A Brief Introduction to Implicit Filtering

C. T. Kelley*

September 14, 2002

Abstract

In this short note we explain the fundamental ideas behind the implicit filtering method. We take the view that implicit filtering is a natural extension of coordinate search and show how a simple convergence result can be derived from that point of view. We point to the literature for convergence results and experience with the algorithm in practice. The URLs for software are listed at the end of the paper.

1 Introduction

Implicit filtering [11,13] is a projected quasi-Newton iteration which uses difference gradients, reducing the difference increment as the optimization progresses. The idea is that a large difference increment will no be sensitive to low-amplitude high-frequency oscillations ("filter them out") and will respond to the large-scale features of the objective function.

The objectives of this paper are to describe the implicit filtering method, put it in the the context of extensions of the coordinate search method, and use that connection to show how one proves a simple convergence result.

Most of the details for the material in this paper can be found in [13], which contains convergence theory implicit filtering and for several related methods. We will discuss unconstrained optimization in this paper. We refer the reader to [3, 6, 8, 11, 13, 16] for descriptions of problems with constraints and how those constraints are handled.

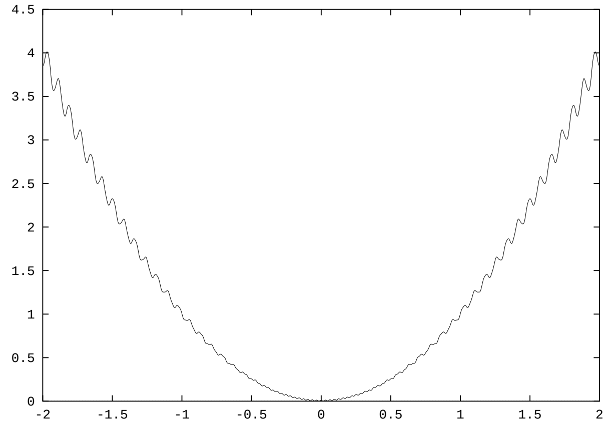
We begin with a description of the types of problems we are trying to solve. We show how coordinate search can be applied to such problems and sketch the convergence analysis. Finally, we describe implicit filtering and its convergence theory.

2 The Problem

Our goal in this paper is to minimize a function f defined on all of R^N. A simple one-dimensional example of the type of function we have in mind is plotted in Figure 1

*North Carolina State University, Center for Research in Scientific Computation and Department of Mathematics, Box 8205, Raleigh, N. C. 27695-8205, USA (Tim_Kelley@ncsu.edu). The research of this author was supported in part by National Science Foundation grants DMS-0070641 and DMS-0112542.

Figure 1: Objective function of one variable



The key features of the objective function are that the large-scale features are those of a very simple and easy to optimize function. However, a gradient-based method would do very poorly on this function because of the local minima. Implicit filtering is designed to “filter” the high-frequency, low-amplitude noise in the function and follow the large-scale features.

Implicit filtering and related algorithms are not designed for problems that are easily solved by gradient-based methods, but rather for problems that have many local minima. On the other hand, implicit filtering is not a method for global optimization. The large-scale simplicity that one sees in Figure 1 is important both for the theory and, to a lesser degree, the practical applications. Problems without such a large-scale structure, such as those that arise in molecular confirmation problems [5,19], are best solved with true global optimization algorithms.

3 Coordinate Search

Coordinate search begins with an current approximation to the optimal point x and a stepsize of **scale** h. One then **samples** f at the $2N$ points in the **stencil**

$$S(x,h) = \{x \pm he_i\}$$

where e_i is the unit vector in the ith coordinate direction. There are two options at this point. Let $z^* \in S(x,h)$ be a point where f is minimized

$$f(z^*) = \min_{z \in S(x,h)} f(z).$$

If $f(z^*) < f(x)$, then we replace x with z^* and continue. If $f(z^*) \geq f(x)$, then we reduce h, and do not replace x.

We will refer to the condition for reducing h

$$f(x) \leq \min_{z \in S(x,h)} f(z), \tag{1}$$

as **stencil failure**, because the stencil has failed to produce a better function value.

So, after each sampling, either $f(x)$ or h has been reduced. If f is continuous and has bounded level sets, then one can reduce f at most finitely many times before reducing h. So, if $\{x_n\}$ and $\{h_n\}$ are the sequences of approximate minimizers and scales, $h_n \to 0$ if f has compact level sets.

You can say more if f is sufficiently smooth. The **stencil failure theorem** [4,13] says that if

- f is Lipschitz continuously differentiable and
- Eq (1) holds

then

$$\|\nabla f(x)\| = O(h).$$

Combining the stencil failure theorem with the observation that $h_n \to 0$ if f is continuous and has bounded level sets leads to a convergence theorem for coordinate search.

Theorem 3.1 *Let f be Lipschitz continuously differentiable and have bounded level sets. Let $\{x_n\}$ and $\{h_n\}$ be the sequences of approximate minimizers and scales. Then*

$$\nabla f(x_n) \to 0$$

and hence every limit point of $\{x_n\}$ is a critical point of f.

What does this have to do with functions like the one in Figure 1? The connection is that the analysis doesn't change much if the perturbations, be they highly oscillatory, nonsmooth, or even discontinuous, have small amplitude, particularly near the minimum.

To quantify the image from Figure 1 we assume that

$$f = f_s + \phi, \tag{2}$$

where f_s is a smooth, easy-to-minimize, function and ϕ, which we will call "noise", is a low-amplitude perturbation. We make no assumptions about ϕ other than about its size. In some applications [16], ϕ may have internal stochastic simulations and therefore not even be a function. We will assume that ϕ is a well-defined function in this paper.

We measure the size of the noise on the stencil as

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

The stencil failure theorem for this case says that

- f_s is Lipschitz continuously differentiable and
- Eq (1) holds

then

$$\|\nabla f(x)\| = O\left(h + \frac{\|\phi\|_{S(x,h)}}{h}\right). \tag{3}$$

This leads to a convergence theorem

Theorem 3.2 *Let f_s be Lipschitz continuously differentiable and have bounded level sets. Let $\{x_n\}$ and $\{h_n\}$ be the sequences of approximate minimizers and scales. Then if*

$$\lim_{n\to\infty}(h_n + h_n^{-1}\|\phi\|_{S(x,h_n)}) = 0 \tag{4}$$

then

$$\nabla f_s(x_n) \to 0$$

and hence every limit point of $\{x_n\}$ is a critical point of f_s.

Theorem 3.2 is an asymptotic result and (4) almost certainly does not hold in real life. However, the result describe the performance of algorithms in practice very well.

The analysis that leads to Theorem 3.2 can be applied directly to several extensions of coordinate search. One can reduce the effort in the search in several ways and still obtain convergence in the sense of Theorem 3.2:

- One does not have to examine the entire stencil once a better point is found [9,12,17,18].
- One can use stencils that are irregular and have fewer than $2N$ points [4, 13, 14].
- The ideas generalize to integer and categorical variables [1, 2].
- One can accelerate the convergence with quasi-Newton methods, which is what implicit filtering is for.

4 Implicit Filtering

Implicit filtering uses the same function values as as coordinate search and, similarly, reduces the scale when stencil failure takes place. However, if there's a better point in the stencil, implicit filtering tries to do better than simply take the best point in the stencil.

The method is simple, use the central difference gradient $\nabla_h f$ as the basisi for a quasi-Newton method. The algorithm **fdquasi** is listed below. The data are the initial iterate, the function, the scale h, and a few termination control parameters.

The important differences between **fdquasi** and a standard quasi-Newton method are:

- Stencil falure triggers an exit from the iteration.
- Failure of the line search is a real possibility, because $\nabla_h f$ may not be a descent direction for f.

Once **fdquasi** has terminated for a given h, one reduces h and restarts the iteration. The simplest form of implicit filtering is now easy to describe. The new data is the sequence of scales.

The basic convergence theorem follows directly from the one for coordinate search. Assume that (2) holds and that fdquasi terminates either with stencil failure or because $\|\nabla_h f\| \leq \tau h$. Either condition [13] implies that (3) holds, and then the proof of Theorem 3.2 is valid.

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

$p = 1$
while $p \le pmax$ and $\|\nabla_h f(x)\| \ge \tau h$ **do**
  compute f and $\nabla_h f$
  if (1) 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
  $x \leftarrow \mathcal{P}(x + \lambda d)$
  $p \leftarrow p + 1$
end while
if $p > pmax$ report **iteration count failure**

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

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

Theorem 4.1 *Let f satisfy (2) and let ∇f_s be Lipschitz continuous. Let $h_k \to 0$, $\{x_k\}$ be the implicit filtering sequence, and $S^k = S(x_k, h_k)$. Assume that fewer than amax backtracks and pmax iterations are taken for all but finitely many k. Then if*

$$\lim_{k\to\infty} (h_k + h_k^{-1} \max_{z \in S^k} |\phi(z)|) = 0 \tag{5}$$

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

Theorem 4.1 is a satisfying asymptotic result, but the practical success of the algorithm depends on the quasi-Newton model Hessian. We refer the reader to [7,11,13] for theory and experiments that show how the quasi-Newton model Hessian performs. While the theoretical results are technical and not predictive, they do describe the performance in practice well.

5 Software

My research group supports a MATLAB implementation of implicit filtering for unconstrained optimization that is, aside from a few heuristic improvements, exactly **imfilter**. We also support a FORTRAN code **IFFCO** for bound constrained problems.

The FORTRAN code supports message-passing parallelism in both PVM [10] and MPI [15].

All of these codes can be found at the IFFCO web page:

http://www4.ncsu.edu/~ctk/iffco.html
and the web page for the software collection for [13]
http://www4.ncsu.edu/~ctk/matlab_darts.html

Acknowledgments

This paper is based on part of the presentation I gave in August, 2002, at *The Second International Workshop on Optimization and Control with Applications*, which was held in Huangshan, China. I'd like to thank the organizers of that excellent meeting for their support. I'd also like to thank Alex Rubinov for the invitation to publish this note in ORB and get some publicity for implicit filtering.

References

[1] C. Audet and J. E. Dennis, *Analysis of generalized pattern searches.* submitted for publication, 2000.

[2] ——, *Pattern search algorithms for mixed variable programming*, SIAM J. Optim., 11 (2000), pp. 573–594.

[3] 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*, Tech. Rep. CRSC-TR00-30, North Carolina State University, Center for Research in Scientific Computation, December 2000. To appear in Optimization and Engineering.

[4] 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.

[5] R. H. Byrd, T. Derby, E. Eskow, K. P. B. Oldenkamp, and R. B. Schnabel, *A new stochastic/perturbation method for large-scale global optimization and its application to water cluster problems*, in Large Scale Optimization: State of the Art, W. W. Hager, D. W. Hearn, and P. Pardalos, eds., Boston, 1994, Kluwer Academic Publishers B.V., pp. 68–81.

[6] R. Carter, J. M. Gablonsky, A. Patrick, C. T. Kelley, and O. J. Eslinger, *Algorithms for noisy problems in gas transmission pipeline optimization*, Optimization and Engineering, 2 (2001), pp. 139–157.

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

[8] 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.

[9] J. E. Dennis and V. Torczon, *Direct search methods on parallel machines*, SIAM J. Optim., 1 (1991), pp. 448 – 474.

[10] A. Geist, A. Beguelin, J. Dongarra, W. J. R. Manchek, and V. Sunderam, *PVM: Parallel Virtual Machine A Users' Guide and Tutorial for Networked Parallel Computing*, MIT Press, Cambridge, MA, 1994.

[11] 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.

[12] 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.

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

[14] 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.

[15] M. Snir, S. Otto, S. Huss-Lederman, D. Walker, and J. Dongarra, *MPI: the complete reference*, Scientific and Engineering Computation, MIT Press, 1996.

[16] 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.

[17] V. Torczon, *Multidirectional Search*, PhD thesis, Rice University, Houston, Texas, 1989.

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

[19] Z. Zou, R. H. Byrd, and R. B. Schnabel, *A stochastic/perturbation global optimization algorithm for distance geometry problems*, J. Global Opt., 11 (1997), pp. 91–105.