IMPLICIT FILTERING AND NONLINEAR LEAST SQUARES PROBLEMS

C. T. Kelley

North Carolina State University
Center for Research in Scientific Computation
Department of Mathematics
Box 8205, Raleigh, N. C. 27695-8205, USA

Tim_Kelley@ncsu.edu

Abstract   In this paper we motivate and analyze a version of the implicit filtering algorithm by viewing it as an extension of coordinate search. We then show how implicit filtering can be combined with the damped Gauss-Newton method to solve noisy nonlinear least squares problems.

1.   Introduction

The purposes of this paper are to show how a version of the implicit filtering algorithm [24, 17, 16] can be motivated and analyzed by viewing it as an elaboration of coordinate search, and to describe and analyze a implicit filtering Gauss-Newton method for nonlinear least squares problems.

Our approach to nonlinear least squares problems is based on a finite-difference form of the damped Gauss-Newton method [11, 24, 32], but differs from that in the MINPACK [30] routine `lmdif.f`. That code uses forward difference Jacobians with a user-defined difference increment, but that increment is set only once. Implicit filtering uses a central difference not only to compute more accurate Jacobians, but more importantly to avoid local minima and to decide when to reduce the difference increment.

Implicit filtering, which we describe in § 2, is a deterministic stencil-based sampling method. In general terms, implicit filtering is a finite-difference quasi-Newton method in which the size of the difference stencil

decreases as the optimization progresses. In this way one hopes to "filter" low-amplitude, high-frequency noise in the objective function.

Sampling methods do not use derivatives, but rather sample the objective function on a stencil or pattern to determine the progress of the iteration and whether or not to change the size, but not the shape, of the stencil. Many of these methods, like implicit filtering, the Hooke-Jeeves [20] method, and multidirectional search [38, 39], reduce the size of the stencil in the course of the optimization. The stencil-size reduction policy leads to a convergence theory [24, 5, 39].

The best-known sampling method is the Nelder-Mead [31] algorithm. This method uses an irregular pattern that changes as the optimization progresses, and hence is not stencil-based in the sense of this paper. Analytical results for the Nelder-Mead algorithm are limited [24, 5, 26]. Theoretical developments are at also a very early stage for more aggressive sampling methods, like the DIRECT [22] algorithm, [14, 15].

Sampling methods, for the most part, need many iterations to obtain a high-precision result. Therefore, when gradient information is available and the optimization landscape is relatively smooth, conventional gradient-based algorithms usually perform far better. Sampling methods do well for problems with complex optimization landscapes like the ones in Figure 1, where nonsmoothness and nonconvexity can defeat most gradient based methods.
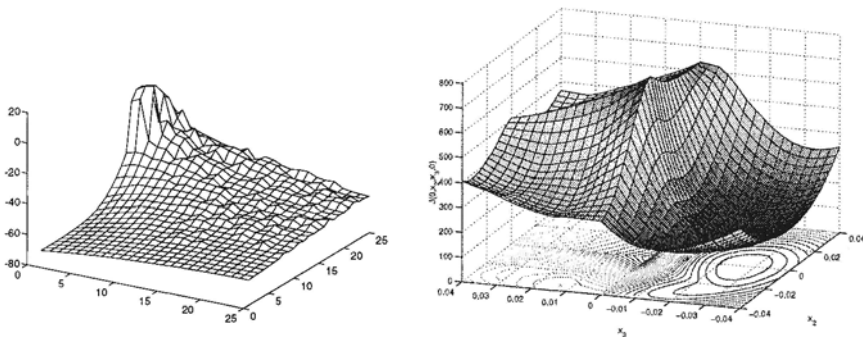

Figure 1. Optimization Landscapes

We caution the reader that sampling methods are not designed to be true global optimization algorithms. Problems with violently oscillatory optimization landscapes are candidates for genetic algorithms [19, 35], simulated annealing [25, 41], or the DIRECT algorithm [22, 21].

The paper is organized as follows. In § 2 we briefly describe the implicit filtering method and some of the convergence results. We describe

the new algorithm in § 3 and prove a local convergence result. In § 4 we illustrate the ideas with a parameter identification problem.

2.   Implicit Filtering

In this section we introduce implicit filtering. We show how the method can be viewed as an enhanced form of a simple coordinate search method. Convergence analysis for methods of this type is typically done in a setting far simpler than one sees in practice. Many results require smooth objective functions [28, 26, 12, 39, 8, 9] or objective functions that are small perturbations of smooth functions [29, 17, 23, 5, 24, 7, 44]. The main results in this paper make the latter assumption. We will also assume that the noise decays near an optimal point. Such decay has been observed in practice [36, 10, 42, 43, 37, 4] and methods designed with this decay in mind can perform well even when the noise does not decay to zero as optimality is approached.

2.1   Coordinate Search

We begin with a discussion of a coordinate search algorithm, the simplest of all sampling methods, and consider the unconstrained problem

$$\min_{x \in R^N} f(x). \tag{1}$$

From a current point x_c and stencil radius or **scale** h_c we sample f at the $2N$ points

$$S(x_c, h_c) = \{x_c \pm h_c e_j\}, \tag{2}$$

where e_j is the unit vector in the jth coordinate direction. Then either x_c or h_c is changed.

- If

$$f(x_c) \le \min_{x \in S(x_c, h_c)} f(x) \tag{3}$$

  then we replace h_c by $h_+ = h_c/2$ and set $x_+ = x_c$.

- Otherwise, we replace x_c by any point in $x_+ \in S$ such that

$$f(x_+) = \min_{x \in S(x,h)} f(x)$$

  and let $h_+ = h_c$.

We refer to (3) as **stencil failure**. If f is Lipschitz continuously differentiable, then [24, 5] stencil failure implies that

$$\|\nabla f(x_c)\| = O(h_c). \tag{4}$$

Now, if f has bounded level sets, h will be reduced infinitely many times because there are only finitely many points on the grid with function values smaller than $f(x_c)$ [38]. Hence, by (4), the gradient of f will be driven to zero, giving subsequential convergence to a point that satisfies the necessary conditions for optimality.

One-sided stencils [24, 36] and more general stencils with $< 2N$ directions have also been used [1, 2, 27] and have similar theoretical properties. Our experience has been that a full centered-difference stencil is better in practice.

Sampling methods do more than solve smooth problems. Consider an objective which is the sum of a smooth function f_x and a non-smooth function ϕ, which we will refer to as the noise.

$$f(x) = f_s(x) + \phi(x) \tag{5}$$

We assume that ϕ is uniformly bounded and small relative to f_s, but make no smoothness or even continuity assumptions beyond that. High-frequency oscillations in ϕ could result in local minima of f which would trap a conventional gradient-based method far from a minimizer of f_s. If ϕ decays sufficiently rapidly near a minimizer of f, then the coordinate search method responds to f_s and, in a sense, "does not see" ϕ.

To quantify the claim above, we return to the concept of stencil failure. Define

$$\|\phi\|_{S(x,h)} = \max_{z \in S(x,h)} |\phi(x)|.$$

If (3) holds and f satisfies (5), then [24, 5]

$$\|\nabla f_s(x_c)\| = O\left(h_c + \frac{\|\phi\|_{S(x_c,h_c)}}{h_c}\right). \tag{6}$$

Now, let $\{x_n\}$ be the sequence of coordinate search iterations and $\{h_n\}$ be the sequence of stencil radii, which we will refer to as **scales**. If f has bounded level sets, then the set of possible iterations for a given scale h is finite, as they lie on a grid [39], hence $h_n \to 0$. If, moreover, the noise decays rapidly enough so that

$$\lim_{n \to \infty} \frac{\|\phi\|_{S(x_n,h_n)}}{h_n} = 0, \tag{7}$$

then $\nabla f_s(x_n) \to 0$, by (6).

This asymptotic result does not address an important practical issue. The number of times that h will be reduced during the optimization needs to be specified when the optimization begins or a limit on the

number of calls to f must be imposed. Most implementations of sampling methods use one or both of these as termination criteria.

In the simple case where f_s is a convex quadratic, for example, coordinate search, therefore, "jumps over" oscillations in ϕ early in the iteration, when h is large, and, after finding a neighborhood of the minimizer, increases the resolution (*i. e.* decreases the scale) and converges.

2.2   Implicit Filtering

The version of implicit filtering which we discuss in this paper accelerates coordinate search with a quasi-Newton method. We use the sample values to construct a centered difference gradient $\nabla_h f(x_c)$. We then try to take a quasi-Newton step

$$x_+ = x_c - H_c^{-1} \nabla_{h_c} f(x_c) \tag{8}$$

where H_c is a quasi-Newton model Hessian. We find that the BFGS [6, 18, 13, 34] works well for unconstrained problems. We reduce the scale when either the norm of the difference gradient is sufficiently small or stencil failure occurs.

We formally describe implicit filtering below as a sequence of calls to a finite-difference quasi-Newton algorithm (`fdquasi`) followed by a reduction in the difference increment. The quasi-Newton iteration is terminated on entry if stencil failure is detected. The other termination criteria of the quasi-Newton iteration reflect the truncation error in the difference gradient. The tolerance for the gradient

$$\|\nabla_h f(x)\| < \tau h \tag{9}$$

is motivated both by the heuristic that the step should be at least of the same order as the scale, by the implication (6) of stencil failure, and by the error estimate [24] [24]

$$\|\nabla f_s(x) - \nabla_h f(x)\| = O\left(h^2 + \frac{\|\phi\|_{S(x,h)}}{h}\right). \tag{10}$$

The performance of implicit filtering can be sensitive to the choice of the parameter τ if, as was the case for the earliest implementations of implicit filtering [36, 17, 10], the test for stencil failure is not incorporated into the algorithm.

The line search is not guaranteed to succeed because the gradient is not exact, therefore we allow only a few reductions in the step length before exiting the quasi-Newton iteration. If the line search fails, then sufficient decrease condition

$$f(x_c - \lambda \nabla_{h_c} f(x_c)) - f(x_c) < -\alpha\lambda\|\nabla_{h_c} f(x_c)\|^2 \tag{11}$$

has been violated. Here, as is standard, [11, 24], α is a small parameter, typically 10^{-4}. If both (9) and (11) fail, then one can show in some cases [17] that the noise is sufficiently larger that the scale to justify terminating the entire optimization. This leads to the question of selection of the smallest scale, which is open. In some special cases, [17] failure of the line search can be related to the size of noise, motivating termination of the entire optimization because the assumption that $\|\phi\|$ is much smaller than h is no longer valid.

Algorithm 1 fdquasi$(x, f, pmax, \tau, h, amax)$

$p = 1$
while $p \leq pmax$ and $\|\nabla_h f(x)\| \geq \tau h$ **do**
  compute f and $\nabla_h f$
  if (2) holds **then**
    terminate and report **stencil failure**
  end if
  update the model Hessian H if appropriate; solve $Hd = -\nabla_h f(x)$
  use a backtracking line search, with at most $amax$ backtracks, to find a step length λ
  if $amax$ backtracks have been taken **then**
    terminate and report **line search failure**
  end if
  $x \leftarrow x + \lambda d$
  $p \leftarrow p + 1$
end while
if $p > pmax$ report **iteration count failure**

Implicit filtering is a sequence of calls to **fdquasi** with the difference increments or **scales** reduced after each return from **fdquasi**.

Algorithm 2 imfilter$(x, f, pmax, \tau, \{h_n\}, amax)$

---

for $k = 0, \ldots$ **do**
  `fdquasi`$(x, f, pmax, \tau, h_n, amax)$
end for

Our analysis of coordinate search depended on the fact that

$$\|\nabla f_s(x_n)\| = O\left(h_n + \frac{\|\phi\|_{S(x_n, h_n)}}{h_n}\right) \tag{12}$$

when stencil failure occurred and that h was reduced when that happened. Since stencil failure directly implies success, as do (6) and (9)

together, the convergence result for coordinate search will hold for implicit filtering provided the line search only fails finitely often and the quasi-Newton iteration terminates because of stencil failure or satisfaction of (9), We summarize these observations in a theorem from [24].

Theorem 1 *Let f satisfy (5) and let ∇f_s be Lipschitz continuous. Let $h_n \to 0$ and let $\{x_n\}$ be the implicit filtering sequence. Assume that either (3) or (11) hold after each call to* fdquasi *(i. e. there is no line search failure or iteration count failure) for all but finitely many k. Then if*

$$\lim_{k \to \infty} (h_n + h_n^{-1} \|\phi\|_{S(x,h_n)}) = 0 \qquad (13)$$

then any limit point of the sequence $\{x_n\}$ is a critical point of f_s.

Theorem 1 does not explain the performance of implicit filtering in practice. In fact, other methods, such as coordinate search, Hooke-Jeeves, and MDS, also satisfy the conclusion of Theorem 1 if (13) holds, [24, 40]. Implicit filtering performs well only if a quasi-Newton model Hessian is used. The reasons for the efficacy of the quasi-Newton methods are not fully understood. A step toward such an understanding is in [7], where a superlinear convergence result is presented. That result is somewhat like the one we give in § 3 and we will summarize it here.

Assumptions on the rate of decrease of $\{h_n\}$ and of the size of ϕ must be made to prove convergence rates. Landscapes like those in Figure 1 motivated the qualitative decay assumption (13). To obtain superlinear convergence one must ask for much more and demand that h and ϕ satisfy

$$\|\nabla_h \phi(x)\| = O(\|x - x^*\|^{1+p}) \qquad (14)$$

for some $p > 0$. Here x^* is a local minimizer of f_s. Satisfaction of (14) is possible in practice if both ϕ and the scales h decrease near x^*. As an example, suppose that f_s has a local minimizer x^*, $\nabla^2 f_s$ is Lipschitz continuous in a neighborhood of x^*, $\nabla^2 f_s(x^*)$ is positive definite, and for x sufficiently near x^*,

$$|\phi(x)| = O(\|x - x^*\|^{2+2p}), \qquad (15)$$

for some $p > 0$. In that case, if one sets

$$h_{n+1} = \|\nabla_{h_n} f(x_{n+1})\|^{1+p}, \qquad (16)$$

and other technical assumptions hold, then one can show that the implicit filtering iteration, with the BFGS update, is locally superlinearly convergent to x^*.

3. Gauss-Newton Iteration and Implicit Filtering

For the remainder of this paper we focus on nonlinear least squares objective functions

$$f(x) = \frac{1}{2} \sum_{i=1}^{M} \|r_i(x)\|_2^2 = \frac{1}{2} R(x)^T R(x). \tag{17}$$

We assume that
$$R(x) = R_s(x) + \Phi(x) \tag{18}$$

where $R_s : R^N \to R^M$ is Lipschitz continuously differentiable. Here the noise Φ in the residual does not correspond to noise in any data in the problem, but rather noise in the computation of R. As an example, if one is doing a nonlinear fit to data, R might have the form $R = M(x) - d$, where d is a vector of data and x are the model parameters. The noise we have in mind is in the computation of M, not in d.

The noise Φ in R can be related to the noise ϕ in f by

$$\phi(x) = R(x)^T \Phi(x) + \Phi(x)^T \Phi(x)/2. \tag{19}$$

3.1 Implicit Filtering Gauss-Newton (IFGN) Algorithm

Our implementation of implicit filtering for nonlinear least squares differs from the one described in § 2 in two ways:

- The Jacobian of the residual, not the gradient of the objective function, is approximated by finite differences.

- The Gauss-Newton model Hessian is used instead of a quasi-Newton model Hessian.

We let $\nabla_h R(x)$ be the centered difference gradient of R based on the stencil $S(x, h)$. Our finite difference Gauss-Newton iteration Algorithm `fdgauss`, must be prepared for stencil failure and failure of the line search. The sufficient decrease condition is now

$$f(x_c - \lambda d) - f(x_c) < -\alpha\lambda((\nabla_h R(x_c))^T R(x_c))^T d. \tag{20}$$

where
$$d = -(\nabla_h R(x_c)^T \nabla_h R(x_c))^{-1} \nabla_h R(x_c)^T R(x_c)$$

is the IFGN direction.

Algorithm 3 fdgauss$(x, R, pmax, \tau, h, amax)$

$p = 1$

while $p \leq pmax$ and $\|(\nabla_h R(x))^T R(x)\| \geq \tau h$ **do**

 compute $f = R(x)^T R(x)/2$ and $\nabla_h R$

 if (2) holds **then**

  terminate and report **stencil failure**

 end if

 set $H = (\nabla_h R(x))^T (\nabla_h R(x))$; solve $Hd = -\nabla_h R(x)^T R(x)$.

 use a backtracking line search, with at most *amax* backtracks, to find a step length λ

 if *amax* backtracks have been taken **then**

  terminate and report **line search failure**

 end if

 $x \leftarrow x + \lambda d$

 $p \leftarrow p + 1$

end while

if $p > pmax$ report **iteration count failure**

The implicit filtering form of the damped Gauss-Newton method, (Algorithm IFGN) calls `fdgauss` repeatedly, reducing the scale with each iteration.

Algorithm 4 IFGN$(x, R, pmax, \tau, \{h_n\}, amax)$

---

for $k = 0, \ldots$ **do**

 `fdgauss`$(x, R, pmax, \tau, h_n, amax)$

end for

3.2   Convergence Analysis

We will make a distinction between the central difference gradient of $f = R^T R/2$ and the difference gradient computed via $(\nabla_h R)^T R$, since the two approximate gradients have different errors, especially in the small residual case.

For any function $\psi : R^N \to R^L$ (here $L = 1$ or $L = M$), define

$$\|\psi\|_{S(x,h)} = \max_{z \in S(x,h)} \|\psi(x)\|.$$

and

$$E(x, h, \psi) = h^2 + \frac{\|\psi\|_{S(x,h)}}{h}.$$

We can rewrite (10) as

$$\|\nabla f_s(x) - \nabla_h f(x)\| = O(E(x, h, \phi)). \tag{21}$$

Lemma 3.1 gives the analog of (21) for nonlinear least squares problems in (24) and refines (21) in (23). The error in $(\nabla_h R(x))^T R(x)$ is scaled by the residual norm, a fact we exploit for zero residual problems in Lemma 3.3.

Lemma 3.1 *Let R be given by (18). Assume that there is $K > 0$ such that*

$$\|\Phi\|_{S(x,h)} \leq K \|R_s(x)\|. \tag{22}$$

Then

$$\|\nabla f_s(x) - \nabla_h f(x)\| = O\left(h^2 + \frac{\|R_s\|_{S(x,h)}\|\Phi\|_{S(x,h)}}{h}\right), \tag{23}$$

$$\|\nabla f_s(x) - (\nabla_h R(x))^T R(x)\| = O(\|R_s(x)\|E(x, h, \Phi)), \tag{24}$$

and

$$\|R'_s(x)^T R'_s(x) - (\nabla_h R(x))^T \nabla_h R(x)\| = O(E(x, h, \Phi)). \tag{25}$$

The constants in the O-terms depend on the norm and the Lipschitz constant of R'.

Proof. The estimate (23) follows from (10) and (19).
We now prove (24). By definition,

$$(\nabla_h R(x))^T R(x) \quad = (\nabla_h (R_s(x) + \Phi(x)))^T (R_s(x) + \Phi(x))$$

$$= (\nabla_h R_s(x))^T R_s(x) + O\left(\|R(x)\|\frac{\|\Phi\|_{S(x,h)}}{h}\right)$$

$$= \nabla f_s(x) + O\left(\|R_s(x)\|E(x, h, \Phi) + \frac{\|\Phi\|^2_{S(x,h)}}{h}\right)$$

$$= \nabla f_s(x) + O\left(\|R_s(x)\|E(x, h, \Phi)\right).$$

as asserted.
The proof of (25) is similar. □
Lemma 3.1 leads directly to a simple convergence result. which, for zero residual problems with only a few stencil failures, requires only that $E(x_n, h_n, \Phi)$ be bounded, a weaker condition than (7).

Theorem 2 *Let R satisfy (18) and assume that R' is Lipschitz continuous. Let $h_n \to 0$ and let $\{x_n\}$ be the implicit filtering sequence. Assume that all but finitely many calls to* `fdgauss` *return with stencil failure or*

$$\|(\nabla_{h_n} R(x_n))^T R(x_n)\| < \tau h_n, \qquad (26)$$

that the model Hessians $R(x_n)^T R(x_n)$ are nonsingular, and that the model Hessians and their inverses are uniformly bounded. Then if

$$\lim_{n \to \infty} E(x_n, h_n, \Phi) = 0 \qquad (27)$$

then any limit point of x_n is a critical point of f. If, moreover, all but finitely many calls to `fdgauss` *return with (26), then (27) can be replaced by*

$$\lim_{n \to \infty} \|R_s(x_n)\| E(x_n, h_n, \Phi) = 0 \qquad (28)$$

Proof. The convergence assumption (27) requires that

$$\|\Phi\|_{S(x_n, h_n)}/h_n \to 0.$$

In view of (19), this is equivalent to (7) if (22) holds. Hence the first assertion of the theorem is simply a restatement of Theorem 1.

If the finite-difference Gauss-Newton iteration terminates all but finitely many times with (26), then

$$\|\nabla f_s(x_n)\| \leq \tau h_n + O(\|R_s(x_n)\| E(x_n, h_n, \Phi))$$

by (24). This completes the proof. □

3.3   Local Convergence

To analyze the local convergence behavior of the IFGN iteration, we must assume that the model Hessians are well conditioned and bounded. Let x^* be a local minimizer of $f_s(x) = R_s^T(x) R_s(x)$ for which the standard assumptions for convergence of the Gauss-Newton iteration

$$x_+^{GN} = x_c - ((\nabla_h R_s(x_c))^T \nabla_h R_s(x_c))^{-1} R_s'(x_c) R_s(x_c),$$

hold (smoothness, nonsingularity of the model Hessian, sufficiently small residual).

To quantify this we will assume:

Assumption 3.1 *There is $\rho_0 > 0$ such that*

- R_s *is Lipschitz continuously differentiable in the set*

$$\mathcal{D} = \{x \mid \|x - x^*\| \leq \rho_0\},$$

- *the Gauss-Newton model Hessian $R_s'(x)^T R_s'(x)$ and its inverse are uniformly bounded in $\mathcal{D}$, and*

- *there are $r_{GN} \in (0,1)$ and $C_{GN} > 1$ such that for all $x_c \in \mathcal{D}$,*

$$\|e_+^{GN}\| \le C_{GN}(\|e_c\|^2 + \|R_s(x^*)\|\|e_c\|) \le r_{GN}\|e_c\|. \qquad (29)$$

As is standard, we let $e = x - x^*$ for $x \in R^N$, with the iteration index for e being inherited from the one for x.

Lemma 3.2 *Let R be given by (18). Let (22) and Assumption 3.1 hold and let $x_c \in \mathcal{D}$. Then if*

$$\sup_{x \in \mathcal{D}} E(x, h, \Phi)$$

is sufficiently small, the IFGN model Hessian $(\nabla_h R(x_c))^T \nabla_h R(x_c)$ is nonsingular. Moreover, if

$$x_+ = x_c - ((\nabla_h R(x_c))^T \nabla_h R(x_c))^{-1} \nabla_h R(x_c)^T R(x_c)$$

then

$$\|e_+\| = \|e_+^{GN}\| + O(\|R_s(x_c)\|E(x_c, h, \Phi)). \qquad (30)$$

Proof. Let $x_c \in \mathcal{D}$. Assumption 3.1 and (25) imply that

$$\|(R_s'(x_c)^T R_s'(x_c))^{-1} - ((\nabla_h R(x_c))^T \nabla_h R(x_c))^{-1}\| = O(E(x_c, h, \Phi)). \qquad (31)$$

Now,

$$x_+ = x_+^{GN} + E_H \nabla f_s(x_c) + (R'(x_c)^T R'(x_c))^{-1} E_g$$

where

$$E_H = (R'(x_c)^T R'(x_c))^{-1} - ((\nabla_h R(x_c))^T \nabla_h R(x_c))^{-1}$$

and

$$E_g = \nabla f_s(x) - (\nabla_h R(x))^T R(x).$$

Since $\nabla f_s(x_c) = O(\|e_c\|)$, we apply (31) to obtain

$$E_H \nabla f_s(x_c) = O(\|R_s(x_c)\|E(x_c, h, \Phi)).$$

The conclusion now follows from (22) and (24). $\square$

Theorem 3 *Let R be given by (18). Let (22) and Assumption 3.1 hold. Let $x_0 \in \mathcal{D}$. Let $h_n \to 0$. Assume that the implicit filtering sequence $\{x_n\} \subset \mathcal{D}$ and that the line search fails only finitely many times. Then if (27) holds then $x_n \to x^*$.*

3.4   Rates of Convergence

To obtain rates of convergence we must make stronger assumptions on Φ, on the scales, and on the convergence rates of the Gauss-Newton iteration for the smooth problem. We must augment (29) with a lower bound that states that the Gauss-Newton iteration for R_s converges no faster than the standard Gauss-Newton convergence rate. This latter assumption is a nondegeneracy condition on R'' and is needed for the superlinear convergence results.

Assumption 3.2 *There are $p \in (0,1]$ and $C_p > 0$ such that*

$$\|\Phi(x_c)\| \le C_p \|e_c\|^{2+2p} \tag{32}$$

for all $x_c \in \mathcal{D}$. In addition to (29),

$$C_{GN}^{-1}(\|e_c\|^2 + \|R_s(x^*)\|\|e_c\|) \le \|e_+^{GN}\| \tag{33}$$

for all $x_c \in \mathcal{D}$.

Lemma 3.3 *Let Assumptions 3.1 and 3.2 hold. Then if x_c is sufficiently near x^* and*

$$C_h^{-1}\|e_c\|^{1+p} \le h_c \le C_h\|e_c\|^{(1+p)/2} \tag{34}$$

then there are $r_{GN} < r < 1$ and $C > 1$ such that

$$C^{-1}\|e_+^{GN}\| \le \|e_+\| \le C\|e_+^{GN}\| \le r\|e_c\|, \tag{35}$$

Proof. We will show that

$$\|R_s(x_c)\|E(x_c, h_c, \Phi) = o(\|e_+^{GN}\|) \tag{36}$$

for x_c near x^*. The result will follow from Lemma 3.2 for x_c sufficiently near x^*.

Lemma 3.3 and (32) imply that

$$E(x_c, h, \Phi) = O(\|e_c\|^{1+p}).$$

We consider two cases. If the smooth problem is a zero residual problem ($R_s(x^*) = 0$), then

$$\|R_s(x_c)\|E(x_c, h_c, \Phi) = O(\|e_c\|^{2+p}).$$

In this case, (33) implies (36).

If $R_s(x^*) \neq 0$, then

$$\|R_s(x_c)\|E(x_c, h_c, \Phi) = O(\|e_c\|^{1+p}).$$

However, in that case (33) implies that

$$\|e_+^{GN}\| \geq C_{GN}^{-1}\|R_s(x^*)\|\|e_c\|$$

and (36) holds. This completes the proof. □

In order to apply Lemma 3.3 we need to make sure that (34) holds throughout the iteration. The most direct way to do this is to update h_n with an analog of (16)

$$h_{n+1} = \|(\nabla_{h_n} R(x_{n+1}))^T R(x_{n+1})\|^{1+p}. \tag{37}$$

Theorem 4 *Let Assumptions 3.1 and 3.2 hold. The if x_0 is sufficiently near x^*,*

$$\|\nabla f_s(x_0)\|^{1+p}/2 \leq h_0 \leq 2\|\nabla f_s(x_0)\|^{(1+p)/2}, \tag{38}$$

and the implicit filtering sequence is defined by Algorithm IFGN *and (37), then $x_n \to x^*$ and*

$$C^{-1}\|e_{n+1}^{GN}\| \leq \|e_{n+1}\| \leq C\|e_{n+1}^{GN}\| \leq r\|e_n\|, \tag{39}$$

for all $n \geq 0$.

Proof. Our assumptions imply that (38) is equivalent to (34) with, for example,

$$C_h = \sup_{x \in \mathcal{D}} \|\nabla^2 f_x(x)\|.$$

Hence, proceeding by induction, we need only show that

$$\|\nabla f_s(x_n)\|^{1+p}/2 \leq h_n \leq 2\|\nabla f_s(x_n)\|^{(1+p)/2} \tag{40}$$

for $n > 0$.

By (24), if h_n satisfies (40), then

$$\begin{aligned}
h_{n+1} &= (\|\nabla f_s(x_{n+1})\| + \|R_s(x_{n+1})\|E(x_{n+1}, h_n, \Phi))^{1+p} \\
&= \left(\|\nabla f_s(x_{n+1})\| + o(\|e_{n+1}^{GN}\|)\right)^{1+p} \\
&= \|\nabla f_s(x_{n+1})\|^{1+p} + o(\|\nabla R_s(x_{n+1})\|^{1+p}).
\end{aligned}$$

Hence h_{n+1} satisfies (40) for x_0 sufficiently near x^*. □

Remark: Theorem 4 says that the local convergence of IFGN is as good asymptotically as Gauss-Newton, if one counts only nonlinear iterations. For zero residual problems, one need not reduce the scales as rapidly. If we replace (34) by

$$C_h^{-1}\|e_c\|^{1+p} \le h_c \tag{41}$$

then (35) becomes

$$\|e_+\| \le C\|e_+^{GN}\| + O(\|R_s(x_c)\|(h_c^2 + \|e_c\|^{1+p})). \tag{42}$$

This will imply superlinear convergence for zero residual problems for which (22) and (32) hold if $h_n \to 0$. The computations in § 4 illustrate this.

4.   Numerical Example

We report on the performance of IFGN on a parameter identification problem taken from [24, 7, 3]. Here $N = 2$ and $M = 100$. The problem is to identify the stiffness k and damping c in a harmonic oscillator so that the numerical solution of

$$u'' + cu' + ku = 0; u(0) = u_0, u'(0) = 0$$

best fits the data in the least squares sense.

For this example the data are values of the exact solution at $t_i = i/100$ for $1 \le i \le 100$. The numerical solution was computed with the MATLAB ODE15s integrator [33].

We compare three variations of implicit filtering, IFGN with a fixed sequence of scales and an adaptive sequence that attempts to satisfy (37), and a version of the implicit filtering/BFGS algorithm from [24, 7] that has been modified to use adaptive scales. In all three we limit the optimization to a budget of 100 calls to the function. This does not mean that an iteration is terminated before completion, rather we monitor the number of function evaluations after a call to the finite difference optimizer returns and stop the optimization if the number of function evaluations has exceeded the budget after the completion of the iteration.

For all the computations the initial iterate is $(c, k) = (2, 3)$. The sequence of scales used in the examples is

$$h_n^{(1)} = 2^{-n}, \; n = 4, \ldots, 13. \tag{43}$$

Following [7], we implement adaptive scales based on a scaled and safe-guarded form of (37),

$$h_{n+1}^{(2)} = \max\left(\min\left[h_{n+1}^{(1)}, \left(\frac{\|(\nabla_{h_n} R(x_{n+1})^T R(x_{n+1})\|}{\|(\nabla_{h_0} R(x_0)^T R(x_0)\|}\right)^{1+p}\right], h_{min}\right)$$
$$(44)$$

where $p = 1/2$ and $h_{min} = 10^{-5}$. h_{min} is roughly the cube root of machine roundoff and is the optimal choice of h for a central difference.

In the examples the line search strategy is to reduce the step by half if the sufficient decrease condition (either (11) for implicit filtering or (20) for IFGN) fails. Within both algorithms `fdquasi` and `fdguass`, $amax = 10$ and $pmax = 100$.
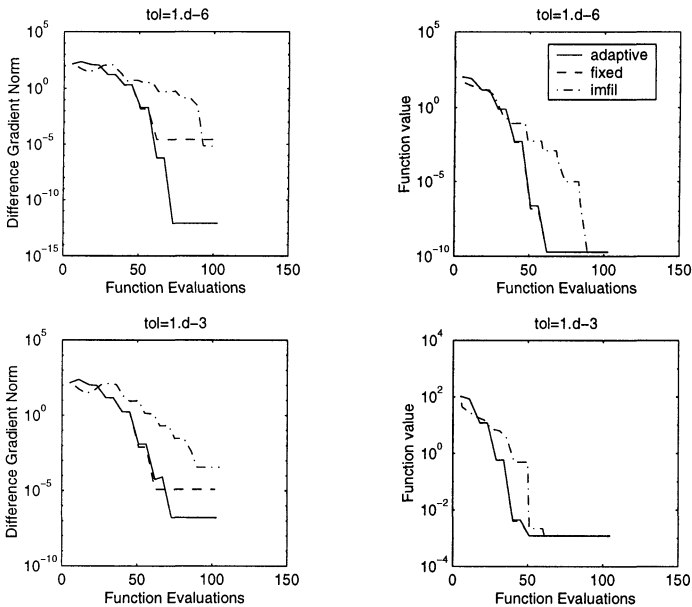

Figure 2. Parameter Identification Example

In Figure 4 we plot the norm of the difference gradient and the size of the function for the three variations of implicit filtering and two values, 10^{-6} and 10^{-3}, of the tolerance given to ODE15s. One can see that the two variations of IFGN did substantially better than an implementation of implicit filtering that did not exploit the least squares structure. A more subtle difference, explained by the remark at the end of § 3, is that while the use adaptive scales made no visible difference in IFGN's ability to reduce the residual (the curves overlap, indicating that the rate of convergence for both methods is equally fast, *i. e.* superlinear), it did

make the difference gradient a much better indicator of the progress of the optimization (the scales that are reduced most rapidly produce more accurate gradients).

We see similar behavior for a small, but non-zero, residual problem. In Figure 2 we show the results from the parameter ID problem with uniformly distributed random numbers in the interval $[0, 10^{-4}]$ added to the data. The gradients behave in the same way as in the experiment with exact data, while the limiting function values reflect the non-zero residual in the high-accuracy simulation. In the low-accuracy simulation, the tolerances given to the integrator are smaller than the noise in the data, so the figures are almost identical to the one for the noise-free case.
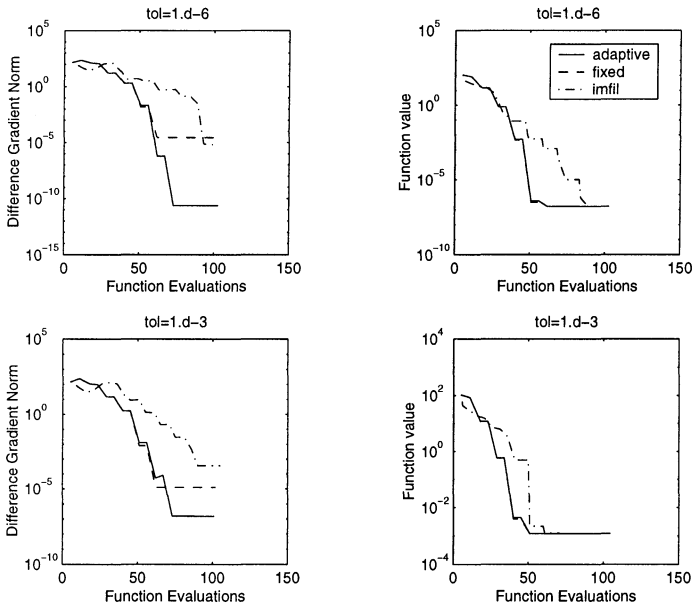

Figure 3.   Parameter Identification Example; Random Noise in Data

References

[1] C. AUDET AND J. E. DENNIS, *Analysis of generalized pattern searches.* submitted for publication, 2000.

[2] ———, *A pattern search filter method for nonlinear programming without derivatives.* submitted for publication, 2000.

[3] H. T. BANKS AND H. T. TRAN, *Mathematical and experimental modeling of physical processes.* Department of Mathematics, North Carolina State University, unpublished lecture notes for Mathematics 573-4, 1997.

[4] A. BATTERMANN, J. M. GABLONSKY, A. PATRICK, C. T. KELLEY, T. COFFEY, K. KAVANAGH, AND C. T. MILLER, *Solution of a groundwater control problem with implicit filtering*, Optimization and Engineering, 3 (2002), pp. 189–199.

[5] D. M. BORTZ AND C. T. KELLEY, *The simplex gradient and noisy optimization problems*, in Computational Methods in Optimal Design and Control, J. T. Borggaard, J. Burns, E. Cliff, and S. Schreck, eds., vol. 24 of Progress in Systems and Control Theory, Birkhäuser, Boston, 1998, pp. 77–90.

[6] C. G. BROYDEN, *A new double-rank minimization algorithm*, AMS Notices, 16 (1969), p. 670.

[7] T. D. CHOI AND C. T. KELLEY, *Superlinear convergence and implicit filtering*, SIAM J. Optim., 10 (2000), pp. 1149–1162.

[8] A. R. CONN, K. SCHEINBERG, AND P. L. TOINT, *On the convergence of derivative-free methods for unconstrained optimization*, in Approximation Theory and Optimization: Tributes to M. J. D. Powell, A. Iserles and M. Buhmann, eds., Cambridge, U.K., 1997, Cambridge University Press, pp. 83–108.

[9] ——, *Recent progress in unconstrained optimization without derivatives*, Math. Prog. Ser. B, 79 (1997), pp. 397–414.

[10] J. W. DAVID, C. T. KELLEY, AND C. Y. CHENG, *Use of an implicit filtering algorithm for mechanical system parameter identification*, 1996. SAE Paper 960358, 1996 SAE International Congress and Exposition Conference Proceedings, Modeling of CI and SI Engines, pp. 189–194, Society of Automotive Engineers, Washington, DC.

[11] 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.

[12] J. E. DENNIS AND V. TORCZON, *Direct search methods on parallel machines*, SIAM J. Optim., 1 (1991), pp. 448 – 474.

[13] R. FLETCHER, *A new approach to variable metric methods*, Comput. J., 13 (1970), pp. 317–322.

[14] J. M. GABLONSKY, *Modifications of the DIRECT Algorithm*, PhD thesis, North Carolina State University, Raleigh, North Carolina, 2001.

[15] J. M. GABLONSKY AND C. T. KELLEY, *A locally-biased form of the DIRECT algorithm*, Journal of Global Optimization, 21 (2001), pp. 27–37.

[16] P. GILMORE, *An Algorithm for Optimizing Functions with Multiple Minima*, PhD thesis, North Carolina State University, Raleigh, North Carolina, 1993.

[17] P. GILMORE AND C. T. KELLEY, *An implicit filtering algorithm for optimization of functions with many local minima*, SIAM J. Optim., 5 (1995), pp. 269–285.

[18] D. GOLDFARB, *A family of variable metric methods derived by variational means*, Math. Comp., 24 (1970), pp. 23–26.

[19] J. H. HOLLAND, *Genetic algorithms and the optimal allocation of trials*, SIAM J. Comput., 2 (1973).

[20] R. HOOKE AND T. A. JEEVES, *'Direct search' solution of numerical and statistical problems*, Journal of the Association for Computing Machinery, 8 (1961), pp. 212–229.

[21] D. R. JONES, *The DIRECT global optimization algorithm.* to appear in the Encylopedia of Optimization, 1999.

[22] D. R. JONES, C. C. PERTTUNEN, AND B. E. STUCKMAN, *Lipschitzian optimization without the Lipschitz constant*, J. Optim. Theory Appl., 79 (1993), pp. 157–181.

[23] C. T. KELLEY, *Detection and remediation of stagnation in the Nelder-Mead algorithm using a sufficient decrease condition*, SIAM J. Optim., 10 (1999), pp. 43–55.

[24] ——, *Iterative Methods for Optimization*, no. 18 in Frontiers in Applied Mathematics, SIAM, Philadelphia, 1999.

[25] S. KIRKPATRICK, C. D. GEDDAT, AND M. P. VECCHI, *Optimization by simulated annealing*, Science, 220 (1983), pp. 671–680.

[26] J. C. LAGARIAS, J. A. REEDS, M. H. WRIGHT, AND P. E. WRIGHT, *Convergence properties of the Nelder-Mead simplex algorithm in low dimensions*, SIAM J. Optim., 9 (1998), pp. 112–147.

[27] R. M. LEWIS AND V. TORCZON, *Rank ordering and positive bases in pattern search algorithms*, Tech. Rep. 96-71, Institute for Computer Applications in Science and Engineering, December 1996.

[28] S. LUCIDI AND M. SCIANDRONE, *On the global convergence of derivative free methods for unconstrained optimization.* Reprint, Università di Roma "La Sapienza", Dipartimento di Informatica e Sistemistica, 1997.

[29] ——, *A derivative-free algorithm for bound constrained optimization.* Reprint, Instituto di Analisi dei Sistemi ed Informatica, Consiglio Nazionale delle Richerche, 1999.

[30] J. J. MORÉ, B. S. GARBOW, AND K. E. HILLSTROM, *User guide for MINPACK-1*, Tech. Rep. ANL-80-74, Argonne National Laboratory, 1980.

[31] J. A. NELDER AND R. MEAD, *A simplex method for function minimization*, Comput. J., 7 (1965), pp. 308–313.

[32] J. NOCEDAL AND S. J. WRIGHT, *Numerical Optimization*, Springer, New York, 1999.

[33] L. F. SHAMPINE AND M. W. REICHELT, *The MATLAB ODE suite*, SIAM J. Sci. Comput., 18 (1997), pp. 1–22.

[34] D. F. SHANNO, *Conditioning of quasi-Newton methods for function minimization*, Math. Comp., 24 (1970), pp. 647–657.

[35] M. SRINIVAS AND L. M. PATNAIK, *Genetic algorithms: a survey*, Computer, 27 (1994), pp. 17–27.

[36] D. STONEKING, G. BILBRO, R. TREW, P. GILMORE, AND C. T. KELLEY, *Yield optimization using a GaAs process simulator coupled to a physical device model*, IEEE Transactions on Microwave Theory and Techniques, 40 (1992), pp. 1353–1363.

[37] D. E. STONEKING, G. L. BILBRO, R. J. TREW, P. GILMORE, AND C. T. KELLEY, *Yield optimization using a GaAs process simulator coupled to a physical device model*, in Proceedings IEEE/Cornell Conference on Advanced Concepts in High Speed Devices and Circuits, IEEE, 1991, pp. 374–383.

[38] V. TORCZON, *Multidirectional Search*, PhD thesis, Rice University, Houston, Texas, 1989.

[39] ——, *On the convergence of the multidimensional direct search*, SIAM J. Optim., 1 (1991), pp. 123–145.

[40] ——, *On the convergence of pattern search algorithms*, SIAM J. Optim., 7 (1997), pp. 1–25.

[41] P. VAN LAARHOVEN AND E. AARTS, *Simulated annealing, theory and practice*, Kluwer, Dordrecht, 1987.

[42] T. A. WINSLOW, R. J. TREW, P. GILMORE, AND C. T. KELLEY, *Doping profiles for optimum class B performance of GaAs mesfet amplifiers*, in Proceedings IEEE/Cornell Conference on Advanced Concepts in High Speed Devices and Circuits, IEEE, 1991, pp. 188–197.

[43] ——, *Simulated performance optimization of GaAs MESFET amplifiers*, in Proceedings IEEE/Cornell Conference on Advanced Concepts in High Speed Devices and Circuits, IEEE, 1991, pp. 393–402.

[44] S. K. ZAVRIEV, *On the global optimization properties of finite-difference local descent algorithms*, J. Global Optimization, 3 (1993), pp. 67–78.