Modeling of Ship Surface with Non Uniform B-Spline

Vo Trong Cang, Tat-Hien Le , *Members IAENG*

Abstract **- In the digital design process, surface modeling is utilized in the areas of analysis, rapid prototyping and manufacturing. In recent years, the genetic algorithm (GA) has gained increasing attention as a multimodal optimization solution for efficient surface fitting. The Non Uniform B-spline surface can be readily translated into many CAD/CAM packages, which facilitate the smooth data transition across the different design.**

Index Terms **- Surface Fitting, Surface Generation, Genetic Algorithm, Multimodal Optimization**

I. Introduction

Surface modeling is the key to integration of design, analysis, manufacturing, and other calculation [3]. There are two principle categories for surface fitting techniques. The first one works with reverse engineering in two steps; first the given data points are rearranged into the rectangular mesh, then the surface is constructed by some matrices inversion and B-spline algorithm procedures. The first problem is that the data points are often scattered. The second problem comes from the matrices inversion. In fact, the matrices inversion gets the ill conditioned problem, and it must also avoid round off error magnifications in back-substitution calculation and large storage capacity. The second surface generation technique is to approximate the given data points by B-spline algorithm. Many conventional methods have been proposed [8,1]. The main problem is the parameterization of B-spline surface. It needs to be estimated from an initial unknown surface. The approximation of surface fitting using genetic algorithm is developed by Birmingham [5], Le et al [6]. However, this method gets stuck in case of complicated surfaces. The main contribution of this research is to use non uniform B-spline (NUB) surface fitting through a GA technique. This approach has more advantages with regard to surface representation, and fairness of the interior surface.

Vo Trong Cang (Dr) is with the HoChiMinh city University of Technology (HCMUT)/Vietnam National University-HCM, (VNU-HCM), Vietnam. He is now Deputy Head of the Naval Architecture and Marine Engineering.
Since 2011 Dr Cang is invited researcher at the Digital Control & Systems Engineering Key Laboratory (DCSELAB)/ VNU-HCM
(e-mail: vtcang@hcmut.edu.vn)

Le Tat-Hien (PhD) is with Naval Architecture and Marine Engineering department, Faculty of Transportation Engineering, HCMUT. (e-mail: letathien2212@yahoo.com)

II. Overview of the B-spline Fitting Algorithm

A. *Surface fitting basis*

B-spline surface data D(u,w) is given by:

$$D(u,w)=\sum_{i=1}^{n+1}\sum_{j=1}^{m+1}B_{i,j}^{h}N_{i,k}(u)M_{j,l}(w) \qquad (1)$$

Let x_i and y_j be knots value and $B_{i,j}$ is the vertex point of B-spline for data fitting, and Ni,k(u) and Mj,l(w) are the basis functions. Eq. 1 shows that shape of NUB surface will be regenerated if we change the location of vertex point and knot value, as in Fig. 1. In GA process, we changed the shape of surface automatically in this way. Therefore, the final surface can get the good solution time by time.

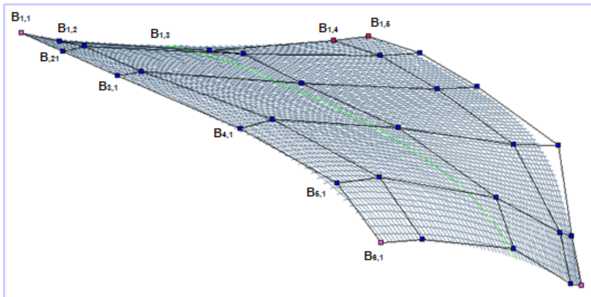

Figure 1. Rectangular vertex points

B. *Matrices Inversion Problems in B-spline Fitting*

Generally speaking, many conventional methods have been proposed in NUB surface fitting. Writing Eq. (1) for each single B-spline data point yields

$$\begin{aligned}&D_{1,1}(u_1,w_1)=\\&\quad N_{1,k}(u_1)\left[M_{1,l}(w_1)B_{1,1}+M_{2,l}(w_1)B_{1,2}+\ldots+M_{m+1}(w_1)B_{1,m+1}\right]+\\&\quad .\\&\quad .\\&\quad .\\&\quad N_{n+1,k}(u_1)\left[M_{1,l}(w_1)B_{n+1,1}+M_{2,l}(w_1)B_{n+1,2}+\ldots+M_{m+1}(w_1)B_{n+1,m+1}\right]\end{aligned} \qquad (2)$$

In the matrix form of NUB surface point of Eq. (2), moving in space with 2 degrees of freedom, u and w, is given by:

$$[D]=[C][B] \qquad (3)$$

Where [D] is an r x s x 3 matrix containing the 3D coordinates of surface data points, [C] is an r x s x n x m matrix of the products of the basis functions, and [B] is an n x m x 3 matrix of the 3D coordinates of the required vertices.

If [C] is not square, the solution is given by:

$$[B]=\left[[C]^T[C]\right]^{-1}[C]^T[D] \qquad (4)$$

ISBN: 978-988-19251-2-1
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

Nevertheless, the matrix inversion gets the ill conditioned problem when determinant close to zero. The Fig. 2 illustrates the difficulties of surface fitting in matrices inversion.

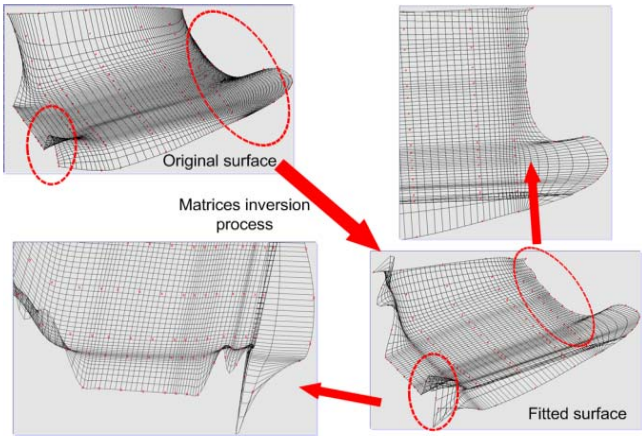

Figure 2. The difficulties of fitted surface in matrices inversion method

For curve fitting, an optimization method with GA has been presented by Yoshimoto [4]. However, the matrices inversion is not a solution for irregular given data points. Instead, moving the location of the vertex point using GA is efficient way to improve the surface quality.

III. New Approach to Surface Fitting for the Given Interior Data Point by Using GA

Vertices Encoding for Initial Population

For an initial population, usually the base surface is created by vertices. The vertices can be generated with the same x and z coordinate value of the given data points and deviation δ in y direction (see in Fig. 3).

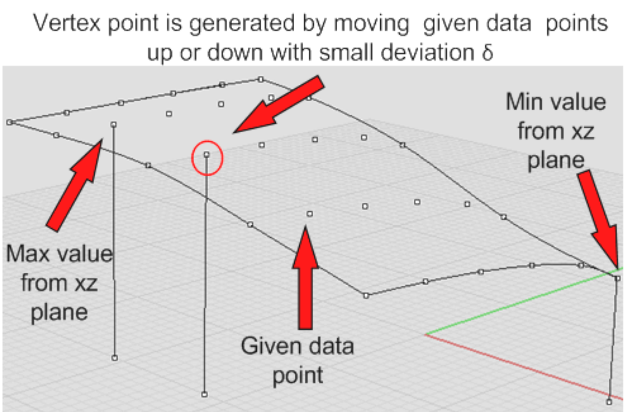

Figure 3. The generation of vertices from the given data points

Crossover Process

The idea here is to exchange a single knot value in two parents to form two new individuals. Fig. 4 shows that the individual 1 received a new sub-string from individual 2 to generate a new individual in a next population.

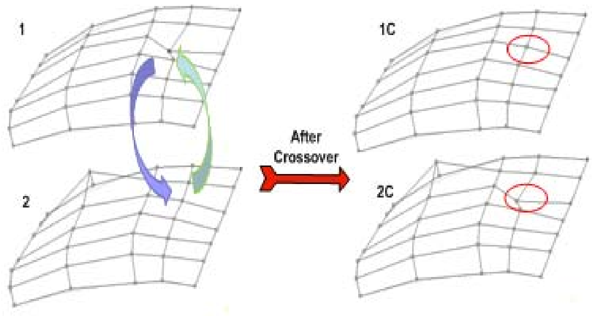

Figure 4. Crossover procedure

Mutation Process

In this method, the location of the new vertex point is created by moving in a circle of small radius δ around the old vertex point. In such an approach, δ should be recommended less than 5% of the distance between old vertex point and the nearest neighboring vertex point. The variable design has a deviation with the deviation size δ as given by Eq. (6).

$$\left[B_{new}\right]=\left[B_{old}\right]+\delta \tag{6}$$

Therefore, the surface quality improves a good result after each time. Deformation of a NUB surface can be achieved by moving the vertices that define it (see Fig. 5).

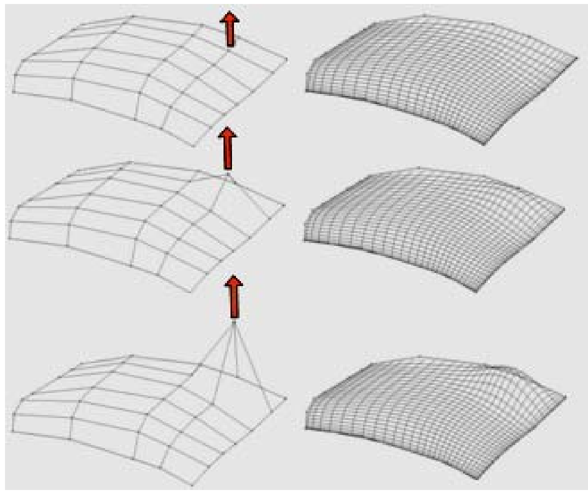

Figure 5. The effect of moving one of the vertex points of the surface

Jacobian inversion for finding nearest points

This can be solved by Jacobian iteration within a few steps [1].

- Input: a original point (G) and initial parameter (u, w) for parametric surface
- Compute the distance d = G – r(u, w)
- Compute Δu, Δw and update u = u + Δu, w = w + Δw, with

$$\Delta u=\frac{\left(r_u.d\right)\left(r_u.r_w\right)-\left(r_w.d\right)\left(r_u.r_w\right)}{\left(r_u.r_u\right)\left(r_w.r_w\right)-\left(r_u.r_w\right)^2}$$

$$\Delta w=\frac{\left(r_w.d\right)\left(r_u.r_u\right)-\left(r_u.d\right)\left(r_u.r_w\right)}{\left(r_u.r_u\right)\left(r_w.r_w\right)-\left(r_u.r_w\right)^2}$$

ISBN: 978-988-19251-2-1
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

- If ($|\Delta u| > \varepsilon$) or ($|\Delta w| > \varepsilon$) then return Step2
  Else Stop
- Final step

$$Q = \sum_{i=1,\,j=1}^{N,M} \{D(i,j) - G\}^2 \tag{7}$$

In that case, the location of vertices will be optimized to fit the given data points using GA technique.

IV. Application Examples

A. Ship Hull Surface

The Fig. 6 illustrates the given data points of surface. In this case, the surface shows total error values converged at 20,000th generation and the best fitting after 20,000 generations (see Fig. 7 & 8). The normalized error value after 20,000 generations is computed in Table 1.

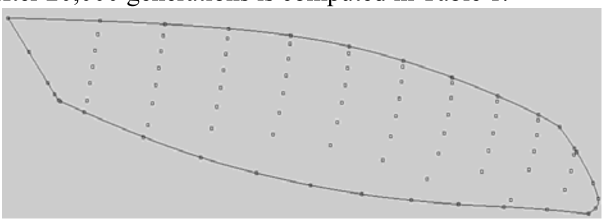

Figure 6. The given data points of yacht surface

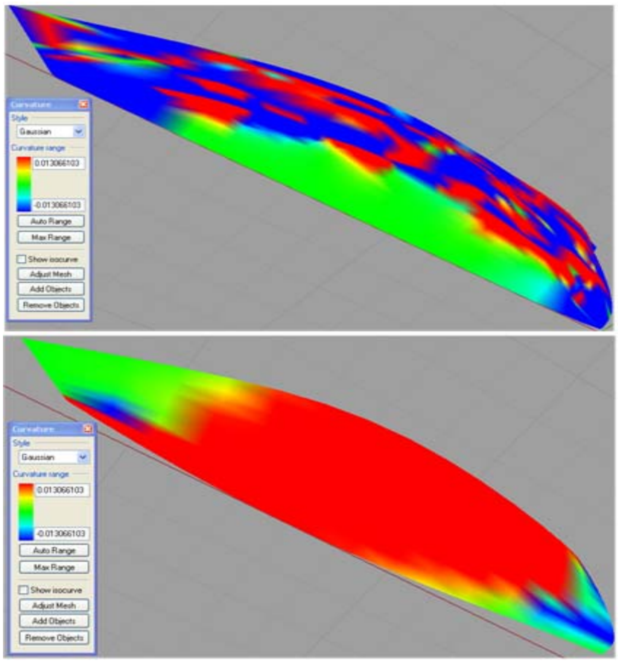

Figure 7. The Gaussian curvature of surface at 1st generation and 20,000th generation

TABLE 1
NORMALIZED ERROR BETWEEN THE GIVEN DATA POINTS AND THE SURFACE POINTS

No. of vertex points in u and w directions	Computing time	Normalized error
10×14	15min	0.0044834

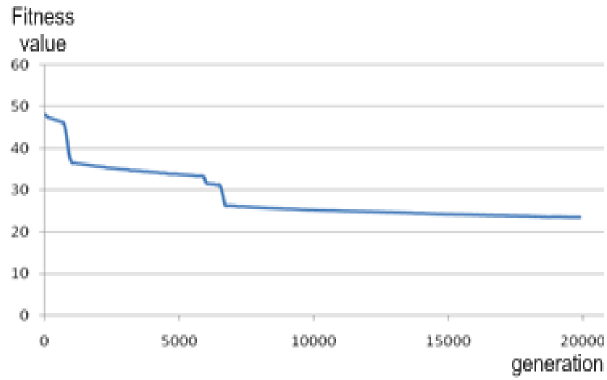

Figure 8. The fitness value during generations of complicated shape

B. Complicated Surface

The complicated mesh (Fig. 9) consists of 91 given data points. The initial result and the best result at 20,000th generation were illustrated from Fig. 10 and 11. Finally, a value of normalized error converged at the 20,000th generation (see Table 2).

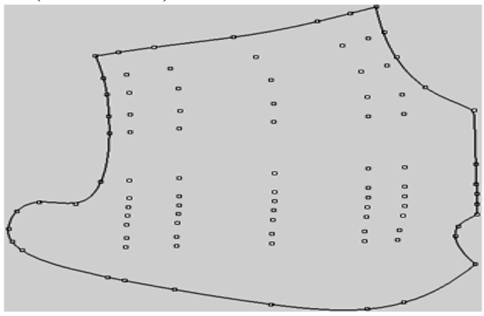

Figure 9. The given data points of complicated surface

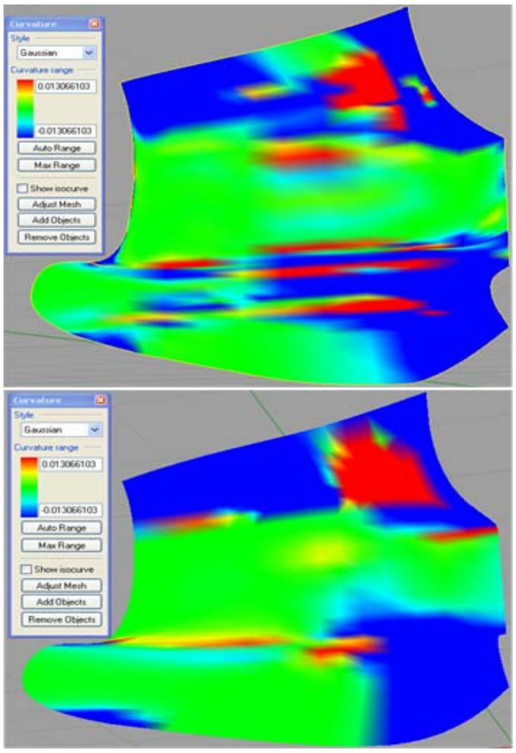

Figure 10. The Gaussian curvature of the complicated surface at 1st generation and 20,000th generation

ISBN: 978-988-19251-2-1
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

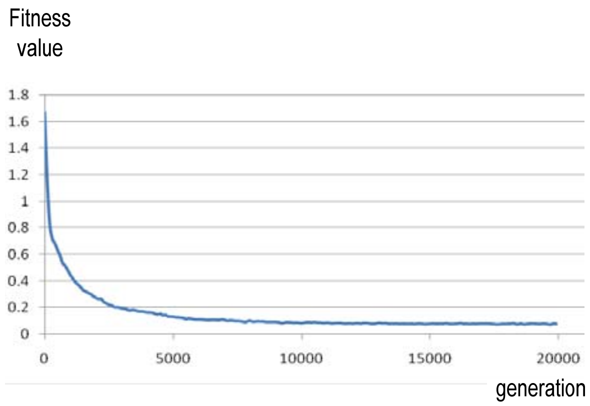

Figure 11. The fitness value during generations of complicated shape

TABLE 2
NORMALIZED ERROR BETWEEN THE GIVEN DATA POINTS AND THE COMPLICATED SURFACE POINTS

No. of vertex points in u and w directions	Computing time	Normalized error
17×7	30 min	0.001461274

V. CONCLUSIONS

In this study, a NUB surface is generated in a fully automatic approach to the location of vertex points and knot value until satisfactory precision is reached. The main contribution of this research is to construct the NUB surface by using GA implementation for the optimization. All the above techniques have been implemented in a NUB surface generation effectively.

As application examples, this technique represents the high visual quality in the case of a hull ship surface and complicated shape. The accuracy of each surface modeling is dependent on the generation time of the GA process, while the convergent solution is guaranteed by the required precision.

For future research in this study, emphasis should be on finding a faster speed of optimization for ship hull surface and the required fairness criterion for other analysis applications after achieving the fitted NUB surface.

The department of NAME of the HoChiMinh city University of Technology (HCMUT) since 2000 has studies and training courses in computer applications in ship hull design based on Maxsurf suite and other softwares [7]. The approach in this paper with regard to curve representation, boundary conditions, and fairness of the interior surface will be used as the extending for these courses.

ACKNOWLEDGEMENTS

This research was supported by the DCSELAB, National University Ho Chi Minh City (VNU-HCM), Vietnam.

REFERENCES

[1] B. K. Choi, “Surface modeling for CAD/CAM.” Elsevier Amsterdam – Oxford – New York – Tokyo Book, 1991.

[2] D. E. Goldberg, “Genetic algorithm in search, optimization. and machine learning.” Addison-Wesley Book chapter, 1989.

[3] D. F. Rogers, S. G. Satterfield, F. A. Rodriguez, “Ship hulls, B-spline surfaces and CAD/CAM.” IEEE Comput Graphics and Applications, vol 3, no.9, Dec 1983, pp. 37-45 (ISSN 0272-1716).

[4] F. Yoshimoto, T. Harada, M. Moriyama, Y. Yoshimoto, “Data Fitting with a Spline using a Real Coded Genetic Algorithm.” Comput Aided Design, 2003, Vol. 35, pp. 751-760.

[5] R. W. Birmingham, T. A. G. Smith, “Automatic hull form generation: a practical tool for design and research.” In: *Proceedings of the Seventh International Symposium on Practical Design of Ships and Mobile Units,* 1988, pp. 281-287.

[6] T.H. Le, D. J. Kim, K. C. Min, S. W. Pyo, “B-spline surface fitting using genetic algorithm.” Journal of the Society of Naval Architectures of Korea, 2009, Vol. 46, pp. 87-95.

[7] Vo Trong Cang, Tran Van Tao. “Geometric design of ships.” Journal of Science & Technology Development, Vol 4, No 5&6-2001, pp 13-19 (in Vietnamese)

[8] W. Gordon, R. Riesenfeld, “B-spline Curves and Surfaces.” Comput Aided Geometric Des (Barnhill and Riesenfeld Ed.), Academic Press, 1974, pp. 95-126.