Exponentially-fitted Gauss-Laguerre quadrature rule for integrals over an unbounded interval

D. Conte[a], L. Gr. Ixaru[b,c], B. Paternoster[d], G. Santomauro[e]

[a] *Dipartimento di Matematica, Università degli Studi di Salerno, Via Giovanni Paolo II n. 132, I-84084 Fisciano (Sa), Italy*

[b] *"Horia Hulubei" National Institute of Physics and Nuclear Engineering, Department of Theoretical Physics, P.O.Box MG-6, Bucharest, Romania*

[c] *Academy of Romanian Scientists, 54 Splaiul Independenţei, 050094, Bucharest, Romania*

[d] *Dipartimento di Matematica, Università degli Studi di Salerno, Via Ponte don Melillo, I-84084 Fisciano (Sa), Italy*

[e] *Dipartimento di Matematica e Applicazioni "R. Caccioppoli", Università degli Studi di Napoli "Federico II", Via Cintia, I-80126 Napoli, Italy*

Abstract

New quadrature formulae are introduced for the computation of integrals over the whole positive semiaxis when the integrand has an oscillatory behavior with decaying envelope. The new formulae are derived by exponential fitting, and they represent a generalization of the usual Gauss-Laguerre formulae. Their weights and nodes depend on the frequency of oscillation in the integrand, and thus the accuracy is massively increased. Rules with one up to six nodes are treated with details. Numerical illustrations are also presented.

Keywords:
quadrature formulae on infinite intervals, exponential fitting,
Gauss-Laguerre formulae

1. Introduction

The accurate computation of integrals of oscillatory functions over an

Email addresses: dajconte@unisa.it (D. Conte), ixaru@theory.nipne.ro (L. Gr. Ixaru), beapat@unisa.it (B. Paternoster), giuseppe.santomauro@unina.it (G. Santomauro)

infinite domain is needed in numerous applications in various branches of physics, engineering, and economics, as for inscance in wave absorption and quantum mechanics; see, e.g., [1],[2],[16],[18],[20],[22],[37]. The problem is also on steady interest for mathematicians, see the recent contribution [6] and references therein.

In this paper we consider the numerical computation of the integral

$$I = \int_0^\infty e^{-x} f(x) dx, \tag{1.1}$$

when the integrand $f(x)$ is an oscillatory function of the form

$$f(x) = f_1(x)\sin(\omega x) + f_2(x)\cos(\omega x). \tag{1.2}$$

The coefficients $f_1(x)$ and $f_2(x)$ are assumed smooth enough to be well approximated by polynomials.

We construct Gauss-Laguerre quadrature rules of the form

$$I \simeq \sum_{k=1}^{N} w_k f(x_k), \tag{1.3}$$

where the weights w_k and the nodes x_k, $k = 1, 2, ..., N$ depend on the frequency ω of function $f(x)$. The new rules should be contrasted with the classical Gauss-Laguerre rules [17] whose (constant) weights and nodes are derived on the assumption that the whole $f(x)$ is smooth enough to be well approximated by polynomials. The classical rules actually represent the limit case $\omega \to 0$ of the new ones.

To build up the new rules we use the exponential fitting (EF) approach, which is a well established procedure for the construction of approximation formulae tuned on functions of special forms; form (1.2) is one of these. For a monograph on the EF approach see [27].

Looking back in the history, the first contributions which gradually led to the formulation of what in the meantime became the EF approach are rather old but for a long period it has been believed that this approach is useful only for amending algorithms for ordinary differential equations, [10, 11, 12, 13, 14, 15, 25, 35, 39]. The fact that the EF technique can be applied for many other operations, including numerical differentiation, quadrature or interpolation, became clear much more recently, [24], and since then an important number of contributions have been published in these new domains. In particular, EF-based versions for numerical quadrature have been obtained in [24] for the Simpson rule, in [29, 31, 32] for the more general Newton-Cotes rule, in [26, 34, 30, 38] for the Gauss-Legendre rule, and in [3, 4, 5] for integral equations. However, it is important to underline that in all these cases the quadrature interval is finite, and, more general, the definition interval for $f(x)$ is finite in all existing EF applications, irrespective of area. Outside the EF technique, in [21, 28] the integral of an oscillatory function over a finite domain has been computed by using an approach related to steepest descent methods. The case we treat in this paper, where the integration interval is infinite, is completely new in the context of the EF technique, except for some preliminary results on the same problem recently reported in [9].

The spirit of our work is to adapt the classical Gauss-Laguerre quadrature formulae to the case of integrals of oscillatory functions of the form (1.1). The idea to adapt existing formulae to the computation of particular integrals over infinite intervals has been used also in the recent paper [36], where the authors consider the steepest descent, extrapolation and sequence transformation methods, and they adapt the three methods, by means of an algorithmic refinement, to the computation of three particular semi-infinite integrals, not necessarely with an oscillatory behaviour. A Filon-type approach for the computation of infinite range oscillatory integrals has been considered in the paper [19], where a smoother variation of the weight is accepted but the frequency has to increase with x. The computation of integrals of oscillatory functions over infinite intervals has finally been considered also in the recent work [33], where the idea is to transform the integral into a non oscillatory one, in order to apply the classical Gauss-Laguerre quadrature rules. On the contrary we use a direct approach, by modifying the quadrature rules in order to directly accurately compute the oscillatory integral.

The paper is organized as follows. In Section 2 we present the basic theoretical ingredients for the construction of the new EF-based Gauss-Laguerre quadrature rules, in Section 3 we come with details on the numerical computation of the weights and nodes of these rules, while in Section 4 numerical experiments are carried out. Finally some conclusions are reported in Section 5. The paper also contains an Appendix where the properties of the $\eta_m(Z)$ functions, frequently used in the paper, are recalled.

2. The exponentially-fitted Gauss-Laguerre quadrature rule

The classical Gauss-Laguerre quadrature rule [17] is of the form (1.3), where the weights and the nodes are obtained by imposing that the rule is exact on the functions

$$x^{n-1},\ n = 1, 2, ..., 2N.$$

By defining the functional

$$\mathcal{L}[f(x), \mathbf{a}] = \int_0^\infty e^{-x} f(x)dx - \sum_{k=1}^{N} w_k f(x_k),$$

where $\mathbf{a}$ is a vector with $2N$ components which collects the weights and the nodes, viz. $\mathbf{a} = [w_{1,}w_2, ..., w_N, x_1, x_2, ..., x_N]$, the desired values of the components of $\mathbf{a}$ are obtained by imposing the condition

$$\mathcal{L}[x^{n-1}, \mathbf{a}] = 0,\ n = 1, 2, ..., 2N.$$

The expression of the error is (see Eq. (3.6.3) of [17])

$$e_{GL} = \frac{(N!)^2}{(2N)!} f^{(2N)}(\theta), \quad \theta \in \,]0, +\infty[\,. \tag{2.4}$$

The EF Gauss-Laguerre quadrature rule is instead obtained by imposing that the formula is exact on the functions

$$x^{n-1}e^{\pm\mu x},\ n = 1, 2, ..., N,$$

i.e. by imposing

$$\mathcal{L}[x^{n-1}e^{\pm\mu x}, \mathbf{a}] = 0,\ n = 1, 2, ..., N. \tag{2.5}$$

Theorem 2.1. *The weights and the nodes of the EF Gauss-Laguerre quadrature rule are solution of the nonlinear system*

$$\left\{ \begin{array}{ll} \displaystyle\sum_{k=1}^{N} w_k x_k^{2n-2} \eta_{n-2}(x_k^2 Z) - \frac{2^{n-1}(n-1)!}{(1-Z)^n} = 0, & n = 1, \ldots, N \\ \displaystyle\sum_{k=1}^{N} w_k x_k^{2n-1} \eta_{n-1}(x_k^2 Z) - \frac{2^{n-1}(n-1)!}{(1-Z)^n} = 0, & n = 1, \ldots, N \end{array} \right. , \tag{2.6}$$

where $Z = \mu^2 = -\omega^2$.

Proof: We follow the procedure introduced in [24]. Thus we compute

$$\mathcal{L}[e^{\mu x}, \mathbf{a}] = \frac{1}{1-\mu} - \sum_{k=1}^{N} w_k e^{\mu x_k}, \quad \mathcal{L}[e^{-\mu x}, \mathbf{a}] = \frac{1}{1+\mu} - \sum_{k=1}^{N} w_k e^{-\mu x_k},$$

and use these for expressing

$$G^{+}(Z, \mathbf{a}) = \frac{1}{2}\left[\mathcal{L}[e^{\mu x}, \mathbf{a}] + \mathcal{L}[e^{-\mu x}, \mathbf{a}]\right], \; G^{-}(Z, \mathbf{a}) = \frac{1}{2\mu}\left[\mathcal{L}[e^{\mu x}, \mathbf{a}] - \mathcal{L}[e^{-\mu x}, \mathbf{a}]\right].$$

We obtain

$$G^{+}(Z, \mathbf{a}) = \frac{1}{1-Z} - \sum_{k=1}^{N} w_k \eta_{-1}(x_k^2 Z), \quad G^{-}(Z, \mathbf{a}) = \frac{1}{1-Z} - \sum_{k=1}^{N} w_k x_k \eta_0(x_k^2 Z).$$

Also important are the expressions of the successive derivatives of G^{+} and G^{-} with respect to Z. By using the differentiation properties of the $\eta_m(Z)$ functions (see Appendix) the following expressions result:

$$\begin{cases} G^{+^{(m)}}(Z, \mathbf{a}) = \dfrac{m!}{(1-Z)^{m+1}} - \dfrac{1}{2^m} \displaystyle\sum_{k=1}^{N} w_k x_k^{2m} \eta_{m-1}(x_k^2 Z), \\ G^{-^{(m)}}(Z, \mathbf{a}) = \dfrac{m!}{(1-Z)^{m+1}} - \dfrac{1}{2^m} \displaystyle\sum_{k=1}^{N} w_k x_k^{2m+1} \eta_{m}(x_k^2 Z). \end{cases} \tag{2.7}$$

Since, from [24], the nonlinear system (2.5) is equivalent to

$$\begin{cases} G^{+^{(n-1)}}(Z, \mathbf{a}) = 0 \\ G^{-^{(n-1)}}(Z, \mathbf{a}) = 0 \end{cases}, \; n = 1, 2, ..., N, \tag{2.8}$$

then (2.6) immediately follows. □

We observe that (2.6) represents a nonlinear system of dimension $2N$ in the nodes and the weights, whose solution is a vector $\mathbf{a}$ depending on $Z = -\omega^2$, i.e. on the frequency ω of oscillation:

$$\mathbf{a} = \mathbf{a}(\omega) = [w_1(\omega), w_2(\omega), ..., w_N(\omega), x_1(\omega), x_2(\omega), ..., x_N(\omega)]. \tag{2.9}$$

By setting $\omega = 0$, we obtain the classical Gauss-Laguerre quadrature formulae, in which the nodes $\bar{x}_k := x_k(0)$ and the weights $\bar{w}_k := w_k(0)$ are given by

$$L_N(\bar{x}_k) = 0, \quad \bar{w}_k = \frac{\bar{x}_k}{(N+1)^2 \left[L_{N+1}(\bar{x}_k)\right]^2},$$

where $L_N(x)$ denotes the Laguerre polynomial of degree N.
As for the error of the EF Gauss-Laguerre quadrature rule, the direct application of Eq. (3.57) of [27] for $h = 1$, $\mu = i\omega$, $Z = -\omega^2$, $K = -1$ and $P = N - 1$ gives the following expression for its leading term:

$$lte_{EF} = T(\mathbf{a}(\omega))(D^{(2)} + \omega^2)^N f(0), \tag{2.10}$$

where $D^{(2)} = \frac{d^2}{dx^2}$ and

$$T(\mathbf{a}(\omega)) = \frac{G^+(0, \mathbf{a}(\omega))}{\omega^{2N}} = \frac{1 - \sum_{k=1}^{N} w_k(\omega)}{\omega^{2N}}. \tag{2.11}$$

Remark 2.1. *We have $\lim_{\omega \to 0} T(\mathbf{a}(\omega)) = (N!)^2/(2N)!$, as it is normal because the new rule tends to the classical one in this limit.*

Remark 2.2. *The expression for the genuine error of the EF version is a sum of two terms of form (2.10) but with different arguments in f, viz.:*

$$e_{EF} = T^+(\mathbf{a}(\omega))(D^{(2)}+\omega^2)^N f(\theta^+)+T^-(\mathbf{a}(\omega))(D^{(2)}+\omega^2)^N f(\theta^-), \quad \theta^{\pm}(\omega) \in\,]0, +\infty[$$

where $T^{\pm}$ which satisfy $T^+(\mathbf{a}(\omega)) + T^-(\mathbf{a}(\omega)) = T(\mathbf{a}(\omega))$ can be determined numerically, see [7] for the theory. As also shown in [7], the two forms (leading term and genuine expression) may predict slightly different rates for the error variation when ω is increased. Yet, in both frames the error is found to extinct down, in contrast to the classical rule where it increases as ω^{2N}. See also Sec.4 below.

3. Computation of weights and nodes

In this section we develop an algorithm for the computation of the weights and the nodes of the EF Gauss-Laguerre quadrature rule.
We have to solve the nonlinear algebraic system (2.6). We will use an iteration procedure whose first stage consists in a convenient split of this system with $2N$ equations into two subsystems of N equations each. In the iteration procedure the first subsystem will be used as a linear system for the weights w_k while the second as a nonlinear system for the nodes x_k. Our procedure is somehow related, but not similar, to that used for the EF Gauss-legendre rule, [26].
Each equation in (2.6) contains products of form $w_k x_k^p$ and the idea of the

splitting consists in collecting the equations with the biggest p in the linear subsystem while all the others are retained in the nonlinear subsystem. Specifically, by denoting $s = \left\lfloor \frac{N}{2} \right\rfloor$ and $r = N - s$ the linear and nonlinear subsystems are

$$\left\{ \begin{array}{ll} \displaystyle\sum_{k=1}^{N} w_k x_k^{2n-2} \eta_{n-2}(x_k^2 Z) - \frac{2^{n-1}(n-1)!}{(1-Z)^n} = 0, & n = r+1, \ldots, N \\ \displaystyle\sum_{k=1}^{N} w_k x_k^{2n-1} \eta_{n-1}(x_k^2 Z) - \frac{2^{n-1}(n-1)!}{(1-Z)^n} = 0, & n = s+1, \ldots, N \end{array} \right. . \tag{3.12}$$

and

$$\left\{ \begin{array}{ll} \displaystyle\sum_{k=1}^{N} w_k x_k^{2n-2} \eta_{n-2}(x_k^2 Z) - \frac{2^{n-1}(n-1)!}{(1-Z)^n} = 0, & n = 1, \ldots, r, \\ \displaystyle\sum_{k=1}^{N} w_k x_k^{2n-1} \eta_{n-1}(x_k^2 Z) - \frac{2^{n-1}(n-1)!}{(1-Z)^n} = 0, & n = 1, \ldots, s, \end{array} \right. . \tag{3.13}$$

respectively. Remember that $Z = -\omega^2$.

Example 3.1. *Let us consider the case N=1. Then the systems* (3.12) *and* (3.13) *lead to:*

$$\left\{ \begin{array}{c} w_1 \eta_{-1}(x_1^2 Z) - \frac{1}{1-Z} = 0 \\ w_1 x_1 \eta_0(x_1^2 Z) - \frac{1}{1-Z} = 0 \end{array} \right.$$

i. e.,

$$\left\{ \begin{array}{c} w_1 \eta_{-1}(-x_1^2 \omega^2) = \frac{1}{1+\omega^2} \\ w_1 x_1 \eta_0(-x_1^2 \omega^2) = \frac{1}{1+\omega^2} \end{array} \right.$$

whose analytical solutions are:

$$x_1(\omega) = \frac{\arctan(\omega)}{\omega} + \frac{k\pi}{\omega}, \quad k \in \mathbb{Z},$$

$$w_1(\omega) = \left\{ \begin{array}{ll} \dfrac{1}{\sqrt{1+\omega^2}}, & |k| \textit{ even} \\ -\dfrac{1}{\sqrt{1+\omega^2}}, & |k| \textit{ odd} \end{array} \right. . \tag{3.14}$$

We observe that when $k = 0$, the unique EF node $x_1(\omega) = \frac{\arctan(\omega)}{\omega} \in [-\pi/(2\omega), \pi/(2\omega)]$ tends to classic Gauss-Laguerre node $x_1 = 1$ as ω goes to zero. Also the EF weight $w_1(\omega) = \frac{1}{\sqrt{1+\omega^2}}$ tends to classic Gauss-Laguerre weight $w_1 = 1$ as ω goes to zero.

Example 3.2. *For N=3 the linear system (3.12) and the nonlinear system (3.13) have the forms*

$$\begin{bmatrix} x_1^4\eta_1(x_1^2Z) & x_2^4\eta_1(x_2^2Z) & x_3^4\eta_1(x_3^2Z) \\ x_1^3\eta_1(x_1^2Z) & x_2^3\eta_1(x_2^2Z) & x_3^3\eta_1(x_3^2Z) \\ x_1^5\eta_2(x_1^2Z) & x_2^5\eta_2(x_2^2Z) & x_3^5\eta_2(x_3^2Z) \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix} = \begin{bmatrix} \frac{8}{(1-Z)^3} \\ \frac{2}{(1-Z)^2} \\ \frac{8}{(1-Z)^3} \end{bmatrix}$$

and

$$\begin{cases} w_1\eta_{-1}(x_1^2Z) + w_2\eta_{-1}(x_2^2Z) + w_3\eta_{-1}(x_3^2Z) - \frac{1}{1-Z} = 0 \\ w_1x_1^2\eta_0(x_1^2Z) + w_2x_2^2\eta_0(x_2^2Z) + w_3x_3^2\eta_0(x_3^2Z) - \frac{2}{(1-Z)^2} = 0 \\ w_1x_1\eta_0(x_1^2Z) + w_2x_2\eta_0(x_2^2Z) + w_3x_3\eta_0(x_3^2Z) - \frac{1}{1-Z} = 0 \end{cases}$$

respectively.

Remark 3.1. *The linear system* (3.12) *in the weights $w = (w_1, \ldots, w_N)^T$ and the nonlinear system* (3.13) *in the nodes $x = (x_1, \ldots, x_N)^T$ can be written as*

$$A(Z, x)w = b(Z), \tag{3.15}$$

and as

$$F(Z, w, x) = D(Z, x)w - d(Z) = 0, \tag{3.16}$$

respectively, where

$$A_{ij}(Z, x) = \begin{cases} x_j^{2(i+r-1)}\eta_{i+r-2}(x_j^2Z), & i = 1, \ldots, s, \\ \\ x_j^{2i-1}\eta_{i-1}(x_j^2Z), & i = s+1, \ldots, N, \end{cases} \quad j = 1, \ldots, N, \tag{3.17}$$

$$b_i(Z) = \begin{cases} \dfrac{2^{i+r-1}(i+r-1)!}{(1-Z)^{i+r}}, & i = 1, \ldots, s, \\ \\ \dfrac{2^{i-1}(i-1)!}{(1-Z)^{i}}, & i = s+1, \ldots, N, \end{cases},$$

$$D_{ik}(Z,x) = \begin{cases} x_k^{2i-2}\eta_{i-2}(x_k^2 Z), & i = 1,\dots,r, \\ \\ x_k^{2(i-r)-1}\eta_{i-r-1}(x_k^2 Z), & i = r+1,\dots,N, \end{cases} \quad , \quad k = 1,\dots,N, \tag{3.18}$$

$$d_i(Z) = \begin{cases} \dfrac{2^{i-1}(i-1)!}{(1-Z)^i}, & i = 1,\dots,r, \\ \\ \dfrac{2^{i-r-1}(i-r-1)!}{(1-Z)^{i-r}}, & i = r+1,\dots,N. \end{cases} \tag{3.19}$$

The numerical solution of the nonlinear system (3.16) is carried out by means of the Newton's iterative method. On each iteration, the new, corrected values of x, denoted by x^{new}, are determined in terms of the input node values x by the formula

$$x^{new} = x + \Delta x.$$

Here the deviation Δx is the solution of the linear system

$$B(Z,w,x)\Delta x = -D(Z,x)w + d(Z), \tag{3.20}$$

where the matrix B denotes the Jacobian of $F(Z,w,x)$ with respect to x, and the matrix D and vector d are defined in (3.18) and (3.19), respectively. The Jacobian matrix B can be computed by using the differentiation properties of the $\eta_m(Z)$ functions, as shown in the following theorem.

Theorem 3.1. *The Jacobian matrix B of the Newton iterative method* (3.20) *is*

$$B(Z,w,x) = C(Z,x)\cdot diag(w) + D(Z,x)\cdot J_x w, \tag{3.21}$$

where the matrix $J_x w$ is computed by solving

$$A(Z,x)\cdot J_x w = -J_x A \cdot diag(w). \tag{3.22}$$

Here $diag(w)$ is the diagonal matrix

$$diag(w) = (w_i \delta_{ij})_{i,j=1,\dots,N}\,.$$

$$C_{ik}(Z,x) = \begin{cases} x_k^{2i-3}\left[(2i-2)\eta_{i-2}(x_k^2 Z) + x_k^2 Z \eta_{i-1}(x_k^2 Z)\right], \\ \qquad\qquad\qquad\qquad\qquad\qquad i = 1,\dots,r, \\ \\ x_k^{2(i-r-1)}\left[(2(i-r)-1)\,\eta_{i-r-1}(x_k^2 Z) + x_k^2 Z \eta_{i-r}(x_k^2 Z)\right], \\ \qquad\qquad\qquad\qquad\qquad\qquad i = r+1,\dots,N, \end{cases} \tag{3.23}$$

for $k = 1,\dots,N$

$$(J_x A)_{ij} = \begin{cases} x_j^{2(i+r)-3}\left[2(i+r-1)\eta_{i+r-2}(x_j^2 Z) + x_j^2 Z \eta_{i+r-1}(x_j^2 Z)\right], \\ \qquad\qquad\qquad\qquad\qquad\qquad i = 1,\dots,s, \\ \\ x_j^{2(i-1)}\left[(2i-1)\eta_{i-1}(x_j^2 Z) + x_j^2 Z \eta_i(x_j^2 Z)\right] \\ \qquad\qquad\qquad\qquad\qquad\qquad i = s+1,\dots,N, \end{cases} \tag{3.24}$$

for $j = 1,\dots,N$*, and the matrices* D *and* A *are given by* (3.17) *and* (3.18)*, respectively.*

Proof:

From (3.16), by observing that the element D_{ik} of the matrix $D(Z,x)$ in (3.18) depends only on Z and on the variable x_k, that is $D_{ik}(Z,x) = D_{ik}(Z,x_k)$, the Jacobian matrix B can be computed as

$$B_{ij}(Z,w,x) = \frac{\partial F_i(Z,w,x)}{\partial x_j} = \frac{\partial}{\partial x_j}\left[\textstyle\sum_{k=1}^N w_k(x) D_{ik}(Z,x_k) - d_i(Z)\right] =$$
$$= \sum_{k=1}^N \frac{\partial w_k}{\partial x_j} D_{ik} + \sum_{k=1}^N w_k \frac{\partial D_{ik}}{\partial x_k}\delta_{k,j}.$$

So we obtain

$$B_{ij}(Z,w,x) = \sum_{k=1}^N \frac{\partial w_k}{\partial x_j} D_{ik} + w_j \frac{\partial D_{ij}}{\partial x_j}. \tag{3.25}$$

By defining the matrix $\mathrm{J}_x w$ as

$$\mathrm{J}_x w = \left(\frac{\partial w_i}{\partial x_j}\right)_{i,j=1,\dots,N},$$

and by observing that the matrix $C(Z,x)$ defined in (3.23) satisfies

$$C_{ik} = \left(\frac{\partial D_{ik}}{\partial x_k}\right)_{i,k=1,\dots,N}$$

we can rewrite the Jacobian (3.25) as (3.21).

For the computation of the matrix $J_x w$, we start from the linear system in (3.15). By definition of the matrix $A(Z, x)$, we remind that the element A_{ij} depends only on Z and on the variable x_j, that is $A_{ij}(Z, x) = A_{ij}(Z, x_j)$, for $i, j = 1, \dots, N$, and the element b_i depends only on Z, that is $b_i = b_i(Z)$. Then we make the derivative with respect to x_j of the i-th equation of the linear system (3.15), obtaining

$$\begin{aligned} 0 = \frac{\partial b_i}{\partial x_j} &= \frac{\partial}{\partial x_j}\left(A \cdot w\right)_i = \frac{\partial}{\partial x_j}\left[\sum_{k=1}^{N} A_{ik} \cdot w_k\right] = \sum_{k=1}^{N}\left[\frac{\partial}{\partial x_j}\Big(A_{ik} \cdot w_k\Big)\right] \\ &= \sum_{k=1}^{N}\left[\frac{\partial A_{ik}}{\partial x_j} \cdot w_k + A_{ik} \cdot \frac{\partial w_k}{\partial x_j}\right] = \sum_{k=1}^{N}\left[\frac{\partial A_{ik}}{\partial x_j} \cdot \delta_{kj} \cdot w_k\right] + \sum_{k=1}^{N}\left[A_{ik} \cdot \frac{\partial w_k}{\partial x_j}\right]. \end{aligned}$$

So, in matrix form, we have that

$$\mathrm{J}_x A \cdot \mathrm{diag}(w) + A \cdot \mathrm{J}_x w = 0,$$

which is equivalent to (3.22), where

$$\mathrm{J}_x A = \left(\frac{\partial A_{ij}}{\partial x_j}\right)_{i,j=1,\dots,N},$$

which gives (3.24).

□

To summarize, each iteration of the Newton's method, which takes the vector x for input to compute correspondingly updated values for the vectors of weights and of nodes, requires to:

- solve the linear system (3.15) to update the vector of weights w;
- solve the linear system (3.20) after computing the matrix $\mathrm{J}_x w$ from (3.22) and the matrix B as in (3.21). Note that (3.22) for matrix $\mathrm{J}_x w$ consists in N linear systems having the same coefficient matrix A and different second hand side for each column. This is an important ingredient for an efficient computation of the whole $\mathrm{J}_x w$.

4. Numerical illustrations

In this section we give some technical details on how the effective numerical computation of the weights and nodes of the new rules should be carried out, and report on two numerical experiments in which the classical and the new EF-based rules are compared for accuracy. The computations have been done on a node with CPU Intel Xeon 6 core X5690 3,46GHz, belonging to the E4 multi-GPU cluster of Mathematics Department of Salerno University.

4.1. EF Gauss-Laguerre formulae for $N = 1, 2, ..., 6$

As shown in Example 3.1, in the case $N = 1$ the weights and the nodes can be computed directly; their expressions are given in (3.14). This no more possible for bigger N such that for each $N \geq 2$ we use the numerical algorithm described in the previous section, based on Newton's iterative process. The important issue is how the starting vector of nodes should be taken in order to ensure a fast convergence of the iteration process. We opted for the idea of taking a form inspired from (3.14): for each given N and ω we take the initial approximation x_k^* of the form

$$x_k^*(\omega) = \bar{x}_k \frac{\arctan(\alpha_k \omega)}{\alpha_k \omega}, \quad k = 1, \ldots, N. \tag{4.26}$$

Here $\bar{x}_k$ are the nodes of the N-th degree Laguerre polynomial and α_k are suitable chosen constants determined after a long set of experimental investigations. The values of α_k for $N = 2, 3, 4, 5$ and 6 are listed in Table 1. The number of iterations needed in order to obtain an accuracy of 10^{-14} is around 10 in all cases.

It is also worth noticing that in our procedure starting data are required only for the nodes, in contrast to the iteration procedure developed in [26] for the EF Gauss-Legendre rule, where starting values were required also for the weights.

The variation with ω of the weights and of the nodes for $N = 1, 2, 3, 4, 5$ and 6, and for ω between 0 and 50, are presented in Figs. 1, 2 and 3. We observe that in all these cases the weights are inside $[0, 1]$. Moreover the weights and the nodes tend to zero as ω increases. Due to the oscillatory behaviour of η functions for negative $Z = -\omega^2$, different solutions may exist also for $N \geq 2$, as happens in Example 3.1 for $N = 1$. We choose the initial approximation (4.26) in such a way that all the coefficients of the EF Gauss-Laguerre formulae tend to classical ones when ω goes to zero, as shown in Figures 1, 2 and 3. However, the existence of further solutions may have only a minor influence on the accuracy of the new quadrature rule. As a matter of fact, if we consider the expression of $T(\mathbf{a}(\omega))$ in (2.11), we observe

Table 1: Values of α_i for $N = 2, 3, 4, 5, 6$.

N	ω	α_1	α_2	α_3	α_4	α_5	α_6
2	$0 \le \omega \le 50$	2/3	4/3				
3	$0 \le \omega \le 50$	0.500	1.000	1.500			
4	$0 \le \omega \le 50$	0.500	0.750	1.085	1.565		
5	$0 \le \omega < 6$	0.465	0.670	0.905	1.205	1.600	
	$6 \le \omega < 15$	0.465	0.670	0.905	1.205	1.610	
	$15 \le \omega \le 50$	0.465	0.670	0.905	1.205	1.620	
6	$0 \le \omega < 3.5$	0.443	0.560	0.735	0.940	1.210	1.600
	$3.5 \le \omega < 7.5$	0.443	0.585	0.765	0.975	1.245	1.625
	$7.5 \le \omega < 10$	0.443	0.585	0.770	0.995	1.265	1.640
	$10 \le \omega < 15$	0.443	0.600	0.788	1.005	1.275	1.650
	$15 \le \omega \le 50$	0.443	0.605	0.795	1.020	1.290	1.665

that it shows a decrease like ω^{-2N}. Any different values of x_k and w_k, if they exist, will affect only the numerator in $T(\mathbf{a}(\omega))$, while the decrease of $T(\mathbf{a}(\omega))$ as ω^{-2N} is untouched. As regard the accuracy in the computation of weigths and nodes, it is worth mentioning that the condition number of Jacobian matrix B in (3.20) increases with ω and N. Therefore for values of ω outside the considered range $[0, 50]$ and for $N > 6$, the algorithm can show instability, see for instance [26].

4.2. Numerical tests

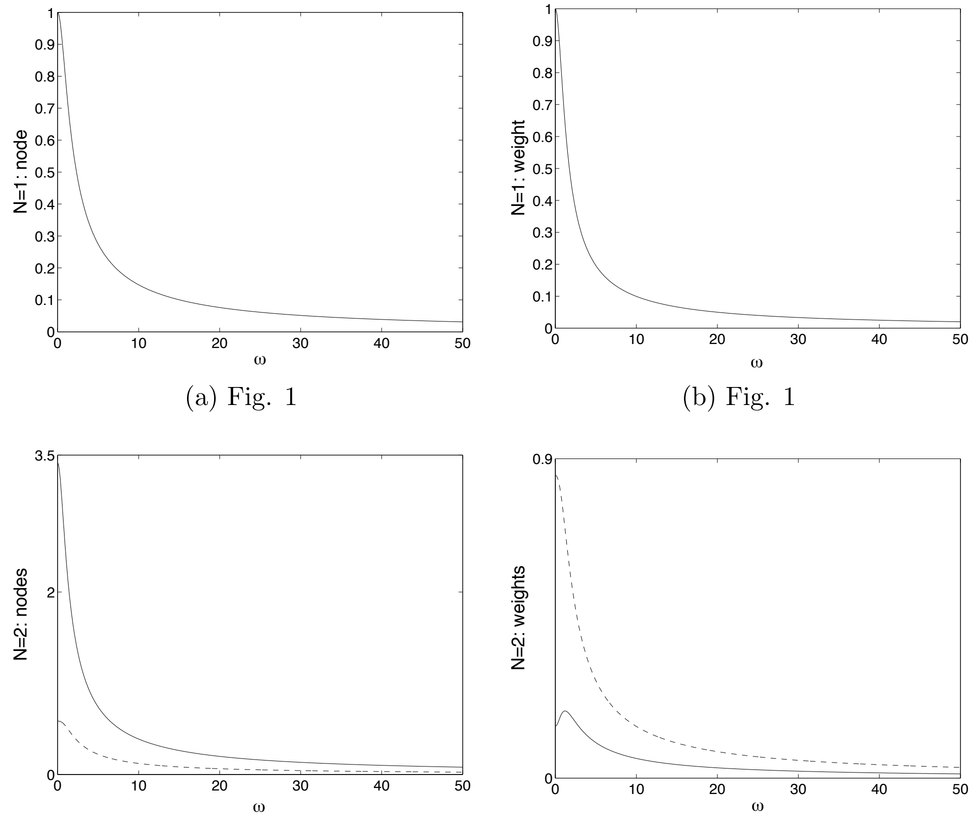
Test case 1. We consider the function

$$f(x) = x\cos(\omega x) + x\sin(\omega x), \tag{4.27}$$

for which we have

$$\int_0^\infty e^{-x} f(x)dx = \frac{1 + 2\omega - \omega^2}{(1+\omega^2)^2}. \tag{4.28}$$

In Table 2 we compare the absolute errors $|I_{exact} - I_{comput}|$ of the results from classical and EF-based Gauss-Laguerre rules for $N = 3, 4$ and various values of ω. We observe that the error from classical version is within the round-off



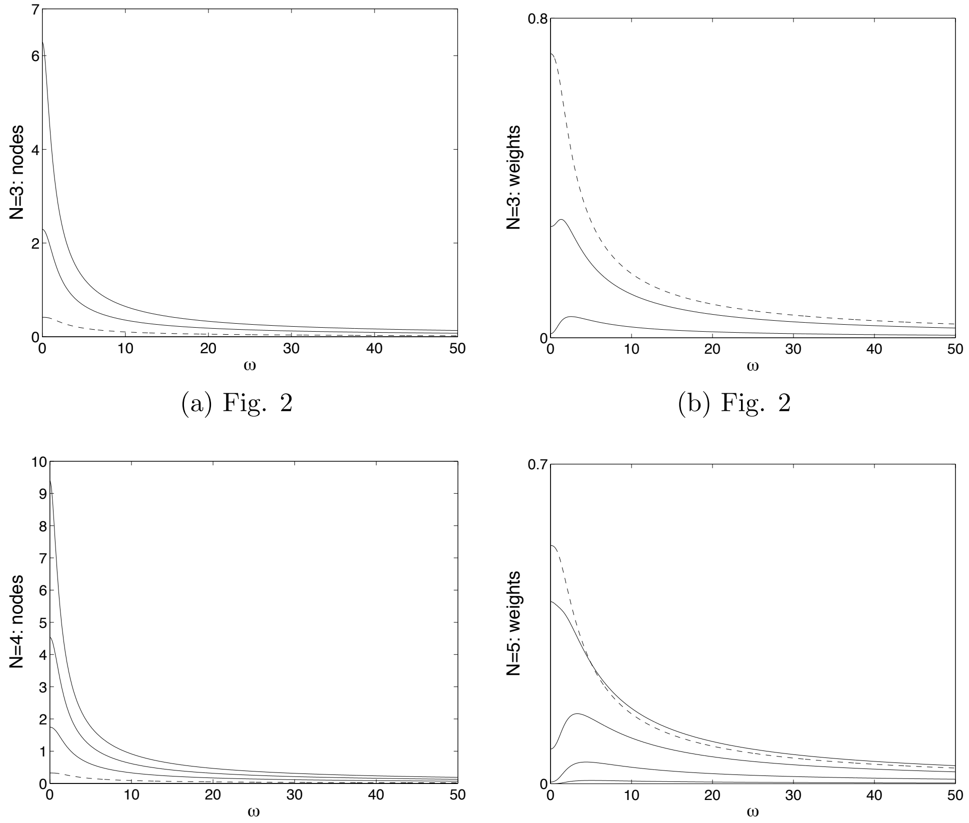
(a) Fig. 1

(b) Fig. 1

(c) Fig. 1

(d) Fig. 1

Figure 1: Variation with ω of the nodes and the weights of the N-point EF Gauss-Laguerre rule. (a) $N = 1$: node x_1; (b) $N = 1$: weight w_1; (c) $N = 2$: nodes x_1 (dashed), x_2 (solid); (d) $N = 2$: weights w_1 (dashed), w_2 (solid).



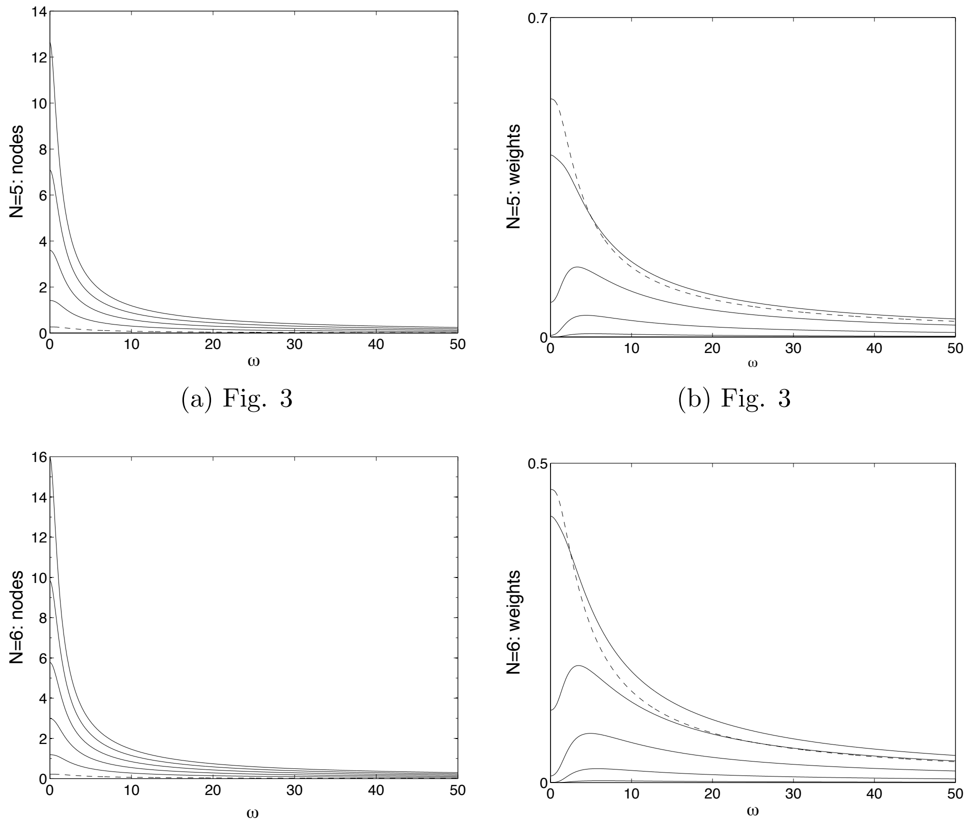
(a) Fig. 2

(b) Fig. 2

(c) Fig. 2

(d) Fig. 2

Figure 2: Variation with ω of the nodes and the weights of the N-point EF Gauss-Laguerre rule. (a) $N = 3$: nodes x_1 (dashed) $\leq x_2 \leq x_3$ (solid); (b) $N = 3$: weights w_1 (dashed) $\geq w_2 \geq w_3$ (solid); (c) $N = 4$: nodes x_1 (dashed) $\leq x_2 \leq x_3 \leq x_4$ (solid); (d) $N = 4$: weights w_1 (dashed) $\geq w_2 \geq w_3 \geq w_4$ (solid).



(a) Fig. 3

(b) Fig. 3

(c) Fig. 3

(d) Fig. 3

Figure 3: Variation with ω of the nodes and the weights of the N-point EF Gauss-Laguerre rule. (a) $N = 5$: nodes x_1 (dashed) $\leq x_2 \leq x_3 \leq x_4 \leq x_5$ (solid); (c) $N = 5$: weights w_1 (dashed), w_2 (solid) $\geq w_3 \geq w_4 \geq w_5$; (c) $N = 6$: nodes x_1 (dashed) $\leq x_2 \leq x_3 \leq x_4 \leq x_5 \leq x_6$ (solid); (d) $N = 6$: weights w_1 (dashed), w_2 (solid) $\geq w_3 \geq w_4 \geq w_5 \geq w_6$.

margin for $\omega = 0$, but abnormally big for the other values. The result is just normal because this version is exact if $f(x)$ is a polynomial of the $(2N-1)$-th degree at most, and this holds true only when $\omega = 0$ (where it becomes a first degree polynomial, actually). For contrast, the EF-based version is affected only by round-off error for all ω. This is also normal because for this test function the new rule is exact irrespective of ω.

N	rules	$\omega = 0$	$\omega = 10$	$\omega = 20$	$\omega = 30$	$\omega = 40$	$\omega = 50$
3	Classic	1.11e-16	1.22e+00	6.04e-01	7.77e-01	1.23e+00	8.62e-01
	EF	1.11e-16	4.51e-17	1.64e-17	4.98e-18	6.18e-18	5.69e-18
4	Classic	2.24e-12	4.95e-01	6.74e-01	2.62e-01	1.13e+00	8.14e-02
	EF	2.24e-12	3.58e-15	1.35e-16	2.09e-16	2.50e-17	1.46e-16

Table 2: Error produced by the EF Gauss-Laguerre rule with $N = 3, 4$ on problem (4.27).

Test case 2. The function

$$f(x) = \cos[(\omega + 1)x] \tag{4.29}$$

is of form (1.2) with $f_1(x) = \sin(x)$ and $f_2(x) = \cos(x)$, and

$$\int_0^\infty e^{-x} f(x)dx = \frac{1}{1 + (1 + \omega)^2}. \tag{4.30}$$

In Tables 3 and 4 we report the results obtained by the classical and the EF rule with $N = 5$ and $N = 6$ for different values of ω. The improvement in accuracy with the new rule is impressive. For a better insight into the things, in Fig. 4 we plot the variation with ω of the errors from the two rules. The behaviors of the two errors confirm what we qualitatively expect on the basis of Eqs.(2.4) and (2.10). Indeed, for the classical rule the error is given by Eq.(2.4) which is a product of a constant and $f^{(2N)}$. For functions of form (1.2), $f^{(2N)}$ will contain a term with ω^{2N} and therefore, when ω is increased, the error is also expected to increase.

For the EF Gauss-Laguerre rule the error is given by Eq.(2.10). Here the front factor has the classical value when $\omega = 0$ but it tends to behave like

$1/\omega^{2N}$ when ω is increased; this is because the sum of the weights in the numerator tends to zero for big ω. The other factor, i.e. $\left(\omega^2 + D^{(2)}\right)^N f$, increases only as ω^N so that, altogether, at large ω the error decreases as ω^{-N}, a remarkable fact, indeed. This also suggests that the error decrease is faster and faster when N is increased. This property is also nicely confirmed in Fig. 4.

$N = 5$	$\omega = 0$	$\omega = 10$	$\omega = 20$	$\omega = 30$	$\omega = 40$	$\omega = 50$
Classic	5.41e-04	9.32e-01	3.88e-01	2.30e-01	1.05e-01	3.52e-02
EF	5.41e-04	2.10e-06	6.04e-08	6.39e-09	1.24e-09	3.44e-10

Table 3: Error produced by the five-point Gauss-Laguerre quadrature rule on problem (4.29).

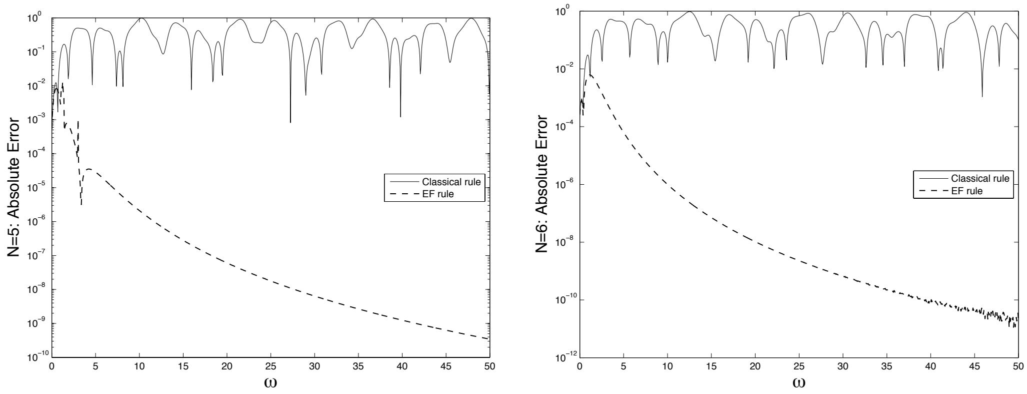
$N = 6$	$\omega = 0$	$\omega = 10$	$\omega = 20$	$\omega = 30$	$\omega = 40$	$\omega = 50$
Classic	2.62e-04	1.70e-02	5.04e-01	6.49e-01	5.34e-01	1.00e-01
EF	2.62e-04	9.96e-07	1.03e-08	6.47e-10	9.35e-11	3.16e-11

Table 4: Error produced by the six-point Gauss-Laguerre quadrature rule on problem (4.29).

5. Conclusions

We constructed a new class of EF Gauss-Laguerre rules for the computation of integrals of oscillating functions over infinite intervals. We developed an algorithm for the computation of the weights and the nodes and demonstrated the massive improvement in accuracy provided by the new formulae when the frequency of oscillation increases.

Acknowledgement One of the authors (L.Gr.I.) has been partially supported by the project PN-II-ID-PCE-2011-3-0092 of the Romanian Ministry of Education and Research.



(a) Fig. 4

(b) Fig. 4

Figure 4: The ω dependence of the errors produced by classic (solid) and EF (dashed) Gauss-Laguerre quadrature rule for $N = 5$ (a) and $N = 6$ (b).

Appendix

The set of functions $\eta_m(Z)$, $m = -1, 0, 1, 2, \dots$ has been originally introduced in [23] in the context of CP methods for the Schrödinger equation. The functions $\eta_m(Z)$ with $m = -1, 0$ are first defined by some formulae, viz.:

$$\eta_{-1}(Z) = \begin{cases} \cos(|Z|^{1/2}) & \text{if } Z \leq 0 \\ \\ \cosh(Z^{1/2}) & \text{if } Z > 0 \end{cases}, \ \eta_0(Z) = \begin{cases} \sin(|Z|^{1/2})/|Z|^{1/2} & \text{if } Z < 0 \\ 1 & \text{if } Z = 0 \\ \sinh(Z^{1/2})/Z^{1/2} & \text{if } Z > 0 \end{cases} \tag{A.1}$$

and those with $m > 0$ are further generated by recurrence

$$\eta_m(Z) = \frac{1}{Z}[\eta_{m-2}(Z) - (2m-1)\eta_{m-1}(Z)], \quad m = \ 1,\ 2,\ 3, \dots \tag{A.2}$$

if $Z \neq 0$, and by following values at $Z = 0$:

$$\eta_m(0) = \frac{1}{(2m+1)!!}, \quad m = \ 1,\ 2,\ 3, \dots \tag{A.3}$$

The differentiation of these functions is of direct concern for this paper. The rule is

$$\eta'_m(Z) = \frac{1}{2}\eta_{m+1}(Z)\,,\ m = -1,\ 0,\ 1,\ 2,\ 3, \dots \tag{A.4}$$

For more details on these functions see [8, 27] or the Appendix of [24].

References

[1] G. Arfken, Mathematical Methods for Physicists, third ed., Academic Press, Orlando, FL, 1985.

[2] G. Bao, W. Sun, *A fast algorithm for the electromagnetic scattering from a large cavity*, SIAM J. Sci. Comput. 27, 553-574, 2005.

[3] A. Cardone, M. Ferro, L. Gr. Ixaru and B. Paternoster, *A Family of Exponential Fitting Direct Quadrature Methods for Volterra Integral Equations*, AIP Conf. Proc. 1281, 2204-2207 (2010).

[4] A. Cardone, L. Gr. Ixaru and B. Paternoster, *Exponential fitting direct quadrature methods for Volterra integral equations*, Numerical Algorithms 55, 467–480, 2010.

[5] A. Cardone, B. Paternoster, G. Santomauro, *Exponential fitting quadrature rule for functional equations*, AIP Conference Proceedings, AIP Conf. Proc. 1479, 1169-1172 Springer (2012).

[6] Ruyun Chen, *Numerical approximations to integrals with a highly oscillatory Bessel kernel*, Applied Numerical Mathematics 62, 636–648, 2012.

[7] J. P. Coleman, L. Gr. Ixaru, *Truncation errors in exponential fitting for oscillatory problems*, SIAM J. Numer. Anal. 44, 1441–1465, 2006.

[8] D. Conte, E. Esposito, B. Paternoster, L. Gr. Ixaru, *Some new uses of the $\eta_m(Z)$ functions*, Comput. Phys. Commun. 181, 128–137, 2010.

[9] D. Conte, B. Paternoster, G. Santomauro, *An exponentially fitted quadrature rule over unbounded intervals*, AIP Conf. Proc. 1479, 1173-1176 Springer (2012).

[10] R. D'Ambrosio, E. Esposito, B. Paternoster, *Exponentially fitted two-step hybrid for $y'' = f(x, y)$*, Journal Comp. and Appl. Math. 235, 4888-4897, 2011.

[11] R. D'Ambrosio, E. Esposito, B. Paternoster, *Parameter estimation in exponentially fitted hybrid methods for second order differential problems*, J. Math. Chem. 50 (1), 155-168 (2012).

[12] R. D'Ambrosio, E. Esposito, B. Paternoster, *Exponentially fitted two-step Runge-Kutta methods: construction and parameter selection*, Appl. Math. Comput 218 7468-7480 (2012).

[13] R. D'Ambrosio, M. Ferro, B. Paternoster, *Two-step hybrid collocation methods for $y'' = f(x, y)$*, Appl. Math. Lett. 22 (2009), 1076-1080.

[14] R. D'Ambrosio, M. Ferro, B. Paternoster, *Trigonometrically fitted two-step hybrid methods for special second order ordinary differential equations*, Mathematics and Computers in Simulation, 81, 1068–1084, 2011

[15] R. D'Ambrosio, L. Gr. Ixaru, B. Paternoster, *Construction of the ef-based Runge- Kutta methods revisited*, Comput. Phys. Commun. 182, 322-329, 2011.

[16] P.J. Davies, D.B. Duncan, *Stability and convergence of collocation schemes for retarded potential integral equations*, SIAM J. Numer. Anal. 42, 1167-1188, 2004.

[17] P. J., Davis, P. Rabinowitz, *Methods of numerical integration*, Computer Science and Applied Mathematics. Academic Press [A subsidiary of Harcourt Brace Jovanovich, Publishers] New York-London, 1975.

[18] R. Frontczak and R. Schoebel, *On modified Mellin transforms, Gauss-Laguerre quadrature, and the valuation of American call options*, J. Comput. Appl. Math. 234, 1559–1571, 2010.

[19] A.I. Hascelik, An asymptotic Filon-type method for infinite range highly oscillatory integrals with exponential kernel, Appl. Numer. Math. 63(2013) , 1-13.

[20] S. M. Hasheminejad and M. Aghabeigi, *Liquid sloshing in half-full horizontal elliptical tanks*,J. of Sound and Vibration 324, 332–349, 2009.

[21] D. Huybrechs, S. Vandewalle, *On the evaluation of highly oscillatory integrals by analytic continuation*, SIAM J. Numer. Anal. 44 (2006), no. 3, 1026-1048.

[22] D. Huybrechs, S. Vandewalle, *A sparse discretization for integral equation formulations of high frequency scattering problems*, SIAM J. Sci. Comput. 29, 2305-2328, 2007.

[23] L. Gr. Ixaru, *Numerical methods for differential equations and applications*, Reidel, Dordrecht-Boston-Lancaster, 1984.

[24] L. Gr. Ixaru, *Operations on Oscillatory Functions*, Comput. Phys. Commun. 105, 1–19, 1997.

[25] L. Gr. Ixaru, *Runge-Kutta method with equation dependent coefficients*, Comput. Phys. Commun. 183, 63–69, 2012.

[26] L. Gr. Ixaru, B. Paternoster, *A Gauss quadrature rule for oscillatory integrands*, Comput. Phys. Commun. 133, 177–188, 2001.

[27] L. Gr. Ixaru and G. Vanden Berghe, *Exponential Fitting*, Kluwer, Boston - Dordrecht - London, 2004.

[28] H. Kang and S. Xiang *On the calculation of highly oscillatory integrals with an algebraic singularity*, Appl. Math. Comput. 217 (2010), 3890-3897.

[29] K. J. Kim *Quadrature rules for the integration of the product of two oscillatory functions with different frequencies*, Comput. Phys. Commun. 153, 135–144, 2003.

[30] K. J Kim, *Two-frequency-dependent Gauss quadrature rules*, J. Comput. Appl. Math. 174, 43–55, 2005.

[31] K. J Kim, R. Cools, L. Gr. Ixaru, *Quadrature rules using first derivatives for oscillatory integrands*, J. Comput. Appl. Math. 140, 479–497, 2002.

[32] K. J Kim, R. Cools, L. Gr. Ixaru, *Extended quadrature rules for oscillatory integrands*, Appl. Numer. Math. 46, 59–73, 2003.

[33] H. Majidian, Numerical approximation of highly oscillatory integrals on semi-finite intervals by steepest descent method, to appear in Numer. Algor. DOI 10.1007/s11075-012-9639-x.

[34] G.V. Milovanovic, A.S. Cvetkovic, M.P. Stanic,*Gaussian quadratures for oscillatory integrands*, Appl. Math. Lett. 20, 853-860, 2007.

[35] T. E. Simos, *An exponentially-fitted Runge-Kutta method for the numerical integration of initial-value problems with periodic or oscillating solutions*. Comput. Phys. Comm. 115, 1–8, 1998.

[36] R.M. Slevinsky and H. Safouhi, A comparative study of numerical steepest descent, extrapolation, and sequence transformation methods in computing semi-infinite integrals, Numer. Algor. 60 (2012), 315-337.

[37] W. Sun, N.G. Zamani, *Adaptive mesh redistribution for the boundary element method in elastostatics*, Comput. Struct. 36, 1081-1088, 1990.

[38] M. Van Daele, G. Vanden Berghe, H. Vande Vyver, *Exponentially fitted quadrature rules of Gauss type for oscillatory integrands.* Appl. Numer. Math. 53, 509–526, 2005.

[39] M. Van Daele, T. Van Hecke, G. Vanden Berghe, H. De Meyer, *Deferred correction with mono-implicit Runge-Kutta methods for first-order IVPs. Numerical methods for differential equations* (Coimbra, 1998). J. Comput. Appl. Math. 111 (1999), no. 1-2, 3747.