HESSIAN OF THE RIEMANNIAN SQUARED DISTANCE FUNCTION ON CONNECTED LOCALLY SYMMETRIC SPACES WITH APPLICATIONS

Ricardo Ferreira * **João Xavier** *

* *ricardo@isr.ist.utl.pt jxavier@isr.ist.utl.pt*
Instituto de Sistemas e Robótica (ISR),
Instituto Superior Técnico
Av. Rovisco Pais, 1049-001 Lisbon, Portugal

Abstract: Many optimization problems formulated on Riemannian manifolds involve their intrinsic Riemannian squared distance function. A notorious and important example is the centroid computation of a given finite constellation of points. In such setups, the implementation of the fast intrinsic Newton optimization scheme requires the availability of the Hessian of this intrinsic function. Here, a method for obtaining the Hessian of this function is presented for connected locally-symmetric spaces on which certain operations, e.g. exponential, logarithm and curvature maps, are easily carried out. Particularly, naturally reductive homogeneous spaces provide the needed information, hence applications will be shown from this set. We illustrate the application of this theoretical result in two engineering scenarios: (i) centroid computation and (ii) maximum a posteriori (MAP) estimation. Our results confirm the quadratic convergence rate of the intrinsic Newton method derived herein.

Keywords: Riemannian manifolds, Hessian of intrinsic distance, Intrinsic Newton method, Centroid computation, MAP estimation

1. INTRODUCTION AND MOTIVATION

Due to its quadratic convergence rate near the solution, Newton's method has for a long time been the method of choice for optimization problems, especially when high precision is required. This is true in many fields, from engineering to numerical analysis where it is used extensively to obtain many digits of precision. Intrinsic quasi-Newton and Newton algorithms for optimization problems on smooth manifolds have been discussed (among others) in (Gabay, 1982) and in (Edelman *et al.*, 1998) and (Manton, 2002). Applications can be found in robotics (Belta and Kumar, 2002), signal processing (Manton, 2005), image processing (Helmke *et al.*, 2004), etc.

Implementation of the intrinsic Newton scheme relies on the availability of the intrinsic Hessian of the function to be optimized. In this paper, our interest lies on those optimization problems involving the intrinsic squared Riemannian distance. As a special case, this subsumes the class of optimization problems known as centroid computation.

Many applications of centroid computation exist in the literature. For example (Moakher, 2002) mentions centroid computation in $\mathbb{SO}(3)$ for the study of plate tectonics and sequence-dependent continuum modeling of DNA. In these, experimental observations are obtained with a significant amount of noise that needs to be smoothed. Positive definite symmetric matrices are used as covariance matrices in (Pennec *et al.*, 2004) for statistical characterization of deformations and encoding of principle diffusion directions in Diffusion

Tensor Imaging (DTI), expanding the range of applications to medicine. Computation of centers of mass also find applications for analyzing shapes in medical imaging, see (Fletcher *et al.*, 2004). It is also a mandatory step when considering the extension of the K-means algorithm to manifolds.

Most of these approaches rely on gradient methods for computing the intrinsic Riemannian centroid with a few exceptions: Hüper and Manton in (Hüper and Manton, 2005) developed a Newton method for the special orthogonal group and (Absil *et al.*, 2004) introduced a Newton method applicable to Grassmann manifolds which operates on an approximation of the intrinsic cost function, yielding the intrinsic centroid.

1.1 Contribution We present and discuss a method for computing the intrinsic Hessian of the Riemannian squared distance function for locally symmetric Riemannian manifolds. This builds on our previous work in (Ferreira *et al.*, 2006). In this paper we simplify the description of the intrinsic Hessian by providing a concise matrix expression in each tangent space. This entails an appreciable simplification with respect to (Ferreira *et al.*, 2006), e.g. the need for parallel translation of tangent vectors is dropped, thus relaxing the amount of required differential geometric functions initially imposed.

Adding to the already described cases of the embedded sphere $\mathbb{S}^n$, special orthogonal group $\mathbb{SO}(n)$ and symmetric positive definite matrices $Sym^+(n)$, the method will also be shown to work for the special Euclidean group $\mathbb{SE}(n)$ and the Grassmann manifold $\mathbb{G}(n,p)$. Notice that the real projective plane $\mathbb{P}^n$ is a particular case of the Grassmann manifold ($\mathbb{P}^n \cong \mathbb{G}(n+1,1)$). As an example of other applications, a simple example of MAP estimation is described and results are shown.

1.2 Paper Organization The paper is structured as follows:

- Section 2 provides a short review of Newton's method on Riemannian manifolds and also presents the main result of this article - the matrix version of the Hessian of the Riemannian squared distance function in a given tangent space. This enhances our previous result in (Ferreira *et al.*, 2006) and describes it in an implementation-friendly format.
- To construct the Hessian, several differential-geometric operations on the manifold are needed, e.g. computation of the curvature tensor. Section 3 mentions a method for obtaining the needed computational tools for a class of commonly used manifolds: naturally reductive homogeneous spaces (all the manifolds mentioned earlier fall under this category, here $\mathbb{SE}(n)$ is seen as $\mathbb{SO}(n) \times \mathbb{R}^n$). Further, when dealing with constant sectional curvature manifolds, we show how to decrease the computational complexity of the algorithm.
- Section 4 illustrates the application of the theory to the problem of intrinsic centroid computation by a Newton method. A pseudo-code implementation of the algorithm is given. Results for the special Euclidean group, the Grassmann manifold and real projective plane are presented. Another application, concerning MAP position estimation in the context of robot navigation is shown as well.
- Finally, some conclusions are drawn and directions for future work are delineated in section 5.

2. HESSIAN OF THE RIEMANNIAN SQUARED DISTANCE FUNCTION

2.1 Review of Newton's Method in Riemannian Manifolds Let $q_k \in M$ henceforth designate the kth iterate in an optimization method formulated on a Riemannian manifold M. Newton's method on a manifold is essentially the same as in $\mathbb{R}^n$ (see (Edelman *et al.*, 1998), (Manton, 2002) and (Hüper and Trumpf, 2004) for some generalizations). It generates a search direction $d_k \in T_{q_k}M$ as the solution of the linear system

$$H \cdot d_k = -\operatorname{grad} f(q_k)\,, \tag{1}$$

where H is the bilinear Hessian tensor of the smooth cost function $f : M \to \mathbb{R}$ evaluated at q_k and $\operatorname{grad} f(q_k) \in T_{q_k}M$ is its gradient. Some care is needed though, since the Hessian and the gradient are not as simple to find as in $\mathbb{R}^n$, but are in fact given as the solutions of $(df)_q X_q = \langle \operatorname{grad} f(q), X_q \rangle$ and $\operatorname{Hess} f(q)(X_q, Y_q) = \langle \nabla_{X_q} \operatorname{grad} f, Y_q \rangle$, where $q \in M$ and $X_q, Y_q \in T_qM$ are any tangent vectors. Here $(df)_q$ denotes the differential of the function f at the point q and ∇ denotes the Levi-Civita connection of the manifold. Here, $\langle \cdot, \cdot \rangle$ denotes the inner product on T_qM.

Once a Newton direction has been obtained, it should be checked if it's a descent direction (its inner product with the gradient vector should be negative). If so, the update equation $q_{k+1} = exp_{q_k}(\alpha_k d_k)$, can be used to obtain a better estimate. Here α_k is a step size, given for example by Armijo's rule, and exp denotes the Riemannian exponential map. If the inner product is negative, a safe negative gradient direction should be used.

Although the gradient is usually easy to compute, determination of the Hessian is more involved. The next section describes a method to calculate it on particular Riemannian manifolds (connected

and locally-symmetric where the curvature tensor and Riemannian logarithm maps are known).

2.2 Hessian in Matrix Form This section starts with a slightly upgraded version of the main result presented in (Ferreira *et al.*, 2006), stating:

Theorem 1. Consider M to be a connected locally-symmetric n-dimensional Riemannian manifold with curvature endomorphism R. Let $B_\epsilon(p)$ be a geodesic ball centered at $p \in M$ and $r_p : B_\epsilon(p) \to \mathbb{R}$ the function returning the intrinsic (geodesic) distance to p. Let $\gamma : [0, r] \to B_\epsilon(p)$ denote the unit speed geodesic connecting p to a point $q \in B_\epsilon(p)$, where $r = r_p(q)$, and let $\dot{\gamma}_q \equiv \dot{\gamma}(r)$ be its velocity vector at q. Define the function $k_p : B_\epsilon(p) \to \mathbb{R}$, $k_p(x) = \frac{1}{2} r_p(x)^2$ and consider any $X_q, Y_q \in T_qM$. Then

$$\mathrm{Hess}(k_p)_q(X_q, Y_q) = \left\langle {X_q}^{\parallel}, Y_q \right\rangle + \sum_{i=1}^{n} \mathrm{ctg}_{\lambda_i}(r) \left\langle {X_q}^{\perp}, E_{iq} \right\rangle \langle Y_q, E_{iq} \rangle \ .$$

where $\{E_{iq}\} \subset T_qM$ is an orthonormal basis which diagonalizes the linear operator $\mathcal{R} : T_qM \to T_qM$, $\mathcal{R}(X_q) = R(X_q, \dot{\gamma}_q)\dot{\gamma}_q$ with eigenvalues λ_i, this means $\mathcal{R}(E_{iq}) = \lambda_i E_{iq}$. Also,

$$\mathrm{ctg}_\lambda(t) = \begin{cases} \sqrt{\lambda}\, t / \tan(\sqrt{\lambda}\, t) & \lambda > 0 \\ 1 & \lambda = 0 \\ \sqrt{-\lambda}\, t / \tanh(\sqrt{-\lambda}\, t) & \lambda < 0 \end{cases} .$$

Here the $\parallel$ and $\perp$ signs denote parallel and orthogonal components of the vector with respect to the velocity vector of γ, i.e. $X_q = {X_q}^{\parallel} + {X_q}^{\perp}$, $\left\langle {X_q}^{\perp}, {X_q}^{\parallel} \right\rangle = 0$, and $\left\langle {X_q}^{\perp}, \dot{\gamma}_q \right\rangle = 0$.

The improvement relative to (Ferreira *et al.*, 2006), which might be unnoticed at first sight, is the point of the manifold M at which the self-adjoint operator $\mathcal{R}$ is diagonalized, i.e. at point q. Previously, the diagonalization took place at $p \in M$ and the result was then parallel-transported to q (where it is needed). Here, we see that parallel translation is no longer required, and better yet this allows for the formula to be written in matrix notation as described next. Note that the theorems are formulated intrinsically, and nowhere is it assumed an embedded manifold characterization. This means that the presented work is independent of the representation chosen for the manifold M, which can range from simple cartesian products of submanifolds of $\mathbb{R}$, quotient manifolds or any other abstract construction.

Theorem 2. Under the same assumptions as above, consider $\{F_{iq}\} \subset T_qM$ an orthonormal basis. If $X_q \in T_qM$ is a vector, let the notation $\hat{X}$ denote the column vector describing the decomposition of X_q with respect to the basis $\{F_{iq}\}$, i.e. $[\hat{X}]_i = \langle X_q, F_{iq} \rangle$, let R_k be the matrix with entries $[R_k]_{ij} = \left\langle F_{iq}, \mathcal{R}(F_{j_q}) \right\rangle$ and consider the eigenvalue decomposition $R_k = E\Lambda E^T$. Here λ_i will be used to describe the i'th diagonal element of Λ. Then the Hessian matrix (a representation for the bilinear Hessian tensor on the finite dimensional tangent space with respect to the fixed basis) is given by:

$$H_k = E\Sigma E^T \tag{2}$$

where Σ is diagonal with elements σ_i given by $\sigma_i = \mathrm{ctg}_{\lambda_i}(r)$. Hence $\mathrm{Hess}(k_p)_q(X_q, Y_q) = \hat{X}^T H_k \hat{Y}$.

This result follows from theorem 1 by decomposing each vector with respect to the considered basis and noting that, due to the symmetries of the curvature endomorphism, the operator has a null eigenvalue associated with the eigenvector $\dot{\gamma}_q$.

3. IMPLEMENTATION

3.1 Naturally Reductive Homogeneous Spaces The theory of naturally reductive homogeneous spaces, henceforth denoted by NRHS, provides a practical way of obtaining the needed curvature endomorphism R, Riemannian exponential and logarithm maps for those cases where the manifold M can be written as a coset space G/H of a group G, acting on M transitively and by isometries, with isotropy subgroup H. The situation is particularly favorable from the computational viewpoint if G can be taken as $\mathbb{SO}(n)$ or $\mathbb{GL}(n)$, given the easy formula (matrix exponential) for its one parameter subgroups. If certain additional properties are verified (e.g. existence of a Lie subspace as defined for example, in (O'Neil, 1983)), all of the required maps can be found from the corresponding maps in G. For a description of these spaces see for example (O'Neil, 1983). Many manifolds in engineering can be described by this construction, particularly all the manifolds mentioned are NRHS (Grassmann, sphere, special orthogonal group, special Euclidean group, positive definite matrices and the projective plane). Note though that the NRHS set is not a subset of connected locally symmetric manifolds, hence care should be taken to ensure that the particular manifold on which optimization is to be performed verifies these conditions. For example, the Stiefel manifold (the set of k dimensional orthogonal frames in $\mathbb{R}^n$) is an NRHS space but is not locally symmetric, except in the cases where $k = 1$ which results in the sphere. Note that when $k = n$ the Stiefel is not connected (it is actually $\mathbb{O}(n)$).

3.2 Special Considerations Spaces with constant sectional curvature deserve special mention. In these spaces, it is easily seen that the eigenvalues of the operator $\mathcal{R}$ are the constant value of the curvature, except for one, which is 0 and

is associated with eigenvector $\dot{\gamma}_q$. Hence, if λ is the value of the curvature, the matrices can be decomposed as follows:

$$E = \begin{bmatrix} \dot{\gamma}_q & \dot{\gamma}_q^{\perp} \end{bmatrix} \qquad \Lambda = \lambda \begin{bmatrix} 0 & 0 \\ 0 & I \end{bmatrix}$$

where $\dot{\gamma}_q^{\perp}$ is an orthonormal complement of $\dot{\gamma}_q$. Thus the Hessian matrix becomes:

$$\begin{aligned} H_k &= \dot{\gamma}_q \dot{\gamma}_q^T + \mathrm{ctg}_\lambda(r) \dot{\gamma}_q^{\perp} \dot{\gamma}_q^{\perp^T} \\ &= \dot{\gamma}_q \dot{\gamma}_q^T + \mathrm{ctg}_\lambda(r)(I - \dot{\gamma}_q \dot{\gamma}_q^T) \\ &= \mathrm{ctg}_\lambda(r) I + (1 - \mathrm{ctg}_\lambda(r)) \dot{\gamma}_q \dot{\gamma}_q^T . \end{aligned} \quad (3)$$

This removes the need for the numerical computation of the eigenvalue decomposition.

All manifolds with an NRHS structure (possibly others where isometries are known) allow for another, sometimes important, optimization. Suppose that there's a privileged point $o \in M$ where computations are 'cheaper' to carry out. Since there's a group G acting transitively on M by isometries, an element $g_o \in G$ can be found such that $o = g_o \cdot q$, where $q \in M$ is the current Newton estimate. Applying this isometry to the whole constellation yields $\{p_i' : p_i' = g_o \cdot p_i\}$, i.e., a new constellation 'centered' at o. The result is that all optimization steps can be carried out at o simply by pre-applying g_0 and after the Newton iteration applying g_o^{-1} to the new estimate. Besides simplifying the computation of the Riemannian curvature, logarithm and exponential maps, it also allows for a tangent basis $\{F_{i_o}\}$ to be fixed for all iterations.

4. RESULTS

4.1 Centroid Computation Let M be a connected locally-symmetric Riemannian manifold and $\mathcal{X} = \{p_1, \ldots, p_L\} \subset M$ a constellation of L points. Let $r : M \times M \to \mathbb{R}$ be the function that returns the intrinsic distance of any two points on the manifold and define a cost function $C_{\mathcal{X}} : M \to \mathbb{R}$ as

$$C_{\mathcal{X}}(q) = \frac{1}{2} \sum_{l=1}^{L} r(p_l, q)^2 = \sum_{l=1}^{P} k_{p_l}(q) , \quad (4)$$

where the functions $k_{p_l} : M \to \mathbb{R}$ consider the distance to each point individually and are defined as $k_{p_l}(q) = 1/2\ r(p_l, q)^2$. The Fréchet mean set of the constellation is defined as the set of solutions to the optimization problem $m_f(\mathcal{X}) = \mathrm{argmin}_{q \in M} C_{\mathcal{X}}(q)$. Each element of the set $m_f(\mathcal{X})$ will be called a centroid of $\mathcal{X}$. Note that depending on the manifold M a generic constellation might have more than one centroid (for example if the sphere is considered with a constellation consisting of two antipodal points, all the equator points are centroids). The set of points at which the function (4) attains a local minimum is called the Karcher mean set and is denoted as $m_k(\mathcal{X})$. The objective will be to find a centroid for the given constellation (which in the applications of interest should be unique), but the possibility of convergence to a local minimum is not dealt with. If the points on the constellation are close enough to each other, it is known that the global set $m_f(\mathcal{X})$ has a single element and so the centroid is unique as stated in (Manton, 2004) and (Karcher, 1977).

Using linearity of the gradient and the Hessian operators (meaning in particular that if $f, g : M \to \mathbb{R}$ then $\mathrm{Hess}(f+g) = \mathrm{Hess}\, f + \mathrm{Hess}\, g$ and $\mathrm{grad}(f+g) = \mathrm{grad}\, f + \mathrm{grad}\, g$), the cost function in equation (4) allows for the decomposition

$$\begin{aligned} \mathrm{grad}\, C_{\mathcal{X}}(q) &= \sum_{l=1}^{L} \mathrm{grad}\, k_{p_l}(q) = -\sum_{l=1}^{L} log_q(p_n) \\ \mathrm{Hess}\, C_{\mathcal{X}}(q) &= \sum_{l=1}^{L} \mathrm{Hess}\, k_{p_l}(q) , \end{aligned} \quad (5)$$

where the fact that the gradient of the squared Riemannian distance function is the symmetric of the Riemannian *log* map is used (as stated in (Lee, 1997) as a corollary to Gauss's lemma).

Here is the outline of the Newton method as applied to this problem:

Input: Constellation $\mathcal{X} = \{p_1, ..., p_L\} \in M$
Output: Karcher Mean $q \in m_k(\mathcal{X})$
Initialization:
Choose $q_0 \in M$, tolerance $\epsilon > 0$. Set $k \leftarrow 0$.
Loop:

- * Apply initial isometry f to constellation and taking the current estimate q_k to o.
- Compute intrinsic gradient $g_k = \mathrm{grad}\, C_{\mathcal{X}}(q) \in T_{q_k} M$ by equation 5.
- If $|g_k| \leq \epsilon$ set $q = q_k$ and return.
- Compute Hessian matrix $H = \sum_{l=1}^{L} H_l$, where each H_l is given by equation 2 or 3.
- Compute Newton direction d_k as the solution of the system $H d_k = -g_k$.
- If $\langle d_k, g_k \rangle \geq 0$ set $d_k = -g_k$.
- Apply Armijo rule (a popular line search algorithm, see for example (Bertsekas, 1999)) to obtain $\alpha_k \approx \mathrm{argmin}_{\alpha \geq 0}\, exp_{q_k}(\alpha d_k)$.
- Set $q_{k+1} = exp_{q_k}(\alpha_k d_k)$. Please note that due to finite precision limitations, after a few iterations the result should be enforced to lie on the manifold.
- * If initial isometry was applied, set $q_{k+1} \leftarrow f^{-1}(q_{k+1})$.
- Set $k \leftarrow k+1$ and re-run the loop.

Where the steps marked with an asterisk are optional computational optimizations.

4.2 The $\mathbb{SE}(3) \subset M(3,4)$ Manifold An example on the special Euclidean group $\mathbb{SE}(3)$ (seen as a Riemannian submanifold of $M(3,4)$) with a constellation of 5 points is shown in figure 1.

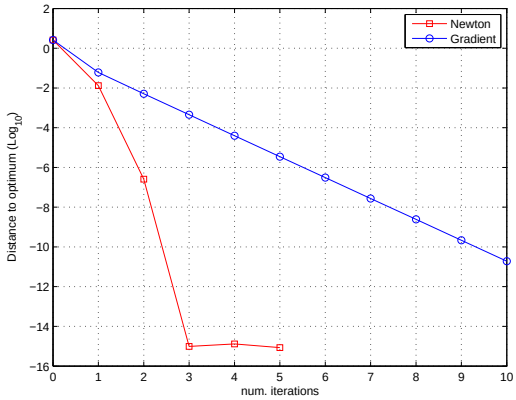

Fig. 1. Intrinsic centroid computation on $\mathbb{SE}(3)$ with a constellation of 5 points.

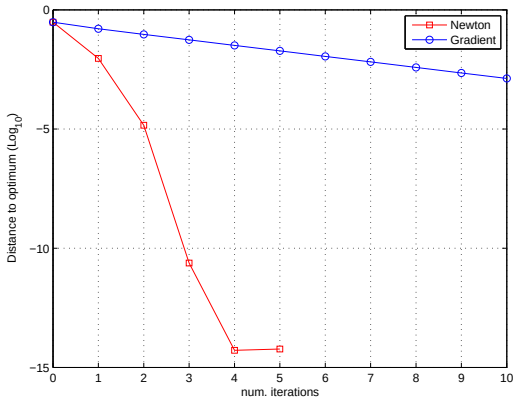

Fig. 2. Intrinsic centroid computation on the $\mathbb{G}(6,3)$ with a constellation of 5 points.

The results, shown in a logarithmic scale, clearly show the quadratic convergence rate of Newton's method and the almost perfectly linear convergence rate of the gradient method. Note the plateau at a precision of 10^{-15} resulting from numeric precision limitations.

4.3 The $\mathbb{G}(6,3) \cong \frac{\mathbb{SO}(6)}{\mathbb{SO}(3)\times\mathbb{SO}(3)}$ Manifold The results for $\mathbb{G}(6,3)$, the Grassmann manifold (here represented as a coset space of $\mathbb{SO}(n)$ as in (Edelman *et al.*, 1998)) of 3 dimensional subspaces in $\mathbb{R}^6$, are shown in figure 2. Again the quadratic convergence rate of Newton's method is clear. Note that this class of manifolds do not admit a canonic embedding in $\mathbb{R}^n$, thus showing that the presented results are not bound to embeddable manifolds.

4.4 The $\mathbb{P}^5 \cong \mathbb{G}(6,1)$ Manifold The method is also valid for the projection space, which is a special case of the Grassmann manifold. Results are presented in figure 3 for the manifold $\mathbb{P}^5$.

4.5 Robot Navigating in $\mathbb{R}^n$ As a simple application consider a robot moving freely in $\mathbb{R}^n$. Its state may be represented as a point $T \in \mathbb{SE}(n)$,

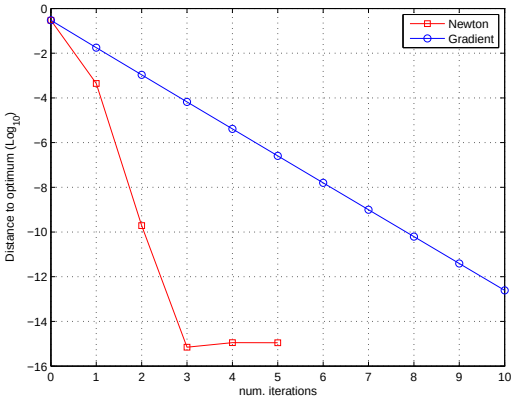

Fig. 3. $\mathbb{P}^5$ with a constellation of 25 points.

which acts on points in the world referential, taking them to the local referential. To keep the experiment simple, assume also that there are some known landmarks in the world $\{x_1, ..., x_k\} \in \mathbb{R}^n$ which the robot observes. Hence, in the local referential, the robot observes the points $T\,x_i$. Assuming that the robot is known to be in position T_0 with a certain uniform uncertainty, it is possible to build a prior knowledge probability density function as:

$$p(T) = K_1\, \mathrm{e}^{-\frac{1}{2}d(T,T_0)^2/\sigma^2}$$

where K_1 is a normalizing constant and σ^2 encodes the level of uncertainty. Notice that all directions are treated equally which is usually not the case, but to keep the example simple assume that this description is usefull. Assume also that the robot's sensor is not perfect and the observations obey the following Gaussian probability distribution:

$$p(y_i|T) = K_2\, \mathrm{e}^{-(y_i - T\,x_i)^T R^{-1} (y_i - T\,x_i)}$$

where, again K_2 is a normalizing constant and R is a matrix encoding the uncertainty of the sensor. With these assumptions and assuming the observations are independent, the MAP estimator of the robot's position is given by

$$T^* = \arg\max_{T\in\mathbb{SE}(n)} p(T|y_1, y_2, ..., y_k)$$

which is equivalent to

$$\begin{aligned} T^* &= \arg\max_{T\in\mathbb{SE}(n)} \sum_{i=1}^{k} \log(p(y_i|T)) + \log(p(T)) \\ &= \arg\max_{T\in\mathbb{SE}(n)} \sum_{i=1}^{k} -(y_i - Tx_i)^T R^{-1}(y_i - Tx_i) \\ &\qquad - \tfrac{1}{2}d(T,T_0)^2/\sigma^2 \end{aligned}$$

This is an optimization problem on $\mathbb{SE}(n)$. The gradient of each term is readily available and the Hessian of the first terms can be obtained using standard techniques. The result presented in this paper allows for the Hessian of the last term to be obtained as well, thus allowing for

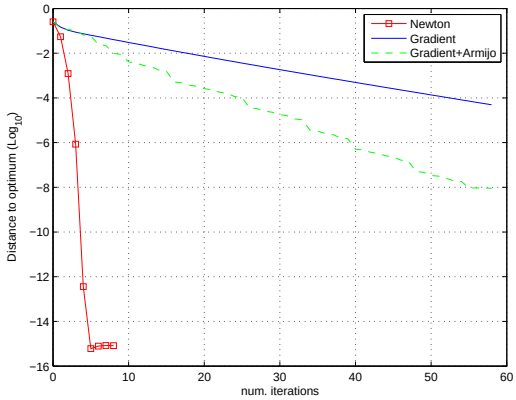

Fig. 4. Distance to optimum MAP estimate for 5 observations, $\sigma = 1$ and $R = I$.

a Newton algorithm to be implemented. Figure 4 shows the results when both the gradient and Newton methods are applied to 5 observations (using $\sigma = 1$ and $R = I$). The gradient method (both with and without the Armijo step selection rule) is clearly outperformed since the Newton method takes only 5 iterations to hit the objective at the given precision.

5. CONCLUSION

A simple formula for the Hessian matrix of the squared distance function on a connected locally-symmetric manifold was presented. The range of manifolds for which the method has been proven to work comprises the important cases of $\mathbb{SO}(n)$, $Sym^+(n)$, $\mathbb{S}^n$, $\mathbb{SE}(n)$, $\mathbb{G}(n,p)$ and $\mathbb{P}^n$. A simple example of MAP position estimation was presented, illustrating how to use the result on other problems besides centroid computation. Possible future work involves approximating the general Hessian expression (2), by a constant sectional curvature approximation. Although the quadratic convergence rate should be lost, a super-linear convergence rate might be possible without the need for an EVD decomposition. At this stage an implementation of a K-means clustering algorithm should be trivial as well.

REFERENCES

Absil, P., R. Mahony and R. Sepulchre (2004). Riemannian geometry of grassmann manifolds with a view on algorithmic computation. *Acta Appl. Math* **80(2)**, 199–220.

Belta, Calin and Vijay Kumar (2002). An svd-based projection method for interpolation on se(3). *IEEE transactions on Robotics and Automation* **18(3)**, 334–345.

Bertsekas, Dimitri P. (1999). *Nonlinear Programming: 2nd Edition*. Athena Scientific.

Edelman, Alan, Tomas A. Arias and Steven T. Smith (1998). The geometry of algorithms with orthogonality constraints. *SIAM J. Matrix Anal. Appl.* **20**(2), 303–353.

Ferreira, R., J. Xavier, J. Costeira and V. Barroso (2006). Newton method for Riemannian centroid computation in naturally reductive homogeneous spaces. *International Conference on Acoustics, Speech, and Signal Processing ICASSP'06*.

Fletcher, P., C., Lu S.Pizer and S. Joshi (2004). Principal geodesic analysis for the study of nonlinear statistics of shape. *IEEE Transactions on Medical Imaging* **23(8)**, 995–1005.

Gabay, D. (1982). Minimizing a differentiable function over a differential manifold. *J. Optim. Theory Appl.*

Helmke, Uwe, Knut Hüper, Pei Lee and John Moore (2004). Essential matrix estimation via Newton-type methods. *16th International Symposium on Mathematical Theory of Network and System (MTNS), Leuven*.

Hüper, K. and J. Manton (2005). The Karcher mean of points on the orthogonal group. *Preprint*.

Hüper, Knut and Jochen Trumpf (2004). Newton-like methods for numerical optimization on manifolds. *Asilomar Conference on Signals, Systems and Computers* pp. 136–139.

Karcher, H. (1977). Riemannian center of mass and mollifier smoothing. *Comm. Pure Appl. Math.* **30**, 509–541.

Lee, John M. (1997). *Riemannian Manifolds: An Introduction to Curvature*. Springer.

Manton, Jonathan H. (2002). Optimisation algorithms exploiting unitary constraints. *IEEE Transactions on Signal Processing* **50**(3), 635–650.

Manton, Jonathan H. (2004). A globally convergent numerical algorithm for computing the centre of mass on compact Lie groups. In: *Eighth International Conference on Control, Automation, Robotics and Vision*. Kunming, China.

Manton, Jonathan H. (2005). A centroid (Karcher mean) approach to the joint approximate diagonalisation problem: The real symmetric case. *Digital Signal Processing*.

Moakher, Maher (2002). Means and averaging in the group of rotations. *SIAM J. Matrix Anal. Appl.* **24**(1), 1–16.

O'Neil, Barret (1983). *Semi-Riemannian Geometry*. Academic Press.

Pennec, Xavier, Pierre Fillard and Nicholas Ayache (2004). A Riemannian framework for tensor computing. Research Report 5255. INRIA. To appear to the Int. Journal of Computer Vision.