

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

**การเข้ารหัสสัญญาณเสียงพูดด้วยวิธีอะแด็ปทีฟที่พัลส์เฟอ์เรนเชียล
พัลส์โค้ดมอดูเลชันโดยใช้เทคนิคออโต้คอร์เรเลชันคู่**

**SPEECH CODING BY ADAPTIVE DIFFERENTIAL PULSE CODE
MODULATION USING DUAL AUTOCORRELATION TECHNIQUE**



วรกาน วังศ์สายเชื้อ
WORAKARN WONGSAICHUA

เลขหมู่.....
เลขหมู่เขียน.....
วัน,เดือน,ปี.....

2544

80850

23 พ.ค. 2551

11919891
b.....
i.....

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมไฟฟ้า
บัณฑิตวิทยาลัย
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ.2544

ISBN 974-648-118-5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



COPYRIGHT 2001

SCHOOL OF GRADUATE STUDIES

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**SPEECH CODING BY ADAPTIVE DIFFERENTIAL PULSE CODE
MODULATION USING DUAL AUTOCORRELATION TECHNIQUE**



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

2001

ISBN 974-648-118-5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อวิทยานิพนธ์	การเข้ารหัสสัญญาณเสียงพูดด้วยวิธีอะแด็ปทีฟคิฟเฟอร์เรนเซียลพัลส์โค้ดมอดูเลชัน โดยใช้เทคนิคออดิโอคอร์เรชัน
นักศึกษา	นายวรการ วงศ์สายเชื้อ
รหัสประจำตัว	41061010
ปริญญา	วิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชา	วิศวกรรมไฟฟ้า
พ.ศ.	2544
อาจารย์ผู้ควบคุมวิทยานิพนธ์	ผศ.ดร.ไกรสิน ส่งวัฒนา

บทคัดย่อ

วิทยานิพนธ์นี้นำเสนอการเข้ารหัสสัญญาณเสียงพูดแบบอะแด็ปทีฟคิฟเฟอร์เรนเซียลพัลส์โค้ดมอดูเลชัน (ADPCM) โดยใช้ออดิโอคอร์เรชันในการประมาณค่าสัญญาณ โดยทั่วไปค่าออดิโอคอร์เรชันของสัญญาณเสียงพูดที่อยู่ติดกันจะมีค่ามากกว่าสัญญาณที่ย้อนหลังไปหลายค่า ดังนั้นการประมาณค่าของสัญญาณปัจจุบันโดยใช้สัญญาณถัดไปหนึ่งค่าและย้อนหลังหนึ่งค่า จึงให้ค่าประมาณที่ใกล้เคียงกว่าการใช้เฉพาะค่าสัญญาณย้อนหลังอย่างเดียว ซึ่งได้ทำการทดลองโดยนำหลักการประมาณสัญญาณดังกล่าวมาใช้ในการเข้ารหัส ADPCM แล้วเปรียบเทียบประสิทธิภาพกับการเข้ารหัส ADPCM ตามมาตรฐาน

Thesis Title	Speech Coding by Adaptive Differential Pulse Code Modulation using Dual Autocorrelation Technique
Student	Mr. Worakarn Wongsachua
Student ID.	41061010
Degree	Master of Engineering
Programme	Electrical Engineering
Year	2001
Thesis Advisor	Assist.Prof.Dr.Kraisin Songwatana

ABSTRACT

This thesis proposes speech coding by Adaptive Differential Pulse Code Modulation (ADPCM) by using the dual autocorrelation to predict the signal. Generally, the autocorrelation of adjacent speech sample is greater than the autocorrelation of several order time delayed samples. So, the prediction of present sample by using the one past and next sample is more effective than only using the several order time delayed past samples. This prediction method is used in ADPCM encoding and the result compares with the standard ADPCM encoding.

กิตติกรรมประกาศ

ขอขอบพระคุณ ผศ.ดร. ไกรสิน ส่วงวัฒนา อาจารย์ที่ปรึกษาที่ได้กรุณาให้คำปรึกษา ข้อเสนอแนะ และให้ใช้ห้องทดลองในการทำวิทยานิพนธ์

ขอขอบคุณบัณฑิตวิทยาลัย ที่ให้ทุนสนับสนุนการทำวิทยานิพนธ์

ขอขอบคุณพี่ๆ เพื่อนๆ และน้องๆ ทุกคนที่ให้ความช่วยเหลือ

ขอขอบคุณเจ้าของเสียงที่ใช้ในการทดลองทุกคน

และขอกราบขอบพระคุณคุณพ่อ คุณแม่ ที่ให้โอกาส สนับสนุน และให้กำลังใจ

วรการ วงศ์สายเชื้อ



สารบัญ

	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญตาราง	VI
สารบัญรูป	VII
บทที่ 1 บทนำ	1
1.1 ความเป็นมาและความสำคัญของปัญหา	1
1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา	1
1.3 สมมติฐานของการศึกษา	1
1.4 ทฤษฎีหรือแนวความคิดที่ใช้ในการวิจัย	2
1.5 ขอบเขตการวิจัย	2
1.6 ขั้นตอนการศึกษา	2
บทที่ 2 ทฤษฎีที่เกี่ยวข้อง	3
2.1 กล่าวนำ	3
2.2 การเข้ารหัส DPCM	3
2.3 ควอนไทเซอร์	4
2.3.1 ยูนิฟอร์มควอนไทเซอร์	4
2.3.2 อะแด็ปทีฟควอนไทเซอร์แบบหน่วยความจำหนึ่งค่า	5
2.4 ตัวประมาณ	5
บทที่ 3 ตัวประมาณเชิงเส้น	6
3.1 กล่าวนำ	6
3.2 ตัวประมาณเชิงเส้นแบบเหมาะสม	6
3.3 ตัวประมาณเชิงเส้นแบบอะแด็ปทีฟ	9
3.4 ขอบเขตในการปรับค่าสัมประสิทธิ์ของตัวประมาณเชิงเส้น	10
3.5 ตัวประมาณเชิงเส้นตามมาตรฐาน ITU G.726	12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า
บทที่ 4 ตัวประมาณออคอร์รีเลขชั้นคู่	13
4.1 กล่าวนำ	13
4.2 ตัวประมาณออคอร์รีเลขชั้นคู่แบบธรรมดา	13
4.3 ตัวประมาณออคอร์รีเลขชั้นคู่แบบเหมาะสมในทางทฤษฎี	15
4.4 ตัวประมาณออคอร์รีเลขชั้นคู่แบบอะแด็ปทีฟ	17
4.5 ขอบเขตในการปรับค่าสัมประสิทธิ์ของตัวประมาณออคอร์รีเลขชั้นคู่	18
4.6 ตัวประมาณออคอร์รีเลขชั้นคู่แบบเหมาะสมจากการทดลอง	20
บทที่ 5 ผลการทดลอง	22
5.1 กล่าวนำ	22
5.2 ผลการทดลองเมื่อไม่มีความผิดพลาดจากการส่ง	22
5.2.1 SNR	22
5.2.2 ค่าความผิดพลาดกำลังสองและสัมประสิทธิ์ของตัวประมาณ	24
5.2.3 รูปคลื่นของสัญญาณอินพุท สัญญาณผลต่าง และสัญญาณสร้างกลับมา	28
5.2.4 ผลตอบสนองทางความถี่	32
5.2.5 เวลาที่ใช้ในการเข้ารหัส	34
5.3 ผลการทดลองเมื่อมีความผิดพลาดจากการส่ง	35
บทที่ 6 สรุปผลการวิจัยและข้อเสนอแนะ	42
เอกสารอ้างอิง	43
ภาคผนวก	
ภาคผนวก ก โปรแกรมที่พัฒนาขึ้นในการวิจัย	44
ภาคผนวก ข ผลงานวิจัยที่ได้รับการตีพิมพ์	60
ประวัติผู้เขียน	66

สารบัญตาราง

ตารางที่	หน้า
2.1 ตัวคูณของอะแด็ปทีฟควอนไทเซอร์แบบหน่วยความจำหนึ่งค่า	5
3.1 ค่าออดิโตรีเลชันของสัญญาณเสียงพูดแซมปลิง 8 kHz	8
4.1 ค่าออดิโตรีเลชัน(ACF) เทียบกับค่าออดิโตรีเลชันคู่(Dual ACF)	13
5.1 SNR ของการเข้ารหัส ADPCM 4 บิตต่อแซมเปิ้ลเมื่อไม่มีความผิดพลาดจากการส่ง	23
5.2 SNR ของการเข้ารหัส ADPCM 3 บิตต่อแซมเปิ้ลเมื่อไม่มีความผิดพลาดจากการส่ง	23
5.3 SNR ของการเข้ารหัส ADPCM 2 บิตต่อแซมเปิ้ลเมื่อไม่มีความผิดพลาดจากการส่ง	23
5.4 เวลาเฉลี่ยที่ใช้ในการเข้ารหัส ADPCM	34
5.5 SNR ของการเข้ารหัส ADPCM 4 บิตต่อแซมเปิ้ลเมื่อมีความผิดพลาดจากการส่ง	36
5.6 SNR ของการเข้ารหัส ADPCM 3 บิตต่อแซมเปิ้ลเมื่อมีความผิดพลาดจากการส่ง	37
5.7 SNR ของการเข้ารหัส ADPCM 2 บิตต่อแซมเปิ้ลเมื่อมีความผิดพลาดจากการส่ง	38
5.8 ลำดับของตัวประมาณตามค่าที่ใช้วัดเมื่อเข้ารหัส 4 บิตต่อแซมเปิ้ล	41
5.9 ลำดับของตัวประมาณตามค่าที่ใช้วัดเมื่อเข้ารหัส 3 บิตต่อแซมเปิ้ล	41
6.10 ลำดับของตัวประมาณตามค่าที่ใช้วัดเมื่อเข้ารหัส 2 บิตต่อแซมเปิ้ล	41

สารบัญรูป

รูปที่	หน้า
2.1 ตัวเข้ารหัส DPCM	3
2.2 ตัวถอดรหัส DPCM	4
2.3 ยูนิฟอร์มควอนไทเซอร์ 3 บิต	4
3.1 ตัวประมาณเชิงเส้นอันดับ N	6
3.2 ขอบเขตในการปรับค่าสัมประสิทธิ์ของตัวประมาณเชิงเส้นอันดับสอง	11
4.1 ตัวเข้ารหัส DPCM ที่ใช้ตัวประมาณออโต้คอร์เรลชันคู่แบบธรรมดา	14
4.2 ตัวถอดรหัส DPCM ที่ใช้ตัวประมาณออโต้คอร์เรลชันคู่แบบธรรมดา	14
4.3 ตัวเข้ารหัส DPCM ที่ใช้ตัวประมาณออโต้คอร์เรลชันคู่แบบเหมาะสม	15
4.4 ตัวถอดรหัส DPCM ที่ใช้ตัวประมาณออโต้คอร์เรลชันคู่แบบเหมาะสม	15
4.5 ขอบเขตในการปรับค่าสัมประสิทธิ์ของตัวประมาณออโต้คอร์เรลชันคู่	19
4.6 SNR เฉลี่ยเมื่อเข้ารหัส ADPCM 4 บิตต่อแซมเปิ้ลโดยใช้ตัวประมาณออโต้คอร์เรลชันคู่ ซึ่งมีค่า h_1 และ h_2 ต่างๆกัน	20
4.7 SNR เฉลี่ยเมื่อเข้ารหัส ADPCM 3 บิตต่อแซมเปิ้ลโดยใช้ตัวประมาณออโต้คอร์เรลชันคู่ ซึ่งมีค่า h_1 และ h_2 ต่างๆกัน	21
4.8 SNR เฉลี่ยเมื่อเข้ารหัส ADPCM 2 บิตต่อแซมเปิ้ลโดยใช้ตัวประมาณออโต้คอร์เรลชันคู่ ซึ่งมีค่า h_1 และ h_2 ต่างๆกัน	21
5.1 SNR เฉลี่ยของการเข้ารหัส ADPCM 4 บิต 3 บิต และ 2 บิตต่อแซมเปิ้ลเมื่อไม่มี ความผิดพลาดจากการส่ง	24
5.2 ค่าความผิดพลาดกำลังสองและสัมประสิทธิ์ของตัวประมาณ Fixed LP2 เมื่อเข้ารหัส ADPCM 4 บิตของไฟล์เสียงผู้ชาย 1 ระหว่างแซมเปิ้ล 3001 ถึง 4000	25
5.3 ค่าความผิดพลาดกำลังสองและสัมประสิทธิ์ของตัวประมาณ Adaptive LP2 เมื่อเข้ารหัส ADPCM 4 บิตของไฟล์เสียงผู้ชาย 1 ระหว่างแซมเปิ้ล 3001 ถึง 4000	25
5.4 ค่าความผิดพลาดกำลังสองและสัมประสิทธิ์ของตัวประมาณ Fixed Dual 1 เมื่อเข้ารหัส ADPCM 4 บิตของไฟล์เสียงผู้ชาย 1 ระหว่างแซมเปิ้ล 3001 ถึง 4000	26
5.5 ค่าความผิดพลาดกำลังสองและสัมประสิทธิ์ของตัวประมาณ Fixed Dual 2 เมื่อเข้ารหัส ADPCM 4 บิตของไฟล์เสียงผู้ชาย 1 ระหว่างแซมเปิ้ล 3001 ถึง 4000	26
5.6 ค่าความผิดพลาดกำลังสองและสัมประสิทธิ์ของตัวประมาณ Fixed Dual 3 เมื่อเข้ารหัส ADPCM 4 บิตของไฟล์เสียงผู้ชาย 1 ระหว่างแซมเปิ้ล 3001 ถึง 4000	27

สารบัญรูป (ต่อ)

รูปที่	หน้า
5.7 ค่าความผิดพลาดกำลังสองและสัมประสิทธิ์ของตัวประมาณ Adaptive Dual เมื่อเข้ารหัส ADPCM 4 บิตของไฟล์เสียงผู้ชาย 1 ระหว่างแซมเปิ้ล 3001 ถึง 4000	27
5.8 ค่าความผิดพลาดกำลังสองและสัมประสิทธิ์ของตัวประมาณ G.726 เมื่อเข้ารหัส ADPCM 4 บิตของไฟล์เสียงผู้ชาย 1 ระหว่างแซมเปิ้ล 3001 ถึง 4000	28
5.9 สัญญาณอินพุต สัญญาณผลต่างและสัญญาณสร้างกลับมาของตัวประมาณ Fixed LP2 เมื่อเข้ารหัส ADPCM 4 บิตของไฟล์เสียงผู้ชาย 1 ระหว่างแซมเปิ้ล 3001 ถึง 4000	29
5.10 สัญญาณอินพุต สัญญาณผลต่างและสัญญาณสร้างกลับมาของตัวประมาณ Adaptive LP2 เมื่อเข้ารหัส ADPCM 4 บิตของไฟล์เสียงผู้ชาย 1 ระหว่างแซมเปิ้ล 3001 ถึง 4000	29
5.11 สัญญาณอินพุต สัญญาณผลต่างและสัญญาณสร้างกลับมาของตัวประมาณ Fixed Dual 1 เมื่อเข้ารหัส ADPCM 4 บิตของไฟล์เสียงผู้ชาย 1 ระหว่างแซมเปิ้ล 3001 ถึง 4000	30
5.12 สัญญาณอินพุต สัญญาณผลต่างและสัญญาณสร้างกลับมาของตัวประมาณ Fixed Dual 2 เมื่อเข้ารหัส ADPCM 4 บิตของไฟล์เสียงผู้ชาย 1 ระหว่างแซมเปิ้ล 3001 ถึง 4000	30
5.13 สัญญาณอินพุต สัญญาณผลต่างและสัญญาณสร้างกลับมาของตัวประมาณ Fixed Dual 3 เมื่อเข้ารหัส ADPCM 4 บิตของไฟล์เสียงผู้ชาย 1 ระหว่างแซมเปิ้ล 3001 ถึง 4000	31
5.14 สัญญาณอินพุต สัญญาณผลต่างและสัญญาณสร้างกลับมาของตัวประมาณ Adaptive Dual เมื่อเข้ารหัส ADPCM 4 บิตของไฟล์เสียงผู้ชาย 1 ระหว่างแซมเปิ้ล 3001 ถึง 4000	31
5.15 สัญญาณอินพุต สัญญาณผลต่างและสัญญาณสร้างกลับมาของตัวประมาณ G.726 เมื่อเข้ารหัส ADPCM 4 บิตของไฟล์เสียงผู้ชาย 1 ระหว่างแซมเปิ้ล 3001 ถึง 4000	32
5.16 ผลตอบสนองทางความถี่ของตัวประมาณแบบต่างๆเมื่อเข้ารหัส ADPCM 4 บิตต่อแซมเปิ้ล	33
5.17 ผลตอบสนองทางความถี่ของตัวประมาณแบบต่างๆเมื่อเข้ารหัส ADPCM 3บิตต่อแซมเปิ้ล	33
5.18 ผลตอบสนองทางความถี่ของตัวประมาณแบบต่างๆเมื่อเข้ารหัส ADPCM 4 บิตต่อแซมเปิ้ล	34
5.19 เวลาเฉลี่ยที่ใช้ในการเข้ารหัส ADPCM	35
5.20 SNR เฉลี่ยของการเข้ารหัส ADPCM 4 บิตต่อแซมเปิ้ลเมื่อมีความผิดพลาดจากการส่ง	39
5.21 SNR เฉลี่ยของการเข้ารหัส ADPCM 3 บิตต่อแซมเปิ้ลเมื่อมีความผิดพลาดจากการส่ง	39
5.22 SNR เฉลี่ยของการเข้ารหัส ADPCM 2 บิตต่อแซมเปิ้ลเมื่อมีความผิดพลาดจากการส่ง	40

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ปัจจุบัน โครงข่ายโทรคมนาคมได้เปลี่ยนระบบการส่งสัญญาณจากแบบอนาล็อกมาเป็นดิจิทัลกันมากขึ้น ซึ่งการที่จะส่งสัญญาณเสียงพูดผ่านโครงข่ายดิจิทัลได้นั้นต้องแปลงสัญญาณเสียงพูดจากอนาล็อกให้เป็นดิจิทัลก่อน วิธีหนึ่งที่นิยมใช้คือการเข้ารหัสแบบพัลส์โค้ดมอดูเลชัน (PCM) โดยทำการแซมปลิง (Sampling) สัญญาณอนาล็อก จากนั้นนำไปควอนไทซ์ (Quantize) แล้วเข้ารหัส 8 บิตต่อหนึ่งแซมเปิล แต่เนื่องจากแซมเปิลของสัญญาณเสียงพูดที่อยู่ติดกันจะมีค่าใกล้เคียงกัน จึงมีการนำคุณสมบัติข้อนี้มาพัฒนาการเข้ารหัสเพื่อลดจำนวนบิตที่ใช้ในการเข้ารหัส โดยใช้ผลต่างระหว่างแซมเปิลข้างเคียงมาทำการควอนไทซ์แล้วเข้ารหัสแทนการใช้ค่าของแซมเปิลนั้น โดยตรงเรียกว่าดิฟเฟอเรนเชียลพัลส์โค้ดมอดูเลชัน (DPCM) ส่วนประกอบสำคัญของ DPCM คือควอนไทเซอร์ (Quantizer) และตัวประมาณ (Predictor) ถ้าควอนไทเซอร์เป็นแบบอะแดปทีฟ (Adaptive) หรือทั้งควอนไทเซอร์และตัวประมาณเป็นแบบอะแดปทีฟ จะเรียกว่าอะแดปทีฟดิฟเฟอเรนเชียลพัลส์โค้ดมอดูเลชัน (ADPCM) ซึ่งเป็นมาตรฐานหนึ่งในการส่งสัญญาณดิจิทัลของ ITU คือ G.726 [1]

การเข้ารหัสตามมาตรฐาน G.726 มีวิธีการที่ค่อนข้างซับซ้อน จึงมีแนวคิดที่จะลดความซับซ้อนลงเพื่อลดเวลาที่ใช้เข้ารหัส โดยยังคงประสิทธิภาพเท่าเดิมหรือดีกว่าเดิม โดยการปรับปรุงควอนไทเซอร์หรือตัวประมาณ

1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา

วิทยานิพนธ์นี้ศึกษาเรื่องตัวประมาณซึ่งใช้เทคนิคออดิโอรีเลชันคู่ [2] เพื่อนำมาใช้ในการเข้ารหัสเสียงพูดแบบ ADPCM ให้มีประสิทธิภาพเท่าเทียมหรือดีกว่าการเข้ารหัส ADPCM ตามมาตรฐาน ITU G.726 โดยใช้เวลาเข้ารหัสสั้นลง

1.3 สมมติฐานของการศึกษา

ตัวประมาณซึ่งใช้เทคนิคออดิโอรีเลชันคู่จะประมาณสัญญาณปัจจุบัน โดยใช้สัญญาณถัดไปหนึ่งค่าและย้อนหลังหนึ่งค่า ซึ่งจะประมาณสัญญาณได้ดีกว่าการใช้สัญญาณย้อนหลังเพียงอย่างเดียวซึ่งใช้ในมาตรฐาน G.726 ดังนั้นเมื่อนำหลักการนี้มาใช้ในการเข้ารหัส ADPCM จึงมีประสิทธิภาพดีกว่า G.726

1.4 ทฤษฎีหรือแนวความคิดที่ใช้ในการวิจัย

วิทยานิพนธ์นี้ใช้หลักการของออดิโอคอร์เลชันฟังก์ชันมาปรับปรุงเป็นเทคนิคออดิโอคอร์เลชันคู่เพื่อทำเป็นตัวประมาณในการเข้ารหัสเสียงพูดแบบ ADPCM โดยทำงานร่วมกับอะแด็ปทีฟควอนไทเซอร์

1.5 ขอบเขตการวิจัย

วิทยานิพนธ์นี้ทำการเข้ารหัสเสียงพูดแบบ ADPCM ซึ่งใช้ตัวประมาณแบบต่างๆ ทำงานร่วมกับอะแด็ปทีฟควอนไทเซอร์ โดยเขียนเป็นซอฟต์แวร์เข้ารหัส 4, 3 และ 2 บิตต่อแซมเปิ้ล แล้ววัดประสิทธิภาพของการเข้ารหัสเปรียบเทียบกันระหว่างตัวประมาณแต่ละแบบ

1.6 ขั้นตอนการศึกษา

วิทยานิพนธ์นี้มีขั้นตอนการเริ่มจากการศึกษาหลักการทำงานของการเข้ารหัส DPCM ซึ่งประกอบด้วยควอนไทเซอร์และตัวประมาณ จากนั้นศึกษารายละเอียดของควอนไทเซอร์และตัวประมาณแต่ละแบบแล้วหาวิธีปรับปรุงประสิทธิภาพให้ดีขึ้น โดยเลือกทำตัวประมาณซึ่งแต่เดิมใช้หลักการของออดิโอคอร์เลชันฟังก์ชัน ปรับปรุงเป็นออดิโอคอร์เลชันคู่แล้วนำมาใช้ร่วมกับอะแด็ปทีฟควอนไทเซอร์ในการเข้ารหัส ADPCM

บทที่ 2

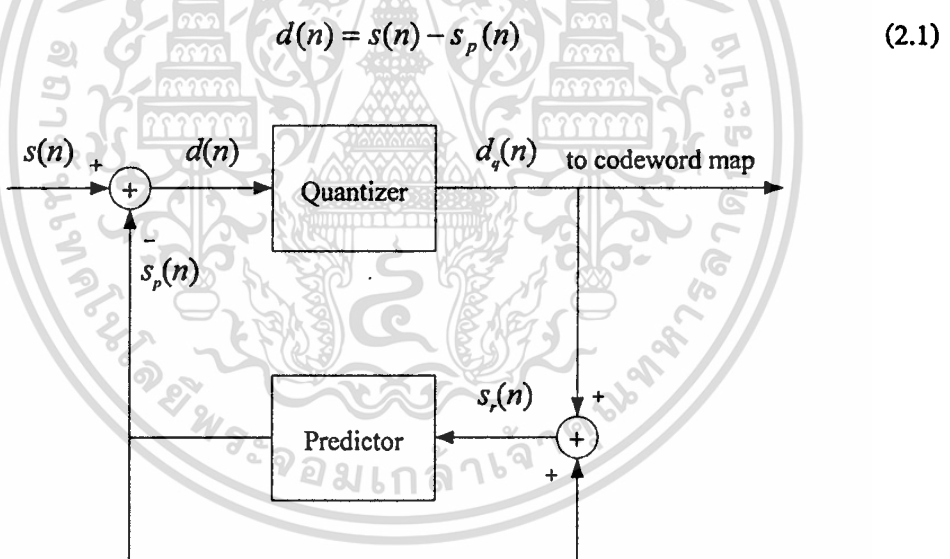
ทฤษฎีที่เกี่ยวข้อง

2.1 กล่าวนำ

การเข้ารหัส ADPCM มีพื้นฐานมาจากการเข้ารหัส DPCM ซึ่งมีส่วนประกอบสำคัญคือ ควอนไทเซอร์และตัวประมาณ ในบทนี้จะกล่าวถึงหลักการของ DPCM ควอนไทเซอร์ และตัวประมาณ

2.2 การเข้ารหัส DPCM

ตัวเข้ารหัส DPCM จะนำสัญญาณผลต่าง $d(n)$ มาควอนไทซ์แล้วเข้ารหัสแทนการใช้สัญญาณอินพุต $s(n)$ โดยตรงดังรูปที่ 2.1 สัญญาณผลต่างได้จากการลบสัญญาณประมาณ $s_p(n)$ ออกจากสัญญาณอินพุต

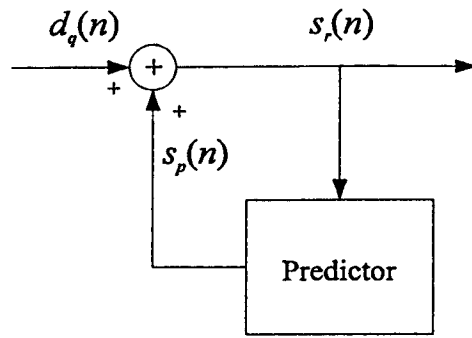


รูปที่ 2.1 ตัวเข้ารหัส DPCM

สัญญาณผลต่างที่ได้จะนำไปควอนไทซ์เป็นสัญญาณผลต่างควอนไทซ์ $d_q(n)$ จากนั้นเข้ารหัสตามจำนวนบิตที่ต้องการแล้วส่งไปยังด้านรับ ในขณะที่เดียวกันสัญญาณผลต่างควอนไทซ์ก็จะป้อนกลับมารวมกับสัญญาณประมาณเป็นสัญญาณสร้างกลับมา $s_r(n)$ ดังสมการ (2.2) ตัวถอดรหัส DPCM ก็คือส่วนป้อนกลับของตัวเข้ารหัสดังแสดงในรูปที่ 2.2

$$s_r(n) = d_q(n) + s_p(n) \quad (2.2)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.2 ตัวถอครหัส DPCM

ADPCM คือ DPCM ที่ควอนไทเซอร์เป็นแบบอะแด็ปทีฟหรือทั้งควอนไทเซอร์และตัวประมาณเป็นแบบอะแด็ปทีฟ

2.3 ควอนไทเซอร์

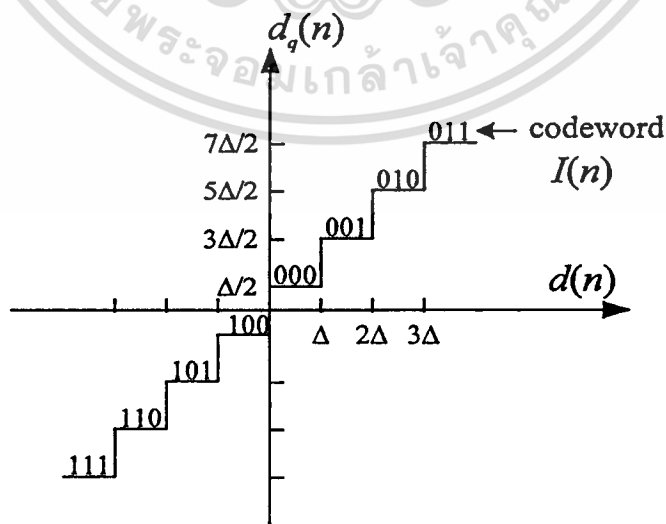
2.3.1 ยูนิฟอร์มควอนไทเซอร์

ในการเข้ารหัส DPCM ควอนไทเซอร์เป็นตัวแปลงสัญญาณผลต่างเป็นสัญญาณผลต่างควอนไทซ์ซึ่งอยู่ในรูป

$$d_q(n) = H(n)\Delta(n)/2 \quad (2.3)$$

$$\pm H(n) = 1, 3, 5, \dots, 2^R - 1 ; R \geq 2$$

โดยที่ Δ คือ step size และ R คือจำนวนบิตที่ใช้เข้ารหัส ถ้า $R = 3$ ค่าของ $|H(n)|$ ที่เป็นไปได้คือ 1, 3, 5 และ 7 ดังรูปที่ 2.3



รูปที่ 2.3 ยูนิฟอร์มควอนไทเซอร์ 3 บิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.2 อะแด็ปทีฟควอนไทเซอร์แบบหน่วยความจำหนึ่งคำ

อะแด็ปทีฟควอนไทเซอร์แบบหน่วยความจำหนึ่งคำ (Adaptive quantizer with a one-word memory) จะปรับ step size ตามสัญญาณผลต่างที่เปลี่ยนไป step size ถัดไปหาได้จากผลคูณของ step size ปัจจุบันกับตัวคูณ M ซึ่งเป็นฟังก์ชันของ $|I(n)|$ [4]

$$\Delta(n+1) = M(|I(n)|)\Delta(n) \quad (2.4)$$

โดยที่ $I(n)$ คือ codeword ซึ่งมี MSB (Most Significant Bit) เป็นบิตเครื่องหมาย ตัวคูณ M จะมีค่าสอดคล้องตามค่าของ $|I(n)|$ เช่น ถ้า $R=3$ ตัวคูณคือ M_1 ถึง M_4 ซึ่งสอดคล้องกับ $|I(n)|=0, 1, 2$ และ 3 ตามลำดับ ค่าของตัวคูณสำหรับ $R=2, 3$ และ 4 แสดงดังตารางที่ 2.1

ตารางที่ 2.1 ตัวคูณของอะแด็ปทีฟควอนไทเซอร์แบบหน่วยความจำหนึ่งคำ [3]

R	M_1	M_2	M_3	M_4	M_5	M_6	M_7	M_8
4	0.90	0.90	0.90	0.90	1.20	1.60	2.00	2.40
3	0.90	0.90	1.25	1.75				
2	0.80	1.60						

จากตารางที่ 2.1 จะเห็นว่าในการปรับ step size นั้นจะปรับให้เพิ่มขึ้นอย่างรวดเร็วแต่จะลดลงอย่างช้าๆ ทั้งนี้เพื่อให้เข้ากับลักษณะรูปคลื่นของสัญญาณเสียงพูด

2.4 ตัวประมาณ

ระบบ DPCM ใช้ตัวประมาณเชิงเส้นในการประมาณค่าของสัญญาณอินพุตคั้งสมการ [3]

$$s_p(n) = \sum_{j=1}^N h_j s(n-j) \quad (2.5)$$

โดยที่ h_j เป็นสัมประสิทธิ์ของตัวประมาณเชิงเส้นซึ่งทำให้ความผิดพลาดกำลังสองเฉลี่ย (Mean Square Error) จากสมการ (2.1) มีค่าน้อยที่สุด ถ้า h_j มีค่าเปลี่ยนตามเวลาจะเรียกว่าตัวประมาณเชิงเส้นแบบอะแด็ปทีฟ ซึ่งเรื่องของตัวประมาณเชิงเส้นจะกล่าวถึงโดยละเอียดในบทที่ 3

บทที่ 3

ตัวประมาณเชิงเส้น

3.1 กล่าวนำ

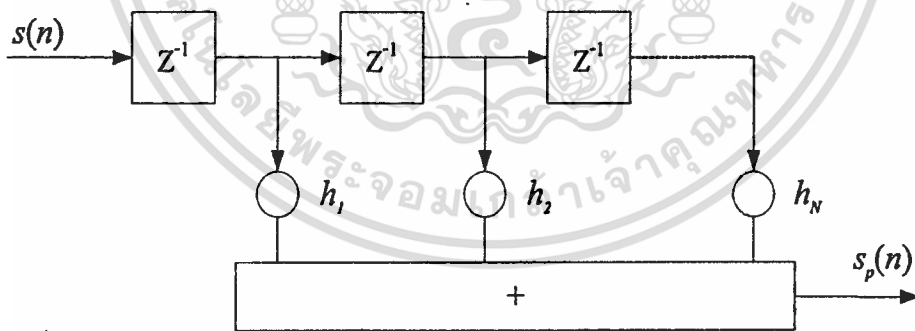
ในบทนี้จะกล่าวถึงตัวประมาณเชิงเส้นแบบต่างๆ ที่ใช้ในการเข้ารหัส DPCM ซึ่งประกอบไปด้วย ตัวประมาณเชิงเส้นแบบเหมาะสม ตัวประมาณเชิงเส้นแบบอะแดปทีฟ และตัวประมาณเชิงเส้นตามมาตรฐาน ITU G.726

3.2 ตัวประมาณเชิงเส้นแบบเหมาะสม

ตัวประมาณเชิงเส้นใช้ประมาณค่าของสัญญาณอินพุตดังสมการ (3.1)

$$s_p(n) = \sum_{j=1}^N h_j s(n-j) \quad (3.1)$$

ซึ่งก็คือผลรวมถ่วงน้ำหนักของแซมเปิลย้อนหลังไป N ค่า โดยที่ h_j เป็นสัมประสิทธิ์ของตัวประมาณเชิงเส้น โครงสร้างของตัวประมาณเชิงเส้นอันดับ N แสดงดังรูปที่ 3.1



รูปที่ 3.1 ตัวประมาณเชิงเส้นอันดับ N

ในการประมาณสัญญาณเสียงพูด ประสิทธิภาพของตัวประมาณเชิงเส้นจะอิมิตัวที่ $N = 2$ [3] จากตัวเข้ารหัส DPCM ในรูปที่ 2.1 ถ้า $N = 2$ จะได้สัญญาณประมาณคือ

$$s_p(n) = h_1 s_r(n-1) + h_2 s_r(n-2) \quad (3.2)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้าไม่คิดผลของนอยส์จากการควอนไทซ์ จะได้สัญญาณประมาณเป็น

$$s_p(n) = h_1 s(n-1) + h_2 s(n-2) \quad (3.3)$$

ถ้าคิดว่าระบบเป็นแบบ stationary จะได้ค่าความผิดพลาดกำลังสองเฉลี่ยคือ

$$\sigma_d^2 = E[d^2(n)] = E\{[s(n) - s_p(n)]^2\} \quad , \quad E[x] = \frac{1}{N} \sum_{i=1}^N x_i \quad (3.4)$$

$$= E\{[s(n) - (h_1 s(n-1) + h_2 s(n-2))]^2\}$$

$$= R_{ss}(0) - 2h_1 R_{ss}(1) - 2h_2 R_{ss}(2) + h_1^2 R_{ss}(0) + 2h_1 h_2 R_{ss}(1) + h_2^2 R_{ss}(0)$$

โดยที่

$$R_{ss}(k) = \frac{1}{N} \sum_{n=0}^{N-1-k} s(n)s(n+k) \quad (3.5)$$

จากนิยามของความผันแปร (Variance)

$$\sigma_s^2 = \frac{1}{N} \sum_{n=0}^{N-1} [s(n) - m_s]^2 \quad (3.6)$$

โดย m_s คือค่าเฉลี่ย (Mean) ถ้าสัญญาณเชิงสุ่มมีค่าเฉลี่ยเป็นศูนย์ จะได้

$$R_{ss}(0) = \sigma_s^2 \quad (3.7)$$

และถ้าค่าออกได้คอริเลชันที่นอร์มัลไลซ์ คือ

$$\rho_k = \frac{R_{ss}(k)}{R_{ss}(0)} \quad (3.8)$$

ดังนั้น

$$\sigma_d^2 = (1 - 2\rho_1 h_1 - 2\rho_2 h_2 + h_1^2 + 2\rho_1 h_1 h_2 + h_2^2) \sigma_s^2 \quad (3.9)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค่า h_1 และ h_2 ที่ทำให้ค่าความผิดพลาดกำลังสองเฉลี่ยมีค่าน้อยที่สุดหาได้จาก

$$\frac{\partial \sigma_d^2}{\partial h_1} = -2\rho_1 + 2h_1 + 2\rho_1 h_2 = 0$$

$$h_1 + \rho_1 h_2 = \rho_1 \quad (3.10)$$

$$\frac{\partial \sigma_d^2}{\partial h_2} = -2\rho_2 + 2h_2 + 2\rho_1 h_1 = 0$$

$$\rho_1 h_1 + h_2 = \rho_2 \quad (3.11)$$

จากสมการ (3.10) และ (3.11) จะได้ [3]

$$h_1 = \frac{\rho_1(1-\rho_2)}{1-\rho_1^2}; h_2 = \frac{\rho_2 - \rho_1^2}{1-\rho_1^2} \quad (3.12)$$

ตารางที่ 3.1 ค่าออดิโอรีเลย์ของสัญญาณเสียงพูดแอมป์ถึง 8 kHz [3]

Speech Source	LPF Source [Noll, 1972]	BPF Source [Noll, 1972]	LPF Source [McDonald, 1966]	BPF Source [Grizmalala, 1972]
ρ_0	1.00	1.00	1.0000	1.000
ρ_1	0.86	0.79	0.8644	0.825
ρ_2	0.64	0.38	0.5570	0.562
ρ_3	0.40	0.05	0.2274	0.308
ρ_4	0.26	-0.08	-0.0297	0.004
ρ_5	0.20	-0.16	-0.1939	-0.243

จากตารางที่ 3.1 ซึ่งแสดงค่าออดิโอรีเลย์ของสัญญาณเสียงพูดแอมป์ถึง 8 kHz เลือก $\rho_1 = 0.86$ $\rho_2 = 0.64$ แทนลงในสมการ (3.12) จะได้

$$h_1 = \frac{\rho_1(1-\rho_2)}{1-\rho_1^2} = 1.189 \quad (3.13)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$h_2 = \frac{\rho_2 - \rho_1^2}{1 - \rho_1^2} = -0.383 \quad (3.14)$$

3.3 ตัวประมาณเชิงเส้นแบบอะแด็ปทีฟ

สัมประสิทธิ์ของตัวประมาณเชิงเส้นในหัวข้อ 3.2 กำหนดมาจากค่าเฉลี่ยทางสถิติในช่วงเวลายาว(Long-timed-averaged statistics) แต่สัญญาณเสียงพูดมีคุณสมบัติเป็น nonstationary ซึ่งมีค่าทางสถิติเปลี่ยนตามเวลา ดังนั้นตัวประมาณที่เหมาะสมควรจะเป็นแบบอะแด็ปทีฟซึ่งจะปรับค่าสัมประสิทธิ์ให้ค่าความผิดพลาดกำลังสองเฉลี่ยมีค่าน้อยที่สุด โดยใช้อัลกอริธึมเกรเดียนต์แบบ Steepest Descent [5] ดังนี้

$$\mathbf{h}(n+1) = \mathbf{h}(n) - \frac{1}{2} \mu(n) \nabla_{\mathbf{h}} [E\{d^2(n)\}] \quad (3.15)$$

โดยที่ $\mathbf{h}(n) = \begin{bmatrix} h_1(n) \\ h_2(n) \end{bmatrix}$ $\mu(n)$ คือ Step size และ $\nabla_{\mathbf{h}} = \begin{bmatrix} \frac{\partial}{\partial h_1} \\ \frac{\partial}{\partial h_2} \end{bmatrix}$

วิธีการนี้จะปรับสัมประสิทธิ์ไปในทิศทางตรงข้ามกับเกรเดียนต์ของ $E[d^2(n)]$ วิธีหาค่าเกรเดียนต์ของ $E[d^2(n)]$ อย่างง่ายวิธีหนึ่งคือวิธีกำลังสองเฉลี่ยน้อยที่สุด(Least Mean Square:LMS) [5] วิธีนี้จะใช้ค่า $d^2(n)$ แทนค่า $E[d^2(n)]$ แต่ที่ด้านรับไม่มี $d(n)$ จึงใช้ $d_q(n)$ แทน

จากสมการ (2.2) $s_r(n) = d_q(n) + s_p(n)$

จะได้ $d_q(n) = s_r(n) - s_p(n)$

แทนค่า $s_p(n)$ จากสมการ (3.2)

$$d_q(n) = s_r(n) - h_1 s_r(n-1) - h_2 s_r(n-2) \quad (3.16)$$

ดังนั้นค่าของเกรเดียนท์คือ

$$\nabla_{\mathbf{h}} [d^2(n)] \approx \nabla_{\mathbf{h}} [d_q^2(n)] = 2d_q(n)\nabla_{\mathbf{h}} d_q(n) = -2d_q(n)\mathbf{s}_r(n) \quad (3.17)$$

$$\text{ซึ่ง } \mathbf{s}_r(n) = \begin{bmatrix} s_r(n-1) \\ s_r(n-2) \end{bmatrix}$$

แทนสมการ (3.17) ลงในสมการ (3.15) จะได้

$$\mathbf{h}(n+1) = \mathbf{h}(n) + \mu(n)d_q(n)\mathbf{s}_r(n) \quad (3.18)$$

โดยที่ $\mu(n) = \frac{g}{100 + \mathbf{s}_r^T(n)\mathbf{s}_r(n)}$ ซึ่ง g คือ ค่าที่ได้จากการทดลอง [4]

3.4 ขอบเขตในการปรับค่าสัมประสิทธิ์ของตัวประมาณเชิงเส้น

ขอบเขตในการปรับค่า h หาได้จากผลตอบสนองอิมพัลส์ (Impulse response) ของระบบ เริ่มจากการแปลงแซด (Z-transform) สมการ (3.16) ได้เป็น

$$D_q(z) = S_r(z) - h_1 S_r(z)z^{-1} - h_2 S_r(z)z^{-2} \quad (3.19)$$

จะได้ฟังก์ชันถ่ายโอน (Transfer function) คือ

$$H(z) = \frac{S_r(z)}{D_q(z)} = \frac{1}{1 - h_1 z^{-1} - h_2 z^{-2}} = \frac{z^2}{z^2 - h_1 z - h_2} = \frac{z^2}{(z-a)(z-b)}$$

$$\text{โดยที่ } a, b = \frac{-(-h_1) \pm \sqrt{(-h_1)^2 - 4(1)(-h_2)}}{2} = \frac{h_1 \pm \sqrt{h_1^2 + 4h_2}}{2}$$

$$H(z) = \frac{1}{(1-az^{-1})(1-bz^{-1})}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$= \left(\frac{a}{a-b} \right) \left(\frac{1}{1-az^{-1}} \right) + \left(\frac{b}{b-a} \right) \left(\frac{1}{1-bz^{-1}} \right) \quad (3.20)$$

จะได้

$$h(n) = \left(\frac{a}{a-b} \right) a^n u(n) + \left(\frac{b}{b-a} \right) b^n u(n) \quad (3.21)$$

ซึ่ง $u(n)$ เป็น unit step sequence โดยที่ $u(n) = \begin{cases} 1, & n \geq 0 \\ 0, & n < 0 \end{cases}$

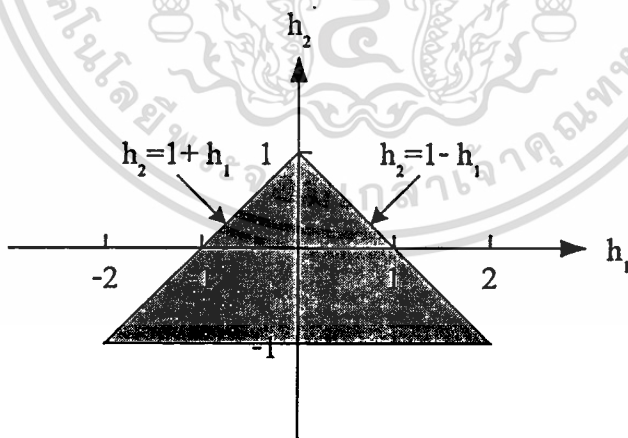
ระบบเชิงเส้นที่ไม่เปลี่ยนแปลงตามเวลา (Linear Time Invariant) จะมีเสถียรภาพเมื่อ [6]

$$\sum_{n=-\infty}^{\infty} |h(n)| < \infty \quad (3.22)$$

นั่นคือ $|a| < 1$ และ $|b| < 1$ จะได้ขอบเขตในการปรับค่าสัมประสิทธิ์คือ [3]

$$|h_2| < 1, |h_1| < 1 - h_2 \quad (3.23)$$

ซึ่งเป็นบริเวณที่แรเงาในรูปที่ 3.2



รูปที่ 3.2 ขอบเขตในการปรับค่าสัมประสิทธิ์ของตัวประมาณเชิงเส้นอันดับสอง

3.5 ตัวประมาณเชิงเส้นตามมาตรฐาน ITU G.726

ตัวประมาณ G.726 [1] ใช้สัญญาณย้อนหลัง 2 ค่าและผลต่างย้อนหลัง 6 ค่าในการประมาณสัญญาณคือ

$$s_p(n) = \sum_{i=1}^2 a_i(n-1)s_r(n-i) + \sum_{i=1}^6 b_i(n-1)d_q(n-i) \quad (3.24)$$

การปรับค่าสัมประสิทธิ์ใช้อัลกอริทึมเกรเดียนต์ดังนี้

$$a_1(n) = \frac{255}{256} a_1(n-1) + \frac{3}{256} \text{sgn}[p(n)] \text{sgn}[p(n-1)] \quad (3.25)$$

$$a_2(n) = \frac{127}{128} a_2(n-1) + \frac{1}{128} \{ \text{sgn}[p(n)] \text{sgn}[p(n-2)] - f[a_1(n-1)] \text{sgn}[p(n)] \text{sgn}[p(n-1)] \} \quad (3.26)$$

$$p(n) = d_q(n) + \sum_{i=1}^6 b_i(n-1)d_q(n-i) \quad (3.27)$$

$$f[a_1] = \begin{cases} 4a_1, & |a_1| \leq 0.5 \\ 2 \text{sgn}[a_1], & |a_1| > 0.5 \end{cases} \quad (3.28)$$

$$b_i(n) = \frac{255}{256} b_i(n-1) + \frac{1}{128} \text{sgn}[d_q(n)] \text{sgn}[d_q(n-i)] \quad (3.29)$$

โดยจะปรับค่าสัมประสิทธิ์ให้อยู่ในขอบเขตดังนี้

$$|a_2| \leq 0.75, |a_1| \leq \frac{15}{16} - a_2 \quad (3.30)$$

$$|b_i(n)| < 2 \quad (3.31)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ตัวประมาณอัตโนมัติคอร์เรเลชันคู่

4.1 กล่าวนำ

ในบทนี้จะกล่าวถึงตัวประมาณอัตโนมัติคอร์เรเลชันคู่แบบต่างๆ ที่ใช้ในการเข้ารหัส DPCM ซึ่งประกอบไปด้วย ตัวประมาณอัตโนมัติคอร์เรเลชันคู่แบบธรรมดา ตัวประมาณอัตโนมัติคอร์เรเลชันคู่แบบเหมาะสมในทางทฤษฎี ตัวประมาณอัตโนมัติคอร์เรเลชันคู่แบบอะแดปทีฟ และตัวประมาณอัตโนมัติคอร์เรเลชันคู่แบบเหมาะสมจากการทดลอง

4.2 ตัวประมาณอัตโนมัติคอร์เรเลชันคู่แบบธรรมดา

โดยทั่วไปค่าอัตโนมัติคอร์เรเลชัน (ACF) ของแอมพลิจูดสัญญาณเสียงพูดที่อยู่ติดกันจะมีค่ามากกว่าแอมพลิจูดที่ห่างออกไป ในตารางที่ 4.1 แสดงค่าอัตโนมัติคอร์เรเลชันที่คำนวณโดยวิธีอัตโนมัติคอร์เรเลชันธรรมดาตามสมการ (4.1) [3] เปรียบเทียบกับค่าอัตโนมัติคอร์เรเลชันคู่ (Dual ACF) ที่คำนวณจากสมการ (4.2) [2] ซึ่งใช้ค่าแอมพลิจูดปัจจุบันและค่าเฉลี่ยของแอมพลิจูดถัดไปและแอมพลิจูดย้อนหลัง จะเห็นว่าแบบ Dual ACF จะมีค่ามากกว่าแบบ ACF ดังนั้นการประมาณค่าแอมพลิจูดปัจจุบันโดยใช้ Dual ACF จึงให้ค่าใกล้เคียงกว่า ACF ตัวเข้ารหัสและถอดรหัส DPCM ที่ใช้ตัวประมาณอัตโนมัติคอร์เรเลชันคู่แสดงดังรูปที่ 4.1 และ 4.2 ตามลำดับ

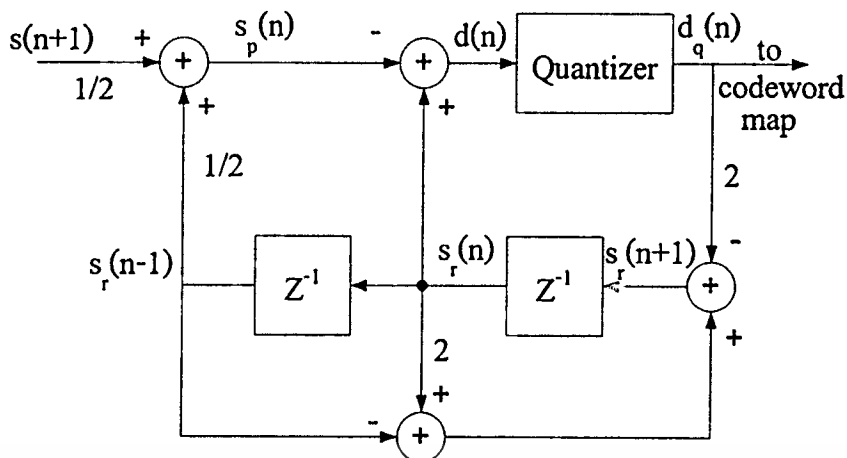
$$ACF(k) = \frac{1}{N} \sum_{n=0}^{N-1-k} s(n)s(n+k) \quad (4.1)$$

$$Dual ACF(k) = \frac{1}{N} \sum_{n=k}^{N-1-k} s(n) \left\{ \frac{s(n-k) + s(n+k)}{2} \right\} \quad (4.2)$$

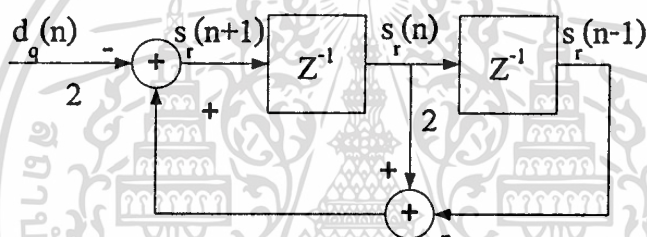
ตารางที่ 4.1 ค่าอัตโนมัติคอร์เรเลชัน (ACF) เปรียบเทียบกับค่าอัตโนมัติคอร์เรเลชันคู่ (Dual ACF) [2]

k	ผู้หญิง		ผู้ชาย	
	ACF	Dual ACF	ACF	Dual ACF
1	0.921	0.969	0.884	0.945
2	0.843	0.906	0.758	0.816
3	0.741	0.840	0.582	0.700
4	0.631	0.749	0.354	0.609

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.1 ตัวเข้ารหัส DPCM ที่ใช้ตัวประมาณออกได้คอรืเลขชั้นคู่แบบธรรมดา



รูปที่ 4.2 ตัวถอดรหัส DPCM ที่ใช้ตัวประมาณออกได้คอรืเลขชั้นคู่แบบธรรมดา

สัญญาณผลต่าง $d(n)$ ได้จากการลบสัญญาณประมาณ $s_p(n)$ ออกจากสัญญาณสร้างกลับมาปัจจุบัน $s_r(n)$

$$d(n) = s_r(n) - s_p(n) \quad (4.3)$$

สัญญาณประมาณคือค่าเฉลี่ยของสัญญาณอินพุตตัวถัดไป $s(n+1)$ กับสัญญาณสร้างกลับมาตัวที่แล้ว $s_r(n-1)$

$$s_p(n) = \frac{1}{2} \{s(n+1) + s_r(n-1)\} \quad (4.4)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สัญญาณสร้างกลับมาตัวถัดไปคือ

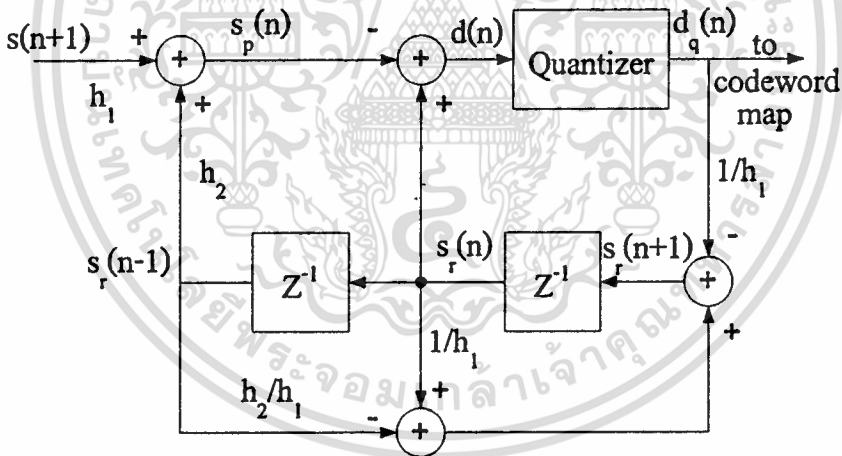
$$s_r(n+1) = -2d_q(n) + 2s_r(n) - s_r(n-1) \quad (4.5)$$

จะได้สัญญาณสร้างกลับมาปัจจุบันคือ

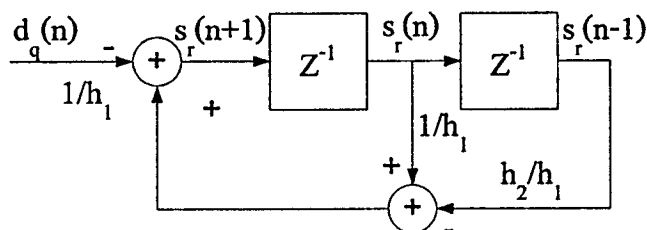
$$s_r(n) = d_q(n) + \frac{1}{2}\{s_r(n+1) + s_r(n-1)\} \quad (4.6)$$

4.3 ตัวประมาณออโต้คอริเลชันแบบเหมาะสมในทางทฤษฎี

ตัวประมาณออโต้คอริเลชันแบบธรรมดาจากหัวข้อ 4.2 ประมาณสัญญาณโดยใช้ค่าสัมประสิทธิ์เป็น $1/2$ แต่ในตัวประมาณออโต้คอริเลชันแบบเหมาะสมเราจะเลือกใช้ค่าสัมประสิทธิ์ที่ทำให้ความผิดพลาดกำลังสองเฉลี่ยมีค่าน้อยที่สุด ตัวเข้ารหัสและตัวถอดรหัส DPCM ที่ใช้ตัวประมาณออโต้คอริเลชันแบบเหมาะสมแสดงดังรูปที่ 4.3 และ 4.4 ตามลำดับ



รูปที่ 4.3 ตัวเข้ารหัส DPCM ที่ใช้ตัวประมาณออโต้คอริเลชันแบบเหมาะสม



รูปที่ 4.4 ตัวถอดรหัส DPCM ที่ใช้ตัวประมาณออโต้คอริเลชันแบบเหมาะสม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 4.3 สัญญาณประมาณคือ

$$s_p(n) = h_1 s(n+1) + h_2 s_r(n-1) \quad (4.7)$$

สัญญาณสร้างกลับมาตัวถัดไปคือ

$$s_r(n+1) = \frac{1}{h_1} \{-d_q(n) + s_r(n) - h_2 s_r(n-1)\} \quad (4.8)$$

จะได้สัญญาณสร้างกลับมาตัวปัจจุบันคือ

$$s_r(n) = d_q(n) + h_1 s_r(n+1) + h_2 s_r(n-1) \quad (4.9)$$

ค่าความผิดพลาดกำลังสองเฉลี่ยคือ

$$\sigma_d^2 = E[d^2(n)] = E\{[s_r(n) - s_p(n)]^2\}$$

แทนค่า $s_p(n)$ จากสมการ (4.7)

$$\sigma_d^2 = E\{[s_r(n) - (h_1 s(n+1) + h_2 s_r(n-1))]\}^2 \quad (4.10)$$

ถ้าไม่คิดผลของนอยส์จากการควอนไทซ์ จะได้ $s_r(n) \approx s(n)$ และ $s_r(n-1) \approx s(n-1)$ ดังนั้น

$$\sigma_d^2 = E\{[s(n) - (h_1 s(n+1) + h_2 s(n-1))]\}^2 \quad (4.11)$$

ทำเช่นเดียวกับสมการ (3.4) จะได้

$$\sigma_d^2 = (1 - 2\rho_1 h_1 - 2\rho_1 h_2 + h_1^2 + 2\rho_2 h_1 h_2 + h_2^2) \sigma_s^2 \quad (4.12)$$

ค่า h_1 และ h_2 ที่ทำให้ค่าความผิดพลาดกำลังสองเฉลี่ยมีค่าน้อยที่สุดหาได้จาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\frac{\partial \sigma_d^2}{\partial h_1} = -2\rho_1 + 2h_1 + 2\rho_2 h_2 = 0$$

$$h_1 + \rho_2 h_2 = \rho_1 \tag{4.13}$$

$$\frac{\partial \sigma_d^2}{\partial h_2} = -2\rho_1 + 2h_2 + 2\rho_2 h_1 = 0$$

$$\rho_2 h_1 + h_2 = \rho_1 \tag{4.14}$$

จากสมการ (4.13) และ (4.14) จะได้

$$h_1 = h_2 = \frac{\rho_1(1-\rho_2)}{1-\rho_2^2} \tag{4.15}$$

จากตารางที่ 3.1 แทนค่า $\rho_1 = 0.86$, $\rho_2 = 0.64$ ลงในสมการ (4.15)

$$h_1 = h_2 = \frac{\rho_1(1-\rho_2)}{1-\rho_2^2} = 0.5244 \tag{4.16}$$

4.4 ตัวประมาณอโต้คอรีเลขันคู่แบบอะแด็ปทีฟ

ตัวประมาณอโต้คอรีเลขันคู่แบบอะแด็ปทีฟใช้วิธีการเช่นเดียวกับหัวข้อ 3.3 คือ

$$\mathbf{h}(n+1) = \mathbf{h}(n) - \frac{1}{2} \mu(n) \nabla_{\mathbf{h}} [E\{d^2(n)\}] \tag{4.17}$$

โดยที่ $\mathbf{h}(n) = \begin{bmatrix} h_1(n) \\ h_2(n) \end{bmatrix}$ $\mu(n)$ คือ Step size และ $\nabla_{\mathbf{h}} = \begin{bmatrix} \frac{\partial}{\partial h_1} \\ \frac{\partial}{\partial h_2} \end{bmatrix}$

จากสมการ (4.9)

$$s_r(n) = d_q(n) + h_1 s_r(n+1) + h_2 s_r(n-1)$$

จะได้
$$d_q(n) = s_r(n) - h_1 s_r(n+1) - h_2 s_r(n-1) \tag{4.18}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังนั้นค่าของเกรเดียนท์คือ

$$\nabla_{\mathbf{h}} [d^2(n)] \approx \nabla_{\mathbf{h}} [d_q^2(n)] = 2d_q(n) \nabla_{\mathbf{h}} d_q(n) = -2d_q(n) \mathbf{s}_r(n) \quad (4.19)$$

$$\text{ซึ่ง } \mathbf{s}_r(n) = \begin{bmatrix} s_r(n+1) \\ s_r(n-1) \end{bmatrix}$$

แทนสมการ (4.19) ลงในสมการ (4.17) จะได้

$$\mathbf{h}(n+1) = \mathbf{h}(n) + \mu(n) d_q(n) \mathbf{s}_r(n) \quad (4.20)$$

โดยที่ $\mu(n) = \frac{g}{100 + \mathbf{s}_r^T(n) \mathbf{s}_r(n)}$ ซึ่ง g คือ ค่าที่ได้จากการทดลอง

4.5 ขอบเขตในการปรับค่าสัมประสิทธิ์ของตัวประมาณออคอร์เลชันคู่

เช่นเดียวกับหัวข้อ 3.4 ขอบเขตในการปรับค่า h หาได้จากผลตอบสนองอิมพัลส์ของระบบ เริ่มจากการแปลงแซคสมการ (4.18) ได้เป็น

$$D_q(z) = S_r(z) - h_1 S_r(z)z - h_2 S_r(z)z^{-1} \quad (4.21)$$

จะได้ฟังก์ชันถ่ายโอนคือ

$$H(z) = \frac{S_r(z)}{D_q(z)} = \frac{1}{1 - h_1 z - h_2 z^{-1}} = \frac{z}{z - h_1 z^2 - h_2} = \frac{z}{(z-a)(z-b)}$$

$$\text{โดยที่ } a, b = \frac{-1 \pm \sqrt{1 - 4(-h_1)(-h_2)}}{2(-h_1)} = \frac{1 \pm \sqrt{1 - 4h_1 h_2}}{2h_1}$$

$$H(z) = \frac{z^{-1}}{(1 - az^{-1})(1 - bz^{-1})}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$= \left(\frac{1}{a-b} \right) \left(\frac{1}{1-az^{-1}} \right) + \left(\frac{1}{b-a} \right) \left(\frac{1}{1-bz^{-1}} \right) \quad (4.22)$$

จะได้

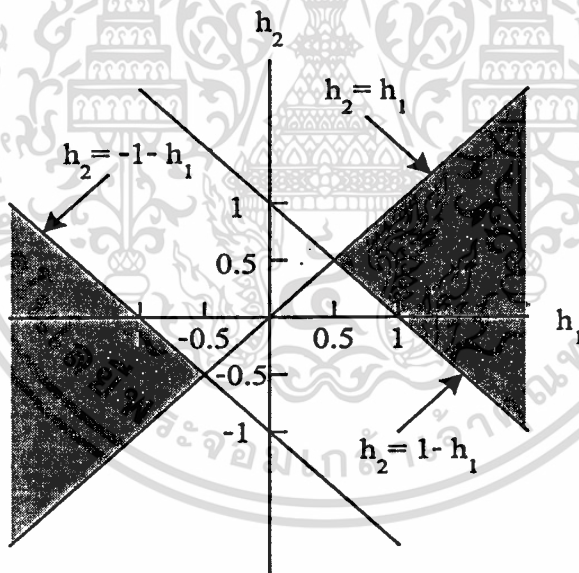
$$h(n) = \left(\frac{1}{a-b} \right) a^n u(n) + \left(\frac{1}{b-a} \right) b^n u(n) \quad (4.23)$$

ระบบจะมีเสถียรภาพเมื่อ $|a| < 1$ และ $|b| < 1$ จะได้ขอบเขตในการปรับค่าสัมประสิทธิ์คือ

$$h_1 > 0.5, 1-h_1 < h_2 < h_1$$

$$h_1 < -0.5, h_1 < h_2 < -1-h_1 \quad (4.24)$$

ซึ่งเป็นบริเวณที่แรเงาในรูปที่ 4.5



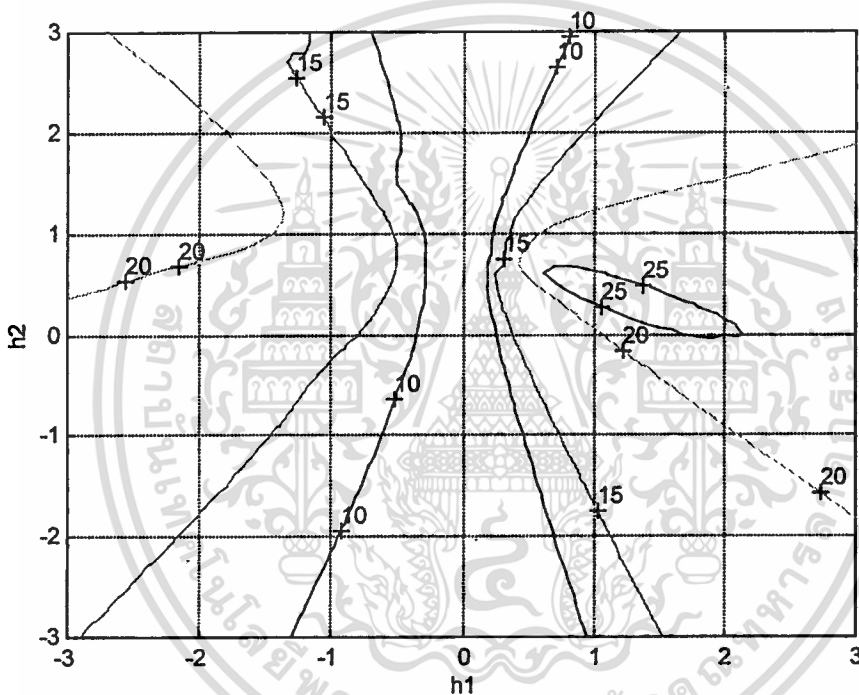
รูปที่ 4.5 ขอบเขตในการปรับค่าสัมประสิทธิ์ของตัวประมาณออโต้คอรีเลชันคู่

ตัวประมาณออโต้คอรีเลชันคู่แบบธรรมดาจากหัวข้อ 4.2 มีค่า $h_1 = h_2 = 0.5$ และตัวประมาณออโต้คอรีเลชันคู่แบบเหมาะสมในทางทฤษฎีจากหัวข้อ 4.3 มีค่า $h_1 = h_2 = 0.5244$ จะเห็นว่าค่าสัมประสิทธิ์ของตัวประมาณทั้งสองแบบอยู่นอกขอบเขตที่ทำให้เกิดเสถียรภาพตามรูปที่ 4.5 ดังนั้นจึงต้องหาค่าสัมประสิทธิ์ที่เหมาะสมจากการทดลอง

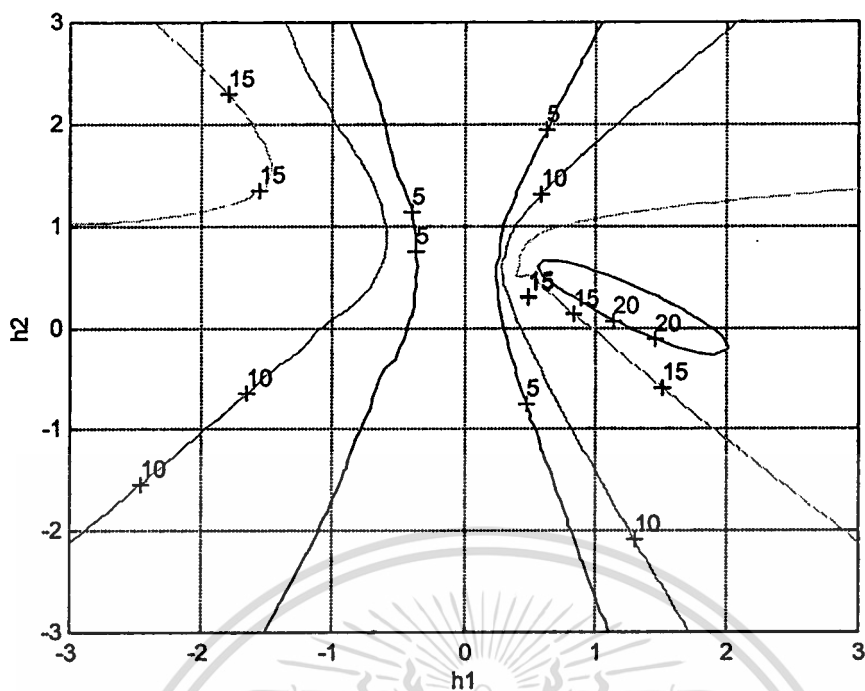
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.6 ตัวประมาณออคอร์รีเลชันแบบเหมาะสมจากการทดลอง

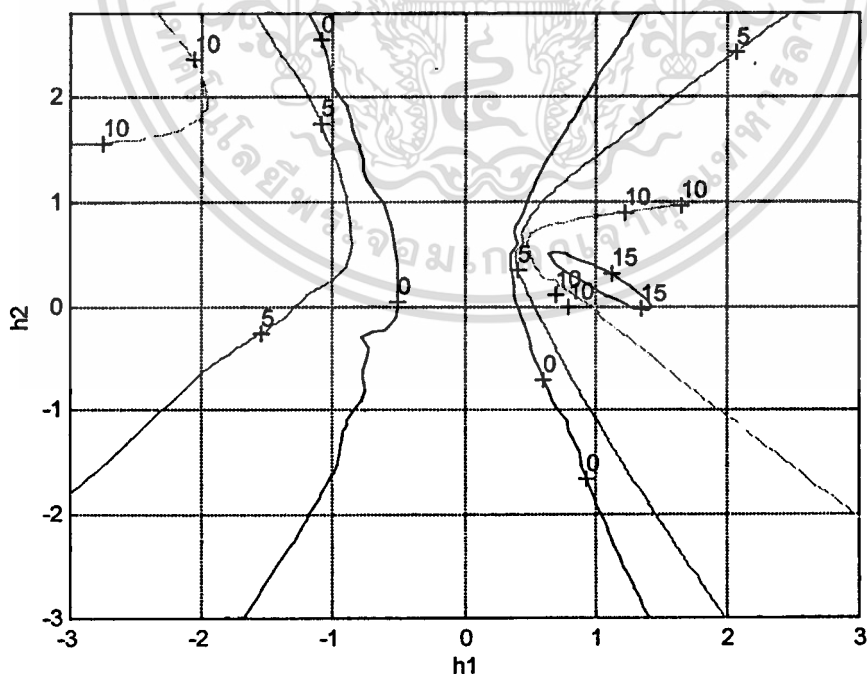
ค่าสัมประสิทธิ์ที่เหมาะสมเป็นค่าสัมประสิทธิ์ที่ทำให้ SNR(Signal to Noise Ratio) มีค่ามากที่สุด ซึ่งหาได้จากการเข้ารหัส ADPCM ไฟล์เสียง 5 ไฟล์แบบ 4 บิต 3 บิต และ 2 บิตต่อแซมเปิ้ล โดยใช้ค่าสัมประสิทธิ์ h_1 และ h_2 ต่างๆกันดังรูปที่ 4.6 ถึง 4.8 จะเห็นว่าบริเวณที่มีค่า SNR สูงที่สุดอยู่ในส่วนที่แรเงาด้านขวาของรูปที่ 4.5 เลือกใช้ค่า $h_1 = 1$ และ $h_2 = 0.35$ ซึ่งเป็นค่าสัมประสิทธิ์ที่ทำให้ SNR เฉลี่ยมีค่ามากที่สุดทั้งสามแบบ



รูปที่ 4.6 SNR เฉลี่ย (dB) เมื่อเข้ารหัส ADPCM 4 บิตต่อแซมเปิ้ลโดยใช้ตัวประมาณออคอร์รีเลชันคู่ซึ่งมีค่า h_1 และ h_2 ต่างๆกัน



รูปที่ 4.7 SNR เฉลี่ย (dB) เมื่อเข้ารหัส ADPCM 3 บิตต่อแซมเปิ้ลโดยใช้ตัวประมาณออคอร์เรลชันคู่ซึ่งมีค่า h_1 และ h_2 ต่างๆกัน



รูปที่ 4.8 SNR เฉลี่ย (dB) เมื่อเข้ารหัส ADPCM 2 บิตต่อแซมเปิ้ลโดยใช้ตัวประมาณออคอร์เรลชันคู่ซึ่งมีค่า h_1 และ h_2 ต่างๆกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

ผลการทดลอง

5.1 กล่าวนำ

ทำการทดลองโดยใช้ซอฟต์แวร์ Turbo C เขียนเป็นโปรแกรมเข้ารหัส ADPCM บนเครื่องคอมพิวเตอร์ส่วนบุคคล Pentium II 300 MHz โดยใช้อะแด็ปทีฟควอนไทเซอร์แบบหน่วยความจำหนึ่งค่าตามหัวข้อ 2.3.2 ร่วมกับตัวประมาณแบบต่างๆ ดังนี้

- 1) Fixed LP2 ใช้ตัวประมาณเชิงเส้นอันดับสองตามหัวข้อ 3.2
- 2) Adaptive LP2 ใช้ตัวประมาณเชิงเส้นอันดับสองแบบอะแด็ปทีฟตามหัวข้อ 3.3
- 3) Fixed Dual 1 ใช้ตัวประมาณออโต้คอรีเลชันคู่แบบธรรมดาตามหัวข้อ 4.2
- 4) Fixed Dual 2 ใช้ตัวประมาณออโต้คอรีเลชันคู่แบบเหมาะสมในทางทฤษฎีตามหัวข้อ 4.3
- 5) Fixed Dual 3 ใช้ตัวประมาณออโต้คอรีเลชันคู่แบบเหมาะสมจากการทดลองตามหัวข้อ 4.6
- 6) Adaptive Dual ใช้ตัวประมาณออโต้คอรีเลชันคู่แบบอะแด็ปทีฟตามหัวข้อ 4.4
- 7) G.726 ใช้ตัวประมาณมาตรฐาน ITU G.726 ตามหัวข้อ 3.5

แต่ละแบบเข้ารหัส 4 บิต 3 บิต และ 2 บิตต่อแซมเปิ้ล สัญญาณอินพุทเป็นไฟล์เสียงนามสกุล WAV ความยาว 60 วินาที 5 ไฟล์ เป็นเสียงผู้ชาย 3 คนและผู้หญิง 2 คน แซมปลิ่ง 8 kHz PCM 8 บิตต่อแซมเปิ้ล แล้ววัดค่า Signal to Noise Ratio(SNR) ผลตอบสนองทางความถี่และเวลาที่ใช้ในการเข้ารหัสเมื่อไม่มีความผิดพลาดจากการส่ง จากนั้นวัดค่า SNR เมื่อมีความผิดพลาดจากการส่ง โดยที่

$$SNR(dB) = 10 \log \frac{\langle s^2(n) \rangle}{\langle (s(n) - s_r(n))^2 \rangle} \quad (5.1)$$

ซึ่ง $\langle \rangle$ คือค่าเฉลี่ย

5.2 ผลการทดลองเมื่อไม่มีความผิดพลาดจากการส่ง

5.2.1 SNR

SNR ของการเข้ารหัส ADPCM เมื่อไม่มีความผิดพลาดจากการส่งแสดงดังตารางที่ 5.1 ถึง

5.3 และ SNR เฉลี่ยแสดงดังรูปที่ 5.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.1 SNR ของการเข้ารหัส ADPCM 4 บิตต่อแซมเปิ้ลเมื่อไม่มีความผิดพลาดจากการส่ง

ไฟล์	ตัวประมาณ						G.726
	Fixed LP2	Adaptive LP2	Fixed Dual 1	Fixed Dual 2	Fixed Dual 3	Adaptive Dual	
ผู้ชาย 1	24.71	26.98	19.17	22.48	26.19	25.40	23.01
ผู้ชาย 2	18.28	19.99	16.10	18.24	23.92	24.96	17.38
ผู้ชาย 3	19.91	22.00	17.53	20.05	26.81	26.05	19.37
ผู้หญิง 1	22.31	23.18	16.96	18.78	24.90	23.80	26.03
ผู้หญิง 2	24.58	24.84	19.96	21.53	27.36	25.26	26.67
เฉลี่ย	21.96	23.20	17.94	20.22	25.84	25.09	22.29

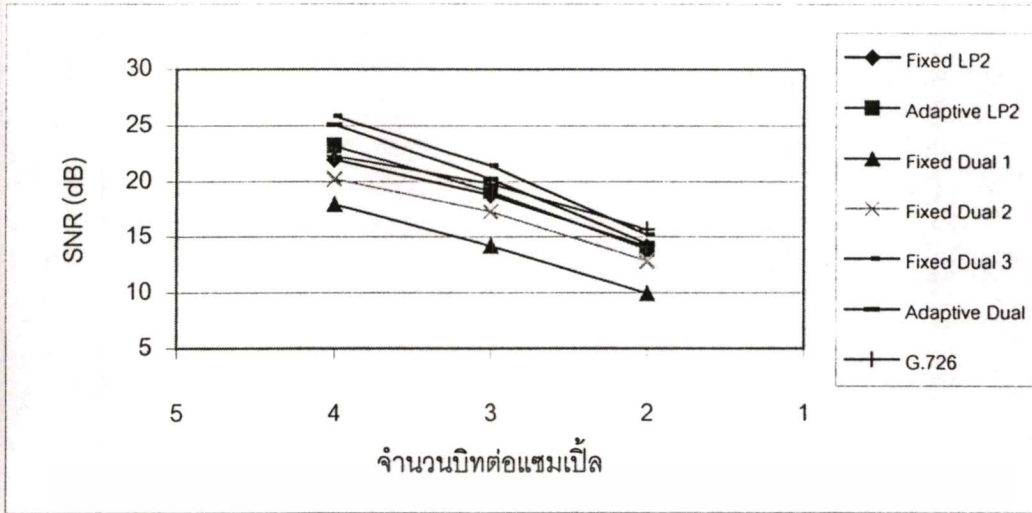
ตารางที่ 5.2 SNR ของการเข้ารหัส ADPCM 3 บิตต่อแซมเปิ้ลเมื่อไม่มีความผิดพลาดจากการส่ง

ไฟล์	ตัวประมาณ						G.726
	Fixed LP2	Adaptive LP2	Fixed Dual 1	Fixed Dual 2	Fixed Dual 3	Adaptive Dual	
ผู้ชาย 1	21.45	21.24	14.53	18.72	21.24	21.27	20.61
ผู้ชาย 2	16.88	17.05	13.17	16.49	21.24	20.08	16.05
ผู้ชาย 3	17.48	18.69	14.11	18.02	22.67	21.00	18.18
ผู้หญิง 1	18.46	18.53	13.95	16.57	20.06	18.72	21.13
ผู้หญิง 2	20.57	19.90	16.05	17.62	21.83	19.96	23.04
เฉลี่ย	18.77	19.08	14.16	17.28	21.41	20.21	19.80

ตารางที่ 5.3 SNR ของการเข้ารหัส ADPCM 2 บิตต่อแซมเปิ้ลเมื่อไม่มีความผิดพลาดจากการส่ง

ไฟล์	ตัวประมาณ						G.726
	Fixed LP2	Adaptive LP2	Fixed Dual 1	Fixed Dual 2	Fixed Dual 3	Adaptive Dual	
ผู้ชาย 1	16.74	16.63	10.11	12.67	15.07	15.85	17.28
ผู้ชาย 2	12.44	13.03	9.54	13.55	16.10	14.83	13.86
ผู้ชาย 3	14.07	14.53	10.42	14.75	16.34	15.24	15.95
ผู้หญิง 1	12.83	12.46	8.95	10.41	13.68	12.55	14.87
ผู้หญิง 2	14.54	13.54	10.77	12.74	14.72	13.46	16.49
เฉลี่ย	14.12	13.84	9.96	12.82	15.18	14.39	15.69

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

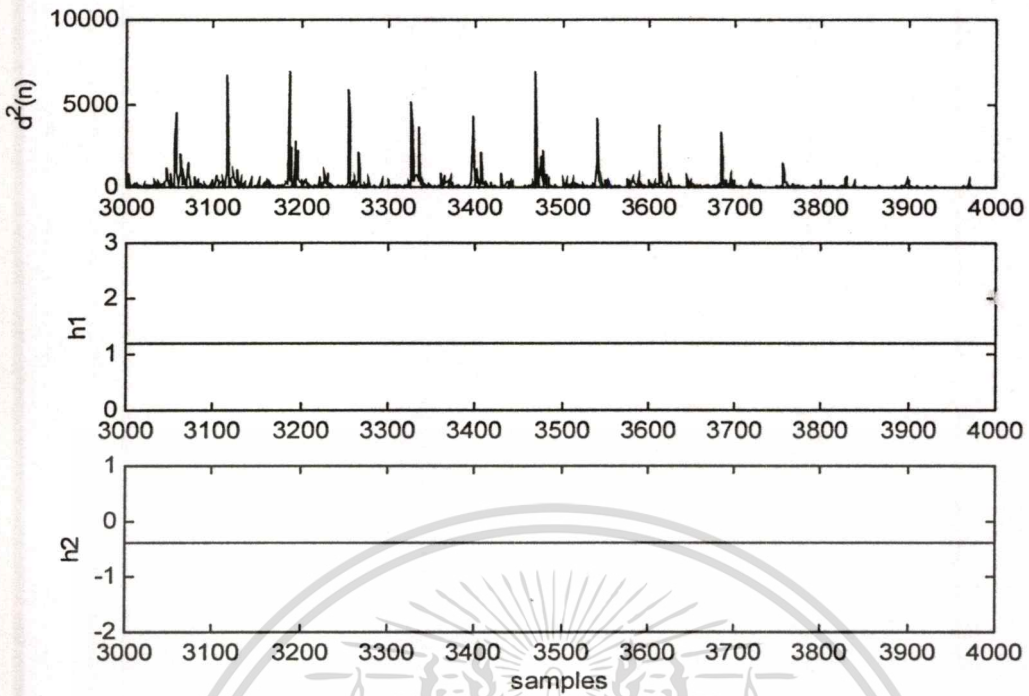


รูปที่ 5.1 SNR เหลือของการเข้ารหัส ADPCM 4 บิต 3 บิต และ 2 บิตต่อแซมเปิ้ลเมื่อไม่มีความผิดพลาดจากการส่ง

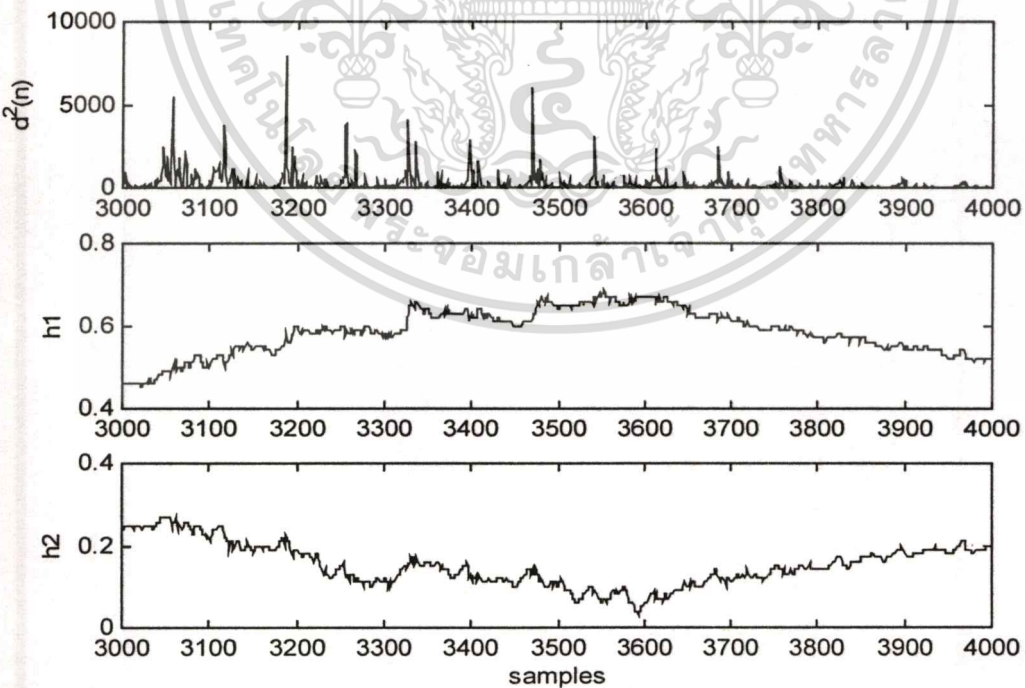
จากรูปที่ 5.1 ค่า SNR ของ Fixed Dual 3 มีค่ามากที่สุดเมื่อเข้ารหัส 4 บิตและ 3 บิตต่อแซมเปิ้ล ในขณะที่เมื่อเข้ารหัส 2 บิตต่อแซมเปิ้ล G.726 จะมีค่ามากที่สุด โดยที่ Fixed Dual 1 มีค่า SNR น้อยที่สุดไม่ว่าจะเข้ารหัส 4 บิต 3 บิต หรือ 2 บิตต่อแซมเปิ้ล จะเห็นว่า SNR ของ Fixed Dual 2 มีค่ามากกว่า Fixed Dual 1 แต่น้อยกว่า Fixed Dual 3 และ SNR ของ Adaptive Dual จะมีค่าใกล้เคียงกับ Fixed Dual 3 ส่วนตัวประมาณ Adaptive LP2 มีค่า SNR มากกว่า Fixed LP2 เมื่อเข้ารหัส 4 บิต แต่จะมีค่าใกล้เคียงกันเมื่อเข้ารหัส 3 บิตและ 2 บิตต่อแซมเปิ้ล

5.2.2 ค่าความผิดพลาดกำลังสองและสัมประสิทธิ์ของตัวประมาณ

ตัวอย่างค่าความผิดพลาดกำลังสองและสัมประสิทธิ์ของตัวประมาณแบบต่างๆเมื่อเข้ารหัส ADPCM 4 บิตของไฟล์เสียงผู้ชาย 1 ระหว่างแซมเปิ้ลที่ 3001 ถึง 4000 แสดงดังรูปที่ 5.2 ถึง 5.8 จะเห็นว่าค่า h_1 และ h_2 ของตัวประมาณแบบ Fixed จะมีค่าคงที่ ในขณะที่ค่า h_1 และ h_2 ของตัวประมาณแบบ Adaptive จะมีค่าเปลี่ยนแปลงตามเวลาของ h_1 ของ Adaptive Dual ซึ่งมีค่าคงที่ดังรูปที่ 5.7



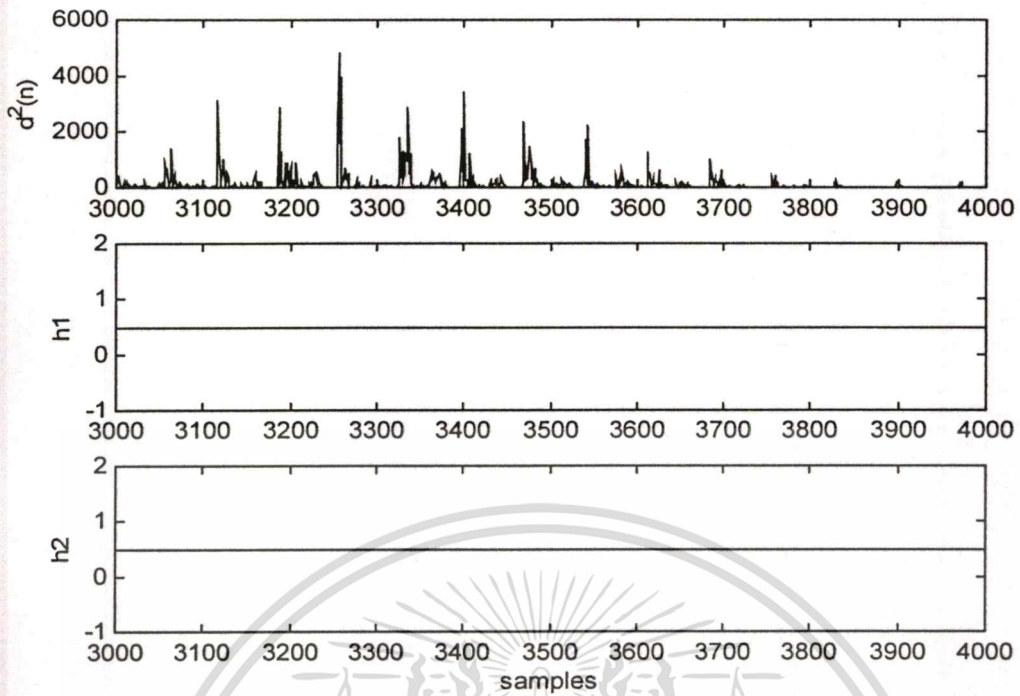
รูปที่ 5.2 ค่าความผิดพลาดกำลังสองและสัมประสิทธิ์ของตัวประมาณ Fixed LP2 เมื่อเข้ารหัส ADPCM 4 บิตของไฟล์เสียงผู้ชาย 1 ระหว่างแซมเปิลที่ 3001 ถึง 4000



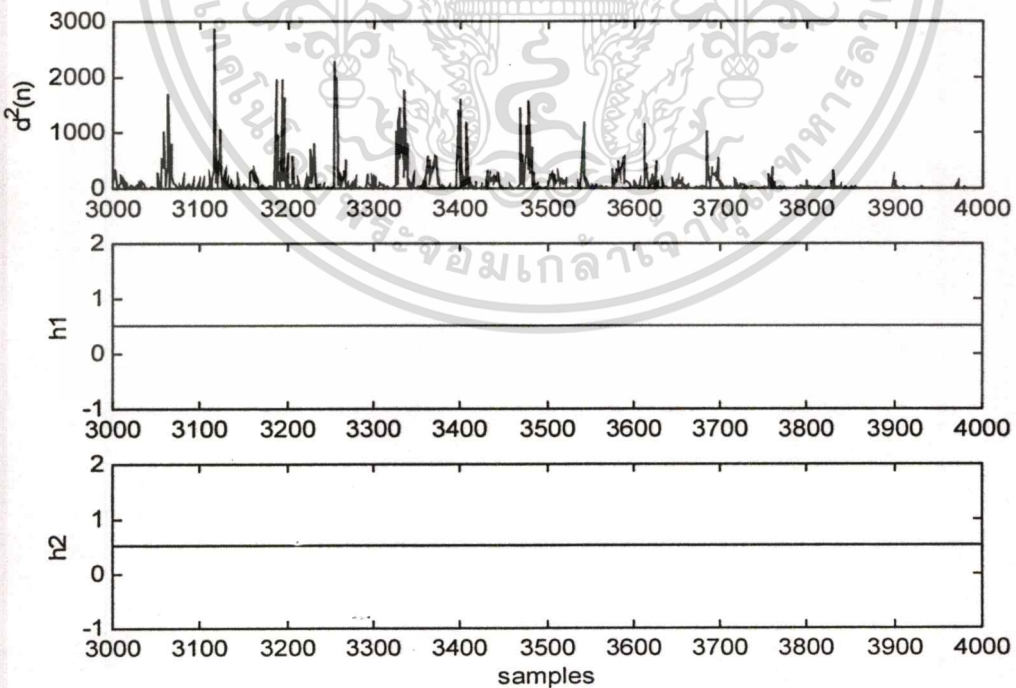
รูปที่ 5.3 ค่าความผิดพลาดกำลังสองและสัมประสิทธิ์ของตัวประมาณ Adaptive LP2 เมื่อเข้ารหัส

ADPCM 4 บิตของไฟล์เสียงผู้ชาย 1 ระหว่างแซมเปิลที่ 3001 ถึง 4000

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

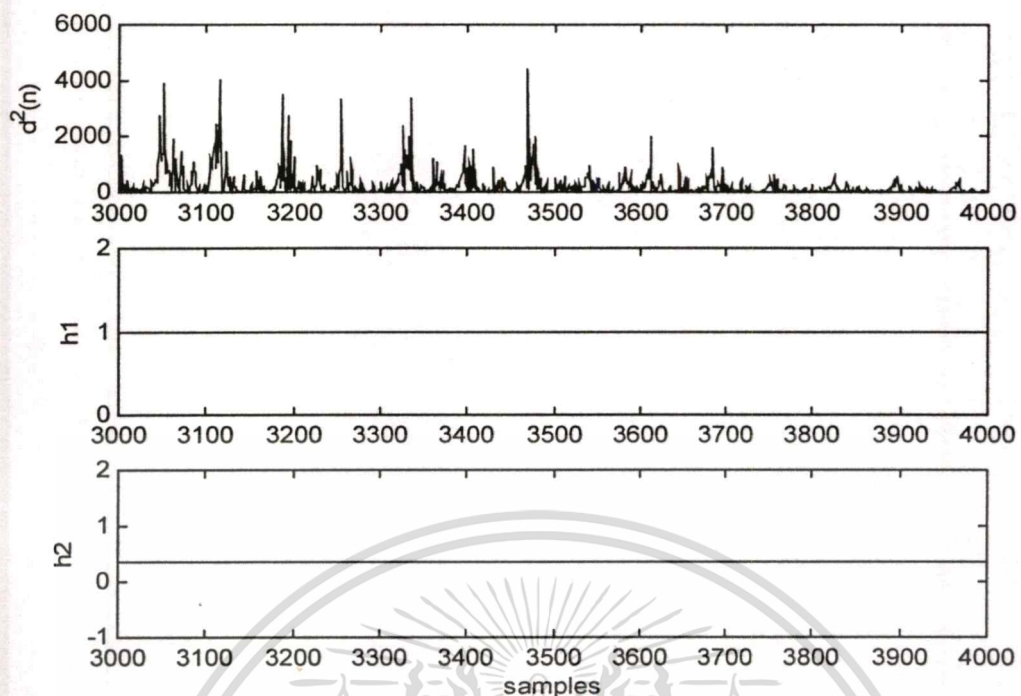


รูปที่ 5.4 ค่าความผิดพลาดกำลังสองและสัมประสิทธิ์ของตัวประมาณ Fixed Dual 1 เมื่อเข้ารหัส ADPCM 4 บิตของไฟล์เสียงผู้ชาย 1 ระหว่างแซมเปิ้ลที่ 3001 ถึง 4000

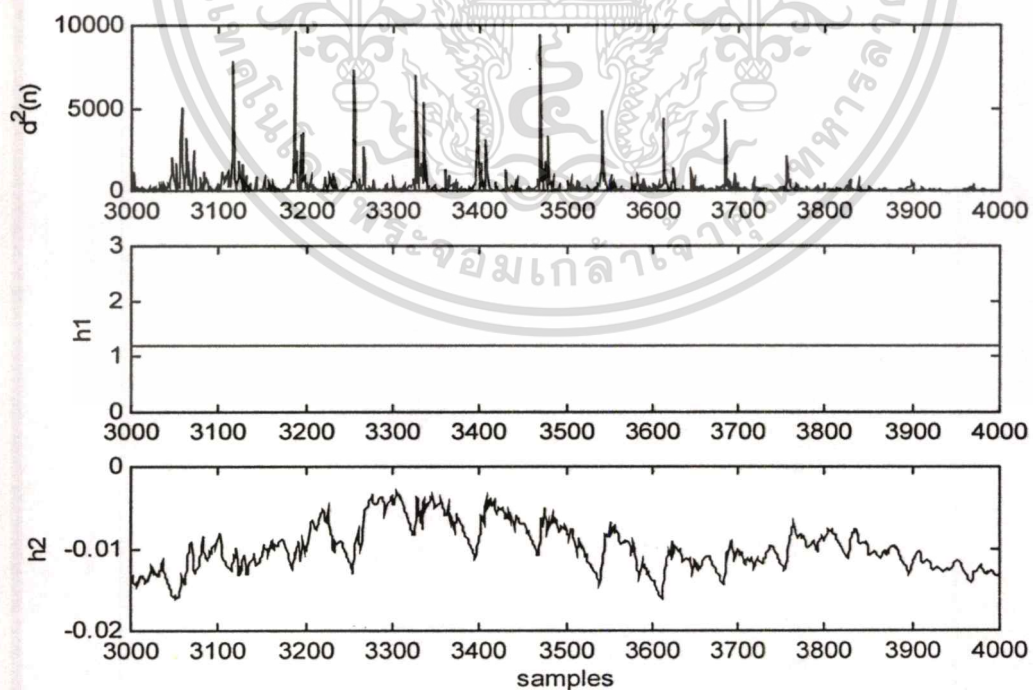


รูปที่ 5.5 ค่าความผิดพลาดกำลังสองและสัมประสิทธิ์ของตัวประมาณ Fixed Dual 2 เมื่อเข้ารหัส ADPCM 4 บิตของไฟล์เสียงผู้ชาย 1 ระหว่างแซมเปิ้ลที่ 3001 ถึง 4000

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

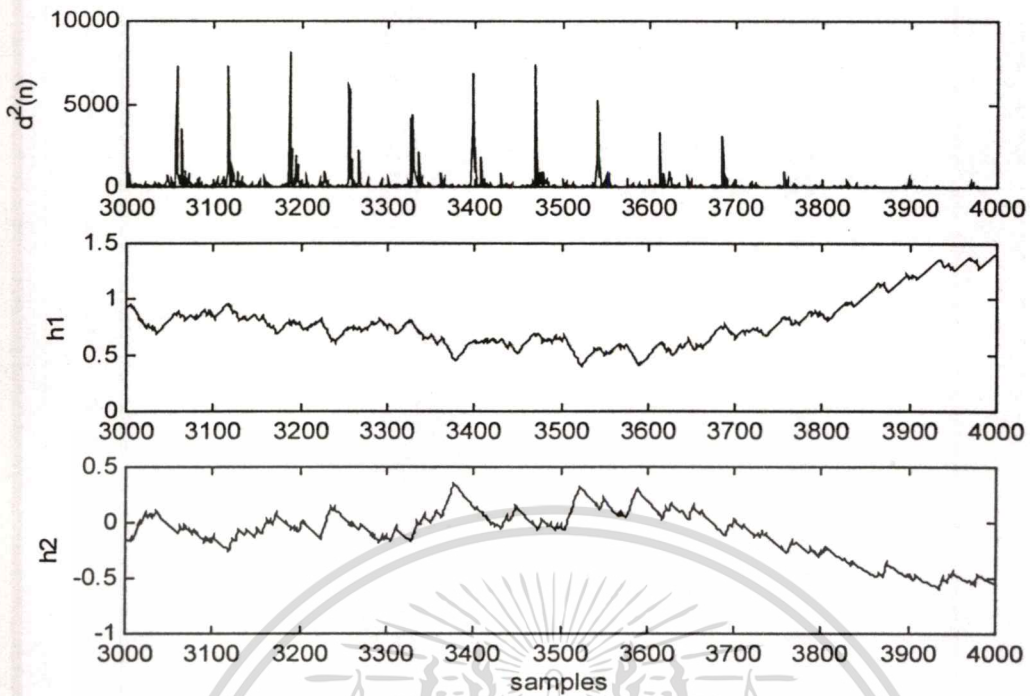


รูปที่ 5.6 ค่าความผิดพลาดกำลังสองและสัมประสิทธิ์ของตัวประมาณ Fixed Dual 3 เมื่อเข้ารหัส ADPCM 4 บิตของไฟล์เสียงผู้ชาย 1 ระหว่างแซมเปิ้ลที่ 3001 ถึง 4000



รูปที่ 5.7 ค่าความผิดพลาดกำลังสองและสัมประสิทธิ์ของตัวประมาณ Adaptive Dual เมื่อเข้ารหัส ADPCM 4 บิตของไฟล์เสียงผู้ชาย 1 ระหว่างแซมเปิ้ลที่ 3001 ถึง 4000

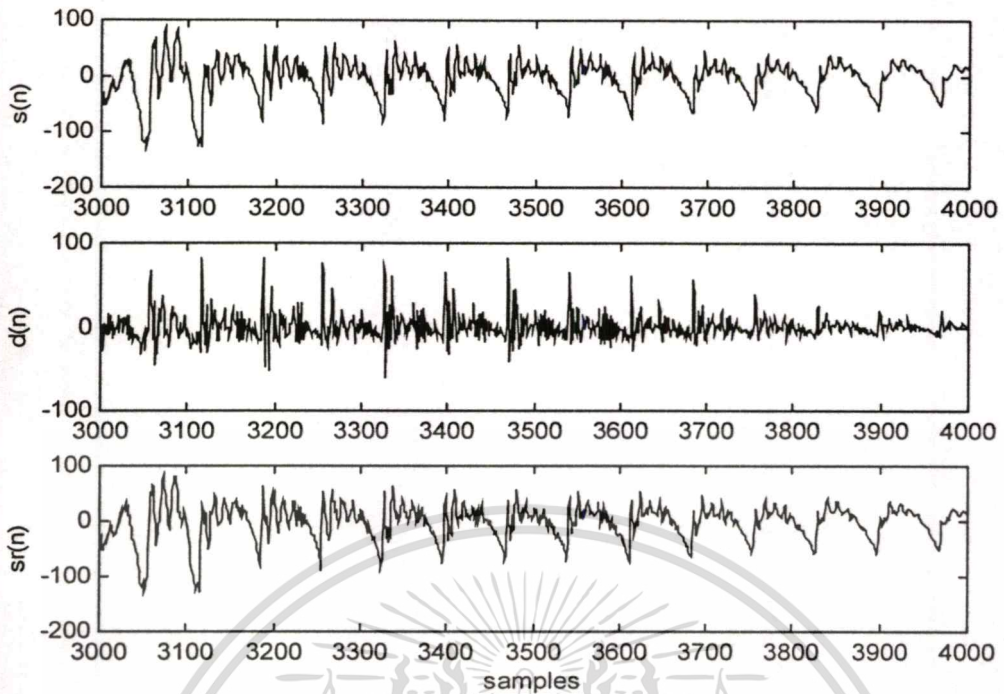
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



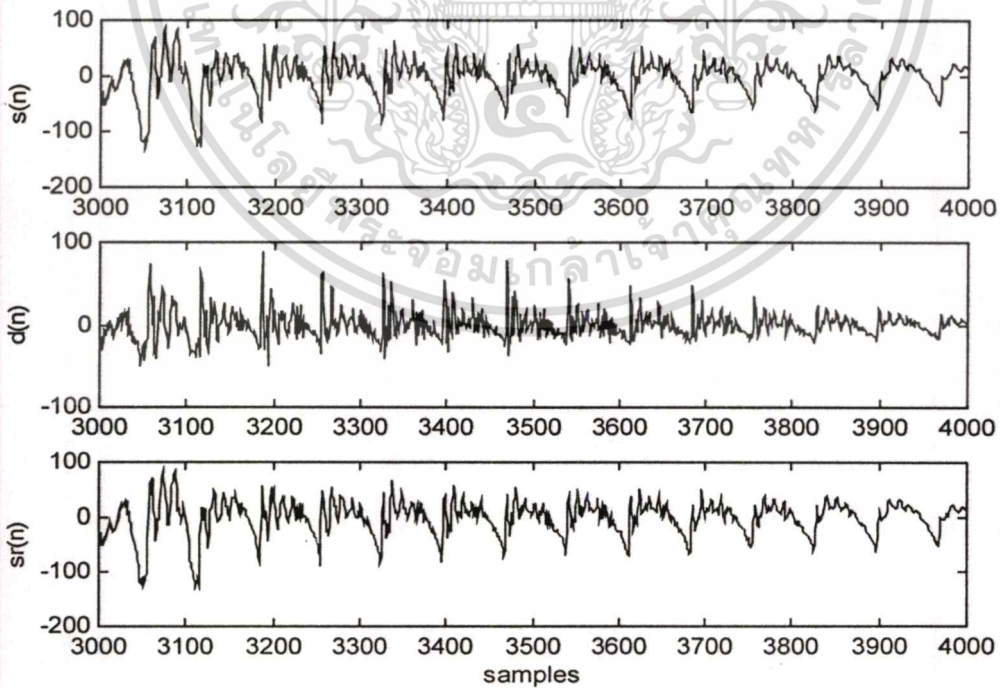
รูปที่ 5.8 ค่าความผิดพลาดกำลังสองและสัมประสิทธิ์ของตัวประมาณ G.726 เมื่อเข้ารหัส ADPCM 4 บิตของไฟล์เสียงผู้ชาย 1 ระหว่างแซมเปิลที่ 3001 ถึง 4000

5.2.3 รูปคลื่นของสัญญาณอินพุต สัญญาณผลต่างและสัญญาณสร้างกลับมา

สัญญาณอินพุต สัญญาณผลต่างและสัญญาณสร้างกลับมาของตัวประมาณแบบต่างๆ เมื่อเข้ารหัส ADPCM 4 บิตของไฟล์เสียงผู้ชาย 1 ระหว่างแซมเปิลที่ 3001 ถึง 4000 แสดงดังรูปที่ 5.9 ถึง 5.15 ซึ่งจะเห็นว่าสัญญาณสร้างกลับมามีความคล้ายคลึงกับสัญญาณอินพุตไม่ว่าจะใช้ตัวประมาณแบบใด ขนาดของ $d(n)$ จะมีค่ามากเมื่อสัญญาณอินพุตเปลี่ยนแปลงเร็ว แต่จะมีค่าน้อยเมื่อสัญญาณอินพุตเปลี่ยนแปลงช้า

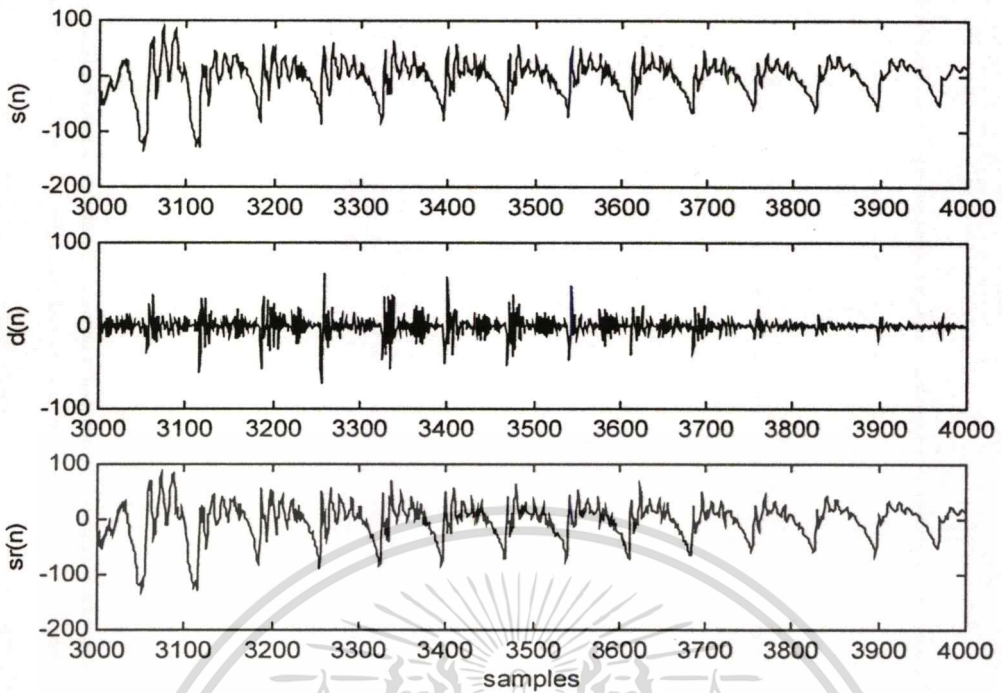


รูปที่ 5.9 สัญญาณอินพุต สัญญาณผลต่างและสัญญาณสร้างกลับมาของตัวประมาณ Fixed LP2 เมื่อเข้ารหัส ADPCM 4 บิตของไฟล์เสียงผู้ชาย 1 ระหว่างแซมเปิลที่ 3001 ถึง 4000



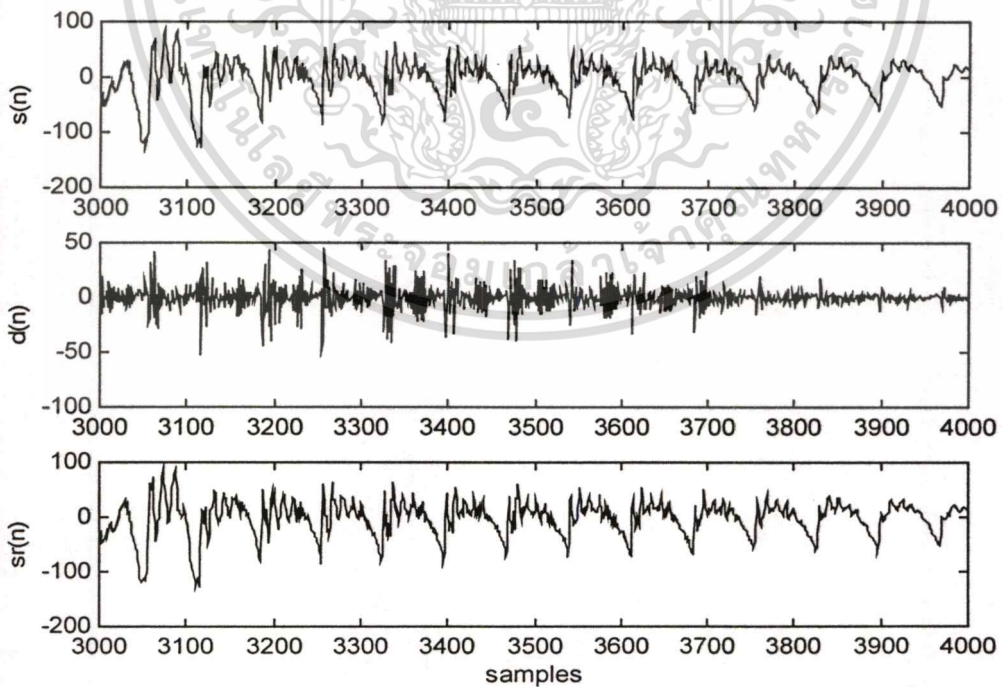
รูปที่ 5.10 สัญญาณอินพุต สัญญาณผลต่างและสัญญาณสร้างกลับมาของตัวประมาณ Adaptive LP2 เมื่อเข้ารหัส ADPCM 4 บิตของไฟล์เสียงผู้ชาย 1 ระหว่างแซมเปิลที่ 3001 ถึง 4000

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.11 สัญญาณอินพุต สัญญาณผลต่างและสัญญาณสร้างกลับมาของตัวประมาณ Fixed Dual

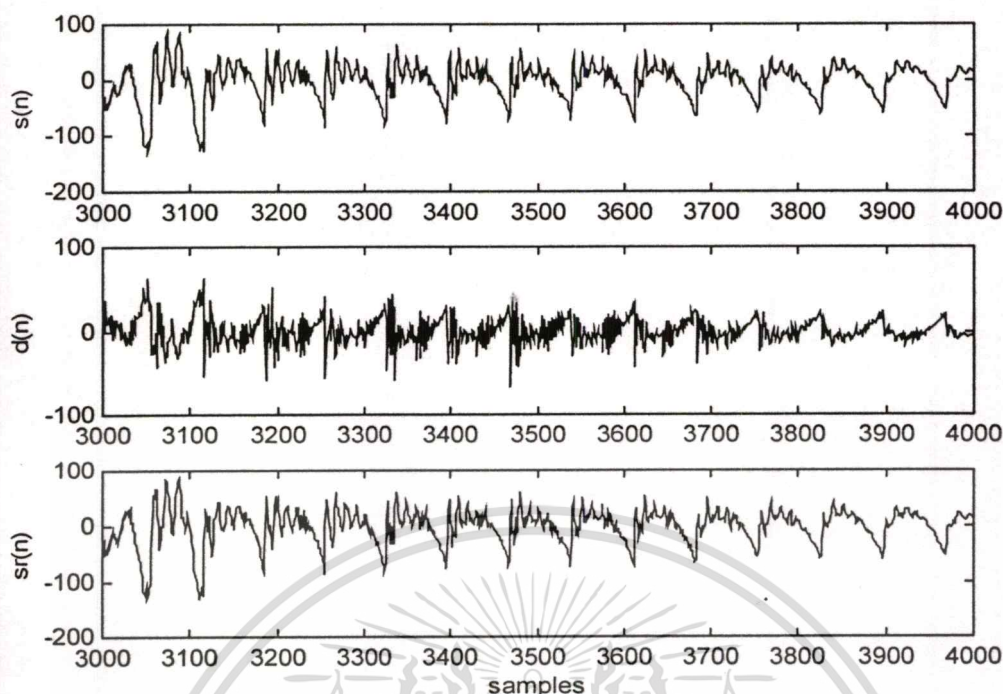
1 เมื่อเข้ารหัส ADPCM 4 บิตของไฟล์เสียงผู้ชาย 1 ระหว่างแซมเปิ้ลที่ 3001 ถึง 4000



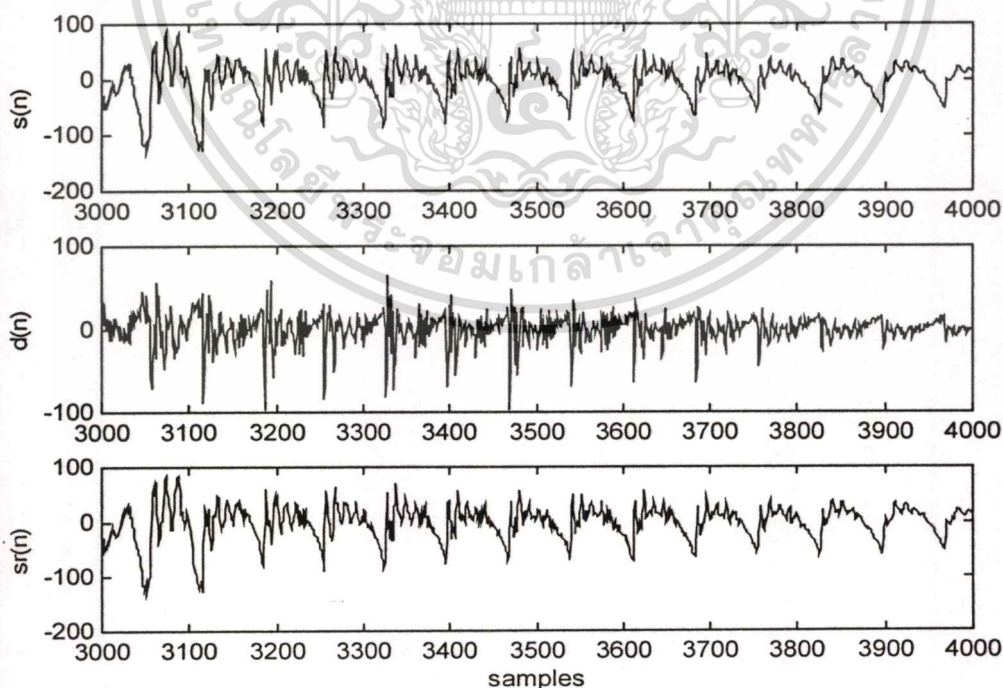
รูปที่ 5.12 สัญญาณอินพุต สัญญาณผลต่างและสัญญาณสร้างกลับมาของตัวประมาณ Fixed Dual

2 เมื่อเข้ารหัส ADPCM 4 บิตของไฟล์เสียงผู้ชาย 1 ระหว่างแซมเปิ้ลที่ 3001 ถึง 4000

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

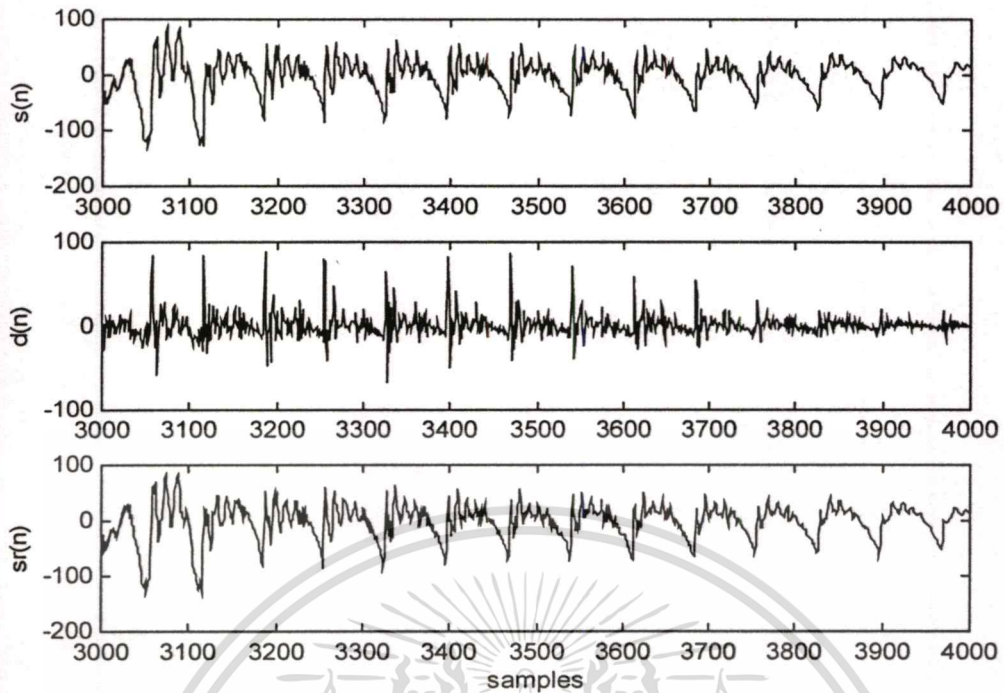


รูปที่ 5.13 สัญญาณอินพุต สัญญาณผลต่างและสัญญาณสร้างกลับมาของตัวประมาณ Fixed Dual 3 เมื่อเข้ารหัส ADPCM 4 บิตของไฟล์เสียงผู้ชาย 1 ระหว่างแซมเปิ้ลที่ 3001 ถึง 4000



รูปที่ 5.14 สัญญาณอินพุต สัญญาณผลต่างและสัญญาณสร้างกลับมาของตัวประมาณ Adaptive Dual เมื่อเข้ารหัส ADPCM 4 บิตของไฟล์เสียงผู้ชาย 1 ระหว่างแซมเปิ้ลที่ 3001 ถึง 4000

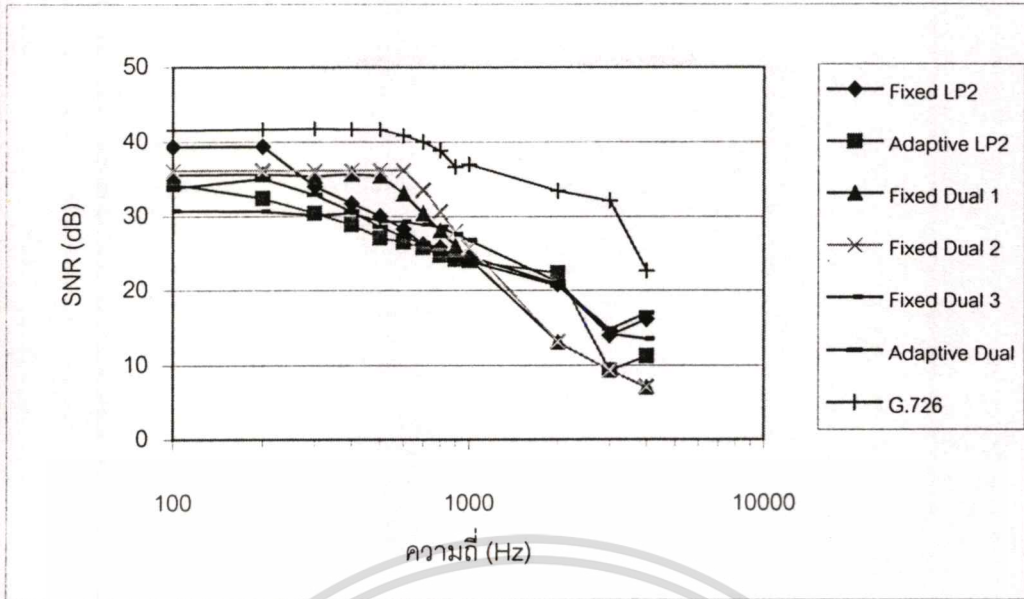
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



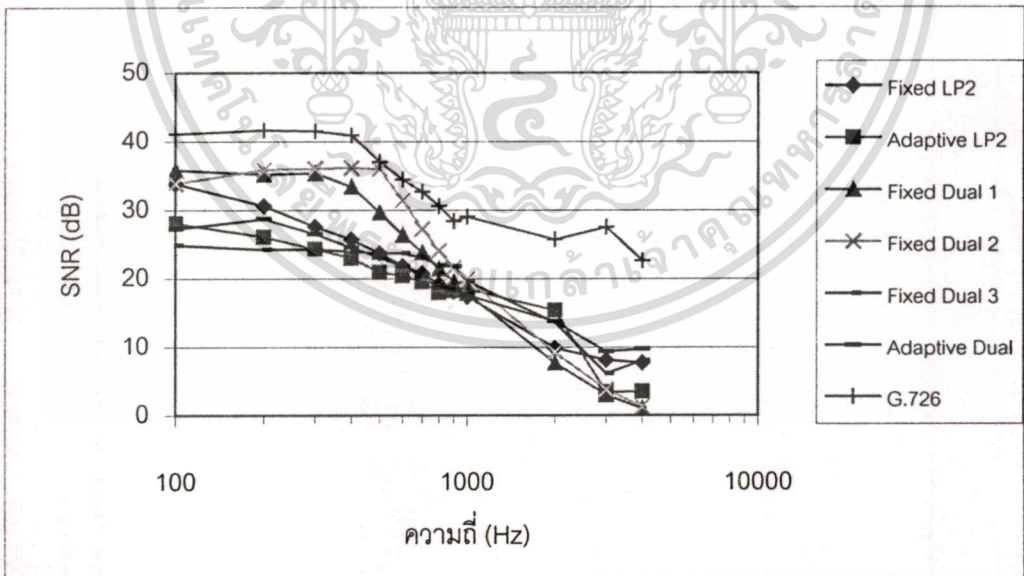
รูปที่ 5.15 สัญญาณอินพุต สัญญาณผลต่างและสัญญาณสร้างกลับมาของตัวประมาณ G.726 เมื่อเข้ารหัส ADPCM 4 บิตของไฟล์เสียงผู้ชาย 1 ระหว่างแซมเปิลที่ 3001 ถึง 4000

5.2.4 ผลตอบสนองทางความถี่

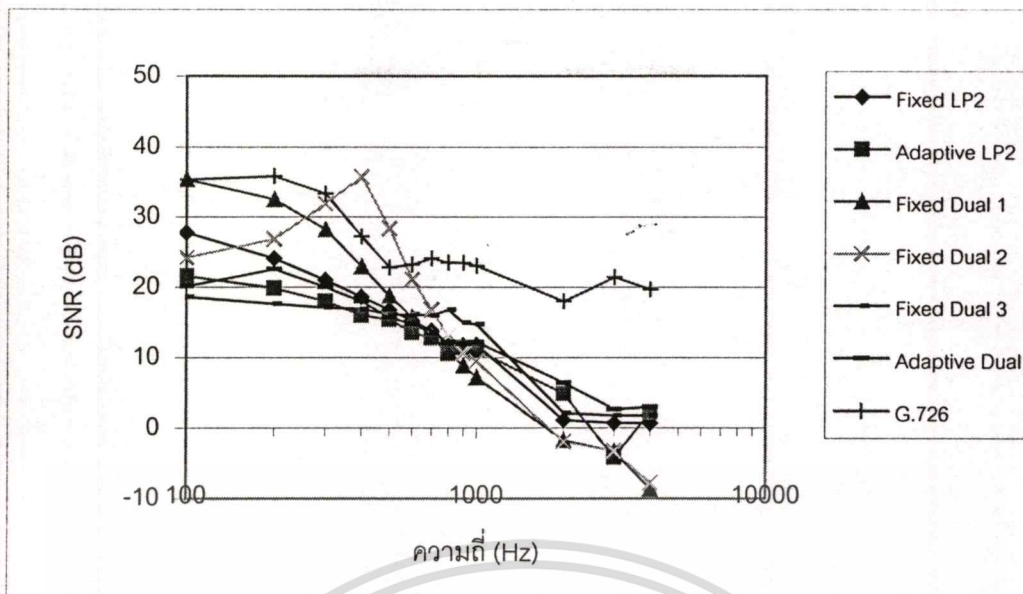
ผลตอบสนองทางความถี่ของตัวประมาณแบบต่างๆ ได้จากการป้อนสัญญาณอินพุตเป็นคลื่นรูปไซน์ในช่วงความถี่เสียงพูดแล้ววัด SNR ออกมาดังรูปที่ 5.16 ถึง 5.18 ซึ่งจะเห็นว่า มีลักษณะเป็นตัวกรองแบบผ่านความถี่ต่ำ ตัวประมาณ G.726 มี SNR มากที่สุดทุกช่วงความถี่เมื่อเข้ารหัส 4 บิตและ 3 บิตต่อแซมเปิล เมื่อเข้ารหัส 2 บิตต่อแซมเปิลที่ความถี่ 100 Hz Fixed Dual 1 และ G.726 จะมีค่าใกล้เคียงกัน ที่ความถี่ 400-500 Hz ตัวประมาณ Fixed Dual 2 มีค่า SNR มากที่สุด ส่วนที่ความถี่อื่น ตัวประมาณ G.726 ยังคงมีค่า SNR มากที่สุดเช่นเดิม



รูปที่ 5.16 ผลตอบสนองทางความถี่ของตัวประมาณแบบต่างๆ เมื่อเข้ารหัส ADPCM 4 บิตต่อแซมเปิล



รูปที่ 5.17 ผลตอบสนองทางความถี่ของตัวประมาณแบบต่างๆ เมื่อเข้ารหัส ADPCM 3 บิตต่อแซมเปิล



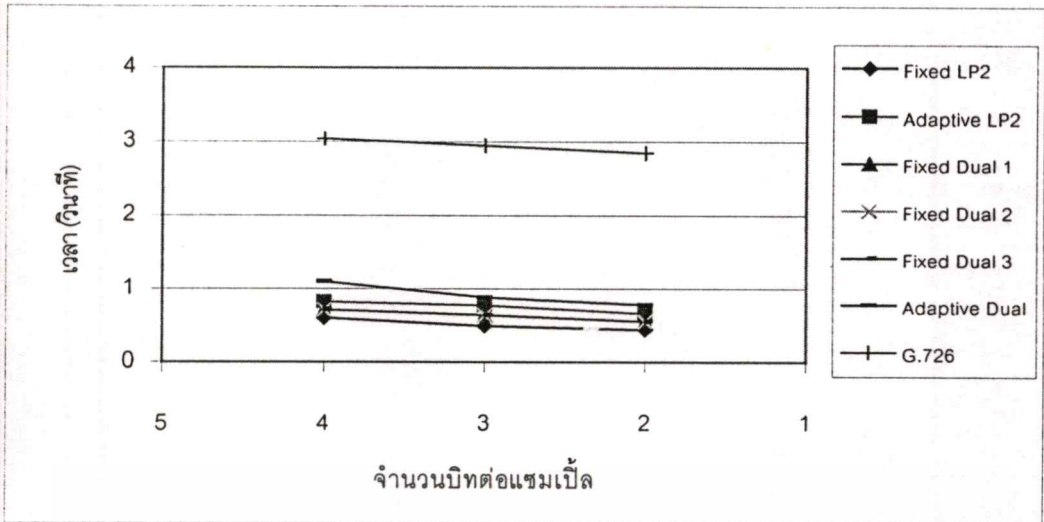
รูปที่ 5.18 ผลตอบสนองทางความถี่ของตัวประมาณแบบต่างๆ เมื่อเข้ารหัส ADPCM 2 บิตต่อแซมเปิ้ล

5.2.5 เวลาที่ใช้ในการเข้ารหัส

เวลาเฉลี่ยที่ใช้ในการเข้ารหัส ADPCM แต่ละแบบแสดงดังตารางที่ 5.4 และรูปที่ 5.19

ตารางที่ 5.4 เวลาเฉลี่ยที่ใช้ในการเข้ารหัส ADPCM (วินาที)

บิตต่อแซมเปิ้ล	ตัวประมาณ						
	Fixed LP2	Adaptive LP2	Fixed Dual 1	Fixed Dual 2	Fixed Dual 3	Adaptive Dual	G.726
4	0.60	0.82	0.71	0.71	0.71	1.10	3.04
3	0.49	0.77	0.64	0.64	0.64	0.88	2.95
2	0.44	0.66	0.55	0.55	0.55	0.78	2.85



รูปที่ 5.19 เวลาเฉลี่ยที่ใช้ในการเข้ารหัส ADPCM

จากตารางที่ 5.4 และรูปที่ 5.19 ค่าสัมประสิทธิ์ของตัวประมาณ Fixed LP2 และ Fixed Dual ทั้งสามแบบเป็นค่าคงที่ จึงใช้เวลาเข้ารหัสน้อย โดยที่ Fixed LP2 จะใช้เวลาน้อยกว่า Fixed Dual ทั้งสามแบบ เมื่อมีการปรับค่าสัมประสิทธิ์ก็จะใช้เวลาเข้ารหัสมากขึ้น ดังจะเห็นได้จากตัวประมาณ Adaptive LP2, Adaptive Dual และ G.726 ซึ่ง G.726 ใช้เวลาเข้ารหัสมากที่สุดเนื่องจากการปรับค่าสัมประสิทธิ์ซับซ้อนที่สุด

5.3 ผลการทดลองเมื่อมีความผิดพลาดจากการส่ง

SNR ของการเข้ารหัส ADPCM 4 บิต 3 บิต และ 2 บิตต่อแรมเปิ้ลเมื่อมีความผิดพลาดจากการส่งแสดงดังตารางที่ 5.5 ถึง 5.7 ตามลำดับ โดยใช้ค่าอัตราบิตผิดพลาด(Bit Error Rate:BER) เท่ากับ 0 0.0001 0.001 และ 0.01 กราฟของค่า SNR เฉลี่ยแสดงดังรูปที่ 5.20 ถึง 5.22 ตามลำดับ เมื่อเข้ารหัส 4 บิตและ 3 บิตต่อแรมเปิ้ล Fixed Dual 3 และ Adaptive Dual มี SNR มากที่สุดและที่ BER=0.01 มีค่าใกล้เคียงกับ Adaptive LP2 เมื่อเข้ารหัส 2 บิตต่อแรมเปิ้ลที่ BER ต่ำ Fixed Dual 3 และ G.726 มี SNR มากที่สุด ที่ BER สูง ตัวประมาณทุกแบบมีค่า SNR ใกล้เคียงกัน ยกเว้นตัวประมาณ Fixed Dual 1 และ Fixed Dual 2 ซึ่งมี SNR ต่ำมากเมื่อมีความผิดพลาดจากการส่งไม่ว่าจะเข้ารหัสกี่บิตก็ตามทั้งนี้เพราะตัวประมาณทั้งสองแบบมีค่า $h_1 = h_2$ ซึ่งอยู่นอกขอบเขตที่ทำให้เกิดเสถียรภาพตามรูปที่ 4.5

ตารางที่ 5.5 SNR ของการเข้ารหัส ADPCM 4 บิตต่อแซมเปิลเมื่อมีความผิดพลาดจากการส่ง (dB)

Fixed LP2						
BER	ไฟล์เสียง					
	male1	male2	male3	female1	female2	average
0	24.42	16.73	18.63	21.78	24.12	21.14
0.0001	23.49	16.60	18.37	20.99	22.93	20.48
0.001	17.95	16.62	16.60	16.66	17.08	16.58
0.01	8.45	10.62	10.46	6.42	7.70	8.73
Adaptive LP2						
BER	ไฟล์เสียง					
	male1	male2	male3	female1	female2	average
0	26.83	18.52	20.91	23.05	24.61	22.58
0.0001	23.97	18.53	20.51	22.01	23.21	21.65
0.001	18.41	17.25	18.21	16.56	17.39	17.56
0.01	9.51	11.87	11.24	7.76	8.33	9.74
Fixed Dual 1						
BER	ไฟล์เสียง					
	male1	male2	male3	female1	female2	average
0	19.15	14.78	16.97	16.44	19.84	17.44
0.0001	-9.56	-2.24	-4.88	-6.99	-6.46	-6.03
0.001	-9.71	-4.27	-6.36	-7.35	-6.73	-6.68
0.01	-10.62	-6.37	-6.37	-8.35	-7.75	-7.69
Fixed Dual 2						
BER	ไฟล์เสียง					
	male1	male2	male3	female1	female2	average
0	22.43	17.69	19.71	18.45	21.39	19.93
0.0001	-0.73	7.99	7.27	1.22	1.36	3.42
0.001	-4.46	2.41	1.74	-2.76	-1.68	-0.95
0.01	-6.81	-0.89	-1.56	-6.18	-4.06	-3.70
Fixed Dual 3						
BER	ไฟล์เสียง					
	male1	male2	male3	female1	female2	average
0	26.23	23.01	26.60	24.60	27.39	26.57
0.0001	24.72	22.62	26.37	22.67	24.72	24.02
0.001	18.65	20.13	20.82	16.86	17.96	18.88
0.01	9.43	12.47	11.38	7.38	8.84	9.90
Adaptive Dual						
BER	ไฟล์เสียง					
	male1	male2	male3	female1	female2	average
0	26.49	23.75	26.86	23.83	26.48	24.88
0.0001	23.53	23.23	24.58	22.67	24.18	23.64
0.001	17.54	20.47	20.67	17.29	18.62	18.92
0.01	8.67	11.93	11.10	8.49	9.39	9.92
G.726						
BER	ไฟล์เสียง					
	male1	male2	male3	female1	female2	average
0	22.75	16.67	18.87	24.67	26.35	21.86
0.0001	20.69	16.47	18.31	21.79	23.14	20.08
0.001	16.50	14.71	16.43	16.58	16.31	16.51
0.01	6.35	9.35	8.26	6.49	6.34	7.16

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.6 SNR ของการเข้ารหัส ADPCM 3 บิตต่อแซมเปิลเมื่อมีความผิดพลาดจากการส่ง (dB)

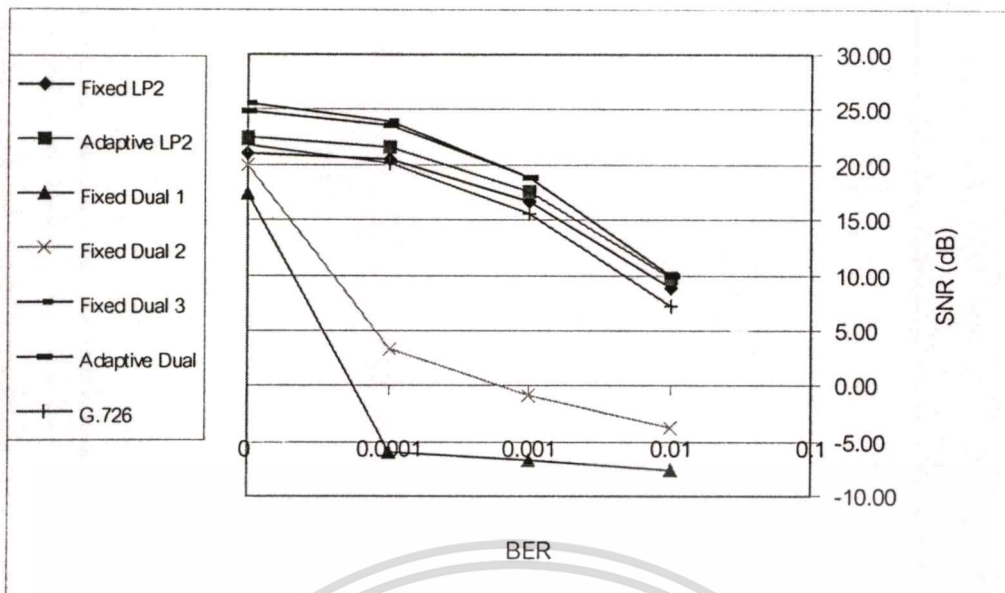
Fixed LP2						
BER	ไฟล์เสียง					
	male1	male2	male3	female1	female2	average
0	21.40	14.15	16.06	18.01	19.95	17.91
0.0001	20.25	14.11	16.90	17.61	19.45	17.46
0.001	16.55	13.65	14.87	14.77	16.93	16.15
0.01	8.22	10.37	10.24	7.11	8.02	8.79
Adaptive LP2						
BER	ไฟล์เสียง					
	male1	male2	male3	female1	female2	average
0	21.56	16.56	17.68	18.43	19.79	18.60
0.0001	21.04	16.50	17.48	18.07	19.33	18.28
0.001	18.11	14.92	16.19	16.57	16.49	16.26
0.01	9.72	11.00	10.98	8.17	9.36	9.85
Fixed Dual 1						
BER	ไฟล์เสียง					
	male1	male2	male3	female1	female2	average
0	14.38	11.90	13.22	13.03	14.40	13.39
0.0001	-9.53	-1.29	-4.36	-6.87	-6.23	-6.66
0.001	-9.74	-3.91	-6.23	-7.30	-6.65	-6.57
0.01	-10.60	-4.99	-6.21	-8.30	-7.65	-7.55
Fixed Dual 2						
BER	ไฟล์เสียง					
	male1	male2	male3	female1	female2	average
0	18.95	16.81	17.57	14.80	17.00	16.83
0.0001	-0.31	10.91	9.74	2.07	1.20	4.72
0.001	-4.32	4.45	3.10	-1.92	-1.19	0.02
0.01	-6.80	-0.32	-1.16	-6.04	-3.66	-3.40
Fixed Dual 3						
BER	ไฟล์เสียง					
	male1	male2	male3	female1	female2	average
0	21.70	20.36	22.48	19.80	21.73	21.21
0.0001	21.20	20.14	21.92	19.38	21.01	20.73
0.001	17.57	18.34	18.98	16.17	17.23	17.66
0.01	9.24	11.84	11.21	8.03	9.35	9.93
Adaptive Dual						
BER	ไฟล์เสียง					
	male1	male2	male3	female1	female2	average
0	20.64	19.97	20.80	18.43	19.51	19.87
0.0001	20.21	19.77	20.48	18.00	19.12	19.52
0.001	17.03	18.11	18.43	16.80	16.50	17.17
0.01	9.35	11.38	10.91	8.63	9.41	9.94
G.726						
BER	ไฟล์เสียง					
	male1	male2	male3	female1	female2	average
0	20.29	14.95	17.19	20.63	22.51	19.11
0.0001	19.39	14.84	16.93	19.61	20.99	18.35
0.001	16.18	13.90	14.77	16.33	16.87	16.01
0.01	6.50	9.25	8.17	6.40	7.04	7.47

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

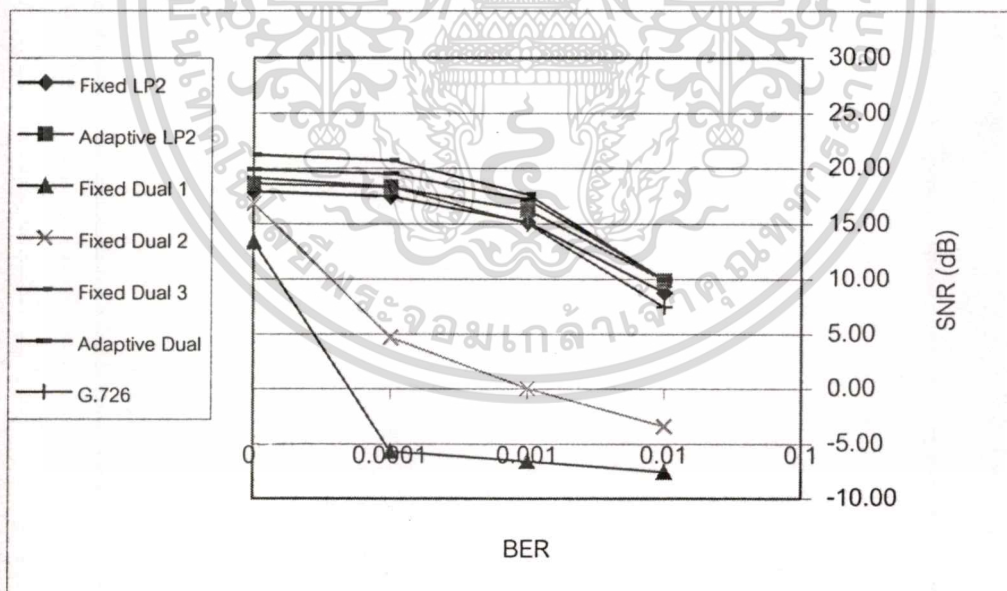
ตารางที่ 5.7 SNR ของการเข้ารหัส ADPCM 2 บิตต่อแซมเปิลเมื่อมีความผิดพลาดจากการส่ง (dB)

Fixed LP2						
BER	ไฟล์เสียง					
	male1	male2	male3	female1	female2	average
0	16.90	11.26	13.33	12.56	14.11	13.63
0.0001	16.32	11.21	13.19	12.30	13.75	13.35
0.001	12.72	10.68	11.98	10.38	11.40	11.43
0.01	3.59	7.47	6.50	3.38	3.79	4.95
Adaptive LP2						
BER	ไฟล์เสียง					
	male1	male2	male3	female1	female2	average
0	16.88	12.15	13.87	12.23	13.27	13.48
0.0001	16.43	12.08	13.70	12.04	13.01	13.25
0.001	12.58	11.44	12.41	10.45	11.24	11.62
0.01	3.85	7.83	6.70	3.87	4.37	6.32
Fixed Dual 1						
BER	ไฟล์เสียง					
	male1	male2	male3	female1	female2	average
0	10.29	8.59	9.71	8.81	10.40	9.56
0.0001	-9.49	-0.37	-3.50	-6.46	-6.02	-6.17
0.001	-9.67	-3.50	-4.87	-7.28	-6.72	-6.41
0.01	-10.48	-4.65	-6.08	-8.42	-7.67	-7.46
Fixed Dual 2						
BER	ไฟล์เสียง					
	male1	male2	male3	female1	female2	average
0	13.36	13.13	14.57	10.29	12.69	12.81
0.0001	-0.40	10.96	10.20	2.13	2.45	6.07
0.001	-4.16	4.32	2.90	-2.76	-1.37	-0.21
0.01	-6.74	-0.53	-1.71	-6.72	-4.47	-3.83
Fixed Dual 3						
BER	ไฟล์เสียง					
	male1	male2	male3	female1	female2	average
0	16.71	16.68	16.18	13.45	14.49	16.10
0.0001	16.27	16.48	16.90	13.17	14.20	14.80
0.001	12.32	14.03	13.94	11.48	12.18	12.79
0.01	3.70	8.27	7.06	4.48	4.96	6.69
Adaptive Dual						
BER	ไฟล์เสียง					
	male1	male2	male3	female1	female2	average
0	14.65	14.33	14.38	12.00	12.64	13.60
0.0001	14.29	14.19	14.19	11.86	12.43	13.39
0.001	11.84	13.00	12.74	10.57	11.04	11.84
0.01	3.79	8.07	6.95	4.52	4.96	6.66
G.726						
BER	ไฟล์เสียง					
	male1	male2	male3	female1	female2	average
0	17.17	12.53	14.76	14.83	16.30	16.12
0.0001	16.41	12.43	14.48	14.44	16.69	14.69
0.001	12.41	11.53	12.65	11.85	12.60	12.21
0.01	3.53	7.34	6.29	3.78	4.39	6.07

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

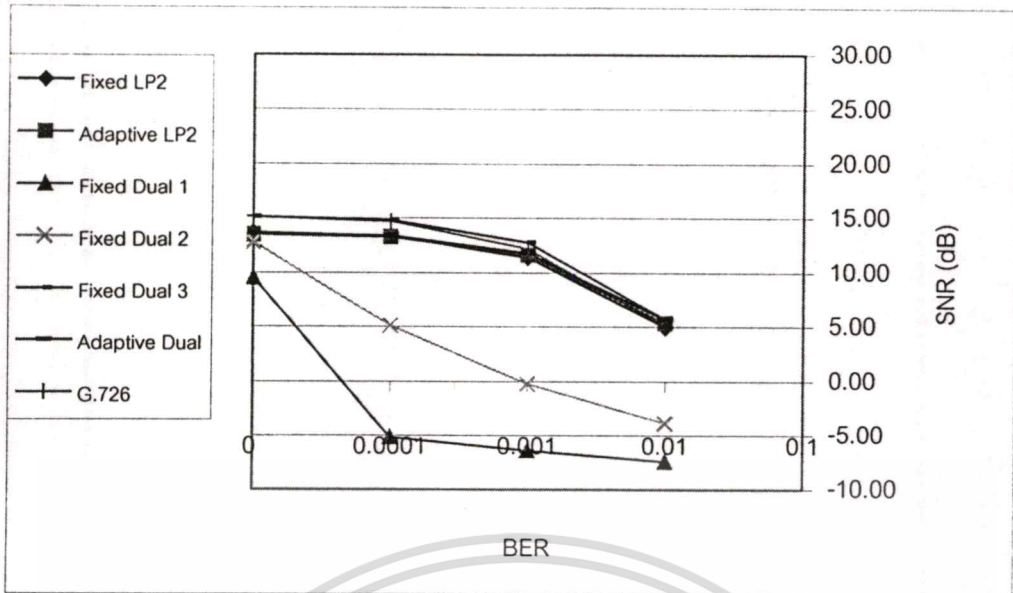


รูปที่ 5.20 SNR เฉลี่ยของการเข้ารหัส ADPCM 4 บิตต่อแชนเนลเมื่อมีความผิดพลาดจากการส่ง



รูปที่ 5.21 SNR เฉลี่ยของการเข้ารหัส ADPCM 3 บิตต่อแชนเนลเมื่อมีความผิดพลาดจากการส่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.22 SNR เล็กลงของการเข้ารหัส ADPCM 2 บิตต่อแอมป์เปิดเมื่อมีความผิดพลาดจากการส่ง



จากผลการทดลองต่างๆที่ผ่านมา สามารถจัดลำดับตัวประมาณตามค่าที่ใช้วัดได้ดังตารางที่ 5.8 ถึง 5.10 โดยกำหนดให้ 1 เป็นค่าที่ดีที่สุด

ตารางที่ 5.8 ลำดับของตัวประมาณตามค่าที่ใช้วัดเมื่อเข้ารหัส 4 บิตต่อแซมเปิล

ค่าที่ใช้วัด	Fixed LP2	Adaptive LP2	Fixed Dual 1	Fixed Dual 2	Fixed Dual 3	Adaptive Dual	G.726
SNR เมื่อไม่มีความผิดพลาดจากการส่ง	5	3	7	6	1	2	4
SNR เมื่อมีความผิดพลาดจากการส่ง	4	3	7	6	1	2	5
เวลาที่ใช้เข้ารหัส	1	3	2	2	2	4	5

ตารางที่ 5.9 ลำดับของตัวประมาณตามค่าที่ใช้วัดเมื่อเข้ารหัส 3 บิตต่อแซมเปิล

ค่าที่ใช้วัด	Fixed LP2	Adaptive LP2	Fixed Dual 1	Fixed Dual 2	Fixed Dual 3	Adaptive Dual	G.726
SNR เมื่อไม่มีความผิดพลาดจากการส่ง	5	4	7	6	1	2	3
SNR เมื่อมีความผิดพลาดจากการส่ง	4	3	7	6	1	2	5
เวลาที่ใช้เข้ารหัส	1	3	2	2	2	4	5

ตารางที่ 5.10 ลำดับของตัวประมาณตามค่าที่ใช้วัดเมื่อเข้ารหัส 2 บิตต่อแซมเปิล

ค่าที่ใช้วัด	Fixed LP2	Adaptive LP2	Fixed Dual 1	Fixed Dual 2	Fixed Dual 3	Adaptive Dual	G.726
SNR เมื่อไม่มีความผิดพลาดจากการส่ง	4	5	7	6	2	3	1
SNR เมื่อมีความผิดพลาดจากการส่ง	3	3	5	4	1	3	2
เวลาที่ใช้เข้ารหัส	1	3	2	2	2	4	5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6

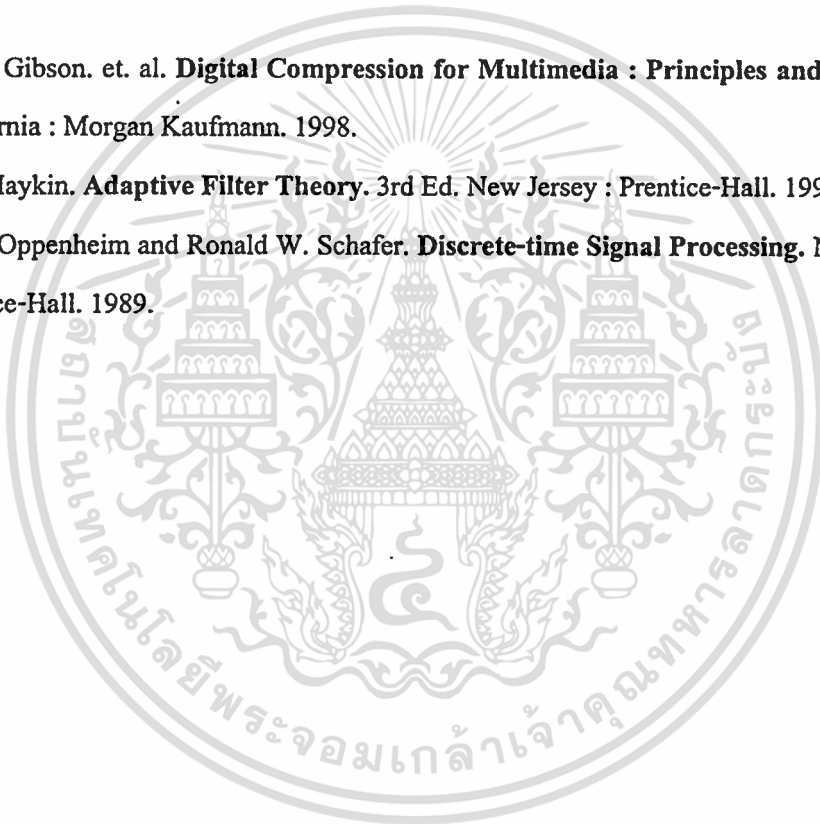
สรุปผลการวิจัยและข้อเสนอแนะ

วิทยานิพนธ์นี้ได้นำเสนอตัวประมาณออดิโกรีเลชันคู่โดยนำมาใช้ในการเข้ารหัสเสียงพูดด้วยวิธี ADPCM เริ่มจากตัวประมาณออดิโกรีเลชันคู่แบบธรรมดา(Fixed Dual 1) ซึ่งมีค่าสัมประสิทธิ์ $h_1 = h_2 = 0.5$ แล้วพัฒนาเป็นตัวประมาณออดิโกรีเลชันคู่แบบเหมาะสมในทางทฤษฎี(Fixed Dual 2) ซึ่งมีค่า $h_1 = h_2 = 0.5244$ จากนั้นทำเป็นตัวประมาณออดิโกรีเลชันคู่แบบอะแดปทีฟ(Adaptive Dual) ซึ่ง h_1 และ h_2 มีค่าเปลี่ยนแปลงตลอดเวลา แต่จากการสังเกตพบว่า h_1 มีค่าค่อนข้างคงที่ จึงปรับปรุงเป็นตัวประมาณออดิโกรีเลชันคู่แบบเหมาะสมจากการทดลอง(Fixed Dual 3) ซึ่งมีค่า $h_1 = 1$ และ $h_2 = 0.35$ ซึ่ง Adaptive Dual และ Fixed Dual 3 ได้ช่วยแก้ปัญหาเรื่องเสถียรภาพของ Fixed Dual 1 และ Fixed Dual 2 เมื่อมีความผิดพลาดจากการส่ง

จากการทดลองจะเห็นว่าตัวประมาณ Fixed Dual 3 มีค่า SNR ดีที่สุดเมื่อเข้ารหัส 4 บิต และ 3 บิตต่อแซมเปิลทั้งในกรณีที่มีและไม่มีผิดพลาดจากการส่งโดยใช้เวลาเข้ารหัสน้อยกว่าแบบมาตรฐาน แต่ผลตอบสนองทางความถี่จะไม่ดีเท่ากับตัวประมาณแบบมาตรฐาน ซึ่งอาจปรับปรุงโดยใช้อัลกอริธึมอะแดปทีฟแบบอื่น ใช้อันดับของสัมประสิทธิ์มากขึ้น หรือใช้ผลต่างถัดไป และย้อนหลังมาช่วยในการประมาณสัญญาณแต่จะทำให้ตัวประมาณมีความซับซ้อนมากขึ้น

เอกสารอ้างอิง

- [1] ITU-T. **Recommendation G.726 : 40, 32, 24, 16 kbit/s Adaptive Differential Pulse Code Modulation (ADPCM)**. 1990.
- [2] Bae M., Kim D., Jeon H. and Ann S. "On a New Predictor for the Waveform Coding of Speech Signal by using the Dual Autocorrelation and the Sigma-Delta Technique," **IEEE International Symposium on Circuits and Systems**, June 1994. pp. 261-264.
- [3] N. S. Jayant and Peter Noll. **Digital Coding of Waveforms**. New Jersey : Prentice-Hall. 1984.
- [4] Jerry D. Gibson. et. al. **Digital Compression for Multimedia : Principles and Standards**. California : Morgan Kaufmann. 1998.
- [5] Simon Haykin. **Adaptive Filter Theory**. 3rd Ed. New Jersey : Prentice-Hall. 1996.
- [6] Alan V. Oppenheim and Ronald W. Schaffer. **Discrete-time Signal Processing**. New Jersey : Prentice-Hall. 1989.



ภาคผนวก ก

โปรแกรมที่พัฒนาขึ้นในการวิจัย

```
/* ADPCM with Adaptive Dual ACF predictor and Adaptive Quantizer with one-word memory */
```

```
/* Input : wav file 8 bits/sample, sampling 8 kHz, mono channel */
```

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
#include <conio.h>
```

```
#include <process.h>
```

```
#include <alloc.h>
```

```
#include <math.h>
```

```
void Encode(void);
```

```
void Decode_at_receiver(void);
```

```
void AdaptiveQuantizer4bit(float);
```

```
void AdaptiveQuantizer3bit(float);
```

```
void AdaptiveQuantizer2bit(float);
```

```
void InverseAdaptiveQuantizer2bit(int);
```

```
void InverseAdaptiveQuantizer3bit(int);
```

```
void InverseAdaptiveQuantizer4bit(int);
```

```
void InverseAdaptiveQuantizer2bit_r(int);
```

```
void InverseAdaptiveQuantizer3bit_r(int);
```

```
void InverseAdaptiveQuantizer4bit_r(int);
```

```
void StepsizeAdjust4bit_r(void);
```

```
void StepsizeAdjust3bit_r(void);
```

```
void StepsizeAdjust2bit_r(void);
```

```
void Calculate_SNR_r(void);
```

```
void StepsizeAdjust4bit(int);
```

```
void StepsizeAdjust3bit(int);
```

```
void StepsizeAdjust2bit(int);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

void Write_output_header(void);

void Calculate_SNR(void);

int roundoff(double);

FILE *input_wav_file,*output_wav_file;
char input_wav_filename[40],output_wav_filename[40];
float progress_s,sp=0,d,Dn,M,Stepsize=1;
float SNR=0,Signal_variance=0,Noise_variance=0;
int number_of_bits,input_level,output_level,Hn,i,j;
float h1,h2,n=0,dq,progress_sr=0,sr=0,delay_sr=0;
int I,abs_I,abs_I_r;
float d_r,progress_sr_r=0,sr_r=0,dq_r,sp_r=0,delay_sr_r=0;
float Noise_variance_r=0,SNR_r=0,Stepsize_r=1;
float Kh1,Kh2,Kh1_r,Kh2_r,h1_r,h2_r;

void main(void)
{
    /* initial encoder coefficients */
    h1=10;
    h2=10;

    /* initial decoder coefficients */
    h1_r=10;
    h2_r=10;

    /* get input.wav filename */
    printf("\nEnter input wav filename : ");
    gets(input_wav_filename);

    /* get output.wav filename */
    printf("\nEnter output wav filename : ");
    gets(output_wav_filename);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

/* get number of bits to encode */
printf("\nEnter number of bits to encode : ");
scanf("%d",&number_of_bits);

/* open input wav file */
input_wav_file=fopen(input_wav_filename,"rb");

/* open output wav file */
output_wav_file=fopen(output_wav_filename,"wb");

Write_output_header(); /* write header to output wav file */
fseek(input_wav_file,58L,SEEK_SET); /* point to the first input data */
fseek(output_wav_file,58L,SEEK_SET); /* point to the first output data */
input_level=fgetc(input_wav_file); /* get input data */

while ( !feof(input_wav_file) )
{
    Encode(); /* encode and decode data at encoder */
    Decode_at_receiver(); /* decode data at decoder */
    input_level=fgetc(input_wav_file);
}

fclose(input_wav_file);
fclose(output_wav_file);

Calculate_SNR(); /* calculate SNR at encoder */
Calculate_SNR_r(); /* calculate SNR at receiver */
}

/* function encode and decode data at encoder */
void Encode(void)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

{
    progress_s=input_level-128; /* get input data */
    Signal_variance += progress_s*progress_s; /* calculate signal variance */
    sp = h1*progress_s+h2*delay_sr; /* calculate predicted signal */
    d = sr - sp; /* calculate difference signal */

    /* quantize and dequantize */
    /* quantize : convert difference signal to codeword */
    /* dequantize : convert codeword to quantized difference signal */
    switch (number_of_bits)
    {
        case 2 : AdaptiveQuantizer2bit(d);
                InverseAdaptiveQuantizer2bit(I);
                break;
        case 3 : AdaptiveQuantizer3bit(d);
                InverseAdaptiveQuantizer3bit(I);
                break;
        case 4 : AdaptiveQuantizer4bit(d);
                InverseAdaptiveQuantizer4bit(I);
                break;
    }

    progress_sr = (-dq+sr-h2*delay_sr)/h1; /* calculate reconstructed signal */

    /* limit signal in range -128 to 127 */
    if (progress_sr>127)
        progress_sr=127;
    if (progress_sr<-128)
        progress_sr=-128;

    /* calculate coefficients by LMS adaptive algorithm */
    Kh1=0.001*(progress_sr)/(100+progress_sr*progress_sr);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Kh2=0.001*(delay_sr)/(100+delay_sr*delay_sr);
```

```
h1=(127.0/128.0)*h1+Kh1*(dq);
```

```
h2=(127.0/128.0)*h2+Kh2*(dq);
```

```
/* limit h1 */
```

```
if (h1<=2.0)
```

```
    h1=2.0;
```

```
/* limit h1,h2 in stability region */
```

```
if (h2<=1-h1)
```

```
    h2=1-h1;
```

```
if (h2>=h1)
```

```
    h2=h1;
```

```
/* delay reconstructed signal */
```

```
delay_sr = sr;
```

```
sr = progress_sr;
```

```
output_level=roundoff(sr)+128; /* calculate output signal */
```

```
fputc(char(output_level),output_wav_file); /* write output signal */
```

```
/* calculate noise variance */
```

```
Noise_variance += (progress_s-progress_sr)*(progress_s-progress_sr);
```

```
}
```

```
/* function adaptive quantizer 4 bits/sample */
```

```
void AdaptiveQuantizer4bit(float d)
```

```
{
```

```
    Dn = (d<0) ? -d:d;
```

```

if ((Dn>0)&&(Dn<=Stepsize))
    I=0;

if ((Dn>Stepsize)&&(Dn<=2*Stepsize))
    I=1;

if ((Dn>2*Stepsize)&&(Dn<=3*Stepsize))
    I=2;

if ((Dn>3*Stepsize)&&(Dn<=4*Stepsize))
    I=3;

if ((Dn>4*Stepsize)&&(Dn<=5*Stepsize))
    I=4;

if ((Dn>5*Stepsize)&&(Dn<=6*Stepsize))
    I=5;

if((Dn>6*Stepsize)&&(Dn<=7*Stepsize))
    I=6;

if (Dn>7*Stepsize)
    I=7;

if (d<0)
    I=I+8;
}

```

```

/* function adaptive quantizer 3 bits/sample */

```

```

void AdaptiveQuantizer3bit(float d)

```

```

{

```

```

    Dn = (d<0) ? -d:d;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if ((Dn>0)&&(Dn<=Stepsize))
    I=0;

if ((Dn>Stepsize)&&(Dn<=2*Stepsize))
    I=1;

if ((Dn>2*Stepsize)&&(Dn<=3*Stepsize))
    I=2;

if (Dn>3*Stepsize)
    I=3;

if (d<0)
    I=I+4;
}

/* function adaptive quantizer 2 bits/sample */
void AdaptiveQuantizer2bit(float d)
{
    Dn = (d<0) ? -d:d;

    if ((Dn>0)&&(Dn<=Stepsize))
        I=0;

    if (Dn>Stepsize)
        I=1;

    if (d<0)
        I=I+2;
}

/* function inverse adaptive quantizer 2 bits/sample */

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void InverseAdaptiveQuantizer2bit(int I)
```

```
{
  abs_I=1&I;
  switch (abs_I)
  {
    case 0 : dq=Stepsize/2;break;
    case 1 : dq=3*Stepsize/2;break;
  }
  if (I>1)
    dq=-dq;
```

```
  StepsizeAdjust2bit(abs_I);
```

```
}
```

```
/* function inverse adaptive quantizer 3 bits/sample */
```

```
void InverseAdaptiveQuantizer3bit(int I)
```

```
{
  abs_I=3&I;
  switch (abs_I)
  {
    case 0 : dq=Stepsize/2;break;
    case 1 : dq=3*Stepsize/2;break;
    case 2 : dq=5*Stepsize/2;break;
    case 3 : dq=7*Stepsize/2;break;
  }
```

```
  if (I>3)
```

```
    dq=-dq;
```

```
  StepsizeAdjust3bit(abs_I);
```

```
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
/* function inverse adaptive quantizer 4 bits/sample */
```

```
void InverseAdaptiveQuantizer4bit(int I)
```

```
{
  abs_I=7&I;
  switch (abs_I)
  {
    case 0 : dq=Stepsize/2;break;
    case 1 : dq=3*Stepsize/2;break;
    case 2 : dq=5*Stepsize/2;break;
    case 3 : dq=7*Stepsize/2;break;
    case 4 : dq=9*Stepsize/2;break;
    case 5 : dq=11*Stepsize/2;break;
    case 6 : dq=13*Stepsize/2;break;
    case 7 : dq=15*Stepsize/2;break;
  }
  if (I>7)
    dq=-dq;
  StepsizeAdjust4bit(abs_I);
}
```

```
/* function adjust step size 4 bits/sample */
```

```
void StepsizeAdjust4bit(int abs_I)
```

```
{
  switch (abs_I)
  {
    case 0 : M=0.9;break;
    case 1 : M=0.9;break;
    case 2 : M=0.9;break;
    case 3 : M=0.9;break;
    case 4 : M=1.2;break;
  }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

case 5 : M=1.6;break;
case 6 : M=2.0;break;
case 7 : M=2.4;break;
}
Stepsize=M*pow(Stepsize,63.0/64.0);
if(Stepsize<1)
    Stepsize=1;
}

/* functin adjust step size 3 bits/sample */
void StepsizeAdjust3bit(int abs_I)
{
switch (abs_I)
{
case 0 : M=0.9;break;
case 1 : M=0.9;break;
case 2 : M=1.25;break;
case 3 : M=1.75;break;
}
Stepsize=M*pow(Stepsize,63.0/64.0);
if(Stepsize<1)
    Stepsize=1;
}

.

/* function adjust step size 2 bits/sample */
void StepsizeAdjust2bit(int abs_I)
{
switch (abs_I)
{
case 0 : M=0.8;break;
case 1 : M=1.6;break;
}
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Stepsize=M*pow(Stepsize,63.0/64.0);
if (Stepsize<1)
    Stepsize=1;
}

/* Write header to output wav file */
void Write_output_header(void)
{
    fseek(input_wav_file,0L,SEEK_SET);
    fseek(output_wav_file,0L,SEEK_SET);
    for (int i=0;i<58;i++)
    {
        char input=fgetc(input_wav_file);
        fputc(input,output_wav_file);
    }
}

/* function calculate SNR at encoder */
void Calculate_SNR(void)
{
    SNR = 10*log10(Signal_variance/Noise_variance);
    printf("SNR = %2.2f dB \n",SNR);
}

/* Function roundoff number */
int roundoff(double number)
{
    double integer;
    double fraction=modf(number,&integer);
    if (fraction<0.5)
        return(integer);
    else

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

return(integer+1);
}

/* function decode data at decoder */
void Decode_at_receiver(void)
{
switch (number_of_bits)
{
case 2 : InverseAdaptiveQuantizer2bit_r(I);break;
case 3 : InverseAdaptiveQuantizer3bit_r(I);break;
case 4 : InverseAdaptiveQuantizer4bit_r(I);break;
}

progress_sr_r = (-dq_r+sr_r-h2_r*delay_sr_r)/h1_r;

if (progress_sr_r>127)
progress_sr_r=127;
if (progress_sr_r<-128)
progress_sr_r=-128;

d_r=progress_s-progress_sr_r;

Kh1_r=0.001*(progress_sr_r)/(100+progress_sr_r*progress_sr_r);
Kh2_r=0.001*(delay_sr_r)/(100+delay_sr_r*delay_sr_r);
h1_r=(127.0/128.0)*h1_r+Kh1_r*(dq_r);
h2_r=(127.0/128.0)*h2_r+Kh2_r*(dq_r);

if (h1_r<=2.0)
h1_r=2.0;

if (h2_r<=1-h1_r)
h2_r=1-h1_r;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if (h2_r>=h1_r)
    h2_r=h1_r;

delay_sr_r = sr_r;
sr_r = progress_sr_r;

Noise_variance_r += (progress_s-progress_sr_r)*(progress_s-progress_sr_r);
}

```

```

/* function inverse adaptive quantizer 2 bits/sample at decoder */

```

```

void InverseAdaptiveQuantizer2bit_r(int I)

```

```

{
    abs_I_r=1&I;
    switch (abs_I_r)
    {
        case 0 : dq_r=Stepsize_r/2;break;
        case 1 : dq_r=3*Stepsize_r/2;break;
    }
    if (I>1)
        dq_r=-dq_r;

```

```

    StepsizeAdjust2bit_r();
}

```

```

/* function inverse adaptive quantizer 3 bits/sample at decoder */

```

```

void InverseAdaptiveQuantizer3bit_r(int I)

```

```

{
    abs_I_r=3&I;
    switch (abs_I_r)
    {
        case 0 : dq_r=Stepsize_r/2;break;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

case 1 : dq_r=3*Stepsize_r/2;break;
case 2 : dq_r=5*Stepsize_r/2;break;
case 3 : dq_r=7*Stepsize_r/2;break;
}

```

```

if (I>3)
    dq_r=-dq_r;

```

```

StepsizeAdjust3bit_r();
}

```

```

/* function inverse adaptive quantizer 4 bits/sample */

```

```

void InverseAdaptiveQuantizer4bit_r(int I)

```

```

{
    abs_I_r=7&I;
    switch (abs_I_r)
    {
        case 0 : dq_r=Stepsize_r/2;break;
        case 1 : dq_r=3*Stepsize_r/2;break;
        case 2 : dq_r=5*Stepsize_r/2;break;
        case 3 : dq_r=7*Stepsize_r/2;break;
        case 4 : dq_r=9*Stepsize_r/2;break;
        case 5 : dq_r=11*Stepsize_r/2;break;
        case 6 : dq_r=13*Stepsize_r/2;break;
        case 7 : dq_r=15*Stepsize_r/2;break;
    }
}

```

```

if (I>7)
    dq_r=-dq_r;

```

```

StepsizeAdjust4bit_r();
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
/* function adjust step size 4 bits/sample at decoder */
```

```
void StepsizeAdjust4bit_r(void)
```

```
{
  switch (abs_I_r)
  {
    case 0 : M=0.9;break;
    case 1 : M=0.9;break;
    case 2 : M=0.9;break;
    case 3 : M=0.9;break;
    case 4 : M=1.2;break;
    case 5 : M=1.6;break;
    case 6 : M=2.0;break;
    case 7 : M=2.4;break;
  }
  Stepsize_r=M*pow(Stepsize_r,63.0/64.0);
  if (Stepsize_r<1)
    Stepsize_r=1;
}
```

```
/* function adjust step size 3 bits/sample at decoder */
```

```
void StepsizeAdjust3bit_r(void)
```

```
{
  switch (abs_I_r)
  {
    case 0 : M=0.9;break;
    case 1 : M=0.9;break;
    case 2 : M=1.25;break;
    case 3 : M=1.75;break;
  }
  Stepsize_r=M*pow(Stepsize_r,63.0/64.0);
  if (Stepsize_r<1)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    Stepsize_r=1;
}

/* function adjust step size 2 bits/sample at decoder */
void StepsizeAdjust2bit_r(void)
{
    switch (abs_I_r)
    {
        case 0 : M=0.8;break;
        case 1 : M=1.6;break;
    }
    Stepsize_r=M*pow(Stepsize_r,63.0/64.0);
    if (Stepsize_r<1)
        Stepsize_r=1;
}

/* function calculate SNR at decoder */
void Calculate_SNR_r(void)
{
    SNR_r = 10*log10(Signal_variance/Noise_variance_r);
    printf("SNR_r = %.2f dB\n",SNR_r);
}

```

ภาคผนวก ข

ผลงานวิจัยที่ได้รับการตีพิมพ์

W. Wongsachua and K. Songwatana. "A Comparison of the Dual Autocorrelation Technique for ADPCM Speech Coding to that of Standard ADPCM Speech Coding Methods." **The Third International Symposium on Wireless Personal Multimedia Communications**. Bangkok, Thailand, 12-15 Nov. 2000. pp. 213-217.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

A Comparison of the Dual Autocorrelation Technique for ADPCM Speech Coding to that of Standard ADPCM Speech Coding Methods

W. Wongsachua and K. Songwatana

Department of Telecommunication Engineering, King Mongkut's Institute of Technology Ladkrabang
Chalongkrung Road, Ladkrabang, Bangkok 10520, Thailand. E-mail: kskraisi@kmitl.ac.th

Abstract - This paper presents speech coding by Adaptive Differential Pulse Code Modulation (ADPCM) using dual autocorrelation technique. Generally, the value of autocorrelation function of adjacent speech samples is greater than other samples. To predict the present sample by using one past sample and one next sample is more efficient than only using past samples. This prediction method is used in ADPCM voice coding and the result compares with the standard ones.

Keywords: ADPCM, Speech coding, Dual autocorrelation.

I. INTRODUCTION

To transmit speech signals over digital communication network, analog to digital conversion is required. The conventional conversion method is Pulse Code Modulation (PCM) [1]. Speech signals are sampled, quantized, and encoded with 8 bits per sample. Because the successive speech samples are highly correlated, the difference of the adjacent speech samples has smaller variance as compare to the amplitude of one sample. This property is used in Differential PCM (DPCM) to reduce number of bits per sample. The main parts of DPCM are a quantizer and a predictor. If the quantizer is adaptive or both the quantizer and the predictor are adaptive, it is referred to Adaptive DPCM (ADPCM) [2]. ADPCM is adopted by ITU as the standard of digital transmission systems [3]. This paper presents ADPCM using dual autocorrelation technique to predict the present speech sample by using the one past and the one next sample [4]. Adaptive algorithm has been used to enhance the performance.

II. DPCM

In the DPCM encoder (Figure 1), a difference signal $d(n)$ is obtained by subtracting a predicted signal $s_p(n)$ from an input signal $s(n)$.

$$d(n) = s(n) - s_p(n) \quad (1)$$

The difference signal is quantized, $d_q(n)$, and mapped to codeword for transmitting to the decoder. The predicted signal is added to the quantized difference signal to produce a reconstructed signal $s_r(n)$.

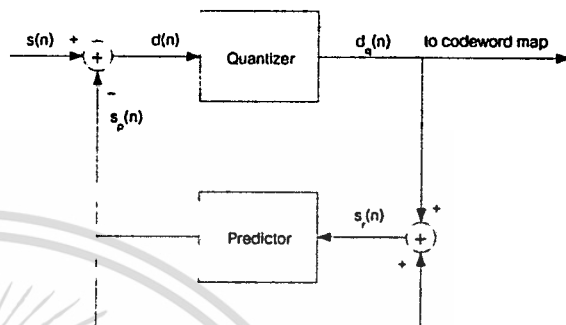


Figure 1. DPCM encoder.

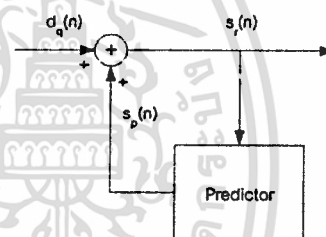


Figure 2. DPCM decoder.

$$s_r(n) = d_q(n) + s_p(n) \quad (2)$$

The reconstructed signal is used to produce the predicted signal. At the receiver, the DPCM decoder is the feedback part of the encoder as shown in Figure 2.

II.1 Quantizer

In DPCM, the quantizer is used to transform the difference signal into the quantized difference signal as shown in Figure 3. The quantized difference signal is

$$d_q(n) = H(n)\Delta(n)/2 \quad (3)$$

$$\pm H(n) = 1, 3, 5, \dots, 2^R - 1 ; R \geq 2$$

where $\Delta(n)$ is a step size of the n^{th} sample and R is a number of bits. In Figure 3, $R = 4$, the possible values of $|H(n)|$ are 1, 3, 5, 7, 9, 11, 13 and 15. In DPCM, $\Delta(n)$ is constant over n .

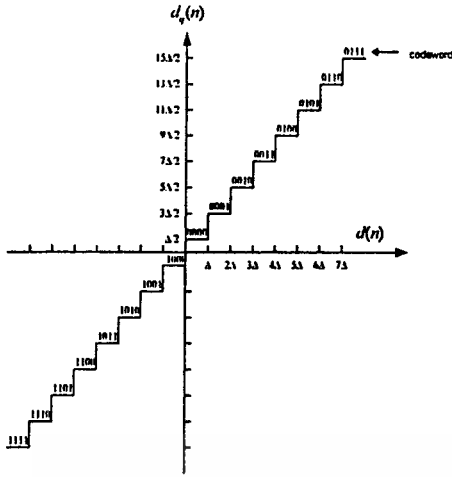


Figure 3. 4-bit quantizer.

II.2 Linear Predictor

A linear predictor is used to predict the input signal. Figure 4 shows the structure of a linear predictor of N^{th} order in the form

$$s_p(n) = \sum_{j=1}^N h_j s(n-j) \quad (4)$$

which is a weighted sum of N previous samples. The weighted h_j are linear predictor coefficients. The performance of DPCM is evaluated by the prediction gain G_p .

$$G_p = \frac{\sigma_s^2}{\sigma_d^2} \quad (5)$$

In speech prediction, the value of G_p typically saturates with $N = 2$ [2]. In Figure 1, if $N = 2$ and the quantization noise is small, the predicted signal is

$$s_p(n) \approx h_1 s(n-1) + h_2 s(n-2) \quad (6)$$

and the mean square error is

$$\sigma_d^2 = E[D^2(n)] = E\{[S(n) - S_p(n)]^2\} \quad (7)$$

The values of h_1 and h_2 which minimize the mean square error can be obtained by setting

$$\frac{\partial \sigma_d^2}{\partial h_j} = 0; j = 1, 2 \quad (8)$$

If a variance-normalized autocorrelation function is defined in the form

$$\rho_k = \frac{R_{ss}(k)}{R_{ss}(0)} \quad (9)$$

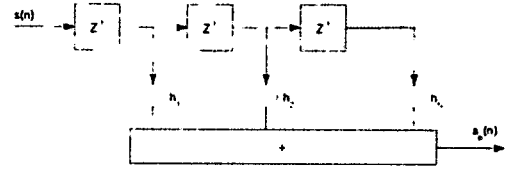
Figure 4. Linear predictor of order N .

Table 1
Step size multipliers of 4-bit adaptive quantizer with a one-word memory for speech signal

M_1	M_2	M_3	M_4	M_5	M_6	M_7	M_8
0.9	0.9	0.9	0.9	1.2	1.6	2.0	2.4

We get

$$h_{1(\text{optimum})} = \frac{\rho_1(1-\rho_2)}{1-\rho_1^2}; h_{2(\text{optimum})} = \frac{\rho_2-\rho_1^2}{1-\rho_1^2} \quad (10)$$

III. ADPCM

ADPCM is DPCM with adaptive quantizer only, or with both adaptive quantizer and adaptive predictor.

III.1 Adaptive quantizer with a one-word memory [2]

A step size of the adaptive quantizer is adjusted proportional to the difference signal. The next step size $\Delta(n)$ is the product of $\Delta(n-1)$ and a multiplier M that is a function of a magnitude index $|H(n-1)|$.

$$\Delta(n) = M(|H(n-1)|)\Delta(n-1) \quad (11)$$

In Figure 3, the eight values of $|H(n)|$ lead to the respective multipliers M_1 to M_8 as shown in Table 1. The step size increases are faster than step size decreases. This is to match the patterns of speech waveform.

III.2 Adaptive linear predictor

The linear predictor coefficients in subsection II.2 are obtained from long-timed-averaged statistics. But the speech signals are nonstationary and statistics are time-varying. So, the suitable predictor should be adaptive. The adaptive algorithm is

$$\mathbf{h}(n+1) = \mathbf{h}(n) + \mu(n)s_r(n)d_q(n) \quad (12)$$

which $\mathbf{h}(n) = [h_1(n) \ h_2(n) \ \dots \ h_N(n)]^T$, μ is a step size and $s_r(n) = [s_r(n-1) \ s_r(n-2) \ \dots \ s_r(n-N)]^T$.

III.3 ITU G.726 ADPCM

The ITU G.726 ADPCM standard defines the following adaptive quantizer and adaptive predictor.

Table 2
Step size multiplier of 4-bit ITU G.726 adaptive quantizer

M_1	M_2	M_3	M_4	M_5	M_6	M_7	M_8
0.984	1.006	1.037	1.070	1.142	1.283	1.585	4.482

III.3.1 G.726 Adaptive Quantizer

The G.726 adaptive quantizer is a dynamic locking quantizer which stops adapting in the presence of a stationary input. The unlocked step size $\Delta_u(n)$ is

$$\Delta_u(n) = M \left(|H(n-1)| \right) \Delta^\beta(n-1); \beta = \frac{31}{32} \quad (13)$$

where β is a leakage factor. The eight possible values of multiplier M are shown in Table 2.

The logarithm of the locked step size $\Delta_l(n)$ is

$$\log \Delta_l(n) = \frac{63}{64} \log \Delta_l(n-1) + \frac{1}{64} \log \Delta_u(n) \quad (14)$$

And the logarithm of the final step size is

$$\log \Delta(n) = w \log \Delta_u(n) + (1-w) \log \Delta_l(n); 0 \leq w \leq 1 \quad (15)$$

where w is a weighting parameter.

III.3.2 G.726 Adaptive Predictor

The G.726 adaptive predictor uses two poles and six zeros. The predicted signal is

$$s_p(n) = \sum_{i=1}^2 a_i(n-1) s_r(n-i) + \sum_{i=1}^6 b_i(n-1) d_q(n-i) \quad (16)$$

The reconstructed signal is

$$s_r(n) = d_q(n) + s_p(n) \quad (17)$$

Both sets of predictor coefficients are updated using a simplified gradient algorithm.

$$a_1(n) = \frac{255}{256} a_1(n-1) + \frac{3}{256} \text{sgn}[p(n)] \text{sgn}[p(n-1)] \quad (18)$$

$$a_2(n) = \frac{127}{128} a_2(n-1) + \frac{1}{128} \left\{ \text{sgn}[p(n)] \text{sgn}[p(n-2)] - f[a_1(n-1)] \text{sgn}[p(n)] \text{sgn}[p(n-1)] \right\} \quad (19)$$

$$p(n) = d_q(n) + \sum_{i=1}^6 b_i(n-1) d_q(n-i) \quad (20)$$

$$f[a_1] = \begin{cases} 4a_1, & |a_1| \leq 0.5 \\ 2 \text{sgn}[a_1], & |a_1| > 0.5 \end{cases} \quad (21)$$

$$|a_2| \leq 0.75, |a_1| \leq \frac{15}{16} - a_2 \quad (22)$$

Table 3
Comparison between ACF and Dual ACF

k	female		male	
	ACF	Dual ACF	ACF	Dual ACF
1	0.921	0.969	0.884	0.945
2	0.843	0.906	0.758	0.816
3	0.741	0.840	0.582	0.700
4	0.631	0.749	0.354	0.609

$$b_i(n) = \frac{255}{256} b_i(n-1) + \frac{1}{128} \text{sgn}[d_q(n)] \text{sgn}[d_q(n-i)] \quad (23)$$

IV. PREDICTION BY USING DUAL AUTOCORRELATION TECHNIQUE

IV.1 Fixed Dual ACF Predictor

The value of autocorrelation function (ACF) of adjacent speech samples is greater than other samples. Table 3 [5] shows that the value of dual autocorrelation function (Dual ACF) follows (25) is greater than the value of ACF follows (24). It indicates that to predict the present sample by using one past sample and one next sample is more effective than only using past samples.

$$ACF(k) = \frac{1}{N} \sum_{n=0}^{N-1-k} s(n) s(n+k) \quad (24)$$

$$DualACF(k) = \frac{1}{N} \sum_{n=k}^{N-1-k} s(n) \left\{ \frac{s(n-k) s(n+k)}{2} \right\} \quad (25)$$

The difference signal is obtained by subtracting the predicted signal from the present reconstructed signal $s_r(n)$

$$d(n) = s_r(n) - s_p(n) \quad (26)$$

The predicted signal is an average value of the next input signal $s(n+1)$ and the past reconstructed signal $s_r(n-1)$

$$s_p(n) = \frac{1}{2} \{ s(n+1) + s_r(n-1) \} \quad (27)$$

The next reconstructed signal $s_r(n+1)$ is

$$s_r(n+1) = -2d_q(n) + 2s_r(n) - s_r(n-1) \quad (28)$$

The encoder and decoder of ADPCM using Dual ACF are shown in Figure 5 and 6 respectively.

IV.2 Adaptive Dual ACF Predictor

Equation (27) can be written in general form as

$$s_p(n) = h_1 s(n+1) + h_2 s_r(n-1) \quad (29)$$

where h_1 and h_2 are predictor coefficients. Equation (26) becomes

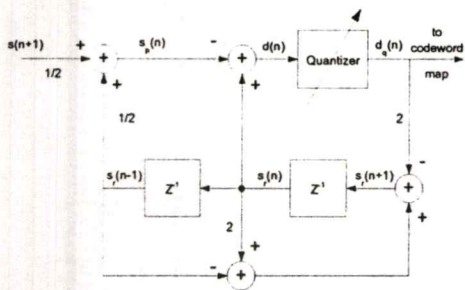


Figure 5. Dual ACF encoder.

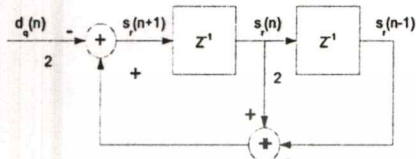


Figure 6. Dual ACF decoder.

$$d(n) = s_r(n) - h_1 s(n+1) - h_2 s_r(n-1) \quad (30)$$

that h_1 and h_2 are adapted to minimize the mean square error by the gradient algorithm

$$h(n+1) = h(n) - \frac{1}{2} \mu(n) \nabla_h [E[d^2(n)]] \quad (31)$$

where $h(n) = \begin{bmatrix} h_1(n) \\ h_2(n) \end{bmatrix}$ and the step size $\mu(n) = \begin{bmatrix} \mu_1(n) \\ \mu_2(n) \end{bmatrix}$.

The simple solution to the gradient of (31) is the least mean square (LMS) algorithm [6]. This method is based on using $d^2(n)$ instead of $E[d^2(n)]$. But at the decoder $d(n)$ is not available. So, $d_q(n)$ is used in gradient calculation.

$$d_q(n) = s_r(n) - h_1 s_r(n+1) - h_2 s_r(n-1) \quad (32)$$

And the gradient is

$$\begin{aligned} \nabla_h [d^2(n)] &\approx \nabla_h [d_q^2(n)] = 2d_q(n) \nabla_h d_q(n) \\ &= -2d_q(n) \{s_r(n+1) + s_r(n-1)\} \end{aligned} \quad (33)$$

Equation (31) becomes

$$h_1(n+1) = \alpha h_1(n) + \mu_1(n) d_q(n) s_r(n+1) \quad (34a)$$

$$h_2(n+1) = \alpha h_2(n) + \mu_2(n) d_q(n) s_r(n-1) \quad (34b)$$

$$\mu_1(n) = \frac{g}{100 + s_r^2(n+1)} \quad (35a)$$

$$\mu_2(n) = \frac{g}{100 + s_r^2(n-1)} \quad (35b)$$

Table 4
SNR for encoding 4 bits/sample (dB)

file	method			
	LP2	Fixed Dual	Adaptive Dual	G.726
male1	24.71	19.17	25.40	24.70
male2	18.28	16.10	24.96	22.08
male3	19.91	17.53	26.05	23.99
female1	22.31	16.96	23.08	25.46
female2	24.58	19.96	25.26	27.38
average	21.96	17.94	25.09	24.72

Table 5
SNR for encoding 3 bits/sample (dB)

file	method			
	LP2	Fixed Dual	Adaptive Dual	G.726
male1	21.45	14.53	21.27	20.35
male2	15.88	13.17	20.08	19.55
male3	17.48	14.11	21.00	20.89
female1	18.46	13.95	18.72	19.90
female2	20.57	15.05	19.96	21.64
average	18.77	14.16	20.21	20.47

Table 6
SNR for encoding 2 bits/sample (dB)

file	method			
	LP2	Fixed Dual	Adaptive Dual	G.726
male1	16.74	10.11	15.85	14.96
male2	12.44	9.54	14.83	15.14
male3	14.07	10.42	15.24	14.93
female1	12.83	8.95	12.55	13.13
female2	14.54	10.77	13.46	14.72
average	14.12	9.96	14.39	14.78

where α is the damping parameter and g is the adaptation gain [7]. For stability, h_1 and h_2 are adjusted in the boundary

$$h_1 > 0.5, 1 - h_1 < h_2 < h_1 \quad (36)$$

V. EXPERIMENTAL RESULTS

Four ADPCM algorithms are simulated on Pentium-II 300-MHz personal computer. They are as follows.

- 1) Adaptive quantizer and second-order linear predictor (LP2)
- 2) Adaptive quantizer and fixed Dual ACF predictor (Fixed Dual)
- 3) Adaptive quantizer and adaptive Dual ACF predictor (Adaptive Dual)
- 4) ITU G.726 adaptive quantizer and adaptive predictor (G.726)

Algorithm 1) to 3) use the adaptive quantizer with a one-word memory. The speech data are 60 second .WAV files from 3 male and 2 female speakers. They are sampled at 8 kHz and saved in 8 bits/sample PCM format. This files are encoded to 4, 3 and 2 bits/sample. Signal to

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Table 7
Computation time for encoding 4, 3 and 2 bits/sample (seconds)

bits/ sample	method			
	LP2	Fixed Dual	Adaptive Dual	G.726
4	0.60	0.71	1.10	6.68
3	0.49	0.64	0.88	6.13
2	0.44	0.55	0.78	6.76

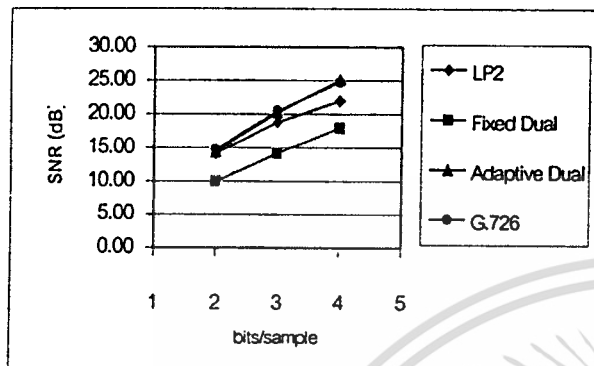


Figure 7. SNR for encoding 4, 3 and 2 bits/sample.

noise ratio (SNR) and computation time are measured, the results are shown in Table 4 to 7 and Figure 7.

For 4-bit encoding, the SNR of the Adaptive Dual and G.726 are nearly equivalent and higher than any others. The result is the same for 3-bit encoder. LP2, Adaptive Dual and G.726 are nearly equivalent in SNR for 2-bit encoding. Fixed Dual has the lowest SNR in all cases, because it has a good prediction in voiced segment in which ACF is high but the SNR degraded in unvoiced segment when ACF is low.

Predictor coefficients of LP2 and Fixed Dual are constant, so their computation time are low. LP2 has lower computation time than Fixed Dual because $s_r(n)$ of LP2 is easily calculated from $d_q(n)$ and $s_p(n)$ as shown in (2), while $s_r(n+1)$ of Fixed Dual is calculated from $d_q(n)$, $s_r(n)$ and $s_r(n-1)$ as shown in (28). When the predictor coefficients are adapted in Adaptive Dual, the computation time is higher than LP2 and Fixed Dual. G.726 has the most complicated algorithm, so its computation time is the highest. G.726 2-bit encoding has the higher computation time than G.726 4-bit and 3-bit encoding because 2-bit encoding in G.726 requires calculation on $d_q(n)$ when the codeword is zero while in 4-bit and 3-bit case, $d_q(n)$ can be set to zero.

VI. CONCLUSION

We have experimented with different methods of Adaptive Differential Pulse Code Modulation, namely

- 1) LP2
- 2) Fixed Dual ACF
- 3) Adaptive Dual ACF
- 4) G.726

The results show that the proposed Fixed Dual ACF method has low computation time, but has the worst SNR. Adaptive Dual ACF method has the nearly equivalent performance to the standard with lower computation time.

VII. REFERENCES

- [1] K. Shenoi, *Digital Signal Processing in Telecommunications*, Prentice Hall, New Jersey, 1995.
- [2] N. S. Jayant and P. Noll, *Digital Coding of Waveforms*, Prentice Hall, New Jersey, 1984.
- [3] ITU-T Recommendation G.726, "40, 32, 24, 16 kbit/s Adaptive Differential Pulse Code Modulation (ADPCM)," Geneva, 1990.
- [4] M. Bae, M. Lee, K. Min and S. Ann, "On the Predictor for the Waveform Coding of Speech Signals by using the Dual First Order Difference Values," in *IEEE Region 10 Conference of Tenco 92*, Melbourne, Australia, Nov. 1992, pp. 46-50.
- [5] M. Bae, D. Kim, H. Jeon and S. Ann, "On a New Predictor for the Waveform Coding of Speech Signal by using the Dual Auto-correlation and the Sigma-Delta Technique," in *IEEE International Symposium on Circuits and Systems*, June 1994, pp. 261-264.
- [6] O. Macchi, *Adaptive Processing: the least mean square approach*, John Wiley & Sons, Chichester, 1995.
- [7] B. Koo and J. D. Gibson, "Experimental Comparison of All-Pole, All-Zero, and Pole-Zero Predictors for ADPCM Speech Coding," *IEEE Trans. on Comm.*, vol. 34, pp. 285-290, March 1986.

ประวัติผู้เขียน

นายวรการ วงศ์สายเชื้อ เกิดเมื่อวันที่ 30 เมษายน พ.ศ. 2518 ที่จังหวัดร้อยเอ็ด สำเร็จการศึกษาระดับปริญญาตรี วิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมโทรคมนาคม จากสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังเมื่อปี พ.ศ. 2540 จากนั้นเข้าทำงานที่บริษัทสามารถเทลคอม ในตำแหน่งวิศวกรโครงข่ายเป็นเวลา 1 ปี



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้