

การออกแบบอะแดปทีฟฟิลเตอร์แบบแปรค่าเสถียรไซส์โครงสร้างแลตทิซ
สำหรับขจัดสัญญาณสะท้อนในระบบโทรศัพท์

DESIGN OF VARIABLE STEP-SIZE LATTICE STRUCTURE FILTER FOR
ECHO CANCELLATION IN TELEPHONE SYSTEM

เสถียรวุฒิ สุขุมาลวงศ์
SETHAWUT SUKHUMALWONG

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชาวิศวกรรมสารสนเทศ

บัณฑิตวิทยาลัย

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ. 2550

การออกแบบอะแดปทีฟฟิลเตอร์แบบแปรค่าเสต็ปไซส์โครงสร้างแลตทิซ
สำหรับขจัดสัญญาณสะท้อนในระบบโทรศัพท์

DESIGN OF VARIABLE STEP-SIZE LATTICE STRUCTURE FILTER FOR
ECHO CANCELLATION IN TELEPHONE SYSTEM

เสถียรวุฒิ สุขุมาลวงศ์

SETHAWUT SUKHUMALWONG

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชาวิศวกรรมสารสนเทศ

บัณฑิตวิทยาลัย

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ.2550

**DESIGN OF VARIABLE STEP-SIZE LATTICE STRUCTURE FILTER FOR
ECHO CANCELLATION IN TELEPHONE SYSTEM**

SETHAWUT SUKHUMALWONG

**A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENT FOR THE DEGREE OF
MASTER OF ENGINEERING IN INFORMATION ENGINEERING
SCHOOL OF GRADUATE STUDIES
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG**

2007

COPYRIGHT 2007

SCHOOL OF GRADUATE STUDIES

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

หัวข้อวิทยานิพนธ์	การออกแบบอะแดปทีฟฟิลเตอร์แบบแปรค่าเสถียรไปไซส์โครงสร้างแลตทิซสำหรับขจัดสัญญาณสะท้อนในระบบโทรศัพท์
นักศึกษา	นายเสกฐวุฒิ สุขุมาลวงศ์
รหัสประจำตัว	45061120
ปริญญา	วิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชา	วิศวกรรมสารสนเทศ
พ.ศ.	2550
อาจารย์ที่ปรึกษาวิทยานิพนธ์	รศ. ดร.ชวลิต เบญจางคประเสริฐ

บทคัดย่อ

วิทยานิพนธ์นี้นำเสนอ การออกแบบอะแดปทีฟอัลกอริทึมแบบแปรค่าเสถียรไปไซส์ สำหรับตัวกรองอะแดปทีฟ IIR แบบนอตช์อันดับสองที่ใช้โครงสร้างแลตทิซ ทดสอบสมรรถนะของอัลกอริทึม โดยการประมาณค่าสัญญาณชาวน์ที่ปนมากับสัญญาณรบกวนแบบเกาส์เซียนและประยุกต์ใช้สำหรับขจัดสัญญาณสะท้อนกลับในระบบโทรศัพท์ อัลกอริทึมที่นำเสนอนี้มีด้วยกันสามแบบคือ อัลกอริทึมแรกใช้อัตสหสัมพันธ์ของสัญญาณเอาต์พุต และค่าอัตสหสัมพันธ์ของค่าความคลาดเคลื่อนในการปรับเปลี่ยนค่าเสถียรไปไซส์ อัลกอริทึมแบบที่สองทำการปรับเปลี่ยนค่าเสถียรไปไซส์โดยการปรับสัญญาณเกรเดียนต์ใหม่โดยใช้ผลต่างของสัญญาณระหว่างอินพุตในอดีตและผลคูณของเอาต์พุตในอดีตกับค่าสัมประสิทธิ์ที่ใช้กำหนดแบนด์วิดท์ของตัวกรอง ซึ่งทั้งสองวิธีที่กล่าวมานั้นใช้วิธีกำหนดแบนด์วิดท์ของตัวกรองแบบคงที่ และอัลกอริทึมแบบสุดท้ายใช้วิธีการปรับเปลี่ยนแบนด์วิดท์แบบแปรค่าตามเวลา จากผลการจำลองการทำงานของอัลกอริทึมทั้งสาม พบว่า ทั้งการประมาณค่าสัญญาณชาวน์ และการประยุกต์ใช้สำหรับขจัดสัญญาณสะท้อนในระบบโทรศัพท์ มีการลู่เข้าที่รวดเร็ว ค่าความคลาดเคลื่อนต่ำ สามารถติดตามสัญญาณและขจัดสัญญาณสะท้อนได้ดีขึ้น ตามลำดับ

Thesis Title	Design of Variable Step-size Lattice Structure Filter for Echo Cancellation in Telephone System
Student	Mr. Sethawut Sukhumalwong
Student ID.	45061120
Degree	Master of Engineering
Program	Information Engineering
Year	2007
Thesis Adviser	Assoc. Prof. Dr. Chawalit Benjangkprasert

ABSTRACT

In this thesis presents design of variable step-size algorithms for adaptive IIR notch filter lattice form structure. To evaluate the performance of the proposed algorithms are used to estimate sinusoidal signal corrupted by white Gaussian noise and application in telephone network adaptive filtering for echo cancellation. In this work provides three types of variable step-size algorithms. The first type is using the autocorrelation of output signal and autocorrelation of the error signal to control the adaptation process. The second type is using the new gradient signal. In both cases, the first type and second type algorithms are used a fixed bandwidth parameter of the filter. Finally, the third type is using a variable bandwidth technique to control the adaptation process of the algorithm. It is shown that the proposed algorithms provide fast convergence speed, low mean square error, fast tracking and low average echo residue relative to the previous one. The simulation results are provided to confirm the performance of the proposed three algorithms.

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลงได้ด้วยดี ด้วยคำแนะนำ คำปรึกษาจากอาจารย์ผู้ควบคุมวิทยานิพนธ์ รศ.ดร. ชวลิต เบนญางคประเสริฐ ที่ได้ให้การสนับสนุนในด้านต่าง ๆ ที่เป็นประโยชน์ในการวิจัย ด้วยความเมตตาอนุเคราะห์จากท่าน ตลอดจนน้ำใจไมตรีอันดีเยี่ยมตลอดเวลา ผู้วิจัยรู้สึกทราบบ้างมาก และ ขอกราบขอบพระคุณมา ณ ที่นี้ด้วย

ขอขอบพระคุณ บริษัท ทีโอที จำกัด (มหาชน) ที่ให้โอกาส และให้การสนับสนุนด้านการศึกษาตลอดมา

ขอกราบขอบพระคุณ รศ.นภพินท์ อนันตรศิริชัย รศ.อรลภ แสงอรุณ รวมทั้ง ครุบาอาจารย์ท่านอื่น ๆ ที่มอบคำแนะนำดี ๆ ให้ทุก ๆ ท่าน และ ต้องกราบขอภัยที่ไม่ได้เอ่ยนามไว้ ณ ที่นี้ด้วย

ขอขอบคุณห้องวิจัย Signal transmission laboratory สำนักวิจัยการสื่อสารและเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ที่ให้การสนับสนุนโปรแกรม MATLAB

ขอกราบขอบพระคุณ คุณพ่อ คุณแม่ และ ครอบครัวอันเป็นที่รักยิ่ง ที่ได้ให้การสนับสนุน ให้กำลังใจ ส่งเสริมด้านการศึกษา ด้วยดีตลอดมา

ขอขอบคุณ คุณวัชรพงษ์ ดิ่งจันทร์ และ พี่ ๆ เพื่อน ๆ น้อง ๆ ภาควิชาวิศวกรรมสารสนเทศ ที่ให้คำแนะนำ ความคิดเห็น และความช่วยเหลือในทุก ๆ ด้านตลอดมา

และ ขอขอบคุณ บัณฑิตวิทยาลัยที่ให้การสนับสนุนการทำวิทยานิพนธ์ในครั้งนี้

สุดท้ายนี้ คุณค่าและประโยชน์อันพึงได้จากวิทยานิพนธ์ฉบับนี้ ข้าพเจ้าขอบอบแต่ผู้มีพระคุณทุกท่าน

เสถียรวุฒิ สุขุมาลวงศ์

สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญตาราง.....	VIII
สารบัญรูป.....	IX
บทที่ 1 บทนำ.....	1
1.1 ความเป็นมาและความสำคัญของปัญหา.....	1
1.2 ความมุ่งหมาย และวัตถุประสงค์ของการศึกษา.....	2
1.3 แนวความคิดที่ใช้ในการทำวิจัย.....	2
1.4 ขอบเขตการวิจัย.....	3
บทที่ 2 สัญญาณสะท้อนกลับในระบบ โทรศัพท์.....	4
2.1 การเกิดสัญญาณสะท้อนกลับในระบบ โทรศัพท์.....	5
2.2 การควบคุมสัญญาณเสียงพูดและการขจัด.....	7
2.2.1 วงจรระงับสัญญาณสะท้อนกลับ.....	7
2.2.2 วงจรขจัดสัญญาณเสียงพูดสะท้อนกลับ.....	7
2.3 วงจรขจัดสัญญาณสะท้อนกลับของการรับส่งข้อมูลใน โมเด็ม.....	9
2.3.1 การขจัดสัญญาณสะท้อนกลับสำหรับการส่งข้อมูล.....	10
2.3.2 ข้อบังคับของการขจัดสัญญาณสะท้อนกลับ.....	12
2.4 การขจัดสัญญาณสะท้อนกลับใน Subscriber-loop แบบคิจิตอล.....	13
2.5 วงจรขจัดสัญญาณสะท้อนแบบ Data-driven Nyquist.....	13
2.5.1 โครงสร้างของวงจรขจัดสัญญาณข้อมูลสะท้อนกลับใน โมเด็ม.....	13
2.5.2 โครงสร้างและการทำงานของวงจรขจัดสัญญาณสะท้อนแบบ Data-driven Nyquist ใน โมเด็ม.....	14
บทที่ 3 ตัวกรองอะแคปทีฟและการประยุกต์ใช้งาน.....	19
3.1 ตัวกรองความถี่แบบ IIR.....	22

สารบัญ(ต่อ)

	หน้า
3.2 โครงสร้างของตัวกรองความถี่แบบ IIR.....	24
3.2.1 ตัวกรอง โครงสร้างรูปแบบโดยตรง.....	24
3.2.2 ตัวกรอง โครงสร้างรูปแบบแลตทิซ.....	26
3.3 การทดสอบเสถียรภาพ.....	28
3.4 แบบจำลองพื้นฐานตัวกรองอะแดปทีฟ.....	31
3.4.1 อัลกอริทึมแบบ LMS.....	32
3.4.2 วิธี Steepest Descent.....	35
3.4.3 อัลกอริทึมแบบ MNG.....	41
3.5 การประยุกต์ใช้งานตัวกรองอะแดปทีฟ.....	43
3.5.1 การจำลองระบบและ สถานการณ์ล่วงหน้า.....	43
3.5.2 Equalization and Inverse Modeling.....	44
3.5.3 Adaptive Linear Prediction.....	46
3.5.4 Adaptive Autoregressive Spectrum Analysis.....	49
3.5.5 การขจัดสัญญาณสะท้อน.....	51
3.5.5.1 วงจรขจัดสัญญาณสะท้อนในโครงข่าย.....	51
3.5.5.2 วงจรขจัดสัญญาณสะท้อนในระบบเสียง.....	52
3.5.6 การขจัดสัญญาณรบกวนแบบอะแดปทีฟ.....	54
บทที่ 4 การประมาณค่าใน โครงสร้างแลตทิซ.....	57
4.1 การจำลองระบบ.....	57
4.2 การประมาณค่าแบบ MMSE.....	60
4.2.1 ความสัมพันธ์การประมาณค่าแบบ MMSE ต่อสัญญาณคงที่.....	60
4.2.2 ความเป็นออร์ทอโกนอล.....	62
4.2.3 อัตราสัมพันธ์ของความคลาดเคลื่อน.....	62
4.3 การประมาณแบบ MMSE กับ อันดับที่ซ้ำกัน.....	63
4.4 การประมาณค่าแบบ MMSE สำหรับ โครงสร้างแบบแลตทิซ.....	64
4.4.1 การคาดคะเนเดีนหน้าและย้อนกลับแบบลิเนียร์.....	64
4.4.2 การคาดคะเนอันดับที่ซ้ำกันแบบลิเนียร์ โครงสร้างแลตทิซ.....	67

สารบัญ(ต่อ)

	หน้า
4.4.3 การประมาณค่าร่วมของ MMSE โครงสร้างแลตทิซ.....	70
4.5 คุณสมบัติของตัวกรองประมาณแลตทิซแบบ MMSE.....	71
4.5.1 ประสิทธิภาพเสตทต่อเสตท.....	71
4.5.2 คุณสมบัติความสัมพัทธ์ของตัวกรองคาดคะเนความคลาดเคลื่อน.....	73
4.5.3 การคำนวณอันดับซ้ำจากการประมาณค่า $y(k)$	74
4.5.4 การคำนวณค่าสัมประสิทธิ์โดยประมาณค่าจากพารามิเตอร์อื่นๆของแลตทิซ.....	74
4.5.5 การทำซ้ำอันดับของการคาดคะเนความคลาดเคลื่อนยกกำลังสองเฉลี่ย α_m	75
บทที่ 5 ตัวกรองอะแดปทีฟ IIR แบบนอตซ์อันดับสอง.....	76
5.1 ตัวกรองอะแดปทีฟแบบ IIR โดยวิธี OUTPUT ERROR.....	79
5.1.1 วิธีการปรับค่าความคลาดเคลื่อนเอาต์พุต.....	80
5.1.2 วิธีการปรับค่าความคลาดเคลื่อนสมการ.....	83
5.2 โครงสร้างของตัวกรองความถี่อะแดปทีฟแบบ IIR อันดับสอง.....	85
5.2.1 โครงสร้างแบบโดยตรง.....	86
5.2.2 โครงสร้างรูปแบบแลตทิซ.....	88
5.3 อะแดปทีฟอัลกอริทึมที่ใช้ในการวิจัย.....	91
5.3.1 อัลกอริทึมสำหรับปรับเสถียรไซส์แบบเดิม.....	91
5.3.2 อัลกอริทึม VSS ที่นำเสนอแบบที่ 1.....	92
5.3.3 อัลกอริทึม VSS ที่นำเสนอแบบที่ 2.....	93
5.3.4 อัลกอริทึม VSS ที่นำเสนอแบบที่ 3.....	93
5.3.5 สรุปสมการที่ใช้ในการจำลองการทำงาน.....	94
5.4 สรุป.....	96
บทที่ 6 ผลการวิจัย.....	97
6.1 การทดสอบระบบด้วยสัญญาณไซน์คลื่นเดี่ยวในสัญญาณรบกวนแบบเกาส์เซียน.....	97
6.2 ความถูกต้องของคำตอบ.....	102
6.2.1 ผลการทดลองจากการวัดค่าความแปรปรวนของสัมประสิทธิ์.....	103
6.2.2 ผลการทดลองจากการวัดค่า bias ของสัมประสิทธิ์.....	104

สารบัญ(ต่อ)

	หน้า
6.2.3 ผลการทดลองจากการวัดค่า MSE.....	106
6.2.4 ผลการทดลองจากการวัดค่า MSE.....	107
6.3 การประยุกต์ใช้งานในการจัดสัญญาณเสียงสะท้อนแบบอะแคปทีฟในระบบ โทรศัพท์.....	109
6.3.1 ผลการทดลองการจัดสัญญาณสะท้อนในระบบ โทรศัพท์.....	111
6.3.1.1 ผลการจำลองอัลกอริทึม VSS ที่นำเสนอแบบที่ 1 เปรียบเทียบกับ อัลกอริทึมเดิม [2].....	112
6.3.1.2 ผลการจำลองอัลกอริทึม VSS ที่นำเสนอแบบที่ 2 เปรียบเทียบกับ อัลกอริทึมเดิม [2].....	113
6.3.1.3 ผลการจำลองอัลกอริทึม VSS ที่นำเสนอแบบที่ 3 เปรียบเทียบกับ อัลกอริทึมเดิม [2].....	114
6.3.1.4 ผลการจำลองอัลกอริทึม VSS ที่นำเสนอทั้งสามแบบเปรียบเทียบกับ อัลกอริทึมเดิม [2].....	116
6.4 สรุป.....	117
บทที่ 7 สรุปผลวิจัยและข้อเสนอแนะ.....	118
บรรณานุกรม.....	120
ภาคผนวก.....	123
ภาคผนวก ก ผลงานวิจัยที่ได้รับการตีพิมพ์.....	124
ประวัติผู้เขียน.....	125

สารบัญตาราง

ตารางที่	หน้า
4.1 คุณสมบัติความสัมพันธ์ของตัวกรองคาดคะเนความคลาดเคลื่อน.....	72

สารบัญรูป

รูปที่	หน้า
1.1 แนวความคิดในการขจัดสัญญาณสะท้อนตัวกรองดิจิทัลแบบปรับตัว.....	1
2.1 การเชื่อมต่อโทรศัพท์ผ่านโครงข่าย PSTN.....	5
2.2 การขจัดสัญญาณสะท้อนกลับและการระงับ.....	6
2.3 โครงสร้างของวงจรขจัดสัญญาณสะท้อนกลับเสียงพูดในระบบโทรศัพท์.....	8
2.4 การสื่อสารข้อมูลด้วยโมเด็มและวงจรขจัดสัญญาณสะท้อนกลับ.....	10
2.5 คุณลักษณะของสัญญาณสัญญาณสะท้อนกลับทั้งใกล้และไกล.....	11
2.6 โครงสร้างทั่วไปของวงจรขจัดสัญญาณสะท้อนกลับในโมเด็ม.....	11
2.7 บล็อกไดอะแกรมของโมเด็มที่มีส่วนของวงจรขจัดสัญญาณสะท้อนกลับแบบ data-driven.....	14
2.8 การสร้างสัญญาณส่งโมเด็ม: (a) การสร้างสัญญาณ Tx (b) สัญญาณต่าง ๆ และ แถบความถี่.....	16
2.9 การกรองและการมอดูเลตสัญญาณส่ง Tx.....	17
3.1 ผลตอบสนองต่อสัญญาณอิมพัลส์ของตัวกรองแบบ FIR และแบบ IIR.....	20
3.2 Signal flow graph ของโครงสร้างรูปแบบโดยตรง 1 สำหรับระบบที่มี N ลำดับ.....	25
3.3 Signal flow graph ของโครงสร้างรูปแบบโดยตรง 2 สำหรับระบบที่มี N อันดับ.....	26
3.4 โครงสร้างตัวกรองแบบ Predictor-error (a) forward PEF และ (b) backward PEF.....	27
3.5 ลักษณะของตัวกรองโครงสร้างรูปแบบแลตทิซ (a) โครงสร้างของตัวกรองแลตทิซ (b) โครงสร้างของตัวกรองแลตทิซ m อันดับ.....	27
3.6 สามเหลี่ยมเสถียรภาพสำหรับฟังก์ชันถ่ายโอนแบบดิจิทัลอันดับสอง.....	30
3.7 บล็อกไดอะแกรมแบบจำลองระบบอะแดปทีฟ.....	31
3.8 บล็อกไดอะแกรมของ LMS อัลกอริทึม.....	33
3.9 วงจรควบคุมน้ำหนักถ่วงอะแดปทีฟ (Adaptive weights control) ของตัวกรองแบบทราน - สเวอร์ชอล.....	33
3.10 วิธีการควบคุมการปรับน้ำหนัก (a) LMS algorithm (b) steepest descent method.....	34
3.11 วงจรกรองแบบทรานสเวอร์ส.....	36
3.12 ตัวกรองแบบปรับตัวได้แบบทรานเวอร์ส N แท้ป.....	38
3.13 แสดงค่า MSE ของอัลกอริทึมแบบ Modified MNG เมื่อกำหนดให้ค่าก้าวกระโดดมีค่าแตกต่างกัน.....	42
3.14 บล็อกไดอะแกรมแบบจำลองระบบอะแดปทีฟ.....	43
3.15 บล็อกไดอะแกรมแบบจำลองระบบอะแดปทีฟอินเวอร์ส.....	44

สารบัญรูป (ต่อ)

รูปที่	หน้า
3.16 บล็อกไดอะแกรมตัวกรองแบบคาดคะเนเดิหน้า	46
3.17 บล็อกไดอะแกรมการคาดคะเนเดิหน้าทรานสเวอร์ซอล	47
3.18 บล็อกไดอะแกรมการคาดคะเนย้อนกลับทรานสเวอร์ซอล	48
3.19 Autoregressive process modeling	49
3.20 บล็อกไดอะแกรมตัวกรองคาดคะเนเดิหน้าแบบทรานสเวอร์ซอล	50
3.21 บล็อกไดอะแกรมของวงจรถัดสัญญาณสะท้อนในโครงข่าย	51
3.22 บล็อกไดอะแกรมของวงจรถัดสัญญาณสะท้อนในระบบเสียง	52
3.23 บล็อกไดอะแกรมของการสื่อสารที่ใช้วงจรถัดสัญญาณสะท้อนทั้งสองชนิดร่วมกัน	53
3.24 บล็อกไดอะแกรมการขจัดสัญญาณรบกวนแบบอะแดปทีฟ	54
3.25 บล็อกไดอะแกรมของการขจัดสัญญาณรบกวนอะแดปทีฟในสายไฟฟ้า	55
3.26 สัญญาณอินพุตและเอาต์พุตของวงจรถัดสัญญาณรบกวนอะแดปทีฟ	55
4.1 โครงสร้างแลตทิซสำหรับการคาดคะเน MMSE แบบลิเนียร์	69
4.2 ตัวกรองประมาณค่าร่วมแลตทิซแบบ MMSE	71
5.1 แนวความคิดในการกำจัดสัญญาณรบกวนแบบอะแดปทีฟ	76
5.2 บล็อกไดอะแกรมตัวกรองแบบ IIR	79
5.3 บล็อกไดอะแกรมวิธีการปรับตัวจากค่าความคลาดเคลื่อนเอาต์พุตของตัวกรองอะแดปทีฟ IIR	80
5.4 บล็อกไดอะแกรมของวิธีการปรับตัวจากค่าความคลาดเคลื่อนสมการ	84
5.5 ส่วนประกอบของตัวกรองความถี่อะแดปทีฟแบบ IIR	85
5.6 การประยุกต์ใช้งานเพื่อทำการขจัดสัญญาณสะท้อนกลับในระบบโทรศัพท์	86
5.7 ผลตอบสนองทางขนาดต่อความถี่ของตัวกรองแบบนอตช์โครงสร้างโดยตรง	87
5.8 ตัวกรองอะแดปทีฟแบบนอตช์ที่มีโครงสร้างแบบแลตทิซ	89
5.9 ผลตอบสนองทางขนาดต่อความถี่ของตัวกรองแบบนอตช์โครงสร้างโดยตรง	90
6.1 บล็อกไดอะแกรมสำหรับการประมาณค่าสัญญาณชาวน์โดยใช้ ANF	98
6.2 (a) ลักษณะของสัญญาณชาวน์ (b) สัญญาณรบกวน (c) สัญญาณอินพุต และ (d) สัญญาณเอาต์พุต	99
6.3 ลักษณะของสัญญาณเอาต์พุตขณะเกิดการปรับตัวของอัลกอริทึม	100
6.4 เปรียบเทียบสัมประสิทธิ์ของตัวกรอง “ $k(n)$ ”	101

สารบัญรูป (ต่อ)

รูปที่	หน้า
6.5 เปรียบเทียบค่า Variance ของสัมประสิทธิ์ " $k(n)$ " เมื่อ SNR = 0 - 25 dB	103
6.6 เปรียบเทียบค่า bias ของสัมประสิทธิ์ " $k(n)$ " เมื่อกำหนด SNR = 0 - 25 dB	105
6.7 การเปรียบเทียบค่า MSE ของสัมประสิทธิ์ " $k(n)$ " เมื่อเมื่อกำหนด SNR = 0 - 70 dB	106
6.8 การเปรียบเทียบค่า MSE ของสัมประสิทธิ์ " $k(n)$ " เมื่อเมื่อกำหนด SNR = 0 - 30 dB	106
6.9 โครงสร้างวงจรขจัดเสียงสะท้อนแบบอะแคปทีฟ	108
6.10 สัญญาณจำลอง Echo path	111
6.11 ผลการจำลองอัลกอริทึม VSS ที่นำเสนอแบบที่ 1 เปรียบเทียบกับอัลกอริทึมเดิม [2]	112
6.12 ผลการจำลองอัลกอริทึม VSS ที่นำเสนอแบบที่ 2 เปรียบเทียบกับอัลกอริทึมเดิม [2]	113
6.13 ผลการจำลองอัลกอริทึม VSS ที่นำเสนอแบบที่ 3 เปรียบเทียบกับอัลกอริทึมเดิม [2]	115
6.14 ผลการจำลองอัลกอริทึม VSS ที่นำเสนอทั้งสามแบบ เปรียบเทียบกับอัลกอริทึมเดิม [2]	116

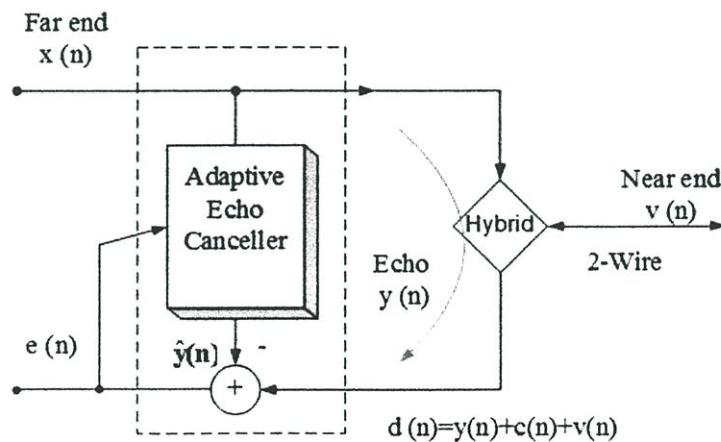
บทที่ 1

บทนำ

1.1 ความเป็นมา และความสำคัญของปัญหา

การขจัดสัญญาณสะท้อนกลับ มักจะใช้ตัวกรองแบบปรับตัวได้ (Adaptive filter) ตัวกรองแบบปรับตัวได้ จะสร้างสัญญาณสะท้อนกลับโดยการประมาณค่าสัญญาณสะท้อนที่เวลาจริง (Real time) และทำการขจัดสัญญาณสะท้อนกลับด้วยการลบสัญญาณสะท้อนที่สร้างขึ้นกับสัญญาณสะท้อนที่รับได้ สัญญาณสะท้อนจำนวนมากเกิดจากวัตถุที่อยู่รอบ ๆ แหล่งกำเนิดเสียง (Acoustic source) ซึ่งเรียกว่า สัญญาณสะท้อนกลับแบบอะคูสติก (Acoustic Echoes) หรืออาจจะเกิดขึ้นในโครงข่ายโทรศัพท์ แบบ PSTN (Public Switched Telephone Network) ซึ่งจะเรียกว่า สัญญาณสะท้อนกลับโทรศัพท์ (Telephone echoes)

ตัวกรองความถี่ที่นำมาใช้สำหรับการประมาณค่าสัญญาณสะท้อนนั้น สามารถใช้ได้ทั้งตัวกรองแบบคงที่ (Fixed filter) และตัวกรองแบบอะแดปทีฟ (Adaptive filter) แต่การออกแบบตัวกรองแบบคงที่นั้น ต้องทราบถึงคุณสมบัติที่สำคัญของสัญญาณก่อน (Primary knowledge) ทั้งสัญญาณที่ต้องการประมาณ และสัญญาณรบกวน แต่สำหรับตัวกรองแบบปรับตัวนั้น มีความสามารถในการปรับแต่งตัวเองได้ และการออกแบบก็ไม่ต้องทราบคุณสมบัติของสัญญาณก่อนหรือทราบเพียงเล็กน้อยก็เพียงพอในการประมวลผลแล้ว ในวิทยานิพนธ์นี้จะใช้ตัวกรองแบบนอตซ์อันดับสอง โครงสร้างแลคทิส ทำงานร่วมกับอัลกอริทึมสำหรับขจัดสัญญาณสะท้อนในระบบโทรศัพท์ที่สามารถแปรค่าเสต็ปไซส์ได้



รูปที่ 1.1 แนวความคิดในการขจัดสัญญาณสะท้อนกลับโดยใช้ตัวกรองดิจิทัลแบบปรับตัว

จากรูปที่ 1.1 แสดงการขจัดสัญญาณสะท้อนในระบบโทรศัพท์ (Echo cancellation) ที่นำเสนอในวิทยานิพนธ์นี้ เมื่อ $d(n)$ คือสัญญาณเสียงที่จะส่งเข้าไปในระบบโทรศัพท์ ประกอบไปด้วย สัญญาณสะท้อนที่ย้อนผ่านไฮบริคเข้ามา $y(n)$ รวมกับ สัญญาณรบกวน $c(n)$ และสัญญาณเสียงพูดต้นทาง $v(n)$ $x(n)$ คือ สัญญาณสะท้อนด้านไกลที่ใช้เป็นสัญญาณอ้างอิง ซึ่งโดยทั่วไปจะถูกรบกวนด้วยสัญญาณรบกวน วัตถุประสงค์ของอัลกอริทึมคือ พยายามทำให้ค่า $\hat{y}(n)$ มีค่าใกล้เคียงกับสัญญาณสะท้อนมากที่สุด เพื่อไม่ให้มีสัญญาณสะท้อนย้อนกลับไปทางภาคส่ง $e(n)$ ได้

ส่วนการประยุกต์ใช้งานอื่น ๆ สำหรับการกำจัดสัญญาณรบกวนแบบอะแคปทีฟ ได้แก่ การประมาณค่า (Estimation) การตรวจวัด (Detecting) และการยกระดับ (Enhancement) สัญญาณคลื่นไซน์ (Sinusoid) ความถี่ต่ำ โดยใช้ตัวกรองความถี่แบบหยุด (Notch filter) ซึ่งง่ายต่อการปรับเปลี่ยนความถี่เป็น infinite null และสามารถติดตามความถี่ที่ถูกต้องของสัญญาณได้

กระบวนการที่สำคัญสำหรับการขจัดสัญญาณสะท้อนในระบบโทรศัพท์ คือ กระบวนการแก้ปัญหาโดยการปรับสัมประสิทธิ์ของตัวกรองความถี่ โดยหลักการแล้วสามารถแบ่งได้สองวิธี คือ กระบวนการแบบ Stochastic gradient approach เช่น อัลกอริทึมแบบ LMS (Least-mean-square algorithm) และ Least-square estimation เช่น การประมาณค่าแบบ RLS (Recursive least-square estimation) ซึ่งกระบวนการแบบ stochastic gradient approach เป็นกระบวนการที่น่าสนใจเนื่องจากมีโครงสร้างที่ง่ายกว่า

1.2 ความมุ่งหมาย และวัตถุประสงค์ของการศึกษา

วิทยานิพนธ์นี้ นำเสนออะแคปทีฟอัลกอริทึมแบบแปรค่าเสต็ปไซส์ โดยใช้ตัวกรองแบบนอกรีตอันดับสอง โครงสร้างแลตทิซ ทั้ง 3 อัลกอริทึม เพื่อเพิ่มประสิทธิภาพในการหาคำตอบให้มีความรวดเร็วยิ่งขึ้น สามารถทนต่อสัญญาณรบกวนได้ดี ซึ่งแต่ละแบบจะพัฒนามาจากอะแคปทีฟอัลกอริทึมในบทวิจัย [2] ดังนี้คือ

- 1) แบบปรับเปลี่ยนค่าเสต็ปไซส์ที่ทนต่อสัญญาณรบกวน [3] ร่วมกับค่าอัตราสัมพันธของค่าความคลาดเคลื่อน $e(n)$, $e(n-1)$ และค่า MSE $\hat{e}(n)^2$ [21]
- 2) แบบปรับเปลี่ยนค่าเสต็ปไซส์ด้วยการปรับสัญญาณเกรเดียนต์ใหม่
- 3) แบบปรับเปลี่ยนค่าแถบความถี่หยุดให้เปลี่ยนแปลงตามเวลา [20]

1.3 แนวความคิดที่ใช้ในการวิจัย

แนวความคิดแรกที่ใช้ในบทวิจัย [3] [21] คือการใช้เทคนิคการปรับเปลี่ยนค่าก้าวกระโดดแบบทนต่อสัญญาณรบกวน โดยใช้อัตราสัมพันธของสัญญาณเอาต์พุตควบคุมการปรับเปลี่ยนค่าเสต็ปไซส์ ซึ่งมีผลต่อการเข้าสู่ค่าที่แท้จริงของสัมประสิทธิ์ และความถูกต้อง เนื่องจากการใช้

อัลกอริทึมที่อยู่บนพื้นฐานของเกรเดียนต์ ทำให้เกิดสัญญาณรบกวนเกรเดียนต์ ซึ่งมีผลต่อความแปรปรวน (Variance) ของสัมประสิทธิ์ ถ้าสามารถลดความแปรปรวนของสัมประสิทธิ์ลงได้ ความถูกต้องของสัมประสิทธิ์ตัวกรองก็จะยิ่งมากขึ้น หมายถึง คำตอบที่ได้ มีความถูกต้องมากขึ้น และ ทำการปรับเปลี่ยนค่าออสสัมพันธ์ของค่าความคลาดเคลื่อน $e(n)$, $e(n-1)$ และค่า $MSE \hat{e}(n)^2$ เพื่อควบคุมการปรับเปลี่ยนค่าก้าวกระโดดให้มีความละเอียดยิ่งขึ้น ค่าความคลาดเคลื่อนที่ป้อนกลับเข้ามาในระบบจะมีความถูกต้องมากยิ่งขึ้น ความแปรปรวนของสัมประสิทธิ์ก็จะลดลง แนวความคิดที่สองที่ใช้ในงานวิจัย คือ แบบปรับเปลี่ยนค่าเสถียรไซส์ด้วยการปรับสัญญาณเกรเดียนต์ใหม่ โดยใช้ผลต่างของสัญญาณระหว่างอินพุตในอดีตและผลคูณของเอาต์พุตในอดีตกับค่า ρ ทำให้คำตอบที่ได้ มีความถูกต้องมากยิ่งขึ้น ส่วนแนวความคิดสุดท้ายที่ใช้ในงานวิจัย คือ การปรับเปลี่ยนขนาดของแถบความถี่ใหญ่ให้เปลี่ยนแปลงตาม [20] วิธีนี้จะทำให้ได้ความถูกต้องของคำตอบที่แท้จริงสูงสุด การประมาณค่าสัญญาณสะท้อนทำได้แม่นยำมากขึ้น

1.4 ขอบเขตการวิจัย

วิทยานิพนธ์นี้จะขยายความจากบทวิจัย [20-21] โดยจะกล่าวถึงที่มาของหลักการที่ใช้ในบทวิจัย รวมถึงการพัฒนาประสิทธิภาพของอะแดปทีฟอัลกอริทึมให้มีความรวดเร็วในการหาค่าสัมประสิทธิ์ของตัวกรองและ มีความแม่นยำถูกต้องมากยิ่งขึ้น ลดค่าผิดพลาดในการปรับตัวในช่วง steady state และสามารถทนทานต่อสัญญาณรบกวนแบบเกาส์เซียนได้ โดยการทดสอบการทำงานของอัลกอริทึมด้วยสัญญาณชาวันซ์ ที่ความถี่ 1 kHz ก่อนเพื่อทดสอบการทำงานกับสัญญาณแบบ stationary วัดค่าความคลาดเคลื่อนที่ได้ จากนั้นจะทดสอบด้วยสัญญาณสะท้อนกลับ (non-stationary) วัดค่า ERLE (Echo Return Loss Enhancement) เปรียบเทียบสมรรถนะของอัลกอริทึมทั้งสามรูปแบบในการขจัดสัญญาณสะท้อน โดยใช้โปรแกรมคอมพิวเตอร์ในการจำลองการทำงาน

บทที่ 2

สัญญาณสะท้อนกลับในระบบโทรศัพท์

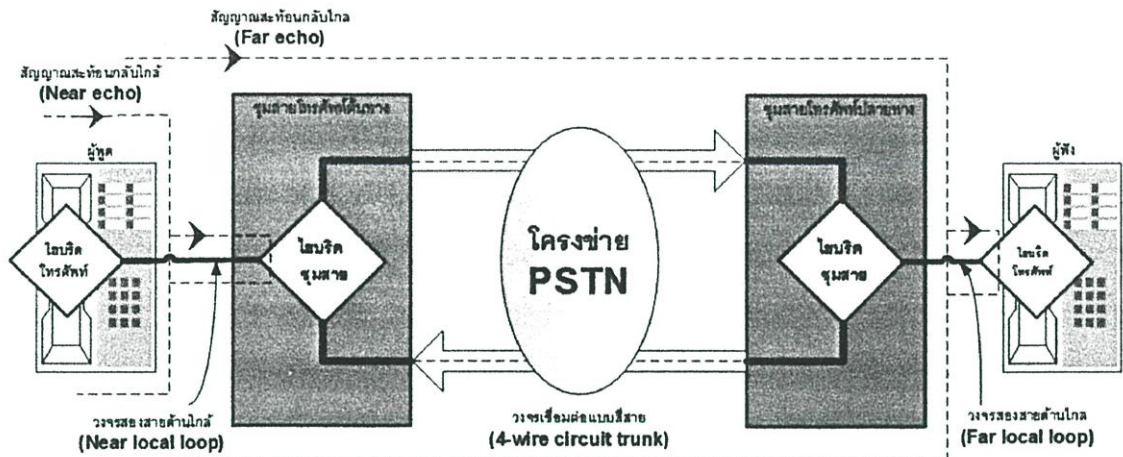
สัญญาณสะท้อนกลับจำนวนมากเกิดจากวัตถุที่อยู่รอบ ๆ แหล่งกำเนิดเสียง (Acoustic source) ซึ่งเรียกว่า สัญญาณสะท้อนกลับแบบอะคูสติก (Acoustic echoes) หรืออาจจะเกิดขึ้นในโครงข่ายโทรศัพท์ แบบ PSTN (Public Switched Telephone Network) ซึ่งจะเรียกว่า สัญญาณสะท้อนกลับโทรศัพท์ (Telephone echoes)

ในระบบโทรศัพท์ การขจัดสัญญาณสะท้อนกลับ มีอยู่ 3 รูปแบบ คือ การขจัดสัญญาณเสียงพูดสะท้อนกลับ (Voice echo cancellation) การขจัดสัญญาณข้อมูลสะท้อนกลับในโมเด็ม (Data echo cancellation) และ การขจัดสัญญาณข้อมูลสะท้อนกลับในโครงข่ายโทรศัพท์แบบดิจิทัล (Integrated Service Digital Network (ISDN))

สัญญาณเสียงพูดสะท้อนกลับในระบบโทรศัพท์ มีอยู่ 2 ชนิด คือ สัญญาณสะท้อนกลับใกล้ (Near end) และสัญญาณสะท้อนกลับไกล (Far end) สัญญาณสะท้อนกลับใกล้ หมายถึง สัญญาณสะท้อนกลับที่มีการหน่วงเวลาสั้น ๆ เช่น ในขณะที่สนทนาผ่านโทรศัพท์ จะได้ยินเสียงของตัวเองในทันทีที่พูด สัญญาณเสียงจะผ่านไมโครโฟน แล้วย้อนกลับมายังหูฟัง ทำให้ได้ยินเสียงตัวเองเป็นสัญญาณที่ถูกหน่วงเวลาเพียงสั้น ๆ ไม่เกิน 35 ms หากมีการหน่วงเวลามากกว่านั้นถึง 50 ms ก็จะได้ยินเป็นเสียงก้อง สัญญาณสะท้อนกลับแบบนี้เป็นการตั้งใจให้เกิดขึ้นเพื่อให้เหมือนกับธรรมชาติการพูดของมนุษย์ที่ต้องได้ยินเสียงตัวเองในขณะที่พูด มิฉะนั้นแล้วผู้พูดจะต้องเปล่งเสียงให้ดังขึ้นอีก ส่วนสัญญาณสะท้อนกลับไกล หมายถึง สัญญาณสะท้อนกลับที่มีการหน่วงเวลายาวนานมาก เกิน 128 ms ขึ้นไป เกิดจากการเชื่อมต่อของระบบโทรศัพท์ที่มีระยะทางไกล ๆ สัญญาณจะเกิดการหน่วงเวลาจากการสะท้อนในตัวกลางสื่อสาร เช่น ไฮบริด (2-4 wire converter) ไยแก้วนำแสง อุปกรณ์ทวนสัญญาณดาวเทียม อุปกรณ์สัญญาณไมโครเวฟ เป็นต้น และอาจย้อนกลับมาอีกเป็นครั้งที่สองและสามได้อีก ซึ่งจะต้องขจัดสัญญาณสะท้อนกลับแบบนี้ (Far end) ให้เหลือน้อยที่สุด

การขจัดสัญญาณสะท้อนกลับ มักจะใช้ตัวกรองแบบปรับตัวได้ (Adaptive filter) ตัวกรองแบบปรับตัวได้ จะสร้างสัญญาณสะท้อนกลับโดยการประมาณค่าสัญญาณสะท้อนกลับที่เวลาจริง (Real time) และทำการขจัดสัญญาณสะท้อนกลับ ด้วยการลบสัญญาณสะท้อนกลับที่สร้างขึ้นกับสัญญาณสะท้อนกลับที่รับได้

2.1 การเกิดสัญญาณสะท้อนกลับในระบบโทรศัพท์



รูปที่ 2.1. การเชื่อมต่อ โทรศัพท์ผ่าน โครงข่าย PSTN

การเชื่อมต่อโทรศัพท์โดยทั่วไป ดังรูปที่ 2.1 เครื่องโทรศัพท์จะต่อกับปลายสายด้านวงจรสองสาย และอีกด้านหนึ่งสายโทรศัพท์จะต่อเข้ากับชุมสายโทรศัพท์ท้องถิ่น (Local telephone central office) ส่วนนี้จะเรียกว่า local loop ในหูฟังโทรศัพท์จะมีไมโครโฟนและลำโพงเล็ก ๆ อยู่อย่างละตัว ไมโครโฟนเป็นอุปกรณ์เปลี่ยนสัญญาณเสียงพูดเป็นสัญญาณทางไฟฟ้าแล้วส่งเข้าไปใน local loop อีกส่วนหนึ่งส่งไปให้ลำโพงในรูปของสัญญาณไฟฟ้า ลำโพงก็จะเปลี่ยนสัญญาณไฟฟ้ากลับเป็นสัญญาณเสียงให้ได้ยิน เป็นที่น่าสังเกตว่า local loop สามารถส่งสัญญาณได้ทั้งสองทิศทางในเวลาเดียวกัน

ในเครื่องโทรศัพท์ลำโพงหูฟังและไมโครโฟนก่อนจะต่อเข้ากับวงจรสองสาย local loop จะต้องมีการเปลี่ยนจากวงจรสองสายเป็นวงจรสี่สาย โดยใช้อุปกรณ์ที่เรียกว่า ไฮบริค (Hybrid coupler) ขณะเดียวกันที่ชุมสายท้องถิ่นก็จะมี ไฮบริคต่อเชื่อมจาก local loop เปลี่ยนวงจรสองสายเป็นวงจรสี่สายเหมือนกัน เพื่อแยกสัญญาณด้านรับด้านส่งออกจากกันผ่านวงจร trunk ส่งต่อไปยังชุมสายปลายทาง สัญญาณจะแยกจากกันได้เด็ดขาดสมบูรณ์ก็ต่อเมื่อ อิมพีแดนซ์ทั้งด้านรับและด้านส่งมีค่าเท่ากัน ในทางปฏิบัติไม่สามารถเกิดขึ้นได้ เป็นผลให้สัญญาณส่งย้อนกลับมาทางด้านรับ ซึ่งเป็นสาเหตุหลักของการเกิดสัญญาณสะท้อนกลับในระบบโทรศัพท์

สัญญาณสะท้อนกลับที่เกิดจากอิมพีแดนซ์ที่ไม่เท่ากันของไฮบริคนี้ ทั้งส่วนที่อยู่ในเครื่องโทรศัพท์ และที่ชุมสายท้องถิ่น รวมกันเรียกว่า สัญญาณสะท้อนกลับด้านใกล้ (Near echo) ซึ่งไม่เป็นปัญหาในการสนทนาแต่อย่างใด แต่กลับสร้างปัญหาใหญ่หลวงในการส่งข้อมูลบนสายโทรศัพท์ ในทำนองเดียวกัน อีกด้านหนึ่งของคู่สนทนา จะเกิดสัญญาณสะท้อนกลับใน

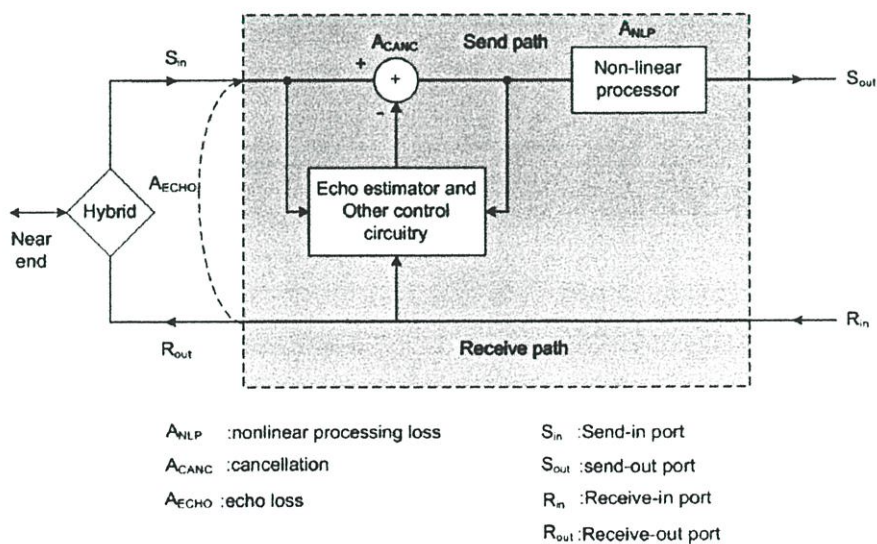
ลักษณะเดียวกันนี้ จะเรียกสัญญาณสะท้อนกลับปลายทางว่า สัญญาณสะท้อนกลับด้านไกล (Far echo)

เนื่องจากสัญญาณสะท้อนกลับด้านไกล ผ่านเข้าไปในวงจรสี่สายทั้งหมด ในการเชื่อมต่อวงจรสี่สายด้วยสายเคเบิลทองแดง สัญญาณจะถูกหน่วงเวลาเป็นสองเท่า หรือในการส่งสัญญาณในเส้นทางไมโครเวฟการหน่วงเวลาอาจมากถึง 150 ms และ การส่งสัญญาณผ่านดาวเทียมประมาณ 1,200 ms สัญญาณสะท้อนกลับด้านไกลนี้ จะมีกำลังอ่อนกว่าสัญญาณที่ต้องการ (เสียงพูดของคู่สนทนาปลายทาง) เนื่องจากมันถูกลดทอนถึงสองครั้งในวงจรสี่สาย ในขณะที่สัญญาณที่ต้องการถูกลดทอนเพียงครั้งเดียว

ในการสนทนาทางโทรศัพท์สัญญาณสะท้อนกลับด้านไกลจะเป็นปัญหามาก และต้องควบคุมเพื่อไม่ให้รบกวนการสนทนา ส่วนการสื่อสารข้อมูลสัญญาณสะท้อนกลับด้านไกลจะถูกจัดได้ง่ายกว่าสัญญาณสะท้อนกลับด้านใกล้ เนื่องจากมีกำลังสัญญาณอ่อนกว่า แต่ความถี่อาจจะเปลี่ยนแปลงไปได้ เนื่องจากผ่านเข้าไปยังวงจรเปลี่ยนแปลงความถี่ (Modulator and demodulator) เรียกว่าการเกิด เฟสโรล (Phase roll) หรือ ฟริควเ็นซี ออฟเซ็ท (Frequency offset) ซึ่งเกิดจากการเลื่อนเฟสของสัญญาณสะท้อนกลับด้านไกลและการสูญเสียแบบนอนลิเนียร์ (Non-linear impairment) ดังนั้นในการขจัดสัญญาณสะท้อนกลับด้านไกล จึงต้องใช้เทคนิคในการชดเชยความถี่แบบนอนลิเนียร์ร่วมด้วยหากมีการเลื่อนเฟสเกิดขึ้น

สัญญาณสะท้อนกลับทั้งด้านใกล้และไกล ที่ไม่มีการเลื่อนเฟส สามารถขจัดได้ง่ายด้วยระบบลิเนียร์ โดยการสังเคราะห์สัญญาณสะท้อนกลับจากสัญญาณที่ส่งออกไป จากค่าพารามิเตอร์ต่าง ๆ ของระบบ เพื่อใช้สร้างสัญญาณสะท้อนกลับและนำมาลบออกจากสัญญาณที่ได้รับ ดังรูปที่ 2.2

[6]



รูปที่ 2.2 การขจัดสัญญาณสะท้อนกลับและการระงับ (Canceller and suppressor)

2.2 การควบคุมสัญญาณเสียงพูดและการขจัด (Voice echo control and cancellation)

ในการส่งสัญญาณเสียงพูดทางโทรศัพท์ สัญญาณสะท้อนกลับด้านไกลไม่จำเป็นต้องถูกขจัด แต่สัญญาณสะท้อนกลับด้านไกลที่มีการหน่วงเวลานานมากกว่า 10 ms ถึง 20 ms เป็นสิ่งหนึ่งที่ต้องคำนึงถึง เพราะจะทำให้เกิดการสูญเสียในช่องส่งสัญญาณโทรศัพท์ สัญญาณสะท้อนกลับจะมีขนาดเพิ่มขึ้นและการหน่วงเวลาจะเพิ่มขึ้นเป็นทวีคูณ ในขณะที่ช่องสัญญาณโทรศัพท์สามารถทนต่อสัญญาณสะท้อนกลับที่มีค่าต่ำกว่าสัญญาณเสียงพูดได้เพียง 1.4 dB เท่านั้น เมื่อสัญญาณสะท้อนกลับ มีการหน่วงเวลาเพิ่มขึ้นถึง 20 60 และ 120 ms แล้ว สัญญาณสะท้อนกลับด้านไกลจะต้องถูกลดทอนลงถึง 11.1 22.7 และ 30.9 dB ตามลำดับ ดังนั้นเพื่อหลีกเลี่ยงการรบกวนคู่สนทนา จะต้องหาวิธีการต่าง ๆ ที่ใช้อยู่ในโครงข่ายโทรศัพท์ปัจจุบัน มาควบคุมสัญญาณสะท้อนกลับควบคู่กับการขจัดสัญญาณสะท้อนกลับด้วย

2.2.1 วงจรระงับสัญญาณสะท้อนกลับ (Echo suppressor)

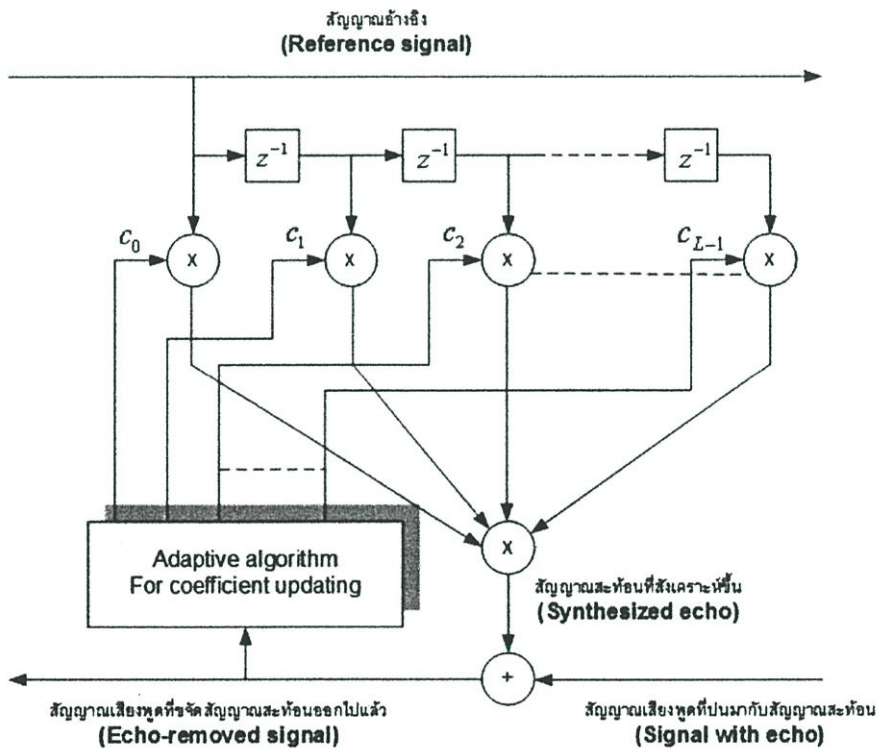
ในการเชื่อมต่อการสนทนาทางโทรศัพท์ที่มีค่าหน่วงเวลาน้อยกว่า 40 ms หรือ การเชื่อมต่อที่มีระยะทางสั้นกว่า 3,222 km และในแต่ละสายส่งที่มีค่าสูญเสียต่อความยาวในระดับที่ยอมรับได้ สัญญาณสะท้อนกลับด้านไกลจะถูกลดทอนลง 2 เท่า เช่นเดียวกับสัญญาณที่ต้องการ แต่ถ้ามีการเชื่อมต่อที่ยาวมากกว่านี้ จะใช้วงจรระงับสัญญาณสะท้อนกลับจากรูปที่ 2.2 ในบล็อก non-linear processor เพื่อป้องกันไม่ให้สัญญาณเสียงพูดถูกลดทอนลงมากเกินไป

วงจรระงับสัญญาณสะท้อนกลับเป็นสวิทช์อิเล็กทรอนิกส์ที่จำเป็น ในการทำงานเกี่ยวกับเสียงพูดเพื่อคอยสังเกตและควบคุมสัญญาณเสียงพูด ที่วิ่งอยู่บนโครงข่ายทั้งสองทิศทางในระหว่างการสนทนาที่มีการเชื่อมต่อระยะทางไกล ๆ วงจรระงับสัญญาณสะท้อนกลับจะสังเกตคู่สนทนาที่กำลังพูดและจะทำการกั้นสัญญาณในทิศทางตรงกันข้าม เพื่อป้องกันไม่ให้สัญญาณสะท้อนกลับผ่านเส้นทางการส่งของผู้ฟังกลับมายังผู้พูดได้ ปัญหาหลักของวงจรระงับสัญญาณสะท้อนกลับคือ มักจะระงับสัญญาณอย่างรวดเร็ว เมื่อคู่สนทนาพูดกันอย่างรวดเร็วทั้งไปและกลับ ทำให้คำพูดอาจจะหายไปบางส่วน ยิ่งไปกว่านั้น เมื่อมีการพูดพร้อมกัน วงจรระงับสัญญาณจะไม่สามารถกั้นสัญญาณได้เลย เป็นผลให้การควบคุมสัญญาณสะท้อนกลับไม่มีประสิทธิภาพ

2.2.2 วงจรขจัดสัญญาณเสียงพูดสะท้อนกลับ (Voice echo cancellation)

การวางตำแหน่งของวงจรขจัดสัญญาณสะท้อนกลับในโครงข่ายโทรศัพท์แสดงในรูปที่ 2.2 ซึ่งแสดงเพียงด้านเดียวของการสนทนา จากรูปจะเห็นว่าวงจรขจัดสัญญาณสะท้อนกลับจะวางไว้ในส่วนที่เป็นวงจรสี่สาย ใกล้กับจุดกำเนิดของสัญญาณสะท้อนกลับ (ไฮบริดในชุมสายโทรศัพท์ที่ต่อกับด้าน local loop) เมื่อสัญญาณไปถึงปลายทางซึ่งจะประกอบไปด้วยสัญญาณเสียงพูดและสัญญาณสะท้อนกลับ วงจรขจัดสัญญาณสะท้อนกลับจะใช้เป็นสัญญาณอ้างอิง (Reference signal) ทำการสุ่มตัวอย่าง ถ้าสัมประสิทธิ์ของตัวกรองสร้างสัญญาณเอาต์พุตได้เหมือนกับ

สัญญาณสะท้อนกลับแล้ว จะสามารถขจัดสัญญาณสะท้อนกลับได้ โดยการนำเอาสัญญาณสะท้อนกลับที่สร้างขึ้น ลบออกจากสัญญาณด้านส่ง หรือ ด้านเอาต์พุตของไฮบริด จะเห็นได้ว่า แม้จะมีการสนทนาพร้อมกัน สัญญาณด้านเอาต์พุตของไฮบริดจะประกอบไปด้วย สัญญาณสะท้อนกลับด้านไกล และสัญญาณเสียงพูดด้านไกล การขจัดสัญญาณสะท้อนกลับก็ยังคงสามารถทำงานได้ดีมีประสิทธิภาพเพียงพอ จะเห็นได้ว่าวงจรขจัดสัญญาณสะท้อนกลับเสียงพูด จะทำการขจัดสัญญาณสะท้อนกลับเสียงของผู้พูดอีกด้านหนึ่ง ซึ่งจะแตกต่างจากการขจัดสัญญาณสะท้อนกลับของข้อมูล เนื่องจากวงจรขจัดสัญญาณสะท้อนกลับอยู่ใกล้แหล่งกำเนิดสัญญาณสะท้อนกลับจะถูกหน่วงเวลาเพียงสั้น ๆ จากสัญญาณอ้างอิง ดังนั้น เส้นทางการหน่วงเวลาของวงจรขจัดสัญญาณสะท้อนกลับก็จะสั้นไปด้วย



รูปที่ 2.3 โครงสร้างของวงจรขจัดสัญญาณสะท้อนกลับเสียงพูดในระบบโทรศัพท์

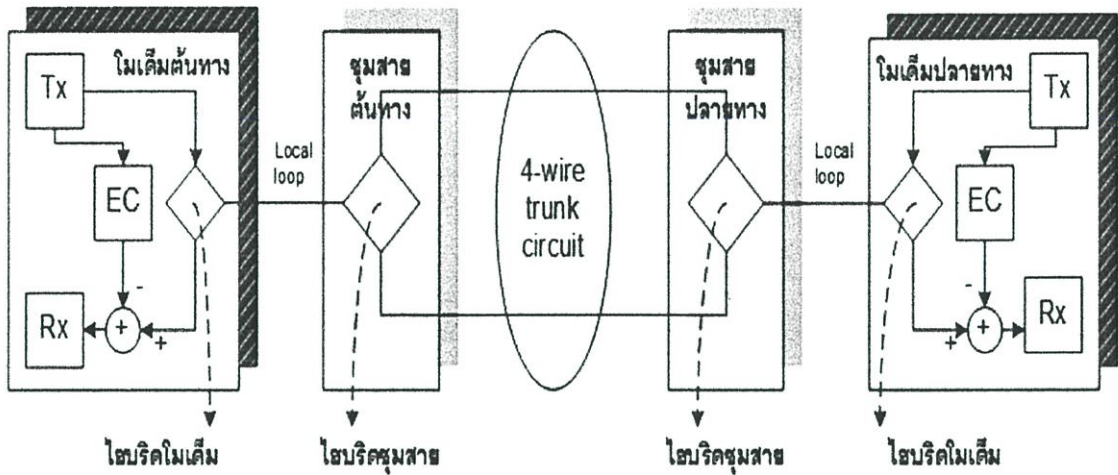
ในการใช้งานจริงจะใช้อัลกอริทึมแบบ LMS [17] เป็นส่วนมากและมีการนำมาใช้อย่างกว้างขวางเนื่องจากประสิทธิภาพในการคำนวณ ส่วนอัลกอริทึมอื่น ๆ เช่น variable least-squares (LS) และ lattice [17] ก็อาจจะมีการใช้งานบ้างในบางกรณี เนื่องจากมีความเร็วในการหาคำตอบ และมีประสิทธิภาพในการติดตามสัญญาณได้ดีกว่า

2.3 วงจรขจัดสัญญาณสะท้อนกลับของการรับส่งข้อมูลในโมเด็ม (Modem data cancellation)

โครงข่ายโทรศัพท์ในปัจจุบันไม่ได้ใช้เฉพาะการส่งสัญญาณเสียงพูดเท่านั้น แต่ยังทำหน้าที่เป็นช่องสัญญาณในการส่งผ่านข้อมูลได้อีกด้วย การส่งข้อมูลผ่านโครงข่ายโทรศัพท์ จะต้องมีอุปกรณ์ที่เรียกว่าโมเด็ม การส่งข้อมูลแบ่งออกได้เป็นสองชนิด คือ แบบฮาล์ฟดูเพล็กซ์ (Half duplex) และฟูลดูเพล็กซ์ (Full duplex) แบบฮาล์ฟดูเพล็กซ์ข้อมูลจะถูกส่งผ่านไปทิศทางเดียวตามช่วงเวลาที่มีการจัดสรรให้ ส่วนแบบฟูลดูเพล็กซ์ ข้อมูลด้านส่งและรับจะถูกแยกออกจากกัน ปริมาณข้อมูลที่ส่งผ่านจึงมีจำนวนมากกว่า คล้ายกับการรับส่งในวงจรที่ตาย แต่ใช้สายโทรศัพท์เพียงคู่เดียว (วงจรสองสาย) ดังนั้น สัญญาณที่ส่งผ่านเข้าไปในวงจรโทรศัพท์จะเกิดการย้อนกลับมาทางด้านรับของโมเด็ม ในรูปแบบของสัญญาณสะท้อนกลับที่รบกวนการทำงานของโมเด็ม ซึ่งเป็นอุปสรรคสำคัญของการสื่อสารข้อมูลแบบฟูลดูเพล็กซ์บนวงจรโทรศัพท์แบบสองสาย

วิธีการแยกสัญญาณด้านรับออกจากสัญญาณสะท้อนกลับด้านส่งในโมเด็ม มีอยู่สองวิธี คือ TDM (Time-Division Multiplexing) และ FDM (Frequency-Division Multiplexing) TDM เป็นระบบการส่งในรูปแบบของฮาล์ฟดูเพล็กซ์ โดยจะแบ่งเวลาให้โมเด็มแต่ละตัวทำงานจนจบกระบวนการส่งข้อมูล เมื่อโมเด็มตัวที่หนึ่งทำหน้าที่ส่งข้อมูล โมเด็มตัวที่สองจะทำหน้าที่รับข้อมูลอย่างเดียวไม่มีการส่งข้อมูลใด ๆ ดังนั้น จะไม่มีสัญญาณสะท้อนกลับ มาที่ภาครับของโมเด็มตัวที่สอง ส่วนในระบบ FDM นั้น จะใช้การแบ่งแบนด์วิดท์ของช่องสัญญาณโทรศัพท์ในส่วนของ การเชื่อมต่อสัญญาณเสียงพูด (Voice band) ที่มีความถี่อยู่ระหว่าง 300 Hz ถึง 3,400 Hz ออกเป็นสองช่วง คือช่วงแถบความถี่สูงและแถบความถี่ต่ำ เช่น โมเด็มตัวที่หนึ่งใช้แถบความถี่สูงในการส่ง โมเด็มตัวที่สองก็จะใช้แถบความถี่ต่ำในการส่ง ดังนั้น สัญญาณสะท้อนกลับและสัญญาณที่รับได้ของโมเด็มทั้งคู่จะอยู่ในช่วงแถบความถี่ต่างกัน จึงเป็นการแยกสัญญาณรับกับส่งออกจากกันด้วยการกรองที่แม่นยำได้ เช่น ใน echo cancellation modem

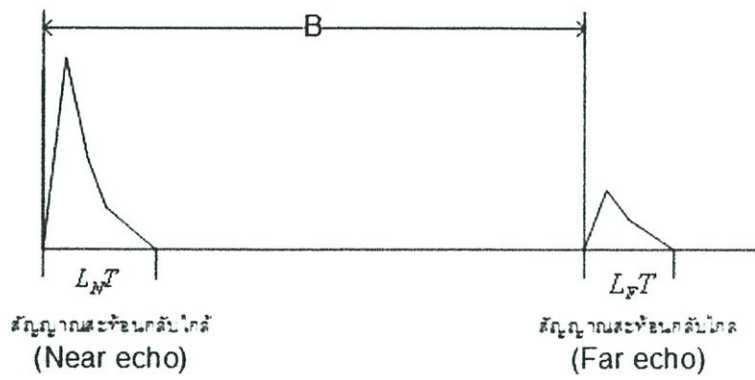
2.3.1 การขจัดสัญญาณสะท้อนกลับสำหรับการส่งข้อมูล (Echo cancellation for data transmission)



รูปที่ 2.4 การสื่อสารข้อมูลด้วยโมเด็มและวงจรขจัดสัญญาณสะท้อนกลับ

วงจรขจัดสัญญาณสะท้อนกลับสำหรับการส่งข้อมูลจะถูกติดตั้งไว้ในโมเด็ม ซึ่งจะแตกต่างจากวงจรขจัดสัญญาณสะท้อนกลับเสียงพูด ที่วงจรขจัดสัญญาณสะท้อนกลับถูกติดตั้งไว้ในโครงข่ายโทรศัพท์แล้ว รูปที่ 2.4 แสดงการขจัดสัญญาณสะท้อนกลับของโมเด็มที่เชื่อมต่อกันผ่านวงจรสองสายไปยังโครงข่ายโทรศัพท์ วงจรส่งจะเปลี่ยนข้อมูลเป็นสัญญาณที่เหมาะสมที่ใช้ในการส่งหรือสัญญาณ Tx ส่งไปบนโครงข่ายโทรศัพท์ โดยผ่านไฮบริคภายในโมเด็ม วงจรด้านรับจะรับสัญญาณ Rx ผ่านไฮบริค ซึ่งสัญญาณ Rx จะเสียหายจากสัญญาณสะท้อนกลับและการรบกวนภายในช่องสัญญาณ วงจรขจัดสัญญาณสะท้อนกลับจะประมาณค่าผลตอบสนองของเส้นทางการสะท้อนกลับ และ สังเคราะห์สัญญาณสะท้อนกลับขึ้นมาใหม่จากสัญญาณที่ส่งออกไปและการจำลองเส้นทางการสะท้อนกลับ สัญญาณสะท้อนกลับที่อยู่ในสัญญาณ Rx จะถูกขจัด โดยการนำสัญญาณสะท้อนกลับที่สังเคราะห์ขึ้นลบออกจากสัญญาณสะท้อนกลับที่รับได้

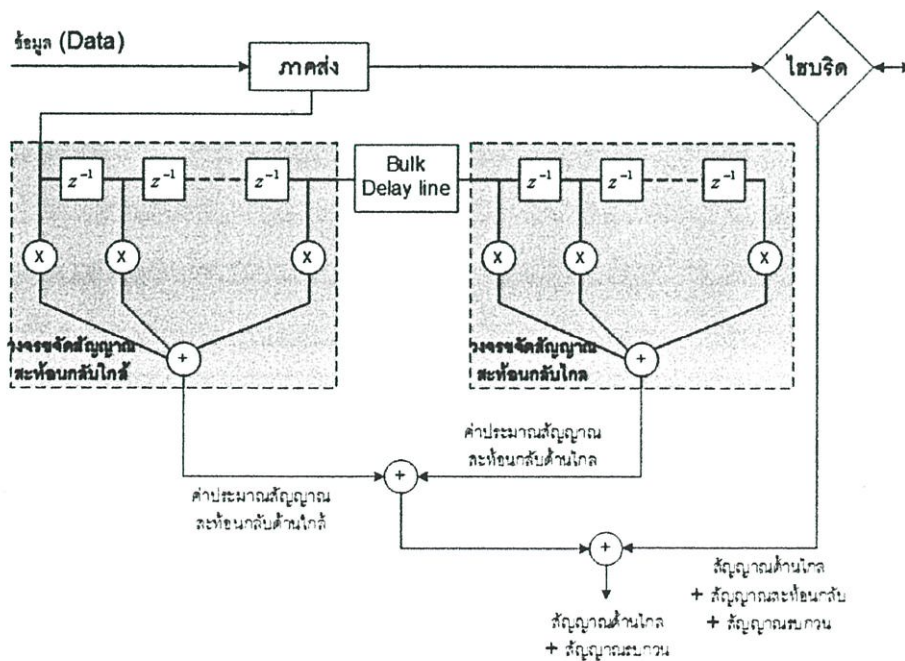
เนื่องจากสัญญาณสะท้อนกลับกลับทั้งด้านไกลและใกล้ ยังคงมีอยู่ในโครงข่ายโทรศัพท์ ผลรวมของผลตอบสนองเส้นทางการสะท้อนกลับทั้งหมด อาจจะมีค่ายาวนานมากกว่า 600 ms



คุณลักษณะของสัญญาณสะท้อนกลับใกล้และไกล
 สัญญาณสะท้อนกลับใกล้:
 จากไฮบริดโมเด็มและคุณสมบัติของอิน
 หนองเวลาสั้น, มีกำลังแรง
 สัญญาณสะท้อนกลับไกล:
 จากคุณสมบัติของสายส่งที่ด้านไกล
 มีกำลังอ่อนกว่าสัญญาณ RX, อาจมีการเลื่อนเฟส

รูปที่ 2.5 คุณลักษณะของสัญญาณสัญญาณสะท้อนกลับทั้งใกล้และไกล

จากรูปที่ 2.5 จะเห็นได้ว่า สัญญาณสะท้อนกลับทั้งใกล้และไกลมีช่วงห่างกันมาก ดังนั้นในการ ออกแบบวงจรขจัดสัญญาณสะท้อนกลับ สามารถแบ่งการทำงานออกเป็นช่วงสั้น ๆ สองส่วน แต่ ละส่วนจะมีความกว้างประมาณ 10 ms ถึง 20 ms ซึ่งจะเรียกว่าวงจรขจัดสัญญาณสะท้อนกลับใกล้ และวงจรขจัดสัญญาณสะท้อนกลับไกล มีโครงสร้างดังรูปที่ 2.6



รูปที่ 2.6 โครงสร้างทั่วไปของวงจรขจัดสัญญาณสะท้อนกลับใน โมเด็ม

2.3.2 ข้อบังคับของการขจัดสัญญาณสะท้อนกลับ

ในหัวข้อนี้จะใช้มาตรฐาน CCITT V.39 [12] ซึ่งเป็นมาตรฐานของการขจัดสัญญาณสะท้อนในโมเด็มเป็นตัวอย่างในการอธิบาย ที่ความเร็วในการส่งข้อมูลที่ 2400 baud หรือ 9600 บิต ต่อวินาที ใช้การเข้ารหัสสัญญาณแบบ trellis-coded (TCM) จากมาตรฐานของ V.32 มีค่า $SNR \geq 18$ dB ซึ่งแน่ใจได้ว่าสัญญาณสะท้อนกลับจะไม่มีผลกระทบต่อการทำงานของภาครับโดยทั่วไป มักจะให้สัญญาณสะท้อนกลับมีค่าต่ำกว่าระดับสัญญาณรบกวนที่ยอมรับได้อยู่ 6 dB ระดับของสัญญาณที่ใช้ส่งใน local loop ปกติอยู่ที่ -9 dBm และ ความกว้างระดับสัญญาณที่ภาครับมีค่าระหว่าง -43 dBm ถึง -9 dBm ที่ระดับต่ำสุด (-43 dBm) ซึ่งกำหนดเป็นข้อบังคับของการขจัดสัญญาณสะท้อนที่เข้มงวดที่สุด ระดับสัญญาณสะท้อนกลับที่เหลืออยู่ (residual echo) จะอยู่ที่ -67 dBm ($= -43 - 18 - 6$) หรือต่ำกว่านั้น ซึ่งก็คือการสูญเสียจากการลดทอนสัญญาณของไฮบริคั่นเอง กล่าวคือ การลดทอนจากวงจรภาคส่งไปภาครับมีค่าเท่ากับ 9 dB และ การสูญเสียน้อยที่สุดของไฮบริคมีค่าเท่ากับ 6 dB ระดับสัญญาณสะท้อนกลับสูงสุดที่ภาครับผ่านไฮบริคคือ -15 dBm ดังนั้น วงจรขจัดสัญญาณสะท้อนกลับจะถูกบังคับให้ต้องขจัดสัญญาณสะท้อนกลับที่ 52 dB เพื่อลดทอนสัญญาณสะท้อนกลับที่มีระดับสัญญาณสูงที่สุด (-15 dBm) และ ระดับสัญญาณสะท้อนกลับที่เหลืออยู่ (-67 dBm) อัตราส่วนของการสะท้อนกลับก่อนและหลังการขจัด จะเรียกว่า echo return loss enhancement (ERLE) ของวงจรขจัดสัญญาณสะท้อนกลับ

จากสมมุติฐานที่แน่นอนแล้วว่า สัญญาณสะท้อนกลับที่เกิดจากสัญญาณที่ส่งออกมาเปลี่ยนแปลงเป็นแบบลิเนียร์ ซึ่งเป็นไปได้ในทางทฤษฎีที่จะทำให้ค่า ERLE มีค่าได้ไม่จำกัด แต่ในทางปฏิบัติ จะมีตัวแปรหลัก ๆ อยู่สองตัวแปร สิ่งแรก คือ เส้นทางของสัญญาณสะท้อนมักจะเป็นแบบนอนลิเนียร์ ดังนั้น แบบจำลองลิเนียร์ก็จะทำงานได้ในทางอุดมคติเท่านั้น นั่นคือสัญญาณสะท้อนแบบนอนลิเนียร์ ไม่สามารถขจัดได้ด้วยวงจรขจัดสัญญาณสะท้อนแบบลิเนียร์ได้ สิ่งที่สอง คือ วงจรขจัดสัญญาณสะท้อนในโมเด็ม ปกติจะออกแบบโดยใช้เครื่องมือประมวลผลสัญญาณแบบดิจิทัล ความถูกต้องของการคำนวณเส้นทางการสะท้อนมักจะถูกจำกัดจากความเที่ยงตรงในการคำนวณ

สัญญาณสะท้อนที่แรงที่สุด คือ ส่วนของสัญญาณส่งร่วผ่านไฮบริคเข้ามาทางภาครับ ซึ่งก็คือ ส่วนของสัญญาณสะท้อนกลับด้านใกล้ จะมีความแรงมากกว่าสัญญาณที่รับได้ ในขณะที่สัญญาณสะท้อนกลับด้านไกล มีความแรงน้อยกว่า แต่อาจจะกลายเป็นการมอดูเลตที่เป็นผลสืบเนื่องมาจากความไม่สมบูรณ์ในการเชื่อมต่อระบบของวงจรเข้ารหัสและถอดรหัสในระบบนำพาสัญญาณได้ และ เนื่องจากการเข้ารหัสเป็นการทำงานแบบนอนลิเนียร์ จึงต้องมีการชดเชยแบบนอนลิเนียร์เพื่อขจัดสัญญาณสะท้อนด้านไกลด้วย

2.4 การขจัดสัญญาณสะท้อนกลับใน Subscriber-loop แบบดิจิทัล

การส่งข้อมูลเกือบทั้งหมดบนโครงข่ายโทรศัพท์แบบใหม่จะอยู่ในรูปแบบดิจิทัล ส่วนที่ยังเป็นอนาล็อกอยู่ก็คือส่วนที่เป็น local loop จะเรียกใหม่ว่า subscriber loop วงจรขจัดสัญญาณสะท้อนที่ใช้ในการส่งข้อมูลความเร็วสูงในส่วนนี้เรียกว่า วงจรขจัดสัญญาณสะท้อน subscriber loop การส่งข้อมูลความเร็วสูงใน subscriber loop เป็นการให้บริการของโครงข่ายบริการร่วมดิจิทัล ISDN (Integrated Service Digital Network) มีอัตราความเร็วในการส่งข้อมูลอยู่ที่ 144 kbps ในโครงข่าย ISDN จะมีการขจัดสัญญาณสะท้อนกลับอยู่เพียงชนิดเดียวคือ สัญญาณสะท้อนกลับด้านไกล เนื่องจากไม่มีสัญญาณสะท้อนกลับด้านไกล และยังมีแหล่งจ่ายสัญญาณนาฬิกาจากชุมสายโทรศัพท์ ทำหน้าที่ควบคุมการส่งข้อมูลทั้งสองทิศทาง ทำให้สัญญาณนาฬิกาของภาครับและภาคส่งสอดคล้อง (Synchronous) กันเสมอ การออกแบบวงจรขจัดสัญญาณสะท้อนกลับจึงทำได้ง่ายขึ้น ในทางกลับกัน จากการที่อัตราการส่งข้อมูลที่สูงขึ้น วงจรขจัดสัญญาณสะท้อนกลับจะมีการคำนวณที่มากขึ้น ระดับสัญญาณสะท้อนกลับก็จะสูงขึ้นตามไปด้วย อัตราส่วนการลดลงของสัญญาณสะท้อนจะอยู่ที่ 70 dB ส่งผลให้มีการนำเอาตัวกรองแบบ FIR มาขจัดสัญญาณสะท้อนแบบลิเนียร์ ร่วมกับตัวกรองแบบ IIR ใช้ขจัดสัญญาณสะท้อนแบบนอนลิเนียร์ในโครงข่าย ISDN

2.5 วงจรขจัดสัญญาณสะท้อนแบบ Data-driven Nyquist

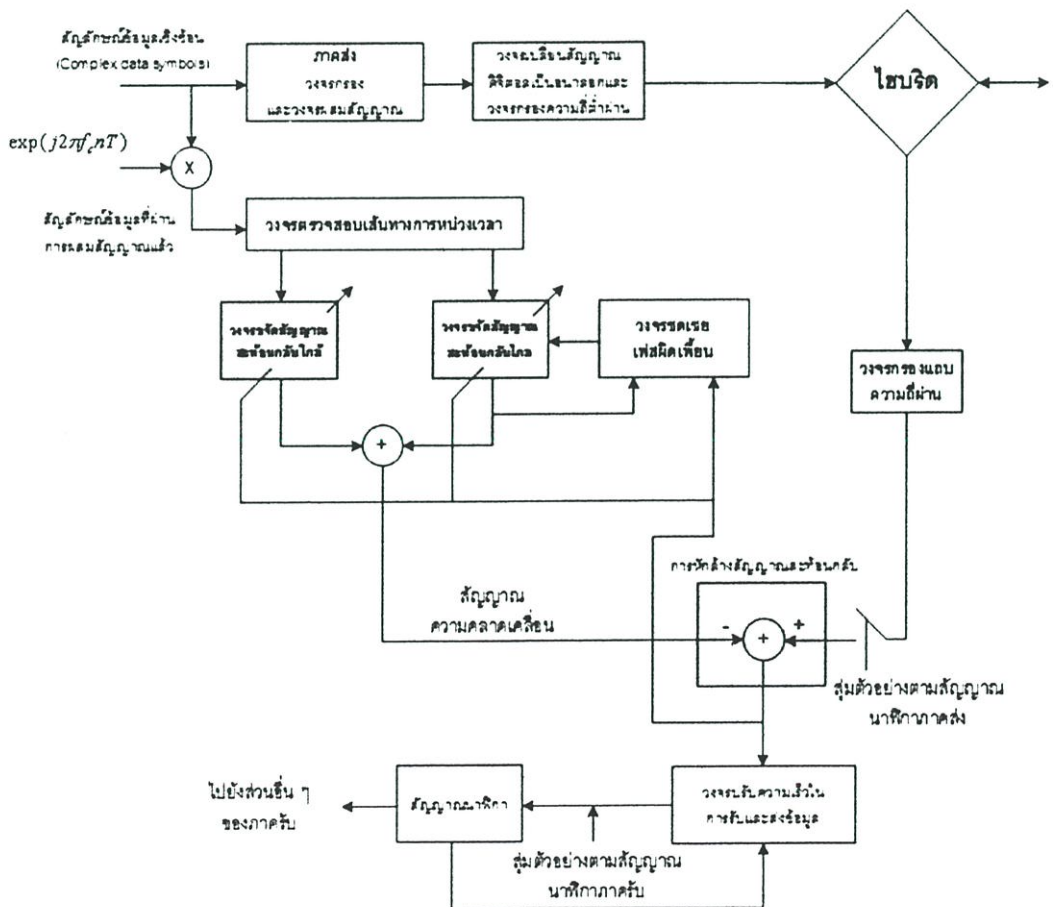
ในส่วนนี้จะยกตัวอย่างชนิดของวงจรขจัดสัญญาณสะท้อนกลับที่เรียกว่า Data driven echo canceller [13]

2.5.1 โครงสร้างของวงจรขจัดสัญญาณข้อมูลสะท้อนกลับในโมเด็ม

จากทฤษฎีเบื้องต้นในการขจัดสัญญาณสะท้อนกลับ คือ การสังเคราะห์สัญญาณสะท้อนกลับให้เหมือนกับสัญญาณสะท้อนกลับที่อยู่ในสัญญาณที่รับ ได้มีความถูกต้องมากที่สุด ด้วยวิธีการลิเนียร์คอมไบเนชัน ตัวอย่างของสัญญาณที่ส่งออกมา สัญญาณสะท้อนกลับที่สังเคราะห์ขึ้นจะนำไปลบออกจากสัญญาณที่รับเข้ามาเพื่อขจัดสัญญาณสะท้อน รูปแบบของสัญญาณสะท้อนกลับที่สังเคราะห์ขึ้น อาจจะใช้การสุ่มตัวอย่างของสัญลักษณ์ข้อมูลทางด้านส่ง (Tx data symbol) หรือสัญญาณส่งของวงจรถ่ายความถี่ผ่าน (Pass band Tx signal) วงจรขจัดสัญญาณสะท้อนกลับที่ใช้สัญลักษณ์ข้อมูล Tx ในการสร้าง จะเรียกว่า วงจรขจัดสัญญาณสะท้อนกลับแบบ Data-driven (Data-driven echo canceller) เนื่องจากข้อมูลที่อยู่ในเส้นทางสัญญาณอินพุตมีแถบความถี่ราบเรียบเสมอกัน (White) ดังนั้นค่าไอเกนเมตริกซ์ของข้อมูลอัดสัมพันธ์จึงมีค่าเท่ากันทั้งหมด ผลที่ตามมาคือ ไม่ว่าจะใช้อัลกอริทึมในการปรับตัว (Adaptive algorithm) แบบไหนก็ตาม วงจรขจัดสัญญาณสะท้อนกลับแบบ Data-driven จะให้สมรรถนะการลู่เข้าและการติดตามสัญญาณได้ดี

การหักล้างกันของสัญญาณสะท้อนกลับสามารถทำได้ตามอัตราการแบ่งขั้นสัญญาณในการสุ่มสัญลักษณ์ข้อมูลภาครับ ซึ่งซึ่งโครโนซ์กับอัตราการแบ่งขั้นสัญญาณภาคส่งอีกด้านหนึ่ง[13,14] หรือจะใช้อัตราการสุ่มสัญญาณที่รับเข้ามาตามเงื่อนไขแบบ Nyquist กล่าวคือ อัตราการสุ่มตัวอย่างอย่างน้อยต้องเป็นสองเท่าของความถี่สูงสุดของสัญญาณ [12,14] โดยปกติจะเลือกค่าที่เป็นจำนวนเต็มของการคูณกับอัตราการแบ่งขั้นสัญญาณสัญลักษณ์ข้อมูล Tx ดังนั้น การขจัดสัญญาณสะท้อนกลับจากกรรมวิธีที่กล่าวมาจึงเรียกว่า symbol rate echo cancellation และ Nyquist echo cancellation ตามลำดับ symbol rate echo cancellation จะใช้การขจัดสัญญาณสะท้อนกลับในโครงข่าย ISDN ส่วน Nyquist echo cancellation จะใช้ในเฉพาะในโมเด็ม เนื่องจากลดความยุ่งยากในการซิงโครไนซ์อัตราการแบ่งขั้นสัญญาณจากภาคส่งทั้งสองด้าน ซึ่งมักจะไม่สามารถตั้งกันโมเด็มแต่ละตัว

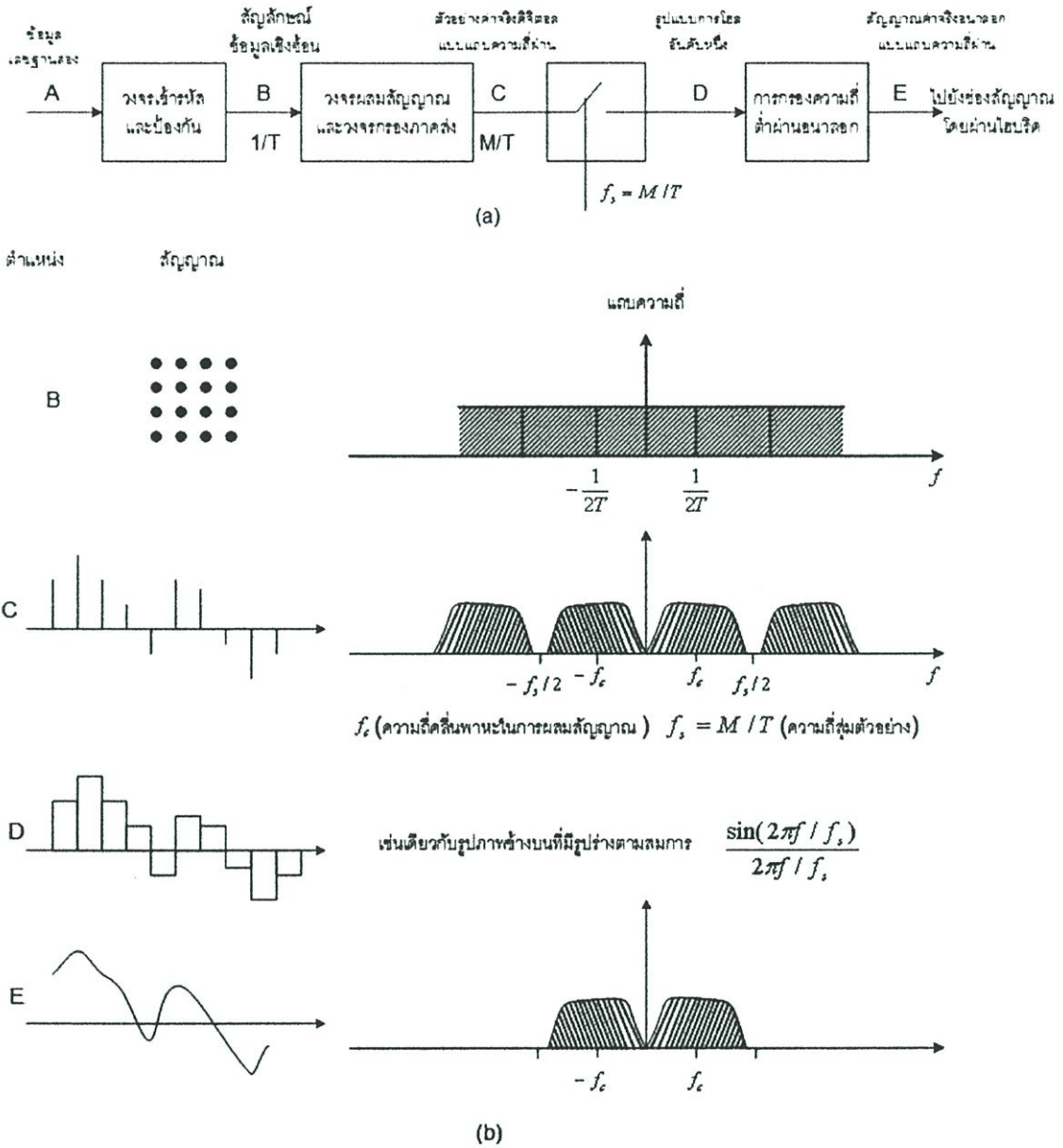
2.5.2 โครงสร้างและการทำงานของวงจรขจัดสัญญาณสะท้อนแบบ data-driven Nyquist ในโมเด็ม



รูปที่ 2.7 บล็อกไดอะแกรมของโมเด็มที่มีส่วนของวงจรขจัดสัญญาณสะท้อนกลับแบบ Data-driven

วงจรถัดสัญญาณสะท้อนกลับแบบ Nyquist สามารถแบ่งออกเป็นสองชนิด คือ วงจรถัดสัญญาณสะท้อนแถบความถี่ผ่านจริง (Real passband echo canceller) และวงจรถัดสัญญาณสะท้อนแบบวิเคราะห์ (Analytic echo canceller) ซึ่งแบบวิเคราะห์นี้ ยังสามารถแยกออกได้เป็นสองชนิด คือ แบบวิเคราะห์แถบความถี่ผ่าน (Passband –analytic) และ แบบวิเคราะห์แถบความถี่ฐาน (Baseband –analytic)

จากรูปที่ 2.7 โมเด็มจะถูกต่อเข้ากับอุปกรณ์ส่งข้อมูลปลายทาง DTE (Data terminal equipment) เช่น เครื่องคอมพิวเตอร์ปลายทาง ข้อมูลที่เป็นเลขฐานสองจาก DTE จะถูกสุ่มตัวอย่างด้วยวงจรป้องกันข้อมูล(Scrambler)ก่อน แล้วจึงทำการเข้ารหัสโดยวงจรเข้ารหัสช่องสัญญาณโทรศัพท์และนำค่าสัญลักษณ์ข้อมูลเชิงซ้อนมาเรียงใหม่ในรูปแบบอนุกรม ซึ่งก็คือ วิธีการสุ่มตัวอย่าง จากผลของการสุ่มตัวอย่างนี้ ความสัมพันธ์ของลำดับสัญลักษณ์ที่เรียงต่อกันก็คือ ฟังก์ชันอิมพัลส์แบบหนึ่งเช่นเดียวกับ ลำดับของข้อมูลชุดหนึ่ง ๆ ที่เรียกว่า white เนื่องจากมีขนาดของแถบกำลังงานราบเรียบเท่ากันหมด โดยนิยามแล้ว แถบกำลังงาน ก็คือ การแปลงฟูเรียร์ของฟังก์ชันอิมพัลส์ของลำดับข้อมูลนั่นเอง จุดประสงค์ของการสุ่มคือ เพื่อให้การส่งข้อมูลเข้าไปในช่องสัญญาณโทรศัพท์ทำได้ง่ายและมีประสิทธิภาพ และจากทฤษฎีวงจรรองแบบอะแคปทีฟที่ว่า การหาค่าตอบจะดีที่สุดเมื่อสัญญาณอินพุตเป็นสัญญาณที่มีความถี่ราบเรียบเสมอกัน ดังนั้น จากตัวอย่างของการสุ่มตัวอย่างที่กล่าวมานี้จึงเป็นการเพิ่มประสิทธิภาพการหาค่าตอบของวงจรถัดสัญญาณสะท้อนกลับอีกด้วย

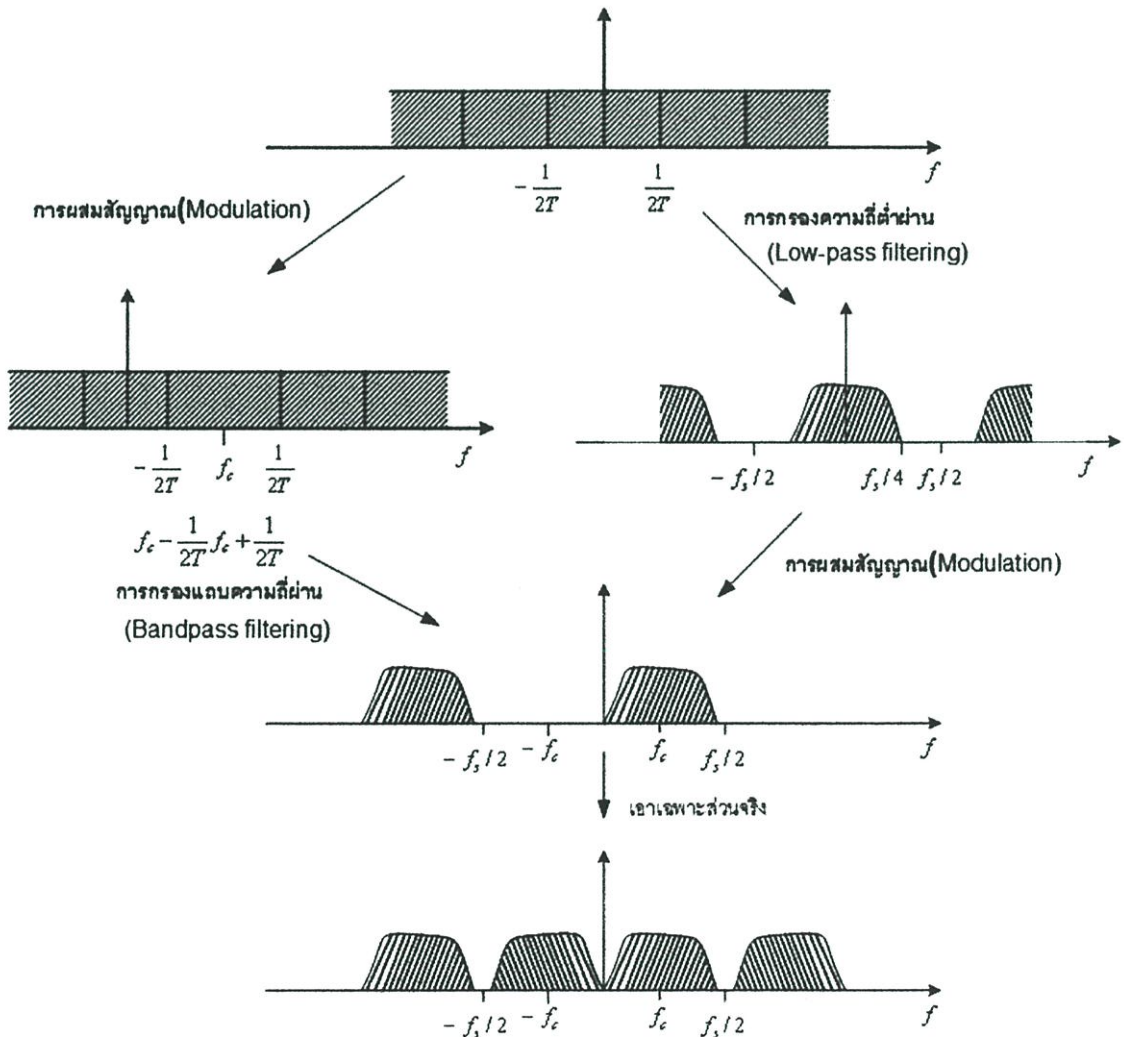


รูปที่ 2.8 การสร้างสัญญาณส่ง โมเด็ม:(a) การสร้างสัญญาณส่ง Tx (T , ความกว้างสัญญาณ, $M > 1$) (b) สัญญาณต่าง ๆ และ แถบความถี่

จากรูปที่ 2.8 เป็นการอธิบายขั้นตอนการทำงานในการสร้างสัญญาณข้อมูลที่จะใช้ส่งผ่านเข้าไปในระบบโทรศัพท์ ทั้งทางเวลาและความถี่ ดังนี้

ค่าข้อมูลเชิงซ้อน คือ สัญลักษณ์ต่าง ๆ ของข้อมูลที่สร้างขึ้นจากความสัมพันธ์ของข่าวสารที่จะใช้ในการส่ง ที่มีจำนวนสัญลักษณ์ที่แน่นอน เรียกว่า อัตราบอด(Baud rate) หรือความเร็วในการส่งข้อมูล มีค่าเท่ากับ $1/T$ เมื่อ T คือความกว้างของสัญลักษณ์ หลังจากนั้นจะถูกผสมเข้ากับความถี่พาหะ f_c ด้วยวงจรผสมสัญญาณ และสร้างรูปร่างของแถบความถี่ขึ้นด้วยวงจรกรองภาคส่ง

หลังจากขั้นตอนการทำงานนี้แล้ว แถบความถี่ของสัญญาณที่ได้จะถูกกำหนดโดยความถี่ของวงจรกรองภาคส่ง และเหมาะสมสำหรับการส่งไปในช่องสัญญาณ โทรศัพท์ เป็นที่น่าสังเกตว่า ทั้งการผสมสัญญาณและการกรอง ถูกออกแบบให้อยู่ในรูปแบบดิจิทัล เนื่องจากข้อมูลที่เข้ามาเป็นข้อมูลดิจิทัลเชิงซ้อน การผสมสัญญาณจึงทำได้ง่ายด้วยการวนแบบตัวอย่างต่อตัวอย่าง ส่วนของสัญญาณดิจิทัลจริง ๆ ที่ถูกกรองและถูกผสมสัญญาณแล้ว จะเรียกว่า สัญญาณแถบความถี่ผ่านจริง (Real passband signal) โดยทั่วไปความถี่ในการสุ่มตัวอย่างจะอยู่ที่ M/T เมื่อ M คือจำนวนเต็มที่มีค่า $M \geq 3$ หลังจากนั้นสัญญาณแถบความถี่ผ่านจริงจะถูกเปลี่ยนให้อยู่ในรูปอนุภาคด้วยวงจรแปลงสัญญาณดิจิทัลไปเป็นสัญญาณอนุภาค และส่งไปยังด้านส่งของไฮบริคเพื่อส่งไปยังโมเด็มปลายทางผ่านช่องสัญญาณ โทรศัพท์ต่อไป



รูปที่ 2.9 การกรองและการผสมสัญญาณส่ง

การผสมสัญญาณและการกรองสัญญาณที่ภาคส่ง จากรูปที่ 2.8 สามารถทำได้สองทาง ดังรูปที่ 2.9 แนวทางแรก อาจะเริ่มต้นด้วยการผสมสัญญาณสัญลักษณ์ข้อมูลเชิงซ้อนก่อน(ด้านซ้ายมือ) หลังจากนั้นจึงทำการแยกให้เป็นรูปร่าง โดยใช้วงจรกรองแบบแถบความถี่ผ่าน (Bandpass filter) ก็จะได้สัญญาณเชิงซ้อนที่ผ่านการมอดูเลตแล้ว ซึ่งจะเรียกว่า สัญญาณวิเคราะห์แถบความถี่ผ่าน (Passband-analytic signal) ทางเลือกที่สองคือ เริ่มด้วยการสร้างรูปร่างสัญลักษณ์ข้อมูลเชิงซ้อนก่อน โดยใช้วงจรกรองความถี่ต่ำผ่าน (Lowpass filter) สัญญาณที่ถูกกรองแล้วจะเรียกว่า สัญญาณวิเคราะห์แถบความถี่ฐาน (Baseband-analytic signal) หลังจากนั้นจึงจะทำการผสมสัญญาณเพื่อสร้างสัญญาณวิเคราะห์แถบความถี่ผ่าน(Passband-analytic signal) กระบวนการนี้จะแสดงในรูปด้านขวาของรูปที่ 2.9 สัญญาณดิจิทัลจริงของแถบความถี่ผ่าน ซึ่งก็คือส่วนจริงของสัญญาณวิเคราะห์แถบความถี่ผ่าน (Passband-analytic signal) จะถูกส่งเข้าไปยังวงจรแปลงสัญญาณจากดิจิทัลไปเป็นสัญญาณอนาลอกต่อไป

จะสังเกตได้ว่า วงจรกรองหรือวงจรผสมสัญญาณส่ง เป็นระบบที่มีอัตราการสุ่มหลายอัตรา โดยเฉพาะอัตราการสุ่มเอาต์พุตของสัญญาณแถบความถี่ผ่าน (Passband signal) จะต้องสูงกว่าอัตราการสุ่มแบบ Nyquist เนื่องจากเอาต์พุตของวงจรกรองจะมีเฉพาะส่วนจริงเท่านั้นที่นำไปใช้งาน ดังนั้น ในการเลือกอัตราการสุ่มตัวอย่างสัญญาณเอาต์พุตของวงจรกรองภาคส่งจึงมักจะเลือกเป็นค่าจำนวนเต็มการคูณของความเร็วสัญลักษณ์ในการส่งข้อมูลอินพุต ($1/T$) วงจรกรองภาคส่งจึงจะแน่ใจได้ว่าไม่มีการรบกวนของสัญญาณที่ไม่ต้องการหรือไม่มีองค์ประกอบความถี่ที่มีค่าเป็นลบที่จะทำให้เกิดการ aliasing กับแถบความถี่ใช้งานได้

บทที่ 3

ตัวกรองอะแดปทีฟและการประยุกต์ใช้งาน

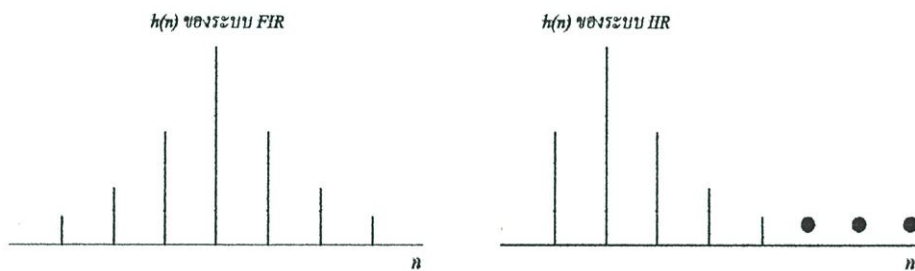
ในการประมวลผลสัญญาณ และการควบคุม ซึ่งสัญญาณต่าง ๆ และฟังก์ชันถ่ายโอนจะไม่เปลี่ยนแปลงตามเวลา การออกแบบตัวกรองคงที่และตัวควบคุมแบบคงที่จะให้ผลที่ต้องการในช่วงเวลาหนึ่งเท่านั้น แต่ในความเป็นจริงสัญญาณต่าง ๆ ฟังก์ชันการถ่ายโอนและสภาพแวดล้อมในการทำงานของระบบมักจะเปลี่ยนแปลงตามเวลา เช่น การควบคุมสัญญาณรบกวน และการสั่นสะเทือน ระบบตัวกรองแบบคงที่และตัวควบคุมแบบคงที่จึงไม่สามารถควบคุมได้ตลอดเวลา ดังนั้นจึงมีความจำเป็นในการออกแบบตัวกรองหรือตัวควบคุมแบบอะแดปทีฟให้ทำงานได้ดีในขณะที่สภาพแวดล้อมมีการเปลี่ยนแปลงอยู่เสมอ โดยปกติมักจะทำการปรับสัมประสิทธิ์ตัวกรองดิจิทัลหรือตัวควบคุมแบบออนไลน์ ด้วยการกำหนดค่าสัมประสิทธิ์ที่เหมาะสมไว้ล่วงหน้า ตัวกรองอะแดปทีฟอาจจะสร้างขึ้นจากโครงสร้างตัวกรองพื้นฐานต่าง ๆ โครงสร้างตัวกรองอะแดปทีฟที่มีใช้กันอย่างกว้างขวางก็คือ โครงสร้างทรานสเวอร์ซอล (Transversal structure) เนื่องจากมีเสถียรภาพและง่ายต่อการวิเคราะห์ โครงสร้างแบบลิเนียร์คอมไบเนอรัลก็เป็นรูปแบบทั่วไปของโครงสร้างทรานสเวอร์ซอล และถูกนำมาใช้บ่อยมากในการประมวลผลสัญญาณแบบอาร์เรย์ ตัวกรองอะแดปทีฟแบบรีเคอร์ซีฟนำมาใช้ในการยกระดับสัญญาณทางสาย โดยการสร้างสัญญาณที่ผิดเพี้ยนไปแล้วขึ้นมาใหม่และ ยกกระตบช่องสัญญาณ รวมทั้งโครงสร้างแบบแลตทิซซึ่งเป็นที่นิยมใช้ในงานที่ต้องคาดคะเนการปรับตัวอย่างสม่ำเสมอ ส่วนอัลกอริทึมที่ใช้ในการปรับสัมประสิทธิ์ของตัวกรองเหล่านี้ สามารถแบ่งออกได้เป็นสองแนวทางหลัก ๆ แนวทางแรกเกิดจากค่าที่เหมาะสมทางสถิติ ได้แก่ อัลกอริทึมแบบ LMS (Least Mean Squares algorithm) และอนุพันธ์ย่อย แนวทางที่สองเกิดจากค่าที่เหมาะสมจากการสุ่ม ได้แก่ อัลกอริทึมแบบ RLS (Recursive Least Squares algorithm) และอนุพันธ์ย่อย

วงจรรอง ตัวกรอง หรือ ฟิลเตอร์ (Filters) หมายถึง รูปแบบอุปกรณ์วงจรอิเล็กทรอนิกส์ หรือ โปรแกรมคอมพิวเตอร์ ที่นำมาประยุกต์ใช้งานด้านข้อมูลข่าวสารกับกลุ่มของข้อมูลที่มีสัญญาณรบกวนมาก เพื่อที่จะดึงเอาข้อมูลที่สนใจในช่วงหนึ่งออกมาใช้งาน กระบวนการทำงานหลักของตัวกรอง มีอยู่สามลักษณะ คือ

- การกรอง (Filtering) หมายถึง การนำเอาข้อมูลข่าวสารเฉพาะที่ให้ความสนใจในช่วงเวลาใดเวลาหนึ่งเท่านั้น
- การทำให้เรียบ (Smoothing) ต่างจากการกรอง โดยที่กลุ่มข่าวสารที่ต้องการไม่จำเป็นที่จะต้องมียู่ในช่วงเวลาใดเวลาหนึ่งเท่านั้น ข้อมูลที่ได้มาหลังจากการสังเกตการณ์ก็สามารถนำมาใช้งานในการวิเคราะห์เพื่อสร้างข้อมูลขึ้นมาใหม่ได้ หมายถึงข้อมูลข่าวสาร

ที่ต้องการอาจจะมีการหน่วงเวลาเกิดขึ้น จะใช้ข้อมูลทั้งสองชุดมาทำการวิเคราะห์ร่วมกัน ทำให้ผลที่ได้มีความแม่นยำมากกว่าการกรอง

- และการคาดคะเน (Prediction) เป็นการคาดคะเนล่วงหน้าจากการประมาณค่าข้อมูลที่มีอยู่ สามารถคาดคะเนถึงข้อมูลที่ต้องการในอนาคตได้ โดยใช้ข้อมูลในอดีตและข้อมูลปัจจุบัน
- ตัวกรองความถี่ นับเป็นวงจรีเล็กทรอนิกส์ที่มีความสำคัญมาก ไม่ว่าจะทางด้านอิเล็กทรอนิกส์ การสื่อสาร การควบคุม และทางด้านเครื่องมือแพทย์ ปัจจุบันตัวกรองความถี่แบ่งออกเป็นสองรูปแบบตามลักษณะของการทำงาน คือตัวกรองความถี่แบบแอนะล็อก (Analog filter) และตัวกรองความถี่แบบดิจิทัล (Digital filter) นอกจากนั้นแล้ว ยังสามารถแบ่งตัวกรองความถี่ตามคุณลักษณะของผลตอบสนองความถี่ (Frequency response) ได้อีกคือ ตัวกรองความถี่แบบผ่านต่ำ (Low-pass filter, LPF) ตัวกรองความถี่แบบผ่านสูง (High-pass filter, HPF) ตัวกรองความถี่แบบผ่านแถบ (Band-pass filter, BPF) ตัวกรองความถี่แบบหยุดแถบ (Band-stop filter, BSF) หรือ Notch filter และตัวกรองความถี่แบบผ่านทุกความถี่ (All-pass filter)



รูปที่ 3.1 ผลตอบสนองต่อสัญญาณอิมพัลส์ของตัวกรองแบบ FIR และแบบ IIR

ตัวกรองความถี่แบบดิจิทัล (Digital filter) แบ่งตามผลตอบสนองอิมพัลส์ของระบบได้เป็นสองชนิด คือ ตัวกรองความถี่แบบผลตอบสนองอิมพัลส์จำนวนจำกัดหรือตัวกรองแบบ FIR (Finite impulse response) และตัวกรองความถี่แบบผลตอบสนองอิมพัลส์จำนวนไม่จำกัดหรือตัวกรองแบบ IIR (Infinite impulse response) โดยตัวกรองความถี่แบบ FIR นั้น ฟังก์ชันถ่ายโอน (Transfer function) จะมีเฉพาะซีโร (Zero) ไม่มีโพล (Pole) ทำให้ระบบมีความเสถียรภาพแน่นอน และมีผลตอบสนองทางเฟสเป็นเชิงเส้น (Linear phase) แต่มีข้อเสียคือ ต้องใช้จำนวนอันดับ (Order) สูงจึงจะทำให้ตัวกรองทำงานได้ดี ส่วนตัวกรองความถี่แบบ IIR นั้น ฟังก์ชันถ่ายโอนจะประกอบด้วยซีโรและโพล ทำให้มีปัญหาเรื่องเสถียรภาพที่ไม่แน่นอน แต่ถ้ามีการออกแบบที่ดี จะทำให้ปัญหาดังกล่าวน้อยลง หรือไม่เกิดขึ้นเลย ข้อดีของตัวกรองความถี่แบบ IIR เมื่อเทียบกับแบบ FIR คือ ที่สมรรถนะ (Performance) เท่ากัน ตัวกรองความถี่แบบ IIR จะใช้จำนวนอันดับน้อยกว่า

ซึ่งทำให้การคำนวณน้อยกว่า การเลือกใช้งานระหว่าง FIR และ IIR สามารถเลือกให้เหมาะสมกับการประยุกต์ใช้ การเลือกใช้งานระหว่าง FIR และ IIR สามารถสรุปได้คร่าว ๆ ดังนี้

- I. การประยุกต์ใช้งานทางด้านการส่งข้อมูล (Data transmission) เครื่องมือทางการแพทย์ (Biomedicine) การประมวลผลภาพ (Image processing) จำเป็นต้องใช้ตัวกรองความถี่ที่ให้ผลตอบสนองทางเฟสเป็นเชิงเส้น ดังนั้นตัวกรองความถี่แบบ FIR จึงเหมาะสมกว่าแบบ IIR
- II. สำหรับการใช้งานที่เวลาจริงซึ่งต้องมีการจำกัดจำนวนบิตนั้น การคำนวณสัมประสิทธิ์ของตัวกรองจะต้องทำการปัดเศษ (Round-off) ซึ่งจะทำให้เกิด round-off noise ขึ้น ซึ่ง round-off noise นี้จะเกิดขึ้นในตัวกรองแบบ FIR น้อยกว่าที่เกิดขึ้นในตัวกรองแบบ IIR
- III. ในกรณีที่มีความต้องการความชันในช่วงแถบความถี่ตัด (Cut off) สูงนั้นตัวกรองแบบ FIR ต้องใช้จำนวนอันดับสูงกว่า ทำให้มีสัมประสิทธิ์จำนวนมากกว่าและมีความยุ่งยากกว่าตัวกรองแบบ IIR เนื่องจากตัวกรองแบบ IIR มีผลตอบสนองทางแอมพลิจูดดีกว่าตัวกรองแบบ FIR
- IV. ตัวกรองความถี่แบบ IIR สามารถออกแบบได้โดยตรงจากตัวกรองที่เป็นแบบแอนะล็อก แต่ FIR ไม่สามารถทำได้
- V. ตัวกรองความถี่แบบ FIR จะสังเคราะห์ได้ค่อนข้างยาก ถ้าหากไม่ใช้คอมพิวเตอร์ (CAD) ช่วยในการออกแบบ

จากทั้งหมดที่กล่าวมาพอจะสรุปเป็นแนวทางสำหรับการใช้งานได้ดังนี้

- ในงานที่มีความต้องการความชันในช่วงแถบความถี่ตัดสูง ตัวกรองแบบ IIR จะใช้จำนวนอันดับที่น้อยกว่า ทำให้ตัวหน่วง และความยุ่งยากของตัวกรองน้อยกว่าตัวกรองแบบ FIR
- ตัวกรองแบบ FIR จะเหมาะสมกับงานที่ต้องการความแม่นยำทางเฟส เช่น เครื่องมือทางการแพทย์ หรือการประมวลผลภาพ

ในวิทยานิพนธ์นี้ได้นำเอาตัวกรองความถี่แบบ IIR notch filter มาใช้งาน เนื่องจากต้องการความชันในช่วงแถบความถี่ตัดสูง ดังนั้นเนื้อหาที่จะได้กล่าวต่อไป มีเพียงเฉพาะตัวกรองความถี่แบบ IIR เท่านั้น ส่วนตัวกรองความถี่แบบ FIR สามารถดูได้จากเอกสารอ้างอิง [17]

3.1 ตัวกรองความถี่แบบ IIR (Infinite Impulse Response filter)

ตัวกรองความถี่แบบ IIR เป็นหนึ่งในการประยุกต์ใช้ที่สำคัญของระบบการประมวลผลแบบไม่ต่อเนื่องทางเวลา (Discrete-time processing) มีข้อดีหลาย ๆ ข้อที่เหนือกว่าตัวกรองแบบ FIR โดยเฉพาะอย่างยิ่ง ด้านผลตอบสนองทางแอมพลิจูด เช่น เมื่อมีความต้องการให้ช่วงแถบเปลี่ยน (Transition band) ของช่วงแถบความถี่ผ่าน (Pass band) หรือแถบความถี่หยุด (Stop band) มีความชันมาก ๆ โดยที่ตัวกรองแบบ IIR จะใช้จำนวนอันดับน้อยกว่าตัวกรองแบบ FIR และยังมีความยุ่งยากในการคำนวณน้อยกว่าอีกด้วย

ตัวกรองความถี่แบบ IIR จะมีสมการของสัญญาณเอาต์พุตที่เป็นฟังก์ชันของสัญญาณอินพุตปัจจุบัน อินพุตในอดีต และสัญญาณเอาต์พุตในอดีต ซึ่งสามารถเขียนให้อยู่ในรูปของสมการเชิงผลต่าง (Difference equation) ได้ดังนี้

$$y(n) - \sum_{k=1}^N a_k y(n-k) = \sum_{k=0}^M b_k x(n-k) \quad (3.1)$$

เมื่อ $x(n)$ คือลำดับของสัญญาณอินพุต และ $y(n)$ คือลำดับของสัญญาณเอาต์พุต N, M คือ จำนวนตัวอย่าง (Samples) ทั้งหมดของสัญญาณเอาต์พุตและอินพุตที่ทราบค่า สมการที่ (3.1) สามารถนำไปใช้ในการคำนวณหาสัญญาณเอาต์พุตต่อ ๆ กันไปได้ รูปแบบของการนำเอาสัญญาณเอาต์พุตก่อนหน้ามาใช้สำหรับคำนวณหาสัญญาณเอาต์พุตต่อ ๆ กันไปจะเรียกว่า recursive โดยทั่วไป ตัวกรองความถี่แบบ IIR และตัวกรองแบบ recursive มักจะนำมาใช้ในความหมายเดียวกัน ทั้งนี้เพราะสมการที่ (3.1) สามารถนำไปใช้สร้างตัวกรองความถี่ทั้งสองแบบได้เหมือนกัน คำว่า IIR จะหมายถึงรูปแบบของผลตอบสนองอิมพัลส์ (Impulse response) ของตัวกรองความถี่ ในขณะที่คำว่า recursive หมายถึงตัวกรองความถี่นี้ถูกสร้างขึ้นมาอย่างไร ตัวกรองความถี่แบบ FIR ก็สามารถสร้างในรูปแบบ recursive ได้ ในขณะที่เดียวกันตัวกรองความถี่แบบ IIR ก็สามารถสร้างในรูปแบบ non-recursive ได้เหมือนกัน

เพื่อความสะดวกจะนิยามตัวกรองความถี่แบบรีเคอร์ซีฟด้วยฟังก์ชันถ่ายโอน หรือ ฟังก์ชันของระบบ (System function) โดยฟังก์ชันของระบบคือการแปลง z (Z-transform) ของผลตอบสนองอิมพัลส์ของตัวกรองความถี่ ซึ่งจะอยู่ในรูปเศษส่วน (Rational function) ในตัวแปร z^{-1} ตามระบบสมการที่ (3.1) จะมีฟังก์ชันของระบบเป็นไปตามสมการคือ

$$H(z) = \frac{\sum_{k=0}^M b_k z^{-k}}{1 + \sum_{k=1}^N a_k z^{-k}} \quad (3.2)$$

จากสมการที่ (3.2) ถ้ากำหนดให้เงื่อนไขเริ่มต้นเป็นศูนย์ จะสามารถเขียนใหม่ได้ คือ

$$H(z) = G \frac{\prod_{k=0}^M (1 - \beta_k z^{-1})}{\prod_{k=1}^N (1 - \alpha_k z^{-1})} \quad (3.3)$$

ราก (Root) ของพหุนามตัวเศษ β_k เรียกว่าซีโร่ (Zeros) ของตัวกรองความถี่ และรากของตัวส่วน α_k เรียกว่าโพล (Poles) และ G เป็นอัตราขยายซึ่งมีค่าคงที่ โดยทั่วไป จำนวนของซีโร่และโพลจะขึ้นอยู่กับความต้องการของผู้ใช้งานในการออกแบบ จำนวนอันดับ (Order) ของตัวกรองความถี่แบบ IIR จะหาได้จากจำนวนรากของโพลที่อยู่ในระนาบ z ที่มีค่าจำกัด (Finite z -plane)

ตัวกรองความถี่ที่มีความเป็นเชิงเส้นไม่แปรตามเวลา (Linear time-invariant, LTI) จะมีคุณสมบัติเป็น causal ถ้าผลตอบสนองอิมพัลส์มีค่าเท่ากับศูนย์เมื่อ $n < 0$ ตัวอย่างสัญญาณเอาต์พุตของตัวกรองความถี่ที่เป็น causal จะขึ้นกับตัวอย่างสัญญาณอินพุตปัจจุบันและในอดีตเท่านั้น ถ้าจำกัดสัญญาณอินพุตให้มีค่าเป็นศูนย์ $x(n) = 0$ สำหรับ $n < 0$ และค่าเริ่มต้นของ $y(-1) = y(-2) = \dots = y(-N) = 0$ จะทำให้ตัวกรองความถี่แบบรีเคอร์ซีฟตามสมการที่ (3.1) มีคุณสมบัติเป็น causal ความเป็น causal ของตัวกรองความถี่ มีความสำคัญมากในการประยุกต์ใช้งานที่เวลาจริง (Real time) เมื่อมีการ tick ของสัญญาณนาฬิกาจะได้ตัวอย่างอินพุต 1 ตัวอย่าง จากนั้นตัวกรองความถี่จะต้องสร้างตัวอย่างของเอาต์พุตออกมาด้วย

ข้อควรคำนึงอีกอันหนึ่งของตัวกรองความถี่แบบ IIR คือความเสถียร (Stable) ถ้าตัวกรองความถี่ที่ไม่เสถียร (Unstable) จะทำให้ลำดับของสัญญาณเอาต์พุตเพิ่มขึ้นอย่างไม่มีขอบเขต ถ้าสัญญาณอินพุตยังคงป้อนให้อยู่ ซึ่งเสถียรภาพของตัวกรองความถี่แบบ IIR จะขึ้นอยู่กับตำแหน่งโพล ของฟังก์ชันระบบในระนาบ z ตัวกรองความถี่แบบ IIR ที่มีคุณสมบัติเป็น causal LTI จะมีความเสถียร ถ้าค่าโพลเป็นไปตามเงื่อนไข $|\alpha_k| < 1$ ซึ่งหมายความว่าตำแหน่งโพลทุกตัวจะต้องอยู่ในวงกลมหนึ่งหน่วย (Unit circle) บนระนาบ z นั้นเอง

3.2 โครงสร้างของตัวกรองความถี่แบบ IIR (Basic structure for IIR systems)

โครงสร้างของตัวกรองแบบ IIR มีหลายรูปแบบ เช่น รูปแบบโดยตรง 1 และ 2 (Direct Forms I และ II), Transpose Form, Parallel Form, and Cascade Form ข้อสำคัญในการเลือกรูปแบบของโครงสร้างของตัวกรองนั้น พิจารณาจากความยุ่งยากในการคำนวณ เช่น จำนวนการคูณ (Multiplication) และการหน่วง (Delay) ซึ่งการคูณนอกจากจะมีราคาแพงเมื่อนำไปทำเป็นฮาร์ดแวร์ (Hardware) แล้ว ยังสิ้นเปลืองเวลา (Time-consuming) ในการทำงานของตัวกรอง เมื่อสามารถลดจำนวนของการคูณลงได้ ตัวกรองจะสามารถทำงานได้เร็วขึ้น ส่วนการหน่วงเวลาจะใช้หน่วยความจำ (Memory) ซึ่งเมื่อสามารถลดจำนวนการหน่วงเวลา จะสามารถลดหน่วยความจำที่ใช้ได้ ในหัวข้อนี้จะกล่าวถึงโครงสร้างของตัวกรองความถี่แบบ IIR แสดงในรูปของ signal flow graph [24] โดยจะกล่าวถึงเฉพาะ โครงสร้างรูปแบบโดยตรง (Direct form) และ โครงสร้างรูปแบบแลตทิซ (Lattice form)

3.2.1 ตัวกรองโครงสร้างรูปแบบโดยตรง

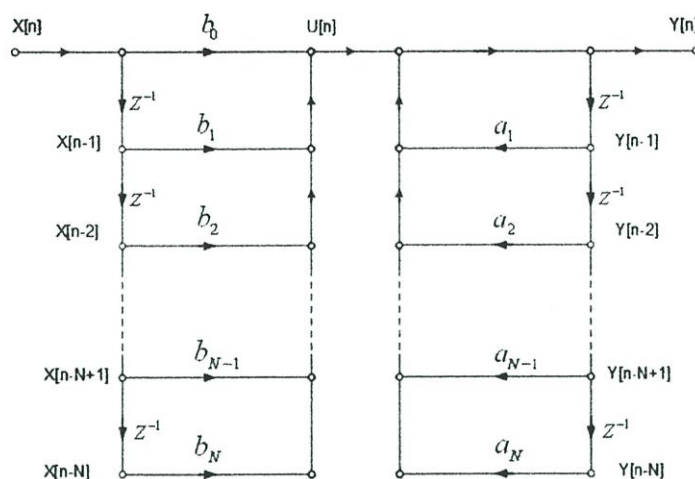
จากสมการที่ (3.1) และ (3.2) ประกอบไปด้วยการบวกจำนวน $M+1$ ครั้ง สำหรับพจน์อินพุต $x(n), x(n-1), \dots, x(n-M)$ และ N ครั้งสำหรับพจน์เอาต์พุต $y(n-1), y(n-2), \dots, y(n-N)$ และแต่ละพจน์จะถูกถ่วงน้ำหนักด้วยสัมประสิทธิ์ของตัวกรองความถี่ จากสมการที่ (3.1) สามารถเขียนใหม่ได้ว่า

$$u(n) = b_0x(n) + b_1x(n-1) + \dots + b_Mx(n-M) \quad (3.4)$$

$$y(n) = -a_1y(n-1) - a_2y(n-2) - \dots - a_Ny(n-N) + u(n) \quad (3.5)$$

จากสมการที่ (3.4) และ (3.5) จะเห็นได้ว่าตัวกรองความถี่แบบ IIR จะประกอบด้วยระบบย่อย (Subsystem) สองระบบต่อกันเป็นชั้น (Cascade) กล่าวคือ ชั้นที่หนึ่งเป็นชั้นของฟังก์ชันเลขในสมการที่ (3.2) และชั้นที่สองเป็นของฟังก์ชันส่วนของสมการที่ (3.2) เมื่อนำสองส่วนนี้มารวมกัน จะได้โครงสร้างรูปแบบโดยตรง 1 (Direct form I) ซึ่งสามารถเขียนเป็น signal flow graph สำหรับตัวกรองความถี่อันดับ N ตามรูปที่ 3.2

จะสังเกตเห็นว่าโครงสร้างแบบโดยตรง 1 จะมีจำนวนพจน์ทั้งหมด $M+N+1$ พจน์ คือ $x(n), x(n-1), \dots, x(n-M), y(n-1), y(n-2), \dots, y(n-N)$ แต่ละพจน์จะถูกคูณด้วยสัมประสิทธิ์ และนำมาบวกกัน ซึ่งจำนวนของการคูณสำหรับตัวอย่างเอาต์พุตแต่ละค่าจะเท่ากับจำนวนสัมประสิทธิ์ที่เป็น nontrivial เท่านั้น (การคูณด้วย 1, -1 และ 0 จะไม่นำมาคิดเป็นการคูณ)



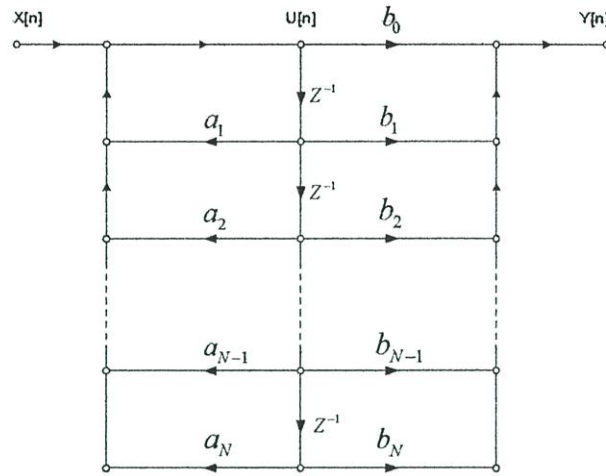
รูปที่ 3.2 Signal flow graph ของโครงสร้างรูปแบบโดยตรง 1 สำหรับระบบที่มี N ลำดับ

จากสมการที่ (3.2) ตัวเศษของฟังก์ชันระบบ ต้องการสัญญาณอินพุตเป็น $x(n)$ และให้สัญญาณเอาต์พุตเป็น $u(n)$ และตัวส่วนต้องการสัญญาณอินพุตเป็น $u(n)$ และให้สัญญาณเอาต์พุตเป็น $y(n)$ จากรูปที่ 3.2 ถ้าทำการพลิก (Reverse) แต่ละระบบย่อย จะทำให้ตัวอย่างข้อมูลที่ถูเก็บในระบบย่อยที่ 1 สามารถเก็บไว้ร่วมกับระบบย่อยที่ 2 ได้ ซึ่งโครงสร้างลักษณะนี้จะเรียกว่ารูปแบบโดยตรง 2 (Direct form II) หรือ canonic direct form ซึ่งมีข้อดีคือต้องการจำนวนตัวหน่วงเวลาน้อยกว่า โดยจะมีค่าสูงสุดเท่ากับ M , N และจะใช้เวลาคูณเท่ากับ $M + N + 1$ ครั้ง โครงสร้างรูปแบบโดยตรง 2 จะมีสมการผลต่าง คือ

$$u(n) = x(n) - a_1u(n-1) - a_2u(n-2) - \dots - a_Nu(n-N) \quad (3.6)$$

$$y(n) = b_0u(n) + b_1u(n-1) + \dots + b_Mu(n-M) \quad (3.7)$$

และ signal flow graph ของโครงสร้างแบบโดยตรง 2 แสดงดังรูปที่ 3.3

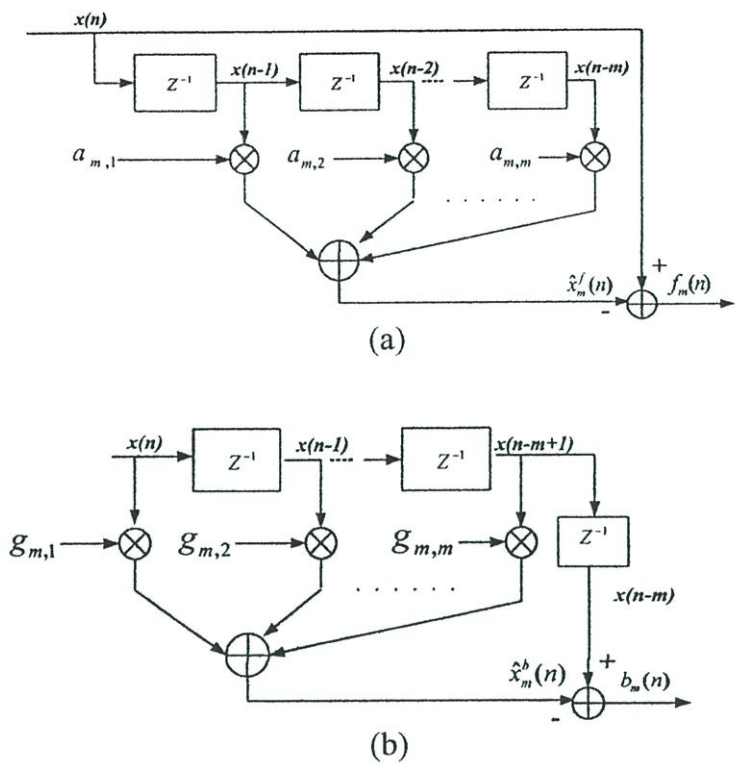


รูปที่ 3.3 Signal flow graph ของโครงสร้างรูปแบบโดยตรง 2 สำหรับระบบที่มี N อันดับ

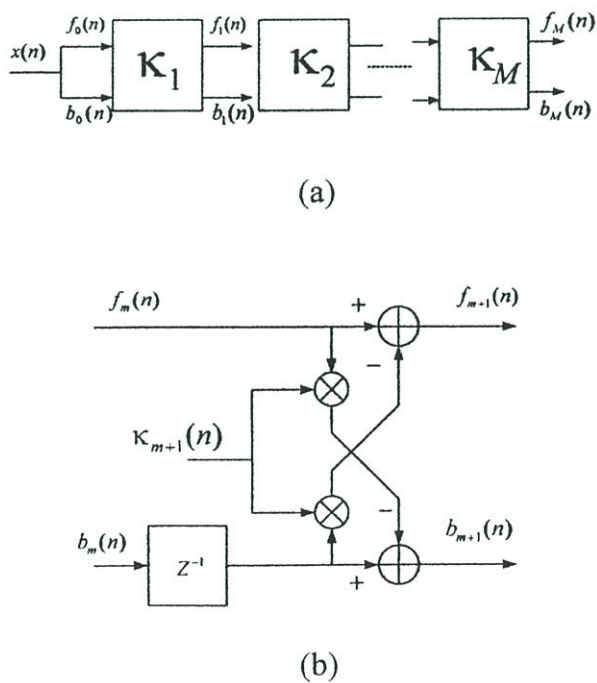
ข้อสังเกตจากโครงสร้างรูปแบบโดยตรง 1 ในรูปที่ 3.2 และโครงสร้างรูปแบบโดยตรง 2 ในรูปที่ 3.3 จะเห็นได้ว่าทั้งสองโครงสร้างต่างสอดคล้องกับสมการที่ (3.4) และ (3.5) เพียงแต่โครงสร้างรูปแบบโดยตรง 1 จะใช้จำนวน unit delay (z^{-1}) เท่ากับ $M+N$ ในขณะที่โครงสร้างรูปแบบโดยตรง 2 จะใช้จำนวนตัวหน่วงเวลา (z^{-1}) เพียง $\max\{M,N\}$ ซึ่งเป็นการประหยัดตัวหน่วงเวลาลงได้หนึ่งเท่าตัว

3.2.2 ตัวกรองโครงสร้างรูปแบบแลตทิซ

โครงสร้างรูปแบบแลตทิซ [22] เป็นโครงสร้างตัวกรองแบบคาดคะเนความคลาดเคลื่อน หรือ PEF (Prediction-error filter) ซึ่งประกอบด้วยการคูณสัญญาณด้วยสัมประสิทธิ์หลายครั้ง เพื่อให้กำลังงานความคลาดเคลื่อนเอาต์พุตของตัวกรองถูกทำให้น้อยที่สุด PEF มีอยู่สองรูปแบบแบ่งตามลำดับของอินพุต คือ forward PEF เป็นรูปแบบที่ออกแบบให้ลดค่ากำลังสองเฉลี่ยของค่าความคลาดเคลื่อนคาดคะเนเดินหน้า (Forward prediction-error) คือความแตกต่างของค่าที่ประมาณได้ในอนาคตและค่าที่แท้จริง ในขณะที่ backward PEF ได้ออกแบบให้ลดค่ากำลังสองเฉลี่ยของค่าความคลาดเคลื่อนคาดคะเนย้อนกลับ (Backward prediction-error) คือความแตกต่างของค่าที่ประมาณได้ในอดีต และค่าที่แท้จริง จำนวนอันดับ (Order) ของตัวกรองคือจำนวนชั้น (Stage) ของโครงสร้างแบบแลตทิซนั่นเอง



รูปที่ 3.4 โครงสร้างตัวกรองแบบคาดคะเนความคลาดเคลื่อน (a) forward PEF และ (b) backward PEF



รูปที่ 3.5 ลักษณะของตัวกรองโครงสร้างรูปแบบแลตทิซ (a) โครงสร้างของตัวกรองแลตทิซ (b) โครงสร้างของตัวกรองแลตทิซ m อันดับ

เมื่อให้ $x(n)$ เป็นสัญญาณอินพุตจะได้เอาต์พุตของรูปที่ 3.5 (b) ดังนี้

$$f_{m+1} = f_m(n) - \kappa_{m+1} b_m(n-1) \quad (3.8)$$

$$b_{m+1}(n) = b_m(n) - \kappa_{m+1} f_m(n) \quad (3.9)$$

เมื่อ $m = 0, 1, 2, \dots$ และสำหรับค่าเริ่มต้นอันดับที่ศูนย์ กำหนดให้ $f_0(n) = b_0(n) = x(n)$ และค่าสัมประสิทธิ์ κ_{m+1} กำหนดดังนี้

$$\kappa_{m+1} = \frac{E[f_m(n)b_m(n-1)]}{P_m} \quad (3.10)$$

$$P_{m+1} = (1 - \kappa_{m+1}^2) P_m \quad (3.11)$$

โครงสร้างแบบแลตทิซมีข้อดีที่น่าสนใจ เช่น เป็นโครงสร้างที่มีประสิทธิภาพสูง แต่ละชั้นของตัวกรองเป็นอิสระต่อกัน โดยที่ค่าความคลาดเคลื่อนคาดคะเนย้อนกลับของเอาต์พุตแต่ละชั้นนั้นตั้งฉากต่อกันเมื่อมีสัญญาณอินพุตแบบ WSS (Wide sense stationary) เมื่อต้องการเพิ่มอันดับของตัวกรอง สามารถเพิ่ม stage ของตัวกรองได้เลย โดยไม่มีผลต่อ stage ก่อนหน้าของตัวกรอง และทุกชั้นของตัวกรองมีโครงสร้างเหมือน เหมาะสำหรับการสร้างตัวกรองบน VLSI (Very large scale integration)

3.3 การทดสอบเสถียรภาพ (Stability test) [28]

เพื่อที่จะทำให้ฟังก์ชันถ่ายโอนของตัวกรองแบบ IIR ในสมการที่ (3.3) เป็น causal และมีความเสถียรแบบ BIBO (Bound In Bound Out) นั้น จะต้องทำให้โพลทุกตัวอยู่ในวงกลมหนึ่งหน่วย (Unit circle) เท่านั้น การวิเคราะห์เพื่อหาตำแหน่งโพลของฟังก์ชันถ่ายโอนที่อันดับสูงกว่าอันดับสองนั้น ทำได้ค่อนข้างยากและจะต้องใช้โปรแกรมคอมพิวเตอร์ช่วยในการหาราก แต่สำหรับฟังก์ชันถ่ายโอนอันดับสองมีวิธีการง่ายๆ ในการทดสอบเสถียรภาพ จากสมการที่ (3.2)

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

ให้

$$D(z) = 1 + a_1 z^{-1} + a_2 z^{-2} \quad (3.12)$$

โดยที่ $D(\bullet)$ คือตัวส่วนอันดับสองของสมการที่ (3.2) จะสามารถเขียนให้อยู่ในรูปโพลดังสมการที่ (3.12) จะได้ว่า

$$D(z) = (1 - \alpha_1 z^{-1})(1 - \alpha_2 z^{-2}) = 1 - (\alpha_1 + \alpha_2) z^{-1} + \alpha_1 \alpha_2 z^{-2} \quad (3.13)$$

เปรียบเทียบสมการที่ (3.12) กับสมการที่ (3.13) จะได้ว่า

$$a_1 = -(\alpha_1 + \alpha_2), \quad a_2 = \alpha_1 \alpha_2 \quad (3.14)$$

เพื่อให้ฟังก์ชันถ่ายโอนมีเสถียรภาพ ตำแหน่งโพลจะต้องอยู่ภายในวงกลมหนึ่งหน่วย นั่นคือ

$$|\alpha_1| < 1, \quad |\alpha_2| < 1$$

เนื่องจากสมการที่ (3.12) สัมประสิทธิ์ a_2 ถูกกำหนดเป็นผลคูณของโพล ดังนั้นจะได้ว่า

$$|a_2| < 1 \quad (3.15)$$

และจะสามารถกำหนดรากของพหุนาม $D(z)$ ดังนี้ คือ

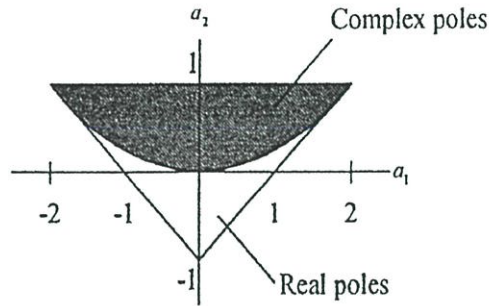
$$\alpha_1 = -\frac{a_1 + \sqrt{a_1^2 - 4a_2}}{2}, \quad \alpha_2 = -\frac{a_1 - \sqrt{a_1^2 - 4a_2}}{2} \quad (3.16)$$

และจากสมการที่ (3.16) จะได้ว่า

$$|a_1| < 1 + a_2 \quad (3.17)$$

ถ้าสัมประสิทธิ์ของตัวรองความถี่เป็นไปตามสมการที่ (3.15) และสมการที่ (3.17) จะสามารถวาดขอบเขตของค่าสัมประสิทธิ์ที่ยังคงทำให้ตัวรองความถี่มีความเสถียรเป็นรูปสามเหลี่ยมใน

ระนาบ (a_1, a_2) ซึ่งรู้จักกันทั่วไปในชื่อของสามเหลี่ยมเสถียรภาพ (Stability triangle) สำหรับ ฟังก์ชันถ่ายโอนดิจิทัลอันดับสอง แสดงดังรูปที่ 3.6



รูปที่ 3.6 สามเหลี่ยมเสถียรภาพสำหรับฟังก์ชันถ่ายโอนแบบดิจิทัลอันดับสอง

โพลที่เกิดจากฟังก์ชันถ่ายโอนอันดับสองนั้นจะมีสามชนิด คือ โพลสังยุคเชิงซ้อน (Complex conjugate poles) โพลค่าจริงที่ไม่ซ้ำกัน (Real and unequal) และ โพลค่าจริงและเท่ากัน (Real and equal) โพลที่เป็นสังยุคเชิงซ้อนมักจะเกิดขึ้นถ้า $a_1^2 < 4a_2$ ซึ่งกรณีนี้ โพลแต่ละค่าจะอยู่ที่ตำแหน่ง รัศมี r จากจุดกำเนิดของวงกลมหนึ่งหน่วย และที่มุม θ ซึ่งกำหนดได้ตามสมการดังนี้คือ

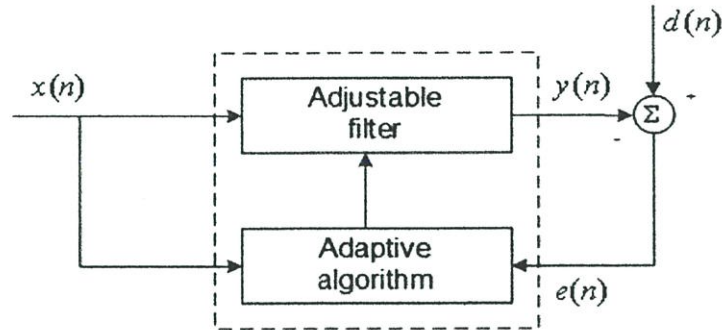
$$a_1 = r \angle \theta \quad (3.18)$$

เมื่อ

$$r = a_2^{1/2}, \theta = \cos^{-1} \left(\frac{-a_1}{2r} \right) \quad (3.19)$$

จากสมการที่ (3.18) และ (3.19) สามารถใช้เป็นสูตรสำหรับประมาณจำนวนบิตเมื่อนำไปใช้งานที่เวลาจริง ที่ทำให้ระบบมีเสถียรภาพได้ซึ่งเป็นหนึ่งในหลาย ๆ วิธีที่มีใช้กัน สำหรับวิธีการหาเงื่อนไขของสัมประสิทธิ์ที่ทำให้มีเสถียรภาพของฟังก์ชันถ่ายโอนอันดับสูง ๆ สามารถดูได้จากเอกสารอ้างอิง [28]

3.4 แบบจำลองพื้นฐานตัวกรองอะแดปทีฟ



รูปที่ 3.7 บล็อกไดอะแกรมแบบจำลองตัวกรองอะแดปทีฟ

ตัวกรองอะแดปทีฟคือส่วนที่อยู่ในกรอบจุดไข่ปลา ประกอบไปด้วยสองส่วน คือส่วนของตัวกรอง และ ส่วนของอัลเพคท ส่วนของตัวกรองคือ Adjustable filter สามารถที่จะใช้ตัวกรองโครงสร้างใด ๆ ก็ได้ ในการคำนวณหาสัญญาณเอาต์พุตตัวกรอง $y(n)$ ให้ได้ค่าที่ใกล้เคียงกับสัญญาณที่ต้องการ $d(n)$ มากที่สุด ส่วนอัลเพคทจะทำการปรับสัมประสิทธิ์ตัวกรองทีละน้อย เพื่อให้สัญญาณคลาดเคลื่อน $e(n)$ มีค่าน้อยที่สุด สัญญาณคลาดเคลื่อนหาได้จากความแตกต่างระหว่างสัญญาณที่ต้องการ $d(n)$ และ เอาต์พุตตัวกรอง $y(n)$ ตามสมการ

$$e(n) = d(n) - y(n) \quad (3.20)$$

การทำให้สัญญาณคลาดเคลื่อนมีค่าน้อยที่สุด จะเป็นตัวแสดงถึง ประสิทธิภาพในการกรองที่ดี โดยใช้วิธีการทั้งทางสถิติและ การหาค่าที่เหมาะสม

วิธีการทางสถิติ (Statistical performance) ที่นิยมใช้ทั่ว ๆ ไปคือ ค่าเฉลี่ยกำลังสองของสัญญาณคลาดเคลื่อน $\zeta(n) = E\{e^2(n)\}$ ซึ่ง $E\{\cdot\}$ คือ ตัวกระทำค่าคาดหวัง เมื่อส่วนอัลเพคทของตัวกรองอะแดปทีฟทำการปรับสัมประสิทธิ์ให้ค่าเฉลี่ยกำลังสองของสัญญาณคลาดเคลื่อนมีค่าน้อยที่สุดแล้ว ในทางอุดมคติก่าเฉลี่ยทางสถิติของความคลาดเคลื่อนจะเป็นศูนย์ เอาต์พุตของตัวกรองจะเท่ากับสัญญาณที่ต้องการ ตัวกรองอะแดปทีฟจึงจะทำงานได้ดีที่สุดในขณะที่อินพุต $x(n)$ และ สัญญาณที่ต้องการ $d(n)$ เป็นสัญญาณแบบคงที่ (วิธีการนี้เรียกว่า ตัวกรองวินเนอร์) ในทางปฏิบัติจะใช้อัลกอริทึมแบบ LMS ข้อมูลต่าง ๆ ที่ใช้ในการปรับสัมประสิทธิ์จะเป็นตัวกำหนดช่วงเวลาของการปรับตัว (Convergence time) ในการหาค่าตอบสุดท้าย และความแตกต่างระหว่างค่าตอบสุดท้ายกับค่าตอบที่เหมาะสม (Final misadjustment)

ส่วนวิธีการแบบสุ่ม (Deterministic performance) คือ การให้น้ำหนักผลรวมค่ายกกำลังสองของตัวอย่างสัญญาณความคลาดเคลื่อนก่อนหน้า $\zeta(n) = \sum_{k=1}^n \lambda^{n-k} e^2(k)$ เมื่อ λ เป็นค่าคงที่มีค่าเข้าใกล้ 1 แต่น้อยกว่า 1 และ $k=1,2,\dots,n$ การเลือกค่าตัวอย่างสัญญาณความคลาดเคลื่อนค่าใหม่ที่ใส่เข้าไปมีความสำคัญอย่างยิ่ง ค่าตัวอย่างเดิมจะค่อย ๆ หายไปที่ละนิด ๆ จึงเป็นเหตุผลที่จะเรียกว่า forgetting factor การทำให้ค่า $\zeta(n)$ มีค่าน้อยที่สุดทำให้ได้กลุ่มของสัมประสิทธิ์ตัวกรองที่ดีเป็นกลุ่มข้อมูลที่จะทำให้เอาต์พุตตัวกรอง $y(n)$ มีค่าใกล้เคียงกับสัญญาณที่ต้องการ $d(n)$ ตามวิธีการแบบ Least square อย่างไรก็ตามถ้ากลุ่มข้อมูลเป็นไปตามคุณสมบัติทางสถิติที่ถูกต้องและข้อมูลมีความยาวมากพอแล้ว สัมประสิทธิ์ตัวกรองที่ได้จากการสุ่มก็สามารถหาคำตอบตามวิธีของวินเนอร์ได้ วิธีการสุ่มนี้เป็นพื้นฐานของอัลกอริทึมแบบ RLS(Recursive Least Squares) ซึ่งจะหาคำตอบได้รวดเร็วและมีคุณสมบัติในการติดตามสัญญาณอินพุตได้ดี

อัลกอริทึมที่น่าสนใจ

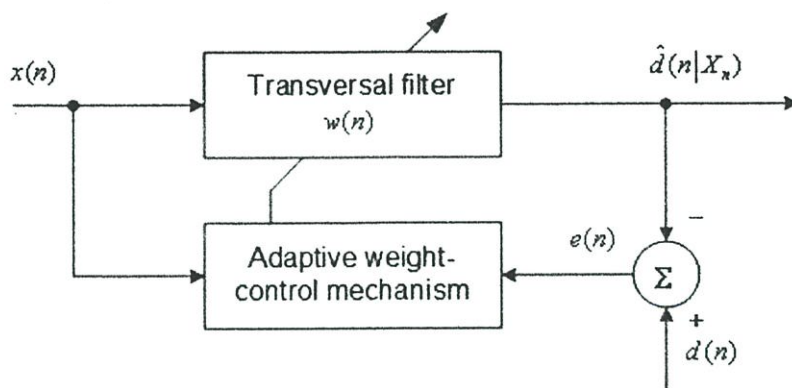
อัลกอริทึมแบบ LMS เป็นอัลกอริทึมที่สำคัญจัดอยู่ในกลุ่มของอัลกอริทึมแบบ stochastic gradient ซึ่งจะแบ่งแยก อัลกอริทึมแบบ LMS ออกจากวิธีของ steepest descent ที่ใช้การคำนวณค่าความลาดเอียง (Gradient) ป้อนกลับของตัวกรองวินเนอร์ที่มีอินพุตแบบ stochastic คุณสมบัติที่น่าสนใจ คือ ความง่ายในการคำนวณ ไม่ต้องหาฟังก์ชันความสัมพันธ์โดยตรงใด ๆ หรือการหาเมตริกซ์ผกผัน ดังนั้น อัลกอริทึมนี้จึงไม่เหมือนกับวิธีการกรองอะแดปทีฟแบบลิเนียร์อื่น ๆ ที่เป็นมาตรฐานแล้ว

3.4.1 อัลกอริทึมแบบ LMS (Least-mean-square algorithm)

อัลกอริทึมแบบ LMS เป็นวิธีการกรองอะแดปทีฟแบบลิเนียร์วิธีหนึ่ง ซึ่งประกอบด้วยกระบวนการพื้นฐานอยู่สองส่วน คือ

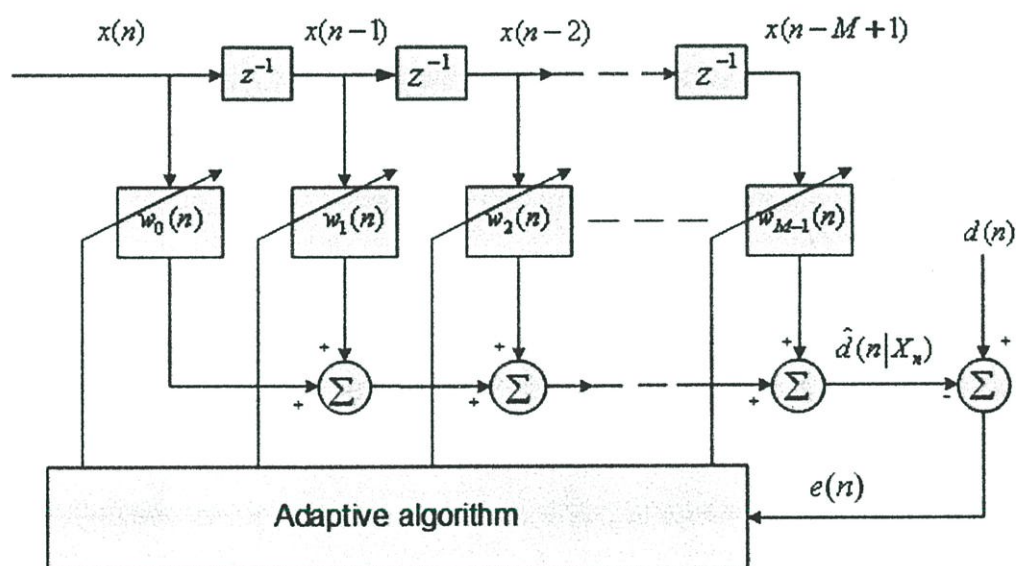
1. กระบวนการกรอง (Filtering process) มีขั้นตอนดังนี้
 - i) คำนวณเอาต์พุตของตัวกรองแบบลิเนียร์ที่ตอบสนองต่อสัญญาณอินพุต และ
 - ii) สร้างค่าความคลาดเคลื่อนที่ประมาณได้จากการเปรียบเทียบเอาต์พุตของตัวกรองกับ ผลตอบสนองที่ต้องการ
2. กระบวนการปรับตัว (Adaptive process) ซึ่งจะปรับค่าพารามิเตอร์ต่าง ๆ ของตัวกรองโดยอัตโนมัติตามค่าความคลาดเคลื่อนที่ประมาณได้

การทำงานร่วมกันของกระบวนการทั้งสองจะสร้างลูปป้อนกลับ (Feedback loop) ดังรูปที่ 3.8



รูปที่ 3.8 บล็อกไดอะแกรมของ LMS อัลกอริทึม

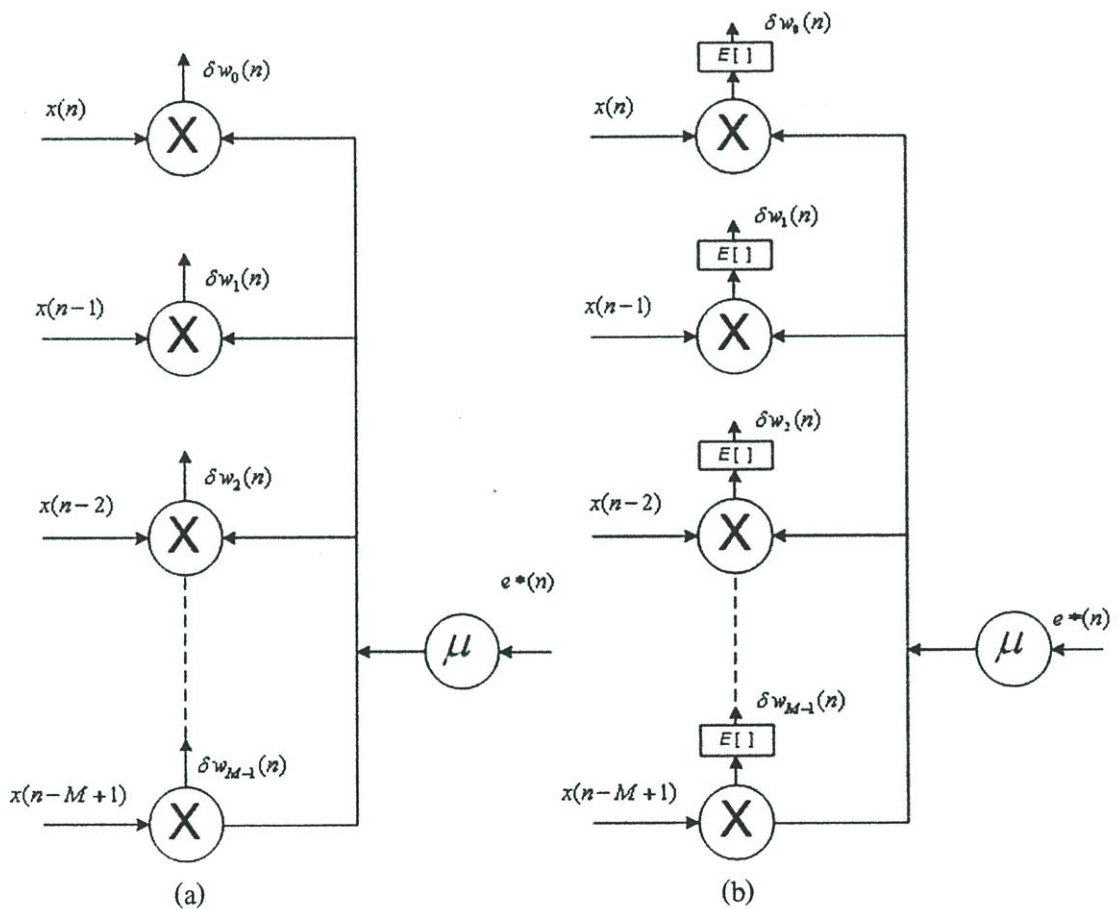
จากรูปที่ 3.8 จะเห็นว่า อัลกอริทึมแบบ LMS สร้างมาจาก ตัวกรองแบบทรานสเวอร์ซอล ส่วนประกอบต่าง ๆ ที่ทำหน้าที่ในกระบวนการกรอง



รูปที่ 3.9 วงจรควบคุมน้ำหนักถ่วงอะแดปทีฟ (Adaptive weights control) ของตัวกรองแบบ ทรานสเวอร์ซอล

รูปที่ 3.9 ตัวกรองแบบทรานสเวอร์ซอล ที่ใช้แสดงกระบวนการควบคุมการปรับน้ำหนักถ่วงแบบแยกส่วน ส่วนของอินพุต $x(n), x(n-1), \dots, x(n-M+1)$ จะสร้างรูปแบบของเวกเตอร์อินพุต $x(n)$ ที่มีขนาด $M-1$ ซึ่ง $M-1$ เป็นจำนวนของตัวหน่วง (Delay element) ที่วางอยู่ในที่ว่างหลายมิติ X_n สอดคล้องกับน้ำหนักถ่วง $w_0(n), w_1(n), \dots, w_{M-1}(n)$ ที่สร้างรูปแบบส่วนประกอบของเวกเตอร์น้ำหนักถ่วง $w(n)$ ที่มีขนาด $M-1$ ค่าเวกเตอร์ที่คำนวณได้จาก

อัลกอริทึมแบบ LMS จะเป็นค่าโดยประมาณที่อาจจะใกล้เคียงกับวิธีหาคำตอบของวินเนอร์ w_0 (ความหมายอย่างกว้าง ๆ ในสถานะเวกเตอร์แบบคงที่) จากการทำซ้ำจนถึงอนันต์ ตลอดช่วงระยะเวลาของกระบวนการกรอง ผลตอบสนองที่ต้องการ $d(n)$ จะถูกส่งออกมาตามการประมวลผลของเวกเตอร์อินพุตแต่ละส่วน $x(n)$ เมื่อมีการป้อนอินพุต ตัวกรองทรานสเวอร์ซอล จะสร้างเอาต์พุต $\hat{d}(n|X_n)$ จากการประมาณค่าของผลตอบสนองที่ต้องการ $d(n)$ ดังนั้นอาจจะนิยามความคลาดเคลื่อนจากการประมาณ $e(n)$ ว่าเป็น ความแตกต่างระหว่างผลตอบสนองที่ต้องการและเอาต์พุตจริงของตัวกรอง ดังที่แสดงในส่วนเอาต์พุตของรูปที่ 3.9 ค่าความคลาดเคลื่อนที่ประมาณได้ $e(n)$ และ เวกเตอร์อินพุต $x(n)$ ที่ป้อนให้กับวงจรควบคุมและส่วนของการป้อนกลับไปยังน้ำหนักถ่วงเป็นการควบคุมแบบลูปปิด



รูปที่ 3.10 วิธีการควบคุมการปรับน้ำหนัก (a) LMS algorithm (b) Steepest descent method

รูปที่ 3.10 แสดงรายละเอียดวิธีการควบคุมการปรับน้ำหนักถ่วงในรูปแบบขนาดของผลคูณเวกเตอร์สองเวกเตอร์ระหว่างค่าความคลาดเคลื่อนจากการประมาณ $e(n)$ และอินพุตแต่ละส่วน $x(n-k)$ จะถูกคำนวณ เมื่อ $k = 0, 1, 2, \dots, M-1$. ค่า $\delta w_k(n)$ ที่ได้รับการแก้ไขให้ถูกต้อง

แล้วจะถูกป้อนให้กับน้ำหนักถ่วง $w_k(n)$ ทุกการทำซ้ำ $n+1$ ตัวประกอบในการคูณ μ (Scaling factor) ที่ใช้ในการคำนวณนี้ กำหนดให้เป็นค่าบวก จะเรียกว่า ค่าก้าวกระโดด (Step-size parameter)

$$\mu(n+1) = k_0(n) + \mu(n) \cdot x(n) \cdot e(n)$$

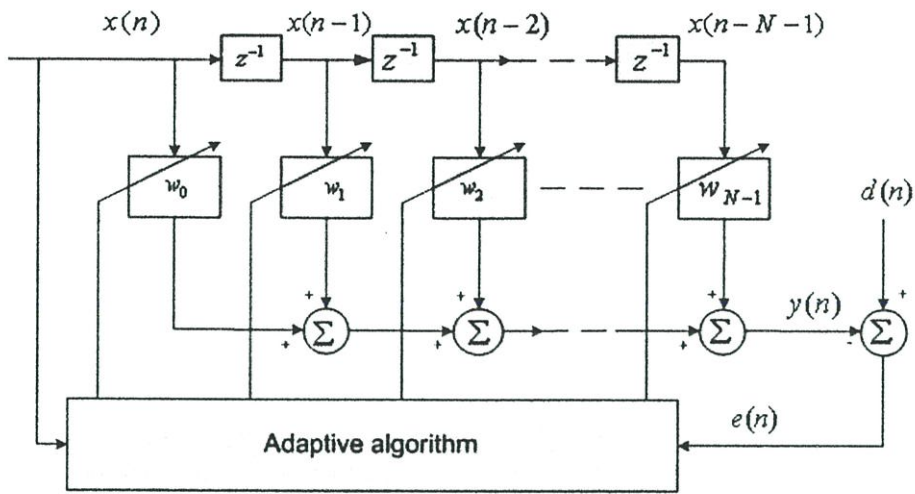
$$\begin{bmatrix} \text{เวกเตอร์} \\ \text{สัมประสิทธิ์ของวงจ} \\ \text{กรองที่ปรับค่าแล้ว} \end{bmatrix} = \begin{bmatrix} \text{เวกเตอร์} \\ \text{สัมประสิทธิ์ของ} \\ \text{วงจกรองเดิม} \end{bmatrix} + \begin{bmatrix} \text{ค่าช่วงก้าว} \\ \text{สำหรับ} \\ \text{การประมาณค่า} \end{bmatrix} \begin{bmatrix} \text{เวกเตอร์} \\ \text{สัญญาณอินพุต} \\ \text{ของวงจกรอง} \end{bmatrix} \begin{bmatrix} \text{ค่าผิดพลาด} \\ \text{จากการ} \\ \text{ประมาณ} \end{bmatrix}$$

เปรียบเทียบรูปที่ 3.10 (a) และ (b) วิธีการควบคุมของรูปที่ 3.10(a) คือ อัลกอริทึมแบบ LMS และ รูปที่ 3.10 (b) คือ วิธีของ steepest descent จะเห็นว่า อัลกอริทึมแบบ LMS จะได้ค่าประมาณของส่วนประกอบที่ k ของเวกเตอร์เกรเดียนต์ จากการคูณของ $x(n-k)e^*(k)$ ที่เป็นคุณลักษณะของวิธี steepest descent หรือ กล่าวได้ว่า ตัวกระทำความคาดหวัง (Expectation operator) ถูกย้ายออกไปทั้งหมดแล้วดังในรูปที่ 3.10 (a) ดังนั้นในการคำนวณการป้อนกลับในแต่ละน้ำหนักถ่วงในวิธี LMS จึงมีสัญญาณรบกวนแบบเกรเดียนต์เสมอ

3.4.2 วิธี Steepest Descent (Method of steepest descent)

ทฤษฎีของ steepest descent เป็นวิธีการปรับตัวบนพื้นฐานของเกรเดียนต์ (Gradient-based adaptation) ซึ่งจะทำการติดตามสัญญาณเมื่อเกิดการเปลี่ยนแปลงทางเวลาได้โดยไม่ต้องใช้สมการที่ยุ่งยาก วิธี steepest descent ใช้การทำซ้ำ หรือวิธีการปรับตัวเองในการเปลี่ยนแปลงค่าสัมประสิทธิ์ของตัวกรองที่เล็กทีละน้อย ในทิศทางที่ลดค่าของฟังก์ชันจุดประสงค์ที่มากที่สุด โดยการปรับค่าน้ำหนักถ่วงให้ cost function $J(w)$ เป็นฟังก์ชันแบบ continuously differentiable ของค่าถ่วงน้ำหนักที่ไม่ทราบค่า w ต้องการหาค่าที่เหมาะสม w_0 ตามเงื่อนไข

$$J(w_0) \leq J(w) \quad (3.21)$$



รูปที่ 3.11 วงจรกรองแบบทรานสเวอร์ส

จากรูปที่ 3.11 อินพุตของวงจรกรอง คือ $x(n)$ ผลตอบสนองที่ต้องการ คือ $d(n)$ ซึ่งสมมติเป็นลำดับค่าจริง ค่าสัมประสิทธิ์วงจรกรอง w_0, w_1, \dots, w_{N-1} สมมติให้เป็นค่าจริง ซึ่งสามารถเขียนให้อยู่ในรูปของเวกเตอร์ได้ตามลำดับดังนี้

$$\mathbf{w} = [w_0 \quad w_1 \quad \dots \quad w_{N-1}]^T \quad (3.22)$$

และ

$$\mathbf{x}(n) = [x(n) \quad x(n-1) \quad \dots \quad x(n-N+1)]^T \quad (3.23)$$

จะได้เอาต์พุตของวงจรกรองเป็น

$$y(n) = \mathbf{w}^T \mathbf{x}(n) = \mathbf{x}^T(n) \mathbf{w} \quad (3.24)$$

สัญญาณผิดพลาด $e(n)$ คือผลต่างระหว่างคำตอบที่ต้องการ $d(n)$ และเอาต์พุตของวงจรกรอง $y(n)$

$$e(n) = d(n) - \mathbf{w}^T \mathbf{x}(n) = d(n) - \mathbf{x}^T(n) \mathbf{w} \quad (3.25)$$

โดยมีฟังก์ชันจุดประสงค์เป็น

$$\xi = E|e^2(n)| \quad (3.26)$$

จากสมการที่ (3.25) เมื่อยกกำลังสองจะได้

$$e^2(n) = d^2(n) - 2\mathbf{w}^T \mathbf{x}(n)d(n) - \mathbf{w}^T \mathbf{x}(n)\mathbf{x}^T(n)\mathbf{w} \quad (3.27)$$

หาค่าคาดหวังของสมการที่ (3.27) ได้ดังนี้

$$\xi = E|e^2(n)| = E|d^2(n)| - 2\mathbf{w}^T E[\mathbf{x}(n)d(n)] - \mathbf{w}^T E[\mathbf{x}(n)\mathbf{x}^T(n)]\mathbf{w} \quad (3.28)$$

นิยามให้เวกเตอร์ \mathbf{p} คือ สหสัมพันธ์ข้ามระหว่างผลตอบสนองที่ต้องการ $d(n)$ ซึ่งเป็นค่าสเกลลาร์ กับเวกเตอร์อินพุตของวงจรกรอง $\mathbf{x}(n)$

$$\mathbf{p} = E[\mathbf{x}(n)d(n)] = [p_0 \ p_1 \ \dots \ p_{N-1}]^T \quad (3.29)$$

และนิยามให้เมตริกซ์อัตโนมัติของอินพุตเป็น

$$\mathbf{R} = E[\mathbf{x}(n)\mathbf{x}^T(n)] = \begin{bmatrix} r_{0,0} & r_{0,1} & \dots & r_{0,N-1} \\ r_{1,0} & r_{1,1} & \dots & r_{1,N-1} \\ \vdots & \vdots & \vdots & \vdots \\ r_{N-1,0} & r_{N-1,1} & \dots & r_{N-1,N-1} \end{bmatrix} \quad (3.30)$$

จากค่า \mathbf{p} และ \mathbf{R} ที่นิยามไว้นี้ สามารถเขียนสมการของค่ากำลังสองเฉลี่ยในสมการที่ (3.28) ได้ดังนี้

$$\xi = E|e^2(n)| = E|d^2(n)| - 2\mathbf{w}^T \mathbf{p} + \mathbf{w}^T \mathbf{R} \mathbf{w} \quad (3.31)$$

จากสมการนี้ทำให้ได้ฟังก์ชันจุดประสงค์ ซึ่งมีรูปร่างเป็นพื้นผิวพาราโบล่าหงาย การปรับค่าสัมประสิทธิ์ของตัวกรองเพื่อให้ได้ค่าความผิดพลาดต่ำที่สุด ก็คือ การวิ่งวนไปตามพื้นผิวนี้นจนถึงจุดที่ต่ำที่สุด ซึ่งสามารถทำได้โดยการหาค่าเกรเดียนต์ ∇ ของฟังก์ชันจุดประสงค์นี้

$$\nabla \xi = \left[\frac{\partial E[e^2(n)]}{\partial w_0} \quad \frac{\partial E[e^2(n)]}{\partial w_1} \quad \dots \quad \frac{\partial E[e^2(n)]}{\partial w_n} \right] = -2\mathbf{p} + 2\mathbf{R}\mathbf{w} \quad (3.32)$$

เมื่อให้ $\nabla \xi = 0$ จะได้เวกเตอร์ของค่าสัมประสิทธิ์ของตัวกรองที่เหมาะสม w_0 ซึ่งเรียกว่า “Wiener weight vector”

$$w_0 = \mathbf{R}^{-1}\mathbf{p} \quad (3.33)$$

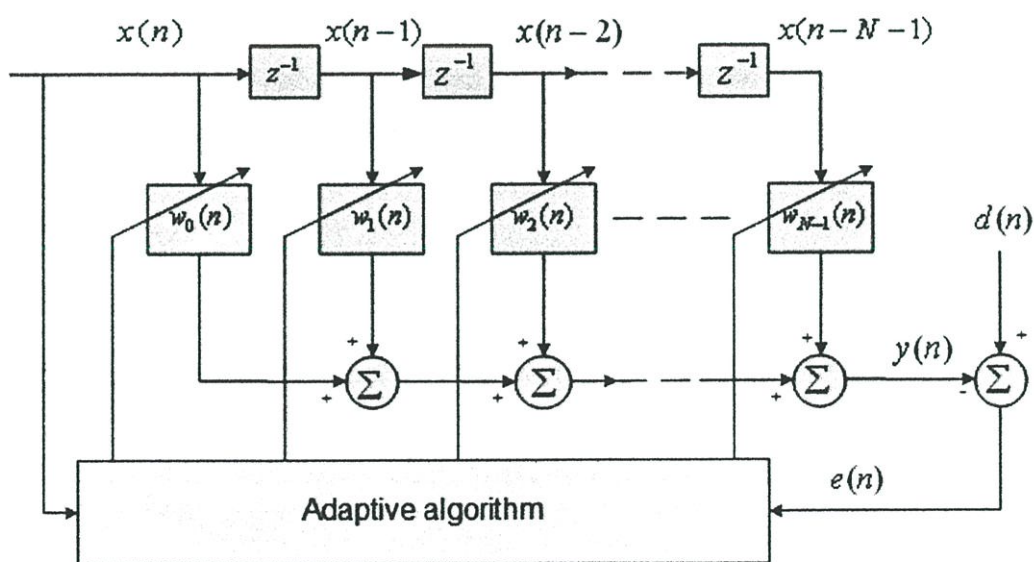
ตามวิธี steepest descent สามารถหาค่าเวกเตอร์สัมประสิทธิ์ของตัวกรองใหม่ได้ดังนี้

$$w(n+1) = w(n) - \mu \nabla_n \xi \quad (3.34)$$

โดย $w(n)$ เป็นเวกเตอร์สัมประสิทธิ์ของตัวกรอง ที่การทำซ้ำครั้งที่ n

μ เป็นค่าก้าวกระโดด

$\nabla_n \xi$ เป็นเวกเตอร์ของค่าเกรเดียนต์ของ $\nabla \xi$ ที่ $w = w(n)$



รูปที่ 3.12 ตัวกรองแบบปรับตัวได้แบบทรานเวอร์ส N แท็ป

จากรูปที่ 3.12 แสดงตัวกรองแบบปรับตัวได้แบบทรานเวอร์ส N แท็ป อินพุตของตัวกรอง คือ $x(n)$ ผลตอบที่ต้องการคือ $d(n)$ และจะได้เอาต์พุตของตัวกรองเป็น

$$y(n) = \sum_{i=0}^{N-1} w_i(n)x(n-i) \quad (3.35)$$

สัญญาณผิดพลาด $e(n)$ คือ ผลต่างระหว่างค่าตอบที่ต้องการ $d(n)$ และเอาต์พุตของตัวกรอง $y(n)$

$$e(n) = d(n) - y(n) \quad (3.36)$$

เมื่อสัมประสิทธิ์ของตัวกรอง หรือค่าสัมประสิทธิ์ของตัวกรอง $w_0(n), w_1(n), \dots, w_{N-1}(n)$ สามารถปรับค่าได้ เวกเตอร์สัมประสิทธิ์ของวงจรรอง $w(n)$ คือ

$$w(n) = [w_0(n) \ w_1(n) \ \dots \ w_{N-1}(n)]^T \quad (3.37)$$

อะแดปทีฟอัลกอริทึมโดยทั่วไปมีจุดประสงค์เพื่อปรับค่าสัมประสิทธิ์ของตัวกรองให้ค่าผิดพลาดกำลังสองเฉลี่ยมีค่าต่ำที่สุด เมื่อให้อินพุตของตัวกรอง $x(n)$ และผลตอบสนองที่ต้องการ $d(n)$ เป็นข้อมูลทางสถิติแบบคงที่ (Statistically stationary) คือมีค่าไม่ขึ้นกับเวลาและให้ค่าสัมประสิทธิ์ของตัวกรองเป็นค่าคงที่ใด ๆ

อัลกอริทึมแบบ LMS เป็นวิธีการในทางปฏิบัติในการหาค่าประมาณใกล้เคียงของผลเฉลยของสมการดังกล่าวในเวลาจริง (Real time) โดยไม่ต้องคำนวณฟังก์ชันสหสัมพันธ์ (Correlation function) และไม่ต้องหาเมตริกซ์ผกผัน สามารถปรับค่าเวกเตอร์สัมประสิทธิ์ของตัวกรองได้ดังนี้

$$w(n+1) = w(n) - \mu \nabla e^2(n) \quad (3.38)$$

โดย μ คือค่าก้าวกระโดด (Step size) และ ∇ เป็นการกระทำของเกรเดียนต์นิยามโดย

$$\nabla = \left[\frac{\partial}{\partial w_0} \quad \frac{\partial}{\partial w_1} \quad \dots \quad \frac{\partial}{\partial w_{N-1}} \right]^T \quad (3.39)$$

ที่ตำแหน่ง i ของเวกเตอร์เกรเดียนต์ $\nabla e^2(n)$

$$\frac{\partial e^2(n)}{\partial w_i} = 2e(n) \frac{\partial e(n)}{\partial w_i} \quad (3.40)$$

แทนสมการที่ (3.35) ในสมการที่ (3.40) และเนื่องจาก $d(n)$ ไม่ได้ขึ้นกับ w_i จะได้

$$\frac{\partial e^2(n)}{\partial w_i} = -2e(n) \frac{\partial y(n)}{\partial w_i} \quad (3.41)$$

แทน $y(n)$ จากสมการที่ (3.35) จะได้

$$\frac{\partial e^2(n)}{\partial w_i} = -2e(n)x(n-i) \quad (3.42)$$

จากสมการที่ (3.39) และ (3.42) จะได้

$$\nabla e^2(n) = -2e(n)x(n) \quad (3.43)$$

เมื่อ $x(n) = [x(n)x(n-1), \dots, x(n-N+1)]^T$ แทนสมการที่ (3.43) ในสมการที่ (3.38)

$$w(n+1) = w(n) + 2\mu e(n)x(n) \quad (3.44)$$

ค่าเกรเดียนต์เวกเตอร์ $\nabla \xi$ ในวิธีของ steepest descent สามารถประมาณได้ด้วย $e(n)x(n)$ หรือกล่าวได้ว่าตัวกระทำความคาดหวัง (Expectation operator) ได้ถูกนำออกไปจากสมการในวิธี steepest descent ซึ่งการประมาณ ค่าเกรเดียนต์เวกเตอร์โดยการละตัวกระทำความคาดหวังนี้ ทำให้เกิดผลตามมา คือ การคำนวณค่าสัมประสิทธิ์ของตัวกรองแต่ละครั้งจะได้รับผลกระทบจากสัญญาณรบกวนแบบเกรเดียนต์ (Gradient noise) ดังนั้นในการคำนวณโดยอัลกอริทึมแบบ LMS จึงเป็นการเคลื่อนไหวแบบสุ่มรอบ ๆ ค่าต่ำสุดของพื้นผิวความผิดพลาด (Error surface) หรือลู่อู่เข้าสู่ค่าเฉลี่ยแทนที่จะลู่อู่เข้าสู่จุดต่ำสุดอย่างแท้จริง

เนื่องจากอัลกอริทึมแบบ LMS ใช้กระบวนการป้อนกลับในการทำงาน ดังนั้นจึงต้องคำนึงถึงเสถียรภาพของระบบด้วย ซึ่งเกณฑ์ที่จะทำให้เกิดเสถียรภาพนี้คือ

$$\xi(n) \rightarrow \xi(\infty) \text{ เมื่อ } n \rightarrow \infty \quad (3.45)$$

เมื่อ $\xi(n)$ คือ ค่าความผิดพลาดกำลังสองเฉลี่ยของอัลกอริทึมแบบ LMS ที่เวลา n ใด ๆ และ $\xi(\infty)$ คือ ค่าความผิดพลาดกำลังสองเฉลี่ยที่สถานะ steady state ของอัลกอริทึมแบบ LMS

เป็นค่าคงที่ ค่าเฉลี่ยกำลังสองจะเข้าสู่สถานะ steady state เมื่อ $n \rightarrow \infty$ สำหรับอัลกอริทึมแบบ LMS นั้น ค่าก้าวกระโดด μ ที่เหมาะสม จะขึ้นไปตามเงื่อนไขความสัมพัทธ์โครงสร้างโอเกนเมตริกซ์ของสหสัมพันธ์อินพุต ซึ่งจะทำให้ค่าเฉลี่ยกำลังสองเข้าสู่ตามเกณฑ์ดังกล่าวได้

ความแตกต่างระหว่างค่าความผิดพลาดกำลังสองเฉลี่ยที่สถานะ steady state $\xi(\infty)$ และค่าความผิดพลาดกำลังสองเฉลี่ยต่ำที่สุด ξ_{\min} ที่ได้จากผลเฉลี่ยของ Wiener เรียกว่า ค่าความคลาดเคลื่อนกำลังสองเฉลี่ยส่วนเกิน (excess mean-squared error) หรือ $\xi_{\text{ex}}(\infty)$ ความแตกต่างนี้แสดงถึงความผิดพลาดที่เกิดขึ้นจากกระบวนการปรับสัมประสิทธิ์ของตัวกรองที่ใช้อัลกอริทึมแบบ LMS แทนที่จะใช้วิธีการแบบ steepest descent

Misadjustment คือ อัตราส่วนระหว่าง $\xi(\infty)$ กับ ξ_{\min} เป็นค่าที่แสดงว่าผลตอบสนองที่สถานะ steady state ของอัลกอริทึมแบบ LMS อยู่ห่างจากผลเฉลี่ยของตัวกรองวินเนอร์มากน้อยแค่ไหน และสามารถกำหนดค่าความผิดพลาดนี้ให้มีค่ามากน้อยตามต้องการได้ โดยขึ้นอยู่กับค่าการปรับค่าก้าวกระโดดที่ใช้ในอัลกอริทึมแบบ LMS เมื่อค่าก้าวกระโดดมีค่าน้อยๆ กระบวนการปรับตัวจะดำเนินไปค่อนข้างช้า ส่งผลให้สัญญาณรบกวนแบบเกรเดียนต์ของสัมประสิทธิ์ตัวกรองถูกกรองออกไปมาก ค่าผิดพลาดในการปรับตัวก็จะลดลง แต่ถ้าใช้ค่าก้าวกระโดดสูงๆ กระบวนการนี้จะเร็วขึ้น ค่าผิดพลาดในการปรับตัวก็จะมากขึ้นตามไปด้วย ดังนั้นการใช้อัลกอริทึมแบบ LMS จึงต้องให้ความสนใจกับค่าก้าวกระโดดเป็นอย่างมาก เนื่องจากค่าก้าวกระโดดที่ใช้ในอัลกอริทึมแบบ LMS เป็นค่าคงที่ตลอดกระบวนการปรับตัวของตัวกรอง ด้วยเหตุนี้จึงมีผู้พัฒนาวิธีการใหม่ ๆ ให้ตัวกรองมีการปรับค่าก้าวกระโดดไปพร้อม ๆ กับกระบวนการกรอง เพื่อให้การหาค่าตอบของตัวกรองใช้เวลาเข้าสู่ที่รวดเร็วยิ่งขึ้น และมีความถูกต้องมากขึ้น

3.4.3 อัลกอริทึมแบบ MNG (Memoryless Nonlinear Gradient)

สำหรับอะแดปทีฟอัลกอริทึมที่น่าสนใจ ซึ่งเป็นอัลกอริทึมแบบ Memoryless Nonlinear Gradient (MNG) [25] ซึ่งเป็นอัลกอริทึมที่ปรับปรุงมาจากอัลกอริทึมแบบ Plain Gradient (PG) โดยปรับปรุงในส่วนของสัญญาณเกรเดียนต์ให้เป็นฟังก์ชันของสัญญาณเกรเดียนต์ที่เหมาะสมที่สุด ทำให้ค่าความแปรปรวนมีค่าลดน้อยลง สมการสำหรับปรับค่าสัมประสิทธิ์ของตัวกรองมีดังนี้คือ

$$a(n+1) = a(n) - \mu \cdot y(n) \cdot \frac{g(n)}{1 + \mu \cdot g^2(n)} \quad (3.46)$$

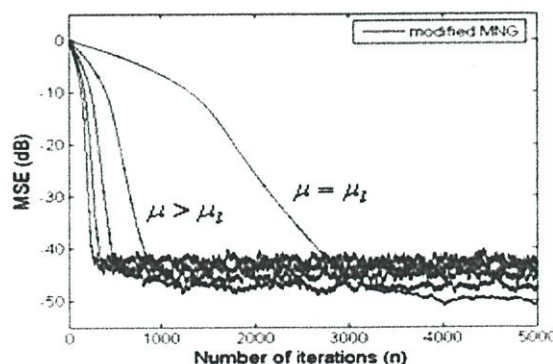
$$a(n+1) = a(n) - \mu \cdot y(n) \cdot \frac{g(n)}{1 + \varepsilon \cdot g^2(n)} \quad , \varepsilon \geq \mu \quad (3.47)$$

เมื่อ $a(n)$ คือ ค่าสัมประสิทธิ์ของตัวกรอง $y(n)$ คือ สัญญาณเอาต์พุตของตัวกรอง $g(n)$ คือค่าเกรเดียนต์ μ คือ ค่าก้าวกระโดด และ ε คือค่าพารามิเตอร์อิสระ เมื่อ $\varepsilon \geq \mu$ สมการที่ (3.46) จะให้ความเร็วในการลู่เข้าที่เท่ากับอัลกอริทึมแบบ PG แต่ค่าความแปรปรวนของสัมประสิทธิ์ต่ำกว่า และในสมการที่ (3.47) ความเร็วและค่าความแปรปรวนของสัมประสิทธิ์จะขึ้นอยู่กับข้อกำหนดค่าของพารามิเตอร์ ε ถ้า ε มีค่ามาก ค่าความแปรปรวนของสัมประสิทธิ์ก็จะต่ำลงไปด้วย อย่างไรก็ตาม อัลกอริทึมแบบ MNG ถึงจะมีความซับซ้อนในการคำนวณมากกว่าอัลกอริทึมแบบ PG แต่ก็เพียงเล็กน้อยเท่านั้น

อะแดปทีฟอัลกอริทึมแบบ Modified-MNG [27] เป็นอัลกอริทึมที่ปรับปรุงมาจากอัลกอริทึมแบบ MNG โดยการทำให้การลู่เข้าสู่ค่าตอบของอัลกอริทึมมีความรวดเร็วเพิ่มมากขึ้น และสามารถลดความซับซ้อนในการคำนวณลงได้ สมการสำหรับปรับค่าสัมประสิทธิ์ของอัลกอริทึมแบบ Modified-MNG แสดงได้ดังสมการ

$$a(n+1) = a(n) - \mu \cdot y(n) \cdot \operatorname{sgn} \left[\frac{g(n)}{1 + g(n)} \right] \quad (3.48)$$

ค่าก้าวกระโดด μ นอกจากจะมีผลต่อความเร็วในการลู่เข้าสู่ค่าตอบที่แท้จริงของอะแดปทีฟอัลกอริทึมแล้ว ยังมีผลต่อความละเอียดในการคำนวณอีกด้วย ถ้าค่าก้าวกระโดดมีค่ามาก ๆ ความรวดเร็วในการลู่เข้าสู่ค่าตอบของอัลกอริทึมก็ยิ่งรวดเร็ว แต่ในขณะเดียวกัน ความละเอียดในการคำนวณก็จะลดลง ทำให้ความผิดพลาดของค่าตอบสูงขึ้น เมื่อกำหนดให้ค่าก้าวกระโดดมีค่าน้อย ๆ ความละเอียดในการคำนวณจะเพิ่มขึ้น ความผิดพลาดของค่าตอบลดลง แต่ความรวดเร็วในการลู่เข้าสู่ค่าตอบก็จะช้าตามไปด้วย รูปที่ 3.13 แสดงค่า MSE ของสัมประสิทธิ์ของอัลกอริทึมแบบ Modified MNG เมื่อ กำหนดให้ค่าก้าวกระโดดมีค่าแตกต่างกัน

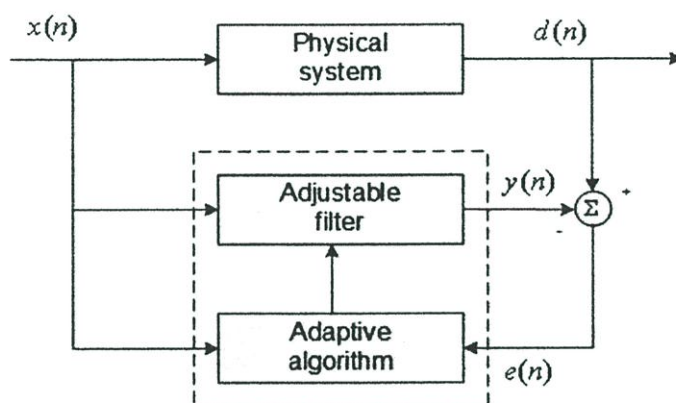


รูปที่ 3.13 ค่า MSE ของอัลกอริทึมแบบ Modified MNG เมื่อกำหนดให้ค่าก้าวกระโดดมีค่าแตกต่างกัน

3.5 การประยุกต์ใช้งานตัวกรองอะแดปทีฟ

ตัวกรองอะแดปทีฟเป็นองค์ประกอบส่วนหนึ่งของระบบที่มีประสิทธิภาพในอุตสาหกรรมสมัยใหม่ มีการนำมาใช้ในระบบที่มีประสิทธิภาพสูง เช่น การจำลองระบบและสถานการณ์ล่วงหน้า การปรับระดับสัญญาณและการกลับสัญญาณ การคาดคะเนสัญญาณแบบลิเนียร์ การวิเคราะห์ที่กู้คืนความถี่ที่สูญเสีย การขจัดสัญญาณสะท้อน และการขจัดสัญญาณสอดแทรก

3.5.1 การจำลองระบบและสถานการณ์ล่วงหน้า (System Identification and Forward Modeling)

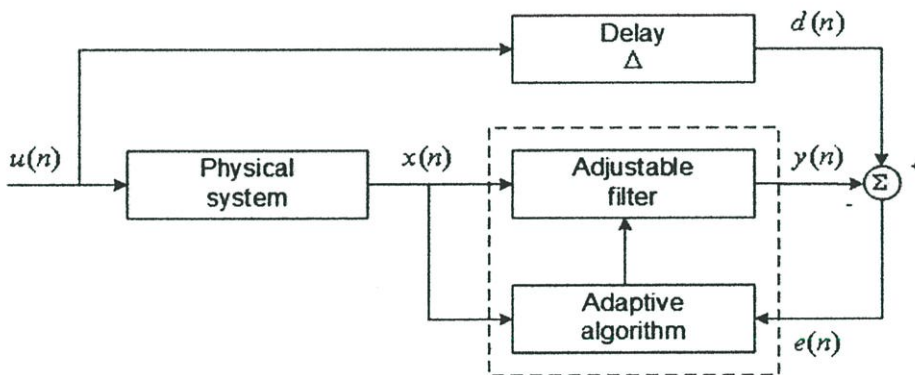


รูปที่ 3.14 บล็อกไดอะแกรมแบบจำลองระบบอะแดปทีฟ

การจำลองระบบเป็นขั้นตอนที่สำคัญในการออกแบบระบบควบคุม เพื่อสร้างแบบจำลองระบบกายภาพ (Physical system) ที่ควบคุมได้ ตามเงื่อนไขในการออกแบบที่ต้องการ ระบบต่าง ๆ จำนวนมากจะถูกควบคุมให้เปลี่ยนแปลงตามเวลาอย่างช้า ๆ จึงไม่เป็นที่นิยมใช้ออกแบบระบบควบคุมอุปกรณ์จำลอง ระบบควบคุมแบบอะแดปทีฟจึงเข้ามาแทนที่ระบบเดิม การจำลองระบบที่ทำงานแบบออนไลน์จะ ใช้การจำลองแบบอะแดปทีฟ ดังรูปที่ 3.14 ซึ่งตัวควบคุมสามารถปรับตัวเองได้ เช่น การควบคุมสัญญาณรบกวนและการสั่นสะเทือน (Active noise and vibration control (ANVC)) อัลกอริทึมทั้งหมดของ ANVC จะใช้ในเส้นทางที่สองที่อยู่ระหว่างตัวควบคุมและตัวตรวจจับความคลาดเคลื่อน ระบบ ANVC จะทำการจำลองระบบก่อนเข้าสู่การทำงานจริง หรือการอัปเดต (Update) ตามความเร็วของการเปลี่ยนแปลงระบบกายภาพที่เกิดขึ้น แบบจำลองอะแดปทีฟจะถูกกระตุ้นจากสัญญาณอินพุต $x(n)$ จำนวนมากและมีความต่อเนื่องกัน อัลกอริทึมอะแดปทีฟจะได้คำตอบเมื่อได้ค่าความคลาดเคลื่อนที่มีค่าน้อยที่สุดจากความแตกต่างระหว่างเอาต์พุตระบบกายภาพ $d(n)$ และ เอาต์พุตตัวกรองอะแดปทีฟ $y(n)$ ในทางอุดมคติค่าความคลาดเคลื่อนจะลดลงเป็นศูนย์ เมื่อ ค่า $y(n)$ เท่ากับ $d(n)$ หมายความว่า ผลตอบสนอง

อิมพัลส์ตัวกรอง เข้าใกล้ ผลตอบสนองระบบกายภาพ สิ่งที่สำคัญในการจำลองระบบกายภาพแบบอะแดปทีฟให้ประสบความสำเร็จ เริ่มต้นจากการเลือกโครงสร้างตัวกรองอะแดปทีฟที่ต้องการ เมื่อผลตอบสนองอุปกรณ์มีการสั้นสะท้อนตลอดเวลา จะใช้ตัวกรองอะแดปทีฟแบบผลตอบสนองอิมพัลส์ที่ไม่จำกัด (IIR) ทำการอัปเดต โดยใช้อัลกอริทึมแบบค่าความคลาดเคลื่อนสมการ (Equation error algorithm) หรือ อัลกอริทึมแบบค่าความคลาดเคลื่อนเอาต์พุต (Output error algorithm) แต่ถ้าผลตอบสนองอุปกรณ์มีการสั้นสะท้อนเป็นระยะเวลาสั้น ๆ มักจะใช้ตัวกรอง FIR แบบทรานสเวอร์ซอลเนื่องจากมีเสถียรภาพที่ดีกว่า และอัปเดตโดยใช้อัลกอริทึมแบบ LMS หรือ NLMS สำหรับแบบจำลอง FIR ที่มีความยาวมาก ๆ จะใช้อัลกอริทึมอะแดปทีฟในโดเมนความถี่ เช่น BFDAF(Block Frequency Domain Adaptive Filter) และ PBFDAF(Partitioned Block Frequency Domain Adaptive Filter) ซึ่งจะให้สมรรถนะที่ดีกว่าและมีการคำนวณที่ไม่ซับซ้อนมาก ส่วนตัวกรองอะแดปทีฟแบบแลตทิซก็มีความได้เปรียบกว่าแบบอื่น และประสบความสำเร็จในการประยุกต์ใช้งานหลาย ๆ อย่าง เช่น การจำลองพื้นผิวโลกในการสำรวจแผ่นดินไหว มักจะใช้อัลกอริทึมแบบ LMS-lattice หรือ RLS-lattice

3.5.2 Equalization and Inverse Modeling

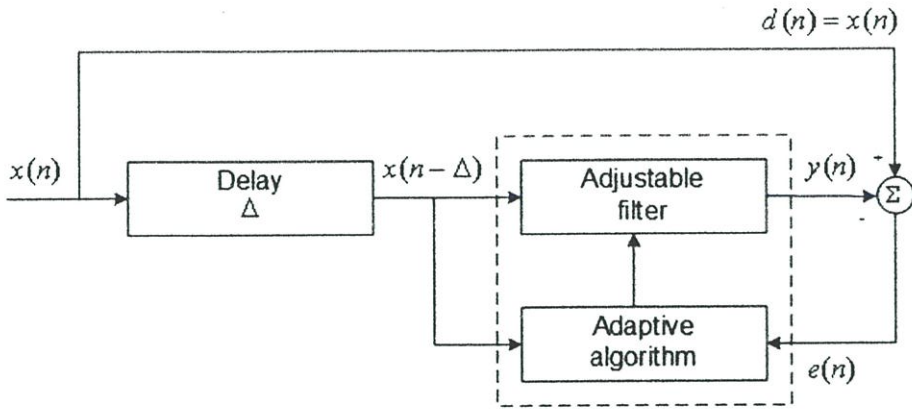


รูปที่ 3.15 บล็อกไดอะแกรมแบบจำลองระบบอะแดปทีฟอินเวอร์ส

แนวคิดพื้นฐานของการจำลองแบบอินเวอร์ส รู้จักกันดีว่าเป็น Deconvolution หรือ Equalization ดังรูปที่ 3.15 สัญญาณอินพุต $u(n)$ จะถูกกรองผ่านระบบกายภาพ เช่น ช่องสัญญาณสื่อสาร สัญญาณอินพุต $x(n)$ ที่ผิดเพี้ยนจะถูกกรองผ่านแบบจำลองอะแดปทีฟอินเวอร์สของระบบกายภาพ ได้สัญญาณเอาต์พุต $y(n)$ ที่มีค่าใกล้เคียงกับสัญญาณอินพุต $u(n)$ เท่าที่จะเป็นไปได้ สัมประสิทธิ์ตัวกรองอะแดปทีฟจะปรับตัวให้ความแตกต่างระหว่างสัญญาณเอาต์พุต $y(n)$ และสัญญาณอินพุต $u(n)$ ที่หน่วงเวลาออกไป มีค่าน้อยที่สุด ค่าหน่วงเวลา Δ จะต้องเลือกให้เท่ากับผลรวมของการหน่วงเวลาของระบบกายภาพและตัวกรองอะแดปทีฟ การคอนโวลูชัน

ระหว่างผลตอบสนองตัวกรองอะแคปทีฟและผลตอบสนองระบบกายภาพมีค่าเท่ากับ อิมพัลส์ที่ถูก
 หน่วงเวลา $\delta(n - \Delta)$ ผลตอบสนองความถี่ของตัวกรองอะแคปทีฟ $W(z)$ คือ การประมาณค่า
 ของส่วนกลับผลตอบสนองความถี่ของระบบกายภาพ $H(z)$ นั่นคือ $W(z) \cong z^{-\Delta} / H(z)$
 ตัวกรองอะแคปทีฟที่ประยุกต์ใช้งานตามตัวอย่างนี้ เป็นพื้นฐานของความพยายามกู้คืนสัญญาณ
 อินพุตที่ผิดเพี้ยนจากระบบกายภาพให้ได้มากที่สุด แบบจำลองอินเวอร์ตมีกรใช้งานในระบบ
 ควบคุมและระบบสื่อสารจำนวนมาก เช่น การปรับระดับช่องสัญญาณ (Channel equalization)
 ซึ่งระบบกายภาพคือ ช่องสัญญาณการสื่อสารหนึ่งช่อง และตัวกรองอะแคปทีฟที่อ้างถึงก็คือ ตัว
 กรองปรับระดับช่องสัญญาณอะแคปทีฟ ในกรณีนี้สัญญาณอินพุต $u(n)$ คือ ข้อมูลที่ถูกส่ง
 มักจะเป็นสัญญาณพัลส์ที่มอดูเลตแล้ว ข้อมูลที่สูญเสียเกิดจากช่องสัญญาณสื่อสารในทิศทางต่าง
 ๆ การสูญเสียของสัญญาณที่รุนแรงที่สุดคือ การแทรกสอดระหว่างสัญลักษณ์ (Inter-symbol
 interference) ซึ่งเป็นผลมาจากความจริงที่ว่า ผลตอบสนองช่องสัญญาณไม่เป็นแบบอิมพัลส์ แต่
 จะเป็นแบบ nonzero ส่งผลให้เกิดการรบกวนสัญลักษณ์ข้อมูลข้างเคียง ทำให้การค้นหา
 สัญลักษณ์แบบกำหนดระดับอย่างง่าย ๆ ใช้ไม่ได้ผล อัตราค่าความคลาดเคลื่อนสัญลักษณ์มีค่า
 เพิ่มขึ้น วงจรปรับระดับแบบอะแคปทีฟ (Adaptive equalizer) จึงถูกนำมาใช้เพื่อลดการสูญเสีย
 จากการแทรกสอดระหว่างสัญลักษณ์ ปัญหาในการกำหนดค่าเริ่มต้นของวงจรปรับระดับ
 ช่องสัญญาณ คือ การที่ภาครับไม่สามารถหาสัญญาณอ้างอิง $u(n)$ ในภาวะการณส่งแบบปกติเพื่อ
 ใช้ในการอัปเดตสัมประสิทธิ์ของวงจรปรับระดับได้ จึงแก้ปัญหาดังกล่าวด้วยการเทรนนิ่งก่อนส่งจริง
 ภาคส่งจะส่งลำดับสัญลักษณ์เทรนนิ่งที่ภาครับรู้อยู่แล้วว่าเป็นอะไร ภาครับจะทำการสร้างลำดับ
 เทรนนิ่งขึ้นเอง และใช้ในการปรับสัมประสิทธิ์วงจรปรับระดับให้อัตราค่าความคลาดเคลื่อน
 สัญลักษณ์มีค่าน้อยที่สุด เมื่อได้สัมประสิทธิ์ที่มีความเหมาะสมแล้ว สัญลักษณ์ที่ตรวจจับได้ก็จะ
 เหมือนกับด้านส่งและใช้เป็นสัญญาณ $d(n)$ ในการปรับสัมประสิทธิ์ตัวต่อไปของวงจรปรับ
 ระดับเพื่อติดตามการเปลี่ยนแปลงในช่องสัญญาณได้ การทำงานวิธีนี้เป็นวิธีการแบบตรง ๆ และ
 สามารถทำงานติดตามการเปลี่ยนแปลงในช่องสัญญาณสื่อสารที่มีความเร็วไม่มากนักได้เป็นอย่างดี
 ในการออกแบบมักจะใช้ตัวกรอง อะแคปทีฟทรานสเวอร์สซอล FIR

3.5.3 Adaptive Linear Prediction



รูปที่ 3.16 บล็อกไดอะแกรมตัวกรองแบบคาดคะเนเดินหน้า

ระบบคาดคะเนเดินหน้า เป็นการประมาณค่าตัวอย่างของลำดับอินพุตปัจจุบัน $x(n)$ จากการถึเนียร์คอมไบเนชันของตัวอย่างอินพุต M ในอดีต $x(n-\Delta), x(n-\Delta-1), \dots, x(n-\Delta-M+1)$ สัญญาณที่ต้องการจะถูกป้อนให้เป็นอินพุตของระบบ $d(n) = x(n)$ และ อินพุตของตัวกรองอะแดปทีฟ คือ อินพุตของระบบที่ถูกหน่วงเวลาออกไป $x(n-\Delta)$ เอาต์พุตตัวกรอง $y(n)$ ได้จากการคอมไบเนชันของตัวอย่างอินพุตในอดีต อัลกอริทึมจะปรับสัมประสิทธิ์ของตัวกรองอะแดปทีฟให้สัญญาณคลาดเคลื่อนมีค่าน้อยที่สุด $e(n) = d(n) - y(n)$ เมื่อตัวกรองถูกเข้าหาคำตอบได้แล้ว สัญญาณคลาดเคลื่อน $e(n)$ จะไม่มีความสัมพันธ์กับสัญญาณอินพุตของตัวกรอง $x(n-\Delta)$ จะเห็นได้ว่า $x(n)$ มีคุณลักษณะเช่นเดียวกับลิเนียร์คอมไบเนชันของ $y(n)$ บวกกับ $e(n)$ ตัวกรองอะแดปทีฟ (adjustable filter) ตามรูปที่ 3.16 เรียกว่า ตัวกรองคาดคะเนเดินหน้า (forward predictor) อาจจะใช้ตัวกรองพื้นฐานชนิดใด ๆ ก็ได้ โครงสร้างตัวกรองที่ใช้มากที่สุดในการคาดคะเนคือ ตัวกรองแบบทรานสเวอร์และ ตัวกรองแบบแลตทิซ เอาต์พุตของตัวกรองคาดคะเนล่วงหน้าที่มีอันดับ M^{th} ของอินพุต $x(n)$ สามารถกำหนดได้ดังสมการ

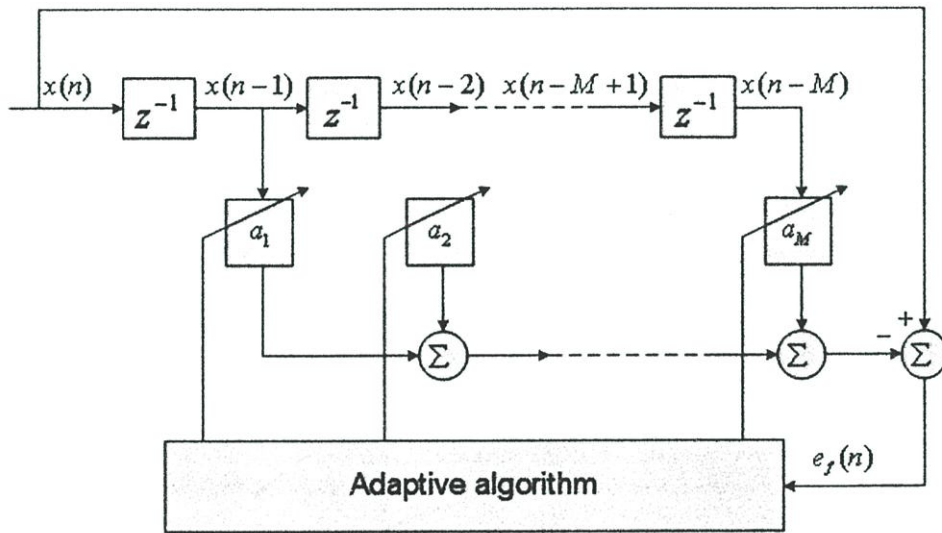
$$y(n) = \sum_{i=1}^M a_i x(n-i) \quad (3.49)$$

สัญญาณคลาดเคลื่อน $e_f(n) = x(n) - y(n)$ คือ ค่าความคลาดเคลื่อนคาดคะเนล่วงหน้าที่มีอันดับ M^{th} การแก้ปัญหาตัวกรองแบบวินเนอร์ คือ การทำให้ค่าความคลาดเคลื่อนสมบูรณ์ยกกำลังสอง $|e_f(n)|^2$ มีค่าน้อยที่สุด ค่าสัมประสิทธิ์คาดคะเนเดินหน้าที่เหมาะสม a สามารถหาได้จากสมการ

$$a = Rr \quad (3.50)$$

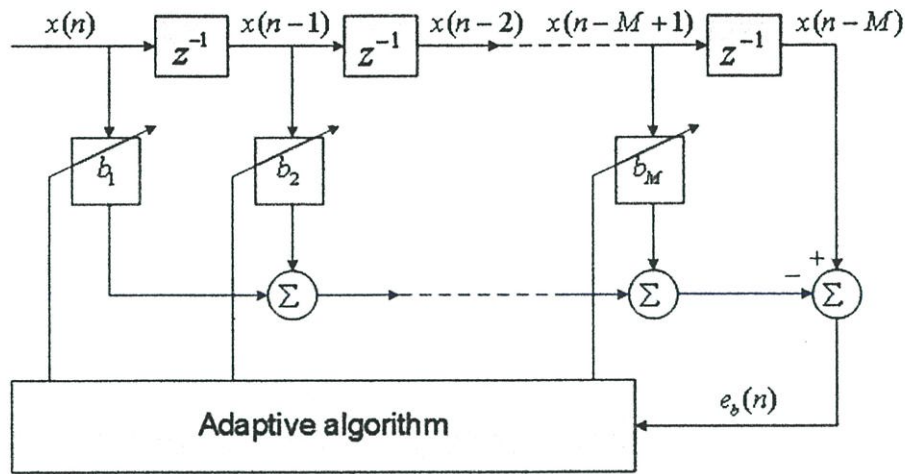
นิยามฟังก์ชันสหสัมพันธ์ของอินพุตที่หน่วงเวลาไป k คือ $r(k) = E[x(n)x(n-k)]$ ดังนั้น R และ r เขียนได้ว่า

$$R = \begin{bmatrix} r(0) & r(1) & \cdots & r(m-1) \\ r(1) & r(0) & \cdots & r(m-2) \\ \vdots & \vdots & \ddots & \vdots \\ r(m-1) & r(m-2) & \cdots & r(0) \end{bmatrix}, \quad r = \begin{bmatrix} r(1) \\ r(2) \\ \vdots \\ r(m) \end{bmatrix} \quad (3.51)$$



รูปที่ 3.17 บล็อกไดอะแกรมการคาดคะเนเดินหน้าทรานสเวอร์ซอล

ระบบที่มีอินพุต $x(n)$ และ เอาต์พุต $e(n)$ คือ ตัวกรองแบบความคลาดเคลื่อนคาดคะเนเดินหน้า (forward prediction error filter) ซึ่งจะเหมือนกับ วงจรคาดคะเนย้อนกลับอันดับ M ที่มีสัญญาณที่ต้องการ $d(n) = x(n-M)$ และ อินพุต $x(n)$ ดังรูปที่ 3.18



รูปที่ 3.18 บล็อกไดอะแกรมการคาดคะเนย้อนกลับทรานสเวอร์ซอล

วงจรคาดคะเนย้อนกลับ จะทำการประมาณค่า $x(n-M)$ จากลิเนียร์คอมไบเนชันของ $x(n), x(n-1), \dots, x(n-M+1)$ โดยการทำให้ค่าสัญญาณความคลาดเคลื่อน $e_b(n) = x(n-M) - y(n)$ มีค่าน้อยที่สุด เอาต์พุตของตัวกรองได้จากการคาดคะเนย้อนกลับที่อันดับ M^{th} ของอินพุต $x(n)$ สามารถหาได้จากสมการ

$$y(n) = \sum_{i=1}^M b_i x(n-i+1) \quad (3.52)$$

สัญญาณความคลาดเคลื่อน $e_b(n)$ คือ ความคลาดเคลื่อนคาดคะเนย้อนกลับอันดับที่ M^{th} การแก้ปัญหาวัดกรองแบบวินเนอร์ คือ การทำให้ค่าความคลาดเคลื่อนสมบูรณ์ยกกำลังสอง $|e_b(n)|^2$ มีค่าน้อยที่สุด ด้วยค่าสัมประสิทธิ์คาดคะเนเดินหน้าที่เหมาะสม b กำหนดได้ด้วยสมการ

$$b = Rr_b \quad (3.53)$$

เมื่อ R คือ เมตริกซ์ เช่นเดียวกับสมการที่ (3.51) พิจารณาอินพุต $x(n)$ เป็นแบบสเตชันนารี และ r_b จะได้

$$R = \begin{bmatrix} r(0) & r(1) & \cdots & r(m-1) \\ r(1) & r(0) & \cdots & r(m-2) \\ \vdots & \vdots & \ddots & \vdots \\ r(m-1) & r(m-2) & \cdots & r(0) \end{bmatrix}, \quad r_b = \begin{bmatrix} r(m) \\ r(m-1) \\ \vdots \\ r(1) \end{bmatrix} \quad (3.54)$$

ซึ่ง r_b จะเหมือนกับ r ในสมการที่ (3.51) เพียงแต่อันดับอีลิเมนต์จะวางย้อนกลับกัน อธิบายได้ว่าสัมประสิทธิ์ที่เหมาะสมของวงจรถาดคคะเนย้อนกลับมีค่าเหมือนกับสัมประสิทธิ์ของวงจรถาดคคะเนคินหน้า เพียงแต่วางอันดับอีลิเมนต์ย้อนกลับกัน ดังเช่น

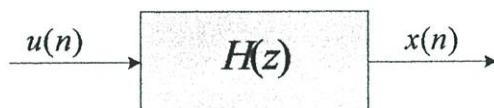
$$b_i = a_{M+1-i}, \quad \text{เมื่อ } i = 1, 2, \dots, M \quad (3.55)$$

ตัวกรองแบบความคลาดเคลื่อนคคะเนย้อนกลับ คือ ระบบที่มีอินพุต $x(n)$ และ เอาต์พุต $e(n)$ เมื่อได้สัมประสิทธิ์ที่เหมาะสมแล้ว ลำดับสัญญาณอินพุต $x(n)$ และ ความคลาดเคลื่อนคคะเนย้อนกลับ $e_b(n)$ จะไม่มีความสัมพันธ์กัน นอกจากนั้นความคลาดเคลื่อนคคะเนย้อนกลับ $e_{bJ}(n)$ อันดับที่ J^{th} เมื่อ $J = 0, 1, \dots, M$ จะไม่มีความสัมพันธ์กับค่าใด ๆ เลย นี่คือคุณสมบัติของแลตทิซที่ใช้ในการประมาณค่าร่วมแลตทิซ ตามความสัมพันธ์ตัวอย่างลำดับอินพุต

นอกจากวงจรถาดคคะเนแบบทรานสเวอร์ซอลแล้ว วงจรถาดคคะเนแบบแลตทิซก็มีการนำมาใช้งานอย่างกว้างขวางเช่นกัน ทั้งสองแบบมีความคล้ายกันมาก เป็นความสัมพันธ์แบบเฉพาะระหว่างสัมประสิทธิ์ของวงจรถาดคคะเนทรานสเวอร์ซอล (ล่องหน้าและย้อนกลับ) ที่ลำดับ M^{th} กับ สัมประสิทธิ์การสะท้อนที่เหมาะสมของวงจรถาดคคะเนแบบแลตทิซ ที่มีอันดับเท่ากัน

3.5.4 Adaptive Autoregressive Spectrum Analysis

แถบพลังงานของกระบวนการสุ่มทางเวลาไม่ต่อเนื่อง หาได้จากการสมมุติฐานกระบวนการจำลองแบบลิเนียร์ ดังรูปที่ 3.19



รูปที่ 3.19 Autoregressive process modeling.

จากรูป $x(n)$ เป็นเอาต์พุตของตัวกรองลิเนียร์ $H(\omega)$ อินพุตเป็นสัญญาณรบกวนขาว $u(n)$ แถบพลังงานของ $x(n)$ หาได้จาก

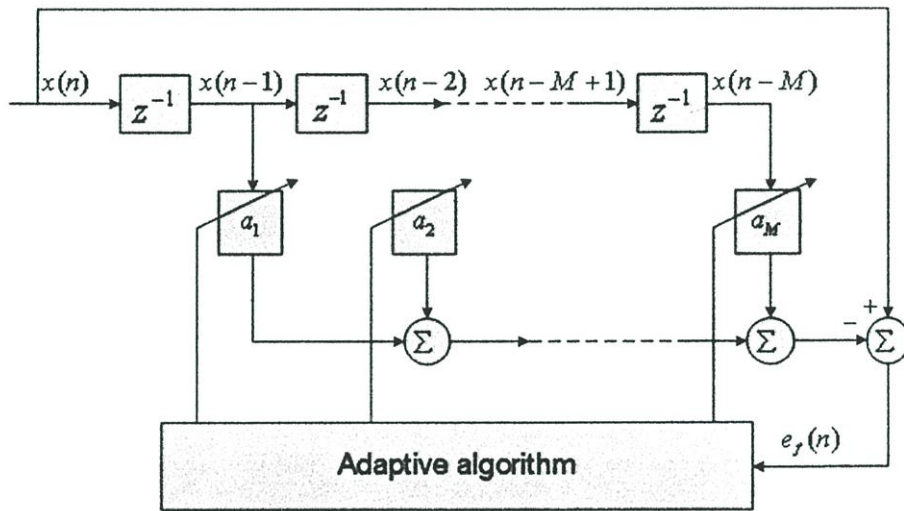
$$S_x(\omega) = \sigma_u^2 |H(\omega)|^2 \quad (3.56)$$

เมื่อ σ_u^2 คือ ความแปรปรวนของสัญญาณรบกวนขาว $u(n)$ ที่มีแถบพลังงานเรียบเท่ากันหมด จากสมการที่ (3.56) จะเห็นได้ว่า แถบพลังงานของเอาต์พุต $x(n)$ หาได้จากการประมาณค่า

ฟังก์ชันถ่ายโอนของระบบ $H(\omega)$ ด้วยการใช้โครงสร้างวงจรคาดคะเนแบบปรับตัว ในทางปฏิบัติเมื่อตัวกรอง $H(\omega)$ เป็นแบบ All-pole autoregressive model และมีฟังก์ชันถ่ายโอนตามสมการ

$$H(\omega) = \frac{1}{1 - \sum_{k=1}^M a_k e^{-j\omega k}} \quad (3.57)$$

กรณีนี้ พารามิเตอร์ของ AR $\{a_1, a_2, \dots, a_M\}$ เป็นค่าประมาณที่ได้จากตัวกรองความคลาดเคลื่อนคาดคะเนเดินหน้าปรับได้แบบทรานสเวอร์ซอล ในรูปที่ 3.20



รูปที่ 3.20 บล็อกโคโตะแกรมตัวกรองคาดคะเนเดินหน้าแบบทรานสเวอร์ซอล

แถบพลังงานของ $x(n)$ หาได้จากสมการ

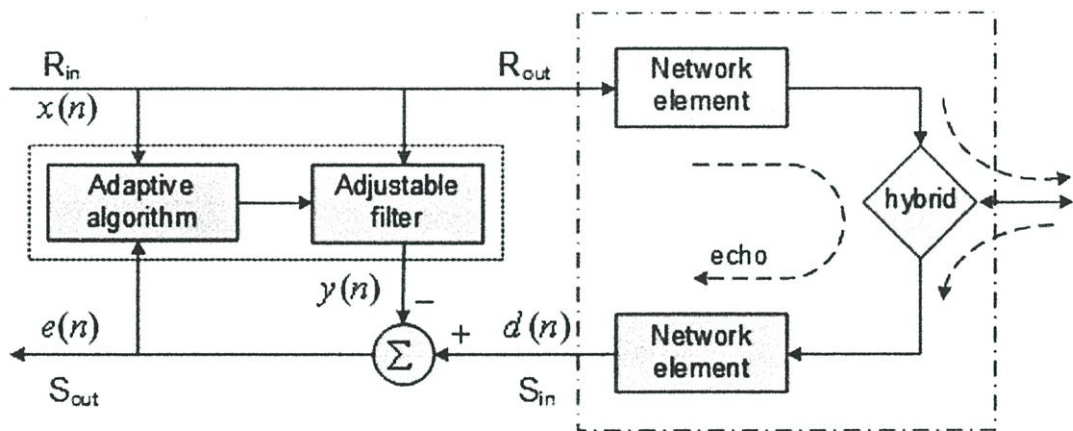
$$S_x(\omega, n) = \frac{\sigma_u^2}{\left| 1 - \sum_{k=1}^M a_k(n) e^{-j\omega k} \right|^2} \quad (3.58)$$

จากสมการที่ (3.58) แถบพลังงาน $S_x(\omega, n)$ และ พารามิเตอร์ $a_k; k=1, 2, \dots, M$ จะเปลี่ยนแปลงตามเวลา เป็นการวัดขนาดของความถี่ ณ ช่วงเวลานั้นของการประมวลผล $x(n)$ จากพารามิเตอร์ Autoregressive

3.5.5 การขจัดสัญญาณสะท้อน (Echo Cancellation)

วงจรขจัดสัญญาณสะท้อนเป็นองค์ประกอบที่สำคัญในการประยุกต์ใช้งานต่าง ๆ โดยเฉพาะ การสื่อสารโทรคมนาคม วงจรขจัดสัญญาณสะท้อนแบ่งออกได้เป็น 2 ชนิด คือ วงจรขจัดสัญญาณสะท้อนในโครงข่าย (Network Echo Cancellers (NEC)) และ วงจรขจัดสัญญาณสะท้อนในระบบเสียง (Acoustic Echo Cancellers (AEC)) วงจรขจัดสัญญาณสะท้อนทั้งสองชนิดใช้ตัวกรองอะแดปทีฟในการประมาณค่าสัญญาณสะท้อนและสัญญาณที่ตามมา เพื่อลดสัญญาณสะท้อนจากสัญญาณต้นส่ง ดังนี้

3.5.5.1 วงจรขจัดสัญญาณสะท้อนในโครงข่าย (Network Echo Cancellers (NEC))

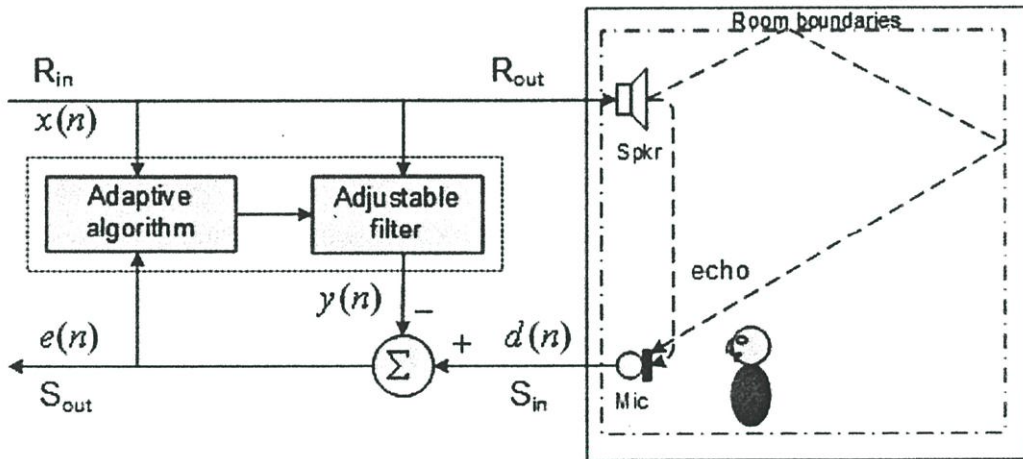


รูปที่ 3.21 บล็อกไดอะแกรมของวงจรขจัดสัญญาณสะท้อนในโครงข่าย

วงจรขจัดสัญญาณสะท้อนในโครงข่ายโทรศัพท์เป็นสิ่งจำเป็นมากในโครงข่ายโทรศัพท์ โดยเฉพาะอย่างยิ่งในการสนทนาที่มีระยะทางไกล ๆ การสะท้อนในสายโทรศัพท์ส่วนมากจะเกิดจากอุปกรณ์ที่เรียกว่า ไฮบริด ซึ่งติดตั้งอยู่ที่ชุมสายโทรศัพท์ หน้าทีของไฮบริดคือ การเปลี่ยนการเชื่อมต่อด้านปลายทางที่ต่อเครื่องโทรศัพท์เข้ากับชุมสายโทรศัพท์ โดยใช้สายทองแดงสองเส้น (ด้านขวามือในรูปที่ 3.21) และการเชื่อมต่อแบบสี่สายที่ใช้เชื่อมต่อระหว่างชุมสาย (ด้านซ้ายมือในรูปที่ 3.21) ไฮบริดที่ดีควรจะส่งผ่านสัญญาณเสียงพูดขาเข้าทั้งหมด (R_{in}) ไปยังด้านสายทองแดงสองเส้นโดยไม่มีการสูญหาย แต่ในทางปฏิบัติสัญญาณเสียงขาเข้าส่วนหนึ่งจะย้อนกลับไปที่ทางด้านส่ง (S_{in}) ทำให้ผู้เรียกได้ยินเสียงของตัวเองที่ถูกหน่วงเวลาออกไป ตามการเดินทางที่ล่าช้าของสัญญาณเสียงในโครงข่าย ถ้าโครงข่ายมีความยาวมาก การรบกวนก็จะมากขึ้นตามไปด้วย เมื่อนำโครงข่ายโทรศัพท์ไปใช้ในการสื่อสารข้อมูล เช่น โมเด็ม หรือ แฟกซ์ สัญญาณสะท้อนไม่ว่าจะสั้นหรือยาวจะส่งผลกระทบต่อที่รุนแรงได้ การแก้ปัญหานี้ ทำได้โดยการใช้วงจรขจัดสัญญาณสะท้อนในโครงข่าย (NEC) ต่อเข้ากับด้านวงจรสี่สายของไฮบริด ดังรูปที่ 3.21 โดยทั่วไปมักจะใช้

ตัวกรองอะแดปทีฟทรานสเวอร์ซอล สัมประสิทธิ์ตัวกรองจะปรับให้สัญญาณคลาดเคลื่อน $e(n)$ ที่ด้าน S_{out} มีค่าน้อยที่สุด สัมประสิทธิ์ตัวกรองอะแดปทีฟจะมีลักษณะคล้ายกับแบบจำลอง FIR ของวงจรไฮบริด เนื่องจาก ไฮบริดและตัวกรองอะแดปทีฟใช้อินพุตร่วมกัน และค่าความคลาดเคลื่อนจะได้จากความแตกต่างระหว่าง เอาต์พุตของไฮบริดและสัญญาณจากเอาต์พุตของตัวกรอง ตัวกรองอะแดปทีฟจะกรองสัญญาณที่รับได้ ทำการจำลองไฮบริดเพื่อประมาณค่าสัญญาณสะท้อน ทำให้ได้สัญญาณเอาต์พุต $y(n)$ ดังรูปที่ 3.21 นำมาลบกับสัญญาณที่จุด S_{in} ทำให้สัญญาณสะท้อนที่จุด S_{out} ลดลง การลดสัญญาณสะท้อนที่มีประสิทธิภาพ สัมประสิทธิ์ของตัวกรองแบบ ทรานสเวอร์ซอลควรที่จะประมาณค่าผลตอบสนองอิมพัลส์ของไฮบริดได้ถูกต้อง โดยทั่วไปผลตอบสนองอิมพัลส์ไฮบริดตั้งสมมุติฐานไว้ที่ค่าประมาณ 20 ถึง 30 ms วงจรขจัดสัญญาณสะท้อนทำงานที่ความถี่สุ่มตัวอย่าง 8 kHz จำนวนสัมประสิทธิ์ของตัวกรองอะแดปทีฟ โดยทั่วไปจะมีความยาวอยู่ในช่วง 240 ค่า อัลกอริทึมที่ใช้ ได้แก่ NLMS Leaky NLMS เป็นต้น

3.5.5.2 วงจรขจัดสัญญาณสะท้อนในระบบเสียง (Acoustic Echo Cancellers (AEC))

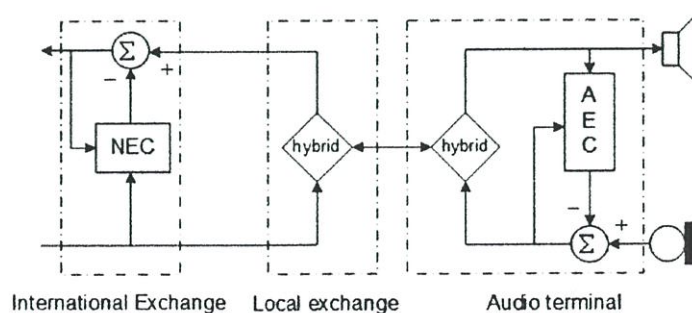


รูปที่ 3.22 บล็อกไดอะแกรมของวงจรขจัดสัญญาณสะท้อนในระบบเสียง

การใช้งานที่จำเป็นของวงจรขจัดสัญญาณสะท้อนในระบบเสียง เช่น ระบบแฮนด์ฟรีหรือการพูดผ่านไมโครโฟนที่ตัวเครื่องโดยไม่ต้องยกหูโทรศัพท์ (Speakerphone) การสื่อสารระหว่างคอมพิวเตอร์ การประชุมทั้งภาพและเสียง การสั่งงานด้วยเสียงพูด และ อื่น ๆ อีกมากมาย ปัญหาที่เกิดจากการสะท้อนของเสียงแสดงในรูปที่ 3.22 จากรูปเป็นการแสดงการประชุมทางเสียงเพียงด้านเดียว สัญญาณเสียงพูดที่เข้ามา หรือ สัญญาณเสียง (R_{out}) ส่งผ่านไปยังลำโพงให้ผู้ฟังในห้องประชุมได้ยินเสียงจากคู่สนทนา สัญญาณเสียงพูดของผู้พูดที่อยู่ในห้อง (S_{in}) จะถูกส่งผ่านไปไมโครโฟน ผ่านช่องสัญญาณสื่อสารไปยังคู่สนทนาปลายทาง สิ่งที่เป็นปัญหาคือ สัญญาณเสียง

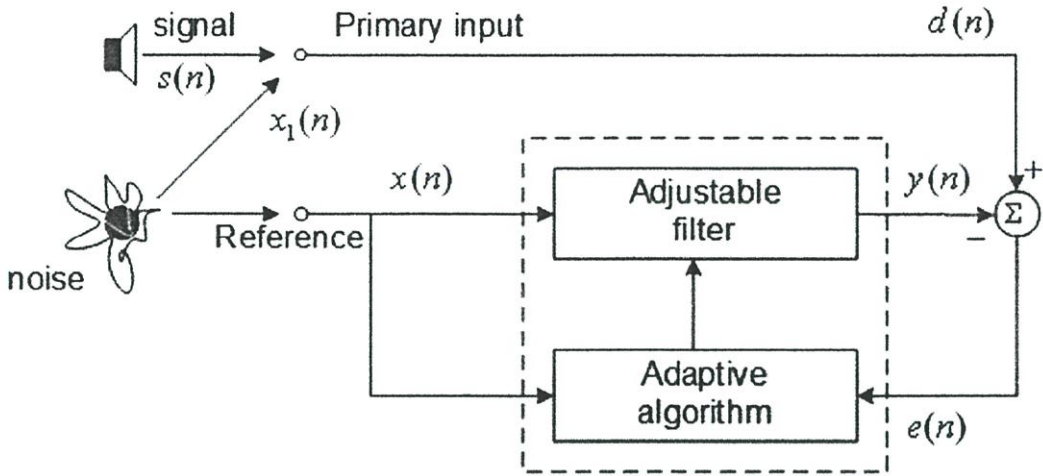
พูดจากผู้พูดด้านไกล เมื่อส่งผ่านออกไปยังลำโพงแล้วจะสะท้อนกับผนังห้องทำให้เกิดการหน่วงเวลา ย้อนเข้าไมโครโฟน และจะถูกส่งไปพร้อมกับสัญญาณเสียงพูดของผู้พูดที่อยู่ในห้อง ไปยังปลายทางอีกด้านหนึ่ง ด้านปลายทางผู้พูดจะได้ยินเสียงตัวเองที่ซำออกไป ตามความยาวของโครงข่ายและการหน่วงเวลาของเสียงอะคูสติก เป็นการรบกวนที่เกิดจากเสียงสะท้อนกลับของตัวเอง ผู้พูดก็จะหยุดพูดจนกว่าเสียงสะท้อนของตัวเองหายไป และเกิดซ้ำ ๆ กันไปเรื่อยตลอดการสนทนา ถ้าการหน่วงเวลาใช้เวลานานขึ้น เช่น ในกรณีของ มือถือ และ Voice Over Internet Protocol สัญญาณเสียงสะท้อนจะทำให้คุณภาพการสื่อสารลดลงอย่างเห็นได้ชัด จากรูปที่ 3.22 หน้าที่หลักของ AEC จะต้องสร้างค่าประมาณสัญญาณเสียงสะท้อน และนำมาลบออกจากสัญญาณไมโครโฟน กรณีที่ดีที่สุด ค่าที่ประมาณได้ถูกต้อง สัญญาณเสียงสะท้อนจะถูกขจัดได้อย่างสมบูรณ์ แต่ในทางปฏิบัติไม่ใช่เรื่องง่ายเลย โดยทั่วไป AEC มักจะใช้ ตัวกรองแบบ FIR ทรานส์เวอร์ซอลในการขจัดสัญญาณเสียงสะท้อน สัมประสิทธิ์ของตัวกรอง FIR จะถูกปรับโดยใช้อะแดปทีฟอัลกอริทึมเพื่อให้สัญญาณ S_{out} มีค่าน้อยที่สุด ซึ่งก็คือ สัญญาณความคลาดเคลื่อนของตัวกรองอะแดปทีฟ สัญญาณอ้างอิงหรือสัญญาณอินพุตของตัวกรองอะแดปทีฟคือสัญญาณเสียงพูดด้านไกล R_{in} ขณะที่สัญญาณที่ต้องการ คือสัญญาณเสียงพูดด้านใกล้ S_{in} เมื่อการหาคำตอบของตัวกรองเข้าใกล้คำตอบแล้ว สัมประสิทธิ์ของตัวกรองอะแดปทีฟจะมีผลตอบสนองอิมพัลส์ เท่ากับ ผลตอบสนองอิมพัลส์ทางเสียงระหว่างลำโพง และ ไมโครโฟน รวมทั้งการสะท้อนของห้องด้วย เนื่องจากผลตอบสนองอิมพัลส์ของเสียงมีค่าหลายร้อยมิลลิวินาที การใช้ตัวกรองอะแดปทีฟ FIR จึงต้องใช้สัมประสิทธิ์จำนวนหลายพันค่า การปรับค่าสัมประสิทธิ์ที่มีจำนวนมาก จึงเป็นความท้าทายอย่างมาก ดังนั้น จึงต้องหาอัลกอริทึมที่ดีและมีประสิทธิภาพมาใช้เพื่อลดความซับซ้อนในการคำนวณลงและทำให้ความรวดเร็วในการหาคำตอบเพิ่มขึ้น ตัวอย่างอัลกอริทึมที่ใช้งานอยู่ เช่น Block Frequency Domain Adaptive Filter (BFDAF) Partitioned BFDAF(PBFDAF) NLMS Leaky NLMS

โดยทั่วไปแล้ว การขจัดสัญญาณสะท้อน จะใช้วงจรขจัดสัญญาณสะท้อนทั้งสองชนิดร่วมกัน ดังรูปที่ 3.22



รูปที่ 3.23 บล็อกไดอะแกรมของการสื่อสารที่ใช้วงจรขจัดสัญญาณสะท้อนทั้งสองชนิดร่วมกัน

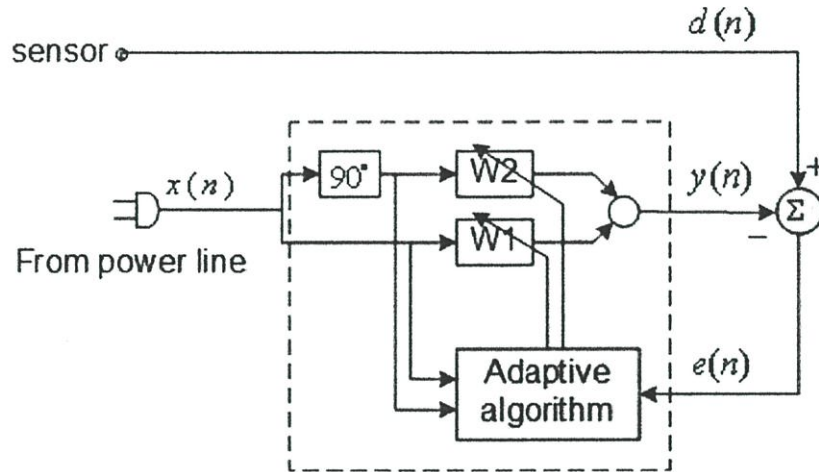
3.5.6 การขจัดสัญญาณรบกวนแบบอะแดปทีฟ



รูปที่ 3.24 บล็อกไดอะแกรมการขจัดสัญญาณรบกวนแบบอะแดปทีฟ

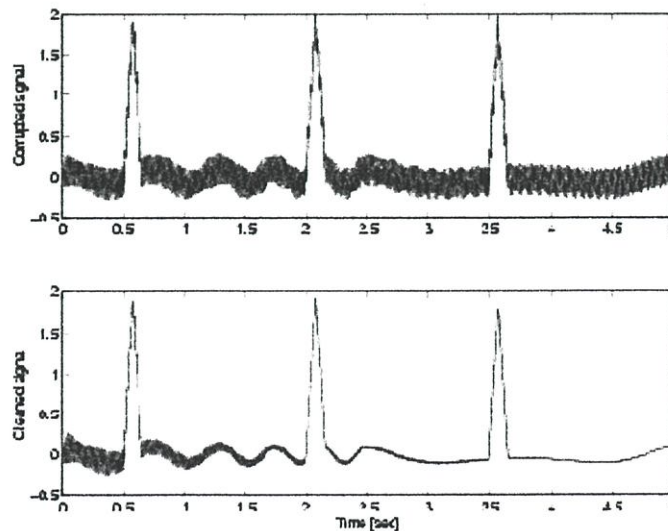
การใช้งานตัวกรองอะแดปทีฟจำนวนมาก เช่น ในการขจัดสัญญาณรบกวน การควบคุมการสั่นสะเทือน การสร้างลำแสง และการขจัดสัญญาณสะท้อน ซึ่งจัดอยู่ในประเภทของการขจัดสัญญาณรบกวนแบบอะแดปทีฟ ทฤษฎีเบื้องต้นของการขจัดสัญญาณรบกวนแบบอะแดปทีฟแสดงในรูปที่ 3.24 ในการขจัดสัญญาณรบกวนทั้งหมด สัญญาณที่ต้องการ จะถูกรบกวนด้วยสัญญาณรบกวนที่ไม่มีความสัมพันธ์กัน และ ต้องกู้คืนสัญญาณที่ต้องการจากสัญญาณรบกวนให้ได้ การขจัดสัญญาณรบกวนแบบอะแดปทีฟ มาจากการใช้ตัวเซ็นเซอร์รับสัญญาณสองตัวแยกกัน เช่น เซอร์ตัวแรกจะใช้เป็นอินพุตอ้างอิงหลัก รับทั้งสัญญาณที่ต้องการและสัญญาณรบกวน $d(n) = s(n) + x_1(n)$ ขณะที่ $s(n)$ คือ สัญญาณที่สร้างขึ้นจากแหล่งจ่ายสัญญาณและถูกส่งผ่านเข้าไปยังช่องสัญญาณที่อยู่ระหว่างแหล่งจ่ายและเซ็นเซอร์หลัก และ $x_1(n)$ คือสัญญาณรบกวนที่เซ็นเซอร์หลัก เช่น เซอร์ตัวที่สอง (อินพุตอ้างอิง) จะรับสัญญาณรบกวน $x(n)$ ซึ่งมีความสัมพันธ์กันกับ $x_1(n)$ แต่จะไม่มีความสัมพันธ์กับสัญญาณ $s(n)$ สัญญาณรบกวนอ้างอิง $x(n)$ จะถูกกรองด้วยตัวกรองอะแดปทีฟเพื่อสร้างเอาต์พุต $y(n)$ สัมประสิทธิ์ตัวกรองอะแดปทีฟจะถูกปรับให้เอาต์พุต $y(n)$ มีค่าใกล้เคียงกับสัญญาณรบกวนที่เซ็นเซอร์หลัก $x_1(n)$ มากที่สุด จะได้สัญญาณที่กู้คืนมาเมื่อ $e(n) = d(n) - y(n)$ จากทฤษฎีนี้ จะเห็นว่าสัญญาณรบกวนต่าง ๆ ที่ไม่ต้องการ สามารถขจัดได้ด้วยการต่อสัญญาณรบกวนอ้างอิงเพิ่มเข้ามาในวงจร ประกอบไปด้วยตัวเซ็นเซอร์รับสัญญาณที่มีความสัมพันธ์กันกับสัญญาณรบกวน แต่ไม่มีความสัมพันธ์กับสัญญาณที่ต้องการ เอาต์พุตจากเซ็นเซอร์อ้างอิงแต่ละตัวจะถูกกรองด้วยตัวกรองอะแดปทีฟ เอาต์พุตของตัวกรองอะแดปทีฟทั้งหมดจะรวมเข้าด้วยกันเพื่อประมาณค่าสัญญาณรบกวนที่มีอยู่ทั้งหมด แล้วนำผลลบออกจากสัญญาณจริงที่เสียหายจากสัญญาณรบกวน ก็

จะได้สัญญาณที่ต้องการ การประยุกต์ใช้งานที่พบบ่อยๆ เช่น การขจัดสัญญาณรบกวนในสายไฟฟ้า การบันทึกคลื่นไฟฟ้าหัวใจ ECG การวัดการสั่นสะเทือนที่ไม่รุนแรง การวัดความถี่เสียงโดยใช้ไมโครโฟน และ งานอื่น ๆ ที่ใช้เซ็นเซอร์ในการเก็บข้อมูลอินพุต บางครั้งสัญญาณรบกวนสามารถลดลงได้ โดยการต่อสายดินและใช้สายเคเบิลที่มีการฉีลด์อย่างดี แต่ก็ไม่สามารถขจัดได้อย่างสมบูรณ์



รูปที่ 3.25 บล็อกไดอะแกรมของการขจัดสัญญาณรบกวนอะแดปทีฟในสายไฟฟ้า

รูปที่ 3.25 เป็นตัวอย่างของการขจัดสัญญาณรบกวนแบบอะแดปทีฟในสายไฟฟ้า สัญญาณอินพุตอ้างอิงมาจากสายไฟฟ้า และ อินพุตหลักได้มาจากเซ็นเซอร์ วงจรขจัดสัญญาณรบกวนอะแดปทีฟจะติดตามการเปลี่ยนแปลงทั้งเฟส และ ขนาดของสัญญาณไฟฟ้า โดยใช้สัมประสิทธิ์ตัวกรองเพียง สองค่าเท่านั้น



รูปที่ 3.26 สัญญาณอินพุตและเอาต์พุตของวงจรขจัดสัญญาณรบกวนแบบอะแดปทีฟ

รูปที่ 3.26 เป็นสัญญาณอินพุตและ เอาต์พุต ของรูปที่ 3.25 โดยใช้อัลกอริทึมแบบ LMS ทำการปรับค่าสัมประสิทธิ์สองค่าแบบลิเนียร์คอมไบเนอร์ อินพุตที่เข้าสู่ลิเนียร์คอมไบเนอร์ทั้งสองคือ สัญญาณรบกวนแบบตรง ๆ หนึ่งอินพุต และ สัญญาณรบกวนที่เลื่อนเฟสไป 90 องศา อีกหนึ่งอินพุต สัญญาณที่ปราศจากการรบกวน คือ สัญญาณค่าความคลาดเคลื่อน $e(n)$ ของตัวกรองลิเนียร์คอมไบเนอร์นั่นเอง เทคนิคเดียวกันนี้ได้มีการนำมาใช้งานในด้านต่าง ๆ เช่น การจัดการรบกวนคลื่นไฟฟ้าหัวใจของแม่กับลูก การจัดการรบกวนคลื่นไฟฟ้าหัวใจในการเปลี่ยนหัวใจใหม่ การจัดสัญญาณรบกวนออกจากเสียงพูด และการจัดสัญญาณรบกวนจากสายอากาศข้างเคียง เป็นต้น

บทที่ 4

การประมาณค่าในโครงสร้างแลตทิซ

ในบทนี้จะกล่าวถึงอัลกอริทึมพื้นฐานที่เป็นมาตรฐาน โครงสร้างแลตทิซ ที่ใช้ในการออกแบบ การจำลองระบบ

4.1 การจำลองระบบ(System identification)

การจำลองระบบ จะสัมพันธ์กับการกำหนดว่าระบบที่จำลอง หรือ เลียนแบบนั้น สามารถอธิบายรายละเอียดของกลุ่มข้อมูลได้ดีที่สุดเพียงใด เช่น กลุ่มลำดับข้อมูล $z(t), M \leq t \leq N$ ที่มีองค์ประกอบอยู่สองส่วน ดังสมการ

$$z(t) = [y(t), u(t)]^T \quad (4.1)$$

ส่วนแรก สัญญาณเอาต์พุต $y(t)$ ของระบบใด ๆ ที่ไม่ทราบว่าเป็นระบบอะไร สมมติว่าชื่อระบบ S โดยป้อนสัญญาณอินพุตเข้าไปในระบบ คือ $u(t)$ และสัญญาณรบกวน $\eta(t)$ สามารถอธิบายความสัมพันธ์นี้ได้ว่า

$$y = S(u, \eta) \quad (4.2)$$

ซึ่งเป็นเวกเตอร์ที่ใช้แสดงถึงจำนวนของตัวอย่างที่ต่อเนื่องกัน ในการจำลองระบบนี้จะ ต้องออกแบบตัวกรองประมาณ

$$\hat{y} = \hat{S}(z) \quad (4.3)$$

ซึ่งสามารถ สร้างเอาต์พุต \hat{y} ที่ปราศจากสัญญาณรบกวน หลังจากป้อนลำดับข้อมูลอินพุตนั้น เข้าสู่ระบบ \hat{S} การเลือกแบบจำลองระบบ \hat{S} ให้สามารถแปลความหมายข้อมูลสัญญาณอินพุต ได้เหมาะสมและถูกต้องเหมือนกับระบบ S สามารถวัดสัญญาณคลาดเคลื่อนได้จากสมการ

$$e(t) = y(t) - \hat{y}(t) \quad (4.4)$$

จะสังเกตได้ว่า แบบจำลอง \hat{S} พยายามสร้างข้อมูลขึ้นมาใหม่และเอาสัญญาณรบกวนออกไป การสร้างแบบจำลองนี้ โดยทั่วไปจะเลือกตัวกรองประมาณแบบความแปรปรวนน้อย (Minimum variance estimator) ซึ่งมาจากการทำให้ ค่าความคลาดเคลื่อนกำลังสองเฉลี่ยที่น้อยที่สุด

$$E[(y(t) - g(Z_{t-1}))^2]$$

เมื่อ

$$Z_{t-1} = [z(0), z(1), \dots, z(t-1)]^T \quad (4.5)$$

ซึ่งก็คือ เวกเตอร์ตัวอย่างในอดีต และ ต้องทำให้ g ซึ่งเป็นฟังก์ชันนอนลิเนียร์มีค่าน้อยที่สุด $E[.]$ คือ ค่าความคาดหวัง การแก้ปัญหานี้ต้องมีเงื่อนไขว่า

$$\hat{y}(t) = E(y(t)|Z_{t-1}) \quad (4.6)$$

จะเรียกตัวกรองประมาณนี้ว่า ตัวกรองคาดคะเนแบบความแปรปรวนน้อย (Minimum variance predictor)

วิธีการประมาณค่าที่สำคัญ ๆ มักจะถูกเรียกรวม ๆ ว่า วิธีการแบบลำดับ หรือ วิธีการแบบออนไลน์ เป็นเทคนิคที่รวมเอาทั้งการประมาณค่าข้อมูลที่เรียงลำดับกันจำนวนมากและการประมาณค่าระบบของตัวมันเองรวมเข้าด้วยกัน ถึงแม้ว่าจะเป็นรูปแบบที่ดี แต่ตัวกรองคาดคะเนความแปรปรวนน้อยก็ไม่สามารถใช้เทคนิคการคำนวณค่าเฉลี่ยข้อมูลดิบจากพื้นฐานการประมาณ ฟังก์ชันความหนาแน่นความน่าจะเป็นได้ง่าย ๆ โดยเฉพาะอย่างยิ่งวิธีการแบบออนไลน์ จะวางแนวทางความหมาย ของการจำลองระบบการทำงานแบบเรียลไทม์ จากสมมุติฐานตัวกรองคาดคะเนดังนี้

$$\hat{S}(z) = \hat{F}(z, \theta) \quad (4.7)$$

ซึ่ง \hat{F} คือ ฟังก์ชันถ่ายโอนของระบบที่ทราบค่า และ θ คือ พารามิเตอร์ที่มีขอบเขตจำกัด กำหนดได้จาก

$$\theta \in \Theta \subset R^p$$

สมการที่ (4.7) ยังสามารถใช้ได้กับระบบอนาลอกที่มีขอบเขตจำกัด ซึ่งไม่ทราบค่า จากสมมุติฐานตามสมการที่ (4.2) ดังนี้

$$y = S(u, \eta) = F(u, \eta, \theta)$$

หลังจากที่ตัวกรองประมาณ \hat{S} สามารถจำลองระบบได้ทั้งหมดแล้ว จะเหลือเพียงการหาค่า θ ซึ่งจะเรียกว่า การจำลองระบบแบบพารามตริก การหาค่า θ สามารถหาได้จากการวัดค่าสัญญาณความคลาดเคลื่อนน้อยที่สุด ดังสมการ

$$e(t|\theta) = y(t) - \hat{y}(t|\theta) \quad (4.8)$$

จากความสัมพันธ์ของ θ เงื่อนไขสมรรถนะที่นิยมใช้กัน คือ ฟังก์ชันควอดราติก (Quadratic function)

$$V_N(\theta) = \sum_{t=M+P-1}^N e^2(t|\theta) \quad (4.9)$$

ข้อมูลที่อยู่นอกขอบเขต $[M, N]$ จะไม่มีผลต่อการคำนวณ จากสมมุติฐานว่า สัญญาณอินพุตและเอาต์พุตเป็น คอซอล กล่าวคือ $u(t) = 0, y(t) = 0$, เมื่อ $t < 0$ ในกรณีของ prewindowed และการลดค่าใช้จ่ายในการออกแบบให้น้อยที่สุด จะได้ว่า

$$V_N(\theta) = \sum_{t=0}^N e^2(t|\theta) \quad (4.10)$$

บ่อยครั้งที่มักจะใช้ในรูปแบบความสัมพันธ์กับ ค่าถ่วงน้ำหนักต่าง ๆ ที่มีการเปลี่ยนแปลงตามเวลา เช่น

$$V_N(\theta) = \sum_{t=M+P-1}^N w_N(t) e^2(t|\theta) \quad (4.11)$$

ฟังก์ชันหน้าต่าง (Window function) $w_N(t)$ จะเป็นตัวกำหนด ค่าของสัญญาณความคลาดเคลื่อนที่เปลี่ยนแปลงตามเวลา ซึ่งจะสะท้อนกลับมาในรูปแบบข้อมูลในอดีตและปัจจุบัน ดังนั้นจึง

ได้รับความสนใจในการจำลองระบบที่มีการเปลี่ยนแปลงตามเวลาแบบไม่คงที่ (Time-varying dynamics) วิธีการควบคุมจึงขึ้นอยู่กับความสามารถในการปรับตัว (Adapting) และการติดตาม (Tracking) ระบบหรือสัญญาณที่มีการเปลี่ยนแปลงแบบไม่คงที่ ด้วยเหตุนี้จึงเรียกว่า อะแดปทีฟ อัลกอริทึม

คุณสมบัติที่น่าสนใจของแลตทิซอัลกอริทึม คือ อันดับของการทำซ้ำ (Order recursiveness) จากนิยามที่ว่า อันดับของตัวกรองประมาณที่ได้จากเงื่อนไขบางประการที่ดีที่สุดมีค่าเท่ากับขนาดมิติของเวกเตอร์ข้อมูลที่ใช้ประมาณนั้น แลตทิซอัลกอริทึมจะคำนวณความคลาดเคลื่อนโดยประมาณตั้งแต่อันดับ 1 ถึง p ซึ่ง p เป็นอันดับสูงสุดของตัวกรองประมาณ ตัวกรองประมาณแบบ MMSE (Minimums mean square error) สามารถใช้อธิบายคุณลักษณะของตัวกรองวินเนอร์ได้ เนื่องมาจากสมการพื้นฐานการประมาณแบบ MMSE ได้มาจากคุณสมบัติของสัญญาณอินพุตตัวมันเอง

4.2 การประมาณค่าแบบ MMSE (MMSE Estimation)

พิจารณาลำดับสัญญาณที่ต้องการ $\{y(k)\}$ และลำดับของเวกเตอร์ข้อมูลที่มีขนาดมิติ $\{u_m(k) \equiv (u_1(k)u_2(k)\dots u_m(k))^T, k = -\infty, \dots, k, \dots, \infty\}$ ซึ่ง $y(k)$ และ $u_i(k), i = 1, 2, \dots, m$ เป็นตัวอย่างเชิงซ้อน เนื่องมาจากการประมวลผลร่วมกันระหว่าง Stochastic และ Stationary ทำให้ได้ตัวแปรสุ่มเชิงซ้อน ในการประมาณค่าแบบ MMSE ต้องการแค่ stationary ร่วมอันดับสองก็เพียงพอแล้ว จะได้ว่า $E[y^*(k)y(k-\tau)], E[y^*(k)u_i(k-\tau)]$ และ $E[u_i^*(k)u_j(k-\tau)]$ เป็นฟังก์ชันของ τ และไม่ขึ้นกับค่า k ใดๆ ตัวกระทำความคาดหวัง $E[\cdot]$ ก็คือการหาค่าเฉลี่ยของชุดข้อมูลที่อยู่ภายใน

4.2.1 ความสัมพันธ์การประมาณค่าแบบ MMSE ต่อสัญญาณคงที่

การประมาณค่าแบบ MMSE โดยทั่วไป เริ่มจากค่าประมาณ $\hat{y}(k)$ ที่ได้จากการลิเนียร์คอมไบเนชันของ $u_i(k), i = 1, 2, \dots, m$, จากสมการ

$$\hat{y}(k) = -\sum_{i=1}^m c_i u_i(k) \equiv -c_m^T u_m(k)$$

เมื่อ c_m เป็นเวกเตอร์สัมประสิทธิ์ที่มีขนาดมิติ m จุดประสงค์ของการประมาณค่าแบบ MMSE คือ การหาค่าเวกเตอร์สัมประสิทธิ์ที่เหมาะสม ที่จะทำให้ค่าความคาดหวังของสัญญาณความคลาดเคลื่อนยกกำลังสอง $|e(k)|^2$ ระหว่าง $\hat{y}(k)$ และ $y(k)$ มีค่าน้อยที่สุด หาได้จากสมการ

$$\sigma_e^2 = E[|e(k)|^2] \equiv E[|y(k) - \hat{y}(k)|^2] = E[|y(k) + c_m^T u_m(k)|^2] \quad (4.12)$$

เวกเตอร์สัมประสิทธิ์ที่เหมาะสมนี้ จะเรียกว่า $c_{m,MMSE}$ ซึ่งก็คือ

$$c_{m,MMSE} = -R_{u_m u_m}^{-1} d_m^{(y)} \quad (4.13)$$

ซึ่ง $R_{u_m u_m}$ คือ เมตริกซ์อัตโนมัติสหสัมพันธ์ของ $u_m(k)$ ได้มาจาก

$$R_{u_m u_m} = E[u_m^*(k) u_m^T(k)] \quad (4.14)$$

และ $d_m^{(y)}$ คือ เวกเตอร์ความสัมพัทธ์ไขว้ระหว่าง $u_m(k)$ และ $y(k)$ ได้จากสมการ

$$d_m^{(y)} = E[u_m^*(k) y(k)]$$

ค่าประมาณแบบ MMSE ที่ลำดับ m ของ $y(k)$ กับเวกเตอร์ข้อมูล $u_m(k)$ ใช้สัญลักษณ์ เป็น $\hat{y}_{m,MMSE}(k)$ ได้มาจาก

$$\hat{y}_{m,MMSE}(k) = -c_{m,MMSE}^T u_m(k) \quad (4.15)$$

ในทำนองเดียวกัน กำหนดให้ $e_{m,MMSE}^{(y)}(k)$ เป็นค่าความคลาดเคลื่อนระหว่าง $y(k)$ และ $\hat{y}_{m,MMSE}(k)$ ซึ่งเป็นค่าความคลาดเคลื่อนโดยประมาณที่อันดับ m ของ $y(k)$ กับเวกเตอร์ข้อมูล $u_m(k)$ นั่นคือ

$$e_{m,MMSE}(k) \equiv y(k) - \hat{y}_{m,MMSE}(k) = y(k) + c_{m,MMSE}^T u_m(k) \quad (4.16)$$

การประมาณค่าแบบ MMSE คือ การหาค่าประมาณที่เหมาะสมด้วยฟังก์ชันควอดราติก จากสมการที่ (4.13) ซึ่งสามารถนำไปประยุกต์ใช้งานกับการแก้ปัญหา การประมาณค่าที่เหมาะสมอื่น ๆ ด้วยฟังก์ชันควอดราติกใด ๆ

4.2.2 ความเป็นออร์ทอโกนอล (Orthogonality principle)

จากสมการที่ (4.16) ถ้าคูณทั้งสองข้างด้วย $u_m^*(k)$ แล้วใส่ตัวกระทำความคาดหวัง จากสมการที่ (4.13) และ (4.14) จะได้

$$\begin{aligned} E[u_m^*(k)e_{m,MMSE}^{(y)}(k)] &= E[u_m^*(k)y(k)] + E[u_m^*(k)u_m^T(k)]c_{m,MMSE} \\ &= d_m^{(y)} - R_{u_m u_m}^{-1} R_{u_m u_m}^{-1} d_m^{(y)} = 0 \end{aligned} \quad (4.17)$$

ซึ่งใช้ความสัมพันธ์ $u_m^T(k)c_{m,MMSE} = c_{m,MMSE}^T u_m(k)$ สมการที่ (4.17) เรียกว่า เป็นสมการออร์ทอโกนอลพื้นฐานสำหรับการประมาณแบบ MMSE ในที่นี้ จะกล่าวได้ว่า ค่าความคลาดเคลื่อนจากการประมาณแบบ MMSE เป็น ออร์ทอโกนอล กับ ทุกองค์ประกอบของเวกเตอร์ข้อมูล

4.2.3 อัตสหสัมพันธ์ของความคลาดเคลื่อน (Error autocorrelations)

จากสมการที่ (4.12) สามารถสรุป ความคลาดเคลื่อนยกกำลังสองเฉลี่ยน้อยที่สุด หรือ อัตสหสัมพันธ์ของความคลาดเคลื่อน $e_{m,MMSE}^{(y)}(k)$ ในรูปของ $\alpha_m^{(yy)}$ ได้ดังนี้

$$\alpha_m^{(yy)} = \sigma_{m,MMSE}^{2(y)} = E[|e_{m,MMSE}^{(y)}(k)|^2] = E[e_{m,MMSE}^{(y)*}(k)\{y(k) + c_m^T u_m(k)\}]$$

และจากทฤษฎีออร์ทอโกนอลพื้นฐานจะได้ว่า

$$\begin{aligned} \alpha_m^{(yy)} &= E[e_{m,MMSE}^{(y)*}(k)y(k)] = E[|y(k)|^2] + E[c_m^{*T} u_m^*(k)y(k)] \\ &= \rho_0^y + c_{m,MMSE}^H d_m^{(y)} = \rho_0^y - d_m^{(y)H} R_{u_m u_m}^{-1} d_m^{(y)} \end{aligned} \quad (4.18)$$

H คือ คอนจูเกตเชิงซ้อน และการกระทำทรานสโพส หรือ Hermitian ของเวกเตอร์หรือเมตริกซ์ และ $\rho_0^y \equiv E[|y(k)|^2]$ คือ อัตสหสัมพันธ์ หรือ ความแปรปรวนของ $y(k)$

4.3 การประมาณแบบ MMSE กับ อันดับที่ซ้ำกัน (Order recursive)

เป็นวิธีการประมาณค่าแบบ MMSE รูปแบบแลดทึซ ซึ่งเป็นกรณีพิเศษที่ใช้ในการประมาณค่าความคลาดเคลื่อนที่มีอันดับที่มากกว่า เริ่มจากการพิจารณาความสัมพันธ์ทั่วไปของอันดับการป้อนกลับแบบปกติ ด้วยทฤษฎีที่ 1 โดยการแบ่งขนาดมิติของเวกเตอร์ข้อมูล $u_{m+1}(k)$ เป็นสองส่วน คือ

$$u_{m+1}(k) = (u_m^T(k), u'(k))^T$$

ค่าความคลาดเคลื่อนจากการประมาณแบบ MMSE ของเอาต์พุต $y(k)$ ที่อันดับ $(m+1)$ ที่ได้จากเวกเตอร์ข้อมูลอินพุต $u_{m+1}(k)$ ซึ่งก็คือ $e_{m+1,MMSE}^{(y)}(k)$ สามารถคำนวณได้จาก

$$e_{m+1,MMSE}^{(y)}(k) = e_{m,MMSE}^{(y)}(k) - \left[\frac{\alpha_m^{(yu')}}{\alpha_m^{(u'u)}} \right] e_{m,MMSE}^{(u')}(k) \quad (4.19)$$

เมื่อ $e_{m,MMSE}^{(y)}(k)$ และ $e_{m,MMSE}^{(u')}(k)$ คือ ค่าความคลาดเคลื่อนที่อันดับ m จากการประมาณแบบ MMSE ของเอาต์พุต $y(k)$ และ $u'(k)$ ซึ่งมาจากเวกเตอร์ข้อมูล $u_m(k)$ ตามลำดับ และ $\alpha_m^{(u'u)}$ คือ อัตราสหสัมพันธ์ของ $e_{m,MMSE}^{(u')}(k)$ หาได้จาก

$$\alpha_m^{(u'u)} \equiv E \left[|e_{m,MMSE}^{(u')}(k)|^2 \right]$$

และ $\alpha_m^{(yu')}$ คือ ความสัมพันธ์ไขว้ระหว่าง $e_{m,MMSE}^{(y)}(k)$ และ $e_{m,MMSE}^{(u')}(k)$ กำหนดได้ดังนี้

$$\alpha_m^{(yu')} \equiv E \left[e_{m,MMSE}^{(y)}(k) e_{m,MMSE}^{(u')*}(k) \right] \quad (4.20)$$

จากทฤษฎีที่ 1 แสดงให้เห็นได้ว่า สามารถหาค่าความคลาดเคลื่อนจากการประมาณแบบ MMSE ที่มีอันดับสูงกว่า จากค่าความคลาดเคลื่อนจากการประมาณแบบ MMSE ที่อันดับต่ำกว่าได้ ดังนั้นจึงเรียกว่า ความสัมพันธ์ของอันดับที่ซ้ำกันในการประมาณแบบ MMSE

4.4 การประมาณค่าแบบ MMSE สำหรับโครงสร้างแบบแลตทิซ

พิจารณาการประมาณค่าแบบพิเศษ ซึ่งเวกเตอร์ข้อมูล $u_m(k)$ เป็นส่วนหนึ่งของข้อมูลอนุกรมทางเวลา $\{u_m(k), k = -\infty, 0, \dots, k, \dots, \infty\}$ นั่นคือ

$$\{u_m(k) = (u(k), u(k-1), \dots, u(k-m+1))^T$$

จะแสดงให้เห็นว่า รูปแบบแลตทิซเป็นโครงสร้างมาตรฐานของ การประมาณค่า MMSE ในรูปแบบ Order recursive กับข้อมูลเวกเตอร์ต่าง ๆ เพื่อให้เข้าใจถึงวิธีการประมาณค่าแลตทิซแบบ MMSE จะเริ่มจากการพิจารณาการประมาณค่าที่เรียกว่า การคาดคะเนแบบลิเนียร์ (Linear prediction)

4.4.1 การคาดคะเนเดิหน้าและย้อนกลับแบบลิเนียร์

การคาดคะเนแบบลิเนียร์ เป็นการแก้ปัญหาที่สำคัญมากแบบหนึ่งในการวิเคราะห์อนุกรมทางเวลา และใช้งานอย่างกว้างขวางในการเข้ารหัสสัญญาณเสียงพูดและการวิเคราะห์แถบพลังงานที่มีความละเอียดสูง จึงมีความสำคัญต่อการพัฒนาวิธีการประมาณค่าแลตทิซแบบ MMSE รวมทั้งการประยุกต์ใช้งานในทางปฏิบัติอื่น ๆ การคาดคะเนแบบลิเนียร์มีสองรูปแบบ คือ แบบคาดคะเนเดิหน้า และแบบคาดคะเนย้อนกลับ

การคาดคะเนเดิหน้าที่อันดับ m จะจัดรูปแบบการประมาณค่าแบบลิเนียร์ของตัวอย่างอินพุต เดียว $u(k)$ ที่อยู่ในช่วงเวลาหนึ่ง $\{u(k)\}$ ซึ่งจะขึ้นอยู่กัตัวอย่างก่อนหน้าถึงลำดับที่ m การคาดคะเนเดิหน้าแบบลิเนียร์ของ MMSE จะต้องหาเวกเตอร์สัมประสิทธิ์ที่จะทำให้ค่าความคาดหวังของความคลาดเคลื่อนคาดคะเนเดิหน้า ที่ลำดับ m ยกกำลังสองมีค่าน้อยที่สุด หาได้จากสมการ

$$\sigma_m^{(f)2} = E[|e_m^f(k)|^2] \equiv E[|u(k) - \hat{u}_m(k)|^2]$$

เมื่อ

$$\hat{u}_m^f(k) = -a_m^T u_m(k-1)$$

และ a_m คือ เวกเตอร์สัมประสิทธิ์คาดคะเนเดิหน้า

จากการประมาณแบบ MMSE ปกติทั่วไปที่อธิบายในหัวข้อที่ผ่านมา ก็จะเหมือนกับกรณีนี้ จะได้ว่า เวกเตอร์สัมประสิทธิ์ของการคาดคะเนเดิหน้าแบบ MMSE ที่เหมาะสม $a_{m,MMSE}$ เป็นไปตามสมการ

$$a_{m,MMSE} = -R_{u_m u_m}^{-1} d_m^f$$

เมื่อ

$$R_{u_m u_m}^T = E[u_m^*(k-1)u_m^T(k-1)] = E[u_m^* u_m^T]$$

เช่นเดียวกับ สมการที่ (4.14) และ d_m^f คือ เวกเตอร์ที่มีความสัมพันธ์ระหว่าง $u(k)$ และ $u_m(k-1)$ กำหนดเป็น

$$d_m^f = E[u(k)u_m^*(k-1)] = (\rho_1, \rho_2, \dots, \rho_m)^T \quad (4.21)$$

ซึ่ง $\rho_i \equiv E[u(k)u^*(k-i)]$ ดังนั้น ค่าความคลาดเคลื่อนคาดคะเนเดิหน้าแบบ MMSE ที่อันดับ m จะได้ ดังนี้

$$e_{m,MMSE}^f(k) = u(k) - \hat{u}_{m,MMSE}^f(k) \equiv u(k) + a_{m,MMSE}^T u_m(k-1)$$

ส่วนการคาดคะเนแบบลิเนียร์ย้อนกลับที่อันดับ m จะใช้ การคอมไบเนชันแบบลิเนียร์ของ ตัวอย่างอินพุตหลาย ๆ ตัวที่เรียงกันถึงอันดับ m $u(k), u(k-1), \dots, u(k-m+1)$, เป็นรูปแบบในการประมาณตัวอย่างที่มาก่อนหน้าแล้ว นั่นคือ $u(k-m)$ ในการคาดคะเนย้อนกลับแบบ MMSE จะหาเวกเตอร์สัมประสิทธิ์ที่คาดคะเนย้อนกลับ $b_{m,MMSE}$ ที่ทำให้ค่าความคาดหวังของความคลาดเคลื่อนคาดคะเนย้อนกลับ ที่ลำดับ m ยกกำลังสองมีค่าน้อยที่สุด หาได้จากสมการ

$$\sigma_m^{(b)2} = E[|e_m^b(k)|^2] \equiv E[|u(k-m) - \hat{u}_m^b(k-m)|^2] \equiv E[|u(k-m) + b_m^T u_m(k)|^2]$$

เวกเตอร์สัมประสิทธิ์การคาดคะเนย้อนกลับแบบ MMSE เป็นไปตามสมการ

$$b_{m,MMSE} = -R_{u_m u_m}^{-1} d_m^b \quad (4.22)$$

เมื่อ d_m^b คือ เวกเตอร์ที่มีความสัมพันธ์ระหว่าง $u(k-m)$ และ $u_m(k)$ กำหนดเป็น

$$d_m^b = E[u(k-m)u_m^*(k)] = (\rho_m^*, \rho_{m-1}^*, \dots, \rho_1^*)^T \quad (4.23)$$

เทอมหลังสุดของสมการที่ (4.23) ได้มาจากความสัมพันธ์

$$E[u(k-m)u^*(k-i)] = E[u(k-m+i)u^*(k)] = E\left[\{u(k)u^*(k-m+i)\}^*\right] \equiv \rho_{m-i}^*$$

ดังนั้น ค่าความคลาดเคลื่อนค่าคคะเนย้อนกลับแบบ MMSE ที่อันดับ m จะได้ ดังนี้

$$e_{m,MMSE}^b(k) = u(k-m) - \hat{u}_{m,MMSE}^b(k) = u(k-m) + b_{m,MMSE}^T u_m(k)$$

จากสมการที่ (4.21) และ สมการ (4.23) จะได้ว่า

$$d_m^b = Jd_m^{f*} \quad (4.24)$$

เมื่อ J คือ เมทริกซ์แลกเปลี่ยน กำหนดจาก

$$J = \begin{pmatrix} 0 & \dots & 0 & 1 \\ 0 & \dots & 1 & 0 \\ \vdots & & \vdots & \vdots \\ 1 & \dots & 0 & 0 \end{pmatrix}$$

จะเห็นว่า $JJ = I$ จากนิยามของ $R_{u_m u_m}$ สามารถแสดงได้ว่า

$$R_{u_m u_m} = [JR_{u_m u_m} J]^* \quad (4.25)$$

แทนค่าสมการที่ (4.25) และ (4.24) ลงในสมการที่ (4.22) จะได้

$$b_{m,MMSE} = -J[R_{u_m u_m}^*]^{-1} Jd_m^{f*} = -J[R_{u_m u_m}^{-1} d_m^f]^* = Ja_{m,MMSE}^*$$

จะเห็นว่า เวกเตอร์สัมประสิทธิ์การคาดคะเนย้อนกลับแบบ MMSE มีค่าเท่ากับ คอนจูเกตเชิงซ้อนของเวกเตอร์สัมประสิทธิ์การคาดคะเนเดิหน้าแบบ MMSE ที่มีอันดับย้อนกลับกัน

4.4.2 การคาดคะเนอันดับที่ซ้ำกันแบบลิเนียร์โครงสร้างแลคทิกซ์

พิจารณา การคาดคะเนเดิหน้าแบบลิเนียร์ MMSE ที่อันดับ $(m+1)$ โดยใช้ความสัมพันธ์ทั่วไปของอันดับที่ซ้ำกันในการประมาณแบบ MMSE ด้วยทฤษฎีที่ 1 สำหรับการคาดคะเนเดิหน้า ที่อันดับ $(m+1)$ โดยการแยกส่วนที่อยู่สูงสุดของเวกเตอร์ข้อมูล $u_{m+1}(k-1)$ ของการคาดคะเนเดิหน้าได้ดังนี้

$$u_{m+1}(k-1) = \begin{pmatrix} u_m(k-1) \\ u(k-m-1) \end{pmatrix} \quad (4.26)$$

จากทฤษฎีที่ 1 สมการที่ (4.26) ความคลาดเคลื่อนคาดคะเนเดิหน้าแบบ MMSE ที่อันดับ $(m+1)$ $e_{m+1,MMSE}^f(k)$ สามารถคำนวณได้จาก ค่าความคลาดเคลื่อนแบบ MMSE สองชุด คือ ค่าความคลาดเคลื่อนแบบ MMSE ของ $u(k)$ และ $u(k-m-1)$ ที่มาจากเวกเตอร์ข้อมูลเดียวกัน $u_m(k-1)$ จากทฤษฎีนี้ จะเห็นว่า ค่าความคลาดเคลื่อนส่วนแรก มีสมการคล้ายกับ $\{e_{m,MMSE}^{(v)}(k)\}$ ในสมการที่ (4.19) ซึ่งก็คือค่าความคลาดเคลื่อนแบบ MMSE ของ $u(k)$ ที่ขึ้นอยู่กับ $u_m(k-1)$ จากนิยามนี้ จะเห็นได้ว่า คือ ค่าความคลาดเคลื่อนคาดคะเนเดิหน้าแบบ MMSE ที่อันดับ m $e_{m,MMSE}^f(k)$ ส่วนค่าความคลาดเคลื่อนในส่วนที่สอง มีสมการคล้ายกับ $e_m^{(u)}(k)$ ในสมการที่ (4.19) ซึ่งก็คือ ค่าความคลาดเคลื่อนแบบ MMSE ของ $u(k-m-1)$ ที่ขึ้นอยู่กับ $u_m(k-1)$ จากนิยามนี้ จะเห็นได้ว่า คือ ค่าความคลาดเคลื่อนคาดคะเนย้อนกลับแบบ MMSE ที่อันดับ m $e_{m,MMSE}^b(k-1)$ ดังนั้น อันดับที่ซ้ำกันของความคลาดเคลื่อนคาดคะเนเดิหน้า สรุปได้ดังสมการ

$$e_{m+1,MMSE}^f(k) = e_{m,MMSE}^f(k) - \left[\frac{\beta_m^f}{\alpha_m^b} \right] e_{m,MMSE}^b(k-1) \equiv e_{m,MMSE}^f(k) + k_m^f e_{m,MMSE}^b(k-1) \quad (4.27)$$

เมื่อ

$$\beta_m^f = E \left[e_{m,MMSE}^f(k) e_{m,MMSE}^{b*}(k-1) \right] \quad (4.28)$$

และ

$$\alpha_m^b = E \left[\left| e_{m,MMSE}^b(k) \right|^2 \right] = E \left[\left| e_{m,MMSE}^b(k-1) \right|^2 \right] \quad (4.29)$$

ในการทำงานเดียวกัน ก็สามารถอธิบายอันดับที่ซ้ำกันในการประมาณแบบ MMSE ด้วยทฤษฎีที่ 1 สำหรับการคาดคะเนย้อนกลับได้เหมือนกัน สัญญาณที่ต้องการจากการคาดคะเนย้อนกลับที่อันดับ $(m+1)$ ก็คือ $u(k-m-1)$ และเวกเตอร์ข้อมูลของ $u_{m+1}(k)$ ใช้ส่วนต่ำสุดของเวกเตอร์ข้อมูล จะได้ว่า

$$u_{m+1}(k) = \begin{pmatrix} u_m(k) \\ u_m(k-1) \end{pmatrix} \quad (4.30)$$

จากทฤษฎีบทที่ 1 สามารถคำนวณค่าความคลาดเคลื่อนคาดคะเนย้อนกลับแบบ MMSE ที่อันดับ $(m+1)$ $e_{m+1,MMSE}^b(k)$ จากค่าความคลาดเคลื่อนแบบ MMSE ของ $u(k-m-1)$ และ $u(k)$ ที่มาจากเวกเตอร์ข้อมูลเดียวกัน $u_m(k-1)$ เหมือนกับการพิจารณาจากที่ผ่านมาแล้ว จะได้ว่า

$$e_{m+1,MMSE}^b(k) = e_{m,MMSE}^b(k-1) - \left[\frac{\beta_m^b}{\alpha_m^f} \right] e_{m,MMSE}^f(k) \equiv e_{m,MMSE}^b(k-1) + k_m^b e_{m,MMSE}^f(k) \quad (4.31)$$

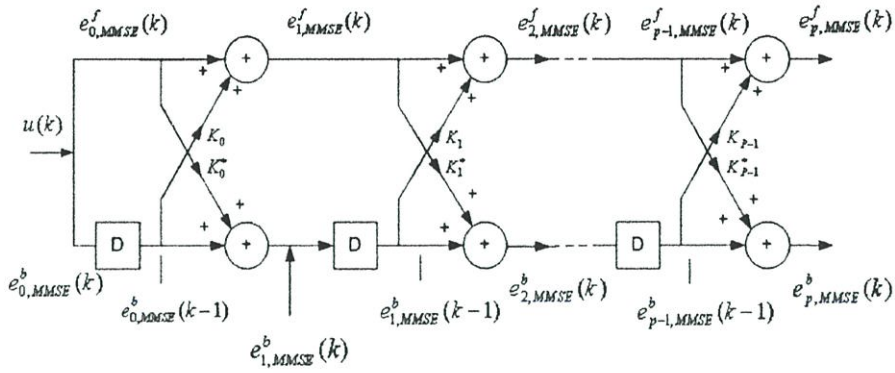
เมื่อ

$$\beta_m^b = E \left[e_{m,MMSE}^{f*}(k) e_{m,MMSE}^b(k-1) \right] \quad (4.32)$$

และ

$$\alpha_m^f = E \left[\left| e_{m,MMSE}^f(k) \right|^2 \right]$$

จะเห็นได้ว่า ค่าความคลาดเคลื่อนของการคาดคะเนทั้งเดินหน้าและย้อนกลับที่อันดับ $(m+1)$ สามารถคำนวณได้จาก การประมาณค่าความคลาดเคลื่อนที่เป็น สเกลลาร์ของ MMSE ได้ โดยการใช้ ค่าความคลาดเคลื่อนของการคาดคะเนแบบ MMSE ทั้งเดินหน้าและย้อนกลับที่อันดับต่ำกว่าได้ เมื่อ $m \geq 1$ และ เมื่อ $m = 0$ สามารถใช้ค่าจากการประมาณแบบ MMSE ที่เป็น ค่าสเกลลาร์ตรง ๆ ในการคำนวณความคลาดเคลื่อนคาดคะเนอันดับหนึ่ง เนื่องจากสัญญาณที่ต้องการและเวกเตอร์ข้อมูล $\{u(k)\}$ และ $\{u(k-1)\}$ สำหรับการคาดคะเนเดินหน้า และ $\{u(k)\}$ และ $\{u(k-1)\}$ สำหรับการคาดคะเนย้อนกลับ ดังรูปที่ 4.1



รูปที่ 4.1 โครงสร้างแลตทิซสำหรับการคาดคะเน MMSE แบบลิเนียร์

จะได้

$$e_0^f(k) \equiv e_0^b(k) \equiv u(k) \quad (4.33)$$

จากการพิจารณาที่ผ่านมา สรุปได้ว่า การคาดคะเนเดิหน้าและย้อนกลับที่อันดับ p^{th} สามารถใช้ในการออกแบบในรูปแบบของการทำซ้ำอันดับ (Order recursive) จากอนุกรมของการคาดคะเนเดิหน้าและย้อนกลับแบบซ้ำอันดับ ที่มีอันดับ $m=0$ ถึง $p-1$ โครงสร้างของวิธีการคาดคะเนแบบนี้ แสดงได้ในรูปที่ 4.1 ซึ่งจะเรียกว่า รูปแบบแลตทิซ (Lattice form) หรือ โครงสร้างแบบแลตทิซ (Lattice structure) ตัวกรองประมาณที่ใช้ในการออกแบบจะเรียกว่า ตัวกรองประมาณแบบแลตทิซ (Lattice predictor) ซึ่งจะสร้างสัญญาณคลาดเคลื่อนคาดคะเนทั้งเดิหน้าและย้อนกลับ ของอันดับต่าง ๆ จาก 1 ถึง p

จากที่ได้อธิบายมาแล้วว่า สัมประสิทธิ์การประมาณเป็นค่าแบบสเกลลาร์ $k_m^f \equiv -\beta_m^f / \alpha_m^b$ ในสมการที่ (4.27) และ $k_m^b \equiv -\beta_m^b / \alpha_m^f$ ในสมการที่ (4.31) จะเรียกว่า สัมประสิทธิ์ของการสะท้อนเดิหน้าและสัมประสิทธิ์ของการสะท้อนย้อนกลับ ตามลำดับ

จากสมการที่ (4.18) เมื่อพิสูจน์ต่อไปจะได้ว่า

$$\alpha_m^f = \rho_0^u + a_{m,MMSE}^H d_m^f = \rho_0^u - d_m^{fH} R_{u_m u_m}^{-1} d_m^f \quad (4.34)$$

และ

$$\alpha_m^b = \rho_0^u + b_{m,MMSE}^H d_m^b = \rho_0^u - d_m^{bH} R_{u_m u_m}^{-1} d_m^b \quad (4.35)$$

จากที่ได้อธิบายไว้แล้วว่า

$$\rho_0^u \equiv E[|u(k)|^2] = [u(k-m)^2]$$

เนื่องจาก $u(k)$ เป็น Stationary จะได้ว่า α_m^f และ α_m^b เป็นค่าจริง แทนค่าสมการที่ (4.24) และ (4.25) ลงในสมการที่ (4.35) และเปรียบเทียบกับสมการที่ (4.34) จะได้ $\alpha_m^b = \alpha_m^f$ ดังนั้นสรุปได้ว่า

$$\alpha_m \equiv \alpha_m^b = \alpha_m^f$$

เมื่อเปรียบเทียบสมการที่ (4.32) กับสมการที่ (4.28) จะเห็นว่า β_m^b มีค่าเท่ากับคอนจูเกตเชิงซ้อนของ β_m^f ดังนี้

$$\beta_m^b = \beta_m^{f*}$$

และจะได้ว่า $k_m^b = k_m^{f*}$ จึงสรุปได้ว่า

$$k_m \equiv k_m^b = k_m^{f*}$$

4.4.3 การประมาณค่าร่วมของ MMSE โครงสร้างแลตทิซ

จากหัวข้อที่ผ่านมาทราบแล้วว่า การประมาณค่าแบบ MMSE ของสัญญาณที่ต้องการ $y(k)$ ขึ้นอยู่กับเวกเตอร์ข้อมูล $u_m(k)$ จึงเรียกว่า การประมาณค่าร่วม (Joint-process estimation) เนื่องจาก $\{y(k)\}$ เกิดจากกระบวนการร่วมของ $\{u(k)\}$ ในลักษณะเดียวกัน การประมาณค่าความคลาดเคลื่อนก็จะเรียกว่า การประมาณค่าความคลาดเคลื่อนร่วม (Joint estimation error)

จากทฤษฎีที่ 1 และการแยกข้อมูลด้านบนสุดของ $u_{m+1}(k)$ การประมาณค่าร่วมแบบ MMSE ที่อันดับ $m+1$ $e_{m+1,MMSE}^{(y)}(k)$ สามารถคำนวณได้โดยใช้การประมาณค่าความคลาดเคลื่อนร่วมจากการประมาณค่าสเกลลาร์แบบ MMSE เพราะเป็นการประมาณค่าความคลาดเคลื่อนของ $y(k)$ และ $u(k-m)$ จากเวกเตอร์ข้อมูล $u_m(k)$ โดยการกำหนดการประมาณค่าร่วมแบบ MMSE ที่อันดับ m $e_{m,MMSE}^{(y)}(k)$ และการคาดคะเนความคลาดเคลื่อนย้อนกลับแบบ MMSE $e_{m,MMSE}^b(k)$ ตามลำดับ การทำซ้ำอันดับของการประมาณค่าความคลาดเคลื่อนร่วม ได้จาก

$$e_{m+1,MMSE}^{(y)}(k) = e_{m,MMSE}^{(y)}(k) - \left[\beta_m^{(y)} / \alpha_m \right] e_{m,MMSE}^b(k) \quad (4.36)$$

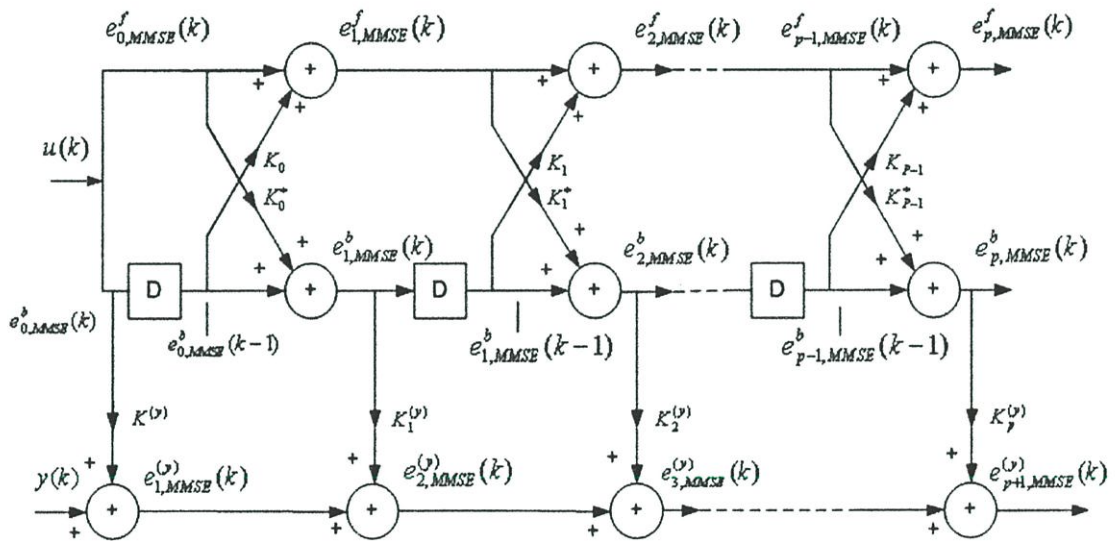
เมื่อ $\beta_m^{(y)}$ คือ ความสัมพันธ์ไขว้ระหว่าง $e_{m,MMSE}^{(y)}(k)$ และ $e_{m,MMSE}^b(k)$ จากสมการ

$$\beta_m^{(y)} = E[e_{m,MMSE}^{(y)}(k)e_{m,MMSE}^{b*}(k)] \quad (4.37)$$

และ $\alpha_m \equiv \alpha_m^b$ ในสมการที่ (4.29) จะคล้ายกับการสะท้อนของสัมประสิทธิ์เดิหน้าและย้อนกลับ ดังนั้น จึงกำหนด ค่าประมาณสัมประสิทธิ์ร่วม $k_m^{(y)}$ ได้เป็น

$$k_m^{(y)} = -\beta_m^{(y)} / \alpha_m \quad (4.38)$$

จาก สมการที่ (4.33) สามารถกำหนดให้ $e_0^{(y)}(k) \equiv y(k)$ ดังนั้นจะเห็นได้ว่า สามารถประมาณค่าร่วมที่อันดับ p โดยใช้ตัวกรองคาดคะเนแลตทิซที่อันดับ $(p-1)$ ได้ ซึ่งก็คือค่าความคลาดเคลื่อนย้อนกลับ ดังแสดงในรูปที่ 4.2



รูปที่ 4.2 ตัวกรองประมาณค่าร่วมแลตทิซแบบ MMSE

4.5 คุณสมบัติของตัวกรองประมาณแลตทิซแบบ MMSE

4.5.1 ประสิทธิภาพเสถียรต่อเสถียร

ข้อได้เปรียบอย่างหนึ่งของโครงสร้างแลตทิซ คือ ให้ประสิทธิภาพในการประมาณที่ดีตั้งแต่เสถียรแรกถึงเสถียรสุดท้าย เป็นคุณสมบัติที่สำคัญไม่ทราบมาก่อนว่าอันดับของตัวกรองประมาณจะใช้มากเท่าไรและจะต้องกำหนดในระหว่างการประมาณค่า ในโครงสร้างแบบแลตทิซการประมาณค่าเสถียรแบบ MMSE จะเพิ่มขึ้นเพียงสามครั้ง เพื่อคำนวณหาค่าที่อันดับ $(m+1)$

ของตัวกรองประมาณค่าร่วมแบบ MMES จากผลของตัวกรองประมาณค่าแบบ MMSE ที่อันดับ m และการประมาณค่าเสกสลาห์แบบ MMSE จะเพิ่มขึ้นเพียงสองครั้งสำหรับตัวกรองคาดคะเนแบบ MMSE หรือกล่าวได้ว่า รูปแบบโดยตรงของตัวกรองประมาณค่าแบบ MMSE จะไม่มีการทำซ้ำอันดับ สามารถคำนวณค่าเวกเตอร์สัมประสิทธิ์ที่อันดับ m ได้จากการประมาณค่าแบบ MMSE เพื่อประมาณค่าตัวกรองประมาณค่าแบบ MMSE ตัวอื่นๆที่มีอันดับต่าง ๆ กันได้ เพียงแต่คำนวณค่าเวกเตอร์สัมประสิทธิ์ใหม่โดยใช้สมการเมตริกซ์ใด ๆ ก็ได้ ดังนั้น โครงสร้างแลตทิซจึงได้รับความสนใจในการประยุกต์ใช้งานต่าง ๆ มากมาย

โดยทั่วไปแล้ว ค่า MSE (Mean square error) ของสัญญาณคลาดเคลื่อนเอาต์พุตเป็นตัวชี้วัดที่ดีถึงประสิทธิภาพของตัวกรองประมาณค่า ยกตัวอย่างเช่น ถ้าค่า MSE มีการลดลงไม่มากนักที่เสตท m ซึ่ง $m > m_0$ สรุปได้ว่า ตัวกรองประมาณที่อันดับ m_0 คือ ตัวกรองประมาณที่มีประสิทธิภาพดีที่สุดและมีอันดับต่ำที่สุด สำหรับตัวกรองคาดคะเนลิเนียร์แบบ MMSE ค่า MSE ที่เสตทต่าง ๆ จะได้จากค่า α_m^f และ α_m^b ซึ่งหาได้ง่ายจากตัวกรองคาดคะเนแบบแลตทิซ และไม่ต้องการคำนวณเพิ่ม ส่วนตัวกรองประมาณค่าร่วมที่มีอันดับต่าง ๆ กัน จำเป็นต้องมีการเพิ่มการคำนวณค่า MSE ที่แต่ละเสตทนั้น

4.5.2 คุณสมบัติความสัมพันธ์ของตัวกรองคาดคะเนความคลาดเคลื่อน

คุณสมบัติที่สำคัญที่สุดคือ ความเป็น ออโทโกนอล ระหว่าง ค่าความคลาดเคลื่อนคาดคะเนย้อนกลับและค่าอื่น ๆ ในตารางที่ 4.1

ตารางที่ 4.1 คุณสมบัติของค่าความคลาดเคลื่อนคาดคะเนย้อนกลับแบบ MMSE

$E[e_m^f(k)u^*(k-i)] = 0$	for $1 \leq i \leq m$
$E[e_m^b(k)u^*(k-i)] = 0$	for $0 \leq i \leq m-1$
$E[e_m^f(k)u^*(k)] = E[e_m^b(k)u^*(k-m)] = c_m$	
$E[e_i^f(k)e_j^{f*}(k)] = \alpha_{\max(i,j)}$	for
$E[e_i^f(k)e_j^{f*}(k-l)] = 0$	$\begin{cases} 1 \leq l \leq i-j, & i > j \\ -1 \geq l \geq i-j, & i < j \end{cases}$
$E[e_i^b(k)e_j^{b*}(k-l)] = 0$	for
$E[e_i^f(k+i)e_j^{f*}(k+j)] = 0$	$\begin{cases} 0 \leq l \leq i-j-1, & i > j \\ 0 \geq l \geq i-j+1, & i < j \end{cases}$
$E[e_i^b(k+i)e_j^{b*}(k+j)] = \alpha_{\max(i,j)}$	for $i \neq j$
$E[e_i^f(k)e_j^{b*}(k)] = \begin{cases} k_{j-1}^* \alpha_i, & i \geq j \\ 0, & i < j \end{cases}$	
$E[e_i^f(k)e_j^{b*}(k-1)] = \begin{cases} -k_j^* \alpha_i, & i \geq j \\ 0, & i > j \end{cases}$	
$E[e_i^{b*}(k-1)u(k)] = E[e_i^f(k+1)u^*(k-i)] = \beta_i^f = -k_i \alpha_i$	

สิ่งที่สำคัญที่สุดของตัวกรองประมาณแบบแลตทิซ คือ ค่าความคลาดเคลื่อนคาดคะเนย้อนกลับที่อันดับต่างกัน เป็นออโทโกนอลกับค่าอื่น ๆ ทุกค่า นั่นคือ

$$E[e_{i,MMSE}^{b*}(k)e_{j,MMSE}^b(k)] = 0 \quad \text{for } i \neq j \quad (4.39)$$

พิสูจน์ สมการที่(4.39) สมมุติว่า $i > j$ จะสังเกตได้ว่า จากคุณสมบัติออโทโกนอลพื้นฐาน $e_{i,MMSE}^b(k)$ เป็น ออโทโกนอล กับ $\{u_i(k)\}$ ส่วน $e_{j,MMSE}^b(k)$ เป็นลิเนียร์คอมไบเนชันของ

อินพุต $u(k)$ ถึง $u(k-j)$ ซึ่งเป็นองค์ประกอบของ $u_i(k)$ เมื่อ $i > j$ ซึ่งเป็นไปตามสมการที่ (4.39)

4.5.3 การคำนวณอันดับซ้ำจากการประมาณค่า $y(k)$

เริ่มจากสมการที่ (4.36) ถึง (4.38) ดังนี้

$$e_{m+1,MMSE}^{(y)}(k) = e_{i,MMSE}^{(y)}(k) + \sum_{l=i}^m k_l^{(y)} e_{l,MMSE}^b(k)$$

เนื่องจาก $e_{0,MMSE}^{(y)}(k) \equiv y(k)$ และ ให้ $i=0$ จะได้

$$e_{m+1,MMSE}^{(y)}(k) = y(k) + \sum_{l=0}^m k_l^{(y)} e_{l,MMSE}^b(k) \quad (4.40)$$

เปรียบเทียบสมการที่ (4.40) กับ สมการที่ (4.16) สรุปได้ว่า

$$\hat{y}_{m+1,MMSE}(k) = -\sum_{l=0}^m k_l^{(y)} e_{l,MMSE}^b(k)$$

และ จะได้ การทำซ้ำอันดับสำหรับการประมาณค่าร่วม $\hat{y}_{m,MMSE}(k)$ ดังสมการ

$$\hat{y}_{m+1,MMSE}(k) = \hat{y}_{m,MMSE}(k) - k_m^{(y)} e_{m,MMSE}^b(k)$$

ทำนองเดียวกัน จะได้การทำซ้ำอันดับสำหรับการคาดคะเนเดิหน้า จากสมการ

$$\hat{u}_{m+1,MMSE}^f(k) = \hat{u}_{m,MMSE}^f(k) - k_m e_{m,MMSE}^b(k-1)$$

4.5.4 การคำนวณค่าสัมประสิทธิ์โดยประมาณค่าจากพารามิเตอร์อื่น ๆ ของแลตทิซ

จากที่ผ่านมาจะเห็นแล้วว่า สามารถประมาณค่า $y(k)$ ได้โดยไม่ต้องหาค่าสัมประสิทธิ์ตามแบบ MMSE เมื่อสัมประสิทธิ์สะท้อนกลับแลตทิซและ สัมประสิทธิ์จากการประมาณค่าร่วมสามารถหาได้แล้ว การประมาณค่าสัมประสิทธิ์ของตัวกรองแบบทรานสเวอร์ซอลก็สามารถหาได้จากวิธีการของ Levinson algorithm เช่นกัน

4.5.5 การทำซ้ำอันดับของการคาดคะเนความคลาดเคลื่อนยกกำลังสองเฉลี่ย α_m

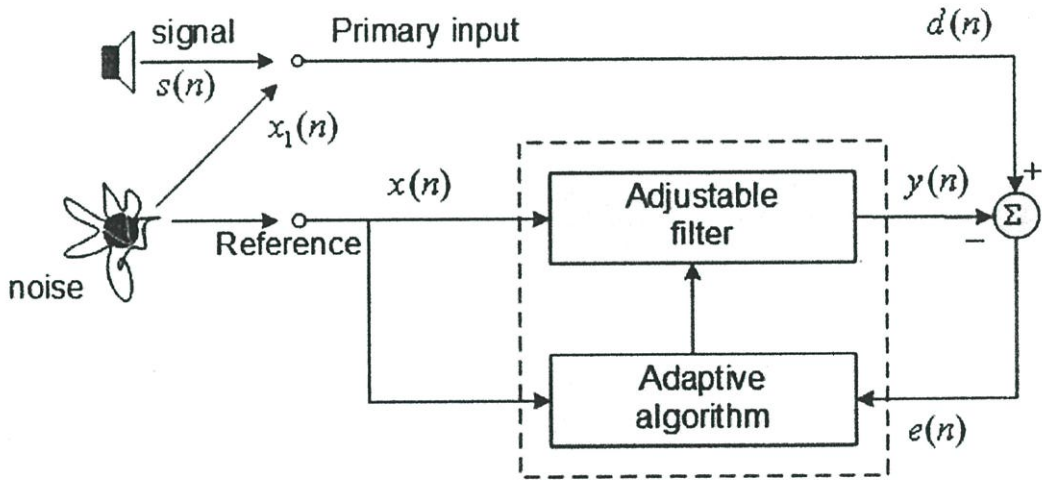
ค่า MSE ของความคลาดเคลื่อนคาดคะเนทั้งเดินหน้าและย้อนกลับที่อันดับ $(m + 1)$ สามารถคำนวณหาได้จาก ค่า MSE ที่อันดับ m ดังนี้

$$\begin{aligned}\alpha_{m+1} &= \alpha_m + \beta_m^{f*} k_m \\ &= \alpha_m - |\beta_m^f|^2 / \alpha_m = \alpha_m - \alpha_m |k_m|^2 = \alpha_m (1 - |k_m|^2)\end{aligned}\quad (4.41)$$

คุณสมบัติต่าง ๆ ที่ได้อธิบายมาแล้ว ไม่เพียงแต่เป็นประโยชน์สำหรับตัวกรองประมาณค่า หรือ ตัวกรองคาดคะเนแลตทิซแบบ MMSE เท่านั้น ยังได้รับการยอมรับสำหรับตัวกรองประมาณค่าอื่น ๆ ที่ใช้ฟังก์ชัน quadratic cost อีกด้วย

ตัวกรองอะแดปทีฟ IIR แบบนอตช์อันดับสอง

Widrow [26] ได้นำเสนอแนวความคิดในการกำจัดสัญญาณรบกวนแบบอะแดปทีฟ แสดงได้ดังรูปที่ 5.1 แนวความคิดนี้ถูกนำมาประยุกต์ใช้กับระบบการประมวลสัญญาณดิจิทัลและระบบสื่อสารอย่างกว้างขวาง เช่น adaptive equalization, echo cancellation, noise reduction, speech analysis and synthesis เป็นต้น ตัวกรองแบบอะแดปทีฟมีข้อดีที่สามารถปรับสัมประสิทธิ์ของตัวกรองได้ จึงไม่จำเป็นต้องทราบคุณสมบัติที่สำคัญ ๆ ของสัญญาณที่ต้องการประมาณ และสัญญาณรบกวนก็ได้ (Primary knowledge) หรือถ้าต้องการก็เพียงเล็กน้อยเท่านั้น



รูปที่ 5.1 แนวความคิดในการกำจัดสัญญาณรบกวนแบบอะแดปทีฟ

การประมาณค่าสัญญาณขายน้กลั่นเดี่ยวที่ปนอยู่ในสัญญาณรบกวนนั้น สามารถทำได้โดยการป้อนสัญญาณดังกล่าว ผ่านตัวกรองความถี่ ซึ่งตัวกรองความถี่จะทำการลดทอนสัญญาณรบกวนให้มีขนาดต่ำลง และยอมให้สัญญาณที่ต้องการผ่านออกไปโดยไม่มีคามผิดเพี้ยน วิธีการหนึ่งที่น่าสนใจสำหรับการตรวจวัดคลื่นความถี่ขายน้ที่มีความถี่เดี่ยว คือ การใช้ตัวกรองความถี่อะแดปทีฟแบบนอตช์ IIR อันดับสอง (Second order adaptive IIR notch filter, ANF)

การออกแบบสร้างตัวกรองโดยทั่วไปแล้ว จำเป็นต้องทราบคุณลักษณะข้อมูลทางสถิติของสัญญาณล่วงหน้า (Prior information) ซึ่งเป็นของสัญญาณนั้น ๆ ตัวกรองนั้นจะเหมาะสมที่สุดเมื่อคุณลักษณะของสัญญาณอินพุตตรงกันกับข้อมูลของคุณลักษณะที่ได้ใช้ทำการออกแบบ แต่ถ้าไม่ทราบข้อมูลคุณลักษณะทั้งหมดของสัญญาณที่ต้องการ หรือข้อมูลมีการเปลี่ยนแปลงไป ตัวกรองที่ได้ออกแบบไว้แล้วนั้น จะไม่สามารถทำงานได้ดีที่สุด วิธีการแก้ไขโดยตรงเรียกว่า

“estimate and plug” คือ การประมาณค่าสัมประสิทธิ์ทางสถิติของสัญญาณที่ต้องการ และทำการป้อนกลับผลลัพธ์ที่ได้เข้าสู่ตัวกรอง วิธีการนี้เมื่อนำไปใช้งานจริงจะมีข้อเสียที่ข้อมูลที่ใช้ในการประมวลผลมีความละเอียดมากเกินไป และมีราคาแพง ดังนั้นจึงมีการนำตัวกรองแบบอะแดปทีฟ (Adaptive filter) ซึ่งเป็นตัวกรองที่สามารถปรับตัวเองได้ ด้วยวิธีการป้อนกลับมาใช้งานแทน เนื่องจากสามารถปรับตัวให้สอดคล้องกับการเปลี่ยนแปลงของสัญญาณที่ไม่ทราบล่วงหน้าได้ กระบวนการปรับตัวหรืออะแดปทีฟอัลกอริทึม (Algorithm) จะเริ่มต้นการทำงานจากเงื่อนไขที่ตั้งค่าไว้ในตอนแรก (Initial condition) สัญญาณที่มีลักษณะเป็น stationary กระบวนการปรับของอัลกอริทึมจะลู่เข้า (Converge) สู่ค่าที่เหมาะสมที่สุด ส่วนในสัญญาณที่มีลักษณะแบบ nonstationary อัลกอริทึมจะต้องสามารถติดตาม (Tracking) สัญญาณอินพุตที่เปลี่ยนแปลงตามเวลาได้

สิ่งที่สำคัญของตัวกรองแบบอะแดปทีฟคือ อัลกอริทึมที่ใช้ในการปรับค่าสัมประสิทธิ์ของตัวกรอง ดังนั้น อัลกอริทึมที่มีประสิทธิภาพที่ดี ควรมีคุณสมบัติดังต่อไปนี้ [2] อัตราการลู่เข้า (Rate of convergence) จะบอกถึงจำนวนรอบที่อัลกอริทึมประมวลผลต่อการตอบสนองของสัญญาณ และลู่เข้าสู่ค่าที่เหมาะสมที่สุด อัตราการลู่เข้าที่รวดเร็วจะทำให้อัลกอริทึมลู่เข้าสู่ค่าตอบอย่างรวดเร็ว เมื่อสัญญาณที่ไม่ทราบค่ามีลักษณะเป็นแบบ stationary

- I. Misadjustment คือ ค่าความผิดพลาดในการปรับตัวของตัวกรอง เมื่อค่าที่ประมาณได้จากอัลกอริทึมเบี่ยงเบนจากค่าที่เหมาะสมที่สุด คำนวณจากค่าความคลาดเคลื่อนยกกำลังสองเฉลี่ย (mean-square error) โดยวิธีการหาค่าเฉลี่ยเชิงสถิติของข้อมูลทั้งหมด
- II. การติดตาม (Tracking) เมื่ออัลกอริทึมของการกรองแบบอะแดปทีฟทำงานในสถานะที่สัญญาณมีลักษณะแบบ nonstationary อัลกอริทึมต้องสามารถติดตามค่าที่เปลี่ยนแปลงตามเวลานั้นได้ คุณสมบัติของการติดตามของอัลกอริทึมที่ดีได้แก่ 1) อัตราการลู่เข้า และ 2) การกระเพื่อมของสัมประสิทธิ์ที่สถานะอยู่ตัว (Steady-state fluctuation) ของอัลกอริทึม
- III. Robustness อัลกอริทึมควรมีความทนทานต่อการรบกวนเล็กน้อย (เช่นการรบกวนจากพลังงานเล็ก ๆ ซึ่งอาจเกิดขึ้นจากภายในหรือภายนอกของตัวกรอง) และควรมีผลเพียงเล็กน้อยต่อการทำงานของอัลกอริทึม
- IV. Computational requirements ได้แก่ 1) จำนวนของการคูณ การหาร การบวก และการลบที่เกิดขึ้นต่อหนึ่งกระบวนการซ้ำของอัลกอริทึม 2) ขนาดของหน่วยความจำ (Memory) ที่ใช้สำหรับเก็บข้อมูลและโปรแกรม และ 3) ทรัพยากรที่ต้องใช้ในการกำหนดวิธีการของอัลกอริทึม
- V. โครงสร้าง (Structure) ของอัลกอริทึม รวมถึงโครงสร้างของการนำข้อมูลเข้าสู่อัลกอริทึม ซึ่งเป็นการกำหนดวิธีการในการสร้างฮาร์ดแวร์ เช่น อัลกอริทึมที่มีโครงสร้างเป็น โมดูล

(Module) มีความคล้ายคลึงกัน (Parallelism) และทำงานพร้อมกัน (Concurrency) จะเหมาะสำหรับสร้างด้วย VLSI (Very large-scale integration)

VI. คุณสมบัติทางระบบตัวเลข (Numerical properties) เมื่ออัลกอริทึมทำการคำนวณทางตัวเลข สัญญาณรบกวนที่เกิดจากการควอนไทซ์ (Quantization errors) ของการแปลงสัญญาณ อินพุตแบบอะนาล็อกเป็นสัญญาณดิจิทัล จะมีผลต่อการคำนวณตัวเลขทางดิจิทัล คุณสมบัติทางตัวเลขนี้ ยังสามารถแยกออกเป็น Numerical stability ซึ่งเป็นคุณสมบัติของอัลกอริทึมของการกรองแบบอะแดปทีฟ และ Numerical accuracy คือ จำนวนบิตที่ใช้สำหรับข้อมูลและสัมประสิทธิ์ของตัวกรอง

ตัวกรองความถี่ดิจิทัลแบบปรับตัวได้ (Adaptive digital filter) ได้เข้ามามีบทบาทเป็นอย่างมากในงานประมวลผลสัญญาณดิจิทัล เช่น ระบบสื่อสาร ระบบควบคุม เรดาร์ โซนาร์ และเครื่องมือทางการแพทย์ ทั้งนี้เพราะความสามารถในการปรับเปลี่ยนคุณลักษณะของระบบให้สอดคล้องกับสัญญาณที่เข้ามาได้ด้วยตัวเอง ซึ่งจะสามารถทำงานกับสัญญาณที่มีลักษณะเป็น non-stationary ได้ และเป็นสัญญาณที่เกิดขึ้นจริงในทางปฏิบัติ ตัวกรองความถี่แบบอะแดปทีฟ นั้น มีสองชนิด คือ ตัวกรองอะแดปทีฟแบบ FIR (Finite Impulse Response) และ แบบ IIR (Infinite Impulse Response) ตัวกรองแบบอะแดปทีฟ IIR ได้ถูกนำมาใช้งานต่าง ๆ แทนตัวกรองอะแดปทีฟแบบ FIR มากขึ้น โดยเฉพาะงานที่ใช้ตัวกรองแบบ FIR ที่มีจำนวนอันดับสูง ๆ เมื่อเปลี่ยนมาใช้ตัวกรองแบบ IIR แล้ว จะสามารถลดอันดับลงได้ โดยที่สมรรถนะยังคงเดิม อย่างไรก็ตาม ตัวกรองความถี่อะแดปทีฟแบบ IIR โครงสร้างแบบโดยตรง (Direct form) จะมีข้อเสียในเรื่อง Finite-precision effect และการวิเคราะห์หาเสถียรภาพของตัวกรอง (Complexity of stability monitoring) จึงทำให้มีการค้นหาโครงสร้างใหม่ๆ เช่น โครงสร้างแบบแลตทิซ (Lattice) ซึ่งเป็นโครงสร้างที่มีลักษณะแบบไขว้ ซึ่งสามารถลดความยุ่งยากในการคำนวณลงได้

จากความต้องการตัวกรองอะแดปทีฟที่มีความสามารถในการตอบสนองต่ออิมพัลส์ให้ยาวนานมากขึ้น เช่น การขจัดสัญญาณเสียงสะท้อน (Acoustic echo cancellation) ในการประชุมทางไกลผ่านโครงข่ายโทรศัพท์ อาจจะใช้อัลกอริทึมแบบ NLMS หรือ LMS ในการขจัดสัญญาณสะท้อนก็ได้ หรืออาจจะเลือกตามความต้องการอย่างใดอย่างหนึ่งได้ ดังนี้

- สร้างกระบวนการปรับตัวของตัวกรองแบบลิเนียร์ให้มีผลตอบสนองต่ออิมพัลส์ได้ไม่จำกัด(IIR) ซึ่งจะใช้การคำนวณน้อยกว่าตัวกรองอะแดปทีฟแบบ FIR มาก
- ใช้กระบวนการทำซ้ำของตัวกรองแบบอะแดปทีฟ เช่น ตัวกรองแบบทรานสเวอร์ซอลลาแกร์ ที่รวมเอาตัวกรองแบบ FIR และ IIR เข้าไว้ด้วยกัน

วิธีการในการออกแบบตัวกรองอะแดปทีฟตามความต้องการที่กล่าวมาทั้งสอง มีดังต่อไปนี้

5.1 ตัวกรองอะแดปทีฟแบบ IIR โดยวิธี OUTPUT ERROR [17]

โครงสร้างพื้นฐานตัวกรองแบบ IIR โดยทั่วไปจะคล้ายกับ autoregressive moving-average (ARMA) model ซึ่งมีความสัมพันธ์อินพุตและเอาต์พุต ดังนี้

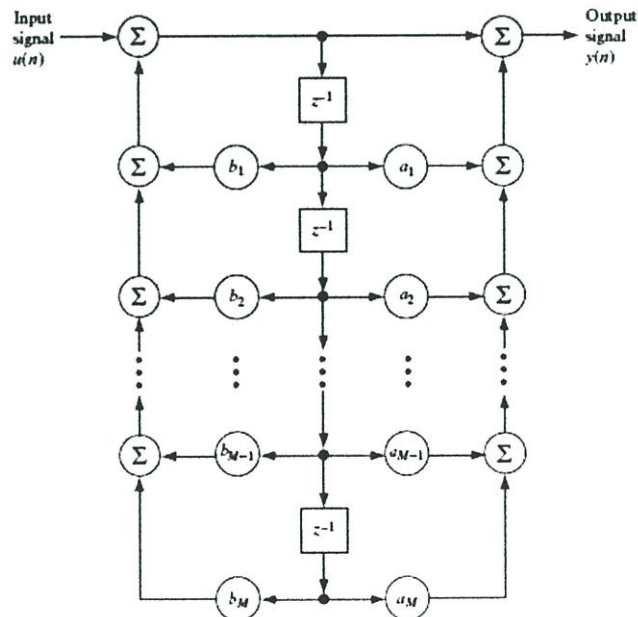
$$y(n) = \sum_{i=0}^M a_i(n)u(n-i) + \sum_{i=1}^N b_i(n)y(n-i) \quad (5.1)$$

ซึ่ง $a_i(n)$ และ $b_i(n)$ คือ สัมประสิทธิ์ปรับค่าได้ของตัวกรอง แบบจำลองรูปที่ 5.2 แสดงบล็อกไดอะแกรมของสมการที่ (5.1) ถ้า $N=M$ และ $a_0=1$ สัมประสิทธิ์ตัวกรองคงที่ จะมีคุณสมบัติฟังก์ชันถ่ายโอนเหมือนแบบจำลอง ARMA

$$H(z) = \frac{A(z)}{1-B(z)}, \quad (5.2)$$

ซึ่ง

$$A(z) = \sum_{i=0}^M a_i z^{-i} \quad (5.3)$$



รูปที่ 5.2 บล็อกไดอะแกรมตัวกรองแบบ IIR

และ

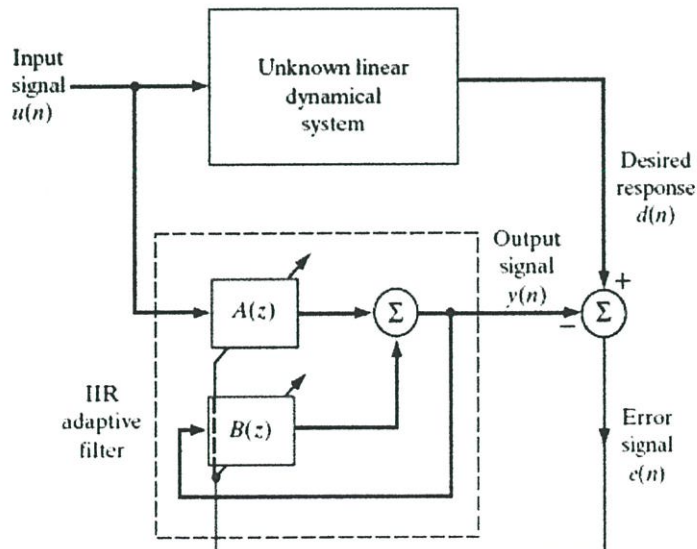
$$B(z) = \sum_{i=1}^N b_i z^{-i}. \quad (5.4)$$

ในการออกแบบตัวกรองปรับตัวได้แบบ IIR ให้มีผลตอบสนองต่ออิมพัลส์ที่ยาวนานมากขึ้น ด้วยการปรับค่าสัมประสิทธิ์จำนวนน้อย ๆ อาจจะใช้หนึ่งในสองวิธีดังนี้ คือ วิธีการปรับค่าความคลาดเคลื่อนจากเอาต์พุต (Output error method) หรือ วิธีการปรับค่าความคลาดเคลื่อนจากสมการ (Equation error method)

5.1.1 วิธีการปรับค่าความคลาดเคลื่อนเอาต์พุต (Output error method)

ในการจำลองระบบที่มีการเปลี่ยนแปลงแบบลิเนียร์ที่ไม่ทราบคุณลักษณะของระบบมาก่อน จะแสดงค่าในโดเมน Z สัญญาณอินพุต $u(n)$ จะถูกป้อนให้กับระบบที่ไม่รู้จักและตัวกรองอะแดปทีฟ พร้อม ๆ กัน จะได้ผลตอบสนองที่ต้องการ $d(n)$ และเอาต์พุตจริงของตัวกรอง $y(n)$ ตามลำดับ เวกเตอร์นำหน้ารวมของตัวกรองอะแดปทีฟกำหนดให้สัมประสิทธิ์ทั้งสองกลุ่มเชื่อมโยงถึงกันทั้ง $\{a_i(n)\}_{i=0}^M$ และ $\{b_i(n)\}_{i=1}^N$ ตามสมการ

$$\hat{W}(n) = [a_0(n), a_1(n), \dots, a_M(n), b_1(n), b_2(n), \dots, b_N(n)]^T. \quad (5.5)$$



รูปที่ 5.3 บล็อกโคอะแกรมวิธีการปรับตัวจากค่าความคลาดเคลื่อนเอาต์พุตของตัวกรองอะแดปทีฟ

IIR

สันนิษฐานได้ว่า สัญญาณเป็นค่าจำนวนจริง การปรับเวกเตอร์น้ำหนัก $\hat{W}(n)$ จะกำหนดจากการทำให้ cost function ชั่วขณะ (Instantaneous) มีค่าน้อยที่สุด

$$J(n) = \frac{1}{2} e^2(n), \quad (5.6)$$

ซึ่งสัญญาณความคลาดเคลื่อน หาได้จาก

$$e(n) = d(n) - y(n) \quad (5.7)$$

$e(n)$ คือ ความแตกต่างระหว่างผลตอบสนองที่ต้องการและเอาต์พุตจริงของตัวกรอง เอาต์พุต $y(n)$ กำหนดโดยสมการที่ (5.1) ที่มาจากอัลกอริทึมแบบ LMS ซึ่ง การปรับค่า $\Delta\hat{W}(n)$ จะเปลี่ยนแปลงตามเวกเตอร์น้ำหนัก $\hat{W}(n)$ ทุกค่า n กำหนดได้จาก

$$\Delta\hat{W}(n) = -\mu \frac{\partial J(n)}{\mu \partial \hat{W}(n)}, \quad (5.8)$$

ซึ่ง μ คือ ตัวแปรกำหนดค่าเสถียรไซส์ การดิฟเฟอเรนเชียล cost function $J(n)$ เทียบกับ $\hat{W}_j(n)$ ทำให้ได้ส่วนประกอบของเวกเตอร์เกรเดียนต์ $\partial J(n)/\partial \hat{W}_j(n)$ ที่อันดับ J^{th}

$$\begin{aligned} \frac{\partial J(n)}{\partial \hat{W}_j(n)} &= e(n) \frac{\partial e(n)}{\partial \hat{W}_j(n)} \\ &= -e(n) \frac{\partial y(n)}{\partial \hat{W}_j(n)} \end{aligned} \quad (5.9)$$

จากสมการที่ (5.1) เห็นได้ว่า เอาต์พุตของตัวกรอง $y(n)$ จะขึ้นกับค่าเอาต์พุตในอดีต $y(n-1)$ ด้วย ดังนั้นจะทำการคำนวณด้วยการดิฟเฟอเรนเชียลทีละส่วน จะได้

$$\frac{\partial y(n)}{\partial \hat{W}_j(n)} = \begin{cases} u(n-j) + \sum_{i=1}^N b_i(n) \frac{\partial y(n-i)}{\partial a_j(n)} & \text{for } \hat{W}_j(n) = a_j(n) \\ y(n-j) + \sum_{i=1}^N b_i(n) \frac{\partial y(n-i)}{\partial b_j(n)} & \text{for } \hat{W}_j(n) = b_j(n) \end{cases} \quad (5.10)$$

เพื่อให้ดูง่ายขึ้น จะแทนค่าใหม่ตามนิยามทั้งสอง จะได้

$$\frac{\partial y(n)}{\partial a_j(n)} = \alpha_j(n) \quad \text{for } j = 0, 1, \dots, M; \quad (5.11)$$

$$\frac{\partial y(n)}{\partial b_j(n)} = \beta_j(n) \quad \text{for } j = 0, 1, \dots, N; \quad (5.12)$$

สันนิษฐานอีกว่า เมื่อให้ค่า M และ N มีค่าน้อยมาก ๆ ตัวกรองจะปรับตัวช้า ๆ เมื่อ $i = 0, 1, \dots, N$; จะประมาณได้ว่า

$$\frac{\partial y(n-i)}{\partial a_j(n)} \approx \frac{\partial y(n-i)}{\partial a_j(n-i)} = \alpha_j(n-i); \quad (5.13)$$

$$\frac{\partial y(n-i)}{\partial b_j(n)} \approx \frac{\partial y(n-i)}{\partial b_j(n-i)} = \beta_j(n-i); \quad (5.14)$$

ผลจากการทำซ้ำในสมการที่ (5.13) และ (5.14) เขียนสมการที่ (5.10) ใหม่ในรูปของ $\alpha_j(n)$ และ $\beta_j(n)$ ได้ว่า

$$\alpha_j(n) \approx u(n-j) + \sum_{i=1}^N b_i(n) \alpha_j(n-1) \quad \text{for } j = 0, 1, \dots, M; \quad (5.15)$$

$$\beta_j(n) \approx y(n-j) + \sum_{i=1}^N b_i(n) \beta_j(n-1) \quad \text{for } j = 0, 1, \dots, N; \quad (5.16)$$

ดังนั้น จะใช้สมการที่ (5.8) (5.11) และ (5.12) เป็นรูปแบบในการกำหนดสมการอັพเดทในการคำนวณเวกเตอร์น้ำหนักของตัวกรองอะแดปทีฟ IIR ดังนี้

$$\begin{aligned} \hat{W}(n+1) &= \hat{W}(n) + \Delta \hat{W}(n) \\ &\approx \hat{W}(n) + \mu e(n) \gamma(n), \end{aligned} \quad (5.17)$$

เมื่อ

$$\gamma(n) = [\alpha_0(n), \alpha_1(n), \dots, \alpha_M(n), \beta_1(n), \beta_2(n), \dots, \beta_N(n)]^T \quad (5.18)$$

$\gamma(n)$ เรียกว่า filtered regressor vector เป็นเวกเตอร์จากการป้อนกลับที่ผ่านการกรองแล้ว การคำนวณสัมประสิทธิ์ $\alpha_j(n)$ และ $\beta_j(n)$ ที่อยู่ในรูปของ $\gamma(n)$ จะถูกทำซ้ำ โดยสมการความแตกต่างที่ (5.15) และ (5.16)

ข้อจำกัดของวิธีการปรับค่าความคลาดเคลื่อนเอาต์พุต (Output error method)

วิธีการปรับค่าความคลาดเคลื่อนเอาต์พุต (Output error method) ที่นำไปใช้ในการออกแบบตัวกรองอะแดปทีฟ IIR ในทางปฏิบัติมีข้อจำกัดอยู่ สองข้อ คือ

- I. พื้นผิวสมรรถนะความคลาดเคลื่อนจะมี local minima จำนวนมากรวมอยู่กับ global minima ซึ่งหมายความว่า ไม่สามารถแน่ใจได้เลยว่า ตัวกรองอะแดปทีฟ IIR ที่ออกแบบนั้น ทำงานได้ในจุดที่เหมาะสมที่สุด (Globally optimum)
- II. ไม่มีการรับรองได้ว่า โพล ของแบบจำลอง ARMA นั้นจะอยู่ภายในวงกลมหนึ่งหน่วยในระนาบ Z เสมอไป หมายความว่า ตัวกรองอะแดปทีฟ IIR อาจจะไม่เสถียรได้

5.1.2 วิธีการปรับค่าความคลาดเคลื่อนสมการ (Equation error method)

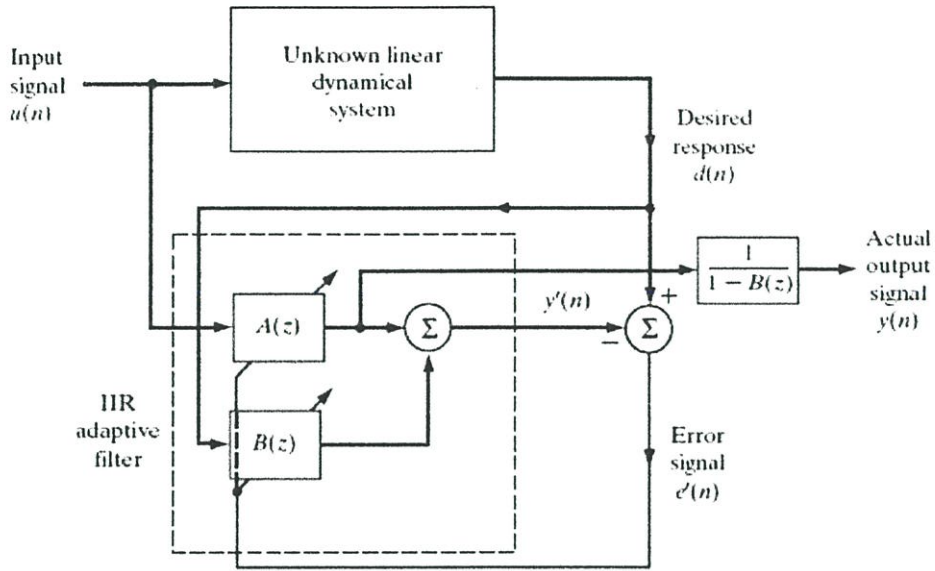
วิธีการออกแบบตัวกรองอะแดปทีฟ IIR วิธีที่สองนี้ ในขณะที่ตัวกรองเรียนรู้ (Training) ผลตอบสนองที่ต้องการ $d(n)$ จะถูกนำไปแทนที่ค่าเอาต์พุตจริงของตัวกรอง $y(n)$ ในเทอมผลบวกที่สองของสมการที่ (5.1)

$$y(n) = \sum_{l=0}^M a_l(n)u(n-l) + \sum_{l=1}^N b_l(n)d(n-l) \quad (5.19)$$

กำหนดสัญญาณความคลาดเคลื่อนได้จาก

$$e'(n) = d(n) - y'(n), \quad (5.20)$$

เป็นการกำหนดค่าเอาต์พุตจากสมการแทนที่จะเป็นค่าเอาต์พุตจริงเหมือนกับวิธีแรก จากการจำลองระบบแสดงได้ในรูปที่ 5.4



รูปที่ 5.4 บล็อกไดอะแกรมของวิธีการปรับตัวจากค่าความคลาดเคลื่อนสมการ

วิธีการนี้จะมีคุณลักษณะจากพื้นผิวสมรรถนะความคลาดเคลื่อนแบบ quadratic (Quadratic error performance surface) และจะสามารถแก้ปัญหาการเกิด local minima ได้

ความคลาดเคลื่อนที่ได้จากสมการ $e'(n)$ จะมีความสัมพันธ์ที่เป็นธรรมชาติของความคลาดเคลื่อนเอาต์พุต $e(n)$ โดยเฉพาะการแทนค่าสมการที่ (5.16) ด้วยสมการที่ (5.21) และแทนนิยามของ $y(n)$ ด้วย $y'(n)$ จากสมการที่ (5.1) และ (5.20) ตามลำดับ จะได้

$$\begin{aligned}
 e'(n) - e(n) &= y'(n) - y(n) \\
 &= -\sum_{i=1}^N b_i(n)(d(n-i) - y(n-i)) \\
 &= -\sum_{i=1}^N b_i(n)e(n-i).
 \end{aligned} \tag{5.21}$$

จัดรูปสมการใหม่จะได้

$$e'(n) = e(n) - \sum_{i=1}^N b_i(n)e(n-i). \tag{5.22}$$

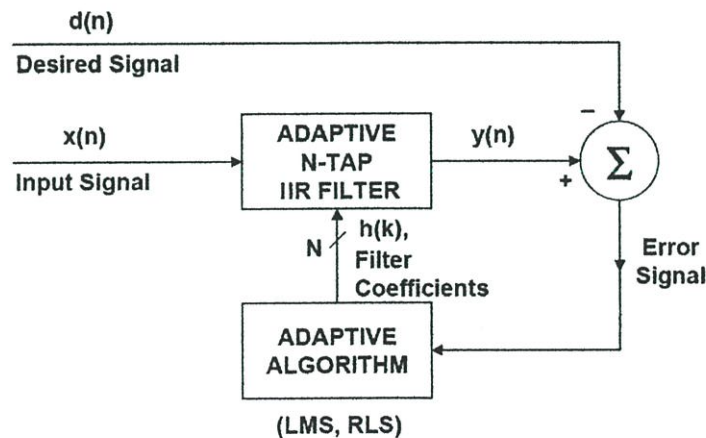
เทอมของผลบวกในสมการที่ (5.22) คือ การคอนโวลูชันของลำดับข้อมูลสองชุด คือ b_k และ $e(k)$ และเปลี่ยนให้อยู่ในรูป Z- transform จะ ได้

$$E'(z) = (1 - B(z))E(z), \quad (5.23)$$

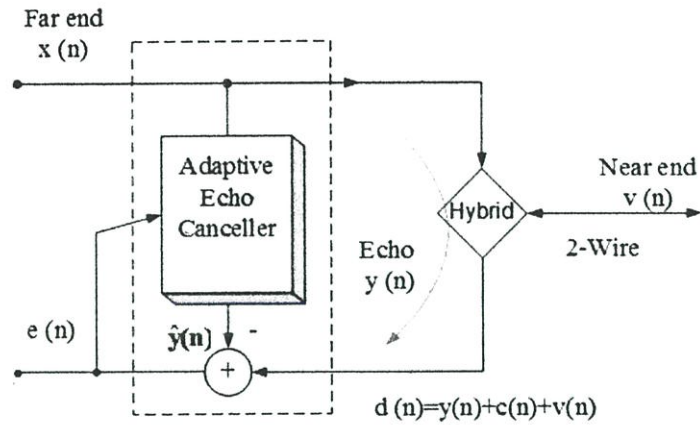
เมื่อ $E'(z)$ และ $E(z)$ คือ Z-transform ของความคลาดเคลื่อนสมการ $e'(n)$ และ ความคลาดเคลื่อนเอาต์พุต $e(n)$ ตามลำดับ และ Z- transform ของ $B(z)$ กำหนดจากสมการ (5.4) ซึ่งเป็นค่าที่กำหนดไว้ตายตัว สมการที่ (5.23) แสดงให้เห็นว่า ความคลาดเคลื่อนสมการ $e'(n)$ ได้มาจากการผ่านสัญญาณความคลาดเคลื่อนจริงเข้าไปในระบบที่มีฟังก์ชันถ่ายโอนเท่ากับ $1 - B(z)$

5.2 โครงสร้างของตัวกรองความถี่อะแดปทีฟแบบ IIR อันดับสอง

รูปที่ 5.5 แสดงส่วนประกอบโครงสร้างทั่วไปของตัวกรองความถี่อะแดปทีฟ เป้าหมายคือการกรองสัญญาณอินพุต $x(n)$ ด้วยตัวกรองแบบอะแดปทีฟเพื่อให้สัญญาณนั้นเหมือนกับสัญญาณที่ต้องการ $d(n)$ (Desired signal) มากที่สุด จากนั้นสัญญาณ $d(n)$ จะถูกนำไปลบกับสัญญาณที่ได้จากตัวกรอง $y(n)$ จะได้สัญญาณผิดพลาด $e(n)$ (Error) และสัญญาณผิดพลาดนี้จะใช้ป้อนกลับเข้าไปในวงจรเพื่อควบคุมการทำงานของอัลกอริทึมที่ใช้สำหรับปรับสัมประสิทธิ์ของตัวกรอง เพื่อลดค่าผิดพลาดนั้นให้ต่ำที่สุด



รูปที่ 5.5 ส่วนประกอบของตัวกรองความถี่อะแดปทีฟแบบ IIR



รูปที่ 5.6 การประยุกต์ใช้งานเพื่อทำการขจัดสัญญาณสะท้อนกลับในระบบโทรศัพท์

ตัวอย่างการนำไปประยุกต์ใช้งานที่แสดงได้ดังรูปที่ 5.6 เป็นการขจัดสัญญาณสะท้อนในระบบโทรศัพท์ (Echo cancellation) ที่นำเสนอในวิทยานิพนธ์นี้ เมื่อ $d(n)$ คือสัญญาณเสียงที่จะส่งเข้าไปในระบบโทรศัพท์ ประกอบไปด้วย สัญญาณสะท้อนที่ย้อนผ่านไฮบริดเข้ามา $y(n)$ รวมกับ สัญญาณรบกวน $c(n)$ และ สัญญาณเสียงพูดต้นทาง $v(n)$ $x(n)$ คือ สัญญาณสะท้อนด้านไกลที่ใช้เป็นสัญญาณอ้างอิง ซึ่งโดยทั่วไปจะถูกรบกวนด้วยสัญญาณรบกวน วัตถุประสงค์ของอัลกอริทึมคือ พยายามทำให้ค่า $\hat{y}(n)$ มีค่าใกล้เคียงกับสัญญาณสะท้อนมากที่สุดเพื่อไม่ให้มีสัญญาณสะท้อนย้อนกลับไปทางภาคส่ง $e(n)$ ได้

5.2.1 โครงสร้างแบบโดยตรง

สำหรับฟังก์ชันถ่ายโอนของวงจรรองคิจิตอล IIR แบบนอคซ์อันดับสองในโครงสร้างแบบโดยตรงสามารถแสดงได้ดังนี้

$$H_d(z) = \frac{1 + a(n)z^{-1} + z^{-2}}{1 + \rho a(n)z^{-1} + \rho^2 z^{-2}} \quad (5.24)$$

เมื่อ d คือ โครงสร้างแบบโดยตรง ρ คือค่าคงที่ที่มีค่าอยู่ระหว่าง $0 < \rho < 1$ ใช้สำหรับกำหนดค่า Q-factor ของวงจรรองความถี่ และ $a(n)$ คือสัมประสิทธิ์แปรค่าของตัวกรอง ซึ่งจะถูกระมาณค่าโดยอะแดปทีฟอัลกอริทึมมีค่าอยู่ระหว่าง $-2 \leq a(n) \leq 2$ เพื่อกำจัดสัญญาณชาयน์ที่มีความถี่ ω_0 และสามารถหาค่า ω_0 และค่า Q-factor ของวงจรรองความถี่ได้ตามสมการที่ (5.25) และ (5.26) ตามลำดับ

$$a(n) = -2 \cos(\omega_0) \quad (5.25)$$

$$BW = \pi(1 - \rho) \quad (5.26)$$

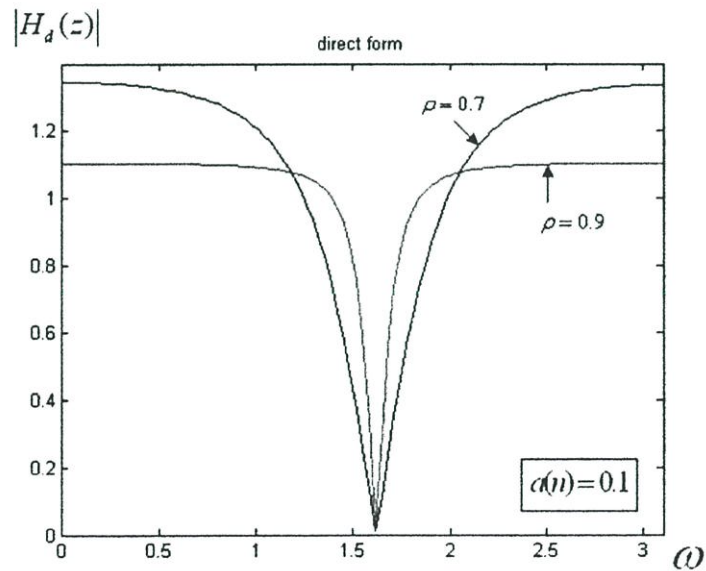
$$Q = \frac{\omega_0}{\pi(1 - \rho)} \quad (5.27)$$

จากสมการฟังก์ชันถ่ายโอนในโครงสร้างแบบโดยตรง สามารถแสดงสมการเอาต์พุตของตัวกรองได้ โดยทำการแปลง z กลับ (Inverse z-transform) ดังนี้

$$y(n) = x(n) + a(n) \cdot x(n-1) + x(n-2) - a(n) \cdot \rho \cdot y(n-1) - \rho^2 \cdot y(n-2) \quad (5.28)$$

และสามารถหาสมการสัญญาณกระตุ้นของตัวกรองโครงสร้างรูปแบบโดยตรงได้จาก $\frac{\partial y(n)}{\partial a(n)}$ ดังนี้จะได้

$$g_d(n) = x(n-1) - \rho y(n-1) \quad (5.29)$$



รูปที่ 5.7 ผลตอบสนองทางขนาดต่อความถี่ของตัวกรองแบบนอกรีตโครงสร้างโดยตรงเมื่อเปลี่ยนค่าสัมประสิทธิ์ ρ

จากรูปที่ 5.7 แสดงผลตอบสนองทางขนาดต่อความถี่ $|H_d(z)|$ โดยการปรับเปลี่ยนค่าพารามิเตอร์ ρ และให้ $a(n) = 0.1$ จะเห็นได้ว่า เมื่อ ρ เข้าใกล้ 1 จะทำให้แบนด์วิดท์ของตัวกรองแคบลง ทำให้ $|H_d(z)|$ มีค่าใกล้เคียงกับผลตอบสนองทางความถี่ในอุดมคติ $a(n)$ เป็นค่าพารามิเตอร์ที่ใช้กำหนดความถี่นอตช์

5.2.2 โครงสร้างรูปแบบแลตทิซ [1]

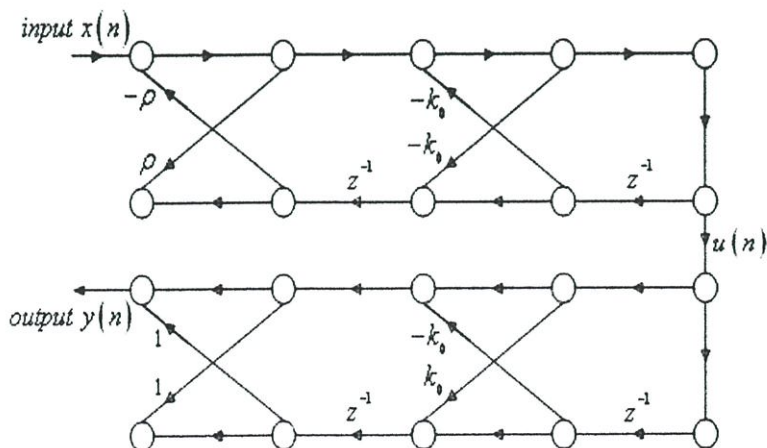
ตัวกรองอะแดปทีฟ IIR แบบนอตช์โครงสร้างรูปแบบแลตทิซ (Adaptive lattice notch filter) เป็นโครงสร้างที่ลดความซับซ้อนทางโครงสร้าง ทำให้ตัวกรองอะแดปทีฟมีลักษณะการลู่เข้าที่ดีขึ้น เมื่อเปรียบเทียบกับโครงสร้างรูปแบบโดยตรง ฟังก์ชันถ่ายโอนสำหรับวงจรกรองแบบนอตช์อันดับสองในรูปแบบแลตทิซแสดงได้ดังนี้

$$H_l(z) = \frac{1 + 2k_0(n)z^{-1} + z^{-2}}{1 + k_0(n)(1 + \rho)z^{-1} + \rho z^{-2}} \quad (5.30)$$

เมื่อ l คือโครงสร้างแลตทิซ $k_0(n)$ คือสัมประสิทธิ์ของตัวกรองแลตทิซ รูปที่ 5.8 (a) แสดงโครงสร้างตัวกรองแบบ cascading all-pole และ all-zero ของ ANF สามารถเขียนยวบรวมให้อยู่ในรูปแบบอื่นได้ดังรูปที่ 5.8 (b) โดยที่

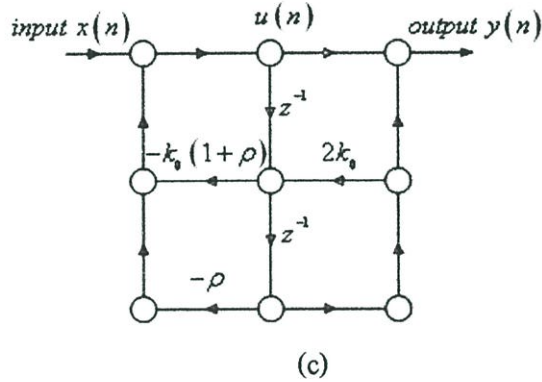
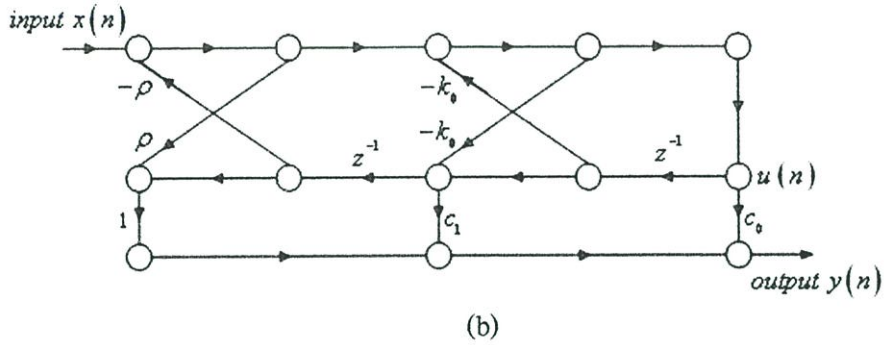
$$c_1 = k_0(1 - \alpha) \quad (5.31a)$$

$$c_0 = (1 - k_0^2)(1 - \alpha) \quad (5.31b)$$



(a)

โครงสร้างของตัวกรองในรูปที่ 5.8 (b) นั้น ต้องใช้จำนวนการคูณมากกว่าโครงสร้างในรูปที่ 5.8 (a) แต่มีข้อดีตรงที่สามารถลดความซับซ้อนของโครงสร้างลงไปได้มาก รูปที่ 5.8(c) เป็นโครงสร้างที่เรียกว่า Normalized lattice ซึ่งสามารถลดความซับซ้อนของโครงสร้างและการคำนวณลงได้อีก



รูปที่ 5.8 ตัวกรองอะแดปทีฟแบบนอกรีตที่มีโครงสร้างแบบแลตทิซ

และจะได้สมการของเอาต์พุตจากฟังก์ชันถ่ายโอนของตัวกรองที่มีโครงสร้างแบบแลตทิซคือ

$$y(n) = u(n) + 2 \cdot k_0(n) \cdot u(n-1) + u(n-2) \quad (5.32)$$

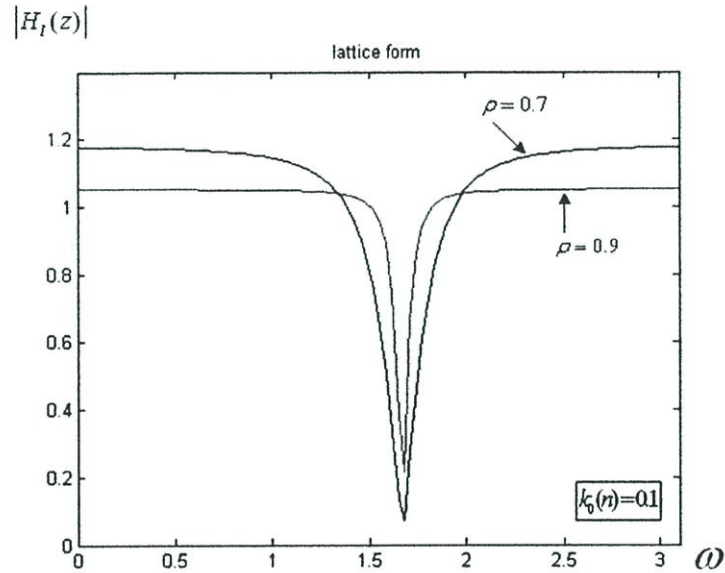
โดยที่

$$u(n) = x(n) - k_0(n) \cdot (1 + \rho) \cdot u(n-1) - \rho \cdot u(n-2) \quad (5.33)$$

สัญญาณกรเดียนต์ของตัวกรองโครงสร้างรูปแบบแลตทิซ สามารถหาได้จาก $\frac{\partial y(n)}{\partial k_0(n)}$ จะ

ได้

$$g_l(n) \approx 2u(n-1) \quad (5.34)$$



รูปที่ 5.9 ผลตอบสนองทางขนาดต่อความถี่ของตัวกรองแบบนอตช์โครงสร้างแลตทิซเมื่อเปลี่ยนค่าสัมประสิทธิ์ ρ

จากรูปที่ 5.9 แสดงผลตอบสนองทางขนาดต่อความถี่ $|H_l(z)|$ โดยการปรับเปลี่ยนค่าพารามิเตอร์ ρ และให้ $k_0(n) = 0.1$ จะเห็นได้ว่า เมื่อ ρ เข้าใกล้ 1 จะทำให้แบนด์วิดท์ของตัวกรองแคบลง ทำให้ $|H_l(z)|$ มีค่าใกล้เคียงกับผลตอบสนองทางความถี่ในอุดมคติ $k_0(n)$ เป็นค่าพารามิเตอร์ที่ใช้กำหนดความถี่นอตช์

5.3 อะแดปทีฟอัลกอริทึมที่ใช้ในการวิจัย

วิทยานิพนธ์นี้แนะนำให้เสนออะแดปทีฟอัลกอริทึมที่ทำการพัฒนาจากอัลกอริทึมแบบ Modified-MNG [19] คือ อัลกอริทึมแบบ VSS (Variable step-size algorithm) ทั้งสามรูปแบบ เพื่อใช้ในการขจัดสัญญาณเสียงสะท้อนในระบบโทรศัพท์ โดยใช้โครงสร้างแลตทิซนอตซ์ฟิลเตอร์ อะแดปทีฟอัลกอริทึมแต่ละแบบต่างก็มีข้อดีข้อเสียแตกต่างกันไป ดังนี้

5.3.1 อัลกอริทึมสำหรับปรับเสถียรพีชสแอมเปิล [2]

จากสมการอินพุต เอาต์พุต ของตัวกรอง IIR แบบนอตซ์ โครงสร้างแลตทิซ มีดังนี้

$$\begin{aligned} \hat{y}(n) &= u(n) + 2\hat{k}_0(n)u(n-1) + u(n-2) \\ u(n) &= x(n) - \hat{k}_0(n)(1 + \rho)u(n-1) - \rho u(n-2) \end{aligned} \quad (5.35)$$

เมื่อ $\hat{k}_0(n)$ คือ เวกเตอร์สัมประสิทธิ์ของตัวกรองแบบแลตทิซที่เวลา n อัลกอริทึมที่ใช้ในการปรับสัมประสิทธิ์ตัวกรอง [2] ตามสมการนี้คือ

$$k_0(n+1) = k_0(n) - \mu(n) \cdot y(n) \cdot \text{sgn}\left(\frac{g(n)}{1 + |g(n)|}\right) \quad (5.36)$$

$\mu(n)$ คือ ค่าเสถียรพีชที่ปรับค่าตามเวลาได้ $g_l(n)$ คือ สัญญาณเกรเดียนต์ของตัวกรอง IIR นอตซ์ฟิลเตอร์อันดับสอง โครงสร้างแลตทิซมีค่าดังสมการ

$$g_l(n) = \frac{\partial \hat{y}(n)}{\partial \hat{k}_0(n)} = 2u(n-1) \quad (5.37)$$

สมการที่ใช้ในการปรับค่าเสถียรพีชเพื่อควบคุมสัมประสิทธิ์ของตัวกรอง IIR นอตซ์ฟิลเตอร์ กำหนดได้ดังสมการ

$$\mu(n+1) = \alpha(n)\mu(n) + \gamma p^2(n) \quad (5.38)$$

$\alpha(n)$ คือ ค่าประมาณที่ได้จากสัญญาณควบคุมความผิดพลาด $\hat{e}(n)$ และ $p(n)$ คือ ค่าประมาณที่ได้จากสัญญาณควบคุมเอาต์พุต $\hat{y}(n)$

การทำงานของอัลกอริทึม มีการปรับอยู่สองส่วน ส่วนแรกคือการปรับความเร็วในการหาคำตอบ ซึ่งก็คือ ค่า $\mu(n)$ ที่ได้จากการประมาณค่าเฉลี่ยทางเวลาของสัญญาณเอาต์พุต และกำลังงาน

ของสัญญาณเอาต์พุตเพื่อควบคุมกระบวนการปรับตัว กำหนดได้ดังนี้

$$p(n) = \sigma p(n-1) + (1 - \sigma) \hat{y}^2(n) \quad (5.39)$$

ในตอนเริ่มต้น $p^2(n)$ มีค่าสูงสุด ทำให้ฟังก์ชัน $\mu(n)$ มีค่าสูงสุดด้วย และ ประสิทธิภาพการหาคำตอบทำได้รวดเร็ว หลังจากเข้าสู่ใกล้ค่าคำตอบที่ต้องการแล้ว $p^2(n)$ จะมีค่าต่ำสุดและทำให้ฟังก์ชัน $\mu(n)$ มีค่าต่ำสุดด้วย

ส่วนที่สอง คือ การลดผลกระทบของสัญญาณรบกวนที่มีผลต่อการปรับเสถียรไซส์ เพื่อให้ได้คำตอบที่ถูกต้องที่สุด โดยการใช้อัตราสหสัมพันธ์ของสัญญาณความคลาดเคลื่อน $\hat{e}(n) \cdot \hat{e}(n-1)$ ดังนี้

$$\alpha(n) = \varepsilon \alpha(n-1) + (1 - \varepsilon) \hat{e}(n) \cdot \hat{e}(n-1) \quad (5.40)$$

$$\hat{e}(n) = d(n) - \hat{y}(n) \quad (5.41)$$

ซึ่ง γ, ε และ σ เป็นค่าบวกคงที่มีค่าอยู่ระหว่าง 0 ~ 1

$\alpha(n)$ จะทำให้ค่า $\mu(n)$ ถูกต้องมากขึ้นเพื่อใช้ในการปรับตัว ทั้งในช่วงแรก ๆ (ซึ่งมีสัญญาณรบกวนมาก) และจนจบกระบวนการปรับตัวของอัลกอริทึม

5.3.2 อัลกอริทึม VSS ที่นำเสนอแบบที่ 1

การลดข้อผิดพลาดในการปรับเสถียรไซส์ ในอัลกอริทึม VSS ที่นำเสนอแบบที่ 1 มีด้วยกันสองวิธี วิธีการแรกจะใช้ Robust step-size algorithm ในการเพิ่มประสิทธิภาพของระบบ จากสมการปรับค่าเสถียรไซส์ $\mu(n)$ ในสมการที่ (5.38) จะได้สมการใหม่ดังนี้

$$\mu(n+1) = \alpha(n) \mu(n) + \gamma p^2(n) \cdot m^2(n) \quad (5.42)$$

$$m(n) = \lambda m(n-1) + (1 - \lambda) \hat{y}(n) \hat{y}(n-1) \quad (5.43)$$

โดยที่ λ เป็นค่าบวกคงที่มีค่าอยู่ระหว่าง 0 ~ 1

$m(n)$ จะทำให้การปรับค่าเสถียรไซส์ $\mu(n)$ มีความถูกต้องมากยิ่งขึ้น การหาคำตอบเข้าสู่ค่าที่เหมาะสมและไม่มีผลกระทบจากสัญญาณรบกวน โดยใช้ค่าประมาณเวลาเฉลี่ยเอาต์พุต $y(n)y(n-1)$ มาควบคุมการปรับค่าเสถียรไซส์

วิธีการที่สอง จากสมมุติฐานที่ว่า ในสถานะ steady-state สัญญาณความผิดพลาด $\hat{e}(n)$ จะไม่มีความสัมพันธ์กัน (un-correlated) ส่งผลให้ยังมีค่าเฉลี่ยปัลไซค์ค่าเล็ก ๆ อยู่จำนวนหนึ่ง อัลกอริทึมต้องทำซ้ำอีกหลายครั้งกว่าจะได้คำตอบที่แท้จริง เพื่อแก้ปัญหานี้จะทำการปรับค่าเฉลี่ยปัลไซค์ในช่วง steady state ให้ละเอียดยิ่งขึ้น โดยการใช้ค่าประมาณเวลาเฉลี่ยของสัญญาณผิดพลาด $\hat{e}(n) \cdot \hat{e}(n-1)$ และ ค่า MSE $|\hat{e}^2(n)|$ จากสมการที่ (5.40) สามารถเขียนสมการใหม่ได้ดังนี้

$$\alpha(n) = \varepsilon\alpha(n-1) + (1-\varepsilon)\{\hat{e}(n) \cdot \hat{e}(n-1) + \hat{e}^2(n)\} \quad (5.44)$$

หรือ

$$\alpha(n) = \varepsilon\alpha(n-1) + (1-\varepsilon)\{\hat{e}(n)[\hat{e}(n-1) + \hat{e}(n)]\} \quad (5.45)$$

5.3.3 อัลกอริทึม VSS ที่นำเสนอแบบที่ 2

จากอัลกอริทึมที่ใช้สำหรับปรับเสถียรปัลไซค์ในรูปแบบที่ 1 หัวข้อที่ผ่านมาเพื่อเพิ่มสมรรถนะการจัดสัญญาณสะท้อนในระบบโทรศัพท์ ทำได้โดยการปรับสัญญาณเกรเดียนต์ของตัวกรองฟิลเตอร์อันดับสองใหม่ โดยใช้ผลต่างระหว่างอินพุตในอดีตและผลคูณของเอาต์พุตในอดีตกับค่า ρ ทำให้คำตอบที่ได้ มีความถูกต้องมากยิ่งขึ้น ตามสมการ

$$g(n) = x(n-1) - \rho\hat{y}(n-1) \quad (5.46)$$

ส่วนอัลกอริทึมอื่น ๆ ใช้สมการเช่นเดียวกันกับอัลกอริทึมที่นำเสนอแบบที่ 1

5.3.4 อัลกอริทึม VSS ที่นำเสนอแบบที่ 3

โดยการใช้เทคนิคของการปรับแบนด์วิดท์ของนอตซ์ฟิลเตอร์ให้เปลี่ยนแปลงตามเวลา ในช่วงแรกแบนด์วิดท์ของนอตซ์ฟิลเตอร์ จะปรับให้กว้างไว้ก่อน การปรับตัวของอัลกอริทึมยังห่างไกลจากคำตอบ เมื่ออัลกอริทึมปรับตัวเข้าสู่จุดที่เหมาะสมแล้ว นอตซ์ฟิลเตอร์ก็จะปรับตัวให้แบนด์วิดท์แคบลง สมการของการปรับแบนด์วิดท์ จะใช้สมการคล้ายกับสมการของการปรับสัมประสิทธิ์ของตัวกรอง ดังสมการ

$$\rho(n+1) = \rho(n) - \mu(n) \cdot \hat{y}(n) \cdot \text{sgn} \left(\frac{g_\rho(n)}{1 + |g_\rho(n)|} \right) \quad (5.47)$$

เมื่อสัญญาณเกรเดียนต์ $g_\rho(n)$ ที่ใช้ในการปรับแบนด์วิดท์ คือ

$$g_\rho(n) = \frac{\partial u(n)}{\partial \rho(n)} \approx -\hat{k}_0(n)u(n-1) + u(n-2) \quad (5.48)$$

5.3.5 สรุปสมการที่ใช้ในการจำลองการทำงาน

อัลกอริทึมแบบเดิม

$$\hat{y}(n) = u(n) + 2\hat{k}_0(n)u(n-1) + u(n-2)$$

$$u(n) = x(n) - \hat{k}_0(n)(1 + \rho)u(n-1) - \rho u(n-2)$$

$$k_0(n+1) = k_0(n) - \mu(n) \cdot y(n) \cdot \operatorname{sgn}\left(\frac{g(n)}{1 + |g(n)|}\right)$$

$$g(n) = \frac{\partial \hat{y}(n)}{\partial \hat{k}_0(n)} = 2u(n-1)$$

$$\mu(n+1) = \alpha(n)\mu(n) + \gamma p^2(n)$$

$$p(n) = \sigma p(n-1) + (1 - \sigma)\hat{y}^2(n)$$

$$\alpha(n) = \varepsilon\alpha(n-1) + (1 - \varepsilon)\hat{e}(n) \cdot \hat{e}(n-1)$$

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

อัลกอริทึม VSS ที่นำเสนอแบบที่ 1

$$\hat{y}(n) = u(n) + 2\hat{k}_0(n)u(n-1) + u(n-2)$$

$$u(n) = x(n) - \hat{k}_0(n)(1 + \rho)u(n-1) - \rho u(n-2)$$

$$k_0(n+1) = k_0(n) - \mu(n) \cdot y(n) \cdot \operatorname{sgn}\left(\frac{g(n)}{1 + |g(n)|}\right)$$

$$g(n) = \frac{\partial \hat{y}(n)}{\partial \hat{k}_0(n)} = 2u(n-1)$$

$$* \mu(n+1) = \alpha(n)\mu(n) + \gamma p^2(n) \cdot m^2(n)$$

$$** m(n) = \lambda m(n-1) + (1 - \lambda)\hat{y}(n)\hat{y}(n-1)$$

$$p(n) = \sigma p(n-1) + (1 - \sigma)\hat{y}^2(n)$$

$$* \alpha(n) = \varepsilon\alpha(n-1) + (1 - \varepsilon)\{\hat{e}(n)[\hat{e}(n-1) + \hat{e}(n)]\}$$

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

อัลกอริทึม VSS ที่นำเสนอแบบที่ 2

$$\hat{y}(n) = u(n) + 2\hat{k}_0(n)u(n-1) + u(n-2)$$

$$u(n) = x(n) - \hat{k}_0(n)(1 + \rho)u(n-1) - \rho u(n-2)$$

$$k_0(n+1) = k_0(n) - \mu(n) \cdot y(n) \cdot \operatorname{sgn}\left(\frac{g(n)}{1 + |g(n)|}\right)$$

$$** g(n) = x(n-1) - \rho \cdot \hat{y}(n-1)$$

$$* \mu(n+1) = \alpha(n)\mu(n) + \gamma p^2(n) \cdot m^2(n)$$

$$** m(n) = \lambda m(n-1) + (1 - \lambda)\hat{y}(n)\hat{y}(n-1)$$

$$p(n) = \sigma p(n-1) + (1 - \sigma)\hat{y}^2(n)$$

$$* \alpha(n) = \varepsilon \alpha(n-1) + (1 - \varepsilon)\{\hat{e}(n)[\hat{e}(n-1) + \hat{e}(n)]\}$$

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

อัลกอริทึม VSS ที่นำเสนอแบบที่ 3

$$\hat{y}(n) = u(n) + 2\hat{k}_0(n)u(n-1) + u(n-2)$$

$$u(n) = x(n) - \hat{k}_0(n)(1 + \rho)u(n-1) - \rho u(n-2)$$

$$k_0(n+1) = k_0(n) - \mu(n) \cdot y(n) \cdot \operatorname{sgn}\left(\frac{g(n)}{1 + |g(n)|}\right)$$

$$** g_p(n) = -\hat{k}_0(n) \cdot u(n-1) + u(n-2)$$

$$** \rho(n+1) = \rho(n) - \mu(n) \cdot \hat{y}(n) \cdot \operatorname{sgn}\left(\frac{g_p(n)}{1 + |g_p(n)|}\right)$$

$$** g(n) = x(n-1) - \rho \cdot \hat{y}(n-1)$$

$$* \mu(n+1) = \alpha(n)\mu(n) + \gamma p^2(n) \cdot m^2(n)$$

$$** m(n) = \lambda m(n-1) + (1 - \lambda)\hat{y}(n)\hat{y}(n-1)$$

$$* p(n) = \sigma p(n-1) + (1 - \sigma)\hat{y}^2(n)$$

$$* \alpha(n) = \varepsilon \alpha(n-1) + (1 - \varepsilon)\{\hat{e}(n)[\hat{e}(n-1) + \hat{e}(n)]\}$$

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

หมายเหตุ

*สมการที่มีการเปลี่ยนแปลง

**สมการเพิ่มเติมใหม่

5.4 สรุป

ในบทนี้ได้กล่าวถึงตัวกรองแบบอะแคปทีฟ IIR แบบนอตซ์อันดับสอง ที่ใช้ในวิทยานิพนธ์ โดยเริ่มจากแนวความคิดของการขจัดสัญญาณรบกวนแบบอะแคปทีฟ และการประยุกต์ใช้งานในการขจัดสัญญาณสะท้อนในระบบโทรศัพท์ พื้นฐานของตัวกรองแบบอะแคปทีฟ คือการนำเอาสัญญาณความคลาดเคลื่อนของตัวกรองมาใช้เพื่อปรับสัมประสิทธิ์ของตัวกรองอีกครั้งหนึ่ง สิ่งที่สำคัญของตัวกรองแบบอะแคปทีฟคืออัลกอริทึมในการปรับตัว คุณสมบัติของอะแคปทีฟอัลกอริทึมที่มีประสิทธิภาพคือสามารถปรับตัวได้รวดเร็ว และถูกต้อง สามารถติดตามสัญญาณที่เปลี่ยนแปลงได้ มีความทนทานต่อการรบกวน มีความซับซ้อนในการคำนวณน้อย รวมถึงโครงสร้างของอัลกอริทึมที่ดีด้วย โครงสร้างที่น่าสนใจของตัวกรองแบบอะแคปทีฟคือโครงสร้างรูปแบบแลตทิซ ที่สามารถลดจำนวนของการคำนวณไปได้ ทำให้การทำงานและการควบคุมประสิทธิภาพของอัลกอริทึมง่ายขึ้น อัลกอริทึมที่นำเสนอในวิทยานิพนธ์เป็นการนำเอาอัลกอริทึมแบบ Modified MNG มาทำการปรับเปลี่ยนการประมาณค่าก้าวกระโดดใหม่ทั้งหมด สามรูปแบบ ซึ่งให้ความรวดเร็วในการปรับตัวดีกว่าอัลกอริทึมแบบเดิมมาก และทำให้ค่า variance bias และ MSE ลดลง สามารถลดสัญญาณรบกวนและ ขจัดสัญญาณสะท้อนกลับในระบบโทรศัพท์ได้เป็นอย่างดี

บทที่ 6

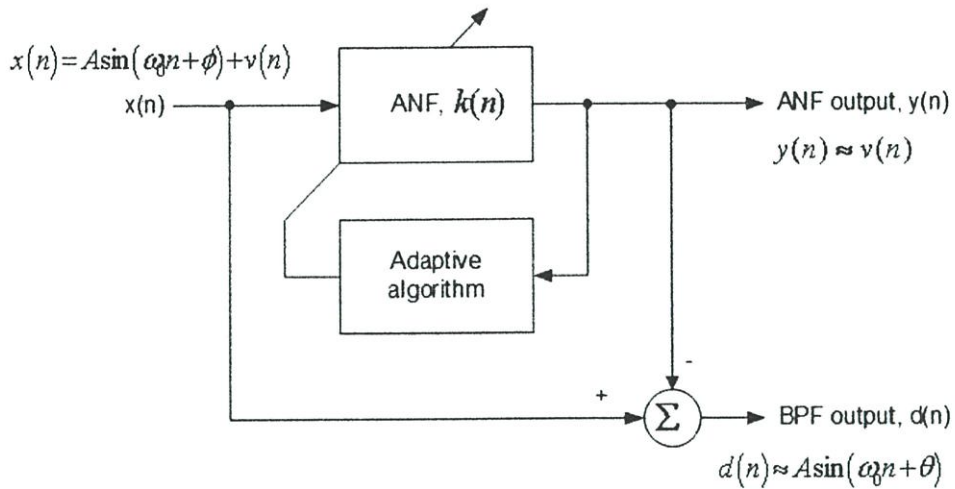
ผลการวิจัย

ในบทนี้จะกล่าวถึง ผลการทดสอบสมรรถนะอะแดปทีฟอัลกอริทึม จากที่ได้นำเสนอไว้ในบทที่ 5 โดยเปรียบเทียบการทำงานของอัลกอริทึมแบบ Variable step size เดิม [2] กับ อัลกอริทึมใหม่ที่ได้นำเสนอทั้งสามแบบ การทดสอบการทำงานโดยการจำลองการทำงานด้วยโปรแกรม MATLAB โดยแบ่งการทดสอบออกเป็นสองขั้นตอน เริ่มจากการทดสอบด้วยสัญญาณชานันด์ที่มีความถี่เท่ากับ $\pi/3$ เรเดียน หรือ มีค่าประมาณ 1 kHz และมีขนาดของสัญญาณเท่ากับหนึ่งหน่วย ($A=1$) เพื่อทดสอบการปรับตัวของสัมประสิทธิ์ตัวกรอง (Filter coefficient) และวัดค่าความผิดพลาดในการหาคำตอบ (Misadjustment) จากค่าความแปรปรวนของสัมประสิทธิ์ตัวกรอง ค่า bias และ ค่าความคลาดเคลื่อนเฉลี่ยกำลังสอง MSE (Mean square error) ส่วนขั้นตอนที่สองแสดงผลการทดสอบ ด้วยสัญญาณสะท้อนจำลอง [7] โดยการวัดค่า ERLE (Echo return loss enhancement) ซึ่งเป็นการวัดประสิทธิภาพของการขจัดสัญญาณสะท้อน ในที่นี้ คือการเปรียบเทียบประสิทธิภาพของอัลกอริทึมที่ออกแบบไว้ทั้งสามรูปแบบเพื่อขจัดสัญญาณสะท้อนในระบบโทรศัพท์โดยใช้ตัวกรองโครงสร้างแลตทิซแบบนอติซ์

อัลกอริทึมที่ดีควรมีประสิทธิภาพในการปรับตัวเพื่อหาคำตอบของสัมประสิทธิ์ได้อย่างรวดเร็ว มีความถูกต้อง และทนทานต่อสัญญาณรบกวนได้ดี

6.1 การทดสอบระบบด้วยสัญญาณชานันด์เคลื่อนเดียวในสัญญาณรบกวนแบบเกาส์เซียน

หัวข้อนี้ จะเป็นการทดสอบการปรับตัวของตัวกรองอะแดปทีฟ ในการประมาณค่าสัญญาณชานันด์ที่มีความถี่ เท่ากับ $\pi/3$ เรเดียน หรือ มีค่าประมาณ 1 kHz ที่ปนมากับสัญญาณรบกวนแบบเกาส์เซียน (White Gaussian noise, WGN) โดยจะทำการเปรียบเทียบผลการทำงานของอะแดปทีฟอัลกอริทึมทั้งสี่รูปแบบในโครงสร้างแบบแลตทิซ โดยใช้เงื่อนไขในการทดสอบที่ดีที่สุดของแต่ละอัลกอริทึมมาเปรียบเทียบกัน ซึ่งจะช่วยให้สามารถเปรียบเทียบสมรรถนะของอะแดปทีฟอัลกอริทึมแต่ละแบบได้อย่างชัดเจน บล็อกไดอะแกรมสำหรับการประมาณค่าความถี่ แสดงได้ดังรูปที่ 6.1



รูปที่ 6.1 บล็อกโคอะแดมสำหรับประมาณค่าสัญญาณชานน์โดยใช้ ANF

จากรูปที่ 6.1 $x(n)$ คือสัญญาณอินพุต ประกอบด้วยสัญญาณชานน์ความถี่เดียว และสัญญาณรบกวนสามารถกำหนดได้ตามสมการ

$$x(n) = A \sin(\omega_0 n + \phi) + v(n) \quad (6.1)$$

เมื่อ A แทนขนาดของสัญญาณชานน์

ω_0 เป็นความถี่จិតอลที่ไม่ทราบค่า มีค่าอยู่ในช่วง $[0, \pi]$

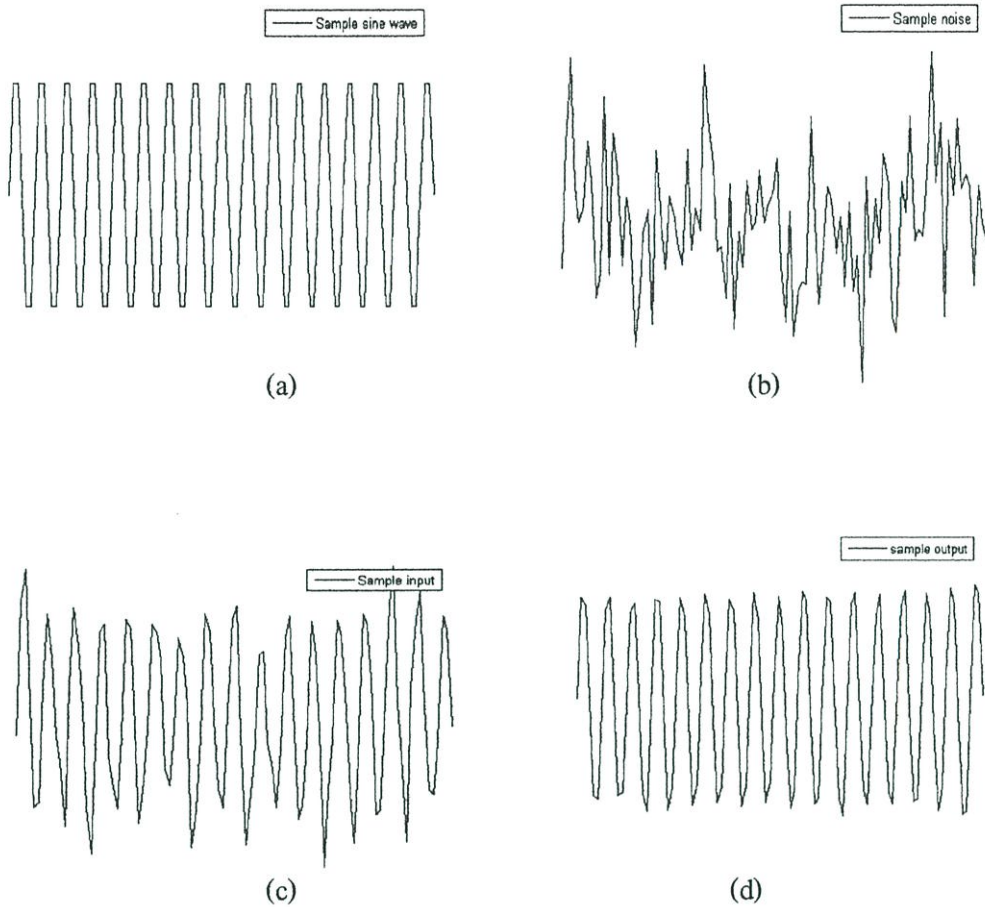
ϕ แทนเฟสของสัญญาณมีค่าอยู่ในช่วง $[0, 2\pi]$

$v(n)$ คือสัญญาณรบกวนแบบเกาส์เซียน

$y(n)$ เป็นสัญญาณเอาต์พุตของ ANF ซึ่งหลังจากเข้าสู่แล้ว $y(n) \approx v(n)$ ความถี่ที่ประมาณได้จะดูได้จากสัมประสิทธิ์ $k(n)$ ของตัวกรอง ซึ่งสัมประสิทธิ์ของตัวกรอง ควรจะเข้าสู่ค่า $-\cos(\omega_0)$

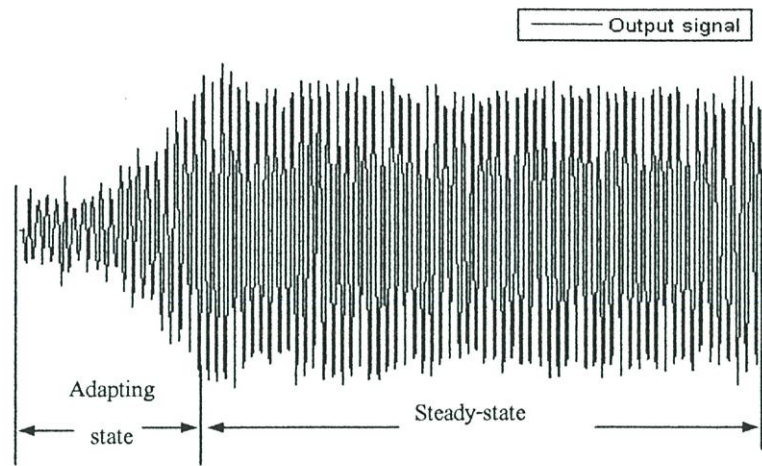
$d(n)$ เป็นสัญญาณเอาต์พุตของ Band-pass filter (BPF) ถ้าทำการประมาณได้อย่างสมบูรณ์ จะได้ $d(n) \approx A \sin(\omega_0 n + \theta)$

ลักษณะของสัญญาณต่าง ๆ ในช่วงเวลาหนึ่ง ๆ ที่ใช้ในการจำลองผลการประมาณค่าความถี่สัญญาณชานน์แสดงได้ดังรูปที่ 6.2



รูปที่ 6.2 (a) ลักษณะของสัญญาณไซน์ (b) สัญญาณรบกวน (c) สัญญาณอินพุตที่ใช้สำหรับจำลองทำงานและ (d) สัญญาณเอาต์พุตที่เป็นสัญญาณไซน์ที่ถูกประมวลแล้ว

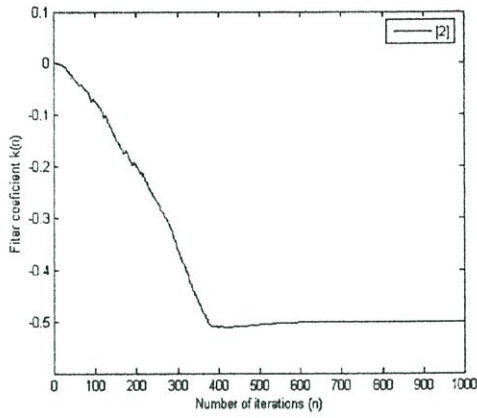
ในรูปที่ 6.2 (a) เป็นลักษณะของสัญญาณไซน์ที่มีความถี่เท่ากับ $\pi/3$ เรเดียน และมีขนาดของสัญญาณเท่ากับหนึ่งหน่วย ($A = 1$) ในรูปที่ 6.2 (b) เป็นลักษณะของสัญญาณรบกวนแบบเกาส์เซียน (Gaussian noise, WGN) ที่มี (Signal-to-noise ratio) หรือ $SNR = 10$ dB ในรูปที่ 6.2 (c) สัญญาณไซน์ที่ถูกรบกวนด้วยสัญญาณรบกวนที่ใช้เป็นอินพุตของระบบ และรูปที่ 6.2 (d) สัญญาณเอาต์พุตซึ่งเป็นสัญญาณไซน์ที่ถูกประมวลแล้วจากการประมวลค่าสัมประสิทธิ์ของอัลกอริทึม ความสมบูรณ์ของสัญญาณที่ถูกประมวลแล้วจะขึ้นอยู่กับคุณภาพของการประมวลค่าของแต่ละอัลกอริทึม



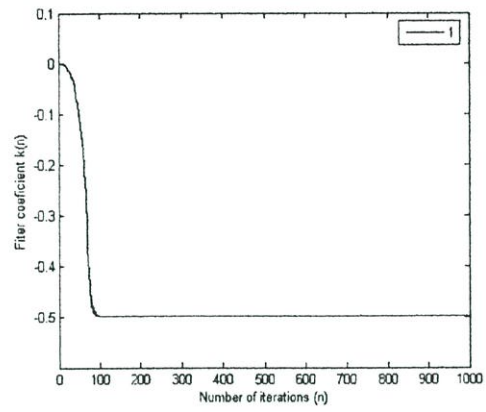
รูปที่ 6.3 ลักษณะของสัญญาณเอาต์พุตขณะเกิดการปรับตัวของอัลกอริทึม

โดยที่รูปที่ 6.3 แสดงลักษณะของสัญญาณเอาต์พุตในขณะที่เกิดการปรับตัวของอัลกอริทึม จากรูปที่ได้จะเห็นว่าการทำงานของระบบจะมีช่วงการทำงานที่สำคัญสองช่วง คือ ในช่วงแรก (Adapting state) จะเป็นการเริ่มปรับตัวของตัวกรองเพื่อหาสัญญาณชาแนลที่ปนอยู่ในสัญญาณรบกวน อัลกอริทึมจะทำการปรับตัวเพื่อปรับสัมประสิทธิ์ของตัวกรองให้เอาต์พุตเข้าใกล้ค่าที่แท้จริงมากที่สุดและรวดเร็วที่สุด และช่วงที่สอง (Steady-state) เมื่อการประมาณค่าของอัลกอริทึมเข้าใกล้ค่าที่แท้จริงแล้ว ตัวกรองจะปรับสัมประสิทธิ์เพียงเล็กน้อยความแปรปรวนของสัมประสิทธิ์ก็ควรจะน้อยลงด้วย

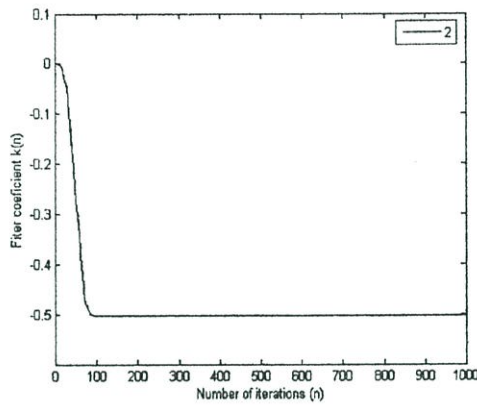
การกำหนดตัวแปรสำหรับทดสอบจำลองการทำงานของอัลกอริทึมที่นำเสนอนี้ ความถี่ของสัญญาณชาแนลเท่ากับ $\omega_0 = \pi/3$ เรเดียน ขนาดของสัญญาณชาแนลได้กำหนดให้เท่ากับหนึ่งหน่วย ($A = 1$) และถูกรบกวนด้วยสัญญาณรบกวนแบบเกาส์เซียน ที่มีอัตราส่วนของสัญญาณต่อสัญญาณรบกวน SNR = 10 dB สำหรับโครงสร้างแบบแลตทิซที่ใช้ในวิทยานิพนธ์นี้ เมื่อกำหนดให้ $\omega_0 = \pi/3$ เรเดียน แล้ว สัมประสิทธิ์ของตัวกรอง “ $k(n)$ ” จะเข้าสู่ค่า $-\cos(\omega_0) = -0.5$ ซึ่งเป็นคำตอบที่แท้จริง ค่าตัวแปรต่าง ๆ ที่กำหนดสำหรับจำลองการทำงานของอัลกอริทึมมีดังนี้ $\rho = 0.9$, $\sigma = 0.98$, $\lambda = 0.98$, $\alpha = 0.98$, $\gamma = 0.01$, และ $\varepsilon = 0.0001$



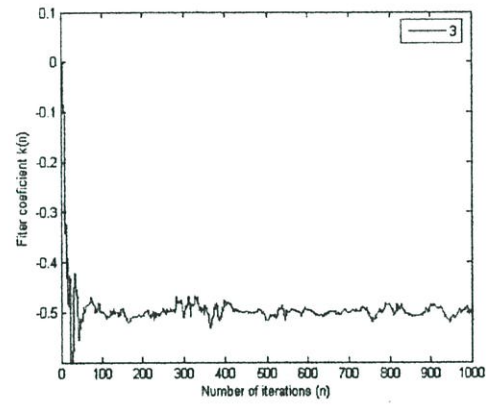
(a)



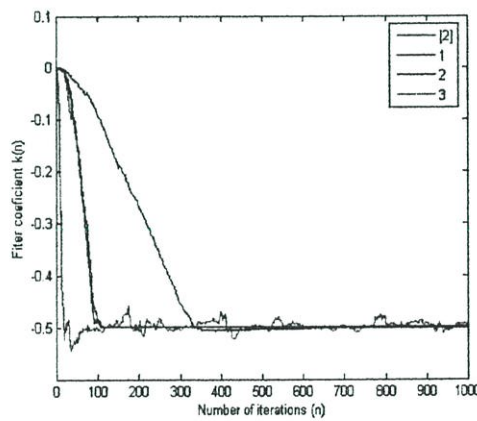
(b)



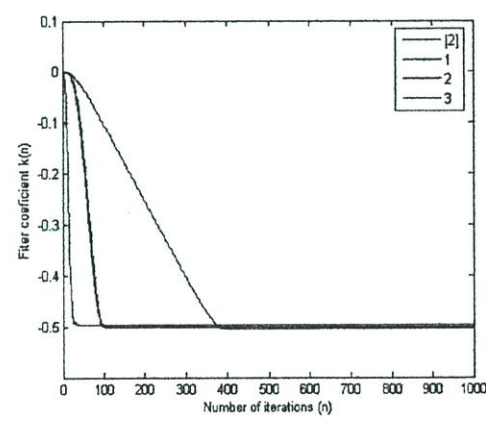
(c)



(d)



(e)



(f)

รูปที่ 6.4 เปรียบเทียบสัมประสิทธิ์ของตัวกรอง “ $k(n)$ ”

จากรูปที่ 6.4 (a) แสดงการปรับสัมประสิทธิ์ตัวกรองของอัลกอริทึมแบบเดิม จะเห็นได้ว่าการเปลี่ยนแปลงช่วงก่อนที่จะเข้าสู่ steady state จะมีการสวิตเล็กน้อย การลู่เข้าสู่คำตอบช้ามากกว่าแบบอื่นจะเห็นว่าอยู่ที่ประมาณ 380 ตัวอย่าง

จากรูปที่ 6.4 (b) และ(c) แสดงการปรับสัมประสิทธิ์ตัวกรองของอัลกอริทึม VSS แบบที่ 1 และ VSS แบบที่ 2 เห็นชัดได้ว่า การปรับตัวมีความรวดเร็วที่ใกล้เคียงกัน การเข้าสู่ค่าตอบอยู่ที่ประมาณ 80 ตัวอย่าง

จากรูปที่ 6.4 (d) แสดงการปรับสัมประสิทธิ์ตัวกรองของอัลกอริทึม VSS แบบที่ 3 จะเห็นได้ว่าการปรับตัวของสัมประสิทธิ์มีความเปลี่ยนแปลงที่รวดเร็ว การหาคำตอบทำได้รวดเร็วมากจากกราฟจะเห็นว่า อยู่ที่ประมาณ 30 ตัวอย่าง แต่เมื่อช่วงเข้าสู่สถานะ steady state การปรับสัมประสิทธิ์ตัวกรองยังคงรุนแรง ไม่สามารถหาคำตอบที่ถูกต้องแน่นอนได้

จากรูปที่ 6.4 (e) แสดงการเปรียบเทียบ อัลกอริทึมทั้งหมดที่ SNR=10 dB

จากรูปที่ 6.4 (f) แสดงการเปรียบเทียบ อัลกอริทึมทั้งหมด ที่ SNR = 20 dB จะเห็นได้ว่าเมื่อ SNR มีค่าเพิ่มขึ้น อัลกอริทึม VSS แบบที่ 3 จะไม่มีการเปลี่ยนแปลงของสัมประสิทธิ์ในช่วง steady state อีกต่อไปและให้ผลคำตอบที่ถูกต้อง

6.2 ความถูกต้องของคำตอบ (Misadjustment)

คุณสมบัติของตัวกรองอะแดปทีฟที่มีประสิทธิภาพคือความถูกต้องของคำตอบ (Misadjustment) สามารถสังเกตได้จากค่า Mean-square error หรือ MSE ซึ่งคำนวณโดยการหาค่าเฉลี่ยเชิงสถิติทั้งหมดของตัวกรองอะแดปทีฟ โดยการใช้วิธี Monte Carlo simulation เพื่อหาค่า variance และ bias อย่างประมาณของสัมประสิทธิ์ของตัวกรอง ด้วยการกำหนดจำนวนความยาวของข้อมูลของแต่ละอัลกอริทึมให้มีค่า $N=1000$ ทำการทดลองซ้ำเป็นจำนวน $M=100$ ครั้ง จากนั้นจะนำค่าที่ได้ในแต่ละครั้ง มาหาค่าเฉลี่ยเชิงสถิติทั้งหมด (Ensemble average) ทำการเปรียบเทียบค่า SNR ต่าง ๆ กัน

สำหรับสมการที่ใช้ในการหาค่า mean ค่า variance ค่า bias และค่า MSE ของสัมประสิทธิ์ของตัวกรองโครงสร้างแบบแลตทิซ “ $k(n)$ ” กำหนดได้ ตามลำดับดังนี้ คือ

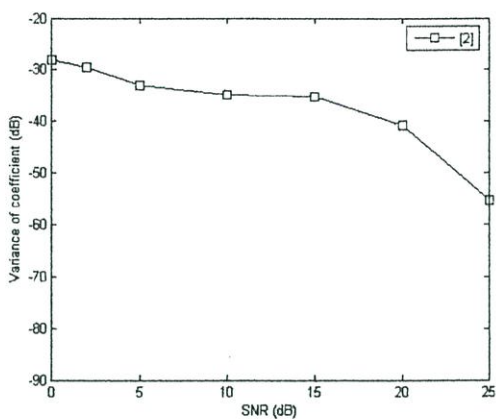
$$E\{k\} = \frac{1}{M} \sum_{i=1}^M k_i(n) \quad ; n = 0, 1, 2, \dots, N \quad (6.6)$$

$$\text{var}(k) = \frac{1}{M} \sum_{i=1}^M (k_i(n) - E(k))^2 ; n = 0, 1, 2, \dots, N \quad (6.7)$$

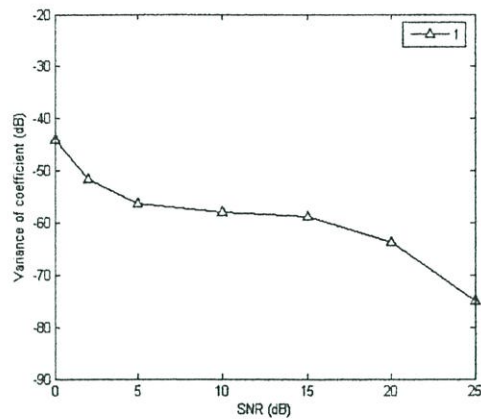
$$\text{bias}(k) = k_0 - E\{k\} \quad (6.8)$$

$$\text{MSE} = \text{var}(k) + \text{bias}^2(k) \quad (6.9)$$

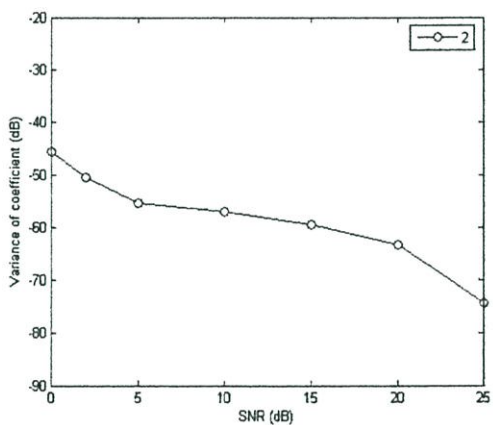
6.2.1 ผลการทดลองจากการวัดค่าความแปรปรวนของสัมประสิทธิ์ (Variance of coefficients)



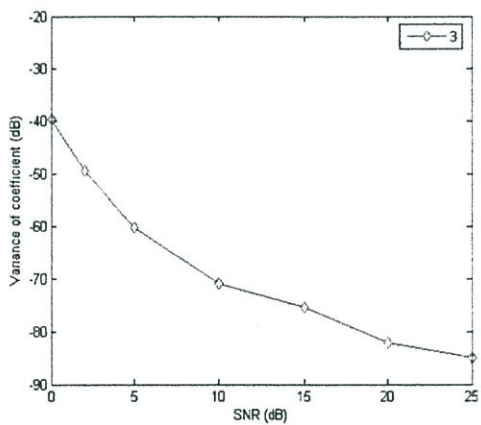
(a)



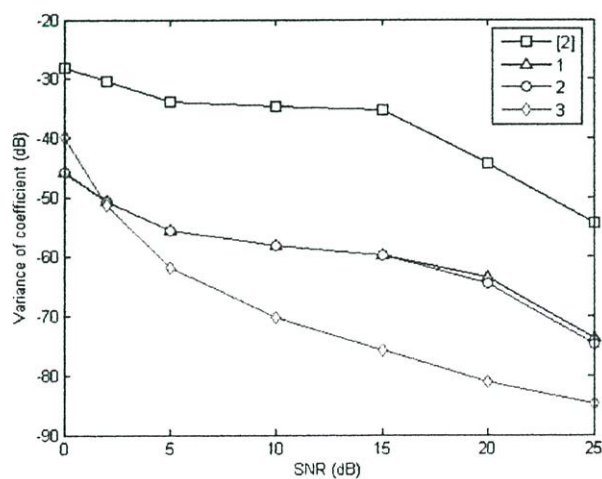
(b)



(c)



(d)



(e)

รูปที่ 6.5 เปรียบเทียบค่า variance ของสัมประสิทธิ์ “ $k(n)$ ” เมื่อ SNR = 0 dB ถึง 25 dB

รูปที่ 6.5 เป็นการวัดความแปรปรวนของสัมประสิทธิ์ที่ $\text{SNR} = 0 \text{ dB}$ ถึง $\text{SNR} = 25 \text{ dB}$ ในสถานะที่มีสัญญาณรบกวนมาก จากรูปที่ 6.5(a) 6.5 (b) และ 6.5(c) จะเห็นชัดเจนว่า ความแปรปรวนของสัมประสิทธิ์มีค่าลดลงอย่างสม่ำเสมอคล้ายกัน แต่มีระดับที่แตกต่างกัน ดังนี้

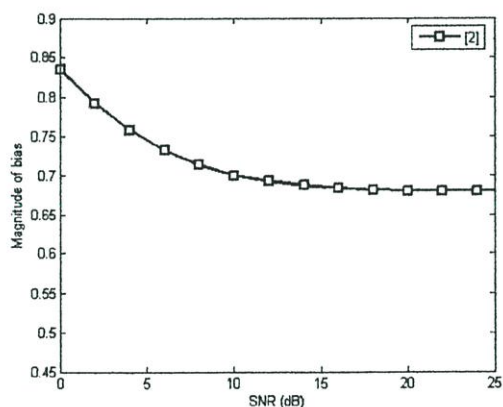
อัลกอริทึมแบบเดิมจากรูปที่ 6.5(a) ความแปรปรวนของสัมประสิทธิ์จะมากที่สุด เริ่มจาก -28 dB ที่ $\text{SNR} = 0 \text{ dB}$ ลดลงอย่างสม่ำเสมอจนถึงประมาณ -55 dB ที่ $\text{SNR} = 25 \text{ dB}$

อัลกอริทึม VSS แบบที่ 1 รูปที่ 6.5(b) และ VSS แบบที่ 2 รูปที่ 6.5(c) เริ่มจาก -45 dB ที่ $\text{SNR} = 0 \text{ dB}$ ลดลงอย่างสม่ำเสมอจนถึงประมาณ -75 dB ที่ $\text{SNR} = 25 \text{ dB}$ ความแปรปรวนของสัมประสิทธิ์มีค่าใกล้เคียงกัน และ ลดลงใกล้เคียงกัน

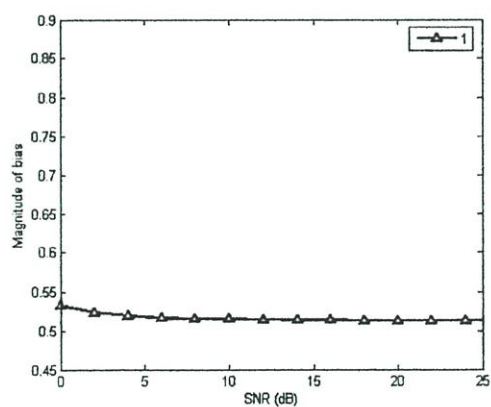
ส่วนอัลกอริทึมสุดท้าย อัลกอริทึม VSS แบบที่ 3 รูปที่ 6.5 (b) ความแปรปรวนของสัมประสิทธิ์มีค่าลดลงอย่างรวดเร็วแบบเอกซ์โปเนนเชียล จาก -40 dB ที่ $\text{SNR} = 0 \text{ dB}$ ลดลงจนถึงประมาณ -85 dB ที่ $\text{SNR} = 25 \text{ dB}$

รูปที่ 6.5 (e) เป็นการเปรียบเทียบค่า variance ของสัมประสิทธิ์ของอัลกอริทึม VSS ทั้งสี่แบบ จะเห็นชัดเจนว่า ความแปรปรวนของสัมประสิทธิ์ มีค่าลดน้อยลงเรื่อย ๆ ตามลำดับ นั่นแสดงว่า อัลกอริทึมที่ได้พัฒนาในบทวิทยานิพนธ์นี้ มีการปรับตัวที่รวดเร็ว ความถูกต้องของคำตอบสูงขึ้นตามลำดับ

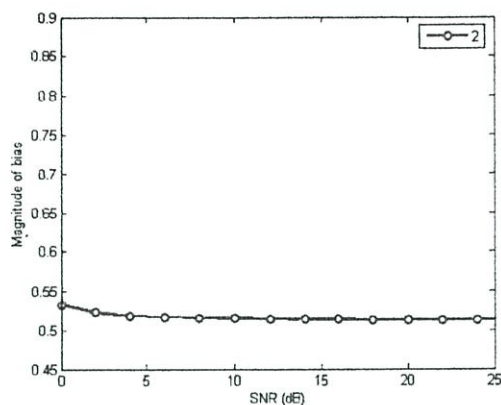
6.2.2 ผลการทดลองจากการวัดค่า bias ของสัมประสิทธิ์ (Magnitude of bias)



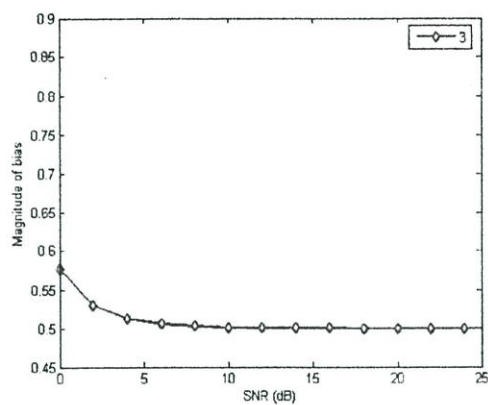
(a)



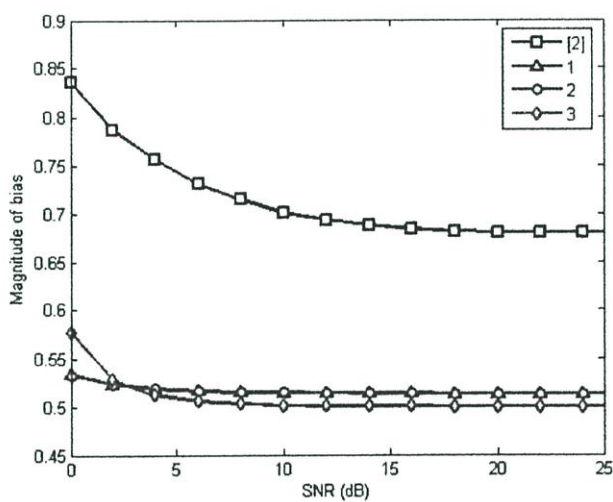
(b)



(c)



(d)

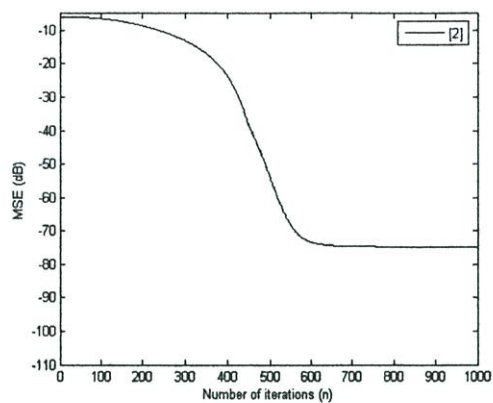


(e)

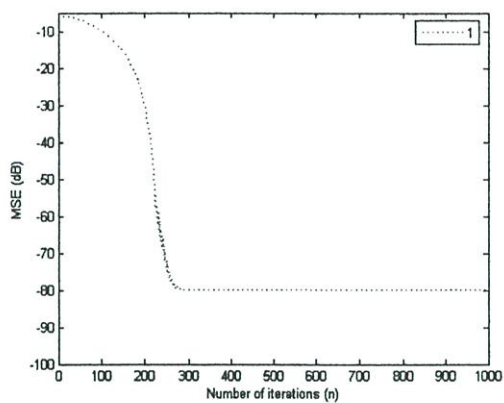
รูปที่ 6.6 เปรียบเทียบค่า bias ของสัมประสิทธิ์ “ $k(n)$ ” เมื่อกำหนด SNR = 0 dB ถึง 25 dB

รูปที่ 6.6 (a) (b) (c) (d) จะเห็นได้ว่า ค่า bias ของอัลกอริทึมทั้งสี่แบบ มีค่าที่ลดลงอย่างสม่ำเสมอใกล้เคียงกัน รูปที่ 6.6(e) จะเห็นชัดเจนว่า ค่า bias ของอัลกอริทึม VSS แบบที่ 1 และ 2 มีค่าเกือบจะเท่ากัน คิดว่าแบบเดิมมาก หมายความว่า จะทนต่อสัญญาณรบกวนได้ดีกว่า ส่วนอัลกอริทึม VSS แบบที่ 3 จะแตกต่างจากแบบที่ 1 และ 2 อยู่บ้าง ในช่วงที่ SNR มีค่าจาก 0 dB ถึงค่าประมาณ 3 dB แสดงให้เห็นได้ว่า อัลกอริทึม VSS แบบที่ 3 ไม่สามารถทนต่อสัญญาณรบกวนได้ดีกว่าแบบที่ 1 และ 2 แต่เมื่อ SNR มีค่าเพิ่มขึ้น มากกว่า 5 dB แล้ว แบบที่ 3 จะมีค่า bias ที่น้อยที่สุด และเป็นอัลกอริทึมที่ดีที่สุด

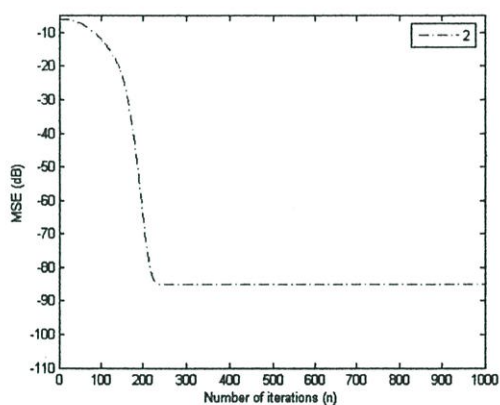
6.2.3 ผลการทดลองจากการวัดค่า MSE (Mean-square error)



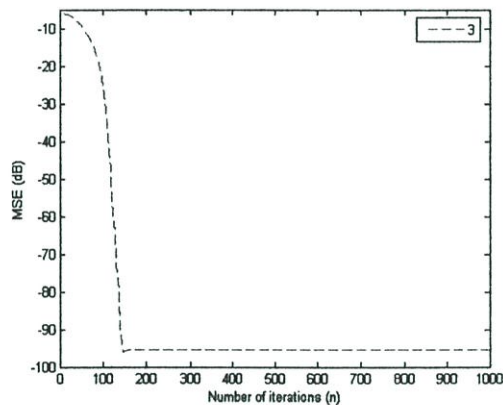
(a)



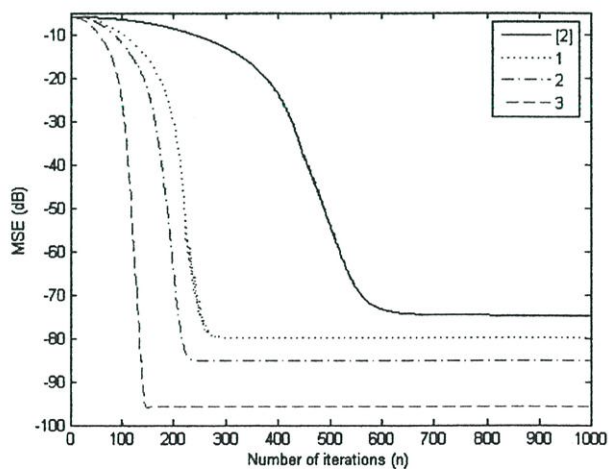
(b)



(c)



(d)



(e)

รูปที่ 6.7 การเปรียบเทียบค่า MSE ของสัมประสิทธิ์ “ $k(n)$ ” เมื่อ SNR = 0 ~ 70 dB

จากรูปที่ 6.7 (a) เป็นกราฟของอัลกอริทึมแบบเดิม พบว่า ค่า MSE ที่ได้มีค่าต่ำกว่าทุกอัลกอริทึม มาก หมายความว่า การหาคำตอบ มีความแม่นยำน้อยที่สุด และใช้จำนวนตัวอย่างมากกว่า 550 ตัวอย่างขึ้นไป ค่า MSE มีค่าน้อยประมาณ -75 dB

จากรูปที่ 6.7 (b) เป็นกราฟของอัลกอริทึม VSS แบบที่ 1 พบว่า ค่า MSE ที่ได้มีค่าสูงกว่าแบบเดิม ค่า MSE มีค่าประมาณ -80 dB การหาคำตอบมีความแม่นยำมากกว่าแบบแรก และมีการลู่เข้าที่รวดเร็วกว่าใช้จำนวนตัวอย่างน้อยกว่าประมาณ 250 ตัวอย่าง

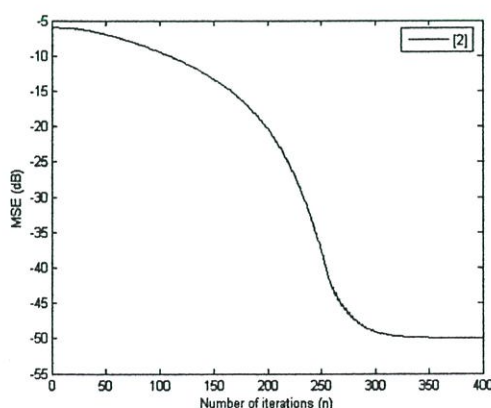
จากรูปที่ 6.7 (c) เป็นกราฟของอัลกอริทึม VSS แบบที่ 2 พบว่า ค่า MSE ที่ได้มีค่าต่ำกว่า VSS แบบที่ 1 และ อัลกอริทึมแบบเดิม ค่า MSE มีค่าประมาณ -85 dB การลู่เข้าใช้จำนวนตัวอย่างน้อยประมาณ 210 ตัวอย่าง การหาคำตอบมีความแม่นยำเพิ่มมากขึ้น

จากรูปที่ 6.7 (d) เป็นกราฟของอัลกอริทึม VSS แบบที่ 3 พบว่า ค่า MSE ที่ได้มีค่าน้อยที่สุด มีค่าประมาณ -95 dB และ การลู่เข้ารวดเร็วกว่าที่สุดเมื่อเทียบกับอัลกอริทึมทั้งหมด ใช้จำนวนตัวอย่างน้อยที่สุดประมาณ 130 ตัวอย่าง การหาคำตอบมีความแม่นยำมากที่สุด

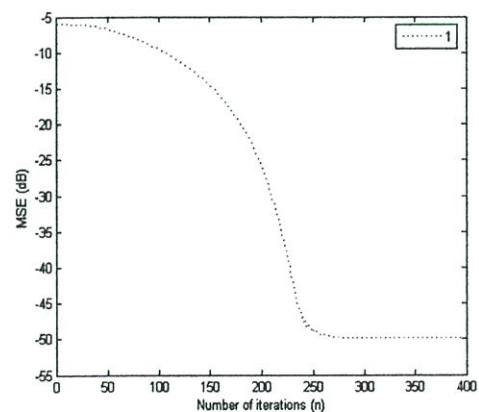
จากรูปที่ 6.7 (e) เปรียบเทียบค่า MSE ของสัมประสิทธิ์ " $k(n)$ " ของอัลกอริทึมทั้งหมด เมื่อนำมาคิดเป็นอัตราส่วนร้อยละเปรียบเทียบความถูกต้องของคำตอบกับอัลกอริทึมเดิม จะได้ว่า

- อัลกอริทึม VSS แบบที่ 1 มีความถูกต้องเพิ่มขึ้น 6.66 เปอร์เซ็นต์
- อัลกอริทึม VSS แบบที่ 2 มีความถูกต้องเพิ่มขึ้น 13.33 เปอร์เซ็นต์
- อัลกอริทึม VSS แบบที่ 3 มีความถูกต้องเพิ่มขึ้น 26.66 เปอร์เซ็นต์

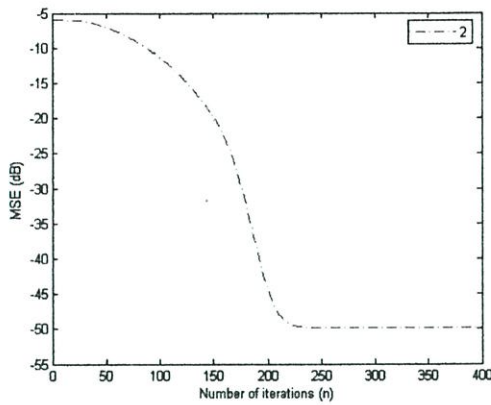
6.2.4 ผลการทดลองจากการวัดค่า MSE (Mean-square error) เมื่อให้ค่า MSE มีค่าใกล้เคียงกันที่ -50 dB



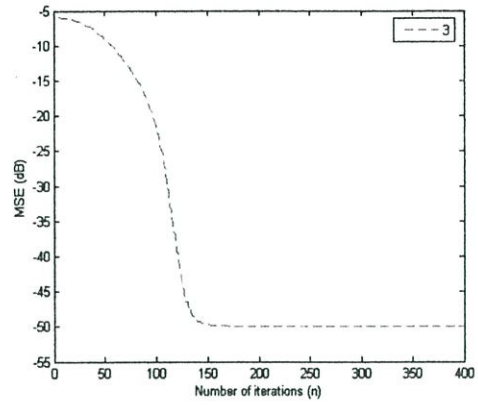
(a)



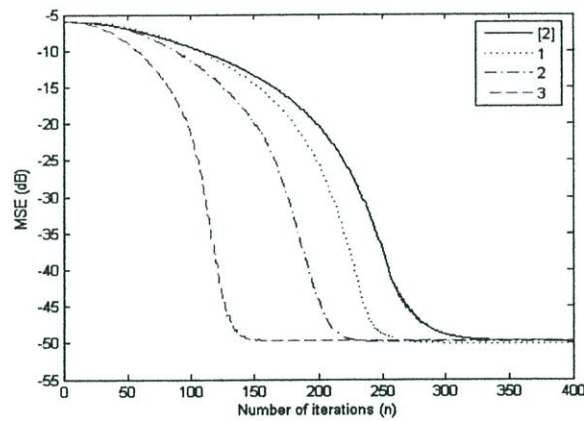
(b)



(c)



(d)



(e)

รูปที่ 6.8 การเปรียบเทียบค่า MSE ของสัมประสิทธิ์ “ $k(n)$ ” เมื่อ SNR = 0 dB ถึง 30 dB และค่า MSE มีค่าใกล้เคียงกันที่ -50 dB

จากรูปที่ 6.8 เป็นการเปรียบเทียบค่า MSE ของสัมประสิทธิ์ “ $k(n)$ ” เมื่อทำการปรับค่า SNR = 0 ~ 30 dB และให้ค่า MSE มีค่าใกล้เคียงกันที่ -50 dB

จากรูปที่ 6.8 (a) เป็นกราฟของอัลกอริทึมแบบเดิม พบว่า ความเร็วในการปรับตัวช้ามากกว่าทุกอัลกอริทึม ใช้จำนวนตัวอย่างที่ประมาณ 320 ตัวอย่างขึ้นไป อัลกอริทึมนี้ถึงจะหาคำตอบได้

จากรูปที่ 6.8 (b) เป็นกราฟของอัลกอริทึม VSS แบบที่ 1 พบว่า ความเร็วในการปรับตัวเพิ่มขึ้นกว่าอัลกอริทึมเดิมมาก ใช้จำนวนตัวอย่างที่ประมาณ 260 ตัวอย่าง สามารถลดจำนวนตัวอย่างที่ใช้คำนวณลงได้ประมาณ 60 ตัวอย่าง

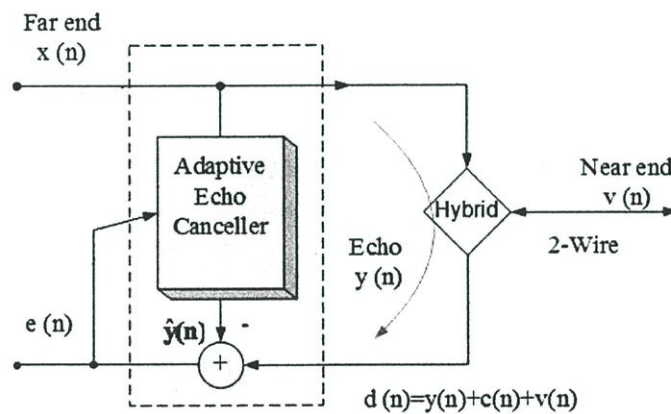
จากรูปที่ 6.8 (c) เป็นกราฟของอัลกอริทึม VSS แบบที่ 2 พบว่า ความเร็วในการปรับตัวเพิ่มขึ้นกว่าอัลกอริทึมเดิมมากขึ้น ใช้จำนวนตัวอย่างที่ประมาณ 225 ตัวอย่าง เมื่อเทียบกับอัลกอริทึม VSS แบบที่ 1 สามารถลดจำนวนตัวอย่างที่ใช้คำนวณลงได้อีกประมาณ 35 ตัวอย่าง

จากรูปที่ 6.8 (d) เป็นกราฟของอัลกอริทึม VSS แบบที่ 3 พบว่า ความเร็วในการปรับตัวเพิ่มขึ้นกว่าอัลกอริทึมเดิมมากที่สุด ใช้จำนวนตัวอย่างที่ประมาณ 140 ตัวอย่าง เมื่อเทียบกับอัลกอริทึม VSS แบบที่ 2 สามารถลดจำนวนตัวอย่างที่ใช้คำนวณลงได้อีกประมาณ 85 ตัวอย่าง

ในรูปที่ 6.8 (e) เปรียบเทียบค่า MSE ของสัมประสิทธิ์ “ $k(n)$ ” ของอัลกอริทึมทั้งหมด เมื่อนำมาคิดเป็นอัตราส่วนร้อยละเปรียบเทียบความเร็วกับอัลกอริทึมเดิม จะได้ว่า

- อัลกอริทึม VSS แบบที่ 1 มีความเร็วเพิ่มขึ้น 18.75 เปอร์เซ็นต์
- อัลกอริทึม VSS แบบที่ 2 มีความเร็วเพิ่มขึ้น 29.69 เปอร์เซ็นต์
- อัลกอริทึม VSS แบบที่ 3 มีความเร็วเพิ่มขึ้น 56.25 เปอร์เซ็นต์

6.3 การประยุกต์ใช้งานในการขจัดสัญญาณเสียงสะท้อนแบบอะแคปทีฟในระบบโทรศัพท์



รูปที่ 6.9 โครงสร้างวงจรขจัดเสียงสะท้อนแบบอะแคปทีฟ

พิจารณาโครงสร้างวงจรขจัดสัญญาณสะท้อนที่ใช้วงจรกรองแบบอะแคปทีฟ ดังแสดงในรูปที่ 6.9 เมื่อ $x(n)$ คือ สัญญาณเสียงสะท้อนที่มาจากปลายทาง (Far end) และ $d(n)$ คือ สัญญาณที่ย้อนกลับไปยังปลายทาง โดยมีสัญญาณสะท้อน $y(n)$ รวมกับสัญญาณรบกวน $c(n)$ และสัญญาณเสียงพูดต้นทาง $v(n)$ สัญญาณสะท้อนจะรบกวนคู่สนทนา หากการหน่วงเวลาในช่วงที่ยาวมาก จำเป็นจะต้องลดทอนสัญญาณสะท้อนให้เหลือน้อยที่สุด เพื่อมิให้คุณภาพของเสียงมีค่าต่ำกว่ามาตรฐานที่ยอมรับได้ ในการขจัดสัญญาณเสียงสะท้อนในระบบโทรศัพท์ จะขจัดสัญญาณเสียงสะท้อนด้านไกล (Far end) ไม่ให้ย้อนออกไปทางภาคส่ง $e(n)$ ได้ โดยใช้อะแคปทีฟฟิลเตอร์ทำการวิเคราะห์คุณลักษณะที่เปลี่ยนแปลงของสัญญาณสะท้อน และสร้างสัญญาณสะท้อน

ที่เหมือนกันชิ้นใหม่ แล้วนำไปหักล้างกับสัญญาณที่จะส่งไปยังปลายทาง $d(n)$ สมการของวงจรจัดสัญญาณสะท้อนโดยทั่วไปแสดงได้ดังนี้

$$\hat{y}(n) = W^T(n)X(n) \quad (6.10)$$

$$e(n) = d(n) - \hat{y}(n) \quad (6.11)$$

$$d(n) = v(n) + y(n) + c(n) \quad (6.12)$$

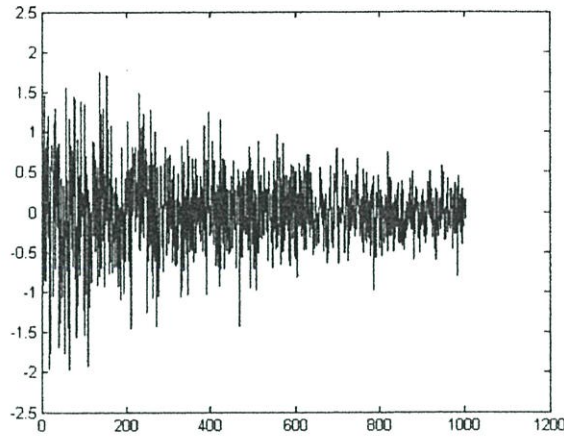
สำหรับ $X(n)$ คือ อินพุตเวกเตอร์ และ $W(n)$ คือ สัมประสิทธิ์เวกเตอร์ของฟิลเตอร์สำหรับวงจรจัดสัญญาณสะท้อนโดยทั่วไป แสดงได้ดังสมการ

$$X(n) = [x(n) \quad x(n-1) \dots \dots \quad x(n-L+1)]^T \quad (6.13)$$

$$W(n) = [w_0(n) \quad w_1(n-1) \dots \dots \quad w_{L-1}(n)]^T \quad (6.14)$$

โครงสร้างแลตทิซมีความเป็นโมดูล ทำให้สามารถเพิ่มหรือลดอันดับของโครงสร้างได้และง่ายต่อการพิจารณาเสถียรภาพ ในวิทยานิพนธ์นี้ได้ใช้ตัวกรองแบบนอกรีตอันดับสอง โครงสร้างแลตทิซ ร่วมกับอัลกอริทึม VSS ทั้งสามแบบ ที่พัฒนามาจากอัลกอริทึมแบบ Modified-MNG การปรับเสถียรไซส์ซึ่งเป็นประโยชน์มาก เมื่อค่าสัมประสิทธิ์ของฟิลเตอร์เข้าใกล้ค่าตอบที่ต้องการ อัลกอริทึมจะปรับค่าเสถียรให้มีค่าน้อย ๆ ทำให้การหาค่าตอบถูกต้องมากขึ้น และ เมื่อค่าของสัมประสิทธิ์ยังห่างไกลจากค่าตอบ อัลกอริทึมจะปรับค่าเสถียรไซส์ให้มีค่ามากขึ้น ทำให้สามารถบรรลุดังค่าตอบได้เร็วขึ้น สัญญาณเสียงสะท้อนในระบบโทรศัพท์ มีเส้นทางของการสะท้อนตามแบบจำลองใน [7] สามารถกำหนดได้จากสมการ

$$Echo \quad path = \sum_{i=0}^n R_i \exp\left[-\frac{(i+1)}{880}\right] \delta(n-i) \quad (6.15)$$



รูปที่ 6.10 สัญญาณจำลอง Echo path จากสมการที่ (6.15)

เมื่อ n คือ จำนวนตัวอย่างของสัญญาณ และ $R(i)$ คือ ค่าสุ่ม ซึ่งมีค่าอยู่ในช่วง $+2$ ถึง -2 และ $\delta(n)$ คือ Dirac function

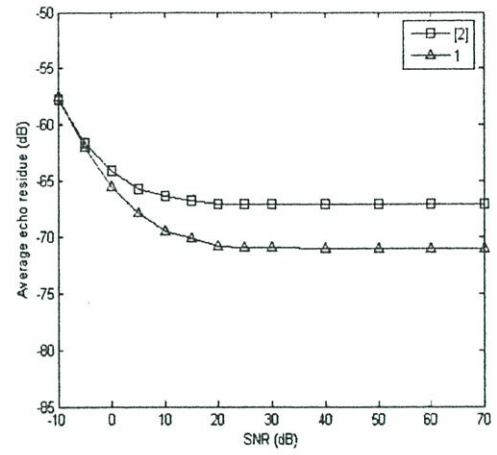
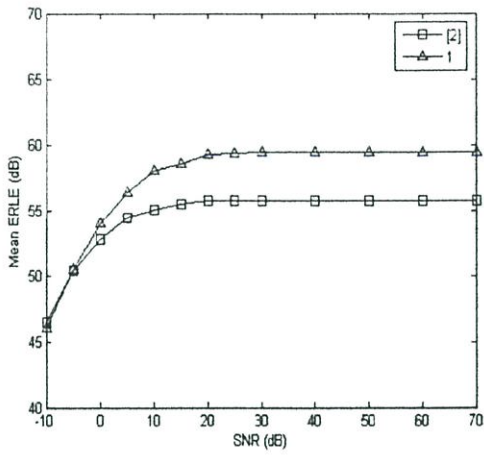
6.3.1 ผลการทดลองการจัดสัญญาณสะท้อนในระบบโทรศัพท์

จากแบบจำลองการจัดสัญญาณสะท้อนในระบบโทรศัพท์ สัญญาณอินพุตที่ได้จากสมการที่ (6.15) ป้อนให้กับระบบ ทำการวัดสมรรถนะในการจัดสัญญาณสะท้อนว่าสามารถจัดสัญญาณสะท้อนได้อย่างสมบูรณ์เพียงใด การวัดสมรรถนะที่นิยมมากที่สุดของวงจรจัดสัญญาณสะท้อนแบบอะแดปทีฟ คือ การวัดค่า ERLE (Echo Return Loss Enhancement) จากสมการดังนี้

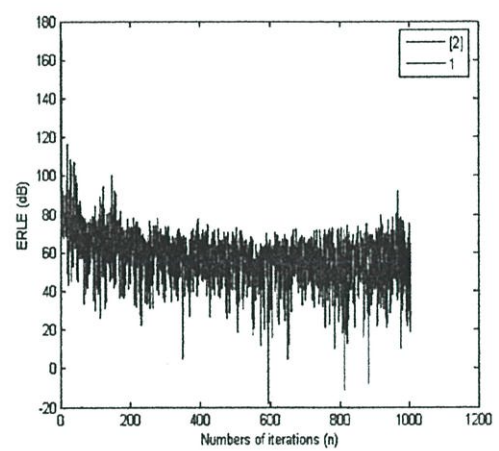
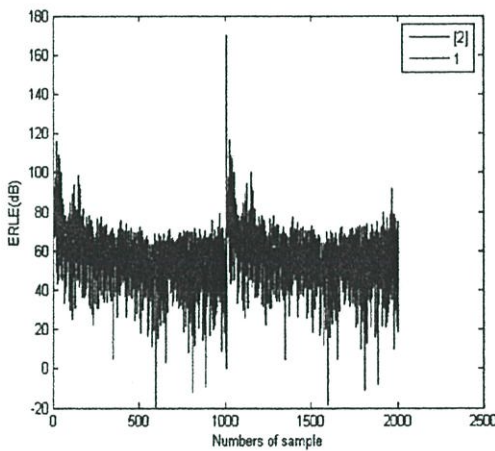
$$\text{ERLE(dB)} = 10 \log_{10} \frac{E[y^2(n)]}{E[e^2(n)]} \quad (6.16)$$

ค่า ERLE คือ อัตราส่วนของสัญญาณ ต่อ สัญญาณรบกวนที่ไม่สามารถจัดออกจากสัญญาณได้ในอีกความหมายหนึ่งก็คือ อัลกอริทึมที่กำหนดเป็นอัตราส่วนกำลังระหว่างสัญญาณสะท้อนจริง ต่อค่าสัญญาณสะท้อนที่ยังคงอยู่ ในวิทยานิพนธ์นี้ได้แสดงค่า ERLE โดยการจำลองการทำงานด้วยโปรแกรมคอมพิวเตอร์ จากการเปรียบเทียบกับอัลกอริทึมเดิม [2] และ อัลกอริทึม VSS ใหม่ ทั้ง 3 รูปแบบ โดยมีการปรับค่าพารามิเตอร์ในการหาคำตอบ ๆ ของแต่ละโครงสร้างให้มีสมรรถนะช่วง steady state มีค่าเท่ากัน โดยกำหนดพารามิเตอร์ต่าง ๆ มีค่าดังนี้ $\hat{k}_0(0) = 0$, $\mu(0) = 0.007$, $\rho(0) = 0.99996$, $\gamma = 0.0000001$, $\sigma = 0.99$, $\varepsilon = 0.0001$, $\lambda = 0.95$ ให้ข้อมูลมีความยาว $N = 1,000$ โดยมีการทดลองทำซ้ำ 1,000 ครั้ง ผลการจำลองการทำงานโดยคอมพิวเตอร์เปรียบเทียบค่า ERLE ระหว่าง 4 อัลกอริทึม

6.3.1.1 การจำลองอัลกอริทึม VSS ที่นำเสนอแบบที่ 1 เปรียบเทียบกับอัลกอริทึมเดิม [2]



(a) Mean ERLE (SNR= -10 to 70 dB) (b) Average echo residue (SNR= -10 to 70 dB)



(c) Tracking (SNR= 70 dB) (d) ERLE (SNR= 70 dB)

รูปที่ 6.11 ผลการจำลองอัลกอริทึม VSS ที่นำเสนอแบบที่ 1 เปรียบเทียบกับอัลกอริทึมเดิม [2]

รูปที่ 6.11 (a) แสดงการเปรียบเทียบวัดหาค่าเฉลี่ย (mean ERLE) ที่ SNR= -10 to 70 dB จะเห็นได้ว่า อัลกอริทึม VSS ที่นำเสนอแบบที่ 1 ให้ผลที่ดีกว่าอัลกอริทึมแบบเดิมเมื่อ SNR > -5 dB ขึ้นไป และมีค่ามากกว่าอย่างคงที่เมื่อ SNR > 20 dB โดยค่าเฉลี่ย (mean ERLE) จะอยู่ที่ 59 dB ขณะที่อัลกอริทึมเดิมมีค่าอยู่ที่ประมาณ 56 dB ให้ผลดีกว่าอัลกอริทึมแบบเดิมประมาณ 3 dB

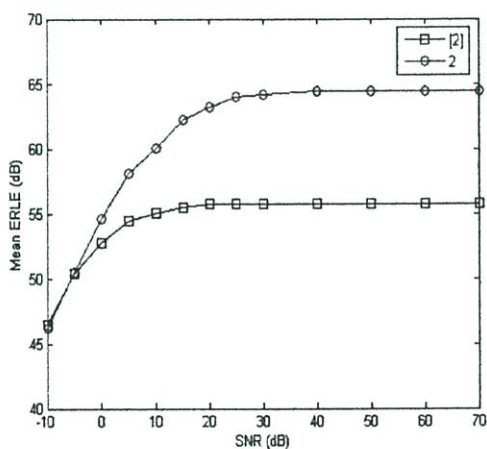
รูปที่ 6.11 (b) แสดงค่าเฉลี่ยสะสมสัญญาณสะท้อนที่เหลือจากการขจัดในระบบ (Average echo residue) ที่ SNR = -10 to 70 dB จะเห็นได้ว่า อัลกอริทึม VSS ที่นำเสนอแบบที่ 1 ยังคงให้ผลที่ดีกว่าอัลกอริทึมแบบเดิมเมื่อ SNR > -5 dB ขึ้นไป และมีค่าน้อยกว่าอย่างคงที่เมื่อ SNR > 20 dB

ขณะที่อัลกอริทึมเดิมมีค่าเฉลี่ยสะสมสัญญาณสะท้อนที่เหลือจากการขจัดในระบบ อยู่ที่ประมาณ -67 dB อัลกอริทึม VSS ที่นำเสนอแบบที่ 1 มีค่าอยู่ประมาณ -71 dB ซึ่งให้ผลที่ดีกว่าอัลกอริทึมเดิมประมาณ 4 dB

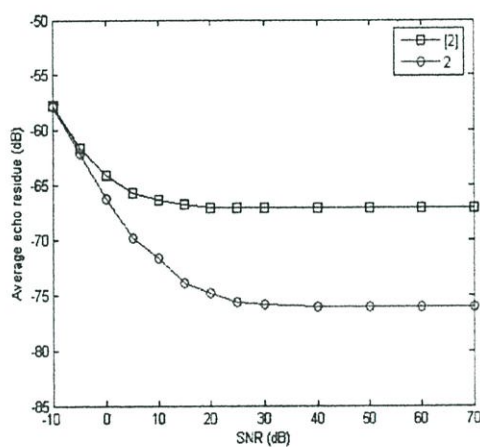
รูปที่ 6.11 (c) แสดงการติดตามสัญญาณของอัลกอริทึม (Tracking) ที่ $\text{SNR} = 70$ dB เห็นได้ว่าอัลกอริทึมอัลกอริทึม VSS ที่นำเสนอแบบที่ 1 มีความสามารถในการติดตามสัญญาณสะท้อนได้ดี

รูปที่ 6.11 (d) แสดงค่า ERLE ที่ $\text{SNR} = 70$ dB ในสภาวะที่มีสัญญาณรบกวนน้อยมาก ๆ จะเห็นได้ว่าอัลกอริทึมอัลกอริทึม VSS ที่นำเสนอแบบที่ 1 จะเริ่มมีค่าสูงกว่าอัลกอริทึมเดิมเมื่อจำนวนตัวอย่างมากกว่า 500 ตัวอย่างขึ้นไป

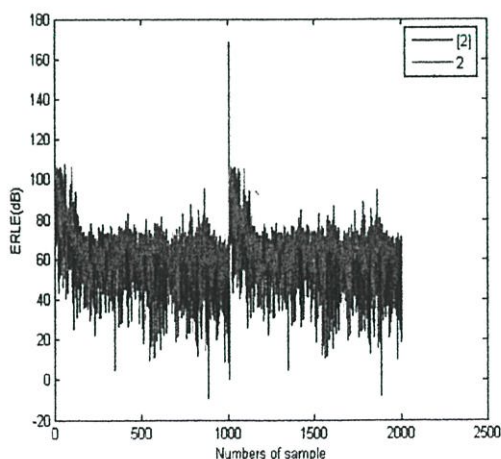
6.3.1.2 การจำลองอัลกอริทึม VSS ที่นำเสนอแบบที่ 2 เปรียบเทียบกับอัลกอริทึมเดิม [2]



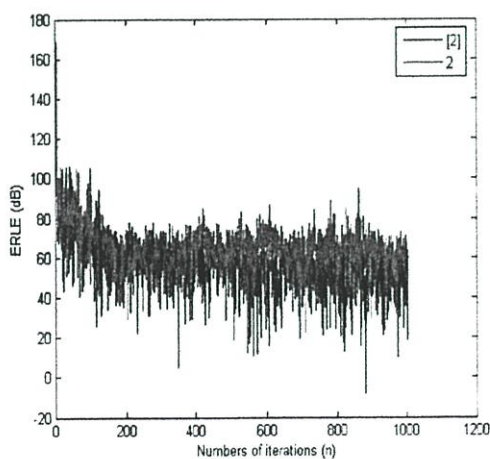
(a) Mean ERLE (SNR= -10 to 70 dB)



(b) Average echo residue (SNR= -10 to 70 dB)



(c) Tracking (SNR= 70 dB)



(d) ERLE (SNR= 70 dB)

รูปที่ 6.12 ผลการจำลองอัลกอริทึม VSS ที่นำเสนอแบบที่ 2 เปรียบเทียบกับอัลกอริทึมเดิม [2]

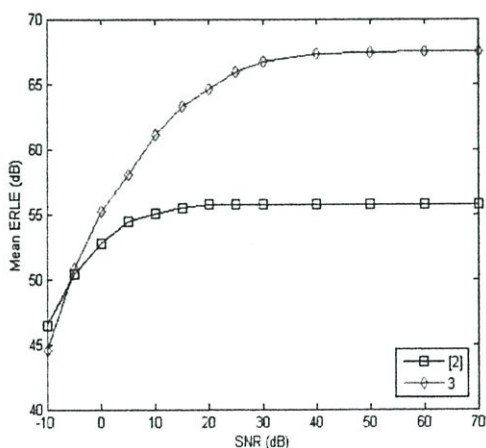
รูปที่ 6.12 (a) แสดงการเปรียบเทียบวัดหาค่าเฉลี่ย (mean ERLE) ที่ SNR = -10 to 70 dB จะเห็นได้ว่า อัลกอริทึม VSS ที่นำเสนอแบบที่ 2 ให้ผลที่ดีกว่าอัลกอริทึมแบบเดิมเมื่อ SNR > -5 dB ขึ้นไป และมีค่ามากกว่าอย่างคงที่เมื่อ SNR > 20 dB โดยค่าเฉลี่ย (mean ERLE) จะอยู่ที่ 64 dB ขณะที่อัลกอริทึมเดิมมีค่าอยู่ที่ประมาณ 56 dB ให้ผลดีกว่าอัลกอริทึมแบบเดิมประมาณ 8 dB

รูปที่ 6.12 (b) แสดงค่าเฉลี่ยสะสมสัญญาณสะท้อนที่เหลือจากการขจัดในระบบ (Average echo residue) ที่ SNR = -10 to 70 dB จะเห็นได้ว่า อัลกอริทึม VSS ที่นำเสนอแบบที่ 2 ยังคงให้ผลที่ดีกว่าอัลกอริทึมแบบเดิมเมื่อ SNR มีค่ามากกว่า -5 dB ขึ้นไป และมีค่าน้อยกว่าอย่างคงที่เมื่อ SNR มีค่ามากกว่า 20 dB ขณะที่อัลกอริทึมเดิมมีค่าเฉลี่ยสะสมสัญญาณสะท้อนที่เหลือจากการขจัดในระบบ อยู่ที่ประมาณ -67 dB อัลกอริทึม VSS ที่นำเสนอแบบที่ 2 มีค่าอยู่ประมาณ -76 dB ซึ่งให้ผลที่ดีกว่าอัลกอริทึมเดิมประมาณ 9 dB

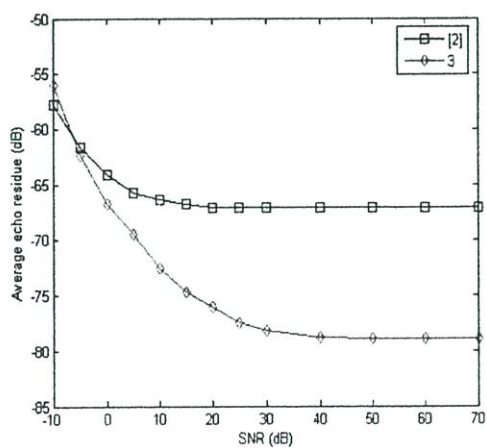
รูปที่ 6.12 (c) แสดงการติดตามสัญญาณของอัลกอริทึม (Tracking) ที่ SNR = 70 dB เห็นได้ว่า อัลกอริทึมอัลกอริทึม VSS ที่นำเสนอแบบที่ 2 มีความสามารถในการติดตามสัญญาณสะท้อนได้ดีขึ้น

รูปที่ 6.12 (d) แสดงค่า ERLE ที่ SNR = 70 dB ในสภาวะที่มีสัญญาณรบกวนรบกวนน้อยมาก ๆ จะเห็นได้ว่าอัลกอริทึมอัลกอริทึม VSS ที่นำเสนอแบบที่ 2 จะเริ่มมีค่าสูงกว่าอัลกอริทึมเดิมเมื่อจำนวนตัวอย่างมากกว่า 300 ตัวอย่างขึ้นไป

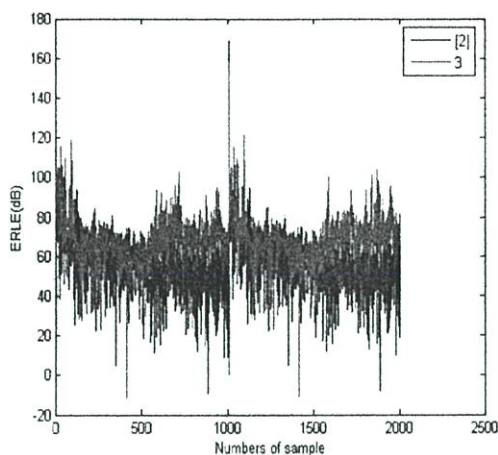
6.3.1.3 การจำลองอัลกอริทึม VSS ที่นำเสนอแบบที่ 3 เปรียบเทียบกับอัลกอริทึมเดิม [2]



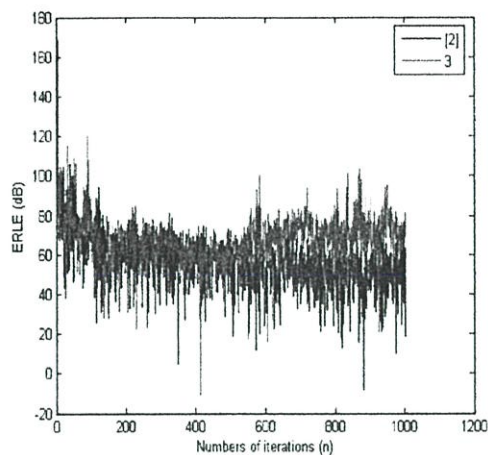
(a) Mean ERLE (SNR = -10 to 70 dB)



(b) Average echo residue (SNR = -10 to 70 dB)



(c) Tracking (SNR= 70 dB)



(d) ERLE (SNR= 70 dB)

รูปที่ 6.13 ผลการจำลองอัลกอริทึม VSS ที่นำเสนอแบบที่ 3 เปรียบเทียบกับอัลกอริทึมเดิม [2]

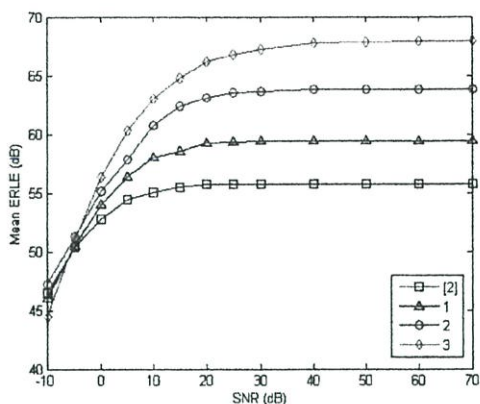
รูปที่ 6.13 (a) แสดงการเปรียบเทียบวัดหาค่าเฉลี่ย ERLE (mean ERLE) ที่ SNR = -10 to 70 dB จะเห็นได้ว่า ในช่วงแรก ๆ ที่ SNR = -10 dB อัลกอริทึม VSS ที่นำเสนอแบบที่ 3 ให้ผลที่ดีกว่า อัลกอริทึมแบบเดิม แต่เมื่อ SNR มีค่าเพิ่มขึ้นมากกว่า -5 dB ขึ้นไป อัลกอริทึมนี้จะให้ค่าเฉลี่ย ERLE ที่ดีขึ้นมาก จากกราฟจะเห็นว่าอยู่ที่ประมาณ 68 dB ขณะที่อัลกอริทึมเดิมมีค่าอยู่ที่ ประมาณ 56 dB ให้ผลดีกว่าอัลกอริทึมแบบเดิมประมาณ 12 dB

รูปที่ 6.13 (b) แสดงค่าเฉลี่ยสะสมสัญญาณสะท้อนที่เหลือจากการขจัดในระบบ (Average echo residue) ที่ SNR = -10 to 70 dB จะเห็นได้ว่า เมื่อ SNR = -10 dB อัลกอริทึม VSS ที่นำเสนอ แบบที่ 3 จะให้ผลที่ดีกว่าอัลกอริทึมแบบเดิม แต่เมื่อ SNR มีค่ามากกว่า -5 dB ขึ้นไปแล้ว ค่าเฉลี่ยสะสมสัญญาณสะท้อนที่เหลือจากการขจัดในระบบ จะลดลงอย่างมากอยู่ที่ประมาณ -79 dB ขณะที่อัลกอริทึมเดิมมีค่าอยู่ประมาณ -67 dB ซึ่งให้ผลที่ดีกว่าอัลกอริทึมเดิมประมาณ 12 dB

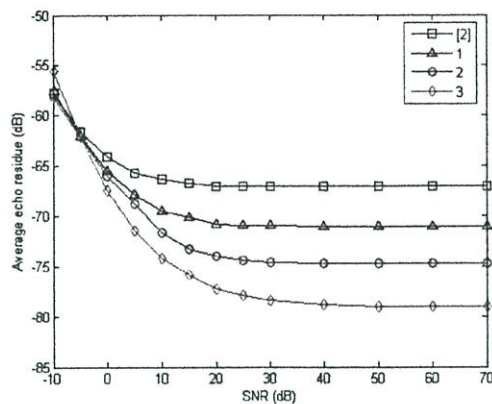
รูปที่ 6.13 (c) แสดงการติดตามสัญญาณของอัลกอริทึม (Tracking) ที่ SNR = 70 dB เห็นได้ว่า อัลกอริทึมอัลกอริทึม VSS ที่นำเสนอแบบที่ 3 มีความสามารถในการติดตามสัญญาณสะท้อนได้ดี ขึ้น

รูปที่ 6.13 (d) แสดงค่า ERLE ที่ SNR = 70 dB ในสถานะที่มีสัญญาณรบกวนน้อยมาก ๆ จะ เห็นได้ว่าอัลกอริทึมอัลกอริทึม VSS ที่นำเสนอแบบที่ 3 จะเริ่มมีค่าสูงกว่าอัลกอริทึมเดิมโดยใช้ จำนวนตัวอย่างประมาณ 100 ตัวอย่างเท่านั้นเอง

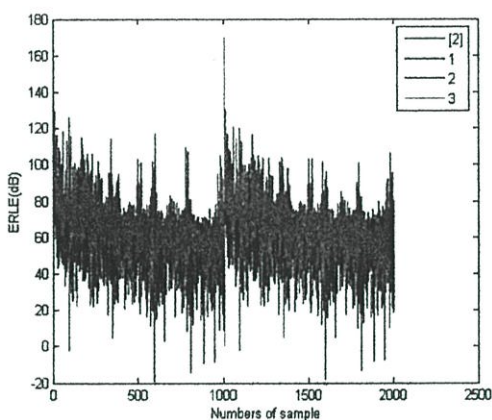
6.3.1.4 การจำลองอัลกอริทึม VSS ที่นำเสนอทั้งสามแบบเปรียบเทียบกับอัลกอริทึมเดิม [2]



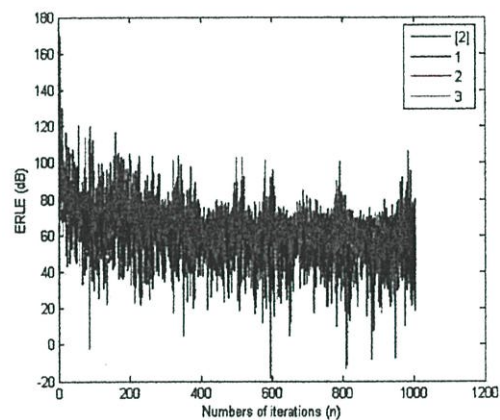
(a) Mean ERLE (SNR= -10 to 70 dB)



(b) Average echo residue (SNR= -10 to 70 dB)



(c) Tracking (SNR= 70 dB)



(d) ERLE (SNR= 70 dB)

รูปที่ 6.14 ผลการจำลองอัลกอริทึม VSS ที่นำเสนอทั้งสามแบบ เปรียบเทียบกับอัลกอริทึมเดิม [2]

จากรูปที่ 6.14 แสดงการเปรียบเทียบการขจัดสัญญาณสะท้อน จะเห็นได้ชัดเจนว่าอัลกอริทึมที่นำเสนอทั้ง 3 อัลกอริทึม มีสมรรถนะที่ดีกว่าอัลกอริทึมเดิม [2]

- รูปที่ 6.14 (a) แสดงสมรรถนะของวงจรถัดสัญญาณสะท้อนแบบอะแดปทีฟ ด้วย ค่า mean ERLE โดยการเลือกค่า SNR ให้มีค่าอยู่ในช่วง -10 dB ถึง 70 dB จะเห็นได้ว่า
 - อัลกอริทึม VSS แบบที่ 1 มีค่า mean ERLE สูงสุดที่ประมาณ 59 dB
 - อัลกอริทึม VSS แบบที่ 2 มีค่า mean ERLE สูงสุดที่ประมาณ 64 dB
 - อัลกอริทึม VSS แบบที่ 3 มีค่า mean ERLE สูงสุดที่ประมาณ 68 dB

ส่วนในสภาวะที่มีสัญญาณรบกวนมาก คือช่วง -10 dB ถึง -6 dB อัลกอริทึม VSS แบบที่ 3 จะมีค่า mean ERLE ต่ำกว่าอัลกอริทึมอื่นเล็กน้อย แต่เมื่อผ่านช่วง SNR ที่มีค่าสูง ๆ แล้ว อัลกอริทึมที่นำเสนอในวิทยานิพนธ์ทั้งสามแบบนี้ จะทำงานได้ดีขึ้นตามลำดับ

- รูปที่ 6.14 (b) แสดงการเปรียบเทียบ กำลังงานเฉลี่ยสะสมของสัญญาณสะท้อนที่เหลืออยู่ (Average echo residue) โดยการเลือกค่า SNR ให้มีค่าอยู่ในช่วง -10 dB ถึง 70 dB พบว่า
 - อัลกอริทึม VSS แบบที่ 1 มีค่า Average echo residue สูงสุดที่ประมาณ -71 dB
 - อัลกอริทึม VSS แบบที่ 2 มีค่า Average echo residue สูงสุดที่ประมาณ -75 dB
 - อัลกอริทึม VSS แบบที่ 3 มีค่า Average echo residue สูงสุดที่ประมาณ -79 dB
 ส่วนในสภาวะที่มีสัญญาณรบกวนมาก คือช่วง -10 dB ถึง -6 dB อัลกอริทึม VSS แบบที่ 3 จะมีค่าสูงกว่าอัลกอริทึมอื่นเล็กน้อย ซึ่งจะมีลักษณะที่สัมพันธ์กับค่า mean ERLE จากรูปที่ 6.13 (a) ที่ได้กล่าวมาแล้ว
- รูปที่ 6.14 (c) แสดงสมรรถนะ ในการติดตามสัญญาณ โดยการเลือกค่า SNR ให้มีค่า 70 dB จะเห็นได้ว่า อัลกอริทึม VSS ที่ได้นำเสนอ ยังคงให้สมรรถนะในการติดตามสัญญาณที่ดีขึ้น
- รูปที่ 6.14 (d) แสดงค่า ERLE ที่ $SNR = 70$ dB สภาวะที่มีสัญญาณรบกวนน้อย พบว่า อัลกอริทึม VSS ทั้งสามแบบ สามารถขจัดสัญญาณสะท้อนกลับได้ดีขึ้นตามลำดับ

6.4 สรุป

จากผลการทดลองพบว่าอัลกอริทึม VSS ทั้งสามแบบที่นำเสนอในวิทยานิพนธ์นี้ โดยใช้ตัวกรองอะแดปทีฟนอซ์ อันดับสอง โครงสร้างแบบแลคทิกซ์ ผลการทดสอบอัลกอริทึมด้วยการจำลองการทำงานบนโปรแกรม MATLAB โดยเริ่มจากการจำลองการทำงานการประมาณค่าสัญญาณขาหน้าที่ถูกรบกวนด้วยสัญญาณรบกวนแบบเกาส์เซียน พบว่าสามารถทำงานได้ดีมีประสิทธิภาพสูง กล่าวคือ สามารถปรับสัมประสิทธิ์เข้าสู่ค่าที่ต้องการได้เร็ว มีความผิดพลาดที่ค่าความแปรปรวนต่ำ และยังมีค่า bias ที่ต่ำอีกด้วย ต่อมาเป็นการประยุกต์ใช้งานโดยนำอัลกอริทึมที่นำเสนอไปจำลองการขจัดสัญญาณสะท้อนในระบบ โทรศัพท์ที่มีการรบกวนแบบเกาส์เซียน แสดงโดยค่า ERLE ทั้งค่าเฉลี่ย ERLE (Mean ERLE) ที่มีค่าสูงกว่า 55 dB และ ค่าเฉลี่ยสะสม ERLE (Average ERLE) มีค่าต่ำกว่า -70 dB แสดงว่าความสามารถในการประมาณค่าสัญญาณสะท้อนกลับของตัวกรองอะแดปทีฟและอัลกอริทึมมีความถูกต้อง แม่นยำมากยิ่งขึ้น รวมทั้งยังมีประสิทธิภาพในการติดตามสัญญาณสะท้อนกลับที่ได้รวดเร็วขึ้น สามารถขจัดสัญญาณสะท้อนกลับในระบบโทรศัพท์ได้ดีขึ้นตามลำดับ

บทที่ 7

สรุปผลวิจัยและข้อเสนอแนะ

วิทยานิพนธ์นี้ ได้นำเสนออะแดปทีฟอัลกอริทึม VSS ใหม่ทั้งสามรูปแบบ เน้นการพัฒนา ด้านความเร็ว ความทนทานต่อสัญญาณรบกวน และความถูกต้องของการประมาณให้มากที่สุด โดยใช้ตัวกรองโครงสร้างแลตทิซแบบนอกรีตอันดับสอง ทำให้การหาคำตอบของตัวกรองแบบอะแดปทีฟสามารถทำได้รวดเร็วยิ่งขึ้น โครงสร้างแลตทิซมีความเป็นโมดูล สามารถเพิ่มหรือลด อันดับของโครงสร้างได้ง่ายและสะดวกต่อการพิจารณาเสถียรภาพ อะแดปทีฟอัลกอริทึมแบบ VSS ในวิทยานิพนธ์นี้ได้พัฒนามาจากอัลกอริทึมแบบ Modified-MNG [19] โดยทำการแปรค่าเสถียร ไซส์ใหม่ทั้งสามรูปแบบ จำลองการทำงานด้วยโปรแกรม MATLAB ดังนี้

อัลกอริทึม VSS แบบที่ 1 จะเป็นการปรับเปลี่ยนแปรค่าเสถียร ไซส์ เพื่อควบคุมการปรับค่าสัมประสิทธิ์ของตัวกรองให้ทนต่อสัญญาณรบกวนดังที่ใช้ในบทวิจัย [3] [21] คือ การใช้เทคนิคการปรับเปลี่ยนค่าก้าวกระโดด โดยใช้ค่าประมาณเวลาเฉลี่ยของสัญญาณเอาต์พุตควบคุมการปรับเปลี่ยนค่าเสถียร ไซส์ ซึ่งทำให้ผลของการถ่วงน้ำหนักที่แท้จริงของสัมประสิทธิ์มีความถูกต้อง มากยิ่งขึ้น จากการใช้อัลกอริทึมที่อยู่บนพื้นฐานของเกรเดียนต์ ทำให้เกิดสัญญาณรบกวนเกรเดียนต์ ซึ่งมีผลต่อความแปรปรวน (Variance) ของสัมประสิทธิ์ วิธีการนี้สามารถลดความแปรปรวนของสัมประสิทธิ์ลงได้ โดยใช้ค่าประมาณเวลาเฉลี่ยของค่าความคลาดเคลื่อน $e(n)$, $e(n-1)$ และค่า $MSE \hat{e}(n)^2$ เพื่อควบคุมการปรับเปลี่ยนค่าก้าวกระโดดให้มีความละเอียดยิ่งขึ้นในสถานะ steady state ค่าความคลาดเคลื่อนที่ป้อนกลับเข้ามาในระบบจึงมีความถูกต้องมากยิ่งขึ้น

อัลกอริทึม VSS แบบที่ 2 คือ แบบปรับเปลี่ยนค่าเสถียร ไซส์ด้วยการปรับสัญญาณเกรเดียนต์ ใหม่ โดยใช้ผลต่างของสัญญาณระหว่างอินพุตในอดีตและผลคูณของเอาต์พุตในอดีตกับค่า ρ ทำให้คำตอบที่ได้ มีความถูกต้องมากขึ้น

ส่วนอัลกอริทึม VSS แบบที่ 3 คือ การปรับเปลี่ยนขนาดของแถบความถี่หูดให้เปลี่ยนแปลงตามเวลา [20] วิธีนี้จะทำให้ได้ความถูกต้องของคำตอบที่แท้จริงสูงสุด การประมาณค่าสัญญาณทำได้ถูกต้องมากยิ่งขึ้น

ผลที่ได้จากการประมาณค่าสัญญาณชายน์พบว่า อัลกอริทึมที่พัฒนาขึ้นมาทั้งสามแบบ มีความรวดเร็วในการหาคำตอบเพิ่มมากขึ้นตามลำดับ ผลจากการนำค่าความคลาดเคลื่อนที่ได้จากการประมาณค่าแบบเกรเดียนต์ และ ค่าเฉลี่ยพลังงานเอาต์พุต รวมทั้งอัตราสัมพัทธ์ของสัญญาณความคลาดเคลื่อน นำมาใช้ในการพัฒนาอัลกอริทึม VSS ใหม่ทั้งสามรูปแบบ ทำให้การกรองสัญญาณดีขึ้น มีประสิทธิภาพมากยิ่งขึ้น ให้ค่า variance และ MSE ลดลง การปรับตัวทำได้

รวดเร็ว ถูกต้อง สามารถติดตามสัญญาณที่เปลี่ยนแปลงได้ และยังมีความทนทานต่อการรบกวนของสัญญาณรบกวนแบบเกาส์เซียนได้ดี ตามลำดับ

ส่วนผลที่ได้จากการจำลองขจัดสัญญาณสะท้อนกลับในระบบโทรศัพท์ สมรรถนะของวงจรขจัดสัญญาณสะท้อนกลับแบบอะแคปทีฟ สามารถแสดงด้วย ค่า mean ERLE โดยการเลือกค่า SNR ให้มีค่าอยู่ในช่วง -10 dB ถึง 70 dB จะเห็นได้ว่า ในสถานะที่มีสัญญาณรบกวนมาก คือ ช่วง -10 dB ถึง -5 dB อัลกอริทึม VSS แบบที่ 3 มีค่า mean ERLE ต่ำกว่าอัลกอริทึมอื่นเล็กน้อย แต่เมื่อ SNR มีค่าสูงขึ้น การขจัดสัญญาณสะท้อนจะทำได้ดีขึ้น ตามลำดับ

อะแคปทีฟอัลกอริทึม VSS ใหม่ทั้งสามรูปแบบที่นำเสนอในวิทยานิพนธ์นี้ ยังสามารถนำไปประยุกต์ใช้งานในการขจัดสัญญาณรบกวนอื่นๆ ได้

บรรณานุกรม

- [1] N. I. Cho, "On the adaptive lattice notch filter for the detection of sinusoidal," IEEE Trans. Circuits Syst., vol. 40, no. 7, pp. 405-416, July 1993.
- [2] C. Benjangkprasert, S. Sukhumalwong, S. Teerasakworakun, and K. Janchitrapongvej, "The new variable step-size algorithm adaptive lattice structure for echo cancellation," Proceeding of ICCAS2003, pp. 2090-2092, Oct. 2003.
- [3] C. Benjangkprasert, S. Teerasakworakun, S. Jorpochaudom, and S. Puntheeranurak, "An imple-mentation of superior noise-robust step-size lattice structure for echo cancellation," Proceeding of NOLTA2002, pp. 667-670, Oct. 2002.
- [4] J. H. Yoo, S. H. Cho, and D. H. Youn, "A Lattice/Transversal Joint (LTJ) structure for an acoustic echo canceller," 1995 Proceeding of IEEE International Symposium on Circuits and Systems, Vol. 2, pp. 1090-1093, 30 Apr-3 May 1995.
- [5] J. F. Chicharo and T. S. Ng, "Gradient-based adaptive IIR notch filtering for frequency estimation," IEEE Trans. Acoust., Speech, Signal Processing, vol. ASSP-38, pp. 769-777, May 1990.
- [6] ITU-T Recommendation G.168 : **Digital Network Echo canceller**, 2003.
- [7] K. C. Ho, Q. G. Liu, R. Rabipour, and P. Yatrou. "A new sampling of echo paths in north America networks," Proceeding of ICASSP '00, vol. 2, pp. 809-812, June 2000.
- [8] B. Farhang - Boroujeny, **Adaptive filters: Theory and applications**, Chic hester : John Wiley, 1998.
- [9] S. C. Pei and C. C. Tseng , "Adaptive IIR notch filter based on least mean p-power error criterion," IEEE Trans. Circuit Syst. -II, vol. 40, pp. 525-529, Aug. 1993.
- [10] L. J. Griffiths, "An adaptive lattice structure for noise cancelling application," Proc. of ICASSP, pp. 69-70, Apr. 1978.
- [11] T. Aboulnasr and K. Mayyas, "A robust variable step-size LMS- type algorithm analysis and simulation," IEEE Trans. Signal Processing, vol. 45, no. 3, pp. 631-639, March 1997.
- [12] S.-I. Amari, "Learning patterns and pattern sequences by self-organizing nets of threshold element", IEEE Trans.Comput., vol. C-21, pp. 1197-206, Nov. 1972.

- [13] J.J. Hopfield and D.W. Tank, "Neural computation of decision in optimization problem", *Biol. Cybern.*, vol. 52, pp. 141-152, July 1985.
- [14] S.V. Aiyer, N. Niranjana and F. Fallside, "A theoretical investigation into the performance of the Hopfield model", *IEEE Trans. Neural*, vol. 1, pp. 204-15, June 1990.
- [15] P. Tupchai, C. Benjangkprasert, O. Sangaroon, and K. Janchitrapongvej, "A new algorithm of adaptive IIR notch filter for the detection of sinusoids," *Proc. in IEEE Asia-Pacific Conference on Circuits and Systems (APCCAS 2002)*, pp. 505–508, Oct. 2002.
- [16] B. Widrow et al. "Adaptive noise canceling: Principles and applications," *Proc. IEEE*, vol. 63, pp. 692-1716, Dec. 1975.
- [17] S. Haykin, **Adaptive Filter Theory**, 2nd ed. New Jersey : Prentice-Hall, 2002.
- [18] N. Kalouptsidis and S. Theodoridis, **Adaptive System Identification and Signal Processing Algorithm**, Great Britain, Cambridge : Prentice-Hall, 1993.
- [19] P. Wattanaluk, C. Benjangkprasert, O. Sangaroon, and K. Janchitrapongvej, "New variable step-Size algorithm for lattice form structure adaptive IIR filter," *Proc. of Electronics, computer, telecommunications, and information technology association conference (ECTI-CON 2006)*, pp. 14-17, May. 2006.
- [20] S. Sukhumalwong, C. Benjangkprasert, N. Anantrasirichai, and K. Janchitrapongvej, "Echo cancellation using the new variable step-size algorithm adaptive lattice filters," *Proceeding of SICE-CIASE*, Oct. 18-21, Bexco, Korea, 2006.
- [21] S. Sukhumalwong, C. Benjangkprasert, and K. Janchitrapongvej, "Adaptive lattice structure filters using variable step-size algorithm for echo cancellation," *Proceeding of IEEE TENCON*, Nov. 14-17, Hong kong, 2006.
- [22] C. J. Gibson and S. Haykin, "Learning characteristics of adaptive lattice filtering algorithm," *IEEE Trans. on Acoustics, Speech, and Signal Processing*, Vol. ASSP-28, No. 6, Dec. 1980.
- [23] N. I. Cho and S. U. Lee, "Adaptive line enhancement by using an IIR lattice notch filter," *IEEE Trans. on Acoustics, Speech, and Signal Processing*, Vol. 37, No. 4, pp.585-589, April 1989.
- [24] A. V. Oppenheim, R. W. Schaffer, and J. R. Buck, **Discrete-Time Signal Processing**, New Jersey : Prentice-Hall, 1998.

- [25] Y. Xiao, Y. Tadakoro, and Y. Kobayashi, "A new memoryless nonlinear gradient algorithm for a second-order adaptive IIR notch filter and its performance analysis," *IEEE Trans. on Circuit and Systems*, Vol. 45, No. 4, April 1998.
- [26] B. Widrow et al. "Adaptive noise canceling: Principles and applications," *Proc. IEEE*, vol. 63, pp. 1692-1716, Dec. 1975.
- [27] C. Benjangkaprasert, S. Phuvasitkul, W. Limwong, and K. Janchitrapongvej, "Fast convergence algorithm for adaptive IIR notch filter using combination of genetic search and variable step-size algorithm," *Proc. in IEEE Intelligent Transportation Systems*, pp. 948-952, Oct. 2003.
- [28] S. K. Mitra, **Digital signal processing a computer-based approach**, McGraw-hill international edition. , 1998.

ภาคผนวก

ภาคผนวก ก

ผลงานวิจัยที่ได้รับการตีพิมพ์

- [1] S. Sukhumalwong, C. Benjangkaprasert, N. Anantrasirichai, K. Janchitrapongvej, "Echo cancellation using the new variable step-size algorithm adaptive lattice filters," Proceeding of SICE-CIASE, Oct. 18-21, Bexco, Korea, 2006.
- [2] S. Sukhumalwong, C. Benjangkaprasert, K. Janchitrapongvej, "Adaptive lattice structure filters using variable step-size algorithm for echo cancellation," Proceeding of IEEE TENCON, Nov.14-17, Hong kong, 2006.

ประวัติผู้เขียน

นายเสถียรวุฒิ สุขุมลวงค์ เกิดเมื่อวันที่ 4 กุมภาพันธ์ 2510 ที่จังหวัดอุบลราชธานี สำเร็จการศึกษาระดับปริญญาตรี หลักสูตรอุตสาหกรรมศาสตรบัณฑิต สาขาวิชาเทคโนโลยีอิเล็กทรอนิกส์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ปีการศึกษา 2538 และเข้าศึกษาต่อในหลักสูตรวิศวกรรมศาสตรมหาบัณฑิต สาขาวิชาวิศวกรรมสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ปีการศึกษา 2545 ปัจจุบันทำงานอยู่ที่ บริษัท ทีโอที จำกัด (มหาชน) ตำแหน่ง วิศวกร 7