Extensions of Classical Multidimensional Scaling: Computational Theory[1]

Michael W. Trosset

Department of Mathematics, College of William & Mary, P.O. Box 8795, Williamsburg, VA 23187-8795.

Summary

Classical multidimensional scaling constructs a configuration of points that minimizes a certain measure of discrepancy between the configuration's interpoint distance matrix and a fixed dissimilarity matrix. Recent extensions of classical multidimensional scaling have replaced the fixed dissimilarity matrix with a closed and convex set of dissimilarity matrices. These formulations replace fixed dissimilarities with optimization variables (disparities) that are permitted to vary subject to application-specific constraints. For example, simple bound constraints are suitable for distance matrix completion problems (Trosset, 2000) and for inferring molecular conformation from information about interatomic distances (Trosset, 1998b); whereas order constraints are suitable for nonmetric multidimensional scaling (Trosset, 1998a). This paper describes the computational theory that provides a common foundation for these formulations.

[1]This research was supported by grant DMS-9622749 from the National Science Foundation and by the W. M. Keck Center for Computational Biology under Medical Informatics Training grant 1T1507093 from the National Library of Medicine. Many people contributed to this research. I am especially grateful to Pedro Barahona, Jan de Leeuw, Timothy Havel, Bruce Hendrickson, Ludwig Krippahl, Chi-Kwong Li, Roy Mathias, Thomas

1 Introduction

The influential survey of de Leeuw and Heiser (1982) defined *scaling* to be techniques for constructing a configuration of points in a target metric space from information about interpoint distances and *multidimensional scaling* (MDS) to be scaling in the case that the target space is Euclidean. Many specific formulations of MDS are possible; a useful organizing principle, adopted by de Leeuw and Heiser (1982) and by Trosset (1997), is to formulate MDS as a collection of optimization problems.

Classical MDS (Torgerson, 1952; Gower, 1966) can be formulated as an optimization problem with an objective function that is sometimes called the *strain criterion*. Recently, extensions of classical MDS (Trosset, 1998a, 1998b, 2000) have been developed in which the strain criterion is minimized in more general, application-specific settings. This paper describes the computational theory that provides a common foundation for these formulations.

We begin by recalling some familiar definitions:

Definition 1 *A hollow matrix is a square matrix whose diagonal elements vanish.*

Definition 2 *A dissimilarity matrix is a symmetric hollow matrix with nonnegative elements.*

We will denote dissimilarity matrices by $\Delta = [\delta_{ij}]$. The δ_{ij} are called dissimilarities.

Definition 3 *A p-dimensional Euclidean distance matrix is an $n \times n$ matrix $D = [d_{ij}]$ for which there exist $x_1, \ldots, x_n \in \Re^p$ such that $d_{ij} = \|x_i - x_j\|$.*

We will denote the set of $n \times n$ p-dimensional distance matrices by $\mathcal{D}_n(p)$. Given a set of points $x_1, \ldots, x_n \in \Re^p$, we store the coordinates of x_i in row i of the $n \times p$ *configuration matrix* X and denote the matrix of interpoint distances by $D(X)$.

It is obvious that a distance matrix is necessarily a dissimilarity matrix. Determining whether or not a specified dissimilarity matrix is a distance matrix is a famous problem in classical distance geometry. We state the standard solution of this problem, implicit in Torgerson's (1952) formulation of (classical) MDS and demonstrated by Gower (1966). The standard solution is a trivial modification of the solution independently discovered by Schoenberg (1935) and by Young and Householder (1938). Its statement requires some additional definitions:

Definition 4 *Let $A = [a_{ij}]$ and $B = [b_{ij}]$ denote $m \times n$ matrices. The Hadamard product of A and B is $A * B = [a_{ij} b_{ij}]$.*

Milligan, George Phillips, Richard Tapia, and Pablo Tarazaga.

Thus, if Δ is a matrix of dissimilarities, then $\Delta * \Delta = [\delta_{ij}^2]$ is a matrix of squared dissimilarities. Notice that $\Delta * \Delta$ is itself a dissimilarity matrix.

Definition 5 *The double-centering operator τ is a linear mapping on square matrices. If $A = [a_{ij}]$ is an $n \times n$ matrix, then $B = [b_{ij}] = \tau(A)$ is the $n \times n$ matrix defined by*

$$b_{ij} = -\frac{1}{2}\left(a_{ij} - \frac{1}{n}\sum_{j=1}^{n} a_{ij} - \frac{1}{n}\sum_{i=1}^{n} a_{ij} + \frac{1}{n^2}\sum_{i,j=1}^{n} a_{ij}\right).$$

This operation is called double centering because the row and column sums of $\tau(A)$ equal zero. Let I_n denote the $n \times n$ identity matrix and let $1_n \in \Re^n$ denote the vector whose elements are 1. Then double centering can also be represented by the matrix equation

$$\tau(A) = -\frac{1}{2}\left(I_n - \frac{1}{n}1_n 1_n'\right) A \left(I_n - \frac{1}{n}1_n 1_n'\right). \tag{1}$$

Notice that $\tau(A)$ is symmetric if A is symmetric. A detailed study of τ and related mappings was made by Critchley (1988).

Let Ω_n denote the set of symmetric positive semidefinite $n \times n$ matrices. Let $\Omega_n(p)$ denote the matrices in Ω_n whose rank is no greater than p. We now state the embedding theorem on which classical MDS is based:

Theorem 1 *Let Δ be an $n \times n$ dissimilarity matrix. Then $\Delta \in \mathcal{D}_n(p)$ if and only if $\tau(\Delta * \Delta) \in \Omega_n(p)$. Furthermore, if the $n \times p$ matrix X is such that $X'X = \tau(\Delta * \Delta)$, then $D(X) = \Delta$.*

Of course, if $B \in \Omega_n(p)$, then it is easy to use the spectral decomposition of B to construct an $n \times p$ matrix X such that $X'X = B$.

Given a set of dissimilarity matrices and a target dimension p, MDS is concerned with constructing a configuration of points in $\Re^p$ whose interpoint distance matrix matches or approximates one of the specified dissimilarity matrices. This concern can be expressed by specifying an optimization problem in which the objective function measures the discrepancy between a dissimilarity matrix and a distance matrix. In metric MDS, a single dissimilarity matrix is specified. In applications of distance geometry to molecular conformation, the set of dissimilarity matrices is specified by bound constraints on the dissimilarities and the target dimension is necessarily $p = 3$. In nonmetric MDS, the set of dissimilarity matrices is specified by order constraints on the dissimilarities.

Because of Theorem 1, a natural way to measure the discrepancy between a dissimilarity matrix Δ and a distance matrix D is

$$\|\tau(\Delta * \Delta) - \tau(D * D)\|_F^2,$$

where $\|\cdot\|_F$ denotes the Frobenius norm. The constraints typically imposed on the dissimilarities are such that the constraints on $\Delta * \Delta$ are closed and convex. If $D \in \mathcal{D}_n(p)$, then $\tau(D * D) \in \Omega_n(p) \subseteq \Omega_n$. Thus, the optimization problems that we will consider are of the general form

$$\begin{array}{ll} \text{minimize} & \|\tau(\Delta) - B\|_F^2 \\ \text{subject to} & \Delta \in \mathcal{C}_n, \\ & B \in \mathcal{B}_n, \end{array} \tag{2}$$

where $\mathcal{C}_n$ is a closed convex set of dissimilarity matrices and $\mathcal{B}_n$ is a closed subset of Ω_n. Because τ is a linear transformation, $\tau(\mathcal{C}_n)$ is a closed convex set; thus, Problem (2) is the problem of minimizing the distance between a (certain type of) closed convex set of symmetric matrices and a closed subset of the symmetric positive semidefinite matrices.

The set Ω_n is closed and convex. Hence, if $\mathcal{B}_n = \Omega_n$, then Problem (2) is convex. Unfortunately, there are few applications for which $\mathcal{B}_n = \Omega_n$. (For example, we have already noted that applications to computational chemistry require $\mathcal{B}_n = \Omega_n(3)$.) Accordingly, we will develop methods that can be applied to Problem (2) without assuming convexity.

In Section 2 we introduce certain subsets $\mathcal{B}_n \subseteq \Omega_n$. These subsets have a special structure, but are more general than the subsets $\Omega_n(p)$. We restrict attention to the case of a single dissimilarity matrix, i.e. $\mathcal{C}_n = \{\Delta\}$, and derive an explicit global solution of Problem (2). Our result generalizes a characterization of classical MDS due to Gower (1966) and Mardia (1978).

In Section 3 we restore the weaker assumption that $\mathcal{C}_n$ is a closed convex set of dissimilarity matrices. We discuss the general structure of Problem (2) and describe two general optimization strategies, variable alternation and variable reduction, for exploiting that structure.

In Sections 4 and 5 we establish some properties of the variable alternation and variable reduction approaches to Problem (2). We emphasize the latter, for which the theory is more satisfying. Using this approach, we propose a gradient projection method for finding local solutions of Problem (2). Section 6 presents a numerical example with simple bound constraints, in which case a more efficient algorithm is available. Section 7 concludes with a discussion of the prospects for developing more efficient methods for solving Problem (2).

2 Projection into Subsets of Ω_n

In this section we restrict attention to the case of a fixed dissimilarity matrix, i.e. $\mathcal{C}_n = \{\Delta\}$. In this case, Problem (2) specializes to the problem of finding the matrix in $\mathcal{B}_n \subseteq \Omega_n$ that is nearest the symmetric $n \times n$ matrix $T = \tau(\Delta)$. When $\mathcal{B}_n = \Omega_n$, this is the problem of projecting T into the closed convex cone of symmetric positive semidefinite matrices. In most applications, $\mathcal{B}_n$ is

not convex; often, however, it is still possible to find an element of $\mathcal{B}_n$ that is nearest T.

Toward that end, let K be any closed and convex subset of

$$\{x \in \Re^n \ : \ x_1 \geq \cdots \geq x_n \geq 0\} \tag{3}$$

and let $\Omega_n(K)$ denote the set of symmetric $n \times n$ matrices of the form $U\Gamma U'$, where U is orthogonal and $\mathrm{diag}(\Gamma) \in K$. Notice that, if

$$K = \{x \in \Re^n \ : \ x_1 \geq \cdots \geq x_{p+1} = \cdots = 0\},$$

then $\Omega_n(K) = \Omega_n(p)$. In this section we consider the following special case of Problem (2):

$$\begin{array}{ll} \text{minimize} & \|T - B\|_F^2 \\ \text{subject to} & B \in \Omega_n(K). \end{array} \tag{4}$$

The following result generalizes Theorem 14.4.2 in Mardia, Kent, and Bibby (1979):

Theorem 2 *Let $Q\Lambda Q'$ represent any spectral decomposition of T for which the eigenvalues $\lambda = diag(\Lambda)$ satisfy $\lambda_1 \geq \cdots \geq \lambda_n$. Let π denote projection into K, $\bar{\lambda} = \pi\lambda$, and $\bar{\Lambda} = diag(\bar{\lambda})$. Then $B^* = Q\bar{\Lambda}Q'$ is a global minimizer of Problem (4).*

Proof: Given $B \in \Omega_n(K)$, write $B = U\Gamma U'$ and $R = Q'U$, whereby

$$\|T - B\|_F^2 = \|Q\Lambda Q' - U\Gamma U'\|_F^2 = \|\Lambda - Q'U\Gamma U'Q\|_F^2 = \|\Lambda - R\Gamma R'\|_F^2.$$

For any fixed $\Gamma = \mathrm{diag}(\gamma)$,

$$\|\Lambda - R\Gamma R'\|_F^2 = \|\Lambda\|_F^2 - 2\left(\Lambda, R\Gamma R'\right)_F + \|\Gamma\|_F^2 \tag{5}$$

is minimized by choosing the orthogonal matrix $R = [r_{ij}]$ to maximize the inner product

$$\left(\Lambda, R\Gamma R'\right)_F = \mathrm{tr}\left(\Lambda R\Gamma R'\right) = \sum_{i=1}^n \lambda_i \left(\sum_{j=1}^n \gamma_j r_{ij}^2\right) = \sum_{i=1}^n \alpha_i \lambda_i. \tag{6}$$

Notice that $\alpha_i \geq 0$ and

$$\sum_{i=1}^n \alpha_i = \sum_{i=1}^n \sum_{j=1}^n \gamma_j r_{ij}^2 = \sum_{j=1}^n \gamma_j.$$

Because $\lambda_1 \geq \cdots \geq \lambda_n$, we maximize (6) by choosing α_1 as large as possible. Because $\gamma_1 \geq \cdots \geq \gamma_n \geq 0$ and R is orthogonal, this is accomplished by choosing $r_{11} = 1$, which forces $r_{12} = \cdots = r_{1n} = 0$. Similarly, we maximize

$\sum_{i=2}^{n} \alpha_i \lambda_i$ by choosing $r_{22} = 1$, and so on. It follows that, for any Γ, (6) is maximized, hence (5) is minimized, by choosing $R = I$. We conclude that

$$\begin{aligned} \|\Lambda - B\|_F^2 = \|\Lambda - R\Gamma R'\|_F^2 &\geq \|\Lambda - \Gamma\|_F^2 = \|\lambda - \gamma\|^2 \\ &\geq \|\lambda - \bar{\lambda}\|^2 = \|\Lambda - \bar{\Lambda}\|_F^2 \\ &= \|Q(\Lambda - \bar{\Lambda}Q'\|_F^2 = \|Q\Lambda Q' - Q\bar{\Lambda}Q'\|_F^2 \\ &= \|T - B^*\|_F^2, \end{aligned}$$

as claimed. □

We conclude this section by recording several expressions for the global minimum of Problem (4). For reasons that will become apparent in Section 5, we represent the minimum as a function, $F_K \circ \tau$, of Δ. Because τ is a linear function of Δ and $F_K(T)$ is the squared distance from T to $\Omega_n(K)$, $F_K \circ \tau$ is a continuous function of Δ. In general,

$$\begin{aligned} F_K \circ \tau(\Delta) &= \|Q\Lambda Q' - Q\bar{\Lambda}Q'\|_F^2 = \|Q(\Lambda - \bar{\Lambda})Q'\|_F^2 \\ &= \operatorname{tr}\left[(\Lambda - \bar{\Lambda})^2\right] = \sum_{i=1}^{n} (\lambda_i - \bar{\lambda}_i)^2. \end{aligned} \tag{7}$$

If $\Omega_n(K) = \Omega_n(p)$, then these expressions specialize to

$$\begin{aligned} F_p \circ \tau(\Delta) &= \sum_{i=1}^{p} [\lambda_i - \max(\lambda_i, 0)]^2 + \sum_{i=p+1}^{n} \lambda_i^2 \\ &= \sum_{i=1}^{p} \phi(\lambda_i) + \sum_{i=p+1}^{n} \psi(\lambda_i), \end{aligned} \tag{8}$$

where $\phi(t) = t^2 I_{(-\infty,0]}(t)$ and $\psi(t) = t^2$ are continuously differentiable functions.

3 Reducible Programming Formulations

Problem (2) has the general form

$$\begin{aligned} &\text{minimize} && f(a, b) \\ &\text{subject to} && a \in A,\ b \in B. \end{aligned} \tag{9}$$

This is an optimization problem in two sets of variables with no mixed constraints. Furthermore, in our application, if one fixes either a or b and optimizes the other, then the resulting subproblem is much easier to solve than Problem (9) in its entirety. (Specifically, if one fixes Δ and minimizes B in Problem (2), then one obtains Problem (4), for which Theorem 2 provides an explicit formula for a global solution. Alternatively, if one fixes B and

minimizes Δ, then one obtains a convex problem.) Nonlinear programs with these characteristics are sometimes called *problems whose variables separate*, e.g. by Golub and Pereyra (1973). Parks (1985), however, has argued persuasively for the superiority of the term *reducible*. Her comprehensive study of reducible nonlinear programming contains many valuable references. Following Trosset (1997), we describe two important strategies for exploiting reducible structure. Various studies have suggested that both strategies inevitably improve on methods that fail to exploit the structure of reducible nonlinear programs.

3.1 Variable Alternation

Variable alternation is the simple optimization strategy described in Figure 1. It has most commonly been used when each subproblem is projection into a subspace, in which case it is often called the Method of Alternating (orthogonal) Projections (MAP). MAP was first studied, in 1933, by von Neumann (1950), who considered the problem of projecting into the intersection of two closed linear subspaces of a Hilbert space. In this setting, alternately projecting into each subspace converges to projection into their intersection. A recent survey of the MAP literature was made by Deutsch (1992).

A more relevant use of variable alternation was made by Cheney and Goldstein (1959), who considered the problem of minimizing the distance between two closed convex sets in Hilbert space, say K_1 and K_2. Let P_i denote projection into K_i. Cheney and Goldstein established sufficient conditions for the sequence $(P_1P_2)^k x$ to converge to a point in K_1 nearest K_2. Specifically, convergence is assured either if one set is compact or if one set is finite-dimensional and the distance between the sets is attained.

Because of its simplicity and the fact that it produces a nonincreasing sequence of objective function values, variable alternation has appealed to a great many statisticians. A survey of variable alternation methods in statistics was made by de Leeuw (1994). Variable alternation has been widely used in algorithms for MDS, most notably by Takane, Young, and de Leeuw (1977), in which context it is usually called the method of alternating least squares. The Data Box Algorithm of Glunt, Hayden and Raydan (1993) applies variable alternation to a distance geometry problem with bound constraints.

Under very weak conditions, the convergence theory of Zangwill (1969) can usually be exploited to establish that every accumulation point of a sequence produced by variable alternation is a fixed point, i.e. a pair (a_*, b_*) for which $a_* = \operatorname{argmin} f(a, b_*)$ and $b_* = \operatorname{argmin} f(a_*, b)$. However, there is no general guarantee that (a_*, b_*) solves Problem (9). For instance, Trosset (1997) constructed a simple example in which a global *maximizer* was fixed under variable alternation.

In practice, variable alternation often does converge to a local minimizer. Unfortunately, the convergence rate of these methods is typically linear and

1. Fix a_0. Set $b_0 = \text{argmin } f(a_0, b)$ and $k = 1$.
2. Do until convergence:
   (a) $a_k = \text{argmin } f(a, b_{k-1})$
   (b) $b_k = \text{argmin } f(a_k, b)$
   (c) $k = k + 1$

Figure 1: A variable alternation strategy.

often painfully slow.

3.2 Variable Reduction

The optimization strategy described in Figure 2 is usually employed when one of the subproblems can be solved explicitly. By a variable reduction method, we mean *any* method for solving

$$\begin{array}{ll} \text{minimize} & \bar{f}(a) \\ \text{subject to} & a \in A. \end{array} \tag{10}$$

In contrast to variable alternation, variable reduction possesses the following property:

Theorem 3 *If a_* is a global (local) solution of Problem (10), then $(a_*, b(a_*))$ is a global (local) solution of Problem (9).*

Proof: If a_* is a global minimizer of Problem (10), then

$$f(a_*, b(a_*)) = \bar{f}(a_*) \leq \bar{f}(a) = f(a, b(a)) \leq f(a, b) \tag{11}$$

for all $(a, b) \in A \times B$. If a_* is a local minimizer of Problem (10), then there exists a neighborhood $N(a_*)$ in which $\bar{f}(a_*) \leq \bar{f}(a)$ and (11) holds for all $(a, b) \in N(a_*) \times B$. $\square$

Variable reduction eliminates one set of variables, albeit at the cost of complicating the objective function. A simple example examined by Trosset (1997) illustrates that variable reduction can be considerably more efficient than variable alternation. The potential difficulty with variable reduction is that the value function may not be differentiable, although when it is the derivatives of $\bar{f}$ usually have a very simple relation to the derivatives of f. Generalized differentiability of value functions has been the subject of extensive investigation in nonsmooth optimization, e.g. Section 6.5 in Clarke (1983).

1. For $\bar{a}$ fixed, define the *value function* by $b(\bar{a}) = \operatorname{argmin} f(\bar{a}, b)$.
2. Minimize the *variable projection functional* $\bar{f}(a) = f(a, b(a))$.

Figure 2: A variable reduction strategy.

For the special case of the semilinear least squares problem, differentiating the value function entails differentiating the Moore-Penrose pseudoinverse of a matrix. This was accomplished by Golub and Pereyra (1973), who compared the performance of Gauss-Newton algorithms on the full and reduced forms of this problem. Ruhe and Wedin (1980) determined that variable alternation on this problem exhibits linear convergence, whereas Gauss-Newton on the reduced problem exhibits superlinear convergence if Gauss-Newton on the full problem does.

Variable reduction has also been used in algorithms for MDS, most notably by Kruskal (1964a, 1964b). Variable reduction was first applied to the so-called additive constant problem by Saito (1978); de Leeuw and Heiser (1982) and Trosset, Baggerly, and Pearl (1996) subsequently suggested using the variable projection functional $F_p \circ \tau$. Trosset (1998b) proposed a formulation of nonmetric MDS using $F_p \circ \tau$ and Trosset (1997, 1998a, 2000) suggested $F_p \circ \tau$ for distance geometry problems with bound constraints.

4 Optimization by Variable Alternation

Henceforth, we restrict attention to $\mathcal{B}_n = \Omega_n(K)$ for K a closed convex subset of (3), i.e. we study

$$\begin{array}{ll} \text{minimize} & f(\Delta, B) = \|\tau(\Delta) - B\|_F^2 \\ \text{subject to} & \Delta \in \mathcal{C}_n, \\ & B \in \Omega_n(K). \end{array} \tag{12}$$

In this section we establish some consequences of applying variable alternation to Problem (12). Our analysis requires an additional hypothesis:

Assumption 1 *For any $\Delta^0 \in \mathcal{C}_n$, the level set*

$$\mathcal{L}_K(\Delta^0) = \{\Delta \in \mathcal{C}_n : F_K \circ \tau(\Delta) \leq F_K \circ \tau(\Delta^0)\}$$

is bounded.

In the applications to which we have alluded, Assumption 1 can be established by demonstrating that $\lim_{k\to\infty} F_K \circ \tau(\Delta^k) = \infty$ if $\{\Delta^k\}$ is unbounded.

The ensuing discussion requires a precise definition of variable alternation on Problem (12). The variable alternation subproblems are Problem (4) for $\Delta \in \mathcal{C}_n$ fixed and

$$\begin{array}{ll} \text{minimize} & \|\tau(\Delta) - B\|_F^2 \\ \text{subject to} & \Delta \in \mathcal{C}_n \end{array} \tag{13}$$

for $B \in \Omega_n(K)$ fixed. Notice that Problem (13) is the strictly convex problem of projecting B into the closed convex set $\tau(\mathcal{C}_n)$.

Let M_1 be any function that assigns a *single* global solution of Problem (4) to each fixed $\Delta \in \mathcal{C}_n$. Let M_2 be any function that assigns the unique solution of Problem (13) to each fixed $B \in \Omega_n(K)$.

Definition 6 *By variable alternation on Problem (12) from $\Delta^0 \in \mathcal{C}_n$, we mean the method of generating from Δ^0 the sequence $\{(\Delta^k, B^k)\}$, where $B^k = M_1(\Delta^k)$ and $\Delta^{k+1} = M_2(B^k)$. Furthermore, we say that (Δ^*, B^*) is a fixed point of variable alternation on Problem (12) if $B^* = M_1(\Delta^*)$ and $\Delta^* = M_2(B^*)$.*

We begin by considering the case of $\Omega_n(K) = \Omega_n$, in which case variable alternation on Problem (12) consists of alternating projections between the closed and convex sets $\tau(\mathcal{C}_n)$ and Ω_n.

Theorem 4 *If $\Omega_n(K) = \Omega_n$, then variable alternation on Problem (12) from any $\Delta \in \mathcal{C}_n$ converges to a global minimizer of Problem (12).*

Proof: Let $\{\Delta^k\}$ be the sequence of dissimilarity matrices constructed by variable alternation on Problem (12) from Δ^0. Let

$$\bar{f}_1(\Delta) = f(\Delta, M_1(\Delta)) = \|\tau(\Delta) - M_1(\Delta)\|_F^2 = F_K \circ \tau(\Delta). \tag{14}$$

Because

$$\bar{f}_1(\Delta^{k+1}) = f(\Delta^{k+1}, B^{k+1}) \leq f(\Delta^{k+1}, B^k) \leq f(\Delta^k, B^k) = \bar{f}_1(\Delta^k), \tag{15}$$

$\{\Delta^k\} \subset \mathcal{L}_K(\Delta^0)$ and variable alternation on Problem (12) from Δ^0 is equivalent to variable alternation from Δ^0 on the following problem:

$$\begin{array}{ll} \text{minimize} & f(\Delta, B) \\ \text{subject to} & \Delta \in \mathcal{L}_K(\Delta^0), \\ & B \in \Omega_n(K). \end{array} \tag{16}$$

Furthermore, it is transparent that the global minimizers of Problems (16) and (12) are identical.

By Assumption 1, the closed set $\mathcal{L}_K(\Delta^0)$ is bounded, hence compact. Therefore, it follows from Theorem 4 in Cheney and Goldstein (1959) that variable alternation on Problem (16) from Δ^0 converges to a global minimizer of Problem (16). □

We now consider the case of general $\Omega_n(K)$. Because Problem (12) is not convex except when $\Omega_n(K) = \Omega_n$, we obtain a weaker result.

Theorem 5 *Any sequence of dissimilarity matrices $\{\Delta^k\}$ obtained by applying variable alternation to Problem (12) will have at least one accumulation point. If Δ^* is an accumulation point of $\{\Delta^k\}$ and $B^* = M_1(\Delta^*)$, then (Δ^*, B^*) is a fixed point of variable alternation on Problem (12).*

Proof: Because Problem (4) may have multiple solutions, it is convenient to apply the theory of point-to-set maps. Insofar as we can do so consistently, we adopt the terminology and notation of Hogan (1973) and Zangwill (1969).

Following Hogan (1973), let Ω denote the constant point-to-set map defined by $\Omega(\Delta) \equiv \Omega_n(K)$. Since $\Omega_n(K)$ is a closed set, it is easily verified that Ω is continuous as a point-to-set map. Next let

$$v(\Delta) = \inf\{f(\Delta, B) : B \in \Omega_n(K)\} = \bar{f}_1(\Delta) = F_K \circ \tau(\Delta)$$

and

$$\bar{M}_1(\Delta) = \{B \in \Omega(\Delta) : f(\Delta, B) \leq v(\Delta)\}.$$

Since Ω is a continuous map and f is a continuous function, it follows from Theorem 8 in Hogan (1973) that the point-to-set map $\bar{M}_1$ is closed. (An elementary proof of this fact is possible, but the notation is cumbersome.)

Similarly, let Θ denote the constant point-to-set map defined by $\Theta(B) = \mathcal{C}_n$. Like Ω, Θ is continuous. Let

$$w(B) = \inf\{f(\Delta, B) : \Delta \in \Theta(B)\}$$

and

$$M_2(B) = \{\Delta \in \Theta(B) : f(\Delta, B) \leq w(B)\}.$$

Like $\bar{M}_1$, M_2 is a closed map. Moreover, we have already remarked that $M_2(B)$ contains a single element, so that M_2 is in fact a function.

Following Zangwill (1969), we now define an algorithmic point-to-set map A by $A = M_2 \circ \bar{M}_1$. By construction, $\Delta^{k+1} \in A(\Delta^k)$. Suppose that $\Delta^+ \in A(\Delta^c)$, in which case $\Delta^+ = M_2(B^c)$ for some $B^c \in \bar{M}_1(\Delta^c)$. Then, as in (15),

$$v(\Delta^c) = f(\Delta^c, B^c) \geq f(M_2(B^c), B^c) = f(\Delta^+, B^c) \geq v(\Delta^+). \tag{17}$$

Because $M_2(B^c)$ is unique, the first inequality in (17) is strict if and only if $\Delta^+ = M_2(B^c) = \Delta^c$. Thus, either $A(\Delta^c) = \{\Delta^c\}$ or $\Delta^c \notin A(\Delta^c)$. We define Δ^c to be a solution of A if $A(\Delta^c) = \{\Delta^c\}$ and note that $v(\Delta^c) > v(\Delta^+)$ if Δ^c is not a solution.

By Assumption 1, the closed set $\mathcal{L}_K(\Delta^0) = \{\Delta \in \mathcal{C}_n : v(\Delta) \leq v(\Delta^0)\}$ is bounded, hence compact. Since it follows from (17) that $v(\Delta^k) \leq v(\Delta^0)$ if Δ^k is generated by A, we conclude that any sequence generated by A must lie in a compact set and hence must have an accumulation point.

Since $v(\Delta)$ is the distance between Δ and $\Omega_n(K)$, the closed set $\bar{M}_1(\Delta)$ must be bounded, hence compact. It follows that, if $\Delta^j \to \Delta$, then $\cup_j \bar{M}_1(\Delta^j)$

is contained in a compact set. Hence, if $B^j \in \bar{M}_1(\Delta^j)$, then the sequence $\{B^j\}$ must contain a convergent subsequence. We can therefore apply Lemma 4.2 in Zangwill (1969) to conclude that $A = M_2 \circ \bar{M}_1$, the composition of closed maps, must itself be closed.

We now apply Convergence Theorem A in Zangwill (1969) to conclude that any accumulation point Δ^* of any sequence generated by A must satisfy $A(\Delta^*) = \{\Delta^*\}$. For any choice of M_1, we then have $\Delta^* = M_2 \circ M_1(\Delta^*)$, so $(\Delta^*, M_1(\Delta^*))$ is a fixed point of variable alternation on Problem (12). □

In practice, variable alternation on Problem (12) does seem to find solutions. However, it converges *very* slowly.

5 Optimization by Variable Reduction

In principle, there are two distinct variable reduction strategies for solving Problem (12). First, consider the variable projection functional

$$\bar{f}_2(B) = f(M_2(B), B) = \|\tau(M_2(B)) - B\|_F^2.$$

Because $\tau(M_2(B))$ is the projection of B into the closed convex set $\mathcal{C}_n$, $\bar{f}_2$ is a convex function. An explicit formula for $M_2(B)$ will rarely be available; however, in most applications, Problem (13) will be a quadratic programming problem and therefore $M_2(B)$ can be computed reasonably efficiently. This approach results in the following semidefinite programming problem:

$$\begin{array}{ll} \text{minimize} & \bar{f}_2(B) \\ \text{subject to} & B \in \Omega_n(K). \end{array} \tag{18}$$

If $\Omega_n(K) = \Omega_n$, then Problem (18) is convex and various optimization strategies are available. For example, for the problem of completing a partial Euclidean distance matrix, Alfakih, Khandani, and Wolkowicz (1998) proposed a related semidefinite programming formulation and a primal-dual interior-point algorithm for solving it. However, we are concerned with situations in which $\Omega_n(K)$ is a nonconvex subset of Ω_n, particularly $\Omega_n(K) = \Omega_n(p)$.

If a partial Euclidean distance matrix with $|E|$ known interpoint distances can be completed to an r-dimensional Euclidean distance matrix, then an algorithm presented by Alfakih and Wolkowicz (1998) can be used to construct an r^*-dimensional Euclidean distance matrix, where r^* is the greatest integer that does not exceed $(\sqrt{8|E|+1}-1)/2$. Unfortunately, it appears that $r^* > p$ in most applications. For example, if a molecule has n atoms, then inevitably $|E| > n-1$ and $r^* > 3$ if $n > 10$.

Returning to Problem (18), Tarazaga and Trosset (1993) studied methods of managing the constraint $B \in \Omega_n(p)$. They reparametrized B by writing $B = XX'$ for $X \in \Re^{n \times p}$, then introduced a penalty function to remove the resulting indeterminancy in the representation of B. Unfortunately, this

approach destroys many of the pleasant properties of Problem (18). Thus, the development of semidefinite programming methods that could manage rank restrictions directly would be of enormous value. In the absence of such methods, we explore an alternative approach.

Consider the variable projection functional defined by (14). Henceforth, we restrict attention to the case $\Omega_n(K) = \Omega_n(p)$ and consider the problem

$$\begin{array}{ll} \text{minimize} & F_p \circ \tau(\Delta) \\ \text{subject to} & \Delta \in \mathcal{C}_n. \end{array} \tag{19}$$

This problem has a closed convex feasible set, typically defined by linear constraints. (For example, in applications to computational chemistry, one usually imposes simple bound constraints on the δ_{ij}.) Here, in contrast to Problem (18), our concern lies with the objective function.

Theorem 6 *Let Δ be a dissimilarity matrix and let $T = \tau(\Delta)$ have eigenvalues $\lambda_1(T) \geq \cdots \geq \lambda_n(T)$. Then $F_p \circ \tau$ is continuously differentiable at Δ, unless $\lambda_p(T) = \lambda_{p+1}(T) \geq 0$.*

Proof: Because τ is linear, it suffices to show that F_p is continuously differentiable at T. From (8), we see that F_p is a spectral function of T, i.e. it depends only on the eigenvalues of T.

Given $\mu \in \Re^n$, define components $\mu_{(i)}$ of μ by $\mu_{(1)} \geq \cdots \mu_{(n)}$. Define $f_p : \Re^n \to \Re$ by

$$f_p(\mu) = \sum_{i=1}^{p} \phi\left(\mu_{(i)}\right) + \sum_{i=p+1}^{n} \psi\left(\mu_{(i)}\right).$$

Then f_p is symmetric, i.e. $f(P\mu) = f(\mu)$ for any permutation matrix P,

$$\frac{\partial}{\partial \mu_{(i)}} f_p(\mu) = \left\{ \begin{array}{cc} 2\mu_{(i)} & \mu_{(i)} \leq 0 \\ 0 & \mu_{(i)} \geq 0 \end{array} \right\}$$

for $i = 1, \ldots, p$, and

$$\frac{\partial}{\partial \mu_{(i)}} f_p(\mu) = 2\mu_{(i)}$$

for $i = p+1, \ldots, n$. It follows that ∇f_p exists (and is continuous) at μ unless $\mu_{(p)} = \mu_{(p+1)} \geq 0$.

Let $\mathrm{Diag}(\mu)$ denote the diagonal matrix with diagonal elements $\mu_1, \ldots, \mu_n$, let $\lambda(T) = (\lambda_1(T), \ldots, \lambda_n(T))'$, and write $T = Q\Lambda Q' = Q[\mathrm{Diag}(\lambda(T))]Q'$. Then Theorem 1.1 in Lewis (1996) states that F_p is differentiable at T if and only if f_p is differentiable at $\lambda(T)$, in which case

$$\begin{aligned} \nabla F_p(T) &= Q\left[\mathrm{Diag}\left(\nabla f_p(\lambda(T))\right)\right]Q' \\ &= Q\left[2\left(\Lambda - \bar{\Lambda}\right)\right]Q' \\ &= 2\left[Q\Lambda Q' - Q\bar{\lambda}Q'\right] \\ &= 2\left[T - B^*(T)\right], \end{aligned} \tag{20}$$

where $B^*(T) \in \Omega_n(p)$ denotes a solution of Problem (4) with $\Omega_n(K) = \Omega_n(p)$. □

It is instructive to consider why F_p is not everywhere differentiable. If T is such that Problem (4) with $\Omega_n(K) = \Omega_n(p)$ has a unique solution, $B^*(T)$, then F_p is differentiable at T and $-\nabla F_p(T)$ is the direction of steepest descent toward $B^*(T)$. If T is such that Problem (4) with $\Omega_n(K) = \Omega_n(p)$ has multiple solutions, then F_p is not differentiable at T. In this case, however, one can still compute the last expression in (20) for each solution, and each $-2[T - B^*(T)]$ so obtained is the direction of steepest descent toward the corresponding solution. Thus, (20) provides meaningful information even when it is not formally the gradient of F_p.

To facilitate developing algorithms for solving Problem (19), we now reinterpret $F_p \circ \tau$ as a function of the $m = n(n-1)/2$ subdiagonal elements of the dissimilarity matrix Δ and $\mathcal{C}_n$ as a closed convex subset of $\Re^m$. Then, writing

$$\delta = (\delta_{i_1 j_1}, \ldots, \delta_{i_m j_m})' \in \Re^m,$$

we can rewrite Problem (19) as:

$$\begin{array}{ll} \text{minimize} & F_p \circ \tau(\delta) \\ \text{subject to} & \delta \in \mathcal{C}_n. \end{array} \tag{21}$$

When it exists, it is not difficult to compute the gradient of $F_p \circ \tau$ with respect to the $\delta_{i_k j_k}$. We define $n \times n$ matrices $E^{rs} = [e_{ij}^{rs}]$ by setting $e_{rs}^{rs} = e_{sr}^{rs} = 1$ and all other elements equal to zero. From (1), it is easily seen that

$$\frac{\partial}{\partial \delta_{i_k j_k}} \tau(\delta) = \tau(E^{i_k j_k}).$$

Applying the chain rule yields

$$\frac{\partial}{\partial \delta_{i_k j_k}} F_p \circ \tau(\delta) = \left(\nabla F_p(\tau(\delta)), \tau(E^{i_k j_k}) \right)_F. \tag{22}$$

Then, using (20) with $S = B^*(\tau(\delta)) - \tau(\delta)$ and summing over indices denoted by $+$, some computation simplifies (22) to

$$\frac{\partial}{\partial \delta_{i_k j_k}} F_p \circ \tau(\delta) = \frac{2}{n^2} S_{++} - \frac{2}{n} \left(S_{i_k +} + S_{j_k +} \right) + 2 S_{i_k j_k}. \tag{23}$$

Now we recall that double centering results in matrices $T = \tau(\Delta)$ for which the row and column sums vanish. This is equivalent to stating that 1_n is an eigenvector of $T = Q \Lambda Q'$ with corresponding eigenvalue 0. The matrix $B^*(T) = Q \bar{\Lambda} Q'$ has the same eigenvectors as T and $\bar{\lambda}_i = 0$ if $\lambda_i = 0$. Hence, 1_n is an eigenvector of $B^*(T)$ with corresponding eigenvalue 0, i.e. the row and column sums of $B^*(T)$ also vanish. It follows that the row and column sums of S vanish, and consequently that (23) simplifies to

$$\frac{\partial}{\partial \delta_{i_k j_k}} F_p \circ \tau(\delta) = 2 S_{i_k j_k}. \tag{24}$$

Because the feasible set for Problem (21) is closed and convex, it is natural to contemplate a gradient projection method for its solution. Let $\delta^0 \in \mathcal{C}_n$ be given and let P denote projection into $\mathcal{C}_n$. Referring to (20), we define a gradient projection method for Problem (21) to mean the construction of a sequence

$$\delta^{k+1} = P\left(\delta^k - 2t_k\left[\tau(\delta^k) - B^*(\tau(\delta^k))\right]\right), \tag{25}$$

where t_k is obtained by a line search, i.e. by solving the univariate optimization problem

$$\begin{array}{ll} \text{minimize} & F_p \circ \tau\left[P\left(\delta^k - 2t\left[\tau(\delta^k) - B^*(\tau(\delta^k))\right]\right)\right] \\ \text{subject to} & t \geq 0. \end{array} \tag{26}$$

As we have noted, there is some ambiguity in the definition of $B^*(\tau(\delta))$ when Problem (4) has multiple solutions at $T = \tau(\delta)$. In practice, one might perform a single line search using an arbitrary solution or perform separate line searches using each solution and take the best δ so obtained.

We now borrow from McCormick and Tapia (1972):

Definition 7 *Given $\delta \in \mathcal{C}_n$, let $\bar{A}(\delta)$ denote the closure of the tangent cone to $\mathcal{C}_n$ at δ and let P_δ denote projection into $\bar{A}(\delta)$. Then δ is a constrained stationary point of Problem (21) if*

$$P_\delta\left[-\nabla F_p \circ \tau(\delta)\right] = 0.$$

Theorem 7 *Suppose that δ^* is an accumulation point of the gradient projection sequence defined by (25). If $F_p \circ \tau$ is differentiable at δ^*, then δ^* is a constrained stationary point of Problem (21). If Assumption 1 is satisfied, then the sequence will have at least one accumulation point.*

Proof: Because $\{F_p \circ \tau(\delta^k)\}$ is a nonincreasing sequence, Assumption 1 implies that the sequence $\{\delta^k\}$ is contained in a compact set and therefore has an accumulation point.

If $F_p \circ \tau$ is differentiable at the accumulation point δ^*, then the argument used to establish Theorem 1 in McCormick and Tapia (1972) is valid in a neighborhood of δ^* and establishes that δ^* is a constrained stationary point of Problem (21). □

Theorem 7 is stronger than Theorem 5 in the following sense: if δ^* is a constrained stationary point of Problem (21), then $(\tau(\delta^*), B^*(\tau(\delta^*)))$ is a fixed point under variable alternation on the corresponding Problem (12), but not conversely. Furthermore, although gradient projection methods are linearly convergent, sequences constructed by gradient projection on Problem (21) tend to converge *much* more rapidly than sequences constructed by variable alternation on Problem (12).

6 Examples with Bound Constraints

Trosset (1998a) considered the application of MDS to the problem of inferring 3-dimensional structure from information about a molecule's interatomic distances. Suppose that L and U are dissimilarity matrices of lower and upper bounds on the interatomic distances. Then the embedding problem is to determine an $n \times 3$ configuration matrix X such that $D(X) \doteq \Delta \in [L, U]$.

Using the methods of Section 5, we can formulate the embedding problem as a nonlinear optimization problem with simple bound constraints:

$$\begin{array}{ll} \text{minimize} & F_3 \circ \tau(\Delta) \\ \text{subject to} & \Delta \in [L * L, U * U]. \end{array} \tag{27}$$

We find a 3-dimensional structure that is consistent with the interatomic distance information by first finding a matrix Δ^* of squared dissimilarities that solves Problem (27), then using classical MDS to recover X from Δ^*.

Problem (27) requires $m = n(n-1)/2$ variables, one for each interatomic distance. For large molecules, this is a great many variables. In contrast, the more conventional parametrization by Cartesian coordinates, $D = D(X)$, requires only $3n-6$ variables (one for each atomic coordinate, less six to remove translational and rotational indeterminancy). However, certain advantages appear to accrue from allowing each (squared) interatomic dissimilarity to vary independently of the others—additional memory is the price that we pay for these advantages.

For n small, as in Trosset (2000), one can solve Problem (27) with a quasi-Newton method for bound-constrained optimization. For molecules with hundreds or thousands of atoms (n large), as is typical in structural molecular biology, the memory requirements of such methods are prohibitive. The typical Hessian matrix is completely dense; hence, to store a symmetric approximation of the Hessian matrix requires $(n^4-2n^3+3n^2-2n)/8$ variables, which is unrealistic for large molecules.

One possibility is to try to solve Problem (27) with a first-order method, e.g. the gradient projection method discussed in Section 5. Unfortunately, $F_3 \circ \tau$ does not have much curvature near solutions. This can be inferred from the well-known fact that classical MDS is extremely stable under perturbations of the dissimilarity data, a phenomenon that has been studied by Mardia (1978) and by Sibson (1979). Hence, we would expect first-order methods to be inefficient.

In light of the preceding, our computational dilemma is to balance the need for second-order information with the expense of managing it. A natural compromise is a limited memory method. Such methods use a small number of recent updates to approximate the current Hessian matrix of the objective function. For the numerical experiments reported below, we used a limited memory algorithm that was developed by Byrd, Lu, Nocedal, and Zhu (1995) for solving large nonlinear optimization problems with simple bound

constraints. Their Fortran 77 implementation (L-BFGS-B, version 2.1) was described by Zhu, Byrd, Lu, and Nocedal (1994) and is known to run very efficiently.

For large n, the computational expense of repeatedly evaluating $F_3 \circ \tau$ and $\nabla[F_3 \circ \tau]$ dominates the computational expense of solving Problem (27). Considerable care must be exercised to evaluate these quantities efficiently. The expressions derived in previous sections, e.g. (8), although convenient for mathematical analysis, should not be used for computation. These expressions involve the entire spectral decomposition of the symmetric $n \times n$ matrix $\tau(\Delta)$. In fact, only the $p = 3$ largest eigenvalues (and the corresponding eigenvectors) are required.

Let $\lambda_1, \lambda_2, \lambda_3$ denote the $p = 3$ largest eigenvalues of $\tau(\Delta)$ and let q_1, q_2, q_3 denote the corresponding eigenvectors. Then

$$\begin{aligned} F_3 \circ \tau(\Delta) &= \sum_{i=1}^{3} \left[\lambda_i - \max(\lambda_i, 0)\right]^2 + \sum_{i=4}^{n} \lambda_i^2 \\ &= \sum_{i=1}^{n} \lambda_i^2 - \sum_{i=1}^{3} \left[\lambda_i - \max(\lambda_i, 0)\right]^2 \\ &= \|\tau(\Delta)\|_F^2 - \sum_{i=1}^{3} \left[\lambda_i - \bar{\lambda}_i\right]^2 \end{aligned}$$

and

$$B^*(\tau(\Delta)) = \sum_{i=1}^{3} \bar{\lambda}_i q_i q_i',$$

so evaluating $F_3 \circ \tau$ and $\nabla[F_3 \circ \tau]$ only requires computing $\lambda_1, \lambda_2, \lambda_3$ and q_1, q_2, q_3.

Thus, the following computational considerations are crucial for efficiently solving Problem (27):

- Each function/gradient evaluation requires computing $p = 3$ extreme eigenvalues of a symmetric $n \times n$ matrix.
- We would like to solve problems for which n is in the hundreds, possibly several thousands.
- As optimization progresses and the iterates begin to converge, $\tau(\Delta^{k+1})$ will be close to $\tau(\Delta^k)$.

These considerations argue for using an iterative (Arnoldi/Lanczos) method to compute eigenvalues and eigenvectors. As described by Trefethen and Bau (1997), these methods (Lanczos is Arnoldi in the special case of a symmetric/Hermitian matrix) are especially efficient ways of computing a small number of extreme eigenvalues when n is large. Furthermore, their efficiency varies with the choice of the starting vector used to initialize the method. Therefore, as Lehoucq, Sorensen, and Yang (1997) commented,

> "If this eigenvalue calculation is one of a sequence of closely related problems, then convergence may be accelerated if a suitable starting vector is specified. Typical choices in this situation might be to use the final value of the starting vector from the previous eigenvalue calculation..."
>
> (ARPACK Users' Guide, Section 2.3.7, page 18)

For the numerical experiments reported below, we computed eigenvalues and eigenvectors using ARPACK, a Fortran 77 implementation of an implicitly restarted Arnoldi method developed at Rice University (Sorensen, 1992; Lehoucq, 1995; Lehoucq and Sorensen, 1996; Lehoucq, Sorensen, and Yang, 1997) and available at `http://www.caam.rice.edu/`.

We now elaborate on the computational details of some numerical experiments reported by Krippahl, Trosset, and Barahona (2001). These experiments were performed on a workstation with a 600 MHz Pentium III processor and 256MB of RAM, using version 5.0 of Absoft's ProFortran for Linux compiler. Machine precision was determined to be approximately `1.084D-19` and all floating point operations were performed in double precision.

The 3-dimensional structures of the following five molecules were obtained from the Protein Data Bank:

	n	m	$m(7)$
Desulforedoxin (monomer)	260	33670	5951
Trypsin Inhibitor	448	100128	11613
Mutant P53 Anti-Oncogene	514	131841	12938
Phosphotransferase	639	203841	17206
Barstar Mutant	693	239778	18996

Here, n is the number of atoms, $m = n(n-1)/2$ is the number of atomic pairs, and $m(7)$ is the number of pairs in which the interatomic distance $d_{ij} \leq 7.00$ angstroms. If $d_{ij} \leq 7.00$, then we set $\ell_{ij} = d_{ij} - 0.01$ and $u_{ij} = d_{ij} + 0.01$; if $d_{ij} > 7.00$, then we set $\ell_{ij} = 7.00$ and determined u_{ij} from repeated applications of the triangle inequality. The resulting L and U thus mimicked the lower and upper bounds that might be derived from the molecule's chemical structure and NMR spectroscopy, in that only information about nearby atoms is encoded. Of course, our L and U included considerably more—and considerably more accurate—information than would be available in practice.

Each attempt to solve Problem (27) began with the construction of a plausible configuration that was used to determine Δ^0, the initial matrix of squared dissimilarities. This configuration was determined by constraint programming, as described by Krippahl and Barahona (1999). The following

table summarizes what resulted from starting L-BFGS-B from Δ^0:

n	Tnf	Time	$F_3 \circ \tau$	Nact	MaxViol
260	1427	425	0.00286	539	0.00547
448	4757	4703	0.05879	1067	0.08208
514	3763	4888	0.07370	1489	0.02526
639	2662	5398	0.03191	1874	0.00488
693	2824	6621	0.05103	2168	0.00791

The L-BFGS-B algorithm took `Tnf` function/gradient evaluations and `Time` seconds to solve Problem (27). Notice that the number of seconds per evaluation increases with n, from 0.30 for $n = 260$ to 2.34 for $n = 693$. In each case, the search was terminated because the relative reduction in the objective function fell below a specified threshhold.

Each problem was constructed to have a global minimum of $F_3 \circ \tau(\Delta^*) = 0$. By the usual standards of numerical optimization, our solutions are of fairly low accuracy. However, considerable experimentation suggests that attaining this level of accuracy in solving Problem (27) results in configurations that are quite acceptable by the standards of computational chemistry.

For each problem, the L-BFGS-B algorithm returned an $n \times n$ squared dissimilarity matrix $\Delta^{**} \in [L * L, U * U]$. Of the m squared dissimilarities δ_{ij}^{**}, a total of `Nact` equalled the corresponding ℓ_{ij}^2 or u_{ij}^2. (These ij identify the active constraints.) The matrix Δ^{**} was almost—but not quite—a 3-dimensional squared distance matrix. The final configuration was extracted from it by classical MDS. Inevitably this procedure slightly violated several constraints, by amounts no greater than `MaxViol`. In each case, the maximal violation was quite small—negligible in comparison to the accuracy of the bounds that might be imposed in practice. These results offer considerable promise that the methods described herein can be used to solve actual problems in structural molecular biology.

7 Discussion

Various extensions of classical MDS can be formulated as special cases of the general problem of minimizing the (squared) distance (in Frobenius norm) between certain closed convex sets of symmetric $n \times n$ matrices and certain subsets of the symmetric positive semidefinite $n \times n$ matrices. The former sets are obtained by linear transformation of closed convex sets of dissimilarity matrices. In most applications, the latter sets are defined by rank restrictions. We have presented several formulations of this general problem; at present, our preferred formulation is Problem (21).

Let us reflect on some of the properties possessed by Problem (21):

1. The number of variables, $m = n(n-1)/2 = O(n^2)$, is potentially huge. In applications to computational chemistry, n may be several hundreds, if not thousands.

2. The objective function is (usually) continuously differentiable and analytic first derivatives are available. The cost of a function or gradient evaluation is effectively the cost of computing the p largest eigenvalues and the corresponding eigenvectors of a symmetric $n \times n$ matrix.

3. The objective function does not have much curvature near solutions.

4. The Hessian matrix is completely dense and $m(m+1)/2 = O(n^4)$ variables are required to specify it (or an approximation thereof).

5. The feasible set is closed and convex. In most applications, it is polyhedral. In some applications, it is defined by specifying simple bounds on each variable.

There is an evident tension between certain of these properties, in that we would like to exploit information about the curvature of the objective function to construct algorithms with fast local convergence, but such information is expensive to obtain and to manage. Although we have suggested a gradient projection method for general use, first-order methods may be inadequate when the number of variables is large and/or accurate solutions are required.

As we argued in Section 6, limited memory quasi-Newton methods appear to be admirably suited to the task of solving Problem (21). At present, such methods are available for problems with bound constraints, e.g. Problem (27), but not for more general linear inequality constraints. Hence, existing limited memory methods are applicable to distance matrix completion and molecular conformation problems, but not to nonmetric MDS. In the future, we hope to remedy this situation by extending limited memory methods to accommodate order constraints.

References

[1] Alfakih, A. Y., Khandani, A., and Wolkowicz, H. (1998). Solving Euclidean distance matrix completion problems via semidefinite programming. *Computational Optimization and Applications*, 12:13–30.

[2] Alfakih, A. Y. and Wolkowicz, H. (1998). On the embeddability of weighted graphs in Euclidean spaces. Research Report CORR 98-12, Department of Combinatorics & Optimization, University of Waterloo, Waterloo, Ontario N2L 3G1, Canada.

[3] Byrd, R. H., Lu, P., Nocedal, J., and Zhu, C. (1995). A limited memory algorithm for bound constrained optimization. *SIAM Journal on Scientific Computing*, 16:1190–1208.

[4] Cheney, W. and Goldstein, A. A. (1959). Proximity maps for convex sets. *Proceedings of the American Mathematical Society*, 10:448–450.

[5] Clarke, F. H. (1983). *Optimization and Nonsmooth Analysis*. John Wiley & Sons, New York.

[6] Critchley, F. (1988). On certain linear mappings between inner-product and squared-distance matrices. *Linear Algebra and Its Applications*, 105:91–107.

[7] de Leeuw, J. (1994). Block relaxation algorithms in statistics. In Bock, H. H., Lenski, W., and Richter, M. M., editors, *Information Systems and Data Analysis*, pages 308–325. Springer, Berlin.

[8] de Leeuw, J. and Heiser, W. (1982). Theory of multidimensional scaling. In Krishnaiah, P. R. and Kanal, I. N., editors, *Handbook of Statistics*, volume 2, chapter 13, pages 285–316. North-Holland Publishing Company, Amsterdam.

[9] Deutsch, F. (1992). The method of alternating orthogonal projections. In Singh, S. P., editor, *Approximation Theory, Spline Functions and Applications*, pages 105–121. Kluwer, Netherlands.

[10] Glunt, W., Hayden, T. L., and Raydan, M. (1993). Molecular conformations from distance matrices. *Journal of Computational Chemistry*, 14:114–120.

[11] Golub, G. H. and Pereyra, V. (1973). The differentiation of pseudo-inverses and nonlinear least squares problems whose variables separate. *SIAM Journal on Numerical Analysis*, 10:413–432.

[12] Gower, J. C. (1966). Some distance properties of latent root and vector methods in multivariate analysis. *Biometrika*, 53:315–328.

[13] Hogan, W. W. (1973). Point-to-set maps in mathematical programming. *SIAM Review*, 15:591–603.

[14] Krippahl, L. and Barahona, P. (1999). Applying constraint programming to protein structure determination. In *Principles and Practice of Constraint Programming*, pages 289–302. Springer Verlag, New York.

[15] Krippahl, L., Trosset, M., and Barahona, P. (2001). Combining constraint programming and multidimensional scaling to solve distance geometry problems. In *Proceedings of the Third International Workshop on Integration of AI and OR Techniques*. To appear.

[16] Kruskal, J. B. (1964a). Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. *Psychometrika*, 29:1–27.

[17] Kruskal, J. B. (1964b). Nonmetric multidimensional scaling: A numerical method. *Psychometrika*, 29:28–42.

[18] Lehoucq, R. B. (1995). Analysis and implementation of an implicitly restarted iteration. Technical Report 95-13, Department of Computational & Applied Mathematics, Rice University, 6100 Main Street, Houston, TX 77005-1892. Author's Ph.D. thesis.

[19] Lehoucq, R. B. and Sorensen, D. C. (1996). Deflation techniques for an implicitly restarted Arnoldi iteration. *SIAM Journal on Matrix Analysis and Applications*, 17:789–821.

[20] Lehoucq, R. B., Sorensen, D. C., and Yang, C. (1997). *ARPACK Users' Guide: Solution of Large Scale Eigenvalue Problems with Implicitly Restarted Arnoldi Methods*. Department of Computational & Applied Mathematics, Rice University, 6100 Main Street, Houston, TX 77005-1892. Available at `http://www.caam.rice.edu/`.

[21] Lewis, A. S. (1996). Derivatives of spectral functions. *Mathematics of Operations Research*, 21:576–588.

[22] Mardia, K. V. (1978). Some properties of classical multi-dimensional scaling. *Communications in Statistics—Theory and Methods*, A7:1233–1241.

[23] Mardia, K. V., Kent, J. T., and Bibby, J. M. (1979). *Multivariate Analysis*. Academic Press, Orlando.

[24] McCormick, G. P. and Tapia, R. A. (1972). The gradient projection method under mild differentiability conditions. *SIAM Journal on Control*, 10:93–98.

[25] Parks, T. A. (1985). Reducible nonlinear programming problems. Technical Report 85-8, Department of Mathematical Sciences, Rice University, Houston, TX. Author's Ph.D. dissertation.

[26] Ruhe, A. and Wedin, P. A. (1980). Algorithms for separable nonlinear least squares problems. *SIAM Review*, 22:318–337.

[27] Saito, T. (1978). The problem of the additive constant and eigenvalues in metric multidimensional scaling. *Psychometrika*, 43:193–201.

[28] Schoenberg, I. J. (1935). Remarks to Maurice Fréchet's article "Sur la définition axiomatique d'une classe d'espaces distanciés vectoriellement applicable sur l'espace de Hilbert". *Annals of Mathematics*, 38:724–732.

[29] Sibson, R. (1979). Studies in the robustness of multidimensional scaling: Perturbational analysis of classical scaling. *Journal of the Royal Statistical Society, Series B*, 41:217–229.

[30] Sorensen, D. C. (1992). Implicit application of polynomial filters in a k-step Arnoldi method. *SIAM Journal on Matrix Analysis and Applications*, 13:357–385.

[31] Takane, Y., Young, F. W., and de Leeuw, J. (1977). Nonmetric individual differences multidimensional scaling: An alternating least squares method with optimal scaling features. *Psychometrika*, 42:7–67.

[32] Tarazaga, P. and Trosset, M. W. (1993). An optimization problem on subsets of the symmetric postive semidefinite matrices. *Journal of Optimization Theory and Applications*, 79:513–524.

[33] Torgerson, W. S. (1952). Multidimensional scaling: I. Theory and method. *Psychometrika*, 17:401–419.

[34] Trefethen, L. N. and Bau, D. (1997). *Numerical Linear Algebra*. SIAM, Philadelphia, PA.

[35] Trosset, M. W. (1997). Numerical algorithms for multidimensional scaling. In Klar, R. and Opitz, P., editors, *Classification and Knowledge Organization*, pages 80–92, Berlin. Springer. Proceedings of the 20th annual conference of the Gesellschaft für Klassifikation e.V., held March 6–8, 1996, in Freiburg, Germany.

[36] Trosset, M. W. (1998a). Applications of multidimensional scaling to molecular conformation. *Computing Science and Statistics*, 29:148–152.

[37] Trosset, M. W. (1998b). A new formulation of the nonmetric STRAIN problem in multidimensional scaling. *Journal of Classification*, 15:15–35.

[38] Trosset, M. W. (2000). Distance matrix completion by numerical optimization. *Computational Optimization and Applications*, 17:11–22.

[39] Trosset, M. W., Baggerly, K. A., and Pearl, K. (1996). Another look at the additive constant problem in multidimensional scaling. Technical Report 96-7, Department of Statistics—MS 138, Rice University, Houston, TX 77005-1892.

[40] von Neumann, J. (1950). *Functional Operators, Volume II. The Geometry of Orthogonal Spaces*. Annals of Mathematical Studies, #22. Princeton University Press, Princeton, NJ. Reprinted from mimeographed lecture notes first distributed in 1933.

[41] Young, G. and Householder, A. S. (1938). Discussion of a set of points in terms of their mutual distances. *Psychometrika*, 3:19–22.

[42] Zangwill, W. I. (1969). *Nonlinear Programming: A Unified Approach*. Prentice-Hall, Englewood Cliffs, NJ.

[43] Zhu, C., Byrd, R. H., Lu, P., and Nocedal, J. (1994). L-BFGS-B: Fortran subroutines for large-scale bound constrained optimization. Technical Report NAM-11, Department of Electrical Engineering & Computer Science, Northwestern University, Evanston, IL 60208. Revised October 8, 1996.