

## MỘT LÝ THUYẾT MÔ PHỎNG HÌNH HỌC CÁC ĐỐI TƯỢNG TRONG KHÔNG GIAN TRÊN CƠ SỞ NỘI SUY

Hoàng Thái Hồ\*, Yu. N. Kosnikov\*\*

### TÓM TẮT

Bài báo phân tích và trình bày ứng dụng của nội suy hàm cơ sở xuyên tâm (radial basic function (RBF)), làm cơ sở lý thuyết chuyển đổi sang hàm trộn của cơ sở trực giao (blending functions of orthogonal basis (BFOB)). Từ đó đề xuất lý thuyết toán học mô phỏng các đối tượng trong không gian, bao gồm các bước sau: áp dụng RBF xây dựng các mô hình bề mặt trên tập hợp các điểm quy chiếu cho trước; thay thế tập hợp điểm quy chiếu cho trước bằng tập hợp điểm quy chiếu mới; phân bố đều các điểm trên bề mặt; xây dựng mô hình toán học của bề mặt trên cơ sở BFOB; áp dụng phương pháp sai phân hữu hạn trong quá trình tính tọa độ các điểm trung gian.

Từ khóa: Nội suy, hàm trộn, cơ sở xuyên tâm, cơ sở trực giao, sai phân hữu hạn.

### ABSTRACT

The article presents and analyzes applications of radial basis function (RBF), the theory of the transition to blending functions of orthogonal basis (BFOB). Recommended a simulating mathematical theory of objects of the virtual environment interface, it includes the following steps: construction of surface patterns according to the set of points of reference given to the application of RBF, replacement set where the reference point given by the new set of reference points, evenly distributed over the surface, construction of mathematical models on the basis BFOB surface, applying finite difference method for computing of the coordinate of intermediate points.

Keywords: interpolation, blending function, radial basis, orthogonal basis, finite-difference.

### 1. Đặt vấn đề

Các hệ thống xử lý và biểu thị thông tin ngày nay thường có độ phức tạp cao. Trong

những trường hợp con người phải hoạt động trong trạng thái thời gian thực (real time RT). Họ phải chịu rất nhiều những áp lực, điều đó sẽ làm giảm hiệu quả công việc. Thông thường con người giao tiếp với các phần tử của hệ thống thông qua một giao diện. Để tăng tính hiện thực cho đối tượng trong giao diện đó thì người ta ngày càng mở rộng việc áp dụng các nguyên tắc của thực tế ảo (Virtual reality VR). Con người sẽ hiểu rõ hơn các tình huống trong

\* Nghiên cứu sinh, Khoa Tin học và Hệ thống tính toán trường Đại học Tổng hợp Quốc gia Penza, thành phố Penza, Liên Bang Nga

\*\* Giáo sư - Tiến sĩ khoa học, Trường khoa Tin học và Hệ thống tính toán trường Đại học Tổng hợp Quốc gia Penza, thành phố Penza, Liên Bang Nga

hệ thống ảo nếu các phần tử của nó đã từng được gặp trong thế giới thực. Khi đó giao diện môi trường ảo (virtual environment interface VEI) trở thành một môi trường giao tiếp hiệu quả giữa con người với các phần tử trong hệ thống mà những phần tử này được tạo nên bởi công nghệ mô phỏng và đồ họa máy tính. Tuy vậy quá trình thiết lập VEI ta cần xét đến tính đặc trưng của từng bài toán cụ thể cũng như các đặc điểm của đối tượng trong giao diện đó.

**2. Phương pháp nội suy R-PRIMITIVE**

Để mô phỏng các đối tượng trong không gian trên dựa trên một tập hợp các điểm quy chiếu cho trước ta thường sử dụng phương pháp nội suy. Tuy vậy để mô tả được chính xác dạng hình học của đối tượng thì các phương pháp nội suy cổ điển như Newton, Lagrange không còn phù hợp. Ngay cả các phương pháp nội suy spline (B-spline, Spline Bezier) cũng có những hạn chế nhất định [1, 3].

So với spline thì R-primitive là một phương pháp đơn giản hơn. R-primitive được viết dưới dạng:

$$z = \sum_{i=1}^N z_i bf_{xi} bf_{yi} \quad (1)$$

với  $i$  - Số thứ tự của điểm quy chiếu;

$z_i$  - chiều cao của điểm quy chiếu thứ  $i$ ;

$bf_{xi}, bf_{yi}$  - hàm trộn tại điểm quy chiếu thứ  $i$ ;

$N$  - Số lượng điểm quy chiếu (chúng có ảnh hưởng tới việc xây dựng các phần bề mặt).

Tọa độ  $z$  của điểm hiện tại được tính bằng tổng hệ số cao của các điểm quy chiếu, với hệ số cao chính là giá trị của hàm trộn. Những giá trị này tỉ lệ thuận với khoảng cách giữa các điểm hiện tại và điểm quy chiếu. Hàm

trộn R-primitive tại điểm quy chiếu thứ  $i$  được viết như sau:

$$bf_{xi} = \begin{cases} \left(1 - \frac{|x - x_i|}{A_x}\right)^{mx} & , |x - x_i| < A_x \\ 0 & , |x - x_i| \geq A_x \end{cases} \quad (2)$$

$$bf_{yi} = \begin{cases} \left(1 - \frac{|y - y_i|}{A_y}\right)^{my} & , |y - y_i| < A_y \\ 0 & , |y - y_i| \geq A_y \end{cases} \quad (3)$$

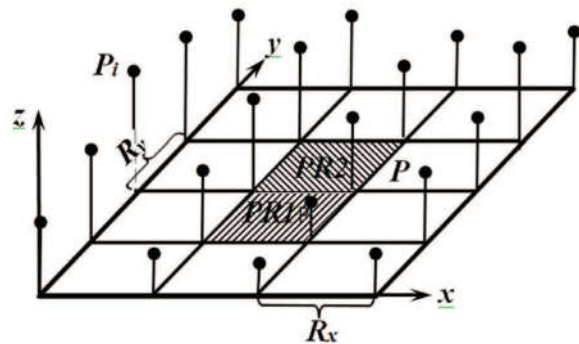
với:  $x_i, y_i$  - tọa độ điểm quy chiếu thứ  $i$  trên mặt phẳng đối số;

$x, y$  - tọa độ điểm hiện tại trên mặt phẳng đối số;

$mx, my$  - hệ số lồi lõm của bề mặt;

$A_x, A_y$  - Kích thước vùng ảnh hưởng của điểm quy chiếu thứ  $i$  dọc theo nửa trục tọa (vùng âm và vùng dương).

Độ mịn tại giao của các lưới có thể tăng lên bằng cách mở rộng vùng ảnh hưởng của các điểm quy chiếu. Trên hình 1 biểu thị 20 điểm quy chiếu trong không gian decart từ  $P_i$  ( $i = 1, \dots, 20$ ).



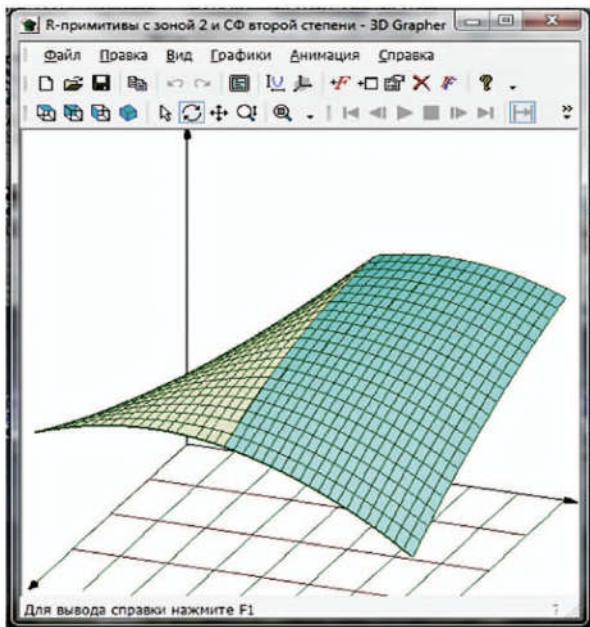
Hình 1- Hai lưới R-primitive từ tập hợp 20 điểm quy chiếu

Vùng ảnh hưởng của điểm quy chiếu được giả định là hai bước của lưới tọa độ, trên hình 1 các bước của lưới tọa độ đó được ký hiệu là  $R_x, R_y$ . Hình chiếu của hai mặt lưới kế tiếp được thể hiện bằng phần gạch chéo (PR1, PR2). Với hệ số lỗi lờm bằng 2. Khi đó để tính độ cao của điểm hiện tại đầu tiên ( $z_I$ ) và thứ hai ( $z_{II}$ ) của lưới ta sử dụng phương trình sau:

$$z_I = \sum_{i=1}^{16} z_i \left(1 - \frac{|x-x_i|}{2R_x}\right)^2 \left(1 - \frac{|y-y_i|}{2R_y}\right)^2 \quad (4)$$

$$z_{II} = \sum_{i=5}^{20} z_i \left(1 - \frac{|x-x_i|}{2R_x}\right)^2 \left(1 - \frac{|y-y_i|}{2R_y}\right)^2 \quad (5)$$

Hình 2 thể hiện mô hình R-primitive dựa trên cơ sở phương trình (4), (5) xây dựng trên môi trường 3D Grapher. Nó thể hiện rõ độ mịn giữa giao của các phần lưới.



Hình 2 - Sự kết hợp của hai mặt lưới trong mô hình R-primitive

Tuy vậy trong trường hợp điểm hiện tại trùng với một số điểm quy chiếu  $k$  ( $x=x_k, y=y_k$ ) thì hạng tử tương ứng sẽ nhận giá trị  $z_k$  của

điểm quy chiếu thứ  $k$ . Nghĩa là một số hạng tử nhận các giá trị khác không, còn một số thì nhận các giá trị được tính bởi tổng các  $z_k$ . Tóm lại chiều cao tính được của điểm hiện tại không trùng với chiều cao của điểm quy chiếu. Điều đó dẫn tới bề mặt thu được không đi qua các điểm cho trước.

**Phương pháp khắc phục:**

Chúng ta cần phải thêm vào các hệ số tỉ lệ dành riêng cho mỗi điểm quy chiếu. Khi đó mô hình lưới R-primitive có dạng:

$$z = \sum_{i=1}^N m_i z_i b f_{x_i} b f_{y_i},$$

với:  $b f_{x_i}, b f_{y_i}$  - giá trị hàm trộn tại điểm quy chiếu thứ  $i$  dọc trên trục tọa độ;

$N$  - số lượng điểm quy chiếu ảnh hưởng tới điểm hiện tại của lưới.

Đặt  $\lambda_i = m_i z_i$  và định nghĩa  $\lambda_i$  như là hệ số ảnh hưởng của điểm quy chiếu thứ  $i$ , ta thu được phương trình:

$$z = \sum_{i=1}^N \lambda_i b f_{x_i} b f_{y_i} \quad (6)$$

Phương trình này được xem tương tự như phương trình nội suy trên hàm cơ sở xuyên tâm mà chúng ta sẽ xem ở phần tiếp theo.

**3. Nội suy hàm cơ sở xuyên tâm và cơ sở trực giao**

Ta thường sử dụng phương pháp nội suy trên cơ sở RBF để phục hồi những phần khuyết của bề mặt đối tượng. Phương trình nội suy tổng quát có dạng:

$$F(x, y, z) = \sum_{i=1}^N \lambda_i \phi(r_i) = 0, \quad (7)$$

với:  $\phi(r_i)$  - giá trị RBF đối với điểm quy chiếu thứ  $i$ ;

$\lambda_i$  - hệ số ảnh hưởng của điểm quy chiếu thứ  $i$ .

Trên cơ sở (7) từ điều kiện mặt nội suy đi qua các điểm quy chiếu ta thành lập được một hệ phương trình. Để loại trừ nghiệm tầm thường (nghiệm 0) ta cần thêm vào (7) một đa thức bậc thấp [4]. Nghiệm của hệ chính là hệ số ảnh hưởng  $\lambda_i$  của các điểm quy chiếu. Thay  $\lambda_i$  vào (7) ta thu được phương trình mô tả dạng phân tích của bề mặt. Để mô hình hóa bề mặt từ (7) ta cần tìm các điểm trung gian chính là các đỉnh của lưới đa giác. Tiếp theo tại mỗi đỉnh đó ta cần tìm pháp tuyến tới bề mặt để sử dụng trong mô phỏng ánh sáng tới đối tượng. Tọa độ các đỉnh và các vector pháp tuyến sẽ được đưa vào quá trình xử lý đồ họa của máy tính để mô phỏng bề mặt đối tượng.

Thuật toán này khó phù hợp trong mô phỏng và xử lý các đối tượng động trong trạng thái RT. Đối với bề mặt rộng thì số lượng các hạng tử trong (7) có thể đạt đến vài trăm, còn với các dạng đã biết của RBF (thin-plate spline, Gaussian, ...) thì khá phức tạp vì chúng còn có chứa cả hàm mũ, hàm logarit, hàm căn, hàm thập phân. Quá trình tính toán sẽ tốn rất nhiều thời gian và không thể thực hiện trong trạng thái RT.

### Phương pháp khắc phục

Chúng ta có thể sử dụng hàm trộn trên cơ sở trực giao (BFOB) thay cho RBF trong mô tả toán học của đối tượng và quay lại dạng phương trình (6). BFOB có ưu điểm là giá trị của hàm phụ thuộc vào khoảng cách giữa điểm hiện tại với các điểm quy chiếu của bề mặt và chúng được tính riêng theo tọa độ của đối số. Ví dụ trong không gian decart chúng được tính theo tọa độ  $x$  và  $y$  như ở phương trình (2), (3). Nếu các điểm quy chiếu là các đỉnh của lưới tọa độ trực giao thì tọa độ các điểm trung gian được tính chỉ với những phép tính cộng [5]. Ví dụ BF  $bf_x$  được tính theo phương trình:

$$\Delta^k bf_{x(j+1)} = \Delta^k bf_{xj} + \Delta^{k+1} bf_{xj}, k = 0, 1, \dots, \quad (8)$$

với:  $\Delta^k bf_{xj}$  - sai phân hữu hạn của hàm  $bf_x$  bậc  $k$  tại bước tính thứ  $j$ , với  $\Delta^0 bf_x \equiv bf_x$ .

Áp dụng BFOB và công thức sai phân hữu hạn giúp chúng ta đưa được bề mặt đa giác vào trạng thái RT.

## 4. Lý thuyết đề xuất cho mô phỏng toán học các đối tượng không gian

**Bước 1.** Áp dụng RBF để mô tả bề mặt đối tượng.

Bước này sẽ đơn giản hơn nếu hàm nội suy là duy nhất. Khi đó chúng ta có thể viết nó dưới dạng tường minh và tham số. Để làm được điều đó, tập hợp các điểm quy chiếu cho trước được chuyển vào một hệ tọa độ phụ thích hợp (Auxiliary coordinate system ACS) bằng các phép tịnh tiến và quay hình học [8]. Và công thức của chúng có dạng:

$$z(x, y) = \sum_{i=1}^M \lambda_i \phi(r_i), \quad (9)$$

với:  $r_i$  - khoảng cách decart giữa hình chiếu của điểm quy chiếu thứ  $i$  tới điểm hiện tại trên mặt phẳng đối số;  $r_i = \sqrt{(x - x_i)^2 + (y - y_i)^2}$

$M$  - số lượng điểm quy chiếu.

Các hệ số ảnh hưởng  $\lambda_i (i=1, \dots, M)$  được xác định từ điều kiện bề mặt phải đi qua các điểm quy chiếu đó (giải hệ phương trình thu được từ 7).

**Bước 2.** Thay thế tập hợp điểm quy chiếu bằng một tập hợp điểm mới.

Áp dụng phương trình (9) với các đối số  $(x, y)$  ta sẽ tìm ra các giá trị mới của  $z$ . Tọa độ  $x, y, z$  thu được nhận giá trị tọa độ của các điểm quy chiếu mới và sẽ thiết lập mô tả mới của bề mặt trong ACS.

**Bước 3.** Phân bố có trật tự các điểm quy chiếu mới.

Bằng cách sắp xếp các đối số của ACS, các điểm quy chiếu sẽ được sắp xếp có trật tự trong không gian.

**Bước 4.** Xây dựng mô hình toán học trên cơ sở BFOB.

Theo phương trình (6), mô hình bề mặt được thiết lập trên cơ sở BFOB. Dạng BFOB và kích thước vùng ảnh hưởng của chúng được lấy trước đó từ điều kiện sai số nội suy. Ví dụ: thực nghiệm chỉ ra rằng BFBO cho kết quả tốt có dạng:

$$bf_{xi} = \begin{cases} 1 - \left( \frac{x - x_i}{A_x} \right)^2, & |x - x_i| < A_x \\ 0, & |x - x_i| \geq A_x \end{cases} \quad (10)$$

$$bf_{yi} = \begin{cases} 1 - \left( \frac{y - y_i}{A_y} \right)^2, & |y - y_i| < A_y \\ 0, & |y - y_i| \geq A_y \end{cases} \quad (11)$$

Áp dụng những BF này giúp ta dễ dàng tính được pháp tuyến tới bề mặt. Thật vậy, trong trường hợp bề mặt viết dưới dạng tường minh thì các pháp tuyến  $n_x, n_y, n_z$  được tính bằng biểu thức sau:

$$n_x = \frac{\partial z}{\partial x}, n_y = \frac{\partial z}{\partial y}, n_z = -1$$

Khi đó từ (6) kết hợp với (10), (11) ta thu được phương trình:

$$n_x = \sum_{i=1}^N \lambda_i \overline{bf_{xi}} \overline{bf_{yi}}, n_y = \sum_{i=1}^N \lambda_i \overline{bf_{xi}} \overline{bf_{yi}}$$

Trong đó bao gồm BF (10), (11) và dạng

BF khác:  $\overline{bf_{xi}} = \frac{2(x_i - x)}{A_x^2}, \overline{bf_{yi}} = \frac{2(y_i - y)}{A_y^2}$ .

Như vậy, mô tả bề mặt và các pháp tuyến của nó có cấu trúc đồng nhất.

**Bước 5.** Vận dụng sai phân hữu hạn vào mô hình bề mặt đối tượng.

Đầu tiên chúng ta tính các giá trị sai phân hữu hạn ban đầu trong mô hình bề mặt và pháp tuyến của chúng. Sau đó trong trạng thái RT theo công thức (8) ta tính được tọa độ điểm trung gian của bề mặt và pháp tuyến bề mặt tại các điểm đó. Điểm thu được chính là các đỉnh của đa giác trong mô hình đa giác của bề mặt đối tượng.

Tập hợp các đỉnh và pháp tuyến bề mặt của các điểm trung gian cũng như các thông số vị trí của ACS trong hệ tọa độ ban đầu được đưa vào vi xử lý đồ họa để thực hiện quá trình tính toán và mô phỏng.

## 5. Kết luận

Việc mô phỏng một đối tượng dựa trên tập hợp hữu hạn các điểm quy chiếu giúp tiết kiệm được thời gian cũng như tài nguyên khi lưu trữ vào họa máy tính. Sự phân bố có trật tự các điểm quy chiếu sẽ làm đơn giản hóa thuật toán chuyển đổi tới mô hình đa giác. Và bằng việc áp dụng BFOB tạo điều kiện dễ dàng cho áp dụng thuật toán sai phân hữu hạn khi tính tọa độ các điểm trung gian và pháp tuyến của chúng. Tóm lại bằng việc áp dụng lý thuyết được đề xuất trên sẽ cung cấp cho chúng ta hiệu suất tính toán cao kết hợp với khả năng biểu thị hình ảnh tốt và tiết kiệm được tài nguyên của hệ thống đồ họa máy tính trong quá trình mô phỏng các đối tượng trong không gian. Trong thời gian tới lý thuyết này sẽ là cơ sở để xây dựng phần mềm mô phỏng hình học các đối tượng trong không gian có dạng hình học xác định hoặc không xác định.

## TÀI LIỆU THAM KHẢO

- [1] Adams J.A., Rogers D. F. *Mathematical Elements for Computer Graphics (2nd ed.)*, McGraw-Hill, Inc., NY, USA, 1990.
- [2] Aleksandrova N.V., Zimin A.P., Kosnikov Yu.N., Hoang Thai Ho, “Blending Functions in Geometrical Modeling and Visualization of Freeform Surfaces”, *XXI Century: Resumes of the Past and Challenges of the Present plus*. Scientific periodical. - Penza: PenzSTU Publishing House, 2015, № 03(25), P.51 - 60.
- [3] 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”, *SIGGRAPH '01 Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, 2001, P. 67-76.
- [4] Cuomo S., Galletti A., Giunta G., Starace A., “Surface Reconstruction from Scattered Point via RBF Interpolation on GPU”, *Proceedings of the 2013 Federated Conference on Computer Science and Information Systems*, 2013, P. 433-440.
- [5] Kosnikov Yu. N. “The Integrated Primitives Application for Mapping Three-dimensional Objects in Real-time Mode”, *Actual Problems of Aviation and Aerospace Systems*, No. 2(10), 2000, P. 37-42.
- [6] Kosnikov Yu. N., “Increasing the Information Capacity of the Interface of the Control System for Multiparametric Objects”, *9th International Conference «Application of Information and Communication Technologies - AICT2015*, Rostov-on-Don, Russia, 2015, P. 446 - 449.
- [7] Kosnikov Yu. N., “Technique and Technology of Computer Modeling of Free Forms Surfaces with Radial Basis Functions Application” *XXI Century: Resumes of the Past and Challenges of the Present plus*. *Scientific periodical*, Penza: PenzSTU Publishing House, 2014. - № 03(19). - P.177 - 184.
- [8] Morse B. S., Yoo T.S., Rheingans P., Chen D.T., Subramanian K.R., “Interpolating Implicit Surfaces From Scattered Surface Data Using Compactly Supported Radial Basis Functions”, *Proceedings of the International Conference on Shape Modeling and Applications (SMI '01)*, 2015, P. 92-102.

Ngày nhận bài: 23/2/2017

Ngày gửi phản biện: 25/2/2017