ON THE VARIOUS GENERALISATIONS OF OPTIMISATION ALGORITHMS TO MANIFOLDS

Jonathan H. Manton

Department of Electrical and Electronic Engineering
The University of Melbourne, Victoria 3010, Australia.
j.manton@ee.mu.oz.au

ABSTRACT

Numerical minimisation of a cost function on Euclidean space is a well studied problem. Sometimes though, the most appropriate formulation of an optimisation problem is not in Euclidean space, but rather, it is required to minimise a cost function defined on a (not necessarily Riemannian) manifold. A natural question is how best to generalise existing optimisation algorithms to the manifold setting. This paper reviews and draws connections between existing techniques. It also proposes several new ideas. These ideas are in their infancy; they are intended to motivate further research in this area.

1. INTRODUCTION

Practical problems requiring the numerical optimisation of a cost function defined on a manifold can be found in [3, 4, 14, 17, 21, 23, 26, 27], to name just a few. Numerical optimisation on manifolds though is a relatively recent research area; while two early papers are [8, 13], its appearance in the signal processing literature, for instance, is only recent [7, 18].

Since the "optimisation on manifold" problem is a generalisation of the "optimisation on Euclidean space" problem, the natural question is how to generalise the wealth of algorithms for the latter problem to the former. The motivation for this paper is the fact that there is more than just a single sensible way of generalising standard optimisation algorithms, such as the Newton method [25], to the manifold setting.

The modest aims of this paper are to

- summarise and draw connections between numerical optimisation on manifold algorithms;
- state new ways of generalising Euclidean based algorithms to the manifold setting;
- give a qualitative assessment of when one generalisation should be preferred over another.

For brevity, the focus will be on Newton methods.

The remainder of this section states the requisite background ideas and notation. Section 2 explains why there is more than one way to generalise the Newton algorithm to the manifold setting. The ideas appearing there are also useful to keep in mind when comparing various algorithms. Section 3 outlines what is called here the Riemannian approach, and is the approach appearing in [7], for instance. A more general approach is presented in Section 4, which helps place the Riemannian approach in perspective. Section 5 propounds a novel generalisation which is in its infancy. Section 6 briefly discusses the case when the manifold has extra "symmetrical" structure while, for completeness, Section 7 indicates the challenges that lie ahead for extending quasi-Newton methods to manifolds.

Originally, a manifold M was defined to be a subset of $\mathbb{R}^n$ which was well behaved (smooth) in some sense, such as a sphere. Such a set $M \subset \mathbb{R}^n$ is now called a *concrete manifold*, to distinguish it from the apparently more general concept of an *abstract manifold*. While the Whitney embedding theorem implies that every abstract manifold can be realised as a concrete manifold $M \subset \mathbb{R}^n$ for sufficiently large n, it is often more appropriate (such as in this paper) to think of an abstract manifold as just that, rather than think of it as being embedded in $\mathbb{R}^n$.

The formal definition of a manifold is beyond the scope of this paper. (Standard references include [2, 10, 30].) The key property we require of a manifold[1] M though is that about any point $p \in M$, there exists a diffeomorphism between a neighbourhood of p in M and a neighbourhood of the origin in $\mathbb{R}^n$, where n is the dimension of the manifold. This is typically expressed by saying that locally a manifold looks like $\mathbb{R}^n$.

The specific problem this paper investigates is: Given a manifold M and a class of cost functions Ω, where every element $f \in \Omega$ is a smooth function from M to $\mathbb{R}$, develop

Invited paper presented at MTNS 2004 in KU Leuven, Belgium. This work was supported by the Australian Research Council and the ARC Special Research Centre for Ultra-Broadband Information Networks (CUBIN).

[1] Throughout, we implicitly assume all manifolds are connected.

a numerical algorithm which takes as input an element f of Ω and returns a point $p \in M$ which is a local minimum of the function $f : M \to \mathbb{R}$.

Optimisation on manifold problems arise in two ways. One way is that the constrained optimisation problem of minimising $f(x)$ subject to $g(x) = 0$, where $f : \mathbb{R}^n \to \mathbb{R}$ and $g : \mathbb{R}^n \to \mathbb{R}^m$, can naturally be thought of as an optimisation on manifold problem provided the constraint set $M = \{x \in \mathbb{R}^n : g(x) = 0\}$ forms a manifold. The second way is that a practical problem may already be of the correct form. For instance, the Grassmann manifold $\mathrm{Gr}(m, n)$ is the collection of all m dimensional subspaces of $\mathbb{R}^n$. In signal processing, a number of problems require finding a particular subspace (say, the signal subspace or the noise subspace), and can be posed as minimising a cost function defined on the Grassmann manifold. Note that the Grassmann manifold is an abstract manifold; although it can be embedded in $\mathbb{R}^p$ for sufficiently large p (as can all manifolds), such an embedding is not required in its definition.

Manifolds can be endowed with extra structure. If an inner product is assigned to each tangent plane in a sufficiently smooth way (called a metric structure), the manifold becomes a Riemannian manifold [12, 24]. A manifold which has a compatible group structure is called a Lie group [9, 30]. As will be seen later, this extra structure can suggest ways of generalising optimisation algorithms to such manifolds. It should not be forgotten though that *unless the class of cost functions is somehow related to this extra structure, there is no compelling reason to make use of this structure*.

2. GENERALISING THE NEWTON METHOD

Given a cost function $f : \mathbb{R}^n \to \mathbb{R}$, the Newton iteration is

$$x_{k+1} = x_k - [f''(x_k)]^{-1} f'(x_k) \tag{1}$$

where prime denotes differentiation.

The Newton iteration enjoys the following properties, the first of which is explained in greater detail below.

1. Invariance to affine transformations.
2. Convergence in a single iteration for quadratic functions.
3. Locally quadratic rate of convergence to a local minimum in general.
4. Requires only knowledge up to second order of the function at the current point.
5. Given by the iteration in (1).

Invariance to affine transformations means the following. Let $g : \mathbb{R}^n \to \mathbb{R}^n$ be an arbitrary invertible affine transformation (that is, $g(x) = Ax+b$ for some non-singular matrix A and vector b). If (1) generates the sequence $\{x_k\}_{k=0}^{\infty}$ when applied to f, then when applied to $f \circ g$, it generates the sequence $\left\{g^{-1}(x_k)\right\}_{k=0}^{\infty}$.

Although the author has not seen this stated anywhere else, it seems natural to define a generalisation of the Newton method to the manifold setting to be any algorithm on a manifold which preserves some of the above properties. Note that in general, it is not possible to preserve all the properties at once. Moreover, since affine transformations and quadratic functions are not defined on an arbitrary manifold, these properties need to be generalised to a manifold setting before an algorithm with these properties can be developed. Finally, note that quite clearly, Property 3 is an essential property if the algorithm is to be useful.

3. THE RIEMANNIAN APPROACH

One fairly typical approach to the optimisation on manifold problem, such as taken in [7], is first to endow the manifold with a metric structure, thus making it into a Riemannian manifold. Whereas the gradient and Hessian of a function on a manifold is not defined, these concepts make sense on a Riemannian manifold. Moreover, the idea of a straight line in Euclidean space is generalised to that of a geodesic on a Riemannian manifold.

The Newton iterate (1) can be thought of as computing x_{k+1} by starting at x_k and moving in a straight line in the direction given by the vector $-[f''(x_k)]^{-1} f'(x_k)$.

Based on this line of reasoning, the Newton method on a Riemannian manifold is to retain (1) as the iteration, but with $f'(\cdot)$ and $f''(\cdot)$ replaced by the Riemannian gradient and Hessian of the cost function, and with x_{k+1} computed by starting at x_k and moving along a geodesic rather than a straight line.

Referring to Section 2, this algorithm maintains Properties 3 and 4, and in the sense explained above, Property 5 too.

The author's opinion is that, from a theoretical perspective, this approach makes the most sense only when the cost function itself is somehow related to the Riemannian geometry. An example of this is if the cost function is defined in terms of the induced distance function on the Riemannian manifold, such as if it was required to find the center of mass [11] of a collection of points on a Riemannian manifold [20, 19].

The best situation is if the cost function is convex with respect to the Riemannian geometry [29], in which case this Riemannian generalisation of the Newton algorithm can be expected to be globally convergent.

In practice, if the cost function is not related to the Rie-

mannian geometry, other generalisations of the Newton algorithm may well be better. Here, an algorithm is defined to be better if it requires fewer (floating point) operations to achieve the same accuracy. One potential disadvantage of the Riemannian approach is that geodesics are often expensive to compute.

4. THE VARYING PARAMETERISATION APPROACH

On an arbitrary manifold, the only structure that is available is that about any point, the manifold locally looks like $\mathbb{R}^n$. Therefore, the following general framework was proposed in [18].

Given an n dimensional abstract manifold M, define beforehand, for every point $p \in M$, a distinguished parameterisation $\phi_p : \mathbb{R}^n \to M$ such that $\phi_p(0) = p$ and ϕ_p is a diffeomorphism onto its image. (In a more general setting, it is possible to take the range of ϕ_p to be an open subset of $\mathbb{R}^n$ but this is not considered here for simplicity.)

A Newton iterate on M can then be defined as follows: At the current point x_k, consider the local cost function $f \circ \phi_{x_k}$ about the origin. (Recall $\phi_{x_k}(0) = x_k$.) Since the local cost function maps from $\mathbb{R}^n$ to $\mathbb{R}$, a single Newton iterate (1) can be applied to it, thus moving from the origin to a new point, call it $z \in \mathbb{R}^n$. The next point x_{k+1} is defined to be $x_{k+1} = \phi_{x_k}(z)$.

To define a particular Newton method, it is necessary to specify the choice of parameterisations ϕ_p. Some possible choices are given below. First though, it is noted that the Riemannian approach of Section 3 is in fact a special case of this more general framework. It corresponds to choosing ϕ_p to be the inverse of normal coordinates about the point p.

If the manifold is an abstract manifold then the very definition of it may already be described in terms of parameterisations (or their inverses, coordinate charts), and hence there is often at least one sensible (if not natural) choice of parameterisations to use.

Alternatively, an abstract manifold (such as the Grassmann manifold) can be specified as the quotient of a concrete manifold with a group action. In such cases, if a sensible set of parameterisations for the concrete manifold is given, it is often possible to extract from each parameterisation a parameterisation for the abstract manifold. (Such an approach was done in [18], where a parameterisation of the Grassmann manifold was determined in a natural way from the parameterisation of the Stiefel manifold.)

Given a concrete manifold, one way of choosing the parameterisations ϕ_p is to make use of the Euclidean (or other) projection operator in the ambient space. (This does **not** result in an algorithm similar to the well-known "projection Newton methods" in the constrained optimisation literature, such as found in [25].) This works as follows. At a point p, consider the tangent space T_pM of the manifold M. Since $M \subset \mathbb{R}^n$ is a concrete manifold, its tangent space T_pM can be realised as a d-dimensional plane in $\mathbb{R}^n$, where d is the dimension of the manifold. Choose $\psi_p : \mathbb{R}^d \to \mathbb{R}^n$ to be a parameterisation of T_pM, with $\psi_p(0) = p$. Define $\pi : \mathbb{R}^n \to M \subset \mathbb{R}^n$ to be some projection operator, such as the Euclidean projection[2] operator, onto M. Then, define the parameterisations $\phi_p : \mathbb{R}^d \to M$ by $\phi_p(z) = \pi(\psi_p(z))$. This method was used in [18] to construct parameterisations for the Stiefel manifold.

The question of which family of parameterisations ϕ_p is best is a fundamental question. While finding the best parameterisation appears intractable for all but trivial examples, it is straightforward to explain intuitively which parameterisations are good. Indeed, the Euclidean Newton algorithm converges in a single iteration when applied to a quadratic cost function, and one can expect that it continues to converge quickly when applied to approximately quadratic functions, that is, functions with negligible coefficients in the third and higher order terms of their Taylor series. (This can be made precise by using the results in [1].) Therefore, given the cost function f, the best ϕ_p to use is the one for which the local cost function $f \circ \phi_p$ is as close to quadratic as possible.

Usually, trying to choose ϕ_p to make $f \circ \phi_p$ approximately quadratic is harder than solving the original optimisation problem. Moreover, it cannot be done offline because, as stated in Section 1, the optimisation algorithm doesn't know the function f beforehand, it only knows that f comes from a class Ω. In general, there will not exist ϕ_p such that $f \circ \phi_p$ is approximately quadratic for all $f \in \Omega$.

Therefore, given that ϕ_p cannot be chosen simply on the basis of hoping to achieve fast global convergence, a more pragmatic approach is to choose ϕ_p so as to minimise the computational effort per iteration. This computational effort includes evaluating $f \circ \phi_p$ and its first two derivatives, and often, this computational burden is quite high.

To summarise, the choice of ϕ_p depends on the manifold itself as well as on the class of cost functions Ω likely to be encountered. Unless Ω has special properties which can be exploited, it is recommended to choose ϕ_p to try to minimise the computational burden per iteration.

Referring to Section 2, this general framework yields algorithms satisfying Properties 3 and 4. (The author hopes to publish in the near future a "universal convergence proof" which shows that provided the parameterisations ϕ_p vary smoothly with p, the resulting Newton algorithm achieves local quadratic convergence.)

Although the framework described in this section is gen-

[2] For the Euclidean projection to be well-defined everywhere, the manifold M should be closed. Now, if M is not closed, the original optimisation problem may not have a solution!

eral enough to encompass all the published optimisation on manifold algorithms the author is aware of, it does not necessarily include the new approach suggested in the next section.

5. THE FUNCTION MATCHING APPROACH

Whereas all useful Newton algorithms have an asymptotically quadratic rate of convergence, their global behaviour is often hard to determine. In special cases though, such as if the cost function is convex, the Newton method is known to be globally convergent. This section introduces an idea which is aimed at improving the global properties of a Newton algorithm for non-convex problems.

The Newton iterate (1) converges in a single iteration if the cost function f is quadratic (Property 2 in Section 2) by design; (1) is derived by approximating f about the point x_k by a quadratic function and then moving to the critical point of that function.

Property 2 cannot carry over directly to arbitrary manifolds because it is not possible to define a quadratic function in general. However, the following generalisation of Property 2 can be carried over.

Let Θ denote a set of functions on the manifold. In the Euclidean case, Θ could be the set of all quadratic functions. The generalised Newton iterate is defined roughly as follows: When at the point x_k, first find a function $g \in \Theta$ which "best" approximates the cost function and then set x_{k+1} to be the minimum of g. To recover the standard Newton iterate, "best" is defined as requiring the second order Taylor series about the point x_k of the cost function and of the approximant g to match.

On an arbitrary manifold, there are two choices to be made; the choice of approximants Θ and the method used to find the best approximant about any given point. How to do this is the subject of current research and will be reported on elsewhere. The author currently believes though that it is the generalisation of Property 2 in Section 2 which is the key to designing Newton algorithms with desirable global performance.

6. LIE GROUPS AND HOMOGENEOUS SPACES

In the Euclidean setting, the affine invariance property of the Newton method (Property 1 in Section 2) is desirable because Euclidean space "looks the same" regardless of what affine transformation has been applied to it. In other words, changing the coordinate system in an affine way should not affect the performance of the algorithm. (Here, of course, it is assumed that the cost function is somehow related to the Euclidean structure, such as if the cost function represents some real life quantity. Otherwise, as stated at the end of Section 1, there is no compelling reason to make use of the Euclidean structure.)

Lie groups and homogeneous spaces [2] are two types of manifolds with extra structure. This extra structure means the space about any two points looks the same, at least locally. Therefore, it is sensible to consider generalised Newton algorithms which are invariant to the appropriate structure on these spaces. This has been done in [15, 16, 22].

It is remarked though that in all cases the author is aware of, the resulting Newton algorithm can be written in the form described in Section 4. Therefore, designing Newton algorithms with certain invariance properties is equivalent to choosing a set of parameterisations ϕ_p which preserve the symmetrical structure. (Roughly speaking, ϕ_p must map the action of the affine group on $\mathbb{R}^n$ onto the relevant group action on the manifold, the relevant group action being the one we want the Newton method to be invariant to.)

7. QUASI-NEWTON METHODS

Quasi-Newton methods [5, 6, 28] build up an approximation to the Hessian over successive iterations. They are therefore computationally less expensive than a Newton method, and have super-linear convergence rather than quadratic convergence locally.

Whereas the Newton iterate (1) only uses information at the point x_k to calculate x_{k+1}, a quasi-Newton method implicitly uses information at $x_{k-1}, x_{k-2}, \cdots$ too. In the Euclidean setting, if the Hessian of a function at a point x is H, a reasonable approximation to the Hessian of the function at a neighbouring point is again H. Because a manifold twists and turns though, this is not true for a general manifold. (Recall too that a Riemannian structure is needed before the Hessian of a function can be defined.)

Quasi-Newton methods have been extended to Riemannian manifolds in [8]. On Riemannian manifolds, there is such a thing as parallel transport, and this is what is used to adjust information obtained at $x_{k-1}, x_{k-2}, \cdots$ so that it becomes relevant at x_k.

However, as stated earlier, unless the cost function is related to the Riemmanian geometry, it is usually preferable to develop an optimisation algorithm which uses only the manifold structure and not the extra Riemannian structure. While there are a number of ways this can be done, the author is not aware of any published papers in this area.

8. CONCLUSION

Comparisons between optimisation on manifold techniques in the literature have been made and several new ideas propounded. It is hoped these new ideas will motivate further research in this area.

9. REFERENCES

[1] L. Blum, F. Cucker, M. Shub, and S. Smale. *Complexity and Real Computation*. Springer, 1997.

[2] W. M. Boothby. *An Introduction to Differentiable Manifolds and Riemannian Geometry*. Academic Press, second edition, 1986.

[3] R. W. Brockett. Dynamical systems that sort lists, diagonalise matrices, and solve linear programming problems. *Linear Algebra Appl.*, 146:79–91, 1991.

[4] F. De Bruyne, B. D. O. Anderson, M. Gevers, and N. Linard. Gradient expressions for a closed-loop identification scheme with a tailor-made parametrization. *Automatica*, 35(11):1867–1871, 1999.

[5] J. E. Dennis, Jr. and J. J. More. Quasi-Newton methods, motivation and theory. *SIAM Review*, 19(1):46–89, 1977.

[6] J. E. Dennis, Jr. and R. B. Schnabel. Least change secant updates for Quasi-Newton methods. *SIAM Review*, 21(4):443–459, 1979.

[7] A. Edelman, T. A. Arias, and S. T. Smith. The geometry of algorithms with orthogonality constraints. *SIAM Journal on Matrix Analysis and Applications*, 20(2):303–353, 1998.

[8] D. Gabay. Minimizing a differentiable function over a differentiable manifold. *Journal of Optimization Theory and Applications*, 37(2):177–219, 1982.

[9] S. Helgason. *Differential Geometry, Lie Groups, and Symmetric Spaces*. American Mathematical Society, 2001.

[10] M. W. Hirsch. *Differential Topology*. Springer-Verlag, 1994.

[11] H. Karcher. Riemannian center of mass and mollifier smoothing. *Communications on Pure and Applied Mathematics*, 30:509–541, 1977.

[12] J. M. Lee. *Riemannian Manifolds: An Introduction to Curvature*. Springer, 1997.

[13] D. G. Luenberger. The gradient projection method along geodesics. *Management Science*, 18:620–631, 1972.

[14] A. G. Madievski, B. D. O. Anderson, and M. R. Gevers. Optimum realizations of sampled data controllers for FWL sensitivity minimization. *Automatica*, 31(3):367–379, 1994.

[15] R. E. Mahony. The constrained Newton method on a Lie group and the symmetric eigenvalue problem. *Linear Algebra and Its Applications*, 248:67–89, 1996.

[16] R. E. Mahony and J. H. Manton. The geometry of the Newton method on non-compact Lie groups. *Journal of Global Optimization*, 23(3):309–327, August 2002.

[17] J. H. Manton. An improved least squares blind channel identification algorithm for linearly and affinely precoded communication systems. *IEEE Signal Processing Letters*, 9(9):282–285, September 2002.

[18] J. H. Manton. Optimisation algorithms exploiting unitary constraints. *IEEE Transactions on Signal Processing*, 50(3):635–650, March 2002.

[19] J. H. Manton. A globally convergent numerical algorithm for computing the centre of mass on compact Lie groups. In *Eighth International Conference on Control, Automation, Robotics and Vision*, Kunming, China, December 2004. Submitted.

[20] J. H. Manton and K. Hüper. The Karcher mean of points on the special orthogonal group. In *IEEE Conference on Decision and Control*, Atlantis, Bahamas, December 2004. Submitted.

[21] J. H. Manton, R. Mahony, and Y. Hua. The geometry of weighted low rank approximations. *IEEE Transactions on Signal Processing*, 51(2):500–514, 2003.

[22] B. Owren and B. Welfert. The Newton iteration on Lie groups. *BIT*, 40(1):121–145, 2000.

[23] J. E. Perkins, U. Helmke, and J. B. Moore. Balanced realizations via gradient flow techniques. *Systems and Control Letters*, 14:369–380, 1990.

[24] P. Petersen. *Riemannian Geometry*. Springer-Verlag, 1998.

[25] E. Polak. *Optimization: Algorithms and Consistent Approximations*. Springer-Verlag, 1997.

[26] K. Rahbar and J. P. Reilly. Blind source separation of convolved sources by joint approximate diagonalization of cross-spectral density matrices. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, Salt Lake City, Utah, May 2001.

[27] K. Rahbar, J. P. Reilly, and J. H. Manton. Blind identification of MIMO FIR systems driven by quasi-stationary sources using second order statistics: A frequency domain approach. *IEEE Transactions on Signal Processing*, 2003. Accepted.

[28] M. J. Todd. Quasi-Newton updates in abstract vector spaces. *SIAM Review*, 26(3):367–377, 1984.

[29] C. Udrişte. *Convex Functions and Optimization Methods on Riemannian Manifolds*. Kluwer Academic Publishers, 1994.

[30] F. W. Warner. *Foundations of Differentiable Manifolds and Lie Groups*. Springer-Verlag, 1983.