The Geometry of Algorithms with Orthogonality Constraints

Alan Edelman*
Tomás Arias†
Steven T. Smith‡

February 2, 1997

Abstract

In this paper we develop new Newton and conjugate gradient algorithms on the Grassmann and Stiefel manifolds. These manifolds represent the constraints that arise in such areas as the symmetric eigenvalue problem, nonlinear eigenvalue problems, electronic structures computations, and signal processing. In addition to the new algorithms, we show how the geometrical framework gives penetrating new insights allowing us to create, understand, and compare algorithms. The theory proposed here provides a taxonomy for numerical linear algebra algorithms that provide a top level mathematical view of previously unrelated algorithms. It is our hope that developers of new algorithms and perturbation theories will benefit from the theory, methods, and examples in this paper.

Key words. conjugate gradient, Newton's method, orthogonality constraints, Grassmann manifold, Stiefel manifold, eigenvalues and eigenvectors, invariant subspace, Rayleigh quotient iteration, eigenvalue optimization, sequential quadratic programming, reduced gradient method, electronic structures computation, subspace tracking

AMS subject classifications. 49M07, 49M15, 53B20, 65F15, 15A18, 51F20, 81V55

*Department of Mathematics Room 2-380, Massachusetts Institute of Technology, Cambridge, MA 02139, `edelman@math.mit.edu`, `http://www-math.mit.edu/~edelman`. Supported by a fellowship from the Alfred P. Sloan Foundation.

†Department of Physics, Massachusetts Institute of Technology, Cambridge, MA 02139, `muchomas@mit.edu`. Supported by an NSF/MRSEC Seed Project grant from the MIT Center for Material Science and Engineering.

‡MIT Lincoln Laboratory, 244 Wood Street, Lexington, MA 02173, `stsmith@ll.mit.edu`. Sponsored by DARPA under Air Force contract F19628-95-C-0002. Opinions, interpretations, conclusions, and recommendations are those of the author and are not necessarily endorsed by the United States Air Force.

Submitted to *SIAM J. Matrix Anal. Appl.*

Contents

1 Introduction

Problems on the Stiefel and Grassmann manifolds arise with sufficient frequency that a unifying investigation of algorithms designed to solve these problems is warranted. Understanding these manifolds, which represent orthogonality constraints (as in the symmetric eigenvalue problem), yields penetrating insight into many numerical algorithms and unifies seemingly unrelated ideas from different areas.

The optimization community has long recognized that linear and quadratic constraints have special structure that can be exploited. The Stiefel and Grassmann manifolds also represent special constraints. The main contribution of this paper is a framework for algorithms involving these constraints, which draws upon ideas from numerical linear algebra, optimization, differential geometry, and has been inspired by certain problems posed in engineering, physics, and chemistry. Though we do review the necessary background for our intended audience, this is not a survey paper. This paper uses mathematics as a tool so that we can understand algorithms in a new deeper more insightful manner so that algorithms may be understood not as vaguely related to each other, but as part of a deeper geometrical structure.

In our first concrete problem we minimize a function $F(Y)$, where Y is constrained to the set of n-by-p matrices such that $Y^TY = I$ (we call such matrices orthonormal), and we make the further homogeneity assumption that $F(Y) = F(YQ)$, where Q is any p-by-p orthogonal matrix. In other words, the objective function depends only on the subspace spanned by the columns of Y; it is invariant to any choice of basis. The set of p-dimensional subspaces in $\mathbf{R}^n$ is called the Grassmann manifold. (Grassmann originally developed the idea in 1848, but his writing style was considered so obscure [1] that it was appreciated only many years later. One can find something of the original definition in his later work [48, Chap. 3, Sec. 1, Article 65].) To the best of our knowledge, the geometry of the Grassmann manifold has never been explored in the context of optimization algorithms, invariant subspace computations, physics computations, or subspace tracking. Useful ideas from these areas, however, may be put into the geometrical framework developed in this paper.

In our second problem we minimize $F(Y)$ without the homogeneity condition $F(Y) = F(YQ)$ mentioned above, i.e., the optimization problem is defined on the set of n-by-p orthonormal matrices. This constraint surface is known as the Stiefel manifold, which is named for Eduard Stiefel, who considered its topology in the 1930s [82]. This is the same Stiefel who in collaboration with Magnus Hestenes in 1952 originated the conjugate gradient algorithm [49]. Both Stiefel's manifold and his conjugate gradient algorithm play an important role in this paper. The geometry of the Stiefel manifold in the context of optimization problems and subspace tracking was explored by Smith [75]. In this paper we use numerical linear algebra techniques to simplify the ideas and algorithms presented there so that the differential geometric ideas seem natural and illuminating to the numerical linear algebra and optimization communities.

The first author's original motivation for studying this problem came from a response to a linear algebra survey [30], which claimed to be using conjugate gradient to solve large dense eigenvalue problems. The second and third authors were motivated by two distinct engineering and physics applications. The salient question became: "What does it mean to use conjugate gradient to solve eigenvalue problems?" Is this the Lanczos method? As we shall describe, there are dozens of proposed variations on the conjugate gradient and Newton methods for eigenvalue and related problems, none of which are Lanczos. These algorithms are not all obviously related. The connections among these algorithms have apparently not been appreciated in the literature while in some cases numerical experiments have been the only basis for comparison when no theoretical understanding was available. The existence of so many variations in so many applications compelled us to ask for the big picture: What is the mathematics that unifies all of these apparently distinct algorithms. This paper contains our proposed unification.

We summarize by itemizing what is new in this paper.

- Algorithms for Newton and conjugate gradient methods on the Grassmann and Stiefel manifolds that naturally use the geometry of these manifolds. In the special cases that we are aware of, our general algorithms are competitive up to small constant factors with the best known special algorithms. Conjugate gradient and Newton on the Grassmann manifold have never been studied before explicitly. Stiefel algorithms have been studied before [75], but the ideas here represent considerable simplifications.

- A geometrical framework with the right mix of abstraction and concreteness to serve as a foundation for any numerical computation or algorithmic formulation involving orthogonality constraints, including the symmetric eigenvalue problem. We believe that this is a useful framework because it connects apparently unrelated ideas; it is simple and mathematically natural. The framework provides new insights into existing algorithms in numerical linear algebra, optimization, signal processing, and electronic structures computations, and it suggests new algorithms. For example, we connect the ideas of geodesics and the cubic convergence of the Rayleigh quotient iteration, the CS decomposition, and sequential quadratic programming. We also interpret the ill-conditioning of eigenvectors of a symmetric matrix with multiple eigenvalues as the singularity of Stiefel and Grassmann coordinates.

- Though geometrical descriptions of the Grassmann and Stiefel manifolds are available in many references, ours is the first to use methods from numerical linear algebra emphasizing computational efficiency of algorithms rather than abstract general settings.

The remainder of this paper is organized into three sections. The geometrical ideas are developed in Section 2. This section 4 provides a self-contained introduction to geometry, which may not be familiar to some readers, while deriving the new geometrical formulas necessary for the algorithms of Section 3 and the insights of Section Section 3 provides descriptions of new algorithms for optimization on the Grassmann and Stiefel manifolds. Concrete examples of the new insights gained from this point of view are presented in Section 4. Because we wish to discuss related literature in the context developed in Sections 2 and 3, we defer discussion of the literature to Section 4 where specific applications of our theory are organized.

2 Differential Geometric Foundation for Numerical Linear Algebra

A geometrical treatment of the Stiefel and Grassmann manifolds appropriate for numerical linear algebra cannot be found in standard differential geometry references. For example, what is typically required for practical conjugate gradient computations involving n-by-p orthonormal matrices are algorithms with complexity of order np^2. In this section we derive new formulas that may be used in algorithms of this complexity in terms of standard operations from numerical linear algebra. These formulas will be used in the algorithms presented in the following section. Because we focus on computations, our approach differs from the more general (and powerful) coordinate-free methods used by modern geometers [18, 47, 54, 62, 79, 87]. Boothby [8] provides an undergraduate level introduction to the coordinate-free approach.

For readers with a background in differential geometry, we wish to point out how we use extrinsic coordinates in a somewhat unusual way. Typically, one uses a parameterization of the manifold (e.g., $x = \cos u \sin v$, $y = \sin u \sin v$, $z = \cos v$ for the sphere) to derive metric coefficients and Christoffel symbols in terms of the parameters (u and v). Instead, we only use extrinsic coordinates subject to constraints (e.g., (x, y, z) such that $x^2 + y^2 + z^2 = 1$). This represents points with more parameters than are intrinsically necessary, but we have found that the simplest (hence computationally most useful) formulas for the metric and Christoffel symbol are obtained in this manner. The choice of coordinates does not matter abstractly, but on a computer the correct choice is essential.

We now outline this section. After defining the manifolds of interest to us in Section 2.1, we take a close look at the Stiefel manifold as a submanifold of Euclidean space in Section 2.2. This introduces elementary ideas from differential geometry and provides the geometric structure of the orthogonal group (a special case of the Stiefel manifold), which will be used throughout the rest of the paper. However, the Euclidean metric is not natural for the Stiefel manifold, which inherits a canonical metric from its definition as a quotient space. Therefore, we introduce the quotient space point of view in Section 2.3. With this viewpoint, we then derive our formulas for geodesics and parallel translation for the Stiefel and Grassmann manifold in Sections 2.4 and 2.5. Finally, we describe how to incorporate these formulae into conjugate gradient and Newton methods in Section 2.6.

2.1 Manifolds Arising in Numerical Linear Algebra

For simplicity of exposition, but for no fundamental reason, we will concentrate on real matrices. All ideas carry over naturally to complex matrices. Spaces of interest are

- The orthogonal group O_n consisting of n-by-n orthogonal matrices
- The Stiefel manifold $V_{n,p}$ consisting of n-by-p "tall-skinny" orthonormal matrices
- The Grassmann manifold $G_{n,p}$ obtained by identifying those matrices in $V_{n,p}$ whose columns span the same subspace (a quotient manifold).

Table 1: *Representations of subspace manifolds*

SPACE	SYMBOL	MATRIX REP.	QUOTIENT REP.
Orthogonal Group	O_n	$\boxed{Q}$	–
Stiefel Manifold	$V_{n,p}$	$\boxed{Y}$	O_n/O_{n-p}
Grassmann Manifold	$G_{n,p}$	None	$\left\{ \begin{array}{c} V_{n,p}/O_p \\ \text{or} \\ O_n/\left(O_p \times O_{n-p}\right) \end{array} \right\}$

Table 1 summarizes the definitions of these spaces. Our description of $G_{n,p}$ is necessarily more abstract than O_n or $V_{n,p}$. $G_{n,p}$ may be defined as the set of all p-dimensional subspaces of an n-dimensional space.

We shall benefit from two different yet equivalent modes of describing our spaces: concrete representations and quotient space representations. Table 2 illustrates how we store elements of $V_{n,p}$ and $G_{n,p}$ in a computer. A point in the Stiefel manifold $V_{n,p}$ is represented by an n-by-p matrix. A point on the Grassmann manifold $G_{n,p}$ is a linear subspace, which may be specified by an arbitrary orthogonal basis stored as an n-by-p matrix. An important difference here is that, unlike points on the Stiefel manifold, the choice of matrix is not unique for points on the Grassmann manifold.

The second mode of representation, the more mathematical, is useful for obtaining closed-form expressions for the geometrical objects of interest. It is also the "proper" theoretical setting for these manifolds. Here, we represent the manifolds as quotient spaces. Points in the Grassmann manifold are equivalence classes of n-by-p orthogonal matrices, where two matrices are equivalent if their columns span the same p-dimensional subspace. Equivalently, two matrices are equivalent if they are related by right multiplication of an orthogonal p-by-p matrix. Therefore, $G_{n,p} = V_{n,p}/O_p$. On the computer, by necessity, we must pick a representative of the equivalence class to specify a point.

The Stiefel manifold may also be defined as a quotient space but arising from the orthogonal group. Here we identify two orthogonal matrices if their first p columns are identical or, equivalently, if they are related by right multiplication of a matrix of the form $\left(\begin{smallmatrix} I & 0 \\ 0 & Q \end{smallmatrix}\right)$, where Q is an orthogonal $(n-p)$-by-$(n-p)$ block. Therefore, $V_{n,p} = O_n/O_{n-p}$. With the Stiefel manifold so represented, one has yet another representation of the Grassmann manifold, $G_{n,p} = O_n/(O_p \times O_{n-p})$.

Table 2: *Computational representation of subspace manifolds*

Space	Data Structure	Represents	Tangents Δ
Stiefel Manifold	$\boxed{Y}$	one point	$Y^T\Delta =$ skew-symmetric
Grassmann Manifold	$\boxed{Y}$	entire equivalence class	$Y^T\Delta = 0$

2.2 The Stiefel Manifold in Euclidean Space

The Stiefel manifold $V_{n,p}$ may be embedded in the np dimensional Euclidean space of n-by-p matrices. When $p = 1$, we simply have the sphere, while when $p = n$, we have the group of orthogonal matrices known as O_n. These two special cases are the easiest, and arise in numerical linear algebra the most often.

Much of this section, which consists of three subsections, is designed to be a painless and intuitive introduction to differential geometry in Euclidean space. Section 2.2.1 is elementary. It derives formulas for projections onto the tangent and normal spaces. In Section 2.2.2, we derive formulas for geodesics on the Stiefel manifold in Euclidean space. We then discuss parallel translation in Section 2.2.3.

In the two special cases when $p = 1$ and $p = n$, the Euclidean metric and the canonical metric to be discussed in Section 2.4 are the same. Otherwise they differ.

2.2.1 Tangent and Normal Space

Intuitively, the tangent space at a point is the plane tangent to the submanifold at that point, as shown in Figure 1. For d-dimensional manifolds, this plane is a d-dimensional vector space with origin at the point of tangency. The normal space is the orthogonal complement. On the sphere, tangents are perpendicular to radii, and the normal space is radial. In this subsection, we will derive the equations for the tangent and normal spaces on the Stiefel manifold. We also compute the projection operators onto these spaces.

An equation defining tangents to the Stiefel manifold at a point Y is easily obtained by differentiating $Y^TY = I$, yielding $Y^T\Delta + \Delta^TY = 0$, i.e., $Y^T\Delta$ is skew-symmetric. This condition imposes $p(p+1)/2$ constraints on Δ, or equivalently, the vector space of all tangent vectors Δ has dimension

$$np - \frac{p(p+1)}{2} = \frac{p(p-1)}{2} + p(n-p). \tag{1}$$

Both sides of Eq. (1) are useful for the dimension counting arguments that will be employed.

The normal space is defined to be the orthogonal complement of the tangent space. Orthogonality depends upon the definition of an inner product, and because in this sub-

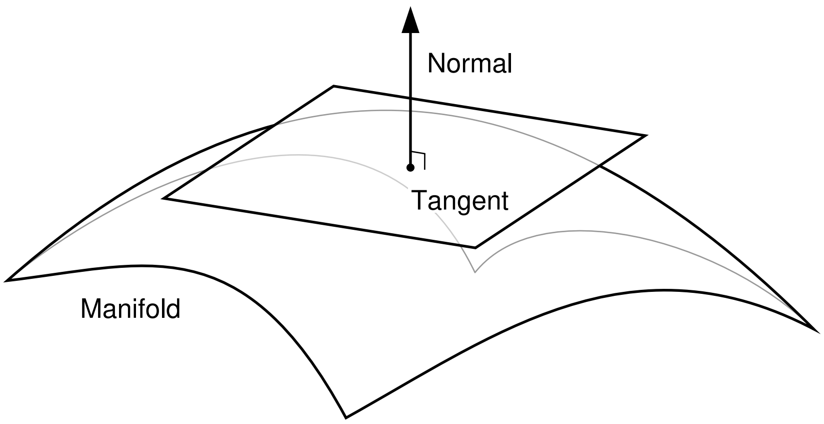

Figure 1: The tangent and normal spaces of an embedded or constraint manifold.

section we view the Stiefel manifold as an embedded manifold in Euclidean space, we choose the standard inner product

$$g_e(\Delta_1, \Delta_2) = \operatorname{tr} \Delta_1^T \Delta_2, \tag{2}$$

in np-dimensional Euclidean space (hence the subscript e), which is also the Frobenius inner product for n-by-p matrices. Therefore, the normal space at a point Y consists of all matrices N which satisfy

$$\operatorname{tr} \Delta^T N = 0$$

for all Δ in the tangent space. It follows that the normal space is $p(p+1)/2$ dimensional. It is easily verified that if $N = YS$, where S is p-by-p symmetric, then N is in the normal space. Since the dimension of the space of such matrices is $p(p+1)/2$, we see that the normal space is exactly the set of matrices $\{ YS \}$, where S is any p-by-p symmetric matrix.

Let Z be any n-by-p matrix. Letting $\operatorname{sym}(A)$ denote $(A + A^T)/2$ and $\operatorname{skew}(A) = (A - A^T)/2$, it is easily verified that at Y

$$\pi_N(Z) = Y \operatorname{sym}(Y^T Z) \qquad \text{(Euclidean)} \tag{3}$$

defines a projection of Z onto the normal space. Similarly, at Y,

$$\pi_T(Z) = Y \operatorname{skew}(Y^T Z) + (I - YY^T)Z \tag{4}$$

is a projection of Z onto the tangent space at Y (this is also true of the canonical metric to be discussed in Section 2.4). Eq. (4) suggests a form for the tangent space of $V_{n,p}$ at Y that will prove to be particularly useful. Tangent directions Δ at Y then have the general form,

$$\Delta = YA + Y_\perp B \tag{5}$$
$$= YA + (I - YY^T)C \tag{6}$$

where A is p-by-p skew-symmetric, B is $(n-p)$-by-p, C is n-by-p, B and C are both arbitrary, and $Y_\perp$ is any n-by-$(n-p)$ matrix such that $YY^T + Y_\perp {Y_\perp}^T = I$; note that $B = {Y_\perp}^T C$. The entries in the matrices A and B parameterize the tangent space at Y with $p(p-1)/2$ degrees of freedom in A and $p(n-p)$ degrees of freedom in B, resulting in $p(p-1)/2 + p(n-p)$ degrees of freedom as seen in Eq. (1).

In the special case $Y = I_{n,p} \equiv \binom{I_p}{0}$ (the first p columns of the n-by-n identity matrix), called the origin, the tangent space at Y consists of those matrices

$$X = \begin{pmatrix} A \\ B \end{pmatrix},$$

for which A is p-by-p skew-symmetric and B is $(n-p)$-by-p arbitrary.

2.2.2 Embedded Geodesics

A geodesic is the curve of shortest length between two points on a manifold. A straightforward exercise from the calculus of variations reveals that *for the case of manifolds embedded in Euclidean space* the acceleration vector at each point along a geodesic is normal to the submanifold so long as the curve is traced with uniform speed. This condition is necessary and sufficient. In the case of the sphere, acceleration for uniform motion on a great circle is directed radially and therefore normal to the surface; therefore, great circles are geodesics on the sphere. One may consider embedding manifolds in spaces with arbitrary metrics. See Spivak [79, vol. 3, p. 4] for the appropriate generalization.

Through Eq. (3) for the normal space to the Stiefel manifold, it is easily shown that the geodesic equation for a curve $Y(t)$ on the Stiefel manifold is defined by the differential equation

$$\ddot{Y} + Y(\dot{Y}^T\dot{Y}) = 0. \tag{7}$$

To see this, we begin with the condition that $Y(t)$ remains on the Stiefel manifold,

$$Y^TY = I_p. \tag{8}$$

Taking two derivatives,

$$Y^T\ddot{Y} + 2\dot{Y}^T\dot{Y} + \ddot{Y}^TY = 0. \tag{9}$$

To be a geodesic, $\ddot{Y}(t)$ must be in the normal space at $Y(t)$ so that

$$\ddot{Y}(t) + Y(t)S = 0 \tag{10}$$

for some symmetric matrix S. Substitute Eq. (10) into (9) to obtain the geodesic equation Eq. (7). Alternatively Eq. (7) could be obtained from the Euler-Lagrange equation for the calculus of variations problem

$$d(Y_1, Y_2) = \min_{Y(t)} \int_{t_1}^{t_2} (\operatorname{tr} \dot{Y}^T\dot{Y})^{1/2}\, dt \quad \text{such that } Y(t_1) = Y_1,\ Y(t_2) = Y_2. \tag{11}$$

We identify three integrals of motion of the geodesic equation Eq. (7). Define

$$C = Y^TY, \qquad A = Y^T\dot{Y}, \qquad S = \dot{Y}^T\dot{Y}. \tag{12}$$

Directly from the geodesic equation Eq. (7),

$$\begin{aligned} \dot{C} &= A + A^T, \\ \dot{A} &= -CS + S, \\ \dot{S} &= [A, S], \end{aligned}$$

where

$$[A,S] = AS - SA \tag{13}$$

is the Lie bracket of two matrices. Under the initial conditions that Y is on the Stiefel manifold ($C = I$) and $\dot{Y}$ is a tangent (A is skew-symmetric) then the integrals of the motion have the values

$$\begin{aligned} C(t) &= I, \\ A(t) &= A(0), \\ S(t) &= e^{At}S(0)e^{-At}. \end{aligned}$$

These integrals therefore identify a constant speed curve on the Stiefel manifold. In most differential geometry books, the equation of motion for geodesics is written in intrinsic coordinates in terms of so-called Christoffel symbols which specify a quadratic form of the tangent vectors. In our formulation, the form $\Gamma_e(\dot{Y},\dot{Y}) = Y\dot{Y}^T\dot{Y}$ is written compactly in extrinsic coordinates.

With these constants of the motion, we can write an integrable equation for the final geodesic,[1]

$$\frac{d}{dt}\left(Ye^{At},\ \dot{Y}e^{At}\right) = \left(Ye^{At},\ \dot{Y}e^{At}\right)\begin{pmatrix} A & -S(0) \\ I & A \end{pmatrix},$$

with integral

$$Y(t) = \left(Y(0),\ \dot{Y}(0)\right)\ \exp t\begin{pmatrix} A & -S(0) \\ I & A \end{pmatrix}\ I_{2p,p}e^{-At}.$$

This is an exact closed form expression for the geodesic on the Stiefel manifold, but we will not use this expression in our computation. Instead we will consider the non-Euclidean canonical metric on the Stiefel manifold in Section 2.4.

We mention in the case of the orthogonal group ($p = n$), the geodesic equation is obtained simply from $A = Q^T\dot{Q} =$ constant, yielding the simple solution

$$Q(t) = Q(0)e^{At}. \tag{14}$$

From Eq. (14) it is straightforward to show that on connected components of O_n,

$$d(Q_1,Q_2) = \left(\sum_{k=1}^{n}\theta_k^2\right)^{1/2} \tag{15}$$

where $\{e^{i\theta_k}\}$ are the eigenvalues of the matrix $Q_1^TQ_2$ (cf. Eq. (67) and Section 4.3).

2.2.3 Parallel Translation

In Euclidean space, we move vectors parallel to themselves simply by moving the base of the arrow. On an embedded manifold, if we move a tangent vector to another point on the manifold by this technique, it is generally not a tangent vector. One can, however, transport tangents along paths on the manifold by infinitesimally removing the component of the transported vector in the normal space.

---

[1] We thank Ross Lippert [56] for this observation.

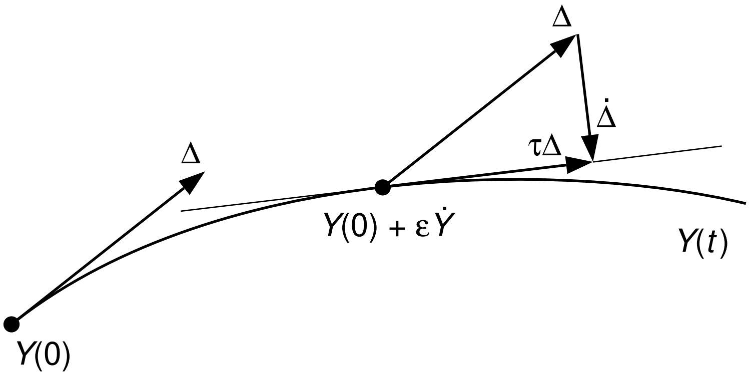

Figure 2: Parallel transport in a submanifold of Euclidean space (infinitesimal construction).

Figure 2 illustrates the idea: Imagine moving a tangent vector Δ along the curve $Y(t)$ in such a manner that every infinitesimal step consists of a parallel displacement of Δ in the Euclidean np dimensional space which is then followed by the removal of the normal component. If we move from $Y(0) = Y$ to $Y(\epsilon)$ then to first order, our new location is $Y + \epsilon\dot{Y}$. The equation for infinitesimally removing the component generated in the normal space as we move in the direction $\dot{Y}$ is obtained by differentiating Eq. (3):

$$\dot{\Delta} = -Y(\dot{Y}^T\Delta + \Delta^T\dot{Y})/2, \tag{16}$$

We are unaware of any closed form solution to this system of differential equations along geodesics.

By differentiation, we see that parallel transported vectors preserve the inner product. In particular, the square length of Δ ($\operatorname{tr}\Delta^T\Delta$) is preserved. Additionally, inserting $\dot{Y}$ into the parallel transport equation, one quickly sees that a geodesic always parallel transports its own tangent vector. This condition may be taken as the definition of a geodesic.

Observing that $\operatorname{tr}\Delta^T\Delta$ is the sum of the squares of the singular values of Δ, we conjectured that the individual singular values of Δ might also be preserved by parallel transport. Numerical experiments show that this is not the case.

In the case of the orthogonal group ($p = n$), however, parallel translation of Δ along the geodesic $Q(t) = Q(0)e^{At}$ is straightforward. Let $\Delta(t) = Q(t)B(t)$ be the solution of the parallel translation equation

$$\dot{\Delta} = -Q(\dot{Q}^T\Delta + \Delta^T\dot{Q})/2,$$

where $B(t)$ is a skew-symmetric matrix. Substituting $\dot{\Delta} = \dot{Q}B + Q\dot{B}$ and $\dot{Q} = QA$, we obtain

$$\dot{B} = -\frac{1}{2}[A, B] \tag{17}$$

whose solution is $B(t) = e^{-At/2}B(0)e^{At/2}$; therefore,

$$\Delta(t) = Q(0)e^{At/2}B(0)e^{At/2}. \tag{18}$$

These formulas may be generalized to arbitrary connected Lie groups [47, Chap. 2, Ex. A.6].

So as to arrive at the general notion of parallel transport, let us formalize what we did here. We saw that the geodesic equation may be written

$$\ddot{Y} + \Gamma_e(\dot{Y}, \dot{Y}) = 0,$$

where in the Euclidean case

$$\Gamma_e(\Delta_1, \Delta_2) = Y(\Delta_1^T\Delta_2 + \Delta_2^T\Delta_1)/2.$$

Anticipating the generalization, we interpret Γ as containing the information of the normal component that needs to be removed. Knowing the quadratic function $\Gamma(\Delta, \Delta)$ is sufficient for obtaining the bilinear function $\Gamma(\Delta_1, \Delta_2)$; the process is called polarization. We assume that Γ is a symmetric function of its arguments (this is the so-called torsion-free condition), and we obtain

$$4\Gamma(\Delta_1, \Delta_2) = \Gamma(\Delta_1 + \Delta_2, \Delta_1 + \Delta_2) - \Gamma(\Delta_1 - \Delta_2, \Delta_1 - \Delta_2).$$

For the cases we study in this paper, it is easy in practice to guess a symmetric form for $\Gamma(\Delta_1, \Delta_2)$ given $\Gamma(\Delta, \Delta)$.

We will give a specific example of why this formalism is needed in Section 2.4. Let us mention here that the parallel transport defined in this manner is known to differential geometers as the Levi-Civita connection. We also remark that the function Γ when written in terms of components defines the Christoffel symbols. Switching to vector notation, in differential geometry texts the ith component of the function $\Gamma(v, w)$ would normally be written as $\sum_{jk}\Gamma^i_{jk}v_jw_k$, where the constants Γ^i_{jk} are called Christoffel symbols. We prefer the matrix notation over the scalar notation.

2.3 Geometry of Quotient Spaces

Given a manifold whose geometry is well understood (where there are closed form expressions for the geodesics and, perhaps also, parallel transport), there is a very natural, efficient, and convenient way to generate closed form formulas on quotient spaces of that manifold. This is precisely the situation with the Stiefel and Grassmann manifolds, which are quotient spaces within the orthogonal group. As just seen in the previous section, geodesics and parallel translation on the orthogonal group are simple. We now show how the Stiefel and Grassmann manifolds inherit this simple geometry.

2.3.1 The Quotient Geometry of the Stiefel Manifold

The important ideas here are the notions of the horizontal and vertical spaces, the metric, and their relationship to geodesics and parallel translation. We use brackets to denote equivalence classes. We will define these concepts using the Stiefel manifold $V_{n,p} = O_n/O_{n-p}$ as an example. The equivalence class $[Q]$ is the set of all orthogonal matrices with the same first p columns as Q. A point in the Stiefel manifold is the equivalence class

$$[Q] = \left\{ Q \begin{pmatrix} I_p & \\ & Q_{n-p} \end{pmatrix} : Q_{n-p} \in O_{n-p} \right\}, \tag{19}$$

that is, a point in the Stiefel manifold is a particular subset of the orthogonal matrices. Notice that in this section we are working with equivalence classes rather than n-by-p matrices $Y = QI_{n,p}$.

The vertical and horizontal spaces at a point Q are complementary linear subspaces of the tangent space at Q. The vertical space is defined to be vectors tangent to the set $[Q]$. The horizontal space is defined as the tangent vectors at Q orthogonal to the vertical space. At a point Q, the vertical space is the set of vectors of the form

$$\Phi = Q \begin{pmatrix} 0 & 0 \\ 0 & C \end{pmatrix}, \tag{20}$$

where C is $(n-p)$-by-$(n-p)$ skew-symmetric, and we have hidden post-multiplication by the isotropy subgroup $\left(\begin{smallmatrix} I_p & \\ & O_{n-p} \end{smallmatrix}\right)$. Such vectors are clearly tangent to the set $[Q]$ defined in Eq. (19). It follows that the horizontal space at Q is the set of tangents of the form

$$\Delta = Q \begin{pmatrix} A & -B^T \\ B & 0 \end{pmatrix} \tag{21}$$

(also hiding the isotropy subgroup), where A is p-by-p skew-symmetric. Vectors of this form are clearly orthogonal to vertical vectors with respect to the Euclidean inner product. The matrices A and B of Eq. (21) are equivalent to those of Eq. (5).

The significance of the horizontal space is that it provides a representation of tangents to the quotient space. Intuitively, movements in the vertical direction make no change in the quotient space. Therefore, the metric, geodesics, and parallel translation must all be restricted to the horizontal space. A rigorous treatment of these intuitive concepts is given by Kobayashi and Nomizu [54] and Chavel [18].

The canonical metric on the Stiefel manifold is then simply the restriction of the orthogonal group metric to the horizontal space (multiplied by 1/2 to avoid factors of 2 later on). That is, for Δ_1 and Δ_2 of the form in Eq. (21),

$$g_c(\Delta_1, \Delta_2) = \frac{1}{2} \operatorname{tr} \left(Q \begin{pmatrix} A_1 & -B_1^T \\ B_1 & 0 \end{pmatrix} \right)^T Q \begin{pmatrix} A_2 & -B_2^T \\ B_2 & 0 \end{pmatrix} = \tfrac{1}{2} \operatorname{tr} A_1^T A_2 + \operatorname{tr} B_1^T B_2. \tag{22}$$

It is important to realize that this is *not* equal to the Euclidean metric g_e defined in Section 2.2 (except for $p = 1$ or n), even though we use the Euclidean metric for the orthogonal group in its definition. The difference arises because the Euclidean metric counts the independent coordinates of the skew-symmetric A matrix twice and those of B only once, whereas the canonical metric counts all independent coordinates in A and B equally. This point is discussed in detail in Section 2.4.

Notice that the orthogonal group geodesic

$$Q(t) = Q(0) \exp t \begin{pmatrix} A & -B^T \\ B & 0 \end{pmatrix} \tag{23}$$

has horizontal tangent

$$\dot{Q}(t) = Q(t) \begin{pmatrix} A & -B^T \\ B & 0 \end{pmatrix} \tag{24}$$

at every point along the curve $Q(t)$. Therefore they are curves of shortest length in the quotient space as well, i.e., geodesics in the Grassmann manifold are given by the simple formula

$$\text{Stiefel geodesics} = [Q(t)], \tag{25}$$

where $[Q(t)]$ is given by Eqs. (19) and (23). This formula will be essential for deriving an expression for geodesics on the Stiefel manifold using n-by-p matrices in Section 2.4.

In a quotient space, parallel translation works in a way similar to the embedded parallel translation discussed in Section 2.2.3. Parallel translation along a curve (with everywhere horizontal tangent) is accomplished by infinitesimally removing the vertical component of the tangent vector. The equation for parallel translation along the geodesics in the Stiefel manifold is obtained by applying this idea to Eq. (17), which provides translation along geodesics for the orthogonal group. Let

$$\mathfrak{A} = \begin{pmatrix} A_1 & -B_1^T \\ B_1 & 0 \end{pmatrix} \quad \text{and} \quad \mathfrak{B} = \begin{pmatrix} A_2 & -B_2^T \\ B_2 & 0 \end{pmatrix} \tag{26}$$

be two horizontal vectors at $Q = I$. The parallel translation of $\mathfrak{B}$ along the geodesic $e^{\mathfrak{A}t}$ is given by the differential equation

$$\dot{\mathfrak{B}} = -\frac{1}{2}[\mathfrak{A}, \mathfrak{B}]_H, \tag{27}$$

where the subscript H denotes the horizontal component (lower right block set to zero). Note that the Lie bracket of two horizontal vectors is not horizontal, and that the solution to Eq. (27) is not given by the formula $(e^{-\mathfrak{A}t/2}\mathfrak{B}(0)e^{\mathfrak{A}t/2})_H$. This is a special case of the general formula for reductive homogeneous spaces [18, 75]. This first order linear differential equation with constant coefficients is integrable in closed form, but it is an open question whether this can be accomplished with $O(np^2)$ operations.

2.3.2 The Quotient Geometry of the Grassmann Manifold

We quickly repeat this approach for the Grassmann manifold $G_{n,p} = O_n/(O_p \times O_{n-p})$. The equivalence class $[Q]$ is the set of all orthogonal matrices whose first p columns span the same subspace as those of Q. A point in the Grassmann manifold is the equivalence class

$$[Q] = \left\{ Q \begin{pmatrix} Q_p & \\ & Q_{n-p} \end{pmatrix} : Q_p \in O_p,\ Q_{n-p} \in O_{n-p} \right\}, \tag{28}$$

i.e., a point in the Grassmann manifold is a particular subset of the orthogonal matrices, and the Grassmann manifold itself is the collection of all these subsets.

The vertical space at a point Q is the set of vectors of the form

$$\Phi = Q \begin{pmatrix} A & 0 \\ 0 & C \end{pmatrix}, \tag{29}$$

where A is p-by-p skew-symmetric and C is $(n-p)$-by-$(n-p)$ skew-symmetric. The horizontal space at Q is the set of matrices of the form

$$\Delta = Q \begin{pmatrix} 0 & -B^T \\ B & 0 \end{pmatrix}. \tag{30}$$

Note that we have hidden post-multiplication by the isotropy subgroup $\begin{pmatrix} O_p & \\ & O_{n-p} \end{pmatrix}$ in Eqs. (29) and (30).

The canonical metric on the Grassmann manifold is the restriction of the orthogonal group metric to the horizontal space (multiplied by 1/2). Let Δ_1 and Δ_2 be of the form in Eq. (30). Then

$$g_c(\Delta_1, \Delta_2) = \operatorname{tr} B_1^T B_2. \tag{31}$$

As opposed to the canonical metric for the Stiefel manifold, this metric is in fact equivalent to the Euclidean metric (up to multiplication by 1/2) defined in Eq. (2).

The orthogonal group geodesic

$$Q(t) = Q(0) \exp t \begin{pmatrix} 0 & -B^T \\ B & 0 \end{pmatrix} \tag{32}$$

has horizontal tangent

$$\dot{Q}(t) = Q(t) \begin{pmatrix} 0 & -B^T \\ B & 0 \end{pmatrix} \tag{33}$$

at every point along the curve $Q(t)$; therefore,

$$\text{Grassmann geodesics} = [Q(t)], \tag{34}$$

where $[Q(t)]$ is given by Eqs. (28) and (32). This formula gives us an easy method for computing geodesics on the Grassmann manifold using n-by-p matrices, as will be seen in Section 2.5.

The method for parallel translation along geodesics in the Grassmann manifold is the same as for the Stiefel manifold, although it turns out the Grassmann manifold has additional structure that makes this task easier. Let

$$\mathfrak{A} = \begin{pmatrix} 0 & -A^T \\ A & 0 \end{pmatrix} \quad \text{and} \quad \mathfrak{B} = \begin{pmatrix} 0 & -B^T \\ B & 0 \end{pmatrix} \tag{35}$$

be two horizontal vectors at $Q = I$. It is easily verified that $[\mathfrak{A}, \mathfrak{B}]$ is in fact a vertical vector of the form of Eq. (29). If the vertical component of Eq. (17) is infinitesimally removed, we are left with the trivial differential equation

$$\dot{\mathfrak{B}} = 0. \tag{36}$$

Therefore, the parallel translation of the tangent vector $Q(0)\mathfrak{B}$ along the geodesic $Q(t) = Q(0)e^{\mathfrak{A}t}$ is simply given by the expression

$$\tau\mathfrak{B}(t) = Q(0)e^{\mathfrak{A}t}\mathfrak{B}, \tag{37}$$

which is of course horizontal at $Q(t)$. Here we introduce the notation τ to indicate the transport of a vector; it is not a scalar multiple of the vector. It will be seen in Section 2.5 how this formula may be computed using $O(np^2)$ operations.

As an aside, if H and V represent the horizontal and vertical spaces, respectively, it may be verified that

$$[V, V] \subset V, \quad [V, H] \subset H, \quad [H, H] \subset V. \tag{38}$$

The first relationship follows from the fact that V is a Lie algebra, the second follows from the reductive homogeneous space structure [54] of the Grassmann manifold, also possessed by the Stiefel manifold, and the third follows the symmetric space structure [47, 54] of the Grassmann manifold, which the Stiefel manifold does not possess.

2.4 The Stiefel Manifold with its Canonical Metric

2.4.1 The Canonical Metric

The Euclidean metric

$$g_e(\Delta, \Delta) = \operatorname{tr} \Delta^T \Delta$$

used in Section 2.2 may seem natural, but one reasonable objection to its use is that it weighs the independent degrees of freedom of the tangent vector unequally. Using the representation of tangent vectors $\Delta = YA + Y_\perp B$ given in Eq. (5), it is seen that

$$\begin{aligned} g_e(\Delta, \Delta) &= \operatorname{tr} A^T A + \operatorname{tr} B^T B. \\ &= 2 \sum_{i<j} a_{ij}^2 + \sum_{ij} b_{ij}^2 \end{aligned}$$

The Euclidean metric counts the $p(p+1)/2$ independent coordinates of A twice. At the origin $I_{n,p}$, a more equitable metric would be $g_c(\Delta, \Delta) = \operatorname{tr} \Delta^T (I - \frac{1}{2} I_{n,p} I_{n,p}^T) \Delta = \frac{1}{2} \operatorname{tr} A^T A + \operatorname{tr} B^T B$. To be equitable at all points in the manifold, the metric must vary with Y according to

$$g_c(\Delta, \Delta) = \operatorname{tr} \Delta^T (I - \tfrac{1}{2} Y Y^T) \Delta. \tag{39}$$

This is called the *canonical metric* on the Stiefel manifold. This is precisely the metric derived from the quotient space structure of $V_{n,p}$ in Eq. (22); therefore, the formulas for geodesics and parallel translation for the Stiefel manifold given in Section 2.3.1 are correct if we view the Stiefel manifold as the set of orthonormal n-by-p matrices with the metric of Eq. (39). Note that if $\Delta = YA + Y_\perp B$ is a tangent vector, then $g_c(\Delta, \Delta) = \frac{1}{2} \operatorname{tr} A^T A + \operatorname{tr} B^T B$, as seen previously.

2.4.2 Geodesics

The path length

$$L = \int g_c(\dot{Y}, \dot{Y})^{1/2}\, dt \tag{40}$$

may be minimized with the calculus of variations. Doing so is tedious but yields the new geodesic equation

$$\ddot{Y} + \dot{Y} \dot{Y}^T Y + Y \Big((Y^T \dot{Y})^2 + \dot{Y}^T \dot{Y} \Big) = 0, \tag{41}$$

Direct substitution into Eq. (41) using the fact that

$$(I - I_{n,p} I_{n,p}^T) X (I - I_{n,p} I_{n,p}^T) = 0,$$

if X is a skew-symmetric matrix of the form

$$X = \begin{pmatrix} A & -B^T \\ B & 0 \end{pmatrix},$$

verifies that the paths of the form

$$Y(t) = Q e^{Xt} I_{n,p} \tag{42}$$

are closed form solutions to the geodesic equation for the canonical metric.

We now turn to the problem of computing geodesics with algorithms of complexity $O(np^2)$. Our current formula $Y(t) = Q \exp t \left(\begin{smallmatrix} A & -B^T \\ B & 0 \end{smallmatrix}\right) I_{n,p}$ for a geodesic is not useful. Rather we want to express the geodesic $Y(t)$ in terms of the current position $Y(0) = Y$ and a direction $\dot{Y}(0) = H$. For example, $A = Y^T H$ and we have $C := B^T B = H^T(I - YY^T)H$. In fact the geodesic only depends on $B^T B$ rather than B itself. The trick is to find a differential equation for $M(t) = I_{n,p}^T \exp t \left(\begin{smallmatrix} A & -B^T \\ B & 0 \end{smallmatrix}\right) I_{n,p}$.

The following theorem makes clear that the computational difficulty inherent in computing the geodesic is the solution of a constant coefficient second order differential equation for $M(t)$. The answer is obtained not by a differential equation solver but rather by solving the corresponding quadratic eigenvalue problem:

Theorem 2.1 *If $Y(t) = Q e^{t\left(\begin{smallmatrix} A & -B^T \\ B & 0 \end{smallmatrix}\right)} I_{n,p}$, with $Y(0) = Y$ and $\dot{Y}(0) = H$, then*

$$Y(t) = YM(t) + (I - YY^T)H \int_0^t M(t)\, dt, \tag{43}$$

where $M(t)$ is the solution to the second order differential equation with constant coefficients,

$$\ddot{M} - A\dot{M} + CM = 0; \qquad M(0) = I_p, \quad \dot{M}(0) = A, \tag{44}$$

$A = Y^T H$ is skew-symmetric, and $C = H^T(I - YY^T)H$ is non-negative definite.

Proof A direct computation verifies that $M(t)$ satisfies Eq. (44). By separately considering $Y^T Y(t)$ and $(I - YY^T)Y(t)$, we may derive Eq. (43). $\square$

The solution of the differential equation Eq. (44) may be obtained [25, 88] by solving the quadratic eigenvalue problem

$$(\lambda^2 I - A\lambda + C)x = 0.$$

Such problems are typically solved in one of three ways: (1) by solving the generalized eigenvalue problem

$$\begin{pmatrix} C & 0 \\ 0 & I \end{pmatrix} \begin{pmatrix} x \\ \lambda x \end{pmatrix} = \lambda \begin{pmatrix} A & -I \\ I & 0 \end{pmatrix} \begin{pmatrix} x \\ \lambda x \end{pmatrix},$$

(2) by solving the eigenvalue problem

$$\begin{pmatrix} 0 & I \\ -C & A \end{pmatrix} \begin{pmatrix} x \\ \lambda x \end{pmatrix} = \lambda \begin{pmatrix} x \\ \lambda x \end{pmatrix},$$

or (3) any equivalent problem obtained by factoring $C = K^T K$ and then solving the eigenvalue problem

$$\begin{pmatrix} A & -K^T \\ K & 0 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \lambda \begin{pmatrix} x \\ y \end{pmatrix}.$$

Problems of this form arise frequently in mechanics, usually with A symmetric. Some discussion of physical interpretations for skew-symmetric matrices may be found in the context of rotating machinery [21]. If X is the p-by-$2p$ matrix of eigenvectors and Λ denotes the eigenvalues, then $M(t) = Xe^{\Lambda t}Z$, and its integral is $\int M(t)\, dt = Xe^{\Lambda t}\Lambda^{-1}Z$, where Z is chosen so that $XZ = I$ and $X\Lambda Z = A$.

Alternatively, the third method along with the matrix exponential may be employed:

Corollary 2.1 *Let Y and H be n-by-k matrices such that $Y^TY = I_p$ and $A = Y^TH$ is skew-symmetric. Then the geodesic on the Stiefel manifold emanating from Y in direction H is given by the curve*

$$Y(t) = YM(t) + QN(t), \tag{45}$$

where

$$QR := K = (I - YY^T)H \tag{46}$$

is the compact QR-decomposition of K (Q n-by-p, R p-by-p), and $M(t)$ and $N(t)$ are p-by-p matrices given by the matrix exponential

$$\begin{pmatrix} M(t) \\ N(t) \end{pmatrix} = \exp t \begin{pmatrix} A & -R^T \\ R & 0 \end{pmatrix} \begin{pmatrix} I_p \\ 0 \end{pmatrix}. \tag{47}$$

Note that Eq. (47) is easily computed by solving a $2p$-by-$2p$ skew-symmetric eigenvalue problem, which can be accomplished efficiently using the SVD or algorithms specially tailored for this problem [86].

2.4.3 Parallel Translation

We now develop a notion of parallel transport that is consistent with the canonical metric. The geodesic equation takes the form $\ddot{Y} + \Gamma(\dot{Y}, \dot{Y}) = 0$, where, from Eq. (41), it is seen that the Christoffel function for the canonical metric is

$$\Gamma_c(\Delta, \Delta) = \Delta\Delta^T Y + Y\Delta^T(I - YY^T)\Delta. \tag{48}$$

By polarizing we obtain the result

$$\Gamma_c(\Delta_1, \Delta_2) = \tfrac{1}{2}(\Delta_1\Delta_2^T + \Delta_2\Delta_1^T)Y + \tfrac{1}{2}Y\Big(\Delta_2^T(I - YY^T)\Delta_1 + \Delta_1^T(I - YY^T)\Delta_2\Big). \tag{49}$$

Parallel transport is given by the differential equation

$$\dot{\Delta} + \Gamma_c(\Delta, \dot{Y}) = 0, \tag{50}$$

which is equivalent to Eq. (27). As stated after this equation, we do not have an $O(np^2)$ method to compute $\Delta(t)$.

2.4.4 The Gradient of a Function

Both conjugate gradient and Newton's method require a computation of the gradient of a function, which depends upon the choice of metric. For a function $F(Y)$ defined on the Stiefel manifold, the gradient of F at Y is defined to be the tangent vector ∇F such that

$$\operatorname{tr} F_Y^T\Delta = g_c(\nabla F, \Delta) = \operatorname{tr}(\nabla F)^T(I - \tfrac{1}{2}YY^T)\Delta \tag{51}$$

for all tangent vectors Δ at Y, where F_Y is the n-by-p matrix of partial derivatives of F with respect to the elements of Y, i.e.,

$$(F_Y)_{ij} = \frac{\partial F}{\partial Y_{ij}}. \tag{52}$$

Solving Eq. (51) for ∇F such that $Y^T(\nabla F) =$ skew-symmetric yields

$$\nabla F = F_Y - Y F_Y^T Y. \tag{53}$$

Eq. (53) may also be derived by differentiating $F\big(Y(t)\big)$, where $Y(t)$ is the Stiefel geodesic given by Eq. (45).

2.4.5 The Hessian of a Function

Newton's method requires the Hessian of a function, which depends upon the choice of metric. The Hessian of a function $F(Y)$ defined on the Stiefel manifold is defined as the quadratic form

$$\operatorname{Hess} F(\Delta, \Delta) = \left.\frac{d^2}{dt^2}\right|_{t=0} F\big(Y(t)\big), \tag{54}$$

where $Y(t)$ is a geodesic with tangent Δ, i.e., $\dot{Y}(0) = \Delta$. Applying this definition to $F(Y)$ and Eq. (45) yields the formula

$$\begin{aligned}\operatorname{Hess} F(\Delta_1, \Delta_2) = F_{YY}(\Delta_1, \Delta_2) &+ \tfrac{1}{2}\operatorname{tr}\Big((F_Y^T \Delta_1 Y^T + Y^T \Delta_1 F_Y^T)\Delta_2\Big) \\ &- \tfrac{1}{2}\operatorname{tr}\Big((Y^T F_Y + F_Y^T Y)\Delta_1^T \Pi \Delta_2\Big),\end{aligned} \tag{55}$$

where $\Pi = I - YY^T$, F_Y is defined in Eq. (52), and the notation $F_{YY}(\Delta_1, \Delta_2)$ denotes the scalar $\sum_{ij,kl}(F_{YY})_{ij,kl}(\Delta_1)_{ij}(\Delta_2)_{kl}$ where

$$(F_{YY})_{ij,kl} = \frac{\partial^2 F}{\partial Y_{ij} \partial Y_{kl}}. \tag{56}$$

This formula may also readily be obtained by using Eq. (49) and the formula

$$\operatorname{Hess} F(\Delta_1, \Delta_2) = F_{YY}(\Delta_1, \Delta_2) - \operatorname{tr} F_Y^T \Gamma_c(\Delta_1, \Delta_2). \tag{57}$$

For Newton's method, we must determine the tangent vector Δ such that

$$\operatorname{Hess} F(\Delta, X) = \langle -G, X\rangle \quad \text{for all tangent vectors } X, \tag{58}$$

where $G = \nabla F$. We shall express the solution to this linear equation as $\Delta = -\operatorname{Hess}^{-1} G$, which may be expressed as the solution to the linear problem

$$F_{YY}(\Delta) - Y \operatorname{skew}(F_Y^T \Delta) - \operatorname{skew}(\Delta F_Y^T)Y - \frac{1}{2}\Pi \Delta Y^T F_Y = -G, \tag{59}$$

$Y^T\Delta =$ skew-symmetric, where $\operatorname{skew}(X) = (X - X^T)/2$ and the notation $F_{YY}(\Delta)$ means the unique tangent vector satisfying the equation

$$F_{YY}(\Delta, X) = \langle F_{YY}(\Delta), X\rangle \quad \text{for all tangent vectors } X. \tag{60}$$

Example problems are considered in Section 3.

2.5 The Grassmann Manifold with its Canonical Metric

A quotient space representation of the Grassmann manifold was given in Section 2.3.2; however, for computations we prefer to work with n-by-p orthonormal matrices Y. When performing computations on the Grassmann manifold, we will use the n-by-p matrix Y to represent an entire equivalence class

$$[Y] = \{\, YQ_p : Q_p \in O_p \,\}, \tag{61}$$

i.e., the subspace spanned by the columns of Y. Any representative of the equivalence class will do.

We remark that an alternative strategy is to represent points on the Grassmann manifold with projection matrices YY^T. There is one unique such matrix corresponding to each point on the Grassmann manifold. On first thought it may seem foolish to use n^2 parameters to represent a point on the Grassmann manifold (which has dimension $p(n-p)$), but in certain *ab initio* physics computations [43], the projection matrices YY^T that arise in practice tend to require only $O(n)$ parameters for their representation.

Returning to the n-by-p representation of points on the Grassmann manifold, the tangent space is easily computed by viewing the Grassmann manifold as the quotient space $G_{n,p} = V_{n,p}/O_p$. At a point Y on the Stiefel manifold then, as seen in Eq. (5), tangent vectors take the form $\Delta = YA + Y_\perp B$, where A is p-by-p skew-symmetric, B is $(n-p)$-by-p, and $Y_\perp$ is any n-by-$(n-p)$ matrix such that $(Y,\ Y_\perp)$ is orthogonal. From Eq. (61) it is clear that the vertical space at Y is the set of vectors of the form

$$\Phi = YA; \tag{62}$$

therefore, the horizontal space at Y is the set of vectors of the form

$$\Delta = Y_\perp B. \tag{63}$$

Because the horizontal space is equivalent to the tangent space of the quotient, the tangent space of the Grassmann manifold at $[Y]$ is given by all n-by-p matrices Δ of the form in Eq. (63) or, equivalently, all n-by-p matrices Δ such that

$$Y^T\Delta = 0. \tag{64}$$

Physically, this corresponds to directions free of rotations mixing the basis given by the columns of Y.

We already saw in Section 2.3.2 that the Euclidean metric is in fact equivalent to the canonical metric for the Grassmann manifold. That is, for n-by-p matrices Δ_1 and Δ_2 such that $Y^T\Delta_i = 0$ $(i = 1,\, 2)$,

$$\begin{aligned} g_c(\Delta_1, \Delta_2) &= \operatorname{tr} \Delta_1^T(I - \tfrac{1}{2}YY^T)\Delta_2, \\ &= \operatorname{tr} \Delta_1^T\Delta_2, \\ &= g_e(\Delta_1, \Delta_2). \end{aligned}$$

2.5.1 Geodesics

A formula for geodesics on the Grassmann manifold was given via Eq. (32); the following theorem provides a useful method for computing this formula using n-by-p matrices.

Theorem 2.2 *If* $Y(t) = Qe^{t\left(\begin{smallmatrix} 0 & -B^T \\ B & 0 \end{smallmatrix}\right)} I_{n,p}$, *with* $Y(0) = Y$ *and* $\dot{Y}(0) = H$, *then*

$$Y(t) = \begin{pmatrix} YV & U \end{pmatrix} \begin{pmatrix} \cos \Sigma t \\ \sin \Sigma t \end{pmatrix} V^T, \tag{65}$$

where $U\Sigma V^T$ *is the compact singular value decomposition of* H.

Proof 1 It is easy to check that either formulation for the geodesic satisfies the geodesic equation $\ddot{Y} + Y(\dot{Y}^T\dot{Y}) = 0$, with the same initial conditions. $\square$

Proof 2 Let $B = (U_1,\ U_2)\binom{\Sigma}{0}V^T$ be the singular value decomposition of B (U_1 n-by-p, U_2 p-by-$(n-p)$, Σ and V p-by-p). A straightforward computation involving the partitioned matrix

$$\begin{pmatrix} 0 & -B^T \\ B & 0 \end{pmatrix} = \begin{pmatrix} V & 0 & 0 \\ 0 & U_1 & U_2 \end{pmatrix} \begin{pmatrix} 0 & -\Sigma & 0 \\ \Sigma & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} V^T & 0 \\ 0 & U_1^T \\ 0 & U_2^T \end{pmatrix} \tag{66}$$

verifies the theorem. $\square$

A subtle point in Eq. (65) is that if the rightmost V^T is omitted, then we still have a representative of the same equivalence class as $Y(t)$; however, due to consistency conditions along the equivalent class $[Y(t)]$, the tangent (horizontal) vectors that we use for computations must be altered in the same way. This amounts to post-multiplying everything by V, or for that matter, any p-by-p orthogonal matrix.

The path length between Y_0 and $Y(t)$ (distance between subspaces) is given by [89]

$$d\Big(Y(t), Y_0\Big) = t\|H\|_F = t\left(\sum_{i=1}^{p} \sigma_i^2\right)^{1/2}, \tag{67}$$

where σ_i are the diagonal elements of Σ. (Actually, this is only true for t small enough to avoid the issue of conjugate points, e.g., long great circle routes on the sphere.) An interpretation of this formula in terms of the CS decomposition and principal angles between subspaces is given in Section 4.3.

2.5.2 Parallel Translation

A formula for parallel translation along geodesics of complexity $O(np^2)$ can also be derived:

Theorem 2.3 *Let* H *and* Δ *be tangent vectors to the Grassmann manifold at* Y. *Then the parallel translation of* Δ *along the geodesic in the direction* $\dot{Y}(0) = H$ *[Eq. (65)] is*

$$\tau\Delta(t) = \left(\begin{pmatrix} YV & U \end{pmatrix} \begin{pmatrix} -\sin \Sigma t \\ \cos \Sigma t \end{pmatrix} U^T + (I - UU^T)\right) \Delta. \tag{68}$$

Proof 1 A simple computation verifies that Eqs. (68) and (65) satisfy Eq. (16). □

Proof 2 Parallel translation of Δ is given by the expression

$$\tau\Delta(t) = Q \exp t \begin{pmatrix} 0 & -A^T \\ A & 0 \end{pmatrix} \begin{pmatrix} 0 \\ B \end{pmatrix}$$

(which follows from Eq. (37)), where $Q = (Y,\ Y_\perp)$, $H = Y_\perp A$, and $\Delta = Y_\perp B$. Decomposing $\left(\begin{smallmatrix} 0 & -A^T \\ A & 0 \end{smallmatrix}\right)$ as in Eq. (66) (note well that A has replaced B), a straightforward computation verifies the theorem. □

2.5.3 The Gradient of a Function

We must compute the gradient of a function $F(Y)$ defined on the Grassmann manifold. Similarly to Section 2.4.4, the gradient of F at $[Y]$ is defined to be the tangent vector ∇F such that

$$\operatorname{tr} F_Y^T \Delta = g_c(\nabla F, \Delta) = \operatorname{tr}(\nabla F)^T \Delta \tag{69}$$

for all tangent vectors Δ at Y, where F_Y is defined by Eq. (52). Solving Eq. (69) for ∇F such that $Y^T(\nabla F) = 0$ yields

$$\nabla F = F_Y - YY^T F_Y. \tag{70}$$

Eq. (70) may also be derived by differentiating $F\big(Y(t)\big)$, where $Y(t)$ is the Grassmann geodesic given by Eq. (65).

2.5.4 The Hessian of a Function

Applying the definition for the Hessian of $F(Y)$ given by Eq. (54) in the context of the Grassmann manifold yields the formula

$$\operatorname{Hess} F(\Delta_1, \Delta_2) = F_{YY}(\Delta_1, \Delta_2) - \operatorname{tr}(\Delta_1^T \Delta_2 Y^T F_Y), \tag{71}$$

where F_Y and F_{YY} are defined in Section 2.4.5. For Newton's method, we must determine $\Delta = -\operatorname{Hess}^{-1} G$ satisfying Eq. (58), which for the Grassmann manifold is expressed as the linear problem

$$F_{YY}(\Delta) - \Delta(Y^T F_Y) = -G, \tag{72}$$

$Y^T\Delta = 0$, where $F_{YY}(\Delta)$ denotes the unique tangent vector satisfying Eq. (60) for the Grassmann manifold's canonical metric.

Example problems are considered in Section 3.

2.6 Conjugate Gradient on Riemannian Manifolds

As demonstrated by Smith [75, 76], the benefits of using the conjugate gradient algorithm for unconstrained minimization can be carried over to minimization problems constrained to Riemannian manifolds by a covariant translation of the familiar operations of computing gradients, performing line searches, the computation of Hessians, and carrying vector information from step to step in the minimization process. In this section we will review the ideas in [75, 76] and then in the next section we formulate concrete algorithms for conjugate gradient on the Stiefel and Grassmann manifolds. Here one can see how the geometry provides insight into the true difference among the various formulas that are used in linear and nonlinear conjugate gradient algorithms.

Figure 3 sketches the conjugate gradient algorithm in flat space and Figure 4 illustrates the algorithm on a curved space. An outline for the iterative part of the algorithm (in either flat or curved space) goes as follows: at the $(k-1)$st iterate x_{k-1}, step to x_k, the minimum of f along the geodesic in the direction H_{k-1}, compute the gradient $G_k = \nabla f(x_k)$ at this point, choose the new search direction to be a combination of the old search direction and the new gradient:

$$H_k = G_k + \gamma_k \tau H_{k-1}, \tag{73}$$

and iterate until convergence. Note that τH_{k-1} in Eq. (73) is the parallel translation of the vector H_{k-1} defined in Section 2.2.3, which in this case is simply the direction of the geodesic (line) at the point x_k (see Figure 4). Also note the important condition that x_k is a minimum point along the geodesic:

$$\langle G_k, \tau H_{k-1} \rangle = 0. \tag{74}$$

Let us begin our examination of the choice of γ_k in flat space before proceeding to arbitrary manifolds. Here parallel transport is trivial so that

$$H_k = G_k + \gamma_k H_{k-1}.$$

In both linear and an idealized version of nonlinear conjugate gradient, γ_k may be determined by the exact conjugacy condition for the new search direction:

$$f_{xx}(H_k, H_{k-1}) = 0,$$

i.e., the old and new search direction must be conjugate with respect to the Hessian of f. (With $f_{xx} = A$, the common notation [45, page 523] for the conjugacy condition is $p_{k-1}^T A p_k = 0$.) The formula for γ_k is then

$$\text{Exact Conjugacy:} \quad \gamma_k = -f_{xx}(G_k, H_{k-1}) / f_{xx}(H_{k-1}, H_{k-1}). \tag{75}$$

The standard trick to improve the computational efficiency of linear conjugate gradient is to use a formula relating a finite difference of gradients to the Hessian times the direction ($r_k - r_{k-1} = -\alpha_k A p_k$ as in [45]). In our notation,

$$\langle G_k - G_{k-1}, \bullet \rangle \approx \alpha f_{xx}(\bullet, H_{k-1}), \tag{76}$$

where $\alpha = \|x_k - x_{k-1}\| / \|H_{k-1}\|$.

The formula is exact for linear conjugate gradient on flat space, otherwise it has the usual error in finite difference approximations. By applying the finite difference formula Eq. (76) in both the numerator and denominator of Eq. (75), and also applying Eq. (74) twice (once with k and once with $k-1$), one obtains the formula

$$\text{Polak-Ribière:} \quad \gamma_k = \langle G_k - G_{k-1}, G_k \rangle / \langle G_{k-1}, G_{k-1} \rangle. \tag{77}$$

Therefore the Polak-Ribiére formula is the exact formula for conjugacy through the Hessian, where one uses a difference of gradients as a finite difference approximation to the second derivative. If $f(x)$ is well approximated by a quadratic function, then $\langle G_{k-1}, G_k\rangle \approx 0$, and we obtain

$$\text{Fletcher-Reeves:} \quad \gamma_k = \langle G_k, G_k\rangle / \langle G_{k-1}, G_{k-1}\rangle. \tag{78}$$

For arbitrary manifolds, the Hessian is the second derivative along geodesics. In differential geometry it is the second covariant differential of f. Here are the formulas:

$$\begin{aligned} \text{Exact Conjugacy:} \quad & \gamma_k = -\operatorname{Hess}f(G_k, \tau H_{k-1}) / \operatorname{Hess}f(\tau H_{k-1}, \tau H_{k-1}) & (79)\\ \text{Polak-Ribière:} \quad & \gamma_k = \langle G_k - \tau G_{k-1}, G_k\rangle / \langle G_{k-1}, G_{k-1}\rangle & (80)\\ \text{Fletcher-Reeves:} \quad & \gamma_k = \langle G_k, G_k\rangle / \langle G_{k-1}, G_{k-1}\rangle & (81) \end{aligned}$$

which may be derived from the finite difference approximation to the Hessian,

$$\langle G_k - \tau G_{k-1}, \bullet\rangle \approx \alpha \operatorname{Hess}f(\bullet, \tau H_{k-1}), \quad \alpha = d(x_k, x_{k-1}) / \|H_{k-1}\|.$$

Asymptotic analyses appear in Section 3.6.

3 Geometric Optimization Algorithms

The algorithms presented here are our answer to the question: "What does it mean to perform the Newton and conjugate gradient methods on the Stiefel and Grassmann manifolds?" Though these algorithms are idealized, they are of identical complexity up to small constant factors with the best known algorithms. In particular, no differential equation routines are used. It is our hope that in the geometrical algorithms presented here, the reader will recognize elements of any algorithm that accounts for orthogonality constraints. These algorithms are special cases of the Newton and conjugate gradient methods on general Riemannian manifolds. If the objective function is nondegenerate, then the algorithms are guaranteed to converge quadratically [75, 76].

3.1 Newton's Method on the Grassmann Manifold

In flat space, Newton's method simply updates a vector by subtracting the gradient vector pre-multiplied by the inverse of the Hessian. The same is true on the Grassmann manifold (or any Riemannian manifold for that matter) of p-planes in n-dimensions, with interesting modifications. Subtraction is replaced by following a geodesic path. The gradient is the usual one (which must be tangent to the constraint surface), and the Hessian is obtained by twice differentiating the function along a geodesic. We show in Section 4.9 that this Hessian is related to the Hessian of the Lagrangian; the two Hessians arise from the difference between the intrinsic and extrinsic viewpoints. It may be suspected that following geodesics may not be computationally feasible, but because we exploit the structure of the constraint surface, this operation costs $O(np^2)$, which is required even for traditional algorithms for the eigenvalue problem—our simplest example.

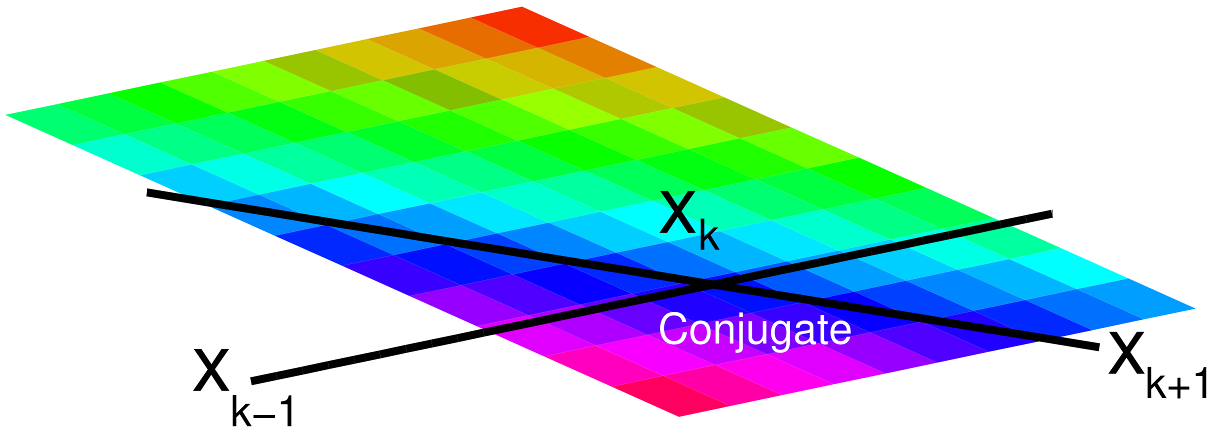

Figure 3: Conjugate gradient in flat space.

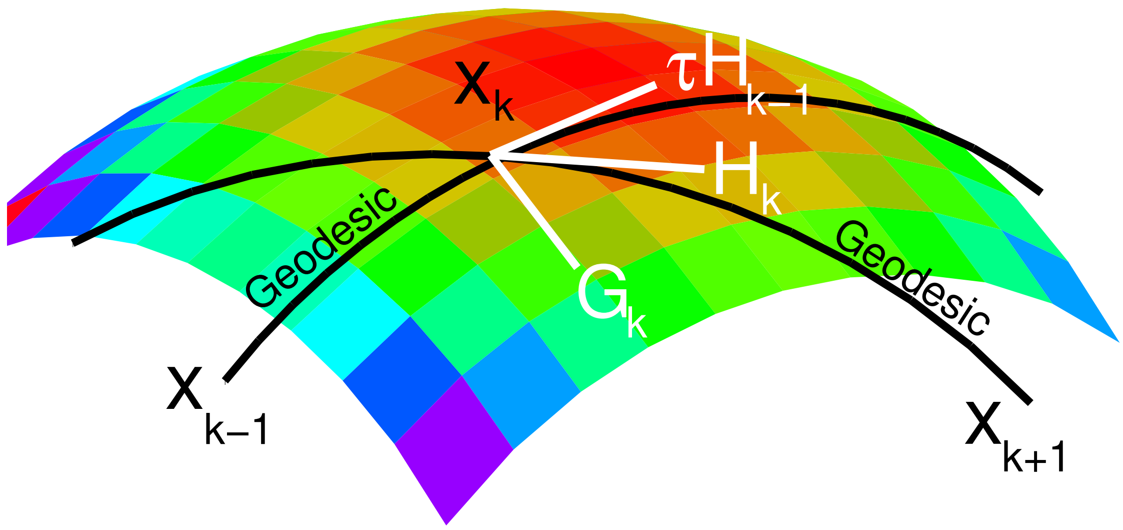

Figure 4: Conjugate gradient in curved space.

Let $F(Y)$ be a smooth function on the Grassmann manifold, i.e., $F(Y) = F(YQ)$ for any p-by-p orthogonal matrix Q, where Y is an n-by-p matrix such that $Y^TY = I_p$. We compute formulas for F_Y and $F_{YY}(\Delta)$ using the definitions given in Section 2.5.4. Newton's method for minimizing $F(Y)$ on the Grassmann manifold is:

NEWTON'S METHOD FOR MINIMIZING $F(Y)$ ON THE GRASSMANN MANIFOLD

- Given Y such that $Y^TY = I_p$,
  - Compute $G = F_Y - YY^TF_Y$.
  - Compute $\Delta = -\operatorname{Hess}^{-1} G$ such that $Y^T\Delta = 0$ and

$$F_{YY}(\Delta) - \Delta(Y^TF_Y) = -G.$$

- Move from Y in direction Δ to $Y(1)$ using the geodesic formula

$$Y(t) = YV\cos(\Sigma t)V^T + U\sin(\Sigma t)V^T$$

  where $U\Sigma V^T$ is the compact singular value decomposition of Δ (meaning U is n-by-p and both Σ and V are p-by-p).
- Repeat.

The special case of minimizing $F(Y) = \frac{1}{2}\operatorname{tr} Y^TAY$ (A n-by-n symmetric) gives the geometrically correct Newton method for the symmetric eigenvalue problem. In this case $F_Y = AY$ and $F_{YY}(\Delta) = (I - YY^T)A\Delta$. The resulting algorithm requires the solution of a Sylvester equation. It is the idealized algorithm whose approximations include various forms of Rayleigh quotient iteration, inverse iteration, a number of Newton style methods for invariant subspace computation, and the many variations of Davidson's eigenvalue method. These ideas are discussed in Sections 4.1 and 4.8.

3.2 Newton's Method on the Stiefel Manifold

Newton's method on the Stiefel manifold is conceptually equivalent to the Grassmann manifold case. Let Y be an n-by-p matrix such that $Y^TY = I_p$, and let $F(Y)$ be a smooth function of Y, without the homogeneity condition imposed for the Grassmann manifold case. Compute formulas for F_Y and $F_{YY}(\Delta)$ using the definitions given in Section 2.4.5. Newton's method for minimizing $F(Y)$ on the Stiefel manifold is:

NEWTON'S METHOD FOR MINIMIZING $F(Y)$ ON THE STIEFEL MANIFOLD

- Given Y such that $Y^T Y = I_p$,
  - Compute $G = F_Y - Y F_Y^T Y$.
  - Compute $\Delta = -\operatorname{Hess}^{-1} G$ such that $Y^T\Delta =$ skew-symmetric and

  $$F_{YY}(\Delta) - Y \operatorname{skew}(F_Y^T\Delta) - \operatorname{skew}(\Delta F_Y^T)Y - \tfrac{1}{2}\Pi\Delta Y^T F_Y = -G,$$

  where $\operatorname{skew}(X) = (X - X^T)/2$ and $\Pi = I - YY^T$.
- Move from Y in direction Δ to $Y(1)$ using the geodesic formula

  $$Y(t) = YM(t) + QN(t)$$

  where QR is the compact QR decomposition of $(I - YY^T)\Delta$ (meaning Q is n-by-p and R is p-by-p), $A = Y^T\Delta$, and $M(t)$ and $N(t)$ are p-by-p matrices given by the $2p$-by-$2p$ matrix exponential

  $$\begin{pmatrix} M(t) \\ N(t) \end{pmatrix} = \exp t \begin{pmatrix} A & -R^T \\ R & 0 \end{pmatrix} \begin{pmatrix} I_p \\ 0 \end{pmatrix}.$$

- Repeat.

For the special case of minimizing $F(Y) = \frac{1}{2}\operatorname{tr} Y^T AYN$ (A n-by-n symmetric, N p-by-p symmetric) [75], $F_Y = AYN$ and $F_{YY}(\Delta) = A\Delta N - YN\Delta^T AY$. Note that if N is not a multiple of the identity, then $F(Y)$ does not have the homogeneity condition required for a problem on the Grassmann manifold. If $N = \operatorname{diag}(p, p-1, \ldots, 1)$, then the optimum solution to maximizing F over the Stiefel manifold yields the eigenvectors corresponding to the p largest eigenvalues.

For the orthogonal Procrustes problem [32], $F(Y) = \frac{1}{2}\|AY - B\|_F^2$ (A m-by-n, B m-by-k, both arbitrary), $F_Y = A^T AY - A^T B$ and $F_{YY}(\Delta) = A^T A\Delta - Y\Delta^T A^T AY$. Note that $Y^T F_{YY}(\Delta) =$ skew-symmetric.

3.3 Conjugate Gradient Method on the Grassmann Manifold

Conjugate gradient techniques are considered because they are easy to implement, have low storage requirements, and provide superlinear convergence in the limit. The Newton equations may be solved with finitely many steps of linear conjugate gradient; each nonlinear conjugate gradient step, then, approximates a Newton step. In flat space, the nonlinear conjugate gradient method performs a line search by following a direction determined by conjugacy with respect to the Hessian. On Riemannian manifolds, conjugate gradient performs minimization along geodesics with search directions defined using the Hessian described above [75, 76]. Both algorithms approximate Hessian conjugacy with a subtle formula involving only the gradient directions, resulting in an algorithm that captures second derivative information by computing only first derivatives. To "communicate" information from one iteration to the next, tangent vectors must parallel transport along geodesics. Conceptually, this is necessary because, unlike flat space, the definition of tangent vectors changes from point to point.

Using these ideas and formulas developed in Section 3.1, the conjugate gradient method on the Grassmann manifold is:

Conjugate Gradient for Minimizing $F(Y)$ on the Grassmann Manifold

- Given Y_0 such that $Y_0^T Y_0 = I$, compute $G_0 = F_{Y_0} - Y_0 Y_0^T F_{Y_0}$ and set $H_0 = -G_0$.
- For $k = 0, 1, \ldots,$
  - Minimize $F\bigl(Y_k(t)\bigr)$ over t where

    $$Y(t) = YV\cos(\Sigma t)V^T + U\sin(\Sigma t)V^T$$

    and $U\Sigma V^T$ is the compact singular value decomposition of H_k.
  - Set $t_k = t_{\min}$ and $Y_{k+1} = Y_k(t_k)$.
  - Compute $G_{k+1} = F_{Y_{k+1}} - Y_{k+1}Y_{k+1}^T F_{Y_{k+1}}$.
  - Parallel transport tangent vectors H_k and G_k to the point Y_{k+1}:

    $$\tau H_k = (-Y_k V \sin\Sigma t_k + U\cos\Sigma t_k)\Sigma V^T, \tag{82}$$
    $$\tau G_k = G_k - \Bigl(Y_k V\sin\Sigma t_k + U(I - \cos\Sigma t_k)\Bigr)U^T G_k. \tag{83}$$

  - Compute the new search direction

    $$H_{k+1} = -G_{k+1} + \gamma_k \tau H_k \qquad \text{where} \qquad \gamma_k = \frac{\langle G_{k+1} - \tau G_k, G_{k+1}\rangle}{\langle G_k, G_k\rangle}$$

    and $\langle \Delta_1, \Delta_2\rangle = \operatorname{tr}\Delta_1^T\Delta_2$.
  - Reset $H_{k+1} = -G_{k+1}$ if $k+1 \equiv 0 \bmod p(n-p)$.

3.4 Conjugate Gradient Method on the Stiefel Manifold

As with Newton's method, conjugate gradient on the two manifolds is very similar. One need only replace the definitions of tangent vectors, inner products, geodesics, gradients, and parallel translation. Geodesics, gradients, and inner products on the Stiefel manifold are given in Section 2.4. For parallel translation along geodesics on the Stiefel manifold, we have no simple, general formula comparable to Eq. (83). Fortunately, a geodesic's tangent direction is parallel, so a simple formula for τH_k comparable to Eq. (82) is available, but a formula for τG_k is not. In practice, we recommend setting $\tau G_k := G_k$ and ignoring the fact that τG_k will not be tangent at the point Y_{k+1}. Alternatively, setting $\tau G_k := 0$ (also not parallel) results in a Fletcher-Reeves conjugate gradient formulation. As discussed in the next section, either approximation does not affect the superlinear convergence property of the conjugate gradient method.

The conjugate gradient method on the Stiefel manifold is:

Conjugate Gradient for Minimizing $F(Y)$ on the Stiefel Manifold

- Given Y_0 such that $Y_0^T Y_0 = I$, compute $G_0 = F_{Y_0} - Y_0 F_{Y_0}^T Y_0$ and set $H_0 = -G_0$.
- For $k = 0, 1, \ldots,$
  - Minimize $F\big(Y_k(t)\big)$ over t where

    $$Y_k(t) = Y_k M(t) + QN(t),$$

    QR is the compact QR decomposition of $(I - Y_k Y_k^T)H_k$, $A = Y_k^T H_k$, and $M(t)$ and $N(t)$ are p-by-p matrices given by the $2p$-by-$2p$ matrix exponential appearing in Newton's method on the Stiefel manifold in Section 3.2.
  - Set $t_k = t_{\min}$ and $Y_{k+1} = Y_k(t_k)$.
  - Compute $G_{k+1} = F_{Y_{k+1}} - Y_{k+1} F_{Y_{k+1}}^T Y_{k+1}$.
  - Parallel transport tangent vector H_k to the point Y_{k+1}:

    $$\tau H_k = H_k M(t_k) - Y_k R^T N(t_k). \tag{84}$$

    As discussed above, set $\tau G_k := G_k$ or 0, which is *not* parallel.
  - Compute the new search direction

    $$H_{k+1} = -G_{k+1} + \gamma_k \tau H_k \qquad \text{where} \qquad \gamma_k = \frac{\langle G_{k+1} - \tau G_k, G_{k+1}\rangle}{\langle G_k, G_k\rangle}$$

    and $\langle \Delta_1, \Delta_2 \rangle = \operatorname{tr} \Delta_1^T (I - \frac{1}{2} Y Y^T) \Delta_2$.
  - Reset $H_{k+1} = -G_{k+1}$ if $k + 1 \equiv 0 \bmod p(n-p) + p(p-1)/2$.

3.5 Numerical Results and Asymptotic Behavior

3.5.1 Trace Maximization on the Grassmann Manifold

The convergence properties of the conjugate gradient and Newton's methods applied to the trace maximization problem $F(Y) = \operatorname{tr} Y^T A Y$ are shown in Figure 5, as well as the convergence of an approximate conjugate gradient method and the Rayleigh quotient iteration for comparison. This example shows trace maximization on $G_{5,3}$, i.e., 3-dimensional subspaces in 5 dimensions. The distance between the subspace and the known optimum subspace is plotted versus the iteration number, where the distance in radians is simply the square root of the sum of squares of the principal angles between the subspaces. The dimension of this space equals $3(5-3) = 6$; therefore, a conjugate gradient algorithm with resets should at least double in accuracy every six iterations. Newton's method, which is cubically convergent for this example (this point is discussed in Section 4.1), should triple in accuracy every iteration. Variable precision numerical software is used to demonstrate the asymptotic convergence properties of these algorithms.

The thick black curve (CG-1) shows the convergence of the conjugate gradient (CG) algorithm using the Polak-Ribière formula. The accuracy of this algorithm is at least dou-

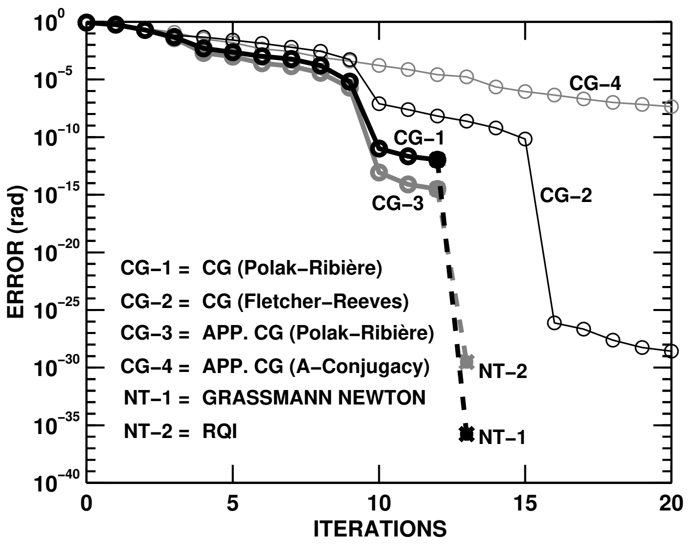

Figure 5: Convergence of the conjugate gradient and Newton's method for trace maximization on the Grassmann manifold $G_{5,3}$. The error (in radians) is the arc length distance between the solution and the subspace at the ith iterate [Eq. (67) and Section 4.3]. Quadratic convergence of CG is evident, as is cubic convergence of Newton's method, which is a special property of this example.

bled between the first and sixth and the seventh and twelfth iterations, demonstrating this method's superlinear convergence. Newton's method is applied to the twelfth CG iterate, which results in a tripling of the accuracy and demonstrates cubic convergence of Newton's method, shown by the dashed thick black curve (NT-1).

The thin black curve (CG-2) shows CG convergence using the Fletcher-Reeves formula

$$\gamma_k = \langle G_{k+1}, G_{k+1} \rangle / \langle G_k, G_k \rangle. \tag{85}$$

As discussed below, this formula differs from the Polak-Ribière formula by second order and higher terms, so it must also have superlinear convergence. The accuracy of this algorithm more than doubles between the first and sixth, seventh and twelfth, and thirteenth and eighteenth iterations, demonstrating this fact.

The algorithms discussed above are actually performed on the constraint surface, but extrinsic approximations to these algorithms are certainly possible. By perturbation analysis of the metric given below, it can be shown that the CG method differs from its flat space counterpart only by cubic and higher terms close to the solution; therefore, a flat space CG method modified by projecting search directions to the constraint's tangent space will converge superlinearly. This is basically the method proposed by Bradbury and Fletcher [9] and others for the single eigenvector case. For the Grassmann (invariant subspace) case, we have performed line searches of the function $\phi(t) = \operatorname{tr} Q(t)^T A Q(t)$, where $Q(t)R(t) := Y + t\Delta$ is the compact QR decomposition, and $Y^T\Delta = 0$. The QR decomposition projects the solution back to the constraint surface at every iteration. Tangency of the search direction at the new point is imposed via the projection $I - YY^T$.

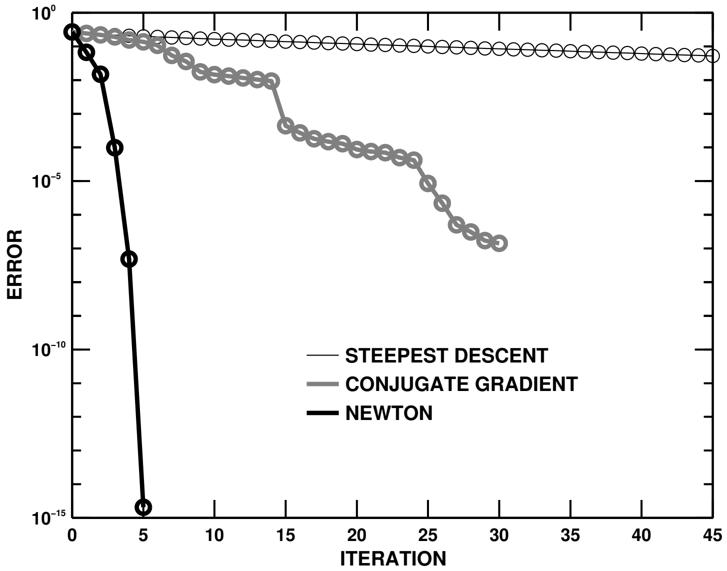

Figure 6: Convergence of the conjugate gradient and Newton's method for the orthogonal Procrustes problem on the Stiefel manifold $V_{5,3}$. The error is the Frobenius norm between the ith iterate and the known solution. Quadratic convergence of the CG and Newton methods is evident. The Newton iterates correspond to those of Table 3.

The thick gray curve (CG-3) illustrates the superlinear convergence of this method when the Polak-Ribière formula is used. The Fletcher-Reeves formula yields similar results. In contrast, the thin gray curve (CG-4) shows convergence when conjugacy through the matrix A is used, i.e., $\gamma_k = -(H_k^T A G_{k+1})/(H_k^T A H_k)$, which has been proposed by several authors [67, Eq. (5)], [19, Eq. (32)], [36, Eq. (20)]. This method cannot be expected to converge superlinearly because the matrix A is in fact quite different from the true Hessian on the constraint surface. This issue is discussed further in Section 4.4.

To compare the performance of Newton's method to the Rayleigh quotient iteration (RQI), which approximates Newton's method to high order (or vice versa), RQI is applied to the approximate CG method's twelfth iterate, shown by the dashed thick gray curve (NT-2).

3.5.2 Orthogonal Procrustes Problem on the Stiefel Manifold

The orthogonal Procrustes problem [32]

$$\min_{Y \in V_{n,p}} \|AY - B\|_F \quad A,\, B \text{ given matrices}, \tag{86}$$

is a minimization problem defined on the Stiefel manifold that has no known analytical solution for p different from 1 or n. To ensure that the objective function is smooth at optimum points, we shall consider the equivalent problem

$$\min_{Y \in V_{n,p}} \frac{1}{2}\|AY - B\|_F^2. \tag{87}$$

Iterate i	$\|Y_i - \hat{Y}\|_F$	Y_i
0	2.68×10^{-01}	$\begin{pmatrix} 0.98341252163956 & -0.09749309852408 & -0.06630579165572 \\ 0.08482117605077 & 0.99248149019173 & -0.02619408666845 \\ 0.08655810575052 & 0.02896396566088 & 0.98816425471159 \\ 0.01388126419090 & 0.00902267322408 & 0.00728525462855 \\ 0.13423928340551 & 0.06749272129685 & -0.13563090573981 \end{pmatrix}$
1	6.71×10^{-02}	$\begin{pmatrix} 0.99954707914921 & 0.01554828497046 & 0.00423211303447 \\ -0.01656743168179 & 0.99905154070826 & 0.01216605832969 \\ -0.00306529752246 & -0.01070234416262 & 0.99915251911577 \\ -0.00910501510207 & -0.01286811040265 & 0.00924631200657 \\ -0.02321334579158 & -0.03706941336228 & 0.03798454294671 \end{pmatrix}$
2	1.49×10^{-02}	$\begin{pmatrix} 0.99993878247585 & 0.00296823825310 & 0.00486487784745 \\ -0.00301651579786 & 0.99998521441661 & 0.00192519989544 \\ -0.00479673956404 & -0.00191288709538 & 0.99996440819180 \\ -0.00311307788732 & -0.00157358730922 & 0.00121316839587 \\ -0.00897953054292 & -0.00382429023234 & 0.00650669969719 \end{pmatrix}$
3	9.77×10^{-05}	$\begin{pmatrix} 0.99999999888990 & 0.00000730457866 & -0.00003211124313 \\ -0.00000730341460 & 0.99999999951242 & 0.00000603747062 \\ 0.00003210887572 & -0.00000603508216 & 0.99999999682824 \\ 0.00000457898008 & -0.00001136276061 & 0.00002209393458 \\ 0.00003339025497 & -0.00002750041840 & 0.00006919392999 \end{pmatrix}$
4	4.81×10^{-08}	$\begin{pmatrix} 1.00000000000000 & 0.00000000813187 & 0.00000001705718 \\ -0.00000000813187 & 1.00000000000000 & 0.00000000613007 \\ -0.00000001705718 & -0.00000000613007 & 1.00000000000000 \\ -0.00000001001345 & -0.00000000397730 & 0.00000000429327 \\ -0.00000002903373 & -0.00000000827864 & 0.00000002197399 \end{pmatrix}$
5	2.07×10^{-15}	$\begin{pmatrix} 1.00000000000000 & 0.00000000000000 & 0.00000000000000 \\ 0.00000000000000 & 1.00000000000000 & 0.00000000000000 \\ 0.00000000000000 & 0.00000000000000 & 1.00000000000000 \\ 0.00000000000000 & 0.00000000000000 & 0.00000000000000 \\ 0.00000000000000 & 0.00000000000000 & 0.00000000000000 \end{pmatrix}$

$$A = \begin{pmatrix} 0.59792470347241 & -1.60148995048070 & 1.29611959631725 & 0.00742708895676 & -0.09653196026400 \\ -0.34991267564713 & 1.03005546700300 & 0.38145454055699 & 0.14195063498923 & -0.16309797180034 \\ 0.16783050038338 & 0.51739189509778 & -0.42204935150912 & 1.75394028742695 & -0.63865179066515 \\ 0.24927536521443 & -1.34694675520019 & 0.92362255783368 & 0.62648865033822 & -0.31561702752866 \\ -0.24846337483192 & -0.44239067350975 & -1.52598136000449 & 0.89515519875598 & 0.87362106204727 \end{pmatrix}$$

Table 3: Newton's method applied to the orthogonal Procrustes problem on the Stiefel manifold using the MATLAB code given in this section. The matrix A is given below the numerical results, and $B = AI_{5,3}$. The quadratic convergence of Newton's method, shown by the Frobenius norm of the difference between Y_i and $\hat{Y} = I_{5,3}$, is evident. This convergence is illustrated in Figure 6. It is clear from this example that the difference $Y_i - \hat{Y}$ approaches a tangent vector at $\hat{Y} = I_{n,p}$, i.e., $\hat{Y}^T(Y_i - \hat{Y}) \to$ skew-symmetric.

Derivatives of this function appear at the end of Section 3.2. MATLAB code for Newton's method applied to this problem appears below. Convergence of this algorithm for the case $V_{5,3}$ and test matrices A and B is illustrated in Table 3 and Figure 6. The quadratic convergence of Newton's method and the conjugate gradient algorithm is evident. The dimension of $V_{5,3}$ equals $3(3-1)/2+6=9$; therefore, the accuracy of the conjugate gradient should double every nine iterations, as it is seen to do in Figure 6. Note that the matrix B is chosen such that a trivial solution $\hat{Y}=I_{n,p}$ to this test optimization problem is known.

MATLAB Code for Procrustes Problem on the Stiefel Manifold

```
n = 5; p = 3;

A = randn(n);
B = A*eye(n,p);
Y0 = eye(n,p);          % Known solution Y0
H = 0.1*randn(n,p);     H = H - Y0*(H'*Y0);  % small tangent vector H at Y0
Y = stiefgeod(Y0,H);    % Initial guess Y (close to know solution Y0)

% Newton iteration (demonstrate quadratic convergence)
d = norm(Y-Y0,'fro')
while d > sqrt(eps)
  Y = stiefgeod(Y,procrnt(Y,A,B));
  d = norm(Y-Y0,'fro')
end
```

function stiefgeod

```
function [Yt,Ht] = stiefgeod(Y,H,t)
%STIEFGEOD Geodesic on the Stiefel manifold.
%       STIEFGEOD(Y,H) is the geodesic on the Stiefel manifold
%       emanating from Y in direction H, where Y'*Y = eye(p), Y'*H =
%       skew-hermitian, and Y and H are n-by-p matrices.
%
%       STIEFGEOD(Y,H,t) produces the geodesic step in direction H scaled
%       by t.  [Yt,Ht] = STIEFGEOD(Y,H,t) produces the geodesic step and the
%       geodesic direction.

[n,p] = size(Y);

if nargin < 3, t = 1; end
A = Y'*H;  A = (A - A')/2;  % Ensure skew-symmetry
[Q,R] = qr(H - Y*A,0);
MN = expm(t*[A,-R';R,zeros(p)]);  MN = MN(:,1:p);

Yt = Y*MN(1:p,:) + Q*MN(p+1:2*p,:);  % Geodesic
if nargout > 1, Ht = H*MN(1:p,:) - Y*(R'*MN(p+1:2*p,:)); end  % Geodesic direction
```

function procrnt

```
function H = procrnt(Y,A,B)
%PROCRNT Newton Step on Stiefel Manifold for 1/2*norm(A*Y-B,'fro')^2.
%       H = PROCRNT(Y,A,B) computes the Newton step on the Stiefel manifold
%       for the function 1/2*norm(A*Y-B,'fro')^2, where Y'*Y = eye(size(Y,2)).

[n,p] = size(Y);
AA = A'*A;  FY = AA*Y - A'*B;  YFY = Y'*FY;  G = FY - Y*YFY';

% Linear conjugate gradient to solve a Newton step
dimV = p*(p-1)/2 + p*(n-p);  % == dim Stiefel manifold

% This linear CG code is modified directly from Golub and Van Loan
H = zeros(size(Y));  R1 = -G;  P = R1;  P0 = zeros(size(Y));
```

```
for k=1:dimV
  normR1 = sqrt(stiefip(Y,R1,R1));
  if normR1 < prod(size(Y))*eps, break; end
  if k == 1, beta = 0; else, beta = (normR1/normR0)^2; end
  P0 = P;  P = R1 + beta*P;  FYP = FY'*P;  YP = Y'*P;
  LP = AA*P - Y*(P'*AA*Y) ...    % Linear operation on P
     - Y*((FYP-FYP')/2) - (P*YFY'-FY*YP')/2 - (P-Y*YP)*(YFY/2);
  alpha = normR1^2/stiefip(Y,P,LP);  H = H + alpha*P;
  R0 = R1;  normR0 = normR1;  R1 = R1 - alpha*LP;
end
```

function stiefip

```
function ip = stiefip(Y,A,B)
%STIEFIP Inner product (metric) for the Stiefel manifold.
%       ip = STIEFIP(Y,A,B) returns trace(A'*(eye(n)-1/2*Y*Y')*B),
%       where Y'*Y = eye(p), Y'*A & Y'*B = skew-hermitian, and Y, A,
%       and B are n-by-p matrices.

ip = sum(sum(conj(A).*(B - Y*((Y'*B)/2))));
```

3.6 Convergence Rates of Approximate Methods

The algorithms presented in the previous sections are idealized in that geometrically natural ideas such as geodesics and parallel translation are used in their definitions. However, approximations can yield quadratic rates of convergence. In the limit, the Riemannian algorithms approach their Euclidean counterparts in the tangent plane of the solution point. A perturbation analysis shows which terms are necessary and which terms are not necessary to achieve quadratic convergence. The following argument holds for any Riemannian manifold, and therefore applies to either the Grassmann or Stiefel manifold case.

Consider the CG method applied to a function $F(Y)$ starting at a point Y within distance ϵ (small) of the solution $\hat{Y}$. For a manifold of dimension d, we must perform a sequence of d steps that take us within distance $O(\epsilon^2)$ of the solution $\hat{Y}$. The Riemannian CG method

$$\begin{aligned} H_{\text{new}} &= -G_{\text{new}} + \gamma\tau H_{\text{old}}, \quad \gamma = \frac{\langle G_{\text{new}} - \tau G_{\text{old}}, G_{\text{new}}\rangle}{\|G_{\text{old}}\|^2}; \\ Y_{\text{new}} &= Y(t_{\min}), \quad Y(0) = Y_{\text{old}}, \quad \dot{Y}(0) = H_{\text{new}}; \end{aligned}$$

does this, but we wish to approximate this procedure. Within a ball of size $O(\epsilon)$ around $\hat{Y}$, these quantities have sizes of the following orders:

Order	Quantity
$O(1)$	$t_{\min}$, γ
$O(\epsilon)$	G, H (new or old)
$O(\epsilon^2)$	$\|G\|^2$, $\|H\|^2$ (new or old)
$O(\epsilon^3)$	$\langle \tau G_{\text{old}}, G_{\text{new}}\rangle$

Also, by perturbation analysis of the Riemannian metric [18, 79, Vol. 2, Chap. 4, Props. 1

and 6], we have

$$\begin{aligned} Y(\epsilon) &= Y(0) + \epsilon\Delta + O(\epsilon^3) \\ \tau G(\epsilon) &= G + O(\epsilon^2) \\ \langle\,,\rangle &= I + O(\epsilon^2) \end{aligned}$$

where $Y(\epsilon)$ is a geodesic in direction Δ, $\tau G(\epsilon)$ is parallel translation of G along $Y(\epsilon)$, and the last approximation is valid for an orthonormal basis of the tangent plane at $Y(\epsilon\Delta)$ and I is the identity.

Inserting these asymptotics into the formulas for the CG method show that near the solution, eliminating the Riemannian terms gives $O(\epsilon^3)$ perturbations of the CG sequence, and therefore does not affect the quadratic rate of convergence. Furthermore, it can also be seen that eliminating the Polak-Ribière term $-\langle \tau G_{\text{old}}, G_{\text{new}}\rangle \big/ \|G_{\text{old}}\|^2$, yielding the Fletcher-Reeves algorithm, perturbs the CG sequence by $O(\epsilon^2)$ terms, which does not affect the quadratic rate of convergence. Thus the approximate CG methods discussed in 3.5.1 converge quadratically.

4 Examples: Insights and Applications

In this section, we consider ideas from the literature as applications of the framework and methodology developed in this paper. It is our hope that some readers who may be familiar with the algorithms presented here, will feel that they now really see them with a new deeper, but ultimately clearer understanding. It is our further hope that developers of algorithms that may somehow seem new will actually find that they also already fit inside of our geometrical framework. Finally, we hope that readers will see that the many algorithms that have been proposed over the past several decades are not just vaguely connected to each other, but are elements of a deeper mathematical structure. The reader who sees the depth and simplicity of Section 4.10, say, has understood our message.

4.1 Rayleigh Quotient Iteration

If A is a symmetric matrix, it is well known that Rayleigh quotient iteration (RQI) is a cubically convergent algorithm. It is easy to derive formulas and show that it is true, here we will explain our view of *why* it is true. Let $r(x)$ denote the Rayleigh quotient x^TAx and, abusing notation, let $r(\theta)$ denote the Rayleigh quotient on a geodesic with $\theta = 0$ corresponding to an eigenvector of A.

Here is the intuition. Without writing down any formulas, it is obvious that $r(\theta)$ is an even function of θ; hence $\theta = 0$ is an extreme point. Newton's optimization method, usually quadratically convergent, converges cubically on nondegenerate even functions. Keeping in mind that $A - r(x)I$ is the second covariant derivative of the Rayleigh quotient, inverting it must amount to applying Newton's method. Following this intuition, RQI must converge cubically. The intution is that simple.

Indeed, along a geodesic, $r(\theta) = \lambda\cos^2\theta + \alpha\sin^2\theta$ (we ignore the degenerate case $\alpha = \lambda$). The kth step of Newton's method for $r(\theta)$ is readily verified to be

$$\theta_{k+1} = \theta_k - \tfrac{1}{2}\tan(2\theta_k) = -\tfrac{4}{3}\theta_k^3 + O(\theta_k^5).$$

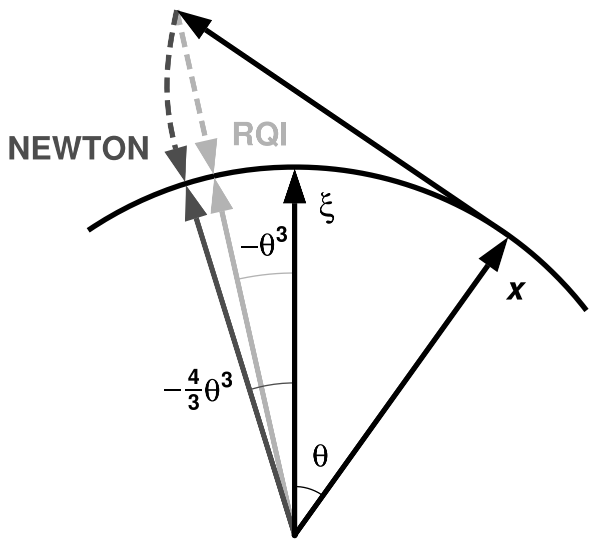

Figure 7: Cubic convergence of RQI and Newton's method applied to Rayleigh's quotient. The vector ξ is an eigenvector.

We think of updating θ as moving along the circle. If we actually moved tangent to the circle by the Newton update $-\frac{1}{2}\tan(2\theta_k)$ and then projected to the circle, we would have the Newton iteration

$$\theta_{k+1} = \theta_k - \arctan\Big(\tfrac{1}{2}\tan(2\theta_k)\Big) = -\theta_k^3 + O(\theta_k^5).$$

This is the mechanism that underlies Rayleigh quotient iteration. It "thinks" Newton along the geodesic, but moves along the tangent. The angle from the eigenvector goes from θ to $-\theta^3$ almost always. (Readers comparing with Parlett [65, Eq. (4-7-3)] will note that only positive angles are allowed in his formulation.)

When discussing the mechanism, we only needed one variable: θ. This is how the mechanism should be viewed because it is independent of the matrix, eigenvalues, and eigenvectors. The algorithm, however, takes place in x space. Since $A - r(x)I$ is the second covariant derivative of $r(x)$ in the tangent space at x, the Newton update δ is obtained by solving $\Pi(A - r(x)I)\delta = -\Pi A x = -(A - r(x)I)x$, where $\Pi = I - xx^T$ is the projector. The solution is $\delta = -x + y/(x^T y)$, where $y = (A - r(x)I)^{-1}x$. The Newton step along the tangent direction is then $x \to x + \delta = y/(x^T y)$, which we project to the unit sphere. This is exactly an RQI step. These ideas are illustrated in Figure 7.

One subtlety remains. The geodesic in the previous paragraph is determined by x and the gradient rather than x and the eigenvector. The two geodesics converge to each other by the inverse iteration process (almost always) allowing the underlying mechanism to drive the algorithm.

One trivial example where these issues arise is the generalization and derivation of Davidson's method [74, 26, 22]. In this context there is some question as to the interpretation of $D - \lambda I$ as a preconditioner. One interpretation is that it preconditions the eigenproblem by creating better eigenvalue spacings. We believe that there is a more appropriate point of view. In linear conjugate gradient for $Ax = b$, preconditioners are used to invert M which is an approximation to A (the Hessian of $\frac{1}{2}x^T Ax - x^T b$) against the gradient. This is an approximate Newton step. In nonlinear conjugate gradient, there is no consensus as to whether inverting the Hessian (which is approximated by $D - \lambda I$!) would constitute the ideal preconditioner, but it is a Newton step. Therefore with the link between nonlinear conjugate gradient preconditioning and approximate Newton step, we see that Davidson's method is deserving of being called a preconditioner from the conjugate gradient point of view.

4.2 Coordinate Singularities of Symmetric Matrices

An important open problem in numerical linear algebra is the complete understanding of the influence of singularities on computations [52, 17]. In this section we shall describe the singularity associated with multiple eigenvalues of symmetric matrices in terms of coordinate singularities, i.e., the breakdown of the coordinate representation. In Section 4.10, we will describe how understanding this coordinate singularity underlies a regularization approach to eigenvalue optimization.

Matrix factorizations are nothing more than changes in variables or coordinate changes. In the plane, Cartesian and polar coordinates both give orthogonal systems, but polar coordinates have a coordinate singularity at the origin. A small perturbation near the origin can violently change the angle coordinate. This is ill-conditioning. If the r coordinate goes through the origin we have a singularity of the form $|r|$.

Consider traceless, symmetric, 2-by-2 matrices:

$$A = \begin{pmatrix} x & y \\ y & -x \end{pmatrix}.$$

The positive eigenvalue is $r = \sqrt{x^2 + y^2}$, and one of the orthogonal eigenvectors is $\binom{\cos \frac{1}{2}\theta}{\sin \frac{1}{2}\theta}$, where $\tan \theta = y/x$. The conversion between matrix elements and the eigendecomposition is exactly the conversion from Cartesian to polar coordinates. Whatever ill-conditioning one associates with a symmetric matrix with two close eigenvalues, it is the same numerical difficulty associated with the origin in polar coordinates. The larger eigenvalue behaves like $|r|$ at the origin, and the eigenvector behaves like θ changing violently when perturbed. If one wants to think about all 2-by-2 symmetric matrices, add z as the trace, and the resulting interpretation is cylindrical coordinates.

We now generalize. Let S_n be the space of n-by-n symmetric matrices. Suppose that the largest p eigenvalues $\lambda_1, \ldots, \lambda_p$ coalesce. The corresponding eigenvectors are not uniquely determined, but the invariant subspace is. Convenient parameterizations are

$$\begin{aligned} S_n &\equiv \quad \text{Symmetric Matrices} &&= \mathbf{R}^p \times V_{n,p} \times S_{n-p} \\ S_{n,p} &\equiv \{\, S_n : \lambda_1 \text{ has multiplicity } p \,\} &&= \mathbf{R} \times G_{n,p} \times S_{n-p} \end{aligned}$$

That is, any symmetric matrix may be parameterized by its p largest eigenvalues, the corresponding eigenvectors in order, and the $(n-p)$-by-$(n-p)$ symmetric operator on the space orthogonal to these eigenvectors. To parameterize a symmetric matrix with eigenvalue λ of multiplicity p, we must specify the invariant subspace corresponding to this eigenvalue and, once again, the $(n-p)$-by-$(n-p)$ symmetric operator on the orthogonal subspace. It is worth mentioning that the parameters in these decompositions give an orthonormal system (surfaces with constant parameters intersect orthogonally). The codimension of $S_{n,p}$ in S_n is $p(p+1)/2-1$, obtained by adding $p-1$ (corresponding to $\lambda_2, \ldots, \lambda_p$) to $p(p-1)/2$ (the codimension of $G_{n,p}$ in $V_{n,p}$).

Another interpretation of the well-known fact that when eigenvalues coalesce, eigenvectors, but not invariant subspaces, are ill-conditioned, is that the Stiefel manifold collapses to the Grassmann manifold. As with polar coordinates we have a coordinate singularity corresponding to ill-conditioning near $S_{n,p}$. Near this set, a small perturbation will violently move the Stiefel component. The singularity associated with the coallescing of eigenvalues is very much the singularity of the function $f(x) = |x|$.

4.3 The CS Decomposition

The CS decomposition [45] should be recognized as the geodesic between two points on the Grassmann manifold. Any n-by-n orthogonal matrix Q may be written as

$$Q = \begin{pmatrix} V & 0 \\ 0 & U \end{pmatrix} \begin{pmatrix} C & -S & 0 \\ S & C & 0 \\ 0 & 0 & I \end{pmatrix} \begin{pmatrix} \tilde{V} & 0 \\ 0 & \tilde{U} \end{pmatrix}^T \tag{88}$$

for some p-by-p orthogonal matrices V and $\tilde{V}$ and $(n-p)$-by-$(n-p)$ orthogonal matrices U and $\tilde{U}$, and p angles θ_i where $C = \operatorname{diag}(\cos\theta_1, \ldots, \cos\theta_p)$ and $S = \operatorname{diag}(\sin\theta_1, \ldots, \sin\theta_p)$. Comparing this with the geodesic formula Eq. (65) and letting $\theta_i = t\sigma_i$ ($i = 1, \ldots, p$) where σ_i are the diagonal elements of Σ, we see that the first p columns of the CS decomposition traverse a geodesic emanating from $Y(0) = \binom{I}{0}$ (the origin). The next p columns give an orthogonal basis for the velocity vector along the geodesic (in fact, they are the orthogonal component of its polar decomposition).

As is well known, the θ_i are the principal angles between the subspaces spanned by the first p columns of Q and the origin. In general, let θ_i ($i = 1, \ldots, p$) be the principal angles between the two subspaces spanned by the columns of n-by-p orthonormal matrices Y_1 and Y_2, i.e., $U(\cos\Theta)V^T$ is the singular value decomposition of $Y_1^TY_2$, where Θ is the diagonal matrix of principal angles. Also let θ and $\sin\theta$ represent the p-vectors formed by the θ_i and $\sin\theta_i$. These principal angles provide several different definitions of the distance between two subspaces:

- **The arc length distance:** $d(Y_1, Y_2) = \|\theta\|_2$.
- **The chordal distance:** $\min_{Q_1,Q_2 \in O_p} \|Y_1Q_1 - Y_2Q_2\|_F = \|Y_1U - Y_2V\|_F = \|2\sin\frac{1}{2}\theta\|_2$.
- **Projection matrix 2-norm distance** [45]: $\|Y_1Y_1^T - Y_2Y_2^T\|_2 = \|\sin\theta\|_\infty$.
- **Projection matrix Frobenius-norm distance:** $\|Y_1Y_1^T - Y_2Y_2^T\|_F = \sqrt{2}\,\|\sin\theta\|_2$.

4.4 Conjugate Gradient for the Eigenvalue Problem

Conjugate gradient algorithms to minimize $\frac{1}{2}y^TAy$ (A symmetric) on the sphere ($p = 1$) is easy and has been proposed in many sources. The correct model algorithm for $p > 1$ presented in this paper is new. We were at first bewildered by the number of variations [2, 9, 33, 34, 3, 39, 35, 36, 69, 70, 38, 67, 19, 46, 93] most of which propose "new" algorithms for conjugate gradient for the eigenvalue problem. Most of these algorithms are for computing extreme eigenvalues and corresponding eigenvectors. It is important to note that none of these methods are equivalent to Lanczos [31]. It seems that the correct approach to the conjugate gradient algorithm for invariant subspaces ($p > 1$) has been more elusive. We are only aware of three papers [2, 70, 36] that directly consider conjugate gradient style algorithms for invariant subspaces of dimension $p > 1$. None of the proposed algorithms are quite as close to the new idealized algorithms as the $p = 1$ algorithms are. Each is missing important features which are best understood in the framework that we have developed. We discuss these algorithms below.

The simplest non-trivial objective function on the Grassmann manifold $G_{n,p}$ is the quadratic form

$$F(Y) = \frac{1}{2}\operatorname{tr} Y^TAY,$$

where A is a symmetric n-by-n matrix. It is well known that the solution to the minimization of F is the sum of the p smallest eigenvalues of A, with an optimal Y providing a basis for the invariant subspace corresponding to the p smallest eigenvalues.

To solve the eigenvalue problem, one may use the template directly from Section 3.3 after deriving the gradient

$$\nabla F(Y) = AY - Y(Y^TAY)$$

and the second covariant derivative of $F(Y)$:

$$\operatorname{Hess} F(\Delta_1, \Delta_2) = \operatorname{tr}\Bigl(\Delta_1^TA\Delta_2 - (\Delta_1^T\Delta_2)Y^TAY\Bigr).$$

The line minimization problem may be solved as p separate two-by-two problems in parallel, or it may be solved more completely by solving the $2p$-by-$2p$ eigenvalue problem. This does not follow the geodesic directly, but captures the main idea of the block Lanczos algorithm which in some sense is optimal [23, 24].

If one is really considering the pure linear symmetric eigenvalue problem then pure conjugate gradient style procedures must be inferior to Lanczos. Every step of all proposed non-preconditioned conjugate gradient algorithms builds vectors inside the same Krylov space in which Lanczos gives an optimal solution. However, exploring conjugate gradient is worthwhile. When the eigenvalue problem is non-linear or the matrix changes with time, the Lanczos procedure is problematic because it stubbornly remembers past information that perhaps it would do well to forget. (Linear conjugate gradient, by contrast, benefits from the memory of this past information.) Applications towards non-linear eigenvalue problems or problems that change in time drive us to consider the conjugate gradient method. Even the eigenvalue problem still plays a worthy role: it is the ideal model problem that allows us to understand the procedure much the way the Poisson equation on the grid is the model problem for many linear equation solvers.

Conjugate gradient on the sphere ($p = 1$) computes the smallest eigenvalue of a symmetric matrix A. Two papers [67, 19] consider imposing conjugacy through A. This is an unfortunate choice by itself because A is quite different from the Hessian $A - r(x)I$, where $r(x)$ is the Rayleigh quotient. A few authors directly consider conjugacy through the unconstrained Hessian [39, 93]. Others attempt to approximate conjugacy through the Hessian by using Polak-Ribiére or Fletcher-Reeves [9, 33, 34, 3, 35, 38, 46, 93, 69]. It is quite possible that most of these variations might well be competitive with each other and also our idealized algorithm, but we have not performed the numerical experiments because ultimately the $p = 1$ case is so trivial. A comparison that may be of more interest is the comparison with restarted Lanczos. We performed an informal numerical experiment that showed that the conjugate gradient method is always superior to two step Lanczos with restarts (as it should be since this is equivalent to the steepest descent method), but is typically slightly slower than four step Lanczos. Further experimentation may be needed in practice.

Turning to the $p > 1$ case, the three papers that we are aware of are [2, 70, 36]. The algorithm proposed in Alsén [2], has a built-in extra feature not in the idealized algorithm. Though this may not be obvious, it has one step of orthogonal iteration built in. This may be viewed as a preconditioning procedure giving the algorithm an advantage. The Sameh-Wisniewski [70] algorithm begins with many of the ideas of an idealized Grassmann algorithm, including the recognition of the correct tangent on the Grassmann manifold (though they only mention imposing the Stiefel constraint). Informal experiments did not reveal this algorithm to be competitive, but further experimentation might be appropriate. The more recent Fu and Dowling algorithm [36] imposes conjugacy through A and therefore we do not expect it to be competitive.

4.5 Conjugate Gradient for the Generalized Eigenvalue Problem

It is well known that the generalized eigenvalue problem $Ax = \lambda Bx$ may also be posed as a constrained optimization problem. Now we must find

$$\min \operatorname{tr} Y^T A Y$$

subject to the constraint that

$$Y^T B Y = I_p.$$

With the change of variables

$$\bar{Y} = B^{1/2} Y \tag{89}$$
$$\bar{\Delta} = B^{1/2} \Delta \tag{90}$$
$$\bar{A} = B^{-1/2} A B^{-1/2} \tag{91}$$

the problem becomes

$$\min \operatorname{tr} \bar{Y}^T \bar{A} \bar{Y} \quad \text{subject to} \quad \bar{Y}^T \bar{Y} = I_p.$$

The numerical algorithm will be performed on the non-overlined variables, but the algorithm will be mathematically equivalent to one performed on the overlined variables.

Notice that the condition on tangents in this new coordinate system is that

$$\Delta^T B Y = 0.$$

It is readily checked that the gradient of the trace minimization problem becomes

$$G = (B^{-1} - YY^T)AY$$

(note that $G^TBY = 0$).

Geodesics may be followed in any direction Δ for which $\Delta^T BY = 0$ by computing a compact variation on the SVD of Δ:

$$\Delta = U\Sigma V^T, \quad \text{where } U^TBU = I.$$

For simplicity, let us assume that Δ has full rank p. The V vectors are the eigenvectors of the matrix $\Delta^T B\Delta$, while the U vectors are the eigenvectors of the matrix $\Delta\Delta^T B$ corresponding to the non-zero eigenvalues. There is also a version involving the two matrices

$$\begin{pmatrix} 0 & 0 & \Delta \\ B & 0 & 0 \\ 0 & \Delta^T & 0 \end{pmatrix} \quad \text{and} \quad \begin{pmatrix} 0 & 0 & B \\ \Delta^T & 0 & 0 \\ 0 & \Delta & 0 \end{pmatrix}.$$

This SVD may be expressed in terms of the quotient SVD [45, 27].

Given the SVD, we may follow geodesics by computing

$$Y(t) = (\, YV \quad U\,) \begin{pmatrix} C \\ S \end{pmatrix} V^T.$$

All the Y along this curve have the property that $Y^TBY = I$. For the problem of minimizing $\frac{1}{2}\operatorname{tr} Y^TAY$, line minimization decouples into p two-by-two problems just as in the ordinary eigenvalue problem.

Parallel transport, conjugacy, and the second covariant derivative may all be readily worked out.

4.6 Electronic Structures Computations

In this section, we briefly survey a research area where conjugate gradient minimization of non-quadratic but smooth functions on the Stiefel and Grassmann manifolds arise, the *ab initio* calculation of electronic structure within the local density approximation. Such approaches use only the charge and mass of electrons and atomic nuclei as input and have greatly furthered understanding of the thermodynamic properties of bulk materials [12], the structure and dynamics of surfaces [51, 61], the nature of point defects in crystals [60], and the diffusion and interaction of impurities in bulk materials [84]. Less than ten years ago, Car and Parrinello [13] in a watershed paper proposed minimization through simulated annealing. Teter and Gillan [42, 83] later introduced conjugate gradient based schemes and demonstrated an order of magnitude increase in the convergence rate. These initial approaches, however, ignored entirely the effects of curvature on the choice of conjugate search directions. Taking the curvature into *partial* account using a generalization of the Riemannian projection led to a further improvement in computation times by over a factor of three under certain conditions [5].

Our ability to compute *ab initio*, using only the charge and mass of electrons and atomic nuclei as input, the behavior of systems of everyday matter has advanced greatly in recent years. However, the computational demands of the approach and the attendant bounds on the size of systems which may be studied (several hundred atoms) have limited the direct impact of the approach on materials and chemical engineering. Several *ab initio* applications which will benefit technology tremendously remain out of reach, requiring an order of magnitude increase in the size of addressable systems. Problems requiring the simultaneous study of thousands of atoms include defects in glasses (fiber optics communications), complexes of extended crystalline defects (materials' strength and processing), and large molecules (drug design).

The theoretical problem of interest is to find the smallest eigenvalue E_0 of the Schrödinger equation in the space of $3N$ dimensional skew-symmetric functions,

$$H\psi = E_0\psi,$$

where the Hamiltonian operator H is defined by

$$H = \sum_{1 \le n \le N} \Big(-\frac{1}{2}\nabla_n^2 + V_{ion}(r_n) \Big) + \frac{1}{2} \sum_{1 < n \ll m \le N} \frac{1}{\|r_n - r_m\|^2}.$$

Here, N is the number of electrons in the system under study, now typically on the order of several hundred, r_i is the position of the ith electron, $V_{ion}(r)$ is the potential function due to the nuclei and inner electrons, and the second summation is recognized as the usual Coulomb interactions. Directly discretizing this equation at M grid-points in space would lead to absurdly huge eigenvalue problems where the matrix would be M^N-by-M^N. This is not just a question of dense versus sparse methods, a direct approach is simply infeasible.

The fundamental theorems which make the *ab initio* approach tractable come from the density functional theory of Hohenberg and Kohn [50] and Kohn and Sham [55]. Density functional theory states that the ground states energy of a quantum mechanical system of interacting electrons and ions is equal to the solution of the problem of minimizing an energy function over all possible sets of N three-dimensional functions (electronic orbitals) obeying the constraints of orthonormality. Practical calculations generally use a finite basis to expand the orbitals, but for purposes of discussion, we may discretize the problem onto a finite spatial grid consisting of M points. The Kohn-Sham minimization then becomes,

$$\begin{aligned} E_0 &= \min_{X^T X = I_N} E(X) \\ &\equiv \min_{X^T X = I_N} \operatorname{tr}(X^T H X) + f\big(\rho(X)\big), \end{aligned} \tag{92}$$

where each column of X is a different electronic orbital sampled on the spatial grid, ρ is the vector $\rho_i(X) \equiv \sum_n |X_{in}|^2$, H is an M-by-M matrix (single-particle Hamiltonian), and f is a function which we leave unspecified in this discussion. In full generality the X are complex, but the real case applies for physical systems of large extent that we envisage for this application [66], and we, accordingly, take X to be real in this note.

Recent advances in computers have enabled such calculations on systems with several hundreds of atoms [4, 11]. Further improvements in memory and performance will soon make feasible computations with upwards of a thousand atoms. However, with growing interest in calculations involving larger systems has come the awareness that as the physical length of systems under study increases, the Hessian about the minimum of Eq. (92) becomes increasingly ill-conditioned and non-conjugate minimization approaches exhibit a critical slowing down [83]. This observation prompted workers [42, 83] to apply conjugate gradient concepts to the problem, and now dozens of researchers have written papers using some form of the conjugate gradient method. In particular, one has a Grassmann problem when the number of electrons in each state is constant (i.e., two one spin up and one spin down). This is what happens in calculations on semiconductors and "closed shell" atoms and molecules. Otherwise, one has a Stiefel problem such as when one has metals or molecules with partially filled degenerate states.

The framework laid out in this note may be of practical use to the *ab initio* density-functional community when the inner product computation through the Hessian of $E(X)$ is no more computationally complex to evaluate than calculating the energy function $E(X)$ or maintaining the orthonormality constraints $X^T X = I_N$. A suitable form for this inner product computation is

$$\frac{1}{2}\sum_{in,jm} Y_{in}\frac{\partial^2 E}{\partial X_{in}\partial X_{jm}}Z_{jm} = \operatorname{tr}\Big(Y^T(H+V)Z\Big) + \sum_{ij}\sigma_i\left(2\frac{\partial^2 f}{\partial\rho_i\partial\rho_j}\right)\tau_j \qquad (93)$$
$$- \operatorname{tr}\Big(X^T(H+V)(XY^TZ)\Big)$$

where V is the diagonal matrix defined by $V_{ij} = (\partial f/\partial\rho_i)\delta_{ij}$, $\sigma_i \equiv \sum_n Y_{in}X_{in}$, $\tau_i \equiv \sum_n Z_{in}X_{in}$. Written this way, the first two terms of Eq. (93) have the same form and may be evaluated in the same manner as the corresponding terms in Eq. (92), with σ and τ playing roles similar to ρ. The third term, coming from the curvature, may be evaluated in the same way as the first term of Eq. (93) once given the object XY^TZ, which is no more computationally complex to obtain than the Gram-Schmidt orthonormalization of an object like X.

4.7 Subspace Tracking

The problem of computing the principal invariant subspace of a symmetric or Hermitian matrix arises frequently in signal processing applications, such as adaptive filtering and direction finding [64, 72, 6, 73, 68]. Frequently, there is some time-varying aspect to the signal processing problem, and a family of time-varying principal invariant subspaces must be tracked. The variations may be due to either the addition of new information as in covariance matrix updating, a changing signal environment, or both. For example, compute the principal invariant subspace of either of the covariance matrices

$$R_k = R_{k-1} + x_k x_k^T \qquad k = 1, 2, \ldots, \text{ and } x_k \text{ is given} \qquad (94)$$

$$R(t) = \text{a continuous function of } t \qquad (95)$$

at every iteration or at discrete times. Eq. (94) typically arises from updating the sample covariance matrix estimate; Eq. (95), the more general case, arises from a time-varying interference scenario, e.g., interference for airborne surveillance radar [85, 77]. Solving this eigenvalue problem via the eigenvalue or singular value decompositions requires a large computational effort. Furthermore, only the span of the first few principal eigenvectors may be required, whereas decomposition techniques compute all eigenvectors and eigenvalues, resulting in superfluous computations. Approaches to this problem may be classified as standard iterative methods [44], methods exploiting rank 1 updates [64, 53, 73, 94, 58, 81, 14, 57], i.e., Eq. (94), Lanczos based methods [20, 91, 90], gradient based methods [64, 92, 10], conjugate gradient based methods [38, 19, 71, 93, 75, 36, 78], which are surveyed by Edelman and Smith [31], Rayleigh-Ritz based methods [37, 20], and methods that exploit covariance matrix or array structure [68, 91, 90].

If the subspace does not change quickly over (discrete or continuous) time, then the desired solution will be close to the previously computed solution, and an iterative gradient-based algorithm such as the conjugate gradient algorithm may be computationally attractive for the subspace tracking problem. Thus the subspace tracking problem is treated as a time-varying optimization problem. Other conjugate gradient methods for computing principal invariant subspaces in a signal processing context have appeared [19, 71, 93, 36]; however, these conjugate gradient techniques do not exploit the structure of the subspace constraint (see Section 4.4). Instead, we employ the conjugate gradient method on the Grassmann manifold, or an approximation of it discussed in Section 3.5. Comon and Golub [20] describe and compare a wide variety of different algorithms for the problem of exponential covariance matrix updates, with particular emphasis on Lanczos and gradient-based algorithms. Yang, Sarkar, and Arvas [93] survey some conjugate gradient algorithms applied to computing the principal invariant subspace of a fixed symmetric matrix. We adopt the general assumption that the matrix may change arbitrarily over time, but that it must vary "slowly enough" so that using a conjugate gradient based approach is computationally efficient. This last constraint is, of course, dependent upon the application. For the example of space-time adaptive processing for airborne radar with a rotating antenna, Smith [78] shows that this method is capable of tracking the principal invariant subspace of clutter interference; however, when the interference dimension p is increased to account for new interference eigenvalues, one does better to compute the eigendecomposition from scratch and use it to initiate a new subspace track.

4.8 Newton's Method for Invariant Subspace Computations

Methods for refining estimates for invariant subspace computations have been proposed by Chatelin [15, 16], Dongarra, Moler, and Wilkinson [29], and Stewart [80]. Demmel [28, Section 3] proposes a unified approach by showing that they are all solutions to a Riccati equation.

These algorithms, when applied to symmetric matrices, are all variations on our geometrical Newton algorithm and may be understood in this context. There is nothing special about the eigenvalue problem; Newton's method for any function on the Grassmann manifold yields a Sylvester equation in the tangent space. The reason a Riccati equation arises rather than a Sylvester equation is that the previous algorithms formulate the problem in an affine space with arbitrary constraints. Previous researchers knew the quadratic term in the Riccati equation belonged there, and knew that it somehow is related to the orthogonality constraints, but we now see that it is an artifact of a flat space derivation.

Let us take a closer look. Previous researchers proposed algorithms for invariant subspaces by asking for a solution to the matrix equation

$$AY - YB = 0$$

made nondegenerate by imposing the affine constraint

$$Z^T Y = I,$$

for some arbitrary choice of Z. In the Dongarra et al. case, Z may be obtained by inverting and transposing an arbitrary $p \times p$ minor of the $n \times p$ matrix Y. In Moler's Matlab notation `Z=zeros(n,p); Z(r,:)=inv(Y(r,:))'`, where `r` denotes a p-vector of row indices. For Stewart, $Z = Y(Y^T Y)^{-1}$.

A mathematically insightful approach would require no arbitrary choice for Z. We would simply specify the problem by performing Newton's method on the function $F(Y) = \frac{1}{2} \operatorname{tr} Y^T A Y$ on the Grassmann manifold. The stationary points of $F(Y)$ are the invariant subspaces. There is no need to specify any further constraints and there are no degeneracies. (Notice that asking for the solution to $AY = Y(Y^T A Y)$ subject to $Y^T Y = I$ is a degenerate problem.)

Newton's method requires the solution Δ to the Sylvester equation

$$\Pi\Big(A\Delta - \Delta(Y^T A Y)\Big) = -\Pi A Y,$$

where $\Pi = (I - YY^T)$ denotes the projection onto the tangent space of the Grassmann manifold and $G = \Pi A Y$ is the gradient. The solution is $\Delta = -Y + Z(Y^T Z)^{-1}$, where Z is the solution to the Sylvester equation $AZ - Z(Y^T A Y) = Y$. Y may be chosen so that $Y^T A Y$ is diagonal, yielding simultaneous Rayleigh quotient iterations. If we move along the tangent and project rather than the geodesic we have the iteration sending Y to the Q factor in the QR decomposition of Z.

4.9 Reduced Gradient Methods, Sequential Quadratic Programming, and Lagrange Multipliers

In this section, we generalize beyond the Stiefel and Grassmann manifolds to show how the language and understanding of differential geometry provides insight into well-known algorithms for general non-linear constrained optimization. We will show the role that geodesics play in these algorithms. In the next subsection, we will then apply the geometrical intuition developed here to directly formulate regularized sequential quadratic programs as is needed in eigenvalue optimization.

Here we study sequential quadratic programming (SQP) and reduced gradient methods (RGM). By SQP we mean the algorithm denoted as Newton SQP by Boggs and Tolle [7, p. 14], SQP by Nash and Sofer [59, p. 512], and QP-based projected Lagrangian by Gill, Murray, and Wright [41, p. 238, Eq. (6.41)]. By RGM, we specifically mean the method sometimes denoted as the reduced Hessian method [7, p. 25], other times simply denoted RGM [59, p. 520], and yet other times considered an example of an RGM [41, p. 221, Eq. (6.17)]. The difference is that RGM is derived based (roughly) on the assumption that one starts at a feasible point, whereas SQP does not.

We begin by interpreting geometrically the Lagrangian function as it is used in constrained optimization. Consider the optimization problem

$$\min_{x \in \mathbf{R}^n} f(x) \quad \text{given the constraint that} \quad h(x) = 0 \in \mathbf{R}^p. \tag{96}$$

For simplicity we consider the case where the level surfaces $h(x) = c$ are manifolds ($\partial h/\partial x$ has full rank everywhere) and we work with the Euclidean metric. In the Euclidean case, the formulations are routine in the optimization community, but we have not seen the geometric intuition (particularly geometric interpretations away from the optimization point and the role that geodesics play "behind-the-scenes") in the optimization references that we have consulted. Numerical Lagrange multiplier issues are discussed in [40] and [41], for example. In this paper, we give the new interpretation that the Hessian of the Lagrangian is the correct matrix for computing second derivatives along geodesics at every point, not only as an approximation to the result at the optimal point.

At every point $x \in \mathbf{R}^n$, it is possible to project the gradient of f onto the tangent space of the level surface through x. This defines a sort of flattened vector field. In terms of formulas, projection onto the tangent space (known as computing least-squares Lagrange multiplier estimates) means finding λ that minimizes the norm of

$$\mathcal{L}_x = f_x - \lambda \cdot h_x, \tag{97}$$

i.e.,

$$\lambda = f_x h_x^T (h_x h_x^T)^{-1}. \tag{98}$$

At every point $x \in \mathbf{R}^n$ (not only the optimal point) Lagrange multipliers are the coordinates of f_x in the normal space to a level surface of the constraint, i.e., the row space of h_x. (Our convention is that f_x is a 1-by-n row vector, and h_x is a p-by-n matrix whose rows are the linearizations of the constraints.)

If $x(t)$ is any curve starting at $x(0) = x$ that is constrained to the level surface at x, then $\mathcal{L}_x \dot{x}$ computes the derivative of f along the curve. (In other words, $\mathcal{L}_x$ is the first covariant derivative.) The second derivative of f along the curve is

$$\frac{d^2}{dt^2} f\big(x(t)\big) = \dot{x}^T \mathcal{L}_{xx} \dot{x} + \mathcal{L}_x \ddot{x}. \tag{99}$$

At the optimal point $\mathcal{L}_x$ is 0, and therefore $\mathcal{L}_{xx}$ is a second order model for f on the tangent space to the level surface. The vanishing of the term involving $\mathcal{L}_x$ at the optimal point is well-known.

The idea that we have not seen in the optimization literature and that we believe to be new is the geometrical understanding of the quantity at a non-optimal point: At any point at all, $\mathcal{L}_x$ is tangent to the level surface while $\ddot{x}(t)$ is normal when x is a geodesic. The second term in Eq. (99) conveniently vanishes here too because we are differentiating along a geodesic! Therefore, the Hessian of the Lagrangian has a natural geometrical meaning, it is the second derivative of f along geodesics on the level surface, i.e., it is the second covariant derivative in the Euclidean metric.

We now describe the RG method geometrically. Starting at a point x on (or near) the constraint surface $h(x) = 0$, the quadratic function

$$\mathcal{L}_x \dot{x} + \tfrac{1}{2} \dot{x}^T \mathcal{L}_{xx} \dot{x}$$

models f (up to a constant) along geodesics emanating from x. The $\dot{x}$ that minimizes this function is the Newton step for the minimum for f. Intrinsic Newton would move along the geodesic in the direction of $\dot{x}$ a length equal to $\|\dot{x}\|$. Extrinsically, we can move along the tangent directly from x to $x+\dot{x}$ and then solve a set of nonlinear equations to project back to the constraint surface. This is RGM. It is a static constrained Newton method in that the algorithm models the problem by assuming that the points satisfy the constraints rather than trying to dynamically move from level surface to level surface as does the SQP.

In SQP, we start on some level surface. We now notice that the quadratic function

$$\mathcal{L}_x\dot{x} + \tfrac{1}{2}\dot{x}^T\mathcal{L}_{xx}\dot{x} \tag{100}$$

can serve as a model not only the first and second covariant derivative of f on the level surface through x but also on level surfaces for points near x. The level surface through x is specified by the equation $h_x\dot{x} = 0$. Other parallel level surfaces are $h_x\dot{x} + c = 0$. The right choice for c is $h(x)$, which is a Newton step towards the level surface $h(x) = 0$. Therefore if the current position is x, and we form the problem of minimizing $\mathcal{L}_x\dot{x} + \frac{1}{2}\dot{x}^T\mathcal{L}_{xx}\dot{x}$ subject to the constraint that $h_x\dot{x} + h(x) = 0$, we are minimizing our model of f along geodesics through a level surface that is our best estimate for the constraint $h(x) = 0$. This is the SQP method.

Practicalities associated with implementing these algorithms are discussed in the aforementioned texts. Generalizations to other metrics (non-Euclidean) are possible, but we do not discuss this in detail. Instead we conclude by making clear the relationship between Lagrange multipliers and the Christoffel symbols of differential geometry.

To derive the geodesic equation, let $f(x) = x_k$, the kth coordinate of x. From Eq. (98), the Lagrange multipliers are $h_{x_k}^T(h_xh_x^T)^{-1}$. Since $f_{xx} = 0$ we then have that the geodesic equations are $\ddot{x}_k = \dot{x}^T\mathcal{L}_{xx}^k\dot{x}$ ($k = 1, \ldots, n$), where $\mathcal{L}_{xx}^k$ denotes, $-h_{x_k}^T(h_xh_x^T)^{-1}\cdot h_{xx}$, the Hessian of the Lagrangian function of x_k. The matrix $\Gamma^k = -\mathcal{L}_{xx}^k$ is the Christoffel symbol of differential geometry.

4.10 Eigenvalue Optimization

The geometric approach allows the formulation of sequential quadratic programming problems when the Lagrange multiplier formalism breaks down due to coordinate singularities. Specifically, the geometric insight from the previous subsection is that during the execution of a sequential quadratic program there are three types of directions. The first direction is towards the constraint manifold. SQP performs a Newton step in that direction. The second family of directions is parallel to the constraint manifold. SQP forms a quadratic approximation to the objective function in the parallel level surface obtained from the Newton step. The remaining directions play no role in an SQP and should be ignored.

Consider the problem of minimizing the largest eigenvalue of $A(x)$, an n-by-n real symmetric matrix-valued function of $x \in \mathbf{R}^m$ when it is known that at the minimum, exactly p of the largest eigenvalues coalesce. Overton and Womersley [63] formulated SQPs for this problem using Lagrange multipliers and sophisticated perturbation theory. The constraint in their SQP was that the p largest eigenvalues were identical. We will here consider the case of $m > p(p+1)/2$. One interesting feature that they observed was the non-differentiability of the largest eigenvalue at the optimum. Following the geometry of the previous section, a new algorithm without Lagrange multipliers may be readily devised. There will be no Lagrange multipliers because there will be no consideration of the third directions mentioned above.

We will write A for $A(x)$. Let $\Lambda = Y^TAY$, where the orthonormal columns of Y span the invariant subspace for the p largest eigenvalues of A, $\lambda_1, \ldots, \lambda_p$. We let $F(A) = \lambda_1$ and $\mathcal{L}(A) = \mathrm{tr}(\Lambda) = \lambda_1 + \cdots + \lambda_p$. Unlike the function $F(A)$, $\mathcal{L}(A)$ is a differentiable function at the optimal point. One might have guessed that this $\mathcal{L}(A)$ was the right $\mathcal{L}(A)$, but here is how one can logically deduce it:

The trick is to rely not on the Lagrange multiplier formalism of constraint functions, but rather on the geometry. Geometry has the power to replace a long complicated derivation with a short powerful one. Once the techniques are mastered, geometry provides the more intuitive understanding. There is no convenient $h(A)$ to express the constraint of multiple eigenvalues; artificially creating one leads to unnecessary complications due to the coordinate singularity when one moves from the level surface $h(A) = 0$ to another level surface. The right way to understand the coordinate singularity was described in Section 4.2. The direction of the Newton step must be the first order constraint of the coallescing of the eigenvalues. Using the notation of Section 4.2, the parallel directions are the tangent vectors of $S_{n,p}$. All other directions play no role.

The message from Section 4.9 is that whatever function we are interested in, we are only interested in the component of the gradient in the direction parallel to $S_{n,p}$. The very construction of a Lagrangian $\mathcal{L}$ then may be viewed as the construction of an appropriate function with the property that $\mathcal{L}_x$ is parallel to the tangent vectors of $S_{n,p}$. Of course the tangent space to $S_{n,p}$ (see Section 4.2) includes projection matrices of the form $\sum_{i=1}^{p} \alpha_i y_i y_i^T$, where y_i is the eigenvector corresponding to λ_i, only when the α_i are all equal. This corresponds to an identical shift of these eigenvalues. Therefore to form the correct gradient of the objective function $F(A) = \lambda_1$ everywhere, we should replace the true gradient, which is well known to be the spectral projector $y_1 y_1^T$, with its component in the direction YY^T, which is an $S_{n,p}$ tangent vector. Integrating, we now see that the act of forming the Lagrangian, which we now understand geometrically to mean replacing $y_1 y_1^T$ with YY^T (projecting the gradient to the surface of uniform shifts) amounts to nothing more than changing the objective function from $F(x)$ to $\mathcal{L}(x) = \mathrm{tr}(\Lambda) = \mathrm{tr}\, Y^TAY$. While one might have guessed that this was a convenient Langrangian, we deduced it by projecting the gradient of $f(x)$ on the tangent space of a level surface. The components of $f(x)$ that we removed implicitly would have contained the Lagrange multipliers, but since these components are not well defined at the coordinate singularity, it is of little value to be concerned with them.

Now we must explicitly consider the dependence of $\mathcal{L}$ on x. Our optimization step is denoted Δx, and $\dot{A}$ and $\ddot{A}$ respectively denote $[A_x \Delta x]$ and $[A_{xx} \Delta x \Delta x]$ (notation from [63]). It is easy to verify that

$$\mathcal{L}_x = \mathrm{tr}\, Y^T\dot{A}Y, \tag{101}$$

$$\mathcal{L}_{xx} = \mathrm{tr}(Y^T\ddot{A}Y + Y^T\dot{A}\dot{Y} + \dot{Y}^T\dot{A}Y), \tag{102}$$

where $\dot{Y}$ is the solution to

$$\dot{Y}\Lambda - A\dot{Y} = (I - YY^T)\dot{A}Y \tag{103}$$

that satisfies $Y^T\dot{Y} = 0$. The resulting sequential quadratic program over Δx is then

$$\min \mathcal{L}_x + \frac{1}{2}\mathcal{L}_{xx}, \tag{104}$$

subject to the linear constraint (on Δx) that

$$Y^T \dot{A} Y + \Lambda = \alpha I, \tag{105}$$

where the scalar α is arbitrary.

Let us explain all of these steps in more detail. The allowable $\dot{Y}$ are Grassmann directions, $Y^T\dot{Y} = 0$. Otherwise, we are not parallel to the constraint surface. Equation (101) is the derivative of Y^TAY. Noting that $AY = Y\Lambda$ and $Y^T\dot{Y} = 0$, two terms disappear. Equation (102) is trivial but we note the problem that we do not have an explicit expression for $\dot{Y}$, we only have A, Y and $\dot{A}$. Fortunately, the perturbation theory for the invariant subspace is available from Equation (103). It may be derived by differentiating $AY = Y\Lambda$ and substituting $\dot{\Lambda} = Y^T\dot{A}Y$.[2] The solution to Equation (103) is unique so long as no other eigenvalue of A is equal to any of $\lambda_1, \ldots, \lambda_p$.

The linear constraint on Δx is the one that infinitesimally moves us to the constraint surface. It is the condition that moves us to a diagonal matrix. Therefore, $\dot{\Lambda} = Y^T\dot{A}Y$ when added to Λ must be a scalar multiple of the identity. This is a linear condition on $\dot{A}$ and therefore on Δx. The α does not explicitly appear in the constraint.

5 Conclusions

This paper offers a new approach to the algorithms in numerical analysis involving orthogonality constraints. We have found that these algorithms should be understood as optimization algorithms in the correct geometrical setting; however, they rarely are.

As a concluding example of the insight gained, we propose a Grassmann based taxonomy for problems related to the symmetric eigenproblem. This taxonomy allows us to view algorithms not as isolated entities, but as objects with a coherent mathematical structure. It is our hope that developers of new algorithms and perturbation theories will benefit from the analytical approach that lead to our taxonomy.

In this taxonomy, algorithms are viewed as either restrictions or approximations of their parent. Ultimately, we have Newton's method on arbitrary Riemannian manifolds as the root. One can then restrict to a particular manifold such as the Stiefel manifold, or as we illustrate in 5 the Grassmann manifold. Along the vertical axis in the left column we begin with Newton's method which may be approximated first with PCG or approximate Newton methods, then pure conjugate gradient, and finally steepest descent. Moving from left to right the idealized algorithms are replaced with more practical versions that specialize for particular problems. The second column contains block algorithms, while the third contains single eigenvector related algorithms. This abstraction would not be possible without geometry.

[2] Alert readers may notice that this is really the operator used in the definition of "sep" in numerical linear algebra texts. The reader really understands the theory that we have developed in this paper if he or she now can picture the famous "sep" operator as a Lie bracket with a Grassmann tangent and is convinced that this is the "right" way to understand "sep" .

	$p > 1$	$p = 1$
Newton on the Grassmann Manifold	**Block Rayleigh Quotient**	**RQI**
	Newton Subspace Improvement Demmel 87 Chatelin 84 93 Dongarra, Moler, Wilkinson 83 Stewart 73	
PCG or approximate Newton on the Grassmann Manifold	**Nonlinear PCG** Gillan 89 Arias 92 Payne, Teter, Allan 92	
	Block Inverse Iteration	**Inverse Iteration** **Davidson's Method** **PCG** Perdon, Gambolati 89
	Block PCG Alsén 71	
CG on the Grassmann Manifold	**Nonlinear CG**	**Hessian Based** Geradin 71 **PR or FR Based** Bradbury, Fletcher 66 Fox, Kapoor 69 Fried 69, 72 Anderson 71 Haimi-Cohen, Cohen 87 Ruhe 74 Yang, Sarkar, Arvas 89 Fuhrmann, Liu 84 **Conjugate through A** Chen, Sarkar 86
	Linear Eigenvalue CG Sameh, Wisniewski 82 Fu, Dowling 95	
SD on the Grassmann Manifold	**Gradient Flows**	**Power Method**

Acknowledgments

The first author would like to thank Jim Demmel, Velvel Kahan, and Beresford Parlett who have stressed the importance of geometry in numerical linear algebra. We also thank Scott Axelrod, Victor Guillemin, and Shoshichi Kobayashi for a number of interesting conversations concerning differential geometry, and Roger Brockett for the idea of investigating conjugate gradient methods on symmetric spaces. We further wish to thank Dianne O'Leary, Mike Todd, Mike Overton, and Margaret Wright for equally interesting conversations in the area of numerical optimization. We are indebted to the San Diego crowd consisting of Scott Baden, Beth Ong, Ryoichi Kawai (University of Alabama), and John Weare for working together towards understanding the electronic structure problem. In particular the first author thanks John Weare for his hospitality during visits in San Diego where we explored the issue of conjugate gradient minimization. Steve Vavasis asked the penetrating question of how this relates to Lagrange multipliers which we answer in Section 4.9. Furthermore, Ross Lippert has made a number of valuable suggestions that are acknowledged in this paper. We are indebted to Gene Golub for inviting Steve Smith to the 1993 Householder Symposium in Lake Arrowhead, California which serendipitously launched this collaboration.

References

[1] David Abbott, editor. *The Biographical Dictionary of Scientists; Mathematicians.* P. Bedrick Books, New York, 1986.

[2] Britt Marie Alsén. Multiple step gradient iterative methods for computing eigenvalues of large symmetric matrices. Technical Report UMINF-15, University of Umeå, Sweden, 1971.

[3] Ingemar Andersson. Experiments with the conjugate gradient algorithm for the determination of eigenvalues of symmetric matrices. Technical Report UMINF-4.71, University of Umeå, Sweden, 1971.

[4] Tomás A. Arias and J. D. Joannopoulos. *Ab initio* theory of dislocation interactions: from close-range spontaneous annihilation to the long-range continuum limit. *Phys. Rev. Lett.*, 73:680–683, 1994.

[5] Tomás A. Arias, M. C. Payne, and John D. Joannopoulous. *Ab initio* molecular dynamics: Analytically continued energy functionals and insights into iterative solutions. *Physical Review Letters*, 71:1077–1080, 1992.

[6] G. Bienvenu and L. Kopp. Optimality of high resolution array processing using the eigensystem approach. *IEEE Trans. Acoust., Speech, Signal Processing*, ASSP-31(5):1235–1248, 1983.

[7] Paul T. Boggs and Jon W. Tolle. Sequential quadratic programming. *Acta Numerica*, pages 199–242, 1995.

[8] William M. Boothby. *An Introduction to Differentiable Manifolds and Riemannian Geometry.* Academic Press, Inc., New York, 2d edition, 1986.

[9] W. W. Bradbury and R. Fletcher. New iterative methods for solutions of the eigenproblem. *Numerische Mathematik*, 9:259–267, 1966.

[10] Roger W. Brockett. Dynamical systems that learn subspaces. In *Mathematical Systems Theory: The Influence of R. E. Kalman*, Berlin, 1991. Springer-Verlag.

[11] K. Brommer, M. Needels, B. E. Larson, and J. D. Joannopoulos. *Ab initio* theory of the Si(111)-(7×7) surface reconstruction: a challenge for massively parallel computation. *Phys. Rev. Lett.*, 68:1355–1358, 1992.

[12] F. Buda, R. Car, and M. Parrinello. Thermal expansion of *c*-Si via *ab initio* molecular dynamics. *Phys. Rev. B*, 41:1680–1683, 1990.

[13] R. Car and M. Parrinello. Unified approach for molecular dynamics and density-functional theory. *Physical Review Letters*, 55:2471–2474, 1985.

[14] Benoit Champagne. Adaptive eigendecomposition of data covariance matrices based on first-order perturbations. *IEEE Trans. Signal Processing*, 42(10):2758–2770, 1994.

[15] Françoise Chatelin. Simultaneous newton's iteration for the eigenproblem. *Computing Suppl.*, 5:67–74, 1984.

[16] Françoise Chatelin. *Eigenvalues of Matrices*. John Wiley & Sons, New York, 1993.

[17] Françoise Chatelin and Valérie Frayssé. *Lectures on Finite Precision Computations*. SIAM, Philadelphia, PA, 1996.

[18] Isaac Chavel. *Riemannian Geometry—A Modern Introduction*. The Cambridge University Press, 1993.

[19] Huanquin Chen, Tapan K. Sarkar, Soheil A. Dianat, and John D. Brulé. Adaptive spectral estimation by the conjugate gradient method. *IEEE Trans. Acoust., Speech, Signal Processing*, ASSP-34:272–284, 1986.

[20] P. Comon and G. H. Golub. Tracking a few extreme singular values and vectors in signal processing. *Proc. IEEE*, 78(8):1327–1343, 1990.

[21] Stephen H. Crandall. Rotordynamic software. In J. H. Kim and W.-J. Wang, editors, *Rotating Machinery Dynamics, Proceedings of the Third International Symposium on Transport Phenomena and Dynamics of Rotating Machinery (ISROMAC-3)*, pages 3–21. Hemisphere Publ. Corp., 1992.

[22] M. Crouzeix, B. Philippe, and M. Sadkane. The Davidson method. *SIAM J. Sci. Comp.*, 15:62–76, 1994.

[23] Jane K. Cullum. The simultaneous computation of a few of the algebraically largest and smallest eigenvalues of a large, sparse, symmetric matrix. *BIT*, 18:265–275, 1978.

[24] Jane K. Cullum and Ralph A. Willoughby. *Lanczos Algorithms for Large Symmetric Eigenvalue Computations*, volume 1. Birkhäuser, Stuttgart, 1985.

[25] Biswa Datta. *Numerical Linear Algebra and Applications*. Brooks/Cole, Pacific Grove, California, 1995.

[26] Ernest R. Davidson. The iterative calculation of a few of the lowest eigenvalues and corresponding eigenvectors of large real symmetric matrices. *J. Computational Physics*, 17:87–94, 1975.

[27] Bart De Moor. The Riemannian singular value decomposition. In Marc Moonen and Bart De Moor, editors, *Proc. 3d Intl. Workshop on SVD and Signal Processing*, volume 3, pages 61–78. Elsevier Science B. V., 1995.

[28] James W. Demmel. Three methods for refining estimates of invariant subspaces. *Computing*, 38:43–57, 1987.

[29] Jack J. Dongarra, Cleve B. Moler, and James H. Wilkinson. Improving the accuracy of computed eigenvalues and eigenvectors. *SIAM J. Num. Analysis*, 20:46–58, 1983.

[30] Alan Edelman. Large dense numerical linear algebra in 1993: The parallel computing influence. *J. Supercomputing Applications*, 7:113–128, 1993.

[31] Alan Edelman and Steven T. Smith. On conjugate gradient-like methods for eigen-like problems. *BIT*, to appear. See also *Proc. Linear and Nonlinear Conjugate Gradient-Related Methods*, eds. Loyce Adams and J. L. Nazareth. SIAM, Philadelphia, PA, 1996.

[32] Lars Eldén. Algorithms for the regularization of ill-conditioned least-squares problems. *BIT*, 17:134–145, 1977.

[33] R. L. Fox and M. P. Kapoor. A miminimization method for the solution of the eigenproblem arising in structural dynamics. In L. Berke, R. M. Bader, W. J. Mykytow, J. S. Przemieniecki, and M. H. Shirk, editors, *Proc. 2d Conf. Matrix Methods in Structural Mechanics*, pages 271–306. Wright-Patterson Air Force Base, Ohio, 1969.

[34] I. Fried. Gradient methods for finite element eigenproblems. *AIAA J.*, 7:739–741, 1969.

[35] I. Fried. Optimal gradient minimization scheme for finite element eigenproblems. *J. Sound Vib.*, 20:333–342, 1972.

[36] Z. Fu and E. M. Dowling. Conjugate gradient eigenstructure tracking for adaptive spectral estimation. *IEEE Trans. Signal Processing*, 43(5):1151–1160, 1995.

[37] Daniel R. Fuhrmann. An algorithm for subspace computation, with applications in signal processing. *SIAM J. Matrix Anal. Appl.*, 9(2):213–220, 1988.

[38] Daniel R. Fuhrmann and B. Liu. An iterative algorithm for locating the minimal eigenvector of a symmetric matrix. In *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, pages 45.8.1–4, 1984.

[39] M. Geradin. The computational efficiency of a new minimization algorithm for eigenvalue analysis. *J. Sound Vibration*, 19:319–331, 1971.

[40] Phillip E. Gill and Walter Murray. The computation of Lagrange multipier estimates for constrained minimization. *Math. Prog.*, 17:32–60, 1979.

[41] Phillip E. Gill, Walter Murray, and Margaret H. Wright. *Practical Optimization*. Academic Press, New York, 2d edition, 1981.

[42] M. J. Gillan. Calculation of the vacancy formation energy in aluminium. *J. Physics, Condensed Matter*, 1:689–711, 1989.

[43] S. Godecker and L. Colombo. An efficient linear scaling algorithm for tight-binding molecular dynamics. *Phys. Rev. Lett.*, 73, 1994.

[44] Gene Golub and Dianne O'Leary. Some history of the conjugate gradient and Lanczos methods. *SIAM Review*, 31:50–102, 1989.

[45] Gene H. Golub and Charles F. Van Loan. *Matrix Computations*. Johns Hopkins University Press, Baltimore, Maryland, 2d edition, 1989.

[46] R. Haimi-Cohen and A. Cohen. Gradient-type algorithms for partial singular value decomposition. *IEEE Trans. Pattern. Anal. Machine Intell.*, PAMI-9:137–142, 1987.

[47] Sigurdur Helgason. *Differential Geometry, Lie Groups, and Symmetric Spaces*. Academic Press, New York, 1978.

[48] Hermann Günther Grassmann. *Die Ausdehnungslehre*. Enslin, Berlin, 1862.

[49] M. R. Hestenes and E. Stiefel. Methods of conjugate gradients for solving linear systems. *J. Res. National Bureau of Standards*, 49:409–436, 1952.

[50] P. Hohenberg and W. Kohn. Inhomogeneous electron gas. *Physical Review*, 136:B864–B871, 1964.

[51] S. Ihara, S. L. Ho, T. Uda, and M. Hirao. *Ab initio* molecular-dynamics study of defects on the reconstructed Si(001) surface. *Physical Review Letters*, 65:1909–1912, 1990.

[52] William Kahan and James Demmel. Personal communications, 1994–96.

[53] Juha Karhunen. Adaptive algorithms for estimating eigenvectors of correlation type matrices. In *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, pages 14.6.1–4, 1984.

[54] S. Kobayashi and K. Nomizu. *Foundations of Differential Geometry*, volume 1 and 2. Wiley, New York, 1969.

[55] W. Kohn and L. J. Sham. Self-consistent equations including exchange and correlation effects. *Physical Review*, 140:A1133–A1138, 1965.

[56] Ross A. Lippert. Personal communication (see also `http://www.mit.edu/people/ripper/grass/grassmann.html`), 1995.

[57] G. Mathew, Vellenki U. Reddy, and Soura Dasgupta. Adaptive estimation of eigensubspace. *IEEE Trans. Signal Processing*, 43(2):401–411, 1995.

[58] Marc Moonen, Paul Van Dooren, and Joos Vandewalle. A singular value decomposition updating algorithm for subspace tracking. *SIAM J. Matrix Anal. Appl.*, 13(4):1015–1038, 1992.

[59] Stephen G. Nash and Ariela Sofer. *Linear and Nonlinear Programming*. McGraw-Hill, New York, 1995.

[60] M. Needels, J. D. Joannopoulos, Y. Bar-Yam, and S. T. Pantelides. Oxygen complexes in silicon. *Physical Review B*, 43:4208–4215, 1991.

[61] M. Needels, M. C. Payne, and J. D. Joannopoulos. High order reconstructions of the Ge(100) surface. *Physical Review B*, 38:5543–5546, 1988.

[62] Barrett O'Neill. *Semi-Riemannian Geometry with Applications to Relativity*. Academic Press, New York, 1983.

[63] Michael L. Overton and Robert S. Womersley. Second derivatives for optimizing eigenvalues of symmetric matrices. *SIAM J. Matrix Anal. Appl.*, 16:698–718, 1995.

[64] Norman L. Owsley. Adaptive data orthogonalization. In *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, pages 109–112, 1978.

[65] Beresford Parlett. *The Symmetric Eigenvalue Problem.* Prentice-Hall, Englewood Cliffs, NJ, 1980.

[66] M. C. Payne, M. P. Teter, D. C. Allan, T. A. Arias, and J. D. Joannopoulos. Iterative minimization techniques for *ab initio* total-energy calculations: Molecular dynamics and conjugate gradients. *Rev. Mod. Phys*, 64:1045–1097, 1992.

[67] Annamaria Perdon and Giuseppe Gambolati. Extreme eigenvalues of large sparse matrices by Rayleigh quotient and modified conjugate gradients. *Comp. Methods Appl. Mech. Engin.*, 56:251–264, 1986.

[68] Richard Roy and Thomas Kailath. ESPRIT—estimation of signal parameters via rotational invariance techniques. *IEEE Trans. Acoust., Speech, Signal Processing*, 37(7):984–995, 1989.

[69] Axel Ruhe. Iterative eigenvalue algorithms for large symmetric matrices. In *Numerische Behandlung von Eigenwertaaufgaben Oberwolfach 1972*, Intl. Series Numerical Math. Vol. 24, pages 97–115, 1974.

[70] Ahmed H. Sameh and J. A. Wisniewski. A trace minimization algorithm for the generalized eigenvalue problem. *SIAM J. Numerical Analysis*, 19:1243–1259, 1982.

[71] Tapan P. Sarkar and N. Sangruji. An adaptive nulling system for a narrow-band signal with a look direction constraint utilizing the conjugate gradient method. *IEEE Trans. Antennas, Propagation*, 37(7):940–944, 1989.

[72] Ralph O. Schmidt. Multiple emitter location and signal parameter estimation. *IEEE Trans. Antennas, Propagation*, AP-34(3):276–280, 1986. Reprinted from *Proc. RADC Spectrum Estimation Workshop*, Griffiss Air Force Base, NY, 1979.

[73] Robert Schreiber. Implementation of adaptive array algorithms. *IEEE Trans. Acoust., Speech, Signal Processing*, ASSP-34(5):1038–1045, 1986.

[74] Gerard L. G. Sleijpen and Henk A. Van der Vorst. A generalized Jacobi-Davidson iteration method for linear eigenvalue problems. *SIAM J. Matrix Anal. Appl.*, 17, 1996.

[75] Steven T. Smith. *Geometric Optimization Methods for Adaptive Filtering.* PhD thesis, Harvard University, Cambridge, Massachusetts, 1993.

[76] Steven T. Smith. Optimization techniques on Riemannian manifolds. *Fields Institute Communications*, 3:113–146, 1994.

[77] Steven T. Smith. Space-Time clutter covariance matrix computation and interference subspace tracking. In *Proc. 29th Asilomar Conf. Signals, Systems, Computers*, volume 2, pages 1193–97, 1995.

[78] Steven T. Smith. Linear and nonlinear conjugate gradient methods for adaptive processing. In *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, volume 3, pages 1834–37, 1996.

[79] Michael Spivak. *A Comprehensive Introduction to Differential Geometry*, volume 1–3. Publish or Perish, Inc., Houston, TX, 2d edition, 1979.

[80] G. W. Stewart. Error and perturbation bounds for subspaces associated with certain eigenvalue problems. *SIAM Review*, 15:752–764, 1973.

[81] G. W. Stewart. An updating algorithm for subspace tracking. *IEEE Trans. Signal Processing*, 40(6):1535–1541, 1992.

[82] E. Stiefel. Richtungsfelder und fernparallelismus in n-dimensionalem mannig faltigkeiten. *Commentarii Math. Helvetici*, 8:305–353, 1935–1936.

[83] Mike P. Teter, M. C. Payne, and D. C. Allan. Solution of Schrödinger's equation for large systems. *Phys. Review B*, 40:12255–12263, 1989.

[84] C. G. Van de Walle, Y. Bar-Yam, and S. T. Pantelides. Theory of hydrogen diffusion and reactions in crystalline silicon. *Physical Review Letters*, 60:2761–2764, 1988.

[85] James Ward. Space-Time adaptive processing for airborne radar. Technical Report 1015, DTIC No. ESC-TR-94-109, MIT Lincoln Laboratory, December 1994. See also *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, 1995.

[86] R. C. Ward and L. J. Gray. Eigensystem computation for skew-symmetric matrices and a class of symmetric matrices. *ACM Trans. Math. Softw.*, 4(3):278–285, 1978.

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

[88] J. H. Wilkinson. *The Algebraic Eigenvalue Problem.* Clarendon Press, Oxford, England, 1965.

[89] Y.-C. Wong. Differential geometry of Grassmann manifolds. *Proc. Natl. Acad. Sci. USA*, 57:589–594, 1967.

[90] Guanghan Xu and Thomas Kailath. Fast estimation of principal eigenspace using Lanczos algorithm. *SIAM J. Matrix Anal. Appl.*, 15(3):974–994, 1994.

[91] Guanghan Xu and Thomas Kailath. Fast subspace decomposition. *IEEE Trans. Signal Processing*, 42(3):539–551, 1994.

[92] J.-F. Yang and Mostafa Kaveh. Adaptive eigenspace algorithms for direction or frequency estimation and tracking. *IEEE Trans. Acoust., Speech, Signal Processing*, 36(2):241–251, 1988.

[93] X. Yang, T. P. Sarkar, and E. Arvas. A survey of conjugate gradient algorithms for solution of extreme eigen-problems of a symmetric matrix. *IEEE Trans. Acoust., Speech, Signal Processing*, 37(10):1550–1556, 1989.

[94] K.-B. Yu. Recursive updating the eigenvalue decomposition of a covariance matrix. *IEEE Trans. Signal Processing*, 39(5):1136–1145, 1991.