สัมมนาวิชาการวิศวกรรมเครื่องกลแห่งประเทศไทยครั้งที่ 13
2-3 ธันวาคม 2542 โรงแรม รอยัลคลิฟบีชรีสอร์ท พัทยาใต้ ชลบุรี

การจำลองผิวลำเรือ

Surface Modeling of Ship Hulls

อุดมเกียรติ นนทแก้ว
ภาควิชาวิศวกรรมเครื่องกล คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าพระนครเหนือ
1518 ถ.พิบูลสงคราม บางซื่อ กรุงเทพ 10800
โทร 62(2) 913-2500 ต่อ 8320, โทรสาร 62(2) 587-0026, E-mail: unk@kmitnb.ac.tn

บทคัดย่อ

การคำนวณเชิงตัวเลขของปัญหาพลศาสตร์ของน้ำที่เกี่ยวข้องกับเรือส่วนใหญ่อิงอยู่กับทฤษฎีซิงกิวแลริทิ หาผลเฉลยด้วยวิธีมูลฐานขอบเขตโดยการกระจายองค์ประกอบซิงกิวล่าร์มูลฐานลงบนผิวของลำเรือส่วนที่จมน้ำ แล้วคำนวณหาการกระจายของความดันบนผิวนั้นๆอันนำไปสู่การคำนวณภาระที่กระทำต่อเรือ การจำลองผิวของเรือจึงเป็นขั้นตอนแรกของการศึกษาและมีความสำคัญยิ่ง อัลกอริธึมจำลองผิวลำเรือที่พัฒนาขึ้นในที่นี้ตั้งอยู่บนหลักการของการฟิตผิวผ่านกลุ่มข้อมูลในปริภูมิ 3 มิติ ผิวของเรือจะถูกแทนด้วยพาเนลซึ่งสร้างขึ้นโดยเลียนแบบกระบวนการทำ *Lofting* เริ่มต้นด้วยการสร้างโครงข่ายเส้นโค้งพาราเมตริก B-Spline กำลัง 3 ผ่านจุดที่ผู้ออกแบบเรือกำหนด แบ่งพาเนลให้ละเอียดด้วยการประมาณค่าภายใน จากนั้นเปลี่ยนรูปแบบการบรรยายเส้นโค้งจาก B-Spline กำลัง 3 ให้เป็นรูปแบบ Bézier กำลัง 3 ด้วยเทคนิคการซ้ำน็อท แล้วคำนวณเวคเตอร์สัมผัสเส้นโค้งที่ตำแหน่งจุดยอดพาเนลทุกจุดและสุดท้ายคำนวณเวกเตอร์ตั้งฉากกับผิว

Abstract

Most of numerical studies in naval hydrodynamics are based on the theory of singularities. The problem is solved by a boundary element method (BEM) where the singularity elements are distributed on wet surfaces of ship. The method allows determining pressure distribution on surfaces and therefore the sea-load exerting on the ship. This makes surface modeling of ship hull be a first important step. The present algorithm of surface modeling is based on surface fitting through a set of data in space. Hull surface is represented by panels, which are created following lofting procedure. The procedure starts with creating cubic B-Spline curve network through given data, refine the panel by interpolation and converts the curve representation from B-Spline to Bézier by multiplicity of knot. Finally, tangent vector at panel nodes and vector normal to the surface are calculated.

1. บทนำ

วัตถุประสงค์หลักของการพัฒนาอัลกอริธึมสำหรับจำลองผิวลำเรือคือการสร้างสมการคณิตศาสตร์แทนผิวเรือเพื่อให้เหมาะสำหรับใช้เป็นขอบเขตในการคำนวณการไหลของของไหลสมบูรณ์ผ่านเรือในการศึกษาปัญหาแรงต้านคลื่นและความคงตัวของเรือขณะแล่นไปบนคลื่น

โปรแกรมคำนวณการไหลของของไหลสมบูรณ์ผ่านเรือส่วนใหญ่ที่ใช้กันอยู่ในปัจจุบัน พัฒนาขึ้นบนพื้นฐานของการแทนรูปทรงของเรือด้วยโครงข่ายพาเนลและใช้แบบจำลองทางคณิตศาสตร์ที่สร้างขึ้นตามทฤษฎีซิงกิวแลริทิ โดยการแปลงปัญหาที่อธิบายในเทอมของสมการเชิงอนุพันธ์ย่อยไปเป็นปัญหาของการหาผลเฉลยของสมการอินทิกรัลด้วยการใช้วิธีกระจายซิงกิวแลริทิ (กระจาย Source-Doublet หรือ Doublet อย่างเดียว) ลงบนแต่ละพาเนลที่แทนผิวเรือเฉพาะส่วนที่จมน้ำ จากนั้นก็ใช้เงื่อนไขความเร็วของน้ำขนานกับผิวของวัตถุสร้างระบบสมการพีชคณิตเชิงเส้นที่มีตัวไม่รู้ค่าคือความเข้มของ Doublet ที่กระจายอยู่บนพาเนล ตัวอย่างโปรแกรมที่ใช้เทคนิคนี้คือโปรแกรมคำนวณหาแรงต้านของคลื่นที่ชื่อ REVA [1] และโปรแกรมทำนายพฤติกรรมการตอบสนองของเรือต่อคลื่นที่ชื่อ AQUAPLUS [2] และโปรแกรมคำนวณการไหลรอบเรือใบ ของ Ponizy [3] เป็นต้น

เส้นโค้งหรือผิวที่พบในงานออกแบบทางวิศวกรรมโดยทั่วไปแบ่งออกได้เป็นสามพวกพวกใหญ่ๆคือ พวกที่ต้องการความสวยงามเป็นหลักเช่นเส้นโค้งผิวของรถยนต์ พวกที่ต้องสนองฟังก์ชั่นการทำงานเป็นหลักเช่นปีกเครื่องบินที่ต้องทำหน้าที่สร้างแรงยก ลำเรือหรือลำตัวเครื่องบินที่ต้องมีรูปทรงที่ไม่ก่อให้เกิดแรงต้านการเคลื่อนที่มากเกินไป และพวกที่ต้องการทั้งความสวยงามและฟังก์ชั่นการทำงานเช่นยางรถยนต์เป็นต้น สำหรับงานวิจัยนี้จัดอยู่ในพวกที่สองและสาม โดยมีเป้าหมายคือต้องการใช้เป็นเครื่องมือช่วยในการออกแบบเรือและใบพัดสำหรับงานวิจัยของนักศึกษาระดับปริญญาโท-เอกของภาควิชาฯ ทั้งในแง่การคำนวณเชิงตัวเลขและการสร้างแบบจำลองย่อส่วนสำหรับการทดลอง

การจำลองผิวเริ่มต้นด้วยการกำหนดอนุกรมของเส้นโค้งในปริภูมิ 3 มิติขึ้นมาจำนวนหนึ่งแล้วใช้กระบวนการคำนวณประมาณค่าภายในสร้างผืนผิวขึ้นจากเส้นโค้งที่กำหนด เทคนิคนี้เป็นที่นิยมใช้กันอย่าง

แพร่หลายสำหรับสร้างวัตถุที่มีรูปทรงเรขาคณิตซับซ้อนจากเซ็ตของเส้นโค้งเช่นภาคตัดกระดูกงู จากแนวคิดเบื้องต้นดูเหมือนว่าจะสามารถสร้างรูปทรได้ง่าย แต่ในการปฏิบัติจริงแล้วต้องแก้ปัญหาขจัดอุปสรรคหลายประการสำหรับการบรรยายรูปทรงด้วยสมการคณิตศาสตร์ในรูปแบบพาราเมตริกเพื่อให้ผิวของวัตถุสนองต่อความต้องการฟังก์ชั่นการใช้งานทางวิศวกรรม และดูสวยงามในขณะเดียวกัน

สำหรับผืนผิวที่จะใช้ในงานออกแบบทางเครื่องกล อย่างน้อยๆ ต้องสนองความต้องการพื้นฐานต่อไปนี้

- รูปลักษณ์ของผืนผิวมองดูแล้วต้องเรียบเกลาสบายตา
- ลักษณะผิวต้องไม่เป็นลอนโดยไม่จำเป็น
- ผิวต้องไม่เปลี่ยนแปลงภายใต้ *Affine transformation*

ความต้องการสุดท้ายหมายถึงถ้าผืนผิวถูกเคลื่อนย้ายไปยังตำแหน่งอื่นๆ เรขาคณิตของผิวหลังการย้ายจะต้องเหมือนเดิม นอกจากนี้คุณสมบัติ *Affine invariance* ยังประกันว่าสัดส่วนของรูปทรงไม่เปลี่ยนไปแม้ว่าได้รับการออกแบบในระบบหน่วยวัดที่แตกต่างกัน เช่นใช้ระบบอังกฤษแทนระบบเมตริกเป็นต้น

2. คณิตศาสตร์ของเส้นโค้ง

วัตถุจำนวนมากมีรูปทรงเรียบเกลาโดยธรรมชาติฉะนั้นถ้าต้องการจำลองรูปทรงของวัตถุก็ต้องใช้เส้นสายที่มีธรรมชาติเรียบเกลา เราอาจกำหนดสมการคณิตศาสตร์ขึ้นมาอธิบายแทนรูปทรงที่จะออกแบบ หรือไม่ก็อาจนำแผ่นผิวหรือเส้นโค้งย่อยๆมาปะติดปะต่อ (ปะ-ชุน) ขึ้นมาเป็นรูปทรงที่ต้องการ

2.1 เส้นโค้ง Bézier กำลัง 3

รูปแบบของเส้นโค้งแบ่งตามรูปสมการของฟังก์ชั่นมูลฐาน เส้นโค้ง Bézier ตั้งตามชื่อของวิศวกรของบริษัทผลิตรถยนต์เรโนลด์ ผู้พัฒนาซอฟต์แวร์ *UNISURF* กล่าวกันว่าทฤษฎีเส้นโค้ง Bézier มีความสละสลวยและเข้ากับธรรมชาติเรขาคณิตของรูปทรงได้ดี ทำให้มีการใช้เส้นโค้งชนิดนี้กันมากที่สุดในงานคอมพิวเตอร์กราฟฟิก เส้นโค้ง Bézier กำลัง 3 กำหนดในเทอมของจุดควบคุม p_i ด้วยสมการ

$$Q(u) = \sum_{i=0}^{3} p_i B_{i,3}(u) \qquad (1)$$

ฟังก์ชั่นมูลฐาน, $B_{i,3}(u)$ เป็นพหุนาม *Bernstein* ซึ่งนิยามด้วย

$$B_{i,3}(u) = {}^3C_i u^i (1-u)^{3-i} \qquad (2)$$

เมื่อสัมประสิทธิ์ทวินาม ${}^3C_i = \dfrac{3!}{i!(n-i)!}$

เส้นโค้ง Bézier กำลัง 3 ในรูปที่ 1 มี p_0^0, p_1^0, p_2^0 และ p_3^0 เป็นจุดควบคุม เป็นที่น่าสังเกตว่าพิกัดของจุดควบคุม p_0^0 และ p_3^0 ทับกับจุดเริ่มต้นและจุดสิ้นสุดของเส้นโค้ง การเคลื่อนย้ายจุดควบคุมจะส่งผลให้รูปร่างเส้นโค้งเปลี่ยน ฟังก์ชันมูลฐาน $B_{i,3}(u)$ ทั้ง 4 ฟังก์ชั่นที่สัมพันธ์กับจุดควบคุมแต่ละจุด p_i นิยามด้วย

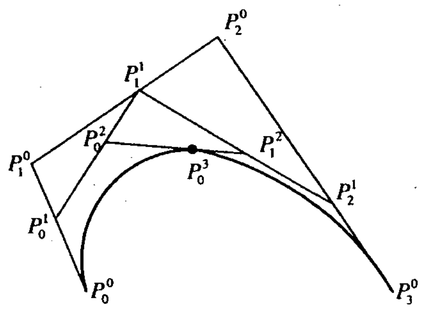
$$\begin{aligned} B_{0,3}(u) &= (1-u^2)^3 \\ B_{1,3}(u) &= 3u(1-u^2)^2 \\ B_{2,3}(u) &= 3u^2(1-u^2) \\ B_{3,3}(u) &= u^3 \end{aligned} \qquad (3)$$

สำหรับตัวแปรพาราเมตริก u ค่าใดๆตำแหน่งของเส้นโค้งจะคำนวณจากแต่ละฟังก์ชันมูลฐาน จุดควบคุม p_0 มีอิทธิพลต่อเส้นโค้งมากสุดที่ $u = 0$ (เนื่องจาก $B_{1,3}, B_{2,3}$ และ $B_{3,3}$ เป็นศูนย์ทั้งหมด) ขณะที่ u มีค่าเพิ่มขึ้น $p_0 B_{0,3}$ และ $p_1 B_{1,3}$ เป็นตัวหลักในการกำหนดรูปร่างเส้นโค้ง จุดควบคุม p_1 และ p_2 ที่อิทธิพลสูงสุดที่ $u = 1/3$ และ $u = 2/3$ ตามลำดับ ด้วยรูปแบบที่ฟังก์ชั่นมูลฐานมีอิทธิพลต่อรูปร่างเส้นโค้งตามที่กล่าวมาเป็นเหตุผลที่ทำให้เรียกว่า *Blending function*

นอกเหนือจากพหุนาม *Bernstein* แล้วเส้นโค้ง Bézier สามารถเขียนแทนด้วยสมการของ de Casteljau ซึ่งเป็นรูปสมการที่มีความเหมาะสมและมีประโยชน์กว่าในงานคอมพิวเตอร์กราฟฟิก ตำแหน่งต่างๆของเส้นโค้งคำนวณหาด้วยการประมาณค่าภายในเชิงเส้นซ้ำๆ

อัลกอริธึ่ม de Casteljau มีหลักการคือ ถ้า $p_0, p_1, \cdots, p_n$ เป็นจุดควบคุมเส้นโค้งจำนวน n จุดของ Bézier กำลัง n และให้ค่าประมาณภายในครั้งที่ r ของ $p_i^r(u) = (1-u)p_i^{r-1}(u) + up_{i+1}^{r-1}(u)$ เมื่อ $r = 1, \cdots, n$ และ $i = 0, \cdots, n-r$ โดยที่ $p_i^0(u) = p_i$ ดังนั้นจุดต่างๆบนเส้นโค้งสำหรับค่าพาราเมตริก u ใดๆคำนวณจาก

$$Q(u) = p_0^n \qquad (4)$$



รูปที่ 1 แสดงการประมาณค่าภายในของ de Casteljau

รูปที่ 1 แสดงหลักการของอัลกอริธึ่ม de Casteljau โดยเริ่มจาก $p_i^0, (i = 0,1,2,3)$ แบ่งด้านของ *control polygon* ออกเป็น 2 ส่วนที่ตำแหน่ง p_0^1, p_1^1 และ p_2^1 ด้วยสัดส่วน u กับ $1-u$ $(u = 0.6$ สำหรับกรณีของรูปที่ 1) ตัวดัชนีบนหมายถึงระดับการ recursive จากนั้นก็แบ่งด้านในระดับ p_i^2, $(i = 0,1)$ และหยุดแบ่งลงเมื่อเหลือเพียงหนึ่งด้าน จุดแบ่งของด้านสุดท้ายคือ p_0^3 ซึ่งเป็นจุดบนเส้นโค้ง $Q(u)$

อนุพันธ์อันดับหนึ่งของเส้นโค้ง Bézier กำลัง n เป็นเส้นโค้ง Bézier กำลัง $n-1$

$$\frac{dQ(0)}{du} = n(p_1 - p_0)$$
$$\frac{dQ(1)}{du} = n(p_n - p_{n-1}) \quad (5)$$

2.2 เส้นโค้ง B-Spline กำลัง 3

ต้นกำเนิดของเส้นโค้ง B-Spline มาจากไม้บรรทัดกระดูกงูที่ช่างเขียนแบบใช้ช่วยในการเขียนเส้นโค้ง เป็นการเขียนเซ็กเมนต์ย่อยๆ ต่อกันด้วยความต่อเนื่องแบบ C^2 ให้ Q_i เป็นเซ็กเมนต์ที่ i ของเส้นโค้ง B-Spline แต่ละเซ็กเมนต์มีจุดควบคุมอยู่ 4 จุดคือ p_i, p_{i+1}, p_{i+2} และ p_{i+3} ดังนั้น Q_{i+1} จะมี $p_{i+1}, p_{i+2}, p_{i+3}$ และ p_{i+4} เป็นจุดควบคุม จะเห็นได้ว่า p_{i+1}, p_{i+2} และ p_{i+3} เป็นจุดควบคุมร่วมของ Q_i

สำหรับเซ็กเมนต์ใดๆของเส้นโค้ง B-Spline $Q_i(u)$ นิยามในช่วง $0 \le u \le 1$ ด้วยฟังก์ชั่นมูลฐาน $B_k(u), (k = 0, \cdots, 3)$ และจุดควบคุม p_i, p_{i+1}, p_{i+2} และ p_{i+3} สามารถแทนด้วยสมการ

$$Q_i(u) = \sum_{k=0}^{3} p_{i+k} B_k(u) \quad (6)$$

ฟังก์ชั่นมูลฐาน

$$B_0(u) = \frac{(1+u)^3}{6}$$
$$B_1(u) = \frac{3u^3 - 6u^2 + 4}{6}$$
$$B_2(u) = \frac{-3u^3 + 3u^2 + 3u + 1}{6} \quad (7)$$
$$B_3(u) = \frac{u^3}{6}$$

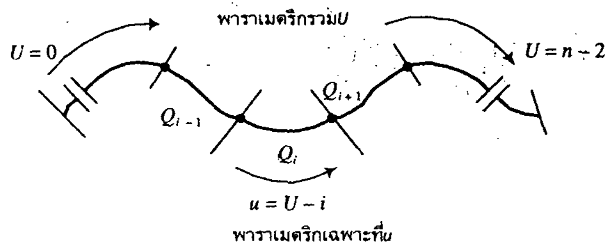
ต่อไปนี้จะมองในภาพกว้างของทฤษฎี *Spline* สมมุติว่ามีเส้นโค้งย่อยจำนวน $n-2$ เซ็กเมนต์ Q_i ดังนั้นจะมีจุดควบคุมจำนวน $n+1$ จุด $p_i, (i = 0, \cdots, n)$ เส้นโค้งย่อยทั้งหมดเชื่อมต่อกันเป็นเส้นเดียว ดูเหมือนจะสะดวกกว่าถ้าอธิบายเส้นโค้งที่ได้มาด้วยสมการเดียวบนพารามิเตอร์ตัวใหม่ U ที่ครอบคลุมตลอดทั้งเส้น แต่ยังคงให้ u เป็นพารามิเตอร์ของเส้นโค้งย่อยต่อไป โดยกำหนดให้ $U \in [0, n-2]$ ขณะที่ $u \in [0,1]$ สำหรับเซ็กเมนต์ที่ i พารามิเตอร์ $u = U - i$ ดังแสดงในรูปที่ 2

การเขียนสมการด้วยพารามิเตอร์รวมจะทำให้ทราบอิทธิพลของจุดควบคุมจุดหนึ่งต่อเส้นโค้งทั้งเส้น สมการ (6) บอกว่าเส้นโค้งย่อยแต่ละเส้นกำหนดด้วยจุดควบคุม 4 จุด ดังนั้นจุดควบคุมใดๆ (ยกเว้นจุดควบคุมที่ปลายทั้งสองข้างของเส้นโค้ง) จะส่งอิทธิพลต่อเส้นโค้งย่อย 4 เซ็กเมนต์ เช่นจุด p_i อยู่ในสมการของเซ็กเมนต์ $Q_{i-3}, Q_{i-2}, Q_{i-1}$ และ Q_i โดยคูณอยู่กับฟังก์ชั่นมูลฐาน B_3, B_2, B_1 และ B_0 ตามลำดับ เพื่อความสะดวกในการคำนวณจึงรวมฟังก์ชั่นมูลฐานเข้าด้วยกันเป็นฟังก์ชั่นมูลฐานเฉพาะของจุดควบคุม p_i แล้วจัดรูปสมการเสียใหม่เป็น

$$Q(u) = \sum_{i=0}^{n} p_i N_i(U) \quad (8)$$

ฟังก์ชั่นมูลฐานเฉพาะ

$$N_i(U) = \begin{cases} B_3(u_0) & i-3 \le U < i-2 \\ B_2(u_1) & i-2 \le U < i-1 \\ B_1(u_2) & i-1 \le U < i \\ B_0(u_3) & i \le U < i+1 \end{cases} \quad (9)$$



รูปที่ 2 พารามิเตอร์เฉพาะที่และพารามิเตอร์รวมสำหรับเซ็กเมนต์ของเส้นโค้ง B-Spline

ฟังก์ชั่น $N_i(U)$ รับพารามิเตอร์รวม U แล้วเปลี่ยนเป็นพารามิเตอร์ย่อย u_j ประจำเซ็กเมนต์ เมื่อ $u_j = U - i + 3 - j, (j = 0, \cdots, 3)$ การอธิบายเส้นโค้งดังที่กล่าวมานี้เรียกว่า *Uniform B-Splines* เพราะว่าช่วงค่าพารามิเตอร์รวม U สำหรับแต่ละเซ็กเมนต์มีค่าเท่ากันเช่นจุดปลายจะอยู่ที่ $[0,1,\ldots,m]$ ค่าเหล่านี้เรียกว่าน็อท รวมกันทั้งหมดเรียกว่าเวกเตอร์น็อท โดยทั่วไปแล้วช่วงค่าพารามิเตอร์รวมไม่จำเป็นต้องเท่ากัน โดยให้ช่วงน็อทอยู่ในรูป $t_0 < t_1 < \ldots < t_n$ และถ้าช่วงห่างระหว่างน็อทไม่เท่ากันจะเรียกว่า *Non-uniform B-Spline*

สำหรับ *Spline* อันดับ k (กำลัง $k-1$) มี $N_{i,k}(u)$ เป็นฟังก์ชั่นมูลฐานนิยามแบบ *recursion* ในช่วงน็อท $[t_i, t_{i+k}]$ ตามอัลกอริธึมของ Cox de Boor ซึ่งเทียบเท่ากับอัลกอริธึมของ de Casteljau ในเส้นโค้ง Bézier ดังนี้

$$N_{i,1}(u) = \begin{cases} 1 & \text{if } t_i \le u < t_{i+1} \\ 0 & \text{otherwise} \end{cases}$$
$$N_{i,k}(u) = \frac{(u - t_i) N_{i,k-1}(u)}{t_{i+k-1} - t_i} + \frac{(t_{i+k} - u) N_{i+1,k-1}(u)}{t_{i+k} - t_{i+1}} \quad (10)$$

ข้อพึงสังเกตคือฟังก์ชั่นมูลฐานขึ้นกับค่าน็อทที่ตำแหน่งต่างๆบนช่วง $[t_0, \ldots, t_n]$ และมีคุณสมบัติดังต่อไปนี้

1. ไม่เป็นค่าติดลบ $N_{i,k}(u) \ge 0$ สำหรับทุกๆค่า i, k, u
2. ผลรวมที่จุดใดๆเป็นหนึ่ง $\sum_{i=0}^{n} N_{i,k}(u) = 1$
3. การคำนวณค่า $N_{i,k}(u)$ เกี่ยวข้องกับน็อท เฉพาะในช่วงตั้งแต่ t_i ถึง t_{i+k} เท่านั้น อาจกล่าวได้ว่า *support* ของ ฟังก์ชั่นมูลฐาน $N_{i,k}(u)$ คือ $[t_i, t_{i+k}]$
4. ให้ถือว่า 0/0 (ศูนย์หารด้วยศูนย์) มีค่าเท่ากับ 0 เป็นผลให้สามารถกำหนดค่าน็อท ซ้ำกันได้ (multiple knots)

ความสัมพันธ์ของเวกเตอร์น็อทกับจุดควบคุมหาจากการวิเคราะห์เส้นโค้ง B-Spline

$$Q(u) = \sum_{i=0}^{n} p_i N_{i,k}(u) \quad (11)$$

เนื่องจากแต่ละจุดของจุดควบคุมทั้ง $n+1$ จุดมีฟังก์ชันมูลฐานคู่กันอยู่ซึ่งมีด้วยกันทั้งหมด $n+1$ ฟังก์ชั่น แต่ละฟังก์ชั่นมูลฐาน $N_{i,k}(u)$ มี $[t_i,t_{i+k}]$ เป็น *Support* ดังนั้นจึงต้องการ $n+k+1$ น็อทรวมตัวกันเป็นเวกเตอร์ $[t_0,\cdots,t_{n+k}]$ โดยทั่วไปแล้วจำนวนของน็อทในเวกเตอร์จะเท่ากับจำนวนจุดควบคุมบวกอันดับของพหุนาม (ตัวอย่างเช่นพหุนามกำลังสามมีอันดับ $k=4$) คุณลักษณะของ B-Spline แบบไม่สม่ำเสมอจะเป็นไปตามคุณลักษณะของฟังก์ชั่นมูลฐาน

1.คุณสมบัติเฉพาะที่ นั่นคือ $N_{i,k}(u)$ มีค่าเป็นศูนย์ภายนอกช่วง $[t_i,t_{i+k}]$ ทำให้จุดควบคุม p_i ไม่มีผลต่อเส้นโค้งภายนอกช่วงที่กล่าวมา เคลื่อนย้ายจุดควบคุม p_i ส่งผลให้รูปร่างของเส้นโค้งเปลี่ยนเฉพาะในช่วงน็อท $[t_i,t_{i+k}]$

2.คุณสมบัติต่อเนื่องสำหรับเส้นโค้ง B-Spline ไม่มีปัญหาเรื่องความต่อเนื่องระหว่างเซ็กเมนต์ เป็นที่ทราบกันว่าเส้นโค้งกำลังสามมีความต่อเนื่อง C^0,C^1 และ C^2

ย้อนกับไปพิจารณาสมการ (11) พบว่าเซ็กเมนต์เส้นโค้งถูกนิยามบนหนึ่งช่วงจุดควบคุมและฟังก์ชั่นมูลฐานจะเกี่ยวข้องจำนวน k ช่วง ผลที่ตามมาคือมีจำนวนฟังก์ชั่นมูลฐานได้มากสุดจำนวน k ฟังก์ชั่นที่ไม่เป็นศูนย์บนช่วงใดๆที่กำหนด เพราะฉะนั้นจึงต้องการจุดควบคุมจำนวน k จุดสำหรับนิยามเส้นโค้งแต่ละเซ็กเมนต์ กลุ่มจุดควบคุมกลุ่มแรกคือ $(p_0,\cdots,p_{k-1})$ และมี $(p_{n-k+1},\cdots,p_n)$ เป็นกลุ่มสุดท้าย ดังนั้นมีจำนวนเส้นโค้งทั้งหมด $n-k+2$ เซ็กเมนต์ หมายความว่าต้องใช้ช่วงน็อทจำนวน $n-k+2$ ช่วงที่ไม่ซ้ำกันนั่นคือ $n-k+3$ น็อท เนื่องจากในเวกเตอร์น็อทต้องการสมาชิกจำนวน $n+k+1$ จึงจำเป็นต้องกำหนดเพิ่มเติมอีก $(n+k+1)-(n-k-3)=2k-2$ น็อท

ชนิดของเวกเตอร์น็อทที่นิยมใช้กันโดยทั่วไปคือแบบไม่เป็นคาบด้วยการซ้ำน็อทตัวแรกและตัวสุดท้าย $k-1$ ครั้ง ซึ่งเขียนในรูป

$$T=(t_0,\cdots,t_{n+k})=(0,\cdots,0,t_k,\cdots,t_n,1,\cdots,1) \quad (12)$$

ในทางปฏิบัติมักกำหนดสมาชิกของเวกเตอร์น็อทแบบสม่ำเสมอในแบบ

$$t_i=\begin{cases}0 & i=0,\cdots,k-1\\ i-k+1 & i=k,\cdots,n-1\\ n-k+2 & i=n,\cdots,n+k-1\end{cases} \quad (13)$$

ค่าน็อทซ้ำกัน k ครั้งที่ปลายทั้งสองข้างของเส้นโค้งเป็นผลให้ที่ตำแหน่งนั้นเส้นโค้งซ้ำกับจุดควบคุม นั่นคือ $Q(t_{k-1})=p_0$ และ $Q(t_{n+1})=p_n$ เส้นโค้งไม่นิยามนอกช่วง $[t_{k-1},t_{n+1}]$ เนื่องจาก $t_j=t_{j+1}$ ทำให้เซ็กเมนต์ Q_j จะยุบลงเหลือเพียงจุดเดียว ซึ่งกลายเป็นเอกลักษณ์ที่สำคัญในการประยุกต์ใช้ B-Spline

เป็นไปได้ที่จะเพิ่มจำนวนน็อทและจุดควบคุมในที่ใดที่หนึ่งตลอดทั้ง B-Spline โดยไม่ทำให้รูปร่างเส้นโค้งเปลี่ยนแปลง การเพิ่มจำนวนน็อท ให้ผลสองประการ

1. การเพิ่มน็อทใหม่เข้าไปในเวกเตอร์น็อท (knot vector) ช่วยให้การปรับแต่งรายละเอียดของรูปทรงเส้นโค้งทำได้ง่ายขึ้น

2.การเพิ่มน็อทใหม่ที่มีค่าซ้ำกับน็อทที่มีอยู่แล้วจะลดอันดับความต่อเนื่องของรอยต่อเซ็กเมนต์ลงเช่น $Q(t_i)$ มีอนุพันธ์ต่อเนื่อง $k-m-1$ ครั้งที่ตำแหน่งน็อท t_i ที่มีค่าซ้ำกัน m ครั้ง $t_i=t_{i+1}=\cdots=t_{i+m-1}$ สำหรับเวกเตอร์น็อทแบบไม่เป็นคาบมีน็อทซ้ำกัน k ครั้งที่จุดปลายทั้งสองข้างของเส้นโค้งทำให้เส้นโค้งทับกับจุดควบคุม โดยทั่วไปถ้าน็อท t_i ใดๆซ้ำกัน k ครั้งทำให้จุดบนเส้นโค้ง $Q(t_i)$ เป็นจุดเดียวกับจุดควบคุม ข้อพึงสังเกตคือน็อทซ้ำกัน k ครั้งส่งผลให้แบ่งแยกเส้นโค้ง B-Spline เป็นสองเส้น

กระบวนการเพิ่มน็อทลงในเวกเตอร์น็อทมีขั้นตอนดังนี้ ให้เส้นโค้ง B-Spline ที่กำหนดด้วยสมการ (11) มีเวกเตอร์น็อทเป็น $[t_0,\cdots,t_{n+k}]$ ถ้าต้องการเพิ่มน็อทใหม่ $t'\in[t_0,t_{n+k}]$ โดยที่ $t_j\le t'\le t_{j+1}$ เส้นโค้งเดิมกำหนดด้วยจุดควบคุมใหม่ p_i'

$$Q(u)=\sum_{i=0}^{n+1}p_i'N_{i,k}(u) \quad (14)$$

เมื่อ $p_0'=p_0$, $p_{n+1}'=p_n$ และ $p_i'=(1-a_i)p_{i-1}+a_ip_i$

$$a_i=\begin{cases}1 & i=1,\ldots,j-k\\ \dfrac{t'-t_i}{t_{i+k}-t_i} & i=j-k+1,\ldots,j\\ 0 & i=j+1,\ldots,n\end{cases} \quad (15)$$

2.3 ความสัมพันธ์ระหว่างเส้นโค้ง B-Spline กับ Bézier

พิจารณา B-Spline อันดับ k ที่เวกเตอร์น็อทมีสมาชิก $2k$ ตัว โดยไม่มีน็อทภายใน (interior knot) คือมีเฉพาะที่ปลายทั้งสองข้างของเส้นโค้ง กำหนดด้วย

$$t_i=\begin{cases}0 & 0\le i<k\\ 1 & k\le i<2k\end{cases} \quad (16)$$

$N_{i,k}(u)$ ไม่เป็นศูนย์เฉพาะที่ $u\in[i,i+k]$ มีช่วง $[k-1,k]$ ประกอบอยู่ จากจุดนี้พออนุมานได้ว่า $i\le k-1$ และ $i+k\ge k$ จากสมการ (17) ได้ $t_i=0$ และ $t_{i+k}=1$ สำหรับทุกๆค่า i,k เมื่อแทนลงในอัลกอริธึม Cox de Boor จะได้

$$N_{i,k}(u)=uN_{i,k-1}(u)+(1-u)N_{i+1,k-1}(u) \quad (17)$$

ซึ่งตรงกับอัลกอริธึม de Casteljau สำหรับเส้นโค้ง Bézier จากนี้เราสามารถสรุปได้ว่าเส้นโค้ง B-Spline ที่ไม่มีน็อทภายในคือ Bézier

ต่อไปนี้เราสามารถนำเวกเตอร์น็อทใดๆมาเพิ่มน็อทใหม่เข้าไปให้แต่ละน็อทซ้ำกัน k ครั้ง แยกเส้นโค้งออกเป็นเซ็กเมนต์ย่อยๆต่อเนื่องกันแต่ละเซ็กเมนต์มีเวกเตอร์น็อทที่ไม่มีน็อทภายใน (มีแค่ $2k$ น็อท) คล้ายเส้นโค้ง Bézier ด้วยวิธีการนี้สามารถเปลี่ยนเส้นโค้ง B-Spline ให้กลายเป็นเส้นโค้ง Bézier

2.4 การประมาณค่าภายใน

การประมาณค่าภายในถูกใช้เป็นเครื่องมือสำหรับจำลองรูปร่างหรือพื้นผิวในงานคอมพิวเตอร์กราฟฟิก ตัวอย่างเช่นการฟิตผิวจากข้อมูลที่ได้จากการทำดิจิไตเซอร์หรือสแกนเนอร์ด้วยวิธีการประมาณค่าภายใน ทำให้สามารถสร้างสมการขึ้นมาแทนวัตถุจริงได้

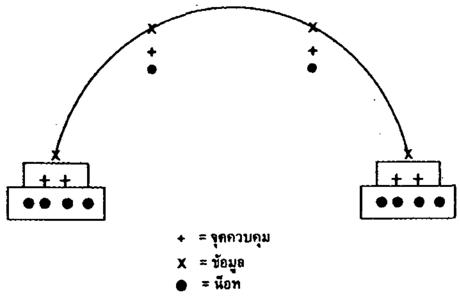
ขั้นตอนการประมาณค่าภายในประกอบด้วย การหาเส้นโค้งหรือผิวที่ผ่านข้อมูลที่กำหนด โดยทั่วไปมักใช้ Spline กำลัง 3 เนื่องจากให้ความต่อเนื่องถึงอนุพันธ์อันดับสอง มีความยืดหยุ่นสูงและคณิตศาสตร์ไม่ซับซ้อนมากนัก สมมุติว่าทราบค่าเวกเตอร์น็อทชนิดไม่เป็นคาบ ในที่

นี้จะใช้ B-Splines กำลัง 3 ($k=4$) เนื่องจากเวกเตอร์น็อทชนิดไม่เป็นคาบก่อให้เกิดความยุ่งยากที่จุดปลายทั้งสองข้างของเส้นโค้ง สมมุติทราบค่าจุดควบคุมซึ่งมีอยู่จำนวน $n+1$ จุด ตำแหน่งจุดปลายทั้งสองของเส้นโค้งคือ $p_0 = Q(t_3)$ และ $p_n = Q(t_n+1)$ เส้นโค้งนี้นิยามเฉพาะในช่วง $[t_3, t_{n+1}]$ เท่านั้น นั่นคือเส้นโค้งประกอบด้วยเซ็กเมนต์ย่อย $n-2$ เซ็กเมนต์ มีข้อมูล จำนวน $n-1$ จุดบนเส้นโค้ง จุดเหล่านี้ก็คือข้อมูลที่นำมาทำการประมาณค่าภายในนั่นเอง ดังนั้นปัญหาการประมาณค่าภายในก็คือการคำนวณหาจุดควบคุมจำนวน $n+1$ ที่บังคับให้เส้นโค้งพาดผ่านจุดข้อมูลทั้ง $n-1$ จุด จากเวกเตอร์น็อทที่สมมุติขึ้นและเซ็ทของข้อมูล X_i, $i=0,1,\cdots,n-2$ ที่เส้นโค้งต้องพาดผ่าน $n-1$ จุด ถึงตอนนี้ระบบสมการไม่สามารถให้ผลเฉลยได้เนื่องจากมีตัวไม่รู้ค่ามากกว่าจำนวนสมการ จึงจำเป็นต้องหาเงื่อนไขเพิ่มเติมอีกสองเงื่อนไขโดยใช้จุดเริ่มต้นและจุดสิ้นสุดของเส้นโค้ง ทำผลเฉลยให้ *Unique* โดยให้จุดปลายทั้งคู่ซ้ำกันสองครั้ง ระบบสมการคือ

$$Q(t_{i+3}) = X_i,\quad i=0,1,\cdots,n-2 \tag{18}$$

และเงื่อนไขที่ปลายเส้นโค้งทั้งสองข้าง

$$p_0 = p_1,\quad p_n = p_{n-1} \tag{19}$$



รูปที่ 3 การประมาณค่าภายในด้วย B-Spline

สมการ (19) บอกเราว่าต้องคำนวณฟังก์ชั่นมูลฐานที่ตำแหน่งค่าน็อท ฟังก์ชั่น $N_{i,1}(t_j)$ ไม่เป็นศูนย์เฉพาะที่ $i=j$ เท่านั้น เนื่องจาก $N_{i,4}(u)$ มี (t_i, t_{i+4}) เป็น *support* ดังนั้นเราต้องคำนวณแค่ 5 น็อทภายใน *support* เท่านั้น และมีเพียงสามน็อทเท่านั้นที่ให้ค่าไม่เป็นศูนย์ ซึ่งประกอบด้วย

$$\begin{aligned}
N_{i,4}(t_{i+1}) &= \frac{(t_{i+1}-t_i)^2}{(t_{i+}\ -t_i)(t_{i+3}-t_i)} \\
N_{i,4}(t_{i+2}) &= \frac{(t_{i+2}-t_i)(t_{i+3}-t_{i+2})}{(t_{i+3}-t_{i+1})(t_{i+3}-t_i)} + \frac{(t_{i+4}-t_{i+2})(t_{i+2}-t_{i+1})}{(t_{i+4}-t_{i+1})(t_{i+3}-t_{i+1})} \\
N_{i,4}(t_{i+3}) &= \frac{(t_{i+4}-t_{i+3})^2}{(t_{i+4}-t_{i+1})(t_{i+4}-t_{i+2})}
\end{aligned} \tag{20}$$

พิกัดเส้นโค้งที่ตำแหน่ง $Q(t_i)$ กำหนดด้วยสมการ

$$Q(t_i) = N_{i-3,4}(t_i)p_{i-3} + N_{i-2,4}(t_i)p_{i-2} + N_{i-1,4}(t_i)p_{i-1} \tag{21}$$

หรือ

$$Q(t_{i+3}) = N_{i,4}(t_{i+3})p_i + N_{i+1,4}(t_{i+3})p_{i+1} + N_{i+2,4}(t_{i+3})p_{i+2} \tag{22}$$

สำหรับจุดข้อมูลใดๆบนเส้นโค้งสามารถเขียนแทนด้วย

$$X_i = \alpha_i p_i + \beta_i p_{i+1} + \gamma_i p_{i+2} \tag{23}$$

เมื่อ

$$\begin{aligned}
\alpha_i &= N_{i,4}(t_{i+3}) \\
\beta_i &= N_{i+1,4}(t_{i+3}) \\
\gamma_i &= N_{i+2,4}(t_{i+3})
\end{aligned} \tag{24}$$

ระบบสมการที่ให้ความสัมพันธ์ระหว่างจุดควบคุมกับข้อมูลที่นำมาฟิตเส้นโค้งสามารถเขียนในรูปเมตริกคือ

$$\begin{bmatrix} 1 & & & \\ \alpha_1 & \beta_1 & \gamma_1 & \\ & & & \\ & & & \\ & \alpha_{n-3} & \beta_{n-3} & \gamma_{n-3} \\ & & & 1 \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \\ \cdot \\ \cdot \\ \cdot \\ p_{n-2} \\ p_{n-1} \end{bmatrix} = \begin{bmatrix} X_0 \\ X_1 \\ \cdot \\ \cdot \\ \cdot \\ X_{n-3} \\ X_{n-2} \end{bmatrix} \tag{25}$$

สมการ (24) ได้ผลเฉลยเป็นเซ็ทของจุดควบคุม $[\mathbf{P}]$ ที่บังคับให้เส้นโค้ง B-Spline กำลัง 3 พาดผ่านเซ็ทของข้อมูล, $[\mathbf{X}]$

3. การจำลองผิวด้วยพาเนล

การจำลองผิวด้วยพาเนลคือการสร้างแผ่นผิวจำนวนมากมาแทนตำแหน่งข้อมูลในปริภูมิ 3 มิติ การฟิตผิวพาราเมตริกผ่านเซ็ทของข้อมูลด้วยการประมาณค่าภายในทำได้ยาก ดังนั้นในที่นี้จะจำลองผิวรูปทรงด้วยการประยุกต์ใช้วิธีการสร้างเรืออันเก่าแก่และปฏิบัติกันมานานที่เรียกว่าการทำ *lofting* เป็นแนวทางในการจำลองผิวด้วยพาเนล

3.1 การสร้างเมชด้วย B-Spline

ขั้นตอนแรกจะสร้างโครงข่ายของเส้นโค้งที่ผ่านจุดข้อมูลที่กำหนดตามแบบอย่างกระบวนการทำ *lofting* สร้างกลุ่มเส้นโค้งในทิศทางหนึ่งสมมุติให้เป็นทิศทางของพาราเมตริก u ซึ่งตรงกับภาคตัด (กระดูกงูในกรณีของเรือ) จากนั้นสร้างเส้นโค้งกลุ่มที่สองให้พาดผ่านเส้นโค้งกลุ่มแรกในทิศทางพารามิเตอร์ v สัมพันธ์กับแนวยาวของเรือ เส้นโค้งทั้งสองกลุ่มเชื่อมต่อกันที่ตำแหน่งจุดข้อมูลที่ต้องการให้เส้นโค้งผ่าน ผสานกันเป็นโครงข่ายเส้นโค้ง กระบวนการประมาณค่าภายในที่ใช้สร้างโครงข่ายเส้นโค้งจากข้อมูลตำแหน่งที่ต้องการให้เส้นโค้งผ่านจำแนกได้เป็นขั้นตอนต่อไปนี้

1. แยกข้อมูลที่ต้องการให้เส้นโค้งพาดผ่านออกเป็นกลุ่มๆ แต่ละกลุ่มไม่จำเป็นต้องมีจำนวนข้อมูลเท่ากัน สร้างเส้นโค้งให้เป็นฟังก์ชั่นของพาราเมตริก u ให้พาดผ่านข้อมูลต่างๆ ที่แบ่งไว้ของแต่ละกลุ่ม ซึ่งเส้นโค้งแต่ละเส้นจะมีจุดควบคุมและเวกเตอร์น็อทเป็นของตัวเองไม่ซ้ำกัน เรียกกลุ่มเส้นโค้งเหล่านี้ว่า *U-curves*
2. สร้างเส้นโค้งในทิศทางพาราเมตริก v ซึ่งตั้งฉากกับทิศทาง u ด้วยข้อมูลที่ได้จากการประมาณภายในของ U-curves แต่ละเส้น เส้นโค้ง V-

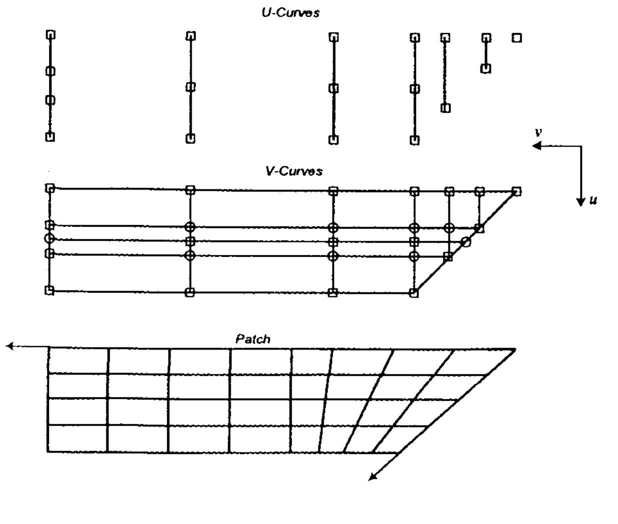
curves ที่สร้างขึ้นใหม่มีจำนวนมากเท่าที่ต้องการ แต่ละเส้นมีเวกเตอร์น็อทและจุดควบคุมเป็นของตัวเองไม่ซ้ำกัน

3. จากนิยามของแผ่นผิวพาราเมตริก $Q(u,v)$ ต้องการเวกเตอร์น็อทในแกนพาราเมตริก u และ v อย่างละหนึ่งเวกเตอร์เท่านั้น ดังนั้นจึงต้องกำหนดเวกเตอร์น็อทในแกน v ใหม่ให้มีจำนวนสอดคล้องกับจำนวนเส้นโค้ง U-curves ที่ต้องการในตอนสุดท้าย (สำหรับข้อมูล n ตัว ต้องการ $n+1$ จุดควบคุมและ $n+2+4$ น็อท สำหรับสร้างเส้นโค้ง B-Spline กำลัง 3) กระทำการแก้ไขเปลี่ยนแปลงจุดควบคุมของเส้นโค้งแต่ละเส้นให้สอดคล้องกับเวกเตอร์น็อทใหม่ด้วยกระบวนการเพิ่มน็อท

4. ถึงตอนนี้ U-curves เริ่มต้นที่สร้างในขั้นตอนที่ 1 อาจจะไม่ตัดกับ V-curves ที่ตำแหน่ง v คงที่ จึงต้องสร้างเส้นโค้งในทิศทางพาราเมตริก u ขึ้นใหม่อีกครั้ง ข้อมูลที่นำมาใช้ฟิตเส้นโค้ง U-curves คือตำแหน่งของเส้นโค้ง V-curves แต่ละเส้นที่จุดน็อท $v = t_i$

5. รูปที่ (4) แสดงโครงข่ายของเส้นโค้งและความสัมพันธ์กับข้อมูลเริ่มต้น เส้นโค้งสองกลุ่ม (u คงที่กับ v คงที่) ตัดกันเกิดเป็นแผ่นผิวสี่เหลี่ยมเล็กๆหรือพาเนลในปริภูมิ u-v จำนวน $(nU-1)*(nV-1)$ เมื่อ nU และ nV เป็นจำนวนเส้นโค้งในทิศทางของพาราเมตริก u และ v ตามลำดับ

6. เปลี่ยนเส้นโค้ง B-Spline แต่ละเส้นที่สร้างในขั้นตอนที่ 1 ถึง 4 ให้เป็นเส้นโค้ง Bézier โดยใช้กระบวนการเพิ่มน็อทที่ให้แต่ละน็อทมีค่าซ้ำกัน k ครั้ง ($k=4$ สำหรับเส้นโค้งกำลัง 3) ต่อแต่นี้ไปโครงข่ายเส้นโค้งจะถูกแทนด้วยเซ็กเมนต์ Bézier ย่อยๆที่ไม่ต่อเนื่อง



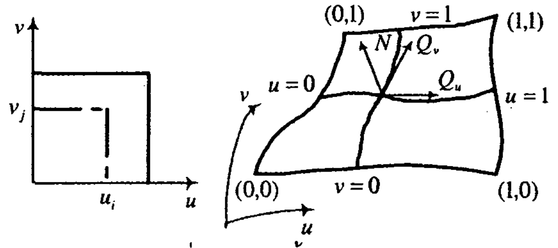
รูปที่ 4 โครงข่ายของเส้นโค้งและความสัมพันธ์กับข้อมูลเริ่มต้น

3.2 เวกเตอร์ตั้งฉากกับแผ่นผิว

จากนิยามเวกเตอร์สัมผัส Q_u และ Q_v สัมผัสกับผิว $Q(u,v)$ ด้วย ฉะนั้นเวกเตอร์ตั้งฉากกับผิว N เป็นเวกเตอร์โปรดักต์ของ

$$N(u,v) = Q_u(u,v) \times Q_v(u,v) \tag{26}$$

ตัวอย่างกรณีพาเนลห้อมล้อมด้วยเส้นโค้ง Bézier เช่นเราทราบว่า



รูปที่ 5 เวกเตอร์ตั้งฉากกับผิว

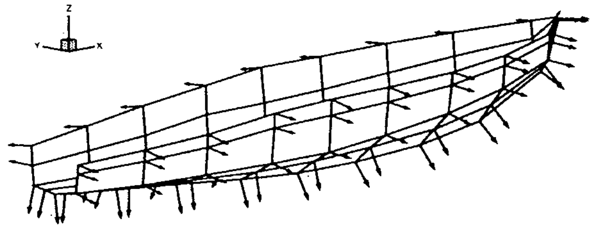
$$Q_u(0,0) = 3(p_{10} - p_{00}) \tag{27}$$

$$Q_v(0,0) = 3(p_{01} - p_{00}) \tag{28}$$

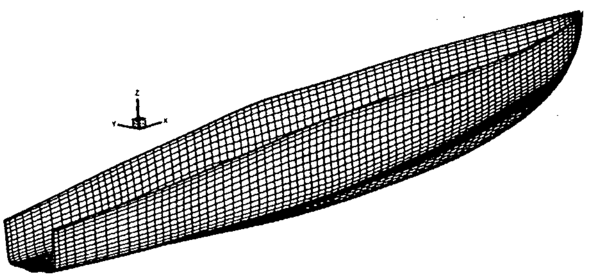
4. ตัวอย่างประกอบ

4.1 เรือแคทะมะแรน *Gemini*

กรณีตัวอย่างที่หนึ่งเป็นการสร้างพาเนลสำหรับจำลองเรือแคทะมะแรนที่มีชื่อว่า *Gemini* จากข้อมูลที่เป็นหน้าตัดกระดูกงูจำนวน 10 ภาคตัด แต่ละภาคตัดมีจำนวนข้อมูลเท่ากัน (10 จุดเฉพาะในระนาบสมมาตร) รูปที่ 6 แสดงรูปร่างเรืออย่างคร่าวๆที่สร้างจากข้อมูลภาคตัดกระดูกงูและเวคเตอร์ตั้งฉากที่คำนวณได้ รูปที่ 7 แสดงเรือ *Gemini* ที่แทนด้วยพาเนลจำนวน 4266 พาเนลจากการประมาณค่าภายใน



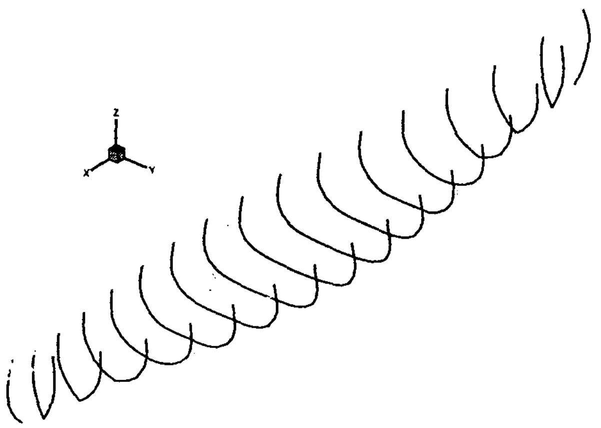
รูปที่ 6 พาแนลสร้างจากกระดูกงูและเวกเตอร์ตั้งฉาก



รูปที่ 7 เรือ Gemini ที่แทนด้วยพาเนลจำนวน 4266 พาเนล

4.2 เรือคะนู

กรณีตัวอย่างที่สองเป็นการสร้างพาเนลสำหรับจำลองเรือคะนูที่มีชื่อว่า *Peterborough canoe* [12] ข้อมูลที่เป็นหน้าตัดกระดูกงูจำนวนในรูปที่ 8 แต่ละหน้าตัดมีจำนวนข้อมูลไม่เท่ากัน เรือรูปร่างสมบูรณ์แทนด้วยพาเนลจำนวน 4524 พาเนล แสดงในรูปที่ 9



รูปที่ 8 เส้นโค้งกระดูกงูของเรือคะนู

5. สรุปผลการวิจัย

อัลกอริธึมสำหรับจำลองผิวของเรือที่พัฒนาขึ้นในที่นี้ตั้งอยู่บนหลักการที่ต้องกำหนดอนุกรมของเส้นโค้งในปริภูมิ 3 มิติที่เป็นตัวแทนของเรือที่ต้องการจำลองขึ้นจำนวนหนึ่ง แล้วใช้กระบวนการคำนวณสร้างผืนผิวขึ้นจากการประมาณค่าภายในด้วยพาราเมตริก B-Spline กำลัง 3 จากนั้นก็เปลี่ยนเป็นพาราเมตริก Bézier กำลัง 3 ด้วยกระบวนการซ้ำน็อทเพื่อคำนวณเวกเตอร์สัมผัสเส้นโค้งเพื่อใช้สำหรับคำนวณเวคเตอร์ตั้งฉาก ให้ผลการคำนวณที่มีความถูกต้องแม่นยำสามารถนำไปใช้งานได้ตามวัตถุประสงค์เบื้องต้น นอกจากนี้ยังสามารถนำไปใช้จำลองผิวใบพัดเรือหรือกังหันลมได้

กิตติกรรมประกาศ

งานวิจัยชิ้นนี้ได้รับการสนับสนุนทางการเงินจากสำนักงานพัฒนาวิทยาศาสตร์และเทคโนโลยีแห่งชาติ (สวทช) ในโครงการทุนสนับสนุนนักวิจัยใหม่ รหัส CO-B-06-44-21-002 จึงขอขอบพระคุณมา ณ ที่นี้

เอกสารอ้างอิง

[1] G. Delhommeau et J.J. Maisonneuve, *Extensions du code de calcul de resistancede vagues* REVA: *prise en compte des effets de fond et de portance*, Compte-Rendu des 1eres Journees de l'Hydrodynamique, Nantes, pp. 248-254, 1987.

[2] G. Delhommeau, *Les problemes de diffraction-radiation et de resistance de vagues: etude theorique et resolution numerique par la methode des singularites*, These de Doctorat-es Sciences, Ecole Centrale de Nantes, 1987.

[3] B. Ponizy, *Calcul rapide de l'ecoulement non-symtrique autour de navires a l'aide de singularites de* Kelvin: *Aplication au cas des voiliers*, These de Doctorat, Universite de Poitiers, Juin 1996.

[4] B. Allessandrini, *Maillage de carenes automatique*, Ecole Centrale de Nantes, 1996.

[5] David F. Rogers, Steven G. Satterfield and Francisco A. Rodriguez, *Ship Hulls, B-Spline Surfaces, and CAD/CAM*, IEEE Computer Graphic & Application, pp. 37-45, December 1983.

[6] David F. Rogers and J. Alan Adams, *Mathematical Elements for Computer Graphics*, 2nd Edition, McGRAW-HILL 1990.

[7] F.S. Hill, JR, *Computer graphics*, Macmillan Publishing Co., 1990.

[8] Krzysztof Marciniak, *Geometric Modelling for Numerically Controlled Machining*, Oxford Science Publications, 1991.

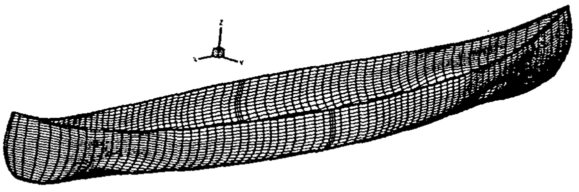
[9] Alan Watt and Mark Watt, *Advanced Animation and Rendering Techniques*, Addison Wesley, 1992.

[10] A. Watt, *3D-Computer graphics*, Addison-Wesley, 1993.

[11] Farin, G., *Curves and Surfaces for Computer Aided Design*, Academic press 1992.

[12] Daniel B. Olfe, *Computer Graphics for Design*, Prentice-Hall, 1995.

[13] Amtec Engineering, Inc., *Tecplot User's Manual*, Version 7, August 1996.



รูปที่ 9 เรือคะนูรูปร่างสมบูรณ์แทนด้วย 4524 พาเนล