Probabilistic Interpolation of the Curve via the Method of Hurwitz-Radon Matrices

Dariusz Jakóbczak

Department of Electronics and Computer Science, Technical University of Koszalin
Sniadeckich 2, 75-453 Koszalin, Poland
dariusz.jakobczak@tu.koszalin.pl

Abstract
Mathematics and computer science are interested in methods of curve interpolation using the set of key points (knots). A proposed method of Hurwitz- Radon Matrices (MHR) is such a method. This novel method is based on the family of Hurwitz-Radon (HR) matrices which possess columns composed of orthogonal vectors. Two-dimensional curve is interpolated via different functions as probability distribution functions: polynomial, sinus, cosine, tangent, cotangent, logarithm, exponent, arcsin, arccos, arctan, arcctg or power function, also inverse functions. It is shown how to build the orthogonal matrix operator and how to use it in a process of curve reconstruction.
Keywords: *Curve Interpolation, Hurwitz-Radon Matrices, Coefficient Of MHR Method, Probabilistic Interpolation*

1. Introduction

The author presents new approach to a subject of 2D curve interpolation. This is not polynomial or trigonometric interpolation but probabilistic interpolation. The method of Hurwitz-Radon Matrices (MHR) consists in the calculations of each interpolated point via parameter $\alpha \in [0;1]$. Second coordinate y is computed using the probability distribution functions $\gamma = F(\alpha)$ for random variable $\alpha \in [0;1]$. The family of Hurwitz-Radon matrices, applied in MHR method, requires square matrices of dimension $N = 1, 2, 4$ and 8. Interpolated point $(x;y)$ of the curve is calculated via successive $2N$ knots.

Curve interpolation[1] represents one of the most important problems in mathematics: how to model the curve[2] via discrete set of two-dimensional points[3]? Also the matter of curve representation and parameterization is still opened in mathematics and computer sciences[4]. The author wants to approach a problem of curve modeling by characteristic points. Proposed method relies on functional modeling of curve points situated between the basic set of the nodes. The functions that are used in calculations represent whole family of elementary functions with inverse functions: polynomials, trigonometric, cyclometric, logarithmic, exponential and power function. These functions are treated as probability distribution functions in the range[0;1]. Nowadays methods apply mainly polynomial functions, for example Bernstein polynomials in Bezier curves, splines and NURBS[5]. Numerical methods for data interpolation are based on polynomial or trigonometric functions, for example Lagrange, Newton, Aitken and Hermite methods. These methods have some weak sides[6] and are not sufficient for curve interpolation in the situations when the curve cannot be build by polynomials or trigonometric functions. Proposed curve interpolation is the functional modeling via any elementary functions and it helps us to fit the curve during the computations.

The author presents novel method of curve interpolation. This paper takes up new method of two-dimensional curve modeling via the family of Hurwitz-Radon matrices. The method of Hurwitz-Radon Matrices (MHR) requires minimal assumptions: the only information about a curve is the set of at least two nodes. Proposed method of Hurwitz-Radon Matrices (MHR) is applied in curve modeling via different coefficients: polynomial, sinusoidal, cosinusoidal, tangent, cotangent, logarithmic, exponential, arcsin, arccos, arctan, arcctg or power. Function for MHR calculations is chosen individually at each interpolation and it represents probability distribution function of parameter $\alpha \in [0;1]$ for every point situated between two interpolation knots. MHR method uses two-dimensional vectors (x,y) for curve modeling - knots $(x_i,y_i) \in \mathbf{R}^2$ in MHR method:

1. MHR version with no matrices ($N = 1$) needs 2 knots or more;
2. At least five knots (x_1,y_1), (x_2,y_2), (x_3,y_3), (x_4,y_4) and (x_5,y_5) if MHR method is implemented with matrices of dimension $N = 2$;
3. For better interpolation knots ought to be settled at key points of the curve, for example local minimum or maximum and at least one node between two successive local extrema.

Condition 2 is connected with important features of MHR method: MHR version with matrices of dimension $N = 2$ (MHR-2) requires at least five nodes, MHR version with matrices of dimension $N = 4$ (MHR-4) needs at least nine nodes and MHR version with matrices of dimension $N = 8$

ACSIJ Advances in Computer Science: an International Journal, Vol. 3, Issue 3, No.9 , May 2014
ISSN : 2322-5157
www.ACSIJ.org

(MHR-8) requires at least 17 nodes. Condition 3 means for example the highest point of the curve in a particular orientation, convexity changing or curvature extrema. So this paper wants to answer the question: how to interpolate the curve by a set of knots?

Coefficients for curve interpolation are computed using probability distribution functions: polynomials, power functions, sinus, cosine, tangent, cotangent, logarithm, exponent or arcsin, arccos, arctan, arcctg.

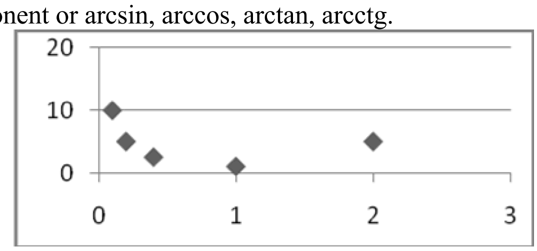

Fig. 1 Knots of the curve before interpolation.

2. Probabilistic Interpolation

The method of Hurwitz – Radon Matrices (MHR) is computing points between two successive nodes of the curve. Calculated points are interpolated and parameterized for real number $\alpha \in [0;1]$ in the range of two successive nodes. MHR calculations are dealing with square matrices of dimension N = 1, 2, 4 or 8. Matrices A_i, i = 1,2…m satisfying

$A_jA_k + A_kA_j = 0, \quad A_j^2 = -I \quad$ for $j \neq k$; j, k = 1,2...m

are called a family of Hurwitz - Radon matrices. They were discussed by Adolf Hurwitz and Johann Radon separately in 1923. A family of Hurwitz - Radon (HR) matrices[7] are skew-symmetric: $A_i^T = -A_i$ and $A_i^{-1} = -A_i$. Only for dimensions N = 1, 2, 4 or 8 the family of HR matrices consists of N - 1 matrices. For N = 1 there is no matrices but only calculations with real numbers. For N = 2:

$$A_1 = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}.$$

For N = 4 there are three HR matrices with integer entries:

$$A_1 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \quad A_2 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{bmatrix},$$

$$A_3 = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \end{bmatrix}.$$

For N = 8 we have seven HR matrices with elements 0, ±1. So far HR matrices have found applications in Space-Time Block Coding (STBC)[8] and orthogonal design[9], in signal processing[10] and Hamiltonian Neural Nets[11].

How coordinates of knots are applied for interpolation? If knots are represented by the following points $\{(x_i,y_i),\ i = 1, 2, \ldots, n\}$ then HR matrices combined with the identity matrix I_N are used to build the orthogonal Hurwitz - Radon Operator (OHR). For point $p_1=(x_1,y_1)$ and $x_1 \neq 0$ OHR of dimension N=1 is the matrix (real number) M_1:

$$M_1(p_1) = \frac{1}{x_1^2}[x_1 \cdot y_1] = \frac{y_1}{x_1}. \qquad (0)$$

For points $p_1=(x_1,y_1)$ and $p_2=(x_2,y_2)$ OHR of dimension N = 2 is build via matrix M_2:

$$M_2(p_1,p_2) = \frac{1}{x_1^2 + x_2^2}\begin{bmatrix} x_1y_1 + x_2y_2 & x_2y_1 - x_1y_2 \\ x_1y_2 - x_2y_1 & x_1y_1 + x_2y_2 \end{bmatrix}. \qquad (1)$$

For points $p_1=(x_1,y_1)$, $p_2=(x_2,y_2)$, $p_3=(x_3,y_3)$ and $p_4=(x_4,y_4)$ OHR M_4 of dimension N = 4 is introduced:

$$M_4(p_1,p_2,p_3,p_4) = \frac{1}{x_1^2 + x_2^2 + x_3^2 + x_4^2}\begin{bmatrix} u_0 & u_1 & u_2 & u_3 \\ -u_1 & u_0 & -u_3 & u_2 \\ -u_2 & u_3 & u_0 & -u_1 \\ -u_3 & -u_2 & u_1 & u_0 \end{bmatrix} \qquad (2)$$

where

$$u_0 = x_1y_1 + x_2y_2 + x_3y_3 + x_4y_4, \quad u_1 = -x_1y_2 + x_2y_1 + x_3y_4 - x_4y_3,$$

$$u_2 = -x_1y_3 - x_2y_4 + x_3y_1 + x_4y_2,$$

$$u_3 = -x_1y_4 + x_2y_3 - x_3y_2 + x_4y_1.$$

For knots $p_1=(x_1,y_1)$, $p_2=(x_2,y_2)$,… and $p_8=(x_8,y_8)$ OHR M_8 of dimension N = 8 is constructed[12] similarly as (1) and (2):

$$M_8(p_1,p_2 \ldots p_8) = \frac{1}{\sum_{i=1}^{8} x_i^2}\begin{bmatrix} u_0 & u_1 & u_2 & u_3 & u_4 & u_5 & u_6 & u_7 \\ -u_1 & u_0 & u_3 & -u_2 & u_5 & -u_4 & -u_7 & u_6 \\ -u_2 & -u_3 & u_0 & u_1 & u_6 & u_7 & -u_4 & -u_5 \\ -u_3 & u_2 & -u_1 & u_0 & u_7 & -u_6 & u_5 & -u_4 \\ -u_4 & -u_5 & -u_6 & -u_7 & u_0 & u_1 & u_2 & u_3 \\ -u_5 & u_4 & -u_7 & u_6 & -u_1 & u_0 & -u_3 & u_2 \\ -u_6 & u_7 & u_4 & -u_5 & -u_2 & u_3 & u_0 & -u_1 \\ -u_7 & -u_6 & u_5 & u_4 & -u_3 & -u_2 & u_1 & u_0 \end{bmatrix} \qquad (3)$$

where

$$\underline{u} = \begin{bmatrix} y_1 & y_2 & y_3 & y_4 & y_5 & y_6 & y_7 & y_8 \\ -y_2 & y_1 & -y_4 & y_3 & -y_6 & y_5 & y_8 & -y_7 \\ -y_3 & y_4 & y_1 & -y_2 & -y_7 & -y_8 & y_5 & y_6 \\ -y_4 & -y_3 & y_2 & y_1 & -y_8 & y_7 & -y_6 & y_5 \\ -y_5 & y_6 & y_7 & y_8 & y_1 & -y_2 & -y_3 & -y_4 \\ -y_6 & -y_5 & y_8 & -y_7 & y_2 & y_1 & y_4 & -y_3 \\ -y_7 & -y_8 & -y_5 & y_6 & y_3 & -y_4 & y_1 & y_2 \\ -y_8 & y_7 & -y_6 & -y_5 & y_4 & y_3 & -y_2 & y_1 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \end{bmatrix} \qquad (4)$$

and $\underline{u} = (u_0, u_1, \ldots, u_7)^T$ (4). OHR operators M_N (0)-(3) satisfy the condition of interpolation

$$M_N \cdot x = y \qquad (5)$$

for $\mathbf{x} = (x_1,x_2\ldots,x_N)^T \in \boldsymbol{R}^N$, $\mathbf{x} \neq \mathbf{0}$, $\mathbf{y} = (y_1,y_2\ldots,y_N)^T \in \boldsymbol{R}^N$ and N = 1, 2, 4 or 8.

2.1 Distribution Functions in MHR Interpolation

Points settled between the nodes are computed[13] using MHR method[14]. Each real number $c \in [a;b]$ is calculated by a convex combination $c = \alpha \cdot a + (1 - \alpha) \cdot b$ for

$$\alpha = \frac{b-c}{b-a} \in [0;1]. \qquad (6)$$

The weighted average OHR operator M of dimension N = 1, 2, 4 or 8 is build:

$$M = \gamma \cdot A + (1-\gamma) \cdot B\,. \tag{7}$$

The OHR matrix A is constructed (1)-(3) by every second knot $p_1=(x_1=a,y_1)$, $p_3=(x_3,y_3)$, …and $p_{2N-1}=(x_{2N-1},y_{2N-1})$:
$A = M_N(p_1, p_{3,\dots,} p_{2N-1})$.

The OHR matrix B is computed (1)-(3) by knots $p_2=(x_2=b,y_2)$, $p_4=(x_4,y_4)$,… and $p_{2N}=(x_{2N},y_{2N})$:
$B = M_N(p_2, p_{4,\dots,} p_{2N})$.

Vector of first coordinates C is defined for

$$c_i = \alpha \cdot x_{2i-1} + (1-\alpha) \cdot x_{2i} \quad , \quad i = 1, 2,\dots, N \tag{8}$$

and $C = [c_1, c_2,\dots, c_N]^T$. The formula to calculate second coordinates $y(c_i)$ is similar to the interpolation formula (5):

$$Y(C) = M \cdot C \tag{9}$$

where $Y(C) = [y(c_1), y(c_2),\dots, y(c_N)]^T$. So interpolated value $y(c_i)$ from (9) depends on two, four, eight or sixteen ($2N$) successive nodes. For example N = 1 results in computations without matrices:

$$A = M_1(p_1) = \frac{y_1}{x_1}, \quad B = M_1(p_2) = \frac{y_2}{x_2}, \quad C = c_1 = \alpha \cdot x_1 + (1-\alpha) \cdot x_2\,,$$

$$Y(C) = y(c_1) = (\gamma \frac{y_1}{x_1} + (1-\gamma)\frac{y_2}{x_2}) \cdot c_1,$$

$$y(c_1) = \alpha \cdot \gamma \cdot y_1 + (1-\alpha)(1-\gamma) y_2 + \gamma(1-\alpha)\frac{y_1}{x_1} x_2 + \alpha(1-\gamma)\frac{y_2}{x_2} x_1\,. \tag{10}$$

Key question is dealing with coefficient γ in (7). Basic MHR version means $\gamma = \alpha$ and then (10):

$$y(c_1) = \alpha^2 \cdot y_1 + (1-\alpha)^2 y_2 + \alpha(1-\alpha)(\frac{y_1}{x_1} x_2 + \frac{y_2}{x_2} x_1). \tag{11}$$

Formula (11) represents the simplest way of MHR calculations (N = 1, $\gamma = \alpha$) and it differs from linear interpolation $y(c) = \alpha \cdot y_1 + (1-\alpha) y_2$. MHR is not a linear interpolation.

Each interpolation requires specific distribution of parameter α (7) and γ depends on parameter $\alpha \in [0;1]$:
$\gamma = F(\alpha)$, $F:[0;1] \rightarrow [0;1]$, $F(0) = 0$, $F(1) = 1$
and F is strictly monotonic.

Coefficient γ is calculated using different functions (polynomials, power functions, sinus, cosine, tangent, cotangent, logarithm, exponent, arcsin, arccos, arctan or arcctg, also inverse functions) and choice of function is connected with initial requirements and curve specifications. Different values of coefficient γ are connected with applied functions $F(\alpha)$. These functions (12)-(41) represent the probability distribution functions for random variable $\alpha \in [0;1]$ and real number $s > 0$:

1. power function

$$\gamma = \alpha^s \quad \text{with} \quad s > 0. \tag{12}$$

For s = 1: basic version of MHR method when $\gamma = \alpha$.

2. sinus

$$\gamma = \sin(\alpha^s \cdot \pi/2)\,, \quad s > 0 \tag{13}$$

or

$$\gamma = \sin^s(\alpha \cdot \pi/2)\,, \quad s > 0. \tag{14}$$

For $s = 1$: $\gamma = \sin(\alpha \cdot \pi/2)$. (15)

3. cosine

$$\gamma = 1-\cos(\alpha^s \cdot \pi/2)\,, \quad s > 0 \tag{16}$$

or

$$\gamma = 1-\cos^s(\alpha \cdot \pi/2)\,, \quad s > 0. \tag{17}$$

For $s = 1$: $\gamma = 1-\cos(\alpha \cdot \pi/2)$. (18)

4. tangent

$$\gamma = \tan(\alpha^s \cdot \pi/4)\,, \quad s > 0 \tag{19}$$

or

$$\gamma = \tan^s(\alpha \cdot \pi/4)\,, \quad s > 0. \tag{20}$$

For $s = 1$: $\gamma = \tan(\alpha \cdot \pi/4)$. (21)

5. logarithm

$$\gamma = \log_2(\alpha^s + 1)\,, \quad s > 0 \tag{22}$$

or

$$\gamma = \log_2{}^s(\alpha + 1)\,, \quad s > 0. \tag{23}$$

For $s = 1$: $\gamma = \log_2(\alpha + 1)$. (24)

6. exponent

$$\gamma = (\frac{a^\alpha - 1}{a-1})^s \,, \; s > 0 \text{ and } a > 0 \text{ and } a \neq 1. \tag{25}$$

For $s = 1$ and $a = 2$: $\gamma = 2^\alpha - 1$. (26)

7. arc sine

$$\gamma = 2/\pi \cdot \arcsin(\alpha^s)\,, \quad s > 0 \tag{27}$$

or

$$\gamma = (2/\pi \cdot \arcsin \alpha)^s\,, \quad s > 0. \tag{28}$$

For $s = 1$: $\gamma = 2/\pi \cdot \arcsin(\alpha)$. (29)

8. arc cosine

$$\gamma = 1-2/\pi \cdot \arccos(\alpha^s)\,, \quad s > 0 \tag{30}$$

or

$$\gamma = 1-(2/\pi \cdot \arccos \alpha)^s\,, \quad s > 0. \tag{31}$$

For $s = 1$: $\gamma = 1-2/\pi \cdot \arccos(\alpha)$. (32)

9. arc tangent

$$\gamma = 4/\pi \cdot \arctan(\alpha^s)\,, \quad s > 0 \tag{33}$$

or

$$\gamma = (4/\pi \cdot \arctan \alpha)^s\,, \quad s > 0. \tag{34}$$

For $s = 1$: $\gamma = 4/\pi \cdot \arctan(\alpha)$. (35)

10. cotangent

$$\gamma = ctg(\pi/2 - \alpha^s \cdot \pi/4)\,, \quad s > 0 \tag{36}$$

or

$$\gamma = ctg^s(\pi/2 - \alpha \cdot \pi/4), \quad s > 0. \tag{37}$$

For $s = 1$: $\gamma = ctg(\pi/2 - \alpha \cdot \pi/4)$. (38)

11. arc cotangent

$$\gamma = 2 - 4/\pi \cdot arcctg(\alpha^s)\,, \quad s > 0 \tag{39}$$

or

$$\gamma = (2 - 4/\pi \cdot arcctg\, \alpha)^s\,, \quad s > 0. \tag{40}$$

For $s = 1$: $= 2 - 4/\pi \cdot arcctg(\alpha)$. (41)

Functions used in γ calculations (12)-(41) are strictly monotonic for random variable $\alpha \in [0;1]$ as $\gamma = F(\alpha)$ is probability distribution function. Also inverse function $F^{-1}(\alpha)$ is appropriate for γ calculations. Choice of function and value s depends on curve specifications and individual requirements. Interpolating of coordinates for curve points using (6)-(9) is called by author the method of Hurwitz - Radon Matrices (MHR)[15]. So here are five steps of MHR interpolation:

Step 1: Choice of knots at key points.

Step 2: Fixing the dimension of matrices N = 1, 2, 4 or 8: N = 1 is the most universal for calculations (it needs only two

ACSIJ Advances in Computer Science: an International Journal, Vol. 3, Issue 3, No.9 , May 2014
ISSN : 2322-5157
www.ACSIJ.org

nodes to compute unknown points between them) and it has the lowest computational costs (10); MHR with $N = 2$ uses four successive nodes to compute unknown coordinate; MHR version for $N = 4$ applies eight successive nodes to get unknown point and MHR with $N = 8$ needs sixteen successive nodes to calculate unknown coordinate (it has the biggest computational costs).

Step 3: Choice of distribution $\gamma = F(\alpha)$: basic distribution for $\gamma = \alpha$.

Step 4: Determining values of α: $\alpha = 0.1, 0.2…0.9$ (nine points) or $0.01, 0.02…0.99$ (99 points) or others.

Step 5: The computations (9).

These five steps can be treated as the algorithm of MHR method of curve modeling and interpolation (6)-(9).

Considering nowadays used probability distribution functions for random variable $\alpha\in[0;1]$ - one distribution is dealing with the range[0;1]: beta distribution. Probability density function f for random variable $\alpha \in[0;1]$ is:

$$f(\alpha) = c\cdot\alpha^s\cdot(1-\alpha)^r \quad , s\geq 0, r\geq 0. \qquad (42)$$

When $r = 0$ probability density function (42) represents $f(\alpha) = c\cdot\alpha^s$ and then probability distribution function F is like (12), for example $f(\alpha) = 3\alpha^2$ and $\gamma = \alpha^3$. If s and r are positive integer numbers then γ is the polynomial, for example $f(\alpha) = 6\alpha(1-\alpha)$ and $\gamma = 3\alpha^2-2\alpha^3$. So beta distribution gives us coefficient γ in (7) as polynomial because of interdependence between probability density f and distribution F functions:

$$f(\alpha) = F'(\alpha) \quad , \quad F(\alpha) = \int_0^\alpha f(t)dt\,. \qquad (43)$$

For example (43): $f(\alpha) = \alpha\cdot e^\alpha$ and $\gamma = F(\alpha) = (\alpha-1)e^\alpha + 1$.

What is very important: two curves may have the same set of nodes but different N or γ results in different interpolations (Fig.6-13). Here are some applications of MHR method with basic version ($\gamma = \alpha$): MHR-2 is MHR version with matrices of dimension $N = 2$ and MHR-4 means MHR version with matrices of dimension $N = 4$.

Figures 2-5 show interpolation of continues functions connected with determined formula. So these functions are interpolated and modeled. Without knowledge about the formula, curve interpolation has to implement the coefficients γ (12)-(43), but MHR is not limited only to these coefficients. Each strictly monotonic function F between points (0;0) and (1;1) can be used in MHR interpolation.

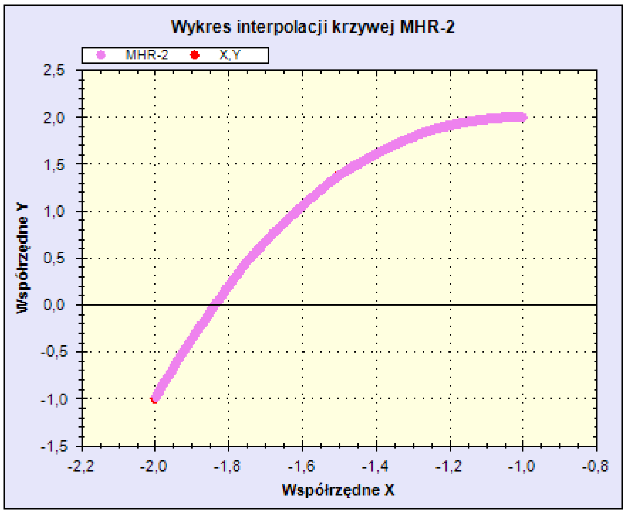

Fig. 2 Function $f(x) = x^3+x^2-x+1$ with 396 interpolated points using basic MHR-2 with 5 nodes.

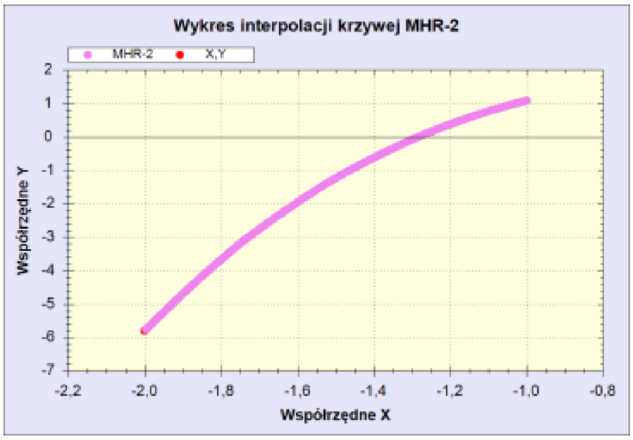

Fig. 3 Function $f(x) = x^3+\ln(7-x)$ with 396 interpolated points using basic MHR-2 with 5 nodes.

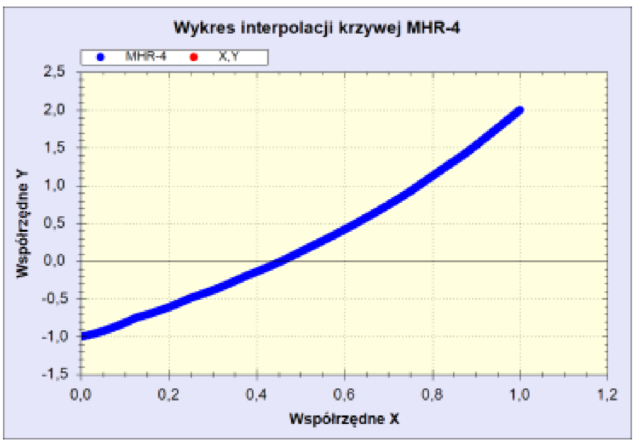

Fig. 4 Function $f(x) = x^3+2x-1$ with 792 interpolated points using basic MHR-4 with 9 nodes.

ACSIJ Advances in Computer Science: an International Journal, Vol. 3, Issue 3, No.9 , May 2014
ISSN : 2322-5157
www.ACSIJ.org

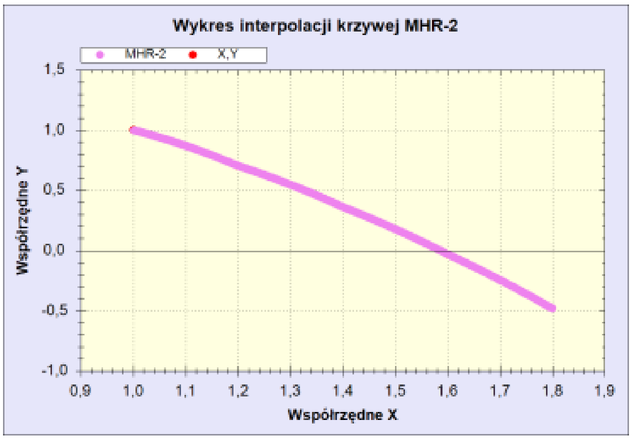

Fig. 5 Function $f(x) = 3\text{-}2^x$ with 396 interpolated points using basic MHR-2 with 5 nodes.

3. Implementations of MHR Probabilistic Interpolation

Curve knots (0.1;10), (0.2;5), (0.4;2.5), (1;1) and (2;5) from Fig.1 are used in some examples of MHR interpolation with different γ. Points of the curve are calculated for N = 1 and $\gamma = \alpha$ (11) in example 1 and with matrices of dimension N = 2 in examples 2 - 8 for $\alpha = 0.1, 0.2,\ldots,0.9$.

Example 1

Curve interpolation for N = 1 and $\gamma = \alpha$.

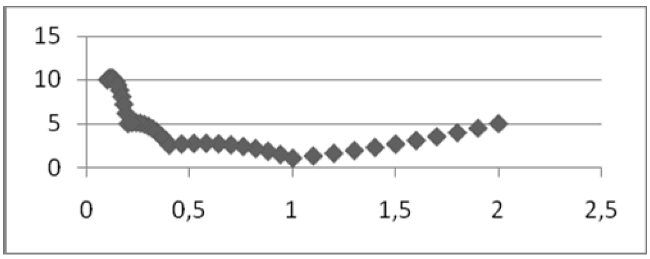

Fig. 6. Modeling without matrices (N = 1) for nine reconstructed points between nodes.

Example 2

Sinusoidal interpolation with $\gamma = sin(\alpha \cdot \pi/2)$.

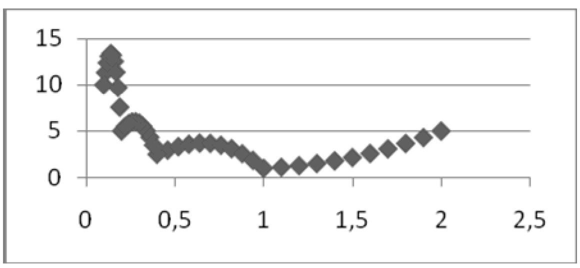

Fig. 7 Sinusoidal modeling with nine reconstructed curve points between nodes.

Example 3

Tangent interpolation for $\gamma = tan(\alpha \cdot \pi/4)$.

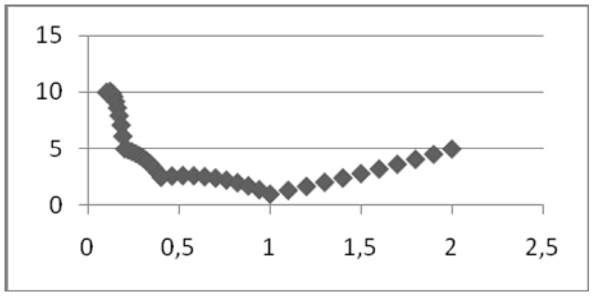

Fig. 8 Tangent curve modeling with nine interpolated points between nodes.

Example 4

Tangent interpolation with $\gamma = tan(\alpha^s \cdot \pi/4)$ and $s = 1.5$.

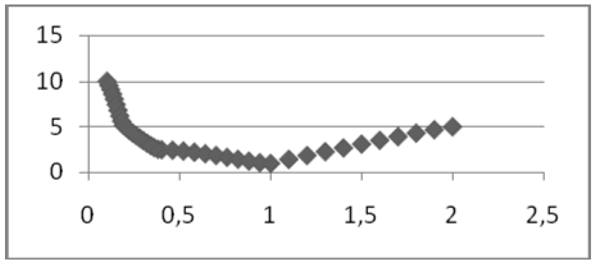

Fig. 9 Tangent modeling with nine recovered points between nodes.

Example 5

Tangent curve interpolation for $\gamma = tan(\alpha^s \cdot \pi/4)$ and $s = 1.797$.

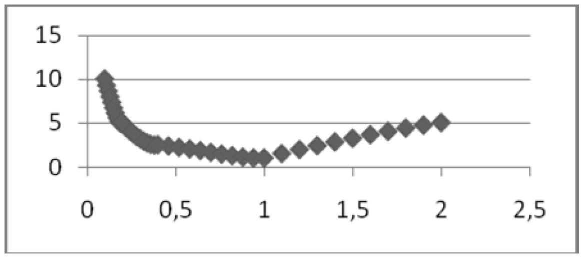

Fig. 10 Tangent modeling with nine reconstructed points between nodes.

Example 6

Sinusoidal interpolation with $\gamma = sin(\alpha^s \cdot \pi/2)$ and $s = 2.759$.

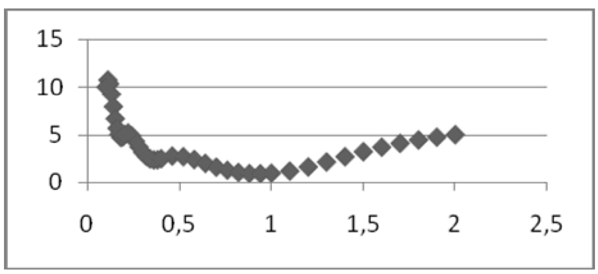

Fig. 11 Sinusoidal modeling with nine interpolated curve points between nodes.

Example 7

Power function modeling for $\gamma = \alpha^s$ and $s = 2.1205$.

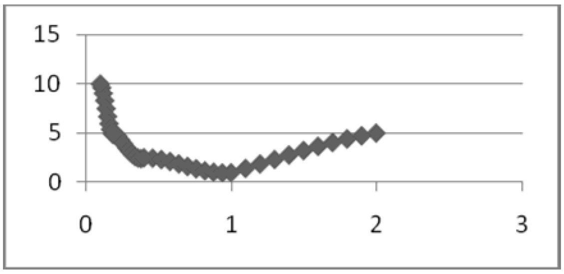

Fig. 12 Power function curve modeling with nine recovered points between nodes.

Example 8

Logarithmic curve modeling with $\gamma = log_2(\alpha^s + 1)$ and s = 2.533.

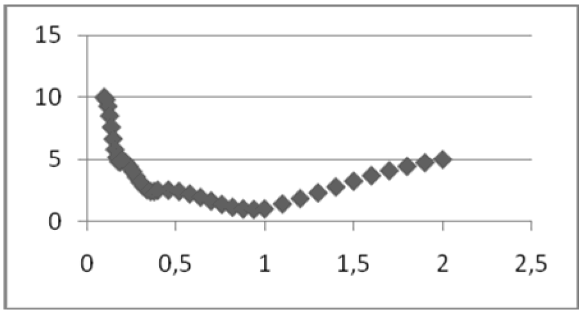

Fig. 13 Logarithmic modeling with nine reconstructed points between nodes.

These eight examples demonstrate possibilities of curve interpolation for key nodes. Reconstructed values and interpolated points, calculated by MHR method, are applied in the process of curve modeling. Every curve can be interpolated by some distribution function as parameter γ. This parameter is treated as probability distribution function for each curve.

The author presents new approach to a subject of 2D curve interpolation. This is not polynomial or trigonometric interpolation but probabilistic interpolation. The method of Hurwitz-Radon Matrices (MHR) consists in the calculations of each interpolated point via parameter $\alpha\in[0;1]$. Second coordinate y is computed using the probability distribution functions $\gamma = F(\alpha)$ for random variable $\alpha \in[0;1]$. The family of Hurwitz-Radon matrices, applied in MHR method, requires square matrices of dimension N = 1, 2, 4 and 8. Interpolated point $(x;y)$ of the curve is calculated via successive $2N$ knots.

4. Conclusions

The method of Hurwitz-Radon Matrices (MHR) enables interpolation of two-dimensional curves using different coefficients γ: polynomial, sinusoidal, cosinusoidal, tangent, cotangent, logarithmic, exponential, arcsin, arccos, arctan, arcctg or power function[16], also inverse functions. Function for γ calculations is chosen individually at each curve modeling and it is treated as probability distribution function: γ depends on initial requirements and curve specifications. MHR method leads to curve interpolation via discrete set of fixed knots. So MHR makes possible the combination of two important problems: interpolation and modeling. Main features of MHR method are:

a) the smaller distance between knots the better;

b) calculations for coordinate x close to zero and near by extremum require more attention;

c) MHR interpolation of the function is more precise then linear interpolation;

d) minimum two interpolation knots for calculations without matrices when N=1, but MHR is not a linear interpolation;

e) interpolation of L points is connected with the computational cost of rank $O(L)$;

f) MHR is well-conditioned method (orthogonal matrices)[17];

g) coefficient γ is crucial in the process of curve probabilistic interpolation and it is computed individually for a single curve.

Future works are going to: choice and features of coefficient γ, implementation of MHR in object recognition[18], shape geometry, contour modeling and parameterization[19].

References

[1] Collins II, G.W.: Fundamental Numerical Methods and Data Analysis. Case Western Reserve University (2003)

[2] Chapra, S.C.: Applied Numerical Methods. McGraw-Hill (2012)

[3] Ralston, A., Rabinowitz, P.: A First Course in Numerical Analysis – Second Edition. Dover Publications, New York (2001)

[4] Zhang, D., Lu, G.: Review of Shape Representation and Description Techniques. Pattern Recognition 1(37), 1-19 (2004)

[5] Schumaker, L.L.: Spline Functions: Basic Theory. Cambridge Mathematical Library (2007)

[6] Dahlquist, G., Bjoerck, A.: Numerical Methods. Prentice Hall, New York (1974)

[7] Eckmann, B.: Topology, Algebra, Analysis- Relations and Missing Links. Notices of the American Mathematical Society 5(46), 520-527 (1999)

[8] Citko, W., Jakóbczak, D., Sieńko, W.: On Hurwitz - Radon Matrices Based Signal Processing. Workshop Signal Processing at Poznan University of Technology (2005)

[9] Tarokh, V., Jafarkhani, H., Calderbank, R.: Space-Time Block Codes from Orthogonal Designs. IEEE Transactions on Information Theory 5(45), 1456-1467 (1999)

[10] Sieńko, W., Citko, W., Wilamowski, B.: Hamiltonian Neural Nets as a Universal Signal Processor. 28th Annual Conference of the IEEE Industrial Electronics Society IECON (2002)

[11] Sieńko, W., Citko, W.: Hamiltonian Neural Net Based Signal Processing. The International Conference on Signal and Electronic System ICSES (2002)

[12] Jakóbczak, D.: 2D and 3D Image Modeling Using Hurwitz-Radon Matrices. Polish Journal of Environmental Studies 4A(16), 104-107 (2007)

[13] Jakóbczak, D.: Shape Representation and Shape Coefficients via Method of Hurwitz-Radon Matrices. Lecture Notes in Computer Science 6374 (Computer Vision and Graphics:

Proc. ICCVG 2010, Part I), Springer-Verlag Berlin Heidelberg, 411-419 (2010)

[14] Jakóbczak, D.: Curve Interpolation Using Hurwitz-Radon Matrices. Polish Journal of Environmental Studies 3B(18), 126-130 (2009)

[15] Jakóbczak, D.: Application of Hurwitz-Radon Matrices in Shape Representation. In: Banaszak, Z., Świć, A. (eds.) Applied Computer Science: Modelling of Production Processes 1(6), pp. 63-74. Lublin University of Technology Press, Lublin (2010)

[16] Jakóbczak, D.: Object Modeling Using Method of Hurwitz-Radon Matrices of Rank k. In: Wolski, W., Borawski, M. (eds.) Computer Graphics: Selected Issues, pp. 79-90. University of Szczecin Press, Szczecin (2010)

[17] Jakóbczak, D.: Implementation of Hurwitz-Radon Matrices in Shape Representation. In: Choraś, R.S. (ed.) Advances in Intelligent and Soft Computing 84, Image Processing and Communications: Challenges 2, pp. 39-50. Springer-Verlag, Berlin Heidelberg (2010)

[18] Jakóbczak, D.: Object Recognition via Contour Points Reconstruction Using Hurwitz-Radon Matrices. In: Józefczyk, J., Orski, D. (eds.) Knowledge-Based Intelligent System Advancements: Systemic and Cybernetic Approaches, pp. 87-107. IGI Global, Hershey PA, USA (2011)

[19] Jakóbczak, D.: Curve Parameterization and Curvature via Method of Hurwitz-Radon Matrices. Image Processing & Communications- An International Journal 1-2(16), 49-56 (2011)

Author Dariusz Jacek Jakóbczak was born in Koszalin, Poland, on December 30, 1965. He graduated in mathematics (numerical methods and programming) from the University of Gdansk, Poland in 1990. He received the Ph.D. degree in 2007 in computer science from the Polish – Japanese Institute of Information Technology, Warsaw, Poland. From 1991 to 1994 he was a civilian programmer in the High Military School in Koszalin. He was a teacher of mathematics and computer science in the Private Economic School in Koszalin from 1995 to 1999. Since March 1998 he has worked in the Department of Electronics and Computer Science, Technical University of Koszalin, Poland and since October 2007 he has been an Assistant Professor in the Chair of Computer Science and Management in this department. His research interests connect mathematics with computer science and include computer vision, shape representation, curve interpolation, contour reconstruction and geometric modeling, probabilistic methods and discrete mathematics.