Available online at www.sciencedirect.com
SCIENCE DIRECT®
ELSEVIER Applied Mathematics and Computation 158 (2004) 225–235
APPLIED MATHEMATICS AND COMPUTATION
www.elsevier.com/locate/amc

A novel analytical scheme to compute the n-fold convolution of exponential-sum distribution functions

N.-Y. Ma *, Feng Liu

Department of Materials Science and Engineering, University of Utah, 122 S Central Campus Dr., Salt Lake City, UT 84112, USA

Abstract

A general analytical scheme for computing the n-fold convolution of exponential-sum distribution functions has been developed in this paper. The n-fold convolution is first expressed by multiple sums of recursive integrals. These recursive integrals are then reconstructed with a series of delta functions to avoid separations of integrations. Part of the recursive integrals has been solved analytically by either direct integration or with Maple-like symbolic software packages. The general analytical solution of the n-fold convolution of exponential-sum distribution functions is obtained in two steps: first developing a general pattern of Laplace transform of the recursive integrals, and then performing an inverse Laplace transform operation to the general pattern of the developed Laplace transform. The solution presented in this paper provides another option for computing the n-fold convolution of exponential-sum distribution functions.

Keywords: N-fold convolution; Exponential-sum distribution functions; Laplace transform; Recursive integrals; Reconstructed recursive integrals

1. Introduction

In a previous work [1], the problem of computing the n-fold convolution of exponential-sum distribution functions was studied. A general analytical

* Corresponding author.
E-mail address: nma@eng.utah.edu (N.-Y. Ma).

0096-3003/$ - see front matter
doi:10.1016/j.amc.2003.08.087

multiple-sum solution was derived there with the application of complete multinomial expansion theorem and Laplace transform technique. However, the solution presented there involves rather complex derivation. In this paper, we present a novel analytical scheme with less complexity to compute the n-fold convolution of exponential-sum distribution functions. The new general analytical solution of the n-fold convolution of exponential-sum distribution functions is much simpler to understand conceptually and much easier to implement computationally.

We begin with readdressing the problem as follows.

A density distribution function with exponential sums is expressed as

$$f(t) = \sum_{i=1}^{m} \alpha_i \mathrm{e}^{-\lambda_i t}, \tag{1}$$

where m, a finite positive integer, is the number of exponentials; α_i and λ_i are positive constant real numbers; $\lambda_i \neq \lambda_j$ if $i \neq j$; and $f(t)$ is defined in positive time domain where $t \in [0, \infty)$. The corresponding cumulative distribution function of $f(t)$ is

$$F(t) = \int_0^t f(x)\,\mathrm{d}x. \tag{2}$$

The n-fold convolution of $f(t)$ is defined as

$$f^{(n)}(t) = \int_0^t f^{(n-1)}(x) f(t-x)\mathrm{d}x \tag{3}$$

and the n-fold convolution of $F(t)$ is

$$F^{(n)}(t) = \int_0^t f^{(n)}(x)\mathrm{d}x. \tag{4}$$

Our objective here is to find less complex solutions for (3) and (4) than those presented in [1] provided that $f(t)$ and $F(t)$ are defined in (1) and (2), respectively.

2. The solution with recursive integrals

N-fold convolution has a recursive trait in nature. We shall show in the following theorem that the problem posed above can be reduced to the computation of some simple recursive integrals.

Theorem 1. *The n-fold convolution $f^{(n)}(t)$ of exponential-sum distribution function $f(t)$ defined in* (1) *can be calculated as*

$$f^{(n)}(t) = \sum_{m_1=1}^{m} \sum_{m_2=1}^{m} \cdots \sum_{m_n=1}^{m} \alpha_{m_1}\alpha_{m_2}\cdots\alpha_{m_n} \mathrm{e}^{-\lambda_{m_n} t} \varphi_n(t). \tag{5}$$

Correspondingly, the n-fold convolution $F^{(n)}(t)$ of cumulative distribution function $F(t)$ can be calculated as

$$F^{(n)}(t) = \sum_{m_1=1}^{m} \sum_{m_2=1}^{m} \cdots \sum_{m_n=1}^{m} \alpha_{m_1}\alpha_{m_2}\cdots\alpha_{m_n} \int_0^t \mathrm{e}^{-\lambda_{m_n} x} \varphi_n(x)\,\mathrm{d}x. \tag{6}$$

$\varphi_n(t)$ can be recursively calculated using the following integrals:

$$\varphi_n(t) = \int_0^t \mathrm{e}^{-(\lambda_{m_{n-1}} - \lambda_{m_n})x} \varphi_{n-1}(x)\,\mathrm{d}x, \tag{7}$$

with

$$\varphi_1(t) = 1. \tag{8}$$

Proof. Let us prove Theorem 1 by induction.

For $n = 1$, $f^{(1)}(t) = f(t)$, and obviously Theorem 1 holds.

Assume Theorem 1 is correct for $n = N$. Let us check the situation when $n = N + 1$. From the assumption, we have

$$f^{(N)}(t) = \sum_{m_1=1}^{m} \sum_{m_2=1}^{m} \cdots \sum_{m_N=1}^{m} \alpha_{m_1}\alpha_{m_2}\ldots\alpha_{m_N} \mathrm{e}^{-\lambda_{m_N} t} \varphi_N(t). \tag{9}$$

According to the definition in (3), we can calculate the $N + 1$-fold convolution of $f(t)$ as follows:

$$\begin{aligned}
f^{(N+1)}(t) &= \int_0^t f^{(N)}(x) f(t-x)\,\mathrm{d}x \\
&= \int_0^t \left(\sum_{m_1=1}^{m} \sum_{m_2=1}^{m} \cdots \sum_{m_N=1}^{m} \alpha_{m_1}\alpha_{m_2}\cdots\alpha_{m_N} \mathrm{e}^{-\lambda_{m_N} x} \varphi_N(x) \right) \\
&\quad \times \left(\sum_{m_{N+1}=1}^{m} \alpha_{m_{N+1}} \mathrm{e}^{-\lambda_{m_{N+1}}(t-x)} \right) \mathrm{d}x \\
&= \sum_{m_1=1}^{m} \sum_{m_2=1}^{m} \cdots \sum_{m_N=1}^{m} \sum_{m_{N+1}=1}^{m} \alpha_{m_1}\alpha_{m_2}\cdots\alpha_{m_N}\alpha_{m_{N+1}} \mathrm{e}^{-\lambda_{m_{N+1}} t} \\
&\quad \times \int_0^t \mathrm{e}^{-(\lambda_{m_N} - \lambda_{m_{N+1}})x} \varphi_N(x)\,\mathrm{d}x.
\end{aligned} \tag{10}$$

Let us define

$$\varphi_{N+1}(t) = \int_0^t e^{-(\lambda_{m_N} - \lambda_{m_{N+1}})x} \varphi_N(x)\, dx. \tag{11}$$

Substituting (11) into (10), we have

$$f^{(N+1)}(t) = \sum_{m_1=1}^{m} \sum_{m_2=1}^{m} \cdots \sum_{m_{N+1}=1}^{m} \alpha_{m_1} \alpha_{m_2} \cdots \alpha_{m_{N+1}} e^{-\lambda_{m_{N+1}} t} \varphi_{N+1}(t). \tag{12}$$

Substituting (12) into (4), we acquire

$$F^{(N+1)}(t) = \sum_{m_1=1}^{m} \sum_{m_2=1}^{m} \cdots \sum_{m_{N+1}=1}^{m} \alpha_{m_1} \alpha_{m_2} \cdots \alpha_{m_{N+1}} \int_0^t e^{-\lambda_{m_{N+1}} x} \varphi_{N+1}(x)\, dx. \tag{13}$$

Therefore, for $n = N + 1$, Theorem 1 also holds. Thus, we have proved by induction that for any $n > 0$ Theorem 1 is correct. □

Theorem 1 provides a concise solution for the n-fold convolution of exponential-sum distribution functions. It is quite easy to program the solution with a series of nested for-loops of equal sizes. The solution is not necessarily cheap in cost, but it is simple in concept. If the number of exponentials m in (1) is small, we may directly use Theorem 1 to obtain analytical solutions. For instance, if $m = 1$, then $\lambda_i = \lambda_j$, $\varphi_n(t) = t^{n-1}/(n-1)!$ and $f^{(n)}(t) = \alpha_1^n t^{n-1} e^{-\lambda_1 t}/(n-1)!$. This is the standard solution of the n-fold convolution of one-exponential distribution functions, presented in many textbooks [2].

If $m > 1$, we have to consider separately the situations for $m_{n-1} = m_n$ vs. $m_{n-1} \neq m_n$ in evaluating $\varphi_n(t) = \int_0^t e^{-(\lambda_{m_{n-1}} - \lambda_{m_n})x} \varphi_{n-1}(x)\, dx$ for $n > 1$. Because $\varphi_n(t)$ is recursively evaluated, we would have to use 2^{n-1} separate expressions to represent $\varphi_n(t)$. Thereby, it is necessary to reconstruct $e^{-(\lambda_{m_{n-1}} - \lambda_{m_n})t}$ into some simpler equivalent function forms, so that the separate expressions of $\varphi_n(t)$ can be avoided.

Let us define

$$\delta_n = \begin{cases} 1, & m_{n-1} = m_n, \\ 0, & m_{n-1} \neq m_n \end{cases} \tag{14}$$

and

$$\beta_n = \delta_n + \lambda_{m_{n-1}} - \lambda_{m_n}. \tag{15}$$

The complement of δ_n is $\bar{\delta}_n$. Then, we have

$$\bar{\delta}_n = \begin{cases} 0, & m_{n-1} = m_n, \\ 1, & m_{n-1} \neq m_n \end{cases} \tag{16}$$

and

$$\beta_n = \begin{cases} 1, & m_{n-1} = m_n, \\ \lambda_{m_{n-1}} - \lambda_{m_n}, & m_{n-1} \neq m_n. \end{cases} \tag{17}$$

Consequently, we have the following equality for $n > 1$:

$$\mathrm{e} - (\lambda_{m_{n-1}} - \lambda_{m_n})t = \delta_n + \bar{\delta}_n \mathrm{e} - \beta_n t. \tag{18}$$

Substituting (18) into (7), we obtain the following reconstructed recursive integrals:

$$\varphi_n(t) = \int_0^t (\delta_n + \bar{\delta}_n \mathrm{e}^{-\beta_n x})\varphi_{n-1}(x)\,\mathrm{d}x, \quad n > 1 \tag{19}$$

plus the base case (8).

Therefore, we have reduced the problem of computing the n-fold convolution of exponential-sum distribution functions to solving the recursive integrals (19) and (8).

Next, we discuss two different schemes to solve the recursive integrals.

3. Solution of the recursive integrals by direct integration

The recursive integrals (19) and (8) can be solved straightforwardly by direct integration, using either numerical integration or symbolic integration software packages, such as Maple. For a specific problem with a known exponential-sum distribution function, Maple-like symbolic software packages can generate nice analytical solutions for the recursive integrals with some simple coding. However, Maple-like software packages are not smart enough to recognize the solution pattern for the general problem we are discussing here. As a matter of fact, Maple-like solutions are more or less messed up. Consequently, we shall take direct integration by hand to solve the recursive integrals.

Examining the recursive integrals, we can find that the functions in the recursive integrals has the form of $\sum_i a_i x^{m_i} \mathrm{e}^{-b_i x}$, where, a_i and b_i are arbitrary constant real numbers, and m_i are non-negative integers. Such integrations are guaranteed solvable. The following are sample solutions of the recursive integrals for some small n values

$$\begin{aligned}
\varphi_1(t) &= 1,\\
\varphi_2(t) &= \delta_2 t + \frac{\bar{\delta}_2}{(-\beta_2)}(\mathrm{e}^{-\beta_2} - 1),\\
\varphi_3(t) &= \frac{\delta_2\bar{\delta}_3}{(-\beta_3)^2} - \frac{\bar{\delta}_2\delta_3}{(-\beta_2)^2} + \frac{\bar{\delta}_2\bar{\delta}_3}{(-\beta_3)[-(\beta_3+\beta_2)]} - \frac{\bar{\delta}_2\delta_3}{(-\beta_2)}t\\
&\quad + \frac{\delta_2\delta_3}{2!}t^2 + \frac{\delta_2\bar{\delta}_3}{(-\beta_3)}t\mathrm{e}^{-\beta_3 t} - \frac{\delta_2\bar{\delta}_3}{(-\beta_3)^2}\mathrm{e}^{-\beta_3 t}\\
&\quad - \frac{\bar{\delta}_2\delta_3}{(-\beta_2)^2}\mathrm{e}^{-\beta_2 t} + \frac{\bar{\delta}_2\bar{\delta}_3}{(-\beta_2)}\mathrm{e}^{-\beta_3 t}\left\{\frac{1}{[-(\beta_3+\beta_2)]}\mathrm{e}^{-\beta_2 t} - \frac{1}{(-\beta_3)}\right\}
\end{aligned}$$

Clearly, the analytical solutions of the recursive integrals quickly become extremely complicated with the increase of n. It is difficult to identify the general solution pattern from these sample solutions.

If the goal is to only compute the n-fold convolution, it may be sufficient to use the solutions of the recursive integrals by the direct integrations with the help of Maple-like symbolic integration packages. However, it is often desirable to obtain the general solution pattern of the recursive integrals, so the n-fold convolution can be performed more efficiently, such as as a part of scientific and engineering models. Below, we use Laplace transform technique to solve this problem.

4. Solution of the recursive integrals with Laplace transform

Let us define the Laplace transform of the nth order recursive integral as $g_n(s)$, i.e.

$$g_n(s) = \mathsf{L}[\varphi_n(t)], \tag{20}$$

where L represents the operation of forward Laplace transform. We summarize the general pattern of the Laplace transform of the recursive integrals in the following theorem.

Theorem 2. *The Laplace transform of the recursive integrals expressed in* (19) and (8) *can be arithmetically expressed by the following formula*:

$$g_n(s) = \frac{1}{s}\sum_{i=1}^{2^{n-1}} \frac{\prod_{j=1}^{n-1} \delta_{n-j+1}^{b_{ij}} \bar{\delta}_{n-j+1}^{\bar{b}_{ij}}}{\prod_{j=1}^{n-1}\left(s + \sum_{k=1}^{j} \bar{b}_{ik}\beta_{n-k+1}\right)}, \quad n > 1, \tag{21}$$

$$g_1(s) = \frac{1}{s}, \quad n = 1, \tag{22}$$

where, b_{ij} *is a bit number*, *either* 1 *or* 0, *and* $\bar{b}_{ij}$ *is the complement of* b_{ij}, *i.e.* $\bar{b}_{ij} = 1 - b_{ij}$.

Proof. Let us prove Theorem 2 by induction.

Obviously, for $n = 1$ and $n = 2$, Theorem 2 is correct. Let us assume the theorem is correct for $n = N$, so, we have

$$g_N(s) = \frac{1}{s}\sum_{i=1}^{2^{N-1}} \frac{\prod_{j=1}^{N-1} \delta_{N-j+1}^{b_{ij}} \bar{\delta}_{N-j+1}^{\bar{b}_{ij}}}{\prod_{j=1}^{N-1}\left(s + \sum_{k=1}^{j} \bar{b}_{ik}\beta_{N-k+1}\right)}. \tag{23}$$

From (19), we have

$$\varphi_{N+1}(t) = \int_0^t (\delta_{N+1} + \bar{\delta}_{N+1}\mathrm{e}^{-\beta_{N+1}x})\varphi_N(x)\,\mathrm{d}x. \tag{24}$$

Taking Laplace transform operation to (24), we have

$$g_{N+1}(s) = \frac{1}{s}\{\delta_{N+1}g_N(s) + \bar{\delta}_{N+1}g_N(s+\beta_{N+1})\}. \tag{25}$$

Substituting (23) into (25) and making relevant rearrangement as follows:

$$\begin{aligned}
g_{N+1}(s) =& \frac{1}{s}\left\{\frac{\delta_{N+1}}{s}\sum_{i=1}^{2^{N-1}}\frac{\prod_{j=1}^{N-1}\delta_{N-j+1}^{b_{ij}}\bar{\delta}_{N-j+1}^{\bar{b}_{ij}}}{\prod_{j=1}^{N-1}\left(s+\sum_{k=1}^{j}\bar{b}_{ik}\beta_{N-k+1}\right)}\right.\\
&\left.+\frac{\bar{\delta}_{N+1}}{s+\beta_{N+1}}\sum_{i=1}^{2^{N-1}}\frac{\prod_{j=1}^{N-1}\delta_{N-j+1}^{b_{ij}}\bar{\delta}_{N-j+1}^{\bar{b}_{ij}}}{\prod_{j=1}^{N-1}\left(s+\beta_{N+1}+\sum_{k=1}^{j}\bar{b}_{ik}\beta_{N-k+1}\right)}\right\}\\
=& \frac{1}{s}\left\{\sum_{i=1}^{2^{N-1}}\frac{\delta_{N+1}^{1}\bar{\delta}_{N+1}^{0}\delta_N^{b_{i1}}\bar{\delta}_N^{\bar{b}_{i1}}\delta_{N-1}^{b_{i2}}\bar{\delta}_{N-1}^{\bar{b}_{i2}}\cdots\delta_2^{b_{iN-1}}\bar{\delta}_2^{\bar{b}_{iN-1}}}{(s+0\beta_{N+1})(s+0\beta_{N+1}+\bar{\delta}_{i1}\beta_N)\cdots(s+0\beta_{N+1}+\bar{\delta}_{i1}\beta_N+\bar{\delta}_{i2}\beta_{N-1}+\cdots+\bar{\delta}_{iN-1}\beta_2)}\right.\\
&\left.+\sum_{i=1}^{2^{N-1}}\frac{\delta_{N+1}^{0}\bar{\delta}_{N+1}^{1}\delta_N^{b_{i1}}\bar{\delta}_N^{\bar{b}_{i1}}\delta_{N-1}^{b_{i2}}\bar{\delta}_{N-1}^{\bar{b}_{i2}}\cdots\delta_2^{b_{iN-1}}\bar{\delta}_2^{\bar{b}_{iN-1}}}{(s+1\beta_{N+1})(s+1\beta_{N+1}+\bar{\delta}_{i1}\beta_N)\cdots(s+1\beta_{N+1}+\bar{\delta}_{i1}\beta_N+\bar{\delta}_{i2}\beta_{N-1}+\cdots+\bar{\delta}_{iN-1}\beta_2)}\right\}.
\end{aligned}$$

Extending the dimension of matrix $\{b_{ij}\}$ from $2^{N-1}\times(N-1)$ to $2^N\times N$, and rearranging b_{ij} and the corresponding $\bar{b}_{ij}$ in the above Laplace transform, we have

$$g_{N+1}(s) = \frac{1}{s}\sum_{i=1}^{2^{(N+1)-1}}\frac{\prod_{j=1}^{(N+1)-1}\delta_{(N+1)-j+1}^{b_{ij}}\bar{\delta}_{(N+1)-j+1}^{\bar{b}_{ij}}}{\prod_{j=1}^{(N+1)-1}\left(s+\sum_{k=1}^{j}\bar{b}_{ik}\beta_{(N+1)-k+1}\right)}. \tag{26}$$

Therefore, for $n = N+1$, Theorem 2 is also correct, and we have proved Theorem 2 by induction. □

Now, we need to determine the matrix $\{b_{ij}\}$ for the Laplace transform of the nth order recursive integral. Let us use Bn to represent the matrix, i.e.

$$Bn = \{b_{ij}\}_n, \tag{27}$$

and $\overline{B}n$ to represent the complement of Bn, i.e.

$$\overline{B}n = \{1-b_{ij}\}_n. \tag{28}$$

Let us define $Bn[i]$ as the ith row of the matrix Bn and treat it as a binary integer number, and then $Bn[i][j]$ will be the bit number of b_{ij}. Examining the Laplace transform of recursive integrals, we can find the pattern of $Bn[i]$

$$Bn[i+1] = Bn[i] - 1, \tag{29}$$

$$Bn[1] = \overbrace{11\ldots1}^{n-1}. \tag{30}$$

Correspondingly, we have

$$\overline{B}n[i+1] = \overline{B}n[i] + 1, \tag{31}$$

$$\overline{B}_n[1] = \overbrace{00\ldots0}^{n-1}. \tag{32}$$

This pattern is not surprising at all. The row number of the matrix corresponds to the term number in the Laplace transform. In each term, δ_i ($i = n\ldots2$) in (21) will be either alive (present) or dead (absent). Therefore, $Bn[i]$ is just representing the possible combinations of $\delta_n\ldots\delta_2$ which are either alive or dead. Because each δ_i has two possibilities, the product of all δ_i's will have 2^{n-1} possibilities. That means the total terms in the Laplace transform (21) and the total rows in the matrix Bn (27) will both be 2^{n-1}. For instance, in term 1 of the Laplace transform or row 1 of the matrix Bn, $\delta_n\ldots\delta_2$ are all alive, and $Bn[1] = 11\ldots1$; in term 2^{n-1} of the Laplace transform or row 2^{n-1} of the matrix Bn, $\delta_n\ldots\delta_2$ are all dead, and $Bn[2^{n-1}] = 00\ldots0$. Going from row i to row $(i+1)$, the combination of the states of $\delta_n\ldots\delta_2$ differs by one. We may write a simple program to compute the matrices Bn and $\overline{B}_n$. Fig. 1 shows a summary of a *c*-like code fragment for such computation.

To this point, we have completed the pattern recognition of Laplace transform of the recursive integrals. In order to acquire the general analytical solution of the recursive integrals, we also need to calculate the inverse Laplace transform of (21). The inverse Laplace transform can be easily solved from inverse Laplace transform pairs [3]. However, the inverse Laplace transform pairs require that all the same factors in the denominator of each term in (21) must be combined together. Using the above algorithm to calculate Bn, we can write out the expression of each denominator of (21). Some denominators corresponding to their term numbers are shown in Table 1.

We can also write a piece of code to do the symbolic calculation to obtain the expression of each denominator.

```
B_n[1]=11...1;
        for (i=2; i<=2^{n-1}; i++)
            B_n[i]=B_n[i-1]-1;
        for (i=1; i<=2^{n-1} ; i++)
            for(j=1; j<=n-1; j++)
            {
                _b_ij = B_n[i][j];
                 b_ij = ;1- b_ij ;
            }
```

Fig. 1. A *C*-like code fragment to compute matrix Bn and $\overline{B}n$.

Table 1
Sample expressions of denominators

Term no. i	$\overline{B}n[i]$	Expressions of denominators
1	00...0000	s^n
2	00...0001	$s^{n-1}(s+\beta_2)$
3	00...0010	$s^{n-2}(s+\beta_3)^2$
4	00...0011	$s^{n-2}(s+\beta_3)(s+\beta_3+\beta_2)$
5	00...0100	$S^{n-3}(s+\beta_4)^3$
6	00...0101	$S^{n-3}(s+\beta_4)^2(s+\beta_4+\beta_2)$
7	00...0110	$s^{n-3}(s+\beta_4)(s+\beta_4+\beta_3)^3$
8	00...0111	$s^{n-3}(s+\beta_4)(s+\beta_4+\beta_3)(s+\beta_4+\beta_3+\beta_2)$
	...	
2^{n-2}	01...1111	$s^2(s+\beta_{n-1})(s+\beta_{n-1}+\beta_{n-2})\cdots(s+\beta_{n-1}+\beta_{n-2}+\cdots+\beta_2)$
$2^{n-2}+1$	10...0000	$s(s+\beta_n)^{n-1}$
$2^{n-2}+2$	10...0001	$s(s+\beta_n)^{n-2}(s+\beta_n+\beta_2)$
$2^{n-2}+3$	10...0010	$s(s+\beta_n)^{n-3}(s+\beta_n+\beta_3)^2$
$2^{n-2}+4$	10...0011	$s(s+\beta_n)^{n-3}(s+\beta_n+\beta_3)(s+\beta_n+\beta_3+\beta_2)$
$2^{n-2}+5$	10...0100	$s(s+\beta_n)^{n-4}(s+\beta_n+\beta_4)^3$
$2^{n-2}+6$	10...0101	$s(s+\beta_n)^{n-4}(s+\beta_n+\beta_4)^2(s+\beta_n+\beta_4+\beta_2)$
$2^{n-2}+7$	10...0110	$s(s+\beta_n)^{n-4}(s+\beta_n+\beta_4)(s+\beta_n+\beta_4+\beta_3)^3$
$2^{n-2}+8$	10...0111	$s(s+\beta_n)^{n-4}(s+\beta_n+\beta_4)(s+\beta_n+\beta_4+\beta_3)(s+\beta_n+\beta_4+\beta_3+\beta_2)$
	...	
2^{n-1}	11...1111	$s(s+\beta_n)(s+\beta_n+\beta_{n-1})(s+\beta_n+\beta_{n-1}+\beta_{n-2})\cdots(s+\beta_n+\beta_{n-1}+$ $\beta_{n-2}+\cdots+\beta_2)$

By combining the same factors in each denominator of (21), the Laplace transform of the nth order recursive integral can be re-expressed as follows:

$$g_n(s)=\sum_{i=1}^{2^{n-1}}\frac{\prod_{j=1}^{n-1}\delta_{n-j+1}^{b_{ij}}\bar{\delta}_{n-j+1}^{\bar{b}_{ij}}}{\prod_{j=1}^{p}(s+\gamma_{ij})^{v_{ij}}}. \tag{33}$$

For different values of j in a given denominator γ_{ij} have different expressions, which can be determined from Table 1. v_{ij} are integer numbers, which can also be determined from Table 1.

The inverse Laplace transform of (33) will be the solution of recursive integrals. Thus, we have

$$\varphi_n(t)=\sum_{i=1}^{2^{n-1}}\prod_{j=1}^{n-1}\delta_{n-j+1}^{b_{ij}}\bar{\delta}_{n-j+1}^{\bar{b}_{ij}}\mathsf{L}^{-}\left[\frac{1}{\prod_{j=1}^{p}(s+\gamma_{ij})^{v_{ij}}}\right], \tag{34}$$

where L^{-} represents the operation of inverse Laplace transform. As the result, we have reduced the problem to solving the inverse Laplace transform of $1/\prod_{j=1}^{p}(s+\gamma_{ij})^{v_{ij}}$. These types of problems have been solved in [1], which we will not repeat here.

5. Solution of the *n*-fold convolution with Laplace transform

After solving the recursive integrals with Laplace transform, we now proceed to solve the *n*-fold convolution of exponential-sum distribution functions.

Substituting (34) into (5), we obtain

$$f^{(n)}(t) = \sum_{m_1=1}^{m} \sum_{m_2=1}^{m} \cdots \sum_{m_n=1}^{m} \alpha_{m_1}\alpha_{m_2}\cdots\alpha_{m_n} \mathrm{e}^{-\lambda_{m_n} t} \times \sum_{i=1}^{2^{n-1}} \prod_{j=1}^{n-1} \delta_{n-j+1}^{b_{ij}} \bar{\delta}_{n-j+1}^{\bar{b}_{ij}} \mathsf{L}^{-}\left[\frac{1}{\prod_{j=1}^{p}(s+\gamma_{ij})^{v_{ij}}}\right]. \tag{35}$$

Taking Laplace transform of (6), we have

$$\mathsf{L}[F^{(n)}(t)] = \sum_{m_1=1}^{m} \sum_{m_2=1}^{m} \cdots \sum_{m_n=1}^{m} \alpha_{m_1}\alpha_{m_2}\cdots\alpha_{m_n} \frac{1}{s} g_n(s+\lambda_{m_n}).$$

For $n > 1$, substituting (33) into the above equation, we obtain

$$\mathsf{L}[F^{(n)}(t)] = \sum_{m_1=1}^{m} \sum_{m_2=1}^{m} \cdots \sum_{m_n=1}^{m} \alpha_{m_1}\alpha_{m_2}\cdots\alpha_{m_n} \sum_{i=1}^{2^{n-1}} \frac{\prod_{j=1}^{n-1} \delta_{n-j+1}^{b_{ij}} \bar{\delta}_{n-j+1}^{\bar{b}_{ij}}}{s\prod_{j=1}^{p}(s+\lambda_{m_n}+\gamma_{ij})^{v_{ij}}}.$$

Taking inverse Laplace transform of the above equation, we obtain the following formula to calculate $F^{(n)}(t)$:

$$F^{(n)}(t) = \sum_{m_1=1}^{m} \sum_{m_2=1}^{m} \cdots \sum_{m_n=1}^{m} \alpha_{m_1}\alpha_{m_2}\cdots\alpha_{m_n} \times \sum_{i=1}^{2^{n-1}} \prod_{j=1}^{n-1} \delta_{n-j+1}^{b_{ij}} \bar{\delta}_{n-j+1}^{\bar{b}_{ij}} \mathsf{L}^{-}\left[\frac{1}{s\prod_{j=1}^{p}(s+\lambda_{m_n}+\gamma_{ij})^{v_{ij}}}\right]. \tag{36}$$

The computation of $\mathsf{L}^{-}[1/\{s\prod_{j=1}^{p}(s+\lambda_{m_n}+\gamma_{ij})^{v_{ij}}\}]$ follows the same line as that of $1/\prod_{j=1}^{p}(s+\gamma_{ij})^{v_{ij}}$ at the end of Section 4, and similar solutions can be found in [1].

6. Conclusions

A simplified general analytical solution of the *n*-fold convolution of exponential-sum distribution functions has been developed. The solution has been expressed by some recursive integrals. These recursive integrals are reconstructed with a series of delta functions to avoid separations of integrations. The recursive integrals can be solved analytically by direct integrations, or with Maple-like symbolic software packages, or by Laplace transform technique. A general analytical solution of the *n*-fold convolution of exponential-sum distribution functions with Laplace transform technique has been presented in detail. This solution is easy to understand conceptually, and simple to implement with computers.

References

[1] N.-Y. Ma, R.P. King, A general analytical solution for calculating *n*-fold convolution power of exponential-sum distribution functions, Appl. Math. Comput. 133 (2002) 83–91.
[2] D.R. Cox, Renewal Theory, Chapman and Hall, London, 1962.
[3] G.E. Roberts, H. Kaufman, Table of Laplace Transforms, W.B. Saunders Company, Philadelphia, PA, 1966.