Bull. Nov. Comp. Center, Num. Anal., 2 (1993), 11–17

Data compression with $\Sigma\Pi$-approximations based on splines

O.E. Baklanova and V.A. Vasilenko

The paper contains short description of $\Sigma\Pi$-algorithm for the approximation of the function with two independent variables by the sum of products of one-dimensional functions. Some realizations of this algorithm based on the continuous and discrete splines are presented here. Few examples concerning with compression in the solving of approximation problems and colour image processing are described and discussed.

1. Theoretical background [1]

In the modern formulation the problem of the best $\Sigma\Pi$-approximation means the following. Let $X(\Omega_x)$ and $Y(\Omega_y)$ be two Hilbert functional spaces over multidimensional domains $\Omega_x \subset R^{n_x}$, $\Omega_y \subset R^{n_y}$, $n_x \geq 1$, $n_y \geq 1$, $\Omega = \Omega_x \times \Omega_y$ and $Z(\Omega) = X(\Omega_x) \otimes Y(\Omega_y)$ be the tensor product of the spaces [2]. For the function $f(x,y) \in Z(\Omega)$ we need to find the best $\Sigma\Pi$-approximation in the form

$$\sum_{k=1}^{s} \Phi^{(k)}(x) \cdot \Psi^{(k)}(y), \tag{1}$$

where $\Phi^{(k)} \in X(\Omega_x)$, $\Psi^{(k)} \in Y(\Omega_y)$, or in the particular case [1], they belong to the finite dimensional subspaces $X_n(\Omega_x)$ and $Y_m(\Omega_y)$ of the spaces $X(\Omega_x)$ and $Y(\Omega_y)$ respectively. For the unknown function $\Phi^{(k)}(x)$, $\Psi^{(k)}(y)$ we have the expansions

$$\begin{aligned} \Phi^{(k)}(x) &= \sum_{i=1}^{n} \alpha_i^{(k)} \varphi_i(x), \qquad n = \dim X_n, \\ \Psi^{(k)}(y) &= \sum_{j=1}^{m} \beta_j^{(k)} \psi_j(y), \qquad m = \dim Y_m, \end{aligned} \tag{2}$$

where $\varphi_1, \varphi_2, \ldots, \varphi_n$ form the basis in X_n, and $\psi_1, \psi_2, \ldots, \psi_m$ form the basis in Y_m. Thus, to find the best $\Sigma\Pi$-approximation we need to determine the coefficients $\alpha_i^{(k)}$, $\beta_j^{(k)}$, $i = \overline{1,n}$, $j = \overline{1,m}$, $k = \overline{1,s}$ from the minimization of the functional

$$E_{n,m}^{(s)}(\overline{\alpha}, \overline{\beta}) = \left\| f(x,y) - \sum_{k=1}^{s} \left[\sum_{i=1}^{n} \alpha_i^{(k)} \varphi_i(x) \sum_{j=1}^{m} \beta_j^{(k)} \psi_j(y) \right] \right\|_{Z(\Omega)}^2, \qquad (3)$$

with respect to these coefficients. Without lost of generality we assume that (see [1])

$$\|\Phi^{(k)}\|_X = \|\Psi^{(k)}\|_Y, \quad (\Phi^{(k)}, \Phi^{(l)})_X = (\Psi^{(k)}, \Psi^{(l)})_Y = 0, \quad k \neq l.$$

Let us introduce two Gram matrices

$$A = \left\{ (\varphi_i, \varphi_j)_X \right\}_{i,j=1}^{n}, \quad B = \left\{ (\psi_i, \psi_j)_Y \right\}_{i,j=1}^{m}$$

and the rectangular $m \times n$ - matrix F given by formula

$$F = \left\{ f_{ij} \right\}_{i,j=1}^{n,m}, \quad f_{ij} = \big(f(x,y), \varphi_i(x) \psi_j(y) \big)_Z,$$

and consider the generalized eigenvalue problem with the block matrices

$$\begin{bmatrix} \mathbf{0} & F \\ F^* & \mathbf{0} \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = \lambda \begin{bmatrix} A & \mathbf{0} \\ \mathbf{0} & B \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix}. \qquad (4)$$

It is easy to see [1] that the problem (4) has $s \leq \min\{n, m\}$ positive eigenvalues

$$\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_s > 0$$

and corresponding normalized linear independent eigenvectors

$$(Au^{(k)}, u^{(k)}) + (Bv^{(k)}, v^{(k)}) = 1, \quad k = \overline{1,s}$$

are connected with the optimal coefficients $\overline{\alpha}^{(k)}, \overline{\beta}^{(k)}$ by formula

$$\overline{\alpha}^{(k)} = \sqrt{2\lambda_k}\, u^{(k)}, \quad \overline{\beta}^{(k)} = \sqrt{2\beta_k}\, v^{(k)}. \qquad (5)$$

Taking into account the "aposteriori" error estimate [1]

$$E_{n,m}^{(s)} = \|f\|_Z^2 - \sum_{k=1}^{s} \lambda_k^2 \qquad (6)$$

one can see that the optimal strategy is the search of eigenvalues in order of decay to provide the best minimization of $E_{n,m}^{(s)}(\overline{\alpha}, \overline{\beta})$.

Using the well-known trick based on the Cholessky decomposition $A = LL^*$, $B = MM^*$ of the Gram matrices, it is possible to reduce the generalized eigenvalue problem (4) to usual eigenvalue problem

$$(L^{-1}FM^{*-1})^*(L^{-1}FM^{*-1})w = \lambda^2 w \tag{7}$$

of the order n or m.

It is easy to describe the main steps in the construction of the best ΣΠ-approximation with the given accuracy level $\varepsilon > 0$

Step 1. Choice of the basic Hilbert spaces X and Y

Step 2. Definition of the Hilbert tensor product $X \otimes Y$ corresponding to the suitable cross-norm

Step 3. Choice of the finite dimensional subspaces $X_n \subset X$, $Y_m \subset Y$ and their basises

Step 4. Assembling of the Gram matrices A and B

Step 5. Assembling of the rectangular $n \times m$ matrix F

Step 6. Consequent determination of the eigenvalues and eigenvectors to provide the given accuracy level in ΣΠ-approximation

The structure of the software for the construction of the best ΣΠ-approximation usually repeats this structure with some differences concerning with the concrete algorithm for the eigenvalue problem and some additional possibilities like visual control.

2. ΣΠ-approximation and polynomial splines

Let $\Omega_x = [a,b]$, $\Omega_y = [c,d]$, and $X(\Omega_x) = H^s(a,b)$, $Y(\Omega_y) = H^r(c,d)$ be the Hilbert spaces with the scalar products

$$\begin{aligned} (u,v)_{H^s(a,b)} &= \int_a^b [uv + u^{(s)}v^{(s)}]dx, \\ (u,v)_{H^r(c,d)} &= \int_c^d [uv + u^{(r)}v^{(r)}]dy. \end{aligned} \tag{8}$$

The tensor product $H^{s,r}(\Omega)$, $\Omega = [a,b] \times [c,d]$ of these two spaces with respect to the suitable cross-norm consists of the function of 2 variables with the norm

$$\|u(x,y)\|_{H^{r,s}(\Omega)} = \left[\int_a^b\int_c^d \left[u^2 + \left(\frac{\partial^s u}{\partial x^s}\right)^2 + \left(\frac{\partial^r u}{\partial y^r}\right)^2 + \left(\frac{\partial^{s+r} u}{\partial x^s \partial y^r}\right)^2\right] dxdy\right]^{1/2}$$

and corresponding scalar product.

Let us divide the intervals $[a, b]$ and $[c, d]$ by some mesh points,

$$a = x_1 < x_2 < \ldots < x_n = b, \quad c = y_1 < y_2 < \ldots < y_m = d$$

and connect with these meshes the finite dimensional subspaces $X_n = S_n \subset H^s(a, b)$, $Y_m = S_m \subset H^r(c, d)$ of the polynomial splines of the defect one with the suitable smoothness. It is well-known fact that the basises of the local functions (B-splines) do exist in three spline spaces [3]. If the function $f(x, y)$ belongs to the space $H^{r,s}(\Omega)$, then the prescribed $\Sigma\Pi$-procedure can be applied to approximate $f(x, y)$ by the sum of products of one-dimensional splines.

To estimate the compression coefficient we consider bicubic spline $f_h(x, y)$ connected with the uniform 6×6 rectangular grid in the unit square which interpolates the function

$$f(x, y) = \sin xy \cdot \exp(x^2 + y^2)$$

with the accuracy 10^{-3} in $H^{2,2}$-norm (see Fig.1). The exact representation of this spline with two-dimensional B-splines requires $8 \times 8 = 64$ coefficients. If we construct $\Sigma\Pi$-approximation based on the one-dimensional cubic splines with the same accuracy level 10^{-3} in $H^{2,2}$-norm, then only 2 eigenvalues are necessary to provide this error, and a number of coefficients in $\Sigma\Pi$-approximation is $2 \times 8 \times 2 = 32$; thus the compression coefficient is $64/32 = 2$.

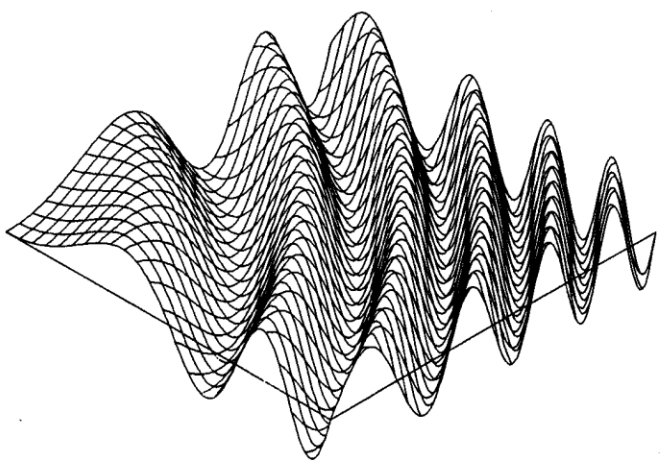

Figure 1

3. Using the discrete splines for data compression

If the initial two-dimensional function is given in the discrete form $f(i,j)$ like the image after scanning, then it is natural to apply the discrete splines instead continuous polynomial splines in ΣΠ-algorithm of data compression. Let

$$\Omega_x = \{1,2,\ldots,n\}, \quad \Omega_y = \{1,2,\ldots,m\}.$$

We denote by $H^r(\Omega_x)$ and $H^s(\Omega_y)$ the Hilbert spaces of the mesh functions with the norms

$$\|u\|_{H^r(\Omega_x)} = \left[\frac{1}{n}\sum u^2(i) + \frac{1}{n-r}\sum[(\Delta_r u)(i)]^2\right]^{1/2},$$

$$\|v\|_{H^s(\Omega_y)} = \left[\frac{1}{m}\sum v^2(i) + \frac{1}{m-s}\sum[(\Delta_s v)(i)]^2\right]^{1/2},$$

where Δ_r, Δ_s mean the divided differences of the order r or m. The tensor product $H^{r,s}(\Omega_x \times \Omega_y)$ can be defined as the Hilbert space of two-dimensional mesh function with the cross-norm

$$\begin{aligned}\|u\|^{H^{r,s}} = & \left[\frac{1}{nm}\sum u^2(i,j) + \frac{1}{(n-r)m}\sum(\Delta_r u)^2(i,j)\right. \\ & + \frac{1}{n(m-s)}\sum(\Delta_s u)^2(i,j) \\ & \left. + \frac{1}{(n-r)(m-s)}\sum(\Delta_r\Delta_s u)^2(i,j)\right]^{1/2},\end{aligned}$$

where Δ_r and Δ_s mean the finite difference differentiation with respect to 1-st and 2-nd variable.

Let $w_x \geq 1$, $w_y \geq 1$ be some integers, and $n = k_x \cdot w_x$, $m = k_y \cdot w_y$ where k_x, k_y are also integer. It is easy to connect with the "uniform mesh" $1, w, 2w, \ldots, k \cdot w$ the space of discrete splines with the basis of discrete B-splines; these B-splines can be obtained by the few discrete convolutions of the mesh piecewise constant function

$$\chi(i) = \begin{cases} 1, & -\left[\frac{w}{2}\right] \leq i \leq \left[\frac{w}{2}\right], \\ 0, & \text{otherwise.} \end{cases}$$

We describe now an example of data compression with the discrete splines. Let the function

$$f(x,y) = \sin(4\pi(x^2 + y^2 + xy))$$

be given at the unit square $[0,1] \times [0,1]$ (see Fig.2). We consider the corresponding mesh function at 130×130 uniform grid and approximate it with the accuracy level 0.01 in the mesh $H^{2,2}$-norm by the $\Sigma\Pi$-function with one dimensional discrete splines, corresponding to $w = 3$. We need only nine terms to provide this accuracy, and compression coefficient in this case is about 41.4.

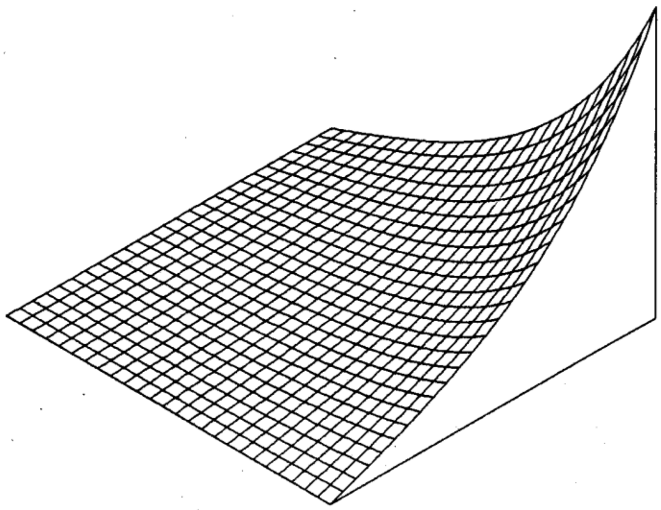

Figure 2

4. Compression of colour images

In the red-green-blue (RGB) representation the colour image is the vector function of these components. At first we extract from the colour image the half-tone component by the following law: every colour pixel (r_i, g_i, b_i) corresponds to the scalar $a_i = \min\{r_i, g_i, b_i\}$. The rest part of colour image consists of the pixels $(r_i - a_i, g_i - a_i, b_i - a_i)$. Formally we divide the initial colour image into four half-tone components $I_1 = \{a_i\}$, $I_2 = \{r_i - a_i\}$, $I_3 = \{g_i - a_i\}$, $I_4 = \{b_i - a_i\}$. Each of these components is the scalar mesh function of two integer variables, and for each of this functions we apply $\Sigma\Pi$-algorithm based on discrete one-dimensional splines with $w = 1$. In this case the "degree" of the splines does not matter, and the norm which we choose is $H^{0,0}$-norm.

There is no way to demonstrate the compressed and the initial colour image in this edition because colour photographs are impossible here. By these means we give only an expert evaluations for some colour 320×200-

images. To provide the exact restoration of this image we need 200 terms in ΣΠ-approximation. But the visual control shows that the number of terms is extremely less, especially for the components I_2, I_3, I_4. We show the expert evaluation of the colour image quality in the following table

Number of eigenvalues				Expert evaluation
I_1	I_2	I_3	I_4	
50	1	1	1	fair
50	3	2	3	good
50	7	6	10	excellent

References

[1] V.A.Vasilenko, The best finite dimensional ΣΠ-approximation, Sov. J. Num. Anal. Math. Mod., Vol. 5, No. 4/5, 1990, 435–443.

[2] W.A.Light, E.W.Cheney, Approximation theory in tensor product spaces, Lectures Notes in Math., Springer Verlag, 1985.

[3] C.De.Boor, A practical guide to splines, Appl.Math.Sci., 27, Springer Verlag, 1978.