Timestepping for solving nonlinear equations

Kramer, M.E.; Mattheij, R.M.M.

Published: 01/01/1990

Document Version
Publisher's PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:

- A submitted manuscript is the author's version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
- The final author version and the galley proof are versions of the publication after peer review.
- The final published version features the final layout of the paper including the volume, issue and page numbers.

Link to publication

Citation for published version (APA):

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

Take down policy
If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Download date: 23. Jan. 2019
TIMESTEPPING FOR SOLVING NONLINEAR EQUATIONS

by

M.E. Kramer
R.M.M. Mattheij

RANA 90-12
October 1990

Reports on Applied and Numerical Analysis
Department of Mathematics and Computing Science
Eindhoven University of Technology
P.O. Box 513
5600 MB Eindhoven
The Netherlands
Timestepping for solving nonlinear equations

by

M.E. Kramer
R.M.M. Mattheij

Eindhoven University of Technology
Dept. of Math. and Comp.Sc.
P.O. Box 513
5600 MB Eindhoven
ABSTRACT

This paper describes and analyses a method for solving nonlinear equations. It is based on defining a suitable initial value ODE, which is solved by a special implicit numerical discretisation scheme. This is shown to be asymptotically equivalent to (damped) Newton if one uses Davidenko's equation. The theory is illustrated by numerical examples and comparisons with related results are discussed.
§1 Introduction

There exists a wide variety of methods for studying nonlinear problems of the type

\[ \begin{align*}
(1.1) \quad f(x^*) &= 0.
\end{align*} \]

A large class of methods can be found among Newton's algorithm and its variants (cf. [9]).

Given an initial estimate \( x_0 \) for \( x^* \), a sequence \( \{x_i\} \) is generated through

\[ \begin{align*}
(1.2) \quad x_{i+1} &= x_i - J^{-1}(x_i) f(x_i), \quad i \geq 0
\end{align*} \]

where \( J \) is the functional derivative of \( f \). The thus resulting discrete initial value problem might be considered as a discretization of the (continuous) initial value problem (cf. [10], [12])

\[ \begin{align*}
(1.3) \quad x(0) &= x_0 \\
\frac{dx}{dt} &= J^{-1}(x) f(x), \quad t > 0.
\end{align*} \]

The latter differential equation, often called Davidenko's equation, see [6], is sometimes referred to as the closure of (1.2), cf. [2, Ch. 6]; indeed if one discretizes (1.3) through Euler forward with stepsize 1, then Newton's method appears. If this stepsize is smaller than 1 this results in a damped Newton method, which is often used when full Newton fails to converge. Unfortunately, this interesting point of view does not reveal why and how certain stepsizes should be chosen and even less why full Newton converges so fast when it does. It may even seem unnatural that one is not allowed to take larger steps eventually, as one is in fact trying to find a stable restpoint of an ODE (of course it is well-known that Euler forward limits the maximum stepsize).

Integration methods for the IVP (1.3) have been discussed by various authors (see [1], [3], [4]). In [3] the trapezoidal rule is used to solve (1.3). This method is A-stable, but if \( \{x_i\} \) approaches the steady state \( x = x^* \), convergence is still slow. This is due to the fact that once the stepsize is chosen large, the errors \( (x_i - x^*) \) satisfy \( x_{i+1} - x^* \approx x^* - x_i \). The mixed Euler method, we introduce in §2, does not suffer from this drawback as will be shown in §4. In [1] the use of explicit Runge-Kutta and multistep methods is investigated, where the stepsize converges to a special (finite) value to yield ultimate second order convergence. However, since the stepsize is bounded, a considerable number of steps may be necessary. Various other continuation methods have been studied see e.g. [8], [12].

Comparing the imbedding (1.3) to so-called false transient techniques where one tries to approximate restpoints for parabolic problems (cf. [9]), it makes sense to try and solve (1.3) by a suitable implicit method, which does not have stepsize restrictions. We shall discuss and analyze a hybrid Euler discretization, which has some nice properties. If one takes \( M(x) = -J^{-1}(x) \), then it is competitive with damped Newton and at the same time it gives a reasonable model for determining the stepsize; moreover the method is asymptotically identical to full Newton, thus appearing to be a much better candidate for understanding Davidenko's equation as a closure.

Instead of (1.3) we consider more generally the IVP

\[ \begin{align*}
1
\end{align*} \]
\[ x(0) = x_0 \]

(1.4) \[ \frac{dx}{dt} = M(x)f(x), \quad t > 0 \]

where \( M(x) \) should be chosen such that \( x^* \) is a stable restpoint of the ODE above in a large domain.

The IVP (1.4) is considered in more detail in the next section. In §3 we introduce our numerical integration scheme for this initial value problem and show that the method is asymptotically equivalent to backward Euler applied to a slightly perturbed ODE. Next we look into the convergence properties of the algorithm as a solver for nonlinear problems in section 4. Finally we describe some computational aspects and give a number of numerical examples in §5.
§2 Finding zeros through initial value problems

Our aim is to solve the equation

\[(2.1) \quad f(x) = 0 , \]

\(f \in C^2(\mathbb{R}^n \to \mathbb{R}^n)\), if only an approximation \(x_0\) of the solution \(x^*\) is available. A well-known path following method is Davidenko's equation (cf. [6])

\[(2.2a) \quad \dot{x}(t) = -J^{-1}(x)f(x), \quad t \geq 0 \]

\[(2.2b) \quad x(0) = x_0 . \]

Apparently \(x^*\) is a rest point of the ODE (2.2a) which is asymptotically stable if \(J(x^*)\) is nonsingular. Monitoring the path \(x(t)\) can be troublesome, if it crosses an area where \(J(x)\) is poorly conditioned or even singular. In this section we investigate a generalization of (2.2); this will enable us to avoid this drawback in some cases:

\[(2.3a) \quad \dot{x}(t) = M(x)f(x), \quad t \geq 0 \]

\[(2.3b) \quad x(0) = x_0 . \]

If the preconditioner \(M(x)\) is non-singular, then any restpoint of (2.3) is a zero of \(f(x)\) and vice versa. Below we examine the conditions on \(M(x)\) that guarantee any zero \(x^*\) of (2.1) to be an asymptotically stable rest point of (2.3a).

In this paper \(|\cdot|\) will denote the Euclidian norm and the brackets \(<\cdot,\cdot>\) the underlying inner product. For any matrix \(A \in \mathbb{R}^{nxn}\) the logarithmic norm \(\mu : \mathbb{R}^{nxn} \to \mathbb{R}\) is defined by (cf. [11])

\[(2.4) \quad \mu[A] = \max_{\xi \neq 0} \frac{<A\xi,\xi>}{<\xi,\xi>} . \]

2.5 Property

Let \(f \in C^2(\mathbb{R}^n \to \mathbb{R}^{nxn})\) with \(M(x)\) nonsingular. Let \(r > 0\) be such that both \(f''(x)\) and \(M(x)\) are bounded on \(B(x^*,r)\) by \(C''\) and \(C_M\) resp. and

\[
\exists \alpha > 0 \forall x \in B(x^*,r) : \quad \mu[M(x)J(x)] \leq -\alpha .
\]

If \(0 < R < r\) is such that \(-\tau := -\alpha + RC''C_M < 0\), then

\[
\forall x_0 \in B(x^*,R) : \quad - (2.3) \text{ has a solution } x(t), \ t \in [0,\infty) \]

- \(x(t)\) remains in \(B(x^*,R)\)

- \(|x(t) - x^*| \leq \exp(-\tau t)|x_0 - x^*|

3
Proof
\[ \forall \epsilon \in B(x^*, r) : <x - x^*, M(x)f(x)> \]
\[ = <x - x^*, M(x)J(x)(x - x^*)> + <x - x^*, M(x)[f(x) - f(x^*) - J(x)(x - x^*)]> \]
\[ \leq -\alpha\|x - x^*\|^2 + C''C_M\|x - x^*\|^3 \]
\[ \leq (-\alpha + RC''C_M)\|x - x^*\|^2 \]

So if \( x(t) \in B(x^*, R) \), then
\[ \frac{d}{dt} \|x(t) - x^*\|^2 = 2 <x(t) - x^*, \dot{x}> \]
\[ = 2 <x - x^*, M(x)f(x)> \]
\[ \leq -\tau\|x(t) - x^*\|^2 < 0 \]

Now \( \|x(t) - x^*\| \) is decreasing, i.e. \( x(t) \) remains in \( B(x^*, R) \) and
\[ \|x(t) - x^*\| \leq \exp(-\tau t)\|x_0 - x^*\| \]

Once we have established that the solution \( x(t) \) converges to \( x^* \), a result from [5] yields that the distance \( \|x(t) - x^*\| \) satisfies the sharper bound

\[ (2.6) \quad \|x(t) - x^*\| \leq \exp(-\alpha t)\|x_0 - x^*\|. \]

Remarks
1. If \( J^{-1}(x) \) exists and is bounded on a neighbourhood of \( x^* \), then \( M(x) = -J^{-1}(x) \) satisfies the conditions of Property 2.5 with \( \alpha = 1 \).
2. Under the conditions of Property 2.5 the ODE (2.3) is asymptotically stable at \( x = x^* \).
3. If \( J(x) \) is singular, then for any preconditioner \( M(x) \) the matrix \( M(x)J(x) \) is singular, too. Since \( \mu[M(x)J(x)] \geq \max\{Re \lambda : \lambda \in \sigma(M(x)J(x))\} \), the best situation achievable is
\[ \mu[M(x)J(x)] < 0 \text{ if } J(x) \text{ is nonsingular} \]
\[ \mu[M(x)J(x)] = 0 \text{ if } J(x) \text{ is singular}. \]
4. The guaranteed convergence domain is invariant with respect to scaling of the preconditioner; for if \( M(x) \) is multiplied by a factor \( \beta \) say, both \( \mu[MJ] \) and \( \|M(x)\| \) are increased by that same factor and so the radius \( R \) is unchanged.

If one uses the preconditioner \( M(x) = -J^{-1}(x) \), property 2.5 gives a convergence domain of the time-stepping process which is approximately just as large as the convergence domain established by the affine invariant version of the Newton-Kantorovich theorem (cf. [7]). For the latter one can prove convergence for any starting value \( x_0 \) that satisfies
\[ (2.7) \quad \|J^{-1}(x_0)(f(x_0) - f(x^*))\| \leq \frac{1}{2\omega} \]

with \( \omega \) a constant such that on a neighbourhood \( D \) of \( x_0 \):

\[ (2.8) \quad \forall z, y \in D : \quad \|J^{-1}(x_0)(J(y) - J(z))\| \leq \omega \|y - z\| . \]

The equation (2.7) is qualitatively equal to

\[ (2.9) \quad \|x_0 - x^*\| \leq \frac{1}{2\omega} . \]

A closer look at the estimates used in the proof of property 2.5 yields that convergence of the time-stepping process is guaranteed for \(-\alpha + \omega R < 0\), i.e.

\[ (2.10) \quad \|x_0 - x^*\| \leq \frac{1}{\omega} . \]

This requirement can be very restrictive if \( x^* \) is close to a region where \( J(x) \) is singular.

For an arbitrary preconditioner \( M(x) \) the estimate in property 2.5 has a vital difference, viz. \( \|J^{-1}(x_0)\| \) is replaced by a bound on \( \|M(x)\| \). So one should look for a preconditioner \( M(x) \) which is reasonably bounded on a relevant area and is such that \( \mu[M(x)J(x)] \) is (preferably large) negative.

As for the Newton method, the proper behaviour of a solution \( x(t) \) can be monitored by an object function. A good choice might be \( \|M(x(t))f(x(t))\| \), as the following property reveals that \( \|Mf(x(t))\| \) decreases if the preconditioner \( M(x) \) is locally constant; this implies that \( \|f(x(t))\| \) is decreasing if \( \|M^{-1}(x)\| \) is bounded on a relevant domain.

2.11 Property

Let \( M \in \mathbb{R}^{n \times n} \). Let \( x(t) \) be a solution of the ODE

\[ \dot{x}(t) = Mf(x) \quad t \geq 0 \]
\[ x(0) = x_0 . \]

Then

\[ \forall t > 0 \quad \|Mf(x(t))\| \leq \exp(\int_0^t \mu[MJ(x(\sigma))]d\sigma)\|Mf(x_0)\| . \]

Proof

\[ \frac{d}{dt}\|Mf(x(t))\|^2 = 2 < MJ(x(t))\dot{x}, Mf(x(t)) > \]
\[ = 2 < MJ(x)Mf(x), Mf(x) > \]
\[ \leq \mu[MJ \ast x(t)] \|Mf(x(t))\|^2 . \]
Even if the initial guess is outside the convergence domain derived in property 2.5, solving the ODE (2.3) may still be worthwhile; for in practice this domain is often considerably larger, as some examples show.

2.12 Examples

1 Take \( f(x) = \frac{x^2}{1 + x^2} - \beta \), \( 0 < \beta < 1 \).

The Jacobian of \( f(x) \) is (almost) zero for \( |x| \approx 0 \) or \( |x| \) large. There are several possible preconditioners for this problem.

a. The classical choice \( M(x) = -J^{-1}(x) \).

This expression is undefined at \( x = 0 \), i.e. \( x = 0 \) is a singular point of the ODE. It is easily seen that for every nonzero starting vector the solution to (2.3) converges to a root of \( f(x) \).

b. \( M(x) = -x \).

This preconditioner is defined for every real \( x \). However an extra (unstable) node at \( x = 0 \) is created. For this preconditioner the solutions of (2.3) also converge to a root of \( f(x) \) for every \( x_0 \neq 0 \).

c. \( M(x) = -1 \).

Any solution with \( x_0 \in (0, \infty) \) converges to the positive root of \( f(t) \). The advantage of this choice is that \( M(x) \) is bounded.

For all three choices the solution \( x(t) \) converges to a zero of \( f(x) \) for any \( x_0 \) in a very large domain, although the convergence domains as predicted by lemma 2.5 are rather limited. This is shown in table 2.1, where the convergence domain for the positive root according to the theory is tabulated.

<table>
<thead>
<tr>
<th>( \beta )</th>
<th>( \beta^* )</th>
<th>Newton</th>
<th>( M = -J^{-1} )</th>
<th>( M = -x )</th>
<th>( M = -1 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>9.501e-3</td>
<td>0,1</td>
<td>[0.0708,0.251]</td>
<td>[0.0501,0.198]</td>
<td>[0.0619,0.136]</td>
<td>[0.0501,0.296]</td>
</tr>
<tr>
<td>0,0385</td>
<td>0,2</td>
<td>[0.142,0.370]</td>
<td>[0.101,0.385]</td>
<td>[0.125,0.269]</td>
<td>[0.101,0.570]</td>
</tr>
<tr>
<td>0,2</td>
<td>0,5</td>
<td>[0.367,0.659]</td>
<td>[0.267,0.798]</td>
<td>[0.326,0.627]</td>
<td>[0.267,0.798]</td>
</tr>
<tr>
<td>0,5</td>
<td>1</td>
<td>[0.847,1.106]</td>
<td>[0.750,1.201]</td>
<td>[0.767,1.201]</td>
<td>[0.500,1.201]</td>
</tr>
<tr>
<td>0,8</td>
<td>2</td>
<td>[1.957,2.038]</td>
<td>[1.920,2.074]</td>
<td>[1.840,2.074]</td>
<td>[1.840,2.074]</td>
</tr>
<tr>
<td>0,9</td>
<td>3</td>
<td>[2.985,3.015]</td>
<td>[2.970,3.029]</td>
<td>[2.940,3.029]</td>
<td>[2.940,3.029]</td>
</tr>
</tbody>
</table>

Table 2.1
In case $\beta \geq 0.5$ the derivative $f'(x)$ is almost zero around $x = x^*$. The convergence domain with any of the three preconditioners mentioned is about twice as large as the convergence domain of Newton’s method.

2 A familiar example (cf. [1], [4]) is Rosenbrock’s function

$$f_1 = 4 \cdot a \cdot x_1(x_1^2 - x_2) + 2 \cdot (x_1 - b)$$

$$f_2 = -2 \cdot a \cdot (x_1^2 - x_2)$$

$a, b > 0$.

The Jacobian $J(x)$ is singular on the parabola $x_2 = x_1^2 + 1/2a$ and any method that follows the descent directions of $f(x)$ will stay close to the neighbouring parabola $x_2 = x_1^2$. In [4] the trajectories of the ODE (2.2) are plotted for $a = 0.5$ and $b = 1$. This shows that any solution starting under the singularity-parabola converges to the solution $x^* = (1, 1)$. □
§3 The integration method

The reason for solving the ODE

\begin{align}
(3.1a) & \quad \dot{x}(t) = M(x)f(x) \quad t \geq 0 \\
(3.1b) & \quad x(0) = x_0
\end{align}

is to obtain a zero $x^*$ of $f(x)$.

When using explicit integration methods, numerical stability considerations invariably lead to stepsize restrictions. For our purposes this can be a disadvantage, since the solution $x(t)$ approximates the restpoint $x^*$ better for larger $t$.

However, not all implicit methods necessarily have profitable stability properties for larger stepsizes. The trapezoidal scheme as used in [3] does not yield ultimate fast convergence for large stepsizes since then $x_{i+1} - x^* \approx x^* - x_i$. So we are interested in a method that allows large stepsizes as $x(t)$ approaches the restpoint 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.

\begin{equation}
(3.2) \quad x_{i+1} = x_i + h_i M(x_{i+1})f(x_{i+1})
\end{equation}

However using an iterative scheme to solve (3.2) 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.

\begin{equation}
(3.3) \quad x_{i+1} = x_i + h_i M(x_i)f(x_{i+1})
\end{equation}

Solving this equation requires essentially less work and, as we prove later on, still inhibits the stability properties of an implicit Euler method. It is clear that for $h_i$ small enough a vector $x_{i+1}$ can be found from (3.3). If $f(x)$ and $M(x)$ are sufficiently smooth, the existence requirement does not force $h_i$ to tend to zero.

Throughout the paper we will consider only preconditioners $M(x)$ that satisfy the conditions of prop. 2.5. This implies that there is a sphere $D$ with center $x^*$ such that

\begin{equation}
(3.4) \quad \forall x \in D : \langle x - x^*, M(x) \cdot (f(x) - f(x^*)) \rangle \leq -\alpha \cdot \|x - x^*\|^2.
\end{equation}

We now show that the mixed Euler method is consistent of order 1, which then implies convergence.

3.5 Lemma

The discretization error $\delta(x_i, h_i)$ of the mixed Euler method defined by

\begin{align}
(3.5a) & \quad \delta(x_i, h_i) := h_i^{-1} \cdot \{x(t_{i+1}) - x(t_i) - h_i M(x(t_i))f(x(t_{i+1}))\}
\end{align}

is bounded, as follows
with \( \nu_1, \nu_2, \nu_3 \) in the convex hull of \( x(t_i) \) and \( x(t_{i+1}) \).

**Proof**

The estimate follows immediately from the relation

\[
\delta(x_i, h_i) = h_i^{-1} \cdot \{ x(t_{i+1}) - x(t_i) - h_i M(x(t_i)) f(x(t_{i+1})) \} \\
+ M(x(t_i)) \cdot \{ f(x(t_i)) - f(x(t_{i+1})) \} 
\]

3.6 **Remark**

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

\[
\delta(x_i, h_i) \leq 0.5 \cdot h_i \cdot \| \dot{\nu}_1 \| + h_i \cdot \| \dot{\nu}_2 \| + O(h_i^2) . \tag{3.6} \]

If \( x_i \) and \( x(t_i) \) are close to \( x^* \), the solution \( x(t) \) will be almost constant. Consequently stepsizes based on controlling the discretization errors will give rapidly increasing values for \( h_i \). These large values for \( h_i \) do not endanger the existence of the next iterate (see §5).

We next investigate when the sequence \( \{x_i\} \) (produced by the mixed Euler method) converges and when the limit is a zero of \( f(x) \). For this we consider the linear interpolant of \( (t_i, x_i) \) (with \( t_0 = 0 \) and \( t_{i+1} = t_i + h_i \)), viz.

\[
v(t) := \left[ (t_{i+1} - t)x_i + (t - t_i)x_{i+1} \right] / h_i \quad t \in [t_i, t_{i+1}) . \tag{3.7} \]

Now \( v(t) \) satisfies an ODE which is just a perturbation of (3.1). Hence we have the following lemma about perturbed ODE's to start with.

3.8 **Lemma**

Consider the ODE

\[
\dot{z} = A(t)z + g(z) + \varepsilon(t) \tag{3.8a} \\
z(0) = 0 \tag{3.8b}
\]

with \( A(t) \in C^1([0, \infty) \to \mathbb{R}^{n \times n}), g(z) \in C(\mathbb{R}^n \to \mathbb{R}^n), \varepsilon(t) \in C([0, \infty) \to \mathbb{R}^n) \) and

\[
\exists C > 0 \ \forall z \in \mathbb{R}^n \quad \| g(z) \| \leq C \| z \| 
\]

and

\[
\| \varepsilon(t) \| \to 0, t \to \infty .
\]

Let \( K, \alpha > 0 \) be constants such that every fundamental solution \( Z(t) \) of the homogeneous ODE

\[
|Z(t)|_K, \alpha \leq \frac{C \| \varepsilon(t) \|}{K \alpha} .
\]
satisfies

\begin{equation}
\forall 0 \leq \tau \leq t : \|Z(t)Z^{-1}(\tau)\| \leq K \exp(-\alpha(t - \tau)) .
\end{equation}

Then the solution \( z(t) \) of (3.8a,b) is bounded and \( \lim_{t \to \infty} \|z(t)\| = 0 \). And there is a constant \( c \) such that

\begin{equation}
\forall t > 0 : \|z(t)\| \leq c \cdot \max\{\|\epsilon(\tau)\| : \tau \leq t\} .
\end{equation}

Condition (3.8d) is equivalent with uniform asymptotical stability of ODE (3.8c). This can also be formulated in terms of the logarithmical norm, viz.

\begin{equation}
\forall \tau \geq 0 : \mu[A(t)] \leq -\alpha .
\end{equation}

### 3.10 Theorem

Consider the ODE (3.1) with \( f \in C^2(\mathbb{R}^n \to \mathbb{R}^n), M \in C^1(\mathbb{R}^n \to \mathbb{R}^{n \times n}) \). Let \( x^* \) be a zero of \( f(x) \) and let \( T > 0 \). Consider the class of sequence-pairs \( \{h_i\}, \{x_i\} \) satisfying (3.3) with \( x_i \to x^* \).

a) For every sequence let there be an index \( j \in \mathbb{N} \) with \( T = \sum_{i=0}^{j-1} h_i \), then

\[ \|x(T) - x_j\| \to 0, \text{ if } \max(h_i) \to 0 , \]

i.e. convergence of mixed Euler.

b) Let \( x(t) \) denote the solution of (3.1), then \( x(t) \to x^*, t \to \infty \).

**Proof**

Let \( v(t) \) be as defined in (3.7). Then \( v(t) \) is differentiable outside the interpolation points and satisfies the ODE

\[ \dot{v} = M(v)f(v) + \epsilon(t) \quad t \geq 0 \]

\[ v(0) = x_0 \]

where \( \epsilon(t) \) is defined by

\[ \epsilon(t) = M(x_i)f(x_{i+1}) - M(v(t))f(v(t)), \quad t \in [t_i, t_{i+1}) . \]

Because \( M(x) \) and \( f(x) \) are Lipschitz continuous, there is a constant \( C_\epsilon \) such that \( \epsilon(t) \) is bounded by \( C_\epsilon\|x_i - x_{i+1}\| \) on \( [t_i, t_{i+1}) \). So \( \epsilon(t) = O(h_i) \) and \( \lim_{i \to \infty} \epsilon(t) = 0 \).

Let \( x(t) \) be the solution of the ODE (3.1) and define \( \xi(t) := x(t) - v(t) \).

Then
\[
\begin{align*}
\xi(0) &= 0 \\
\forall t > 0 \quad \dot{\xi}(t) &= M(x)f(x) - M(v)f(v) - \varepsilon(t) \\
&= M(x)j(x)\xi(t) - \varepsilon(t) \\
&\quad + M(x)[f(x) - f(v) - J(x)(x - v)] + [M(x) - M(v)]f(v).
\end{align*}
\]

Now define the last line of this ODE as \(g(\xi)\), then it satisfies the conditions of lemma 3.8. The proof now follows from the relation \(x(t_i) - v(t_i) = x(t_i) - x_i\).

Now that both consistency and usefulness of the mixed Euler method have been established, we would like to present an alternative, theoretical view to it. After sequences \(\{h_i\}, \{x_i\}\) have been obtained by the mixed Euler method, an ODE can be constructed such that Euler backward applied to it with stepsizes \(\{h_i\}\), yields the same sequences \(\{x_i\}\). To this end consider the ODE

\[
\begin{align*}
(3.11a) \quad \dot{y} &= L(t)f(y) \\
(3.11b) \quad y(0) &= x_0
\end{align*}
\]

where \(L \in C^1([0, \infty) \to \mathbb{R}^{n \times n})\) satisfies

\[
(3.12) \quad \forall i : \quad L(t_{i+1}) = M(x_i).
\]

The sequence \(\{x_i\}\) can be regarded as the result of the implicit Euler method applied to (3.11) with stepsizes \(\{h_i\}\), viz.

\[
x_{i+1} = x_i + h_iL(t_{i+1})f(x_{i+1}) = x_i + h_iM(x_i)f(x_{i+1}).
\]

We show that for any function \(L(t)\) satisfying (3.12), the difference between the solution \(y(t)\) of the "artificial" ODE (3.11) and the solution \(x(t)\) of the original ODE (3.1) is of order \(h_i\). Moreover the limitpoint \(x^*\) of \(\{x_i\}\) is a steady state of both \(y(t)\) and \(x(t)\).

3.13 Theorem
Let \(\{h_i\}, \{x_i\}\) satisfy (3.3) with \(x_i \to x^*\). Let \(L(t) \in C^1([0, \infty) \to \mathbb{R}^{n \times n})\) satisfy (3.12). If \(x(t), y(t)\) are solutions of (3.1) and (3.11) resp., then

\[
\exists C > 0 \forall i \forall t \in [t_i, t_{i+1}] : \quad \|x(t) - y(t)\| \leq Ch_i
\]

and \(\lim_{t \to \infty} \|y(t) - x^*\| = 0\).

Proof
The difference \(\eta(t) := x(t) - y(t)\) satisfies the ODE
\( \eta(0) = 0 \)

and

\[
\dot{\eta}(t) = M(x)f(x) - L(t)f(y)
= M(x)J(x)\eta
+ M(x)[f(x) - f(y) - J(x)(x - y)] + [M(x) - M(y)]f(y)
+ [M(y) - L(t)]f(y).
\]

Application of lemma 3.8 with \( g(\eta) \) the second and \( \varepsilon(t) \) the third line completes the proof. \( \Box \)
§4 Convergence of the iterative process

In the previous section we looked at the properties of the mixed Euler method as an ODE-solver. Now we investigate its behaviour as a "root-finder". As in section 3 let $D$ be the ball around $x^*$ such that

$$(4.1) \quad \exists \alpha > 0 \quad \forall x \in D : \quad < x - x^*, M(x)(f(x) - f(x^*)) > \leq -\alpha \|x - x^*\|^2.$$  

If $x_i$ and $x_{i+1}$ are both in $D$ and $h_i$ is sufficiently small, then $x_{i+1}$ is in the small sphere shown in figure 4.1. The larger dotted sphere shows the bound that is usually derived in these kind of situations.

![Figure 4.1](image)

4.2 Theorem (convergence of the iterative process to $x^*$)

Let $C_J$ be an upperbound for $\|J(x)\|$ on the sphere $D$ and let $L_M$ denote the Lipschitz constant of $M(x)$ on $D$. Let $x_i$ be in $D$ and $h_i$ be sufficiently small to guarantee that $x_{i+1}$ is in $D$ and the constant $b$ defined by

$$(4.2) \quad b := 1 + h_i (a - C_J L_M \|x_i - x_{i+1}\|)$$

is larger than 1. Then the vector $x_{i+1}$ satisfying (3.3) is in the sphere with center

$$\left(1 - \frac{1}{2b}\right)x^* + \frac{1}{2b}x_i$$

and radius $\frac{\|x_i - x^*\|}{2b}$.

Proof

Define $e_i := x_i - x^*$. From (3.4) and (3.5) we get

$$< e_{i+1}, e_{i+1} > = < e_i, e_{i+1} > + h_i < e_{i+1}, M(x_i)f(x_{i+1}) >$$

$$= < e_i, e_{i+1} > + h_i < e_{i+1}, M(x_{i+1})(f(x_{i+1}) - f(x^*)) >$$

$$+ h_i < e_{i+1}, (M(x_i) - M(x_{i+1}))(f(x_{i+1}) - f(x^*)) >$$

$$\leq < e_i, e_{i+1} > + \alpha h_i \|e_{i+1}\|^2 + h_i L_M C_J \|x_i - x_{i+1}\| \|e_{i+1}\|^2.$$  

13
So

\[ (1) \quad b\|e_{i+1}\|^2 < \epsilon_i, e_{i+1} >. \]

This implies of course that \( \|e_{i+1}\| \leq \frac{1}{b}\|e_i\| \), but we can get a sharper result:

\[
\|x_{i+1} - \left(1 - \frac{1}{2b}\right)x^* - \frac{1}{2b}x_i\|^2 = \|e_{i+1}\|^2 - 2\frac{1}{2b} < \epsilon_{i+1}, \epsilon_i > + \left(\frac{1}{2b}\right)^2 \|e_i\|^2
\]

\[
\leq \frac{1}{b} < \epsilon_i, \epsilon_{i+1} > - 2\frac{1}{2b} < \epsilon_{i+1}, \epsilon_i > + \left(\frac{1}{2b}\right)^2 \|e_i\|^2
\]

\[
= \left(\frac{1}{2b}\right)^2 \|x_i - x^*\|^2 .
\]

**Corollary 4.3**

This theorem implies of course that \( \|x_{i+1} - x^*\| \leq b^{-1}\|x_i - x^*\| \).

Once \( x_i \) is in \( D \) this theorem can be applied, since a suitable choice of the variable \( h_i \) can guarantee that

(i) \( x_{i+1} \) is in \( D \)

(ii) \( b \geq 1 \).

As soon as \( \|x_i - x^*\| < \alpha/C_JL_M \), the constant \( b \) is larger than 1 independent of \( h_i \); i.e. the restrictions on \( h_i \) are lifted. In that case it is favourable to choose \( h_i \) large since that yields superlinear convergence, viz.

\[
\lim_{i \to \infty} \frac{\|x_{i+1} - x^*\|}{\|x_i - x^*\|} = \lim_{i \to \infty} \frac{1}{1 + h_i(\alpha - C_JL_M\|x_i - x_{i-1}\|)} = 0, \quad \text{if} \quad \lim_{i \to \infty} h_i = \infty .
\]

**4.4 Example**

a Consider Rosenbrock’s function (see Ex.2.12) with \( a = 100, b = 1 \). We have applied the mixed Euler method to it with \( M(x) = -J^{-1}(x) \) and a stepsize control based on \( \|h\delta\| < 0.1 + 0.1 \cdot \|x_i\| \). In table 4.1a the last steps of the process are tabulated, where

\[
b^{-1} := \frac{\|x_i - x^*\|}{\|x_{i-1} - x^*\|} \quad \text{and} \quad \tau \text{ such that } b = 1 + \tau h .
\]

Table 4.1a clearly shows that \( h \) increases rapidly near the end of the process. Since the asymptotic convergence factor \( b^{-1} \) converges to 0, the process has final superlinear convergence. The values of \( \tau \) approximately converge to 1; this duly coincides with the theoretical result of theorem 4.2 that \( \lim \tau = -\mu[M(x^*)J^{-1}(x^*)] = 1 \).

b Next we applied the mixed Euler method to

\[
f(x) = \frac{x^2}{1 + x^2} - 0.1
\]
with the same stepsize control and output specifications, but with preconditioner \( M(x) = -x \). The results, presented in table 4.1b, again demonstrate the increase in the stepsize and the rapid final convergence. The theoretical limit for \( \tau \):

\[
\lim_{\tau} \tau = -\mu \left[ \frac{-2x^2}{(1 + x^2)^2} \right] = 0.18 \quad \text{for} \quad x = x^* = \frac{1}{3}
\]

is confirmed by the numerical results.

| \( h \) | \( ||z_i - z^*|| \) | \( b^{-1} \) | \( \tau \) | \( h \) | \( ||z_i - z^*|| \) | \( b^{-1} \) | \( \tau \) |
|---|---|---|---|---|---|---|---|
| 3.2e0 | 2.89e-1 | | | 2.2e0 | 4.7e-2 | | |
| 3.2e0 | 1.66e-1 | 5.75e-1 | 0.23 | 2.2e0 | 3.2e-2 | 0.69 | 0.20 |
| 3.2e0 | 7.21e-2 | 4.33e-1 | 0.41 | 4.4e0 | 1.7e-2 | 0.54 | 0.20 |
| 6.3e0 | 1.58e-2 | 2.20e-1 | 0.56 | 4.4e0 | 9.5e-3 | 0.55 | 0.19 |
| 6.3e0 | 3.00e-3 | 1.90e-1 | 0.68 | 1.2e1 | 3.0e-3 | 0.31 | 0.18 |
| 2.1e1 | 1.76e-4 | 5.92e-2 | 0.75 | 2.9e1 | 4.8e-4 | 0.16 | 0.18 |
| 1.5e2 | 1.36e-6 | 7.65e-3 | 0.87 | 1.0e2 | 2.4e-5 | 5.0e-2 | 0.18 |
| 4.3e3 | 3.25e-10 | 2.40e-4 | 0.97 | | | | |

Table 4.1
§5 Implementation

In the previous sections we addressed convergence and justification 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 $x_{i+1}$

(iii) stepsize 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 a subsequent paper we will derive a preconditioner for the nonlinear equations arising from the multiple shooting method applied to a class of nonlinear boundary value problems.

To obtain the next iterate $x_{i+1}$ from formula (3.3), we have to solve the non-linear equation

$$g_i(x) = 0$$

with

$$g_i(x) = h_i^{-1}(x - x_i) - M(x_i)f(x) .$$

Convergence of the Newton method with starting point $x_i$ can be influenced by the choice of the stepsize $h_i$. To see this recall the affine invariant version of the Newton-Kantorovich theorem (cf. [7]).

Let $g \in C^2(D_0 \to \mathbb{R}^n), D_0 \subseteq \mathbb{R}^n; g'(y_0)$ be invertible an let there be constants $\alpha, \beta$ such that

(i) $\|g'(y_0)^{-1}g(y_0)\| \leq \alpha$

(ii) $\forall x, y \in D_0 : \|g'(y_0)^{-1}(g'(x) - g'(y))\| \leq \beta \|x - y\|$

(iii) $\alpha \beta \leq 0.5$

(iv) $B(y_0, r) \subseteq D_0, r = \frac{1 - \sqrt{1 - 2\alpha \beta}}{\beta}$.

Then (5.1) has a solution and the Newton sequence starting in $y_0$ converges to it.

In this case $g'(x_i) = h_i^{-1}I - M(x_i)J(x_i)$. And from the definition of logarithmic norm

$$\forall y \in \mathbb{R}^n : < y, M(x_i)J(x_i)y > \leq \mu[M(x_i)J(x_i)]\|y\|^2$$

it follows that
\( \forall y \in \mathbb{R}^n : < y, [h_i^{-1} I - M(x_i)J(x_i)] y > \geq (h_i^{-1} - \mu[M(x_i)J(x_i)])\|y\|^2. \)

So if \( \mu[M(x_i)J(x_i)] < 0 \), then \( g'(x_i) \) is invertible for every positive stepsize \( h_i \) and

\[
(5.3) \quad \|g'(x_i)^{-1}\| \leq \frac{h_i}{1 - \mu h_i}.
\]

The conditions (i), (ii) and (iii) combine to

\[
(5.4) \quad \|M(x_i)\|^2 L \|f(x_i)\| \frac{h_i^2}{(1 - \mu h_i)^2} \leq 0.5
\]

with \( L \) the Lipschitz of \( f'(x) \). If \( M = -J^{-1}(x) \), the first three factors on the left hand side are equal to Newton-Kantorovich requirements for the original problem \( f(x) = 0 \).

Formula (5.4) shows that it is always possible to choose \( h_i \) small enough to guarantee convergence.

This disproves an often used argument to reject implicit integration methods for Davidenko's equation, viz. it would require at each step solving a nonlinear equation, which is, wrongly, considered to be the original problem.

Notice that once \( x_i \) is close to \( x^* \), the term \( \|f(x_i)\| \) will approach zero and the stepsize can be chosen correspondingly larger.

If we use \( -J^{-1}(x) \) as a preconditioner, the first Newton step reads

\[
(5.5) \quad y_1 = x_i - g'(x_i)^{-1}g(x_i) = x_i - \frac{h_i}{1 + h_i}J^{-1}(x_i)f(x_i)
\]

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 (5.1). So \( y_1 \) is not the next iterate, but only an intermediate result. Secondly we base our choice of the damping factor on controlling the discretization error and not on iterative adapting the damping factor until the value of some object function decreases. However, once the iterates \( x_i \) approach \( x^* \) the first Newton iterate on \( g(x) \) is accepted as \( x_{i+1} \). At the same time \( g_i \) tends to infinity, so the implementation of the mixed Euler method converges 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.

\[
(5.6) \quad y_0 = x_i
\]

\[
y_{j+1} = y_j - g'(x_i)^{-1}g(y_j), \quad j \geq 0.
\]

Now only one evaluation of the preconditioner \( M(x) \) per time step is necessary. The convergence of this process is controlled by monitoring \( \|M_i f_i\| \). At the same time we have to following the path \( x(t) \). If the latter is computed with given absolute and relative tolerances ATOL and RTOL resp., then we may take the tolerance on the Newton process (NTOL) equal to
\[ \text{NTOL} = \text{ATOL} + \text{RTOL} \| x(t_i) \|. \]

The choice of the stepsize \( h_i \) influences several aspects of the process. One of them is controlling the local error. The latter is estimated by

\[ \text{EST} := 0.5 h_i^2 \| \frac{x_i - x_{i-1}}{h_i} - \frac{x_{i-1} - x_{i-2}}{h_{i-2}} \| \frac{2}{h_{i-1} + h_{i-2}} \]

(the use of the Euler backward discretization error for this process is based on Th.3.13).

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

Now the stepsize \( h_i \) is determined at every step as follows:

- take \( h_i \) equal to \( h_{i-1} \).
- double the stepsize if it has not been changed in three consecutive steps, to prevent conservatism.
- if the Newton process has not converged in 5 iterations, halve the step size until convergence is reached. (If \( h_i \) has not been changed we do not expect nonconvergence, unless the path has entered a troublesome area.).
- at every step compute the quantity \( \text{TEST} := \text{EST} / (\text{ATOL} + \| x_i \| \cdot \text{RTOL}) \).

If \( \text{TEST} \in [0.25, 4] \) the step is accepted.
If \( \text{TEST} > 4 \) the discretization error is too large and \( x_i \) is recalculated for the stepsize

\[ h_{\text{new}} = h_{\text{old}} / \sqrt{\text{TEST}} \]

If \( \text{TEST} < 0.25 \) the path is followed "too accurately". Now accept \( x_i \) and increase \( h_i \) according to formula (5.8) or choose a more conservative formula that prevents sharp increases of the stepsize when far from the solution.

Remark
We have used the algorithm outlined above to test the ideas presented in this paper. But we are well-aware of the fact that various refinements and modifications can improve the efficiency of it. However, the results obtained with this program, as presented in Example 5.2, already indicate a relatively good performance.

Example 5.1
We have tested the algorithm described above on the problems considered in [1] with \( M(x) = -J^{-1}(x) \). As a measure for the amount of work we used the number of function calls \( (#f) \) plus \( n \) (=dimension of the system) times the number of Jacobian evaluations \( (#J) \).

For the conservative stepsize update in case \( \text{TEST} < 0.25 \) we used

\[ h_{\text{new}} = h_{\text{old}} \cdot \min \left( 1/\sqrt{\text{TEST}}, \max(2, -10 \log \| M_i f_i \|) \right). \]
If the path ran close to an area where $J(x)$ is singular, the careful stepsize update (5.9) works considerably cheaper than (5.8). This occurred especially at Rosenbrock's function (see example 2.12) where we spend a lot of function calls on discovering that the stepsize is too large. In the other easier cases the less conservative method (5.8) required less work (see table 5.1).

The results also showed that the condition number of $J(x)$ is not always a good indication for entering a problem area. For the Rosenbrock function the stepsize sometimes had to be reduced to yield convergence of the Newton process, when the condition number of $J$ was less than 10. On the other hand close to the solution the condition number rose to 2000, where $h_i$ reached values of about 200 without problems. At the second problem we had to reduce the tolerances of the mixed Euler method. The path gets very close to a line where the Jacobian is singular, although this was not visible in the values of the condition number.

Results for the problems from [1] with RTOL=ATOL=1e-1

<table>
<thead>
<tr>
<th>Problem no.</th>
<th>#f</th>
<th>#J</th>
<th>#f + n#J</th>
<th>#f</th>
<th>#J</th>
<th>#f + n#J</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>16</td>
<td>14</td>
<td>44</td>
<td>16</td>
<td>12</td>
<td>40</td>
</tr>
<tr>
<td>2(*)</td>
<td>45</td>
<td>23</td>
<td>91</td>
<td>42</td>
<td>20</td>
<td>82</td>
</tr>
<tr>
<td>3</td>
<td>11</td>
<td>9</td>
<td>29</td>
<td>9</td>
<td>6</td>
<td>21</td>
</tr>
<tr>
<td>4</td>
<td>128</td>
<td>32</td>
<td>192</td>
<td>219</td>
<td>30</td>
<td>279</td>
</tr>
<tr>
<td>5</td>
<td>35</td>
<td>17</td>
<td>86</td>
<td>28</td>
<td>13</td>
<td>67</td>
</tr>
<tr>
<td>6</td>
<td>23</td>
<td>13</td>
<td>101</td>
<td>20</td>
<td>10</td>
<td>80</td>
</tr>
<tr>
<td>7</td>
<td>24</td>
<td>14</td>
<td>66</td>
<td>24</td>
<td>13</td>
<td>63</td>
</tr>
<tr>
<td>8</td>
<td>33</td>
<td>16</td>
<td>81</td>
<td>32</td>
<td>14</td>
<td>74</td>
</tr>
</tbody>
</table>

(*) RTOL=ATOL=5e-2

Table 5.1

Example 5.2

We compared the results of the mixed Euler method with the results from other integration methods as presented in [1]. Table 5.2 gives the amount of work measured by #f + n#J.

<table>
<thead>
<tr>
<th>#f + n#J for the test problems of [1]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
</tr>
<tr>
<td>---</td>
</tr>
<tr>
<td>MEp</td>
</tr>
<tr>
<td>MEc</td>
</tr>
<tr>
<td>RK3</td>
</tr>
<tr>
<td>AB3</td>
</tr>
<tr>
<td>PECE</td>
</tr>
</tbody>
</table>

MEp=Mixed Euler with progressive stepsize choice
MEc=Mixed Euler with conservative stepsize choice
RK3=third order Runge Kutta
AB3=Adams-Bashforth variable step method order 3
PECE=Trapezoidal rule as described by Boggs

Table 5.2
This show that the mixed Euler method is generally performing better than explicit integrators and the trapezoidal rule.

**Example 5.3**

Timestep 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

\[ x_{i+1} = x_i - \lambda_i J^{-1}(x_i)f(x_i) \]

to the functions used in the two previous examples, where the damping factor \( \lambda_i \) is chosen according to

- \( \lambda_i = \min(2 \cdot \lambda_{i-1}, 1) \),
- if the object function does not decrease \( \lambda_i \) is halved until it does or \( \lambda_i < 10^{-3} \).

Table 5.3 shows the number of iterations necessary to reach convergence (i.e. \( \|J^{-1}(x_i)f(x_i)\| \leq 10^{-6} \)) for three object functions.

For the most reliable (and expensive) choice \( \|J^{-1}(x_{\text{new}})f(x_{\text{new}})\| \) damped Newton failed in 3 cases. Furthermore, damped Newton’s method with either object function could not solve the second problem, whereas the mixed Euler method reaches the solution in 20 steps.

<table>
<thead>
<tr>
<th>( f(x_{\text{new}}) )</th>
<th>( J^{-1}(x_{\text{old}})f(x_{\text{new}}) )</th>
<th>( J^{-1}(x_{\text{new}})f(x_{\text{new}}) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>8</td>
<td>6</td>
</tr>
<tr>
<td>2</td>
<td>( \lambda &lt; 10^{-3} )</td>
<td>( \lambda &lt; 10^{-3} )</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>( \geq 50 )</td>
<td>23</td>
</tr>
<tr>
<td>5</td>
<td>2</td>
<td>6</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>7</td>
<td>7</td>
<td>7</td>
</tr>
<tr>
<td>8</td>
<td>8</td>
<td>8</td>
</tr>
</tbody>
</table>

**Table 5.3**

**Conclusion**

We have seen that the mixed Euler method has stability properties similar to those of the implicit Euler backward method. The price for this is a restriction on the stepsize if we are far from the steady state. On the other hand every timestep 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 stepsizes 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 converges to the full Newton’s method.
References


### PREVIOUS PUBLICATIONS IN THIS SERIES:

<table>
<thead>
<tr>
<th>Number</th>
<th>Author(s)</th>
<th>Title</th>
<th>Month</th>
</tr>
</thead>
<tbody>
<tr>
<td>90-04</td>
<td>F.J.L. Martens</td>
<td>A representation of $GL(q, R)$ in $L_2(S^{q-1})$</td>
<td>April '90</td>
</tr>
<tr>
<td>90-05</td>
<td>J. de Graaf</td>
<td>Skew-Hermitean representations of Lie algebras of vectorfields on the unit-sphere</td>
<td>April '90</td>
</tr>
<tr>
<td>90-06</td>
<td>Y. Shindo, K. Horiguchi, A.A.F. van de Ven</td>
<td>Bending of a magnetically saturated plate with a crack in a uniform magnetic field</td>
<td>April '90</td>
</tr>
<tr>
<td>90-07</td>
<td>M. Kuipers, A.A.F. van de Ven</td>
<td>Unilateral contact of a springboard and a fulcrum</td>
<td>July '90</td>
</tr>
<tr>
<td>90-08</td>
<td>P.H. van Lieshout, A.A.F. van de Ven</td>
<td>A variational approach to the magnetoelastic buckling problem of an arbitrary number of superconducting beams</td>
<td>July '90</td>
</tr>
<tr>
<td>90-09</td>
<td>A. Reusken</td>
<td>A multigrid method for mixed finite element discretizations of current continuity equations</td>
<td>August '90</td>
</tr>
<tr>
<td>90-10</td>
<td>G.A.L. van de Vorst, R.M.M. Mattheij, H.K. Kuiken</td>
<td>A boundary element solution for 2-dimensional viscous sintering</td>
<td>October '90</td>
</tr>
<tr>
<td>90-11</td>
<td>A.A.F. van de Ven</td>
<td>A note on 'A nonequilibrium theory of thermoelastic superconductors' by S-A. Zhou and K. Miya</td>
<td>October '90</td>
</tr>
<tr>
<td>90-12</td>
<td>M.E. Kramer, R.M.M. Mattheij</td>
<td>Timestepping for solving nonlinear equations</td>
<td>October '90</td>
</tr>
</tbody>
</table>