2-2014

Numerical Inversion and Assessment of 2D Laplace Transforms using the Brancik Algorithm and its use in 3D Holography

Monish Ranjan Chatterjee
University of Dayton, mchatterjee1@udayton.edu

Le Feng
University of Dayton

Numerical Inversion and Assessment of 2-D Laplace Transforms Using the Brancik Algorithm and Its Use in 3D Holography

Monish R. Chatterjee* and Le Feng

Department of Electrical & Computer Engineering
University of Dayton, Dayton, OH 45469
*Corresponding author
Email: mchatterjee1@udayton.edu; Tel.: (937) 886 2427

Abstract

An analytic examination of 3D holography under a 90° recording geometry was carried out earlier in which 2D spatial Laplace transforms were introduced in order to develop transfer functions for the scattered outputs under readout [1,2]. Thereby, the resulting reconstructed output was obtained in the 2D Laplace domain whence the spatial information would be found only by performing a 2D Laplace inversion. Laplace inversion in 2D was attempted by testing a prototype function for which the analytic result was known using two known inversion algorithms, viz., the Brancik and the Abate [2]. The results indicated notable differences in the 3D plots between the algorithms and the analytic result, and hence were somewhat inconclusive. In this paper, we take a closer look at the Brancik algorithm in order to understand better the implications of the choices of key parameters such as the real and imaginary parts of the Bromwich contour and the grid sizes of the summation operations. To assess the inversion findings, three prototype test cases are considered for which the analytic solutions are known. For specific choices of the algorithm parameters, optimal values are determined that minimize errors in general. It is found that even though errors accumulate near the edges of the grid, overall reasonably accurate inversions are possible to obtain with optimal parameter choices that are verifiable via cross-sectional views. Further work is ongoing whereby the optimized algorithm is to be applied to the 3D holography problem.

Keywords: Holography, 2D Laplace inversion, Brancik algorithm, Hadamard product, Lozenge diagram, ε -algorithm, object wave, reference wave, 90-degree geometry

1. Introduction

Holography is a technology enabling 3D imaging to be achieved. In general, holography involves the use of coherent, monochromatic light via interference patterns between an object and a reference beam. Subsequent reconstruction involves illuminating the recorded hologram with a matched reference or read beam, resulting in a reconstructed image wavefront. Standard planar holography involves interfering wavefronts applied upon the same plane of a recording medium. On the other hand, if the interfering wavefronts are applied from different planes, the resulting holographic grating is created within the bulk (volume) of the material along the bisector in the plane of the interfering beams. Analytic investigation of the 3-D or volume holography problem is inherently more complex, and may involve integrals in the complex domain.

A possible approach to solving the problem was developed earlier in which the two reconstructed orders generated from a volume hologram with 90-degree recording geometry were modeled linearly via transfer functions derived in the 2-D Laplace domain for given spatial Laplace transforms of the

Practical Holography XXVIII: Materials and Applications, edited by Hans I. Bjelkhagen,
V. Michael Bove, Jr., Proc. of SPIE Vol. 9006, 90060V · © 2014 SPIE
CCC code: 0277-786X/14/$18 · doi: 10.1117/12.2053628

recording beams. This enabled the derivation of reconstructed wavefronts in the 2-D Laplace domain. However, obtaining the inverse Laplace results from the derived Laplace spectra proved considerably difficult, in particular from the analytic perspective.

Unlike 1-D Laplace inversion involving techniques such as partial fractions, inversion of 2-D Laplace functions is much more complicated. Possible alternatives include numerical inversions involving algorithms with limited ranges of accuracy, and sometimes serious convergence problems. Brancik and Abate are two such popular algorithms used for solving 2D Laplace inversion. Chatterjee *et. Al* showed earlier that the Brancik algorithm appears to follow the analytical result more closely than the Abate algorithm in performing 2-D Laplace inversions [3].

In this paper, we present a series of numerical inversions based on the Brancik algorithm, determining thereby the optimal parameters that yield accurate inversions over a reasonable range of physical dimensions. Specifically, three test functions with known analytic solutions are applied to the numerical simulations in order to ascertain acceptable optimal parameter values for accuracy. Thereafter, we apply the inversion strategy to the inversion of actual 2-D Laplace functions for the 3-D holography problem under volume holographic recording and readout for 90-degree geometry.

Section 2 presents a brief overview of the Brancik algorithm and some early results. In section 3, we present several inversion plots obtained by applying the Brancik algorithm to 2- Laplace functions for which the analytic inversion is known, and use the latter for comparison purposes. We also discuss the results in terms of optimal inversion parameters needed for minimal error within the inversion grid. Some volume holography test results for straightforward cases are presented and results discussed in section 4. Section 5 offers concluding remarks and plans for ongoing and future work.

2. Overview of the Brancik algorithm

In principle, the method is based on the summation of a two-dimensional complex Fourier series using the FFT algorithm. A complex Fourier series can be expanded successfully using simple substitutions leading to a power series. Following Brancik's work [3], the original 2-D Laplace transform of a real function of two variables $t_1 \geq 0, t_2 \geq 0$ and its inversion are defined as

$$F(s_1, s_2) = \int_0^\infty \int_0^\infty e^{-s_1 t_1 - s_2 t_2} f(t_1, t_2) dt_1 dt_2 \quad , \tag{1}$$

$$f(t_1, t_2) = -\frac{1}{4\pi} \int_{c_1 - j\infty}^{c_1 + j\infty} \int_{c_2 - j\infty}^{c_2 + j\infty} e^{s_1 t_1 + s_2 t_2} F(s_1, s_2) ds_1 ds_2 \quad . \tag{2}$$

Since Matlab can only perform finite summations, we have to use a sampling method to transform the double integral into a double summation. Substitutions of the Laplace frequency variables $s_i = c_i + j\omega_i$, $i = 1,2$, into (2) and applying a step frequency $\Omega_i = 2\pi / N_i T_i$, $i = 1,2$ where T_i, $i = 1,2$ are sampling periods in the original domain, we obtain the discrete points $t_{k_i} = k_i T_i$, $k_i = 0,1,2...N_1 - 1$. After eliminating the extra term, the discrete version of the inversion in eq. (3) becomes:

$$\bar{f}^{k_1,k_2} = C^{k_1,k_2}\left\{2\,\mathrm{Re}[\sum_{n_1=0}^{\infty}\sum_{n_2=0}^{\infty}F_{-n_1,-n_2}E_{-n_1,-n_2}^{k_1,k_2} + \sum_{n_1=0}^{\infty}(\sum_{n_2=0}^{\infty}F_{-n_1,n_2}E_{n_2}^{k_2})E_{-n_1}^{k_1} - \sum_{n_1=0}^{\infty}F_{-n_1,0}E_{-n_1}^{k_1} - \sum_{n_2=0}^{\infty}F_{0,-n_2}E_{-n_2}^{k_2}] + F_{0,0}\right\} \tag{3}$$

where $F_{n_1,n_2} = F(c_1 + jn_1\Omega_1, c_2 + jn_2\Omega_2)$, (4)

$$E_{n_1,n_2}^{k_1,k_2} = e^{jk_1T_1n_1\Omega_1 + jk_2T_1n_2\Omega_2} = E_{n_1}^{k_1}E_{n_2}^{k_2} \quad \text{, and} \tag{5}$$

$$C^{k_1,k_2} = \frac{\Omega_1\Omega_2}{4\pi^2}e^{c_1k_1T_1 + c_2k_2T_2} = C^{k_1}C^{k_2} \quad . \tag{6}$$

In order to increase the accuracy of $\bar{f}^{k_1,k_2}$, one needs to make N_1, N_2 as large as possible. Making these integers (which define the grid-size in the 2-D inversion plane) too large, on the other hand, would result in higher CPU cost as well as impact convergence, especially around the grid edges. In order to find a useful compromise for both integers, Brancik has suggested applying the so-called ε-algorithm. In what follows, we briefly discuss this algorithm along the lines in ref. [3].

One may describe the ε-algorithm as a numerical evaluation of sums and products that samples a number of additional terms in the double summation series and then extrapolates them to a polynomial multiplied by a decaying exponential. The principle of the ε-algorithm can be expressed graphically using a *lozenge* diagram as shown in Fig.1.

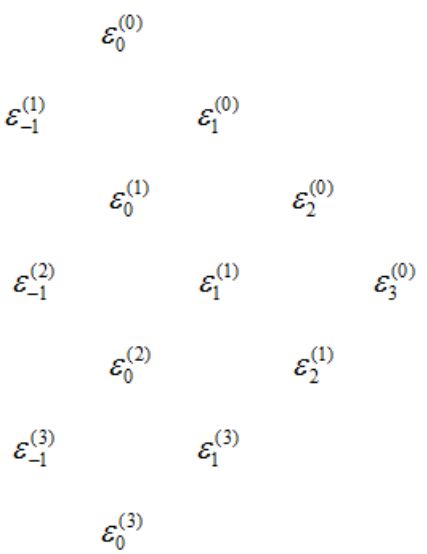

Fig.1. The ε-algorithm lozenge diagram (per Brancik [3]).

All ε terms in the lozenge diagram are matrices with the same order. We manually initialize the first column $\varepsilon_{-1}^{(s)} = 0$, where s is an integer, and 0 means a zero or null matrix. The second column is defined as

$$\varepsilon_0^{(s)} = \varepsilon_0^{(s-1)} + F_{-n_1,n_2}^{(N_1+s,:)} \otimes [E_I^{(N_1+s,:)}]^H, \quad s = 1,2,....P \quad . \tag{7}$$

In eq. (7), $\otimes$ is the Kronecker tensor product of matrices. In practice, the extension parameter P is used to justify the result. Thus, the hologram created is assumed to have the order $(N_1 + 2P) \times (N_2 + 2P)$. According to Brancik, the error increases rapidly near far ends of the original 2-D region, thus requiring selection of data from the front (and inner) region of holography. The matrix F_{-n_1,n_2} enters into an $M_1 = \frac{N_1}{2}$- point FFT algorithm (along the axis X in the simulation), and the result is truncated to the order $M_1 \times (N_2 + 2P)$. In this one obtains the initial matrix $\varepsilon_0^{(0)}$. The matrix next enters into an $M_2 = \frac{N_2}{2}$- point FFT algorithm (along the axis Y in the simulation), and the result is truncated to the order $M_1 \times M_2$. The remaining columns are calculated by a recursion law as follows:

$$\varepsilon_{r+1}^{(s)} = \varepsilon_{r-1}^{(s+1)} + [\varepsilon_r^{(s+1)} - \varepsilon_r^{(s)}]^{-1}, \; r,s = 1,2,...2P \; . \tag{8}$$

The inversion in eq.(3) may now be rewritten in terms of matrices as:

$$f^{k_1,k_2} = C^{k_1,k_2} \circ \{2\,\mathrm{Re}[G_{12} + G_{21} - G_{11} - G_{22}] + G_{00}\} \; . \tag{9}$$

All the terms in eq.(9) are $M_1 \times M_2$ matrices. The following describe each term in eq.(9):

$$C^{k_1,k_2} = C^{k_2} \otimes C^{k_1} \; , \tag{10}$$

$$G_{00} = F_{0,0}[I_{M_2} \otimes I_{M_1}] \; , \tag{11}$$

$$\xi\{\underset{<X>}{FFT}(F_{-n_1,n_2})\} \to G_1, \qquad \xi\{\underset{<Y>}{FFT}(G_1^*)\} \to G_{12} \; , \tag{12}$$

$$\xi\{\underset{<Y>}{FFT}(F_{-n_1,-n_2})\} \to G_2, \quad \xi\{\underset{<X>}{FFT}(G_2^*)\} \to G_{21}, \tag{13}$$

$$G_{11} = I_{M_2} \otimes G_1^{(:,1)} \; , \text{ and} \tag{14}$$

$$G_{22} = I_{M_1} \otimes G_2^{(1,:)} \quad . \tag{15}$$

The subscript $< X >$ and $< Y >$ are the FFT along axis X and axis Y respectively. The symbol $\circ$ is the Hadamard product of matrices. $\zeta\{\}$ represents the operator of the ε -algorithm which is applied

to the result of each FFT operation. I_{M_1} and I_{M_2} are $M_1 \times 1$ and $1 \times M_2$ matrices all of whose elements are equal to 1. $G_1^{(:,1)}$ and $G_2^{(1,:)}$ are the Matlab language of 1st column and 1st row of G_1 and G_2.

Eq.(9) is useful in computing the inverse of a 2-D Laplace function. The result obtained is impacted by 3 important parameters: the extension coefficient P, the sampling frequency N_i, $i = 1,2$, and the coefficients c_i, $i = 1,2$. In the next section, we focus on how the 3 parameter play a role in determining the correct inverse and finding the optimum parameters. For simplicity, we assume $N_1 = N_2$ and $c_1 = c_2$.

3. Simulation results

The motivation for the current work is to simulate numerical 2-D Laplace inversion using eq. (9). In order to make a good comparison, we select 3 test functions whose inverse Laplace expressions are already known.

Case 1

We first consider the 2-D Laplace function

$$F(s_1, s_2) = \frac{(s_1 s_2 - 1)}{(s_1^2 + 1)(s_2^2 + 1)} \leftrightarrow f(x, y) = \cos(x + y), \tag{16}$$

where $f(x,y)$ is the analytic inverse transform.

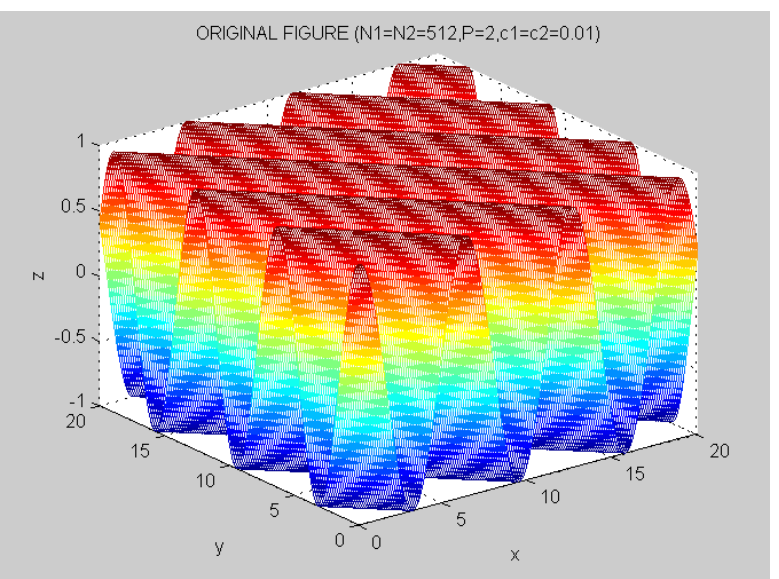

Fig.2. The original figure of $f(x, y) = \cos(x + y)$

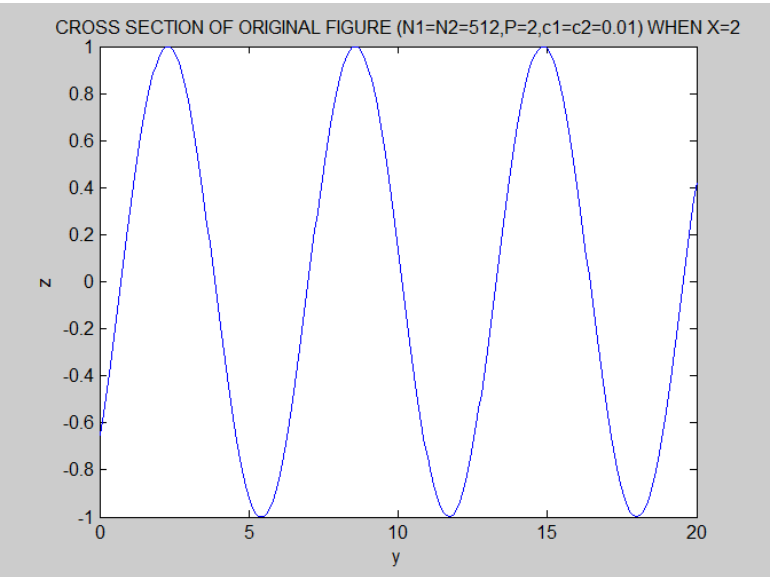

Fig.3. The $x = 2$ cross section of $f(x, y) = \cos(x + y)$

Figs.2 and 3 show the analytic 3-D plot of the inverse transform and the cross-section of the plot along $x = 2$. The 2-D plot shows the phase-shifted cosine function, as expected.

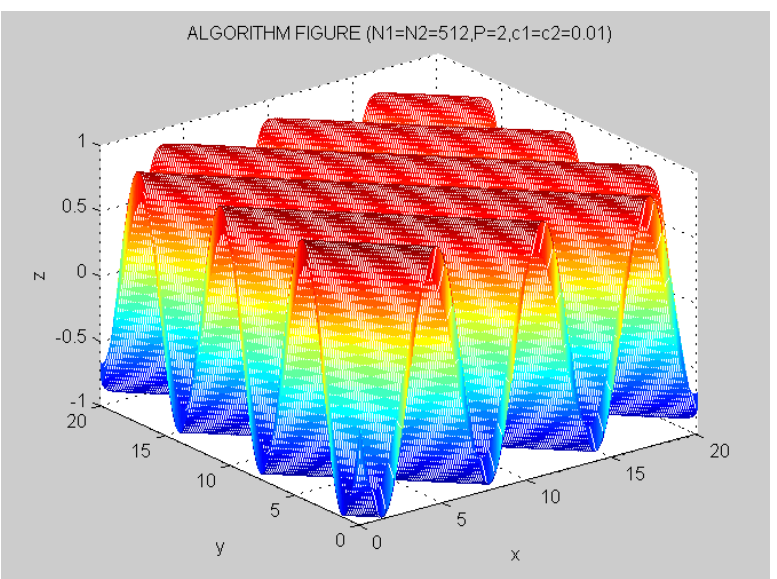

Fig 4. The Brancik algorithm figure coming from $F(s_1, s_2) = \dfrac{(s_1 s_2 - 1)}{(s_1^2 + 1)(s_2^2 + 1)}$

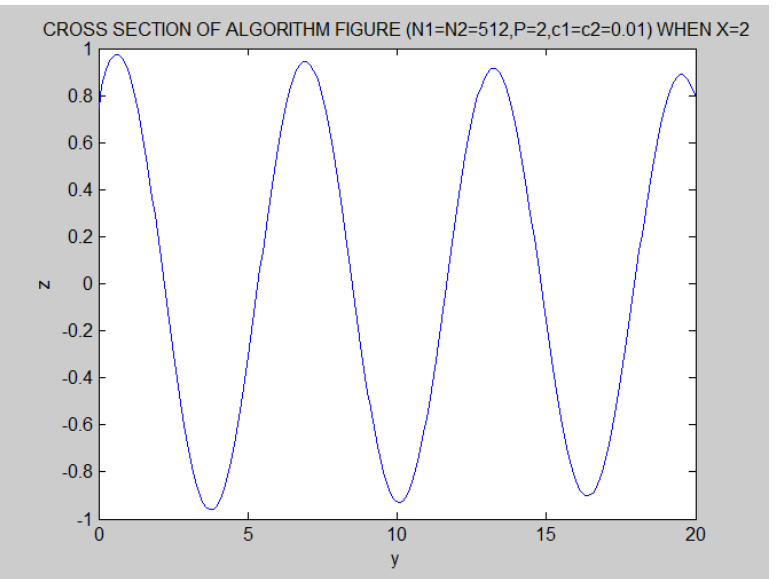

Fig 5. The $x = 2$ cross section of Fig 4.

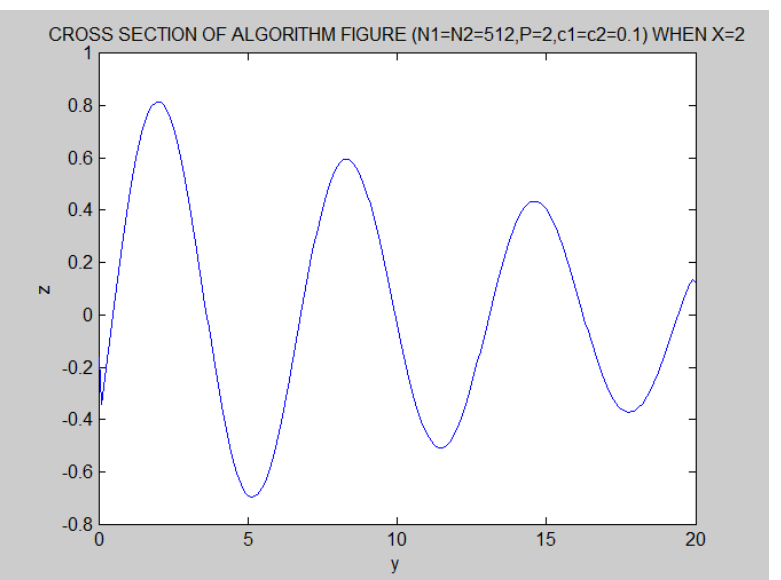

Fig 6. The $x = 2$ cross section algorithm figure, but $c_1 = c_2 = 0.1$.

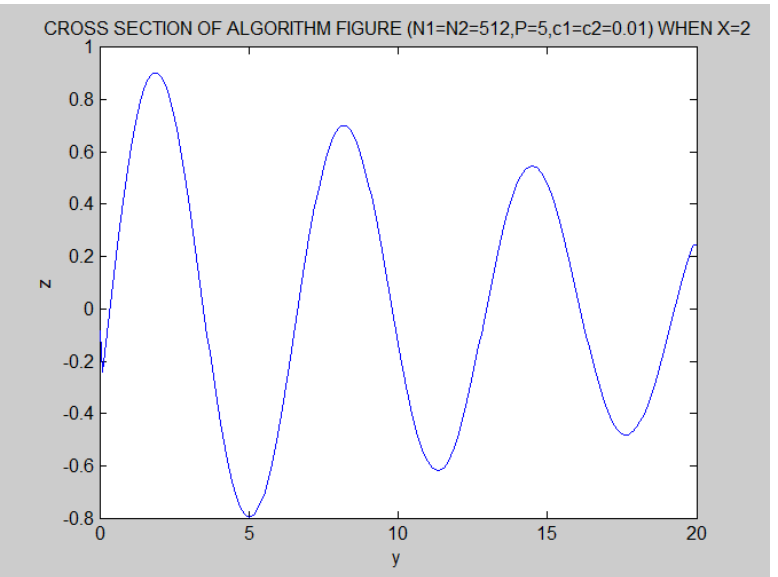

Fig 7. The $x = 2$ cross section algorithm figure, but $P = 5$.

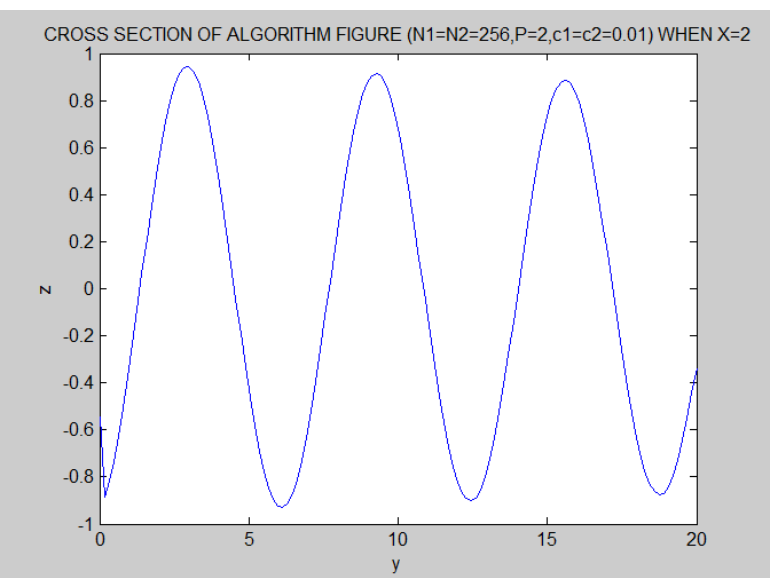

Fig 8. The $x = 2$ cross section algorithm figure, but $N_1 = N_2 = 256$

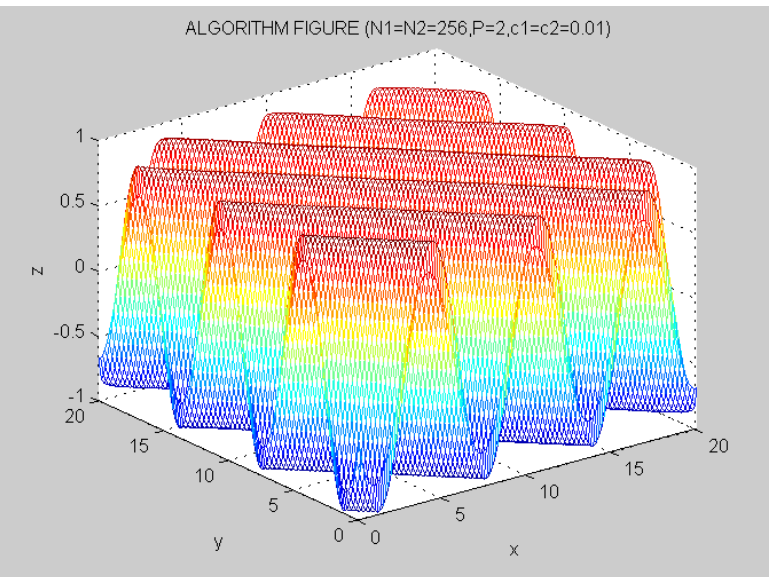

Fig 9. The 3D figure corresponding to Fig.8.

Figs.4-9 depict a series of plots based on the numerical inversions of the function in eq. (16). Figs.4 and 9 correspond to the 3-D representation with the chosen inversion parameters, whereas Figs.5-8 show 2-D cross sections along $x = 2$.

Fig.3 is a perfect sine or cosine function with a phase shift. Figs.5-7 show the envelope of the phase-shifted sine function with a relatively steep decay as we change c_1, c_2 or P. Figs.3 and 5 are overall comparable, except for a slight attenuation for larger values of x in Fig.5. Figs.6 and 7 show considerably greater attenuation in the inversion waveform for $c_1 = c_2 = 0.1$ and $P = 5$ respectively. In Fig.8 with the choice of $c_1 = c_2 = 0.01$, $P = 2$, and $N_1 = N_2 = 256$, we observe an inversion plot virtually indistinguishable from that of the analytic result in Fig.3. The corresponding 3-D algorithmic plot with these "optimal" parameters, as in Fig.9, appears to be visually lighter; this effect occurs because for a grid number $N_1 = N_2 = 256$, the number of generated points within the Matlab program is fewer than with $N_1 = N_2 = 512$. Thus, there is a lower density of points with smaller grid number, resulting in the lighter plot visibility. Thus, there is a tradeoff between higher or lower values of N_1, N_2 in the sense that while higher values of this parameter yields higher density of points, it also leads to increased computational time or CPU cost.

When we compare Fig.5 to Fig.3, we find that the envelope of the sinusoidal plot decays slowly and each extreme point has a left shift which happens in the next two cases as well. In the 'near' region, roughly defined in the first half of the plot, the error is less than 5%.

To ascertain the most appropriate opitmal parameter in the Brancik algorithm, we eventually adopted the standard parameters $c_1 = c_2 = 0.01, P = 2, N_1 = N_2 = 512$, which yield satisfactory results for a number of inversion simulations. While several test cases were analyzed, we have chosen to discuss only three in this paper.

Case 2

We next examine the 2-D Laplace pair:

$$F(s_1, s_2) = \frac{\sqrt{s_2}}{(s_1 s_2 + 1)(s_1 s_2 + s_2 + 1)} \leftrightarrow f(x, y) = erf(\sqrt{x}) J_0(2\sqrt{xy}) \ . \tag{17}$$

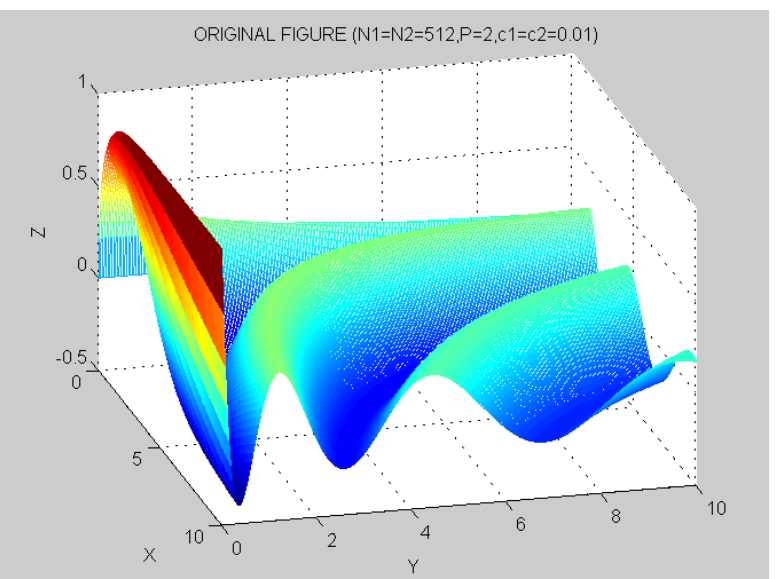

Fig.10. The original figure of $f(x, y) = erf(\sqrt{x}) J_0(2\sqrt{xy})$.

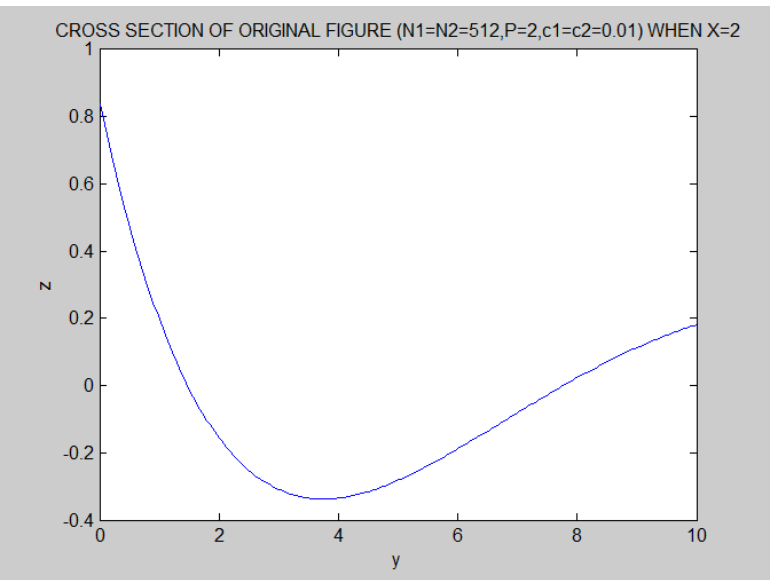

Fig.11. The $x = 2$ cross section of Fig.10.

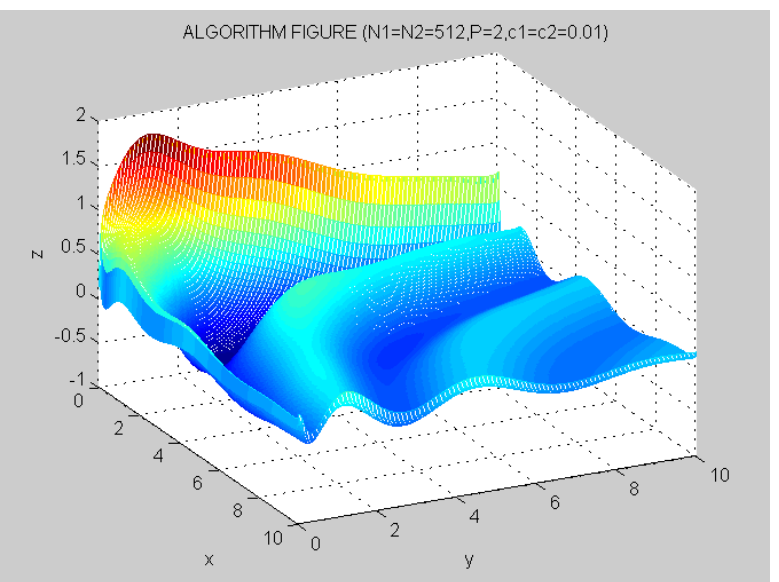

Fig.12. The Brancik algorithm figure coming from $F(s_1, s_2) = \dfrac{\sqrt{s_2}}{(s_1 s_2 + 1)(s_1 s_2 + s_2 + 1)}$.

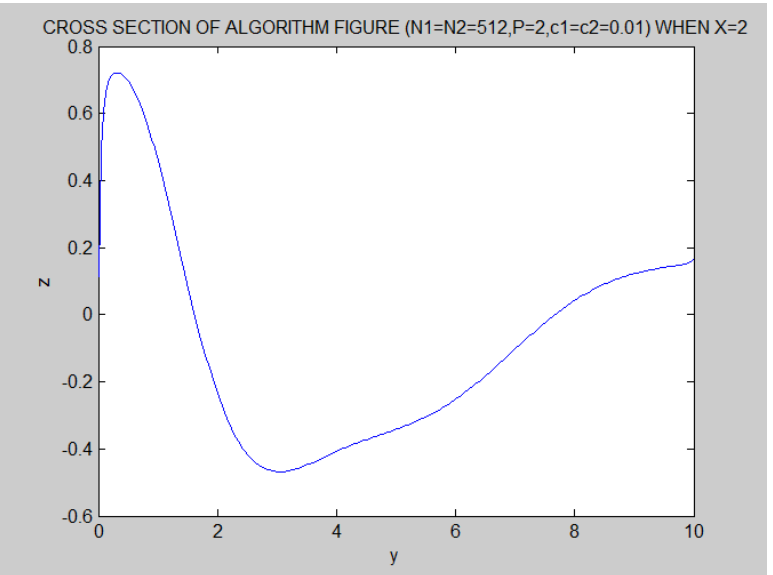

Fig 13. The $x = 2$ cross section of Fig.12.

For this problem, we again show the analytic 3-D and 2-D cross section plots in Figs.10 and 11. For optimal parameters as mentioned earlier, we then show the inversion plots obtained from the algorithm. Note the overall similarity between both the 3-D plot and the 2-D section along $x = 2$ obtained via the algorithm and those from the analytic solution.

The maximum value of the function in Fig.11 is at $x = 0$. Fig.13 shows the peak to be slightly shifted to the right, implying that a small left shift of the plot in Fig.13 would nearly recover the original plot in Fig.11.

CASE 3

The test function for this case is given below:

$$F(s_1, s_2) = \frac{e^{-s_1}}{s_1 s_2 + 1} \leftrightarrow f(x, y) = J_0[2\sqrt{(x-1)y}]. \tag{18}$$

Incidentally, this function is the same as the one originally examined by Chatterjee *et. al* [2] in order to compare the Brancik against the Abate algorithm .

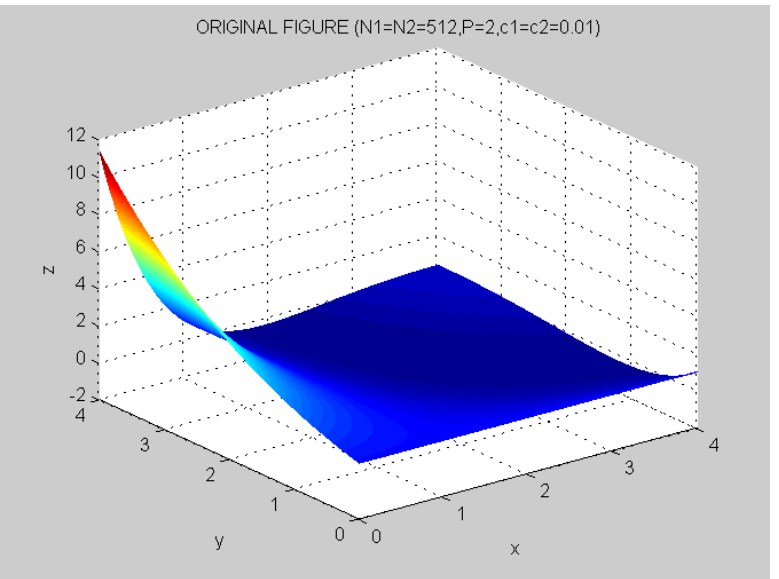

Fig.14. The original analytic figure of $f(x,y) = J_0[2\sqrt{(x-1)y}]$.

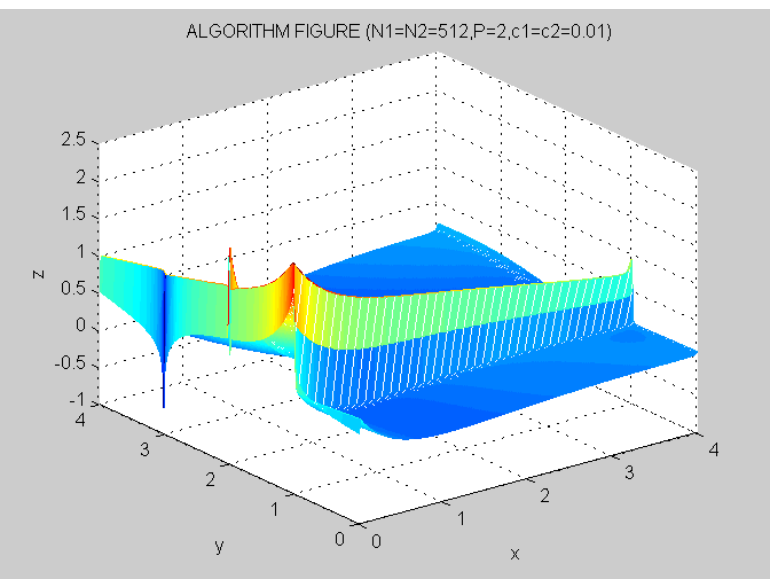

Fig.15. The Brancik algorithm for $F(s_1,s_2) = \dfrac{e^{-s_1}}{s_1 s_2 + 1}$.

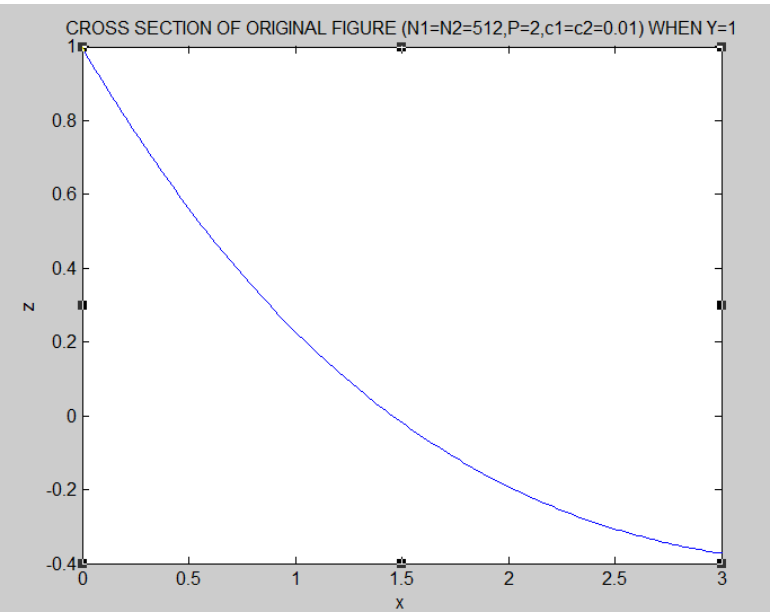

Fig.16. The $y=1$ cross section of Fig.14.

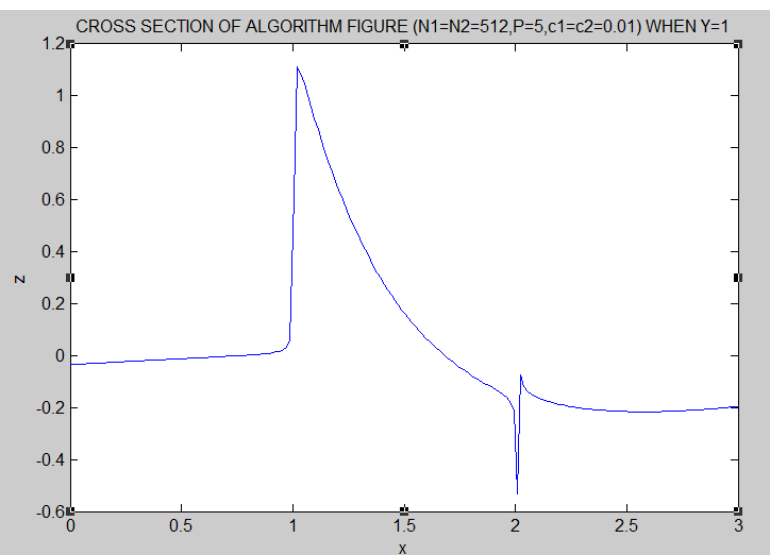

Fig.17. The $y=1$ cross section of Fig.15.

The plots in Figs.14-17 correspond to the two analytic and two algorithmic plots for the chosen test function. Since $x \geq 1$ is the condition of the original expression, we have to start the comparison beginning at $x = 1$. The maximum point in Fig.17 is past the range $x \in (0,1)$, and slightly to the right. Hence, the 'left shift' needed to restore the algorithm to the analytic result still exists. The tendency of each plot to decay and nearly end at $z = f = -0.2$ in the 'near' region is evident between Figs.16 and 17; thus, both plots decay to around – 0.2 in the neighborhood of x = 2. Beyond x = 2, the algorithmic plot begins to somewhat diverge from the analytic result. Therefore, for this problem, the "near" region is within the range $1 \leq x \leq 2$, and for all practical purposes, the inversion is more accurate within this range. We also observe some unexpected oscillation in the neighborhood of about $x = 2$ in Fig.17. Overall, however, within a relatively small (i.e., near) region of the grid coordinates, the algorithm appears to yield reasonably accurate results.

4. Volume holography with 90-degree geometry and Laplace inversion

Banerjee et. al have shown that the two readout beams reconstructed from a 90° volume hologram may be defined via two reduced coupled differential equations [2]:

$$\partial \psi_{e0} / \partial x = -j\kappa\psi_{e1}, \partial \psi_{e1} / \partial z = -j\kappa\psi_{e0} \tag{19}$$

where $\kappa = k\varepsilon_2 / \varepsilon_1$ is a coupling parameter, $k = 2\pi\sqrt{\varepsilon_1} / \lambda_2$ is the propagation constant of the reconstruction or read wavefront, ψ_{e0} and ψ_{e1} are the zeroth- and first-order readout beams in the phasor domain. We next present a simple application of the above coupled system to illustrate use of the 2-D Laplace and its inversion in understanding readout from a volume hologram.

Uniform Plane Wave Readout

With the boundary condition for the 0th and 1st orders taken as: $\psi_{e0}(0,z) = Cu(z)$, $\psi_{e1}(x,0) = 0$,

where $u(z)$ is a unit step in the z direction, and C is a constant. We Laplace transform eq.(19) in 2-D between (x,z) and (s_x,s_z) to get:

$$s_x\psi_{e0}(s_x,s_z) - \tilde{\psi}_{e0}(x=0,s_z) = -j\kappa\psi_{e1}(s_x,s_z), \text{ and} \tag{20a}$$

$$s_z\psi_{e1}(s_x,s_z) - \tilde{\psi}_{e1}(s_x,z=0) = -j\kappa\psi_{e0}(s_x,s_z), \tag{20b}$$

where $\tilde{\psi}$ is the Laplace transform of ψ. By solving the above coupled equations, we obtain

$$\tilde{\psi}_{e0}(s_x,s_z) = \frac{C}{s_xs_z + \kappa^2}, \tilde{\psi}_{e1}(s_x,s_z) = \frac{-j\kappa C}{s_z(s_xs_z + \kappa^2)} \quad . \tag{21}$$

To obtain the Laplace inversion of the above in 2-D, we next apply the Brancik algorithm, assuming $\varepsilon_1 = 2.5\varepsilon_0$, $\varepsilon_2 = 0.4\varepsilon_0$, $\lambda_2 = 10^{-6} m$, and $C = 1$. Analytically, we inverse Laplace transform eq.(21) first with the respect s_x, and then with the respect s_z (using a power series expansion), finally obtain the result:

$$\psi_{e1}(x,z) = jC\sqrt{\frac{z}{x}}J_1(2\kappa\sqrt{xz}). \tag{19}$$

Generally, the first-order beam is of greater interest, because it represents the diffraction from the hologram, and depicts the reconstructed object wave.

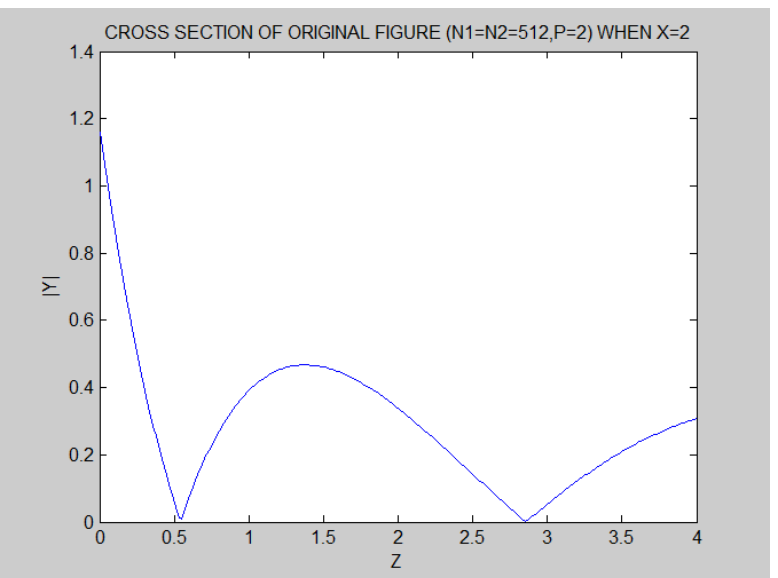

Fig.18. The $x = 2$ cross section of analytic inversion of $\psi_{e1}(x,z) = jC\sqrt{\frac{z}{x}}J_1(2\kappa\sqrt{xz})$.

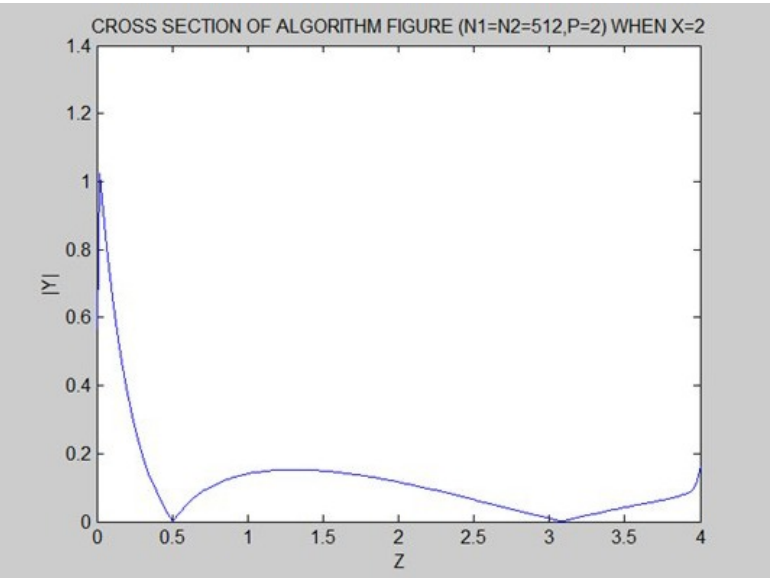

Fig.19. The $x = 2$ cross section arising from numerical inversion of $\tilde{\psi}_{e1}(s_x, s_z) = \dfrac{-j\kappa C}{s_z(s_x s_z + \kappa^2)}$.

The basic shape and tendencies for both figures are quite similar. We find that both right and left shifts exist at the extreme points in this case. At least so far, we have found that the optimum parameters chosen provide accurate inversion for the uniform plane wave case..

5. Conclusion and further work

In this paper, we were able to determine approximately the optimum inversion parameters needed to apply the Brancik algorithm for 2-D Lapace inversion with some accuracy. Successive runs of the algorithm indicated that the coefficients c_1, c_2 impact the values of the extreme (or grid boundary) points, which leads to envelope decay. Optimal choices were found by selecting $c_1 = c_2 = 0.01, N_1 = N_2 = 512, P = 2$ via a series of comparisons with known analytic test cases. The results were generally satisfactory, even though there continued to be errors and limitations, especially w.r.t. the grid boundaries, and also the range or size of the inversion map. Laplace inversion is the most direct way to solve the reconstruction problem in holography, but not the only way. Fortunately, the result displayed on Fig.19 provides some confidence that the inversion strategy will likely work for other specific cases involving the object and reference wavefronts in the 90-degree holographic write mode, such as, say, point source illumination or even more general cases. However, as has been seen, there exist left and right shifts in the inverted functions for practically all the cases. Further investigations are needed relative to the properties of the readout beam and its spatial behavior. In order to examine this with greater precision, we intend to use error algorithms to find relative and absolute errors in the inversion on a point by point basis.

References

1. J.W. Goodman, *Introduction to Fourier Optics*, 3rd edition, McGraw-Hill: New York (2005).

2. M.R. Chatterjee, P.P. Banerjee and G. Nehmetallah, "Analysis of Beam Propagation in 90-Degree Holographic Recording and Readout Using Transfer Functions and Numerical 2-D-Laplace Inversion," DH Topical Meeting, Vancouver, Canada, PMA6, June 18-20, 2007.

3. L. Brancik "An Improvement of FFT-Based Numerical Inversion of 2D Laplace Transforms by Means of ε -Algorithm" ISCAS-IEEE international Symposium on Circuits and Systems, pp.IV-581-584 (2000).

4. P.P.Banerjee, M.R.Chatterjee, N.Kukhtarev and T.Kukhtareva, "Volume Holographic Recording and Readout for 90-deg Geometry," Opt. Eng. 43, 9, 2053-2060 (Sept. 2004).