Yokohama Publishers
ISSN 1349-8169 ONLINE JOURNAL

Pacific Journal of Optimization
Vol. 9, No. 1, pp. 77–91, January 2013

EXPLICIT PSEUDO-TRANSIENT CONTINUATION*

C. T. Kelley and Li-Zhi Liao

Abstract: We analyze and extend an explicit pseudo-transient continuation algorithm proposed by Han and Han. We present a convergence proof, extend the method to bound-constrained optimization, and proposed an improved step-size control strategy.

Key words: *psuedo-transient continuation, explicit method, gradient flow*

Mathematics Subject Classification: *65H10, 65K10, 65L05*

1 Introduction

The purpose of this paper is to analyze and extend an explicit version of pseudo-transient continuation (ΨTC) which has been proposed in [6] for nonlinear equations and in [7] for unconstrained optimization. This method is appealing, as are all explicit methods, because it does not require solutions of linear systems. The disadvantage, of course, is that more iterations will be needed because the fast local convergence of Newton's method will be lost. In this introductory section we explain ΨTC , review some existing methods and describe the simplest version of the method from [6]. We follow that with the analysis and extensions, and finish the paper with a numerical example.

The objective of the method of pseudo-transient continuation [11, 4, 12] is to integrate an initial value problem

$$u' = -F(u); u(0) = u_0 \tag{1.1}$$

to steady state. Here u and F are valued in R^N and F is Lipschitz continuous. By integration to steady state we mean computation of

$$u^* = \lim_{t\to\infty} u(t),$$

if it exists, as we will assume it does.

Clearly

$$F(u^*) = 0 \tag{1.2}$$

and, in principle, a Newton-like method [5, 17, 8, 10] could be used. However ΨTC exploits the dynamics in (1.1) to select the unique solution of (1.2) which is the limit of the dynamics. This property is very useful in preserving accurate physics [14, 13].

*This work has been partially supported by National Science Foundation Grant CDI-0941253, Army Research Office Grants W911NF-11-1-0367, and W911NF-07-1-0112, USACE contracts W912HZ-10-P-0221, and W912HZ-11-P-0385, and GRF grants HKBU201409 and HKBU201611 from the Research Grant Council of Hong Kong.

ISSN 1348-9151

Most variants of the ΨTC in the literature are implicit and use the iteration

$$u_{n+1} = u_n - (\delta_n^{-1} I + F'(u_n))^{-1} F(u_n). \tag{1.3}$$

ΨTC algorithms which are based on (1.3) attempt to increase δ_n as the iteration progresses and still maintain stability when far from the solution. Ideally the formula should evolve towards Newton's method ($\delta_n \equiv \infty$) in the terminal phase of the iteration.

The various ΨTC algorithms which use (1.3) differ in their strategies for managing the sequence $\{\delta_n\}$. Several of these methods have been analyzed in [11, 4, 12]. The proofs of convergence share three common features:

1. If δ_0 is sufficiently small, the iteration will be close to an explicit Euler integration, and therefore be able to approach u^* arbitrarily closely. In particular, u_n will eventually lie in the domain of attraction for Newton's method.

2. Once near u^* the time step control will allow δ_n to safely grow while keeping u in the domain of attraction for Newton's method.

3. Once δ_n is sufficiently large, the local convergence theory for Newton-like methods will take over, and the convergence in the terminal phase will be fast.

Other approaches, which replace Euler's method with a higher-order method have also been considered [18, 15].

Computing u_{n+1} with (1.3) requires the solution of the linear system

$$(\delta_n^{-1} I + F'(u_n))d = -F(u_n),$$

and hence we refer to it as an implicit method. The method of [6] does not require solution of linear equations, and is in that sense explicit.

We continue this paper in § 1.1 by describing the method from [6]. In § 2 we give a convergence proof, extend the method to the case of optimization problems with simple bound constraints, and propose a new stepsize management scheme. In § 3 we illustrate the results with a bound-constrained parameter identification problem [9, 2, 12].

1.1 Explicit ΨTC

The simplest version of the algorithm fixes δ for the entire iteration. ϵ, a tolerance τ, and an initial iterate u_0 are given. We define

$$\omega = \frac{\delta}{\delta + \epsilon}.$$

We will discuss changing δ as the iteration progresses later in § 2.4.

The algorithm is expressed in terms of u_n and u_n^0 because the analysis uses the error in u_n, while the termination criterion uses $F(u_n^0)$.

We have elected to return the final value of u_n^0 as the result because the residual has been evaluated there. If the method terminates successfully, u_{n+1}^0 will be the converged result. There is only one function evaluation at each step.

Algorithm 1 The Method from [6]

$n = 0$; $z_0 = \delta F(u_0)$;
$u_1^0 = u_0 - z_0$;
Compute $F(u_1^0)$ and $\rho = \|F(u_1^0)\|$
while $\rho > \tau$ **do**
  $z_{n+1} = \omega(\epsilon F(u_{n+1}^0) + z_n)$
  $u_{n+1} = u_n - z_{n+1}$
  $n \leftarrow n + 1$
  $u_{n+1}^0 = u_n - z_n$;
  Compute $F(u_{n+1}^0)$ and $\rho = \|F(u_{n+1}^0)\|$
end while

2 Theory

The first thing we need to prove is that z_n is a good approximation of an explicit Euler step. Once we do that we can use the method from our previous papers on pseudo-transient continuation (ΨTC) [11, 4, 12] to show that the iteration will track the dynamics if δ is sufficiently small.

We begin with a list of our assumptions on F and (1.1)

Assumption 2.1. 1. F is uniformly Lipschitz continuously differentiable.

2. The solution of the IVP (1.1) converges to $u^* \in R^N$ as $t \to \infty$.
3. There is $\Delta_0 > 0$ such that if $|y - u_0| < \Delta_0$ then the solution of

$$v' = -F(v); v(0) = y \tag{2.1}$$

converges to u^* as $t \to \infty$.

4. There is a neighborhood $\mathcal{N}$ of $\{u(t) \,|\, t > 0\}$ on which F is uniformly bounded.
5. $F'(u^*)$ has positive real eigenvalues.

The last assumption on $F'(u^*)$ is needed for the terminal phase of the iteration, and is part of the price we pay for an explicit method.

2.1 Getting Close to u^*

We can follow the plan for analysis of implicit ΨTC only as far at the first step, and do so in Theorem 2.2. The proof in § 2.2 of convergence to u^* in the terminal phase requires something different.

In what follows $\|\cdot\|$ denotes any norm on R^N.

Theorem 2.2. *Assume that Assumption 2.1 holds. Let the parameter ϵ in Algorithm 1 be given. Then for any $\Delta > 0$ and all $\delta > 0$ sufficiently small there is $n > 0$ such that*

$$\|u_n - u^*\| < \Delta, \|u_n^0 - u^*\| < \Delta, \|z_n\| < \Delta,$$

and $u_k \in \mathcal{N}$ for all $0 \le k \le n$.

Proof. We will apply the technique from [11, 4, 12], where we show that the iteration differs from a forward Euler step by $O(\delta^2)$, where the constant in the O-term depends only on F. This directly implies the result using exactly the analysis from [11].

Let

$$M = \sup_{x \in \mathcal{N}} \|F(u)\|, \tag{2.2}$$

and let $T > 0$ be such that

$$\|x(T)\| < \Delta. \tag{2.3}$$

Let $\delta = T/n$. We will change n as we complete the proof.

From the relation $\omega\epsilon = (1-\omega)\delta$ we have

$$\|z_{n+1}\| \le (1-\omega)\delta M + \omega\|z_n\| \tag{2.4}$$

which, since $\|z_0\| \le \delta M$ implies that

$$\|z_n\| \le M\delta, \tag{2.5}$$

for all n.

Let

$$E_n = z_n - \delta F(u_n) \text{ and } \mu_n = \|E_n\|/\delta^2.$$

We can write

$$\begin{aligned} z_{n-1} &= \delta F(u_{n-1}) + E_{n-1} \\ &= \delta F(u_n) + \delta(F(u_{n-1}) - F(u_n)) + \mu_{n-1}\delta^2. \end{aligned} \tag{2.6}$$

We let L denote the Lipschitz constant of F. From (2.5) and (2.6) we have

$$\|z_{n-1} - \delta F(u_n)\| \le (LM + \mu_{n-1})\delta^2. \tag{2.7}$$

Similarly

$$\|F(u_n^0) - F(u_n)\| \le LM\delta. \tag{2.8}$$

We combine (2.7) and (2.8) with

$$z_n = \omega(\epsilon F(u_n^0) + z_{n-1})$$

to conclude that

$$\begin{aligned} z_n &= \omega(\epsilon F(u_n^0) + z_{n-1}) = (1-\omega)\delta F(u_n^0) + \omega z_{n-1} \\ &= (1-\omega)\delta F(u_n) + \omega\delta F(u_n) + E_n \\ &= \delta F(u_n) + E_n, \end{aligned} \tag{2.9}$$

where

$$\begin{aligned} \|E_n\| &= \|(1-\omega)\delta(F(u_n^0) - F(u_n)) + \omega(z_{n-1} - \delta F(u_n))\| \\ &\le (1-\omega)2LM\delta^2 + \omega(LM + \mu_{n-1})\delta^2 = (LM + \omega\mu_{n-1})\delta^2. \end{aligned}$$

So,

$$\mu_n \le 2LM + \omega\mu_{n-1}, \tag{2.10}$$

which implies, since $\mu_0 = 0$, that

$$\mu_n \leq \mu_{max} \equiv 2LM/(1-\omega). \tag{2.11}$$

Now let $\Delta_0 < \Delta/2$ be such that if $\|u - u(t)\| < \Delta_0$ for any $t \geq 0$ then $u \in \mathcal{N}$. Let T be such that

$$\|u(t) - u^*\| < \Delta_0/2$$

for all $t > T$.

We can use the same analysis as was used in [11] to argue that there is $C > 0$ such that

$$\|u_k - u(k\delta)\| \leq C\delta \tag{2.12}$$

whenever $(k-2)\delta \leq T$. Assume that

$$\delta < \frac{\Delta_0}{2(C+M)}, \tag{2.13}$$

and define n by

$$(n-2)\delta \leq T < (n-1)\delta.$$

Then (2.12) and (2.13) imply that

$$u_k \in \mathcal{N} \text{ for all } 0 \leq k \leq n.$$

Since $(n-1)\delta > T$, we see that for $k = n-1, n$,

$$\|u_k - u^*\| \leq \|u_k - u(k\delta)\| + \|u(k\delta) - u^*\| < \Delta_0 < \Delta/2,$$

and hence, setting $k = n$, $\|u_n - u^*\| < \Delta$. Finally, to show that $\|u_n^0 - u^*\| < \Delta$ we note that

$$\|u_n^0 - u^*\| \leq \|u_{n-1} - u^*\| + \|z_{n-1}\| \leq \Delta/2 + M\delta \leq \Delta/2 + \Delta_0/2 < \Delta,$$

completing the proof. □

2.2 Convergence Near the Solution

We must now prove convergence from points near the solution. We will begin with the convergence result for linear problems from [6].

Theorem 2.3. *Let A have positive real eigenvalues and let $F(u) = Au$. Then if $\epsilon\rho(A) < 4/3$ the iteration converges to $u^* = 0$ for all $\delta > 0$.*

Proof. We use the formulae

$$\begin{aligned} u_{n+1}^0 &= u_n - z_n \\ z_{n+1} &= \omega(\epsilon A u_{n+1}^0 + z_n) = \omega(\epsilon A(u_n - z_n) + z_n) \\ &= \omega(\epsilon A u_n + (I - \epsilon A) z_n) \end{aligned}$$

and

$$u_{n+1} = u_n - z_{n+1} = (I - \omega\epsilon A)u_n - \omega(I - \epsilon A)z_n$$

to see that

$$\begin{pmatrix} u_{n+1} \\ z_{n+1} \end{pmatrix} = \mathcal{A} \begin{pmatrix} u_n \\ z_n \end{pmatrix}. \tag{2.14}$$

In (2.14) the $2N \times 2N$ matrix $\mathcal{A}$ is

$$\mathcal{A} = \begin{pmatrix} I - \omega\epsilon A & -\omega I + \omega\epsilon A \\ \\ \omega\epsilon A & \omega I - \omega\epsilon A \end{pmatrix}.$$

We want to show that the spectral radius of the matrix in (2.14) is smaller than one. Let μ be an eigenvalue of $\mathcal{A}$ and let $U = (\xi^T, \eta^T)^T$ be the corresponding eigenvector. The equation $\mathcal{A}U = \mu U$ implies that

$$(I - \omega\epsilon A)\xi + (-\omega I + \omega\epsilon A)\eta = \mu\xi \tag{2.15}$$

and

$$\omega\epsilon A\xi + (\omega I - \omega\epsilon A)\eta = \mu\eta. \tag{2.16}$$

If we add (2.15) to (2.16) we obtain

$$\xi = \mu(\xi + \eta). \tag{2.17}$$

Hence η is a scalar multiple of ξ, and both are eigenfunctions of A. If $A\xi = \lambda\xi$, then (2.16) and (2.17) imply that

$$\mu^2 - (1 + \omega - 2\omega\epsilon\lambda)\mu + \omega(1 - \epsilon\lambda) = 0.$$

A tedious but routine calculation shows that the roots of this polynomial are < 1 for all $\delta > 0$ if $\epsilon\lambda < 4/3$.

Having shown that $u_n \to 0$ and $z_n \to 0$, it is clear that $u_n^0 \to 0$. □

This result may seem strange, since the stability region for explicit methods must be bounded [1]. However, it is not correct in this case to interpret δ as a step size in the traditional sense.

To complete the convergence analysis we must show that if u_m and u_m^0 are sufficiently near u^* for some $m > 0$, that $u_n \to u^*$, $z_n \to 0$, and therefore $u_n^0 \to u^*$. This will connect the result from Theorem 2.2, which says that for δ sufficiently small one can get arbitrarily close to u^*, with local convergence in the terminal phase.

As is standard we define $e = u - u^*$.

Theorem 2.4. *Let Assumption 2.1 hold and assume that $F'(u^*)$ has positive real eigenvalues. Assume that*

$$\epsilon\rho(F'(u^*)) < 4/3. \tag{2.18}$$

Let $\{u_n\}$ be the iteration from Algorithm 1. Then there is $\Delta > 0$ and a norm $\|\cdot\|_*$ on R^{2N} such that if

$$\left\| \begin{pmatrix} e_m \\ z_m \end{pmatrix} \right\|_* < \Delta$$

then

$$\left\| \begin{pmatrix} e_n \\ z_n \end{pmatrix} \right\|_* < \Delta$$

for all $n \geq m$ and

$$\begin{pmatrix} u_n \\ z_n \end{pmatrix} \to \begin{pmatrix} u^* \\ 0 \end{pmatrix}$$

q-linearly.

Proof. For u_n near u^* and z_n sufficiently small we have

$$\begin{pmatrix} e_{n+1} \\ z_{n+1} \end{pmatrix} = \mathcal{J}^* \begin{pmatrix} e_n \\ z_n \end{pmatrix} + E(u_n, z_n), \tag{2.19}$$

where the $2N \times 2N$ matrix $\mathcal{J}^*$ is

$$\mathcal{J}^* = \begin{pmatrix} I - \omega\epsilon F'(u^*) & -\omega I + \omega\epsilon F'(u^*) \\ \\ \omega\epsilon F'(u^*) & \omega I - \omega\epsilon F'(u^*) \end{pmatrix}. \tag{2.20}$$

Now let $\sigma^* = \rho(\mathcal{J}^*)$ be the spectral radius of $\mathcal{J}^*$. Theorem 2.3 says that $\sigma^* < 1$. Let $\|\cdot\|_*$ be a vector norm on R^{2N} for which

$$\|\mathcal{J}^*\|_* \le \sigma \equiv \frac{1+\sigma^*}{2}.$$

Taylor's theorem implies that the error E can be bounded by

$$\|E\|_* \le K \left\| \begin{pmatrix} e_{n+1} \\ z_{n+1} \end{pmatrix} \right\|_*^2. \tag{2.21}$$

Let m satisfy the assumptions of the theorem and let $\Delta > 0$ be small enough so that

$$\left\| \begin{pmatrix} e_m \\ z_m \end{pmatrix} \right\|_* < \frac{1-\sigma}{2K}$$

where K is the constant in the bound (2.21). Then

$$\left\| \begin{pmatrix} e_{m+1} \\ z_{m+1} \end{pmatrix} \right\|_* \le (\sigma + (1-\sigma)/2) \left\| \begin{pmatrix} e_m \\ z_m \end{pmatrix} \right\|_*.$$

Since $(\sigma + (1-\sigma)/2) < (1+\sigma)/2 < 1$, the proof is complete. □

We can now combine Theorems 2.2 and 2.4 to prove a complete convergence result.

Theorem 2.5. *Let Assumption 2.1 hold. Assume there is K so that $\delta_n \le K\delta_0$ for all n. Then if $\epsilon\rho(\mathcal{J}^*) < 4/3$ and δ_0 is sufficiently small, $u_n^0 \to x$ and $z_n \to 0$ q-linearly.*

Proof. We define a norm on R^{2N}

$$\left\| \begin{pmatrix} u \\ z \end{pmatrix} \right\|_0 = \|u\| + \|z\|.$$

Since all norms on R^{2N} are equivalent, there is $\nu > 1$ so that

$$\nu^{-1} \left\| \begin{pmatrix} u \\ z \end{pmatrix} \right\|_0 \le \left\| \begin{pmatrix} u \\ z \end{pmatrix} \right\|_* \le \nu \left\| \begin{pmatrix} u \\ z \end{pmatrix} \right\|_0. \tag{2.22}$$

By Theorem 2.2 we may adjust δ_0 so that

$$\left\| \begin{pmatrix} e_m \\ z_m \end{pmatrix} \right\|_0 \le \Delta^*/\nu$$

for some $m < \infty$. This will imply that

$$\left\| \begin{pmatrix} e_m \\ z_m \end{pmatrix} \right\|_* \le \Delta.$$

Hence the conclusion of Theorem 2.4 holds and the vector $(u_n, z_n)^T \to 0$ q-linearly in the $\|\cdot\|_*$ norm. □

2.3 Bound Constraints

In this section we will indicate how the methods of [12] can be applied to extend the explicit ΨTC method to optimization problems with simple bound constraints. The optimization problem is

$$\min f(u) \text{ subject to the bound constraints } L \le u \le U \tag{2.23}$$

where the inequalities are component wise. We assume that f is twice Lipschitz continuously differentiable.

The gradient flow equation associated with (2.23) is

$$u' = -F(u) \tag{2.24}$$

where

$$F(u) = u - \mathcal{P}(u - \nabla f(u)). \tag{2.25}$$

In (2.25) $\mathcal{P}(u)_i = \max(L_i, \min(U_i, (u)_i))$, where $(u)_i$ denotes ith component of a vector $u \in R^N$.

The explicit ΨTC algorithm projects onto the bound constraints at the end of each step. The algorithm is

Algorithm 2 Explicit ΨTC for Bound Constraints

$n = 0$; $z_0 = hF(u_0)$;
$u_1^0 = \mathcal{P}(u_0 - z_0)$;
Compute $F(u_1^0)$ and $\rho = \|F(u_1^0)\|$
while $\rho > \tau$ **do**
  $z_{n+1} = \omega(\epsilon F(u_n^0) + z_n)$
  $u_{n+1} = \mathcal{P}(u_n - z_{n+1})$
  $n \leftarrow n + 1$
  $u_{n+1}^0 = \mathcal{P}(u_n - z_n)$;
  Compute $F(u_{n+1}^0)$ and $\rho = \|F(u_{n+1}^0)\|$
end while

Clearly F is Lipschitz continuous, so the convergence results from § 2.1 apply directly. However, F is not differentiable in the classical sense, but has a structure which allows us to extend the results in § 2.2, as we do in this section.

We must modify Assumption 2.1 slightly. We assume that the initial data u_0 is such that

Assumption 2.6. 1. ∇f is uniformly Lipschitz continuously differentiable.

2. The solution of the IVP (1.1) converges to $u^* \in R^N$ as $t \to \infty$.
3. There is $\Delta_0 > 0$ such that if $|y - u_0| < \Delta_0$ then the solution of

$$v' = -F(v); v(0) = y, \tag{2.26}$$

converges to u^* as $t \to \infty$.

4. There is a neighborhood $\mathcal{N}$ of $\{u(t) \,|\, t > 0\}$ on which F is uniformly bounded.
5. u^* satisfies the second-order sufficient conditions for optimality.

The last part of Assumption 2.6 is the same as that of the last part of Assumption 2.1 in the unconstrained case. To explain how the bound constrained case differs we must review the taxonomy of constraints [9, 3, 12]. We define three sets of indices. The set $\mathcal{I}$ of inactive indices is

$$\mathcal{I} = \{i \,|\, L_i < (u^*)_i < U_i\}.$$

The set of active indices is the complement of $\mathcal{I}$

$$\mathcal{A} = \{i \,|\, (u^*)_i = L_i \text{ or} (u^*)_i = U_i\}.$$

Finally, the set of binding constraints is

$$\mathcal{B} = \{i \,|\, (u^*)_i = L_i \text{ and } \partial f(u^*)/\partial u_i > 0\} \cup \{i \,|\, (u^*)_i = U_i \text{ and } \partial f(u^*)/\partial u_i < 0\}.$$

Note that if i is active but not binding, then

$$\partial f(u^*)/\partial u_i = 0.$$

We define the modified Hessian H^* of F at u^* by

$$H^*_{ij} = \begin{cases} \delta_{ij} & \text{if } i \in \mathcal{B} \text{ or } j \in \mathcal{B}, \\ \frac{\partial^2 f(u^*)}{\partial u_i \partial u_j} & \text{otherwise.} \end{cases}$$

The sufficient conditions for optimality at u^* are that $F(u^*) = 0$ and H^* is positive definite.

For implicit ΨTC , [12], one could show that the binding constraints were identified in finitely many steps and then apply the theory for unconstrained problems. That is not so with explicit ΨTC , and that fact makes the theory a bit weaker.

Theorem 2.7. *Assume that Assumption 2.6 holds and that $\mathcal{A} = \mathcal{B}$. Then F is differentiable near u^*, $F'(u^*)$ has positive real eigenvalues, and the conclusions of Theorem 2.4 hold.*

Proof. Without loss of generality we may assume that the active indices are $1 \le i \le k$ and the inactive indices are $k+1 \le i \le N$. Then the modified Hessian is block diagonal

$$H^* = \begin{pmatrix} I & 0 \\ 0 & R^* \end{pmatrix}. \tag{2.27}$$

In (2.27) I is the $k \times k$ identity matrix and R^* is the $(N-k) \times (N-k)$ reduced Hessian. The second-order sufficiency conditions imply that R^* is positive definite.

Let $\Delta > 0$ be small enough so that if $\|u - u^*\| < \Delta$ that

$$F(u)_i = (u)_i - \min(U_i, \max((u)_i - \partial f/\partial u_i, L_i)) = (u)_i - (u^*)_i$$

for all $i \in \mathcal{A}$. We can do this because $\mathcal{A} = \mathcal{B}$. So, if $\|u - u^*\| < \Delta$ then the first k rows of F' are the same as the first k rows of H^*.

We can also, reducing Δ if necessary, arrange things so that if $i \in \mathcal{I}$ then $|(u)_i - (u^*)_i| < \max(U^*_i - (u^*)_i, (u^*)_i - L_i)$. Then

$$F(u)_i = -\partial f/\partial u_i,$$

and hence, for all $i \in \mathcal{I}$ the ith row of F' is simply the i row of the Hessian of f.

Therefore, for all u sufficiently near u^*,

$$F'(u) = \begin{pmatrix} I & 0 \\ C & R^* \end{pmatrix}, \tag{2.28}$$

where C is the $(N-k) \times k$ block of mixed partials $\partial^2 f/\partial u_i \partial u_j$ with $i \in \mathcal{I}$ and $j \in \mathcal{A}$. Since R^* is symmetric positive definite, the eigenvalues of F' are $\lambda = 1$ and the eigenvalues of R^*.

The remainder of the proof is exactly the same as that of Theorem 2.4. □

2.4 Step Size Control

An efficient implementation requires an algorithm for stepsize control. In this section we propose an algorithm based on a hybrid of a standard approach for implicit ΨTC with safeguarding to maintain stability of the explicit algorithm. The approach in [6] was to increase δ by doubling whenever $\|F(u_n)\|$ declined below a given threshold. The algorithm in [6] allowed for at most three increases of δ. Our objective is to provide a more flexible method with fewer arbitrary decisions and a way to reduce the step.

The assumption in Theorem 2.5 that $\delta_n \le K\delta_0$ is only necessary in the first phase of the iteration. Once u_n is sufficiently near u^*, one could in principle increase δ very rapidly, since Theorem 2.4 requires no upper limit on δ. However, it is not clear how one determines when δ can be increased more rapidly. Our experiments show little benefit from dramatic increases of δ.

Our implementation uses a variation on the "switched evolution relaxation" (SER) [16] which is a common approach in the implicit case. In the implicit case the update formula is

$$\delta_{n+1} = \delta_n \frac{\|F(u_n)\|}{\|F(u_{n+1})\|} \tag{2.29}$$

which may be safeguarded to keep δ from becoming too large

$$\delta_{n+1} = \min\left(\delta_{max}, \delta_n \frac{\|F(u_n)\|}{\|F(u_{n+1})\|}\right)$$

where δ_{max} is an upper limit on δ.

In the implicit case we have found that we must take more care with both the size and the frequency of the changes in δ. Our approach is to monitor the logarithmic change in $\|F\|$

$$\sigma_n = \log\|F(u_{n+1})\| - \log\|F(u_n)\|$$

and to only update the step if $\sigma_n > -1/2$. This keeps the step from rapid oscillation. If $\sigma_n > -1/2$, we use SER to update the step, but safeguard the update by limiting any change to 50%.

In the special case where $F(u) = \nabla f(u)$, we also test the initial step δ_0 and reduce it, if necessary, to make $f(u_1) < f(u_0)$.

3 Parameter Identification Example

This is a small $N = 2$ example taken from [9, 2, 12]. The problem is to determine the damping coefficient c and spring constant k for a simple harmonic oscillator using measured data. This example has bound constraints.

The underlying differential equation is

$$w'' + cw' + kw = 0; w(0) = w_0, w'(0) = 0, \tag{3.1}$$

on the interval $[0, 1]$. As in our previous work, we let $u = (c, k)^T$ and use as data the values of the analytic solution at 1000 equally spaced points. We integrate the dynamics with the MATLAB code `ode15s` with relative and absolute error tolerances set to 10^{-6}.

This is a nonlinear least squares problem with residual R having components

$$R(u)_i = w^{exact}(t_i) - w_i(u)$$

for $1 \leq i \leq 1000$. Here w_i is the solution from `ode15s` and w^{exact} the analytic solution.

The objective function is

$$f(u) = \|R(u)\|_2^2/2$$

and the gradient which drives our ΨTC algorithm is

$$\nabla f(u) = R'(u)^T R(u)$$

where R' is the Jacobian of R. The ΨTC algorithm seeks to integrate

$$u' = -F(u)$$

to steady state, where $F(u)$ is a descent direction for f at u. In [12] we used a projected Levenberg-Marquardt iteration, which was a successful approach for implicit ΨTC .

The implicit ΨTC method proposed in [12] solved this problem with simple bound constraints very efficiently. The objective here is to test the explicit approach without constraints. In [12] we imposed bound constraints

$$0 \leq c, k \leq 10$$

and arranged the data so that

$$u^* = (c^*, k^*)^T = (1, 1)^T$$

is the solution.

One way to attach this problem is to drive the dynamics with the Gauss-Newton direction

$$F(u) = R'(u)^\dagger R(u) = (R'(u)^T R'(u))^{-1} R'(u)^T R(u). \tag{3.2}$$

Assuming that R' has full rank throughout the domain of interest then the assumptions of the convergence results are satisfied if we bound u away from $(0, 0)^T$, where the Jacobian is singular. This was not necessary in our work on the implicit approach because the iteration avoided the singularity. That is not the case here and we have seen the iteration fail because the sequence of iterations was too near the singularity. Hence we use

$$\Omega = \{c, k \,|\, .1 \leq c, k \leq 10\}$$

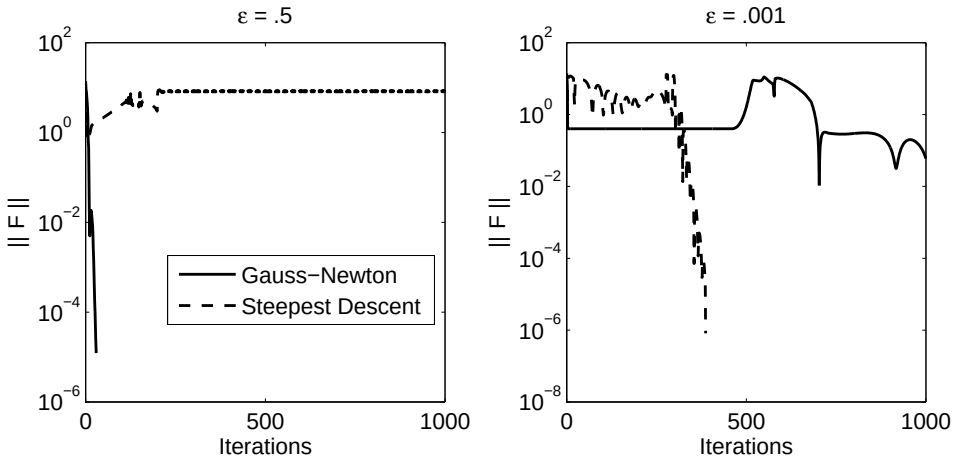
as the feasible set in this section.

Since $F'(u^*) = I$, a choice of $\epsilon = 1/2$ satisfies the convergence theory. Using (3.2) is unrealistic, of course, since a linear least squares problem must be solved at each step, so the work per step is no less than the implicit approach from [12]. It is useful to look at this case to see how one might benefit from a well-conditioned problem. We compare ΨTC with the Gauss-Newton direction to ΨTC with the steepest descent direction

$$F(u) = \nabla f(u) = R'(u)^T R(u) \tag{3.3}$$

with an initial iterate of $u_0 = (10, 10)^T$. We set $\epsilon = 1/2$ and $\delta_0 = .1$ in this example. We terminated the iteration when the norm of F had been reduced by a factor of 10^{-6}. As one can see from the iteration history in left side of Figure 1, ΨTC converged (in 11 iterations) using the Gauss-Newton direction and completely failed with the steepest descent direction. The reason for this difference in performance is that $\epsilon = 1/2$ is a good choice for the Gauss-Newton formulation and a very poor one for steepest descent.

Figure 1: Parameter Identification: Unconstrained



If we solve the problem again with $\epsilon = 1.d-3$, we see a different result. In this case ϵ is well-chosen for the steepest descent iteration. The right side of Figure 1 shows the relative performances of the two formulations are reversed, illustrating the effect of ϵ on the performance of the iteration.

If we impose simple bounds,

$$L_i \le u_i \le U_i \text{ for } i = 1, 2 \tag{3.4}$$

we must alter F by projection onto the tangent space of constraint set. In the steepest descent case we use

$$F(u) = u - \mathcal{P}(u - \nabla f(u))$$

and for Gauss-Newton we use

$$F(u) = u - \mathcal{P}(u - (R'(u)^T R'(u))^{-1} \nabla f(u)).$$

In both cases $\mathcal{P}$ is the projection onto the constraint set

$$\mathcal{P}(u)_i = \max(L_i \min(U_i, u_i)).$$

In the computations reported here we terminate the iteration once the residual norm has been reduced by a factor of 10^{-3}.

Figures 2 and 3 apply this formulation to two other examples from [12]. Both are bound constrained problems with upper bounds $(10, 10)^T$. The effect of changing ϵ is visible in both examples. In Figure 2 the lower bounds are $(2, .1)^T$. This problem does not have zero residual because the global minimum is outside the bounds. The constraint $c = 2$ is active at the solution. In Figure 3 the lower bounds are $(1, 1)^T$, and the the problem is degenerate, which means that there are non-binding active constraints. Theorem 2.7 does not apply to this case, but the numerical results look promising.

4 Conclusions

In this paper we have analyzed and extended the results from [6] on explicit pseudo-transient continuation. The method has promise because, as with all explicit methods, no linear

Figure 2: Parameter Identification: Active Constraint

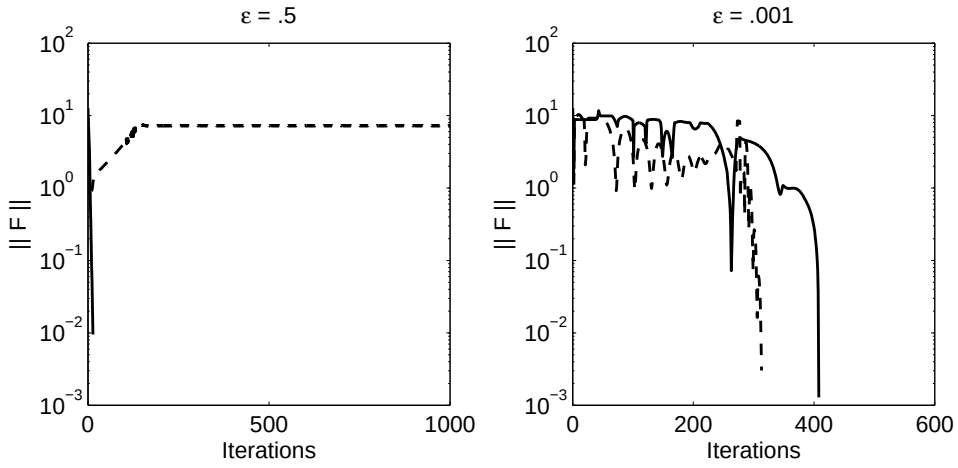
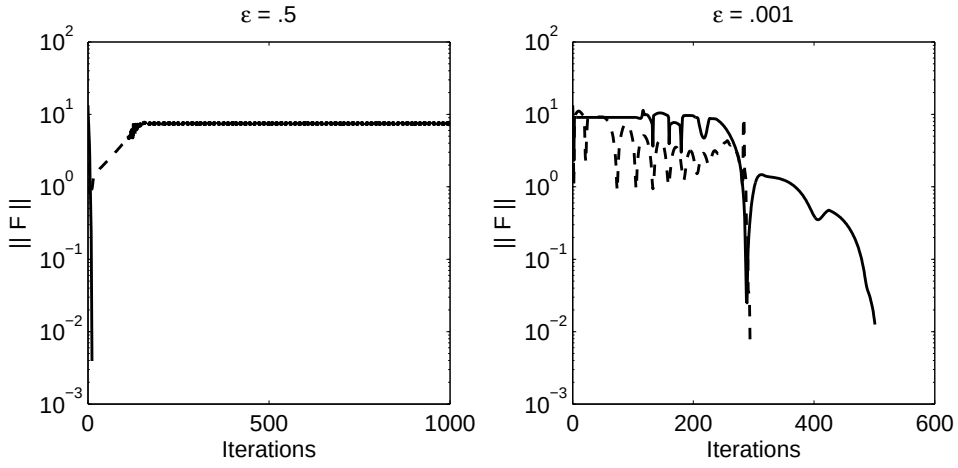


Figure 3: Parameter Identification: Degenerate



equations need to be solved to compute the iterations. The price one pays for this is that more iterations are required than the standard implicit pseudo-transient algorithms. The method, at least as it stands now, performs best if one has knowledge of the spectrum of the Jacobian. That knowledge may be difficult to obtain, and further research is needed to find ways to adaptively tune the method and reduce the number of iterations needed for convergence.

References

[1] U.M. Ascher and L.R. Petzold, *Computer Methods for Ordinary Differential Equations and Differential Algebraic Equations*, SIAM, Philadelphia, 1998.

[2] H.T. Banks and H.T. Tran, *Mathematical and experimental modeling of physical pro-*

cesses, Department of Mathematics, North Carolina State University, unpublished lecture notes for Mathematics 573-4, 1997.

[3] D.P. Bertsekas, On the Goldstein-Levitin-Polyak gradient projection method, *IEEE Trans. Autom. Control* 21 (1976) 174–184.

[4] T. Coffey, C.T. Kelley, and D.E. Keyes, Pseudo-transient continuation and differential-algebraic equations, *SIAM J. Sci. Comp.* 25 (2003) 553–569.

[5] J.E. Dennis and R.B. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, no. 16 in Classics in Applied Mathematics, SIAM, Philadelphia, 1996.

[6] T. Han and Y. Han, Solving large scale nonlinear equations by a new ODE numerical integration method, *Applied Mathematics* 1 (2010) 222–229.

[7] T. Han, X. Luo, and Y. Han, Solving large scale unconstrained minimization problems by a new ODE numerical integration method, *Applied Mathematics* 2 (2011) 527–532.

[8] C.T. Kelley, *Iterative Methods for Linear and Nonlinear Equations*, no. 16 in Frontiers in Applied Mathematics, SIAM, Philadelphia, 1995.

[9] C.T. Kelley, *Iterative Methods for Optimization*, no. 18 in Frontiers in Applied Mathematics, SIAM, Philadelphia, 1999.

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

[11] C. T. Kelley and D. E. Keyes, Convergence analysis of pseudo-transient continuation, *SIAM J. Numer. Anal.* 35 (1998) 508–523.

[12] C.T. Kelley, L.-Z. Liao, L. Qi, M.T. Chu, J.P. Reese, and C. Winton, Projected pseudo-transient continuation, *SIAM J. Numer. Anal.* 46 (2008) 3071–3083.

[13] D.E. Keyes and M.D. Smooke, Flame sheet starting estimates for counterflow diffusion flame problems, *J. Comput. Phys.* 72 (1987) 267–288.

[14] D.E. Keyes and M.D. Smooke, A parallelized elliptic solver for reacting flows, in *Parallel Computations and Their Impact on Mechanics*, A.K. Noor (ed.), ASME, 1987, pp. 375–402.

[15] X.-L. Luo, C.T. Kelley, L.-Z. Liao, and H.-W. Tam, Combining trust-region techniques and Rosenbrock methods for gradient systems, *J. Optim. Theory Appl.* 140 (2009) 265–286.

[16] W. Mulder and B. V. Leer, Experiments with implicit upwind methods for the Euler equations, *J. Comp. Phys.* 59 (1985) 232–246.

[17] J.M. Ortega and W.C. Rheinboldt, *Iterative Solution of Nonlinear Equations in Several Variables*, Academic Press, New York, 1970.

[18] L.-H. Zhang, C.T. Kelley, and L.-Z. Liao, A continuous Newton-type method for unconstrained optimization, *Pacific Journal of Optimization* 4 (2008) 259–277.

Manuscript received 23 November 2011
revised 18 April 2012
accepted for publication 18 April 2012

C. T. Kelley
North Carolina State University, Department of Mathematics
Box 8205, Raleigh, NC 27695-8205, USA
E-mail address: Tim_Kelley@ncsu.edu

Li-Zhi Liao
Department of Mathematics, Hong Kong Baptist Unversity
Kowloon Tong, Kowloon, Hong Kong, China
E-mail address: liliao@hkbu.edu.hk