ONE-DIMENSIONAL PHASE RETRIEVAL BY DIRECT METHODS

Corneliu Rusu¹,² and Jaakko Astola¹

¹Tampere International Center for Signal Processing,
Tampere University of Technology,
P.O.Box 553, FIN-33101 Tampere, FINLAND, [Corneliu.Rusu;Jaakko.Astola]@tut.fi
²Faculty of Electronics, Telecommunications and Information Technology
Technical University of Cluj-Napoca,
Baritiu 26-28, RO-400027 Cluj-Napoca, ROMANIA, Corneliu.Rusu@bel.utcluj.ro

ABSTRACT

In this paper we reconsider one dimensional phase retrieval problem by direct methods. First we review previous results obtained by signal processing groups, then we present the main methods for solving directly the one-dimensional discrete problem. Results of simulations are also shown.

1. INTRODUCTION

For both continuous-time and discrete-time signals, the magnitude and phase of the Fourier transform are, in general, independent functions, i.e., the signal cannot be recovered from knowledge of either alone [1]. Since the recovering problem does not have a unique solution in general, researchers have tried many ways by providing information of the signal a priori or constraining the properties of the signal [2]. Nevertheless, in certain cases relationships may exist between these components leading to certain signal reconstruction methods using only partial information in frequency domain [3]. The condition under which such signal reconstruction problems have unique answer is known [4]. Often these solutions are closed form expressions in terms of the given partial knowledge, but they still may be computationally intractable.

Signal reconstruction from Fourier transform magnitude has been called phase retrieval [5]. The term comes from the fact that the Fourier phase is not known and the signal should be reconstructed [6]. As an example, in a Fourier transform coding system, both the magnitude and the phase are usually coded and transmitted. However, for signals which can be recovered from only the magnitude, unnecessary redundancy is inherent in the coder. Therefore for these signals it may be possible to realize a significant bit-rate reduction by simply coding the magnitude and then reconstructing the sequence at the receiver from the coded magnitude [7]. Besides coding, the phase retrieval problem has attracted considerable interest in recent years because of its importance in a variety of applications, including optical astronomy, microscopy [8], Fourier-transform spectroscopy, x-ray crystallography, particle scattering, speckle interferometry, lens testing, single-side communication, and design of radar signals [9].

In this paper we reconsider 1-D (one dimensional) phase retrieval problem by direct methods. First we recall 1-D discrete phase retrieval problem (Section 2), then we present the main methods for solving directly the one-dimensional discrete problem (Section 3). Results of simulations are also shown. We shall use the following notations:

- $z^*$ complex conjugate of $z$
- $r(n)$ autocorrelation of $x(n)$
- $\hat{r}(n)$ circular autocorrelation of $x(n)$
- $X(z)$ $z$-transform of $x(n)$
- $X(\omega)$ Fourier transform of $x(n)$
- $\mathcal{F}\{x(n)\}$ Fourier transform of $x(n)$
- $X(k)$ DFT of $x(n)$
- $S(z)$ $z$-transform of $r(n)$
- $\hat{S}(z)$ $z$-transform of $\hat{r}(n)$
- $c_x(n)$ cepstral coefficient of $x(n)$

Although nonminimum-phase phase retrieval is an interesting and promising subject, in this work we shall focus only on minimum-phase phase retrieval.

2. 1-D DISCRETE PHASE RETRIEVAL PROBLEM

The 1-D discrete phase retrieval problem deals with sequences having finite length and finite length spectrum. It follows that for the mentioned purpose, any kind of time or frequency aliasing has been properly avoided previously. Furthermore, the $z$-transform of involved sequences are always polynomials.

Let consider the signal with $M$-point support. Since its autocorrelation function has the support

$$[-(M-1), M-1]$$
the sampling of the Fourier transform magnitude at \( \omega_k = \frac{2\pi}{Nk} \), with \( N \geq 2M - 1 \) will be sufficient to extract autocorrelation without time-domain aliasing. Note that it is imperative to know the support or some bound of the support, otherwise one cannot specify the sampling requirements of Fourier transform magnitude [8].

Although certain constraints may be added according to application, the main 1-D discrete phase retrieval problem can be stated as follows [6]:

Let \( x(n) \) be a discrete signal of length \( N \) and let \( X(k) \) be its \( N \)-point Discrete Fourier Transform (DFT):

\[
X(k) = \sum_{n=0}^{N-1} x(n)e^{-j2\pi kn/N}, \quad k = 0, 1, \ldots, N-1. \tag{1}
\]

Given knowledge that only \( M \) consecutive values of \( x(n) \) differ from zero, i.e. \( x(n) \) has \( M \)-point support, and given the values of the DFT magnitudes \( |X(k)|, k = 0, 1, \ldots, N-1 \), determine \( x(n) \) or equivalently \( X(k) \).

Certain constraints should be imposed on the type of signal \( x(n) \), otherwise a zero-phase (Appendix A) or even random phase associated with given magnitudes can provide a valid signal \( x(n) \).

Traditionally real and imaginary parts of Fourier transforms \( X(\omega) \) are related each other when signal is causal. For the case of finite-length sequences where DFT is usually implemented to compute the spectrum, this leads to the concept of "causal" periodic sequence [10], i.e. a sequence which is zero on the second half. In such situation the number of constraints in time-domain equalizes the number of relationships between real and imaginary parts.

Moreover, if we need to compute phase from magnitude, we focus on the logarithm of Fourier transform \( \ln X(\omega) \) [11], thus the sequence should be minimum (maximum) phase, i.e. all zeros of \( X(z) \) should be inside (outside) unit circle [12].

The support of \( x(n) \) should satisfy:

\[
2M - 1 \leq N, \tag{2}
\]

otherwise we have an ill-posed problem. Indeed, the set of squares of the DFT magnitudes is the DFT of the circular autocorrelation \( \hat{r}(n) \) of \( x(n) \):

\[
\hat{r}(n) = \sum_{k=0}^{N-1} X(k)x((k-n)_N).
\]

On the other hand, the autocorrelation \( r(n) \) of \( x(n) \):

\[
r(n) = x(n) \ast x(-n),
\]

has \( 2M - 1 \) length, if \( x(n) \) has \( M \)-point support. If (2) is not satisfied, then \( \hat{r}(n) \) will be corrupted because of time-aliasing, and \( r(n) \) cannot be recovered from \( \hat{r}(n) \).

Furthermore, even when time-aliasing has been avoided, we note that the succession of samples of \( r(n) \) and \( \hat{r}(n) \) is not the same. Actually the first \( M \) samples of \( r(n) \) are shifted to obtain the last \( M \) samples of \( \hat{r}(n) \), and the last \( M \) samples of \( r(n) \) are equal with the first \( M \) samples of \( \hat{r}(n) \) with positive index:

\[
\hat{r}(n) = \begin{cases} 
  r(0) & n = 0 \\
  r(n) & n = 1, 2, \ldots, M - 1 \\
  r(n - N) & n = N - 1, \ldots, N - M + 1 \\
  0 & \text{otherwise.}
\end{cases}
\]

Thus in general, the z-transforms \( S(z) \) and \( \hat{S}(z) \) will not have common zeros [13].

Basically, a sequence is not uniquely defined by its magnitude, as is illustrated by the observation that a sequence convolved with any all-pass sequence will produce another sequence with the same magnitude [7]. Thus, without some assumptions about the sequence, the magnitude may, at best, uniquely specify a sequence to within an arbitrary all-pass factor. However, if some additional knowledge is available and under certain conditions the sequence may be uniquely defined by its magnitude. Such case is when all zeros of the finite length sequence are within the unit circle, the sequence is minimum-phase and thus it is uniquely defined to within a shifting factor by its magnitude.

Even we constrain \( x(n) \) to be a finite-length sequence, then other finite-length sequences having the same Fourier magnitude as \( x(n) \) may be generated by the process of zero flipping [2]. Actually, in the discrete phase retrieval problem, we have an overdetermined system of equations where the null space of the data matrix gives the desired flip coefficients [6]. Consequently, there are some ambiguities in phase-retrieval problem: if \( x(n) \) is a solution, then \( x^*(-n) \), \( cx(n) \) and \( x(n-b) \) are also solutions for any integer \( b \) and any complex number \( c \) having unity magnitude \( c = 1 \). If \( x(n) \) is a real sequence, then \( c = \pm 1 \). These are called the trivial ambiguities and they are associated solutions to a given solution \( x(n) \). Excluding these associated solutions [14], there are almost everywhere \( 2^M \) solutions to the discrete 1-D phase retrieval. If \( x(n) \) is a real sequence, the zeros must be chosen in complex conjugate pairs, so there are only \( 2^{M-1} \) solutions if all zeros of \( X(z) \) are complex [15]. Note that only one of these solutions is a minimum-phase sequence.

In order to find this minimum-phase solution when this is a real sequence, the most common approaches are iterative transform algorithms [3], which alternate between time and frequency domains. This type of algorithms can implement very easily time-domain constraints like compactness of the support. It has been observed that these algorithms fail to converge to a solution as they usually stagnate [16]. The iterative transform algorithms will be considered in the companion paper [17]. Alternative for solving the 1-D phase retrieval problem are:

1. finding the zeros of z-transform (Section 3.1);
2. Hilbert transform (Section 3.2);
3. computation of cepstral coefficients (Section 3.3);
4. solving linear systems of equations (Section 3.4).
3. APPROACHES TO PHASE RETRIEVAL

In the following we shall present the standard algorithms for direct 1-D phase retrieval.

The most straightforward method for solving the 1-D phase retrieval problem consists of finding the zeros of $z$-transform (Section 3.1) of the autocorrelation. From the all $2^M$ choices, we should select that one who has all zeros inside the unit circle.

One can reconstruct a sequence using Hilbert transform [10]. As we mentioned previously, there is a Hilbert transform relationship between the real and the imaginary parts of a causal signal. Moreover, the cepstrum of any minimum-phase signal is causal and therefore, the log-magnitude and the phase have a Hilbert transform correspondence. Since the sequence is of finite length, the resulting phase will be an approximation to the minimum-phase (Section 3.2).

A non-iterative algorithm for minimum-phase signal reconstruction, described in [18, 19], involves computation of cepstral coefficients first, and then of the phase function from the odd symmetric sequence of the cepstral coefficients. Using the given magnitude and reconstructed phase, the original minimum-phase signal can be obtained through the inverse DFT (Section 3.3).

Another method for minimum-phase phase retrieval begins by inverting the Fourier transform of autocorrelation and continues by solving linear systems of equations (Section 3.4).

To test all of these direct methods for discrete phase retrieval, in every case we shall generate random positive sequences of certain length, as inputs of algorithms. They may represent the values $|X(k)|^2$ of DFT square magnitudes corresponding to a certain sequence $x(n)$. However, this sequence may be not real and may have certain length which is greater than $M$. Thus we also generate symmetric random positive sequences as inputs of algorithms. Moreover, we consider as inputs the magnitudes of DFT of $L$-point length sequences, where $L = M, M + 1, \ldots, N$.

For every method we reconstruct the signal $x(n)$ and we verify if its DFT magnitudes are equal with initial data. Also we shall check if the obtained sequence is minimum-phase. A sensitive aspect is the compact support of the resulting minimum-phase sequence. Indeed, zero-phase or random phase can be easily joined the DFT magnitude, but the resulting reconstructed signal will have a $2M - 1$-point support instead of $M$-point length. Finally, an estimation of computational overload is mentioned for every method.

3.1. Minimum-phase phase retrieval by finding zeros

The method of minimum-phase phase retrieval by finding zeros is based on the result:

If $X(z)$ is $z$-transform of $x(n)$, then $X(z)X^*(1/z^*) = S(z)$ is the $z$-transform of $r(n)$. The zeros of $S(z)$ occur in conjugate reciprocal pairs. From every pair, one of the zeros should be selected to form $X(z)$. If the selected zeros are chosen inside the unit circle, we get a minimum-phase sequence.

Within the mentioned framework we run many times and the outcomes show that all the time the phase retrieval test passes, but minimum-phase sequence is not always detected. Sometimes we can have zeros on unit circle. This happens more likely for sequences having length greater than $M$, then for antisymmetric magnitudes, and finally for symmetric DFT magnitudes. The minimum-phase test passes whenever the sequence length is less or equalize $M$.

For $M = 9$ the results are presented in Figure 3.1. The number of runs was 1000 and it can be seen that phase retrieval has been always detected when the sequence length is less or is equal with 9. From computational load point of view we estimated the number of flops around 459000 for every iteration, and this number is very large. However there should be noticed that unlike other methods this can be used for both real and complex sequences.

With respect to sorting algorithm, a branch-and-bound algorithm may offer significant improvement over the exhaustive search [20]. Recently it was discovered that the zero configuration does not change much from one 1-D problem to its neighbor [15], and this may reduce the computational complexity.

3.2. Minimum-phase phase retrieval using Hilbert transform

The phase retrieval by Hilbert transform uses the fact that the log-magnitude and the phase have a Hilbert transform correspondence [10]:

$$j \arg X(k) = \frac{1}{M} \sum_{m=0}^{M-1} \ln X(k)V_M((k-m)_M),$$

for $0 \leq k \leq M - 1$, where

$$V_N(k) = \begin{cases} -j \cot \left( \frac{k\pi}{M} \right) & 0 < k < M - 1 \ k \ odd; \\ 0 & \text{otherwise}. \end{cases}$$

Figure 1. Number of minimum-phase sequences detected.
Since the sequence is of finite length, the resulting phase will be an approximation to the minimum-phase. Indeed, let us assume that \( X(z) \) is the \( z \)-transform of the finite-length sequence \( x(n) \). Then

\[
\frac{d \ln X(z)}{dz} = \frac{d X(z)}{dz} \frac{1}{X(z)}
\]

For a finite-length sequence, the last term of equation (3) should be infinite-length series. Using the differentiation property of \( z \)-transform, \( \ln X(z) \) will have also an infinite-length series. Thus the computed sequence will correspond to an aliased sequence [11, 21]:

\[
\hat{x}(n) = \sum_{k=-\infty}^{\infty} \hat{x}(n + kN),
\]

where

\[
\hat{x}(n) = \text{DFT}^{-1}\{\ln X(k)\}.
\]

Consequently the length of DFT should be as large as possible, in order to reduce the alias effect. This method provides approximations to the minimum phase that are useful in some practical situations [10].

Although Hilbert transform can be computed as a convolution [22], the FFT implementation of Hilbert transform has been used. Since filter length is greater than 30, the FFT implementation of convolution is known to be faster than implementation by convolution. The FFT implementation of Hilbert transform is described below:

1. Given the log-magnitude \( \ln X(k) \)
2. Compute the phase:
   \[
   \arg X(k) = \text{FFT}\{v(n) \cdot \text{FFT}^{-1}\{\ln X(k)\}\}
   \]
   where
   \[
   v(n) = \begin{cases} 
   0, & n = 0, N/2, \\
   1, & n = 1, 2, \ldots, N/2 - 1, \\
   -1, & n = N/2 + 1, \ldots, N - 1.
   \end{cases}
   \]

In our case to avoid alias effect for an 19-length sequence, we used an FFT of length 256 to assure that the computed sequence is not corrupted. However, there were cases when the computed sequence had zeros on unit circle. The magnitude test always passes, but difficulties appear to assure compact a support of the sequence.

A dramatic reduction of computational overload has been noticed in comparison with zero finding method: in Hilbert transform technique we needed only approximately 32500 flops per iteration. Unfortunately there is a drawback: unlike finding roots method which works for both real and complex sequences, Hilbert transform technique can be used only for real sequences.

Figure 2. The noniterative algorithm to reconstruct a minimum phase signal from its spectral magnitude function

### 3.3. Minimum-phase phase retrieval with cepstral coefficients

Another non-iterative algorithm for signal reconstruction has been described in [18, 19], and this method involves computation of cepstral coefficients first. After that the phase function is calculated from the odd symmetric sequence of the cepstral coefficients.

Let \( X(\omega) \) be the Fourier transform of a minimum-phase sequence \( x(n) \) of \( N \)-length. We have:

\[
X(\omega) = \sum_{n=0}^{N-1} x(n)e^{-j\omega n}
\]

and

\[
X(\omega) = [X(\omega)]e^{j\arg X(\omega)}.
\]

For a minimum-phase signal \( \ln X(\omega) \) and \( \arg X(\omega) \) can be written as

\[
\ln X(\omega) = c_x(0)/2 + \sum_{n=1}^{\infty} c_x(n)\cos \omega n; \quad (4)
\]

\[
\arg X(\omega) = -\sum_{n=1}^{\infty} c_x(n)\sin \omega n, \quad (5)
\]

where \( c_x(n) \) are called the cepstral coefficients of \( x(n) \).

Using the given magnitude and reconstructed phase, the original minimum-phase signal can be obtained through the inverse DFT (Figure 2), where

\[
c_o(n) = \begin{cases} 
0, & n = 0 \\
c_x(n), & n = 0, 1, \ldots, N/2 \\
-c_x(n), & n = N/2 + 1, \ldots, N - 1.
\end{cases}
\]

---

Given magnitude function \( |X(k)| \)
STEP1: Compute \( \ln X(k) \);
STEP2: Compute the cepstrum \( c_x \) by applying
N-point DFT to \( \ln X(k) \);
STEP3: Form the odd symmetric sequence \( c_o(n) \) using (6).
STEP4: Compute the phase \( \arg X(k) \) by applying
N-point DFT to \( c_o(n) \);
STEP5: Compute \( X(k) = |X(k)|e^{j\arg X(k)} \);
STEP5: Compute \( x(n) \) by applying
N-point IDFT to \( X(k) \).
There is a strong similarity between this method and logarithmic Hilbert transform, and this similarity applies to both advantages and drawbacks. From implementation point of view, there are certain ways to compute cepstrum and associated minimum-phase sequence, however they do not guarantee M-point compact support for resulting sequence. For instance, Figure 3 shows the sequence:

\[ x(n) = \begin{cases} 
    n, & \text{if } n = 1, 2, 3, 4, 5; \\
    0, & \text{if } n = 6, 7, 8, 9. 
\end{cases} \]

and its associated minimum-phase sequence computed with RCEPS from Matlab. It can be easily seen that the minimum-phase sequence is nonzero for all \( n = 1, 2, \ldots, 9. \)

3.4. Minimum-phase phase retrieval by solving linear systems of equations

Let \( x(n) \) be real and minimum-phase. The phase retrieval problem can be rewritten as follows [6]:

Compute \( X(z) \) from \( S(z) \), where \( S(z) = X(z)X(1/z) \), thus

\[ R^{-1}(z)X(z) = X^{-1}(1/z). \]

Let \( S_1(z) = S^{-1}(z) \) have inverse \( z \)-transform \( r_1(n) \):

\[ Z\{r_1(n)\} = S_1(z). \]

Then \( r_1(n) \) can be computed using inverse Fourier transform:

\[ r_1(n) = \frac{1}{2\pi} \int_{0}^{2\pi} S_1(\omega)e^{-i\omega}d\omega, \]

and this can be evaluated using a DFT of sufficiently high order. Since \( X(z) \) has all zeros inside the unit circle, equating the coefficients of \( z^i \) for \( i = 1, 2, \ldots, M - 1 \) gives us:

\[
\begin{bmatrix}
  r_1(0) & r_1(1) & \cdots & x(0) \\
  r_1(1) & r_1(0) & \cdots & x(1) \\
  \vdots & \vdots & \ddots & \vdots \\
  \end{bmatrix}
\begin{bmatrix}
  0 \\
  1 \\
  \vdots \\
  \end{bmatrix}
= A(0)
\]

as \( r_1(n) \) is an even sequence. Here \( A(0) \) is an unknown constant which scales the solution.

With this method alias still needs to be avoided. Consequently we need in implementation with a high order DFT. Also inverse of Toeplitz matrix will charge computational complexity.

4. CONCLUSIONS

In this paper we have overviewed the standard methods for direct 1-D phase retrieval. It seems that two reasons make iterative techniques more attractively. First, we can easily satisfy time-domain constraints like compact support of sequence. Complexity overload is very large for almost convenient direct methods.

There are three fundamental issues involved in the phase-retrieval problem: the uniqueness of the solution, the development of algorithms for reconstructing a signal from the magnitude of its Fourier transform, and the sensitivity of the reconstruction to measurement errors and computational noise. In this paper we have focused in the first two aspects. However, in general, algorithms for phase retrieval tend to be sensitive to noise, and the algorithms presented in this paper are no exception.

We have concentrated only on minimum-phase phase retrieval. For non-minimum phase retrieval the algorithms are similar to other phase-retrieval algorithms in the sense that some signal information, other than the magnitude of its Fourier transform is assumed to be known.

A. ZERO-PHASE SEQUENCES

The Fourier transform \( X(\omega) \) of a sequence \( x(n) \) is real whenever:

\[ X(\omega) = X^*(\omega), \]

and this is true if and only if \( x(n) \) is conjugate-symmetric:

\[ x(n) = x^*(-n). \]

Taking \( z \)-transform on both sides, we get that

\[ X(z) = X^*(1/z^*). \]

This implies that

- If there is a zero at \( z = z_0 \), there is also a zero at \( z = 1/z_0 \).
- If there is a pole at \( z = p_0 \), there is also a pole at \( z = 1/p_0 \).

These symmetry relationships apply to poles and zeros at zero and infinity.

**Definition 1.** The sequence \( x(n) \) is zero-phase sequence if its Fourier transform \( X(\omega) \) is positive semi-definite.

In this case \( X(\omega) \) is real and \( X(\omega) \geq 0. \)
B. REFERENCES


