NORTH-HOLLAND

The Constrained Newton Method on a Lie Group and the Symmetric Eigenvalue Problem*

R. E. Mahony
Department of Systems Engineering
Research School of Physical Sciences and Engineering
Australian National University
Canberra ACT 0200, Australia

Submitted by Richard A. Brualdi

ABSTRACT

The Newton method for estimating a critical point of a real function is formulated in a coordinate free manner on an arbitrary Lie group. Convergence proofs for the numerical method are given. An application of the general approach to computing the eigenvalues of a symmetric matrix is given, and the resultant algorithm is compared with the classical shifted QR algorithm. Properties of the method described suggest that it is of interest for certain computations in online and adaptive environments.

1. INTRODUCTION

The classical approach to solving an equality constrained nonlinear optimization problem (with n states and $1 \leqslant m < n$ constraints) involves minimizing a suitable Lagrangian function over n states and m Lagrange multipliers. In contrast, if the constraints define an $(n - m)$-dimensional submanifold of $\mathbb{R}^n$, then at least locally the constrained optimization problem can be rephrased as an unconstrained optimization problem in $n - m$

* The authors wish to acknowledge the funding of the activities of the Cooperative Research Centre for Robust and Adaptive Systems by the Australian Commonwealth Government under the Cooperative Research Centres Program and separate support by Boeing Commercial Aircraft Cooperation Inc.

LINEAR ALGEBRA AND ITS APPLICATIONS 248:67–89 (1996)

655 Avenue of the Americas, New York, NY 10010
0024-3795/96/$15.00
SSDI 0024-3795(95)00171-M

variables (local coordinates). The class of problems which can be tackled using a local-coordinate formulation is restricted to those for which the constraint set is easily expressed as a submanifold of $\mathbb{R}^n$. An important class of such problems are those where the constraint set is a Lie group or homogeneous space.

Optimization problems on Lie groups and homogeneous spaces have been studied recently in the context of using continuous-time differential equations for solving problems in numerical linear algebra. For example, Brockett (1988, 1991) [see also independent work by Chu and Driessel (1990)] considered the following problem:

$$\begin{aligned} &\text{Maximize} \quad \operatorname{tr}\left(U^T H_0 U D\right), \quad D = \operatorname{diag}(1, \ldots, N), \quad H_0 = H_0^T \in \mathbb{R}^{N \times N} \\ &\text{subject to} \quad U^T U = I_N, \end{aligned}$$

where $D \in \mathbb{R}^{N \times N}$ is the diagonal matrix with diagonal entries $1, 2, \ldots, N$, H_0 is a fixed symmetric matrix, and U lies in the set of orthogonal matrices, $U \in \mathrm{O}(N) = \{U \in \mathbb{R}^{N \times N} \mid U^T U = I_N\}$, with I_N the $N \times N$ identity matrix. Brockett showed that the maximum $U_* \in \mathrm{O}(N)$ occurs when ${U_*}^T H_0 U_*$ is a diagonal matrix with diagonal entries (eigenvalues of H_0) in ascending order. Thus, solving the nonlinear optimization problem is equivalent to solving the numerical linear-algebra problem of computing the eigenvalues and eigenvectors (columns of U_*) of the matrix H_0.

The symmetric eigenvalue problem is by no means the only linear-algebra problem that has been formulated as an optimization problem on a Lie group or homogeneous space. Following Brockett's work, independent work by Helmke and Moore (1990) and Smith (1991) proposed and solved optimization problems for the singular-value decomposition of an arbitrary matrix. Applications in systems theory are proposed in Perkins, Helmke, and Moore (1990) for computing balanced realizations and in Helmke and Moore (1994, Section 5.3) for pole placement of general linear systems. One of the advantages of the optimization approach is its ability to incorporate the analytic structure inherent in the problem considered, as for example in the discussion of pole-placement algorithms for symmetric systems (Mahony, Helmke, and Moore, 1993; Mahony and Helmke, 1995). Similar techniques are also applicable to a wider class of optimization problems such as linear programming (Bayer and Lagarias, 1989; Faybusovich, 1991). In each of the problems mentioned above the fundamental computation is a nonlinear optimization problem on a constraint set which is either a homogeneous space or a Lie group directly.

A distinct advantage in using optimization based techniques for solving numerical linear-algebraic problems over classical approaches is the associ-

ated computational robustness of the algorithms. Whereas linear-algebraic factorization methods require decomposition of the entire matrix to obtain a solution, iterative optimization techniques act step by step to decrease a cost criterion which usually has direct bearing on the computational accuracy of the result. This consideration is particularly interesting when one considers online computations (in adaptive engineering applications) where minor corrections to present estimates need to be regularly performed. Thus, though optimization based solutions of numerical linear-algebra problems may not be of interest for a single computation, they appear to be very important for online and adaptive applications.

There has been little work on investigating numerical optimization algorithms for the problems described above. In Yan *et al.* (1994) a number of recursive algorithms based on gradient methods for L^2 sensitivity optimization as well as Euclidean-norm balancing are discussed. More recently some work has been done on developing a generic understanding of numerical gradient ascent algorithms for many of the above problems (Mahony et al., 1993; Brockett, 1993; Moore, Mahony, and Helmke, 1994). A general discussion of algorithms along with applications is given in the Ph.D. thesis by Mahony (1994). Quadratically convergent Newton algorithms on homogeneous manifolds have been used in general path following and constrained interior-point methods [for example in recent work by Shub and Smale (1993)] and semi definite programming methods (Nesterov and Nemirovskii, 1994, §6.4). These methods, however, rely on the constraint set being embedded in a linear space whose affine structure is exploited for the computation of each Newton iterate. A more general development on Riemannian manifolds is presented by Smith (1993). In particular, Smith studies the Euler method, the conjugate-gradient method, and a version of the Newton method motivated by consideration of parallelism generated by the Levi-Civita connection.

In this paper I develop a version of the Newton iteration on an arbitrary Lie group. The algorithm is motivated by interpolation and approximation of the cost function using one-parameter subgroups of the Lie group. In this manner the algorithm presented is independent of affine connections on the Lie group and differs from Steven Smith's approach (1993, p. 57). The algorithm is presented first with respect to an arbitrary set of canonical coordinates and later in a coordinate free form. The algorithm is not equivalent to performing repeated Newton iterations in a given local coordinate chart. It is shown that the Lie-group Newton method is quadratically convergent to critical points in the constraint set. To provide an example I consider the optimization problem (discussed above) for computing the eigenvalues of a symmetric matrix. An explicit algebraic form for the Newton method applied to the symmetric eigenvalue problem is given. It is shown

that the necessary computations for the Lie-group Newton iteration are realized by solving a set of linear matrix equations and computing a matrix exponential at each step of the algorithm. A numerical study is done, and comparisons are made between the Lie-group Newton method and the shifted QR algorithm.

The paper is divided into four sections. Section 2 develops the general form of the Newton method on a Lie group and proves quadratic convergence of the algorithm in a neighborhood of a given critical point. Section 3 provides a coordinate free formulation of the Newton method. The theory is applied to the symmetric eigenvalue problem in Section 4, and a comparison is made with the performance of the QR algorithm.

2. NEWTON METHOD ON LIE GROUPS

In this section a general formulation of the Newton method is proposed which evolves explicitly on a Lie group. The iteration can be expressed in terms of Lie derivatives and the exponential map. In practice, one still has to solve a linear system of equations to determine the regression vector. The reader is referred to Helgason (1978) and Warner (1983) for technical details on differential geometry, Lie groups, and homogeneous spaces.

The Newton method is a classical (quadratically convergent) optimization technique for determining the critical points of a smooth function (Minoux, 1986, p. 94). Given a twice differentiable function $f: \mathbb{R}^n \to \mathbb{R}$, let $\operatorname{grad} f(x) := (\partial f/\partial x^1(x), \ldots, \partial f/\partial x^n(x))^T$ be the gradient vector field on $\mathbb{R}^n$ associated with f. The Hessian matrix $\mathscr{H}_x f \in \mathbb{R}^{n\times n}$ at a point x is defined as

$$(\mathscr{H}_x f)_{ij} = \frac{\partial^2 f}{\partial x^i x^j}(x).$$

Then the Newton method is given by

ALGORITHM 2.1 (Newton method on $\mathbb{R}^n$).
Given $q_k \in \mathbb{R}^n$, compute $\operatorname{grad} f(q_k)$.
Compute the Hessian matrix $\mathscr{H}_{q_k} f$.
Set $h = -(\mathscr{H}_{q_k} f)^{-1} \operatorname{grad} f(q_k)$.
Set $q_{k+1} = q_k + h$.
Set $k \leftarrow k + 1$ and repeat.

The convergence properties of the Newton method are given by the following proposition (Minoux, 1986, p. 105).

PROPOSITION 2.2. *Let $f: \mathbb{R}^n \to \mathbb{R}$ be a smooth function on $\mathbb{R}^n$, and $p \in \mathbb{R}^n$ be a nondegenerate critical point of f, that is,* grad $f(p) = 0$ *and $\mathcal{H}_p f$ is a full-rank positive definite matrix. Then there is a neighborhood U of p and a constant C such that the Newton method (Algorithm 2.1) converges to p for any initial estimate $q_0 \in U$ and the error decreases quadratically:*

$$\|q_{k+1} - p\| \leq C\|q_k - p\|^2.$$

Given a Lie group G, let $\phi: G \to \mathbb{R}$ be a smooth function on G. Denote the identity element of G by e, and associate the tangent space T_eG with the Lie algebra $\mathfrak{g}$ of G in the canonical manner. Given $X \in T_eG$, define a right invariant vector field $\tilde{X}: G \to TG$ by $\tilde{X} := dr_\tau X$ for $i = 1, \ldots, n$, where $r_\tau(\mu) := \mu\tau$ and dr_τ denotes the differential of r_τ (Helgason, 1978, p. 102). Recall that the map $t \mapsto \exp(tX)$ [where the exponential is the unique Lie group homomorphism associated with the Lie-algebra homomorphism $\lambda(d/dt) \mapsto \lambda X$; cf. Warner (1983, p. 102)] is an integral curve of $\tilde{X}$ passing through e at time zero. Given $\sigma \in G$ arbitrary, the map $t \mapsto \exp(tX)\sigma$ is an integral curve of the right invariant vector field $\tilde{X}$ passing through the point $\sigma \in G$ at time zero. It follows directly that

$$\tilde{X}\phi(\exp(tX)\sigma) = \frac{d}{d\tau}\,\phi(\exp(\tau X)\sigma)\Big|_{\tau=t}, \tag{1}$$

where $\tilde{X}\phi$ is the Lie derivative of ϕ with respect to $\tilde{X}$. Indeed, there is a natural extension of this idea which generalizes to higher-order derivatives. These derivatives can be combined into a Taylor expansion for analytic real functions on a Lie group, though for the purposes of this paper it is only necessary to explicitly know the first two terms in the Taylor expansion. The following proposition follows directly from the full Taylor expansion (Varadarajan, 1984, p. 96).

PROPOSITION 2.3. *Given a Lie group G and a smooth function $\phi: G \to \mathbb{R}$ in a neighborhood of a point $\sigma \in G$, let $X_1, \ldots, X_n \in T_eG$ be a basis for the identity tangent space of G. Define the associated right invariant vector fields*

$\tilde{X}_i = dr_\tau X_i$ *for* $i = 1, \ldots, n$. *Then*

$$\phi(\exp(t_1 X_1 + \cdots + t_n X_n)\sigma) = \phi(\sigma) + \sum_{j=1}^{n} t_j \big(\tilde{X}_j \phi\big)(\sigma) + \mathbf{O}(\|t\|^2), \tag{2}$$

where $t = (t_1, \ldots, t_n)$ *and* $\mathbf{O}(\|t\|^2)$ *indicates that the remainder term decreases like* $\|t\|^2$ *as* $t \to 0$.

Taking the derivative of (2) with respect to the vector fields $\tilde{X}_i$ and discarding the higher-order terms, one obtains the approximation

$$\tilde{X}_i \phi(\exp(t_1 X_1 + \cdots + t_n X_n)\sigma) \approx \tilde{X}_i \phi(\sigma) + \sum_{j=1}^{n} \big(\tilde{X}_i \tilde{X}_j \phi\big)(\sigma) t_j. \tag{3}$$

Define the Hessian matrix of ϕ to be the $n \times n$ matrix with (i, j)th element

$$(\mathscr{H}_\sigma \phi)_{ij} = \big(\tilde{X}_i \tilde{X}_j \phi\big)(\sigma), \tag{4}$$

which is dependent on the choice of basis $X_1, \ldots, X_n$ for $T_e G$. Define the two column vectors $t = (t_1, \ldots, t_n)^T$ and $\Delta\phi(\sigma) = (\tilde{X}_1 \phi(\sigma), \ldots, \tilde{X}_n \phi(\sigma))^T$. Recalling the classical derivation of the Newton method on $\mathbb{R}^n$ (Minoux, 1986, p. 94), it is natural to consider the following iteration. (Assume that an initial point $\sigma_0 \in G$ and a choice of n basis elements $\{X_1, \ldots, X_n\}$ for $T_e G$ are known.)

ALGORITHM 2.4 (Newton method on a Lie group G).
Given $\sigma_k \in G$, compute $\Delta\phi(\sigma_k)$.
Compute the Hessian matrix $(\mathscr{H}_{\sigma_k}\phi)$.
Set $t = -(\mathscr{H}_{\sigma_k}\phi)^{-1} \Delta\phi(\sigma_k)$.
Set $\sigma_{k+1} = \exp(t_1 X_1 + \cdots + t_n X_n)\sigma_k$.
Set $k \leftarrow k + 1$ and repeat.

It is desirable to prove a similar result to Proposition 2.2 for the Newton method on a Lie group. To compute the rate of convergence one needs to define a measure of distance in a neighborhood of the critical point considered. Let $\mu \in G$ be a critical point of a smooth function $\phi: G \to \mathbb{R}$, and let $\{X_1, \ldots, X_n\}$ be a basis for $T_e G$ as above. There exists an open neighborhood

of 0, $U \subseteq \mathbb{R}^n$, and an open neighborhood of μ, $W \subseteq G$, such that the canonical coordinates of the first kind on G centered at μ, $\theta_\mu: U \to W$, $\theta_\mu(t^1, \ldots, t^n) := \exp(t^1 X_1 + \cdots + t^n X_n)\mu$, are a local diffeomorphism of U onto W (Helgason, 1978, p. 104). One defines distance within W by the distance induced on canonical coordinates centered at μ by the Euclidean norm in $\mathbb{R}^n$,

$$\|\exp(t^1 X_1 + \cdots + t^n X_n)\mu\| = \left(\sum_{i=1}^{n} (t^i)^2 \right)^{1/2}.$$

LEMMA 2.5. *Given a smooth function $\phi: G \to R$ on a Lie group G, let $\mu \in G$ be a nondegenerate critical point of ϕ, that is, $V\phi(\mu) = 0$ for any vector field V and the Hessian $\mathscr{H}_\mu \phi$ is a full-rank positive definite matrix. Let $X_1, \ldots, X_n \in T_e G$ be a basis for the identity tangent space of G. Then there exists a neighborhood $W \subset G$ of μ and a constant $C > 0$ such that the Newton method on G (Algorithm 2.4 with respect to the choice of basis elements $X_1, \ldots, X_n$) converges to μ for any initial estimate $\sigma_0 \in W$ and the error, measured with respect to distance induced by canonical coordinates of the first kind, decreases quadratically:*

$$\|\sigma_{k+1} - \mu\| \leqslant C\|\sigma_k - \mu\|^2.$$

Proof. The set W is constructed by taking the image (via canonical coordinates on G) of the intersection of several open neighborhoods of zero in $\mathbb{R}^n$. The open neighborhoods are constructed to ensure that the various approximation arguments necessary for the proof are valid.

Let $U_0 \subset \mathbb{R}^n$ be an open neighborhood of 0 such that the canonical coordinates of the first kind, $\theta_\mu: U_0 \to G$, $\theta_\mu(x) := \exp(x^1 X_1 + \cdots + x^n X_n)\mu$, are a diffeomorphism.

A standard result concerning the exponential of the sum of any two elements X and Y of an arbitrary Lie algebra is (Helgason, 1978, p. 106)

$$\exp(X)\exp(Y) = \exp((X + Y) + \mathbf{O}(\|X\|\,\|Y\|)),$$

for X and Y sufficiently small. Given the choice of basis elements $X_1, \ldots, X_n$, there is a isomorphism $x \mapsto x^1 X_1 + \cdots + x^n X_n$ from $\mathbb{R}^n$ to $\mathfrak{g}$, the Lie algebra of G. Setting $X = \sum_{i=1}^n x^i X_i$ and $Y = \sum_{i=1}^n y^i X_i$ it follows that there exists an open neighborhood $U_1 \subseteq \mathbb{R}^n$ of zero and $C_1 > 0$ such that for any

$x, y \in U_1$ there exists $z \in \mathbb{R}^n$ with

$$\exp\left(\sum_{i=1}^{n} z^i X_i\right) = \exp\left(\sum_{i=1}^{n} x^i X_i\right) \exp\left(\sum_{i=1}^{n} y^i X_i\right),$$

$$\|z\| \leqslant C_1 \|x\| \, \|y\|.$$

Since ϕ is smooth, the Hessian matrix $\mathscr{H}_\tau \phi$ is a smooth function of $\tau \in G$. For $\tau = \theta_\mu(x)$, then, by continuity there exists an open neighborhood $U_2 \subseteq \mathbb{R}^n$ of zero and a constant C_2 such that

$$\|\mathscr{H}_{\theta_\mu(x)} \phi - \mathscr{H}_\mu \phi\|_2 \leqslant C_2 \|x\|,$$

where $\|\cdot\|_2$ is the matrix 2-norm.

Similarly,

$$\Delta\phi(\tau) = \Delta\phi\big(\theta_\mu(x)\big) : \mathbb{R}^n \to \mathbb{R}^n$$

is a smooth vector field on $\mathbb{R}^n$, and one can obtain the Taylor expansion

$$\Delta\phi\big(\theta_\mu(x)\big) = \Delta\phi(\mu) + \sum_{i=1}^{n} x^i \left.\frac{\partial \Delta\phi\big(\theta_\mu(z)\big)}{\partial z^i}\right|_{z=0} + \mathbf{O}\big(\|x\|^2\big).$$

It is easily verified that $\Delta\phi(\mu) = 0$ (since μ is a critical point) and

$$\sum_{i=1}^{n} x^i \left.\frac{\partial \Delta\phi\big(\theta_\mu(z)\big)}{\partial z^i}\right|_{z=0} = \mathscr{H}_\mu \phi \cdot x.$$

Fixing $q_k \in U_0$, let $\sigma_k := \theta_\mu(q_k)$ and consider the regression vector for the next Newton iteration,

$$t = -\big(\mathscr{H}_{\sigma_k} \phi\big)^{-1} \Delta\phi(\sigma_k). \tag{5}$$

One has

$$-\mathscr{H}_{\sigma_k} \phi \cdot t = \Delta\phi(\sigma_k) = \mathscr{H}_\mu \phi \cdot q_k + \mathbf{O}\big(\|q_k\|^2\big)$$

and consequently

$$-\mathscr{H}_\mu\phi\cdot(t+q_k)+(\mathscr{H}_\mu\phi-\mathscr{H}_{\sigma_k}\phi)t=\mathbf{O}(\|q_k\|^2).$$

Thus, there exists an open neighborhood $U_3\subseteq\mathbb{R}^n$ of zero and a constant C_3 such that for $q_k\in U_3$

$$\begin{aligned}\|\mathscr{H}_\mu\phi\cdot(t+q_k)\| &\leqslant \|\mathscr{H}_\mu\phi-\mathscr{H}_{\sigma_k}\phi\|_2\,\|t\|+C_3\,\|q_k\|^2\\ &\leqslant C_2\,\|q_k\|\,\|t\|+C_3\,\|q_k\|^2,\end{aligned}$$

where the second inequality follows when $q_k\in U_2$.

Let $\lambda_{\min}$ be the smallest eigenvalue of $\mathscr{H}_\mu\phi$. Then

$$\lambda_{\min}\,\|t+q_k\|\leqslant\|\mathscr{H}_\mu\phi\cdot(t+q_k)\|.$$

Simple manipulations yield

$$\|t\|(\lambda_{\min}-C_2\,\|q_k\|)\leqslant\lambda_{\min}\,\|q_k\|+C_3\,\|q_k\|^2.$$

Now, to ensure that $\|t\|$ is bounded, choose $\|q_k\|\leqslant\lambda_{\min}/2C_2$. Thus,

$$\|t\|\leqslant 2\,\|q_k\|+\frac{2C_3\,\|q_k\|^2}{\lambda_{\min}}. \tag{6}$$

Letting $U_4\subset\mathbb{R}^n$ be the intersection

$$U_4=U_2\cap U_3\cap\left\{q\in\mathbb{R}^n\middle|\|q\|\leqslant\min\left\{\frac{\lambda_{\min}}{2C_2},\frac{1}{2}\right\}\right\},$$

then (since $\|q\|\leqslant\frac{1}{2}$ and thus $\|q_k\|^3\leqslant\|q_k\|^2$) one has that for $q_k\in U_4$

$$\|t+q_k\|\leqslant C_4\,\|q_k\|^2 \tag{7}$$

where $C_4=(2C_2+C_3)/\lambda_{\min}+2C_2C_3/\lambda_{\min}^2$.

Since U_1 is an open neighborhood of 0 there exists $\delta>0$ such that $\{x\in\mathbb{R}^n\mid\|x\|\leqslant\delta\}\subset U_1$. Let $U_5\subseteq\mathbb{R}^n$ be the open neighborhood of 0 such that

$$U_5=\left\{x\in\mathbb{R}^n\middle|2\,\|x\|+\frac{2C_3\,\|x\|^2}{\lambda_{\min}}<\delta\right\}.$$

In particular, it follows from (6) that when $q_k\in U_5$ then $t\in U_1$.

The proof proceeds by induction. Let

$$U = U_0 \cap U_1 \cap U_2 \cap U_3 \cap U_4 \cap U_5 \cap \left\{ q \in \mathbb{R}^n \,\middle|\, \|q\| < \frac{1}{2C} \right\},$$

where C is the constant

$$C = C_4 + 2C_1 + \frac{C_1 C_3}{\lambda_{\min}},$$

and let $W = \theta_\mu(U) \subset G$. Given $q_k \in U$ and $\sigma_k = \theta_\mu(q_k)$, let $\sigma_{k+1} = \exp(t^i X_i)\sigma_k$ [where t is given by (5)] be the next iteration of the Newton method on G. One has

$$\sigma_{k+1} = \exp\left(t^i X_i\right) \exp\left(q_k^i X_i\right) \mu,$$

and thus (since $q_k \in U$ and consequently $t \in U_1$) there exists $q_{k+1} \in \mathbb{R}^n$ such that $\sigma_{k+1} = \exp(q_{k+1}^i X_i)\mu$ and

$$\|q_{k+1} - (t + q_k)\| \leqslant C_1 \|q_k\| \, \|t\|.$$

Using (7) and (6), one obtains

$$\begin{aligned} \|q_{k+1}\| &\leqslant \left[C_4 + C_1 \left(2 + \frac{2C_3 \|q_k\|}{\lambda_{\min}} \right) \right] \|q_k\|^2 \\ &\leqslant C \, \|q_k\|^2, \end{aligned}$$

where the second line follows from $\|q_k\| \leqslant \frac{1}{2}$. Now $\|q_k\| \leqslant 1/2C$ and thus $q_{k+1} \leqslant \frac{1}{2}\|q_k\|$, which ensures that the sequence q_k converges to zero. It follows that for $\sigma_0 \in W$ the Newton iteration $\sigma_k = \theta_\mu(q_k)$ converges to μ on G and satisfies the quadratic error bound

$$\|\sigma_{k+1} - \mu\| \leqslant C \, \|\sigma_k - \mu\|^2. \qquad \blacksquare$$

REMARK 2.6. The complexity of this proof follows from the fact that the Newton method on a Lie group is not equivalent to a Euclidean Newton method applied multiply in a fixed coordinate chart on G. It is interesting to

know, however, that each separate step of the Newton method on a Lie group is related to an iteration of the Euclidean Newton method via canonical coordinates centered at the present iterate on G (Mahony, 1994, p. 165).

REMARK 2.7. The requirement that ϕ be smooth can be relaxed to ϕ twice differentiable with the Hessian $\mathscr{H}_\tau \phi$ at least Lipschitz continuous. The proof remains as given.

3. COORDINATE FREE FORMULATION OF THE NEWTON METHOD

The method presented in the previous section for computing a Newton iteration on a Lie group G depends on the construction of the Hessian matrix $\mathscr{H}_\sigma \phi$ [cf. (4)], which is explicitly defined in terms of an arbitrary choice of n basis vectors $\{X_1, \ldots, X_n\} \in T_eG$. In this section the Newton method on an arbitrary Lie group equipped with a right invariant Riemannian metric is formulated as a coordinate free iteration.

Let G be a Lie group with an inner product $g_e(\cdot, \cdot)$ defined on T_eG. Denote the right invariant group metric[1] that g_e generates on G by g. Choose a basis $\{X_1, \ldots, X_n\}$ for T_eG which is orthonormal with respect to the inner product $g_e(\cdot, \cdot)$, [i.e., $g_e(X_i, X_j) = \delta_{ij}$, where δ_{ij} is the *Kronecker delta* function, $\delta_{ij} = 0$ unless $i = j$, in which case $\delta_{ij} = 1$]. Define the right invariant vector fields

$$\tilde{X}_i = dr_\sigma \, X_i$$

associated with the basis vectors $\{X_1, \ldots, X_n\}$. Since the basis $\{X_1, \ldots, X_n\}$ was chosen to be orthonormal, it follows that the decomposition of an arbitrary smooth vector field $V: G \to TG$, $V(\sigma) \in T_\sigma G$, can be written

$$V = \sum_{j=1}^{n} v_j \tilde{X}_j = \sum_{j=1}^{n} g\left(\tilde{X}_j, V\right) \tilde{X}_j.$$

[1] Let $\eta, \xi \in T_\sigma G$; then the right invariant group metric generated by an inner product g_e on T_eG is

$$g(\eta, \xi) = g_e\left(dr_\sigma^{-1} \, \eta, dr_\sigma^{-1} \, \xi\right),$$

where $r_\sigma(\tau) := \tau\sigma$ is right multiplication by σ.

In particular, let $\phi: G \to \mathbb{R}$ be a smooth map on G, and grad ϕ be defined with respect to the metric g (Helmke and Moore, 1994, p. 356):

$$\operatorname{grad} \phi = \sum_{j=1}^{n} g\big(\tilde{X}_j, \operatorname{grad} \phi\big)\tilde{X}_j = \sum_{j=1}^{n} \big(\tilde{X}_j \phi\big)\tilde{X}_j. \tag{8}$$

Let $t = (t_1, \ldots, t_n) \in \mathbb{R}^n$, and define the vector field $\tilde{X}: G \to TG$ by $\tilde{X} = \sum_{j=1}^{n} t_j \tilde{X}_j$, which is the right invariant vector field associated with the unique element $X = \sum_{j=1}^{n} t_j X_j \in T_e G$. Observe that $\sum_{j=1}^{n} t_j (\tilde{X}_j \phi)(\sigma) = (\tilde{X}\phi)(\sigma)$, and consequently, post multiplying (3) by $\tilde{X}_i$ and summing over $i = 1, \ldots, n$, one obtains the approximation

$$\sum_{i=1}^{n} \tilde{X}_i \phi(\exp(X)\sigma)\tilde{X}_i \approx \sum_{i=1}^{n} [X_i \phi(\sigma)]\tilde{X}_i + \sum_{i=1}^{n} \left(\tilde{X}_i \sum_{j=1}^{n} t_j \tilde{X}_j \phi(\sigma) \right) \tilde{X}_i$$

$$= \operatorname{grad} \phi(\sigma) + \operatorname{grad}\big(\tilde{X}\phi\big)(\sigma).$$

Now assuming that $\exp(X)\sigma$ is a critical point of ϕ, then the left-hand side of this relation is zero. Thus, computing the regression vector for the Newton method is equivalent to solving the coordinate free equation

$$0 = \operatorname{grad} \phi(\sigma) + \operatorname{grad}\big(\tilde{X}\phi\big)(\sigma), \tag{9}$$

for the vector field $\tilde{X}$ (or equivalently the tangent vector $X \in T_e G$ that uniquely defines $\tilde{X}$). In Algorithm 2.4 the choice of $\{X_1, \ldots, X_n\}$ was arbitrary, and solving directly for $\tilde{X}$ using (9) is equivalent to setting $X = t_1 X_1 + \cdots + t_n X_n$, where $t = (t_1, \ldots, t_n)$ is the error estimate $t = -(\mathscr{H}_\sigma \phi)\, \Delta\phi(\sigma)$. Given an initial point $\sigma_0 \in G$, the Newton method on a Lie group G can be written in a coordinate free form as:

ALGORITHM 3.1 (Coordinate free Newton method).
Find $X^k \in T_e G$ such that $\tilde{X}^k(\tau) := dr_\tau\, X^k$ solves

$$0 = \operatorname{grad} \phi(\sigma_k) + \operatorname{grad}\big(\tilde{X}^k \phi\big)(\sigma_k).$$

Set $\sigma_{k+1} = \exp(X^k)\sigma_k$.
Set $k \leftarrow k + 1$ and repeat.

REMARK 3.2. It is worth noting that the solution $\tilde{X}$ of (9) is not generally a solution of

$$0 = \text{grad } \phi + \nabla_{\tilde{X}} \text{ grad } \phi,$$

where ∇ is the Levi-Civita connection.[2] This follows because in general $\nabla_{\tilde{X}} \text{ grad } \phi \neq \text{grad}(\tilde{X}\phi)$ (Mahony, 1994, p. 171). In fact, $\nabla_{\tilde{X}} \text{ grad } \phi$ does degenerate to the Hessian at a critical point μ, grad $\phi(\mu) = 0$, and as a consequence the two-form $\nabla \text{ grad } \phi$ is positive definite and symmetric in a neighborhood of μ. It is not surprising then that solving the above equation for the regression direction X in Algorithm 3.1 will also yield a quadratically convergent method. This is indeed the case, as shown by Smith (1993, pp. 57–59).

4. SYMMETRIC EIGENVALUE PROBLEM

In this section the general structure developed in the previous two sections is used to derive a coordinate free Newton method for the symmetric eigenvalue problem. A comparison is made with the shifted QR algorithm, which provides an indication of the potential of the algorithm. I stress, however, that the method is not proposed as competition for state-of-the-art numerical linear-algebra methods for solving the classical symmetric eigenvalue problem. Rather, the focus is still on adaptive and online applications where the iterative and robust nature of the algorithm are of greatest benefit.

4.1. Gradient Ascent Algorithm for Symmetric Eigenvalue Problem

To overcome the numerical difficulty associated with the lack of global convergence for the Newton method, I have combined it with a gradient ascent method derived in earlier work (Moore, Mahony and Helmke, 1992, 1994). In this subsection I briefly review the Lie geometry of the set of orthogonal matrices and the gradient ascent method.

The orthogonal group $O(N) = \{U \in \mathbb{R}^{N\times N} \mid U^T U = I_N\}$ has the following properties

1. The identity tangent space of $O(N)$ is the set of skew symmetric matrices (Warner, 1983, p. 107)

$$T_{I_N}O(N) = \text{Sk}(N) = \{\Omega \in \mathbb{R}^{N\times N} \mid \Omega = -\Omega^T\}.$$

[2] The Levi-Civita connection is the unique symmetric affine connection induced by a Riemannian structure on G.

2. The tangent space at a point $U \in \mathrm{O}(N)$ is given by the image of $T_{I_N}\mathrm{O}(N)$ via the linearization of $r_U: \mathrm{O}(N) \to \mathrm{O}(N)$, $r_U(W) := WU$ (right translation by U),

$$T_U\mathrm{O}(N) = \{\Omega U \in \mathbb{R}^{N\times N} \mid \Omega \in \mathrm{Sk}(N)\}. \tag{10}$$

3. By inclusion $\mathrm{Sk}(N) \subset \mathbb{R}^{N\times N}$ is a Lie subalgebra of the Lie algebra of $\mathrm{GL}(N, \mathbb{R})$. In particular, $\mathrm{Sk}(N)$ is closed under the matrix Lie-bracket operation $[X, Y] = XY - YX$, $[X, Y] \in \mathrm{Sk}(N)$ if X and Y are skew symmetric.

4. The scaled Euclidean inner product on $\mathrm{Sk}(N)$,

$$\langle \Omega_1, \Omega_2 \rangle = 2\,\mathrm{tr}(\Omega_1^T\Omega_2),$$

generates a right invariant group metric on $\mathrm{O}(N)$,

$$g(\Omega_1 U, \Omega_2 U) = 2\,\mathrm{tr}(\Omega_1^T\Omega_2). \tag{11}$$

Observe that

$$\langle \Omega_1 U, \Omega_2 U \rangle = 2\,\mathrm{tr}(U^T\Omega_1^T\Omega_2 U) = 2\,\mathrm{tr}(\Omega_1^T\Omega_2) = g(\Omega_1 U, \Omega_2 U),$$

since $U^TU = I_N$. Thus the right invariant group metric on $\mathrm{O}(N)$ is the scaled Euclidean inner product restricted to each individual tangent space.

Let $D \in \mathbb{R}^{N\times N}$ be the diagonal matrix with diagonal entries $1, 2, \ldots, N$, and let $H_0 = H_0^T$ be some symmetric matrix whose eigenvalues are desired. Define a cost function $\phi: \mathrm{O}(N) \to \mathbb{R}$, $\phi(U) := \mathrm{tr}(UH_0U^TD)$. Then the optimization problem outlined in the introduction becomes

$$\begin{aligned} &\text{Maximize } \phi(U) \\ &\text{subject to } U \in \mathrm{O}(N). \end{aligned}$$

To apply the theory contained in the previous sections one must compute both $\mathrm{grad}\,\phi$ and $\mathrm{grad}(\tilde{X}\phi)$ for an arbitrary right invariant vector field $\tilde{X}: \mathrm{O}(N) \to T\mathrm{O}(N)$. The results of these computations are expressed in the explicit forms provided above for the tangent space of $\mathrm{O}(N)$.

LEMMA 4.1. *Let $H_0 = H_0^T$ be a symmetric matrix, $D = \mathrm{diag}(1, \ldots, N)$, and define*

$$\begin{aligned} &\phi: \mathrm{O}(N) \to \mathbb{R}, \\ &\phi(U) := \mathrm{tr}(DUH_0U^T). \end{aligned}$$

Then:

(a) *The gradient of* ϕ *on* $O(N)$ (*with respect to the right invariant group metric* (11)) *is*

$$\operatorname{grad} \phi = [UH_0U^T, D]U.$$

(b) *Let* $X \in \mathrm{Sk}(N)$ *be arbitrary, and set* $\tilde{X} = XU = dr_U X$, *the right invariant vector field on* $O(N)$ *generated by* X. *The gradient of* $\tilde{X}\phi$ *on* $O(N)$ *is*

$$\operatorname{grad}(\tilde{X}\phi) = -[[X, D], UH_0U^T]U.$$

Proof. The Fréchet derivative of ϕ in a direction $\Omega U \in T_U O(N)$ is

$$D\phi \mid_U (\Omega U) = \operatorname{tr}(D\Omega U H_0 U^T) + \operatorname{tr}(DUH_0(\Omega U)^T)$$

$$\operatorname{tr}([D, UH_0U^T]^T\Omega) = g([D, UH_0U^T]U, \Omega U).$$

Observing that $[D, UH_0U^T]U \in T_U O(N)$, then it follows that grad $\phi(U) := [D, UH_0U^T]U$ (Helmke and Moore, 1994, p. 356).

For part (b) observe that

$$\tilde{X}\phi = D\phi \mid_U (XU)$$

$$= \operatorname{tr}(-[X, D]UH_0U^T).$$

Taking a second derivative of this in an arbitrary direction ΩU, one obtains

$$D \operatorname{tr}(-[X, D]UH_0U^T)|_U (\Omega U) = \operatorname{tr}(-[UH_0U^T, [X, D]]\Omega)$$

$$= g(-[[X, D], UH_0U^T]U, \Omega U).$$

and thus $\operatorname{grad}(\tilde{X}\phi) = -[[X, D], UH_0U^T]U$. ∎

The gradient ascent algorithm for ϕ on $O(N)$ is given by (Moore et al., 1994)

$$U_{k+1} = e^{\alpha_k[D, U_k H_0 U_k^T]}U_k, \tag{12}$$

where α_k is the step size. This algorithm is similar to a Euclidean gradient ascent algorithm except that the linear interpolation $U_{k+1} = U_k + \alpha_k[D, U_k H_0 U_k^T]U_k$ is replaced by a geodesic interpolation on $O(N)$. Details of gradient ascent methods on Lie groups and homogeneous spaces are contained in Mahony (1994). Choosing the step size α_k to ensure that the gradient ascent algorithm converges to a maximum of ϕ is discussed in Moore et al. (1994). A suitable choice is

$$\alpha_k = \frac{1}{2\|[H_k, D]\|} \log\left(\frac{\|[H_k, D]\|^2}{\|H_0\| \, \|[D, [H_k, D]]\|} + 1 \right),$$

where $H_k = U_k H_0 U_k^T$ and the norm is the matrix Frobenius norm $\|X\|^2 = \mathrm{tr}(X^T X)$. The algorithm generated by (12) is globally convergent to an orthogonal matrix U_* such that $U_* H_0 U_*^T$ is a diagonal matrix with diagonal entries in ascending order (Moore et al., 1994).

4.2. Newton Method for the Symmetric Eigenvalue Problem

Recall the equation for the coordinate free Newton method (9). Rewriting this in terms of the expression derived in Lemma 4.1 gives the algebraic equation

$$0 = -[UH_0U^T, D]U + [[X, D], UH_0U^T]U, \tag{13}$$

which one wishes to solve for $X \in \mathrm{Sk}(N)$. Thus, as expected, computing the regression vector for the Newton method is equivalent to solving a linear matrix equation.

REMARK 4.2. To see that a solution skew symmetric to this equation exists, observe that given a general solution $X \in \mathbb{R}^{N \times N}$ of (13) (which always exists, since the equation is a linear system of N^2 equations in N^2 unknowns), then

$$\begin{aligned} [[(-X^T), D], UH_0U^T] &= -[[D, X]^T, UH_0U^T] \\ &= -[[X, D], UH_0U^T]^T \\ &= -[UH_0U^T, D]^T = [UH_0U^T, D]. \end{aligned}$$

Thus, $-X^T$ is also a solution, and by linearity so is $(X - X^T)/2$. The question of uniqueness for a skew symmetric solution $X \in \mathrm{Sk}(N)$ obtained is

unclear. In the case where $UH_0U^T = \Lambda$ is diagonal with distinct eigenvalues it can be shown that $[[X, D], \lambda] = 0 \Rightarrow [X, D] = 0 \Rightarrow X = 0$ and the solution is unique. It follows that genericity of the eigenvalues of H_0 is a sufficient condition for the uniqueness of the solution X. I expect that this condition is in fact both necessary and sufficient; however, I have no proof of this at the present time.

Given an initial matrix $H_0 = H_0^T$ and choosing $U_0 = I_N$, the Newton-method solution to the symmetric eigenvalue problem is:

ALGORITHM 4.3 (Newton method for spectral decomposition).
Find $X_k \in \mathrm{Sk}(N)$ such that

$$\left[[X_k, D], U_k H_0 U_k^T\right] = \left[U_k H_0 U_k^T, D\right]. \tag{14}$$

Set $U_{k+1} = e^{X_k} U_k$, where e^{X_k} is the matrix exponential of X_k.
Set $k \leftarrow k + 1$ and repeat.

REMARK 4.4. To solve (14) I used the vec operation[3] on both sides of (14), giving

$$\Big[\left(DUH_0U^T\right)^T \otimes I_N - \left(UH_0U^T\right) \otimes D - D \otimes \left(UH_0U^T\right) + I_N \otimes \left(UH_0U^TD\right)\Big]\mathrm{vec}(X_k) = \mathrm{vec}\left(\left[UH_0U^T, D\right]\right),$$

where $\otimes$ denotes the Kronecker product of two matrices (Helmke and Moore, 1994, p. 314). Since it is known that a skew symmetric solution to (14) exists, the linearly independent $\frac{1}{2}N(N-1) \times \frac{1}{2}N(N-1)$ submatrix of the $N^2 \times N^2$ Kronecker product can be extracted and the resulting system of equations solved directly for the free variables X_{ij}, $i > j$.

4.3. Simulations and Comparisons

Two examples of the Newton method applied to the symmetric eigenvalue problem are outlined below. The second example also provides a direct comparison with the shifted QR algorithm.

In the following simulations it is useful to express the results in graphical form. The best measure of cost for a graphical expression of the results is a

[3] vec(A) is the vector generated by taking the columns of the matrix A one on top of the other.

least-squares measure based on the Frobenius norm,

$$\|UH_0U^T - D\|^2 = \|H_0\|^2 + \|D\|^2 - 2\,\mathrm{tr}\big(UH_0U^TD\big).$$

Indeed, minimizing $\|UH_0U^T - D\|^2$ is equivalent to maximizing $\phi(U) := \mathrm{tr}(UH_0U^TD)$. Moreover, the Frobenius norm measures the least-squares difference between the elements of UH_0U^T and D. It is not surprising that this distance is minimized when UH_0U^T is diagonal (has the same structure as D).

The first simulation (Figure 1) is an example of combining the gradient ascent algorithm (12) with the Newton method (Algorithm 4.3). The aim of the simulation is to compare the linear convergence rate displayed by the gradient ascent algorithm with the quadratic convergence of the Newton method. The initial condition used was generated via a random orthogonal congruency transformation of the matrix $D = \mathrm{diag}(1, 2, 3)$,

$$H_0 = \begin{pmatrix} 2.1974 & -0.8465 & -0.2401 \\ -0.8465 & 2.0890 & -0.4016 \\ -0.2401 & -0.4016 & 1.7136 \end{pmatrix}.$$

Thus, the eigenvalues of H_0 are 1, 2, and 3, and the minimum distance between D and UH_0U^T $[U \in \mathrm{O}(N)]$ is zero. In Figure 1 the distance

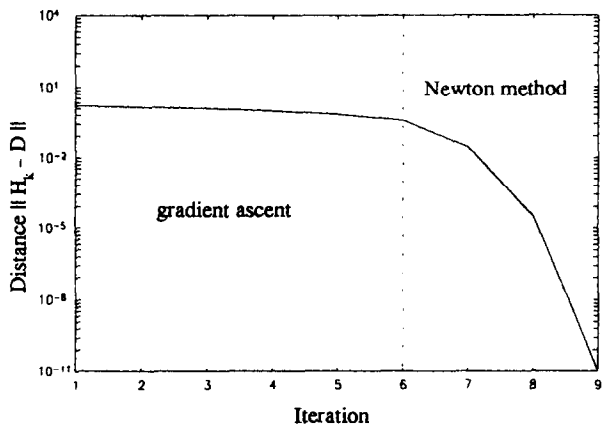

FIG. 1. Plot of $\|H_k - D\|$ where $H_k = U_kH_0U_k^T$ and U_k is a solution to first (12) and then Algorithm 4.3. The eigenvalues of H_0 are chosen to be $(1, 2, 3)$, the eigenvalues of D, though H_0 is not diagonal. Thus, the minimum Euclidean distance between $H_k = U_kH_0U_k^T$ and D is zero. By plotting the Euclidean-norm distance $\|H_k - D\|$ on a logarithmic scale the quadratic convergence characteristics of Algorithm 4.3 are displayed.

$\|H_k - D\|$ is plotted for $H_k = U_k H_0 U_k^T$, and U_k is a solution to first (12) and then Algorithm 4.3. In this example the modified gradient ascent method (12) was used for the first five iterations and the Newton method was used for the remaining three iterations. The distance $\|H_k - D\|$ is expressed on a log scale to best display the linear and quadratic convergence behavior.

To provide a comparison of the coordinate free Newton method with a classical algorithm, the second simulation was undertaken for both the Newton method and the shifted QR algorithm (Golub and Van Loan, 1989, Section 8.2). The example chosen is taken from pg. 424 of Golub and Van Loan (1989), and rather than simulate the symmetric QR algorithm again, the results used are taken directly from the example given in the book. The initial condition considered is the tridiagonal matrix

$$H_0^* = \begin{pmatrix} 1 & 2 & 0 & 0 \\ 2 & 3 & 4 & 0 \\ 0 & 4 & 5 & 6 \\ 0 & 0 & 6 & 7 \end{pmatrix}. \tag{15}$$

To display the convergence properties of the QR algorithm Golub and Van Loan (1989) give a table in which they list the values of the lower off-diagonal elements of each iterate generated for the example considered. This table is included as Table 1. Each element $(H_k)_{ij}$ is said to have converged when it has norm of order 10^{-12} or smaller. The initial condition H_0^* is tridiagonal, and the QR algorithm preserves tridiagonal structure; consequently the elements $(H_k)_{31}$, $(H_k)_{41}$, and $(H_k)_{42}$ remain zero for all

TABLE 1
THE EVOLUTION OF THE LOWER OFF-DIAGONAL ENTRIES OF THE SHIFTED QR METHOD[a]

Iteration	$(H_k)_{21}$	$(H_k)_{31}$	$(H_k)_{41}$	$(H_k)_{32}$	$(H_k)_{42}$	$(H_k)_{43}$
0	2	Zero	Zero	4	Zero	6
1	1.6817			3.2344		0.8649
2	1.6142			2.5755		0.0006
3	1.6245			1.6965		10^{-13}
4	1.6245			0.0150		Converg.
5	1.5117			10^{-9}		
6	1.1195			Converg.		
7	0.7071					
8	Converg.					

[a] Golub and Van Loan (1989, Algorithm 8.2.3, p. 423). The initial condition used is H_0^* given by (15).

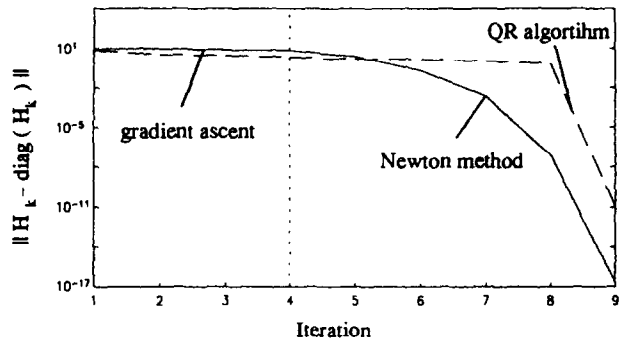

FIG. 2. A comparison of $\|H_k - \mathrm{diag}(H_k)\|$ where H_k is a solution to the symmetric QR algorithm (dashed line) and $H_k = U_k H_0 U_k^T$ for U_k a solution to both (12) and Algorithm 4.3 (solid line). The initial condition is H_0^* (15).

iterates. The convergence behavior of the symmetric QR algorithm is cubic in successive off-diagonal entries. Thus, $(H_k)_{43}$ converges cubically to zero, then $(H_k)_{32}$ converges cubically, and so on (Wilkinson, 1968). The algorithm as a whole, however, does not converge cubically, since each off-diagonal entry must converge in turn.

It is interesting to display these results in a graphical format (Figure 2). Here the norm

$$\|H_k - \mathrm{diag}(H_k)\|$$

$$= \left[(H_k)_{21}^2 + (H_k)_{31}^2 + (H_k)_{41}^2 + (H_k)_{32}^2 + (H_k)_{42}^2 + (H_k)_{43}^2\right]^{1/2}$$

is plotted verses iteration. This would seem to be an important quantity which indicates robustness and stability margins of the numerical methods considered when the values of H_k are uncertain or subject to noise. The dashed line shows the behavior of the QR algorithm. The plot displays the property of the QR algorithm that it must be run to completion to obtain an accurate solution with respect to this cost measure.

Results from the same calculation undertaken using the Newton method (combined with the gradient ascent method) are given in Table 2. In this case the tridiagonal structure of the initial condition is not preserved, and each

TABLE 2
THE EVOLUTION OF THE LOWER OFF-DIAGONAL ENTRIES OF $H_K = U_K H_0 U_K^T$ WHERE U_K IS A SOLUTION TO ALGORITHM 4.3[a]

Iteration	$(H_k)_{21}$	$(H_k)_{31}$	$(H_k)_{41}$	$(H_k)_{32}$	$(H_k)_{42}$	$(H_k)_{43}$
0	2	0	0	4	0	6
1	2.5709	−0.0117	−0.0233	4.9252	−0.4733	4.0717
2	3.7163	−0.2994	0.2498	4.3369	−0.2838	1.4798
3	4.7566	−0.7252	−0.1088	2.5257	−0.0176	0.8643
4	1.1572	−0.2222	−0.8584	1.1514	−0.1216	0.2822
5	−0.0690	−0.0362	0.0199	−0.1112	0.0649	0.0075
6	0.0011	10^{-6}	10^{-5}	10^{-5}	10^{-6}	0.0011
7	Converg.	10^{-9}	10^{-10}	10^{-10}	10^{-9}	10^{-11}
8		Converg.	Converg.	Converg.	Converg.	Converg.

[a] The initial condition is H_0^* given by (15).

off-diagonal element converges simultaneously to zero. Since the aim of this simulation is to show the potential of Newton method, the parameters were optimized to provide good convergence properties. The step size for (12) was chosen as a constant $\alpha_k = 0.1$, which is somewhat larger than the variable step size used in the first simulation. This ensures slightly faster convergence in this example, although in general there are initial conditions H_0 for which the modified gradient ascent algorithm is unstable with step-size selection fixed at 0.1. The point at which the modified gradient ascent algorithm was halted and the Newton method was begun was also chosen by experiment. The modified gradient ascent algorithm (12) was used for the first three iterations and then the Newton method for the remaining five iterations. The comparison with the QR algorithm is best made using Figure 2. Note that the Newton method acts directly to decrease the cost $\|H_k - \text{diag}(H_k)\|$, at least in a neighborhood of the critical point. It is this aspect of the algorithm that makes it suitable for online or adaptive applications where only a single iteration need be made to correct for small noise deviations.

REMARK 4.5. It is interesting to note that in this example the combination of the modified gradient ascent algorithm (12) and the Newton method (Algorithm 4.3) converges in the same number of iterations as the QR algorithm.

I would like to thank Leonid Faybusovich for his helpful suggestions during his visit to Canberra, 1994.

REFERENCES

Bayer, D. A. and Lagarias, J. C. 1989. The nonlinear geometry of linear programming I, affine and projective scaling trajectories; II, Legendre transform coordinates and central trajectories, *Trans. Amer. Math. Soc.* 314:499–581.

Brockett, R. W. 1988. Dynamical systems that sort lists, diagonalise matrices and solve linear programming problems, in *Proceedings IEEE Conference on Decision and Control*, pp. 799–803.

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

Brockett, R. W. 1993. Differential geometry and the design of gradient algorithms, *Proc. Sympos. Pure Math.* 54:69–92.

Chu, M. T. and Driessel, K. R. 1990. The projected gradient method for least squares matrix approximations with spectral constraints, *SIAM J. Numer. Anal.* 27(4):1050–1060.

Faybusovich, L. E. 1991. Hamiltonian structure of dynamical systems which solve linear programming problems, *Phys. D.*

Golub, G. H. and Van Loan, C. F. 1989. *Matrix Computations*, Johns Hopkins U.P., Baltimore.

Halgason, S. 1978. *Differential Geometry, Lie Groups and Symmetric Spaces*, Academic, New York.

Helmke, U. and Moore, J. B. 1990. Singular value decomposition via gradient flows. *Systems Control Lett.* 14:369–377.

Helmke, U. and Moore, J. B. 1994. *Optimization and Dynamical Systems*, Comm. Control Eng. Set., Springer-Verlag, London.

Mahony, R. E. 1994. Optimization Algorithms on Homogeneous Spaces: With Applications in Linear Systems Theory, Ph.D. Thesis, Dept. of Systems Engineering, Univ. of Canberra, Australia.

Mahony, R. E. and Helmke, U. 1995. System assignment and pole placement for symmetric realisations, *J. Math. Systems Estim. Control*, to appear.

Mahony, R. E., Helmke, U., and Moore, J. B. 1993. Pole placement algorithms for symmetric realisations, in *Proceedings of IEEE Conference on Decision and Control*, San Antonio, Tex.

Minoux, M. 1986. *Mathematical Programming: Theory and Algorithms*, Wiley, Chichester.

Moore, J. B., Mahony, R. E., and Helmke, U. 1992. Recursive gradient algorithms for eigenvalue and singular value decompositions, in *Proceedings of the American Control Conference*, Chicago.

Moore, J. B., Mahony, R. E., and Helmke, U. 1994. Numerical gradient algorithms for eigenvalue and singular value calculations, *SIAM J. Matrix Anal.* 15(3):881–902.

Nesterov, Y. and Nemirovskii, A. 1994. *Interior-Point Polynomical Algorithms in Convex Programming*, SIAM, Philadelphia.

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

Shub, M. and Smale, S. 1993. Complexity of Bezout's theorem. I: Geometric aspects, *J. Amer. Math. Soc.* 6(2):459–501.

Smith, S. T. 1991. Dynamical systems the perform the singular value decomposition, *Systems Control Lett.* 16(5):319–327.

Smith, S. T. 1993. Geometric Optimization Methods for Adaptive Filtering, Ph.D. Thesis, Division of Applied Science.

Varadarajan, V. S. 1984. *Lie Groups, Lie Algebras, and Their Representations*, Grad. Texts in Math. 10.2, Springer-Verlag, New York.

Warner, F. W. 1983. *Foundations of Differentiable Manifolds and Lie Groups*, Grad. Texts in Math., Springer-Verlag, New York.

Wilkinson, J. H. 1968. Global convergence of the *QR* algorithm, *Linear Algebra Appl.*

Yan, W.-Y., Moore, J. B., and Helmke, U. 1994. Recursive algorithms for solving a class of nonlinear matrix equations with applications to certain sensitivity optimization problems, *SIAM J. Control Optim.*, to appear.

Received 29 July 1994; final manuscript accepted 17 February 1995