An FFT-Based Algorithm for 2D Power Series Expansions

CHYI HWANG*
Department of Chemical Engineering, National Chung Cheng University
Chia-Yi 621, Taiwan, R.O.C.

JIA-CHYU GUO
Department of Chemical Engineering, National Cheng Kung University
Tainan 701, Taiwan, R.O.C.

TONG-YI GUO
Department of Chemical Engineering, National Kaohsiung Institute of Technology
Kaohsiung 807, Taiwan, R.O.C.

(Received July 1997; revised and accepted January 1999)

Abstract—An effective numerical algorithm based on inverting a specialized Laplace transform is derived for computing the two-dimensional power-series expansion coefficients of a two-variable function. Due to the special structure of the constructed 2D Laplace transform, the accuracy of the inverted function values can be assured effectively by the generalized Riemann zeta function evaluation and the multiple sets of 2D FFT computation. Therefore, the algorithm is particularly amenable to modern computers having multiprocessors and/or vector processors. © 1999 Elsevier Science Ltd. All rights reserved.

Keywords—Two-dimensional systems, Fast Fourier transform, Numerical Laplace transform inversion, Riemann zeta function.

1. INTRODUCTION

Consider a two-variable function \( f(z_1, z_2) \) which permits the two-dimensional (2D) power series expansion

\[
f(z_1, z_2) = \sum_{j=0}^{\infty} \sum_{k=0}^{\infty} f_{j,k} z_1^j z_2^k.
\]  

In the above expansion, the coefficients \( f_{j,k} \) are related to the function \( f(z_1, z_2) \) by [1]

\[
f_{j,k} = \frac{1}{j! k!} \frac{\partial^{j+k} f(0,0)}{\partial z_1^j \partial z_2^k}
\]

or by [2]

\[
f_{j,k} = \frac{1}{(2\pi i)^2} \oint_{|z_1|=c(r_1)} \oint_{|z_2|=c(r_2)} \frac{f(z_1, z_2)}{z_1^{j+1} z_2^{k+1}} dz_1 dz_2, \quad i = \sqrt{-1},
\]

*Author to whom all correspondence should be addressed.
This work was sponsored by the National Science Council of Republic of China under Grant NSC-79-0402-E006-03.

0898-1221/99/$ - see front matter. © 1999 Elsevier Science Ltd. All rights reserved. Typeset by AMS-TEX

PII: S0898-1221(99)00122-4
where \( c(r_1) \) and \( c(r_2) \) are circles with radii \( r_1 \) and \( r_2 \), respectively, within which \( f(z_1, z_2) \) is analytic. The problem of obtaining the expansion coefficients \( f_{j,k} \) for the 2D power series (1) arises in many disciplines. Since it has many applications to 2D system theory and signal processing, such as Fourier series expansion, Taylor series expansion, generating functions, and \( z \)-transform and Laplace transform inversions, the 2D power series expansion plays an important role in engineering [3].

The coefficients \( f_{j,k} \) of the 2D power series (1) can be evaluated by taking partial derivatives of \( f(z_1, z_2) \), as shown in (2). However, if the form of \( f(z_1, z_2) \) is complicated and/or the orders \( j \) and \( k \) are high, the direct differentiation procedure inevitably involves tedious algebraic manipulations. It is thus desirable to have computationally efficient numerical methods for evaluating the high-order derivatives of the two-variable analytic function \( f(z_1, z_2) \). Surprisingly, there is hardly any literature on the numerical evaluation of 2D power series though its 1D counterpart has received much attention [4-9]. This may partly ascribe to the fact that most numerical methods of evaluating 1D power-series-expansion coefficients can be readily extended to the 2D case. For example, the classical difference formulas [4,6] or the numerical approach of computing the complex integral by the FFT algorithm [5,9] can be directly applied without any modifications to obtain the 2D power series of \( f(z_1, z_2) \).

In this paper, the efficient numerical approach for the 1D power series expansion which was originally proposed by Abate and Dubner [10] and was recently improved by Hwang et al. [11], is extended to the 2D case. Since the extension is not so straightforward, it is worth presenting the approach and the computational algorithm in detail. The organization of the paper is as follows. In Section 2, a 2D Laplace transform function is defined such that the values of the inverted function at integer grids are explicitly related to the coefficients \( f_{j,k} \). The computational algorithm for inverting the specialized 2D Laplace transform is derived in Section 3 through the use of double trapezoidal approximations to the 2D Bromwich integral associated with the constructed 2D Laplace transform function. It involves the evaluation of the generalized Riemann zeta function and multiple sets of 2D FFT computations. In Section 4, several examples are provided to show the effectiveness of the algorithm in evaluating 2D power series expansions. Finally, in Section 5, the conclusion is given.

### 2. THE APPROACH

Consider a double-indexed sequence \( g_{m,n} \) defined for \( m \geq 0 \) and \( n \geq 0 \) by

\[
g_{m,n} = \sum_{j=0}^{m-1} \sum_{k=0}^{n-1} f_{j,k}
\]

with \( g_{m,n} = 0 \) for \( m = 0 \) or \( n = 0 \), where \( f_{j,k} \) are the coefficients of the 2D power series (1). It is obvious form (4) that

\[
g_{m+1,n} = g_{m,n} + \sum_{k=0}^{n-1} f_{m,k},
\]

\[
g_{m,n+1} = g_{m,n} + \sum_{j=0}^{m-1} f_{j,n},
\]

and

\[
g_{m+1,n+1} = g_{m,n} + \sum_{j=0}^{m-1} f_{j,n} + \sum_{k=0}^{n-1} f_{m,k} + f_{m,n}.
\]

A combination of (4)-(7) yields

\[
f_{m,n} = g_{m+1,n+1} - g_{m+1,n} - g_{m,n+1} + g_{m,n}.
\]
Now, let us define a continuous two-variable function \( g(t_1, t_2) \) such that \( g(m, n) = g_{m,n} \) for positive integers \( m \) and \( n \), and \( g(t_1, t_2) = 0 \) for \( t_1 < 0 \) or \( t_2 < 0 \). It can then be shown that the function \( g(t_1, t_2) \) defined below satisfies these requirements:

\[
g(t_1, t_2) = \sum_{m=0}^{\infty} \sum_{n=0}^{\infty} \left\{ [u(t_1 - m) - u(t_1 - m - 1)][u(t_2 - n) - u(t_2 - n - 1)] 
\times \left[ g_{m,n} + (t_1 - m) \sum_{k=0}^{n-1} f_{m,k} + (t_2 - n) \sum_{j=0}^{m-1} f_{j,n} + (t_1 - m)(t_2 - n)f_{m,n} \right] \right\},
\]

where \( u(t) \) is the unit-step function defined by \( u(t) = 0 \) for \( t < 0 \) and \( u(t) = 1 \) for \( t \geq 0 \).

Let the 2D Laplace transform of \( g(t_1, t_2) \) be denoted by \( G(s_1, s_2) \), i.e.,

\[
G(s_1, s_2) = \int_{0}^{\infty} \int_{0}^{\infty} g(t_1, t_2)e^{-s_1 t_1}e^{-s_2 t_2} dt_1 dt_2.
\]

After substituting (9) for \( g(t_1, t_2) \) and performing some lengthy algebraic manipulations, we have the Laplace transform \( G(s_1, s_2) \) of \( g(t_1, t_2) \) as follows:

\[
G(s_1, s_2) = \frac{1 - e^{-s_1} - e^{-s_2}}{s_1^2} \sum_{j=0}^{\infty} \sum_{k=0}^{\infty} f_{j,k}e^{-js_1}e^{-ks_2}.
\]

Replacing the double infinite series in the last equation by

\[
f(e^{-s_1}, e^{-s_2}) = \sum_{j=0}^{\infty} \sum_{k=0}^{\infty} f_{j,k}e^{-js_1}e^{-ks_2},
\]

which comes from (1) by letting \( z_1 = e^{-s_1} \) and \( z_2 = e^{-s_2} \), we have

\[
G(s_1, s_2) = \frac{1 - e^{-s_1} - e^{-s_2}}{s_1^2} f(e^{-s_1}, e^{-s_2}).
\]

This equation simply relates the given function \( f(z_1, z_2) \) to the Laplace transform of \( g(t_1, t_2) \). To obtain the power series of \( f(t_1, t_2) \), we can first invert \( G(s_1, s_2) \) to find the values of \( g(t_1, t_2) \) at integer grids, and then determine the coefficients \( f_{m,n} \) by (8).

Before ending this section, it is worth mentioning that although the above derivations follow the 1D results [10], the construction of \( g(t_1, t_2) \) in (9) is by no means trivial or straightforward.

### 3. NUMERICAL INVERSION OF THE 2D LAPLACE TRANSFORM \( G(s_1, s_2) \)

The objective of this section is to derive an efficient algorithm for numerically inverting the 2D Laplace transform \( G(s_1, s_2) \). The algorithm is a generalization of a recently result for 1D power series [11], which takes full advantages of the method proposed by Dubner and Abate [10], and that by Hwang et al. [12].

To begin, we note that the inversion formula for \( G(s_1, s_2) \) is given by [13]

\[
g(t_1, t_2) = \frac{1}{(2\pi i)^2} \int_{c_1-i\infty}^{c_1+i\infty} \int_{c_2-i\infty}^{c_2+i\infty} G(s_1, s_2)e^{s_1 t_1 + s_2 t_2} ds_1 ds_2, \quad i = \sqrt{-1},
\]

where \( \sigma_1 > \sigma_1^0, \sigma_2 > \sigma_2^0 \), and \( \sigma_1^0 \) and \( \sigma_2^0 \) are the convergence abscissae of the function \( G(s_1, s_2) \). That is, \( G(s_1, s_2) \) is analytic for \( \text{Re} \{s_1\} > \sigma_1^0 \) and \( \text{Re} \{s_2\} > \sigma_2^0 \), where \( \text{Re} \{\cdot\} \) denotes the real
part of the argument. Since $z_i = e^{-s_i}, i = 1, 2,$ it follows from (13) that $\sigma^0_i = \max\{0, -\ln c(r_i)\}, i = 1, 2.$

By letting $s_1 = \sigma_1 + iw_1$ and $s_2 = \sigma_2 + iw_2,$ the inversion formula (14) becomes

$$g(t_1, t_2) = \frac{e^{\sigma_1 t_1 + \sigma_2 t_2}}{(2\pi)^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} G(\sigma_1 + iw_1, \sigma_2 + iw_2)e^{(w_1 t_1 + w_2 t_2)}dw_1 dw_2.$$  

(15)

Applying first, the double trapezoidal approximations with step lengths of $\Delta w_1 = 2\pi/(m_1 N_1)$ and $\Delta w_2 = 2\pi/(m_2 N_2)$ to the double improper integral (15), and then letting $t_1 = i_1$ and $t_2 = i_2,$ we have

$$g(i_1, i_2) = \frac{\exp(i_1 \sigma_1 + i_2 \sigma_2)}{m_1 m_2 N_1 N_2} \sum_{r_1 = -u_1}^{u_1} \sum_{r_2 = -u_2}^{u_2} \left\{ \sum_{k_1 = 0}^{N_1 - 1} \sum_{k_2 = 0}^{N_2 - 1} G_{r_1, r_2}(k_1, k_2) W_1^{k_1} W_2^{k_2} \right\} \cdot W_1^{i_1 r_1/m_1} W_2^{i_2 r_2/m_2},$$

where $m_1, m_2, N_1,$ and $N_2$ are positive integers,

$$u_j = \text{integer part of } \frac{m_j - 1}{2}, \quad j = 1, 2,$$

$$v_j = \text{integer part of } \frac{m_j}{2}, \quad j = 1, 2,$$

$$W_j = \exp \left( \frac{i 2\pi}{N_j} \right), \quad j = 1, 2,$$

and

$$G_{r_1, r_2}(k_1, k_2) = \sum_{p_1 = -\infty}^{\infty} \sum_{p_2 = -\infty}^{\infty} G \left( \sigma_1 + \frac{2\pi}{N_1} \left( k_1 + \frac{r_1}{m_1} + p_1 N_1 \right), \sigma_2 + \frac{2\pi}{N_2} \left( k_2 + \frac{r_2}{m_2} + p_2 N_2 \right) \right).$$

(17)

It is observed that the double sums in the square brackets of (16) can be written as

$$\left\{ \sum_{k_1 = 0}^{N_1 - 1} \sum_{k_2 = 0}^{N_2 - 1} G_{r_1, r_2}(k_1, k_2) W_1^{k_1} W_2^{k_2} \right\}^*$$

where the star "*" denotes the complex conjugate. Hence, if both $N_1$ and $N_2$ are selected as powers of 2, the computation of the 2D sequence $\{g_{r_1, r_2}(i_1, i_2)\}_{N_1 \times N_2}$ can be efficiently accomplished by the algorithm of 2D Fast Fourier Transform (FFT), i.e.,

$$\{G_{r_1, r_2}(i_1, i_2)\}_{N_1 \times N_2} \longrightarrow \text{2D-FFT} \longrightarrow \{g_{r_1, r_2}(i_1, i_2)\}_{N_1 \times N_2},$$

(19)

where a sequence $\{h_{r_1, r_2}(i_1, i_2)\}_{N_1 \times N_2}$ is defined by

$$\{h_{r_1, r_2}(i_1, i_2)\}_{N_1 \times N_2} \Delta \{g_{r_1, r_2}(i_1, i_2) : i_1 = 0, 1, \ldots, N_1 - 1; i_2 = 0, 1, \ldots, N_2 - 1 \}.$$

Once the sequences $\{g_{r_1, r_2}(k_1, k_2)\}_{N_1 \times N_2}$ for $r_j = -u_j, \ldots, -1, 0, 1, \ldots, v_j; j = 1, 2$ are obtained, the sequence $\{g(i_1, i_2)\}_{N_1 \times N_2}$ can be computed as

$$g(i_1, i_2) = \frac{\exp(\sigma_1 i_1 + \sigma_2 i_2)}{m_1 m_2 N_1 N_2} \sum_{r_1 = -u_1}^{u_1} \sum_{r_2 = -u_2}^{u_2} g_{r_1, r_2}(i_1, i_2) W_1^{i_1 r_1/m_1} W_2^{i_2 r_2/m_2},$$

$$i_1 = 0, 1, \ldots, N_1 - 1, \quad i_2 = 0, 1, \ldots, N_2 - 1.$$  

(20)
In view of (16), the computation of the sequence \{g(i_1, i_2)\} \_N_1 \times N_2 appears to need \(m_1 m_2\) sets of \(N_1 \times N_2\) 2D FFT operations. However, as shown below, the required computational burden can be significantly reduced if we select both \(m_1\) and \(m_2\) to be odd integers.

For odd \(m_1\) and \(m_2\), we have \(u_1 = v_1\) and \(u_2 = v_2\), and

\[
G_{0,0}(k_1, k_2) = G_{0,0}^*(x_1, x_2),
\]

\[
G_{r_1, r_2}(k_1, k_2) = G_{-r_1, -r_2}^*(x_1, x_2), \quad r_1 = 0, 1, \ldots, u_1, \quad r_2 = 0, 1, \ldots, u_2,
\]

where

\[
x_1 = \begin{cases} k_1, & k_1 = 0, \\ N_1 - k_1, & k_1 \neq 0, \end{cases}
\]

\[
x_2 = \begin{cases} k_2, & k_2 = 0, \\ N_2 - k_2, & k_2 \neq 0. \end{cases}
\]

Hence, the data \(g_{r_1, r_2}(i_1, i_2)\), \(i_1 = 0, 1, \ldots, N_1 - 1; \ i_2 = 0, 1, \ldots, N_2 - 1\) satisfy the following relations:

\[
g_{0,0}(i_1, i_2) = g_{0,0}^*(i_1, i_2) = \text{real},
\]

\[
g_{r_1, r_2}(i_1, i_2) = g_{-r_1, -r_2}^*(i_1, i_2).
\]

Consequently, the number of sets of \(N_1 \times N_2\) 2D FFT computations required to obtain the sequence \(\{g(i_1, i_2)\}\) \_N_1 \times N_2 now becomes \((m_1 m_2/2) + 1\). Let \(g_{r_1, r_2}(i_1, i_2) = R_{r_1, r_2}(i_1, i_2) + iI_{r_1, r_2}(i_1, i_2)\) and write

\[
W_1^{1r_1/m_1} = \cos \left( \frac{2 \pi i r_1}{m_1 N_1} \right) + i \sin \left( \frac{2 \pi i r_1}{m_1 N_1} \right) = C_{r_1}(i_1) + i S_{r_1}(i_1),
\]

\[
W_2^{1r_2/m_2} = \cos \left( \frac{2 \pi i r_2}{m_2 N_2} \right) + i \sin \left( \frac{2 \pi i r_2}{m_2 N_2} \right) = C_{r_2}(i_2) + i S_{r_2}(i_2),
\]

we can finally obtain the following formula for computing \(g(i_1, i_2)\):

\[
g(i_1, i_2) = \frac{\exp(\sigma_1 i_1 + \sigma_2 i_2)}{m_1 m_2 N_1 N_2} \left\{ g_{0,0}(i_1, i_2) + \sum_{r_1=1}^{u_1} \sum_{r_2=-u_2}^{u_2} \right\}.
\]

In numerical computations, the double infinite series in (17) for \(G_{r_1, r_2}(k_1, k_2)\) must be truncated. To avoid the truncation error, we apply in the following the generalized Riemann zeta function to sum the infinite series. Note that the exponential function \(e^{-z}\) is \(2\pi i\)-periodic. Hence, substituting (13) for \(G(s_1, s_2)\), and using the periodic property of \(e^{-z}\), we have

\[
G_{r_1, r_2}(k_1, k_2) = (1 - e^{-s_{k_1, r_1}}) (1 - e^{-s_{k_2, r_2}}) \int \frac{e^{-s_{k_1, r_1}z_1} e^{-s_{k_2, r_2}z_2}}{z_1 z_2} S(s_{k_1, r_1}) S(s_{k_2, r_2}),
\]

where

\[
s_{k_j, r_j} = \sigma_j + \frac{2 \pi}{m_j N_j} (m_j k_j + r_j), \quad j = 1, 2
\]
and

\[
S(s_{kj}, r_j) = \sum_{l=-\infty}^{\infty} \frac{1}{(s_{kj}, r_j + i2\pi l)^2}, \quad j = 1, 2. \tag{30}
\]

The above infinite series can be split as

\[
S(s_{kj}, r_j) = \sum_{l=-\infty}^{\infty} \frac{1}{(s_{kj}, r_j + i2\pi l)^2} = \sum_{l=0}^{\infty} \frac{1}{(s_{kj}, r_j + i2\pi l)^2} + \sum_{l=-1}^{-\infty} \frac{1}{(s_{kj}, r_j + i2\pi l)^2} \tag{31}
\]

\[
= \sum_{l=0}^{\infty} \frac{1}{(s_{kj}, r_j + i2\pi l)^2} + \left[ \sum_{l=0}^{\infty} \frac{1}{(s_{N_j + k_j, r_j} + i2\pi l)^2} \right]^*.
\]

It is noted that the last two semi-infinite series are related to the generalized Riemann zeta function \( \zeta(2, w) \) [14], which is defined by

\[
\zeta(2, w) = \sum_{l=0}^{\infty} \frac{1}{(l + w)^2}. \tag{32}
\]

Therefore, the infinite sum \( S(s_{kj}, r_j) \) in (31) can be expressed in terms of the generalized Riemann zeta function as follows:

\[
S(s_{kj}, r_j) = -\frac{1}{4\pi^2} \left[ \zeta\left(2, \frac{s_{kj}, r_j}{i2\pi}\right) + \zeta^*\left(2, \frac{s_{N_j + k_j, r_j}}{i2\pi}\right) \right], \quad j = 1, 2. \tag{33}
\]

As shown in [15], the generalized Riemann zeta function \( \zeta(2, w) \) can be accurately approximated by the asymptotic expansion

\[
\zeta(2, w) = \sum_{l=0}^{\infty} \frac{1}{(l + w)^3} = \sum_{l=0}^{L-1} \frac{1}{(l + w)^2} + \frac{1}{(L + w)^2} + \frac{1}{2(L + w)^2} + \frac{B_1}{(L + w)^3} - \frac{B_2}{(L + w)^5} + \cdots + (-1)^{n_1+1} \frac{B_{n_2}}{(L + w)^{2n_1+1}} \tag{34}
\]

\[
\triangleq \zeta_{L,n_2}(2, w),
\]

where \( B_1, B_2, \ldots, B_{n_2} \) are the Bernoulli numbers [16]. The absolute error is in the order of

\[
\frac{B_{n_1+1}}{(L + w)^{2n_1+3}}. \tag{35}
\]

For example, if we take \( L = 10 \) and \( n_2 = 2 \), the error can be reduced to \( 10^{-9} \).

4. NUMERICAL RESULTS

This section presents numerical results of some examples to illustrate the effectiveness of the proposed algorithm for computing 2D power series. The tested two-variable functions and their
respective 2D power series are listed below:

\[
(z_1 + z_2)^{30} = \sum_{j=0}^{30} \binom{30}{j} z_1^{30-j} z_2^j,
\]

\[
\exp(-z_1 - z_2) = \sum_{j=0}^{\infty} \sum_{k=0}^{\infty} (-1)^{j+k} \frac{1}{j!k!} z_1^j z_2^k,
\]

\[
\sin(z_1 + z_2) = \sum_{j=0}^{\infty} \sum_{k=0}^{2j+1} (-1)^j \frac{z_1^j}{j!} \frac{z_2^{j+1-k}}{(2j+1-k)!},
\]

\[
\frac{1}{(1 - z_1)(1 - z_2)} = \sum_{j=0}^{\infty} \sum_{k=0}^{j} z_1^j z_2^k.
\]

All numerical computations were carried out in an IBM-PC compatible computer with a 300 MHz Intel Pentium II CPU. In applying the proposed algorithm to compute the 2D power series coefficients of the above four functions, the parameter values \(N_1 = N_2 = 32\), \(L = 1000\), and \(n_z = 9\) were used. For measuring the accuracy of the computed coefficients, the root of mean squared-error

\[
E_{\text{rms}} = \left( \frac{1}{N_1 - 2} \sum_{j=0}^{N_1-2} \sum_{k=0}^{N_2-2} \left( f_{j,k} - \hat{f}_{j,k} \right)^2 \right)^{1/2}
\]

and the maximum absolute error

\[
E_{\text{max}} = \max_{j=0,1,\ldots,N_1-2, k=0,1,\ldots,N_2-2} \left| f_{j,k} - \hat{f}_{j,k} \right| \equiv | f - \hat{f} |,
\]

where \(f_{j,k}\) and \(\hat{f}_{j,k}\) denote, respectively, the exact and computed coefficients, were computed for each test function.

The computed results for various values of \(m = m_1 = m_2\) and \(\sigma = \sigma_1 = \sigma_2\) are shown in Tables 1–4. The CPU times (denoted by \(T_m\) seconds) used to obtain each set of 2D power series coefficients are also included in these tables. As can be seen from these tables, the proposed algorithm can give very accurate coefficients. It is noted that the functions \(G(s_1, s_2)\) associated with the four test functions have the same convergence abscissae \(\sigma_0 = 0\) and \(\sigma_0^2 = 0\). Since the first three test functions are entire functions, the calculated performance indices \(E_{\text{rms}}\) and \(E_{\text{max}}\) shown in Tables 1–3 are insensitive to the parameters \(\sigma_1\) and \(\sigma_2\) chosen. As for the fourth test

<table>
<thead>
<tr>
<th>(m)</th>
<th>(\sigma)</th>
<th>(E_{\text{rms}})</th>
<th>(E_{\text{max}})</th>
<th>(T_m) (sec.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.001</td>
<td>9.43903E−04</td>
<td>1.08643E−02</td>
<td>155117520</td>
</tr>
<tr>
<td></td>
<td>0.01</td>
<td>9.13949E−04</td>
<td>1.04618E−02</td>
<td>155117520</td>
</tr>
<tr>
<td></td>
<td>0.1</td>
<td>9.13904E−04</td>
<td>1.04623E−02</td>
<td>155117520</td>
</tr>
<tr>
<td></td>
<td>0.5</td>
<td>9.61742E−04</td>
<td>1.04621E−02</td>
<td>155117520</td>
</tr>
<tr>
<td>3</td>
<td>0.001</td>
<td>9.13452E−04</td>
<td>1.04675E−02</td>
<td>155117520</td>
</tr>
<tr>
<td></td>
<td>0.01</td>
<td>9.13888E−04</td>
<td>1.04620E−02</td>
<td>155117520</td>
</tr>
<tr>
<td></td>
<td>0.1</td>
<td>9.13905E−04</td>
<td>1.04622E−02</td>
<td>155117520</td>
</tr>
<tr>
<td></td>
<td>0.5</td>
<td>9.14841E−04</td>
<td>1.04620E−02</td>
<td>155117520</td>
</tr>
<tr>
<td>5</td>
<td>0.001</td>
<td>9.13945E−04</td>
<td>1.04980E−02</td>
<td>155117520</td>
</tr>
<tr>
<td></td>
<td>0.01</td>
<td>9.13904E−04</td>
<td>1.04621E−02</td>
<td>155117520</td>
</tr>
<tr>
<td></td>
<td>0.1</td>
<td>9.13906E−04</td>
<td>1.04620E−02</td>
<td>155117520</td>
</tr>
<tr>
<td></td>
<td>0.5</td>
<td>9.15951E−04</td>
<td>1.04621E−02</td>
<td>155117520</td>
</tr>
</tbody>
</table>
### Table 2. Computed results for $\exp(-z_1 - z_2)$.

<table>
<thead>
<tr>
<th>$m$</th>
<th>$\sigma$</th>
<th>$E_{\text{rms}}$</th>
<th>$E_{\text{max}}$</th>
<th>$\hat{f}$</th>
<th>$\tilde{f}$</th>
<th>$T_m$ (sec.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.001</td>
<td>8.72078E-12</td>
<td>1.12777E-10</td>
<td>1</td>
<td>1</td>
<td>4.17</td>
</tr>
<tr>
<td></td>
<td>0.001</td>
<td>8.45709E-12</td>
<td>1.18021E-10</td>
<td>1</td>
<td>1</td>
<td>4.23</td>
</tr>
<tr>
<td></td>
<td>0.01</td>
<td>8.46018E-12</td>
<td>1.18032E-10</td>
<td>1</td>
<td>1</td>
<td>4.18</td>
</tr>
<tr>
<td></td>
<td>0.1</td>
<td>8.46019E-12</td>
<td>1.18032E-10</td>
<td>1</td>
<td>1</td>
<td>4.17</td>
</tr>
<tr>
<td>3</td>
<td>0.0001</td>
<td>8.43340E-12</td>
<td>1.17325E-10</td>
<td>1</td>
<td>1</td>
<td>25.16</td>
</tr>
<tr>
<td></td>
<td>0.001</td>
<td>8.45958E-12</td>
<td>1.18030E-10</td>
<td>1</td>
<td>1</td>
<td>25.16</td>
</tr>
<tr>
<td></td>
<td>0.01</td>
<td>8.46016E-12</td>
<td>1.18031E-10</td>
<td>1</td>
<td>1</td>
<td>25.15</td>
</tr>
<tr>
<td></td>
<td>0.1</td>
<td>8.46019E-12</td>
<td>1.18031E-10</td>
<td>1</td>
<td>1</td>
<td>25.16</td>
</tr>
<tr>
<td>5</td>
<td>0.0001</td>
<td>8.45479E-12</td>
<td>1.17780E-10</td>
<td>1</td>
<td>1</td>
<td>62.89</td>
</tr>
<tr>
<td></td>
<td>0.001</td>
<td>8.46007E-10</td>
<td>1.18033E-10</td>
<td>1</td>
<td>1</td>
<td>62.89</td>
</tr>
<tr>
<td></td>
<td>0.01</td>
<td>8.46016E-12</td>
<td>1.18032E-10</td>
<td>1</td>
<td>1</td>
<td>62.89</td>
</tr>
<tr>
<td></td>
<td>0.1</td>
<td>8.46019E-12</td>
<td>1.18032E-10</td>
<td>1</td>
<td>1</td>
<td>62.89</td>
</tr>
</tbody>
</table>

### Table 3. Computed results for $\sin(z_1 + z_2)$.

<table>
<thead>
<tr>
<th>$m$</th>
<th>$\sigma$</th>
<th>$E_{\text{rms}}$</th>
<th>$E_{\text{max}}$</th>
<th>$\hat{f}$</th>
<th>$\tilde{f}$</th>
<th>$T_m$ (sec.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.0001</td>
<td>2.17895E-11</td>
<td>7.27596E-11</td>
<td>1</td>
<td>1</td>
<td>4.06</td>
</tr>
<tr>
<td></td>
<td>0.001</td>
<td>3.75764E-12</td>
<td>6.74163E-11</td>
<td>1</td>
<td>1</td>
<td>4.06</td>
</tr>
<tr>
<td></td>
<td>0.01</td>
<td>3.75879E-12</td>
<td>6.74483E-11</td>
<td>1</td>
<td>1</td>
<td>4.07</td>
</tr>
<tr>
<td></td>
<td>0.1</td>
<td>3.75884E-12</td>
<td>6.74476E-11</td>
<td>1</td>
<td>1</td>
<td>4.06</td>
</tr>
<tr>
<td>3</td>
<td>0.0001</td>
<td>4.98229E-12</td>
<td>6.73026E-11</td>
<td>1</td>
<td>1</td>
<td>24.34</td>
</tr>
<tr>
<td></td>
<td>0.001</td>
<td>3.76000E-12</td>
<td>6.74589E-11</td>
<td>1</td>
<td>1</td>
<td>24.33</td>
</tr>
<tr>
<td></td>
<td>0.01</td>
<td>3.75882E-12</td>
<td>6.74472E-11</td>
<td>1</td>
<td>1</td>
<td>24.33</td>
</tr>
<tr>
<td></td>
<td>0.1</td>
<td>3.75882E-12</td>
<td>6.74472E-11</td>
<td>1</td>
<td>1</td>
<td>24.29</td>
</tr>
<tr>
<td>5</td>
<td>0.0001</td>
<td>3.95012E-12</td>
<td>6.73026E-11</td>
<td>1</td>
<td>1</td>
<td>60.75</td>
</tr>
<tr>
<td></td>
<td>0.001</td>
<td>3.75799E-12</td>
<td>6.74589E-11</td>
<td>1</td>
<td>1</td>
<td>60.80</td>
</tr>
<tr>
<td></td>
<td>0.01</td>
<td>3.75884E-12</td>
<td>6.74472E-11</td>
<td>1</td>
<td>1</td>
<td>60.80</td>
</tr>
<tr>
<td></td>
<td>0.1</td>
<td>3.75884E-12</td>
<td>6.74472E-11</td>
<td>1</td>
<td>1</td>
<td>60.81</td>
</tr>
</tbody>
</table>

### Table 4. Computed results for $1/(1 - z_1)(1 - z_2)$.

<table>
<thead>
<tr>
<th>$m$</th>
<th>$\sigma$</th>
<th>$E_{\text{rms}}$</th>
<th>$E_{\text{max}}$</th>
<th>$\hat{f}$</th>
<th>$\tilde{f}$</th>
<th>$T_m$ (sec.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.2</td>
<td>3.33142E-03</td>
<td>3.33142E-03</td>
<td>1</td>
<td>1</td>
<td>4.11</td>
</tr>
<tr>
<td></td>
<td>0.3</td>
<td>1.35471E-04</td>
<td>1.35471E-04</td>
<td>1</td>
<td>1</td>
<td>4.12</td>
</tr>
<tr>
<td></td>
<td>0.4</td>
<td>5.52015E-06</td>
<td>5.67599E-06</td>
<td>1</td>
<td>1</td>
<td>4.12</td>
</tr>
<tr>
<td></td>
<td>0.5</td>
<td>4.32147E-06</td>
<td>6.25677E-05</td>
<td>1</td>
<td>1</td>
<td>4.18</td>
</tr>
<tr>
<td></td>
<td>0.6</td>
<td>1.30689E-03</td>
<td>3.27268E-02</td>
<td>1</td>
<td>1</td>
<td>4.12</td>
</tr>
<tr>
<td>3</td>
<td>0.2</td>
<td>9.17546E-09</td>
<td>9.20809E-09</td>
<td>1</td>
<td>1</td>
<td>24.77</td>
</tr>
<tr>
<td></td>
<td>0.3</td>
<td>2.92614E-11</td>
<td>4.30305E-10</td>
<td>1</td>
<td>1</td>
<td>24.77</td>
</tr>
<tr>
<td></td>
<td>0.4</td>
<td>3.75197E-09</td>
<td>5.54841E-08</td>
<td>1</td>
<td>1</td>
<td>24.82</td>
</tr>
<tr>
<td></td>
<td>0.5</td>
<td>1.19523E-06</td>
<td>2.49144E-05</td>
<td>1</td>
<td>1</td>
<td>24.78</td>
</tr>
<tr>
<td></td>
<td>0.6</td>
<td>2.79967E-04</td>
<td>5.85728E-03</td>
<td>1</td>
<td>1</td>
<td>24.77</td>
</tr>
<tr>
<td>5</td>
<td>0.2</td>
<td>4.36671E-12</td>
<td>3.37479E-11</td>
<td>1</td>
<td>1</td>
<td>61.95</td>
</tr>
<tr>
<td></td>
<td>0.3</td>
<td>1.44012E-11</td>
<td>1.86901E-10</td>
<td>1</td>
<td>1</td>
<td>61.95</td>
</tr>
<tr>
<td></td>
<td>0.4</td>
<td>1.98967E-09</td>
<td>3.17127E-08</td>
<td>1</td>
<td>1</td>
<td>61.95</td>
</tr>
<tr>
<td></td>
<td>0.5</td>
<td>1.85970E-06</td>
<td>3.86687E-05</td>
<td>1</td>
<td>1</td>
<td>61.96</td>
</tr>
<tr>
<td></td>
<td>0.6</td>
<td>2.35901E-04</td>
<td>4.77137E-03</td>
<td>1</td>
<td>1</td>
<td>61.96</td>
</tr>
</tbody>
</table>
function, the calculated performance indices shown in Table 4 degrade if the chosen values of $\sigma_1$ and $\sigma_2$ are too small or too large. This is due to the fact that the test function $f(z) = 1/[(1 - z_1)(1 - z_2)]$ has a singularity at $(z_1, z_2) = (1, 1)$ and the associated function $G(s_1, s_2)$ has a singularity at $(s_1, s_2) = (0, 0)$ with multiplicities $(3, 3)$.

As shown in Tables 1–4, the solution accuracy can be effectively improved by incrementing the integer parameters $m_1$ and $m_2$. The increments of $m_1$ and $m_2$ lead the computation to involve more sets of 2D FFT operations. As can be seen from Tables 1–4, the CPU time used to obtain a set of $(N \times N)$ 2D power series coefficients increases proportionally to $(m_1m_2/2 + 1)$. However, since each set of 2D FFT operations in the proposed algorithm can be performed independently, multiple sets of 2D FFT operations do not take more computation time than a single set of 2D FFT operations in a parallel computing environment.

5. CONCLUSION

The method of computing 1D power-series expansion coefficients based on using the numerical inversion of the Laplace transform and the accurate evaluation of the generalized Riemann zeta function has been successfully extended to the 2D case. The algorithm has been devised such that the computation of $N_1 \times N_2$ 2D coefficients can be accurately accomplished by performing $(m_1m_2/2 + 1)$ sets of 2D FFT operations. The computation is thus particularly suitable for modern computers with vector processors and/or multiprocessors. It is believed that after this success of extension, the further generalization of the method to three- or higher-dimensional cases becomes clear and quite straightforward.

REFERENCES