KKU Res.j. 2014; 19(4) : 550-559
http://resjournal.kku.ac.th

วิธีการจัดรูปแบบบิตข้อมูลแบบหลายแทร็กสำหรับระบบบีพีเอ็มอาร์
A Multi-track Recorded-Bit Patterning Scheme for Bit-Patterned Media Recording System

อรรถสิทธิ์ อารยางกูร[1], ชานนท์ วริสาร[1], ปิยะ โควินท์ทวีวัฒน์[2]*
*Autthasith Arrayangkool[1], Chanon Warisarn[1] *, and Piya Kovintavewat[2]*

[1]วิทยาลัยนวัตกรรมการจัดการข้อมูล สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
[2]ศูนย์วิจัยเทคโนโลยีการบันทึกข้อมูล มหาวิทยาลัยราชภัฏนครปฐม
**Correspondent author: kwchanon@kmitl.ac.th*

บทคัดย่อ

การแทรกสอดระหว่างสัญลักษณ์ (inter-symbol interference: ISI) และการแทรกสอดระหว่างแทร็ก (inter-track interference: ITI) เป็นปัญหาสำคัญในระบบการบันทึกข้อมูลของสื่อบันทึกที่มีการจัดรูปแบบบิตหรือที่เรียกว่าระบบบีพีเอ็มอาร์ (bit-patterned media recording: BPMR) เพราะระยะห่างระหว่างบิตและระยะห่างระหว่างแทร็กมีน้อยมาก สำหรับระบบบีพีเอ็มอาร์ที่ความจุข้อมูลสูง วิธีหนึ่งที่สามารถช่วยลดผลกระทบที่เกิดจาก ISI และ ITI คือการเข้ารหัสก่อนการบันทึกข้อมูลลงในสื่อบันทึก โดยทั่วไปวิธีการเข้ารหัสสองมิติได้ถูกนำมาใช้เพื่อการเข้ารหัสข้อมูล อย่างไรก็ตามวิธีดังกล่าวต้องการบิตส่วนเกินจำนวนมากจึงส่งผลให้มีอัตรารหัสต่ำ ดังนั้นบทความนี้จึงได้นำเสนอวิธีการเข้ารหัสสองมิติแบบใหม่เรียกว่า “วิธีการจัดรูปแบบบิตข้อมูลแบบหลายแทร็ก (Multi-track Recorded-Bit Patterning: M-RBP)” ซึ่งมีอัตรารหัสที่สูง โดยจะทำการเลื่อนลำดับข้อมูลในแต่ละแทร็กเพื่อหารูปแบบของข้อมูลที่ดีสุดจากตารางค้นหา (look-up table) ที่ถูกสร้างไว้แล้วก่อนหน้า จากนั้นจึงบันทึกข้อมูลที่จัดรูปแบบเรียบร้อยแล้วลงไปยังสื่อบันทึกข้อมูล ซึ่งวิธีการนี้ช่วยลดผลกระทบที่เกิดจาก ISI และ ITI ได้เป็นอย่างดี จากผลการทดลองพบว่า วิธีการที่ได้นำเสนอสามารถช่วยเพิ่มสมรรถนะในด้านอัตราส่วนของสัญญาณต่อสัญญาณรบกวน (Signal-to-Noise Ratio, SNR) ของระบบ เมื่อเปรียบเทียบกับระบบที่ไม่มีการเข้ารหัสข้อมูลได้ถึง 1 เดซิเบล ที่ระดับอัตราบิตผิดพลาดเท่ากับ 10^{-4}

Abstract

The inter-symbol interference (ISI) and inter-track interference (ITI) become a major impairment in bit-patterned media recording (BPMR) because of small bits and track pitches. One way to alleviate ISI and ITI is to encode a data sequence before recording. In general, a two-dimensional (2D) coding scheme is used for this purpose; however, it requires lots of redundant bits, thus lowering a code rate. This paper proposes a novel 2D coding scheme, referred to as a multi-track recorded-bit patterning (M-RBP) scheme, with a high code rate to combat both ISI and ITI effects. Specifically, the data bits will first be patterned by finding the best pattern based on a look-up table before recording that can avoid severe ISI and ITI effects. Simulation results indicate that the proposed scheme can provide a significant signal to noise ratio (SNR) performance improvement when compared to the system without any coding and yield for about 1 dB gain at BER = 10^{-4}.

คำสำคัญ: บีพีเอ็มอาร์; สัญญาณรบกวนแบบเลื่อนตำแหน่ง; การจัดรูปแบบบิตข้อมูล; อีควอไลเซชันแบบสองมิติ
Keywords: BPMR; position jitter noise; recording-bit patterning (RBP); two-dimensional equalization

1. บทนำ

ในปัจจุบันนี้พื้นที่การจัดเก็บข้อมูลดิจิทัลมีความต้องการเพิ่มขึ้นอย่างรวดเร็ว จึงส่งผลให้มีการพัฒนาเพิ่มความจุข้อมูลของฮาร์ดดิสก์ไดร์ฟให้เพียงพอที่จะรองรับความต้องการในการจัดเก็บข้อมูลที่มากขึ้นในอนาคต ระบบบีพีเอ็มอาร์ คือเทคโนโลยีการบันทึกข้อมูลเชิงแม่เหล็กประเภทหนึ่งที่มีความเป็นไปได้ที่จะนำมาใช้ในฮาร์ดดิสก์ไดร์ฟยุคต่อไป ซึ่งคาดว่าจะสามารถเพิ่มความหนาแน่นของการบันทึกข้อมูล (recording density) ได้สูงถึง 4 เทระบิตต่อตารางนิ้ว (Tb/in^2) (1) อย่างไรก็ตามเมื่อความหนาแน่นของการบันทึกข้อมูลสูงขึ้นย่อมส่งผลให้ระยะห่างระหว่างเกาะข้อมูล (bit island) ทั้งในทิศทางขวางแทร็กและในแนวตามแทร็กมีค่าน้อยลง ซึ่งมีผลทำให้ระบบเผชิญกับการแทรกสอดสองมิติที่ประกอบด้วยการแทรกสอดระหว่างสัญลักษณ์ (ISI) และการแทรกสอดระหว่างแทร็ก (ITI) ที่รุนแรงเพิ่มมากขึ้น (2-5) โดยผลกระทบที่เกิดจากการแทรกสอดทั้งสองนี้จะส่งผลให้ระบบบีพีเอ็มอาร์ มีสมรรถนะลดลงอย่างมาก

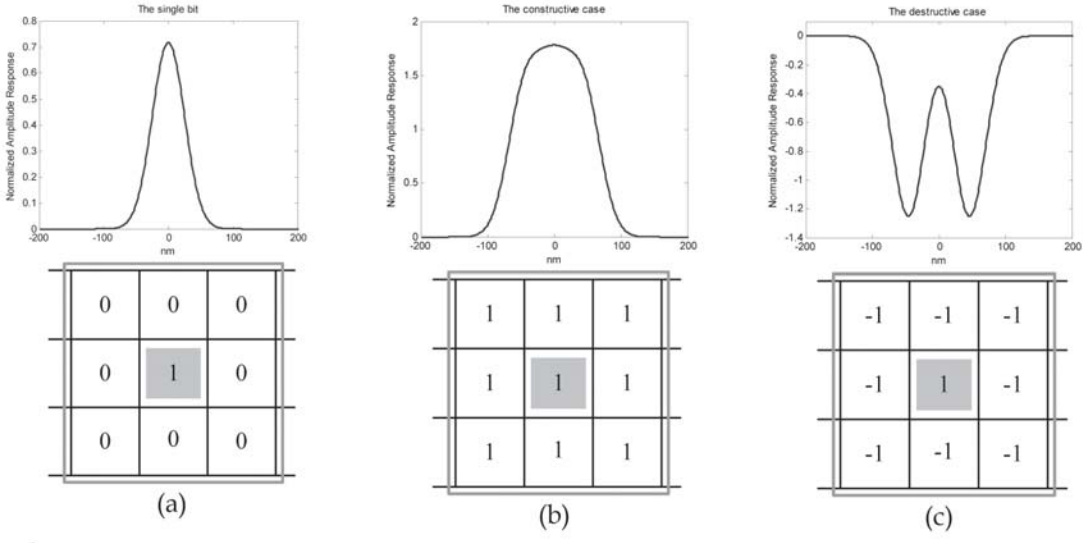
โดยทั่วไปผลกระทบจากการแทรกสอดสองมิติแบ่งเป็น 2 แบบคือ แบบส่งเสริม (constructive) และแบบลดทอน (destructive) โดยพิจารณาจากรูปคลื่นของสัญญาณอ่านกลับ (readback signal waveform) เทียบกับบิตข้อมูลที่ถูกบันทึก บทความนี้ได้ให้ความสนใจเฉพาะสัญญาณอ่านกลับที่ปราศจากโอเวอร์ชู้ต (overshoot) เนื่องจากเป็นสัญญาณที่เกิดขึ้นจริงจากสื่อบันทึกข้อมูล (media) ที่ใช้งานอยู่ในปัจจุบัน ตามที่แสดงในรูปที่ 1 (6) โดยจะสังเกตเห็นว่าในกรณีที่บิตข้อมูลถูกบันทึกลงในสื่อบันทึกเพียงบิตเดียวโดยไม่มีบิตข้อมูลอื่นล้อมรอบ สัญญาณอ่านกลับจะมีค่าแอมพลิจูดประมาณ 0.7 ซึ่งจะเรียกกรณีนี้ว่า “บิตข้อมูลเดี่ยว” ตามรูปที่ 1(a) ซึ่งบิตข้อมูลเดี่ยวนี้จะถูกนำมาใช้เป็นตัวแบ่งแยกระหว่างกรณีการแทรกสอดแบบส่งเสริมและกรณีแทรกสอดแบบลดทอน ในกรณีที่เกิดการแทรกสอดสองมิติแบบส่งเสริมนั้น สัญญาณอ่านกลับของบิตกลางจะมีค่าแอมพลิจูดเพิ่มสูงขึ้น (มากกว่าในกรณีที่เป็นบิตข้อมูลเดี่ยว) เช่น ถ้าบิตข้อมูล ‘1’ ถูกล้อมรอบด้วยบิตข้อมูล ‘1’ ทั้งหมด จะถือว่าเป็นกรณีที่ดีสุดตามรูปที่ 1(b) ในขณะที่ถ้าบิตข้อมูล ‘1’ ถูกล้อมรอบด้วยบิต ‘–1’ จะถือว่าเป็นกรณีที่ด้อยสุดตามรูปที่ 1(c) เพราะจะพบว่าสัญญาณอ่านกลับของบิตกลางจะมีค่าแอมพลิจูดลดต่ำลงมาก ซึ่งจะเรียกเหตุการณ์นี้ว่าระบบเผชิญกับการแทรกสอดสองมิติแบบลดทอน ซึ่งเป็นสิ่งที่ควรหลีกเลี่ยงในการบันทึกข้อมูล เพื่อหลีกเลี่ยงการแทรกสอดสองมิติแบบลดทอน งานวิจัยก่อนหน้านี้ได้มีการนำเสนอรหัส ITI แบบส่งเสริม (constructive ITI code) (2) ที่พิจารณาการบันทึกข้อมูลแบบ 2 แทร็กในระบบการบันทึกข้อมูลเชิงแม่เหล็กแบบแนวตั้ง (perpendicular magnetic recording) จากนั้นวิธีการเข้ารหัสสองมิติ (3) ที่มีอัตรารหัส 7/9 ได้ถูกนำเสนอเพื่อหลีกเลี่ยงบางกรณีที่ก่อให้เกิดการแทรกสอดแบบลดทอน โดยการแทรกบิตส่วนเกินลงไปในตำแหน่งที่ระบุไว้ในอาร์เรย์ขนาด 3×3 แต่เนื่องจากบิตส่วนเกินที่ใช้ในวิธีการเข้ารหัสนี้ไม่มีความสามารถในการแก้ไขข้อผิดพลาด ดังนั้น Shao *et al.* (4) ได้นำเสนอวิธีการเข้ารหัสสองมิติแบบใหม่สำหรับระบบบีพีเอ็มอาร์ที่มีโอเวอร์ชู้ตที่อัตรารหัส 5/6 ซึ่งนอกจากจะให้สมรรถนะดีกว่ารหัสสองมิติที่ได้นำเสนอก่อนหน้านี้แล้วยังมีจำนวนบิตส่วนเกินน้อยกว่าและบิตส่วนเกินยังสามารถช่วยแก้ไขข้อผิดพลาดได้ด้วย สำหรับงานวิจัยที่พิจารณาระบบบีพีเอ็มอาร์ที่ปราศจากโอเวอร์ชู้ตก็ได้มีการนำเสนอวิธีการเข้ารหัสสองมิติเช่นกัน ยกตัวอย่างเช่น วิธีการจัดรูปแบบบิตข้อมูลก่อนการบันทึก (recorded-bit patterning: RBP) แบบง่าย (5) ที่มีอัตรารหัสสูง เพื่อหลีกเลี่ยงการแทรกสอดสองมิติแบบลดทอน ซึ่งถูกนำเสนอให้ใช้งานร่วมกับแบบจำลองช่องสัญญาณแบบ 3 แทร็กที่มีแถบป้องกัน (guard band) คั่นระหว่างแทร็ก ซึ่งการแทรกแถบป้องกันลงไปในสื่อบันทึกนั้นจะเป็นการสูญเสียพื้นที่บางส่วนในการบันทึกข้อมูล ดังนั้นบทความนี้จะนำเสนอวิธีการจัดรูปแบบบิตข้อมูลแบบหลายแทร็ก (M-RBP) ที่ใช้ในแบบจำลองช่องสัญญาณหลายแทร็กที่ไม่มีแถบป้องกันคั่นระหว่างแทร็กทำให้ได้พื้นที่ในการบันทึกข้อมูลเพิ่มมากขึ้น โดยเริ่มต้นจากการนำลำดับข้อมูลที่ต้องการบันทึกมาเลื่อนวน (circular shift) เพื่อหารูปแบบของบิตข้อมูลที่ดีสุดที่ไม่ก่อให้เกิดการแทรกสอดสองมิติแบบลดทอนจากตารางค้นหา (look-up table) ก่อนจะทำการบันทึกข้อมูลที่มีรูปแบบข้อมูลที่ดีสุดลงในสื่อบันทึก จากผลการทดลองพบว่าวิธีการที่ได้นำเสนอสามารถช่วยเพิ่มสมรรถนะในด้านอัตราส่วนของสัญญาณต่อสัญญาณ

รบกวน (Signal-to-Noise Ratio, SNR) ของระบบได้เป็นอย่างดีเมื่อเปรียบเทียบกับระบบที่ไม่มีการเข้ารหัสข้อมูลและระบบ RBP (5)ได้ถึง 1 เดซิเบล และ 0.5 เดซิเบลตามลำดับที่ระดับอัตราบิตผิดพลาดเท่ากับ 10^{-4}

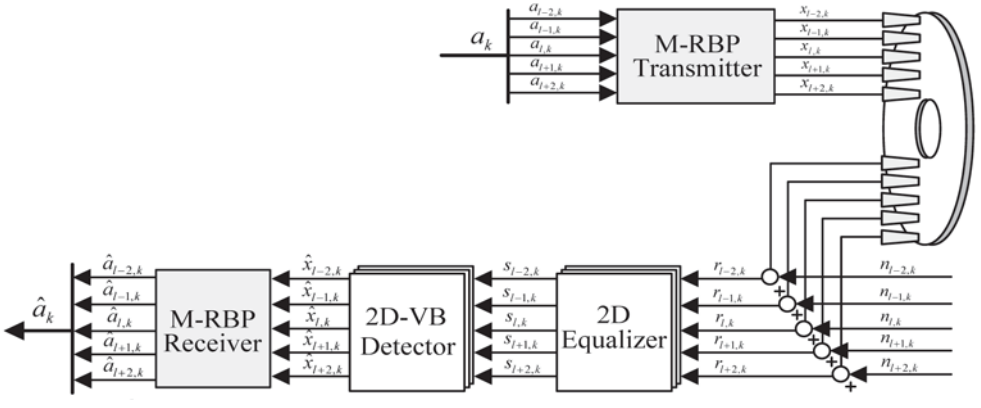
บทความนี้จะอธิบายแบบจำลองช่องสัญญาณของระบบ บีพีเอ็มอาร์ แบบหลายแทร็กในหัวข้อที่ 2 จากนั้นหัวข้อที่ 3 อธิบายการทำงานของวิธีการจัดรูปแบบบิตข้อมูลแบบหลายแทร็ก หัวข้อที่ 4 นำเสนอผลลัพธ์จากการจำลองระบบ และหัวข้อที่ 5 สรุปบทความนี้

2. ช่องสัญญาณระบบการบันทึกข้อมูลเชิงแม่เหล็กแบบบีพีเอ็มอาร์

แบบจำลองช่องสัญญาณของระบบบีพีเอ็มอาร์ที่ใช้ในบทความนี้แสดงในรูปที่ 2 โดยลำดับข้อมูลแบบไบนารี $a_k \in \{\pm 1\}$ ที่ต้องการบันทึกจะถูกแบ่งออกเป็น 5 แทร็ก ก่อนที่ข้อมูลทั้ง 5 แทร็กจะถูกส่งต่อไปยังวงจร M-RBP เพื่อเข้ารหัสข้อมูลก่อนการบันทึกลงในสื่อบันทึก



รูปที่ 1 สัญญาณอ่านกลับของ (a) บิตข้อมูลเดี่ยว (b) กรณีที่ดีสุด เมื่อบิตข้อมูล '1' ถูกล้อมรอบด้วยบิตข้อมูล '1' และ (c) กรณีที่ด้อยสุด เมื่อบิตข้อมูล '1' ถูกล้อมรอบด้วยบิตข้อมูล '–1'



รูปที่ 2 แบบจำลองช่องสัญญาณของระบบ บีพีเอ็มอาร์ ที่ใช้อัลกอริทึม M-RBP

สัญญาณอ่านกลับของบิตข้อมูลที่ k^{th} บนแทร็กที่ l^{th} เขียนเป็นสมการคณิตศาสตร์ได้ดังนี้ (6)

$$r_{l,k} = \sum_n \sum_m h_{m,n} x_{l-m,k-n} + n_{l,k} = x_{l,k} \otimes h_{l,k} + n_{l,k} \quad 1$$

เมื่อ $x_{l,k}$ คือบิตข้อมูลที่จะบันทึกลงบนสื่อบันทึกข้อมูล, $h_{m,n}$ คือค่าสัมประสิทธิ์ของผลตอบสนองช่องสัญญาณสองมิติ, n และ m แทนตำแหน่งของเกาะข้อมูลในทิศทางตามแนวแทร็กและแนวขวางแทร็กตามลำดับ, $\otimes$ คือเครื่องหมายการดำเนินการคอนโวลูชันแบบสองมิติ (2D convolution), และ $n_{l,k}$ คือสัญญาณรบกวนเกาส์สีขาวแบบบวก (additive white Gaussian noise) ในทางปฏิบัติค่าสัมประสิทธิ์ของผลตอบสนองของช่องสัญญาณสองมิติ $h_{m,n}$ ในระบบ บีพีเอ็มอาร์ หาได้จากการชักตัวอย่าง (sampling) ผลตอบสนองสัญญาณพัลส์ของเกาะข้อมูลเดี่ยวที่มีคาบเวลาบิต (bit period) T_x และความกว้างของแทร็ก (track pitch) T_z คือ

$$h_{m,n} = P\left(-mT_x, -nT_z\right), \quad \{m,n\} \in \left(-L, 0, L\right) \quad 2$$

โดย $P(x, z)$ คือผลตอบสนองสัญญาณพัลส์เกาส์เซียนแบบสองมิติ (6) และ $2L+1$ คือจำนวนสัมประสิทธิ์ของช่องสัญญาณแบบสองมิติ โดยผลตอบสนองสัญญาณพัลส์เกาส์เซียนแบบสองมิติสำหรับระบบบีพีเอ็มอาร์ในงานวิจัยนี้หาได้จาก (6, 8)

$$P(x,z) = A\exp\left\{-\frac{1}{2}\left[\left(\frac{x+\Delta_x}{c\mathrm{PW}_x}\right)^2 + \left(\frac{z+\Delta_z}{c\mathrm{PW}_z}\right)^2\right]\right\} \quad 3$$

เมื่อ $A = 1$ คือแอมพลิจูดสูงสุดของผลตอบสนองสัญญาณพัลส์เกาส์เซียนแบบสองมิติ, Δ_x คือความคลาดเคลื่อนของตำแหน่งตามแนวแทร็ก, Δ_z คือความคลาดเคลื่อนของตำแหน่งตามแนวขวางแทร็ก, PW_x คือค่า PW_{50} ของสัญญาณพัลส์ตามแนวแทร็ก, PW_z คือค่า PW_{50} ของสัญญาณพัลส์ตามแนวขวางแทร็ก, PW_{50} คือความกว้างของสัญญาณพัลส์ ณ จุดครึ่งหนึ่งของแอมพลิจูดสูงสุด, และ c = 1/2.3548 คือค่าคงตัวที่เป็นความสัมพันธ์ระหว่าง PW_{50} และค่าเบี่ยงเบนมาตรฐานของสัญญาณเกาส์เซียน ในที่นี้สัญญาณรบกวนแบบเลื่อนตำแหน่ง (position jitter noise) ตามแนวแทร็ก Δ_x และตามแนวขวางแทร็ก Δ_z จะถูกจำลองโดยฟังก์ชันการแจกแจงความน่าจะเป็นแบบเกาส์เซียนที่มีค่าเฉลี่ยศูนย์และค่าความแปรปรวน σ_j^2 เมื่อ σ_j เป็นร้อยละของความยาวบิต T_x

จากรูปที่ 2 พบว่าลำดับข้อมูล a_k จะถูกแบ่งออกเป็น 5 แทร็ก $\{a_{l-2,k}, a_{l-1,k}, a_{l,k}, a_{l+1,k}, a_{l+2,k}\}$ เพื่อส่งเข้าไปในวงจรเข้ารหัส M-RBP เพื่อจัดเรียงบิตข้อมูลให้พร้อมสำหรับการบันทึกลงในสื่อบันทึกเป็น 5 ลำดับข้อมูล $\{x_{l-2,k}, x_{l-1,k}, x_{l,k}, x_{l+1,k}, x_{l+2,k}\}$ จำนวน 5 แทร็กติดกันคือแทร็กที่ (l-2), (l-1), (l), (l+1), และ (l+2) ตามลำดับ ในกระบวนการอ่านข้อมูล ณ วงจรภาครับ ข้อมูลทั้ง 5 แทร็กจะถูกอ่านโดยใช้ 5 หัวอ่านในเวลาเดียวกัน (หรือใช้ 1 หัวอ่านเพื่ออ่านข้อมูลทีละแทร็ก แล้วใช้บัฟเฟอร์หรือหน่วยความจำช่วยในการเก็บข้อมูลทีละแทร็ก) จากนั้นในกระบวนการอ่าน (read process) ลำดับข้อมูลของสัญญาณอ่านกลับ $r_{l,k}$ จะถูกส่งไปยังวงจรอีควอไลเซอร์แบบสองมิติ (2D equalizer) เพื่อปรับสัญญาณให้เป็นลำดับข้อมูล $s_{l,k}$ และส่งต่อไปยังวงจรตรวจหาวิเทอร์บิแบบสองมิติ (2D Viterbi detector) (6) เพื่อหาลำดับข้อมูลที่เป็นไปได้มากสุด ก่อนจะส่งข้อมูลที่ตรวจหาได้ไปยังวงจรถอดรหัส M-RBP เพื่อจัดเรียงลำดับข้อมูลใหม่โดยอาศัยข้อมูลการเลื่อนลำดับข้อมูลย้อนกลับที่เก็บเอาไว้ในหน่วยความจำหรือบัฟเฟอร์

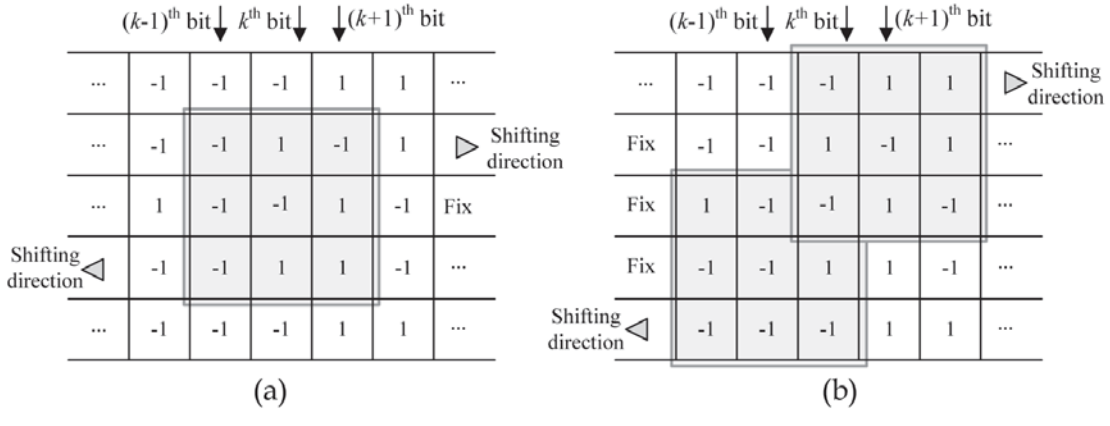
3. วิธีการจัดรูปแบบบิตข้อมูลแบบหลายแทร็ก

เนื่องจากความรุนแรงของการแทรกสอดสองมิติขึ้นอยู่กับทั้งรูปแบบสัญญาณอ่านกลับของบิตข้อมูลที่ตรวจหาและบิตข้อมูลที่อยู่รอบข้าง บทความนี้นำเสนออัลกอริทึม M-RBP เพื่อหลีกเลี่ยงรูปแบบข้อมูลบางรูปแบบที่ก่อให้เกิดข้อผิดพลาดได้ง่าย ณ วงจรตรวจหา โดยการจัดเรียงบิตข้อมูลใหม่ (เพื่อไม่ให้มีรูปแบบที่ก่อให้เกิดข้อผิดพลาดง่าย) ก่อนการบันทึกลงในสื่อบันทึก

พิจารณาหน้าต่างอาร์เรย์ขนาด 3×3 ที่ครอบคลุมบิตข้อมูลทั้งหมด 9 บิต คือ 3 บิตจากแทร็กบน $[a_{l-1,k-1}\ a_{l-1,k}\ a_{l-1,k+1}]$, 3 บิตจากแทร็กกลาง $[a_{l,k-1}\ a_{l,k}\ a_{l,k+1}]$, และ 3 บิตจากแทร็กล่าง $[a_{l+1,k-1}\ a_{l+1,k}\ a_{l+1,k+1}]$ ตามที่แสดงในรูปที่ 3 โดยหน้าต่างนี้จะใช้เพื่อค้นหารูปแบบข้อมูล $[a_{l-1,k-1}\ a_{l-1,k}\ a_{l-1,k+1}\ a_{l,k-1}\ a_{l,k}\ a_{l,k+1}\ a_{l+1,k-1}\ a_{l+1,k}\ a_{l+1,k+1}]$ ที่ก่อให้เกิดข้อผิดพลาดได้

ง่ายในกระบวนการอ่าน ณ วงจรตรวจหา ในการทำงานเริ่มต้นจะกำหนดให้รูปคลื่นของสัญญาณอ่านกลับของบิตข้อมูลเดี่ยวที่ปราศจากโอเวอร์ชู้ตตามรูปที่ 1(a) เป็นสัญญาณอ้างอิง (reference signal) ซึ่งได้มาจากแบบจำลองช่องสัญญาณที่ไม่มีสัญญาณรบกวนใดๆ จากนั้นจะพิจารณารูปแบบข้อมูลทั้งหมดที่อยู่ภายในหน้าต่างอาร์เรย์ขนาด 3×3 ด้วยการตรวจสอบว่าค่าแอมพลิจูดของสัญญาณอ่านกลับของบิตกลาง $\{r_{l,k}\}$ แตกต่างจากค่าแอมพลิจูดของสัญญาณอ้างอิงหรือไม่อย่างไร ถ้ารูปแบบข้อมูลใดที่มีผลทำให้ค่าแอมพลิจูดของสัญญาณอ่านกลับของบิตกลาง $\{r_{l,k}\}$ มีเครื่องหมายตรงข้ามกับเครื่องหมายของค่าแอมพลิจูดของสัญญาณอ้างอิง จะถือว่าเป็นรูปแบบข้อมูลแบบลดทอน (destructive data pattern) ในขณะเดียวกันถ้ารูปแบบข้อมูลอื่นๆ นอกเหนือจากนี้จะถือว่าเป็นรูปแบบข้อมูลแบบส่งเสริม (constructive data pattern) ซึ่งในตารางที่ 1 ได้แสดงรูปแบบข้อมูลแบบลดทอนทั้งหมดที่ก่อให้เกิดข้อผิดพลาด ณ บิตกลาง เมื่อเทียบกับสัญญาณอ่านกลับที่ปราศจากสัญญาณรบกวนใดๆ โดยรายละเอียดการทำงานของวงจรเข้ารหัส M-RBP ที่นำเสนอในบทความนี้สามารถอธิบายได้ตามรูปที่ 3 ดังนี้

ขั้นตอนที่ 1 ให้พิจารณาข้อมูล 3 แทร็กแล้วทำการเลื่อนหน้าต่างขนาด 3×3 ที่มีศูนย์กลางอยู่ที่แทร็กที่ *l* (แทร็กกลาง) จากตำแหน่งเริ่มต้น (ด้านซ้ายสุดของแทร็ก) ไปยังตำแหน่งสุดท้าย (ด้านขวาสุดของแทร็ก) แล้วนับจำนวนรูปแบบข้อมูลแบบลดทอนทั้งหมดที่เกิดขึ้นโดยการเปรียบเทียบกับรูปแบบข้อมูลในตารางที่ 1 จากนั้นกำหนดให้แทร็กที่ *l* คงที่ แล้วเลื่อนวนในแทร็กที่ *l*-1 และแทร็กที่ *l*+1 เพื่อสร้างรูปแบบข้อมูลที่เป็นไปได้ทั้งหมด ซึ่งในแต่ละครั้งของการเลื่อนวนในแทร็กที่ *l*-1 หรือแทร็กที่ *l*+1 ก็จะทำซ้ำกระบวนการเลื่อนหน้าต่างขนาด 3×3 จากตำแหน่งเริ่มต้นไปยังตำแหน่งสุดท้ายของแทร็กอีกครั้ง เพื่อนับจำนวนรูปแบบข้อมูลแบบลดทอนทั้งหมดที่เกิดขึ้น โดยเมื่อทำการเลื่อนวนข้อมูลทั้งหมดแล้ว จะตรวจสอบว่าการเลื่อนวนของแทร็กที่ *l*-1 และแทร็กที่ *l*+1 ครั้งใดที่มีผลทำให้ได้รูปแบบข้อมูลที่ดีสุด (นั่นคือมีจำนวนรูปแบบข้อมูลแบบลดทอนน้อยสุด) จากนั้นก็จะบันทึกจำนวนครั้งที่ใช้ในการเลื่อนวนของข้อมูลชุดนั้นไว้ในหน่วยความจำหรือบัฟเฟอร์ เพื่อนำไปใช้ในการเลื่อนวนข้อมูลกลับมายังตำแหน่งเดิม ณ วงจรถอดรหัส M-RBP ที่ภาครับ จากขั้นตอนที่ 1 ซึ่งเป็นวิธีการของการจัดรูปแบบของ RBP จะเห็นว่ามีการจัดรูปแบบเฉพาะแทร็กกลาง แทร็กบน และแทร็กล่างที่อยู่ติดกันเท่านั้น ซึ่งยังมีแทร็กอื่นๆที่นอกเหนือจากทั้งสามแทร็กที่ต้องการการจัดรูปแบบข้อมูลก่อนการบันทึกลงไปในสื่อบันทึก ดังนั้นขั้นตอนที่ 2 ซึ่งเป็นขั้นตอนที่เพิ่มเติมจากวิธี RBP เพื่อหลีกเลี่ยงการใช้แถบป้องกันส่งผลให้พื้นที่การบันทึกข้อมูลเพิ่มมากขึ้นและยังช่วยให้สมรรถนะของระบบเพิ่มมากขึ้นอีกด้วย โดยสามารถอธิบายในรายละเอียดได้ดังนี้



รูปที่ 3 วิธีการ M-RBP ที่นำเสนอ (a) ขั้นตอนที่ 1 และ (b) ขั้นตอนที่ 2

ตารางที่ 1 รูปแบบข้อมูลที่ก่อให้เกิดการแทรกสอดสองมิติแบบลดทอน

Destructive Pattern Number	Upper Track			Center Track			Lower Track		
	$(k\text{-}1)^{th}$	k^{th}	$(k\text{+}1)^{th}$	$(k\text{-}1)^{th}$	k^{th}	$(k\text{+}1)^{th}$	$(k\text{-}1)^{th}$	k^{th}	$(k\text{+}1)^{th}$
1	1	1	1	1	-1	1	1	1	1
2	-1	1	1	1	-1	1	1	1	1
3	1	1	-1	1	-1	1	1	1	1
4	-1	1	-1	1	-1	1	1	1	1
5	1	1	1	-1	-1	1	1	1	1
6	1	1	1	1	-1	-1	1	1	1
7	1	1	1	1	-1	1	-1	1	1
8	-1	1	1	1	-1	1	-1	1	1
9	1	1	-1	1	-1	1	-1	1	1
10	-1	1	-1	1	-1	1	-1	1	1
11	-1	-1	1	-1	1	-1	1	-1	1
12	1	-1	-1	-1	1	-1	1	-1	1
13	-1	-1	-1	-1	1	-1	1	-1	1
14	1	-1	1	-1	1	-1	-1	-1	1
15	-1	-1	1	-1	1	-1	-1	-1	1
16	1	-1	-1	-1	1	-1	-1	-1	1
17	-1	-1	-1	-1	1	-1	-1	-1	1
18	1	1	1	1	-1	1	1	1	-1
19	-1	1	1	1	-1	1	1	1	-1
20	1	1	-1	1	-1	1	1	1	-1
21	-1	1	-1	1	-1	1	1	1	-1
22	1	1	1	1	-1	1	-1	1	-1
23	-1	1	1	1	-1	1	-1	1	-1
24	1	1	-1	1	-1	1	-1	1	-1
25	1	-1	1	-1	1	-1	1	-1	-1
26	-1	-1	1	-1	1	-1	1	-1	-1
27	1	-1	-1	-1	1	-1	1	-1	-1
28	-1	-1	-1	-1	1	-1	1	-1	-1
29	-1	-1	-1	-1	1	1	-1	-1	-1
30	-1	-1	-1	1	1	-1	-1	-1	-1
31	1	-1	1	-1	1	-1	-1	-1	-1
32	-1	-1	1	-1	1	-1	-1	-1	-1
33	1	-1	-1	-1	1	-1	-1	-1	-1
34	-1	-1	-1	-1	1	-1	-1	-1	-1

ขั้นตอนที่ 2 ขั้นตอนนี้จะใช้หน้าต่างขนาด 3×3 จำนวน 2 ชุดที่เป็นอิสระต่อกัน โดยมีตำแหน่งศูนย์กลางของหน้าต่างอยู่ที่แทร็ก l-1 และ l+1 ตามลำดับ ตามรูปที่ 3(b) จากนั้นกำหนดให้ข้อมูลในแทร็กที่ l-1, l และ l+1 คงที่แล้วทำการเลื่อนวนแทร็กที่ l-2 และ l+2 เพื่อหาจำนวนครั้งในการเลื่อนวนที่ทำให้ได้รูปแบบข้อมูลที่ดีสุด (เช่นเดียวกับขั้นตอนที่ 1) ของชุดข้อมูลในแทร็ก {l-2, l-1 และ l} และแทร็ก {l, l+1 และ l+2} ตามลำดับ เมื่อทำขั้นตอนที่ 2 เสร็จแล้ว ก็จะได้รูปแบบข้อมูลที่ดีสุดทั้ง 5 แทร็ก ที่พร้อมจะถูกบันทึกลงในสื่อบันทึก และจำนวนครั้งที่ใช้ในการเลื่อนวนที่ดีสุดของแทร็ก l-2 และ l+2 ก็จะถูกบันทึกลงในหน่วยความจำเช่นเดียวกัน ในกรณีที่ข้อมูลมีมากกว่า 5 แทร็ก ก็ยังคงสามารถใช้วิธีการนี้ได้เช่นกัน ดำเนินการในลักษณะเดียวกันกับขั้นตอนที่ 2 ซ้ำจนกว่าจะครบทุกแทร็ก โดยไม่จำเป็นต้องมีแถบป้องกันคั่นระหว่างแทร็ก ซึ่งถือว่าเป็นข้อดีอีกประการหนึ่งของวิธีการที่เราได้นำเสนอเพื่อช่วยเพิ่มพื้นที่ในการจัดเก็บข้อมูลให้มากขึ้นด้วย

ในทางปฏิบัติอัลกอริทึม M-BPR ที่นำเสนอจะใช้เวลาในการประมวลผลค่อนข้างมาก โดยเฉพาะอย่างยิ่งเมื่อแทร็กข้อมูลมีความยาวมาก อย่างไรก็ตามถ้าต้องการลดเวลาในการประมวลผลเพื่อให้การประมวลผลรวดเร็วขึ้นก็สามารถทำได้ด้วยการแบ่งแทร็กข้อมูลออกเป็นส่วนย่อยๆ ในงานวิจัยนี้ได้แบ่งลำดับข้อมูล $\{a_k\}$ ทั้งหมด 6720 บิต ออกเป็น 5 แทร็ก แต่ละแทร็กมีความยาว L_t = 1344 บิต จากนั้นจึงแบ่งข้อมูลขนาด 5×1344 (6720 บิต) ออกเป็น 21 ส่วน โดยแต่ละส่วนมีขนาด 5×64 บิต เพื่อลดเวลาในการประมวลผล ด้วยวิธีการแบ่งส่วนย่อยนี้ทำให้สามารถคำนวณหาขนาดของหน่วยความจำที่ต้องใช้ในการเก็บค่าจำนวนครั้งที่ใช้ในการเลื่อนวนได้เท่ากับ $n_p 4\log_2(n_b)$ บิต โดย $n_b = L_t/n_p$ คือขนาดของส่วนย่อยที่ถูกแบ่ง (บิต), n_p คือ

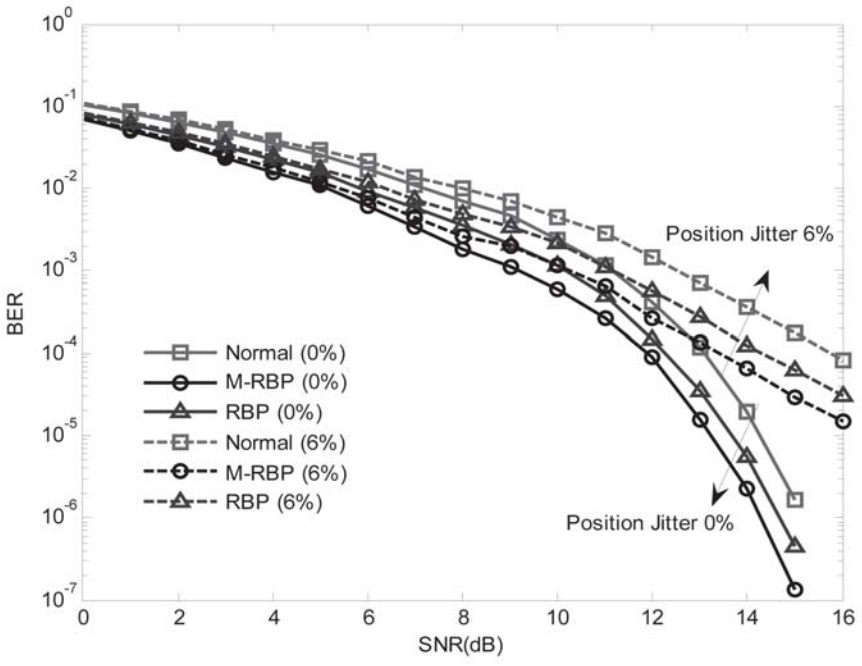
จำนวนส่วนย่อยที่ถูกแบ่ง, และ n_t คือจำนวนแทร็ก (ในงานวิจัยนี้ใช้ $n = 5$ แทร็ก) ดังนั้นหน่วยความจำที่ใช้ทั้งหมดจึงมีค่าเท่ากับ 6×4×21 = 504 บิตต่อข้อมูล 6720 บิต

4. ผลการจำลองระบบและการอภิปรายผล

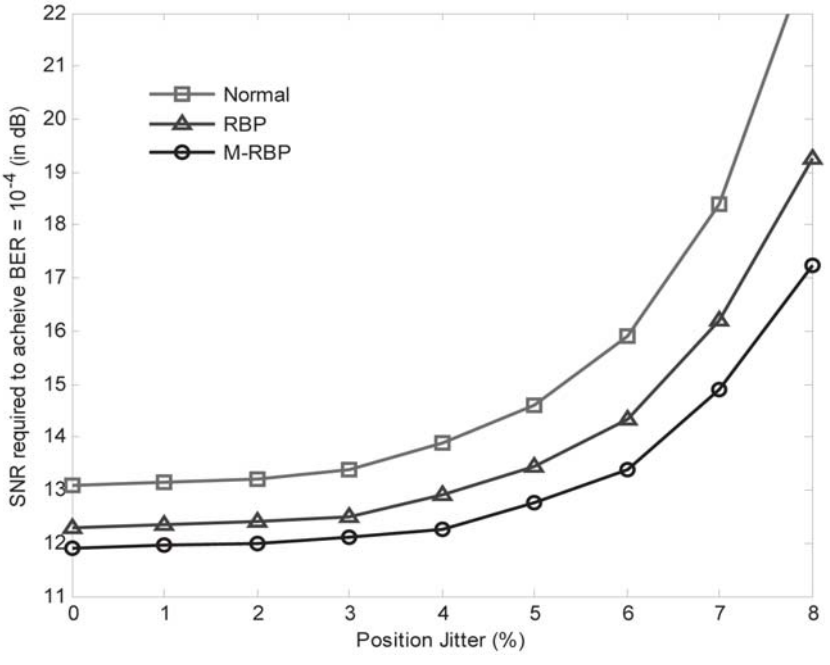
ในการทดสอบอัลกอริทึม M-RBP ที่นำเสนอในระบบ บีพีเอ็มอาร์ ที่ความหนาแน่นของการบันทึกข้อมูล 2.5 Tb/in^2 ซึ่งมีความกว้างของบิตและความกว้างของแทร็กเท่ากับ 16 นาโนเมตร (nm), PW_{50} ในแนวตามแทร็กกว้างเท่ากับ 19.4 nm และ PW_{50} ในแนวขวางแทร็กกว้างเท่ากับ 24.8 nm เช่นเดียวกับ (8) โดยอัตราส่วนของสัญญาณต่อสัญญาณรบกวนนิยามโดย SNR = $20\log_{10}(1/\sigma)$ มีหน่วยเป็นเดซิเบล (decibel: dB) เมื่อ σ คือค่าเบี่ยงเบนมาตรฐานของสัญญาณรบกวนเกาส์เซียนสีขาว นอกจากนี้ระบบ บีพีเอ็มอาร์ จะใช้ทาร์เก็ต (target) แบบสองมิติขนาด 3×3 และอีควอไลเซอร์สองมิติขนาด 3×7 ซึ่งถูกออกแบบโดยวิธีการหาค่าผิดพลาดกำลังสองเฉลี่ยน้อยสุด (minimum mean-squared error) (6, 9) ที่ทำให้ระบบเกิดอัตราบิตผิดพลาด (bit-error rate: BER) เท่ากับ 10^{-4} โดยในการหาค่าอัตราข้อผิดพลาดบิตของแต่ละค่า SNR จะต้องเกิดบิตผิดพลาดอย่างน้อย 500 บิต และกำหนดให้ข้อมูล 1 เซกเตอร์ (sector) มี 6720 บิต

รูปที่ 4 แสดงอัตราบิตผิดพลาดของระบบ บีพีเอ็มอาร์ ที่มีสัญญาณรบกวนแบบเลื่อนตำแหน่งเป็น s_j = 0% และ s_j = 6% ในที่นี้ “Normal”, “M-RBP” และ “RBP” ใช้แทนกรณีปรกติในการบันทึกข้อมูล, กรณีที่ใช้อัลกอริทึม M-RBP ในการลดจำนวนของรูปแบบข้อมูลแบบลดทอนให้น้อยที่สุด, และกรณีที่ใช้อัลกอริทึม RBP ที่นำเสนอใน (5) ตามลำดับ โดยเครื่องหมาย “%” หมายถึง ร้อยละของความผิดพลาดของการเลื่อนตำแหน่งข้อมูล จากรูปจะพบว่ากรณีที่ระบบไม่มีความผิดพลาดของตำแหน่ง (s_j = 0%) ณ ค่า BER = 10^{-4} ระบบที่ใช้อัลกอริทึม M-RBP จะมีสมรรถนะดีกว่าระบบที่ใช้วิธีการบันทึกแบบปรกติและระบบที่ใช้อัลกอริทึม RBP ประมาณ 0.6 เดซิเบล และ 0.3 เดซิเบล ตามลำดับ นอกจากนี้เมื่อระบบมีความผิดพลาดของตำแหน่งเพิ่มมากขึ้น เช่น s_j = 6% ก็จะพบว่าระบบที่นำเสนอมีสมรรถนะดีมากยิ่งขึ้น (มากกว่าระบบที่ใช้วิธีการบันทึกแบบปรกติและระบบที่ใช้อัลกอริทึม RBP ประมาณ 1 เดซิเบล และ 0.5 เดซิเบล ตามลำดับ) แสดงว่าวิธีการที่นำเสนอสามารถลดผลกระทบที่เกิดจากการแทรกสอดสองมิติได้เป็นอย่างดี นอกจากนี้ยังได้เปรียบเทียบสมรรถนะของระบบในรูปของค่า SNR ที่ระบบใช้ที่ทำให้เกิด BER = 10^{-4} ณ ระดับความผิดพลาดของการเลื่อนตำแหน่งข้อมูลต่างๆ ดังแสดงในรูปที่ 5 ซึ่งจะเห็นได้ว่าระบบที่ใช้อัลกอริทึม M-RBP มีสมรรถนะดีกว่าระบบอื่นๆ โดยเฉพาะอย่างยิ่งเมื่อความผิดพลาดของการเลื่อนตำแหน่งข้อมูลมีความรุนแรงเพิ่มมากขึ้น ทั้งนี้เนื่องจากวิธีการแบบ M-RBP สามารถลดรูปแบบข้อมูลแบบหักล้างได้ในทุกแทร็กข้อมูลที่ต้องการบันทึกลงไปยังสื่อบันทึกข้อมูลซึ่งแตกต่างจากวิธี RBP ที่สามารถลดรูปแบบข้อมูลแบบหักล้างได้เฉพาะแทร็กกลางเท่านั้น ทำให้รูปแบบของข้อมูลที่ไม่เหมาะสมต่อการบันทึกยังคงหลงเหลืออยู่ เมื่อถูกบันทึกลงไปจึงทำให้ระบบการตรวจจับข้อมูลเกิดข้อผิดพลาดได้โดยง่าย ส่งผลให้สมรรถนะของระบบลดลง

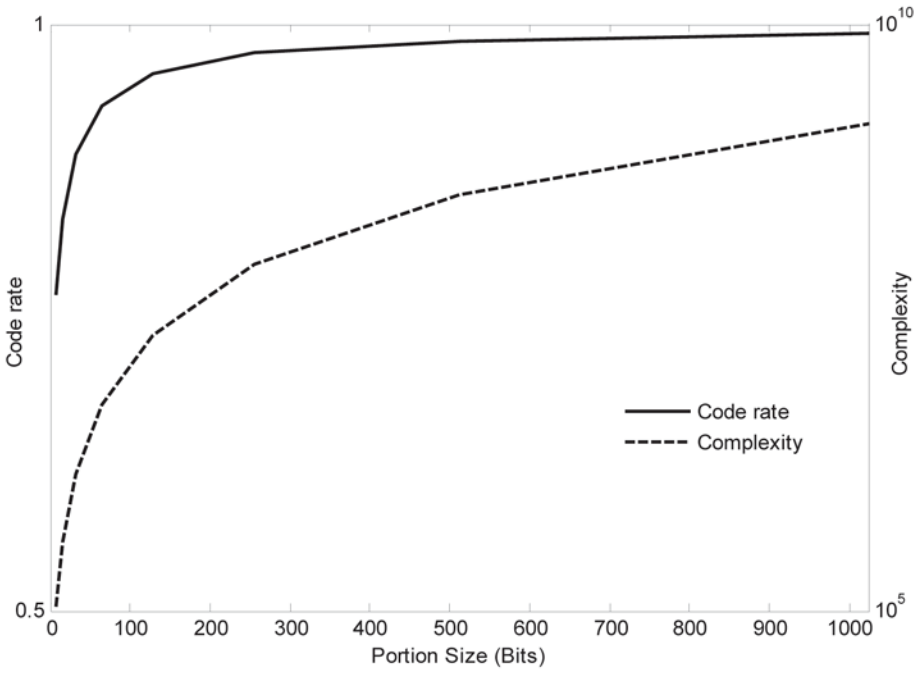
ในส่วนต่อไปนี้จะแสดงสมรรถนะของอัลกอริทึม M-RBP ที่นำเสนอในรูปของความซับซ้อน (complexity) และอัตรารหัส (code rate) ที่ใช้ เมื่อส่วนย่อยของข้อมูลมีขนาดกว้าง 8, 16, 32, 64, 128, 256, 512 และ 1024 บิต โดยในที่นี้ความซับซ้อน C ของอัลกอริทึม M-RBP จะพิจารณาจากปริมาณหน่วยความจำที่ต้องใช้ในการเก็บค่าจำนวนครั้งที่ใช้ในการเลื่อนวนแต่ละแทร็ก โดยในขั้นตอนที่ 1 จะได้ความซับซ้อน $C_1 = n_b \times n_b \times n_b \times n_p = n_b \times n_b \times L_t$ บิต (นับจากการเลื่อนแทร็กบน การเลื่อนแทร็กล่าง และการเลื่อนหน้าต่างเพื่อนับจำนวนรูปแบบข้อมูลแบบลดทอนในทุกช่วงข้อมูลที่ถูกแบ่ง) และในขั้นตอนที่ 2 จะได้ความซับซ้อน $C_2 = 2 \times n_b \times n_b \times n_p = 2 \times n_b \times L_t$ หน่วย (เกิดจากการเลื่อนหน้าต่างสองชุดและเลื่อนข้อมูลใน 2 แทร็กที่อิสระต่อกันของทุกช่วงข้อมูลที่ถูกแบ่ง) ดังนั้นอัลกอริทึม M-RBP จะมีความซับซ้อนรวมทั้ง 2 ขั้นตอนเท่ากับ $C = C_1 + C_2 = (2+n_b) \times n_b \times L_t$ หน่วย นอกจากนี้อัตรารหัสคำนวณหาได้จาก $R = (n_t \times n_b)/(n_t \times n_b + 4\log_2(n_b))$ ตารางที่ 2 และรูปที่ 6 แสดงความซับซ้อนและอัตรารหัสของอัลกอริทึม M-RBP เมื่อเทียบกับขนาดของส่วนย่อยข้อมูลที่ถูกแบ่ง ซึ่งจะเห็นได้ชัดว่าเมื่อขนาดของส่วนย่อยข้อมูลมีค่าเพิ่มขึ้น ความซับซ้อนและอัตรารหัสก็จะเพิ่มขึ้น



รูปที่ 4 อัตราข้อผิดพลาดบิตของระบบที่มีสัญญาณรบกวนแบบเลื่อนตำแหน่งที่ 0 และ 6 %



รูปที่ 5 สมรรถนะของระบบต่างๆ ในรูปของค่า SNR ที่ระบบต้องการเพื่อทำให้ได้ BER = 10^{-4} ณ ระดับความรุนแรงของความผิดพลาดของการเลื่อนตำแหน่งข้อมูล



รูปที่ 6 ความซับซ้อนและอัตรารหัส เมื่อหนึ่งแทร็กข้อมูลมีความกว้างเท่ากับ 1344 บิต

ตารางที่ 2 ความซับซ้อนและอัตรารหัสเทียบกับขนาดของส่วนย่อยข้อมูลที่ถูกแบ่ง เมื่อหนึ่งแทร็กข้อมูลมีความกว้างเท่ากับ 1344 บิต

Portion sizes	Complexity	Complexity, C	Code rate, R
		L_t = 1344	L_t = 1344
8	80 L_t	107520	0.7692
16	288 L_t	387072	0.8333
32	1088 L_t	1462272	0.8888
64	**4224 L_t**	**5677056**	**0.9302**
128	16640 L_t	22364160	0.9580
256	66048 L_t	88768512	0.9756
512	263168 L_t	353697792	0.9861
1024	1050624L_t	1412038656	0.9922

5. สรุป

บทความนี้ได้เสนอวิธีการจัดเรียงรูปแบบข้อมูลแบบหลายแทร็กหรือ M-RBP เพื่อทำการจัดเรียงข้อมูลก่อนการบันทึกลงในสื่อบันทึกในระบบบีพีเอ็มอาร์ ซึ่งการจัดเรียงข้อมูลก่อนการบันทึกจะช่วยลดผลกระทบที่เกิดจากการแทรกสอดสองมิติที่เกิดขึ้นในระบบการบันทึกข้อมูลเชิงแม่เหล็กได้แนวคิดสำหรับวิธี M-RBP ก็คือการหลีกเลี่ยงรูปแบบข้อมูลที่ส่งผลทำให้เกิดการลดทอนของสัญญาณอ่านกลับซึ่งอาจทำให้เกิดข้อผิดพลาดได้ง่ายที่วงจรตรวจหาโดยทำการเปลี่ยนรูปแบบข้อมูลให้เหมาะสมก่อนที่จะทำการบันทึกข้อมูลลงในสื่อบันทึกข้อมูล ผลการทดลองแสดงให้เห็นว่าวิธีที่เราได้นำเสนอสามารถช่วยเพิ่มสมรรถนะของระบบได้เป็นอย่างมาก โดยเฉพาะอย่างยิ่งเมื่อระบบต้องเผชิญกับความผิดพลาดของการเลื่อนตำแหน่งข้อมูลที่รุนแรง อย่างไรก็ตามถึงแม้ว่าวิธีที่นำเสนอจะมีสมรรถนะ

ดีและใช้หน่วยความจำเพิ่มเติมน้อยแต่ก็ต้องแลกมาด้วยเวลาที่ใช้ในการประมวลผลที่เพิ่มมากขึ้นด้วย ดังนั้นงานวิจัยต่อไปในอนาคตสิ่งที่ต้องพัฒนาก็คือการลดเวลาในการประมวลผลและหน่วยความจำให้น้อยลงต่อไป

6. กิตติกรรมประกาศ

งานวิจัยนี้ได้รับเงินงบประมาณสนับสนุนภายใต้ทุนโครงการวิจัยเงินรายได้ รหัสทุน 2556-02-10003 จากวิทยาลัยนวัตกรรมการจัดการข้อมูล สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

7. เอกสารอ้างอิง

(1) W. Chang and J. R. Cruz, “Inter-Track Interference Mitigation for Bit-Patterned Magnetic Recording,” *IEEE Trans. Magn.*, vol. 46, no. 11, pp. 3899–3908, Nov. 2010.

(2) Y. Kurihara, Y. Takeda, Y. Takaishi, Y. Koizumi, H. Osawa, M.Z. Ahmed, and Y. Okamoto, “Constructive ITI-coded PRML system based on a two-track model for perpendicular magnetic recording,” Journal of Magnetism and Magnetic Materials 320, pp. 3140-3143, Aug. 2008.

(3) J. P. J. Groenland and L. Abelmann, “Two dimentional Coding for Probe Recording on Magnetic Patterned Media,” *IEEE Trans. Magn.*, vol. 43, no. 6, pp. 2307–2309, Jun. 2007.

(4) X. Shao, L. Alink, J.P.J. Groenland, L. Abelmann, and C. H. Slump, “A Simple Two-Dimensional Coding Scheme for Bit Patterned Media,” *IEEE Trans. Magn.*, vol. 47, no. 10, pp. 2559-2562, Oct. 2011.

(5) A. Arrayangkool, C. Warisarn, Lin M. M. Myint, and P. Kovintavewat, “A Simple Recorded-Bit Patterning Scheme for Bit Patterned Media Recording,” *in Proc. of ECTI-CON 2013*, Krabi, May 15-17, 2013.

(6) S. Nabavi, B. V. K. Vijaya Kumar, and J. A. Bain, Signal processing for bit-patterned media channel with inter-track interference, Ph.D. dissertation, Dept. Elect.Eng. Comp. Sci., Carnegie Mellon University, Pittsburgh, PA, 2008.

(7) S. Koonkarnkhai, N. Chirdchoo, and P. Kovintavewat, “Iterative Decoding for High-Density Bit-Patterned Media Recording,” Procedia Engineering 32, pp.323-328, Nov. 2012.

(8) S. Nabavi *et al.*, “Application of Image Processing to Characterize Patterning Noise in Self-assembled Nano-masks for Bit-Patterned media,” *IEEE Trans. Magn.*, vol. 45, no. 10, pp. 3523–3527, Oct. 2009.

(9) J. Moon and W. Zeng, “Equalization for maximum like-lihood detector,” *IEEE Trans. Magn*., vol. 31, no. 2, pp. 1083–1088, Mar. 1995.