การออกแบบเมทริกซ์พาริตีเช็กสำหรับรหัสแอลดีพีซีแบบไม่คงที่ที่มีขนาดบล็อกสั้น

DESIGN OF PARITY-CHECK MATRIX FOR SHORT BLOCK IRREGULAR LDPC CODES

ชุติมา ประสาทแก้ว
CHUTIMA PRASARTKAEW

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาปรัชญาดุษฎีบัณฑิต
สาขาวิชาเทคโนโลยีการบันทึกข้อมูล
วิทยาลัยนวัตกรรมการจัดการข้อมูล
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
พ.ศ. 2556

KMITL-2013-DS-D-001-02

DESIGN OF PARITY-CHECK MATRIX FOR SHORT BLOCK IRREGULAR LDPC CODES

CHUTIMA PRASARTKAEW

A THESIS SUBMITTED IN PARTIAL FULFILLMENT

OF THE REQUIREMENT FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY IN DATA STORAGE TECHNOLOGY

COLLEGE OF DATA STORAGE INNOVATION

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

2013

KMITL-2013-DS-D-001-02

วิทยาลัยนวัตกรรมการจัดการข้อมูล
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ใบรับรองวิทยานิพนธ์

หัวข้อวิทยานิพนธ์ การออกแบบเมทริกซ์พาริตีเช็กสำหรับรหัสแอลดีพีซีแบบไม่คงที่ที่มีขนาดบล็อกสั้น

Thesis Title DESIGN OF PARITY-CHECK MATRIX FOR SHORT BLOCK IRREGULAR LDPC CODES

นักศึกษา นางชุติมา ประสาทแก้ว

รหัสประจำตัว 51064903

ปริญญา ปรัชญาดุษฎีบัณฑิต

สาขาวิชา เทคโนโลยีการบันทึกข้อมูล

อาจารย์ที่ปรึกษาวิทยานิพนธ์ รองศาสตราจารย์ ดร.สมศักดิ์ ชุมช่วย

หมายเลขวิทยานิพนธ์ KMITL-2013-DS-D-001-02

คณะกรรมการสอบวิทยานิพนธ์		ลายมือชื่อ
ดร.กสิน	วิเชียรชม	
รองศาสตราจารย์ ดร.สมศักดิ์	ชุมช่วย	
ดร.เลิศศักดิ์	เลขวัต	
ดร.ชานนท์	วริสาร	
รองศาสตราจารย์ ดร.ปิยะ	โควินท์ทวีวัฒน์	

วัน/เดือน/ปี ที่สอบ 18 มีนาคม 2556 เวลา 08.30 - 10.30 น.

สถานที่สอบ อาคารเฉลิมพระเกียรติ 55 พรรษา สมเด็จพระเทพรัตนราชสุดาฯ สยามบรมราชกุมารี

วิทยาลัยนวัตกรรมการจัดการข้อมูล รับรองแล้ว

(รองศาสตราจารย์ ดร.อภินันท์ ธนชยานนท์)
คณบดี วิทยาลัยนวัตกรรมการจัดการข้อมูล

วันที่ 11 เมษายน พ.ศ. 2556

หัวข้อวิทยานิพนธ์	การออกแบบเมทริกซ์พาริตีเช็กสำหรับรหัสแอลดีพีซีแบบไม่คงที่ที่มีขนาดบล็อกสั้น
นักศึกษา	นางชุติมา ประสาทแก้ว
รหัสนักศึกษา	51064903
ปริญญา	ปรัชญาดุษฎีบัณฑิต
สาขาวิชา	เทคโนโลยีการบันทึกข้อมูล
พ.ศ.	2556
อาจารย์ผู้ควบคุมวิทยานิพนธ์	รองศาสตราจารย์ ดร.สมศักดิ์ ชุมช่วย

บทคัดย่อ

โครงสร้างของเมทริกซ์พาริตีเช็กมีผลต่อสมรรถนะรหัสแอลดีพีซี เนื่องจากการเข้า-ถอดรหัสแอลดีพีซีนั้นจำเป็นต้องใช้เมทริกซ์พาริตีเช็กส่งผ่านข้อมูลเพื่อให้สามารถแก้ไขข้อผิดพลาดของข้อมูลที่ส่งในช่องสัญญาณที่มีสัญญาณรบกวนได้ นั่นคือถ้ามีการออกแบบเมทริกซ์พาริตีเช็กที่ดีจะส่งผลต่อสมรรถนะในด้านการเข้า-ถอดรหัส ทำให้ง่าย เร็ว ลดความซับซ้อนลง และสามารถแก้ไขข้อผิดพลาดของข้อมูลที่ส่งได้ โดยประเด็นความง่ายพิจารณาจากการคำนวณในการเข้ารหัสเป็นเพียงการคูณเมทริกซ์ที่มีความหนาแน่นของจำนวนเลขหนึ่งน้อยมากเมื่อเทียบกับขนาดของเมทริกซ์เอง อีกทั้งยังส่งผลให้การเข้ารหัสทำได้เร็ว ส่วนประเด็นความซับซ้อนพิจารณาจากวิธีการสร้างเมทริกซ์พาริตีเช็กหรือจำนวนรอบของการถอดรหัสแบบวนซ้ำ วิธีการสร้างเมทริกซ์พาริตีเช็ก จำแนกได้ 2 กลุ่มหลัก คือ กลุ่มสร้างด้วยวิธีการแน่นอนเรียกว่ากลุ่มโครงสร้างกับกลุ่มสร้างด้วยวิธีการสุ่ม ซึ่งงานวิจัยนี้ สร้างด้วยวิธีการกลุ่มโครงสร้าง ด้วยการนำคณิตศาสตร์พื้นฐานมาประยุกต์ ทำให้วิธีการสร้างเมทริกซ์พาริตีเช็กลดความซับซ้อนและการถอดรหัสแบบวนซ้ำมีจำนวนรอบน้อยลง แต่ยังคงสามารถแก้ไขข้อผิดพลาดได้ดี พิจารณาจากค่าอัตราความผิดพลาดของข้อมูลลดลง ส่วนวิธีการอื่นต้องคำนวณด้วยคณิตศาสตร์ที่ยุ่งยากหรือเป็นคณิตศาสตร์ขั้นสูง นอกจากนั้นยังสามารถจำแนกเมทริกซ์พาริตีเช็กได้เป็น 2 กลุ่ม เมื่อพิจารณาจากจำนวนเลขหนึ่งของเมทริกซ์พาริตีเช็ก คือจำนวนเลขหนึ่งคงที่กับไม่คงที่ ซึ่งแบบจำนวนเลขหนึ่งไม่คงที่นั้น จะสามารถทำให้มีอัตรารหัสสูงได้ ดังนั้นจึงเลือกพิจารณาที่จำนวนเลขหนึ่งไม่คงที่ ด้วยอัตรารหัสสูงและขนาดความยาวบล็อกสั้น ซึ่งเป็นสิ่งที่ท้าทาย เนื่องจากรหัสแอลดีพีซีจะมีสมรรถนะดี เมื่อขนาดความยาวบล็อกยาวด้วยอัตรารหัสสูง งานวิจัยนี้มีวัตถุประสงค์เพื่อออกแบบเมทริกซ์พาริตีเช็กสำหรับรหัสแอลดีพีซีแบบไม่คงที่ที่มีขนาดความยาวบล็อกสั้น เพื่อนำไปประยุกต์ใช้กับเทคโนโลยีการสื่อสารไร้สายในระบบที่มีหน่วยความจำน้อยและช่องสัญญาณที่จำกัดหรือกับเทคโนโลยีการบันทึกข้อมูล เช่น ระบบฮาร์ดดิสก์ เป็นต้น

การวิจัยครั้งนี้ ได้ศึกษาวิธีการใหม่เพื่อออกแบบเมทริกซ์พาริตีเช็กหลายแนวทางอันได้แก่ พิจารณาที่รูปแบบของเมทริกซ์พาริตีเช็กโดยลดเลขหนึ่งลงด้วยการตัดจำนวนเลขหนึ่งในแนวเส้น

ทแยงมุมทั้งด้านซ้ายและขวาของเมทริกซ์โดยใช้หลักการทรานสโพรสเมทริกซ์ ซึ่งผลแสดงให้เห็นว่าวิธีการได้มาซึ่งเมทริกซ์พาริตีเช็กนั้นง่ายไม่ซับซ้อน การเข้า-ถอดรหัสเร็วขึ้น จำนวนรอบการวนซ้ำลดลง เนื่องจากสามารถลดจำนวนเลขหนึ่งลงได้ และที่ขนาดความยาวบล็อกสั้นมีสมรรถนะเหมือนวิธีการอื่นในกลุ่มเดียวกัน แต่ที่ความยาวบล็อกยาวสมรรถนะด้อยลงเล็กน้อย จากนั้นได้พิจารณาวิธีการคำนวณและกำหนดค่าตัวแปรเพื่อใช้สร้างเมทริกซ์พาริตีเช็กให้มีสมรรถนะดีขึ้น โดยพัฒนาจากรหัสแอลดีพีซีแบบอาร์เรย์เป็นพื้นฐานที่มีตัวแปรเริ่มต้นในการสร้างรหัส คือค่า j, k และ p เป็นจำนวนเต็มบวก เมื่อ j<k<p และค่า p ต้องเป็นจำนวนเฉพาะ นำมาพัฒนาผสมกับวิธีการกำหนดรูปแบบการเลื่อนวนเป็นแบบสมมาตรที่ส่งผลให้เมทริกซ์ปราศจากลูป 4 พร้อมกับปรับค่า p เป็นแบบจำนวนเต็มบวกที่ไม่ใช่จำนวนเฉพาะเพียงอย่างเดียว ผลการทดสอบสมรรถนะพบว่า ขนาดความยาวบล็อกสั้นสมรรถนะของรหัสไม่ดี ส่วนขนาดความยาวบล็อกยาวสมรรถนะของรหัสใกล้เคียงกับวิธีอื่นที่เป็นกลุ่มเดียวกัน และสามารถกำหนดค่าตัวแปรเพื่อสร้างเมทริกซ์พาริตีเช็กสำหรับการเข้า-ถอดรหัสแอลดีพีซีได้ขนาดความยาวบล็อกที่หลากหลายมากขึ้น จากนั้นได้มีการนำทฤษฎีทางคณิตศาสตร์พื้นฐานที่เรียกว่า เมจิกสแควร์หรือจัตุรัสมหัศจรรย์ ประยุกต์ใช้เพื่อออกแบบเมทริกซ์พาริตีเช็ก เป็นวิธีการใหม่ซึ่งไม่เคยมีใครนำมาใช้สร้างเมทริกซ์พาริตีเช็ก เพื่อให้ได้ผลการทดสอบสมรรถนะที่ดีเหมือนการออกแบบด้วยวิธีการสุ่มหรือเหนือกว่าหรือเข้าใกล้ขีดจำกัดของแชนนอลตามทฤษฎีข่าวสารของแชนนอล ภายใต้ระบบที่มีสัญญาณรบกวนแบบเกาส์สีขาว ผลการทดสอบแสดงให้เห็นว่าการออกแบบด้วยวิธีการสร้างเมทริกซ์พาริตีเช็กแบบใหม่ที่ใช้เมจิกสแควร์มาประยุกต์ในกระบวนการสร้างนั้น มี 3 วิธีที่แตกต่างกัน แต่ผลการทดสอบพบว่ามีเพียงวิธีเดียว ได้แก่ วิธีที่สาม (III) สามารถส่งผลให้สมรรถนะดีได้ คือ การเข้ารหัสสามารถกระทำได้ด้วยวิธีการอย่างง่าย จำนวนรอบการถอดรหัสแบบวนซ้ำและอัตราความผิดพลาดของข้อมูลน้อยลงเหนือกว่าการออกแบบด้วยวิธีการสุ่มและวิธีในกลุ่มเดียวกันที่มีขนาดความยาวบล็อกสั้นและอัตรารหัสสูง วิธีการได้มาซึ่งเมทริกซ์พาริตีเช็กมีกระบวนการขั้นตอนที่ง่ายและซับซ้อนน้อยกว่าวิธีการในกลุ่มเดียวกัน รวมถึงวิธีการสร้างยังใช้หลักการทางสถิติมาประยุกต์เพื่อประกอบในการพิจารณาตัดสินใจดูสภาพการกระจายตัวของเลขหนึ่งในเมทริกซ์พาริตีเช็กเป็นแบบปกติหรือไม่ จากผลการทดสอบทำให้เห็นถึงความสัมพันธ์กันว่า ถ้าการกระจายตัวของเลขหนึ่งในเมทริกซ์พาริตีเช็กเข้าใกล้โค้งแบบปกติมากกว่า จะส่งผลให้สมรรถนะดีกว่าเมทริกซ์พาริตีเช็กที่มีค่าการกระจายตัวของเลขหนึ่งที่เข้าใกล้โค้งแบบปกติน้อยกว่า

คำสำคัญ: รหัสแอลดีพีซีแบบไม่คงที่, เมจิกสแควร์, รหัสความยาวบล็อกสั้น, เมทริกซ์พาริตีเช็ก

Thesis Title	Design of Parity-Check Matrix for Short Block Irregular LDPC Codes
Student	Chutima Prasartkaew
Student ID.	51064903
Degree	Doctor of Philosophy
Program	Data Storage Technology
Year	2013
Thesis Advisor	Assoc.Prof.Dr.Somsak Choomchuay

ABSTRACT

The structure of parity-check matrix effects on the LDPC codes performance. As the error correction code, LDPC codes with the appropriate parity-check matrix can be efficiently applied in the coding algorithm for information transfer the noisy channel. Therefore, the good parity-check matrix should be designed to obtain the simple, quick, low complexity encoding-decoding scheme. Regarding the simple and quick parity check matrix design, it is attributed to the matrix multiplication encoding where this matrix is so sparse of 1's, compared to its size. So, the easy encoding-decoding calculation can be quickly done. As the LDPC codes are linear block codes defined by a sparse parity check matrix, there two groups of parity-check matrix: structured and random parity check matrix groups. The first group can be constructed using the known algorithm (can be possibly implemented) while the random one has no curtain algorithm (cannot be implemented). This research is focused on the construction design for the first group using the basic mathematical algebra (while almost all the existing proposed algorithms are derived from advanced mathematics). From the sparse point of view, there are two kinds of parity check matrix, regular and irregular. The matrix elements can be randomly generated with the desirable condition that the number of 1s in each row and column must be the same; this is known as a regular structure LDPC code while another kind, called 'irregular', is vice versa. Literature reveals that the irregular LDPC code yields better performance. Usually, LDPC codes are efficiently implemented for long block length and high code rate. In many applications, however, short block length with high code rate is required. This research aims at designing the parity-check matrix for the short

block length and high code rate irregular LDPC codes. This code can be suitable for any applications that the block length less than 1000 bits is required.

In this research, few parity check matrix designs are proposed. Firstly, the parity check matrix feature was considered and the quantity of 1's was reduced by cutting all 1's in the right and left diagonal of matrix using the matrix transpose. With the proposed algorithm, the ease parity check matrix construction, quick encode-decode and low iteration LDPC codes were obtained. The results also demonstrated that, for short block length, the obtained performance is similar to the existing codes, in the same class. For long block length, however, the obtained performance is slightly lower than the existing codes. Secondly, the new approach is first proposed to construct high performance irregular LDPC codes which similar to random parity check matrix or close to Shannon limit. The mathematical theory, called 'magic square', was applied for this proposed parity-check matrix construction. Three methods of parity check matrix construction are proposed in this research. Over the AWGN channel, the results demonstrated that only the third method (method III) is promising, the better performance can be obtained. It can be said that, with the proposed algorithm, the simple encode scheme can be used, and the number of iteration and coding error is lower compared to the random construction of codes in the same class; short block length and high code rate. It can be concluded that, with the proposed design, the easier and simpler parity check matrix construction is obtained. Moreover, the statistical parameter used for selecting the suitable magic square has been proposed. The normal curve distribution of 1's in the constructed parity check matrix is a criterion parameter for this selection. The study results show that the higher normal distribution of 1's, the better code performance is obtained.

Keywords: Irregular LDPC Codes, Magic square, Short block length, Parity check matrix

กิตติกรรมประกาศ

ในการวิจัยครั้งนี้ ผู้วิจัยขอขอบพระคุณ รองศาสตราจารย์ ดร.สมศักดิ์ ชุมช่วย อาจารย์ที่ปรึกษา ที่กรุณาให้คำปรึกษาแนะนำเกี่ยวกับการทำวิจัยจนสำเร็จลุล่วงไปได้ด้วยดี รวมถึงการให้ประสบการณ์และแนวทางในการดำเนินการวิจัยที่เป็นประโยชน์ติดตัวผู้วิจัยให้สามารถคิด วิเคราะห์และทำวิจัยอย่างเป็นระบบ ขอขอบคุณมหาวิทยาลัยเทคโนโลยีราชมงคลธัญบุรีที่สนับสนุนทุนการศึกษาระดับปริญญาเอกครั้งนี้ ขอขอบคุณคณะกรรมการสอบวิทยานิพนธ์ที่ให้คำแนะนำที่เป็นประโยชน์เพื่อให้วิทยานิพนธ์มีความสมบูรณ์เรียบร้อย ขอขอบคุณนักวิจัยในห้องปฏิบัติการระบบสัญญาณดิจิตอล (DSS Lab) ภาควิชาวิศวกรรมอิเล็กทรอนิกส์ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง และสุดท้ายขอขอบพระคุณ รองศาสตราจารย์ ดร.ปิยะ โควินท์ทวีวัฒน์ ที่กรุณาให้คำปรึกษาและคำแนะนำเกี่ยวกับช่องสัญญาณฮาร์ดดิสก์และการใช้โปรแกรม MathLab มา ณ โอกาสนี้ด้วย ประโยชน์อันพึงมีที่เกิดจากงานวิจัยนี้ ผู้วิจัยขออุทิศความดีนั้นให้แก่ บิดา-มารดา ครู-อาจารย์ ตลอดจนผู้ให้การช่วยเหลือในงานวิจัยนี้ทุกท่าน ส่งผลให้ท่านมีความเจริญและดำเนินกิจอันใดที่ดีต่อตนเอง ครอบครัว และสังคมขอให้ประสบความสำเร็จทุกประการ

ชุติมา ประสาทแก้ว

สารบัญ

หน้า

สารบัญ(ต่อ)

หน้า

สารบัญ(ต่อ)

หน้า

สารบัญตาราง

สารบัญรูป

สารบัญรูป(ต่อ)

รายการคำย่อและสัญลักษณ์

AWGN	ช่องสัญญาณรบกวนแบบเกาส์สีขาว
BER	อัตราความผิดพลาดของบิตข้อมูล
CRT	ทฤษฎีเศษเหลือของจีน
ECC	รหัสแก้ไขข้อผิดพลาด
GB	หน่วยวัดความจุของหน่วยความจำขนาดประมาณพันล้านบิต
IMAC	รหัสแอลดีพีซีแบบอินเทอร์ลีฟมอดิฟายอาร์เรย์
KB	หน่วยวัดความจุของหน่วยความจำขนาดประมาณพันบิต
LLR	ค่าความน่าเชื่อถือหรือค่าความน่าจะเป็นไปได้
LMS	วิธีการจัดวางของเมจิกสแควร์แบบ LUX
LUX	วิธีการจัดวางของเมจิกสแควร์กลุ่มที่ 4 แบบหารด้วย 4 ไม่ลงตัว
MAC	รหัสแอลดีพีซีแบบมอดิฟายอาร์เรย์
MB	หน่วยวัดความจุของหน่วยความจำขนาดประมาณล้านบิต
MS	เมทริกซ์ของเมจิกสแควร์ขนาด zxz
MSBA	อัลกอริทึมที่สร้างเมทริกซ์พาริตีเช็กบนพื้นฐานของเมจิกสแควร์
QC	การวนกลับ
SMS	วิธีการจัดวางของเมจิกสแควร์แบบ Strachey
SNR	อัตราส่วนของพลังงานสัญญาณต่อสัญญาณรบกวน
TB	หน่วยวัดความจุของหน่วยความจำขนาดประมาณล้านล้านบิต
exp{.}	ฟังก์ชันเลขชี้กำลัง
mod	การหารเอาเศษ
0	เมทริกซ์ศูนย์
A	ตำแหน่งเลขหนึ่งในเมทริกซ์พาริตีเช็ก
E_b/N_0	อัตราส่วนของพลังงานสัญญาณต่อสัญญาณรบกวน
G	เมทริกซ์กำเนิด
H	เมทริกซ์พาริตีเช็ก
I	เมทริกซ์เอกลักษณ์
K	เมทริกซ์บิตข้อมูล
$L(c_i)$	อัตราส่วนความน่าเชื่อถือแบบล็อก
$L(q_{ij})$	ค่าที่ส่งจากโหนดสัญลักษณ์ที่ i ไปยังโหนดเช็กที่ j
$L(r_{ji})$	ค่าที่ส่งจากโหนดเช็กที่ j ไปยังโหนดสัญลักษณ์ที่ i

รายการคำย่อและสัญลักษณ์(ต่อ)

N	เมทริกซ์คำรหัส
P	เมทริกซ์พาริตี
R	อัตรารหัส
R_d	อัตรารหัสที่ออกแบบ
S	เมทริกซ์สมมาตร
V_{Ni}	ค่าข้อมูลจากทุกโหนดสัญลักษณ์ที่เชื่อมต่อกับโหนดเช็กที่ j ยกเว้นโหนดสัญลักษณ์ที่กำลังพิจารณา
W/H_z	หน่วยของค่าความหนาแน่นสเปกตรัมกำลังงาน
X, Y, Z	จำนวนครั้งในการเลื่อนวนของเมทริกซ์ย่อย
a, b, c, l	ค่าคงที่
c_i	โหนดสัญลักษณ์ตัวที่ i
$\hat{c}$	ค่าซอฟต์เอาต์พุตที่ทำการตัดสินใจแบบฮาร์ด
d_{min}	ค่าระยะห่างต่ำสุด
e	ค่าความสามารถในการแก้ไขความผิดพลาด
f_i	โหนดเช็กตัวที่ i
g	ลูปขนาดใหญ่
h_{ij}	สมาชิกของเมทริกซ์พาริตีเช็กตำแหน่ง i, j ที่เป็นเลขหนึ่ง
j, k, p	ค่าคงที่ที่ใช้ในการออกแบบเมทริกซ์พาริตีเช็กของรหัสแอลดีพีซี
m	ขนาดแถวของเมทริกซ์พาริตีเช็ก
m_x	ค่าเฉลี่ยตัวที่ x
n	ขนาดคอลัมน์ของเมทริกซ์พาริตีเช็ก
q	จำนวนสมาชิกของฟิลด์จำกัด
s_{ij}	สมาชิกของเมทริกซ์สมมาตรตำแหน่งที่ i, j
t	ค่าความสามารถในการตรวจจับความผิดพลาด
u, v	ค่าตำแหน่งใดๆ ในเมจิกสแควร์
w_c	น้ำหนักของคอลัมน์หรือจำนวนเลขหนึ่งในคอลัมน์ของเมทริกซ์
w_r	น้ำหนักของแถวหรือจำนวนเลขหนึ่งในแถวของเมทริกซ์
x_i	สมาชิกแต่ละตัว
x, y	ค่าคงที่
y_i	สัญญาณที่ได้รับผ่านช่องสัญญาณ

รายการคำย่อและสัญลักษณ์(ต่อ)

z	ขนาดของเมจิกสแควร์
α	เมทริกซ์เลื่อนวน
α_{ij}	ค่าของเครื่องหมายของค่า $L(q_{ij})$
β	จำนวนครั้งในการเลื่อนวนของเมทริกซ์ย่อย
β_{ij}	ค่าสัมบูรณ์ของค่า $L(q_{ij})$
ρ	ค่าความหนาแน่นของเลขหนึ่งของแต่ละแถว
λ	ค่าความหนาแน่นของเลขหนึ่งของแต่ละหลัก
$\emptyset$	ค่า $\log\{(e^x+1)/(e^x-1)\}$
σ^2_x	ค่าความแปรปรวนหรือความแปรผันตัวที่ x

บทที่ 1
บทนำ

บทนี้จะกล่าวถึงความเป็นมาและความสำคัญของปัญหาที่เป็นที่มาของการทำงานวิจัยครั้งนี้ รวมถึงบอกวัตถุประสงค์สมมุติฐานทฤษฎีหรือแนวความคิดที่ใช้ในการวิจัย ขอบเขตและขั้นตอนการวิจัยโดยมีรายละเอียดดังต่อไปนี้

1.1 ความเป็นมาและความสำคัญของปัญหา

เพื่อให้สามารถจัดเก็บข้อมูลที่มีอยู่จำนวนมาก อุปกรณ์สำหรับการบันทึกข้อมูลที่ใช้กันทั่วไปในปัจจุบัน อาทิ แผ่นบันทึกแม่เหล็ก ฮาร์ดดิสก์ไดร์ฟ แผ่นซีดี และแผ่นดีวีดี ซึ่งมักจะพบการใช้งานอุปกรณ์เหล่านี้อย่างเห็นได้ชัดในเครื่องคอมพิวเตอร์ นอกจากนี้ในระบบเชื่อมต่อกับเครือข่ายต่าง ๆ ก็ยิ่งส่งผลทำให้เกิดความต้องการในการจัดเก็บข้อมูลมากขึ้นตลอดเวลา โดยทั่วไปเครื่องคอมพิวเตอร์จะประกอบด้วย ไมโครโปรเซสเซอร์ หน่วยความจำหลักแบบสารกึ่งตัวนำ และอุปกรณ์สำหรับการบันทึกข้อมูล โดยไมโครโปรเซสเซอร์จะทำหน้าที่หลักในการเข้าถึงและจัดเก็บข้อมูลในหน่วยความจำหลักแบบสารกึ่งตัวนำ เช่น ดีแรม (Dynamic Random Access Memory : DRAM) เนื่องจากมีความรวดเร็วในการเข้าถึงข้อมูล แต่จัดเป็นหน่วยความจำแบบลบเลือน (Volatile Memory) และมีความจุข้อมูลน้อย ดังนั้นจึงมีความต้องการใช้งานอุปกรณ์เสริมสำหรับการบันทึกข้อมูล ที่ได้กล่าวไปแล้วเป็นแบบหน่วยความจำไม่ลบเลือน (Non-Volatile Memory) ถึงแม้อุปกรณ์เหล่านี้จะมีความจุข้อมูลสูง แต่ความเร็วในการเข้าถึงข้อมูลช้ากว่าดีแรม ซึ่งความเร็วในการเข้าถึงมีราคาแพง คือ ราคาต่อหนึ่งกิกะไบต์ (Gigabyte : GB) จะมีราคาสูงมาก เมื่อเทียบกับอุปกรณ์บันทึกข้อมูลทั่วไป

อย่างไรก็ตาม ฮาร์ดดิสก์ไดร์ฟจัดเป็นอุปกรณ์ที่คุ้มค่ามากที่สุด เมื่อพิจารณาจากปัจจัยหลายๆ อย่าง ได้แก่ ความจุข้อมูล ราคา ความน่าเชื่อถือ และประสิทธิภาพ เพราะฉะนั้นในงานวิจัยนี้จึงสนใจเทคโนโลยีที่จะเพิ่มประสิทธิภาพเกี่ยวกับการบันทึกข้อมูลสำหรับอุปกรณ์ฮาร์ดดิสก์ไดร์ฟ โดยพิจารณาระบบสื่อสารข้อมูลที่มีระบบคล้ายกัน และมีปัญหาเช่นเดียวกันคือสัญญาณรบกวน ซึ่งวิธีการแก้ปัญหาสัญญาณรบกวนจึงใช้หลักการเหมือนระบบการสื่อสารมาใช้เป็นกรณีศึกษา ที่ใช้วิธีการเข้ารหัสข้อมูลก่อนจะส่งสัญญาณเหมือนการเข้ารหัสก่อนบันทึกข้อมูลลงฮาร์ดดิสก์ เพื่อให้สัญญาณที่จะอ่านข้อมูลจากฮาร์ดดิสก์ไดร์ฟนั้นมีประสิทธิภาพที่ดี เทคโนโลยีของฮาร์ดดิสก์ไดร์ฟได้พัฒนาอย่างรวดเร็ว จนมีความจุมากขึ้นเป็นระดับเทราไบต์ (Terabyte : TB) นั่นก็หมายความว่า เพิ่มประสิทธิภาพการทำงานเพื่อที่จะทำให้มีความจุเพิ่มขึ้น และคงไว้ซึ่งขนาดของฮาร์ดดิสก์ที่เล็กเท่าเดิมหรือเล็กลง

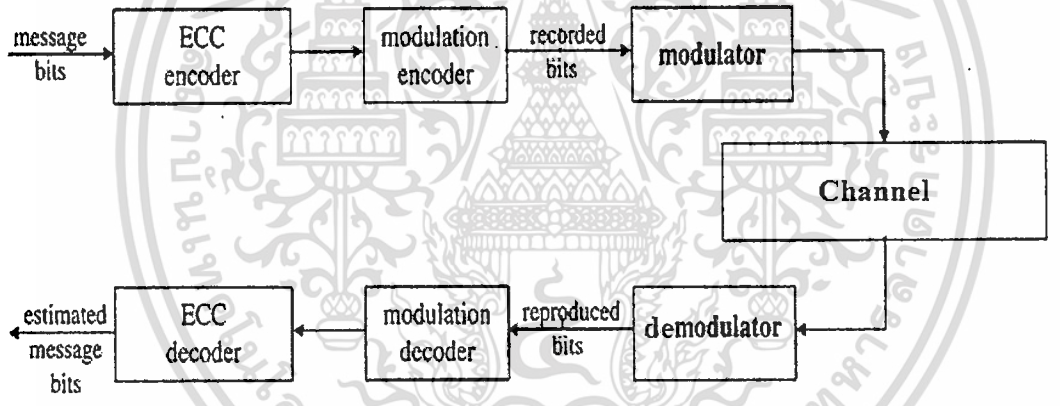
แนวโน้มพัฒนาการของระบบฮาร์ดดิสก์จะมีความจุมากกว่า 1 เทราบิตต่อตารางนิ้วนั้น การเขียน-อ่านข้อมูล ที่อัตราถ่ายข้อมูลของปรีแอมป์อาจสูงได้ถึง 10 กิกะบิตต่อวินาที ในขณะที่ความเร็วรอบในการหมุนจะพัฒนาไปสู่ 22,000 รอบต่อนาที เนื่องจากความจุของสื่อบันทึกข้อมูลที่สูงขึ้น ความเร็วรอบที่สูงขึ้นและอัตราการส่งถ่ายข้อมูลที่เร็วขึ้น สัญญาณรบกวนและความผิดพลาดสามารถเกิดได้ง่ายขึ้นไม่ว่าเพราะขนาดของบิตบนสื่อที่เล็กลงวงจรทำงานที่ความเร็วสูง รวมทั้งการสั่นเนื่องจากการหมุนของแผ่นที่ความเร็วสูงการควบคุมอัตราความผิดพลาดของข้อมูล (Bit Error Rate: BER) และอัตราส่วนของพลังงานสัญญาณต่อสัญญาณรบกวน (Signal to Noise Ratio : SNR) จึงเป็นประเด็นที่สำคัญ

การสื่อสารรับ-ส่งข้อมูลหลีกเลี่ยงไม่ได้ที่จะต้องมีสัญญาณรบกวน (Noise) ที่มีอยู่ในระบบสื่อสารทุกชนิด และถ้าสัญญาณรบกวนมีค่าสูงจะส่งผลกระทบถึงอัตราบิตผิดพลาดที่จะเกิดขึ้นสูงตามไปด้วย จึงทำให้ต้องเพิ่มสมรรถนะของสัญญาณให้มีอัตราความผิดพลาดของข้อมูลน้อยลง หรืออยู่ในระดับที่ยอมรับได้ ซึ่งสามารถทำได้โดยปรับปรุงตัวกลางและช่องสัญญาณที่ใช้ในการส่งข้อมูล เพิ่มกำลังส่งสัญญาณที่ภาคส่งให้มีสัดส่วนของสัญญาณที่ใช้ในการส่งผ่านช่องสัญญาณกับสัญญาณรบกวนใช้สูงขึ้น แต่วิธีการเหล่านี้อาจไม่เหมาะเนื่องจากการปรับปรุงตัวกลาง หรือการเพิ่มกำลังส่งของสัญญาณนั้นหมายถึงค่าใช้จ่ายที่สูงขึ้นมีอีกวิธีที่นิยมเรียกว่าการเข้ารหัสช่องสัญญาณ (Channel Coding) หรือที่เรียกว่า รหัสแก้ไขข้อผิดพลาด (Error Correcting Code: ECC) กระทำโดยการเพิ่มจำนวนบิตพิเศษที่เรียกว่า พาริตีบิต (Parity Bitหรือ Redundant Bit) เข้าไปกับชุดข้อมูลเดิมก่อนส่งผ่านช่องสัญญาณ โดยบิตพิเศษที่เพิ่มเข้ามาจะช่วยให้สามารถตรวจจับความผิดพลาดได้ หรือถ้าเพิ่มเข้าไปในจำนวนที่มากพอ ก็อาจจะสามารถแก้ไขความผิดพลาดของข้อมูลได้ด้วย การเข้ารหัสข้อมูลแบ่งออกเป็น 2 ประเภท คือ รหัสซีเควนเชียล(Sequential Code) และรหัสบล็อก (Block Code) รหัสสองประเภทนี้ต่างกันตรงขั้นตอนเข้ารหัสโดยที่รหัสซีเควนเชียล ได้แก่ รหัสคอนโวลูชันเป็นการเข้ารหัสข้อมูลทีละบิต รหัสคอนโวลูชันที่รู้จักกันดีคือรหัสเทอร์โบ (Turbo Codes) [1] ที่ทำงานเข้าใกล้ขีดจำกัดของแชนนอลในขณะที่รหัสบล็อกจะเข้ารหัสข้อมูลทีละบล็อก และมีงานวิจัยพบว่ารหัสบล็อกเชิงเส้นที่เรียกว่า รหัสแอลดีพีซี (Low-Density Parity-Check Codes : LDPC Codes) มีสมรรถนะทำงานเข้าใกล้ขีดจำกัดของแชนนอลเช่นกัน [2] ตามทฤษฎีข่าวสารของแชนนอล (Shannon's Information Theory)

ระบบการจัดเก็บข้อมูลดิจิตอล (Digital Data Storage System) ในฮาร์ดดิสก์ไดร์ฟสามารถอธิบายระบบได้ดังนี้ บิตข่าวสาร (Message Bits) จะถูกเข้ารหัสโดยวงจรเข้ารหัสแก้ไขข้อผิดพลาด (ECC Encoder) จากนั้นข้อมูลที่เข้ารหัสจะถูกเข้ารหัสอีกครั้งหนึ่งด้วยวงจรเข้ารหัสมอดุโล(Modulation Encoder) เพื่อทำหน้าที่ปรับคุณสมบัติข้อมูลให้เหมาะสมกับช่องสัญญาณของฮาร์ดดิสก์ไดร์ฟ ข้อมูลเอาต์พุตที่ได้จากวงจรเข้ารหัสมอดุโลจะถือว่าเป็นข้อมูลที่จะถูกเขียนเข้าไปในสื่อบันทึกข้อมูล การอ่านข้อมูลจากสื่อบันทึกข้อมูลผลลัพธ์ที่ได้จากการอ่านจะออกมาเป็นสัญญาณ

รูปคลื่นแรงดันไฟฟ้า เรียกว่า สัญญาณ Read-Back สัญญาณจะถูกถอดรหัสด้วยวงจรถอดรหัสมอดุโล (Modulation Decoder) และวงจรถอดรหัสแก้ไขข้อผิดพลาด (ECC Decoder) เพื่อหาค่าประมาณของบิตข่าวสารที่ต้องการ โดยข้อดีของการเข้ารหัสแก้ไขข้อผิดพลาดจะช่วยให้ข้อผิดพลาดในระบบน้อยลง ส่งผลให้ความน่าเชื่อถือของการจัดเก็บข้อมูลในฮาร์ดดิสก์ไดร์ฟเพิ่มสูงขึ้น ส่วนวงจรถอดรหัสแก้ไขข้อผิดพลาดสามารถที่จะแก้ไขข้อผิดพลาดที่เกิดขึ้นที่วงจรภาครับได้อย่างอัตโนมัติ โดยที่รหัสแอลดีพีซีจะเป็นรหัสแก้ไขข้อผิดพลาดประเภทหนึ่งที่ใช้ในฮาร์ดดิสก์ไดร์ฟเพราะรหัสแอลดีพีซีมีความสามารถในการแก้ไขข้อผิดพลาดแบบหลายบิตติดกันได้อย่างมีประสิทธิภาพ

ระบบการสื่อสารสามารถจำลองเป็นแผนภาพทั่วไปได้ดังรูปที่ 1.1 บิตข่าวสารถูกเข้ารหัสโดยวงจรเข้ารหัสแก้ไขข้อผิดพลาดจากนั้นข้อมูลจะถูกเข้ารหัสอีกครั้งด้วยวงจรเข้ารหัสมอดุโลเพื่อทำหน้าที่ปรับคุณสมบัติข้อมูลให้เหมาะสมกับช่องสัญญาณข้อมูลเอาต์พุตที่ได้จากวงจรเข้ารหัสมอดุโลเป็นข้อมูลที่จะส่งไปยังผู้รับในระบบสื่อสาร แล้วถูกถอดรหัสด้วยวงจรถอดรหัสมอดุโลและวงจรถอดรหัสแก้ไขข้อผิดพลาดเพื่อหาบิตข่าวสารที่ส่งมา



รูปที่ 1.1 แผนภาพแสดงส่วนประกอบหลักในระบบสื่อสาร

รหัสพาริตีเช็กความหนาแน่นต่ำได้รับการคิดค้นขึ้นในปีค.ศ. 1960 โดย R. G.Gallager [3] แห่งสถาบัน MIT ประเทศสหรัฐอเมริกาและต่อมาในปีค.ศ. 1962 บทความได้รับการตีพิมพ์โดยใช้ชื่อว่า Low-Density Parity-Check Codes : LDPC Codesในวารสาร IRE Transactions onInformation Theory แต่ในขณะนั้นแทบจะไม่ได้รับความสนใจเนื่องจากหลาย ๆ เหตุผล เช่น ปัญหาเรื่องฮาร์ดแวร์ซึ่งในสมัยนั้นฮาร์ดแวร์ยังมีความสามารถที่ต่ำไม่เหมาะกับรหัสที่มีความซับซ้อน จึงทำให้รหัสพาริตีเช็กความหนาแน่นต่ำเงียบหายไปจนกระทั่ง [2] ได้นำเสนอการค้นพบที่เกี่ยวข้องกับรหัสพาริตีเช็กความหนาแน่นต่ำรู้จักกันในนามรหัสแอลดีพีซีโดยใช้วิธีการประมาณด้วยขั้นตอนของ Belief Propagation จึงทำให้รหัสแอลดีพีซีได้รับความสนใจนำกลับมาใช้ใหม่ [4] ได้แสดงให้เห็นถึงสมรรถนะของรหัสแอลดีพีซีที่เข้าใกล้ขีดจำกัดของแชนนอลโดยห่างกัน 0.0045 dB เท่านั้น เนื่องจาก

มีประสิทธิภาพในการแก้ไขข้อผิดพลาดได้ดีเยี่ยมทำให้นักวิจัยต่าง ๆ สนใจกันอย่างมาก การวิจัยจะมุ่งเน้นไปที่การวิเคราะห์และปรับปรุงรหัสให้ดีขึ้น โดยสมรรถนะรหัสแอลดีพีซีนั้นขึ้นกับโครงสร้างของเมทริกซ์พาริตีเช็กที่มีผลต่อการเข้า-ถอดรหัสแอลดีพีซี จำเป็นต้องใช้เมทริกซ์พาริตีเช็กส่งผ่านข้อมูลเพื่อให้สามารถแก้ไขข้อผิดพลาดของข้อมูลที่ส่งในช่องสัญญาณที่มีสัญญาณรบกวนได้ นั่นคือ ถ้ามีการออกแบบเมทริกซ์พาริตีเช็กที่ดีจะส่งผลต่อสมรรถนะในด้านการเข้า-ถอดรหัส ทำให้ง่าย เร็ว ลดความซับซ้อนลง และสามารถแก้ไขข้อผิดพลาดของข้อมูลที่ส่งได้

ปัจจัยสำคัญที่มีผลต่อคุณภาพของระบบการสื่อสารและระบบบันทึกข้อมูลเชิงแม่เหล็ก คือ ปัญหาการแทรกสอดระหว่างสัญลักษณ์ (Intersymbol Interference : ISI) เนื่องจากความหนาแน่นในการสื่อสารและการบันทึกข้อมูลมีปริมาณที่สูง อัตราการแทรกสอดทางด้านสัญลักษณ์ก็มีค่าสูงตามไปด้วย ดังนั้นวิธีการเข้ารหัสแก้ไขข้อผิดพลาดจึงเป็นทางเลือกที่เหมาะสม ในการเพิ่มประสิทธิภาพการทำงานของระบบสื่อสารและระบบฮาร์ดดิสก์ โดยเลือกใช้รหัสแอลดีพีซี แต่เนื่องจากรหัสแอลดีพีซีต้องใช้เมทริกซ์พาริตีเช็กในกระบวนการเข้า-ถอดรหัส ซึ่งสามารถจำแนกเมทริกซ์พาริตีเช็กได้เป็น 2 กลุ่มเมื่อพิจารณาจากจำนวนเลขหนึ่งของเมทริกซ์พาริตีเช็ก คือ จำนวนเลขหนึ่งคงที่กับไม่คงที่ โดยแบบจำนวนเลขหนึ่งไม่คงที่นั้นจะสามารถทำให้มีอัตรารหัสสูงได้ ดังนั้นจึงเลือกพิจารณาที่จำนวนเลขหนึ่งไม่คงที่ ด้วยอัตรารหัสสูงและขนาดความยาวบล็อกสั้น ซึ่งเป็นสิ่งที่ท้าทายเนื่องจากรหัสแอลดีพีซีจะมีสมรรถนะดีเมื่อขนาดความยาวบล็อกยาวด้วยอัตรารหัสสูง และมีรูปแบบการสร้างเมทริกซ์พาริตีเช็ก 2 กลุ่มได้แก่กลุ่มสร้างแบบวิธีการสุ่มกับแบบโครงสร้าง ในแต่ละแบบมีข้อดี-ข้อเสียต่างกัน คือ การสร้างเมทริกซ์พาริตีเช็กแบบวิธีการสุ่ม ส่งผลให้สมรรถนะของอัตราความผิดพลาดของข้อมูลดีกว่า แต่วิธีการสร้างเมทริกซ์พาริตีเช็กนั้นยากต่อการออกแบบด้านฮาร์ดแวร์เพื่อนำไปใช้ประโยชน์จริง ส่วนแบบโครงสร้างมีผลตรงข้ามกับแบบวิธีการสุ่ม ดังนั้นการวิจัยครั้งนี้จึงสนใจที่ออกแบบเมทริกซ์พาริตีเช็กของรหัสแอลดีพีซีให้มีสมรรถนะดีเหมือนหรือดีกว่าแบบวิธีการสุ่ม บนโครงสร้างเมทริกซ์พาริตีเช็กที่มีรูปแบบแน่นอน และลดความซับซ้อนในการออกแบบลงด้วยวิธีการแบบใหม่ที่ไม่มีใครเคยใช้มาก่อน ได้แก่ การใช้ทฤษฎีทางคณิตศาสตร์พื้นฐานมาประยุกต์เพื่อให้เหมาะสมกับการสื่อสารที่ความจุและความเร็วสูงประยุกต์ใช้กับเทคโนโลยีการสื่อสารไร้สายภายใต้ระบบที่มีหน่วยความจำที่จำกัดเป็นประโยชน์ต่อระบบสื่อสารข้อมูลทั่วไปและฮาร์ดดิสก์ไดร์ฟด้วยเช่นกัน

1.2 วัตถุประสงค์การวิจัย

เพื่อออกแบบเมทริกซ์พาริตีเช็กสำหรับรหัสแอลดีพีซีแบบไม่คงที่ที่มีขนาดความยาวบล็อกสั้นให้มีสมรรถนะที่ดีเหมือนหรือเหนือกว่าการออกแบบด้วยวิธีการสุ่ม โดยมีอัตรารหัสสูงและมีอัตราความผิดพลาดของข้อมูลลดลง

1.3 สมมุติฐานการวิจัย

รูปแบบเมทริกซ์พาริตีเช็กที่ออกแบบใหม่ของรหัสแอลดีพีซีแบบไม่คงที่ที่มีขนาดความยาวบล็อกสั้นภายใต้ระบบสัญญาณรบกวนแบบเกาส์สีขาวที่อัตรารหัสสูงและอัตราความผิดพลาดของข้อมูลลดลงนั้น ส่งผลให้ระบบมีสมรรถนะที่ดีเหมือนหรือเหนือกว่าการออกแบบด้วยวิธีการสุ่ม

1.4 ทฤษฎีหรือแนวความคิดที่ใช้ในการวิจัย

รหัสแอลดีพีซีเป็นรหัสที่ให้เกน (Gain) ของอัตราส่วนของพลังงานสัญญาณต่อสัญญาณรบกวนเข้าใกล้ขีดจำกัดของแชนนอลรหัสแอลดีพีซีเป็นรหัสที่มีจำนวนของเลขหนึ่งน้อยเมื่อเทียบกับขนาดของเมทริกซ์พาริตีเช็ก ทั้งนี้ก็เพื่อให้มีระยะห่างต่ำสุดของรหัส d_{min} (Minimum Distance) สูง เป็นรหัสช่องสัญญาณแบบบล็อกเชิงเส้นที่เข้ารหัสผ่านเมทริกซ์พาริตีเช็กที่มีโครงสร้างดั้งเดิมแบบสุ่ม สามารถนำเสนอด้วยกราฟความสัมพันธ์ที่เรียกว่า กราฟแทนเนอร์ (Tanner Graphหรือ Bipartite Graph) [5]เพื่อช่วยในการออกแบบหรือเพิ่มความเข้าใจการถอดรหัสง่ายขึ้นรหัสแอลดีพีซีได้รับความสนใจอย่างแพร่หลายและได้รับความสนใจมาจนถึงปัจจุบัน ด้วยคุณสมบัติที่ดีมีสมรรถนะสูงทำงานเข้าใกล้ขีดจำกัดของแชนนอล [2] รหัสแอลดีพีซีเป็นรหัสที่ดีสำหรับการแก้ไขข้อผิดพลาดของข้อมูล มีระดับความผิดพลาดต่ำ (Low Error Floor) ในการถอดรหัสแอลดีพีซีมีความเป็นเชิงเส้นในรูปแบบของเวลาเหมาะสมกับการทำงานแบบคู่ขนานคำรหัสที่ยาวมากๆ จะทำให้มีการวนซ้ำมากขึ้น และจะทำให้ค่าอัตราส่วนของพลังงานสัญญาณต่อสัญญาณรบกวนต่ำคำรหัสที่สั้นจะเข้ารหัสง่าย และจะได้ค่าอัตรารหัส (Code Rate : R) ≈ 1 ในช่วงแรกที่ Gallager [3] เสนอไม่ได้รับความสนใจ ต่อมาได้รับการนำเสนอใหม่โดย [2] ที่มีการเสนอว่ารหัสแอลดีพีซีมีสมรรถนะเข้าใกล้ขีดจำกัดของแชนนอลเหมือนรหัสเทอร์โบ จากนั้นจึงทำให้รหัสแอลดีพีซีได้รับความสนใจต่อมาจนถึงปัจจุบันที่ได้มีการนำรหัสแอลดีพีซีไปใช้ในระบบฮาร์ดดิสก์และระบบการสื่อสารระยะใกล้เพื่อใช้เป็นรหัสในบล็อกของการเข้ารหัสแก้ไขข้อผิดพลาด

รหัสแอลดีพีซีเมื่อพิจารณาจากจำนวนเลขหนึ่งในเมทริกซ์พาริตีเช็กสามารถแบ่งเป็นสองประเภทหลักคือ รหัสแอลดีพีซีที่มีการกระจายตัวของเลขหนึ่งในเมทริกซ์พาริตีเช็กเป็นแบบคงที่ (Regular LDPC Codes) ซึ่งเป็นรูปแบบเดียวกับรหัสของ R.Gallager และมีผู้ทำงานวิจัย[6]ได้นำเสนอโครงสร้างของเมทริกซ์พาริตีเช็กอาร์เรย์ ที่เรียกว่า รหัสแอลดีพีซีแบบอาร์เรย์มีสมรรถนะของรหัสดีเทียบเท่ากับรหัสแอลดีพีซีที่มีโครงสร้างของเมทริกซ์พาริตีเช็กแบบสุ่มแบบดั้งเดิม ปราศจากลูป 4 มีระดับความผิดพลาดต่ำ และสามารถใช้อัลกอริทึมการเข้ารหัสอย่างง่ายและถอดรหัสแบบทั่วไปได้ คือการเข้ารหัสด้วยการคูณเมทริกซ์และถอดรหัสด้วยหลักการแบบวนซ้ำ รหัสแอลดีพีซีอีกประเภทหนึ่งที่มีการกระจายตัวของเลขหนึ่งเป็นแบบไม่คงที่ (Irregular LDPC Codes) ซึ่งเป็นรหัสที่พัฒนามาจาก Regular LDPC Codes พัฒนาขึ้นในปี 2001 โดย [7] วิธีการของเขาทำให้สมรรถนะของรหัสแอลดีพีซีแบบไม่คงที่ดีกว่าแบบคงที่และมีอีกหลายงานวิจัย [8] และ [9] พบว่ารหัสแอลดีพีซีในแบบ

ไม่คงที่ทำงานได้ดีสำหรับอัตรารหัสที่น้อยกว่าหรือเท่ากับ 3/4 ($R \leq 3/4$)และที่ความยาวคำรหัสมากกว่าหรือเท่ากับ 5,000 ($n \geq 5,000$) มีการพัฒนาอย่างต่อเนื่อง [10] โดยได้นำเสนอโครงสร้างของเมทริกซ์พาริตีเช็กสำหรับรหัสแอลดีพีซีแบบอาร์เรย์ใหม่ ที่เรียกว่า รหัสแอลดีพีซีแบบมอดิฟายอาร์เรย์ (MAC) โดยใช้วิธีการที่เรียกว่า เลื่อนวน (Cyclic Shift) มีสมรรถนะของรหัสที่ดี เทียบเท่ากับรหัสแอลดีพีซีแบบสุ่ม ปราศจากลูป4 มีระดับความผิดพลาดต่ำ สามารถเข้ารหัสด้วยวิธีอย่างง่ายได้ และสามารถใช้อัลกอริทึมการถอดรหัสทั่วไปแบบวนซ้ำได้ [11] ได้ปรับปรุงเมทริกซ์พาริตีเช็กของรหัสแอลดีพีซีแบบมอดิฟายอาร์เรย์ ด้วยเมทริกซ์วนกลับ (Quasi-Cyclic Matrix) ในระดับบิตภายใต้แนวคิดการเลื่อนวนให้กลายเป็นตัวใหม่เรียกว่า รหัสแอลดีพีซีแบบอินเทอร์ลีฟมอดิฟายอาร์เรย์ (IMAC) มีสมรรถนะของรหัสที่ดี เทียบเท่ากับรหัสแอลดีพีซีแบบสุ่ม ปราศจากลูป 4 มีระดับความผิดพลาดต่ำ สามารถเข้ารหัสด้วยวิธีอย่างง่ายได้ สามารถใช้อัลกอริทึมการถอดรหัสทั่วไปแบบวนซ้ำได้ (ความยาวบล็อกยิ่งมากจะทำให้รหัสแอลดีพีซีแบบอินเทอร์ลีฟมอดิฟายอาร์เรย์มีสมรรถนะที่ดีกว่ารหัสแอลดีพีซีแบบมอดิฟายอาร์เรย์)

[12] ได้ประเมินสมรรถนะของรหัสแอลดีพีซีแบบคงที่ ด้วยการออกแบบเมทริกซ์พาริตีเช็กโดยใช้หลักการของเกาส์ซอดอง (จัดรูปให้ค่าแต่ละตัวที่อยู่ใต้แนวเส้นทแยงมุมเป็น 0) จะได้ $G = [IP]$ เมื่อ P ที่ได้จากการใช้หลักการของเกาส์ซอดองที่เสนอวิธีการจัดรูปเมทริกซ์แต่จำนวนเลขหนึ่งไม่ลดน้อย ส่งผลให้การเข้ารหัสมีความซับซ้อน (เมื่อความยาวบล็อกมาก) การออกแบบกำหนดพารามิเตอร์ N, K, W_cที่แตกต่างกัน 3 ชุด คือ 1) N=20, K=10, W_c=1 2) N=20, K=5, W_c=3 และ 3) N=20, K=4, W_c=4 ที่ความยาวบล็อกเท่ากัน แล้วประเมินค่าอัตราความผิดพลาดของข้อมูลกับค่าอัตรารหัส พบว่าการเข้ารหัสมีความซับซ้อนมาก รหัสจะมีประสิทธิภาพดีต้องสร้างคำรหัสยาวและมีจำนวนการวนซ้ำมาก ค่าอัตรารหัสต่ำ และค่าอัตราความผิดพลาดของข้อมูลไม่ดีข้อสังเกต : การกำหนดโครงสร้างเมทริกซ์พาริตีจะช่วยลดความซับซ้อนในการสร้างได้หรืออาจทำได้โดยหลีกเลี่ยงการสร้าง G ทั้งหมดใช้เป็นกำหนดหนึ่งส่วนและคำนวณอีกหนึ่งส่วน [13] ได้ออกแบบรหัสแอลดีพีซีที่มีโครงสร้างแบบไม่คงที่ โดยใช้หลักการเลื่อนวน (Cyclic Shifts) สร้างเมทริกซ์พาริตีเช็กบนหลักการ คือต้องไม่เกิดลูป4 และต้องมีระยะห่างต่ำสุดที่สูง (2 ประเด็นที่มีผลต่อสมรรถนะของรหัส) แก้ปัญหาการเกิดลูป 4 โดยการจัดให้อยู่ในรูปสลับฟันปลา (Zigzag Pattern) และเพิ่มค่าระยะห่างต่ำสุดให้สูงโดยการสร้างอัลกอริทึมที่มีขั้นตอนการทำงาน 6 ขั้นตอน ในการสร้างเมทริกซ์พาริตีเช็กย่อย ($\tilde{H}$) พบว่า มีความถูกต้องแม่นยำสูง (เกิดจากค่าระยะห่างต่ำสุดมีค่าสูง)เมทริกซ์พาริตีเช็กย่อย πA ขนาดเหมาะสมคือประมาณ 5 มีการออกแบบได้เมทริกซ์พาริตีเช็ก 2 แบบที่แตกต่างกันเพื่อนำมาเปรียบเทียบกับมาตรฐาน IEEE 802.16 โดยได้ผลคือ มีค่าระดับความผิดพลาดต่ำกว่าของมาตรฐาน IEEE 802.16 ที่ค่าอัตราส่วนของสัญญาณต่อสัญญาณรบกวนสูงๆ จะมีค่าผลต่างของอัตราส่วนของพลังงานสัญญาณต่อสัญญาณรบกวนระหว่างรหัสที่ยังไม่ผ่านการเข้ารหัสกับรหัสที่ผ่านการเข้ารหัส(Coding Gains) 0.2 dB มีการเข้ารหัสที่ง่ายกว่า แต่อัลกอริทึมที่ออกแบบมีความซับซ้อนมาก [14] รหัสแอลดี-

พีซี สำหรับช่องบันทึกสัญญาณแม่เหล็ก (LDPC Codes for Magnetic Recording Channel) สมรรถนะของอัตรารหัส 8/9 รหัสแอลดีพีซีบนหน่วยความจำที่มีจำกัดและบนช่องสัญญาณเกาส์สีขาว มีความยาวบล็อก N=4,656 อัตราส่วนของพลังงานสัญญาณต่อสัญญาณรบกวน $(E_b/N_o)=E_b/\sigma^2$ สมรรถนะของรหัสแอลดีพีซีมีการเปรียบเทียบกันที่จำนวนรอบการวนซ้ำเท่ากับ 1, 5 และ 10 ที่อัตราส่วนของพลังงานสัญญาณต่อสัญญาณรบกวนมากกว่า 6 dB มีค่าอัตราความผิดพลาดของข้อมูลเท่ากับ 10^{-6}เมื่อเปรียบเทียบกับรหัสเทอร์โบ ที่อัตรารหัสเท่ากันรหัสแอลดีพีซีมีสมรรถนะที่ดีกว่า คือที่อัตรารหัส 8/9 จะให้เกณฑ์ค่าอัตราส่วนของพลังงานสัญญาณต่อสัญญาณรบกวนเท่ากับ 5.3 dB ค่าอัตราความผิดพลาดของข้อมูลประมาณ 10^{-5} มากกว่าระบบที่ไม่เข้ารหัส แต่ใช้วิธีการวีเทอร์บิ 0.2 dB ซึ่งน้อยกว่าเกนอัตราส่วนของพลังงานสัญญาณต่อสัญญาณรบกวนรหัสเทอร์โบด้วยอัตรารหัสเท่ากัน จากบทความวิจัย [8-10] แสดงให้เห็นว่าความสำคัญในการออกแบบรหัสแอลดีพีซีแบบไม่คงที่จะให้สมรรถนะดีกว่ารหัสแอลดีพีซีแบบคงที่ และรหัสเทอร์โบเพราะจะให้ค่าผลต่างของอัตราส่วนของพลังงานสัญญาณต่อสัญญาณรบกวนระหว่างรหัสที่ยังไม่ผ่านการเข้ารหัสกับรหัสที่ผ่านการเข้ารหัสที่สูงกว่าบนช่องบันทึกสัญญาณแม่เหล็ก [15] เสนอการสร้างกลุ่มของรหัสแอลดีพีซีบน ring-mixed alphabet ของจำนวนเต็มซึ่งจะทำให้รหัสแอลดีพีซีถูกขยายบน Z สัญลักษณ์ที่ถูกเพิ่มขึ้นจะใช้ตัวอักษรที่ใหญ่กว่า ในขณะเดียวกันยังแก้ปัญหาผลกระทบที่เกิดจากตัวหารเป็นศูนย์ด้วย และสร้างอัลกอริทึมสำหรับการเพิ่มโหนดเช็กที่เป็นบิตที่เพิ่มเข้ามาในกราฟแทนเนอร์รวมถึงสรุปผลว่าการทำงานของรหัสแอลดีพีซีจะทำงานได้ดีเมื่อมีการทำงานแบบวนซ้ำ [16] เสนอวิธีการลดความซับซ้อนของรหัสแอลดีพีซี ด้วย QC-LDPC Codes ได้พัฒนาเกี่ยวกับเมทริกซ์ย่อยแบบหมุนวน (CirculantSub-Matrices) ที่อัตราส่วนของพลังงานสัญญาณต่อสัญญาณรบกวนสูง ให้สมรรถนะประมาณ 1 dB ที่ 0.0003 มีอัตราแก้ไขข้อผิดพลาดเท่ากับ 10^{-4} รหัสที่นำเสนอได้ทดสอบใน 4 รูปแบบของอัลกอริทึมการถอดรหัส ได้แก่ 1) Bit Flipping (BF) Decoding Algorithm 2) Weighted Bit Flipping (WBF) 3) Implementation-Efficient Reliability Ratio Based Weighted Bit Flipping (IRRWBF) และ4) Standard Belief Propagation Algorithm ซึ่งวิธีที่ 4 ให้สมรรถนะดีกว่าวิธีอื่น มีการแสดงให้เห็นว่าการทดสอบค่าHorizontal Concatenation เป็น 2, 3 และ 6 ปรากฏว่า 2 ให้สมรรถนะที่ดีกว่า

รหัสแอลดีพีซีแบบควอไซไซคลิก (QC-LDPC Codes) ได้สร้างจากเมทริกซ์สลับที่สามารถให้ระยะห่างของเกริธ(Girth)เท่ากับ 12 และโครงสร้างในลักษณะนี้มีผลอีกอย่าง คือทำให้ระยะห่างต่ำสุด (Minimum Distant) มีค่าสูงซึ่งเป็นผลดีต่อการเข้ารหัส [17] การเข้ารหัสของรหัสแอลดีพีซีแบบควอไซไซคลิกได้รับการนำเสนอการออกแบบการเข้ารหัสแบบ VLSI Implementation โดยพิจารณาถึงการลดความซับซ้อนในการเข้ารหัสจึงได้นำเสนอ Pivotingและ Bit-Reverse อัลกอริทึม [18] LDPC Codes ได้รับการออกแบบที่ความยาวบล็อก 648,1,296 และ 1,944 บิตที่อัตรารหัส

1/2, 2/3, 3/4 และ 5/6 ตามมาตรฐาน IEEE 802.11 n และได้มีการออกแบบโครงสร้างการเข้า-ถอดรหัสเชิงฮาร์ดแวร์ [19]

จากผลการวิจัยที่ได้กล่าวมานั้น เป็นแนวทางทำให้เกิดแนวคิดในการทำงานวิจัยด้านการออกแบบเมทริกซ์พาริตีเช็กสำหรับรหัสแอลดีพีซีโดยพิจารณาที่รูปแบบยังสามารถพัฒนาได้อีก รวมถึงวิธีการสร้างเมทริกซ์พาริตีเช็กยังสามารถลดความซับซ้อนในการสร้างและการเพิ่มสมรรถนะของรหัสให้ได้ผลดีเหมือน หรือดีกว่าการออกแบบด้วยวิธีการสุ่ม หรือให้เข้าใกล้ขีดจำกัดของแชนนอลตามทฤษฎีข่าวสารของแชนนอลเพื่อใช้กับความยาวบล็อกขนาดสั้น แต่ให้อัตรารหัสสูงซึ่งงานวิจัยนี้ได้ออกแบบเมทริกซ์พาริตีเช็กแบบใหม่โดยการนำทฤษฎีทางด้านคณิตศาสตร์พื้นฐานมาประยุกต์ ทฤษฎีนั้นคือเมจิกสแควร์หรือจัตุรัสมหัศจรรย์มาใช้ออกแบบวิธีการสร้างเมทริกซ์พาริตีเช็ก (ยังไม่มีใครนำมาใช้สร้างเมทริกซ์พาริตีเช็ก) เพื่อลดขั้นตอนการคำนวณลง แต่ให้ผลของรหัสมีสมรรถนะที่ดีทั้งการสร้างที่มีความซับซ้อนลดลง การเข้ารหัสง่าย สามารถใช้การถอดรหัสแบบวนซ้ำได้และจำนวนรอบการวนซ้ำน้อยลง ได้อัตรารหัสที่สูงบนขนาดความยาวบล็อกสั้น ที่มีสมรรถนะดีกว่าแบบวิธีการสุ่มหรือเปรียบเทียบกับรหัสที่มีคุณลักษณะในกลุ่มเดียวกัน

1.5 ขอบเขตการวิจัย

ในการวิจัยครั้งนี้จะทำการศึกษาเกี่ยวกับการออกแบบเมทริกซ์พาริตีเช็กเพื่อสร้างเมทริกซ์พาริตีเช็กไปใช้เข้า-ถอดรหัสด้วยรหัสแอลดีพีซีแบบไม่คงที่ เน้นที่มีความยาวบล็อกสั้นภายใต้ระบบสัญญาณรบกวนแบบเกาส์สีขาวออกแบบด้วยวิธีแบบโครงสร้าง โดยการทดสอบแบบจำลองด้วยโปรแกรมในระบบคอมพิวเตอร์ เพื่อเปรียบเทียบผลงานวิจัยนี้กับงานอื่นที่มีเสนอในปัจจุบันอยู่ในกลุ่มและประเภทเดียวกันและเปรียบเทียบสมรรถนะกับรหัสแอลดีพีซีที่ออกแบบด้วยวิธีการสุ่มโดยวิธีการในการออกแบบของงานวิจัยนี้อยู่ภายใต้การประยุกต์ใช้ทฤษฎีทางด้านคณิตศาสตร์พื้นฐานที่เรียกว่าเมจิกสแควร์หรือจัตุรัสมหัศจรรย์ มาออกแบบวิธีการสร้างเมทริกซ์พาริตีเช็กแบบใหม่ โดยวิทยานิพนธ์ฉบับนี้ประกอบด้วยบทต่าง ๆ 6 บท ดังนี้ บทที่ 1 บทนำ กล่าวถึงความเป็นมาและความสำคัญของปัญหาวัตถุประสงค์สมมุติฐานทฤษฎีหรือแนวความคิดที่ใช้ในการวิจัย ขอบเขตและขั้นตอนการวิจัย บทที่ 2 รหัสแอลดีพีซี กล่าวถึงรหัสแก้ไขข้อผิดพลาด วิวัฒนาการและประเภทของรหัสแอลดีพีซี การออกแบบเมทริกซ์พาริตีเช็กการเข้า-ถอดรหัสและการทดสอบสมรรถนะของรหัสแอลดีพีซี บทที่ 3 ทฤษฎีและงานวิจัยเกี่ยวข้อง กล่าวถึงคณิตศาสตร์พื้นฐานและการเข้ารหัสช่องสัญญาณ เมจิกสแควร์สถิติประยุกต์กับงานวิจัยงานวิจัยเกี่ยวกับรหัสแอลดีพีซีด้านการออกแบบเมทริกซ์พาริตีเช็ก งานวิจัยเกี่ยวกับรหัสแอลดีพีซีกลุ่มเลขหนึ่งไม่คงที่กลุ่มขนาดความยาวบล็อกสั้นกลุ่มเข้ารหัสและกลุ่มถอดรหัส บทที่ 4 การออกแบบเมทริกซ์พาริตีเช็กของการวิจัย กล่าวถึงการออกแบบโดยพิจารณารูปแบบเมทริกซ์พาริตีเช็ก เพื่อลดจำนวนเลขหนึ่ง พิจารณากำหนดค่าตัวแปรกับรูปแบบสมมาตร และการออกแบบโดยประยุกต์ทฤษฎีคณิตศาสตร์พื้นฐาน บทที่ 5 ผลการทดสอบสมรรถนะของงานวิจัย

กล่าวถึงระบบช่องสัญญาณรบกวนแบบเกาส์สีขาว ผลการทดสอบลูป 4 ทดสอบการเข้า-ถอดรหัสแอลดีพีซี วิเคราะห์ผลในเชิงสถิติประยุกต์ และผลการเปรียบเทียบในกลุ่มรหัสแอลดีพีซีแบบไม่คงที่เน้นที่กลุ่มขนาดความยาวบล็อกสั้นและบทที่ 6 สรุปและข้อเสนอแนะจากการวิจัยกล่าวถึงการสรุปผลการทดสอบสมรรถนะของงานวิจัยและข้อเสนอแนะจากการวิจัย

1.6 ขั้นตอนการวิจัย

1.6.1 ศึกษาส่วนต่าง ๆ ในรูปแบบบล็อกไดอะแกรม หน้าที่การทำงานและข้อกำหนดทางเทคนิคของระบบการสื่อสารและฮาร์ดดิสก์ไดร์ฟ

1.6.2 ศึกษาเกี่ยวกับรหัสแบบต่าง ๆ ที่ใช้ในการเข้าและถอดรหัสในระบบการสื่อสารและฮาร์ดดิสก์ไดร์ฟ

1.6.3 ศึกษาคณิตสาสตร์พื้นฐานสำหรับทำความเข้าใจเรื่องรหัสแก้ไขข้อผิดพลาด

1.6.4 ศึกษาวิเคราะห์วิธีการเข้าและถอดรหัสแบบต่าง ๆ

1.6.5 ศึกษารหัสแอลดีพีซีที่จะใช้ในการวิจัย

1.6.6 ศึกษางานวิจัยที่เกี่ยวข้อง

1.6.7 ออกแบบเมทริกซ์พาริตีเช็กเพื่อใช้กับรหัสแอลดีพีซี

1.6.8 ทดสอบระบบในส่วนของการเข้า และถอดรหัสด้วยรหัสแอลดีพีซี ตามเมทริกซ์พาริตีเช็กที่ได้ออกแบบภายใต้ระบบสัญญาณรบกวนแบบเกาส์สีขาว

1.6.9 ปรับปรุงแบบและทดสอบตามแบบที่ได้ปรับ

1.6.10 นำเสนอรูปแบบของเมทริกซ์พาริตีเช็กใหม่ที่ได้ออกแบบ

1.6.11 ทดสอบปรับปรุงและพัฒนาให้ได้ผลงานที่มีประสิทธิภาพดีภายใต้อัตรารหัสสูงและเน้นที่ความยาวบล็อกสั้น

1.6.12 เปรียบเทียบสิ่งที่ออกแบบกับงานวิจัยอื่นในกลุ่มเดียวกันหรือใกล้เคียงกัน

1.6.13 แสดงผลงานที่ได้ดำเนินการทั้งหมด

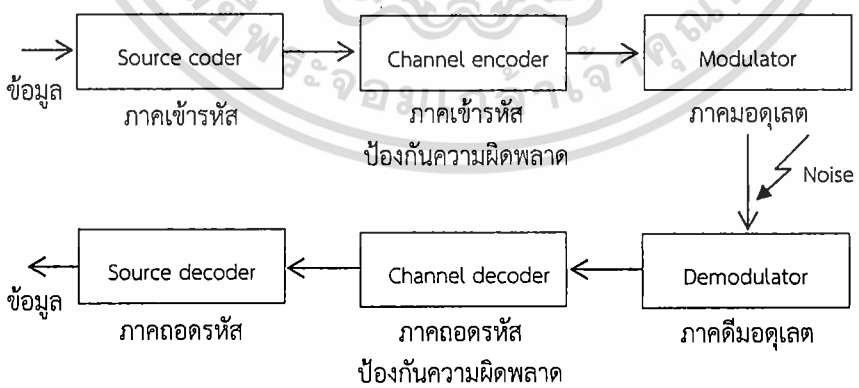
1.6.14 จัดทำรายงานการวิจัยฉบับสมบูรณ์และวิทยานิพนธ์

บทที่ 2
รหัสแอลดีพีซี

บทนี้ จะกล่าวถึงรหัสแก้ไขข้อผิดพลาดได้แก่รหัสบล็อกและรหัสซีเควนเชียล วิวัฒนาการและประเภทของรหัสแอลดีพีซีการออกแบบเมทริกซ์พาริตีเช็กสำหรับรหัสแอลดีพีซีแบบคงที่และแบบไม่คงที่ การเข้ารหัสอย่างง่ายการถอดรหัสแบบวนซ้ำและการทดสอบสมรรถนะของรหัสแอลดีพีซี เกี่ยวกับการปราศจากลูป 4 การหาค่าอัตรารหัสค่าอัตราความผิดพลาดของข้อมูลค่าอัตราส่วนของพลังงานสัญญาณต่อสัญญาณรบกวนค่าระยะห่างต่ำสุดและค่าผลต่างของอัตราส่วนของพลังงานสัญญาณต่อสัญญาณรบกวนระหว่างรหัสที่ยังไม่ผ่านการเข้ารหัสกับรหัสที่ผ่านการเข้ารหัส โดยมีรายละเอียดดังต่อไปนี้

2.1 รหัสแก้ไขข้อผิดพลาด

การเข้ารหัสข้อมูลในระบบสื่อสารเป็นวิธีการที่ใช้ในการลดความผิดพลาดของการส่งข้อมูลผ่านระบบสื่อสารซึ่งทำให้มีการเกิดความผิดพลาดลดลงเมื่อเปรียบเทียบกับกรณีของระบบสื่อสารที่ไม่มีการเข้ารหัสแก้ไขข้อผิดพลาดของข้อมูลสำหรับวิธีการที่ใช้ในการทำงานนั้นเป็นการนำข้อมูลที่เป็นข้อมูลแบบดิจิตอลที่จะทำการส่งผ่านระบบสื่อสารมาทำการเปลี่ยนแปลงรูปแบบของข้อมูลให้อยู่ในอีกลักษณะหนึ่งซึ่งถูกเรียกว่าคำรหัสที่มีคุณสมบัติพิเศษจะสามารถแก้ไขหรือตรวจจับข้อมูลที่เกิดความผิดพลาดระหว่างการส่งข้อมูลผ่านระบบสื่อสารให้กลับมาเป็นข้อมูลที่ถูกต้องได้เมื่อข้อมูลนั้นถูกส่งมาถึงปลายทาง ดังรูปที่ 2.1 แสดงระบบสื่อสารที่มีการเข้า-ถอดรหัสข้อมูล



รูปที่ 2.1 แผนผังระบบสื่อสารที่มีการเข้า-ถอดรหัสข้อมูล

โดยกระบวนการที่สำคัญสองกระบวนการที่ใช้ในการเข้า-ถอดรหัสข้อมูลในระบบสื่อสารได้แก่ กระบวนการเข้ารหัสช่องสัญญาณ (Channel Encoder)ที่ใช้ในการเปลี่ยนแปลงข้อมูลที่จะทำการส่งให้อยู่ในรูปของคำรหัสและกระบวนการถอดรหัสช่องสัญญาณ (Channel Decoder) ที่ใช้สำหรับการนำข้อมูลที่ได้รับปลายทางนั้นมาทำการประมวลผลเพื่อหาข้อมูลที่คาดว่าต้นทางส่งมาซึ่งลักษณะของข้อมูลที่ได้หลังจากกระบวนการเข้ารหัสข้อมูลที่จะถูกส่งผ่านระบบสื่อสารนั้นจะมีขนาดของข้อมูลที่มากกว่าข้อมูลที่ไม่มีการเข้ารหัสข้อมูลโดยรหัสที่ใช้เพื่อเข้า-ถอดรหัสข้อมูลในระบบสื่อสารเรียกว่า รหัสแก้ไขข้อผิดพลาดของข้อมูล (Error Correction Codes : ECC) สามารถแบ่งออกได้เป็น 2 รูปแบบหลักเมื่อพิจารณาจากกระบวนการเข้ารหัส คือรหัสบล็อก (Block Codes) กับรหัสซีเควนเชียล (Sequential Codes) ซึ่งจะมีรายละเอียดดังนี้

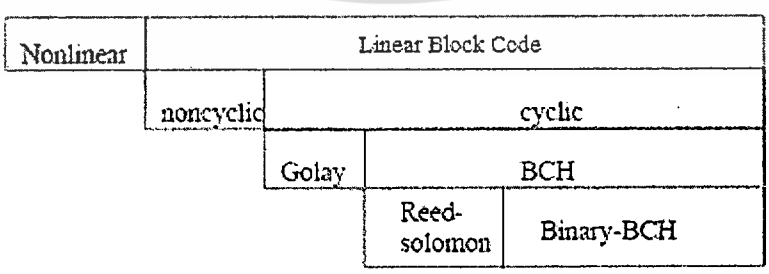
2.1.1 รหัสบล็อก การเข้ารหัสบล็อกนั้นจะเป็นการนำข้อมูลดิจิตอลที่จะทำการเข้ารหัสมาทำการแบ่งออกเป็นชุดหรือบล็อกย่อยๆซึ่งแต่ละบล็อกมีขนาดเท่ากับ k บิตจะถูกนำมาประมวลผลเพื่อหาค่าของคำรหัส (Codeword) ที่มีความยาวเท่ากับ n บิต โดยที่ n>k ดังนั้นจึงมักเรียกการเข้ารหัสนี้ว่า (n,k) โดยปกติชุดรหัสที่ได้จากการเข้ารหัสจะยังคงประกอบด้วยส่วนของบิตข้อมูลเดิม k บิตและส่วนของบิตพิเศษที่เพิ่มเข้าไปอีก n-k บิตหรือเรียกว่าบิตเช็ก (Check Bit) เพื่อใช้สำหรับตรวจสอบว่ามีความผิดพลาดในบิตข้อมูลในระหว่างที่ส่งผ่านช่องสัญญาณหรือไม่ ณ ที่ภาครับก็จะมีวงจรที่ทำหน้าที่ถอดรหัสเพื่อดึงบิตข้อมูลต้นฉบับเดิมกลับคืนออกมา โดยทั่วไปในกระบวนการถอดรหัสจะมีการคำนวณค่าที่เรียกว่า ซินโดรม (Syndrome) ซึ่งค่าซินโดรมนี้ใช้สำหรับบ่งบอกว่ามีความผิดพลาดเกิดขึ้นในบิตข้อมูลหรือไม่ หรืออาจใช้ในการบ่งบอกถึงตำแหน่งของบิตที่ผิดด้วยซึ่งจะมีการทำงานเช่นนี้ตั้งแต่ข้อมูลบล็อกแรกจนถึงบล็อกสุดท้ายจึงจะสิ้นสุดการเข้ารหัสตัวอย่างรหัสบล็อก ได้แก่ รหัสพาริตีเช็ก (Parity Check Codes) รหัสวน (Cyclic Codes) รหัสรีดโซโลมอน(Reed-Solomon codes)และรหัสแอลดีพีซี (LDPC Codes) เป็นต้น

2.1.2 รหัสซีเควนเชียล แตกต่างจากรหัสแบบบล็อกตรงที่ ข้อมูลที่จะเข้ารหัสไม่จำเป็นต้องนำมาแบ่งออกเป็นบล็อกที่มีขนาดความยาวตายตัวก่อนจะนำไปผ่านกระบวนการเข้า เราสามารถป้อนข้อมูลเข้าสู่วงจรเข้ารหัสซีเควนเชียลได้อย่างต่อเนื่อง และกระบวนการเข้ารหัสจะดำเนินต่อไปจนกว่าจะหยุดการป้อนข้อมูลเข้าไป ฉะนั้นจุดแตกต่างที่สำคัญคือรหัสซีเควนเชียลจะไม่นิยามชุดรหัสในรูปของ (n,k) เนื่องจากไม่มีการระบุขอบเขตความยาวของข้อมูลที่จะทำการเข้ารหัสที่แน่นอน การนิยามคุณสมบัติของรหัสซีเควนเชียลจึงแสดงในรูปของอัตราส่วนการเข้ารหัส เช่น 1/n แทน กล่าวคือเมื่อเราป้อนข้อมูล 1 บิตเข้าสู่วงจรเข้ารหัส จะได้เป็นรหัสที่มีความยาวเพิ่มขึ้น n เท่า อัตราส่วนการเข้ารหัสอาจมีค่าที่ต่างไปจากนี้ได้ เช่น การป้อนข้อมูล 2 บิตและให้ผลเป็นรหัสที่มีความยาว 3 บิต อัตราส่วนการเข้ารหัสในกรณีนี้

มีค่าเท่ากับ 2/3 ตัวอย่างรหัสซีเควนเชียลที่สำคัญ ได้แก่ รหัสคอนโวลูชัน (Convolution Codes)รหัสเทอร์โบ (Turbo Codes) เป็นต้น

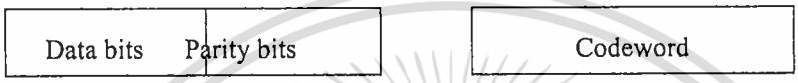
ในการออกแบบหรือใช้งานระบบสื่อสารแบบดิจิตอลนั้นจะต้องมีการพิจารณาถึงองค์ประกอบในหลายๆส่วนด้วยกันโดยสิ่งหนึ่งที่ต้องพิจารณาคือข้อมูลดิบที่ถูกส่งจากต้นทางไปถึงปลายทางนั้นมีข้อมูลที่เกิดความผิดพลาดขึ้นหรือไม่ที่เกิดจากสาเหตุต่าง ๆ หลายสาเหตุด้วยกันโดยสาเหตุหลักจะทำให้เกิดความผิดพลาดดังกล่าว คือ การที่ระบบสื่อสารนั้นถูกรบกวนจากสัญญาณรบกวนต่าง ๆ หากขนาดสัญญาณรบกวนที่เกิดขึ้นในระบบสื่อสารมีค่าสูงจะส่งผลให้อัตราการเกิดความผิดพลาดของข้อมูลมีค่าสูงตามไปด้วย ในการลดอัตราการเกิดความผิดพลาดของข้อมูลให้มีค่าที่ลดลงนั้นสามารถทำได้หลายรูปแบบด้วยกัน เช่น เพิ่มกำลังของเครื่องส่งทำให้ขนาดของสัญญาณรบกวนมีค่าน้อยลง การเข้ารหัสข้อมูลเป็นต้น สำหรับการลดอัตราการเกิดความผิดพลาดที่เกิดขึ้นในการส่งข้อมูลด้วยวิธีการเข้ารหัสนั้น จะเป็นการนำข้อมูลดิบที่จะทำการส่งผ่านระบบสื่อสารที่เป็นข้อมูลแบบดิจิตอลมาทำการผ่านกระบวนการ “เข้ารหัส” หรือ “Encoding” เพื่อเปลี่ยนรูปแบบของข้อมูลที่จะถูกส่งผ่านระบบสื่อสารให้อยู่ในรูปแบบที่สามารถนำข้อมูลมาทำการแก้ไขข้อผิดพลาดของข้อมูลที่เกิดขึ้นเนื่องจากการถูกรบกวนจากสัญญาณรบกวนต่างๆได้ โดยที่ข้อมูลที่ได้จากการทำงานนั้น จะเป็นข้อมูลที่จะถูกส่งออกไปผ่านระบบสื่อสารและเมื่อข้อมูลดังกล่าวถูกส่งมาถึงปลายทางจะมีการนำข้อมูลที่รับมาผ่านกระบวนการ “ถอดรหัส” หรือ“Decoding” เพื่อเปลี่ยนรูปแบบของข้อมูลที่รับได้ให้กลับมาอยู่ในรูปของข้อมูลดิบพร้อมทั้งทำการแก้ไขข้อมูลที่คาดว่าจะเกิดความผิดพลาดขึ้นให้มีค่าที่ถูกต้อง

การเข้ารหัสแบบบล็อกเป็นรูปแบบหนึ่งของการเข้ารหัสที่ใช้ในการส่งข้อมูลผ่านระบบสื่อสารซึ่งวิธีการที่ใช้ในการเข้ารหัสข้อมูลเพื่อเปลี่ยนแปลงข้อมูลดิบให้กลายเป็นคำรหัสที่จะถูกส่งผ่านระบบสื่อสารจากการคำนวณหาค่าซินโดรมไปใช้ตรวจจับหรือแก้ไขข้อมูลที่ผิดพลาดให้กลับมาถูกต้องซึ่งรูปแบบของการเข้ารหัสแบบบล็อกสามารถแบ่งรูปแบบที่ใช้ในการทำงานได้หลายรูปแบบด้วยกันสามารถแบ่งเป็นกลุ่มใหญ่ ๆ ได้ดังรูปที่ 2.2



รูปที่ 2.2 กลุ่มของรหัสแบบบล็อก

นอกจากการแบ่งรูปแบบของการเข้ารหัสออกเป็นรูปแบบต่าง ๆ ดังแสดงในรูปที่ 2.2 แล้วยังมีการพิจารณาถึงลักษณะของรูปแบบในการเข้ารหัสตามลักษณะของข้อมูลที่ได้หลังจากการเข้ารหัสซึ่งจะสามารถแบ่งออกได้เป็น 2 รูปแบบด้วยกันได้แก่รหัสระบบ (Systematic Codes)และรหัสไม่เป็นระบบ (Non-Systematic Codes)โดยรหัสระบบนั้นจะเป็นรูปแบบของการเข้ารหัสข้อมูลที่มีผลลัพธ์จากการเข้ารหัสจะยังคงมีข้อมูลส่วนหนึ่งที่ยังคงมีลักษณะเหมือนข้อมูลดิบอยู่ ส่วนรหัสไม่เป็นระบบนั้นข้อมูลของคำรหัสที่ได้จากการเข้ารหัสจะไม่มีลักษณะที่เหมือนกับข้อมูลดิบมีลักษณะของข้อมูลดังรูปที่2.3



ก) ข)

รูปที่ 2.3 รูปแบบของรหัส ก) รหัสระบบ ข) รหัสไม่เป็นระบบ

2.2 วิวัฒนาการและประเภทของรหัสแอลดีพีซี

2.2.1 วิวัฒนาการของรหัสแอลดีพีซี ระบบสื่อสารสิ่งที่หลีกเลี่ยงไม่ได้คือสัญญาณรบกวนที่มีอยู่ในระบบการสื่อสารทุกชนิดและถ้าสัญญาณรบกวนมีค่าสูงขึ้นจะส่งผลกระทบถึงอัตราความผิดพลาดของข้อมูลที่จะเกิดขึ้นสูงตามไปด้วยซึ่งเป็นสิ่งที่ไม่ดีทำให้การแก้ไขข้อผิดพลาดมีความจำเป็นมากและการแก้ไขข้อผิดพลาดของข้อมูลนั้นสามารถทำได้หลายวิธีเช่นการเพิ่มกำลังเครื่องส่งข้อมูลหรือการเข้า-ถอดรหัสข้อมูลสำหรับการเข้า-ถอดรหัสข้อมูลนั้นเริ่มตั้งแต่ยุคแรก ๆ เป็นการเข้า-ถอดรหัสแบบบล็อกได้แก่ รหัสหมุนและรหัสรีดโซโลมอนจากนั้นพัฒนาเป็นรหัสคอนโวลูชันซึ่งสามารถรับส่งข้อมูล (Throughput) สูงขึ้นต่อมาเป็นรหัสเทอร์โบซึ่งเป็นรหัสที่สามารถแก้ไขข้อผิดพลาดปริมาณสูงยิ่งขึ้นและถูกนำไปใช้อย่างแพร่หลายในยุคที่สาม (3G) และต่อมาก็เป็นรหัสแอลดีพีซี ซึ่งรหัสแอลดีพีซีได้รับการคิดค้นขึ้นมาโดย Robert Gallager แห่งสถาบัน MIT ในปี 1960 [3] แต่ในขณะนั้นไม่ได้รับความสนใจและเงียบหายไปนานจนกระทั่งในปี 1996 David MacKay ได้นำรหัสแอลดีพีซี [2] กลับมาใช้ใหม่โดยได้รับการค้นคว้าวิจัยและพัฒนาให้ดีขึ้นและเนื่องจากรหัสแอลดีพีซีมีสมรรถนะในการแก้ไขข้อผิดพลาดได้ดีทำให้นักวิจัยสนใจมากโดยการวิจัยมุ่งเน้นไปที่การวิเคราะห์และปรับปรุงรหัสให้ดีขึ้นและรหัสแอลดีพีซีได้รับการนำเสนอไปใช้ร่วมกับระบบต่าง ๆ มากมายในปัจจุบันเช่นนำไปใช้ร่วมกับระบบการสื่อสารไร้สายระบบการสื่อสารดาวเทียมและระบบฮาร์ดดิสก์ไดร์ฟนอกจากนี้รหัสแอลดีพีซียังได้รับการนำเสนอให้ใช้ในยุคที่สี่ (4G) [4] และที่สำคัญรหัสแอลดีพีซีไม่มีสิทธิบัตรคุ้มครอง สามารถพัฒนาและผลิตโดยไม่เสียค่าใช้สิทธิบัตรแต่อย่างใด รหัสแก้ไขข้อผิดพลาดทำหน้าที่ลดจำนวนบิตผิดพลาดของข้อมูลในระบบสื่อสารแบบดิจิตอลโดยยังคงระดับอัตราส่วนของพลังงานสัญญาณต่อสัญญาณรบกวนในระดับที่เหมาะสมหรือระดับ

อัตราบิตผิดพลาดเท่ากัน รหัสแอลดีพีซีให้เกนของอัตราส่วนของพลังงานสัญญาณต่อสัญญาณรบกวนเข้าใกล้ขีดจำกัดของแชนนอล

รหัสแอลดีพีซีหรือชื่อในภาษาอังกฤษว่า Low-Density Parity-Check (LDPC) Codes คือ รหัสที่มีจำนวนของเลขหนึ่งน้อยเมื่อเทียบกับขนาดของเมทริกซ์พาริตีเช็ก ทั้งนี้ก็เพื่อให้มีระยะห่างต่ำสุดของรหัสสูง ได้ถูกสร้างขึ้นครั้งแรกในปีค.ศ.1960 (2503) ในวิทยานิพนธ์ระดับปริญญาเอกของ R.Gallagerที่ Massachusetts Institute of Technology : MIT [3] ประเทศสหรัฐอเมริกาโดยจัดเป็นรหัสช่องสัญญาณแบบบล็อกเชิงเส้นชนิดหนึ่งที่การเข้ารหัสจะดำเนินการผ่านเมทริกซ์พาริตีเช็กมีโครงสร้างแบบสุ่มแต่รหัสแอลดีพีซียังไม่เป็นที่สนใจมากนักในขณะนั้นในปี ค.ศ.1981 (2524) R.M.Tanner [5] ได้นำเสนอการใช้กราฟแสดงความสัมพันธ์ที่เกิดจากการเข้ารหัสที่ชื่อว่า กราฟแทนเนอร์ (Tanner Graph หรือ Bipartite Graph) ซึ่งทำให้สามารถนำภาพที่ได้มาช่วยในการออกแบบการถอดรหัสได้ง่ายขึ้น จนกระทั่งในปี ค.ศ.1990 (2533) D.J.C.Mackay [2] ได้แสดงผลงานวิจัยที่พบว่ารหัสแอลดีพีซีมีสมรรถนะการทำงานที่เข้าใกล้ขีดจำกัดของแชนนอลได้เช่นเดียวกับรหัสเทอร์โบ หลังจากนั้นรหัสแอลดีพีซีจึงได้รับความสนใจอย่างแพร่หลายจากรหัสอาร์เรย์ (Array Codes) เป็นรหัสแก้ไขข้อผิดพลาดที่มีความสามารถในการแก้ไขข้อผิดพลาดแบบหลายบิตติดกัน (Bursts Error โดย [20] ได้กล่าวว่ารูปแบบการถอดรหัสวนซ้ำแบบซอฟต์ สามารถนำไปประยุกต์ใช้กับรหัสอาร์เรย์ได้ [6] ได้เสนอการสร้างเมทริกซ์พาริตีเช็กแบบโครงสร้าง และได้รหัสที่รู้จักกันในนามของรหัสแอลดีพีซีแบบอาร์เรย์ (Array LDPC Codes) เมื่อการสร้างเมทริกซ์พาริตีเช็กมีความเป็นโครงสร้างมากขึ้น เมื่อเปรียบเทียบกับรหัสแบบสุ่มจึงทำให้รหัสแอลดีพีซีแบบอาร์เรย์ สามารถนำไปประยุกต์ใช้กับอุปกรณ์การเข้ารหัส-ถอดรหัสอย่างง่ายได้ ต่อมา [17] ได้ทำการศึกษาการสร้างเมทริกซ์พาริตีเช็กสำหรับรหัสแอลดีพีซี โดยการใช้เมทริกซ์สับเปลี่ยนหมุนวน (Circulant Permutation Matrices) และชี้ให้เห็นว่ารหัสที่ได้ที่มีเกริธใหญ่กว่า 12 จะไม่สามารถแทนได้ด้วยกราฟแทนเนอร์ เขาจึงได้เสนอเงื่อนไขที่ง่ายและจำเป็นที่เพียงพอสำหรับกราฟแทนเนอร์ของ Quasi-Cyclic (QC) LDPC Codes ในเฉพาะบางกรณี คือ ที่เกริธ (g) = 6, 8, 10 และ 12 เงื่อนไขดังกล่าวทำให้การสร้าง QC-LDPC Codes ง่ายขึ้นปานกลาง ส่วนสมรรถนะดีขึ้นมากอย่างเห็นได้ชัดเมื่อเปรียบเทียบกับ [3] ด้วยวิธีการถอดแบบวนซ้ำ Belief Propagation Algorithm : BPAต่อมา [21] เสนอวิธีการสร้างสำหรับ QC-LDPC Codes ที่มีความยาวบล็อกยาว โดยการรวม QC-LDPC Codes ที่มีความยาวบล็อกสั้นหลาย ๆ รหัสเข้าด้วยกันผ่านทฤษฎีเศษเหลือของจีน(Chinese Remainder Theorem : CRT) เมื่อจำนวนเฉพาะ(Prime Numbers) ถูกนำไปใช้เป็นตัวแปรในการสร้างรหัส และพบว่าเกริธของรหัส QC-LDPC Codes ที่ได้จะใหญ่กว่าหรือเท่ากับเกริธของรหัสองค์ประกอบแต่ละรหัส โดยสมรรถนะของรหัสที่สร้างขึ้นด้วยวิธีนี้ใกล้เคียงกับรหัสแบบสุ่ม ขณะที่การศึกษาส่วนใหญ่ก่อนหน้านี้ [2-6] ได้มุ่งเน้นเกี่ยวกับรหัสแอลดีพีซีแบบคงที่ [22] ได้พัฒนาสมรรถนะของรหัส โดยนำเสนอเกี่ยวกับเมทริกซ์แบบไม่คงที่ โดย [10] เสนอรหัสแอลดีพีซีแบบอาร์เรย์แบบไม่คงที่สำหรับนำไปประยุกต์ใช้ใน

ระบบ Digital Subscriber Lines : DSL โดยการสร้างรหัสอยู่บนพื้นฐานของรหัสแอลดีพีซีอาร์เรย์ที่เสนอโดย [6] รูปแบบของเมทริกซ์พาริตีเช็กที่เป็นรูปสามเหลี่ยมเป็นสิ่งที่พึงปรารถนาเพื่อให้การถอดรหัสมีประสิทธิภาพสามารถนำไปใช้ได้จริง อย่างไรก็ตามรหัสที่ถูกออกแบบนั้นไม่สามารถนำไปใช้ได้กับทุกขนาดความยาวที่ต้องการ (Arbitrary Lengths) เมื่อความยาวบล็อกถูกจำกัดด้วยจำนวนเฉพาะ [23] มีการศึกษาเกี่ยวกับระบบในช่องสัญญาณการบันทึกข้อมูลแบบแม่เหล็ก (Magnetic Recording Channels) ที่เป็นช่องทางเพื่อให้สามารถนำรหัสแอลดีพีซีมาประยุกต์พัฒนาต่อไปได้ ส่วน [24] ได้เสนอรหัสแอลดีพีซีอาร์เรย์ที่เข้ากันได้กับขนาดที่หลากหลาย(Size Compatible (SC) Array LDPC Codes) โดยพวกเขาได้เสนอการเลื่อนวน (Cyclic Shift) แบบใหม่ แทนที่เมทริกซ์สับเปลี่ยนแบบเดิม เพื่อที่จะกำจัดลูป 4 พวกเขาได้สับเปลี่ยนตำแหน่งของเมทริกซ์ย่อย (Sub-Matrix) บนพื้นฐานของจำนวนแถว ต่อมา [25] เสนอโครงสร้างรหัสแบบไม่คงที่ที่ใช้ได้กับความยาวที่หลากหลายมากขึ้น ซึ่งคล้ายกับ [21] โดยการกำหนดสูตรของพวกเขายังอยู่บนพื้นฐานของทฤษฎีเศษเหลือของจีนที่เสนอโดย [21] แต่ตัวแปรที่ใช้ไม่ได้จำกัดที่จำนวนจำเพาะ (Non-Prime Number) ซึ่งถ้าไม่สนใจเกี่ยวกับความซับซ้อนในการสร้างรหัสจะเห็นได้ว่ารหัสที่ได้มีความยืดหยุ่นของขนาดบล็อกในการออกแบบมากขึ้น สมรรถนะดีที่รหัสความยาวบล็อกยาว

2.2.2 **ประเภทของรหัสแอลดีพีซี** สามารถแบ่งเป็นสองประเภทหลักคือรหัสแอลดีพีซีที่มีการกระจายตัวของเลขหนึ่งในเมทริกซ์พาริตีเช็กเป็นแบบคงที่ (Regular LDPC Codes) ซึ่งเป็นรหัสที่มีรูปแบบเดียวกับรหัสของ R.Gallager และรหัสแอลดีพีซีอีกชนิดหนึ่งที่มีการกระจายตัวของเลขหนึ่งเป็นแบบไม่คงที่ (Irregular LDPC Codes) ซึ่งเป็นรหัสที่พัฒนามาจากรหัสแอลดีพีซีแบบคงที่ การเข้า-ถอดรหัสเริ่มต้นจากการสร้างเมทริกซ์พาริตีเช็กนอกจากนั้นยังสามารถแบ่งได้เป็นรหัสแอลดีพีซีแบบเชิงระบบ (Systematic) คือการที่คำรหัสมีการแบ่งส่วนของบิตข้อมูลกับบิตของพาริตีชัดเจน กับรหัสแอลดีพีซีไม่เป็นระบบ (Non-Systematic) คือการที่คำรหัสไม่สามารถบอกได้ว่าบิตใดเป็นบิตข้อมูลและบิตใดเป็นบิตของพาริตีได้ชัดเจนและถ้าพิจารณาจากโครงสร้างของเมทริกซ์พาริตีเช็กก็สามารถแบ่งได้เป็นสองประเภทคือแบบโครงสร้างหมายถึงเมทริกซ์พาริตีเช็กที่สามารถกำหนดโครงสร้างได้กับแบบวิธีการสุ่มรูปแบบของเมทริกซ์พาริตีไม่มีกำหนดโครงสร้างรูปแบบที่แน่นอนได้

2.3 การออกแบบเมทริกซ์พาริตีเช็กสำหรับรหัสแอลดีพีซี

2.3.1 รหัสแอลดีพีซีแบบคงที่ มีการกระจายตัวของเลขหนึ่งเป็นแบบคงที่นั้นมีที่มาจากการที่จำนวนเลขหนึ่งในแต่ละแถวหรือแต่ละหลักของเมทริกซ์พาริตีเป็นค่าคงที่มีการกระจายตัวของเลขหนึ่งเป็นแบบคงที่ C (n, W_r, W_c) โดยเมทริกซ์พาริตีเช็กขนาด mxn

เมื่อ $H_{m\times n}$ ที่สร้างขึ้นจากการสุ่มข้อมูลศูนย์หนึ่ง

W_c คือ จำนวนเลขหนึ่งในหลักของเมทริกซ์พาริตีเช็กโดยที่$W_c<<m$

W_r คือ จำนวนเลขหนึ่งในแถวของเมทริกซ์พาริตีเช็กและ$W_r= W_c(n/k)$, $W_r<<n$

H ที่สร้างขึ้นจะต้องปราศจากลูป 4

อัตรารหัส (R) = $1 - (W_c/W_r)$

ค่าความหนาแน่นของเลขหนึ่ง $\rho =(W_r/n) =(W_c/m)$ ทำให้ $m =(W_c/W_r)\times n$ และ$\lim_{n\to\infty}\rho = 0$

รหัสนี้มีความยาวข้อมูลอินพุตเท่ากับ n-m บิตความยาวคำรหัสเท่ากับ n และจำนวนพาริตีบิตเท่ากับmบิต

พิจารณารหัสแอลดีพีซีขนาด (10, 5) ซึ่งมีค่า $W_c= 2$ และ $W_r=W_c(n/k) = 2\times(10/5) = 4$ เมทริกซ์พาริตีเช็กที่ได้ดังสมการที่ (2.1)

$$H=\begin{bmatrix}1&1&1&1&0&0&0&0&0&0\\1&0&0&0&1&1&1&0&0&0\\0&1&0&0&1&0&0&1&1&0\\0&0&1&0&0&1&0&1&0&1\\0&0&0&1&0&0&1&0&1&1\end{bmatrix}_{(m=5,n=10)} \quad (2.1)$$

ตัวอย่างเมทริกซ์พาริตีเช็กที่ได้นั้นสอดคล้องกับเงื่อนไขทั้ง 4 ข้อกล่าวคือ$W_c = 2$; $W_r= 4$ รวมถึงปราศจากลูป 4

2.3.2 รหัสแอลดีพีซีแบบไม่คงที่ มีที่มาจากจำนวนเลขหนึ่งในแต่ละแถวหรือแต่ละหลักของเมทริกซ์พาริตีเช็กมีจำนวนไม่คงที่ในทุกแถว หรือทุกหลักของเมทริกซ์พาริตีเช็กถูกพัฒนาขึ้นมาในปี ค.ศ. 2001 (2544) โดย T.Richardson [7] และด้วยเหตุที่การกระจายตัวของเลขหนึ่งเป็นแบบไม่คงที่ค่าความหนาแน่นของเลขหนึ่ง (ρ) ที่นิยามไว้ดังนี้

รหัสแอลดีพีซีแบบไม่คงที่คือรหัสแอลดีพีซีที่การกระจายตัวเลขหนึ่งไม่คงที่

1) ค่าความหนาแน่นของเลขหนึ่งของแต่ละแถวนิยามโดย $[\rho_2,\rho_3,\ldots,\rho_n]$

2) ค่าความหนาแน่นของเลขหนึ่งของแต่ละหลักนิยามโดย $[\lambda_2, \lambda_3,\ldots,\lambda_n]$

3) อัตรารหัส $R = 1 - \left[\left(\sum_{j=2}^{n}\rho_j/j\right)/\left(\sum_{j=2}^{n}\lambda_j/j\right)\right]$

ด้วยวิธีการของ T.Richardson นั้นสมรรถนะการทำงานของรหัสแอลดีพีซีแบบไม่คงที่ดีกว่ารหัสแอลดีพีซีแบบคงที่ของ D.J.C. MacKay แต่ก็มีหลายงานวิจัยพบว่ารหัสแอลดีพีซีแบบไม่คงที่ในแบบของ T.Richardson นั้นจะทำงานได้ดีเมื่ออัตรารหัส $R\leq 3/4$ และที่ขนาดความยาวบล็อกคำรหัส $n\geq 5{,}000$

2.4 การเข้าและถอดรหัสแอลดีพีซี

2.4.1 การเข้ารหัสอย่างง่าย กระทำหลังจากสร้างเมทริกซ์พาริตีเช็ก H แล้วอาศัยความสัมพันธ์ในการสร้างคำรหัสดังนี้

$$C_{(1\times n)}H^{T}_{(n\times m)} = 0_{(1\times m)} \tag{2.2}$$

เมื่อ $C = [c_1\ c_1\ c_2c_2\ \dots\ c_n\ ]_{(1\times n)}$ เป็นเมทริกซ์คำรหัสขนาด (1×n)

$0_{(1\times m)}$ เป็นเมทริกซ์ศูนย์ขนาด (1×m)

ในกรณีรหัสเชิงระบบคำรหัสC เขียนได้ในรูป

$$C = [p_1\ p_2\ p_3\ \dots\ p_{n-k}\ m_1\ m_2\ m_3\ \dots\ m_k]_{(1\times n)} \tag{2.3}$$

หรือ

$$C = [p_{(1\times n-k)} \mid m_{(1\times k)}]_{(1\times n)} \tag{2.4}$$

โดย $p_{(1\times n-k)}$ คือ เมทริกซ์พาริตีเช็กขนาด (1×n-k)

$m_{(1\times k)}$ คือ เมทริกซ์ข้อมูลขนาด (1×k)

เขียนเมทริกซ์พาริตีเช็กในรูป

$$\mathbf{H} = [\mathbf{H}_1 : \mathbf{H}_2] \tag{2.5}$$

$$\mathbf{H}^T = \begin{bmatrix} \mathbf{H}_1^T \\ \mathbf{H}_2^T \end{bmatrix} \tag{2.6}$$

จากนั้นแทนค่าสมการจะได้สมการใหม่ดังนี้

$$\mathbf{CH}^T = [\mathbf{p} \mid \mathbf{m}] \begin{bmatrix} \mathbf{H}_1^T \\ \mathbf{H}_2^T \end{bmatrix} = \mathbf{0} \tag{2.7}$$

$$\mathbf{CH}^T = \mathbf{mH}_1^T + \mathbf{pH}_2^T = \mathbf{0} \tag{2.8}$$

$$\mathbf{p} = \mathbf{mH}_1^T + \left(\mathbf{H}_2^T\right)^{-1} \tag{2.9}$$

เมทริกซ์พาริตีเช็ก H_2 มีขนาดเท่ากับ (m×m)

การเข้ารหัส คือ การคำนวณค่าเมทริกซ์ p จากสมการ จากนั้นทำการแทนค่า p ลงในสมการ จะได้คำรหัสสำหรับรหัสแอลดีพีซี ส่วนสมการ $CH^T = 0$ ใช้ในการตรวจความถูกต้องของคำรหัส

ตัวอย่าง การเข้ารหัสแอลดีพีซีกำหนดให้ข้อมูลที่ต้องการเข้ารหัสคือ [1 0 1 1] ทำการเข้ารหัสดังนี้

ให้ $H = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 & 1 & 1 \end{bmatrix}$

สูตร $A^{-1} = \frac{1}{|A|}$; $|A| = \begin{matrix} a & b & c & a & b \\ d & e & f & d & e \\ g & h & i & g & h \end{matrix}$ (− − − ; + + +)

หา $mH_1^T \rightarrow [1\ 0\ 1\ 1] \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix} = [1+0+0+0 \quad 0+0+0+1 \quad 0+0+1+1]$

$= [1 \quad 1 \quad 0]$ # (ตอนบวกเป็นการบวกแบบมอดูโล 2)

หา $(H_2^T)^{-1}$; จาก $H_2 = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix}$; $H_2^T = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \end{bmatrix}$

จากสูตร $(A_{2\times2})^{-1} = \frac{1}{|A|}\begin{bmatrix} d & -b \\ -c & a \end{bmatrix}$; หา $|H_2^T| = \begin{matrix} 1 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 \end{matrix}$

$(A_{3\times3})^{-1} = \frac{1}{|A|}.\text{adjoint } A$

$\therefore |H_2^T| = (0+1+0) - (0+1+1) = 1 - 2 = -1$

$$A_{11} = +\begin{bmatrix}0 & 1\\ 1 & 1\end{bmatrix}; \qquad A_{12} = -\begin{bmatrix}1 & 1\\ 1 & 1\end{bmatrix}; \qquad A_{13} = +\begin{bmatrix}1 & 0\\ 1 & 1\end{bmatrix}$$

$$= +(0-1) = -1 \qquad = -(1-1) = 0 \qquad = +(1-0) = 1$$

$$A_{21} = -\begin{bmatrix}1 & 0\\ 1 & 1\end{bmatrix}; \qquad A_{22} = +\begin{bmatrix}1 & 0\\ 1 & 1\end{bmatrix}; \qquad A_{23} = +\begin{bmatrix}1 & 1\\ 1 & 1\end{bmatrix}$$

$$= -(1-0) = -1 \qquad = +(1-0) = 1 \qquad = -(1-1) = 0$$

$$A_{31} = +\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}; \qquad A_{32} = -\begin{bmatrix}1 & 0\\ 1 & 1\end{bmatrix}; \qquad A_{33} = +\begin{bmatrix}1 & 1\\ 1 & 0\end{bmatrix}$$

$$= +(1-0) = 1 \qquad = -(1-0) = -1 \qquad = +(0-1) = -1$$

$$\left(H_2^T\right)^{-1} = \frac{1}{-1}\begin{bmatrix}-1 & 0 & 1\\ -1 & 1 & 0\\ 1 & -1 & -1\end{bmatrix}; \quad \left[\left[H_2^T\right]^{-1}\right]^T = \begin{bmatrix}+1 & +1 & -1\\ 0 & -1 & +1\\ -1 & 0 & +1\end{bmatrix}$$

$$[1 \quad 1 \quad 0]\begin{bmatrix}1 & 1 & -1\\ 0 & -1 & 1\\ -1 & 0 & 1\end{bmatrix} = [1+0+0 \quad 1-1+0 \quad -1+1+0] = [1 \quad 0 \quad 0] \quad \#$$

เพราะฉะนั้น ผลการเข้ารหัสจะได้คำรหัสดังนี้ [1 0 1 1 1 0 0]

2.4.2 **การถอดรหัสแบบวนซ้ำ** การเข้ารหัสมีผลทำให้ข้อมูลแต่ละบิตมีความสัมพันธ์กันผ่านทางโครงสร้างของเมทริกซ์พาริตีเช็กซึ่งการถอดรหัสก็จะอาศัยความสัมพันธ์เหล่านี้มาช่วยในการถอดรหัส อัลกอริทึมสำหรับการถอดรหัสแอลดีพีซีนั้น มีชื่อเรียกที่หลากหลายทั้งSum-Product Algorithm (SPA), Belief Propagation Algorithm (BPA) และ Message Passing Algorithm (MPA) ซึ่งเป็นรูปแบบการถอดรหัสที่เรียกว่าการถอดรหัสวนซ้ำแบบซอฟต์ (Soft Iterative Decoding)โดยขั้นตอนการถอดรหัสประกอบด้วยขั้นตอนหลักสองขั้นตอน คือ สร้างสมการจากโครงสร้างของเมทริกซ์พาริตีเช็กแล้วจึงเขียนแผนภาพกราฟแทนเนอร์จากนั้นจึงคำนวณหาค่าของข้อมูลแต่ละบิตตามโครงสร้างอัลกอริทึมที่ใช้ในการถอดรหัส

1) สร้างสมการจากโครงสร้างของเมทริกซ์พาริตีเช็ก และเขียนแผนภาพกราฟแทนเนอร์ พิจารณารหัสแอลดีพีซีแบบคงที่ขนาด (10, 5) ซึ่งมีค่า W_c=2 และ W_r=4 ตามสมการเมทริกซ์พาริตีเช็ก ที่ได้ดังสมการที่ (2.1)

จากโครงสร้างของเมทริกซ์พาริตีเช็กจะได้ความสัมพันธ์เป็นสมการที่แสดงความสัมพันธ์ของแต่ละบิต โดยที่แต่ละแถวของเมทริกซ์พาริตีเช็ก เรียกว่า โหนดเช็ก (Check Node) และแต่ละหลักเรียกว่า โหนดสัญลักษณ์หรือโหนดบิต (Bit Node) หรือโหนดตัวแปร (Variable Node) แสดงสมการได้ดังนี้

โหนดเช็กที่ 1: $c_1 + c_2 + c_3 + c_4 = 0$

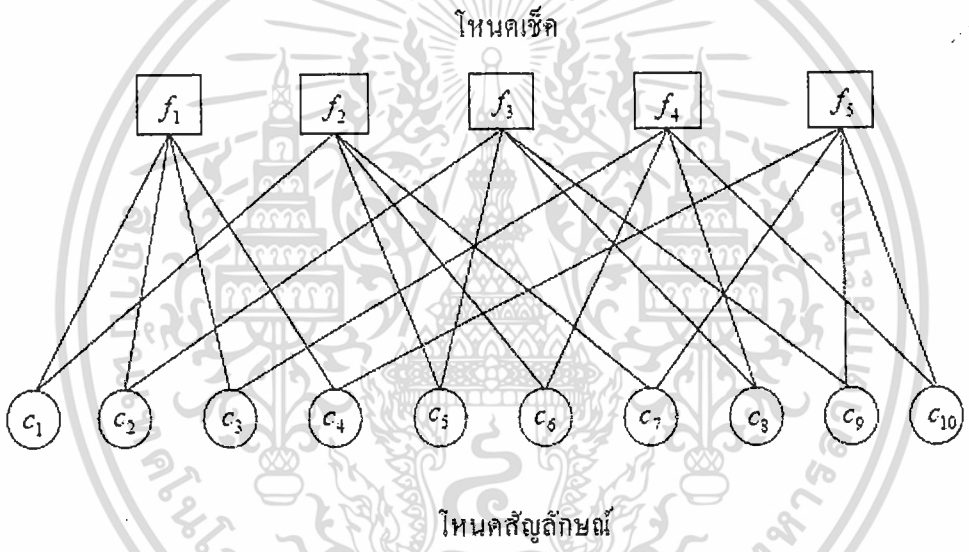
โหนดเช็กที่ 2: $c_1 + c_5 + c_6 + c_7 = 0$

โหนดเช็กที่ 3: $c_2 + c_5 + c_8 + c_9 = 0$

โหนดเช็กที่ 4: $c_3 + c_6 + c_8 + c_{10} = 0$

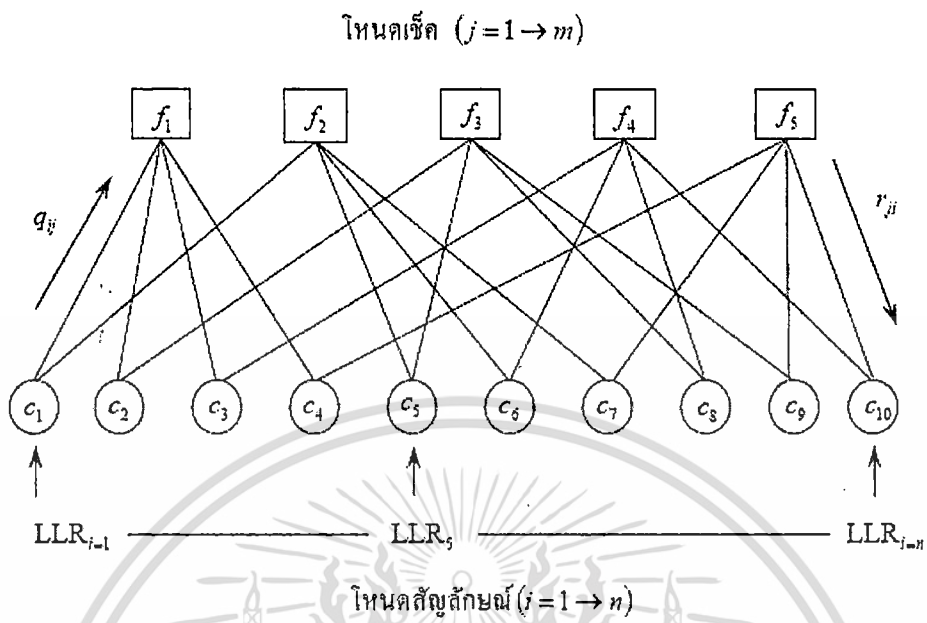
โหนดเช็กที่ 5: $c_4 + c_7 + c_9 + c_{10} = 0$

โดย c_i แทนตำแหน่งของเลขหนึ่งของแต่ละโหนดสัญลักษณ์ในโหนดเช็กที่กำลังพิจารณาจากสมการข้างต้น สามารถสร้างกราฟแทนเนอร์เพื่อช่วยในการถอดรหัสดังรูปที่ 2.4 โดยเริ่มจากวาดรูปสี่เหลี่ยมตามจำนวนของโหนดเช็ก จากนั้นวาดรูปวงกลมตามจำนวนของโหนดสัญลักษณ์ ในขั้นตอนสุดท้ายเป็นการลากเส้นตรงเชื่อมความสัมพันธ์ระหว่างโหนดเช็กและโหนดสัญลักษณ์ตามความสัมพันธ์ในสมการ



รูปที่ 2.4 กราฟแทนเนอร์

2) การคำนวณหาค่าของข้อมูลแต่ละบิต ตามโครงสร้างของอัลกอริทึมที่ใช้ในการถอดรหัสจากกราฟแทนเนอร์ จะสามารถถอดรหัสได้ด้วยหลายวิธีแต่ในวิทยานิพนธ์นี้ ใช้วิธีการที่เรียกว่า Log-Domain SPA Decoder



รูปที่ 2.5 แสดงการส่งผ่านข้อมูลระหว่างโหนดสัญลักษณ์และโหนดเช็ก

โดยหลักการทำงานจะเป็นการแลกเปลี่ยนข่าวสารแบบซอฟต์ระหว่างโหนดสัญลักษณ์ที่ i และโหนดเช็กที่ j พิจารณารูปที่ 2.5 อินพุตของการถอดรหัสจะอยู่ในรูปของอัตราส่วนความน่าเป็นไปได้แบบล็อก (Log Likelihood Ratios : LLRs) ของตัวแปรสุ่ม c_i ตามสมการ โดยในแต่ละโหนดสัญลักษณ์ที่ i จะส่งค่าความน่าจะเป็นของข่าวสารแบบซอฟต์ไปที่โหนดเช็กที่ j ผ่านตามเส้นความสัมพันธ์ที่เชื่อมถึงกัน และจากนั้นโหนดเช็กที่ j ก็จะทำการคำนวณค่าความน่าจะเป็นของข่าวสารที่ส่งมาจากโหนดสัญลักษณ์ที่ i แล้วจึงส่งผลที่ได้ของข่าวสารแบบซอฟต์ไปให้โหนดสัญลักษณ์ที่ i อีกครั้ง เพื่อนำข้อมูลที่ได้ไปใช้ในการตัดสินใจแบบฮาร์ดว่าควรจะเป็น 0 หรือ 1

กำหนดให้

q_{ij} = ความน่าจะเป็นของข่าวสารบิตที่iที่ส่งจากโหนดสัญลักษณ์ที่ iไปที่โหนดเช็กjว่าเป็น 0 หรือ 1

r_{ji} = ความน่าจะเป็นของข่าวสารบิตที่iที่ส่งจากโหนดเช็กที่ jไปที่โหนดสัญลักษณ์ iว่าเป็น 0 หรือ 1

ขั้นตอนและวิธีการสำหรับการถอดรหัสแบบ Log Domain SPA algorithm ประกอบด้วย 5 ขั้นตอนหลัก คือ

ขั้นตอนที่ 1 : คำนวณค่าเริ่มต้นของ $L(q_{ij})$ ที่ส่งจากโหนดสัญลักษณ์ที่ iไปยังโหนดเช็กที่ j ของในแต่ละบิต i ตั้งแต่บิตที่ 1 จนถึงบิตที่nผ่านสมการที่ (2.10)

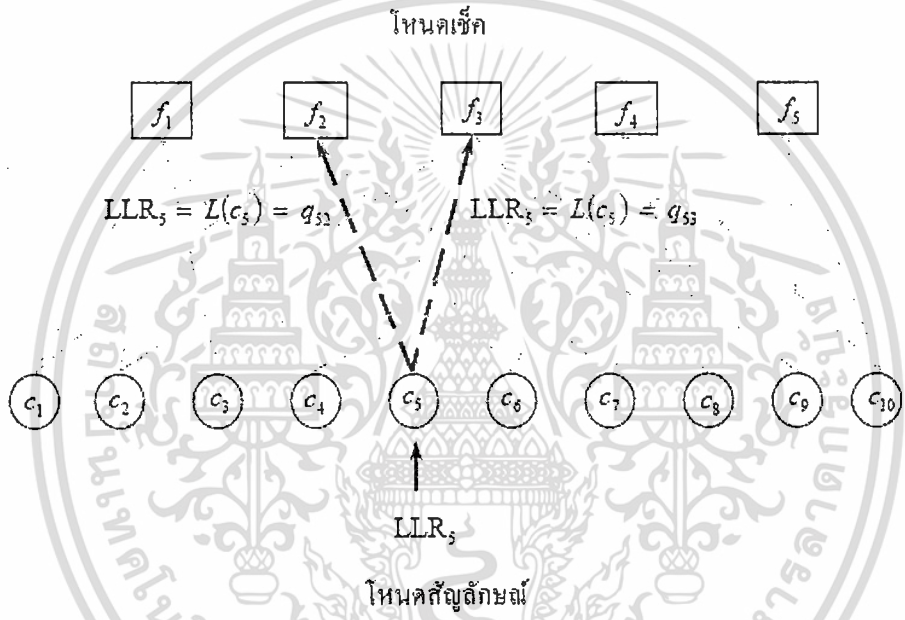
$$L(q_{ij}) = L(c_i) = 2y_i/\sigma^2 \quad (2.10)$$

$$L(c_i) = LLR_i = \log[P(c_i = 0|y_i)/P(c_i = 1|y_i)] \quad (2.11)$$

เมื่อ $L(c_i)$ คือ อัตราส่วนความน่าเชื่อถือหรือความน่าเป็นไปได้แบบล็อก

y_i คือ สัญญาณที่ได้รับผ่านช่องสัญญาณ

σ^2 คือ ค่าเบี่ยงเบนของสัญญาณรบกวนแบบเกาส์สีขาว



รูปที่ 2.6 แสดงค่า $L(q_{ji})$ ที่ส่งจากโหนดสัญลักษณ์ i ไปยังโหนดเช็ก j

ขั้นตอนที่ 2: คำนวณค่า $L(r_{ji})$ ที่ส่งจากโหนดเช็กที่ j ไปที่โหนดสัญลักษณ์ที่ i ตามเส้นความสัมพันธ์ที่เชื่อมถึงกันผ่านสมการที่ (2.12) ในแต่ละบิตที่ตั้งแต่บิตที่ 1 ถึงบิตที่ n

$$L(r_{ji}) = \prod_{i' \in V_j \setminus i} \alpha_{i'j} \cdot \emptyset\left(\sum_{i' \in V_j \setminus i} \emptyset(\beta_{i'j})\right) \quad (2.12)$$

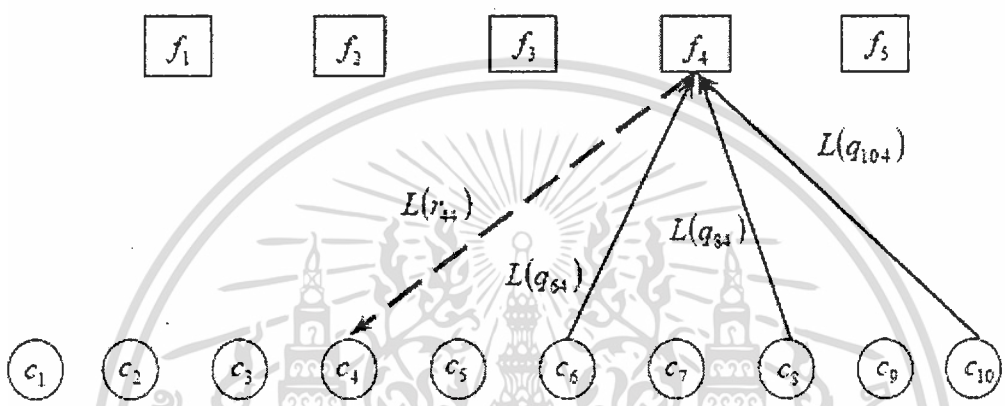
เมื่อ

$$\alpha_{ij} = \text{sgn}\{L(q_{ij})\} \text{และ } \beta_{ij} = |L(q_{ij})| \quad (2.13)$$

นิยามให้

$$\emptyset(x) = \log\{(e^x + 1)/(e^x - 1)\} \qquad (2.14)$$

V_{Ni} โดยแทนการพิจารณาข่าวสารจากทุกโหนดสัญลักษณ์ที่เชื่อมต่อกับโหนดเช็กที่ J ยกเว้นโหนดสัญลักษณ์ที่กำลังพิจารณารูปที่ 2.7 แสดงแผนภาพการคำนวณค่า L (r_{44})

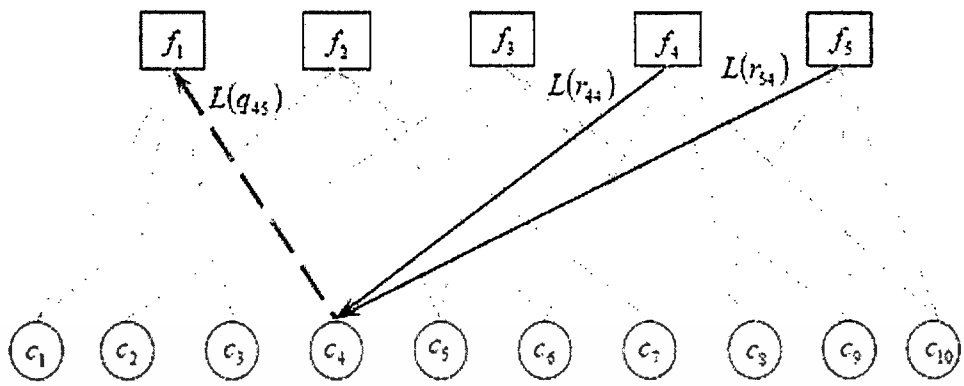


รูปที่ 2.7 แสดงค่า $L(r_{ij})$ ที่ส่งจากโหนดเช็ก j ไปยังโหนดสัญลักษณ์ i

ขั้นตอนที่ 3 : จะเป็นการปรับปรุงข่าวสารของ $L(q_{ij})$ เพื่อที่จะใช้เป็นอินพุตของการถอดรหัสแบบวนซ้ำที่ส่งจากโหนดสัญลักษณ์ที่ i ไปยังโหนดเช็กที่ j ของในแต่ละบิตที่ iตั้งแต่บิตที่ 1 จนถึงบิตที่nผ่านทางสมการที่ (2.15)

$$L(q_{ij}) = L(c_i) + \sum_{j' \in c_{i \setminus j}} L(r_{j'i}) \qquad (2.15)$$

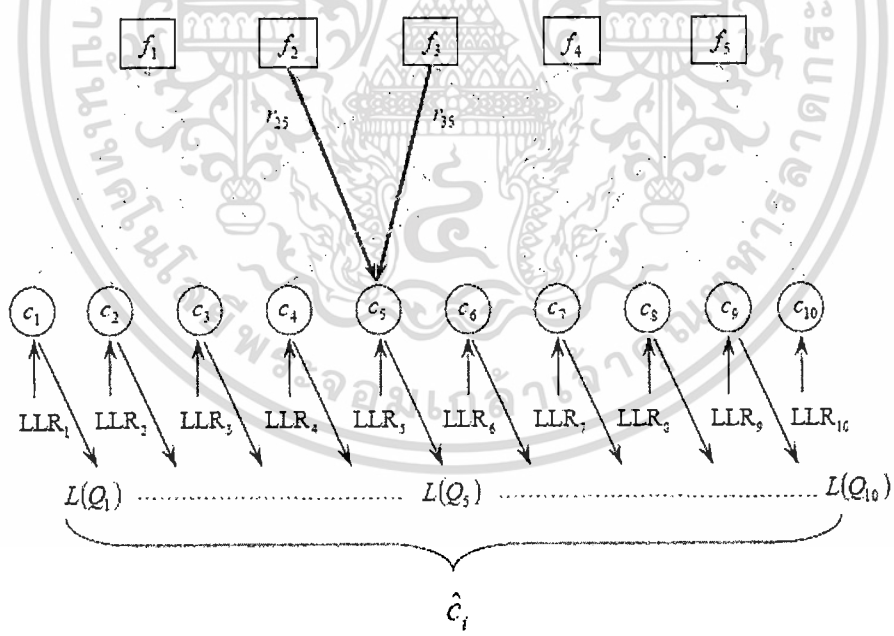
C_{Nj} แทนการพิจารณาผลรวมของข่าวสาร $L(r_{ji})$ จากทุกโหนดเช็กที่ j ที่เชื่อมต่อกับโหนดสัญลักษณ์ ยกเว้นข่าวสารที่ใช้เส้นทางเดียวกับ $L(q_{ij})$ รูปที่ 2.8 แสดงแผนภาพการปรับปรุงข่าวสารของ $L(q_{ij})$



รูปที่ 2.8 แสดงค่า $L(q_{ij})$ เพื่อใช้เป็นอินพุตของการถอดรหัสแบบวนซ้ำ

ขั้นตอนที่ 4 : เป็นการคำนวณหาค่าซอฟต์เอาต์พุตของการถอดรหัสของแต่ละบิตที่ i ตั้งแต่บิตที่ 1ถึงบิตที่nผ่านสมการที่ (2.16)

$$L(Q_i) = L(c_i) + \sum_{j \in C_i} L(r_{ji}) \qquad (2.16)$$



รูปที่ 2.9 แผนภาพการหาค่าซอฟต์เอาต์พุตของการถอดรหัส

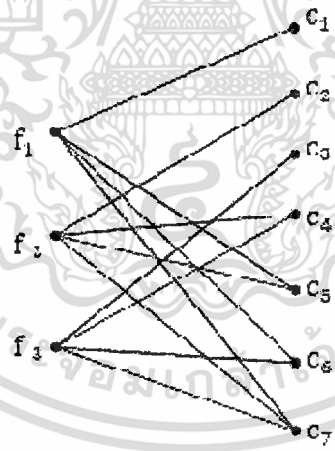
ขั้นตอนที่ 5 : เป็นการนำค่าซอฟต์เอาต์พุตที่ได้จากขั้นตอนที่ 4 ของแต่ละบิตมาทำการตัดสินใจแบบฮาร์ดผ่านสมการที่ (2.17)

$$\hat{\mathbf{c}}_i = 1 \text{ ถ้ามิฉะนั้น } \hat{\mathbf{c}}_i = 0 \qquad (2.17)$$

จากขั้นตอนที่ 1 – 5 ก็จะเป็นการเสร็จสิ้นขั้นตอนในการถอดรหัสหนึ่งรอบ ซึ่งจะพบว่าถ้าระบบที่ใช้การถอดรหัสแบบไม่มีการวนซ้ำ ขั้นตอนที่ 3 ก็สามารถยกเลิกและข้ามมายังขั้นตอนที่ 4 ได้เลย หรือในกรณีที่ ใช้การวนซ้ำก็จะทำตามขั้นตอนที่ 1 – 5 ตามจำนวนรอบการวนซ้ำที่ได้กำหนดไว้หรือใช้สมการ $\hat{\mathbf{c}}\mathbf{H}^T=0$ เป็นเงื่อนไขเสร็จสิ้นขั้นตอนในการถอดรหัส

ตัวอย่าง การถอดรหัสด้วยรหัสแอลดีพีซี

$$\begin{array}{c} c_1\ c_2\ c_3\ c_4\ c_5\ c_6\ c_7 \\ H = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 & 1 & 1 \end{bmatrix} \begin{matrix} f_1 \\ f_2 \\ f_3 \end{matrix} \end{array} \quad \begin{matrix} f \to c = r \\ c \to f = q \end{matrix}$$



$c = [1\ \ 0\ \ 1\ \ 1\ \ 1\ \ 0\ \ 0]$

Modulate coder word $\rightarrow [1\ \ -1\ \ \ 1\ \ \ 1\ \ \ 1\ \ -1\ \ -1]$

$$\sigma^2 = \frac{\sum(x_i - \bar{x})^2}{N} = \frac{\sum x_i^2}{N} - (\bar{x})^2$$

$$\sum x = 0.7 - 0.7 + 1.3 + 0.7 + 0.7 - 0.7 - 1.3 = 0.7$$

$$\bar{x} = \frac{0.7}{7} = 0.1; \quad \bar{x}^2 = 0.01$$

$$\sum x_i^2 = 0.49 + 0.49 + 1.69 + 0.49 + 0.49 + 0.49 + 1.69 = 5.83$$

$$\frac{\sum x_i^2}{N} = \frac{5.83}{7} = 0.833$$

$$\sigma^2 = 0.833 - 0.01 = 0.823 \quad \#$$

$$y_i = [\text{mod} - \text{coder word} + \text{noise}]$$

$$\text{noise} = [0.3 \rightarrow -0.3]$$

$$y_i = [1 - 0.3\ \ -1 + 0.3\ \ 1 + 0.3\ \ 1 - 0.3\ \ 1 - 0.3\ \ -1 + 0.3\ \ -1 - 0.3]$$

$$= [0.7\ \ -0.7\ \ \ 1.3\ \ \ 0.7\ \ \ 0.7\ \ -0.7\ \ -1.3]$$

$$L(c_i) = 2y_i/\sigma^2; \quad \sigma^2 = 0.823$$

$$\overset{1}{L}(c_i) = [\overset{1}{1.701}\ \ \overset{2}{-1.701}\ \ \overset{3}{3.159}\ \ \overset{4}{1.701}\ \ \overset{5}{1.701}\ \ \overset{6}{-1.701}\ \ \overset{7}{-3.159}] \leftarrow \text{รอบที่ } 1$$

$$L(r_{ji}) = \emptyset\left[\sum \emptyset\left(L(q_{i'j})\right)\right] - \left[\emptyset\left(L(q_{i'j})\right)\right] \cdot [\text{sgn}]\left(L(q_{i'j})\right); \quad i \neq i'$$

$$L(r_{11}) = [\text{sgn}]\emptyset[\emptyset L(q_{51}) + \emptyset L(q_{61}) + \emptyset L(q_{71})]$$

$$= [+]\emptyset[\emptyset(1.701) + \emptyset(-1.701) + \emptyset(-3.159)] = 3.159$$

$$L(r_{15}) = [\text{sgn}]\emptyset[\emptyset L(q_{11}) + \emptyset L(q_{61}) + \emptyset L(q_{71})]$$

$$= [+]\emptyset[\emptyset(1.701) + \emptyset(-1.701) + \emptyset(-3.159)] = 3.159$$

$$L(r_{16}) = [\text{sgn}]\emptyset[\emptyset L(q_{11}) + \emptyset L(q_{51}) + \emptyset L(q_{71})]$$

$$= [-]\emptyset[\emptyset(1.701) + \emptyset(1.701) + \emptyset(-3.159)] = -0.243$$

$$L(r_{17}) = [\text{sgn}]\emptyset[\emptyset L(q_{11}) + \emptyset L(q_{51}) + \emptyset L(q_{61})]$$

$$= [-]\emptyset[\emptyset(1.701) + \emptyset(1.701) + \emptyset(-1.701)] = -1.701$$

$$L(r_{22}) = [sgn]\emptyset[\emptyset L(q_{42}) + \emptyset L(q_{52}) + \emptyset L(q_{72})]$$

$$= [-]\emptyset[\emptyset(1.701) + \emptyset(1.701) + \emptyset(-3.159)] = -0.243$$

$$L(r_{24}) = [sgn]\emptyset[\emptyset L(q_{22}) + \emptyset L(q_{52}) + \emptyset L(q_{72})]$$

$$= [+]\emptyset[\emptyset(-1.701) + \emptyset(1.701) + \emptyset(-3.159)] = 3.159$$

$$L(r_{25}) = [sgn]\emptyset[\emptyset L(q_{22}) + \emptyset L(q_{42}) + \emptyset L(q_{72})]$$

$$= [+]\emptyset[\emptyset(-1.701) + \emptyset(1.701) + \emptyset(-3.159)] = 3.159$$

$$L(r_{27}) = [sgn]\emptyset[\emptyset L(q_{22}) + \emptyset L(q_{42}) + \emptyset L(q_{52})]$$

$$= [-]\emptyset[\emptyset(-1.701) + \emptyset(1.701) + \emptyset(1.701)] = -1.701$$

$$L(r_{33}) = [sgn]\emptyset[\emptyset L(q_{43}) + \emptyset L(q_{63}) + \emptyset L(q_{73})]$$

$$= [+]\emptyset[\emptyset(1.701) + \emptyset(-1.701) + \emptyset(-3.159)] = 3.159$$

$$L(r_{34}) = [sgn]\emptyset[\emptyset L(q_{33}) + \emptyset L(q_{63}) + \emptyset L(q_{73})]$$

$$= [+]\emptyset[\emptyset(3.159) + \emptyset(-1.701) + \emptyset(-3.159)] = 1.701$$

$$L(r_{36}) = [sgn]\emptyset[\emptyset L(q_{33}) + \emptyset L(q_{43}) + \emptyset L(q_{73})]$$

$$= [-]\emptyset[\emptyset(3.159) + \emptyset(1.701) + \emptyset(-3.159)] = -1.701$$

$$L(r_{37}) = [sgn]\emptyset[\emptyset L(q_{33}) + \emptyset L(q_{43}) + \emptyset L(q_{63})]$$

$$= [-]\emptyset[\emptyset(3.159) + \emptyset(1.701) + \emptyset(-1.701)] = -3.159$$

$$L(r_{ij}) = [3.519\ \ 3.519 - 0.243 - 1.701 - 0.243\ \ 3.519\ \ 3.519 - 1.701\ \ 3.519\ \ 1.701 - 1.701 - 3.159]$$
(11 15 16 17 22 24 25 27 33 34 36 37)

$$L(Q_i) = L(c_i) + [\sum L(r_{ij})] - L(r_{ij}),\ j \neq j'$$

$$L(Q_1) = 1.701 + 3.159 = 4.86 \rightarrow 1 \quad (c_{11}, r_{11})$$

$$L(Q_2) = (-1.701) + (-0.243) = -1.944 \rightarrow 0 \quad (c_{12}, r_{22})$$

$$L(Q_3) = 3.159 + 3.159 = 6.318 \rightarrow 1 \quad (c_{13}, r_{33})$$

$$L(Q_4) = 1.701 + 3.159 + 1.701 = 6.561 \rightarrow 1 \quad (c_{14}, r_{24}, r_{34})$$

$$L(Q_5) = 1.701 + 3.159 + 3.159 = 8.019 \rightarrow 1 \quad (c_{15}, r_{15}, r_{25})$$

$$L(Q_6) = (-1.701) + (-0.243) + (-1.701) = -3.645 \rightarrow 0 \quad (c_{16}, r_{16}, r_{36})$$

$$L(Q_7) = (-3.159) + (-1.701) + (-1.701) + (-3.159) = -6.561 \rightarrow 0 \quad (c_{17}, r_{17}, r_{27}, r_{37})$$

$$\therefore c_i = [1\ \ 0\ \ 1\ \ 1\ \ 1\ \ 0\ \ 0] \quad (c_1\ c_2\ c_3\ c_4\ c_5\ c_6\ c_7)$$

ตัวอย่าง การถอดรหัสแอลดีพีซี (กรณีมีสัญญาณรบกวนทำให้เกิดข้อผิดพลาด 1 บิต)

$$yi = [2.1\ \ -2.1\ \ 2.1\ \ -0.1\ \ 2.1\ \ -2.1\ \ -2.1]$$
$$\bar{x}^2 = -0.014\ ;\quad \sigma^2 = 4.41\#$$
$$L(c_i) = 2y_i/\sigma^2$$
1 2 3 4 5 6 7

$$L(c_i) = [0.952\ \ -0.952\ \ 0.952\ \ -0.045\ \ 0.952\ \ -0.952\ \ -0.952] \leftarrow \text{รอบที่ 1}$$

$$L(r_{ji}) = [\text{sgn}]\left(L(q_{i'j})\right).[\sum \emptyset(|L(q_{i'j})|)];i \neq i';$$
$$\emptyset(x) = \log((e^x + 1)/(e^x - 1))$$

$$L(r_{11}) = [\text{sgn}]\emptyset[\emptyset L(q_{51}) + \emptyset L(q_{61}) + \emptyset L(q_{71})]$$
$$= [+\ -\ -]\emptyset[0.3535 + 0.3535 + 0.3535] = 0.3137$$

$$L(r_{15}) = [sgn]\emptyset[\emptyset L(q_{11}) + \emptyset L(q_{61}) + \emptyset L(q_{71})]$$
$$= [+\ -\ -]\emptyset[0.3535 + 0.3535 + 0.3535] = \emptyset 0.888 = 0.3137$$

$$L(r_{16}) = [sgn]\emptyset[\emptyset L(q_{11}) + \emptyset L(q_{51}) + \emptyset L(q_{71})]$$
$$= [+ + -]\emptyset[0.3535 + 0.3535 + 0.3535] = \emptyset 0.888 = -0.265$$

$$L(r_{17}) = [sgn]\emptyset[\emptyset L(q_{11}) + \emptyset L(q_{51}) + \emptyset L(q_{61})]$$
$$= [+ + -]\emptyset[0.3535 + 0.3535 + 0.3535] = \emptyset 0.888 = -0.265$$

$$L(r_{22}) = [sgn]\emptyset[\emptyset L(q_{42}) + \emptyset L(q_{52}) + \emptyset L(q_{72})]$$
$$= [-\ +\ -]\emptyset[0.0453 + 0.3535 + 0.3535] = \emptyset 0.593 = 0.134$$

$$L(r_{24}) = [sgn]\emptyset[\emptyset L(q_{22}) + \emptyset L(q_{52}) + \emptyset L(q_{72})]$$
$$= [-\ +\ -]\emptyset[0.3535 + 0.3535 + 0.3535] = \emptyset 0.888 = 0.265$$

$$L(r_{25}) = [sgn]\emptyset[\emptyset L(q_{22}) + \emptyset L(q_{42}) + \emptyset L(q_{72})]$$
$$= [-\ -\ -]\emptyset[0.3535 + 0.0453 + 0.3535] = \emptyset 0.593 = -0.134$$

$$L(r_{27}) = [sgn]\emptyset[\emptyset L(q_{22}) + \emptyset L(q_{42}) + \emptyset L(q_{52})]$$
$$= [- - +]\emptyset[0.3535 + 0.0453 + 0.3535] = \emptyset 0.593 = 0.134$$

$$L(r_{33}) = [sgn]\emptyset[\emptyset L(q_{43}) + \emptyset L(q_{63}) + \emptyset L(q_{73})]$$
$$= [- - -]\emptyset[0.0453 + 0.3535 + 0.3535] = \emptyset 0.593 = -0.134$$

$$L(r_{34}) = [sgn]\emptyset[\emptyset L(q_{33}) + \emptyset L(q_{63}) + \emptyset L(q_{73})]$$
$$= [+ - -]\emptyset[0.3535 + 0.3535 + 0.3535] = \emptyset 0.888 = 0.265$$

$$L(r_{36}) = [sgn]\emptyset[\emptyset L(q_{33}) + \emptyset L(q_{43}) + \emptyset L(q_{73})]$$
$$= [+ - -]\emptyset[0.3535 + 0.0453 + 0.3535] = \emptyset 0.593 = 0.134$$

$$L(r_{37}) = [sgn]\emptyset[\emptyset L(q_{33}) + \emptyset L(q_{43}) + \emptyset L(q_{63})]$$
$$= [+ - -]\emptyset[0.3535 + 0.0453 + 0.3535] = \emptyset 0.593 = 0.134$$

$$L(r_{ji}) =$$

$$\underset{11}{[0.3137} \quad \underset{15}{0.3137} \quad \underset{16}{-0.3137} \quad \underset{17}{-0.3137} \quad \underset{22}{0.0829} \quad \underset{24}{0.3137} \quad \underset{25}{-0.0829} \quad \underset{27}{0.0829}$$
$$-0.0829 \quad 0.3137 \quad 0.0829 \quad 0.0829]$$
$$\underset{33}{} \quad \underset{34}{} \quad \underset{36}{} \quad \underset{37}{}$$

$$L(Q_i) = L(c_i) + \left[\sum L(r_{ji})\right] - L(r_{ji}) ; \ j \neq j'$$

$$L(Q_1) = \underset{c_{11}}{0.9521} + \underset{r_{11}}{0.3137} = 1.2658 \rightarrow 1$$

$$L(Q_2) = \underset{c_{12}}{(-0.9521)} + \underset{r_{22}}{0.0829} = -0.8691 \rightarrow 0$$

$$L(Q_3) = \overset{c_{13}}{0.9521} + \overset{r_{33}}{(-0.0829)} = 0.8691 \rightarrow 1$$

$$L(Q_4) = \overset{c_{14}}{(-0.0453)} + \overset{r_{24}}{0.3137} + \overset{r_{34}}{0.3137} = 0.5821 \rightarrow 1$$

$$L(Q_5) = \overset{c_{15}}{0.9521} + \overset{r_{15}}{0.3137} + \overset{r_{25}}{(-0.0829)} = 1.1828 \rightarrow 1$$

$$L(Q_6) = \overset{c_{16}}{(-0.9521)} + \overset{r_{16}}{(-0.3137)} + \overset{r_{36}}{0.0829} = -1.1828 \rightarrow 0$$

$$L(Q_7) = \overset{c_{17}}{(-0.9521)} + \overset{r_{17}\ r_{27}\ r_{37}}{(-0.3137) + (0.0829) + (0.0829)} = -1.0999 \rightarrow 0$$

$$\overset{c_1\ c_2\ c_3\ c_4\ c_5\ c_6\ c_7}{\therefore c_i = [1\ \ 0\ \ 1\ \ 1\ \ 1\ \ 0\ \ 0]} \ ; \ m = [1\ 0\ 1\ 1] \ ; \ p = [1\ 0\ 0]\#$$

2.5 การทดสอบสมรรถนะของรหัสแอลดีพีซี

พิจารณาจากลักษณะเฉพาะของรหัสแอลดีพีซี ดังนี้ สมรรถนะของรหัสแอลดีพีซีทำงานเข้าใกล้ขีดจำกัดของแชนนอลเป็นรหัสบล็อกที่ดีสำหรับการแก้ไขข้อผิดพลาดของข้อมูล มีระดับความผิดพลาดต่ำในการถอดรหัสแอลดีพีซีจะมีความเป็นเชิงเส้นในรูปแบบของเวลาเหมาะสมกับการทำงานแบบคู่ขนานคำรหัสที่ยาวมาก ๆ จะทำให้มีการวนซ้ำมากขึ้นและจะทำให้ค่าอัตราส่วนของพลังงานสัญญาณต่อสัญญาณรบกวนต่ำและคำรหัสที่สั้นจะเข้ารหัสง่ายและจะได้ค่าอัตรารหัสสูง หรือค่า R≈1

2.5.1 การปราศจากลูป 4 (Without Loop 4) พิจารณานิยามลูปของรหัสแอลดีพีซี ดังนี้

เมทริกซ์พาริตีเช็กใด ๆ จะมีลูปขนาดเท่ากับ 4 เมื่อตำแหน่งของเลข 1 ในเมทริกซ์พาริตีเช็กเกิดลูปปิดตามสมการที่ (2.18)

$$(A_{i,j}), (A_{i,b}), (A_{a,b}), (A_{a,j}) \qquad (2.18)$$

เมื่อ A เป็นตำแหน่งของเลข 1 ในเมทริกซ์พาริตีเช็ก และค่าคงที่ I,j,a และ b เป็นค่าของแถวและหลักของเมทริกซ์พาริตีเช็กโดยที่ i, a≤m และ j, b≤n หรืออาจกล่าวได้ว่าลูปขนาดเท่ากับ 4 ในเมทริกซ์พาริตีเช็ก คือ ลูปปิดของเลขหนึ่งที่มีการใช้แถวและหลักร่วมกันเท่ากับ 2 แถว และ 2 หลัก

พิจารณารหัสแอลดีพีซีขนาด (10,5) ซึ่งมีค่า W_c=2 และ W_r=W_c (n/k) = 4 และมีลูป 4 ดังสมการที่ (2.19)

$$\mathbf{H} = \begin{bmatrix} \tilde{1} & 1 & 1 & \tilde{1} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \hat{1} & \hat{1} & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & \hat{1} & \hat{1} & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ \tilde{1} & 0 & 0 & \tilde{1} & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}_{(m=5,n=10)} \quad (2.19)$$

จากเมทริกซ์พาริตีเช็กจะพบว่าเกิดลูปขนาดเท่ากับ 4 จำนวนถึง 2 ลูป กล่าวคือ

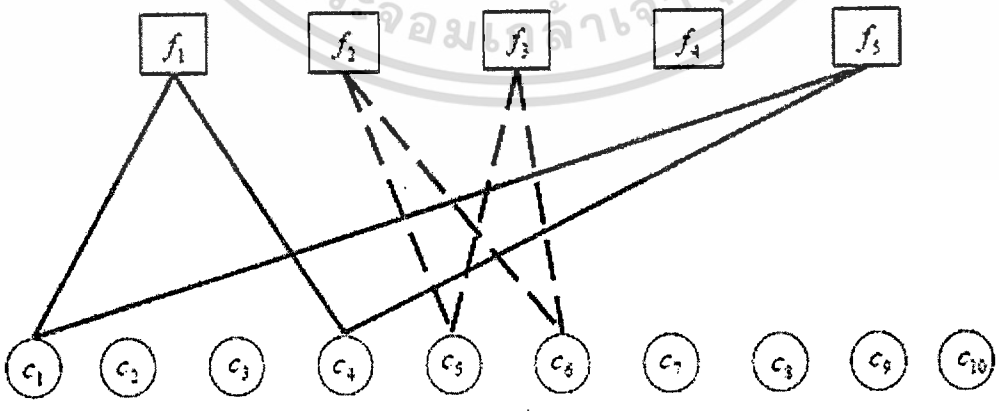
ลูปที่ 1: ตำแหน่งของ $\tilde{1}$ กับลูปปิด $(A_{1,1})$, $(A_{1,4})$, $(A_{5,4})$, $(A_{5,1})$

ลูปที่ 2: ตำแหน่งของ $\hat{1}$ กับลูปปิด $(A_{2,5})$, $(A_{2,6})$, $(A_{3,6})$, $(A_{3,5})$

เหตุผลของลูป 4 ที่เป็นเรื่องต้องห้ามสำหรับรหัสแอลดีพีซีก็คือ อัลกอริทึมสำหรับการถอดรหัสนั้น จะอาศัยหลักการของความน่าจะเป็นในการส่งผ่านข้อมูลโดยที่ความน่าจะเป็นของแต่ละเหตุการณ์นั้นเป็นอิสระต่อกัน ซึ่งผลของการมีลูป 4 นั้น ทำให้ความน่าจะเป็นในการส่งผ่านข้อมูลไม่เป็นอิสระต่อกันและส่งผลต่อสมรรถนะของการถอดรหัสเป็นอย่างมาก

จากหัวข้อที่กล่าวถึง เงื่อนไขของการสร้างเมทริกซ์พาริตีเช็กว่าจะต้องปราศจากลูป 4 และได้อธิบายถึงนิยามของลูปนั้นเมื่อพิจารณาจากอัลกอริทึมการถอดรหัสจะพบว่าเป็นการอาศัยการแลกเปลี่ยนข่าวสารแบบซอฟต์ระหว่างโหนดสัญลักษณ์ที่iและโหนดเช็กที่ j กับหลักการของความน่าจะเป็นว่าเหตุการณ์แต่ละเหตุการณ์นั้นเป็นอิสระต่อกันซึ่งในกรณีของเมทริกซ์พาริตีเช็กที่มีลูป 4 นั้นจะมีผลทำให้เกิดลูปปิดและเหตุการณ์การแลกเปลี่ยนข่าวสารแบบซอฟต์จะไม่เป็นอิสระต่อกัน

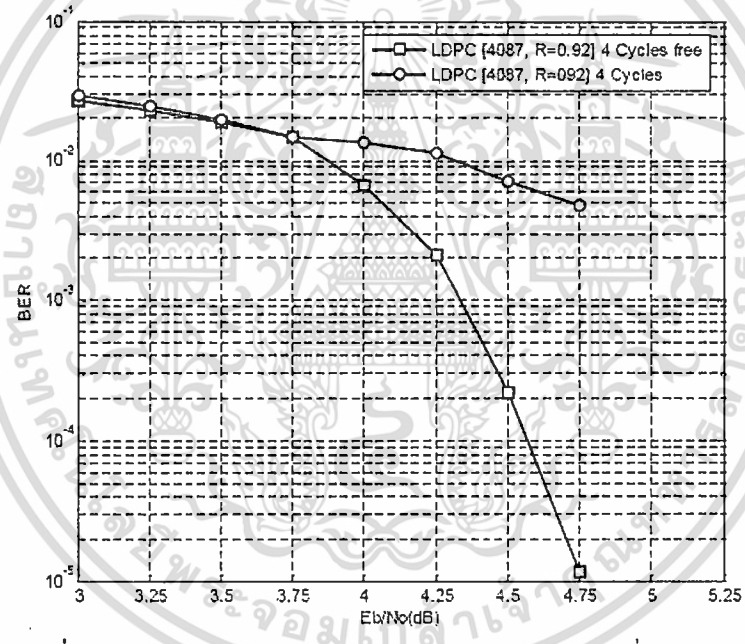
พิจารณาเมทริกซ์พาริตีเช็กในสมการแล้วสร้างกราฟแทนเนอร์พบว่ามีลูป 4 จำนวนสองลูปโดยลูปแรกพบที่โหนดสัญลักษณ์ c_1, c_4 กับโหนดเช็ก f_1, f_5 ส่วนลูปที่สองจะพบที่โหนดสัญลักษณ์ c_5, c_6 กับโหนดเช็ก f_2 , f_3 ดังรูปที่ 2.10



รูปที่ 2.10 แผนภาพของเมติกพาริตีเช็กที่มีลูป4

รูปที่ 2.11 เป็นการจำลองสมรรถนะการทำงานของระบบบันทึกข้อมูลแบบดิจิตอลที่มีอินพุต±1 และช่องสัญญาณเป็นแบบอุดมคติที่มีเพียงสัญญาณรบกวนแบบเกาส์สีขาวและใช้วิธีการเข้ารหัสช่องสัญญาณแบบแอลดีพีซีที่มีเมทริกซ์พาริตีเช็กมีลูป 4 เปรียบเทียบกับเมทริกซ์พาริตีเช็กที่ไม่มีลูป 4 กับความยาวคำรหัสขนาด 4087ด้วยอัตรารหัส 0.92 เท่ากันโดยทำการมอดุเลตแบบ Binary Phase Shift Keying (BPSK)

โดย E_b เป็นพลังงานเฉลี่ยของบิตข้อมูลและ $2N_0$ คือ ค่าความหนาแน่นสเปกตรัมกำลังงาน ซึ่งมีหน่วยเป็น W/Hz จากผลการจำลองสมรรถนะการทำงานอัตราความผิดพลาดบิตจะพบว่า รหัสแอลดีพีซีที่มีลูป 4 จะส่งผลต่อสมรรถนะอัตราความผิดพลาดบิตโดยให้อัตราบิตผิดพลาดที่สูงกว่าที่อัตราส่วนของพลังงานสัญญาณต่อสัญญาณรบกวนเท่ากัน



รูปที่ 2.11 แสดงสมรรถนะการทำงานของรหัสแอลดีพีซีที่มีลูป 4

2.5.2 อัตรารหัส (CodeRate : R) รหัสแอลดีพีซีมีขนาดบล็อกข้อมูลเป็น k มีขนาดบล็อกคำรหัสเป็น n และมีขนาดพาริตีเป็น m=n–k ดังรูปที่ 2.12 อัตรารหัสของรหัสนี้นิยามได้ดังสมการที่ (2.20)

$$R \geq R_d \triangleq \frac{k}{n} = 1 - \frac{m}{n} \qquad (2.20)$$

โดย R_d= อัตรารหัสที่ออกแบบ

อัตรารหัส R จะเท่ากับอัตรารหัสที่ออกแบบ $R=R_d$ เมื่อเมทริกซ์พาริตีเช็กมีลำดับที่เต็ม หรือแถวของเมทริกซ์พาริตีเช็กมีความเป็นอิสระจากกันทั้งหมด นอกจากนี้การนิยามโดยใช้ $\lambda(x)$ และ $\rho(x)$ สามารถทำได้ดังสมการที่ (2.21)

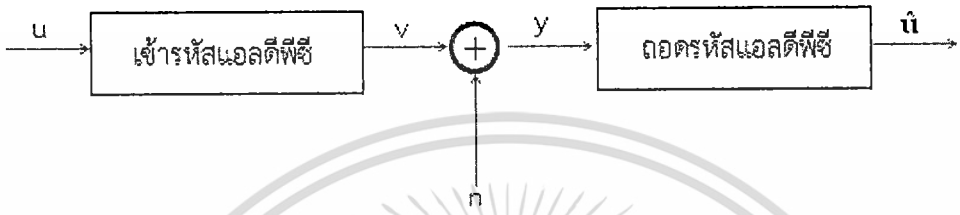
$$R_d = 1 - \frac{m}{n} = 1 - \frac{\sum_{i=2}^{d_c^{max}} \rho_i / i}{\sum_{i=2}^{d_v^{max}} \lambda_i / i} \qquad (2.21)$$

รูปที่ 2.12 แบบจำลองโครงสร้างบล็อกคำรหัส

เมื่อ k = ข้อมูลขนาด k บิต
n-k = จำนวนบิตพาริตี
n = คำรหัสขนาดn บิต
R = อัตรารหัสk/n ; $0 \le R \le 1$

2.5.3 **อัตราความผิดพลาดของข้อมูล** (Bit Error Rate : BER) หรืออัตราส่วนของจำนวนบิตผิดพลาดต่อจำนวนบิตข้อมูลทั้งหมด สามารถใช้เป็นตัวชี้บอกในการเปรียบเทียบสมรรถนะของระบบได้ คือกำหนดให้ระบบ 2 ระบบใช้อัตราส่วนของพลังงานสัญญาณต่อสัญญาณรบกวนเท่ากัน ดังนั้นระบบใดให้อัตราความผิดพลาดของข้อมูลที่วัดด้านขาออกของวงจรตรวจหาน้อยกว่าก็จะถือว่าระบบนั่นมีสมรรถนะมากกว่า โดยทั่วไปแล้วค่าอัตราความผิดพลาดจะเป็นฟังก์ชันของอัตราส่วนของพลังงานสัญญาณต่อสัญญาณรบกวนนั่นคือถ้าระบบใช้อัตราส่วนของพลังงานสัญญาณต่อสัญญาณรบกวนมากขึ้นค่าอัตราความผิดพลาดของข้อมูลก็จะลดลง อย่างไรก็ตามในทางปฏิบัติ ไม่สามารถกำหนดให้ระบบทำงานที่อัตราส่วนของพลังงานสัญญาณต่อสัญญาณรบกวนสูง ๆ เพราะว่าจะเสียค่าใช้จ่ายมาก นอกจากนี้ความสามารถในการออกแบบระบบการประมวลผลสัญญาณของระบบใดสามารถทำงานที่ระดับอัตราส่วนของพลังงานสัญญาณต่อสัญญาณรบกวนน้อย ๆ ได้จะเสมือนกับว่า ระบบที่ถูกออกแบบมานั้นสามารถที่จะทำงานที่ระดับความจุข้อมูลสูงได้ โดยปกติค่าอัตราความผิดพลาดของข้อมูลที่ใช้เป็นตัวกำหนดระดับความน่าเชื่อถือของระบบสำหรับงานประยุกต์ต่าง ๆ การรับส่งสัญญาณเสียงจะมีคุณภาพดีก็ต่อเมื่อค่าอัตราความผิดพลาดของข้อมูลน้อยกว่าหรือเท่ากับ 10^{-3} การรับส่งสัญญาณข้อมูลจะมี

คุณภาพดีก็ต่อเมื่อค่าอัตราความผิดพลาดของข้อมูลน้อยกว่าหรือเท่ากับ 10^{-5} การรับส่งสัญญาณผ่านสายใยแก้วนำแสงจะมีคุณภาพดีก็ต่อเมื่อค่าอัตราความผิดพลาดของข้อมูลน้อยกว่าหรือเท่ากับ 10^{-12} และอุปกรณ์ฮาร์ดดิสก์ไดร์ฟจะมีคุณภาพดีก็ต่อเมื่อค่าอัตราความผิดพลาดของข้อมูลน้อยกว่าหรือเท่ากับ 10^{-20} โดยแบบจำลองที่ใช้ในการหาอัตราความผิดพลาดบิตแสดงดังรูปที่ 2.13



รูปที่ 2.13 แบบจำลองระบบที่มีสัญญาณรบกวนแบบเกาส์สีขาว

เมื่อ u= ข้อมูลขนาด k บิต

v= คำรหัสขนาด m บิต

n= สัญญาณรบกวนเกาส์แบบสีขาวที่มีความหนาแน่นสเปกตรัมกำลังงานเท่ากับ $N_0/2$ W/Hz

y = สัญญาณที่ได้รับผ่านช่องสัญญาณ

ข้อมูลจะถูกเข้ารหัสด้วยรหัสแอลดีพีซีเพื่อให้ได้คำรหัส จากนั้นคำรหัสจะถูกมอดุเลตก่อนนำไปผ่านช่องสัญญาณรบกวนแบบเกาส์สีขาว หลังจากนั้นในฝั่งรับจะทำการถอดรหัสแอลดีพีซี และนำข้อมูลที่ได้ไปคำนวณหาอัตราความผิดพลาดของข้อมูลโดยที่อัตราส่วนของพลังงานสัญญาณต่อสัญญาณรบกวนคำนวณได้ตามสมการที่ 2.22

$$\left(\frac{E_b}{N_o}\right)_{dB} = 10\log_{10}\left(\frac{1}{R}\cdot\frac{E_c}{N_o}\right) \qquad (2.22)$$

เมื่อ E_c= พลังงานสัญญาณเฉลี่ยของบิตรหัส

E_b= พลังงานสัญญาณเฉลี่ยของบิตข้อมูล

R = อัตรารหัส

หมายเหตุ อัตราความผิดพลาดของข้อมูลจะถูกคำนวณจนกว่าจะได้บิตผิดพลาดรวมเท่ากับ 1,000 ในแต่ละอัตราส่วนของพลังงานสัญญาณต่อสัญญาณรบกวน

2.5.4 อัตราส่วนของพลังงานสัญญาณต่อสัญญาณรบกวน (Signal-to-Noise Ratio : SNR) เป็นตัวชี้บอกที่สามารถใช้ในการเปรียบเทียบสมรรถนะของระบบ ตัวอย่างเช่น สมมติให้ระบบ 2 ระบบ มีสมรรถนะในรูปของอัตราความผิดพลาดของข้อมูลวัดที่ด้านขาออกของวงจรตรวจหา (Detector) เท่ากัน ถ้าระบบใดใช้กำลังในการส่งข้อมูลน้อยกว่า นั่นคือใช้อัตราส่วนของพลังงานสัญญาณต่อสัญญาณรบกวนน้อยกว่า ก็ถือว่าระบบนั้นมีประสิทธิภาพมากกว่า ในทางปฏิบัติอัตราส่วนของพลังงานสัญญาณต่อสัญญาณรบกวนสามารถที่จะนิยามได้หลายลักษณะ ทั้งนี้ขึ้นอยู่กับเงื่อนไขที่กำหนดมาให้ เช่น ตำแหน่งที่ใช้เป็นจุดอ้างอิงในการวัดอัตราส่วนของพลังงานสัญญาณต่อสัญญาณรบกวนของระบบ นอกจากนี้ในการวิเคราะห์ระบบการประมวลผลสัญญาณของระบบ องค์ประกอบของสัญญาณรบกวนที่ใช้ในการคำนวณหาค่าอัตราส่วนของพลังงานสัญญาณต่อสัญญาณรบกวนอาจจะประกอบด้วย สัญญาณรบกวนแบบบวก (Additive Noise) และสัญญาณรบกวนระบบรวมกันก็ได้ โดยจะขึ้นอยู่กับข้อกำหนดของการใช้งานแต่ละงานประยุกต์

2.5.5 ค่าระยะห่างต่ำสุด (Minimum Distance : d_{min}) คือ ค่าที่พิจารณาจากเลขหนึ่งหรือเลขศูนย์ของเมทริกซ์พาริตีเช็กของแต่ละแถวที่อยู่ติดกันเหมือนกันจุดต่อจุดมีกี่จุด แล้วนำค่าที่ได้เป็นค่าน้อยสุด มาเป็นค่าระยะห่างต่ำสุด หรือเรียกว่าค่าระยะห่างแฮมมิง พิจารณาจากเมทริกซ์พาริตีเช็กดังสมการที่ (2.1)

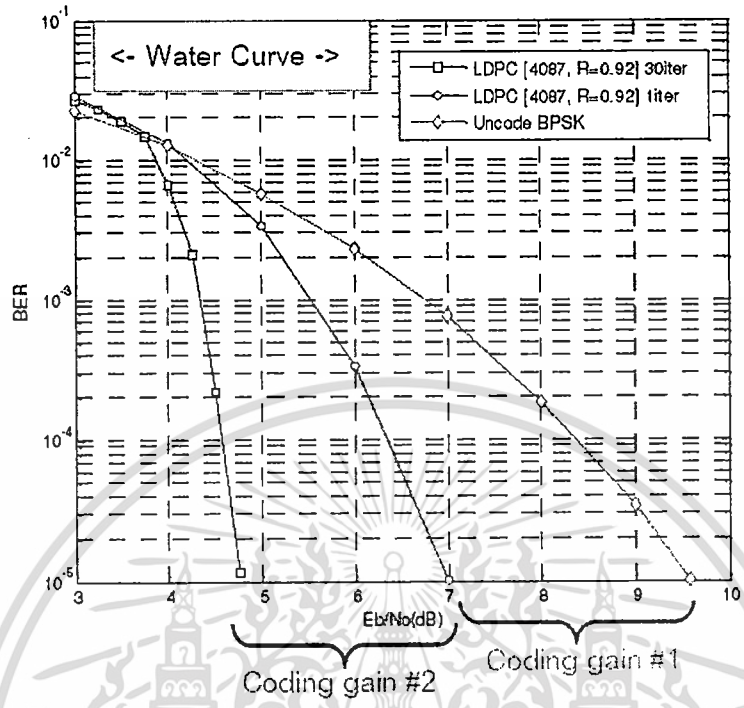
สำหรับเมทริกซ์พาริตีเช็กสามารถแก้ไขข้อผิดพลาดได้ e = $(d_{min}-1)/2$ =(4-1)/2 = 1.5 และสามารถตรวจเช็กข้อผิดพลาดได้t = d_{min}-1 = 4-1 = 3 โดยd_{min} หาได้จากระยะห่างแฮมมิ่งหรือระยะห่างต่ำสุดของเมทริกซ์พาริตีเช็ก ได้แก่ ข้อมูลแถว 1 และข้อมูลแถว 2 มีตัวเลขหนึ่งหรือศูนย์ตรงกันอยู่ 4 ตำแหน่ง ข้อมูลแถว 2 กับข้อมูลแถว 3 ก็มีตรงกันอยู่ 4 ตำแหน่งพิจารณาดูแล้วระยะห่างต่ำสุดคือ 4 (d_{min}=4)

2.5.6 ค่าผลต่างของอัตราส่วนของพลังงานสัญญาณต่อสัญญาณรบกวนระหว่างรหัสที่ยังไม่ผ่านการเข้ารหัสกับรหัสที่ผ่านการเข้ารหัส (Coding gain) เกนของรหัส G หาได้จากผลต่างของอัตราส่วนของพลังงานสัญญาณต่อสัญญาณรบกวนของกรณีที่ไม่มีการเข้ารหัสกับกรณีที่มีการเข้ารหัส ณ ระดับความน่าจะเป็นของบิตผิดพลาดของระบบใด ๆ ดังสมการที่ (2.23) และแสดงดังรูปที่ 2.14

$$G = (E_b/N_0)_u(dB)-(E_b/N_0)_c(dB) \qquad (2.23)$$

โดย $(E_b/N_0)_u$= อัตราส่วนของพลังงานสัญญาณต่อสัญญาณรบกวนในกรณีที่ไม่มีการเข้ารหัส

$(E_b/N_0)_c$= อัตราส่วนของพลังงานสัญญาณต่อสัญญาณรบกวนในกรณีที่มีการเข้ารหัส



รูปที่ 2.14 แสดงค่าผลต่างของอัตราส่วนของพลังงานสัญญาณต่อสัญญาณรบกวนระหว่างรหัสที่ยังไม่ผ่านการเข้ารหัสกับรหัสที่ผ่านการเข้ารหัส

บทที่ 3
ทฤษฎีและงานวิจัยเกี่ยวข้อง

ในบทนี้จะกล่าวถึงเนื้อหาที่เป็นทฤษฎีและงานวิจัยที่เกี่ยวข้องกับงานวิจัยในครั้งนี้ ได้แก่ คณิตศาสตร์พื้นฐานและการเข้ารหัสช่องสัญญาณ เมจิกสแควร์สถิติประยุกต์กับงานวิจัยงานวิจัยเกี่ยวกับรหัสแอลดีพีซีด้านการออกแบบเมทริกซ์พาริตีเช็ก งานวิจัยเกี่ยวกับรหัสแอลดีพีซีกลุ่มเลขหนึ่งไม่คงที่ กลุ่มขนาดบล็อกสั้น กลุ่มเข้ารหัสและกลุ่มถอดรหัสโดยมีรายละเอียดดังต่อไปนี้

3.1 คณิตศาสตร์พื้นฐานและการเข้ารหัสช่องสัญญาณ

3.1.1 คณิตศาสตร์พื้นฐานสำหรับการเข้ารหัสช่องสัญญาณ

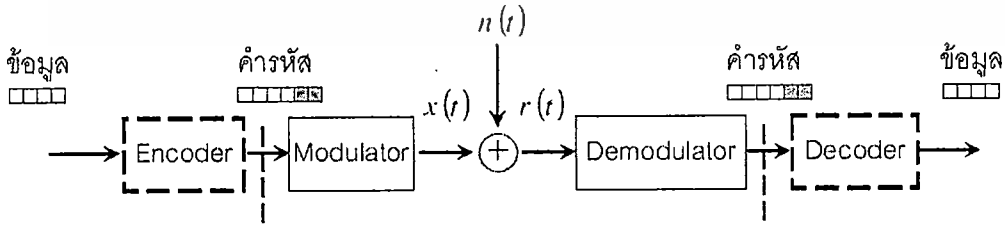
ฟิลด์ (Fields) เป็นเซตที่มีการดำเนินการบวกและคูณ เช่น การเปลี่ยนกลุ่มภายใต้การบวกฟิลด์มีคุณสมบัติปิดภายใต้การคูณและเซตของสมาชิกที่ไม่เป็นศูนย์สามารถเปลี่ยนกลุ่มภายใต้การคูณได้ภายใต้การดำเนินการบวก เอกลักษณ์ คือ 0 และค่าส่วนกลับ (Inverse) การบวกของa คือ –a ภายใต้การดำเนินการคูณ เอกลักษณ์คือ 1 และค่าส่วนกลับการคูณของ a คือ a^{-1} นอกจากนี้ยังมีการลบคือ (a-b) = a+ (-b) การหาร (a/b) = ab^{-1} กฎการกระจาย (a+b)c = ac+b cสำหรับ a,b,cทั้งหมดในฟิลด์ F ซึ่งมีสมาชิก q เรียกว่า ฟิลด์ที่มีจำนวนจำกัด หรือ Galois field แทนด้วย GF(q) โดยจำนวนสมาชิกที่น้อยที่สุดที่ฟิลด์สามารถมีได้คือ 2 (0,1) เรียกว่า Binary Field ซึ่งจะแทนด้วย GF(2)

โพลิโนเมียล (Polynomials) เป็นสัญลักษณ์ทางพีชคณิตซึ่งเขียนจากโมโนเมียลหนึ่งจำนวน หรือหลาย ๆ จำนวนด้วยการบวกหรือการลบของโมโนเมียลเหล่านั้น เช่น $4ab+2b^2-5b^3+4b^4$ สำหรับการบวกและการคูณของจำนวนนับเล็ก ๆ (Finite Number) สามารถทำได้ก็ต่อเมื่อจำนวนเลขเหล่านี้เป็นกำลัง (Power) ของจำนวนเฉพาะ (Prime Number) เมื่อเป็นเช่นนี้สามารถใช้กฎเกณฑ์ธรรมดาของคณิตศาสตร์ในการบวกและคูณได้ กรณีของการใช้รหัสตัวเลขสำหรับการติดต่อสื่อสารซึ่งจะมีสัญลักษณ์ “0” กับ “1” การบวกและการคูณแบบมอดุโล -2(Modulo-2) โดยถือว่า 2 เท่ากับ 0 ดังนั้น 1+1 = 0 และ 1 = -1 สัญลักษณ์ 0 กับ 1 รวมกับการบวกและการคูณแบบมอดุโล -2 จะรวมกันเป็นฟิลด์ ซึ่งเรียกว่า ไบนารีฟิลด์ (Binary Field) เขียนเป็น GF(2) การคำนวณโพลิโนเมียล (Polynomial) ที่มีสัมประสิทธิ์เป็น 1 หรือ 0 สำหรับเลขจำนวนจริงถ้ามี λ เป็นรากของโพลีโนเมียล f(x) นั่นคือ $f(\lambda) = 0f(x)$ จะถูกหารด้วย $x-\lambda$ ลงตัว กฎเกณฑ์นี้ยังคงเป็นจริงสำหรับ f(x) ซึ่งมีสัมประสิทธิ์เป็นเลขฐานสอง ให้ $f(x) = x^4+x^3+x^2+1$ ดังนั้น $f(1) = 1^4+1^3+1^2+1 = 1+1+1+1 = 0$ นั่นคือ f(x) หารด้วย x–1 ซึ่งเท่ากับ x+1) ได้ลงตัว

ฟิลด์ที่มีจำนวนจำกัด (Galois field) ฟิลด์ที่มีสัญลักษณ์อยู่จำนวน 2^m ตัวเรียกว่าเป็น GF(2^m) ซึ่งมีความสำคัญมากในการใช้ศึกษาพวกรหัสวน (Cyclic Code) ในทางปฏิบัติจะถูกนำไปใช้ในการถอดรหัสบีซีเอช (BoseChaudhuriHocquenghem Codes: BCH) และใช้เป็นสัญลักษณ์ในรหัสรีดโซโลมอนสัญลักษณ์ 2^m หาได้โดยโพลิโนเมียล p(x) มีอันดับเท่ากับ m ซึ่งมี 2 สัญลักษณ์ ให้ α เป็นรากของโพลิโนเมียล กล่าวคือ $p(\alpha)=0$ และกำหนดว่า 2=0 ในฟิลด์ที่มีสองสัญลักษณ์ ถ้าหากเลือกโพลิโนเมียล p(x) ที่เหมาะสมจะพบว่ากำลังของ α จะมีค่าถึง 2^m-2 ที่แตกต่างกัน โดยที่กำลังของ α ที่ 2^m-1 จะกลับมาเท่ากับ 1 ใหม่ ดังนั้น $0,1,\alpha,\alpha^2, \ldots, \alpha^{2^m-2}$ จะเป็นเซตของ 2^mสมาชิกฟิลด์แต่ละสมาชิกสามารถเขียนได้ด้วยผลรวมของพวกสมาชิกที่เป็น $1,\alpha,\alpha^2, \ldots, \alpha^{m-1}$ ตัวอย่างสำหรับ m=4, $p(x)=x^4+x+1$ ให้ $p(\alpha) = \alpha^4+\alpha+1 = 0$ จะได้ $\alpha^4=\alpha+1$ α จะมีกำลังที่ให้ค่าแตกต่างกันถึงกำลังที่ $2^4-2=14$ โดยสามารถกระจายเทอมกำลังต่าง ๆ ของ α ได้สมาชิก α นี้เรียกว่าไพรมีทีฟอีลิเมนท์ของ GF(2^m) โดยทั่ว ๆ ไปสมาชิกใด ๆ ของ GF(2^m) ที่มีกำลังสร้างได้ไม่เป็นศูนย์ของ GF(2^m) เรียกว่าไพรมีทีฟ

3.1.2 การเข้ารหัสช่องสัญญาณ

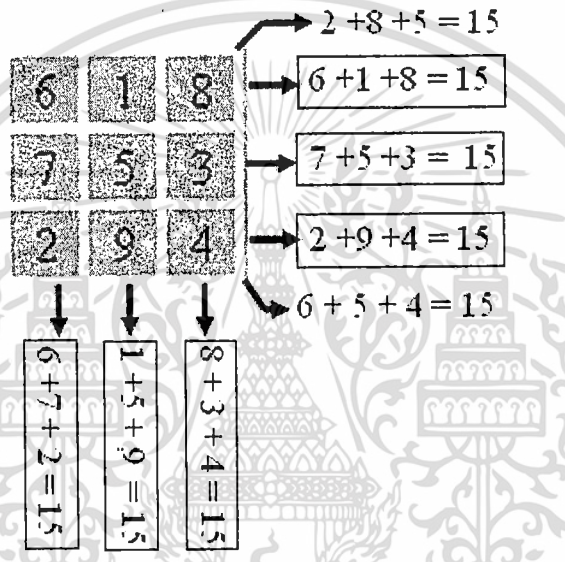
เพื่อให้สัญญาณที่ส่งสามารถส่งไปให้ผู้รับได้ในเนื้อหาที่ตรงกันนั้น จำเป็นต้องใช้เทคนิคการเข้ารหัสแก้ไขข้อผิดพลาด หรือการเข้ารหัสช่องสัญญาณ (Channel Coding) เป็นการเข้ารหัสข้อมูลที่จะส่งออกไปยังช่องสัญญาณเพื่อให้ข้อมูลสามารถที่จะตรวจจับ (Detection) และสามารถแก้ไขความผิดพลาด (Correction) ของข้อมูลที่ส่งออกไปได้ ซึ่งข้อมูลที่เข้ารหัสจะมีขนาดเพิ่มขึ้น ทำให้ต้องใช้แบนด์วิธเพิ่มขึ้น และทำให้อัตราการส่งลดลง โดยมีวิธีการแก้ไขความผิดพลาด 2 วิธี คือ Automatic Repeat Request : ARQ เป็นการแก้ไขความผิดพลาดโดยการตรวจจับความผิดพลาดที่ภาครับ เมื่อเกิดความผิดพลาดขึ้นก็จะร้องขอกลับมาที่ต้นทางให้ส่งข้อมูลกลับมาใหม่ และ Forward Error Correction : FEC เป็นการแก้ไขความผิดพลาดที่ตรวจจับได้ที่ภาครับโดยแก้ไขข้อมูลให้ถูกต้องที่ภาครับโดยไม่มีการส่งข้อมูลกลับมาที่ภาคส่งดังแสดงในรูปที่ 3.1



รูปที่ 3.1 การเข้า-ถอดรหัสช่องสัญญาณ

3.2 เมจิกสแควร์

เมจิกสแควร์ (Magic Squares) หรือจัตุรัสมหัศจรรย์คือการนำตัวเลข 1, 2, 3, ...,n มาจัดวางในลักษณะเป็นแถวและหลัก (Array) ที่มีจำนวนแถวและจำนวนหลักเท่ากัน (จัตุรัส) โดยเมื่อนำตัวเลขทั้งหมดในแถว หรือ หลัก หรือตามแนวเส้นทแยงมุมมาบวกกันจะได้ผลบวกเดียวกัน ตัวอย่างเช่น เมจิกสแควร์แบบธรรมดาขนาด 3×3 จะมีการจัดวางตัวเลขดังรูปที่ 3.2 ซึ่งผลรวมของตัวเลขในแถวหลัก หรือแนวเส้นทแยงมุมจะมีค่าเท่ากับ 15 เป็นต้น



รูปที่ 3.2 เมจิกสแควร์แบบธรรมดาขนาด 3×3

3.2.1 ประวัติความเป็นมาของเมจิกสแควร์ เมจิกสแควร์ได้ถูกศึกษาและมีผู้เสนอไว้มากว่าสามพันปีแล้ว จากการค้นพบหลักฐานที่มีการบันทึกเกี่ยวกับเรื่องนี้ในประเทศจีนเมื่อประมาณ 2,200 ปีก่อนคริสตกาล ในศตวรรษที่ 9 นักดาราศาสตร์ชาวอาหรับใช้เมจิกสแควร์ในทางโหราศาสตร์ ต่อมาเมื่อประมาณ 1,300 ปีหลังคริสตกาล เมจิกสแควร์ได้แพร่เข้าไปในโลกตะวันตกซึ่งได้มีการจารึกในหลักศิลาโดยศิลปินชาวเยอรมันชื่อ Albrecht Dürer โดยระบุปีที่จารึก คือ ปี 1514 และเนื่องจากหลักการของเมจิกสแควร์สามารถเข้าใจได้ง่ายมาก ทำให้มันมีความน่าสนใจสำหรับการทำเลขปริศนาและนักคณิตศาสตร์สมัยนั้น แม้แต่นักคณิตศาสตร์ชื่อ Benjamin Franklin ที่ได้ทำการสร้างเมจิกสแควร์ขนาด 8×8 โดยทำแบบไม่จริงจังนัก แต่ยังได้รับความสนใจเป็นจำนวนมาก

ประเทศไทยก็มีประวัติเกี่ยวข้องเกี่ยวกับวิธีการสร้างเมจิกสแควร์ โดยโดนัลด์ อี คนุธเป็นผู้เชี่ยวชาญทางวิทยาการคอมพิวเตอร์ที่โด่งดัง ได้กล่าวอ้างไว้ในหนังสือที่มีชื่อของท่านเล่มหนึ่ง คือ “The Art of Computer Programing” ข้อความว่า “เมจิกสแควร์ ถูกนำมาจากสยามไปสู่ฝรั่งเศส โดย เดอ ลา

ลูแบร์ (De la Loubère) ในปี 1687 (พ.ศ. 2230)" อัลกอริทึมนี้ ได้สร้างความตื่นเต้นในยุโรปยุคนั้นไม่น้อย จนเกิดมีการค้นคว้าเรื่องตารางอาถรรพ์รูปแบบต่าง ๆ ขึ้นมาอีกมากมาย และมีการเขียนตำราเกี่ยวกับอัลกอริทึมชนิดนี้หลายเล่ม และมีผู้เชื่อว่า ลูแบร์ คงมาพบตารางอาถรรพ์นี้จากการลงตัวเลขในผ้ายันต์ หรือ "เลขยันต์" ในสมัยกรุงศรีอยุธยา

3.2.2 อัลกอริทึมของการสร้างเมจิกสแควร์วิธีการสร้างเมจิกสแควร์แบบธรรมดา สรุปเป็นประโยคง่าย ๆ ได้ดังนี้

1) ต้องคิดว่าขอบตารางที่อยู่ตรงข้ามกัน อยู่ติดกันเป็นเสมือนว่าตารางม้วนมาบรรจบกันที่ขอบ คือขอบบนต่อกับขอบล่าง ขอบซ้ายต่อกับขอบขวา

2) ลงเลข 1 ที่ช่องกลางแถวบนสุดของตารางก่อน

3) เดินขึ้น 1 ช่อง เลี้ยวซ้าย 1 ช่อง ถ้าช่องตารางนี้ว่างอยู่ ให้ลงเลขถัดไป เช่น 2, 3,4, ... เดินหน้าเช่นนี้เรื่อย ๆ ถ้าช่องตารางยังว่าง เนื่องจากเราเริ่มจากแถวบนสุด การเดินขึ้น 1 ช่องจึงมาอยู่ที่ช่องกลางของแถวล่างสุด จากนั้นก็เดินซ้ายไปอีก 1 ช่อง จึงลงเลข 2 สังเกตดัง n=3ดังรูปที่ 3.2

4) กรณีเดินตามข้อ 3 แล้ว ช่องตารางมีตัวเลขที่ลงไว้ก่อน ให้ถอยกลับที่เดิม แล้วเดินลงล่าง 1 ช่อง ลงตัวเลขถัดไปในช่องนั้น ดูตำแหน่งเลข 4 ในรูจากนั้นเดินต่อไปอีกตามข้อ 3 เลขสุดท้ายที่ทำให้ตารางเต็ม คือ n ยกกำลัง 2 เช่น n=3 ตัวเลขสุดท้ายคือ 9 ซึ่งตารางจะเต็มพอดี

สรุปเป็นอัลกอริทึมทางคณิตศาสตร์ในรูปทั่วไปคือเมื่อต้องการสร้างเมจิกสแควร์ขนาด $z \times z$ ผลรวมของตัวเลขของเมจิกสแควร์ในแนวแถวหรือหลักหรือเส้นทแยงมุมจะมีค่าเท่ากับ $z \times M$ โดย M คือจำนวนตัวเลขที่นำมาบวกกัน เขียนเป็นประโยคสัญลักษณ์ดังสมการที่ (3.1)

$$\sum_{i=1}^{z^2} i = z \times M \qquad (3.1)$$

ดังนั้นจากความสัมพันธ์ข้างต้นสามารถคำนวณหาค่า Mได้ดังสมการที่ (3.2)

$$M = [z(z^2 + 1)]/2 \qquad (3.2)$$

ปัจจุบันเมจิกสแควร์แบ่งตามลักษณะการสร้างได้เป็น 4 กลุ่ม ดังนี้

1) เมจิกสแควร์ที่เกี่ยวข้องกับดวงดาวทางดาราศาสตร์ (Magic Squares Associated to the Astrological Planets)

2) เมจิกสแควร์ที่มีขนาดเป็นเลขคี่ (Odd Order)

3) เมจิกสแควร์ที่มีขนาดหารด้วยสี่ลงตัว (Doubly Even Order)

4) เมจิกสแควร์ที่มีขนาดหารด้วยสี่ไม่ลงตัว (Singly Even Order)

ซึ่งมีที่มาและรายละเอียดดังนี้

1) เมจิกสแควร์ที่เกี่ยวข้องกับดวงดาวทางดาราศาสตร์ประมาณปี 1510 ผู้เขียนชื่อ Heinrich Cornelius Agrippa ได้เขียนหนังสือชื่อ De OccultaPhilosophia ซึ่งมีเนื้อหาเกี่ยวกับเรื่องลี้ลับและงานของหมอผีที่ชื่อ MarsilioFicinoและPicodellaMirandola เนื้อหาของหนังสือนี้ เขาได้บรรยายเกี่ยวกับความลี้ลับของเมจิกสแควร์ไว้ด้วย โดยมีการเชื่อมโยงเมจิกสแควร์แต่ละขนาดเข้ากับดวงดาวในอวกาศ ซึ่งประกอบด้วย Saturn, Jupiter, Mars, Sol, Venus, Mercury, and Luna [26] แสดงดังรูปที่ 3.3

4	14	15	1
9	7	6	12
5	11	10	8
16	2	3	13

ก) Jupiter

11	24	7	20	3
4	12	25	8	16
17	5	13	21	9
10	18	1	14	22
23	6	19	2	15

ข) Mars

6	32	3	34	35	1
7	11	27	28	8	30
19	14	16	15	23	24
18	20	22	21	17	13
25	29	10	9	26	12
36	5	33	4	2	31

ค) Sol

22	47	16	41	10	35	4
5	23	48	17	42	11	29
30	6	24	49	18	36	12
13	31	7	25	43	19	37
38	14	32	1	26	44	20
21	39	8	33	2	27	45
46	15	40	9	34	3	28

ง) Venus

8	58	59	5	4	62	63	1
49	15	14	52	53	11	10	56
41	23	22	44	45	19	18	48
32	34	35	29	28	38	39	25
40	26	27	37	36	30	31	33
17	47	46	20	21	43	42	24
9	55	54	12	13	51	50	16
64	2	3	61	60	6	7	57

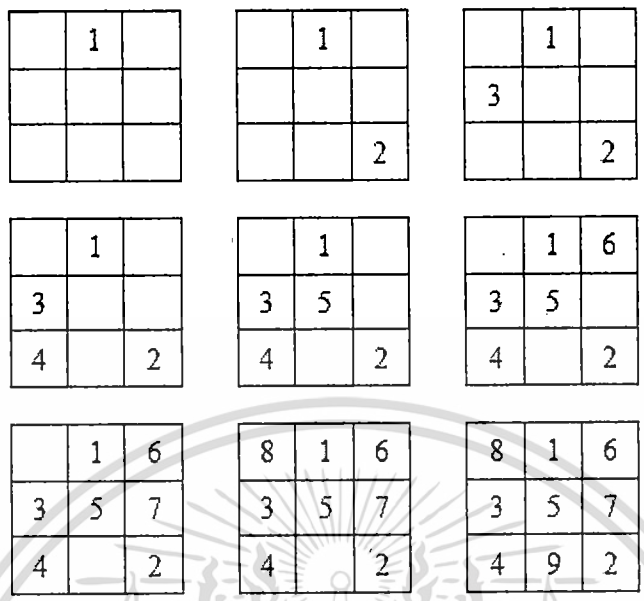
จ) Mercury

37	78	29	70	21	62	13	54	5
6	38	79	30	71	22	63	14	46
47	7	39	80	31	72	23	55	15
16	48	8	40	81	32	64	24	56
57	17	49	9	41	73	33	65	25
26	58	18	50	1	42	74	34	66
67	27	59	10	51	2	43	75	35
36	68	19	60	11	52	3	44	76
77	28	69	20	61	12	53	4	45

ฉ) Luna

รูปที่ 3.3 เมจิกสแควร์เกี่ยวข้องกับดวงดาวทางดาราศาสตร์

2) เมจิกสแควร์ที่มีขนาดเป็นเลขคี่ดังได้กล่าวไว้แล้วตอนต้นว่า เมจิกสแควร์ ถูกนำออกไปจากประเทศไทย (สมัยนั้นเรียกว่าประเทศสยาม) ไปสู่ฝรั่งเศส โดย เดอ ลา ลูแบร์ (De la Loubère) ในปี 1687ซึ่งต่อมาได้มีการตั้งชื่อวิธีการสร้างเมจิกสแควร์ที่มีขนาดที่เป็นเลขคี่ว่า "Siamese or staircase method" ดังที่ เดอ ลา ลูแบร์ ได้นำไปเสนอ ซึ่งมีวิธีการสร้างโดยสรุป คือ เริ่มจากการเติมเลข 1 ตรงกลางของหลักของแถวแรก จากนั้นให้เติมตัวเลขถัดไปที่ละตัวในตำแหน่งเฉียงขึ้นทางขวาครั้งละหนึ่งตัวเลข ในกรณีถ้าตำแหน่งที่ต้องการเติมได้มีการเติมไปแล้วก่อนหน้า ให้เติมตัวเลขนั้นในแนวดิ่งลงมาหนึ่งตำแหน่ง แล้วจึงทำซ้ำการเติมตัวเลขในตำแหน่งเฉียงขึ้นทางขวาต่อไป ในกรณีที่ตำแหน่งต่อไปออกนอกเมจิกสแควร์ให้พิจารณาเปรียบเสมือนว่าเมจิกสแควร์เป็นแผ่นกระดาษที่สามารถม้วนได้ ทั้งแนวนอนและแนวดิ่ง แล้วให้เติมตัวเลขถัดไปลงในตำแหน่งของแถวที่หลักที่ม้วนมาบรรจบกัน [27] ตัวอย่างการเติมตัวเลขด้วยวิธีนี้แสดงดังรูปที่ 3.4



รูปที่ 3.4 เมจิกสแควร์มีขนาดเป็นเลขคี่

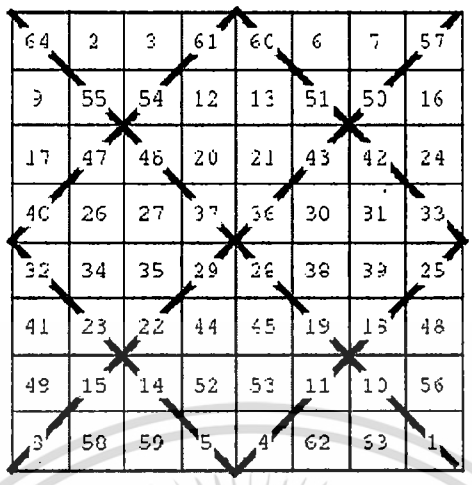
3) เมจิกสแควร์ที่มีขนาดหารด้วยสี่ลงตัวการสร้างเมจิกสแควร์ขนาดที่หารด้วยสี่ลงตัว (Doubly Even Order) เสนอโดยคอเนเรียสเอกริบบา (Cornelius Agrippa) มีขั้นตอนไม่ซับซ้อนเพียงสามขั้นตอน คือ เริ่มจากการเติมตัวเลขทั้งหมดตามลำดับทีละแถวโดยเริ่มจากแถวแรกซ้ายไปขวา (ดังรูปที่ 3.5 ก) ตัวเลขที่ไม่อยู่ในแนวเส้นทแยงมุมให้คงที่ไว้ส่วนตัวเลขที่อยู่ในแนวเส้นทแยงมุมให้พลิกกลับด้านกันในแนวเส้นทแยงมุมเดิม (ดังรูปที่ 3.5 ข) สำหรับเมจิกสแควร์ที่มีขนาดตั้งแต่8ขึ้นไป (และหารด้วย 4 ลงตัว) ให้ดำเนินการดังลักษณะตามรูปที่ 3.5 ค

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16

ก)

16	2	3	13
5	11	10	8
9	7	6	12
4	14	15	1

ข)



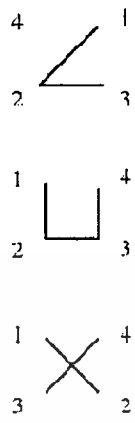
ค)

รูปที่ 3.5 เมจิกสแควร์มีขนาดหารด้วยสี่ลงตัว

4) เมจิกสแควร์ที่มีขนาดหารด้วยสี่ไม่ลงตัวสำหรับเมจิกสแควร์ที่มีขนาดหารด้วยสี่ไม่ลงตัว (Singly Even Order) ปัจจุบันมีวิธีสร้างอยู่ 2 วิธี คือ วิธีของ Ralph Strachey และวิธีของ LUX การสร้างเมจิกสแควร์ด้วยวิธีของ Ralph Strachey เริ่มด้วยการแบ่งเมจิกสแควร์ออกเป็นสี่ส่วนเท่า ๆ กัน เช่น เมจิกสแควร์ขนาด 6×6 สามารถแบ่งเป็นเมจิกสแควร์ขนาด 3×3 ได้ 3 เมจิกสแควร์ ดังรูปที่ 3.6 ก) จากนั้นให้ดำเนินการแต่ละเมจิกสแควร์ย่อยด้วยวิธีของ De la Loubère ที่ใช้กับเมจิกสแควร์ที่มีขนาดเป็นเลขคี่ อีกวิธีหนึ่งเติมตัวเลขลงในเมจิกสแควร์ที่มีขนาดหารด้วยสี่ไม่ลงตัวซึ่งค้นพบโดย J. H. Conway และถูกเรียกว่า วิธี LUX ซึ่งมาจากลักษณะของลำดับการเติมตัวเลขตามรูป 3.6 ข) นั่นเอง

8	1	6	26	19	24
3	5	7	21	23	25
4	9	2	22	27	20
35	28	33	17	10	15
30	32	34	12	14	16
31	36	29	13	18	11

ก)



68	65	96	93	4	1	32	29	60	57
L		L		L		L		L	
66	67	94	95	2	3	30	31	58	59
92	89	20	17	28	25	56	53	64	61
L		L		L		L		L	
90	91	18	19	26	27	54	55	62	63
16	13	24	21	49	52	80	77	88	85
L		L		U		L		L	
14	15	22	23	50	51	78	79	86	87
37	40	45	48	76	73	81	84	9	12
U		U		L		U		U	
38	39	46	47	74	75	82	83	10	11
41	44	69	72	97	100	5	8	33	36
X		X		X		X		X	
43	42	71	70	99	98	7	6	35	34

ข)

รูปที่ 3.6 เมจิกสแควร์มีขนาดหารด้วยสี่ไม่ลงตัว

3.3 สถิติประยุกต์กับงานวิจัย

3.3.1 การทดสอบการกระจายแบบปกติ (Normality Test) เมื่อมีการสุ่มตัวอย่างจำนวนหนึ่งออกมาจากประชากรที่มีการกระจายแบบปกติ (Normal Distribution) โดยปกติกลุ่มตัวอย่างดังกล่าวจะมีการกระจายเป็นแบบปกติตามประชากรด้วยเช่นกัน แต่นั่นก็ไม่ใช่กฎตายตัว เป็นไปได้ที่ตัวอย่างที่สุ่มออกมามีการกระจายตัวไม่เป็นแบบปกติ (Non-Normal Distribution) ซึ่งนั่นไม่ใช่ประเด็นปัญหา หากว่าไม่ได้นำข้อมูลไปทำการอนุมานกลับไปหาประชากรอีกที เมื่อใดก็ตามต้องนำข้อมูลของกลุ่มตัวอย่างไปอนุมานถึงประชากร จะต้องแน่ใจว่าข้อมูลดังกล่าว มีการกระจายตัวเป็นแบบปกติเสมอ หากไม่เช่นนั้นแล้วการทดสอบสมมติฐาน หรือการอนุมานด้วยเครื่องมือทางสถิติอื่น ก็จะให้ผลคาดเคลื่อน ตั้งแต่น้อยจนถึงไม่อาจยอมรับได้ ขึ้นอยู่กับลักษณะความการกระจายที่ไม่เป็นแบบปกติ เมื่อเป็นเช่นนี้การทดสอบว่าข้อมูลของกลุ่มตัวอย่างที่ได้มานั้นมีการกระจายแบบปกติหรือไม่ จึงเป็นสิ่งจำเป็นที่จะกระทำโดยไม่อาจหลีกเลี่ยงได้โดยการวัดการกระจายของข้อมูลมี 2 ชนิด คือ การวัดการกระจายสัมบูรณ์ (Absolute Variation) กับการวัดการกระจายสัมพัทธ์ (Relative Dispersion) ทั้งสองวิธีต่างกันคือการวัดการกระจายของข้อมูลชุดเดียวกับการวัดการกระจายของข้อมูลทั้ง 2 ชุดขึ้นไปเปรียบเทียบกัน

3.3.2 ส่วนเบี่ยงเบนมาตรฐาน (Standard Division: S.D.) เป็นค่าวัดการกระจายของข้อมูลชุดเดียว โดยวัดระยะห่างของค่าสังเกตแต่ละค่าจากค่าเฉลี่ยซึ่งเรียกว่า ค่าเบี่ยงเบนจากค่าเฉลี่ย ผลรวมของค่าเบี่ยงเบนจากค่าเฉลี่ยของค่าสังเกตทั้งหมดเท่ากับศูนย์ จึงไม่แสดงการกระจายของข้อมูลแต่อย่างใด แต่เมื่อคำนวณหาผลรวมของค่าเบี่ยงเบนโดยไม่คิดเครื่องหมาย บวก ลบ (ใส่เครื่องหมายสัมบูรณ์) แล้ว

หารด้วยจำนวนค่าสังเกตทั้งหมด ค่าที่ได้เรียกว่า ส่วนเบี่ยงเบนเฉลี่ย ใช้วัดการกระจายของข้อมูล เป็นวิธีที่ดี ใช้ค่าสังเกตทุกค่ามาคำนวณ แต่เนื่องจากการใช้เครื่องหมายสัมบูรณ์ยากต่อการปรับสมการด้วยทฤษฎีทางคณิตศาสตร์ จึงเลี่ยงโดยการยกกำลังสองค่าเบี่ยงเบน ดังนั้นค่าเฉลี่ยของค่าเบี่ยงเบนกำลังสองแล้วถอดรากที่สอง ค่าที่ได้เรียกว่า ส่วนเบี่ยงเบนมาตรฐาน ดังสมการที่ (3.3) หรือ (3.4) เป็นค่าที่ใช้วัดการกระจายที่ดีที่สุด เป็นที่ยอมรับและนิยมใช้จากนักสถิติ นักวิจัย และนักวิชาการทั่วไป

การหาส่วนเบี่ยงเบนมาตรฐาน กรณีที่ 1 กรณีข้อมูลไม่ได้จัดเป็นช่วง

$$S.D. = \sqrt{\frac{\sum_{i=1}^{n}(x_i-\bar{x})^2}{n}} \qquad (3.3)$$

โดย x_i = สมาชิกตัวที่ i

$\bar{x}$ = ค่าเฉลี่ยเลขคณิต

n = จำนวนสมาชิกทั้งหมด

หรือ

$$S.D. = \sqrt{\frac{\sum_{i=1}^{n}x_i^2}{n} - (\bar{x})^2} \qquad (3.4)$$

กรณีที่ 2 กรณีข้อมูลจัดเป็นช่วงแจกแจงความถี่ดังสมการที่ (3.5) หรือ (3.6)

$$S.D. = \sqrt{\frac{\sum_{i=1}^{n}f_i(x_i-\bar{x})^2}{\sum_{i=1}^{n}f_i}} \qquad (3.5)$$

เมื่อ f_i = ความถี่ของแต่ช่วง

หรือ

$$S.D. = \sqrt{\frac{\sum_{i=1}^{n}f_ix_i^2}{\sum_{i=1}^{n}f_i} - (\bar{x})^2} \qquad (3.6)$$

3.3.3 สัมประสิทธิ์ของการแปรผัน (Coefficient of variation : C.V.) เป็นการวัดการกระจายสัมพัทธ์ที่นิยมใช้มากที่สุด คืออัตราส่วนของส่วนเบี่ยงเบนมาตรฐาน (S.D.) กับค่าเฉลี่ยเลขคณิต($\bar{x}$)ดังสมการที่ (3.7)

$$C.V. = \frac{S.D.}{\bar{x}} \qquad (3.7)$$

3.4 งานวิจัยรหัสแอลดีพีซีด้านการออกแบบเมทริกซ์พาริตีเช็ก

3.4.1 รหัสแอลดีพีซีแบบอาร์เรย์ (Array LDPC Codes) [6] รหัสแอลดีพีซีแบบอาร์เรย์ถูกสร้างขึ้นครั้งแรกในปีค.ศ. 2000 (2543) โดย J. Fan มีโครงสร้างเมทริกซ์พาริตีเช็กเป็นแบบอาร์เรย์จึงช่วยแก้ปัญหาความซับซ้อนในการสร้างเมทริกซ์พาริตีเช็กอีกทั้งยังมีสมรรถนะที่ใกล้เคียงกับรหัสแอลดีพีซีที่มีโครงสร้างของเมทริกซ์พาริตีเช็กเป็นแบบสุ่มสำหรับรหัสแอลดีพีซีแบบอาร์เรย์อธิบายได้ด้วยพารามิเทอร์ 3 ค่าได้แก่จำนวนเฉพาะ p และจำนวนเต็ม j และ k โดยที่ j, k≤p ได้เมทริกซ์พาริตีเช็กที่มีขนาด jp×kp เขียนได้ดังสมการที่ (3.8)

$$H=\begin{bmatrix} I & I & I & \cdots & I \\ I & \alpha & \alpha^{2} & \cdots & \alpha^{k-1} \\ I & \alpha^{2} & \alpha^{4} & \cdots & \alpha^{2(k-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ I & \alpha^{j-1} & \alpha^{2(j-1)} & \cdots & \alpha^{(j-1)(k-1)} \end{bmatrix}_{(jp\times kp)} \tag{3.8}$$

โดยที่ I คือ เมทริกซ์เอกลักษณ์และ α คือ เมทริกซ์สลับตำแหน่งขนาด p×p ซึ่งเกิดจากการเลื่อนแถวหรือหลักไปทางซ้ายหรือขวาของ I โดยที่กำลังของ α นั่น คือจำนวนครั้งของการเลื่อนแถวของเมทริกซ์สลับตำแหน่งความยาวคำรหัสที่ได้มีค่าเท่ากับ kp ความยาวพาริตีบิตมีค่าเท่ากับ jp และอัตรารหัส R มีค่าดังสมการที่ (3.9)

$$R=1-\left(\frac{p.j-j+1}{p^{2}}\right) \tag{3.9}$$

ตัวอย่าง การสร้างเมทริกซ์ α ขนาด 5×5 จากเมทริกซ์เอกลักษณ์ด้วยวิธีการเลื่อนแถวและมีคุณสมบัติที่สำคัญดังนี้

$$I=\begin{bmatrix}1&0&0&0&0\\0&1&0&0&0\\0&0&1&0&0\\0&0&0&1&0\\0&0&0&0&1\end{bmatrix}_{(5\times5)} \qquad \alpha=\begin{bmatrix}0&1&0&0&0\\0&0&1&0&0\\0&0&0&1&0\\0&0&0&0&1\\1&0&0&0&0\end{bmatrix}_{(5\times5)}$$

$$\alpha^{2}=\begin{bmatrix}0&0&1&0&0\\0&0&0&1&0\\0&0&0&0&1\\1&0&0&0&0\\0&1&0&0&0\end{bmatrix}_{(5\times5)} \qquad \alpha^{3}=\begin{bmatrix}0&0&0&1&0\\0&0&0&0&1\\1&0&0&0&0\\0&1&0&0&0\\0&0&1&0&0\end{bmatrix}_{(5\times5)}$$

$$\alpha^4 = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}_{(5\times5)} \qquad \alpha^5 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \end{bmatrix}_{(5\times5)}$$

สาเหตุที่เมทริกซ์ α มีชื่อว่า เมทริกซ์สลับตำแหน่ง หรือเมทริกซ์ก็เพราะว่า เมื่อคูณกับเมทริกซ์ใด ๆ นั้น ผลลัพธ์ที่ได้จะมีค่าเท่ากับเมทริกซ์นั้นถูกสลับตำแหน่ง นอกจากนี้เมทริกซ์ผกผัน α^{-1} ของเมทริกซ์สลับตำแหน่งมีค่าเท่ากับทรานสโพรส α^T ของเมทริกซ์สลับตำแหน่งนอกจากนี้ J. Fan ยังได้พิสูจน์ให้เห็นว่ารหัสแอลดีพีซีแบบอาร์เรย์ปราศจากลูป 4 และรหัสนี้สามารถใช้อัลกอริทึมการถอดรหัสได้เช่นเดียวกับรหัสแอลดีพีซีชนิดทั่วไปด้วยงานวิจัยของ J.Fan นี้เองได้ช่วยแก้ปัญหาข้อด้อยของรหัสแอลดีพีซีในเรื่องของการสร้างเมทริกซ์พาริตีเช็ก H จากการสุ่มข้อมูลศูนย์หนึ่งการควบคุมจำนวนเลขหนึ่งในแถวและหลักและการหลีกเลี่ยงลูป 4 ได้นำเสนอโครงสร้างของเมทริกซ์พาริตีเช็กอาร์เรย์ เรียกว่า รหัสแอลดีพีซีแบบอาร์เรย์ดังสมการที่ (3.10) และสมการที่ (3.11)

$$H(p,j,k) \triangleq \begin{bmatrix} I & I & I & \cdots & I \\ I & \alpha & \alpha^2 & \cdots & \alpha^{k-1} \\ I & \alpha^2 & \alpha^4 & \cdots & \alpha^{2(k-1)} \\ \vdots & & & \ddots & \vdots \\ I & \alpha^{j-1} & \alpha^{2(j-1)} & \cdots & \alpha^{(j-1)(k-1)} \end{bmatrix}_{(jp\times ip)} \begin{matrix} \} 1 \\ \} 2 \\ \vdots \\ \vdots \\ \} j \end{matrix} \qquad (3.10)$$

$$\underbrace{\qquad\qquad\qquad\qquad}_{1 \quad 2 \quad 3 \quad \cdots \quad k}$$

$$H = \begin{bmatrix} I & I & I & I \\ I & \alpha & \alpha^2 & \alpha^3 \end{bmatrix}_{(2\times4)} \qquad (3.11)$$

เมื่อ I = เมทริกซ์เอกลักษณ์ขนาด p×p

α = เมทริกซ์สลับตำแหน่งขนาดp×p(กำลังคือจำนวนครั้งของการเลื่อน)

c = ความยาวคำรหัส= kp

p = จำนวนบิตพาริตีเช็ก= jp

R = อัตรารหัสมีค่าเท่ากับ= $1-((p\cdot j-j+1)/p^2)$

ผลของงานวิจัย [6]

1) มีสมรรถนะรหัสที่ดีเทียบเท่ากับรหัสแอลดีพีซีที่มีโครงสร้างเมทริกซ์พาริตีเช็กแบบสุ่ม
2) ปราศจากลูป 4
3) มีระดับความผิดพลาดที่ต่ำ
4) สามารถใช้อัลกอริทึมการเข้าและถอดรหัสทั่วไปได้

3.4.2 การประเมินสมรรถนะของรหัสแอลดีพีซี [12] งานวิจัยนี้จัดเป็นการประเมินสมรรถนะของรหัสแอลดีพีซีแบบคงที่โดยมีรูปแบบการออกแบบเมทริกซ์พาริตีเช็ก ดังสมการที่ (3.12) ใช้หลักการของเกาส์ชอดอง จัดรูปให้ค่าแต่ละตัวที่อยู่ใต้แนวเส้นทแยงมุมเป็น 0 จะได้

$G = [\, IP \,]$ เมื่อ P โดยทั่วไปจะไม่หนาแน่นต่ำหรือมีจำนวนเลขหนึ่งเพิ่มขึ้น ส่งผลให้การเข้ารหัสมีความซับซ้อน (ความยาวบล็อกยาวยิ่งซับซ้อนมาก)

$$H = [\, A \;\; I_{n-k} \,] \tag{3.12}$$

เมื่อ A = เมทริกซ์Binary ที่มีขนาด (n-k)×k

I_{n-k} = เมทริกซ์เอกลักษณ์ขนาด(n-k)×(n-k)

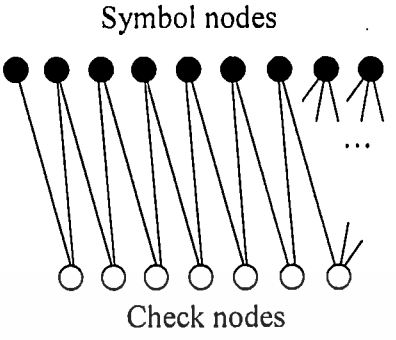
G = $[\, I_k A^T \,]$

โดยกำหนดพารามิเตอร์ N, K, W_cที่แตกต่างกัน 3 ชุด คือ 1) N=20, K=10, W_c=1 2) N=20, K=5, W_c=3 3) N=20, K=4, W_c=4 แต่มีความยาวรหัสเท่ากัน แล้วประเมินค่าอัตราความผิดพลาดของข้อมูลกับค่าอัตรารหัสพบว่ามีผลของงานวิจัยดังนี้

1) การเข้ารหัสมีความซับซ้อนมาก
2) รหัสจะมีประสิทธิภาพดีต้องสร้างคำรหัสยาว และมีจำนวนการวนซ้ำมาก
3) ค่าอัตรารหัสต่ำ และค่าอัตราความผิดพลาดของข้อมูลไม่ดี

ข้อสังเกต การกำหนดโครงสร้างเมทริกซ์พาริตีจะช่วยลดความซับซ้อนในการสร้างได้หรืออาจทำได้โดยหลีกเลี่ยงการสร้าง G ทั้งหมดใช้เป็นกำหนดหนึ่งส่วนและคำนวณอีกหนึ่งส่วน

3.4.3 การออกแบบรหัสแอลดีพีซีแบบไม่คงที่ [13] ได้ออกแบบรหัสแอลดีพีซีที่มีโครงสร้างแบบไม่คงที่ โดยใช้หลักการเลื่อนวนสร้างเมทริกซ์พาริตีเช็กบนหลักการคือต้องไม่เกิดลูปและต้องมีระยะห่างต่ำสุดที่สูง (2 ประเด็นที่มีผลต่อสมรรถนะของรหัส) แก้ปัญหาการเกิดลูปโดยการจัดให้อยู่ในรูปสลับฟันปลา (Zigzag Pattern) ดังรูปที่ 3.7 และเพิ่มค่าระยะห่างต่ำสุดให้สูงโดยการสร้างอัลกอริทึมที่มีขั้นตอนการทำงาน 6 ขั้นตอน ในการสร้างเมทริกซ์พาริตีเช็กย่อย



รูปที่ 3.7 การแก้ปัญหาการเกิดลูปโดยจัดให้อยู่ในรูปฟันปลา

มีรูปแบบเมทริกซ์พาริตีเช็ก ดังสมการที่ (3.13) และ (3.14)

$$H = [\,\tilde{H} | \tilde{\tilde{H}}\,] \tag{3.13}$$

$$H = \underbrace{\left[\begin{array}{ccccc|c} I & I & 0 & 0 & 0 & \\ 0 & I & I & 0 & 0 & \\ 0 & 0 & I & 0 & 0 & \\ 0 & 0 & 0 & I & I & \tilde{\tilde{H}} \\ 0 & 0 & 0 & I & 0 & \\ 0 & 0 & 0 & 0 & I & \end{array}\right]}_{H} \tag{3.14}$$

เมื่อ $\tilde{H}$ = เมทริกซ์จัตุรัสขนาดr×rและเป็น Non-Singular Matrix

$\tilde{\tilde{H}}_{(ps \times pt)}$ = เมทริกซ์พาริตีเช็กขนาด ps×pt

r = ps

n−r = pt

p, s, t = เป็นจำนวนเต็มบวก

πA = เมทริกซ์พาริตีเช็ก

π = เมทริกซ์สลับตำแหน่ง = $(i \bmod p)s + \left|\frac{i}{p}\right|$

A = เมทริกซ์ที่ประกอบด้วย $\Delta^{P}ij$ เมื่อ p×p ดังสมการที่ (3.15)

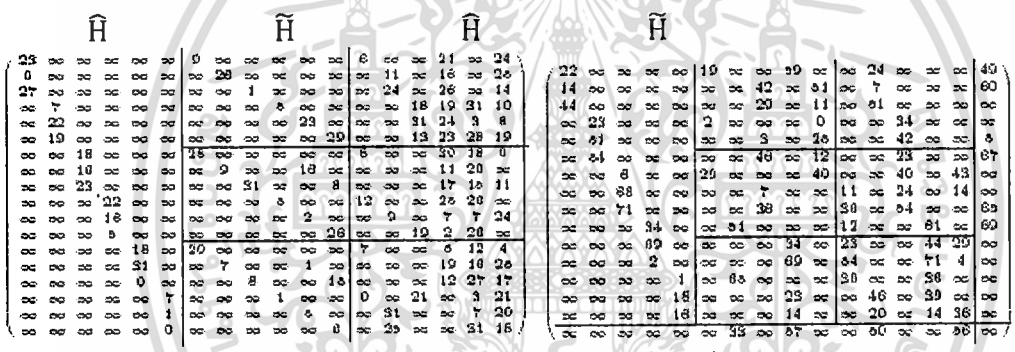
$$\Delta = \begin{pmatrix} 0 & 0 & 0 & & 0 & 1 \\ 1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 1 & 0 & & 0 & 0 \\ & \vdots & & \ddots & \vdots & \\ 0 & 0 & 0 & \cdots & 1 & 0 \end{pmatrix} \tag{3.15}$$

ผลของงานวิจัย [13]

1) มีความถูกต้องแม่นยำสูง (เกิดจากค่าระยะห่างต่ำสุดมีค่าสูง)

2) เมทริกซ์ย่อย πA ขนาดเหมาะสมคือประมาณ 5

3) มีการออกแบบได้เมทริกซ์พาริตีเช็ก H สองแบบที่แตกต่างกัน โดยได้ผลดังรูปที่ 3.8 และสมการที่ (3.16) เพื่อนำมาเปรียบเทียบกับมาตรฐาน IEEE 802.16



แบบที่ 1 แบบที่ 2

รูปที่ 3.8 รูปแบบเมทริกซ์พาริตีเช็กจากงานวิจัย [13]

$$H = \left[\begin{array}{c|c|c} \widetilde{H} & \pi A & \pi A & \pi A \\ \hline & \pi A & \pi A & \pi A \\ \hline & \pi A & \pi A & \pi A \end{array}\right] \tag{3.16}$$

4) มีค่าระดับความผิดพลาดต่ำกว่าของมาตรฐาน IEEE 802.16

5) ที่ค่าอัตราส่วนของพลังงานสัญญาณต่อสัญญาณรบกวนสูง ๆ จะมีผลต่างของอัตราส่วนของพลังงานสัญญาณต่อสัญญาณรบกวน ระหว่างรหัสที่ยังไม่ผ่านการเข้ารหัสกับรหัสที่ผ่านการเข้ารหัส 0.2 dB

6) มีการเข้ารหัสที่ง่ายกว่า

7) อัลกอริทึมที่ออกแบบมีความซับซ้อนมาก

3.5 งานวิจัยรหัสแอลดีพีซีกลุ่มเลขหนึ่งไม่คงที่

3.5.1 รหัสแอลดีพีซีแบบมอดดิฟายอาร์เรย์ [10] จากงานวิจัยของ T.Richardson [7] ได้กล่าวว่าการเพิ่มประสิทธิภาพในการเข้ารหัสและทำให้การเข้ารหัสมีความซับซ้อนในระดับเชิงเส้นสามารถทำได้โดยการแปลงให้เมทริกซ์พาริตีเช็ก (H) มีรูปร่างสามเหลี่ยมดังนั้นในปี ค.ศ. 2002 (2545) E. Eleftheriou [10] ได้เสนอโครงสร้างของเมทริกซ์พาริตีเช็กสำหรับรหัสแอลดีพีซีแบบอาร์เรย์แบบใหม่ ซึ่งมีชื่อเรียกว่ารหัสแอลดีพีซีแบบมอดิฟายอาร์เรย์โดยใช้วิธีการที่เรียกว่าการเลื่อนวนสมรรถนะของรหัสที่ได้ยังมีค่าดีเทียบเท่ากับรหัสแอลดีพีซีที่มีโครงสร้างของเมทริกซ์พาริตีเช็กเป็นแบบสุ่มเมทริกซ์พาริตีเช็กของรหัสนี้สามารถแสดงได้ดังสมการที่ (3.17)

$$H=\begin{bmatrix} I & I & \cdots & I & I & \cdots & \cdots & I \\ 0 & I & \alpha & \cdots & \alpha^{j-2} & \alpha^{j-1} & \cdots & \alpha^{k-2} \\ 0 & 0 & I & \cdots & \alpha^{2(j-3)} & \alpha^{2(j-2)} & \cdots & \alpha^{2(k-3)} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \cdots & I & \alpha^{(j-1)} & \cdots & \alpha^{(j-1)(k-j)} \end{bmatrix}_{(jp\times kp)} \quad (3.17)$$

เมื่อ 0 คือ เมทริกซ์ศูนย์ขนาด p×p รหัสนี้ มีความยาวข้อมูลอินพุตเท่ากับ (k-j)p ความยาวคำรหัสเท่ากับkpและจำนวนบิตพาริตีเช็กเท่ากับjpอัตรารหัสมีค่าเท่ากับ (1-j/ k) เมทริกซ์ข้างต้น ปราศจากลูป 4 มีคุณสมบัติที่ดีในเรื่องระดับความผิดพลาดที่ต่ำและสามารถใช้อัลกอริทึมการถอดรหัสเดิมเช่นเดียวกับรหัสแอลดีพีซีชนิดทั่วไปสังเกตว่ารูปแบบสามเหลี่ยมทำให้การกระจายตัวของเลขหนึ่งเปลี่ยนจากแบบคงที่เป็นแบบไม่คงที่ทำการทรานสโพรสสมการจะได้สมการใหม่ดังสมการที่ (3.18)

$$H_{(m\times n)}C^{T}_{(n\times 1)}=0^{T}_{(m\times 1)} \quad (3.18)$$

เขียนเมทริกซ์Cให้อยู่ในรูปแบบเชิงระบบจะได้สมการหลักดังสมการที่ (3.19) ที่จะใช้ในการเข้ารหัสสำหรับรหัสแอลดีพีซีแบบมอดิฟายอาร์เรย์

$$H_{(m\times n)}\left[\frac{p}{m}\right]^{T}=0^{T}_{(m\times 1)} \quad (3.19)$$

ผลที่ได้คือความซับซ้อนในการเข้ารหัสจะลดลงอย่างมากเมื่อเทียบกับสมการในการเข้ารหัสที่ได้กล่าวมาก่อนหน้านี้เนื่องจากการหาค่าพาริตีบิตนั้นไม่จำเป็นต้องหาค่าเมทริกซ์ผกผัน

ตัวอย่าง การเข้ารหัสสำหรับรหัสแอลดีพีซีแบมอดิฟายอาร์เรย์กำหนดให้พารามิเทอร์p=5, j=4 และ k=5 จากสมการที่ (3.20)

$$H_{(20\times25)}\left[\frac{p_{(20\times1)}}{m_{(5\times1)}}\right] = 0_{(20\times1)} \tag{3.20}$$

เมื่อทำการคูณเมทริกซ์พาริตีเช็กกับเมทริกซ์ คำรหัสจะได้สมการทั้งหมด 20 สมการ จากนั้นจึงใช้สมการเหล่านี้ ในการหาค่าพาริตีบิตด้วยวิธีการมอดุโล 2

$$\begin{aligned} p_{20} + m_3 &= 0 \\ p_{19} + m_2 &= 0 \\ p_{18} + m_1 &= 0 \\ &\vdots \quad \vdots \quad \vdots \\ p_2 + p_7 + p_{12} + p_{17} + m_2 &= 0 \\ p_1 + p_6 + p_{11} + p_{16} + m_1 &= 0 \end{aligned} \tag{3.21}$$

จากสมการที่ (3.21) แสดงให้เห็นว่าความซับซ้อนในการแก้สมการจะเป็นแบบเชิงเส้นอันเป็นผลมาจากการแปลงให้เมทริกซ์พาริตีเช็ก H มีรูปร่างสามเหลี่ยม

[10] ได้นำเสนอโครงสร้างของเมทริกซ์พาริตีเช็กสำหรับรหัสแอลดีพีซีแบบอาร์เรย์ใหม่ ที่เรียกว่ารหัสแอลดีพีซีแบบมอดิฟายอาร์เรย์ โดยใช้วิธีการที่เรียกว่าการเลื่อนวนดังสมการที่ (3.22) และ (3.23)

$$H = \begin{bmatrix} I & I & \cdots I & I & \cdots & \cdots & I \\ 0 & I & \alpha \cdots & \alpha^{j-2} & \alpha^{j-1} & \cdots & \alpha^{k-2} \\ 0 & 0 & I \cdots & \alpha^{2(j-3)} & \alpha^{2(j-2)} \cdots & & \alpha^{2(k-3)} \\ \vdots & \vdots & \vdots \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots \cdots & I & \alpha^{(j-1)} \cdots & & \alpha^{(j-1)(k-j)} \end{bmatrix}_{(jp\times kp)} \tag{3.22}$$

$$H = \begin{bmatrix} I & I & I & I \\ 0 & I & \alpha & \alpha^2 \end{bmatrix}_{(2\times4)} \tag{3.23}$$

เมื่อ 0 = เมทริกซ์ศูนย์ขนาด p×p

I = เมทริกซ์เอกลักษณ์ขนาดp×p

α = เมทริกซ์สลับตำแหน่งขนาดp×p(กำลังคือจำนวนครั้งของการเลื่อน)

m = ความยาวข้อมูล= (k-j)p

c = ความยาวคำรหัส= kp

p = จำนวนบิตพาริตีเช็ก= jp

R = อัตรารหัสมีค่าเท่ากับ= (1-j/k)

ผลของงานวิจัย [10]

1) มีสมรรถนะของรหัสที่ดี เทียบเท่ากับรหัสแอลดีพีซีแบบสุ่ม
2) ปราศจากลูป 4
3) มีระดับความผิดพลาดที่ต่ำ
4) สามารถเข้ารหัสด้วยวิธีอย่างง่ายได้
5) สามารถใช้อัลกอริทึมการถอดรหัสทั่วไปได้

3.5.2 รหัสแอลดีพีซีแบบมอดดิฟายอาร์เรย์ที่มีการสลับบิต [11] ทำการปรับปรุง เมทริกซ์พาริตีเช็กของรหัสมอดิฟายอาร์เรย์ด้วยเมทริกซ์เลื่อนกลับ (Quasi-Cyclic Matrix) ภายใต้แนวคิดที่ทำการเลื่อนวนในระดับบิตด้วยเมทริกซ์เลื่อนกลับซึ่งเป็นส่วนที่เพิ่มเข้าไปในเมทริกซ์พาริตีเช็กของรหัสแอลดีพีซีแบบมอดิฟายอาร์เรย์ให้กลายเป็นรหัสแอลดีพีซีแบบอินเทอร์ลีฟมอดิฟายอาร์เรย์ และสามารถอธิบายได้ด้วยพารามิเทอร์ 3 ค่าได้แก่จำนวนเฉพาะ p จำนวนเต็ม j และ k โดยที่ j, k≤p เมทริกซ์นี้มีขนาด jp×kp เช่นเดียวกับรหัสมอดิฟายอาร์เรย์เมทริกซ์พาริตีเช็กใหม่ที่แสดงดังสมการที่ (3.24)

$$H=\begin{bmatrix} I & I & I\omega & I\omega^2 & I\omega^3 & \cdots & I\omega^{j} \\ 0 & I & \alpha\omega & \alpha^2\omega^2 & \alpha^3\omega^3 & \cdots & \alpha^{(k-2)}\omega^{j} \\ 0 & 0 & I & \alpha^2\omega^2 & \alpha^4\omega^3 & \cdots & \alpha^{2(k-3)}\omega^{j} \\ \vdots & \vdots & \vdots & I & \alpha^3\omega^3 & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & I & \cdots & \alpha^{(j-1)}\omega^{(k-j)} \end{bmatrix}_{(jp\times kp)} \quad (3.24)$$

จากสมการโครงสร้างของเมทริกซ์พาริตีเช็กที่ได้ยังคงรูปร่างสามเหลี่ยมทำให้ประสิทธิภาพในการเข้ารหัสยังคงเดิมเมทริกซ์ 0, α และ I มีคุณสมบัติเช่นเดียวกับมอดิฟายอาร์เรย์ความยาวข้อมูลอินพุตเท่ากับ (k -j)p ความยาวคำรหัสเท่ากับkpและจำนวนพาริตีเช็กเท่ากับjpอัตรารหัสมีค่าเท่ากับ (1-j/ k)

โดยที่ ω คือ เมทริกซ์เลื่อนกลับ ที่สร้างขึ้นจากเมทริกซ์ I โดยทำการเลื่อนวนเป็นวงกลมกับทุกแถวของเมทริกซ์ I ตัวอย่างเมทริกซ์เลื่อนกลับที่สร้างจากเมทริกซ์ I ขนาด 5×5 แสดงดังสมการที่ (3.25)

$$I=\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}_{(5\times5)} \rightarrow \quad \omega=\begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}_{(5\times5)} \quad (3.25)$$

คุณสมบัติที่สำคัญของเมทริกซ์เลื่อนกลับขนาด n×n มีดังนี้

1) เมื่อทำการยกกำลังเท่ากับขนาดของเมทริกซ์จะมีค่าเท่ากับตัวมันเองดังสมการที่ (3.26) และ (3.27)

$$\omega^n = \omega \tag{3.26}$$

$$\omega^5 = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}_{(5\times5)} \rightarrow \omega = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}_{(5\times5)} \tag{3.27}$$

2) เมื่อทำการยกกำลังเมทริกซ์เลื่อนกลับเท่ากับขนาดของเมทริกซ์ลบหนึ่งจะมีค่าเท่ากับเมทริกซ์เอกลักษณ์ดังสมการที่ (3.28) และ (3.29)

$$\omega^{n-1} = I \tag{3.28}$$

$$\omega^4 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}_{(5\times5)} \rightarrow I = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}_{(5\times5)} \tag{3.29}$$

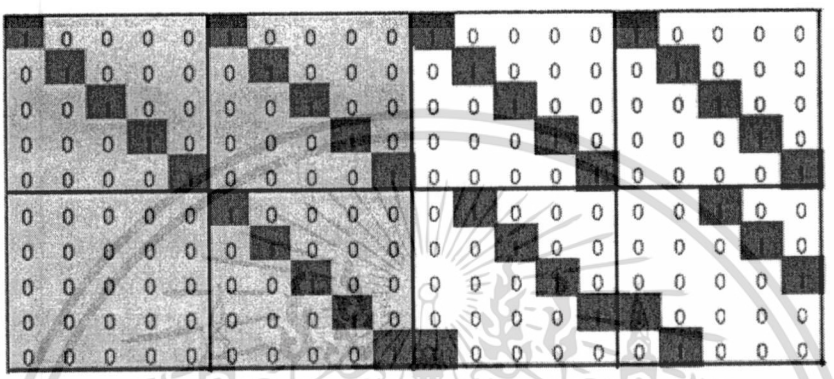
3) รหัสแอลดีพีซีที่นำเสนอมีชื่อว่าอินเทอร์ลีฟมอดิฟายอาร์เรย์เนื่องจากเมื่อนำเมทริกซ์เลื่อนกลับ ω คูณกับเมทริกซ์ใดๆนั้นผลลัพธ์ที่ได้ คือเมทริกซ์นั้นถูกอินเทอร์ลีฟดังสมการที่ (3.30) และ (3.31)

$$\alpha \cdot \omega = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix}_{(5\times5)} \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}_{(5\times5)} \tag{3.30}$$

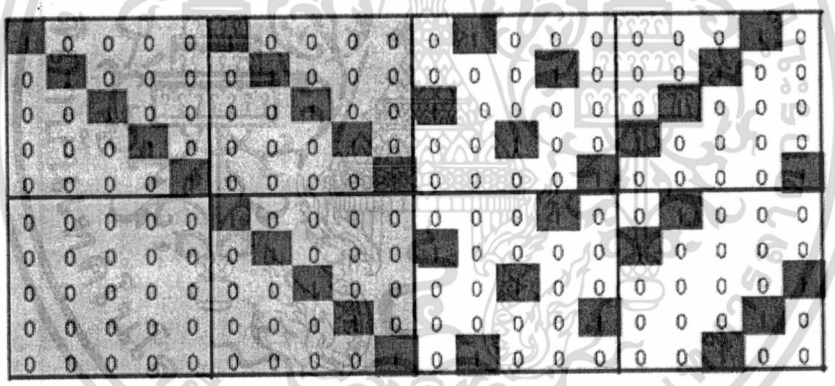
$$\alpha \cdot \omega = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}_{(5\times5)} \tag{3.31}$$

ตัวอย่าง เมทริกซ์พาริตีเช็กสำหรับรหัสแอลดีพีซีแบบมอดิฟายอาร์เรย์ และอินเทอร์ลีฟมอดิฟายอาร์เรย์ กำหนดให้พารามิเทอร์ p=5, j=2 และ k=4 จากสมการจะได้เมทริกซ์พาริตีเช็กดังสมการที่ (3.32)

$$H = \begin{bmatrix} I & I & I & I \\ 0 & I & \alpha & \alpha^2 \end{bmatrix}_{(2\times4)} \rightarrow H = \begin{bmatrix} I & I & I\omega & I\omega^2 \\ 0 & I & \alpha\omega & \alpha^2\omega^2 \end{bmatrix}_{(2\times4)} \qquad (3.32)$$



ก) รหัสแอลดีพีซีแบบมอดิฟายอาร์เรย์



ข) รหัสแอลดีพีซีแบบอินเทอร์ลีฟมอดดิฟายอาร์เรย์

รูปที่ 3.9 พาริตีเช็กสำหรับรหัสแอลดีพีซีแบบมอดิฟายอาร์เรย์และอินเทอร์ลีฟมอดิฟายอาร์เรย์

เมื่อพิจารณาโครงสร้างของเมทริกซ์พาริตีเช็กในรูปที่ 3.9 ก และ ข พบว่าตำแหน่งของเลขหนึ่งในหลักที่หนึ่งถึงสิบ มีค่าเหมือนกันอันเนื่องมาจากตำแหน่งเหล่านี้ไม่มีการคูณด้วยเมทริกซ์เลื่อนกลับ จึงไม่มีความแตกต่างระหว่างรหัสมอดิฟายอาร์เรย์และอินเทอร์ลีฟมอดิฟายอาร์เรย์ ในขณะที่ตำแหน่งของเลขหนึ่งในหลักที่สิบเอ็ดถึงยี่สิบมีค่าต่างกัน คือ เมทริกซ์นั้นถูกอินเทอร์ลีฟ อันสืบเนื่องมาจากผลกระทบจากการถูกคูณด้วยเมทริกซ์เลื่อนกลับ จำนวนของเลขหนึ่งในแต่ละแถวและแต่ละหลักของรหัสทั้งสองแบบมีค่าเท่ากันแต่รหัสทั้งสองแบบปราศจากลูป 4 จากโครงสร้างเมทริกซ์ของเมทริกซ์พาริตีเช็กในรูปที่ 3.9 ก และ ข สามารถอธิบายได้ดังนี้

เลข 1 ในคอลัมน์ที่ 11 บรรทัดที่ 1 รูป ก มารูป ข เกิดจากการเลื่อนมาทางขวามือ 1 ตำแหน่ง

เลข 1 ในคอลัมน์ที่ 12 บรรทัดที่ 2 รูป ก มารูป ข เกิดจากการเลื่อนมาทางขวามือ 2 ตำแหน่ง

เลข 1 ในคอลัมน์ที่ 13 บรรทัดที่ 3 รูป ก มารูป ข เกิดจากการเลื่อนมาทางขวามือ 3 ตำแหน่ง

เลข 1 ในคอลัมน์ที่ 14 บรรทัดที่ 4 รูป ก มารูป ข เกิดจากการเลื่อนมาทางขวามือ 4 ตำแหน่ง

เลข 1 ในคอลัมน์ที่ 15 บรรทัดที่ 5 รูป ก มารูป ข เกิดจากการเลื่อนมาทางขวามือ 5 ตำแหน่ง จึงอยู่ที่เดิม

เลข 1 ในคอลัมน์ที่ 12 บรรทัดที่ 6 รูป ก มารูป ข เกิดจากการเลื่อนมาทางขวามือ 2 ตำแหน่ง

เลข 1 ในคอลัมน์ที่ 13 บรรทัดที่ 7 รูป ก มารูป ข เกิดจากการเลื่อนมาทางขวามือ 3 ตำแหน่ง

เลข 1 ในคอลัมน์ที่ 14 บรรทัดที่ 8 รูป ก มารูป ข เกิดจากการเลื่อนมาทางขวามือ 4 ตำแหน่ง

เลข 1 ในคอลัมน์ที่ 15 บรรทัดที่ 9 รูป ก มารูป ข เกิดจากการเลื่อนมาทางขวามือ 5 ตำแหน่ง จึงอยู่ที่เดิม

เลข 1 ในคอลัมน์ที่ 11 บรรทัดที่ 10 รูป ก มารูป ข เกิดจากการเลื่อนมาทางขวามือ1 ตำแหน่ง

เลข 1 ในคอลัมน์ที่ 16 บรรทัดที่ 1 รูป ก มารูป ข เกิดจากการเลื่อนมาทางขวามือ 3 ตำแหน่ง

เลข 1 ในคอลัมน์ที่ 17 บรรทัดที่ 2 รูป ก มารูป ข เกิดจากการเลื่อนมาทางขวามือ 6 ตำแหน่ง

เลข 1 ในคอลัมน์ที่ 18 บรรทัดที่ 3 รูป ก มารูป ข เกิดจากการเลื่อนมาทางขวามือ 8 ตำแหน่ง

เลข 1 ในคอลัมน์ที่ 19 บรรทัดที่ 4 รูป ก มารูป ข เกิดจากการเลื่อนมาทางขวามือ 12 ตำแหน่ง

เลข 1 ในคอลัมน์ที่ 20 บรรทัดที่ 5 รูป ก มารูป ข เกิดจากการเลื่อนมาทางขวามือ 15 ตำแหน่ง จึงอยู่ที่เดิม

เลข 1 ในคอลัมน์ที่ 18 บรรทัดที่ 6 รูป ก มารูป ข เกิดจากการเลื่อนมาทางขวามือ 8 ตำแหน่ง

เลข 1 ในคอลัมน์ที่ 19 บรรทัดที่ 7 รูป ก มารูป ข เกิดจากการเลื่อนมาทางขวามือ 12 ตำแหน่ง

เลข 1 ในคอลัมน์ที่ 20 บรรทัดที่ 8 รูป ก มารูป ข เกิดจากการเลื่อนมาทางขวามือ 15 ตำแหน่ง จึงอยู่ที่เดิม

เลข 1 ในคอลัมน์ที่ 16 บรรทัดที่ 9 รูป ก มารูป ข เกิดจากการเลื่อนมาทางขวามือ 3 ตำแหน่ง

เลข 1 ในคอลัมน์ที่ 17 บรรทัดที่ 10 รูป ก มารูป ข เกิดจากการเลื่อนมาทางขวามือ 6 ตำแหน่ง

รหัสมอดิฟายอาร์เรย์และอินเทอร์ลีฟมอดิฟายอาร์เรย์จะได้ความสัมพันธ์เป็นสมการ และในแต่ละสมการจะบอกความสัมพันธ์แต่ละบิตระหว่างโหนดสัญลักษณ์และโหนดเช็ก จำนวน 10 สมการ ดังตารางที่ 3.1

ตารางที่ 3.1 สมการที่ได้จากเมทริกซ์พาริตีเช็กสำหรับรหัสแอลดีพีซีแบบมอดิฟายอาร์เรย์และอินเทอร์ลีฟมอดิฟายอาร์เรย์

สมการที่	มอดิฟายอาร์เรย์	อินเทอร์ลีฟมอดิฟายอาเรย์
1	$c_1 + c_6 + c_{11} + c_{16}$	$c_1 + c_6 + c_{12} + c_{19}$
2	$c_2 + c_7 + c_{12} + c_{17}$	$c_2 + c_7 + c_{14} + c_{18}$
3	$c_3 + c_8 + c_{13} + c_{18}$	$c_3 + c_8 + c_{11} + c_{17}$
4	$c_4 + c_9 + c_{14} + c_{19}$	$c_4 + c_9 + c_{13} + c_{16}$
5	$c_5 + c_{10} + c_{15} + c_{20}$	$c_5 + c_{10} + c_{15} + c_{20}$
6	$c_6 + c_{12} + c_{18}$	$c_6 + c_{14} + c_{17}$
7	$c_7 + c_{13} + c_{19}$	$c_7 + c_{11} + c_{16}$
8	$c_8 + c_{14} + c_{20}$	$c_8 + c_{13} + c_{20}$
9	$c_9 + c_{15} + c_{16}$	$c_9 + c_{15} + c_{19}$
10	$c_{10} + c_{11} + c_{17}$	$c_{10} + c_{12} + c_{18}$

ตารางที่ 3.1 แสดงสมการที่ได้จากในแต่ละโหนดเช็กของมอดิฟายอาร์เรย์ และอินเทอร์ลีฟมอดิฟายอาร์เรย์ในแต่ละสมการของโหนดเช็ก จะประกอบด้วยตำแหน่งของโหนดสัญลักษณ์ที่เหมือนกันและต่างกันระหว่างรหัสมอดิฟายอาร์เรย์และอินเทอร์ลีฟมอดิฟายอาร์เรย์ ดังเช่น ในโหนดเช็กที่ 1 ประกอบด้วยโหนดสัญลักษณ์ที่มีค่าเหมือนกันคือ c_1 และ c_6 อันเนื่องมาจากตำแหน่งเหล่านี้ ไม่มีการคูณด้วยเมทริกซ์เลื่อนกลับและโหนดสัญลักษณ์ที่มีค่าต่างกัน คือ รหัสมอดิฟายอาร์เรย์จะประกอบด้วยโหนดสัญลักษณ์ c_{11}, c_{16} ในขณะที่อินเทอร์ลีฟมอดิฟายอาร์เรย์ประกอบด้วยโหนดสัญลักษณ์ c_{12}, c_{19} อันสืบเนื่องมาจากผลกระทบจากการถูกคูณด้วยเมทริกซ์เลื่อนกลับ ซึ่งความต่างนี้เองจะมีผลทำให้กราฟแทนเนอร์ที่ได้ของรหัสทั้งสองแบบมีค่าต่างกัน และเหตุที่ขั้นตอนการถอดรหัสจะอาศัยโครงสร้างกราฟแทนเนอร์มาช่วยในการถอดรหัส จึงมีความเป็นไปได้ที่สมรรถนะอัตราความผิดพลาด บิตของรหัสมอดิฟายอาร์เรย์และอินเทอร์ลีฟมอดิฟายอาร์เรย์จะมีค่าที่ต่างกัน

[11] ได้ปรับปรุงเมทริกซ์พาริตีเช็กของรหัสแอลดีพีซีแบบมอดิฟายอาร์เรย์ ด้วยเมทริกซ์เลื่อนกลับในระดับบิตภายใต้แนวคิดการเลื่อนวนกลายเป็นตัวใหม่เรียกว่า รหัสแอลดีพีซีแบบอินเทอร์ลีฟมอดิฟายอาร์เรย์ ดังสมการที่ (3.33) และ (3.34)

$$H = \begin{bmatrix} I & I & I\omega & I\omega^2 & I\omega^3 & \cdots & I\omega^j \\ 0 & I & \alpha\omega & \alpha^2\omega^2 & \alpha^3\omega^3 & \cdots & \alpha^{(k-2)}\omega^j \\ 0 & 0 & I & \alpha^2\omega^2 & \alpha^4\omega^3 & \cdots & \alpha^{2(k-3)}\omega^j \\ \vdots & \vdots & \vdots & I & \alpha^3\omega^3 & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & I & \cdots & \alpha^{(j-1)}\omega^{(k-j)} \end{bmatrix}_{(jp \times kp)} \quad (3.33)$$

$$H = \begin{bmatrix} I & I & I\omega & I\omega^2 \\ 0 & I & \alpha\omega & \alpha^2\omega^2 \end{bmatrix}_{(2\times 4)} \quad (3.34)$$

เมื่อ 0 = เมทริกซ์ศูนย์ขนาด p×p

I = เมทริกซ์เอกลักษณ์ขนาด p×p

α = เมทริกซ์สลับตำแหน่งขนาด p×p (กำลังคือจำนวนครั้งของการเลื่อน)

ω = เมทริกซ์สลับตำแหน่งที่สร้างจากเมทริกซ์ I เลื่อนวนเป็นวงกลม

m = ความยาวข้อมูล = (k-j)p

c = ความยาวคำรหัส = kp

p = จำนวนบิตพาริตีเช็ก = jp

R = อัตรารหัสมีค่าเท่ากับ = (1-j/k)

ผลของงานวิจัย [11]

1) มีสมรรถนะของรหัสที่ดี เทียบเท่ากับรหัสแอลดีพีซีแบบสุ่ม

2) ปราศจากลูป4

3) มี error floor ที่ต่ำ

4) สามารถเข้ารหัสด้วยวิธีอย่างง่ายได้

5) สามารถใช้อัลกอริทึมการถอดรหัสทั่วไปได้ [11] มีคุณสมบัติที่ดีทำได้เหมือน [10]

ผลการเปรียบเทียบข้อดีระหว่าง [11] กับ [10]

1) มีค่าอัตราความผิดพลาดบิตข้อมูลดีกว่าในทุกอัตรารหัส (0.52, 0.89, 0.92)

2) มีค่าที่ดีกว่าในทุกอัตราส่วนต่อสัญญาณรบกวน สำหรับอัตรารหัส 0.52

3) มีอัตราความผิดพลาดบิตข้อมูลจะต่ำกว่าที่อัตราส่วนของพลังงานสัญญาณต่อสัญญาณรบกวน 4.25 และ 4.75 dB สำหรับอัตรารหัส 0.89 และ 0.92

4) สมรรถนะอัตราความผิดพลาดบิตข้อมูลดีกว่าแบบ [10] เมื่อจำนวนรอบการวนซ้ำเท่ากับ 10 และ 20 รอบ ที่อัตรารหัสสูง

3.6 งานวิจัยรหัสแอลดีพีซีกลุ่มขนาดบล็อกสั้น

3.6.1 รหัสแอลดีพีซีขนาดบล็อกสั้นสำหรับระบบสื่อสารกำลังความถี่ต่ำ [28] ระบบสื่อสาร power-line TWACS ถูกใช้โดยอุปกรณ์ไฟฟ้าสำหรับการสื่อสารระยะไกลร่วมกับเครื่องมือวัดและอุปกรณ์อื่น กลุ่มของข้อมูลที่ใช้มักมีลักษณะเป็นบล็อกสั้นและมีการความต้องการเพิ่มขึ้นสำหรับ bandwidth ที่ต้องการให้มีการตรวจจับและแก้ไขข้อผิดพลาดเพิ่มมากขึ้นด้วย เขาได้ศึกษารหัสแอลดีพีซีเพื่อนำมาใช้แทนรูปแบบการเข้า-ถอดรหัสที่มีการควบคุมข้อผิดพลาดแบบดั้งเดิม บทความของเขาศึกษาความสามารถในการตรวจจับความผิดพลาดของรหัสแอลดีพีซีที่มีอัตราสูงและมีขนาดความยาวบล็อกสั้น ภายใต้เงื่อนไขขอบเขตที่กำหนดขึ้นใหม่เมื่อมีการเพิ่มความหนาแน่นของรหัสจะทำให้เพิ่มลูปสั้น (Short Cycles) ในกราฟแทนเนอร์ของรหัส ซึ่งจะทำให้อัตราการแก้ไขข้อผิดพลาดแย่ลง และเขายังพบอีกว่ามันทำให้อัตราการตรวจไม่พบความผิดพลาดเพิ่มมากขึ้น จึงเป็นข้อดีข้อเสียระหว่างการแก้ไขและการตรวจไม่พบข้อผิดพลาดที่สามารถถูกปรับได้โดยการเปลี่ยนความหนาแน่นของรหัส ด้วยวิธีการนี้ทำให้รหัสแอลดีพีซีสามารถถูกออกแบบเพื่อให้สามารถเพิ่มได้ทั้งการตรวจจับและการแก้ไขความผิดพลาดของรูปแบบการเข้า-ถอดรหัสที่มีอยู่ในปัจจุบัน

สรุปงานนี้ออกแบบรหัสแอลดีพีซีที่มีความซับซ้อนในการเข้ารหัสต่ำ อัตรารหัสสูง และขนาดบล็อกสั้น โดยเลือกกลุ่มของรหัสบล็อก-วนที่มีน้ำหนักแถวคงที่และน้ำหนักหลักไม่คงที่ อย่างไรก็ตามรหัสที่มีขนาดบล็อกสั้นเช่นนี้ มีระยะห่างต่ำสุดที่น้อยอย่างสัมพันธ์กัน ส่งผลให้ความน่าจะเป็นของความผิดพลาดที่ตรวจจับไม่สามารถตรวจจับได้ และได้ทดสอบรหัสที่หลากหลายด้วยน้ำหนักที่สูงขึ้น พบว่าโดยทั่วไปรหัสมีแนวโน้มที่จะมีคุณสมบัติระยะห่างต่ำสุดสูงขึ้นนำมาซึ่งการลดอัตราการตรวจไม่พบข้อผิดพลาด การลดลงนี้อยู่ภายใต้เงื่อนไขด้านขนาดที่เสนอจึงทำให้เพิ่มลูปสั้นในกราฟแทนเนอร์ของรหัสอย่างหลีกเลี่ยงไม่ได้ จึงเป็นการเพิ่มความน่าจะเป็นความผิดพลาดรหัสโดยรวม ผลลัพธ์ก็คือ อัตราความผิดพลาดโดยรวมกับอัตราการตรวจพบความผิดพลาดจะแปรผกผันกัน การศึกษาพบว่ารหัสมีค่าความน่าจะเป็นในการตรวจไม่พบข้อผิดพลาดต่ำกว่ารหัสการตรวจจับความผิดพลาด ซึ่งดีเช่นเดียวกับสมรรถนะเกนโดยรวมซึ่งสัมพันธ์กับรหัสการแก้ไขความผิดพลาดที่มีอยู่ในปัจจุบัน

3.6.2 การสร้างรหัสแอลดีพีซีแบบไม่คงที่ความยาวบล็อกสั้น [29] นำเสนออัลกอริทึมการสร้างสำหรับรหัสแอลดีพีซีไม่คงที่ความยาวบล็อกสั้น บนพื้นฐานการอธิบาย (Interpretation) แบบใหม่เกี่ยวกับชุดหยุด (Stopping Sets) ในส่วนของเมทริกซ์พาริตีเช็ก โดยนำเสนออัลกอริทึมการค้นหาบนพื้นฐานของเทอริสโดยประมาณการตรวจจับชุดหยุดหลาย ๆ ชุด ด้วยการทำให้ H ใหญ่ขึ้นโดยการรวมวิธีการสร้างแบบสุ่มและวิธีการค้นหาบนพื้นฐานของเทอริสทำให้ได้รหัสที่มีระดับความผิดพลาดต่ำกว่ารหัสที่ถูกสร้างแบบสุ่มและดีกว่ารหัสที่สร้างด้วยวิธีการอื่นที่เปรียบเทียบอย่างเห็นได้ชัดขณะที่

asymptotic arguments ถูกทำให้เกิดขึ้นสำหรับการไม่มีชุดหยุดขนาดเล็ก ๆ ของกลุ่มรหัสแอลดีพีซีแบบไม่คงที่เกี่ยวกับข้อจำกัดของความยาวบล็อกใหญ่ เทคนิคการสร้างที่หลีกเลี่ยงชุดหยุดขนาดเล็ก ๆ ทางปฏิบัติจริงเป็นสิ่งที่จำเป็น

สรุปงานนี้นำเสนออัลกอริทึมการสร้างเมทริกซ์พาริตีเช็กสำหรับรหัสแอลดีพีซีแบบไม่คงที่ความยาวบล็อกสั้นที่หลีกเลี่ยงชุดหยุดชุดเล็ก ๆ รหัสเหล่านี้มีสมรรถนะที่ดีกว่ารหัสที่สร้างด้วยวิธีแบบสุ่ม โดยมีสมรรถนะใกล้เคียงกับรหัสของพวกเขารหัสที่ถูกสร้างโดยใช้การรวมกันระหว่างวิธีสองวิธีจะมีสมรรถนะดีกว่าอย่างมาก

3.7 งานวิจัยรหัสแอลดีพีซีกลุ่มเข้ารหัส

3.7.1 การเข้ารหัสเชิงเส้นของรหัสแอลดีพีซี [30] งานวิจัยนี้เสนอวิธีการเข้ารหัสที่มีความซับซ้อนเชิงเส้นสำหรับรหัสแอลดีพีซีที่สามารถกำหนดได้ตามต้องการ เริ่มจากวิธีการถอดรหัสบนพื้นฐานกราฟอย่างง่าย ที่เรียกว่า “label-and-decide”พิสูจน์ว่าวิธีการ “label-and-decide” สามารถประยุกต์ใช้กับกราฟแทนเนอร์ได้โดยโครงสร้างลำดับชั้น (Hierarchical หรือ Pseudo-Trees) และส่งผลต่อความซับซ้อนของการเข้ารหัสเป็นเชิงเส้นกับความยาวบล็อกของรหัส จากนั้นกำหนดแบบที่สองของกราฟแทนเนอร์ เรียกว่า ชุดหยุดของรหัส ชุดหยุดของรหัสถูกเข้ารหัสในความซับซ้อนเชิงเส้นโดยอัลกอริทึม label-and-decide ที่ถูกแก้ไขใหม่ เรียกว่า “label-decide-recomputed” สุดท้ายพิสูจน์ว่ากราฟแทนเนอร์ใด ๆ สามารถถูก partitioned ไปในชุดหยุดการเข้ารหัสได้ โดยการเข้ารหัสแต่ละชุดหยุดหรือโครงสร้างลำดับชั้นอย่างเป็นลำดับ งานนี้ได้พัฒนาวิธีการเข้ารหัสความซับซ้อนเชิงเส้นสำหรับรหัสแอลดีพีซีทั่วไปโดยที่ความซับซ้อนในการเข้ารหัสถูกพิสูจน์เพื่อให้น้อยกว่า$4 \cdot M \cdot (\bar{k} - 1)$ โดยที่ Mคือจำนวนของแถวอิสระในเมทริกซ์พาริตีเช็กและ $\bar{k}$ แทนน้ำหนักแถวเฉลี่ยของเมทริกซ์พาริตีเช็กงานวิจัยนี้เสนอวิธีการเข้ารหัสความซับซ้อนเชิงเส้นสำหรับรหัสแอลดีพีซีทั่วไป โดยการวิเคราะห์การเข้า-ถอดรหัสกราฟแทนเนอร์ของพวกมัน แสดงให้เห็นว่ารูปแบบเฉพาะของกราฟแทนเนอร์คือ โครงสร้างลำดับชั้นและชุดหยุดของการเข้ารหัส สามารถถูกเข้ารหัสในเวลาเชิงเส้นได้ จากนั้นได้พิสูจน์ว่ากราฟแทนเนอร์ใด ๆ สามารถถูกแยกไปเป็น โครงสร้างลำดับชั้นและชุดหยุดการเข้ารหัสได้ โดยการเข้ารหัสโครงสร้างลำดับชั้นและชุดหยุดการเข้ารหัสในลักษณะเป็นลำดับถัดกันไป ประสบความสำเร็จการถอดรหัสที่มีความซับซ้อนเชิงเส้นสำหรับรหัสแอลดีพีซีในวิธีการที่เสนอสามารถประยุกต์ใช้ได้ในช่วงกว้างของรหัส กล่าวคือมันจะไม่ถูกจำกัดสำหรับรหัสแอลดีพีซีสามารถประยุกต์ใช้ได้ทั้งรหัสแอลดีพีซีแบบคงที่และไม่คงที่ ความจริงแล้ววิธีการเข้ารหัสเวลาเชิงเส้นที่เสนอสามารถประยุกต์ใช้กับรหัสบล็อกรูปแบบใด ๆ ก็ได้ มันทำให้ปัญหาความซับซ้อนการเข้ารหัสที่สูงหมดไปได้สำหรับรหัสความยาวบล็อกยาว ซึ่งในอดีตถูกเข้ารหัสโดยการคูณเมทริกซ์

3.7.2 รหัสแอลดีพีซีแบบไม่คงที่ที่มีสมรรถนะสูงเข้ารหัสง่ายบนพื้นฐานเชิงระบบ [31] งานวิจัยนี้ เสนอการสร้างรหัสแอลดีพีซีแบบใหม่ในกลุ่มแบบไม่คงที่และเป็นแบบระบบ บนเมทริกซ์พาริตีเช็กที่มีความหนาแน่นน้อย ซึ่งจะช่วยลดความซับซ้อนในการเข้ารหัส ถูกเสนอบนหลักการของรหัสบล็อกในแถวและคอลัมน์ที่ออกแบบใหม่ที่ถูกกำหนดคุณลักษณะโดยกราฟและการถอดรหัสแบบทำซ้ำ บนช่องสัญญาณรบกวนแบบเกาส์สีขาว (AWGN) โดยมีเมทริกซ์พาริตีเช็กตามความสัมพันธ์ผ่านสมการที่ (3.35) ถึง (3.44) และดังรูปที่ 3.10

$$H_{(m\times n)}x^{T}_{(1\times n)} = 0^{T} \tag{3.35}$$

$$\tilde{H}_{(m\times n)} = \begin{bmatrix} \tilde{h}_{0,0} & \cdots & \tilde{h}_{0,n-m-1} & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \vdots & \vdots & \ddots & 1 & 0 \\ \tilde{h}_{m-1,0} & \cdots & \cdots & \cdots & \tilde{h}_{m-1,n-2} & 1 \end{bmatrix} \tag{3.36}$$

$$p_0 = \sum_{i=0}^{n-m-1} \tilde{h}_{0,i}s_i \tag{3.37}$$

$$p_l = \sum_{i=0}^{n-m-1} \tilde{h}_{l,i}s_i + \sum_{j=0}^{l-1} \tilde{h}_{l,j}p_j \tag{3.38}$$

$$G_{(n-m)\times n} = [I_{(n-m)\times(n-m)} \quad P_{(n-m)\times m}] \tag{3.39}$$

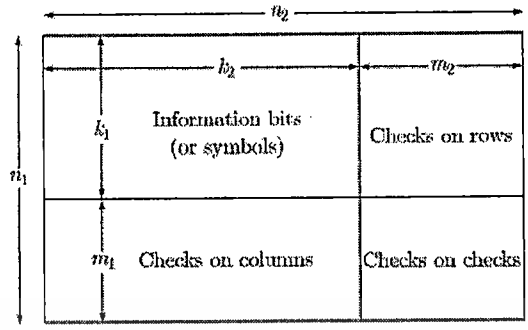
$$(H_0)_{m\times n} = [U_{m\times(n-m)} \quad I_{m\times m}] \tag{3.40}$$

$$d'_v = \frac{m(d_c-1)}{n-m} = \frac{d_v-m/n}{1-m/n} = d_v\frac{1-1/d_c}{1-d_v/d_c} \tag{3.41}$$

$$P_{60\times 60} = \begin{bmatrix} I_{20\times 20} & I_{20\times 20} & I_{20\times 20} \\ I_{20\times 20} & I_{20\times 20} & I_{20\times 20} \\ I_{20\times 20} & I_{20\times 20} & I_{20\times 20} \end{bmatrix} \tag{3.42}$$

$$(H_0)_{60\times 120} = P^{T}_{60\times 60}[I_{60\times 60}] = [P_{60\times 60} \quad I_{60\times 60}] \tag{3.43}$$

$$G_{60\times 120} = [I_{60\times 60} \quad P_{60\times 60}] \tag{3.44}$$



รูปที่ 3.10 แผนภาพของรหัส $P = C_1 \times C_2$

3.8 งานวิจัยรหัสแอลดีพีซีกลุ่มถอดรหัส

รหัสสำหรับการถอดรหัสแบบทำซ้ำจากเรขาคณิตบางส่วน [32] งานวิจัยนี้ พัฒนารหัสที่เหมาะสำหรับการถอดรหัสแบบทำซ้ำโดยใช้อัลกอริทึมผลรวม การพิจาณากลุ่มของโครงสร้างการรวมตัวกันด้วยเรขาคณิตบางส่วน (Partial Geometries) สามารถที่จะกำหนดกลุ่มต่าง ๆ ของรหัสแอลดีพีซีได้ซึ่งรวมเอากลุ่มที่รู้จักหลาย ๆ กลุ่มที่มีอยู่รวมกันเป็นกรณีพิเศษ เนื่องจากรหัสแอลดีพีซีแบบพีชคณิตถูกจำกัด ดังนั้น กลุ่มใหม่ของรหัสที่ได้โดยการทำให้อยู่ในรูปทั่วไปแบบเรขาคณิตบางส่วนจึงเพิ่มช่วงของตัวเลือกของความยาว และอัตรารหัสที่มีอยู่ได้ พิสูจน์หาขอบเขตบนระยะห่างต่ำสุด เฉลี่ยและเกริธสำหรับรหัสจากเรขาคณิตบางส่วน และเสนอผลการสร้างและสมรรถนะสำหรับกลุ่มของเรขาคณิตบางส่วนซึ่งยังไม่เคยถูกนำเสนอมาก่อนสำหรับใช้กับการถอดรหัสแบบทำซ้ำ แสดงให้เห็นว่ารหัสใหม่นี้สามารถประสบความสำเร็จในการพัฒนาสมรรถนะการแก้ไขข้อผิดพลาดได้ดีกว่ารหัสแอลดีพีซีแบบสุ่มในบางกรณี ประสบความสำเร็จพร้อมกับลดความซับซ้อนการถอดรหัสอย่างเห็นได้ชัด งานวิจัยนี้เป็นกลุ่มของรหัสแอลดีพีซีที่พิสูจน์ได้มาจากเรขาคณิตบางส่วนถูกเสนอ ได้คำนวณหาสมการ หรือขอบเขตสำหรับเป็นคุณสมบัติหลักของรหัสที่ถูกกำหนดจากเรขาคณิตบางส่วน คือ ระยะห่างต่ำสุดเกริธและมิติรหัสจากเรขาคณิตบางส่วนให้สมรรถนะการแก้ไขข้อผิดพลาดที่ดีขึ้นกว่ารหัสแอลดีพีซีแบบสุ่ม

บทที่ 4
การออกแบบเมทริกซ์พาริตีเช็กของการวิจัย

บทนี้แสดงวิธีการออกแบบเมทริกซ์พาริตีเช็กของงานวิจัยนี้ โดยเริ่มจากการศึกษารูปแบบการออกแบบเมทริกซ์พาริตีเช็กสำหรับรหัสแอลดีพีซีในแบบต่าง ๆ ที่ได้กล่าวแล้วในบทที่ 3 จากนั้นทำการออกแบบโดยเริ่มพิจารณาที่รูปแบบของเมทริกซ์พาริตีเช็กด้วยการลดเลขหนึ่งลงจากการตัดจำนวนเลขหนึ่งในแนวเส้นทแยงมุมทั้งด้านซ้ายและขวาของเมทริกซ์โดยใช้หลักการทรานสโพรสเมทริกซ์ต่อมาได้พิจารณาวิธีการคำนวณและกำหนดค่าตัวแปรเพื่อใช้สร้างเมทริกซ์พาริตีเช็กให้มีสมรรถนะดีขึ้น โดยพัฒนาจากรหัสแอลดีพีซีแบบอาร์เรย์เป็นพื้นฐานที่มีตัวแปรเริ่มต้นในการเข้า-ถอดรหัส ด้วยค่า j, k และp เป็นจำนวนเต็มบวก เมื่อ j<k<p และค่า p ต้องเป็นจำนวนเฉพาะ การกำหนดรูปแบบการเลื่อนหมุนวนเป็นแบบสมมาตรที่ส่งผลให้เมทริกซ์ปราศจากลูป 4 พร้อมกับปรับค่า p เป็นแบบจำนวนเต็มบวกที่ไม่ใช่จำนวนเฉพาะเพียงอย่างเดียว จากนั้นได้มีการนำทฤษฎีทางคณิตศาสตร์พื้นฐานที่เรียกว่า เมจิกสแควร์หรือจัตุรัสมหัศจรรย์ มาประยุกต์ใช้เพื่อออกแบบเมทริกซ์พาริตีเช็ก ให้ได้ผลการทดสอบสมรรถนะที่ดีเหมือนการออกแบบด้วยวิธีการสุ่ม หรือเข้าใกล้ขีดจำกัดของแชนนอล ภายใต้ระบบที่มีสัญญาณรบกวนแบบเกาส์สีขาวออกแบบได้ 3 วิธี จากนั้นทำการทดสอบทุกวิธีที่ออกแบบ เพื่อเปรียบเทียบผลหาวิธีที่ดีที่สุด นำไปเปรียบเทียบกับวิธีการออกแบบเมทริกซ์พาริตีเช็กสำหรับรหัสแอลดีพีซีแบบอื่นในกลุ่มคล้ายกัน ได้แก่ กลุ่มรหัสแอลดีพีซีแบบอาร์เรย์ กลุ่มทฤษฎีเศษเหลือของจีน และกลุ่มออกแบบวิธีการสุ่มเพื่อวิเคราะห์ผลและใช้สถิติประยุกต์เป็นข้อมูลสนับสนุนผลต่อไป

4.1 ออกแบบโดยพิจารณารูปแบบเมทริกซ์พาริตีเช็กเพื่อลดจำนวนเลขหนึ่ง

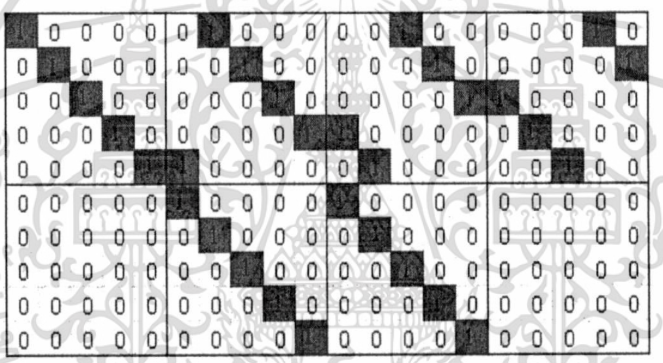
การออกแบบเมทริกซ์พาริตีเช็กสำหรับรหัสแอลดีพีซีนี้ อยู่บนพื้นฐานของรหัสแอลดีพีซีแบบอาร์เรย์ [6] และแบบมอดิฟายย์อาร์เรย์ [10] ดังข้อเสนอของ Othman และคณะ [12] กล่าวว่า เมทริกซ์พาริตีเช็กที่มีการกำหนดโครงสร้างสามารถนำไปใช้เพื่อลดความซับซ้อนของการใช้งานจริงได้อย่างเห็นได้ชัด มีสมรรถนะที่ดีสำหรับเมทริกซ์พาริตีเช็กที่สามารถกำหนดขนาดได้ตามความต้องการ แต่การสร้างเมทริกซ์พาริตีเช็กนั้นไม่สามารถสร้างได้ทั้งหมดเป็นเพียงการออกแบบบางส่วน จึงนำแนวคิดและผลดังกล่าวใช้ออกแบบเมทริกซ์พาริตีเช็กของงานวิจัยครั้งนี้ ที่อยู่บนพื้นฐานงานวิจัยของ Eleftheriou [10] และ Singhaudom [11] โดยทำการลดจำนวนเลข 1 ในสามเหลี่ยมบน (Upper Triangle) โดยประยุกต์ใช้การทรานสโพรสเมทริกซ์ และการสลับแถวและหลักเพื่อหลีกเลี่ยงลูป 4 จากผลการศึกษาจะได้เมทริกซ์พาริตีเช็กใหม่ จำนวนทั้งหมด 4 แบบ แต่เนื่องจากสองแบบแรกให้สมรรถนะค่อนข้างต่ำ แบบที่สามมีรูปแบบที่ดีสามารถลดความซับซ้อนของการถอดรหัสได้ และให้สมรรถนะในการแก้ไขความผิดพลาดได้ดี

ขณะที่แบบที่สี่ไม่สามารถใช้ในการถอดรหัสได้ ดังนั้น ในวิทยานิพนธ์นี้จึงนำเสนอรูปแบบที่สามเท่านั้น โดยกระบวนการสร้างเมทริกซ์พาริตีเช็ก ดังสมการที่ (4.1)

$$\left.\begin{array}{l} \begin{bmatrix} I & I \\ 0 & I \end{bmatrix}\begin{bmatrix} I & 0 \\ I & I \end{bmatrix}\begin{bmatrix} I & II & 0 \\ 0 & II & I \end{bmatrix} \\ \begin{bmatrix} I & II & I \\ 0 & II & 0 \end{bmatrix}\begin{bmatrix} I & \alpha\alpha^2 & \alpha^3 \\ 0 & I & 0 \end{bmatrix} \end{array}\right\} \Rightarrow \quad (4.1)$$

ดังนั้นจะได้ดังสมการที่ (4.2) และรูปที่ 4.1

$$H = \begin{bmatrix} I & \alpha\alpha^2 & \alpha^3 \\ 0 & I\ I & 0 \end{bmatrix} \quad (4.2)$$



รูปที่ 4.1 เมทริกซ์พาริตีเช็กสำหรับรหัสแอลดีพีซีที่ออกแบบ

ผลของงานวิจัยแสดงดัง [33] และเสนอผลการทดสอบในบทถัดไป

4.2 ออกแบบโดยพิจารณากำหนดค่าตัวแปรกับรูปแบบสมมาตร

การออกแบบเมทริกซ์พาริตีเช็กที่อยู่บนหลักการของการทรานสโพรสเมทริกซ์พาริตีเช็กที่สมมาตร ดังสมการที่ (4.3)

$$S = S^T \tag{4.3}$$

สมาชิกของเมทริกซ์สมมาตรจะสมมาตรได้ด้วยความสัมพันธ์ของสมาชิกที่แบ่งด้วยแนวเส้นทแยงมุม (ด้านบนซ้ายไปยังด้านล่างขวา) สามารถเขียนได้ว่า ถ้าสมาชิกคือ $s \equiv s_{ij}$ แล้ว $s_{ij} \equiv s_{ji}$จะได้ดังสมการที่ (4.4)

$$S^{-1}S^T=I \tag{4.4}$$

โดยที่ I = เมทริกซ์เอกลักษณ์

ดังนั้น เมทริกซ์สมมาตรS จึงมีรูปแบบดังสมการที่ (4.5)

$$S = \begin{bmatrix} s_{11} & s_{12} & \cdots & s_{1n} \\ s_{12} & s_{22} & \cdots & s_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ s_{1n} & s_{2n} & \cdots & s_{nn} \end{bmatrix} \tag{4.5}$$

ในหัวข้อนี้จะอธิบายเกี่ยวกับรายละเอียดการกำหนดสมการของเมทริกซ์พาริตีเช็กโดยใช้แนวความคิดของเมทริกซ์สมมาตรแทนที่จะเป็นการใช้เมทริกซ์สับเปลี่ยน (Permutation) วิธีการออกแบบรูปแบบใหม่นี้เป็นการกำหนดให้เมทริกซ์ย่อยมีความสมมาตรกันเพื่อกำหนดเป็นโครงสร้างของเมทริกซ์พาริตีเช็กซึ่งมีวิธีการคล้ายกับการสร้างรหัสแอลดีพีซีแบบอาร์เรย์ ที่เสนอโดย [6]

การออกแบบเมทริกซ์พาริตีเช็กใช้รูปแบบของเมทริกซ์สมมาตรจะได้รูปแบบเมทริกซ์พาริตีเช็กแสดงดังสมการที่ (4.5) ซึ่งเป็นเมทริกซ์สมมาตร S ขนาด qxq โดยที่q สามารถเป็นได้ทั้งจำนวนเต็มบวกทั่วไปหรือจำนวนเฉพาะก็ได้ สำหรับการออกแบบเมทริกซ์ขนาด qxq ให้กำหนดสมาชิกฐานสองของเมทริกซ์เป็น s_{xy}= {0,1} และดรรชนีrเชื่อมต่อไปยังr=(q/3)*2 ค่า x และ y คือ ค่าดรรชนีแถวและหลักตามลำดับ ในการออกแบบเมทริกซ์ S สมาชิกแถวสามารถคำนวณได้โดยรหัสเทียม (Pseudo Code) ดังรูปที่ 4.2

```
Odd number row:
    bb=0;  aa=0;  x=1;
For cc=1 to q              (row)
for dd=1 to q            (column)
    if  ((2*q)-dd)-(2*bb)=dd   then
s=1;  goto 10;
else
s=0
end;
        end;
10:     cc=cc+2;  bb=bb+1;
if   cc = ((2*q)-cc)-(2*bb)  then
    goto 20;
      end;
20: end
```

รูปที่ 4.2 รหัสเทียมสร้างเมทริกซ์พาริตีเช็กแบบสมมาตร [34]

หลักที่เป็นเลขคี่สามารถสร้างได้โดยการทรานสโพรสแถวที่เป็นเลขคี่ที่คำนวณได้ดังนี้ $S_{y,x}=(S_{x,y})^T$ การสร้างแถวและหลักที่เป็นเลขคู่ (2, 4, 6, ...) ของเมทริกซ์ย่อย (S) เป็นการกำหนดเลข “1” ที่ตำแหน่ง x=y ของแต่ละหลัก ถ้ายังไม่มีเลข “1” ให้เติมตำแหน่งนั้นด้วยเลข “1” กรณีอื่นให้เติมเลข “0” เมทริกซ์ผลลัพธ์แสดงดังสมการที่ (4.6)

$$S=\begin{bmatrix}
0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & s_{i,j} \\
0 & s_{2,2} & 0 & 0 & & 0 & 0 & 0 & & 0 & 0 \\
0 & 0 & 0 & 0 & & 0 & 0 & 0 & \cdots & s_{(x+2),(y-1)} & 0 \\
0 & 0 & 0 & s_{4,4} & & 0 & 0 & 0 & & 0 & 0 \\
 & & & \ddots & \ddots & \vdots & \vdots & \vdots & & & \\
\vdots & \vdots & \vdots & & & s_{r,r} & 0 & 0 & & & \\
 & & & & & 0 & 0 & s_{(r+2),(r-1)} & & & \\
\vdots & \vdots & \vdots & & & 0 & s_{(r+1),(r-2)} & 0 & & & \\
 & & & & & & & & & & \\
0 & 0 & s_{(y-1),(x+2)} & & & & & & & & \\
s_{y,x} & 0 & 0 & 0 & \cdots & \cdots & \cdots & \cdots & \cdots & 0 & 0
\end{bmatrix} \quad (4.6)$$

ตัวอย่าง การออกแบบเมทริกซ์ที่ค่า q = 10ดังสมการที่ (4.7)

$$S = \begin{bmatrix} 0 & 0 & 00 & 0 & 00 & 0 & 0 & 1 \\ 0 & 0 & 00 & 0 & 00 & 0 & 0 & 0 \\ 0 & 0 & 00 & 0 & 00 & 0 & 1 & 0 \\ 0 & 0 & 01 & 0 & 00 & 0 & 0 & 0 \\ 0 & 0 & 00 & 0 & 00 & 1 & 0 & 0 \\ 0 & 0 & 00 & 0 & 10 & 0 & 0 & 0 \\ 0 & 0 & 00 & 0 & 01 & 0 & 0 & 0 \\ 0 & 0 & 00 & 1 & 00 & 0 & 0 & 0 \\ 0 & 0 & 10 & 0 & 00 & 0 & 0 & 0 \\ 1 & 0 & 00 & 0 & 00 & 0 & 0 & 0 \end{bmatrix} \quad (4.7)$$

จากสมการที่ (4.7) จะเห็นได้ว่าเมทริกซ์นี้มีคุณสมบัติการทรานสโพรส $S^T = S$ ซึ่งไม่มีลูป 4 ในเมทริกซ์ย่อย ดังนั้น เราสามารถสร้างด้วยการเลื่อนหรือS^1เป็นการเลื่อนวนทางขวามือจากรูปแบบดั้งเดิมของเมทริกซ์ S ที่ได้แสดงดังสมการที่ (4.8)

$$S^1 = \begin{bmatrix} 1 & 0 & 00 & 0 & 00 & 0 & 0 & 0 \\ 0 & 0 & 10 & 0 & 00 & 0 & 0 & 0 \\ 0 & 0 & 00 & 0 & 00 & 0 & 0 & 1 \\ 0 & 0 & 00 & 1 & 00 & 0 & 0 & 0 \\ 0 & 0 & 00 & 0 & 00 & 0 & 1 & 0 \\ 0 & 0 & 00 & 0 & 01 & 0 & 0 & 0 \\ 0 & 0 & 00 & 0 & 00 & 1 & 0 & 0 \\ 0 & 0 & 00 & 0 & 10 & 0 & 0 & 0 \\ 0 & 0 & 01 & 0 & 00 & 0 & 0 & 0 \\ 0 & 1 & 00 & 0 & 00 & 0 & 0 & 0 \end{bmatrix} \quad (4.8)$$

การเลื่อนด้วยจำนวนครั้งในการเลื่อนอื่น ๆ สามารถทำได้คล้าย ๆ กัน รูปแบบการเลื่อนบล็อกทำลักษณะคล้าย ๆ กับที่เสนอใน [24] สุดท้ายเราสามารถสร้างเมทริกซ์พาริตีเช็กได้จากสมการ (4.9) ถึง (4.11) ซึ่งสูตรในสมาการนี้ ถูกนำไปใช้ในการออกแบบรหัสแล้วนำไปทดสอบสมรรถนะซึ่งจะได้กล่าวถึงผลการทดสอบในบทถัดไป ดังเสนอผลงานใน [34] ในการออกแบบรหัสนี้มีตัวแปรในการออกแบบ 3 ตัว ประกอบด้วย j,k และ q (j, k ≤ q) โดยที่ j และ k เป็นจำนวนเต็ม ส่วน q คือขนาดของเมทริกซ์ย่อยที่เป็นเลขทั่วไปที่ไม่จำเป็นต้องเป็นจำนวนเฉพาะ

$$H = \begin{bmatrix} I & I & I & I & \cdots & \cdots & I \\ 0 & I & \lambda^{(2,3)} & \cdots & \lambda^{(2,j)} & \cdots & \lambda^{(2,k-1)} \\ 0 & 0 & I & \lambda^{(3,4)} & \lambda^{(3,j)} & \cdots & \lambda^{(3,k-2)} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \cdots & I & \cdots & \lambda^{(j,k-j+1)} \end{bmatrix} \quad (4.9)$$

เมื่อ $\lambda^{(j,k)} = S^{P_T(j,k)}$ (4.10)

และ

$$P_T(j,k) = (j-1)(k-1) + \left|\frac{(j-1)(k-j)}{q}\right| \quad (4.11)$$

ผลของงานวิจัยแสดงดัง [34] และเสนอผลการทดสอบในบทถัดไป

4.3 ออกแบบโดยประยุกต์ทฤษฎีคณิตศาสตร์พื้นฐาน

เป็นที่ทราบกันว่าสมรรถนะของรหัสแอลดีพีซีนั้น จะมีผลเป็นอย่างไรขึ้นอยู่กับปัจจัยหนึ่ง คือ เมทริกซ์พาริตีเช็กที่ออกแบบ ดังนั้นจึงต้องออกแบบให้เหมาะสมที่สุด ซึ่งในหัวข้อนี้ได้ออกแบบด้วยการนำคณิตศาสตร์พื้นฐานเรื่องเมจิกสแควร์มาประยุกต์ใช้ในการสร้างเมทริกซ์พาริตีเช็กสำหรับรหัสแอลดีพีซี มีวิธีการสร้าง 3 แบบ คือ 1) สร้างโดยใช้ตัวเลขเมจิกสแควร์มาเป็นค่าการเลื่อนวนของเมทริกซ์ย่อยทุกเมทริกซ์ 2) สร้างโดยใช้ตัวเลขเมจิกสแควร์มาเป็นค่าการเลื่อนวนของเมทริกซ์ย่อยทุกเมทริกซ์สลับกับตัวเลขที่ได้จากการคำนวณค่าที่สัมพันธ์กับความยาวบล็อกของคำรหัส และ 3) สร้างโดยใช้ตัวเลขเมจิกสแควร์มาเป็นค่าการเลื่อนวนของเมทริกซ์ย่อยทุกเมทริกซ์สลับกับตัวเลขที่ได้จากการคำนวณค่าที่สัมพันธ์กับความยาวบล็อกของคำรหัส บวกด้วยการจัดวางที่ปราศจากลูป 4

ดังที่ได้กล่าวไปแล้วว่า โครงสร้างของเมทริกซ์พาริตีเช็กมีผลต่อสมรรถนะของรหัสเป็นอย่างมาก ในอัลกอริทึมที่นำเสนอในงานวิจัยนี้จะเป็นการสร้างเมทริกซ์พาริตีเช็กโดยการนำตัวเลขที่แตกต่างกันทั้งหมดจากเมจิกสแควร์มาใช้เป็นค่าการเลื่อนวนของเมทริกซ์ย่อยแต่ละตัว ซึ่งมีขั้นตอนดังนี้

1) กำหนดตัวแปรการสร้างเมทริกซ์พาริตีเช็ก คือ j, k และ p ซึ่งทั้งหมดต้องเป็นจำนวนเต็มและต้องมีค่ามากกว่าหรือเท่ากับ 3 โดยที่ $j<k\leq p$ ให้ λ เป็นจำนวนตัวเลขที่จะนำมาใช้เป็นค่าจำนวนครั้งในการเลื่อนวนของเมทริกซ์ย่อยทั้งหมด (ที่จะนำมาจากเมจิกสแควร์) ดังนั้นเพื่อให้แน่ใจว่าจำนวนตัวเลขจากเมจิกสแควร์จะเพียงพอหรือไม่ (ซึ่งตัวเลขทั้งหมดต้องไม่ซ้ำกัน) ต้องมีการตรวจสอบด้วยสมการที่ (4.12)

$$\lambda = (j \times k) - \left(\frac{j(j+1)}{2}\right) - (k-1) \leq z^2 \quad (4.12)$$

ตัวอย่างการสร้างเมทริกซ์พาริตีเช็กสำหรับรหัสแอลดีพีซีแบบไม่คงที่ที่มีความยาวบล็อกเท่ากับ 513 และมีอัตรารหัสเท่ากับ 0.7 (j=3, k=9 และ p=57) ดังนั้น เมื่อตรวจสอบโดยสมการ (4.12) ค่า z ต้องมีค่ามากกว่า4

2) สร้างเมจิกสแควร์ขนาด zxz ที่สัมพันธ์กับความยาวบล็อกและอัตรารหัสที่กำหนดเพื่อให้ครอบคลุมเมจิกสแควร์ที่เป็นไปได้ทั้งหมดและเนื่องจากมีเมจิกสแควร์จำนวน 7 เมจิกสแควร์ในกลุ่มที่ 1 (z=3 ถึง 9) เมจิกสแควร์ขนาด 4 ถึง 9 จึงถูกนำมาพิจารณาใช้ในการสร้าง เมทริกซ์พาริตีเช็ก

3) นำตัวเลขในเมจิกสแควร์มาใช้เป็นจำนวนครั้งการเลื่อนวนของเมทริกซ์ย่อยแต่ละเมทริกซ์ (เมทริกซ์ย่อยมีขนาด pxp) เพื่อศึกษาผลที่มีต่อสมรรถนะของรหัสและเพื่อหาโครงสร้างหรือรูปแบบที่ดีที่สุดของเมทริกซ์พาริตีเช็กจึงได้เสนอแนวทางออกแบบซึ่งเกี่ยวข้องกับวิธีการเลื่อนวนเมทริกซ์ย่อยไว้ 3 วิธี สรุปเบื้องต้นได้ดังนี้

วิธีที่หนึ่ง (I) จำนวนหรือตัวเลขที่นำมาใช้เป็นจำนวนครั้งในการเลื่อนวนของเมทริกซ์ย่อยได้มาจากตัวเลขในเมจิกสแควร์ทั้งหมดโดยไม่มีการจัดเรียงใหม่

วิธีที่สอง (II) จำนวนหรือตัวเลขที่นำมาใช้เป็นจำนวนครั้งในการเลื่อนวนของเมทริกซ์ย่อย ได้มาจากตัวเลขในเมจิกสแควร์ทั้งหมดแต่มีการจัดเรียงบางอย่าง กล่าวคือ ตัวเลขเหล่านี้เป็นตัวเลขดั้งเดิมที่มีในเมจิกสแควร์แต่มีความสัมพันธ์กันระหว่างแต่ละแถว (เมจิกสแควร์มีผลบวกแต่ละแถว คอลัมน์ และแนวเส้นทแยงจะเท่ากัน) เพื่อลบความสัมพันธ์จึงเลือกตัวเลขที่เริ่มต้นจากการหาเลข 1 ในตารางเมจิกสแควร์ก่อน จากนั้นเลือกตัวเลขที่อยู่ถัดไปจัดวางจนเต็มเมทริกซ์พาริตีเช็กที่ต้องการ จึงกล่าวได้ว่าตัวเลขที่นำมาใช้มีความเป็นอิสระต่อกัน

วิธีที่สาม (III) ตัวเลขเพียงบางตัวเท่านั้นจะถูกเลือกมาจากเมจิกสแควร์และมีการจัดวางบางอย่างเพื่อให้มีการกระจายของตัวเลขเป็นแบบปกติ (Normal Curve Distribution) โดยจะนำตัวเลขจากเมจิกสแควร์มาจำนวนครึ่งหนึ่งของตัวเลขทั้งหมด ส่วนตัวเลขที่เหลือจะได้มาจากการคำนวณ
วิธีการสร้างแต่ละวิธีมีรายละเอียดดังนี้

1) วิธีที่หนึ่งเป็นการสร้างเมทริกซ์พาริตีเช็ก โดยใช้ตัวเลขและตำแหน่งดั้งเดิมของตารางเมจิกสแควร์ซึ่งจะมีความสัมพันธ์ระหว่างแถว คอลัมน์ และแนวเส้นทแยงมุมทั้งหมดที่มีผลบวกของตัวเลขแต่ละแนวดังกล่าวเท่ากัน ทุกขนาดของเมจิกสแควร์ที่สัมพันธ์กับขนาดความยาวบล็อกและอัตรารหัส ซึ่งจะทำให้แน่ใจได้ว่าจำนวนตัวเลขเพียงพอต่อการสร้างเมทริกซ์พาริตีเช็กแต่ละครั้งทุกขนาดของเมจิกสแควร์จะถูกนำมาใช้ในการศึกษาทั้งหมด

ก) เตรียมเมทริกซ์พาริตีเช็กดังแสดงในตารางที่ 4.1 โดยที่ X หมายถึงจำนวนครั้งการเลื่อนวนของเมทริกซ์ย่อยแต่ละเมทริกซ์ และ I หมายถึง เมทริกซ์เอกลักษณ์

ตารางที่ 4.1 แสดงเมทริกซ์พาริตีเช็กแบบไม่คงที่

H	1	2	3	4	5	6	7	8	9
1	I	I	I	I	I	I	I	I	I
2	0	I	X	X	X	X	X	X	X
3	0	0	I	X	X	X	X	X	X

ข) ตัวเลขในแถวแรก และแถวที่สองของเมจิกสแควร์ถูกนำมาแทนในแถวที่สอง และสามของเมทริกซ์พาริตีเช็กในตำแหน่งของ X ในแต่ละแถวแบบตัวต่อตัว ซึ่งมีวิธีการแทน 3 ลักษณะ (ขึ้นอยู่กับขนาดของเมจิกสแควร์) ดังนี้

ข.1) ถ้าจำนวนตัวเลขในแถวแรกของเมจิกสแควร์น้อยกว่า X ในแถวแรกของเมทริกซ์พาริตีเช็ก (ดังแสดงในตารางที่ 4.1 โดยใช้ตัวเลขจากตารางเมจิกสแควร์ขนาด 5×5 ในรูปที่ 4.3) ให้แทน X ที่เหลือ (ดังตารางที่ 4.2 จะเหลือ X อยู่ 3 ตัว) ด้วยเลขประจำหลัก แต่ถ้าเลขประจำหลักถูกใช้ไปแล้ว ให้พิจารณาตัวเลขถัดไปจนกว่าจะเป็นเลขที่ยังไม่ได้ใช้ที่ใกล้เคียงที่สุด ผลการแทนที่ Xแสดงดังตารางที่ 4.3

11	24	7	20	3
4	12	25	8	16
17	5	13	21	9
10	18	1	14	22
23	6	19	2	15

รูปที่ 4.3 เมจิกสแควร์ขนาด 5×5 (Mars)

ตารางที่ 4.2 การแทนที่ด้วยเมจิกสแควร์

H	1	2	3	4	5	6	7	8	9
1	I	I	I	I	I	I	I	I	I
2	0	I	11	24	7	20	3	X	X
3	0	0	I	4	12	25	8	16	X

ตารางที่ 4.3 การแทนที่ X ด้วยค่าประจำตำแหน่งของคอลัมน์

H	1	2	3	4	5	6	7	8	9
1	I	I	I	I	I	I	I	I	I
2	0	I	11	24	7	20	3	*9*	*10*
3	0	0	I	4	12	25	8	16	*6*

ข.2) ถ้าจำนวนตัวเลขในแถวแรกของเมจิกสแควร์เพียงพอที่จะใช้เติม (แทนในตำแหน่ง X) ในเมทริกซ์พาริตีเช็กแบบแถวต่อแถว แต่มีตัวเลขบางตัวมีค่ามากกว่าหรือเท่ากับ p ให้เลือกใช้แถวถัดไปจนกว่าจะไม่มีกรณีดังกล่าว เนื่องจากจำนวนที่มากกว่าหรือเท่ากับ p สามารถทำให้เกิดผลที่เรียกว่า “modulo operation” เช่น 58 mod 57 = 1

ตัวอย่างเช่น ตัวเลขในแถวแรกของเมจิกสแควร์ขนาด 8×8 (ดังรูปที่ 4.4) ประกอบด้วยเลข 8, 58, 59, 5, 4, 62, 63 และ 1 ซึ่งจะเห็นว่าแถวนี้ มีตัวเลขที่มีค่ามากกว่า p ดังนั้น จึงต้องพิจารณาแถวที่สองและสามเพื่อนำไปใช้เติมในเมทริกซ์พาริตีเช็กดังแสดงในตารางที่ 4.4

8	58	59	5	4	62	63	1
49	15	14	52	53	11	10	56
41	23	22	44	45	19	18	48
32	34	35	29	28	38	39	25
40	26	27	37	36	30	31	33
17	47	46	20	21	43	42	24
9	55	54	12	13	51	50	16
64	2	3	61	60	6	7	57

รูปที่ 4.4 เมจิกสแควร์ขนาด 8×8 (Mercury)

ตารางที่ 4.4 การใช้แถว 2 และ 3 แทนค่าด้วยเมจิกสแควร์

H	1	2	3	4	5	6	7	8	9
1	I	I	I	I	I	I	I	I	I
2	0	I	49	15	14	52	53	11	10
3	0	0	I	41	23	22	44	45	19

ข.3) ถ้าจำนวนตัวเลขในแถวแรกของเมจิกสแควร์เพียงพอสำหรับการแทนที่ X แบบแถวต่อแถว แต่มีตัวเลขบางตัวในแถวของเมจิกสแควร์มีค่ามากกว่าหรือเท่ากับ p ให้ใช้ค่ามอดุโล p (modulop) ของค่านั้นในการแทนที่ X

ตัวอย่างเช่น ตัวเลขในแถวแรกของเมจิกสแควร์ขนาด 9×9 (ดังแสดงในรูปที่ 4.5) คือ 37, 78, 29, 70, 21, 62, 13, 54 และ 5 ตัวเลขในแถวที่สองคือ 6, 38, 79, 30,71, 22, 63, 14 และ 46 ซึ่งจะเห็นว่ามีตัวเลขบางตัวมีค่ามากกว่า p คือ 78, 70, 62, 79และ 71ซึ่งค่ามอดุโล p ของเลขเหล่านี้ คือ 21, 13, 5, 22 และ 14 ถ้าค่ามอดุโล p ของเลขเหล่านี้ยังได้ถูกใช้ ให้นำค่ามอดุโล p ของเลขเหล่านี้ไปใช้แทนค่า X แบบตัวต่อตัว แต่ถ้าค่ามอดุโล p ของเลขตัวใดถูกใช้แล้วให้พิจารณาตัวเลขที่ใกล้ที่สุดในลำดับถัดไป ผลของการจัดวางตัวเลขในกรณีตัวอย่างนี้ แสดงดังตารางที่ 4.5

37	78	29	70	21	62	13	54	5
6	38	79	30	71	22	63	14	46
47	7	39	80	31	72	23	55	15
16	48	8	40	81	32	64	24	56
57	17	49	9	41	73	33	65	25
26	58	18	50	1	42	74	34	66
67	27	59	10	51	2	43	75	35
36	68	19	60	11	52	3	44	76
77	28	69	20	61	12	53	4	45

รูปที่ 4.5 เมจิกสแควร์ขนาด 9×9 (Luna)

ตารางที่ 4.5 การจัดวางแบบมอดุโล p

H	1	2	3	4	5	6	7	8	9
1	I	I	I	I	I	I	I	I	I
2	0	I	37	20	29	12	21	5	13
3	0	0	I	6	38	23	30	14	22

จากขั้นต้อนข้างต้นของวิธีการสร้างเมทริกซ์พาริตีเช็กวิธีที่หนึ่งจะเห็นว่าเมทริกซ์พาริตีเช็กสามารถสร้างได้จากการใช้เมจิกสแควร์อื่น ๆ ได้อีก เช่น เมจิกสแควร์กลุ่มที่ 4 (เมจิกสแควร์ที่เรียกว่า "Strachey" และ "LUX" ดังแสดงในรูปที่ 3.6 ซึ่งผลลัพธ์ของกรณีตัวอย่างแสดงดังตารางที่ 4.6 และ 4.7

ตารางที่ 4.6 การจัดวางโดยใช้เมจิกสแควร์ขนาด 6×6 (Strachey)

H	1	2	3	4	5	6	7	8	9
1	I	I	I	I	I	I	I	I	I
2	0	I	8	1	6	26	19	24	9
3	0	0	I	3	5	7	21	23	25

ตารางที่ 4.7 การจัดวางโดยใช้เมจิกสแควร์ขนาด 6×6 (LUX)

H	1	2	3	4	5	6	7	8	9
1	I	I	I	I	I	I	I	I	I
2	0	I	32	29	4	1	24	21	9
3	0	0	I	30	31	2	3	22	23

2) วิธีที่สองนี้ มีจุดมุ่งหมายที่จะสร้างเมทริกซ์พาริตีเช็กโดยการใช้ตัวเลขและตำแหน่งดั้งเดิมที่มาจากเมจิกสแควร์โดยไม่นำความสัมพันธ์เกี่ยวกับผลรวมที่เท่ากันของแต่ละแถวมาด้วย ซึ่งความสัมพันธ์นี้จะหายไปเมื่อมีการเลือกตัวเลขโดยเริ่มจากหาตำแหน่งเลข 1 ที่อยู่ในเมจิกสแควร์แล้วเริ่มเลือกตัวเลขจากตำแหน่งนี้ (รวมเลข 1 ด้วย) ตัวเลขที่นำมาใช้ทั้งหมดจะเป็นอิสระต่อกันเพื่อให้การวิจัยครอบคลุมทุกขนาดของเมจิกสแควร์ที่เป็นไปได้ที่สัมพันธ์กับความยาวบล็อกและอัตรารหัส อีกทั้งเพื่อให้การสร้างเมทริกซ์พาริตีเช็กมีระเบียบวิธีที่แน่นอน จึงมีขั้นตอนการสร้างดังนี้

a) เตรียมรูปแบบของเมทริกซ์พาริตีเช็กดังแสดงในตารางที่ 4.1

b) แทนที่ X ในแถวที่สองด้วยตัวเลขจากเมจิกสแควร์แบบตัวต่อตัว โดยหาตำแหน่งของเลข 1 ในเมจิกสแควร์แล้วเลือกตัวเลขทั้งหมดจากตำแหน่งของเลข 1 นี้ (ใช้เลข 1 ด้วย) แทนที่ Xทั้งหมดจากซ้ายไปขวาทีละแถวไปเรื่อย ๆ ระหว่างการแทนที่นี้ถ้ามีตัวเลขที่มากกว่าหรือเท่ากับ pให้ข้ามตัวเลขตัวนั้นแล้วใช้ตัวเลขตัวถัดไป

ตัวอย่างผลการสร้างโดยใช้ตารางเมจิกสแควร์ขนาด 5×5 และ 9×9 แสดงดังตารางที่ 4.8 และ 4.9 ตามลำดับ

ตารางที่ 4.8 สร้างเมทริกซ์พาริตีเช็กด้วยวิธีที่สองเมจิกสแควร์ 5×5

H	1	2	3	4	5	6	7	8	9
1	I	I	I	I	I	I	I	I	I
2	0	I	1	14	22	23	6	19	2
3	0	0	I	15	11	24	7	20	3

ตารางที่ 4.9 สร้างเมทริกซ์พาริตีเช็กด้วยวิธีที่สองเมจิกสแควร์ 9×9

H	1	2	3	4	5	6	7	8	9
1	I	I	I	I	I	I	I	I	I
2	0	I	1	42	34	27	10	51	2
3	0	0	I	43	35	36	19	11	52

3) วิธีที่สามนี้มีจุดมุ่งหมายที่จะสร้างเมทริกซ์พาริตีเช็กโดยใช้ตัวเลขบางตัวจากตารางเมจิกสแควร์ ซึ่งตัวเลขเหล่านั้นทั้งหมดต้องเป็นอิสระต่อกัน และตัวเลขบางส่วนที่เหลือได้มาจากการจัดวางตัวเลขเพื่อให้เกิดการกระจายรูปแบบปกติของตัวเลขทั้งหมดที่ใช้เพื่อให้การวิจัยครอบคลุมทุกขนาดของเมจิกสแควร์ที่เป็นไปได้ที่สัมพันธ์กับความยาวบล็อก และอัตรารหัส อีกทั้งเพื่อให้การสร้างเมทริกซ์พาริตีเช็ก มีระเบียบวิธีที่แน่นอน จึงมีขั้นตอนการสร้างดังนี้

ก) เตรียมเมทริกซ์พาริตีเช็กดังแสดงในตารางที่ 4.10 โดยที่ X, I, β, Y และ Z หมายถึง จำนวนครั้งการเลื่อนวนของเมทริกซ์ย่อยแต่ละเมทริกซ์

ตารางที่ 4.10 เมทริกซ์พาริตีเช็กแบบไม่คงที่

H	1	2	3	4	5	6	7	8	9
1	I	I	I	I	I	I	I	I	I
2	0	I	X	X	X	X	X	X	X
3	0	0	I	β_1	Y_1	Y_2	β_2	Z_1	Z_2

ข) แทนที่ X ในแถวที่สองด้วยตัวเลขจากเมจิกสแควร์แบบตัวต่อตัว โดยหาตำแหน่งของเลข 1 ในเมจิกสแควร์แล้วเลือกตัวเลขทั้งหมดจากตำแหน่งของเลข 1 นี้ (ใช้เลข 1 ด้วย) แทนที่ X ทั้งหมดจากซ้ายไปขวาทีละแถวไปเรื่อย ๆ ระหว่างการแทนที่นี้ถ้ามีตัวเลขที่มากกว่าหรือเท่ากับ p ให้ข้ามตัวเลขตัวนั้นแล้วใช้ตัวเลขตัวถัดไป

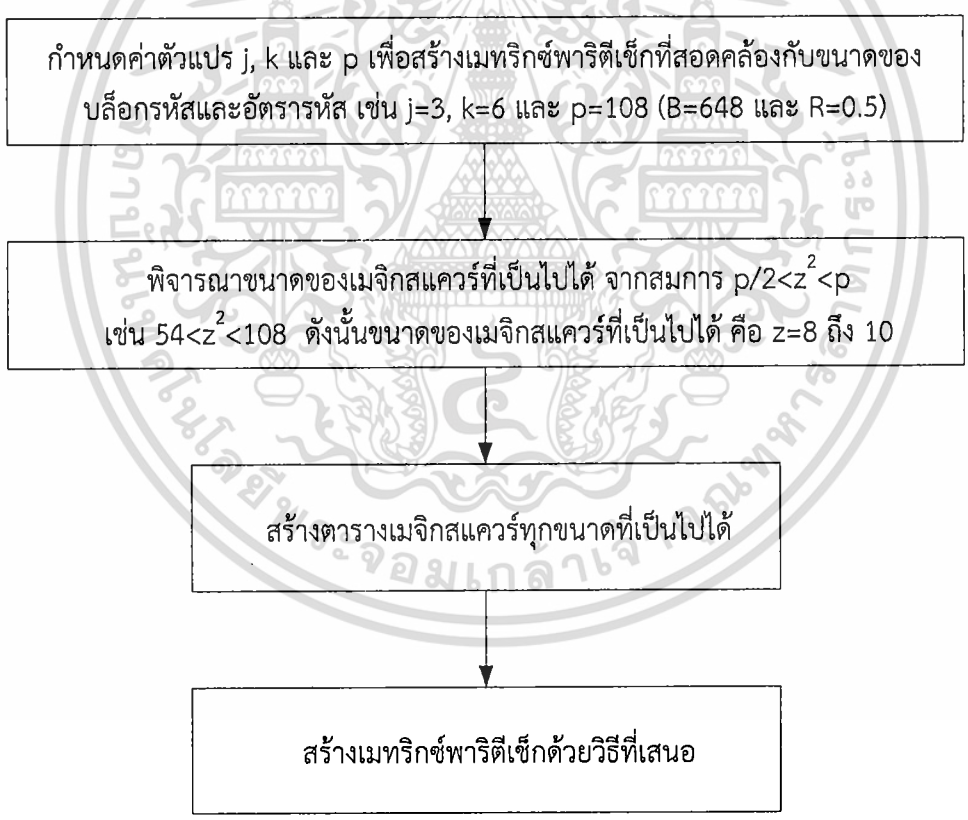
ค) ค่าของ β_1 (แถวที่ 3) จะเป็นค่าที่ได้จาก p/2 (ถ้า p เป็นเลขคู่) หรือได้จาก (p-1)/2 (ถ้า p เป็นเลขคี่) อย่างไรก็ตามถ้าค่าที่คำนวณได้นี้ถูกใช้ไปแล้วให้ใช้ตัวเลขถัดไปที่ใกล้เคียงที่สุด จากนั้นแทนที่ Y_1, Y_2, ...ด้วย β_1-1, β_1-2, ... ตามลำดับ ระหว่างการแทนที่นี้ถ้ามีตัวเลขใดถูกใช้ไปแล้วให้ใช้ตัวเลขที่ต่ำกว่าในตำแหน่งถัดลงไป

ง) ค่าของ β_2 จะเป็นค่าที่ได้จาก $\beta_1/2$ (ถ้า β_1 เป็นเลขคู่) หรือ $(\beta_1-1)/2$ (ถ้า β_1 เป็นเลขคี่ อย่างไรก็ตามถ้าตัวเลขใดถูกใช้ไปแล้วให้ใช้ตัวเลขที่ต่ำกว่าในตำแหน่งถัดลงไป จากนั้นแทนที่ Z_1, Z_2, ... ด้วยβ_2-1, β_2-2, ... ตามลำดับ ระหว่างการแทนที่นี้ถ้ามีตัวเลขใดถูกใช้ไปแล้วให้ใช้ตัวเลขที่ต่ำกว่าในตำแหน่งถัดลงไป ตัวอย่างผลการสร้างโดยใช้เมจิกสแควร์ขนาด 7×7 แสดงดังตารางที่ 4.11

ตารางที่ 4.11 สร้างเมทริกซ์พาริตีเช็กด้วยวิธีที่สาม

H	1	2	3	4	5	6	7	8	9
1	I	I	I	I	I	I	I	I	I
2	0	I	I	10	19	28	38	47	7
3	0	0	I	27	26	25	14	13	12

สรุปขั้นตอนการออกแบบของผลงาน [35] แสดงได้ดังรูปที่ 4.6



รูปที่ 4.6 แสดงขั้นตอนการออกแบบของผลงาน [35]

สามารถแสดงรูปแบบเมทริกซ์พาริตีเช็กได้ดังสมการที่ (4.13) ถึง (4.16)

$$MS = \begin{bmatrix} (u,v) & (u,v+1) & (u,v+2) & \dots & (u,z-1) & (u,z) \\ (u+1,v) & (u+1,v+1) & (u+1,v+2) & \dots & (u+1,z-1) & (u+1,z) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ (z-1,v) & (z-1,v+1) & (z-1,v+2) & \dots & (z-1,z-1) & (z-1,z) \\ (z,v) & (z,v+1) & (z,v+2) & \dots & (z,z-1) & (z,z) \end{bmatrix}_{z \times z} \tag{4.13}$$

H1

$$= \begin{bmatrix} I & I & I & \dots & I & \dots & \dots & I & I & \dots & \dots & I \\ 0 & I & \alpha^{(u,v)} & \alpha^{(u,v+1)} & \dots & \alpha^{(u,z-1)} & \alpha^{(u,z)} & \alpha^{(u+1,v)} & \alpha^{(u+1,v+1)} & \dots & \alpha^{(u+i,z-1)} & \alpha^{(u+i,z)} \\ 0 & 0 & I & \alpha^{(u+2,v)} & \dots & \alpha^{(u+2,z-1)} & \alpha^{(u+2,z)} & \alpha^{(u+3,v)} & \alpha^{(u+3,v+1)} & \dots & \alpha^{(u+i,z-1)} & \alpha^{(u+i,z)} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & I & \alpha^{(z-1,v)} & \dots & \alpha^{(z-1,v-1)} & \alpha^{(z,v)} & \alpha^{(z,v+1)} & \dots & \alpha^{(z,z-1)} & \alpha^{(z,z)} \end{bmatrix}_{j \times k}$$

(4.14)

H2

$$= \begin{bmatrix} I & I & I & \dots & I & \dots & \dots & I & I & \dots & \dots & I \\ 0 & I & \alpha^{(u,v)} & \alpha^{(u,v+1)} & \dots & \alpha^{(u+1,v)} & \alpha^{(u+1,v)} & \alpha^{(u+1,v+2)} & \alpha^{(u+1,v+3)} & \dots & \alpha^{(u+i,z-1)} & \alpha^{(u+i,z)} \\ 0 & 0 & I & \alpha^{(u+2,z)} & \dots & \alpha^{(z-1,v)} & \alpha^{(z-1,v+1)} & \alpha^{(z-1,v+2)} & \alpha^{(z-1,v+3)} & \dots & \alpha^{(z,v)} & \alpha^{(z,v+1)} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & I & \alpha^{(z,z-1)} & \dots & \alpha^{(u,v)} & \alpha^{(u,v+1)} & \alpha^{(u,v+2)} & \dots & \alpha^{(u,z-1)} & \alpha^{(u,z)} \end{bmatrix}_{j \times k}$$

(4.15)

H3

$$= \begin{bmatrix} I & I & I & \dots & I & \dots & \dots & I & I & \dots & \dots & I \\ 0 & I & \alpha^{(u,v)} & \alpha^{(u,v+1)} & \alpha^{(u,v+2)} & \dots & \alpha^{(u,v+(z-1))} & \alpha^{(u+1,v+1)} & \alpha^{(u+1,v+2)} & \dots & \alpha^{(u+i,v+(k-1))} & \alpha^{(u+i,v+k)} \\ 0 & 0 & I & \alpha^{(p/2)} & \alpha^{(p/2)-1} & \dots & \alpha^{(p/2)-(i+1)} & \alpha^{(p/2)-(i+2)} & \alpha^{(p/2)-(i+3)} & \dots & \alpha^{(p/2)-(k-1)} & \alpha^{(p/2)-(k)} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & I & \alpha^{(z,z-1)} & \dots & \alpha^{(u+j,v+(i+1))} & \alpha^{(u+j,v+(i+2))} & \alpha^{(u+j,v+(i+3))} & \dots & \alpha^{(u+j,v+(k-1))} & \alpha^{(u+j,v+k)} \end{bmatrix}_{j \times k}$$

(4.16)

ผลของงานวิจัยแสดงดัง [35] และเสนอผลการทดสอบในบทถัดไป

บทที่ 5
ผลการทดสอบสมรรถนะของงานวิจัย

บทนี้แสดงผลการทดสอบสมรรถนะของรหัส โดยการออกแบบเมทริกซ์พาริตีเช็กสำหรับรหัสแอลดีพีซีที่ได้กล่าวไว้ในบทที่ 4 ทำการออกแบบแล้วดำเนินการทดสอบสมรรถนะในระบบช่องสัญญาณรบกวนแบบเกาส์สีขาว การเข้า-ถอดรหัส ทำให้ง่าย เร็ว ลดความซับซ้อนลง และความสามารถแก้ไขข้อผิดพลาดของข้อมูล โดยทดสอบสมรรถนะด้วยผลการทดสอบลูป 4 ผลการทดสอบการเข้ารหัสผลการทดสอบการถอดรหัสแบบวนซ้ำและผลการเปรียบเทียบกับงานวิจัยอื่น ๆ ที่อยู่ในกลุ่มเดียวกัน และวิเคราะห์ผลในเชิงสถิติประยุกต์

5.1 ระบบช่องสัญญาณรบกวนแบบเกาส์สีขาว

สัญญาณรบกวนแบบเกาส์สีขาว พบมากในระบบการประมวลผลสัญญาณของระบบบันทึกข้อมูล และการสื่อสารข้อมูล ตัวอย่างเช่น สัญญาณรบกวนจากความร้อน ส่งผลให้การทำงานของวงจรภาครับเกิดความผิดพลาดขึ้นดังนั้นในการทดสอบสมรรถนะของรหัสแอลดีพีซี จึงได้ทำการทดสอบภายใต้ระบบที่มีสัญญาณรบกวนแบบเกาส์สีขาวตัวแปรสุ่มแบบเกาส์เซียน (Gaussian Random Variable) เป็นที่นิยมใช้งานในการวิเคราะห์ระบบสื่อสาร ทั้งนี้เนื่องจากพฤติกรรมของตัวแปรสุ่มแบบเกาส์เซียนมีลักษณะคล้ายกับข้อมูลที่รับส่งภายในระบบสื่อสาร เช่น สัญญาณรบกวน และข้อมูลข่าวสาร เป็นต้น ตัวแปรสุ่มแบบเกาส์เซียน x จะมีฟังก์ชันความหนาแน่นความน่าจะเป็นดังสมการที่ (5.1)

$$Px(x) = \frac{1}{\sqrt{2\pi\sigma_x^2}} \exp\left\{-\frac{(x-m_x)^2}{2\sigma_x^2}\right\} \qquad (5.1)$$

เมื่อ exp {.} = ฟังก์ชันเลขชี้กำลัง (Exponential Function)

m_x = ค่าเฉลี่ยตัวที่x

σ_x^2 = ค่าความแปรปรวนหรือความแปรผันตัวที่x

ในทางปฏิบัติ ฟังก์ชันความหนาแน่นของตัวแปรสุ่มแบบเกาส์เซียนจะถูกนิยามด้วยค่าเฉลี่ย m_x และค่าความแปรผัน σ_x^2 สำหรับในกรณีที่ $m_x = 0$ และ$\sigma_x^2 = 1$ จะเรียกฟังก์ชันความหนาแน่น ความน่าจะเป็น ว่ามีลักษณะการแจกแจงปกติแบบมาตรฐาน (Standard Normal Distribution)

5.2 ผลการทดสอบลูป 4

การใช้กราฟแทนเนอร์เป็นวิธีนำเสนอเพื่อให้เข้าใจเกี่ยวกับลูป 4 (Cycle-4 หรือ Cycle of Length 4) ของรหัสแอลดีพีซี กราฟนี้สามารถวาดได้โดยตรงจากเมทริกซ์พาริตีเช็กซึ่งกราฟจะประกอบด้วย m โหนดเช็ก และ n โหนดสัญลักษณ์โดยโหนดเช็ก f_i จะเชื่อมต่ออยู่กับโหนดสัญลักษณ์ c_i ถ้าสมาชิก h_{ij} ของเมทริกซ์พาริตีเช็กเป็นเลข 1 โดยปกติลูปในกราฟแทนเนอร์ จะหมายถึง ชุดเล็ก ๆ ของโหนดที่เชื่อมต่อกัน ลูปจะเริ่มและสิ้นสุดที่โหนดเดียวกัน และจะเป็นไปตามเงื่อนไขที่ว่าจะไม่มีโหนดเกิดขึ้นมากกว่าหนึ่งครั้ง

เป็นที่รู้กันแล้วว่าลูป 4 จะเกิดขึ้นเมื่อจำนวนครั้งในการเลื่อนของเมทริกซ์ย่อย การเปลี่ยนลำดับถูกเลื่อนในจำนวนครั้งที่เท่ากันอย่างน้อยหนึ่งคู่ในแถวเดียวกัน และ/หรืออย่างน้อยสองคู่ในหลักใด ๆ ดังแสดงในตารางที่ 5.1

ตารางที่ 5.1 เมทริกซ์พาริตีเช็กที่มีลูป 4

H	1	2	3	4	5	6	7	8	9
1	1	1	1	1	①	①	1	1	1
2	0	1	x	①	②	②	x	①	x
3	0	0	1	①	x	x	x	①	x

ตัวเลขที่แตกต่างของสมาชิกทุกตัวในเมจิกสแควร์ถูกพิจารณานำมาใช้สร้างเมทริกซ์พาริตีเช็กในการวิจัยนี้ก็เพื่อกำจัดลูป 4 หรือกล่าวได้ว่าถ้าเมทริกซ์พาริตีเช็กแบบโครงสร้างถูกสร้างด้วยตัวเลขที่ไม่ซ้ำกัน ที่สามารถทำให้เกิดขึ้นได้ด้วยระเบียบวิธีที่แน่นอน หรือมีอัลกอริทึมที่แน่นอนเมทริกซ์พาริตีเช็กที่ได้จะไม่มีลูป 4 ซึ่งจะส่งผลให้สมรรถนะดีนั่นเอง

5.3 ผลการทดสอบการเข้ารหัส

พิจารณาจากรหัสบล็อกเชิงเส้นมีความสัมพันธ์ดังสมการที่ (2.2) ในรูปแบบเชิงระบบสามารถแสดงสมการที่ (2.4) ถึง (2.9) ดังนั้น หน้าที่ของตัวเข้ารหัสคือคำนวณหาเมทริกซ์พาริตีเช็ก ที่สามารถถูกผนวกต่อท้ายไปกับข้อความเพื่อสร้างคำรหัสออกมา

ในงานวิจัยนี้ เพื่อให้การเข้า-ถอดรหัสด้วยเมทริกซ์พาริตีเช็ก ที่สร้างขึ้นทำให้ง่ายมากขึ้น วิธีการคำนวณด้วยระเบียบวิธีเชิงตัวเลข (Numerical Method) ที่เรียกว่า “LU decomposition method” ได้ถูกนำมาประยุกต์ใช้ (ผลงานดัง [33]) โดยการแยกเมทริกซ์ให้เป็นสองเมทริกซ์ คือ สามเหลี่ยมล่าง (Upper Triangular Matrix) และสามเหลี่ยมบน (Lower Triangular Matrix) ตามความสัมพันธ์ $[H] = [L][U]$ แสดงดังสมการที่ (5.2)

$$\begin{bmatrix} l_{11} & 0 & \cdots & 0 \\ l_{21} & l_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ l_{n1} & l_{n2} & \cdots & l_{nn} \end{bmatrix} \underbrace{\begin{bmatrix} u_{11} & u_{12} & \cdots & u_{1n} \\ 0 & u_{22} & \cdots & u_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & u_{nn} \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \\ \vdots \\ p_n \end{bmatrix}}_{Y} = \begin{bmatrix} m_1 \\ m_2 \\ \vdots \\ m_n \end{bmatrix} \quad (5.2)$$

ดังแสดงในสมการ (5.2) ให้ [Y] = [U][P]และเมื่อทำการแทนไปข้างหน้า (Forward Substitution) เพื่อแก้สมการ [L] [Y] = [M] ในสมการที่ (5.3)

$$\begin{bmatrix} l_{11} & 0 & \cdots & 0 \\ l_{21} & l_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ l_{n1} & l_{n2} & \cdots & l_{nn} \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} m_1 \\ m_2 \\ \vdots \\ m_n \end{bmatrix} \quad (5.3)$$

สุดท้ายทำการแทนค่าย้อนกลับ (Backward Substitution) เพื่อแก้สมการ [U] [P] = [Y] ในสมการที่ (5.4) จะได้เวกเตอร์ $\{p_i\}$ ที่ต้องการ

$$\begin{bmatrix} u_{11} & u_{12} & \cdots & u_{1n} \\ 0 & u_{22} & \cdots & u_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & u_{nn} \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \\ \vdots \\ p_n \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} \quad (5.4)$$

5.4 ผลการทดสอบการถอดรหัสแบบวนซ้ำ

มีวิธีการถอดรหัสหลายวิธีที่ใช้กับรหัสแอลดีพีซี ซึ่งแต่ละวิธีได้มาจากการกระบวนการคิดที่แตกต่างกันเล็กน้อย ตัวอย่างวิธีการถอดรหัส เช่น วิธี Believe Propagation (BP) วิธี Sum-Product (SP) และวิธี Message Passing (MP) เป็นต้น

สามารถวาดกราฟแทนเนอร์ได้โดยตรงจากเมทริกซ์พาริตีเช็กดังแสดงในรูปที่ 2.4 ซึ่งจากรูปจะเห็นว่ากราฟประกอบด้วย m โหนดเช็ก และ n โหนดสัญลักษณ์ โดยโหนดเช็ก f_i จะเชื่อมต่ออยู่กับโหนดสัญลักษณ์ c_i ถ้าสมาชิก h_{ij} ของเมทริกซ์พาริตีเช็กเป็นเลข 1 ดังสมการที่ 2.1

ในอัลกอริทึม Log-Domain Sum-Product ข้อความจะส่งผ่านระหว่างโหนดเช็กและโหนดสัญลักษณ์ในแต่ละรอบการส่ง ค่าความน่าเป็นไปได้หรือค่าความน่าเชื่อถือ (Log Likelihood Ratio) จะถูกบันทึกสำหรับความน่าจะเป็นที่เป็นเหมือนสัญลักษณ์ของมัน โดยสรุปก็คือ ตัวถอดรหัสดำเนินการ 5 ขั้นตอนดังสมการที่ (2.10) ถึง (2.17) ผลงานแสดงดัง [33-38]

5.5 วิเคราะห์ผลในเชิงสถิติประยุกต์

ความแปรผัน (Variance) เป็นการวัดการกระจายของข้อมูลว่ามีการกระจายเป็นอย่างไร สำหรับงานวิจัยนี้ ค่าความแปรปรวนหรือความแปรผันแสดงถึงการกระจายตัวของตัวเลขที่เป็นสมาชิกของเมทริกซ์พาริตีเช็ก ว่าพวกมันถูกทำให้กระจายออกจากค่าเฉลี่ย (Mean Value) ของพวกมันอย่างไร โดยค่าความแปรผันถูกคำนวณเหมือนกับเป็นการเบี่ยงเบนกำลังสองเฉลี่ย (Average Squared Deviation) ของตัวเลขแต่ละตัวที่เบี่ยงเบนไปจากค่าเฉลี่ยของมัน โดยทั่วไปการวัดความแปรผันจะใช้บ่งบอกลักษณะของข้อมูลนั่นคือ ถ้าค่าความแปรผันของข้อมูลชุดนั้นมีค่าสูงแสดงว่าข้อมูลชุดนั้น มีความแตกต่างกันมาก ค่าเบี่ยงเบนมาตรฐาน (Standard Deviation :S.D.) เป็นค่าที่มีการพิสูจน์แล้วว่าเป็นการวัดการกระจายของข้อมูลที่มีประโยชน์มากเนื่องจากเป็นสมการที่ปรับเปลี่ยนได้เชิงคณิตศาสตร์ ค่าเบี่ยงเบนมาตรฐานสามารถแสดงดังสมการที่ (3.3) หรือ (3.4)

อย่างไรก็ตามเพื่อเป็นการเปรียบเทียบการกระจายของชุดข้อมูลกับค่าเฉลี่ยที่แตกต่างกันอย่างเห็นได้ชัด การใช้ค่าสัมประสิทธิ์ของการเปลี่ยนแปลง (Coefficient of Variation : C.V.) จึงดีกว่าค่าเบี่ยงเบนมาตรฐานธรรมดา โดยคำนวณได้ดังสมการที่ (3.7)

ค่าสัมประสิทธิ์ของการเปลี่ยนแปลงสามารถถูกนำไปใช้เป็นเงื่อนไขสำหรับการเลือกว่าเมทริกซ์พาริตีเช็กที่สร้างขึ้น เมทริกซ์ใดดีที่สุดได้ ซึ่งตัวแปรเชิงสถิตินี้จะเป็นตัวชี้วัดว่าการกระจายตัวหรือตำแหน่งของเลข “1” ในเมทริกซ์พาริตีเช็กเป็นอย่างไร กล่าวคือ ถ้าเมทริกซ์พาริตีเช็กใดที่มีค่าสัมประสิทธิ์ของการเปลี่ยนแปลงสูงกว่าแสดงว่ามีการกระจายตัวของเลขหนึ่งได้ดีกว่านั่นเอง

ผลการจำลองสมรรถนะของรหัสที่ได้จากการสร้างด้วยวิธีที่หนึ่งถึงสามแสดงดังรูปที่ 5.5 ถึง 5.7 ตามลำดับ แสดงให้เห็นว่ารหัสแอลดีพีซีที่ใช้เมทริกซ์พาริตีเช็ก จากการสร้างด้วยวิธีที่หนึ่งมีสมรรถนะต่ำกว่าวิธีอื่นทุกชนิดของเมจิกสแควร์ ซึ่งเป็นผลมาจากความไม่เป็นอิสระของตัวเลขจากเมจิกสแควร์ที่นำมาใช้ ดังที่ [39] และ [40] ได้กล่าวไว้เกี่ยวกับความขึ้นต่อกันของข่าวสารนั้น ในขั้นตอนการถอดรหัสแบบวนซ้ำจะส่งผลต่อสมรรถนะที่ไม่ดีของรหัสแอลดีพีซี ดังนั้นการใช้ตัวเลขที่นำมาจากเมจิกสแควร์ทั้งค่าและตำแหน่งที่มีความสัมพันธ์กันนี้จะไม่ทำให้ได้สมรรถนะของรหัสที่ดีที่สุด วิธีที่สามให้สมรรถนะดีกว่าวิธีที่สองในทุกชนิดของเมจิกสแควร์ทั้งนี้เนื่องจากตัวเลขจากเมจิกสแควร์ที่นำมาใช้ในวิธีที่สามนี้ ถูกทำให้เป็นอิสระต่อกันและความเป็นปกติ (Normality) ของกราฟการกระจายตัวของตัวเลข (Number Distribution Curve) มีค่าสูงกว่าวิธีอื่น ซึ่งความเป็นปกตินี้ได้ถูกประมาณการโดยวิธีการจำลองความน่าจะเป็น (Probability Plot) ซึ่งค่าความน่าจะเป็นที่ได้สำหรับวิธีที่หนึ่งถึงสามมีค่าเท่ากับ 0.143, 0.249 และ 0.336 ตามลำดับ เมื่อใช้โปรแกรม มินิแท็บ (Mini Tab) ที่เป็นโปรแกรมสำเร็จรูปใช้หาค่าของชุดข้อมูลว่ามีการกระจายตัวเข้าใกล้โค้งปกติกี่เปอร์เซ็นต์ ซึ่งจะเห็นได้ว่ารหัสที่สร้างโดยเมจิกสแควร์ขนาด

6×6 จะให้สมรรถนะสูงที่สุดในทุกๆ วิธีการสร้าง สังเกตได้จากค่าสัมประสิทธิ์ของการเปลี่ยนแปลงของเมทริกซ์พาริตีเช็กที่สร้างโดยใช้เมจิกสแควร์ขนาด 6×6 มีค่าน้อยกว่าเมจิกสแควร์ขนาดอื่น

การใช้หลักการทางสถิติมาประยุกต์ใช้เพื่อประกอบในการพิจารณาตัดสินใจนั้น เพื่อดูสภาพการกระจายตัวของเลขหนึ่งในเมทริกซ์พาริตีเช็กเป็นแบบปกติหรือไม่ จากผลการทดสอบทำให้เห็นถึงความสัมพันธ์กันว่า ถ้าการกระจายตัวของเลขหนึ่งในเมทริกซ์พาริตีเช็กเข้าใกล้โค้งแบบปกติมากกว่าจะส่งผลให้สมรรถนะดีกว่าเมทริกซ์พาริตีเช็กที่มีค่าการกระจายตัวของเลขหนึ่งที่เข้าใกล้โค้งแบบปกติน้อยกว่า

5.6 ผลการเปรียบเทียบกับงานวิจัยอื่น

5.6.1 จากการออกแบบเมทริกซ์พาริตีเช็กโดยพิจารณาจากการปรับรูปแบบใหม่ด้วยการลดเลขหนึ่งลงตามแนวเส้นทแยงมุมซ้ายและขวาของเมทริกซ์พาริตีเช็ก

ในการศึกษาเบื้องต้นกับการถอดรหัสที่มีความยาวบล็อกยาว เช่น 4096 บิตจากการศึกษาผลงานวิจัย [33] และ [36] พบว่า สมรรถนะของรหัสที่สร้างขึ้นต่ำกว่าสมรรถนะของรหัส [11] และ [16] ที่ความยาวบล็อกยาว แต่สำหรับรหัสที่มีความยาวบล็อกสั้นและปานกลางสมรรถนะของรหัสที่ออกแบบในงานวิจัยนี้มีสมรรถนะที่ดี ภายใต้ตัวแปรการทดสอบที่แสดงดังตารางที่ 5.2 ถึง 5.4

ตารางที่ 5.2 ตัวแปรออกแบบสำหรับรหัสบล็อกสั้น

ตัวแปร	ค่าตัวแปรที่ใช้ทดสอบ			
j	3	3	4	4
k	15	18	15	18
p	17	29	17	29
อัตรารหัส=1-(j/k)	0.80	0.833	0.733	0.778
บิตข้อมูล=p(k-j)	204	435	187	406
บิตพาริตี=jp	51	87	68	116
คำรหัส=kp	255	522	255	522
รอบการวนซ้ำ	5, 10, 20			

เมทริกซ์พาริตีเช็กที่ออกแบบในงานวิจัยนี้ ถูกนำไปใช้กับรหัสแอลดีพีซีแบบไม่คงที่ที่มีความยาวบล็อกสั้น ปานกลางและยาว ผลการทดสอบสมรรถนะของรหัสแสดงดังรูปที่ 5.1 จากรูปแสดงให้เห็นว่ารหัสที่มีความยาวบล็อกยาวจะยิ่งมีสมรรถนะดีส่วนรหัสที่เสนอในงานวิจัย [33] และ [36] จะมีสมรรถนะดี

ที่เป็นไปได้เมื่อนำไปใช้กับรหัสบล็อกความยาวสั้นแสดงดังรูปที่ 5.2 แต่ที่ความยาวบล็อกมากสมรรถนะด้อยกว่ารหัสที่เปรียบเทียบเล็กน้อย

ตารางที่ 5.3 ตัวแปรออกแบบสำหรับรหัสบล็อกปานกลาง

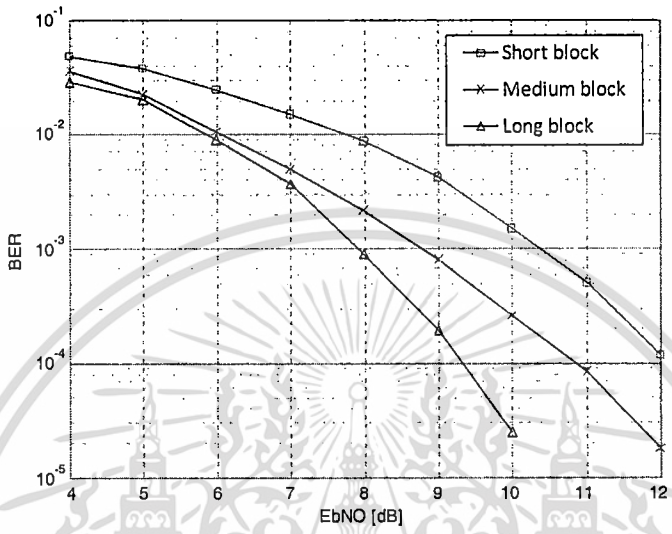
ตัวแปร	ค่าตัวแปรที่ใช้ทดสอบ			
j	3	3	4	4
k	31	43	31	43
p	37	47	37	47
อัตรารหัส=1-(j/k)	0.903	0.93	0.871	0.907
บิตข้อมูล=p(k-j)	1,036	1,880	999	1,833
บิตพาริตี=jp	111	141	148	188
คำรหัส=kp	1,147	2,021	1,147	2,021
รอบการวนซ้ำ	5, 10, 20			

ตารางที่ 5.4 ตัวแปรออกแบบสำหรับรหัสบล็อกยาว

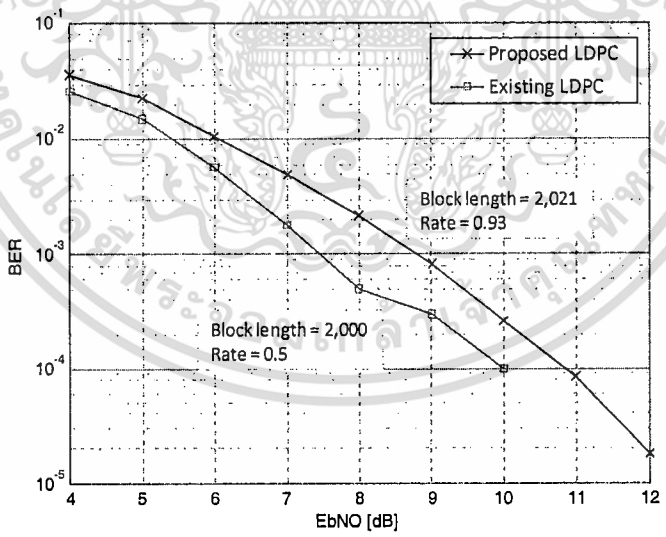
ตัวแปร	ค่าตัวแปรที่ใช้ทดสอบ			
j	4	4	5	5
k	61	47	61	47
p	67	89	67	89
อัตรารหัส=1-(j/k)	0.934	0.915	0.918	0.894
บิตข้อมูล=p(k-j)	3,819	3,827	3,752	3,738
บิตพาริตี=jp	268	356	335	445
คำรหัส=kp	4,087	4,183	4,087	4,183
รอบการวนซ้ำ	10, 20, 30			

การศึกษาได้ทำการเปรียบเทียบสมรรถนะของรหัสที่เสนอในงานวิจัย [33] กับผลการศึกษาของ Rakibul และคณะ [16] ที่ขนาดความยาวบล็อกใกล้เคียงกัน ผลการศึกษาแสดงให้เห็นว่า รหัสที่นำมาเปรียบเทียบให้สมรรถนะดีกว่ารหัสที่เสนอในงานวิจัยนี้เล็กน้อย อย่างไรก็ตามอัตรารหัสของ [16]

มีค่าเพียง 0.5 เท่านั้น ขณะที่อัตรารหัสในงานวิจัย [33] คือ 0.93 รูปที่ 5.2 แสดงสมรรถนะของรหัส [33] กับรหัส [16]



รูปที่ 5.1 สมรรถนะของรหัสความยาวบล็อกสั้น ปานกลางและยาว



รูปที่ 5.2 ผลการเปรียบเทียบของรหัส[33] กับรหัส[16]

ในงานวิจัยนี้ ยังได้ศึกษาเกี่ยวกับจำนวนครั้งการวนซ้ำที่มีผลต่อสมรรถนะของรหัส โดยทำการทดสอบที่จำนวนรอบการวนซ้ำเท่ากับ 5, 10 และ 20 รอบผลการศึกษาพบว่าที่จำนวนการวนซ้ำ

เท่ากับ 5 รอบ ค่าอัตราบิตผิดพลาดของรหัสทั้งที่มีความยาวบล็อกสั้นและความยาวบล็อกปานกลางมีค่าใกล้เคียงกัน และค่าอัตราบิตผิดพลาดสามารถทำให้ดีขึ้นได้โดยเพิ่มจำนวนรอบการวนซ้ำเป็น 10 รอบ และเมื่อเพิ่มจำนวนรอบการวนซ้ำเป็น 20 รอบ พบว่า ค่าอัตราบิตผิดพลาดของรหัสที่มีความยาวบล็อกปานกลางเท่านั้นที่เพิ่มขึ้นเพียงเล็กน้อย อย่างไรก็ตามค่าอัตราบิตผิดพลาดที่ได้ก็ไม่ต่ำมากนักเป็นธรรมดาที่รหัสแอลดีพีซีที่ความยาวบล็อกไม่ยาวจะมีค่าอัตราบิตผิดพลาดไม่ดีแต่สำหรับผู้วิจัยเห็นว่ามันเป็นสิ่งที่ท้าทาย ควรหาวิธีออกแบบให้มีสมรรถนะดีให้ได้ต่อไป

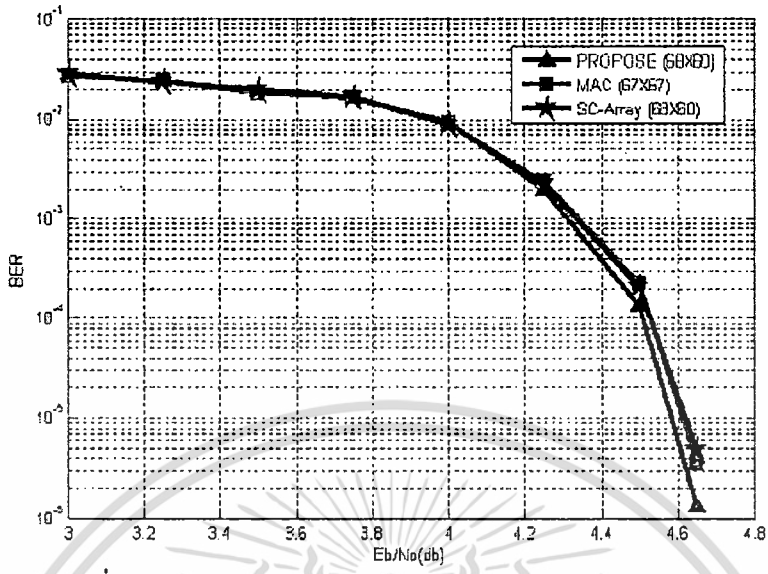
5.6.2 จากการออกแบบเมทริกซ์พาริตีเช็กโดยพิจารณากำหนดค่าตัวแปรกับรูปแบบสมมาตรบนหลักการทรานสโพรสเมทริกซ์พาริตีเช็ก

รหัสที่เสนอในงานวิจัย [34] ถูกเปรียบเทียบกับรหัสแอลดีพีซีแบบ [10] และ [24] โดย สมรรถนะของเมทริกซ์พาริตีเช็กที่สร้างขึ้นได้ถูกศึกษาที่ความยาวบล็อกประมาณ 4,000 บิต ซึ่งเป็นความยาวบล็อกที่นำไปประยุกต์ใช้ในระบบบันทึกข้อมูลแบบแม่เหล็กที่มีขนาดเซกเตอร์ (Sector) เท่ากับ 512 ไบต์ เพื่อเปรียบเทียบกับ [10] และ [24] เราจึงใช้เมทริกซ์ย่อยขนาด 68 ตัวแปรการทดสอบแสดงดังตารางที่ 5.5 โดยใช้จำนวนครั้งวนซ้ำเท่ากับ 30 ระยะห่างต่ำสุดของงานวิจัย [10] ศึกษาที่ d_{min}=336 งานวิจัย [34] มีค่าระยะห่างต่ำสุดเท่ากับ 341 จึงแสดงให้เห็นว่างานวิจัย [34] ควรมีความสามารถในการตรวจจับและแก้ไขข้อผิดพลาดที่สูงกว่า

ตารางที่ 5.5 ตัวแปรที่ออกแบบกับเมทริกซ์ [34]

แบบรหัส	j	k	q	R	ความยาวบล็อก (บิต)
[10]	5	61	67	0.918	4087
[24]	5	60	68	0.917	4080
[34]	5	60	68	0.917	4080

รูปที่ 5.3 แสดงผลการเปรียบเทียบสมรรถนะของรหัสที่เสนอในงานวิจัย [34] ถูกเปรียบเทียบกับรหัสแอลดีพีซีแบบ [10] และ [24] ที่จำนวนรอบการวนซ้ำเท่ากับ 30 รอบพบว่า สมรรถนะของงานวิจัย [34] มีสมรรถนะดีกว่าเล็กน้อยเมื่อค่าอัตราสัญญาณต่อสัญญาณรบกวนสูงกว่า 4.2 dB ที่นำมาเปรียบเทียบรหัส [10] และ [24] สิ่งที่ดีกว่าอีกประเด็นคือการกำหนดขนาดของเมทริกซ์พาริตีเช็กได้มากกว่า เนื่องจากตัวแปรที่กำหนดในการสร้างเมทริกซ์พาริตีเช็กแทนที่จะเป็นจำนวนเฉพาะ งานวิจัย [34] สามารถสร้างจากจำนวนเต็มบวกที่ไม่จำกัดเพียงแค่จำนวนเฉพาะโดยใช้ตัวแปรออกแบบดังตารางที่ 5.6 เพื่อทดสอบอัตรารหัสที่แตกต่างกัน ผลแสดงดังรูปที่ 5.4



รูปที่ 5.3 เปรียบเทียบสมรรถนะของรหัส[10] [24] และ[34]

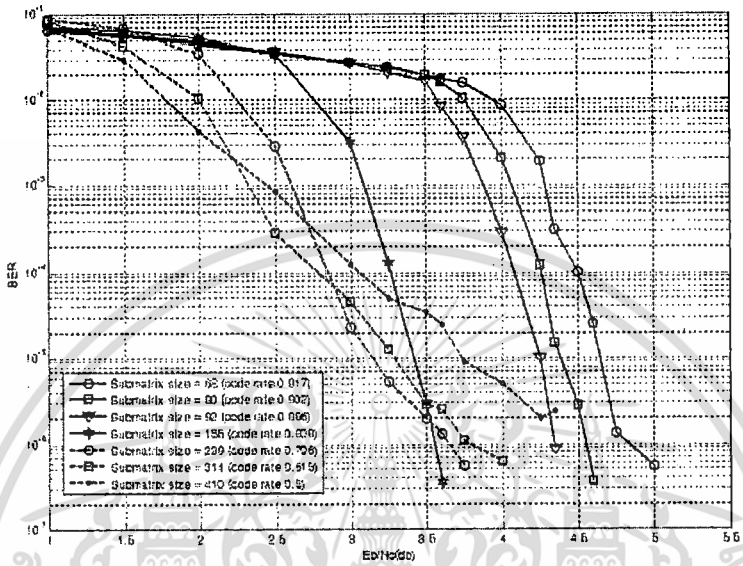
เมทริกซ์พาริตีเช็กที่ถูกปรับแต่งในงานวิจัย [34] ทดสอบที่ความยาวบล็อกขนาดเฉพาะเจาะจง คือ ที่ประมาณ 4080 แต่ทดสอบที่ขนาดเมทริกซ์ย่อยที่แตกต่างกัน ซึ่งจะส่งผลต่อการเปลี่ยนแปลงค่าอัตรารหัส ดังนั้นเราจึงสนใจอัตรารหัสที่ดีกว่า 0.5 ตัวแปรในการทดสอบแสดงดังตารางที่ 5.6

ตารางที่ 5.6 กำหนดค่าเพื่อทดสอบอัตรารหัสที่แตกต่างกัน

ความยาวบล็อก (บิต)	j	k	q	R	รอบการวนซ้ำ
4080	5	60	68	0.917	30
4080	5	51	80	0.902	30
4048	5	44	92	0.886	30
4030	5	26	155	0.808	30
4063	5	17	239	0.706	30
4082	5	13	314	0.615	30
4100	5	10	410	0.5	30

ด้วยจำกัดจำนวนรอบการวนซ้ำที่ 30 ผลการทดสอบแสดงดังรูปที่ 5.4 โดยทำการเปลี่ยนขนาดของเมทริกซ์ย่อยจาก 68 ถึง 155 จากรูปจะเห็นได้ว่าสมรรถนะของรหัสสามารถถูกทำให้ดีขึ้นได้เมื่อขนาด

เมทริกซ์ย่อยมีขนาดเพิ่มขึ้น ที่อัตรารหัส 0.8 สมรรถนะจะลดลงเมื่อขนาดของเมทริกซ์ย่อยมากกว่า 155 ดังนั้น สมรรถนะที่ดีจะสามารถทำให้เกิดขึ้นได้เมื่อขนาดของเมทริกซ์ย่อยมีขนาดเพียงช่วง ๆ หนึ่งเท่านั้น



รูปที่ 5.4 ผลการทดสอบอัตรารหัสที่แตกต่างกัน

งานวิจัย [34] อธิบายเกี่ยวกับการออกแบบรหัสที่มีอัตรารหัสสูงที่มีขนาดความยาวบล็อกเหมาะสำหรับระบบบันทึกข้อมูลแบบแม่เหล็กซึ่งความยาวบล็อกที่ใช้งานจริงในปัจจุบันมีค่าประมาณ 4,000 บิต อย่างไรก็ตาม ขนาดในทางปฏิบัติถูกคำนวณบนพื้นฐานของอัตราข้อมูลที่จำกัด (Mbit/Sec) ซึ่งทางปฏิบัติต้องหาค่าที่เหมาะสมที่สุดด้วยตัวแปรหลาย ๆ ตัว เพื่อให้เงื่อนไขขอบเขตในการออกแบบให้ง่าย จึงพยายามออกแบบรหัสที่รองรับทุกความยาวบล็อกของเมทริกซ์พาริตีเช็กที่เสนอออกแบบบนพื้นฐานของคุณสมบัติความสมมาตรแทนที่จะเป็นการสับเปลี่ยน แม้ว่ารหัสที่ออกแบบได้นั้นมีสมรรถนะดีพอ ๆ หรือดีกว่าเล็กน้อยเมื่อเทียบกับสมรรถนะของ [10] และ [24] แต่จากผลการศึกษาแสดงให้เห็นว่าขนาดของเมทริกซ์ย่อยมีผลกระทบต่อสมรรถนะของรหัส ขนาดเมทริกซ์ย่อยที่ใหญ่ขึ้นจะทำให้ระยะห่างต่ำสุดมีค่าสูงขึ้น และผลก็คือทำให้สมรรถนะดีขึ้น อย่างไรก็ตามขนาดเมทริกซ์ย่อยควรต่ำกว่า 155 นอกจากนี้เมทริกซ์ที่เสนอในงานวิจัยนี้ ยังคงมีคุณสมบัติที่เหมาะสำหรับระบบบันทึกข้อมูลแบบแม่เหล็กเช่น กระบวนการเข้ารหัสมีความซับซ้อนต่ำ มีค่าระดับความผิดพลาดต่ำ และมีความสามารถในการตรวจจับและแก้ไขข้อผิดพลาดแบบหลายบิตติดกันได้

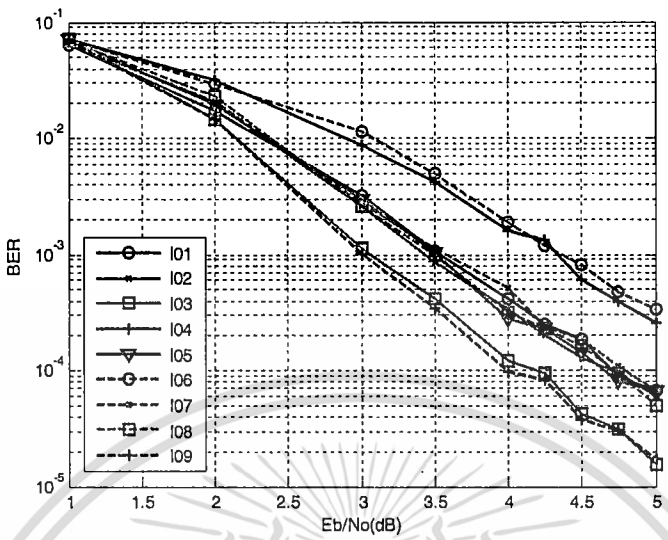
5.6.3 จากการออกแบบเมทริกซ์พาริตีเช็กโดยประยุกต์ทฤษฎีคณิตศาสตร์พื้นฐาน เรื่องเมจิกสแควร์เพื่อให้ได้อัลกอริทึมในการสร้างเมทริกซ์บนพื้นฐานของ MSBA ที่ดีที่สุดเมทริกซ์พาริตีเช็กที่ได้จาก

การสร้างมีวิธี 3 วิธีจะถูกนำมาทดสอบว่าสมรรถนะของรหัสที่ได้จากวิธีการสร้างเมทริกซ์พาริตีเช็กวิธีใดมีสมรรถนะดีที่สุด โดยการทดสอบจะทดสอบภายใต้เงื่อนไขเดียวกันทั้งสามวิธี กล่าวคือภายใต้ความยาวบล็อก อัตรารหัสและจำนวนรอบการวนซ้ำเดียวกัน และใช้ขนาดเมจิกสแควร์ที่ใกล้เคียงกันมากที่สุด ดังนั้นเพื่อให้ครอบคลุมจำนวนเมจิกสแควร์ที่มีทั้งหมดที่กล่าวในหัวข้อที่ผ่านมา จะได้เมทริกซ์พาริตีเช็กทั้งหมด 27 เมทริกซ์ ดังแสดงในตารางที่ 5.7 ซึ่งผลการทดสอบสมรรถนะรหัสด้วยการจำลองทั้งหมดจะถูกนำมาเปรียบเทียบกันดังแสดงในรูปที่ 5.5 ถึง 5.7

ตารางที่ 5.7 รายการเมจิกสแควร์สำหรับความยาวบล็อก 513

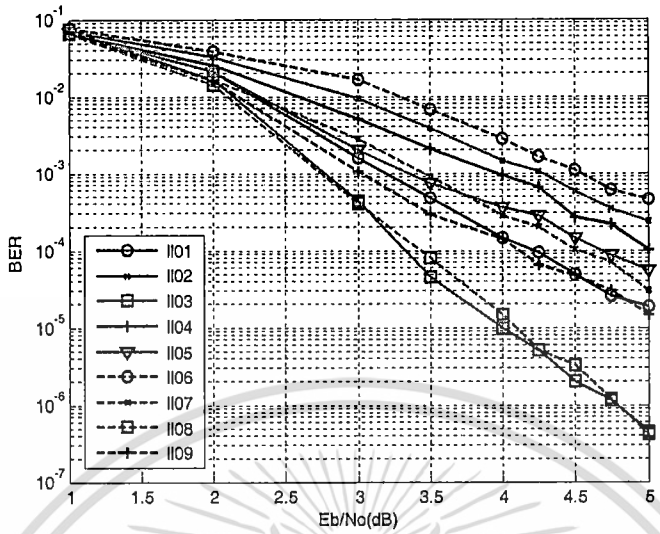
กลุ่ม	ชนิด	หมายเลข	ขนาด (zxz)
G.1	Jupiter	I 01,II 01,III 01	4x4
	Mars	I 02,II 02,III 02	5x5
	Sol	I 03,II 03,III 03	6x6
	Venus	I 04,II 04,III 04	7x7
G.2	Odd order	I 05,II 05,III 05	5x5
		I 06,II 06,III 06	7x7
G.3	Doubly even	I 07,II 07,III 07	4x4
G.4	Singly even	I 08,II 08,III 08	6x6 (LUX)
		I 09,II 09,III 09	6x6 (Strachey)

ในบรรดากราฟที่แสดงสมรรถนะทั้งหมดจะเห็นได้ว่าสมรรถนะที่ดีที่สุดได้มาจากการสร้างด้วยวิธีที่สามและใช้เมจิกสแควร์ขนาด 6x6 เป็นเมจิกสแควร์ในกลุ่มที่ 4 (หารด้วย 4 ไม่ลงตัว และเป็นเมจิกสแควร์ที่สร้างด้วย Strachey method) ซึ่งเป็นที่เห็นได้ชัดเจนว่ามีเมจิกสแควร์ที่เหมาะสมเพียงหนึ่งเมจิกสแควร์เท่านั้น สำหรับขนาดบล็อกและอัตรารหัสนี้ ไม่ใช่ว่าเราจะสามารถใช้เมจิกสแควร์ทั้งหมดได้ ซึ่งเราสามารถขยายการศึกษาในลักษณะนี้สำหรับรหัสที่มีความยาวบล็อกและอัตรารหัสอื่นต่อไปได้

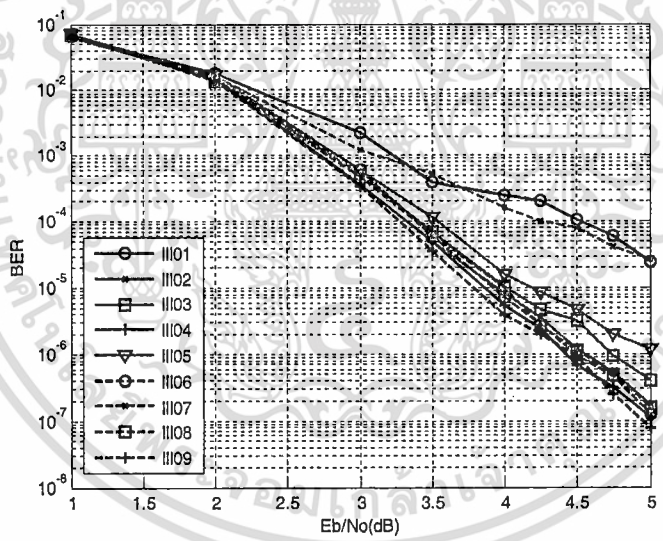


รูปที่ 5.5 ผลการทดสอบรหัสสร้างด้วยวิธีที่หนึ่ง(R=0.7)

เมจิกสแควร์ขนาด 6×6 (กลุ่มที่ 4) สามารถสร้างได้โดยด้วยระเบียบวิธี 2 ระเบียบวิธี คือ วิธีของ Strachey (เรียกว่า Strachey Magic Square : SMS) และวิธี LUX (เรียกว่า LUX Magic Square : LMS) ผลการศึกษาพบว่าสมรรถนะของรหัสที่ได้จากการใช้ตัวเลขจากเมจิกสแควร์ทั้งสองนี้ มีความแตกต่างกัน กล่าวคือ รหัสที่สร้างจากการใช้ SMS จะให้สมรรถนะที่ดีกว่า รหัสที่สร้างจาก LMS โดยค่า BER ของรหัสทั้งสองมีค่าประมาณ 10^{-7} และ 10^{-6} ตามลำดับ ค่าสัมประสิทธิ์ของการเปลี่ยนแปลงสามารถนำไปใช้ในการจำแนกว่าเมทริกซ์พาริตีเช็กใดเหมาะสมที่จะนำไปใช้กว่ากัน (ค่าต่ำกว่าจะดีกว่า) ซึ่งจะเห็นว่าค่าสัมประสิทธิ์ของการเปลี่ยนแปลงของ เมทริกซ์พาริตีเช็กที่สร้างด้วยการนำตัวเลขจาก SMS มาใช้มีค่าต่ำกว่า เมทริกซ์พาริตีเช็กที่สร้างด้วย LMS (มีค่า 66.5% และ 69.9% ตามลำดับ) หรือกล่าวได้ว่าการกระจายตัวที่สม่ำเสมอของสมาชิกในเมทริกซ์ที่มากขึ้น จะทำให้สมรรถนะของรหัสสูงขึ้นนั่งเอง



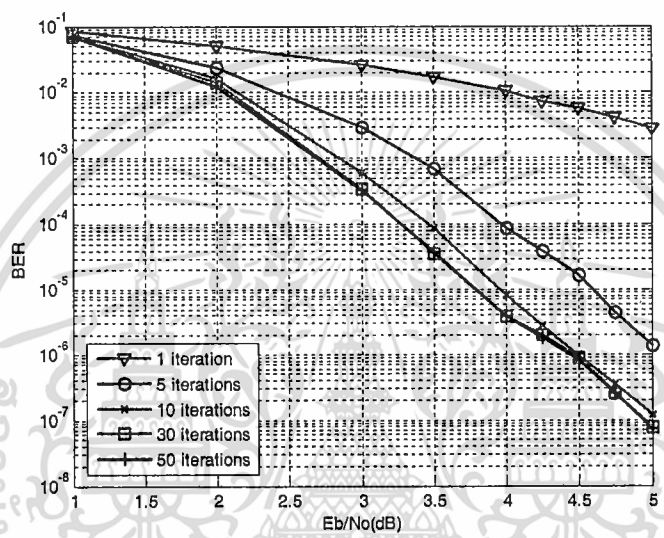
รูปที่ 5.6 ผลการทดสอบรหัสสร้างด้วยวิธีที่สอง(R=0.7)



รูปที่ 5.7 ผลการทดสอบรหัสสร้างด้วยวิธีที่สาม(R=0.7)

หลังจากได้เมจิกสแควร์ที่ต้องการแล้ว (ในกรณีนี้ คือเมจิกสแควร์ SMS ขนาด 6×6) การทดสอบเพื่อประเมินเกี่ยวกับประสิทธิภาพการถอดรหัสแบบวนซ้ำ (Iterative Decoding) ของรหัสแอลดีพีซีที่เสนอนี้ ว่ามีความรวดเร็วหรือยากง่ายเพียงใดซึ่งจะพิจารณาได้จากจำนวนรอบของการวนซ้ำ ซึ่งการทดสอบจะทำการเปลี่ยนค่าจำนวนรอบในการวนซ้ำว่าส่งผลต่อสมรรถนะรหัสมากน้อยเพียงใด โดยค่าจำนวนครั้งในการวนซ้ำ มีค่าดังนี้ 1, 5, 10, 30 และ 50 ครั้ง ที่รหัสความยาวบล็อกประมาณ 500 และ

อัตรารหัสประมาณ 0.7 ผลการศึกษาแสดงดังรูปที่ 5.8 จากผลการศึกษาแสดงให้เห็นว่าสมรรถนะของรหัสจะค่อย ๆ ดีขึ้น เมื่อจำนวนรอบการวนซ้ำเพิ่มขึ้น จนถึงระดับหนึ่งคือ 30 รอบสมรรถนะ จะคงที่เมื่อจำนวนรอบในการวนซ้ำสูงกว่า 30 รอบ นั่นคือ ค่าจำนวนรอบในการวนซ้ำที่เหมาะสมที่สุดสำหรับรหัสแอลดีพีซีบนพื้นฐานของ MSBA ที่เสนอนี้มีค่าเท่ากับ 30 รอบ ซึ่งจะนำไปใช้ในการศึกษาขั้นตอนต่อไปทั้งหมดในการศึกษานี้



รูปที่ 5.8 สมรรถนะจำนวนรอบการวนซ้ำ (ความยาว 513 เมจิกสแควร์ 6×6)

จากผลการศึกษาที่ผ่านมาทำให้เราทราบว่า รหัสที่มีความยาวบล็อกเท่ากับ 513 และอัตรารหัสประมาณ 0.7 ที่สร้างบนพื้นฐาน MSBA วิธีที่สามนั้นให้สมรรถนะดีที่สุด และจำนวนรอบการวนซ้ำเท่ากับ 30 รอบ มีความเหมาะสมกับรหัสนี้มากที่สุด การศึกษาขั้นต่อไปจะเป็นการศึกษาว่าในกรณีที่รหัสมีความยาวบล็อกที่ยาวขึ้น(ในขณะที่อัตรารหัสเท่าเดิม)นั้น เมจิกสแควร์ที่เหมาะสมที่สุด (ตัวแปรต้น) จะเป็นเมจิกสแควร์เดิมหรือไม่และส่งผลอย่างไรต่อสมรรถนะของรหัส (ตัวแปรตาม) ภายใต้สภาวะ (ตัวแปรควบคุม)เดียวกัน (ดังแสดงในตารางที่ 5.8 และ 5.9 ที่ความยาวบล็อกเท่ากับ 1010) ผลการศึกษาพบว่า สมรรถนะการถอดรหัสที่ดีที่สุดได้จากการใช้เมจิกสแควร์ ขนาด 9×9 (กลุ่มที่ 2) ดังแสดงในรูปที่ 3.2 จึงกล่าวได้ว่าเมื่อความยาวบล็อกเปลี่ยนทั้งกลุ่มและขนาดของเมจิกสแควร์ที่เหมาะสมที่สุดสำหรับแต่ละความยาวบล็อกอาจเปลี่ยนด้วย ซึ่งแสดงว่าเมจิกสแควร์สามารถนำมาใช้ในการประยุกต์เพื่อออกแบบเมทริกซ์พาริตีเช็กได้โดยกำหนดตัวแปรในการสร้างให้สอดคล้องกับขนาดความยาวบล็อก และอัตรารหัสที่ต้องการ

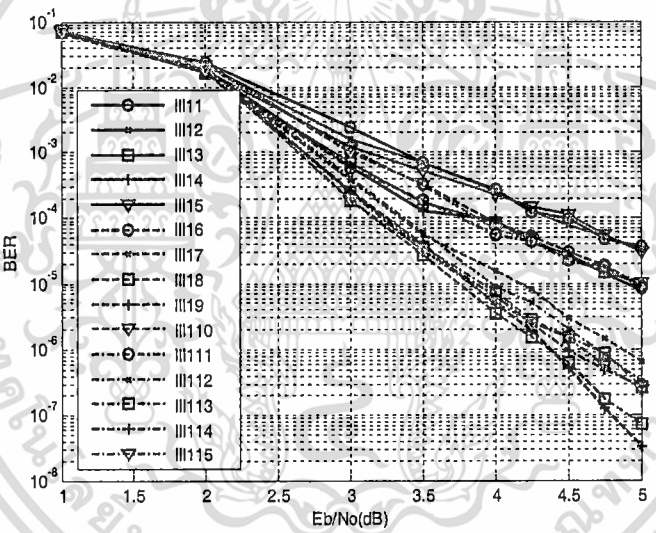
ตารางที่ 5.8 รายการเมจิกสแควร์สำหรับความยาวบล็อก1010

กลุ่ม	ชนิด	หมายเลข	ขนาด (zxz)
G.1	Jupiter	III 11	4x4
	Mars	III 12	5x5
	Sol	III 13	6x6
	Venus	III 14	7x7
	Mercury	III 15	8x8
	Lunar	III 16	9x9
G.2	Odd order	III 17	5x5
		III 18	7x7
		III 19	9x9
G.3	Doubly even	III 110	4x4
		III 111	8x8
G.4	Singly even	III 112	6x6 (LUX)
		III 113	6x6 (Strachey)
		III 114	10x10 (LUX)
		III 115	10x10 (Strachey)

การศึกษาขั้นต่อไปจะเป็นการทดสอบสมรรถนะของรหัสที่มีอัตรารหัสสูงขึ้น ขณะที่ความยาวบล็อกมีค่าคงที่ โดยศึกษาที่อัตรารหัสเท่ากับ 0.8 ภายใต้อัตรารหัสนี้ เมทริกซ์พาริตีเช็ก ที่เป็นไปได้ทั้งหมดถูกสร้างขึ้นบนพื้นฐานของ MSBA โดยใช้เมจิกสแควร์ที่เป็นไปได้ดังแสดงในตารางที่ 5.10 ภายใต้ตัวแปรการสร้างเดียวกัน ดังแสดงในตารางที่ 5.9 (คอลัมน์สุดท้าย ผลการศึกษาพบว่า สมรรถนะรหัสที่ดีที่สุด ได้จากการสร้างเมทริกซ์พาริตีเช็กด้วยเมจิกสแควร์ขนาด 7×7 (กลุ่มที่ 2)) จึงกล่าวได้ว่าเมื่ออัตรารหัสเปลี่ยนขนาดของเมจิกสแควร์ที่เหมาะสมที่สุดสำหรับแต่ละอัตรารหัสอาจเปลี่ยนด้วย

ตารางที่ 5.9 ตัวแปรสร้างรหัส [10] [25] และ [35] (ความยาวบล็อกสั้น)

ตัวแปร	รหัส [10]			รหัส [25]			รหัส [35]		
j	3	3	3	3	3	3	3	3	3
k	9	10	15	9	10	15	9	10	15
p	59	101	67	57	102	68	57	101	67
อัตรารหัส=1-(j/k)	0.7	0.7	0.8	0.7	0.7	0.8	0.7	0.7	0.8
บิตข้อมูล=p(k-j)	354	707	804	342	714	816	342	707	804
บิตพาริตี=jp	177	303	201	171	306	204	171	303	201
คำรหัส=kp	531	1010	1005	513	1020	1020	513	1010	1005



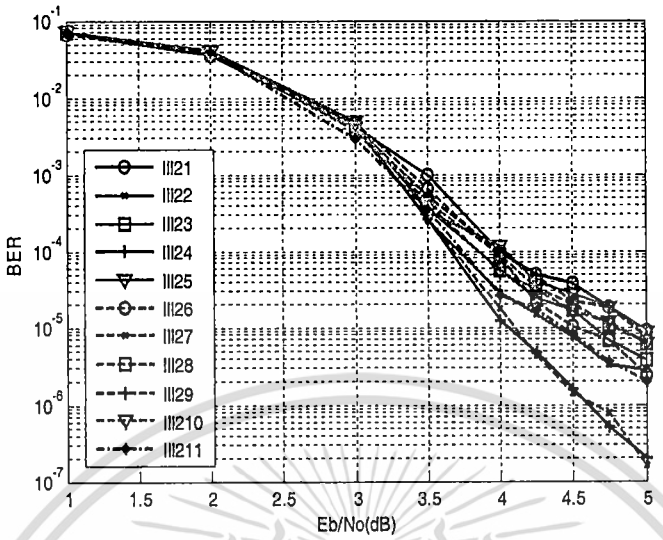
รูปที่ 5.9 ผลการทดสอบที่ความยาวบล็อก 1010 อัตรารหัส 0.7

จากรูปที่ 5.9 แสดงให้เห็นว่าที่ความยาวบล็อก 1010 และอัตรารหัส 0.7 เมทริกซ์พาริตีเช็กที่ออกแบบแล้วให้สมรรถนะดีที่สุด คือ เมทริกซ์พาริตีเช็กที่สร้างจากเมจิกสแควร์ในกลุ่มที่ 4 (หารด้วย 4 ไม่ลงตัว) ที่ขนาด 6×6 กับ 10×10 มีสมรรถนะที่ดีพอ ๆ กัน ส่งผลให้สามารถออกแบบได้หลากหลาย เพราะมีขนาดของเมจิกสแควร์ให้เลือกได้มากกว่า 1 ขนาด แสดงถึงความยืดหยุ่นในการออกแบบ

ตารางที่ 5.10 รายการทดสอบที่อัตรารหัสเท่ากับ 0.8

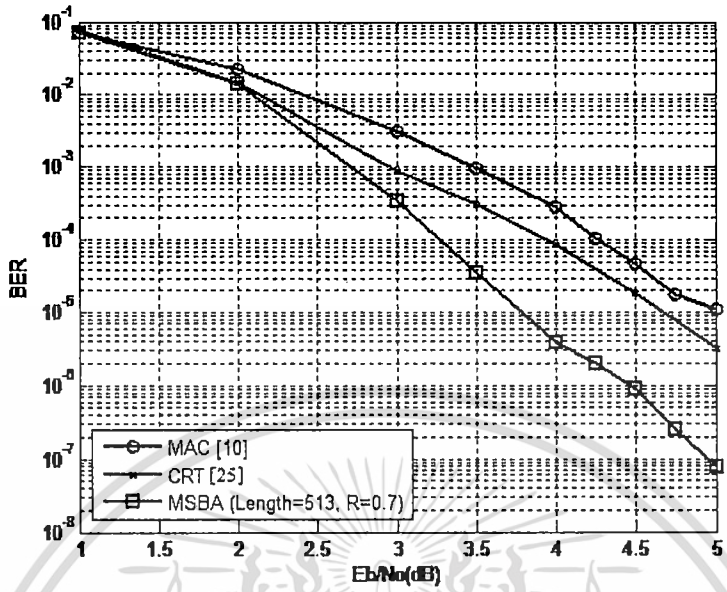
กลุ่ม	ชนิด	หมายเลข	ขนาด (zxz)
G.1	Jupiter	III 21	4x4
	Mars	III 22	5x5
	Sol	III 23	6x6
	Venus	III 24	7x7
	Mercury	III 25	8x8
G.2	Odd order	III 26	5x5
		III 27	7x7
G.3	Doubly even	III 28	4x4
		III 29	8x8
G.4	Singly even	III 210	6x6 (LUX)
		III 211	6x6 (Strachey)

เพื่อประเมินว่าอัลกอริทึมบนพื้นฐานของ MSBA ที่เสนอนี้ส่งผลให้สมรรถนะของรหัสดีขึ้นมากน้อยเพียงใด การศึกษาในขั้นตอนต่อไปจึงได้ทำการศึกษาเชิงเปรียบเทียบระหว่างสมรรถนะของรหัสที่สร้างบนพื้นฐาน MSBA กับสมรรถนะของรหัสที่นำเสนอในงานวิจัยของผู้อื่นที่ศึกษามาแล้วก่อนหน้านี้ คือ [10] และ [25] ภายใต้ความยาวบล็อกและอัตรารหัสเดียวกัน โดยมีตัวแปรควบคุมต่าง ๆ ดังแสดงในตารางที่ 5.9 อย่างไรก็ตาม เนื่องจากการใช้ตัวแปรที่ใช้ในการสร้างรหัส (ตัวแปรควบคุม) ของแต่ละพื้นฐาน ([10] [25] และ [35]) มีข้อจำกัดที่ไม่สามารถให้ตัวเลขเดียวกันได้ (เช่น ข้อจำกัดเกี่ยวกับจำนวนเฉพาะ) จึงต้องใช้ตัวเลขที่ใกล้เคียงกันมากที่สุดเท่านั้น ซึ่งผลการศึกษานี้ จะเป็นการเปรียบเทียบสมรรถนะของรหัสแอลดีพีซี ในคลาสเดียวกันนั่นเอง ผลการเปรียบเทียบพบว่ารหัสแอลดีพีซีบนพื้นฐานของ MSBA ที่เสนอนี้ มีสมรรถนะดีกว่าอย่างเห็นได้ชัด



รูปที่ 5.10 ผลการทดสอบที่ความยาวบล็อก1005 อัตรารหัส0.8

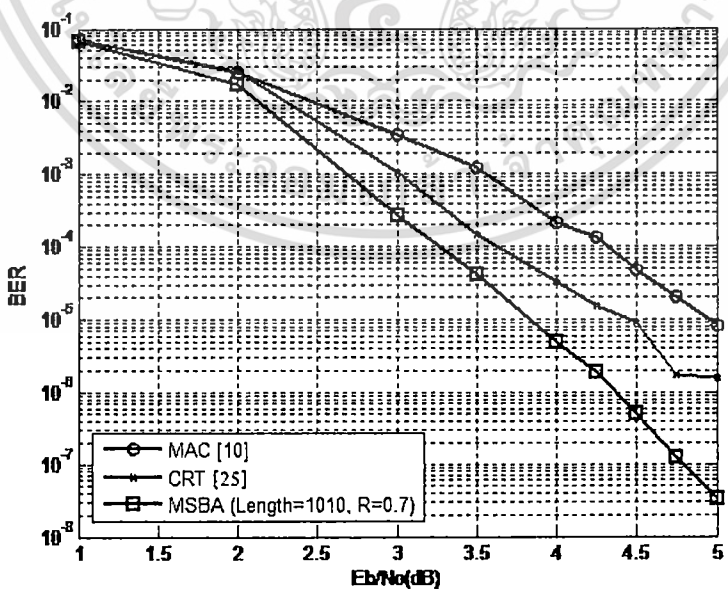
จากรูปที่ 5.10 แสดงให้เห็นว่าที่ความยาวบล็อก 1005 อัตรารหัส 0.8 เมทริกซ์พาริตีเช็กที่ออกแบบแล้วให้สมรรถนะดีที่สุด คือ เมทริกซ์พาริตีเช็กที่สร้างจากเมจิกสแควร์ที่มีขนาด 7×7 ทั้งสองเส้น (เส้นสีดำทึบ (III24) กับเส้นสีแดงประ (III27) มีสมรรถนะที่ดีพอ ๆ กัน ทำให้ทราบว่าการออกแบบเมทริกซ์พาริตีเช็กด้วยการนำเมจิกสแควร์มาใช้นั้น ขนาดที่ควรพิจารณาเป็นอันดับแรก คือ ขนาดที่อยู่ในกลุ่มเลขคี่ซึ่งเมื่อวิเคราะห์ถึงสาเหตุพบว่าขนาดของเมจิกสแควร์ในกลุ่มเลขคี่มีการจัดวางตัวเลขกระจายตัวได้ดี และเข้าใกล้การจัดวางแบบสุ่ม จากการคำนวณค่าทางสถิติมาพิจารณาร่วมด้วย ได้แก่ ค่าสัมประสิทธิ์ของการเปลี่ยนแปลง ปรากฏว่าค่าดังกล่าวของเมทริกซ์พาริตีเช็กที่สร้างจากเมจิกสแควร์ขนาด 7×7 ให้ค่าสูงกว่าเมทริกซ์พาริตีเช็กที่สร้างจากเมจิกสแควร์ขนาดอื่นที่ไม่ใช่เลขคี่



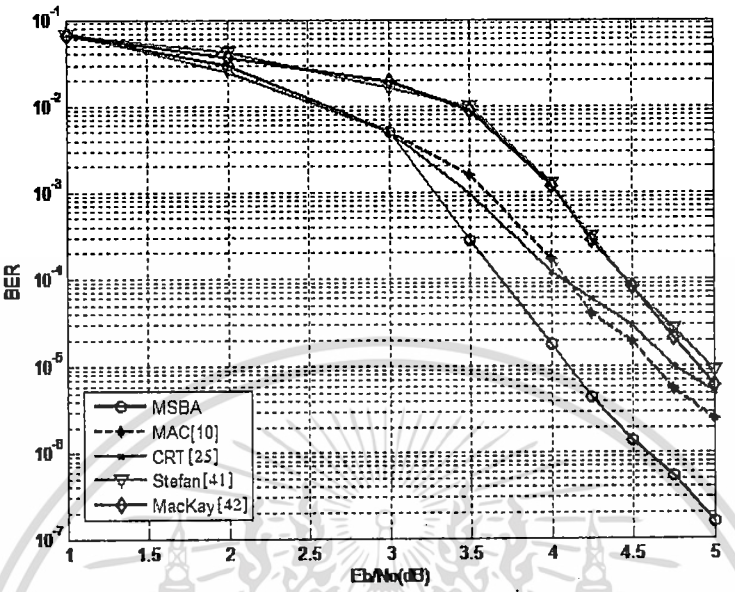
รูปที่ 5.11 ผลการทดสอบที่ความยาว ~500 อัตรารหัส 0.7

จากการศึกษาพบว่า วิธีการตรวจสอบว่าเมจิกสแควร์ที่จะนำมาใช้ในการสร้างเมทริกซ์พาริตีเช็ก มีความเหมาะสมที่จะนำมาใช้หรือไม่สามารถตรวจสอบได้จากเงื่อนไขแสดงดังสมการที่ (5.5)

$$\frac{p}{2} \le z^2 \le p \qquad (5.5)$$



รูปที่ 5.12 ผลการทดสอบที่ความยาว ~1000อัตรารหัส 0.7



รูปที่ 5.13 ผลการทดสอบที่ความยาว ~1000 อัตรารหัส 0.8

จากเงื่อนไขในสมการที่ (5.5) จะช่วยให้สามารถเลือกเมจิกสแควร์ได้ง่ายขึ้น เนื่องจากจะมีค่า z เพียงสองสามค่าเท่านั้น (จากการตรวจสอบเมจิกสแควร์ด้วยสมการที่ (5.5)) ในการเลือกค่า z ที่ดีที่สุดเพื่อสร้างเมจิกสแควร์ที่ต้องการนั้น จำเป็นต้องใช้ตัวแปรทางสถิติในการตรวจสอบ การวัดการแปรปรวนถูกนำมาใช้ในการคำนวณว่าตัวเลขทั้งหมดใน เมทริกซ์พาริตีเช็กมีการกระจายตัวสม่ำเสมอหรือไม่อย่างไร จากสมการที่ (3.7) ในบทที่ 3 จะได้ว่าค่าสัมประสิทธิ์ของการเปลี่ยนแปลงที่ต่ำกว่าจะแสดงว่าการกระจายตัวของตัวเลขทั้งหมดในเมทริกซ์พาริตีเช็กกระจายตัวได้ดีกว่า

ตารางที่ 5.11 รหัสที่สร้างด้วยวิธีที่สามเมจิกสแควร์ขนาด 6×6

H	1	2	3	4	5	6	7	8	9
1	1	1	1	1	1	1	1	1	1
2	0	1	1	6	26	19	24	3	32
3	0	0	1	28	27	25	14	13	12

ตัวอย่างเช่น กรณีเมจิกสแควร์ 2 ขนาด คือ 6×6 และ 7×7 หากจะนำเมจิกสแควร์ ทั้งสองมาใช้ในการสร้างเมทริกซ์พาริตีเช็กสำหรับรหัสแอลดีพีซีที่มีความยาวบล็อกเท่ากับ 513 (j=3, k=9 และ p=57) เมื่อ p/2=(57-1)/2=28 ดังนั้น 4×4=16 และ 5×5=25 ไม่เป็นไปตามสมการที่ (5.5) เพื่อเลือกเมจิกสแควร์ที่ดีที่สุด สามารถใช้การพิจารณาค่าสัมประสิทธิ์ของการเปลี่ยนแปลงที่คำนวณได้จากสมการที่

(3.7) ค่าสัมประสิทธิ์ของการเปลี่ยนแปลงของเมทริกซ์พาริตีเช็กที่สร้างจากเมจิกสแควร์ขนาด 6×6 (ดังแสดงในตารางที่ 5.11) และ 7×7 (ดังแสดงในตารางที่ 4.11) มีค่าเท่ากับ 66.5 % และ 71.2 % ตามลำดับจะเห็นว่าค่าสัมประสิทธิ์ของการเปลี่ยนแปลงกรณี z=6 ต่ำกว่ากรณี z=7 จึงเห็นได้ว่าการใช้เมจิกแควร์ขนาด 6×6 จะส่งผลให้สมรรถนะของรหัสดีกว่านั่นเอง ซึ่งผลลัพธ์แสดงดังรูปที่ 5.7 ซึ่งสนับสนุนหลักการนี้ได้เป็นอย่างดี

จากการศึกษาเชิงเปรียบเทียบโดยเปรียบเทียบสมรรถนะของรหัสแอลดีพีซีในคลาสเดียวกัน คือ รหัสแอลดีพีซีที่สร้างบนพื้นฐาน MSBA กับสมรรถนะของรหัสที่นำเสนอในงานวิจัยผู้อื่นที่ศึกษามาก่อนหน้านี้ คือ [10] [25] [41] และ [42] ภายใต้ความยาวบล็อกและอัตรารหัสเดียวกัน ผลการศึกษาพบว่ารหัสแอลดีพีซีบนพื้นฐานของ MSBA ที่เสนอนี้ มีสมรรถนะดีกว่าอย่างเห็นได้ชัด นอกจากนี้หากพิจารณาเปรียบเทียบระหว่างการสร้างเมทริกซ์พาริตีเช็กที่เสนอในงานวิจัยนี้ (ซึ่งสร้างเมจิกสแควร์ที่มีระเบียบวิธีที่แน่นอน) กับการสร้างเมทริกซ์พาริตีเช็กด้วยวิธีที่เสนอโดย [10] และ [25] แล้วจะเห็นว่า อัลกอริทึมการสร้างเมทริกซ์พาริตีเช็กบนพื้นฐานของเมจิสแควร์นี้มีความซับซ้อนน้อยกว่า และเมื่อทดสอบกับรหัสที่ออกแบบด้วยวิธีการสุ่ม [42] ผลแสดงให้เห็นว่างานวิจัย [35] มีสมรรถนะดีกว่าอย่างเห็นได้ชัด ดังนั้นการนำอัลกอริทึมนี้ไปประยุกต์ใช้งานจริงหรือประยุกต์ใช้กับช่องสัญญาณเสมือนจริงจึงเป็นสิ่งที่น่าเป็นไปได้ในอนาคตและน่าจะส่งผลที่ดีต่อระบบ

บทที่ 6
สรุปและข้อเสนอแนะจากการวิจัย

บทนี้จะกล่าวถึงผลการทดสอบแบบสรุป และข้อเสนอแนะจากการวิจัย รวมถึงปัญหา มุมมองของผู้วิจัย และแนวทางในการพัฒนางานต่อไปในอนาคต เพื่อให้ผู้มาอ่านหรือผู้มาศึกษางานวิจัยนี้ได้รับประโยชน์ แง่คิด แนวทางในการพัฒนาหรือต่อยอดงานหรือทำวิจัยใหม่ต่อไป

6.1 สรุปผลการทดสอบสมรรถนะของงานวิจัย

6.1.1 ผลการทดสอบ การออกแบบเมทริกซ์พาริตีเช็ก โดยพิจารณาจากการปรับรูปแบบใหม่ด้วยการลดเลขหนึ่งลงตามแนวเส้นทแยงมุมซ้ายและขวาของเมทริกซ์พาริตีเช็ก

จากการออกแบบเมทริกซ์พาริตีเช็ก เพื่อใช้สำหรับรหัสแอลดีพีซีบนหลักการทรานสโพรสเมทริกซ์ย่อย เพื่อลดเลขหนึ่งลงตามแนวเส้นทแยงมุมซ้ายและขวาของเมทริกซ์พาริตีเช็ก ผลจากการทดสอบปรากฏว่าเมื่อเปรียบเทียบกับงานวิจัยที่คล้ายกันในกลุ่มการออกแบบเมทริกซ์พาริตี้เช็กของงานที่ใช้หลักการเลื่อนวนแบบใหม่ (IMAC) [11] และกลุ่มความยาวบล็อกขนาดกลางและสั้น [16] มีสมรรถนะดีที่ยอมรับได้ที่ความยาวบล็อกสั้นและปานกลาง ส่วนความยาวบล็อกขนาดยาว สมรรถนะด้อยกว่างานวิจัยที่นำมาเปรียบเทียบ ข้อดีของงานวิจัยเราที่ได้ออกแบบ [33] คือสามารถออกแบบเมทริกซ์พาริตีเช็ก โดยลดจำนวนเลขหนึ่งลง ส่งผลให้การเข้า-ถอดรหัสเร็วขึ้น (เมื่อพิจารณาจากการคูณเมทริกซ์เฉพาะตำแหน่งที่เป็นเลขหนึ่ง) ปราศจากลูป 4มีสมรรถนะที่ดีเมื่อเปรียบเทียบกับรหัสแอลดีพีซีแบบคงที่ มีอัตรารหัสสูงและมีความซับซ้อนในการออกแบบที่ต่ำกว่างานวิจัยที่นำมาเปรียบเทียบ บนพื้นฐานแนวความคิดของงานวิจัยนี้จึงมีโอกาสที่จะพัฒนาการออกแบบเมทริกซ์พาริตีเช็ก เพื่อให้ได้สมรรถนะใกล้เคียงกับงานที่ใช้หลักการเลื่อนวนแบบใหม่

6.1.2 ผลการทดสอบการออกแบบเมทริกซ์พาริตีเช็กโดยพิจารณาการกำหนดค่าตัวแปรกับรูปแบบสมมาตรบนหลักการทรานสโพรสเมทริกซ์พาริตีเช็ก

การออกแบบรหัสที่มีอัตรารหัสสูงและมีขนาดความยาวบล็อกเหมาะสมกับระบบบันทึกข้อมูลแบบแม่เหล็กได้ถูกนำเสนอในงานวิจัยของเรา[34] โดยความยาวบล็อกที่ใช้ประมาณ 4,000 บิต อย่างไรก็ตาม ขนาดความยาวบล็อกที่ใช้ในงานจริงจะถูกหาบนพื้นฐานของอัตราข้อมูลเฉพาะ มีหน่วยเป็นเมกะบิตต่อวินาที(Mbit/Sec)ซึ่งในทางปฏิบัติค่านี้จะถูกทำให้เหมาะสมที่สุดภายใต้ตัวแปรหลายตัวแปร เพื่อทำให้เงื่อนไขขอบเขตของการออกแบบง่าย ในงานวิจัยนี้ พยายามที่จะออกแบบรหัสที่รองรับความยาวบล็อก

ทุกขนาดด้วยการกำหนดค่าตัวแปรในการสร้างเมทริกซ์พาริตีเช็กใหม่โดยสามารถใช้ตัวแปรที่ไม่ใช่จำนวนเฉพาะได้เมทริกซ์พาริตีเช็กที่เสนอนี้ถูกออกแบบบนพื้นฐานของคุณสมบัติเชิงสมมาตรจัดวางให้เหมาะสมก่อนการเลื่อนวนแม้ว่ารหัสที่ถูกออกแบบในงานวิจัยนี้ จะให้สมรรถนะที่ใกล้เคียง หรือดีกว่าเล็กน้อยเมื่อเปรียบเทียบกับผลการศึกษารหัสแอลดีพีซีแบบอาร์เรย์แนวใหม่ [10] และงานวิจัยที่ออกแบบขนาดเหมือนของรหัสแอลดีพีซีแบบอาร์เรย์ [24] ผลการจำลองในงานวิจัยนี้แสดงให้เห็นว่า ขนาดของเมทริกซ์ย่อยมีผลต่อสมรรถนะของรหัส เมทริกซ์ย่อยที่มีขนาดใหญ่ขึ้นจะทำให้ระยะห่างต่ำสุดมีขนาดสูงขึ้น และได้สมรรถนะดีขึ้นมากกว่ารหัสที่มีเมทริกซ์ย่อยขนาดเล็ก อย่างไรก็ตาม ขนาดของเมทริกซ์ย่อยควรมีค่าต่ำกว่า 155 เมทริกซ์ที่เสนอในงานวิจัยนี้ยังคงมีคุณสมบัติที่เหมาะสมสำหรับระบบบันทึกข้อมูลแบบแม่เหล็ก เช่น มีความซับซ้อนในการถอดรหัสและมีระดับความผิดพลาดต่ำมีความสามารถในการตรวจจับและแก้ไขความผิดพลาดแบบหลายบิตติดกัน (Burst Errors) ได้ดีเป็นต้น

จากการศึกษาวิจัยสรุปได้ว่า ข้อดีของเมทริกซ์สมมาตร มีดังนี้ 1) ออกแบบง่ายและเร็ว 2) ได้สมรรถนะดีใกล้เคียงกับการสร้างแบบสุ่ม และ 3) ได้ค่าระยะห่างต่ำสุดที่สูงกว่า และมีความสามารถในการตรวจจับและแก้ไขข้อผิดพลาดที่ติดกันหลายบิตได้ สำหรับงานวิจัยขั้นต่อไปควรทำการประเมินสมรรถนะของรหัสที่ 4,000 ไบต์ ที่ใช้ในเทคโนโลยีการบันทึกข้อมูลรูปแบบใหม่บนช่องสัญญาณแบบฮาร์ดดิสก์ไดร์ฟ

6.1.3 ผลการทดสอบการออกแบบเมทริกซ์พาริตีเช็ก โดยพิจารณาการออกแบบเมทริกซ์พาริตีเช็กโดยประยุกต์ทฤษฎีคณิตศาสตร์พื้นฐานเรื่องเมจิกสแควร์ เพื่อให้ได้อัลกอริทึมในการสร้างเมทริกซ์บนพื้นฐานของ MSBA

การศึกษาวิจัยครั้งนี้ [35] เสนออัลกอริทึมใหม่การสร้างเมทริกซ์พาริตีเช็กสำหรับรหัสแอลดีพีซีแบบไม่คงที่ที่มีความยาวบล็อกสั้นในช่องสัญญาณรบกวนแบบเกาส์สีขาว เพื่อให้ได้เมทริกซ์พาริตีเช็กที่ดีที่สุดจำเป็นต้องใช้จำนวนการเลื่อนเมทริกซ์สับเปลี่ยนย่อยที่แตกต่างกันในจำนวนครั้งที่เหมาะสมงานวิจัยนี้อธิบายรายละเอียดเกี่ยวกับเมจิกสแควร์ทั้งหมดที่มีอยู่ในปัจจุบันโดยแบ่งได้เป็นสี่กลุ่ม จากนั้นนำเมจิกสแควร์ทั้งหมด มาใช้ในการออกแบบอัลกอริทึมใหม่ในการสร้างเมทริกซ์พาริตีเช็กซึ่งการศึกษาได้เสนอวิธีการนำตัวเลขจากเมจิกสแควร์มาใช้ด้วยวิธีการจัดวางที่แตกต่างกัน 3 วิธี ผลการศึกษาแสดงให้เห็นว่าอัลกอริทึม MSBA ที่เสนอสามารถนำมาประยุกต์ใช้สร้างเมทริกซ์พาริตีเช็กได้เป็นอย่างดี ด้วยอัลกอริทึม MSBA ที่เสนอนี้ จะทำให้ได้เมทริกซ์พาริตีเช็กที่ปราศจากลูป 4 การทดสอบสมรรถนะของรหัสภายใต้ตัวแปรการสร้างเดียวกันพบว่า การเลือกและจัดวางตัวเลขวิธีที่สามให้สมรรถนะของรหัสที่ดีที่สุด เนื่องจากตัวเลขที่นำมาใช้มีความเป็นอิสระต่อกันและมีการกระจายตัวของตัวเลขในลักษณะการกระจายแบบปกติ อย่างไรก็ตาม การเลือกเมจิกสแควร์ที่เหมาะสม (ทั้งขนาดและกลุ่มหรือรูปแบบ) จะต้องสัมพันธ์กับความ

ยาวบล็อกและอัตรารหัส จากการศึกษาเชิงเปรียบเทียบโดยเปรียบเทียบสมรรถนะของรหัสแอลดีพีซีในคลาสเดียวกันคือรหัสแอลดีพีซีที่สร้างบนพื้นฐาน MSBA กับสมรรถนะของรหัสที่นำเสนอในงานวิจัยผู้อื่นที่ศึกษามาก่อนหน้านี้ คือ รหัสแอลดีพีซีแบบอาร์เรย์ [10] รหัสแอลดีพีซีที่ออกแบบประยุกต์บนพื้นฐานทฤษฎีเศษเหลือของจีนที่ใช้ค่าตัวแปรที่ไม่ใช่จำนวนเฉพาะ [25] รหัสที่ออกแบบที่ใช้หลักการสี่เหลี่ยมละติน [41] และรหัสแอลดีพีซีที่ออกแบบด้วยวิธีการสุ่ม [42] ภายใต้ความยาวบล็อกและอัตรารหัสเดียวกัน (เน้นที่ความยาวบล็อกไม่เกิน 1,000 บิต และอัตรารหัสไม่ต่ำกว่า 0.8) ผลการศึกษาพบว่ารหัสแอลดีพีซีบนพื้นฐานของ MSBA ที่เสนอนี้ มีสมรรถนะดีกว่าอย่างเห็นได้ชัด นอกจากนี้หากพิจารณาเปรียบเทียบระหว่างการสร้างเมทริกซ์พาริตีเช็กที่เสนอในงานวิจัยนี้ (ซึ่งสร้างเมจิกสแควร์ที่มีระเบียบวิธีที่แน่นอน) กับการสร้างเมทริกซ์พาริตีเช็กด้วยวิธีที่เสนอรหัสแอลดีพีซีแบบอาร์เรย์โดย [10] และการประยุกต์ทฤษฎีเศษเหลือของจีนที่ใช้ตัวแปรที่ไม่ใช่จำนวนเฉพาะ [25] จะเห็นว่า อัลกอริทึมการสร้างเมทริกซ์พาริตีเช็กบนพื้นฐานของเมจิสแควร์นี้ มีความซับซ้อนน้อยกว่า และเมื่อทดสอบกับรหัสที่ออกแบบด้วยวิธีการสุ่ม [42] ผลแสดงให้เห็นว่างานวิจัย [35] มีสมรรถนะดีกว่าอย่างเห็นได้ชัด ดังนั้นการนำอัลกอริทึมนี้ ไปประยุกต์ใช้งานจริงจึงเป็นสิ่งที่น่าจะมีอนาคตที่ดี

เมทริกซ์พาริตีเช็กแบบมีโครงสร้างที่ได้จากงานวิจัยนี้สามารถนำไปใช้ได้อย่างเหมาะสมสำหรับรหัสที่มีขนาดความยาวบล็อกสั้นมีอัตรารหัสสูงกว่า 0.7 ซึ่งจะได้ค่าอัตราความผิดพลาดประมาณ 10^{-7} ที่ค่า SNR เท่ากับ 5 dB โดยจะมีจำนวนรอบในการถอดรหัสแบบทำซ้ำต่ำ เพียง 30รอบเท่านั้น รหัสนี้สามารถถูกนำไปประยุกต์ใช้ในระบบโทรศัพท์เคลื่อนที่ หรือระบบสื่อสารไร้สารที่มีข้อจำกัดเกี่ยวกับหน่วยความจำและช่องสัญญาณได้ สำหรับงานวิจัยที่จะทำต่อไปนั้น จะนำการออกแบบรหัสนี้ไปประยุกต์ใช้กับช่องสัญญาณชนิดอื่นที่ต้องการใช้รหัสที่มีขนาดความยาวบล็อกสูงขึ้นหรือนำไปประยุกต์ใช้งานร่วมกับการเข้า-ถอดรหัสแบบอื่น การใช้หลักการทางสถิติมาประยุกต์ใช้เพื่อประกอบในการพิจารณาตัดสินใจ เพื่อดูสภาพการกระจายตัวของเลขหนึ่งในเมทริกซ์พาริตีเช็กเป็นแบบปกติหรือไม่ จากผลการทดสอบทำให้เห็นถึงความสัมพันธ์กันว่า ถ้าการกระจายตัวของเลขหนึ่งในเมทริกซ์พาริตีเช็กเข้าใกล้โค้งแบบปกติมากกว่า จะส่งผลให้สมรรถนะดีกว่าเมทริกซ์พาริตีเช็กที่มีค่าการกระจายตัวของเลขหนึ่งที่เข้าใกล้โค้งแบบปกติน้อยกว่า

6.2 ข้อเสนอแนะจากการวิจัย

6.2.1 การออกแบบเมทริกซ์พาริตีเช็กมีผลต่อสมรรถนะของรหัสแอลดีพีซีอย่างมาก ซึ่งพิสูจน์มาแล้วจากการวิจัย ดังนั้นถ้าคิดวิธีการออกแบบเมทริกซ์พาริตีเช็กที่มีกระบวนการได้มาซึ่งเมทริกซ์พาริตีเช็กที่ง่ายและสามารถลดจำนวนเลขหนึ่งลดได้จะส่งผลต่อการเข้า-ถอดรหัสให้เร็วได้ นอกจากนี้ยังเห็นว่า

แนวทางการออกแบบเมทริกซ์พาริตีเช็ก ยังสามารถพัฒนาได้อีกเพื่อให้ได้สมรรถนะที่ดี ซึ่งแนวทางการใช้หลักการทางคณิตศาสตร์พื้นฐานหรือเกมคณิตศาสตร์ที่ง่ายมาประยุกต์ เช่น หลักการของเกมซูโดกุ เป็นต้น ยังไปได้อีก ซึ่งยังมีทฤษฎีทางคณิตศาสตร์อีกมากยังรอการประยุกต์ พัฒนาเพื่อทดสอบดูสมรรถนะหรือต่อยอดความคิดต่อไป

6.2.2 การกำหนดโครงสร้างเมทริกซ์พาริตีเช็กจะช่วยลดความซับซ้อนในการสร้างได้หรืออาจทำได้โดยหลีกเลี่ยงการสร้างเมทริกซ์พาริตีเช็กทั้งหมด โดยใช้เป็นการกำหนดหนึ่งส่วนและคำนวณอีกหนึ่งส่วน

6.2.3 การออกแบบเมทริกซ์พาริตีเช็กแบบคงที่นั้นถ้ามีจำนวนลูปขนาดใหญ่ ยิ่งใหญ่มากยิ่งดี เช่น ลูปขนาด 6, 8 หรือ 12 เป็นต้น ลูปขนาดใหญ่จะส่งผลต่อสมรรถนะที่ดี หลักการลูปคล้ายหลักการลูป 4 แต่สำหรับลูป 4 นั้นส่งผลต่อสมรรถนะที่ไม่ดีต่อรหัสแอลดีพีซี

6.2.4 ลูป 4 คือชุดความสัมพันธ์ของตำแหน่งเลขหนึ่งในเมทริกซ์พาริตีเช็กค่า 4 ค่าวนกันเป็นลูป

6.2.5 การออกแบบเมทริกซ์พาริตีเช็ก แบบไม่คงที่จะได้สมรรถนะไม่ดี ถ้ามีลูป 4 แต่ถ้าเป็นการออกแบบเมทริกซ์พาริตีเช็กแบบคงที่จะดี ถ้าพัฒนาจากลูป 4 เป็นเกริธขนาดใหญ่จะส่งผลดีต่อสมรรถนะ

6.2.6 รหัสแอลดีพีซีแบบอาร์เรย์นั่น ก็คือรหัสแอลดีพีซีแบบเลื่อนวน ที่เรียกชื่อต่างกันแต่หลักการคล้ายกัน

6.2.7 การใช้คำศัพท์ต่าง ๆ ยังมีความหลากหลาย ไม่เป็นระบบเดียวกัน ต่างคนต่างบัญญัติขึ้นมา ซึ่งในอนาคตควรมีการทำให้เป็นระบบเดียวกัน โดยบัญญัติร่วมกันในกลุ่มนักวิชาการด้านระบบการสื่อสารหรือสถาบันระดับโลกคือ IEEE และ ACM เป็นต้น

เอกสารอ้างอิง

[1] C. Berrou, A.Glanvieux and P. Thitimajshima. "Near Shannon limit error-correcting coding and decoding : Turbo Codes" **in Proc. IEEE Intl.Conf.**, May. 1993., pp. 1064-1070

[2] D.J.C. Mackay and R. Neal. "Near Shannon limit performance of low density parity check codes" **Electronics Letters**, vol.33, Mar. 1997., pp. 457-458

[3] R. Gallager. "Low-density parity-check codes" **IRE Trans. Information Theory**, Jan. 1962., pp. 21-28

[4] C. Sae-Young, G. D. Forney, Jr., T. J. Richardson, and R. Urbanke. "On the design of low-density parity-check codes within 0.0045 dB of the Shannon limit" IEEE **Communication Letter**, vol. 5.2001., pp. 58-60

[5] R. M. Tanner. "A recursive approach to low complexity codes" IEEE **Trans. Information Theory**, Sep. 1981., pp. 533-547

[6] J.L. Fan. "Array Codes as low-density parity-check codes" **Proc. 2nd Int. Symposium Turbo Code**, Beit, France, Sep. 2000., pp. 543-546

[7] T. J. Richardson, M. A. Shokrollahi, R. L. Urbanke. "Design of capacity-Approaching irregular low-density parity-check codes" **Information Theory IEEE Trans.**, Volume 47, Feb. 2001., pp. 619-637

[8] M. Chai and A. Ventura. "Design and Performance evaluation of some high-rate irregular low-density parity-check codes" **Proc. 2001 IEEE GlobeCom. Conf.**, Nov. 2001., pp.990-994

[9] M.Yang and W.E. Ryan. "Lowering the error rate floors of moderate-length high rate LDPC codes" **Proc. 2003 Int. Symposium on Inf. Theory,** Jun-Jul. 2003.

[10] E. Eleftheriou and S. Olcer. "Low-density parity check codes for digital subscriber lines" **Proc. 2002 Int. Conf. on Comm.,** Apr.-May. 2002., pp. 1752-1757

[11] W. Singhaudom, S. Noppankeepong, P. Suphithi. "Design of High-Rate Modified ArrayCodes for Magnetic Recording System" **The 2007 ECTI International Conference,** May. 2007.

[12] O. Othman Khalifa, S. Khan, M. Zaid, and M. Nawawi. "Performance Evaluation of Low Density Parity Check Codes" **International Journal of Computer Science and Engineering 2**, 2008.

[13] P.Trifonov. "Design of Structured Irregular LDPC Codes" **Proceedings of SIBIRCON,** 2008.

[14] D.A. Thompson and J.S. Best. "The future Magnetic Data Storage Technology" **IBM J. Res. Develop.,** vol. 44, no.3, May. 2000., pp. 311-322

[15] E. Mo and M. A. Armand. "Design and Performance of LDPC Codes Extended with Parity-Check Symbols from a Larger Alphabet" **ICICS 2007**, National University of Singapore, 2007.

[16] M. R. Islam and J. Kim. "On the use of QC-LDPC code for data transfer using short and medium block length" **Kyung Hee University,** ISBN 978-89-5519-139-4, 2009., pp. 1804-1809

[17] M. P. C. Fossorier. "Quasi-Cyclic Low-Density Parity-Check Codes From Circulant Permutation Matrices" **IEEE Trans. Inf. Theory**, vol. 50, no. 8, Aug. 2004.

[18] S. Chae , Y. Park. "Low Complexity Encoding of Improved Regular LDPC Codes" **IEEE**, 2004.

[19] Y. Sun , M. Karkooti and J. R. Cavallaro. "High Throughput Parallel Scalable LDPC Encoder/Decoder Architecture for OFDM Systems"

[20] M. Blaum, P. Farrell, and H. van Tilborg. "Array codes, in Handbook of Coding Theory" V. S. Pless and W. C. Huffiman Eds.,**Elsevier**, 1998.

[21] S. Myung and K. Yang. "A combing method of quasi-cyclic LDPC codes by the Chinese Remainder Theorem" **IEEE Communications Letter,** vol. 9, 2005., pp. 823-825

[22] M. G. Luby, M. Mitzenmacher, M.A. Shokrollahi, and D.A. Spielman. "Efficient erasure correcting codes" **IEEE Transaction Information Theory**, vol. 47, 2001., pp. 569-584

[23] P. Kovintavewat. "Oversampled timing recovery for magnetic recording channels" **in Proc. of ECTI-CON 2006**, UbonRatchathani, Thailand, vol. I/II, May. 2006., pp. 235-238

[24] D. Abematsu, T. Ohtsuki, S. PW Jarot, and T. Kashima. "Size Compatible (SC)-Array LDPC Codes" **IEEE Vehicular Technology Conference**, 2007., pp.1147-1151

[25] C. Chusin, C. Prasartkaew, S. Timakul and S. Choomchuay. "A Design of Nonprime Block Irregular LDPC Codes via CRT" **ISCIT International Conference**, Japan, 2010.

[26] W. H. Benson and O. Jacoby. **New Recreations with Magic Squares.** New York: Dover, 1976.

[27] W. W. R. Ball. **Mathematical recreations and essays.** London : Macmillan & Co Ltd., 1959.

[28] Q. H.Spencer. "Short-Block LDPC Codes for a Low-Frequency Power-Line Communications System" **IEEE Communications Society**, 2005., pp. 95-99

[29] R. Aditya and W. Richard. "Construction of Short Block Length Irregular Low-Density Parity-Check Codes" **IEEE Communications Society**, 2004., pp. 410-414

[30] J. Lu and J. M. F. Moura. "Linear Time Encoding of LDPC Codes" **IEEE Transactions on Information Theory**, vol. 56, no. 1, Jan. 2010., pp. 233-249

[31] Y.Fengfan and Y. Ming ."High-Performance Simple-Encoding Generator-Based Systematic Irregular LDPC Codesand Resulted Product Codes" **Journal of Electronic (China)**, vol.24, no.5, Sep. 2007., pp. 613-621

[32] S. J. Johnson and S. R. Weller. "Codes for Iterative Decoding From Partial Geometries" **IEEE Transactions on Communications**, vol. 52, no. 2, Feb. 2004., pp. 236-243

[33] C.Prasartkaew and S.Choomchuay. "A Design of Parity Check Matrix for Irregular LDPC Codes" **ISCIT International Conference,** Korea, Sep. 2009., pp. 239-242

[34] C. Chusin, C. Prasartkaew and S. Choomchuay. "Non-Prime Parameters of LDPC Codes with Symmetrical Sub-Matrix" **Proceeding of ITC-CSCC,** 2010.

[35] C.Prasartkaew and S.Choomchuay. "A Design of Parity Check Matrix for Irregular LDPC Codes via MSBA" **IJECET International Journal Electronic and Communication Engineering Technology,** 2013.

[36] C. Prasartkaew and S. Choomchuay. "A Parity Check Matrix Design for Irregular LDPC Codes with 2K Block Length" **ISPACS Int. Symp. on Intelligent Signal Processing and Comm. Systems,** Dec. 2009., pp. 550-553

[37] C. Chusin, C. Prasartkaew and S. Choomchuay. "A Design of Non-prime LDPC Based on Interleave Modified Array Codes" **Proceeding of DST-CON,** Oct. 2010., pp. 222-225

[38] C.Prasartkaew and S.Choomchuay. "Parity Check Matrix Construction via Magic Square Based Algorithm" **ISCIT International Conference,** Australia, Oct. 2012., pp. 54-59

[39] J. Fan, Y. Xiao and K. Kim. "Design LDPC Codes without Cycles of Length 4and 6" **Hindawi Publishing Corporation Research Letters in Communications,** 2008.

[40] R. Lucas, M. P. C. Fossorier, Y. Kou and S. Lin. "Iterative decoding of one-step majority logic decodable codes based on belief propagation" **IEEE Transactions on Communications,** vol. 48, no. 6, 2000., pp. 931-937

[41] L. Stefan and M.Olgica. "LDPC Codes Based on Latin Squares: Cycle Structure, Stopping Set, and Trapping Set Analysis" **IEEE Transactions on Communications,** vol. 55, no. 2, 2007., pp. 303-312

[42] D. MacKay **"Gallager code esources"** [Online]. Available: http://www.inference.phy.cam.ac.uk/mackay/CodesFiles.html

ภาคผนวก

ภาคผนวก ก.

ผลงานวิจัยที่ได้รับการตีพิมพ์เผยแพร่

1. C. Prasartkaew and S. Choomchuay, "**A Design of Parity Check Matrix for Irregular LDPC Codes,**" The 9th International Symposium on Communication and Information Technologies, Incheon, Korea, pp. 239-242, September 28-30,2009.
2. C. Prasartkaew and S. Choomchuay, "**A Parity Check Matrix Design for Irregular LDPC Codes with 2K Block Length,**"2009 International Symposium on Intelligent Signal Processing and Communication Systems,Kanazawa, Japan, pp. 550-553,December 7-9,2009.
3. C. Chusin, C. Prasartkaew and S. Choomchuay, "**Non-Prime Parameters of LDPC Codes with Symmetrical Sub-Matrix,**"25th International Technical Conference on Circuit/Systems, Computers and Communications, Pattaya, Thailand, pp. 934-938, July4-7, 2010.
4. C. Chusin, C. Prasartkaew and S. Choomchuay, "**A Design of Non-prime LDPC Based on Interleave Modified Array Codes,**" The 3rd International Data Storage Technology Conference (DST-CON 2010), Bangkok, Thailand, pp.222-225, October26-29, 2010.
5. C. Prasartkaew and S. Choomchuay, "**A MSBA for IEEE 802.11n Block Compatible LDPC Codes,**"10th International Symposium on Communications and Information Technologies, pp. 610-612, Tokyo, Japan, October22-23,2011.
6. C. Prasartkaew and S. Choomchuay, "**A Design of IEEE 802.11n Block Compatible LDPC Codes Using MSB Algorithm,**"2012 International Conference on Embedded Systems and Intelligent Technology, Nara, Japan, pp. 91-94, January27-29, 2012.

7. C. Prasartkaew and S. Choomchuay, **"Parity Check Matrix Construction via Magic Square Based Algorithm,"**The 12th International Symposium on Communication and Information Technologies, Gold Coast, Australia, pp. 54-59, October2-5, 2012.

8. C. Prasartkaew and S. Choomchuay, **"A Design of Parity Check Matrix for Short Irregular LDPC Codes via Magic Square Based Algorithm,"**International Journal of Electronics and Communication Engineering And Technology, ISSN Print: 0976-6464, ISSN Online: 0976-6472, 2013.

DST-CON 2010

A Design of Non-prime LDPC Based on Interleave Modified Array Codes

Chalit Chusin[1], Chutima Prasartkaew[2], and Somsak Choomchuay[3]
[1,2] College of Data Storage Technology and Applications, King Mongkut's Institute of Technology Ladkrabang BKK, Thailand
[1]Tel:/Fax: + 66-3-881-5966, E-mail: ithonene@hotmail.com
[2] Tel:/Fax: + 66-2-326-4731, E-mail: prasartkaew@yahoo.com
[3]Faculty of Engineering, King Mongkut's Institute of Technology Ladkrabang, BKK 10520, Thailand
Tel: +66-2-326-4222 Ext.114, Fax: +66-2-739-2398, E-mail: kchsomsa@kmitl.ac.th

Abstract— **Interleave Modified Array Codes (IMAC) is a type of Low Density Parity Check Codes (LDPC), which is used to achieve high rate codes and good error rate performance in additive white Gaussian noise (AWGN) channels. IMAC has many advantages in magnetic recording system such as low error floor, capability of detecting and correcting burst error etc. However, IMAC doesn't support arbitrary code lengths, as the code lengths of IMAC with good error rate performance are limited by a multiple of a prime number. This paper presents a design of parity check matrix to support arbitrary code length. The design is based on the construction rows that don't have the same sub-matrix in same row. We conduct computer simulations to evaluate the bit error rate (BER) performance of Non-prime IMAC in AWGN channels. We show that even if the sub-matrix size is a non-prime number, IMAC achieve the same error rate performance as the prime sub-matrix size in AWGN channels.**

Keywords- LDPC; Non-prime blocklenth; Array Codes

I. INTRODUCTION

Low Density Parity Check (LDPC) codes were first introduced by Gallager [1] in 1962.They have recently attracted wide attention in the coding community because their performance is near Shannon limit when use iterative message-passage decoding [2] for decode.

In magnetic recording systems, higher codes rate must be considered since the internal clock rates of read channels gets higher [3]. So, the high rate codes are obviously desirable. "Modified Array Codes" was proposed to use in the communication systems for medium code rate [4]. The codes have the performance similar to Regular / Irregular LDPC codes but they have more attractive properties such as capability of detecting and correcting burst error, simple encoding and low error floor. Interleave Modified Array Codes was designed for high rate applications which the parity matrix is superior to Modified Array Codes when the block length is particularly long [5].

Interleave Modified Array Codes (IMAC) is specified by three parameters; j,k,L where $(j,k) \le L$. Parameters j, k and L represent row weights, column weights and the sub-matrix size of the LDPC code. In general, to construct IMAC with good error rate performance, the sub-matrix size *(L)* has to be prime in order because it avoid the possibility of 4-cycle. The codes length of an IMAC can construct with the multiple of sub-matrix size. Obviously, it is difficult to construct an IMAC with good error rate performance for support arbitrary lengths. Consequently, we have to add a dummy bit to the code when IMAC isn't implemented on some code length. The transmission rate will be apparently degraded since the dummy bits contain no information. In this paper, we are proposing the new design of IMAC that can supports arbitrary code lengths. This feature can be obtained when the sub-matrix size is non-prime. The rest of this paper is organized as follows. The encoder and decoder are given in brief in the next section. Sum product algorithm (SPA) is highlighted. In section III, we detail the design of a **H** matrix that can support arbitrary block lengths. This section begins with some necessary works in **H** matrix design. In the second part we elaborate the non-prime block length design. Then in section IV, before the conclusion, we do evaluate our design compared to some published literatures.

II. LDPC CODES

The LDPC codes are generally specified by sparse parity check matrix **H** that can be represented by a Tanner graph. The graph expresses the relationship between codeword bits and parity checks bit. A parity-check matrix **H** has m rows and n columns. The codeword consists of n bits which satisfy m checks. The number of message bits will be $k = (n - m)$ and the rate of codes is $R = k/n$. The number 1's in the parity check matrix in rows and columns represents an edge between the *i-th* bit node c_i and the *j-th* check node f_j.

An example of the parity check matrix **H** of a LDPC code of dimension m, $n = (3, 7)$ is shown as

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 & 1 & 1 \end{bmatrix}_{(m \times n)} \quad (1)$$

The Tanner graph representing the LDPC code of the above matrix (1) can be constructed as shown in Fig. 1.

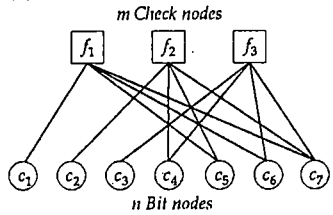


Figure 1. A Tanner graph for the matrix given in (1)

Number of 1s in the matrix reflects whether the code is regular or irregular one. In addition, the performance of the designed code is very much depended on the structure of the matrix. In the subsequent section, encoding and decoding of LDPC codes are given.

A. Encoder

Similar to other linear block codes, LDPC codes can be encoded with the relation;

$$C_{(1\times n)}H^{T}_{(n\times m)} = 0 \tag{2}$$

where C is a codeword matrix, and **H** is a parity check matrix. In a systematic form, C can be written as:

$$C_{(1\times n)} = \left[m_{(1\times m)} \,|\, p_{(1\times n-m)} \right] \tag{3}$$

Here $m_{(1\times m)}$ denotes the message portion, and $p_{(1\times n-m)}$ denotes the parity portion respectively.

B. Decoders

There are several methods used to decoding the LDPC codes. There are: Believe Propagation (BP), another name is Message Passing (MP) or Sum-Product (SP) [1] and Log-domain Sum-Product algorithm (log-SPA) [6]. In the Log-domain Sum-Product algorithm (SPA), the message simply passes between check nodes and bit nodes. In each pass the log likelihood ratio (LLR) [9] is recorded for is probability of its likely symbol.

The basic operation for the decoding algorithm is shown in Tanner graph in Fig 2. In the graph, the check nodes and bit nodes have interchange soft information. The variable q_{ij} is the message sent from the i^{th} bit node to j^{th} check node along a connecting edge, and r_{ji} is the message sent from j^{th} check node to the i^{th} bit node along a connection edge. The message q_{ij} is computed based on the values sent from check nodes connecting to the i^{th} bit node excluding the j^{th} check node.

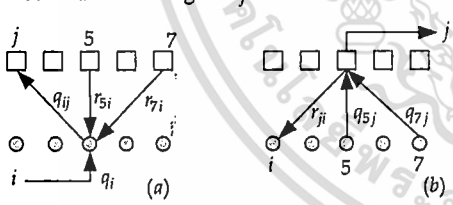


Figure 2. (a) A message is passed from *i-th* bit node to *j-th* check node and (b) A message is passed from the *j-th* check node to the *i-th* bit node.

III. PARITY CHECK MARIX DESIGN

A parity check matrix reflects directly the performance of the encoder and the corresponding decoder. Therefore, a design of a parity check matrix is one of the challenges in designing the LDPC codes. In this paper, the **H** matrix is designed by construction rows that don't have the same sub-matrix in same row. This mean that it can supports arbitrary code lengths because we can design the sub-matrix size with Non-prime. The design of the parity check matrix in this paper is based on Interleave Modified Array LDPC.

A. Some Previous Works

Eleftheriou and Olcer [4] have proposed the modified array codes (MAC) by applying cyclic shift to Fan's [7] array.

$$H = \begin{bmatrix} I & I & \cdots & I & I & \cdots & \cdots & I \\ 0 & I & \alpha & \cdots & \alpha^{(j-2)} & \alpha^{(j-1)} & \cdots & \alpha^{(k-2)} \\ 0 & 0 & I & \cdots & \alpha^{2(j-3)} & \alpha^{2(j-2)} & \cdots & \alpha^{2(k-3)} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \cdots & I & \alpha^{(j-1)} & \cdots & \alpha^{(j-1)(k-j)} \end{bmatrix} \tag{4}$$

This structure has the code rate of $R = 1-(j/k)$. MAC offers superior performance to Fan's array as it can reduce number of "1" in the lower triangle. It is simple for encoding, low error floor and capability of detecting and correcting burst error.

where I is an identity matrix of $(p\times p)$.

α is the position permutation matrix, also of $(p\times p)$.

Singhaudom[5] has proposed the interleaved modified array LDPC or IMAC by Eq. (5) by introducing the quasi cyclic matrix into the cyclic shift. This interleaved LDPC with the parity matrix given by Eq. (6). The IMAC is superior to Fan's array LDPC when the block length is particularly long.

$$H = \begin{bmatrix} I & I & I\omega & I\omega^2 & I\omega^3 & \cdots & I\omega^{j} \\ 0 & I & \alpha\omega & \alpha^2\omega^2 & \alpha^3\omega^3 & \cdots & \alpha^{(k-2)}\omega^{j} \\ 0 & 0 & I & \alpha^2\omega^2 & \alpha^4\omega^3 & \cdots & \alpha^{2(k-3)}\omega^{j} \\ \vdots & \vdots & \vdots & I & \alpha^3\omega^3 & \cdots & \vdots \\ 0 & 0 & \cdots & 0 & I & \cdots & \alpha^{(j-1)}\omega^{(k-j)} \end{bmatrix}_{(jp\times kp)} \tag{5}$$

where ω is the quasi cyclic matrix that constructed from identity matrix by cyclically-shifting of the matrix I, i.e. $\omega^{n-1} = I$.

$$I = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}_{(5\times 5)} \quad and \quad \omega = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}_{(5\times 5)} \tag{6}$$

Abematsu [8] has proposed the Size Compatible (SC) -Array LDPC Codes which the **H** matrix given in (7). The design supports arbitrary lengths achieve good error rate performance. It contains very few or no cycle-4 structure. In the SC-array LDPC code, the permutation sub-matrix is decided to eliminate all cycle-4. The proposed cyclic shift $p_{sc}(j,k)$ is expressed by Eq. (8).

$$H=\begin{bmatrix} I & I & \cdots & \cdots & I & \cdots & I \\ \alpha^{p_{sc}(2,k)} & I & \cdots & \cdots & \alpha^{p_{sc}(2,j)} & \cdots & \alpha^{p_{sc}(2,k-1)} \\ \alpha^{p_{sc}(3,k)} & \alpha^{p_{sc}(3,k-1)} & I & \cdots & \alpha^{p_{sc}(3,i)} & \cdots & \alpha^{p_{sc}(3,k-2)} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \cdots & \vdots \\ \alpha^{p_{sc}(j,k)} & \alpha^{p_{sc}(j,k-1)} & \cdots & \cdots & I & \cdots & \alpha^{p_{sc}(2,k-j+1)} \end{bmatrix} \quad (7)$$

where

$$p_{sc}(j,k)=(j-1)(k-1)+\left|\frac{(j-1)(k-1)}{L}\right| \quad (8)$$

This paper uses the idea of Abematsu's matrix that doesn't have the same sub-matrix in same row. It can prevent the cycle-4 when construct the **H** matrix with non-prime number. So, we will get the good performance because it has no cycle-4 structure in the **H** matrix.

B. Our Designs

The IMAC has the high code rate in particular when it is employed for long block size. One drawback is that it cannot support arbitrary code lengths. The performance of IMAC is good when designed with prime sub-matrix. But, we must add dummy bits to comply block length requirement. Non-prime sub-matrix could be designed to corporate the IMAC. However, such a straight forward implementation can result in obtaining a rather poor performance according to the appearance of cycle-4.

In our approach, we separate the design for 2 parts. The first is trying to make quasi cyclic matrix applicable to non-prime sub-matrix. The resulted modification is trying to obtain the row that contains non-identical sub-matrix. We do this by using the idea of the Size Compatible (SC)-Array LDPC Codes.

$$\omega=\begin{bmatrix} 0&1&0&0&0&0\\ 0&0&0&1&0&0\\ 0&0&0&0&0&1\\ 0&1&0&0&0&0\\ 0&0&0&1&0&0\\ 0&0&0&0&0&1 \end{bmatrix}_{(6\times6)} \quad T=\begin{bmatrix} 0&1&0&0&0&0\\ 0&0&0&1&0&0\\ 0&0&0&0&0&1\\ 1&0&0&0&0&0\\ 0&0&1&0&0&0\\ 0&0&0&0&1&0 \end{bmatrix}_{(6\times6)} \quad (9)$$

Shown in (9), the (6×6) ω matrixes make some column to 0's when we use it directly to interleave the array LDPC. It can reduce the performance of the code. So, we do left-shifting of the row 4-6 to generate a new matrix (T) with no cycle-4. Subsequently, we construct T^2 by doing right-shifting of the matrix. T^3 can be obtained similarly. They are shown in (10). Fairly obvious, $T^n=T$. The final **H** matrix then achieved as shown in (11).

In (11), we shift left Abematsu's matrix to the form of MAC to get the properties of MAC. Then, interleave the matrix with T. we show the **H** matrix for the Non-prime IMAC with non-prime sub-matrix size (j = 3, k = 12 and L = 12) by Eq. (13).

$$T^2=\begin{bmatrix} 0&0&1&0&0&0\\ 0&0&0&0&1&0\\ 1&0&0&0&0&0\\ 0&1&0&0&0&0\\ 0&0&0&1&0&0\\ 0&0&0&0&0&1 \end{bmatrix}_{(6\times6)} \quad T^3=\begin{bmatrix} 0&0&0&1&0&0\\ 0&0&0&0&0&1\\ 0&1&0&0&0&0\\ 0&0&1&0&0&0\\ 0&0&0&0&1&0\\ 1&0&0&0&0&0 \end{bmatrix}_{(6\times6)} \quad (10)$$

and

$$H=\begin{bmatrix} I & I & IT & IT^2 & \cdots & \cdots & IT^j \\ 0 & I & \alpha^{p_T(2,3)}T & \cdots & \alpha^{p_T(2,j)}T^{j-1} & \cdots & \alpha^{p_T(2,k-1)}T^j \\ 0 & 0 & I & \alpha^{p_T(3,4)}T^2 & \alpha^{p_T(3,i)}T^{j-1} & \cdots & \alpha^{p_T(3,k-2)}T^j \\ \vdots & \vdots & \vdots & \ddots & \vdots & \cdots & \vdots \\ 0 & 0 & \cdots & \cdots & I & \cdots & \alpha^{p_T(j,k-j+1)}T^j \end{bmatrix} \quad (11)$$

where

$$p_T(j,k)=(j-1)(k-1)+\left|\frac{(j-1)(k-j)}{L}\right| \quad (12)$$

$$H=\begin{bmatrix} I & I & IT & IT^2 & IT^3 & IT^4 & IT^5 & IT^6 & IT^7 & IT^8 & IT^9 & IT^{10} \\ 0 & I & \alpha T & \alpha^2T^2 & \alpha^3T^3 & \alpha^4T^4 & \alpha^5T^5 & \alpha^6T^6 & \alpha^7T^7 & \alpha^8T^8 & \alpha^9T^9 & \alpha^{10}T^{10} \\ 0 & 0 & I & \alpha^1T^2 & \alpha^4T^3 & \alpha^6T^4 & \alpha^8T^5 & \alpha^{10}T^6 & \alpha T^7 & \alpha^3T^8 & \alpha^5T^9 & \alpha^7T^{10} \end{bmatrix} \quad (13)$$

C. Cycles of Purpose Codes

We will consider the general from of a H matrix below.

$$H=\begin{bmatrix} \alpha^{a_0,0} & \alpha^{a_0,1} & \alpha^{a_0,2} & \cdots & \alpha^{a_0,p-1} \\ 0 & \alpha^{a_1,1} & \alpha^{a_1,2} & \cdots & \alpha^{a_1,p-1} \\ 0 & 0 & \alpha^{a_2,2} & \cdots & \alpha^{a_2,p-1} \\ \vdots & \vdots & \vdots & \cdots & \cdots \\ 0 & 0 & \cdots & \cdots & \alpha^{a_{j-1},p-1} \end{bmatrix} \quad (14)$$

A cycle in the Tanner graph of purpose codes with a H matrix and block labels $a_0, a_1, \ldots, a_{j-1}$ that there exists a close path as

$$(i_1,j_1),(i_1,j_2),(i_2,j_2),(i_2,j_3),\ldots,(i_k,j_k),(i_k,j_1).$$

The length of the cycles in the permutation is described as shown below.

$$\alpha^{a_{i1},j_1}\left(\alpha^{a_{i1},j_2}\right)^{-1}\alpha^{a_{i2},j_2}\left(\alpha^{a_{i2},j_3}\right)^{-1}\ldots\alpha^{a_{ik},j_k}\left(\alpha^{a_{ik},j_1}\right)^{-1} \quad (15)$$

From the result of short cycles properties in [7], the cycles of length 4 depends on the following permutation.

$$\alpha^{a_{i1},j_1}\left(\alpha^{a_{i1},j_2}\right)^{-1}\alpha^{a_{i2},j_2}\left(\alpha^{a_{i2},j_3}\right)^{-1}=\alpha^{(a_{i1}-a_{i2})(j_1-j_2)} \quad (16)$$

which is exact when it is equal to 1. This occurs when $(a_{i1}-a_{i2})(j_1-j_2)\equiv 0$, which is clearly impossible since $i_1\neq i_2$ and $j_1\neq j_2$. Hence, there are no cycles-4 for purpose code.

IV. Performance Evaluation

We investigate our formulation by applying it to the long block length code. The 4K block length is of our interest according to its application to magnetic recording

system. The test parameters are shown in table 1 and 2 below. The codes were run for 30 iterations and the performances are observed.

TABLE 1: PARAMETERS FOR 4K BLOCK LENGTH

	j	k	p	R
Prime IMAC	5	61	67	0.918
Prime MAC	5	61	67	0.918
Non-prime IMAC	5	60	68	0.917
Non-prime MAC	5	60	68	0.917
This paper	5	60	68	0.917

TABLE 2: PARAMETERS FOR DIFFERENT NON-PRIME BLOCK LENGTH

Block Length	j	k	p	R
528	3	22	24	0.864
528	4	22	24	0.818
2016	4	42	48	0.905
4080	5	60	68	0.917

Regarding the works published by Eleftheriou and Olcer [4], and Singhaudom [5], we compare the result with that of IMAC for similar block length and code rate. The purposed matrix with non-prime IMAC has offered the similar result compared to the prime IMAC proposed by [5] and better than prime MAC proposed by [4]. BER versus E_b / N_0 plot is shown in Fig. 3.

Our proposed non-prime IMAC has superior performance over the the Non-prime MAC and Non-prime IMAC ones. Non-prime MAC and non-prime IMAC have shown rather poor performance according to generated cycle-4. The proposed matrix does not hold cycle-4 since Abematsu's matrix was interleaved by matrix T.

The modified matrix proposed in this paper was also test for different block length codes. The result is shown in Figure 4. Although we can have a desirable performance of the long codes, the purposed matrix cannot give the good result for the short block length.

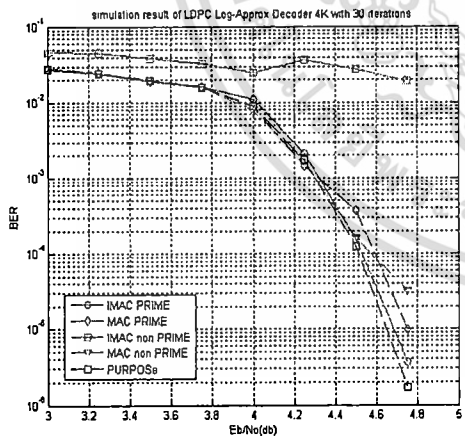


Figure 3. The proposed LDPC and the existing LDPC's performance

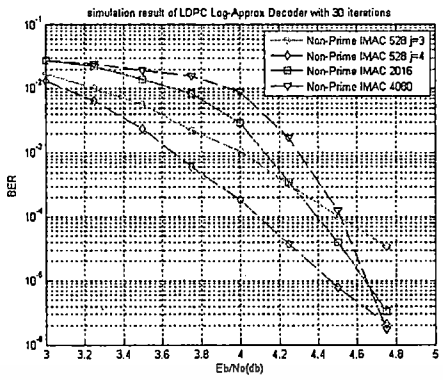


Figure 4. The proposed LDPC for different block length codes

V. CONCLUSIONS

Purpose matrix is designed by use the idea of SC-Array LDPC Codes and IMAC. We can get a parity matrix that has the good performance and can use for arbitrary code lengths in the long block length. For short block length, our idea is not good because it is not near the Shannon limit. This code has comparable performance when compared with the prime IMAC. The purpose matrix has attractive properties such as simple encoding, low error floor and capability of detecting and correcting burst error same as MAC and IMAC.

REFERENCES

[1] R. Gallager, "Low-density Parity-check Code," *IRE Trans.Information Theory*, Jan. 1962, pp.21-28.

[2] D.J.C. Mackey and R. Neal "Near Shannon limit performance of low density parity check codes," *Electronics Letters*, vol.33, pp. 457-458, Mar 1997.

[3] M. Tuchler, J. Hagenauer, E. Eleftheriou, A. Dholakia, C. Weiss, "Application of High-Rate Tail-biting Codes to Generalized Partial Response Channels," Global Telecommunications Conference, 2001

[4] E. Eleftheriou and S. Olcer, "Low-density parity check codes for digital subscriber lines," *Proc. 2002 Int. Conf. on Comm.*, pp.1752-1757., April – May, 2002.

[5] W. Singhaudom, S. Noppankeepong, P. Suphithi, "Design of Hign-Rate Modified Array Codes for Magnetic Recording System,"*ECTI International Conference*, 2007.

[6] D.J.C. MacKay."Good error-correcting codes based on very sparse matrices". *IEEE Trans. Inform. Theory*, 45(2):pp. 399.431, March 1999

[7] J. L. Fan, "Array codes as low-density parity-check codes," in *Proc.* 2nd *Int. Symp. Turbo Codes*, Best, France, Sep 2000, pp 543-546.

[8] D. Abematsu, T. Ohtsuki, S.P.W. Jarot, T. Kashima," Size Compatible (SC)-Array LDPC Codes" *Vehicular Technology Conference*, 2007.

[9] L. Ping and W.K. Leung. "Decoding low density parity check Codes with _nite quantization bits". *IEEE Comm. Letters*, 4(2):pp.62.64,February 2000.

A Design of Parity Check Matrix for Irregular LDPC Codes

Chutima Prasartkaew* and Somsak Choomchuay†

*College of Data Storage Technology and Applications, King Mongkut's Institute of Technology Ladkrabang, BKK, Thailand
Tel:/Fax: + 66-2-326-4731, E-mail: prasartkaew@yahoo.com

†Faculty of Engineering, King Mongkut's Institute of Technology Ladkrabang, BKK 10520, Thailand
Tel: +66-2-326-4222 Ext.114, Fax: +66-2-739-2398, E-mail: kchsomsa@kmitl.ac.th

Abstract— **This paper outlines a work on a design of parity check matrix for Irregular LDPC codes. The design is based on the adjustment of the modified array LDPC codes and interleave-modified array LDPC codes. The code rate of 0.930 is obtained. LDPC Codes based on this design suits the short and medium block length.**

I. INTRODUCTION

Error Correction Codes (ECC) is one of many tools made available for achieving consistent data transmission. Low-Density Parity-Check codes (LDPC), as ones of many kinds, are also linear block codes that have been studied vastly in this decade. The main advantages of the codes are that they provide the performance at that close to limited capacity for many different channels and linear time complex algorithms for decoding. They also suit well the parallel realization.

In this paper we propose a modified method to obtain parity check matrices. The obtained result is comparable to a published work [1] but with a higher code rate. The rest of this paper is organized as follows: General perspective of LDPC codes is given in section II where encoder and decoder are also included. A design of parity check matrices is outlined in section III and the corresponding performance tests are reported in section IV. The paper is concluded in section V.

II. LDPC CODES

Of its discovery in 1960 by Gallager [2], the LDPC code has been ignored for some ten years. This was because the code itself is quite complex. In the same time the more highly-structured code; Reed Solomon code was introduced [3]. The introduction of Turbo code by Berrou, Glavieux and Thitimajshima in 1993 [4] has drawn great attention to researchers since the code performance is close to Shannon limit. LDPC codes were recovered in 1998 by Richardson and Urbanke [5] and in 1999 by Mackey [6]. LDPC became more popular and widely developed for wider area of applications including communications and data storage.

There are two different ways to represent LDPC codes; matrix representation and graphical representation. In the matrix point of view, as it is named, LDPC codes hold small number of "1" in each row and column, i.e. $W_c << n$ and $W_r << m$ for a dimension $m \times n$ parity matrix. This can provide large minimum distance of the code. However such a circumstance results a large parity check matrix. In the graph point of view, Tanner graph [7] is an efficient graphical representation of LDPC codes. There are m check nodes (c-nodes; number of parity bits) and n variable nodes (v-nodes; number of bits in a codeword).

LDPC codes are said to be regular if W_c is constant for every column, and $W_r = W_c(n/m)$. If the parity matrix H is low density but the number of "1" in each row or column are not constant, the code is said to be an irregular one.

A. Encoder

Similar to all other linear block codes, we have the relation;

$$C_{(1\times n)} H^T_{(n\times m)} = 0 \tag{1}$$

Where C is a codeword matrix, and H is a parity check matrix. In a systematic form, C can be written as:

$$C_{(1\times n)} = \left[m_{(1\times m)} \mid p_{(1\times n-m)} \right] \tag{2}$$

Where $p_{(1\times n-m)}$ denotes the parity portion, and $m_{(1\times m)}$ denotes the message portion. With $H = \left[H_1 \, H_2 \right]$ or $H^T = \begin{bmatrix} H_1^T \\ H_2^T \end{bmatrix}$ we can have;

$$CH^T = \left[m \mid p \right] \begin{bmatrix} H_1^T \\ H_2^T \end{bmatrix} = mH_1^T + pH_2^T = 0 \tag{3}$$

Or

$$p = mH_1^T + \left(H_2^T \right)^{-1} \tag{4}$$

The task of the encoder is then to compute the parity matrix P that can be directly appended to the message to produce the codeword.

1-4244-4522-6/09/$25.00 ©2009 IEEE

For the matrix H to be more manageable, LU decomposition method can be preferably applied; i.e. $[H]=[L][U]$. Thus,

$$\begin{bmatrix} l_{11} & 0 & \cdots & 0 \\ l_{21} & l_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ l_{n1} & l_{n2} & \cdots & l_{nn} \end{bmatrix} \underbrace{\begin{bmatrix} u_{11} & u_{12} & \cdots & u_{1n} \\ 0 & u_{22} & \cdots & u_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & u_{nn} \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \\ \vdots \\ p_n \end{bmatrix}}_{Y} = \begin{bmatrix} m_1 \\ m_2 \\ \vdots \\ m_n \end{bmatrix} \quad (5)$$

Let $[Y]=[U][P]$, then we can use forward substitution to solve $[L][Y]=[M]$.

$$\begin{bmatrix} l_{11} & 0 & \cdots & 0 \\ l_{21} & l_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ l_{n1} & l_{n2} & \cdots & l_{nn} \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} m_1 \\ m_2 \\ \vdots \\ m_n \end{bmatrix} \quad (6)$$

Finally, the backward substitution is employed to solve for P of which $[U][P]=[Y]$. There, we can get $\{p_i\}$ as needed.

$$\begin{bmatrix} u_{11} & u_{12} & \cdots & u_{1n} \\ 0 & u_{22} & \cdots & u_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & u_{nn} \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \\ \vdots \\ p_n \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} \quad (7)$$

B. Decoder

There are several methods used in decoding the LDPC codes. Each was derived individually. These are, for instance, Believe Propagation (BP), Sum-Product (SP), and Message Passing (MP).

The Tanner graph (Fig. 1) can be drawn directly from the H matrix (given in (8)) as shown be below:

$$H = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix} \quad (8)$$

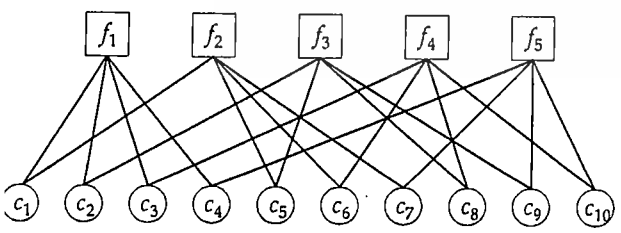


Fig. 1. Tanner graph of the H matrix given in (8)

The graph contains m check nodes (number of parity bits) and n variable notes (number of bits in a codeword). Check node f_i is connected to a variable node c_i if the element h_{ij} of H is a "1".

In the Log-domain Sum-Product algorithm, the message passes between check nodes and variable nodes. In each pass the log likelihood ratio (LLR) is recorded for is probability of its likely symbol. In summary, the decoder goes through 5 steps as follows:

Step 1:

Compute the initial value of $L(q_{ij})$ transmitted from the variable node i to check node j; for all $i; 1 \le i \le n$.

$$L(q_{ij}) = L(c_i) = \frac{2y_i}{\sigma^2} = LLR_i = \log\left(\frac{p_{(c_i=0)|y_i}}{p_{(c_i=1)|y_i}}\right) \quad (9)$$

Where $L(c_i)$ denotes log likelihood ratio

σ^2 denotes derivation of white noise

$p_{(c_i=x)}|y_i$ denotes probability for given input y_i

Step 2:

Compute $L(r_{ji})$ transmitted from the check node j to variable node i; for all $i; 1 \le i \le n$. Let $\phi(x) = \log\left(\frac{e^x+1}{e^x-1}\right)$.

$$L(r_{ji}) = \prod_{i' \in V_{j\backslash i}} \alpha_{i'j} \phi\left(\sum_{i' \in V_{j\backslash i}} \phi\left(\beta_{i'j}\right)\right) \quad (10)$$

Where $\alpha_{ij} = \mathrm{sgn}\{L(q_{ij})\}$, and $\beta_{ij} = |L(q_{ij})|$.

Step 3:

Modify $L(q_{ij})$ and used it as the data transmitted from the variable node i to check node j; for all $i; 1 \le i \le n$.

$$L(q_{ij}) = L(c_i) + \sum_{j' \in C_{i\backslash j}} L\left(r_{j'i}\right) \quad (11)$$

Step 4:

Compute the soft output.

$$L(Q_i) = L(c_i) + \sum_{j \in C_i} L\left(r_{ji}\right) \quad (12)$$

Step 5:

The soft output obtained in step 4 is then used in the hard decision as,

$$\hat{c}_i = 1 \text{ if } L(Q_i) < 0, \text{ otherwise } \hat{c}_i = 0.$$

III. Designs of Parity Check Matrices

Resulted design of a H matrix is a crucial step in obtaining the code performance. A design of the parity check matrix outlined in this paper is based on the adjustment of the modified array LDPC and interleave-modified array LDPC.

A. Some Previous Works

Fan [8] has introduced the array structure parity matrix that can offer comparable performance when compared to a random generated parity matrix. Some other features are: low noise floor and no existence of cycle of 4. The said matrix is shown below.

$$H(p,j,k) \triangleq \begin{bmatrix} I & I & I & \cdots & I \\ I & \alpha & \alpha^2 & \cdots & \alpha^{k-1} \\ I & \alpha^2 & \alpha^4 & \cdots & \alpha^{2(k-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ I & \alpha^{j-1} & \alpha^{2(j-1)} & \cdots & \alpha^{(j-1)(k-1)} \end{bmatrix}_{(jp \times kp)} \quad (13)$$

Where I is an identity matrix $(p \times p)$.

α is a position permutation matrix $(p \times p)$.

This yields the code rate of $R = 1 - (\frac{pj-j+1}{p^2})$.

Eleftheriou and Olcer [9] have proposed the modified array structure (MAC) by applying cyclic shift to Fan's array. This structure yields the code rate of $R = 1-(j/k)$. MAC offers superior performance to Fan's array as it can reduce number of "1" in the lower triangle. The effort also led to a simpler encoder.

$$H = \begin{bmatrix} I & I & \cdots & I & I & \cdots & \cdots & I \\ 0 & I & \alpha & \cdots & \alpha^{(j-2)} & \alpha^{(j-1)} & \cdots & \alpha^{(k-2)} \\ 0 & 0 & I & \cdots & \alpha^{2(j-3)} & \alpha^{2(j-2)} & \cdots & \alpha^{2(k-3)} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \cdots & I & \alpha^{(j-1)} & \cdots & \alpha^{(j-1)(k-j)} \end{bmatrix}_{(jp \times kp)} \quad (14)$$

Singhaudom *et. al.* [10] has proposed the interleaved modified array LDPC or IMAC by introducing the quasi-cyclic matrix into the cyclic shift of (13). This interleaved LDPC of which the parity matrix given below is superior to Fan's array LDPC when the block length is particularly long.

$$H = \begin{bmatrix} I & I & I\omega & I\omega^2 & I\omega^3 & \cdots & I\omega^j \\ 0 & I & \alpha\omega & \alpha^2\omega^2 & \alpha^3\omega^3 & \cdots & \alpha^{(k-2)}\omega^j \\ 0 & 0 & I & \alpha^2\omega^2 & \alpha^4\omega^3 & \cdots & \alpha^{2(k-3)}\omega^j \\ \vdots & \vdots & \vdots & I & \alpha^3\omega^3 & \cdots & \vdots \\ 0 & 0 & \cdots & 0 & I & \cdots & \alpha^{(j-1)}\omega^{(k-j)} \end{bmatrix}_{(jp \times kp)} \quad (15)$$

B. Our Construction

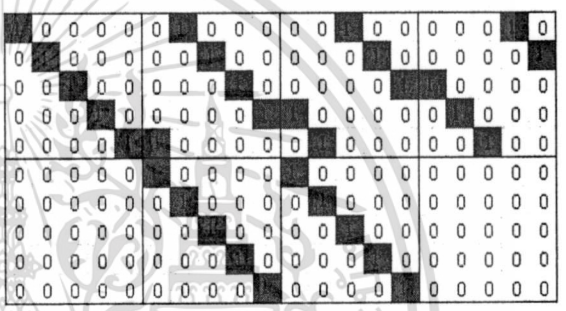
The implementation complexity can be greatly reduced significantly if a suitable structured parity check matrix [11] is assigned. In our construction, the matrix is based on Eleftheriou's and Singhaudom's works. We reduced "1" in the upper triangle by applying matrix transposition, and row and column interchanging to avoid cycle of 4. As a result we can have 4 forms of the new matrix. Unfortunately, the first two have less potential. The third offers the feature of generality and simplified encoding as well as the ability of error correction. The fourth has failed its decoding capability. We, hence, consider the third achieve in this paper. The procedure of obtaining the parity matrix is summarized below.

$$\begin{bmatrix} I & I \\ 0 & I \end{bmatrix} \Rightarrow \begin{bmatrix} I & 0 \\ I & I \end{bmatrix} \Rightarrow \begin{bmatrix} I & I & I & 0 \\ 0 & I & I & I \end{bmatrix} \Rightarrow$$

$$\begin{bmatrix} I & I & I & I \\ 0 & I & I & 0 \end{bmatrix} \Rightarrow \begin{bmatrix} I & \alpha & \alpha^2 & \alpha^3 \\ 0 & I & I & 0 \end{bmatrix}$$

Therefore,

$$H = \begin{bmatrix} I & \alpha & \alpha^2 & \alpha^3 \\ 0 & I & I & 0 \end{bmatrix} \quad (16)$$



IV. Performance Evaluation

Upon preliminary investigation of long block length (such as 4096 bits) decoding, we found that our matrix cannot compete the IMAC one. But for the short and medium block length, the proposed matrix offers considerably good result. The test parameters are tabulated in table 1, 2 and 3 below.

TABLE 1: PARAMETERS FOR SHORT BLOCK TESTING

j	3	3	4	4
k	15	18	15	18
p	17	29	17	29
$R = 1-(j/k)$	0.80	0.833	0.733	0.778
$m = p(k-j)$	204	435	187	406
Parity $= jp$	51	87	68	116
$c = kp$	255	522	255	522
Iterations	5, 10, 20			

TABLE 2: PARAMETERS FOR MEDIUM BLOCK TESTING

j	3	3	4	4
k	31	43	31	43
p	37	47	37	47
$R = 1-(j/k)$	0.903	0.93	0.871	0.907
$m = p(k-j)$	1,036	1,880	999	1,833
Parity $= jp$	111	141	148	188
$c = kp$	1,147	2,021	1,147	2,021
Iterations	5, 10, 20			

TABLE 3: PARAMETERS FOR LONG BLOCK TESTING

j	4	4	5	5
k	61	47	61	47
p	67	89	67	89
$R = 1-(j/k)$	0.934	0.915	0.918	0.894
$m = p(k-j)$	3,819	3,827	3,752	3,738
$Parity = jp$	268	356	335	445
$c = kp$	4,087	4,183	4,087	4,183
Iterations	10, 20, 30			

The designed LDPC code offers the code rate of 0.833, 0.930 and 0.915 when applied to short block (j=3, c=522), medium block (j=3, c=2021) and long block (j=4, c=4183) respectively. With these parameters, BER are investigated and the obtained results are shown in Fig.2 and Fig.3 below.

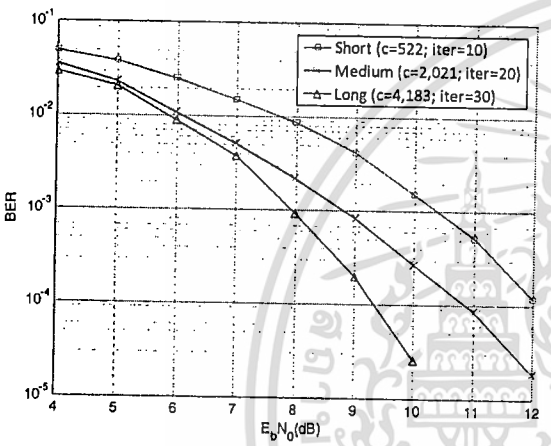


Fig. 2. Short, medium and long block length's performance

We also has compared our result to an existing published work proposed by Rakibul *et al* [1]. With similar block length, Rakibul's LDPC gives slightly better performance compared to ours. However, their code rate is only 0.5 whilst ours is 0.930. BER versus E_bN_0 plot is shown in Fig. 3 below.

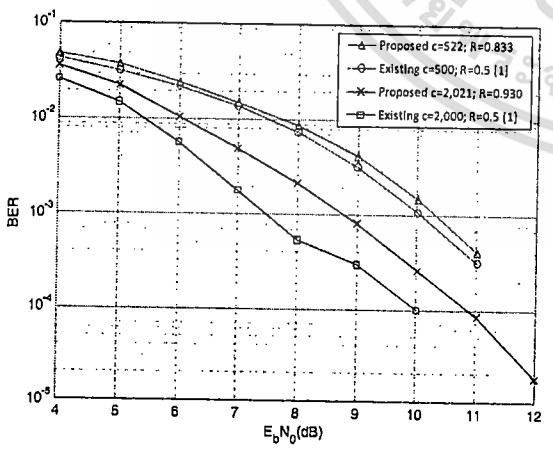


Fig. 3. The proposed LDPC and the existing LDPC's performance

In additions, we investigated number of iterations versus the performance obtained. Our LDPC codes were tested at 5, 10, and 20 iterations. At 5 iterations, BER of both short and medium block length are similar. BER can be improved when we increased the number of iterations to 10. When the number of iteration is of 20, only the BER of a medium block length can be slightly improved. However, the obtained BER is still not so low. This is quite common that one cannot get so low BER when the block length is not large.

V. CONCLUSIONS

Combining modified array with interleaved modified array by avoiding the cycle of 4, we can get a different parity matrix that leads to a moderate performance irregular LDPC code. This code has comparable performance compared with the regular one; in particular when the block lengths are of short and medium. For the long block length, the obtained LDPC cannot compete the existing the interleaved modified array. Based on our idea, there could be opportunity to develop a parity check matrix to obtain the performance of IMAC but with less complexity.

REFERENCES

[1] Mohammad Rakibul, Jinsang Kim, "On the use of QC-LDPC code for data transfer using short and medium block length," *Conference ICACT 2009* Feb. 15-18, 2009.

[2] R. Gallager, "Low-density Parity-check Code," *IRE Trans. Information Theory*, Jan. 1962, pp. 21-28.

[3] Reed, I. S. and Solomon, G., "Polynomial Codes Over Certain Finite Fields," *SIAM Journal of Applied Math.*, vol. 8, 1960, pp. 300-304.

[4] C Berrou, A. Glavieux and P. Thitimajshima, "Near Shannon limit error-correcting Coding and Decoding," *Proc. IEEE Int. Conf. Comm.*, pp. 1064-1070, May 1993.

[5] T.J. Richardson, M.A. Shokrollahi, and R.L. Urbanke, "Design of Capacity-approaching Low-density Parity-check Codes," *IEEE Trans. on Info. Theory*, Vol. 47, pp. 619-637, Feb. 2001.

[6] D.J.C. Mackey and R. Neal, "Near Shannon Limit Performance of Low Density Parity Check Code," *Electronics Letter*, Vol. 33, Mar 1997, pp. 457-458.

[7] R.M. Tanner, "A Recursive Approach to Low Complexity Code," *IEEE Trans. Information Theory*, Sept. 1981, pp. 533-547.

[8] J.L. Fan, "Array Codes as low-density parity-check codes," *Proc. 2nd Int. Symp. Turbo Code*, Beit, France, pp. 543-546, Sep. 2000.

[9] E. Eleftheriou and S. Olcer, "Low-density parity-Check Codes for Digital Subscriber Lines," *Proc. 2002 Int. Conf. on Comm.*, pp. 1752-1757, April-May 2002.

[10] W. Singhaudom, S. Noppankeepong, P. Suphithi, "Design of Hign-Rate Modified Array Codes for Magnetic Recording System," *ECTI International Conference*, May 2007.

[11] O. Othman Khalifa, S. Khan, M. Zaid, and M. Nawawi, "Performance Evaluation of Low Density Parity Check Codes," *International Journal of Computer Science and Engineering*, Vol. 2, No. 2, 2008.

Parity Check Matrix Construction via Magic Square Based Algorithm

Chutima Prasartkaew* and Somsak Choomchuay†

*Faculty of Science and Technology, Rajamangala University of Technology Thanyaburi, Pathumthani, Thailand
Tel:/Fax: + 66-2-549-4197, E-mail: prasartkaew@yahoo.com

†Faculty of Engineering, King Mongkut's Institute of Technology Ladkrabang, BKK 10520, Thailand
Tel: +66-2-329-8344 Ext.114, Fax: +66-2-329-8346, E-mail: kchsomsa@kmitl.ac.th

Abstract— **This paper presents a construction algorithm for the short block irregular LDPC codes. By applying a magic square theorem as a part of the matrix construction, a newly developed algorithm, the so-called Magic Square Based Algorithm (MSBA), is obtained. The modified array codes are focused on in this study since the reduction of 1s can lead to simple encoding and decoding schemes. Simulation results based on AWGN channels show that with the code rate of 0.8 and SNR = 5 dB, the BER of 10^{-7} can be obtained whilst the number of decoding iteration is relatively low.**

Keywords: Irregular LDPC codes, Magic square, Short block length, Parity check matrix.

I. INTRODUCTION

In the past decade, low-density parity-check codes (LDPC) had drawn a great attention to researchers in many areas, including wireless communications and data storage technology. The most attractive feature of such codes is its distinctive near Channon's limit performance. However, according to its complexity, the codes have been almost forgotten for more than twenty years after its invention in 1962. The code performance is very much relied on the structure of parity check matrix (**H**). Generation of such a parity check matrix of the codes can be generally performed randomly or systematically. However the random generated one can be very computational intensive; in particular when the size of the matrix is big. The systematic one, on another hand, can be systematically constructed with some rules and need less resource. There are huge works that try to seek for the best algorithm which is so far perhaps not actually yet found. A method proposed in this paper is also a case of those efforts.

LDPC codes are linear block codes defined by a sparse parity check matrix, originally proposed by [1]. The code performance can be made close to Shannon's limit. [2] investigated the performance of the code when it is applied to an erasure channel and pointed out that the decoding complexity is linear when using Sum-product Algorithm (SPA). Therefore, there is a constraint for codes with long block length.

Array codes are error-correcting codes that have the capability to correct error bursts. [3] has mentioned that the soft iterative decoding scheme can be applied to array codes. Most previous studies [1-4] have focused on regular structure of the **H** matrix, [5] improved the performance of the code by introducing an irregular matrix. [7] proposed Size Compatible (SC) array LDPC codes that the new cyclic shift method is used instead of circulant permutation matrices. In order to eliminate cycle-4, they have permuted a sub-matrix position based on row number. Subsequently, [8] proposed the arbitrary length comparable irregular structure which is similar to the SC-Array proposed by [7]. Their formulation is also based on the Chinese Remainder Theorem (CRT) method proposed by [4]; however, the non-prime number parameters are used. Regardless of the construction complexity, the obtained code yields good design flexibility. On the dim side, the code performance deteriorates when it is applied to short blocks.

This research aims to reduce the construction complexity, compared to [8], and to achieve a better performance of short irregular LDPC codes. Based on the modified array codes proposed by [6], together with the use of non-prime parameters as with [7], this paper alternatively proposes an algorithm based on the magic square theorem. As a matter of fact, the structure of the parity check matrix contains many sub-permutation matrices and they should be cyclically shifted. With this regard, there is a chance for the magic square to be useful. Rather than random generation, the shifting orders are implicitly generated by a known procedure of magic square generation.

The rest of this paper is organized as follows: magic square theorem is briefly reviewed in section II. Some necessary conditions for the parity check matrix in order that the magic square could be incorporated are elaborated on in section III. New construction details of the parity check matrices are given in section IV. The simulation procedure and results are reported in section V. Finally, this paper is concluded in section VI.

II. MAGIC SQUARE

A magic square is a square array of the numbers 1, 2, 3, ...,z^2, with the property that the summation of every row, column and both diagonals, is the same number. According to how they are generated, the existing magic squares can be classified into four groups:

978-1-4673-1157-1/12/$31.00 © 2012 IEEE

Group 1: Astrological planets

In about 1510, Heinrich Cornelius Agrippa wrote *De Occulta Philosophia*, drawing on the Hermetic and magical works of Marsilio Ficino and Pico della Mirandola, and in it he expounded on the magical virtues of seven magical squares of orders 3 to 9, each associated with one of the astrological planets: Saturn, Jupiter, Mars, Sol, Venus, Mercury, and Luna [10].

Group 2: Odd order

The Siamese or staircase method is derived by De la Loubère [11]. Starting from the central column of the first row with the number 1, the fundamental movement for filling the cells is diagonally up and right, one step at a time. If a filled cell is encountered, the next move should then be vertically down one cell, before continuing as mentioned above. When a move leaves the square, it is wrapped around to the last row or first column, respectively.

Group 3: Doubly even order (z is divisible by four)

All the numbers are written in order from left to right across each row in turn, starting from the top left corner. Numbers are then either retained in the same place or interchanged with their diametrically opposite numbers in a certain regular pattern.

Group 4: Singly even order (z is not divisible by four)

In this group, there are two existing construction methods: Ralph Strachey and LUX methods. In the Ralph Strachey method, the magic square is divided into equal quarters. For example, a 6×6 square will give four 3×3 squares. Each of these can then be formed using De la Loubère's method for odd order squares [11]. Another method for generating singly even was found by J. H. Conway and is called the LUX method. The shapes of the letters L, U, and X naturally are used for the filling order; hence came the name of the algorithm [9].

III. Construction of Parity Check Matrices

As mentioned in the previous section, the structure of the parity check matrix affects the coding and performance substantially. A good matrix must be carefully designed. In this proposed algorithm, the matrix is constructed using the different numbers in a magic square as cyclic shifting orders of each sub-permutation. Detailed procedures are as follows:

1) Define the construction parameters of the code; j, k and p. All are integers and must be greater than or equal to 3 where $j < k \le p$. Let λ be the summation value of positions which will be placed by the number from the magic square. To assure that there are enough numbers for all sub-permutation shifting whilst all numbers are different, the condition given in Eq. (5) must be satisfied.

$$\lambda = (j\times k) - \left(\frac{j(j+1)}{2}\right) - (k-1) \le z^2 \qquad (5)$$

For example, the construction of the parity check matrix for irregular LDPC codes with the block length of 513 and with the code rate of 0.7 (j=3, k=9 and p=57) is demonstrated. According to equation (5), z must be greater than 4.

2) Generate the $z\times z$ magic squares, with respect to the block length and code rate. In order to cover all possible magic squares in group 1 (z=3 to 9), the magic square sizes of 4 to 9 are taken into this consideration.

3) Use the magic square numbers as shifting orders of each sub-permutation matrix (size of $p\times p$). To investigate the variation in performance as well as to find the best structure, three methods of shifting are proposed. In method I, all numbers are picked from the magic square without any rearrangement. In method II, all numbers are picked from the magic square with some rearrangement. The numbers used are the original numbers from the magic square where there is no relationship between each row and all the rows are the same. The number 1 was searched for and thereafter the numbers were selected; thus, all used numbers are independent from each other. In method III, some numbers are selected from the magic square with some rearrangement. To ensure the normal distribution of numbers, only half of all the numbers are taken from the magic square and the rest are randomly selected.

Method I: This method entails constructing the parity check matrix using the original numbers and positions of the magic square where the relationship between all the rows is that they have the same summation product (magic square property). All sizes of magic squares that relate to the block length and rate are investigated.

a) The parity check matrix can be arranged as shown in Table 1, where the letter X stands for the appropriate shifting order of numbers for each sub-permutation matrix.

Table 1. Irregular pattern of the parity check matrix.

H	1	2	3	4	5	6	7	8	9
1	I	I	I	I	I	I	I	I	I
2	0	I	X	X	X	X	X	X	X
3	0	0	I	X	X	X	X	X	X

b) The second and the third rows of the parity check matrix are filled with the numbers obtained from the first and the second rows of the magic square. This is done one by one, interchanging the positions of the Xs in each row. Depending on the size of the magic square, there are three possible placing conditions.

b.1) If the numbers in the first row of the magic square are less than X in the first row of the parity check matrix, replace the remaining Xs with the column number (as shown in Table 2, where a 5×5 magic square shown in Fig. 1 was used). If the present number of the column has already been used, try the closest number which has not been used and replace all remaining Xs. For example, the first remaining X is in the eighth column; however, 8 and 7 have already been used in the previous step. Thus, 9 is used for this position. In the same manner, the remaining Xs will be found. The

resulted matrix is shown in Table 3.

11	24	7	20	3
4	12	25	8	16
17	5	13	21	9
10	18	1	14	22

Fig. 1. A 5×5 magic square (Mars)

Table 2. Replacement of two rows of a magic square

H	1	2	3	4	5	6	7	8	9
1	I	I	I	I	I	I	I	I	I
2	0	I	11	24	7	20	3	X	X
3	0	0	1	4	12	25	8	16	X

Table 3. Replace Xs with column number

H	1	2	3	4	5	6	7	8	9
1	I	I	I	I	I	I	I	I	I
2	0	I	11	24	7	20	3	*9*	*10*
3	0	0	1	4	12	25	8	16	*6*

b.2) If the numbers in the first row of the magic square are enough for replacement Xs, row by row; but some numbers are greater than or equal to p, then one should pick the number from the next row. The number that is equal or greater than p can yield the same result, according to modulo operation (e.g. $58 \bmod 57 \equiv 1$). For example, the numbers in the first row of the 8×8 magic square shown in Fig. 2 are 8, 58, 59, 5, 4, 62, 63, and 1. Selection of this row should be discarded since this row contains some numbers that are greater than p which is 57. The number from the subsequence rows must be used instead. As a result, the second and the third rows of the matrix can be successfully filled and shown in Table 4.

8	58	59	5	4	62	63	1
49	15	14	52	53	11	10	56
41	23	22	44	45	19	18	48
32	34	35	29	28	38	39	25
40	26	27	37	36	30	31	33
17	47	46	20	21	43	42	24
9	55	54	12	13	51	50	16
64	2	3	61	60	6	7	57

Fig. 2. A 8×8 magic square (Mercury).

Table 4. Using the second and third rows.

H	1	2	3	4	5	6	7	8	9
1	I	I	I	I	I	I	I	I	I
2	0	I	49	15	14	52	53	11	10
3	0	0	1	41	23	22	44	45	19

b.3) If the numbers in the first row of the magic square are enough for Xs replacement row by row, but some values in all rows of the magic square are higher than or equal to p, their p modulo is used as the replacement. For example, the numbers in the first row of a 9×9 magic square shown in Fig. 3 are 37, 78, 29, 70, 21, 62, 13, 54, and 5. The numbers in the second row are 6, 38, 79, 30, 71, 22, 63, 14, and 46. From this list, there are some numbers that are greater than p such as 78, 70, 62, 79, and 71. Their p modulo are 21, 13, 5, 22, and 14 respectively. If the p modulo numbers are not yet used, they can be interchanged. If these numbers have already been used, then the closest number can be tried. The result of this rearrangement is shown in Table 5.

37	78	29	70	21	62	13	54	5
6	38	79	30	71	22	63	14	46
47	7	39	80	31	72	23	55	15
16	48	8	40	81	32	64	24	56
57	17	49	9	41	73	33	65	25
26	58	18	50	1	42	74	34	66
67	27	59	10	51	2	43	75	35
36	68	19	60	11	52	3	44	76
77	28	69	20	61	12	53	4	45

Fig. 3. A 9×9 magic square (Luna).

Table 5. Using p modulo

H	1	2	3	4	5	6	7	8	9
1	I	I	I	I	I	I	I	I	I
2	0	I	37	*20*	29	*12*	21	*5*	13
3	0	0	I	6	38	*23*	30	*14*	22

With the method detailed above one can construct the parity check matrix using some of the other magic squares. For instance, we can use a magic square in group 4 such as Strachey and LUX magic squares shown in Fig. 4 and Fig. 5. Matrices for irregular LDPC codes with the same code length and rate are shown in Tables 6 and 7 respectively.

8	1	6	26	19	24
3	5	7	21	23	25
4	9	2	22	27	20
35	28	33	17	10	15
30	32	34	12	14	16
31	36	29	13	18	11

Fig. 4. A 6×6 magic square (Strachey method).

Table 6. Obtained matrix using of Strachey-6×6.

H	1	2	3	4	5	6	7	8	9
1	I	I	I	I	I	I	I	I	I
2	0	I	8	I	6	26	19	24	9
3	0	0	I	3	5	7	21	23	25

32	29	4	1	24	21
30	31	2	3	22	23
12	9	17	20	28	25
10	11	33	17	10	15
13	16	36	33	5	8
14	15	34	35	6	7

Fig. 5. A 6×6 magic square (LUX method).

Table 7. Obtained parity check matrix using of LUX-6×6

H	1	2	3	4	5	6	7	8	9
1	I	I	I	I	I	I	I	I	I
2	0	I	32	29	4	1	24	21	9
3	0	0	1	30	31	2	3	22	23

Method II: Method II aims to construct the parity check matrix using the original numbers and positions of the magic square without the equality of summation. All used numbers are independent of each other. This relationship is broken down by searching for number 1 in the magic square and to begin picking up numbers from this position (including the number 1). To investigate for all sizes of magic squares which relate to block length and rate, the following steps are used:

a) The irregular pattern of the parity check matrix is prepared, as shown in Table 1.

b) Xs are replaced with the magic square elements, one by one. Use the 1 in the magic square as the starting point, start replacing the first X with 1. All Xs are replaced in order from left to right across each row in turn. During this replacement, if the number is greater than or equal to p, such a number is skipped and the next number is used. As a result, the constructed matrix using 5×5 and 9×9 magic squares shown in Fig. 1 and Fig. 3 are shown in Table 8 and Table 9 respectively.

Table 8. Method II with 5×5 magic square.

H	1	2	3	4	5	6	7	8	9
1	1	1	1	1	1	1	1	1	1
2	0	1	1	14	22	23	6	19	2
3	0	0	1	15	11	24	7	20	3

Table 9. Method II with 9×9 magic square.

H	1	2	3	4	5	6	7	8	9
1	1	1	1	1	1	1	1	1	1
2	0	1	1	42	34	27	10	51	2
3	0	0	1	43	35	36	19	11	52

Method III: This method aims to construct the parity check matrix using some of the numbers in a magic square. All are independent from each other and the rest of the numbers are arranged to ensure the normal curve distribution property of all the used numbers. To investigate all possible sizes of the magic squares which relate to block length and rate, the following steps are used:

a) The irregular pattern of the parity check matrix, as shown in Table 10, is prepared.

Table 10. Irregular pattern for method III.

H	1	2	3	4	5	6	7	8	9
1	1	1	1	1	1	1	1	1	1
2	0	1	X	X	X	X	X	X	X
3	0	0	1	β_1	Y_1	Y_2	β_2	Z_1	Z_2

b) Xs in the second row of the parity check matrix are replaced with the magic square elements, one by one. Use the '1' in the magic square as the starting point. All Xs are replaced in order from left to right across each row in turn. During this replacement, if the number is greater than or equal to p, it should be skipped and the next number should be used.

c) β_1 (in the third row) will be either $p/2$ (if p is even) or $(p-1)/2$ (if p is odd). If this value is used, try another closest number for this position. Replace Y_1, Y_2, ... with β_1-1, β_1-2, ... , respectively. During this replacing, if the number is already used, try another lower one.

30	39	48	1	10	19	28
38	47	7	9	18	27	29
46	6	8	17	26	35	37
5	14	16	25	34	36	45
13	15	24	33	42	44	4
21	23	32	41	43	3	12
22	31	40	49	2	11	20

Fig. 6. A 7×7 magic square (Siamese method).

d) β_2 will be either $\beta_1/2$ (if β_1 is even) or $(\beta_1-1)/2$ (if β_1 is odd). If this value has already been used, try another lower one. Replace Z_1, Z_2, ... with β_2-1, β_2-2, ... , respectively. During this replacing, if the number is already used, try another lower one. Given here as an example, the resulted matrix using a 7×7 magic squares is shown in Fig. 6 shown in Table 11.

Table 11. Method III with 7×7 magic square.

H	1	2	3	4	5	6	7	8	9
1	1	1	1	1	1	1	1	1	1
2	0	1	1	10	19	28	38	47	7
3	0	0	1	27	26	25	14	13	12

Figure 7 shows the flow chart of the construction procedures.

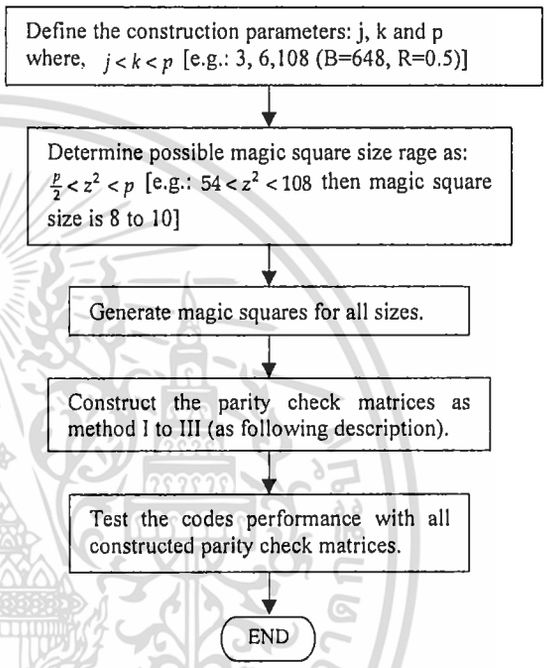


Fig. 7. Matrix construction procedures

IV. Performance Evaluation

To obtain the best parity check matrix construction algorithm based on MSBA, three rearrangement methods proposed in the previous section are investigated and the best one will be chosen. Each method is investigated using the same conditions of block length, code rate, number of iteration and magic square size. A total of 27 constructed parity check matrices (shown in Table 12) were individually investigated.

Table 12. List of all magic square for block length of 513.

Group	Group alias	No.	Size (z×z)
G.1	Jupiter	I 01,II 01,III 01	4x4
	Mars	I 02,II 02,III 02	5x5
	Sol	I 03,II 03,III 03	6x6
	Venus	I 04,II 04,III 04	7x7

G.2	Odd order	I 05,II 05,III 05	5x5
		I 06,II 06,III 06	7x7
G.3	Doubly even	I 07,II 07,III 07	4x4
G.4	Singly even	I 08,II 08,III 08	6x6 (LUX)
		I 09,II 09,III 09	6x6 (Strachey)

It is shown in Fig. 8 that the code performance cannot be further improved when the number of iteration is above 30. Therefore, MSBA can offer the best performance at iteration of 30. This number is used throughout the study.

For the code with the length of 513, it has been shown that method III is the best one and iteration of 30 is appropriate for this method. The code performance for a longer block length with a fixed rate is also investigated. For the length of 1010, among all the constructed parity matrices, the best decoding performance can be obtained by using the 9×9 magic square (Group 2: Siamese method) as shown in Fig. 9. It can be noted that when the block length is varied, the appropriate magic square group and size may be changed.

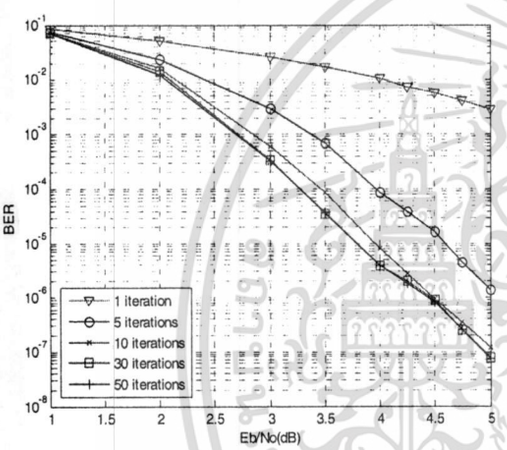


Fig. 8. Code Performances as a function of iterations (block length of 513 with 6×6 magic square)

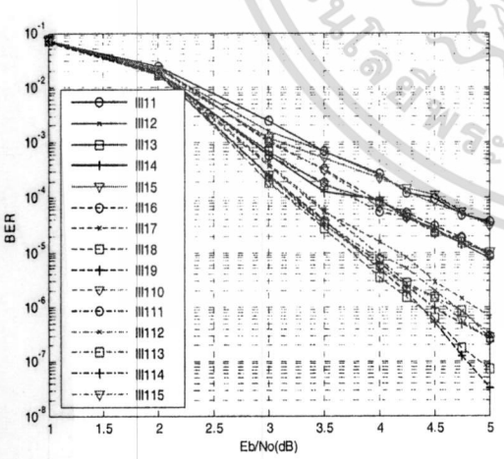


Fig. 9. Test result block length of 1010 (R=0.7)

To investigate the merit of the proposed algorithm, the proposed MSBA is compared with MAC of [6] and CRT of [8] at the same block length and code rate. Test parameters are given in Table 13 for block lengths: 513, 531, 1010, 1020 (at code rate of 0.7) and 1005, 1020 (at code rate of 0.8). It should be noted that there are some differences in construction parameters which are due to the limitations of each construction method. Finally, it is fairly reasonable to define the best parameters for each method in the same class of irregular LDPC codes.

The obtained results (as shown in Fig. 10 to 12) show that MSBA outperforms the others for all block lengths and code rates, especially at SNR = 5 dB.

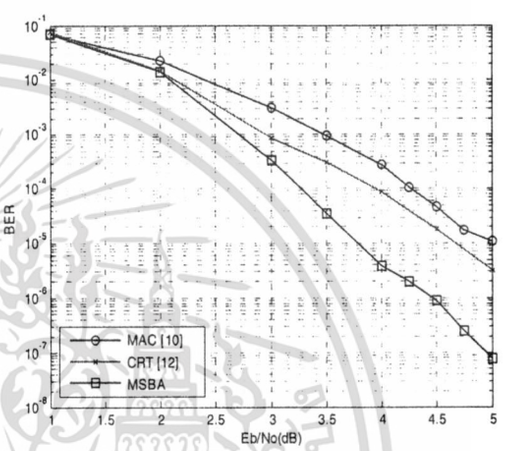


Fig. 10. Block length of about 500 at rate of 0.7

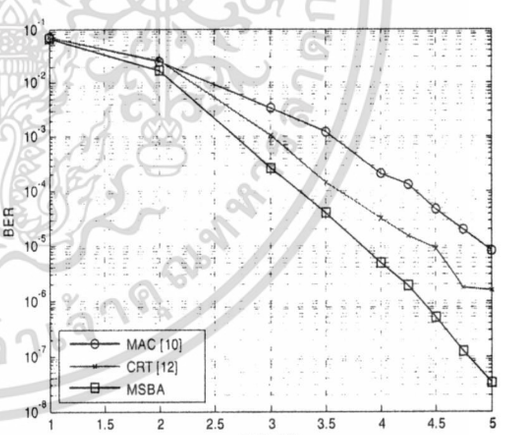


Fig. 11. Block length of about 1000 at rate of 0.7

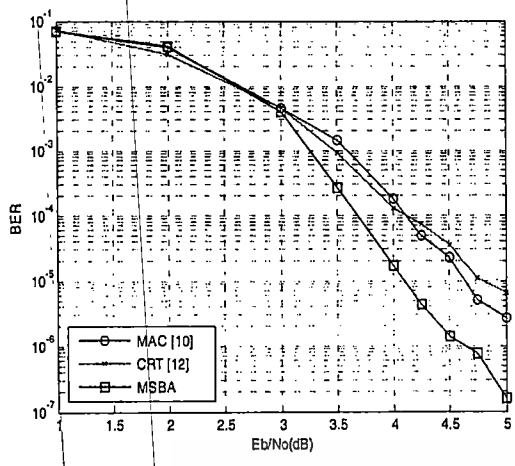


Fig. 12. Block length of about 1000 at rate of 0.8

V. CONCLUSIONS

We have proposed an alternative and new construction algorithm for a parity check matrix of irregular LDPC codes. The method is based on the known classic magic square algorithm. As a result, the complexity of matrix construction is fairly low. The obtained structured matrix suites well short block length codes with the rate of 0.7. The results show that BER of 10^{-7} at SNR of 5 dB can be achieved when the code is applied to AWGN channel. Moreover, to achieve such a performance the number of iteration used is also relatively low. It is possible that the similar construction could be utilized to design a matrix for longer block codes.

Table 13. Construction parameters of MAC, CRT and MSBA for the short block length codes

	MAC [6]			CRT [8]			MSBA		
J	3	3	3	3	3	3	3	3	3
K	9	10	15	9	10	15	9	10	15
P	59	101	67	57	102	68	57	101	67
$R = 1-(j/k)$	0.7	0.7	0.8	0.7	0.7	0.8	0.7	0.7	0.8
$b = p(k-j)$	354	707	804	342	714	816	342	707	804
$Parity = jp$	177	303	201	171	306	204	171	303	201
$c = kp$	531	1010	1005	513	1020	1020	513	1010	1005

REFERENCES

[1] R. Gallager, "Low-density Parity-check Code", *IRE Transaction Information Theory*, pp. 21-28, 1962.

[2] D. J. C. MacKay, "Good error-correcting codes based on very sparse matrices", *IEEE Transaction Information Theory*, vol. 45, No. 2, pp. 399-431, 1999.

[3] M. Blaum, P. Farrell, and H. van Tilborg, "Array codes, in Handbook of Coding Theory", V. S. Pless and W. C. Huffiman Eds., Elsevier 1998.

[4] S. Myung and K. Yang, "A combing method of quasi-cyclic LDPC codes by the Chinese Remainder Theorem", *IEEE Communications Letter*, vol. 9, pp. 823-825, 2005.

[5] M. G. Luby, M. Mitzenmacher, M.A. Shokrollahi, and D.A. Spielman, "Efficient erasure correcting codes", *IEEE Transaction Information Theory*, vol. 47, pp. 569-584, 2001.

[6] E. Eleftheriou and S.Olcer, "LDPC Codes for Digital Subscriber Lines", *Proceeding International Conference on Communication*, pp. 1752-1757, 2002.

[7] D. Abematsu, T. Ohtsuki, S. PW Jarot, and T. Kashima, "Size Compatible (SC)-Array LDPC Codes", *IEEE Vehicular Technology Conference*, pp.1147-1151, 2007.

[8] C. Chusin, C. Prasartkaew, S. Timakul and S. Choomchuay, "A Desing of Nonprime Block Irregular LDPC Codes via CRT", *ISCIT International Conference*, Japan, 2010.

[9] Weisstein, E. W. Magic Square [online]. Available from http://mathworld.olfram.com/MagicSquare.html, 2003.

[10] J. Fan, Y. Xiao, and K. Kim, "Design LDPC Codes without Cycles of Length 4 and 6", *Hindawi Publishing Corporation Research Letters in Communications*, 2008.

[11] R. Lucas, M. P. C. Fossorier, Y. Kou, and S. Lin, "Iterative decoding of one-step majority logic decodable codes based on belief propagation," *IEEE Transactions on Communications*, vol. 48, no. 6, pp. 931-937, 2000.

WP1-A-4

A Parity Check Matrix Design for Irregular LDPC Codes with 2K Block Length

Chutima Prasartkaew * and Somsak Choomchuay †

*College of Data Storage Technology and Applications, King Mongkut's Institute of Technology Ladkrabang, BKK, Thailand
Tel:/Fax: + 66-2-326-4731, E-mail: prasartkaew@yahoo.com

†Faculty of Engineering, King Mongkut's Institute of Technology Ladkrabang, BKK 10520, Thailand
Tel: +66-2-326-4222 Ext.114, Fax: +66-2-739-2398, E-mail: kchsomsa@kmitl.ac.th

***Abstract*— This paper outlines the work on another design of a parity check matrix for Irregular LDPC codes. The design is based on the pattern of Modified Array and Interleaved Modified Array LDPC codes. The application of matrix transposition Quasi-cyclic shifting has resulted in the reduction of 1's. The designed matrix is suitable for codes with short and medium block lengths. The code rate of 0.56 at the BER of 10^{-4} is obtained.**

I. Introduction

Error Correction Codes (ECC) is one of many tools made available for achieving consistent data transmission. It can improve bit error rate (BER) in alternative to increasing the signal to noise ratio (SNR). Two major type of ECCs known as block code and convolution codes have played their good roles in many applications, in particular modern communication and data storage technology. Low-Density Parity-Check codes (LDPC) are also linear block code that has been studied vastly in this decade. The main advantage of the codes is that they provide the performance at that very close to the capacity for a lot of different channels and linear time complex algorithms for decoding [12]. They also suit well the parallel realization.

In this paper we propose a modified method to obtain parity check matrices. The obtained result is comparable to a published work [1] but with a higher code rate. The rest of this paper is organized as follows: General perspective of LDPC codes is given in section II. Encoder and decoder are included. A design of parity check matrices is outline in section III and the corresponding performance tests are reported in section IV.

II. LDPC Codes

Of its discovery in 1960 by Gallager [2], the LDPC code has been ignored for some ten years. This was because the code itself is quite complex. In the same time the more highly-structured code; Reed Solomon code, was introduced [3]. The introduction of Turbo code by Berrou, Glavieux and Thitimajshima in 1993 [4] has drawn great attention since the code performance is close to Shannon limit. LDPC codes was recovered in 1998 by Rechardson and Urbanke [5] and in 1999 by Mackay and Neal [6]. Since then, LDPC became more popular and widely developed for wider area of applications including communications and data storage.

According to their parity check matrix, LDPC codes can be termed as, random parity check matrix LDPC and structured parity check matrix LDPC. A random parity check matrix LDPC can have better BER compared to its competitor. However, it hold more complex parity check matrix. Many works have focused on the efficient design of a structured one [8, 9, 10].

There are two different ways to represent LDPC codes; matrix representation and graphical representation. In the matrix representation, as it is named, LDPC codes hold small number of "1" in each row and column, i.e. $W_c << n$ and $W_r << m$ for a dimension $m \times n$ parity matrix. This can provide large minimum distance of the code. However such a circumstance results a large parity check matrix. In the graphical representation, Tanner graph [7] provides an efficient view of LDPC codes. There are m check nodes (c-nodes; number of parity bits) and n variable nodes (v-nodes; number of bits in a codeword).

LDPC codes are said to be regular if W_c is constant for every column, and $W_r = W_c(n/m)$. If **H** is low density but the number of "1" in each row or column are not constant, the code is said to be an irregular one.

A. Encoder

Similar to all other linear block codes, we have the relation;

$$\mathbf{C}_{(1\times n)}\mathbf{H}^{T}_{(n\times m)} = 0 \tag{1}$$

$$\mathbf{CH}^{T} = [\mathbf{B} \mid \mathbf{P}]\begin{bmatrix}\mathbf{H}_1^{T}\\ \mathbf{H}_2^{T}\end{bmatrix} = \mathbf{BH}_1^{T} + \mathbf{PH}_2^{T} = 0 \tag{2}$$

or

$$\mathbf{P} = \mathbf{BH}_1^{T} + \left(\mathbf{H}_2^{T}\right)^{-1} \tag{3}$$

Where **C** is a codeword matrix, and **H** is a parity check matrix. $\mathbf{H}_2$ is a parity check matrix of the dimension $(m \times m)$. It is a part of the matrix **H**.

978-1-4244-5016-9/09/$25.00 ©2009 IEEE

The task of the encoder is then to compute the matrix **P** that can be directly appended to the message to produce the codeword.

For the matrix **H** to be more manageable, LU decomposition method is preferably applied; i.e. $[\mathbf{H}]=[\mathbf{L}][\mathbf{U}]$. Thus,

$$\begin{bmatrix} l_{11} & 0 & \cdots & 0 \\ l_{21} & l_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ l_{n1} & l_{n2} & \cdots & l_{nn} \end{bmatrix} \underbrace{\begin{bmatrix} u_{11} & u_{12} & \cdots & u_{1n} \\ 0 & u_{22} & \cdots & u_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & u_{nn} \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \\ \vdots \\ p_n \end{bmatrix}}_{Y} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix} \quad (4)$$

Let $[\mathbf{Y}]=[\mathbf{U}][\mathbf{P}]$, then we can use forward substitution to solve $[\mathbf{L}][\mathbf{Y}]=[\mathbf{B}]$.

$$\begin{bmatrix} l_{11} & 0 & \cdots & 0 \\ l_{21} & l_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ l_{n1} & l_{n2} & \cdots & l_{nn} \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix} \quad (5)$$

Finally, use backward substitution to solve $[\mathbf{U}][\mathbf{P}]=[\mathbf{Y}]$. There we can get $\{p_i\}$ as need.

$$\begin{bmatrix} u_{11} & u_{12} & \cdots & u_{1n} \\ 0 & u_{22} & \cdots & u_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & u_{nn} \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \\ \vdots \\ p_n \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} \quad (6)$$

B. Decoder

The decoding algorithm used for LDPC codes was discovered independently and comes under different names. The most common are the belief propagation algorithm, the message passing algorithm and the sum-product algorithm.

Tanner graph is an intuitive way in understanding the LDPC decoder. The graph can be drawn directly from the **H** matrix as shown below:

$$\mathbf{H} = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix} \quad (7)$$

The graph contains m check nodes (number of parity bits) and n variable nodes (number of bits in a codeword). Check node f_i is connected to a variable node c_i if the element h_{ij} of **H** is a 1.

In the Log-domain Sum-Product algorithm, the message passes between check nodes and variable nodes. In each pass the log likelihood ratio is recorded for its probability of its likely symbol.

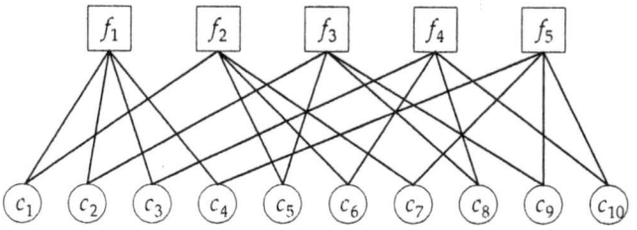


Fig. 1. Tanner graph of the H matrix given in (7)

III. Designs of Parity Check Matrices

The structure of **H** matrix can have great effect to an encoder, decoder and code performance. A design of the parity check matrix outlined in this paper is based on the modified array LDPC and interleave-modified array LDPC.

A. Some Previous Works

Fan [8] has introduced the array structure parity matrix that can offer comparable performance when compared to a random generated parity matrix. Other features are: low noise floor and no existence of cycle of 4. Fan's matrix is shown below.

$$\mathbf{H}(\mathrm{p,j,k}) \triangleq \begin{bmatrix} \mathbf{I} & \mathbf{I} & \mathbf{I} & \cdots & \mathbf{I} \\ \mathbf{I} & \boldsymbol{\alpha} & \boldsymbol{\alpha}^2 & \cdots & \boldsymbol{\alpha}^{k-1} \\ \mathbf{I} & \boldsymbol{\alpha}^2 & \boldsymbol{\alpha}^4 & \cdots & \boldsymbol{\alpha}^{2(k-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{I} & \boldsymbol{\alpha}^{j-1} & \boldsymbol{\alpha}^{2(j-1)} & \cdots & \boldsymbol{\alpha}^{(j-1)(k-1)} \end{bmatrix} \quad (8)$$

This yields the code rate of $R=1-\frac{pj-j+1}{p^2}$. Where **I** is an identity matrix $(p\times p)$, codeword is a $k\times p$ size, parity bit is a $j\times p$ size and $\boldsymbol{\alpha}$ is a permutation matrix $(p\times p)$ representing a single left or right cyclic shift. For example, for $p=5$, we can have **I**, $\boldsymbol{\alpha}$ and $\boldsymbol{\alpha}^2$ as shown below.

$$\mathbf{I} = \begin{bmatrix} 10000 \\ 01000 \\ 00100 \\ 00010 \\ 00001 \end{bmatrix}_{(5\times5)} \quad \boldsymbol{\alpha} = \begin{bmatrix} 01000 \\ 00100 \\ 00010 \\ 00001 \\ 10000 \end{bmatrix}_{(5\times5)} \quad \boldsymbol{\alpha}^2 = \begin{bmatrix} 00100 \\ 00010 \\ 00001 \\ 10000 \\ 01000 \end{bmatrix}_{(5\times5)}$$

Eleftheriou and Olcer [9] have proposed the modified array structure (MAC) by applying cyclic shift to Fan's array. This structure (shown in Eq. (9)) yields the code rate of $R=1-(j/k)$. MAC has superior performance to Fan's array that it can reduce the 1's in the lower triangle. The obtained matrix can also utilize a simple encoder. Efficient encoding is achieved from **H** without the need to compute the generator matrix of the code. As the upper triangular form of **H**, there are no cycles of length 4 in the corresponding Tanner graph.

$$\mathbf{H}=\begin{bmatrix} \mathbf{I} & \mathbf{I} & \cdots & \mathbf{I} & \mathbf{I} & \cdots & \cdots & \mathbf{I} \\ \mathbf{0} & \mathbf{I} & \boldsymbol{\alpha} & \cdots & \boldsymbol{\alpha}^{(j-2)} & \boldsymbol{\alpha}^{(j-1)} & \cdots & \boldsymbol{\alpha}^{(k-2)} \\ \mathbf{0} & \mathbf{0} & \mathbf{I} & \cdots & \boldsymbol{\alpha}^{2(j-3)} & \boldsymbol{\alpha}^{2(j-2)} & \cdots & \boldsymbol{\alpha}^{2(k-3)} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \cdots & \mathbf{I} & \boldsymbol{\alpha}^{(j-1)} & \cdots & \boldsymbol{\alpha}^{(j-1)(k-j)} \end{bmatrix} \quad (9)$$

Where $\mathbf{0}$ is the $p\times p$ null matrix.

Singhaudom *et. al.* [10] has proposed the interleaved modified array LDPC or IMAC by introducing the quasi-cyclic matrix into the cyclic shift of [9]. This interleaved LDPC of which the parity matrix given below in Eq. (10) is superior to Fan's array LDPC when the block length is particularly long. The parameter is the circular shift permutation matrix.

$$\mathbf{H}=\begin{bmatrix} \mathbf{I} & \mathbf{I} & \mathbf{I}\boldsymbol{\omega} & \mathbf{I}\boldsymbol{\omega}^2 & \mathbf{I}\boldsymbol{\omega}^3 & \cdots & \mathbf{I}\boldsymbol{\omega}^j \\ \mathbf{0} & \mathbf{I} & \boldsymbol{\alpha}\boldsymbol{\omega} & \boldsymbol{\alpha}^2\boldsymbol{\omega}^2 & \boldsymbol{\alpha}^3\boldsymbol{\omega}^3 & \cdots & \boldsymbol{\alpha}^{(k-2)}\boldsymbol{\omega}^j \\ \mathbf{0} & \mathbf{0} & \mathbf{I} & \boldsymbol{\alpha}^2\boldsymbol{\omega}^2 & \boldsymbol{\alpha}^4\boldsymbol{\omega}^3 & \cdots & \boldsymbol{\alpha}^{2(k-3)}\boldsymbol{\omega}^j \\ \vdots & \vdots & \vdots & \mathbf{I} & \boldsymbol{\alpha}^3\boldsymbol{\omega}^3 & \cdots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{I} & \cdots & \boldsymbol{\alpha}^{(j-1)}\boldsymbol{\omega}^{(k-j)} \end{bmatrix} \quad (10)$$

Where $\boldsymbol{\omega}$ is the $p\times p$ permutation matrix ($\boldsymbol{\omega}^n=\boldsymbol{\omega}$) representing a single left or right quasi-cyclic shift. For example, for $p=5$, we can have $\mathbf{I}, \boldsymbol{\omega}$ and $\boldsymbol{\omega}^n=\boldsymbol{\omega}$ as shown below.

$$\mathbf{I}=\begin{bmatrix} 10000 \\ 01000 \\ 00100 \\ 00010 \\ 00001 \end{bmatrix}_{(5\times5)} \qquad \boldsymbol{\omega}=\begin{bmatrix} 01000 \\ 00010 \\ 10000 \\ 00100 \\ 00001 \end{bmatrix}_{(5\times5)} \qquad \boldsymbol{\omega}^5=\boldsymbol{\omega}=\begin{bmatrix} 01000 \\ 00010 \\ 10000 \\ 00100 \\ 00001 \end{bmatrix}_{(5\times5)}$$

$$\boldsymbol{\alpha}\times\boldsymbol{\omega}=\begin{bmatrix} 01000 \\ 00100 \\ 00010 \\ 00001 \\ 10000 \end{bmatrix}_{(5\times5)}\times\begin{bmatrix} 01000 \\ 00010 \\ 10000 \\ 00100 \\ 00001 \end{bmatrix}_{(5\times5)}=\begin{bmatrix} 00010 \\ 10000 \\ 00100 \\ 00001 \\ 01000 \end{bmatrix}_{(5\times5)}$$

B. Our Construction

In our work, the structured parity check matrix was designed aiming at a fairly simple check matrix. To get the suitable arbitrary parity check matrices, a good approach is to avoid constructing $\mathbf{H}$ matrix at all [11].

In our construction, the first half of $\mathbf{H}$ was designed by taking the left-hand half of the IMAC. The second half is primarily formed by transposition of the first half. There are 4 possibilities for the next efforts; 1) the matrix in the second half was cyclically shifted, 2) the matrix in the second half was quasi-cyclically shifted, 3) the matrix in the second half was flipped horizontally and cyclically shifted is applied to both halves, and 4) the matrix in the second half was flipped horizontally and quasi-cyclically shifted is applied to both halves. Option 3) works well compared to others [13]. Further modification to option 3) can be done by applying quasi-cyclic shifting. The final obtained matrix is shown in Eq. (11) below.

$$\mathbf{H}=\begin{bmatrix} \mathbf{I} & \boldsymbol{\omega} & \boldsymbol{\omega}^2 & \boldsymbol{\omega}^3 & \cdots & \boldsymbol{\omega}^{j-1} & \boldsymbol{\omega}^j \\ \mathbf{0} & \mathbf{I} & \boldsymbol{\alpha}\boldsymbol{\omega} & \boldsymbol{\alpha}^2\boldsymbol{\omega}^2 & \cdots & \boldsymbol{\alpha}^{(k-2)}\boldsymbol{\omega}^j & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I} & \boldsymbol{\alpha}^2\boldsymbol{\omega}^2 & \cdots & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & \vdots & \mathbf{I} & \cdots & \cdots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{I} & \cdots & \mathbf{0} \end{bmatrix} \quad (11)$$

The matrix is also a triangular one with the dimension of $jp\times kp$. We still have the code rate of $R=1-(j/k)$ where p is prime number and $j\le k\le p$, $\mathbf{0}$ is the $p\times p$ null matrix, and $\boldsymbol{\omega}$ is the $p\times p$ quasi cyclic matrix.

IV. Performance Evaluation

Our preliminary investigation was focused to the short and medium block length. Parameters are also designed to maintain the code rate of greater than 0.5. Test parameters are given in table 1 and table 2 below. The proposed matrix offers considerably good result compared to the existing LDPC codes.

For high BER, it can be observed that the parameter p is big compared to the parameter j. Then the sets of parameters which gives the highest BER and require block lengths (of ~1,000 bits for short and ~2,000 bits for medium block lengths) was selected and simulated for performance evaluation.

TABLE 1: PARAMETERS FOR SHORT BLOCK TESTING

j	3	4	5
k	11	11	11
p	97	97	97
$R=1-(j/k)$	0.727	0.636	0.545
$b=p(k-j)$	776	679	582
$Parity=jp$	291	388	485
$c=kp$	1,067	1,067	1,067
Iterations	5, 10, 20		

TABLE 2: PARAMETERS FOR MEDIUM BLOCK TESTING

j	5	6	7
k	16	16	16
p	137	137	137
$R=1-(j/k)$	0.688	0.625	0.563
$b=p(k-j)$	1,507	1,370	1,233
$Parity=jp$	685	822	959
$c=kp$	2,192	2,192	2,192
Iterations	5, 10, 20		

We firstly compared our design to the existing regular design [1] of the similar code rate. The code rate proposed by Rakibul *et. al.* [1] is 0.5 whilst ours is 0.563. The obtained result is shown in Fig. 1. It is clear that our code offers better performance, in particular when SNR is less than 6.5 dB.

When compared to existing irregular MAC and IMAC, our design also offers better performance. This is in particular when SNR is less 9 dB as shown in Fig.2.

We also investigated number of iterations versus the performance obtained. Our LDPC codes were tested at 5, 10, and 20 iterations. At 5 iterations, BER of both short and medium block length are similar. BER can be improved when we increased the number of iterations to 10. When the number of iteration is of 20, only the BER of a medium block length can be slightly improved. One may note that the obtained BER is still not so low. This is quite common that one cannot get so low BER when the block length is not large.

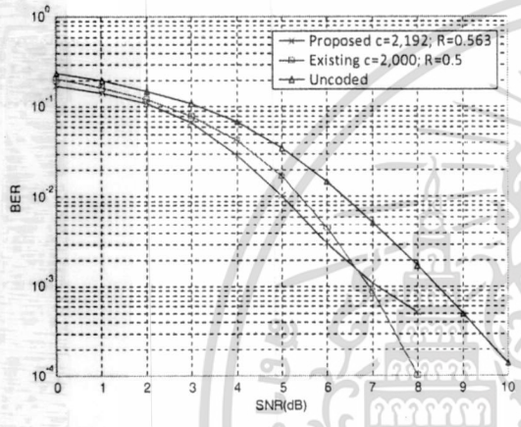


Fig. 2. Performance of our design compared to existing regular LDPC

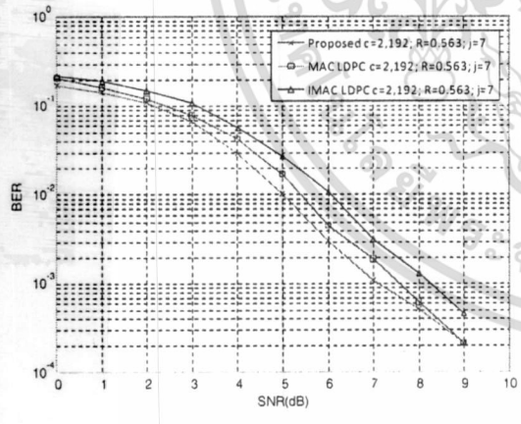


Fig. 3. The proposed LDPC and the existing LDPC's performance

V. CONCLUSIONS

We have proposed a design of parity check matrix for LDPC code based on the concept of array, modify array and interleaved modify array LDPC codes. The design is suitable for short and medium block length. The resulted design offers better performance compared to the regular design one. When comparing to the existing MAC and IMAC, our design also gives better performance. For the long block length, the obtained LDPC cannot compete the existing the interleaved modified array. Based on our idea, there could be opportunity to develop a parity check matrix on obtain the performance of IMAC for longer block length but with less complexity.

REFERENCES

[1] Mohammad Rakibul, Jinsang Kim, "On the use of QC-LDPC code for data transfer using short and medium block length," Conference ICACT 2009 Feb. 15-18, 2009.

[2] R. Gallager, "Low-density Parity-check Code," IRE Trans. Information Theory, Jan. 1962, pp.21-28.

[3] Reed, I. S. and Solomon, G., "Polynomial Codes Over Certain Finite Fields," *SIAM Journal of Applied Math.*, vol. 8, 1960, pp. 300-304.

[4] C Berrou, A. Glavieux and P. Thitimajshima, "Near Shannon limit error-correcting Coding and Decoding," proc. IEEE int. Conf., pp. 1064-1070, May 1993.

[5] T.J. Richardson, M.A. Shokrollahi, and R.L. Urbanke, "Design of Capacity-approaching Low-density Parity-check Codes," IEEE Trans. on Info. Theory, Vol. 47, pp. 619-637, Feb. 2001.

[6] D.J.C. Mackay and R. Neal, "Near Shannon Limit Performance of Low Density Parity Check Code," Electronics Letter, Vol.33, Mar 1997, pp.457-458.

[7] R.M. Tanner, "A Recursive Approach to Low Complexity Code," IEEE Trans. Information Theory, Sept. 1981, pp.533-547.

[8] J.L. Fan, "Array Codes as low-density parity-check codes," Proc. 2nd Int. Symp. Turbo Code, Beit, France, pp. 543-546, Sept. 2000.

[9] E. Eleftheriou and S.Olcer, "Low-density parity-Check Codes for Digital Subscriber Lines," Proc. 2002 Int. Conf. on Comm., pp. 1752-1757, April-May 2002.

[10] W. Singhaudom, S. Noppankeepong, P. Suphithi, "Design of Hign-Rate Modified Array Codes for Magnetic Recording System," ECTI International Conference, May 2007.

[11] O. Othman Khalifa, S. Khan, M. Zaid, and M. Nawawi, "Performance Evaluation of Low Density Parity Check Codes," International Journal of Computer Science and Engineering, 2008.

[12] W. E. Ryan, "An Introduction to LDPC Codes," *in CRC Handbook for Coding and Signal Processing for Recording Systems* (B. Vasic, ed.) CRC Press, 2004.

[13] Chutima Prasartkaew and Somsak Choomchuay, "A Design of Parity Check Matrix for Irregular LDPC Codes," ISCIT International Conference, Korea, Sept. 2009.

NON-PRIME PARAMETERS OF LDPC CODES WITH SYMMETRICAL SUB-MATRIX

Chalit Chusin[1] , Chutima Prasartkaew[2] and Somsak Choomchuay[3]

[1]College of Data Storage Technology and Applications, King Mongkut's Institute of Technology Ladkrabang, BKK, Thailand
Tel:/Fax: + 66-3-881-5966, E-mail: ithonene@hotmail.com
[2]College of Data Storage Technology and Applications, King Mongkut's Institute of Technology Ladkrabang, BKK, Thailand
Tel:/Fax: + 66-2-326-4731, E-mail: prasartkaew@yahoo.com
[3]Faculty of Engineering, King Mongkut's Institute of Technology Ladkrabang, BKK 10520, Thailand
Tel: +66-2-326-4222 Ext.114, Fax: +66-2-739-2398, E-mail: kchsomsa@kmitl.ac.th

ABSTRACT

The performance of Low-density parity-check (LDPC) codes is based on the parity check matrix design. In this paper, the irregular symmetrical parity check matrix was designed using of non-prime number. Then the arbitrary block length can be obtained. The advantages of symmetrical matrix are: 1) easily and quickly design, 2) the good performance as good as random construction was obtained and 3) higher minimum distance, the higher error detection and correction ability was obtained. Moreover, the performance of designed non-prime parity check matrix is higher when compared with MAC and SC-Array. Our designed parity check matrix still has the general properties of LDPC codes.

Index Terms— *LDPC Codes, non-prime, symmetrical, coding*

1. INTRODUCTION

A Low Density Parity Check (LDPC) codes is error correcting codes which has code performance close to Shannon limit. This code has many advantages such as low error floor, high code rate, no cycle-4 and capability of detecting and correcting burst error. In this decade, many researches are concerning the LDPC codes performance development. A design of parity check matrix is one challenge of those researches, since of the performance of LDPC codes depends on the parity check matrix design. In the parity check matrix design, the sparse one's should be decreased but must be enough for data error correction. Richardson *et al.* [1] presented the LDPC codes construction algorithm to yield the performance very close Shannon limit with a difference of only 0.0045 dB. It has very high performance when used for long block length. Eleftheriou and Olcer [2] and Singhaudom *et al.* [3] proposed a parity check matrix structure for LDPC codes, using 'Cyclic Shift' construction. Their code performances are equivalence to the random parity check matrix structure LDPC codes. The performance test results show that the longer block length, the higher performance yielded.

LDPC Codes hold two types of parity check matrix: regular and irregular. For regular parity check matrix, Gallager [4] purposed a construction of a random parity check matrix which has the same number of one element for every row and also has the same number of one element for every column, and without cycle-4. Fan [5] presented a design of an array parity check matrix structure. The complexity of parity check matrix construction is lower but still has the good performance as good as of the random parity check matrix. For irregular parity check matrix, because of its higher performance, when compared to a regular one, researchers are interesting on the later, e.g., [7], [2], [3] and [6]. In those works, the LDPC codes performances were improved by the variety of parity check matrix design. The complexity of the encoder can be made lower by using the triangular form matrices [2], [3] and [6].

In the design of parity check matrix of $jp \times kp$ sizes, the design parameters such as j, k and p should be suitably defined. Where j and k are integers, p is prime number and $j,k \leq p$. The traditional designs using of prime number parameter are presented in [4], [1], [5], [2], [3] and [6]. [4] has proposed the construction of regular LDPC codes with random parity check matrix and [1] have presented the evaluation of performance of this LDPC codes and the results shown that it performance close to the Shannon limit. [5] has proposed the array structure sub-matrix of parity check matrix with is modified from [4] and still be the regular LDPC codes for long block length. [2] have presented the modification of array structure sub-matrix from regular LDPC codes of [5] to be irregular using the concept of quasi-cyclic shift (α) into sub-matrix. [3] modified the parity check matrix from [2] by using interleave quasi-cyclic shift (ω). [6] have presented the parity check matrix for short and medium block lengths which is modified from [2] and [3].

For non-prime number sub-matrix constructions, this was firstly presented by Abematsu *et al.* [8] and followed by Chusin *et al.* [9]. Where, [8] modified the work proposed by [2] but with non-prime number construction parameters instead of prime number. Similarly [9] have modified the sub-matrix given by [3] and with non-prime number.

In this paper we propose a new design parity check matrix for arbitrary code length aims at improves the coding performance suitable for 4K bits block length using of symmetrical sub-matrix with non-prime number construction parameters. This paper is organized as follows: General perspective of LDPC codes is given in section II where

encoding and decoding are also included. Section III gives brief details of a symmetrical matrix. A design of parity check matrices for arbitrary and non-prime code lengths is outlined in section IV and the corresponding performance tests are reported in section V. Finally, the paper is concluded in section VI.

2. LDPC CODES

Low-density parity-check (LDPC) codes proposed by Gallager in 1962 [4], have attracted much attention given their good performance. The encoding and decoding of LDPC codes can be done by using of sparse parity check matrix H. The relationship between codeword bits and parity checks bit of the parity check matrix can be graphically presented by a Tanner graph [10]. A parity-check matrix H has n columns and m rows, and the codeword consists of n bits, which satisfy m checks, the number of message bits will be $k=m$, and the code rate is $R_c=k/n$. The number 1's in the parity check matrix in rows and columns represents an edge between the *i-th* bit node c_i and the *j-th* check node f_j.

For encoding of the LDPC codes, it is similar to other linear block codes, it has the relation;

$$C_{(1\times n)} = \left[m_{(1\times m)} \middle| p_{(1\times n-m)} \right] \tag{1}$$

where $p_{(1\times n-m)}$ denotes the parity portion, and $m_{(1\times m)}$ denotes the message portion respectively.

$$C_{(1\times n)} H^T{}_{(n\times m)} = 0 \tag{2}$$

where C is a codeword matrix, and H is a parity check matrix. In a systematic form, C can be written as:

There are several methods for decoding the LDPC codes e.g.: Believe Propagation (BP), Sum-Product (SP), and Message Passing (MP). The Log-domain Sum-Product algorithm was used in this paper; it is the message passes between check nodes and bit nodes. In each pass the log likelihood ratio (LLR) is recorded for is probability of its likely symbol. By means of this method, the variable q_{ij} be the message sent from the i^{th} bit node to j^{th} check node along a connecting edge, and r_{ji} is the message sent from j^{th} check node to the i^{th} bit node along a connection edge. The message q_{ij} is computed based on the values sent from check nodes connecting to the i^{th} bit node excluding the j^{th} check node.

3. SYMMETRICAL MATRIX

A symmetrical matrix is a square matrix that is equal to its transpose matrix. Let A is a symmetrical matrix. Then

$$A = A^T \tag{3}$$

The entries of a symmetrical matrix are symmetric with respect to the main diagonal (top left to bottom right). So if the entries are written as $A = \left(a_{ij}\right)$, then $a_{ij} = a_{ji}$. This also implies that

$$A^{-1}A^T = I, \tag{4}$$

where I is the identity matrix. The elements of a symmetric matrix A have the form:

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{12} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{1n} & a_{2n} & \cdots & a_{nn} \end{bmatrix} \tag{5}$$

The parity check matrix based on the symmetrical matrix can easily be constructed and yields a good performance as same as randomly parity check matrix. It can be seen that the '1' elements in the constructed symmetrical parity check matrix is fairly well distributed. As a result, the higher value of the minimum distance is obtained. Consequently, the better error correction can be achieved. The code performance should be as good as, or similar to the random construction presented in [7].

4. DESIGN OF PARITY CHECK MATRICES

In this section, we detail the formulation of the H matrix by using the idea of symmetrical matrix instead of permutation. The new design symmetrical sub matrix is used to form H matrix in the same manner of the forming of array code proposed by [8].

4.1 Some Related Works

Fan [5] has introduced the array structure parity matrix that can offer comparable performance when compared to a random generated parity matrix reported by Gallager [4]. Other features: low noise floor and no existence of cycle-4, are kept as the original features of LDPC codes proposed in [4]. Fan's matrix is shown below.

$$H(p,j,k) \triangleq \begin{bmatrix} I & I & I & \cdots & I \\ I & \alpha & \alpha^2 & \cdots & \alpha^{k-1} \\ I & \alpha^2 & \alpha^4 & \cdots & \alpha^{2(k-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ I & \alpha^{j-1} & \alpha^{2(j-1)} & \cdots & \alpha^{(j-1)(k-1)} \end{bmatrix} \tag{6}$$

Where I is an identity matrix $(p\times p)$,

α is a position permutation matrix $(p\times p)$.

This yields the code rate of $R = 1-\frac{pj-j+1}{p^2}$.

Eleftheriou and Olcer [2] have proposed the modified array codes (MAC) by applying cyclic shifting to Fan's array [5] given herewith in Eq. (7). The structure noted by Eq. (7) have the code rate of $R = 1-(j/k)$. MAC offers superior performance to Fan's array as it can reduce number of "1" in the lower triangle. This leads to simpler encoder while preserving other LDPC's features.

$$H=\begin{bmatrix} I & I & \cdots & I & I & \cdots & \cdots & I \\ 0 & I & \alpha & \cdots & \alpha^{(j-2)} & \alpha^{(j-1)} & \cdots & \alpha^{(k-2)} \\ 0 & 0 & I & \cdots & \alpha^{2(j-3)} & \alpha^{2(j-2)} & \cdots & \alpha^{2(k-3)} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \cdots & I & \alpha^{(j-1)} & \cdots & \alpha^{(j-1)(k-1)} \end{bmatrix} \quad (7)$$

where I is an identity matrix $(p\times p)$.

α is a position permutation matrix $(p\times p)$.

Singhaudom *et al.* [3] have proposed the interleaved modified array LDPC developed base on the concept of [2] and named it as IMAC. By introducing the quasi cyclic matrix into the cyclic shifting the obtained matrix is given in Eq. (8). The identity matrix and the interleave matrix are shown in Eq. (9). The IMAC is superior to MAC when the block length is particularly long.

$$H=\begin{bmatrix} I & I & I\omega & I\omega^2 & I\omega^3 & \cdots & I\omega^j \\ 0 & I & \alpha\omega & \alpha^2\omega^2 & \alpha^3\omega^3 & \cdots & \alpha^{(k-2)}\omega^j \\ 0 & 0 & I & \alpha^2\omega^2 & \alpha^4\omega^3 & \cdots & \alpha^{2(k-3)}\omega^j \\ \vdots & \vdots & \vdots & I & \alpha^3\omega^3 & \cdots & \vdots \\ 0 & 0 & \cdots & 0 & I & \cdots & \alpha^{(j-1)}\omega^{(k-j)} \end{bmatrix}_{(jp\times kp)} \quad (8)$$

Where ω is the quasi cyclic matrix that constructed from dentity matrix by cyclically-shifting of the matrix, I, i.e. $\omega^{n-1}=I$.

$$I=\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}_{(5\times5)} \text{ and } \omega=\begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}_{(5\times5)} \quad (9)$$

Abematsu *et al.* [8] have proposed the Size Compatible (SC)-Array LDPC Codes of which the parity check matrix shown in Eq. (10). The design can support arbitrary code lengths while achieving good error rate performance; it contains few or no cycle-4. It should be noted that [8] has used non-prime sub-block while [5] has. For the SC-array LDPC code, the permutation sub-matrix is decided to eliminate all cycle-4. Such a proposed cyclic shift $P_{sc}(j,k)$ is expressed by Eq. (11).

$$H=\begin{bmatrix} I & I & \cdots & \cdots & I & \cdots & I \\ \alpha^{P_{SC}(2,k)} & I & \cdots & \cdots & \alpha^{P_{SC}(2,j)} & \cdots & \alpha^{P_{SC}(2,k-1)} \\ \alpha^{P_{SC}(3,k)} & \alpha^{P_{SC}(3,k-1)} & I & \cdots & \alpha^{P_{SC}(3,j)} & \cdots & \alpha^{P_{SC}(3,k-2)} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \cdots & \vdots \\ \alpha^{P_{SC}(j,k)} & \alpha^{P_{SC}(j,k-1)} & \cdots & \cdots & I & \cdots & \alpha^{P_{SC}(j,k-j+1)} \end{bmatrix} \quad (10)$$

where

$$P_{sc}(j,k)=(j-1)(k-1)+\left|\frac{(j-1)(k-1)}{L}\right| \quad (11)$$

The IMAC cannot support the arbitrary code lengths, since the code length is restricted to be a multiple of prime number. Chusin *et al.* [9] have presented a design of parity check matrix to address this problem. Their design is based on the construction rows that don't have the same sub-matrix in the same row. They have shown that, with non-prime number, they achieved the same error rate performance as the prime sub-matrix size of IMAC in AWGN channels. The performance for long block lengths is also better when compared with [8].

4.2 LDPC Codes with Symmetrical Sub-matrix

We designed the H matrix by using of symmetrical matrix shown in Eq. (12) instead of permutation matrix. The symmetrical matrix S is of dimension $q\times q$ where q could be either prime or non-prime number. For the designed matrix size $q\times q$, let's define the binary element of matrix be $s_{xy}=\{0,1\}$ and the index r is bound to $r=(q/3)\times2$. x and y are row and column indices respectively.

In designing of S matrix, the row elements can be computed as given by a pseudo code below:

```
Odd number row:
    bb=0;  aa=0;  x=1;
For cc=1 to q              (row)
for dd=1 to q            (column)
    if  ((2*q)-dd)-(2*bb)=dd   then
s=1;  goto 10;
else
s=0
end;
      end;
10:    cc=cc+2;  bb=bb+1;
if   cc = ((2*q)-cc)-(2*bb)  then
   goto 20;
    end;
20; end
```

The odd number columns can be generated by the transposition of obtained odd number rows as $S_{y,x}=\left(S_{x,y}\right)^T$.

The construction or even number (2, 4, 6, ...) row and column of sub-matrix, S is to generate '1' at the $x=y$ position of each column. If there is no the '1', add this position with the '1', otherwise adding the '0'.

The resulted matrix is shown in Eq. (12);

$$S=\begin{bmatrix}
0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & s_{i,j} \\
0 & s_{2,2} & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & s_{(x+2),(y-1)} & 0 \\
0 & 0 & 0 & s_{4,4} & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 \\
 & & & & \ddots & \vdots & \vdots & \vdots & & \vdots & \vdots \\
\vdots & \vdots & \vdots & & \ddots & s_{r,r} & 0 & 0 & \vdots & \vdots & \vdots \\
 & & & & & 0 & 0 & s_{(r+2),(r-1)} & \vdots & \vdots & \vdots \\
 & & & & & 0 & s_{(r+1),(r-2)} & 0 & \vdots & \vdots & \vdots \\
\vdots & \vdots & \vdots & & & & & & \vdots & \vdots & \vdots \\
0 & 0 & s_{(y-1),(x+2)} & & & & & & \vdots & \vdots & \vdots \\
s_{y,x} & 0 & 0 & 0 & \cdots & \cdots & \cdots & \cdots & \cdots & 0 & 0
\end{bmatrix} \quad (12)$$

The example designed matrix of which $q=10$ is shown in Eq. (13).

$$S=\begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix} \quad (13)$$

As shown in Eq. (13), it can be seen that the matrix hold transpose property or $S^T = S$ and there is not cycles 4 in sub-matrix. We then construct one time shifting or S^1 by performing cyclically-right shifting of the original S. The obtained matrix is given in Eq. (14).

$$S^1=\begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix} \quad (14)$$

Other degree shifting can be performed similarly. The block shifting scheme is arranged similar to that proposed by [8]. Finally, the parity check matrix can be generated as given in Eq. (15). We use this formulation in code design for performance evaluation. In this design, three important parameters consist of j, k and L (j, k ≤ q). Where j and k are integers and q is non-prime number of sub-matrix size.

$$H=\begin{bmatrix}
I & I & I & I & \cdots & \cdots & I \\
0 & I & \lambda^{(2,3)} & \cdots & \lambda^{(2,j)} & \cdots & \lambda^{(2,k-1)} \\
0 & 0 & I & \lambda^{(3,4)} & \lambda^{(3,i)} & \cdots & \lambda^{(3,k-2)} \\
\vdots & \vdots & \vdots & \ddots & \vdots & \cdots & \vdots \\
0 & 0 & \cdots & \cdots & I & \cdots & \lambda^{(j,k-j+1)}
\end{bmatrix} \quad (15)$$

where

$\lambda^{(j,k)} = S^{P_T(j,k)}$ and,

$$P_T(j,k) = (j-1)(k-1) + \left|\frac{(j-1)(k-j)}{q}\right|$$

5. PERFORMANCE EVALUATION

The performance of our constructed parity check matrix was investigated for 4K bits blocks length. This is actually useful in the application of magnetic recording system where the sector size is 512 bytes. To compare with some exist publication the sub-matrix size of 68 is used. The test parameters are shown in Table I as below with iteration number of 30. The minimum distance of available published work was determined at d_{min}= 336 [2]. Our work has d_{min}= 341. This implies that our work should have higher capability of error detection and correction.

TABLE I: PARAMETERS FOR 4KBIT BLOCK SIZE

	j	k	q	R	Block size (bit)
MAC [2]	5	61	67	0.918	4087
SC-Array[8]	5	60	68	0.917	4080
This paper	5	60	68	0.917	4080

The design code is run for 30 iterations to the random-generated input sequence as also performed by MAC and SC-array. The obtained performance is shown in Fig. 1 below. The proposed code offers the same performance as MAC and SC-Array do. It only yields slightly better performance when E_b / N_0 is greater than 4.2 dB.

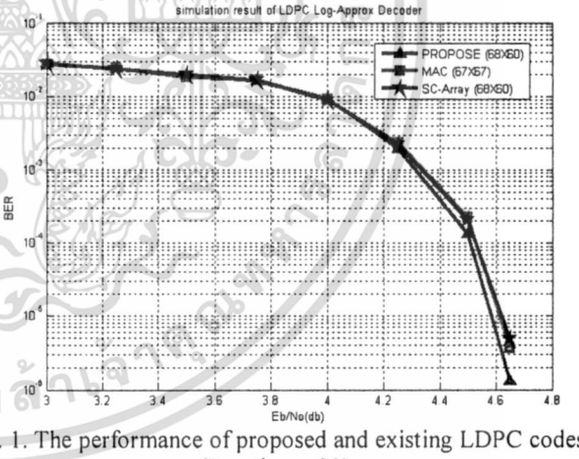


Fig. 1. The performance of proposed and existing LDPC codes (Iteration = 30)

The modified matrix proposed in this paper was also test for the block length of a particular size – say 4080, but at different sub-matrix size. This effect results in the changing of the code rate. We therefore intend for the code rate of better than 0.5. Parameters are given in Table II.

TABLE II: PERFORMANCE TEST PARAMETERS FOR EACH SUB-MATRIX SIZE

Block Size (bit)	j	k	q	R	Iterations
4080	5	60	68	0.917	30
4080	5	51	80	0.902	30
4048	5	44	92	0.886	30
4030	5	26	155	0.808	30
4063	5	17	239	0.706	30
4082	5	13	314	0.615	30
4100	5	10	410	0.5	30

Of the 30 iterations limit, the obtained result is shown in Fig. 2. It can be seen that the performance of the code can be improved when the sub-matrix size is increased as the sub-matrix size varies from 68 to 155. On the other hand, when the code rate is better than 0.8. However the decline in performance is observed when the sub-matrix size is more than 155. Therefore the good performance could be obtained only in particular range of sub-matrix size.

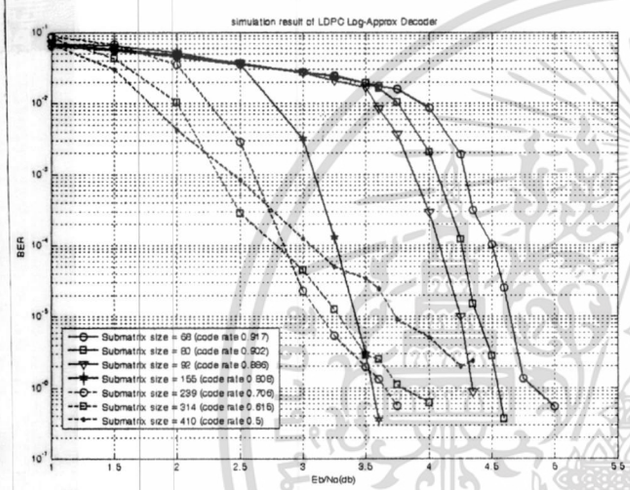


Fig. 2. The test results using different sub-matrix sizes (Iteration = 30)

6. CONCLUSIONS

The design of high-rate code with block sizes suitable for magnetic recording system is illustrated. The block size of the current application is about 4K bits. However, the actual size is determined based on particular data rate (Mbit/sec.) which practically optimized upon several parameters. To ease the design constrain we are trying to design a code that support any block size. The proposed parity check matrix is designed based on symmetrical property rather than permutation effort. Although the design code offers similar or slightly better performance compared to the published MAC [2] and SC-Array [8], the simulation shows that the size of sub-matrix affects the code performance. The larger sub-matrix leads to higher minimum distance and achieve the better performance than the smaller one. However, the sub-matrix size should less than 155, as discussed in the previous section. This proposed matrix still possesses suitable properties for magnetic recording system such as low-complexity encoding process, low error floor and capability of detecting and correcting burst errors.

In summary, the advantages of symmetrical matrix are: 1) easily and quickly design, 2) the good performance as good as random construction was obtained and 3) higher minimum distance, the higher error detection and correction ability was obtained. For future works, we plan to evaluate the performance to the 4Kbytes that using in the new generation of recording technology.

7. REFERENCES

[1] Chung, S.-Y., Forney, G. D., Jr. Richardson, T. J., and Urbanke, R. L., "On the design of low-density parity-check codes within 0.0045 dB of the shannon limit," *Electron. Lett.*, vol. 5, pp. 58-60, Feb. 2001.

[2] E. Eleftheriou and S. Olcer, "Low-density parity check codes for digital subscriber lines," *Proc. 2002 Int. Conf. on Comm.*, pp.1752-1757., April – May, 2002.

[3] W. Singhaudom, S. Noppankeepong, P. Suphithi, "Design of Hign-Rate Modified Array Codes for Magnetic Recording System," *ECTI International Conference*, May 2007.

[4] R. Gallager, "Low-density Parity-check Code," *IRE Trans. Information Theory*, Jan. 1962, pp.21-28.

[5] J. L. Fan, "Array codes as low-density parity-check codes," *Proc. 2nd Int. Symp. Turbo Codes*, France, Sep 2000, pp 543-546.

[6] C. Prasartkaew, S. Choomchuay, "A Parity Check Matrix Design for Irregular LDPC Codes with 2K Block Length," *ISPACS International Symposium on Intelligent Signal Processing and Communication Systems*, 2009.

[7] Richardson T.J., Shokrollahi MA, Urbanke R.L. "Design of capacity-approaching irregular low-density parity-check codes," *Information Theory IEEE Trans.* Volume 47, pp.619-637, Feb 2001.

[8] Abematsu, D.; Ohtsuki, T.; Jarot, S.P.W.; Kashima, T.," Size Compatible (SC)-Array LDPC Codes" *Vehicular Technology Conference*, 2007.

[9] C. Chusin, C. Prasartkaew, S. Choomchuay, , "A Design of Non-prime LDPC Based on Interleave Modified Array Codes," *Proceeding of DST-CON 2010*, Bangkok, 2010.

[10] R. M. Tanner, "A recursive approach to low complexity codes," *IEEE Trans. Information Theory*, pp.533-547, Sept.1981.

Simulation Design of IEEE 802.11n Block Compatible LDPC Codes via MSBA

Chutima Prasartkaew[1] and Somsak Choomchuay[2]

[1] Faculty of Science and Technology, Rajamangala University of Technology Thunyaburi, Thailand
[2] Faculty of Engineering, King Mongkut's Institute of Technology Ladkrabang, Thailand

Abstract—This paper details the design of IEEE 802.11n block compatible Low Desity Parity Check codes. The new algorithm, modified magic square algorithm, has been investigated. The codes has been designed using simulation for the block length of 648 bits whilst the code rate are made vary from 1/2 to 5/6. The result obtained form AWGN channel simulation has shown the comparative performace when comapred to the standard matrix.

Keywords—Irregular LDPC Codes, Magic Square Technique, IEEE802.11n

1 Introduction

The Low Density Parity Check (or LDPC) codes is a linear block code defined by a spare parity check matrix. It is firstly proposed by Gallager in 1960 [1]. At that time this codes could not draw much attention according to its complexity and in the similar period the more intuitive codes, RS code, was also introduced. The good performance of the later invented code, turbo code, has drawn great attention and has been applied to many applications. However, there is a restriction of the Turbo code since it has been patented. The original designed LDPC codes was re-tested by MacKey [2, 3]. The obtained results show that the code performance can be made close to Shannon's limit on AWGN and binary erasure channels. With that notes, many researchers were interested in LDPC codes. The parity check matrix of the originally Gallager code is randomly generated and the number of '1' in each row and column of sparse matrix must be constant and the same. This is generally known as a regular LDPC code.

Array codes [4] are error-correcting codes that have the capability to correct error bursts using an algebraic decoder, but can also be fully incorporated into soft iterative decoding schemes with an algebraic decoder. The symbols for array codes can be very large, making them especially suitable for correcting long burst of errors. Fan [5] proposed a construction of structured parity check matrix, the obtained code is known as array LDPC codes. Since the construction complexity is lower than that of the random construction; the array LDPC code can be possibly implemented with the fairly simple coding devices.

Toward the irregular structure matrix where the number of "1" in each row or column are not equal, Eleftheriou [6] proposed irregular array LDPC codes for an application in Digital Subscriber Lines (DSL). The construction is based on array codes given by Fan. To achieve efficient encoding, a parity-check matrix in triangular form is desirable. Although Gaussian elimination could be used to this end, the attempt increases the processing complexity and makes this approach less attractive. Alternatively, we can define a new matrix **H** by cyclically shifting the rows of the previous matrix **H** in a block-wise manner. To obtain the parity-check matrix in the desirable form, the lower-triangular elements of the new matrix **H** are still set to zero. Array LDPC codes are high-rate codes that can achieve good error rate performance. However, they do not support arbitrary code lengths since the code length of an array LDPC code with good error rate performance is limited to a multiple of a prime number. Recognizing this problem, Abemastu [7] proposed a Size Compatible (SC)-array LDPC codes. They have achieved good error rate performance while supporting arbitrary code lengths. Chusin [8], later on proposed the arbitrary length comparable irregular LDPC codes similar to SC-Array via CRT method and used non-prime number as a construction parameter. With such a proposed non-prime number construction parameter, the variety of block length and rate of codes can be designed with higher flexibility.

Toward the area of application, LDPC codes are now widely used in both data storage technology and communications. In the area of communication LDPC codes are now included in many standards such as DVB-S2 and DVB-T2 [9] for digital video broadcasting, Wireless Local Area Networks (WiFi) (IEEE 802.11n) [10], Wireless Metropolitan Area Networks (WiMAX) (802.16e) [11] and Wireless Regional Area Networks (WRAM) (IEEE 802.22) for wireless networks. Although IEEE802.11n has defined matrix prototype of both n=648 (z=27) and n=1296 (z=54), and with the code rate of 1/2, 2/3, 3/4, and 5/6, in this paper we have considered the length = 648 bits only. The similar idea can be easily extended to the block length of 1296 bits.

As the matter of fact, the constructed sparse parity check matrix affects the code performance substantially. In the parity check matrix construction, either random or structured algorithm can be used. Most previous methods for designing LDPC codes are based on random construction techniques. The lack of structure implied by this randomness presents serious disadvantages in term of storing and accessing a large parity-check matrix, encoding data, and analyzing code performance. Encoder and decoder implementation of long length linear block codes is difficult in practice. Moreover, due to the limited amount of memory, the structured algorithm is more attractive. The rest of this paper is organized as follows: magic square technique is briefly reviewed in section 2. Construction details of the parity check matrices are given in section 3. The simulation procedure and results are reported in section 4. Finally, this paper is concluded in section 5.

2 A Magic Square Technique

In the previous, many algorithm and methods used in matrix formulation have been mentioned. Many of them are quite complicate. In this paper we demonstrate an application of a classic simple method to construct a matrix; a magic square technique. A Magic Squares theorem is a specific way to construct a square array of numbers. It has been studied for at least three thousand years as the earliest recorded appearance dating to about 2200 BC, in China. In the 9th century, Arab astrologers used this theory in calculating horoscopes. A magic square is a square array of the numbers 1, 2, 3, ..., n^2, with the property that the summation of every row, column and both diagonals, is the same number. Since there are n rows, the summation of all the numbers in the magic square must be $n \times M$, where, M is the number that each row, column and diagonal must add up to. In summation notation;

$$\sum_{i=1}^{n^2} i = n \times M \qquad (1)$$

Then solving for M gives $M = [n(n^2+1)]/2$. For example, a 3×3 magic square must have its rows, columns and diagonals adding to 15.

The existing magic squares can be classified into four groups: 1) magic squares associated to the astrological planets, 2) odd order, 3) doubly even order (n divisible by four), and 4) singly even order (n is even, but not divisible by four). The first group, consists of only seven magic square (n = 3 to 9), where each magic square associated with one of the astrological planets. The next two groups can be simply generated and each usually has only one magic square. The forth group, singly even order, magic squares is more difficult to be generated. However the

existing construction methods: e.g.: Ralph Strachey and LUX methods work well.

In this paper we consider a modification to the second and the forth groups. This is because we are considering that these groups should offer us more random-like set of number. The detail design is given in the following section.

2.1 Odd order

The Siamese or staircase method is derived by De la Loubère [15]. Starting from the central column of the first row with the number 1, the fundamental movement for filling the cells is diagonally up and right, one step at a time. If a filled cell is encountered, the next move should then be vertically down one cell, before continuing as before. When a move leaves the square, it is wrapped around to the last row or first column, respectively (as shown in Fig. 1).

30	39	48	1	10	19	28
38	47	7	9	18	27	29
46	6	8	17	26	35	37
5	14	16	25	34	36	45
13	15	24	33	42	44	4
21	23	32	41	43	3	12
22	31	40	49	2	11	20

Fig. 1: Odd order magic square (Siamese method)

2.2 Singly even order

The construction of singly even (or $n = 4p + 2$, where p is an integer) magic squares are more difficult than odd and doubly even orders. There are few existing construction methods: Ralph Strachey and LUX methods.

In the construction method of Ralph Strachey, the magic square is divided into equal quarters (as shown in Fig. 2). For example, in a 6×6 square, this will give four 3×3 squares. Each of these smaller squares can be formed using De la Loubère's method for odd order squares.

8	1	6	26	19	24
3	5	7	21	23	25
4	9	2	22	27	20
35	28	33	17	10	15
30	32	34	12	14	16
31	36	29	13	18	11

Fig. 2: Singly even order magic square (Strachey method)

A method for generating singly even magic squares was found by J. H. Conway and is called the LUX method. The shapes of the letters L, U, and X are naturally used for the filling order, hence the name of the algorithm. Create an array consisting of $m+1$ rows of Ls, 1 row of Us, and $m-1$ rows of Xs, all are of length $n/2 = 2m+1$. The sub-operation is completed by interchanging the middle U with the L above it. Next generate the magic square of order $2m+1$ using the Siamese method centered on the array of letters (starting at the center location of the top row), but fill each set of four squares surrounding a letter sequentially according to the order prescribed by the letter.

3 Parity Check Matrix Construction

The objective of our design is to construct a parity check matrix in the form of modified array. As such, the top row contain identity matrices of the size $p \times p$, the lower-left corner of this parity check matrix contain zero matrices of the same size. *I* denotes the identity matrix of the size $p \times p$ and the number represents the order of shifts applied to *I*. As the block length is defined as 648, it is fairly easy to seek for k and p. p will determine the magic square size while $R = 1-(j/k)$ defines the code rate. The obtained magic square has been modified slightly before using its member to shift the identity matrix. The obtained matrices are given below.

B=648, R=0.50, j=4, k=8, p=81

I	I	I	I	I	I	I	I
0	I	1	10	19	28	38	47
0	0	I	40	39	37	36	35
0	0	0	I	7	9	29	46

B=648, R=0.7, j=4, k=12, p=54

I	I	I	I	I	I	I	I	I	I	I	I
0	I	1	10	19	28	38	47	7	9	29	46
0	0	I	27	26	25	24	23	22	21	20	18
0	0	0	I	6	8	17	35	37	5	14	16

B=648, R=0.75, j=3, k=12, p=54

I	I	I	I	I	I	I	I	I	I	I	I
0	I	1	10	19	28	38	47	7	9	29	46
0	0	I	27	26	25	24	23	22	21	20	18

B=648, R=0.83, j=3, k=18, p=36

I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I
0	I	1	6	26	19	24	3	5	7	21	23	25	4	9	2	22	27
0	0	I	18	17	16	14	13	12	11	10	9	8	7	5	4	3	2

In this paper, the parity matrix is constructed using the different numbers in a magic square as cyclic shifting orders of each sub-permutation. The construction procedures are as shown in Fig. 3.

This method aims to construct the parity check matrix using some of the numbers of the magic square (as shown in Eq. (2)). All are independent from each other and the rest of the numbers are arranged to ensure the total normal curve distribution property of all the used numbers. The matrix structure constructed using the pattern shown in Eq. (3).

The algorithm presented in this paper can be conducted for the parity check matrix construction with required block length and code. However, with this algorithm, sub-block size (p) is different from those recommended by IEEE. The main reason is to avoid the presence of cycle-4. .

Given here as examples, to construct the parity check matrix with block length of 1296 bits and 1944 bits, the resulted constructions are shown in Table 1 and Table 2 respectively.

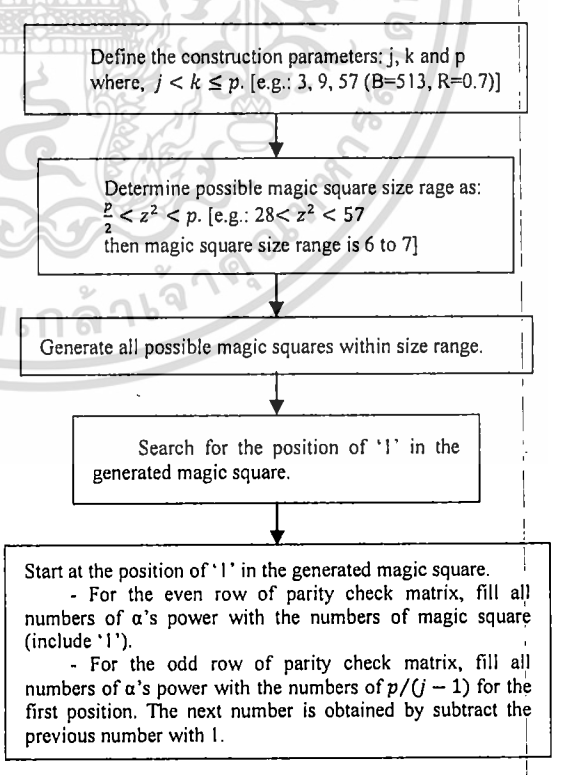


Fig. 3: Construction steps

$$MS = \begin{bmatrix} (u,v) & (u,v+1) & (u,v+2) & \cdots & (u,z-1) & (u,z) \\ (u+1,v) & (u+1,v+1) & (u+1,v+2) & \cdots & (u+1,z-1) & (u+1,z) \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ (z-1,v) & (z-1,v+1) & (z-1,v+2) & \cdots & (z-1,z-1) & (z-1,z) \\ (z,v) & (z,v+1) & (z,v+2) & \cdots & (z,z-1) & (z,z) \end{bmatrix}_{(z\times z)} \quad (2)$$

$$H = \begin{bmatrix} I & I & I & \cdots & I & \cdots & \cdots & I & I & \cdots \\ 0 & I & \alpha^{(u,v)} & \alpha^{(u,v+1)} & \alpha^{(u,v+2)} & \cdots & \alpha^{(u+i,z-1)} & \alpha^{(u+i,z)} & \alpha^{(u+i+1,v)} & \cdots \\ 0 & 0 & I & \alpha^{(p/(j-1))} & \alpha^{(p/(j-1))-1} & \cdots & \alpha^{(p/(j-1))-i} & \cdots & \cdots & \cdots \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & \cdots & I & \cdots & \cdots & \cdots & \cdots & \cdots \end{bmatrix}_{(j\times k)} \quad (3)$$

where, (u, v) is the position of '1' in the generated magic square.

Table 1: Construction parameters for block length of 648

Rate	j	k	p
1/2	4	8	81
2/3	4	12	54
3/4	3	12	54
5/6	3	18	36

Table 1: Construction parameters for block length of 1296

Rate	j	k	p
1/2	6	12	108
2/3	4	12	108
3/4	3	12	108
5/6	3	18	72

Table 2: Construction parameters for block length of 1944

Rate	j	k	p
1/2	6	12	162
2/3	4	12	162
3/4	3	12	162
5/6	3	18	108

4 Simulation Results

Based on AWGN channel, the designed codes detailed in the previous section were simulated with the block errors of 50. The decoder runs for 30 iterations with maximum E_b/N_o of 5 dB. The block length and code rate consistent with IEEE802.11n standard were taken into consideration. The sub-permutation matrix with the sub-block sizes of 54 and 81 bits were used in this study. However, as sub-block size of 27 bits (generally used for codeword block length n = 648 bits), the parity check matrix can not constructed using this proposed algorithm. Then the sub-permutation matrix with the sub-block sizes of 54 and 81 bits were used instead. The result is shown in the graph below.

At all code rate we can obtain bit error rate (BER) lower than 5×10^{-5} and in particular we can obtain BER of 2×10^{-6} for the rate of 1/2.

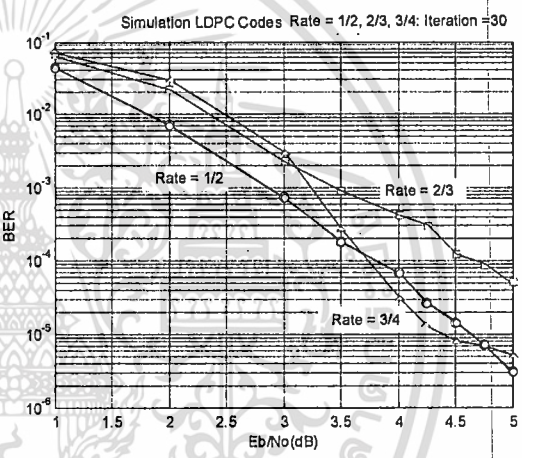


Fig. 4: Simulation of LDPC Codes designed with MSBA (Block length = 648 bits)

5 Conclusion

This paper has demonstrated the use of magic square based algorithm (MSBA) in designing the LDPC codes using simulation with the block length of 648 bits, sub-block sizes are 54 and 81. This block length is also recommended in IEEE802.11n wireless communication. Although the in dept analysis is superciliously touched, the current obtained result has shown reasonable good performance compared to standard matrix denoted in the IEEE802.11n document. Some advantages could be clearly observed when E_b/N_o is lower than 10^{-3} (noisy environment). We hope the MSBA could be further modified to get better result. Although an application of the algorithm to the block length of 1296 and 1944 bits are not given in details, the similar procedure can be obviously investigated. However, we have not yet investigated the hardware complexity of the new matrix in comparison with the standard matrix.

References

[1] R. Gallager, Low-density Parity-check Code, IRE Transaction Information Theory, (1962), pp. 21-28.

[2] D. J. C. Mackay and R. Neal, Near Shannon Limit Performance of Low Density Parity Check Code, Electronics Letter 33, (1997), pp. 457-458.

[3] D. J. C. MacKay, Good error-correcting codes based on very sparse matrices, IEEE Transaction Information Theory 45 (2) (1999), pp. 399-431.

[4] M. Blaum, P. Farrell, and H. van Tilborg, Array codes, in Handbook of Coding Theory, V. S. Pless and W. C. Huffiman Eds., Elsevier 1998.

[5] J. L. Fan, Array Codes as Low-Density Parity-Check Codes, Proceeding 2nd International Symposium TB & Relate Topics, (2000), pp. 543-546.

[6] E. Eleftheriou and S. Olcer, "Low-density parity check codes for digital subscriber lines," Proc. 2002 Int. Conf. on Comm., pp. 1752-1757., April-May,2002.

[7] D. Abematsu, T. Ohtsuki, S. PW Jarot, and T. Kashima, Size Compatible (SC)-Array LDPC Codes, IEEE Vehicular Tech. Conf. (2007), pp.1147-1151.

[8] C. Chusin, C. Prasartkaew, S. Timakul and S. Choomchuay, A Design of Nonprime Block Irregular LDPC Codes via CRT, ISCIT Inter. Conf., Japan, (2010)

[9] "Frame structure channel coding and modulation for the second generation digital terrestrial television broadcasting system (DVB-T2)," *DVB* Document A122, 2008.

[10] Draft STANDARD for Information Technology – Telecommunications and information exchange between systems - Local and metropolitan area networks - Specific requirements-, IEEE P802.11n./D3.00, Sept. 2007.

[11] "Air interface for fixed and mobile broadband wireless access systems," in *P802.16e/D12 Draft*, (Washington, DC, USA), pp. 100–105, IEEE, 2005

INTERNATIONAL JOURNAL OF ELECTRONICS AND COMMUNICATION ENGINEERING & TECHNOLOGY (IJECET)

ISSN 0976 – 6464(Print)
ISSN 0976 – 6472(Online)
Volume 4, Issue 1, January- February (2013), pp. 146-160
: www.iaeme.com/ijecet.asp
Journal Impact Factor (2012): 3.5930 (Calculated by GISI)
www.jifactor.com

IJECET
©IAEME

A DESIGN OF PARITY CHECK MATRIX FOR SHORT IRREGULAR LDPC CODES VIA MAGIC SQUARE BASED ALGORITHM

Chutima Prasartkaew[1] and Somsak Choomchuay[2]

[1]College of Data Storage Technology and Applications, King Mongkut's Institute of Technology Ladkrabang, Bangkok 10520, Thailand, E-mail: prasartkaew@yahoo.com
[2]Faculty of Engineering, King Mongkut's Institute of Technology Ladkrabang, Bangkok 10520, Thailand, E-mail: kchsomsa@kmitl.ac.th

ABSTRACT

This paper presents a construction algorithm for the short block irregular low-density parity-check (LDPC) codes. By applying a magic square theorem as a part of the matrix construction, a newly developed algorithm, the so-called Magic Square Based Algorithm (MSBA), is obtained. The modified array codes are focused on in this study since the reduction of 1s can lead to simple encoding and decoding schemes. Simulation results based on AWGN channels show that with the code rate of 0.8 and SNR 5 dB, the BER of 10^{-7} can be obtained whilst the number of decoding iteration is relatively low.

Keywords: Irregular LDPC codes, Magic square, Short block length, Parity check matrix

I. INTRODUCTION

The LDPC codes are linear block codes defined by a sparse parity check matrix. In the originally proposed version by [1], the matrix elements are randomly generated with the desirable condition that the number of 1s in each row and column must be the same and there should be no existence of cycle-4. This is known as a regular structure LDPC code. Twenty years later, [2] re-considered Gallager's LDPC codes and proposed his own notation of using a bipartite graph, which is sometimes called a Tanner graph. The originally designed code was also re-tested by [3]. The results show that the code performance can be made close to Shannon's limit. [4] investigated the performance of the code when it is applied to an erasure channel and pointed out that the decoding complexity is linear when using Sum-product Algorithm (SPA). Therefore, there is a constraint for codes with long block length.

International Journal of Electronics and Communication Engineering & Technology (IJECET), ISSN 0976 – 6464(Print), ISSN 0976 – 6472(Online) Volume 4, Issue 1, January- February (2013), © IAEME

Array codes are error-correcting codes that have the capability to correct error bursts. [5] mentions that the soft iterative decoding scheme can be applied to array codes. [6] proposes a construction of a structured parity check matrix, and the obtained code is known as array LDPC codes. Since the matrix construction is more structured compared to that of the random-generated method, the array LDPC codes can be possibly implemented with simple coding devices.

More recently, [7] investigated the construction of LDPC codes using circulant permutation matrices and pointed out that the obtained codes with a girth of larger than 12 cannot be represented by a Tanner graph. He derived the simple, necessary and sufficient conditions for the Tanner graph of the Quasi-Cyclic (QC) LDPC codes. In particular cases of girth g = 6, 8, 10 and 12, the condition is fairly easy for the QC-LDPC code construction. The resulting code performs significantly better compared to [1], when iteratively decoded with the Belief Propagation Algorithm (BPA). Later, [8] proposed a construction method for long QC-LDPC codes by combining several short QC-LDPC codes via the Chinese Remainder Theorem (CRT) where prime numbers are selected as a construction parameter. The girth of the obtained QC-LDPC code is always larger than or equal to that of each component code. The performances of the obtained codes constructed in this way are similar to the random regular ones.

While most previous studies [1-8] have focused on regular structure, [9] improved the performance of the code by introducing an irregular matrix. [10] proposed irregular array LDPC codes for an application in Digital Subscriber Lines (DSL). Their construction is based on the regular structure "array LDPC codes" proposed by [6]. For the encoder to be efficiently realized, a parity check matrix in triangular form is desirable. However, the designed codes do not support arbitrary lengths, since the block length is limited to a multiple of a prime number. To soften the problem, [11] proposed Size Compatible (SC) array LDPC codes. They propose the new cyclic shift instead of circulant permutation matrices. In order to eliminate cycle-4, they have permuted a sub-matrix position based on row number. Subsequently, [12] proposed the arbitrary length comparable irregular structure which is similar to the SC-Array of [11]. Their formulation is also based on the CRT method proposed by [8]; however, the non-prime number parameters are used. Regardless of the construction complexity, the obtained code yields good design flexibility. The code performance deteriorates when it is applied to short blocks.In additional, [13] proposed some sets of combinatorial design parameters of new class of LDPC codes based on idempotent and symmetric Latin squares. Theirresults showed that, under iterative decoding, their constructed code shows almost no performance loss, compared with MacKay's random code [14].

This research aims to reduce the construction complexity, compared to [12], and to achieve a better performance of short irregular LDPC codes. Based on the modified array codes proposed by [10], together with the use of non-prime parameters as with [11], this paper alternatively proposes an algorithm based on the magic square theorem. As a matter of fact, the structure of the parity check matrix contains many sub-permutation matrices and they should be cyclically shifted. With this regard, there is a chance for the magic square to be useful. Rather than random generation, the shifting orders are implicitly generated by a known procedure of magic square generation.

The rest of this paper is organized as follows: magic square theorem is briefly reviewed in section II. Section III presents the details about the coding theory background of LDPC codes, Some necessary conditions for the parity check matrix in order that the magic square could be incorporated are elaborated on in section IV. New construction details of the parity check matrices are given in section V. The simulation procedure and results are reported in section VI. Finally, this paper is concluded in section VII.

International Journal of Electronics and Communication Engineering & Technology (IJECET), ISSN 0976 – 6464(Print), ISSN 0976 – 6472(Online) Volume 4, Issue 1, January- February (2013), © IAEME

II. MAGIC SQUARE

A magic square is a square array of the numbers 1, 2, 3, ..., z^2, with the property that the summation of every row, column and both diagonals, is the same number. Since there are z rows and z columns, the summation of all the numbers in the magic square must be $z \times M$, where M is the number that each row, column and diagonal must add up to. In summation notation:

$$\sum_{i=1}^{z^2} i = z \times M \qquad (1)$$

Then solving for M gives $M = [z(z^2 + 1)]/2$ [15]. According to how they are generated, the existing magic squares can be classified into four groups: 1) magic squares associated to the astrological planets, 2) odd order, 3) doubly even order, and 4) singly even order.

1. Astrological Planets

In about 1510, Heinrich Cornelius Agrippa wrote De OccultaPhilosophia, drawing on the Hermetic and magical works of MarsilioFicino and Pico dellaMirandola, and in it he expounded on the magical virtues of seven magical squares of orders 3 to 9, each associated with one of the astrological planets: Saturn, Jupiter, Mars, Sol, Venus, Mercury, and Luna [16].

2. Odd Order

The Siamese or staircase method is derived by De la Loubère [17]. Starting from the central column of the first row with the number 1, the fundamental movement for filling the cells is diagonally up and right, one step at a time. If a filled cell is encountered, the next move should then be vertically down one cell, before continuing as before. When a move leaves the square, it is wrapped around to the last row or first column, respectively.

3. Doubly Even Order (z is divisible by four)

All the numbers are written in order from left to right across each row in turn, starting from the top left corner. Numbers are then either retained in the same place or interchanged with their diametrically opposite numbers in a certain regular pattern.Figure 1 shows the examples of doubly even order magic squares generatedas: a) writing numbers from left to right and b) interchanged with their diametrically opposite numbers in a certain regular pattern.

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16

a)

16	2	3	13
5	11	10	8
9	7	6	12
4	14	15	1

b)

Fig. 1. Examples of doubly even order magic squares.

International Journal of Electronics and Communication Engineering & Technology (IJECET), ISSN 0976 – 6464(Print), ISSN 0976 – 6472(Online) Volume 4, Issue 1, January- February (2013), © IAEME

4. Singly Even Order (z is not divisible by four)

There are two existing construction methods: Ralph Strachey and LUX methods. In the Ralph Strachey method, the magic square is divided into equal quarters. For example, a 6 × 6 square will give four 3 × 3 squares. Each of these can then be formed using De la Loubère's method for odd order squares [17]. Another method for generating singly even was found by J. H. Conway and is called the LUX method. The shapes of the letters L, U, and X naturally are used for the filling order; hence the name of the algorithm [15].

III. SOME NECESSARY CONDITIONS

This section describes briefly the matrix construction of LDPC codes. The theories of variation measurement as the criterion of the parity check matrix construction are also discussed.

1. The Parity Check Matrix

LDPC codes with jp × kpparity check matrix are basically specified by three parameters: j, k and p where, $j < k \leq p$. The matrix structure of irregular LDPC codes used in this study is shown in (2). This structure yields the code rate of $R = 1 - (j/k)$. The obtained matrix can also utilize a simple encoder.

$$\mathbf{H} = \begin{bmatrix} \mathbf{I} & \mathbf{I} & \cdots & \mathbf{I} & \mathbf{I} & \cdots & \cdots & \mathbf{I} \\ \mathbf{0} & \mathbf{I} & \boldsymbol{\alpha} & \cdots & \boldsymbol{\alpha}^{(j-2)} & \boldsymbol{\alpha}^{(j-1)} & \cdots & \boldsymbol{\alpha}^{(k-2)} \\ \mathbf{0} & \mathbf{0} & \mathbf{I} & \cdots & \boldsymbol{\alpha}^{2(j-3)} & \boldsymbol{\alpha}^{2(j-2)} & \cdots & \boldsymbol{\alpha}^{2(k-3)} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \cdots & \mathbf{I} & \boldsymbol{\alpha}^{(j-1)} & \cdots & \boldsymbol{\alpha}^{(j-1)(k-j)} \end{bmatrix} \quad (2)$$

where,$\mathbf{0}$ is null matrix,$\mathbf{I}$ is the identity matrix and$\boldsymbol{\alpha}$ is the permutation matrix, and all are p × p matrices. The power of α represents the order of cyclic shifting.

The systematic codeword with the length of (k×p) bits may comprise of (k-j)×p message bits and (j×p) parity bits. Similar to all other linear block codes, encoding is achieved by using the relation of $\mathbf{C} \cdot \mathbf{H}^{T} = \mathbf{0}$. For the encoder to be efficiently implemented, the $\mathbf{H}$ matrix of irregular LDPC codes can be rearranged by using the LU decomposition method [18].

2. Measure of Variation

The variance is a measurement of how spread out a distribution is or, in this study; it shows how much the parity check matrix elements have dispersed from their average value (mean value). The variance is computed as the average squared deviation of each number from its mean. Measures of variance are generally used to feature data; the more the data is different, the higher the variance is. The standard deviation (SD) has proven to be an extremely useful measure of spread in part because it is mathematically tractable. The SD can be calculated from:

International Journal of Electronics and Communication Engineering & Technology (IJECET), ISSN 0976 – 6464(Print), ISSN 0976 – 6472(Online) Volume 4, Issue 1, January- February (2013), © IAEME

$$SD = \sqrt{\frac{\sum (x - \bar{x})^2}{n-1}} \quad (3)$$

where,x are the power (the order of cyclic shifting)of α, $\bar{x}$ isthemean of the data set and n is the number of samples.

However, to compare the spreading of data sets with significant different mean values, using the coefficient of variation (CV) is better than the common SD, where;

$$CV = \frac{SD}{\bar{x}} \times 100\% \quad (4)$$

The CV can be used as a condition for choosing the best parity check matrix. This statistical parameter will show how spread out or the dispersion location of '1's in the parity check matrix. The elements of the parity check matrix with a higher CV value have a higher degree of dispersion.In a parity check matrix with a normal curve distribution, the lower the CV value, the more uniformed the dispersion of its elements, and thus the cause of higher performance.

IV. NEW CONSTRUCTION OF PARITY CHECK MATRICES

The structure of the parity check matrix affects the coding and performance substantially. In this proposed algorithm, the matrix is constructed using the different numbers in a magic square as cyclic shifting orders of each sub-permutation. Fig.2 shows the flow chart of the construction procedures.

1. Method I

Method I entailsconstructing the parity check matrix using the original numbers and positions of the magic square, as shown in (5), where the relationship between all the rows is that they have the same summation product (magic square property). The matrix structure constructed using method I is shown in (6). All sizes of magic squares that relate to the block length and rate are investigated.

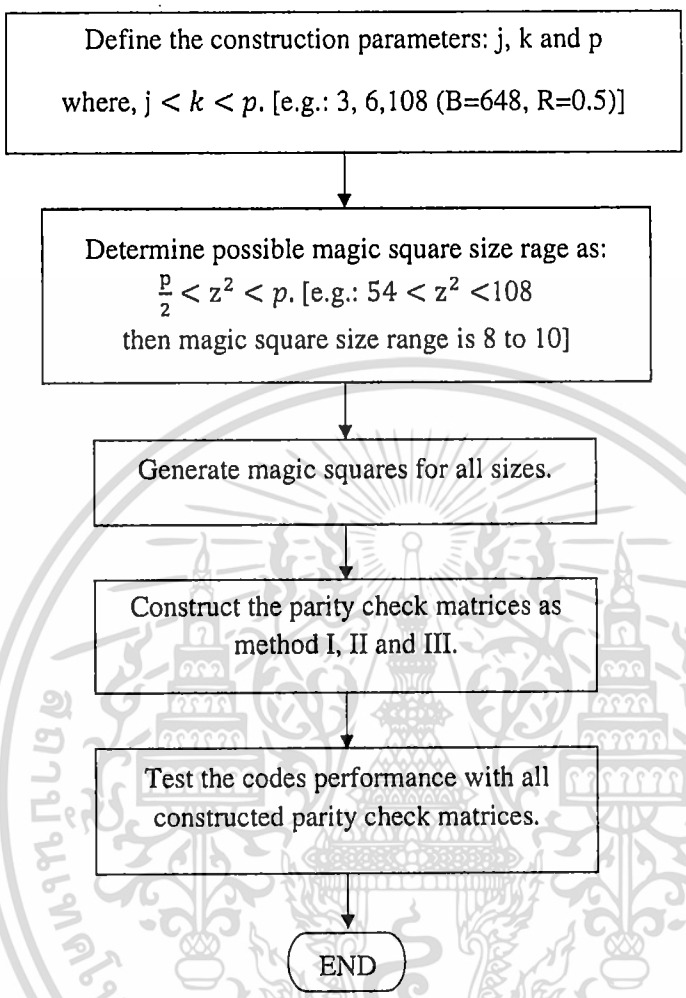
2. Method II

Method II aims toconstruct the parity check matrix using the original numbers and positions of the magic square without the equality of summation. All used numbers are independent of each other. This relationship is broken down by searching for number 1 in the magic square and to begin picking up numbers from this position (including the number 1). The matrix structure constructed using method II is shown in (7).

3. Method III

This method aimsto construct the parity check matrix using some of the numbers of the magic square. All are independent from each other and the rest of the numbers are arranged to ensure the total normal curve distribution property of all the used numbers. The matrix structure constructed using method III is shown in (8).As examples, the construction results using 7 × 7 magic squares (as shown in Fig.3) are shown in Fig.4.

International Journal of Electronics and Communication Engineering & Technology (IJECET), ISSN 0976 – 6464(Print), ISSN 0976 – 6472(Online) Volume 4, Issue 1, January- February (2013), © IAEME



$$MS = \begin{bmatrix} (u,v) & (u,v+1) & (u,v+2) & \cdots & (u,z-1) & (u,z) \\ (u+1,v) & (u+1,v+1) & (u+1,v+2) & \cdots & (u+1,z-1) & (u+1,z) \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ (z-1,v) & (z-1,v+1) & (z-1,v+2) & \cdots & (z-1,z-1) & (z-1,z) \\ (z,v) & (z,v+1) & (z,v+2) & \cdots & (z,z-1) & (z,z) \end{bmatrix}_{(z\times z)} \quad (5)$$

$$H1 = \begin{bmatrix} I & I & I & \cdots & I & \cdots & \cdots & I & I & \cdots & \cdots & I \\ 0 & I & \alpha^{(u,v)} & \alpha^{(u,v+1)} & \cdots & \alpha^{(u,z-1)} & \alpha^{(u,z)} & \alpha^{(u+1,v)} & \alpha^{(u+1,v+1)} & \cdots & \alpha^{(u+i,z-1)} & \alpha^{(u+i,z)} \\ 0 & 0 & I & \alpha^{(u+2,v)} & \cdots & \alpha^{(u+2,z-1)} & \alpha^{(u+2,z)} & \alpha^{(u+3,v)} & \alpha^{(u+3,v+1)} & \cdots & \alpha^{(u+i,z-1)} & \alpha^{(u+i,z)} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & I & \alpha^{(z-1,v)} & \cdots & \alpha^{(z-1,v-1)} & \alpha^{(z,v)} & \alpha^{(z,v+1)} & \cdots & \alpha^{(z,z-1)} & \alpha^{(z,z)} \end{bmatrix}_{(j\times k)} \quad (6)$$

International Journal of Electronics and Communication Engineering & Technology (IJECET), ISSN 0976 – 6464(Print), ISSN 0976 – 6472(Online) Volume 4, Issue 1, January- February (2013), © IAEME

$$H2=\begin{bmatrix} I & I & I & \cdots & I & \cdots & \cdots & I & I & \cdots & \cdots & I \\ 0 & I & \alpha^{(u,v)} & \alpha^{(u,v+1)} & \cdots & \alpha^{(u+1,v)} & \alpha^{(u+1,v+1)} & \alpha^{(u+1,v+2)} & \alpha^{(u+1,v+3)} & \cdots & \alpha^{(u+i,z-1)} & \alpha^{(u+i,z)} \\ 0 & 0 & I & \alpha^{(u+2,z)} & \cdots & \alpha^{(z-1,v)} & \alpha^{(z-1,v+1)} & \alpha^{(z-1,v+2)} & \alpha^{(z-1,v+3)} & \cdots & \alpha^{(z,v)} & \alpha^{(z,v+1)} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & I & \alpha^{(z,z-1)} & \cdots & \alpha^{(u,v)} & \alpha^{(u,v+1)} & \alpha^{(u,v+2)} & \cdots & \alpha^{(u,z-1)} & \alpha^{(u,z)} \end{bmatrix}_{(jxk)} \quad (7)$$

$$H3=\begin{bmatrix} I & I & I & \cdots & I & \cdots & \cdots & I & I & \cdots & \cdots & I \\ 0 & I & \alpha^{(u,v)} & \alpha^{(u,v+1)} & \alpha^{(u,v+2)} & \cdots & \alpha^{(u,v+(z-1))} & \alpha^{(u+1,v+1)} & \alpha^{(u+1,v+2)} & \cdots & \alpha^{(u+i,v+(k-1))} & \alpha^{(u+i,v+k)} \\ 0 & 0 & I & \alpha^{(p/2)} & \alpha^{(p/2)-1} & \cdots & \alpha^{(p/2)-(i+1)} & \alpha^{(p/2)-(i+2)} & \alpha^{(p/2)-(i+3)} & \cdots & \alpha^{(p/2)-(k-1)} & \alpha^{(p/2)-(k)} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & I & \alpha^{(u+j,v+i)} & \cdots & \alpha^{(u+j,v+(i+1))} & \alpha^{(u+j,v+(i+2))} & \alpha^{(u+j,v+(i+3))} & \cdots & \alpha^{(u+j,v+(k-1))} & \alpha^{(u+j,v+k)} \end{bmatrix}_{(jxk)} \quad (8)$$

30	39	48	1	10	19	28
38	47	7	9	18	27	29
46	6	8	17	26	35	37
5	14	16	25	34	36	45
13	15	24	33	42	44	4
21	23	32	41	43	3	12
22	31	40	49	2	11	20

Fig.3 .7 × 7magic square (Siamese method).

H	1	2	3	4	5	6	7	8	9
1	**I**	**I**	**I**	**I**	**I**	**I**	**I**	**I**	**I**
2	**0**	**I**	**1**	**10**	**19**	**28**	**38**	**47**	**7**
3	**0**	**0**	**I**	**27**	**26**	**25**	**14**	**13**	**12**

Fig.4. Method III with 7 × 7 magic square.

V. SIMULATION PROCEDURE AND RESULTS

To obtain the best parity check matrix construction algorithm based on MSBA, three rearrangement methods proposed in the previous section are investigated and the best method should be chosen. Each method is investigated using the same conditions of block length, code rate, iteration and magic square size. A total of 27 constructed parity check matrices (as shown in Table 1) were individually investigated using the parameters shown in Table 3 (for MSBA at a block length of 513).

International Journal of Electronics and Communication Engineering & Technology (IJECET), ISSN 0976 – 6464(Print), ISSN 0976 – 6472(Online) Volume 4, Issue 1, January- February (2013), © IAEME

Table 1. List of all magic square for block length of 513.

Group	Group alias	No.	Size(z × z)
G.1	Jupiter	I 01,II 01,III 01	4x4
	Mars	I 02,II 02,III 02	5x5
	Sol	I 03,II 03,III 03	6x6
	Venus	I 04,II 04,III 04	7x7
G.2	Odd order	I 05,II 05,III 05	5x5
		I 06,II 06,III 06	7x7
G.3	Doubly even	I 07,II 07,III 07	4x4
G.4	Singly even	I 08,II 08,III 08	6x6 (LUX)
		I 09,II 09,III 09	6x6 (Strachey)

The results of methods I, II, and III, which are shown in Figures 5 to 7, respectively,indicate that method I yields a lower performance than methods II and III, for all the magic square sizes used. This is because the dependence of the extrinsic information, exchanged in the iterative decoding, degrades the performance of the LDPC decoders as mentioned by [19] and [20]. By using the original numbers and positions according to the magic square property, the optimal performance cannot be achieved. Method III outperforms method II for all the magic square sizes used. All the used numbers are independent from each other and the normality of the number distribution curve is higher than in methods I and II. The normality was tested by using the probability plot. The obtained probability values of methods I, II, and III are 0.143, 0.249, and 0.336, respectively. It can be observed that the construction with the 6 × 6 magic square yields the highest performance of all the methods. The CV values of the parity check matrices constructed with these 6 × 6magic squares are less than the other magic square sizes.

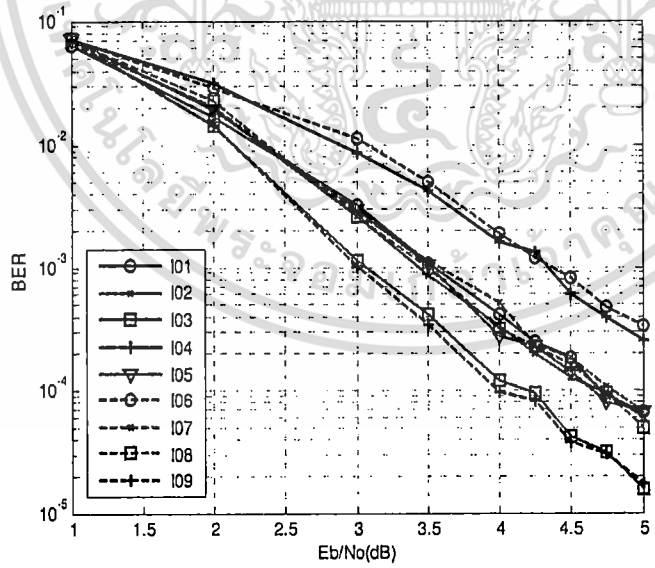


Fig. 5. Test results of method I (R=0.7).

International Journal of Electronics and Communication Engineering & Technology (IJECET), ISSN 0976 – 6464(Print), ISSN 0976 – 6472(Online) Volume 4, Issue 1, January- February (2013), © IAEME

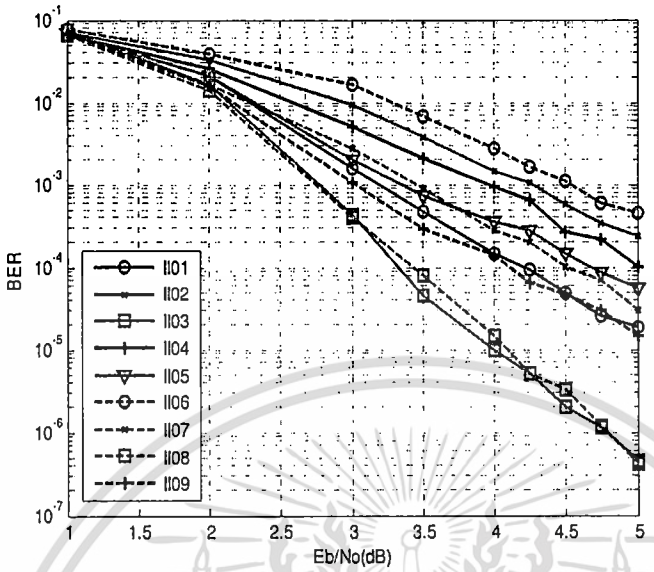


Fig. 6. Test results of method II (R=0.7).

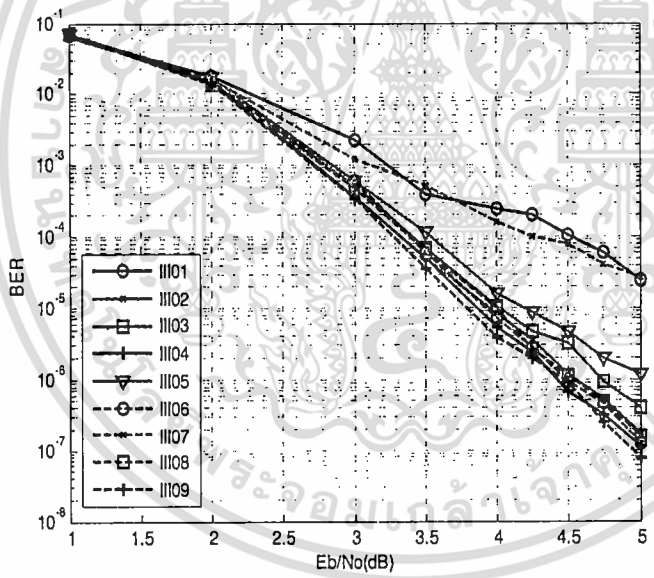


Fig. 7. Test results of method III (R=0.7).

Among all the curves, the best codes are obtained by using method III with a 6×6 magic square (Type 4: The Strachey method). Clearly, there is only one appropriate magic square for this block length and rate; not all magic square sizes can be used. Furthermore, the suitable magic square for each block length and rate can be discerned.Therefore, in all subsequent sections in this study, method III will be the only construction method that will be considered.

International Journal of Electronics and Communication Engineering & Technology (IJECET), ISSN 0976 – 6464(Print), ISSN 0976 – 6472(Online) Volume 4, Issue 1, January- February (2013), © IAEME

The 6 × 6 magic squares can be either generated by the Strachey or LUX methods. The results show that the parity check matrix constructions with these magic squares yield different performances; Strachey gives a better performance than LUX. Obtained BER are about 10^{-7} and 10^{-6}, respectively. CV value can be used to identify the proper formulated matrix. The CV of the parity check matrix constructed with the Strachey method is lower than that of the LUX magic square (66.5% and69.9%, respectively). A more uniformed dispersion of the elements leads to a higher performance.

After the desirable magic square was obtained (6 × 6 magic square, in this case), the number of iterations is then investigated. The number of iterations varied at 1, 5, 10, 30 and50. The obtained results are shown in Fig. 8. It is shown that the code performance cannot be further improved when the number of iteration is above 30. Therefore, MSBA can offer the best performance at iteration of 30. This number is used throughout the study.

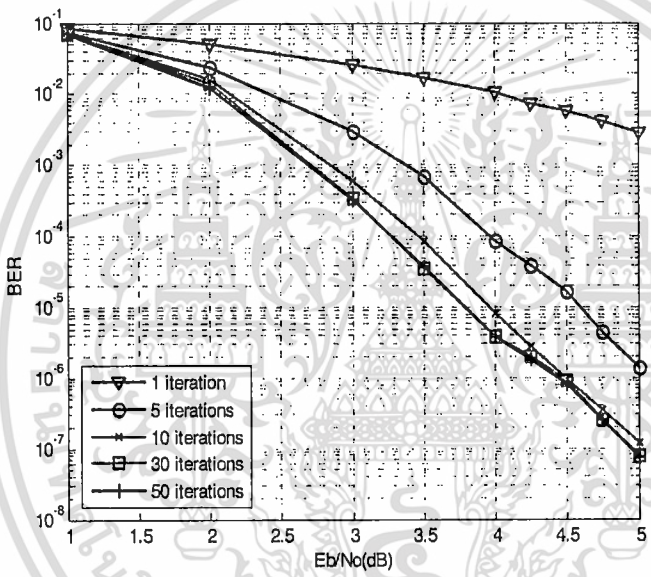


Fig. 8. Code Performances as a function of iterations (block length of 513 with 6 × 6 magic square).

For the code with the length of 513, it has been shown that method III is the best and iteration of 30 is appropriate for this method. Moreover, the code performance for a longer block length with a fixed rate is investigated. The parameters used are shown in Tables 2 and 3 (forthe length of 1010). Among all the constructed parity matrices, the best decoding performance can be obtained by using the 9 × 9 magic square (Group 2: Siamese method) as shown in Fig. 9. It can be noted that when the block length is varied, the appropriate magic square group and size may be changed.

International Journal of Electronics and Communication Engineering & Technology (IJECET), ISSN 0976 – 6464(Print), ISSN 0976 – 6472(Online) Volume 4, Issue 1, January- February (2013), © IAEME

Table 2. List for block length of 1010.

Group	Group alias	No.	Size (z × z)
G.1	Jupiter	III 11	4x4
	Mars	III 12	5x5
	Sol	III 13	6x6
	Venus	III 14	7x7
	Mercury	III 15	8x8
	Lunar	III 16	9x9
G.2	Odd order	III 17	5x5
		III 18	7x7
		III 19	9x9
G.3	Doubly even	III 110	4x4
		III 111	8x8
G.4	Singly even	III 112	6x6 (LUX)
		III 113	6x6 (Strachey)
		III 114	10x10 (LUX)
		III 115	10x10 (Strachey)

Table 3. Construction parameters for the short block length codes.

	MAC [10]			CRT [12]			MSBA		
J	3	3	3	3	3	3	3	3	3
K	9	10	15	9	10	15	9	10	15
P	59	101	67	57	102	68	57	101	67
R = 1-(j/k)	0.7	0.7	0.8	0.7	0.7	0.8	0.7	0.7	0.8
b = p(k-j)	354	707	804	342	714	816	342	707	804
Parity = jp	177	303	201	171	306	204	171	303	201
c = kp	531	1010	1005	513	1020	1020	513	1010	1005

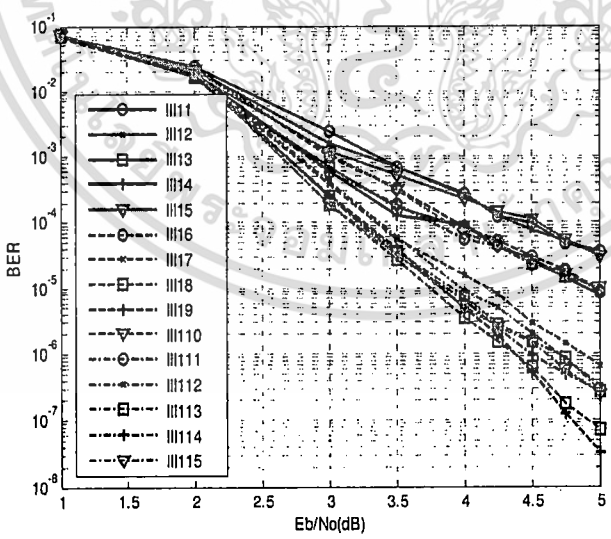


Fig. 9. Test result block length of 1010 (R=0.7).

International Journal of Electronics and Communication Engineering & Technology (IJECET), ISSN 0976 – 6464(Print), ISSN 0976 – 6472(Online) Volume 4, Issue 1, January- February (2013), © IAEME

Next, the code performance for a higher rate with a fixed block length is investigated. The code rate was changed to 0.8. The parity check matrices were constructed using possible magic squares listed in Table 4 with the same construction parameters as shown in Table 3 (last column). The results are shown in Fig. 10. It can be seen that with the 7×7 magic square (Group 2:Siamese method), the best performance can be obtained. This implies that when the rate is varied, the appropriate magic square group and size may be changed.

Table 4.List for rate of 0.8.

Group	Group alias	No.	Size ($z \times z$)
G.1	Jupiter	III 21	4x4
	Mars	III 22	5x5
	Sol	III 23	6x6
	Venus	III 24	7x7
	Mercury	III 25	8x8
G.2	Odd order	III 26	5x5
		III 27	7x7
G.3	Doubly even	III 28	4x4
		III 29	8x8
G.4	Singly even	III 210	6x6 (LUX)
		III 211	6x6 (Strachey)

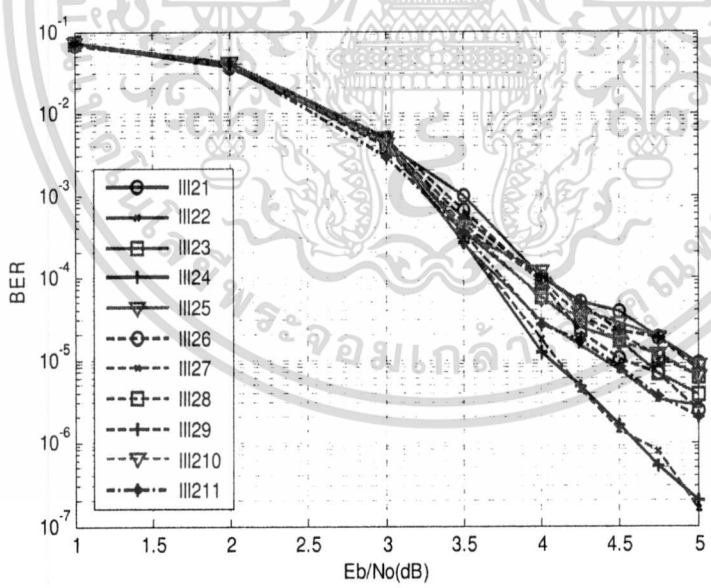


Fig.10. Test resultsfor code rate of 0.8, (length=1005).

International Journal of Electronics and Communication Engineering & Technology (IJECET), ISSN 0976 – 6464(Print), ISSN 0976 – 6472(Online) Volume 4, Issue 1, January- February (2013), © IAEME

With the simulation results, it can be noted that the selection of the appropriate magic square for the parity check matrix construction can be specified as:

$$\frac{p}{2} \le z^2 \le p \tag{9}$$

With the condition given in (9) above, a few values of z remain considerable. In choosing the best value of z for the magic square generation, the statistical parameter was taken into account. The measure of variation was conducted to determine whether the distribution of numbers in the parity check matrix is uniform. According to (4) given in Section III, the lower the CV value, the more uniform distribution of numbers in the parity check matrix.

For example, two magic squares, 6 × 6 and 7 × 7.They are used in a parity check matrix construction for the LDPC codes with a block length of 513 (j=3, k=9 and p=57). Sincep/2=(57-1)/2=28, therefore4 × 4=16 and 5 × 5=25 do not satisfy (9). To choose the best magic square, the CV of the parity check matrix can be computed by using (4). The CV of the parity check matrix constructed from 6 × 6 (as shown in Fig. 11) and 7 × 7 (as shown in Fig. 4) magic squares are 66.5 % and71.2 %, respectively. The CV of z = 6 is less thanz = 7.The calculation results shown that, using a 6 × 6 magic square results in a better performance of the code. The results shown in Fig. 7 positively support this argument.

H	1	2	3	4	5	6	7	8	9
1	**I**	**I**	**I**	**I**	**I**	**I**	**I**	**I**	**I**
2	**0**	**I**	**1**	**6**	**26**	**19**	**24**	**3**	**32**
3	**0**	**0**	**I**	**28**	**27**	**25**	**14**	**13**	**12**

Fig. 11. Method III with 6 × 6 magic square.

To investigate the merit of the proposed algorithm, MSBA is compared with MAC of [10], CRT of [12], Stefan [13]andMacKay [14]at the same block length and code rate. Test parameters are given in Table 3 for block lengths about 1,000 (at code rate of 0.8). It should be noted that there are some differences in construction parameters which are due to the limitations of each construction method. Finally, it is fairly reasonable to define the best parameters for each method in the same class of irregular LDPC codes. The obtained results (as shown in Fig. 12) show that MSBA outperforms the others.

International Journal of Electronics and Communication Engineering & Technology (IJECET), ISSN 0976 – 6464(Print), ISSN 0976 – 6472(Online) Volume 4, Issue 1, January- February (2013), © IAEME

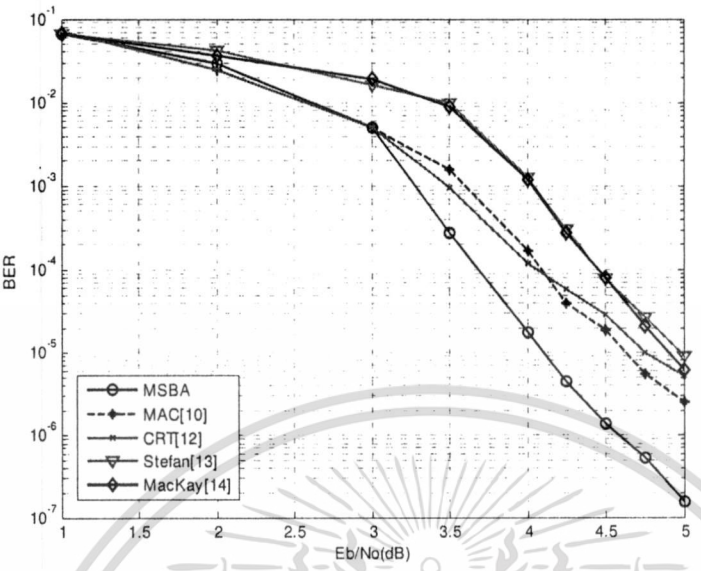


Fig. 12. Block length of about 1000 at rate of 0.8.

VI. CONCLUSION

This study proposes a construction algorithm of a parity check matrix for a short irregular LDPC code applied to an AWGN channel. To obtain the best matrix, the appropriate different shifting orders are needed for a sub-permutation. It has been shown that the proposed magic square base algorithm can be well applied to a part of the parity check matrix construction. With this algorithm, the obtained parity check matrices can be constructed without cycle-4. Four groups of available magic squares and their generation algorithms are presented and investigated in this study.Three new construction methods are proposed and the best method is determined. With the same construction parameters, method III yields the best performance due to all the numbers being independent of each other and that their numbers follow a normal curve distribution. The appropriate magic square (size and group) must, however, relate to block length and rate. There are two parametric studies, increment of block length with fixed rate and increment of rate with fixed block length. The simulation results show that the performances are improved for both cases. The obtained results were also compared with the existing short irregular codes reported by [10],[12], [13] and [14]. MSBA significantly outperforms these codes for short block lengths and high code rates. It is found thatthis construction algorithm has a lower complexity as the generation of numbers via the MSBA method is simpler than the CRT method [12]. Obviously, the use of magic square in a parity check matrix construction is promising. However it should be noted that for a better performance of the constructed code, some modifications of the numbers selection must be considered. Moreover, the obtained structured parity check matrix can be suitably used for a short block length with a rate higher than 0.7. The BER of about 10^{-7} at SNR of 5 dB was achieved with a low iteration of only 30. These codes can be used in wireless and mobile applications where memory is restricted.

International Journal of Electronics and Communication Engineering & Technology (IJECET), ISSN 0976 – 6464(Print), ISSN 0976 – 6472(Online) Volume 4, Issue 1, January- February (2013), © IAEME

REFERENCES

[1] R. Gallager, "Low-density Parity-check Code", IRE Transaction Information Theory, pp. 21-28, 1962.
[2] R. M. Tanner, "A recursive approach to low complexity codes", IEEE Transaction Information Theory, vol. 27, pp. 533-547, 1981.
[3] D. J. C. Mackay and R. Neal, "Near Shannon Limit Performance of LDPC Code", Electronic Letter, vol. 33, pp. 457-458, 1997.
[4] D. J. C. MacKay, "Good error-correcting codes based on very sparse matrices", IEEE Transaction Information Theory, vol. 45, No. 2, pp. 399-431, 1999.
[5] M. Blaum, P. Farrell, and H. van Tilborg, "Array codes, in Handbook of Coding Theory", V. S. Pless and W. C. Huffiman Eds., Elsevier 1998.
[6] J. L. Fan, "Array Codes as Low-Density Parity-", Proceeding 2nd International Symposium TB & Relate Topics, pp. 543-546, 2000.
[7] M. P. C. Fossorier, "Quasi-Cyclic Low-Density Parity-Check Codes from Circulant Permutation Matrices", IEEE Transaction Information Theory, vol. 50, No. 8, 2004.
[8] S. Myung and K. Yang, "A combing method of quasi-cyclic LDPC codes by the Chinese Remainder Theorem", IEEE Communications Letter, vol. 9, pp. 823-825, 2005.
[9] M. G. Luby, M. Mitzenmacher, M.A. Shokrollahi, and D.A. Spielman, "Efficient erasure correcting codes", IEEE Transaction Information Theory, vol. 47, pp. 569-584, 2001.
[10] E. Eleftheriou and S.Olcer, "LDPC Codes for Digital Subscriber Lines", Proceeding International Conference on Communication, pp. 1752-1757, 2002.
[11] D. Abematsu, T. Ohtsuki, S. PW Jarot, and T. Kashima, "Size Compatible (SC)-Array LDPC Codes", IEEE Vehicular Technology Conference, pp.1147-1151, 2007.
[12] C. Chusin, C. Prasartkaew, S. Timakul and S. Choomchuay, "A Desing of Nonprime Block Irregular LDPC Codes via CRT", ISCIT International Conference, Japan, 2010.
[13]L. Stefan and M.Olgica, "LDPC Codes Based on Latin Squares: Cycle Structure, Stopping Set, and Trapping Set Analysis", IEEE Transactions on Communications, vol. 55, No. 2, pp. 303-312, 2007.
[14] D. MacKay, Gallager code esources [Online], Available: http://www.inference.phy.cam.ac.uk/mackay/CodesFiles.html
[15] Weisstein, E. W. Magic Square [online]. Available from http://mathworld.olfram.com/MagicSquare.html, 2003.
[16] William H. Benson and Oswald Jacoby, New Recreations with Magic Squares. New York: Dover, 1976.
[17] Ball, W. W. R., Mathematical recreations and essays. London. Macmillan & Co Ltd., 1959.
[18] C. Prasartkaew and S. Choomchuay, "A Parity Check Matrix Design for Irregular LDPC Codes with 2K Block Length", ISPACS International Symposium on Intelligent Signal Processing and Communication Systems, Japan, 2009.
[19] J. Fan, Y. Xiao, and K. Kim, "Design LDPC Codes without Cycles of Length 4 and 6", Hindawi Publishing Corporation Research Letters in Communications, 2008.
[20] R. Lucas, M. P. C. Fossorier, Y. Kou, and S. Lin, "Iterative decoding of one-step majority logic decodable codes based on belief propagation," IEEE Transactions on Communications, vol. 48, no. 6, pp. 931-937, 2000.

ประวัติผู้เขียน

ชื่อ-นามสกุล	ผู้ช่วยศาสตราจารย์ชุติมา ประสาทแก้ว
วัน เดือน ปีเกิด	4 มกราคม 2515 ที่ลพบุรี
ที่อยู่	578/10 ซอยลาดพร้าว 112 แขวงพลับพลา เขตวังทองหลาง กรุงเทพฯ 10310
ประวัติการศึกษา	
พ.ศ. 2538	ครุศาสตรบัณฑิต สาขาวิชาคอมพิวเตอร์ศึกษา วิทยาลัยครูเทพสตรี
พ.ศ. 2543	ครุศาสตร์อุตสาหกรรมมหาบัณฑิต สาขาวิชาคอมพิวเตอร์และเทคโนโลยีสารสนเทศ มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี
ความชำนาญเฉพาะด้าน	1) คอมพิวเตอร์และเทคโนโลยีสารสนเทศ 2) คณิตศาสตร์สำหรับคอมพิวเตอร์
ประสบการณ์การทำงาน	
พ.ศ. 2538-2541	รับราชการอาจารย์คณะศิลปศาสตร์ สถาบันเทคโนโลยีราชมงคล สอนวิชาคอมพิวเตอร์ วิชาคณิตศาสตร์และวิชาสถิติ
พ.ศ. 2541-2548	อาจารย์คณะวิทยาศาสตร์ สถาบันเทคโนโลยีราชมงคล สาขาวิทยาการคอมพิวเตอร์และสาขาเทคโนโลยีคอมพิวเตอร์
พ.ศ.2548-2555	ตำแหน่งผู้ช่วยศาสตราจารย์ประจำคณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยเทคโนโลยีราชมงคลธัญบุรี เป็นผู้ประเมินคุณภาพการศึกษาภายในระดับอุดมศึกษา ผู้ประเมินฯ ภายนอกระดับอาชีวศึกษา และผู้ประเมินฯภายนอก กศน. จนถึงปัจจุบัน
ปัจจุบัน	ดำรงตำแหน่งหัวหน้าภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์ มหาวิทยาลัยเทคโนโลยีราชมงคลธัญบุรี
ทุนที่ได้รับ	
พ.ศ. 2541	ทุนศึกษาต่อระดับปริญญาโทจากสถาบันเทคโนโลยีราชมงคล
พ.ศ. 2551	ทุนวิจัยจากสภาวิจัยแห่งชาติ(วช.) และทุนศึกษาต่อระดับปริญญาเอกจากมหาวิทยาลัยเทคโนโลยีราชมงคลธัญบุรี