

POLITECHNIKA ŚLĄSKA
ZESZYTY NAUKOWE
Nr 1615

GEOMETRIA I GRAFIKA INŻYNIERSKA

Zeszyt szósty

Gliwice

2004

Kolegium redakcyjne

REDAKTOR NACZELNY — Prof. dr hab. inż. Andrzej BUCHACZ
REDAKTOR DZIAŁU — Dr inż. Janina GŁOMB
SEKRETARZ REDAKCJI — Mgr Elżbieta LEŚKO

Redakcja

Mgr Aleksandra KŁOBUSZOWSKA

Redakcja techniczna

Alicja NOWACKA

Wydano za zgodą

Rektora Politechniki Śląskiej

PL ISSN 1427-9274

WYDAWNICTWO POLITECHNIKI ŚLĄSKIEJ
ul. Akademicka 5, 44-100 Gliwice, tel. (0-32) 237-13-81
<http://loki.polsl.pl/wydawnictwo>

Sprzedaż i Marketing
tel. (0-32) 237-18-48

Nakł. 100+40+10	Ark. wyd. 5	Ark. druk. 5	Papier offset. 70x100, 80 g
Oddano do druku 29.03.2004 r.		Podpis. do druku 29.03.2004 r.	Druk ukończ. w kwietniu 2004 r.

Druk wykonano w Zakładzie Graficznym Politechniki Śląskiej w Gliwicach, ul. Kujawska 1
zam. 148/04

Karel UHLIR*, Vaclav SKALA

Department of Computer Science and Engineering
University of West Bohemia in Pilsen

A SURVEY OF METHOD FOR IMPLICITIZATION OF POLYGONAL MODEL

Summary. This paper describes a method for the representation of the polygonal model or model defined by the point-cloud data with the implicit function. The basic aspects of the method based on representation of the model with a Radial Basis Functions (RBFs) are analyzed.

BADANIE MODELU WIELOKĄTNEGO METODĄ UWIKŁANIA

Streszczenie. Artykuł ten opisuje metodę reprezentacji modelu wielokątnego lub modelu zdefiniowanego za pomocą danych w postaci chmury punktów z funkcją uwikłaną. Przeanalizowane zostały podstawowe aspekty metody oparte na przedstawieniu modelu z funkcjami podstawy promienia (RBFs).

1. Introduction

It is well known that a perfect description of an object is if the object is defined by an implicit equation ($f(x) = 0$). The object can be directly visualized from the implicit form. Visualizing implicit surfaces typically consists of finding the zero-set of f , which may be performed either by polygonizing the surface [1, 2, 4] or by direct ray tracing [3]. There is a lot of techniques for visualization and rendering of the implicit surfaces defined by a function $f(x) = 0$ [6]. The implicitly defined object can be evaluated at any point in order to produce a mesh with required resolution. The object can be also stored in its implicit form for later use as well. A lot of objects,

* This work was supported by Grant No.: MSM 235200005

especially the basic primitives for the CSG (Constructive Solid Geometry) modeling [5], are described by the implicit equation. So it is good to have object defined by the implicit equation.

In case of the object model with no implicit description, one may use two various methods to create implicit representations. The first approach uses parametric expression of a primitive or a patch on the beginning as an input. The implicit representation of the object is generated by using symbolic operations for the parametric expression of object. The group of these methods is called *Variables elimination methods* [7]. In the other method we start with polygonal mesh or point-cloud data. Iso value, which provides information about particular point position, can be directly calculated from this representation. Then iso value means, whether the point is inside, outside or on the surface. This method is based on *Variational implicit surfaces*, for the implicit representation of the object surface. The Radial Basis Function (RBF) method, which is based on Variational implicit surfaces, will be elaborated below.

1.1. Implicit surfaces

The surface representation problem can be expressed as

Problem:

Given n distinct points x_1, x_2, \dots, x_n on a surface S in \mathbb{R}^3 , find a surface S' that is a reasonable approximation to S .

Our approach is to model the surface implicitly with a function $f(x, y, z)$. If a surface S consists of all the points (x, y, z) that satisfy the equation

$$f(x, y, z) = 0 \tag{1}$$

then we say that f implicitly defines S . Note, that the object can be defined like point-cloud data or triangular mesh. In case the object is defined as a triangular mesh it is helpful to define a set of equation. This case will be discussed later.

The shape transformation refers to the problem of *scattered data interpolation*. The problem of scattered interpolation is to create a smooth function that passes through a given set of data points [8].

2. Radial Basis Function interpolation

Having defined the problem to be solved, let us now describe the solution of a variational implicit functions. Scattered data interpolation can be achieved using *radial basis functions* centered at the constraints. Radial basis functions are circularly symmetric functions centered at a particular point. Radial basis functions may be used to interpolate a function with n points by using n radial basis functions centered at these points. The resulting interpolated function thus becomes

$$f(x) = \sum_{j=1}^n \lambda_j \phi(\|x - c_j\|). \quad (2)$$

where c_j are locations of constrains, λ_j are the *weights* and $\phi(r)$ is a *radial basis function* where $\phi(0) = 0$. It can be evaluated in radial r . Defined by the difference of the point in which we want to evaluate this function and the constraints.

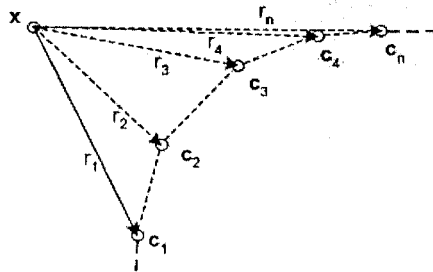


Fig. 1. Solution of equation (2) for an arbitrary point x
 Rys.1. Rozwiązanie równania (2) dla dowolnego punktu x

In order to find a set of λ_j that will satisfy the interpolation constraints $h_i = f(c_i)$, we can define the right side of the equation (2) for $f(c_i)$, which gives:

$$f(\mathbf{c}_i) = \sum_{j=1}^n \lambda_j \phi(\|\mathbf{c}_i - \mathbf{c}_j\|) = h_i \quad (3)$$

For the given constrains \mathbf{c}_i we can rewrite the equation (2) to the form $h_i = f(\mathbf{c}_i)$ and the

Fig. 1 shows how the scalar value for an arbitrary point x is evaluated.

Since this equation is linear with respect to the unknowns λ_j , it can be formulated as a system of linear equation. Let us define $\mathbf{c}_i = [c_i^x, c_i^y, c_i^z]^T$ and $\phi_j = \phi(\mathbf{c}_i - \mathbf{c}_j)$ for interpolation in 3-d space. Then this linear system can be written as follows:

$$\begin{bmatrix} \phi_{11} & \phi_{12} & \cdots & \phi_{1n} \\ \phi_{21} & \phi_{22} & \cdots & \phi_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \phi_{n1} & \phi_{n2} & \cdots & \phi_{nn} \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_n \end{bmatrix} = \begin{bmatrix} h_1 \\ h_2 \\ \vdots \\ h_n \end{bmatrix} \quad (4)$$

In some cases (including the thin-plate spline solution), it is necessary to add a first-degree polynomial $P(\mathbf{x}) = a \cdot c^x + b \cdot c^y + c \cdot c^z + d$ to account for linear and constant portion of f and ensure positive-determination of the solution. Then equation (2) is modified to equation (5).

$$f(\mathbf{x}) = \sum_{j=1}^n \lambda_j \phi(\|\mathbf{x} - \mathbf{c}_j\|) + P(\mathbf{x}) \quad (5)$$

If a polynomial is included, Eq. (4) similarly becomes [11]

$$\begin{bmatrix} \phi_{11} & \phi_{12} & \dots & \phi_{1n} & c_1^x & c_1^y & c_1^z & 1 & \lambda_1 \\ \phi_{21} & \phi_{22} & \dots & \phi_{2n} & c_2^x & c_2^y & c_2^z & 1 & \lambda_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \phi_{n1} & \phi_{n2} & \dots & \phi_{nn} & c_n^x & c_n^y & c_n^z & 1 & \lambda_n \\ c_1^x & c_2^x & \dots & c_n^x & 0 & 0 & 0 & 0 & a \\ c_1^y & c_2^y & \dots & c_n^y & 0 & 0 & 0 & 0 & b \\ c_1^z & c_2^z & \dots & c_n^z & 0 & 0 & 0 & 0 & c \\ 1 & 1 & \dots & 1 & 0 & 0 & 0 & 0 & d \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_n \\ 0 \\ 0 \\ 0 \\ 0 \\ d \end{bmatrix} = \begin{bmatrix} h_1 \\ h_2 \\ \vdots \\ h_n \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (6)$$

If we denote

$$A_{i,j} = \phi(\|c_i - c_j\|), \quad i, j = 1, \dots, n$$

$$p_i = [c_i^x, c_i^y, c_i^z, 1]^T, \quad i = 1, \dots, n$$

$$a = [a, b, c, d]^T \quad (7)$$

$$\lambda = [\lambda_1, \lambda_2, \dots, \lambda_n]^T,$$

$$P = [p_1 | p_2 | \dots | p_n],$$

then we can rewrite the equation system (6) to form

$$B \begin{bmatrix} \lambda \\ a \end{bmatrix} = \begin{bmatrix} h \\ 0 \end{bmatrix}, \quad \text{where } B = \begin{bmatrix} A & P \\ P^T & 0 \end{bmatrix} \quad (8)$$

The linear equation system (6) must be solved. Then the function $f(x)$ can be evaluated for an arbitrary point. Discussion on the system solution and the variables' determination will be provided in the following paragraphs.

3. Solution

If basis function ϕ is known then we know the values of all elements of the matrix B ($n \times n$). It is obvious, that condition $h_i = 0$ is satisfied for all on-surface points thus h is a zero vector. So it can be easily seen, that it is possible to create the

homogeneous linear equation system, from which it is possible to calculate values of vectors λ and \mathbf{a} .

This system consists of n homogeneous linear equations and has always zero solution $\mathbf{0} = [0, 0, \dots, 0]^T$. If the matrix of the homogeneous linear system has rank k , then the system has $(n-k)$ linear independent solutions and each solution of this system is a linear combination of these $(n-k)$ solutions. Especially if $k = n$, the system has only zero solution $\mathbf{0} = [0, 0, \dots, 0]^T$. If $m = n$ (number of rows is equal to number of columns) then the system has nonzero solution if and only if the determinant of the system is equal to zero. The system matrix \mathbf{B} has at the main diagonal zero elements and non-diagonal are non-zero. It can be verified that the rank of our system matrix \mathbf{B} is equal to n and its determinant is nonzero.

Now we can say that our system has only one solution, zero (trivial) solution. We need a non-trivial solution that approximate surface represented by set the given points. Therefore it is necessary to add a perturbation [9, 10] to input data set to get non zero solution. This perturbation adds new equations to the system, so the linear system is no longer homogeneous.

An obvious choice for perturbation representation is a *signed-distance function*. The signed-distance function defines points in a distance d from on-surface points. Points outside the object are assigned positive values, while points inside are assigned negative values. Similar to Turk & O'Brien [10], these off-surface points are generated by projecting along surface normals.

It should be noted that LU factorization [8, 10, 11] is the most commonly used method for solving the linear system defined by the equation (6) and (4). Methods like Cholesky factorization or GMRES iterative method can be also used.

4. Results

Implementation of this method is based on the complex description of the RBF method presented above. Because the RBF method is very computationally expensive, therefore the method was tested for few points only.

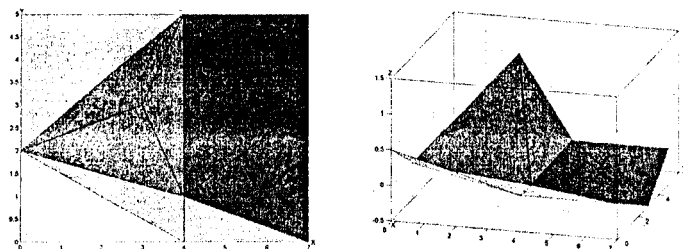


Fig. 2. The surface used for testing
Rys. 2. Powierzchnia używana do testowania

The first experiment with the RBF method was made on a surface defined by a few points in 3-d. Figure 2 shows the definition of on-surface points. Additional points were assigned with different values in order to solve the linear system of equations. Value -1 was used for points lying under the surface and, respectively, value 1 was used for points lying above the surface. All additional points have the same distance from on-surface points and are positioned in the direction of z -axis. The system of linear equations has been solved, while new points and the vector $[\lambda | \mathbf{a}]^T$ were determined.

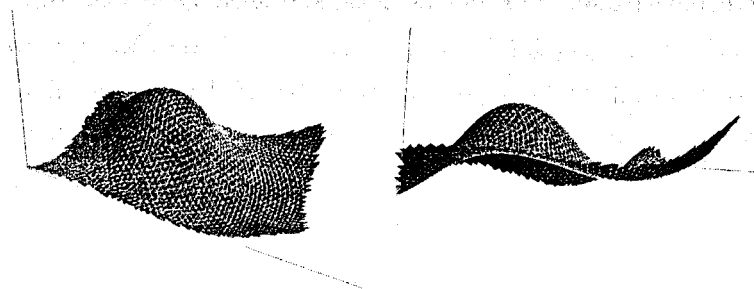


Fig. 3. The surface visualized with the marching triangle method
Rys. 3. Powierzchnia pokazana za pomocą metody ruchomego trójkąta

Now we can use any method for the iso-surface visualization (Marching cubes, Marching triangle etc.). Figure 3. presents visualization of the surface defined above on Figure 2 using matching triangle method [2]. The function $\phi(r) = r^3$ was used for this experiment.

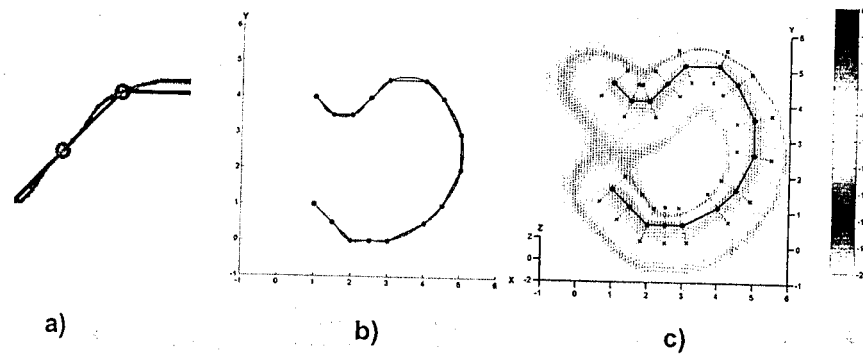


Fig. 4. Final curve in 2-d (center, left) and visualization of scalar values (Eq. (3)) on grid (right)

Rys. 4. Ostateczna krzywa w 2D (środek, lewa strona) oraz przedstawienie wartości skalar-nych (z równania (3)) na siatce (prawa strona)

In the second experiment the RBF method application for the 2-d polyline was tested. A polyline was defined and how the RBF method approximates of this line was tested. In this case the basic function $\phi(r) = r^2 \log r$ was used.

The final smooth curve is presented at Figure 4.b. Figure 4.c presents how perturbation points were defined, how to find smooth curve, original polyline and a curve defined as a contour for the non-zero threshold.

5. Summary and future research directions

A complex method for implicitization of the object defined by the triangular mesh or point-cloud data was introduced, and fundamental properties were presented. Final implicit description and final quality of visualized object depend on basic function selection and its radius of support. This parameter will be analyzed and described in further work.

Implementation of the RBF method for the cases of 2-d polyline and 3-d surface was described in the paper.

BIBLIOGRAPHY

1. Bloomenthal, J.: Polygonizaion of Implicit Surface, Computer-Aided Geometric Design, vol. 5, no. 4, pp. 341-355, 1988.

2. Čermák M., Skala V. (2002).: Polygonization by the Edge Spinning. Algoritmy 2002 Conf.proceedings, Univ.of Technology, Slovakia, pp.245-252, 2002.
3. Hart, J.C.: Sphere tracing: A geometric method for the antialiased ray tracing of implicit surfaces, The Visual Computer 12 (10), Dec. 1996, pp. 527-545.
4. Hartmann, E.: A marching method for the triangulation of surfaces, The Visual Computer 14, 3, 95-108, 1998.
5. Uhlir, K., Skala,V.: The Implicit Function Modelling System - Comparison of C++ and C# Solutions, C# and .NET Technologies'2003, University of West Bohemia, Czech Republic 2003.
6. Bloomenthal, J., Bajaj, C., Blinn, J., Cini-Gascuel, M. P., Rockwood, A., Wyvill, B., Wyvill, G.: Introduction to Implicit Surfaces, Morgan Kaufmann, 1997.
7. Hoffmann, C.M.: Implicit Curves and Surfaces in CAGD, IEEE Computer Graphics and Appl. 13, Jan. 1993, 79-88.
8. Turk, G., O'Brien, J.: Shape Transformation Using Variational Implicit Functions, SIGGRAPH 99, August 1999, pp. 335-342, 1999.
9. Carr, J. C., Beatson, R. K., Cherrie, J. B., Mitchell, T. J., Fright, W. R., McCallum, B. C., Evans, T. R.: Reconstruction and Representation of 3D Objects with Radial Basis Functions, Computer Graphics (SIGGRAPH 2001 proceedings), pp. 67-76, August 2001.
10. Turk, G., O'Brien, J.F.: Modelling with Implicit Surfaces that Interpolate, ACM Transactions on Graphics, Vol. 21, No. 4, pp. 855-873, October 2002.
11. Morse, B., Yoo, T. S., Rheingans, P., Chen, D. T., Subramanian, K.R.: Interpolating Implicit Surfaces from Scattered Surface Data Using Compactly Supported Radial Basis Functions, in Proceedings of the Shape Modeling conference, Genova, Italy, 89-98, May 2001.

Recenzent: Dr inż. Renata A. GÓRSKA

Abstract

This paper describes a method for the representation of the polygonal model or model defined by the point-cloud data with an implicit function. The basic aspects of the method based on representation of the model with a Radial Basis Functions

(RBFs) are analyzed. This method is very complex and can be used also for other purposes than the representation of the polygonal model. The radial basis function method can help the reconstruction of the gaps or the detection of sharp features. The solution of this problem requires large memory and it is computationally very complex. An implementation of the method proved expected properties of the RBF, too.