Research on None Uniform Rational B-spline Surface and Agent for Numerically Controlled Layout Design

Zhigang XU, Jingli Li
CAD Centre, School of Mechanical Eng., Shandong University,
17923# Jingshi Road, 250061, Jinan, Shandong China
Tel.: +86-531-88392851-8
E-mail: zhgxu@sdu.edu.cn

Abstract: Research on the integrated NC conceptual layout design (I-NCC) concerned with a broader area of interests. The key issues of I-NCC system are associated with NURBS and agent. Firstly, formulas for the derivatives and normal vectors of non-rational B-spline and NURBS are proved based on de BOOR’s recursive formula. Compared with the existing approaches targeting at the non-rational B-spline basis functions, these equations are directly targeted at the controlling points, so the algorithms and programs for NURBS curve and surface can also be applied to the derivatives and normals, the calculating performance is increased. A simplified equation is also proved in this paper. Secondly, the NC conceptual configuration design is transformed into a 3D cuboids layout problem by the introduction of three typical modules: translation module, rotation module and base module based on the analysis of the normal unit vector of work piece surface (in NURBS format). 3D cuboids layout problem is viewed as a generalization of the quadratic assignment problem and therefore belongs to the class of NP hard problems. Apart from the complexity and variety of 3D layout optimization algorithms, this paper introduces agent oriented cooperative design system. Agent models and the corresponding design management systems are put forward to deal with the creative NC layout design. Though the key theoretical issues are now applied to the NC system design, there should be more industrial applications because of the prevalent proliferation nature of NURBS and agent. Copyright © 2014 IFSA Publishing, S. L.

Keywords: NURBS, Normal vector, 3D layout design, Agent.

1. Introduction

The mass-production of parts of complex shape to close tolerances often requires highly sophisticated numerically controlled (NC) machines. NC machines are typically classified as 3-, 4- and 5-axis machines depending on the effective number of degrees of freedom (DOF) enjoyed by the cutter. The design of 3-, 4- and 5-axis NC machine movement structures depends mostly on the analysis of work-piece surface, so does the selection of the commercial existing NC systems. This motivates the following research interests:

Free form expression and analyzing, esp., the calculation of the normal unit vector of NURBS etc;

NC system conceptual structure expression, design and selection;
Methodology on 3D layout configuration within NC system design context;
NC system layout and embodiment design.

Now CAD/CAM society feels strongly about None Uniform Rational B-spline Surface (NURBS). The rapid proliferation of NURBS is due partly to their excellent properties as simplicity in data preparation and small amount of surface data compared to that of other surfaces, and it soon incorporate in such international standards as Product Data Exchange Specification, and International Standard for the Exchange of Product Model Data [1] (Piegal, 1991).
Derivatives and normal vectors of parametric curves are important issues in computer graphics and geometric modeling [2] (Boehm, 1984). Although the derivative at any single point on a NURBS curve or surface can be calculated by subdivision, it is often useful to access the possible range of derivatives or normal vectors for an entire curve segment of surface path. For example, gasps and inflection points on a curve can be detected by analysis. Bounds of derivative and normal directions also help in detecting intersections between two curves or surfaces. For this end, [3] Saito (1995) developed a set of efficient methods to calculate hodographs and normals of rational Bezier curves and surfaces containing a proof that the normal vector of a rational Bezier path of degree (n, m) maybe described by a polynomial path of degree (3n-2, 3m-2), which is used to obtain estimates of direction of the normal vector of path p. [4] Ueda (1996) developed two formulas to compute the mean normal vector to a surface bounded by Bezier curves. [5] Pendharkar (1999) puts forward a different approach to calculate the normal vector patch of a rational Bezier patch; a different algorithm of computing control points of the normal vector patch is put forward. Till now researches relating to derivatives and normal vectors are abundant, however few approaches are directly applied to NURBS.

NC modular design is prevalent recently, and the conceptual configuration can be expressed simply by the arrangement of three different kinds of modules: the translation module, rotation module and base module [6] (Hatamura, 1995). Different combination of the above modules can express the NC movement layout, which belongs to a 3D layout problem. While NC 3D layout design is defined as the coordinates and orientations of components that minimize the cost and satisfies certain placement requirement. Recent research efforts have led to several approaches for the layout of 3D objects of complex geometry. A variety of optimization algorithms have been applied to the layout problems, they are summarized as the following types: heuristic rule-based approaches [7] (Dai, 1994), traditional optimization approaches [8] (Scheithauer, 1995), genetic algorithms [9] (Ikonen, 1997), simulated annealing algorithms [10] (Cagan, 1998), extended pattern search algorithms [11] (Yin, 2000) and hybrid approaches [12] (Hills, 1997). While NC 3D layout design is different from the traditional 3D cuboids layout problems in that the objectives and constraints are either too flexible or rigid due to design, manufacture, assembly and maintenance reasons etc [13] (Dimarogonas, 2000). So we tackle this problem in a different way, where each module is regarded as an autonomous agent, and the NC system is adopted as a model of flexible organism in which self-contained entities organize them-selves without the power of an external force.

Actually agent technology radically alters the way in which we conceptualize and build software systems [14] (Maja, 1995). It has been widely used in Flexible Manufacturing Systems (FMS) and decision-making in assembly [15] (Zha, 2002) etc. The connotation of open expert systems employing dynamic integrating agents has now been applied in concurrent configuration problems [16] (Koulinitch, 1998), dynamic scheduling in manufacturing [5] (Pendharkar, 1999), and creative design in conceptual design stage [17] (Petrovic, 1997) etc. The idea of a set of design agents confronted to a creative design process may connect the work on the subjective techniques based on the co-operation of human design agents [18] (Chao, 2002). With the help of computer aided design possibilities offered by the new computational resources, current research is dedicated to the relationships of computers and creativity. So combination of traditional 3D layout problems with modern AI technology is a promising way.

This article firstly proves the formula of normal unit vector to NURBS. The normal vector calculation paradigm shown in this paper is proved to be more effective to specify the cutter translation in x, y and z directions and the rotation around the x, y and z axis. The agent oriented NC system creative design paradigm is also put forward. The working flow of I-NCC system is given as in Fig. 1.

![Fig. 1. Work flow of I-NCC system.](image-url)
So the arrangement of the paper is as the follows. Section 2 proves a formula of the normal unit vector to nonrational B-spline and NURBS and a theorem that the normal of nonrational B-spline is the special case of that of NURBS is also proved; section 3 talks about NC system design requirement based on the analysis of normals of NURBS; section 4 gives the module definitions and constraints for NC conceptual design; section 5 introduces the agent architecture and the agent oriented creative design paradigm; and section 6 draws conclusion of this paper.

2. Normal Unit Vector of NURBS


The derivatives are based on the differentiated basis functions of lower degree, so the \( r \)th derivatives of the basis function of degree \( p \) are calculated by the recursive formulas (1) applied twice when adding derivatives of NURBS, so the programs based on the non-rational B-spline applied directly to the basis functions. While in our approach, the first derivative of the nonrational B-spline curve is calculated based on the differentiation of the original basis function of (p-r) degree. And the recursive formula is set up based on the nonrational B-spline basis function of degree \( p \) are calculated by the similar to [21] G. Farin’s work (1995).

The following is the derivation and prove of the formula (2) applied twice when adding derivatives of NURBS, so the programs based on the non-rational B-spline applied directly to the basis functions. While in our approach, the first derivative of the nonrational B-spline curve is calculated based on the differentiation of the original basis function of degree \( p \) are calculated by the similar to [21] G. Farin’s work (1995).

\[
p'(u) = \sum_{j=0}^{l} d_j N_{j,k-1}(u),
\]

where

\[
d_j = \begin{cases} 
  d_j & l = 0 \\
  (1 - \alpha_j') d_{j-1} + \alpha_j' d_j & j = i - k + 1, \ldots, i + 1
\end{cases}
\]

where, \( \alpha_j' = \frac{u - u_j}{u_{j+k+1} - u_j} \)

\[
d_j = k \frac{d_j - d_{j-1}}{u_{j+k+1} - u_j}, \quad j = i - k + 1, i - k + 2, \ldots, i;
\]

A B-spline curve is obtained by taking bidirectional net of control points, say, the \((m+1) \times (n+1)\) points \( d_{i,j} ; i = 0, 1, \ldots, m, \quad j = 0, 1, \ldots, n. \)

Forming a controlling nets, two-knot vectors (say, \( U = [u_0, u_1, \ldots, u_{m+k+1}] \), \( V = [v_0, v_1, \ldots, u_{n+k+1}] \)), and the products of the univariate B-spline functions.

A \( k \times l \) nonrational B-spline surface is the generalization of tensor-product and is defined as follows:

\[
p(u, v) = \sum_{i=0}^{m} \sum_{j=0}^{n} d_{i,j} \cdot N_{i,k}(u) \cdot N_{j,l}(v),
\]

and \( N_{j,l}(v) \) is the B-Spline basis function of degree \( k \) and \( l \) in \( U \) and \( V \) directions defined over the following knot vectors:

\[
U = [0, \ldots, 0, u_{k+1}, \ldots, u_{r-k-1}, 1, \ldots, 1], \quad r = m + k + 1, \quad m \in \mathbb{N}, \quad k \geq 0.
\]

\[
V = [0, \ldots, 0, v_{l+1}, \ldots, v_{s-l-1}, 1, \ldots, 1], \quad s = n + l + 1. 
\]

There is a so-called “\( v \) curve” and “\( u \) curve” at the point \((u_t, v_t)\) on B-spline surface, the partial derivatives

\[
p_u(u_t, v_t) = \frac{\partial p(u, v)}{\partial u} \bigg|_{u=u_t};
\]

is the tangential vector in \( U \) direction.

The derivative:

\[
p_v(u_t, v_t) = \frac{\partial p(u, v)}{\partial v} \bigg|_{v=v_t};
\]

is the tangential vector in \( V \) direction.

The unit normal vector at \((u_t, v_t)\) is calculated by the following formula:

\[
\hat{n}(u_t, v_t) = \frac{p_u(u_t, v_t) \times p_v(u_t, v_t)}{|p_u(u_t, v_t) \times p_v(u_t, v_t)|},
\]

Based on the \( r \)th order derivatives of B-spline curve suggested by De Boor–Cox (1972). So the first order derivative of B-spline curve is:
And two formulas are put forward.

\[ p_s(u, v) = \sum_{j=0}^{a} \frac{\sum_{i=0}^{b} (d_{i,j} - d_{i,j-1}) \cdot N_{i,j}(v)}{v_{i+1} - v_i} \cdot N_{i,j}(u); \]  

(4)

\[ p_s(u, v) = \sum_{j=0}^{a} \frac{\sum_{i=0}^{b} (d_{i,j} - d_{i,j-1}) \cdot N_{i,j}(u)}{v_{j+1} - v_j} \cdot N_{i,j}(v); \]  

(5)

Given:

\[ u_a \leq u < u_{a+1}, \quad a = k, k + 1, \cdots, m + 1, \]

\[ v_b \leq v < v_{b+1}, \quad b = l, l + 1, \cdots, n + 1, \]

2.1. Proof of the Formula (4) and (5)

It is well known that B-Spline surface is formed by recursively calculating the controlling points (like in formula (1)) alongside \( U \) and \( V \) directions sequentially. Take \( U \) direction as the first calculation odor, be \( u = u_a \), calculating each \( k+1 \) controlling nodes in \( n+1 \) arrays by formula (1), we can get \( n+1 \) new controlling points \( D_j \).

Continuing to calculate these \( n+1 \) new controlling points \( D_j \) by formula (1), we can get one curve on B-Spline surface in \( V \) direction. This curve is called "\( U \) curve". From equation (2) we can get the tangent derivative in \( V \) direction is like this:

\[ p(u, v) = \sum_{j=b-l+1}^{a} \frac{D_j - D_{j-1}}{v_{j+1} - v_j} \cdot N_{j,l}(v) \]

\[ v_b \leq v \leq v_{b+1}; \quad b = l, l + 1, \cdots, n + 1 \]

whereas:

\[ D_j = \sum_{i=a-k}^{a} d_{i,j} \cdot N_{i,k}(u) \bigg|_{u=u_a}, \quad u_a \leq u < u_{a+1}, \]

and

\[ D_{j-1} = \sum_{i=a-k}^{a} d_{i,j-1} \cdot N_{i,k}(u) \bigg|_{u=u_a}, \quad u_a \leq u < u_{a+1}, \]

So,

\[ p_s(u, v) = \sum_{j=b-l+1}^{a} \frac{\sum_{i=a-k}^{a} (d_{i,j} - d_{i,j-1}) \cdot N_{i,j}(v)}{v_{i+1} - v_i} \cdot N_{i,j-1}(u); \]

where

\[ a = k, k + 1, \cdots, m + 1, \]

\[ v_b \leq v < v_{b+1} \cdot \quad b = l, l + 1, \cdots, n + 1, \]

Analogous to the last equation, we can get:

\[ p_s(u, v) = \sum_{j=b-l+1}^{a} \frac{\sum_{i=a-k}^{a} (d_{i,j} - d_{i,j-1}) \cdot N_{i,j}(u)}{v_{j+1} - v_j} \cdot N_{j,l+1}(v); \]

where

\[ u_a \leq u < u_{a+1}, \quad a = k, k + 1, \cdots, m + 1, \]

\[ b = l, l + 1, \cdots, n + 1, \]

This is the end of the proof of equation (4) and (5).

2.2. Derivation of Unit Normal Vector of NURBS

The controlling \((m+1) \times (n+1)\) net points \( d_{i,j}, \quad i=0,1, m, j=0,1, n \) and corresponding weights \( \omega_{i,j} \) form a controlling net. A NURBS surface is the rational generalization of tensor-product rational B-Spline surface and is defined as follows:

\[ p(u, v) = \sum_{i=0}^{m} \sum_{j=0}^{n} \frac{d_{i,j} \cdot \omega_{i,j} \cdot N_{i,k}(u) \cdot N_{j,l}(v)}{\sum_{i=0}^{m} \sum_{j=0}^{n} \omega_{i,j} \cdot N_{i,k}(u) \cdot N_{j,l}(v)} \]

(6)

where

\[ \left\{ \begin{array}{l} u_k \leq u < u_{k+1} \\ v_l \leq v < v_{l+1} \end{array} \right. \]

\( N_{i,k}(u) \) and \( N_{j,l}(v) \) are the nonrational B-Spline basis functions of degree \( k \) and \( l \) defined over the following knot vectors:

\[ U = [0, \ldots, 0, \quad u_{k+1}, \ldots, u_{r-k-1}, 1, \ldots, 1], \]

\[ r = m + k + 1. \]

\[ V = [0, \ldots, 0, \quad v_{l+1}, \ldots, v_{s-l-1}, 1, \ldots, 1], \]

\( s = n + l + 1. \)
The normal unit vector on \((u_s, v_t)\), is defined as follows:

\[
\mathbf{n}(u, v) = \left. \frac{p_s(u, v)}{\|p_s(u, v)\|} \times \frac{p_t(u, v)}{\|p_t(u, v)\|} \right|_{u=u_k, v=v_l},
\]

where

\[
p_u(u_s, v_t) = \frac{\partial p(u, v)}{\partial u} \bigg|_{u=u_k},
\]

is the tangential vector in \(U\) direction.

\[
p_v(u_s, v_t) = \frac{\partial p(u, v)}{\partial v} \bigg|_{v=v_l},
\]

is the tangential vector in \(V\) direction.

In 

\[
A(u, v) = \sum_{i=0}^{m} \sum_{j=0}^{n} d_{i,j} \cdot \omega_{i,j} \cdot N_{i,k}(u) \cdot N_{j,l}(v); \quad B(u, v) = \sum_{i=0}^{m} \sum_{j=0}^{n} \omega_{i,j} \cdot N_{i,k}(u) \cdot N_{j,l}(v);
\]

Then:

\[
p(u, v) = \frac{A(u, v)}{B(u, v)},
\]

The \(u\) and \(v\) directional derivative is as:

\[
p_u(u_s, v_t) = \frac{A_u(u_s, v_t) \cdot B(u_s, v_t) - A(u_s, v_t) \cdot B_u(u_s, v_t)}{B^2(u_s, v_t)}, \quad (7)
\]

and

\[
p_v(u_s, v_t) = \frac{A_v(u_s, v_t) \cdot B(u_s, v_t) - A(u_s, v_t) \cdot B_v(u_s, v_t)}{B^2(u_s, v_t)}, \quad (8)
\]

According to equation (4), we can get the following equations:

\[
A_u(u, v) = \sum_{i=0}^{m} \sum_{j=0}^{n} \left(d_{i,j} \cdot \omega_{i,j} - d_{i-1,j} \cdot \omega_{i-1,j} \right) N_{j,l}(v) \bigg|_{v=v_l} \cdot N_{i,k-1}(u);
\]

\[
B_u(u, v) = \sum_{i=0}^{m} \sum_{j=0}^{n} \left(d_{i,j} \cdot \omega_{i,j} - d_{i-1,j} \cdot \omega_{i-1,j} \right) N_{i,k}(u) \bigg|_{u=u_k} \cdot N_{j,l-1}(v);
\]

\[
B_v(u, v) = \sum_{i=0}^{m} \sum_{j=0}^{n} (\omega_{i,j} - \omega_{i,j-1}) \cdot N_{i,k}(u) \bigg|_{u=u_k} \cdot N_{j,l-1}(v);
\]

Substituting

\[
A_u(u, v), B_u(u, v), A(v, u), B(v, u) \quad \text{in equation (7), } p_u(u, v) \quad \text{is got.}
\]

Similarly,

\[
A_v(u, v) = \sum_{i=0}^{m} \sum_{j=0}^{n} \left(d_{i,j} \cdot \omega_{i,j} - d_{i-1,j} \cdot \omega_{i-1,j} \right) N_{i,k}(u) \bigg|_{u=u_k} \cdot N_{j,l-1}(v);
\]

\[
B_v(u, v) = \sum_{i=0}^{m} \sum_{j=0}^{n} (\omega_{i,j} - \omega_{i,j-1}) \cdot N_{i,k}(u) \bigg|_{u=u_k} \cdot N_{j,l-1}(v);
\]

Substituting

\[
A_v(u, v), B_v(u, v), A(v, u), B(v, u) \quad \text{in equation (8), function } P_v(u_s, v) \quad \text{can be calculated.}
\]

Now \(\mathbf{n}(u, v)\) is got by the substitution of equation (7) and (8) in formula (2).

**Theorem 1:** The normal unit vector of non-rational B-Spline surface is the special case of the unit normal vector of NURBS.

**Proof of Theorem 1:**

AS we all know that the difference between non-rational B-Spline surface and NURBS lies in the existence of the weight \(\omega_{i,j}\). If \(\omega_{i,j} = \omega\), the weight \(\omega_{i,j}\) has no effect on B-Spline surface, in fact equation (6) has been changed into the form of non-rational B-Spline surface equation.

In equation (8), when \(\omega_{i,j} = \omega (\text{constant}),\)
\[ A(u,v) = \omega \sum_{j=0}^{n-1} \sum_{i=0}^{m-1} \frac{\left| d_{j,i} - d_{j,i+1} \right|}{v_{j+1} - v_j} N_j(u) N_i(v) \]

\[ B(u,v) = 0 \]

 Normalize the B-Spline basis functions, \[ B(u,v) = \omega \], in equation (7).

So equation (8) has been changed into equation (5). Analogously, when \[ \omega_{i,j} = \omega \], equation (7) has been changed into equation (4), which means the equation of the unit normal vector of NURBS has been changed into the equation of the unit normal vector of non-rational B-Spline surface. This is the end of proof of theorem 1.

Now give a calculation example based on the original data of \( \omega_{i,j} \) and \( d_{i,j} \) in Table 1.

At the point (0.8, 0.5), we can get the result of:

\[ p_u(0.8,0.5) = (0.151, -0.632, -0.341) \]

\[ p_v(0.8,0.5) = (0.642, -0.136, 0.236) \]

By the equation:

\[ \vec{n}(0.8, 0.5) = p_u(0.8, 0.5) \times p_v(0.8, 0.5) \]

\[ = \begin{vmatrix} p_u(0.8, 0.5) \times p_v(0.8, 0.5) \\ (0.371, -0.205, 0.904) \end{vmatrix} \]

The result was shown in Fig. 2.

Table 1. Input points and corresponding weights for NURBS surface.

<table>
<thead>
<tr>
<th>( \omega_{i,j} )</th>
<th>( d_{i,j} )</th>
<th>Column 0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Row 0 3, (0,0,53)</td>
<td>4, (12,1,63)</td>
<td>3, (33,1,46)</td>
<td>6, (50,1,53)</td>
<td>7, (74,0,32)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 5, (1,21,23)</td>
<td>7, (11,20,35)</td>
<td>5, (33,20,43)</td>
<td>4, (52,20,34)</td>
<td>4, (74,20,54)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2 6, (2,32,26)</td>
<td>5, (12,30,48)</td>
<td>8, (32,31,38)</td>
<td>7, (54,32,27)</td>
<td>6, (73,34,33)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3 8, (1,42,26)</td>
<td>3, (12,40,35)</td>
<td>6, (32,42,36)</td>
<td>6, (53,43,32)</td>
<td>7, (74,40,27)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4 4, (1,52,45)</td>
<td>1, (10,50,54)</td>
<td>4, (34,50,45)</td>
<td>7, (53,50,26)</td>
<td>9, (75,50,33)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Fig. 2. Normal unit vectors of NURBS.

### 3. NC System Design Requirement

NC conceptual layout design is the first and fundamental step in the whole design process. Some complex freeform surfaces require 4-5-axis NC machining. The ability of a cutter to access a work piece surface depends not only on the DOF of the NC systems but also on the nature of cutter itself. Cutters are usually classified as flat-end, fillet-end, drum-side and ball-end types, usually flat-end, fillet-end, and drum-side cutters offer high surfacing and accuracy, but the ability to access the work piece is more strictly limited than the ball-end cutter.

Cutter’s DOF is defined by the maximum angle, which means the inclination of the cutter’s axis to the normal of the surface at a giving point. It can be deduced that in a flat-end cutter, \( \theta = 0 \), in a fillet-end and drum-side cutter, \( \theta > 0 \), and in a ball-end cutter, \( \theta = 90^\circ \). Clearly ball-end cutter offers the largest range of access to the workpiece, while flat-end cutter has no extra DOFs besides that of the NC systems. Actually the summation of the cutter’s DOF and the NC system’s DOF is the cutting DOF. So the efficient NC system should supply enough potential DOF for the flat-end, drum-side and fillet-end cutter to access the surface of the work piece within the range of the machine’s DOF. So the analysis of the normal unit vector of workpiece surface supplies the basic DOF requirement of the NC system. Fig. 3 gives the example of the cutting process of the steam turbine blade.

In Fig. 3 (a), when the cutter moves along a route: \( y = f(x) \), it will be mapped to a \( u \) curve (or a \( v \) curve) on the NURBS surface. In the section B-B of Fig. 3 (a), \( \vec{n} \) changes from \( \vec{n}(u_s,0) \) to \( \vec{n}(u_s,1) \).

The rotation angle \( \theta = \cos^{-1} \frac{\vec{n}(u_s,0) \cdot \vec{n}(u_s,1)}{\left| \vec{n}(u_s,0) \cdot \vec{n}(u_s,1) \right|} \) is the rotating angle around \( y \) axis. Similarly in section A-A, the cutter move along one \( v \) curve, \( \vec{n} \) change from \( \vec{n}(0,v_t) \) to \( \vec{n}(1,v_t) \), and the rotating angle \( \alpha = \cos^{-1} \frac{\vec{n}(0,v_t) \cdot \vec{n}(1,v_t)}{\left| \vec{n}(0,v_t) \cdot \vec{n}(1,v_t) \right|} \) is the rotating angle around \( x \) axis.
Fig. 3. Cutter definitions and the cutting simulation.

So the cutter translate in x, y and z axis during the blade cutting process, in the mean while the cutter’s rotation requirement around x and y axis is also suggested. Fig. 3 (b) is the blade shape and the position on the turbine. Fig. 3 (c) is the selection of cutters in large-scale cutting and finishing. Fig. 3 (d) is the definition of the degree of freedom (DOF) of cutters, where x, y and z denotes translations along x, y and z axis and A, B and C denotes rotations around x, y and z axis respectively.

The movement requirement is proposed based on the calculation result of the changing range of the normal vector of the blade surface denoted by NURBS or B-spline by formula (3). In order to access all the blade surface area, 5-axis NC movement is required in this example. And the translation along x, y and z axis and rotation round x and y axis is required, i.e. the NC system should offer the cutter three translation DOF along the x, y and z axis and two rotation DOF around x and y axis. This is the fundamental functional requirement of the NC system.

And in next chapter we will talk about the conceptual 3D NC system layout design according to the preliminary movement requirement, where modular definitions and constraints are firstly proposed.

4. Module Definitions and Constraints

Definition 1. Base module is a kind of supporting module; it cannot translate or rotate along or around any axis (Fig. 4.a (1)). There are typically two kinds of Base modules, denoted by H-Base and V-Base for horizontal and vertical positioning.

Definition 2. Translation module is a kind of linear movement module along x, y and z axis, denoted by T-X, T-Y and T-Z respectively (Fig. 4.a (2)).

Definition 3. Rotation module is a kind of module that can rotate around x, y and z axis at one time, denoted by R-X, R-Y and R-Z respectively (Fig. 4.a (3)).

Derivation. Two rotation modules cannot connect with each other, but can connect with other types of modules, e.g. Base or T modules.

From the above module definitions, the following module constraints can be deduced.

\[
\begin{align*}
R_i \cap R_j &= 0; & H_i \cap H_j &= 0; \\
B_i \cap B_j &= 1; & R_i \cap H_j &= 1; & 0 \text{ means illegal;}
\end{align*}
\]

1 means legal.

The other design constraints are also summarized as the follows.
1. If the workpiece is heavy then the workpiece cannot move vertically;
2. If the precision requirement of the workpiece is high then the workpiece should be fixed or has only one horizontal movement;
3. If the workpiece is heavy and has high precision requirement then the workpiece fixture should be connected with the base module.

5. NC Layout Design Based on Agent

The engineering design is viewed as a distributed system consisting of various autonomous entities completing the special task through cooperation. There are two types of basic elements in the system: entities and activities. The entity includes the
following elements: the CAD systems dealing with the graphic issues, human computer interface etc, users and various information-handling agents. Activity is the process of the entity performing the design process. In the design process there are also two categories of information: foreground and background [13] (Dimarogonas, 2000), Table 2 is the list of information and the corresponding activities and participants.

![Diagram of Module Definitions](image)

(a) Module definitions

![Diagram of Conceptual Module Combination](image)

(b) Conceptual Module Combination

![Diagram of "C" Circle Structure](image)

(c) “C” Circle Structure

**Fig. 4.** Module definitions and the NC conceptual layout.

**Table 2.** Information management of the design process.

<table>
<thead>
<tr>
<th>Information</th>
<th>Activities</th>
<th>Participants</th>
</tr>
</thead>
<tbody>
<tr>
<td>Design requirements</td>
<td>User input, file input</td>
<td>CAD system, user</td>
</tr>
<tr>
<td>Design intents</td>
<td>Deduction, input</td>
<td>user, agent, CAD system</td>
</tr>
<tr>
<td>Design methods</td>
<td>Search, inquire</td>
<td>user, agent, CAD system</td>
</tr>
<tr>
<td>Design standards</td>
<td>Input, display</td>
<td>user, agent, CAD system</td>
</tr>
<tr>
<td>Standard parts</td>
<td>Data base inquire</td>
<td>user, agent, CAD system</td>
</tr>
<tr>
<td>Design history</td>
<td>Record, compare, modify</td>
<td>agent, CAD system</td>
</tr>
<tr>
<td>Engineering analysis records</td>
<td>Calculate, state change</td>
<td>agent, CAD system</td>
</tr>
<tr>
<td>Design modification records</td>
<td>Record, compare, modify, decide</td>
<td>agent, CAD system</td>
</tr>
<tr>
<td>Design process controlling</td>
<td>Begin, end, interrupt</td>
<td>user, CAD system</td>
</tr>
<tr>
<td>Massages</td>
<td>Receive, send, accept, refuse</td>
<td>agent, user, CAD system</td>
</tr>
<tr>
<td>Participants state</td>
<td>Active, freeze</td>
<td>agent, user, CAD system</td>
</tr>
</tbody>
</table>

### 5.1. Functional Structure of NC Design Agent

An agent contains five modules (Fig. 5). The function and performance of the modules are as follows: *Acquaintance module* is the representation of other agents and of the local domain level system respectively. *Cooperate module* is responsible for managing the agent’s social activities, the need being detected by the *situation assessment module*. *Control module* is the interface to the domain level system. *Situation assessment module* decides which activities should be performed locally and which should be delegated to other agent, which requests for cooperation should be honored, how requests should be realized, and what action should be taken as a result of freshly arriving information, etc. It issues instructions to, and receives feedback from the other modules. The *self-module* is a repository for all the data, which the underlying domain level system has generated or which has been received as a result of interaction with others. The *communication manager* sends and receives messages to/from other agents in the community.
Definition 4. Agent is described as a sextuple: (Aid, Communication, Acquaintance, Cooperate, Assessment, Control),

\[ \text{Agent} ::= \text{Aid} \cdot \text{Communication} \cdot \text{Acquaintance} \cdot \text{Cooperate} \cdot \text{Assessment} \cdot \text{Control} \]

Definition 5. The agent oriented cooperative design system DS is a quintuple \( \{A, Tr, Ev, OR\} \), among which, \( A \) is the set of agents, \( Tr \) is the design Transaction, \( Ev \) is the set of events, \( OR \) is the organization (including CAD system and various users etc). Event is the system state change triggered by the following factors: users, CAD system and agents taking part in the design process. Here is the definition of event.

Definition 6. Event is a sextuple \( \{EID, ESTATE, EPRE, EAC, EPOST\} \), where \( EID \) is the name of the event; \( ESTATE \) is the present state of the design system; \( EPRE \) is the precondition triggering the event; \( EAC \) is the set of activated agent; \( EAC \) is the action taking by the agent; \( EPOST \) is the state caused by this event. Fig. 6 gives the event description.

\[ EID = 10012001-1001 \]

In order to access the turbine blade, section 3 gives the design requirement of the NC system, i.e., the cutter should move simultaneously in five axis, i.e. \( x, y, z, A \) and \( B \), and it is the design requirement. If the part is light, from the design constraints in section 3, the movement of the part has no limitations. So the content of \( EPRE \) is as the follows: The cutter has five DOFs, i.e. \( x, y, z, A, B \); and 2.) The property of the workpiece is light.

Definition 7. Transaction is a sequence of events; one transaction is an acceptable design solution satisfying the module constrains and the design constraints.

\[ T ::= \langle EID1 \rangle \langle EID2 \rangle \ldots \langle EIDn \rangle \]

5.2. Design Examples

NC system has a “C” list structure (Fig. 4 (c)), which takes the work-piece as the starting point and the cutter as the ending point [6] (Hatamura, 1995) or vice versa. Design events are as the follows. For the event \( EID1 \), it has the following initial conditions:

- \( EPRE \) has five DOFs, i.e. \( x, y, z, A \) and \( B \), and the part property is light.
- The agent state \( EAG \) is as Table 3.

![EAG state diagram](image)

Table 3. The EAG state.

<table>
<thead>
<tr>
<th>H_B</th>
<th>H_V</th>
<th>H_X</th>
<th>H_Y</th>
<th>R_Z</th>
<th>R_V</th>
<th>R_X</th>
<th>R_Y</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

- User state is 0, means waiting.
- CAD system state is 0, means waiting.
- From this starting point all the modules can be triggered, if \( R_Y \) is triggered which can realize the “y” directional translation. As a result, state of \( EPOST \) will change.
- \( EPOST \): cutter lost one DOF.

So for the second event \( EID2 \), the situation changed as the follows.

- \( EPRE \) has four DOFs, i.e. \( x, z, A \) and \( B \) and the part property is light.
- Agent state \( EAG \) from the module constrains, \( R_Y \) can not connect with \( H_B, H_V, R_V \) or \( R_Z \), so the \( EAC \) is like Table 4.
- User state is 0, means waiting.
- CAD system state is 0, means waiting.
At this time, if the H_Z is triggered, then,

- **EPOST**: cutter lost one DOF again.

<table>
<thead>
<tr>
<th>H B</th>
<th>H V</th>
<th>H X</th>
<th>H Y</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>H Z</td>
<td>R V</td>
<td>R X</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 4. The EAG state.

After six events the DOF of cutter reached zero, one design transaction finished. The result of one transaction is viewed as one acceptable solution. Various possible solutions will be recorded and compared. Among each transaction, CAD system will display the conceptual result, and the user can interact with the CAD system. Different results are compared and the better one is selected among the existing solutions with other designers. Fig. 4 (c) is one transaction, which denotes a 5 axes NC conceptual layout, though it may not be the optimal solution.

6. Conclusions

On the basis of de Boor’s recursive equation, formula of the normal unit vector of NURBS based on the controlling points is put forward. And the NC preliminary multi-solution design paradigm is put forward by the introduction of translating module, rotating module and base module. The NC system conceptual configuration design is based on the agent oriented cooperative design paradigm. Agent architecture and agent oriented creative design paradigm is put forward. The future work will focus the mechanism verification system etc.

Though the key theoretical issues put forward in this paper are now applied to the NC system conceptual design, there should been more industrial applications because of the prevalent proliferation nature of NURBS and agent.

Acknowledgments

The above-mentioned research work is supported by the Chinese NSFC, 61272017, P.R. China, and the Key Laboratory of High-efficiency and Clean Mechanical Manufacture at Shandong University, Ministry of Education.

References


2013 Copyright ©, International Frequency Sensor Association (IFSA). All rights reserved.
(http://www.sensorsportal.com)