Acta Crystallographica Section A
Foundations of Crystallography
ISSN 0108-7673

Received 20 September 2004
Accepted 16 December 2004

Reconstruction of complex single-particle images using charge-flipping algorithm

J. S. Wu* and J. C. H. Spence

Department of Physics and Astronomy, Arizona State University, Tempe, AZ 85287-1504, USA. Correspondence e-mail: jinsong.wu@asu.edu

An iterative algorithm is developed to retrieve the complex exit-face wavefunction for a two-dimensional projection of a nanoparticle from a measurement of the oversampled modulus of its Fourier transform in reciprocal space. The algorithm does not require the support (boundary) of the object to be known. A loose support for the complex object is gradually found using the Oszlányi–Sütő charge-flipping algorithm, and a compact support is then iteratively developed using a dynamic Gerchberg–Saxton–Fienup algorithm. At the same time, the complex object is reconstructed using this compact support. The algorithm applies to the reconstruction of complex images with any distribution of phase values from 0 to 2π. Modification of the algorithm by using real-value constraints for a complex object in the charge-flipping algorithm leads to faster reconstruction of the object whose phase value is smaller than $\pi/2$.

1. Introduction

Diffractive imaging promises to improve resolution, sensitivity and the range of wavelengths used in X-ray microscopy while eliminating the aberrations due to lenses in electron microscopy. While the atomicity constraint allows direct methods to provide a practical solution of the phase problem for atomic resolution data, the 'oversampling' of data at twice the Bragg rate has provided a similar practical method for non-periodic objects at any resolution. This diffractive imaging method for non-periodic objects retrieves the phases of non-periodic objects by iterative numerical techniques. The object and diffraction pattern are assumed to be related by simple Fourier transform, and no multiple scattering or Ewald sphere curvature effects are considered. Recently, the reconstruction of non-periodic structures by diffractive imaging has stimulated considerable interest as a contribution to new methods for solving protein structures that cannot be crystallized (Spence & Doak, 2004; Miao *et al.*, 1999; Miao & Sayre, 2000; De Caro *et al.*, 2002; He *et al.*, 2003). The first successful application of diffractive imaging to electron diffraction data was reported by Weierstall *et al.* (2001) and it has been used more recently to produce the first atomic resolution image of a single carbon nanotube (Zuo *et al.*, 2003). Developed from the Gerchberg–Saxton algorithm (Gerchberg & Saxton, 1972), the error-reduction (ER) and later hybrid input–output (HIO) algorithm (Fienup, 1982) are the two most widely used methods for phase recovery applied to single-particle Fraunhofer diffraction patterns (Fienup, 1982). The algorithm is based on three constraints – the known sign of the (real) charge density, the known Fourier moduli and the known boundary (support) of the object. Other algorithms have also been proposed, *e.g. Speden* (Hau-Riege *et al.*, 2004) and a novel density modification approach embedded in *SIR2002* (Carrozzini *et al.*, 2004). In this paper, we discuss a new phase recovery method that can be used to reconstruct a complex image of a single-particle object.

Phase retrieval for the more difficult case of complex objects is important in various areas such as electron microscopy, astronomy, crystallography and wavefront sensing. Multiple scattering of electrons or a spatial variation in the anomalous scattering of X-rays can both give rise to a complex near-field wavefunction, which we attempt to reconstruct as the image of the object. The loss of real and positive constraints in real space must then be compensated by use of other constraints during phase retrieval for complex objects. It has been shown that reconstruction of a complex object is usually possible if a sufficiently accurate support is available (Fienup, 1987), if a low-resolution image can be measured (Fienup & Kowalczyk, 1990) or by using the autocorrelation function of the object to iteratively improve the support estimate (Marchesini *et al.*, 2003). Recently, a simple and effective algorithm was developed by Oszlányi & Sütő (2004) to solve the phase problem for Bragg diffraction. By iteratively reversing (flipping) the sign of the charge density in regions where it lies below a small threshold, the algorithm has been successfully applied to simulated data (Oszlányi & Sütő, 2004) as well as real experimental data (Wu, Spence *et al.*, 2004). In the flipping algorithm, a 'dynamic' support (a support that may be different from one iteration to the next) is found in each iteration. Pixel values above the threshold define the support. Pixels within the support remain unchanged at each iteration, while all others outside the support have their sign reversed. The advantage of the charge-flipping algorithm is that it starts with measured moduli, but does not require either a known support or make the assumption of atomicity, as in direct methods. While it does not use any knowledge of scattering factors, it does require near atomic resolution data for crystals. By combining this flipping algorithm with the HIO algorithm, we have extended the application of the charge-flipping algorithm to non-periodic objects which are real and positive under kinematical scattering conditions (Wu, Weierstall *et al.*, 2004). In this paper, we explore the applicability of the charge-flipping algorithm to structure recovery for complex particles. Finding a compact support is a crucial step in this problem. We have developed a scheme that combines the charge-flipping algorithm with dynamic HIO (where the support is varied during the HIO iterations) and a real-space support extraction method (*e.g.* by setting a threshold) to address the phase retrieval problem for complex objects.

A structure can be uniquely reconstructed from a Fraunhofer diffraction pattern intensity distribution if certain constraints in real space are satisfied, in addition to the modulus constraint in the reciprocal space. For a complex object $f(\mathbf{x})$, $\exp(i\theta_c)f(\mathbf{x}-\mathbf{x}_0)$ and $\exp(i\theta_c)f^*(-\mathbf{x}-\mathbf{x}_0)$, where θ_c is a constant phase and f^* is the conjugate of f, have the same Fourier modulus, and we consider them equivalent images for the purpose of our reconstruction. In our structure reconstruction method, the charge-flipping algorithm is firstly used to find a loose support. After that, a combination of dynamic HIO and a dynamic ER algorithm is used to find a more compact support and retrieve the phases for the complex object.

2. Iterative phase retrieval algorithm

We assume that the modulus $|F(\mathbf{u})|$ has been measured, where $\mathbf{u}$ is a two-dimensional (2D) vector in reciprocal space. The support $s(\mathbf{x})$ for the complex object $f(\mathbf{x})$ is unknown. $\mathbf{x}$ is a 2D vector in real space. The aim of the algorithm is to reconstruct $f(\mathbf{x})$ from $|F(\mathbf{u})|$ by finding the compact support $s(\mathbf{x})$.

In each iteration between real and reciprocal space, the algorithm applies the modulus constraint. It consists of three steps: (i) Fourier transform of the current estimate of the object $g_n(\mathbf{x})$ to obtain $G_n(\mathbf{u})$; (ii) replace the current modulus $|G_n(\mathbf{u})|$ with the known modulus $|F(\mathbf{u})|$; (iii) inverse Fourier transform to yield $g_n'(\mathbf{x})$. This is

$$g_n'(\mathbf{x}) = \Im^{-1}\left\{\Im[g_n(\mathbf{x})]\frac{|F(\mathbf{u})|}{|\Im[g_n(\mathbf{x})]|}\right\}, \tag{1}$$

where $\Im$ and $\Im^{-1}$ represent Fourier and inverse Fourier transforms, respectively. The constraints in real space are differently defined for each iterative phase-retrieval algorithm. For the error-reduction algorithm, this is, at the nth iteration,

$$g_{n+1}(\mathbf{x}) = \begin{cases} g_n'(\mathbf{x}) & \mathbf{x}\in s(\mathbf{x}) \\ 0 & \mathbf{x}\notin s(\mathbf{x}). \end{cases} \tag{2}$$

The HIO algorithm was designed by adding a feedback function, which is (Fienup, 1982)

$$g_{n+1}(\mathbf{x}) = \begin{cases} g_n'(\mathbf{x}) & \mathbf{x}\in s(\mathbf{x}) \\ g_n(\mathbf{x}) - \beta g_n'(\mathbf{x}) & \mathbf{x}\notin s(\mathbf{x}), \end{cases} \tag{3}$$

where β is a constant between 0.5 and 1. $g_n'(\mathbf{x})$ is the result of applying the modulus constraint to $g_n(\mathbf{x})$ as defined in equation (1). The charge-flipping algorithm, which works with a dynamic support $s_d(\mathbf{x})$, can be written as

$$g_{n+1}(\mathbf{x}) = \begin{cases} g_n'(\mathbf{x}) & \mathbf{x}\in s_d(\mathbf{x}) \\ -g_n'(\mathbf{x}) & \mathbf{x}\notin s_d(\mathbf{x}). \end{cases} \tag{4}$$

The charge-flipping algorithm is mathematically similar to Fienup's output–output (not input–output) algorithm with $\beta = 2$, but works with a dynamic support (Wu, Weierstall *et al.*, 2004). Iterative application of the modulus constraints and real-space constraints enables phase retrieval.

Our scheme to reconstruct a complex object starts with the charge-flipping algorithm. Here we firstly discuss the method used to reconstruct a phase object whose phase distribution is within 0 to $\pi/2$. (This exceeds the range of the weak-phase-object or single-scattering approximation, which requires phases $\ll\pi/2$. In the following, we use the term phase object to represent an object whose phase variation lies within 0 and $\pi/2$.) The first estimate of the image $g_1(\mathbf{x})$ is generated by Fourier transform of a set of random phases $\varphi(\mathbf{u})$ combined with the observed modulus $|F(\mathbf{u})|$. The iterative Fourier transform algorithm for the nth iteration is as follows.

1. Determine the dynamic support $s_d(\mathbf{x})$. Sort $|g_k(\mathbf{x})|$ in descending order and define a ratio ϕ. (The threshold will consist of the fraction ϕ of image pixels with the largest values.) Select the m pixels with the highest positive values, such that $m = n\phi$, where n is the total number of pixels. These m pixels define the current support $s_d(\mathbf{x})$. For a complex object, the 'flipping' operation we use is defined as follows: we retain the highest peaks in the current modulus of the complex object in real space and reverse the sign of the real part of all pixels outside the current dynamic support:

$$g_{n+1}(\mathbf{x}) = \begin{cases} |g_n'(\mathbf{x})| & \mathbf{x}\in s_d(\mathbf{x}) \\ -\mathrm{real}\{g_n'(\mathbf{x})\} & \mathbf{x}\notin s_d(\mathbf{x}). \end{cases} \tag{5}$$

2. Apply modulus constraints using equation (1).
3. Go to step 1 with n replaced by $(n+1)$.

After running the charge-flipping algorithm for about 50 iterations, we switch to a combination of charge-flipping and the dynamic HIO algorithm, *i.e.* 10 iterations of HIO followed by 10 iterations of the flipping algorithm. 'Dynamic HIO' refers to an application of the HIO algorithm working with a dynamic support $s_d(\mathbf{x})$ that changes during iterations. The dynamic support $s_d(\mathbf{x})$ is calculated by selecting a certain percentage of the largest pixels from the modulus of the current estimate of the complex object, using the same procedure as discussed before:

$$s_d(\mathbf{x}) = \begin{cases} 1 & \text{if } |g_n(\mathbf{x})| \geq \delta \\ 0 & \text{if } |g_n(\mathbf{x})| < \delta. \end{cases} \tag{6}$$

In equation (6), if the modulus of the current complex object $g_n(\mathbf{x})$ is larger than a threshold δ determined by ϕ (so that the number of pixels greater than δ equals $n\phi$), then this pixel $\mathbf{x}$ is included in the dynamic support. The dynamic HIO is used because of its ability to pull the reconstruction process out of local minima. The initial support used for dynamic HIO is provided by the charge-flipping algorithm. We find that the peaks found by the charge-flipping algorithm do not exactly coincide with those in the modulus, real or imaginary parts of the complex object. However, they do provide a loose support estimate. Once this is obtained, we use a combination of dynamic HIO and the ER algorithm. From then on, the reconstructed object $g_n(\mathbf{x})$ is retained as a complex function in each iteration, although the dynamic support is found using the modulus of the complex object $|g_n(\mathbf{x})|$. Other known information (if any) about the support can also be used, *e.g.* the oversampling rate and the continuous connection property of the support (or a discrete support). For example, if an unknown support is continuous, we can apply this constraint by deleting all the other smaller discrete parts in the current support and filling in the holes inside the largest part of the support. If we know that a complex image has an oversampling rate of 4, we can set the fraction $\phi = 0.25$ when subtracting the support. Meanwhile, in order to derive a support, various image processing techniques can also be applied, such as image contrast enhancement and background reduction, before using the largest pixels in the modulus to form the support.

The reconstruction of a particle whose phase variation ranges from 0 to 2π is more difficult than that of a phase object and involves the problem of phase unwrapping. The estimation of the percentage parameter ϕ must then be much closer to its optimal value. For example, in reconstruction of a phase object, an error of $\pm 20\%$ in the ϕ use is tolerable. However, in the reconstruction of a general complex object, the error can only be around 5%. Meanwhile, the equation (5) used in the above procedure is replaced by equation (4). We also found that it was helpful to use some dynamic HIO iterations alone to let the dynamic support approach its correct form, based on the one found by the charge-flipping algorithm. The dynamic support $s_d^{\rm HIO}$ for these HIO iterations was defined as

$$s_d^{\rm HIO}(\mathbf{x}) = \begin{cases} 1 & \text{if } |g_n(\mathbf{x}) \otimes \text{Circle}(\mathbf{x})| \geq \delta \\ 0 & \text{if } |g_n(\mathbf{x}) \otimes \text{Circle}(\mathbf{x})| < \delta, \end{cases} \tag{7}$$

where $\otimes$ represents convolution and Circle($\mathbf{x}$) is a circle function with radius R of 2–4 pixels:

$$\text{Circle}(\mathbf{x}) = \begin{cases} 1 & |\mathbf{x}| \leq R \\ 0 & |\mathbf{x}| > R. \end{cases}$$

Once a good estimation of the percent parameter ϕ is found and the parameters of the support can be reasonably constrained, dynamic HIO assists the dynamic support to approach to the correct form. The modulus of the current reconstruction $|g(\mathbf{x})|$ is filtered using a Wiener filter to improve the signal/noise ratio.

The progress of the iterations can be followed using a residual R, calculated using the modulus of the current estimate of the object and the observed moduli:

$$R = \sum ||G_k(\mathbf{u})| - |F(\mathbf{u})|| \Big/ \sum |F(\mathbf{u})|. \tag{8}$$

A sharp drop in a plot of R *versus* iteration number is a clear sign of convergence of the algorithm. If the current dynamic support is not correct, *i.e.* $s_d(\mathbf{x}) \neq s(\mathbf{x})$, the R values increase in the dynamic HIO iterations. A value of R approaching zero is needed for a faithful reconstruction of a complex object. We note that this is a different error metric than that normally used for HIO analysis, which measures the density outside the support. The R factor above is similar to that used in crystallography and takes no account of phases. It is therefore not unique in the absence of other constraints.

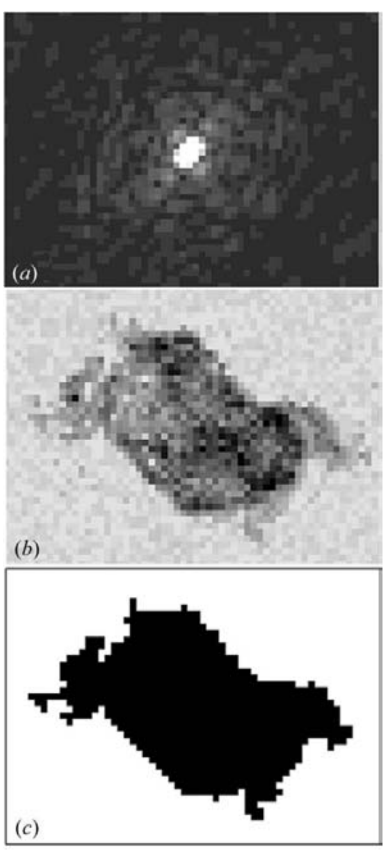

Figure 1
(*a*) Simulated diffraction pattern of the complex object used for phase recovery test. (*b*) The reconstructed image (high value pixels are modulus and low pixels flipped are the real part of the complex object) after using a combination of charge-flipping and the dynamic HIO algorithm. (*c*) The support extracted from (*b*) by selection of the high-valued connected pixels.

3. Results of computational trials

Fig. 1(*a*) shows a simulated diffraction pattern for a complex object (the oversampling rate is 4). Using the charge-flipping algorithm, the highest peaks in the modulus of the reconstructed complex object are used to derive a loose support. The reconstructed image at the 160th iteration is shown in Fig. 1(*b*). The highest peaks are the modulus while the low pixels are the real part of the complex object. The 160 iterations include 60 charge-flipping iterations and 100 combined charge-flipping and dynamic HIO iterations. Since the initial phases were random numbers and we do not have a support domain at the beginning of the program, the origin cannot be fixed and the structure solved by the algorithm has a randomly positioned origin. The loose support extracted from Fig. 1(*b*) is shown in Fig. 1(*c*). With this loose support, a combination of dynamic HIO and ER iterations were employed. The method for finding the dynamic support is the same as that used in the charge-flipping algorithm, which depends on 'peaks' emerging in the modulus of the reconstructed object. At this stage, we keep the support unchanged in the ER iterations, although it is still not exactly equal to the compact support. The dynamic support only develops within the dynamic HIO iterations. Meanwhile, the current reconstructed object is retained in complex form. In about 80% of our tests, this application of the dynamic HIO and ER algorithm led to the correct compact support, and thus the complex object could be correctly reproduced. We find that some errors in the dynamic support can be corrected by investigating the phase of the reconstructed complex object. Usually they are situated at the edge of the dynamic support. Fig. 2(*a*) shows the reconstructed modulus of the complex object in real space after 360 iterations. It reproduces the original modulus. Since the support is calculated from the current modulus by selecting out the highest peaks according to the fractional parameter ϕ, it is easy to introduce errors of a few pixels at the edge of the support. The support derived from Fig. 2(*a*) is shown in Fig. 2(*c*). If we compare Fig. 2(*c*) to Fig. 2(*a*), we find that the support is wrong in one pixel at the place indicated by an arrow. Meanwhile, if we investigate the phase of the reconstructed object shown in Fig. 2(*b*), there is a sharp phase change in the wrong pixel as indicated by an arrow. This is a useful sign for finding errors in the current support, and provides a way to correct the support.

The variation in R factor during iterations of the algorithm is shown in Fig. 3, where the algorithms used are indicated. This consists of 60 charge-flipping iterations, followed by 100 iterations of the combined charge-flipping and dynamic HIO iterations, ending with 200 iterations of the combined dynamic HIO and ER algorithms. We find that in most of the dynamic HIO iterations R increases. This is because the dynamic support is not correct. When the support is correctly reconstructed, R for the HIO also decreases. The final R in our current test approaches almost zero.

The same method works well also for a complex object having a support with many separated parts. Fig. 4(*a*) shows a reconstructed image of such a complex object at the 100th iteration, soon after the application of a combination of the charge-flipping algorithm and the dynamic HIO algorithm. The residual R *versus* iteration number is shown in Fig. 4(*b*). The reconstructed modulus and phase of the complex object are shown in Figs. 4(*c*) and 4(*d*), respectively. They reproduce the starting object. The minimum phase is zero while the maximum phase is $\pi/2$ (the phase lies in the range $0 < \theta < \pi/2$) in the above two tests.

Figs. 5(*a*) and 5(*b*) show the modulus and phase of a randomly generated complex particle, whose phase ranges

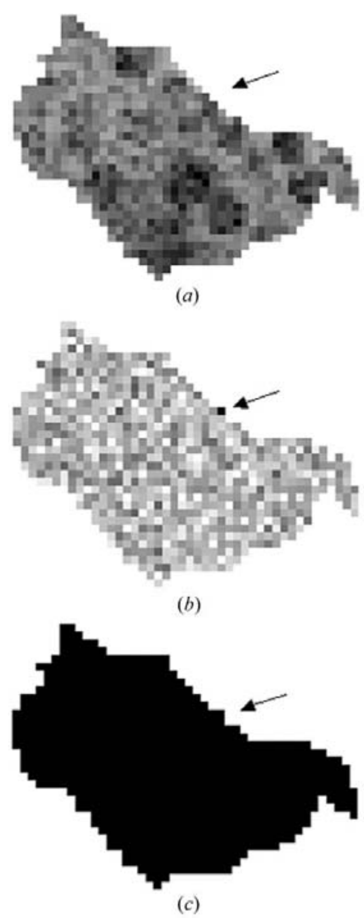

Figure 2
(*a*) The reconstructed modulus and (*b*) phase of the complex object. It reproduces the original object. (*c*) The reconstructed support from the combined use of a dynamic HIO and error-reduction algorithm. Every five times, high modulus and continuous pixels are selected out to form the dynamic support.

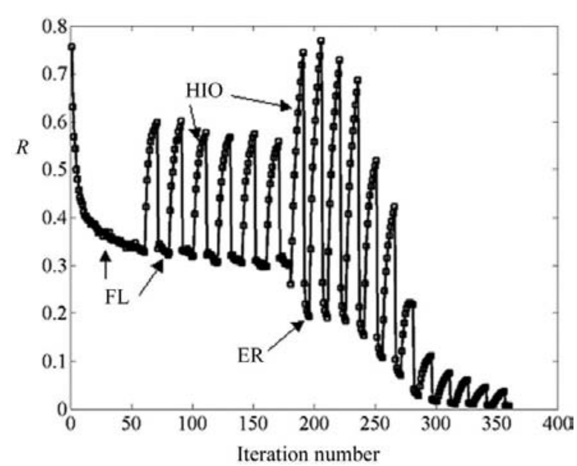

Figure 3
The residual R factors as a function of iteration numbers. The algorithm can be divided into three stages. It starts with merely the charge-flipping (FL) algorithm, continues by a combination of charge flipping and HIO and finishes with a combined use of HIO and the error-reduction (ER) algorithm. All these work with a dynamic support, which gradually approaches the correct support.

from 0 to 2π. The ϕ we used is 0.15, which is larger than the optimal value of 0.132 (the oversampling rate is thus $1/0.132 = 7.57$). An exactly compact support could not be found by using a large ϕ. However, the reconstructed object as shown in Figs. 5(c) and 5(d) reproduced the original particle inside the compact support. Fig. 5(c) shows the reconstructed modulus at iteration 640 and it is the same as Fig. 5(a). The reconstructed phase is shown in Fig. 5(d). Those phases inside the compact support in Fig. 5(d) are related to those shown in Fig. 5(b) by a constant of $\theta_c = 1.52$. The reconstructed particle is thus related to the original one by $\exp(i\theta_c)g(\mathbf{x} - \mathbf{x}_0)$ to $g(\mathbf{x})$, where $\theta_c = 1.52$. Since the image reconstructed by the charge-flipping algorithm has a random origin, for each reconstruction $\mathbf{x}_0$ has a random value. Fig. 5(e) shows the reconstructed modulus at iteration 140 when the combination of the charge-flipping algorithm and the dynamic HIO is finished. Compared to that shown in Fig. 1(b), we find that the reconstructed modulus for a general complex particle is much more error prone at this stage. Fig. 5(f) shows the support found at iteration 640. The support is bigger than the correct one, which is also strictly a loose support. We have thus shown that the correct complex particle could be reconstructed, although the parameter ϕ is not known exactly *a priori*. The variation of the R factor during iteration of the algorithm is shown in Fig. 6. The reconstruction consists of 60 charge-flipping iterations, 80 iterations of the combined charge-flipping and dynamic HIO iterations, followed by 180 dynamic HIO with dynamic support determined by equation (7), and ending with 320 iterations of the combined dynamic HIO and ER algorithms. The final R in our current test is 0.0002.

The oversampling rate as well as the noise can influence greatly the phase recovery process. We found that, when the oversampling rate was larger than 3.12, a faithful reconstruction of the example shown in Fig. 5 could be achieved in the absence of noise. We also add noise to the simulated moduli using noise = (signal/SNR) $\times$ random, where SNR is the signal-to-noise ratio and random is a random number from

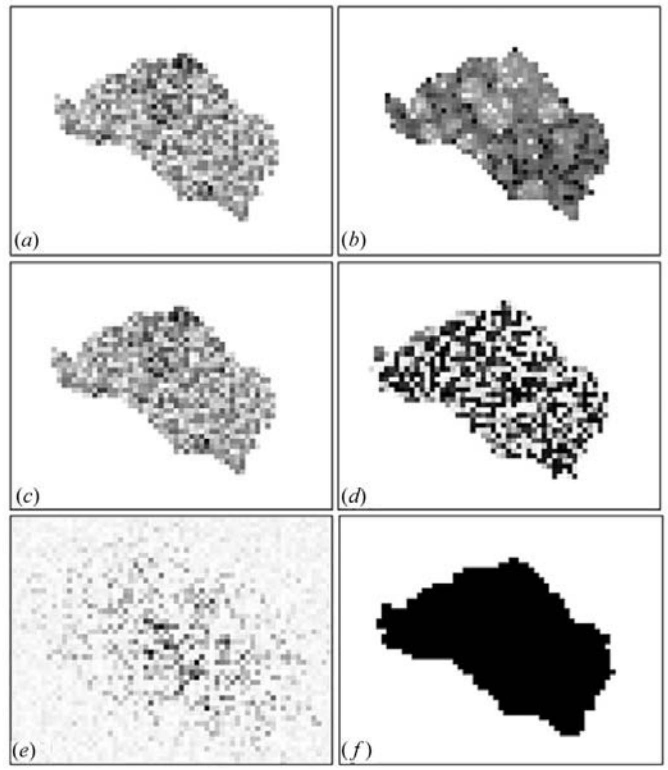

Figure 5
(a) Modulus and (b) phase of a complex object with phase range from 0 to 2π. (c) The reconstructed modulus and (d) phase of the complex object at iteration 660. (e) The reconstructed modulus at iteration 140 when only the flipping and dynamic HIO algorithm were used. (f) The reconstructed loose support at iteration 640 from the combination use of dynamic HIO and error-reduction algorithm.

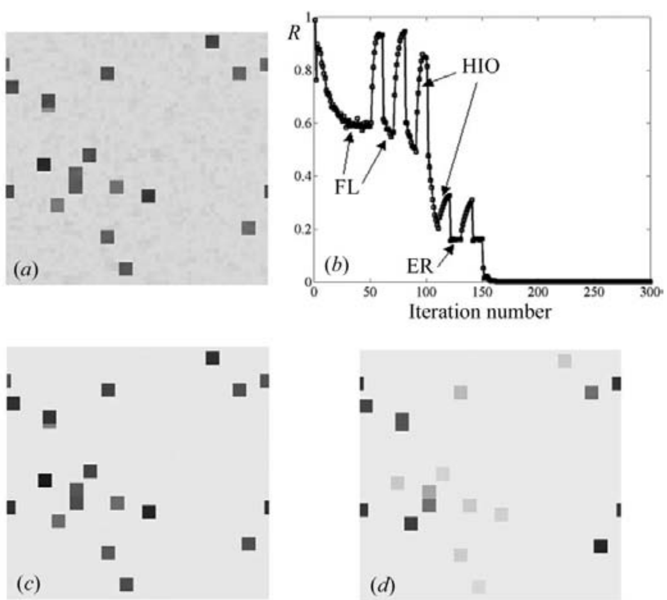

Figure 4
(a) Reconstructed modulus (high peaks) using the charge-flipping algorithm, on which basis a support is subtracted. (b) The residual R factors as a function of iteration numbers. (c) The final reconstructed modulus and (d) phases of the discrete complex object.

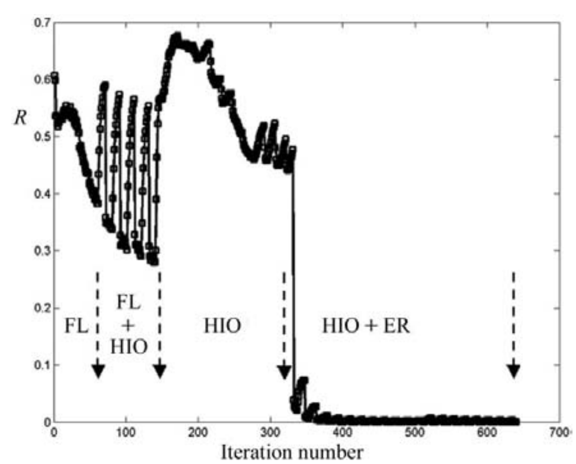

Figure 6
The residual R factors as a function of iteration number. The algorithm can be divided into four stages. It starts with merely the charge-flipping algorithm, continues by a combination of charge flipping and HIO, dynamic HIO only, and finishes with a combined usage of the HIO and ER algorithms. All those work with a dynamic support.

-0.5 to 0.5 (Miao *et al.*, 1998). For the reconstruction shown in Fig. 5, most of the features could be recovered when the SNR was set to a value larger than 10.

4. Discussion

Unlike the conventional HIO algorithm, the flipping algorithm does not require the support in real space to be known. On the contrary, the algorithm finds the support. This combination of dynamic HIO with the charge-flipping algorithm can solve the stagnation problem. We found that sometimes the use of the charge-flipping algorithm alone stagnates in local minima. The extraction of the compact support is the critical part of the current algorithm. Once the correct compact support is found, a combination of the HIO and ER algorithm is shown to correctly reconstruct a complex object. We have also shown that, wherever there is an error in the current estimate of the support, there is a sharp change in the phase of the reconstructed complex object. This rule, combined with other known constraints about the support, may be used to find the compact support. The reconstruction process is thus aided by human intervention, when working with different objects. The support is extracted using the highest pixels in the modulus of the current reconstructed complex object. In our tests, we found that it was quicker to find the correct support by using the peaks that emerged in the modulus, rather than by using the peaks in either the real part or the imaginary part of the complex function.

We used different procedures to solve complex phase objects ($0 < \phi < \pi/2$) and those with stronger phase variation of 0 to 2π. Fig. 7(*a*) shows the moduli of the Fourier transform of a positive object. A phase object is formed by using the positive object as the modulus while assigning random phases in the range from 0 to $\pi/2$. The moduli of its Fourier transform is shown in Fig. 7(*b*). A fully dynamic scattered object is formed by using the same modulus with random phases from 0 to 2π, and the moduli of its Fourier transform is shown in Fig. 7(*c*). Figs. 7(*d*), 7(*e*) and 7(*f*) show the corresponding autocorrelation functions calculated using Figs. 7(*a*), 7(*b*) and 7(*c*). The similarity of the diffraction pattern of the phase object to that of its real object is evident. Their autocorrelation functions show an analogous similarity. This may explain the fact that making $g(\mathbf{x})$ real is helpful in finding the support in the reconstruction of a phase object. Miao *et al.* (1998) have used a positivity constraint on the imaginary part to reconstruct a phase object with *a priori* known loose support. In the present algorithm, we found that an accurate estimation of ϕ was necessary for solving a complex particle with phase variation of 0 to 2π. Meanwhile, repeated application of dynamic HIO alone with reasonable constraints on the support (if known) helped to find a better loose support close to the compact support.

It has been found that an upper bound for the support can be estimated from the autocorrelation function of the object. The recently developed *Shrinkwrap* algorithm for complex-valued non-periodic objects makes use of this fact by making an initial estimate of the support shape based on the autocorrelation function, and iteratively improving this (Marchesini *et al.*, 2003). Unlike the *Shrinkwrap* algorithm, the dynamic support described here may either grow or shrink, so that the search is more flexible.

5. Conclusions

By combining the charge-flipping algorithm, the dynamic hybrid input–output algorithm and an error reduction algor-

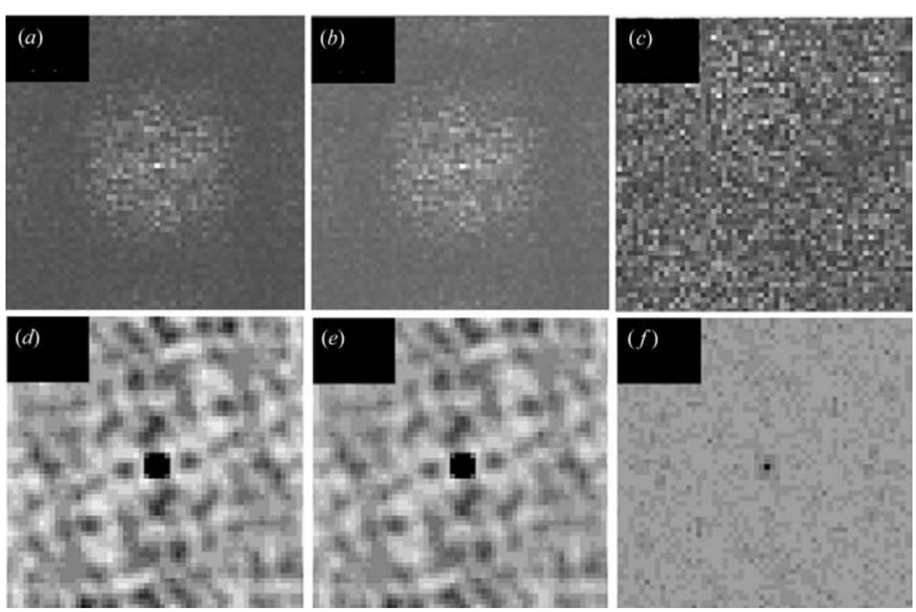

Figure 7
(*a*) Moduli of the Fourier transforms of a real and positive object. (*b*) Moduli of a complex object having the same modulus as (*a*) and random phase varying from 0 to $\pi/2$. (*c*) Moduli of a complex object having the same modulus as (*a*) and random phase varying from 0 to 2π. (*d*), (*e*) and (*f*) are autocorrelation functions calculated by (*a*), (*b*) and (*c*), respectively. Note that the similarity between a real object and a phase object is evident.

ithm, a method is designed to retrieve phases for a complex object with phase variation lying in the range $0 < \theta < 2\pi$. The method does not need a known support to be supplied. Errors in the support estimate can be corrected for by examining the phase of the current reconstructed complex object. The similarity of the autocorrelation function between a complex object obeying the phase-object approximation (where the phase variation is between 0 and $\pi/2$) and a real object enables the application of real constraints to the charge-flipping algorithm for efficient reconstruction.

This work is supported by ARO award DAAD190010500.

References

Carrozzini, B., Cascarano, G. L., De Caro, L., Giacovazzo, C., Marchesini, S., Chapman, H., He, H., Howells, M., Wu, J. S., Weierstall, U. & Spence, J. C. H. (2004). *Acta Cryst.* A**60**, 331–338.

De Caro, L., Giacovazzo, C. & Siliqi, D. (2002). *Acta Cryst.* A**58**, 415–423.

Fienup, J. R. (1982). *Appl. Optics*, **21**, 2758–2769.

Fienup, J. R. (1987). *J. Opt. Soc. Am.* **4**, 118–123.

Fienup, J. R. & Kowalczyk, A. M. (1990). *J. Opt. Soc. Am.* **7**, 450–458.

Gerchberg, R. W. & Saxton, W. O. (1972). *Optik (Stuttgart)*, **35**, 237–246.

Hau-Riege, S. P., Szoke, H., Chapman, H. N., Szoke, A., Marchesini, S., Noy, A., He, H., Howells, M., Uwe, W. & Spence, J. C. H. (2004). *Acta Cryst.* A**60**, 294–305.

He, H., Marchesini, S., Howells, M., Weierstall, U., Hembree, G. & Spence, J. C. H. (2003). *Acta Cryst.* A**59**, 143–152.

Marchesini, S., He, H., Chapman, H., Hau-Riege, S., Noy, A., Howells, M., Weierstall, U. & Spence, J. C. H. (2003). *Phys Rev. B*, **68**, 140101(R).

Miao, J., Charalambous, C., Kirz, J. & Sayre, D. (1999). *Nature (London)*, **400**, 342–344.

Miao, J. & Sayre, D. (2000). *Acta Cryst.* A**56**, 596–605.

Miao, J., Sayre, D. & Chapman, H. N. (1998). *J. Opt. Soc. Am.* A**15**, 1662–1669.

Oszlányi, G. & Sütő, A. (2004) *Acta Cryst.* A**60**, 134–141.

Spence, J. C. H. & Doak, B. (2004). *Phys. Rev. Lett.* **92**, 198102.

Weierstall, U., Chen, Q., Spence, J., Howells, M., Isaacson, M. & Panepucci, R. (2001). *Ultramicroscopy*, **90**, 171–195.

Wu, J. S., Spence, J. C. H., O'Keeffe, M. & Groy, T. (2004). *Acta Cryst.* A**60**, 326–330.

Wu, J. S., Weierstall, U., Spence, J. C. H. & Koch, C. T. (2004). *Opt. Lett.* **29**, 2737–2739.

Zuo, J. M., Vartanyants, I., Gao, M., Zhang, R. & Nagahara, L. A. (2003). *Science*, **300**, 1419–1421.