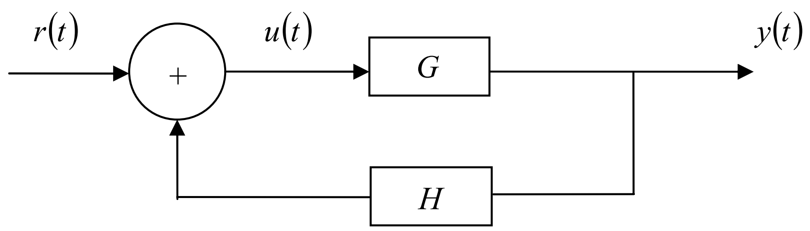
การระบุเอกลักษณ์ของระบบ

1. ความหมายของการระบุเอกลักษณ์ของระบบ

คือ การสร้างแบบจำลองทางคณิตศาสตร์ของระบบ Dynamic โดยใช้ข้อมูลที่ได้จากการทดลอง ซึ่งจะรวมไปถึงการทำการตรวจสอบแบบจำลองทางคณิตศาสตร์ที่เราสร้างขึ้นว่าถูกต้องสมบูรณ์ตามที่เราต้องการหรือไม่ ถ้าเราแบ่งการระบุเอกลักษณ์ตามลักษณะของระบบจะแบ่งได้ 2 แบบคือ การระบุเอกลักษณ์ของระบบเปิด (Open-Loop System Identification) กับ การระบุเอกลักษณ์ของระบบปิด (Closed-Loop System Identification) เนื่องจากระบบปิดเป็นระบบที่มีการป้อนกลับแสดงตามภาพที่ 21 ทำให้เกิดความซับซ้อนมากกว่าระบบเปิดจึงทำให้การระบุเอกลักษณ์ของระบบปิดทำได้ยากกว่าระบบเปิด วิธีการระบุเอกลักษณ์ของระบบปิดแบ่งได้ 3 วิธีดังนี้



ภาพที่ 21 Block Diagram ของระบบปิด (Closed-Loop System)

1. The Direct Approach หมายถึง การใช้วิธีการของ Prediction Error Method (PEM) เข้ามาประยุกต์กับการหาความสัมพันธ์ระหว่างข้อมูลอินพุตกับข้อมูลเอาท์พุตโดยไม่สนใจค่าของข้อมูลอินพุตอ้างอิง แสดงตัวอย่างจากภาพที่ 21 ค่าอินพุตก็วัดจากสัญญาณที่เข้าระบบในที่นี้ก็คือสัญญาณ u และค่าเอาท์พุตก็วัดจากสัญญาณที่ออกจากระบบในที่นี้ก็คือสัญญาณ y เราจะไม่สนใจค่าของสัญญาณอินพุตอ้างอิง r

2. The Indirect Approach หมายถึง การหาความสัมพันธ์ระหว่างข้อมูลอินพุตอ้างอิงกับข้อมูลเอาท์พุต วิธีนี้ก็เหมือนกับว่าเรามองระบบปิดในภาพที่ 21 เป็นระบบรวมของ G และ H โดยมี r เป็นสัญญาณอินพุตและ y เป็นสัญญาณเอาท์พุต

3. The Joint Input-Output Approach หมายถึง การหาความสัมพันธ์ทางคณิตศาสตร์ของระบบปิดในภาพที่ 21 โดยมองสัญญาณ y และ u เป็นค่าของสัญญาณเอาท์พุตของระบบปิด ส่วนสัญญาณอ้างอิง r จะถูกมองเป็นสัญญาณอินพุต

2. <u>วิธีการทางคณิตศาสตร์ที่จะนำมาใช้กับการระบุเอกลักษณ์ของระบบ</u>

ในงานวิทยานิพนธ์นี้เราจะทำการระบุเอกลักษณ์ของระบบมอเตอร์ไฟฟ้ากระแสตรงซึ่งระบบดังกล่าวเป็นระบบเปิด ดังนั้นวิธีการทางคณิตศาสตร์ที่จะกล่าวถึงจะเป็นวิธีที่สามารถใช้กับระบบเปิดได้

ในการระบุเอกลักษณ์ของระบบนั้นเราจะต้องทำการทดลองเก็บข้อมูลจากระบบที่เราต้องการระบุเอกลักษณ์แล้วนำข้อมูลที่ได้มาวิเคราะห์ด้วยวิธีการทางคณิตศาสตร์เพื่อที่จะระบุเอกลักษณ์ ดังนั้นหลังจากที่เราได้ข้อมูลจากการทดลองแล้ว ขั้นตอนต่อไปก็จะเป็นขั้นตอนการใช้วิธีการทางคณิตศาสตร์มาหาความสัมพันธ์ระหว่างข้อมูลที่เป็นอินพุตกับข้อมูลที่เป็นเอาท์พุตซึ่งทำได้หลายวิธีและขั้นตอนนี้จัดว่าเป็นขั้นตอนที่สำคัญที่สุดในเรื่องของการระบุเอกลักษณ์ของระบบ

การหาแบบจำลองทางคณิตศาสตร์ของระบบโดยใช้ข้อมูลที่ได้จากการทดลองจะแบ่งได้เป็น 2 ชนิดหลัก คือ การหาแบบจำลองทางคณิตศาสตร์แบบพารามิเตอร์และการหาแบบจำลองทางคณิตศาสตร์แบบไม่ใช้พารามิเตอร์ ซึ่งการหาแบบจำลองแบบพารามิเตอร์จะเป็นวิธีที่ใช้ในงานวิทยานิพนธ์นี้

การหาแบบจำลองทางคณิตศาสตร์แบบไม่ใช้พารามิเตอร์ (Non-Parametric Model) คือการหาความสัมพันธ์ระหว่างข้อมูลอินพุตและข้อมูลเอาท์พุตของระบบที่ได้จากการทดลองโดยตรง ไม่มีการใช้รูปแบบทางคณิตศาสตร์ใดๆที่จำเพาะเจาะจงหรือรูปแบบที่มีอยู่ก่อนแล้วมาเป็นตัวหาความสัมพันธ์แต่จะมองว่าตัวที่บ่งชี้ความสัมพันธ์นั้นเป็นระบบๆหนึ่งหรือเรียกว่าเป็นการหาความสัมพันธ์โดยไม่สนใจองค์ประกอบย่อยต่างๆภายในระบบ

การหาแบบจำลองทางคณิตศาสตร์แบบพารามิเตอร์ (Parametric Model) คือ การใช้รูปแบบสมการทางคณิตศาสตร์ที่ได้มีอยู่ก่อนแล้วเรียกว่า Model Structure มาเป็นตัวเชื่อมโยงความสัมพันธ์ระหว่างข้อมูลอินพุตและข้อมูลเอาท์พุตของระบบที่ได้จากการทดลอง ในแบบจำลองทางคณิตศาสตร์ดังกล่าวจะมีค่าตัวแปรหรือที่เรียกว่าพารามิเตอร์ประกอบกันอยู่ภายในแบบจำลอง เราจะทำการเลือกแบบจำลองที่คิดว่าเหมาะสมแล้วก็จะทำการหาค่าพารามิเตอร์ภายในแบบจำลองที่ดีที่สุดที่สามารถทำให้การคาดคะเนข้อมูลเอาท์พุตของแบบจำลองนั้นมีค่าใกล้เคียงกับค่าข้อมูลเอาท์พุตที่ได้จากการทดลอง

3. <u>ขั้นตอนการระบุเอกลักษณ์ของระบบ</u>

สำหรับการระบุเอกลักษณ์ของระบบใดๆ ก็ตามสามารถสรุปเป็นขั้นตอนการทำงานโดยรวมได้ดังนี้

1. ออกแบบการทดลอง
2. ทำการทดลองเก็บข้อมูล
3. ตรวจสอบและปรับปรุงข้อมูลดิบ
4. เลือกวิธีการระบุเอกลักษณ์
5. ทำการหาค่าพารามิเตอร์ที่ดีที่สุดของ Model Structure และสร้าง Model Structure โดยใช้ ค่าพารามิเตอร์ที่ดีที่สุด
6. ทำการทดสอบคุณสมบัติของ Model Structure ที่ได้ ถ้าผลการทดสอบคุณสมบัติของ Model Structure ดีเพียงพอต่อความต้องการแล้วก็เป็นอันว่าเสร็จสิ้นขั้นตอนการระบุเอกลักษณ์ของระบบ แต่ถ้ายังไม่ดีพียงพอก็ให้กลับไปทำตั้งแต่ขั้นตอนที่ 4 อีก หรือในบางครั้งก็อาจจะต้องกลับไปทำตั้งแต่ขั้นตอนที่ 1

3.1 ออกแบบการทดลอง

เนื่องจากการระบุเอกลักษณ์ของระบบเป็นการหาแบบจำลองทางคณิตศาสตร์โดยใช้ข้อมูลที่ได้จากการทดลอง การทดลองที่ถูกออกแบบมาไม่เหมาะสมกับระบบก็อาจเป็นสาเหตุที่ทำให้เราได้ข้อมูลที่จะนำไปใช้หาแบบจำลองทางคณิตศาสตร์ไม่ดีเท่าที่ควร ส่งผลให้ผลของการระบุเอกลักษณ์ของระบบก็ไม่ดีเท่าที่ควรไปด้วย อีกทั้งการทดลองที่เหมาะสมก็อาจสามารถลดขั้นตอนการปรับปรุงข้อมูล (ขั้นตอนที่ 3) ได้อีกด้วย ดังนั้นในขั้นตอนการออกแบบการทดลองนี้จึงมีจุดประสงค์คือสร้างวิธีการทดลองที่ถูกต้องและเหมาะสมเพื่อนำไปใช้ในทดลองเก็บค่าข้อมูลดิบ

คำว่า “การทดลองที่เหมาะสม” หมายถึงการทดลองที่สามารถให้ข้อมูลได้เหมาะสมและมีจำนวนเพียงพอต่อความต้องการใช้ในขั้นตอนการระบุเอกลักษณ์ รวมถึงข้อมูลดังกล่าวจะต้องมีแนวโน้มเป็นไปตามที่เราคาดคะเน ดังนั้นในการออกแบบการทดลองจึงมีสิ่งที่ต้องคำนึงถึงด้วยกันหลายอย่างดังนี้

3.1.1 สัญญาณอินพุตที่จะใช้ใส่เข้าไปกระตุ้นระบบ

เราสามารถสร้างการทดลองได้เหมาะสมเมื่อเราใช้อินพุตกระตุ้นระบบได้อย่างถูกวิธีคือ อินพุตต้องประกอบไปด้วยความถี่ทุกๆความถี่ที่ต้องการทดสอบระบบ (Lennart Ljung, 1999: 414)

เรามีวิธีบอกคุณภาพของสัญญาณได้หลายวิธี ตัวอย่างเช่น เนื่องจากเราต้องการให้ Covariance Matrix มีค่าน้อย เราจึงต้องใช้สัญญาณอินพุตให้มีกำลังมากที่สุดเท่าที่จะเป็นไปได้คือใช้อินพุตที่มีค่า Amplitude มากๆ (Lennart Ljung, 1999: 415) และยังมีคุณสมบัติเพิ่มเติมที่จะใช้บอกคุณภาพของสัญญาณอินพุตอีกคือค่าเฉลี่ยของสัญญาณอินพุตตลอดช่วงควรมีค่าเท่ากับศูนย์ ซึ่งคุณภาพของสัญญาณทั้ง 2 อย่างที่กล่าวมาจะใช้สามารถแสดงได้จาก The Crest Factor C_r ซึ่งจะบอกเป็นอัตราส่วนระหว่างค่าที่มากที่สุดของสัญญาณอินพุตกับค่าเฉลี่ยกำลังสองของสัญญาณอินพุตแสดงได้ตามสมการดังนี้

$$C_r^2 = \frac{\max_t u^2(t)}{\lim_{N\to\infty} \frac{1}{N} \sum_{t=1}^{N} u^2(t)} \tag{70}$$

สัญญาณอินพุตที่ดีควรจะมีค่า C_r น้อยๆ ในทางทฤษฎีแล้วค่าน้อยที่สุดที่เป็นไปได้คือ $C_r = 1$ ซึ่งจะเกิดขึ้นได้เมื่อถ้าค่าเฉลี่ยของสัญญาณอินพุตตลอดช่วงควรมีค่าเท่ากับศูนย์และจะได้ว่าค่าเฉลี่ยกำลังสองของสัญญาณอินพุตมีค่าเท่ากับ $u^2(t)$ แสดงตามสมการ

$$\lim_{n\to\infty} \frac{1}{N} \sum_{t=1}^{N} u^2(t) = u^2(t) \tag{71}$$

$$C_r^2 = \frac{u^2(t)}{u^2(t)} = 1 \tag{72}$$

สัญญาณอินพุตที่จะทำให้ $C_r = 1$ จะเป็นสัญญาณประเภทสัญญาณสมมาตรสองระดับ (Binary Symmetric Signals) จะมีค่า $u(t) = \pm u_{\max}$ ฉะนั้นสัญญาณอินพุตประเภทสัญญาณสมมาตรสองระดับจึงมีข้อได้เปรียบมากกว่าสัญญาณอินพุตชนิดอื่นในเรื่องของ The Crest Factor C_r แต่ข้อสำคัญที่ไม่อาจลืมได้คือต้องตระหนักไว้เสมอว่าสัญญาณประเภทนี้ไม่สามารถกระตุ้นให้ระบบไม่

เชิงเส้นให้แสดงคุณสมบัติความไม่เป็นเชิงเส้นออกมาได้ (Lennart Ljung, 1999: 416) ดังนั้นระบบไม่เป็นเชิงเส้นจึงไม่สามารถตรวจสอบได้โดยการใช้สัญญาณประเภทนี้

ยังคงมีสัญญาณที่นำมาใช้เป็นอินพุตอีกหลายชนิดที่มักจะนำมาใช้กระตุ้นระบบตัวอย่างอีกสัญญาณหนึ่งก็คือ Pseudo-Random Binary Signal (PRBS) สัญญาณชนิดนี้มีลักษณะคล้ายกันกับ Random Binary Signal แต่จริงๆแล้วมีความต่างกันคือสัญญาณ PRBS จะสร้างขึ้นจากสูตรสำเร็จ โดยในสูตรสำเร็จนั้นจะต้องใช้ค่าเริ่มต้นที่ได้จากการสุ่ม ถ้าเราใช้ค่าเริ่มต้นที่เหมือนกันในสูตรสำเร็จเดียวกันเราจะได้สัญญาณ PRBS ที่เหมือนกันทุกครั้ง ในการสร้างสัญญาณ PRBS จากสูตรสำเร็จจะสร้างแบบวนซ้ำต่อเนื่องกันไปเรื่อยๆ โดยใช้ค่าเริ่มต้นเดียวกันทำให้สัญญาณ PRBS มีลักษณะเป็นคาบ คุณสมบัติที่สำคัญอีกอย่างหนึ่งก็คือสัญญาณ PRBS นี้เราสามารถสร้างให้มีลักษณะคล้าย White Noise นั่นคือในตัวสัญญาณเราสามารถสร้างให้ประกอบไปด้วยเกือบทุกความถี่ที่เราต้องการซึ่งทำได้โดยใช้หลักการว่าในช่วงคาบเวลาที่ยาวที่สุดของสัญญาณ PRBS (Maximum Length PRBS) จะต้องมีการเปลี่ยนแปลงในรูปแบบที่ไม่ซ้ำกัน สูตรสำเร็จที่นำมาใช้สร้างสัญญาณ PRBS $u(t)$ มีรูปแบบเหมือนสมการผลต่างดังนี้

$$\begin{aligned} u(t) &= rem\,(A(q)\,u(t), 2) \\ &= rem\,(a_1 u(t-1) + \ldots + a_n u(t-n), 2) \end{aligned} \tag{73}$$

โดยที่ $a_1, a_2, a_3, \ldots, a_n \in \{0, 1\}$ และ rem มาจาก Remainder แปลว่า เศษเหลือ ฉะนั้น $rem\,(x, 2)$ หมายถึงเศษของการหาร x ด้วย 2 ซึ่งจริงๆแล้วก็เหมือนกันกับ x modulo 2 นั่นแหละ เศษเหลือตัวนี้จะมีค่าที่เป็นไปได้สองค่าเท่านั้นคือ 0 และ 1 ในการนำไปใช้จริง $u(t)$ จะเป็นสัญญาณสองระดับซึ่งไม่จำเป็นต้องเป็นค่า 0 กับ 1 เท่านั้นแต่จะเป็นสองระดับค่าอื่นๆก็ได้ จากสมการ (73) จะเห็นว่า $u(t)$ สร้างขึ้นมาจากค่าในอดีตของตัวมันเองซึ่งมีเวกเตอร์ค่าในอดีตคือ

$$[u(t-1) \quad u(t-2) \quad u(t-3) \quad \cdots \quad u(t-n)]$$

ในตอนเริ่มต้นเวกเตอร์ค่าในอดีตนี้เราต้องสมมติค่าเริ่มต้นขึ้นเองซึ่งได้จากการสุ่มโดยจะไม่ใช้ค่าเริ่มต้นที่เป็น 0 เพราะจะทำให้สัญญาณ PRBS ที่สร้างขึ้นมีค่าเป็น 0 ตลอดกาล จากเวกเตอร์ดังกล่าวจะเห็นว่าในเวกเตอร์มีสมาชิกทั้งหมด n ตัว ดังนั้นเวกเตอร์นี้จะมีได้ทั้งหมด 2^n รูปแบบ กรณีที่

$n = 0$ จะทำให้เวกเตอร์นี้เป็นเวกเตอร์ศูนย์ซึ่งจะไม่นับรูปแบบนี้ ดังนั้นเวกเตอร์ดังกล่าวจึงมีรูปแบบที่เราสามารถสร้างได้จริง $M = 2^n - 1$ รูปแบบ

จากการสร้างสัญญาณ PRBS $u(t)$ ซึ่งสร้างได้จากสมการ (73) โดยใช้ $n = 7$ เราจะได้ $u(t)$ จะมีค่าเรียงกันต่อกันไปดังนี้

$$u(1), u(2), u(3) \ldots, u(128), u(129), u(130), \ldots, u(255), u(256), u(257), \ldots$$

โดยเราสามารถเลือก $A(q)$ เพื่อให้ผลเป็นดังนี้ได้

$$\begin{array}{llllll}
u(1) & = u(128) & = u(255) & = u(1+k) & : & k = 0, M, 2M, \ldots \\
u(2) & = u(129) & = u(256) & = u(2+k) & : & k = 0, M, 2M, \ldots \\
u(3) & = u(130) & = u(257) & = u(3+k) & : & k = 0, M, 2M, \ldots \\
\vdots & \vdots & \vdots & \vdots & & \vdots \\
u(M) & = u(2M) & = u(3M) & = u(M+k) & : & k = 0, M, 2M, \ldots
\end{array}$$

จากผลดังกล่าวหมายความว่าสัญญาณ PRBS มีลักษณะเป็นคาบที่มีความยาวคาบสูงสุดเท่ากับ M เรียกสัญญาณ PRBS แบบนี้ว่า Maximum Length PRBS

ดังนั้นสำหรับค่า n ใดๆที่เราเลือกมาใช้งานก็จะต้องเลือก $A(q)$ ให้เหมาะสมเพื่อให้เกิดความยาวคาบสูงสุด ซึ่งสรุปได้ตามตารางที่ 1 (Lennart Ljung, 1999: 420) ยังคงมี $A(q)$ ค่าอื่นที่สามารถทำให้สัญญาณ PRBS เป็น Maximum Length PRBS ได้ (สามารถหาข้อมูลเพิ่มเติมได้ที่ http://www.eecircle.com/applets/009/LFSR.html) และสำหรับการเลือกใช้ $A(q)$ ค่าที่ไม่ทำให้เกิด Maximum Length PRBS สัญญาณ PRBS ก็ยังคงมีลักษณะเป็นคาบอยู่ แต่ความยาวคาบก็อาจจะไม่ได้เป็นความยาวสูงสุด

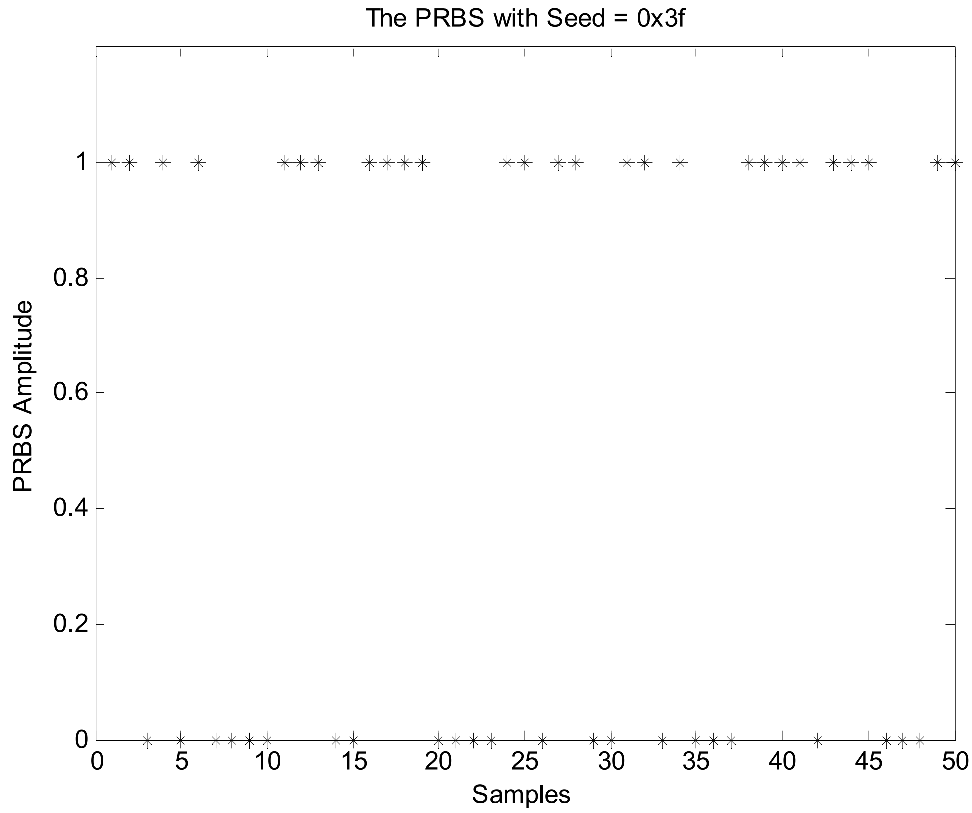
<u>ตารางที่ 1</u> แสดงค่า $A(q)$ ที่ไม่เป็น 0 ที่จะทำให้เกิด Maximum length PRBS ที่ค่า n ต่างๆกัน

Order n	$M = 2^n - 1$	a_k **non-zero for** k
2	3	1, 2
3	7	2, 3
4	15	1, 4
5	31	2, 5
6	63	1, 6
7	127	3, 7
8	255	1, 2, 7, 8
9	511	4, 9
10	1023	7, 10
11	2047	9, 11

ในการคำนวณหาค่าของสัญญาณ PRBS โดยใช้ Web Site ดังกล่าวให้รู้ไว้ด้วยว่าในช่อง Register Seed Data Vector จะให้เราเติมค่า Initial โดยเรียงจาก LSB ไป MSB ตัวอย่างเช่น ถ้าเราจะใช้ค่า Initial เท่ากับ 0x3f เราจะต้องเติมค่าลงไปในช่องดังกล่าวคือ 1 1 1 1 1 1 0 0 และในช่อง Initial Data ก็จะแสดงค่าเป็น 0x3f คือ 0 0 1 1 1 1 1 1 ถ้ากำหนดให้ $n = 8$ และใช้ค่า Initial เท่ากับ 0x3f จาก Web Site ดังกล่าวเราจะได้เวกเตอร์ของ PRBS ดังนี้

PRBS = {1 1 0 1 0 1 0 0 0 0 1 1 1 0 0 1 1 1 1 0 0 0 0 1 1 0 1 1 0 0 1 1 0 1 0 0 0 1 1 1 1 0 1 1 1 0 0
0 1 1 1 0 1 0 0 0 0 0 0 0 1 1 1 1 1 0 0 0 1 0 0 1 0 1 0 0 0 1 0 0 0 1 0 1 1 0 1 0 1 0 1 1 1 0 0 1 0 0 0
1 1 0 0 1 0 0 1 0 0 1 1 1 0 0 0 0 0 0 1 0 0 0 0 1 0 0 1 1 0 1 1 1 1 0 0 1 1 0 0 1 1 1 0 1 1 1 1 1 1 0 0
1 0 1 1 0 0 1 0 1 0 1 1 0 1 1 0 1 0 0 1 0 0 0 0 0 1 1 0 0 0 1 1 0 1 0 1 1 0 0 0 1 0 1 0 1 0 1 0 0 1 1 0
0 0 0 0 1 0 1 1 1 0 1 0 1 1 1 1 1 0 1 1 0 1 1 1 0 1 1 0 0 0 0 1 0 1 0 0 1 0 1 1 1 1 0 1 0 0 1 1 1 1 1 1}

สามารถแสดงค่าของ PRBS เป็นกราฟได้ตามภาพที่ 22



ภาพที่ 22 แสดงค่าของ PRBS

การสร้างสัญญาณ PRBS ให้มีลักษณะเป็น Maximum Length PRBS โดยมี Amplitude เป็น $\pm u_{\max}$ จะทำให้สัญญาณ PRBS ดังกล่าวมีคุณสมบัติกำลังหนึ่งและกำลังสองแสดงได้ดังนี้

คุณสมบัติกำลังหนึ่ง (First Order Property) คือ

$$\left| \frac{1}{M} \sum_{t=1}^{M} u(t) \right| = \frac{u_{\max}}{M} \tag{74}$$

และคุณสมบัติกำลังสอง (Second Order Property) คือ

$$\begin{aligned} R_u(k) &= \frac{1}{M} \sum_{t=1}^{M} u(t) u(t+k) \\ &= \begin{cases} u_{\max}^2 & k = 0, \pm M, \pm 2M \ldots \\ -\dfrac{u_{\max}^2}{M} & \text{else} \end{cases} \end{aligned} \tag{75}$$

จากคุณสมบัติกำลังหนึ่งจะเห็นได้ว่าสัญญาณ PRBS มีค่าเฉลี่ยตลอดช่วงความยาวคาบไม่จำเป็นต้องเท่ากับ 0 ทั้งนี้เนื่องจากสัญญาณ PRBS เป็นสัญญาณแบบสุ่ม (Random) ซึ่งต่างจากสัญญาณสมมาตรสองระดับเป็นสัญญาณที่มีลักษณะสมมาตรจึงทำให้ค่าค่าเฉลี่ยตลอดช่วงความยาวคาบมีค่าเท่ากับ 0 จากคุณสมบัติกำลังสองจะเห็นได้ว่าสัญญาณ PRBS มีลักษณะเป็นคาบและจะมีค่ามากที่สุดเท่ากับ $u_{\max}^2$ ก็ต่อเมื่อสัญญาณ $u(t+k)$ ถูกเลื่อนมาจนเกิดการซ้อนทับกันกับ $u(t)$ พอดีนั่นก็คือที่ $k = M$ เพื่อความง่ายที่จะเข้าใจเราสามารถแสดงให้เห็นได้โดยพิจารณาจาก $u(t)$ เป็นสัญญาณที่มีลักษณะเป็นคาบดังนั้นเราจะได้

$$\begin{aligned} u(t) &= u(t+M) = u(t+2M) = \cdots \\ &= u(t+k) \quad : \quad k = 0, \pm M, \pm 2M, \ldots \end{aligned} \tag{76}$$

จากสมการ (76) แสดงว่าที่ $k = 0, \pm M, \pm 2M, \ldots$ จะได้ $u(t) = u(t+k)$ นั่นก็คือ

$$u(t)u(t+k) = u_{\max}^2 \quad : \quad k = 0, \pm M, \pm 2M, \ldots \tag{77}$$

เพื่อจะดูว่าสัญญาณ PRBS ดังกล่าวมีองค์ประกอบของความถี่ใดบ้างเราจึงทำการเปลี่ยนสัญญาณ PRBS $u(t)$ ในไปอยู่ในรูปของโดเมนความถี่ได้

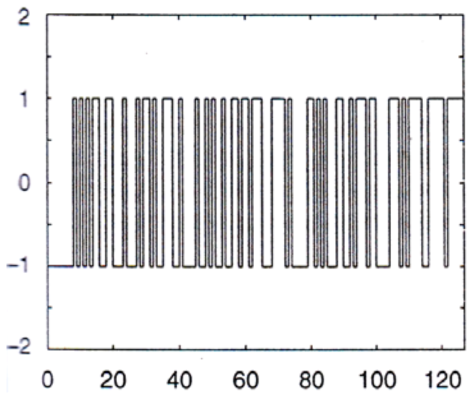
$$\begin{aligned} \Phi_u^p(\omega) &= \sum_{k=0}^{M-1} R_u(k) e^{-ik\omega} \\ &= u_{\max}^2 \left[1 - \frac{1}{M} \sum_{k=1}^{M-1} e^{-ik\omega} \right] \\ &= u_{\max}^2 \left[1 - \frac{1}{M} \frac{e^{-i\omega} - e^{-iM\omega}}{1 - e^{-i\omega}} \right] \\ &= \begin{cases} u_{\max}^2 \dfrac{1}{M} & \text{for} \quad \omega = 0 \\ u_{\max}^2 \left(1 + \dfrac{1}{M} \right) & \text{for} \quad \omega = \dfrac{2\pi k}{M}, \quad k = 1, \ldots, M-1 \end{cases} \end{aligned} \tag{78}$$

จาก (Lennart Ljung, 1999: 421) เราสามารถเขียนสัญญาณ PRBS $u(t)$ ในโดเมนความถี่ได้ดังนี้

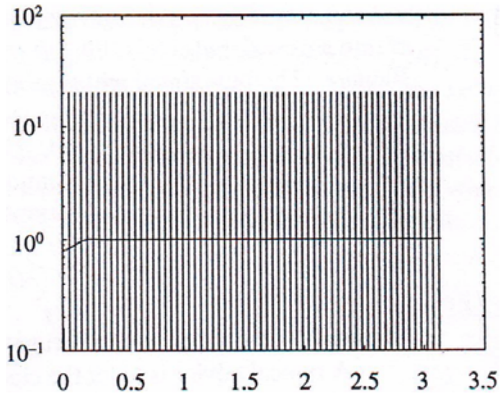
$$\Phi_u(\omega) = \frac{2\pi u_{\max}^2}{M} \sum_{k=1}^{M-1} \delta\left(\omega - \frac{2\pi k}{M} \right), \quad 0 \le \omega \le 2\pi \tag{79}$$

จากสมการ (79) ถ้าเราไม่สนใจเทอมของ $1/M^2$ จะเห็นได้ว่าถ้าเราพิจารณาในช่วง $0 \le \omega \le 2\pi$ จะประกอบไปด้วยความถี่ทั้งหมด $M-1$ ความถี่ (รวมจุดที่ $\omega = 0$ ด้วย) โดยทุกความถี่จะมีค่าสูงสุดเท่ากัน ดังนั้นถ้าเราทำการเพิ่มค่า M จะเห็นได้ว่าจำนวนองค์ประกอบของความถี่ในช่วง $0 \le \omega \le 2\pi$ ก็จะมีค่าเพิ่มขึ้นตามไปด้วย

ภาพที่ 23 แสดงสัญญาณ PRBS ที่มี $M = 127$ จำนวน 1 คาบ ภาพที่ 24 แสดงโดเมนความถี่ของภาพที่ 23 ในการทดลองเราจะใส่สัญญาณ PRBS ตามภาพที่ 23 ต่อเนื่องกันเป็นคาบไปหลายๆรอบดังนั้นผลที่ได้ในโดเมนความถี่ในภาพที่ 24 ก็จะมีต่อเนื่องกันเป็นคาบต่อไปอีก ด้วยคุณสมบัติของ PRBS เราจึงสามารถใช้สัญญาณนี้ไปกระตุ้นระบบเพื่อดูผลตอบสนองทางด้านความถี่ได้ มีข้อสังเกตว่าถ้าสัญญาณ PRBS $u(t)$ ที่นำมาใช้ประกอบไปด้วยข้อมูลที่ไม่ครบทั้งคาบจะทำให้คุณสมบัติกำลังหนึ่งและกำลังสองอาจไม่เป็นจริง



<u>ภาพที่ 23</u> สัญญาณ PRBS $M = 127$

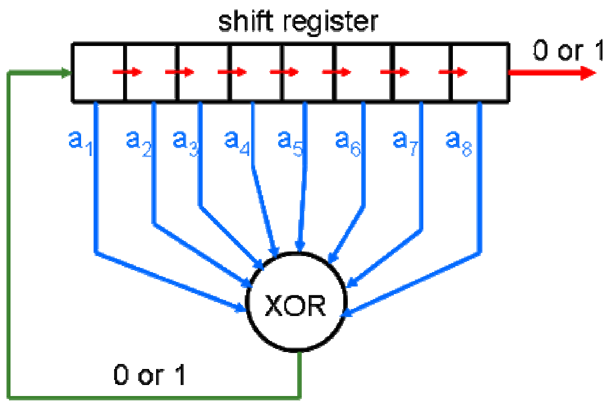


<u>ภาพที่ 24</u> โดเมนความถี่ของภาพที่ 23 หาจากการใช้ฟังค์ชั่น FFT

การสร้างสัญญาณ PRBS เราสามารถสร้างได้โดยใช้โปรแกรมภาษา C หรือภาษาอื่นๆ โดยเราจะมองสมการ (73) ให้อยู่ในรูปที่เข้าใจได้ง่ายขึ้นคือ เมื่อพิจารณาสมการ (73) แล้วจะพบว่าสัญญาณ PRBS $u(t)$ จะมีค่าได้เพียง 2 ค่าคือ 0 หรือ 1 ทำให้เราสามารถมองสมการดังกล่าวได้ใหม่คือ การที่นำค่าของ $a_1u(t-1)+\ldots+a_nu(t-n)$ มา modulo 2 นั้นผลลัพธ์ที่ได้จะมีค่าเทียบเท่ากับการทำ Exclusive OR แสดงความสัมพันธ์ตามสมการดังนี้

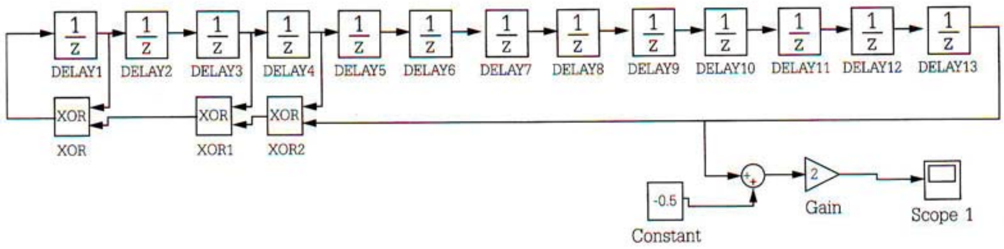
$$u(t) = rem\left(a_1u(t-1)+\ldots+a_nu(t-n), 2\right) = a_1u(t-1) \wedge a_2u(t-2) \wedge \ldots \wedge a_nu(t-n) \quad (80)$$

ในกรณีที่ $n=8$ เราสามารถเขียนสมการ (80) เป็นแผนภาพได้ภาพที่ 25 (Tesheng Hsiao, n.d.) และเราจะใช้แผนภาพตามภาพที่ 25 ไปเป็นต้นแบบในการเขียนโปรแกรมสำหรับ n ค่าอื่นๆได้อีก

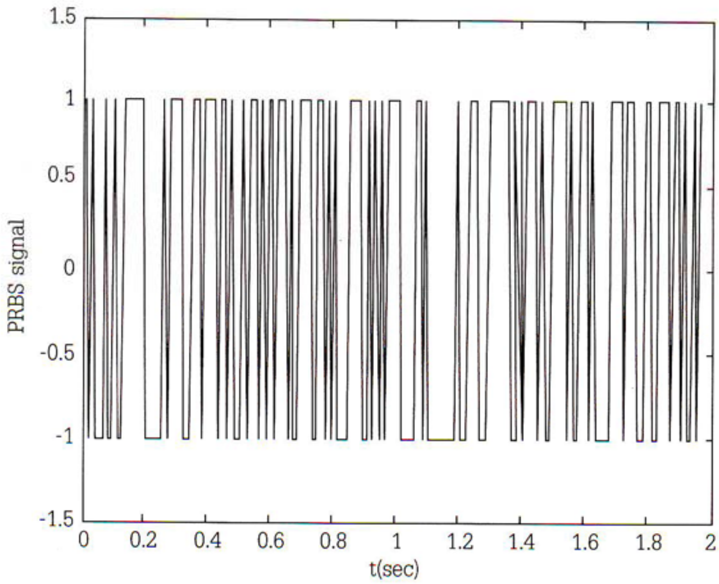


ภาพที่ 25 แสดงแผนผังที่สมมูลกับสมการ (80) โดยที่ $n=8$

นอกจากเราจะสร้างสัญญาณ PRBS ได้โดยการเขียนโปรแกรมแล้ว เรายังสามารถสร้างได้โดยใช้การเชื่อมต่อ Block ในโปรแกรม MATLAB Simulink (ดร.วโรดม ตู้จินดา, 2547) แสดงตัวอย่างได้ตามภาพที่ 26 แผนภาพดังกล่าวจะทำให้เกิดสัญญาณ PRBS แสดงตามภาพที่ 27



ภาพที่ 26 แสดงการต่อเชื่อมสัญญาณต่างในโปรแกรม MATLAB Simulink



ภาพที่ 27 สัญญาณ PRBS ที่สร้างโดยใช้โปรแกรม MATLAB Simulink ในภาพที่ 26

3.1.2 สัญญาณเอาท์พุตที่ต้องการวัด

หลังจากที่เราใส่อินพุตเข้าไปกระตุ้นระบบแล้วส่งผลให้ระบบสร้างผลตอบสนองออกมา ผลตอบสนองดังกล่าวจะออกมาในรูปของการเปลี่ยนแปลงค่าตัวแปรต่างๆของระบบ จุดประสงค์ของเราก็เพื่อที่จะเก็บค่าต่างๆของตัวแปรเหล่านั้นเพื่อนำมาเป็นข้อมูล ตัวแปรที่เราจะเก็บค่ามาก็คือตัวแปรที่เราสนใจ ตัวแปรเหล่านั้นต้องสามารถเปลี่ยนแปลงได้เนื่องจากการเปลี่ยนแปลงค่าอินพุตและต้องมีคุณสมบัติคือไม่ว่าจะทำการทดลองเป็นจำนวนกี่ครั้งก็ตามถ้าอินพุตของระบบเป็นชุดเดิมค่าของตัวแปรเหล่านั้นก็ต้องให้ค่าเหมือนเดิมเช่นกันหรือเรียกว่าผลการทดลองจะเปลี่ยนแปลงได้ก็ต่อเมื่อมีการเปลี่ยนวิธีการทดลองเท่านั้น นั่นคือระบบเราต้องไม่ขึ้นกับ

เวลา (Time-Invariant) ดังนั้นจะเห็นได้ว่าการระบุเอกลักษณ์ของระบบในที่นี้จะกล่าวถึงกับระบบที่ไม่ขึ้นกับเวลา (Time-Invariant System) เท่านั้น

3.1.3 ปัจจัยอื่นๆ ในการทดลอง

หมายถึงสิ่งอื่นๆที่จะทำให้เราไม่สามารถทำการทดลองตามที่เราได้ออกแบบไว้ได้ เช่น การทดลองที่ต้องใช้กระแสไฟฟ้าสูงก็มีความจำเป็นต้องเลือกแหล่งจ่ายไฟให้พอเพียงกับการทดลอง ความละเอียดของอุปกรณ์เครื่องมือวัดว่ามีค่ามากพอเพียงกับความต้องการหรือไม่ รวมถึงสัญญาณรบกวนที่จะมีแฝงอยู่ในทุกที่ทุกอุปกรณ์โดยเฉพาะอุปกรณ์เครื่องมือวัดเป็นแบบชนิดที่มีย่านการทำงานที่สัญญาณต่ำมากๆก็จะเกิดสัญญาณรบกวนขึ้นได้ง่าย อีกทั้งอุปกรณ์ดังกล่าวมักจะต้องมีการขยายสัญญาณเพื่อนำไปใช้ในการประมวลผลทำให้เกิดการขยายสัญญาณรบกวนเพิ่มขึ้นไปอีกทำให้อาจเกิดความผิดพลาดในการหาแบบจำลองหรืออาจใช้เวลาประมวลผลนานขึ้นก็เป็นได้ ฯลฯ จะเห็นว่าเราไม่สามารถหาวิธีการที่จะควบคุมปัจจัยอื่นๆที่แน่นอนได้ทั้งหมด ดังนั้นในขั้นตอนการออกแบบจึงต้องคาดการณ์ถึงเหตุการณ์ที่คาดว่าจะเกิดขึ้นและมีผลต่อระบบอยู่เสมอเพื่อที่จะสามารถแก้ปัญหาได้ทันท่วงที

3.2 ทำการทดลอง

ทำการทดลองเก็บค่าข้อมูลอินพุตและเอาท์พุตตามที่ได้ออกแบบไว้ ในขั้นตอนนี้ข้อมูลที่ได้ยังคงเป็นข้อมูลดิบอยู่ ในขั้นตอนนี้ให้คำนึงถึงสภาพแวดล้อมที่จะมีผลต่อการทำงานของระบบหรือของอุปกรณ์เครื่องมือวัด ซึ่งในช่วงการเก็บข้อมูลการทดลองจะต้องควบคุมสภาพแวดล้อมให้คงที่ให้มากที่สุดเท่าที่จะสามารถทำได้ เช่น ความหนืดหรือความหนาแน่นของเหลวในการทดลองเรื่องการหาการระบุเอกลักษณ์ของปั๊มน้ำ ความหนาแน่นและความเร็วของอากาศในการทดลองเรื่องการระบุเอกลักษณ์ของเครื่องบิน ช่วงสภาพแวดล้อมที่อุปกรณ์เครื่องมือวัดมีการทำงานไม่ผิดพลาด ฯลฯ รวมถึงระยะเวลาในการเสื่อมสภาพของระบบที่เราต้องการระบุเอกลักษณ์ เพราะถ้าระบบเป็นระบบที่มีการเสื่อมสภาพเร็วเกินไปก็เหมือนกับเรากำลังทำการระบุเอกลักษณ์ของระบบระบบที่ขึ้นกับเวลา (Time-Varying System) ซึ่งจะต้องใช้การวิเคราะห์ที่ต่างกันออกไป

ในขั้นตอนนี้ยังต้องคำนึงถึงวิธีการเก็บค่าข้อมูลที่ได้จากการทดลองด้วย กล่าวคือ เนื่องจากการคำนวณหาแบบจำลองทางคณิตศาสตร์เราจะทำในระบบคอมพิวเตอร์ซึ่งเป็นระบบ Digital ฉะนั้นสัญญาณที่เรานำไปใช้ประมวลผลก็จะเป็นลักษณะไม่ต่อเนื่อง การเก็บข้อมูลจึงเป็นแบบการสุ่มเก็บ การสุ่มเก็บข้อมูลเช่นนี้ทำให้เกิดการสูญเสียข้อมูล เราจะต้องทำการเก็บข้อมูลแบบที่ข้อมูลที่เราสูญเสียไปนั้นเป็นข้อมูลที่ไม่เกิดประโยชน์ต่อการประมวลผลจึงทำให้ต้องคำนึงถึงคาบเวลาของการสุ่มเก็บสัญญาณที่เหมาะสมคือไม่มากหรือน้อยเกินไปเพราะการใช้คาบเวลาที่มากเกินความเหมาะสมจะทำให้สูญเสียข้อมูลที่สำคัญและการที่จะใช้คาบเวลาที่น้อยเพื่อให้ได้ข้อมูลที่ละเอียดก็อาจจะส่งผลเสียคือมีข้อมูลที่ไม่เกี่ยวข้องเข้ามาอยู่ในการจัดเก็บอีกทั้งยังส่งผลให้ต้องสูญเสียค่าใช้จ่ายเพิ่มเติมกับอุปกรณ์เปลี่ยนสัญญาณ Analog มาเป็น Digital และก็อาจเป็นเหตุให้ใช้เวลาประมวลผลมากกว่าที่ควรจะเป็นเพราะจำนวนข้อมูลมีมากเกินไปก็เป็นได้ การเลือกคาบเวลาสุ่มเก็บสัญญาณที่ดีที่สุดจึงหมายถึงคาบเวลาที่น้อยที่สุดที่ยังสามารถเก็บข้อมูลที่เป็นประโยชน์ต่อการประมวลผลได้มากที่สุด จาก (Alan et al., 1999) สรุปได้ว่าอัตราการสุ่มเก็บสัญญาณที่เหมาะสมคือค่าของอัตราการสุ่มเก็บสัญญาณต้องไม่น้อยกว่า 2 เท่าของช่วงความถี่ที่เราสนใจของระบบ ความถี่นี้เราอาจเรียกอีกชื่อว่าความถี่ไนควิส

3.3 ตรวจสอบและปรับปรุงข้อมูลดิบ

ในขั้นตอนการเก็บข้อมูลอาจจะมีปัจจัยภายนอกอื่นๆที่ไม่สามารถควบคุมได้เป็นผลให้ระหว่างการเก็บข้อมูลอาจทำให้มีข้อมูลอื่นๆเข้ามาปะปนเข้ากับข้อมูลที่เราต้องการเก็บ ข้อมูลที่ได้จากการทดลองจึงอาจประกอบไปด้วยข้อมูลทั้งส่วนที่จำเป็นและไม่จำเป็นรวมอยู่ด้วยกัน ดังนั้นเราจึงต้องนำข้อมูลดิบที่ได้มาตรวจสอบและอาจต้องทำการปรับปรุงข้อมูล เช่น การนำข้อมูลส่วนที่ไม่สำคัญออก การใส่ Filter กรองความถี่ของข้อมูลช่วงที่ไม่ต้องการออก การกำจัดค่าเฉลี่ย (Remove Mean) ของชุดข้อมูล ฯลฯ เพื่อให้ข้อมูลประกอบไปด้วยส่วนประกอบที่สำคัญเท่านั้น และเนื่องจากการกระทำในขั้นตอนนี้เป็นผลทำให้จำนวนข้อมูลลดลงไปจากเดิมจึงทำให้เป็นผลดีกับการคำนวณหาแบบจำลองทางคณิตศาสตร์เพราะจะมีความซับซ้อนน้อยลง

ไม่ว่าเราจะออกแบบการทดลองและทำการทดลองได้ดีเพียงไรก็ตาม การเก็บข้อมูลจากการทดลองก็มักจะมีสัญญาณรบกวนปะปนเข้ามาด้วยเสมอ สัญญาณรบกวนที่เกิดขึ้นมีด้วยกันหลายชนิด บางชนิดก็มีรูปแบบทางคณิตศาสตร์ที่แน่นอนซึ่งสามารถอธิบายได้โดยใช้สมการคณิตศาสตร์ บางชนิดก็มีลักษณะเป็นสัญญาณแบบสุ่มที่ไม่สามารถคาดคะเนค่าของสัญญาณที่จะเกิดขึ้นต่อไป

ได้แม้ว่าจะรู้ค่าที่เกิดขึ้นก่อนกี่ค่าก็ตามทำให้ไม่สามารถใช้สมการคณิตศาสตร์ทั่วๆไปมาอธิบายได้ สัญญาณรบกวนที่มีรูปแบบทางคณิตศาสตร์ในบางครั้งเราไม่สามารถคาดคะเนค่าที่เกิดขึ้นต่อไปได้โดยตรงแต่อย่างน้อยก็จะสามารถคาดคะเนคุณสมบัติของสัญญาณที่จะเกิดขึ้นต่อไปได้ทำให้การตัดสัญญาณรบกวนออกจากข้อมูลดิบทำได้ไม่ยากจนเกินไป สำหรับสัญญาณรบกวนแบบสุ่มที่ปะปนอยู่ในข้อมูลดิบ การตัดสัญญาณรบกวนออกจากข้อมูลดิบก็จะมีด้วยกันหลายวิธีซึ่งวิธีใดจะเหมาะสมก็ต้องแล้วแต่ชนิดของสัญญาณรบกวน การตัดสัญญาณรบกวนออกจากข้อมูลดิบเพื่อที่จะทำให้ข้อมูลมีคุณภาพมากขึ้นก็คือส่วนหนึ่งของการปรับปรุงคุณภาพของข้อมูล

โดยทั่วไปแล้วสัญญาณรบกวนที่ปะปนเข้ามากับข้อมูลดิบจะเป็นสัญญาณรบกวนที่มีองค์ประกอบของความถี่ที่เป็นความถี่สูง การปรับปรุงคุณภาพของข้อมูลดิบชนิดนี้เราจะใช้ Filter ที่จะกรองความถี่สูงออกไปซึ่งก็มีด้วยกันหลายชนิด ในงานวิจัยนี้จะแนะนำ Filter ที่มีความซับซ้อนไม่มากเพื่อการประมวลผลที่รวดเร็วและสามารถนำไปใช้ในการประมวลผลแบบเวลาจริง (Real Time) ได้ด้วย

3.3.1 Moving Average Filtering or Moving Average Smoothing

สัญญาณรบกวนที่เป็นความถี่สูงที่มีการกระจายอยู่รอบๆข้อมูลดิบที่มีการกระจายที่ประมาณด้วยสายตาแล้วพบว่าค่อนข้างจะคงที่เราสามารถลดการรบกวนของสัญญาณดังกล่าวได้โดยใช้สมการคณิตศาสตร์ดังนี้

$$
\begin{aligned}
y(t) &= \frac{1}{2n+1}\left[x(t-n)+\ldots+x(t-1)+x(t)+x(t+1)+\ldots+x(t+n)\right] \\
&= \frac{1}{2n+1}\sum_{m=-n}^{n} x(t+m)
\end{aligned}
\tag{81}
$$

เราเรียก Filter ตามสมการ (81) ชนิดนี้ว่า Moving Average Filtering or Moving Average Smoothing แบบสมมาตร เราจะใช้จำนวนของสัญญาณที่จะนำมาเฉลี่ยมากน้อยเท่าไหร่ก็ขึ้นกับความเหมาะสม แต่การใช้จำนวนของสัญญาณที่มากเกินก็จะทำให้ไปลดคุณภาพของข้อมูลดิบได้ ตัวอย่าง Moving Average Filtering ที่ใช้จำนวนของสัญญาณเท่ากับ 3 มาเฉลี่ยแสดงได้ดังนี้

$$
y(t) = \frac{1}{3}\left[x(t-1)+x(t)+x(t+1)\right] \tag{82}
$$

จากตัวอย่างจะเห็นได้ชัดว่า Moving Average Filtering แบบสมมาตรจะใช้ก็ต่อเมื่อเราต้องรู้ค่าข้อมูลที่จะเกิดขึ้นต่อไปด้วยเช่น $x(t+1)$ ดังนั้น Moving Average Filtering แบบสมมาตรมักใช้กับข้อมูลที่ได้เก็บข้อมูลทั้งการทดลองมาเรียบร้อยแล้ว สำหรับการประมวลผลแบบเวลาจริงเราไม่สามารถรู้ค่าข้อมูลที่จะเกิดขึ้นต่อไปเราจึงต้องใช้ Filter ชนิดนี้แบบไม่สมมาตร ตัวอย่าง Moving Average Filtering แบบไม่สมมาตรที่ใช้จำนวนของสัญญาณเท่ากับ 3 มาเฉลี่ยแสดงได้ดังนี้

$$y(t)=\frac{1}{3}\left[x(t-2)+x(t-1)+x(t)\right] \tag{83}$$

ในโปรแกรม MATLAB มี Function ที่ใช้ในการคำนวณสมการผลต่างและคำนวณ Filter ได้คือ 'filter' ซึ่งมีรูปแบบสมการดังนี้

$$\begin{aligned} a(1)y(n) &= b(1)x(n)+b(2)x(n-1)+\ldots+b(nb)x(n-nb+1) \\ &\quad -a(2)y(n-1)-\ldots-a(na)y(n-na+1) \end{aligned} \tag{84}$$

เราสามารถนำ Function ดังกล่าวมาใช้คำนวณ Moving Average Filtering แบบไม่สมมาตรได้ ตัวอย่างการคำนวณ Moving Average Filtering แบบไม่สมมาตรในสมการ (83) ถ้าใช้ขนาดของ Window เท่ากับ 3 ทำสามารถเขียนเป็นโปรแกรม MATLAB ได้ดังนี้

```
>> a = 1;
>> b = (1/3)*[1 1 1];
>> y = filter (b, a, x)
```

สามารถอ่านวิธีการใช้งานเพิ่มเติมได้โดยการ Search คำว่า "Difference Equations and Filtering" ใน Help ของ MATLAB

3.3.2 Median Filter

ในระบบคอมพิวเตอร์และอิเล็กทรอนิกส์รวมถึงอุปกรณ์อิเล็กทรอนิกส์กำลังต่างๆที่ใช้ในการทดลองหรือแม้แต่สายส่งสัญญาณที่ไม่ได้รับการต่อลงดินที่ถูกต้องและดีเพียงพอก็อาจจะเป็นสาเหตุทำให้เกิดสัญญาณขึ้นในระบบการทดลอง สัญญาณรบกวนที่เกิดจากการ

เหนี่ยวนำเพียงชั่วครู่ของสนามไฟฟ้าและสนามแม่เหล็กที่มีกำลังสูงกว่าสัญญาณที่เรากำลังวัดก็อาจจะเป็นสาเหตุที่ทำให้เกิดสัญญาณรบกวนประเภท Large-Magnitude Spikes กับข้อมูลที่เราได้มา ซึ่งสัญญาณรบกวนความถี่สูงประเภทนี้จะมีค่า Variant สูงจึงสามารถกำจัดออกไปได้โดยใช้ Median Filter ซึ่งสามารถแสดงความสัมพันธ์ได้ตามสมการดังนี้

$$y(t) = Median\ \{\ x(t-m),\ldots,\ x(t-1),\ x(t),\ x(t+1),\ldots,\ x(t+n)\ \} \qquad (85)$$

จากสมการ (55) จะเห็นได้ว่าค่า $y(t)$ จะมีค่าเท่ากับค่าของข้อมูลที่อยู่กึ่งกลางของชุดข้อมูลอินพุต $\{\ x(t-m),\ldots,\ x(t-1),\ x(t),\ x(t+1),\ldots,\ x(t+n)\ \}$ ที่เรียงจากน้อยไปหามาก ด้วยคุณสมบัติของ Median นี้เองทำให้สัญญาณรบกวนประเภท Large-Magnitude Spikes ถูกกำจัดออกไปได้

ในโปรแกรม MATLAB มี Function ที่ใช้ในการคำนวณ Median Filter ได้คือ ‘medfilt1’ และ ‘medfilt2’ ซึ่งใช้กับการทำ Median Filter ข้อมูล 1 มิติและ 2 มิติตามลำดับ แสดงตัวอย่างการคำนวณ Median Filter ใช้ขนาดของ Window เท่ากับ 10 ทำได้โดยใช้คำสั่งโปรแกรม MATLAB ได้ดังนี้

```
>> y = medfilt1 (x, [10])
```

3.4 เลือกวิธีการระบุเอกลักษณ์

หลังจากที่ทำการปรับปรุงข้อมูลดิบให้เหมาะสมที่จะนำมาใช้ในการหาแบบจำลองทางคณิตศาสตร์เป็นที่เรียบร้อยแล้วในขั้นต่อมาจะทำการเลือกวิธีการหาแบบจำลองว่าเป็นแบบพารามิเตอร์หรือแบบไม่ใช้พารามิเตอร์ การระบุเอกลักษณ์แบบไม่ใช้พารามิเตอร์จะนำข้อมูลสัญญาณอินพุตและเอาท์พุตมาเข้าสมการทางคณิตศาสตร์เพื่อหาผลตอบสนองทางเวลา แล้วนำข้อมูลที่ได้จากผลตอบสนองทางเวลา เช่น Time Constant, Settling Time ไปวิเคราะห์หา Transfer Function อีกทีหนึ่ง สำหรับการระบุเอกลักษณ์แบบใช้พารามิเตอร์ต้องเลือก Model Structure ที่คิดว่าสามารถใช้แสดงความสัมพันธ์ระหว่างข้อมูลสัญญาณอินพุตและเอาท์พุตได้ แล้วทำการหาค่าพารามิเตอร์ภายใน Model Structure เพื่อที่จะให้ Model Structure นั้นสามารถแสดงความสัมพันธ์ระหว่างข้อมูลสัญญาณอินพุตและเอาท์พุตได้ใกล้เคียงความจริงมากที่สุด

ในงานวิทยานิพนธ์นี้เราจะทำการศึกษาการหาแบบจำลองแบบใช้พารามิเตอร์เท่านั้น และมี Model Structure ที่เราจะทำการศึกษาดังนี้

3.4.1 ARX Model Structure

โดยทั่วไปแล้วการแสดงความสัมพันธ์ระหว่างสัญญาณอินพุต $u(t)$ และสัญญาณเอาท์พุต $y(t)$ ที่เวลา t มักจะทำได้โดยใช้สมการเชิงเส้นผลต่าง (Linear Difference Equation) แสดงได้ตามสมการ (86) ข้อสังเกตคือสมการ (86) เป็นสมการเชิงเส้นผลต่างที่ใช้แทนระบบที่ไม่ต่อเนื่อง (Discrete System) เพราะในการเก็บข้อมูลการทดลองเราทำโดยการเก็บแบบ Sampling ดังนั้น t ในที่นี้เป็น Discrete ไม่ได้เป็น Continuous

$$y(t)+a_1y(t-1)+\cdots+a_{n_a}y(t-n_a)=b_1u(t-1)+\cdots+b_{n_b}u(t-n_b) \tag{86}$$

คำย่อ “ARX” มาจากคำเต็มคือ Auto-Regression with eXtra Inputs (or eXogeneous variables) ซึ่งมีความหมายคือ “Regression” คือการใช้ค่าข้อมูลในอดีต “Auto-Regression” คือการใช้ค่าข้อมูลในอดีตของตัวเอง ในที่นี้จะหมายถึง $y(t-n_a)$ “eXtra Inputs (or eXogeneous variables)” คือการใช้ข้อมูลที่เป็นอินพุต ในที่นี้จะหมายถึง $u(t)$ ดังนั้น “ARX” จึงหมายถึง การใช้ค่าข้อมูลในอดีตของตัวเอง $y(t-n_a)$ และของข้อมูลที่เป็นอินพุต $u(t-m_b)$ มาเป็นตัวคำนวณค่าเอาท์พุต $y(t)$ เราจัดรูปสมการ (80) ให้อยู่ในรูปที่เข้าใจง่ายได้ดังนี้

$$\begin{aligned} y(t)=&-\left[a_1y(t-1)+\cdots+a_{n_a}y(t-n_a)\right] \\ &+\left[b_1u(t-1)+\cdots+b_{n_b}u(t-n_b)\right] \end{aligned} \tag{87}$$

ในความเป็นจริงแล้วเราอาจจะใช้ค่าอินพุตที่ $t-n_k$ และระบบก็อาจจะมีสัญญาณรบกวนเข้ามาปะปน จึงต้องปรับเปลี่ยนสมการให้เหมาะสมแสดงได้ดังนี้

$$y(t) = -\left[a_1 y(t-1) + \cdots + a_{n_a} y(t-n_a)\right] + \left[b_{n_k} u(t-n_k) + \cdots + b_{n_b} u(t-n_k-n_b+1)\right] + e(t) \tag{88}$$

เมื่อ $e(t)$ คือ สัญญาณรบกวนสีขาว และเรียกสมการ (88) ว่า Equation Error Model Structure ในกรณีนี้พารามิเตอร์ภายใน Model Structure ที่เราต้อง Estimate คือ

$$\boldsymbol{\theta} = \left[a_1 \quad \cdots \quad a_{n_a} \quad b_{n_k} \quad \cdots \quad b_{n_b}\right]^T \tag{89}$$

จากค่าของพารามิเตอร์ $\boldsymbol{\theta}$ จะพบว่า Order ของ ARX Model มี Order แสดงดังนี้

- n_a คือ Order ของ $y(t)$
- n_b คือ Order ของ $u(t)$
- n_k คือ Order ของการ Delay

ถ้ากำหนดสัญลักษณ์เพิ่มเติมเพื่อทำการปรับเปลี่ยนการเขียนตัวแปรดังนี้

$$A(q) = 1 + a_1 q^{-1} + \cdots + a_{n_a} q^{-n_a} \tag{90}$$

และ

$$B(q) = b_{n_k} q^{-n_k} + \cdots + b_{n_b} q^{-(n_k+n_b+1)} \tag{91}$$

จากสมการ (88) สามารถเขียนใหม่ได้เป็น

$$A(q)y(t) = B(q)u(t) + e(t) \tag{92}$$

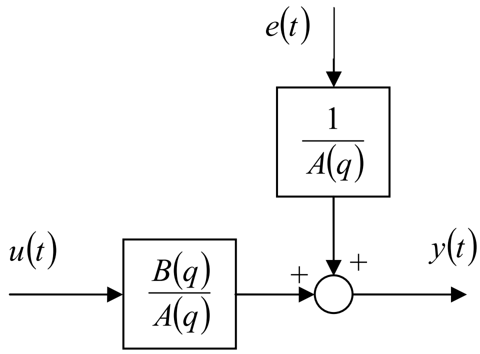
ดังนั้น

$$y(t)=\frac{B(q)}{A(q)}u(t)+\frac{1}{A(q)}e(t) \tag{93}$$

เมื่อมองเปรียบเทียบกับสมการ (68) จะได้ว่า

$$G(q,\boldsymbol{\theta})=\frac{B(q)}{A(q)}, \qquad H(q,\boldsymbol{\theta})=\frac{1}{A(q)} \tag{94}$$

สามารถเขียนเป็น Block Diagram ได้ดังนี้



ภาพที่ 28 โครงสร้างของ ARX Model

3.4.2 ARMAX Model Structure

ARMAX Model คือ Model Structure ที่ปรับปรุงมาจาก ARX Model เพื่อแก้ปัญหาในเรื่องของการที่ในตัว ARX Model นั้นไม่มีพารามิเตอร์ที่ใช้อธิบายรูปร่างลักษณะหรือคุณสมบัติของสัญญาณรบกวนสีขาว $e(t)$ โดยตรงเลย ARMAX Model แก้ปัญหานี้โดยการเพิ่มสัมประสิทธิ์ของ $e(t)$ เข้าไป การเพิ่มสัมประสิทธิ์นี้เปรียบเสมือนการทำ Moving Average ให้กับสัญญาณรบกวนสีขาว $e(t)$ คำย่อ “ARX” จึงเปลี่ยนเป็น “ARMAX” หมายถึง Auto-Regression Moving Average with eXtra Inputs (or eXogeneous Variables) แสดงสมการได้ดังนี้

$$\begin{aligned} y(t) = &-\left[a_1 y(t-1) + \cdots + a_{n_a} y(t-n_a)\right] \\ &+\left[b_{n_k} u(t-n_k) + \cdots + b_{n_b} u(t-n_k-n_b+1)\right] \\ &+\left[e(t) + c_1 e(t-1) + \cdots + c_{n_c} e(t-n_c)\right] \end{aligned} \tag{95}$$

ในกรณีนี้พารามิเตอร์ภายใน Model Structure ที่เราต้อง Estimate คือ

$$\boldsymbol{\theta} = \begin{bmatrix} a_1 & \cdots & a_{n_a} & b_{n_k} & \cdots & b_{n_b} & c_1 & \cdots & c_{n_c} \end{bmatrix}^T \tag{96}$$

จากค่าของพารามิเตอร์ $\boldsymbol{\theta}$ จะพบว่า Order ของ ARMAX Model มี Order แสดงดังนี้

- n_a คือ Order ของ $y(t)$
- n_b คือ Order ของ $u(t)$
- n_c คือ Order ของ $e(t)$
- n_k คือ Order ของการ Delay

ถ้ากำหนดสัญลักษณ์เพิ่มเติมดังนี้

$$C(q) = 1 + c_1 q^{-1} + \cdots + c_{n_c} q^{-n_c} \tag{97}$$

จากสมการ (95) สามารถเขียนใหม่ได้เป็น

$$A(q)y(t) = B(q)u(t) + C(q)e(t) \tag{98}$$

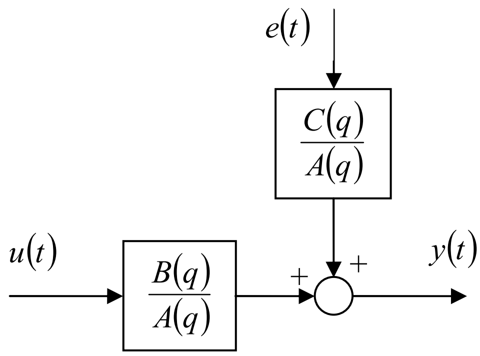
ดังนั้น

$$y(t) = \frac{B(q)}{A(q)}u(t) + \frac{C(q)}{A(q)}e(t) \tag{99}$$

เมื่อมองเปรียบเทียบกับสมการ (68) จะได้ว่า

$$G(q,\boldsymbol{\theta}) = \frac{B(q)}{A(q)}, \qquad H(q,\boldsymbol{\theta}) = \frac{C(q)}{A(q)} \tag{100}$$

สามารถเขียนเป็น Block Diagram ได้ดังนี้

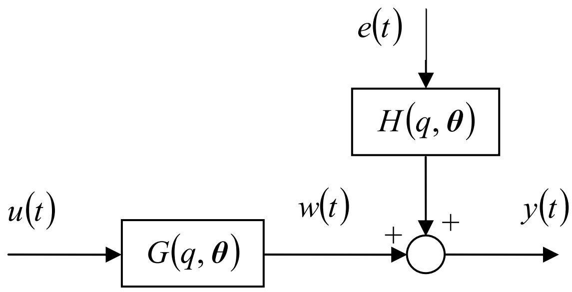


ภาพที่ 29 โครงสร้างของ ARMAX Model

จากการสร้าง ARMAX Model เราจะเห็นว่า ประโยชน์ของ ARMAX Model ที่ดีกว่า ARX Model ในทางปฏิบัติคือการที่ ARMAX Model มีตัวแปรที่จะสามารถจำลองลักษณะของสัญญาณรบกวนได้ ดังนั้น ถ้าระบบที่เราต้องการระบุเอกลักษณ์ถูกรบกวนด้วยสัญญาณรบกวน การใช้ ARMAX Model มาเป็น Model Structure ให้กับการระบุเอกลักษณ์ก็น่าจะให้ผลดีว่าการใช้ ARX Model เป็น Model Structure

3.4.3 Output Error (OE) Model Structure

จากภาพที่ 20 ถ้าเราสมมติค่าสัญญาณ $w(t)$ ขึ้นมาใหม่ให้หมายถึงสัญญาณเอาท์พุตของระบบที่ยังไม่ถูกรบกวนโดยสัญญาณรบกวน $e(t)$ แสดงดังภาพที่ 30



ภาพที่ 30 แสดงการสมมติสัญญาณ $w(t)$ เพิ่มเติมจากภาพที่ 20

จากภาพที่ 30 มีสมการคือ

$$y(t) = w(t) + e(t) \tag{101}$$

จากหลักการของ Linear Model เราจะมองว่าสัญญาณเอาท์พุต $w(t)$ ที่เวลา t สามารถสร้างได้โดยใช้สมการเชิงเส้นผลต่างแสดงดังนี้

$$\begin{aligned} w(t) = &-\left[f_1 w(t-1) + \cdots + f_{n_f} w(t-n_f)\right] \\ &+\left[b_{n_k} u(t-n_k) + \cdots + b_{n_b} u(t-n_k-n_b+1)\right] \end{aligned} \tag{102}$$

ในกรณีนี้พารามิเตอร์ภายใน Model Structure ที่เราต้อง Estimate คือ

$$\boldsymbol{\theta} = \begin{bmatrix} b_{n_k} & \cdots & b_{n_b} & f_1 & \cdots & f_{n_f} \end{bmatrix}^T \tag{103}$$

จากค่าของพารามิเตอร์ $\boldsymbol{\theta}$ จะพบว่า Order ของ OE Model มี Order แสดงดังนี้

- n_b คือ Order ของ $u(t)$
- n_f คือ Order ของ $w(t)$
- n_k คือ Order ของการ Delay

กำหนดสัญลักษณ์เพิ่มเติมดังนี้

$$F(q) = 1 + f_1 q^{-1} + \cdots + f_{n_f} q^{-n_f} \qquad (104)$$

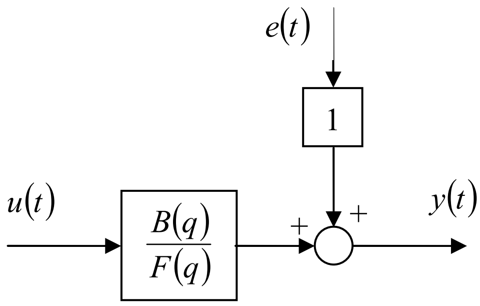
จากสมการ (102) สามารถเขียนใหม่ได้เป็น

$$y(t) = \frac{B(q)}{F(q)} u(t) + e(t) \qquad (105)$$

เมื่อมองเปรียบเทียบกับสมการ (68) จะได้ว่า

$$G(q, \boldsymbol{\theta}) = \frac{B(q)}{F(q)}, \qquad H(q, \boldsymbol{\theta}) = 1 \qquad (106)$$

สามารถเขียนเป็น Block Diagram ได้ดังนี้



ภาพที่ 31 โครงสร้างของ Output Error (OE) Model

3.4.4 Linear Regression and Regression Vector

การที่เราจะสามารถ Estimate ค่าของ $\boldsymbol{\theta}$ โดยใช้วิธี Least-Squares เพื่อให้ Model Structure นั้นเป็นแบบจำลองทางคณิตศาสตร์ให้กับระบบที่เราต้องการระบุเอกลักษณ์ได้นั้น เราจะต้องจัดรูปของสมการของตัว Predictor ให้อยู่ในรูปของ Linear Regression โดยมี $\boldsymbol{\theta}$ เป็น

Regression Vector ให้ได้ ดังนั้นในหัวข้อนี้เราจะกล่าวถึงการสร้าง Predictor ของ Model Structure แต่ละแบบ และการจัดรูปตัว Predictor ดังกล่าวให้อยู่ในรูปของ Linear Regression ให้ได้

3.4.4.1 Linear Regression and Regression Vector of ARX Model

เมื่อเราเลือก ARX Model มาเป็น model structure ที่จะใช้แสดงความสัมพันธ์แล้ว ขั้นตอนต่อไปก็คือการหาค่าพารามิเตอร์ภายใน ARX Model เพื่อให้ ARX Model ทำนายค่าให้ตรงกับความเป็นจริงมากที่สุด เราจะมอง ARX Model เป็นตัว Predictor ชนิดหนึ่งซึ่งเราสามารถหาสมการ Predictor ของ ARX Model ได้จากการแทนค่า

$$G(q,\boldsymbol{\theta}) = \frac{B(q)}{A(q)}, \qquad H(q,\boldsymbol{\theta}) = \frac{1}{A(q)} \tag{107}$$

ลงในสมการ (69) ได้ผลดังนี้

$$\hat{y}(t\,|\,\boldsymbol{\theta}) = B(q)u(t) + [1 - A(q)]y(t) \tag{108}$$

ถ้ากำหนด Regression Vector ดังนี้

$$\varphi(t) = [-y(t-1) \quad \cdots \quad -y(t-n_a) \quad u(t-n_k) \quad \cdots \quad u(t-n_k-n_b+1)]^T \tag{109}$$

เราจะสามารถเขียนสมการ Predictor ที่อยู่ในรูปของ Linear Regression ได้ดังนี้

$$\hat{y}(t\,|\,\boldsymbol{\theta}) = \varphi^T(t)\boldsymbol{\theta} = \boldsymbol{\theta}^T\varphi(t) \tag{110}$$

3.4.4.2 Linear Regression and Regression Vector of ARMAX Model

วิธีการหาค่าพารามิเตอร์ภายใน ARMAX Model เพื่อให้ ARMAX Model ทำนายค่าให้ตรงกับความเป็นจริงมากที่สุดมีลักษณะคล้ายกันกับวิธีที่ใช้กับ ARX Model แสดงได้ดังนี้ เราจะมอง ARMAX Model เป็นตัว Predictor ชนิดหนึ่งซึ่งเราสามารถหาสมการ Predictor ของ ARMAX Model ได้จากการแทนค่า

$$G(q,\boldsymbol{\theta}) = \frac{B(q)}{A(q)}, \qquad H(q,\boldsymbol{\theta}) = \frac{C(q)}{A(q)} \tag{111}$$

ลงในสมการ (69) ได้ผลดังนี้

$$\hat{y}(t\,|\,\boldsymbol{\theta}) = \frac{B(q)}{C(q)}u(t) + \left[1 - \frac{A(q)}{C(q)}\right]y(t) \tag{112}$$

จัดรูปใหม่ได้เป็น

$$C(q)\hat{y}(t\,|\,\boldsymbol{\theta}) = B(q)u(t) + [C(q) - A(q)]y(t) \tag{113}$$

เราสามารถเขียนสมการ (113) ให้อยู่ในรูปของ Linear Regression ได้โดยนำ $[1 - C(q)]\hat{y}(t\,|\,\boldsymbol{\theta})$ บวกเข้าทั้ง 2 ข้างของสมการ (113) ได้ผลคือ

$$\hat{y}(t\,|\,\boldsymbol{\theta}) = B(q)u(t) + [1 - A(q)]y(t) + [C(q) - 1][y(t) - \hat{y}(t\,|\,\boldsymbol{\theta})] \tag{114}$$

จาก

$$\varepsilon(t,\boldsymbol{\theta}) = y(t) - \hat{y}(t\,|\,\boldsymbol{\theta}) \tag{115}$$

ถ้ากำหนด Regression Vector ดังนี้

$$\varphi(t,\boldsymbol{\theta}) = [-y(t-1) \quad \cdots \quad -y(t-n_a) \quad u(t-n_k) \quad \cdots \quad u(t-n_k-n_b+1) \quad \varepsilon(t-1,\boldsymbol{\theta}) \quad \cdots \quad \varepsilon(t-n_c,\boldsymbol{\theta})]^T \tag{116}$$

เราจะสามารถเขียนสมการ Predictor ที่อยู่ในรูปของ Linear Regression ได้ดังนี้

$$\hat{y}(t\,|\,\boldsymbol{\theta}) = \varphi^T(t,\boldsymbol{\theta})\boldsymbol{\theta} = \boldsymbol{\theta}^T\varphi(t,\boldsymbol{\theta}) \tag{117}$$

3.4.4.3 Linear Regression and Regression Vector of Output Error (OE) Model

จากภาพที่ 30 จะเห็นว่า $w(t)$ เป็นค่าของสัญญาณที่เราสมมติขึ้นที่อยู่ภายใน Output Error Model จึงไม่สามารถวัดได้โดยตรง แสดงว่าค่าของ $w(t)$ ก็ต้องขึ้นถูกสร้างขึ้นมาซึ่งจะมีค่าเป็นเท่าไรก็จะต้องขึ้นกับค่าพารามิเตอร์ $\boldsymbol{\theta}$ ด้วย ดังนั้นเราจึงต้องเขียน $w(t)$ ใหม่เป็น $w(t, \boldsymbol{\theta})$ และเขียนสมการ (102) ใหม่เป็น

$$\begin{aligned} w(t, \boldsymbol{\theta}) = &-\left[f_1 w(t-1, \boldsymbol{\theta}) + \cdots + f_{n_f} w(t-n_f, \boldsymbol{\theta})\right] \\ &+ \left[b_{n_k} u(t-n_k) + \cdots + b_{n_b} u(t-n_b)\right] \end{aligned} \tag{118}$$

เราจะมอง Output Error Model เป็นตัว Predictor ชนิดหนึ่งซึ่งเราสามารถหาสมการ Predictor ของ Output Error Model ได้จากการแทนค่า

$$G(q, \boldsymbol{\theta}) = \frac{B(q)}{F(q)}, \qquad H(q, \boldsymbol{\theta}) = 1 \tag{119}$$

ลงในสมการ (69) ได้ผลดังนี้

$$\hat{y}(t \mid \boldsymbol{\theta}) = \frac{B(q)}{F(q)} u(t) = w(t, \boldsymbol{\theta}) \tag{120}$$

ถ้ากำหนด Regression Vector ดังนี้

$$\varphi(t, \boldsymbol{\theta}) = \left[u(t-n_k) \quad \cdots \quad u(t-n_k-n_b+1) \quad -w(t-1, \boldsymbol{\theta}) \quad \cdots \quad -w(t-n_c, \boldsymbol{\theta})\right]^T \tag{121}$$

เราจะสามารถเขียนสมการ Predictor ที่อยู่ในรูปของ Linear Regression ได้ดังนี้

$$\hat{y}(t \mid \boldsymbol{\theta}) = \varphi^T(t, \boldsymbol{\theta}) \boldsymbol{\theta} = \boldsymbol{\theta}^T \varphi(t, \boldsymbol{\theta}) \tag{122}$$

3.5 วิธีการหาค่าพารามิเตอร์ที่ดีที่สุดของ Model Structure (Parameter Estimation Methods)

หลังจากที่เราได้เลือก Model Structure ที่คาดว่า Model Structure ดังกล่าวจะสามารถอธิบายระบบได้แล้ว จากนั้นเราต้องทำการหาค่าพารามิเตอร์ของ Model Structure โดยมีเป้าหมายคือค่าพารามิเตอร์ที่เราหาได้แล้วนำไปใช้นั้นจะต้องสามารถทำให้ Model Structure นั้นสามารถแสดงความสัมพันธ์ระหว่างสัญญาณอินพุตและสัญญาณเอาท์พุตได้ใกล้เคียงความจริงมากที่สุด หมายความว่าเมื่อนำ Model Structure ที่เราสร้างขึ้นไปทดสอบโดยการใส่สัญญาณอินพุตจะต้องทำให้สัญญาณเอาท์พุตที่เกิดขึ้นจาก Model มีค่าเท่ากันหรือใกล้เคียงกับสัญญาณเอาท์พุตที่เกิดขึ้นจากระบบจริง ณ ที่เวลาเดียวกันเมื่อใช้ค่าสัญญาณอินพุตที่เหมือนกัน และเราจะใช้ Model Structure นี้เป็นแบบจำลองทางคณิตศาสตร์ของระบบ ดังนั้นสิ่งที่สำคัญในการหาแบบจำลองทางคณิตศาสตร์ของระบบคือวิธีการคำนวณหาค่าพารามิเตอร์ที่ดีที่สุดของ Model Structure นั้น

ก่อนที่จะศึกษาในขั้นตอนต่างๆถัดไป จะขอกำหนดความหมายของสัญลักษณ์และตัวแปรต่างๆก่อน ในตอนนี้เราได้เลือก Model Structure แล้ว Model Structure จะสามรถเขียนแทนด้วยสัญลักษณ์ M ใน Model Structure M ที่เราเลือกมาจะมีพารามิเตอร์ภายในของ Model Structure ที่เราต้องคำนวณหาค่าที่ดีที่สุด เราจะสมมติตัวแปร $\boldsymbol{\theta}$ เรียกว่าเวกเตอร์พารามิเตอร์ เป็นตัวแปรที่แทนกลุ่มของค่าพารามิเตอร์ภายในทั้งหมดที่ต้องการหาค่าของ Model Structure ดังนั้นตอนนี้เราจะมองได้ว่าการสร้างแบบจำลองทางคณิตศาสตร์ก็คือการหาค่า $\boldsymbol{\theta}$ ที่ดีที่สุดนั่นเอง ขณะนี้ค่าของ Model Structure M จะขึ้นกับพารามิเตอร์ภายในคือ $\boldsymbol{\theta}$ ดังนั้นเขียน $M(\boldsymbol{\theta}) : \boldsymbol{\theta} \in D_M \subset \boldsymbol{R}^d$ โดยที่ D_M คือ โดเมนของ Model Structure M เราสามารถกำหนดเซตของ Model Structure M ได้ดังนี้

$$M^* = \{M(\boldsymbol{\theta}) \mid \boldsymbol{\theta} \in D_M\} \tag{123}$$

หน้าที่ของ Model Structure คือการทำนายสัญญาณเอาท์พุต แต่ละ Model Structure ก็จะมีวิธีการทำนายที่แตกต่างกันไป ดังนั้นต่อไปเราจะเรียก Model Structure ว่าเป็น Predictor

Predictor สามารถจะถูกมองว่าอยู่ในรูปแบบของ Linear Filter ก็ได้ ซึ่งจะมีรูปแบบดังนี้

$$M(\boldsymbol{\theta}) \quad : \quad \hat{y}(t \mid \boldsymbol{\theta}) = W_y(q, \boldsymbol{\theta})y(t) + W_u(q, \boldsymbol{\theta})u(t) \tag{124}$$

จากสมการ (125) จะเห็นว่ามันก็มีรูปแบบคล้ายๆกันกับ One-Step-Ahead Prediction ของระบบ

$$y(t) = G(q, \boldsymbol{\theta})y(t) + H(q, \boldsymbol{\theta})u(t) \tag{125}$$

โดยมี

$$W_y(q, \boldsymbol{\theta}) = \left[1 - H^{-1}(q, \boldsymbol{\theta})\right] \tag{126}$$

$$W_u(q, \boldsymbol{\theta}) = H^{-1}(q, \boldsymbol{\theta})G(q, \boldsymbol{\theta}) \tag{127}$$

ในขั้นตอนนี้เราได้เก็บข้อมูลจากการทดลองมาเรียบร้อยแล้ว เราจะนำข้อมูลที่ได้จากการทดลองมาเขียนเป็นเวกเตอร์ Z^N ดังนี้

$$Z^N = \left[y(0), u(0), y(1), u(1), y(2), u(2), \ldots, y(N), u(N)\right] \tag{128}$$

จะเห็นได้ว่าการระบุเอกลักษณ์ก็คือการนำข้อมูล Z^N ไปเป็นข้อมูลในการสร้างแบบจำลองทางคณิตศาสตร์ให้กับระบบ ข้อมูล Z^N จะนำไปใช้ในการหาค่า $\boldsymbol{\theta}$ ที่ดีที่สุดให้กับ Model Structure ที่เราเลือก ในความเป็นจริงแล้วการใช้ข้อมูล Z^N ซึ่งได้จากการทดลองมาเป็นตัวหาค่า $\boldsymbol{\theta}$ นี้ เราจะไม่สามารถหาค่า $\boldsymbol{\theta}$ ที่ทำให้ค่าของสัญญาณเอาท์พุตของ M ตรงกับสัญญาณเอาท์พุตที่เกิดขึ้นจากระบบจริงได้ 100% แต่เราก็สามารถหาค่า $\boldsymbol{\theta}$ ที่เรียกว่าทำให้สัญญาณเอาท์พุตของ M ตรงกับสัญญาณเอาท์พุตที่เกิดขึ้นจากระบบจริงได้ใกล้เคียงความเป็นจริงมากที่สุดเราจะเรียก $\boldsymbol{\theta}$ ที่ดีที่สุดนี้ว่า $\hat{\boldsymbol{\theta}}$ และ $\hat{\boldsymbol{\theta}}$ ที่สร้างมากจากข้อมูลอินพุต-เอาท์พุตจำนวน N ชุดหรือ Z^N จะเขียนเป็น $\hat{\boldsymbol{\theta}}_N$ ดังนั้นตอนนี้เราจะได้ Model Structure คือ $M\left(\hat{\boldsymbol{\theta}}_N\right)$ โดยทั่วไปแล้ววิธีการที่เราหา $\hat{\boldsymbol{\theta}}_N$ จาก Z^N เราจะเรียกว่า Parameter Estimation Method เขียนเป็น Mapping ได้ดังนี้

$$Z^N \to \hat{\boldsymbol{\theta}}_N \in D_M \tag{129}$$

ดังนั้นวิธีการระบุเอกลักษณ์ก็คือการทำ Parameter Estimation Method นั่นเอง

ตามที่ได้กล่าวมา $M\left(\hat{\boldsymbol{\theta}}_N\right)$ เป็น Predictor ที่สามารถทำนายค่าของสัญญาณเอาท์พุต Predictor ตัวนี้ก็คือแบบจำลองทางคณิตศาสตร์ของระบบ Predictor ที่ดีก็คือ Predictor ที่มีการ

ทำนายค่าสัญญาณเอาท์พุตผิดพลาดน้อยเทียบกับค่าสัญญาณเอาท์พุตที่เราเก็บข้อมูลได้ ฉะนั้น $\hat{\boldsymbol{\theta}}_N$ ที่เราหามาได้ก็จะเป็นตัวบอกถึงคุณภาพในการทำนายค่าสัญญาณเอาท์พุตของ Predictor ดังนั้นในขั้นตอนการหาค่า $\hat{\boldsymbol{\theta}}_N$ จึงจะต้องทำการทดสอบ $M\left(\hat{\boldsymbol{\theta}}_N\right)$ ว่ามีความถูกต้องในการทำนายค่าของสัญญาณเอาท์พุตมากแค่ไหน $\hat{\boldsymbol{\theta}}_N$ เราสามารถเขียนสมการของความผิดพลาดในการทำนายค่าของสัญญาณเอาท์พุต ณ ที่เวลา t ใดๆได้ดังนี้

$$\varepsilon\left(t, \hat{\boldsymbol{\theta}}_N\right) = y(t) - \hat{y}\left(t \mid \hat{\boldsymbol{\theta}}_N\right) \tag{130}$$

$\varepsilon\left(t, \hat{\boldsymbol{\theta}}_N\right)$ เรียกว่า ความผิดพลาดในการทำนาย (Prediction-Error) เพื่อให้ Predictor มีคุณภาพดีเราจะต้องเลือก $\hat{\boldsymbol{\theta}}_N$ ที่ทำให้ $\varepsilon\left(t, \hat{\boldsymbol{\theta}}_N\right)$ $\forall t = 1, 2, 3, \ldots, N$ มีค่าน้อยที่สุดเท่าที่จะเป็นไปได้ คำว่า "ค่าน้อยที่สุด" มันยากที่จะนิยามเนื่องจากไม่มีกฎเกณฑ์ใดๆมาบอกว่าน้อยเท่าใดๆถึงจะเป็นที่น่าพอใจ เราจึงเลือกวิธีการของ Norm มาเป็นตัวเปรียบเทียบขนาดของค่าความผิดพลาด $\varepsilon\left(t, \hat{\boldsymbol{\theta}}_N\right)$

เราจะเริ่มต้นวิธีการเลือก $\boldsymbol{\theta}$ ที่ทำให้ $\varepsilon\left(t, \hat{\boldsymbol{\theta}}_N\right)$ $\forall t = 1, 2, 3, \ldots, N$ มีค่าน้อยที่สุดโดยการเลือกวิธีการที่จะบอกขนาดของ $\varepsilon\left(t, \hat{\boldsymbol{\theta}}_N\right)$ เพราะว่า $\varepsilon\left(t, \hat{\boldsymbol{\theta}}_N\right)$ เป็นเวกเตอร์ที่อยู่ใน R^N ดังนั้นวิธีการที่มักจะใช้ขนาดของเวกเตอร์ดังกล่าวก็คือการใช้ Norm R^N จากนั้นเราเริ่มพิจารณาโดยการใส่ Stable Linear Filter ให้กับ $\varepsilon(t, \boldsymbol{\theta})$ แสดงได้ดังนี้

$$\varepsilon_F(t, \boldsymbol{\theta}) = L(q)\varepsilon(t, \boldsymbol{\theta}), \quad 1 \le t \le N \tag{131}$$

เลือกฟังก์ชั่น $l(\cdot)$ เป็น Scalar Positive Function ใดๆเพื่อนำมาเปลี่ยนให้ค่าของ $\varepsilon_F(t, \boldsymbol{\theta})$ เป็นค่าบวก แสดงตัวอย่างได้ดังนี้

$$l(\varepsilon_F) = \frac{1}{2}\varepsilon_F^2 \tag{132}$$

ใช้ Norm เพื่อบอกขนาด แสดงได้ดังนี้

$$V_N\left(\boldsymbol{\theta}, Z^N\right) = \frac{1}{N}\sum_{t=1}^{N} l(\varepsilon_F(t, \boldsymbol{\theta})) \tag{133}$$

จากสมการ (133) จะเห็นได้ว่า $V_N\left(\boldsymbol{\theta}, Z^N\right)$ จะเป็นค่า Positive Scalar ที่ขึ้นกับค่าของ $\boldsymbol{\theta}$ ดังนั้นการที่เราจะหาค่า $\boldsymbol{\theta}$ ที่ทำให้ $\varepsilon\left(t, \hat{\boldsymbol{\theta}}_N\right)$ มีค่าน้อยที่สุดก็เปรียบเสมือนกับการหา $\boldsymbol{\theta}$ ที่ทำให้ $V_N\left(\hat{\boldsymbol{\theta}}_N, Z^N\right)$ มีค่าน้อยที่สุดเช่นกัน สามารถเขียนเป็นสมการได้ดังนี้

$$\hat{\boldsymbol{\theta}}_N = \hat{\boldsymbol{\theta}}_N\left(Z^N\right) = \underset{\boldsymbol{\theta} \in D_M}{\arg\min}\ V_N\left(\boldsymbol{\theta}, Z^N\right) \tag{134}$$

ในที่นี้ $\arg\min$ หมายถึง “The Minimizing Argument of The Function” สำหรับกรณีที่ $\hat{\boldsymbol{\theta}}_N$ ที่ทำให้ $\varepsilon\left(t, \hat{\boldsymbol{\theta}}_N\right)$ มีค่าน้อยที่สุดมีมากกว่า 1 ค่าเราก็จะมองว่า $\arg\min$ เป็นเซตของ $\hat{\boldsymbol{\theta}}_N$ ที่ทำให้ $\varepsilon\left(t, \hat{\boldsymbol{\theta}}_N\right)$ มีค่าน้อยที่สุด

จากความรู้ทางคณิตศาสตร์ที่มากมายทำให้เรามีวิธีที่จะให้หา $\hat{\boldsymbol{\theta}}_N$ ให้เป็นไปตามสมการ (134) ได้หลายวิธี วิธีต่างๆเหล่านี้จะเรียกว่า Prediction-Error Identification Methods (PEM) บางวิธีก็จะมีชื่อเฉพาะเป็นของตัวเองซึ่ง การตั้งชื่อเฉพาะก็อาจจะมีที่มาจากวิธีการเลือก $L(\cdot), l(\cdot)$ หรือ Model Structure ที่ต่างกัน หรือในบางกรณีก็อาจจะมีที่มาจากวิธีการ Minimizing Argument ที่ใช้

3.5.1 การทำ PEM โดยใช้ Least-Squares Method

หลักจากเราสามารถหาสมการของ Predictor ที่อยู่ในรูปของสมการ Linear Regression โดยมี $\varphi(t)$ เป็น Regression Vector ได้แล้ว เราจะใช้สมการ Predictor ดังกล่าวในการทำนายค่าของเอาท์พุต ณ ที่เวลา t คือ $\hat{y}(t \mid \boldsymbol{\theta})$ โดยที่ $\boldsymbol{\theta}$ เป็นค่าที่ได้จากการ Estimate ฉะนั้นค่าที่ตัว Predictor ทำนายออกมานี้จะมีความถูกต้องมาเพียงได้ก็ขึ้นกับค่าพารามิเตอร์ $\boldsymbol{\theta}$ จากสมการ (130) และสมการ (133) เราสามารถแสดง Function $V_N\left(\boldsymbol{\theta}, Z^N\right)$ ของค่าความผิดพลาดของเอาท์พุตที่ Predictor ณ ที่เวลา t ได้ดังนี้

$$\begin{aligned} V_N\left(\boldsymbol{\theta}, Z^N\right) &= \frac{1}{N}\sum_{t=1}^{N}\left[y(t) - \hat{y}(t \mid \boldsymbol{\theta})\right]^2 \\ &= \frac{1}{N}\sum_{t=1}^{N}\left[y(t) - \varphi^T(t)\boldsymbol{\theta}\right]^2 \end{aligned} \tag{135}$$

จากสมการ (134) เพื่อให้ได้ค่าพารามิเตอร์ $\boldsymbol{\theta}$ ที่ดีที่สุดเราจะต้องเลือก $\boldsymbol{\theta}$ ที่ทำให้ $V_N\left(\boldsymbol{\theta}, Z^N\right)$ มีค่าน้อยที่สุด ซึ่งสามารถทำได้ดังนี้

เนื่องจาก $V_N\left(\boldsymbol{\theta}, Z^N\right)$ เป็นฟังค์ชั่น Quadratic ของ $\boldsymbol{\theta}$ เราจึงสามารถหาค่า $\boldsymbol{\theta}$ ที่ทำให้ $V_N\left(\boldsymbol{\theta}, Z^N\right)$ มีค่าน้อยที่สุดได้โดยการ Differential $V_N\left(\boldsymbol{\theta}, Z^N\right)$ เทียบกับ $\boldsymbol{\theta}$ แล้วกำหนดให้เท่ากับ 0 แสดงได้ดังนี้

$$0 = \frac{d}{d\boldsymbol{\theta}} V_N\left(\boldsymbol{\theta}, Z^N\right) = \frac{2}{N}\sum_{t=1}^{N} \varphi(t)\left[y(t) - \varphi^T(t)\boldsymbol{\theta}\right] \tag{136}$$

จะได้

$$\sum_{t=1}^{N} \varphi(t) y(t) = \sum_{t=1}^{N} \varphi(t)\varphi^T(t)\boldsymbol{\theta} \tag{137}$$

ดังนั้นจะได้

$$\hat{\boldsymbol{\theta}}_N = \left[\sum_{t=1}^{N} \varphi(t) y(t)\right]^{-1} \sum_{t=1}^{N} \varphi(t)\varphi^T(t) \tag{138}$$

เราก็จะได้ค่า $\hat{\boldsymbol{\theta}}_N$ ตามต้องการ

การที่เราจะใช้วิธี Least-Squares Method เพื่อที่จะ Estimate ค่า $\hat{\boldsymbol{\theta}}$ ได้นั้น สัญญาณรบกวนในสมการของ ARX Model, ARMAX Model และ OE Model ต้องเป็นสัญญาณรบกวนสีขาวคือ $e(t)$ ซึ่งจะทำให้ เมื่อเราใช้จำนวนของข้อมูลมีค่ามากๆ ($N \to \infty$) ในการ Estimate $\hat{\boldsymbol{\theta}}$ ที่เราได้นั้นจะมีค่าใกล้เคียงกับค่า $\boldsymbol{\theta}_0$ แต่ถ้าสัญญาณรบกวนไม่ใช่สัญญาณรบกวนสีขาวจะทำให้ไม่ว่าเราจะใช้ $N \to \infty$ ก็ตาม $\hat{\boldsymbol{\theta}}$ ที่เราได้มานั้นจะไม่มีทางเข้าหาค่า $\boldsymbol{\theta}_0$ นั่นก็หมายความว่าเราไม่สามารถใช้วิธี Least-Squares ในการ Estimate ค่า $\hat{\boldsymbol{\theta}}$ ได้

การที่สัญญาณรบกวนไม่ใช่สัญญาณรบกวนสีขาวส่วนหนึ่งสาเหตุอาจจะเกิดจากการที่สัญญาณรบกวนที่เกิดขึ้นนั้นไม่เป็นอิสระจากสัญญาณอินพุต $u(t)$ เราสามารถแก้ได้โดยใช้ Linear

Filter เข้าไปทำให้สัญญาณรบกวนเป็นอิสระจากสัญญาณอินพุต $u(t)$ ได้แล้วค่อยใช้วิธี Least-Squares ในการ Estimate ค่า $\hat{\boldsymbol{\theta}}$ (Lennart Ljung, 1999: 207)

3.5.2 การทำ PEM โดยใช้ Maximum likelihood Method

นอกจากวิธี Least-Squares ยังมีวิธีที่จะใช้ Estimate ค่าพารามิเตอร์ $\boldsymbol{\theta}$ ที่ดีที่สุด ะ โดยใช้ความน่าจะเป็นและสถิติเข้ามาช่วยคือวิธี Maximum Likelihood (Lennart Ljung, 1999: 217) ก่อนที่เราจะทำเช่นนั้นได้ต้องกำหนด Probabilistic Model ของระบบ Dynamic ก่อน จากที่เราเลือก Model Structure ที่จะใช้ทำนายค่าของสัญญาณเอาท์พุตของระบบดังนี้

$$\begin{aligned} M(\boldsymbol{\theta}) \quad : \quad \hat{y}(t \mid \boldsymbol{\theta}) &= g\left(t, Z^{t-1}; \boldsymbol{\theta}\right) \\ \varepsilon(t, \boldsymbol{\theta}) &= y(t) - \hat{y}(t \mid \boldsymbol{\theta}) \end{aligned} \tag{139}$$

ถ้าค่าของ $\varepsilon\left(t, \hat{\boldsymbol{\theta}}_N\right)$ เป็นอิสระ และมี PDF เท่ากับ $f_e(x, t; \boldsymbol{\theta})$ หรือพูดง่ายๆว่า $\varepsilon\left(t, \hat{\boldsymbol{\theta}}_N\right)$ คือสัญญาณรบกวนสีขาวนั่นเอง เราจะเรียกสมการ (139) ว่า Complete Probabilistic Model

ตอนนี้เราจะต้องกำหนด Likelihood Function ของ Complete Probabilistic Model ของระบบ Dynamic ดังกล่าวโดยเริ่มพิจารณาจากสมการ (139) เขียนใหม่เป็น

$$y(t) = \hat{y}(t \mid \boldsymbol{\theta}) + \varepsilon(t, \boldsymbol{\theta}) \tag{140}$$

เราสามารถหา Joint PDF ของ y^N given u^N ได้โดยการแทนค่า x ใน $f_e(x, t; \boldsymbol{\theta})$ ด้วย $y(t) - \hat{y}(t \mid \boldsymbol{\theta})$ (Lennart Ljung, 1999: 163) ผลที่ได้แสดงตามนี้

$$\begin{aligned} \bar{f}_y\left(\boldsymbol{\theta}, y^N\right) &= \prod_{t=1}^{N} f_e\left(y(t) - \hat{y}(t \mid \boldsymbol{\theta}), t; \boldsymbol{\theta}\right) \\ &= \prod_{t=1}^{N} f_e\left(\varepsilon(t \mid \boldsymbol{\theta}), t; \boldsymbol{\theta}\right) \end{aligned} \tag{141}$$

เราจะเรียก $\bar{f}_y\left(\boldsymbol{\theta}, y^N\right)$ ว่า Likelihood Function ของ Complete Probabilistic Model ของระบบ Dynamic ดังกล่าว Maximum Likelihood หมายถึงการทำให้ Joint PDF ของ y^N given u^N มีค่ามากที่สุดหรือก็คือการทำให้ Likelihood Function มีค่ามากที่สุดนั่นเองซึ่งเขียนเป็นสมการได้ดังนี้

$$\hat{\boldsymbol{\theta}}_{ML}\left(y^N\right) = \arg\max_{\boldsymbol{\theta}} \bar{f}_y\left(\boldsymbol{\theta}, y^N\right) \tag{142}$$

เนื่องจาก Log Function เป็น Monotonically Increasing ฉะนั้นการทำให้ $\bar{f}_y\left(\boldsymbol{\theta}, y^N\right)$ มีค่ามากที่สุดในสมการ (142) ก็ย่อมมีค่าเทียบเท่ากับสมการ (143)

$$\arg\max_{\boldsymbol{\theta}} \frac{1}{N}\log \bar{f}_y\left(\boldsymbol{\theta}, y^N\right) \quad = \quad \arg\max_{\boldsymbol{\theta}} \frac{1}{N}\sum_{t=1}^{N}\log f_e\left(\varepsilon\left(t \mid \boldsymbol{\theta}\right), t; \boldsymbol{\theta}\right) \tag{143}$$

ถ้าเรากำหนดให้

$$l\left(\varepsilon, \boldsymbol{\theta}, t\right) = -\log f_e\left(\varepsilon, t; \boldsymbol{\theta}\right) \tag{144}$$

จากสมการ (143) เราสามารถเขียนใหม่ได้ว่า

$$\hat{\boldsymbol{\theta}}_{ML}\left(y^N\right) = \quad \arg\min_{\boldsymbol{\theta}} \frac{1}{N}\sum_{t=1}^{N}\log l\left(\varepsilon\left(t, \boldsymbol{\theta}\right), \boldsymbol{\theta}, t\right) \tag{145}$$

จากสมการ (145) เราจะเห็นว่าการ Estimate พารามิเตอร์ $\boldsymbol{\theta}$ ที่ดีที่สุดด้วยวิธี Maximum Likelihood จะมีลักษณะเหมือนกันกับการทำ PEM แสดงตามสมการ (134)

3.6 ทำการทดสอบคุณสมบัติของ Model Structure ที่ได้

หลักจากที่เราได้ทำการหา Model Structure ได้แล้ว เราจะต้องทำการตรวจสอบ Model Structure ดังกล่าวว่ามีความเหมาะสมตรงตามที่เราต้องการมากน้อยเพียงไร การทดสอบในงานวิทยานิพนธ์นี้จะทำด้วยกัน 3 วิธี คือ

3.6.1 การทดสอบค่าความถูกต้องของ Model Output

การตรวจสอบ Model Output คือการนำข้อมูลสัญญาณอินพุตและเอาท์พุตที่ได้จากการทดลองมาเป็น Validation Data โดยจะทำการใส่สัญญาณอินพุตให้กับ Model Structure แล้วนำสัญญาณเอาท์พุตที่ Model Structure สร้างขึ้นมาเทียบกับสัญญาณเอาท์พุตที่ได้จากการทดลองเมื่อใช้สัญญาณอินพุตที่เวลาเดียวกัน การเปรียบเทียบจะใช้สมการดังต่อไปนี้

$$\%\ \text{ความถูกต้อง} = 100 \times \left[1 - \frac{\sqrt{\sum_{t=1}^{N} [\hat{y}(t) - y(t)]^2}}{\sqrt{\sum_{t=1}^{N} \left[y(t) - \frac{1}{N} \sum_{t=1}^{N} y(t) \right]^2}} \right] \tag{146}$$

3.6.2 การตรวจสอบ Transient Response หรือ Step Response

วิธีนี้คือการนำ Model Structure มาหา Step Response ด้วยวิธีการทางคณิตศาสตร์แล้วใช้ข้อมูลจาก Step Response เพื่อหาค่า Time Constant แล้วนำไปเปรียบเทียบกับค่า Time Constant ที่ได้จากการทดลองกับระบบจริง

3.6.3 การวิเคราะห์ Residual

ตรวจสอบด้วยการวิเคราะห์ Residual (Lennart Ljung, 1999: 511-516) หมายถึงการนำค่า $\varepsilon(t, \hat{\boldsymbol{\theta}}_N)$ มาตรวจสอบความเป็นอิสระ เราทราบกันแล้วว่าหน้าที่ของ Model Structure คือการทำนายค่า $\hat{y}(t \mid \hat{\boldsymbol{\theta}}_N)$ ซึ่งจะทำให้เกิดค่าความผิดพลาด $\varepsilon(t, \hat{\boldsymbol{\theta}}_N)$ เป็นไปตามสมการ

$$\varepsilon(t) = \varepsilon(t, \hat{\boldsymbol{\theta}}_N) = y(t) - \hat{y}(t \mid \hat{\boldsymbol{\theta}}_N) \tag{147}$$

เซตของค่าความผิดพลาด $\varepsilon(t, \hat{\boldsymbol{\theta}}_N)$ ที่เกิดขึ้นจะต้องไม่ Correlate กับเซตของสัญญาณอินพุตเราจะเรียก $\varepsilon(t, \hat{\boldsymbol{\theta}}_N)$ นี้ว่า Residual ในที่นี้คำว่า “Correlate” จะเขียนให้แปลว่าว่า “ขึ้นกับ” ดังนั้น Residual ก็คือการที่เซตของ $\varepsilon(t, \hat{\boldsymbol{\theta}}_N)$ ไม่ขึ้นกับเซตของ $u(t)$ นั่นเอง

คุณภาพของ Model Structure ส่วนหนึ่งสามารถดูได้จากค่า $\varepsilon(t, \hat{\boldsymbol{\theta}}_N)$ ที่เกิดขึ้นจาก Model Structure นั้น เราได้สมมติไว้แล้วในโครงสร้างของ Model Structure แต่ละ Model Structure นั้นมี $\varepsilon(t, \hat{\boldsymbol{\theta}}_N)$ คือค่าของสัญญาณรบกวนที่เกิดขึ้นซึ่งจะต้องเป็นสัญญาณรบกวนสีขาว $e(t)$ ดังนั้น Model Structure ที่เราหามาได้ก็ควรจะสร้าง $\varepsilon(t, \hat{\boldsymbol{\theta}}_N)$ ขึ้นมาให้มีลักษณะใกล้เคียงกับลักษณะของสัญญาณรบกวนสีขาวเช่นกัน เนื่องจากสัญญาณรบกวนสีขาวเกิดขึ้นแบบสุ่ม นั่นหมายความว่าการเกิดค่า $\varepsilon(t, \hat{\boldsymbol{\theta}}_N)$ นี้จะต้องไม่ขึ้นกับค่าใดๆเลย ในที่นี้คือสัญญาณอินพุต $u(t)$ เพราะถ้าเราสมมติว่าค่าของ $\varepsilon(t, \hat{\boldsymbol{\theta}}_N)$ ที่เกิดนี้ขึ้นกับค่าของ $u(t)$ จะทำให้การเปลี่ยนสัญญาณอินพุต $u(t)$ ก็อาจจะเป็นสาเหตุให้ค่าของ $\varepsilon(t, \hat{\boldsymbol{\theta}}_N)$ เปลี่ยนแปลงไปด้วย เช่นถ้าเราเปลี่ยนขนาดของ $u(t)$ ก็อาจจะทำให้ $\varepsilon(t, \hat{\boldsymbol{\theta}}_N)$ เปลี่ยนแปลงในทางที่เพิ่มขึ้น นั่นก็หมายความว่า Model Structure ที่เราหาได้มีข้อจำกัดกับขนาดสัญญาณอินพุตที่ใส่เข้าไป ดังนั้นเพื่อให้ได้ Model Structure ที่ดีเราจึงควรจะต้องทำการตรวจสอบ Residual ด้วย

ตามที่ได้กล่าวมา $\varepsilon(t, \hat{\boldsymbol{\theta}}_N)$ ไม่ควรจะขึ้นกับ $u(t)$ สมการที่ใช้บอกว่าสัญญาณ 2 สัญญาณดังกล่าวมีการ Correlate กันมากน้อยแค่ไหนคือ

$$\hat{R}_{\varepsilon u}^{N}(\tau) = \frac{1}{N}\sum_{t=1}^{N}\varepsilon(t)u(t-\tau) \tag{148}$$

ถ้าค่า $\hat{R}_{\varepsilon u}^{N}(\tau)$ น้อยแสดงว่ามีการ Correlate ระหว่าง $\varepsilon(t, \hat{\boldsymbol{\theta}}_N)$ และ $u(t)$ น้อยก็จะทำให้เราเชื่อมั่นได้ว่า Model Structure ยังคงมีประสิทธิภาพใกล้เคียงเดิมเมื่อเราใส่สัญญาณอินพุตอื่นๆเข้าไป นอกจากเราจะดูการ Correlate ระหว่าง $\varepsilon(t, \hat{\boldsymbol{\theta}}_N)$ และ $u(t)$ แล้ว เราก็น่าจะดูการ Correlate ระหว่าง $\varepsilon(t, \hat{\boldsymbol{\theta}}_N)$ และ $\varepsilon(t, \hat{\boldsymbol{\theta}}_N)$ ด้วยตามสมการดังนี้

$$\hat{R}_{\varepsilon}^{N}(\tau) = \frac{1}{N}\sum_{t=1}^{N}\varepsilon(t)\varepsilon(t-\tau) \tag{149}$$

ถ้า $\varepsilon(t, \hat{\boldsymbol{\theta}}_N)$ ที่เกิดขึ้นเป็นสัญญาณรบกวนสีขาวจริงค่า $\hat{R}_{\varepsilon}^{N}(\tau)$ ก็จะมีค่ามากเมื่อ $\tau = 0$ เท่านั้น