

วงจรถัดสัญญาณรบกวนความเร็วสูงแบบปรับตัวโดยใช้โครงสร้างแบบขนาน

High Speed Adaptive Noise Canceller with Parallel Block Structure



ปริญญาานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรอุตสาหกรรมศาสตรบัณฑิต

สาขาวิชาเทคโนโลยีโทรคมนาคม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2542

เลขหมู่.....

เลขทะเบียน.....37101

วัน, เดือน, ปี - 4 ก.ย. 2543

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์ วงจรขจัดสัญญาณรบกวนความเร็วสูงแบบปรับตัวโดยใช้โครงสร้างแบบขนาน

Project Report High Speed Adaptive Noise Canceller with Parallel Block Structure

ผู้จัดทำ 1. นายชาญวุฒิ สูดสงวน รหัส 41012051  
2. นายณฤตล ทัพพุน รหัส 41012053

อาจารย์ที่ปรึกษา รศ.ชวลิต เบญจางคประเสริฐ

ภาควิชา เทคนิคอุตสาหกรรม

คณะ วิศวกรรมศาสตร์

ปีการศึกษา 2542

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง อนุมัติ  
ให้นำปริญญานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรอุตสาหกรรมศาสตรบัณฑิต

คณะกรรมการสอบปริญญานิพนธ์

.....ประธานกรรมการ  
( )  
.....กรรมการ  
( )  
.....กรรมการ  
( )  
.....กรรมการ  
( )  
.....กรรมการ  
( )  
.....กรรมการ  
( )  
.....กรรมการ  
( )

ลิขสิทธิ์ของคณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## วงจรจัดตั้งสัญญาฉบับความรวดเร็วสูงแบบปรับตัวโดยใช้โครงสร้างแบบขนาน

โดย

นายชาญวุฒิ สุกสงวน

นายณฤศณ ทัพพุน

อาจารย์ที่ปรึกษา

รศ.ชวลิต เบญจางคประเสริฐ

บทคัดย่อ

ปริญญานิพนธ์นี้เป็นการนำเสนออะแดปทีฟอัลกอริทึมแบบใหม่ เพื่อใช้สำหรับตีเทค สัญญาณขายน้คลื่นเดียว ของความถี่ไม่ทราบค่าซึ่งมีสัญญาณแบบเกาส์เข้ามาปะปนด้วย โดยจะใช้ วงจรกรองความถี่แบบไอไออาร์แบนด์พาสฟิลเตอร์ ซึ่งความถี่กลางนั้นจะมีการเปลี่ยนแปลง การ ออกแบบวงจรกรองความถี่โครงสร้างแบบขนานเพื่อการประมวลผลที่รวดเร็วยิ่งขึ้น ซึ่งผลการ ทดลองจะชี้มูลเลขด้วยโปรแกรมคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# High Speed Adaptive Noise Canceller with Parallel Block Structure

By

Mr.Chanwut Sudsanguan

Mr.Naruedon tuppooon

Adviser

Assoc.Prof.Chawalit Benjangkprasert

## ABSTRACT

This paper present a new adaptive algorithm for detecting a single sinusoid of unknown frequency corrupted by Gaussian noise using IIR band pass filter with a variable center frequency. The filter has the parallel block structure for fast processing. The computer simulation results are given to demonstrate the convergence performance

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ

เรื่อง	หน้า
บทคัดย่อ	II
ABSTRACT	III
กิตติกรรมประกาศ	VII
บทนำ	VIII
บทที่ 1 ทฤษฎีของอะแดปทีฟฟิลเตอร์เบื้องต้น	
1.1 คิจิตอลฟิลเตอร์	1
1.2 Application In Digital Filter Design	2
1.3 Realizatation Structures for Digital Filter	2
1.4 ชนิดของคิจิตอลฟิลเตอร์	4
1.5 Summary of key Characteristic Feature of FIR Filter	6
1.6 Summary of The Basic Features of IIR Filter	7
1.7 อะแดปทีฟคิจิตอลฟิลเตอร์	8
1.8 Concepts of Adaptive Digital Filter	9
1.9 ตัวอย่างการใช้งานอะแดปทีฟฟิลเตอร์ในงานต่าง ๆ	10
บทที่ 2 การวิเคราะห์และออกแบบ	
2.1 อะแดปทีฟอัลกอริทึมแบบสเกลาร์	25
2.2 อะแดปทีฟอัลกอริทึมโครงสร้างแบบขนาน	27
2.3 การออกแบบอะแดปทีฟอัลกอริทึมแบบขนาน กรณี $L = 1$	31
2.4 การออกแบบอะแดปทีฟอัลกอริทึมแบบขนาน กรณี $L = 2$	33
2.5 การออกแบบอะแดปทีฟอัลกอริทึมแบบขนาน กรณี $L = 3$	38
2.6 ปัญหาที่เกิดขึ้นของอะแดปทีฟอัลกอริทึมแบบขนาน	44
2.7 อะแดปทีฟอัลกอริทึมโครงสร้างใหม่	45
บทที่ 3 โปรแกรมและผลการทดลอง	
3.1 โปรแกรม	56
3.2 ผลการทดลอง	78
บทที่ 4 สรุปผลการทดลอง	99
บรรณานุกรม	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญโปรแกรม

โปรแกรม	หน้า
- โปรแกรมแสดงคุณสมบัติของ $ H(\omega) $ เทียบกับ $\omega$	56
- โปรแกรมแสดงคุณสมบัติของ $ H(\omega) $ เทียบกับ $\alpha_1(k)$	57
- โปรแกรมแสดงคุณสมบัติของ $\alpha_1(k)$ เทียบกับ $k$ เมื่อ $L=1$	58
- โปรแกรมแสดงคุณสมบัติของ $\alpha_1(k)$ เทียบกับ $k$ เมื่อ $L=1$ กรณีมีการเปลี่ยนแปลงตัวแปร	59
- โปรแกรมแสดงคุณสมบัติของ $\alpha_1(k)$ เทียบกับ $k$ เมื่อ $L=3, l=0$	60
- โปรแกรมแสดงคุณสมบัติของ $\alpha_1(k)$ เทียบกับ $k$ เมื่อ $L=3, l=1$	62
- โปรแกรมแสดงคุณสมบัติของ $\alpha_1(k)$ เทียบกับ $k$ เมื่อ $L=3, l=2$	64
- โปรแกรมแสดงคุณสมบัติการเปลี่ยนแปลงของ $\alpha_1(k)$ กรณี $L=1$	66
- โปรแกรมแสดงคุณสมบัติการเปลี่ยนแปลงของ $\alpha_1(k)$ กรณี $L=3, l=0$	67
- โปรแกรมแสดงคุณสมบัติการเปลี่ยนแปลงของ $\alpha_1(k)$ กรณี $L=3, l=1$	69
- โปรแกรมแสดงคุณสมบัติการเปลี่ยนแปลงของ $\alpha_1(k)$ กรณี $L=3, l=2$	71
- โปรแกรมแสดงคุณสมบัติการเปลี่ยนแปลงของ $\alpha_1(k)$ กรณี $L=3$ โครงสร้างใหม่	73
- โปรแกรมแสดงคุณสมบัติ $\alpha_1(k)$ เทียบกับ $k$ กรณี $L=2$	75
- โปรแกรมแสดงคุณสมบัติของ $\alpha_1(k)$ เทียบกับ $k$ โดยใช้ New Algorithm	76

## สารบัญกราฟ

### กราฟผลการทดลอง

- กราฟแสดงคุณสมบัติของ $ H(\omega) $ เทียบกับ $\omega$	78
- กราฟแสดงคุณสมบัติของ $ H(\omega) $ เทียบกับ $\alpha_1(k)$	82
- กราฟแสดงคุณสมบัติของ $\alpha_1(k)$ เทียบกับ $k$ ในกรณีที่ $L=1$	86
- กราฟแสดงคุณสมบัติของ $\alpha_1(k)$ เทียบกับ $k$ กรณี โครงสร้างแบบขนานที่ $L=3$	89
- กราฟแสดงคุณสมบัติการเปลี่ยนแปลงของ $\alpha_1(k)$ กรณี $L=1$	90
- กราฟแสดงคุณสมบัติการเปลี่ยนแปลงของ $\alpha_1(k)$ กรณี $L=3, l=0,1$ และ $3$	90
- กราฟแสดงคุณสมบัติค่าการเปลี่ยนแปลงของ $\alpha_1(k)$ กรณีโครงสร้างใหม่	93
- กราฟเปรียบเทียบการเปลี่ยนแปลงของ $\alpha_1(k)$ ในกรณีที่ $L=1$ และ $L=3$ กรณีโครงสร้างใหม่	93
- กราฟแสดงคุณสมบัติของ $\alpha_1(k)$ เทียบกับ $k$ กรณีโครงสร้างใหม่ $L=3$	94
- กราฟแสดงคุณสมบัติของการเปลี่ยนแปลง $\alpha_1(k)$ เปรียบเทียบระหว่างที่ $L=1$ และ $L=3$	96
- กราฟแสดงคุณสมบัติของการเปลี่ยนแปลง $\alpha_1(k)$ เปรียบเทียบระหว่างที่ $L=1, L=2$ และ $L=3$	97
- กราฟแสดงคุณสมบัติของ $\alpha_1(k)$	98

## กิตติกรรมประกาศ

การที่โครงการ High speed adaptive noise canceller สำเร็จลงด้วยความเรียบร้อยเป็นอย่างดีนี้ ทางคณะผู้จัดทำขอขอบพระคุณอย่างต่อท่านทั้งหลายที่กรุณาให้ความอนุเคราะห์ความร่วมมือให้คำแนะนำเป็นที่ปรึกษา และอำนวยความสะดวกโดยเฉพาะ รศ.ชวลิต เบญจางคประเสริฐ ในการให้คำปรึกษาชี้แนะ และแก้ไขปัญหาต่างๆ ในการจัดทำโครงการ และในการเขียนปริญญาบัตร ผศ.อรรถสิทธิ์ หล้าสกุล และ อ.พิชญ์ สุพรรณกุล ในการให้คำปรึกษาแนะนำในการเขียนโปรแกรม



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทนำ

ในเครื่องมือที่มีการใช้ความถี่ของสัญญาณเพื่อนำไปประมวลผล เช่นเครื่องตรวจสอบการทำงานของหัวใจ หรือเครื่องมือวัดความถี่ต่างๆ ความสามารถของวงจรกรองความถี่ในวงจร ของเครื่องมือเหล่านี้มีความสำคัญอย่างยิ่ง เพราะถ้าวงจรกรองความถี่ที่ใช้ในวงจรไม่ดี จะทำให้ผลการวัด หรือตรวจสอบไม่ถูกต้อง

ปริญญานิพนธ์นี้จึงสร้างวงจรกรองความถี่ที่สามารถลดทอนสัญญาณรบกวนได้ดี โดยการออกแบบ วงจรอะแคทีฟอัลกอริทึม ประยุกต์เข้ากับวงจรกรองความถี่แบบ ไอไออาร์เบนค์พาสฟิลเตอร์ ผลการออกแบบนี้จะได้รับการพิสูจน์จากการซิมูเลตด้วยโปรแกรมคอมพิวเตอร์



# บทที่ 1

## ทฤษฎีของอะแดปทีฟเบื้องต้น

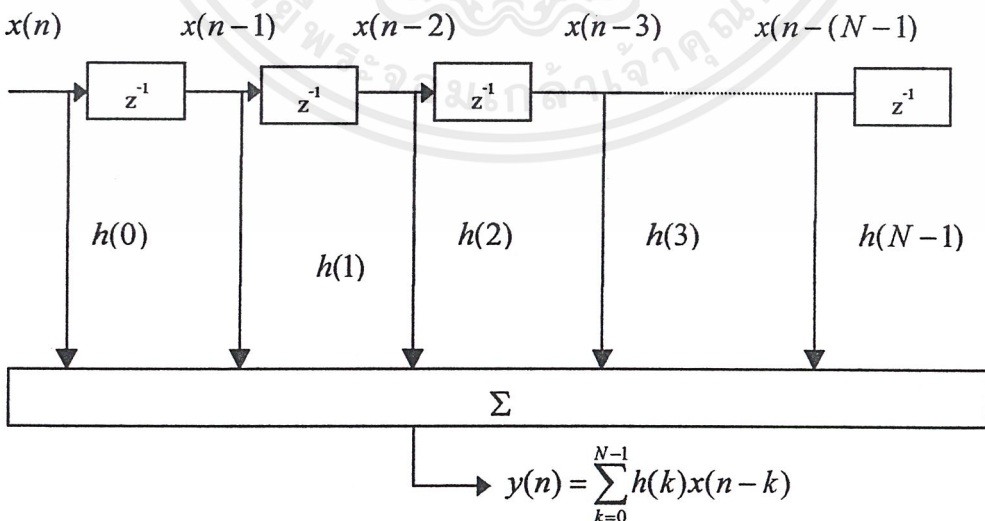
### 1.1 ดิจิตอลฟิลเตอร์

ดิจิตอลฟิลเตอร์ สิ่งสำคัญมากที่สุดอย่างหนึ่งในการนำไปใช้ในระบบ DSP (Digital Signal Processing) เป็นการนำกระบวนการแก้ปัญหาทางคณิตศาสตร์มาใช้ในระบบฮาร์ดแวร์ หรือซอฟต์แวร์ กล่าวคือ สัญญาณดิจิตอลอินพุต จะทำการสร้างสัญญาณดิจิตอลเอาต์พุต โดยการใช้ Filtering Algorithm เพื่อให้ได้รับสัญญาณจากการ Filtering ตามความต้องการเพื่อนำไปใช้ใน ดิจิตอลฟิลเตอร์สำหรับชนิดของ ฟิลเตอร์ ที่สำคัญจะถูกกำหนดโดย

$$y(n) = \sum_{k=0}^{N-1} h(k) * x(n-k) \quad 1.1$$

โดยที่  $h(k)$  คือ สัมประสิทธิ์ ของฟิลเตอร์  $k = 0, 1, \dots, n-1$   
 $x(n)$  คือ อินพุตฟิลเตอร์  
 $y(n)$  คือ เอาท์พุตฟิลเตอร์

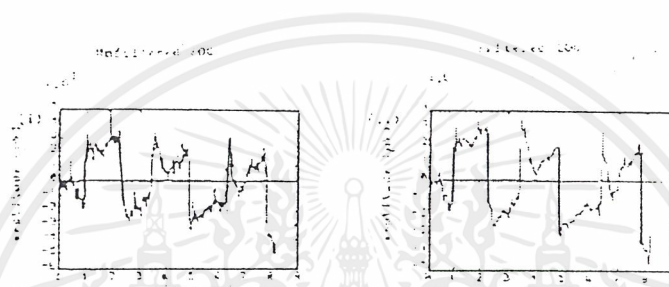
สำหรับฟิลเตอร์ ที่ให้มาค่าสัมประสิทธิ์จะกำหนด Characteristic ของฟิลเตอร์ และ Filtering นี้จะอยู่กับการ Convolution ที่แท้จริงของสัญญาณและการตอบสนองต่ออิมพัลส์ของฟิลเตอร์ในไทม์โดเมน(Time Domain)ซึ่งก็คือ  $h(k)$



รูปที่ 1 บล็อก ไดอะแกรมของ Transversal Filter โดยมี  $h(k)$ , ( $k=0, 1, \dots, N-1$ ) เป็นสัมประสิทธิ์ ฟิลเตอร์และ  $z^{-1}$  แสดงถึงดีเลย์ของช่วงเวลาในการสุ่มตัวอย่างหนึ่งครั้ง

จากรูปจะแสดงถึง บล็อกไดอะแกรมซึ่งแสดงถึงนิยามของ ดิจิตอลฟิลเตอร์ที่กำหนดไว้ก่อนหน้านี โดยที่รูปแบบฟิลเตอร์ ชนิดนี้เป็นที่รู้จักอย่างแพร่หลาย เรียกว่า Transversal Filter จากรูปจะแสดงถึงคีย์ ของช่วงเวลาในการสุ่มตัวอย่างหนึ่งครั้ง

วัตถุประสงค์ของฟิลเตอร์ โดยรวมๆ แล้วเป็นการกำจัดหรือลดสัญญาณรบกวน ในระบบอื่น ๆ อาจจะมีประปนมาด้วยดังนั้นเราจึงทำการกรองสัญญาณที่แท้จริงออกมาเพื่อที่จะสามารถนำไปใช้ในการวิเคราะห์ได้ถูกต้องดังรูป



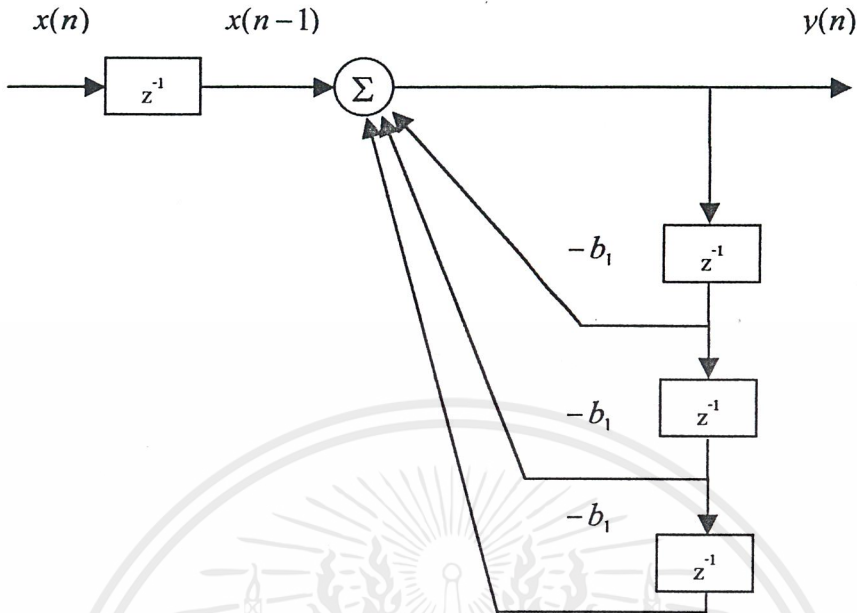
รูปที่ 2 แสดงถึงผลของการใช้ดิจิตอล Low Pass Filtering ในการลดสัญญาณรบกวนความถี่สูงในทางการแพทย์

## 1.2 Application in Digital Filter Design

Z-Transform เป็นสิ่งที่สำคัญที่สุดอย่างหนึ่งในระบบ DSP นั่นคือใช้การออกแบบเพื่อวิเคราะห์สัญญาณ Error ใน ดิจิตอลฟิลเตอร์โดยเฉพาะในระบบ IIR ฟิลเตอร์ซึ่งมักจะถูกใช้ในการคำนวณค่าสัมประสิทธิ์ของ ดิจิตอลฟิลเตอร์และวิเคราะห์ผลของการ Quantization Errors ต่าง ๆ ในโครงสร้างที่ใช้ใน ดิจิตอลฟิลเตอร์และสิ่งที่สำคัญอย่างหนึ่งที่ใช้ Z-Transform ในการออกแบบ Discrete Filter คือใช้ในการแสดงโครงสร้างของ ดิจิตอลฟิลเตอร์

## 1.3 Realization Structures for Digital Filter

Discrete Time Filter จะถูกนำมาแสดงบ่อย ๆ ในรูปแบบของ บล็อกไดอะแกรม หรือ Signal Flow Graph ซึ่ง บล็อกไดอะแกรม จะเป็นการสะดวกกว่าที่ใช้ในการแสดง Differential Equations หรือ Equivalent ของทรานสเฟอร์ฟังก์ชัน ซึ่งสามารถพิจารณาดังตัวอย่าง Discrete-Filter ด้วย Difference Equation

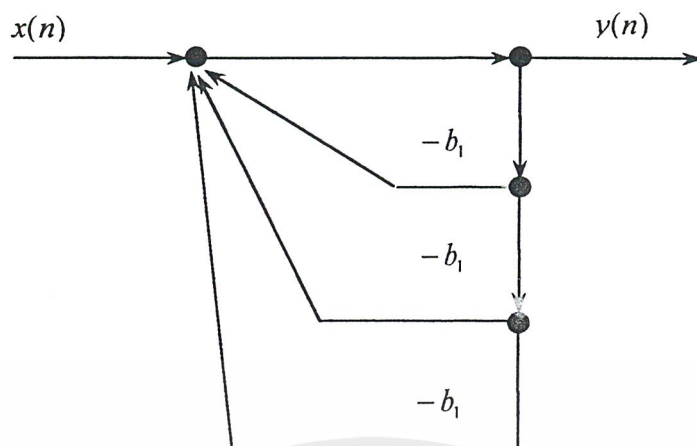


รูปที่ 3 การวิเคราะห์สมการ Difference Equation ในรูปแบบ บล็อกไคอะแกรม โดยสมมติสมการดังนี้

$$y(n) = x(n-1) - b_1 y(n-1) + b_2 y(n-2) + b_3 y(n-3)$$

1.2

บล็อกไคอะแกรม แสดงถึงสมการดังกล่าวจากรูปสัญลักษณ์ จะแสดงถึงคิเล็ย ของช่วง เวลาหนึ่งหน่วย ซึ่งอาจจะพิจารณาในสาขา (Node) ต่าง ๆ ถูกสรแสดงถึงการ Multipliers และ ค่าคงที่ตัวถูกทำมาจากการคูณกับตัวประกอบ ความสัมพันธ์ระหว่าง Difference Equation และ บล็อกไคอะแกรมจะค่อย ๆ ปรากฏขึ้น และ Signal Flow Graph Diagram ก็ให้แสดงกับ Difference Equation เช่นเดียวกันดังรูปที่ 4



รูปที่ 4 การวิเคราะห์สมการ Difference Equation ในรูปแบบ Signal Flow Graph Diagram โดยสมมติสมการดังนี้

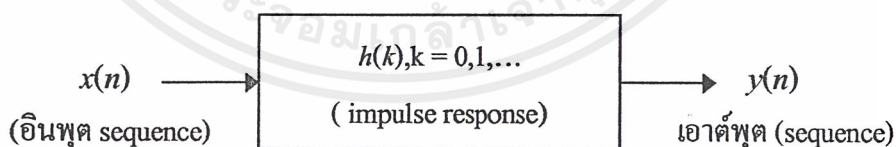
$$y(n) = x(n-1) - b_1 y(n-1) + b_2 y(n-2) + b_3 y(n-3)$$

1.3

#### 1.4 ชนิดคิจิตอลฟิลเตอร์

คิจิตอลฟิลเตอร์ ที่ใช้อย่างกว้างขวางแบ่งออกเป็น 2 ชนิด

1. Finite Impulse Response (FIR)
2. Infinite Impulse Response (IIR)



รูปที่ 5 บล็อกไดอะแกรม แสดงถึงหลักการของคิจิตอลฟิลเตอร์

ฟิลเตอร์แต่ละชนิด จะมีรูปแบบพื้นฐานแต่ละตัวซึ่งสามารถแสดงลำดับการตอบสนองอิมพัลส์  $h(k), (k = 0, 1, 2, \dots)$  ดังรูปที่ 5 สัญญาณ อินพุต และ เอาต์พุต ที่ต่อฟิลเตอร์จะถูกนำมาเกี่ยวใช้โดยผลรวมของการ Convolution ซึ่งสมการแต่ละชนิดแสดงได้ดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$y(n) = \sum_{k=0}^{\infty} h(k) * x(n-k) : \text{IIR}$$

$$y(n) = \sum_{k=0}^{N-1} h(k) * x(n-k) : \text{FIR} \quad 1.4$$

จากสมการข้างต้นสำหรับ IIR ฟิเตอร์ นั้นผลการตอบสนองต่ออิมพัลส์จะอยู่ในช่วงที่ไม่จำกัด แต่สำหรับ FIR ฟิเตอร์จะอยู่ในช่วงที่กำหนด โดยเริ่มที่  $a_k$  สำหรับ FIR จะมีเพียง  $N$  ค่าส่วนในการคำนวณหาค่า เอาต์พุต ของ IIR ไม่สามารถเป็นไปได้ เนื่องจากสมการมีความยาวของผลการตอบสนองต่ออิมพัลส์นั้นยาวมาก (Infinity) ดังนั้นเราจึงแทนสมการของ IIR ฟิเตอร์ ในรูปของ Recursive Form

$$y(n) = \sum_{k=0}^{\infty} h(k) * x(n-k) = \sum_{k=0}^{\infty} a_k * x(n-k) - \sum_{k=0}^{\infty} b_k * x(n-k) \quad 1.5$$

โดยที่  $a_k$  และ  $b_k$  เป็นค่าสัมประสิทธิ์ของฟิเตอร์ ดังนั้นจากสมการข้างต้นคือสมการที่แตกต่างกันระหว่าง FIR ฟิเตอร์ และ IIR ฟิเตอร์จากสมการเหล่านั้นจะมีค่า  $h(k)$  สำหรับ FIR ฟิเตอร์ และ  $a_k$  และ  $b_k$  สำหรับ IIR ฟิเตอร์ซึ่งสมการทั้งสองเป็นสิ่งที่สำคัญมากที่ใช้ในการออกแบบ Filter เป็นส่วนใหญ่ จากฟังก์ชันการสุ่มตัวอย่าง เอาต์พุต  $y(n)$  เป็นฟังก์ชันของเอาต์พุต ที่ผ่านมาซึ่งค่าที่ถูกต้องจะเท่ากับตัวอย่างอินพุต ที่ส่งมา ทั้งในอดีตและปัจจุบัน ซึ่งนั่นคือ IIR ฟิเตอร์ จะมีสมการ IIR ฟิเตอร์ ก็คือ สมการ FIR ฟิเตอร์นั่นเองและจากสมการด้านล่าง นี้ก็คือ ทราสเฟอร์ฟังก์ชันของ FIR และ IIR ฟิเตอร์ซึ่งค่าทราสเฟอร์ฟังก์ชัน เหล่านี้จะมีประโยชน์มาก ในการคำนวณหาค่าใน ผลตอบสนองทางความถี่

$$H(z) = \sum_{k=0}^{N-1} h(k) * z^{-k} : \text{FIR}$$

$$H(z) = \frac{\sum_{k=0}^N a_k * z^{-k}}{1 + \sum_{k=0}^{N-1} b_k * z^{-k}} : \text{IIR}$$

## 1.5 Summary of key Characteristic Features of FIR Filter

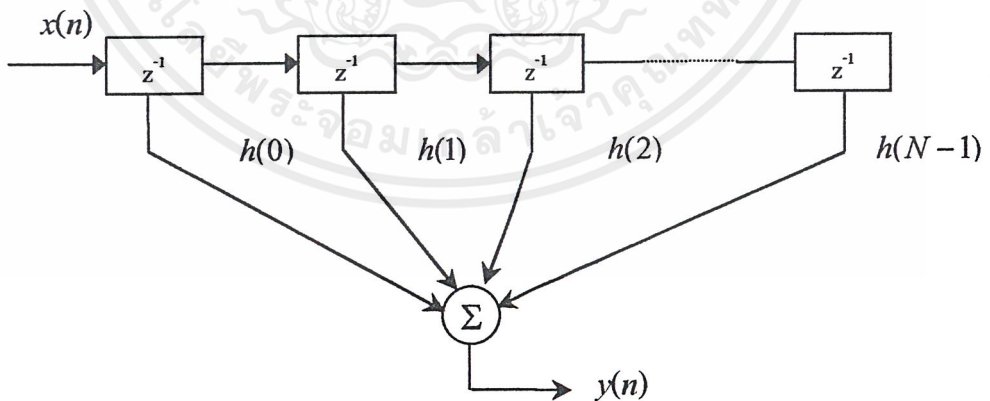
1.Characteristic ของ FIR ฟิลเตอร์จะแสดงดังสมการ

$$y(n) = \sum_{k=0}^{\infty} h(k) * x(n-k)$$

$$H(z) = \sum_{k=0}^{N-1} h(k) * z^{-k}$$

โดยที่  $h(k), k = 0, 1, \dots, n-1$  เป็นค่าสัมประสิทธิ์ในการตอบสนองต่ออิมพัลส์ของฟิลเตอร์  
 $H(z)$  เป็นทราสเฟอร์ฟังก์ชันของฟิลเตอร์  
 $N$  เป็นจำนวนวงจรฟิลเตอร์

นั่นก็คือจำนวนของสัมประสิทธิ์ของฟิลเตอร์ของสมการแรกเป็น FIR Difference Equation ซึ่งมันจะเป็นสมการในไทม์โดเมนและจะพิจารณา FIR ฟิลเตอร์ในรูปของ Non-Recursive Form ซึ่งจะทำให้การคำนวณอย่าง เอาต์พุต  $y(n)$  เป็นฟังก์ชันเพียงค่าของ อินพุตในอดีตและปัจจุบันเท่านั้น  $x(n)$  เมื่อฟิลเตอร์ ถูกนำไปใช้ในรูปแบบนี้ นั่นคือ จะเป็นการหาค่าโดยตรง จะทำให้ได้ค่าคงที่ และ ทราสเฟอร์ฟังก์ชันของฟิลเตอร์จะใช้ในการวิเคราะห์และหาค่าฟิลเตอร์ ในทาง Frequency Response



รูปที่ 6 โครงสร้างของ FIR ดิจิตอลฟิลเตอร์

2. FIR ฟิลเตอร์สามารถมีความเป็น Linear Phase Response ที่แท้จริง
3. FIR ฟิลเตอร์ง่ายที่จะนำไปใช้ในระบบ DSP Processors ทั้งหมดมีความเหมาะสมที่จะใช้กับ FIR Filtering Non-Recursive FIR Filter จะมีข้อผิดพลาดน้อยกว่าผลของความยาวที่จำกัดกว่า IIR ฟิลเตอร์

## 1.6 Summary of The Basic Features of IIR Filter

Characteristic ของ IIR ดิจิตอลฟิลเตอร์ จะถูกกำหนดตาม Recursive Equation ดังนี้

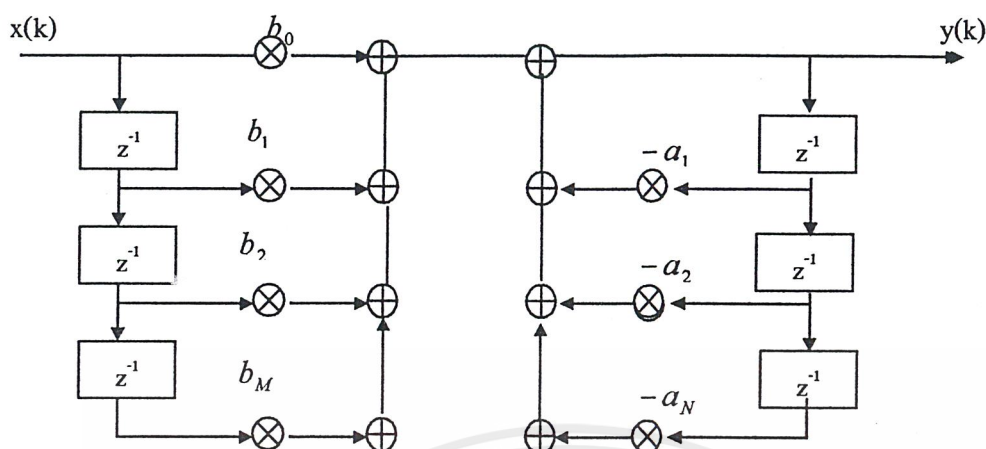
$$y(n) = \sum_{k=0}^{\infty} h(k) * x(n-k) = \sum_{k=0}^{\infty} a_k * x(n-k) - \sum_{k=0}^{\infty} b_k * x(n-k)$$

โดยที่  $h(k)$  เป็นผลการตอบสนองต่ออิมพัลส์ของฟิลเตอร์ ซึ่งจะมีช่วงเวลาที่ไม่มีจำกัดเป็นสัมประสิทธิ์ของฟิลเตอร์  
 $x(n)$  และ  $y(n)$  เป็นอินพุต และเอาต์พุต ของฟิลเตอร์

ทราสเฟอร์ฟังก์ชันสำหรับ IIR ฟิลเตอร์ถูกกำหนดโดย

$$H(z) = \frac{a_0 + a_1 z^{-1} + \dots + a_n z^{-N}}{1 + b_1 z^{-1} + \dots + b_m z^{-M}} = \frac{\sum_{k=0}^N a_k * z^{-k}}{1 + \sum_{k=1}^M b_k * z^{-k}} \quad 1.8$$

ส่วนที่สำคัญในการออกแบบ IIR ฟิลเตอร์คือการค้นหาค่าที่เหมาะสม สำหรับค่าสัมประสิทธิ์และให้สอดคล้องกับลักษณะของฟิลเตอร์ใน Frequency Response ดังนั้นสมการดังกล่าวทั้งสอง จึงเป็น Characteristic Equation สำหรับ IIR ฟิลเตอร์



รูปที่ 7 โครงสร้าง IIR ดิจิตอลฟิลเตอร์

จากสมการที่ 1.5 กล่าวได้ว่าค่าการสุ่มตัวอย่าง เอาต์พุต  $y(n)$  เป็นฟังก์ชันของ เอาต์พุตที่ผ่านมา  $y(n)$  ซึ่งจะเท่ากับค่าสมการสุ่มตัวอย่างอินพุต ทั้งในปัจจุบันและอดีต นั่นคือ IIR ฟิลเตอร์ เป็นระบบป้อนกลับ ซึ่งปกติแล้ว IIR ฟิลเตอร์จะถูกใช้เมื่อจุดคัทออฟ ที่แคบและมี Throughput ที่สูงเป็นความต้องการที่สำคัญ เหตุผลที่จะทำให้ IIR ฟิลเตอร์ไม่เสถียร

## 1.7 อะแดปทีฟดิจิตอลฟิลเตอร์

อะแดปทีฟดิจิตอลฟิลเตอร์ เป็นสิ่งที่จำเป็นในการกรองความถี่แบบดิจิตอล ที่มีลักษณะการปรับค่าต่างๆ ได้ด้วยตัวเอง โดยทำการปรับค่าอย่างอัตโนมัติ เพื่อที่จะเป็นค่าของสัญญาณอินพุตในครั้งต่อไป

อะแดปทีฟดิจิตอลฟิลเตอร์ หมายถึงอุปกรณ์ซึ่งถูกออกแบบให้สามารถปรับค่าเองได้ในความคิดที่ว่า อุปกรณ์จะบรรจุกลุ่มของพารามิเตอร์ที่ปรับค่าได้ และค่ามันจะถูกกำหนดโดยอัตโนมัติขึ้นอยู่กับการประมาณค่าคุณสมบัติทางสถิติของสัญญาณที่เกี่ยวข้อง ฉะนั้นทฤษฎีของอะแดปทีฟฟิลเตอร์จะมีความสัมพันธ์ใกล้ชิดกับการออกแบบฟิลเตอร์ที่ดีที่สุด ในปัญหาการออกแบบต้องการที่จะจัดกลุ่มที่ดีที่สุดของฟิลเตอร์พารามิเตอร์ จากความรู้ของคุณสมบัติของสัญญาณที่เกี่ยวข้อง ให้เหมาะสมกับบรรทัดฐาน และ ปัญหาของ อะแดปทีฟฟิลเตอร์ คือ ความต้องการที่จะหา อัลกอริทึม สำหรับปรับค่าฟิลเตอร์พารามิเตอร์ในสถานการณ์ที่ความรู้เกี่ยวกับคุณสมบัติของสัญญาณที่เกี่ยวข้องไม่สมบูรณ์ ดังนั้นการทำงานของ อะแดปทีฟฟิลเตอร์จะเข้าสู่ลักษณะฟิลเตอร์ที่ดีที่สุดหลังจากผ่านขบวนการซ้ำๆ เกี่ยวกับตัวเลขจำนวนมากของอัลกอริทึม

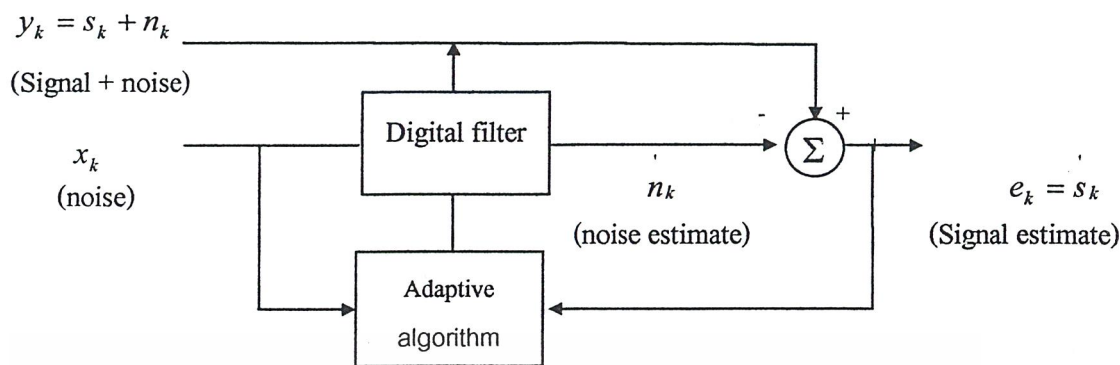
## อะแดปทีฟฟิลเตอร์ มีลักษณะแตกต่างกันดังนี้

1. อะแดปทีฟฟิลเตอร์ลักษณะ Open Loop หรือไม่กระทำซ้ำ ประกอบด้วยขบวนการสองขั้น โดยที่ขบวนการแรก คือ การเรียนรู้สถิติของสัญญาณที่เกี่ยวข้อง แล้วนำผลที่ได้ใส่เข้าไปใน อัลกอริทึมที่มีการกระทำซ้ำ (Non-Recursive Algorithm) ลักษณะเช่นนี้จะมีข้อเสียที่ต้องการความละเอียดมากและฮาร์ดแวร์ราคาแพง
2. อะแดปทีฟฟิลเตอร์ลักษณะ Closed Loop หรือ กระทำซ้ำ ที่เรื่องของสถิติไม่ได้ถูกและมากอย่างเด่นชัด แต่ว่าการออกแบบของ อะแดปทีฟฟิลเตอร์ จะทำได้ในขบวนการเดียวโดยการเฉลี่ยของอัลกอริทึมที่กระทำซ้ำ (Recursive Algorithm) ซึ่งจะเปลี่ยนค่าฟิลเตอร์พารามิเตอร์ โดยอัตโนมัติ จะค่อยๆ เรียนรู้เกี่ยวกับสถิติของสัญญาณที่เกี่ยวข้องและการปรับปรุงกลุ่มของค่าปัจจุบันของฟิลเตอร์ ที่เปลี่ยนค่าพารามิเตอร์ได้ จะถูกคำนวณโดยใช้ข่าวสารใหม่และ Adjust Table Filter (Filter ที่เปลี่ยนค่า Parameters) ค่าพารามิเตอร์ของมันจะเพิ่มขึ้นและการกระทำซ้ำครั้งต่อไปจะมีฐานจากการปฏิบัติการของฟิลเตอร์ ด้วยกลุ่มของค่าที่ปรับปรุงแล้วในลักษณะของ Closed Loop การเรียนรู้และการปฏิบัติการคำนวณจะรวมกันเป็นขบวนการเดียวกันซึ่งดูเหมือนว่าจะมีเครื่องมือที่ง่ายกว่าลักษณะ Open Loop

อะแดปทีฟดิจิทัลฟิลเตอร์ มีคุณสมบัติใน Frequency Response จะทำการปรับ หรือแก้ไข โดยอัตโนมัติ เพื่อที่จะปรับปรุง โครงสร้างให้สอดคล้องกับเกณฑ์บางส่วนในการที่จะทำให้ฟิลเตอร์ ทำการปรับและเปลี่ยนค่าลักษณะสัญญาณอินพุต เนื่องจากการปรับค่าของตัวเอง และมีความง่ายในการสร้าง อะแดปทีฟดิจิทัลฟิลเตอร์จะพบว่าถูกนำมาใช้ในการหลาย ๆ อย่าง เช่น Telephone Echo Canceling, Radar Signal Processing, Navigator System Equation of Communication Channels และ Biomedical Signal Enhancement

### 1.8 Concepts of Adaptive Digital Filter

อะแดปทีฟฟิลเตอร์ as a Noise Canceller ประกอบด้วยสองส่วนที่แตกต่างกัน ดิจิตอลฟิลเตอร์ กับสัมประสิทธิ์ที่ใช้ในการปรับค่า และ อะแดปทีฟอัลกอริทึมซึ่งจะใช้ในการปรับค่าหรือปรับปรุงค่าสัมประสิทธิ์ของฟิลเตอร์ ดังรูป



รูปที่ 8 Adaptive Noise Canceller

สัญญาณ อินพุต 2 สัญญาณ  $y_k$  และ  $x_k$  ถูกป้อนอย่างต่อเนื่องไปยัง อะแดปทีฟฟิลเตอร์ สัญญาณ  $y_k$  ประกอบด้วยสัญญาณที่ต้องการ  $s_k$  และน็อยส์ปะปนอยู่ สมมติว่าสัญญาณทั้งสองเกี่ยวข้องซึ่งกันและกัน โดยที่  $x_k$  เป็นสัญญาณที่ใช้ในการตรวจสอบสัญญาณที่ปะปนมา ซึ่งจะเกี่ยวพันกับน็อยส์ ซึ่งให้เป็น  $n'_k$  ของ  $n_k$  ดังนั้น ค่าประมาณของสัญญาณที่ต้องการจะหาได้จากการหักล้างของเอาต์พุต ของดิจิตอลฟิลเตอร์ จากสัญญาณ  $y_k$

$$s'_k = y_k - n'_k = s_k + n_k - n'_k \quad 1.9$$

วัตถุประสงค์ที่สำคัญใน Noise Canceling ก็เพื่อจะสร้างค่าประมาณที่ดีที่สุดของน็อยส์ใน  $s'_k$  สัญญาณที่ปะปนกัน และค่าประมาณที่ดีที่สุดของสัญญาณซึ่งจะทำได้โดยใช้  $s'_k$  ในระบบ การป้อนกลับเพื่อทำการปรับค่าสัมประสิทธิ์ของดิจิตอลฟิลเตอร์ และ อะแดปทีฟอัลกอริทึมที่เหมาะสมเพื่อลดน็อยส์ให้มากที่สุด ใน  $s'_k$  สัญญาณ เอาต์พุต  $s'_k$  จะนำไปใช้สองส่วนคือ

1. ใช้ในการประมาณค่าสัญญาณที่ต้องการ
2. สัญญาณที่ผิดพลาดจะถูกใช้ในการปรับค่าสัมประสิทธิ์ของฟิลเตอร์

## 1.9 ตัวอย่างการใช้งาน อะแดปทีฟฟิลเตอร์ ในงานต่าง ๆ

### 1. Modeling of an unknown Dynamic System

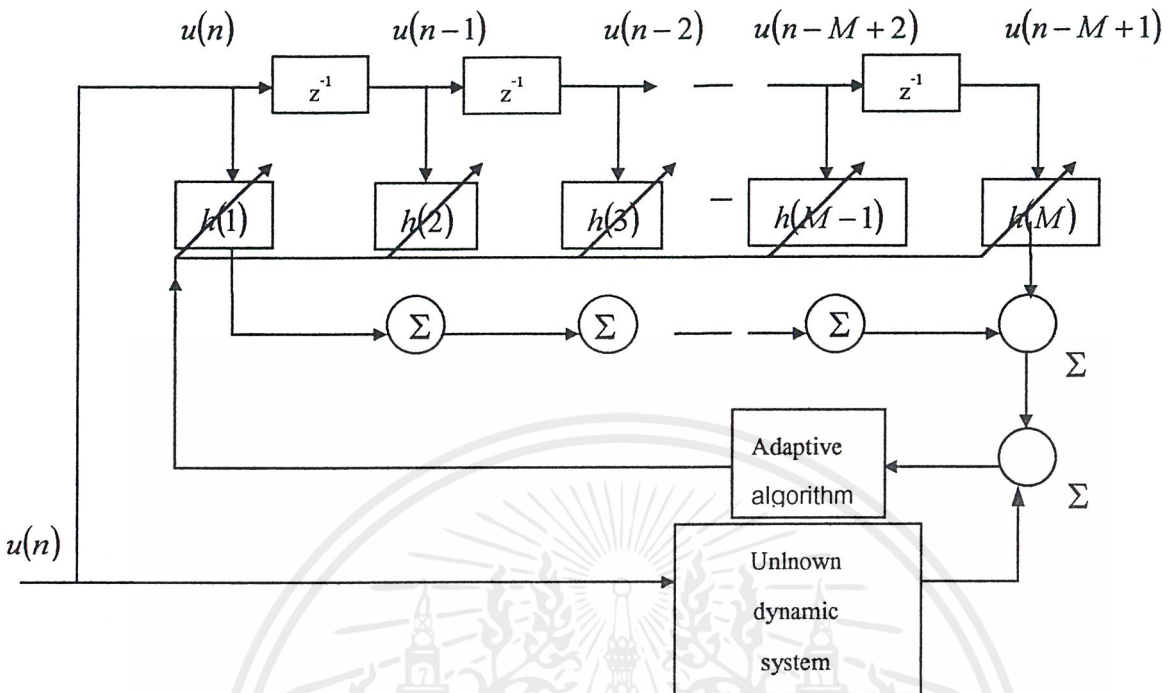
สมมติว่ามีระบบการเปลี่ยนแปลงที่ไม่เป็นที่รู้จัก และมีกลุ่มของเครื่องมือวัด Discrete Time ใช้พิจารณาความหลากหลายของสัญญาณเอาต์พุต ของระบบที่ตอบสนองกับสัญญาณคงที่ป้อนให้กับ อินพุตของระบบ สมมติว่าระบบนี้ไม่เปลี่ยนแปลงไปตามเวลาและมีคุณสมบัติเชิงเส้น และต้องการที่จะพัฒนา Model (แบบจำลอง) สำหรับระบบนี้ในรูปแบบของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Tapped-Delay-Line Filter ซึ่งประกอบด้วยกลุ่มของอุปกรณ์ Delay-Line แต่ละอันจะแสดงด้วยตัวปฏิบัติการ Unit-Delay กลุ่มของค่าสัมประสิทธิ์ที่ปรับค่าได้ที่มีการต่อภายในดังรูปที่ 9 ที่เวลา  $n$  สัญญาณที่มีอยู่ประกอบด้วยกลุ่มของ  $u(n), u(n-1), \dots, u(n-M+1)$  ตัวอย่างเหล่านี้ถูกคูณด้วยกลุ่มที่ตรงกันของค่าสัมประสิทธิ์ที่ปรับค่าได้ มีชื่อว่า  $h(1), h(2), \dots, h(M)$  เพื่อที่จะสร้างสัญญาณเอาต์พุต  $y(n)$  ให้ เอาต์พุต แท้จริงของระบบที่ไม่เป็นที่รู้จักมีสัญลักษณ์  $d(n)$  เอาต์พุต ของ อะแดปทีฟฟิลเตอร์ ถูกเปรียบเทียบกับ เอาต์พุต ของระบบที่ไม่เป็นที่รู้จัก  $d(n)$  เพื่อสร้างสัญญาณผิดพลาด ซึ่งเป็นความแตกต่างระหว่างทั้งสอง

ถึงตอนนี้อาจจะบอกได้ว่าสถานะการทำงานของ อะแดปทีฟฟิลเตอร์ ได้คั้งนี้ ด้วยเอาต์พุตของระบบที่ไม่เป็นที่รู้จัก  $d(n)$  จะเรียกว่าการตอบสนองที่ได้ทำการออกแบบ (desired response) ต้องการที่จะพัฒนาขบวนการ อะแดปทีฟสำหรับการปรับค่าสัมประสิทธิ์ของ Tapped-Delay-Line Filter เพื่อที่จะทำให้สัญญาณผิดพลาด  $e(n)$  มีน้อยที่สุดในลักษณะนี้ บรรทัดฐานซึ่งถูกใช้บ่อย ๆ ในทางปฏิบัติสำหรับการทำให้น้อยที่สุดคือบรรทัดฐานค่าผิดพลาดกำลังสองเฉลี่ย ซึ่งเหมาะสมกับกรณีเมื่อสัญญาณ อินพุตและเอาต์พุต ของระบบที่ไม่เป็นที่รู้จักนั้นคงที่ ซึ่งเป็นผลมาจากการสมมติให้ระบบไม่เปลี่ยนแปลงไปตามเวลา หรือ Time-Invariant จะทำให้สัญญาณผิดพลาดนั้นคงที่ด้วยและค่าเฉลี่ยกำลังสองของ  $e(n)$  เป็นฟังก์ชันอันดับสองของสัมประสิทธิ์ ฟิลเตอร์ ค่าผิดพลาดเฉลี่ยกำลังสองที่ขึ้นอยู่กับสัมประสิทธิ์ฟิลเตอร์ ที่ไม่เป็นที่รู้จัก นั้นอยู่ในรูปของ Multidimensional Parabolic ที่มีจุดต่ำสุดที่ถูกกำหนดไว้จุดเดียว หรือจุดที่มีค่าต่ำสุด อะแดปทีฟฟิลเตอร์ จะทำงานในการค้นหาจุดที่ต่ำที่สุดของพื้นที่อย่างต่อเนื่อง จึงต้องทำให้การทำงานของอุปกรณ์ดีที่สุดในไปได้

เมื่อระบบที่ไม่เป็นที่รู้จักนี้เปลี่ยนแปลงไปตามเวลา เป็นผลให้เอาต์พุต ของระบบ รวมถึงการตอบสนองที่ต้องการที่ได้จาก อะแดปทีฟ Tapped-Delay-Line Filter จะเป็นแบบไม่คงที่ ทำนองเดียวกัน การเปลี่ยนแปลงพื้นที่ของประสิทธิภาพการผิดพลาดเปลี่ยนแปลงไปตามเวลาด้วย ในกรณีนี้ อะแดปทีฟอัลกอริทึมใช้ปรับค่าสัมประสิทธิ์ของ Tapped-Delay-line Filter จะถูกเพิ่มค่าสำหรับติดตามจุดต่ำสุดของประสิทธิภาพการผิดพลาดอย่างต่อเนื่อง



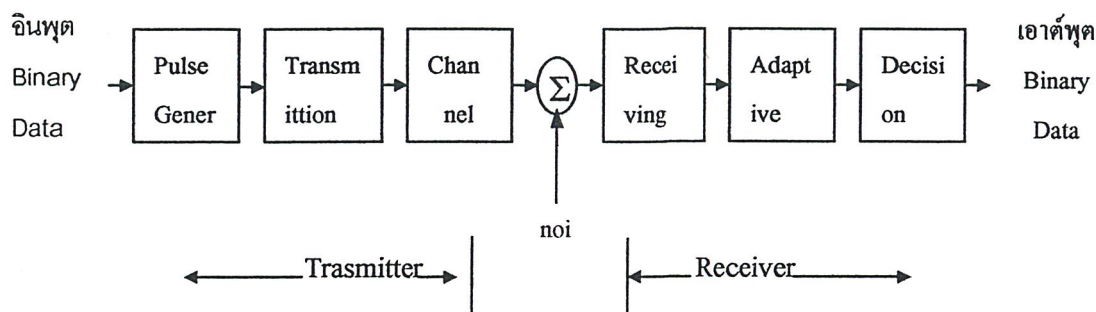
รูปที่ 9 แบบจำลองของระบบที่ไม่เป็นที่รู้จักโดยใช้ Tapped-Delay-Line Filter

## 2.อะแดปทีฟ Equalization For Data Transmission

ในการศึกษาการส่งข้อมูลซึ่งใช้ประโยชน์จากช่องสัญญาณที่มีอยู่ให้มีประสิทธิภาพ จุดมุ่งหมายคือ ออกแบบระบบสำหรับการส่งข้อมูลด้วยอัตราสูงที่สุดเท่าที่เป็นไปได้ ภายใต้ความน่าเชื่อถือที่กำหนดซึ่งปกติถูกวัดอยู่ในรูปของอัตราผิดพลาดหรือความผิดพลาด ความน่าจะเป็นเฉลี่ยของสัญลักษณ์ (Average probability of symbol error) การส่งข้อมูลดิจิทัลผ่านทาง

1. **Intersymbol Interference (ISI)** เกิดจากการแพร่กระจายของรูป Pulse ที่ถูกส่ง ซึ่งเป็นผลมาจากการเบี่ยงเบนของการตอบสนองความถี่ในช่องทาง จากลักษณะทางความคิดที่ ความสูงคงที่และมีเฟสเชิงเส้น เช่น constant delay

2. **Addition Thermal Noise** กำเนิดจากภาคแรกสุดของกลุ่มรับสัญญาณ สำหรับช่องทางที่จำกัดแบนด์วิธเช่น ช่องทางโทรศัพท์ปกติจะพบว่า Intersymbol Interference เป็นองค์ประกอบสำคัญที่ใช้พิจารณาในการออกแบบระบบอัตราส่งข้อมูลสูง



รูปที่ 10 บล็อกไดอะแกรม of a baseband data transmission system

รูปแสดงแบบจำลองเทียบเท่าของระบบข้อมูลสื่อสาร รูปแบบของลำดับข้อมูล  $\{a(n)\}$  ไม่เพียงขึ้นข้อมูลไบนารีที่จะส่งออกไปเท่านั้น แต่ยังขึ้นอยู่กับชนิดของการ Modulate ที่ใช้ด้วย ดังตัวอย่างในกรณีของ M-ary Phase Shift Keying  $a(n)$  ได้มาจากค่า  $M$  จาก  $\exp[j2\pi(m-1)/M]$  เมื่อ  $m=1,2,\dots,M$  โดยปกติ  $M$  จะเท่ากับจำนวนเต็มยกกำลังสอง เมื่อไม่มีนัยสับสนของเอาต์พุตของฟิลเตอร์เครื่องรับในรูปแบบเท่ากับ

$$\tilde{u}(t) = \sum_n a_n p(t - nT) \quad 1.10$$

เมื่อ  $T$  เป็นช่วงเวลาของคาบสัญญาณและ  $p(t)$  เป็นการตอบสนองอิมพัลส์ของจุดต่อลดทอนของฟิลเตอร์ภาคส่งของสัญญาณ และ ฟิลเตอร์ภาครับ โดยการสุ่มตัวอย่าง  $u(t)$  ให้เข้ากันกับภาคส่งและนิยาม  $\tilde{u}(n) = \tilde{u}(nT)$  และ  $p(n) = p(nT)$  ด้านขวามือของสมการ กำหนดสัญลักษณ์ที่ต้องการ และอนุกรมที่เหลือแสดงถึง Intersymbol Interference ที่เป็นผลมาจากการกระทำรวมของฟิลเตอร์ภาคส่งช่องสัญญาณ

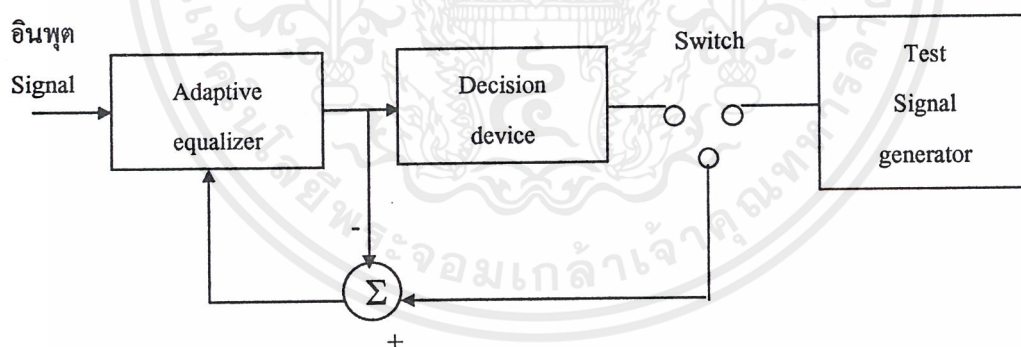
$$\begin{aligned} u(k) &= \sum_n a_n p(k - n) \\ &= a_k p(0) + \sum_{n \neq k} a_n p(k - n) \end{aligned}$$

ฟิลเตอร์ภาครับ Intersymbol Interference นี้ ถ้าปล่อยไว้โดยไม่ตรวจสอบ อาจจะเป็นผลสำหรับการตัดสินใจหาข้อผิดพลาดเมื่อสัญญาณที่ถูกสุ่มที่ ฟิลเตอร์ภาครับถูกเปรียบเทียบกับขอบเขตระดับย่อยที่กำหนดไว้บางตัวโดยได้มาจากอุปกรณ์ตัดสินใจ

เพื่อจะแก้ปัญหา Intersymbol Interference ต้องมีการควบคุมฟังก์ชันของเวลา  $p(t)$  เฉพาะเรื่องนี้ ถ้าลักษณะของช่องส่งสัญญาณเป็นที่รู้อย่างแน่นอนแล้ว จะเสมือนว่ามีทางสำหรับ ออกแบบทั้ง ฟิลเตอร์ภาคส่งและภาครับ ซึ่งจะทำให้ผลของ Intersymbol Interference เวลาสุ่ม ค่อนข้างน้อยและที่เลาเดียวกันจะจำกัดผลของน็อยส์ภาครับที่เพิ่มเข้าไป โดยลดความผิดพลาด ความเป็นไปได้เฉลี่ยของสัญลักษณ์ให้ได้น้อยที่สุด อย่างไรก็ตามในทางปฏิบัติ เราพบว่าช่องสื่อสาร นั้นเป็นแบบสุ่มในกลุ่มของช่องสัญญาณที่เป็นไปได้

ฉะนั้นการใช้คู่ตายตัวของ ฟิลเตอร์ภาคส่งและภาครับถูกออกแบบโดยคุณลักษณะช่องสัญญาณเฉลี่ย อาจจะไม่เพียงพอที่จะลด Intersymbol Interference จึงจะเป็นสิ่งที่ต้องมี อะแดปทีฟ Equalizer ซึ่งจะทำให้ความสามารถของการส่งช่องสัญญาณให้เต็มที่ให้เป็นจริงได้ คำว่า Equalizer ถูกใช้อธิบาย ฟิลเตอร์ที่ใช้บนช่องสัญญาณโทรศัพท์เพื่อทำให้คุณลักษณะความสูงและการหน่วงเวลาของช่องส่งสัญญาณราบเรียบ

โดยพื้นฐานสำหรับ Equalization (การทำให้เท่ากัน) ของระบบการส่งข้อมูลที่มีอยู่ คือ Pre Equalization ที่ภาคส่งและ Post-Equation ที่ภาครับ และเนื่องจากวิธีอย่างแรกต้องใช้ทางย้อนกลับ เราจึงจะพิจารณา Equalization ที่ภาครับ อะแดปทีฟ Equalizer ว่าหลังจาก ฟิลเตอร์ภาครับดังรูปที่ 11



รูปที่ 11 อะแดปทีฟ Filter with a Decision-Directed Mode of Operation

จากรูปแสดง บล็อกไดอะแกรมของ อะแดปทีฟ Equalizer การทำงานที่เกี่ยวข้องกับ Training Mode แล้วถูกตามโดย Training Mode ในช่วง Training Mode สัญญาณทดสอบที่ทราบค่าจะถูกส่งเพื่อวัดช่องสัญญาณทดสอบที่ใช้กันอย่างกว้างขวางประกอบด้วย Maximal-Length Shift จะต้องมีความยาวอย่างน้อยเท่ากับตัว Equalizer เพื่อให้แน่ใจว่า Spectrum ของสัญญาณที่ส่งมีความหนาแน่นเพียงพอในแบนด์วิทของช่องส่งสัญญาณป็นเครื่องรับอะแดปทีฟ Equalizer จะถูกจ่ายด้วยการตอบสนองที่ต้องการเอาต์พุตของ Equalizer ถูกลบออกจากการตอบสนองนี้เพื่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สร้างสัญญาณผิดพลาด ซึ่งใช้ในการปรับสัมประสิทธิ์ของ Equalizer ให้ไปสู่ค่าที่ดีที่สุด ชนิดของ อะแดปทีฟอัลกอริทึมที่เป็นที่นิยมมากที่สุดที่ใช้สำหรับการปรับนี้เกี่ยวข้องกับการพัฒนาให้ดีขึ้นของสัมประสิทธิ์แต่ละตัวของ Equalizer ในแต่ละตัวช่วงของสัญญาณ โดยเริ่มจากค่าเริ่มต้นที่ตั้งไว้

เมื่อช่วง Training เริ่มต้นสำเร็จลง ค่าสัมประสิทธิ์ของ อะแดปทีฟ Equalizer อาจจะถูกเปลี่ยนแปลงต่อไปอีกในลักษณะของ Decision-Directed Mode (mode ของการตัดสินใจโดยตรง) ใน Mode นี้ สัญญาณผิดพลาดจะได้อาจมาจากการประมาณของภาครับช่วงสุดท้าย (ซึ่งไม่จำเป็นต้องเป็นสัญญาณที่ถูกต้อง) ของลำดับการส่ง การประมาณของเครื่องรับได้มาจากการให้เอาต์พุต ของอะแดปทีฟ Equalizer ไปยังอุปกรณ์ตัดสินใจ (Decision Device) ดังแสดงในรูปในการปฏิบัติการปกติ อุปกรณ์ตัดสินใจจะมีความถูกต้องที่มีความเป็นไปได้สูง ดังนั้นการประมาณของสัญญาณผิดพลาดจะถูกต้องบ่อยครั้ง เพียงพอที่จะยอมให้ อะแดปทีฟ Equalizer สามารถปรับค่าสัมประสิทธิ์ ของมัน ได้อย่างถูกต้อง ความสามารถที่น่าสนใจอื่น ๆ ของ Decision Direct อะแดปทีฟ Equalizer คือจริง ๆ แล้วมันสามารถติดตามการเปลี่ยนแปลงอย่างช้า ในคุณลักษณะหรือการกระทำช้า ๆ ของช่องส่งสัญญาณในภาครับด้านหน้า อย่างเช่น การสั่นช้า ๆ ของเฟส

### 3. Digital Representation of Speech

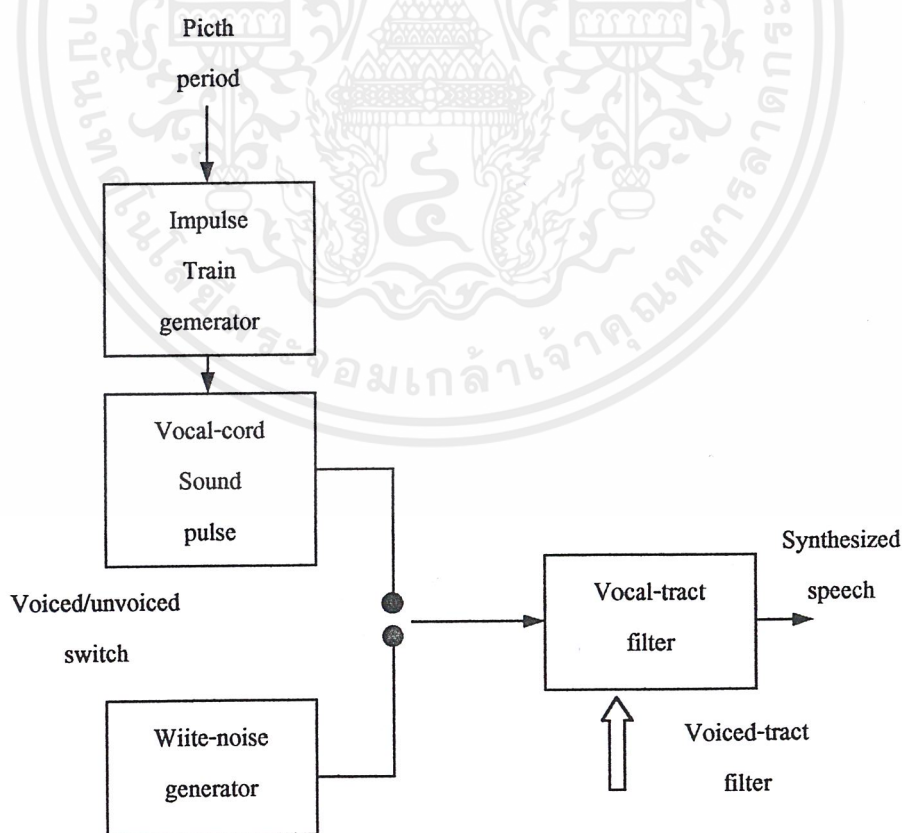
ในช่วง 20 ปีก่อนมีการใช้ขบวนการทางดิจิทัล มากขึ้นสำหรับการเข้ารหัสและการส่งสัญญาณเสียงอย่างมีประสิทธิภาพ องค์ประกอบหลักสองอย่างสำหรับงานนี้คือ อย่างแรกขบวนการ ดิจิตอล มีความเป็นไปได้ที่จะควบคุมผลของน็อยส์ และการรบกวนข้างเคียง ในการสื่อสาร จึงเป็นการปรับปรุงความน่าเชื่อถือของระบบอย่างสำคัญ อย่างที่สองการปฏิวัติทางเทคโนโลยีดิจิทัล การสร้าง Very-Large-Scale Integration VLSI ได้ลดทั้งขนาดและราคาของอุปกรณ์อย่างต่อเนื่อง

Coders (ส่วนเข้ารหัส) สำหรับการแสดงสัญญาณเสียงแบบดิจิทัล แบ่งออกเป็น 2 ชนิดใหญ่คือ Source Coders (ส่วนเข้ารหัสของแหล่งจ่าย) และ Waveform Coders (ส่วนเข้ารหัสของรูปคลื่น) Source Coders นั้นขึ้นอยู่กับแบบจำลอง ซึ่งพวกนี้ใช้ความรู้ก่อนสำหรับสร้างสัญญาณเสียงที่แหล่งจ่าย Source Coders สำหรับการพูดมักจะเรียกกันว่า Vocoder (คำย่อของ voice coder) พวกนี้สามารถปฏิบัติการที่ 4.8 Kbit/s หรือต่ำกว่า อย่างไรก็ตามพวกมันให้คุณภาพที่ประกอบขึ้นคลื่นเสียงที่ส่งมาสร้างขึ้นใหม่ในจุดนี้ Coders ตัวนี้จึงไม่ขึ้นอยู่กับสัญญาณ และอาจจะถูกออกแบบเพื่อสร้างคุณภาพโทรศัพท์สำหรับการพูดที่อัตราการเข้ารหัสต่ำ ๆ เท่ากับ 16 Kbit/s

แบบจำลองของกระบวนการสร้างเสียง รูปที่ 12 แสดง บล็อกไดอะแกรม อย่างง่ายของแบบจำลองเก่า สำหรับการสร้างเสียง ซึ่งสมมติให้ขบวนการกำเนิดเสียงเป็นแบบเชิงเส้น เช่น แหล่งของการกระตุ้น และแยกออกจาก Intelligence-Modulating Vocal-Tract Filter ได้ รูปแบบที่แน่นอนของการกระตุ้นขึ้นอยู่กับว่าเสียงพูดเป็นเสียงคนพูดหรือไม่ ดังที่อธิบายดังนี้

1. **Voice Speech Sound** ถูกกำเนิดจากเสียงของหลอดเสียงที่ลากต่อกัน เครื่องกำเนิดขบวนการของอิมพัลส์สร้างอันดับของอิมพัลส์เป็นพัลส์ที่สั้นมาก ๆ ซึ่งเว้นว่างด้วยช่วงพื้นฐานเท่ากับช่วงของเสียงสูง ในสัญญาณนี้จะกระตุ้นฟิลเตอร์เชิงเส้นที่มีการตอบสนองอิมพัลส์ของเสียงจากหลอดเสียง

2. **Unvoiced Speech Sound** ถูกกำเนิดจากเสียงแบบสุ่มจากอากาศที่ไหลและสั้นอย่างรวดเร็ว ในกรณีนี้ตัวกระตุ้นประกอบด้วยง่าย ๆ เพียง White Noise Source ความเป็นไปได้ในการกระจายของตัวอย่างน้อยสั้นนี้ไม่เป็นจุดสำคัญเลย การตอบสนองความถี่ของ Vocal-Tract Filter สำหรับ Unvoice Speech หรือ กลุ่มของเสียงนั้นถูกคูณด้วย Spectrum ของ Vocal-card Sound Pulse (pulses ของเสียงจากหลอดเสียง) และได้เป็น Short-Time Spectral Envelope ของสัญญาณเสียง



รูปที่ 12 บล็อกไดอะแกรม ของแบบจำลองอย่างง่ายสำหรับการผลิตเสียงพูด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเข้ารหัสคาดหมายแบบเชิงเส้น (Linear Predictive Coding) ขบวนการของ Linear Predictive Coding; LPC เป็นตัวอย่างของ Source Coding (การเข้ารหัสแหล่งจ่าย) ขบวนการนี้สำคัญมากเพราะว่ามันไม่เพียงแต่ให้เทคนิคที่มีอำนาจมากสำหรับการส่งเสียงแบบดิจิทัลที่อัตราบิตต่ำแล้ว มันยังประมาณค่าพารามิเตอร์พื้นฐานของการพูดอย่างถูกต้องด้วย

การพัฒนาของ LPC ขึ้นอยู่กับแบบจำลองของรูป สำหรับขบวนการผลิตเสียงพูดการตอบสนองความถี่ของ Vocal Tract ( กลุ่มของเสียง ) สำหรับ Unvoice Speech หรือ Vocal Tract ที่ถูกคูณ โดย Spectrum ของ Vocal-Cord Sound Pulse สำหรับเสียงพูด สามารถเขียนเป็น ทราสเฟอร์ฟังก์ชัน

$$H(z) = \frac{G}{1 + \sum_{k=1}^M a(k) * z^{-k}} \quad 1.11$$

เมื่อ G เป็นเกนพารามิเตอร์และ  $z^{-1}$  เป็น Unit-Delay Operator รูปแบบของตัวกระตุ้นที่ให้กับฟิลเตอร์นี้ถูกเปลี่ยนโดย Switching ระหว่าง Voiced และ Unvoice Sound ฉะนั้นฟิลเตอร์ที่มีทราสเฟอร์ฟังก์ชัน  $H(z)$  ถูกกระตุ้นด้วยอันดับของอิมพัลส์เพื่อสร้างเสียงหรือ White-Noise Sequence เพื่อสร้าง Unvoice Sounds ใน LPC Linear Prediction (การคาดหมายแบบเชิงเส้น) ถูกใช้เพื่อประมาณค่าพารามิเตอร์ของเสียง ให้กลุ่มของตัวอย่างในอดีตของสัญญาณเสียงให้ชื่อว่า  $u(n-1), u(n-2), \dots, u(n-M)$  การคาดหมายเชิงเส้นของ  $u(n)$  ที่เป็นค่าตัวอย่างปัจจุบัน ถูกกำหนดโดย

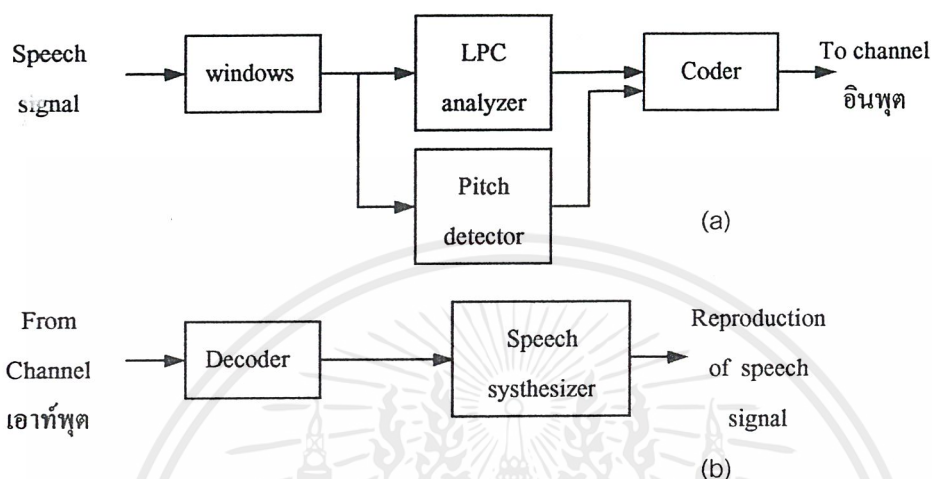
$$\tilde{u}(n, n-1, \dots, n-M) = \sum_{k=1}^M h(k) * u(n-k) \quad 1.12$$

สัมประสิทธิ์ของตัวคาดหมาย  $h(1), h(2), \dots, h(M)$  ถูกทำให้ดีที่สุดโดยทำให้ค่าเฉลี่ยกำลังสองของความผิดพลาดในการคาดหมาย  $e(n)$  มีค่าต่ำที่สุด โดยที่  $e(n)$  กำหนดจากความแตกต่างระหว่าง  $u(n)$  และ  $u(n), n-1, \dots, n-M$  การใช้บรรทัดฐานความผิดพลาดเฉลี่ยกำลังสองต่ำที่สุดสำหรับการทำให้ส่วนคาดหมายดีที่สุด อาจจะต้องถูกปรับเพื่อให้เข้ากับเหตุผลพื้นฐานสองประการ

1. ถ้าสัญญาณเสียงพูดสามารถใช้ได้กับแบบจำลองที่อธิบายไว้ในสมการ และถ้าค่าเฉลี่ยกำลังสองของสัญญาณผิดพลาด  $e(n)$  ถูกทำให้ต่ำที่สุดแล้ว เราพบว่า  $e(n)$  เท่ากับตัวกระตุ้น  $x(n)$  คูณกับเกนพารามิเตอร์ G ในแบบจำลองในรูปที่ 13 และ  $\tilde{u}(n-1), u(n-2), \dots, u(n-M)$

ฉะนั้นสัญญาณผิดพลาด  $e(n)$  ประกอบด้วยขบวนการของอิมพัลส์ในกรณีของ Unvoiced Sound ในกรณีทั้งคู่ สัญญาณผิดพลาด  $e(n)$  อาจจะมีขนาดเล็กมากตลอดเวลา

## 2. การใช้บรรทัดฐานผิดพลาดเฉลี่ยกำลังสองต่ำสุดจะนำไปสู่คณิตศาสตร์ที่หาได้

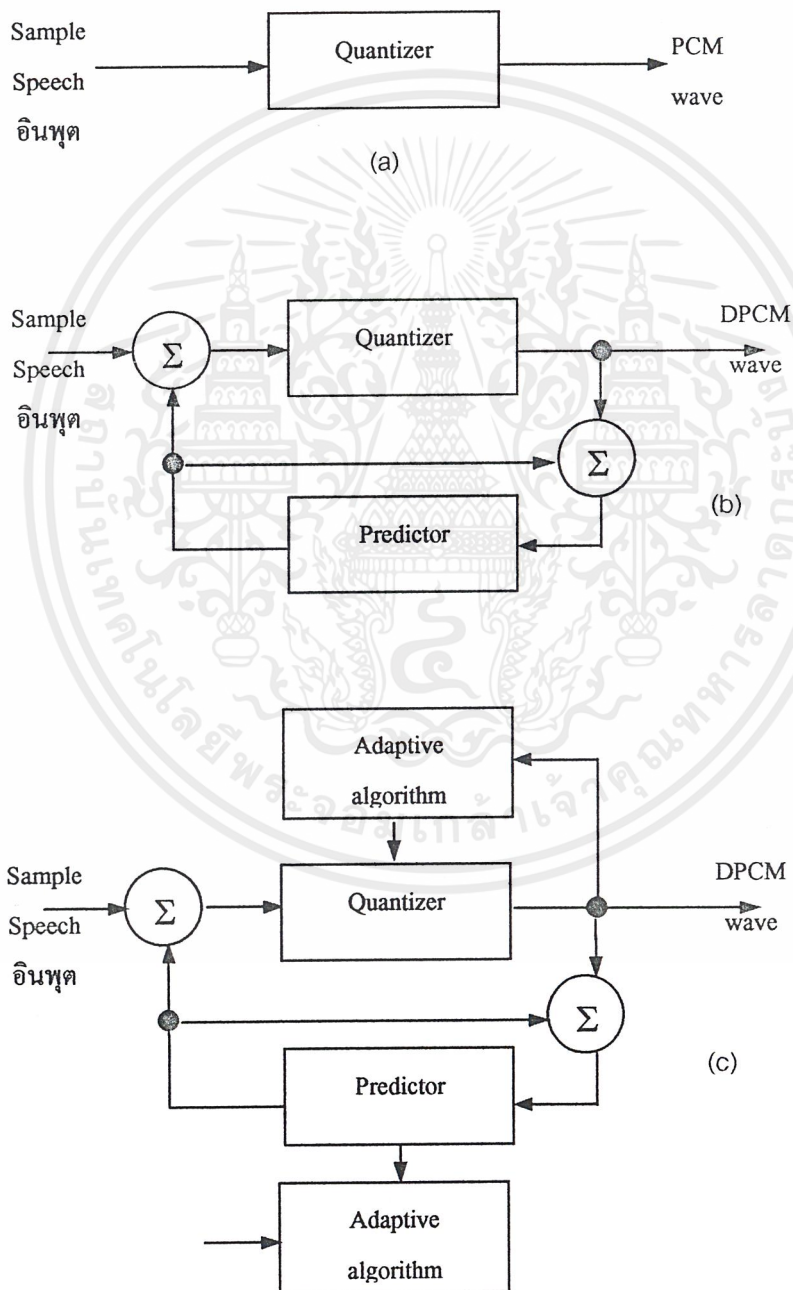


รูปที่ 13 บล็อกไดอะแกรม of LPC Vocoder (a) Transmitter (b) Receiver

รูปแสดงบล็อกไดอะแกรมของ LPC Vocoder ประกอบด้วยเครื่องส่งและเครื่องรับ เครื่องส่งแรกสุดจะส่ง window (โดยปกติมีขนาด 10-30 ms) เข้ากับสัญญาณเสียงอินพุต และกลายเป็นบล็อกของตัวอย่างเสียงพูดสำหรับเข้าขบวนการ window นี้จะสั้นพอที่จะทำให้รูปร่างกลุ่มเสียงเกือบจะคงที่ดังนั้นพารามิเตอร์ของการสร้างเสียงพูดในแบบจำลอง อาจถือว่าเป็นค่าคงที่ที่สำคัญสำหรับช่วงเวลาของ window จากนั้นเครื่องส่งจะวิเคราะห์สัญญาณเสียงพูดอินพุต ในลักษณะของ อะแดปทีฟบล็อกต่อบล็อก โดยการกระทำการคาดหมายอย่างเชิงเส้นและการตรวจจับความสูงของเสียง ท้ายสุดมันจะสร้างรหัสพารามิเตอร์; (1) กลุ่มของสัมประสิทธิ์ส่วนคาดหมาย (2) ช่วงความสูงของเสียง (3) เกนพารามิเตอร์ และ (4) Voiced/Unvoiced Parameter ถ้ารับการส่งไปทั่วช่องส่งสัญญาณ เครื่องรับจะทำปฏิบัติการกลับกัน โดยแรกสุดถอดรหัสพารามิเตอร์ที่เข้ามา ในจุดนี้มันจะคำนวณค่าของสัมประสิทธิ์ส่วนคาดหมาย ช่วงความสูงของเสียงพูด และ เกนพารามิเตอร์และพิจารณาว่าส่วนที่สนใจเป็นเสียงแบบ Voiced หรือ Unvoiced สุดท้าย เครื่องรับจะใช้ค่า Parameters เหล่านี้ในการสังเคราะห์สัญญาณเสียงพูดโดยใช้แบบจำลอง

การเข้ารหัสรูปคลื่น (Waveform Coding) การเข้ารหัสรูปคลื่นที่กระทำบนสัญญาณเสียง ถูกออกแบบเพื่อรักษารูปร่างของสัญญาณ โดยเฉพาะการปฏิบัติการนี้รวมถึง Sampling (การสุ่ม Time Discrete) และ Quantization (การประกอบ Amplitude Discrete) อัตราส่วนของการสุ่มจะเป็นตามคุณสมบัติพื้นฐานของสัญญาณเสียงพูดทั้งหมด ที่มีชื่อว่า bandlimited นี้หมายถึงว่า

สัญญาณเสียงพูดสามารถสุ่มในเวลาอัตราจำกัด ที่เหมาะสมกับทฤษฎีการสุ่ม ตัวอย่างเช่น เครื่องถ่ายรูกริกิโทรส์ท์ที่ถูกออกแบบสำหรับส่งสัญญาณเสียงพูดที่ใช้ แบนด์วิท (300-3,200Hz) เพื่อจะ เป็นไปตามทฤษฎี อัตราการสุ่มที่คงไว้ 8 kHz จะถูกใช้โดยทั่วไปในปฏิบัติ Quantization ถูก ปรับบนพื้นฐานเหล่านี้ และแม้ว่าสัญญาณเสียงพูดจะมีช่วงต่อเนื่องของความสูง (และแน่นอนตัว อย่างของเสียงนี้ก็มีส่วนความสูงที่ต่อเนื่อง) ไม่ว่าจะอย่างไร มันก็ไม่จำเป็นที่จะส่งความสูงได้จำกัด เท่านั้น



รูปที่ 14 ตัวเข้ารหัสรูปคลื่น (a) PCM (b) DPCM (c) ADPCM

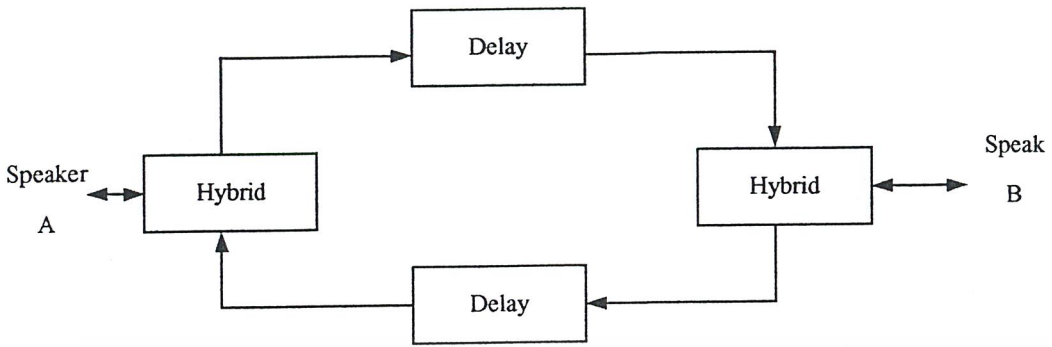
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างของการเข้ารหัสรูปคลื่นรวมถึง Pulse-Code Modulation; PCM และ Differential Pulse-Code Modulation DPCM ใน PCM ดังที่ ถูกใช้ในการโทรศัพท์ สัญญาณเสียงพูด หลังจากผ่าน Low Pass Filter ไปแล้ว จะถูกสุ่มที่อัตรา 8 kHz เป็นแบบ Quantized ไม่เชิงเส้น และจากนั้นเข้ารหัสเป็น 8-bit words ดังรูป ผลที่ได้จะเป็น Signal to Quantization Noise Ratio ที่ดีบนช่วงเปลี่ยนแปลงของระดับสัญญาณ อินพุต DPCM จะเกี่ยวกับการใช้ส่วนคาคหมายดังรูป (b) ส่วนคาคหมายถูกออกแบบเพื่อใช้ประโยชน์จากความสัมพันธ์ซึ่งมีอยู่ระหว่างตัวอย่างข้างเคียงของสัญญาณเสียง เพื่อที่จะทำการลดจำนวนของบิตที่ถูกใช้สำหรับการส่งแต่ละตัวอย่างของสัญญาณและจากนั้นเข้ารหัสความผิดพลาดของการคาคหมาย ซึ่งเป็นผลจากการลบกันจาก เอาท์พุต ของส่วนคาคหมายจากสัญญาณอินพุต ถ้าส่วนคาคหมายถูกทำให้ดีที่สุดแล้วการเปลี่ยนแปลงของความผิดพลาดของส่วนคาคหมายจะน้อยกว่าของสัญญาณอินพุต ถูก Quantized โดยตรงในระบบ PCM มาตรฐาน เหมือนกันสำหรับความผิดพลาดของ Quantizing ของการเปลี่ยนแปลงที่กำหนดได้ DPCM ต้องการจำนวนของระดับ Quantizing น้อยกว่า และแน่นอนกว่ามีอัตรา bit น้อยกว่า PCM

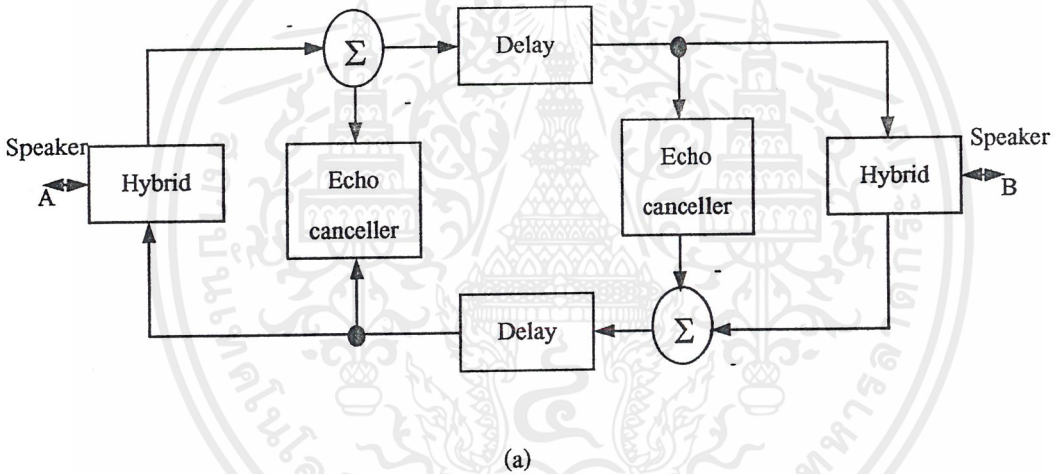
DPCM ใช้ Quantizer คงที่ และส่วนคาคหมายที่คงที่ การส่งสามารถทำได้ต่อไปโดยใช้อะแดปทีฟ Quantizer และ อะแดปทีฟ Predictor ดังแสดงในรูป (c) การเข้ารหัสของรูปคลื่นชนิดนี้ถูกเรียกว่า อะแดปทีฟ Differential Pulse-Code Modulation; ADPCM อะแดปทีฟ Predictor ถูกใช้เพื่อที่จะบันทึกธรรมชาติที่ไม่คงที่ของสัญญาณเสียง

#### 4. Echo Cancellation

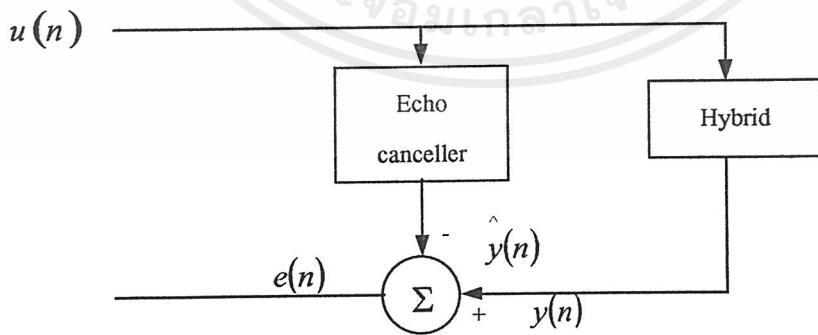
ในการต่อโทรศัพท์ซึ่งเกี่ยวข้องทั้งการส่งแบบสี่สายและส่งแบบสองสาย Echo (เสียงก้อง) นั้นกำเนิดที่ Hybrid ซึ่งต่ำกับการส่งแบบสี่สายไปยังสองสาย เมื่อมีการเรียกโทรศัพท์ทางไกล เช่น โดยการใช้ดาวเทียมประจำที่ Geostationary Satellites Echo จะแสดงถึงการลดทอนซึ่งอาจจะเป็นสิ่งที่น่ารำคาญมากกว่าเสียงเบาหรือมีน้อยรูปที่ 15 แสดงว่าวงจรดาวเทียมซึ่งไม่มีการป้องกัน Echo Hybrids ที่ปลายทั้งสองข้างของวงจรเปลี่ยนการส่งแบบสองสายที่ใช้งาน Customer Loops และกลุ่มสายส่งแบบสี่เส้นเป็นสิ่งจำเป็นสำหรับส่งวงจรพาหะ ช่วงเวลาสูงสุดของดาวเทียมการหน่วงเวลา 270 ms จะเกิดขึ้นในแต่ละทางบนสายแบบสี่สาย โดยแนวความคิด เมื่อมีนาย ก พูดอยู่ทางซ้าย เสียงพูดของเขาควรจะตามทางสายส่งด้านบนไปยัง Hybrid ทางด้านขวา และจะเข้าสู่วงจรสองสายอย่างไรก็ตามในทางปฏิบัติพลังงานเสียงไม่ทั้งหมดที่จะตรงเข้าวงจรสองสาย เป็นผลให้สัญญาณบางส่วนย้อนกลับทางสี่สายด้านล่างทำให้ได้ยิน โดยบุคคลด้านซ้ายจึงเป็น Echo ที่มี การหน่วงเวลาเท่ากับ 540 ms



รูปที่ 15 วงจรดาวเทียมที่ไม่มีกำบังเสียงสะท้อน



(a)



(a)

รูปที่ 16 (a) วงจรดาวเทียมพร้อมกับคู่ของเครื่องกำจัดเสียงสะท้อน (b) นิยามของสัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เพื่อจะชนะปัญหานี้ Echo canceller (เครื่องตัดเสียงสะท้อน) จึงถูกติดตั้งไว้ในเครือข่าย เป็นคู่ดังแสดงดังรูปที่ 16 การกำจัดนี้จะทำได้โดยการประมาณค่าการสะท้อนและลบจากสัญญาณ ที่กลับมา การสมมติข้างล่างนี้เป็นทางกลับของเสียงสะท้อน จากจุดที่ Canceller ต่อที่จุดที่ค่า ประมาณเสียงสะท้อนถูกลบ เป็นเชิงเส้นและไม่ขึ้นอยู่กัเวลา

ฉะนั้นโดยการอ้างอิง Canceller เดียวในรูป เมื่อกำหนดสัญญาณที่ส่งกลับที่เวลา  $n$  อาจ จะแสดงได้เป็น

$$y(n) = \sum_{k=0}^{\infty} h(k) * u(n-k) + v(n) \quad 1.13$$

เมื่อ  $u(n), u(n-1), \dots, u(n-M)$  เป็นตัวอย่างของเสียงพูดที่ไกลสุดจากผู้พูด  $v(n)$  เป็นเสียง พูดที่ใกล้ที่สุดจากผู้พูด  $x$  บวกเข้ากับ Noise ที่เพิ่มเข้ามาที่เวลา  $n$  และ  $h(k)$  เป็นการตอบสนอง Impulse ของเสียงสะท้อน เครื่องตัดเสียงสะท้อนจะทำหน้าที่ประมาณ  $h(k)$  ของการตอบสนอง Impulse ของเส้นทางสะท้อน และจากนั้นประมาณเสียงสะท้อนออกมาเป็น Convolution Sum (ผลรวมผสม)

$$\hat{y}(n) = \sum_{k=0}^M \hat{h}(k) * u(n-k) \quad 1.14$$

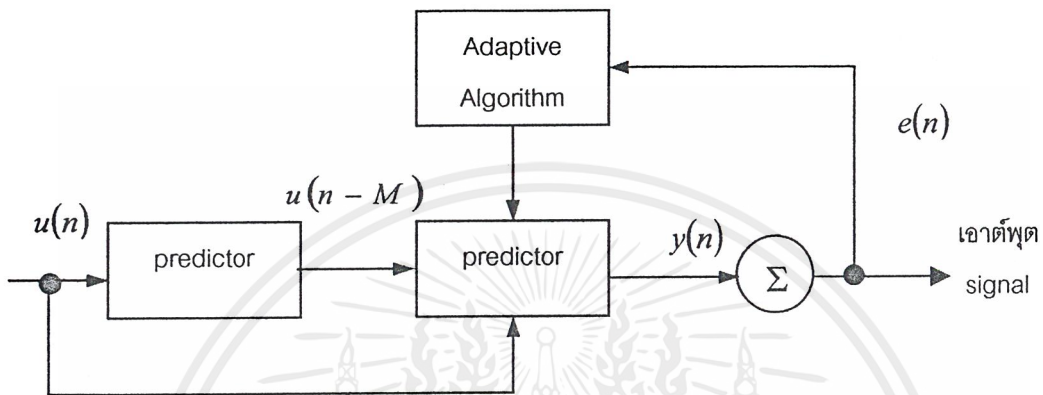
ซึ่งสามารถทำได้โดยใช้ Tapped - Delay - Line Filter กับสัมประสิทธิ์  $\hat{h}(0), \hat{h}(1), \dots, \hat{h}(M)$  สัญญาณ ผิดพลาด  $e(n)$  จะถูกสร้างโดยการลบค่าประมาณ  $\hat{y}(n)$  จาก สัญญาณย้อนกลับ  $y(n)$  ดังแสดงโดย

$$e(n) = y(n) - \hat{y}(n) \quad 1.15$$

สัญญาณผิดพลาด  $e(n)$  ใช้สำหรับควบคุมการเปลี่ยนแปลงของสัมประสิทธิ์ของ Canceller  $\hat{h}(0), \hat{h}(1), \dots, \hat{h}(M)$  ดังนั้นหลังจากที่ผ่านการคำนวณซ้ำ เล็กน้อยแล้ว จะรู้ดีกว่าผล ของการสะท้อนจะน้อยที่สุด

## 5. Adaptive Line Enhancer

อะแดปทีฟ Line Enhancer; ALE เป็นอุปกรณ์ที่สามารถใช้ตรวจสอบคลื่นชายส์ ระดับต่ำที่ถูกฝังไว้ส่วนหลังของน็อยส์ที่เพิ่มขึ้นมาบน Spectrum ที่มี Wire Band



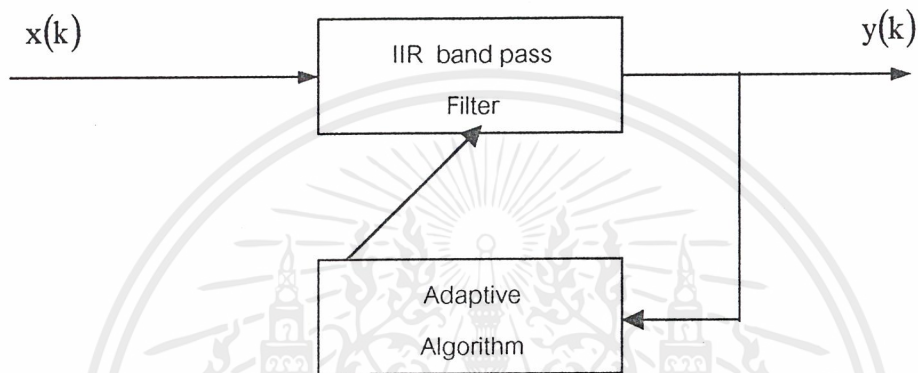
รูปที่ 17 อะแดปทีฟ Line Enhancer

ALE ประกอบด้วยอุปกรณ์หน่วงเวลาและ Linear Predictor (ส่วนคาดหมายเชิงเส้น) เอาต์พุต ของ Predictor  $y(n)$  จะถูกลบกับสัญญาณ  $u(n)$  เพื่อสร้างสัญญาณผิดพลาด  $e(n)$  สัญญาณนี้จะถูกใช้เพื่อควบคุมการเปลี่ยนแปลงของสัมประสิทธิ์ Predictor เท่ากับ  $u(n-D)$  สัญญาณ อินพุต ดั้งเดิม  $u(n)$  ถูกหน่วงด้วย  $D$  วินาที เมื่อ  $D$  นั้นเท่ากับหรือมากกว่าช่วงเวลาของการสุ่ม หน้าที่หลักของ Delay-Parameter  $D$  คือ ตัดความสัมพันธ์ซึ่งอาจจะมีอยู่ระหว่างองค์ประกอบของน็อยส์ใน Predictor อินพุต ที่หน่วงเวลาแล้ว  $u(n-D)$  สำหรับเหตุผลนี้ค่า ดีเลย์พารามิเตอร์  $D$  จึงถูกเรียกอีกชื่อว่า Deceleration Parameter ของ ALE อาจจะถูกมองเป็น อะแดปทีฟฟิลเตอร์ ซึ่งถูกออกแบบเพื่อตัดองค์ประกอบแบบ wire Band เช่น Sine Waves ถูกส่งด้วยการลดทอนเล็กน้อย พุคได้ก็อีกอย่างว่ามันสามารถใช้เพื่อเพิ่มความมี ประสิทธิภาพ การปรากฏของรากของ Sine wave ซึ่ง Spectrum ประกอบด้วย Harmonic Lines

## บทที่ 2

### การวิเคราะห์และออกแบบ

อะแดปทีฟ IIR ดิจิตอลแบนด์พาสฟิลเตอร์สามารถแสดงได้ดังรูปที่ 2.1



รูปที่ 2.1 บล็อกไดอะแกรมของ อะแดปทีฟ IIR แบนด์พาสฟิลเตอร์

จากบล็อกไดอะแกรมในรูปที่ 2.1 สัญญาณอินพุตประกอบด้วยสัญญาณคลื่นไซน์และ  
นอยส์  $(x(k) = A \cos(k \cdot \omega) + n(k))$  วงจร IIR แบนด์พาสฟิลเตอร์จะถูกปรับเปลี่ยนค่า  
โดย อะแดปทีฟอัลกอริทึม (Adaptive algorithm) เพื่อให้ได้เอาต์พุตตามต้องการ ในที่นี้เอาต์พุตจะ  
เป็นสัญญาณรูปคลื่นไซน์ก็ต่อเมื่อค่า  $\alpha_0(k) = \cos(\omega_0)$  ค่า  $\alpha_1(k)$  ในวงจรรองความถี่มีค่า  
เท่ากับ  $\cos(\omega_0)$  นั่นหมายความว่าวงจรสามารถขจัดสัญญาณรบกวนออกไปได้

## 2.1 อะแดปทีฟอัลกอริทึม แบบสเกลาร์

กำหนดให้ช่วงเวลาการแซมปลิง เป็น  $T$  วินาที ถูกนอร์มอลไลซ์อยู่ที่ 1 ตัวแปร  $k$  (เลขจำนวนเต็ม) เป็นเวลาการแซมปลิง และให้สมการอินพุต เป็น

$$x(k) = A \cos(k\omega) + n(k) \quad 2.1$$

โดยที่  $x(k)$ : สัญญาณทางด้านอินพุต ซึ่งประกอบด้วยคลื่นไซน์

$A$ : ค่าคงที่

$\omega$ : ความเร็วเชิงมุม มีค่าอยู่ระหว่าง  $0 < \omega < \pi$

$n(k)$ : สัญญาณรบกวนแบบเกาส์

สำหรับทรานสเฟอร์ฟังก์ชันของวงจร IIR แบบคัพทไฟลเตอร์อันดับ 2 ที่มีความถี่กลางเป็น  $\omega_0$  คือ

$$H(z) = \frac{1 - \alpha_0}{2} \cdot \frac{1 - z^{-2}}{1 - \alpha_1(k)(1 + \alpha_0)z^{-1} + \alpha_0 z^{-2}} \quad 2.2$$

เมื่อ  $\alpha_1(k)$  มีค่า  $-1 < \alpha_1(k) < 1$  เป็นสัมประสิทธิ์แปรค่าที่ใช้กำหนด  $\omega_0$

$$\alpha_1(k) = \cos(\omega_0) \quad 2.3$$

ซึ่งค่า  $Q$ -Factor มีค่าเท่ากับ

$$Q = \frac{\omega_0}{2(1 - \sqrt{\alpha_0})} \quad 2.4$$

เมื่อ  $\alpha_0$  เป็นสัมประสิทธิ์ค่าคงที่ ที่กำหนดค่า  $Q$ -Factor ของวงจรกรองความถี่มีค่า  $0 < \alpha_0 < 1$

จากสมการ 2.2 สามารถหาค่า  $y(k)$  เมื่อ  $Y(z)$  เป็น Z-Transform ของ  $y(k)$  ความสัมพันธ์ของทรานสเฟอร์ฟังก์ชันจะเป็น  $H(z) = \frac{Y(z)}{X(z)}$

$$Y(z) = H(z) \cdot X(z) \quad 2.5$$

$$Y(z) = \left( \frac{1-\alpha_0}{2} \right) \left( \frac{1-Z^{-2}}{1-\alpha_1(k)(1+\alpha_0)Z^{-1} + \alpha_0 Z^{-2}} \right) X(z) \quad 2.6$$

$$Y(z)(1-\alpha_1(k)(1+\alpha_0)Z^{-1} + \alpha_0 Z^{-2}) = \left( \frac{1-\alpha_0}{2} \right) (1-Z^{-2})X(z)$$

$$Y(z) - \alpha_1(k)(1+\alpha_0)Y(z)Z^{-1} + \alpha_0 Y(z)Z^{-2} = \left( \frac{1-\alpha_0}{2} \right) (X(z) - X(z)Z^{-2})$$

Take Inverse Z-Transform

$$(y(k) - \alpha_1(k)(1+\alpha_0)y(k-1) + \alpha_0 y(k-2)) = \left( \frac{1-\alpha_0}{2} \right) (x(k) - x(k-2))$$

$$y(k) = \left( \frac{1-\alpha_0}{2} \right) (x(k) - x(k-2)) + \alpha_1(k)(1+\alpha_0)y(k-1) - \alpha_0 y(k-2) \quad 2.7$$

ค่าเฉลี่ยกำลังสอง ของ  $[y(k)]^2$  จะเป็นสัดส่วนโดยตรงกับ  $|H(e^{j\omega})|^2$  ที่  $z = e^{j\omega}$  เมื่อใช้วิธีการของ Stochastic Gradient ค่า  $\alpha_1(k)$  ที่ทำให้  $[y(k)]^2$  มีค่าสูงสุดจะแคปที่พ้อลกอริทึมสามารถกำหนดได้ดังสมการ

$$\begin{aligned} \alpha_1(k+1) &= \alpha_1(k) + \frac{\mu}{2} \cdot \frac{\partial y^2(k)}{\partial \alpha_1(k)} \\ &= \alpha_1(k) + \mu \cdot y(k) \cdot \psi(k) \end{aligned} \quad 2.8$$

เมื่อ  $\psi(k) = \frac{\partial y(k)}{\partial \alpha_1(k)}$  เป็นสัญญาณควบคุมอะแคปทีฟ  $\alpha_1(k)$

$\mu$  เป็น Step size parameter ถ้าให้ค่านี้นี้เล็กพอจะทำให้เกิดการเปลี่ยนแปลงของ  $y(k)$  ในแต่ละครั้งมีค่าน้อย เราสามารถประมาณสัญญาณควบคุมอะแคปทีฟได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

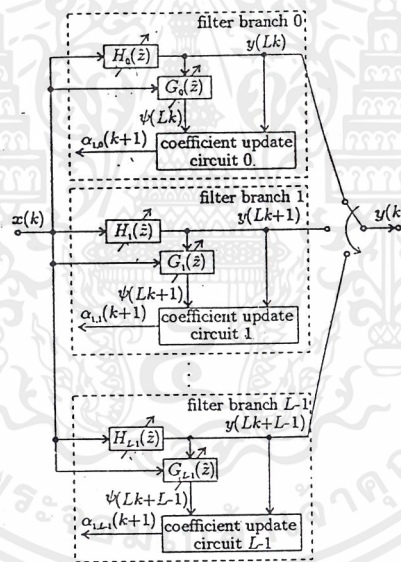
$$\psi(k-i) = \frac{\partial y(k-i)}{\partial \alpha_1(k-i)} \approx \frac{\partial y(k-i)}{\partial \alpha_1(k)} \quad \text{เมื่อ } i \text{ เป็นจำนวนเต็มบวกใดๆ}$$

หาค่า  $\psi(k)$

$$\psi(k) = \alpha_1(k) \cdot (1 + \alpha_0) \cdot \psi(k-1) - \alpha_0 \cdot \psi(k-2) + (1 - \alpha_0) \cdot y(k-1) \quad 2.9$$

## 2.2 อะแดปทีฟอัลกอริทึมโครงสร้างแบบขนาน

ทรานสเฟอร์ฟังก์ชันในสมการที่ 2.2 สามารถหาค่า อินพุตเวกเตอร์  $x(k)$  และเอาต์พุตเวกเตอร์  $y(k)$  จากโครงสร้างของอะแดปทีฟแบบขนาน  $L$ -Order ซึ่งในโครงงานนี้ใช้ อะแดปทีฟแบนด์พาสฟิลเตอร์แบบขนาน แสดงได้ดังรูป



รูปที่ 2.2 โครงสร้างแบบขนาน

สามารถกำหนดอินพุต เอาต์พุต ของวงจร ได้ดังสมการ

$$x(k) = [x(Lk) \cdot x(Lk+1) \dots x(Lk+L-1)]^T \quad 2.10$$

$$y(k) = [y(Lk) \cdot y(Lk+1) \dots y(Lk+L-1)]^T \quad 2.11$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Z-Transform ของสมการที่ 2.10 และ 2.11 สามารถหาค่าได้ดังนี้

$$X(z) = [X_0(z) \cdot X_1(z) \dots X_{L-1}(z)]^t \quad 2.12$$

$$Y(z) = [Y_0(z) \cdot Y_1(z) \dots Y_{L-1}(z)]^t \quad 2.13$$

เมื่อ  $t$  แทน Transpose โดยทรานสเฟอร์ฟังก์ชันแมทริก  $H(z)$  ที่มี  $L$  เป็นอินพุต และ เอาต์พุต สามารถเขียนอยู่ในรูป Toeplitz ดังนี้

$$Y(z) = H(z)X(z) \quad 2.14$$

$$H(z) = \frac{1 - \alpha_0}{z} \cdot \frac{R(z)}{D(z)} \quad 2.15$$

$$D(z) = 1 - a_1 z^{-1} + a_2 z^{-2} \quad 2.16$$

$$R(z) = \begin{bmatrix} r_0(z) & r_1(z) & \dots & r_{L-1}(z) \\ r_1(z) & r_0(z) & \dots & \cdot \\ \cdot & \cdot & \dots & r_1(z) \\ r_{-(L-1)}(z) & \dots & r_{-1}(z) & r_0(z) \end{bmatrix} \quad 2.17$$

$$r_s(z) = \begin{cases} b_{s,1} z^{-1} + b_{s,2} z^{-2} & (s > 0) \\ 1 + b_{s,1} z^{-1} + b_{s,2} z^{-2} & (s = 0) \\ b_{L+s,1} + b_{L+s,2} z^{-1} & (s < 0) \end{cases} \quad 2.18$$

เมื่อสัมประสิทธิ์ของ  $r_1(z)$  มีค่าดังนี้

$$a_1 = \sum_{m=0}^{\lfloor L/2 \rfloor} \frac{(-1)^m L}{L-m} \binom{L-m}{m} * [\alpha(h)(1 + \alpha_0)]^{L-2m} \alpha_0^m \quad 2.19$$

$$a_2 = \alpha_0^L \quad 2.20$$

$$b_{0,1} = -(1 - \alpha_0) \sum_{m=0}^{\lfloor (L-2)/2 \rfloor} (-1)^m \binom{L-m-2}{m} * [\alpha_1(k)(1 + \alpha_0)]^{L-2m-2} \alpha_0^m \quad 2.21$$

$$b_{0,2} = -\alpha_0^{L-1} \quad 2.22$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\begin{aligned}
b_{l,1} &= \sum_{m=0}^{\lfloor (L-l)/2 \rfloor} (-1)^m \binom{L-l-m}{m} * [\alpha_1(k)(1+\alpha_0)]^{L-l-2m} * \alpha_0 \\
&- \sum_{m=0}^{\lfloor (L-l-1)/2 \rfloor} \frac{(-1)^m}{2} \binom{L-l-m-1}{m} * [\alpha_1(k)(1+\alpha_0)]^{L-l-2m} * \alpha_0^{m-1} \\
&+ \sum_{m=0}^{\lfloor (L-l)/2 \rfloor} \frac{(-1)^m (L-1)}{2(L-l-m)} \binom{L-l-m}{m} * [\alpha_1(k)(1+\alpha_0)]^{L-l-2m} * \alpha_0^{m-1} \quad 2.23
\end{aligned}$$

$$\begin{aligned}
b_{l,2} &= \sum_{m=0}^{\lfloor (l-1)/2 \rfloor} \frac{(-1)^m}{2} \binom{l-m-1}{m} * [\alpha_1(k)(1+\alpha_0)]^{l-2m} * \alpha_0^{L-l+m} \\
&- \sum_{m=0}^{\lfloor (L)/2 \rfloor} \frac{(-1)^m l}{2(l-m)} \binom{l-m}{m} * [\alpha_1(k)(1+\alpha_0)]^{l-2m} * \alpha_0^{L-l+m} \\
&- \sum_{m=0}^{\lfloor (L)/2 \rfloor} (-1)^m \binom{l-m}{m} * [\alpha_1(k)(1+\alpha_0)]^{l-2m} * \alpha_0^{L-l+m-1} \quad 2.24
\end{aligned}$$

ค่าเฉลี่ยกำลังสองของ  $[y(Lk+l)]^2$  จะเป็นสัดส่วนโดยตรงกับ  $|H(e^{j\omega})|^2$  ที่  $z = e^{j\omega}$  เมื่อใช้วิธีการของ Stochastic Gradient ค่า  $\alpha_1(k)$  ที่ทำให้  $[y(Lk+l)]^2$  มีค่าสูงสุด ซึ่งอะแดปทีฟ อัลกอริทึม สามารถกำหนดได้ดังสมการ

$$\begin{aligned}
\alpha_{1,l}(k+1) &= \alpha_{1,l}(k) + \frac{\mu}{2} \cdot \frac{\partial y^2(Lk+l)}{\partial \alpha_{1,l}(k)} \\
&= \alpha_{1,l}(k) + \mu \cdot y(Lk+l) \psi(Lk+l) \quad 2.25
\end{aligned}$$

$$\text{เมื่อ } \psi(Lk+1) = \frac{\partial y(Lk+1)}{\partial \alpha_{1,l}(k)} \quad 2.26$$

และ  $\mu > 0$  เป็น Step size parameter ดังนั้นสามารถหาค่า  $\psi(k)$  และ  $\Psi(z)$  ได้ดังสมการ

$$\psi(k) = [\psi(Lk) \psi(Lk+1) \dots \psi(Lk+L-1)]^l \quad 2.27$$

$$\Psi(z) = [\Psi_0(z) \Psi_1(z) \dots \Psi_{L-1}(z)]^l \quad 2.28$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กำหนดให้  $\Psi(z)$  มีค่าเป็น

$$\Psi(z) = \frac{1}{D(z)} \left\{ \frac{1-\alpha_0}{2} \frac{\partial R(z)}{\partial \alpha_1(k)} X(z) + \frac{\partial \alpha_1}{\partial \alpha_1(k)} Y(z) z^{-1} \right\} \quad 2.29$$

ค่า  $\psi(k)$  แต่ละตัวในสมการที่ 2.22 จะเป็นสัญญาณควบคุมอะแดปทีฟ ซึ่งเป็นโครงสร้างแบบสเกลาร์ ในสมการที่ 2.24 เป็นค่าทรานสเฟอร์ฟังก์ชันของฟิลเตอร์แต่ละส่วน ซึ่งมี  $\Psi_1(z)$  แสดงถึง  $G_1(z)$  เราเรียกโครงสร้างที่มี  $L \geq 2$  นี้ว่าโครงสร้างแบบขนาน

เพื่อนำอะแดปทีฟอัลกอริทึม มาใช้งานจึงต้องมีการศึกษาค่าพารามิเตอร์แต่ละตัวของ  $H(z)$  โดยแทน  $z = e^{j\omega}$  เมื่อ  $\omega$  มีค่าระหว่าง  $0 < \omega < \pi$  แล้วพล็อตกราฟของ  $|H(\omega)|$  เทียบกับแกน  $\omega$  และพล็อตกราฟของ  $|H(e^{j\omega})|$  เทียบกับ  $\alpha_1(k)$  เพื่อแสดงถึงหน้าที่ของตัวแปรแต่ละตัวใน อะแดปทีฟอัลกอริทึม

$$H(z) = \frac{1-\alpha_0}{2} \cdot \frac{1-z^{-2}}{1-\alpha_1(k) \cdot (1+\alpha_0) \cdot z^{-1} + \alpha_0 \cdot z^{-2}}$$

แทน  $z = e^{j\omega}$

$$H(e^{j\omega}) = \frac{1-\alpha_0}{2} \cdot \frac{1-(e^{j\omega})^{-2}}{1-\alpha_1(k) \cdot (1+\alpha_0) \cdot (e^{j\omega})^{-1} + \alpha_0 \cdot (e^{j\omega})^{-2}} \quad 2.30$$

คูณพจน์บนและล่างด้วย  $e^{j\omega}$  ทั้งเศษและส่วน

$$H(e^{j\omega}) = \frac{1-\alpha_0}{2} \cdot \frac{e^{j\omega} - e^{-j\omega}}{e^{j\omega} - \alpha_1(k) \cdot (1+\alpha_0) + \alpha_0 \cdot e^{j\omega}} \quad 2.31$$

จากกฎของ euler  $e^{\pm j\theta} = \cos(\theta) \pm j \sin(\theta)$

$$H(e^{j\omega}) = \frac{1-\alpha_0}{2} \cdot \frac{\cos(\omega) + j \sin(\omega) - \cos(\omega) + j \sin(\omega)}{\cos(\omega) + j \sin(\omega) - \alpha_1(k) \cdot (1+\alpha_0) + \alpha_0 \cdot \cos(\omega) - j \cdot \alpha_0 \cdot \sin(\omega)} \quad 2.32$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปรับสมการใหม่

$$H(e^{j\omega}) = \frac{j \sin(\omega)}{\{\cos(\omega) - \alpha_1(k)\} \left( \frac{1 + \alpha_0}{1 - \alpha_0} \right) + j \sin(\omega)} \quad 2.33$$

พบว่าค่าของ  $\alpha_0$  (มีค่าระหว่าง  $0 < \alpha_0 < 1$ ) จะมีผลต่อการเปลี่ยนแปลงของ Q-Factor โดยมีความสัมพันธ์ที่แปรผกผันตรงต่อกัน คือเมื่อค่า  $\alpha_0$  เข้าใกล้ศูนย์แล้ว ค่าของ Q-factor มีน้อยลง (กราฟของ  $|H(e^{j\omega})|$  จะมีความกว้างมาก) ซึ่งจะทำให้วงจรแบนด์พาสฟิลเตอร์ในอะแดปทีฟอัลกอริทึม มีช่วงความถี่ที่ไม่ต้องการเข้าไปด้วย และเมื่อค่าของ  $\alpha_0$  มีค่าเข้าใกล้หนึ่งแล้ว ค่าของ Q-factor จะมีค่าสูงขึ้น (กราฟของ  $|H(e^{j\omega})|$  มีความกว้างน้อยลง) ซึ่งจะทำให้วงจรแบนด์พาสฟิลเตอร์ในอะแดปทีฟอัลกอริทึม กรองเฉพาะช่วงความถี่ที่ต้องการผ่านซึ่งจะมีสัญญาณข้างเคียงน้อย

ค่าของ  $\alpha_1$  จะมีผลทำให้ความถี่กลาง  $\omega_0$  เบี่ยงเบนไป โดยที่ค่าของ  $\alpha_1 = 0$  จะทำให้  $\omega_0$  อยู่ตรงกลางพอดี และ  $\alpha_1$  เป็นลบ ความถี่  $\omega_0$  จะเบี่ยงไปทางซ้าย และ  $\alpha_1$  เป็นบวกความถี่  $\omega_0$  จะเบี่ยงไปทางขวา

### 2.3 การออกแบบอะแดปทีฟอัลกอริทึมแบบขนานกรณี $L=1$

$$Y(z) = H(z)X(z)$$

แทนค่า  $H(z)$  ลงในสมการ 2.14

$$Y(z) = \left( \frac{1 - \alpha_0}{2} \cdot \frac{1 - z^{-2}}{1 - \alpha_1(k) \cdot (1 - \alpha_0)z^{-1} - \alpha_0 z^{-2}} \right) \cdot X(z)$$

$$Y(z) \cdot [1 - \alpha_1(k) \cdot (1 + \alpha_0) \cdot z^{-1} + \alpha_0 \cdot z^{-2}] = \left( \frac{1 - \alpha_0}{2} \right) \cdot (1 - z^{-2}) \cdot X(z)$$

$$Y(z) - \alpha_1(k) \cdot (1 + \alpha_0) \cdot Y(z) \cdot z^{-1} + \alpha_0 \cdot Y(z) \cdot z^{-2} = \left( \frac{1 - \alpha_0}{2} \right) \cdot (X(z) - X(z) \cdot z^{-2}) \quad 2.34$$

Take Inverse Z-Transform

$$y(k) - \alpha_1(k) \cdot (1 + \alpha_0) \cdot y(k-1) + \alpha_0 \cdot y(k-2) = \left( \frac{1 - \alpha_0}{2} \right) \cdot [x(k) - x(k-2)]$$

จะได้

$$y(k) = \left( \frac{1 - \alpha_0}{2} \right) \cdot [x(k) - x(k-2)] + \alpha_1(k) \cdot (1 + \alpha_0) \cdot y(k-1) - \alpha_0 \cdot y(k-2) \quad 2.35$$

เมื่อ  $\psi(k) = \frac{\partial y(k)}{\partial \alpha_1(k)}$  เป็นสัญญาณควบคุมอะแดปทีฟของ  $\alpha_1(k)$

$\mu_1$  เป็น step size parameter ถ้าให้ค่านี้นี้เล็กพอจะทำให้การเปลี่ยนแปลงของ  $y(k)$  ในแต่ละครั้งมีค่าน้อย เราสามารถประมาณสัญญาณควบคุมอะแดปทีฟได้ดังนี้

$$\psi(k-i) = \frac{\partial y(k-i)}{\partial \alpha_1(k-i)} \approx \frac{\partial y(k-i)}{\partial \alpha_1(k)}$$

เมื่อ  $i$  เป็นจำนวนเต็มบวกใดๆ

เพราะฉะนั้นเราสามารถหาค่า  $\psi(k) = \frac{\partial y(k)}{\partial \alpha_1(k)}$  หรือใช้สมการที่ 2.26

จะได้

$$\psi(k) = \alpha_1(k) \cdot (1 + \alpha_0) \cdot \psi(k-1) - \alpha_0 \psi(k-2) + (1 + \alpha_0) \cdot y(k-1) \quad 2.36$$

จากนั้นหาค่าสัญญาณควบคุมอะแดปทีฟอัลกอริทึม โดยใช้สมการที่ 2.25

จะได้

$$\alpha_{1,0}(k+1) = \alpha_{1,0}(k) + \mu \cdot y(L) \psi(L) \quad 2.37$$

พิจารณาความเร็วในการเปลี่ยนแปลงของ  $\alpha_1(k)$  โดยการพล็อตกราฟ  $\alpha_1(k)$  เทียบกับ แกนเวลา  $k$  จะพบว่า

- ความละเอียดของค่า  $\mu$  จะส่งผลต่อการเกิดนอยส์ ถ้าค่านี้นี้มีความละเอียดมากน้อยดี จะลดลง แต่การเปลี่ยนแปลง ก็จะช้าลงด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ความเร็วเชิงมุม  $\omega_0$  ที่มีค่ามากจะทำให้การเปลี่ยนแปลง เร็วกว่าค่า  $\omega_0$  ที่มีค่าน้อย
- ความสูงของสัญญาณอินพุตที่เข้ามาก็มีผลต่อการเปลี่ยนแปลง โดยที่ถ้าความแรงของสัญญาณอินพุตมีมากก็จะทำให้การ เร็วขึ้นด้วย
- $\alpha_0$  ค่ามาก (เข้าใกล้หนึ่ง) จะทำให้มีการแกว่งของสัญญาณ แต่การจะเร็วเมื่อเข้าใกล้สถานะคงที่ และเมื่อค่า  $\alpha_0$  ค่าน้อย (เข้าใกล้ศูนย์) การเปลี่ยนแปลงในช่วงเริ่มต้นเร็ว และแทบไม่มีการแกว่งของสัญญาณ

## 2.4 การออกแบบอะแดปทีฟอัลกอริทึมแบบขนานกรณี $L=2$

สมการทรานสเฟอร์ฟังก์ชันกรณี  $L=2$

$$H(z) = \frac{1 - \alpha_0}{2} \cdot \frac{R(z)}{D(z)} \quad 2.38$$

ซึ่งสามารถหาค่า  $D(z)$  ได้จากสมการที่ 2.16 โดยการแทนค่า  $\alpha_1$  และ  $\alpha_2$  ลงในสมการที่ 2.19 และ 2.20

โดยที่

$$D(z) = 1 - a_1 z^{-1} + a_2 z^{-2}$$

จะได้

$$D(z) = 1 - ((\alpha_1(k)(1 + \alpha_0))^2 - 2\alpha_0)Z^{-1} + \alpha_0^2 Z^{-2} \quad 2.39$$

จากนั้นแทนค่า  $L = 2$  ลงในสมการที่ 2.17 เพื่อหาค่า  $R(z)$

$$R(z) = \begin{bmatrix} r_0(z) & r_1(z) \\ r_{-1}(z) & r_0(z) \end{bmatrix}$$

$$r_0(z) = 1 - (1 - (1 - \alpha_0)Z^{-1} - \alpha_0 Z^{-2})$$

$$r_1(z) = \alpha_1(k)(1 + \alpha_0)Z^{-1} - \alpha_1(k)(1 + \alpha_0)Z^{-2}$$

$$r_{-1}(z) = \alpha_1(k)(1 + \alpha_0) - \alpha_1(k)(1 + \alpha_0)Z^{-1}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นำค่าที่ได้แทนในสมการ 2.33 จะได้  $H(z)$

เพื่อหาการเปลี่ยนแปลงของ  $\alpha_1(k)$  เมื่อมีสัญญาณอินพุตป้อนเข้ามา จึงต้องหาเอาต์พุต  $y(k)$  เมื่อ  $Y(z)$  เป็น Z-transform ของ  $y(k)$  ความสัมพันธ์ของทรานสเฟอร์ฟังก์ชันจะเป็น

$$Y(z) = H(z)H(z)$$

$$Y(z) = \frac{1-\alpha_0}{2} \left[ \frac{R(z)}{D(z)} \right] X(z)$$

ที่  $L = 2$  จะได้

$$\begin{bmatrix} Y_0(z) \\ Y_1(z) \end{bmatrix} = \frac{1-\alpha_0}{2D(z)} \begin{bmatrix} r_0(z) & r_1(z) \\ r_{-1}(z) & r_0(z) \end{bmatrix} \begin{bmatrix} X_0(z) \\ X_1(z) \end{bmatrix} \quad 2.40$$

จะได้

$$Y_0(z) = \frac{1-\alpha_0}{2D(z)} [r_0(z)X_0(z) + r_1(z)X_1(z)] \quad 2.41$$

$$Y_1(z) = \frac{1-\alpha_0}{2D(z)} [r_{-1}(z)X_0(z) + r_0(z)X_1(z)] \quad 2.42$$

แทนค่า  $r_s(z)$  และ  $D(z)$  ลงในสมการ 2.41 จะได้

$$Y_0(z) = \frac{1-\alpha_0}{2D(z)} [X_0(z) - (1-\alpha_0)X_0(z)Z^{-1} - \alpha_0 X_0(z)Z^{-2} + \alpha_1(k)(1+\alpha_0)X_1(z)Z^{-1} - \alpha_1(k)(1+\alpha_0)X_1(z)Z^{-2}]$$

$$Y_0(z) - \alpha_1^2(k)(1+\alpha_0^2)Y_0(z)Z^{-1} - 2\alpha_0 Y_0(z)Z^{-1} + \alpha_0^2 Y_0(z)Z^{-2}$$

$$= \frac{1-\alpha_0}{2} \left[ X_0(z) - (1-\alpha_0)X_0(z)Z^{-1} - \alpha_0 X_0(z)Z^{-2} + \alpha_1(k)(1+\alpha_0)X_1(z)Z^{-1} - \alpha_1(k)(1+\alpha_0)X_1(z)Z^{-2} \right]$$

Take Inverse Z-Transform

$$\begin{aligned} & y(2k) - \alpha_1^2(k)(1 + \alpha_0^2)y(2k-2) - 2\alpha_0 y(2k-2) + \alpha_0^2 y(2k-4) \\ &= \frac{1 - \alpha_0}{2} \left[ x(2k) - (1 - \alpha_0)x(2k-2) - \alpha_0 x(2k-4) + \alpha_1(k)(1 + \alpha_0)x(2k-1) \right] \\ & \quad \left[ -\alpha_1(k)(1 + \alpha_0)x(2k-3) \right] \end{aligned}$$

ให้  $A = \alpha_1(k)(1 + \alpha_0)$

จะได้

$$\begin{aligned} y(2k) &= \frac{1 - \alpha_0}{2} [x(2k) + Ax(2k-1) - (1 - \alpha_0)x(2k-2) - Ax(2k-3) - \alpha_0 x(2k-4)] \\ & \quad + (A^2 + 2\alpha_0)y(2k-2) - \alpha_0^2 y(2k-4) \end{aligned} \quad 2.43$$

แทนค่า  $r_s(z)$  และ  $D(z)$  ลงในสมการ 2.42 จะได้

$$Y_1(z) = \frac{1 - \alpha_0}{2D(z)} \left[ \alpha_1(k)(1 + \alpha_0)X_0(z) - \alpha_1(k)(1 + \alpha_0)X_0(z)Z^{-1} + X_1(z) \right] \left[ -(1 - \alpha_0)X_1(z)Z^{-1} - \alpha_0 X_1(z)Z^{-2} \right]$$

$$\begin{aligned} & Y_1(z) - \alpha_1^2(k)(1 + \alpha_0^2)Y_1(z)Z^{-1} - 2\alpha_0 Y_1(z)Z^{-1} + \alpha_0^2 Y_1(z)Z^{-2} \\ &= \frac{1 - \alpha_0}{2D(z)} \left[ \alpha_1(k)(1 + \alpha_0)X_0(z) - \alpha_1(k)(1 + \alpha_0)X_0(z)Z^{-1} + X_1(z) \right] \left[ -(1 - \alpha_0)X_1(z)Z^{-1} - \alpha_0 X_1(z)Z^{-2} \right] \end{aligned}$$

Take Inverse Z-Transform

$$\begin{aligned} & y(2k+1) - A^2 y(2k-1) - 2\alpha_0 y(2k-1) + \alpha_0^2 y(2k-3) \\ &= \frac{1 - \alpha_0}{2} [Ax(2k) - Ax(2k-2) + x(2k+1) - (1 - \alpha_0)x(2k-1) - \alpha_0 x(2k-3)] \\ & y(2k+1) = \frac{1 - \alpha_0}{2} [Ax(2k) + x(2k+1) - (1 - \alpha_0)x(2k-1) - Ax(2k-2) - \alpha_0 x(2k-3)] \\ & \quad + (A^2 + 2\alpha_0)y(2k-1) - \alpha_0^2 y(2k-3) \end{aligned} \quad 2.44$$

จากสมการ 2.25 จะได้อะแดปทีฟอัลกอริทึม สำหรับ  $\alpha_1(k)$  ดังนี้

$$\alpha_{1,0}(k+1) = \alpha_{1,0}(k) + \mu \cdot y(2k)\psi(2k) \quad 2.45$$

$$\alpha_{1,1}(k+1) = \alpha_{1,1}(k) + \mu \cdot y(2k+1)\psi(2k+1) \quad 2.46$$

ศึกษาความคุมอะแดปทีฟอัลกอริทึม  $\psi(k)$  ในกรณีที่  $L=2$  หาได้โดยการแทนค่าลงในสมการ 2.29 จะได้

$$\begin{bmatrix} \Psi_0(z) \\ \Psi_1(z) \end{bmatrix} = \frac{1}{D(z)} \left\{ \frac{1-\alpha_0}{2} \cdot \frac{\partial}{\partial \alpha_1(k)} \begin{bmatrix} r_0(z) & r_1(z) \\ r_{-1}(z) & r_0(z) \end{bmatrix} \begin{bmatrix} X_0(z) \\ X_1(z) \end{bmatrix} + \frac{\partial a_1}{\partial \alpha_1(k)} \begin{bmatrix} Y_0(z)Z^{-1} \\ Y_1(z)Z^{-1} \end{bmatrix} \right\} \quad 2.47$$

$$\Psi_0(z) = \frac{1}{D(z)} \left\{ \frac{1-\alpha_0}{2} \cdot \frac{\partial}{\partial \alpha_1(k)} [r_0(z)X_0(z) + r_1(z)X_1(z)] + \frac{\partial a_1}{\partial \alpha_1(k)} [Y_0(z)Z^{-1}] \right\} \quad 2.48$$

$$\Psi_1(z) = \frac{1}{D(z)} \left\{ \frac{1-\alpha_0}{2} \cdot \frac{\partial}{\partial \alpha_1(k)} [r_{-1}(z)X_0(z) + r_0(z)X_1(z)] + \frac{\partial a_1}{\partial \alpha_1(k)} [Y_1(z)Z^{-1}] \right\} \quad 2.49$$

$$\frac{\partial r_0(z)}{\partial \alpha_1(k)} = 0$$

$$\frac{\partial r_1(z)}{\partial \alpha_1(k)} = (1+\alpha_0)Z^{-1} - (1+\alpha_0)Z^{-2}$$

$$\frac{\partial r_{-1}(z)}{\partial \alpha_1(k)} = (1+\alpha_0) - (1+\alpha_0)Z^{-1}$$

$$\frac{\partial a_1}{\partial \alpha_1(k)} = 2(\alpha_1(k)(1+\alpha_0)(1+\alpha_0))$$

แทนค่าในสมการ 2.48 จะได้

$$\Psi_0(z) = \frac{1}{D(z)} \left\{ \frac{1-\alpha_0}{2} [(1+\alpha_0)X_1(z)Z^{-1} - (1+\alpha_0)X_1(z)Z^{-2}] + 2A(1+\alpha_0)Y_0(z)Z^{-1} \right\}$$

$$\Psi_0(z) - A^2\Psi_0(z)Z^{-1} - 2\alpha_0\Psi_0(z)Z^{-1} + \alpha_0^2\Psi_0(z)Z^{-2}$$

$$= \frac{1}{D(z)} \left\{ \frac{1-\alpha_0}{2} [(1+\alpha_0)X_1(z)Z^{-1} - (1+\alpha_0)X_1(z)Z^{-2}] + 2A(1+\alpha_0)Y_0(z)Z^{-1} \right\}$$

Take Inverse Z-Transform

$$\begin{aligned} \psi(2k) - A^2\psi(2k-2) - 2\alpha_0\psi(2k-2) + \alpha_0^2\psi(2k-4) \\ = \frac{1-\alpha_0}{2} [(1+\alpha_0)x(2k-1) - (1+\alpha_0)x(2k-2) + 2A(1+\alpha_0)y(2k-2)] \end{aligned}$$

จะได้

$$\begin{aligned} \psi(2k) = \frac{1-\alpha_0}{2} [(1+\alpha_0)x(2k-1) - (1+\alpha_0)x(2k-3)] + 2A(1+\alpha_0)y(2k-2) \\ + (A^2 + 2\alpha_0)\psi(2k-2) - \alpha_0^2\psi(2k-4) \end{aligned} \quad 2.50$$

แทนค่าในสมการ 2.49 จะได้

$$\begin{aligned} \Psi_1(z) = \frac{1}{D(z)} \left\{ \frac{1-\alpha_0}{2} [(1+\alpha_0)X_0(z) - (1+\alpha_0)X_0(z)Z^{-1}] + 2A(1+\alpha_0)Y_1(z)Z^{-1} \right\} \\ \Psi_1(z) - A^2\Psi_1(z)Z^{-1} - 2\alpha_0\Psi_1(z)Z^{-1} + \alpha_0^2\Psi_1(z)Z^{-2} \\ = \frac{1-\alpha_0}{2} \left\{ [(1+\alpha_0)X_0(z) - (1+\alpha_0)X_0(z)Z^{-1}] + 2A(1+\alpha_0)Y_1(z)Z^{-1} \right\} \end{aligned}$$

Take Inverse Z-Transform

$$\begin{aligned} \psi(2k+1) - (A^2 + 2\alpha_0)\psi(2k-1) + \alpha_0^2\psi(2k-3) \\ = \frac{1-\alpha_0}{2} [(1+\alpha_0)x(2k) - (1+\alpha_0)x(2k-2)] + 2A(1+\alpha_0)y(2k-1) \end{aligned}$$

จะได้

$$\begin{aligned} \psi(2k+1) = \frac{1-\alpha_0}{2} [(1+\alpha_0)x(2k) - (1+\alpha_0)x(2k-2)] + 2A(1+\alpha_0)y(2k-1) \\ + (A^2 + 2\alpha_0)\psi(2k-1) - \alpha_0^2\psi(2k-3) \end{aligned} \quad 2.51$$

## 2.5 การออกแบบอะแดปทีฟอัลกอริทึมแบบขนานกรณี $L=3$

สมการทรานสเฟอร์ฟังก์ชันกรณี  $L = 3$

$$H(z) = \frac{1 - \alpha_0}{2} \cdot \frac{R(z)}{D(z)} \quad 2.52$$

ซึ่งสามารถหาค่า  $D(z)$  ได้จากสมการ 2.16 โดยการแทนค่า  $a_1$  และ  $a_2$  ลงในสมการที่ 2.19 และ 2.20 โดยที่

$$D(z) = 1 - a_1 z^{-1} + a_2 z^{-2}$$

จะได้

$$D(z) = 1 - \alpha_1(k)(1 + \alpha_0) \cdot [(\alpha_1(k)(1 + \alpha_0))^2 - 3\alpha_0] Z^{-1} - \alpha_0^3 Z^{-2} \quad 2.53$$

จากนั้นแทนค่า  $L = 2$  ลงในสมการที่ 2.12 เพื่อหาค่า  $R(z)$

$$R(z) = \begin{bmatrix} r_0(z) & r_1(z) & r_2(z) \\ r_{-1}(z) & r_0(z) & r_1(z) \\ r_{-2}(z) & r_{-1}(z) & r_0(z) \end{bmatrix}$$

$$r_0(z) = 1 - [(1 - \alpha_0)(\alpha_1(k)(1 + \alpha_0))] Z^{-1} - \alpha_0^2 Z^{-2}$$

$$r_1(z) = [(\alpha_1(k)(1 + \alpha_0))^2 - \alpha_0 - 1] Z^{-1} - [\alpha_1(k)(1 + \alpha_0)\alpha_0] Z^{-2}$$

$$r_2(z) = [\alpha_1(k)(1 + \alpha_0)] Z^{-1} + [\alpha_0^2 + \alpha_0 - (\alpha_1(k)(1 + \alpha_0))^2] Z^{-2}$$

$$r_{-1}(z) = \alpha_1(k)(1 + \alpha_0) + [\alpha_0^2 + \alpha_0 - (\alpha_1(k)(1 + \alpha_0))^2] Z^{-1}$$

$$r_{-2}(z) = (\alpha_1(k)(1 + \alpha_0))^2 - \alpha_0 - 1 - [(\alpha_1(k)(1 + \alpha_0))\alpha_0] Z^{-1}$$

นำค่าที่ได้แทนในสมการ 2.52 จะได้  $H(z)$

เพื่อหาค่าการเปลี่ยนแปลงของ  $\alpha_1(k)$  เมื่อมีสัญญาณ อินพุตป้อนเข้ามา จึงต้องหาเอาต์พุต  $y(k)$  เมื่อ  $Y(z)$  เป็น Z-Transform ของ  $y(k)$  ความสัมพันธ์ของทรานสเฟอร์ฟังก์ชัน จะเป็น

$$Y(z) = H(z)X(z)$$

$$Y(z) = \frac{1-\alpha_0}{2} \left[ \frac{R(z)}{D(z)} \right] X(z)$$

ที่  $L = 3$  จะได้

$$\begin{bmatrix} Y_0(z) \\ Y_1(z) \\ Y_2(z) \end{bmatrix} = \frac{1-\alpha_0}{2D(z)} \begin{bmatrix} r_0(z) & r_1(z) & r_2(z) \\ r_{-1}(z) & r_0(z) & r_1(z) \\ r_{-2}(z) & r_{-1}(z) & r_0(z) \end{bmatrix} \begin{bmatrix} X_0(z) \\ X_1(z) \\ X_2(z) \end{bmatrix} \quad 2.54$$

$$Y_0(z) = \frac{1-\alpha_0}{2D(z)} [r_0(z)X(z) \quad r_1(z)X_1(z) \quad r_2(z)X_2(z)] \quad 2.55$$

$$Y_1(z) = \frac{1-\alpha_0}{2D(z)} [r_{-1}(z)X_0(z) \quad r_0(z)X_1(z) \quad r_1(z)X_2(z)] \quad 2.56$$

$$Y_2(z) = \frac{1-\alpha_0}{2D(z)} [r_{-2}(z)X_0(z) \quad r_{-1}(z)X_1(z) \quad r_0(z)X_2(z)] \quad 2.57$$

จากสมการ 2.55 จะได้

กำหนดให้  $A = \alpha_1(k)(1 + \alpha_0)$

$$Y_0(z) = \frac{1-\alpha_0}{2D(z)} \left[ \begin{array}{l} X_0(z) - (1-\alpha_0) \cdot AX_0(z)Z^{-1} - \alpha_0^2 X_0(z)Z^{-2} + (A^2 - \alpha_0 - 1)X_1(z)Z^{-1} \\ - A\alpha_0 X_1(z)Z^{-2} + AX_2(z)Z^{-1} + (\alpha_0^2 + \alpha_0 - A^2)X_2(z)Z^{-2} \end{array} \right]$$

$$Y_0(z) - A^3 Y_0(z)Z^{-1} + 3A\alpha_0 Y_0(z)Z^{-1} + \alpha_0^3 Y_0(z)Z^{-2}$$

$$= \frac{1-\alpha_0}{2} \left[ \begin{array}{l} X_0(z) - (1-\alpha_0) \cdot AX_0(z)Z^{-1} - \alpha_0^2 X_0(z)Z^{-2} + (A^2 - \alpha_0 - 1)X_1(z)Z^{-1} \\ - A\alpha_0 X_1(z)Z^{-2} + AX_2(z)Z^{-1} + (\alpha_0^2 + \alpha_0 - A^2)X_2(z)Z^{-2} \end{array} \right]$$

Take Inverse Z-Transform

$$y(3k) - A^3 y(3k-3) + 3A\alpha_0 y(3k-3) + \alpha_0^3 y(3k-6)$$

$$= \frac{1-\alpha_0}{2} \left[ \begin{array}{l} x(3k) - (1-\alpha_0)Ax(3k-3) - \alpha_0^2 x(3k-6) + (A^2 - \alpha_0 - 1)x(3k-2) - A\alpha_0 x(3k-5) \\ + Ax(3k-1) + (\alpha_0^2 + \alpha_0 - A^2)x(3k-4) \end{array} \right]$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะได้

$$y(3k) = \frac{1-\alpha_0}{2} \left[ \begin{aligned} &x(3k) + Ax(3k-1) + (A^2 - \alpha_0 - 1)x(3k-2) - (1-\alpha_0)Ax(3k-3) \\ &+ (\alpha_0^2 + \alpha_0 - A^2)x(3k-4) - A\alpha_0x(3k-5) - \alpha_0^2x(3k-6) \\ &+ (A^3 - 3A\alpha_0)y(3k-3) - \alpha_0^3y(3k-6) \end{aligned} \right] \quad 2.58$$

จากสมการ 2.56 จะได้

$$Y_1(z) = \frac{1-\alpha_0}{2D(z)} \left[ \begin{aligned} &AX_0(z) + (\alpha_0^2 + \alpha_0 - A^2)X_0(z)Z^{-1} + X_1(z) - (1-\alpha_0)AX_1(z)Z^{-1} \\ &- \alpha_0^2X_1(z)Z^{-2} + (A^2 - \alpha_0 - 1)X_2(z)Z^{-1} - A\alpha_0X_2(z)Z^{-2} \end{aligned} \right]$$

$$\begin{aligned} Y_1(z) - A^3Y_1(z)Z^{-1} + 3A\alpha_0Y_1(z)Z^{-2} + \alpha_0^3Y_1(z)Z^{-2} \\ = \frac{1-\alpha_0}{2} \left[ \begin{aligned} &AX_0(z) + (\alpha_0^2 + \alpha_0 - A^2)X_0(z)Z^{-1} + X_1(z) - (1-\alpha_0)AX_1(z)Z^{-1} \\ &- \alpha_0^2X_1(z)Z^{-2} + (A^2 - \alpha_0 - 1)X_2(z)Z^{-1} - A\alpha_0X_2(z)Z^{-2} \end{aligned} \right] \end{aligned}$$

Take Inverse Z-Transform

$$\begin{aligned} y(3k+1) - A^3y(3k-2) + 3A\alpha_0y(3k-2) + \alpha_0^3y(3k-5) \\ = \frac{1-\alpha_0}{2} \left[ \begin{aligned} &Ax(3k) + (\alpha_0^2 + \alpha_0 - A^2)x(3k-3) + x(3k+1) - (1-\alpha_0)Ax(3k-2) - \alpha_0^2x(3k-5) \\ &+ (A^2 - \alpha_0 - 1)x(3k-1) - A\alpha_0x(3k-4) \end{aligned} \right] \end{aligned}$$

จะได้

$$\begin{aligned} y(3k+1) = \frac{1-\alpha_0}{2} \left[ \begin{aligned} &Ax(3k) + x(3k+1) + (A^2 - \alpha_0 - 1)x(3k-1) - (1-\alpha_0)Ax(3k-2) \\ &+ (\alpha_0^2 + \alpha_0 - A^2)x(3k-3) - A\alpha_0x(3k-4) - \alpha_0^2x(3k-5) \end{aligned} \right] \\ + (A^3 - 3A\alpha_0)y(3k-2) - \alpha_0^3y(3k-5) \quad 2.59 \end{aligned}$$

จากสมการ 2.57 จะได้ว่า

$$Y_2(z) = \frac{1-\alpha_0}{2D(z)} \left[ (A^2 - \alpha_0 - 1)X_0(z) - A\alpha_0 X_0(z)Z^{-1} + AX_1(z) + (\alpha_0^2 + \alpha_0 - A^2)X_1(z)Z^{-1} \right. \\ \left. + X_2(z) - (1-\alpha_0)AX_2(z)Z^{-1} - \alpha_0^2 X_2(z)Z^{-2} \right]$$

$$Y_2(z) - A^3 Y_2(z)Z^{-1} + 3A\alpha_0 Y_2(z)Z^{-1} + \alpha_0^3 Y_2(z)Z^{-2} \\ = \frac{1-\alpha_0}{2D(z)} \left[ (A^2 - \alpha_0 - 1)X_0(z) - A\alpha_0 X_0(z)Z^{-1} + AX_1(z) + (\alpha_0^2 + \alpha_0 - A^2)X_1(z)Z^{-1} \right. \\ \left. + X_2(z) - (1-\alpha_0)AX_2(z)Z^{-1} - \alpha_0^2 X_2(z)Z^{-2} \right]$$

Take Inverse Z-Transform

$$y(3k+2) - A^3 y(3k-1) + 3\alpha_0 A y(3k-1) + \alpha_0^3 y(3k-4) \\ = \frac{1-\alpha_0}{2} \left[ (A^2 - \alpha_0 - 1)x(3k) - A\alpha_0 x(3k-3) + Ax(3k+1) + (\alpha_0^2 + \alpha_0 - A^2)x(3k-2) \right. \\ \left. + x(3k+2) - (1-\alpha_0)Ax(3k-1) - \alpha_0^2 x(3k-4) \right]$$

$$y(3k+2) = \frac{1-\alpha_0}{2} \left[ (A^2 - \alpha_0 - 1)x(3k) + Ax(3k+1) + x(3k+2) - (1-\alpha_0)Ax(3k-1) \right. \\ \left. + (\alpha_0^2 + \alpha_0 - A^2)x(3k-2) - A\alpha_0 x(3k-3) - \alpha_0^2 x(3k-4) \right. \\ \left. + (A^3 - 3A\alpha_0)y(3k-1) - \alpha_0^3 y(3k-4) \right] \quad 2.60$$

จากสมการ 2.20 จะได้ว่าอะแดปทีฟอัลกอริทึมสำหรับ  $\alpha_1(k)$  ดังนี้

$$\alpha_{1,0}(k+1) = \alpha_{1,0}(k) + \mu \cdot y(3k)\psi(3k) \quad 2.61$$

$$\alpha_{1,1}(k+1) = \alpha_{1,1}(k) + \mu \cdot y(3k+1)\psi(3k+1) \quad 2.62$$

$$\alpha_{1,2}(k+1) = \alpha_{1,2}(k) + \mu \cdot y(3k+2)\psi(3k+2) \quad 2.63$$

สัญญาณควบคุมอะแดปทีฟอัลกอริทึม  $\psi(k)$  ในกรณีที่  $L = 3$  หาได้โดยการแทนค่าลงในสมการ 2.24 จะได้ว่า

$$\begin{bmatrix} \psi_0(z) \\ \psi_1(z) \\ \psi_{20}(z) \end{bmatrix} = \frac{1}{D(z)} \cdot \left\{ \frac{1-\alpha_0}{2} \cdot \frac{\partial}{\partial \alpha_1(k)} \begin{bmatrix} r_0(z) & r_1(z) & r_2(z) \\ r_{-1}(z) & r_0(z) & r_1(z) \\ r_{-2}(z) & r_{-1}(z) & r_0(z) \end{bmatrix} \begin{bmatrix} X_0(z) \\ X_1(z) \\ X_2(z) \end{bmatrix} + \frac{\partial \alpha_1}{\partial \alpha_1(k)} \begin{bmatrix} Y_0(z)Z^{-1} \\ Y_1(z)Z^{-1} \\ Y_2(z)Z^{-1} \end{bmatrix} \right\}$$

2.64

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\psi_0(z) = \frac{1}{D(z)} \left\{ \frac{1-\alpha_0}{2} \cdot \frac{\partial}{\partial \alpha_1(k)} [r_0(z)X_0(z) + r_1(z)X_1(z) + r_2(z)X_2(z)] + \frac{\partial \alpha_1}{\partial \alpha_1(k)} [Y_0(z)Z^{-1}] \right\}$$

2.65

$$\psi_1(z) = \frac{1}{D(z)} \left\{ \frac{1-\alpha_0}{2} \cdot \frac{\partial}{\partial \alpha_1(k)} [r_{-1}(z)X_0(z) + r_0(z)X_1(z) + r_1(z)X_2(z)] + \frac{\partial \alpha_1}{\partial \alpha_1(k)} [Y_1(z)Z^{-1}] \right\}$$

2.66

$$\psi_2(z) = \frac{1}{D(z)} \left\{ \frac{1-\alpha_0}{2} \cdot \frac{\partial}{\partial \alpha_1(k)} [r_{-2}(z)X_0(z) + r_{-1}(z)X_1(z) + r_{02}(z)X_2(z)] + \frac{\partial \alpha_1}{\partial \alpha_1(k)} [Y_2(z)Z^{-1}] \right\}$$

2.67

$$\frac{\partial r_0(z)}{\partial \alpha_1(k)} = (1-\alpha_0)^2 Z^{-1}$$

$$\frac{\partial r_1(z)}{\partial \alpha_1(k)} = 2A(1+\alpha_0)Z^{-1} - (1+\alpha_0)\alpha_0 Z^{-2}$$

$$\frac{\partial r_2(z)}{\partial \alpha_1(k)} = (1+\alpha_0)Z^{-1} - 2\alpha_1(k)(1+\alpha_0)(1+\alpha_0)Z^{-2}$$

$$\frac{\partial r_{-1}(z)}{\partial \alpha_1(k)} = (1+\alpha_0) - 2A(1+\alpha_0)Z^{-1}$$

$$\frac{\partial r_{-2}(z)}{\partial \alpha_1(k)} = 2A(1+\alpha_0) - (1+\alpha_0)\alpha_0 Z^{-1}$$

$$\frac{\partial r_0(z)}{\partial \alpha_1(k)} = 3A^2(1+\alpha_0) - 3\alpha_0(1+\alpha_0)$$

จากสมการ 2.64 จะได้ว่า

$$\Psi_0(z) = \frac{1}{D(z)} \left\{ \frac{1-\alpha_0}{2} \begin{bmatrix} (1-\alpha_0)^2 X_0(z)Z^{-1} + 2A(1+\alpha_0)X_1(z)Z^{-1} \\ -(1+\alpha_0)\alpha_0 X_1(z)Z^{-2} + (1+\alpha_0)X_2(z)Z^{-1} \\ -2A(1+\alpha_0)X_2(z)Z^{-2} \\ -3\alpha_0(1+\alpha_0)Y_0(z)Z^{-1} \end{bmatrix} + 3A^2(1+\alpha_0)Y_0(z)Z^{-1} \right\}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\begin{aligned} & \Psi_0(z) - A^3 \Psi_0(z) Z^{-1} + 3A\alpha_0 \Psi_0(z) Z^{-1} + \alpha_0^3 \Psi_0(z) Z^{-2} \\ &= \left\{ \frac{1-\alpha_0}{2} \begin{bmatrix} (1-\alpha_0^2)X_0(z)Z^{-1} + 2A(1+\alpha_0)X_1(z)Z^{-1} \\ -(1+\alpha_0)\alpha_0 X_1(z)Z^{-2} + (1+\alpha_0)X_2(z)Z^{-1} \\ -2A(1+\alpha_0)X_2(z)Z^{-2} \end{bmatrix} + 3A^2(1+\alpha_0)Y_0(z)Z^{-1} \right\} \\ & \quad \left\{ -3\alpha_0(1+\alpha_0)Y_0(z)Z^{-1} \right\} \end{aligned}$$

Take Inverse Z-Transform

$$\begin{aligned} \psi(3k) &= \frac{1-\alpha_0}{2} \left[ (1+\alpha_0)x(3k-1) + 2A(1+\alpha_0)x(3k-2) - (1-\alpha_0^2)x(3k-3) \right] \\ & \quad \left[ -2A(1+\alpha_0)x(3k-4) - (1+\alpha_0)\alpha_0 x(3k-5) \right] \\ & \quad + (3A^2(1+\alpha_0) - 3\alpha_0(1+\alpha_0))y(3k-3) + (A^3 + 3A\alpha_0)\psi(3k-3) - \alpha_0^3\psi(3k-6) \end{aligned} \quad 2.68$$

จากสมการ 2.65 จะได้

$$\Psi_1(z) = \frac{1}{D(z)} \left\{ \frac{1-\alpha_0}{2} \begin{bmatrix} ((1+\alpha_0) - 2A(1+\alpha_0)Z^{-1})X_0(z) + ((1-\alpha_0^2)Z^{-1})X_1(z) \\ + (2A(1+\alpha_0)Z^{-1} - (1+\alpha_0)\alpha_0 Z^{-2})X_2(z) \end{bmatrix} \right\} \\ \left\{ + (3A^2(1+\alpha_0) - 3\alpha_0(1+\alpha_0))Y_1(z)Z^{-1} \right\}$$

$$\begin{aligned} & \Psi_1(z) - A^3 \Psi_1(z) Z^{-1} + 3\alpha_0 A \Psi_1(z) Z^{-1} + \alpha_0^3 \Psi_1(z) Z^{-2} \\ &= \left\{ \frac{1-\alpha_0}{2} \begin{bmatrix} ((1+\alpha_0) - 2A(1+\alpha_0)Z^{-1})X_0(z) + ((1-\alpha_0^2)Z^{-1})X_1(z) \\ + (2A(1+\alpha_0)Z^{-1} - (1+\alpha_0)\alpha_0 Z^{-2})X_2(z) \end{bmatrix} \right\} \\ & \quad \left\{ + (3A^2(1+\alpha_0) - 3\alpha_0(1+\alpha_0))Y_1(z)Z^{-1} \right\} \end{aligned}$$

Take Inverse Z-Transform

$$\begin{aligned} \psi(3k+1) = & \frac{1-\alpha_0}{2} \left[ (1+\alpha_0)x(3k) + 2A(1+\alpha_0)x(3k-1) - (1-\alpha_0^2)x(3k-2) - 2A(1+\alpha_0) \right. \\ & \left. \cdot x(3k-3) - 2A(1+\alpha_0)x(3k-3) - (1+\alpha_0)\alpha_0x(3k-4) \right] \\ & + (3A^2(1+\alpha_0) - 3\alpha_0(1+\alpha_0))y(3k-2) + (A^3 - 3A\alpha_0)\psi(3k-2) - \alpha_0^3\psi(3k-5) \end{aligned}$$

2.69

จากสมการที่ 2.66 จะได้ว่า

$$\begin{aligned} \Psi_2(z) = & \frac{1}{D(z)} \left\{ \frac{1-\alpha_0}{2} \left[ (2A(1+\alpha_0) - (1+\alpha_0)\alpha_0Z^{-1})X_0(z) + \right. \right. \\ & \left. \left[ ((1+\alpha_0) - 2A(1+\alpha_0)Z^{-1}) \cdot X_1(z) + (1-\alpha_0^2)X_2(z)Z^{-1} \right] + \right. \\ & \left. (3A^2(1-\alpha_0) - 3\alpha_0(1+\alpha_0))Y_2(z)Z^{-1} \right\} \\ & \Psi_2(z) - A^3\Psi_2(z)Z^{-1} + 3\alpha_0A\Psi_2(z)Z^{-1} + \alpha_0^3\Psi_2(z)Z^{-2} \\ = & \frac{1}{D(z)} \left\{ \frac{1-\alpha_0}{2} \left[ (2A(1+\alpha_0) - (1+\alpha_0)\alpha_0Z^{-1})X_0(z) + \right. \right. \\ & \left. \left[ ((1+\alpha_0) - 2A(1+\alpha_0)Z^{-1}) \cdot X_1(z) + (1-\alpha_0^2)X_2(z)Z^{-1} \right] + \right. \\ & \left. (3A^2(1-\alpha_0) - 3\alpha_0(1+\alpha_0))Y_2(z)Z^{-1} \right\} \end{aligned}$$

Take Inverse Z-Transform

$$\begin{aligned} \psi(3k+2) = & \frac{1-\alpha_0}{2} \left[ 2A(1+\alpha_0)x(3k) + (1+\alpha_0)x(3k+1) + (1-\alpha_0^2)x(3k-1) - 2A(1+\alpha_0) \right. \\ & \left. \cdot x(3k-2) - (1+\alpha_0)\alpha_0x(3k-3) \right] \\ & + (3A^2(1+\alpha_0) - 3\alpha_0(1+\alpha_0))y(3k-1) + (A^3 - 3A\alpha_0)\psi(3k-1) - \alpha_0^3\psi(3k-4) \end{aligned}$$

2.70

## 2.6 ปัญหาที่เกิดขึ้นของอะแดปทีฟอัลกอริทึมแบบขนาน

ปัญหาที่เกิดขึ้นคือ การ converge ของอะแดปทีฟอัลกอริทึมของแต่ละสาขานั้นไม่ถูกต้อง ในลำดับการพิจารณาการเปลี่ยนแปลง นั้นจะถูกกำหนดโดยค่า  $\alpha_1(k)$  ซึ่งเราจะคำนวณได้จาก ผลคูณของ  $y(Lk+1)$  และ  $\psi(Lk+1)$  ดังสมการ

$$m_{\alpha_1}(L, l) = \frac{1}{2K} \sum_{k=0}^{2K-1} y(Lk+1)\psi(Lk+1) \quad 2.71$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

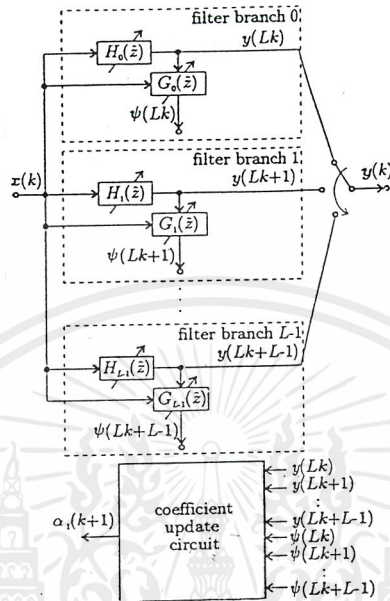
ซึ่ง  $\omega = \frac{\pi}{K}$  โดยที่  $K$  เป็นจำนวนเต็ม ดังรูป 3.25 แสดงถึงค่าเฉลี่ย โดยพิจารณาที่  $\alpha_1(Lk + l)$  ภายใต้เงื่อนไขต่าง ๆ ที่ได้กำหนดไว้

จากนั้นจะพิจารณาคุณสมบัติที่โครงสร้างแบบสเตลาร์กรณิ  $L=1$  ซึ่งจะเห็นได้ว่า การ converge ของ  $L=1$  นั้น รูปกราฟจะมีจุดตัดที่แกน 0 เพียงจุดเดียว นั่นแสดงให้เห็นว่าไม่ว่าจะให้มีค่าเริ่มต้น ( initial value ) ที่ 0 หรือ 0.99 นั้นค่าที่ได้มันจะมีเพียงค่าเดียวคือที่ค่า  $\alpha_1(LK + l) = 0.5$  ดังรูป 3.25

ซึ่งรูป 3.26 ,3.27 และ 3.28 จะแสดงถึงคุณสมบัติของอะแดปทีฟอัลกอริทึมกรณิโครงสร้างแบบขนาน  $L = 3(l = 0,1,2)$  ซึ่งจะเห็นได้ว่าจะมีเพียง  $l = 0$  นั้นมีการเปลี่ยนแปลงทำให้อะแดปทีฟอัลกอริทึม converge ได้ถูกต้อง โดยที่  $l = 1$  และ 2 นั้นมีการ converge ที่ผิดพลาด ซึ่งจะสังเกตได้จากกราฟ 3.27 และ 3.28 นั้นจะมีจุดตัดที่แกน 0 มากกว่า 1 เส้น

## 2.7 อะแดปทีฟอัลกอริทึมโครงสร้างแบบใหม่

เนื่องจากโครงสร้างแบบขนานของอะแดปทีฟอัลกอริทึมการ converge ของ  $\alpha_{1,l}(k)$  มีค่าไม่ถูกต้องในกรณีที่  $L = 3$  ทำให้ค่าเอาต์พุตที่ได้จาก อะแดปทีฟอัลกอริทึมมีค่าไม่ตรงกับสัญญาณอินพุตที่เข้ามา  $\alpha_1(k) \neq \cos \omega$  และเนื่องจากว่า  $\alpha_1(k)$  จะติดตามความถี่ได้เร็วหรือไม่นั้นขึ้นอยู่กับค่าของ Update Function ดังนั้นจึงใช้โครงสร้างแบบใหม่เพื่อใช้ในการแก้ปัญหาซึ่งโครงสร้างแบบใหม่จะช่วยลดสัมประสิทธิ์ใน Coefficient update circuit ของอะแดปทีฟอัลกอริทึมซึ่งจะเห็นได้จากตารางการเปรียบเทียบของอะแดปทีฟอัลกอริทึมโครงสร้างแบบขนานและโครงสร้างแบบใหม่ เมื่อใช้โครงสร้างแบบใหม่จะทำให้ค่า  $\alpha_1(k)$  ติดตามความถี่ทางด้านอินพุตที่เข้ามาได้รวดเร็วและถูกต้อง  $\alpha_1(k) = \cos \omega$  ซึ่งโครงสร้างแบบใหม่ แสดงได้ดังรูป



รูปที่ 2.3 โครงสร้างแบบใหม่

อะแดปทีฟอัลกอริทึมสำหรับ โครงสร้างใหม่

$$\alpha_1(k+1) = \alpha_1(k) + \mu \sum_{i=0}^{L-1} y(Lk+i)\psi(Lk+i) \quad 2.72$$

ซึ่งในกรณี  $L = 2$

$$\alpha_1(k+1) = \alpha_1(k) + \mu \cdot [y(2k)\psi(2k) + y(2k+1)\psi(2k+1)] \quad 2.73$$

ซึ่งในกรณี  $L = 3$

$$\alpha_1(k+1) = \alpha_1(k) + \mu \cdot [y(3k)\psi(3k) + y(3k+1)\psi(3k+1) + y(3k+2)\psi(3k+2)] \quad 2.74$$

ตารางเปรียบเทียบจำนวนของวงจรวกและคูณ ใน โครงสร้างแบบขนานและ โครงสร้างแบบใหม่

Parallel block structure	summation			multiplication		
	L=1	L=2	L=3	L=1	L=2	L=3
filter	3	6	8	4	7	9
generator of $\psi_1(k)$	2	4	7	3	5	9
Coefficient update circuit	1	2	5	3	6	13
total	6	12	20	10	18	31

ตารางที่ 1 โครงสร้างแบบขนาน

New structure	summation			multiplication		
	L=1	L=2	L=3	L=1	L=2	L=3
filter	3	6	8	4	7	9
generator of $\psi_1(k)$	2	4	7	3	5	9
Coefficient update circuit	1	1.5	2.3	3	3.5	5
total	6	11.5	17.3	10	15.5	23

ตารางที่ 2 โครงสร้างใหม่

เมื่อ ได้ทราบสเฟอ์ฟังก์ชันทั้ง 3 กรณีแล้ว สมการที่นำมาออกแบบเป็นวงจระเคปทีฟอัลกอริทึมที่ใช้สำหรับดีเทกต์สัญญาณชาชน มีค่าดังนี้  
กรณีที่  $L=1$

$$y(k) = \left(\frac{1-\alpha_0}{2}\right) \cdot [x(k) - x(k-2)] + \alpha_1(k) \cdot (1+\alpha_0) \cdot y(k-1) - \alpha_0 \cdot y(k-2)$$

$$\psi(k) = \alpha_1(k) \cdot (1+\alpha_0) \cdot \psi(k-1) - \alpha_0 \psi(k-2) + (1+\alpha_0) \cdot y(k-1)$$

$$\alpha_{1,0}(k+1) = \alpha_{1,0}(k) + \mu \cdot y(k)\psi(k)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กรณีที่  $L=2$

$$y(2k) = \frac{1-\alpha_0}{2} [x(2k) + Ax(2k-1) - (1-\alpha_0)x(2k-2) - Ax(2k-3) - \alpha_0 x(2k-4)] \\ + (A^2 + 2\alpha_0)y(2k-2) - \alpha_0^2 y(2k-4)$$

$$y(2k+1) = \frac{1-\alpha_0}{2} [Ax(2k) + x(2k+1) - (1-\alpha_0)x(2k-1) - Ax(2k-2) - \alpha_0 x(2k-3)] \\ + (A^2 + 2\alpha_0)y(2k-1) - \alpha_0^2 y(2k-3)$$

$$\psi(2k) = \frac{1-\alpha_0}{2} [(1+\alpha_0)x(2k-1) - (1+\alpha_0)x(2k-3)] + 2A(1+\alpha_0)y(2k-2) \\ + (A^2 + 2\alpha_0)\psi(2k-2) - \alpha_0^2 \psi(2k-4)$$

$$\psi(2k+1) = \frac{1-\alpha_0}{2} [(1+\alpha_0)x(2k) - (1+\alpha_0)x(2k-2)] + 2A(1+\alpha_0)y(2k-1) \\ + (A^2 + 2\alpha_0)\psi(2k-1) - \alpha_0^2 \psi(2k-3)$$

ค่า  $\alpha_1(k)$  โครงสร้างแบบขนาน

$$\alpha_{1,0}(k+1) = \alpha_{1,0}(k) + \mu \cdot y(2k)\psi(2k)$$

$$\alpha_{1,1}(k+1) = \alpha_{1,1}(k) + \mu \cdot y(2k+1)\psi(2k+1)$$

ค่า  $\alpha_1(k)$  โครงสร้างแบบใหม่

$$\alpha_1(k+1) = \alpha_1(k) + \mu \cdot [y(2k)\psi(2k) + y(2k+1)\psi(2k+1)]$$

กรณีที่  $L=3$

$$y(3k) = \frac{1-\alpha_0}{2} \left[ x(3k) + Ax(3k-1) + (A^2 - \alpha_0 - 1)x(3k-2) - (1-\alpha_0)Ax(3k-3) \right] \\ + (\alpha_0^2 + \alpha_0 - A^2)x(3k-4) - A\alpha_0 x(3k-5) - \alpha_0^2 x(3k-6) \\ + (A^3 - 3A\alpha_0)y(3k-3) - \alpha_0^3 y(3k-6)$$

$$y(3k+1) = \frac{1-\alpha_0}{2} \left[ Ax(3k) + x(3k+1) + (A^2 - \alpha_0 - 1)x(3k-1) - (1-\alpha_0)Ax(3k-2) \right] \\ + (\alpha_0^2 + \alpha_0 - A^2)x(3k-3) - A\alpha_0 x(3k-4) - \alpha_0^2 x(3k-5) \\ + (A^3 - 3A\alpha_0)y(3k-2) - \alpha_0^3 y(3k-5)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$y(3k+2) = \frac{1-\alpha_0}{2} \left[ \begin{aligned} &(A^2 - \alpha_0 - 1)x(3k) + Ax(3k+1) + x(3k+2) - (1-\alpha_0)Ax(3k-1) \\ &+ (\alpha_0^2 + \alpha_0 - A^2)x(3k-2) - A\alpha_0x(3k-3) - \alpha_0^2x(3k-4) \\ &+ (A^3 - 3A\alpha_0)y(3k-1) - \alpha_0^3y(3k-4) \end{aligned} \right]$$

$$\psi(3k) = \frac{1-\alpha_0}{2} \left[ \begin{aligned} &(1+\alpha_0)x(3k-1) + 2A(1+\alpha_0)x(3k-2) - (1-\alpha_0^2)x(3k-3) \\ &- 2A(1+\alpha_0)x(3k-4) - (1+\alpha_0)\alpha_0x(3k-5) \end{aligned} \right]$$

$$+ (3A^2(1+\alpha_0) - 3\alpha_0(1+\alpha_0))y(3k-3) + (A^3 + 3A\alpha_0)\psi(3k-3) - \alpha_0^3\psi(3k-6)$$

$$\psi(3k+1) = \frac{1-\alpha_0}{2} \left[ \begin{aligned} &(1+\alpha_0)x(3k) + 2A(1+\alpha_0)x(3k-1) - (1-\alpha_0^2)x(3k-2) - 2A(1+\alpha_0) \\ &\cdot x(3k-3) - 2A(1+\alpha_0)x(3k-3) - (1+\alpha_0)\alpha_0x(3k-4) \end{aligned} \right]$$

$$+ (3A^2(1+\alpha_0) - 3\alpha_0(1+\alpha_0))y(3k-2) + (A^3 - 3A\alpha_0)\psi(3k-2) - \alpha_0^3\psi(3k-5)$$

$$\psi(3k+2) = \frac{1-\alpha_0}{2} \left[ \begin{aligned} &2A(1+\alpha_0)x(3k) + (1+\alpha_0)x(3k+1) + (1-\alpha_0^2)x(3k-1) - 2A(1+\alpha_0) \\ &\cdot x(3k-2) - (1+\alpha_0)\alpha_0x(3k-3) \end{aligned} \right]$$

$$+ (3A^2(1+\alpha_0) - 3\alpha_0(1+\alpha_0))y(3k-1) + (A^3 - 3A\alpha_0)\psi(3k-1) - \alpha_0^3\psi(3k-4)$$

ถ้า  $\alpha_1(k)$  โครงสร้างแบบขนาน

$$\alpha_{1,0}(k+1) = \alpha_{1,0}(k) + \mu \cdot y(3k)\psi(3k)$$

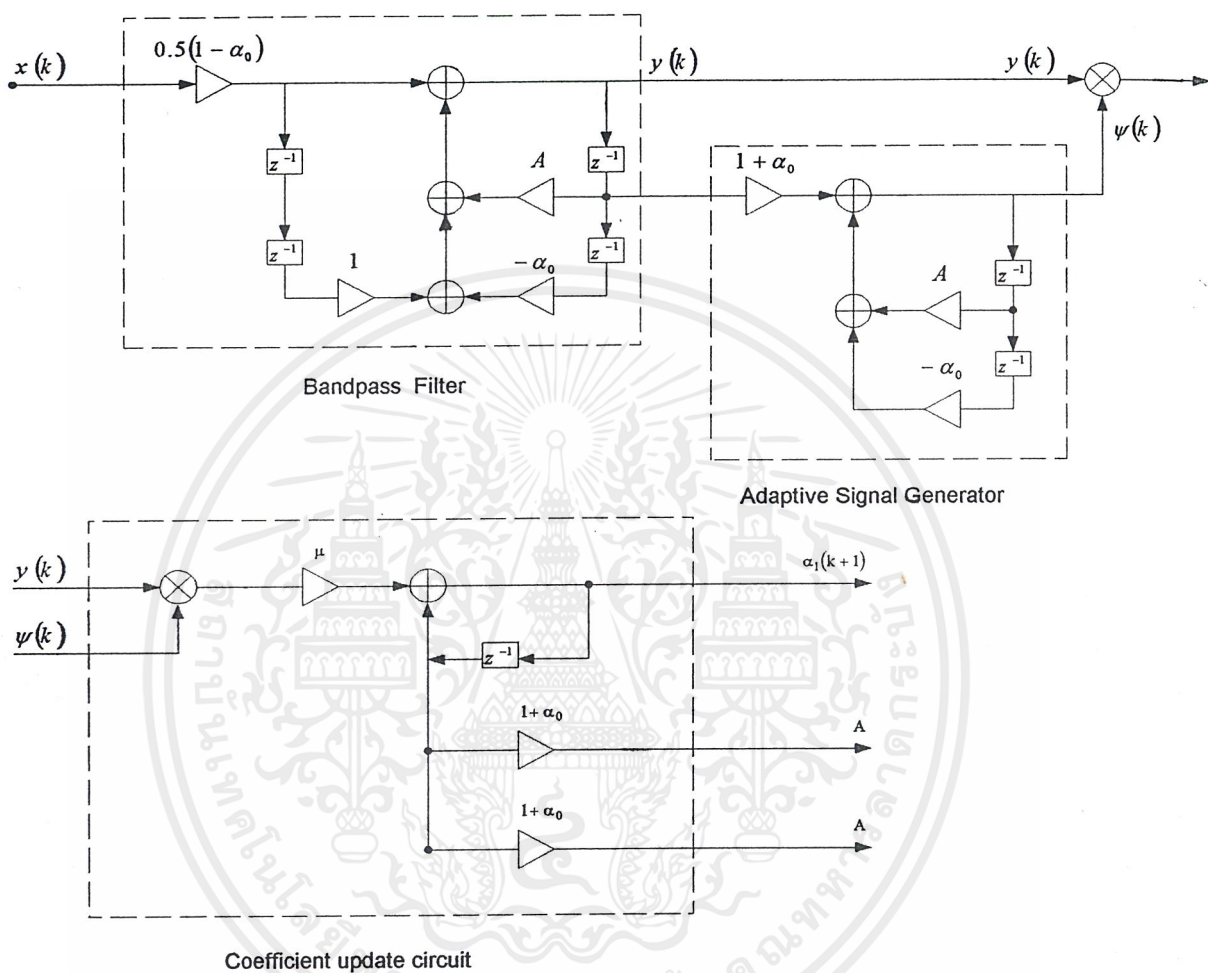
$$\alpha_{1,1}(k+1) = \alpha_{1,1}(k) + \mu \cdot y(3k+1)\psi(3k+1)$$

$$\alpha_{1,2}(k+1) = \alpha_{1,2}(k) + \mu \cdot y(3k+2)\psi(3k+2)$$

ถ้า  $\alpha_1(k)$  โครงสร้างแบบใหม่

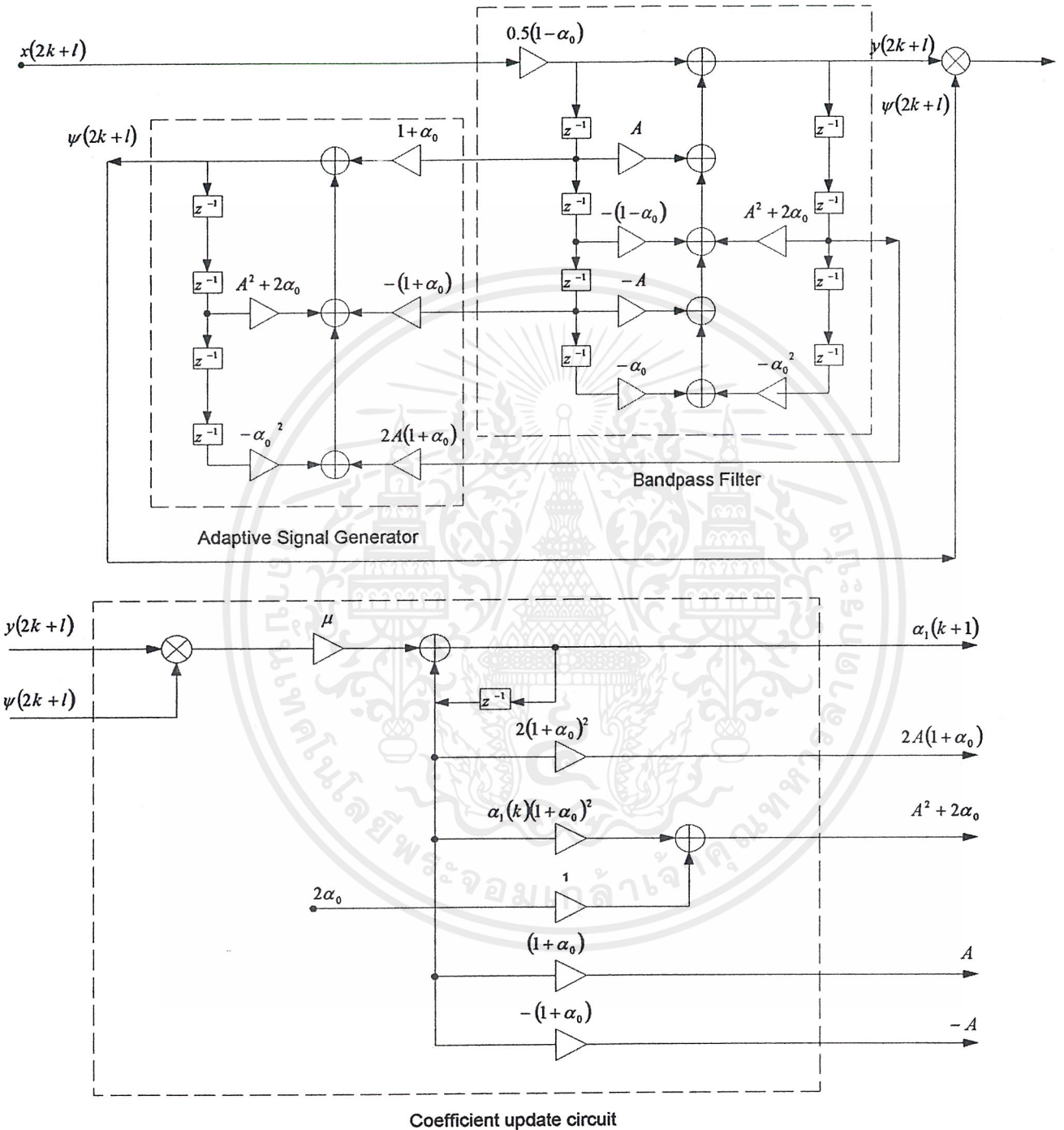
$$\alpha_1(k+1) = \alpha_1(k) + \mu \cdot [y(3k)\psi(3k) + y(3k+1)\psi(3k+1) + y(3k+2)\psi(3k+2)]$$

สามารถนำมาใช้เขียนวงจระแคปทีฟฟิลเตอร์ได้ดังรูป



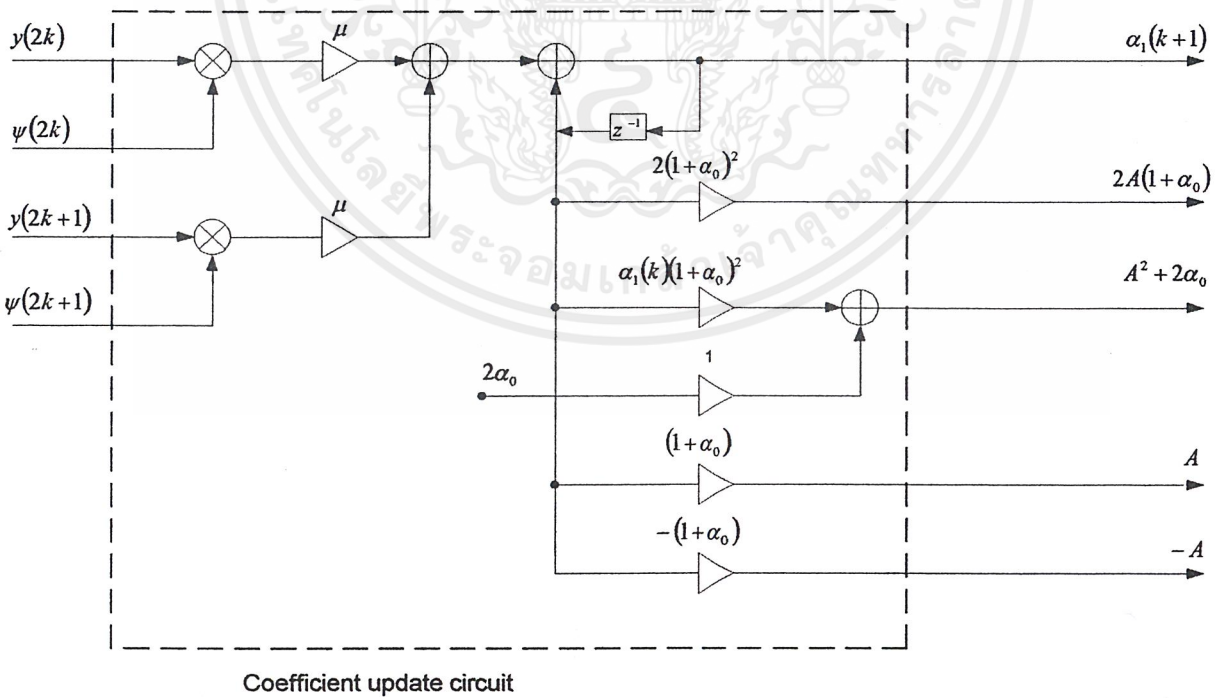
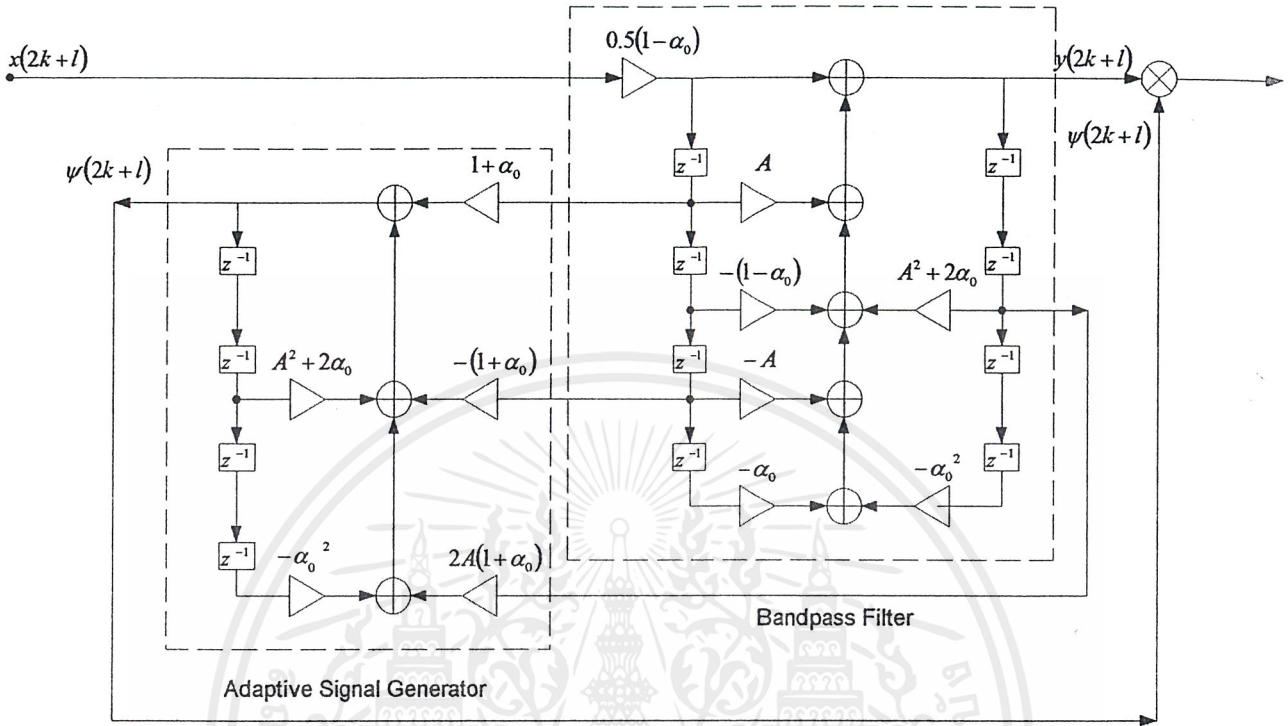
รูปที่ 2.4 วงจระแคปทีฟฟิลเตอร์โครงสร้างแบบขนาน กรณี  $L=1$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



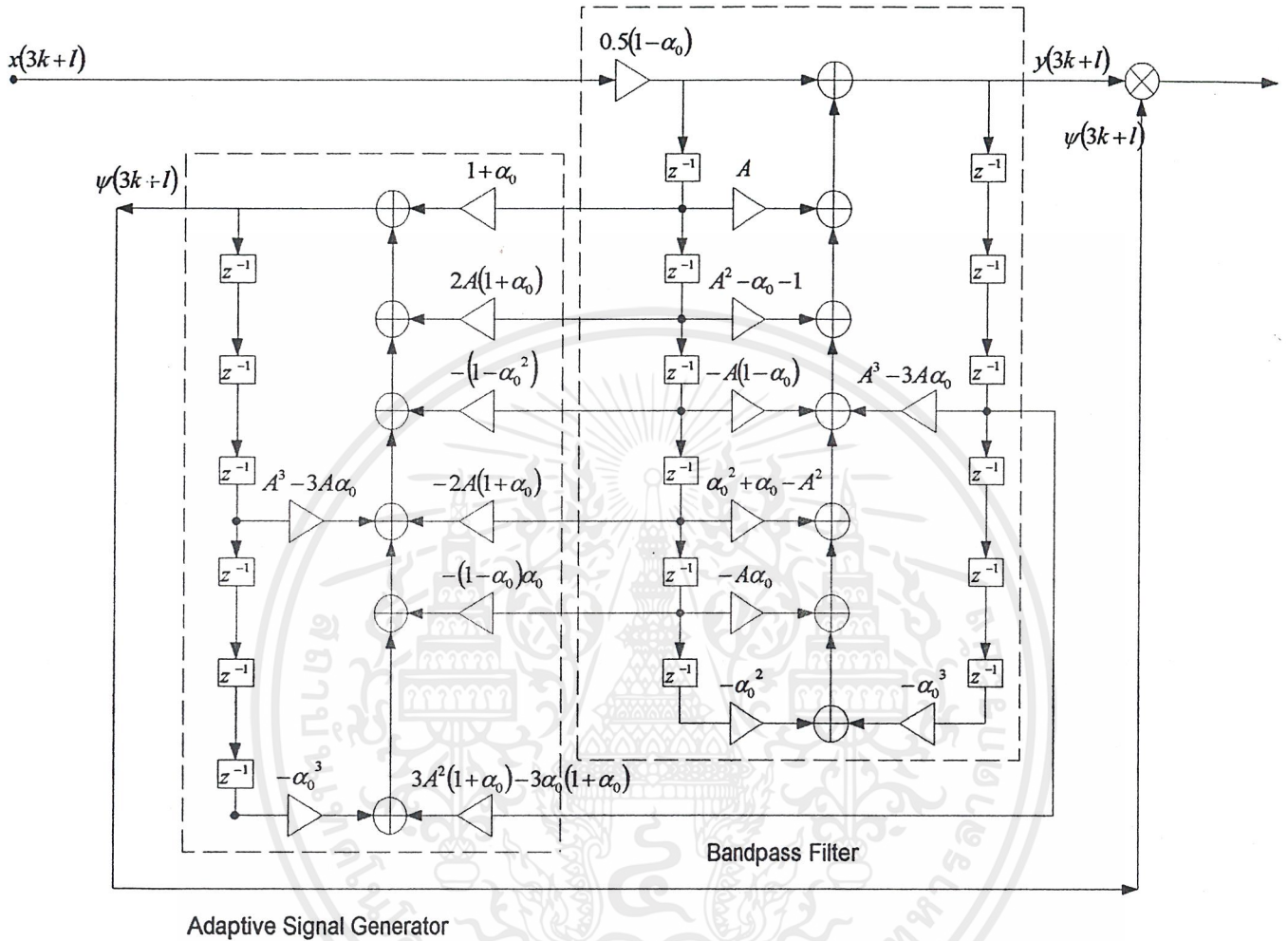
รูปที่ 2.5 วงจรอะแดปทีฟฟิลเตอร์โครงสร้างแบบขนาน กรณี  $L = 2$ ,  $l = 0$  และ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



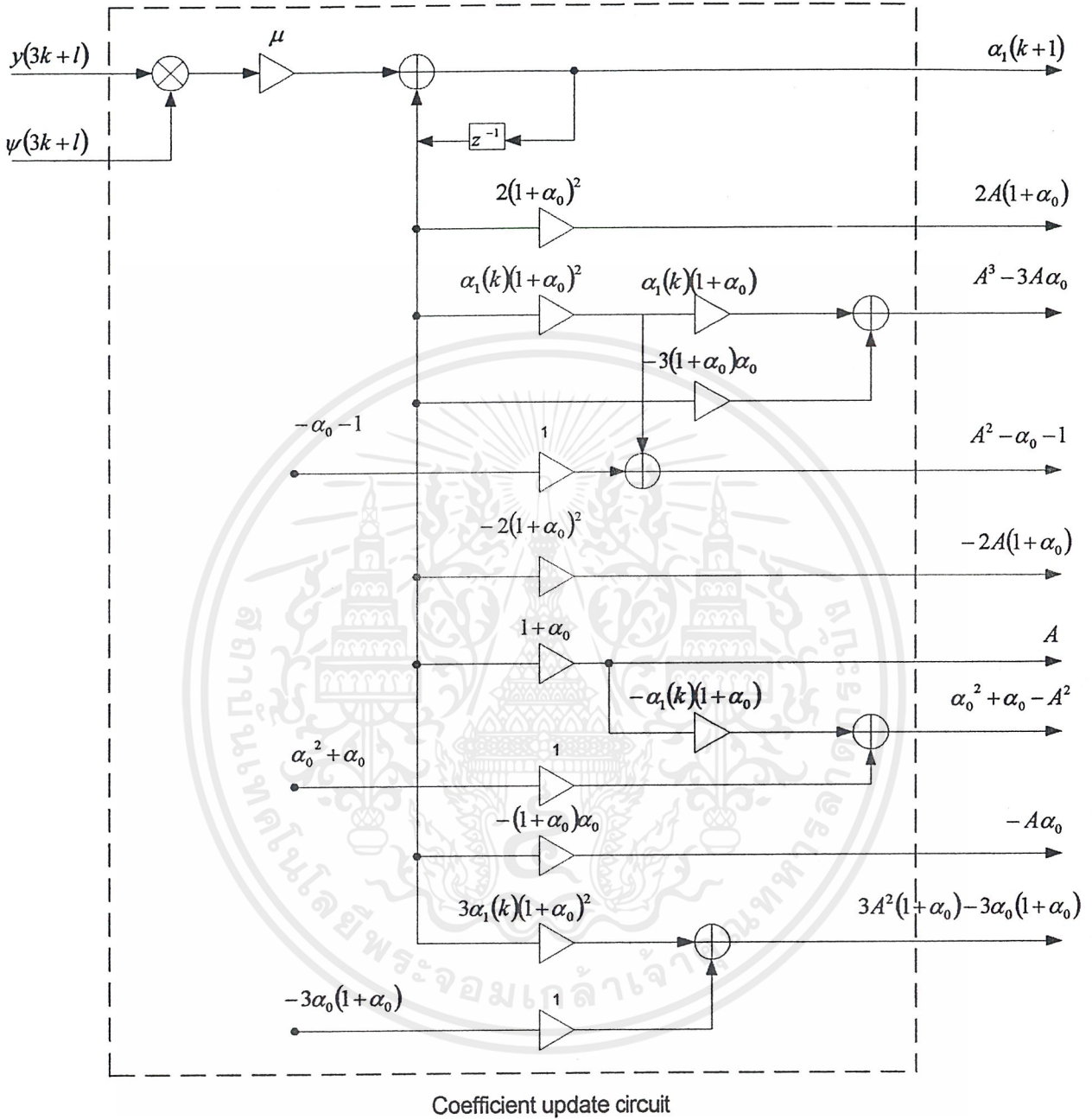
รูปที่ 2.6 วงจรอะแดปทีฟฟิลเตอร์โครงสร้างแบบใหม่กรณี  $L=2$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



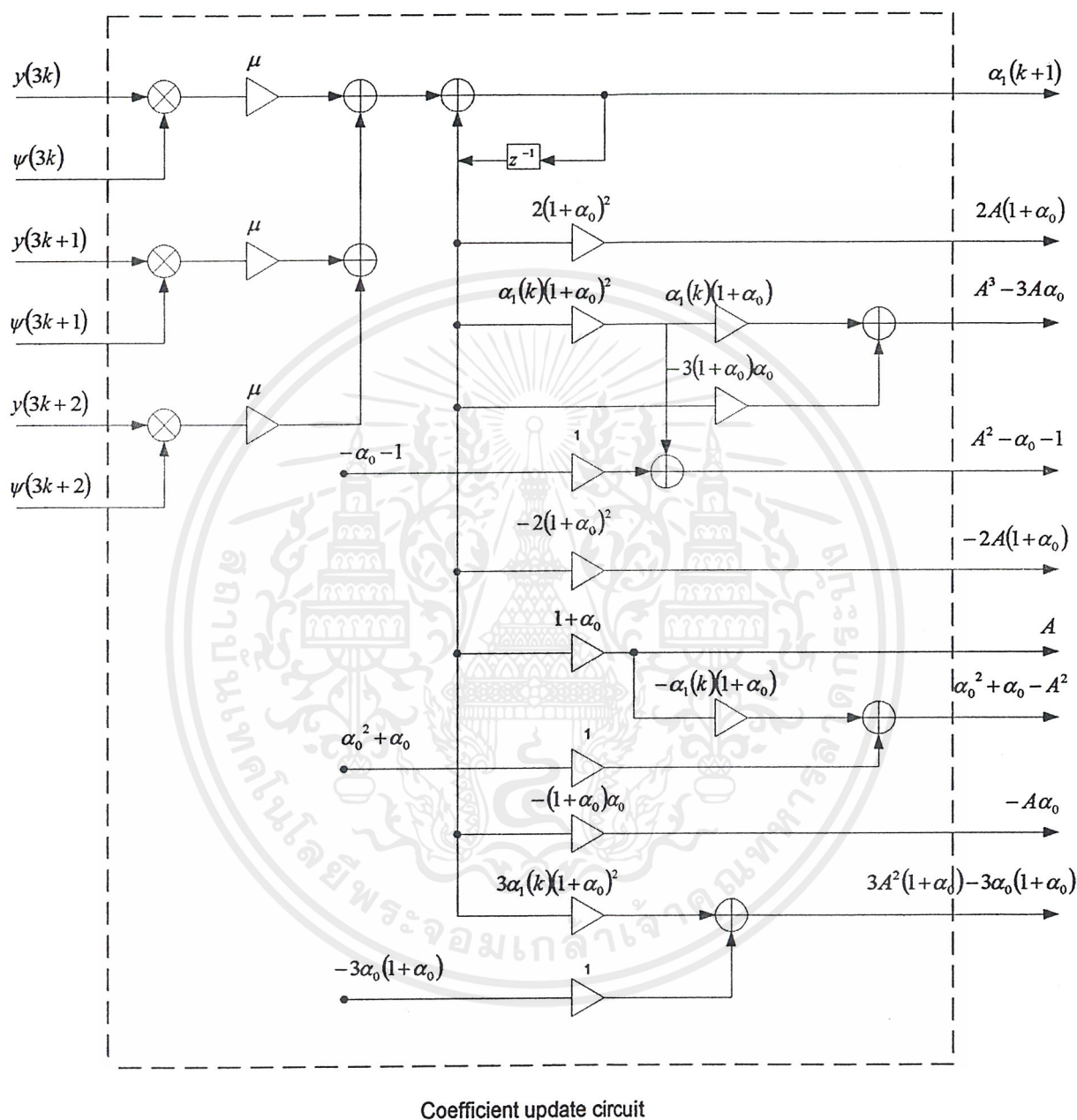
รูปที่ 2.7 วงจรอะแดปทีฟฟิลเตอร์โครงสร้างแบบขนานกรณี  $L=3$ ,  $l=0,1$  และ  $2$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.8 วงจร Coefficient update circuit โครงสร้างแบบขนานกรณี  $L=3$ ,  $l=0, 1$  และ  $2$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.9 วงจร Coefficient update circuit โครงสร้างแบบใหม่กรณี  $L=3$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 3

### โปรแกรมและผลการทดลอง

#### 3.1 โปรแกรม

##### 3.1.1 โปรแกรมแสดงคุณสมบัติของ $|H(\omega)|$ เทียบกับ $\omega$

```
clc
alpha0=-1;
while (alpha0<0|alpha0>1)
    alpha0=input('alpha0 0 to 1\n(alpha0 =)');
end
omega=-1;
while(omega<0|omega>180)
    omega=input('W (0<w<pi)(degree)\n(W=)');
end
alpha1=cos((pi/180)*omega);
b1=1;
b2=0;
b3=-1;
a1=1;
a2=-alpha1.*(1+alpha0);
a3=alpha0;
k=(1-alpha0)./2;
n=[b1,b2,b3];
d=[a1,a2,a3];
[Tz,ws]=freqz(n,d,200);
plot(ws,abs(Tz*k),'r');
xlabel('W(radian)');ylabel('|H(w)|');
grid; hold on;
```

### 3.1.2 โปรแกรมแสดงคุณสมบัติของ $|H(\omega)|$ เทียบกับ $\alpha_1$

```

clc
alpha0=-1;
while(alpha0<0|alpha0>1)
    alpha0=input('Q factor(0<alpha0<1)\n(alpha0=)');
end

omega=-1;
while(omega<0|omega>180)
    omega=input('W (0<w<pi)(pi*)\n(w=)');
end

alpha1=linspace(0,1,100);
h=zeros(size(alpha1));
h=abs([j.*sin(omega)]./[[cos(omega)-alpha1].*[[1+alpha0]./[1-alpha0]]]+j.*sin(omega)]);
plot(alpha1,h);
grid;
xlabel('alpha1(k)');
ylabel('|H(w)|');
hold on

```

### 3.1.3 โปรแกรมแสดงคุณสมบัติของ $\alpha_1(k)$ เทียบกับ $k$ เมื่อ $L = 1$

```

w=-1;
while(w<0|w>180)
    w=input('w(0 to 180)\n(W=)');
end
n=500000;
k=0:n-1;
data=cos((pi*w/180)*k+randn(1,1))+0.1*randn(1,n);
alp0=0.8;
u=0.00005;
x=zeros(n+2,1);
x(3:n+2)=data;
w1=zeros(n+2,1);
alp1=zeros(n+2,1);
y=zeros(n+2,1);
for k=3:n+2;
    y(k)=(1-alp0)/2.*(x(k)-x(k-2))+alp1(k)*(1+alp0).*y(k-1)-alp0.*y(k-2);
    w1(k)=alp1(k).(1+alp0).*w1(k-1)-alp0.*w1(k-2)+(1+alp0).*y(k-1);
    alp1(k+1)=alp1(k)+u*y(k)*w1(k);
end
plot(alp1);
grid;
xlabel('K');ylabel('alp1(k)');
hold on;
clear;

```

### 3.1.4 โปรแกรมแสดงคุณสมบัติของ $\alpha_1(k)$ เทียบกับ $k$ เมื่อ $L = 1$ กรณีมีการเปลี่ยนค่าตัวแปร

S/N = 1.8 dB

```
w=-1;
while(w<0|w>180)
    w=input('w(0 to 180)\n(W=)');      % กำหนดค่า w ของวงจรถองความถี่
end
n=500000;
k=0:n-1;
data=A*cos((pi*w/180)*k+randn(1,1))+0.1*randn(1,n);% A คือ Amplitude input data+gaussian noise
alp0=B;                                % B กำหนดค่า Q-Factor ให้ Adaptive เมื่อ  $0 < \alpha_0 < 1$ 
u=C;                                    % C คือ Step size parameter ( $\mu > 0$ )
x=zeros(n+2,1);
x(3:n+2)=data;
w1=zeros(n+2,1);
alp1=zeros(n+2,1);
y=zeros(n+2,1);
for k=3:n+2;
    y(k)=(1-alp0)/2.*(x(k)-x(k-2))+alp1(k)*(1+alp0).*y(k-1)-alp0.*y(k-2);
    w1(k)=alp1(k).(1+alp0).*w1(k-1)-alp0.*w1(k-2)+(1+alp0).*y(k-1);
    alp1(k+1)=alp1(k)+u*y(k)*w1(k);
end
plot(alp1);                            % กราฟของการเปลี่ยนแปลงของ  $\alpha_1(k)$  เทียบกับ  $k$ 
grid;
xlabel('K');ylabel('alpa1(k)');
hold on;
clear;
```

### 3.1.5 โปรแกรมแสดงคุณสมบัติของ $\alpha_{1,0}(k)$ เทียบกับ $k$ เมื่อ $L = 3, l = 0$

```

w=-1;
while(w<0|w>180)
    w=input('w(0 to 180)\n(W=)');
end
n=500000;
k=0:n-1;
data1=cos(((pi*w/180)*k)+randn(1,1))+0.1*randn(1,n);
alp0=0.8;
u=0.00005;
x=zeros(n+8,1);
x(9:n+8)=data1;
y=zeros(n+8,1);
w1=zeros(n+8,1);
alp1=zeros(n+8,1);
for k=3:((n+8)-mod(n,3))/3
    y(3*k) = (1-alp0)/2*(x(3*k)+alp1(k)*(1+alp0)*x(3*k-1)+(((alp1(k)*(1+alp0)).^2)-alp0-1)*x(3*k-2)-((1-alp0)*alp1(k)*(1+alp0))*x(3*k-3)+(((alp0).^2)+alp0-((alp1(k)*(1+alp0)).^2))*x(3*k-4)-((alp1(k)*(1+alp0))*alp0)*x(3*k-5)-((alp0).^2)*x(3*k-6))+(((alp1(k)*(1+alp0)).^3)-3*alp1(k)*(1+alp0)*alp0)*y(3*k-3)-((alp0).^3)*y(3*k-6);

    y(3*k+1)= (1-alp0)/2*(alp1(k)*(1+alp0)*x(3*k)+(((alp0).^2)+alp0-((alp1(k)*(1+alp0)).^2))*x(3*k-3)+x(3*k+1)-(1-alp0)*alp1(k)*(1+alp0)*x(3*k-2)-((alp0).^2)*x(3*k-5)+(((alp1(k)*(1+alp0)).^2)-alp0-1)*x(3*k-1)-((alp1(k)*(1+alp0))*alp0)*x(3*k-4))+((alp1(k)*(1+alp0)).^3)*y(3*k-2)-3*(alp1(k)*(1+alp0))*alp0*y(3*k-2)-((alp0).^3)*y(3*k-5);

```

$$y(3^*k+2)=(1-\text{alp0})/2*(((\text{alp1}(k)*(1+\text{alp0}))^2)-\text{alp0}-1)*x(3^*k)-\text{alp1}(k)*(1+\text{alp0})*\text{alp0}*x(3^*k-3)+\text{alp1}(k)*(1+\text{alp0})*x(3^*k+1)+((\text{alp0})^2+\text{alp0}-((\text{alp1}(k)*(1+\text{alp0}))^2))*x(3^*k-2)+x(3^*k+2)-\text{alp1}(k)*(1+\text{alp0})*(1-\text{alp0})*x(3^*k-1)-((\text{alp0})^2)*x(3^*k-4))+\text{alp1}(k)*(1+\text{alp0})*(((\text{alp1}(k)*(1+\text{alp0}))^2)-3*\text{alp0})*y(3^*k-1)-((\text{alp0})^3)*y(3^*k-4);$$

$$w1(3^*k) = (1-\text{alp0})/2*((1+\text{alp0})*x(3^*k-1)+2*\text{alp1}(k)*(1+\text{alp0})*(1+\text{alp0})*x(3^*k-2)-(1-\text{alp0})*(1+\text{alp0})*x(3^*k-3)-2*\text{alp1}(k)*(1+\text{alp0})*(1+\text{alp0})*x(3^*k-4)-(1+\text{alp0})*\text{alp0}*x(3^*k-5))+3*((\text{alp1}(k)*(1+\text{alp0}))^2)*(1+\text{alp0})-3*\text{alp0}*(1+\text{alp0}))*y(3^*k-3+0)+(((\text{alp1}(k)*(1+\text{alp0}))^3)-3*\text{alp1}(k)*(1+\text{alp0})*\text{alp0})*w1(3^*k-3+0)-((\text{alp0})^3)*w1(3^*k-6);$$

$$w1(3^*k+1) = ((1-\text{alp0})/2*((1+\text{alp0})*x(3^*k)-2*\text{alp1}(k)*(1+\text{alp0})*(1+\text{alp0})*x(3^*k-3)-(1-((\text{alp0})^2))*x(3^*k-2)+2*\text{alp1}(k)*(1+\text{alp0})*(1+\text{alp0})*x(3^*k-1)-(1+\text{alp0})*\text{alp0}*x(3^*k-4))+3*(1+\text{alp0})*((\text{alp1}(k)*(1+\text{alp0}))^2)-3*\text{alp0}*(1+\text{alp0}))*y(3^*k-2))+((\text{alp1}(k)*(1+\text{alp0}))^3)*w1(3^*k-2)-3*\text{alp0}*\text{alp1}(k)*(1+\text{alp0})*w1(3^*k-2)-((\text{alp0})^3)*w1(3^*k-5);$$

$$w1(3^*k+2) = ((1-\text{alp0})/2*(2*\text{alp1}(k)*(1+\text{alp0})*(1+\text{alp0})*x(3^*k)-(1+\text{alp0})*\text{alp0}*x(3^*k-3)+(1+\text{alp0})*x(3^*k+1)-2*\text{alp1}(k)*(1+\text{alp0})*(1+\text{alp0})*x(3^*k-2)-(1-((\text{alp0})^2))*x(3^*k-1))+3*(1+\text{alp0})*((\text{alp1}(k)*(1+\text{alp0}))^2)-3*\text{alp0}*(1+\text{alp0}))*y(3^*k-1))+((\text{alp1}(k)*(1+\text{alp0}))^3)*w1(3^*k-1)-3*\text{alp0}*\text{alp1}(k)*(1+\text{alp0})*w1(3^*k-1)-((\text{alp0})^3)*w1(3^*k-4);$$

$$\text{alp1}(k+1) = \text{alp1}(k)+u*y(3^*k)*w1(3^*k);$$

end

plot(alp1, 'b');

grid;

xlabel('k');

ylabel('alp1(k)');

hold on;

clear

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.1.6 โปรแกรมแสดงคุณสมบัติของ $\alpha_{1,1}(k)$ เทียบกับ $k$ เมื่อ $L = 3, I = 1$

```

w=-1;
while(w<0|w>180)
    w=input('w(0 to 180)\n(W=)');
end
n=500000;
k=0:n-1;
data1=cos(((pi*w/180)*k)+randn(1,1))+0.1*randn(1,n);
alp0=0.8;
u=0.00005;
x=zeros(n+8,1);
x(9:n+8)=data1;
y=zeros(n+8,1);
w1=zeros(n+8,1);
alp1=zeros(n+8,1);
for k=3:((n+8)-mod(n,3))/3
    y(3*k) =(1-alp0)/2*(x(3*k)+alp1(k)*(1+alp0)*x(3*k-1)+(((alp1(k)*(1+alp0)).^2)-alp0-1)*x(3*k-2)-((1-alp0)*alp1(k)*(1+alp0))*x(3*k-3)+(((alp0).^2)+alp0-((alp1(k)*(1+alp0)).^2))*x(3*k-4)-((alp1(k)*(1+alp0))*alp0)*x(3*k-5)-((alp0).^2)*x(3*k-6)+(((alp1(k)*(1+alp0)).^3)-3*alp1(k)*(1+alp0)*alp0)*y(3*k-3)-((alp0).^3)*y(3*k-6);

    y(3*k+1)=(1-alp0)/2*(alp1(k)*(1+alp0)*x(3*k)+(((alp0)^2)+alp0-((alp1(k)*(1+alp0))^2))*x(3*k-3)+x(3*k+1)-(1-alp0)*alp1(k)*(1+alp0)*x(3*k-2)-((alp0).^2)*x(3*k-5)+(((alp1(k)*(1+alp0))^2)-alp0-1)*x(3*k-1)-(alp1(k)*(1+alp0))*alp0*x(3*k-4)+((alp1(k)*(1+alp0))^3)*y(3*k-2)-3*alp1(k)*(1+alp0)*alp0*y(3*k-2)-((alp0)^3)*y(3*k-5);

```

$$y(3^*k+2)=(1-\text{alp0})/2*(((\text{alp1}(k)*(1+\text{alp0}))^2)-\text{alp0}-1)*x(3^*k)-\text{alp1}(k)*(1+\text{alp0})*\text{alp0}*x(3^*k-3))+\text{alp1}(k)*(1+\text{alp0})*x(3^*k+1)+((\text{alp0})^2+\text{alp0}-((\text{alp1}(k)*(1+\text{alp0}))^2))*x(3^*k-2)+x(3^*k+2)-\text{alp1}(k)*(1+\text{alp0})*(1-\text{alp0})*x(3^*k-1)-((\text{alp0})^2)*x(3^*k-4))+\text{alp1}(k)*(1+\text{alp0})*(((\text{alp1}(k)*(1+\text{alp0}))^2)-3*\text{alp0})*y(3^*k-1)-((\text{alp0})^3)*y(3^*k-4);$$

$$w1(3^*k) = (1-\text{alp0})/2*((1+\text{alp0})*x(3^*k-1)+2*\text{alp1}(k)*(1+\text{alp0})*(1+\text{alp0})*x(3^*k-2)-(1-\text{alp0})*(1+\text{alp0})*x(3^*k-3)-2*\text{alp1}(k)*(1+\text{alp0})*(1+\text{alp0})*x(3^*k-4)-(1+\text{alp0})*\text{alp0}*x(3^*k-5))+3*((\text{alp1}(k)*(1+\text{alp0}))^2)*(1+\text{alp0})-3*\text{alp0}*(1+\text{alp0}))*y(3^*k-3+0)+(((\text{alp1}(k)*(1+\text{alp0}))^3)-3*\text{alp1}(k)*(1+\text{alp0})*\text{alp0})*w1(3^*k-3+0)-((\text{alp0})^3)*w1(3^*k-6);$$

$$w1(3^*k+1) = ((1-\text{alp0})/2*((1+\text{alp0})*x(3^*k)-2*\text{alp1}(k)*(1+\text{alp0})*(1+\text{alp0})*x(3^*k-3)-(1-((\text{alp0})^2))*x(3^*k-2)+2*\text{alp1}(k)*(1+\text{alp0})*(1+\text{alp0})*x(3^*k-1)-(1+\text{alp0})*\text{alp0}*x(3^*k-4))+3*(1+\text{alp0})*((\text{alp1}(k)*(1+\text{alp0}))^2)-3*\text{alp0}*(1+\text{alp0}))*y(3^*k-2))+((\text{alp1}(k)*(1+\text{alp0}))^3)*w1(3^*k-2)-3*\text{alp0}*\text{alp1}(k)*(1+\text{alp0})*w1(3^*k-2)-((\text{alp0})^3)*w1(3^*k-5);$$

$$w1(3^*k+2) = ((1-\text{alp0})/2*(2*\text{alp1}(k)*(1+\text{alp0})*(1+\text{alp0})*x(3^*k)-(1+\text{alp0})*\text{alp0}*x(3^*k-3)+(1+\text{alp0})*x(3^*k+1)-2*\text{alp1}(k)*(1+\text{alp0})*(1+\text{alp0})*x(3^*k-2)-(1-((\text{alp0})^2))*x(3^*k-1))+3*(1+\text{alp0})*((\text{alp1}(k)*(1+\text{alp0}))^2)-3*\text{alp0}*(1+\text{alp0}))*y(3^*k-1))+((\text{alp1}(k)*(1+\text{alp0}))^3)*w1(3^*k-1)-3*\text{alp0}*\text{alp1}(k)*(1+\text{alp0})*w1(3^*k-1)-((\text{alp0})^3)*w1(3^*k-4);$$

$$\text{alp1}(k+1) = \text{alp1}(k)+u*y(3^*k+1)*w1(3^*k+1);$$

end

plot(alp1, 'r');

grid;

xlabel('k');

ylabel('alp1(k)');

hold on

clear

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.1.7 โปรแกรมแสดงคุณสมบัติของ $\alpha_{1,2}(k)$ เทียบกับ $k$ เมื่อ $L = 3, l = 2$

```

w=-1;
while(w<0|w>180)
    w=input('w(0 to 180)\n(W=)');
end
n=500000;
k=0:n-1;
data1=cos(((pi*w/180)*k)+randn(1,1))+0.1*randn(1,n);
alp0=0.8;
u=0.00005;
x=zeros(n+8,1);
x(9:n+8)=data1;
y=zeros(n+8,1);
w1=zeros(n+8,1);
alp1=zeros(n+8,1);
for k=3:((n+8)-mod(n,3))/3

    y(3*k)  =(1-alp0)/2*(x(3*k)+alp1(k)*(1+alp0)*x(3*k-1)+(((alp1(k)*(1+alp0)).^2)-alp0-1)*x(3*k-2)-((1-alp0)*alp1(k)*(1+alp0))*x(3*k-3)+((alp0).^2)+alp0-((alp1(k)*(1+alp0)).^2))*x(3*k-4)-((alp1(k)*(1+alp0))*alp0)*x(3*k-5)-((alp0).^2)*x(3*k-6)+(((alp1(k)*(1+alp0)).^3)-3*alp1(k)*(1+alp0)*alp0)*y(3*k-3)-((alp0).^3)*y(3*k-6);

    y(3*k+1)=(1-alp0)/2*(alp1(k)*(1+alp0)*x(3*k)+(((alp0)^2)+alp0-((alp1(k)*(1+alp0))^2))*x(3*k-3)+x(3*k+1)-(1-alp0)*alp1(k)*(1+alp0)*x(3*k-2)-((alp0).^2)*x(3*k-5)+(((alp1(k)*(1+alp0))^2)-alp0-1)*x(3*k-1)-(alp1(k)*(1+alp0))*alp0*x(3*k-4)+((alp1(k)*(1+alp0))^3)*y(3*k-2)-3*(alp1(k)*(1+alp0))*alp0*y(3*k-2)-((alp0)^3)*y(3*k-5);

```

$$y(3^*k+2)=(1-\text{alp0})/2*(((\text{alp1}(k)*(1+\text{alp0}))^2)-\text{alp0}-1)*x(3^*k)-\text{alp1}(k)*(1+\text{alp0})*\text{alp0}*x(3^*k-3)+\text{alp1}(k)*(1+\text{alp0})*x(3^*k+1)+(((\text{alp0})^2)+\text{alp0}-((\text{alp1}(k)*(1+\text{alp0}))^2))*x(3^*k-2)+x(3^*k+2)-\text{alp1}(k)*(1+\text{alp0})*(1-\text{alp0})*x(3^*k-1)-((\text{alp0})^2)*x(3^*k-4))+\text{alp1}(k)*(1+\text{alp0})*(((\text{alp1}(k)*(1+\text{alp0}))^2)-3*\text{alp0})*y(3^*k-1)-((\text{alp0})^3)*y(3^*k-4);$$

$$w1(3^*k) = (1-\text{alp0})/2*((1+\text{alp0})*x(3^*k-1)+2*\text{alp1}(k)*(1+\text{alp0})*(1+\text{alp0})*x(3^*k-2)-(1-\text{alp0})*(1+\text{alp0})*x(3^*k-3)-2*\text{alp1}(k)*(1+\text{alp0})*(1+\text{alp0})*x(3^*k-4)-(1+\text{alp0})*\text{alp0}*x(3^*k-5))+3*((\text{alp1}(k)*(1+\text{alp0}))^2)*(1+\text{alp0})-3*\text{alp0}*(1+\text{alp0}))*y(3^*k-3+0)+(((\text{alp1}(k)*(1+\text{alp0}))^3)-3*\text{alp1}(k)*(1+\text{alp0})*\text{alp0})*w1(3^*k-3+0)-((\text{alp0})^3)*w1(3^*k-6);$$

$$w1(3^*k+1) = ((1-\text{alp0})/2*((1+\text{alp0})*x(3^*k)-2*\text{alp1}(k)*(1+\text{alp0})*(1+\text{alp0})*x(3^*k-3)-(1-((\text{alp0})^2))*x(3^*k-2)+2*\text{alp1}(k)*(1+\text{alp0})*(1+\text{alp0})*x(3^*k-1)-(1+\text{alp0})*\text{alp0}*x(3^*k-4))+3*(1+\text{alp0})*((\text{alp1}(k)*(1+\text{alp0}))^2)-3*\text{alp0}*(1+\text{alp0}))*y(3^*k-2))+((\text{alp1}(k)*(1+\text{alp0}))^3)*w1(3^*k-2)-3*\text{alp0}*\text{alp1}(k)*(1+\text{alp0})*w1(3^*k-2)-((\text{alp0})^3)*w1(3^*k-5);$$

$$w1(3^*k+2) = ((1-\text{alp0})/2*(2*\text{alp1}(k)*(1+\text{alp0})*(1+\text{alp0})*x(3^*k)-(1+\text{alp0})*\text{alp0}*x(3^*k-3)+(1+\text{alp0})*x(3^*k+1)-2*\text{alp1}(k)*(1+\text{alp0})*(1+\text{alp0})*x(3^*k-2)-(1-((\text{alp0})^2))*x(3^*k-1))+3*(1+\text{alp0})*((\text{alp1}(k)*(1+\text{alp0}))^2)-3*\text{alp0}*(1+\text{alp0}))*y(3^*k-1))+((\text{alp1}(k)*(1+\text{alp0}))^3)*w1(3^*k-1)-3*\text{alp0}*\text{alp1}(k)*(1+\text{alp0})*w1(3^*k-1)-((\text{alp0})^3)*w1(3^*k-4);$$

$$\text{alp1}(k+1) = \text{alp1}(k)+u*y(3^*k+2)*w1(3^*k+2);$$

end

plot(alp1, 'g');

grid;

xlabel('k');

ylabel('alp1(k)');

hold on

clear

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.1.8 โปรแกรมแสดงคุณสมบัติการเปลี่ยนแปลงของ $\alpha_1(k)$ กรณี $L = 1$

```

w=-1;
while(w<0|w>180)
    w=input('w(0 to 180)\n(W=)');
end
n=500;
k=0:n-1;
N=1000;
data=cos(((pi*w/180)*k)+randn(1,1))+0.1*randn(1,length(k));
alp0=0.85;
u=0.00005;
x=zeros(n+2,1);
x(3:n+2)=data;
w1=zeros(n+4,1);
alp1=linspace(0,1,N);
y=zeros(n+4,1);
malp1=zeros(N,1);
for i=1:N
    for k=3:n+2
        y(k)=(1-alp0)/2.*(x(k)-x(k-2))+alp1(i)*(1+alp0).*y(k-1)-alp0.*y(k-2);
        w1(k)=alp1(i).(1+alp0).*w1(k-1)-alp0.*w1(k-2)+(1+alp0).*y(k-1);
        malp1(i)=1/6*(y(k-2)*w1(k-2)+y(k-1)*w1(k-1)+y(k)*w1(k)+y(k+1)*w1(k+1)+y(k+2)*w1(k+2));
    end
end
plot(alp1,malp1,'r');
grid;
xlabel('alpha(k+1)');ylabel('malp');
title('fig'); hold on; clear

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.1.9 โปรแกรมแสดงคุณสมบัติการเปลี่ยนแปลงของ $\alpha_1(k)$ กรณี $L = 3, l = 0$

```

clc
w=-1;
while(w<0|w>180)
    w=input('w(0 to 180)\n(W=)');
end
n=500;
k=0:n-1;
N=1000;
data1=cos(((pi*w/180)*k)+randn(1,1))+0.1*randn(1,length(k));
alp0=0.8;
u=0.00005;
x=zeros(n+8,1);
x(9:n+8)=data1;
y=zeros(n+8,1);
wl=zeros(n+8,1);
alp1=linspace(0,1,N);
malp1=zeros(N,1);
for i=1:N
    for k=3:((n+8)-mod(n,3))/3
        y(3*k) =(1-alp0)/2*(x(3*k)+alp1(i)*(1+alp0)*x(3*k-1)+(((alp1(i)*(1+alp0)).^2)-alp0-1)*x(3*k-2)-((1-alp0)*alp1(i)*(1+alp0))*x(3*k-3)+((alp0).^2)+alp0-((alp1(i)*(1+alp0)).^2))*x(3*k-4)-((alp1(i)*(1+alp0))*alp0)*x(3*k-5)-((alp0).^2)*x(3*k-6)+(((alp1(i)*(1+alp0)).^3)-3*alp1(i)*(1+alp0)*alp0)*y(3*k-3)-((alp0).^3)*y(3*k-6);
        y(3*k+1)=(1-alp0)/2*(alp1(i)*(1+alp0)*x(3*k)+((alp0)^2)+alp0-((alp1(i)*(1+alp0))^2))*x(3*k-3)+x(3*k+1)-(1-alp0)*alp1(i)*(1+alp0)*x(3*k-2)-((alp0)^2)*x(3*k-5)+(((alp1(i)*(1+alp0))^2)-alp0-1)*x(3*k-1)-(alp1(i)*(1+alp0))*alp0*x(3*k-4)+((alp1(i)*(1+alp0))^3)*y(3*k-2)-3*(alp1(i)*(1+alp0))*alp0*y(3*k-2)-((alp0)^3)*y(3*k-5);
    end
end

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$y(3*k+2)=(1-\text{alp0})/2*(((\text{alp1}(i)*(1+\text{alp0}))^2)-\text{alp0}-1)*x(3*k)-\text{alp1}(i)*(1+\text{alp0})*\text{alp0}*x(3*k-3)+\text{alp1}(i)*(1+\text{alp0})*x(3*k+1)+((\text{alp0})^2)+\text{alp0}-((\text{alp1}(i)*(1+\text{alp0}))^2))*x(3*k-2)+x(3*k+2)-\text{alp1}(i)*(1+\text{alp0})*(1-\text{alp0})*x(3*k-1)-((\text{alp0})^2)*x(3*k-4))+\text{alp1}(i)*(1+\text{alp0})*(((\text{alp1}(i)*(1+\text{alp0}))^2)-3*\text{alp0})*y(3*k-1)-((\text{alp0})^3)*y(3*k-4);$$

$$w1(3*k) = (1-\text{alp0})/2*((1+\text{alp0})*x(3*k-1)+2*\text{alp1}(i)*(1+\text{alp0})*(1+\text{alp0})*x(3*k-2)-(1-\text{alp0})*(1+\text{alp0})*x(3*k-3)-2*\text{alp1}(i)*(1+\text{alp0})*(1+\text{alp0})*x(3*k-4)-(1+\text{alp0})*\text{alp0}*x(3*k-5))+3*((\text{alp1}(i)*(1+\text{alp0}))^2)*(1+\text{alp0})-3*\text{alp0}*(1+\text{alp0}))*y(3*k-3+0)+(((\text{alp1}(i)*(1+\text{alp0}))^3)-3*\text{alp1}(i)*(1+\text{alp0})*\text{alp0})*w1(3*k-3+0)-((\text{alp0})^3)*w1(3*k-6);$$

$$w1(3*k+1) = ((1-\text{alp0})/2*((1+\text{alp0})*x(3*k)-2*\text{alp1}(i)*(1+\text{alp0})*(1+\text{alp0})*x(3*k-3)-(1-\text{alp0})*((\text{alp0})^2))*x(3*k-2)+2*\text{alp1}(i)*(1+\text{alp0})*(1+\text{alp0})*x(3*k-1)-(1+\text{alp0})*\text{alp0}*x(3*k-4))+3*(1+\text{alp0})*((\text{alp1}(i)*(1+\text{alp0}))^2)-3*\text{alp0}*(1+\text{alp0}))*y(3*k-2))+((\text{alp1}(i)*(1+\text{alp0}))^3)*w1(3*k-2)-3*\text{alp0}*\text{alp1}(i)*(1+\text{alp0})*w1(3*k-2)-((\text{alp0})^3)*w1(3*k-5);$$

$$w1(3*k+2) = ((1-\text{alp0})/2*(2*\text{alp1}(i)*(1+\text{alp0})*(1+\text{alp0})*x(3*k)-(1+\text{alp0})*\text{alp0}*x(3*k-3)+(1+\text{alp0})*x(3*k+1)-2*\text{alp1}(i)*(1+\text{alp0})*(1+\text{alp0})*x(3*k-2)-(1-((\text{alp0})^2))*x(3*k-1))+3*(1+\text{alp0})*((\text{alp1}(i)*(1+\text{alp0}))^2)-3*\text{alp0}*(1+\text{alp0}))*y(3*k-1))+((\text{alp1}(i)*(1+\text{alp0}))^3)*w1(3*k-1)-3*\text{alp0}*\text{alp1}(i)*(1+\text{alp0})*w1(3*k-1)-((\text{alp0})^3)*w1(3*k-4);$$

$$\text{malp1}(i) = 1/6*(y(k)*w1(k)+y(k+3)*w1(k+3)+y(k+6)*w1(k+6)+y(k+9)*w1(k+9)+y(k+12)*w1(k+12));$$

end

end

plot(alp1,malp1,'b');

grid;

xlabel('alpha(k+1)');

ylabel('malp');

title('fig')

hold on

clear

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.1.10 โปรแกรมแสดงคุณสมบัติการเปลี่ยนแปลงของ $\alpha_1(k)$ กรณี $L = 3, l = 1$

```

clc
w=-1;
while(w<0|w>180)
    w=input('w(0 to 180)\n(W=)');
end
n=500;
k=0:n-1;
N=1000;
data1=cos(((pi*w/180)*k)+randn(1,1))+0.1*randn(1,length(k));
alp0=0.8;
u=0.00005;
x=zeros(n+8,1);
x(9:n+8)=data1;
y=zeros(n+8,1);
w1=zeros(n+8,1);
alp1=linspace(0,1,N);
malp1=zeros(N,1);
for i=1:N
    for k=3:((n+8)-mod(n,3))/3
        y(3*k) =(1-alp0)/2*(x(3*k)+alp1(i)*(1+alp0)*x(3*k-1)+(((alp1(i)*(1+alp0)).^2)-alp0-1)*x(3*k-2)-((1-alp0)*alp1(i)*(1+alp0))*x(3*k-3)+(((alp0).^2)+alp0-((alp1(i)*(1+alp0)).^2))*x(3*k-4)-((alp1(i)*(1+alp0))*alp0)*x(3*k-5)-((alp0).^2)*x(3*k-6)+(((alp1(i)*(1+alp0)).^3)-3*alp1(i)*(1+alp0)*alp0)*y(3*k-3)-((alp0).^3)*y(3*k-6);
        y(3*k+1)=(1-alp0)/2*(alp1(i)*(1+alp0)*x(3*k)+(((alp0)^2)+alp0-((alp1(i)*(1+alp0))^2))*x(3*k-3)+x(3*k+1)-(1-alp0)*alp1(i)*(1+alp0)*x(3*k-2)-((alp0)^2)*x(3*k-5)+(((alp1(i)*(1+alp0))^2)-alp0-1)*x(3*k-1)-(alp1(i)*(1+alp0))*alp0*x(3*k-4)+((alp1(i)*(1+alp0))^3)*y(3*k-2)-3*(alp1(i)*(1+alp0))*alp0*y(3*k-2)-((alp0)^3)*y(3*k-5);
    end
end

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$y(3*k+2)=(1-\text{alp0})/2*(((\text{alp1}(i)*(1+\text{alp0}))^2)-\text{alp0}-1)*x(3*k)-\text{alp1}(i)*(1+\text{alp0})*\text{alp0}*x(3*k-3)+\text{alp1}(i)*(1+\text{alp0})*x(3*k+1)+((\text{alp0})^2)+\text{alp0}-((\text{alp1}(i)*(1+\text{alp0}))^2))*x(3*k-2)+x(3*k+2)-\text{alp1}(i)*(1+\text{alp0})*(1-\text{alp0})*x(3*k-1)-((\text{alp0})^2)*x(3*k-4))+\text{alp1}(i)*(1+\text{alp0})*(((\text{alp1}(i)*(1+\text{alp0}))^2)-3*\text{alp0})*y(3*k-1)-((\text{alp0})^3)*y(3*k-4);$$

$$w1(3*k)=(1-\text{alp0})/2*((1+\text{alp0})*x(3*k-1)+2*\text{alp1}(i)*(1+\text{alp0})*(1+\text{alp0})*x(3*k-2)-(1-\text{alp0})*(1+\text{alp0})*x(3*k-3)-2*\text{alp1}(i)*(1+\text{alp0})*(1+\text{alp0})*x(3*k-4)-(1+\text{alp0})*\text{alp0}*x(3*k-5))+3*((\text{alp1}(i)*(1+\text{alp0}))^2)*(1+\text{alp0})-3*\text{alp0}*(1+\text{alp0}))*y(3*k-3+0)+(((\text{alp1}(i)*(1+\text{alp0}))^3)-3*\text{alp1}(i)*(1+\text{alp0})*\text{alp0})*w1(3*k-3+0)-((\text{alp0})^3)*w1(3*k-6);$$

$$w1(3*k+1)=((1-\text{alp0})/2*((1+\text{alp0})*x(3*k)-2*\text{alp1}(i)*(1+\text{alp0})*(1+\text{alp0})*x(3*k-3)-(1-((\text{alp0})^2))*x(3*k-2)+2*\text{alp1}(i)*(1+\text{alp0})*(1+\text{alp0})*x(3*k-1)-(1+\text{alp0})*\text{alp0}*x(3*k-4))+3*(1+\text{alp0})*((\text{alp1}(i)*(1+\text{alp0}))^2)-3*\text{alp0}*(1+\text{alp0}))*y(3*k-2))+((\text{alp1}(i)*(1+\text{alp0}))^3)*w1(3*k-2)-3*\text{alp0}*\text{alp1}(i)*(1+\text{alp0})*w1(3*k-2)-((\text{alp0})^3)*w1(3*k-5);$$

$$w1(3*k+2)=((1-\text{alp0})/2*(2*\text{alp1}(i)*(1+\text{alp0})*(1+\text{alp0})*x(3*k)-(1+\text{alp0})*\text{alp0}*x(3*k-3)+(1+\text{alp0})*x(3*k+1)-2*\text{alp1}(i)*(1+\text{alp0})*(1+\text{alp0})*x(3*k-2)-(1-((\text{alp0})^2))*x(3*k-1))+3*(1+\text{alp0})*((\text{alp1}(i)*(1+\text{alp0}))^2)-3*\text{alp0}*(1+\text{alp0}))*y(3*k-1))+((\text{alp1}(i)*(1+\text{alp0}))^3)*w1(3*k-1)-3*\text{alp0}*\text{alp1}(i)*(1+\text{alp0})*w1(3*k-1)-((\text{alp0})^3)*w1(3*k-4);$$

$$\text{malp1}(i) = 1/6*(y(k-2)*w1(k-2)+y(k+1)*w1(k+1)+y(k+4)*w1(k+4)+y(k+7)*w1(k+7)+y(k+10)*w1(k+10)+y(k+13)*w1(k+13));$$

end

end

plot(alp1,malp1,'g');

grid;

xlabel('alpha(k+1)');

ylabel('malp');

title('fig')

hold on

clear

### 3.1.11 โปรแกรมแสดงคุณสมบัติการเปลี่ยนแปลงของ $\alpha_1(k)$ กรณี $L = 3, l = 2$

```

clc
w=-1;
while(w<0|w>180)
    w=input('w(0 to 180)\n(W=)');
end
n=500;
k=0:n-1;
N=1000;
data1=cos(((pi*w/180)*k)+randn(1,1))+0.1*randn(1,length(k));
alp0=0.8;
u=0.00005;
x=zeros(n+8,1);
x(9:n+8)=data1;
y=zeros(n+8,1);
w1=zeros(n+8,1);
alp1=linspace(0,1,N);
malp1=zeros(N,1);
for i=1:N
    for k=3:(n+8)-mod(n,3)/3
        y(3*k) =(1-alp0)/2*(x(3*k)+alp1(i)*(1+alp0)*x(3*k-1)+(((alp1(i)*(1+alp0)).^2)-alp0-1)*x(3*k-2)-((1-alp0)*alp1(i)*(1+alp0))*x(3*k-3)+((alp0).^2)+alp0-((alp1(i)*(1+alp0)).^2))*x(3*k-4)-((alp1(i)*(1+alp0))*alp0)*x(3*k-5)-((alp0).^2)*x(3*k-6)+(((alp1(i)*(1+alp0)).^3)-3*alp1(i)*(1+alp0)*alp0)*y(3*k-3)-((alp0).^3)*y(3*k-6);
        y(3*k+1)=(1-alp0)/2*(alp1(i)*(1+alp0)*x(3*k)+((alp0)^2)+alp0-((alp1(i)*(1+alp0))^2))*x(3*k-3)+x(3*k+1)-(1-alp0)*alp1(i)*(1+alp0)*x(3*k-2)-((alp0)^2)*x(3*k-5)+(((alp1(i)*(1+alp0))^2)-alp0-1)*x(3*k-1)-(alp1(i)*(1+alp0))*alp0*x(3*k-4)+((alp1(i)*(1+alp0))^3)*y(3*k-2)-3*(alp1(i)*(1+alp0))*alp0*y(3*k-2)-((alp0)^3)*y(3*k-5);
    end
end

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$y(3*k+2) = (1-\text{alp0})/2 * (((\text{alp1}(i)*(1+\text{alp0}))^2 - \text{alp0} - 1) * x(3*k) - \text{alp1}(i)*(1+\text{alp0}) * \text{alp0} * x(3*k-3) + \text{alp1}(i)*(1+\text{alp0}) * x(3*k+1) + ((\text{alp0})^2 + \text{alp0} - ((\text{alp1}(i)*(1+\text{alp0}))^2)) * x(3*k-2) + x(3*k+2) - \text{alp1}(i)*(1+\text{alp0}) * (1-\text{alp0}) * x(3*k-1) - ((\text{alp0})^2) * x(3*k-4)) + \text{alp1}(i)*(1+\text{alp0}) * (((\text{alp1}(i)*(1+\text{alp0}))^2) - 3 * \text{alp0}) * y(3*k-1) - ((\text{alp0})^3) * y(3*k-4));$$

$$w1(3*k) = (1-\text{alp0})/2 * ((1+\text{alp0}) * x(3*k-1) + 2 * \text{alp1}(i)*(1+\text{alp0}) * (1+\text{alp0}) * x(3*k-2) - (1-\text{alp0}) * (1+\text{alp0}) * x(3*k-3) - 2 * \text{alp1}(i)*(1+\text{alp0}) * (1+\text{alp0}) * x(3*k-4) - (1+\text{alp0}) * \text{alp0} * x(3*k-5)) + (3 * ((\text{alp1}(i)*(1+\text{alp0}))^2) * (1+\text{alp0}) - 3 * \text{alp0} * (1+\text{alp0})) * y(3*k-3+0) + (((\text{alp1}(i)*(1+\text{alp0}))^3) - 3 * \text{alp1}(i)*(1+\text{alp0}) * \text{alp0}) * w1(3*k-3+0) - ((\text{alp0})^3) * w1(3*k-6));$$

$$w1(3*k+1) = ((1-\text{alp0})/2 * ((1+\text{alp0}) * x(3*k) - 2 * \text{alp1}(i)*(1+\text{alp0}) * (1+\text{alp0}) * x(3*k-3) - (1 - ((\text{alp0})^2)) * x(3*k-2) + 2 * \text{alp1}(i)*(1+\text{alp0}) * (1+\text{alp0}) * x(3*k-1) - (1+\text{alp0}) * \text{alp0} * x(3*k-4)) + (3 * (1+\text{alp0}) * ((\text{alp1}(i)*(1+\text{alp0}))^2) - 3 * \text{alp0} * (1+\text{alp0})) * y(3*k-2)) + ((\text{alp1}(i)*(1+\text{alp0}))^3) * w1(3*k-2) - 3 * \text{alp0} * \text{alp1}(i)*(1+\text{alp0}) * w1(3*k-2) - ((\text{alp0})^3) * w1(3*k-5));$$

$$w1(3*k+2) = ((1-\text{alp0})/2 * (2 * \text{alp1}(i)*(1+\text{alp0}) * (1+\text{alp0}) * x(3*k) - (1+\text{alp0}) * \text{alp0} * x(3*k-3) + (1+\text{alp0}) * x(3*k+1) - 2 * \text{alp1}(i)*(1+\text{alp0}) * (1+\text{alp0}) * x(3*k-2) - (1 - ((\text{alp0})^2)) * x(3*k-1)) + (3 * (1+\text{alp0}) * ((\text{alp1}(i)*(1+\text{alp0}))^2) - 3 * \text{alp0} * (1+\text{alp0})) * y(3*k-1)) + ((\text{alp1}(i)*(1+\text{alp0}))^3) * w1(3*k-1) - 3 * \text{alp0} * \text{alp1}(i)*(1+\text{alp0}) * w1(3*k-1) - ((\text{alp0})^3) * w1(3*k-4));$$

$$\text{malp1}(i) = 1/6 * (y(k-1) * w1(k-1) + y(k+2) * w1(k+2) + y(k+5) * w1(k+5) + y(k+8) * w1(k+8) + y(k+11) * w1(k+11) + y(k+14) * w1(k+14));$$

end

end

plot(alp1,malp1,'g');

grid;

xlabel('alpha(k+1)');

ylabel('malp1');

title('fig')

hold on

clear

### 3.1.12 โปรแกรมแสดงคุณสมบัติการเปลี่ยนแปลงของ $\alpha_1(k)$ กรณี $L = 3$ โครงสร้างแบบใหม่

```
clc
```

```
w=-1;
```

```
while(w<0|w>180)
```

```
    w=input('w(0 to 180)\n(W=)');
```

```
end
```

```
n=500;
```

```
k=0:n-1;
```

```
N=1000;
```

```
data1=cos(((pi*w/180)*k)+randn(1,1))+0.1*randn(1,length(k));
```

```
alp0=0.85;
```

```
u=0.00005;
```

```
x=zeros(n+8,1);
```

```
x(9:n+8)=data1;
```

```
y=zeros(n+8,1);
```

```
w1=zeros(n+8,1);
```

```
alp1=linspace(0,1,N);
```

```
malp1=zeros(N,1);
```

```
for i=1:N
```

```
    for k=3:((n+8)-mod(n,3))/3
```

```
        y(3*k) = (1-alp0)/2*(x(3*k)+alp1(i)*(1+alp0)*x(3*k-1)+(((alp1(i)*(1+alp0)).^2)-alp0-1)*x(3*k-2)-((1-alp0)*alp1(i)*(1+alp0))*x(3*k-3)+(((alp0).^2)+alp0-((alp1(i)*(1+alp0)).^2))*x(3*k-4)-((alp1(i)*(1+alp0))*alp0)*x(3*k-5)-((alp0).^2)*x(3*k-6)+(((alp1(i)*(1+alp0)).^3)-3*alp1(i)*(1+alp0)*alp0)*y(3*k-3)-((alp0).^3)*y(3*k-6);
```

```
        y(3*k+1)=(1-alp0)/2*(alp1(i)*(1+alp0)*x(3*k)+(((alp0)^2)+alp0-((alp1(i)*(1+alp0))^2))*x(3*k-3)+x(3*k+1)-(1-alp0)*alp1(i)*(1+alp0)*x(3*k-2)-((alp0)^2)*x(3*k-5)+(((alp1(i)*(1+alp0))^2)-alp0-1)*x(3*k-1)-(alp1(i)*(1+alp0))*alp0*x(3*k-4)+((alp1(i)*(1+alp0))^3)*y(3*k-2)-3*(alp1(i)*(1+alp0))*alp0*y(3*k-2)-((alp0)^3)*y(3*k-5);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$y(3^*k+2)=(1-\text{alp0})/2^*((\text{alp1}(i)^*(1+\text{alp0}))^2)-\text{alp0}-1)^*x(3^*k)-\text{alp1}(i)^*(1+\text{alp0})^*\text{alp0}^*x(3^*k-3)+\text{alp1}(i)^*(1+\text{alp0})^*x(3^*k+1)+((\text{alp0})^2)+\text{alp0}-((\text{alp1}(i)^*(1+\text{alp0}))^2))^*x(3^*k-2)+x(3^*k+2)-\text{alp1}(i)^*(1+\text{alp0})^*(1-\text{alp0})^*x(3^*k-1)-((\text{alp0})^2)^*x(3^*k-4)+\text{alp1}(i)^*(1+\text{alp0})^*((\text{alp1}(i)^*(1+\text{alp0}))^2)-3^*\text{alp0}^*y(3^*k-1)-((\text{alp0})^3)^*y(3^*k-4);$$

$$w1(3^*k)=(1-\text{alp0})/2^*((1+\text{alp0})^*x(3^*k-1)+2^*\text{alp1}(i)^*(1+\text{alp0})^*(1+\text{alp0})^*x(3^*k-2)-(1-\text{alp0})^*(1+\text{alp0})^*x(3^*k-3)-2^*\text{alp1}(i)^*(1+\text{alp0})^*(1+\text{alp0})^*x(3^*k-4)-(1+\text{alp0})^*\text{alp0}^*x(3^*k-5))+3^*((\text{alp1}(i)^*(1+\text{alp0}))^2)^*(1+\text{alp0})-3^*\text{alp0}^*(1+\text{alp0})^*y(3^*k-3+0)+((\text{alp1}(i)^*(1+\text{alp0}))^3)-3^*\text{alp1}(i)^*(1+\text{alp0})^*\text{alp0}^*w1(3^*k-3+0)-((\text{alp0})^3)^*w1(3^*k-6);$$

$$w1(3^*k+1)=((1-\text{alp0})/2^*((1+\text{alp0})^*x(3^*k)-2^*\text{alp1}(i)^*(1+\text{alp0})^*(1+\text{alp0})^*x(3^*k-3)-(1-\text{alp0})^2)^*x(3^*k-2)+2^*\text{alp1}(i)^*(1+\text{alp0})^*(1+\text{alp0})^*x(3^*k-1)-(1+\text{alp0})^*\text{alp0}^*x(3^*k-4))+3^*(1+\text{alp0})^*((\text{alp1}(i)^*(1+\text{alp0}))^2)-3^*\text{alp0}^*(1+\text{alp0})^*y(3^*k-2))+((\text{alp1}(i)^*(1+\text{alp0}))^3)^*w1(3^*k-2)-3^*\text{alp0}^*\text{alp1}(i)^*(1+\text{alp0})^*w1(3^*k-2)-((\text{alp0})^3)^*w1(3^*k-5);$$

$$w1(3^*k+2)=((1-\text{alp0})/2^*(2^*\text{alp1}(i)^*(1+\text{alp0})^*(1+\text{alp0})^*x(3^*k)-(1+\text{alp0})^*\text{alp0}^*x(3^*k-3)+(1+\text{alp0})^*x(3^*k+1)-2^*\text{alp1}(i)^*(1+\text{alp0})^*(1+\text{alp0})^*x(3^*k-2)-(1-\text{alp0})^2)^*x(3^*k-1))+3^*(1+\text{alp0})^*((\text{alp1}(i)^*(1+\text{alp0}))^2)-3^*\text{alp0}^*(1+\text{alp0})^*y(3^*k-1))+((\text{alp1}(i)^*(1+\text{alp0}))^3)^*w1(3^*k-1)-3^*\text{alp0}^*\text{alp1}(i)^*(1+\text{alp0})^*w1(3^*k-1)-((\text{alp0})^3)^*w1(3^*k-4);$$

$$\text{malp1}(i)=1/6^*(y(k-2)^*w1(k-2)+y(k-1)^*w1(k-1)+y(k)^*w1(k)+y(k+1)^*w1(k+1)+y(k+2)^*w1(k+2)+y(k+3)^*w1(k+3)+y(k+4)^*w1(k+4)+y(k+5)^*w1(k+5)+y(k+6)^*w1(k+6)+y(k+7)^*w1(k+7)+y(k+8)^*w1(k+8)+y(k+9)^*w1(k+9)+y(k+10)^*w1(k+10)+y(k+11)^*w1(k+11)+y(k+12)^*w1(k+12)+y(k+13)^*w1(k+13)+y(k+14)^*w1(k+14)+y(k+15)^*w1(k+15));$$

end

end

plot(alp1,malp1,'r');

grid;

xlabel('alpha(k+1)');ylabel('malp');

title('fig')

hold on

clear

### 3.1.13 โปรแกรมแสดงคุณสมบัติของ $\alpha_1(k)$ เทียบกับ $k$ กรณี $L = 2$

```

clc
w=-1;
while(w<0|w>180)
    w=input('w(0 to 180)\n(W=)');
end
n=500000;
k=0:n-1;
data1=cos(((pi*w/180)*k)+randn(1,1))+0.1*randn(1,n);
alp0=0.8;
u=0.00005;
x=zeros(n+8,1);
x(4:n+3)=data1;
y=zeros(n+3,1);
w1=zeros(n+3,1);
alp1=zeros(n+3,1);
for k=3:((n+3)-mod(n,2))/2
    y(2*k) =(1-alp0)/2*(x(2*k)+alp1(k)*(1+alp0)*x(2*k-1)-(1-alp0)*x(2*k-2)-alp1(k)*(1+alp0)*x(2*k-3)-alp0*x
(2*k-4))+(((alp1(k)*(1+alp0))^2)+2*alp0)*y(2*k-2)-((alp0)^2)*y(2*k-4);
    y(2*k+1)=(1-alp0)/2*(alp1(k)*(1+alp0)*x(2*k)+x(2*k+1)-(1-alp0)*x(2*k-1)-alp1(k)*(1+alp0)*x(2*k-2)-
alp0*x(2*k-3))+(((alp1(k)*(1+alp0))^2)+2*alp0)*y(2*k-1)-((alp0)^2)*y(2*k-3);
    w1(2*k)=(1-alp0)/2*((1+alp0)*x(2*k-1)-(1+alp0)*x(2*k-3))+2*alp1(k)*((1+alp0)^2)*y(2*k-2)+((alp1(k)*
(1+alp0))^2)*w1(2*k-2)+2*alp0*w1(2*k-2)-((alp0)^2)*w1(2*k-4);
    w1(2*k+1)=(1-alp0)/2*((1+alp0)*x(2*k)-(1+alp0)*x(2*k-2))+2*alp1(k)*((1+alp0)^2)*y(2*k-1)+((alp1(k)*
(1+alp0))^2)*w1(2*k-1)+2*alp0*w1(2*k-1)-((alp0)^2)*w1(2*k-3);
    alp1(k+1)=alp1(k)+u*(y(2*k)*w1(2*k)+y(2*k+1)*w1(2*k+1));
end
plot(alp1, 'b');
grid;
xlabel('k'); ylabel('alp1(k)');
title('fig')
hold on ; clear;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.1.14 โปรแกรมแสดงคุณสมบัติของ $\alpha_{1,l}(k)$ เทียบกับ $k$ โดยใช้โครงสร้างใหม่

```

w=-1;
while(w<0|w>180)
    w=input('w(0 to 180)\n(W=)');
end
n=500000;
k=0:n-1;
data1=cos(((pi*w/180)*k)+randn(1,1))+0.1*randn(1,n);
alp0=0.8;
u=0.00005;
x=zeros(n+8,1);
x(9:n+8)=data1;
y=zeros(n+8,1);
w1=zeros(n+8,1);
alp1=zeros(n+8,1);
for k=3:(n+8)-mod(n,3)/3

    y(3*k) =(1-alp0)/2*(x(3*k)+alp1(k)*(1+alp0)*x(3*k-1)+(((alp1(k)*(1+alp0)).^2)-alp0-1)*x(3*k-2)-((1-alp0)*alp1(k)*(1+alp0))*x(3*k-3)+(((alp0).^2)+alp0-((alp1(k)*(1+alp0)).^2))*x(3*k-4)-((alp1(k)*(1+alp0))*alp0)*x(3*k-5)-((alp0).^2)*x(3*k-6)+(((alp1(k)*(1+alp0)).^3)-3*alp1(k)*(1+alp0)*alp0)*y(3*k-3)-((alp0).^3)*y(3*k-6);

    y(3*k+1)=(1-alp0)/2*(alp1(k)*(1+alp0)*x(3*k)+(((alp0).^2)+alp0-((alp1(k)*(1+alp0)).^2))*x(3*k-3)+x(3*k+1)-(1-alp0)*alp1(k)*(1+alp0)*x(3*k-2)-((alp0).^2)*x(3*k-5)+(((alp1(k)*(1+alp0)).^2)-alp0-1)*x(3*k-1)-(alp1(k)*(1+alp0))*alp0*x(3*k-4))+((alp1(k)*(1+alp0)).^3)*y(3*k-2)-3*(alp1(k)*(1+alp0))*alp0*y(3*k-2)-((alp0).^3)*y(3*k-5);

    y(3*k+2)=(1-alp0)/2*(((alp1(k)*(1+alp0)).^2)-alp0-1)*x(3*k)-alp1(k)*(1+alp0)*alp0*x(3*k-3)+alp1(k)*(1+alp0)*x(3*k+1)+(((alp0).^2)+alp0-((alp1(k)*(1+alp0)).^2))*x(3*k-2)+x(3*k+2)-alp1

```

$$(k)*(1+alp0)*(1-alp0)*x(3*k-1)-((alp0)^2)*x(3*k-4))+alp1(k)*(1+alp0)*(((alp1(k)*(1+alp0))^2)-3*alp0)*y(3*k-1)-((alp0)^3)*y(3*k-4);$$

$$w1(3*k) = (1-alp0)/2*((1+alp0)*x(3*k-1)+2*alp1(k)*(1+alp0)*(1+alp0)*x(3*k-2)-(1-alp0)*(1+alp0)*x(3*k-3)-2*alp1(k)*(1+alp0)*(1+alp0)*x(3*k-4)-(1+alp0)*alp0*x(3*k-5))+3*((alp1(k)*(1+alp0))^2)*(1+alp0)-3*alp0*(1+alp0))*y(3*k-3+0)+(((alp1(k)*(1+alp0))^3)-3*alp1(k)*(1+alp0)*alp0)*w1(3*k-3+0)-((alp0)^3)*w1(3*k-6);$$

$$w1(3*k+1) = ((1-alp0)/2*((1+alp0)*x(3*k)-2*alp1(k)*(1+alp0)*(1+alp0)*x(3*k-3)-(1-((alp0)^2))*x(3*k-2)+2*alp1(k)*(1+alp0)*(1+alp0)*x(3*k-1)-(1+alp0)*alp0*x(3*k-4))+3*(1+alp0)*((alp1(k)*(1+alp0))^2)-3*alp0*(1+alp0))*y(3*k-2))+((alp1(k)*(1+alp0))^3)*w1(3*k-2)-3*alp0*alp1(k)*(1+alp0)*w1(3*k-2)-((alp0)^3)*w1(3*k-5);$$

$$w1(3*k+2) = ((1-alp0)/2*(2*alp1(k)*(1+alp0)*(1+alp0)*x(3*k)-(1+alp0)*alp0*x(3*k-3)+(1+alp0)*x(3*k+1)-2*alp1(k)*(1+alp0)*(1+alp0)*x(3*k-2)-(1-((alp0)^2))*x(3*k-1))+3*(1+alp0)*((alp1(k)*(1+alp0))^2)-3*alp0*(1+alp0))*y(3*k-1))+((alp1(k)*(1+alp0))^3)*w1(3*k-1)-3*alp0*alp1(k)*(1+alp0)*w1(3*k-1)-((alp0)^3)*w1(3*k-4);$$

$$alp1(k+1) = alp1(k)+u*y(3*k)*w1(3*k) +u*y(3*k+1)*w1(3*k+1) +u*y(3*k+2)*w1(3*k+2);$$

end

plot(alp1, 'b');

grid;

xlabel('k');

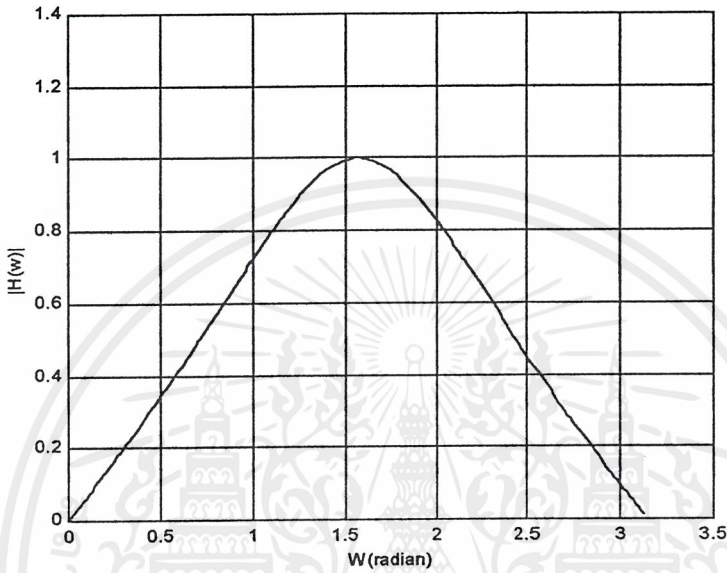
ylabel('alpa1(k)');

hold on

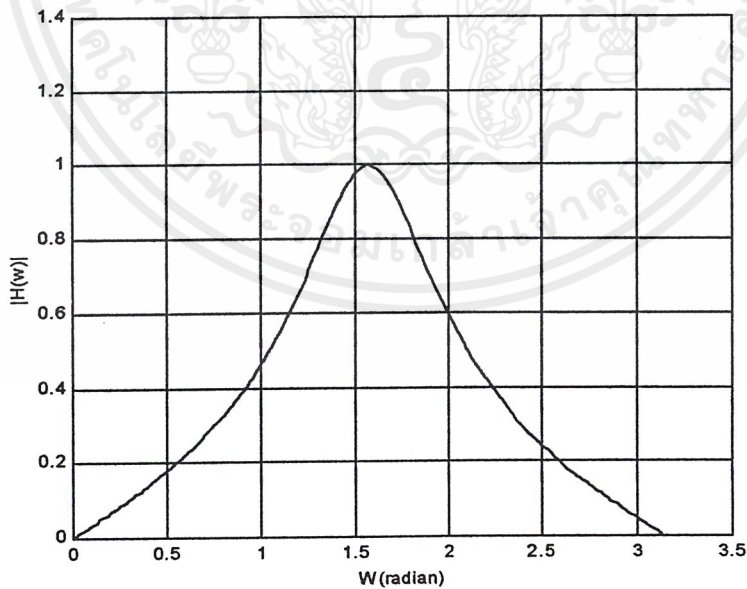
clear

### 3.2 ผลการทดลอง

#### 3.2.1 กราฟแสดงคุณสมบัติของ $|H(\omega)|$ เทียบกับ $\omega$

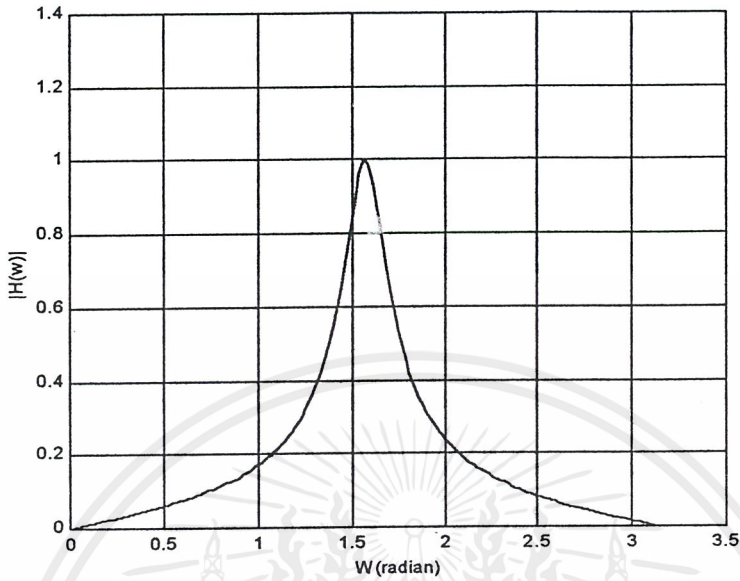


รูปที่ 3.1  $\alpha_0 = 0.2, \omega = \frac{\pi}{2}$

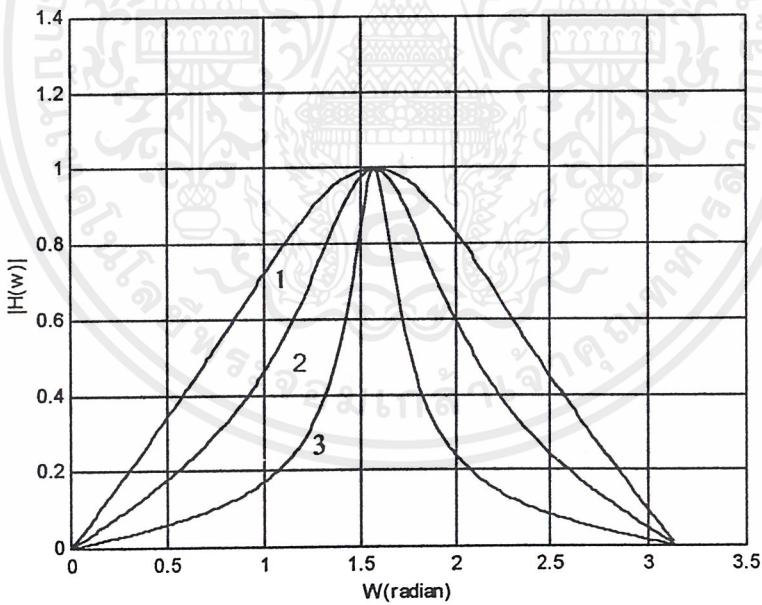


รูปที่ 3.2  $\alpha_0 = 0.5, \omega = \frac{\pi}{2}$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.3  $\alpha_0 = 0.8, \omega = \frac{\pi}{2}$

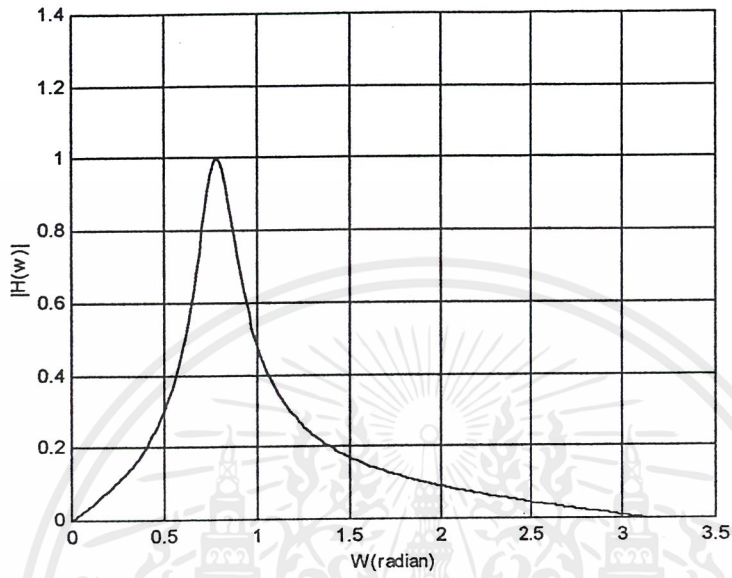


รูปที่ 3.4 1.  $\alpha_0 = 0.2, \omega = \frac{\pi}{2}$

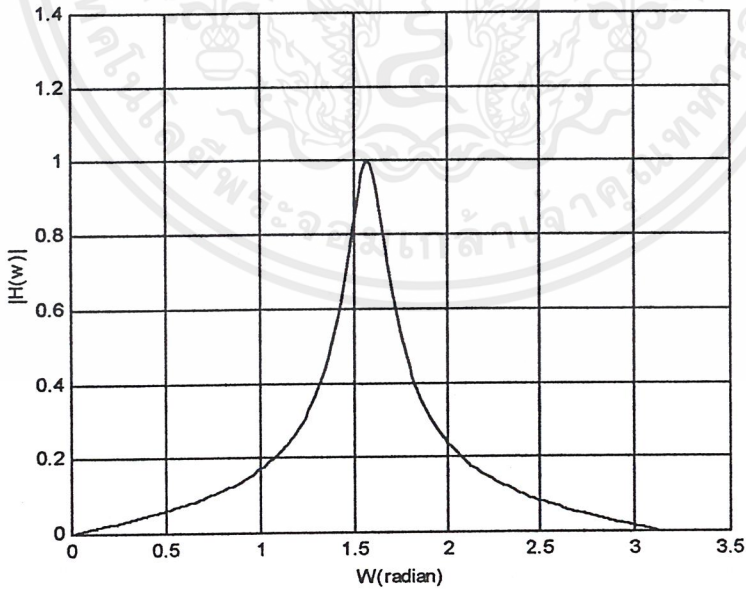
2.  $\alpha_0 = 0.5, \omega = \frac{\pi}{2}$

3.  $\alpha_0 = 0.8, \omega = \frac{\pi}{2}$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

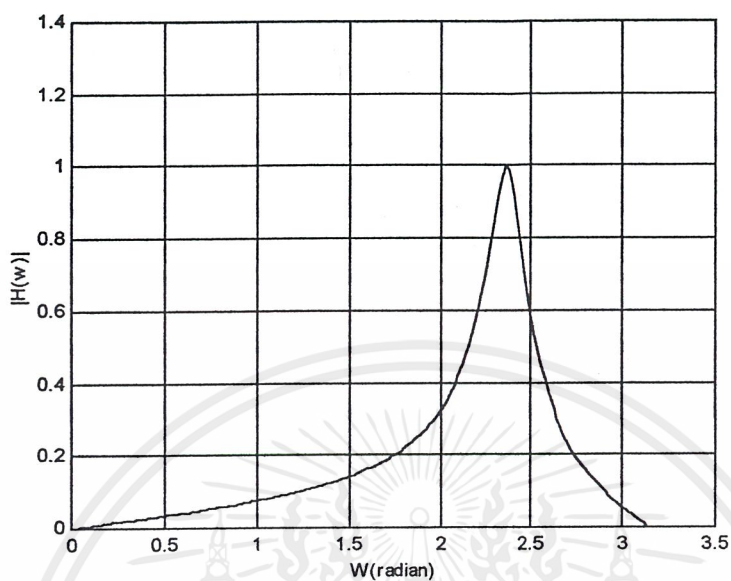


รูปที่ 3.5  $\alpha_0 = 0.8, \omega = \frac{\pi}{4}, \alpha_1 = 0.8$

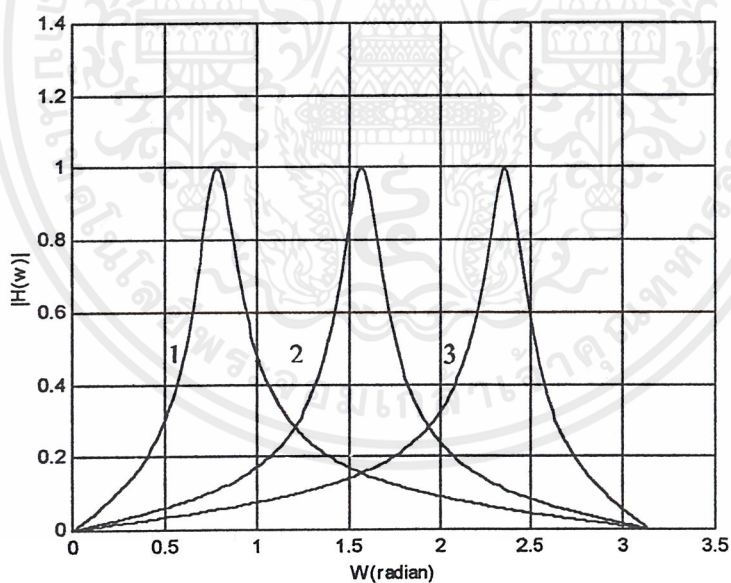


รูปที่ 3.6  $\alpha_0 = 0.8, \omega = \frac{\pi}{2}, \alpha_1 = 0.1$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.7  $\alpha_0 = 0.8$ ,  $\omega = \frac{3\pi}{4}$ ,  $\alpha_1 = -0.8$



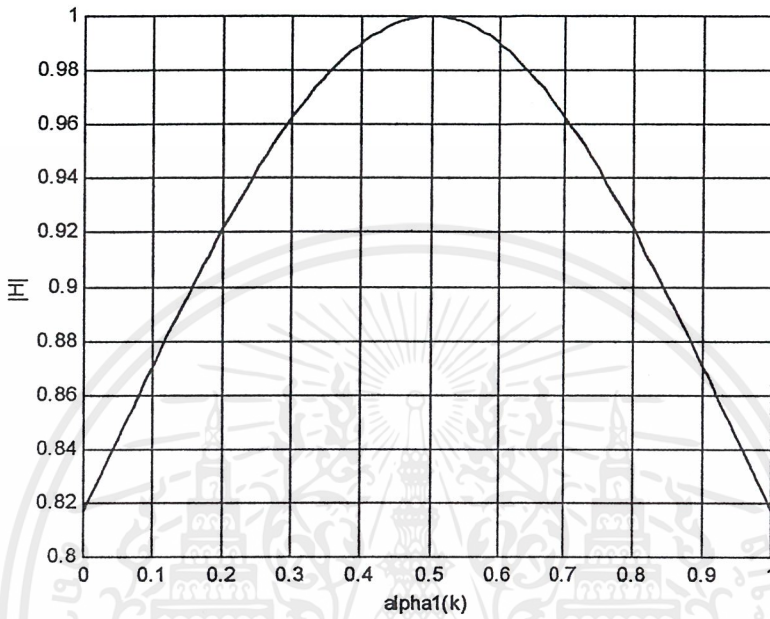
รูปที่ 3.8 1.  $\alpha_0 = 0.8$ ,  $\omega = \frac{\pi}{4}$ ,  $\alpha_1 = 0.8$

2.  $\alpha_0 = 0.8$ ,  $\omega = \frac{\pi}{2}$ ,  $\alpha_1 = 0.1$

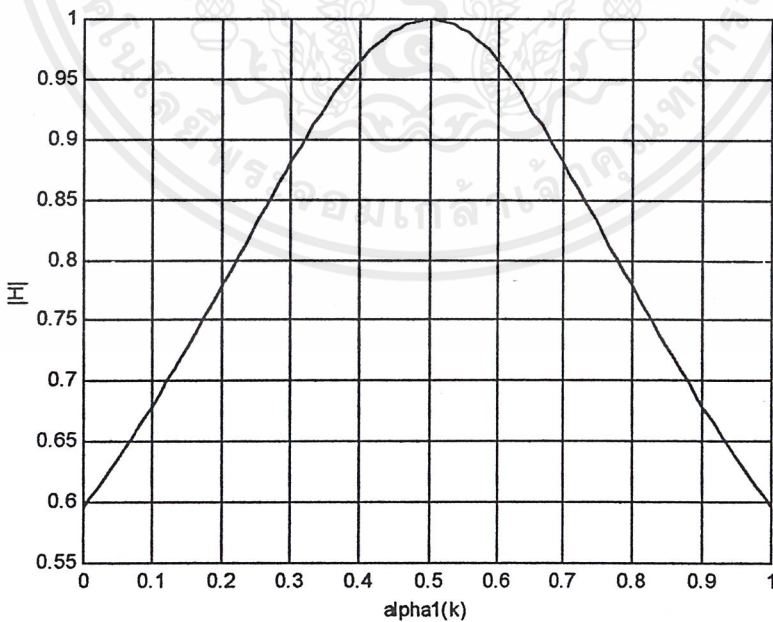
3.  $\alpha_0 = 0.8$ ,  $\omega = \frac{3\pi}{4}$ ,  $\alpha_1 = -0.8$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.2.2 กราฟแสดงคุณสมบัติของ $|H(\omega)|$ เทียบกับ $\alpha_1$

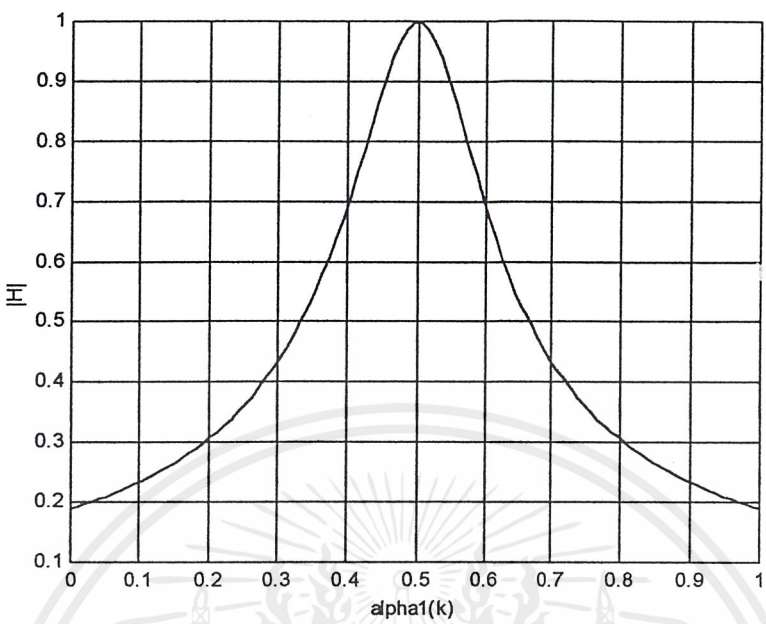


รูปที่ 3.9  $\omega = \frac{\pi}{3}$ ,  $\alpha_0 = 0.1$

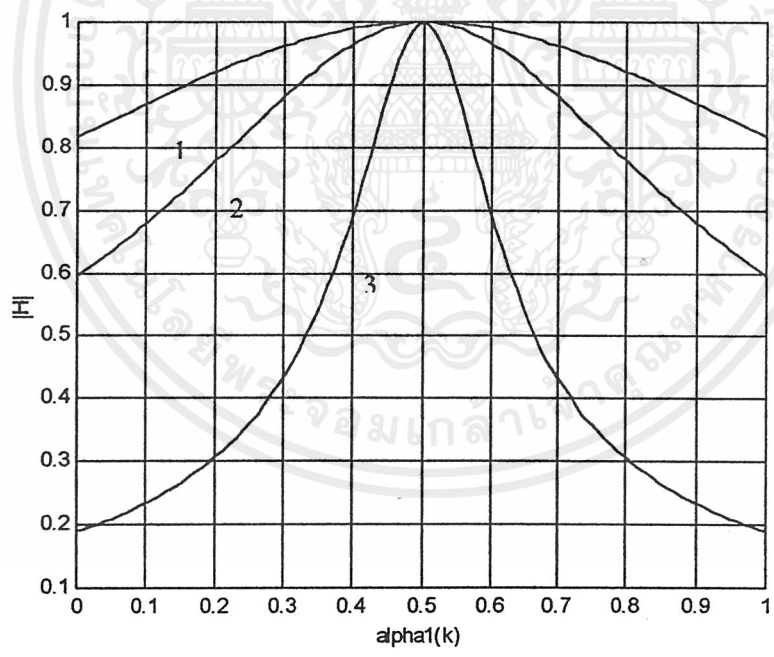


รูปที่ 3.10  $\omega = \frac{\pi}{3}$ ,  $\alpha_0 = 0.4$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.11  $\omega = \frac{\pi}{3}, \alpha_0 = 0.8$

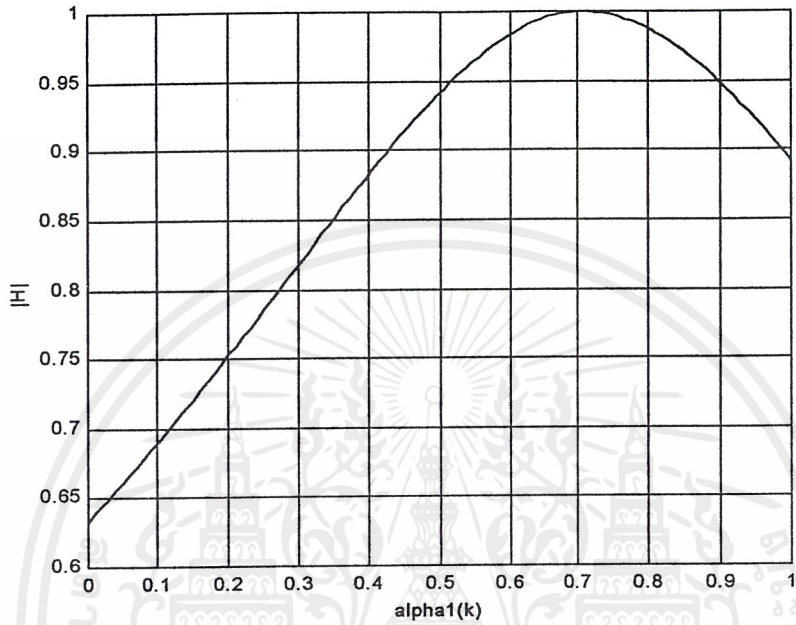


รูปที่ 3.12 1.  $\omega = \frac{\pi}{3}, \alpha_0 = 0.1$

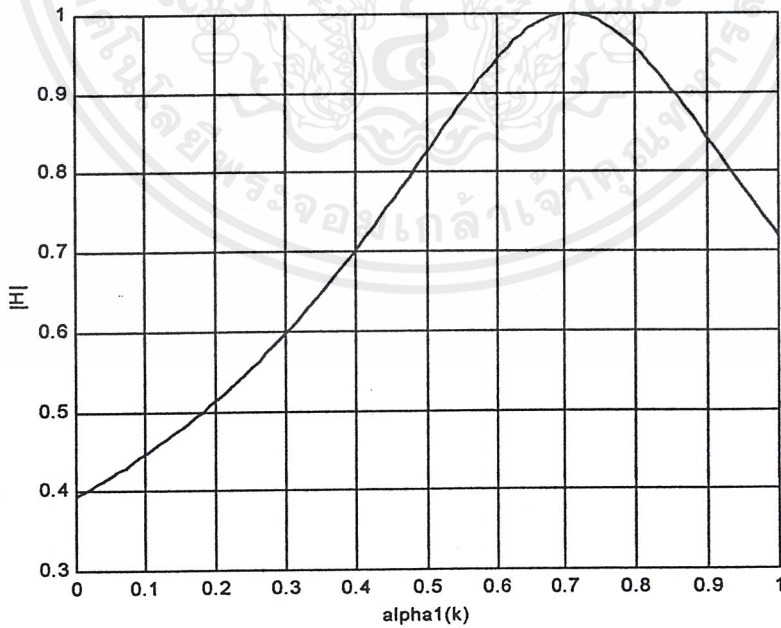
2.  $\omega = \frac{\pi}{3}, \alpha_0 = 0.4$

3.  $\omega = \frac{\pi}{3}, \alpha_0 = 0.8$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

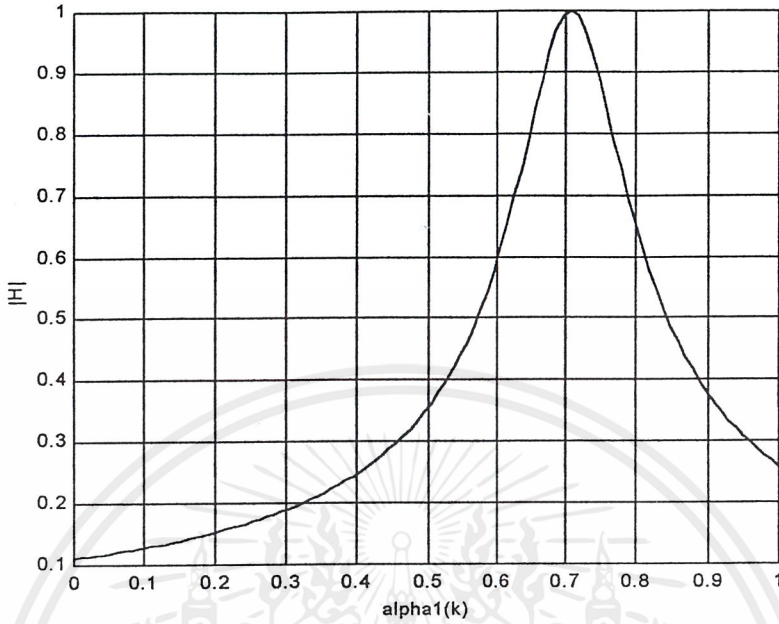


รูปที่ 3.13  $\omega = \frac{\pi}{4}$ ,  $\alpha_0 = 0.1$

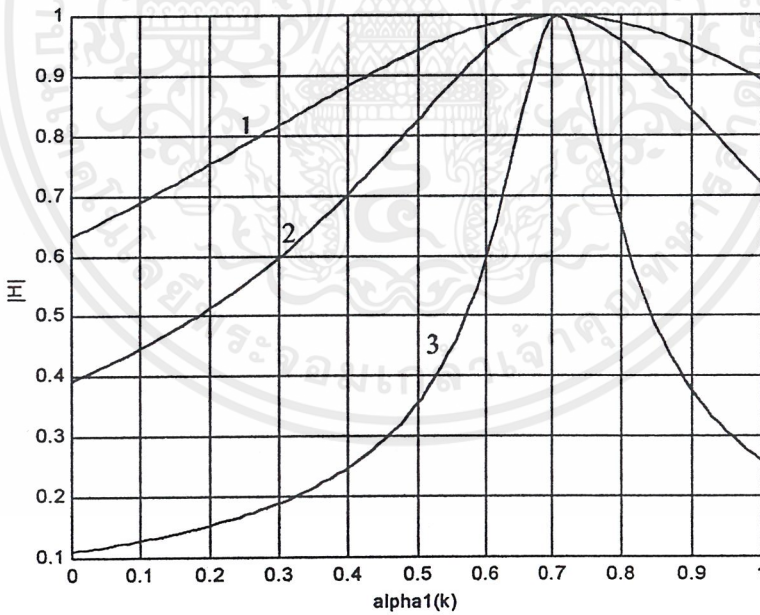


รูปที่ 3.14  $\omega = \frac{\pi}{4}$ ,  $\alpha_0 = 0.4$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.15  $\omega = \frac{\pi}{4}$ ,  $\alpha_0 = 0.8$



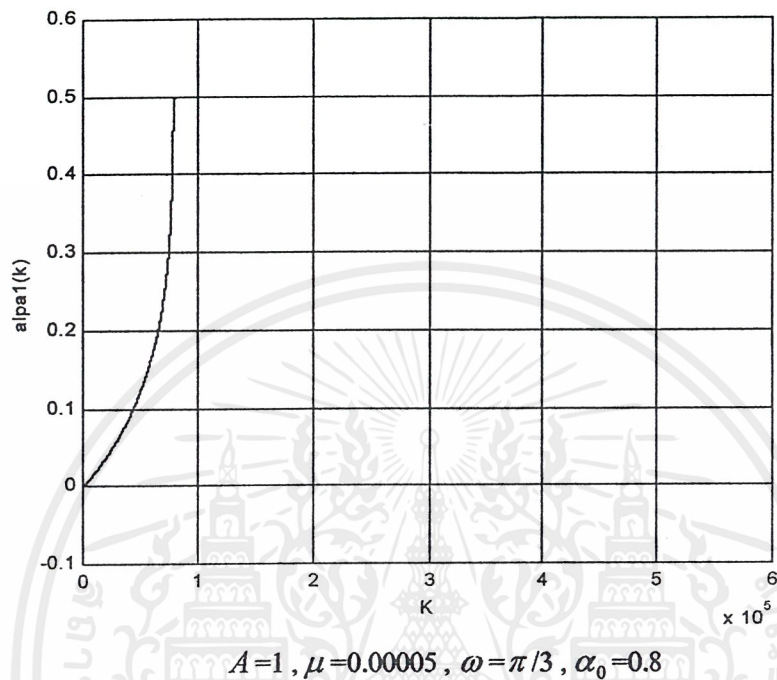
รูปที่ 3.16 1.  $\omega = \frac{\pi}{4}$ ,  $\alpha_0 = 0.1$

2.  $\omega = \frac{\pi}{4}$ ,  $\alpha_0 = 0.4$

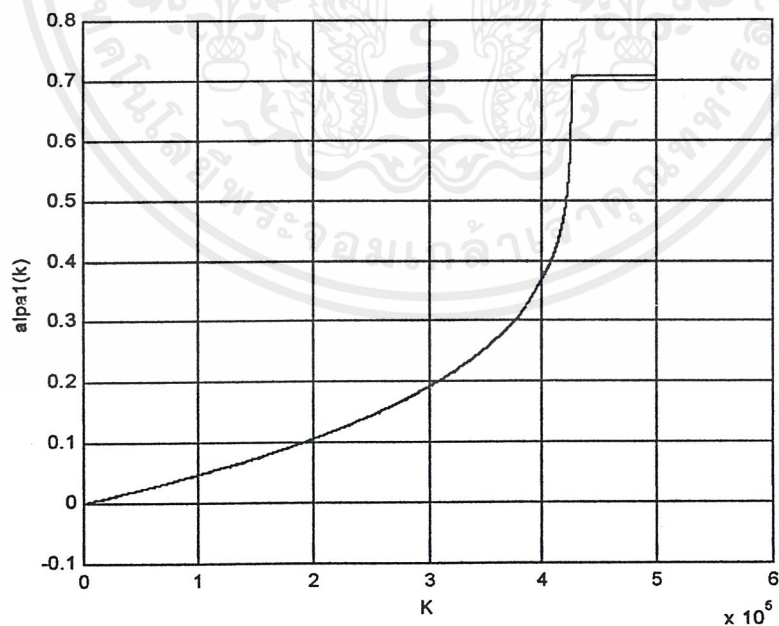
3.  $\omega = \frac{\pi}{4}$ ,  $\alpha_0 = 0.8$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.2.3 กราฟแสดงคุณสมบัติของ $\alpha_1(k)$ เทียบกับ $k$ กรณี $L=1$



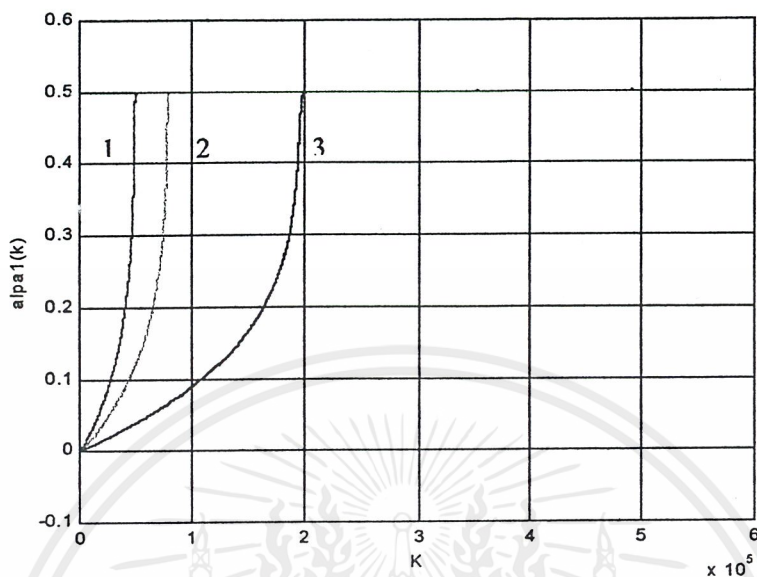
รูปที่ 3.17



$$A=1, \mu=0.00005, \omega=\pi/4, \alpha_0=0.8$$

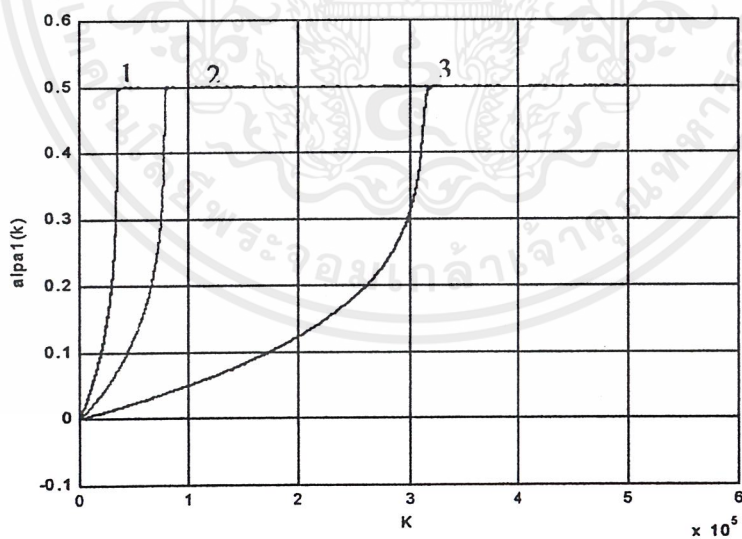
รูปที่ 3.18

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



1.  $A=1, \mu=0.00008, \omega=\pi/3, \alpha_0=0.8$
2.  $A=1, \mu=0.00005, \omega=\pi/3, \alpha_0=0.8$
3.  $A=1, \mu=0.00002, \omega=\pi/3, \alpha_0=0.8$

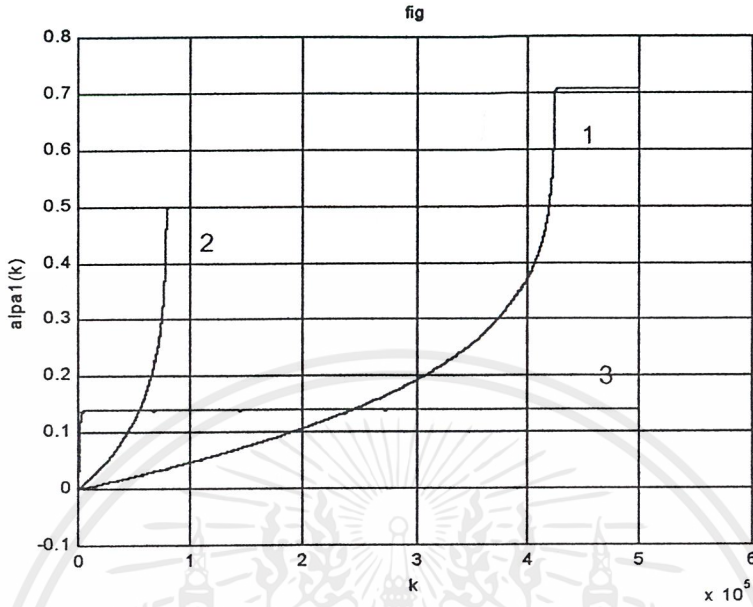
รูปที่ 3.19



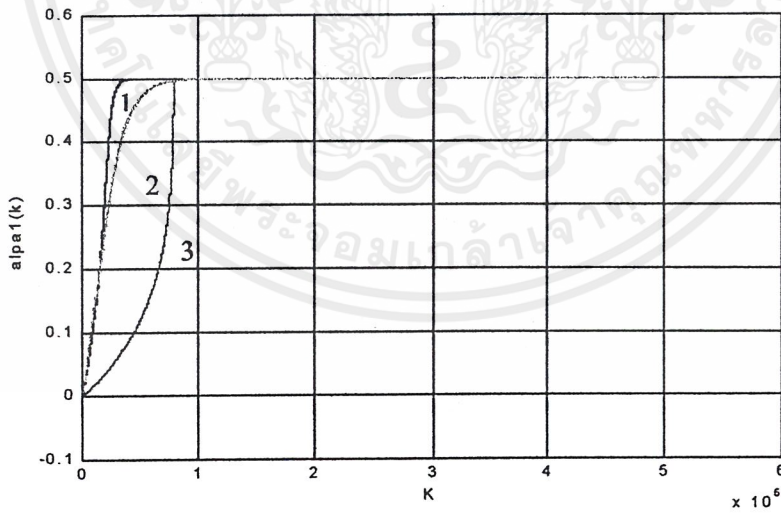
1.  $A=1.5, \mu=0.00005, \omega=\pi/3, \alpha_0=0.8$
2.  $A=1, \mu=0.00005, \omega=\pi/3, \alpha_0=0.8$
3.  $A=0.5, \mu=0.00005, \omega=\pi/3, \alpha_0=0.8$

รูปที่ 3.20

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



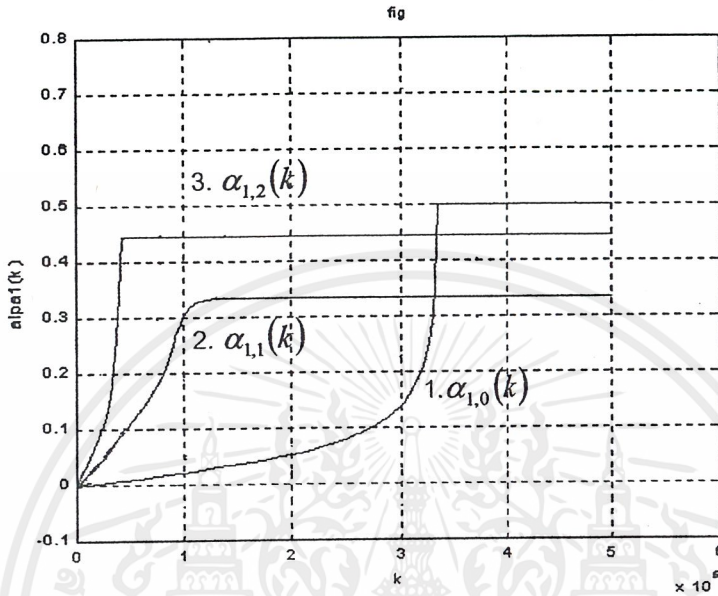
รูปที่ 3.21



รูปที่ 3.22

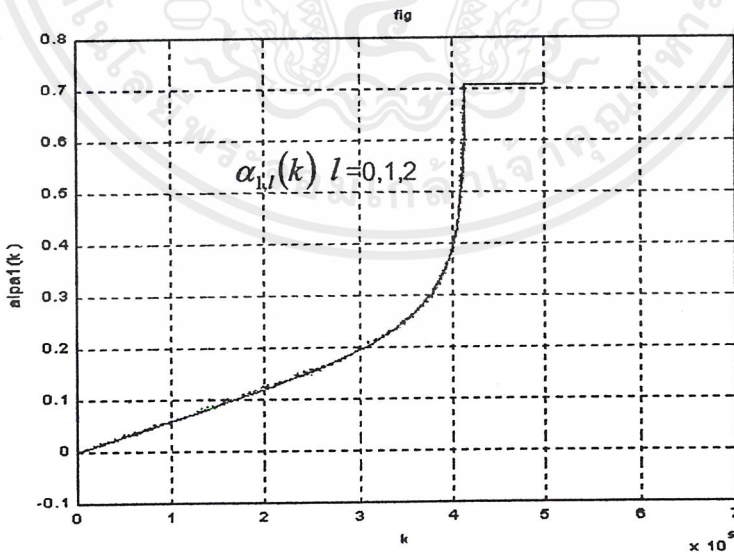
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.5 กราฟแสดงคุณสมบัติของ  $\alpha_1(k)$  เทียบกับ  $k$  กรณีโครงสร้างแบบขนาน ที่  $L = 3$



- 1.  $A=1, \mu=0.00005, \omega=\pi/3, l=0$
- 2.  $A=1, \mu=0.00005, \omega=\pi/3, l=1$
- 3.  $A=1, \mu=0.00005, \omega=\pi/3, l=2$

รูปที่ 3.23

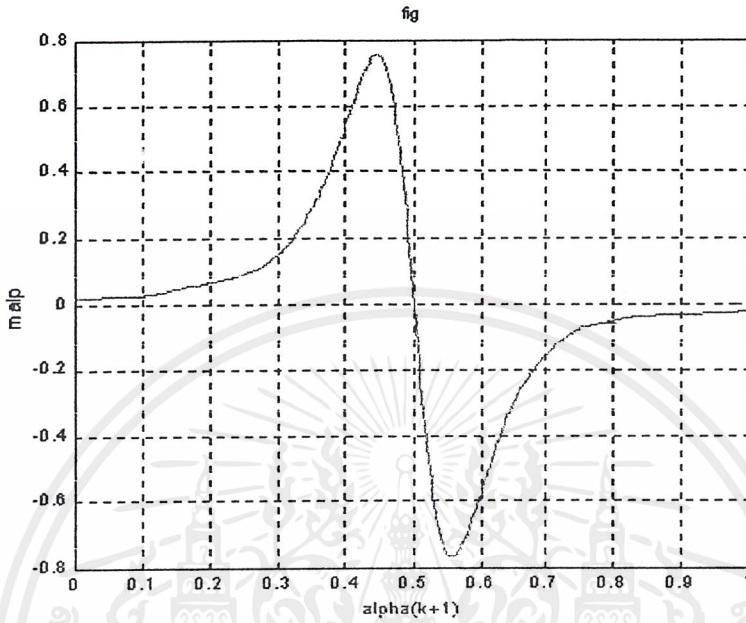


$A=1, \mu=0.00005, \omega=\pi/4, l=0,1$  และ  $2$

รูปที่ 3.24

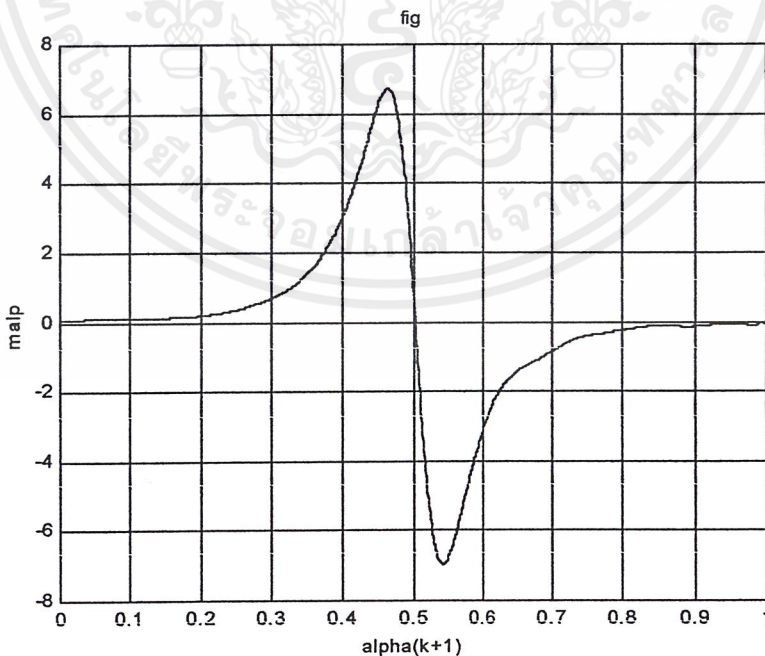
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.2.6 กราฟแสดงคุณสมบัติค่าการเปลี่ยนแปลงของ $\alpha_1(k)$ กรณี $L = 1$



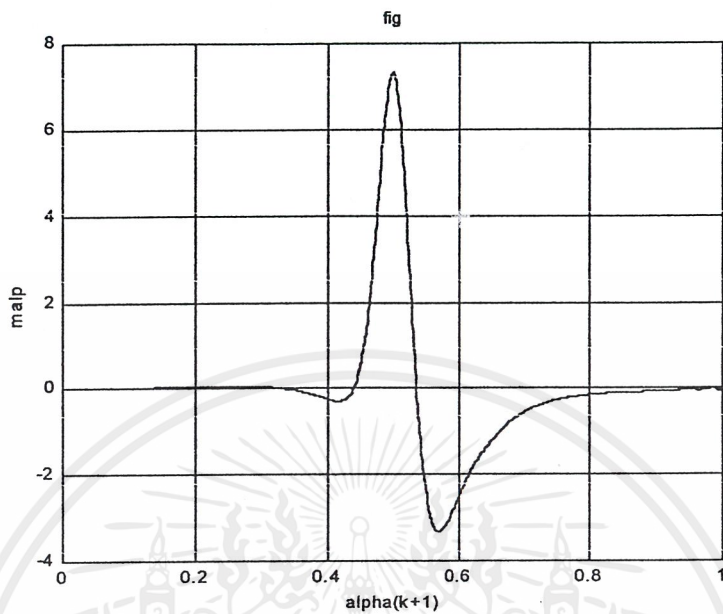
รูปที่ 3.25  $\alpha_0 = 0.85$ ,  $\mu = 0.00005$ ,  $\omega = \pi/3$

### 3.2.7 กราฟแสดงคุณสมบัติค่าการเปลี่ยนแปลงของ $\alpha_1(k)$ กรณี $L = 3$ , $l = 0, 1$ และ $2$

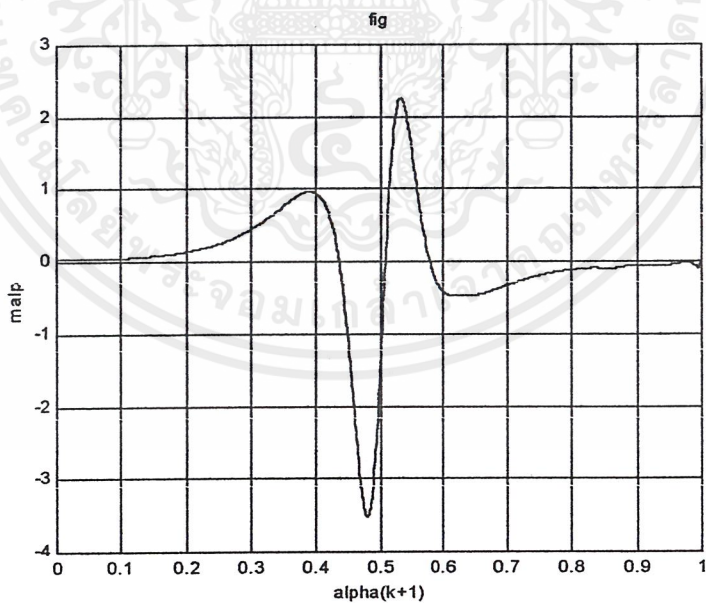


รูปที่ 3.26  $\alpha_0 = 0.85$ ,  $\mu = 0.00005$ ,  $\omega = \pi/3$ ,  $l = 0$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

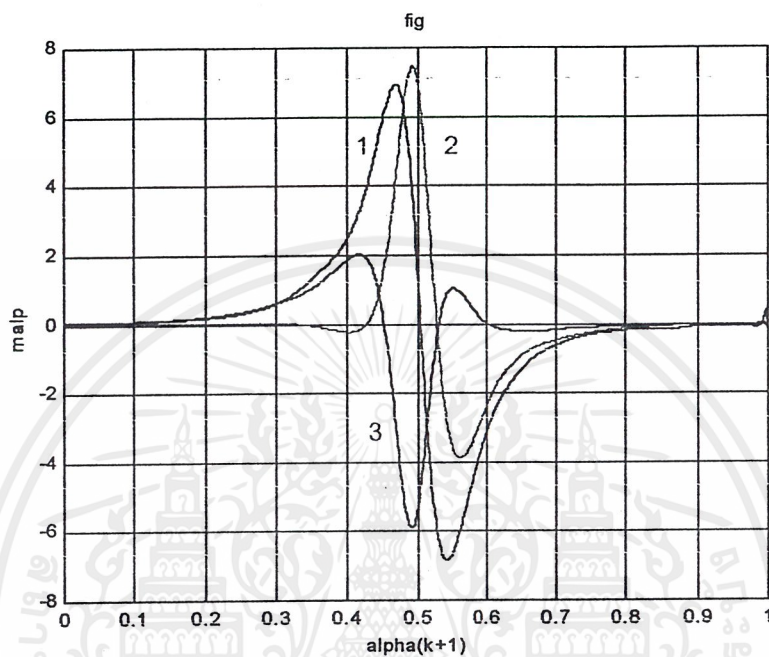


รูปที่ 3.27  $\alpha_0 = 0.85$ ,  $\mu = 0.00005$ ,  $\omega = \pi/3$ ,  $l = 1$



รูปที่ 3.28  $\alpha_0 = 0.85$ ,  $\mu = 0.00005$ ,  $\omega = \pi/3$ ,  $l = 2$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



$$1. \alpha_0 = 0.85, \mu = 0.00005, \omega = \pi/3, l = 0$$

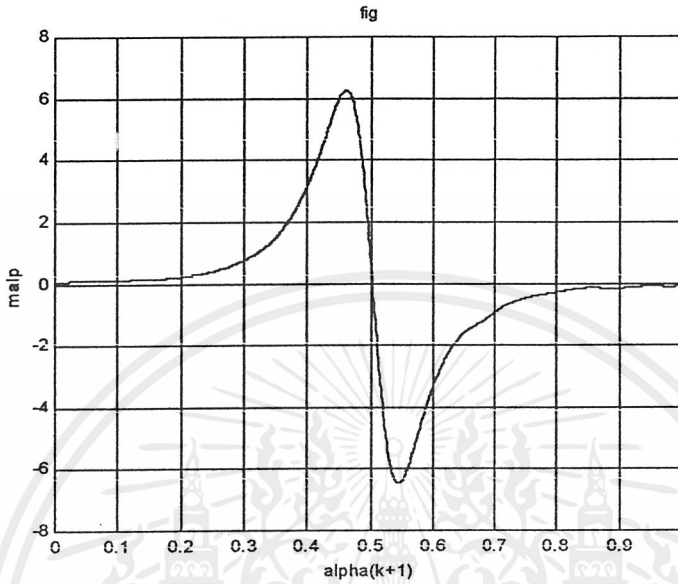
$$2. \alpha_0 = 0.85, \mu = 0.00005, \omega = \pi/3, l = 1$$

$$3. \alpha_0 = 0.85, \mu = 0.00005, \omega = \pi/3, l = 2$$

รูปที่ 3.29

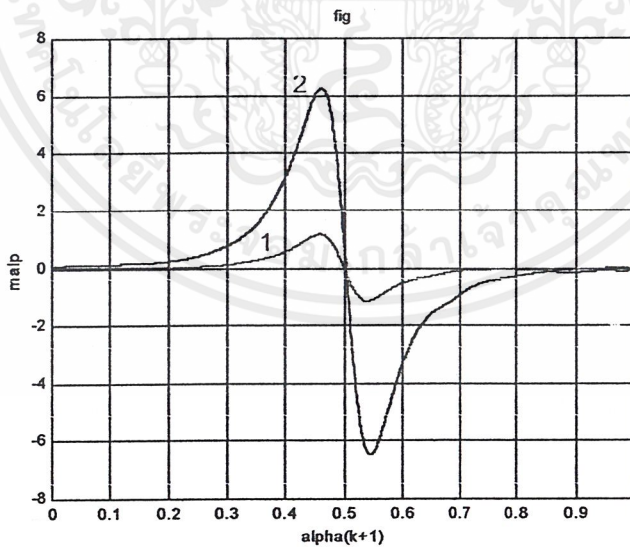
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.2.7 กราฟแสดงคุณสมบัติค่าการเปลี่ยนแปลงของ $\alpha_1(k)$ กรณีโครงสร้างใหม่



รูปที่ 3.30  $\alpha_0 = 0.85$ ,  $\mu = 0.00005$ ,  $\omega = \pi/3$

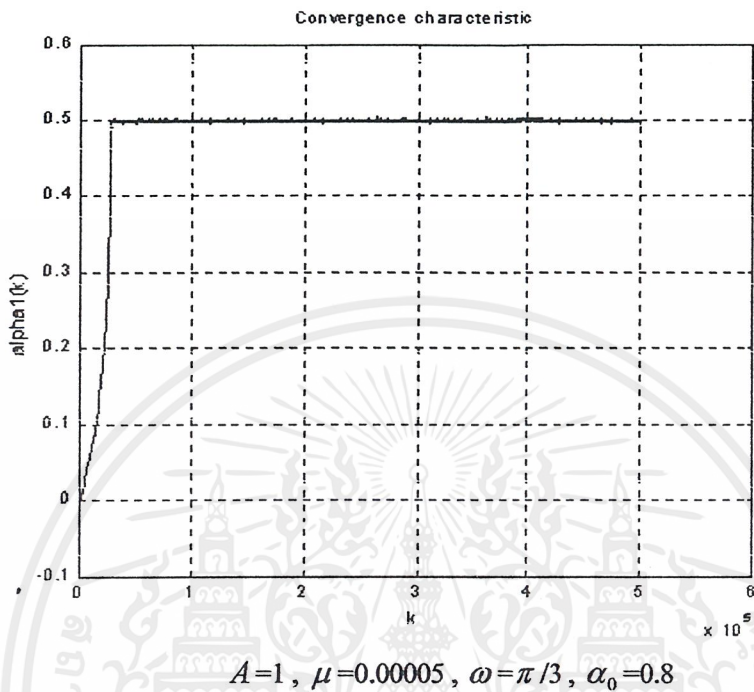
### 3.2.8 รูปแสดงการเปลี่ยนแปลงของ $\alpha_1(k)$ ในกรณีที่ $L=1$ และ $L=3$ โครงสร้างใหม่



รูปที่ 3.31 1.  $\alpha_0 = 0.85$ ,  $\mu = 0.00005$ ,  $\omega = \pi/3$ ,  $L=1$

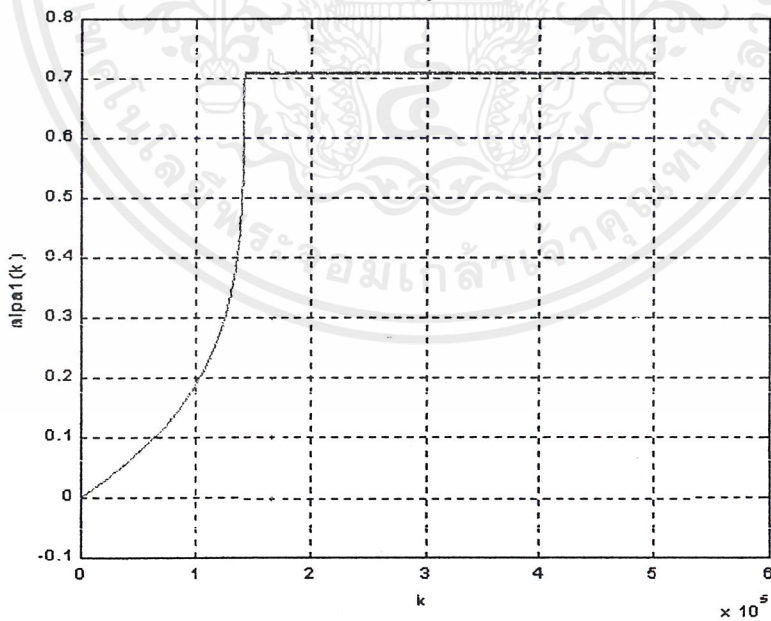
2.  $\alpha_0 = 0.85$ ,  $\mu = 0.00005$ ,  $\omega = \pi/3$ ,  $L=3$

3.2.9 กราฟแสดงคุณสมบัติของ  $\alpha_1(k)$  เทียบกับ  $k$  กรณี โครงสร้างใหม่  $L = 3$   $S/N = 1.8$  dB



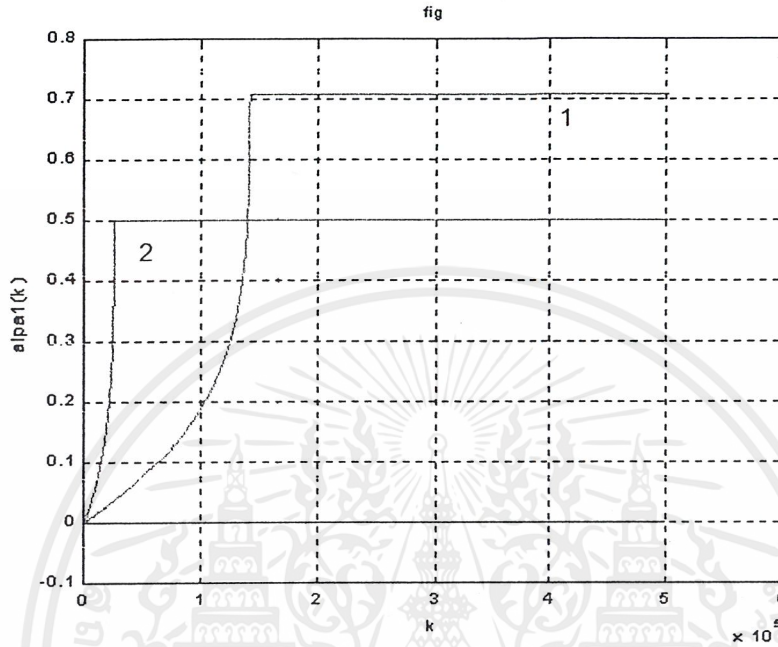
รูปที่ 3.32

fig



รูปที่ 3.33

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



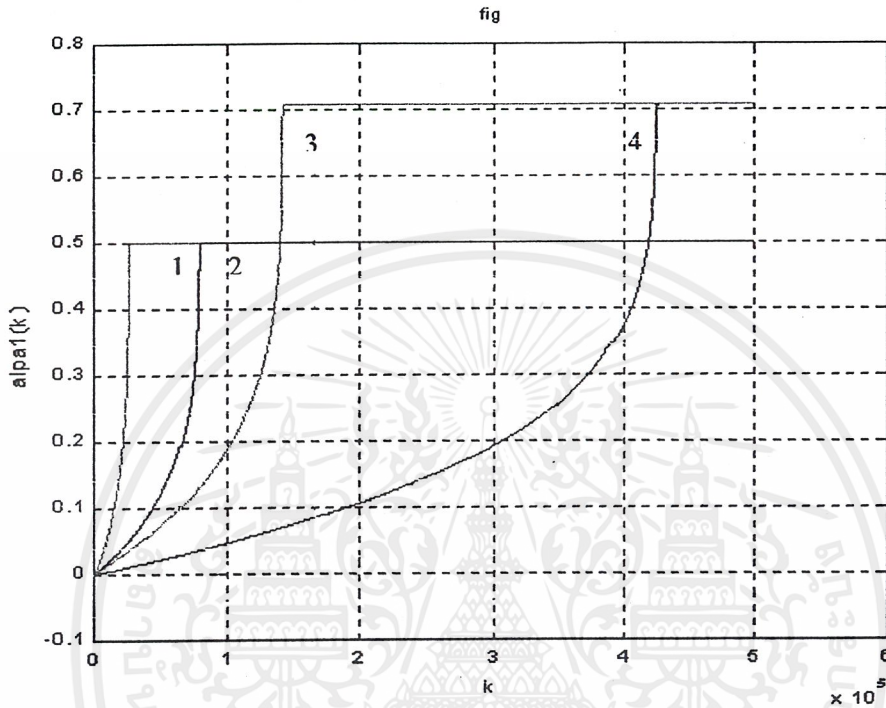
1.  $A=1, \mu=0.00005, \omega=\pi/4, \alpha_0=0.8$

2.  $A=1, \mu=0.00005, \omega=\pi/3, \alpha_0=0.8$

รูปที่ 3.34

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.3.0 กราฟแสดงคุณสมบัติของ $\alpha_1(k)$ เทียบกับ $k$ กรณีโครงสร้างใหม่ $L=1$ และ $L=3$

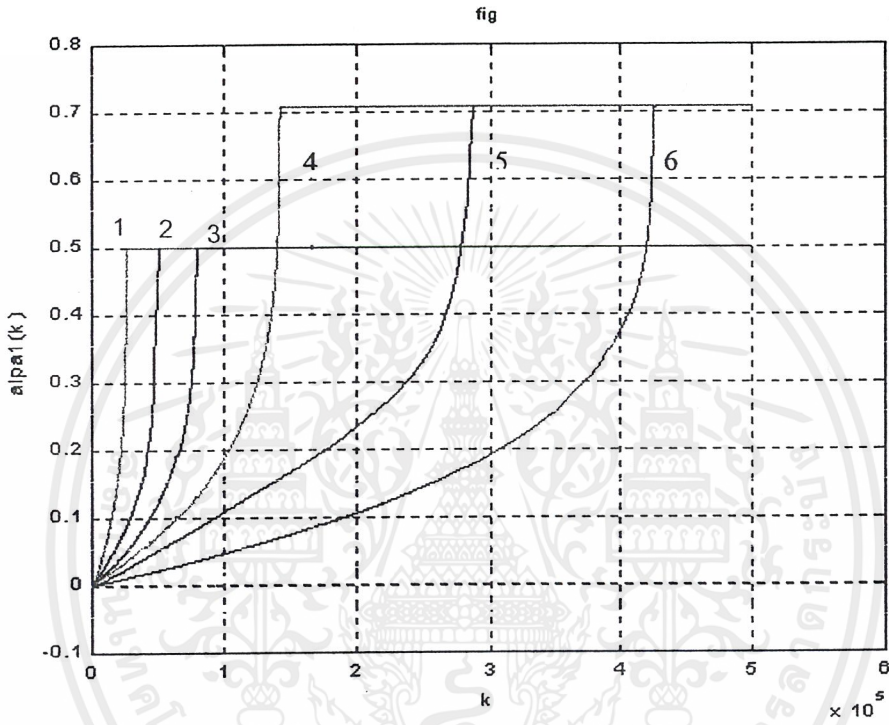


1.  $L=3, \mu=0.00005, \omega=\pi/3, \alpha_0=0.8$
2.  $L=1, \mu=0.00005, \omega=\pi/3, \alpha_0=0.8$
3.  $L=3, \mu=0.00005, \omega=\pi/4, \alpha_0=0.8$
4.  $L=1, \mu=0.00005, \omega=\pi/4, \alpha_0=0.8$

รูปที่ 3.35

3.3.1 กราฟเปรียบเทียบคุณสมบัติของ  $\alpha_1(k)$  เทียบกับ  $k$  กรณีโครงสร้างใหม่  $L=1, L=2$

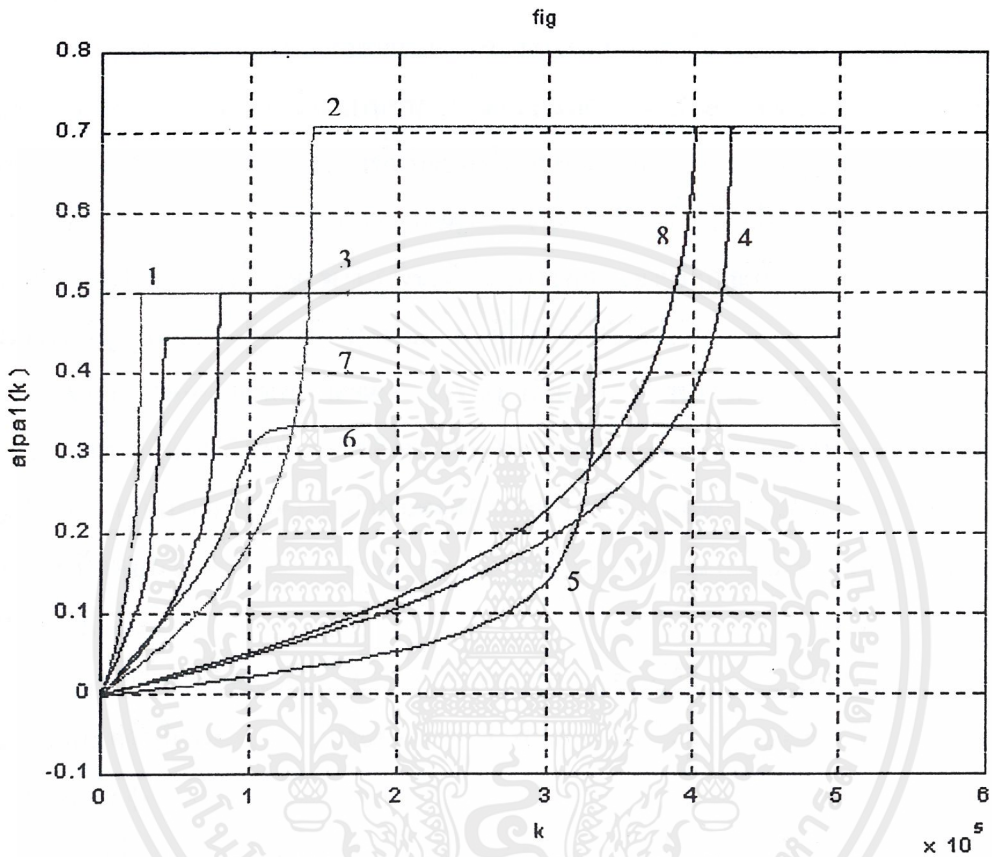
และ  $L=3$



- 1.  $L=3, \mu=0.00005, \omega=\pi/3, \alpha_0=0.8$
- 2.  $L=2, \mu=0.00005, \omega=\pi/3, \alpha_0=0.8$
- 3.  $L=1, \mu=0.00005, \omega=\pi/3, \alpha_0=0.8$
- 4.  $L=3, \mu=0.00005, \omega=\pi/4, \alpha_0=0.8$
- 5.  $L=2, \mu=0.00005, \omega=\pi/4, \alpha_0=0.8$
- 4.  $L=1, \mu=0.00005, \omega=\pi/4, \alpha_0=0.8$

รูปที่ 3.36

### 3.3.2 กราฟแสดงคุณสมบัติของ $\alpha_1(k)$ เทียบกับ $k$



1.  $\alpha_1(k)$ ,  $L=3$ ,  $\omega=\pi/3$ , โครงสร้างใหม่
2.  $\alpha_1(k)$ ,  $L=3$ ,  $\omega=\pi/4$ , โครงสร้างใหม่
3.  $\alpha_1(k)$ ,  $L=3$ ,  $\omega=\pi/3$ , โครงสร้างสเกลาร์
4.  $\alpha_1(k)$ ,  $L=3$ ,  $\omega=\pi/4$ , โครงสร้างสเกลาร์
5.  $\alpha_{1,0}(k)$ ,  $L=3$ ,  $\omega=\pi/3$ , โครงสร้างขนาน
6.  $\alpha_{1,1}(k)$ ,  $L=3$ ,  $\omega=\pi/3$ , โครงสร้างขนาน
7.  $\alpha_{1,2}(k)$ ,  $L=3$ ,  $\omega=\pi/3$ , โครงสร้างขนาน
8.  $\alpha_{1,l}(k)$ ,  $l=0,1,2$ ,  $L=3$ ,  $\omega=\pi/4$ , โครงสร้างขนาน

รูปที่ 3.37

## บรรณานุกรม

- (1) Widrow B., Stearn S. **“Adaptive Signal Processing”**, Prentice-Hall, Inc. Englewood Cliffs, 1985
- (2) Richard G. Lyons **“Understand Digital Signal Processing”**, IEEE Proc., pp.561-751, April 1978
- (3) W. Kenneth Jenkins, Andrew W. Hull, Jeffrey C. Strait, Bernard A. Schnauffer and Xiaohui Li **“Advance Concepts In Adaptive Signal Processing”**
- (4) Simon Haykin **“Introduction To Adaptive Filter”**, Macmillan Publishing Company, New York
- (5) Alan V. Oppenheim and Ronald W. Schaffer **“Digital Signal Processing”**, Prentice-Hall International, Alan V. Oppenheim and Bell Telephone Laboratories, Inc., 1975
- (6) S. Nishimura, S. Mori, J. K. Kim and K. Hirano, **“Parallel block structure for adaptive high-speed noise canceller”**, IEICE Trans., J. 72-A, 5, pp.786-794, May 1989
- (7) D. M. Etter, **“Engineer Problem Solving with Matlab”**, Prentice-Hall Simon & Schuster (Asia), 1996
- (8) Sanjit, K. Mitra, **“Handbook for Digital Signal Processing”**, John Wiley & Sons, Inc, California, 1993
- (9) C. Benjangprasert, K. Kikuchi, N. Takahashi, T. Takabe., **“An Implementation of High-Speed Adaptive Noise canceller with Parallel BlocStructure”**, Proc. APCCAS'94, pp.376-381