Hindawi Publishing Corporation
Abstract and Applied Analysis
Volume 2014, Article ID 436164, 10 pages
http://dx.doi.org/10.1155/2014/436164

*Research Article*

# On Fast and Stable Implementation of Clenshaw-Curtis and Fejér-Type Quadrature Rules

**Shuhuang Xiang,[1] Guo He,[1] and Haiyong Wang[2]**

[1] *School of Mathematics and Statistics, Central South University, Changsha, Hunan 410083, China*
[2] *School of Mathematics and Statistics, Huazhong University of Science and Technology, Wuhan, Hubei 430074, China*

Correspondence should be addressed to Guo He; beibingyang009@163.com

Received 27 June 2014; Accepted 5 August 2014; Published 27 August 2014

Academic Editor: Robert A. Van Gorder



Based upon the fast computation of the coefficients of the interpolation polynomials at Chebyshev-type points by FFT, together with the efficient evaluation of the modified moments by forward recursions or by Oliver's algorithms, this paper presents fast and stable interpolating integration algorithms, by using the coefficients and modified moments, for Clenshaw-Curtis, Fejér's first- and second-type rules for Jacobi weights or Jacobi weights multiplied by a logarithmic function. Numerical examples illustrate the stability, efficiency, and accuracy of these quadratures.

## 1. Introduction

The interpolation quadrature of the Clenshaw-Curtis rule as well as Fejér-type formulas for

$$I[f] = \int_{-1}^{1} f(x)\,w(x)\,dx \approx I_N[f] = \sum_{k=0}^{N} w_k f(x_k) \quad (1)$$

has been extensively studied since Fejér [1, 2] in 1933 and Clenshaw and Curtis [3] in 1960, where the nodes $\{x_k\}$ are of Chebyshev types while the weights $\{w_k\}$ are computed by sums of trigonometric functions. When $x_k = \cos((2k+1)\pi/(2N+2))$ $(k = 0, 1, \dots, N)$, this quadrature is called *Fejér's first-type rule*. This kind of points is called the first kind of Chebyshev points, while *Fejér's second-type rule* is corresponding to the Filippi points $x_k = \cos((k+1)\pi/(N+2))$ $(k = 0, 1, \dots, N)$ and the *Clenshaw-Curtis-type quadrature* to the Clenshaw-Curtis points (the second kind of Chebyshev points) $x_k = \cos(k\pi/N)$ $(k = 0, 1, \dots, N)$. For more details, see Davis and Robinowitz [4], Sloan and Smith [5, 6], Sommariva [7], Trefethen [8], Waldvogel [9], and so forth.

In the case $w(x) \equiv 1$, a connection between the Fejér, Clenshaw-Curtis quadrature rules, and discrete Fourier transforms (DFTs) was given by Gentleman [10, 11], where the Clenshaw-Curtis rule is implemented with $(N+1)$ nodes by means of a discrete cosine transformation (DCT). An independent approach along the same lines, unified algorithms based on DFTs of order $N$ for generating the weights of the two Fejér rules and of the Clenshaw-Curtis rule, was presented in Waldvogel [9]. A streamlined MATLAB code is given as well in [9]. In addition, Clenshaw and Curtis [3], O'Hara and Smith [12], Trefethen [8, 13], Xiang and Bornemann in [14], Xiang et al. [15–17], and so forth showed that the Gauss, Clenshaw-Curtis, and Fejér quadrature rules are about equally accurate.

In this paper, we focus the attention on the weight functions $w(x) = (1-x)^{\alpha}(1+x)^{\beta}$ and $w(x) = \ln((1+x)/2)(1-x)^{\alpha}(1+x)^{\beta}$. For these two weight functions, the Clenshaw-Curtis-type quadrature has been extensively studied in a series of papers of Piessens [18, 19] and Piessens and Branders [20–23], by using Chebyshev interpolant $Q_N[f](x) = \sum_{n=0}^{N} a_n T_n(x)$ of $f(x)$ at the $(N+1)$ Clenshaw-Curtis points together with the modified moments $M_n = \int_{-1}^{1} w(x) T_n(x)\,dx$ [24]:

$$I[f] = \int_{-1}^{1} f(x)\,w(x)\,dx \approx I_N^{\text{C-C}}[f] = \sum_{n=0}^{N} a_n \int_{-1}^{1} T_n(x)\,w(x)\,dx$$

TABLE 1: Computation of $M_n(\alpha,\beta)=\int_{-1}^{1}(1-x)^{\alpha}(1+x)^{\beta}T_n(x)dx$ with different $n$ and $(\alpha,\beta)$ by the forward recursion of (4).

| $n$ | 5 | 10 | 100 |
|---|---|---|---|
| Exact value for (20, −0.5) | −1.734810854604316$e$ + 05 | 4.049003666168904$e$ + 03 | −3.083991348593134$e$ − 41 |
| (4) for (20, −0.5) | −1.734810854604308$e$ + 05 | 4.049003666169083$e$ + 03 | 1.787242305340324$e$ − 11 |
| Exact value for (100, −0.5) | −2.471295049468578$e$ + 29 | 1.174275526131223$e$ + 29 | 2.805165440968788$e$ − 29 |
| (4) for (100, −0.5) | −2.471295049468764$e$ + 29 | 1.174275526131312$e$ + 29 | −1.380038973213404$e$ + 13 |

$$= \sum_{k=0}^{N} a_n M_n, \tag{2}$$

where $T_n(x)$ is the Chebyshev polynomial of degree $n$ and $a_n$ can be efficiently computed by FFT [8, 10, 11] which is widely used for the approximation of highly oscillatory integrals such as [22, 25–30]. The modified moments $M_n=\int_{-1}^{1}w(x)T_n(x)dx$ satisfy the following recurrence formulas for Jacobi weights or Jacobi weights multiplied by $\ln((x+1)/2)$ [20].

(i) For $w(x)=(1-x)^{\alpha}(1+x)^{\beta}$, by using Fasenmyer's technique, the recurrence formula for the evaluation of the modified moments

$$Q_n(\alpha,\beta)=\int_{-1}^{1}(1-x)^{\alpha}(1+x)^{\beta}T_n(x)\,dx \tag{3}$$

is

$$\begin{aligned}(\beta+\alpha+k+2)\,Q_{n+1}(\alpha,\beta)+2(\alpha-\beta)\,Q_n(\alpha,\beta)\\ +(\beta+\alpha-n+2)\,Q_{n-1}(\alpha,\beta)=0\end{aligned} \tag{4}$$

with

$$\begin{aligned}Q_0(\alpha,\beta)&=2^{\beta+\alpha+1}\frac{\Gamma(\alpha+1)\Gamma(\beta+1)}{\Gamma(\beta+\alpha+2)},\\ Q_1(\alpha,\beta)&=2^{\beta+\alpha+1}\frac{\Gamma(\alpha+1)\Gamma(\beta+1)}{\Gamma(\beta+\alpha+2)}\frac{\beta-\alpha}{\beta+\alpha+2}.\end{aligned} \tag{5}$$

The forward recursion is numerically stable [20], except in two cases:

$$\alpha>\beta,\quad \beta=-\frac{1}{2},\frac{1}{2},\frac{3}{2},\ldots \tag{6}$$

$$\beta>\alpha,\quad \alpha=-\frac{1}{2},\frac{1}{2},\frac{3}{2},\ldots. \tag{7}$$

(ii) For $w(x)=\ln((x+1)/2)(1-x)^{\alpha}(1+x)^{\beta}$, for

$$G_n(\alpha,\beta)=\int_{-1}^{1}\ln\left(\frac{x+1}{2}\right)(1-x)^{\alpha}(1+x)^{\beta}T_n(x)\,dx, \tag{8}$$

the recurrence formula [20] is

$$\begin{aligned}(\beta+\alpha+k+2)\,G_{n+1}(\alpha,\beta)+2(\alpha-\beta)\,G_n(\alpha,\beta)\\ +(\beta+\alpha-n+2)\,G_{n-1}(\alpha,\beta)\\ =2Q_n(\alpha,\beta)-Q_{n-1}(\alpha,\beta)-Q_{n+1}(\alpha,\beta)\end{aligned} \tag{9}$$

with

$$\begin{aligned}G_0(\alpha,\beta)&=-\,2^{\beta+\alpha+1}\Phi(\alpha,\beta+1),\\ G_1(\alpha,\beta)&=-\,2^{\beta+\alpha+1}\left[2\Phi(\alpha,\beta+2)-\Phi(\alpha,\beta+1)\right],\end{aligned} \tag{10}$$

where

$$\Phi(\alpha,\beta)=B(\alpha+1,\beta)\left[\Psi(\alpha+\beta+1)-\Psi(\beta)\right]. \tag{11}$$

$B(x,y)$ is the Beta function and $\Psi(x)$ is the Psi function (see Abramowitz and Stegun [31]). The forward recursion is as numerically stable as (4) except for (6) or (7) [20].

Thus, the modified moments can be fast computed by the forward recursions (4) and (9) except the two cases (6) and (7), and the total costs for $I_N[f]$ are $O(N\log N)$ operations.

However, in case (6) or (7), the accuracy of the forward recursion is catastrophic particularly when $\alpha-\beta\gg 1$ or $\beta-\alpha\gg 1$ and $n\gg 1$ (see Table 1). In case (6) the relative errors $\epsilon_n$ of the computed values $Q_n(\alpha,\beta)$ obtained by the forward recursion behave approximately as

$$\epsilon_n\sim n^{2(\alpha-\beta)},\quad n\longrightarrow\infty \tag{12}$$

and in case (7) as

$$\epsilon_n\sim n^{2(\beta-\alpha)},\quad n\longrightarrow\infty. \tag{13}$$

In this paper, we will consider interpolation approaches for Clenshaw-Curtis rules as well as Fejér's first- and second-type formulas for

$$\begin{aligned}I[f]&=\int_{-1}^{1}f(x)\,w(x)\,dx\approx\sum_{k=0}^{N}a_n\int_{-1}^{1}T_n(x)\,w(x)\,dx\\ &=\sum_{k=0}^{N}a_nM_n:=I_N[f]\end{aligned} \tag{14}$$

with $w(x)=(1-x)^{\alpha}(1+x)^{\beta}$ or $w(x)=(1-x)^{\alpha}(1+x)^{\beta}\ln((1+x)/2)$, which can be efficiently calculated in $O(N\log N)$ operations. Computing the modified moments $M_n$ in cases (6) and (7) by Oliver's algorithm [32] or Lozier's algorithm [33] with one starting value and one end value, as well as offering the very short codes for the evaluation of the coefficients by FFT, is the topic of this paper.

This paper is organized as follows. In Section 2.1, we studied the asymptotic expansions of the modified moments $M_n$. Based on the results of the asymptotic expansions, we

Table 2: Computation of $Q_n(\alpha,\beta) = \int_{-1}^{1}(1-x)^{\alpha}(1+x)^{\beta}T_n(x)dx$ with $(\alpha,\beta) = (100, -0.5)$ and different $n$ by Oliver's algorithm.

| $n$ | 100 | 500 | 1000 |
|---|---|---|---|
| Exact value for (100, −0.5) | 2.805165440968788$e$ − 29 | −2.283851909785347$e$ − 198 | −1.247890461118514$e$ − 259 |
| Oliver's method for (100, −0.5) | 2.805165440968861$e$ − 29 | −2.283851909785405$e$ − 198 | −1.247890461118544$e$ − 259 |

Table 3: Computation of $G_n(\alpha,\beta) = \int_{-1}^{1}(1-x)^{\alpha}(1+x)^{\beta}\ln((1+x)/2)T_n(x)dx$ with $(\alpha,\beta) = (-0.5, 100)$ and different $n$ by Oliver's algorithm compared with that computed by the forward recursion (9).

| $n$ | 100 | 500 | 1000 |
|---|---|---|---|
| Exact value for (−0.5, 100) | 1.089944378602585$e$ − 28 | 7.222157005510106$e$ − 198 | 5.715301877322031$e$ − 259 |
| Oliver's method for (−0.5, 100) | 1.089944378602206$e$ − 28 | 7.222157005510654$e$ − 198 | 5.715301877322483$e$ − 259 |
| (9) for (−0.5, 100) | −5.331299059334499$e$ + 14 | −1.061058894110758$e$ + 14 | −5.304494050667818$e$ + 13 |

discussed Oliver's algorithm for the modified moments $M_n$ when the recursion is unstable in Section 2.2. Moreover, we gave some test to verify the accuracy and the CPU time of Oliver's algorithm. In Section 2.3, the concise MATLAB codes for evaluation of the coefficients $a_n$ by FFT are presented. The efficiency and accuracy of the three quadratures are illustrated in Section 3.

## 2. Computation of the Modified Moments and the Coefficients of the Interpolation Polynomials

From (4) and (9), we see that if $Q_n(\alpha,\beta)$ can be efficiently and stably computed then $G_n(\alpha,\beta)$ can be so. Note that the asymptotic behaviour of two linearly independent solutions of (4) satisfies

$$y_{n,1} \sim n^{-2\alpha-2}\left[1+O\left(n^{-2}\right)\right], \quad y_{n,2} \sim n^{-2\beta-2}\left[1+O\left(n^{-2}\right)\right], \quad n \longrightarrow \infty \tag{15}$$

(see Denef and Piessens [34] and Piessens and Branders [20]). Consequently, the forward recursions for (4) and (9) are perfectly stable except cases (6) and (7). In these two cases, both the forward recursions and backward recursions for (4) and (9) are numerically unstable. In the following, we will consider Oliver's algorithms to evaluate modified moments based on their asymptotic formulas of the moments with one starting value and one end value for these two cases.

### *2.1. Asymptotic Expansions of the Modified Moments*

**Lemma 1** (Erdélyi [35]). *If $0 < \lambda, \mu \le 1$, and $\phi(t)$ is $m$ times continuously differentiable for $\alpha \le t \le \beta$, then*

$$\int_{\alpha}^{\beta} e^{ixt}(t-\alpha)^{\lambda-1}(\beta-t)^{\mu-1}\phi(t)\,dt = B_m(x) - A_m(x) + O\left(x^{-m}\right), \tag{16}$$

*where*

$$A_m(x) = \sum_{n=0}^{m-1}\frac{\Gamma(n+\lambda)}{n!}e^{i\pi(n+\lambda-2)/2}x^{-n-\lambda} \times e^{ix\alpha}\left[\frac{d^n}{dt^n}\left\{(\beta-t)^{\mu-1}\phi(t)\right\}\right]_{t=\alpha},$$
$$B_m(x) = \sum_{n=0}^{m-1}\frac{\Gamma(n+\mu)}{n!}e^{i\pi(n-\mu)/2}x^{-n-\mu} \times e^{ix\beta}\left[\frac{d^n}{dt^n}\left\{(t-\alpha)^{\lambda-1}\phi(t)\right\}\right]_{t=\beta}. \tag{17}$$

**Lemma 2.** *If $0 < \lambda$, $0 < \mu$, and $\phi(t)$ is $m$ times continuously differentiable for $\alpha \le t \le \beta$, then*

$$\int_{\alpha}^{\beta} e^{ixt}(t-\alpha)^{\lambda-1}(\beta-t)^{\mu-1}\ln(\beta-t)\,\phi(t)\,dt = B_m(x) - A_m(x) + O\left(x^{-m}\right), \tag{18}$$

*where*

$$A_m(x) = \sum_{n=0}^{m-1}\frac{\Gamma(n+\lambda)}{n!}e^{i\pi(n+\lambda-2)/2}x^{-n-\lambda} \times e^{ix\alpha}\left[\frac{d^n}{dt^n}\left\{(\beta-t)^{\mu-1}\ln(\beta-t)\,\phi(t)\right\}\right]_{t=\alpha},$$
$$B_m(x) = \sum_{n=0}^{m-1}\frac{\Gamma(n+\mu)}{n!}e^{i\pi(n-\mu)/2} \times x^{-n-\mu}\left\{\Psi(n+\mu) - \ln(x) - i\frac{\pi}{2}\right\} \times e^{ix\beta}\left[\frac{d^n}{dt^n}\left\{(t-\alpha)^{\lambda-1}\phi(t)\right\}\right]_{t=\beta}. \tag{19}$$

*Proof.* The proof can be directly derived from the proof of Lemma 1 and the proof of the Theorem 5 in Erdélyi [36]. □

Table 4: Computation of $Q_n(\alpha,\beta)=\int_{-1}^{1}(1-x)^{\alpha}(1+x)^{\beta}T_n(x)dx$ with different $n$ and $(\alpha,\beta)$ by Oliver's algorithm.

| $n$ | 2000 | 4000 | 8000 |
|---|---|---|---|
| Exact value for (0.6, −0.5) | 9.551684021848334$e$ − 12 | 1.039402748103725$e$ − 12 | 1.131065744497495$e$ − 13 |
| Oliver's method for (0.6, −0.5) | 9.551684021848822$e$ − 12 | 1.039402748103918$e$ − 12 | 1.131065744497332$e$ − 13 |
| Exact value for (10, −0.5) | −8.412345942129556$e$ − 57 | −2.005493070382270$e$ − 63 | −4.781368848995069$e$ − 70 |
| Oliver's method for (10, −0.5) | −8.412345942129623$e$ − 57 | −2.005493070382302$e$ − 63 | −4.781368848995179$e$ − 70 |

**Theorem 3.** *If* $-1<\alpha, -1<\beta$, *then*

$$\begin{aligned}Q_n(\alpha,\beta) &= 2^{\beta-\alpha}\sum_{k=0}^{m-1}a_k(\alpha,\beta)\,h(\alpha+k)\\ &\quad+(-1)^n 2^{\alpha-\beta}\sum_{k=0}^{m-1}a_k(\beta,\alpha)\,h(\beta+k)\\ &\quad+O\left(n^{-2\min(\alpha,\beta)-2m}\right),\end{aligned} \tag{20}$$

*where*

$$\begin{aligned}&h(\alpha)=\cos[\pi(\alpha+1)]\,\Gamma(2\alpha+2)\,n^{-2\alpha-2},\\ &a_0(\alpha,\beta)=1,\qquad a_1(\alpha,\beta)=-\frac{\alpha}{12}-\frac{\beta}{4}-\frac{1}{6},\\ &a_2(\alpha,\beta)=\frac{1}{120}+\frac{19\alpha}{1440}+\frac{\alpha^2}{288}+\frac{\alpha\beta}{48}+\frac{\beta}{32}+\frac{\beta^2}{32},\\ &a_3(\alpha,\beta)=-\frac{1}{5040}-\frac{\beta}{960}-\frac{107\alpha}{181440}-\frac{\beta^2}{384}-\frac{\alpha^2}{1920}\\ &\qquad-\frac{\beta^3}{384}-\frac{\alpha^3}{10368}-\frac{7\alpha\beta}{2880}-\frac{\alpha^2\beta}{1152}-\frac{\alpha\beta^2}{384}.\end{aligned} \tag{21}$$

*Proof.* For $-1<\alpha,\beta\le -1/2$, taking $x=\cos(\theta)$ in (3), we have

$$\begin{aligned}Q_n(\alpha,\beta) &= \int_0^{\pi}(1-\cos(\theta))^{\alpha+1/2}\\ &\qquad\times(1+\cos(\theta))^{\beta+1/2}\cos(n\theta)\,d\theta\\ &=\int_0^{\pi}\varphi(\theta)\,\theta^{2\alpha+1}(\pi-\theta)^{2\beta+1}\cos(n\theta)\,d\theta,\end{aligned} \tag{22}$$

where

$$\varphi(\theta)=\left(\frac{1-\cos(\theta)}{\theta^2}\right)^{\alpha+1/2}\left(\frac{1+\cos(\theta)}{(\pi-\theta)^2}\right)^{\beta+1/2}. \tag{23}$$

Consequently, the desired result can be derived by applying (16) to (22).

The above outcome can be extended to the case of $-1/2<\alpha$, $-1/2<\beta$, since $Q_n(\alpha,\beta)$ can be written as

$$\begin{aligned}Q_n(\alpha,\beta) &= \int_0^{\pi}\left[\varphi(\theta)\,\theta^{\lfloor 2\alpha+1\rfloor+1}(\pi-\theta)^{\lfloor 2\beta+1\rfloor+1}\right]\\ &\qquad\times\theta^{2\alpha+1-\lfloor 2\alpha+1\rfloor-1}(\pi-\theta)^{2\beta+1-\lfloor 2\beta+1\rfloor-1}\\ &\qquad\times\cos(n\theta)\,d\theta,\end{aligned} \tag{24}$$

where $\lfloor z\rfloor$ denotes the largest integer less than or equal to $z$. □

**Theorem 4.** *If* $-1<\alpha, -1<\beta$, *then*

$$\begin{aligned}G_n(\alpha,\beta) &= 2^{\beta-\alpha}\sum_{k=0}^{m-1}c_k h(\alpha+k)+(-1)^n 2^{\alpha-\beta}\\ &\quad\times\sum_{k=0}^{m-1}h(\beta+k)\left(2a_k(\beta,\alpha)\,\phi(\beta+k)+b_k\right)\\ &\quad+O\left(n^{-2m}\right),\end{aligned} \tag{25}$$

*where*

$$\begin{aligned}&\phi(\beta)=\Psi(2\beta+2)-\ln(2n)-\frac{\pi}{2}\tan(\pi\beta),\\ &b_0=0,\qquad b_1=-\frac{1}{12},\qquad b_2=\frac{19}{1440}+\frac{\alpha}{48}+\frac{\beta}{144},\\ &b_3=-\frac{7\alpha}{2880}-\frac{\beta}{960}-\frac{\alpha^2}{384}-\frac{\beta^2}{3456}-\frac{107}{181440}-\frac{\alpha\beta}{576},\\ &c_0=0,\qquad c_1=-\frac{1}{4},\qquad c_2=\frac{1}{32}+\frac{\alpha}{48}+\frac{\beta}{16},\\ &c_3=-\frac{7\alpha}{2880}-\frac{\beta}{192}-\frac{\alpha^2}{1152}-\frac{\beta^2}{128}-\frac{1}{960}-\frac{\alpha\beta}{192}.\end{aligned} \tag{26}$$

*Proof.* Letting $x=\cos(\theta)$ in (8) and using $\ln((1+\cos(\theta))/2)=\ln((1+\cos(\theta))/2(\pi-\theta)^2)+2\ln(\pi-\theta)$, we have

$$\begin{aligned}G_n(\alpha,\beta) &= \int_0^{\pi}\ln\left(\frac{1+\cos(\theta)}{2(\pi-\theta)^2}\right)\\ &\qquad\times\varphi(\theta)\,\theta^{2\alpha+1}(\pi-\theta)^{2\beta+1}\cos(n\theta)\,d\theta\\ &\quad+2\int_0^{\pi}\varphi(\theta)\ln(\pi-\theta)\,\theta^{2\alpha+1}\\ &\qquad\times(\pi-\theta)^{2\beta+1}\cos(n\theta)\,d\theta.\end{aligned} \tag{27}$$

Applying Lemmas 1 and 2 to the two integrals on the right hand side of (27), respectively, leads to the desired result. □

*Remark 5.* Piessens and Branders gave the first term of asymptotic expansion for $Q_n(\alpha,\beta)$ and $G_n(\alpha,\beta)$ in [20] using the asymptotic theory of Fourier coefficients (Lighthill [37]). Furthermore, Piessens [18] presented the asymptotic expansion with explicit formulas for the first three terms for

Table 5: Computation of $G_n(\alpha,\beta)=\int_{-1}^{1}(1-x)^{\alpha}(1+x)^{\beta}\ln((1+x)/2)T_n(x)dx$ with different $n$ and $(\alpha,\beta)$ by Oliver's algorithm.

| $n$ | 10 | 100 | 500 |
|---|---|---|---|
| Exact value for (−0.4999, −0.5) | −0.314181354550401 | −0.031418104511487 | −0.006283620842004 |
| Oliver's method for (−0.4999, −0.5) | −0.314181354550428 | −0.031418104511490 | −0.006283620842004 |
| Exact value for (0.9999, −0.5) | −0.895286620533541 | −0.088858164406923 | −0.017770353274330 |
| Oliver's method for (0.9999, −0.5) | −0.895286620533558 | −0.088858164406925 | −0.017770353274330 |

Table 6: The CPU time for calculation of the first $(N+1)$ modified moments by the Oliver's methods for $\alpha=-0.5$ and $\beta=100$.

| Modified moments | $N=10^3$ | $N=10^4$ | $N=10^5$ | $N=10^6$ |
|---|---|---|---|---|
| $\{Q_n(\alpha,\beta)\}_{n=0}^{N}$ | 0.003954 s | 0.011553 s | 0.121553 s | 1.200070 s |
| $\{G_n(\alpha,\beta)\}_{n=0}^{N}$ | 0.006087 s | 0.028416 s | 0.296032 s | 2.933928 s |

```
function a = clenshaw_curtis(f,N)              % (N+1)-coefficients for C-C quadrature
x = cos(pi*(0:N)'/N);                          % C-C points
fx = feval(f,x)/(2*N);                         % f evaluated at these points
g = fft(fx([1:N+1 N:-1:2]));                   % FFT
a = [g(1); g(2:N)+g(2*N:-1:N+2); g(N+1)];      % Chebyshev coefficients
```

Algorithm 1

$Q_n(\alpha,\beta)$, which is equivalent to (20) except a minor clerical error. In [18], $a_1(\alpha,\beta)=-((\alpha+2\beta+2)/3)2^{\beta-\alpha-2}$ should be modified by $a_1(\alpha,\beta)=-((\alpha+3\beta+2)/3)2^{\beta-\alpha-2}$. The first term of (25) is different from the first asymptotic term for $G_n(\alpha,\beta)$ proposed in [18]; that is, $2^{\beta-\alpha}\sum_{k=0}^{1}c_k h(\alpha+k)+(-1)^n 2^{\alpha-\beta}\sum_{k=0}^{0}h(\beta+k)[2a_k(\beta,\alpha)\phi(\beta+k)+b_k]$. In order to ensure the accuracy of the modified moments evaluated by Oliver's method, the end value of modified moments must be computed very precisely. In this paper, we use the first four-term truncation.

*2.2. Oliver's Algorithm.* Let

$$A_N:=\begin{pmatrix} 2(\alpha-\beta) & \alpha+\beta+2+1 & & & \\ \alpha+\beta+2-2 & 2(\alpha-\beta) & \alpha+\beta+2+2 & & \\ & \ddots & \ddots & \ddots & \\ & & \alpha+\beta+2-(N-1) & 2(\alpha-\beta) & \alpha+\beta+2+(N-1) \\ & & & \alpha+\beta+2-N & 2(\alpha-\beta)\end{pmatrix}, \quad (28)$$

$$b_N:=\left(-(\beta+\alpha+1)Q_0(\alpha,\beta)\ \ 0\ \cdots\ 0\ \ -(\alpha+\beta+2+N)Q_{N+1}\right)^T, \quad (29)$$

where "$\cdot^T$" denotes the transpose; then the modified moments can be solved by

$$A_N M=b_N,\qquad M=\left(Q_1(\alpha,\beta),Q_2(\alpha,\beta),\dots,Q_N(\alpha,\beta)\right)^T, \quad (30)$$

where $Q_{N+1}(\alpha,\beta)$ is computed by hypergeometric function [20] when $N\le 2000$:

$$Q_{N+1}(\alpha,\beta)=2^{\alpha+\beta+1}\frac{\Gamma(\alpha+1)\Gamma(\beta+1)}{\Gamma(\alpha+\beta+2)}\times{}_3F_2\left([N+1,-N-1,\alpha+1],\left[\frac{1}{2},\alpha+\beta+2\right],1\right). \quad (31)$$

```
function a = fejer1(f,N)
x = cos(pi*(2*(0:N)'+1)/(2*N+2));                % coefficients for Fejér's first rule
fx = feval(f,x)/(N+1);                           % first kind of Chebyshev points
g = fft(fx([1:N+1 N+1:-1:1]));                   % f evaluated at these points
hx = real(exp(2*1i*pi*(0:2*n+1)/(4*n+4)).*g');   % FFT
a = hx(1:n+1);a(1)=0.5*a(1);                     % Chebyshev coefficients
```

Algorithm 2

```
function a = fejer2(f,N)                          % coefficients for Fejér's second rule
x = cos(pi*(0:N+2)'/(N+2));
fx = feval(f,x)/(2N+4);                           % f evaluated at these points
g = fft(fx([1:N+3 N+2:-1:2]));                    % FFT
b = [g(1); g(2:N+2)+g(2*N+4:-1:N+4); g(N+3)];
b(N+1:-2:1) = b(N+1:-2:1)-2*b(N+3);
b(N:-2:1) = b(N:-2:1)-b(N+2);
b(1) = b(1)+mod(N+1,2)*b(N+3)+mod(N,2)*b(N+2)/2;
a = b(1:N+1);                                     % Chebyshev coefficients
```

Algorithm 3

```
function a = fejer1dct(f,N)                       % coefficients for Fejér's second rule
x = cos(pi*(2*(0:N)'+1)/(2*N+2));                 % The first kind of Chebyshev points
fx = feval(f,x);                                  % f evaluated at these points
a = dct(fx)*sqrt(2/(N+1));a(1)=a(1)/sqrt(2);      % Chebyshev coefficients
```

Algorithm 4

```
function a = fejer2idst(f,N)                      % coefficients for Fejér's second rule
x = cos(pi*(1:N+1)'/(N+2));                       % Filippi points
fx = feval(f,x).*sin(pi*(1:N+1)'/(N+2));          % f evaluated at these points
a = idst(fx);                                     % Chebyshev coefficients
```

Algorithm 5

If $N > 2000$, $Q_{N+1}(\alpha,\beta)$ is computed by using the asymptotic expression (20) with the first four-term truncation. Particularly, Oliver's algorithm can be fast implemented by applying LU factorization (chasing method) with $O(N)$ operations.

In addition, for the weight $w(x) = \ln((x+1)/2)(1-x)^{\alpha}(1+x)^{\beta}$ in case of (6) or (7), the computation of the modified moments can be fixed up by Oliver's algorithm similar to (29) with one starting $G_0(\alpha,\beta) = -2^{\alpha+\beta+1}\Phi(\alpha,\beta+1)$ and one end $G_{N+1}(\alpha,\beta)$. Consequently, the modified moments can be solved by

$$A_N G = C_N, \quad G = \left(G_1(\alpha,\beta), G_2(\alpha,\beta), \dots, G_N(\alpha,\beta)\right)^T, \tag{32}$$

where

$$\begin{aligned} C_N &= \begin{pmatrix} c_1 & c_2 & \cdots & c_{N-1} & c_N \end{pmatrix}^T, \\ c_1 &= -(\beta+\alpha+1)G_0(\alpha,\beta) + 2Q_1(\alpha,\beta) \\ &\quad - Q_0(\alpha,\beta) - Q_2(\alpha,\beta), \\ c_k &= 2Q_k(\alpha,\beta) - Q_{k-1}(\alpha,\beta) - Q_{k+1}(\alpha,\beta), \\ &\qquad k = 2,\dots,N-1, \\ c_N &= -(\beta+\alpha+N+2)G_{N+1}(\alpha,\beta) \\ &\quad + 2Q_N(\alpha,\beta) - Q_{N-1}(\alpha,\beta) - Q_{N+1}(\alpha,\beta). \end{aligned} \tag{33}$$

The end value $G_{N+1}(\alpha,\beta)$ can be calculated by its asymptotic formula (25) with the first four-term truncation.

Tables 2 and 3 show the accuracy of Oliver's algorithm for $\alpha = 100$ and $\beta = -0.5$, $\alpha = -0.5$ and $\beta = 100$, respectively. Table 4 displays the accuracy of Oliver's algorithm for $\alpha = 0.6$ and $\beta = -0.5$, $\alpha = 10$ and $\beta = -0.5$ for $Q_n(\alpha,\beta)$. Table 5 displays the accuracy of Oliver's algorithm for $\alpha = -0.4999$ and $\beta = -0.5$, $\alpha = 0.9999$ and $\beta = -0.5$ for $G_n(\alpha,\beta)$. Table 6 shows the CPU time for implementation of the two Oliver's algorithms. From these tables, we see that Oliver's algorithms have high efficiency and are precise. The MATLAB codes on Oliver's algorithms and all the MATLAB codes in this paper can be downloaded from [38].

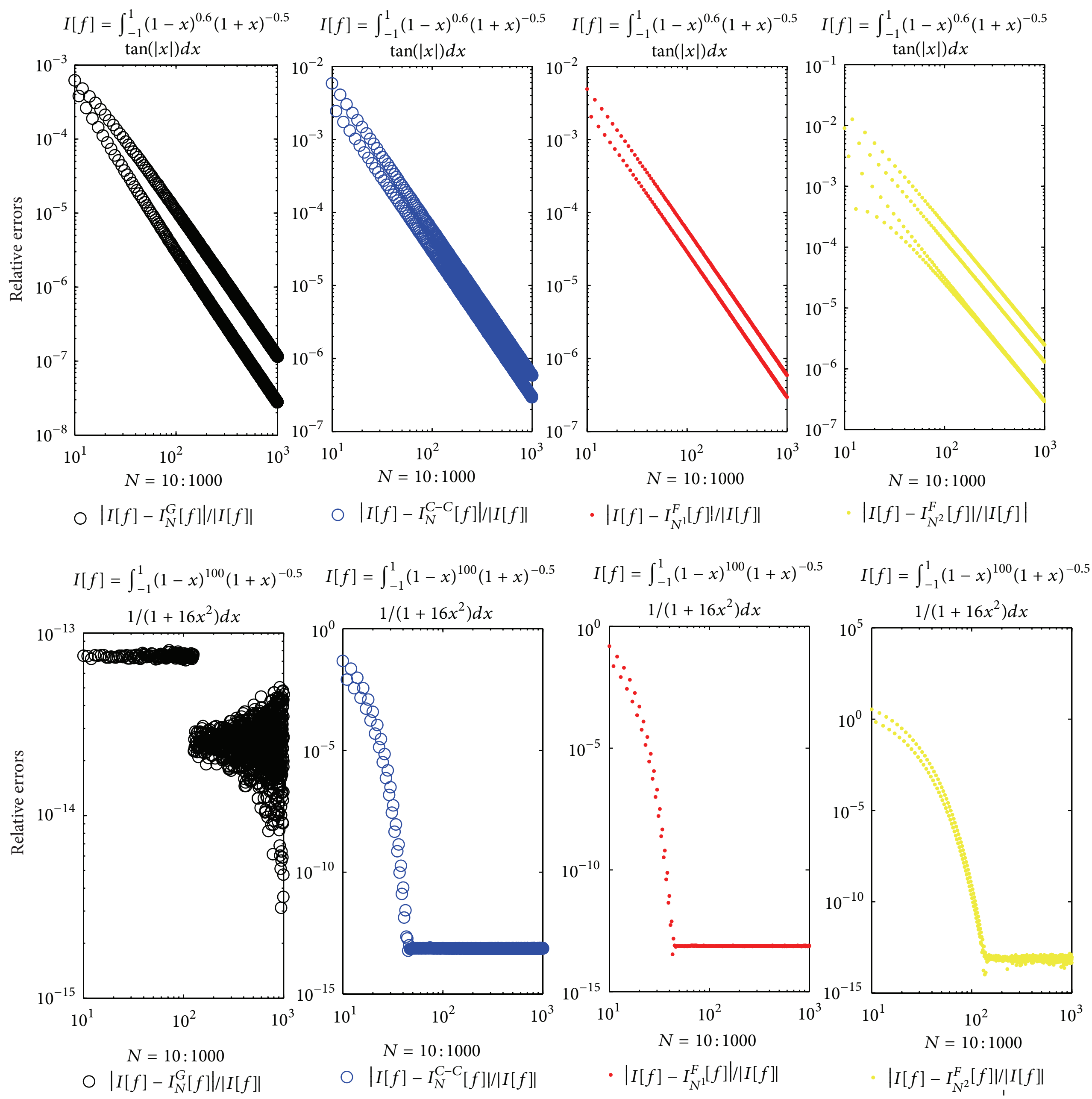


FIGURE 1: The relative errors compared with Gauss quadrature for $\int_{-1}^{1}(1-x)^{\alpha}(1+x)^{\beta}f(x)dx$ evaluated by the Clenshaw-Curtis, Fejér's first- and second-type rules with $N$ nodes, respectively: $f(x) = \tan|x|$ or $1/(1+16x^2)$ and $N = 10 : 1000$.

*2.3. Fast Computation of the Coefficients by FFT.* The coefficients $a_n$ for the interpolation polynomial $Q_N[f](x) = \sum_{n=0}^{N} a_n T_n(x)$ at the three kinds of points can be efficiently computed by FFT. For the Clenshaw-Curtis points, the elegant MATLAB code on the $a_n$ is from [8] (see Algorithm 1).

For the other two classes of points, by the sums of trigonometric functions at these two point-set, it is not difficult to get the FFT implementation. Here, we will not give details but just offer the following MATLAB functions.

For the first kind of Chebyshev points, we presented a MATLAB code for computing the coefficients $a_n$ by FFT in Algorithm 2.

For the Filippi points, we presented a MATLAB code for computing the coefficients $a_n$ by FFT in Algorithm 3.

In addition, the coefficients of the interpolation polynomials $Q_N[f](x)$ at these Chebyshev-type points may be also efficiently evaluated by DCT and inverse discrete sine transform (IDST), respectively. The discrete cosine transform denoted by $Y = \text{dct}(X)$ is closely related to the discrete Fourier transform but using purely real numbers and takes $O(N \log N)$ operations for

$$Y(k) = \sigma(k) \sum_{s=1}^{N} X(s) \cos\left(\frac{(k-1)\pi(2s-1)}{2N}\right)$$

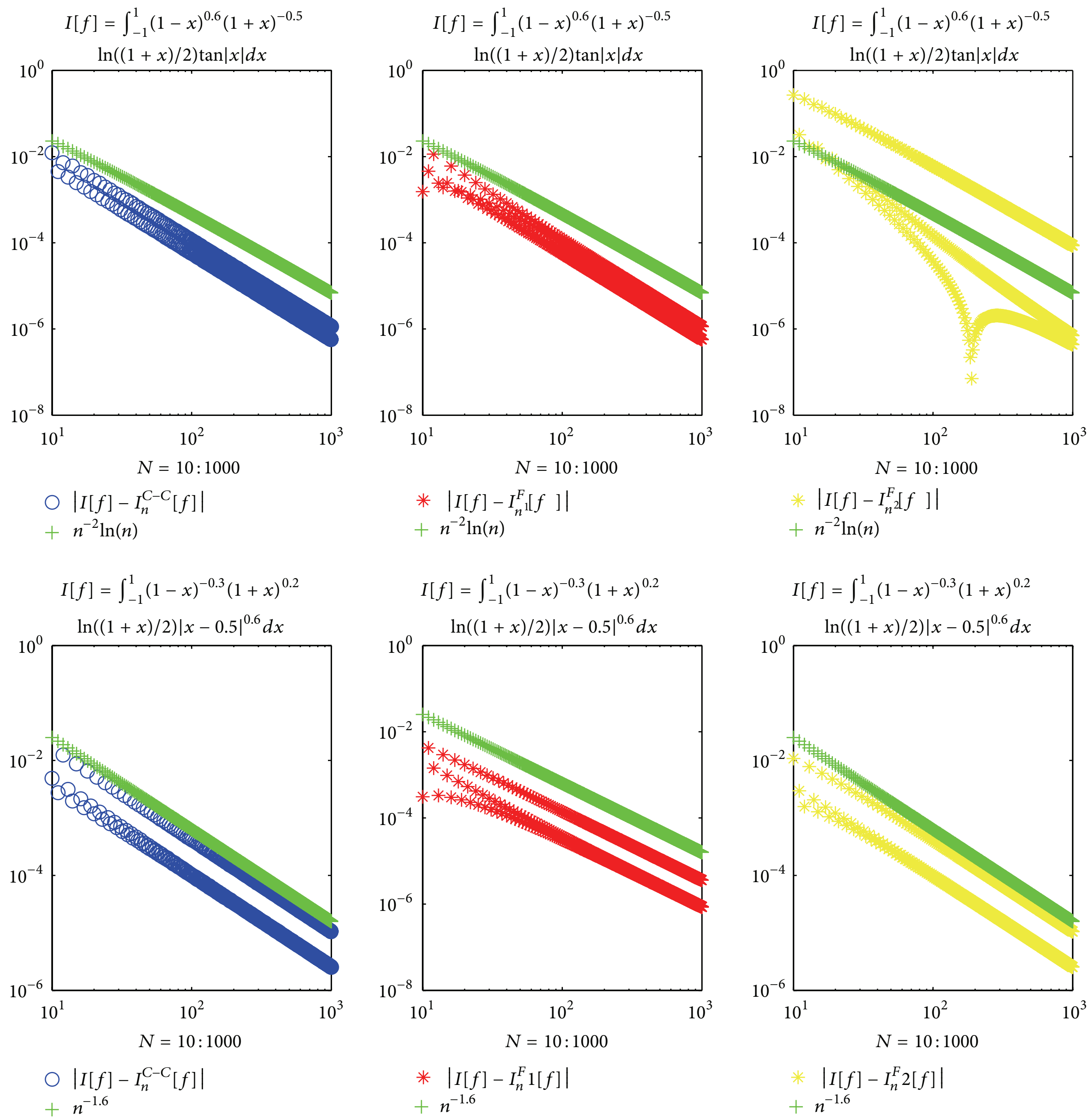


Figure 2: The absolute errors for $\int_{-1}^{1}(1-x)^{\alpha}(1+x)^{\beta}\ln((1+x)/2)f(x)dx$ evaluated by the Clenshaw-Curtis, Fejér's first- and second-type rules with $N$ nodes, respectively: $f(x)=\tan|x|$ or $|x-0.5|^{0.6}$ and $N=10\ :\ 1000$.

with
$$\sigma(1)=\frac{1}{\sqrt{N}},\quad \sigma(k)=\sqrt{\frac{2}{N}},\qquad \text{for } 2\le k\le N. \tag{34}$$

The discrete sine transform denoted by $Y=\operatorname{dst}(X)$ and its inverse by $X=\operatorname{idst}(Y)$ both take $O(N\log N)$ operations for

$$Y(k)=\sum_{s=1}^{N}X(s)\sin\left(\frac{k\pi s}{N+1}\right). \tag{35}$$

Notice that the coefficients $a_j$ for the interpolation polynomial $Q_N[f](x)=\sum_{n=1}^{N+1}{}' a_{n-1}T_{n-1}(x)$ at $\{\cos((2k-1)\pi/(2N+2))\}_{k=1}^{N+1}$ are represented by

$$a_{n-1}=\frac{2}{N+1}\sum_{s=1}^{N+1}f\left(\cos\left(\frac{(2s-1)\pi}{2N+2}\right)\right)\times\cos\left(\frac{(2s-1)(n-1)\pi}{2N+2}\right),\quad n=1,2,\ldots,N+1, \tag{36}$$

and $\widetilde{a}_n$ for the interpolation polynomial $Q_N[f](x) = \sum_{n=1}^{N+1} \widetilde{a}_{n-1} U_{n-1}(x)$ at $\{\cos(k\pi/(N+2))\}_{k=1}^{N+1}$ satisfies

$$f\left(\cos\left(\frac{k\pi}{N+2}\right)\right)\sin\left(\frac{k\pi}{N+2}\right) = \sum_{s=1}^{N+1} \widetilde{a}_{s-1} \sin\left(\frac{sk\pi}{N+2}\right), \quad k = 1, 2, \ldots, N+1. \tag{37}$$

Then both can be efficiently calculated by DCT and IDST, respectively. The MATLAB codes on the DCT and IDST are very short and just need three rows. Now, we give the codes in Algorithm 4 and Algorithm 5.

For the first kind of Chebyshev points, a MATLAB code for computing the coefficients $a_n$ by DCT is presented in Algorithm 4.

As far as the Filippi points, a MATLAB code for computing the coefficients $\widetilde{a}_n$ by IDST is presented in Algorithm 5.

*Remark 6.* For the modified moments on the second kind Chebyshev polynomial $\widetilde{M}_n := \int_{-1}^{1} w(x) U_n(x) dx$, we have

$$\widetilde{M}_0 = M_0, \qquad \widetilde{M}_1 = 2M_1,$$
$$\widetilde{M}_n = 2M_n + \widetilde{M}_{n-2}, \quad n = 2, 3, \ldots; \tag{38}$$

here we use the simple equation $U_{n+2} = 2T_{n+2} + U_n$ (see [31, pp. 778]).

## 3. Numerical Examples

In this section, we illustrate the accuracy and efficiency of the Clenshaw-Curtis, Fejér's first- and second-type rules for the functions $\tan|x|$, $1/(1+16x^2)$ and $|x-0.5|^{0.6}$ by the algorithms presented in this paper, which are compared with the Gauss-Jacobi quadrature used $[x, w] = \text{jacpts}(n, \alpha, \beta)$ in CHEBFUN v4.2 [39] (see Figure 1). The first column computed by Gauss-Jacobi quadrature in Figure 1 takes 85.7386 seconds and the others totally take 2.9211 seconds in a Lenovo computer with Intel Core 3.20 GHz and 3.47 GB RAM. Figure 2 shows the convergence errors by the three quadratures, which takes 7.336958 seconds.

## Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

## Acknowledgment

This paper is supported by the National Natural Science Foundation of China, no. 11371376.

## References

[1] L. Fejér, "On the infinite sequences arising in the theories of harmonic analysis, of interpolation, and of mechanical quadratures," *Bulletin of the American Mathematical Society*, vol. 39, no. 8, pp. 521–534, 1933.

[2] L. Fejér, "Mechanische Quadraturen mit positiven Cotesschen Zahlen," *Mathematische Zeitschrift*, vol. 37, no. 1, pp. 287–309, 1933.

[3] C. W. Clenshaw and A. R. Curtis, "A method for numerical integration on an automatic computer," *Numerische Mathematik*, vol. 2, pp. 197–205, 1960.

[4] P. J. Davis and P. Rabinowitz, *Methods of Numerical Integration*, Academic Press, New York, NY, USA, 2nd edition, 1984.

[5] I. H. Sloan and W. E. Smith, "Product-integration with the Clenshaw-CURtis and related points," *Numerische Mathematik*, vol. 30, no. 4, pp. 415–428, 1978.

[6] I. H. Sloan and W. E. Smith, "Product integration with the Clenshaw-CURtis points: implementation and error estimates," *Numerische Mathematik*, vol. 34, no. 4, pp. 387–401, 1980.

[7] A. Sommariva, "Fast construction of Fejér and Clenshaw-Curtis rules for general weight functions," *Computers & Mathematics with Applications*, vol. 65, no. 4, pp. 682–693, 2013.

[8] L. N. Trefethen, "Is Gauss quadrature better than Clenshaw-Curtis?" *SIAM Review*, vol. 50, no. 1, pp. 67–87, 2008.

[9] J. Waldvogel, "Fast construction of the Fejér and Clenshaw-Curtis quadrature rules," *BIT*, vol. 46, no. 1, pp. 195–202, 2006.

[10] W. M. Gentleman, "Implementing Clenshaw-Curtis quadrature," *Communications of the ACM*, vol. 15, no. 5, pp. 337–346, 1972.

[11] W. Morven, "Algorithm 424: Clenshaw-Curtis quadrature [D1]," *Communications of the ACM*, vol. 15, no. 5, pp. 353–355, 1972.

[12] H. O'Hara and F. J. Smith, "Error estimation in the Clenshaw-CURtis quadrature formula," *The Computer Journal*, vol. 11, pp. 213–219, 1968.

[13] L. N. Trefethen, *Approximation Theory and Approximation Practice*, SIAM, New York, NY, USA, 2013.

[14] S. Xiang and F. Bornemann, "On the convergence rates of Gauss and Clenshaw-CURtis quadrature for functions of limited regularity," *SIAM Journal on Numerical Analysis*, vol. 50, no. 5, pp. 2581–2587, 2012.

[15] S. Xiang, X. Chen, and H. Wang, "Error bounds for approximation in Chebyshev points," *Numerische Mathematik*, vol. 116, no. 3, pp. 463–491, 2010.

[16] S. Xiang, "On convergence rates of Fejér and Gauss-Chebyshev quadrature rules," *Journal of Mathematical Analysis and Applications*, vol. 405, no. 2, pp. 687–699, 2013.

[17] S. Xiang, "On the optimal rates of convergence for quadratures derived from Chebyshev points," http://arxiv.org/abs/1308.4322.

[18] R. Piessens, "Modified Clenshaw-Curtis integration and applications to numerical computation of integral transforms," in *Numerical Integration*, P. Keast and G. Fairweather, Eds., pp. 35–51, 1987.

[19] R. Piessens, "Computing integral transforms and solving integral equations using Chebyshev polynomial approximations," *Journal of Computational and Applied Mathematics*, vol. 121, no. 1-2, pp. 113–124, 2000.

[20] R. Piessens and M. Branders, "The evaluation and application of some modified moments," *BIT Numerical Mathematics*, vol. 13, no. 4, pp. 443–450, 1973.

[21] R. Piessens and M. Branders, "Modified Clenshaw-CURtis method for the computation of Bessel function integrals," *BIT Numerical Mathematics*, vol. 23, no. 3, pp. 370–381, 1983.

[22] R. Piessens and M. Branders, "Computation of Fourier transform integrals using Chebyshev series expansions," *Computing*, vol. 32, no. 2, pp. 177–186, 1984.

[23] R. Piessens and M. Branders, "On the computation of Fourier transforms of singular functions," *Journal of Computational and Applied Mathematics*, vol. 43, no. 1-2, pp. 159–169, 1992.

[24] W. Gautschi, "Computational aspects of three-term recurrence relations," *SIAM Review*, vol. 9, pp. 24–82, 1967.

[25] R. Chen and C. An, "On evaluation of Bessel transforms with oscillatory and algebraic singular integrands," *Journal of Computational and Applied Mathematics*, vol. 264, pp. 71–81, 2014.

[26] R. Chen and C. An, "On the evaluation of infinite integrals involving Bessel functions," *Applied Mathematics and Computation*, vol. 235, pp. 212–220, 2014.

[27] H. Kang and S. Xiang, "Efficient integration for a class of highly oscillatory integrals," *Applied Mathematics and Computation*, vol. 218, no. 7, pp. 3553–3564, 2011.

[28] H. Kang and S. Xiang, "Efficient quadrature of highly oscillatory integrals with algebraic singularities," *Journal of Computational and Applied Mathematics*, vol. 237, no. 1, pp. 576–588, 2013.

[29] H. Wang and S. Xiang, "Uniform approximations to Cauchy principal value integrals of oscillatory functions," *Applied Mathematics and Computation*, vol. 215, no. 5, pp. 1886–1894, 2009.

[30] S. Xiang, Y. J. Cho, H. Wang, and H. Brunner, "Clenshaw-Curtis-Filon-type methods for highly oscillatory Bessel transforms and applications," *IMA Journal of Numerical Analysis*, vol. 31, no. 4, pp. 1281–1314, 2011.

[31] M. Abramowitz and I. A. Stegun, *Handbook of Mathematical Functions*, National Bureau of Standards, Washington, DC, USA, 1964.

[32] J. Oliver, "The numerical solution of linear recurrence relations," *Numerische Mathematik*, vol. 11, pp. 349–360, 1968.

[33] D. W. Lozier, "Numerical solution of linear difference equations," Tech. Rep. NBSIR 80-1976, Math. Analysis, Nat. Bureau of Standards, 1980.

[34] J. Denef and R. Piessens, "The asymptotic behaviour of solutions of difference equations of Poincaré's type," *Bulletin de la Société Mathématique de Belgique*, vol. 26, no. 2, pp. 133–347, 1974.

[35] A. Erdélyi, "Asymptotic representations of Fourier integrals and the method of stationary phase," *Journal of the Society for Industrial and Applied Mathematics*, vol. 3, pp. 17–27, 1955.

[36] A. Erdélyi, "Asymptotic expansions of integrals involving logarithmic singularities," *Journal of the Society for Industrial and Applied Mathematics*, vol. 1, pp. 38–47, 1956.

[37] M. J. Lighthill, *Fourier Analysis and Generalized Functions*, Cambridge University Press, Cambridge, UK, 1959.

[38] http://math.csu.edu.cn/office/teacherpage.aspx?namenumber=56.

[39] L. N. Trefethen et al., "Chebfun Version 4.2," The Chebfun Development Team, 2011, http://www.maths.ox.ac.uk/chebfun/.