$$\Phi_{3_i}(u, v, w) = \Phi_{2_i}(\mathbf{u}_\tau) D \operatorname{sinc}(D\pi w_\tau), \tag{2.5.6.33}$$

and the 3D reconstruction is obtained by the inverse Fourier transform of the sum of contributions given by (2.5.6.33),

$$\mathscr{F}[\hat{b}(\mathbf{r})] = \sum_i \Phi_{2_i}(\mathbf{u}_\tau) D \operatorname{sinc}(D\pi w_\tau). \tag{2.5.6.34}$$

z_τ and w_τ are variables in real and Fourier spaces, respectively, both extending in the direction of the projection direction $\boldsymbol{\tau}(\theta, \psi)_i$.

In this analysis, the transfer function of the backprojection algorithm is obtained by setting $\Phi_{2_i}(\mathbf{u}_\tau) = 1$, that is by finding the response of the algorithm to the input composed of delta functions in real space. This yields the inversion formula for a general weighted backprojection algorithm:

$$\varphi_3(\mathbf{r}) = \mathscr{F}[\hat{b}(\mathbf{r})] c(u, v, w), \tag{2.5.6.35}$$

with the weighting function given by

$$c(u, v, w) = \frac{1}{\sum_i D \operatorname{sinc}(D\pi w_\tau)}. \tag{2.5.6.36}$$

The general weighting function (2.5.6.36) is consistent with analytical solutions (2.5.6.11) and (2.5.6.14), as it can be shown that by assuming infinite support ($D \rightarrow \infty$) and continuous and uniform distribution of projection directions, in two dimensions one obtains $c(u, v) = (u^2 + v^2)^{-1/2}$ (Radermacher, 2000).

The derivation of (2.5.6.36) is based on the analysis of continuous functions and its direct application to discrete data results in reconstruction artifacts; therefore, Radermacher (1992) proposed attenuating the sinc functions in (2.5.6.36) by exponent functions with decay depending on the diameter of the structure D, or simply replacing the sinc functions by exponent functions. This, however, reduces the concept of the weighting function corresponding to the deconvolution to the concept of the weighting function representing the 'local density' of projections (2.5.6.30). The general weighted backprojection reconstruction is implemented in *SPIDER* (using exponent-based weighting functions) (Frank *et al.*, 1996), *Xmipp* (Sorzano *et al.*, 2004) and *SPARX* (Hohn *et al.*, 2007).

Harauz & van Heel (1986) proposed basing the calculation of the density of projections, thus the weighting function, on the overlap of Fourier transforms of projections in Fourier space. Although the concept is general, it can be easily approached in two dimensions. If the diameter of the object is D, the width of a Fourier transform of a projection is $2/D$ (Fig. 2.5.6.5), as follows from the central section theorem (2.5.6.8). Harauz and van Heel postulated that the weighting should be inversely proportional to the sum of geometrical overlaps between a given central section and the remaining central sections. For a pair of projections il, this overlap is

$$o_{il}(R) = T[DR \sin(\Psi_i - \Psi_l)], \tag{2.5.6.37}$$

where T represents the triangle function (selected by the authors because it can be calculated efficiently). Also, owing to the Friedel symmetry of central sections, the angles in (2.5.6.37) are restricted such that $0 \leq \Psi_i - \Psi_l \leq (\pi/2)$. In this formulation, the overlap is limited to the maximum frequency,

$$R_{il}^{\max} = \frac{2}{D \sin(\Psi_i - \Psi_l)}. \tag{2.5.6.38}$$

Thus, the overlap function becomes

$$o_{il}(R) = \begin{cases} 1 - (R/R_{il}^{\max}) & 0 \leq R < u_{il}^{\max} \\ 0 & R > u_{il}^{\max} \end{cases}. \tag{2.5.6.39}$$

In effect, the weighting function, called by the authors an 'exact filter', is

$$c(R, \Psi_i) = \frac{1}{1 + \sum_{l, l \neq i} o_{il}(R)}. \tag{2.5.6.40}$$

The weighting function (2.5.6.40) easily extends to three dimensions; however, the calculation of the overlap between central sections in three dimensions (represented by slabs) is more elaborate (Harauz & van Heel, 1986). The method is conceptually simple and computationally efficient. However, (2.5.6.40) does not approximate the correct weighting well for a uniform distribution of projections [*i.e.*, it should yield $c(R, \Psi_i) = R$]. This, as can be seen by integrating (2.5.6.39) over the whole angular range, is not the case. The exact filter backprojection reconstruction is implemented in the *IMAGIC* (van Heel *et al.*, 1996), *SPIDER* (Frank *et al.*, 1996) and *SPARX* (Hohn *et al.*, 2007) packages.

The 3D reconstruction methods based on filtered backprojection are commonly used in single-particle reconstruction. The reasons are: their versatility, ease of implementation, and – in comparison with iterative methods – good computational efficiency. Unlike in iterative methods, there are no parameters to adjust, although it has been noted that the results depend on the value of the diameter D of the structure in all three weighting functions [(2.5.6.30), (2.5.6.36) and (2.5.6.38)], so the performance of the reconstruction algorithm can be optimized for a particular data-collection geometry by changing the value of D (Paul *et al.*, 2004). However, because computation of the weighting function involves calculation of pairwise distances between projections, the computational complexity is proportional to the square of the number of projections and for large data sets these methods become inefficient. It also has to be noted that the weighting functions (2.5.6.30), (2.5.6.36) and (2.5.6.40) remain approximations of the correct weighting function (2.5.6.29).

2.5.6.6. Direct Fourier inversion

Direct Fourier methods are based on the central section theorem (2.5.6.8). A set of the 2D Fourier transforms of projections yields an approximation to Φ_3 on a nonuniform 3D grid, and a subsequent numerical 3D inverse Fourier transform gives an approximation to φ_3. If the 3D inverse Fourier transform could be realized by means of the 3D inverse fast Fourier transform (FFT), one would have a very fast reconstruction algorithm. Unfortunately, the preprocessing step yields Φ_3 on a nonuniform grid. In effect, the 3D inverse FFT is not applicable and an additional step of recovering samples of Φ_3 on a uniform grid from the available samples on a nonuniform grid is necessary.

One possibility is to resample the nonuniformly sampled version of Φ_3 onto a 3D Cartesian grid by some form of interpolation. For example, Grigorieff used a modified trilinear interpolation scheme in the *FREALIGN* package (Grigorieff, 1998). Simple interpolation methods have been found to give inaccurate results, although more sophisticated interpolation schemes can go a long way to improve the accuracy (Lanza-

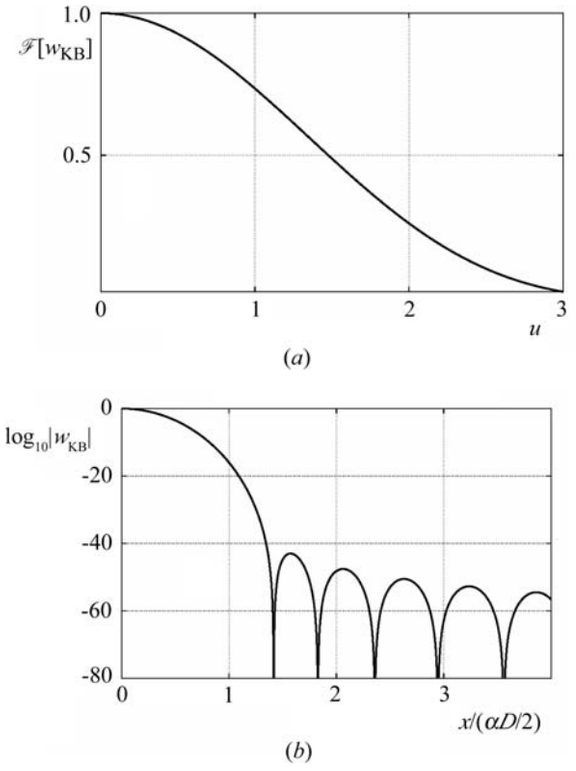

Fig. 2.5.6.6. Kaiser–Bessel window function used in the gridding reconstruction algorithm GDFR. (*a*) In Fourier space, the window function is effectively zero outside the support of six Fourier pixels. (*b*) In real space, the zeros of the window function are beyond the radius of the reconstructed object.

vecchia *et al.*, 1993). Unfortunately, the recommended window size makes them impractical for most applications.

The most accurate Fourier reconstruction methods are those that employ nonuniform Fourier transforms, particularly the 3D gridding method (O'Sullivan, 1985; Schomberg & Timmer, 1995). The gridding-based direct Fourier reconstruction algorithm (GDFR) (Penczek *et al.*, 2004) was developed specifically for single-particle reconstruction. It comprises three steps:

(1) The first step, known as 'gridding', involves calculating for the Fourier transform of each projection the convolution

$$\sum_i \mathscr{F}[w] * (c\mathscr{F}[\varphi_{2_i}]), \tag{2.5.6.41}$$

where c are 'gridding weights' designed to compensate for the nonuniform distribution of the grid points and $\mathscr{F}[w]$ is an appropriately chosen convolution kernel. After processing all projections, this step yields samples of $\mathscr{F}[w] * \mathscr{F}[\varphi_3]$ on a Cartesian grid.

(2) 3D inverse FFT is used to compute

$$w\varphi_3 = \mathscr{F}^{-1}\big[\mathscr{F}[w] * \mathscr{F}[\varphi_3]\big] \tag{2.5.6.42}$$

on a Cartesian grid.

(3) The weights are removed using division:

$$\varphi_3 = w\varphi_3/w. \tag{2.5.6.43}$$

The method involves a number of parameters. First, we need to decide the oversampling factor for the padding with zeros of input projections before FFTs are computed. In GDFR, the oversampling is set to two, although smaller factors can also yield good results. Second, we need a *window function* $\mathscr{F}[w]$ whose support in Fourier space is 'small'. In order to assure good computational efficiency of the convolution step (2.5.6.41), in GDFR this support was set to six Fourier voxels. In addition, in order to prevent division by zeros in (2.5.6.43), the *weighting function* $w = \mathscr{F}^{-1}[\mathscr{F}[w]]$ must be positive within the support of the reconstructed object. A recommendable window function is the separable, bell-shaped Kaiser–Bessel window (O'Sullivan, 1985; Jackson *et al.*, 1991; Schomberg & Timmer, 1995) (Fig. 2.5.6.6*a*):

$$\mathscr{F}[w_{KB}(\mathbf{u})] = \begin{cases} \displaystyle\prod_{v=1}^{d} \frac{I_0\{\pi D\alpha s_v[1-(u_v/s_v)^2]^{1/2}\}}{2s_v}, & \mathbf{u} \in [-\mathbf{s}, \mathbf{s}], \\ 0, & \mathbf{u} \notin [-\mathbf{s}, \mathbf{s}], \end{cases} \tag{2.5.6.44}$$

where I_0 is the zero-order modified Bessel function and $\alpha = 1.25$ is a parameter. The weighting function associated with $\mathscr{F}[w_{KB}]$ is

$$w_{KB}(\mathbf{x}) = \prod_{v=1}^{d} \frac{\sinh\Big[\pi D\alpha s_v\big(1-\{x_v/[\alpha(D/2)]^2\}\big)^{1/2}\Big]}{\pi D\alpha s_v\big(1-\{x_v/[\alpha(D/2)]^2\}\big)^{1/2}}. \tag{2.5.6.45}$$

Finally, the gridding weights c are chosen such that in discrete implementation of (2.5.6.41) and (2.5.6.42) we obtain a Riemann sum approximating the respective integral (Schomberg, 2002). The Voronoi diagram (Okabe *et al.*, 2000) of the sampling points provides a good partitioning of sampling space, so the gridding weights c are computed by constructing a Voronoi diagram of the grid points and by choosing the weights as the volumes of the Voronoi cells (Fig. 2.5.6.7*a*). In cryo-EM, the number of projections, thus the number of sampling points in Fourier space, is extremely large. Thus, although the calculation of the gridding weights *via* the 3D Voronoi diagram of the nonuniformly spaced grid points for $\mathscr{F}[\varphi_3]$ would lead to an accurate direct Fourier method, the method would be very slow and would require excessive computer memory. To circumvent this problem, in GDFR (Penczek *et al.*, 2004) the 2D reverse gridding method is used to compute the Fourier transform of each projection on a 2D polar grid. In this way, $\mathscr{F}[\varphi_3]$ is obtained on a 3D spherical grid, where the grid points are located both on centred spheres and on straight lines through the origin. Accordingly, it becomes possible to partition the sampled region into suitable sampling cells *via* the computation of a 2D Voronoi diagram on a unit sphere, rather than a 3D Voronoi diagram in Euclidean space (Fig. 2.5.6.7*b*). This significantly reduces the memory requirements and improves the computational efficiency of the method, particularly when a fast $O(n \log n)$ algorithm for computing the Voronoi diagram on the unit sphere is employed (Renka, 1997).

The gridding weights in GDFR correspond to the weighting functions in filtered backprojection algorithms. Moreover, setting the gridding weights to be equal to angular regions on the unit sphere directly corresponds to the equating of weighting function (2.5.6.29) in 2D filtered backprojection to the length of an arc on a unitary circle. Thus, both methods yield nearly optimum projection density functions.

The reversed gridding method is obtained by reversing the sequence of steps (1)–(3) of the gridding method:

(1) the input image is divided by the weighting function φ_2/w;

(2) the image is padded with zeros and 2D FFT is used to compute $\mathscr{F}[\varphi_2/w]$;

(3) gridding is used to compute $\mathscr{F}[w] * \mathscr{F}[\varphi_2/w]$ on an arbitrary nonuniform grid.

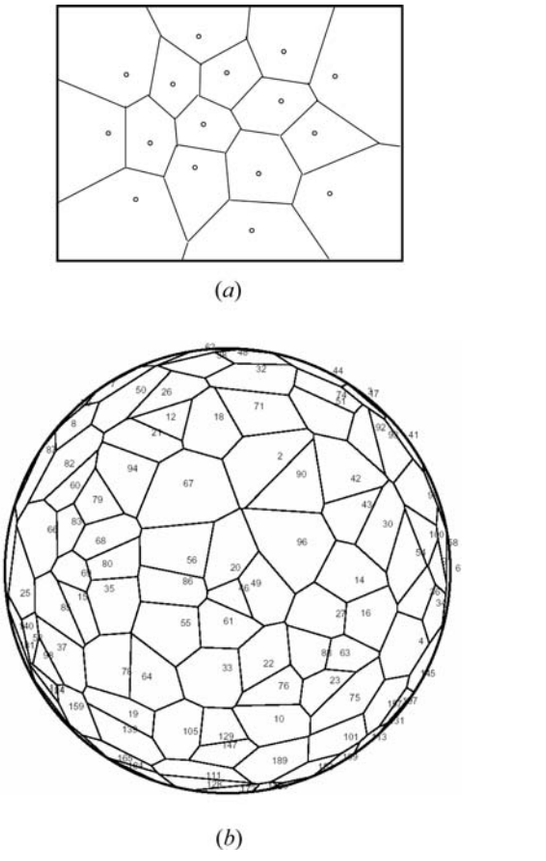

Fig. 2.5.6.7. Partitions of the sampling Fourier space using Voronoi diagrams in direct inversion algorithms. (*a*) Central section of a 3D Fourier volume with sampling points originating from 2D Fourier transforms of projections. Although 2D projections are sampled on a uniform (Cartesian) grid, the arbitrary rotations of projections in 3D space yields a nonuniform distribution of points in three dimensions. In effect, the 3D reconstruction by direct inversion using 3D FFT is not possible. (*b*) Voronoi diagram on a sphere in the GDFR algorithm. Using the reverse gridding method, the 2D Fourier transforms of projections are resampled onto 1D central lines using a constant angular step. In 3D Fourier space, they are located on central sections and their angular directions are evenly distributed on grand circles. However, since central sections have nonuniform distributions, the distribution of angular directions (sampling points on the unitary sphere) is also nonuniform and effectively random.

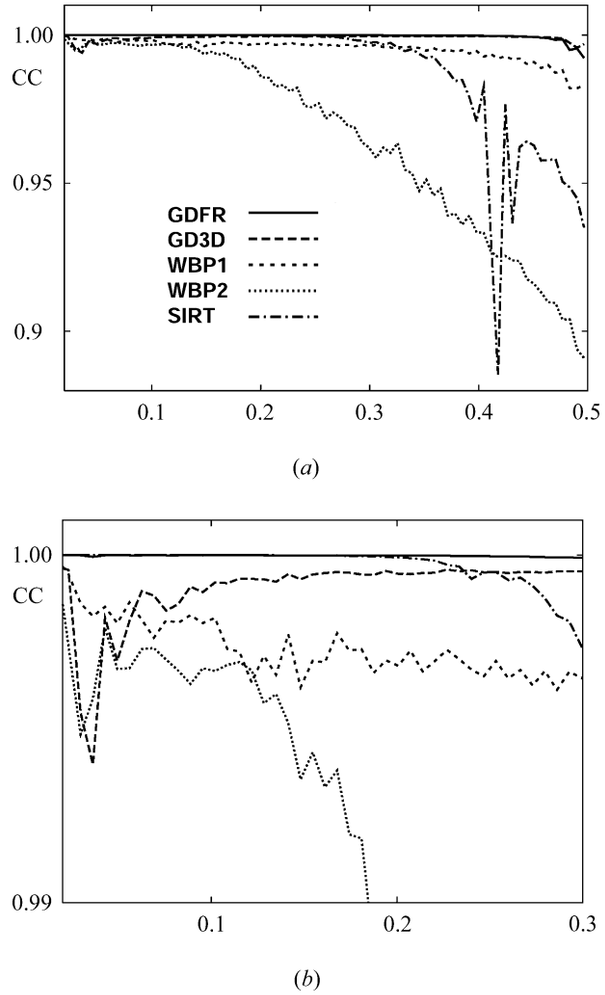

Fig. 2.5.6.8. (*a*) Plot of correlation coefficients (CCs) calculated between Fourier transforms of the reconstructed structure and the original phantom as a function of the magnitude of the spatial frequency using five reconstruction algorithms. GDFR: gridding direct Fourier reconstruction with Voronoi weights; GD3D: gridding reconstruction with simplified gridding weights; WBP1: general weighted backprojection with exponent-based weighting function; WBP2: exact filter weighted backprojection; SIRT: simultaneous iterative reconstruction algorithm. Noise-free projection data were computed in Fourier space using the reverse gridding method. (*b*) Rescaled version of the low-frequency range of (*a*). Note the different scales in (*a*) and (*b*). The horizontal axis is scaled in absolute frequency units with 0.5 equal to the Nyquist frequency.

Note the reverse gridding does not require explicit gridding weights, as in the third step they are constant (Penczek *et al.*, 2004).

In GDFR, the third step of the reversed gridding results in a set of 1D Fourier central lines $\mathcal{F}[\varphi_{1l}]$ calculated using a constant angular step. Clearly, upon inverse Fourier transform they amount to a Radon transform $\varphi_1(u, \psi)$ of the projection and, upon repeating the process for all available projections, they yield a Radon transform $\varphi_1(u, \boldsymbol{\theta})$ of a 3D object φ_3 (albeit non-uniformly sampled with respect to Eulerian angles). Thus GDFR, in addition to being a method of inverting a 3D ray transform, is also a highly accurate method of inverting a 3D Radon transform. GDFR is implemented in the *SPIDER* package (Frank *et al.*, 1996).

The results of a comparison of selected reconstruction algorithms are shown in Fig. 2.5.6.8. The tests were performed using simulated data with the projections of a phantom 3D structure calculated using the inverse gridding method (Penczek *et al.*, 2004). The GDFR yields a virtually perfect reconstruction that agrees with the phantom over the whole frequency range (with the exception of the highest frequencies, but these cannot be reproduced due to geometrical limitations). GD3D is also a gridding-based reconstruction algorithm, in which the gridding weights are calculated directly from contributions of the weighting function to the 3D Fourier pixels and are applied to the voxels in the 3D Fourier volume (Jackson *et al.*, 1991). This yields an approximation to the 'local density' of contributing projections. The simplified gridding weights in GD3D result in deterioration of the reconstruction in the low- and intermediate-frequency ranges. The general weighted backprojection with exponent-based weighting function (WBP1) performs quite well, although reproduction in the low-frequency range is inferior. Similar artifacts are present in the reconstruction using the exact filter weighted backprojection (WBP2), which, in addition, performs disappointingly at higher frequencies. This relatively poor performance is attributed to the nonoptimal weighting schemes used in both methods. It is also interesting to note that the backprojection step is identical in both algorithms; they only differ by the weighting function. The significant difference in their performance attests to the importance of good weighting schemes for high-quality 3D reconstructions from nonuniformly distributed projections. SIRT yields a reconstruction that in the low- and intermediate-frequency ranges matches in quality the reconstruction obtained with GDFR. However, there is a significant loss of quality at high frequencies. This is due to the

inferior linear interpolation scheme used in SIRT (this impediment is shared with WBP1 and WBP2). [The dip of the SIRT fidelity curve at a spatial frequency of 0.42 in Fig. 2.5.6.8(*a*) is due to an inconsistency between the interpolation methods used internally in SIRT to generate intermediate projections of the object to be reconstructed and the Fourier-space-based method used to generate the test data.] In terms of computational efficiency, gridding-based algorithms outperform weighted back-projection algorithms by a small factor while SIRT is approximately ten times slower (depending on the number of iterations used).

2.5.6.7. 3D reconstruction of symmetric objects

Many objects imaged by EM have symmetries; the two types that are often met are helical (phage tails, *F*-actin, microtubules, myosin thick filaments, and bacterial pili and flagella) and point-group symmetries (many macromolecular assemblies, virus capsids). If the object has helical symmetry, it is convenient to use cylindrical coordinates and a dedicated reconstruction algorithm; particularly for the initial analysis of the data that has to be done in Fourier space. Diffraction from such structures with c periodicity and scattering density $\varphi(r, \psi, z)$ is defined by the Fourier–Bessel transform:

$$\begin{aligned}\Phi(R, \Psi, Z) &= \sum_{-\infty}^{+\infty} \exp\Big[in\Big(\Psi + \frac{\pi}{2}\Big)\Big] \int_0^\infty \int_0^{2\pi} \int_0^l \varphi(r, \psi, z) \\ &\quad \times J_n(2\pi r R) \exp[-i(n\psi + 2\pi z Z)] r \, \mathrm{d}r \, \mathrm{d}\psi \, \mathrm{d}z \\ &= \sum_n G_n(R, Z) \exp\Big[in\Big(\Psi + \frac{\pi}{2}\Big)\Big]. \qquad (2.5.6.46)\end{aligned}$$

The inverse transform has the form

$$\rho(r, \psi, z) = \sum_n \int g_n(r, Z) \exp(in\psi) \exp(2\pi i z Z) \, \mathrm{d}Z, \qquad (2.5.6.47)$$

so that g_n and G_n are the mutual Bessel transforms

$$G_n(R, Z) = \int_0^\infty g_n(r, Z) J_n(2\pi r R) 2\pi r \, \mathrm{d}r \qquad (2.5.6.48)$$

and

$$g_n(r, Z) = \int_0^\infty G_n(R, Z) J_n(2\pi r R) 2\pi R \, \mathrm{d}R. \qquad (2.5.6.49)$$

Owing to helical symmetry, (2.5.6.48) and (2.5.6.49) contain only those of the Bessel functions that satisfy the selection rule (Cochran *et al.*, 1952)

$$l = mp + (nq/N), \qquad (2.5.6.50)$$

where N, q and p are the helix symmetry parameters, $m = 0, \pm 1, \pm 2, \ldots$. Each layer l is practically determined by the single function J_n with the lowest n; the contributions of other functions are neglected. Thus, the Fourier transform of one projection of a helical structure, with an account of symmetry and phases, gives the 3D transform (2.5.6.49). However, biological specimens tend to be flexible and disordered, and exact helical symmetry is rarely observed. A possible approach to dealing with flexibility is to computationally straighten filaments (Egelman, 1986), but this has the potential for introducing artifacts. Another difficulty with helical analysis is that the indexing of a pattern can be ambiguous and the wrong symmetry can be chosen (Egelman & Stasiak, 1988). Further complications exist when the filament does not have a precisely defined helical symmetry, such as *F*-actin, which has a random variable twist (Egelman *et al.*, 1982). To address these problems, Egelman developed a real-space refinement method for the reconstruction of helical filaments that is capable of determining the helical symmetry of an unknown structure (Egelman, 2000). In this approach, the Fourier–Bessel reconstruction algorithm is replaced by a general reconstruction algorithm discussed in Section 2.5.6.6 with the real-space projections supplied as input. Thus, the symmetry is not enforced within the reconstruction algorithm; instead, it is determined and imposed subsequently by real-space averaging.

The presence of point-group symmetry in the structure means that any projection yields as many symmetry-related (but differently oriented) copies of itself as the number of symmetry operations in the group. For example, for three dimensions, each projection enters the reconstruction process at six different projection directions, while for icosahedral symmetry (I) the number is sixty! This high multiplicity makes it possible to write a reconstruction program that explicitly takes into account the symmetry and can perform the task much faster than generic algorithms. Such programs are usually part of dedicated software packages that were specifically designed for the determination of icosahedral structures (Fuller *et al.*, 1996; Lawton & Prasad, 1996; Liang *et al.*, 2002). These icosahedral reconstruction programs are heavily indebted to a set of FORTRAN routines written by R. A. Crowther at the Medical Research Council Laboratory of Molecular Biology (MRC LMB) and perform the 3D reconstruction using the Fourier–Bessel transform strategy outlined above [(2.5.6.46)–(2.5.6.49)] (Crowther, Amos *et al.*, 1970; Crowther & Amos, 1972).

The general reconstruction methods (algebraic, filtered back-projection, direct Fourier inversion) easily accommodate symmetries in the data. In major single-particle reconstruction packages, reconstruction programs are implemented such that the point-group symmetry is a parameter of the program. The symmetry operation is internally taken into account during calculation of the weighting function and it is also applied to the set of Eulerian angles assigned to each projection so the multiple copies are implicitly created and processed. This approach results in an extended time of calculations, but it is entirely general. In direct Fourier inversion algorithms the numerical inaccuracy of the symmetrization performed in Fourier space will result in nonsymmetric artifacts in real space. Thus, in *SPIDER* (Frank *et al.*, 1996) an additional real-space symmetrization is performed after the reconstruction is completed. Finally, it has to be noted that although it might be tempting to calculate a 3D reconstruction without enforcement of symmetry and to symmetrize the resulting structure, this approach is incorrect. This fact can be seen from the way weighting functions are constructed: weighting functions calculated with symmetries taken into account are not equal to the weighting function calculated for a unique set of projections and subsequently symmetrized.

2.5.7. Single-particle reconstruction

By P. A. Penczek

2.5.7.1. Formation of projection images in single-particle reconstruction

Cryo-electron microscopy (cryo-EM) in combination with the single-particle approach is a new method of structure determination for large macromolecular assemblies. Currently, resolution in the range 10 to 30 Å can be reached routinely, although in a number of pilot studies it has been possible to obtain structures at 4 to 8 Å. Theoretically, electron microscopy can yield data