Unified 3-D Structure and Projection Orientation Refinement Using Quasi-Newton Algorithm

Chao Yang[a], Esmond G. Ng[a], Pawel A. Penczek[b]

[a]Computational Research Division
Lawrence Berkeley National Laboratory
Berkeley, CA 94720

[b]The University of Texas – Houston Medical Center
Department of Biochemistry and Molecular Biology
6431 Fannin, MSB 6.218, Houston, TX 77030

Corresponding author:
Pawel A. Penczek
phone: (713) 500-5416
fax: (713) 500-0652
E-mail: Pawel.A.Penczek@uth.tmc.edu

Running title: Unified 3-D Structure and Projection Orientation Refinement

List of key words: 3-D structure determination, electron microscopy, Quasi-Newton algorithm.

List of abbreviations: EM - electron microscopy, 2-D - two-dimensional, 3-D - three-dimensional, 3-D EM - three-dimensional electron microscopy, BFGS –Broyden, Fletcher, Goldfarb, and Shannon update, LBFGS – limited-memory BFGS, SNR – Signal-to-Noise Ratio.

ABSTRACT

We describe an algorithm for simultaneous refinement of a three-dimensional density map and of the orientation parameters of two-dimensional projections that are used to reconstruct this map. The application is in electron microscopy, where the three-dimensional structure of a protein has to be determined from a set of two-dimensional projections collected at random but initially unknown angles. The design of the algorithm is based on the assumption that initial low resolution approximation of the density map and reasonable guesses for orientation parameters are available. Thus, the algorithm is applicable in final stages of the structure refinement, when the quality of the results is of main concern. We define the objective function to be minimized in real space and solve the resulting nonlinear optimization problem using a Quasi-Newton algorithm. We calculate analytical derivatives with respect to density distribution and the finite difference approximations of derivatives with respect to orientation parameters. We demonstrate that calculation of derivatives is robust with respect to noise in the data. This is due to the fact that noise is annihilated by the back-projection operations. Our algorithm is distinguished from other orientation refinement methods (i) by the simultaneous update of the density map and orientation parameters resulting in a highly efficient computational scheme and (ii) by the high quality of the results produced by a direct minimization of the discrepancy between the 2-D data and the projected views of the reconstructed 3-D structure. We demonstrate the speed and accuracy of our method by using simulated data.

1. INTRODUCTION

In single particle analysis the data is available in the form of two-dimensional (2-D) electron microscopy (EM) projections of a three-dimensional (3-D) electron density map of a biological macromolecule. The goal of the analysis is to recover the 3-D structure, but the directions of projections are unknown. The initial guess for the projection directions can be established either experimentally using the Random Conical Tilt technique (Radermacher *et al.*, 1987) or computationally (Goncharov, 1986; Goncharov *et al.*, 1987; Penczek *et al.*, 1996; van Heel, 1987). In either case, the errors in projection directions will be large and the resulting initial structure will have low resolution, so the subsequent refinement of the orientation parameters assigned to projections is necessary.

The currently used refinement procedures can be roughly divided into two groups: (i) those that are based on comparison of the 2-D projection data with the systematically generated projections of the current guess of the structure and (ii) those that seek to correct orientation parameters by minimizing an overall alignment error among projections. The first category is prominently represented by the *projection matching* technique (Penczek *et al.*, 1994). In this approach, the step of orientation correction is separated from the step of calculating the 3-D reconstruction of the new density map. Since the method is implemented in real space, i.e., both the projection operations and the 3-D reconstruction are carried out in the object space, the method is reasonably efficient and the interpolation errors are minimized. Nevertheless, because of the separation of the

two key steps of the procedure the reassignment of the projection direction does not necessarily guarantee the improvement of the density map. In addition, any artifacts introduced into the density map by the reconstruction algorithm will propagate into subsequent steps of the procedure. The methods that fall into the second category are usually implemented by using transformations to map the projection data into spaces in which the data can be conveniently manipulated. Examples of such transformations are spherical harmonics transformation (Navaza, 2003; Provencher and Vogel, 1988; Yin *et al.*, 2003), Fourier transformation (Grigorieff, 1998), and Radon transformation (Radermacher, 1994; Radermacher *et al.*, 2001). The main advantage of these approaches is that the problem of separating the orientation search from the 3-D reconstruction is eliminated, as the resulting structure in real space can be calculated only once, after the convergence of the orientation determination algorithm is achieved. Unfortunately, none of the methods in this group can be used to perform exhaustive searches in an efficient way. Consequently, these methods are more appropriate for the final stages of the structure refinement. On a more fundamental level, a major drawback associated with working in transformed spaces is that the data (2-D projections) is represented in polar coordinates, while the resulting 3-D structure must be obtained in (uniform) Cartesian coordinates. Transformation from a nonuniform polar grid to the uniform grid constitutes a difficult inverse problem that is sensitive to the presence of noise in the data and to interpolation errors (Penczek *et al.*, 2004). Therefore, even if an optimum solution for the orientation problem is found in the transformed space, it is not immediately apparent that the solution corresponds to an optimum 3-D structure in Cartesian space, as the two are separated by a potentially ill-posed inverse transformation. So far, little attention has been devoted to the error analysis in the orientation searches in transformed spaces.

In order to overcome some of the shortcomings of the existing structure determination methods we propose a new approach based on the direct and simultaneous optimization of both the density map and orientation parameters of the projection data. In our method we seek solution to the problem that is formulated in the measurement space, i.e., the real space. In this way, we hope to minimize the adverse effect of the interpolation errors on the refinement procedure. In addition, by integrating the inverse to the projection transformation directly into the framework of the optimization problem, we are able to fully explore the interdependence between the orientation of the projection data and the 3-D structure to be reconstructed.

Formally, we state the estimation of the 3-D electron density map (denoted by $f \in R^{n^3}$) of a biological molecule from a large number of 2-D EM projection images, $b_i \in R^{n^2}$, $i = 1,2,...,m,$ of isolated (single) particles with random and unknown orientations as a nonlinear optimization problem:

$$\min_{\phi_i,\theta_i,\psi_i,s_{x_i},s_{y_i},f} \rho(\phi_i,\theta_i,\psi_i,s_{x_i},s_{y_i},f) = \frac{1}{2}\sum_{i=1}^{m}\left\| P(\phi_i,\theta_i,\psi_i)\, f\left(s_{x_i},s_{y_i}\right) - b_i \right\|^2, \qquad (1)$$

where $P(\phi_i, \theta_i, \psi_i)$ is a line integral operator that projects f onto a 2-D plane after f is shifted by (s_{x_i}, s_{y_i}) and rotated by a set of unknown Euler angles $(\phi_i, \theta_i, \psi_i)$. The factor of ½ is included merely for convenience.

The objective function in (1) is clearly nonlinear due to the coupling between the orientation parameters $(\phi_i, \theta_i, \psi_i, s_{x_i}, s_{y_i})$, $i = 1, 2, \ldots, m$, and the 3-D density f. The total number of unknown parameters to be estimated is $n^3 + 5m$. Note that in single particle analysis the number of projection data *m* is far greater than the linear size of the data in pixels, i.e., $m \gg n$.

We are interested in numerical methods for finding an optimal solution to (1) with the assumption that reasonable approximations to f and $\left(\phi_i, \theta_i, \psi_i, s_{x_i}, s_{y_i}\right)$, $i = 1, 2, \ldots, m$ are available. That is, we are concerned with a local optimization scheme instead of trying to tackle (1) as a global optimization problem. Methods for obtaining an initial low resolution approximation to f can be found in (Goncharov, 1986; Goncharov *et al.*, 1987; Penczek *et al.*, 1996; van Heel, 1987).

A generalized coordinate descent algorithm called *projection matching* is presented in (Penczek *et al.*, 1994) to seek a minimizer of (1) in two alternating search directions. Starting from a given low resolution density approximation $f^{(0)}$, the algorithm performs an exhaustive search for the optimal Euler angles $(\phi_i, \theta_i, \psi_i)$ and a restricted search for the optimal translations (s_{x_i}, s_{y_i}) associated with each EM projection image b_i. These searches are carried out by comparing b_i with a set of reference projections p_j, (j = 1, 2,…, m_r) produced by computationally re-projecting $f^{(0)}$ in directions specified by a set of prescribed and quasi-uniformly distributed Euler angles $\left(\hat{\phi}_i, \hat{\theta}_i, \hat{\psi}_i\right)$, $i = 1, 2, \ldots, m_r$. The set of angles and shifts that yields the minimum value of $\left\| b_i - p_j \right\|$ is assigned to b_i. Once each EM projection image has been assigned a set of reference Euler angles $\left(\hat{\phi}_i, \hat{\theta}_i, \hat{\psi}_i\right)$ and shifts $\left(\hat{s}_{x_i}, \hat{s}_{y_i}\right)$, a new density map $f^{(1)}$ is computed by solving a linear least squares problem

$$\min_f \frac{1}{2} \sum_{i=1}^{m} \left\| P(\hat{\phi}_i, \hat{\theta}_i, \hat{\psi}_i) f - b_i \right\|^2 , \qquad (2)$$

preferably using a version of the iterative algebraic reconstruction technique, such as SIRT, which yields a high quality estimate of the density map (Penczek *et al.*, 1992; Penczek *et al.*, 2004). Subsequently, the optimal solution to (2) is used to begin the next cycle of the iterative process until a stationary point of (1) is identified.

The experimental results presented in (Penczek *et al.*, 1994)) demonstrated that projection matching is quite effective for the reconstruction of the ribosome complex eventually leading to determination of the structure of 70S *E. coli* ribosome at 11.5 Å

resolution (Gabashvili *et al.*, 2000). Projection matching also proved to be equally effective for determination of structures of a variety of macromolecular assemblies, both asymmetric (Beckmann *et al.*, 1997; Craighead *et al.*, 2002) and symmetric (Boisset *et al.*, 1995). Unfortunately, little is known about the theoretical convergence properties of the method. Based on our experience, we can state that the overall performance of the projection matching method is mainly limited by the separation of the search for the orientation parameters from the 3-D reconstruction that yields a new density map. Although this separation results in a computational scheme that is reasonably efficient, the convergence of the method is sometimes unpredictable. It is important to notice that during the first phase of each iteration, the correction of the orientation parameters associated with one particular projection image is carried out independently from those associated with the remaining projections. Clearly, this approach does not necessarily guarantee decrease of the target function (1). In fact, it can even increase its value, especially if the second phase of the iteration is not carried out accurately. A more appropriate strategy is perhaps to update the density map by solving (2) after the assignment of orientation parameters to each projection is completed, but such an approach would be prohibitively time consuming. In addition, the result is bound to depend on the order in which projections are processed.

In this paper, we will demonstrate that the search for the optimal density and orientation parameters can be carried out simultaneously by applying a Quasi-Newton algorithm (Norcedal and Wright, 1999) to (1) directly. The simultaneous search offers the benefits of potentially more rapid convergence and lower computational cost. Because it puts the correction of the 3-D structure and the correction of the orientation parameters on an equal footing, the problem of error propagation, which tends to occur in the projection matching algorithm, is mitigated.

2. METHODS

In this section we present the optimization method we use to solve the unconstrained nonlinear problem of the simultaneous 3-D structure and projection orientation refinement given by (1). We selected a Quasi-Newton scheme, in which the approximation to the inverse of the Hessian is constructed incrementally by making use of the gradient information gathered at previous iterations. In the current presentation of the algorithm we assume there are no translational errors; however, this does not restrict the generality of the approach as additional unknown parameters can be introduced naturally within the framework of the selected optimization method. We conclude the section by providing an analysis of the computational complexity of the algorithm and demonstrating that it compares favorably with that of the projection matching method.

In the discussion that follows, we use $\rho(x)$ to represent the objective function to be optimized in (1), where $x^T = \left(f \ \phi_1 \cdots \phi_m \ \theta_1 \cdots \theta_m \ \psi_1 \cdots \psi_m \right)$ is a vector representation of the unknown parameters contained in (1). Note the absence of translational errors. It is convenient to express $\rho(x)$ as

$$\rho(x) = \frac{1}{2}\|r(x)\|^2, \tag{3}$$

where

$$r(x) = \begin{pmatrix} P(\phi_1, \theta_1, \psi_1) f - b_1 \\ P(\phi_2, \theta_2, \psi_2) f - b_2 \\ \vdots \\ P(\phi_m, \theta_m, \psi_m) f - b_m \end{pmatrix} \tag{4}$$

is the residual vector that measures the discrepancy between the data and the re-projected 3-D structure. Note that each sub-component of this residual vector, $P(\phi_i, \theta_i, \psi_i) f - b_i$, provides a measure of consistency between a particle image and a single 2-D projection of the 3-D model along one particular direction. The norm of this sub-component, which will be computed as part of the objective function evaluation in a Quasi-Newton algorithm, plays the same role as a cross-correlation coefficient that is sometimes used in a projection matching algorithm to discard particle images with poor quality. However, removing some of the 2-D images during the course of the refinement essentially amounts to redefining the objective function defined in (1) and would have to be carried out carefully.

The standard numerical procedure for solving an unconstrained nonlinear optimization problem (1) can be described as follows. Given a starting guess $x^{(0)}$ of the optimal solution x, one seeks a search direction *s* such that $\rho(x^{(0)} + \alpha s) < \rho(x^{(0)})$, for some choice of α. Commonly used search directions are the steepest descent direction (negative of the gradient), the Newton direction, the Quasi-Newton direction, and the Gauss-Newton direction. Once a search direction is chosen, one can use either a line search or a trust region strategy (Norcedal, 1991) to select an appropriate step length α. (The use of a trust region also refines the search direction.) If the objective function remains large at the new iterate $x^{(1)} = x^{(0)} + \alpha \cdot s$, a new search direction and step length are computed. These steps are repeated until there is no further reduction in $\rho(x^{(k)})$ for some *k*.

The computation of the search direction usually involves evaluating the derivatives of $\rho(x)$ with respect to each parameter contained in *x*. If the steepest descent direction is chosen as the search direction, one only needs to compute the first derivative of $\rho(x)$ with respect to all elements of *x*. The second derivatives or their approximations are required for Newton, Quasi-Newton, and Gauss-Newton directions.

It is generally difficult to compute the analytical derivatives of $\rho(x)$ with respect to the orientation parameters. However, these derivatives can be approximated through the use of the finite difference technique. For example, one may compute the partial derivative of $\rho(x)$ with respect to ϕ_i as follows:

$$\frac{\partial \rho}{\partial \phi_i} \approx g_i^T r_i , \tag{5}$$

where

$$g_i = \frac{P(\phi_i + \Delta\phi, \theta_i, \psi_i) - P(\phi_i, \theta_i, \psi_i)}{\Delta\phi} f ,$$
$$r_i = P(\phi_i, \theta_i, \psi_i) f - b_i . \tag{6}$$

It is easy to verify that the gradient of $\rho(x)$ can be expressed by

$$\nabla\rho(x) = J^T r, \tag{7}$$

where J is the Jacobian matrix that has the form of

$$J = \begin{pmatrix} P_1 & g_1^{\phi} & & & g_1^{\theta} & & & g_1^{\psi} & & \\ P_2 & & g_2^{\phi} & & & g_2^{\theta} & & & g_2^{\psi} & \\ \vdots & & & \ddots & & & \ddots & & & \ddots \\ P_m & & & & g_m^{\phi} & & & g_m^{\theta} & & g_m^{\psi} \end{pmatrix}, \tag{8}$$

and

$$P_i = P(\phi_i, \theta_i, \psi_i), \tag{9}$$
$$g_i^{\phi} = \frac{P(\phi_i + \Delta\phi, \theta_i, \psi_i) - P(\phi_i, \theta_i, \psi_i)}{\Delta\phi} f ,$$
$$g_i^{\theta} = \frac{P(\phi_i, \theta_i + \Delta\theta, \psi_i) - P(\phi_i, \theta_i, \psi_i)}{\Delta\theta} f ,$$
$$g_i^{\phi} = \frac{P(\phi_i, \theta_i, \psi_i + \Delta\psi) - P(\phi_i, \theta_i, \psi_i)}{\Delta\psi} f .$$

Although the dimension of the Jacobian matrix, which is $n^2 m \times (n^3 + 3m)$, is large, the matrix itself is quite sparse. Furthermore, the gradient calculation does not require J to be stored explicitly; therefore, the computation of $J^T r$ can be accomplished iteratively using a sequence of projection and back-projection calculations. As explained earlier, we restrict our analysis to the case where there are no translational errors. If the shifts were included, the Jacobian matrix (8) would contain two more block diagonal submatrices and its dimensions would increase accordingly. However, the sparsity structure of the matrix would remain essentially the same, and in general terms the analysis presented below would hold.

The negative of the gradient gives the steepest descent direction which one may use directly to search for a local minimum of (1). However, an optimization algorithm based purely on the steepest descent search direction may have a very slow convergence rate. To accelerate the convergence of a gradient-based algorithm, one often needs

additional information about the curvature of the objective function. Because it is generally not practical to compute the Hessian of $\rho(x)$ directly, we resort to a Quasi-Newton scheme in which an approximation to the inverse of the Hessian is constructed incrementally by making use of the gradient information gathered at previous iterations. In particular, one obtains a search direction by solving

$$B_k s_k = -\nabla\rho(x_k), \tag{10}$$

where B_k is an approximate Hessian of $\rho(x)$. We follow the limited-memory BFGS (LBFGS) algorithm (Norcedal, 1980) to update the approximate Hessian (or its inverse) using a low rank modification. The term "limited-memory" refers to the fact that the LBFGS algorithm requires saving only a fixed number of gradient vectors computed in previous iterations. These gradient vectors are used to provide a compact representation of an approximate Hessian. The approximate Hessian matrix B_k (or its inverse) is never stored explicitly.

The predominant computational cost of the LBFGS algorithm is the function and gradient evaluations performed during each iteration step. The evaluation of the objective function (1) involves *m* projection calculations. If linear interpolations are used in these calculations, the function evaluation consumes $O(mn^3)$ floating point operations (flops). To calculate the gradient, six (assuming we are considering shifts) or four (if only the Eulerian angles are refined) projections and one back-projection are required for each 2-D particle image. These operations also have a computational complexity of $O(mn^3)$. Thus, the overall complexity of the function and gradient calculations is $O(mn^3)$ with a constant factor that is less than 10. This complexity analysis compares favorably with that associated with the projection matching algorithm. In projection matching, the search for the optimal in-plane rotation is typically carried out by cross-correlating a particle image with a reference projection, which consumes $O(n^2 \log n)$ flops with a multiplicative factor that depends on the range of translations considered (Joyeux and Penczek, 2002). In addition, the use of SIRT in solving (2) consumes $O(kmn^3)$ flops, where *k* is the number of SIRT iterations required to reach the minimum of (2). Thus, the overall computational complexity of the projection matching algorithm is at least $O(kmn^3 + mn_r n^2 \log n)$, where n_r is the number of reference images generated to carry out exhaustive orientation search. Typically, n_r is much larger than *n*, hence the cost of projection matching tends to be significantly higher than that of LBFGS on a per iteration basis. When good initial guesses of the orientation parameters are available, one can potentially perform a localized search. This can reduce the number of cross-correlations and thus improves the efficiency of projection matching.

3. RESULTS

In this section, we describe experimental results obtained from applying the simultaneous structure and orientation optimization technique developed in Section 2 to simulated data.

3.1 Preparation of the test data

We use the 3-D density map of the multisubunit transcription factor IID (TFIID) complex published in (Andel *et al.*, 1999) to generate our test data. The 3-D map is placed in a volume 64^3 voxels, with the voxel size 7Å. The resolution of the structure is 35Å. Three different 3-D views of the structure are shown in Figure 1. This 3-D map serves as part of the ideal solution to the optimization problem (1) that we try to solve.

We project the ideal 3-D TFIID structure *f* computationally in 34,429 quasi-uniformly distributed directions using a 0.77° angular step (Penczek *et al.*, 1994). Among these 2-D projection images, we randomly select 799 images as the actual 2-D data set to be used in the subsequent computation. That is, these 799 images b_i ($i = 1,2,...,799$) are used to recover the 3-D structure *f* and the random projection (Euler) angles $(\phi_i, \theta_i, \psi_i)$, ($i$=1,2…,799) simultaneously. Trilinear interpolation is used for the projection calculation. The purpose of generating a set of 2-D projection data in such a fashion is to mimic the random distribution of viewing angles associated with real experimental data.

The initial guess of the 3-D structure used to start the limited-memory BFGS optimization procedure is a low resolution 3-D structure obtained from a random conical tilt reconstruction using images pairs collected from tilted (32°) and untilted (0°) samples. During the original work on TFIID structure determination, this random conical tilt structure was used to initiate the projection matching procedure that lead to the eventual determination of the complex (Andel *et al.*, 1999). Three different 3-D views of this initial structure are shown in Figure 2.

Because the unknown parameters contained in our problem formulation include both the density of TFIID at each voxel and the Euler angles associated with each projection image, we need to provide initial guesses for the Euler angles also. These initial guesses for $(\phi_i, \theta_i, \psi_i)$ (i=1,2,…,799) are generated by perturbing the "exact" angles (that are used to generate the projection data) by $\Delta\phi_i$, $\Delta\theta_i$, and $\Delta\psi_i$, respectively, where $\Delta\phi_i$, $\Delta\theta_i$, and $\Delta\psi_i$ are from a Gaussian distribution $N\left(0, 30^\circ\right)$. The distribution of initial guesses for (ϕ_i, θ_i) (which defines the *i*-th projection direction) is shown in Figure 3 along with the distribution of the exact projection directions.

The low resolution 3-D structure and the perturbed orientation parameters form the starting point $x^{(0)}$ required for the LBFGS optimization procedure.

3.2 Convergence history

We monitor the convergence of the LBFGS algorithm by evaluating both the objective function $\rho(x)$ and the relative error in the 3-D structure after each iteration step. If the 3-D structure constructed at the *j*-th iteration is denoted by $f^{(j)}$, then the relative error of the structure is defined by

$$\delta_j = \frac{\left\| f - f^{(j)} \right\|}{\left\| f \right\|}. \qquad (11)$$

In Figure 4 we show that the objective function of the optimization problem defined in (1) decreases monotonically. After 100 iterations, the objective function is reduced by nearly two orders of magnitude. In Figure 5, we show that the relative error in *f* decreases from 0.80 (80% error in norm) to roughly 0.11 (11% error in norm) after 100 iterations. Although the reduction in the relative error is not strictly monotonic, the progress towards convergence is steady.

3.3 Quality of the Reconstruction

In Figure 6 we show the comparison of the reconstructed 3-D structure of TFIID with the original structure that is used to generate the projection data. The isosurface rendering of the reconstructed 3-D structure appears nearly indistinguishable from that of the original TFIID structure.

We also plotted the distributions of the projection directions recovered from the LBFGS calculation. In Figure 7 we show that most of the projection directions (defined by the angles ϕ_i and θ_i) match with the original directions along which the 2-D data is generated. To assess the resolution of the reconstructed 3-D volume, we computed the Fourier shell correlation (FSC) (Saxton and Baumeister, 1982) between $f^{(100)}$ and f. In Figure ~~8~~9 we show that FSC curve drops below the 0.5 cutoff at spatial frequency ~1/39 1/Å.

3.4 Comparison with Projection Matching

The LBFGS algorithm used to simultaneously refine the 3-D structure and the orientation parameters associated with each projection image is a local optimization scheme. The success of this method depends on having a good starting guess for the 3-D structure and orientations of projections. When such a good initial guess is available, the method can be very efficient in finding the optimal solution to (1). The most time-consuming part of the calculation is the gradient evaluation performed at each step. As we illustrated above, the gradient calculation is significantly cheaper than the exhaustive search one typically performs in the projection-matching algorithm.

In Figure 8 we show the cost comparison (in terms of wall clock time) between the simultaneous structure and orientation refinement using LBFGS and two versions of the projection matching algorithm. In the first version of the projection matching method (marked by '+'), an exhaustive search for all possible projection directions with an angular step of 5^o was performed. In this case, the number of reference projection images used in the matching algorithm was 799. In the second version (marked by circles), a localized search within the cone neighborhood of 30^o was carried out. This restriction of the angular search resulted in a smaller number of comparisons and consequently in a decrease in wall clock time. The SPIDER implementation of SIRT algorithm (command 'BP RP') (Penczek *et al.*, 1992) was used to perform the 3-D reconstruction with the

number of iterations set to 100. Both LBFGS and projection matching were implemented within a framework of SPIDER system (Frank *et al.*, 1996) using MPI parallelization (Pacheco, 1996). The calculations were carried out on an IBM SP at the National Energy Research Scientific Computing Center, which comprises of 375 Mhz Power 3 processors. Each Power3 processor is equipped with a 2 MB cache, and it has a peak performance of 1.5Gflops/second. We used 16 processors in our experiments.

We ran 200 iterations of LBFGS iterations. Each iteration took approximately 2 wall clock seconds on 16 processors. At the end of the 200-th iteration, the relative error became less than 0.09. The total wall clock time consumed on 16 processors was slightly over 400 seconds. In contrast, the relative error was reduced significantly during the first few projection matching iterations as the matching algorithm identified the approximate orientation for each projection image. However, the rate of convergence then slowed down considerably during subsequent iterations. When an exhaustive search was performed, each iteration of the projection matching took roughly 85 wall clock seconds on 16 processors. The relative error decreased to 0.12 at the 10-th iteration. A total of over 800 wall clock seconds was consumed. The algorithm appeared to stall at this point because it could no longer resolve the projection directions that did not lie on the quasi-uniform search grid (with a 5° angular step) associated with the reference projections. When a localized search was performed in the projection matching algorithm, each iteration took approximately 54 wall clock seconds. The relative error decreased to 0.14 at the end of 5-th iteration and the algorithm appeared to stall at that point and the relative error even began to increase slightly beyond that point indicating the amplification of the noise introduced by numerical round-offs. More than 500 wall clock seconds were consumed at the end of the 10-th iteration. The reason that a localized search converged to a suboptimal solution was that the initial guesses for some of the Euler angles were not located within the 30^{o} neighborhood of the true Euler angles. Hence the localized search missed these angles leading to a poorer solution.

In Figure ~~10~~9 we show the comparison of the FSC between f and $f_q^{(100)}$ with the FSC between f and $f_p^{(11)}$, where f is the ideal 3-D density function, $f_q^{(100)}$ is the 3-D density recovered at the end of the 100-*th* LBFGS iteration, and $f_p^{(11)}$ is the 3-D density produced at the end of the 11-*th* iteration of the projection matching algorithm. As we already pointed out earlier, the FSC curve associated with the LBFGS refinement drops below the 0.5 cutoff at the spatial frequency ~1/39 1/Å, which is close to the highest resolution (35 Å) one can achieve for this particular data set. Furthermore, the FSC values are close to one (the optimal correlation value) at low frequencies until the resolution limit is nearly reached. In terms of the FSC, the projection matching algorithm produces a solution with a comparable resolution. However, it can be seen from Figure 9 that the FSC curve associated with the projection matching reconstruction is not as close to one in the intermediate frequency range as that associated to the LBFGS reconstruction. This is likely to be even more pronounced if the 3-D model to be reconstructed were to contain prominent high frequency features.

3.5 The Effect of Noise

In electron microscopy, the 2-D projection images are typically noisy. In this section, we illustrate the effect of noise on the convergence of the LBFGS algorithm. We generated the noise-corrupted images as follows. For each projection image we used in the previous experiments, we added zero-mean Gaussian noise scaled such that the resulting Signal-to-Noise Ratio (SNR) in the 2-D projection image was one.

The introduction of noise in the data slowed the convergence of LBFGS slightly (Figure 11). Although the objective function decreases monotonically, which indicates that the gradient vector is not severally corrupted by noise, the relative error in the reconstructed 3-D density appears to reach the minimum around the ~~90-t~~80-th iterations. After the ~~90-t~~80-th iteration, the relative error starts to increase, indicating the amplification of noise in the subsequent refinement iterations.

It is not difficult to see why the presence of noise in the projection data does not have a severe effect on the gradient calculation. We partition the gradient vector $\nabla\rho(x) = J^T r$ as follows

$$\nabla\rho(x) = \begin{pmatrix} h_f \\ h_g \end{pmatrix}, \tag{12}$$

where $h_f \in R^{n^3}$ and $h_g \in R^{3m}$. It is easy to show that

$$h_f = \sum_{j=1}^{m} P_j^T r_j = \sum_{j=1}^{m} P_j^T \left(b_j - P_j f \right) \tag{13}$$

and

$$h_g^T = \left(\gamma_1^\phi \cdots \gamma_m^\phi \; \gamma_1^\theta \cdots \gamma_m^\theta \; \gamma_1^\psi \cdots \gamma_m^\psi \right), \tag{14}$$

where

$$\begin{aligned} \gamma_j^\phi &= r_j^T g_j^\phi, \\ \gamma_j^\theta &= r_j^T g_j^\theta, \\ \gamma_j^\psi &= r_j^T g_j^\psi, \end{aligned} \tag{15}$$

and g_j^ϕ, g_j^θ, and g_j^ψ are given by (9).

From equation (13) it clearly follows that the h_f component of the gradient vector is simply a sum of the back-projected residual vectors. It is important to note that the oscillatory noise components present in the 2-D projection data b_j and those introduced in the intermediate 3-D structure f can often be well represented by linear combinations of the left and right singular vectors associated with the zero or small singular values of the projection operator P_j, respectively (Hansen, 1997). Because the

noise present in b_j tends to lie in the null space of P_j^T, and because the noise introduced in f tends to lie in the null space of P_j, a significant portion of these undesirable components is likely to be annihilated or attenuated in the projection and back projection calculations in (13). Furthermore, when the noise components in 2-D projection images are uncorrelated, the sum of the back-projected residual vectors will tend to have an averaging effect that yields a higher SNR in h_f.

To ascertain the effect of noise on the h_g portion of the gradient vector (14), it is sufficient to notice that g_j^ϕ is in the intersection of the numerical ranges of P_j and $P(\phi_j + \Delta\phi, \theta_j, \psi_j)$. Therefore, g_j^ϕ is numerically orthogonal to the intersection of the null spaces of P_j^T and $P^T(\phi_j + \Delta\phi, \theta_j, \psi_j)$, which contains most of the noise components in r_j. Hence, the γ_j^ϕ element of the h_g vector, which is the inner product between g_j^ϕ and r_j, is unlikely to be contaminated by noise. Similar arguments can be applied to the γ_j^ϕ and γ_j^ϕ elements of the h_g vector.

4. DISCUSSION

We have formulated the single particle reconstruction and orientation determination problems as a unified nonlinear optimization problem. To solve it, we have applied a Quasi-Newton method and demonstrated that the method allows simultaneous refinement of the 3-D density map and of the Eulerian angles that describe orientations of 2-D projections. We have illustrated how to approximate gradient of the objective function through finite difference, and have pointed out that the gradient calculation itself is significantly computationally less expensive than performing an exhaustive search in the orientation parameter space, as is done in the projection matching algorithm. We also have argued that the gradients would not be as sensitive to the noise in the data as one could expect. Using simulated data we have demonstrated that the numerical scheme we have developed indeed converges to the desired optimal solution. In our tests the initial guess used to start off the iterative optimization procedure was within practically encountered vicinity of the true minimizer of the objective function.

The algorithm proposed in this paper remains a local optimization algorithm that is only effective when one has an initial approximation to the 3-D structure and initial estimates to the orientation parameters that are sufficiently close to the optimal solution of the problem (1). Our algorithm is most effective when a globalization strategy is available to bring the initial guess of the 3-D structure and orientation parameters within the convergence radius. While in this work we have omitted the translation errors, the presented mathematical framework is quite general and additional optimization parameters, such as shift and defocus settings can be easily introduced.

Although the gradient calculation can be carried out efficiently, the Quasi-Newton search direction provided by the LBFGS algorithm may not be the best search direction in terms of the convergence rate of the optimization algorithm. An alternative to the LBFGS algorithm is the Gauss-Newton algorithm commonly used to solve nonlinear least squares problems. In a Gauss-Newton algorithm, the true Hessian of $\rho(x)$ is approximated by $B = J^T J$, where J is the Jacobian matrix defined in (8). Consequently, we need to solve the linear system

$$J(x_k)^T J(x_k) s_k = -\nabla\rho(x_k) \tag{16}$$

at each Gauss-Newton iteration to obtain a Gauss-Newton search direction. Because the Jacobian matrix is quite large but sparse, it is more appropriate to solve the above linear system by using an iterative method such as the LSQR algorithm developed in (Paige and Saunders, 1982). An iterative method does not require $B = J^T J$ to be formed explicitly. It only requires one to provide an efficient way to calculate the matrix vector product of the form $y \leftarrow J^T J\, x$. In our case, this is entirely possible due to the sparsity structure of J illustrated in (8). The need to solve the linear system (14) makes the Gauss-Newton method somewhat less attractive because the method is more expensive per iteration. However, since the Gauss-Newton method may provide a better search direction, it may reduce the number of iterations required to reach a local minimizer of $\rho(x)$ significantly. The trade-off between following better search directions to reduce the number of iterative steps required to reach the minimum of (1) and the computation cost for generating these search directions will be explored in future studies.

More work is also required to investigate the performance of the iterative optimization scheme on realistic data. In particular, we plan to further investigate the effectiveness of the optimization scheme on noisy projection images. The method is not expected to be effective when the SNR in the projection data is very low because in that case the derivative calculations based on finite difference typically do not provide a reliable search direction. One possible work around to this problem would be to identify an appropriate surrogate function for $\rho(x)$. A surrogate function is a function that shares the same local minimizer with that of the true objective function. It must be smooth and easy to evaluate, although it may notably differ from the true objective function outside of the neighborhood of the optimal solution. One potential candidate of a surrogate function is

$$\hat{\rho}(\phi_i, \theta_i, \psi_i, f) = \frac{1}{2}\sum_{i=1}^{m} \left\| P(\phi_i, \theta_i, \psi_i) f - \hat{b}_i \right\|^2, \tag{17}$$

where $\hat{b}_i$ is a low-pass filtered version of b_i.

In addition to the use of a surrogate function, we may also apply regularization techniques to the optimization procedure to reduce noise amplification. Several regularization techniques have recently been investigated (Hanke and Hansen, 1993). One commonly used technique is to add a penalty term in (1) to prevent the noise component in the data to grow. That is, we may choose to optimize, for example,

$$\hat{\rho}(\phi_i,\theta_i,\psi_i, f) = \frac{1}{2}\sum_{i=1}^{m}\left\|P(\phi_i,\theta_i,\psi_i) f - \hat{b}_i\right\|^2 + \lambda\left\|f\right\|^2, \qquad (18)$$

where λ is a judiciously chosen regularization parameter. If the Gauss-Newton algorithm is used to choose a search direction, one can then apply the technique of trust-region to regularize the optimization procedure. The resulting algorithm is the well-known Levenberg-Marquardt algorithm (Levenberg, 1944; Marquardt, 1963; More, 1978).

The presented algorithm has to be treated as a proof of concept rather than as a demonstration of a fully functional method. Nevertheless, the preliminary results are sufficiently encouraging to warrant this report. Moreover, we clearly outlined future directions of work and we argued that incorporation of additional terms is feasible and mathematically tractable. The addition of translation parameters is straightforward. The defocus settings, ignored in present work, can be added on two levels. First, the functional (1) can be expanded in order to explicitly take into account various defocus settings of the EM data. This was previously attempted by others and us and the results were encouraging (Penczek *et al.*, 1997; Sorzano *et al.*, 2004; Zhu *et al.*, 1997; Zubelli *et al.*, 2003). Second, since the initial defocus settings of the particles (or groups of them) are usually known, the related variables can be inserted into (1) and refined along with other variables (Mouche *et al.*, 2001). Finally, the unified framework of (1) can be further expanded along the lines of (18) to include additional terms, in particular to integrate the homology modeling (Marti-Renom *et al.*, 2000) with the EM structure determination.

Acknowledgments

We thank Eva Nogales and Frank Andel for providing the TFIID data set. This research used resources of the National Energy Research Scientific Computing Center, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC03-76SF00098. This work was supported by Grants NIH P01 GM 064692 and NIH R01 GM 60635 (to P.A.P.) and NIH P01 GM 064692 (to E.G.N.)

References

Andel, F., Ladurner, A. G., Inouye, C., Tjian, R., Nogales, E., 1999. Three-dimensional structure of the human TFIID-IIA-IIB complex. Science 286, 2153-2156.

Beckmann, R., Bubeck, D., Grassucci, R., Penczek, P., Verschoor, Blobel, G., Frank, J., 1997. Alignment of conduits for the nascent polypeptide chain in the ribosome-Sec61 complex. Science 278, 2123-2126.

Boisset, N., Penczek, P., Taveau, J. C., Lamy, J., Frank, J., 1995. Three-dimensional reconstruction of *Androctonus australis* hemocyanin labeled with a monoclonal Fab fragment. J. Struct. Biol. 115, 16-29.

Craighead, J. L., Chang, W. H., Asturias, F. A., 2002. Structure of yeast RNA polymerase II in solution: implications for enzyme regulation and Interaction with promoter DNA. Structure 10, 1117-1125.

Frank, J., Radermacher, M., Penczek, P., Zhu, J., Li, Y., Ladjadj, M., Leith, A., 1996. SPIDER and WEB: processing and visualization of images in 3D electron microscopy and related fields. J. Struct. Biol. 116, 190-199.

Gabashvili, I. S., Agrawal, R. K., Spahn, C. M., Grassucci, R. A., Svergun, D. I., Frank, J., Penczek, P., 2000. Solution structure of the *E. coli* 70S ribosome at 11.5 Å resolution. Cell 100, 537-549.

Goncharov, A. B., 1986. Integral geometry and three-dimensional reconstruction of objects [in Russian]. Preprint of Cybernetic Council Acad. Sci., Moscow.

Goncharov, A. B., Vainshtein, B. K., Ryskin, A. I., Vagin, A. A., 1987. Three-dimensional reconstruction of arbitrarily oriented identical particles from their electron photomicrographs. Sov. Phys. Crystallography 32, 504-509.

Grigorieff, N., 1998. Three-dimensional structure of bovine NADH: ubiquinone oxidoreductase (complex I) at 22 Å in ice. J. Mol. Biol. 277, 1033-1046.

Hanke, M., Hansen, P. C., 1993. Regularization methods for large-scale problems. Surveys Math. Indust. 3, 253-315.

Hansen, P. C., 1997. Rank-deficient and discrete ill-posed problems. SIAM, Philadelphia, PA.

Joyeux, L., Penczek, P. A., 2002. Efficiency of 2D alignment methods. Ultramicroscopy 92, 33-46.

Levenberg, K., 1944. A method for the solution of certain non-linear problems in least squares. Quarterly of Applied Mathematics 2, 164-168.

Marquardt, D., 1963. An algorithm for least squares estimation of non-linear parameters. SIAM Journal of Applied Mathematics 11, 431-441.

Marti-Renom, M. A., Stuart, A. C., Fiser, A., Sanchez, R., Melo, F., Sali, A., 2000. Comparative protein structure modeling of genes and genomes. Annual Review of Biophysics & Biomolecular Structure 29, 291-325.

More, J. J. (1978) The Levenberg-Marquardt algorithm: Implementation and theory, *in* G. Watson (Ed.), Lecture Notes in Mathematics, No.630 - Numerical Analysis,, pp. 105-116.

Mouche, F., Boisset, N., Penczek, P. A., 2001. *Lumbricus terrestris* hemoglobin - The architecture of linker chains and structural variation of the central toroid. J. Struct. Biol. 133, 176-192.

Navaza, J., 2003. On the three-dimensional reconstruction of icosahedral particles. J. Struct. Biol. 144, 13-23.

Norcedal, J., 1980. Updating Quasi-Newton matrices with limited storage. Math. Comp. 35, 773-782.

Norcedal, J., 1991. Theory of algorithms for unconstrained optimization. Acta Numerica 1, 199-242.

Norcedal, J., Wright, S. J., 1999. Numerical Optimization. Springer, New York.

Pacheco, P. S., 1996. Parallel Programming with MPI. Morgan Kaufmann, San Francisco.

Paige, C. C., Saunders, M. A., 1982. LSQR: An algorithm for sparse linear equations and sparse least squares. ACM Trans. Math. Software 8, 43-71.

Penczek, P., Radermacher, M., Frank, J., 1992. Three-dimensional reconstruction of single particles embedded in ice. Ultramicroscopy 40, 33-53.

Penczek, P. A., Grassucci, R. A., Frank, J., 1994. The ribosome at improved resolution: new techniques for merging and orientation refinement in 3D cryo-electron microscopy of biological particles. Ultramicroscopy 53, 251-270.

Penczek, P. A., Renka, R., Schomberg, H., 2004. Gridding-based direct Fourier inversion of the three-dimensional ray transform. J. Opt. Soc. Am. A 21, 499-509.

Penczek, P. A., Zhu, J., Frank, J., 1996. A common-lines based method for determining orientations for N > 3 particle projections simultaneously. Ultramicroscopy 63, 205-18.

Penczek, P. A., Zhu, J., Schröder, R., Frank, J., 1997. Three-dimensional reconstruction with contrast transfer function compensation from defocus series. Scanning. Microsc. Suppl. 11, 1-10.

Provencher, S. W., Vogel, R. H., 1988. Three-dimensional reconstruction from electron micrographs of disordered specimens. I. Method. Ultramicroscopy 25, 209-21.

Radermacher, M., 1994. Three-dimensional reconstruction from random projections: orientational alignment *via* Radon transforms. Ultramicroscopy 53, 121-136.

Radermacher, M., Ruiz, T., Wieczorek, H., Gruber, G., 2001. The structure of the V(1)-ATPase determined by three-dimensional electron microscopy of single particles. J. Struct. Biol. 135, 26-37.

Radermacher, M., Wagenknecht, T., Verschoor, A., Frank, J., 1987. Three-dimensional reconstruction from a single-exposure, random conical tilt series applied to the 50S ribosomal subunit of *Escherichia coli*. J. Microsc. 146, 113-36.

Saxton, W. O., Baumeister, W., 1982. The correlation averaging of a regularly arranged bacterial envelope protein. J. Microsc. 127, 127-138.

Sorzano, C. O., Marabini, R., Herman, B., Censor, Y., Carazo, J. M., 2004. Transfer function restoration in 3D electron microscopy via iterative data refinement. Phys. Med. Biol. 49, 509-522.

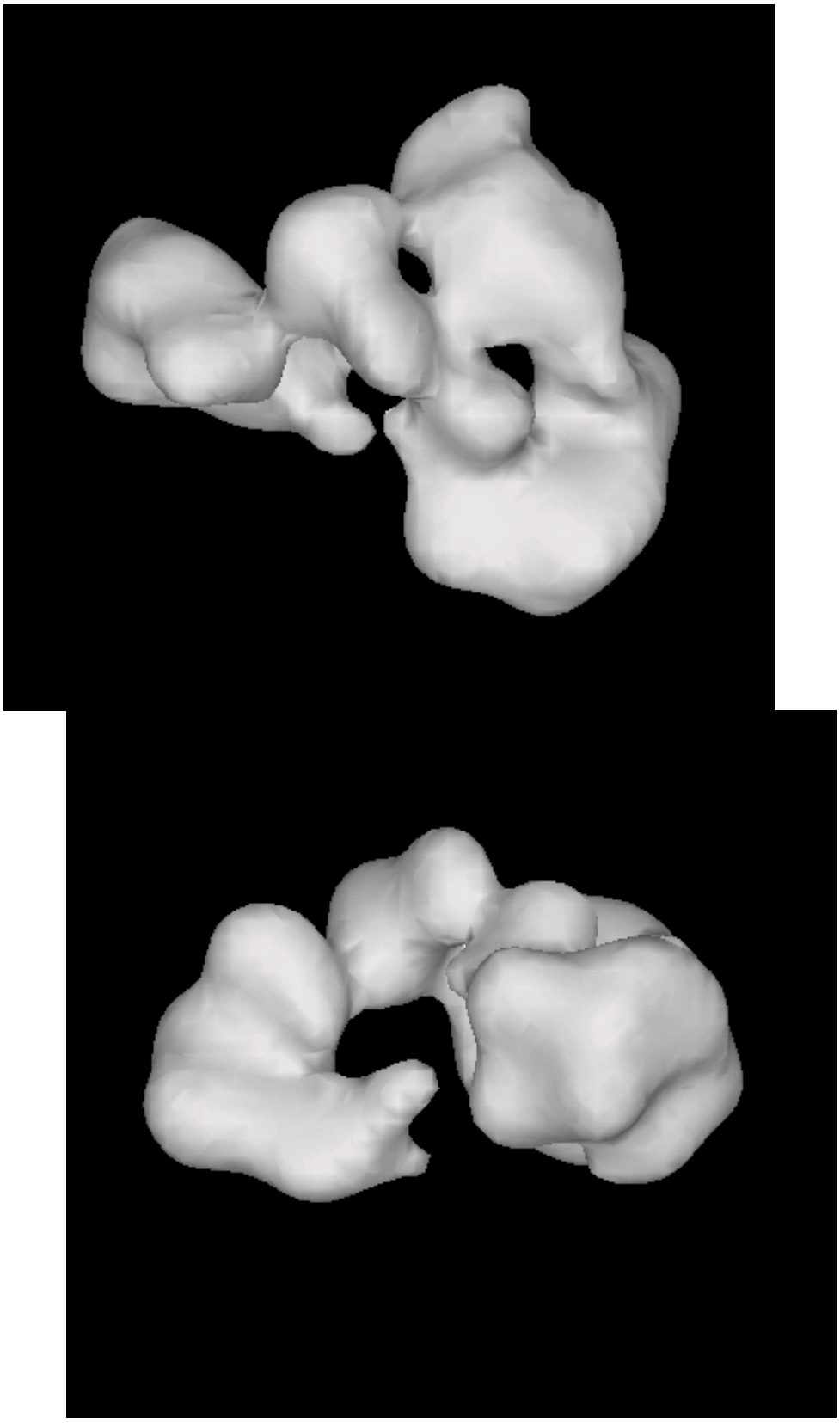
van Heel, M., 1987. Angular reconstitution: *a posteriori* assignment of projection directions for 3D reconstruction. Ultramicroscopy 21, 111-124.

Yin, Z. H., Zheng, Y. L., Doerschuk, P. C., Natarajan, P., Johnson, J. E., 2003. A statistical approach to computer processing of cryo-electron microscope images: virion classification and 3-D reconstruction. J. Struct. Biol. 144, 24-50.

Zhu, J., Penczek, P. A., Schröder, R., Frank, J., 1997. Three-dimensional reconstruction with contrast transfer function correction from energy-filtered cryoelectron micrographs: procedure and application to the 70S *Escherichia coli* ribosome. J. Struct. Biol. 118, 197-219.

Zubelli, J. P., Marabini, R., Sorzano, C. O., Herman, G. T., 2003. Three-dimensional reconstruction by Chahine's method from electron microscopic projections corrupted by instrumental aberrations. Inverse Problems 19, 933-949.

Figure Legends



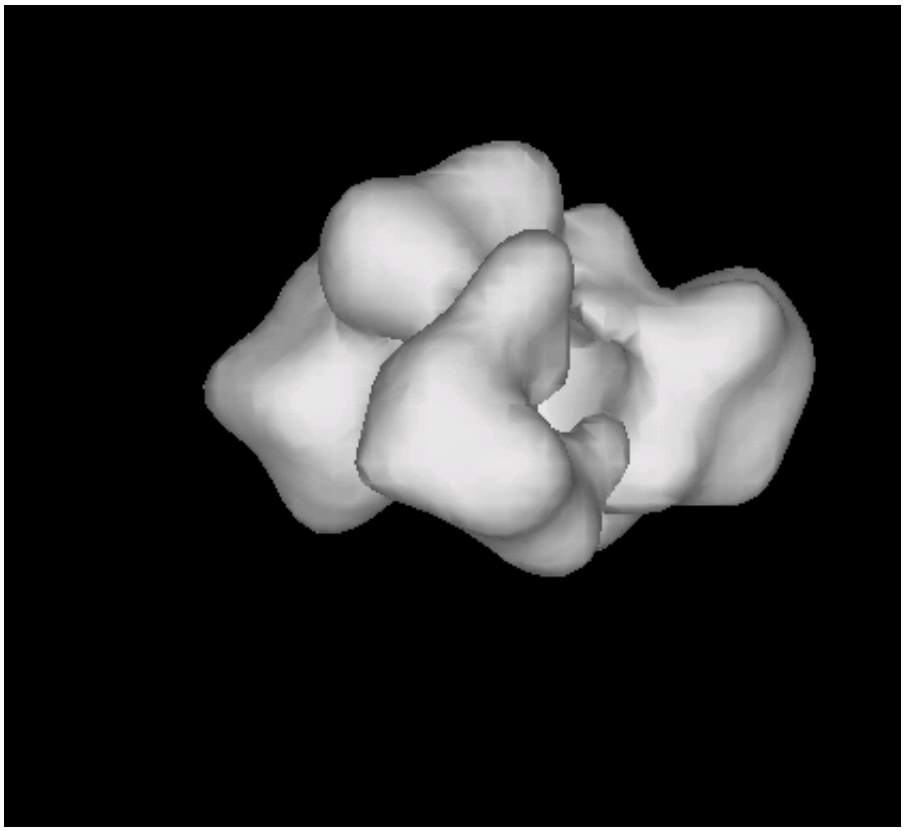
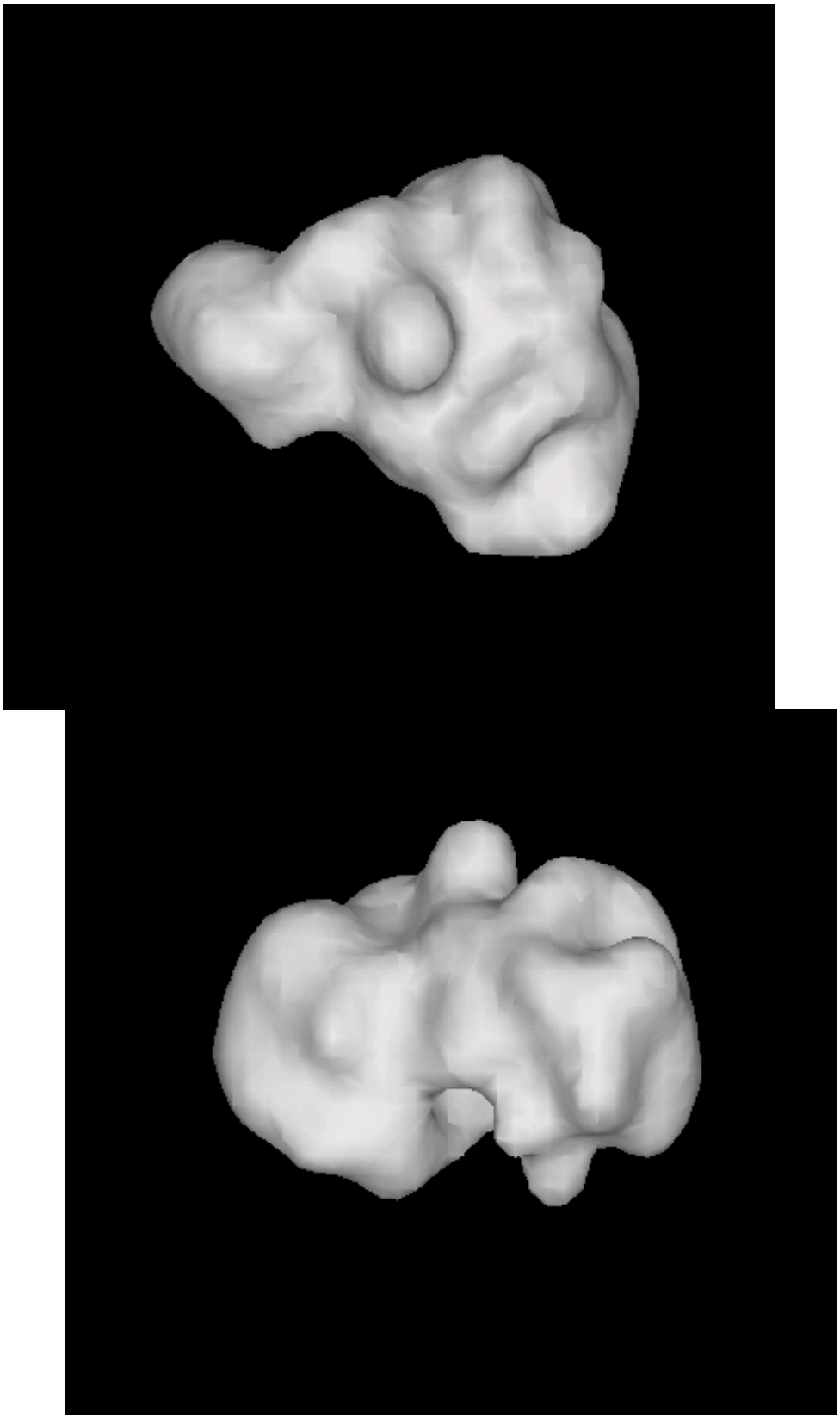


Figure 1. The isosurfaces of the TFIID structure from three different viewing angles. These surface renderings are generated by Vis5D (http://vis5d.sourceforge.net/). The leftmost view (a) is the top view; (b) is the front view obtained by rotating (a) by 90° around the horizontal axis; and (c) is obtained by rotating (b) by another 90° around the vertical axis.



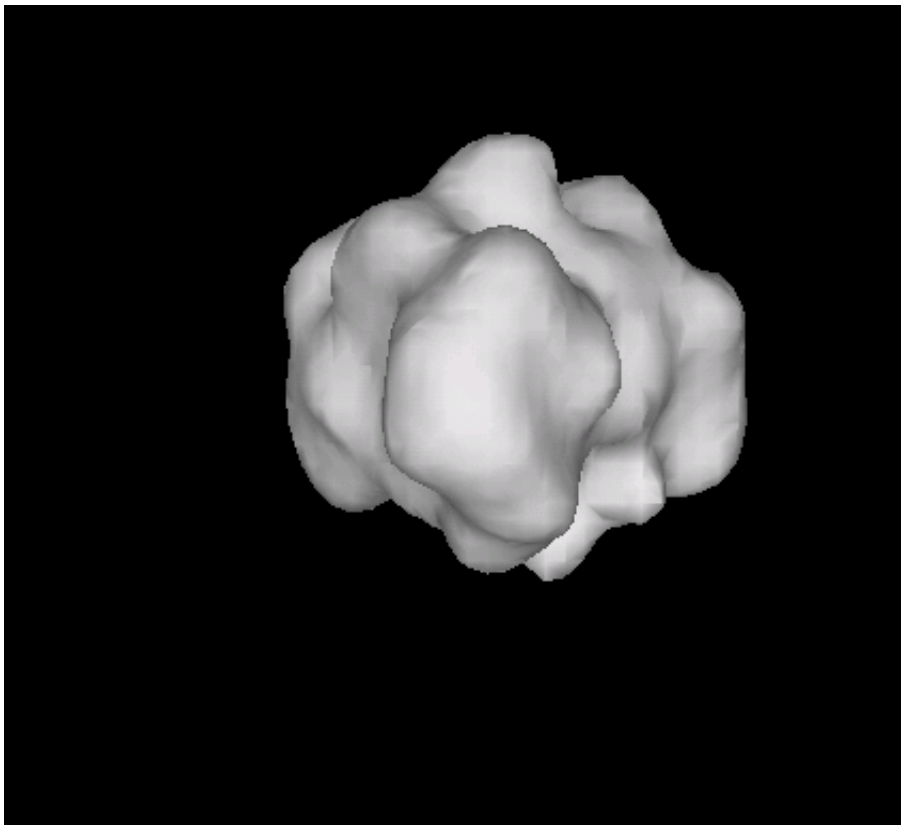


Figure 2. The isosurface of the initial guess of the TFIID structure from three different view angles. The leftmost view (a) is the top view; (b) is the front view obtained by rotating (a) by 90° around the horizontal axis; and (c) is obtained by rotating (b) by another 90° around the vertical axis.

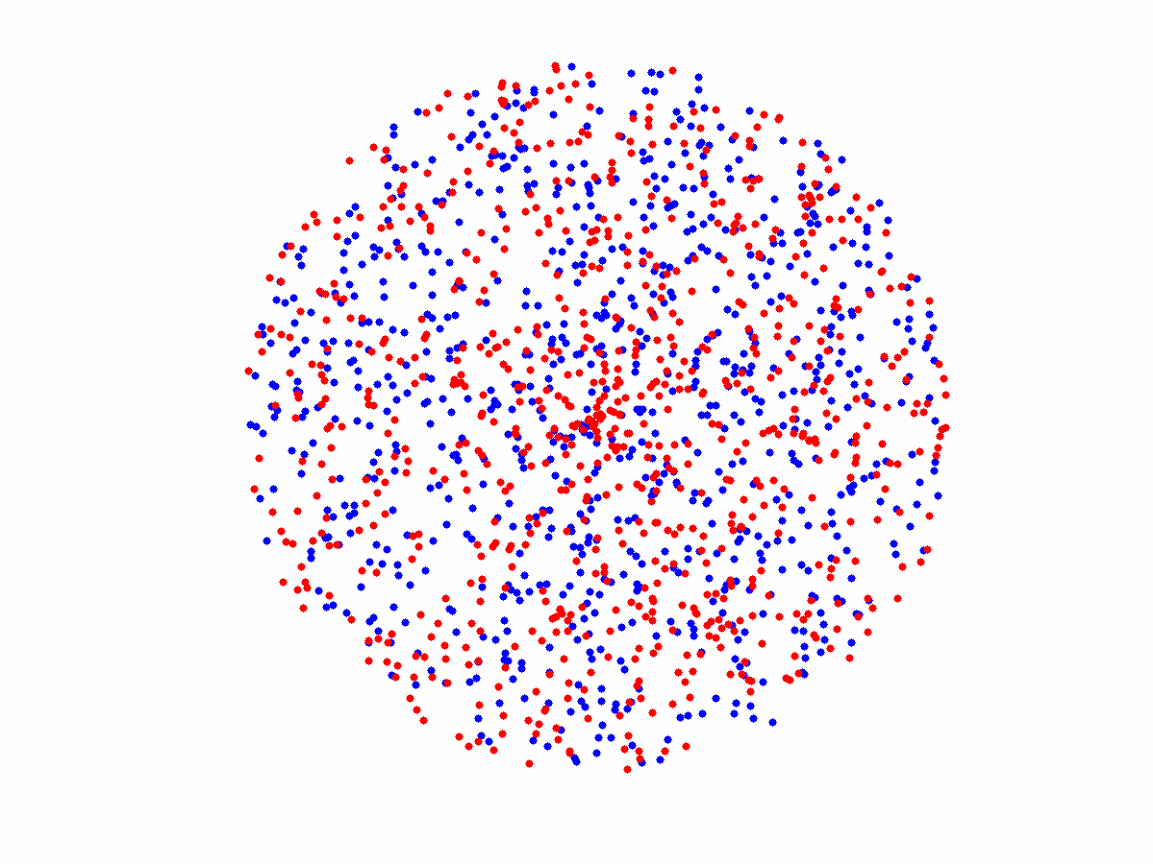


Figure 3. The distribution of the exact projection directions defined by (ϕ_i, θ_i) (the blue dots) and initial guesses of these projection directions defined by $(\hat{\phi}_i, \hat{\theta}_i)$ (the red dots).

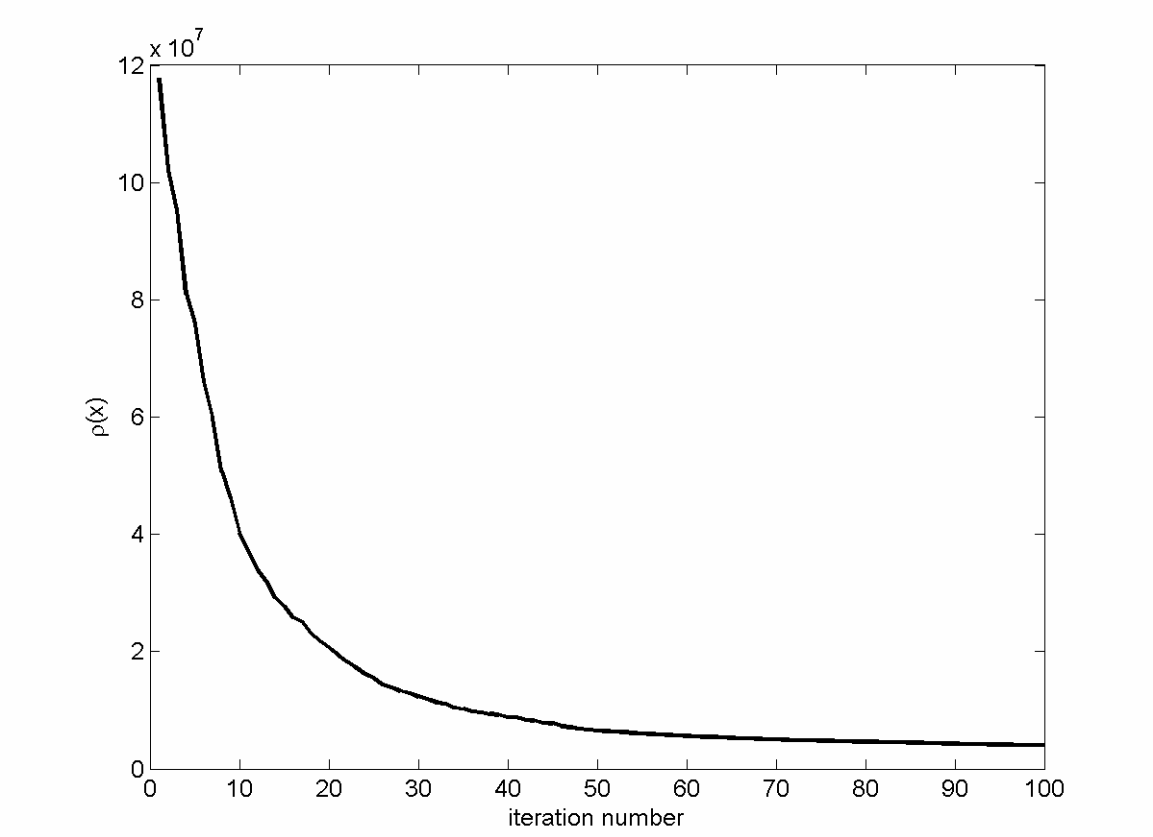


Figure 4. The objective function defined in (1) decreases monotonically during the first 100 LBFGS iterations.

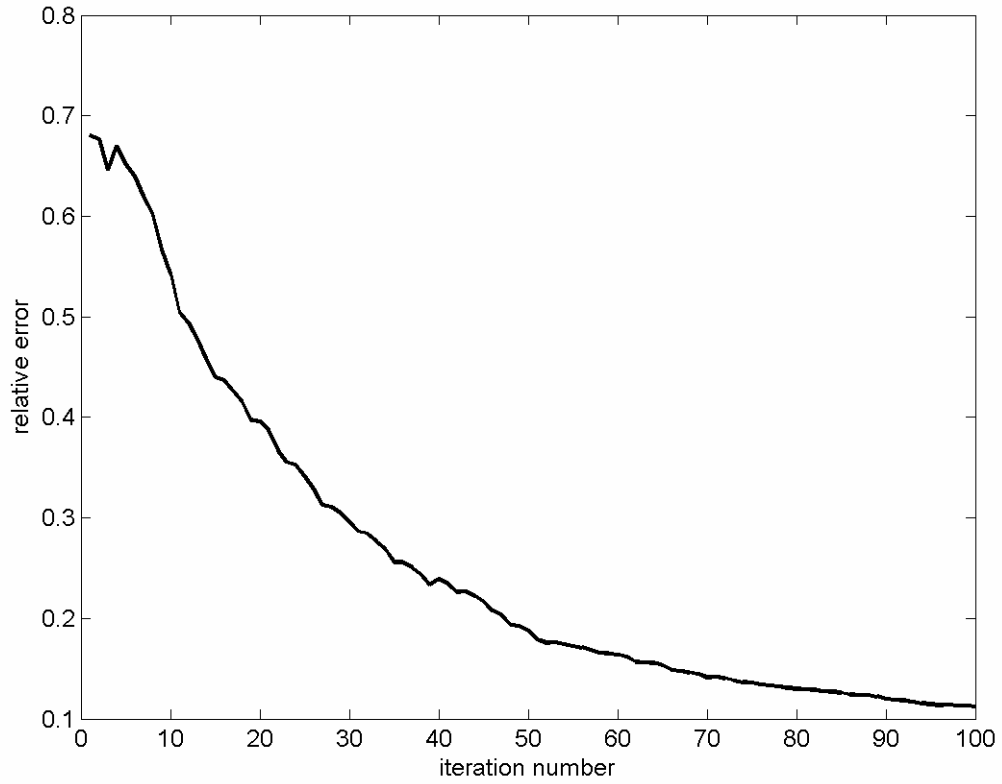
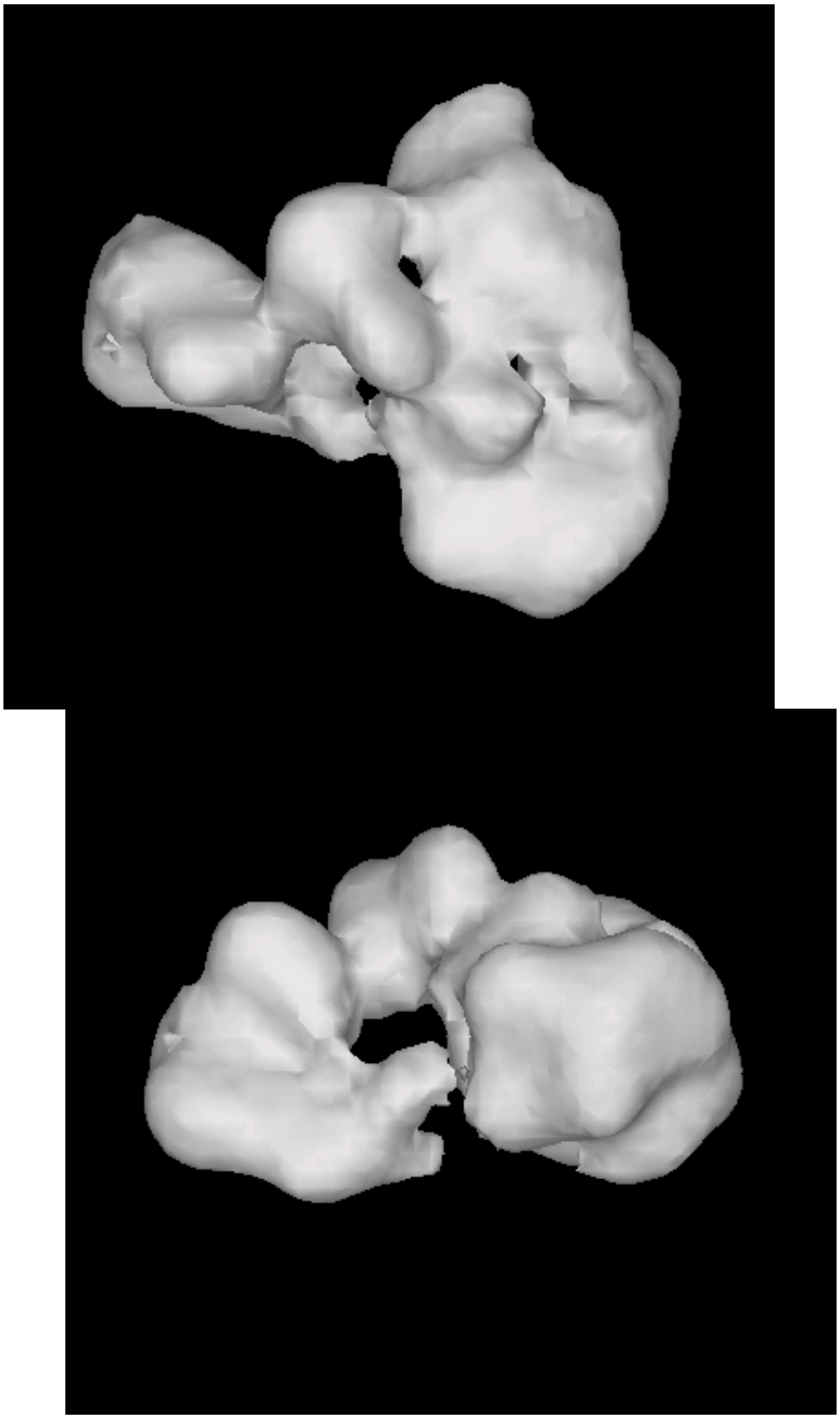


Figure 5. The relative error of the 3-D structure as a function of the number of LBFGS iterations.



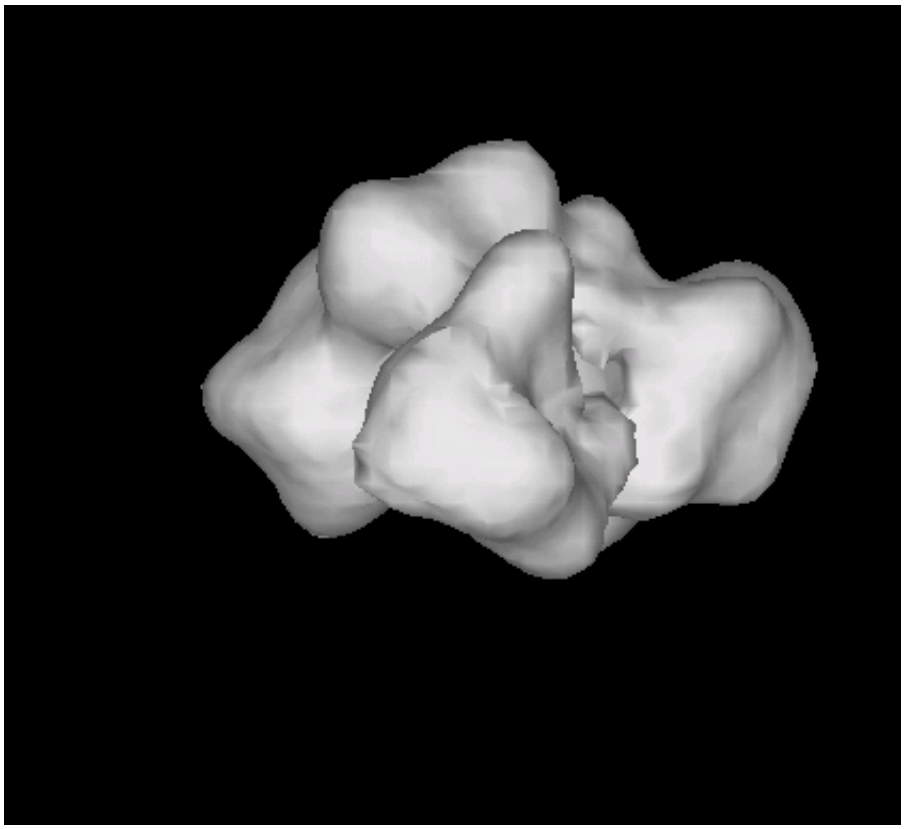


Figure 6. The reconstructed 3-D structure of TFIID. The leftmost view (a) is the top view; (b) is the front view obtained by rotating (a) by 90° around the horizontal axis; and (c) is obtained by rotating (b) by another 90° around the vertical axis.

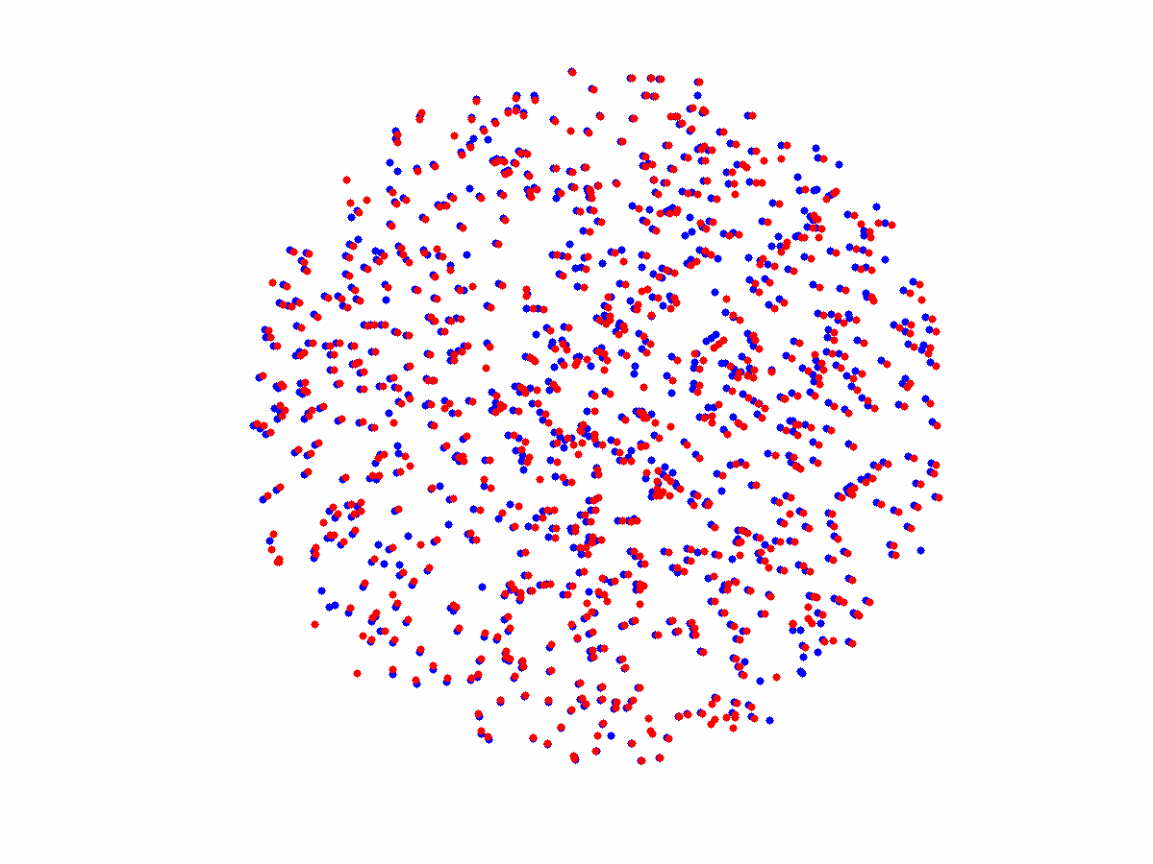


Figure 7. Comparison of the exact (blue dots) and estimated projection directions (red dots).

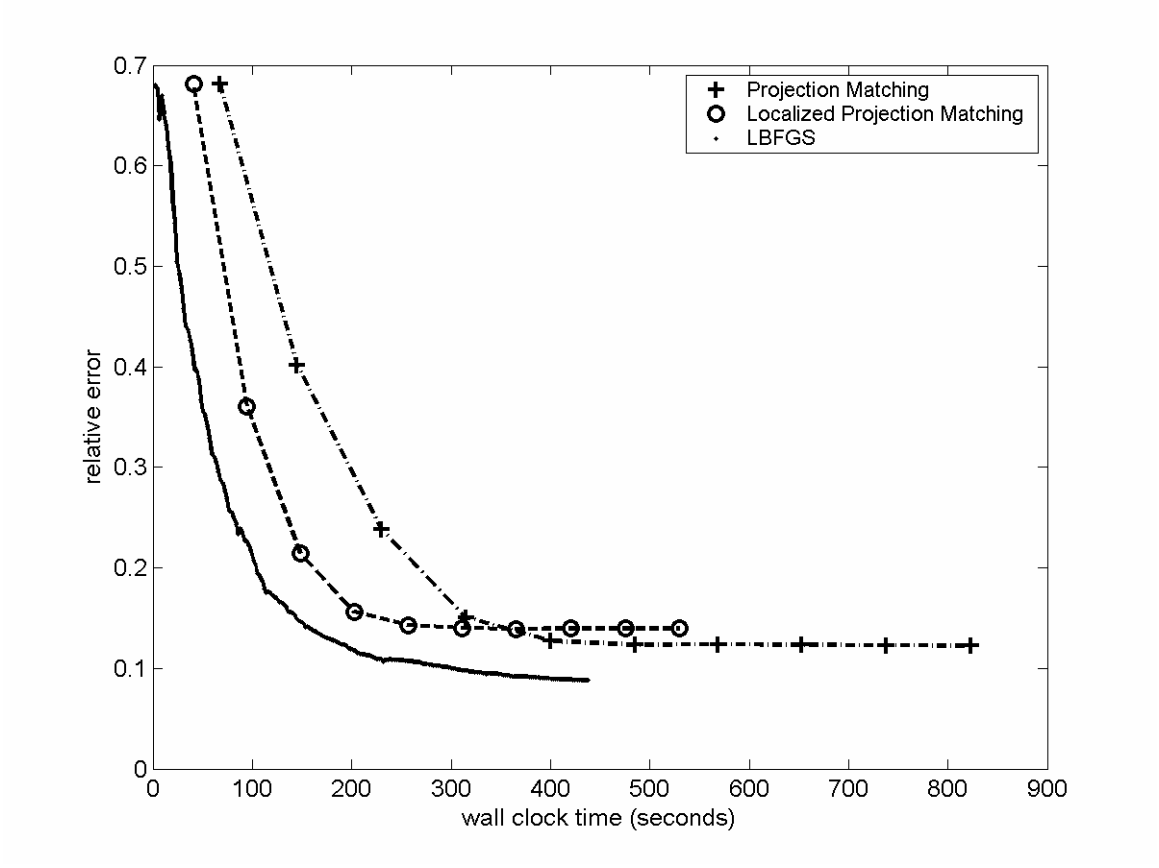


Figure 8. Comparison of computational time required consumed by projection matching and by LBFGS. The relative error is plotted as a function of the wall clock time used on a 16×375Mhz IBM Power3 processors.

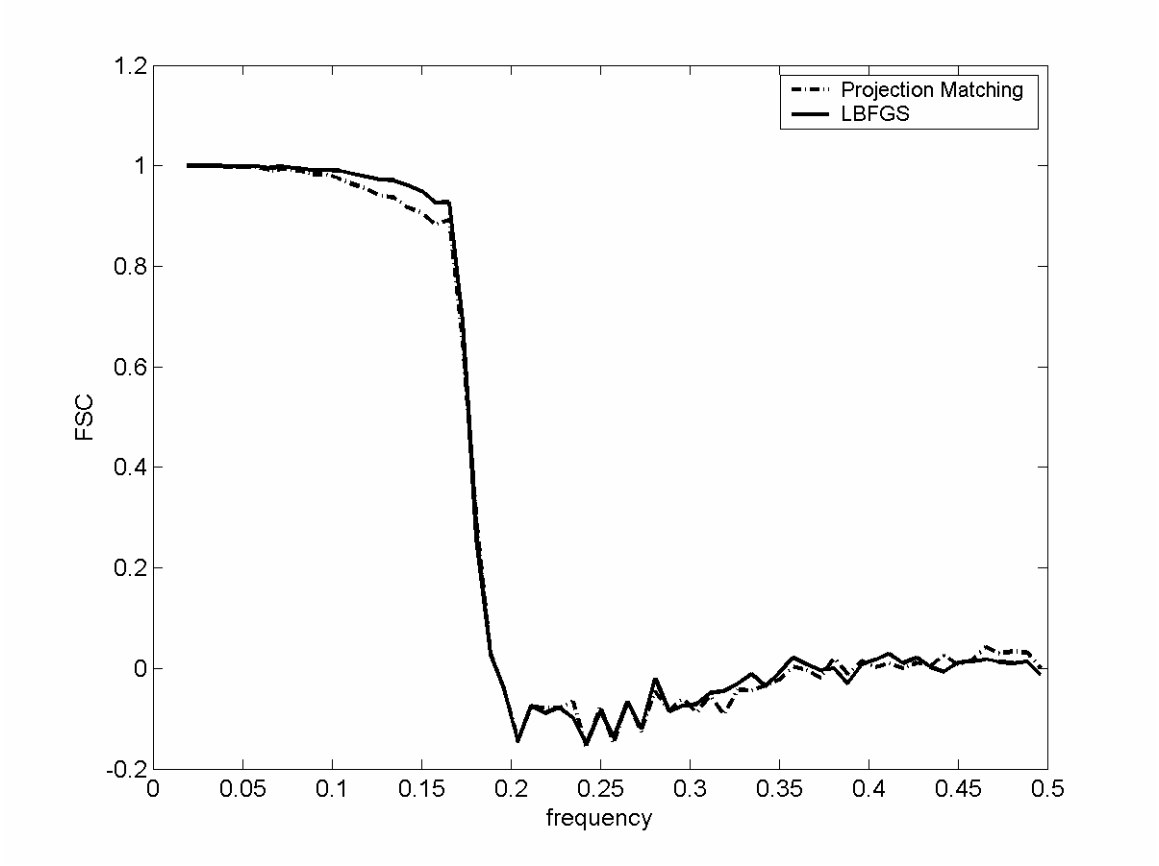


Figure 9. Comparison of the FSC curves produced by projection matching and simultaneous refinement.

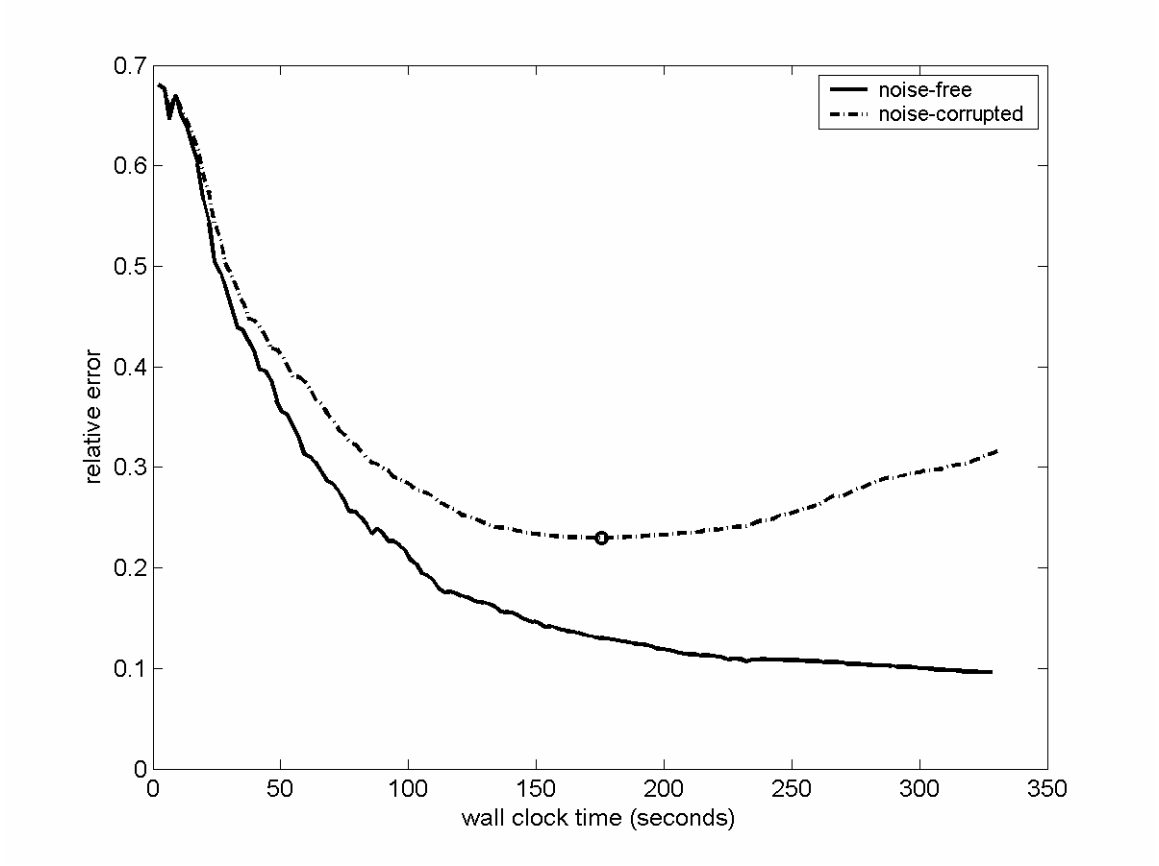


Figure 10. Comparison of relative errors as a function of number of iterations (expressed in terms of wall clock time) for LBFGS algorithm applied to noise-free and noise-corrupted TFIID data.

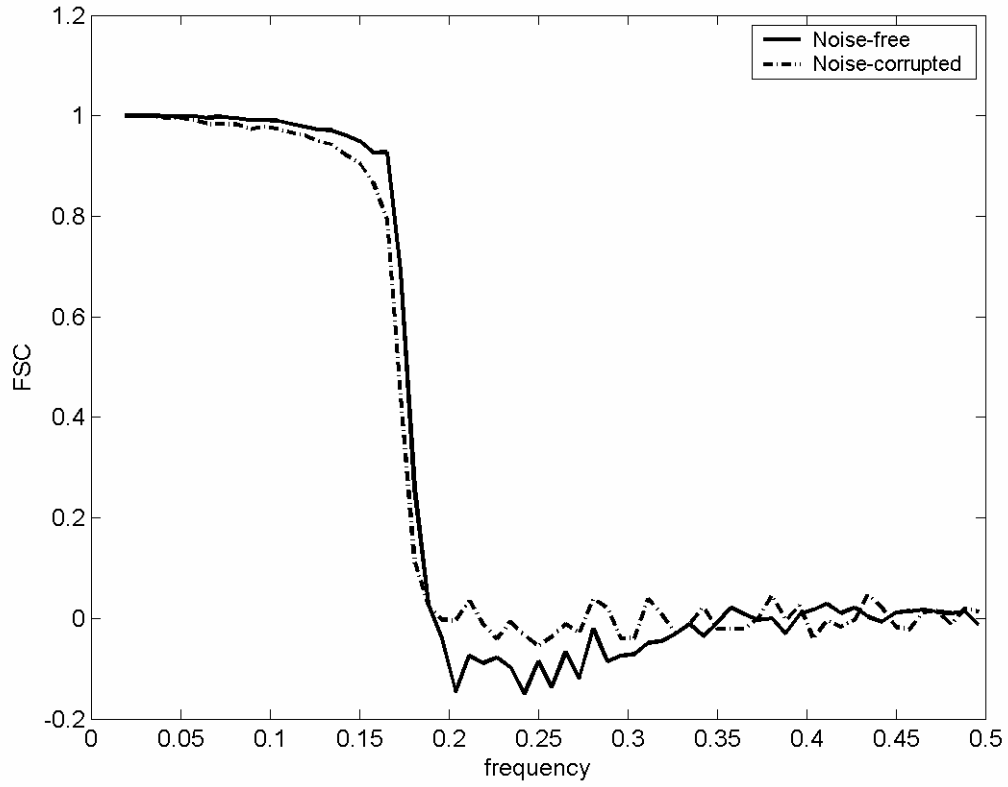


Figure 11. Comparison of the FSC curves associated with a noise-free and a noise-corrupted refinement of the TFIID data.