \varepsilon$, then $\tilde{\kappa}_1 = \tilde{\kappa}_2 = e^{\varepsilon}$. And the estimate

$\tilde{\kappa}_1 \le \kappa_1(1-\kappa_2\varepsilon_A)^{-1} = (1-\varepsilon)^{-1}$ is a good approximation of e^{ε} for small ε, but a poor one for $\varepsilon > \frac{1}{2}$.

♦

Next we look at the influence of perturbations on the dichotomy of ODE's (i.e. only the value of δA plays a role); in literature this is often referred to as *roughness*. In [MaSc66] dichotomy and roughness are described in a topological context for a much larger class of differential equations, viz. where $y(x)$ may be a mapping into an infinite dimensional Banach-space. Here we state a result that was derived in [MaSc58] and [Co67], and has been reformulated into our notation.

1.3.9 Theorem

Suppose that the ODE (1.3.1a) *has an exponential dichotomy, with the fundamental solution* $\Phi(x)$ *satisfying* (1.2.1) *for the projection* P *and constants* (K,λ,μ). *Then for all* $\varepsilon < \min(\lambda,\mu)$ *there is a positive constant* δ, *depending on* $P,K,\lambda+\mu$ *and* ε *such that*

$$\|\delta A\| \le \delta \;\Rightarrow\; \textit{there is a fundamental solution } \tilde{\Phi}(x) \textit{ of } (1.3.2a) \textit{ satisfying}$$

$$|\tilde{\Phi}(x)P\tilde{\Phi}^{-1}(t)| \le \tilde{K}e^{-(\lambda-\varepsilon)(x-t)} \qquad \text{for } x \ge t$$

$$\textit{and} \quad |\tilde{\Phi}(x)(I-P)\tilde{\Phi}^{-1}(t)| \le \tilde{K}e^{-(\mu-\varepsilon)(t-x)} \qquad \text{for } x \le t\ ,$$

with $\tilde{K}$ *depending on* K *and* P.

♦

This theorem is not appropriate for practical use, as δ, though existing, may be very small. A quantitatively more useful result can be found in [Co78]. Unfortunately the proof cannot deal with $\lambda \ne \mu$, hence it renders a weaker result than one could hope for.

1.3.10 Theorem ([Co78] Ch.4)

Define $\alpha = \min(\lambda,\mu)$, *then, under the same assumptions as in Th.*1.3.9, $\varepsilon_A < \frac{1}{4}\alpha K^{-2}$ *implies that* (1.3.2a) *is exponentially dichotomic and moreover that there is a fundamental solution and a projection* $\tilde{P}$ *such that* (1.2.1a,b) *holds for the constants* $(\frac{5}{2}K^2\ ,\ \alpha-2K\varepsilon_A\ ,\ \alpha-2K\varepsilon_A)$.

♦

The literature on roughness of ordinary dichotomy is not very extensive. However, the results stated in §1.2 about the relations between conditioning and dichotomy give some more insight into the matter.

1.3.11 Lemma

If (1.3.1a) *has an ordinary dichotomy with constants* $(K,0,0)$ *and* $K\varepsilon_A < 1$, *then there is a fundamental solution of* (1.3.2a), *which is dichotomic with constants* $(K(1-K\varepsilon_A)^{-1},0,0)$.

Proof

Application of lemma 1.2.8 and corolarry 1.2.9 with $E = P$ yields that there are separated BC

(1.3.11a) $\qquad \hat{B}_a y(a) + \hat{B}_b y(b) = \beta \ ,$

such that the norm of the Green's function of (1.3.1a)+(1.3.11a) is bounded by K. Let $\hat{\kappa}_2$ denote the norm of the Green's function of the perturbed BVP (1.3.2a)+(1.3.11a). Lemma 1.3.7 implies that

$$\hat{\kappa}_2 \le \frac{K}{1-K\varepsilon_A} \ .$$

Application of lemma 1.2.4 yields that the ODE (1.3.2a) is dichotomic with constants $(K(1-K\varepsilon_A)^{-1} , 0 , 0)$.

♦

There is quite an analogy between the dichotomy of perturbed ODE's and the eigenvalues of perturbed matrices. If the dichotomy constants λ and μ are far away from zero (i.e. there is a clear splitting in growing and decaying modes), then small perturbations do not seriously change the dichotomy of the system and similarly if eigenvalues of a matrix are sufficiently separated, the influence of small perturbations will be moderate. On the other hand, even small perturbations of an ordinarily dichotomic ODE may yield a non-dichotomic ODE, as we will show in example 1.3.13, and eigenvalues of a matrix with multiple eigenvalues may change considerably under small perturbations (see e.g. [GoLo]).

The generic upper bounds derived in Lemmas 1.3.10 and 1.3.11 are of course not necessarily sharp as will be demonstrated in example 1.3.12. However, perturbations of an ordinarily dichotomic ODE may cause behaviour of the solution modes that is far from dichotomic, see example 1.3.13.

1.3.12 Example

Consider the BVP

(1.3.12a) $\dot{y} = 0 \quad , \ 0 < x < 1 \ ,$

(1.3.12b) $$\begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} y(0) + \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} y(1) = \beta \ .$$

It has an ordinary dichotomy with constants (1,0,0). Now consider the perturbed BVP

(1.3.12c) $$\dot{y} = \begin{pmatrix} \lambda & 0 \\ 0 & -\lambda \end{pmatrix} y(x) \quad , \ 0 < x < 1 \ .$$

This one has an exponential dichotomy with constants $(1,\lambda,\lambda)$. However, if $\lambda > 0$, then the dichotomy does not properly fit the BC and the conditioning constant will grow, causing lemma 1.2.4 to render a rather pessimistic view.

♦

1.3.13 Example

Again consider the BVP (1.3.12a,b). Now we choose another perturbation of the ODE-part, viz.

$$\tilde{A}(x) = \begin{pmatrix} f(x) & 0 \\ 0 & 0 \end{pmatrix} \text{ with } f(x) = \begin{cases} \varepsilon & , \ 0 \le x < \frac{1}{2} - \delta \\ \varepsilon \delta^{-1} (\frac{1}{2} - x) & , \ \frac{1}{2} - \delta \le x \le \frac{1}{2} + \delta \\ -\varepsilon & , \ \frac{1}{2} + \delta < x \le 1 \ , \end{cases}$$

with $\varepsilon > 0$ and $\delta \in (0,0.5)$ define. The fundamental solution reads

$$\tilde{\Phi}(x) = \begin{pmatrix} g(x) & 0 \\ 0 & 1 \end{pmatrix} \text{ with } g(x) = \begin{cases} \exp(\varepsilon x) & , \ 0 \le x < \frac{1}{2} - \delta \\ \exp(\frac{1}{2}\varepsilon(1 - \delta - \delta^{-1}(\frac{1}{2} - x)^2)) & , \ \frac{1}{2} - \delta \le x \le \frac{1}{2} + \delta \\ \exp(\varepsilon(1 - x)) & , \ \frac{1}{2} + \delta < x \le 1 \ , \end{cases}$$

i.e. the solution function $g(x)$ first increases exponentially and then decreases exponentially. So for larger ε-values the problem becomes non-dichotomic (and ill-conditioned), as is demonstrated by the plot of $g(x)$ for $\varepsilon = 1$ and $\delta = 0.005$.

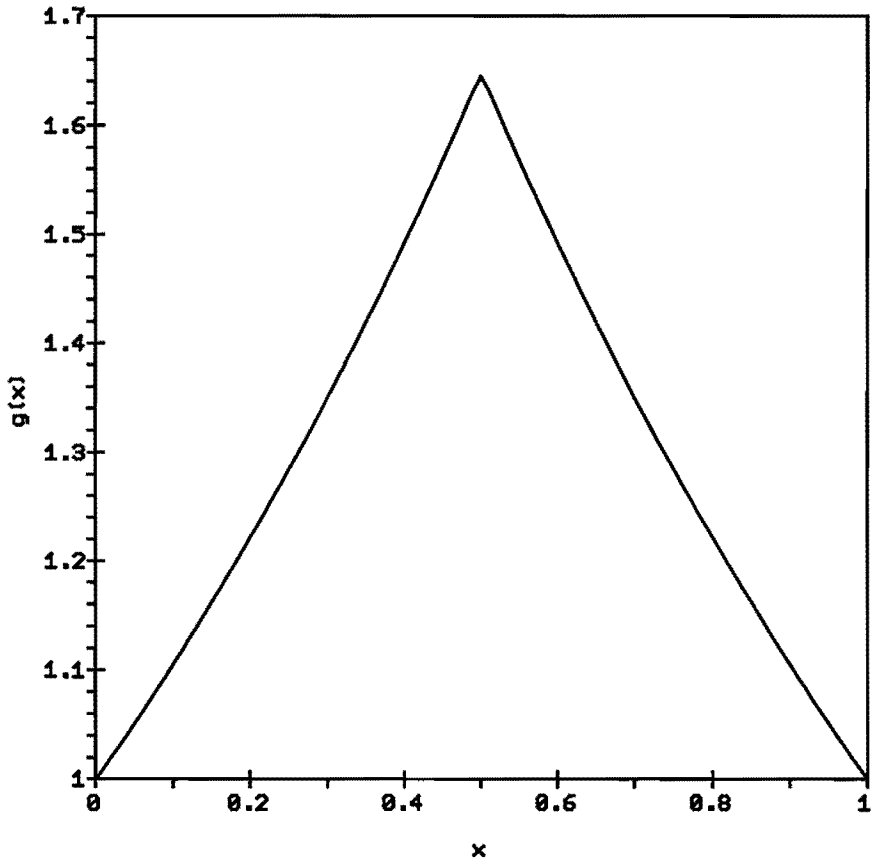


Figure 1.1

§1.4 Conditioning of non-linear BVP's

For linear boundary value problems the concept of well-conditioning and topics related to it, like dichotomy and the influence of boundary conditions, have been studied extensively. There is not much literature available about the conditioning of non-linear BVP's.
The main difference between the linear and the non-linear case is, that the size of the perturbations plays an essential role; multiplying a perturbation by some factor, may cause a considerably larger change in the solution than the original perturbation. Therefore we include the size of the perturbation in the definition of conditioning.

Consider the non-linear boundary value problem

(1.4.1a) $\dot{y} = h(x,y) \quad , \quad a<x<b \,,$
(1.4.1b) $g(y(a),y(b)) = 0 \,,$

which is assumed to have an isolated solution $y^*(x)$, i.e. there is a tube
$\{ y \in C([a,b] \rightarrow \mathbb{R}^n) \mid \| y - y^* \| < \Delta \}$ around y^* in which there is no other solution of the BVP.

1.4.2 Assumption
Let $\Delta > 0$ be a constant such that the convex neighbourhood

$$D_y := \{ y \in C([a,b] \rightarrow \mathbb{R}^n) \mid \| y - y^* \| \leq \Delta \}$$

of $y^*(x)$ satisfies

a. *y^* is the only solution of* (1.4.1) *in D_y ,*

b. *the upper bound C_{gh} on the first and second derivatives of $h(x,y)$ with respect to y and on the first and second (partial) derivatives of $g(u,v)$ is of moderate size.*

Moreover, for all $y \in D_y$ the conditioning constant of the linearization of (1.4.1) *at $y(x)$ is denoted by $\kappa_{lin}(y)$.*

♦

In order to determine the conditioning constant of the BVP, we have to consider a slightly perturbed BVP

(1.4.3a) $\dot{y} = h(x,y) + \delta h(x,y) \quad , \quad a < x < b \,,$
(1.4.3b) $g(y(a),y(b)) + \delta g(y(a)),y(b)) = 0 \,.$

For the non-linear BVP, we now introduce the following conditioning concept.

1.4.4 Definition

Let $y^*(x)$ *be an isolated solution of* (1.4.1). *For every* $\varepsilon > 0$ *define*

(1.4.4a) $$D(\varepsilon) := \{ (\delta g, \delta h) \in C(\mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^n) \times C([a,b] \times \mathbb{R}^n \to \mathbb{R}^n) \mid$$
$$(i)\ (1.4.3)\ \textit{has a solution}\ \bar{y}(x) \in D_y,\ \textit{which is unique in}\ D_y,$$
$$\textit{and}\ (ii)\ |\delta g(\bar{y}(a),\bar{y}(b))| < \varepsilon \text{ and } \|\delta h(.,\bar{y}(.))\| < \varepsilon \,. \} \,.$$

Now define the conditioning constant $\kappa(\varepsilon;y^*)$ *of* (1.4.1) *by*

(1.4.4b)

$$\kappa(\varepsilon;y^*) := \inf \{ c>0 \mid \forall_{(\delta g,\delta h)\in D(\varepsilon)} \ : \ \|\bar{y}-y^*\| \le c(|\delta g(\bar{y}(a),\bar{y}(b))| + \|\delta h(.,\bar{y}(.))\|) \} \,.$$

The BVP is well conditioned *at* $y^*(x)$ *if there is an* ε *such that* $\kappa(\varepsilon;y^*)$ *is of moderate size.*

♦

Note that we do not consider solutions of all BVP's with $|\delta g|$ and $\|\delta h\|$ bounded by ε, but only those that have a unique solution in D_y. The reason for this is that if the original BVP (1.4.1) has several solutions, $\kappa(\varepsilon;y^*)$ would have no finite value for any ε. This definition of conditioning constant differs from the one proposed in [Ma89] in two ways. Firstly we consider a somewhat smaller class of perturbations in order to guarantee existence of $\kappa(\varepsilon;y^*)$. Secondly, definition 1.4.4 is a generalisation of the definition of conditioning for linear BVP, unlike the definition in [Ma89], where in effect the conditioning constant is defined as the least upper bound on the quotient of $\|y^*-y\|$ and $\max(|\delta g|, \|\delta h\|)$. Hence that definition is not a generalisation of the one for linear BVP's. Notice that $\kappa(\varepsilon;y^*)$ is a non-decreasing function of ε and that for linear problems $\kappa(\varepsilon;y^*)$ is constant.

Since the linearization of a BVP describes the first order effect of small perturbations on the non-linear BVP, we expect the conditioning constant of the linearized BVP not to be considerably larger than that of the original non-linear one. Define

(1.4.5a) $$A(x;y) := \left.\frac{\partial h(x,v)}{\partial v}\right|_{v=y(x)} ,$$

(1.4.5b) $$B_a(y) := \left.\frac{\partial g(u,y(b))}{\partial u}\right|_{u=y(a)} \text{ and } B_b(y) := \left.\frac{\partial g(y(a),v)}{\partial v}\right|_{v=y(b)}$$

and consider the linearized BVP with inhomogeneities $q(x)$ and β

(1.4.6) $$\begin{cases} \dot{z}(x) = A(x;y)z(x) + q(x) & , \ a<x<b \,, \\ B_a(y)z(a) + B_b(y)z(b) = \beta \,. \end{cases}$$

1.4.7 Lemma

If the conditioning constant $\kappa_{lin}(y^*)$ *of the linear BVP* (1.4.6) *is finite, it does not exceed*

(1.4.7a) $\inf_{\varepsilon>0} \kappa(\varepsilon; y^*)$.

Proof

Let $\varepsilon > 0$, $q(x) \in C([a,b] \to \mathbb{R}^n)$ and $\beta \in \mathbb{R}^n$. And let $z(x)$ be the solution of the linearized BVP (1.4.6) at $y^*(x)$. For any $\alpha > 0$, the function αz is solution of (1.4.6) with inhomogeneity αq and boundary value $\alpha\beta$. Define $\bar{y}(x;\alpha) := y^*(x) - \alpha z(x)$. Then $\bar{y}(x;\alpha)$ satisfies the perturbed BVP (1.4.3) with

$$\delta h(x,y) := h(x,y^*) - h(x,y) - A(x;y^*)(y^* - y) - \alpha q(x) ,$$
$$\delta g(u,v) := g(y^*(a),y^*(b)) - g(u,v) - B_a(y^*)(u - y^*(a)) - B_b(y^*)(v - y^*(b)) + \alpha\beta .$$

And this perturbed BVP can be rewritten into the following form :

$$\begin{cases} \dot{y} = A(x;y^*)y + [h(x,y^*) - A(x;y^*)y^* - \alpha q(x)] & , \quad a < x < b , \\ B_a(y^*)y(a) + B_b(y^*)y(b) = \alpha\beta + g(y^*(a),y^*(b)) , \end{cases}$$

i.e. the linearization of (1.4.1) at $y^*(x)$. Hence its solution $\bar{y}(x;\alpha)$ is unique if $\kappa_{lin}(y^*)$ is finite. If α is sufficiently small, the solution $\bar{y}(x;\alpha)$ will be in D_y and the size of the perturbations that are bounded by

(*) $\quad |\delta g| \le 3C_{gh}\alpha^2\|z\|^2 + \alpha|\beta| \quad$ and $\quad \|\delta h\| \le C_{gh}\alpha^2\|z\|^2 + \alpha\|q\| ,$

will be smaller than ε. Now application of definition 1.4.4 yields

$$\alpha\|z\| \le \kappa(\varepsilon;y^*)(4C_{gh}\|z\|^2\alpha^2 + \alpha|\beta| + \alpha\|q\|)$$
$$\Rightarrow \quad \|z\| \le \kappa(\varepsilon;y^*)(4\alpha C_{gh}\|z\|^2 + |\beta| + \|q\|) .$$

Finally let α approach zero.

♦

From this lemma we see that, if a non-linear BVP is well conditioned, even on a small domain, then its linearization is well conditioned, too. This in turn implies that the linearized BVP is dichotomic, i.e. the solution space can be split into a subspace of non-decreasing modes and a subspace of non-increasing modes, where the angle between the two spaces is bounded away from zero.

Two remaining questions are whether well-conditioning of the linearized BVP induces well-conditioning of the non-linear one and whether well-conditioning is maintained under small perturbations. Since the proof of the latter uses the conditioning of the linearized BVP, both questions can be answered by one lemma.

1.4.8 Lemma

Let $y(x),z(x) \in D_y$ *be unique solutions of* (1.4.3) *in* D_y *for the functions* δh_y, δg_y *and* δh_z, δg_z, *respectively, and suppose that* $\kappa_{lin}(y)$ *is bounded. Define*

(1.4.8a) $\varepsilon_g := |\delta g_y(y(a),y(b)) - \delta g_z(z(a),z(b))|$ and $\varepsilon_h := \|\delta h_y(.,y(.)) - \delta h_z(.,z(.))\|$.

Then

(1.4.8b) $$\varepsilon_g + \varepsilon_h < \frac{1}{16 C_{gh} \kappa_{lin}^2(y)} \Rightarrow \|y - z\| \le 2\kappa_{lin}(y)(\varepsilon_g + \varepsilon_h) .$$

Proof

Define $w(x) := y(x) - z(x)$. Then it satisfies the linearized BVP (1.4.6) at $y(x)$ with

$$\begin{cases} q(x) = h(x,y(x)) - h(x,z(x)) - A(x;y)w(x) + \delta h_y(x,y(x)) - \delta h_z(x,z(x)) , \\ \beta = g(y(a),y(b)) - g(z(a),z(b)) + B_a(y)w(a) + B_b(y)w(b) \\ \qquad\qquad + \delta g_y(y(a),y(b)) - \delta g_z(z(a),z(b)) . \end{cases}$$

Now the norm of $w(x)$ can be estimated by

$$\|w\| \le \kappa_{lin}(y)(4 C_{gh} \|w\|^2 + \varepsilon_g + \varepsilon_h) .$$

And because of the continuity with respect to ε_g and ε_h we can easily derive that

$$\begin{aligned} \varepsilon_g + \varepsilon_h < \frac{1}{16 C_{gh} \kappa_{lin}^2(y)} \Rightarrow \|w\| &\le \frac{1 - \sqrt{1 - 16 C_{gh} \kappa_{lin}^2(y)(\varepsilon_g + \varepsilon_h)}}{8 C_{gh} \kappa_{lin}(y)} \\ &= \frac{2\kappa_{lin}^2(y)(\varepsilon_g + \varepsilon_h))}{1 + \sqrt{1 - 16 C_{gh} \kappa_{lin}^2(y)(\varepsilon_g + \varepsilon_h)}} \\ &\le 2\kappa_{lin}(y)(\varepsilon_g + \varepsilon_h) . \end{aligned}$$

♦

1.4.9 Corollary

(*i*) Application of the previous lemma for $y = y^*$, i.e. $\delta h_y = 0$ and $\delta g_y = 0$, yields that well-conditioning of the linearized problem implies well-conditioning of the non-linear BVP for small perturbations.

(*ii*) The previous lemma also yields continuity of well conditioning of non-linear BVP's. Indeed, if a non-linear BVP is well conditioned at a solution y^*, then its linearization at y^* is well conditioned, too. Hence, if $y \in D_y$ is an isolated solution of a neighbouring BVP, lemma 1.3.7 shows that for $\| y^* - y \|$ sufficiently small $\kappa_{lin}(y)$ can be bounded in terms of $\kappa_{lin}(y^*)$:

(1.4.9a) $$\kappa_{lin}(y) \le \frac{\kappa_{lin}(y^*)}{1-5\kappa_{lin}(y^*)C_{gh}\|y^*-y\|}\ ,\ \text{if}\ \|y^*-y\| < (5C_{gh}\kappa_{lin}(y^*))^{-1}.$$

Now the lemma 1.4.8 yields that the neighbouring BVP is well conditioned, too, though possibly on a small domain.

♦

We see that well-conditioning of the linearization implies well-conditioning of the original BVP, but possibly, for very small perturbations only. Therefore we have chosen to define conditioning of non-linear BVP's not in terms of the linearized BVP, but in terms of the original BVP. As we saw in §1.3, the situation can be significantly better than stated in the lemma, but it gives a realistic upper bound for the worst case (cf. example 1.3.8).

Finally, we want to mention a pitfall. Based on the definition of conditioning one may be tempted to make the 'inverse' statement, that any neighbouring function of $y^*(x)$ will satisfy a neighbouring BVP. However, conditioning is concerned with the influence of perturbations in the derivative on the primitive function, whereas here we perturb the primitive function and ask for the influence on the derivative. And it is well-known that integration is a 'smoothing' operation, but differentiation may have the inverse effect as the following example shows.

1.4.10 Lemma

Let $y^*(x)$ *be the solution of* (1.4.1). *Then*

(1.4.10a) $$\forall_{\varepsilon:0<\varepsilon<\Delta}\ \forall_{\gamma>0}\ \exists_{y\in C^1([a,b]\to\mathbb{R}^n)}\ :\ \|y-y^*\|<\varepsilon \wedge\ y(x)\ \textit{satisfies}\ (1.4.3)\ \textit{with}\ \|\delta h\| \ge \gamma\ .$$

Proof

Take $y(x) = y^*(x) + \frac{2\varepsilon}{\pi}\arctan(\alpha(x-\frac{1}{2}(a+b))\cdot\xi$ for some unit vector $\xi\in\mathbb{R}^n$ and $\alpha\in\mathbb{R}$. Then $y(x)$ is continuously differentiable and $\|y-y^*\| < \varepsilon$. Its first derivative reads

$$\dot y(x) = \dot y^*(x) + f(x) \quad \text{with} \quad f(x) = \frac{2\varepsilon\alpha}{\pi}\cdot\frac{1}{1+\alpha^2(x-\frac{1}{2}(a+b))^2}\cdot\xi\ .$$

Hence $y(x)$ satisfies (1.4.3) with $\delta h(x,y) = h(x,y) - h(x,y^*) + f(x)$ and

$$\|\delta h\| \ge \frac{2\varepsilon\alpha}{\pi} - C_{gh}\varepsilon\ .$$

Now take α sufficiently large.

♦

2 Solution methods for boundary value problems

In this chapter we give a brief description of some solution methods that are commonly used for solving BVP's. Roughly speaking these methods can be divided into two classes, viz. methods based on IVP techniques, like invariant embedding and multiple shooting, and 'global' methods, like collocation and finite differences. We will not mention all important variants, adaptations and features of both method classes, but only highlight items which are actually used in the remainder of this thesis. A more complete survey can be found in e.g. [AsMaRu,Ke76] and some references therein.

In section 2.1 we describe the IVP-based solution methods of multiple shooting and invariant embedding and pay special attention to a reorthogonalization process. The second section is devoted to global methods, viz. finite differences and collocation. The method descriptions in the first two sections are given for linear BVP's. In the third section we consider the adaptation of those methods for non-linear BVP's and pay special attention to the conditioning of the arising non-linear equations.

§2.1 Initial value techniques

The development of approximative solution methods for initial value problems (IVP's) has matured earlier than the development of such methods for BVP's. One of the obvious reasons is that IVP's can more easily be treated locally: the starting point is known beforehand and information about the local direction field can be used to approximate the solution at neighbouring points. However, BVP's are essentially non-local; there is no starting point available and information from both ends of the interval is needed everywhere. One way to circumvent this problem, is to guess the missing initial conditions and employ available IVP-software to approximate the solution. Of course this 'solution' is very likely not to satisfy the endpoint conditions. Nevertheless, the information from the integration step can be used to improve the initial point guess iteratively. In analogy to the familiar military technique of aiming a cannon correctly by trial and error, this process is called *shooting*. The mathematical description is as follows: consider the linear BVP

$$\text{(2.1.1)} \qquad \begin{cases} \dot{y} = A(x)y + q(x) & , \ a<x<b \ , \\ B_a y(a) + B_b y(b) = \beta \ . \end{cases}$$

A shooting attempt is equal to solving the IVP

$$\text{(2.1.2)} \qquad \begin{cases} \dot{y} = A(x)y + q(x) & , \ a<x<b \ , \\ y(a) = s \ , \end{cases}$$

for some vector $s \in \mathbb{R}^n$. The exact solution will be denoted by $y(x;s)$. Solving (2.1.1) is now equivalent to solving the equation

$$\text{(2.1.3)} \qquad B_a s + B_b y(b;s) = \beta \ .$$

Application of this method may encounter problems, due to the essential difference between IVP's and BVP's. Whereas a well-conditioned (stable) IVP will have no exponentially growing solution modes, a well-conditioned BVP may very well have them, as we saw in Ch.1. However, all stability considerations and error bounds of numerical methods for IVP's are based on the absence of exponentially growing modes. Another problem is that computational errors may be magnified by a factor $\exp(\mu(b-a))$, if μ is the growth factor of the strongest growing solution mode.

These drawbacks can be overcome to some extent by a more refined shooting process, generally referred to as *multiple shooting*. Here the interval is split into several subintervals $[x_k, x_{k+1}]$, $1 \le k \le N$, for some $N \in \mathbb{N}$, with

$$\text{(2.1.4)} \qquad a = x_1 < x_2 < \ldots . < x_{N+1} = b$$

and the shooting process is applied to every subinterval, i.e. the IVP's

$$\text{(2.1.5)} \qquad \begin{cases} \dot{y} = A(x)y + q(x) & , \ x_k < x < x_{k+1} \ , \\ y(x_k) = s_k \end{cases}$$

are solved for the shooting vectors $s_k \in \mathbb{R}^n$, $1 \le k \le N$. The solutions are denoted by $y_k(x;s_k)$. We now have a two level discretization : a *coarse level grid* $\{x_1, x_2, \ldots, x_{N+1}\}$ determining the shooting intervals and a *fine level grid* per subinterval used by the IVP-solver.

Before considering the choice of the coarse grid we first review the process to obtain the correct shooting vector s briefly. Not only has the set of unknowns been enlarged as compared to single shooting, but also, as to be expected, there are more conditions to be satisfied. Besides satisfying the BC, the solution pieces y_k together should form a continuous function on the entire interval eventually. For notational convenience we introduce an additional shooting vector s_{N+1}, representing the value of the solution at $x = b$. Now the vectors s_k are determined by

$$\text{(2.1.6)} \qquad f(s) = 0 \quad \text{with} \quad s^\top := (s_1^\top, s_2^\top, \ldots, s_{N+1}^\top) \quad \text{and} \quad f \in C^1(\mathbb{R}^{n(N+1)} \to \mathbb{R}^{n(N+1)})$$

and $f(s)$ defined by

$$\text{(2.1.7)} \qquad f(s) := \begin{pmatrix} y_1(x_2;s_1) - s_2 \\ \vdots \\ y_N(x_{N+1};s_N) - s_{N+1} \\ B_a s_1 + B_b s_{N+1} - \beta \end{pmatrix} .$$

Since the BVP is linear its solution can be described as

$$\text{(2.1.8a)} \qquad y(x) = \Phi_k(x)s_k + v_k(x) \quad , \ x \in [x_k, x_{k+1}] \ ,$$

where $\Phi_k(x)$ is the fundamental solution on $[x_k, x_{k+1}]$ with

$$\text{(2.1.8b)} \qquad \Phi_k(x_k) = I_n$$

and $v_k(x)$ is the particular solution with $v_k(x_k) = 0$. Now (2.1.6) is equivalent to the linear equation

$$\text{(2.1.9)} \qquad \begin{pmatrix} \Phi_1(x_2) & -I_n & & & \\ & \Phi_2(x_3) & -I_n & & \\ & & \ddots & \ddots & \\ & & & \Phi_N(x_{N+1}) & -I_n \\ B_a & & & & B_b \end{pmatrix} \begin{pmatrix} s_1 \\ s_2 \\ \\ \vdots \\ s_{N+1} \end{pmatrix} = \begin{pmatrix} -v_1(x_2) \\ -v_2(x_3) \\ \vdots \\ -v_N(x_{N+1}) \\ \beta \end{pmatrix} .$$

This equation can be solved in several ways. A very simple idea, induced by the specific form of the Jacobian, is *compactification*. Here the first N block rows are used to express s_k, $k \geq 2$, in terms of s_1 :

$$s_k = \Phi_{k-1}(x_k)\cdots\Phi_1(x_2)s_1 + \xi_k \,, \tag{2.1.10}$$

$$\text{with } \xi_k = v_{k-1}(x_k) + \sum_{j=1}^{k-2} [\Phi_{k-1}(x_k)\cdots\Phi_{j+1}(x_{j+2})]v_j(x_{j+1})$$

and obtain an equation for s_1 from the last block row:

$$[B_a + B_b\Phi_N(x_{N+1})\cdots\Phi_1(x_2)]s_1 = \beta - B_b\xi_{N+1} \,. \tag{2.1.11}$$

Notice that this is equivalent to

$$[B_a + B_b\Phi_1(x_{N+1})]s_1 = \beta - B_b v_1(x_{N+1}) \,. \tag{2.1.12}$$

A major disadvantage of compactification is that the influence of rounding errors may be considerable. The fundamental solutions $\Phi_k(x_{k+1})$ and the particular solutions $v_k(x_{k+1})$ are likely to contain errors of the size ε_M (ε_M being the machine precision). It was shown in [AsMaRu] that, with $\bar{s}_1$ the solution of the error containing version of (2.1.11) $| \bar{s}_1 - s_1 | = O(| \Phi_1(x_{N+1}) | \varepsilon_M)$. For exponentially dichotomic BVP's the norm of the latter matrix will be considerable.
Another, generally more stable, solution method for (2.1.9) could be LU-decomposition. Especially for BVP's with separated BC, several other efficient solution methods can be used, e.g. an LU-decomposition for almost block diagonal matrices [dBWe] and alternate row and column pivoting [Va]. Later in this section we will describe how a stable compactification algorithm can be performed in case of separated BC.

The linear IVP's on the subintervals will generally have (exponentially) growing modes. This has two major consequences. Firstly, the norm of $\Phi_k(x)$ will increase exponentially for larger x. Secondly, in every column of $\Phi_k(x)$ there will be a component of the strongest growing direction, either right from the start or eventually due to computational errors, and for larger x the influence of this mode will become dominant. Hence $\Phi_k(x)$ becomes (almost) singular and vital information about the non-dominant modes may be lost. Often multiple shooting codes use an adaptive choice for shooting points based on these two considerations; i.e. a new point is inserted if either the 'size' of $\Phi_k(x)$ (e.g. the absolute largest element or the $| \cdot |_\infty$) becomes too large or if the condition number of $\Phi_k(x)$ becomes too large, see e.g. [HeBe,MaSt].

In the given setting we automatically restart at every shooting point with the identity matrix. Since multiple shooting is traditionally a method that progresses in the x-direction, a quite natural step is to apply an orthogonalization process to $\Phi_k(x_{k+1})$ before its columns

have become (almost) dependent. This idea was first introduced for separated BC in [Go,Con]. If for instance a QU-decomposition $\Phi_k(x_{k+1}) = Q_{k+1}U_k$ is made, the orthogonal matrix Q_{k+1} can be used as a starting value for the fundamental solution on $[x_{k+1}, x_{k+2}]$. This process can be performed at the end of every subinterval, yielding fundamental solutions that are transformations of the ones used in (2.1.9). Hence some additional matrix vector multiplications have to be performed to solve for the vector s. Schematically the process can be described as

- choose an orthogonal matrix Q_1
- determine QU-decomposition of $\Phi_k(x_{k+1})Q_k = Q_{k+1}U_k$, $k = 1,.., N$, yielding Q_{k+1} and the upper triangular matrix U_k

Equation (2.1.9) now reads

$$\begin{pmatrix} Q_2 & & & & \\ & Q_3 & & & \\ & & \ddots & & \\ & & & Q_{N+1} & \\ & & & & I_n \end{pmatrix} \begin{pmatrix} U_1 & -I_n & & & \\ & U_2 & -I_n & & \\ & & \ddots & \ddots & \\ & & & U_N & -I_n \\ B_aQ_1 & & & & B_bQ_{N+1} \end{pmatrix} \begin{pmatrix} Q_1^T & & & & \\ & Q_2^T & & & \\ & & \ddots & & \\ & & & Q_N^T & \\ & & & & Q_{N+1}^T \end{pmatrix} \begin{pmatrix} s_1 \\ s_2 \\ \vdots \\ s_N \\ s_{N+1} \end{pmatrix} = \begin{pmatrix} -v_1(x_2) \\ -v_2(x_3) \\ \vdots \\ -v_N(x_{N+1}) \\ \beta \end{pmatrix}.$$

This process is not just a way to maintain independence of the columns of the fundamental solution. In [Os,Ma82] it was noted that this process can also be used to decouple the growing and decaying modes. Here we use the notion of consistency introduced in §1.2. Suppose that the ODE (2.1.1) is dichotomic and that Y is a fundamental solution with dichotomy projection $P = \begin{pmatrix} 0 & 0 \\ 0 & I_p \end{pmatrix}$. If the fundamental solution Z_1, with $Z_1(a) = Q_1$, is consistent with Y, then every fundamental solution Z_k, scaled such that $Z_k(x_k) = Q_k$, is consistent with Y. Indeed,

$$\begin{aligned} Z_1(x) &= \Phi_1(x)Q_1 \\ &= \Phi_2(x)\Phi_1(x_2)Q_1 \\ &= \Phi_2(x)Q_2U_1 \\ &\vdots \\ &= \Phi_k(x)Q_kU_{k-1}U_{k-2}\dots U_1 \\ &= Z_k(x)U_{k-1}\dots U_1 \end{aligned} \tag{2.1.13}$$

and the product of upper triangular matrices is again upper triangular. Hence if the first $(n-p)$ columns of Q_1 span the subspace of growing modes at $x = a$, then the first $(n-p)$ columns of Q_k will span the same subspace integrated up to $x = x_k$. In order to apply the consistency results from §1.2 we partition the matrices U_k as

(2.1.14) $$U_k = \begin{pmatrix} B_k & C_k \\ 0 & E_k \end{pmatrix},$$

with $B_k \in \mathbb{R}^{(n-p)\times(n-p)}$, $C_k \in \mathbb{R}^{(n-p)\times p}$ and $E_k \in \mathbb{R}^{p\times p}$.

2.1.15 Lemma

(2.1.15a) $\forall_{k<m}$: $|B_k^{-1}\ldots B_{m-1}^{-1}| \le \tilde{K}\exp(-\mu(x_m - x_k))$,

(2.1.15b) $\forall_{k>m}$: $|E_{k-1}\ldots E_m| \le \tilde{K}\exp(-\lambda(x_k - x_m))$,

with $\tilde{K}$ as in lemma 1.2.14.

Proof

The result follows from lemma 1.2.14 applied to Z_1 and the relation $Z_1(x_k) = Q_k U_{k-1} \cdot \cdot U_1$.

♦

This decoupling feature can be used in several ways. In [AsMaRu] it was pointed out how it can be used to compute the vectors s_k stably. Especially for separated BC the lower p elements can be obtained by forward substitution, after which the upper $(n-p)$ elements can be computed stably by backward substitution. In chapter 4 we will use the decoupling principle in a solution process for the equations arising for non-linear BVP's.

The discrete decoupling performed at the shooting points as described above, has a continuous analogue, named *invariant embedding*. Here we only give a brief outline of the idea, further details and theoretical background can be found e.g. in [vLo,Me73]. Since we need to partition matrices and vectors we use the following notation :

(2.1.16a) $$\forall_{C\in\mathbb{R}^{n\times n}} : C = \begin{pmatrix} C^{11} & C^{12} \\ C^{21} & C^{22} \end{pmatrix}, \quad C^{11}\in\mathbb{R}^{(n-p)\times(n-p)}, \quad C^{22}\in\mathbb{R}^{p\times p},$$

(2.1.16b) $$\forall_{z\in\mathbb{R}^n} : z = \begin{pmatrix} z^1 \\ z^2 \end{pmatrix} \begin{matrix} \updownarrow n-p \\ \updownarrow p \end{matrix}.$$

Instead of solving the BVP (2.1.1) in its original form, we now seek a continuous linear transformation $T(x)$ such that the new variable $w := T^{-1}y$ satisfies the ODE

(2.1.17) $$\dot{w} = U(x)w(x) + T^{-1}(x)q(x) \quad , \quad a<x<b \ ,$$

with $U(x)$ block upper triangular. Hence T has to satisfy the Lyapunov equation

(2.1.18) $\dot{T} = AT - TU$.

This transformation is beneficial to stability, if the differential equation

$$\dot{z} = U^{22}z \quad , \quad a<x<b \quad \text{and} \quad z\in C^1([a,b]\to\mathbb{R}^p) \ ,$$

contains no rapidly growing solution modes and the differential equation

$$\dot{z} = U^{11}z \quad , \quad a<x<b \quad \text{and} \quad z\in C^1([a,b]\to\mathbb{R}^{(n-p)}) \ ,$$

contains no rapidly decreasing solution modes. In that case the function $w(x)$ can be determined stably in a two-phase process, analogously to the discrete case, where first $w^2(x)$ is obtained by forward integration of

(2.1.19a) $\dot{w}^2 = U^{22}w^2 + (T^{-1}q)^2$

and thereafter the remaining elements of w are obtained by backward integration of

(2.1.19b) $\dot{w}^1 = U^{11}w^1 + U^{12}w^2 + (T^{-1}q)^1$.

The requirements on $U(x)$ will be fulfilled if the span of the first $(n-p)$ columns of $T(x)$ contains components of all growing solution modes of (2.1.1) (cf. the consistency of a fundamental solution §1.2).

A variant of invariant embedding is the *Riccati method.* Here the transformation T is required to have the special form

(2.1.20) $$T(x) = \begin{pmatrix} I_{n-p} & 0 \\ R(x) & I_p \end{pmatrix} \quad , \ R(x)\in\mathbb{R}^{p\times(n-p)} \quad , \ x\in[a,b] \ .$$

Now $U(x)$ is in block upper triangular form iff $R(x)$ satisfies the *Riccati differential equation*

(2.1.21) $\dot{R} = A^{21} + A^{22}R - RA^{11} - RA^{12}R$.

§2.2 Global methods

An idea for solving BVP's, which is conceptually different from the ones described in §2.1, is used in 'global' methods. Here one employs only one grid, comparable to the fine level grid in shooting. On this grid a discrete difference operator, approximating the original continuous one, is defined.

In this section we briefly describe a few simple one-step finite difference schemes and give a sketch of a stability proof for this method, because it shows some similarity between the finite difference method and the shooting method; moreover, some of the intermediate results are used later in this thesis. Finally we briefly describe the collocation method.

Let the set of points $\{x_k\}$ with

(2.2.1) $$a = x_1 < x_2 < \ldots < x_{N+1} = b,$$

define a grid on $[a,b]$. The *finite difference method* tries to generate a vector $y \in \mathbb{R}^{n(N+1)}$, subdivided into $N+1$ vectors of length n :

(2.2.2) $$y = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_{N+1} \end{pmatrix}, \quad y_k \in \mathbb{R}^n ,$$

such that y_k is an approximation to $y^*(x_k)$, where $y^*(x)$ is the solution of the BVP (2.1.1). A well-known one-step scheme, for instance, is the trapezoidal rule

(2.2.3) $$\frac{y_{k+1} - y_k}{h_k} = \frac{1}{2}[A(x_{k+1})y_{k+1} + A(x_k)y_k] + \frac{1}{2}[q(x_{k+1}) + q(x_k)] ,$$

where $h_k := x_{k+1} - x_k$. We define $h := \max \{ h_k \mid 1 \le k \le N \}$, a measure for the mesh size.
All one-step schemes can be written into the following generic form:

(2.2.4) $$\mathcal{L}_\pi y = q ,$$

with $q \in \mathbb{R}^{n(N+1)}$ depending both on $q(x)$ and β and the scheme used and $\mathcal{L}_\pi$ a linear operator from $\mathbb{R}^{n(N+1)}$ to $\mathbb{R}^{n(N+1)}$ defined by

$$(2.2.5)\qquad \forall_{y\in\mathbb{R}^{n(N+1)}} \;:\; \mathcal{L}_\pi y := \begin{pmatrix} S_1 & R_1 & & & \\ & S_2 & R_2 & & \\ & & \ddots & \ddots & \\ & & & S_N & R_N \\ B_a & & & & B_b \end{pmatrix} y .$$

The blocks S_k and $R_k \in \mathbb{R}^{n\times n}$ can be written as

$$(2.2.6a)\qquad S_k = -h_k^{-1} I_n + \Psi_1(x_k,h_k) ,$$

$$(2.2.6b)\qquad R_k = h_k^{-1} I_n + \Psi_2(x_k,h_k) ,$$

where Ψ_1 and Ψ_2 are method depending functions, which we assume to be bounded on $[a,b] \times [0,\bar{h}]$, for some $\bar{h} > 0$.

Next we want to estimate the error in the approximation y of the solution $y^*(x)$. Since y and $y^*(x)$ are incomparable quantities, we introduce a projection $\wp \in C(C([a,b] \to \mathbb{R}^n) \to \mathbb{R}^{n(N+1)})$, defined by

$$(2.2.7)\qquad \forall_{z\in C([a,b]\to\mathbb{R}^n)} \;:\; \wp z := \begin{pmatrix} z(x_1) \\ z(x_2) \\ \vdots \\ z(x_{N+1}) \end{pmatrix} .$$

Now we would like to estimate the *global discretization error* $e(h)$,

$$(2.2.8)\qquad e(h) := \wp y^* - y$$

and establish *convergence* of the method, i.e.

$$(2.2.9)\qquad \lim_{h\downarrow 0} | e(h) |_\infty = 0 .$$

This can be done indirectly : assume that the method used is *consistent*, i.e.

$$(2.2.10)\qquad \exists_{C>0}\, \exists_{p\in\mathbb{N}}\, \forall_{h<\bar{h}} \;:\; | \tau_h[y^*] |_\infty \le C h^p ,$$

where the *local discretization error* $\tau_h[y^*]$ is defined by

$$(2.2.11)\qquad \tau_h[y^*] := \mathcal{L}_\pi \wp y^* - \mathcal{L}_\pi y .$$

Then convergence follows if $\mathcal{L}_\pi$ has a bounded inverse, i.e. if the method is *stable*. In the literature several stability proofs can be found, see e.g. [Ke76,AsMaRu]. Essentially they are all based on the fact that the matrix of (2.2.5) is closely related the matrix occurring in

multiple shooting with, additionally, discretization errors. To be more precise, if $Y(x)$ is a fundamental solution of (2.1.1), then $-R_k^{-1}S_k$ is an approximation of $Y(x_{k+1})Y^{-1}(x_k)$. Since every column of $Y(x)$ is a solution of the homogenous part of (2.1.1), consistency implies that

$$|\mathcal{G}_\pi \mathcal{Q} Y - q|_\infty = O(h^p) \quad , \text{ with } \quad q = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ BY \end{pmatrix}, \tag{2.2.12}$$

(i.e. solve (2.2.4) for n different inhomogeneities simultaneously). Therefore

$$S_k Y(x_k) + R_k Y(x_{k+1}) = O(h^p) , \tag{2.2.13}$$

and since $| R_k^{-1} | = O(h)$ this yields

$$-R_k^{-1} S_k = Y(x_{k+1})Y^{-1}(x_k) + O(h^{p+1}) . \tag{2.2.14}$$

This induces the following relation between $\mathcal{G}_\pi$ and the Jacobian occurring in the multiple shooting method :

$$\text{diag}(-R_1^{-1}, -R_2^{-1}, \ldots, -R_N^{-1}, I_n)\, \mathcal{G}_\pi = J + E , \tag{2.2.15}$$

with E a block diagonal matrix with $| E | = O(h^{p+1})$ and

$$J = \begin{pmatrix} Y_1(x_2) & -I_n & & & \\ & Y_2(x_3) & -I_n & & \\ & & \ddots & \ddots & \\ & & & Y_N(x_{N+1}) & -I_n \\ B_a & & & & B_b \end{pmatrix}. \tag{2.2.16}$$

The matrix J is identical to the Jacobian of $f(s)$ in the multiple shooting process. The norm of its inverse is $N\kappa_{lin}$ (see Appendix B). Now we can derive

$$\begin{aligned} \mathcal{G}_\pi^{-1} &= (J+E)^{-1}\text{diag}(-R_1^{-1}, -R_2^{-1}, \ldots, -R_N^{-1}, I_n) \\ &= (I_{n(N+1)} + J^{-1}E)\, J^{-1}\, \text{diag}(-R_1^{-1}, -R_2^{-1}, \ldots, -R_N^{-1}, I_n) \end{aligned}$$

$\Rightarrow$

$$|\mathcal{G}_\pi^{-1}|_\infty \le (1 + O(h^p))(\kappa_{lin}(b-a) + O(h)) = \kappa_{lin}(b-a) + O(h) .$$

Another, though related, class of global solution methods is *collocation*. Here again a grid is chosen on the interval $[a,b]$. Now the solution of the BVP is approximated by a func-

tion $y_\pi \in C([a,b] \to \mathbb{R}^n)$, which is a polynomial of degree m, for some fixed $m \in \mathbb{N}$, on every subinterval $[x_k , x_{k+1}]$. The continuity at grid points plus the boundary conditions give Nn relations for the $N(m+1)n$ unknown polynomial coefficients. In order to form the remaining relations we choose canonical points $\{\rho_j\}$, $1 \le j \le m$, with

(2.2.17) $$0 \le \rho_1 < \rho_2 < \ldots < \rho_m \le 1$$

and require that y_π satisfies the ODE (2.1.1a) at the points

(2.2.18) $$x_{ij} = x_i + h_i \rho_j \quad , \quad j \in \{1,.,m\} \quad , \quad i \in \{1,.,N\} ,$$

i.e.

(2.2.19) $$\dot{y}_\pi(x_{ij}) = A(x_{ij}) y_\pi(x_{ij}) + q(x_{ij}) \quad , \quad j \in \{1,.,m\} \quad , \quad i \in \{1,.,N\} .$$

It was first shown by Weiss [We] that an implicit Runge-Kutta scheme can be formed with canonical points ρ_j, such that y_π restricted to $[x_k , x_{k+1}]$ is the interpolant of the auxiliary RK points y_{ij} (estimates of $y(x_{ij})$). Hence the consistency and stability results of Runge-Kutta schemes (which are one-step finite difference schemes) apply and no additional analysis in this respect is necessary. The use of Gaussian points as canonical points is appealing since it gives a local truncation error of $O(h^{2m})$.

Thus far we have not considered higher order differential equations, mainly because they can easily be transformed into first order ODE (cf.[AsRu]). However, especially for collocation methods one can make an implementation without reducing it to a first order ODE. This has been considered in [AsChRu,BaAs] and implemented in the package COLNEW, which we have used for several numerical experiments in Chapter 5.

Consider the k^{th} order BVP for $y \in C^{(k-1)}([a,b] \to \mathbb{R})$

(2.2.20a) $$y^{(k)} = \sum_{j=0}^{k-1} c_j(x) y^{(j)} + q(x) \quad , \quad a < x < b \quad , \quad m \ge 1 ,$$

(2.2.20b) $$B_a \begin{pmatrix} y(a) \\ \vdots \\ y^{(k-1)}(a) \end{pmatrix} + B_b \begin{pmatrix} y(b) \\ \vdots \\ y^{(k-1)}(b) \end{pmatrix} = \beta .$$

Its solution can be approximated by a function $y_\pi \in C^{(k-1)}([a,b] \to \mathbb{R})$, which is a polynomial of degree $k+m-1$ on every subinterval. Now again the coefficients follow from the $(N-1)k$ continuity relations, the k boundary conditions and the Nm requirements

(2.2.21) $$y_\pi^{(k)}(x_{ij}) = \sum_{l=0}^{m-1} c_l(x_{ij}) y_\pi^{(l)}(x_{ij}) + q(x_{ij}) .$$

An important issue in the actual implementation of global methods is the grid choice. During the process decisions about the redistribution and refinement of the grid have to be made, in order to equidistribute the error and decrease it until the accuracy requirements are met. This item is investigated by several authors, see e.g [Ru].

Comparing global methods with IVP-methods we see that the first type encounters less trouble from exponentially growing modes, partly because only relatively small subintervals are used, partly because only global methods allow for discretization schemes that are capable to conserve the dichotomic character of the solution space.
From a philosophical point of view one could say that IVP-methods use compactification on the fine grid, retaining information on the coarse grid only. Consequently using these methods requires less memory space and leads to the solution of smaller linear systems than using global methods, where information on every fine grid point is retained.

Finally we mention that parallelization of these solution methods consists of two parts: parallelization of the assembly of the large matrix and parallel solution of the resulting linear system. For the latter several stable methods have been developed, see e.g. [AsPC, PaGl,Wr]. For IVP-methods parallelization of the assembly is straightforward since every subinterval can be assigned to a different processor. However, for global methods one needs a proper splitting of the interval into subintervals, especially for non-linear BVP's. We address this issue in Chapter 5.

§2.3 Solution methods for non-linear BVP's

In the previous sections we described solution methods for linear BVP's. Of course the same ideas can be used to solve non-linear BVP's. However, the equation resulting from discretization will generally be non-linear. In this section we will briefly describe the adapted solution methods for non-linear BVP's and consider in particular the use of Newton's method to solve the arising non-linear equations. More specifically we estimate the size of the convergence domain according to the Newton-Kantorovich theorem, see e.g. [OrRh],[DeHe].

Consider the non-linear BVP

$$\text{(2.3.1)} \qquad \begin{cases} \dot{y} = h(x,y(x)) & , \ a<x<b \ , \\ g(y(a),y(b)) = 0 \ . \end{cases}$$

The application of the multiple shooting method to it, starts out in the same way as the application to a linear BVP, i.e. we choose a grid

$$\text{(2.3.2)} \qquad a = x_1 < x_2 < \ldots < x_{N+1} = b$$

and define locally non-linear IVP's :

$$\text{(2.3.3)} \qquad \begin{cases} \dot{y} = h(x,y(x)) & , \ x_k < x < x_{k+1} \ , \ 1 \le k \le N \ , \\ y(x_k) = s_k & , \ s_k \in \mathbb{R}^n \ . \end{cases}$$

The solutions of the local problems, if existing, are denoted by $y_k(x;s_k)$ and $y(x;s)$ is the function, defined on $[a,b]$, that is equal to $y_k(x;s_k)$ on $(x_k , x_{k+1}]$ and satisfies $y(a;s) = y_1(a;s_1)$. The unknown vectors s_k have to be determined such that the local solutions together form a continuous function that satisfies the boundary conditions. Hence they have to be a solution of (2.1.6),(2.1.7). However, this time $f(s)$ is a non-linear function. Its solution will be denoted by s^* and the corresponding solution of (2.3.1) will be denoted by $y^*(x) := y(x;s^*)$.

An often used solution method is Newton's method : let s^0 be the initial guess for the shooting vectors. Then the next iterates are determined by

$$\text{(2.3.4)} \qquad s^{j+1} = s^j + \xi \ , \ j \ge 0 \ ,$$

with ξ the solution of

$$\text{(2.3.5)} \qquad J(s^j)\xi = -f(s^j)$$

and $J(s)$ the Jacobian of $f(s)$. The derivatives $\dfrac{\partial y_k}{\partial s_k}$ can be determined in a special way, viz.

(2.3.6a) $$\frac{\partial}{\partial x}\frac{\partial y_k(x;s_k)}{\partial s_k} = \frac{\partial}{\partial s_k} h(x,y_k(x;s_k)) = \left.\frac{\partial h(x,v)}{\partial v}\right|_{v=y_k(x;s_k)} \frac{\partial y_k(x;s_k)}{\partial s_k} ,$$

with

(2.3.6b) $$\frac{\partial y_k(x_k;s_k)}{\partial s_k} = I_n .$$

i.e. $\frac{\partial y_k}{\partial s_k}$ is a fundamental solution of the linearized ODE at $y_k(x;s_k)$. Since we will often refer to this linearization, we introduce some simplifying notation.

2.3.7 Definition

The derivative of $h(x,y)$ at $y_k(x;s_k)$ is

(2.3.7a) $$L_k(x;s_k) := \left.\frac{\partial}{\partial v} h(x,v)\right|_{v=y_k(x;s_k)} , \quad x_k < x < x_{k+1} ,$$

and the derivative of the boundary conditions are

(2.3.7b) $$B_a(s) := \left.\frac{\partial g(u,s_{N+1})}{\partial u}\right|_{u=s_1} \quad \text{and} \quad B_b(s) := \left.\frac{\partial g(s_1,v)}{\partial v}\right|_{v=s_{N+1}} .$$

This leads to the local linearized systems

(2.3.7c) $$\begin{cases} \dot{z} = L_k(x;s_k)z & , x_k < x < x_{k+1} , \ k = 1,..,N , \\ I_n z(x_k) = \beta . \end{cases}$$

♦

The Jacobian of $f(s)$ for non-linear BVP's has the same structure as its counterpart for linear BVP's, only in the first case the non-zero blocks are fundamental solutions of the linearized ODE and not of the original one.

Inherent to a BVP is that the underlying ODE may contain exponentially growing modes. For the well-conditioning of the problem it is vital that those modes are controlled at the endpoint. However, on the subintervals only initial conditions are imposed. Due to this, a method based on shooting encounters several drawbacks. We have already seen for linear problems that computational errors may be increased considerably. This may of course also occur for non-linear problems. For the latter class of BVP's two other problems may

be encountered as well. Firstly the solution of a local IVP (2.3.3) may not exists over the entire subinterval. And secondly, since $f(s)$ may be overly sensitive to changes in the vector s in certain directions, the convergence domain of Newton's method might be small. To demonstrate this we will estimate the convergence domain according to the Newton-Kantorovich theorem, see e.g [OrRh]. The theorem guarantees convergence of Newton's method with initial guess s^0, if the product $\beta\gamma \mid J^{-1}(s^0)f(s^0) \mid < 0.5$; here γ is the Lipschitz constant of $J(s)$ near s^* and β is an upper bound on $\mid J^{-1}(s^0) \mid$ (see Appendix D for a precise formulation). We are aware that generally the convergence domain may be larger, but the results of the theorem give an indication about the performance of the method.

2.3.8 Assumption

Let D_y be a convex neighbourhood of $y^*(x)$ such that

- *$y^*(x)$ is the unique solution of (2.3.1) in D_y,*
- *the function $g(u,v)$ is twice continuously differentiable in both variables and $h(x,y)$ is twice differentiable with respect to y and all partial derivatives are bounded by a moderate constant, say C_{gh}.*

Let the set D_s be defined by $D_s := \{\, s \in \mathbb{R}^{n(N+1)} \mid y(x;s) \in D_y \,\}$. Finally assume that for all $1 \le k \le N$ there is a constant κ_k such that

$$\forall_{s \in D_s} \;:\; \text{the conditioning constant } \kappa(\varepsilon; y_k(x;s_k)) \text{ of (2.3.3)} \text{ is bounded by } \kappa_k, \text{ with } \varepsilon = \max\{\, |s_k - \sigma_k| \mid s,\sigma \in D_s \,\} .$$

♦

2.3.9 Theorem

$$\text{(2.3.9a)} \qquad \forall_{s,\sigma \in D_s} \;:\; \frac{|\, J(s) - J(\sigma) \,|_\infty}{|\, s - \sigma \,|_\infty} = O\left(C_{gh} \max_k \left[\kappa_k^3 (x_{k+1} - x_k)\right]\right) .$$

Proof

Let $s,\sigma \in D_s$. Furthermore let $Y_k(x;s_k)$ and $G_k(x,t;s_k)$ denote the fundamental solution and the Green's function of (2.3.7c) respectively, where the fundamental solution is scaled such that $Y_k(x;s_k) = I_n$. Now for each of the first $(N-1)$ block rows of $J(s)$ we estimate the difference in the fundamental solutions

$$Y_k(x;\sigma_k) = Y_k(x;s_k) + \int_{x_k}^{x_{k+1}} G_k(x,t;s_k)\,(L_k(t;\sigma_k) - L_k(t;s_k))\, Y_k(t;\sigma_k)\, dt .$$

Since $\mid L_k(x;s_k) - L_k(x;\sigma_k) \mid \le C_{gh} \mid y_k(x;s_k) - y_k(x;\sigma_k) \mid \le C_{gh}\kappa_k \mid s_k - \sigma_k \mid$,
this yields

$$\max_x |\, Y_k(x;s_k) - Y_k(x;\sigma_k) \,| \le \kappa_k^2 (x_{k+1}-x_k) \max_t |\, L_k(t;s_k) - L_k(t;\sigma_k) \,|$$
$$\le C_{gh} \kappa_k^3 (x_{k+1}-x_k) \,|\, s_k - \sigma_k \,| \;.$$

For the last block row we have the estimate

$$|\, B_a(s) - B_a(\sigma) \,| \le C_{gh} \,|\, s_{N+1} - \sigma_{N+1} \,| + C_{gh} \,|\, s_1 - \sigma_1 \,| \le 2 C_{gh} \,|\, s - \sigma \,| \;.$$

For the block containing the endpoint conditions an analogous upper bound can be found. Combining these estimates we get (2.3.9a).

♦

For an exponentially dichotomic BVP with the strongest growing mode growing like $e^{\mu x}$, one can easily prove that $\kappa_k = O(\exp(\mu(x_{k+1}-x_k)))$. Hence taking smaller subintervals does diminish the Lipschitz constant of $J(s)$. However, one has to solve a larger system.

Furthermore we need to estimate $|\, J^{-1}(s^0) \,|$ and $|\, J^{-1}(s^0) f(s^0) \,|$. For linear BVP's the inverse of the Jacobian is essentially bounded by N times the conditioning constant of the global BVP (see Appendix B, Corr. B.6). Hence, $|\, J^{-1}(s^*) \,|$ will be of the order of magnitude of $N \cdot \lim_{\varepsilon \downarrow 0} \kappa(\varepsilon; y(x;s^*))$ and by a continuity argument $|\, J^{-1}(s) \,|$ will be reasonably bounded for s sufficiently close to s^*. However, this bound may become large, because $y(x;s)$ will generally be discontinuous, possibly causing a disruption of the dichotomy behaviour of the linearized BVP and thus have a negative effect on the conditioning.
From the above considerations we see that the convergence of Newton's method to solve $f(s) = 0$ may be jeopardized by a large Lipschitz constant of the Jacobian, i.e. the influence of the second order term in the expression

$$f(s^*) = f(s) + J(s)(s^* - s) + \int_0^1 [J(s + t(s^* - s)) - J(s)] (s^* - s)\, dt$$

may not be negligible.

An alternative for the mulitple shooting method, which we study in more detail in chapter 5, is to use local boundary value problems instead of local initial value problems. The function $f(s)$ will again consist of continuity requirements and the global boundary conditions, but now the Jacobian will consist of fundamental solutions that reflect the conditioning of the local BVP's.

Unlike multiple shooting, the *finite difference method* for non-linear BVP's differs from the one for linear BVP's almost right from the start : let $\{\, x_k \mid k = 1, \ldots, N+1 \,\}$ be a grid on $[a,b]$ with

(2.3.10) $a = x_1 < x_2 < \ldots < x_{N+1} = b$

and let $y \in \mathbb{R}^{n(N+1)}$ be the concatenation of $N+1$ vectors $y_k \in \mathbb{R}^n$:

(2.3.11) $y^\top = (y_1^\top, y_2^\top, \ldots, y_{N+1}^\top)$

Now for a one-step scheme the vectors y_k have to satisfy

(2.3.12) $h_k^{-1}(y_{k+1} - y_k) = \Upsilon(y_k, y_{k+1}, x_k, h_k)$, $k \in \{1, .., N\}$,

where Υ describes the method used; i.e. we seek the solution of $\mathcal{N}_\pi[y] = 0$, with the nonlinear operator $\mathcal{N}_\pi : \mathbb{R}^{n(N+1)} \to \mathbb{R}^{n(N+1)}$ defined by

(2.3.13) $$(\mathcal{N}_\pi[y])_k = \begin{cases} h_k^{-1}(y_{k+1} - y_k) - \Upsilon(y_k, y_{k+1}, x_k, h_k) & , \ k \neq N+1 \ , \\ g(y_1, y_{N+1}) & , \ k = N+1 \ . \end{cases}$$

Analogously to the linear case we define *consistency, stability* and *convergence* as follows, cf. [AsMaRu,Ke76].

2.3.14 Definition
The local discretization error $\tau_h[y^*]$ *is defined by*

(2.3.14a) $\tau_h[y^*] := \mathcal{N}_\pi[\wp y^*(x)]$.

The finite difference scheme is consistent *of order p if*

(2.3.14b) $\exists_{h_0>0} \ \exists_{C>0} \ \forall_{\text{grids}, h \le h_0} \ : \ |\tau_h[y^*]| \le C h^p$.

The scheme is stable *at $y^*(x)$ if there is a ball*

$S_{\rho,\pi}(y^*) := \{ u \in \mathbb{R}^{n(N+1)} \mid |u - \wp y^*|_\infty \le \rho \}$ *around $\wp y^*$ and a constant K of moderate size such that*

(2.3.14c) $\exists_{h_0} \ \forall_{\text{grids}, h \le h_0} \ \forall_{u,v \in S_{\rho,\pi}} \ : \ |u - v|_\infty \le K \, |\mathcal{N}_\pi[u] - \mathcal{N}_\pi[v]|_\infty$.

Finally the scheme is convergent *if*

(2.3.14d) $\lim_{h \downarrow 0} |\wp y^* - y|_\infty = 0$.

♦

As in the linear case convergence is implied by consistency and stability. Generally consistency of a method can easily be established; stability is sometimes more difficult to prove. The following lemma gives sufficient conditions (on the linearization of $\mathcal{N}_\pi$) for stability. The derivative of $\mathcal{N}_\pi$ with respect to y will be denoted by $\mathcal{L}_\pi[y]$ (for a linear

BVP this is equal to the definition in §2.2). Hence with

$$S_k = -h_k^{-1} I_n - \left.\frac{\partial \Upsilon(u, y_{k+1}, x_k, h_k)}{\partial u}\right|_{u=y_k} \tag{2.3.15a}$$

and

$$R_k = h_k^{-1} I_n - \left.\frac{\partial \Upsilon(y_k, v, x_k, h_k)}{\partial v}\right|_{v=y_{k+1}}, \tag{2.3.15b}$$

$$\mathcal{L}_\pi[y] = \begin{pmatrix} S_1 & R_1 & & & \\ & S_2 & R_2 & & \\ & & \ddots & \ddots & \\ & & & S_N & R_N \\ B_a & & & & B_b \end{pmatrix}. \tag{2.3.16}$$

2.3.17 Lemma (see e.g. [AsMaRu,Ke76])
Let $S_{\rho,\pi}(y^*)$ be as in definition 2.3.14. *Let $\mathcal{L}_\pi[y]$ be consistent and stable for all $y \in S_{\rho,\pi}(y^*)$ and let the partial derivatives of Υ with respect to its first and second argument be bounded in the same tube. Furthermore, let there be a 'partial' Lipschitz constant K_L such that*

$$\forall_{u \in S_{\rho,\pi}(y^*)} \quad : \quad |\mathcal{L}_\pi[u] - \mathcal{L}_\pi[\Phi y^*]|_\infty < K_L \, |u - \Phi y^*|_\infty . \tag{2.3.17a}$$

Then $\mathcal{N}_\pi$ is stable on $S_{\rho,\pi}(y^*)$ and $|\mathcal{L}_\pi^{-1}[y]|$ is uniformly bounded on $S_{\rho,\pi}(y^*)$.
♦

In §2.2 we have shown that $|\mathcal{L}_\pi^{-1}[\Phi y^*]|_\infty$ is bounded by the conditioning constant of the linearization of (2.3.1) at $y^*(x)$ for h sufficiently small. Now a continuity argument yields stability for the linearization at neighbouring vectors $\mathbf{y}$. Hence the first conditions of the lemma can be met for well conditioned BVP's.
For multiple shooting we found that the Lipschitz constant of the linearization ($\mathcal{L}_\pi$) could become large. However, recall from §2.2 that $-h_k^{-1} S_k^{-1}$ and $h_k^{-1} R_k^{-1}$ approximate a fundamental solution of the linearized BVP at $x = x_k$, $x = x_{k+1}$, respectively. The matrices $-h_k S_k$ and $h_k R_k$ are, according to lemma 1.2.16, approximations of the fundamental solution $\Phi(x)$ of

$$\dot{z} = -\left(\left.\frac{\partial h}{\partial y}(x,y)\right|_{y=y^*(x)}\right)^\top . z \qquad , \quad x_k < x < x_{k+1} , \tag{2.3.18}$$

with Φ satisfying the BC

$$I_n \cdot \Phi(x_k) + I_n \cdot \Phi(x_{k+1}) = 2I_n + O(h_k) \ .$$

Hence the Lipschitz constant of $\mathfrak{L}_\pi$ can be related to the conditioning constant of this BVP. Since these boundary conditions are non-separated, a poor condition of the BVP can be caused only by rotational activity of the different solution modes. However, this problem can be overcome if we choose the grid sufficiently fine.

3 Davidenko-like differential equations and a special integration method

We have seen in chapter 2 that the convergence domain of Newton's method when applied to the non-linear equation encountered in multiple shooting, may be small. And apparently this is not the only type of problems, where Newton's method does not perform flawlessly, for in literature several alternative solution methods can be found. One class of alternative solution methods is (parameter) continuation : a series of non-linear equations is solved as a (possibly artificial) parameter is varied, using the solution of the previous problem as initial guess for the next one, see e.g. [Me68,Was,OrRh,KuHl,RoSh,DePeRe]. An idea that is theoretically related, though different in implementation, is to embed the non-linear equation into a differential equation, see e.g. [Wa, OrRh,Da]. Indeed, Newton's method can be considered as the application of the explicit Euler integration method with step size 1 on the IVP

$$\text{(3.0.1)} \qquad \begin{cases} \dfrac{dx}{dt} = -J^{-1}(x(t))f(x(t)) & , \quad t>0 \ , \\ x(0) = x_0 \ . \end{cases}$$

This differential equation is often called Davidenko's equation. In this chapter we look at a variant of this method, viz. solving the IVP

$$\text{(3.0.2)} \qquad \begin{cases} \dfrac{dx}{dt} = M(x(t))f(x(t)) & , \quad t>0 \ , \\ x(0) = x_0 \ . \end{cases}$$

in order to obtain a zero x^* of $f(x)$. We will derive sufficient conditions on $M(x)$ to guarantee that (3.0.2) is asymptotically stable at $x = x^*$. Moreover, we introduce an implicit integration method for the IVP, that is computationally cheaper than implicit Euler, but that does have its asymptotic stability properties.

§3.1 Davidenko-like equations

In this chapter we introduce and investigate a path following method for solving non-linear equations. The setting will be quite general and not specifically aimed at the equations arising in multiple shooting (this will be the subject of chapter 4). Therefore we consider in this chapter a function $f \in C^2(\mathbb{R}^m \rightarrow \mathbb{R}^m)$ with a zero x^*. The Jacobian of $f(x)$ is denoted by $J(x)$.

Newton's method for finding a zero of $f(x)$ reads

$$x^{j+1} = x^j - J^{-1}(x^j) f(x^j) \quad , \quad j \geq 0 \quad , \quad x^0 \in \mathbb{R}^m \; . \tag{3.1.1}$$

The justification of the method lies in the relation

$$x^* - x = J^{-1}(x)(f(x^*) - f(x)) + O(|J^{-1}(x) f''(x)| \, |x^* - x|^2) \; . \tag{3.1.2}$$

Here we see that the new update x^{j+1} will be closer to x^*, only if the second order term is small, i.e. the convergence area may be small either if the Jacobian is (nearly) singular or if the Jacobian has a large Lipschitz constant. This is not only a theoretical consideration; when solving BVP's with exponentially growing modes with the multiple shooting method, we actually encountered problems with Newton's method (see chapter 4).

Hence we investigate other solution methods for non-linear equations. It has been noted by many authors, see e.g. [Da,KuHl,OrRh], that Newton's method can be considered as a discretization (with the explicit Euler scheme and step size 1) of the continuous initial value problem :

$$\begin{cases} \dfrac{dx}{dt} = -J^{-1}(x) f(x) \quad , \quad t > 0 \; , \\ x(0) = x^0 \; , \end{cases} \tag{3.1.3}$$

i.e. an artificial time dependency of x is introduced. The latter differential equation, often called *Davidenko's equation*, see [Da], is sometimes referred to as the *closure* of Newton's method. Notice that discretizing (3.1.3) with the explicit Euler scheme and a step size less than 1, yields *damped Newton*.

This view upon Newton's method induces the idea to look at a larger class of initial value problems :

$$\begin{cases} \dfrac{dx}{dt} = M(x) f(x) \quad , \quad t > 0 \; , \\ x(0) = x^0 \; , \end{cases} \tag{3.1.4}$$

with $M(x) \in C(\mathbb{R}^m \rightarrow \mathbb{R}^{m \times m})$. The matrix function $M(x)$ is called the *preconditioner*. It is ob

vious that any zero x^* of $f(x)$ induces a constant solution $x \equiv x^*$ of (3.1.4) and that vice versa any constant solution of the IVP corresponds to a zero of $f(x)$, if $M(x^*)$ is non-singular.

Embedding techniques are also used for solving problems in physics and chemistry. In particular elliptic problems can be embedded into a dissipative time-dependent (hyperbolic) partial differential equation, without any preconditioner (or $M(x) \equiv 1$). These embedding methods, which are often referred to as *false transient* or *time stepping*, have proven to be very helpful in solving difficult problems in chemical engineering and combustion, see e.g. [KuHl,SmMiKe].

Solving the IVP (3.1.4) in order to obtain a zero of $f(x)$, is appropriate, only if the ODE is *asymptotically stable*, i.e.

(3.1.5a) $\exists_{\varepsilon_0}\ \forall_{0<\varepsilon<\varepsilon_0}\ \exists_{\delta>0}\ \forall_{x^0,\,|x^0-x^*|<\delta}$:
the solution $x(t)$ of (3.1.4) with $x(0) = x^0$ satisfies
$\forall_{t\geq 0}$: $|x(t)-x^*| < \varepsilon \ \wedge \ \lim_{t\to\infty} x(t) = x^*$.

A stronger stability requirement is *local contractivity*, which is defined by

(3.1.5b) $\exists_{\delta>0}\ \forall_{x^0,\,|x^0-x^*|<\delta}$: $|x(t)-x^*|$ decreases monotonically to zero,
with $x(t)$ the solution of (3.1.4) with $x(0) = x^0$.

The remainder of this section we will investigate under what conditions the preconditioner satisfies this stability requirement.

A useful concept in this case is the *one-sided Lipschitz constant* and related to it the *logarithmic norm*. Both will be formulated for the following general ODE :

(3.1.6) $\dot{x} = h(t,x)$, $t>0$.

Let $< . , . >$ denote the Euclidian inner product and $|\ .\ |_2$ the corresponding vector norm in $\mathbb{R}^m$.

3.1.7 Definition
Let $T > 0$. Let the functions $\varphi \in C([0,T] \to \mathbb{R}^m)$ and $\psi \in C([0,T] \to (0,\infty))$ define a family of balls, depending on t :

(3.1.7a) $D(t) := \{\, \xi \in \mathbb{R}^m \mid |\xi - \varphi(t)| \leq \psi(t) \,\}$, $0 \leq t \leq T$.

A piecewise continuous function $\nu(t) : [0,T] \to \mathbb{R}$, is a one-sided Lipschitz constant *of* (3.1.6) *on $D(t)$ if*

(3.1.7b) $\forall_{t\in[0,T]}\ \forall_{x,y\in D(t)}\ :\ <h(t,x)-h(t,y),x-y> \le \nu(t)\,|x-y|_2^2$.

♦

The one-sided Lipschitz constant is not unique : if ν satisfies the definition, then any piecewise continuous function $\tilde{\nu}$ with $\tilde{\nu}(t) \ge \nu(t)$, is also a one-sided Lipschitz constant. Moreover, if the function $h(t,x)$ is Lipschitz continuous with respect to x, say with constant L, then $\nu \equiv L$ is a one-sided Lipschitz constant. The one-sided Lipschitz constant is for instance used in the following stability results, see [DeVe].

3.1.8 Theorem

Let $x(t)$ and $\tilde{x}(t)$ be solutions of the ODE (3.1.6) *and assume that*

(3.1.8a) $\forall_{t\in[0,T]}\ :\ x(t)\in D(t)\ \wedge\ \tilde{x}(t)\in D(t)$.

Let $\nu(t)$ be a one-sided Lipschitz constant of (3.1.6) *on $D(t)$. Then*

(3.1.8b) $\forall_{0\le t_1\le t_2\le T}\ :\ |x(t_2)-\tilde{x}(t_2)|_2 \le \exp\left(\int_{t_1}^{t_2}\nu(\tau)\,d\tau\right)|x(t_1)-\tilde{x}(t_1)|_2$.

♦

The ODE (3.1.6) is locally contractive, if there is a negative one-sided Lipschitz constant and asymptotically stable if

$$\lim_{T\to\infty}\int_0^T \nu(\tau)\,d\tau = -\infty .$$

In the literature, special attention has been given to the autonomous linear ODE in connection with this concept. So consider

(3.1.9) $\dot{x} = Ax \quad , \ t\ge 0$,

with $A\in\mathbb{R}^{m\times m}$. In this case all solutions are of the form

(3.1.10) $x(t) = \exp(At).\xi$, for some $\xi\in\mathbb{R}^m$.

Hence we can derive a relation, similar to (3.1.8b), for any two solutions $x(t)$ and $\tilde{x}(t)$:

(3.1.11) $\forall_{0\le t_1\le t_2\le T}\ :\ |x(t_2)-\tilde{x}(t_2)| \le |\exp(A(t_2-t_1))|\,|x(t_1)-\tilde{x}(t_1)|$.

So instead of the one-sided Lipschitz constant, which is defined for inner product norms only, we can use a bound on $|\exp(At)|$ for all vector norms. According to e.g. [St], the minimum of $\{\ \theta\ |\ \forall_{t\ge0}\ :\ |\exp(At)|\le e^{\theta t}\ \}$ is equal to $\lim_{h\downarrow 0}\dfrac{|I_m+hA|-1}{h}$. And this

formula is generally used as definition of logarithmic norm :

3.1.12 Definition
For any matrix $A \in \mathbb{R}^{m\times m}$ *the* logarithmic norm $\mu[A]$ *with respect to* $|\ .\ |$ *is defined by*

$$\mu[A] := \lim_{h\downarrow 0} \frac{|I_m + hA| - 1}{h} . \tag{3.1.12a}$$

♦

The logarithmic norm with respect to the Euclidian norm, denoted by $\mu_2[A]$, can be related to the one-sided Lipschitz constant; since

$$\mu_2[A] = \max_{\xi\neq 0} \frac{<A\xi,\xi>}{<\xi,\xi>} , \tag{3.1.13}$$

see e.g. [Dah], $\mu_2[A]$ is the smallest one-sided Lipschitz constant of (3.1.9).

In Appendix C we have gathered a collection of properties of the logarithmic norm that can be found in the literature. Here we mention only two properties that relate the logarithmic norm to the eigenvalues of a matrix.

3.1.14 Property
Let $A \in \mathbb{R}^{m\times m}$.
(*i*) *For all eigenvalues* λ *of* A : $\mathrm{Re}(\lambda) \le \mu[A]$, *in any vector norm.*
(*ii*) $\mu_2[A] = \max\{\lambda \mid \lambda$ *eigenvalue of* $\frac{1}{2}(A+A^T)\}$.
♦

The logarithmic norm can be used in a stability result, similar to theorem 3.1.8.

3.1.15 Theorem ([Dah])
Let $|\ .\ |$ *be a given norm. Let* $\nu : [0,T] \to \mathbb{R}$ *be a piece wise continuous function satisfying*

$$\forall_{t\in[0,T]}\ \forall_{\xi\in D(t)} : \mu[\frac{\partial h}{\partial x}(t,x)\Big|_{x=\xi}] \le \nu(t) \tag{3.1.15a}$$

Then for any two solutions x *and* $\bar{x}$ *of* (3.1.6) *that lie in* $D(t)$ *for all* $t\in[0,T]$,

$$\forall_{0\le t_1\le t_2\le T} : |x(t_2)-\bar{x}(t_2)| \le \exp(\int_{t_1}^{t_2} \nu(\tau)d\tau)\,|x(t_1)-\bar{x}(t_1)| . \tag{3.1.15b}$$

♦

Let us now return to the starting point of this section. We want to establish asymptotic stability of the IVP (3.1.4) around $x = x^*$. From Theorems 3.1.8 and 3.1.15 we see that a sufficient condition for this is that either

(3.1.16a) $$\frac{<x-x^*, M(x)f(x)>}{|x-x^*|_2^2}$$

or

(3.1.16b) $$\mu[\frac{d}{dx}(M(x)f(x))]$$

is bounded by a negative constant on a neighbourhood of x^*. However, in chapter 4 we will form a preconditioner for the function arising in multiple shooting (see (2.1.7)), for which $\mu[M(x)J(x)]$ is negative on a neighbourhood of x^*.
Let $B(x^*;R)$ denote the set $\{ x \in \mathbb{R}^m \mid |x-x^*| \leq R \}$.

3.1.17 Assumption
Suppose there is a ball $B(x^*;R)$ such that

(*i*) $\exists_{\alpha>0} \quad \forall_{x \in B(x^*,R)} : \mu_2[M(x)J(x)] \leq -\alpha$.

(*ii*) *The functions $f(x)$, $J(x)$ and $M(x)$ are bounded on $B(x^*,R)$ by constants C_f, C_J and C_M, respectively.*

(*iii*) *The functions $J(x)$ and $M(x)$ are Lipschitz continuous on $B(x^*,R)$ with Lipschitz constants L_J and L_M, respectively.*

3.1.18 Definition
The constant $\hat{C}$ is defined by

(3.1.18a) $$\hat{C} := \max_{x \in B(x^*,R)} \frac{<x-x^*, M(x)\int_0^1 [J(x^*+t(x-x^*))-J(x)](x-x^*)dt>}{|x-x^*|^3} .$$

♦

Based on the proof of lemma 3.1.8, we can derive the following stability lemma.

3.1.19 Lemma
If $r < \min(\alpha\hat{C}^{-1}, R)$, then

(3.1.19a) $\forall_{x^0 \in B(x^*;r)}$: *the solution $x(t)$ of* (3.1.4) *with $x(0) = x^0$ remains in $B(x^*;r)$ and* $|x(t)-x^*| \leq \exp((-\alpha+\hat{C}r)t)\,|x^0-x^*|$.

♦

Proof

Let $r < \min(\alpha\hat{C}^{-1}, R)$ and let $x(t)$ be a solution of (3.1.4) with $x(0) \in B(x^*;r)$. Then for $t \geq 0$

$$\frac{d}{dt}|x(t)-x^*|^2 = 2\langle x(t)-x^*, M(x(t))f(x(t))\rangle .$$

For notational convenience we drop the argument t of $x(t)$.

$$\begin{aligned}\langle x-x^*, M(x)f(x)\rangle &= \langle x-x^*, M(x)J(x)(x-x^*)\rangle + \langle x-x^*, M(x)(f(x)-J(x)(x-x^*))\rangle \\ &= \langle x-x^*, M(x)J(x)(x-x^*)\rangle \\ &\quad + \langle x-x^*, \int_0^1 M(x)[J(x^*+s(x-x^*))-J(x)](x-x^*)\,ds\rangle .\end{aligned}$$

Hence $\frac{d}{dt}|x(t)-x^*|^2 \leq 2(-\alpha+\hat{C}r)|x(t)-x^*|^2$, i.e. $|x(t)-x^*|$ is a descending function and $x(t)$ remains in the ball $B(x^*;r)$ for all $t \geq 0$.

♦

Next we compare the contraction domain of (3.1.4) according to lemma 3.1.19 to the convergence domain of Newton's method. According to the affine invariant Newton-Kantorovich theorem the latter domain exists of those points x^0 that satisfy

$$|J^{-1}(x^0)f(x^0)|.\gamma \leq \frac{1}{2},$$

with γ an upper bound on

$$\frac{|J^{-1}(x^0)(J(x)-J(y))|}{|x-y|},$$

where x,y are in a convex neighbourhood of x^0 (for a more precise formulation see Appendix D). Comparing this to lemma 3.1.19 for Davidenko's equation (3.1.3), i.e. $M(x) = -J^{-1}(x)$, we see that γ is of the same order of magnitude as $\hat{C}$ and that

$$|J^{-1}(x^0)f(x^0)| \approx |x^0-x^*|$$

can be identified with r. Since $\alpha = 1$ $(= -\mu[-J^{-1}(x)J(x)])$, the lemma shows a contraction domain for Davidenko's equation of approximately the same size as the convergence domain of Newton's method according to the Newton-Kantorovich theorem.

Hence a non-Davidenko choice for the preconditioner can be beneficial if either the value of $\hat{C}$ is reduced or the value of α is increased.

§3.2 The integration method

In this section we introduce a special integration method for the IVP

(3.2.1a) $\dot{x}(t) = M(x)f(x) \quad , \quad t>0 \ ,$

(3.2.1b) $x(0) = x^0$

and investigate its properties. Remember that our aim is to obtain a zero x^* of $f(x)$; not to obtain an accurate estimate of the solution $x(t)$ of (3.2.1).

When using explicit integration methods, numerical stability considerations invariably lead to step size restrictions. For our purposes this can be a disadvantage, since the solution $x(t)$ approximates the contraction point x^* better if the solution is followed over a larger interval.
However, not all implicit methods necessarily have profitable stability properties for larger step sizes. The trapezoidal scheme as used in [Bo] does not yield ultimate fast convergence for large step sizes since $x^{j+1}-x^* \approx x^*-x^j$. So we are interested in a method that allows large step sizes as $x(t)$ approaches the contraction point and gives rapid final convergence. As we do not require a very small discretization error, we look for a 'least work' , i.e. low order method. The simplest method that answers this description, is of course Euler backward, i.e.

(3.2.2) $x^{j+1} = x^j + h_j M(x^{j+1})f(x^{j+1}) \quad , \quad j \geq 0 \ .$

However using an iterative scheme to solve (3.2.2) involves several evaluations of $M(x)$ at each step. This will not be necessary, if we use the following mixture of implicit and explicit Euler, to be referred to as *mixed Euler*

(3.2.3) $x^{j+1} = x^j + h_j M(x^j)f(x^{j+1}) \quad , \quad j \geq 0 \ .$

Solving this equation requires essentially less work than the equation encountered in the implicit Euler method, but, as we prove later on, the mixed Euler method has stability properties similar to those of the implicit Euler method.

Some authors reject the use of implicit integration methods in this case, see e.g. [AbBr]. An often used argument is that implicit integration methods require the solution of a non-linear equation, which was our original problem. However, equation (3.2.3) contains the step size h_j . We show in §3.3 that this non-linear equation can always be solved with Newton's method, if h_j is sufficiently small. And, moreover, once x^j is close to x^*, Newton's method converges for all step sizes $h_j > 0$.

First we show that the mixed Euler method is consistent of order one.

3.2.4 Lemma

The discretization error $\delta(t_j,x,h_j)$ *of the mixed Euler method defined by*

$$\delta(t_j,x,h_j) := h_j^{-1}.\{x(t_{j+1}) - x(t_j) - h_j M(x(t_j))f(x(t_{j+1}))\} \ , \tag{3.2.4a}$$

is bounded, as follows :

$$|\delta(t_j,x,h_j)| \le \tfrac{1}{2}h_j \max_{t\in[t_j,t_{j+1}]} |\ddot{x}(t)| + h_j |M(x(t_j))J(\nu_2)| \max_{t\in[t_j,t_{j+1}]} |\dot{x}(t)| \ , \tag{3.2.4b}$$

with ν_2 *a convex combination of* $x(t_j)$ *and* $x(t_{j+1})$.

Proof

The estimate follows immediately from the relation

$$\begin{aligned}\delta(t_j,x,h_j) &= h_j^{-1}.\{x(t_{j+1}) - x(t_j) - h_j M(x(t_j))f(x(t_{j+1}))\} \\ &= h_j^{-1}.\{x(t_{j+1}) - x(t_j) - h_j M(x(t_j))f(x(t_j))\} + M(x(t_j))\cdot\{f(x(t_j)) - f(x(t_{j+1}))\} \ .\end{aligned}$$

♦

3.2.5 Remark

For the choice $M(x) = -J^{-1}(x)$ with $J^{-1}(x)$ bounded, the bound on the discretization error can be sharpened to

$$|\delta(t_j,x,h_j)| \le \tfrac{1}{2}h_j \max_{t\in[t_j,t_{j+1}]} |\ddot{x}(t)| + h_j \max_{t\in[t_j,t_{j+1}]} |\dot{x}(t)| + O(h_j^2) \ . \tag{3.2.5a}$$

♦

Since the mixed Euler method is a consistent one-step scheme, it is a convergent integration method if $M(x)f(x)$ is locally Lipschitz continuous on an appropriate domain. For completeness the convergence proof is given in Appendix E.

Thus far we looked at the properties of the mixed Euler method as an ODE-solver. Now we investigate its behaviour as a 'root-finder'. Let assumption 3.1.17 hold; this means in particular that

$$\exists_{\alpha>0} \ \forall_{x\in B(x^*;R)} \ : \ \mu[M(x)J(x)] \le -\alpha \ . \tag{3.2.6}$$

We show that, if x^j and x^{j+1} are both in $B(x^*;r)$, with $r < \min(\alpha\hat{C}^{-1}, R)$, and h_j is sufficiently small, then x^{j+1} is in the small sphere shown in figure 3.1. The larger dotted sphere shows the bound that is usually derived in this kind of situation.

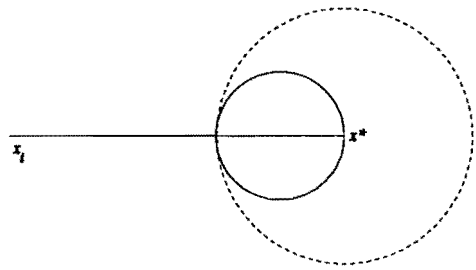


Figure 3.1

3.2.7 Theorem (convergence of the iterative process to x^*)
Let $r < \min(\alpha\hat{C}^{-1}, R)$ and let x^j be in $B(x^*;r)$ and suppose that h_j is sufficiently small to guarantee that x^{j+1} is in $B(x^*;r)$. If $\alpha - \hat{C}r - C_J L_M |x^j - x^{j+1}| > 0$, then the constant b_j defined by

(3.2.7a) $$b_j := 1 + h_j(\alpha - C_J L_M |x^j - x^{j+1}| - \hat{C}|x^j - x^*|) .$$

is larger than 1 and $|x^{j+1} - x^*| \leq \dfrac{|x^j - x^*|}{b_j}$ and, moreover, the vector x^{j+1} is in the sphere with

(3.2.7b) $$\text{centre } \left(1 - \frac{1}{2b_j}\right)x^* + \frac{1}{2b_j}x^j \quad \text{and radius } \frac{|x^j - x^*|}{2b_j} .$$

Proof
Define $e_j := x^j - x^*$. From (3.2.6) we get

$$\begin{aligned} <e_{j+1}, e_{j+1}> &= <e_{j+1}, e_j> + h_j <e_{j+1}, M(x^j) f(x^{j+1})> \\ &= <e_{j+1}, e_j> + h_j <e_{j+1}, M(x^{j+1}) J(x^{j+1}) e_{j+1}> \\ &\quad + h_j <e_{j+1}, (M(x^j) - M(x^{j+1}))(f(x^{j+1}) - f(x^*))> \\ &\quad + h_j <e_{j+1}, M(x^{j+1}) \int_0^1 [J(x^* + s(x^{j+1} - x^*)) - J(x^{j+1})] e_{j+1} ds> . \end{aligned}$$

Hence

(*) $$|e_{j+1}|^2 \leq <e_{j+1}, e_j> + |e_{j+1}|^2 h_j(-\alpha + L_M C_J |x^j - x^{j+1}| + \hat{C}|e_{j+1}|) .$$

This means that if $\alpha - \hat{C}r - C_J L_M |x^j - x^{j+1}| \geq 0$, then $|x^{j+1} - x^*| \leq |x^j - x^*|$.

And (*) shows that $b_j|e_{j+1}|^2 \le \langle e_{j+1}, e_j\rangle$, which in turn implies that

$$\begin{aligned}|x^{j+1} - (1-\frac{1}{2b_j})x^* - \frac{1}{2b_j}x^j|^2 &= |e_{j+1}|^2 - 2\frac{1}{2b_j}\langle e_{j+1}, e_j\rangle + (\frac{1}{2b_j})^2|e_j|^2 \\ &\le \frac{1}{b_j}\langle e_{j+1}, e_j\rangle - \frac{1}{b_j}\langle e_{j+1}, e_j\rangle + (\frac{1}{2b_j})^2|e_j|^2 \\ &= (\frac{1}{2b_j})^2|x^j - x^*|^2 .\end{aligned}$$

◆

Once x^j is in $B(x^*;r)$ this theorem can be applied, since a suitable choice of the variable h_j can guarantee that

(*i*) x^{j+1} is in $B(x^*;r)$,

(*ii*) $\alpha > \hat{C}r + C_J L_M |x^j - x^{j+1}|$.

As soon as $|x^j - x^*| < (\alpha - \hat{C}r)/2C_J L_M$, the constant b_j is larger than 1 independent of h_j , i.e. the restrictions on h_j are lifted. In that case it is favourable to choose h_j large since that yields superlinear convergence, viz.

$$\lim_{j\to\infty} \frac{|x^{j+1} - x^*|}{|x^j - x^*|} \le \lim_{j\to\infty} \frac{1}{1 + h_j(\alpha - C_J L_M |x^j - x^{j+1}|)} = 0, \quad \text{if } \lim_{j\to\infty} h_j = \infty .$$

In the next section we prove that if $x^0 \in B(x^*;r)$, it is indeed possible to choose a step size sequence $\{h_j\}$ such that the corresponding sequence of mixed Euler elements $\{x^j\}$ converges to x^* and reaches $B(x^*; (\alpha - \hat{C}r)/C_J L_M)$ in a finite amount of steps.

Finally we mention that a result similar to theorem 3.2.7 can be derived for the iterates of the implicit Euler's method showing the similarity between the convergence behaviour of mixed and implicit Euler's method.

3.2.8 Remark

Let the ODE (3.2.1a) *satisfy condition* (3.2.6) *and let* $x^j \in B(x^*;R)$. *Let* h_j *be such that* x^{j+1} defined *by*

$$x^{j+1} := x^j + h_j M(x^{j+1}) f(x^{j+1})$$

exists and lies in $B(x^*;R)$. *Then* x^{j+1} *lies in the ball with*

$$\textit{centre} \quad \left(1 - \frac{1}{2(1+\alpha h_j)}\right)x^* + \frac{1}{2(1+\alpha h_j)}x^j \quad \textit{and} \quad \textit{radius} \quad \frac{|x^j - x^*|}{2(1+\alpha h_j)} . \qquad ◆$$

§3.3 Implementation of the mixed Euler method

In the previous sections we addressed convergence of the mixed Euler method. For implementation the following aspects are of interest

(*i*) the choice of the preconditioner $M(x)$

(*ii*) a method to obtain the next iterate x^{j+1}

(*iii*) step size control

The choice of the preconditioner $M(x)$ is strongly problem dependent. In some cases the Davidenko choice $M(x) = -J^{-1}(x)$ is appropriate. In the next chapter we will derive a preconditioner for the non-linear equations arising from the multiple shooting method applied to a class of non-linear boundary value problems.

To obtain the next iterate x^{j+1} in the mixed Euler process from formula (3.2.3), we have to solve the non-linear equation

(3.3.1a) $\quad g(y;x^j,h_j) = 0$,

with

(3.3.1b) $\quad g(y;x^j,h_j) := h_j^{-1}(y-x^j) - M(x^j)f(y)$.

We show that convergence of the Newton method with starting point x^j can be influenced by the choice of the step size h_j. Let the Newton iterates on $g(y;x^j,h_j)$ be denoted by $\{y^i\}$, i.e.

(3.3.2)
$$\begin{cases} y^0 = x^j, \\ y^{i+1} = y^i - \left(\dfrac{dg}{dy}(y^i;x^j,h_j)\right)^{-1} g(y^i;x^j,h_j) \quad , \quad i \geq 0 \ . \end{cases}$$

3.3.3 Lemma

Let $x^j \in B(x^*;R)$. Then under the assumptions 3.1.17 *the following statements hold.*

$$(i)\ \textit{If}\ \frac{h_j}{1+\alpha h_j} \leq \begin{cases} \dfrac{1}{\sqrt{2C_M^2 L_J |f(x^j)|}} & , \textit{ if } \ \frac{1}{2}R^2 L_J > |f(x^j)| \ , \\ \dfrac{2R}{C_M(R^2 L_J + 2|f(x^j)|)} & , \textit{ if } \ \frac{1}{2}R^2 L_J \leq |f(x^j)| \ , \end{cases}$$

then the Newton process (3.3.2) *on $g(y;x^j,h_j)$ converges.*

(ii) If $|f(x^j)| \le \min(\frac{\alpha^2}{2C_M^2 L_J}, \frac{1}{2}R^2 L_J)$, *then the Newton process* (3.3.2) *converges for all step sizes* $h_j > 0$.

Proof

The first derivative of $g(y;x^j,h_j)$ with respect to y reads $g'(y;x^j,h_j) = h_j^{-1} I_m - M(x^j)J(y)$. The logarithmic norm of $M(x)J(x)$ can be used to estimate $|g'(y;x^j,h_j)^{-1}|$:

$$\forall_{\xi \in \mathbb{R}^m} : <\xi, [h_j^{-1} I_m - M(x^j)J(x^j)]\xi> \ge (h_j^{-1} - \mu_2[M(x^j)J(x^j)])|\xi|^2 \ge (h_j^{-1} + \alpha)|\xi|^2$$

$$\Rightarrow |g'(x^j;x^j,h_j)^{-1}|_2 = \frac{1}{\mathrm{glb}_2(g'(x^j;x^j,h_j))} \le \frac{h_j}{1+\alpha h_j} .$$

The conditions of the Newton-Kantorovich theorem now read, that with ν defined as

$$\nu := \left(\frac{h}{1+\alpha h}\right)^2 C_M^2 L_J |f(x^j)| ,$$

ν has to satisfy

(1) $\quad \nu \le \frac{1}{2}$

and

(2) $\quad R > \frac{1-\sqrt{1-2\nu}}{h(1+\alpha h)^{-1} C_M L_J} = \frac{1-\sqrt{1-2\nu}}{\sqrt{\nu}} \sqrt{\frac{|f(x^j)|}{L_J}} .$

Some calculus shows that (2) is satisfied for all $\nu \in [0,\frac{1}{2})$, if $|f(x^j)| < \frac{1}{2}R^2 L_J$. Otherwise (2) imposes the condition

$$\nu \le \frac{4R^2 L_J |f(x^j)|}{(R^2 L_J + 2|f(x^j)|)^2} .$$

This proves (*i*) after the definition of ν has been inserted in the conditions. And (*ii*) can be obtained from (*i*) with some simple calculus.

♦

This lemma disproves an often used argument to reject implicit integration methods for Davidenko's equation, viz. it requires at each step solving a non-linear equation, which is, wrongly, considered to be equal to the original problem. For in this case convergence is guaranteed for appropriate values of h_j.

Now we are able to proof a statement already stated in §3.2, viz. that it is indeed possible the form a sequence of step sizes $\{h_j\}$ such that the corresponding mixed Euler sequence does not stall, but converges to x^*.

3.3.4 Theorem

Let $r < \min(\alpha\hat{C}^{-1}, R)$ *and* $x^0 \in B(x^*,r)$. *There are* $\varepsilon > 0$ *and* $h > 0$ *such that the mixed Euler sequence* $\{x^j\}$ *with step size h exists, lies in* $B(x^*,r)$ *and satisfies*

$$(3.3.4a) \qquad \forall_{j\geq 0} \;:\; \frac{|x^{j+1}-x^*|}{|x^j-x^*|} < \frac{1}{1+\varepsilon} .$$

Proof

Define $h := \min\left(\frac{1}{2}\frac{\alpha-\hat{C}r}{C_J L_M C_M C_f}, \frac{1}{\sqrt{2C_M^2 L_J C_f}}, \frac{2r}{C_M(r^2 L_J + C_f)}\right)$

and $\varepsilon := h(\alpha - \hat{C}r - hC_J L_M C_M C_f)$.

Since $C_f \geq |\, f(x^0)\,|$, the latter two terms of the definition of h imply that the requirements of theorem 3.3.3 are satisfied (with R replaced by r). Hence x^1 exists and lies in $B(x^*,r)$. This is one of the requirements of theorem 3.2.7; the other one is

$$\begin{aligned} \alpha - \hat{C}r - C_J L_M |x^1 - x^0| > 0 \;&\Leftrightarrow\; \alpha - \hat{C}r - C_J L_M h\,|M(x^0)f(x^1)| > 0 \\ &\Leftarrow\; \alpha - \hat{C}r - hC_J L_M C_M C_f > 0 \\ &\Leftrightarrow\; h < \frac{\alpha-\hat{C}r}{C_J L_M C_M C_f} . \end{aligned}$$

Hence we may conclude that $|\, x^1 - x^*\,| < |\, x^0 - x^*\,|$ and, moreover, that

$$\frac{|x^1-x^*|}{|x^0-x^*|} < \frac{1}{1+h(\alpha-\hat{C}r-hC_J L_M C_M C_f)} .$$

The parabola in the denominator takes it maximum value at $\frac{1}{2}\frac{\alpha-\hat{C}r}{C_J L_M C_M C_f}$. An induction argument concludes the proof.

♦

3.3.5 Remark

(*i*) If we do not use the estimate $C_f > |\, f(x^j)\,|$ in the proof of 3.3.4, we are able to form an increasing sequence of step sizes $\{h_j\}$ such that the mixed Euler elements still remain in $B(x^*,r)$, with $|\, x^j - x^*\,|$ decreasing more rapidly.

(*ii*) The sequence $\{x^j\}$ formed in theorem 3.3.4 reaches after a finite amount of steps the ball around x^*, where neither the Newton-Kantorovich theorem nor theorem 3.2.7 imposes any bound on the step size h. Hence the super-linear convergence as predicted in §3.2 is reached eventually.

◆

If we use $-J^{-1}(x)$ as a preconditioner, the first Newton step on $g(y;x^j,h_j)$ reads

$$y^1 = x^j - \frac{h^j}{1+h_j} J^{-1}(x^j) f(x^j), \tag{3.3.6}$$

i.e. a damped Newton step for the original problem $f(x) = 0$.
There are two major differences between damped Newton and our algorithm. First of all we generally perform several Newton steps on (3.3.1). So y^1 is not the next iterate, but only an intermediate result. Secondly, and more importantly, we base our choice of the damping factor on controlling the discretization error and not on iteratively adapting the damping factor until the value of some object function decreases. However, once the iterates x^j approach x^* the first Newton iterate on $g(y;x^j,h_j)$ is accepted as x^{j+1}. At the same time h_j tends to infinity, so the implementation of the mixed Euler method tends asymptotically to the ordinary Newton method. This shows that in this case our method has second order convergence eventually.

Since we want to limit the amount of work per time step, the actual implementation uses a modified Newton method, viz.

$$\begin{cases} y^0 = x^j, \\ y^{i+1} = y^i - g'(x^j;x^j,h_j)^{-1} g(y^i;x^j,h_j) \quad , \quad i \geq 0. \end{cases} \tag{3.3.7}$$

Now only one evaluation of the preconditioner $M(x)$ and the Jacobian $J(x)$ per time step are necessary. As convergence criterion we used the size of $| g'(x^j;x^j,h_j)^{-1} g(y;x^j,h_j) |$.

The Newton process described above, is just an auxiliary tool to follow the path $x(t)$. Hence the step size is determined by a control mechanism on the discretization error. We compute the solution $x(t)$ with given absolute and relative tolerances ATOL and RTOL respectively. Hereto the discretization error is estimated by

$$\text{EST} := \frac{1}{2} h_j^2 \left| \frac{|x^j - x^{j-1}|}{h_{j-1}} - \frac{|x^{j-1} - x^{j-2}|}{h_{j-2}} \right| \cdot \frac{2}{h_{j-1} + h_{j-2}} \, . \tag{3.3.8}$$

In our algorithm we require EST to be approximately equal to ATOL + RTOL $| x^j |$. Small values of RTOL and ATOL increase the robustness of the method, but require many time steps (= work) to reach x^*. In practice values like 10^{-1} or 10^{-2} work very well.

Based on the above considerations the step size h_j is determined at every step as follows :

3.3.9 Algorithm (step size determination)

- take h_j equal to h_{j-1} .
- double the step size if it has not been changed in 3 consecutive steps, to prevent conservatism.
- if the Newton process has not converged in 3 iterations, halve the step size until convergence is reached. (If h_j has not been changed we do not expect non-convergence, unless the path has entered a troublesome area, i.e. $J(x)$ is nearly singular or $J(x)$ has a large Lipschitz constant.)
- at every step compute the quantity TEST := EST/(ATOL + $| x^j |$·RTOL) .

  If TEST $\in$ [0.25 , 4] the step is accepted.

  If TEST > 4 the discretization error is too large and x^j is recalculated for the step size

$$h_{new} = \frac{h_{old}}{\sqrt{\text{TEST}}} \quad . \tag{3.3.9a}$$

  If TEST < 0.25 the path is followed 'too accurately'. Now accept x^j and increase h_j according to (3.3.9a).

♦

Remark

We have used the algorithm outlined above to test the ideas presented in this chapter. But we are well-aware of the fact that various refinements and modifications can improve its efficiency. However, the results obtained with this program, as presented in §3.4, already indicate a relatively good performance.

§3.4 Numerical results

In [AbBr] several explicit integration methods for Davidenko's equation are studied and tested on some problems. We have applied the mixed Euler implementation described above to those test problems, in order to illustrate the performance of the method and to compare it with the explicit integration methods presented in [AbBr] and with damped Newton. First we describe the (8) test problems briefly.

1+2 A function, to be found in [Bo],

$$f\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} := \begin{pmatrix} x_1^2 - x_2 + 1 \\ x_1 - \cos(\frac{\pi}{2} x_2) \end{pmatrix}. \tag{3.4.1}$$

The sought solution is $x^* = (0,1)$ and the initial value is

- (1,0) for the first test problem (**1**)
- (−1,−1) for the second test problem (**2**).

There are several curves where the Jacobian of f is singular, viz.

$$\pi x_1 \sin(\frac{\pi}{2} x_2) = -1. \tag{3.4.2}$$

In Figure 3.2 we have plotted two of those *singularity curves* and the direction field of Davidenko's equation (the lenght of each vector has been divided by 6 to keep overview). One can see that coming from (−1,−1) the 'path' approaches a singularity curve rather closely and this is where the Newton update $-J^{-1}(x)f(x)$ becomes large.

3

$$f\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} \frac{1}{2}\sin(x_1 x_2) - \frac{x_2}{4\pi} - \frac{x_1}{2} \\ (1 - \frac{1}{4\pi})(e^{2x_1} - e) + \frac{e x_2}{\pi} - 2e x_1 \end{pmatrix} \tag{3.4.3}$$

with initial guess (0.6 , 3). the correct solution is (0.5 , π).

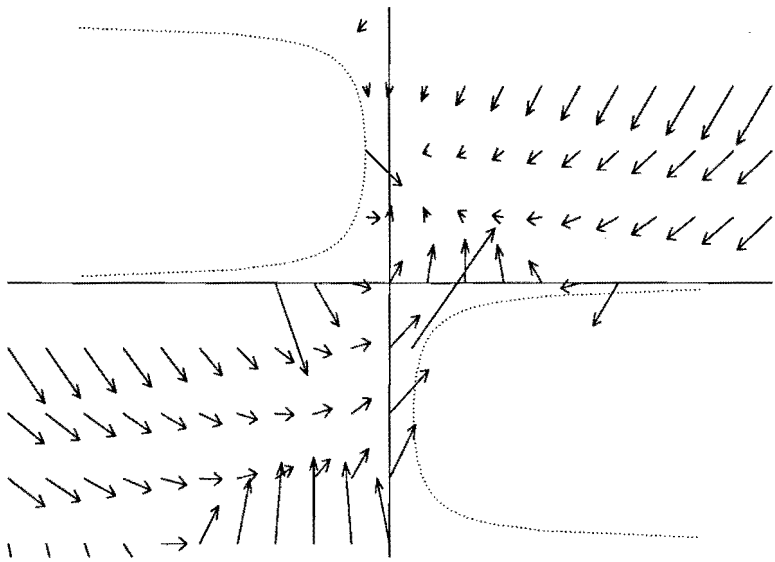


Figure 3.2

4 The gradient of Rosenbrock's function :

$$f\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 400x_1(x_1^2 - x_2) + 2(x_1 - 1) \\ -200(x_1^2 - x_2) \end{pmatrix}, \tag{3.4.4}$$

with initial guess (-1.2 , 1.0). The solution is (1,1). The Jacobian is singular on the parabola $x_1^2 = x_2 - 0.005$, but any zero-finding procedure has a strong tendency to follow the neighbouring parabola $x_1^2 = x_2$. In fact the latter parabola can be seen as a narrow gorge with very steep walls, (a 3-dimensional plot of the situation is given in [Br]). Indeed, the path from (-1.2 , 1) towards the solution immediately heads for the curve $x_1^2 = x_2$ and then follows it up to (1,1).

5 A function found in [Br] :

$$f\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 2\sin(2\pi x_1/5)\sin(2\pi x_3/5) - x_2 \\ 2.5 - x_3 + 0.1x_2\sin(2\pi x_3) - x_1 \\ 1 + 0.1\sin(2\pi x_1) - x_3 \end{pmatrix}, \tag{3.4.5}$$

with initial guess (0,0,0). The correct solution is (1.5 , 1.809.. , 1.0).

6

$$f_i(x) = \sum_{\substack{j=1 \\ j\neq i}}^{6} \cot(\beta_i x_j) \quad , \quad i \in \{1,2,..,6\} \ , \tag{3.4.6}$$

where the coefficients β_i are 2.249×10^{-2} , 2.166×10^{-2} , 2.038×10^{-2} , 2.0×10^{-2} , 1.918×10^{-2} , 1.835×10^{-2} , for $i = 1,2,..,6$, respectively. With initial guess $x_i = 75$ and correct solution approximately (121.9 , 114.2 , 93.6 , 62.3 , 41.3 , 30.5).

7+8 A discretization of the BVP

$$\begin{aligned} &3\ddot{y}y + \dot{y}^2 = 0 \quad , \quad 0<t<1 \ , \\ &y(0) = 0 \ , \\ &y(1) = 20 \ , \end{aligned} \tag{3.4.7}$$

gives rise to the equations

$$\begin{aligned} f_1 &= 3x_1(x_2 - 2x_1) + x_2^2/4 \ , \\ f_i &= 3x_i(x_{i+1} - 2x_i + x_{i-1}) + (x_{i+1} - x_{i-1})^2/4 \ , \quad , \ i = 2,..,n-1 \ , \\ f_n &= 3x_n(20 - 2x_n + x_{n-1}) + (20 - x_{n-1})^2/4 \ . \end{aligned} \tag{3.4.8}$$

The solution of the boundary value problem is $y = 20t^{3/4}$. The initial guess is $x_i = 10$, $i = 1, \ldots, n$. For problem **7** $n = 10$ and problem **8** $n = 20$.

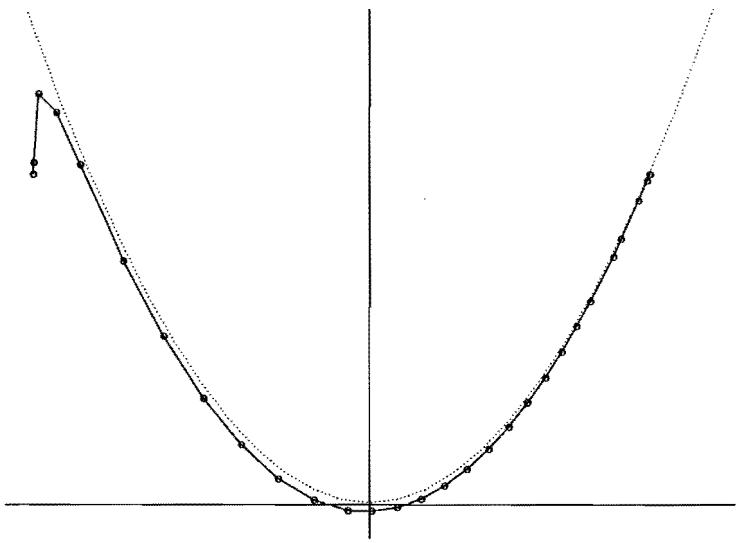
We test the mixed Euler algorithm described in §3.3 on the problems considered in [AbBr] with $M(x) = -J^{-1}(x)$. As a measure for the amount of work we use the number of function calls (#*f*) plus m (= dimension of the system) times the number of Jacobian evaluations (#*J*). Only at problem **7** and **8** we multiply #*J* by 3, since the Jacobian is tridiagonal. This is also done in [AbBr]. We set the tolerances ATOL = RTOL = 10^{-1}, i.e. we require the approximation of the path $x(t)$ to have approximately one correct number. Of course this large discretization error may jeopardize the convergence of the process if the iterates stray off the correct path. On the other hand larger tolerances for the discretization error allow larger step sizes and hence require less function evaluations. For the tolerances RTOL = ATOL = 10^{-1}, the mixed Euler method converges for all eight test problems. The amount of function evaluations to obtain an approximation x^j with error $\| J^{-1}(x^j)f(x^j) \| < 10^{-6}$, is listed in Table 3.1. Note that #*J* is equal to the amount of steps.

Results of the mixed Euler method for the test problems with RTOL=ATOL=10^{-1} and required accuracy 10^{-6}.

problem no.	1	2	3	4	5	6	7	8
#f	13	28	9	111	23	18	18	24
#J	11	16	6	31	13	10	13	14
#f+m#J	35	60	21	173	62	78	57	66

Table 3.1

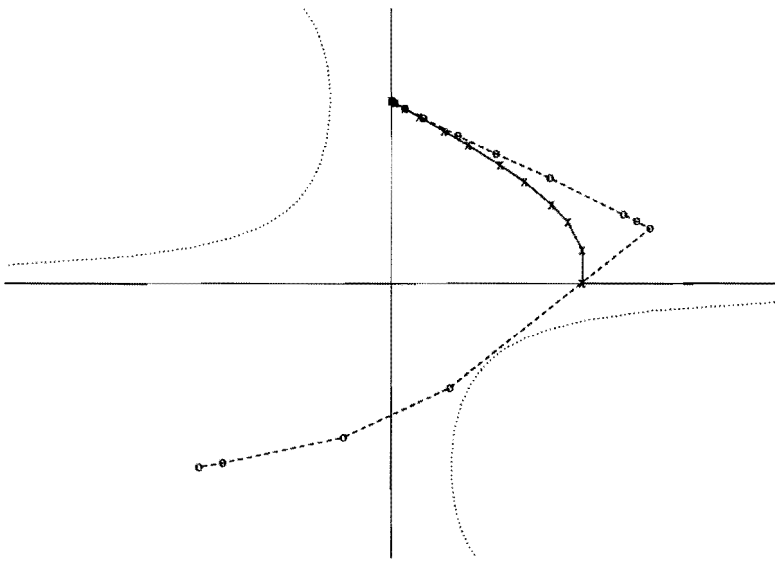
We describe the performance of the mixed Euler method on the problems 1,2 and 4 in more detail. In Figure 3.3 we plot the path of the mixed Euler method for test problem 4, together with the curve on which $J(x)$ is singular. The path runs very close to the singularity curve. Since the Lipschitz constant of the Jacobian is large, the step size h_j is determined by the Newton process on $g(x;x^j,h_j)$; the estimated discretization error is at every step smaller than the bound $10^{-1}(1 + |x^j|)$. Indeed, if we apply the mixed Euler method to problem 4 with ATOL = RTOL = 10^{-2}, the process requires 32 steps, i.e. just 1 more than for the previous case.



o- -o = mixed Euler iterates
..... = singularity curve of $J(x)$

Figure 3.3

Problem 1 and 2 are concerned with solving the same function (3.4.1) from two different starting points. The path from the first starting point (1,0) to the solution $x^* = (0,1)$ leads through a region without singularities of $J(x)$ and with moderate sizes of the direction vectors $-J^{-1}(x)f(x)$. The mixed Euler approximation of the path with ATOL = RTOL = 10^{-1} is plotted in Figure 3.4 by a solid line; the iterates are marked with an 'x'. The path from the starting point (−1,−1) leads through the fourth quadrant and runs close to a singularity curve (cf. the direction field Fig. 3.2). The mixed Euler approximation of the path is plotted in Figure 3.4. by a dashed line and the iterates are marked with an 'o'.



..... = singularity curve of $J(x)$
x—x = mixed Euler iterates of problem 1
o- -o = mixed Euler iterates of problem 2

Figure 3.4

In Table 3.2 we list information about the mixed Euler iteration on the second test problem. The step size h_j is reduced in the third and fourth step, because the 'internal' Newton iteration on $g(x;x^j,h_j)$ does not converge in 3 steps. This is due to the fact that x^2 and x^3 are close to a singularity curve. Based on the size of the discretization error the step size never reduces, but it increases in steps 2, 7, 12, 14, 15 and 16.

j	k	$c(J)$	h	$x[1]$	$x[2]$	EST	$\lvert J^{-1}(x)f(x)\rvert$	$\lvert x^j-x^*\rvert$	b^{-1}	τ
0	0	1.7		-1.0000000E+00	-1.0000000E+00		1.4E+00	2.236E+00		
1	0	1.7	1.0E-01	-8.7461150E-01	-9.7804973E-01	3.0E-03	1.3E+00	2.163E+00	9.672E-01	0.3389
2	0	1.9	8.1E-01	-2.4989789E-01	-8.3995892E-01	1.2E-01	9.3E-01	1.857E+00	8.585E-01	0.2024
3	1	1.5	4.1E-01	3.0763125E-01	-5.6831302E-01	8.2E-02	1.6E+00	1.598E+00	8.607E-01	0.3976
4	2	1.3	1.0E-01	1.3540093E+00	3.0039046E-01	1.8E-01	1.1E+01	1.524E+00	9.536E-01	0.4780
5	0	1.5	1.0E-01	1.2824901E+00	3.4063594E-01	5.6E-01	7.0E-01	1.442E+00	9.462E-01	0.5589
6	0	1.5	1.0E-01	1.2139238E+00	3.7755508E-01	1.7E-03	6.8E-01	1.364E+00	9.460E-01	0.5609
7	0	1.5	1.2E+00	8.3025972E-01	5.7794945E-01	3.7E-01	3.8E-01	9.314E-01	6.827E-01	0.3959
8	0	1.8	1.2E+00	5.4915147E-01	7.1131096E-01	5.1E-02	2.8E-01	6.204E-01	6.661E-01	0.4270
9	0	2.1	1.2E+00	3.4919992E-01	8.1044260E-01	4.1E-02	1.9E-01	3.973E-01	6.404E-01	0.4783
10	0	2.4	2.5E+00	1.6708166E-01	9.0522819E-01	1.6E-01	9.9E-02	1.921E-01	4.834E-01	0.4309
11	0	3.0	2.5E+00	6.8875102E-02	9.5984007E-01	4.2E-02	5.0E-02	7.973E-02	4.151E-01	0.5684
12	0	3.5	5.4E+00	1.6043028E-02	9.9040489E-01	1.1E-01	1.4E-02	1.869E-02	2.345E-01	0.6092
13	0	3.9	5.4E+00	2.8480671E-03	9.9828467E-01	2.0E-02	2.7E-03	3.325E-03	1.779E-01	0.8625
14	0	4.0	1.7E+01	1.7000806E-04	9.9989720E-01	3.0E-02	1.7E-04	1.987E-04	5.976E-02	0.9246
15	0	4.0	4.4E+01	3.8169407E-06	9.9999769E-01	4.9E-03	3.8E-06	4.461E-06	2.246E-02	0.9879
16	0	4.0	2.8E+02	1.3509525E-08	9.9999999E-01	9.2E-04	1.4E-08	1.579E-08	3.539E-03	0.9983

number of function calls : 28
number of Jacobian evaluations : 16
Stop criterion : $\lvert J^{-1}(x^j)f(x^j)\rvert < 10^{-6}$,
Discretization error control : ATOL = RTOL = 10^{-1}.

j : number of time steps,
k : number of changes in the step size h_j during step j,
c(J) : condition number of $J(x^j)$,
EST : approximation of the discretization error according to (3.3.8),

$$b^{-1} := \frac{\lvert x^j - x^*\rvert}{\lvert x^{j-1} - x^*\rvert}, \text{ and } \tau := \frac{(b-1)}{h}.$$

Table 3.2

The last two columns show the behaviour of

$$(3.4.9)\quad b^{-1} := \frac{|x^j - x^*|}{|x^{j-1} - x^*|},$$

cf. §3.2. As predicted by theorem 3.2.7, b^{-1} tends to zero if x^j approaches x^*. Moreover, the factor τ satisfying

$$b = 1 + h\tau$$

(cf. (3.2.7b)) converges to $1 = -\mu[-J^{-1}(x)J(x)]$.

We also compare the results of the mixed Euler method with the results from other explicit integration methods as presented in [AbBr]. Table 3.3 gives the amount of work measured by *#f+m.#J* . This shows that the mixed Euler method performs better on all eight test problems, than the two explicit integrators used here and the trapezoidal rule.

***#f+m#J* for the test problems**

	1	2	3	4	5	6	7	8
ME	35	60	21	173	62	78	57	66
RK3	64	89	55	334	113	169	280	280
AB3	71	95	43	299	109	127	221	229
PECE	133	157	115	337	185	309	347	355

ME = Mixed Euler
RK3 = third order Runge Kutta
AB3 = Adams-Bashforth variable step method order 3
PECE = Trapezoidal rule as described in [Bo]

Table 3.3

Time stepping methods are introduced, because in some cases the convergence domain of Newton's method is too small for practical use. Hence for comparison we applied a version of damped Newton :

$$(3.4.8c)\qquad x^{j+1} = x^j - \lambda_j J^{-1}(x^j) f(x^j) \quad , \quad j \geq 0 \ , \quad \lambda_j \in (0,1],$$

to the test functions. The damping factor λ_j is first chosen to be $\lambda_j = \min(2\lambda_{j-1}, 1)$, but if some object function does not decrease, λ_j is halved until it does or $\lambda_j < 10^{-3}$. In the latter case the process is terminated, which is denoted by FAIL in Table 3.4 (N.B. the mixed Euler process on the test problems converged with a step size $h_j \geq 10^{-1}$).

Three different object functions were used :

(1) $| f(x_{new}) |$,

(2) $| J^{-1}(x_{old})f(x_{new}) |$,

(3) $| J^{-1}(x_{new})f(x_{new}) |$,

where x_{old} is the last accepted Newton iterate and x_{new} is the update obtained using λ_j . If the object function decreases the update is accepted.

Table 3.4 shows the number of iterations necessary to reach convergence : $| J^{-1}(x^j)f(x^j) | \leq 10^{-6}$, for the three different object functions. Damped Newton's method with either object function could not solve the second problem, whereas our implementation of the mixed Euler method converged in 20 steps. For the most reliable (and expensive) choice $| J^{-1}(x_{new})f(x_{new}) |$ damped Newton failed to converge in three cases.

Number of iterations with damped Newton

	$\lvert f(x_{new}) \rvert$	$\lvert J^{-1}(x_{old})f(x_{new}) \rvert$	$\lvert J^{-1}(x_{new})f(x_{new}) \rvert$
1	8	6	6
2	FAIL	FAIL	FAIL
3	4	4	4
4	≥ 50	23	FAIL
5	2	6	FAIL
6	6	6	6
7	7	7	7
8	8	8	7

Table 3.4

Conclusion

We have seen that the mixed Euler method has stability properties similar to those of the implicit Euler method. The price for this is a restriction on the step size if we are far from the steady state. On the other hand every time step requires only 1 computation of the preconditioner $M(x)$, so with respect to computational effort the method is competitive with explicit integration methods. On approaching the steady state the step sizes can increase without jeopardizing stability or existence of the next iterate, yielding a superlinear convergence rate. If the preconditioner $M(x)$ equals $-J^{-1}(x)$ our algorithm tends asymptotically to Newton's method.

♦

4 Preconditioned time stepping in combination with multiple shooting

In chapter 2 we have investigated the multiple shooting method for non-linear BVP's. We have found that the resulting set of non-linear equations, denoted by $f(s) = 0$, may be very sensitive to changes in the vector s in the presence of exponentially growing solution modes of the BVP. This sensitivity may have a negative effect on the performance of Newton's method to solve $f(s) = 0$. Hence now we try to construct a more robust solution method, that better controls the influence of the growing modes.

In chapter 3 we considered preconditioned time stepping for solving non-linear equations, not for $f(s)$ specifically, but in a more general setting. In this chapter we focus on applying the time stepping method to $f(s)$ and deriving a suitable preconditioner, if the underlying BVP has separated BC and its linearization at the solution is exponentially dichotomic. The idea for the preconditioner is partly inspired by the fact that we would like to have an appropriate information flow if a BVP is embedded into a hyperbolic (time dependent) system and it is partly based on a sensitivity analysis of MJ for changes in s, with J the Jacobian of $f(s)$.

In section 4.1 we derive a preconditioner M for the non-linear equations arising in multiple shooting and prove that the logarithmic norm of $M(s)J(s)$ is negative, if s is sufficiently close to the solution. In the second section we investigate the sensitivity of MJ for changes in the shooting vector s. In the last section some numerical results of time stepping with this preconditioner are presented and are being compared with the performance of the (damped) Newton method.

§4.1 Construction of the preconditioner

In this chapter we investigate the use of preconditioned time stepping (cf. chapter 3) to solve the non-linear equation occurring in multiple shooting. The outline of the section is as follows. We start with the multiple shooting formulation for separated BC and give a decomposition of the Jacobian, leading to a sort of 'basic' form for the Jacobian. Subsequently we derive a preconditioner for this 'basic' Jacobian based on considerations about properly embedding continuous BVP's into hyperbolic time dependent systems and considerations about the influence of small changes in the Jacobian. Finally we adapt this preconditioner to make it suitable for use in combination with the original Jacobian.

Consider the BVP with separated boundary conditions :

$$(4.1.1)\qquad \begin{cases} \dfrac{dy}{dx} = h(x,y(x)) \quad , \qquad a<x<b \quad \text{and} \qquad y\in C^1([a,b]\to\mathbb{R}^n) \ , \\ g_1(y(b)) = 0 \quad \text{and} \quad g_2(y(a)) = 0 \quad , \\ \qquad\qquad \text{with} \quad g_1 : \mathbb{R}^n \to \mathbb{R}^{n-p} \quad \text{and} \quad g_2 : \mathbb{R}^n \to \mathbb{R}^p \quad , \ 1\le p\le n \ , \end{cases}$$

that satisfies the following assumption.

4.1.2 Assumption
The BVP (4.1.1) *is well-conditioned at its solution* $y^*(x)$, *the linearization at* $y^*(x)$ *is exponentially dichotomic and the space of growing solution modes has dimension* $n-p$.
♦

Recall from §2.3 that application of the multiple shooting method to the BVP, starts out by choosing a grid

$$(4.1.3)\qquad a = x_1 < x_2 < \ldots < x_{N+1} = b$$

and defining non-linear IVP's locally :

$$(4.1.4)\qquad \begin{cases} \dot{y} = h(x,y(x)) & , \ x_k < x < x_{k+1} \ , \ 1\le k\le N \ , \\ y(x_k) = s_k & , \ s_k\in\mathbb{R}^n \ . \end{cases}$$

The solutions of the local problems, if existing, are denoted by $y_k(x;s_k)$; $y(x;s)$ is the function, defined on $[a,b]$, that is equal to $y_k(x;s_k)$ on $(x_k, x_{k+1}]$ and satisfies $y(a;s) = y_1(a;s_1)$. The vectors s_k are determined by

$$(4.1.5)\qquad f(s) = 0 \quad \text{with} \quad s^\top := (s_1^\top, s_2^\top, \ldots, s_{N+1}^\top) \quad , \quad f\in C^1(\mathbb{R}^{n(N+1)}\to\mathbb{R}^{n(N+1)})$$

and $f(s)$ defined by

$$(4.1.6) \qquad f(s) := \begin{pmatrix} y_1(x_2;s_1) - s_2 \\ y_2(x_3;s_2) - s_3 \\ \vdots \\ y_N(x_{N+1};s_N) - s_{N+1} \\ g_1(s_{N+1}) \\ g_2(s_1) \end{pmatrix} .$$

The solution of (4.1.5) that corresponds to $y^*(x)$ is denoted by s^*. The Jacobian $J(s)$ of $f(s)$ can be related to the linearization of (4.1.1) at $y(x;s)$. Therefore we repeat the notation introduced in §2.3 about this.

4.1.7 Notation

The derivative of $h(x,y)$ at $y_k(x;s_k)$ is

$$(4.1.7a) \qquad L_k(x;s_k) := \frac{\partial}{\partial v} h(x,v)\Big|_{v=y_k(x;s_k)} \quad , \; x_k < x < x_{k+1} ,$$

and the derivatives of the boundary condition functions are

$$(4.1.7b) \qquad B_a(s) := \begin{pmatrix} 0 \\ \hline \dfrac{dg_2(u)}{du}\Big|_{u=s_1} \end{pmatrix} \quad \text{and} \quad B_b(s) := \begin{pmatrix} \dfrac{dg_1(v)}{dv}\Big|_{v=s_{N+1}} \\ \hline 0 \end{pmatrix} ,$$

$B_a(s)$, $B_b(s) \in \mathbb{R}^{n\times n}$. *For $k\in\{1, .. , N\}$, the matrix function $Y_k(x;s_k)$ is the fundamental solution of*

$$(4.1.7c) \qquad \dot{z} = L_k(x;s_k)z \quad , \; x_k < x < x_{k+1} ,$$

that satisfies $Y_k(x_k;s_k) = I_n$.

♦

The Jacobian $J(s)$ of the non-linear multiple shooting equation (4.1.6) is given by

$$(4.1.8) \qquad J(s) = \begin{pmatrix} Y_1(x_2;s_1) & -I_n & & & \\ & Y_2(x_3;s_2) & -I_n & & \\ & & \ddots & \ddots & \\ & & & Y_N(x_{N+1};s_N) & -I_n \\ B_a(s) & & & & B_b(s) \end{pmatrix} ,$$

The large Lipschitz constant of $J(s)$ (as established in §2.3) is due to the fact that the exponentially growing solution modes of the BVP are not properly controlled by the local initial conditions. A first step towards constructing a preconditioner that reduces this effect, is to separate the growing and decaying modes. To this end the concept of consistent fundamental solutions (see §1.2) can be used. Since the BVP is well-conditioned and the BC are separated, it follows from lemma 1.2.15 that a fundamental solution can be constructed, whose first $(n-p)$ columns span a space of growing solution modes. In section 2.1 we have sketched an orthogonalization process for the linearized multiple shooting equation that retains this property. Hence we choose an orthonormal matrix Q_1, that satisfies

$$B_a Q_1 = \begin{pmatrix} 0 & 0 \\ 0 & B_a^{(2)} \end{pmatrix}, \tag{4.1.9a}$$

for some full rank matrix $B_a^{(2)} \in \mathbb{R}^{p\times p}$ (Q_1 may for instance result from a QU-decomposition of $B_a{}^\top$, rendering a lower triangular matrix $B_a^{(2)}$ or Q_1 may be such that $B_a^{(2)} = I_p$). Subsequently we determine orthogonal matrices Q_k as the QU-decomposition of

$$Y_k(x_{k+1};s_k)Q_k = Q_{k+1}U_k \quad , k = 1,..,N+1 . \tag{4.1.9b}$$

Writing the Jacobian in terms of the fundamental matrices U_k , can be realized by differentiating $f(s)$ with respect to a transformed variable. Define

$$\mathcal{Q} := \mathrm{diag}\,(Q_1, Q_2, \ldots, Q_{N+1}) , \tag{4.1.10a}$$

$$\hat{\mathcal{Q}} := \mathrm{diag}\,(Q_2, Q_3, \ldots, Q_{N+1}, I_n) , \tag{4.1.10b}$$

then

(4.1.11)

$$\frac{df(s)}{dQ^\top s} = \begin{pmatrix} Q_2U_1 & -Q_2 & & & \\ & Q_3U_2 & -Q_3 & & \\ & & \ddots & \ddots & \\ & & & Q_{N+1}U_N & -Q_{N+1} \\ B_aQ_1 & & & & B_bQ_{N+1} \end{pmatrix} = \hat{\mathcal{Q}} \cdot \begin{pmatrix} U_1 & -I_n & & & \\ & U_2 & -I_n & & \\ & & \ddots & \ddots & \\ & & & U_N & -I_n \\ B_aQ_1 & & & & B_bQ_{N+1} \end{pmatrix}.$$

So in fact at every subinterval-endpoint x_{k+1} the fundamental solution is decomposed into an orthogonal matrix Q_{k+1} that contains information on the evolution of the *directions* of the various modes and an upper triangular matrix U_k , that contains information on the *growth behaviour* of those modes. This growth behaviour can be described in terms of the dichotomy of the problem and so we can relate the magnitude of the elements of U_k to the

dichotomy constants. Let the upper triangular matrix U_k be partitioned into four blocks as in

$$U_k = \begin{pmatrix} B_k & C_k \\ 0 & E_k \end{pmatrix} \quad \text{with} \quad B_k \in \mathbb{R}^{(n-p)\times(n-p)} \quad \text{and} \quad E_k \in \mathbb{R}^{p\times p}. \tag{4.1.12}$$

Recall from lemma 2.1.15 that

$$\forall_k \; : \; |E_k| \le K e^{-\lambda(x_{k+1}-x_k)} \quad \text{and} \quad |B_k^{-1}| \le K e^{-\mu(x_{k+1}-x_k)}. \tag{4.1.13}$$

Note that the constant $\bar{K}$ of lemma 2.1.15 equals K, because the consistency constant L is zero for the particular choice of fundamental solution, we use.

This shows that, as expected, both $|E_k|$ and $|B_k^{-1}|$ become small as we integrate over larger intervals. The part C_k would be zero if the increasing and decreasing modes were orthogonal to each other. This is highly desirable, from a mathematical point of view, as it would give a complete decoupling between the two modes. However, by a non-orthogonal local coordinate system transformation such a decoupling can be obtained. To this end we employ discrete Riccati-transformations, cf. §2.1, recurring backward from $x = b$.

From equation (4.1.11) one can see that, due to the zero structure of $B_a Q_1$ and $B_b Q_{N+1}$, non-singularity of the Jacobian $J(s)$ implies non-singularity of the left upper block $B_b^{(1)}$ of $B_b Q_{N+1}$:

$$B_b Q_{N+1} = \begin{pmatrix} B_b^{(1)} & B_b^{(2)} \\ 0 & 0 \end{pmatrix}, \; B_b{}^{(1)} \in \mathbb{R}^{(n-p)\times(n-p)}.$$

In other words the endpoint conditions control the space spanned by the first columns of Q_{N+1}, i.e. approximately the space of growing modes. The endpoint conditions can be 'concentrated' in the upper $(n-p)\times(n-p)$ block by a Riccati-like transformation, viz. if R_{N+1} is defined by

$$R_{N+1} := \left(B_b^{(1)}\right)^{-1} B_b^{(2)}, \tag{4.1.14a}$$

then

$$B_b Q_{N+1} = \begin{pmatrix} B_b^{(1)} & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} I_{n-p} & R_{N+1} \\ 0 & I_p \end{pmatrix}. \tag{4.1.15}$$

This type of transformation can be used to decouple the growing and decaying modes fully. If the Riccati-matrices $R_k \in \mathbb{R}^{(n-p)\times p}$ are determined by the backward recursion :

(4.1.14b) $R_k := B_k^{-1}(C_k + R_{k+1}E_k)$ for $k=N$ downto 1 ,

then

(4.1.16) $$\begin{pmatrix} I & R_{k+1} \\ 0 & I \end{pmatrix} \cdot \begin{pmatrix} B_k & C_k \\ 0 & E_k \end{pmatrix} \cdot \begin{pmatrix} I & -R_k \\ 0 & I \end{pmatrix} = \begin{pmatrix} B_k & -B_k R_k + C_k + R_{k+1}E_k \\ 0 & E_k \end{pmatrix} = \begin{pmatrix} B_k & 0 \\ 0 & E_k \end{pmatrix}.$$

Again this transformation can be interpreted as a change of the variable to which $f(s)$ is differentiated. Define the matrices

(4.1.17) $$S_k = \begin{pmatrix} I & R_k \\ 0 & I \end{pmatrix} , \quad S_k \in \mathbb{R}^{n \times n} ,$$

and

(4.1.18) $$S := \operatorname{diag}(S_1, S_2, \ldots, S_{N+1}) \quad \text{and} \quad \hat{S} := \operatorname{diag}(S_2, S_3, \ldots, S_{N+1}, I_n) .$$

Then

(4.1.19) $$\frac{df(s)}{dSQ^{\top}s} = \hat{Q} \cdot \hat{S}^{-1} \cdot \begin{pmatrix} B_1 & 0 & -I_{n-p} & & & & & & & \\ & E_1 & 0 & -I_p & & & & & & \\ & & B_2 & 0 & -I_{n-p} & & & & & \\ & & & E_2 & 0 & -I_p & & & & \\ & & & & \ddots & & \ddots & & & \\ & & & & & \ddots & & \ddots & & \\ & & & & & & B_N & 0 & -I_{n-p} & \\ & & & & & & & E_N & 0 & -I_p \\ 0 & 0 & & & & & & & B_b^{(1)} & 0 \\ 0 & B_a^{(2)} & & & & & & & 0 & 0 \end{pmatrix} .$$

These transformations have created a complete decoupling of the growing and decaying modes. Moreover, the right most matrix of (4.1.19) can be interpreted as a discretization of two ODE's, one of dimension p with initial conditions and another of dimension $n-p$ with end point conditions. The decoupling is more apparent after a permutation of the variables. Define

$$\text{(4.1.20)} \qquad \hat{P} := \left(\begin{array}{ccc:ccc} I_{n-p} & 0 & 0 & 0 & 0 & 0 \\ & 0 & 0 & I_p & 0 & 0 \\ & I_{n-p} & 0 & & 0 & 0 \\ & & 0 & & I_p & 0 \\ & & \vdots & & & \vdots \\ & & I_{n-p} & & & 0 \\ & & 0 & & & I_p \end{array}\right).$$

Then

$$\text{(4.1.21)} \qquad \frac{df(s)}{d\hat{P}^{\top}SQ^{\top}s} = \hat{Q}\hat{S}^{-1}\hat{P} \left(\begin{array}{ccc:ccc} B_1 & -I_{n-p} & & & & \\ & \ddots & \ddots & & & \\ & & B_N & -I_{n-p} & & \\ & & & B_b^{(1)} & & \\ \hdashline & & & & E_1 & -I_p \\ & & & & & \ddots & \ddots \\ & & & & & & E_N & -I_p \\ & & & & B_a^{(2)} & & & 0 \end{array}\right).$$

The right most matrix in this expression will be denoted by $\tilde{J}$, i.e.

$$\text{(4.1.22)} \qquad \tilde{J} := \hat{P}^{\top}\hat{S}\hat{Q}^{\top}JQS^{-1}\hat{P}\ .$$

This matrix is called the *basic form of the Jacobian.* At this point it is important to note that the matrices Q and S used in the decoupling depend on the vector s (although we did not make this explicit in the notation). Indeed, if Q and S were constant, this would mean that with respect to a transformed basis, the original BVP exists of two fully independent ODE's, one with initial conditions and another with end point conditions, i.e. the essential character of a BVP is not present.

We set out to find a preconditioner $M(s)$ such that the IVP

(4.1.23a) $$\frac{ds}{dt} = M(s)f(s) \quad , t>0 \ ,$$

(4.1.23b) $$s(0) = s_0 \ .$$

is asymptotically stable at $s = s^*$. In chapter 3 we saw that a sufficient condition for this is

(4.1.24a) $$\exists_{\alpha>0} \ \exists_{R>0} \ \forall_{s\in B(s^*;R)} \ : \ \mu_2[M(s)J(s)] \le -\alpha \ .$$

Moreover, if $B(s^*;r_0)$ denotes the largest ball such that

$$\forall_{s_0\in B(s^*;r_0)} \ : \ s(0) = s_0 \ \Rightarrow \lim_{t\to\infty} s(t) = s^* \ ,$$

then theorem 3.1.18 gives a larger lower bound for r_0, if

(4.1.24b) $$\max_{s,\sigma\in B(s^*;R)} \frac{|M(s)(J(s)-J(\sigma))|}{|s-\sigma|}$$

is smaller. Based on these considerations we will construct an appropriate preconditioner $\tilde{M}$ for $\tilde{J}$. Thereafter $\tilde{M}$ will be adapted to suit the original Jacobian $J(s)$.

We first concentrate on the requirement (4.1.24a). Based on the form of $\tilde{J}$, we consider how a totally decoupled BVP can be embedded into a time dependent partial differential equation. To this end we employ a simple model problem with $n = 2$ and $p = 1$, i.e. a 2-dimensional BVP with 1 growing and 1 decaying mode :

(4.1.25a) $$\begin{pmatrix} \dot{u}(x) \\ \dot{v}(x) \end{pmatrix} = \begin{pmatrix} \mu & 0 \\ 0 & -\lambda \end{pmatrix} \begin{pmatrix} u(x) \\ v(x) \end{pmatrix} \quad , a<x<b \quad , \lambda,\mu>0 \ .$$

A properly scaled analogue of the linearized BC is

(4.1.25b) $$v(a) = \alpha \ , \ u(b) = \beta \ .$$

These boundary conditions fit the dichotomy of the problem well. If we embed the ODE for u in a time dependent PDE, we obtain a hyperbolic system :

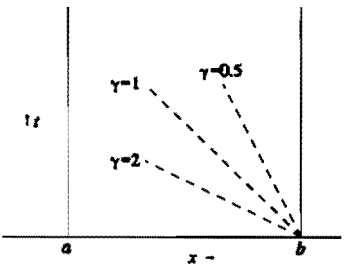
(4.1.26a) $$\frac{\partial u}{\partial t} = \gamma\left(\frac{\partial u}{\partial x} - \mu u\right) \quad , a<x<b \ , \ t>0 \ .$$

The sign of γ has to be chosen in such a way that the streamlines (or characteristics) of (4.1.26a) spread the information of the end point condition $u(b) = \beta$ over the interval, i.e. $\gamma > 0$.

Since the ODE for $v(x)$ has an initial condition, in this case the streamlines should go from a to b, i.e. a well-posed hyperbolic problem for $v(x)$ is

(4.1.26b) $$\frac{\partial v}{\partial t} = \delta\left(\frac{\partial v}{\partial x} + \lambda v\right) \quad , a<x<b \ , \ t>0 \ .$$

with $\delta < 0$.



The size of the constants γ and δ is not important, since they can be absorbed in t, just causing a scaling of (the artificial) time; there is no reason to have a different time scaling for the two equations. Hence we shall use

$$\gamma = 1 \text{ and } \delta = -1$$

for the embedding of the test problem.

Let $u,v \in \mathbb{R}^{N+1}$ be vectors that contain approximations in the grid points of $u(x)$ and $v(x)$, respectively, i.e.

$$u_k \doteq u(x_k), \quad v_k \doteq v(x_k) \ .$$

Define $h_k := x_{k+1} - x_k$. The discretization of (4.1.26) can be done in several ways. Some simple methods like Euler (forward or backward) and trapezoidal rule, all yield a system with a negative logarithmic norm, if a uniform grid is used. For instance a first order discretization of (4.1.26) yields

(4.1.27)

$$\frac{d}{dt}\begin{pmatrix} u \\ v \end{pmatrix} = h^{-1}\left(\begin{array}{cccc:cccc} -1-\mu h & 1 & & & & & & \\ & \ddots & \ddots & & & & & \\ & & -1-\mu h & 1 & & & & \\ & & & -1 & & & & \\ \hdashline & & & & -1 & & & \\ & & & & 1-\lambda h & -1 & & \\ & & & & & \ddots & \ddots & \\ & & & & & & 1-\lambda h & -1 \end{array}\right)\begin{pmatrix} u \\ v \end{pmatrix} + \begin{pmatrix} 0 \\ \vdots \\ 0 \\ h^{-1}\beta \\ h^{-1}\alpha \\ 0 \\ \vdots \\ 0 \end{pmatrix} .$$

Usually, the approximations at the boundary points u_1 and v_{N+1} are incorporated in the inhomogeneity. Here we have chosen to keep them as variables and use the time-derivatives :

$$\frac{du_{N+1}}{dt} = -h^{-1}(u_{N+1}-\beta) \quad , \quad \frac{dv_1}{dt} = -h^{-1}(v_1-\alpha) \; .$$

The reason for this is that the BC of the BVP are inherently part of the Jacobian of the multiple shooting method and that we want to establish a similarity between that Jacobian and the matrix used in this time-dependent system. Note that $u_{N+1}(0) = \beta$ and $v_1(0) = \alpha$ lead to the steady states $u_{N+1}(t) = \beta$ and $v_1(t) = \alpha$.
The logarithmic norm of the matrix in (4.1.27) is equal to the largest eigenvalue of its symmetric part (cf. App.C) and with the use of Gershgorin's circle theorem, one can easily derive that this is negative if $h\lambda \in (0,2)$.

The differential equation (4.1.27) can be rewritten as
(4.1.28a)

$$\frac{d}{dt}\begin{pmatrix} u \\ v \end{pmatrix} = h^{-1} D P^{\top} \left(\begin{array}{cccc:cccc} 1+\mu h & -1 & & & & & & \\ & \ddots & \ddots & & & & & \\ & & 1+\mu h & -1 & & & & \\ & & & 1 & & & & \\ \hdashline & & & & 1-\lambda h & -1 & & \\ & & & & & \ddots & \ddots & \\ & & & & & & 1-\lambda h & -1 \\ & & & & 1 & & & 0 \end{array}\right) \begin{pmatrix} u \\ v \end{pmatrix} + \begin{pmatrix} 0 \\ \vdots \\ 0 \\ h^{-1}\beta \\ h^{-1}\alpha \\ 0 \\ \vdots \\ 0 \end{pmatrix},$$

with

$$(4.1.28b) \qquad D := \operatorname{diag}(\underbrace{-1,\ldots,-1}_{(N+2)\times},\underbrace{1,\ldots,1}_{N\times})$$

and

$$(4.1.28c) \qquad P := \left(\begin{array}{ccc:c} & & & \\ & I_{(N+1)(n-p)} & & \\ & & & \\ \hdashline & & & I_{Np} \\ \hdashline & & I_p & \end{array}\right) \quad , \text{ N.B. in this case } p=1 \text{ and } n-p=1 \; .$$

The matrix in the right hand side of (4.1.28a) corresponds to an approximation of $\tilde{J}$ (cf. (4.1.21)), because $(1+\mu h)$ and $(1-\lambda h)$, approximations of $e^{\mu h}$ and $e^{-\lambda h}$, respectively, fulfil

the same role as B_k and E_k in $\tilde{J}$.

Let $\tilde{J}$ be the basic form of the Jacobian obtained by applying the multiple shooting method to (4.1.25) and assume without loss of generality that $B_k = \exp(\mu h_k)$ and $E_k = \exp(-\lambda h_k)$. We see from the above that a dissipative system

$$\frac{ds}{dt} = \tilde{M}\tilde{J}s \,, \tag{4.1.29}$$

can be obtained if

$$\tilde{M} = DP^{\top} \,, \tag{4.1.30}$$

Then $\tilde{M}\tilde{J}$ has the following form

$$\tilde{M}\tilde{J} = \left(\begin{array}{cccc:cccc} -B_1 & 1 & & & & & & \\ & \ddots & \ddots & & & & & \\ & & -B_N & 1 & & & & \\ & & & -1 & & & & \\ \hdashline & & & & -1 & & & \\ & & & & E_1 & -1 & & \\ & & & & & \ddots & \ddots & \\ & & & & & & E_N & -1 \end{array}\right). \tag{4.1.31}$$

This choice for $\tilde{M}$ does satisfy the requirement (4.1.24a). Let us now consider the other requirement, viz. that (4.1.24b) is moderately bounded. Suppose that the BVP (4.1.25) depends on a parameter s, for instance

$$\begin{pmatrix} \dot{u} \\ \dot{v} \end{pmatrix} = \begin{pmatrix} \mu(s) & 0 \\ 0 & \lambda(s) \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix}. \tag{4.1.32}$$

For any two parameters s and σ

$$\begin{aligned} &|\tilde{M}(s)(\tilde{J}(s) - \tilde{J}(\sigma)|_2 \\ &\quad = |\operatorname{diag}(B_1(s) - B_1(\sigma), .., B_N(s) - B_N(\sigma), 0, 0, E_1(s) - E_1(\sigma), .., E_N(s) - E_N(\sigma))|_2 \\ &\quad = \max_k (\max(|B_k(s) - B_k(\sigma)|_2)) \,. \end{aligned}$$

Suppose that $\dfrac{|\mu(s) - \mu(\sigma)|_2}{|s - \sigma|_2}$ and $\dfrac{|\lambda(s) - \lambda(\sigma)|_2}{|s - \sigma|_2}$ are moderately bounded, say by γ. Now

the differences in E_k and B_k can be estimated by

$$\begin{aligned} |E_k(s) - E_k(\sigma)|_2 &= |e^{-\lambda(s)h_k} - e^{-\lambda(\sigma)h_k}|_2 = |e^{-\lambda(s)h_k}(1 - e^{(\lambda(s)-\lambda(\sigma))h_k})|_2 \\ &= e^{-\lambda(s)h_k}|(\lambda(s)-\lambda(\sigma))|_2 h_k + O(|s-\sigma|_2^2 h_k^2) \\ &\le \gamma h_k e^{-\lambda(s)h_k}|s-\sigma| + O(|s-\sigma|_2^2 h_k^2) . \end{aligned}$$

and

$$\begin{aligned} |B_k(s) - B_k(\sigma)|_2 &= |e^{\mu(s)h_k} - e^{\mu(\sigma)h_k}|_2 = |e^{\mu(s)h_k}(1 - e^{(\mu(\sigma)-\mu(s))h_k})|_2 \\ &= e^{\mu(s)h_k}|(\mu(s)-\mu(\sigma))|_2 h_k + O(|s-\sigma|_2^2 h_k^2) \\ &\le \gamma h_k e^{\mu(s)h_k}|s-\sigma| + O(|s-\sigma|_2^2 h_k^2) . \end{aligned}$$

Hence the local Lipschitz constant for the E_k-components is of the same order of magnitude as γ, but a similar constant for the B_k-components may be considerably larger than γ if $\mu(s) \gg 1$ and h_k is not very small. If we want the Lipschitz constant of $\tilde{M}\tilde{J}$ to be of the same order of magnitude as γ, the B_k-components should be scaled by, for instance, $B_k^{-1}(s)$. This leads to the following preconditioner $\tilde{M}$:

$$\tilde{M}(s) = \operatorname{diag}(-B_1^{-1}(s), -B_2^{-1}(s), \ldots, -B_N^{-1}(s), -I_{n-p}, -I_p, \underbrace{I_p, \ldots, I_p}_{N\times}) \cdot P^T . \tag{4.1.33}$$

In this case

$$\tilde{M}\tilde{J}(s) = \left(\begin{array}{cccc:cccc} -1 & B_1^{-1} & & & & & & \\ & \ddots & \ddots & & & & & \\ & & -1 & B_N^{-1} & & & & \\ & & & -1 & & & & \\ \hdashline & & & & -1 & & & \\ & & & & E_1 & -1 & & \\ & & & & & \ddots & \ddots & \\ & & & & & & E_N & -1 \end{array}\right), \tag{4.1.34}$$

has again a negative logarithmic norm, because the symmetric part of $\tilde{M}\tilde{J}(s)$ is diagonally dominant with negative diagonal elements. Moreover

(4.1.35)

$$\frac{|M(s)(\tilde{J}(s)-\tilde{J}(\sigma))|_2}{|s-\sigma|_2} = \max_k \left(\max\left(\frac{|B_k^{-1}(s)(B_k(s)-B_k(\sigma))|_2}{|s-\sigma|_2}, \frac{|E_k(s)-E_k(\sigma)|_2}{|s-\sigma|_2}\right)\right)$$

$$= \gamma \max_k h_k + O(|s-\sigma|_2 \max_k h_k^2) .$$

The preceding relations were derived for the discretization of the 2-dimensional model problem (4.1.25). Let us now return to the starting point, where $\tilde{J}$ was the basic form (4.1.22) of the original Jacobian. The same idea's as used for the preconditioner for the model problem can be used. If either p or $n-p$ is larger than 1, Gershgorin's circle theorem may not be suitable to estimate the logarithmic norm of $\tilde{M}\tilde{J}$, because $|B_k^{-1}| < 1$ and $|E_k| < 1$ for sufficiently large interval sizes (cf. (4.1.13)) does not imply that any sum of absolute values of row elements is less than 1 as well. Instead a theorem from [St] can be used (cf. Appendix C) which requires the logarithmic norm of the diagonal blocks of $\tilde{M}\tilde{J}$ and the norm of the off-diagonal blocks.
The boundary conditions require special attention. If (4.1.33) is used to precondition $\tilde{J}$, then in (4.1.33) the $(N+1)^{st}$ and $(N+2)^{nd}$ diagonal block would be $-B_b^{(1)}$ and $-B_a^{(2)}$, respectively. Both blocks are non-singular and well-conditioned (see Appendix F). However, the scaling requirements on boundary conditions as made in §1.1 do not guarantee that $\mu_2[-B_a^{(2)}]$ and $\mu_2[-B_b^{(1)}]$ are negative. Hence it is more convenient to incorporate them in $\tilde{M}$ and use

(4.1.36) $$\tilde{M} := \hat{B}^{-1}P^T ,$$

with

(4.1.37) $$\hat{B} = \mathrm{diag}(-B_1, -B_2, \ldots, -B_N, -B_b^{(1)}, -B_a^{(2)}, \underbrace{I_p, \ldots, I_p}_{N\times}) .$$

yielding $-I_p$ and $-I_{n-p}$ at the $(N+1)^{st}$ and $(N+2)^{nd}$ diagonal block of $\tilde{M}\tilde{J}$.

Note that the scaling of $\tilde{M}$ with the B_k^{-1}-components, which was induced by a sensitivity analysis of $\tilde{M}\tilde{J}$, is also favourable in estimating $\mu_2[\tilde{M}\tilde{J}]$ in a higher dimensional case; an estimate of $|B_k^{-1}|$ is available, but $\mu_2[-B_k] < 0$ may not be concluded from the requirements on B_k.

This preconditioner $\tilde{M}$ for the basic form of the Jacobian yields the following form of the product $\tilde{M}\tilde{J}$, this matrix will be denoted by $\hat{J}$:

(4.1.38)
$$\hat{J} := \tilde{M}J = \left(\begin{array}{cccc:cccc} -I_{n-p} & B_1^{-1} & & & & & & \\ & \ddots & \ddots & & & & & \\ & & -I_{n-p} & B_N^{-1} & & & & \\ & & & -I_{n-p} & & & & \\ \hdashline & & & & -I_p & & & \\ & & & & E_1 & -I_p & & \\ & & & & & \ddots & \ddots & \\ & & & & & & E_N & -I_p \end{array}\right).$$

4.1.39 Lemma

Let $0 < \varepsilon < 1$ *and suppose that the interval length* $x_{k+1}-x_k$ *is sufficiently large so that*

(4.1.39a) $\quad \forall_k \;:\; |E_k|_2 < \varepsilon \quad \textit{and} \quad |B_k^{-1}|_2 < \varepsilon\,.$

Then

(4.1.39b) $\quad \mu_2[\hat{J}] < -1+\varepsilon\,.$

Proof

Applying theorem C.4 (see Appendix C) with the same partitioning of blocks as has been used in this section in combination with Gershgorin's circle theorem and the relationship $\mu_2[\hat{J}] = \max\{\tau \mid \tau \in \sigma((\hat{J}+\hat{J}^T)/2)\}$ gives

$$\mu_2[\hat{J}] \le \max_{1\le k\le N+1} \left(\max\left(-1+\frac{b_{k-1}+b_k}{2}\,,\; -1+\frac{e_{k-1}+e_k}{2}\right)\right),$$

$$\text{with} \quad b_k := \begin{cases} |B_k^{-1}|_2 & \text{if } 1\le k\le N\,, \\ 0 & \text{if } k=0 \text{ or } k=N+1\,, \end{cases}$$

$$\text{and} \quad e_k := \begin{cases} |E_k|_2 & \text{if } 1\le k\le N\,, \\ 0 & \text{if } k=0 \text{ or } k=N+1\,. \end{cases}$$

From this we obtain $\mu_2[\hat{J}] < -1+\varepsilon$.

♦

We now finally arrive at the original goal of this section, viz. to obtain a preconditioner $M(s)$ for the Jacobian $J(s)$ of the multiple shooting method. Since

$$J = \hat{Q}\hat{S}^{-1}\hat{P}\cdot\hat{J}\cdot\hat{P}^{\top}SQ^{\top} ,$$

a natural choice for M would be

$$M = T\cdot\hat{B}^{-1}P^{\top}\cdot\hat{P}^{\top}\hat{S}\hat{Q}^{\top} ,$$

where T is a matrix, which we will choose such that $\mu_2[MJ] < 0$. This choice for M yields

(4.1.40) $$MJ = T\cdot\hat{J}\cdot\hat{P}^{\top}SQ^{\top} .$$

Now lemma C.5 states that $\mu_2[\hat{J}] < 0$ implies, that for every non-singular matrix V :

$$\mu_2[V^{\top}\hat{J}V] \le \mu_2[\hat{J}]\ |V^{-1}|_2^{-2} .$$

(i.e. the sign of the logarithmic norm is invariant under a congruent transformation). Comparison with (4.1.40) induces the choice

(4.1.41a) $$T = QS^{\top}\hat{P} ,$$

i.e.

(4.1.41b) $$M(s) = QS^{\top}\hat{P}\cdot\hat{B}^{-1}P^{\top}\cdot\hat{P}^{\top}\hat{S}\hat{Q}^{\top} .$$

The logarithmic norm of MJ is estimated by the following theorem.

4.1.42 Theorem

Let $0 < \varepsilon < 1$ *and suppose that the interval length* $x_{k+1} - x_k$ *is sufficiently large so that*

(4.1.42a) $$\forall_k \quad : \quad |E_k|_2 < \varepsilon \quad \text{and} \quad |B_k^{-1}|_2 < \varepsilon .$$

Let κ_{lin} *be the conditioning constant of the linearized BVP at* $y(x)$. *Then*

(4.1.42b) $$\mu_2[MJ] < \frac{-1+\varepsilon}{(\kappa_{lin}^2+\kappa_{lin}+1)} .$$

Proof

From lemma 4.1.39 we obtain $\mu_2[\hat{J}] < -1+\varepsilon$. Now we apply Lemma C.5 (see Appendix C) with $V := \hat{P}^{\top}SQ^{\top}$, yielding $\mu_2[MJ] \le \mu_2[\hat{J}]\ |V^{-1}|_2^{-2}$. One can easily prove that $|R_k|_2 < \kappa_{lin}$ (see Appendix F) and thus $|V^{-1}|_2^2 < (\kappa_{lin}^2 + \kappa_{lin} + 1)$, hence

$$\mu_2[MJ] < \frac{-1+\varepsilon}{(\kappa_{lin}^2+\kappa_{lin}+1)} .$$

♦

Conclusion

Based on the proper information flow for the embedding of BVP's in hyperbolic time dependent systems and the requirement to reduce the sensitivity of $\boldsymbol{MJ}$ for changes in $\boldsymbol{s}$ we formed the preconditioner

(4.1.43a) $$\boldsymbol{M(s) = QS^{\top}\hat{P}\cdot\hat{B}^{-1}P^{\top}\cdot\hat{P}^{\top}\hat{S}\hat{Q}^{\top}} .$$

Hence

(4.1.43b) $$\boldsymbol{MJ = QS^{\top}\hat{P}\cdot\hat{J}\cdot\hat{P}^{\top}SQ^{\top}} .$$

The preconditioner essentially does three things :

- decouple the growing and decaying solution modes,
- place the initial conditions before the integration of the decaying modes (which is a more natural ordering),
- invert the integration of the growing modes on every subinterval.

♦

§4.2 Comparison of the preconditioner with $-J^{-1}$

In this section we will compare the preconditioner $M(s)$ of the previous section with the Davidenko preconditioner $-J^{-1}$; in particular we compare their formula's and the ratio of

$$\frac{|M(s)(J(s)-J(\sigma))|}{|-J^{-1}(s)(J(s)-J(\sigma))|}, \tag{4.2.1}$$

for any two vectors s,σ near s^*. Additionally we investigate the form of $J(s)-J(\sigma)$ for a simple problem in more detail.

Recall from the previous section that

$$J = \hat{Q}\hat{S}^{-1}\hat{P}\tilde{J}\hat{P}^{\top}S Q^{\top} \;\Rightarrow\; J^{-1} = Q S^{-1}\hat{P}\tilde{J}^{-1}\hat{P}^{\top}\hat{S}\hat{Q}^{\top} \tag{4.2.2}$$

and

$$M = Q S^{\top}\hat{P}\cdot\hat{B}^{-1}P^{\top}\cdot\hat{P}^{\top}\hat{S}\hat{Q}^{\top}. \tag{4.2.3}$$

First we assume that $Q_k = S_k = I_n$, $k\in\{1, \ldots, N+1\}$, to obtain the 'skeleton' versions of M and $-J^{-1}$. In the previous section we found that the permutation matrix P (4.1.28c) can be used to transform J into a block form, with the left upper block representing the integration of the growing modes and the right lower block representing the integration of the decaying modes :

$$J = \hat{P}\tilde{J}\hat{P}^{\top} = \hat{P}P\begin{pmatrix}\tilde{J}^{11} & 0\\ 0 & \tilde{J}^{22}\end{pmatrix}\hat{P}^{\top}, \tag{4.2.4}$$

with

$$\tilde{J}^{11} = \begin{pmatrix} B_1 & -I_{n-p} & & \\ & \ddots & \ddots & \\ & & B_N & -I_{n-p} \\ & & & B_b^{(1)} \end{pmatrix} \quad\text{and}\quad \tilde{J}^{22} = \begin{pmatrix} B_a^{(2)} & & & \\ E_1 & -I_p & & \\ & \ddots & \ddots & \\ & & E_N & -I_p \end{pmatrix}. \tag{4.2.5}$$

The inverse of J can also be expressed in terms of $\tilde{J}^{11}$ and $\tilde{J}^{22}$:

$$-J^{-1} = \hat{P}\begin{pmatrix}\left(-\tilde{J}^{11}\right)^{-1} & 0\\ 0 & \left(-\tilde{J}^{22}\right)^{-1}\end{pmatrix}P^{\top}\hat{P}^{\top}, \tag{4.2.6}$$

(4.2.7a)
$$\left(-\tilde{J}^{22}\right)^{-1} = \begin{pmatrix} -\left(B_a^{(2)}\right)^{-1} & & & & \\ -E_1\left(B_a^{(2)}\right)^{-1} & I_p & & & \\ -E_2E_1\left(B_a^{(2)}\right)^{-1} & E_2 & I_p & & \\ \vdots & & \ddots & \ddots & \\ -E_N E_{N-1}..E_1\left(B_a^{(2)}\right)^{-1} & E_N E_{N-1}..E_2 & \cdots & E_N & I_p \end{pmatrix},$$

and

(4.2.7b)
$$\left(-\tilde{J}^{11}\right)^{-1} = \begin{pmatrix} -B_1^{-1} & -B_1^{-1}B_2^{-1} & \cdots & -B_1^{-1}B_2^{-1}..B_N^{-1} & -B_1^{-1}B_2^{-1}..B_N^{-1}\left(B_b^{(1)}\right)^{-1} \\ 0 & -B_2^{-1} & \cdots & -B_2^{-1}B_3^{-1}..B_N^{-1} & -B_2^{-1}B_3^{-1}..B_N^{-1}\left(B_b^{(1)}\right)^{-1} \\ & \ddots & & \vdots & \vdots \\ & & & -B_N^{-1} & -B_N^{-1}\left(B_b^{(1)}\right)^{-1} \\ 0 & & & 0 & -\left(B_b^{(1)}\right)^{-1} \end{pmatrix}.$$

The matrix $\boldsymbol{M}$ reads

(4.2.8)
$$\begin{aligned} \boldsymbol{M} &= \hat{\boldsymbol{P}}\hat{\boldsymbol{B}}^{-1}\boldsymbol{P}^{\top}\hat{\boldsymbol{P}}^{\top} \\ &= \hat{\boldsymbol{P}}\,\mathrm{diag}\left(-B_1^{-1},-B_2^{-1},\ldots,-B_N^{-1},-\left(B_b^{(1)}\right)^{-1},-\left(B_a^{(2)}\right)^{-1},I_p,\ldots,I_p\right)\boldsymbol{P}^{\top}\hat{\boldsymbol{P}}^{\top}. \end{aligned}$$

These relations show that, after an appropriate permutation, $\boldsymbol{M}$ is only the diagonal part of $-\boldsymbol{J}^{-1}$. Apparently the required minus-signs, derived in the previous section from the correct information flow in hyperbolic problems (and hence from the condition that the logarithmic norm of $\boldsymbol{MJ}$ is negative), are naturally present in $-\boldsymbol{J}^{-1}$.

Let s and σ be two vectors in $\mathbb{R}^{n(N+1)}$ in the neighbourhood of s^*. Define $\delta\boldsymbol{J} := \boldsymbol{J}(s)-\boldsymbol{J}(\sigma)$. Since the transformations $\boldsymbol{Q}$ and $\boldsymbol{S}$ are not necessarily suitable for $\delta\boldsymbol{J}$, we give the explicit

form of M in order to estimate $|M\delta J|$.

(4.2.9)
$$M = \begin{pmatrix} -B_1^{-1} & 0 & & & & & 0 & 0 \\ 0 & 0 & & & & & 0 & -\left(B_a^{(2)}\right)^{-1} \\ 0 & 0 & -B_2^{-1} & 0 & & & & \\ 0 & I_p & 0 & 0 & & & & \\ & & \ddots & & \ddots & & & \\ & & & & 0 & 0 & -B_N^{-1} & 0 \\ & & & & 0 & I_p & 0 & 0 \\ & & & & & & 0 & 0 \\ & & & & & & 0 & I_p \end{pmatrix} \begin{matrix} -\left(B_b^{(1)}\right)^{-1} & 0 \\ 0 & 0 \end{matrix}.$$

One can see that M picks up every block in δJ exactly once, yielding

(4.2.10)

$$|M\delta J|_2 \leq \max\Big(|(0\,|\left(B_a^{(2)}\right)^{-1})(B_a(s)-B_a(\sigma))|_2 , |(\left(B_b^{(1)}\right)^{-1}|0)(B_b(s)-B_b(\sigma))|_2 , \max_k |(B_k^{-1}|0)(Y_k(x_{k+1};s_k)-Y_k(x_{k+1};\sigma_k))|_2 , |(0\,|I)(Y_k(x_{k+1};s_k)-Y_k(x_{k+1};\sigma_k))|_2 \Big) .$$

If we assume that $|B_a(s) - B_a(\sigma)|$ and $|B_b(s) - B_b(\sigma)|$ are negligible compared to the differences in $Y_k(x_{k+1})$ (the latter difference will be of the order of magnitude of the conditioning constant of the local IVP), then

(4.2.11)
$$|M\delta J|_2 \approx |\delta J|_2 .$$

In $J^{-1}\delta J$ there is a build-up of errors, which originates from the fact that $J(s)$ represents the integration of the BVP over $[a,b]$. Hence the difference between the p lowest rows of $Y_1(x_2;s_1) - Y_1(x_2;\sigma_1)$ is 'picked-up' during forward integration of the decaying modes and contaminates (in a damped form) all subsequent $J^{-1}\delta J$ components. Similarly are the upper $(n-p)$ rows of $Y_N(x_{N+1};s_N) - Y_N(x_{N+1};\sigma_N)$ present in all components $J^{-1}\delta J$ due to backward integration of the growing modes. A closed form for the error in $-J^{-1}\delta J$ cannot easily be obtained. However, $|-J^{-1}\delta J|$ may be up to N times larger $|M\delta J|$.

Since we have assumed that $Q_k = S_k = I_n$, the left upper $(n-p)\times(n-p)$ block of $Y_k(x;s)$ contains the growing solution modes. If $|\, s-\sigma \,|$ is sufficiently small, we expect that the same block of $Y_k(x;\sigma)$ also contains a major part of the growing solution modes of the corresponding BVP. Due to the scaling of the local fundamental solutions the changes in the growing modes may be unpleasantly large. Fortunately, both M and $-J^{-1}$ premultiply these changes by a damping factor B_k^{-1}. This worked well for the model problem used in §4.1. However, B_k^{-1} is not always able to control differences in Y_k sufficiently, in particularly not if the directions of the solution modes change. We will illustrate this with an example.

4.2.12 Example

Suppose that, after a change of coordinate system, $J(s)$ is a discretization of the BVP

$$\text{(4.2.12a)} \qquad \dot{z} = Az \qquad \text{with } A = \begin{pmatrix} \mu & 0 & 0 \\ 0 & \nu & 0 \\ 0 & 0 & -\lambda \end{pmatrix}, \quad \mu,\nu,\lambda > 0 \text{ and } \mu > \nu ,$$

with boundary conditions

$$\text{(4.2.12b)} \qquad \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} z(a) + \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix} z(b) = \beta .$$

To simplify matters we assume that λ,μ and ν do not depend on x. The situation may occur that for some σ, near s, the Jacobian $J(\sigma)$ with respect to the same coordinate system as $J(s)$, can be viewed as the discretization of

$$\text{(4.2.12c)} \qquad \dot{z} = QAQ^{\top}z \qquad \text{with } Q = \begin{pmatrix} c & -d & 0 \\ d & c & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad 0<c,d<1 \text{ and } c^2+d^2 = 1 .$$

The left upper block of Q is a plane rotation (we use d, instead of s to denote the sinus-value to prevent confusion with the shooting vector s). Generally the values of λ,μ and ν will have changed as well, however, this type of changes has already been considered in §4.1, here we want to concentrate on the effect of changes of direction. The difference in the BVP's can be estimated by

$$|A-QAQ^{\top}|_2 = \left|\begin{pmatrix} \mu-c^2\mu-d^2\nu & cd(\nu-\mu) & 0 \\ cd(\nu-\mu) & \nu-c^2\nu-d^2\mu & 0 \\ 0 & 0 & 0 \end{pmatrix}\right|_2 = (\mu-\nu)d\ .$$

In a multiple shooting context this difference is bounded by $\kappa|\, s-\sigma\,|$, with κ an upper bound on the conditioning constants of the IVP's on the subintervals. The difference in the fundamental solution $Y_k(x)$ can be described by

$$\begin{aligned} Y_k(x_{k+1};s)-Y_k(x_{k+1};\sigma) &= e^{Ah_k}-Qe^{Ah_k}Q^{\top} \\ &= \begin{pmatrix} (1-c^2)e^{\mu h_k}-d^2e^{\nu h_k} & -cd(e^{\mu h_k}-e^{\nu h_k}) & 0 \\ -cd(e^{\mu h_k}-e^{\nu h_k}) & (1-c^2)e^{\nu h_k}+d^2e^{\mu h_k} & 0 \\ 0 & 0 & 0 \end{pmatrix}. \end{aligned}$$

The second row of this difference contains $d\cdot\exp(\mu h_k)$. This term will be large for large μ and since $B_k = \begin{pmatrix} e^{\mu h_k} & 0 \\ 0 & e^{\nu h_k} \end{pmatrix}$, we see that premultiplication of the difference in $Y_k(x_{k+1})$ by $\begin{pmatrix} B_k^{-1} & 0 \\ 0 & 0 \end{pmatrix}$ reduces the difference in the second row by a factor $\exp(-\nu h_k)$ only. Hence if all subintervals have the same size

$$|M\delta J|_2 \approx cde^{(\mu-\nu)h} \approx \kappa\frac{e^{(\mu-\nu)h}}{\mu-\nu}\,|s-\sigma|$$

and

$$\begin{aligned} |-J^{-1}\delta J|_2 &\approx cde^{(\mu-\nu)h}\sum_{j=0}^{N-1} e^{-\nu jh} \approx \kappa\frac{e^{(\mu-\nu)h}}{\mu-\nu}\,\frac{1-e^{-\nu(b-a)}}{1-e^{-\nu h}}\,|s-\sigma| \\ &\approx N\frac{1-e^{-\nu(b-a)}}{\nu(b-a)}\cdot\kappa\frac{e^{(\mu-\nu)h}}{\mu-\nu}\,|s-\sigma|\,. \end{aligned}$$

♦

This example illustrates that small changes in the directions of the growing modes may cause an error in $B_k(s)$ - $B_k(\sigma)$ which is not controlled by $B_k^{-1}(s)$. The adverse effects on $|\,M\delta J\,|$ of changes in the directions of the solution modes may be overcome by multiplying the preconditioner M by a factor $\exp(-\mu h)$. However,

$$\mu_2[e^{-\mu h}MJ] = e^{-\mu h}\mu_2[MJ]\ ,$$

i.e. the logarithmic norm will be closer to zero. Consequently the premultiplication does not result in an enlargement (or reduction) of the size of the convergence area of

$$\frac{ds}{dt} = \boldsymbol{M}(s)f(s) \ ,$$

as described in lemma 3.1.18. Although this example assumed a simple structure of the solution space, similar effects may well occur for more complex BVP's.

The structure of $\boldsymbol{M}$ and $-\boldsymbol{J}^{-1}$ as depicted in (4.2.8) and (4.2.6),(4.2.7) is valid if $Q_k = S_k = I_n$. Otherwise the situation is more complex. Let $-\boldsymbol{J}^{-1}$ and $\boldsymbol{M}$ be partitioned into $(N+1)^2$ square blocks of size $n \times n$: $(-\boldsymbol{J}^{-1})_{ij}$ and $(\boldsymbol{M})_{ij}$ respectively. And let the orthogonal matrices Q_k be partitioned according to

$$(4.2.13) \qquad Q_k = \underset{\leftrightarrow \ n-p}{\left(Q_{k,1}\right.} \underset{\leftrightarrow \ p}{\left| Q_{k,2}\right)} .$$

Some calculus shows that

(4.2.14)

$$(-\boldsymbol{J}^{-1})_{i,j} = \begin{cases} -Q_{i,1}B_i^{-1}B_{i+1}^{-1}..B_j^{-1}(Q_{j+1,1}^{\top} + R_{j+1}Q_{j+1,2}^{\top}) & , i \le j \le N , \\ (Q_{i,2} - Q_{i,1}R_i)E_{i-1}E_{i-2}..E_{j+1}Q_{j+1,2}^{\top} & , j < i \le N+1 , \\ \left(-Q_{i,1}B_i^{-1}B_{i+1}^{-1}..B_N^{-1}\left(B_b^{(1)}\right)^{-1} \right| (Q_{i,1}R_i - Q_{i,2})E_{i-1}E_{i-2}..E_1\left(B_a^{(2)}\right)^{-1}\Big) & \\ & , j = N+1 . \end{cases}$$

and

$$(4.2.15) \qquad (\boldsymbol{M})_{i,j} = \begin{cases} -(Q_{i,1} + Q_{i,2}R_i^{\top})B_i^{-1}(Q_{i+1,1}^{\top} + R_{i+1}Q_{i+1,2}^{\top}) & , i = j , \\ Q_{i,2}Q_{i,2}^{\top} & , i = j+1 , \\ \left(0 \right| -Q_{1,2}\left(B_a^{(2)}\right)^{-1}\Big) & , j = N+1 , i = 1 , \\ \left(-(Q_{N+1,1} + Q_{N+1,2}R_{N+1}^{\top})\left(B_b^{(1)}\right)^{-1} \right| 0\Big) & , i = j = N+1 . \\ 0 & , \text{otherwise} \end{cases}$$

The preconditioner $\boldsymbol{M}$ is sparse, in particular for larger values of N, while $-\boldsymbol{J}^{-1}$ cannot be expected to have a special zero-structure. The block diagonal and the first lower subdiagonal of $\boldsymbol{M}$ and $-\boldsymbol{J}^{-1}$ are not identical unless $R_k = 0$, i.e. the subspaces of growing and decaying modes are orthogonal. The difference between the two matrices can be interpreted as follows. Let $Z(x)$ be the fundamental solution on $[a,b]$ of the linearized BVP of (4.1.1) at $y(x;s)$, such that

$$Z(a) = Q_1 .$$

According to the consistency theory in §1.2, the first $n-p$ columns of $Z(\xi)$ span the space of growing modes integrated up to $x = \xi$. From the relationships between Q_k and R_k one can derive that

(4.2.16a) $$\text{range}(Z(x_k)(I-P)) = \text{range}(Q_{k,1}),$$

(4.2.16b) $$\begin{aligned}\text{range}(Z(x_k)P) &= \ker(Q_{k,1}^T + R_k Q_{k,2}^T) = \text{range}(Q_{k,1} + Q_{k,2}R_k^T)^\perp \\ &= \text{range}(Q_{k,2} - Q_{k,1}R_k),\end{aligned}$$

with $P = \begin{pmatrix} 0 & 0 \\ 0 & I_p \end{pmatrix}$. (see Appendix F). Hence the diagonal blocks of M project into the subspace orthogonal to the subspace of decaying modes, while the diagonal blocks of $-J^{-1}$ project into the space of growing modes (both subspaces are equal iff $R_k = 0$).

Conclusion

On comparing the preconditioner M with $-J^{-1}$, we have seen that M is a part of $-J^{-1}$, if the subspaces of growing and decaying modes are orthogonal. The matrix M treats the differences in the fundamental solutions (as present in δJ) locally, whereas $-J^{-1}$ transports them either to the begin or end point of the interval. Due to the 'local' nature of M, we may expect that for s,σ near s^*, the value of

$$\frac{|M(s)(J(s)-J(\sigma))|}{|s-\sigma|}$$

may be up to a factor N smaller than

$$\frac{|-J^{-1}(s)(J(s)-J(\sigma))|}{|s-\sigma|}$$

♦

§4.3 Numerical results

The idea of preconditioned time stepping with the preconditioner presented in the previous sections is implemented in a code called TS. We compare this code with two other multiple shooting codes, viz. MUSN and RWPM (see [AsMaRu,MaSt], [HeBe] resp.); both use variants of Newton's method to solve the non-linear equations. The results, presented below, indicate that the time stepping algorithm can increase the convergence domain, sometimes even on problems that, though well-conditioned, do not satisfy the conditions of Th.4.1.42.

In the TS-program the required tolerance for the solution is denoted by TOL. The convergence criterion used is $| M(s^j)f(s^j) |_2 <$ TOL or $| M(s^j)f(s^{j+1}) |_2 <$ TOL, where $f(s)$ must be evaluated with an accuracy smaller than TOL. The program employs the preconditioner $M(s)$ defined in the previous section and uses the mixed Euler method

$$s^{j+1} = s^j + h_j M(s^j) f(s^{j+1}) \tag{4.3.1}$$

for time integration. The discretization error hereof is bounded by the user prescribed tolerances ATOL and RTOL for the absolute and relative error, respectively. Based on these tolerances the TS-program determines the step size h_j. The iterate s^{j+1} is obtained by a modified Newton's method using the Jacobian at s^j only and not at any intermediate point. We want to approximate s^{j+1} with a tolerance NTOL. If this is not obtained within three iterations the step size h_j is halved. This process continues until a sufficiently accurate approximation of s^{j+1} is obtained or h_j drops below a (user set) minimum value. Since the path $s(t)$ is followed with an error ATOL + RTOL $| s^j |$ it would be overdone to approximate s^{j+1} with an essentially smaller error. Hence we set
NTOL = min(ATOL + RTOL$| s^j |$, 10^{-2}); the latter term is used to guarantee at least two correct numbers in s^{j+1}.

The local IVP's on the subintervals are integrated using RKF45 as implemented in MUSN. This process is controlled by a parameter ER. During the RKF45 integration we require the discretization error to be less than ER $(1 + | s^j |)$; i.e. ER is a combined absolute and relative tolerance. Of course this tolerance has to be less than the required tolerance TOL for the solution of the BVP at the end of the time stepping process. However, if the vector s^j is still far from the solution a small value of ER will require more work without increasing the convergence speed considerably. Most components of $f(s)$ contain the difference between two solutions of local IVP's, hence cancelation will reduce the amount of accurate numbers in $f(s^j)$, once s^j is close to the solution s^*. Since at every step $f(s^j)$ should have at least 1 or 2 significant numbers, the tolerance ER should be at most the error in s^j divided by a safety factor, which we chose to be 100. Hence the user has to give an initial value for ER and during the process ER is taken as the minimum of its pre

vious value and $10^{-2}|\, M(s^j)f(s^j)\,|$.

A good indication for the computational costs of BVP-solving algorithms is the number of evaluations of the function defining the field of directions of the BVP ($h(x,y)$ in (4.1.1)). In the tables in this section this quantity is denoted by #f$^{\text{ion}}$-calls (N.B. this is not equal to the number of times $f(s)$ is computed).

4.3.2 Example 1

Consider the problem attributed to Troesch [Tr]

$$\begin{aligned} \ddot{z}(x) &= \lambda \sinh(\lambda z) \qquad , 0<x<1\,, \\ z(0) &= 0\,, \\ z(1) &= 1\,. \end{aligned} \tag{4.3.2a}$$

This has been used as a test problem by many authors (e.g. [DePeRe,ScWa]). The linearization of this problem at its exact solution is exponentially dichotomic with growth factors are of the order of magnitude of λe^{λ} and $\lambda e^{-\lambda}$, respectively. Due to this, forward integration becomes inaccurate over longer subintervals and the non-linear function $f(s)$ is very sensitive to small changes of the starting vector s^j in the direction of the growing mode; in fact the local IVP's are ill-posed, in particular at the end of the interval.

We look at the effect of choosing too large initial values s^j and uniform (i.e. non-optimal) subintervals for rather small values of λ ($\lambda \leq 5$). For the parameters we choose ATOL = RTOL = 10^{-1}, ER = 10^{-3} and set the required tolerance TOL = 10^{-6}. The initial guess to the solution is

$$z(x) = x \;,\quad \dot{z}(x) = 1\,. \tag{4.3.2b}$$

The results (see table 4.1) clearly show that if the Newton's method works it requires less iterations and function calls than time stepping, as has to be expected. However, the time stepping algorithm can solve the problem on coarser grids, i.e. for more difficult cases.
For all choices of λ the upper triangular matrices U_k (see (4.1.12)) satisfy the condition that $|\, B_k^{-1}\,| < 1$ and $|\, E_k\,| < 1$, and coarser grids gave smaller values, i.e. the IVP

$$\frac{ds}{dt} = M(s)f(s) \qquad , t>0\,,$$

is stronger attractive. This does not appear from the number of required iterations, because for coarser grids the initial value of $|\, M(s)f(s)\,|$ is larger and the step size h_j increases slower, since the Newton process to solve (4.3.1) requires a somewhat more careful treatment.
However, it should be clear that once the time stepping method has reached a reasonably small residual, one should switch to full Newton in practice; this would make the comple-

xity for the combined method lower (on top of its, more important, better convergence behaviour).

Troesch problem; TOL = 10^{-6}

		MUSN			TS		
λ	subint	iter	result	#f^{ion}-calls	steps	result	#f^{ion}-calls
2	1	3	fail	3, 350	21	conv	3, 228
2	5	3	conv	2, 923	18	conv	5, 027
3	5	6	fail	16, 955	23	conv	10, 314
3	10	4	conv	6, 710	22	conv	13, 684
4	10	11	fail	38, 625	32	conv	35, 482
4	15	6	conv	12, 978	30	conv	36, 831
5	15(17)*	1	exp.overflow		52	conv	73, 002
5	20(22)*	11	fail	80, 787	50	conv	83, 000
5	25	11	conv	55, 875	51	conv	93, 050

* The code added two shooting points near $x = 1$, since the increase over the subintervals exceeded 10^2

Table 4.1

4.3.3 Example 2

The following problem has been proposed in [Ho] and describes the flow between two rotating discs

$$
\begin{aligned}
\dot{y}_1 &= y_2 \,, \\
\dot{y}_2 &= y_3 \,, \\
\dot{y}_3 &= -\frac{(3-n)}{2} y_1 y_3 - n y_2^2 + 1 - y_4^2 + s y_2 \qquad , \; 0 < x < \infty \,, \\
\dot{y}_4 &= y_5 \,, \\
\dot{y}_5 &= -\frac{(3-n)}{2} y_1 y_5 - (n-1) y_2 y_4 + s(y_4 - 1) \,,
\end{aligned}
\tag{4.3.3a}
$$

with boundary conditions

$$
y_1(0) = y_2(0) = y_4(0) = 0 \,, \quad y_2(\infty) = 0 \,, \quad y_4(\infty) = 1 \,. \tag{4.3.3b}
$$

In practice a (large) value L is taken as endpoint of the interval. Both in [RoSh] and [DePeRe] (4.3.3) is used as a test problem with the parameter set $n = -0.1$, $s = 0.2$.
In [RoSh] $L = 11.3$ is the largest endpoint for which convergence is reached using continuation in L. The algorithm proposed in [DePeRe] can solve the BVP by continuation in L for $L \leq 15$ with forward shooting and $L \leq 132$ with backward shooting. This different

behaviour of forward and backward shooting is due to the fact that the growth factor of the strongest growing mode is essentially larger than the absolute value of the decay factor of the strongest decaying mode. Hence local end point value problems are less ill-conditioned than initial value problems. Using as initial guess

$$(4.3.3c) \qquad \begin{aligned} y_1 &= -x^2 e^{-x} \;\; ; \;\; y_2 = \dot{y}_1 \;\; ; \; y_3 = \dot{y}_2 \;\; ; \\ y_4 &= 1 - e^{-x} \;\; ; \;\; y_5 = \dot{y}_4 \, , \end{aligned}$$

the codes tested here do not encounter this problem.

As the solution mainly shows activity near its initial point, we choose a grid which is basically uniform, but for its first subinterval which is halved. We use three different codes to solve this problem, viz. MUSN, the TS-code and RWPM and look for the coarsest grid on which a solution was obtained with accuracy 10^{-6}. For the TS-code the parametervalues were ATOL = RTOL = 10^{-1}, $h_0 = 10^{-1}$, ER = 10^{-1}. Although the linearized problem has three eigenvalues with negative real part, the rapid rotation of two decaying modes caused $| E_k |$ to exceed 1 on more than half the subintervals. Nevertheless convergence was reached quite easily (in about 20 to 30 steps) even on coarser grids, than either of the two other codes could handle.

L	Least number of subintervals required		
	MUSN	RWPM	TS
12	13	7	6
15	20	9	8
20	27	12	10
30	39	18	14
132	169	73	58

Table 4.2

An even more interesting picture occurs if we plot the amount of BVP-evaluations versus the number of gridpoints for a fixed value of L (L = 15 in Figure 4.1). Even though the TS-code used is not optimal (it does not switch to the Newton method near the convergence point) it performed cheaper than RWPM for coarse grids. This is due to the fact that for those grids over 90% of the iterations in RWPM are damped Newton steps with damping factor between 10^{-4} and 10^{-3}. For finer grids the Newton algorithm in RWPM speeds up considerably, whereas the TS-code does not require essentially less steps. This illustrates that the time stepping algorithm does not only serve its purpose of enlarging the convergence domain, but can occasionally even reduce the computational costs.

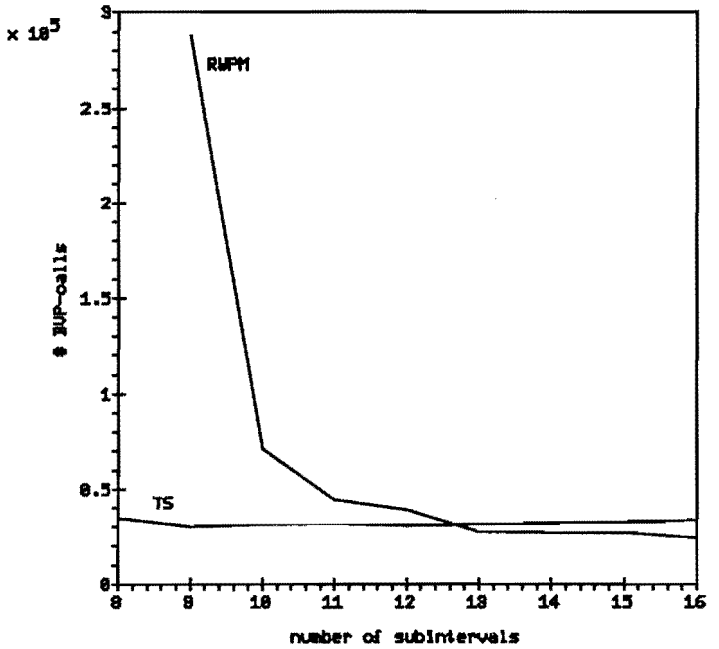


Figure 4.2

♦

Next we want to see the effect of using different values of ER, i.e. the accuracy with which the local IVP's are solved. To this end we solve the BVP's (4.3.3a,b) with $L = 11.3$, $n = -0.1$, $s = 0.2$ and (4.3.2a) with $\lambda = 4$ with the time stepping algorithm for various values of ER with 10 equidistant shooting points, required tolerance TOL = 10^{-6}, initial step size $h_0 = 0.1$, initial guess (4.3.3c), (4.3.2b) resp. and ATOL = RTOL = 10^{-1}. The results are shown in Table 4.3 and 4.4 respectively.

For both test problems the number of time steps did not vary significantly for different values of ER. For larger ER the norm of the residual $M(s^j)f(s^j)$ reduces just a little more slowly. The value of ER has considerable influence on the amount of f^{ion}-calls, since a smaller initial value of ER stands for a more accurate computation of $y_k(x;s_k)$.
In order to decrease the amount of work, one should try to minimize the number of f^{ion}-evaluations and hence choose a large ER. However, this does harbour the danger of divergence of the process, especially for sensitive problems.

Time stepping algorithm for (4.3.3a,b) with $n = -0.1$; $s = 0.2$; $L = 11.3$

ER	steps	#f^{ion}-calls	$\mid R_k \mid_{max}$	max(diag(B_k^{-1}))	max(diag(E_k))
10^{-1}	19	26, 965	[1.9 , 2.9]	[0.6 , 0.9]	[1.7 , 2.2]
10^{-3}	19	28, 784	[1.9 , 2.9]	[0.6 , 0.9]	[1.7 , 2.2]
10^{-5}	19	36, 527	[1.9 , 2.9]	[0.6 , 0.9]	[1.7 , 2.2]

Table 4.3

Time stepping algorithm for (4.3.2a) with $\lambda = 4$

ER	steps	#f^{ion}-calls	$\mid R_k \mid_{max}$	max(diag(B_k^{-1}))	max(diag(E_k))
10^{-1}	30	28, 849	42 ↘ 15	0.92	0.92
10^{-3}	31	30, 393	58 ↘ 15	0.92	0.92
10^{-5}	31	35, 424	58 ↘ 15	0.92	0.92

Table 4.4

The fourth column shows the development of the maximum of $\mid R_k \mid$ during the process. This illustrates quite clearly that the decoupling of the growing modes is much better for Holt's problem than for the Troesch' problem. Additionally we tabulate the range of the maximum values of the diagonal elements of E_k and B_k^{-1} at the various steps, these are indicative for $\mid E_k \mid$ and $\mid B_k^{-1} \mid$ resp.. This shows that the conditions of Th.4.1.42 are satisfied for Troesch' problem, but not for problem (4.3.3a,b).

5 A generalised multiple shooting method

In chapter 2 we described the multiple shooting method for non-linear BVP's and mentioned that in the presence of exponentially growing modes problems may occur, like non-existence of local solutions, serious error amplification and/or a small convergence domain for Newton's method. The source of this trouble lies in the use of initial value conditions for the local problems, which are not able to control growing modes properly.

Some global methods do not encounter the unpleasant features mentioned for IVP methods; global methods, however, may require a larger amount of memory space. Therefore it is an attractive idea to combine the virtues of both classes. To this end the interval $[a,b]$ is divided into subintervals, but now boundary conditions (rather than initial conditions) are defined for a local solution. In particular, one should try and solve these local BVP by a 'global' method rather than an IVP method. This idea has the following advantages. First it results in a more economical memory usage and it renders a potential parallel feature as well. Second it allows for the better convergence and stability properties of global methods and third, as a useful byproduct, it gives an opportunity to 'localize' unpleasant non-linearities, while at the same time the coarse level non-linear equation might become 'easier' to solve.

The outline of this chapter is as follows. In the first section we describe the proposed method in more detail and address the choice of local boundary conditions. Local convergence of the method is proven in §2 and a tolerance strategy based upon this proof is given. As we have two types of non-linear problems, viz. a sequence of local BVP and a global equation, it is also investigated how these two interfere. In section 3 we describe our implementation and give some numerical results. The chapter is concluded by some considerations about parallel implementation of the method.

§5.1 Unbiased multiple shooting

In the section 2.3 we have seen that the convergence domain and behaviour of the Newton's method, as applied in multiple shooting processes, is influenced by the conditioning constant of the local problems. This renders the idea that the convergence behaviour may be improved by defining well conditioned boundary value problems on a set of subintervals instead of IVP's as is done in ordinary multiple shooting. So in order to solve the BVP

$$\dot{y} = h(x,y) \quad , \quad a<x<b \ , \tag{5.1.1a}$$

$$g(y(a),y(b)) = 0 \ , \tag{5.1.1b}$$

the interval $[a,b]$ is divided into N subintervals $[x_k, x_{k+1}]$, $1 \le k \le N$, with

$$a = x_1 < x_2 < \ldots < x_{N+1} = b \ .$$

On each subinterval we seek to solve (with a global method) the BVP

$$\dot{y} = h(x,y) \quad , \quad x_k<x<x_{k+1} \ , \tag{5.1.2a}$$

$$A_k \lim_{x \downarrow x_k} y(x) + B_k \lim_{x \uparrow x_{k+1}} y(x) = s_k \ , \tag{5.1.2b}$$

with $s_k \in \mathbb{R}^n$. The local BC should be such that $(A_k \mid B_k)$ is of full rank and has orthonormal rows (cf. assumption 1.1.16). The solutions of the local problems, assuming they exist, are denoted by $y_k(x;s_k)$; $y(x;s)$ is the function, defined globally on $[a,b]$, that is equal to $y_k(x;s_k)$ on $(x_k, x_{k+1}]$ and satisfies $y(a;s) = y_1(a;s_1)$. An approximation of $y_k(x;s_k)$ will be denoted by $z_k(x;s_k)$ and $z(x;s)$ will be defined similar to $y(x;s)$ as the concatenation of the local approximations. The unknown vectors have to be determined, such that $y(x;s)$ is continuous and satisfies the boundary conditions. Hence they have to be the solution of a set of equations similar to the ones used in the 'original' shooting method (see (2.1.6), (2.1.7)), viz.

$$f(s;z) = 0 \ \text{with} \quad s^\top := (s_1^\top, s_2^\top, \ldots, s_N^\top) \quad \text{and} \quad f \in C^1(\mathbb{R}^{nN} \to \mathbb{R}^{nN}) \tag{5.1.3}$$

and $f(s;z)$ defined by

$$f(s;z(x;s)) := \begin{pmatrix} z_1(x_2;s_1) - z_2(x_2;s_2) \\ z_2(x_3;s_2) - z_3(x_3;s_3) \\ \vdots \\ z_{N-1}(x_N;s_{N-1}) - z_N(x_N;s_N) \\ g(z_1(x_1;s_1), z_N(x_{N+1};s_N)) \end{pmatrix} . \tag{5.1.4}$$

As in the ordinary multiple shooting method, the Jacobian of $f(s;z)$ can be formulated in terms of a linearization of (5.1.2). The notation used will be similar to the one used in

chapter 4. However, in the previous chapter the linearizations were fully determined by the shooting vector s (which referred to a solution of the local IVP's). Here we use additionally the linearization at functions that do not satisfy the ODE (5.1.1a) or (5.1.2a), for instance splines approximating a solution of (5.1.1). therefore we define the notation anew.

5.1.5 Definition

The derivative of $h(x,y)$ on a subinterval $[x_k,x_{k+1}]$ with respect to its second argument at a function $w_k(x)$, which is continuous on the subinterval, is denoted by

$$(5.1.5a) \qquad L_k(x;w_k(x)) := \frac{\partial}{\partial y} h(x,y)\Big|_{y=w_k(x)} \quad , \quad x_k < x < x_{k+1} \ .$$

The fundamental solution $Y_k(x;w_k(x))$ of the linearized system on $[x_k,x_{k+1}]$ is a solution of the ODE

$$(5.1.5b) \qquad \dot{z} = L_k(x;w_k(x))z \quad , \quad x_k < x < x_{k+1} \ ,$$

satisfying the boundary conditions

$$(5.1.5c) \qquad A_k Y_k(x_k;w_k(x)) + B_k Y_k(x_{k+1};w_k(x)) = I_n \ .$$

And the derivatives of the boundary conditions are

$$(5.1.5d) \qquad B_a(w(x)) := \frac{\partial g(u,w(b))}{\partial u}\Big|_{u=w(a)} \quad \text{and} \quad B_b(w(x)) := \frac{\partial g(w(a),v)}{\partial v}\Big|_{v=w(b)} \ .$$

N.B. We suppressed the functional dependency of Y_k with respect to A_k and B_k as this would be apparent from the context.

♦

The Jacobian $J(s;z)$ of $f(s;z)$ can be formulated in terms of $Y_k(x;z)$:

$$(5.1.6) \qquad \begin{pmatrix} Y_1(x_2;z) & -Y_2(x_2;z) & & & \\ & Y_2(x_3;z) & -Y_3(x_3;z) & & \\ & & \ddots & \ddots & \\ & & & Y_{N-1}(x_N;z) & -Y_N(x_N;z) \\ B_a(z)Y_1(x_1;z) & & & & B_b(z)Y_N(x_{N+1};z) \end{pmatrix} .$$

One can easily prove that Theorem 2.3.9, which gives an estimate for the Lipschitz constant of the Jacobian in terms of the conditioning constant of the local problems (5.1.2), also holds for this more general formulation of shooting. In particular for exponentially

dichotomic BVP a correct choice of the local BC will reduce the conditioning constant of (5.1.2) considerably, thus increasing the convergence area of Newton's method for f, according to the Newton-Kantorovich theorem. In [dHMa85] and [AsMa] this formulation of the shooting method was used to analyze the convergence and stability of finite difference methods in a linear context. Here we investigate the actual implementation of this generalisation of multiple shooting. First of all we see that if $A_k \neq 0$ and $B_k \neq 0$, a BVP with linear boundary conditions is defined on every subinterval. Now it may seem unwise to replace one problem by N problems of the same type with additional unknowns s_k, $1 \leq k \leq N$. However, there is some merit in this splitting, as we shall show now.

We just pointed out that the use of ordinary multiple shooting may be disadvantageous. Hence for solving the 'local' problems we only consider the use of global methods, i.e. finite differences or collocation. A divide and conquer method is the following *Unbiased Multiple Shooting* (UMS) algorithm.

5.1.7 UMS-Algorithm

- given an initial estimate for $y^*(x;s^*)$, compute the vector s^0 from (5.1.2b).

- while $| f(s^j; z^j(x;s^j)) |$ is not sufficiently small
  do begin

  (A1) on every subinterval compute by collocation or finite differences, a new approximation $z^j(x;s^j)$ to the solution of (5.1.2) for the new value s^j, with $z^{j-1}(x;s^{j-1})$ as initial guess and compute an approximation of the fundamental solution of the linearized BVP at $z^j(x;s^j)$.

  (A2) compute the residual vector $f(s^j; z^j(x;s^j))$ and perform a Newton iteration rendering s^{j+1}.

  end.

♦

The two steps (A1) and (A2) do not have an equal status. An important difference is that every update of s requires a new approximation to the solution of (5.1.1), i.e. at every iteration only one update on s is made. On the other hand in step (A1) the vector s is kept fixed and obtaining a new approximation $z^j(x;s^j)$ may require several Newton iterations or even choosing a new local collocation grid. In fact (A1) may stand for a call to a collocation algorithm and will generally contain what we henceforth shall call an 'inner' iteration loop (as opposed to the 'outer' iteration on s).

Notice that at step (A1) every subinterval can be treated completely separately. Thus a major part of the memory needed for the collocation or finite difference process at one step can be used again at the next one, as we only store information about $z(x;s)$ and not about the linearized system. In this way it may be possible to handle more difficult problems, that would otherwise require more memory storage.

On the other hand step (A1) lends itself to implementation on a parallel computer in a more or less straightforward way. For every vector s^j the local BVP's on the subintervals can be distributed over the available processors. This could be combined with a stable parallel algorithm to solve the linear equation $J\xi = -f$, for instance the one described in [AsPC] or [Wr].

One may also encounter a situation where the problem at hand has a few regions where the problem is essentially more difficult than elsewhere. This may be due to a locally poor initial guess or to local sensitivity of the BVP. When a collocation code is applied to the BVP on the entire interval, the internal Newton solver may require a considerable amount of iterations. If the interval is split into smooth regions and more difficult ones, application of the same code will generally require only a few iterations on the smooth regions; at the same time it is to be expected that solving the BVP on the difficult subintervals does not take more iterations than solving the BVP on the entire interval. However, these iterations for the former require less function calls and the solution of smaller linear systems. So the unbiased multiple shooting algorithm can reduce the computational costs of solving a BVP, provided that determining the 'shooting vectors' s_k is not too expensive. A nice class of such BVP's is given by singularly perturbed problems, where a *reduced solution* (i.e. the -outer- solution of the reduced problem) is easy to find. We shall demonstrate this by the following example.

5.1.8 Example

Consider the singularly perturbed BVP, cf. [O'Ma],

$$\text{(5.1.8a)} \qquad \varepsilon\ddot{y} = \dot{y} - \dot{y}^3 \quad , 0 < x < 1 \ ,$$

$$\text{(5.1.8b)} \qquad y(0) = 0 \ , \ y(1) = 0.5 \ .$$

The stable limiting solution for $\varepsilon \downarrow 0$,

$$\text{(5.1.8c)} \qquad y(x) = \begin{cases} 0 & , \text{if} \quad 0 \le x \le 0.5 \ , \\ x - 0.5 & , \text{if} \quad 0.5 \le x \le 1 \ , \end{cases}$$

has a discontinuity in its first derivative at $x = 0.5$. We solve this BVP by the collocation code COLNEW cf. [BaAs] on the entire interval and by the UMS algorithm (for details on the implementation and further comments see §3) up to a tolerance 10^{-6}. The reduced so-

lution (5.1.8c) is used as initial guess and the local boundary conditions were the analogue of (5.1.8b), i.e.

$$A_k = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}, \quad B_k = \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix}.$$

The first order approximation of the solution is used to choose two grid points x_2 and x_3 on either sides of the 'crack' $x = 0.5$ such that the estimated error in s_k is less than the required tolerance 10^{-6}. In the UMS algorithm with $\varepsilon \in [10^{-2}, 10^{-4}]$ no iterations on s are needed as the norm of the first update on s is already less than 10^{-8}. The results are listed in the Table 5.1 and 5.2 . The column 'memory use' states the number of double precision places (in standard IBM Fortran) required for the collocation algorithm; the additional memory used for integers is negligible. The UMS algorithm saves about 35% to 75% on both function evaluations and memory use.

UMS on (5.1.8a,b) with tol = 10^{-6}

ε	x_2	x_3	memory use	#f-calls			
				1st	2nd	3rd	interval
10^{-2}	0.380	0.620	9728	18	2968	30	
10^{-3}	0.485	0.515	14592	18	3366	22	
10^{-4}	0.499	0.501	14592	18	3414	22	

Table 5.1

COLNEW on (5.1.8a,b) with tol = 10^{-6}

ε	memory use	#f-calls
10^{-2}	14592	4578
10^{-3}	58368	13140
10^{-4}	58368	13500

Table 5.2

♦

An important issue in our algorithm is the choice of the local boundary conditions (BC's) in such a way that the local BVP's are well-conditioned. Occasionally local BC's are provided naturally by the problem (e.g. (5.1.8a,b)). If we do not have sufficient understanding of the structure of the solution space of the BVP, then it advisable to choose non-separated BC, because separated BC increase the risk of controlling a mode on the wrong side of the interval. The choice

(5.1.9) $A_k = B_k = I_n$

often proves to be an acceptable one. Indeed, if the solution modes show only small changes in direction, we do not expect any cancelation effects to occur in $A_k Y_k(x_k) + B_k Y_k(x_{k+1})$. However, the complexity of most algorithms for collocation and finite differences reduces considerably if the boundary conditions are separated. This can of course be done by adding as many auxiliary variables as there are coupled BC, see [AsRu]. An alternative is to compute separated boundary conditions, according to a method formulated in [dHMa87], which we describe below.

Suppose we have an initial guess $z^0(x;s^0)$ of the solution of (5.1.2),(5.1.9) and we are able to compute the fundamental solution $Y_k(x)$ of the linearized problem at $z^0(x;s^0)$ (i.e. the BVP is not too ill-posed). Now let $U_k \Sigma_k V_k^T$ be the singular value decomposition of $Y_k(x_{k+1}) Y_k^{-1}(x_k)$. Note that this operation is not excessively expensive, since $Y_k(x_{k+1})$ and $Y_k(x_k)$ have to be computed anyway for the Newton iteration on s and a SVD requires only $O(n^3)$ operations; this is essentially smaller than $O(N_k n^3)$, with N_k the number of collocation grid points, required for solving the linear equations to update $z(x;s)$ in the collocation process.

Now let p, $1 \le p \le n$, be such that the singular values $\sigma_i \ge 1$, for $1 \le i \le p$, and $\sigma_i < 1$, for $p+1 \le i \le n$. According to [dHMa87]§3 the boundary conditions

(5.1.10) $$\tilde{A}_k = \begin{pmatrix} 0 & 0 \\ 0 & I_{n-p} \end{pmatrix} V_k^T \qquad \text{and} \qquad \tilde{B}_k = \begin{pmatrix} I_p & 0 \\ 0 & 0 \end{pmatrix} U_k^T$$

induce a BVP (5.1.2) on $[x_k, x_{k+1}]$ with conditioning constant $\alpha + 4\alpha^2$, where α is the conditioning constant of (5.1.2),(5.1.9). This can be unsatisfactory if α is large. However, the conditioning constant of (5.1.2),(5.1.10) can be related to that of the global BVP (5.1.1) at $z^0(x;s^0)$ as is shown by the following lemma.

5.1.11 Lemma

The conditioning constant of (5.1.2),(5.1.10) *does not exceed* $\kappa + 4\kappa^2$, *with* κ *the conditioning constant of the linearization of* (5.1.1) *at* $z^0(x;s^0)$.

Proof

According to lemma 1.2.8 there exists BC on every subinterval of $[a,b]$ such that the conditioning constant of the linearized BVP does not exceed κ. The proof now readily follows from the fact that the product $Y_k(x_{k+1}) Y_k^{-1}(x_k)$ is independent of the local BC.

♦

In practice one has to be careful when implementing boundary conditions such as (5.1.10). From a computational point of view it is preferable to compute the SVD from one of the first approximations $z^0(x;s^0)$ or $z^1(x;s^1)$. However, if they differ greatly from the solution

$y^*(x)$, both the directions and the singular values may be so inaccurate that the conditioning of (5.1.1) is not very good. This is demonstrated by the next example.

5.1.12 Example

Consider the boundary value problem, proposed e.g. in [Ho],

$$
\begin{aligned}
\dot{y}_1 &= y_2 \\
\dot{y}_2 &= y_3 \\
\dot{y}_3 &= -\frac{(3-n)}{2} y_1 y_3 - n y_2^2 + 1 - y_4^2 + s y_2 \qquad , 0 < x < \infty \, , \\
\dot{y}_4 &= y_5 \\
\dot{y}_5 &= -\frac{(3-n)}{2} y_1 y_5 - (n-1) y_2 y_4 + s(y_4 - 1) \, ,
\end{aligned}
\tag{5.1.12a}
$$

with boundary conditions

$$
y_1(0) = y_2(0) = y_4(0) = 0 \, , \quad y_2(\infty) = 0 \, , \quad y_4(\infty) = 1 \, . \tag{5.1.12b}
$$

We apply the UMS algorithm to this BVP for the parameters $n = s = 0.2$ and $L = 60$, with subintervals [0,5], [5,10], [10,30] and [30,60]. First we choose the boundary conditions

$$A_k = B_k = I_5$$

This requires 6 'outer' iterations on s to obtain a precision of 10^{-6}. In the first and fifth iteration the singular value decomposition of $Y_k(x_{k+1}) Y_k^{-1}(x_k)$ is computed. In both cases we find that 3 singular values are larger than 1, implying that the problem has three nondecreasing modes. This seems to be contradicted by the global BC's, that have 2 end point conditions, indicating 2 nondecreasing modes. However, only 2 singular values are considerably larger than 1, ranging from $\sim 10^2$ at [0,5] to $\sim 10^6$ at the last subinterval, and the third singular value is only just larger than 1.

This yields four different sets of separated BC for the local BVP's ; using the SVD results from either the first or fifth iteration and with either 2 or 3 initial conditions.

Let $B_k(i,j)$ denote the endpoint conditions on the k^{th} subinterval resulting from the fundamental solutions obtained in the i^{th} iteration with j initial conditions. Using three initial conditions, convergence is obtained with the local BC resulting from the first iteration, but the computational costs are, as expected, higher than for the local BC resulting from the fifth iteration. The failure on the last interval [30,60] with local BC resulting from the first iteration with 2 initial conditions can be viewed as a standard example of ill-conditioning due to boundary conditions. Namely, the BC resulting from the 5^{th} iteration seem to give rather well-conditioned local BVP's, but the angle between the 3-dimensional subspaces of range($B_4(1,2)$) and range($B_4(5,2)$) is almost 90°, i.e. there is a solution mode which is controlled by $B_4(5,2)$, but not by $B_4(1,2)$.

In a parallel implementation the computing time for the UMS algorithm would be less than for the globally used collocation algorithm, when using the coupled BC $A_k = B_k = I_5$ or the BC resulting from the fifth iteration. However, the reduction of computational time would have been considerably larger if less iterations on the shooting vectors had been necessary, as we saw for the singularly perturbed BVP in example 5.1.8.

The memory requirement for the UMS-algorithm for separated local BC is almost 45% less than for collocation on the entire interval. When using the non-separated BC $A_k = B_k = I_5$, the memory use is considerably larger than for the other, separated BC. This is due to the fact that the collocation solver used needs separated BC; hence we have to add 5 trivial differential equations to create the separation artificially (see [AsRu]). Application of COLNEW to each of the subintervals does not require more grid points for the coupled BC than for the other local BC.

UMS applied to (5.1.12a+b), tol = 10^{-6}

		#f-calls				
	iter	[0,5]	[5,10]	[10,30]	[30,60]	memory use
$A_k = B_k = I$	7	2832	1192	2688	3072	88101
1st iter., 3 init. cond.	8	2960	1408	3264	4256	25961
5th iter., 3 init. cond.	6	1952	928	2224	2560	25961
1st iter., 2 init. cond.	4	fail of COLNEW on [30,60]				
5th iter., 2 init. cond.	7	2378	1408	2336	2816	25641

COLNEW applied to (5.1.11a+b), tol = 10^{-6}

#f-calls	memory use
4080	46336

Table 5.3

♦

§5.2 Convergence

In the *Unbiased Multiple Shooting* algorithm as sketched in the previous section two iterative processes are interacting, viz. a process on s^j and another to obtain $z^j(x;s^j)$; however, the algorithm is not symmetrical with respect to both processes.
One can implement the algorithm 5.1.7 in various ways. For the 'outer' iteration on s^j we can use well known adaptations of Newton's method such as damping and keeping the Jacobian fixed (cf. time stepping algorithm Ch.3). Additionally there are several ways to perform the 'inner' step (A1). One can call a collocation or finite difference routine to obtain y^j and estimates $\hat{Y}_k^j(x)$ for the fundamental solutions $Y_k(x;z_k^j(x;s_k^j))$ (that are required for the Jacobian $J(s^j;z^j(x;s^j))$ with certain prescribed tolerances. These tolerances need not be kept constant during the entire process. At the first few steps, when s^j is far away from the solution s^*, it is not necessary to approximate $y_k(x;s_k^j)$ very well. But, as we show, eventual quadratic convergence requires the tolerance for $z^j(x;s^j)$ to decrease like $|s^j - s^*|^2$ and the tolerance for $\hat{Y}_k^j(x)$ like $|s^j - s^*|$ at the last few steps of the algorithm.

Another way to implement step (A1) hinges even stronger on the thought that it does not pay to compute $z^j(x;s^j)$ very accurately if s^j is still far from s^*. We can choose a fixed collocation grid on every subinterval and at the j^{th} 'outer' iteration step we perform only a few Newton iterations on the collocation scheme to obtain y^j (i.e. without an accuracy requirement). This way the iteration on the vector s^j plays a more dominant role than in the implementation suggested before. Once convergence of s and $z^j(x;s^j)$ on a grid has been established, the discretization error is estimated and the grid adjusted and refined accordingly.

A strategy for the error tolerances for $z^j(x;s^j)$ and $\hat{Y}_k^j(x)$ can be derived from a study on the effects of these errors on the convergence of the 'outer' loop, i.e. Newton's iteration. The Newton update on the vector s^j will not be computed using the real Jacobian $J(s^j;y(x;s^j))$, but only through an approximation J^j. Define the errors

$$\text{(5.2.1a)} \qquad \varepsilon_{y,k}^j := \max_{x\in[x_k,x_{k+1}]} |z_k^j(x;s_k^j) - y_k(x;s_k^j)| \quad , \qquad \varepsilon_y^j := \max_k \varepsilon_{y,k}^j \, ,$$

$$\text{(5.2.1b)} \qquad \varepsilon_{Y,k}^j := \max_{x\in[x_k,x_{k+1}]} |\hat{Y}_k^j(x) - Y_k^j(x;z_k^j(x;s_k^j))| \quad \textit{and} \quad \varepsilon_Y^j := \max_k \varepsilon_{Y,k}^j \, .$$

5.2.2 Assumption

There is a constant $\Delta > 0$, *such that the neighbourhood*

$$D_y := \{ z : [a,b] \to \mathbb{R}^n \mid z \text{ continuous on } (x_k, x_{k+1}) , k \in \{1,..,N\}, \text{ and } \|z - y^*\| \le \Delta \}$$

of $y^*(x)$ *satisfies the following conditions*

(*i*) y^* *is the only solution of* (5.1.1) *in* D_y ,

(*ii*) *the upper bound* C_{gh} *on the first and second derivatives of* $h(x,y)$ *with respect to* z *and on the first and second (partial) derivatives of* $g(u,v)$ *is of moderate size,* $z \in D_y$.

♦

5.2.3 Definition

(*i*) *The constant* κ_k *is an upper bound on the conditioning constant of the linearization of* (5.1.2) *at* $z(x)$ *on* $[x_k, x_{k+1}]$ *for all* $z \in D_y$.

(*ii*) *The set* D_s *is defined by* $D_s := \{ s \in \mathbb{R}^{nN} \mid y(x;s) \in D_y \}$.

♦

To investigate the difference between the Jacobian $J(s^j; y(x;s^j))$ and its approximation J^j, we estimate the difference between $Y_k(x; y_k(x;s_k^j))$ and its computed approximation $\hat{Y}_k^j(x)$, neglecting rounding errors as they are negligible compared to the approximation errors.

5.2.4 Lemma

Let $s^j \in D_s$ *and* $z^j(x;s^j) \in D_y$. *Then*

$$(5.2.4a) \quad \forall_{1 \le k \le N} \ \forall_{x \in [x_k, x_{k+1}]} \ : \ | \hat{Y}_k^j(x) - Y_k(x; y_k(x;s_k^j)) | \le C_{gh} \kappa_k^2 (x_{k+1} - x_k) \varepsilon_{y,k}^j + \varepsilon_{Y,k}^j .$$

If the local BVP's (5.1.2) *are well conditioned there is a constant* C *(depending on* κ_k *and* C_{gh} *) of moderate size such that*

$$(5.2.4b) \qquad | J(s^j; y(x;s^j)) - J^j |_\infty \le C(\varepsilon_Y^j + \varepsilon_y^j) .$$

Proof

The matrix $\hat{Y}_k^j(x)$ contains errors due to

(*i*) the error in $z^j(x;s^j)$

(*ii*) discretization errors in integrating the linearized problem

The effect on the fundamental solution of the difference between $y(x;s^j)$ and $z^j(x;s^j)$ can be estimated similarly as in the proof of theorem 2.3.9. Since $Y_k(x; z_k^j(x;s_k^j))$ and $Y_k(x; y_k(x;s_k^j))$ satisfy the same BC, their difference can be written as

$$Y_k(x;z_k^j(x;s_k^j)) - Y_k(x;y_k(x;s_k^j))$$
$$= \int_{x_k}^{x_{k+1}} G(x,t;y_k(x;s_k^j))\,(L(t;z_k^j(x;s_k^j)) - L(t;y_k(x;s_k^j))\,Y_k(t;z_k^j(x;s_k^j))\,dt\,.$$

Together with $|\,L(x;z_k^j(x;s_k^j)) - L(x;y_k(x;s_k^j))\,| \le C_{gh}\,|\,z_k^j(x;s_k^j) - y_k(x;s_k^j)\,|$ this yields

$$\max_x |\,Y_k(x;z_k^j(x;s_k^j)) - Y_k(x;y_k(x;s_k^j))\,| \le C_{gh}\kappa_k^2\,(x_{k+1}-x_k)\max_x |\,z_k^j(x;s_k^j) - y_k(x;s^j)\,|\,.$$

Now (5.2.4a) follows from the fact that the discretization error mentioned under (*ii*) is controlled by a parameter $\varepsilon^j{}_{Y,k}$.

The last block row of $\boldsymbol{J}(s^j;y(x;s^j))$ contains, besides the fundamental solutions $Y_k(x;y_k(x;s_k^j))$, derivatives of the boundary conditions $g(u,v)$ and one can derive that

$$|\,B_a(z_1^j(x_1;s_1^j))\,\hat{Y}_1^j(x_1) - B_a(y_1(x_1;s_1^j))\,Y_1(x_1;s_1^j)\,| \le C_{gh}^2\kappa_1^2(x_2-x_1)\,\varepsilon_{y,1}^j + C_{gh}\varepsilon_{Y,1}^j$$
$$+ C_{gh}(\varepsilon_{y,1}^j + \varepsilon_{y,N}^j)\kappa_1\,.$$

Now (5.2.4b) follows immediately from this relation and (5.2.3a).

♦

A smaller value of κ_k reduces the influence of the discretization error $\varepsilon_{y,k}^j$ on the error in the approximation of $\boldsymbol{J}(s^j;y(x;s^j))$, as well as it may enlarge the convergence area of Newton's method applied to $\boldsymbol{f}$. From estimate (5.2.4a) we see that the choice of $\varepsilon_{y,k}^j$ per subinterval can be used to equidistribute the errors in the Jacobian. This can be useful if the bound C_{gh} is known to vary over the subintervals.

The iterates s^j result from Newton's method; hence we have locally quadratic convergence, if the Jacobian matrices are determined with sufficient accuracy. In our algorithm the accuracy depends on $\varepsilon_{y,k}^j$ and $\varepsilon^j{}_{Y,k}$.

5.2.5 Lemma

Let $s^j \in D_s$ and $z^j(x;s^j) \in D_y$. Assume there is a moderate bound, say γ, on $|\,\boldsymbol{J}(s)\,|$, $|\,\boldsymbol{J}^{-1}(s)\,|$ and the Lipschitz constant of $\boldsymbol{J}(s)$ on D_s. Let C be the constant of Lemma 5.2.4, *then the following estimate holds*

(5.2.5a)

$$|\,s^{j+1} - s^*\,| \le \gamma^2\,|\,s^j - s^*\,|^2 + C\gamma^3(\varepsilon_Y^j + \varepsilon_y^j)\,|\,s^j - s^*\,| + 2C_{gh}\gamma\,\varepsilon_y^j + O(\varepsilon_y^j\varepsilon_Y^j + (\varepsilon_Y^j)^2)\,.$$

Proof

Application of Newton's algorithm with the approximate Jacobian J^j gives

$$\begin{aligned} s^{j+1} - s^* &= s^j - s^* - (J^j)^{-1} f(s^j; z^j(x; s^j)) \\ &= s^j - s^* - J^{-1}(y(x;s^j)) f(s^j; y(x;s^j)) + J^{-1}(y(x;s^j)) f(s^j; y(x;s^j)) - (J^j)^{-1} f(s^j; z^j(x;s^j)) . \end{aligned}$$

Hence

$$\begin{aligned} |s^{j+1} - s^*| &\le \gamma^2 |s^j - s^*|^2 + |J^{-1}(s^j; y(x;s^j)) - (J^j)^{-1}| . |f(s^j; y(x;s^j))| \\ &\qquad + |(J^j)^{-1}| . |f(s^j; y(x;s^j)) - f(s^j; z^j(x;s^j))| \\ &\le \gamma^2 |s^j - s^*|^2 \\ &\quad + |(J^j)^{-1}(J^j - J(s^j; y(x;s^j)))J^{-1}(s^j; y(x;s^j))| . |f(s^j; y(x;s^j)) - f(s^*; y(x;s^*))| \\ &\quad + (\gamma + \gamma^2 \varepsilon_Y^j + O((\varepsilon_Y^j)^2) 2 C_{gh} \varepsilon_y^j \\ &\le \gamma^2 |s^j - s^*|^2 + (\gamma + \gamma^2 \varepsilon_Y^j + O((\varepsilon_Y^j)^2)[C(\varepsilon_Y^j + \varepsilon_y^j)\gamma^2 |s^j - s^*| + 2\gamma C_{gh} \varepsilon_y^j] \\ &\le \gamma^2 |s^j - s^*|^2 + C\gamma^3 (\varepsilon_Y^j + \varepsilon_y^j) |s^j - s^*| + 2\gamma C_{gh} \varepsilon_y^j + O(\varepsilon_y^j \varepsilon_Y^j + (\varepsilon_Y^j)^2) . \end{aligned}$$

♦

The previous lemmas prove that an implementation of the UMS algorithm where $z^j(x;s^j)$ and $\hat{Y}_k^j(x)$ are computed within given accuracy at every step, is locally convergent. Moreover, this convergence is quadratic if eventually the tolerances are decreased such that

$$\varepsilon_Y^j \approx |s^j - s^*| \quad \text{and} \quad \varepsilon_y^j \approx |s^j - s^*|^2 .$$

The other implementation where the collocation grid is kept fixed and only a few Newton updates for $z^j(x;s^j)$ are computed before computing a new s^j, can give at most a linear convergence rate. Since $\hat{Y}_k^j(x)$ is the solution of a linear BVP the error $\varepsilon^j{}_{Y,k}$ is fully determined by the discretization error, i.e. by the grid choice; hence the factor $\varepsilon_Y^j |s^j - s^*|$ is a linear term in the error estimate (5.2.5a).

§5.3 Numerical results

In the previous sections we looked at theoretical aspects of the UMS algorithm. Next we want to investigate its performance in practice. A code has been written for first order ODE (a higher order can be reformulated into first order, see [AsRu]) and using the existing collocation code COLNEW, cf.[AsChRu,BaAs], to solve the local BVP's on the subintervals. Since COLNEW can deal with separated BC only, the use of coupled BC increases the memory use substantially; for dummy variables have to be added to artificially separate the BC (see example 5.1.12). Although our numerical results show some effects to be attributed to peculiarities of COLNEW, rather than UMS, the overall results indicate a satisfactory agreement with the analysis. Yet, to understand the actual numbers more in detail we shall describe our implementation below.

Our UMS implementation has two precision parameters EPSS and TOLF^j. Convergence of the algorithm is established if the norm of the update δs on the shooting vector s is less than EPSS. The parameter TOLF^j is the error tolerance for the solution of the local BVP's obtained by COLNEW, hence TOLF^j is equivalent to ε_y^j in §2. Accordingly, TOLF^j is rather large at first $(10^{-2},10^{-3})$ and is decreased thereafter. We found that the requirement that $\mathrm{TOLF}^j \approx |\, \delta s^j \,|^2$ is not always sufficient. Sometimes it occurred that $|\, f(s^j) \,| \le \mathrm{TOLF}^j$. Since most components of $f(s^j)$ are a difference between two values of $z^j(x;s^j)$, these components may have no significant number at all, due to cancelation. Hence the computed direction of δs may be inaccurate. In order to prevent this, we want TOLF^j to be less than or equal to a tenth of the expected value of $|\, f(s^{j+1}) \,|$ (i.e. $\approx |\, f(s^j) \,|^2$). This finally leads to the following algorithm to determine TOLF

$$\mathrm{TOLF}^{j+1} = \max\left(\frac{\mathrm{EPSS}}{10}, \min\left(\mathrm{TOLF}^j, \frac{|f(s^j)|^2}{10}, \frac{|\delta s^j|^2}{10}\right)\right) . \tag{5.3.1}$$

Singularly perturbed BVP's, where the position and (approximate) width of the layers can be obtained analytically, are very well suited to show how well the UMS algorithm performs. Often one can use the reduced solution (i.e. for $\varepsilon = 0$) to obtain a very good initial guess for the solution at small ε values. It is advisable to choose the subintervals such that each one of them contains either an entire layer or a smooth region. Then the shooting vector s, obtained from the reduced solution, is quite accurate and none or very few 'outer' iterations are necessary.

When transforming a higher order singularly perturbed BVP into a first order BVP, we have to pay special attention to scaling. In some cases (e.g. example 5.3.3) the first derivative reaches values of the order of ε^{-1}. Hence we scale the derivative term as in

(5.3.2) $$\begin{cases} u(x) = y(x) , \\ v(x) = \varepsilon \dot{y}(x) . \end{cases}$$

The results of applying COLNEW to singularly perturbed problems, as shown in the examples, indicate that the memory use at for $\varepsilon = 10^{-k}$, $k \in \{2,3,4,5\}$, is generally a multiple of that for the previous ε. This is due to the fact that we do not allow COLNEW to use its grid generator, but just let it halve the grid successively until the required tolerances have been obtained. In [AsChRu] this strategy is suggested for this type of problems, because the grid generator fails to 'see' the layers at first and produces a grid on which no convergence can be obtained, leading to failure of the code. We tested several initial grids for the afore-mentioned ε-values and tabulated some results. An unintentional advantage of the UMS algorithm is that the grid generator of COLNEW worked properly on subintervals that contained a layer and a small part of a smooth region only.

5.3.3 Example

Consider the singularly perturbed BVP, cf.[AsMaRu]

(5.5.3a) $\varepsilon \ddot{y} = y(1 - \dot{y})$, $0 < x < 1$,

(5.5.3b) $y(0) = 0.5$,

(5.5.3c) $y(1) = 2$.

The stable solution of the reduced problem is $y(x) = x+1$. Since it satisfies the end point condition, there will be a boundary layer at $x = 0$. Transformation (5.3.2) is used to convert the problem into a first order system. As we anticipate the correctness of s, we set TOLF_u, the required tolerance for the first variable u, to 10^{-6}. The tolerance for the second variable needs special attention. Because $v \approx \varepsilon$ on a mayor part of the interval and COLNEW uses the mixed convergence criterion

$$\| \text{ absolute error in } v \| \leq \mathrm{TOLF}_v \, (\, 1 + \| v \| \,) ,$$

the variable v has only $-\log(\varepsilon^{-1} \, \mathrm{TOLF}_v)$ correct digits. Indeed experiments with $\varepsilon = 10^{-5}$ and $\mathrm{TOLF}_u = \mathrm{TOLF}_v = 10^{-6}$ yielded a highly oscillatory 'solution'. To ensure that v has at least 3 correct digits we imposed the tolerance $\mathrm{TOLF}_v = 10^{-3} \times \varepsilon$.

The results tabulated in Table 5.4 show that the UMS algorithm saves both memory and function calls as compared to COLNEW. Note, however, that the typical doubling of grid points is a COLNEW feature and is open to improvement. The subinterval choice is clearly not optimal in balancing the work load for different processors. However, splitting the

layer region into several subintervals does not pay; because then the correct value of all shooting vectors is not known in advance and several iterations on s are needed.

	UMS, epss = 10^{-6}				COLNEW, tol = 10^{-6}	
ε	x_2	memory use	#f-calls		memory use	#f-calls
			$[x_1,x_2]$	$[x_2,x_3]$		
1e-2	15e-2	5624	1538	18	7296	1932
1e-3	15e-3	6080	1610	18	7296	2484
1e-4	15e-4	12160	2890	18	29184	6204
1e-5	15e-5	24320	5510	18	58368	12540

Table 5.4

§5.4 Parallel computation

The idea for unbiased multiple shooting was derived from the fact that any proper code for solving BVP's should treat growing and decaying modes correctly, unlike the original multiple shooting algorithm. The preconditioned time stepping algorithm described in chapters 3 and 4 can not be implemented an a parallel computer straightforwardly. But, as we indicated in the previous sections the UMS algorithm, like the original shooting, lends itself for parallel implementation.

In this section we briefly consider some more parallelization aspects, in particular estimates of the computational costs. A straightforward parallel implementation of step (A1) of the UMS algorithm (5.1.7) consists of assigning every local BVP to a different processor. We will assume that a sufficient number of processors is available. Hence the k-th processor has to apply COLNEW to the non-linear BVP on $[x_k, x_{k+1}]$. This is an iterative process, say with m_k iterations and suppose the local grid consists of N_k points. Then the costs for one iteration can be approximated by

forming the linearized system :	$O(n^2 N_k)$
solving the linear system $J\xi=-f$ (by SOLVEBLOCK):	$O(n^3 N_k)$
computing the new solution $z_k(x;s_k^j)$:	$O(nN_k)$
	$O(n^3 N_k)$

Additionally choosing new grids and estimating the discretization error involve computations, but they require only $O(nN_k)$ operations, which is negligible compared to the costs of one iteration. If N_k is taken as the size of the largest grid used on $[x_k, x_{k+1}]$, then the total costs of step (A1) on N parallel processors is

$$O(n^3 \max(N_k m_k)).$$

An optimal choice for the subintervals, would be one that equidistributes $m_k N_k$ as much as possible. Hence the coarse grid should be finer in areas where the solution changes rapidly.

Step (A2) of the UMS-algorithm essentially consists of solving a linear system. In the literature several methods are mentioned to perform this in parallel. In [PaGl] a parallel algorithm is presented especially for the systems arising in collocation and finite differences. There the matrix is partitioned into smaller pieces of the same structure, thus performing implicitly an idea similar to unbiased shooting. (However, the choice of local boundary conditions is not addressed, nor is it clear that the algorithm renders well-conditioned local

BVP's.) Under the assumption that there are more processors than local subintervals, the computational costs of the algorithm are $O(n^3N)$.

In [AsPC] several parallel solution methods for the linear system are considered. As a stable method, a least squares formulation is mentioned. A stable odd/even reduction and elimination is used to solve $J^TJ\xi = -J^Tf$ in $O(n^3 \log N)$ time, if $O(N)$ processors are available.
Finally, we mention the structured QR-decomposition described in [Wr]. This method, which is stable for well-conditioned BVP's, partitions the system into blocks and performs a special QR-decomposition on every block (in parallel). If there are approximately $\frac{1}{2}N$ processors available (i.e. even less than we assumed for step (A1)), the algorithm takes $O(n^3 \log N)$ operations.

Let σ denote the number of outer iterations on the vector s in Algorithm 5.1.7. Then parallel computation requires $O(n^3\ \sigma\ (\log N + \max m_k N_k))$ flops. Hence we see that the costs will be minimal if N is taken rather large, with only a few 'fine' grid points per subinterval.

However, we found that this strategy is not an optimal choice for some singularly perturbed problems. Indeed, if the reduced solution and the position of the layers is known, it is favourable to choose the subintervals such that they either contain an entire layer or (a part of) a smooth region. In this case the initial guess for the vector s^0, based on the reduced solution, is already quite accurate, reducing the amount of outer iterations to 1 or 2. Of course the work load of (A1) is poorly distributed over the processors (since the layers require essentially more effort), but this is more than compensated for by the reduction of the outer iterations. Hence it is not useful to use more processors than the number of layers plus 1.

Conclusion

In chapter 2 we described the multiple shooting method for non-linear BVP's. We found that the set of equations $f(s)$, and also the corresponding Jacobian, could be very sensitive for changes in the starting vector s in some directions. In fact the Lipschitz constant of the Jacobian is bounded in terms of the conditioning constant of the local problems. Due to this sensitivity it may be difficult to solve the non-linear equation $f(s)$ with Newton's method.

Based on these considerations we investigated the consequences of defining well-conditioned BVP's on the subintervals. The resulting UMS-algorithm contains two types of ite-

rations. First there is the iterative process on the 'shooting' vector s, analogously to the ordinary multiple shooting algorithm. And since every update on s requires the computation of new solutions for the local BVP's, there is a second iterative process, solving the local non-linear BVP's by collocation or finite differences. Note that in the UMS-algorithm both the 'global' equation $f(s)$ and the local BVP's are well-conditioned problems.

The UMS-algorithm is not only a generalization of multiple shooting, but also of collocation and finite differences. This generalization gives the possibility to make a stable parallel algorithm for solving non-linear BVP's.

In a sequential implementation the algorithm combines the potentially stable features of global solution methods with the more modest memory use of multiple shooting.

Appendix A

Consider the linear BVP

$$\text{(A.1)} \qquad \begin{cases} \dot{y} = A(x)y + q(x) & , \quad a < x < b \ , \\ B_a y(a) + B_b y(b) = \beta \ . \end{cases}$$

Let $Y(x)$ be the fundamental solution that satisfies the dichotomy relations (1.2.1) with projection $P = \begin{pmatrix} 0 & 0 \\ 0 & I_p \end{pmatrix}$ and constants (K,λ,μ). Let $Z(x)$ be another fundamental solution and let $H \in \mathbb{R}^{n\times n}$ be the invertible matrix such that $Z(x) = Y(x)H$. If H is partitioned in the same way as the projection P, then the consistency constant L of Z is defined as

$$\text{(A.2)} \qquad L := \frac{|Y^2(a)H^{21}|}{\mathrm{glb}(Y^1(a)H^{11})} \ .$$

Now L is zero if H^{21} is zero. In [AsMaRu] a bound of the form

$$\max_{d\neq 0} \frac{|Z^1(x)d|}{|Z^1(t)d|} \leq K(1+LK^2)\sqrt{K^2+1}\ .\exp(\mu(x-t)) \qquad , x < t \ ,$$

was derived from the dichotomy of $Y(x)$. However, the limit of this upper bound for L approaching 0, is larger than one may expect from the dichotomy of $Y(x)$. By carefully studying the proof a refinement of this bound can be derived.

Lemma A.3 (Improvement of [AsMaRu] 6.14)
Consider the dichotomic ODE

$$\text{(A.3a)} \qquad \dot{y} = A(x)y \qquad , a < x < b \ ,$$

Let $Y(x)$ and $Z(x)$ be fundamental solutions; $Y(x)$ with dichotomy projection $P = \begin{pmatrix} 0 & 0 \\ 0 & I_p \end{pmatrix}$ and $Z(x) = Y(x)H$. Let ϑ denote the minimum over $x\in[a,b]$ of the angle between range$(Y^1(x))$ *and* range$(Y^2(x))$ *(note that* $\sin(\vartheta) \geq 1/\sqrt{K^2+1}$ *). Then*

$$\text{(A.3b)} \qquad \max_{d\neq 0} \frac{|Z^1(x)d|}{|Z^1(t)d|} \leq \tilde{K}.\exp(\mu(x-t)) \quad , \quad x < t \ ,$$

$$\text{with} \quad \tilde{K} = K\frac{1+LK^2}{\sqrt{\sin^2\vartheta + [\max(0,\cos\vartheta - LK^2)]^2}} \ .$$

Note that if $L = 0$ then $\tilde{K} = K$.

Proof

If $H^{21} = 0$, then $L = 0$ and the relation follows immediately from the dichotomy of $Y(x)$ and $Z^1(x)d = Y^1(x)H^{11}d$. So assume that $H^{21} \neq 0$. Since $Z^1(x)d = Y^1(x)H^{11}d + Y^2(x)H^{21}d$,

$$|Z^1(x)d| \leq |Y^1(x)H^{11}d|\,(1+\frac{|Y^2(x)H^{21}d|}{|Y^1(x)H^{11}d|}) .$$

The dichotomy of $Y(x)$ now yields

$$\frac{|Y^2(x)H^{21}d|}{|Y^1(x)H^{11}d|} = \frac{|Y^2(x)H^{21}d|}{|Y^2(a)H^{21}d|}\frac{|Y^1(a)H^{11}d|}{|Y^1(x)H^{11}d|}\frac{|Y^2(a)H^{21}d|}{|Y^1(a)H^{11}d|} \leq LK^2 e^{(\lambda+\mu)(a-x)} .$$

Hence an upper bound on the numerator is $|Z^1(x)d| \leq (1+LK^2)\,|Y^1(x)H^{11}d|$.

So far we have followed the proof given in [AsMaRu]. However, the derivation of a lower bound on the denominator is different. We can write

$$Z^1(t)d = |Y^1(t)H^{11}d|\,(a+b) ,$$

with the vectors a and b defined by $a = \dfrac{Y^1(t)H^{11}d}{|Y^1(t)H^{11}d|}$ and $b = \dfrac{Y^2(t)H^{21}d}{|Y^1(t)H^{11}d|}$. Let $\theta(t)$ denote the angle between range$(Y^1(t))$ and range $(Y^2(t))$. Then $\theta(t) \in [\vartheta, \frac{\pi}{2}]$ and

$$\begin{aligned} |a+b|^2 &\geq |a|^2 + |b|^2 - 2|a||b|\cos\theta(t) = 1 + |b|^2 - 2|b|\cos\theta(t) \\ &\geq 1 + |b|^2 - 2|b|\cos\vartheta = \sin^2\vartheta + (\cos\vartheta - |b|)^2 . \end{aligned}$$

The smallest value will be obtained if $|b| = \cos\theta(t)$. However, the norm of b may not be that large, indeed

$$\begin{aligned} &|Y^2(t)H^{21}d| \leq K\exp(-\lambda(t-a))\,|Y^2(a)H^{21}d| \\ \text{and}\quad &|Y^1(t)H^{11}d| \geq K^{-1}\exp(\mu(t-a))\,|Y^1(a)H^{11}d| , \end{aligned}$$

we see that

$$|b| \leq K^2 L\exp(-(\lambda+\mu)(t-a)) \leq K^2 L .$$

Hence

$$\begin{aligned} |a+b|^2 &\geq \min\{\ \sin^2\vartheta + (\cos\vartheta - |b|)^2 \ \Big|\ 0 \leq |b| \leq KL^2\ \} \\ &= \sin\vartheta + [\max(0, \cos\vartheta - LK^2)]^2 . \end{aligned}$$

From which we can now derive that

$$\max_{d\neq 0}\frac{|Z^1(x)d|}{|Z^1(t)d|} \leq \frac{1+LK^2}{\sqrt{\sin^2\vartheta + [\max(0,\cos\vartheta - LK^2)]^2}} \cdot \max_{c\neq 0}\frac{|Y^1(x)c|}{|Y^1(t)c|} .$$

♦

Appendix B

Consider the BVP

$$\text{(B.1)} \qquad \begin{cases} \dot{y} = A(x)y + q(x) & , \quad a < x < b \ , \\ B_a y(a) + B_b y(b) = \beta \end{cases}$$

and assume that it has a unique solution for every $\beta \in \mathbb{R}^n$ and $q \in C([a,b] \to \mathbb{R}^n)$. Let $G(x,t)$ denote the Green's function of (B.1) and let $\Phi(x)$ be its fundamental solution with

$$\text{(B.2)} \qquad B_a\Phi(a) + B_b\Phi(b) = I_n \ .$$

Both in multiple shooting and global solution methods we encounter two types of matrices of almost the same form, viz.

$$\text{(B.3a)} \qquad L := \begin{pmatrix} \Phi_1(x_2) & -\Phi_2(x_2) & & & & \\ & \Phi_2(x_3) & -\Phi_3(x_3) & & & \\ & & \ddots & \ddots & & \\ & & & & \Phi_{N-1}(x_N) & -\Phi_N(x_N) \\ B_a\Phi_1(x_1) & & & & & B_b\Phi_N(x_{N+1}) \end{pmatrix}$$

and

$$\text{(B.3b)} \qquad L_+ := \begin{pmatrix} \Phi_1(x_2) & -\Phi_2(x_2) & & & & \\ & \Phi_2(x_3) & -\Phi_3(x_3) & & & \\ & & \ddots & \ddots & & \\ & & & & \Phi_{N-1}(x_N) & -\Phi_N(x_N) \\ & & & & & \Phi_N(x_{N+1}) \quad -I \\ B_a\Phi_1(x_1) & & & & & B_b \end{pmatrix},$$

where $\Phi_k(x)$ are fundamental solutions of (B.1) satisfying the condition

$$\text{(B.4)} \qquad A_k\Phi_k(x_k) + B_k\Phi_k(x_{k+1}) = I_n \ .$$

Again we assume that A_k and B_k are such that $\Phi_k(x)$ is uniquely determined.

B.5 Lemma

The matrices $\boldsymbol{L}$ *and* $\boldsymbol{L}_+$ *are invertible. If* $\boldsymbol{L}^{-1}$ *is partitioned into* N^2 *blocks of size* $n\times n$, *then*

$$(B.5a)\qquad (\boldsymbol{L}^{-1})_{ij} = \begin{cases} -A_i G(x_i,x_{j+1}) - B_i G(x_{i+1},x_{j+1}) + B_i \delta_{i,j} & , j \neq N , \\ A_i \Phi(x_i) + B_i \Phi(x_{i+1}) & , j = N , \end{cases}$$

and if $\boldsymbol{L}_+^{-1}$ *is partitioned into* $(N+1)^2$ *blocks of size* $n\times n$, *then*

$$(B.5b)\qquad (\boldsymbol{L}_+^{-1})_{ij} = \begin{cases} -A_i G(x_i,x_{j+1}) - B_i G(x_{i+1},x_{j+1}) + B_i \delta_{i,j} & , i \le N , j \le N , \\ A_i \Phi(x_i) + B_i \Phi(x_{i+1}) & , i \le N , j = N+1 , \\ -G(x_{N+1},x_{j+1}) & , i = N+1 , j \le N , \\ \Phi(x_{N+1}) & , i = j = N+1 . \end{cases}$$

Proof

First we will consider the 'standard' multiple shooting choice $A_k = I_n$ and $B_k = 0$. Several authors have looked at either the inverse of $\boldsymbol{L}_+$, e.g. [LeOsRu], or at the inverse of $\boldsymbol{L}$, e.g. [dHMa85]. However, a combined proof can be given.

Let the function $v_k(x)$ be the solution of

$$\begin{cases} \dot{v}_k(x) = A(x) v_k(x) + q(x) & , x_k < x < x_{k+1} \\ v_k(x_k) = 0 . \end{cases}$$

Then the solution $y(x)$ of (B.1) can be expressed as

$$y(x) = \Phi_k(x)\, s_k + v_k(x) \quad \text{for } x \in [x_k , x_{k+1}] ,$$

for some $s_k \in \mathbb{R}^n$, with in the $\boldsymbol{L}_+$ case the additional relation

$$y(x_{N+1}) = s_{N+1} .$$

Now the vectors s_k have to be determined by the continuity of $y(x)$ and the boundary conditions, i.e. either

$$\boldsymbol{L} \begin{pmatrix} s_1 \\ \vdots \\ s_{N-1} \\ s_N \end{pmatrix} = \begin{pmatrix} -v_1(x_2) \\ \vdots \\ -v_{N-1}(x_N) \\ \beta - B_b v_N(x_{N+1}) \end{pmatrix} \quad \text{or} \quad \boldsymbol{L}_+ \begin{pmatrix} s_1 \\ \vdots \\ s_{N-1} \\ s_N \\ s_{N+1} \end{pmatrix} = \begin{pmatrix} -v_1(x_2) \\ \vdots \\ -v_{N-1}(x_N) \\ -v_N(x_{N+1}) \\ \beta \end{pmatrix} .$$

Here we see that $\boldsymbol{L}$ contains the same equations as $\boldsymbol{L}_+$ (only the variable s_{N+1} has already been solved). From

$$y(x) = \Phi(x)\beta + \int_a^b G(x,t)\, q(t)\, dt$$

(see §1.1) we obtain the inverse of $\boldsymbol{L}_+$ immediately :

$$
\begin{aligned}
s_k = y(x_k) &= \Phi(x_k)\beta + \sum_{j=1}^{N} \int_{x_j}^{x_{j+1}} G(x_k,t)\,q(t)\,dt \\
&= \Phi(x_k)\beta + \sum_{j=1}^{N} G(x_k,x_{j+1}) \int_{x_j}^{x_{j+1}} \Phi_j(x_{j+1})\Phi_j^{-1}(t)\,q(t)\,dt \\
&= \Phi(x_k)\beta + \sum_{j=1}^{N} G(x_k,x_{j+1})\,v_j(x_{j+1}) \ .
\end{aligned}
$$

For L^{-1} the last term of the sum has to be rewritten as :

$$
G(x_k,x_{N+1})v_N(x_{N+1}) = -\Phi(x_k)B_b\Phi(x_{N+1})\Phi^{-1}(x_{N+1})v_N(x_{N+1}) = -\Phi(x_k)B_b v_N(x_{N+1}) \ .
$$

Now consider any other set of local boundary conditions and denote the related fundamental solutions with $\tilde{\Phi}_k(x)$ and the equivalent of (B.3) with $\tilde{L}$ and $\tilde{L}_+$ respectively. Let $H_k \in \mathbb{R}^{n\times n}$ be such that

$$
\tilde{\Phi}_k(x) = \Phi_k(x)H_k \ ,
$$

then $H_k = [A_k\Phi_k(x_k) + B_k\Phi_k(x_{k+1})]^{-1} = [A_k + B_k\Phi_k(x_{k+1})]^{-1}$.

Hence $\tilde{L} = L\,\mathrm{diag}(H_1,H_2,..,H_N)$ and $\tilde{L}_+ = L_+\mathrm{diag}(H_1,H_2,..,H_N,I_n)$.

Since one can show that $\Phi_k(x_{k+1})\Phi(x_k) = \Phi(x_{k+1})$ for all $k \in \{1, \ldots, N\}$, the relations for $i \neq j$ follow straightforwardly. If $j = i$, then

$$
\begin{aligned}
H_i^{-1}G(x_i,x_{j+1}) &= A_iG(x_i,x_{j+1}) - B_i\Phi_i(x_{i+1})\Phi(x_i)B_b\Phi(x_{N+1})\Phi^{-1}(x_{j+1}) \\
&= A_iG(x_i,x_{j+1}) - B_i\Phi(x_{i+1}).(I_n - B_a\Phi(x_1))\Phi^{-1}(x_{j+1}) \\
&= A_iG(x_i,x_{j+1}) + B_iG(x_{i+1},x_{j+1}) - B_i \ .
\end{aligned}
$$

♦

B.6 Corollary

If the local boundary conditions (B.4) *satisfy the assumption* (1.1.16), *i.e. if for all k, the matrices* ($A_k \mid B_k$) *has orthonormal rows, then*

(B.6a) $\qquad |L^{-1}|_\infty \le \sqrt{n}\,(2N\kappa_{lin}+1) \ ,$

(B.6b) $\qquad |L_+^{-1}|_\infty \le \sqrt{n}\,((2N+2)\kappa_{lin}+1) \ ,$

with κ_{lin} *the conditioning constant of the BVP* (B.1).

Proof

Recall that the conditioning constant was defined in chapter 1 with respect to the Euclidian norm. Now the results of lemma B.5 yield

$$
\begin{aligned}
|L^{-1}|_\infty &\le \max_i \left(\sum_{j=1}^{N-1} |A_i G(x_i, x_{j+1}) + B_i G(x_{i+1}, x_{j+1}) - B_i \delta_{i,j}|_\infty + |A_i \Phi(x_i) + B_i \Phi(x_{i+1})|_\infty \right) \\
&\le \max_i \sqrt{n} \left(\sum_{j=1}^{N-1} |A_i G(x_i, x_{j+1}) + B_i G(x_{i+1}, x_{j+1}) - B_i \delta_{i,j}|_2 + |A_i \Phi(x_i) + B_i \Phi(x_{i+1})|_2 \right) \\
&\le \sqrt{n}\,(2N\kappa_{lin} + 1)\,.
\end{aligned}
$$

For L_+^{-1} a similar estimate can be made.

♦

Appendix C : Logarithmic norm

Let $A \in \mathbb{R}^{m \times m}$ and let $|\ .\ |$ denote any vector norm. The logarithmic norm is defined by

$$\mu[A] := \lim_{h \downarrow 0} \frac{|I_m + hA| - 1}{h}. \tag{C.1}$$

The logarithmic norm depends on the vector norm used.
Define $\rho(A) = \max\{\ \mathrm{Re}(\lambda) \mid \lambda \text{ eigenvalue of } A\ \}$.
The logarithmic norm has the following properties, see e.g. [St,Dah,DeHa]

C.2 Properties

(*a*) $-|A| \le -\mu[-A] \le -\rho(-A) \le \rho(A) \le \mu[A] \le |A|$.

(*b*) $\mu[cA] = c\mu[A] \quad , c \ge 0$.

(*c*) $\mu[A + zI_m] = \mu[A] + \mathrm{Re}(z) \quad , z \in \mathbb{C}$.

(*d*) $\max(\mu[A] - \mu[-B], \mu[B] - \mu[-A]) \le \mu[A+B] \le \mu[A] + \mu[B]$.

(*e*) *convexity* : $\forall_{c \in [0,1]} \ : \ \mu[cA + (1-c)B] \le c\mu[A] + (1-c)\mu[B]$.

(*f*) *continuity* : $|\mu[A] - \mu[B]| \le \max(|\mu[A-B]|, |\mu[B-A]|) \le |A-B|$.

(*g*) *greatest lower bound* : $\min_{\xi \neq 0} \frac{|A\xi|}{|\xi|} \ge \max(-\mu[-A], -\mu[A])$;

hence if A is non-singular, then $\frac{1}{|A^{-1}|} \ge \max(-\mu[-A], -\mu[A])$.

♦

For some Hölder norms an explicit expression of the logarithmic norm can be derived. Let $\mu_p[\ .\ ]$ denote the logarithmic norm with respect to the p-Höldernorm.

C.3 Relations

(*a*) $\mu_1[A] = \max_j (A_{jj} + \sum_{i \neq j} |A_{ij}|)$.

(*b*) $\mu_2[A] = \max_{\xi \neq 0} \frac{\langle A\xi, \xi \rangle}{\langle \xi, \xi \rangle} = \rho(\frac{1}{2}(A + A^T))$.

(*c*) $\mu_\infty[A] = \max_i (A_{ii} + \sum_{j \neq i} |A_{ij}|)$. ♦

Of some special interest are matrices, whose diagonal blocks are $-I_m$ (i.e. their logarithmic norm can easily be seen to be -1) and have small off-diagonal blocks. In order to estimate their logarithmic norm a theorem from [St] can be used (cf. chapter 4).

Let the space $\mathbb{R}^m$ be written as the Cartesian product of lower dimensional spaces, i.e.

$$\forall_{x\in\mathbb{R}^m} \;:\; x = \begin{pmatrix} x_1 \\ \vdots \\ x_N \end{pmatrix}, \quad \textit{where } x_i \in \mathbb{R}^{m_i} \quad \textit{and} \quad \sum_{i=1}^{N} m_i = m \;.$$

Furthermore let $|\,.\,|_i$ denote a monotonic norm on $\mathbb{R}^{m_i}$ and $|\,.\,|_0$ a norm on $\mathbb{R}^N$. Let μ_i be the logarithmic norm based on $|\,.\,|_i$. Partition any matrix $A\in\mathbb{R}^{m\times m}$ into N^2 blocks (A_{ij}) with $A_{ij}\in\mathbb{R}^{m_i\times m_j}$. Let $|A_{ij}|_{ij} := \sup\limits_{x_j\neq 0} \dfrac{|A_{ij}x_j|_i}{|x_j|_j}$. Finally let a norm on $\mathbb{R}^m$ be defined by

$$\forall_{x\in\mathbb{R}^m} \;:\; |x| := \left| \begin{pmatrix} |x_1|_1 \\ \vdots \\ |x_N|_N \end{pmatrix} \right|_0$$

and let μ be the related logarithmic norm.

C.4 Theorem [St]
Let $A\in\mathbb{R}^{m\times m}$. Let $\hat{B}\in\mathbb{R}^{N\times N}$ be defined by

$$\text{(C.4a)} \qquad \hat{B} := \begin{pmatrix} \mu_1[A_{11}] & |A_{12}|_{12} & & \cdots & |A_{1N}|_{1N} \\ |A_{21}|_{21} & \mu_2[A_{22}] & |A_{23}|_{23} & \cdots & |A_{2N}|_{2N} \\ & & \ddots & & \\ & & & \ddots & \\ |A_{N1}|_{N1} & \cdots & & \cdots\; |A_{N-1,N}|_{N-1,N} & \mu_N[A_{NN}] \end{pmatrix}.$$

Then

$$\text{(C.4b)} \qquad \max_i \left(\mu_i[A_{ii}]\right) \le \mu[A] \le \mu_0[\hat{B}] \,.$$

♦

In practice we use either the Euclidian norm or $|\,.\,|_\infty$ for all matrix blocks.

C.5 Lemma

Let $V, J \in \mathbb{R}^{m\times m}$, with V non-singular and $\mu_2[J] < 0$. Then

(C.5a) $$\mu_2[V^T J V] \leq \mu_2[J]\,|V^{-1}|_2^{-2}\,.$$

Proof

$$\mu_2[V^T J V] = \max_{\xi\neq 0} \frac{<V^T J V\xi, \xi>}{<\xi,\xi>} = \max_{\xi\neq 0} \frac{<JV\xi, V\xi>}{<V\xi, V\xi>} \cdot \frac{<V\xi, V\xi>}{<\xi,\xi>}$$

$$= \max_{\xi\neq 0} \left(\mu_2[J] \frac{<V\xi, V, \xi>}{<\xi,\xi>} \right) = \mu_2[J] \min_{\xi\neq 0} \frac{<V\xi, V\xi>}{<\xi,\xi>}$$

$$= \mu_2[J] \cdot |V^{-1}|_2^{-2}\,.$$

◆

Appendix D : Convergence domain of Newton's method

In this thesis we often refer to convergence results for Newton's method from the Newton-Kantorovich theorem and from its affine invariant version. A precise formulation of these theorems is given below.

Let $f \in C^1(D \to \mathbb{R}^m)$, with $D \subset \mathbb{R}^m$, and let $J(x)$ denote the first derivative of $f(x)$. For any starting vector $x^0 \in \mathbb{R}^m$ the Newton iteration is defined by

$$x^{j+1} := x^j - J^{-1}(x^j) f(x^j) \quad , j \geq 0 . \tag{D.1}$$

For any $x \in \mathbb{R}^m$ and $r > 0$, let $B(x;r)$ denote the ball $\{ y \in \mathbb{R}^m \mid | x-y | < r \}$ and let $\bar{B}(x;r)$ denote its closure. A major convergence result is proven in [Ka] and a different proof is presented in [OrRh].

D.2 Theorem (Newton-Kantorovich)
Assume that there is a convex set $D_0 \subset D$ such that

$$\forall_{x,y \in D_0} \; : \; |J(x) - J(y)| \leq \gamma |x-y| . \tag{D.2a}$$

Suppose there exists an $x^0 \in D_0$ such that

$$|J^{-1}(x^0)| \leq \beta \quad , \quad |J^{-1}(x^0) f(x^0)| \leq \eta \quad \textit{and} \quad \alpha := \beta\gamma\eta \leq 0.5 . \tag{D.2b}$$

Define

$$t^* := \frac{1 - \sqrt{1-2\alpha}}{\beta\gamma} \quad , \quad t^{**} := \frac{1 + \sqrt{1-2\alpha}}{\beta\gamma} \tag{D.2c}$$

and assume that $\bar{B}(x^0;t^*) \subset D_0$. Then the Newton iterates $\{x^j\}$ are well-defined, remain in $\bar{B}(x^0;t^*)$ and converge to a solution x^* of $f(x) = 0$, which is unique in $(B(x^0;t^{**}) \cap D_0) \cup \bar{B}(x^0;t^*)$. Moreover, one has the error estimate

$$|x^j - x^*| \leq \frac{(2\alpha)^{2^j}}{\beta\gamma 2^j} \quad , \quad j \geq 0 . \tag{D.2d}$$

♦

In [DeHe] an affine invariant version of this theorem is presented.

D.3 Theorem (affine invariant convergence theorem)
Assume that there is a convex set $D_0 \subset D$ *and a starting vector* $x^0 \in D_0$ *with* $f(x^0)$ *invertible and that there are constants* $\eta, \omega > 0$ *such that*

(D.3a) $|J^{-1}(x^0)f(x^0)| \le \eta$,

(D.3b) $\forall_{x,y \in D_0} : |J^{-1}(x^0)(J(y) - J(x))| \le \omega |y - x|$,

(D.3c) $\alpha := \eta\omega \le 0.5$

and

(D.3d) $\bar{B}(x^0;t^*) \subset D_0$ *with* $t^* := \dfrac{1 - \sqrt{1-2h}}{\omega}$.

Then $J(x)$ *is invertible for all* $x \in B(x^0;t^*)$ *and the Newton iterates remain in* $B(x^0;t^*)$ *and converge to a solution* x^* *of* $f(x) = 0$. *This solution is unique in* $\bar{B}(x^0;t^*) \cup (D_0 \cap B(x^0;t^{**}))$, *where*

$$t^{**} := \frac{1 + \sqrt{1-2h}}{\omega}.$$

Moreover, the following error estimates hold for $j \ge 1$:

(D.3e) $$|x^j - x^*| \le \frac{2\sqrt{1-2\alpha}}{\alpha} \frac{\Theta^{2^j}}{1-\Theta^{2^j}} |x^1 - x^0| \quad , \quad \alpha < \frac{1}{2} ,$$

(D.3f) $$|x^j - x^*| \le 2^{-j+1} |x^1 - x^0| \quad , \quad \alpha = \frac{1}{2} ,$$

$$\text{with } \Theta = \frac{t^*}{t^{**}} = \frac{2\alpha}{(1+\sqrt{1-2\alpha})^2} .$$

♦

Since the conditions of theorem D.2 imply those of D.3 with $\omega = \beta\gamma$, the error bounds (D.3e) and (D.3f) supplement those of theorem D.2.

Appendix E : Convergence of the mixed Euler method

Consider the ODE

$$\text{(E.1)} \qquad \begin{cases} \dot{x}(t) = M(x(t))f(x(t)) & , \ t>0 \ , \ x\in C^1([0,\infty)\to\mathbb{R}^m) \ , \\ x(0) = x^0 \ . \end{cases}$$

The mixed Euler method for this ODE reads

$$\text{(E.2)} \qquad x^{j+1} = x^j + h_j M(x^j)f(x^{j+1}) \quad , j>0 \ .$$

Next we show that the mixed Euler method is a convergent integration method.

E.3 Lemma

Let $D \subseteq \mathbb{R}^m$ be such that $f\in C^2(D\to\mathbb{R}^m)$ and $M\in C^1(D\to\mathbb{R}^{m\times m})$. Let $x^0\in D$ and assume that the solution $x(t)$ of (E.1) *lies in D. Let $T>0$. The class of sequence pairs $S(h)$ is for all $h>0$ defined by*

$$\text{(E.3a)} \qquad S(h) := \{ \textit{sequence pairs } (\{h_j\},\{x^j\}) \textit{ satisfying } (E.2) \mid$$
$$\mid \forall_j : (0<h_j\le h \wedge x^j\in D) \ \wedge \ \exists_{k(\{h_j\})\in\mathbf{N}} : T = \sum_{i=0}^{k(\{h_j\})-1} h_i \} \ .$$

Then

$$\text{(E.3b)} \qquad \lim_{h\downarrow 0} \max \{ \ |x(T)-x^{k(\{h_j\})}| \ \mid (\{h_j\},\{x^j\})\in S(h) \} = 0 \ .$$

Proof

Let $h>0$ and $(\{h_j\},\{x^j\})\in S(h)$. From (E.2) and the definition of discretization error we get

$$\begin{cases} h_j\delta(t_j,x,h_j) = x(t_{j+1})-x(t_j)-h_jM(x(t_j))f(x(t_{j+1})) \ , \\ 0 = x^{j+1}-x^j-h_jM(x^j)f(x^{j+1}) \ . \end{cases}$$

Hence, by subtraction,

$$x(t_{j+1})-x^{j+1} = x(t_j)-x^j + h_j\delta(t_j,x,h_j) + h_jM(x(t_j))[f(x(t_{j+1}))-f(x^{j+1})]$$
$$+ h_j[M(x(t_j))-M(x^j)]f(x^{j+1}) \ .$$

Define $e_j := |\ x(t_j) - x^j\ |$, $j\ge 0$. Now let C_f,C_M,C_J be upper bounds on D on the norms of $f(x)$, $M(x)$, $f'(x)$, respectively, and let L_M be a bound on the Lipschitz constant of $M(x)$ on D. If $h<(C_MC_J)^{-1}$, then

$$e_{j+1} \le e_j + h_j|\delta(x^j,h_j)| + h_j C_M C_J e_{j+1} + h_j L_m C_f e_j$$

$$\Rightarrow \quad e_{j+1} \le \frac{1}{1-h_j C_M C_J}((1+h_j L_M C_f)e_j + h_j|\delta(x^j,h_j)|).$$

For notational convenience we define $C := \max(L_M C_f, C_M C_J)$, $\beta = \frac{1}{1-hC}\ln\frac{2-hC}{hC}$ and $l := k(\{h_j\})-1$. Some simple calculus shows that

$$\forall_{0<h_i<h} \; : \; \frac{1+h_i L_M C_f}{1-h_i C_M C_J} \le \frac{1+h_i C}{1-h_i C} \le e^{\beta C h_i}.$$

Now the error in the approximation of $x(T)$ can be estimated by

$$\begin{aligned}
|x(T)-x^{k(\{h_j\})}| &\le \sum_{i=0}^{l}\prod_{p=i+1}^{l}\frac{1+h_p L_M C_f}{1-h_p C_M C_J}\cdot\frac{h_i|\delta(x^i,h_i)|}{1-h_i C_M C_J} \\
&\le \sum_{i=0}^{l} h_i|\delta(x^i,h_i)|(1+h_i C)\prod_{p=i}^{l}\frac{(1+h_p C)}{(1-h_p C)} \\
&\le \max_{i\le l}|\delta(x^i,h_i)|(1+hC)\sum_{i=0}^{l} h_i e^{\beta C(x_{l+1}-x_i)} \\
&\le \max_{i\le l}|\delta(x^i,h_i)|(1+hC)\,T\,e^{\beta CT} \\
&\to 0 \quad , \text{ if } \quad h\downarrow 0.
\end{aligned}$$

♦

Appendix F : Boundedness of the Riccati-matrices of the preconditioning process

Consider the well-conditioned BVP with separated BC

(F.1a) $\dot{y}(x) = A(x)y(x) \quad , a < x < b ,$

(F.1b) $B_a y(a) + B_b y(b) = \beta ,$

$$\text{with } B_a = \begin{pmatrix} 0 \\ B_{a2} \end{pmatrix} \updownarrow p \quad \text{and } B_b = \begin{pmatrix} B_{b1} \\ 0 \end{pmatrix} \updownarrow n-p .$$

Let $Y(x)$ be the fundamental solution that satisfies

(F.2) $$B_a Y(a) + B_b Y(b) = I_n .$$

Then the Green's function $G(x,t)$ can be expressed in terms of $Y(x)$ by

(F.3) $$G(x,t) = \begin{cases} Y(x)PY^{-1}(t) & , x \geq t , \\ Y(x)(I_n - P)Y^{-1}(t) & , x < t , \end{cases}$$

$$\text{with } P := \begin{pmatrix} 0 & 0 \\ 0 & I_p \end{pmatrix} .$$

Let κ_{lin} denote the conditioning constant of (F.1).

We want to derive an upper bound on the Riccati matrices and the boundary conditions formed in the process to obtain a preconditioner in chapter 4. For the precise formulation of the algorithm we refer to §4.1, here we only mention the relevant relations.

The matrices $Q_k \in \mathbb{R}^{n \times n}$, $k \in \{1,2, \ldots, N+1\}$ are orthogonal and the matrices $R_k \in \mathbb{R}^{(n-p) \times p}$. The following relations hold

(F.4) $$B_a Q_1 = \begin{pmatrix} 0 & 0 \\ 0 & B_a^{(2)} \end{pmatrix} ,$$

(F.5) $$Y(x_{k+1})Y^{-1}(x_k)Q_k = Q_{k+1} \begin{pmatrix} B_k & C_k \\ 0 & E_k \end{pmatrix} , \quad B_k \in \mathbb{R}^{(n-p) \times (n-p)} , \; E_k \in \mathbb{R}^{p \times p} , \quad k \in \{1,..,N\} ,$$

(F.6) $$B_b Q_{N+1} = \begin{pmatrix} B_b^{(1)} & B_b^{(2)} \\ 0 & 0 \end{pmatrix}$$

(F.7) $$R_{N+1} = \left(B_b^{(1)}\right)^{-1} B_b^{(2)} ,$$

(F.8) $$B_k R_k - C_k - R_{k+1} E_k = 0 \quad , k = N, N-1, .., 1 ,$$

(F.9) $$S_k = \begin{pmatrix} I_{n-p} & R_k \\ 0 & I_p \end{pmatrix} \quad , k \in \{1, .., N\} ,$$

(F.10) $$Y(x_{k+1}) Y^{-1}(x_k) = Q_{k+1} S_{k+1}^{-1} \begin{pmatrix} B_k & 0 \\ 0 & E_k \end{pmatrix} S_k Q_k^{\top} \quad , k \in \{1, .., N\} .$$

F.11 Lemma

(*i*) $$|R_k|_2 \le \kappa_{lin} \quad , k \in \{1, .., N+1\} ,$$

(*ii*) $$\left|\left(B_a^{(2)}\right)^{-1}\right|_2 \le \kappa_{lin} \quad \textit{and} \quad \left|\left(B_b^{(1)}\right)^{-1}\right|_2 \le \kappa_{lin} ,$$

(*iii*) $$|B_a^{(2)}|_2 \le 1 \quad \textit{and} \quad |B_b^{(1)}|_2 \le 1 .$$

Proof

Define the $n \times n$ matrices W_k, $k \in \{1, . . ,N+1\}$, by

(F.11a) $$W_k := \begin{pmatrix} B_k^{-1} B_{k+1}^{-1} .. B_N^{-1} \left(B_b^{(1)}\right)^{-1} & 0 \\ 0 & E_{k-1} E_{k-2} .. E_1 \left(B_a^{(2)}\right)^{-1} \end{pmatrix} .$$

First we prove that $Y(x_k) = Q_k S_k^{-1} W_k$. To this end we define the matrices $A_k := Y(x_{k+1}) Y^{-1}(x_k)$; they induce the following difference equation

(*) $$\begin{aligned} y_{k+1} &= A_k y_k \quad , k = 1, .., N , \\ B_a y_1 + B_b y_{N+1} &= \gamma . \end{aligned}$$

Now both $\{Y(x_k)\}$ and $\{ Q_k S_k^{-1} W_k \}$ are fundamental solutions of (*) and they satisfy the same boundary condition, viz.

$$\begin{aligned} B_a Y(x_1) + B_b Y(x_{N+1}) &= I_n , \\ B_a Q_1 S_1^{-1} W_1 + B_b Q_{N+1} S_{N+1}^{-1} W_{N+1} &= I_n . \end{aligned}$$

Hence both fundamental solution have to be identical, i.e.

$$\forall_{k\in\{1,..,N+1\}} \quad : \quad Y(x_k) = Q_k S_k^{-1} W_k .$$

Now the first two statements can easily be proven :

(*i*) $$\kappa_{lin} \geq |Y(x_k)PY^{-1}(x_k)|_2 = |S_k^{-1}W_k P W_k^{-1} S_k|_2$$
$$= |S_k^{-1}PS_k|_2 = \left|\begin{pmatrix} 0 & -R_k \\ 0 & I_p \end{pmatrix}\right|_2 \geq |R_k|_2 .$$

(*ii*) $$\kappa_{lin} \geq |Y(x_1)|_2 \geq |Y(x_1)P|_2 = |Q_1 S_1^{-1} \begin{pmatrix} 0 & 0 \\ 0 & \left(B_a^{(2)}\right)^{-1} \end{pmatrix}|_2$$
$$= \left|\begin{pmatrix} 0 & -R_1\left(B_a^{(2)}\right)^{-1} \\ 0 & \left(B_a^{(2)}\right)^{-1} \end{pmatrix}\right|_2 \geq \left|\left(B_a^{(2)}\right)^{-1}\right|_2 .$$

and $$\kappa_{lin} \geq |Y(x_{N+1})|_2 \geq |Y(x_{N+1})(I_n - P)|_2 = |Q_{N+1}S_{N+1}^{-1} \begin{pmatrix} \left(B_b^{(1)}\right)^{-1} & 0 \\ 0 & 0 \end{pmatrix}|_2$$
$$= \left|\begin{pmatrix} \left(B_b^{(1)}\right)^{-1} & 0 \\ 0 & 0 \end{pmatrix}\right|_2 \geq \left|\left(B_b^{(1)}\right)^{-1}\right|_2 .$$

Finally, (*iii*) can be derived from the assumption that $(B_a \mid B_b)$ has orthonormal rows :

$$|B_a^{(2)}|_2 = |B_a Q_1|_2 = 1 ,$$
$$|B_b^{(1)}|_2 = \left|\begin{pmatrix} B_b^{(1)} & B_b^{(2)} \\ 0 & 0 \end{pmatrix}(I_n - P)\right|_2 = |B_b Q_{N+1}(I_n - P)|_2 \leq 1 .$$

♦

Let the matrices Q_k be partitioned according to

$$Q_k = \underset{n-p}{(Q_{k,1}} \,|\, \underset{p}{Q_{k,2})} .$$

F.12 Lemma

(*i*) $\forall_k$: $\text{range}(Y(x_k)(I_n - P)) = \text{range}(Q_{k,1})$.

(*ii*) $\forall_k$: $\text{range}(Y(x_k)P) = \text{range}(Q_{k,2} - Q_{k,1}R_k) = \text{range}(Q_{k,1} + Q_{k,2}R_k^T)^\perp$.

Proof

In the proof of lemma F.11 it was derived that $Y(x_k) = Q_k S_k^{-1} W_k$, with W_k as in (F.11a).

(*i*) Since W_k is non-singular and has a zero left lower block :

$$\forall_k \quad : \quad \text{range}(Y(x_k)(I_n - P)) = \text{range}(Q_k S_k^{-1}(I_n - P)) = \text{range}(Q_{k,1}) .$$

(*ii*) The matrix product $Q_k S_k^{-1}$ can be written in the form :

$$Q_k S_k^{-1} = (Q_{k,1} \,|\, Q_{k,2} - Q_{k,1}R_k) .$$

Since the matrix W_k is non-singular and has a zero right upper block :

$$\forall_k \quad : \quad \text{range}(Y(x_k)P) = \text{range}(Q_k S_k^{-1} P) = \text{range}(Q_{k,2} - Q_{k,1}R_k) .$$

The second part of the statement uses the orthonormality of the matrices Q_k. The relation

$$(Q_{k,1}^T + R_k Q_{k,2}^T)(Q_{k,2} - Q_{k,1}R_k) = 0 - R_k + R_k - 0 = 0$$

implies that $\text{range}(Q_{k,2} - Q_{k,1}R_k)$ is a subspace of $\ker(Q_{k,1}^T + R_k Q_{k,2}^T)$. Both spaces have to be equal, because they have the same dimension. Hence

$$\forall_k \quad : \quad \text{range}(Q_{k,2} - Q_{k,1}R_k) = \text{range}(Q_{k,1} + Q_{k,2}R_k^T)^\perp .$$

♦

References

[AbBr] J.P. Abbott, R.P. Brent, Fast local convergence with single and multistep methods for nonlinear equations, J.Austral.Math.Soc.19 (series B) (1975), pp 173-199.

[AsChRu] U. Ascher, J. Christiansen, R.D. Russell, A collocation solver for mixed order systems of boundary value problems. Math. Comp. 33 (1979) pp 659-679.

[AsMa] U. Ascher, R.M.M. Mattheij, General framework, stability and error analysis for numerical stiff boundary value methods. Numer. Math. 54 (1988) pp 355-372.

[AsMaRu] U.M. Ascher, R.M.M. Mattheij, R.D. Russell, Numerical solution of boundary value problems for ordinary differential equations. Englewood Cliffs : Prentice-Hall, 1988.

[AsPC] U. Ascher, S.Y. Pat Chan, On parallel methods for boundary value ODE's. Technical report 89-19, University of British Columbia.

[AsRu] U. Ascher, R.D. Russell, Reformulation of boundary value problems into "standard" form. Siam Review 23 (1981) pp 238-254.

[BaAs] G. Bader, U. Ascher, A new basis implementation for a mixed order boundary value ODE solver. SIAM J. Scient. Stat. Comput 8 (1987) pp 483-500.

[Be53] R. Bellman, Stability theory of differential equations. New York : McGraw-Hill, 1953.

[Bo] P.T. Boggs, The solution of nonlinear systems of equations by A-stable integration techniques, SIAM J.Numer.Anal. 8 (1971), pp 767-785.

[dBWe] C. de Boor, R.Weiss, SOLVEBLOCK : A package for solving almost block diagonal linear systems. ACM Trans. Math Software 6 (1980) pp 80-87.

[Br] F.H. Branin jr., Widely convergent methods for finding multiple solutions of simultaneous nonlinear equations, IBM J.Res.Develop. 1972, pp 504-522.

[Co67] W.A. Coppel, Dichotomies and reducibility. J. of Diff.Eq. 3 (1967) pp 500-521.

[Co78] W.A. Coppel, Dichotomies in stability theory. Berlin : Springer Verlag, 1978.

[Con] S.D. Conte, The numerical solution of linear boundary value problems. SIAM Review 8 (1966), pp 309-321.

[Da] D.F. Davidenko, On a new method of numerical solution of systems of nonlinear equations. Dokl.Akad. Nauk SSSR 88 (1953), pp 601-602.

[Dah] G. Dahlquist, Stability and error bounds in the numerical integration of ordinary differential equations. Thesis 1958, in: Trans. Royal Inst. of Technology, No.130, Stockholm.

[DeHa] C. Desoer, H. Haneda, The measure of a matrix as a tool to analyze computer algorithms for circuit analysis. IEEE Trans. Circuit Th. 19, pp 480-486.

[DeHe] P. Deuflhard, G. Heindl, Affine invariant convergence theorems for Newton's method and extensions to related methods, SIAM J. Numer.Anal. 16 (1979) pp 1-10.

[DePeRe] P. Deuflhard, H.-J. Pesch, P. Rentrop, A modified continuation method for the numerical solution of nonlinear two-point boundary value problems by shooting techniques. Numer.Math. 26 (1976) pp 327-343.

[DeVe] K. Dekker, J.G. Verwer, Stability of Runge-Kutta methods for stiff nonlinear differential equations. Amsterdam : CWI Monographs, 1984.

[Go] S.K. Godunov, Numerical solution of boundary value problems for systems of linear ordinary differential equations. Usp.Mat.Nauk 16 (1961), pp 171-174.

[GoLo] G.H. Golub, C.F. van Loan, Matrix computations. Baltimore : Johns Hopkins Univ. Press, 1983, 2nd ed.

[HeBe] M. Hermann, H. Berndt, RWPM, a multiple shooting code for nonlinear two-point boundary value problems: version 4, part I-III Preprint 67,68,69 FSU Jena.

[dHe] C. den Heyer, The numerical solution of nonlinear operator equations by imbedding methods. Amsterdam: Mathematical Centre Tracts 107, 1979.

[Ho] J.F. Holt, Numerical solution of nonlinear two-point boundary problems by finite difference methods. Comm. of ACM 7 (1964), pp 366-373.

[dHMa85] F. de Hoog, R.M.M. Mattheij, The role of conditioning in shooting techniques in : Numerical Boundary value ODE's, U.Ascher, R.Russell eds.. Birkhäuser Boston Inc. 1985.

[dHMa87] F. de Hoog, R.M.M. Mattheij, On dichotomy and well conditioning in BVP. SIAM J. Numer. Anal 24 (1987), pp 89-105.

[Ka] L. Kantorovich, On Newton's method for functional equations (Russian). Dokl. Akad. Nauk SSSR 59 (1948), pp 1237-1240.

[Ke76] H.B. Keller, Numerical solution of two point boundary value problems. CBMS Regional Conference Series in Applied Mathematics, 24. Philadelphia: SIAM, 1976.

[KuHl] M. Kubíček, V. Hlaváček, Numerical solution of nonlinear boundary value problems with applications. Englewood Cliffs: Prentice-Hall, 1983.

[vLo] P.M. van Loon, Continuous decoupling transformations for linear boundary value problems. Amsterdam: CWI-tracts 52, 1988.

[Ma82] R.M.M. Mattheij, The conditioning of linear boundary value problems. SIAM J. Num. Anal. 19 (1982), pp 963-978.

[Ma85] R.M.M. Mattheij, Decoupling and stability of algorithms for bounadry value problems. SIAM Review 27 (1985), pp 1-44.

[Ma89] R.M.M. Mattheij, Conditions and conditioning, stability and stabilization. Appl. Math. Comp. 31 (1989), pp 538-554.

[MaSt] R.M.M. Mattheij, G. Staarink, Implementing multiple shooting for nonlinear BVP. Report of Eindhoven University of Technology RANA report 87-14.

[MaSc58] J.L. Massera, J.J. Schäffer, Linear differential equations and functional analysis I. Annals of Math. 67 (1958), pp 517-573.

[MaSc66] J.L. Massera, J.J. Schäffer, Linear differential equations and function spaces. New York : Academic Press, 1966.

[Me68] G.H. Meyer, On solving nonlinear equations with one-parameter operator imbedding. SIAM J. Numer. Anal. 5 (1968), pp 739-752.

[Me73] G.H. Meyer, Initial value methods for boundary value problems. New York: Academic Press, 1973.

[O'Ma] R. O'Malley Jr., Introduction to singular perturbations. New York: Academic Press, 1974.

[OrRh] J.M. Ortega, W.C. Rheinboldt, Iterative solution of nonlinear equations in several variables. New York : Academic Press, 1970.

[Os] M.R. Osborne, The stabilized march is stable. SIAM J.Num. Anal. 16 (1979), pp 923-933.

[PaGl] M. Paprzycki, I. Gladwell, Solving almost block diagonal systems on parallel computers. SMU Math Rept 89-18.

[Pe] O. Perron, Die Stabilitätsfrage bei Differentialgleichungen. Math.Z. 32 (1930), pp 703-728.

[RoSh] S. Roberts, J. Shipman, Continuation in shooting methods for two-point boundary value problems. J. of Math. Anal. and Appl. 18 (1967), pp 45-58.

[Ru] R.D. Russell, Mesh Selection methods, in Codes for boundary value problems, Lecture Notes in Computer Science 74, Childs et al., eds.. Berlin: Springer, 1979, pp 228-242.

[ScWa] M.R. Scott, H.A. Watts, A systematized collection of codes for solving two-point boundary-value problems. In : Numerical methods for differential systems by L. Lapidus , W.E. Schiesser. New York : Academic Press, 1976, pp 197-228.

[SmMiKe] M.D. Smooke, J.A. Miller, R.J. Kee, Solution of premixed and counterflow diffusion flame problems by adaptive boundary value methods. In: Numerical boundary value ODE's, U.M.Ascher, R.D.Russell, ed.. Boston : Birkhäuser, 1985, pp 303-317.

[St] T. Ström, On logarithmic norms, SIAM J. Numer.Anal. 12 (1975), pp 741-753.

[Tr] B. Troesch, A simple approach to a sensitive two-point boundary value problem. J. of Comput. Physics 21 (1976) pp 279-290.

[Va] J.M. Varah, Alternate row and column elimination for solving linear systems, SIAM J.Numer.Anal. 13(1976), pp 71-75.

[Ve] G.W. Veltkamp, private communication.

[Wa] H. Wacker, A summary of the developments on imbedding methods, in Continuation methods, ed. H.Wacker. New York: Academic Press, 1978, pp 1-35.

[Was] E. Wasserstrom, Numerical solution by the continuation method. SIAM Review 15 (1973) pp 209-224.

[We] R. Weiss, The application of implicit Runge Kutta and collocation methods to boundary value problems. Math. Comp. 28 (1974) pp 449-464.

[Wr] S.J. Wright, Stable parallel algorithms for two-point boundary value problems. To appear in SISSC.

Index

Abbreviations

BC = boundary conditions
BVP = boundary value problem
IVP = intial value problem
ODE = ordinary differential equation
PDE = partial differential equation

Samenvatting

In dit proefschrift wordt de meervoudige schietmethode voor niet-lineaire tweepunts randwaardeproblemen bestudeerd en worden twee varianten op deze methode beschreven en geanalyseerd.

Het randwaardeprobleem (RWP)

(S.1a) $\dot{y}(x) = h(x,y)$, $a<x<b$, $y : [a,b] \rightarrow \mathbb{R}^n$ en $h : [a,b]\times\mathbb{R}^n \rightarrow \mathbb{R}^n$,

(S.1b) $g(y(a),y(b)) = 0$, $g : \mathbb{R}^n\times\mathbb{R}^n \rightarrow \mathbb{R}^n$,

heet goed geconditioneerd rond de geïsoleerde oplossing $y^*(x)$, als kleine veranderingen in de functies h en/of g, slechts een kleine verandering teweeg brengen in de oplossing (dit wordt nauwkeuriger omschreven in §1.4). Het karakter van dit probleem laat toe dat zijn linearisatie rond $y^*(x)$ zowel sterk stijgende als sterk dalende oplossingscomponenten bevat.

Voor de meervoudige schietmethode wordt het interval $[a,b]$ opgedeeld in N deelintervallen $[x_k, x_{k+1}]$, $k\in\{1, \ldots, N\}$, waarbij

$$a = x_1 < x_2 < \ldots < x_{N+1} = b$$

en wordt op ieder deelinterval een beginwaarde probleem geformuleerd

(S.2a) $\dot{y}(x) = h(x,y)$, $x_k<x<x_{k+1}$,

(S.2b) $y(x_k) = s_k$, $s_k\in\mathbb{R}^n$.

De schietvectoren s_k moeten worden opgelost uit de voorwaarde dat de lokale oplossingen tezamen een continue functie op $[a,b]$ vormen die voldoet aan de globale randvoorwaarde (S.1b). Dit kan symbolisch worden weergegeven met de vergelijking

(S.3) $f(s) = 0$,

met $s := (s_1^{\top}, \ldots, s_N^{\top})^{\top} \in \mathbb{R}^{nN}$.

Indien de linearisatie van (S.1) sterk stijgende oplossingscomponenten bevat, is $f(s)$ zeer gevoelig voor veranderingen van s in sommige richtingen. De oorzaak hiervan is gelegen in het feit dat de lokale beginvoorwaarden (S.2b) niet in staat zijn de sterk groeiende componenten te controleren, met andere woorden (S.2) is een slecht geconditioneerd probleem. Dit heeft onder meer tot gevolg dat de norm en de Lipschitz-constante van de Jacobiaan $J(s)$ van f groot zijn, hetgeen de grootte van het convergentiegebied van Newton's methode negatief kan beïnvloeden.

Een alternatieve oplossingsmethode voor (S.3) is het inbedden van $f(s)$ in een geprecondi-tioneerd beginwaardeprobleem.

(S.4a) $\frac{ds}{dt} = M(s)f(s)$, $t>0$,

(S.4b) $s(0) = s^0$, $s^0 \in \mathbb{R}^{nN}$,

Iedere oplossing van (S.3) is een 'steady state' van (S.4). In dit proefschrift wordt een goedkope impliciete integratie methode voor (S.4) beschreven en worden enkele eigen-schappen van deze methode afgeleid. De preconditioneerder $M(s)$ wordt zo gekozen dat deze de stijgende en dalende oplossingscomponenten van elkaar scheidt en bovendien de stijgende componenten per deelinterval effectief terugwaarts gebruikt. Dit geeft een stelsel (S.4) dat asymptotisch stabiel is rond de oplossing van (S.3).

Een andere oplosmethode voor (S.1) is een variant van meervoudig schieten die op ieder deelinterval, in plaats van een beginwaardeprobleem, een randwaardeprobleem definieert :

(S.5a) $\dot{y}(x) = h(x,y)$, $x_k < x < x_{k+1}$,

(S.5b) $A_k y(x_k) + B_k y(x_{k+1}) = s_k$, $s_k \in \mathbb{R}^n$.

Dit biedt de mogelijkheid ook de stijgende componenten te beheersen en lokaal goed geconditioneerde problemen te definiëren. In dat geval zal (S.3) een goed geconditioneerd probleem zijn en mogen we verwachten dat deze oplosbaar is met behulp van Newton's methode. Lokaal hebben we nu echter wederom niet-lineaire tweepunts randwaardeproble-men, zij het dat deze ieder van kleinere omvang zijn. Deze lokale problemen kunnen dan opgelost worden met bijvoorbeeld collocatie of eindige differenties. Bij een sequentiële implementatie leidt dit tot een geringer geheugen gebruik dan toepassing van collocatie of eindige differenties op het oorspronkelijke probleem (S.1). Daarnaast leent deze aanpak zich uitstekend voor parallellisatie.

Dankwoord

Op deze plaats wil ik iedereen bedanken die mij direct of indirect heeft geholpen bij het voltooien van dit 'levenswerk'.

Een bijzonder woord van dank ben ik verschuldigd aan prof.dr. R.M.M. Mattheij voor de enthousiaste begeleiding en goede raad. Onze discussies waren vaak levendig; u en ik waren beiden vasthoudend ten aanzien van een eenmaal gevormde mening.

Daarnaast ben ik prof.dr. G.W. Veltkamp zeer erkentelijk voor de zorgvuldige wijze waarop hij de verschillende versies van het proefschrift heeft gelezen; de daaruit voortvloeiende suggesties voor wijzigingen waren zeer waardevol.

Tenslotte dank ik P.J. den Haan voor zijn goed-gestructureerde implementatie van het UMS-programma en de genereuze wijze waarop hij mij heeft laten delen in zijn expertise op het gebied van WordPerfect en Fortran. Wij hebben ruim 4 jaar dezelfde kamer gedeeld; ondanks onze verschillende ideeën omtrent de ideale kamertemperatuur, ben je voor mij een goede en collegiale kamergenoot geweest.

Curriculum Vitae

De schrijfster van dit proefschrift werd geboren op 17 mei 1964 te Leidschendam.
De eerste drie jaren VWO onderwijs volgde zij aan de Zandvliet scholen gemeenschap te 's-Gravenhage, waarna zij deze opleiding vervolgde en voltooide (in 1982) aan de Nassau scholen gemeenschap te Breda.
Zij studeerde van augustus 1982 tot en met september 1987 wiskunde aan de Technische Universiteit Eindhoven (TUE) en studeerde af bij prof. J. de Graaf op een onderwerp uit de functionaal analyse. Van oktober 1987 t/m december 1991 was zij als AIO in dienst van de TUE en verrichtte promotie onderzoek onder leiding van prof. R.M.M. Mattheij.

Stellingen

mevr. ir. M.E. Kramer
(25.2.'92, TUE)

- 1 -

Oplosmethoden voor tweepunts randwaardeproblemen moeten bij voorkeur rekening houden met de dichotomie-structuur van het probleem.

- 2 -

Zij $f(s) = 0$ de verzameling van vergelijkingen die resulteert bij toepassing van de meervoudige schietmethode op een niet-lineair randwaardeprobleem. Dan kan de Lipschitz-constante van de Jacobiaan van $f(s)$ van dezelfde orde van grootte zijn, als de conditie constanten van de beginwaardeproblemen op de deelintervallen, die gebruikt worden in het schietproces.

- 3 -

Zij $f \in C^1(\mathbb{R}^n \to \mathbb{R}^n)$ met nulpunt x^* en zij $\{x^k\}$ een rij iteranden verkregen door toepassing van impliciete Euler methode met stapgrootten $\{h_k\}$ op de differentiaal vergelijking

$$\dot{x} = f(x) .$$

Indien er een bol $B(x^*;R)$ is zodat $x^0 \in B(x^*;R)$ en

$f(x)$ heeft een negatieve eenzijdige Lipschitz-constante,

of $f'(x)$ heeft een negatieve logaritmische norm,

zeg $-\alpha$, op $B(x^*;R)$, dan geldt dat

$$\forall_{k \geq 1} \quad : \quad x^k \in B\left(x^* + \frac{1}{2(1+h\alpha)}(x^{k-1} - x^*) \; ; \; \frac{|x^{k-1} - x^*|}{2(1+h\alpha)}\right) .$$

- 4 -

Laat $|\,.\,|$ een semi-norm zijn op de ruimte van $n \times n$ matrices, die voldoet aan

$$\forall_{B \in \mathbb{R}^{n \times n}} \quad \forall_{k \in \mathbf{N}} \quad : \quad |B^k| \leq |B|^k .$$

Indien $|\,2B - I_n\,| \leq 1$, dan geldt

$$\forall_{k \in \mathbf{N}} \quad : \quad |(I_n - B)B^k| \leq 2^{-k} \binom{k}{[\frac{1}{2}k]} \leq \sqrt{\frac{2}{\pi k}} .$$

- 5 -

Het begrip numerieke range (voor definitie zie [1]) is niet geschikt om het begrip hermitische operator uit te breiden naar Banachruimten.

[1] F.F. Bonsall, J. Duncan, Numerical ranges of operators on normed spaces and elements of normed algebras. London : Cambridge university press, 1971.

- 6 -

De door Kramer *et.al.* [2] op theoretische gronden voorspelde analogie in het karakter van structurele faseovergangen in silicas, kan met de methode van Tezuki *et.al.* [3] experimenteel getest worden.

[2] G.J. Kramer, B.W.H. van Beest & R.A. van Santen, Nature **351**, 636 (1991)
[3] Y. Tezuki, S. Shin & M. Ishigame, Phys.Rev.Lett. **66**, 2356 (1991).

- 7 -

Het feit dat de programmeertaal FORTRAN veel vrijheid biedt, leidt er vaak toe dat fouten in de programmatuur niet of in een laat stadium worden ontdekt.

- 8 -

De hedendaagse architectuur lijkt vooral met zichzelf in discussie te treden en niet met de maatschappij.

- 9 -

Het heeft er de schijn van dat het merendeel van de mannen de gevoelswaarde van de term 'vrouwonvriendelijk' niet kent.