# Reconstruction and synthesis applications of an iterative algorithm

**J. R. Fienup**
Environmental Research Institute of Michigan
P.O. Box 8618
Ann Arbor, Michigan 48107

**Abstract.** This paper reviews the Gerchberg-Saxton algorithm and variations thereof that have been used to solve a number of difficult reconstruction and synthesis problems in optics and related fields. It can be used on any problem in which only partial information (including both measurements and constraints) of the wavefront or signal is available in one domain and other partial information is available in another domain (usually the Fourier domain). The algorithm combines the information in both domains to arrive at the complete description of the wavefront or signal. Various applications are reviewed, including synthesis of Fourier transform pairs having desirable properties as well as reconstruction problems. Variations of the algorithm and the convergence properties of the algorithm are discussed.

## 1. INTRODUCTION

There exist many problems that are very difficult to solve in astronomy, x-ray crystallography, electron microscopy, spectroscopy, wavefront sensing, holography, particle scattering, superresolution, radar signal and antenna synthesis, filter design, and other disciplines that share an important feature. These are problems that involve the reconstruction or synthesis of a wavefront (or an object or a signal, etc.) when partial information or constraints exists in each of two different domains. The second domain is usually the Fourier transform domain. This paper describes a method of combining all the available information in the two domains to arrive at a complete description, thereby solving the problems.

The problems fall into two general categories: (1) reconstruct the entire information about a function (an image, wavefront, signal, etc.) when only partial information is available in each of two domains; and (2) synthesize a (Fourier) transform pair having desirable properties in both domains. A reconstruction problem arises when only partial information is measured in one domain, and in the other domain either partial information is measured or certain constraints are known *a priori*. The information available in any one domain is insufficient to reconstruct the function or its transform. A synthesis problem typically arises when one wants the transform of a function to have certain desirable properties (such as uniform spectrum, low sidelobes, etc.) while the function itself must satisfy certain constraints or have certain desirable properties. Because arbitrary sets of properties and constraints can be contradictory, there may not exist a transform pair that is completely desirable and satisfies all the constraints. Nevertheless, one seeks a transform pair that comes as close as possible to having the desirable properties and satisfying the constraints in both domains.

Both the reconstruction and the synthesis problems can be expressed as follows, if the meaning of the word "constraints" is broadened to include any kind of measured data, desirable properties, or *a priori* conditions:

> Given a set of constraints placed on a function and another set of constraints placed on its transform, find a transform pair (i.e., a function and its transform) that satisfies both sets of constraints.

Once a solution is found to such a problem, the question often remains: is the solution unique? For synthesis problems, the uniqueness is usually unimportant—one is satisfied with any solution that satisfies all the constraints; often a more important problem is whether there exists *any* solution that satisfies what may be arbitrary and conflicting constraints. For reconstruction problems, the uniqueness properties of the solution are of central importance. If many different functions satisfying the constraints could give rise to the same measured data, then a solution that is found could not be guaranteed to be the correct solution. The question of uniqueness must be studied for each problem. Fortunately, as will be described later, for some important reconstruction problems the solution usually is unique.

An effective approach to solving the large class of problems described above is the use of iterative algorithms related to the Gerchberg-Saxton algorithm.[1] The algorithms involve the iterative transformation back and forth between the two domains, with the known constraints applied repetitively in each domain.

The basic algorithm is presented in Sec. 2. A number of different applications having different types of constraints are described, and examples are shown in Sec. 3. In Sec. 4 the convergence properties of the algorithm are discussed, and improved versions of the algorithm are reviewed. A brief summary and comments are included in Sec. 5.

## 2. THE BASIC ITERATIVE ALGORITHM

The first published account of the iterative algorithm was its use by Gerchberg and Saxton[1] to solve the electron microscopy problem. For this problem both the modulus (magnitude) of a complex-

valued image and the modulus of its Fourier transform are measured, and the goal is to reconstruct the phase in both domains. Apparently unknown to Gerchberg and Saxton, the method was invented somewhat earlier by Hirsch, Jordan, and Lesem[2] to solve a synthesis problem for computer-generated holograms that has a similar set of constraints. (This will be described later in more detail.) The method was again reinvented for a similar problem in computer holography by Gallagher and Liu.[3] The fact that the algorithm was invented repeatedly testifies to its simplicity and effectiveness.

### 2.1. Gerchberg-Saxton algorithm

In what immediately follows, the iterative algorithm is described in terms of its application to the electron microscopy reconstruction problem. An excellent treatment of the electron microscopy phase problem and its solution by this and other methods can be found in Ref. 4. Later it is shown how to apply the same principles to a large class of problems.

Suppose that the electron wave function in an image plane is described by the two-dimensional (2-D) complex-valued function

$$f(x) = |f(x)|\, e^{i\psi(x)} . \tag{1}$$

Its Fourier transform, the wave function in a far-field diffraction plane, is given by

$$F(u) = |F(u)|\, e^{i\theta(u)} = \mathcal{F}[f(x)] = \int_{-\infty}^{\infty} f(x) e^{-i2\pi u \cdot x}\, dx , \tag{2}$$

where x and u are the vector coordinates in the spatial (image) domain and the spatial frequency (far-field diffraction) domain, respectively. The notation used throughout this paper is that functions represented by capital letters are the Fourier transforms of the functions represented by the corresponding lower-case letters. It is assumed that the intensity spatial distributions are measured in each domain, but the phase information is lost. Therefore, one wishes to reconstruct $\psi(x)$ and $\theta(x)$ from $|f(x)|$ and $|F(u)|$.

The iterative algorithm for solving this problem is depicted in Fig. 1. One iteration (the $k^{th}$ iteration) of the algorithm proceeds as follows. A trial solution for the wave function (an estimate of the wave function), $g_k(x)$, is Fourier transformed yielding

$$G_k(u) = |G_k(u)| \exp[i\phi_k(u)] = \mathcal{F}[g_k(x)] . \tag{3}$$

Then a new Fourier-domain function, $G_k'(u)$, is formed by replacing the computed Fourier modulus by the measured Fourier modulus, $|F(u)|$, and keeping the computed phase:

$$G_k'(u) = |F(u)| \exp[i\phi_k(u)] . \tag{4}$$

The resulting $G_k'(u)$, which is in agreement with all the known measurements and constraints in the Fourier domain, is inverse Fourier transformed, yielding the wave function $g_k'(x)$. The iteration is completed by forming a new estimate for the wave function, $g_{k+1}(x)$, which is obtained by replacing the computed modulus of $g_k'(x)$ with the measured modulus $|f(x)|$, and keeping the computed phase.

The algorithm consists of no more than enforcing what information is available on the wave function, Fourier transforming, imposing what information is available on the wave function's Fourier transform, inverse transforming, and repeating these simple operations for a number of iterations. What makes the algorithm practical is the existence of a fast Fourier transform[5] (FFT), so that the number of computations per iteration goes only as NlogN, where N is the number of samples of the function computed. This compares very favorably with some other iterative methods, such as Newton-Raphson,[4] for which the number of computations per iteration goes as $N^3$.

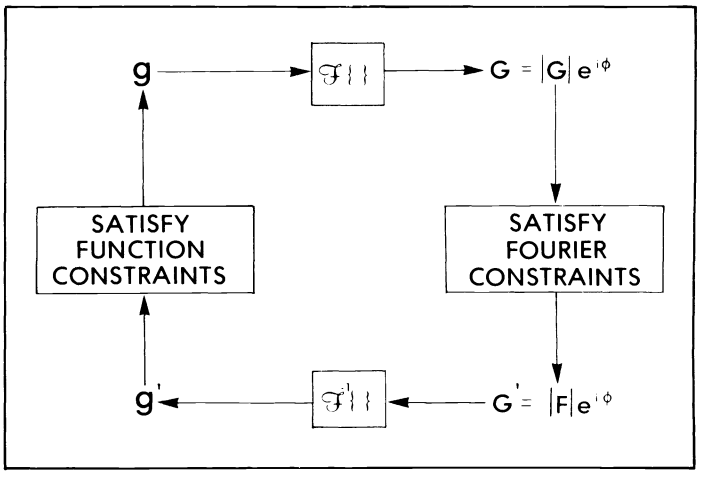


**Fig. 1. Block diagram of the iterative error-reduction algorithm.**

A measure of the progress of the iterations, and a criterion by which one can determine when a solution has been found, is the normalized mean-squared error, which is defined in the Fourier domain by

$$E_F^2 = \frac{\int_{-\infty}^{\infty} [|G_k(u)| - |F(u)|]^2\, du}{\int_{-\infty}^{\infty} |F(u)|^2\, du} \tag{5}$$

or in the image domain by

$$E_0^2 = \frac{\int_{-\infty}^{\infty} [|g_k'(x)| - |f(x)|]^2\, dx}{\int_{-\infty}^{\infty} |f(x)|^2\, dx} . \tag{6}$$

It has been shown that the algorithm converges in the sense that the mean-squared error can only decrease at each iteration.[1,4,6] The issue of convergence will be discussed in greater detail in Sec. 4.

### 2.2. Error-reduction iterative algorithm

It is now known that with slight modifications this same algorithm can be applied to many different problems having a variety of available constraints or measurements.[7] Let the function f(x) represent a wavefront, an object, a signal, an antenna array, a spectral density function, an electron density function, etc., where x is an N-dimensional vector (spatial, angular, time, etc.) coordinate. Depending on the problem, f(x) may be complex valued or real valued and, if real, may or may not be nonnegative. Its Fourier transform, F(u), is given by Eq. (2) and is complex valued for most problems. The N-dimensional vector u is a (spatial, angular, time, etc.) frequency coordinate. One can instead consider another transformation of f(x), such as the Fresnel transform, which has been used for more than one problem.[2,8,9] For simplicity of discussion, the Fourier transform will be assumed, but the reader should keep in mind that what is said also applies to a number of other transformations as well (although the method becomes less attrac-

tive if a fast transform algorithm is not available).

With only slight modifications, the Gerchberg-Saxton algorithm can be used to solve the wide class of problems described in Sec. 1. Referring again to the block diagram of the algorithm in Fig. 1, all that is required is to impose constraints in each domain that are pertinent to the problem of interest. At the $k^{th}$ iteration, $g_k(x)$, an estimate of f(x), is Fourier transformed, yielding $G_k(u)$, which is given by Eq. (3). Then a new Fourier-domain function $G'_k(u)$ is formed from $G_k(u)$ by making the smallest possible changes in $G_k(u)$ that allow it to satisfy the Fourier-domain constraints. For example, if the Fourier-domain constraint is that the Fourier modulus equals $|F(u)|$ over some region of the Fourier domain, then $|F(u)|$ is substituted for $|G_k(u)|$ in that region. The new Fourier-domain function $G'_k(u)$, which satisfies the Fourier-domain constraints, is inverse Fourier transformed to yield $g'_k(x)$. To complete one iteration, a new estimate $g_{k+1}(x)$ is formed from $g'_k(x)$ by making the smallest possible changes in $g'_k(x)$ that allow it to satisfy the function-domain constraints. One example is that if the function is complex valued and it is constrained to have a modulus equal to $|f(x)|$ over some region of space, then $|f(x)|$ is substituted for $|g'_k(x)|$ in that region. A special case of this is when the function is to be zero outside a certain interval (the Fourier function is bandlimited). Another example is that if the function is constrained to be nonnegative, then $g_{k+1}(x)$ is set equal to $g'_k(x)$ for those x where $g'_k(x) \geq 0$, and $g_{k+1}(x)$ is set equal to zero for those x where $g'_k(x) < 0$. In summary, one transforms back and forth between the two domains, forcing the function to satisfy the constraints in each domain.

For reconstruction problems, whatever characteristics of the actual F(u) and f(x) that are measured or are known *a priori* are imposed on $G_k(u)$ and $g'_k(x)$, respectively. For synthesis problems, one imposes on $G_k(u)$ and $g'_k(x)$ whatever characteristics one might desire F(u) and f(x), respectively, to have. Once the constraints are defined, the algorithm proceeds the same for synthesis problems as for reconstruction problems. In fact, there are some synthesis problems that are mathematically indistinguishable from some reconstruction problems, and they are handled identically by the algorithm.

The first iteration of the algorithm can be started in a number of ways, for example, by setting $g_1(x)$ or $\phi_1(x)$ equal to an array of random numbers. The iterations continue until a Fourier transform pair is found that satisfies all the constraints in both domains to within the desired accuracy (or, if convergence is too slow, until one loses interest or the money runs out). The mean-squared error can generally be defined in the Fourier domain by

$$E_F^2 = \frac{\int_{-\infty}^{\infty} |G_k(u) - G'_k(u)|^2 \, du}{\int_{-\infty}^{\infty} |G'_k(u)|^2 \, du} \tag{7}$$

or in the function domain by

$$E_0^2 = \frac{\int_{-\infty}^{\infty} |g_{k+1}(x) - g'_k(x)|^2 \, dx}{\int_{-\infty}^{\infty} |g'_k(x)|^2 \, dx} . \tag{8}$$

In each of these two expressions, the integrand in the numerator is the squared modulus of the amount by which the computed function violates the constraints in that domain. It is easily seen that these expressions reduce to Eqs. (5) and (6), respectively, for the electron microscopy problem.

Just as in the electron microscopy problem, for problems having other sets of constraints it will be shown in Sec. 4 that the algorithm converges, that is, the error decreases at each successive iteration. The algorithm depicted in Fig. 1 may be referred to as the "error-reduction" algorithm for that reason, as well as to distinguish it from algorithms described in Sec. 4 that are related to it but converge faster. Typically, the error is reduced very rapidly for the first few iterations of the error-reduction algorithm, but more slowly for later iterations. For some applications, the error-reduction algorithm has been very successful in finding solutions using a reasonable number of iterations. However, for some other applications, the mean-squared error decreases extremely slowly with each iteration, and an impractically large number of iterations is required. The improved algorithms described in Sec. 4 do much to alleviate this problem.

### 2.3. Alternative descriptions of the algorithm

Once a solution (i.e., a Fourier transform pair satisfying all the constraints in both domains) is found, the error-reduction algorithm ceases to make changes to the estimate, and the algorithm locks on to the solution. The operations of enforcing the constraints in each domain would then leave the function estimate and its Fourier transform unaltered, since they already satisfy the constraints. Now let us define the operation S[g(x)] as the successive Fourier transformation of g(x), followed by the imposition of the Fourier domain constraints, followed by inverse Fourier transformation, followed by imposition of the object domain constraints. That is, the operation S is just the performance of one iteration of the error-reduction algorithm, and

$$g_{k+1}(x) = S[g_k(x)] . \tag{9}$$

From the discussion above, it is evident that any solution f(x) must satisfy the relation

$$f(x) = S[f(x)] . \tag{10}$$

When presented in this form, it is seen that the error-reduction algorithm is a particular implementation of the method of successive approximations.[10]

The method of successive approximations can be more easily understood from the following simple example. Suppose one wishes to solve the following equation for y:

$$4y^4 - 4y + 1 = 0 . \tag{11}$$

Based on the relation $y = y^4 + 1/4$, one could write

$$y_{k+1} = S_1(y_k) = y_k^4 + 1/4 . \tag{12}$$

Using the method of successive approximations to find the solution, one would pick an initial estimate, say $y_0 = 0.1$, and employing Eq. (12) compute $y_1 = 0.2501$, $y_2 = 0.2539$, etc., and rapidly converge to the solution $y' = 0.2541737 \ldots$. However, it converges to y′ only for $y_0 < y'' = 0.8967902 \ldots$. For $y_0 > y''$, Eq. (12) diverges; and for $y_0 = y''$, it stays at y″, the second solution. On the other hand, one could just as logically have chosen

$$y_{k+1} = S_2(y_k) = (y_k - 1/4)^{1/4} . \tag{13}$$

This second form converges to the second solution y″ for $y_0 > y'$, diverges for $y_0 < y'$, and stays at y′ for $y_0 = y'$. Figure 2, a graphical representation of Eq. (12), shows the two solutions, y′ and y″. The irregular staircase between the two curves y and $y^4$ +

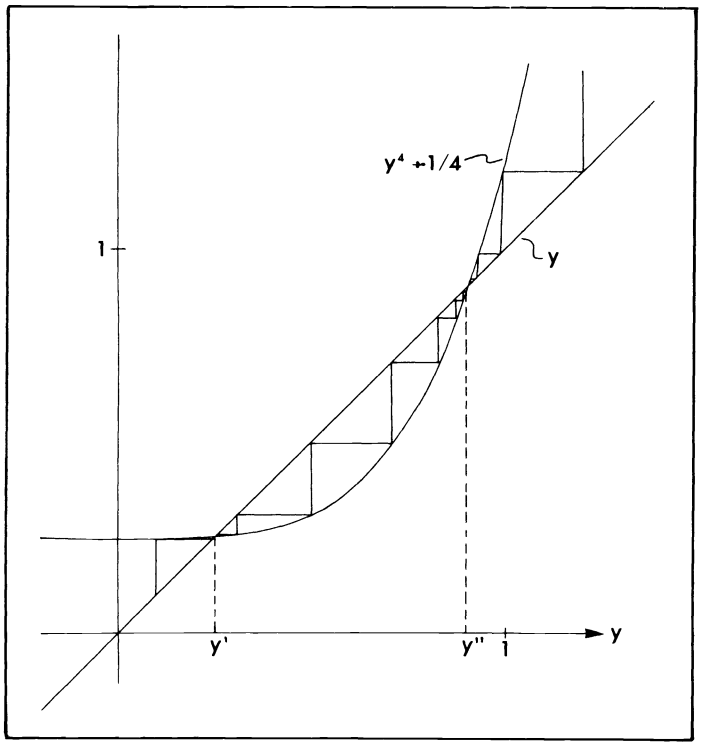


**Fig. 2. Method of successive approximations for solving $4y^4 - 4y + 1 = 0$.**

1/4 indicates how the estimate $y_k$ approaches the two solutions. Criteria on the derivative of S(y) determine whether the algorithm converges.[11]

The error-reduction algorithm, as described by Eqs. (9) and (10), is analogous to the example of successive approximations described above, except that instead of operating on a scalar y, it operates on a function g(x). As seen from the example, the method of successive approximations may or may not converge, depending on the particular form chosen and on the initial estimate. Fortunately, as will be discussed further in Sec. 4, the error-reduction algorithm never diverges. It may, however, stagnate. A simple example of stagnation of the method of successive approximations is shown by the following. In solving $x = 2 - x$ (which has the obvious solution $x = 1$), starting with the initial estimate $x_0$, one obtains $x_1 = 2 - x_0$, $x_2 = 2 - (2 - x_0) = x_0, \ldots, x_{2k-1} = 2 - x_0, x_{2k} = x_0$, etc., and no progress is made toward the solution.

Another way of understanding the error-reduction algorithm, applicable for certain sets of constraints, is the alternating projection of the function onto specified subspaces in a Hilbert space.[12] This, along with the possibility of closed-form solutions,[13] is discussed in the contribution to this volume by Marks and Smith.

## 3. APPLICATIONS

A large number of important problems in optics and related fields fit the problem description in Sec. 1 and can be solved by the iterative algorithm (by the error-reduction algorithm described in Sec. 2 and the related algorithms described in Sec. 4). One particular application, that of spectral extrapolation or superresolution, is discussed in detail in the contribution to this volume by Marks and Smith. In this section, several classes of applications are listed, followed by more detailed discussions of some of the applications, including examples.

In Sec. 1, a distinction was made between reconstruction problems and synthesis problems. Another useful way to classify such problems is according to the type of information available. For one set of problems, the modulus (magnitude or amplitude) of a complex-valued function and the modulus of its Fourier transform are measured (or are given), and one wishes to know the phase of the Fourier transform pair in both domains. These include the phase retrieval problem in electron microscopy, the phase retrieval problem in wavefront sensing, the design optimization of radar signals and antenna arrays having desirable properties, and phase coding and spectrum shaping problems for computer-generated holograms and other applications. These applications often involve the Fresnel transform for the near-field case instead of the Fourier transform.

For another set of problems, the function is known to be real and nonnegative and the modulus of its Fourier transform is measured. These include the phase problems of x-ray crystallography, Fourier transform spectroscopy, imaging through atmospheric turbulence using interferometer data, and pupil function determination.

For another set of problems, a low-resolution (i.e., a low-pass filtered) version of a function is measured (i.e., its complex Fourier transform is measured only over a certain interval), and the function is known to have a finite extent (i.e., it is zero outside of some known region of support). This is the spectral extrapolation or superresolution problem for band-limited time signals or for imaging of objects of finite extent.

For another set of problems, the function is known to be nonnegative and of finite extent and its complex Fourier transform is measured only over a partially filled aperture. These include the interpolation of the complex visibility function for long baseline radio interferometry and the missing-cone problem in x-ray tomography.

For still another set of problems, the modulus of a complex-valued function is given, and one wishes to find an associated phase function that results in a Fourier transform whose complex values fall on a prescribed set of quantized complex values. These include the reduction of quantization noise in computer-generated holograms and in coded signal transmission.

Another problem is to reconstruct the modulus of a complex-valued function from the phase of the function, given the fact that the Fourier transform of the function has finite support.

The number of types of problems solvable by the iterative algorithm appears to be limited only by one's ingenuity in defining different combinations of information that might be available in each of two domains.

### 3.1. Modulus—modulus constraints

#### *3.1.1. Electron microscopy*

Among the applications for which the modulus is given in each of two domains, the electron microscopy phase retrieval problem was one of the earliest applications of the error-reduction algorithm and has been the problem most heavily investigated.[1,4,8,14,15] The error-reduction (Gerchberg-Saxton) algorithm has been shown to perform very successfully for this problem, and the solution is usually unique.[15] The reader is referred to a book by Saxton[4] for a thorough review.

#### *3.1.2. Spectrum shaping*

A second application for which the modulus is given in each of two domains is the spectrum shaping problem. Spectrum shaping is a synthesis problem that can be stated as follows: given the modulus $|f(x)|$ of a complex-valued wavefront, $g(x) = |f(x)| \exp[i\theta(x)]$, find a phase function $\theta(x)$ such that $|\mathcal{F}[g(x)]|$ is equal to a given spectrum $|F(u)|$. Such a problem is the one suggested by the Escher engraving shown in Fig. 3, in which a bird transforms into a fish. One wishes to find a function with modulus being a picture of a fish, which has a Fourier transform with modulus being a picture of a bird. Or, in terms of computer holography, find a phase function to assign to the image of a fish so that the hologram will look like an image of a bird. Figure 4(a) shows the actual "bird" and "fish" binary patterns used for our experiment.[7] For the first iteration, the fish object was random phase coded, Fourier transformed, and the modulus of the Fourier transform was replaced with the modulus of the bird pattern shown in Fig. 4(a). The result was inverse

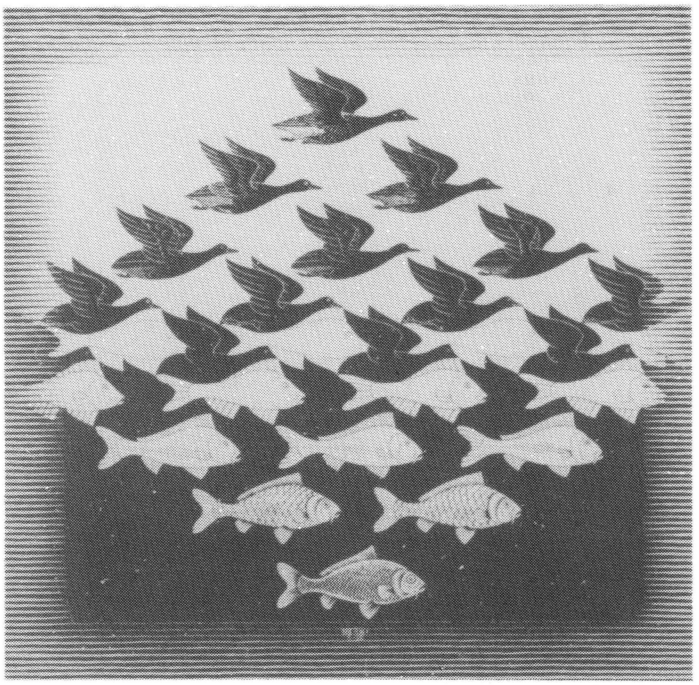

Fig. 3. Bird transforms into fish ("Sky and Water" by M. C. Escher). This reproduction was authorized by the M. C. Escher Foundation, The Hague, Holland/G.W. Breughel.

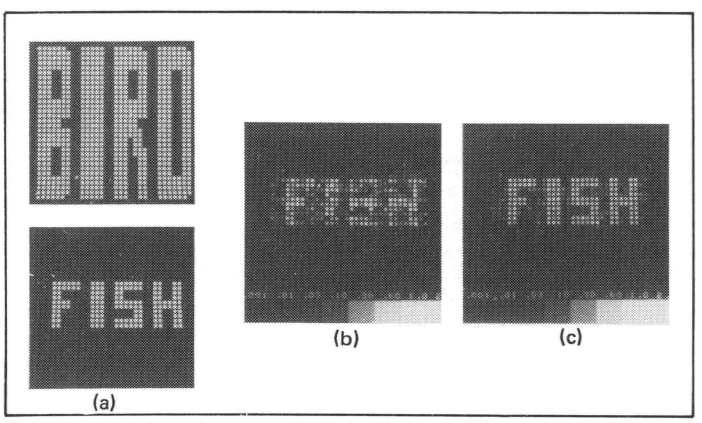


Fig. 4. Example of spectrum shaping. (a) Bird hologram and desired fish image; (b) fish output image after random phase coding of input; (c) output image after seven iterations of the iterative algorithm.

Fourier trsformed, yielding the very noisy output image shown in Fig. 4(b). The iterative algorithm was then used for seven iterations, resulting in the improved image shown in Fig. 4(c). For this example, increasing the number of iterations resulted in a further improvement of the quality of the image; that is, a Fourier transform pair was found that more closely satisfied the constraints in both domains.

Spectrum shaping is also important in computer holography for reducing quantization noise. The objective of computer holography[16] is to synthesize a transparency that can modulate a wavefront according to a calculated wavefront, often corresponding to Fourier coefficients (or samples of the Fourier transform of an image) computed by the discrete Fourier transform. Let $F = \mathcal{F}[f]$ be the desired wavefront modulation and f be the complex-valued function describing the desired image. Due to the limitations of the recording devices and materials used to synthesize computer holograms, it is often not possible to represent exactly any arbitrary complex Fourier coefficient. An extreme example of this is the

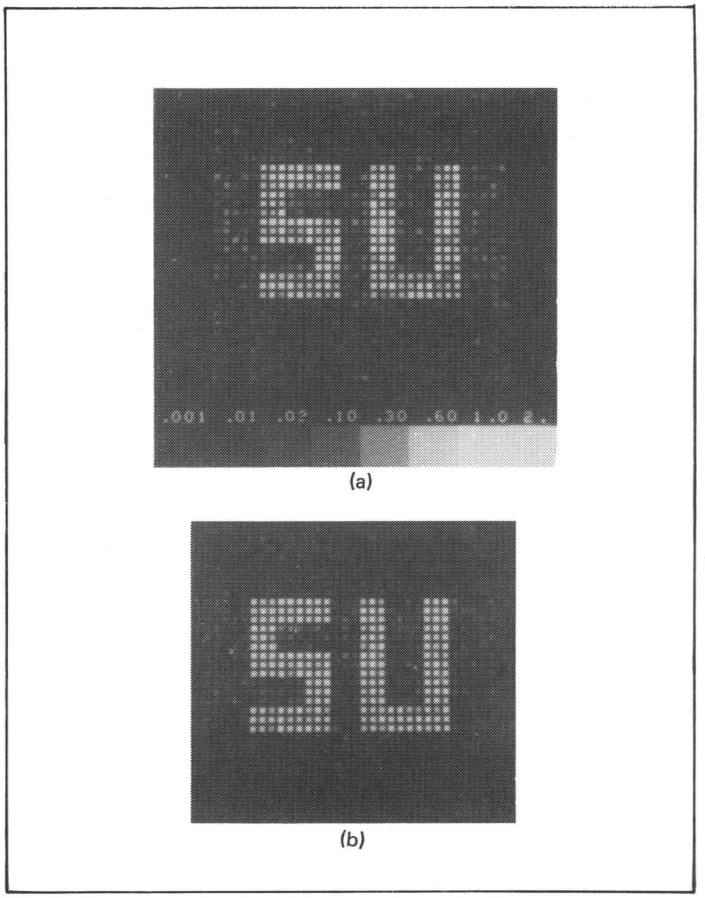


Fig. 5. Computer-simulated images from kinoform. (a) object random phase coded; (b) after eight iterations of the iterative algorithm.

kinoform,[17] which allows nearly continuous phase control by varying the thickness of the recording medium, but which quantizes the modulus to a single level. (If the gray-level recording device used to synthesize a kinoform has a finite number of gray levels, then the phase is quantized as well.) The desired coefficient F is only approximated by the quantized value $F/|F|$. Since only the squared modulus (the intensity) of the image is observed, one is free to choose the phase of the object (phase code the object) in such a way as to reduce the variance (dynamic range) of $|F|$. In this way the quantization noise in kinoforms and, to a lesser extent, in other types of computer-generated holograms can be greatly reduced. Random phase and various deterministic phase codes[18] cause considerable reduction in the variance of $|F|$, but substantial errors remain.[19]

It was for the kinoform application that the iterative algorithm was first invented.[2,3] Figure 5 shows an example of its use for this synthesis problem.[7] Figure 5(a) shows the image resulting when the input image was random phase coded, encoded as a kinoform in the Fourier plane, and reconstructed by inverse Fourier transformation. The ideal image would be the binary ( = 0 or 1) block letters SU. Figure 5(b) shows the improved result after eight iterations of the iterative algorithm. In this case, the image-domain constraint is that the modulus equal the SU pattern, and the Fourier-domain constraint is that the modulus equal a constant.

A problem very similar to the kinoform problem is that of synthesizing a quasi-random radar signal having good autocorrelation properties. Specifically, one would like to synthesize a radar signal f(t) which is a pure phase function, i.e., $|f(t)| = 1$, over some interval of time and which has an autocorrelation function which approaches a delta-function, i.e., its Fourier spectrum $|F(\nu)|^2$ is constant over the bandwidth of interest. From the examples shown

above, it is obvious that the iterative method would be an effective tool for synthesizing such radar signals.

Another spectrum-shaping application is the phasing of elements of an array of antennas in order to achieve a far-field pattern having desirable properties. For example, one might wish to phase the antenna elements in such a way as to minimize the maximum sidelobe of the far-field pattern or to place nulls of the antenna pattern at several different prescribed locations simultaneously. A related application for which the iterative method has been used is the transformation of a Gaussian laser beam into a beam having a more nearly rectangular profile.[20]

### *3.1.3. Wavefront sensing*

The wavefront sensing application is very similar to the electron microscopy problem. Suppose that one measures the image $|f(x)|^2$ of a point source using an aberrated optical system, where the aberrations may be due to atmospheric turbulence or due to the optical system itself. Assuming that the aberration is a pure phase function, then F(u), the Fourier transform of f(x), has modulus $|F(u)|$ equal to the aperture function of the optical system. The problem is to reconstruct the phase of F(u) given $|F(u)|$ and $|f(x)|$. Several investigators[9,21,22] have applied the error-reduction algorithm to this problem with generally good results.

## 3.2. Nonnegativity—modulus constraints

For some reconstruction problems, the physical quantity of interest can be represented as a nonnegative function, and one is able to measure only the modulus of its Fourier transform (or at least the measured modulus information has a much higher signal-to-noise ratio than the measured phase). From the Fourier modulus, one wishes to reconstruct the Fourier phase or, equivalently, the function itself. Since the autocorrelation of the function is available as the inverse Fourier transform of the squared Fourier modulus,[23] this problem is equivalent to reconstructing the function from its autocorrelation. This problem, referred to as the phase retrieval problem of optical coherence theory, arises in spectroscopy,[24] a one-dimensional problem; in astronomy, a two-dimensional problem; and in x-ray crystallography,[25] a three-dimensional problem. In spectroscopy, the nonnegative spectral density, $g(\nu)$, is the Fourier transform of the complex degree of temporal coherence, $\gamma(\tau)$, of which $|\gamma(\tau)|$ is most easily measured. In x-ray crystallography, the nonnegative electron density function, $\varrho(x, y, z)$, which is periodic, is the Fourier transform of the structure factor $F_{hkl}$, of which $|F_{hkl}|$ is measured by a diffractometer. The astronomy problem will be described in more detail later.

### *3.2.1. Uniqueness of solutions*

For the one-dimensional problem, use of the iterative algorithm (or any other method) to reconstruct the function from its Fourier modulus is of limited interest since the solution in the general case is usually not unique.[26,27] The uniqueness of the solution for the one-dimensional problem can be analyzed using the theory of analytic functions, from which one finds that additional solutions can be generated by "flipping zeros" of the Fourier transform analytically extended over the complex plane.[26,27] The additional "solutions" have the same support as the original function, but are not guaranteed to be nonnegative; therefore one could reduce the degree of ambiguity by generating all possible "solutions" and then keeping only the nonnegative ones.[28]

For certain special types of one-dimensional functions, there is a high probability that the solution is unique. For a function having two separated intervals of support, being separated by an interval over which the function is zero, the solution usually is unique,[29,30] but only if the two intervals of support are sufficiently separated.[31] Another special type of function for which the solution is usually unique is one consisting of a summation of a number of delta-functions randomly distributed in space; for such functions, one does not need the iterative method—they can be reconstucted by a simple noniterative method involving the product of three

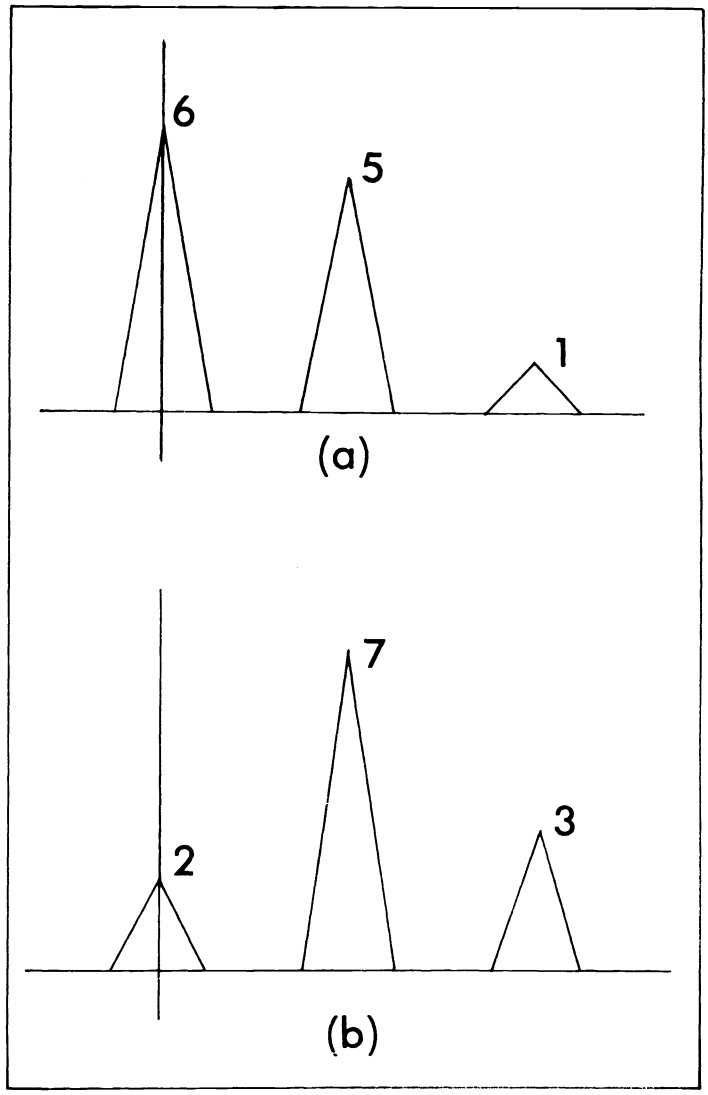

**Fig. 6. Functions (a) and (b) having the same Fourier modulus.**

translates of the autocorrelation function.[32]

In the event that multiple solutions do exist, it would not appear that the algorithm would be biased toward one over another, and one would expect the algorithm to converge to different solutions, depending on the initial input to the algorithm. For example, Fig. 6 shows two functions having the same Fourier modulus. In a computer experiment using the iterative reconstruction algorithm on the functions' Fourier modulus, it converged to one of the solutions in about half of the trials and converged to the other solution in the other half of the trials, depending on the random number sequences used as the initial input to the algorithm.

For the problem in two or more dimensions, it appears that the solution is usually unique. Considering sampled functions defined on a rectangular grid of points, Bruck and Sodin[33] showed that the existence of additional solutions is equivalent to the factorability of a polynomial representation of the Fourier transform. Since a polynomial of one variable of degree M can always be factored into M prime factors, there are $2^{M-1}$ solutions in the one-dimensional case. Once again, only some of the "solutions" may be nonnegative. On the other hand, polynomials of two or more variables having arbitrary coefficients are only rarely factorable; consequently, the two-dimensional problem is usually unique. Attempts have also been made to extend this concept to continuous, as opposed to discrete, functions.[34] Although it is always possible to make up examples in two dimensions that are not unique,[35] it appears to be

true that for two-dimensional functions drawn from the real world, the solution is usually unique. The general uniqueness of the two-dimensional case is indicated by experimental reconstruction results using the iterative algorithm.[36] Furthermore, noise in the Fourier modulus data has had the effect of adding noise to the reconstructed function rather than causing the algorithm to converge to a radically different solution.[37]

### *3.2.2. Astronomical reconstruction*

The problem of reconstructing a two-dimensional nonnegative function from the modulus of its Fourier transform arises in astronomy. Due to atmospheric turbulence, the resolution attainable from large optical telescopes on earth is only about one second of arc, many times worse than the diffraction limit imposed by the diameter of the telescope aperture. For a five-meter telescope aperture, the diffraction-limited resolution would be about 0.02 seconds of arc—fifty times finer. Despite atmospheric turbulence, it is possible to measure the modulus of the Fourier transform of a space object out to the diffraction limit of the telescope using interferometric techniques.[38-41] The autocorrelation of the object can be computed from the Fourier modulus, allowing the diameter of the object to be determined. However, unless the Fourier transform phase is also measured, it was previously not possible to determine the object itself, except for some special cases. Previous attempts to solve this problem had not proven to be practical for complicated two-dimensional objects.

The problem of reconstructing an object from interferometer data can be solved by the iterative method.[42,36] The Fourier-domain constraint is that the Fourier modulus equal the Fourier modulus measured by an interferometer, and the function-domain constraint is that the object function be nonnegative. Figure 7 shows an example. Fig. 7(a) shows a computer-synthesized object used for the experiment—a sun-like disk having "solar flares" and bright and dark "sunspots." The modulus of its Fourier transform is shown in Fig. 7(b). Figure 7(c) shows a square of random numbers used as the initial input for the iterative algorithm. Figures 7(d), 7(e), and 7(f) show the reconstruction results after 20, 230, and 600 iterations, respectively. Figure 7(g) shows the initial input for a second trial, and the reconstruction results after 2 and 215 iterations are shown in Figs. 7(h) and 7(i), respectively. Comparing Figs. 7(f) and 7(i) with the original object in Fig. 7(a), one sees that for both trials, the reconstructed images match the original object very closely. Note that inverted solutions such as Fig. 7(f) are permitted for this problem since the modulus of the Fourier transform of f(–x) equals the modulus of the Fourier transform of f(x) for real-valued f(x). Other successful reconstruction experiments have been performed on data simulated to have the types of noise present in stellar speckle interferometry,[39] and it appears that under realistic levels of photon noise for fairly bright objects, diffraction-limited images can be reconstructed.[37] Initial expements have also been carried out on data from telescopes.[43]

### *3.2.3. Pupil reconstruction and synthesis*

Another case in which one may want to reconstruct a two-dimensional nonnegative function from its Fourier modulus is in pupil function determination. In a diffraction-limited optical system, the point-spread function is the squared Fourier modulus of the system's pupil function. Equivalently, the optical transfer function is the autocorrelation of the pupil function.[44] Given the point-spread function at a given location in an image plane, one could use the iterative algorithm to retrieve the corresponding pupil function, in a way that is mathematically equivalent to the astronomy problem. Turning this problem around, one could use the iterative algorithm to synthesize (design) a pupil function that would yield a given, desired point-spread function while possibly satisfying other desirable constraints as well.

## 3.3. Finite extent—measurement over part of an aperture

In a number of reconstruction problems, there is a function of known finite extent (or support) and one wishes to reconstruct the function with resolution appropriate to an aperture in the Fourier domain more complete than the one over which measurements were actually taken. In some cases, the desired aperture is simply larger than the aperture over which measurements were taken, and so one wishes to extrapolate the function's Fourier transform, i.e., to obtain superresolution of the function. In other cases, one has made measurements over a partially filled aperture, in which case one wishes to interpolate the Fourier transform of the function, and thereby obtain an improved impulse response in the function domain.

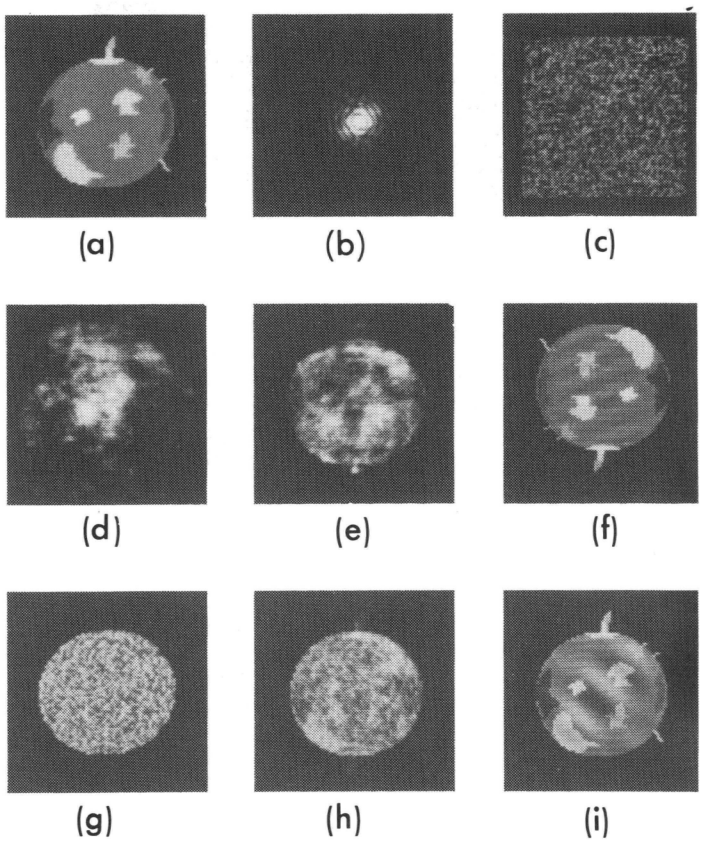


**Fig. 7. Reconstruction of a nonnegative function from its Fourier modulus. (a) Test object; (b) modulus of its Fourier transform; (c) initial estimate of the object (first test); (d)-(f) reconstruction results—number of iterations: (d) 20, (e) 230, (f) 600; (g) initial estimate of the object (second test); (h)-(i) reconstruction results—number of iterations: (h) 2, (i) 215.**

### *3.3.1. Extrapolation or superresolution*

The error-reduction algorithm was first applied to the extrapolation (or superresolution) problem by Gerchberg.[45] Much has been written about the iterative algorithm, specifically the error-reduction algorithm, as it relates to this problem, including various ways of understanding the algorithm (see the end of Sec. 2) and proofs of convergence.[10,12,13,46-48] For this particular problem, the nature of the constraints makes it possible to implement the algorithm by a feedback optical processor[49,50] taking on the order of $10^{-9}$ seconds per iteration even for the two-dimensional case. Marks and Smith describe these matters in detail elsewhere in this volume.

### *3.3.2. Interpolation*

In tomographic imaging systems, many projections of the object are measured, each projection yielding information about a slice through the Fourier transform of the object. When measurements over only a limited cone of angles are made, the effective aperture

in the Fourier domain has gaps, and the impulse response of the system is highly irregular. In applying the iterative algorithm to this problem,[51,52] the function-domain constraint is the finite extent and nonnegativity of the object, and the Fourier domain constraint is that the Fourier transform equal the measured Fourier transform over the measurement aperture.

A problem similar to the tomography problem arises in radio astronomy. The radio sky brightness map is a two-dimensional real, nonnegative function which is the Fourier transform of the complex visibility function. The visibility function is measured by radio interferometry, and in the case of long-baseline interferometry, the visibility function is measured only over a limited set of "tracks" in the Fourier domain, resulting in a partially-filled effective aperture. The error-reduction algorithm has been used to obtain improved maps by, in effect, interpolating the visibility function to fill in the area between the tracks.[53] For this problem, the constraints on the brightness map are that it be nonnegative and be zero outside the known field of view. In the visibility plane, the constraint is that the complex visiblity function equal the measured value within the area of the tracks.

### 3.4. Modulus—quantized values

As mentioned earlier in connection with spectrum shaping, in computer holography one may wish to encode the Fourier transform of an image as a computer-generated hologram, but some types of computer-generated holograms can encode only certain quantized complex values. The kinoform example discussed earlier is a special type of quantization. A more general example is the Lohmann hologram,[54] for which the modulus and phase of a complex sample are determined by the area and relative position, respectively, of an aperture within a sampling cell. The number of allowable quantized values is determined by the number of resolution elements, of the recording device used to fabricate the hologram, used to form one cell. For this synthesis problem, the function-domain constraint is that the modulus of the function equal the desired image modulus and the Fourier-domain constraint is that the complex Fourier coefficients fall on a prescribed set of quantized values. Experiments have shown that synthesizing such a Fourier transform pair is possible using the iterative algorithm.[55,7] For example, Fig. 8(a) shows a simulation of an image produced by a Lohmann hologram having only four modulus and four phase quantization levels when the image was random phase coded. Figure 8(b) shows the image after 13 iterations, a considerable improvement. This problem is one of a more general class of problems regarding the transmission of coded data.

### 3.5. Finite extent—phase

Finally, the iterative algorithm has been used to reconstruct the modulus of a band-limited signal from its phase.[56,57] Or, looking at it in another way, given that a function has finite extent and given the phase of its Fourier transform, reconstruct the modulus of its Fourier transform. For this application, it has been shown that for a wide class of conditions the solution is unique.[56] This application will be discussed further in Sec. 4.

## 4. ALGORITHM CONVERGENCE AND ACCELERATED ALGORITHMS

As mentioned in Sec. 2, the basic iterative algorithm depicted in Fig. 1, referred to as the error-reduction algorithm, has been shown to converge for some applications. In this section, the convergence is proven for all applications. In addition, modified algorithms that often converge much faster than the error-reduction algorithm are discussed.

### 4.1. Convergence of the error-reduction algorithm

For the error-reduction algorithm, the mean-squared error can be defined in general by Eq. (7) or Eq. (8). It is a normalized version of the integral over the square of the amount by which the computed function (or the computed Fourier transform) violates the constraints in the appropriate domain. When the mean-squared error is zero, then a Fourier transform pair has been found that satisfies all the constraints in both domains.

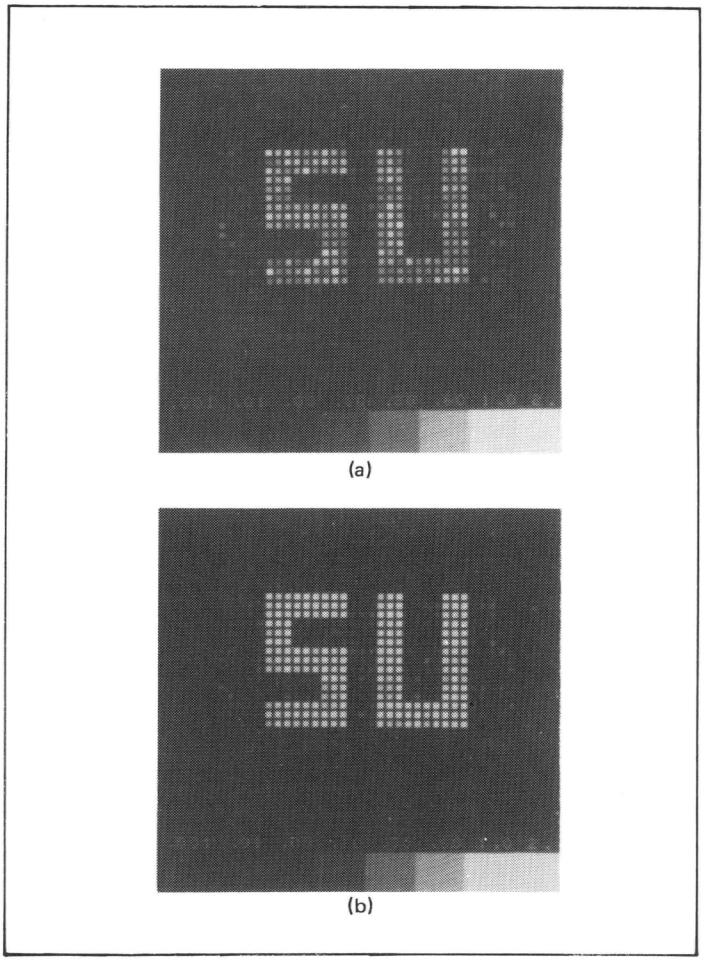


**Fig. 8. Computer-simulated images from hologram with four magnitude and four phase quantized levels. (a) Object random phased coded; (b) after 13 iterations of the iterative method.**

Consider again the steps in the error-reduction algorithm described in Sec. 2. The $k^{th}$ iteration starts with an estimate $g_k(x)$ that satisfies the function-domain constraints. For any coordinate, x, the complex values that g(x) can have that satisfy the function-domain constraints form some set of points in phasor space. For example, if the modulus must equal $|f(x)|$, then the set of such points is a circle of radius $|f(x)|$ in phasor space; if the function must be nonnegative, then the set of such points is the half line on the nonnegative real axis. The function estimate $g_k(x)$ is Fourier transformed, yielding $G_k(u)$. The next step in the algorithm is to form $G'_k(u)$ by changing $G_k(u)$ by the smallest possible amount that allows it to satisfy the Fourier-domain constraints. $G'_k(u)$ is then inverse Fourier transformed, yielding $g'_k(x)$ in the function domain. In the final step, $g_{k+1}(x)$ is formed by changing $g'_k(x)$ by the smallest amount that allows it to satisfy the function-domain constraints. Now consider the unnormalized squared error, given by the numerators in Eqs. (7) and (8). In the Fourier domain, the unnormalized squared error at the $k^{th}$ iteration is

$$e^2_{Fk} = \int_{-\infty}^{\infty} |G_k(u) - G'_k(u)|^2 \, du \tag{14}$$

$$= \int_{-\infty}^{\infty} |g_k(x) - g'_k(x)|^2 \, dx ,$$

where the second line in this equation results from Parseval's theorem. The unnormalized squared error in the function domain at the $k^{th}$ iteration is given by

$$e_{0k}^2 = \int_{-\infty}^{\infty} |g_{k+1}(x) - g'_k(x)|^2 \, dx . \tag{15}$$

Both $g_k(x)$ and $g_{k+1}(x)$ by definition satisfy the function-domain constraints. Also at any given coordinate x, $g_{k+1}(x)$ is the point in phasor space satisfying the function-domain constraints that is closest to $g'_k(x)$. Therefore, for all values of x,

$$|g_{k+1}(x) - g'_k(x)| \leq |g_k(x) - g'_k(x)| , \tag{16}$$

where equality holds only if $g_k(x)$ is just as close in phasor space to $g'_k(x)$ as $g_{k+1}(x)$ is. When there is a point in phasor space satisfying the constraints that is closer to $g'_k(x)$ than $g_k(x)$ is, then the left-hand side of the expression above is strictly less than the right-hand side. Therefore, combining Eqs. (14)–(16),

$$e_{0k}^2 \leq e_{Fk}^2 \tag{17}$$

for a given iteration. From the perfect symmetry of the error-reduction algorithm, as seen from Fig. 1, a similar result holds when one completes the iteration by satisfying the function-domain constraints, thereby forming $g_{k+1}(x)$, and continues the next iteration by Fourier transforming $g_{k+1}(x)$ and causing its transform to satisfy the Fourier-domain constraints. One then finds that

$$e_{F,k+1}^2 \leq e_{0k}^2 \leq e_{Fk}^2 . \tag{18}$$

Therefore, the unnormalized squared error can only decrease (or at least not increase) at each iteration. Since the normalized mean-squared error is simply proportional to the unnormalized squared error, a similar result holds for the errors defined by Eqs. (7) and (8).

While the error-reduction algorithm converges to a solution sufficiently fast for some applications, it is unbearably slow for others. In most cases, the error is reduced rapidly for the first few iterations, and then much more slowly for later iterations.

### 4.2. Input-output algorithms

Resulting from an investigation into the problem of the slow convergence of the error-reduction algorithm, a new and faster-converging algorithm was developed, the input-output algorithm.[55,58,7,36,42] The input-output algorithm differs from the error-reduction algorithm only in the function-domain operation. The first three operations—Fourier transforming g(x), satisfying Fourier domain constraints, and inverse Fourier transforming the result—are the same for both algorithms. Those three operations, if grouped together as shown in Fig. 9, can be considered as a nonlinear system with an input g(x) and an output g′(x). A property of this system is that its output is always a function having a Fourier transform that satisfies the Fourier-domain constraints. Therefore, if the output also satisfies the function-domain constraints, then all the constraints are satisfied and it is a solution to the problem. It is then necessary to determine how to manipulate the input in such a way as to force the output to satisfy the function-domain constraints.

For the error-reduction algorithm, the next input g(x) is chosen to be the current best estimate of the function satisfying the function-domain constraints. However, for the input-output

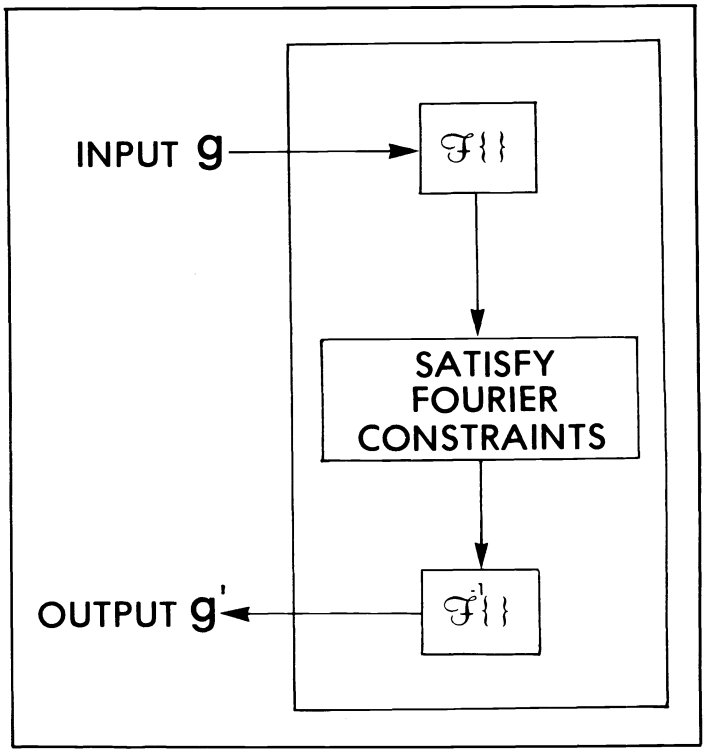


**Fig. 9. Block diagram of the system for the input-output concept.**

algorithm, the input is not necessarily an estimate of the function or a modification of the output, nor does it have to satisfy the constraints; instead, it is viewed as the driving function for the next output. This viewpoint allows one a great deal of flexibility and inventiveness in selecting the next input and allows the invention of an algorithm that converges more rapidly to a solution. As will be seen later, the "input-output algorithm" actually comprises a few different algorithms, all of which are based on the input-output point-of-view.

How the input should be changed in order to drive the output to satisfy the constraints depends on the particular problem at hand. The analysis given in the appendix for a specific application can be generalized as follows. Consider what happens when an arbitrary change is made in the input. Suppose that at the $k^{th}$ iteration the input $g_k(x)$ results in the output $g'_k(x)$. Further, suppose that the input is then changed by adding $\Delta g(x)$:

$$g_{k+1}(x) = g_k(x) + \Delta g(x) . \tag{19}$$

Then one would expect the new output resulting from $g_{k+1}(x)$ to be of the form

$$g'_{k+1}(x) = g'_k(x) + \alpha \Delta g(x) + \text{additional noise.} \tag{20}$$

That is, the expected (or statistical mean) value of the change of the output, due to the change $\Delta g(x)$ of the input, is $\alpha \Delta g(x)$, a constant times the change of the input. The system shown in Fig. 9 is not linear; nevertheless, small changes of the input tend to result in similar changes of the output. The expected value of the change of the output can be predicted, but its actual value cannot be predicted since it has a non-zero variance. In the equation above, this lack of predictability is indicated by the "additional noise" term. The constant $\alpha$ depends on the statistics of $G_k(u)$ and F(u) and on the Fourier-domain constraints.

If the output $g'_k(x)$ does not satisfy the function-domain constraints and if $g'_k(x) + \Delta g_d(x)$ does, then one might try to drive the

output to satisfy the constraints by changing the input in such a way as to cause the output to change by $\Delta g_d(x)$. According to the equation above, the change of the input that will, on the average, cause a change $\Delta g_d(x)$ of the output is

$$\Delta g(x) = \alpha^{-1} \Delta g_d(x) . \tag{21}$$

Thus a logical choice for the new input is

$$g_{k+1}(x) = g_k(x) + \beta \Delta g_d(x) , \tag{22}$$

where $\beta$ is a constant ideally equal to $\alpha^{-1}$, and where $\Delta g_d(x)$ is a function such that $g_k'(x) + \Delta g_d(x)$ satisfies the function-domain constraints. If $\alpha$ is unknown, then a value of $\beta$ only approximately equal to $\alpha^{-1}$ will usually work nearly as well. The use of too small a value of $\beta$ in Eq. (22) will only cause the algorithm to converge more slowly. The noise-like terms in Eq. (20) are kept to a minimum by minimizing $|\beta \Delta g_d(x)|$.

As mentioned earlier, for the input-output algorithm $g_k(x)$ is not necessarily an estimate of the function; it is instead the driving function for the next output. Therefore, it does not matter whether its Fourier transform, $G_k(u)$, satisfies the Fourier-domain constraints. Consequently, for the input-output algorithm, the mean-squared error, $E_F^2$, is unimportant; $E_o^2$ is the meaningful quality criterion. When computing $E_o$ for the input-output algorithm, the $g_{k+1}(x)$ that one should use in the integrand of Eq. (8) is the one determined by the error-reduction algorithm rather than the one computed by the input-output algorithm. That is, $E_o$ should still be a measure of the amount by which the output, $g_k'(x)$, violates the constraints.

Another interesting property of the system shown in Fig. 9 is that if an output $g'(x)$ is used as an input, then its output will be itself. Since the Fourier transform of $g'(x)$ already satisfies the Fourier-domain constraints, $g'(x)$ is unaffected as it goes through the system. Therefore, no matter what input actually resulted in the output $g'(x)$, the output $g'(x)$ can always be considered to have resulted from itself as an input. From this point of view, another logical choice for the new input is

$$g_{k+1}(x) = g_k'(x) + \beta \Delta g_d(x) \tag{23}$$

Note that if $\beta = 1$ in Eq. (23), then this version of the input-output algorithm reduces to the error-reduction algorithm. Since the optimum value of $\beta$ is usually not unity, the error-reduction algorithm can be looked on as a suboptimal subset of one version of the more general input-output algorithm. Depending on the problem being solved, other variations in Eqs. (22) and (23) may be successful ways for choosing the next input.

In order to implement the input-output algorithm using Eq. (22) or (23), one chooses $\Delta g_d(x)$ according to the function-domain constraints. In general, a logical choice is the smallest value of $\Delta g_d(x)$ for which $g_k'(x) + \Delta g_d(x)$ satisfies the function-domain constraints. At those values of x for which $g_k'(x)$ already satisfies the function-domain constraints, one would set $\Delta g_d(x) = 0$. At those values of x for which $g_k'(x)$ violates the function-domain constraints, examples of logical choices of $\Delta g_d(x)$ for various applications are as follows. For the astronomy problem and other applications requiring the function to be nonnegative, choose $\Delta g_d(x) = -g_k'(x)$ where $g_k'(x)$ is negative. For applications requiring the function to be of finite extent, choose $\Delta g_d(x) = -g_k'(x)$ for x outside the known region of support. For applications requiring the function to have modulus equal to $|f(x)|$, choose

$$\Delta g_d(x) = |f(x)| \frac{g_k'(x)}{|g_k'(x)|} - g_k'(x) . \tag{24}$$

In addition to the values of $\Delta g_d(x)$ given above, there are other choices that are successful when used in Eqs. (22) and (23). Any $\Delta g_d(x)$ that moves $g'(x)$ in the general direction of satisfying the function-domain constraints will usually result in an algorithm that works; suboptimum choices of $\Delta g_d(x)$ and of $\beta$ in Eq. (22) or Eq. (23) result in algorithms that converge less rapidly than the optimum. Two examples of other algorithms that converge more rapidly than the "logical" ones described in the preceding paragraph are as follows. For applications requiring the function to have modulus equal to $|f(x)|$, it was noticed that the difference in phase between $g_k'(x)$ and $g_k(x)$ tends to have the same sign as the change of phase of $g_k'(x)$ from one iteration to the next. In order to anticipate the direction that the phase is changing, one could choose a $\Delta g_d(x)$ that tends to rotate the phase angle of the new input toward that of the last output. That is, a good choice for the desired change of the output is

$$\Delta g_d(x) = \left[ |f(x)| \frac{g_k'(x)}{|g_k'(x)|} - g_k'(x) \right] + \left[ |f(x)| \frac{g_k'(x)}{|g_k'(x)|} - |f(x)| \frac{g_k(x)}{|g_k(x)|} \right] \tag{25}$$

in which the first component boosts (or shrinks) the magnitude of the output to match $|f(x)|$ and the second component rotates the phase angle of the input toward the phase angle of the output. For the astronomy problem, it was found that a particularly successful algorithm was to use Eq. (23) at those points where the constraints were satisfied and use Eq. (22) at those points where the constraints were violated, i.e.,

$$g_{k+1}(x) = \begin{cases} g_k'(x), & \text{where constraints satisfied} \\ g_k(x) - \beta g_k'(x), & \text{where constraints violated} \end{cases} \tag{26}$$

Furthermore, it was found that even faster convergence can be obtained by alternating between the above equation and the error-reduction algorithm every few iterations.

Unlike the error-reduction algorithm, the input-output algorithm is not guaranteed to converge; in fact the error may even increase for some of the iterations. However, the input-output algorithm is much less prone to stagnation and therefore in practice converges much faster than the error-reduction algorithm. In some instances during the input-output iterations, $E_o$ may even increase although the visual appearance of the image improves. This behavior, which is poorly understood, is described further in Ref. 59.

From the paragraphs above, it is seen that the "input-output algorithm" is really a family of algorithms. The input-output approach is one that can lead to a number of different algorithms based on the manner in which the nonlinear system of Fig. 9 behaves. One would hope that the principles of control theory and possibly other disciplines could be used to shed further light on this system and help to arrive at algorithms with still more rapid convergence.

It should also be noted that, unlike the error-reduction algorithm, the input-output algorithm does not treat the two domains in a symmetric manner. By reversing the roles of the two domains, one can arrive at a different and possibly more advantageous algorithm.

### 4.3. Relaxation-parameter algorithm

A second method of improved convergence is the use of a relaxation parameter. In solving the problem of reconstructing the magnitude of a band-limited function from its phase (or, equivalently, reconstructing a function of finite extent from the

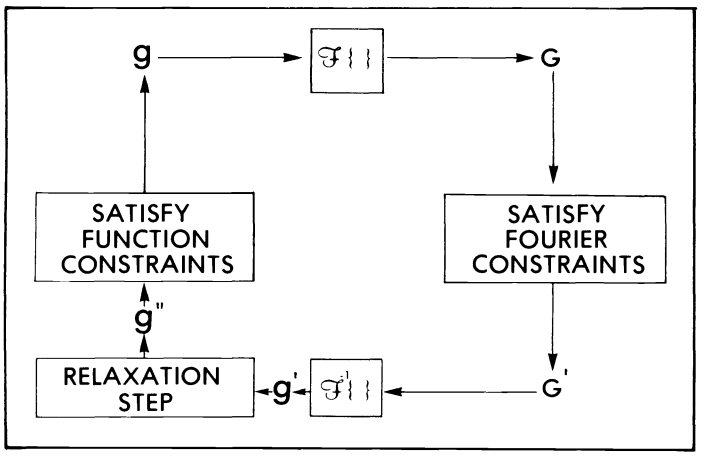


**Fig. 10. Block diagram of the error-reduction algorithm modified to include a relaxation step.**

phase of its Fourier transform), Oppenheim, Hayes, and Lim[57] modified the error-reduction algorithm (Fig. 1) by adding a relaxation step, as shown in Fig. 10. Here the band-limited function is taken to be in the Fourier domain. The function g(x) then must be of finite extent according to the bandwidth of the Fourier-domain function. In the relaxation step, $g_k''(x)$ is formed from $g_k'(x)$ according to

$$g_k''(x) = (1 - \eta_k)g_{k-1}''(x) + \eta_k g_k'(x), \tag{27}$$

and then the new estimate $g_{k+1}(x)$ is formed from $g_k''(x)$ by making it satisfy the function-domain constraints. The parameter $\eta_k$, which is a constant that may vary from one iteration to the next, is the relaxation parameter. For $\eta_k = 1$, $g_k''(x) = g_k'(x)$ and this reduces to the error-reduction approach. For $\eta_k = 0$, $g_k''(x) = g_{k-1}''(x)$, that is, the result from the previous iteration is used. Other values of $\eta_k$ give a linear combination of $g_{k-1}''(x)$ and $g_k'(x)$. For the reconstruction of a function of finite extent from the phase of its Fourier transform or from a segment of its Fourier transform (i.e., the superresolution problem), if $g_1'(x)$ and $g_2'(x)$ both satisfy the Fourier-domain constraint, then the linear combination $\eta g_1'(x) + (1 - \eta)g_2'(x)$ also satisfies the constraint in the Fourier domain. It follows from this that $g_k''(x)$ given by Eq. (27) also satisfies the Fourier-domain constraint. In those cases, it can be shown that the algorithm converges for $0 < \eta_k \leq 1$. However, for other sets of constraints, for example, given the modulus of the Fourier transform, $g_k''(x)$ given by the equation above does not generally satisfy the Fourier-domain constraints and so the relaxation method does not strictly apply.

The optimum value of $\eta_k$ can be determined as follows. Define the function-domain squared error after the relaxation step as

$$e_0^2 = \int_\gamma |g_k''(x)|^2 \, dx, \tag{28}$$

where the region of integration, $\gamma$, is the region over which the function is known to be zero. Setting equal to zero the derivative of $e_0^2$ with respect to $\eta_k$, and solving for $\eta_k$, one finds the optimum value of $\eta_k$ to be given by

$$\eta_k = \frac{-\mathrm{Re}\left\{\int_\gamma g_{k-1}''(x)[g_k'(x) - g_{k-1}''(x)]^* \, dx\right\}}{\int_\gamma |g_k'(x) - g_{k-1}''(x)|^2 \, dx}. \tag{29}$$

The computation of the relaxation parameter by Eq. (29) takes much less time than the computation of one (fast) Fourier transform, and so it does not significantly increase the total computation time of a single iteration.

Use of the relaxation step for the problem of reconstructing a band-limited function from its phase resulted in an order of magnitude improvement in the speed of convergence of the algorithm over that of the error-reduction algorithm.[57]

The relaxation step described above incorporates the optimum combination of the current output with the previous output. It is also possible to extend this concept to include a number of previous outputs,[57] which may result in still more rapid convergence.

It should be noted that the majority of the work referenced in Sec. 3 made use of only the error-reduction algorithm. Improved speed of convergence could be expected if one of the two accelerated algorithms discussed above were employed.

## 5. SUMMARY AND COMMENTS

The iterative error-reduction algorithm, an extension of the Gerchberg-Saxton algorithm to include various types of constraints, has been found to be capable of solving a wide range of difficult problems in optics and other fields. It can be applied to the reconstruction of a function (an object, wavefront, signal, etc.) when only partial information is available in each of two domains, or to the synthesis of a function (wavefront, signal, etc.) having desired properties in each of two domains. The iterative algorithm is reasonably fast for most applications, since the major computational burden, two Fourier transforms per iteration, can be accomplished using the fast Fourier transform (FFT) algorithm. The iterative algorithm has been shown to outperform alternative methods of solving these classes of problems both because of its speed and its tolerance of noise.[4,9] For some applications, a large number of iterations is required for convergence of the error-reduction algorithm. This situation can be remedied by using an algorithm with accelerated convergence, such as the input-output algorithm or an algorithm employing a relaxation step.

The iterative algorithm has been in use for only a few years, yet it has already found numerous applications; and methods of improving the algorithm have been devised. Nevertheless, it is safe to predict that it will be used in the future to solve new problems not discussed here, and it is hoped that further improvements of the algorithm will be discovered.

## POSTSCRIPT

As this book goes to print, further developments relating to the iterative algorithm are occurring at a rapid pace. It has been uncovered that an algorithm equivalent to Gerchberg's[45] error-reduction algorithm for extrapolation was proposed by Ville[60] in 1956, although approached from a different point of view. Relationships between the error-reduction algorithm and gradient search methods have been discovered[59,61,62] and uncovered.[63] And further work on various applications is being reported.[64-83]

## APPENDIX: ANALYSIS OF THE INPUT-OUTPUT SYSTEM

Consider the synthesis problem for kinoforms, for which the Fourier modulus is set equal to a constant. Suppose that the input g(x) to a kinoform system results in the output g'(x). The kinoform has a transmittance $G'(u) = K \exp[i\phi(u)]$, where $\phi(u)$ is the phase of $G(u) = |G(u)| \exp[i\phi(u)] = \mathcal{F}[g(x)]$, and K is a constant. The resulting image is $g'(x) = \mathcal{F}^{-1}[G'(u)]$. Now consider what happens when a change $\Delta g(x)$ is made in the input. As illustrated in the phasor diagrams in Fig. A1, the change $\Delta g(x)$ of the input causes a change $\Delta G(u)$ of its Fourier transform, which causes a change $\Delta G'(u)$ of the kinoform and a corresponding change $\Delta g'(x) = \mathcal{F}^{-1}[\Delta G'(u)]$ of the output image. The goal here is to determine the relationship between the change $\Delta g'(x)$ of the output and the change $\Delta g(x)$ of the input. Figure A2 shows the relationship be-

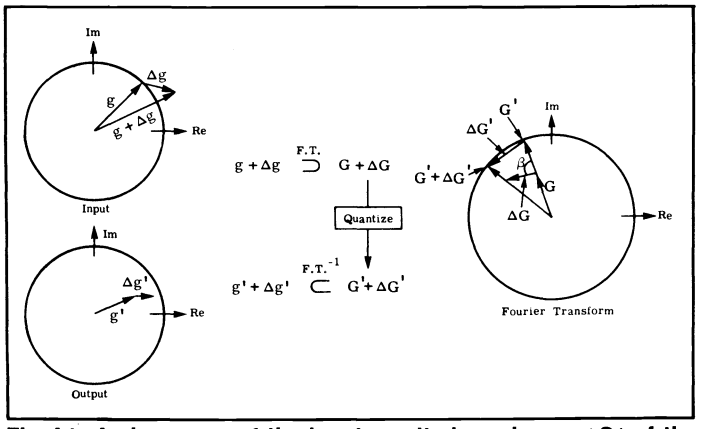


**Fig. A1. A change Δg of the input results in a change ΔG′ of the kinoform and a change of Δg′ of the output.**

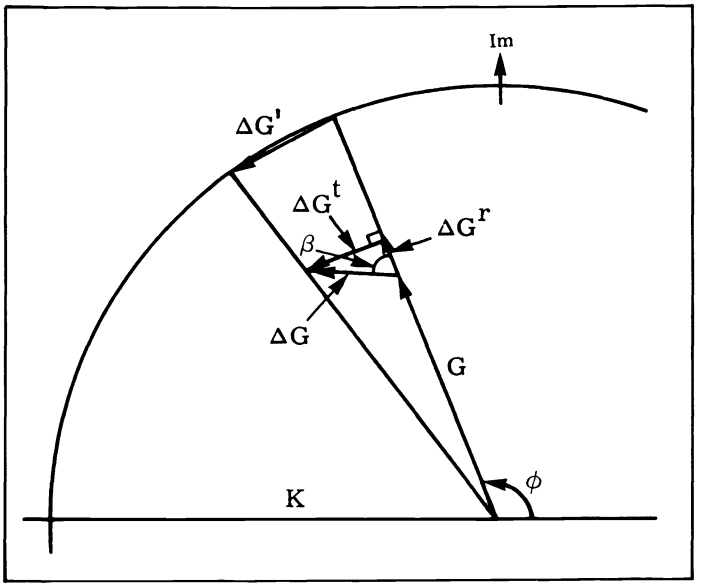


**Fig. A2. Relationship between ΔG′, the change of the kinoform, and two components of ΔG, the Fourier transform of the change of the input.**

tween $\Delta G'(u)$ and two orthogonal components of $\Delta G(u)$. By similar triangles, for $|\Delta G| << |G|$,

$$\Delta G'(u) \simeq \Delta G^t(u) \frac{K}{|G(u)|}, \tag{A1}$$

where the two orthogonal components of $\Delta G(u)$ are

$$\Delta G^r(u) = |\Delta G(u)| \cos\beta(u)\, e^{i\phi(u)} \tag{A2}$$

parallel to G(u), and

$$\Delta G^t(u) = |\Delta G(u)| \sin\beta(u)\, e^{i[\phi(u)+\pi/2]} \tag{A3}$$

orthogonal to G(u); and

$$\Delta G(u) = \Delta G^r(u) + \Delta G^t(u) = |\Delta G(u)|\, e^{i[\phi(u)+\beta(u)]}, \tag{A4}$$

where $\beta(u)$ is the angle between $\Delta G(u)$ and G(u). Only one of the two orthogonal components of $\Delta G(u)$, namely $\Delta G^t(u)$, contributes to $\Delta G'(u)$.

In order to compute the expected change of the output, $E[\Delta g'(x)]$, treat the phase angles $\beta(u)$ and the magnitudes $|G(u)|$ as random variables. Inserting $|\Delta G(u)|$ from Eq. (A4) into Eq. (A3), one obtains

$$\Delta G^t(u) = \Delta G(u)\, e^{-i[\phi(u)+\beta(u)]} \sin\beta(u)\, e^{i\phi(u)}\, e^{i\pi/2}$$

$$= \Delta G(u)[\sin^2\beta(u) + i \sin\beta(u)\cos\beta(u)]. \tag{A5}$$

For $\beta(u)$ uniformly distributed over $[0, 2\pi]$,[19] the expected value of $\Delta G^t(u)$ is

$$E[\Delta G^t(u)] = \Delta G(u)\left(\frac{1}{2} + i\cdot 0\right) = \frac{1}{2}\Delta G(u). \tag{A6}$$

Therefore, the expected value of the change of the output is, using Eqs. (A1) and (A6) and assuming that the magnitudes $|G(u)|$ are identically distributed random variables[19] independent of $\beta(u)$,

$$E[\Delta g'(x)] = E\left[\mathcal{F}(\Delta G')\right]$$

$$= \mathcal{F}[E(\Delta G')] \simeq \mathcal{F}\left[E(\Delta G^t)\, E\left(\frac{K}{|G|}\right)\right]$$

$$\simeq \mathcal{F}\left[\frac{1}{2}\Delta G(u)\right] E\left(\frac{K}{|G|}\right) = \frac{1}{2}\Delta g(x) E\left(\frac{K}{|G|}\right). \tag{A7}$$

That is, the expected change of the output is $\alpha$ times the change of the input, giving us the second term in Eq. (20), where $\alpha = (1/2)E(K/|G|)$. After a few iterations, $|G(u)|$ will not differ greatly from K; then $\alpha \simeq 1/2$.

Similarly, the variance of the change of the output can be shown to be[58]

$$E[|\Delta g'(x)|^2] - |E[\Delta g'(x)]|^2$$

$$\simeq \frac{1}{4}\left\{2E\left(\frac{K^2}{|G|^2}\right) - \left[E\left(\frac{K}{|G|}\right)\right]^2\right\} \cdot \frac{1}{A}\int_{-\infty}^{\infty} |\Delta g(x')|^2\, dx', \tag{A8}$$

where A is the area of the image. That is, the variance of the change of the output $\Delta g'(x)$ at any given x is proportional to the integrated squared change of the entire input. The predictability of $\Delta g'(x)$, and the degree of control with which one can manipulate it, decreases as one makes larger changes of the input. The difference between the actual change of the output and the expected change of the output given by Eq. (A7) is what is meant by the additional noise term in Eq. (20). If, after a few iterations, $|G(u)| \simeq K$, then in Eq. (A8) the factor $(1/4)\{2E(K^2/|G|^2) - [E(K/|G|)]^2\} \simeq 1/4$.

Equations (A7) and (A8) are a justification for the input-output concept: small changes of the input result in similar changes of the output, and so the output can be driven to satisfy the constraints by appropriate changes of the input, as in Eqs. (22) and (23).

## ACKNOWLEDGMENT

The author gratefully acknowledges the support of the U.S. Air Force Office of Scientific Research.

## REFERENCES


1. R. W. Gerchberg and W. O. Saxton, "A practical algorithm for the determination of phase from image and diffraction plane pictures," Optik 35, 237 (1972).
2. P. M. Hirsch, J. A. Jordan, Jr., and L. B. Lesem, "Method of making an object-dependent diffuser," U.S. Patent No. 3,619,022 (Nov. 9, 1971; filed Sept. 17, 1970).
3. N. C. Gallagher and B. Liu, "Method for computing kinoforms that reduces image reconstruction error," Appl. Opt. 12, 2328 (1973).
4. W. O. Saxton, *Computer Techniques for Image Processing in Electron Microscopy* (Academic Press, New York, 1978).
5. W. T. Cochran, J. W. Cooley et al., "What is the fast Fourier transform?" Proc. IEEE 55, 1664 (1967).
6. B. Liu and N. C. Gallagher, "Convergence of a spectrum shaping algorithm," Appl. Opt. 13, 2470 (1974).
7. J. R. Fienup, "Iterative method applied to image reconstruction and to computer-generated holograms," Opt. Eng. 19(3), 297 (1980).
8. D. L. Misell, "A method for the solution of the phase problem in electron microscopy," J. Phys. D.: Appl. Phys. 6, L6-L9 (1973); D. L. Misell, "An examination of an iterative method for the solution of the phase problem in optics and electron optics," J. Phys. D.: Appl. Phys. 6, 2200 (1973).
9. R. Boucher, "Convergence of algorithms for phase retrieval from two intensity distributions," in *1980 International Optical Computing Conference,* W. T. Rhodes, ed., Proc. SPIE 231, 130 (1980).
10. R. W. Schafer, R. M. Mersereau, and M. A. Richards, "Constrained iterative restoration algorithms," Proc. IEEE 69, 432 (1981).
11. G. Dahlquist and A. Björck (translated by N. Anderson), *Numerical Methods* (Prentice-Hall, Englewood Cliffs, N.J., 1974) pp. 2-4.
12. D. C. Youla, "Generalized image restoration by method of alternating orthogonal projections," IEEE Trans. Circuits and Systems CAS-25, 694 (1978).
13. M. S. Sabri and W. Steenaart, "An approach to band-limited signal extrapolation: The extrapolation matrix," IEEE Trans. Circuits and Systems CAS-25, 74 (1978).
14. A. M. J. Huiser, P. Van Toorn, and H. A. Ferwerda, "On the problem of phase retrieval in electron microscopy from image and diffraction pattern I-IV," Optik 47, 123 (1977); J. Gassmann, "Optimal iterative phase retrieval from image and diffraction intensities," Optik 48, 347 (1977).
15. A. M. J. Huiser, A. J. J. Drenth, and H. A. Ferwerda, "On phase retrieval in electron microscopy from image and diffraction pattern," Optik 45, 303 (1976); A. M. J. Huiser and H. A. Ferwerda, "On the problem of phase retrieval in electron microscopy from image and diffraction pattern II: on the uniqueness and stability," Optik 46, 407 (1976); A. J. Devaney and R. Chidlaw, "On the uniqueness question in the problem of phase retrieval from intensity measurements," J. Opt. Soc. Am. 68, 1352 (1978).
16. T. S. Huang, "Digital holography," Proc. IEEE 59, 1335 (1971); W.-H. Lee, "Computer-generated holograms: techniques and applications," in E. Wolf, ed., *Progress in Optics,* Vol. 16 (North-Holland, 1978) pp. 121-232; W. J. Dallas, "Computer-generated holograms," in B. R. Frieden, ed., *The Computer in Optical Research* (Springer-Verlag, N.Y., 1980), Chapter 6.
17. L. B. Lesem, P. M. Hirsch, and J. A. Jordan, Jr., "The kinoform: A new wavefront reconstruction device," IBM J. Res. Develop. 13, 150 (1969).
18. H. Akahori, "Comparison of deterministic phase coding with random phase coding in terms of dynamic range," Appl. Opt. 12, 2336 (1973).
19. R. S. Powers and J. W. Goodman, "Error rates in computer-generated holographic memories," Appl. Opt. 14, 1690 (1975).
20. W.-H. Lee, "Method for converting a Gaussian laser beam into a uniform beam," Opt. Commun. 36, 469 (1981).
21. R. A. Gonsalves, "Phase retrieval from modulus data," J. Opt. Soc. Am. 66, 961 (1976).
22. J. Maeda and K. Murata, "Retrieval of wave aberration from point spread function or optical transfer function data," Appl. Opt. 20, 274 (1981).
23. R. N. Bracewell, *The Fourier Transform and Its Applications,* 2nd Edition (McGraw-Hill, New York, 1978).
24. E. Wolf, "Is a complete determination of the energy spectrum of light possible from measurements of the degree of coherence?" Proc. Phys. Soc. (London) 80, 1269 (1962).
25. G. H. Stout and L. H. Jensen, *X-Ray Structure Determination* (Macmillan, London, 1968).
26. A. Walther, "The question of phase retrieval in optics," Optica Acta 10, 41 (1963).
27. E. M. Hofstetter, "Construction of time-limited functions with specified autocorrelation functions," IEEE Trans. Info. Theory IT-10, 119 (1964).
28. R. H. T. Bates, "Contributions to the theory of intensity interferometry," Mon. Not. R. Astr. Soc. 142, 413 (1969).
29. A. H. Greenaway, "Proposal for phase recovery from a single intensity distribution," Opt. Lett. 1, 10 (1977).
30. R. H. T. Bates, "Fringe visibility intensities may uniquely define brightness distributions," Astron. and Astrophys. 70, L27-L29 (1978).
31. T. R. Crimmins and J. R Fienup, "Ambiguity of phase retrieval for functions with disconnected support," J. Opt. Soc. Am. 71, 1026 (1981).
32. J. R. Fienup, T. R. Crimmins, and W. Holsztynski, "Reconstruction of the support of an object from the support of its autocorrelation," J. Opt. Soc. Am. 72, 610 (1982).
33. Yu. M. Bruck and L. G. Sodin, "On the ambiguity of the image reconstruction problem," Opt. Commun. 30, 304 (1979).
34. W. Lawton, "A numerical algorithm for 2-D wavefront reconstruction from intensity measurements in a single plane," in *1980 International Optical Computing Conference,* W. T. Rhodes, ed., Proc. SPIE 231, 94 (1980).
35. A. M. J. Huiser and P. Van Toorn, "Ambiguity of the phase-reconstruction problem," Opt. Lett. 5, 499 (1980).
36. J. R. Fienup, "Space object imaging through the turbulent atmosphere," Opt. Eng. 18, 529 (1979).
37. G. B. Feldkamp and J. R. Fienup, "Noise properties of images reconstructed from Fourier modulus," in *1980 International Optical Computing Conference,* W. T. Rhodes, ed., Proc. SPIE 231, 84 (1980).
38. A. Labeyrie, "Attainment of diffraction limited resolution in large telescopes by Fourier analysing speckle patterns in star images," Astron. and Astrophys. 6, 85 (1970).
39. D. Y. Gezari, A. Labeyrie, and R. V. Stachnik, "Speckle interferometry: diffraction-limited measurements of nine stars with the 200-inch telescope," Astrophys. J. Lett. 173, L1-L5 (1972).
40. D. G. Currie, S. L. Knapp, and K. M. Liewer, "Four stellar-diameter measurements by a new technique: amplitude interferometry," Astrophys. J. 187, 131 (1974).
41. R. Hanbury Brown and R. Q. Twiss, "Correlation between photons in two coherent beams of light," Nature 177, 27 (1956).
42. J. R. Fienup, "Reconstruction of an object from the modulus of its Fourier transform," Opt. Lett. 3, 27 (1978).
43. J. R. Fienup and G. B. Feldkamp, "Astronomical imaging by processing stellar speckle interferometry data," in *Applications of Speckle Phenomena,* W. H. Carter, ed., Proc. SPIE 243, 95 (1980).
44. J. W. Goodman, *Introduction to Fourier Optics* (McGraw-Hill, San Francisco, 1968).
45. R. W. Gerchberg, "Super-resolution through error energy reduction," Optica Acta 21, 709 (1974).
46. A. Papoulis, "A new algorithm in spectral analysis and band-limited extrapolation," IEEE Trans. Circuits and Systems CAS-22, 735 (1975).
47. J. A. Cadzow, "An extrapolation procedure for band-limited signals," IEEE Trans. Acoust., Speech, Signal Processing ASSP-27, 4 (1978).
48. C. K. Rushforth and R. L. Frost, "Comparison of some algorithms for reconstructing space-limited images," J. Opt. Soc. Am. 70, 1539 (1980).
49. R. J. Marks II, "Coherent optical extrapolation of 2-D band-limited signals: processor theory," Appl. Opt. 19, 1670 (1980).
50. R. J. Marks II and David K. Smith, "Iterative coherent processor for band-limited signal extrapolation," in *1980 International Optical Computing Conference,* W. T. Rhodes, ed., Proc. SPIE 231, 106 (1980).
51. K.-C. Tam and V. Perez-Mendez, "Limited-angle 3-D reconstructions using Fourier transform iterations and Radon transform iterations," in *1980 International Optical Computing Conference,* W. T. Rhodes, ed., Proc. SPIE 231, 142 (1980).
52. T. Sato, S. J. Norton et al., "Tomographic image reconstruction from limited projections using iterative revisions in image and transform spaces," Appl. Opt. 20, 395 (1981).
53. A. E. E. Rogers, "Method of using closure phases in radio aperture synthesis," in *1980 International Optical Computing Conference,* W. T. Rhodes, ed., Proc. SPIE 231, 10 (1980).
54. B. R. Brown and A. W. Lohmann, "Computer-generated binary holograms," IBM J. Res. Develop. 13, 160 (1969); A. W. Lohmann and D. P. Paris, "Binary Fraunhofer holograms, generated by computer," Appl. Opt. 6, 1739 (1967).
55. J. R. Fienup, "Reduction of quantization noise in kinoforms and computer-generated holograms," J. Opt. Soc. Am. 64, 1395 (1974) (Abstract).
56. M. H. Hayes, J. S. Lim, and A. V. Oppenheim, "Signal reconstruction from phase or magnitude," IEEE Trans. Acoust., Speech, Signal Processing ASSP-28, 672 (1980).
57. A. V. Oppenheim, M. H. Hayes, and J. S. Lim, "Iterative procedure for signal reconstruction from phase," in *1980 International Optical Computing Conference,* W. T. Rhodes, ed., Proc. SPIE 231, 121 (1980).
58. J. R. Fienup, "Improved synthesis and computational methods for computer-generated holograms," Ph.D. Thesis, Stanford University, May 1975 (University Microfilms No. 75-25523), Chapter 5.
59. J. R. Fienup, "Phase retrieval algorithms: a comparison," Appl. Opt. 21, 2758 (1982).
60. J.-A. Ville, "Sur le prolongement des signaux a spectre borne," Cables et Transmission 1, 44 (1956).

61. H. Maitre, "Iterative superresolution: some new fast methods," Opt. Acta 28, 973 (1981).
62. A. K. Jain and S. Ranganath, "Extrapolation algorithms for discrete signals with application in spectral estimation," IEEE Trans. Acoustics, Speech, Signal Processing ASSP-29 830 (1981).
63. M. T. Manry and J. K. Aggarwal, "The design of multi-dimensional FIR digital filters by phase correction," IEEE Trans. Circuits and Systems CAS-23, 185 (1976).
64. J. N. Mait and W. T. Rhodes, "Iterative design of pupil functions for bipolar incoherent spatial filtering," in *Processing of Images and Data from Optical Sensors,* W. H. Carter, ed., Proc. SPIE 292, 66 (1981).
65. J. G. Walker, "Object reconstruction from turbulence-degraded images," Opt. Acta 28, 1017 (1981).
66. H. Stark, D. Cahana, and H. Webb, "Restoration of arbitrary finite-energy optical objects from limited spatial and spectral information," J. Opt. Soc. Am. 71, 635 (1981).
67. K. C. Tam and V. Perez-Mendez, "Tomographic imaging with limited-angle input," J. Opt. Soc. Am. 71, 582 (1981).
68. T. F. Quatieri, Jr., and A. V. Oppenheim, "Iterative technique for minimum phase signal reconstruction from phase or magnitude," IEEE Trans. Acoustics, Speech, Signal Processing ASSP-29, 1187 (1981).
69. A. V. Oppenheim, "The importance of phase in signals," Proc. IEEE 69, 529 (1981).
70. L. S. Taylor, "The phase retrieval problem," IEEE Trans. Antennas Propagation AP-3, 386 (1981).
71. R. A. Gonsalves, "Phase retrieval and diversity in adaptive optics," Opt. Eng. 21, 829 (1982).
72. T. Sato, K. Saski, Y. Nakamura, M. Linzer, and S. J. Norton, "Tomographic image reconstruction from limited projections using coherent feedback," Appl. Opt. 20, 3073 (1981).
73. D. Cahana and H. Stark, "Bandlimited image extrapolation with faster convergence," Appl. Opt. 20, 2780 (1981).
74. J. R. Fienup, "Image reconstruction for stellar interferometry," in *Current Trends in Optics,* F. T. Arecchi and F. R. Aussenegg, eds. (Taylor and Francis, London, 1981) pp. 95-102.
75. V. T. Tom, T. F. Quatieri, M. H. Hayes and J. H. McClellan, "Convergence of iterative nonexpansive signal reconstruction algorithms," IEEE Trans. Acoustics, Speech, Signal Processing ASSP-29, 1052 (1981).
76. J. S. Lim and N. A. Malik, "A new algorithm for two-dimensional maximum entropy power spectrum estimation," IEEE Trans. Acoustics, Speech, Signal Processing ASSP-29, 401 (1981).
77. A. Lent, "An iterative method for the extrapolation of band-limited functions," J. Math. Analysis and Applications 83, 554 (1981).
78. W. D. Montgomery, "Optical applications of von Neumann's alternating-projection theorem," Opt. Lett. 7, 1 (1982).
79. W. D. Montgomery, "Restoration of images possessing a finite Fourier series," Opt. Lett. 7, 54 (1982).
80. N. C. Gallagher and D. W. Sweeney, "Infrared holographic optical elements with applications to laser material processing," IEEE J. Quantum Electronics QE-15, 1369 (1979).
81. F. A. Grünbaum, "A study of Fourier space methods for limited angle image reconstruction," Numer. Funct. Anal. and Optiomiz. 2, 31 (1980).
82. I. Kadar, "A robustized vector recursive stabilizer algorithm for image restoration," Information and Control 44, 320 (1980).
83. R. Goutte, R. Prost, and A. Georges, "Déconvolution numérique avec prolongement spectral applications aux signaux et aux images," Analysis 8, 6 (1980).