LA-4515-MS
c.3

LOS ALAMOS SCIENTIFIC LABORATORY
of the
University of California
LOS ALAMOS • NEW MEXICO

An Improved Technique for Using the

Fast Fourier Transform to Solve

Convolution-Type Integral Equations

LOS ALAMOS SCIENTIFIC LABORATORY
of the
University of California
LOS ALAMOS • NEW MEXICO

An Improved Technique for Using the Fast Fourier Transform to Solve Convolution-Type Integral Equations

by

B. R. Hunt

An Improved Technique for Using the
Fast Fourier Transform to Solve Convolution-Type
Integral Equations

by

B. R. Hunt

ABSTRACT

A technique presented by Phillips and Twomey for solving integral
equations is discussed and an extension of the technique to the
use of the fast Fourier transform is presented.

Introduction

The convolution-type integral equation,

$$g(t) = \int_0^t h(t-t_1)\, f(t_1)\, dt_1 \qquad , \qquad (1.)$$

occurs frequently in many problems in electrical engineering. For
example, since the input-output relationships for a time-invariant,
linear system can be formulated as in equation (1.), identification
of the system's impulse response, $h(t)$, from a record of input and output
functions, $f(t)$ and $g(t)$, requires the solution of the above integral
equation. An equivalent problem is to find the system input, $f(t)$,
given the system response, $h(t)$, and the system output, $g(t)$. This
problem is frequently referred to as "deconvolution" [1]. These
problems are difficult to solve because of the ill-conditioned nature
of integral equations of the first kind [2]. In this paper we first
review a technique which can be used to overcome many of the problems
associated with integral equations of the first kind. The technique
was originally investigated by Phillips [2] and Twomey [3, 4] and
requires matrix inversion of a set of modified least-squares normal
equations. However, in the special case of convolution-type integral
equations we demonstrate how to adapt the technique of Phillips and
Twomey to application of the fast Fourier transform. This leads to
great improvements in both the time required to compute solutions
and the number of points that can be used in the computation of the
solution, yet still retains some of the computational advantages
of the Phillips and Twomey technique.

Constrained Least Squares Solutions

In solving equation (1.), we must approximate it in the form of
a discrete sum. Choosing an interval Δt we use the simplest approximation

and have:

$$g(k\Delta t) \cong \sum_{j=0}^{k} h((k-j)\Delta t) f(j\Delta t)\Delta t \quad .$$
(2.)

It is common to let the interval $\Delta t = 1$, and then represent equation (2.) in simpler notation as:

$$g(k) \cong \sum_{j=0}^{k} h(k-j) f(j) \quad .$$
(3.)

The formulation of equation (1.) explicitly assumes "causal" functions that are identically zero for $t < 0$. In addition we assume that $f(t)$ and $h(t)$ go to zero after some finite interval. This means that $g(t)$ goes to zero in some finite interval also. If, using the interval Δt, the sequence $f(j)$ consists of a total of a non-zero points and the sequence $h(j)$ consists of a total of b non-zero points, then the sequence $g(k)$ contains a total of a+b-1 non-zero points. Thus equation (3.) is valid for k = 0, 1, 2, ..., a+b-2. Note that by the upper and lower limits of the sum in (3.) we are assuming the three sequences are extended by zeroes for values of k or j outside of the intervals of non-zero extent. This eliminates the complicated writing of the convolution sum with variable upper and lower limits, as is frequently seen [5].

The difficulty in solving equation (3.) for the unknown sequence $f(j)$ lies in the poor behavior of the integral equation as an operator mapping $f(t)$ into $g(t)$. As demonstrated by Phillips [2], it is always possible to add a finite quantity to any solution $f(t)$ and add only an infinitesimal amount to the observed function $g(t)$. Hence, if the functions $h(t)$ and $g(t)$ contain any error at all, the solution $f(t)$ will be unreliable. Typically the solutions one obtains are unstable

and oscillate wildly between positive and negative values [2]. To
further aggravate matters, it is impossible for the functions h(t)
and g(t) to be free from error. In the case where h(t) and g(t)
are obtained from measurements on a real system, there will always
be errors associated with the measurements. In the case where h(t)
and g(t) are functions defined in a closed-form expression, errors
are still encountered. Equation (3.) is only an approximation
to the integral equation (1.), and the errors resulting from the
approximation can be treated as errors in the actual functions h(t)
and g(t).

Given the existence of errors, equation (3.) can be written in
the form:

$$d(k) \;=\; g(k) \;+\; \in(k) \;=\; \sum_{j=0}^{k} h(k-j)f(j) \;, \tag{4.}$$

where $\in(k)$ is an error term that accounts for the approximation
of the integral in (1.), or for errors in the measurement of h(t)
and g(t), or both. In this form we are making the statement that
the data one has is actually a sequence d(k) composed of the true
sequence g(k) plus an unknown (and unknowable) error term $\in(k)$.

As pointed out by Phillips [2], the existence of errors converts
the solution of equation (4.) for f(j) from a problem with a unique
solution into a problem with a family $\mathcal{F}$ of solutions. There is
a different solution sequence f(j) corresponding to every possible
combination of g(k) and $\in(k)$, and the family of solution sequences,
$\mathcal{F}$, becomes infinite. Phillips proposed, therefore, that to solve
the problem one should impose a constraint on the solution that
would act to select a specific solution from the family $\mathcal{F}$ of solutions.
Since solutions to (4.) are usually wildly oscillatory, Phillips
suggested that the constraint to impose was that the solution sequences

4

be "smooth" in the sense of having minimum second differences over the family $\mathcal{F}$ of solutions. That is, the solution $\hat{f}(j)$ should satisfy:

$$\hat{f}(j) = \underset{f(j) \in \mathcal{F}}{\text{Min}} \sum_{j=0}^{a-1} (f(j-1) - 2f(j) + f(j+1))^2 \qquad (5.)$$

Since the minimum of the second difference operation formulated in equation (5.) is achieved by any constant sequence, Phillips also suggested another constraint in terms of the error, $\in(k)$. Although the actual error is unknown, one usually knows something about the statistics of the errors. For example, if the errors have zero mean with a variance σ^2, then one knows that:

$$\sum_{k=0}^{a+b-2} \in(k)^2 \cong (a+b-2)\sigma^2 = e \quad , \qquad (6.)$$

where the approximation is based on estimation of the variance from a+b-1 samples, and could be posed in terms of a suitable confidence interval. Assuming knowledge in the form of equation (6.), Phillips proposed that besides the constraint on smoothness, one should also choose a solution whose residual was on the same order as e . That is, if we define the residual of a solution $\hat{f}(j)$ as

$$\delta(k) = \sum_{j=0}^{k} h(k-j)\, \hat{f}(j) - d(k) \quad , \qquad (7.)$$

then Phillips suggested choosing $\hat{f}(j)$ such that (5.) is satisfied and the residual has the property:

$$\sum_{k=0}^{a+b-2} \delta(k)^2 = e \quad . \qquad (8.)$$

The solution of (4.) in terms of the constraints imposed by equations (5.) and (6.) was formulated by Twomey [4] as follows. We write equation (4.) as a matrix equation:

$$\underset{\sim}{d} = H \underset{\sim}{f} \quad .$$

The vectors $\underset{\sim}{d}$ and $\underset{\sim}{f}$ correspond to the sequences $d(k)$ and $f(j)$. The matrix H is of dimension $a+b-1$ by a . The kj^{th} element of H is $h(k-j)$, and $h(k-j) = 0$ for $k-j < 0$ or $k-j \geq b$. The constraint on the residual in equation (8.) can then be written as:

$$(H\underset{\sim}{\hat{f}}-\underset{\sim}{d})^{T} (H\underset{\sim}{\hat{f}}-\underset{\sim}{d}) = e \quad . \tag{9.}$$

To include the second-difference operations of the constraint in equation (5.) we consider the matrix C which is of size $a+2$ by a, where:

$$C = \begin{bmatrix}
1 & . & . & . & . & . & . \\
-2 & 1 & & & & & \\
1 & -2 & 1 & & & & \\
. & 1 & -2 & 1 & & & \\
. & & & . & & & \\
. & & & & . & & \\
. & & & & & . & \\
. & & & & 1 & -2 & 1 \\
. & & & & & 1 & -2 \\
. & & & & & & 1
\end{bmatrix}$$

and all other elements of C are zero. It is evident that multiplying C by a vector $\underset{\sim}{f}$ results in another vector which is the second difference sequence of the sequence $f(j)$ represented as a vector $\underset{\sim}{f}$. The right-hand side of equation (5.) is thus represented by the product

$$\underset{\sim}{f}^{T} C^{T} C \underset{\sim}{f} \quad . \tag{10.}$$

The problem at hand is to find a solution vector, $\hat{f}$, which minimizes the quadratic form of (10.), with a residual that satisfies equation (9.). Twomey formulated this problem in terms of the minimization of a criterion function:

$$\phi(f) = (Hf-d)^T(Hf-d) + \gamma(Cf)^T Cf \quad , \tag{11.}$$

where γ is a Lagrange multiplier [3, 4]. By the usual procedure of differentiating with respect to f the solution vector $\hat{f}$ can be computed as:

$$\hat{f} = (H^TH + \gamma C^TC)^{-1} H^T d \tag{12.}$$

The parameter γ is determined iteratively; once a value of e is known equation (12.) is solved for different values of γ until the constraint of equation (8.) is satisfied on the magnitude of the residual of the solution [3, 4].

Generalizations of this technique, by using other forms for the matrix C^TC, are contained in the paper by Twomey [4]. We turn now to the application of the fast Fourier transform to the computation of equation (12.).

Fast Fourier Transform Solution

The discrete Fourier transform of a sequence f(j) is defined as:

$$F(n) = \sum_{j=0}^{N-1} f(j) \, e^{\frac{-i2\pi}{N} jn} \quad , \tag{13.}$$

for n = 0, 1, 2, ..., N-1 .

The fast Fourier transform is a well-known technique for rapidly computing F(n) [6]. The greatest uses of the fast Fourier transform

have been in the rapid computation of convolutions. The discrete
circular convolution of two sequences is defined as:

$$g(k) = \sum_{j=0}^{N-1} h(k-j)f(j) \qquad , \qquad (14.)$$

where the sequences $g(k)$, $h(j)$, $f(j)$ are all assumed to be periodic
with period N. It is easily shown [7] that the discrete Fourier trans-
form of both sides of equation (14.) gives the transform product:

$$G(n) = H(n)F(n) \quad ,$$
$$\text{for } n = 0, 1, 2, \ldots, N-1 \quad , \qquad (15.)$$

where $G(n)$, $H(n)$ and $F(n)$ are the discrete Fourier transforms of
the sequences $g(k)$, $h(j)$ and $f(j)$. If we make the additional
assumption that the sequences $g(k)$, $h(j)$ and $f(j)$ are real-valued,
then it can also be shown [7] that the convolution

$$g(k) = \sum_{j=0}^{N-1} h(j-k)f(j) \qquad (16.)$$

$$\text{for } k = 0, 1, 2, \ldots, N-1 \quad ,$$

can be transformed into the products

$$G(n) = H(n)^* F(n) \qquad , \qquad (17.)$$
$$\text{for } n = 0, 1, 2, \ldots, N-1 \quad ,$$

where * denotes complex conjugate.

Circular convolutions differ from ordinary discrete convolutions,
such as given in equation (3.), by the assumption of periodicity in

the sequences. However, the convolution of sequences which are not periodic can be computed with circular convolutions of periodic sequences by suitable extension of the non-periodic sequences with zeroes and the formation of periodic sequences from the extended sequences. The technique is well-known and consists of a proper choice of N and addition of sufficient zeroes to make periodic sequences of period N[6, 8]. The fast Fourier transform can then be applied to the circular convolution of the extended sequences which will be equal to the non-circular convolution of the extended sequences, in the form of equation (3.), for k = 0, 1, 2, ..., N-1. Since the technique is well documented in the literature, we dwell on it no further. The following results using convolutions are made with the assumption of performing non-circular convolutions by proper circular convolutions.

It is possible to write the matrix equation (12.) in the form:

$$(H^TH + \gamma \, C^TC) \, \hat{\underset{\sim}{f}} = H^T\underset{\sim}{d} \tag{18.}$$

We wish to now rewrite equation (18.) in the classical subscript form of notation. Since the matrices H and C were derived from discrete convolutions, the subscript form of the equation is:

$$\sum_{k=0}^{a+b-2} \sum_{j=0}^{a-1} h(k-r)h(k-j) \, \hat{f}(j) + \gamma \sum_{p=0}^{a+t-2}\sum_{j=0}^{a-1} c(p-r)c(p-j)\hat{f}(j)$$
$$= \sum_{k=0}^{a+b-2} h(k-r)d(k) \quad , \tag{19.}$$

for r = 0, 1, 2, ..., a-1 .

The sequence c(j) is the convolution operator that computes second differences, i.e., $c(j) = \left\{ 1, \, -2, \, 1 \right\}$ and t = 3.

In equation (19.) we have double convolutions with the sequences $h(j)$ and $c(j)$. The summation on j convolves $h(j)$ and $c(j)$ with the sequence $f(j)$ and this is followed by another convolution, but on the index k and with the results of the previous convolution. We indicate this double convolution with intermediate sums:

$$\sum_{k=0}^{a+b-2} h(k-r)\alpha(k) + \gamma \sum_{p=0}^{a+t-2} c(p-r)\,\beta(p) = \sum_{k=0}^{a+b-2} h(k-r)d(k) \ , \qquad (20.)$$

where:

$$\alpha(k) = \sum_{j=0}^{a-1} h(k-j)\hat{f}(j) \ ,$$

$$\qquad (21.)$$

$$\beta(p) = \sum_{j=0}^{a-1} c(p-j)\hat{f}(j) \ .$$

With these convolutions clearly stated we wish to compute them by the fast Fourier transform. We choose a number N and extend the sequences with zeroes such that the circular convolutions of the sequences are equivalent to the non-circular convolutions over the period N. Then equations (20.) and (21.) become:

$$\sum_{k=0}^{N-1} h(k-r)\alpha(k) + \gamma \sum_{p=0}^{N-1} c(p-r)\beta(p) = \sum_{k=0}^{N-1} h(k-r)d(k), \qquad (22.)$$

$$\alpha(k) = \sum_{j=0}^{N-1} h(k-j)\hat{f}(j) \ ,$$

$$\qquad (23.)$$

$$\beta(p) = \sum_{j=0}^{N-1} c(p-j)\hat{f}(j) \ .$$

We now take the discrete Fourier transform of both sides of equations (22.) and (23.). The transforms of the sequences we denote by capital letters, as is usual, and have:

$$H(n)^* A(n) + \gamma\, C(n)^* B(n) = H(n)^* D(n),$$

$$A(n) = H(n)\, \hat{F}(n) \ ,$$
$$B(n) = C(n)\, \hat{F}(n) \ .$$

We perform the indicated substitutions, solve for F(n) and the result is:

$$\hat{F}(n) = \frac{H(n)^*\, D(n)}{H(n)^*\, H(n) + \gamma\, C(n)^* C(n)}$$

$$= \frac{D(n)}{H(n) + \gamma\, \dfrac{C(n)^*\, C(n)}{H(n)^*}} \qquad (24.)$$

for $n = 0, 1, 2, \ldots, N\text{-}1$.

The solution $\hat{f}(j)$ is then computed from the inverse transform:

$$\hat{f}(j) = \frac{1}{N} \sum_{n=0}^{N-1} \hat{F}(n)\, e^{\frac{i2\pi}{N} jn} \qquad .$$

Equation (24.) can be used to solve the problem by iterations on γ until the constraint of equation (8.) is satisfied. This would be directly analogous to the technique of Phillips and Twomey. However, we can use relationships in the frequency domain to solve for the parameter γ directly. We recall that the residual constraint of equation (8.) is given as:

$$\sum_{k=0}^{a+b-2} \delta(k)^2 = e$$

In the case of the extension of sequences by zeroes to develop equations (22.) and (23.), the residual sequence extends to a total of N points as well:

$$\sum_{k=0}^{N-1} \delta(k)^2 = e$$

By Parseval's theorem the transform of the residual sequence $\Delta(n)$, can be related to $\delta(k)$ [7.]:

$$\sum_{k=0}^{N-1} \delta(k)^2 = \frac{1}{N} \sum_{n=0}^{N-1} |\Delta(n)|^2 = e \qquad (25.)$$

However, it is evident from the transform of equation (7.) that $\Delta(n)$ can be computed as:

$$\Delta(n) = H(n) \hat{F}(n) - D(n)$$

We substitute for $F(n)$ from equation (24.) and after manipulation have:

$$\Delta(n) = \left(\frac{-1}{\frac{1}{\gamma} \frac{H(n)^* H(n)}{C(n)^* C(n)} + 1} \right) D(n) \qquad (26.)$$

In equation (26.) $D(n)$ is the only complex quantity, since the products $H(n)^* H(n)$ and $C(n)^* C(n)$ are real-valued sequences. Thus we can substitute into equation (25.) from equation (26.) and have:

$$\frac{1}{N} \sum_{n=0}^{N-1} \frac{D(n)^* D(n)}{\left(\frac{1}{\gamma} \frac{H(n)^* H(n)}{C(n)^* C(n)} + 1 \right)^2} = e \qquad (27.)$$

In equation (27.) only γ is unknown. It can be computed iteratively, i.e., try a value for γ and then increment γ until the left-hand side of equation (27.) is as close to e as desired. Since all the computations are in the transform domain, it is not necessary to repeatedly solve a problem such as equation (12.) or (25.) in the process of varying γ until the constraint of equation (8.) is satisfied.

Discussion

The chief advantages of the transform method for solving the convolution-type integral equation are reduced computing time and reduced computer storage. The computing time for the matrix inversions of the Phillips and Twomey method is nominally proportional to N^3, whereas the transforms can be performed in time proportional to $N \log_2 N$. In addition, the storage required for the matrices of the Phillips and Twomey method is N^2. The transforms require only the storage of N points if the so-called "in-place" version of the fast Fourier transform is used. The disparity in computing time requirements was demonstrated by test examples comparing the Phillips and Twomey method with the transform method. Using 10 to 20 points in the sequences $g(k)$ and $h(j)$, the Phillips and Twomey method resulted in elapsed computing time on the same order of magnitude as the time required for the transform method using sequences of 500 to 1000 points. Both examples were coded in FØRTRAN IV for the CDC 6600 at the Los Alamos Scientific Laboratory, and off-the-shelf routines for matrix inversion and the fast Fourier transform were used. Actual computing times were a function of the initial guess for γ and the number of iterations consequently required.

The major disadvantage to the transform method is a loss in generality from the Phillips and Twomey method. First, the Phillips and Twomey method allows the constraint on the solution to be quite general. The application of discrete Fourier transforms, as shown herein, requires that the constraint be formulated as a convolution operation, so that

the transformation of certain of the matrix products can be performed. Convolution-type constraints are only a small subclass of constraints in the Phillips and Twomey method, which is general enough to allow the constraint to be different over different parts of the solution. In such a case the matrix product $C^T C$ in equation (10.) can be replaced by a general constraint matrix, say V, which embodies the desired constraint. Second, the transform method is applicable only to Volterra type integral equations, i.e., those in which the upper limit of integration in equation (1.) is a variable and not fixed as a constant. If both upper and lower limits of the integration are fixed, the Phillips and Twomey method is still applicable but the transform method is not, except for the special case where the interval between the fixed endpoints extends sufficiently beyond the nonzero extent of the functions in the equation.

The transform technique appears to be of greatest utility in dealing with data generated by real-world measurements of actual data. Digital technology has made common the collection of hundreds or thousands of experimental data points in sampled-data form. If the data must be used in the solution of a convolution-type integral equation, the Phillips and Twomey method is not easily applied to such masses of data, given the storage requirements and computing time. The transform method is not so burdened, however, and is a natural method for the solution of convolution-type integral equations involving the large quantities of data often generated in experimental measurements. Investigation is currently underway at Los Alamos Scientific Laboratory in applying the technique to deconvolve optical spectroscopy data, solar spectra observations and x-ray images.

Several extensions of the transform method are immediately obvious. First, in the formulation of equation (3.) the interval $\Delta t = 1$ was chosen to simplify the notation. Any other integration interval is easily included in the development herein by multiplying the sequence $h(j)$ by the actual value of Δt prior to performing the indicated transforms.

A second extension of the transform method of solution can be seen by examination of equation (24.). The term $C(n)^* C(n)$ is the transform of the sequence $c(p)$ which performs second-differencing of the solution $\hat{f}(j)$. Since $c(p) = \left\{ 1, -2, 1 \right\}$, it is evident that $C(n)$ will predominantly contain high frequencies. Thus, the term $C(n)^* C(n)$ in equation (24.) is actually a digital filter which acts upon the data in such a way as to produce a solution which is smoothest in the sense of minimum second differences. We note also that $C(n)^* C(n)$ is a zero-phase filter, i.e., the transfer function is real, since the product $C(n)^* C(n)$ is always real. Interpreting the term $C(n)^* C(n)$ as the transfer function of a digital filter leads to the generalization of replacing the second difference filter by an arbitrary transfer function, say $W(n)$. Equation (24.) then becomes:

$$\hat{F}(n) = \frac{D(n)}{H(n) + \gamma \frac{W(n)^* W(n)}{H(n)^*}} \qquad (28.)$$

The desirability of being able to replace the second-difference filter by an arbitrary filter is seen when data generated by real-world measurements is considered. For example, suppose the data on hand had been corrupted by noise that was concentrated in a narrow bandwidth of the frequency spectrum. It would not be necessary to filter this band of the spectrum prior to solving the associated convolution-type integral equation. The filtering and solution could be obtained simultaneously. It is also possible to perform two kinds of filtering simultaneously. For example, arbitrary filtering and second-difference filtering could be achieved by:

$$F(n) = \frac{D(n)}{H(n) + \gamma \frac{W(n)^* W(n) + C(n)^* C(n)}{H(n)^*}} \qquad . \quad (29.)$$

We note that since $W(n)$ appears in the denominator of (28.) and (29.) filter design must deal with reciprocals, i.e., a low-pass filtering of the solution $F(n)$ is achieved with a high-pass design for $W(n)$, a band-stop filtering corresponds to a band-pass design for $W(n)$, etc.

A third extension is suggested by Twomey. If one knows a-priori that the solution should assume a particular shape, then the method of Phillips and Twomey assumes the form:

$$\hat{f} = (H^TH + \gamma I)^{-1} (H^Td + \gamma u) \quad , \qquad (30.)$$

where I is the identity matrix and u is the vector that represents the a-priori shape of the solution [4.]. It is easy to show that the frequency domain equivalent of this solution is:

$$\hat{F}(n) = \frac{H(n)^* D(n) + \gamma U(n)}{H(n)^* H(n) + \gamma} \quad . \qquad (31.)$$

A final extension is to let the sequence $h(j)$ be an impulse, $h(0) = 1$, $h(j) = 0$, $j \neq 0$. Then the solution of equation (24.) takes the form:

$$\hat{F}(n) = \frac{D(n)}{1 + \gamma C(n)^* C(n)} \quad . \qquad (32.)$$

This form of solution corresponds to filtering the data in such a way as to give a solution which is smoothest in the sense of second differences and satisfies the constraint of equation (8.).

It is instructive to consider the form of equation (24.) for the special case when $\gamma = 0$. We see that this corresponds to the simple solution of equation (3.) by transforming the sequences and dividing them. Such is the method for solving convolution-type integral equations suggested in a paper by Cooley, Lewis and Welch [8]. This method, however, is often too simple, since it lacks the second-difference filtering of the method of Phillips and Twomey which elimates wildly oscillating solutions. For example, in the majority of real-world systems, the impulse response function is essentially a low-pass filter and the higher frequencies in the input signal are severely

attenuated. Attempting to deconvolve the input signal by simply dividing the input transform by the system response transform results in boosting the high frequencies of the input by a large amount. Unfortunately, most of the noise in the input signal is usually located in the high-frequencies. The deconvolved output is usually an un-stable oscillating function by virtue of the amplification of the noise that results. The author's experience has generally been that the solutions one obtains in applying the straight-forward division of transforms, as suggested by Cooley, Lewis and Welch, are too noisy to be of any utility. Consequently, the transform method developed herein offers the speed and storage advantages of the fast Fourier transform and also retains the properties of smooth, well-behaved solutions of the Phillips and Twomey method. In this sense, the transform method of this document is an improved technique over the simpler solution of dividing transforms as advocated by Cooley, Lewis, Welch [8].

Examples

Figure 1 shows a sequence generated from the expression

$$f(j) = \exp\left(-\left(\frac{j-500}{100}\right)^2\right)$$

for $j=0, 1, \ldots, 1023$.

This sequence was then convolved with a square pulse defined as:

$$h(j) = \begin{cases} 1 \text{ for } 0 \leq j \leq 249 \\ 0 \text{ for } 250 \leq j \leq 1023 \end{cases}.$$

The result of the convolution was then added to a random noise sequence generated by random samplings of a uniform distribution on the interval $(-.05, .05)$. Figure 2 shows the result. The effect of noise contamination

is so small as to be undetectable by unaided eye inspection of Figure 2.
Yet it is more than sufficient to ruin the solution using a conventional
division of transforms, as suggested by Cooley, Lewis and Welch [8].

Figure 3 shows the solution of equation (24.) with $\gamma = 0$. The
noise added to the to the convolution dominates the solution. Figure 4
shows the solution obtained by applying equation (24.) and letting
γ be iterated so as to satisfy the constraint of equation (8.). The
superior quality of Figure 4 is evident. Table I is a partial
tabulation of the original function and the solution.

As a second example, a sum of two functions was used to generate
a sequence:

$$f_j = \exp\left(-\left(\frac{j-400}{75}\right)^2\right) + \exp\left(-\left(\frac{j-600}{75}\right)^2\right)$$

for $j = 0, 1, 2, \ldots, 1023$

This sequence was then convolved with the pulse sequence h_j defined
in the previous example. The sequence f_j is shown in Figure 5
and the resulting convolution is shown in Figure 6. The two separate
peaks have been smeared into one. Noise was then added by random
samples from a uniform distribution on the interval (-5, 5). The
result is shown in Figure 7. Solution of the convolution for $\hat{f}$ with
$\gamma = 0$ is shown in Figure 8. The noise completely obliterates meaning-
ful details. Allowing the program to iterate the parameter γ until
equation (8.) is satisfied produced the result shown in Figure 9.
Table 2 tabulates the original sequence and the solution. The greater
noise content in the data does not give the better restoration shown
in the previous example. But the shape and magnitude of the two peaks
is adequately restored. The restoration of separate details, smeared
together in a convolution, is of practical utility in many problems
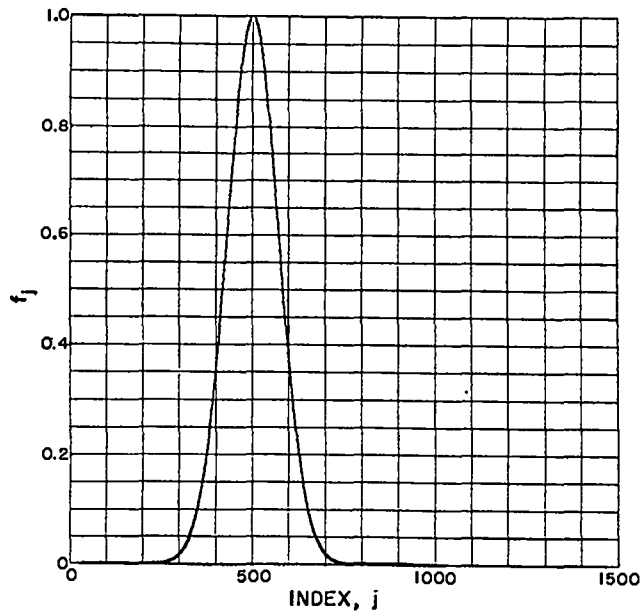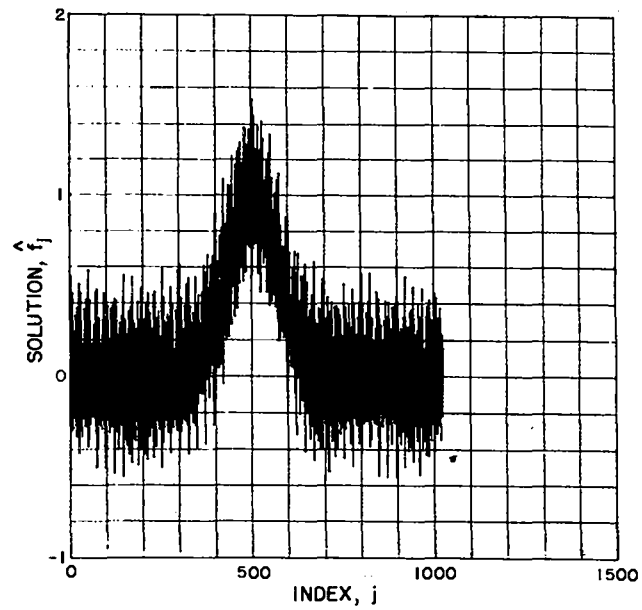in optics, spectroscopy, etc.

Figure 1: Exponential Sequence

Figure 2: Convolved Sequence with Noise Added
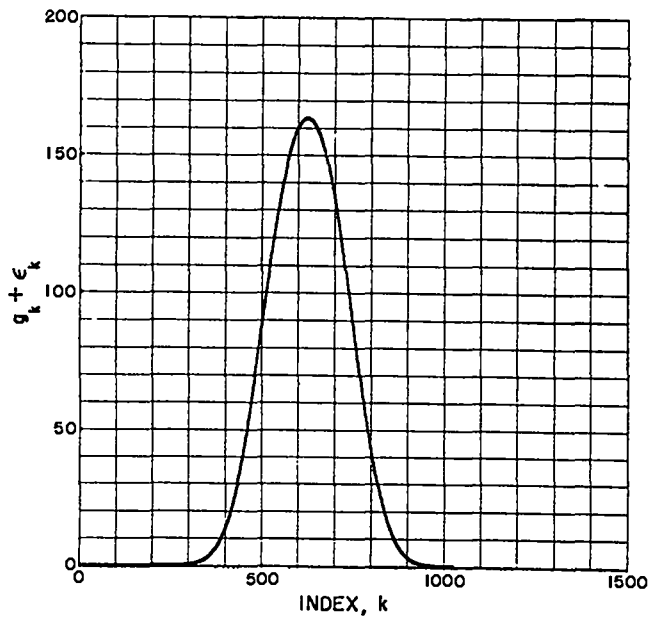

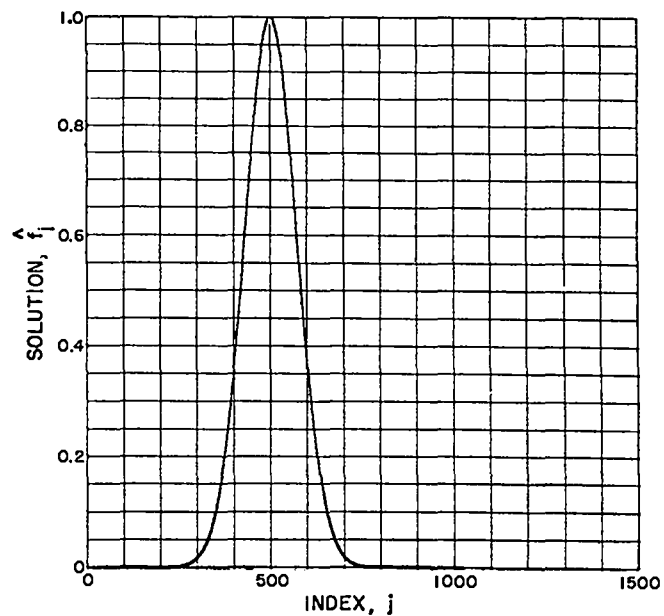Figure 3: Solution for $\gamma = 0$

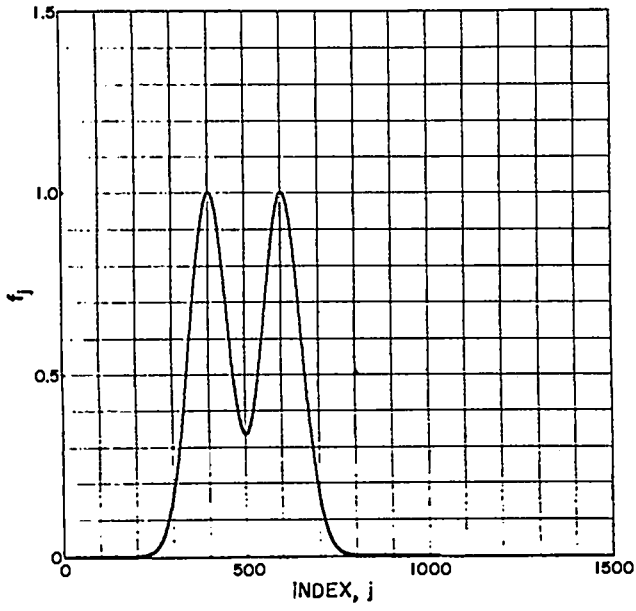Figure 4: Solution after Iteration on γ

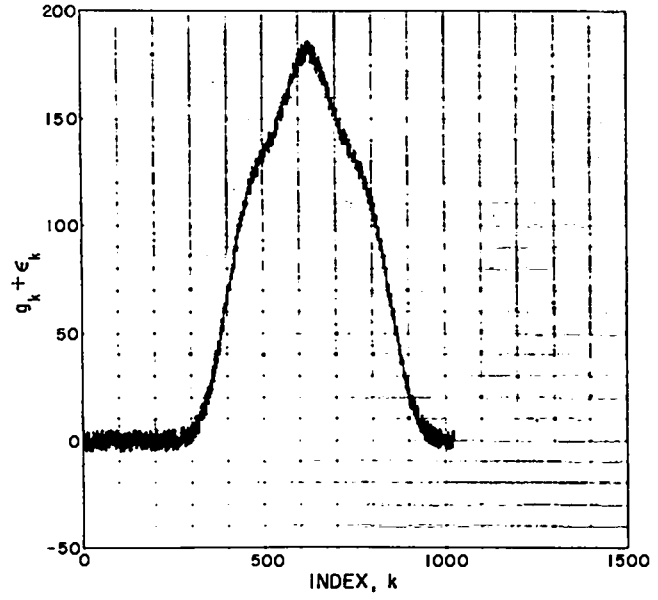Figure 5: Double Peaked Sequence

Figure 7: Noise Added
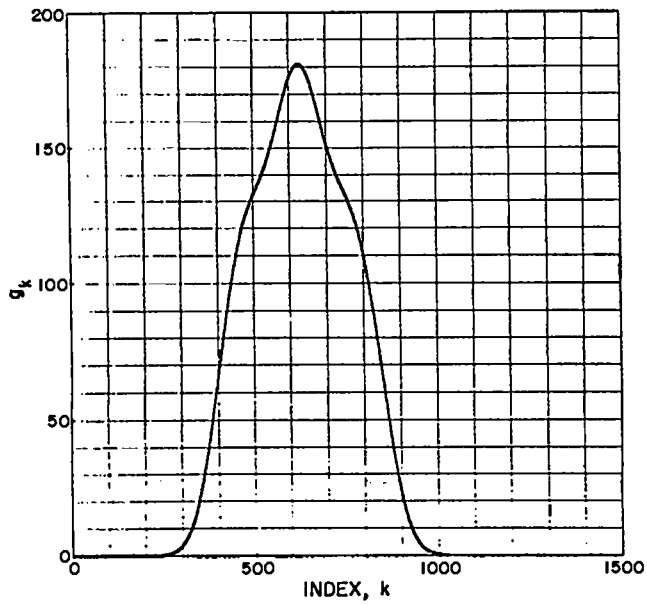
Figure 6: Convolved Sequence
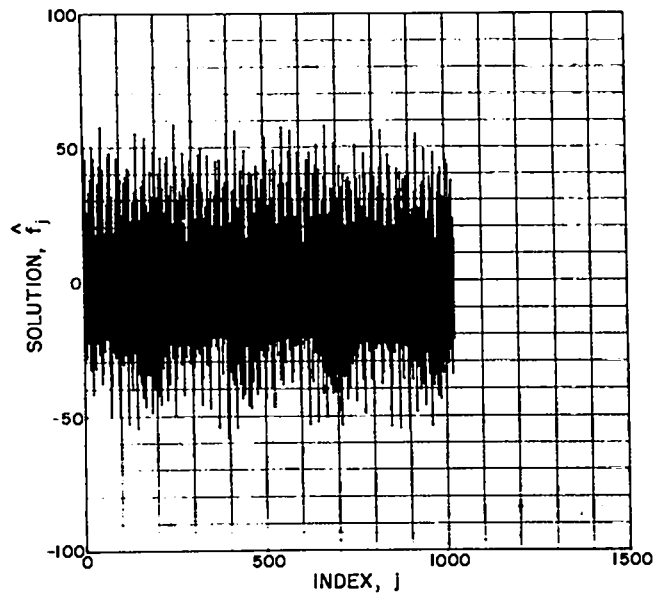
Figure 8: Solution for $\gamma = 0$

20
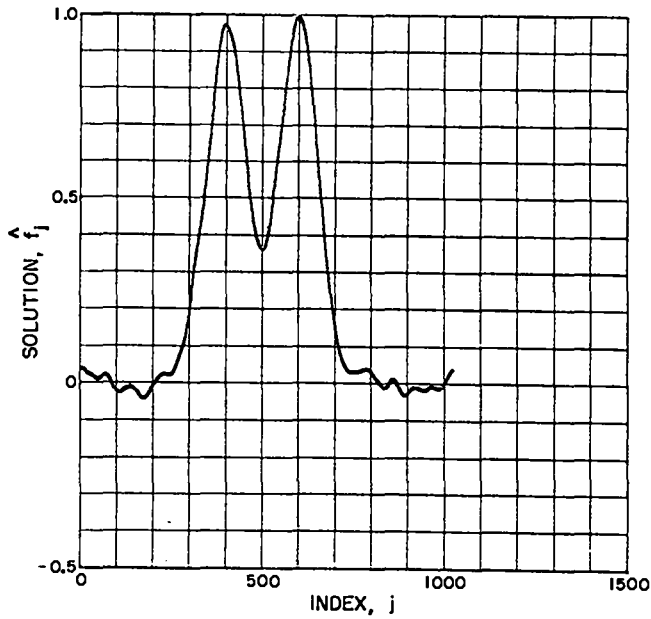
Figure 9: Solution after Iteration on γ

	Table 1			Table 2	
j	f_j	$\hat{f}_j$	j	f_j	$\hat{f}_j$
50	.00000	-.00001	50	.00000	.01020
100	.00000	.00011	100	.00000	-.02219
150	.00000	.00037	150	.00001	-.01716
200	.00012	-.00017	200	.00082	-.00639
250	.00193	-.00119	250	.01832	.02299
300	.01832	.01858	300	.16901	.21030
350	.10540	.10546	350	.64120	.61192
400	.36788	.36793	400	1.00082	.96584
450	.77880	.77855	450	.65950	.65882
500	1.00000	.99949	500	.33803	.36236
550	.77880	.77892	550	.65950	.68809
600	.36788	.36822	600	1.00082	.98263
650	.10540	.10565	650	.64120	.62979
700	.01832	.01809	700	.16901	.15314
750	.00193	.00133	750	.01832	.02913
800	.00012	.00017	800	.00082	.02862
850	.00000	.00050	850	.00001	-.01025
900	.00000	.00028	900	.00000	-.02501
950	.00000	-.00024	950	.00000	-.01748
1000	.00000	-.00073	1000	.00000	-.01035

References

[1] R. W. Shafer, "Echo Removal by Discrete Generalized Linear Filtering," Report No. 466, Research Laboratory of Electronics, MIT, Cambridge, February, 1969.

[2] D. L. Phillips, "A technique for the numerical solution of certain integral equations of the first kind," J. of ACM, Vol. 9, pp. 84-97, January, 1962.

[3] S. Twomey, "On the numerical solution of Fredholm integral equations of the first kind by the inversion of the linear system produced by quadrature," J. of ACM, Vol. 10, pp. 97-101, January, 1963.

[4] S. Twomey, "The application of numerical filtering to the solution of integral equations encountered in indirect sensing measurements," J. Franklin Institute, Vol. 279, pp. 95-109, February, 1965.

[5] T. J. Healy, "Convolution revisited," IEEE Spectrum, Vol. 6, pp. 87-93, April, 1969.

[6] W. T. Cochran, et. al., "What is the fast Fourier transform?" IEEE Trans. Vol. AU-15, pp. 45-55, June, 1967.

[7] J. W. Cooley, P.A.W. Lewis, P. D. Welch, "The finite Fourier transform," IEEE Trans., Vol. AU-17, pp. 77-85, June, 1969.

[8] J. W. Cooley, P.A.W. Lewis, P. D. Welch, "Application of the fast Fourier transform to the computation of Fourier integrals, Fourier series, and convolution integrals," IEEE Trans., Vol. AU-15, pp. 79-84, June, 1967.