

Geometric design of patched spatial structures via WRD

Rubén Arias¹; César Otero²; Cristina Manchado³; Valentin Gomez-Jauregui⁴

Abstract

This paper describes some geometric concepts and their application in the geometric design of patched spatial structures to give designers a method which simplifies the whole process.

Though this kind of composite surface can be fitted into more complex forms, patches, especially their boundaries, are sometimes difficult to define. This is the new approach offered by CAGD – Computer-Aided Geometric Design: patches defined by boundaries instead of surfaces defined in a domain.

In the course of the article, some examples are examined from the designer's point of view, to show how CAGD solves the problem taking all the geometric restrictions into account.

In addition, a CG (Computational Geometry) technique exists which simplifies the three-dimensional problem of spatial structures and transforms it into a two-dimensional one, based on stereographic projection. This is an obvious advantage the development team has introduced.

This article gives a solution which condenses the three-dimensional problem inherent to spatial structures into a single plane in which the designer can define the whole structure: both the geometry of the shape of the spatial structure and the mesh comprised by its constituent struts.

Keywords

Spatial Structures, Domes, Patches, Computational Geometry, Computer Aided Geometric Design, Weighted Radial Displacement.

Theme

Structural and architectural design - conventional loads – steel

¹ MSc Eng. Assistant lecturer at the University of Cantabria, Spain ruben.arias@unican.es

² PhD.Eng. Associate Professor of CAD at the University of Cantabria, Spain cesar.otero@unican.es

³ B.Eng. Assistant lecturer at the University of Cantabria, Spain cristina.manchado@unican.es

⁴ B.Eng, MSc Arch. Assistant lecturer at the University of Cantabria, Spain tensegridad@gmail.com

1 Introduction

The design of a spatial structure begins with its geometric conception: what the structure's macroscopic shape is to be, the elements that are to comprise it, and how those elements are to be arranged. This work examines the geometry of space frames, spatial structures made up of linear elements or struts.

Progress in computing has provided tools capable of dealing with complex three-dimensional surfaces which the user can select or define, so describing the structure's macroscopic shape. However, computer programs tend to locate struts in these surfaces according to their inherent mathematical rules, so separating the designer from this facet of the structure.

This paper proposes a method for carrying out the whole geometric design of a structure from a single plane. Not only the macroscopic shape, but also the placement of the struts. To do this, techniques are required which can interrelate the plane with both parts of the structure.

Instead of some other computational methods based on the optimization of the stress on the mesh (see, for example, Kamenski[1]), the procedure here shown is completely focused on computational geometric considerations.

This contribution is organized as follows:

- Introduction. Computer-Aided Geometric Design. NURBS surfaces. The WRD method. Section 1
- Geometric definition of a bicubic patch. Surfaces of order n. Boundary properties and conditions. Section 2
- The problem of continuity in patches. Types of continuity. Section 3
- Transforming the plane into a three-dimensional surface. Section 4
- The fit of a spatial structure in a patch. Monolayer and bilayer structures. Section 5
- Definition of patch intersections. Section 6

1.1 Computer-Aided Geometric Design

Computer Aided Geometric Design (CAGD) has accurate, efficient methods for graphic representation of objects.

When this discipline came into being, its focus was on industrial design, but it quickly spread to many other fields and can be found in almost any computer modelling tool, including the conventional CAD programs (see Figure 1) used in the design of spatial structures (see Sánchez[2])

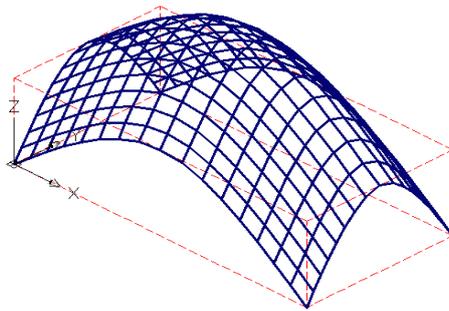


Figure 1. Mesh generated by CAGD techniques in a CAD program

1.2 NURBS

CAGD has also developed quick assessment mathematical expressions for graphic representation of geometric elements. One of such elements is surface, among which this article is concerned with NURBS, Non Uniform Rational B-Splines, which are mathematically defined as follows (see Figure 2):

$$N_{(u)} = \frac{P_0 w_0 B_{0(u)} + P_1 w_1 B_{1(u)} + P_2 w_2 B_{2(u)} + P_3 w_3 B_{3(u)}}{w_0 B_{0(u)} + w_1 B_{1(u)} + w_2 B_{2(u)} + w_3 B_{3(u)}} \quad (1)$$

$$B_i = \binom{3}{i} u^i (1-u)^{3-i} \quad i \in \{0,1,2,3\}$$

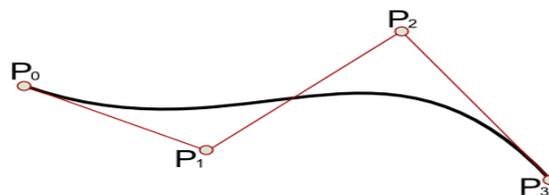


Figure 2. NURBS defined by four control points

Expression (1) is an algebraic sum of a set of coordinates for points P_i . These points generate a control polygon closely linked to the curve. This polygon is used to edit the curve graphically. In standard notation, n is taken as the order of the NURBS, using $n+1$ control points. Each point coordinate is increased by a real

number w_i , weight. The higher the weight of a point, the closer the curve will be to it.

Finally, all the points are combined using special functions B_i , called blending functions. These functions are carefully chosen to give the curve special characteristics.

If the numerator in (1) is taken, a curve called a B-spline is obtained. The denominator gives the curve special properties and will be studied in this article. The general expression for an n th-order NURBS surface is this:

$$N_{(u,v)} = \frac{\sum_{i=0}^n \sum_{j=0}^n P_{ij} w_{ij} B_{ij}(u,v)}{\sum_{i=0}^n \sum_{j=0}^n w_{ij} B_{ij}(u,v)} \quad B_{ij}(u,v) = \binom{n}{i} (1-u)^i u^{n-i} \binom{n}{j} (1-v)^j v^{n-j} \quad (2)$$

This expression (2) is valid for quadrilateral NURBS elements, in which the B_{ij} blending functions are tensorial products of the B_i in equation (1). As in the finite elements technique, there are several kinds of elements, which can be seen in Farin [3]. For this article, triangular NURBS will be used, in which case the expression is slightly different:

$$N_{(u,v)} = \frac{\sum_{i=0}^n \sum_{j=0}^{n-i} P_{ij} w_{ij} B_{ij}(u,v)}{\sum_{i=0}^n \sum_{j=0}^{n-i} w_{ij} B_{ij}(u,v)} \quad B_{ij}(u,v) = \frac{n}{(n-i-j)! i! j!} u^i v^j (1-u-v)^{(n-i-j)} \quad (3)$$

Where P_{ij} is the vector position of the control points, the w_{ij} parameters are the weights of each point and B_{ij} is the blending functions. In the case of a triangular mesh, the necessary number of control points is $(n+1)(n+2)/2$.

1.3 WRD

Weighted Radial Displacement is a method for NURBS transformation by projection of its control points. Sánchez-Reyes [4] describes how to transform planes and straight lines into quadrics and conics as second-order NURBS. The most interesting idea in that paper is the real meaning given to the u and v parameters, relating a plane and a surface (or a straight line with a curve) by means of a simple projection. For this to be possible, the other parameters need to be chosen with a particular criterion.

To introduce the concept, we shall describe the WRD method for conic curves, then show how the same idea applies in the case of surfaces.

1.4 Lines and Conics as NURBS

A straight line can be defined as a sum of two-point coordinates, mixing the coordinates using linear functions. But there is a formula which, keeping the linearity of the u parameter, gives a quadratic expression of the straight line (see Figure 3):

$$b_{(u)} = C_0 (1-u)^2 + C_1 2u(1-u) + C_2 u^2 \quad (4)$$

This equation can be written like this:

$$b_{(u)} = \frac{C_0 B_{0(u)} + C_1 B_{1(u)} + C_2 B_{2(u)}}{B_{0(u)} + B_{1(u)} + B_{2(u)}} \quad (5)$$

$$B_0 = (1-u)^2 \quad B_1 = 2u(1-u) \quad B_2 = u^2$$

Because the blending polynomials comply with $B_0+B_1+B_2=1$. In formula (5), compared with expression (1), the w_i weights are equal to 1. C_1 is the mean point between C_0 and C_2 .

The linearity of the u parameter is essential. Although expression (4) is quadratic, also because of the special selection of C_1 , every differential increase in u makes the increase in the line length $b_{(u)}$ constant. So the following equality can be stated:

$$b_{(u)} = C_0 (1-u)^2 + C_1 2u(1-u) + C_2 u^2 = C_0 (1-u) + C_2 u \quad (6)$$

Which turns out to be the vectorial equation of a u -parameter straight line.

Taking a point away from the straight line $b_{(u)}$ as the projection centres for its control points, a new curve is obtain. The WRD method explains how to change both its control points and their weights:

$$\text{New control points: } P_i = \frac{C_i}{\lambda_i}, \text{ new weights: } w'_i = \lambda_i w_i \quad (7)$$

Each position P_i lies on its respective straight line OC_i . To simplify the formulae, projection point O is taken as the coordinate origin. The λ_i parameters are the projection factors of the control points. Following these steps, a simplified expression for the new curve is obtained:

$$N_{(u)} = \frac{b_{(u)}}{\lambda_{(u)}} = \frac{C_0(1-u) + C_2u}{\lambda_0 B_0 + \lambda_1 B_1 + \lambda_2 B_2} \quad (8)$$

The equation shows how each point on the straight line $b_{(u)}$ is transformed according to a projection factor $\lambda_{(u)}$, so defining a point-to-point relationship between the straight line $b_{(u)}$ and the curve $N_{(u)}$ (see Figure 3). The projection point could be at infinity. In that case, another expression is obtained:

$$N_{(u)} = \frac{P_{(u)}}{w'_{(u)}} = P_0 B_{0(u)} + P_1 B_{1(u)} + P_2 B_{2(u)} \quad (9)$$

Both equations (8) and (9) are conic.

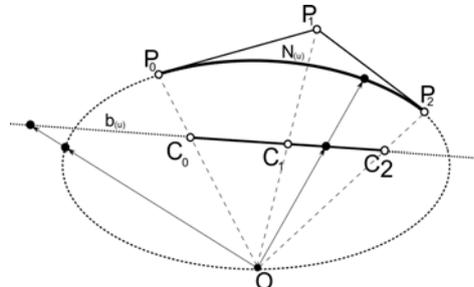


Figure 3. Conic as NURBS in WRD

Although the relationship between conics and NURBS is described in detail in Farin[3] and in Hearn[5], and this formula is the most widely implemented in computing applications, the substantial improvement achieved by Sánchez-Reyes [4] is the materialization of the u parameter for the curves. Now u is the parameter of the vectorial equation of the straight line $b(u)$ and is therefore linearly related with its length $L(u)$ measured from point C_0 in the direction C_0 - C_2 .

$$L(u) = u |C_0 C_2| \quad (10)$$

2 N-order NURBS surfaces and bicubic patches

In Sánchez-Reyes [4], only conics and quadrics are defined using WRD. But this method can be extended to obtain n -order surfaces. In the case of surfaces, biparametric geometric entities, they need to be related with a plane, in a similar way to how a curve was related with a straight line in 1.4, following these steps:

1. Selection of three vertex control points, C_{00} , C_{0n} , C_{n0} , and another projection point O .
2. Other plane control points are needed, and are obtained with this expression:

$$C_{ij} = \frac{nC_{00} + iC_{0n} + jC_{n0}}{n} \quad i \in [0..n] \quad j \in [0..n] \quad i + j \leq n \quad (11)$$

3. The values for P_{ij} and λ_{ij} are obtained using the same expressions in (5)
4. The NURBS is formulated like this:

$$N_{(u,v)} = \frac{b_{(u,v)}}{\lambda_{(u,v)}} = \frac{C_{00}(1-u-v) + C_{0n}u + C_{n0}v}{\sum_{i=0}^n \sum_{j=0}^{n-i} \lambda_{ij} B_{ij}(u,v)} \quad (12)$$

$$B_{ij(u,v)} = \frac{n}{(n-i-j)!i!j!} u^i v^j (1-u-v)^{(n-i-j)} \quad (13)$$

Where $b_{(u,v)}$ is the expression of the plane defining the vertex control points.

This formula is similar to equation (6). Conceptually, the same idea of projection of a point b , from a point O , using a factor λ . Figure 4 shows the geometric situation of the points and parameters defined in this formula for the case of $n=3$.

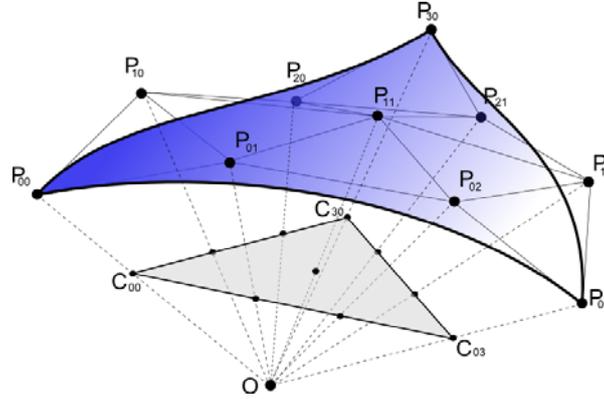


Figure 4. Bicubic patch

If O is at infinity, equation (12) becomes as follows:

$$N_{(u,v)} = \frac{P_{(u,v)}}{w'_{(u,v)}} = \sum_{i=0}^n \sum_{j=0}^{n-i} P_{ij} B_{ij(u,v)} \quad (14)$$

Bicubic patches are for the case of $n=3$, so it is necessary to define $(3+1)(3+2)/2 = 10$ parameters C_{ij} , P_{ij} and λ_i . Bicubic patches are interesting because their outlines have independent tangency restrictions. This makes them very flexible to satisfy complex geometric conditions in a simple way. Mathematically, NURBS are bijective with a plane. This guarantees the biunivocal relationship between points on the plane and on the NURBS surface.

2.1 Boundary conditions and properties.

There are some properties which are worth mentioning and taking into account. Some of them are illustrated in Figure 4, specifically adapted for $n=3$:

1. NURBS surfaces touch control points P_{00} , P_{03} and P_{30} .
2. The surface values along each side only depend on their related control points. For example, on the side P_{00} - P_{03} , only points P_{00} - P_{01} - P_{02} - P_{03} are involved.
3. The tangent plane at end P_{00} is given by points P_{00} , P_{01} and P_{10} , because only functions B_{00} , B_{01} , B_{10} have non-null derivatives there. Similarly, the tangent plane at P_{03} is given by P_{03} , P_{02} and P_{12} , and at P_{30} by P_{30} , P_{20} and P_{21} .
4. Each side is contained in a plane, because of the projection operation. These planes are defined by the vertex control points and point O .
5. The triangular patch is limited by P_{00} - P_{03} - P_{30} . For this, the value range $[0,1]$ is used for parameters u and v , but the surface extends beyond the patch, using the full range of real numbers $[-\infty, \infty]$.

Because of all these properties, the boundary problems can be controlled directly from the projection plane.

3 Patch-continuity problems

The possibility of joining two different patches make possible to put together composite surfaces for more

complex geometric situations. The basic problem is joining two bicubic patches along one side. This work resolves two kinds of continuity problem. Type C^0 problems are related with the value of the surface. Type C^1 problems are related with the directions of planes tangential to the surface.

3.1 Continuity problem C^0

The first and second properties can be used to solve a type C^0 problem directly. If two patches have to share a side, and the values the surface must have along that side need to be the same, the values for the P_{ij} control points they share must be equal. If they share the side defined by C_{03} and C_{30} : (see Figure 5)

$$P_{03} = P'_{03} \quad P_{12} = P'_{12} \quad P_{21} = P'_{21} \quad P_{30} = P'_{30} \quad (15)$$

For this to be valid, projection point O must be the same for both patches.

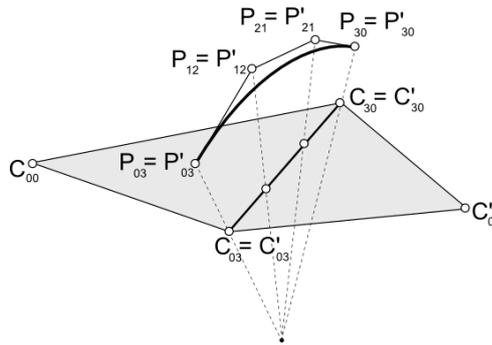


Figure 5

3.2 Continuity problem C^1

If the patches are required to share not just the same values along the shared side but also the same tangency plane, this is a C^1 problem.

Assuming that the patches share side C_{03} - C_{30} , the tangent plane at point P_{03} , according to property 3, is given by P_{03} , P_{02} and P_{12} , for a patch. This plane must be the same as that defined by points P'_{03} , P'_{02} and P'_{12} (see Figure 6.a) or, in other words, points P_{03} , P'_{02} and P_{12} , because as described in 3.1, point P_{03} is equal to P'_{03} and point P_{12} is equal to P'_{12} . Calculating the intersection of line OC'_{02} with the plane P_{03} - P_{02} - P_{12} gives the position of P'_{02} . The same method can be used for the other shared end, C_{30} . (see Figure 6.a).

To get complete C^1 continuity along the shared side, another restriction needs to be introduced. In this case, it is the position of point P'_{11} . This point must be on the plane defined by P_{12} , P_{21} and P_{11} . As before, it is obtained by projection of C'_{11} on that plane (see Figure 6.b).

Apart from the shared side, the other values for the second patch are completely free, and refer to the tangency value and plane at point C'_{00} .

When positions C'_{ij} and P'_{ij} have been found, the λ_{ij} values are calculated according to (7).

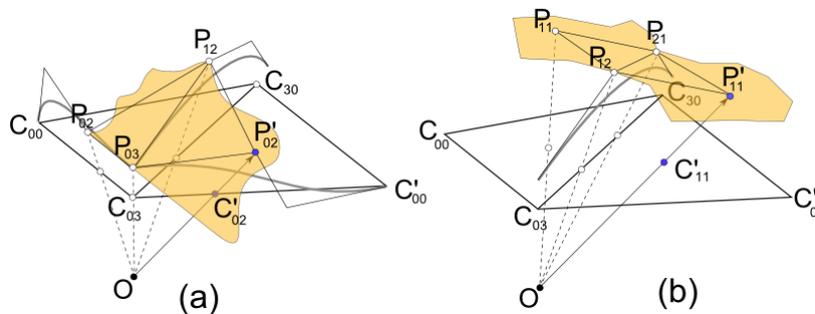


Figure 6. Solution of control point location

4 Transformation of the plane into a three-dimensional surface

So far the expressions defining surfaces have been studied. The problem is that these expressions use u and v parameters corresponding to a particular coordinate system, the system of barycentric coordinates of the triangle C_{00} - C_{03} - C_{30} .

Most engineering applications use the Cartesian coordinate system. It is needed to know how to change from one system to the other. If the vertex control points C_{ij} have Cartesian coordinates (x_{ij}, y_{ij}, z_{ij}) , and we want to transform point X of Cartesian coordinates (x, y, z) , we apply the following method:

$$X = \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} x_{03} - x_{00} & x_{30} - x_{00} & x_{00} \\ y_{03} - y_{00} & y_{30} - y_{00} & y_{00} \\ z_{03} - z_{00} & z_{30} - z_{00} & z_{00} \end{bmatrix} \begin{bmatrix} u \\ v \\ h \end{bmatrix} = MU \quad (16)$$

$$U = M^{-1} X \quad (17)$$

The coordinate change described in (16) and (17) makes the u and v coordinates obtained correspond to the barycentric coordinates of the triangle C_{00}, C_{03}, C_{30} . The h component obtained represents a third coordinate, out of the C_{00}, C_{03}, C_{30} plane, needed to represent three-dimensional space. The u base vector is the C_{03} - C_{00} vector and the v vector is C_{30} - C_{00} . The third base vector is the position vector of C_{00} , which is out of the Q plane defined by C_{00}, C_{03}, C_{30} .

(The standard use of barycentric coordinates of a triangle refers to three coordinates u, v , and w which are not linearly independent. Normally w is defined as $w = 1 - u - v$, and is contained in the plane made by C_{00}, C_{03}, C_{30} . The h and w components refer to different concepts.)

4.1 The method step by step

This section summarizes all the steps necessary to transform a point $X(x,y,z)$ on a plane into a point of a NURBS, illustrated in Figure 7:

1. Selection of a plane Q
2. Selection of three points on the plane C_{00}, C_{03}, C_{30} .
3. Selection of a point O out of the plane. In the case of a point at infinity, its coordinates need to be $O = (O_x, O_y, O_z)$ (18)
4. If $O(x_o, y_o, z_o)$ is finite, change the coordinate origin to O . This will affect all the points on the plane. Specifically, point $X(x,y,z)$ will change to $X(x-x_o, y-y_o, z-z_o)$.
5. Calculate the C_{ij} positions using expression (11)
6. Select the P_{ij} positions, bearing in mind that it must be in the direction OC_{ij} . If O is at infinity, then $P_{ij} = C_{ij} + k \cdot O$
7. Calculate the λ_{ij} values with formula (7)
8. Calculate the u and v parameters of X with (17)
9. Use formula (12) or (14) as appropriate. Point $X'(x',y',z')$ will be obtained
10. Go back to the usual coordinate origin if necessary. $X'(x'y'z') \rightarrow X'(x'+x_o, y'+y_o, z'+z_o)$

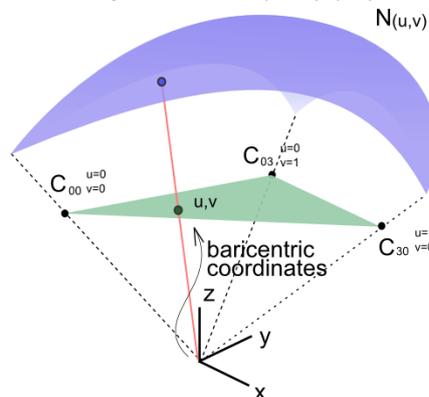


Figure 7. Projection of a point

5 Fitting a spatial structure into a patch

From the designer's point of view, the most important idea in this article is that the whole spatial structure is defined in the plane C_{00} - C_{03} - C_{30} . This idea is related with Otero[7] and Otero[8]. From this plane, it is possible to construct the structure's macroscopic geometry, its shape, which will be a NURBS surface. In the same plane, it is also possible to draw a mosaic, which will later form the struts of the space frame.

5.1 Single-layer structures

Section 4.1 showed how to transform a point. Single-layer structures are made up of a set of linear elements, joined by knots at their ends. These are the nodes which need to be transformed. The workflow for an application is as follows:

1. On plane Q, draw any mesh F, as a set of sides E_i .
2. Define surface N (selecting the vertex control points and the projection point).
3. For each side E_i
 - Transform its first end A_i into A'_i
 - Transform its second end B_i into B'_i
 - The solution will be segment $E'_i : A'_i B'_i$
4. Repeat

Note that only steps 1 and 2 need the intervention of the application user, the rest is automatic. Figure 8 shows an example of execution of the above sequence of commands:

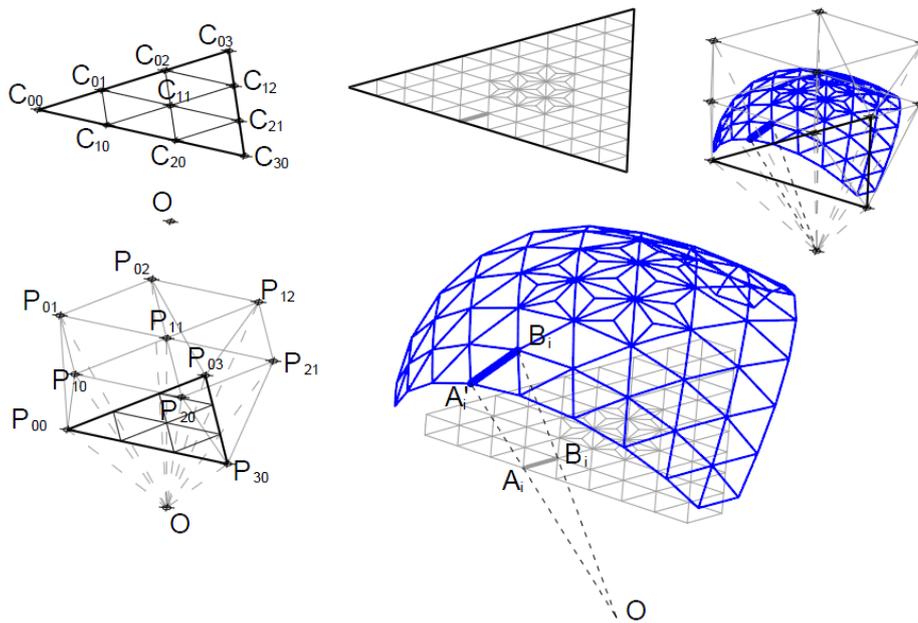


Figure 8. Single-layer spatial structure

5.2 Double-layer structures

Double-layer grids may be defined, Malla[6], as prefabricated systems consisting of two parallel networks of members forming the top and the bottom chords, which are interconnected by vertical and/or inclined web members. But according to Otero[7], this structure can be kept in a flat mosaic, which only shows the position of its diagonal struts.

The next figure, Figure 9, shows a 4,8,8 mosaic: at each vertex, a square and two octagons meet. This mosaic can represent the system of diagonal struts. The figure shows the whole system made up of the upper and lower nodes, and a general view of the structure based on this mosaic.

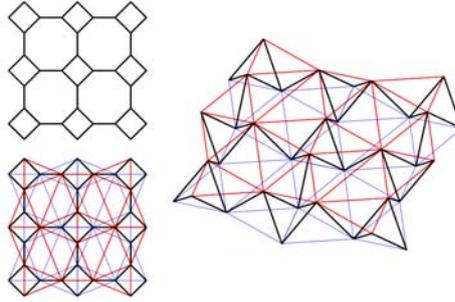


Figure 9. Mosaic and its associated flat double-layer spatial structure

Otero[7] also explains how to determine which points are above and which below for any mosaic in a way which can be made automatic, simply by defining a distribution law and the position of one of them. So a set of points UL in the upper layer and a set LL in the lower layer can be obtained, each making an independent mesh. This article proposes to project each set onto a different surface. There are various ways to do this, but here only one will be formulated: defining a distance.

If all the points UL and LL were projected onto the same surface, a single-layer structure would be obtained. In this case the set LL is to be separated by a distance, obtaining the desired result. This distance is to be measured along the direction of the projection, from O, and not in the normal direction to the surface, so the result will not be an equidistant surface.

The position correction is like this:

$$\text{Finite case: } P'_i = P_i \frac{|P_i| - \delta}{|P_i|}, \text{ Infinite case: } P'_i = P_i - O_i \frac{\delta}{|O_i|} \quad (18)$$

Where δ is the distance required. O_i is the direction (O_x, O_y, O_z) of the projection point in the case of projection from infinity.

If creation of an equidistant surface is required, this can be done by taking the normal direction to the surface, rather than the projection direction. Calculation of the normal unit vector $n(u,v)$ is described in Farin[3]. In this case, expressions (18) become:

$$P'_i = P_i - \delta n_{(u_i, v_i)} \quad (19)$$

The paper uses the expressions (18) in order to state the simplest solution for double-layer mesh structures.

The workflow is as follows:

1. In plane Q, draw a mosaic M as a set of sides D_i , also giving the position of one of the nodes
2. Using plane Q, define a surface N
3. Define distance δ to be put between the layers
4. Create the set of points UL for the upper layer, generating a framework mesh of sides ULE $_i$
5. Create the set of points LL for the lower layer, generating a framework mesh of sides LLE $_i$
6. Transform set ULE $_i$ as a single-layered structure
7. Transform set LLE $_i$ as a single-layered structure, correcting its position
8. For every side D_i
 - Transform its first end A_i into A'_i . Si A_i is in set LL, correct the position A'_i (18)
 - Transform its second end B_i into B'_i . If B_i is in set LL, correct the position B'_i (18)
 - The solution is segment $D'_i = A'_i B'_i$
9. Repeat

Again, only the initial steps, 1, 2 and 3, require user intervention. The result is shown in Figure 10.

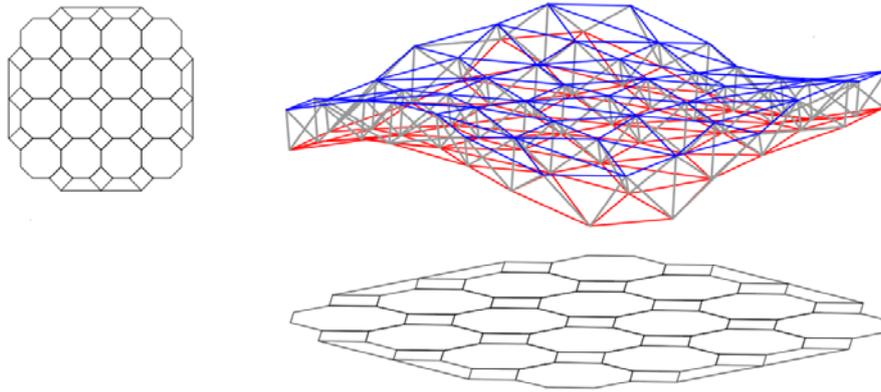


Figure 10. Double-layer mesh and its associated mosaic (4,8,8)

The idea of projecting the points onto a surface other than the one shaping the structure was used in Otero[8], where points of a particular mosaic are projected onto planes tangential to the surface rather than onto it to obtain panel structures.

6 Definition of patch intersections.

An intersection between surfaces is usually a twisted curve. The technique described in this article cannot solve this kind of intersection. However, it can calculate flat intersections. The straight lines to define the intersections can be proposed in the projection plane, and patches constructed on them which have their intersection with the lines proposed.

Section 3 described how to make contiguous patches using one of their sides. The next step is to be able to define a different line as the boundary between patches at any time. If it is possible to change the position of a single vertex control point while keeping the same NURBS, it will then be possible to change the position of a second point, while maintaining the same surface. When the new outline has been defined in this way, a different NURBS surface can be assembled, with the continuity conditions that may be necessary.

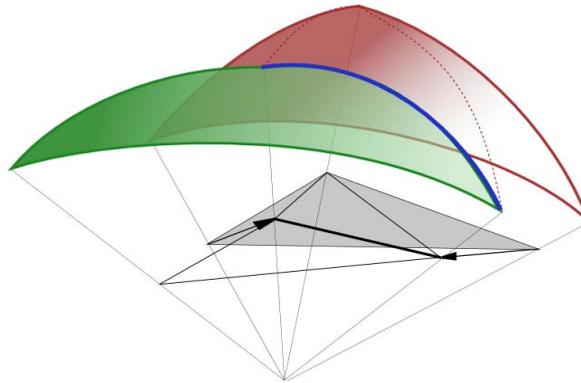


Figure 11. Location of intersections

6.1 Redefining a patch. Changing a vertex control point

NURBS surfaces have to have end control points, as was said in Chapter 2.1. Assuming that the point being changed is that of vertex C_{00} , the same surface needs to be redefined with vertices C'_{00} , C_{03} and C_{30} . From these points, the remaining plane control points can be calculated using expression (11). The values of the points and parameters related with side C_{03} - C_{30} do not change, as shown in Section 3.1. See Figure 12.a Point C'_{00} belongs to projection plane Q, so will have its corresponding barycentric coordinates u'_{00} and v'_{00} . From these coordinates, the corresponding NURBS point N'_{00} is calculated. This point will be the new P'_{00} . See Figure 12.a

Intermediate control point C_{02} will change automatically to position C'_{02} , and its weight should change as well. The new point P'_{02} must be contained in the el plane P_{03}, P_{02}, P_{12} to keep the same tangential plane at P_{03} . Calculating the intersection between that plane and line OC'_{02} will give the new position P'_{02} . The same idea enables calculation of point P'_{20} – which needs to be in the plane $P_{30}-P_{20}-P_{21}$ – and P'_{11} – which needs to be in the plane $P_{21}-P_{12}-P_{11}$. See Figure 12.b

When the P'_{ij} values have been obtained, the corresponding λ'_{ij} values can be obtained simply using formula (11).

The only thing left is determine the position of P'_{10} and P'_{01} . The side corresponding to $P'_{00}-P_{03}$ is defined by the points $P'_{00}-P'_{01}-P'_{02}-P_{03}$, all of which are known except P'_{01} . Taking any point K on the line $C'_{00}-C_{03}$, we can calculate its associated point N on the surface, for the old patch. As the same position N is required for the new patch, the following equation is solved (see Figure 12.c)

$$N = \frac{C'_{00}(1-u_k) + C_{03}u_k}{\lambda'_{00} B_{00(uk,vk)} + \lambda'_{01} B_{01(uk,vk)} + \lambda'_{02} B_{02(uk,vk)} + \lambda_{03} B_{03(uk,vk)}} \quad (20)$$

The values u_k, v_k are the barycentric coordinates of K in the new triangle $C'_{00}-C_{03}-C_{30}$. As K is on the side $C'_{00}-C_{03}$ its barycentric v_k is therefore zero. In expression (20) everything is known except λ'_{01} , the unknown value to be found. If projection point O is at infinity, the following equation will be resolved:

$$N = P'_{00} B_{00(uk,vk)} + P'_{01} B_{01(uk,vk)} + P'_{02} B_{02(uk,vk)} + P'_{03} B_{03(uk,vk)} \quad (21)$$

Resolving for P'_{01} . P'_{10} would be calculated in the same way.

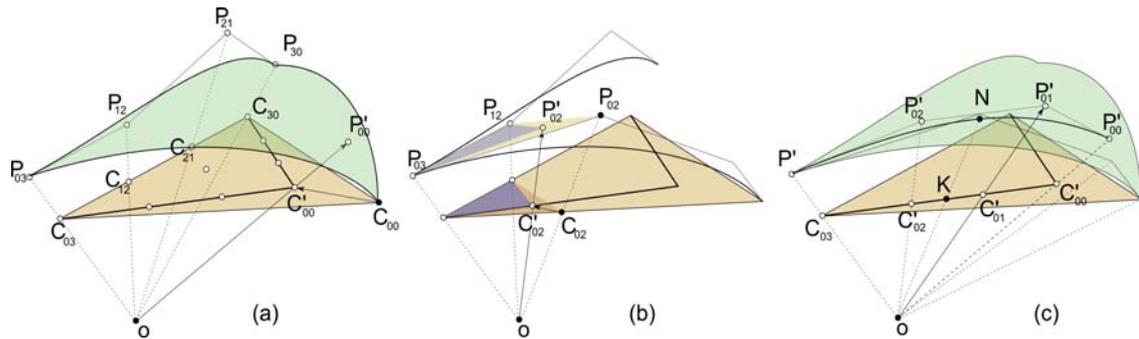


Figure 12. Patch intersection

7 Conclusions

This article has defined a new method for geometric design of spatial structures. This method allows the designer to work in a plane instead of three spatial dimensions, also obtaining a structure which is completely free.

The final geometry of the structure is provided by a NURBS, which is especially suitable for design because of its flexibility, and can also be adapted to more complex designs by using a patch definition.

As well as monolayer structures, this method can also create double-layered spatial structures. In either case, the only tasks the user needs to perform are the design of a flat framework mesh and definition of the final NURBS surface. The whole structure is fully defined in a single plane.

The rest of the process can be fully automated using CAD techniques, which resolve both the problems of strut positioning and those of continuity between patches. It also allows graphic edition of the whole structure, changing both the positions of the projection point and the different control points defining the surface, and the location of the nodes or the introduction of new points comprising the flat mesh to be projected.

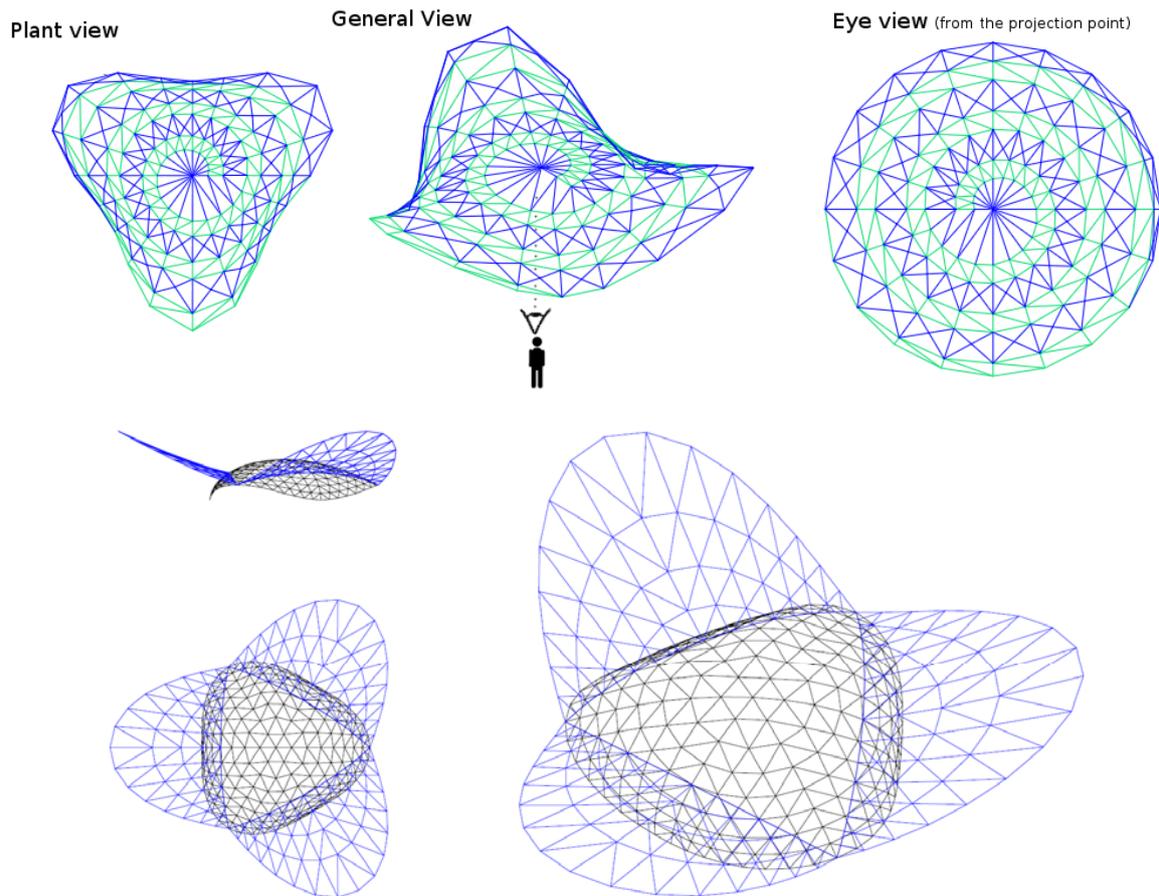


Figure 13. Examples

8 References

- [1] Kameshki E.S., Saka M.P., ,Optimum geometry design of nonlinear braced domes using genetic algorithm. *Computers and Structures*, Vol 85, pp. 71-79, 2007
- [2] Sánchez J., Serna M.A., Morer P., ,Using NURBS fitting techniques to represent tensile structures. *Intenational Conference on Adaptable Building Structures*. Eindhoven. 03-05 July 2006
- [3] Farin G., , *Curves and surfaces for CAGD: a practical guide*. Academic Press Inc., pp 343-358. 1993
- [4] Sánchez-Reyes J., Paluszny, M., ,Weighted Radial Displacement: A geometric look at Bézier conics and quadrics. *Computer-Aided Geometric Design*, Vol 17, pp 267-289, 2000
- [5] Hearn D., Baker M.P., ,*Computer graphics*. 2nd edition, Prentice Hall, pp.365-366. 1994
- [6] Malla R.B., Serrete R.L., ,Double-layer grids: review of static and thermal analysis methods. *Journal of Structural Engineering*, Vol 122 (8), pp 873-881. 1996
- [7] Otero C., Oti J., Villar R., Otero F., ,Classical geometry in flat and simple curved meshes. *Journal of the International Association for Shell and Spatial Structures*, Vol 33 (108), pp. 3-31. 1992
- [8] Otero C., Gil V., Álvaro J.I., , CR-Tangent Meshes. *Journal of the International Association for Shell and Spatial Structures*, Vol 41 (132), pp. 41-48. 2000