DESIGN OF DIGITAL IIR INTEGRATOR USING RADIAL BASIS FUNCTION INTERPOLATION METHOD

Chien-Cheng Tseng[1] *and Su-Ling Lee*[2]

[1]Depart. of Computer and Communication Engineering
National Kaohsiung First University of Sci. and Tech.
Kaohsiung, Taiwan
tcc@ccms.nkfust.edu.tw

[2]Depart. of Computer Sci. and Information Engi.
Chung-Jung Christian University
Tainan, Taiwan
lilee@mail.cjcu.edu.tw

ABSTRACT

In this paper, the design of digital IIR integrator is investigated. First, the radial basis function (RBF) interpolation method is described. Then, fractionally delayed sample estimation of discrete-time sequence is derived by using RBF interpolation approach. Next, the numerical integration rule and fractionally delayed sample estimation are applied to obtain the transfer function of digital integrator. Finally, some numerical comparisons with conventional digital integrators are made to demonstrate the effectiveness of this new design approach.

1. INTRODUCTION

Digital integrators are useful devices in the application areas of control, radar and biomedical engineering [1]-[4]. The methods of digital integrator design can generally be classified into two categories. One is the linear phase FIR filter approach in which the filter coefficients are determined by using maximal flatness constraints [1][2], the other is the IIR filter method in which the filter coefficients are obtained directly from well-known numerical integration rule [3][4]. The ideal frequency response of digital integrator is given by

$$D(\omega) = \frac{1}{j\omega} e^{-j\omega I} \tag{1}$$

where I is a prescribed integer delay. The problem is how to design a digital filter such that its frequency response fits $D(\omega)$ as well as possible. In [3], Ngo presented a third-order digital integrator whose transfer function is given by

$$F_1(z) = \frac{z^{-I}}{24} \frac{9 + 19 z^{-1} - 5 z^{-2} + z^{-3}}{1 - z^{-1}} \tag{2}$$

In [4], Tseng and Lee have used Richardson extrapolation and polyphase decomposition to design digital integrators. From Eq.(62) in [4], the transfer function of a typical forth-order integrator is given by

$$F_2(z) = \frac{z^{-I+1}}{139968} \frac{\begin{pmatrix} -3693 + 67260 z^{-1} + 88650 z^{-2} \\ -14388 z^{-3} + 2139 z^{-4} \end{pmatrix}}{1 - z^{-1}} \tag{3}$$

From Eq.(1), it is clear that the gain of integrator at zero frequency $\omega = 0$ is infinity, so the above transfer functions have one pole at $z = 1$. On the other hand, radial basis function (RBF) has been widely used in multivariate interpolation, neural network, time-series prediction, control of nonlinear systems, mesh-free approximation, and target tracking in video data [5]-[7]. The early work on the RBF theory and implementation is surveyed in the book [8]. A radial basis function $\phi(t)$ is a real-valued function whose value depends only on the distance from the origin, that is, $\phi(t) = \phi(\|t\|)$. The notation $\|\cdot\|$ denotes a norm that is usually taken as be Euclidean. Common used types of radial basis function include Gaussian and inverse multiquadric whose definitions are given by

$$\text{Gaussian:} \quad \phi(t) = e^{-\frac{t^2}{\sigma^2}} \tag{4a}$$

$$\text{Inverse multiquadric:} \quad \phi(t) = \frac{1}{\sqrt{t^2 + \sigma^2}} \tag{4b}$$

where σ is called shape parameter which can be used to adjust the shape of function $\phi(t)$. The purpose of this paper is to use RBF interpolation method to design digital IIR integrators. As a result, the design error can be reduced by suitably choosing the shape parameter of radial basis function.

2. RADIAL BASIS FUNCTION INTERPOLATION

In this section, the radial basis function interpolation method will be described. Given a set of $N+1$ different points $\{t_0, t_1, t_2, \cdots, t_N\}$ and a corresponding set of $N+1$ real numbers $\{s_0, s_1, s_2, \cdots, s_N\}$, the interpolation problem is to find a function $s(t)$ that satisfies the interpolation condition

$$s(t_k) = s_k \qquad k = 0,1,2,\cdots,N \tag{5}$$

The radial basis function interpolation method consists of choosing a function $s(t)$ that has the following form

$$s(t) = \sum_{k=0}^{N} w_k \phi(\|t - t_k\|) \tag{6}$$

That is, the function $s(t)$ is represented as a sum of $N+1$ radial basis functions, each associated with a different center t_k, and weighted by an appropriate coefficient w_k. Substituting the interpolation condition of Eq.(5) into Eq.(6), we get the following simultaneous linear equation

$$\begin{bmatrix} \phi_{00} & \phi_{01} & \phi_{02} & \cdots & \phi_{0N} \\ \phi_{10} & \phi_{11} & \phi_{12} & \cdots & \phi_{1N} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \phi_{N0} & \phi_{N1} & \phi_{N2} & \cdots & \phi_{NN} \end{bmatrix} \begin{bmatrix} w_0 \\ w_1 \\ \vdots \\ w_N \end{bmatrix} = \begin{bmatrix} s_0 \\ s_1 \\ \vdots \\ s_N \end{bmatrix} \quad (7)$$

where $\phi_{mk} = \phi(\|t_m - t_k\|)$. Let vectors S and W be

$$S = [s_0 \quad s_1 \quad \cdots \quad s_N]^T \quad (8a)$$

$$W = [w_0 \quad w_1 \quad \cdots \quad w_N]^T \quad (8b)$$

and Φ denotes an $(N+1)$-by-$(N+1)$ matrix with element ϕ_{mk}, then Eq.(7) can be rewritten as

$$\Phi W = S \quad (9)$$

If t_0, t_1, ..., t_N are distinct points, then matrix Φ is nonsingular. Thus, the unknown vector W is given by

$$W = \Phi^{-1} S \quad (10)$$

Once W has been obtained, the function $s(t)$ in Eq.(6) is known. So, $s(t)$ is computable for the given t. Finally, an example is use to illustrate the RBF interpolation method. The data s_k is obtained by uniformly sampling the continuous sinusoidal function $\cos(0.2\pi t)$ with $t_k = \frac{k}{10}$, that is, $s_k = \cos(\frac{0.2\pi k}{10})$. The radial basis function is chosen as Gaussian in Eq.(4a) with $\sigma = 0.1$. The number of points are $N + 1 = 201$. Fig.1(a)(b) shows the interpolated function $s(t)$ in Eq.(6) and sinusoidal function $\cos(0.2\pi t)$. It is clear that both functions look almost the same. To observe where the errors occur, Fig.1(c) shows the absolute errors $|s(t) - \cos(0.2\pi t)|$. Clearly, the errors are very small except at the edge points $t_0 = 0$ and $t_N = 20$.

3. FRACTIONALLY DELAYED SAMPLE ESTIMATION

In this section, we will use RBF interpolation method to solve fractionally delayed sample estimation problem because the proposed IIR integrator design method is based on this estimation method. The problem to be studied is how to estimate fractionally delayed sample $s(n-I-d)$ from the given integer delayed samples $s(n)$, $s(n-1)$, $s(n-2)$,..., $s(n-N)$, where I and N are integers and d is a fractional number in the interval $[0,1]$. And, I is usually chosen in the range $[0, N-1]$. In this paper, we use weighted average approach to achieve the purpose, that is, fractionally delayed sample is estimated by

$$s(n-I-d) = \sum_{m=0}^{N} h(m,d) s(n-m) \quad (11)$$

Now, the remaining problem is how to use the RBF interpolation method in the preceding section to determine the weights $h(m,d)$. To solve this problem, let us choose $t_k = n-k$ and $s_k = s(n-k)$. Then, the RBF interpolation formula in Eq.(6) becomes

$$\begin{aligned} s(t) &= \sum_{k=0}^{N} w_k \phi(\|t - t_k\|) \\ &= \sum_{k=0}^{N} w_k \phi(\|t - (n-k)\|) \end{aligned} \quad (12)$$

Because $t_k = n-k$ and $t_m = n-m$ are chosen, we have

$$\phi_{mk} = \phi(\|t_m - t_k\|) = \phi(\|k-m\|) \quad (13)$$

Using the above expression and $s_k = s(n-k)$, the simultaneous equation in Eq.(7) reduces to

$$\begin{bmatrix} \phi(0) & \phi(1) & \phi(2) & \cdots & \phi(N) \\ \phi(1) & \phi(0) & \phi(1) & \cdots & \phi(N-1) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \phi(N) & \phi(N-1) & \phi(N-2) & \cdots & \phi(0) \end{bmatrix} \begin{bmatrix} w_0 \\ w_1 \\ \vdots \\ w_N \end{bmatrix} = \begin{bmatrix} s(n) \\ s(n-1) \\ \vdots \\ s(n-N) \end{bmatrix} \quad (14)$$

This equation can be shorten as the form $\Phi W = S$, as described in Eq.(9). Clearly, Φ is a symmetric and Toeplitz matrix. Let the inverse of matrix Φ be denoted by

$$\Phi^{-1} = \begin{bmatrix} \alpha_{00} & \alpha_{01} & \alpha_{02} & \cdots & \alpha_{0N} \\ \alpha_{10} & \alpha_{11} & \alpha_{12} & \cdots & \alpha_{1N} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \alpha_{N0} & \alpha_{N1} & \alpha_{N2} & \cdots & \alpha_{NN} \end{bmatrix} \quad (15)$$

then the solution of simultaneous equation in Eq.(14) is given by

$$\begin{bmatrix} w_0 \\ w_1 \\ \vdots \\ w_N \end{bmatrix} = \Phi^{-1} \begin{bmatrix} s(n) \\ s(n-1) \\ \vdots \\ s(n-N) \end{bmatrix} = \begin{bmatrix} \sum_{m=0}^{N} \alpha_{0m} s(n-m) \\ \sum_{m=0}^{N} \alpha_{1m} s(n-m) \\ \vdots \\ \sum_{m=0}^{N} \alpha_{Nm} s(n-m) \end{bmatrix} \quad (16)$$

This result implies that

$$w_k = \sum_{m=0}^{N} \alpha_{km} s(n-m) \qquad k = 0,1,2,\cdots,N \quad (17)$$

Substituting Eq.(17) into Eq.(12), we get

$$\begin{aligned} s(t) &= \sum_{k=0}^{N} w_k \phi(\|t-(n-k)\|) \\ &= \sum_{k=0}^{N} \left(\sum_{m=0}^{N} \alpha_{km} s(n-m) \right) \phi(\|t-(n-k)\|) \\ &= \sum_{m=0}^{N} \left(\sum_{k=0}^{N} \alpha_{km} \phi(\|t-(n-k)\|) \right) s(n-m) \end{aligned} \quad (18)$$

Taking $t = n-I-d$, the above equation reduces to

$$s(n-I-d) = \sum_{m=0}^{N} \left(\sum_{k=0}^{N} \alpha_{km} \phi(\|k-I-d\|) \right) s(n-m) \quad (19)$$

Compared Eq.(19) with Eq.(11), the weights $h(m,d)$ are given by

$$h(m,d) = \sum_{k=0}^{N} \alpha_{km} \phi(\|k-I-d\|) \quad (20)$$

Finally, given the radical basis function $\phi(t)$ with shape parameter σ, integer N, and delay $I+d$, the procedure to estimate fractionally delayed sample $s(n-I-d)$ from the given integer delayed samples $s(n)$, $s(n-1)$, $s(n-2)$,..., $s(n-N)$ is summarized below:

Step 1: Compute the matrix Φ whose elements are given by $\phi_{mk} = \phi(\|k-m\|)$.

Step2: Calculate the inverse matrix Φ^{-1} with element α_{mk}.

Step 3: Use Eq.(20) to compute the weights $h(m,d)$.

Step 4: The fractionally delayed sample is estimated by $s(n-I-d) = \sum_{m=0}^{N} h(m,d)s(n-m)$.

4. DESIGN OF DIGITAL IIR INTEGRATOR

In this section, the numerical integration rule and RBF-based fractionally delay estimation method are used to design digital IIR integrator. When a signal $s(t)$ passes through the ideal integrator with integer delay I, its output $y(t)$ is given by

$$y(t) = \int_{-\infty}^{t-I} s(\tau)d\tau \tag{21}$$

Setting $t = n-1$ and $t = n$, we have

$$y(n-1) = \int_{-\infty}^{n-I-1} s(\tau)d\tau \tag{22a}$$

$$y(n) = \int_{-\infty}^{n-I} s(\tau)d\tau \tag{22b}$$

Using the following equality:

$$\int_{-\infty}^{n-I} s(\tau)d\tau = \int_{-\infty}^{n-I-1} s(\tau)d\tau + \int_{n-I-1}^{n-I} s(\tau)d\tau \tag{23}$$

we get

$$y(n) = y(n-1) + \int_{n-I-1}^{n-I} s(\tau)d\tau \tag{24}$$

Thus, the design problem reduces to how to evaluate the definite integral of the second term in Eq.(24). This problem can be solved by using various numerical integration rules in textbook [9]. If the trapezoidal rule is used, second term in Eq.(24) can be approximated by

$$\int_{n-I-1}^{n-I} s(\tau)d\tau \approx \frac{1}{2}\big(s(n-I-1) + s(n-I)\big) \tag{25}$$

Substituting Eq.(25) into Eq.(24), we have

$$y(n) = y(n-1) + \frac{1}{2}\big(s(n-I-1) + s(n-I)\big) \tag{26}$$

Taking the z transform at both sides, we obtain

$$H(z) = \frac{Y(z)}{S(z)} = \frac{1}{2}\frac{z^{-I}(1+z^{-1})}{1-z^{-1}} \tag{27}$$

where $S(Z)$ and $Y(Z)$ are the z transforms of $s(n)$ and $y(n)$. The above integrator $H(Z)$ is the well-known trapezoidal integrator in the literature. This integrator has large approximation error in the high frequency region, so RBF interpolation method and other integration rules will be used to improve the design accuracy. Two typical cases are studied below:

Case 1: Simpson 1/3 rule

Using the Simpson 1/3 rule, second term in Eq.(24) can be approximated by

$$\int_{n-I-1}^{n-I} s(\tau)d\tau \approx \frac{1}{6}\begin{pmatrix} s(n-I) + 4s(n-I-\frac{1}{2}) \\ + s(n-I-1) \end{pmatrix} \tag{28}$$

Because $s(n-I-\frac{1}{2})$ is a unknown fractionally delayed sample, the RBF interpolation method is used to estimate its value from the known integer delayed samples $s(n)$, $s(n-1)$, …, $s(n-N)$. The estimation formula has been described in Eq.(11) and Eq.(20). Thus, using Eq.(11), Eq.(28) can be written as

$$\int_{n-I-1}^{n-I} s(\tau)d\tau \approx \frac{1}{6}\begin{pmatrix} s(n-I) + 4\sum_{m=0}^{N} h(m,\frac{1}{2})s(n-m) \\ + s(n-I-1) \end{pmatrix} \tag{29}$$

Substituting Eq.(29) into Eq.(24), we have

$$\begin{aligned} y(n) = {} & y(n-1) + \frac{1}{6}s(n-I) + \frac{1}{6}s(n-I-1) \\ & + \frac{4}{6}\sum_{m=0}^{N} h(m,\tfrac{1}{2})s(n-m) \end{aligned} \tag{30}$$

Taking the z transform at both sides, we obtain

$$H_1(z) = \frac{Y(z)}{S(z)} = \frac{1}{6}\frac{z^{-I} + z^{-I-1} + 4\sum_{m=0}^{N} h(m,\frac{1}{2})z^{-m}}{1-z^{-1}} \tag{31}$$

The above $H_1(Z)$ is the designed Simpson 1/3 integrator using RBF interpolation method.

Case 2: Simpson 3/8 rule

Using the Simpson 3/8 rule, second term in Eq.(24) can be approximated by

$$\int_{n-I-1}^{n-I} s(\tau)d\tau \approx \frac{1}{8}\begin{pmatrix} s(n-I) + 3s(n-I-\frac{1}{3}) \\ + 3s(n-I-\frac{2}{3}) + s(n-I-1) \end{pmatrix} \tag{32}$$

Using fractionally delayed sample estimation formula in Eq.(11), the above equation can be written as

$$\int_{n-I-1}^{n-I} s(\tau)d\tau \approx \frac{1}{8}\begin{pmatrix} s(n-I) + s(n-I-1) \\ + 3\sum_{m=0}^{N} g(m)s(n-m) \end{pmatrix} \tag{33}$$

where

$$g(m) = h(m,\tfrac{1}{3}) + h(m,\tfrac{2}{3}) \tag{34}$$

Substituting Eq.(33) into Eq.(24), we have

$$\begin{aligned} y(n) = {} & y(n-1) + \frac{1}{8}s(n-I) + \frac{1}{8}s(n-I-1) \\ & + \frac{3}{8}\sum_{m=0}^{N} g(m)s(n-m) \end{aligned} \tag{35}$$

Taking the z transform at both sides, we obtain

$$H_2(z) = \frac{Y(z)}{S(z)} = \frac{1}{8}\frac{z^{-I} + z^{-I-1} + 3\sum_{m=0}^{N} g(m)z^{-m}}{1-z^{-1}} \tag{36}$$

The above $H_2(Z)$ is the designed Simpson 3/8 integrator using RBF interpolation method.

5. DESIGN EXAMPLES AND COMPARISON

In this section, we will study the design error of the proposed RBF-based integrator and compare it with conventional integrators. To evaluate the performance, the integral squares error of frequency response is defined by

$$E_k = \sqrt{\int_0^{\lambda\pi} \left| H_k(e^{j\omega}) - D(\omega) \right|^2 d\omega} \qquad (37)$$

Obviously, the smaller the error E_k is, the better the performance of the design method is.

Example 1: In this example, we first study the relation between design error E_k and shape parameter σ for Gaussian radial basis function $\phi(t)$ in Eq.(4a). The parameters are chosen as *N*=30, *I*=15, and $\lambda = 0.95$. Fig.2 shows the error curve E_k of the proposed RBF integrator $H_k(Z)$ for $\sigma \in [1.5,\ 3.5]$. From these results, it is clear that the error E_1 is almost same as the error E_2. And, both errors reach the minimum value 0.013 when $\sigma \approx 2.3$ is chosen. Moreover, Fig.3 depicts the magnitude responses (solid line) of the Simpson 1/3 integrator $H_1(Z)$ for Gaussian RBF with $\sigma = 2.3$. The dashed line is the ideal magnitude response $\frac{1}{\omega}$. Obviously, the specification is fitted well.

Example 2: In this example, we will study the relation between error E_k and parameter σ for inverse multiquadric radial basis function $\phi(t)$ in Eq.(4b). The parameters are chosen as *N*=30, *I*=15, and $\lambda = 0.95$. Fig.4 shows the error curve E_k of the integrator $H_k(Z)$ for $\sigma \in [5,\ 9.5]$. From these results, it is clear that the error E_1 is slightly smaller than the error E_2. And, both errors almost reach the minimum value 0.017 when $\sigma \approx 7$ is chosen. Because this minimum value is slightly larger than the value in Gaussian RBF case, Gaussian RBF is preferred in integrator design. Moreover, Fig.5 depicts the magnitude responses (solid line) of the Simpson 1/3 integrator $H_1(Z)$ for inverse multiquadric RBF with $\sigma = 7$. The dashed line is the ideal magnitude response $\frac{1}{\omega}$. Obviously, the specification is fitted well. Compared Fig.3 with Fig.5, it is easy to see that the magnitude responses are almost the same for Gaussian and inverse multiquadric RBF design cases.

Example 3: In this example, we compare Gaussian RBF integrator with the Ngo integrator in Eq.(2) under the same implementation complexity. The design parameters are chosen as *N*=3, *I*=1 and $\lambda = 0.95$. Fig.6(a) shows the error curve E_1 of the integrator $H_1(Z)$. From this result, it is clear that the error E_1 reach the minimum value 0.192 when $\sigma \approx 3.3$ is chosen. Fig.6(b) shows the frequency response error $20\log_{10}(|D(\omega) - F_1(e^{j\omega})|)$. The dashed line is the error $20\log_{10}(|D(\omega) - H_1(e^{j\omega})|)$ for Gaussian RBF integrator with $\sigma = 3.3$. Obviously, $H_1(z)$ has smaller error than Ngo integrator in the frequency band $[0.18\pi, \pi]$. However, Ngo integrator is better than Gaussian RBF integrator in the low frequency band $[0, 0.18\pi]$

Example 4: In this example, we compare Gaussian RBF integrator with the Tseng integrator in Eq.(3) under the same implementation complexity. The design parameters are chosen as *N*=4, *I*=1 and $\lambda = 0.95$. Fig.7(a) shows the error curve E_1 of the integrator $H_1(Z)$. From this result, it is clear that the error E_1 reach the minimum value 0.157 when $\sigma \approx 1.8$ is chosen. Fig.7(b) shows the frequency response error $20\log_{10}(|D(\omega) - F_2(e^{j\omega})|)$. The dashed line is the error $20\log_{10}(|D(\omega) - H_1(e^{j\omega})|)$ for Gaussian RBF integrator with $\sigma = 1.8$. Obviously, $H_1(z)$ has smaller error than Tseng integrator in the frequency band $[0.4\pi, \pi]$. However, Tseng integrator is better than Gaussian RBF integrator in the low frequency band $[0, 0.25\pi]$

6. CONCLUSIONS

In this paper, the design of digital IIR integrator using radial basis function interpolation method has been presented. The fractionally delayed sample estimation and numerical integration rule are applied to obtain the transfer function of digital integrator. The numerical comparisons with conventional digital integrators are also made. However, only digital integrator design is studied here. Thus, it is interesting to extended RBF interpolation method to design various digital filters in the future.

REFERENCES

[1] B. Kumar, D. Roy Choudhury and A. Kumar, "On the design of linear phase FIR integrators for midband frequencies," *IEEE Trans. on Signal Processing*, vol.44, pp.345-353, Oct. 1996.

[2] B. Kumar and A. Kumar, "FIR linear-phase approximations of frequency response $1/(j\omega)$ for maximal flatness at an arbitrary frequency ω_0, $(0 < \omega_0 < \pi)$," *IEEE Trans. on Signal Processing*, vol.47, pp.1772-1775, June 1999.

[3] N.Q. Ngo, "A new approach for the design of wideband digital integrator and differentiator," *IEEE Trans. on Circuits and Systems-II*, vol.53, pp.936-940, Sept. 2006.

[4] C.C. Tseng and S.L. Lee, "Digital IIR integrator design using Richardson extrapolation and fractional delay," *IEEE Trans. on Circuits and Systems-I,* vol.55, pp.2300-2309, Sept. 2008.

[5] S. Haykin, *Neural Network: A Comprehensive Foundation*, Second Edition, Prentice-Hall, 1999.

[6] G.E. Fasshauer, *Meshfree Approximation Methods with MATLAB*, World Scientific, 2007.

[7] J. Madapura and B. Li, Multi-target tracking based on KLD mixture particle filter with radial basis function support, ICASSP, pp.725-728, Apr. 2008.

[8] M.D. Buhmann, *Radial Basis Functions: Theory and Implementations*, Cambridge University Press, 2003.

[9] S.C. Chapra, *Applied Numerical Methods with MATLAB for Engineers and Scientists*, McGraw-Hill, 2005.

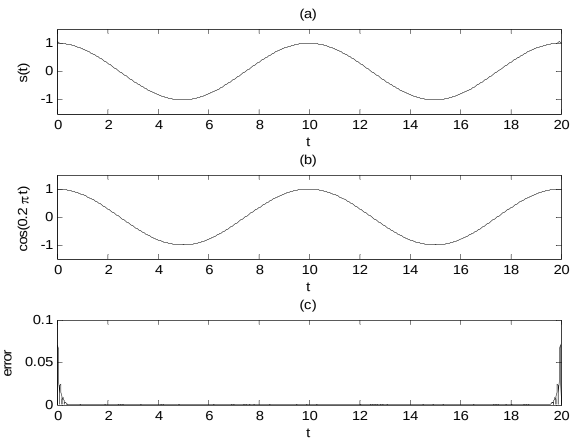

Fig.1 The radial basis function interpolation. (a) The interpolated function $s(t)$. (b) Sinusoidal function $\cos(0.2\pi t)$. (c) The absolute errors $|s(t)-\cos(0.2\pi t)|$.

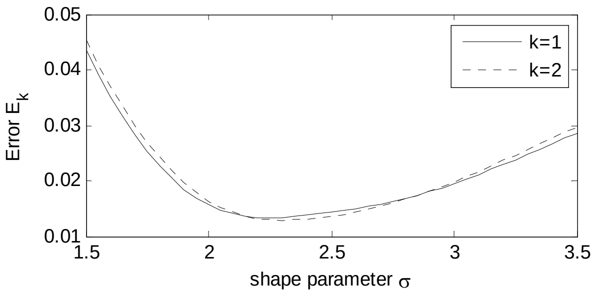

Fig.2 The error curve E_k of the proposed Gaussian RBF method for $\sigma \in [1.5,\ 3.5]$.

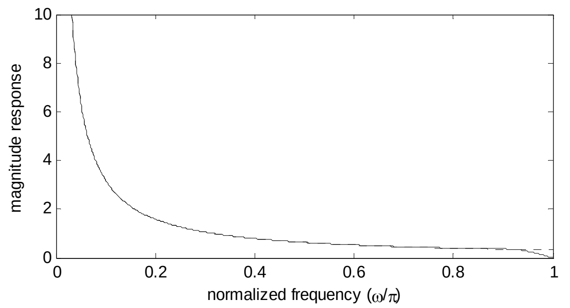

Fig.3 The magnitude response of Simpson 1/3 integrator $H_1(Z)$ designed by Gaussian RBF. The dashed line is the ideal magnitude response.

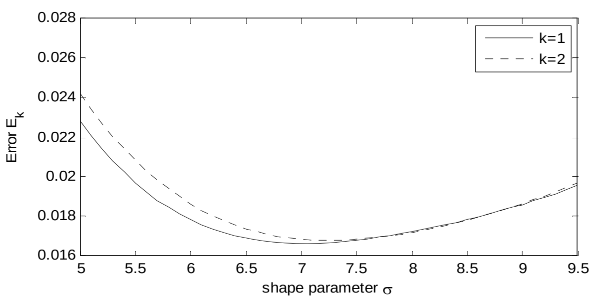

Fig.4 The error curve E_k of the proposed inverse multiquadric RBF method for $\sigma \in [5,\ 9.5]$.

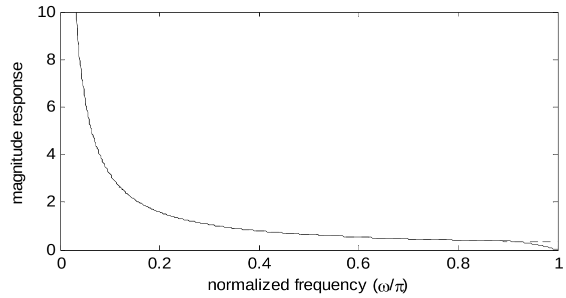

Fig.5 The magnitude response of Simpson 1/3 integrator $H_1(Z)$ designed by inverse multiquadric RBF. The dashed line is the ideal magnitude response.

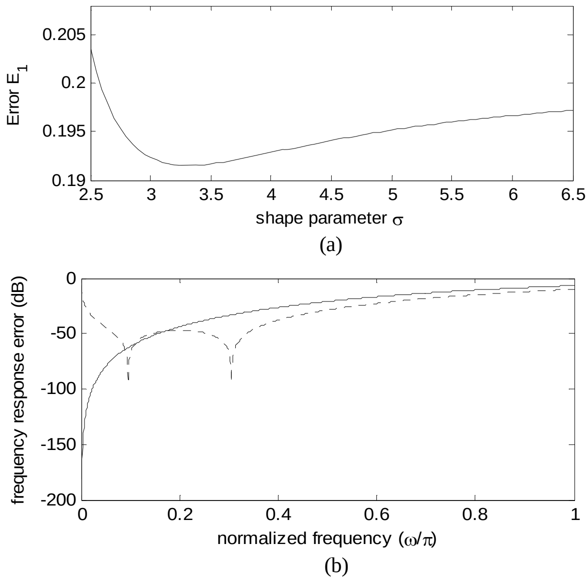

Fig.6 (a) Error curve E_1 . (b) Error $20\log_{10}(|D(\omega)-F_1(e^{j\omega})|)$. The dashed line is the error $20\log_{10}(|D(\omega)-H_1(e^{j\omega})|)$.

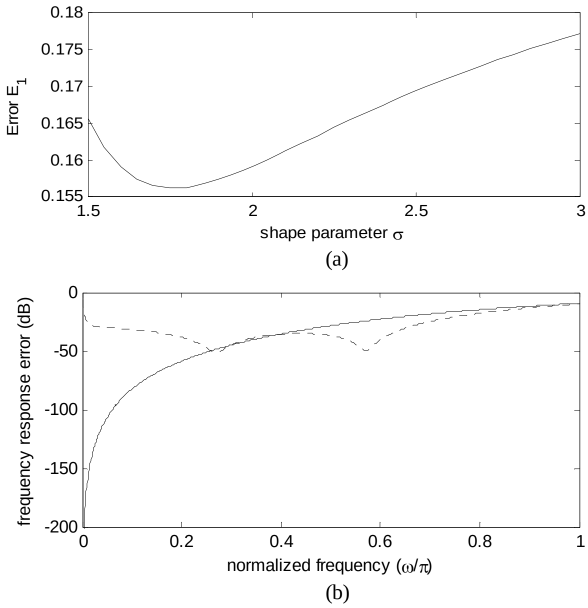

Fig.7 (a) Error curve E_1 . (b) Error $20\log_{10}(|D(\omega)-F_2(e^{j\omega})|)$. The dashed line is the error $20\log_{10}(|D(\omega)-H_1(e^{j\omega})|)$.