โครงการวิจัยย่อยลำดับที่ 18

เรื่อง การพัฒนาสื่อสำหรับการเรียนรู้ระบบสื่อสาร ปีที่ 3

1. ผู้รับผิดชอบโครงการ รองศาสตราจารย์ ดร.ลัญฉกร วุฒิสิทธิกุลกิจ

2. วัตถุประสงค์ของโครงการ

เพื่อพัฒนาสื่อการสอนที่มีโปรแกรมซอฟต์แวร์จำลองการทำงานของระบบเพื่อใช้ประกอบการเรียนรู้

3. ขอบเขตหรือเป้าหมายของโครงการ

ส่วนที่ 1

1) ศึกษาและพัฒนาระบบเข้ารหัสและถอดรหัสเทอร์โบสำหรับการบันทึกข้อมูลบนตัวกลางแม่เหล็ก
2) พัฒนาโปรแกรมคอมพิวเตอร์จำลองระบบที่ศึกษาเพื่อใช้เป็นสื่อการสอนประกอบการเรียนรู้

ส่วนที่ 2

เพื่อพัฒนาอัลกอริทึมการจัดกำหนดการสำหรับการจัดสรรบริการในเซลล์สวิตช์ที่พิจารณาลำดับความสำคัญของเซลล์ และสามารถควบคุมคุณภาพการให้บริการโดยจะแสดงออกมาในรูปของอัตราส่วนเวลาประวิงของเซลล์คลาส 1 ต่อเซลล์คลาส 2 ให้ได้ตามความต้องการอย่างแม่นยำ โดยที่จะจำกัดขอบเขตของวิทยานิพนธ์ ดังต่อไปนี้

1) สวิตช์เป็นแบบเซลล์สวิตช์
2) เป็นสวิตช์ที่มีการวางบัฟเฟอร์ที่พอร์ตอินพุตของสวิตช์
3) ทราฟฟิกเป็นแบบ โดยการมาถึงของแพ็กเกตเป็นแบบ uniform i.i.d.Bernoulli processes และแพ็กเกตมีการแจกแจงที่จะไปยังเอาท์พุตใดๆมีความน่าจะเป็นเท่ากัน
4) ทราฟฟิกมี 2 คลาส คือ high priority และ low priority

4. ส่วนงานที่ได้ดำเนินการไปแล้ว

ส่วนที่ 1

1) ศึกษาการบันทึกข้อมูลบนตัวกลางแม่เหล็ก
2) ศึกษาการนำการเข้ารหัสเทอร์โบไปประยุกต์ใช้งานสำหรับการบันทึกข้อมูลบนตัวกลางแม่เหล็ก
3) พัฒนาระบบการเข้ารหัสและถอดรหัสเทอร์โบที่ให้สมรรถนะสูงสำหรับการบันทึกข้อมูลบนตัวกลางแม่เหล็ก

4) พัฒนาโปรแกรมคอมพิวเตอร์เพื่อจำลองการทำงานของระบบที่พัฒนาขึ้น

5) ทดสอบสมรรถนะของระบบที่เสนอด้วยโปรแกรมที่พัฒนาขึ้น

ส่วนที่ 2

1) ทำการศึกษาสถาปัตยกรรมสวิตช์ที่ใช้ในระบบโครงข่ายสื่อสารในปัจจุบัน

2) ทำการศึกษาอัลกอริทึมการจัดลำดับเซลล์ ISLIP ที่มีการวนรอบ 1 รอบ (one iteration)และแบบ multiple iteration

3) ทำการศึกษาอัลกอริทึมการจัดลำดับเซลล์ ISLIP ที่มีการวนรอบ 1 รอบ (one iteration)และแบบ multiple iteration

4) ทำการศึกษาอัลกอริทึมการจัดลำดับเซลล์ Prioritized ISLIP Algorithm

5) ทดลองเขียนโปรแกรมของอัลกอริทึม Prioritized ISLIP โดยใช้โปรแกรม matlab

6) ทำการวิเคราะห์สมรรถนของอัลกอริทึม ISLIP และ Prioritized ISLIP และเปรียบเทียบผลการทดลองที่ได้จากอัลกอริทึม Prioritized ISLIP

5. รายละเอียดงานวิจัยส่วนที่ 1

5.1 แนวเหตุผล

ในปัจจุบันการเรียนการสอนทางด้านวิศวกรรมไฟฟ้าสื่อสาร โดยส่วนใหญ่จะเน้นทฤษฎีหรือหลักการเป็นหัวใจสำคัญ อย่างไรก็ตามหากนิสิตได้มีโอกาสทดสอบแนวคิดทฤษฎีเหล่านี้ในรูปแบบของอุปกรณ์การทดลองหรือสื่อการสอนด้วยคอมพิวเตอร์ จะสามารถเพิ่มประสิทธิภาพการเรียนการสอนให้ดีขึ้นได้ แต่เนื่องจากสื่อการสอนที่มีอยู่ในปัจจุบันส่วนใหญ่มาจากต่างประเทศ มีราคาแพง จึงทำให้ประเทศต้องเสียงบประมาณไปกับสื่อการสอนของต่างชาติเป็นจำนวนมาก ดังนั้นการพัฒนาสื่อการเรียนการสอนขึ้นเองภายในประเทศ จะช่วยประหยัดค่าใช้จ่ายได้อย่างมาก อีกทั้งเป็นการกระตุ้นให้เกิดการพัฒนาองค์ความรู้ได้เองในประเทศ ลดการพึ่งพิงทางเทคโนโลยีจากต่างประเทศ และเพิ่มประสิทธิภาพการเรียนการสอนได้อย่างเต็มประสิทธิภาพ อีกทั้งความรู้และประสบการณ์ที่ได้รับจากการพัฒนาจะเป็นพื้นฐานสำคัญต่อการเรียนรู้ในเชิงปฏิบัติและเพิ่มศักยภาพในการประยุกต์ใช้งานในภาคอุตสาหกรรมให้ดียิ่งขึ้น

ผู้วิจัยได้เลือกศึกษาระบบการบันทึกข้อมูลบนตัวกลางแม่เหล็กที่ใช้การเข้ารหัสเทอร์โบเพื่อแก้ไขความผิดพลาดในการอ่านข้อมูล ด้วยเหตุที่ว่า ระบบการบันทึกข้อมูลบนตัวกลางแม่เหล็กดังที่สามารถแสดงได้ดังรูปที่ 1 นั้นสามารถพิจารณาได้ว่าเป็นระบบสื่อสารอย่างหนึ่ง ดังนั้นทฤษฎีของช่องสัญญาณและทฤษฎีการเข้ารหัสแก้ไขความผิดพลาดจึงสามารถนำมาประยุกต์ใช้งานได้ เมื่อผู้วิจัยพัฒนาสื่อการเรียนการสอนในรูปเอกสารและโปรแกรมคอมพิวเตอร์สำหรับระบบดังกล่าวสำเร็จ สื่อดังกล่าวนี้จะมีส่วนช่วยให้นักศึกษาหรือผู้เรียนเห็นภาพได้ว่าการประยุกต์ทฤษฎีระบบสื่อสารมิได้จำกัดเฉพาะกับระบบสื่อสารที่เป็นที่รู้จักเท่านั้น แต่ยังสามารถนำไปประยุกต์ใช้กับระบบการบันทึกข้อมูลได้อีกด้วย

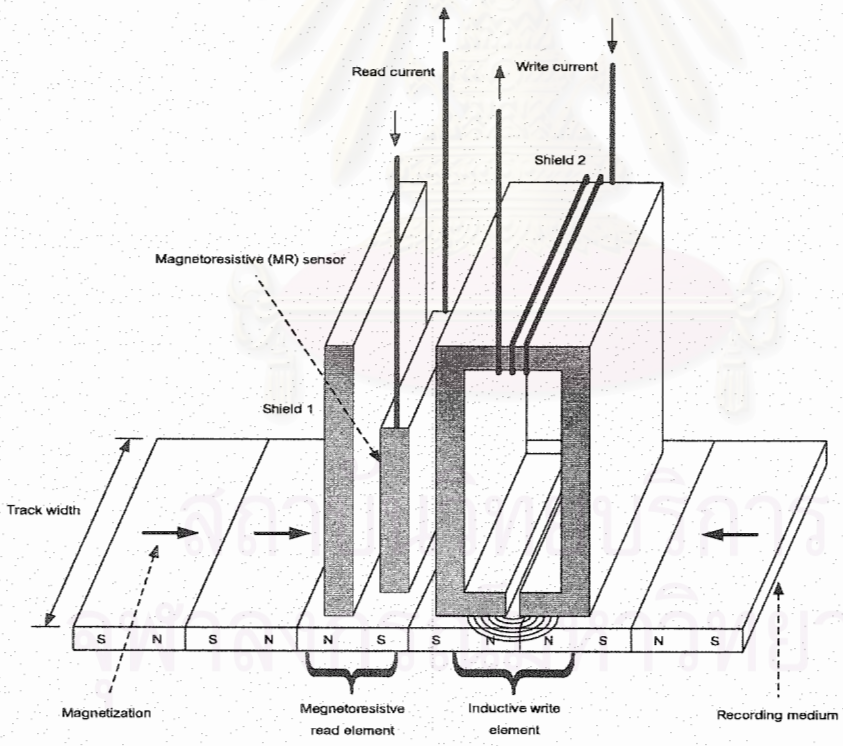
5.2 เอกสารงานวิจัยที่เกี่ยวข้อง

ในอุปกรณ์ฮาร์ดดิสก์ที่นิยมใช้กันอยู่ในปัจจุบันนั้น กระบวนการเขียนและอ่านข้อมูลกลับคืนระหว่างหัวอ่าน/เขียน และตัวกลางแม่เหล็ก สามารถจำลองด้วยแบบจำลองช่องสัญญาณลอเรนซ์ (Lorentzian channel model) [2] ซึ่งสมมุติว่าเอาท์พุทของช่องสัญญาณมีผลตอบสนองต่อสัญญาณต่อฟังก์ชันขั้นบันได (step response) เป็นสัญญาณพัลส์ลอเรนซ์ดังสมการ

$$s(t) = \frac{1}{1+(2t/PW_{50})^2} \qquad (1)$$

โดยที่ PW_{50} แทนความกว้างของพัลส์ที่ 50% ของจุดสูงสุด

ด้วยเหตุที่รูปทรงของสัญญาณพัลส์ดังกล่าวไม่ได้จำกัดอยู่ในช่วงเวลาของบิตข้อมูลของมันเอง แต่ยังแผ่กว้างออกไปซ้อนเกยกับสัญญาณของบิตอื่น ๆ หลายบิต จึงทำให้เกิดปัญหาการแทรกสอดระหว่างสัญลักษณ์ (Intersymbol Interference) ขึ้นอย่างหลีกเลี่ยงไม่ได้ โดยเฉพาะเมื่อใช้ความหนาแน่นในการบันทึก (recording density) สูง ถึงแม้ว่าจะสามารถใช้อีควอไลเซอร์ (equalizer) ที่ทำงานโดยอาศัยการประมาณลำดับแบบความน่าเป็นจริงสูงสุด (maximum-likelihood sequence estimation) เพื่อที่จะขจัดการแทรกสอดระหว่างสัญลักษณ์ได้ ก็พบว่าอีควอไลเซอร์ที่ใช้มีความซับซ้อนสูงมาก



รูปที่ 1 ระบบการอ่านและเขียนบนตัวกลางแม่เหล็ก [1]

H. Kobayashi และ D. T. Tang [3] ได้เสนอให้ใช้หลักการสัญญาณตอบสนองบางส่วน (partial response (PR) signaling) [4] เพื่อมาช่วยลดความซับซ้อนในการตรวจวัด (detection) ลงโดยการปรับแต่ง (equalize) สัญญาณที่อ่านได้จากหัวอ่านด้วยอีควอไลเซอร์ (equalizer) เพื่อให้มีผลตอบสนองต่ออิมพัลส์ (impulse response) เป็นไปตาม ผลตอบสนองเป้าหมาย (target response) จากนั้นจึงสามารถนำสัญญาณที่ได้ไปถอดรหัส ด้วยตัวตรวจวัดแบบอัลกอริทึมวิเทอร์บิ (Viterbi Algorithm (VA) detector) [5]–[6] ซึ่งทำให้ความซับซ้อนที่ต้องใช้ลดลงไปอย่างมาก โดยเราเรียกกระบวนการนี้ว่า การสัญญาณตอบสนองบางส่วนที่อาศัยการประมาณลำดับแบบความน่าเป็นจริงสูงสุด (PRML: partial response signaling using maximum-likelihood sequence estimation)

ในเอกสารอ้างอิงที่ [7] Thapar และ Partel ได้ขยายงานในเอกสารอ้างอิงที่ [5] และ [6] ออกไปโดยการเสนอชุดของสัญญาณในรูปของพหุนาม $(1-D)(1+D)^n$ เมื่อ n=1, 2, 3,... โดยที่ n จะเพิ่มขึ้นเมื่อความหนาแน่นในการบันทึกข้อมูลมีค่าสูงขึ้น พจน์ $(1-D)$ เป็นส่วนที่พิจารณถึงว่าไม่มีองค์ประกอบ DC เกิดขึ้นระหว่างกระบวนการเขียนและอ่านข้อมูล ในขณะที่พจน์ $(1+D)^n$ เป็นส่วนที่พิจารณาถึงการลดทอนในช่วงความถี่สูงของช่วงสัญญาณซึ่งมาจากการสูญเสียที่เกิดจากช่องว่างระหว่างหัวอ่านและตัวกลางแม่เหล็ก และความกว้างที่จำกัดของการเขียนข้อมูลบนตัวกลางแม่เหล็ก [8] เมื่อ $n=0$ ช่องสัญญาณนี้จะถูกเรียกว่า ช่องสัญญาณแบบไดโคด (dicode) และเมื่อ $n=1$ ช่องสัญญาณที่ได้จะรู้จักกันในชื่อ ช่องสัญญาณแบบ PR4 (class-4 partial response (PR4) channel) และเมื่อ $n=2$ ช่องสัญญาณนี้คือช่องสัญญาณแบบ EPR4 (extended PR4 channel)

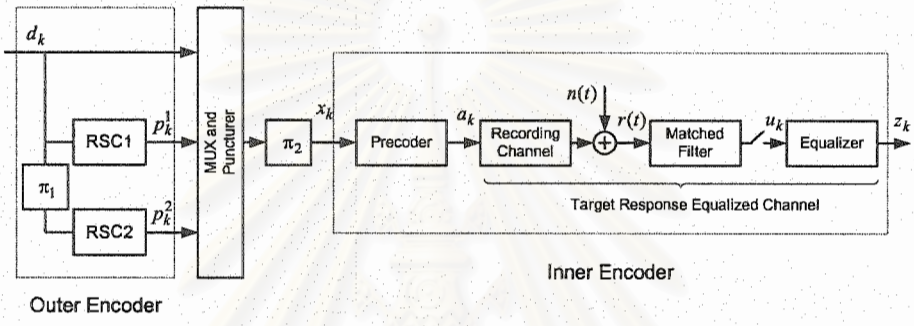
ด้วยเหตุที่ การตรวจวัดแบบ PRML สามารถขจัดปัญหาการแทรกสอดระหว่างสัญลักษณ์ได้ดี จึงทำให้ การตรวจวัดแบบนี้ใช้กันอย่างแพร่หลายในอุปกรณ์บันทึกข้อมูลบนตัวกลางแม่เหล็กตั้งแต่ปี ค.ศ. 1990 [9] แต่อย่างไรก็ดีเมื่อนำหลักการนี้ไปใช้กับกรณีที่ต้องบันทึกข้อมูลด้วยความหนาแน่นสูงก็ทำให้ต้องใช้การปรับแต่งสัญญาณมากตามไปด้วย ซึ่งเป็นเหตุทำให้เกิดความสัมพันธ์ (correlation) ของสัญญาณรบกวนขึ้นที่อินพุทของตัวตรวจวัดแบบอัลกอริทึมวิเทอร์บิ อย่างมาก และส่งผลให้สมรรถนะของตัวตรวจวัดด้อยลง เพื่อที่จะแก้ปัญหานี้ P. R. Chevillat และ คณะ [10] ได้เสนอให้นำเอาความสัมพันธ์ของสัญญาณรบกวนที่เกิดขึ้นมาใช้เพื่อช่วยปรับปรุงสมรรถนะของตัวตรวจวัดให้ดีขึ้น โดยการขจัดความสัมพันธ์ของสัญญาณรบกวนที่อินพุทของตัวตรวจวัด จากนั้นสี่ปีต่อมา E. Eleftheriou และ W. Hirt [11] ได้เสนอการตรวจวัดที่ใช้การทำนายสัญญาณรบกวนผนวกเข้ากับการคำนวณเมทริก (metric) ในอัลกอริทึมวิเทอร์บิ เพื่อที่จะปรับปรุงสมรรถนะของตัวตรวจวัดสำหรับช่องสัญญาณแบบ PR4 และ EPR4

รหัสเทอร์โบ (turbo code) ได้เสนอครั้งแรกเมื่อปี ค.ศ. 1993 [12]–[13] โดยยอมรับกันว่าสามารถให้สมรรถนะเข้าใกล้ขีดจำกัดของแชนนอน (Shannon limit) หลักการที่สำคัญของการถอดรหัสแบบนี้ก็คือการถอดรหัสที่อาศัยการวนซ้ำเพื่อช่วยลดปัญหาความซับซ้อนของการถอดรหัส ด้วยเหตุที่รหัสเทอร์โบให้สมรรถนะที่ดีจึงมีแนวความคิดที่จะนำเอารหัสดังกล่าวมาประยุกต์ใช้ร่วมกับการตรวจวัดสัญญาณที่อาศัยการทำนายสัญญาณรบกวนเพื่อปรับปรุงสมรรนะการเข้าและถอดรหัสสำหรับระบบการบันทึกข้อมูลบนตัวกลางแม่เหล็ก ให้ดียิ่งขึ้น

5.3 ระบบที่เสนอ

5.3.1 ระบบเข้ารหัส

รูปที่ 2 แสดงระบบเข้ารหัสที่เสนอซึ่งประกอบด้วยตัวเข้ารหัสคอนโวลูชันที่มีการป้อนกลับแบบมีระบบ (Recursive Systematic Convolutional (RSC) Encoder) จำนวนสองตัวมาต่อกันแบบขนานด้วยตัวสลับลำดับ (interleaver) ในการเข้ารหัสนั้น ลำดับข้อมูลแบบไบนารี $d_1, d_2, ..., d_N$ ซึ่งเขียนโดยย่อว่า $\underline{d}_1^N$ ได้รับการเข้ารหัสโดยตัวเข้ารหัสคอนโวลูชัน RSC 1 และ RSC 2 จากนั้นบิตพาริตี (parity bit) ที่ได้จากตัวถอดรหัสทั้งสองจะถูกตัดออก (puncture) บางส่วนเพื่อให้ได้อัตราการเข้ารหัส (coding rate) สุทธิเป็น 8/9 จากนั้นคำรหัสที่ได้จะถูกสลับลำดับด้วยตัวสลับลำดับช่องสัญญาณ (channel interleaver) และเข้ารหัสด้วยพรีโคดเดอร์ (precoder) ได้เป็นลำดับ $\underline{a}_1^N$ ก่อนที่บันทึกลงบนตัวกลางแม่เหล็ก



รูปที่ 2 ระบบการเข้ารหัสเทอร์โบและช่องสัญญาณการบันทึกข้อมูลบนตัวกลางแม่เหล็ก

5.3.2 แบบจำลองของช่องสัญญาณการบันทึกข้อมูลบนตัวกลางแม่เหล็ก

ในการกระบวนการอ่านสัญญาณที่บันทึกไว้กลับคืนนั้น ให้พิจารณารูปที่ 2 ในส่วนที่เป็นช่องสัญญาณการบันทึก (recording channel) และส่วนการบวกด้วยสัญญาณรบกวน $n(t)$ ส่วนดังกล่าวคือแบบจำลองช่องสัญญาณของระบบนั่นเอง ในหัวข้อนี้จะกล่าวถึงรายละเอียดในเชิงคณิตศาสตร์ของส่วนที่ว่านี้ต่อจากที่ได้กล่าวไปแล้วส่วนหนึ่งในตอนต้นของหัวข้อที่ 6

ด้วยข้อสมมุติที่ว่า ช่องสัญญาณบันทึกข้อมูลนี้มีคุณสมบัติ linear superposition จะได้ว่า รูปคลื่นสัญญาณต่อเนื่องจากการอ่านกลับ (read back) สามารถแสดงได้ดังนี้ [8], [14]

$$r(t) = \sum_k a_k h(t - kT) + n(t) \qquad (2)$$

โดยที่

T คือ ความกว้างของบิต

$a_k \in \{1, -1\}$ คือ อินพุตข้อมูลของช่องสัญญาณซึ่งโดยทั่วไปจะผ่านการพรีโคดแล้ว

$n(t)$ คือ สัญญาณรบกวนแบบเกาส์สีขาวแบบบวก (AWGN)

$h(t)$ คือ ผลตอบสนองพัลส์ หรือเรียกอีกอย่างว่า ผลตอบสนองไดบิต (dibit response) ซึ่งมีนิยาม ดังนี้ [14]

$$h(t) = \frac{1}{2}(s(t) - s(t-T)) \quad (3)$$

จากนั้นสัญญาณ $r(t)$ จากหัวอ่านจะถูกนำไปผ่านแมทช์ฟิลเตอร์ (matched filter) $h(-t)$ แล้วได้รับการสุ่มตัวอย่าง(sampling) ด้วยความถี่เดียวกับการเขียนข้อมูลเป็น $1/T$ ผลลัพธ์ที่ได้คือ

$$u(D) = X_h(D)a(D) + v(D) \quad (4)$$

โดยที่

D คือ การหน่วงหนึ่งหน่วย (unit delay)

$X_h(D)$ คือ ลำดับอัตสหสัมพันธ์ของผลตอบสนองไดบิตซึ่งสัมประสิทธ์ลำดับที่ k คือ

$$x_{h,k} = \int_{-\infty}^{\infty} h(t)h(t+kT)dt \quad (5)$$

$v(D)$ คือ ลำดับสัญญาณรบกวนแบบเกาส์แบบมีสีที่มีลำดับอัตสหสัมพันธ์เป็น

$$R_v(D) = E\{v(D)v(D^{-1})\} = (N_0/2)\, X_h(D) \quad (6)$$

โดยที่ N_0 คือ ความหนาแน่นกำลังด้านเดียวของสัญญาณรบกวน AWGN $n(t)$

ต่อจากนั้น ลำดับ $u(D)$ จะได้รับการปรับแต่งสัญญาณด้วยอีควอไลเซอร์เพื่อให้สัญญาณเอาท์พุท z_k มีผลตอบสนองต่ออิมพัลส์เป็นไปตามผลตอบสนองเป้าหมาย อีควอไลเซอร์ดังกล่าวเป็นแบบ FIR (Finite Impulse Response filter) ที่มีความยาว N_{EQ} ซึ่งเป็นเลขคี่ ซึ่งแสดงได้ดังนี้

$$C_{FIR}(D) = C_W(D)\, g_{PR}(D) \quad (7)$$

โดยที่

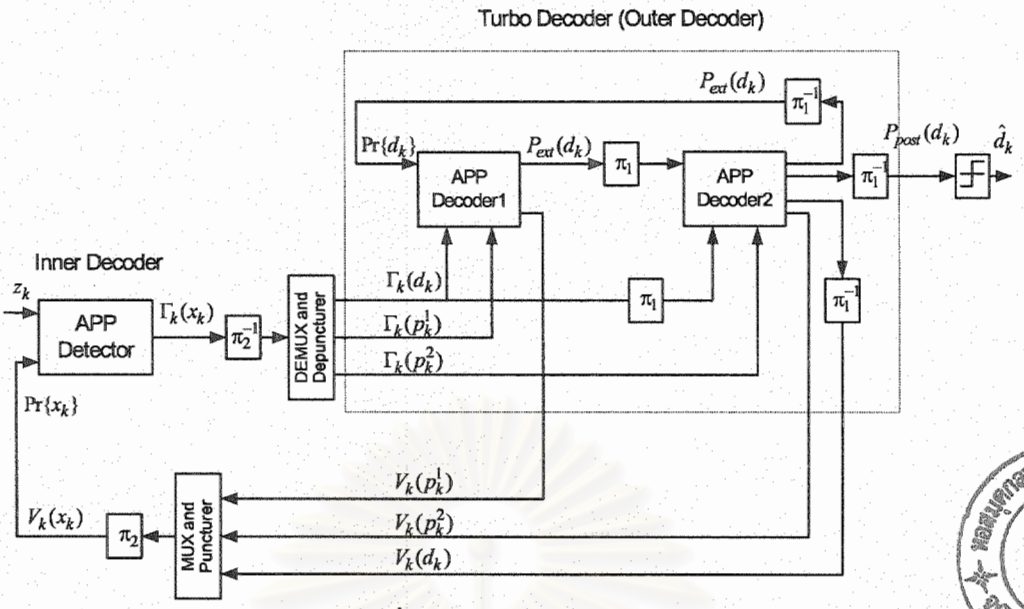
$g_{PR}(D)$ คือ พหุนามผลตอบสนองเป้าหมาย

$C_W(D)$ คือ ฟิลเตอร์ Wiener ผลตอบสนองเต็มซึ่งมีสัมประสิทธิ์ดังนี้

$$\mathbf{c}_W = (R_{XX} + R_v)^{-1}\mathbf{X}_h \quad (8)$$

โดยที่ R_{XX} คือ เมทริกซ์สหสัมพันธ์ (correlation matrix) ของ $X_h(D)X_h(D^{-1})$ และ R_v คือ เมทริกซ์สหสัมพันธ์ของ $v(D)$ กับ $\mathbf{X}_h$

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ 3 ระบบถอดรหัสที่เสนอ

5.3.3 ระบบถอดรหัสที่เสนอ

ในรูปที่ 3 แสดงระบบการถอดรหัสเทอร์โบที่เสนอขึ้นโดยอาศัยความสัมพันธ์ของสัญญาณรบกวนในเอพีพีดีเทกเทอร์ (*a posteriori* probability (APP) detector) ซึ่งมีหลักการทำงานดังนี้ เอพีพีดีเทกเทอร์ รับสัญญาณที่ปรับแต่งแล้ว z_k จากอีควอไลเซอร์ เพื่อใช้คำนวณค่าความน่าจะเป็นของแต่ละบิตในคำรหัส $\Gamma_k(x_k)$ จากนั้นค่าความน่าจะเป็นนี้จะถูกสลับลำดับและแจกจ่ายให้กับตัวถอดรหัสย่อยทั้งสอง ตัวถอดรหัสย่อยทั้งสองก็จะคำนวณค่าความน่าจะเป็นของบิตข้อมูล และบิตพาริตี โดยค่าความน่าจะเป็นเหล่านี้จะถูกตัดออกบางส่วน และสลับลำดับเพื่อป้อนกลับมายังที่ เอพีพีดีเทกเทอร์ ในกระบวนการถอดรหัสแบบวนซ้ำนี้ ค่าความน่าจะเป็นของบิตจะถูกแลกเปลี่ยนกันระหว่างตัวถอดรหัสย่อยทั้งสองและเอพีพีดีเทกเทอร์ และเมื่อระบบลู่เข้า ค่าความน่าจะเป็นของบิตข้อมูลจากตัวถอดรหัสย่อยตัวใดตัวหนึ่งจะถูกนำมาใช้ตัดสินบิตข้อมูล

ในงานวิจัยนี้ได้เสนอ การนำความสัมพันธ์ของสัญญาณรบกวนมาช่วยในการตรวจวัดของเอพีพีดีเทกเทอร์ ทั้งหมด 3 วิธีการดังนี้

1) วิธีที่ 1 (Approach 1) มีหลักการคือ นำความสัมพันธ์ของสัญญาณรบกวนมาใช้ในทำนายสัญญาณรบกวนไปข้างหน้า (forward prediction) ผนวกกับการคำนวณเมทริก (metric) ที่ใช้ในเอพีพีดีเทกเทอร์ การคำนวณค่าต่าง ๆ ที่เกี่ยวข้องมีดังนี้

- การคำนวณค่าความน่าจะเป็นไปข้างหน้าและค่าความน่าจะเป็นย้อนกลับทำได้ตามสูตรดังนี้

$$\alpha_k(S_{k-L},\underline{x}_{k-L+1}^{k}) = \alpha(S_{k-L},x_{k-L+1},...,x_k)$$
$$= \sum_{S_{k-L-1}}\sum_{x_{k-L}} M_k(S_{k-L-1},\underline{x}_{k-L}^{k})\alpha_{k-1}(S_{k-L-1},\underline{x}_{k-L}^{k-1})$$
$$\Pr\{x_k\}\Pr\{S_{k-L}\mid S_{k-L-1},x_{k-L}\} \quad (9)$$

และ

$$\beta_k(S_{k-L},\underline{x}_{k-L+1}^{k}) = \beta(S_{k-L},x_{k-L+1},...,x_k)$$
$$= \sum_{S_{k-L+1}}\sum_{x_{k+1}} M_{k+1}(S_{k-L},\underline{x}_{k-L+1}^{k+1})\beta_{k+1}(S_{k-L+1},\underline{x}_{k-L+2}^{k+1})$$
$$\Pr\{x_{k+1}\}\Pr\{S_{k-L+1}\mid S_{k-L},x_{k-L+1}\} \quad (10)$$

โดยที่

L คือ อันดับการทำนายไปข้างหน้า

$\Pr\{x_k\}$ คือ ความน่าจะเป็นเบื้องแรก (*a priori* probability) ของบิต x_k

$\Pr\{S_k\mid S_{k-1},x_k\}$ ความน่าจะเป็นของการเลื่อนสถานะของตัวเข้ารหัสอินนอร์ (inner encoder) ที่เกี่ยวข้องกับบิต x_k,

$$\Pr\{S_k\mid S_{k-1},x_k\} = \begin{cases} 1, & \text{legitimate state transition} \\ 0, & \text{otherwise.} \end{cases} \quad (11)$$

- การคำนวณค่าเมทริกสำหรับตัวตรวจวัดทำได้ตามสูตรดังนี้

$$M_k(S_{k-L-1},\underline{x}_{k-L}^{k}) = M_k(\underline{y}_{k-L}^{k})$$
$$= \frac{1}{\sqrt{2\pi\sigma_{predict}^2}}\exp\left(-\frac{\left((z_k-y_k)-\sum_{i=1}^{L}P_i(z_{k-i}-y_{k-i})\right)^2}{2\sigma_{predict}^2}\right) \quad (12)$$

- การคำนวณค่าข่าวสารเอกซ์ทรินซิก (extrinsic information) $\Gamma_k(x_k)$ ทำได้ตามสูตรดังนี้

$$\Gamma_k(x_k) = \frac{\sum_{\underline{S}_{k-L-1}^{k-L}}\sum_{\underline{x}_{k-L}^{k-1}}\alpha_{k-1}(S_{k-L-1},\underline{x}_{k-L}^{k-1})M_k(S_{k-L-1},\underline{x}_{k-L}^{k})\beta_k(S_{k-L},\underline{x}_{k-L+1}^{k})\Pr\{S_{k-L}\mid S_{k-L-1},x_{k-L}\}}{\sum_{x_k}\sum_{\underline{S}_{k-L-1}^{k-L}}\sum_{\underline{x}_{k-L}^{k-1}}\alpha_{k-1}(S_{k-L-1},\underline{x}_{k-L}^{k-1})M_k(S_{k-L-1},\underline{x}_{k-L}^{k})\beta_k(S_{k-L},\underline{x}_{k-L+1}^{k})\Pr\{S_{k-L}\mid S_{k-L-1},x_{k-L}\}} \quad (13)$$

2) วิธีที่ 2 (Approach 2) มีหลักการคือ นำความสัมพันธ์ของสัญญาณรบกวนมาใช้ สำหรับการทำนายสัญญาณรบกวนไปข้างหน้า และย้อนกลับ (forward and backward prediction) ผนวกกับการคำนวณเมทริกที่ใช้ในเอพีพีดีเทกเทอร์ การคำนวณค่าต่าง ๆ ที่เกี่ยวข้องมีดังนี้

- การคำนวณค่าเมทริกสำหรับตัวตรวจวัดทำได้ตามสูตรดังนี้

$$M_k(S_{k-L-1}, \underline{x}_{k-L}^{k+K}) = M_k(\underline{y}_{k-L}^{k+K})$$

$$= \frac{1}{\sqrt{2\pi\sigma^2_{estimate}}} \exp\left(- \frac{\left((z_k - y_k) - \sum_{i=-K, i\neq 0}^{L} P_i (z_{k-i} - y_{k-i}) \right)^2}{2\sigma^2_{estimate}} \right) \quad (14)$$

- การคำนวณค่าความน่าจะเป็นไปข้างหน้าและค่าความน่าจะเป็นย้อนกลับทำได้ตามสูตรดังนี้

กำหนดให้ $$\Delta_k = (S_{k-L}, \underline{x}_{k-L+1}^{k+K}) \quad (15)$$

$$\alpha_k(S_{k-L}, \underline{x}_{k-L+1}^{k+K}) = \alpha_k(\Delta_k)$$
$$= \sum_{S_{k-L-1}} \sum_{x_{k-L}} M_k(S_{k-L-1}, \underline{x}_{k-L}^{k+K}) \alpha_{k-1}(S_{k-L-1}, \underline{x}_{k-L}^{k+K-1})$$
$$\Pr\{x_k\} \Pr\{S_{k-L} \mid S_{k-L-1}, x_{k-L}\} \quad (16)$$

และ

$$\beta_k(S_{k-L}, \underline{x}_{k-L+1}^{k+K}) = \beta_k(\Delta_k)$$
$$= \sum_{S_{k-L+1}} \sum_{x_{k+K+1}} M_{k+1}(S_{k-L}, \underline{x}_{k-L+1}^{k+K+1}) \beta_{k+1}(S_{k-L+1}, \underline{x}_{k-L+2}^{k+K+1})$$
$$\Pr\{x_{k+1}\} \Pr\{S_{k-L+1} \mid S_{k-L}, x_{k-L+1}\} \quad (17)$$

- การคำนวณค่าข่าวสารเอกซ์ทรินซิก (extrinsic information) $\Gamma_k(x_k)$ ทำได้ตามสูตรดังนี้

$$\Gamma_k(x_k) = \frac{\sum_{S_{k-L-1}^{k-L}} \sum_{\underline{x}_{k-L}^{k-1}} \sum_{\underline{x}_{k+1}^{k+K}} \alpha_{k-1}(\Delta_{k-1}) M_k(S_{k-L-1}, \underline{x}_{k-L}^{k+K}) \beta_k(\Delta_k) \Pr\{S_{k-L} \mid S_{k-L-1}, x_{k-L}\}}{\sum_{x_k} \sum_{S_{k-L-1}^{k-L}} \sum_{\underline{x}_{k-L}^{k-1}} \sum_{\underline{x}_{k+1}^{k+K}} \alpha_{k-1}(\Delta_{k-1}) M_k(S_{k-L-1}, \underline{x}_{k-L}^{k+K}) \beta_k(\Delta_k) \Pr\{S_{k-L} \mid S_{k-L-1}, x_{k-L}\}} \quad (18)$$

3) วิธีที่ 3 (Approach 3) มีหลักการคือ นำอัลกอริทึมวิเทอร์บิ มาใช้เพื่อช่วยลดความซับซ้อนของวิธีที่ 1 การคำนวณค่าต่าง ๆ ที่เกี่ยวข้องมีดังนี้

- การคำนวณค่าเมตริกเส้นทาง (path metric) $A_k(S_k)$ ของ survivor ที่เกี่ยวข้องกับสถานะเทรลลิส S_k ได้ดังนี้

$$A_k(S_k) = \max_{S_{k-1}} \left(A_{k-1}(S_{k-1}) + \log \Pr\{x_k\} + \log M_k(y_k, \underline{\hat{y}}_{k-L}^{k-1}(S_{k-1})) \right) \quad (19)$$

โดยที่

y_k คือ เอาท์พุทของกิ่ง (S_{k-1}, S_k)

$\underline{\hat{y}}_{k-L}^{k-1}(S_{k-1})$ คือ ลำดับเอาท์พุทของ survivor ที่เกี่ยวข้องกับสถานะเทรลลิส S_{k-1}

- การคำนวณค่าความน่าจะเป็นไปข้างหน้าและค่าความน่าจะเป็นย้อนกลับทำได้ตามสูตรดังนี้

$$\alpha_k(S_k) = \sum_{S_{k-1}} \sum_{x_k} M_k(y_k, \underline{\hat{y}}_{k-L}^{k-1}) \alpha_{k-1}(S_{k-1}) \Pr\{x_k\} \Pr\{S_k \mid S_{k-1}, x_k\} \quad (20)$$

และ

$$\beta_k(S_k) = \sum_{S_{k+1}} \sum_{x_{k+1}} M_{k+1}(y_{k+1}, \underline{\hat{y}}_{k-L+1}^{k}) \beta_{k+1}(S_{k+1}) \Pr\{x_{k+1}\} \Pr\{S_{k+1} \mid S_k, x_{k+1}\} \quad (21)$$

- การคำนวณค่าข่าวสารเอกซ์ทรินซิก (extrinsic information) $\Gamma_k(x_k)$ ทำได้ตามสูตรดังนี้

$$\Gamma_k(x_k) = \frac{\sum_{\underline{S}_{k-1}^{k}} \alpha_{k-1}(S_{k-1}) M_k(y_k, \underline{\hat{y}}_{k-L}^{k-1}) \beta_k(S_k) \Pr\{S_k \mid S_{k-1}, x_k\}}{\sum_{x_k} \sum_{\underline{S}_{k-1}^{k}} \alpha_{k-1}(S_{k-1}) M_k(y_k, \underline{\hat{y}}_{k-L}^{k-1}) \beta_k(S_k) \Pr\{S_k \mid S_{k-1}, x_k\}} \quad (22)$$

5.4 สมรรถนะของระบบถอดรหัสที่เสนอ

พารามิเตอร์ที่ใช้ในการจำลองทางคอมพิวเตอร์เป็นไปดังที่แสดงในตารางที่ 1 ข้างล่างนี้ โดยในที่นี้จะวัดสมรรถนะด้วยค่าอัตราความผิดพลาดบิต (Bit Error Rate)

ตารางที่ 1 พารามิเตอร์ที่ใช้ในการจำลองทางคอมพิวเตอร์

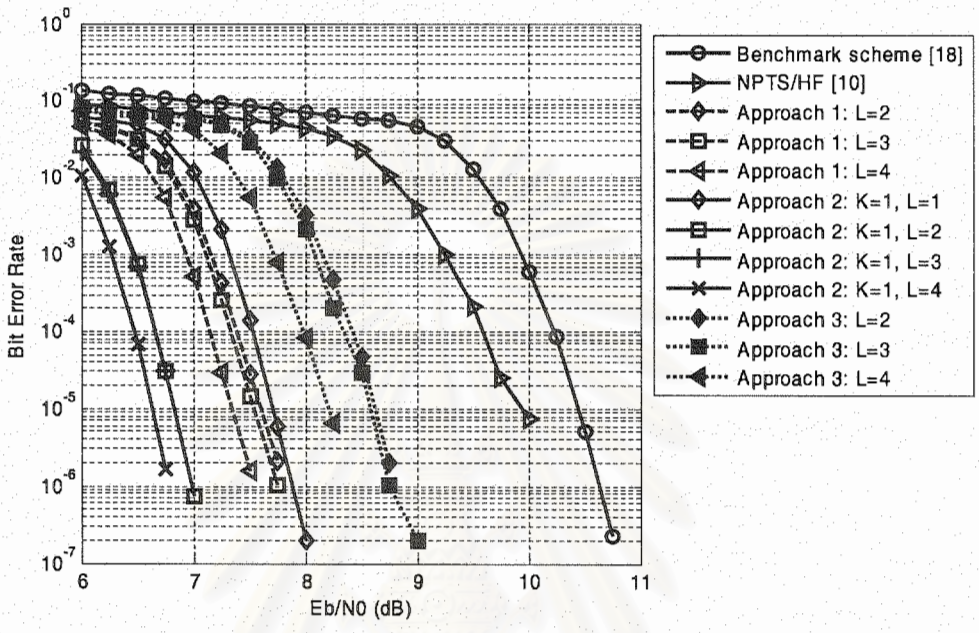
จำนวนการทดลอง (trial)		200,000 รอบ
ความหนาแน่นของการบันทึก (recording density)		2.86
พหุนามเป้าหมาย		PR4
พรีโคดเดอร์		$1/(1 \oplus D)$
จำนวนแทปของอีควอไลเซอร์ N_{EQ}		41
ขนาดของบล็อกข้อมูล		4,096
อัตรารหัส (coding rate)		8/9
ตัวเข้ารหัสย่อยเทอร์โบ (RSC1 และ RSC2)	พหุนามป้อนไปข้างหน้า	$1 \oplus D \oplus D^3 \oplus D^4$
	พหุนามป้อนกลับ	$1 \oplus D^3 \oplus D^4$

รูปที่ 4 และ 5 แสดงกราฟสมรรถนะของระบบที่เสนอแบบต่าง ๆ เปรียบเทียบกับวิธีที่ใช้เป็น benchmark ที่ได้เสนอในบทความ [18] และระบบเทอร์โบแบบมีการทำนายสัญญาณรบกวนที่มีการป้อนกลับแบบฮาร์ด (NPTS/HF) ที่ได้เสนอในบทความ [10] พิจารณารูปที่ 4 จะพบว่า วิธีที่ 2 เป็นวิธีที่ดีที่สุดโดยได้อัตราขยายการเข้ารหัส (coding gain) สูงกว่าวิธีที่ 1 อยู่ 0.7 dB และสูงกว่าวิธีที่ 3 อยู่ 1.6 dB

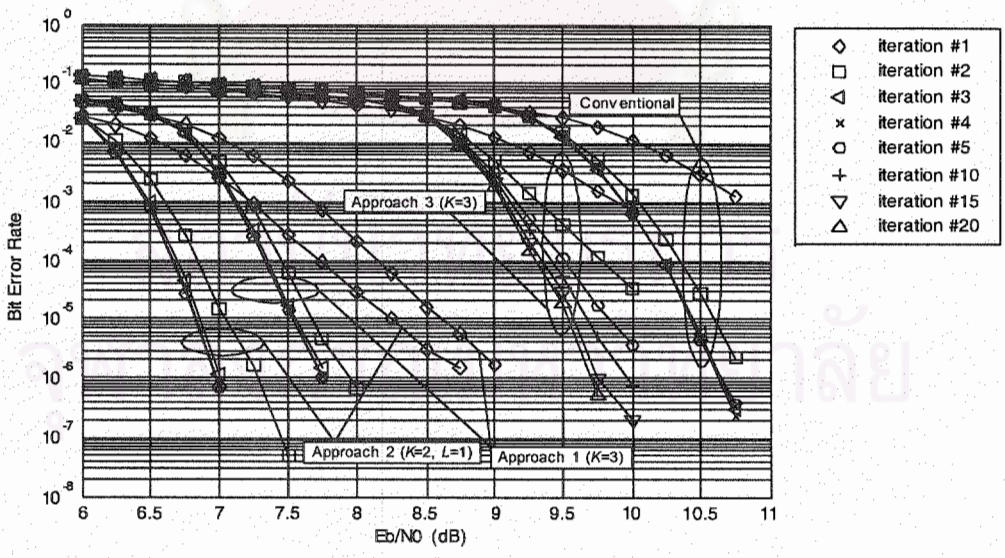
สำหรับการถอดรหัสแบบวนซ้ำ สิ่งที่ได้รับความสนใจศึกษาอย่างหนึ่ง คือ พฤติกรรมการลู่เข้า (Convergence behavior) ผลการจำลองเพื่อศึกษาแสดงในรูปที่ 5 จะพบว่า สามารถแบ่งพฤติกรรมการลู่เข้าออกได้เป็นสองกลุ่มคือ กลุ่มแรก ได้แก่ วิธีที่ใช้เป็น benchmark วิธีที่ 1 และวิธีที่ 2 สมรรถนะจะสูงขึ้นอย่างรวดเร็วเมื่อเพิ่มจำนวนรอบของการถอดรหัสวนซ้ำในช่วงแรก และหลังจากรอบที่ 4 สมรรถนะจึงแทบไม่เพิ่มขึ้นอีก อีกกลุ่มหนึ่ง คือ วิธีที่ 3 พฤติกรรมการลู่เข้าจะแตกต่างออกไปเล็กน้อย สมรรถนะจะเพิ่มขึ้น

เรื่อย ๆ ตามรอบการวนซ้ำที่เพิ่มขึ้น จนกระทั่งประมาณหลังจากรอบที่ 15 สมรรถนะจึงแทบจะไม่เพิ่มขึ้นอีก

จากผลการจำลองทางคอมพิวเตอร์ จะพบว่า การนำความรู้ของความสัมพันธ์ของสัญญาณรบกวนมาใช้สามารถเอื้อให้สมรรถนะของระบบถอดรหัสดีขึ้นอย่างเห็นได้ชัด เมื่อเปรียบเทียบกับวิธีดั้งเดิมที่มิได้อาศัยความสัมพันธ์ของสัญญาณรบกวนมาช่วยในการตรวจวัดสัญญาณ



รูปที่ 4 การเปรียบเทียบสมรรถนะของระบบถอดรหัสที่เสนอและวิธีที่ใช้เปรียบเทียบ



รูปที่ 5 การเปรียบเทียบสมรรถนะของระบบถอดรหัสที่เสนอและวิธีที่ใช้เปรียบเทียบ

6. รายละเอียดงานวิจัยส่วนที่ 2

6.1 ทฤษฎีที่เกี่ยวข้อง

6.1.1 อัลกอริทึม iSLIP

อัลกอริทึม iSLIP ใช้หลักการวนรอบอินพุตแต่ละอินพุต (round robin) เพื่อทำการตัดสินใจที่จะจัดลำดับอินพุตและเอาต์พุตในแต่ละรอบของการวนซ้ำ คุณลักษณะที่สำคัญของอัลกอริทึมก็คือ สามารถนำไปสร้างเป็นฮาร์แวร์ได้ง่ายและทำงานได้ดีที่การสวิตช์ความเร็วสูง แสดงให้เห็นว่า สมรรถนะของอัลกอริทึมสำหรับทราฟฟิกประเภทยูนิฟอร์มมีค่าสูงมาก ถึงแม้ว่า จะไม่มีการวนซ้ำก็ให้ค่าทรูพุตสูงถึง 100 เปอร์เซ็นต์ และเนื่องจากการที่มีการวนซ้ำจึงทำให้ตัวตัดสินใจของอัลกอริทึมแต่ละตัว มีแนวโน้มไม่ซิงโครไนซ์กับตัวตัดสินใจอื่น ๆ

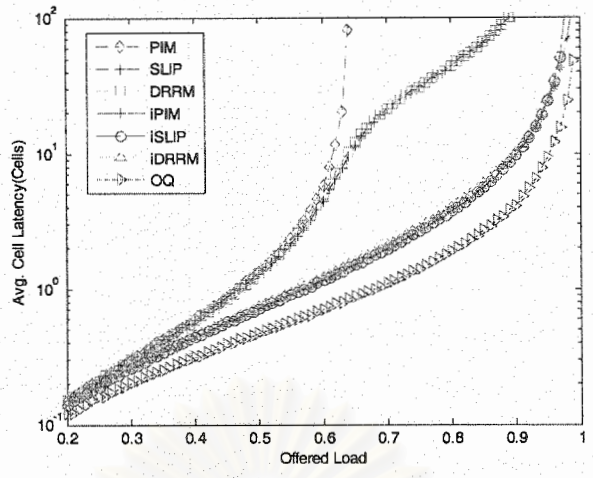
อัลกอริทึม iSLIP แบบไม่วนซ้ำ

อัลกอริทึม iSLIP ได้รับการพัฒนามาจากอัลกอริทึม RRM โดยอัลกอริทึม iSLIP จะไม่ทำการเลื่อนค่าตัวบ่งชี้ g_j ถ้าคำยินยอมที่ส่งไปให้อินพุตไม่ได้รับการตอบรับจากอินพุตนั้น จะเห็นว่า อัลกอริทึม iSLIP ก็คล้ายกันกับอัลกอริทึม PIM เว้นแต่เงื่อนไขในการเลื่อนค่าของตัวบ่งชี้ g_j ในขั้นที่ 2 ที่เปลี่ยนแปลงไป ดังนี้

- ขั้นตอนที่ 1 Request แต่ละอินพุตที่มีเซลล์รออยู่ในคิวและต้องการจะไปยังเอาต์พุตนั้น ส่งคำร้องขอไปยังเอาต์พุตนั้น

- ขั้นตอนที่ 2 Grant เมื่อแต่ละเอาต์พุตได้รับคำร้องขอแล้ว จะทำการเลือกคำร้องขอเพียงคำร้องขอเดียว ที่ปรากฏว่ามีค่าลำดับความสำคัญสูงสุด โดยการวนรอบแต่ละอินพุต โดยมีตัวบ่งชี้ g_j ทำการเลือกแล้วส่งคำยินยอมไปยังอินพุตนั้น และจะชี้อินพุตถัดไปโดยกำหนดให้อินพุตนั้นมีค่าลำดับดับความสำคัญสูงสุดในรอบต่อไป ก็ต่อเมื่อคำยินยอมนั้นได้รับการตอบรับ และทำการเชื่อมต่อระหว่างอินพุตและเอาต์พุต ในขั้นตอนที่ 3 แล้ว

- ขั้นตอนที่ 3 Accept เมื่อแต่ละอินพุตได้รับคำยินยอมแล้วจะเลือกตอบรับคำยินยอมของเอาต์พุตโดยทำการเลือกเอาต์พุตที่มีค่าลำดับความสำคัญสูงสุด โดยมีตัวบ่งชี้ a_i วนรอบแต่ละเอาต์พุต และชี้เอาต์พุตถัดไป (modulo N) โดยกำหนดให้เอาต์พุตนั้นมีค่าลำดับความสำคัญสูงสุด เมื่อได้เลือกเอาต์พุตนั้นและทำการเชื่อมต่อกับอินพุตแล้ว



รูปที่ 6 สวิตช์แบบรอคิวอินพุตและสมรรถนะด้านดีเลย์ของอัลกอริทึมจัดลำดับเซลล์แบบ iSLIP Algorithm

คุณสมบัติของอัลกอริทึม iSLIP ที่ไม่มีการวนซ้ำ มีดังต่อไปนี้

- ข้อที่ 1 ให้ค่าลำดับความสำคัญต่ำสุด สำหรับการเชื่อมต่อที่เป็นปัจจุบันที่สุด ทั้งนี้ เนื่องมาจากเมื่อตัวตัดสินใจเลื่อนค่าของตัวบ่งชี้อินพุตที่ตอบรับคำยินยอมของเอาต์พุตใด และมีการเชื่อมต่อระหว่างอินพุตและเอาต์พุตนั้น ตัวบ่งชี้ g_j จะให้ค่าลำดับความสำคัญของอินพุตนั้นต่ำสุด คือถ้าอินพุต i เชื่อมต่อกับเอาต์พุต j เรียบร้อยแล้ว ทั้ง a_i และ g_j จะเลื่อนค่าของตัวบ่งชี้ไปหนึ่งค่า (modulo N) ทำให้การเชื่อมต่อระหว่างอินพุต i และเอาต์พุต j มีค่าลำดับความสำคัญต่ำสุด ในการเชื่อมต่อในไทม์สล็อตต่อไป

- ข้อที่ 2 ไม่มีการเชื่อมต่อใดเกิดสภาวะสทาร์เวชัน เนื่องจากแต่ละอินพุตจะส่งคำร้องขอต่อเนื่องกันไป จนกว่าจะได้รับการเชื่อมต่อระหว่างอินพุตกับเอาต์พุตนั้น

- ข้อที่ 3 ในกรณีที่ทราฟฟิกหนาแน่น ทุก ๆ คิวที่จะไปยังเอาต์พุตเดียวกัน จะให้ค่าทรูพุตที่เหมือนกัน ซึ่งเป็นผลที่ตามมาจากคุณสมบัติข้อที่ 2 เนื่องจากตัวบ่งชี้ของเอาต์พุต เลื่อนค่าอินพุตในตัวบ่งชี้ ต่อเนื่องกันไปเป็นลำดับ จึงทำให้แต่ละอินพุตให้ค่าทรูพุตที่เหมือนกัน

6.1.2 อัลกอริทึม Prioritized iSLIP

การใช้งานหลายอย่างใช้ชั้นของทราฟฟิกที่มีระดับไพรออริตี้ต่างกัน อัลกอริทึม iSLIP สามารถถูกขยายไปยังรวมทั้งความต้องการที่ระดับไพรออริตี้หลายไพรออริตี้ ซึ่งเราจะเรียกว่าอัลกอริทึม Prioritized iSLIP

ใน Prioritized iSLIP แต่ละอินพุตยังคงรักษาคิว FIFO ที่แยกสำหรับแต่ละระดับไพรออริตี้ และสำหรับแต่ละเอาท์พุต นั่นหมายความว่า สำหรับสวิตช์ขนาด N×N ที่มีระดับ P ไพรออริตี้ แต่ละอินพุตจะมี

คิว P×N FIFO's เราจะให้ชื่อคิวระหว่างอินพุต i และเอาท์พุต j ที่ระดับไพรออริตี้ l ว่า $Q_l(i,j)$ ที่ซึ่ง $1 \leq i, j \leq N, 1 \leq l \leq P$ ดังที่กล่าวแล้วข้างต้น มีเพียงแค่หนึ่งเซลล์สามารถมาถึงในเวลาเซลล์ ดังนั้นสิ่งนี้ไม่ต้องการกระบวนการเพิ่มความเร็วโดยอินพุต

อัลกอริทึม Prioritized iSLIP ให้ไพรออริตี้เคร่งครัดแก่การร้องขอไพรออริตี้สูงสุดในแต่ละเวลาเซลล์ สิ่งนี้หมายความว่า $Q_l(i,j)$ จะถูกบริการถ้าคิว $Q_m(i,j)$, $1 \leq m \leq P$ ทั้งหมดยังว่างอยู่

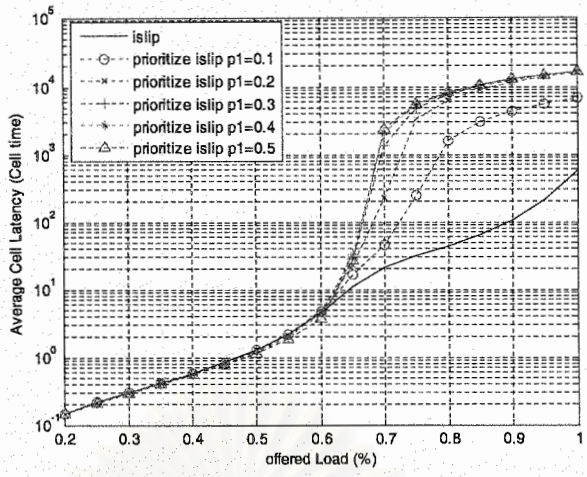
อัลกอริทึม iSLIP ถูกปรับเปลี่ยนดังนี้

Step 1: Request อินพุต i เลือกคิวที่มีลำดับความสำคัญสูงสุดที่มีเซลล์รออยู่ในคิวสำหรับเอาท์พุต j อินพุตส่งระดับของลำดับความสำคัญ l_{ij} ของคิวเหล่านี้ไปยังเอาท์พุต j

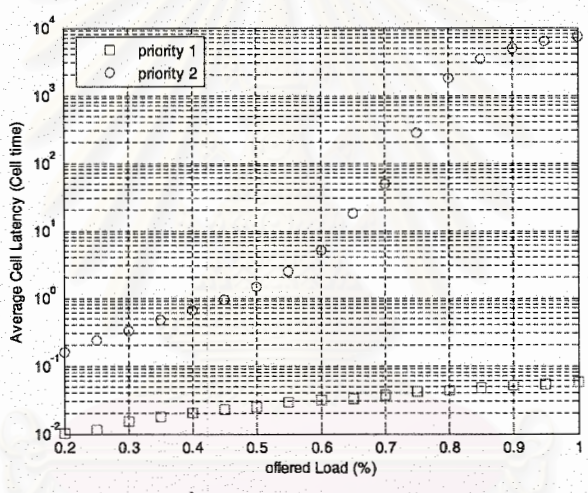
Step 2: Grant ถ้าเอาท์พุต j ได้รับหลายคำร้องขอ มันตัดสินคำร้องขอระดับสูงสุด เช่น มันหา $L(j) = \max_i(l_{ij})$ เอาท์พุตตอนนั้นเลือกอินพุตหนึ่งอินพุตท่ามกลางเฉพาะอินพุตที่มีคำร้องขอที่ระดับ $L(j)$ ตัวตัดสินใจเอาท์พุตรักษาตัวชี้แยกกัน gil สำหรับแต่ละระดับลำดับความสำคัญ เมื่อเลือกท่ามกลางอินพุตที่ระดับ $L(j)$ ตัวตัดสินใจใช้ตัวชี้ $g_{jL(j)}$ และเลือกใช้วิธี round-robin เหมือนกันตามก่อน เอาท์พุตแจ้งแต่ละอินพุตเมื่อคำร้องขอของมันถูกตอบรับ ตัวบ่งชี้ g_j ทำการเลือกแล้วส่งคำยินยอมไปยังอินพุตนั้น และจะชี้อินพุตถัดไปโดยกำหนดให้อินพุตนั้นมีค่าลำดับความสำคัญสูงสุดในรอบต่อไป ก็ต่อเมื่อคำยินยอมนั้นได้รับการตอบรับ และทำการเชื่อมต่อระหว่างอินพุตและเอาต์พุต ในขั้นตอนที่ 3 แล้ว

Step 3: Accept ถ้าอินพุต i ได้รับหลายคำยินยอม มันตัดสินคำตอบรับระดับสูงสุด เช่น มันหา $L'(j) = max_j(l_{ij})$ อินพุตตอนนั้นเลือกเอาท์พุตหนึ่งเอาท์พุตท่ามกลางเฉพาะเอาท์พุตที่มีคำร้องขอที่ระดับ $i_{ij} = L(i)$ ตัวตัดสินใจอินพุตรักษาตัวชี้แยกกัน ail สำหรับแต่ละระดับลำดับความสำคัญ เมื่อเลือกอินพุตที่ระดับ $L'(i)$ ตัวตัดสินใจใช้ตัวชี้ $a_{iL'(i)}$ และเลือกใช้วิธี round-robin เหมือนกันตามก่อน อินพุตแจ้งแต่ละเอาท์พุตเมื่อคำตอบรับของมันถูกยอมรับ โดยมีตัวบ่งชี้ a_i วนรอบแต่ละเอาต์พุต และชี้เอาต์พุตถัดไป (modulo N) โดยกำหนดให้เอาต์พุตนั้นมีค่าลำดับความสำคัญสูงสุด เมื่อได้เลือกเอาท์พุตนั้นและทำการเชื่อมต่อกับอินพุตแล้ว

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ 7 แสดงผลกระทบของลำดับความสำคัญของเซลล์เมื่อสัดส่วนปริมาณเซลล์คลาส 1 เพิ่มขึ้น ที่มีต่อ Prioritized iSLIP algorithm



รูปที่ 8 แสดงความสัมพันธ์ระหว่างเซลล์ที่มีลำดับความสำคัญต่างกัน ใน Prioritized iSLIP algorithm เมื่อสัดส่วนปริมาณเซลล์คลาส 1 เป็น 10%

6.1.3 อัลกอริทึมการจัดกำหนดการสำหรับการจัดสรรบริการในเซลล์สวิตช์ที่พิจารณาลำดับความสำคัญของเซลล์ที่นำเสนอ

หลักการทำงานของอัลกอริทึมคือการควบคุมสัดส่วนการให้บริการของเซลล์แต่ละคลาสที่เข้ามาในแต่ละขั้นตอนการตัดสินใจของอัลกอริทึม iSLIP โดยจะแบ่งเป็น 3 ขั้นตอนได้แก่ request, grant และ accept และใช้ตารางการจัดกำหนดการ round-robin สองชุด ทั้งในขั้นตอน grant และ accept เพื่อให้การจัดกำหนดการยังคงลักษณะของอัลกอริทึม iSLIP อยู่

1) ควบคุมที่ request

ในลำดับแรกเราต้องการที่จะควบคุมสัดส่วนการให้บริการตั้งแต่ต้นหรืออีกนัยหนึ่งคือ ต้องการจะเลือกคลาสของเซลล์ที่จะได้รับบริการตั้งแต่ต้นเพื่อพิจารณาว่าการคัดเลือกตั้งแต่ต้นดังกล่าวจะเป็นการตัดสินใจที่เร็วเกินไปหรือไม่ และสามารถควบคุมได้อย่างมีประสิทธิภาพหรือไม่

ขั้นตอนการทำงานของอัลกอริทึม

ขั้นที่1 : request เมื่อมีเซลล์เข้ามาในบัฟเฟอร์ของอินพุต อินพุตดังกล่าวจะจัดเซลล์นั้นไว้ยัง VOQ ของลำดับความสำคัญของเซลล์นั้นๆ จากนั้นอินพุตแต่ละตัวจะพิจารณาว่ามีเซลล์ใน VOQ ทั้งสองลำดับความสำคัญที่ต้องการจะส่งหรือไม่ ถ้ามี ก็จะพิจารณาว่ามีกี่ลำดับความสำคัญที่จะส่งไปยังเอาท์พุตตัวเดียวกัน หากมีลำดับเดียวก็ทำการส่ง request และลำดับความสำคัญของเซลล์ที่ต้องการจะส่งไปยังเอาท์พุตที่ต้องการ แต่หากมีเซลล์มากกว่าหนึ่งลำดับความสำคัญต้องการจะส่งไปยังเอาท์พุตเดียวกันก็จะเลือกลำดับความสำคัญโดยการสุ่มแบบถ่วงน้ำหนักของแต่ละลำดับความสำคัญ และทำการส่ง request และลำดับความสำคัญของเซลล์ที่เลือกไปยังเอาท์พุตที่ต้องการ

ขั้นที่ 2 : grant เมื่อเอาท์พุตได้รับ request และลำดับความสำคัญจากอินพุตแล้ว เอาท์พุตตัวดังกล่าวจะพิจารณาว่ามีการแย่งชิงกันของเซลล์ที่มีลำดับความสำคัญที่แตกต่างกันหรือไม่ หากไม่มี เอาท์พุตจะเลือกอินพุตตาม ตารางการจัดกำหนดการ round-robin ของลำดับความสำคัญนั้นๆ จากนั้นเอาท์พุตจะส่ง grant และลำดับความสำคัญของเซลล์ที่เลือก ไปยังอินพุตตัวที่เลือก และรอรับ accept จากขั้นที่ 3 จากนั้นตัวชี้ตำแหน่งในตารางการจัดกำหนดการ round robin ของเอาท์พุตที่ลำดับความสำคัญนั้นๆ จะเปลี่ยนแปลงโดยเพิ่มขึ้นทีละหนึ่งลำดับวนไปเรื่อยๆแบบ โมดูโล เมื่อได้รับ accept ตอบรับจากอินพุตดังกล่าว แต่หากมีการแย่งชิงกันของเซลล์ที่มีลำดับความสำคัญที่แตกต่างกันแล้ว เอาท์พุตจะทำการสุ่มเลือกลำดับความสำคัญของเซลล์ที่จะให้บริการโดยให้น้ำหนักในการเลือกที่เท่ากัน จากนั้นเอาท์พุตจะเลือกอินพุตตาม ตารางการจัดกำหนดการ round-robin ของลำดับความสำคัญที่เลือก และส่ง grant พร้อมด้วยลำดับความสำคัญของเซลล์ดังกล่าวไปยังอินพุตตัวที่เลือก และรอรับ accept จากขั้นที่ 3 จากนั้นตัวชี้ตำแหน่งในตารางการจัดกำหนดการ round robin ของเอาท์พุตที่ลำดับความสำคัญนั้นๆ จะเปลี่ยนแปลงโดยเพิ่มขึ้นทีละหนึ่งลำดับวนไปเรื่อยๆ แบบ โมดูโล เมื่อได้รับ accept ตอบรับจากอินพุตดังกล่าว

ขั้นที่ 3 : accept เมื่อเอาท์พุตได้รับ request และลำดับความสำคัญจากอินพุตแล้ว เอาท์พุตตัวดังกล่าวจะพิจารณาว่ามีการแย่งชิงกันของเซลล์ที่มีลำดับความสำคัญที่แตกต่างกันหรือไม่ หากไม่ อินพุตจะเลือกเอาท์พุตตาม ตารางการจัดกำหนดการ round-robin ของลำดับความสำคัญนั้นๆ และส่ง accept ไปยังเอาท์พุตตัวที่เลือก และทำการส่งเซล์ที่ได้รับเลือกจากอินพุตนั้นไปยังเอาท์พุตที่เลือก จากนั้นตัวชี้ตำแหน่งในตารางการจัดกำหนดการ round robin ของอินพุตที่ลำดับความสำคัญนั้นๆ จะเปลี่ยนแปลงโดยเพิ่มขึ้นทีละหนึ่งลำดับวนไปเรื่อยๆ แบบโมดูโล แต่หากมี

การแย่งชิงกันของเซลล์ที่มีลำดับความสำคัญที่แตกต่างกันแล้ว อินพุตจะทำการสุ่มเลือกลำดับความสำคัญโดยให้น้ำหนักในการเลือกที่เท่ากัน จากนั้นอินพุตจะเลือกเอาท์พุตตาม ตารางการจัดกำหนดการ round-robin ของลำดับความสำคัญที่เลือกได้ และส่ง accept ไปยังเอาท์พุตตัวที่เลือก จากนั้นจะทำการส่งเซลล์ที่ได้รับเลือกจากอินพุตนั้นไปยังเอาท์พุตที่เลือก จากนั้นตัวชี้ตำแหน่งในตารางการจัดกำหนดการ round robin ของอินพุตที่ลำดับความสำคัญนั้นๆ จะเปลี่ยนแปลงโดยเพิ่มขึ้นทีละหนึ่งลำดับวนไปเรื่อยๆ แบบโมดูโล

2) ควบคุมที่ grant

หลังจากทำการควบคุมสัดส่วนการให้บริการที่ส่วนแรกคือ request แล้ว จากนั้นเราอยากจะทราบว่าหากควบคุมสัดส่วนการให้บริการที่ขั้นตอน grant ซึ่งเป็นขั้นตอนที่สำคัญที่สุดในการจัดกำหนดการ เนื่องจากเป็นขั้นตอนที่มีผลกระทบต่ออัลกอริทึม iSLIP มากที่สุด (เห็นได้จากสมรรถนะที่ต่างกันของ iSLIP และ RRM) แล้วนั้น จะส่งผลกระทบอย่างไรต่อสมรรถนะ และความสามารถในการควบคุมสัดส่วนการให้บริการในสวิตช์ ในอัลกอริทึมนี้ เราจะไม่ควบคุมในขั้นตอน request เนื่องจากเราต้องการให้เซลล์ที่จะเข้าไปสู่การตัดสินใจในขั้นตอนต่อไปมีจำนวนมากที่สุดเท่าที่จะเป็นไปได้เนื่องจากหากควบคุมสัดส่วนการให้บริการให้เท่ากันที่ขั้นตอน request จะมีเซลล์ที่ไม่สามารถผ่านเข้ามาในขั้นตอน grant อยู่จำนวนหนึ่ง ซึ่งเป็นการตัดโอกาสในการจัดหาเส้นทางที่เหมาะสมของเซลล์เหล่านั้น ซึ่งจะทำให้มีโอกาสได้ทรูพุตที่สูงลดน้อยลง

ขั้นตอนการทำงานของอัลกอริทึม

ขั้นที่1 : request เมื่อมีเซลล์เข้ามาในบัฟเฟอร์ของอินพุต อินพุตดังกล่าวจะจัดเซลล์นั้นไว้ยัง VOQ ของลำดับความสำคัญของเซลล์นั้นๆ จากนั้นอินพุตแต่ละตัวจะพิจารณาว่ามีเซลล์ใน VOQ ที่ต้องการจะส่งหรือไม่ ถ้ามี ก็จะทำการส่ง request และลำดับความสำคัญของเซลล์ที่ต้องการจะส่งไปยังเอาท์พุตที่ต้องการโดยไม่คำนึงถึงลำดับความสำคัญของเซลล์

ขั้นที่ 2 : grant เมื่อเอาท์พุตได้รับ request และลำดับความสำคัญจากอินพุตแล้ว เอาท์พุตตัวดังกล่าวจะพิจารณาว่ามีการแย่งชิงกันของเซลล์ที่มีลำดับความสำคัญที่แตกต่างกันหรือไม่ หากไม่มี เอาท์พุตจะเลือกอินพุตตาม ตารางการจัดกำหนดการ round-robin ของลำดับความสำคัญนั้นๆ จากนั้นเอาท์พุตจะส่ง grant และลำดับความสำคัญของเซลล์ที่เลือก ไปยังอินพุตตัวที่เลือก และรอรับ accept จากขั้นที่ 3 จากนั้นตัวชี้ตำแหน่งในตารางการจัดกำหนดการ round robin ของเอาท์พุตที่ลำดับความสำคัญนั้นๆ จะเปลี่ยนแปลงโดยเพิ่มขึ้นทีละหนึ่งลำดับวนไปเรื่อยๆแบบ โมดูโล เมื่อได้รับ accept ตอบรับจากอินพุตดังกล่าว แต่หากมีการแย่งชิงกันของเซลล์ที่มีลำดับความสำคัญที่แตกต่างกันแล้ว เอาท์พุตจะทำการสุ่มเลือกลำดับความสำคัญของเซลล์ที่จะให้บริการโดยการสุ่มแบบถ่วงน้ำหนักของแต่ละลำดับความสำคัญ จากนั้นเอาท์พุตจะเลือกอินพุตตาม ตารางการจัดกำหนดการ round-robin ของลำดับความสำคัญที่เลือก และส่ง grant พร้อมด้วยลำดับความสำคัญ

ของเซลล์ดังกล่าว ไปยังอินพุตตัวที่เลือก และรอรับ accept จากขั้นที่ 3 จากนั้นตัวชี้ตำแหน่งในตารางการจัดกำหนดการ round robin ของเอาท์พุตที่ลำดับความสำคัญนั้นๆ จะเปลี่ยนแปลงโดยเพิ่มขึ้นทีละหนึ่งลำดับวนไปเรื่อยๆแบบ โมดูโล เมื่อได้รับ accept ตอบรับจากอินพุตดังกล่าว

ขั้นที่ 3 : accept เมื่อเอาท์พุตได้รับ request และลำดับความสำคัญจากอินพุตแล้ว เอาท์พุตตัวดังกล่าวจะพิจารณาว่ามีการแย่งชิงกันของเซลล์ที่มีลำดับความสำคัญที่แตกต่างกันหรือไม่ หากไม่ อินพุตจะเลือกเอาท์พุตตาม ตารางการจัดกำหนดการ round-robin ของลำดับความสำคัญนั้นๆ และส่ง accept ไปยังเอาท์พุตตัวที่เลือก และทำการส่งเซล์ที่ได้รับเลือกจากอินพุตนั้นไปยังเอาท์พุตที่เลือก จากนั้นตัวชี้ตำแหน่งในตารางการจัดกำหนดการ round robin ของอินพุตที่ลำดับความสำคัญนั้นๆ จะเปลี่ยนแปลงโดยเพิ่มขึ้นทีละหนึ่งลำดับวนไปเรื่อยๆ แบบโมดูโล แต่หากมีการแย่งชิงกันของเซลล์ที่มีลำดับความสำคัญที่แตกต่างกันแล้ว อินพุตจะทำการสุ่มเลือกลำดับความสำคัญโดยให้น้ำหนักในการเลือกที่เท่ากัน จากนั้นอินพุตจะเลือกเอาท์พุตตาม ตารางการจัดกำหนดการ round-robin ของลำดับความสำคัญที่เลือกได้ และส่ง accept ไปยังเอาท์พุตตัวที่เลือก จากนั้นจะทำการส่งเซล์ที่ได้รับเลือกจากอินพุตนั้นไปยังเอาท์พุตที่เลือก จากนั้นตัวชี้ตำแหน่งในตารางการจัดกำหนดการ round robin ของอินพุตที่ลำดับความสำคัญนั้นๆ จะเปลี่ยนแปลงโดยเพิ่มขึ้นทีละหนึ่งลำดับวนไปเรื่อยๆ แบบโมดูโล

3) ควบคุมที่ accept

หลังจากเราได้ควบคุมสัดส่วนการให้บริการที่ขั้นตอนการตัดสินใจในสองลำดับแรกแล้ว จากนี้เราอยากทราบว่าหากควบคุมสัดส่วนการให้บริการที่ขั้นตอน accept ซึ่งเป็นขั้นตอนสุดท้ายแล้วนั้น จะเป็นการตัดสินใจที่ช้าเกินไปหรือไม่และมีผลต่อสมรรถนะ และการควบคุมสัดส่วนการให้บริการอย่างไร และเช่นเดียวกันกับอัลกอริทึมการควบคุมที่ขั้นตอน grant เราจะไม่ควบคุมสัดส่วนการให้บริการในขั้นตอน request ด้วยเหตุผลเช่นเดียวกัน แต่เราจำเป็นต้องควบคุมสัดส่วนการให้บริการที่ขั้นตอน grant เนื่องจากจะทำให้อัลกอริทึมการจัดเส้นทางให้กับเซลล์คลาสเดียวกันยังคงรูปแบบของอัลกอริทึม iSLIP อยู่

ขั้นตอนการทำงานของอัลกอริทึม

ขั้นที่1 : request เมื่อมีเซลล์เข้ามาในบัฟเฟอร์ของอินพุต อินพุตดังกล่าวจะจัดเซลล์นั้นไว้ยัง VOQ ของลำดับความสำคัญของเซลล์นั้นๆ จากนั้นอินพุตแต่ละตัวจะพิจารณาว่ามีเซลล์ใน VOQ ที่ต้องการจะส่งหรือไม่ ถ้ามี ก็จะทำการส่ง request และลำดับความสำคัญของเซลล์ที่ต้องการจะส่งไปยังเอาท์พุตที่ต้องการ โดยไม่คำนึงถึงลำดับความสำคัญของเซลล์

ขั้นที่ 2 : grant เมื่อเอาท์พุตได้รับ request และลำดับความสำคัญจากอินพุตแล้ว เอาท์พุตตัวดังกล่าวจะพิจารณาว่ามีการแย่งชิงกันของเซลล์ที่มีลำดับความสำคัญที่แตกต่างกันหรือไม่ หากไม่มี เอาท์พุตจะเลือกอินพุตตาม ตารางการจัดกำหนดการ round-robin ของลำดับความสำคัญนั้นๆ จากนั้นเอาท์พุตจะส่ง grant และลำดับความสำคัญของเซลล์ที่เลือก ไปยังอินพุตตัวที่เลือก และรอ

รับ accept จากขั้นที่ 3 จากนั้นตัวชี้ตำแหน่งในตารางการจัดกำหนดการ round robin ของเอาท์พุตที่ลำดับความสำคัญนั้นๆ จะเปลี่ยนแปลงโดยเพิ่มขึ้นทีละหนึ่งลำดับวนไปเรื่อยๆแบบ โมดูโล เมื่อได้รับ accept ตอบรับจากอินพุตดังกล่าว แต่หากมีการแย่งชิงกันของเซลล์ที่มีลำดับความสำคัญที่แตกต่างกันแล้ว เอาท์พุตจะทำการสุ่มเลือกลำดับความสำคัญของเซลล์ที่จะให้บริการโดยให้น้ำหนักในการเลือกที่เท่ากัน จากนั้นเอาท์พุตจะเลือกอินพุตตาม ตารางการจัดกำหนดการ round-robin ของลำดับความสำคัญที่เลือก และส่ง grant พร้อมด้วยลำดับความสำคัญของเซลล์ดังกล่าวไปยังอินพุตตัวที่เลือก และรอรับ accept จากขั้นที่ 3 จากนั้นตัวชี้ตำแหน่งในตารางการจัดกำหนดการ round robin ของเอาท์พุตที่ลำดับความสำคัญนั้นๆ จะเปลี่ยนแปลงโดยเพิ่มขึ้นทีละหนึ่งลำดับวนไปเรื่อยๆแบบ โมดูโล เมื่อได้รับ accept ตอบรับจากอินพุตดังกล่าว

ขั้นที่ 3 : accept เมื่อเอาท์พุตได้รับ request และลำดับความสำคัญจากอินพุตแล้ว เอาท์พุตตัวดังกล่าวจะพิจารณาว่ามีการแย่งชิงกันของเซลล์ที่มีลำดับความสำคัญที่แตกต่างกันหรือไม่ หากไม่ อินพุตจะเลือกเอาท์พุตตาม ตารางการจัดกำหนดการ round-robin ของลำดับความสำคัญนั้นๆ และส่ง accept ไปยังเอาท์พุตตัวที่เลือก และทำการส่งเซล์ที่ได้รับเลือกจากอินพุตนั้นไปยังเอาท์พุตที่เลือก จากนั้นตัวชี้ตำแหน่งในตารางการจัดกำหนดการ round robin ของอินพุตที่ลำดับความสำคัญนั้นๆ จะเปลี่ยนแปลงโดยเพิ่มขึ้นทีละหนึ่งลำดับวนไปเรื่อยๆ แบบโมดูโล แต่หากมีการแย่งชิงกันของเซลล์ที่มีลำดับความสำคัญที่แตกต่างกันแล้ว อินพุตจะทำการสุ่มเลือกลำดับความสำคัญโดยการสุ่มแบบถ่วงน้ำหนักของแต่ละลำดับความสำคัญ จากนั้นอินพุตจะเลือกเอาท์พุตตาม ตารางการจัดกำหนดการ round-robin ของลำดับความสำคัญที่เลือกได้ และส่ง accept ไปยังเอาท์พุตตัวที่เลือก จากนั้นจะทำการส่งเซล์ที่ได้รับเลือกจากอินพุตนั้นไปยังเอาท์พุตที่เลือก จากนั้นตัวชี้ตำแหน่งในตารางการจัดกำหนดการ round robin ของอินพุตที่ลำดับความสำคัญนั้นๆ จะเปลี่ยนแปลงโดยเพิ่มขึ้นทีละหนึ่งลำดับวนไปเรื่อยๆแบบโมดูโล

4) ควบคุมที่ request และ grant

ในขั้นต้น เราทำการควบคุมสัดส่วนการให้บริการในแต่ละขั้นตอนการตัดสินใจทั้ง 3 ขั้นเพื่อศึกษาผลกระทบต่างๆ ที่เกิดขึ้นในการควบคุมแต่ละขั้นการตัดสินใจแล้ว จากนั้นเราอยากจะทราบอีกว่า หากทำการควบคุมสัดส่วนการให้บริการมากกว่าหนึ่งขั้นตอนพร้อมกัน จะมีผลต่อสมรรถนะ และการควบคุมสัดส่วนการให้บริการอย่างไร มีความแตกต่างจากอัลกอริทึมมีควบคุมเพียงขั้นตอนเดียวหรือไม่อย่างไร การเพิ่มขึ้นของแต่ละขั้นตอนการตัดสินใจที่ต้องการควบคุมมีผลอย่างไร ข้อดีและข้อเสียที่ได้เป็นอย่างไร ซึ่งอันดับแรกเราจะควบคุมที่ส่วนต้นของขั้นตอนการตัดสินใจ นั่นคือส่วนของ request และ grant เพื่อจะทดสอบว่าในการควบคุมที่รุนแรงตั้งแต่ต้นจะมีผลอย่างไร ต่อสมรรถนะ และการควบคุมสัดส่วนการให้บริการ สัดส่วนการให้บริการดังกล่าวจะถูกควบคุมโดยวิธีการสุ่มแบบถ่วงน้ำหนักโดยในแต่ละขั้นตอนที่มีการตัดสินใจด้วยการสุ่มแบบถ่วงน้ำหนักนั้น น้ำหนักของการสุ่มที่ใช้จะเท่ากัน

ขั้นตอนการทำงานของอัลกอริทึม

ขั้นที่1 : request เมื่อมีเซลล์เข้ามาในบัฟเฟอร์ของอินพุต อินพุตดังกล่าวจะจัดเซลล์นั้นไว้ยัง VOQ ของลำดับความสำคัญของเซลล์นั้นๆ จากนั้นอินพุตแต่ละตัวจะพิจารณาว่ามีเซลล์ใน VOQ ทั้งสองลำดับความสำคัญที่ต้องการจะส่งหรือไม่ ถ้ามี ก็จะพิจารณาว่ามีกี่ลำดับความสำคัญที่จะส่งไปยังเอาท์พุตตัวเดียวกัน หากมีลำดับเดียวก็ทำการส่ง request และลำดับความสำคัญของเซลล์ที่ต้องการจะส่งไปยังเอาท์พุตที่ต้องการ แต่หากมีเซลล์มากกว่าหนึ่งลำดับความสำคัญต้องการจะส่งไปยังเอาท์พุตเดียวกันก็จะเลือกลำดับความสำคัญโดยการสุ่มแบบถ่วงน้ำหนักของแต่ละลำดับความสำคัญ และทำการส่ง request และลำดับความสำคัญของเซลล์ที่เลือกไปยังเอาท์พุตที่ต้องการ

ขั้นที่ 2 : grant เมื่อเอาท์พุตได้รับ request และลำดับความสำคัญจากอินพุตแล้ว เอาท์พุตตัวดังกล่าวจะพิจารณาว่ามีการแย่งชิงกันของเซลล์ที่มีลำดับความสำคัญที่แตกต่างกันหรือไม่ หากไม่มี เอาท์พุตจะเลือกอินพุตตาม ตารางการจัดกำหนดการ round-robin ของลำดับความสำคัญนั้นๆ จากนั้นเอาท์พุตจะส่ง grant และลำดับความสำคัญของเซลล์ที่เลือก ไปยังอินพุตตัวที่เลือก และรอรับ accept จากขั้นที่ 3 จากนั้นตัวชี้ตำแหน่งในตารางการจัดกำหนดการ round robin ของเอาท์พุตที่ลำดับความสำคัญนั้นๆ จะเปลี่ยนแปลงโดยเพิ่มขึ้นทีละหนึ่งลำดับวนไปเรื่อยๆแบบ โมดูโล เมื่อได้รับ accept ตอบรับจากอินพุตดังกล่าว แต่หากมีการแย่งชิงกันของเซลล์ที่มีลำดับความสำคัญที่แตกต่างกันแล้ว เอาท์พุตจะทำการสุ่มเลือกลำดับความสำคัญของเซลล์ที่จะให้บริการโดยการสุ่มแบบถ่วงน้ำหนักของแต่ละลำดับความสำคัญ จากนั้นเอาท์พุตจะเลือกอินพุตตาม ตารางการจัดกำหนดการ round-robin ของลำดับความสำคัญที่เลือก และส่ง grant พร้อมด้วยลำดับความสำคัญของเซลล์ดังกล่าว ไปยังอินพุตตัวที่เลือก และรอรับ accept จากขั้นที่ 3 จากนั้นตัวชี้ตำแหน่งในตารางการจัดกำหนดการ round robin ของเอาท์พุตที่ลำดับความสำคัญนั้นๆ จะเปลี่ยนแปลงโดยเพิ่มขึ้นทีละหนึ่งลำดับวนไปเรื่อยๆ แบบ โมดูโล เมื่อได้รับ accept ตอบรับจากอินพุตดังกล่าว

ขั้นที่ 3 : accept เมื่อเอาท์พุตได้รับ request และลำดับความสำคัญจากอินพุตแล้ว เอาท์พุตตัวดังกล่าวจะพิจารณาว่ามีการแย่งชิงกันของเซลล์ที่มีลำดับความสำคัญที่แตกต่างกันหรือไม่ หากไม่ อินพุตจะเลือกเอาท์พุตตาม ตารางการจัดกำหนดการ round-robin ของลำดับความสำคัญนั้นๆ และส่ง accept ไปยังเอาท์พุตตัวที่เลือก และทำการส่งเซล์ที่ได้รับเลือกจากอินพุตนั้นไปยังเอาท์พุตที่เลือก จากนั้นตัวชี้ตำแหน่งในตารางการจัดกำหนดการ round robin ของอินพุตที่ลำดับความสำคัญนั้นๆ จะเปลี่ยนแปลงโดยเพิ่มขึ้นทีละหนึ่งลำดับวนไปเรื่อยๆ แบบโมดูโล แต่หากมีการแย่งชิงกันของเซลล์ที่มีลำดับความสำคัญที่แตกต่างกันแล้ว อินพุตจะทำการสุ่มเลือกลำดับความสำคัญโดยให้น้ำหนักในการเลือกที่เท่ากัน จากนั้นอินพุตจะเลือกเอาท์พุตตาม ตารางการจัดกำหนดการ round-robin ของลำดับความสำคัญที่เลือกได้ และส่ง accept ไปยังเอาท์พุตตัวที่เลือก จากนั้นจะทำการส่งเซล์ที่ได้รับเลือกจากอินพุตนั้นไปยังเอาท์พุตที่เลือก จากนั้นตัวชี้ตำแหน่งใน

ตารางการจัดกำหนดการ round robin ของอินพุตที่ลำดับความสำคัญนั้นๆ จะเปลี่ยนแปลงโดยเพิ่มขึ้นทีละหนึ่งลำดับวนไปเรื่อยๆ แบบโมดูโล

5) ควบคุมที่ request และ accept

หลังจากควบคุมตั้งแต่ตอนต้นนั่นคือที่ขั้นตอน request และ grant แล้ว จากนั้นเราจะทบลองทำการควบคุมที่ส่วนแรกสุด และส่วนท้ายสุด ของขั้นตอนการตัดสินใจ นั่นคือขั้นตอน request และ accept โดยจะให้โอกาสได้รับบริการที่ขั้นตอน grant (ซึ่งเป็นขั้นตอนที่มีผลกระทบต่อสมรรถนะมากที่สุด) ของเซลล์ทั้งสองคลาสเท่ากัน เพื่อดูผลกระทบที่มีต่อสมรรถนะ และประสิทธิภาพในการควบคุมสัดส่วนการให้บริการ

ขั้นตอนการทำงานของอัลกอริทึม

ขั้นที่1 : request เมื่อมีเซลล์เข้ามาในบัฟเฟอร์ของอินพุต อินพุตดังกล่าวจะจัดเซลล์นั้นไว้ยัง VOQ ของลำดับความสำคัญของเซลล์นั้นๆ จากนั้นอินพุตแต่ละตัวจะพิจารณาว่ามีเซลล์ใน VOQ ทั้งสองลำดับความสำคัญที่ต้องการจะส่งหรือไม่ ถ้ามี ก็จะพิจารณาว่ามีกี่ลำดับความสำคัญที่จะส่งไปยังเอาท์พุตตัวเดียวกัน หากมีลำดับเดียวก็ทำการส่ง request และลำดับความสำคัญของเซลล์ที่ต้องการจะส่งไปยังเอาท์พุตที่ต้องการ แต่หากมีเซลล์มากกว่าหนึ่งลำดับความสำคัญต้องการจะส่งไปยังเอาท์พุตเดียวกันก็จะเลือกลำดับความสำคัญโดยการสุ่มแบบถ่วงน้ำหนักของแต่ละลำดับความสำคัญ และทำการส่ง request และลำดับความสำคัญของเซลล์ที่เลือกไปยังเอาท์พุตที่ต้องการ

ขั้นที่ 2 : grant เมื่อเอาท์พุตได้รับ request และลำดับความสำคัญจากอินพุตแล้ว เอาท์พุตตัวดังกล่าวจะพิจารณาว่ามีการแย่งชิงกันของเซลล์ที่มีลำดับความสำคัญที่แตกต่างกันหรือไม่ หากไม่มี เอาท์พุตจะเลือกอินพุตตาม ตารางการจัดกำหนดการ round-robin ของลำดับความสำคัญนั้นๆ จากนั้นเอาท์พุตจะส่ง grant และลำดับความสำคัญของเซลล์ที่เลือก ไปยังอินพุตตัวที่เลือก และรอรับ accept จากขั้นที่ 3 จากนั้นตัวชี้ตำแหน่งในตารางการจัดกำหนดการ round robin ของเอาท์พุตที่ลำดับความสำคัญนั้นๆ จะเปลี่ยนแปลงโดยเพิ่มขึ้นทีละหนึ่งลำดับวนไปเรื่อยๆแบบ โมดูโล เมื่อได้รับ accept ตอบรับจากอินพุตดังกล่าว แต่หากมีการแย่งชิงกันของเซลล์ที่มีลำดับความสำคัญที่แตกต่างกันแล้ว เอาท์พุตจะทำการสุ่มเลือกลำดับความสำคัญของเซลล์ที่จะให้บริการโดยให้น้ำหนักในการเลือกที่เท่ากัน จากนั้นเอาท์พุตจะเลือกอินพุตตาม ตารางการจัดกำหนดการ round-robin ของลำดับความสำคัญที่เลือก และส่ง grant พร้อมด้วยลำดับความสำคัญของเซลล์ดังกล่าวไปยังอินพุตตัวที่เลือก และรอรับ accept จากขั้นที่ 3 จากนั้นตัวชี้ตำแหน่งในตารางการจัดกำหนดการ round robin ของเอาท์พุตที่ลำดับความสำคัญนั้นๆ จะเปลี่ยนแปลงโดยเพิ่มขึ้นทีละหนึ่งลำดับวนไปเรื่อยๆ แบบ โมดูโล เมื่อได้รับ accept ตอบรับจากอินพุตดังกล่าว

ขั้นที่ 3 : accept เมื่อเอาท์พุตได้รับ request และลำดับความสำคัญจากอินพุตแล้ว เอาท์พุตตัวดังกล่าวจะพิจารณาว่ามีการแย่งชิงกันของเซลล์ที่มีลำดับความสำคัญที่แตกต่างกันหรือไม่ หากไม่ อินพุตจะเลือกเอาท์พุตตาม ตารางการจัดกำหนดการ round-robin ของลำดับความสำคัญนั้นๆ และส่ง accept ไปยังเอาท์พุตตัวที่เลือก และทำการส่งเซล์ที่ได้รับเลือกจากอินพุตนั้นไปยังเอาท์พุตที่เลือก จากนั้นตัวชี้ตำแหน่งในตารางการจัดกำหนดการ round robin ของอินพุตที่ลำดับความสำคัญนั้นๆ จะเปลี่ยนแปลงโดยเพิ่มขึ้นทีละหนึ่งลำดับวนไปเรื่อยๆ แบบโมดูโล แต่หากมีการแย่งชิงกันของเซลล์ที่มีลำดับความสำคัญที่แตกต่างกันแล้ว อินพุตจะทำการสุ่มเลือกลำดับความสำคัญ โดยการสุ่มแบบถ่วงน้ำหนักของแต่ละลำดับความสำคัญ จากนั้นอินพุตจะเลือกเอาท์พุตตาม ตารางการจัดกำหนดการ round-robin ของลำดับความสำคัญที่เลือกได้ และส่ง accept ไปยังเอาท์พุตตัวที่เลือก จากนั้นจะทำการส่งเซล์ที่ได้รับเลือกจากอินพุตนั้นไปยังเอาท์พุตที่เลือก จากนั้นตัวชี้ตำแหน่งในตารางการจัดกำหนดการ round robin ของอินพุตที่ลำดับความสำคัญนั้นๆ จะเปลี่ยนแปลงโดยเพิ่มขึ้นทีละหนึ่งลำดับวนไปเรื่อยๆ แบบโมดูโล

6) ควบคุมที่ grant และ accept

หลังจากการทดลองควบคุมที่ขั้นตอนแรกสุดนั่นคือขั้นตอน request รวมกับอีกสองขั้นตอนที่เหลือแล้ว เราอยากทราบอีกว่าหากเราปล่อยให้มีเซลล์เข้ามาในขั้นตอนการตัดสินใจเพิ่มขึ้นเพื่อเพิ่มโอกาสที่จะได้รับทรูพุตที่ดีขึ้น จากนั้นจึงทดลองควบคุมที่ขั้นตอนในการตัดสินใจใน 2 ส่วนหลัง นั่นคือ grant และ accept จะได้ผลที่แตกต่างจากวิธีอื่นโดยเฉพาะอัลกอริทึมการควบคุมที่ grant หรือไม่อย่างไร

ขั้นตอนการทำงานของอัลกอริทึม

ขั้นที่1 : request เมื่อมีเซลล์เข้ามาในบัฟเฟอร์ของอินพุต อินพุตดังกล่าวจะจัดเซลล์นั้นไว้ยัง VOQ ของลำดับความสำคัญของเซลล์นั้นๆ จากนั้นอินพุตแต่ละตัวจะพิจารณาว่ามีเซลล์ใน VOQ ที่ต้องการจะส่งหรือไม่ ถ้ามี ก็จะทำการส่ง request และลำดับความสำคัญของเซลล์ที่ต้องการจะส่งไปยังเอาท์พุตที่ต้องการ โดยไม่คำนึงถึงลำดับความสำคัญของเซลล์

ขั้นที่ 2 : grant เมื่อเอาท์พุตได้รับ request และลำดับความสำคัญจากอินพุตแล้ว เอาท์พุตตัวดังกล่าวจะพิจารณาว่ามีการแย่งชิงกันของเซลล์ที่มีลำดับความสำคัญที่แตกต่างกันหรือไม่ หากไม่มี เอาท์พุตจะเลือกอินพุตตาม ตารางการจัดกำหนดการ round-robin ของลำดับความสำคัญนั้นๆ จากนั้นเอาท์พุตจะส่ง grant และลำดับความสำคัญของเซลล์ที่เลือก ไปยังอินพุตตัวที่เลือก และรอรับ accept จากขั้นที่ 3 จากนั้นตัวชี้ตำแหน่งในตารางการจัดกำหนดการ round robin ของเอาท์พุตที่ลำดับความสำคัญนั้นๆ จะเปลี่ยนแปลงโดยเพิ่มขึ้นทีละหนึ่งลำดับวนไปเรื่อยๆแบบ โมดูโล เมื่อได้รับ accept ตอบรับจากอินพุตดังกล่าว แต่หากมีการแย่งชิงกันของเซลล์ที่มีลำดับความสำคัญที่แตกต่างกันแล้ว เอาท์พุตจะทำการสุ่มเลือกลำดับความสำคัญของเซลล์ที่จะให้บริการ โดยการสุ่มแบบถ่วงน้ำหนักของแต่ละลำดับความสำคัญ จากนั้นเอาท์พุตจะเลือกอินพุตตาม ตารางการจัด

กำหนดการ round-robin ของลำดับความสำคัญที่เลือก และส่ง grant พร้อมด้วยลำดับความสำคัญของเซลล์ดังกล่าว ไปยังอินพุตตัวที่เลือก และรอรับ accept จากขั้นที่ 3 จากนั้นตัวชี้ตำแหน่งในตารางการจัดกำหนดการ round robin ของเอาท์พุตที่ลำดับความสำคัญนั้นๆ จะเปลี่ยนแปลงโดยเพิ่มขึ้นทีละหนึ่งลำดับวนไปเรื่อยๆ แบบ โมดูโล เมื่อได้รับ accept ตอบรับจากอินพุตดังกล่าว

ขั้นที่ 3 : accept เมื่อเอาท์พุตได้รับ request และลำดับความสำคัญจากอินพุตแล้ว เอาท์พุตตัวดังกล่าวจะพิจารณาว่ามีการแย่งชิงกันของเซลล์ที่มีลำดับความสำคัญที่แตกต่างกันหรือไม่ หากไม่ อินพุตจะเลือกเอาท์พุตตาม ตารางการจัดกำหนดการ round-robin ของลำดับความสำคัญนั้นๆ และส่ง accept ไปยังเอาท์พุตตัวที่เลือก และทำการส่งเซล์ที่ได้รับเลือกจากอินพุตนั้นไปยังเอาท์พุตที่เลือก จากนั้นตัวชี้ตำแหน่งในตารางการจัดกำหนดการ round robin ของอินพุตที่ลำดับความสำคัญนั้นๆ จะเปลี่ยนแปลงโดยเพิ่มขึ้นทีละหนึ่งลำดับวนไปเรื่อยๆ แบบโมดูโล แต่หากมีการแย่งชิงกันของเซลล์ที่มีลำดับความสำคัญที่แตกต่างกันแล้ว อินพุตจะทำการสุ่มเลือกลำดับความสำคัญโดยการสุ่มแบบถ่วงน้ำหนักของแต่ละลำดับความสำคัญ จากนั้นอินพุตจะเลือกเอาท์พุตตาม ตารางการจัดกำหนดการ round-robin ของลำดับความสำคัญที่เลือกได้ และส่ง accept ไปยังเอาท์พุตตัวที่เลือก จากนั้นจะทำการส่งเซล์ที่ได้รับเลือกจากอินพุตนั้นไปยังเอาท์พุตที่เลือก จากนั้นตัวชี้ตำแหน่งในตารางการจัดกำหนดการ round robin ของอินพุตที่ลำดับความสำคัญนั้นๆ จะเปลี่ยนแปลงโดยเพิ่มขึ้นทีละหนึ่งลำดับวนไปเรื่อยๆแบบโมดูโล

7) ควบคุมที่ request, grant และ accept

ในตอนต้นเราทำการควบคุมสัดส่วนการให้บริการในแต่ละส่วนของขั้นตอนการตัดสินใจ จากนั้นเราจึงทำการควบคุมร่วมกันระหว่าง 2 ขั้นตอนการตัดสินใจเพื่อเปรียบเทียบข้อแตกต่าง และข้อดีข้อเสียของการตัดสินใจร่วนกัน ในลำดับสุดท้ายนี้เราต้องการทราบว่าในการควบคุมที่รุนแรงที่สุด นั่นคือการควบคุมสัดส่วนการให้บริการในทั้ง 3 ขั้นตอนการตัดสินใจ จะมีผลกระทบอย่างไร และมากน้อยเพียงใด ผลที่ได้แตกต่างจากอัลกอริทึมอื่นอย่างไร มีการรวมข้อดีและข้อเสียของแต่ละอัลกอริทึมหรือไม่อย่างไร

ขั้นตอนการทำงานของอัลกอริทึม

ขั้นที่1 : request เมื่อมีเซลล์เข้ามาในบัฟเฟอร์ของอินพุต อินพุตดังกล่าวจะจัดเซลล์นั้นไว้ยัง VOQ ของลำดับความสำคัญของเซลล์นั้นๆ จากนั้นอินพุตแต่ละตัวจะพิจารณาว่ามีเซลล์ใน VOQ ทั้งสองลำดับความสำคัญที่ต้องการจะส่งหรือไม่ ถ้ามี ก็จะพิจารณาว่ามีกี่ลำดับความสำคัญที่จะส่งไปยังเอาท์พุตตัวเดียวกัน หากมีลำดับเดียวก็ทำการส่ง request และลำดับความสำคัญของเซลล์ที่ต้องการจะส่งไปยังเอาท์พุตที่ต้องการ แต่หากมีเซลล์มากกว่าหนึ่งลำดับความสำคัญต้องการจะส่งไปยังเอาท์พุตเดียวกันก็จะเลือกลำดับความสำคัญโดยการสุ่มแบบถ่วงน้ำหนักของแต่ละลำดับความสำคัญ และทำการส่ง request และลำดับความสำคัญของเซลล์ที่เลือกไปยังเอาท์พุตที่ต้องการ

ขั้นที่ 2 : grant เมื่อเอาท์พุตได้รับ request และลำดับความสำคัญจากอินพุตแล้ว เอาท์พุตตัวดังกล่าวจะพิจารณาว่ามีการแย่งชิงกันของเซลล์ที่มีลำดับความสำคัญที่แตกต่างกันหรือไม่ หากไม่มี เอาท์พุตจะเลือกอินพุตตาม ตารางการจัดกำหนดการ round-robin ของลำดับความสำคัญนั้นๆ จากนั้นเอาท์พุตจะส่ง grant และลำดับความสำคัญของเซลล์ที่เลือก ไปยังอินพุตตัวที่เลือก และรอรับ accept จากขั้นที่ 3 จากนั้นตัวชี้ตำแหน่งในตารางการจัดกำหนดการ round robin ของเอาท์พุตที่ลำดับความสำคัญนั้นๆ จะเปลี่ยนแปลงโดยเพิ่มขึ้นทีละหนึ่งลำดับวนไปเรื่อยๆ แบบ โมดูโล เมื่อได้รับ accept ตอบรับจากอินพุตดังกล่าว แต่หากมีการแย่งชิงกันของเซลล์ที่มีลำดับความสำคัญที่แตกต่างกันแล้ว เอาท์พุตจะทำการสุ่มเลือกลำดับความสำคัญของเซลล์ที่จะให้บริการโดยการสุ่มแบบถ่วงน้ำหนักของแต่ละลำดับความสำคัญ จากนั้นเอาท์พุตจะเลือกอินพุตตาม ตารางการจัดกำหนดการ round-robin ของลำดับความสำคัญที่เลือก และส่ง grant พร้อมด้วยลำดับความสำคัญของเซลล์ดังกล่าว ไปยังอินพุตตัวที่เลือก และรอรับ accept จากขั้นที่ 3 จากนั้นตัวชี้ตำแหน่งในตารางการจัดกำหนดการ round robin ของเอาท์พุตที่ลำดับความสำคัญนั้นๆ จะเปลี่ยนแปลงโดยเพิ่มขึ้นทีละหนึ่งลำดับวนไปเรื่อยๆ แบบ โมดูโล เมื่อได้รับ accept ตอบรับจากอินพุตดังกล่าว

ขั้นที่ 3 : accept เมื่อเอาท์พุตได้รับ request และลำดับความสำคัญจากอินพุตแล้ว เอาท์พุตตัวดังกล่าวจะพิจารณาว่ามีการแย่งชิงกันของเซลล์ที่มีลำดับความสำคัญที่แตกต่างกันหรือไม่ หากไม่ อินพุตจะเลือกเอาท์พุตตาม ตารางการจัดกำหนดการ round-robin ของลำดับความสำคัญนั้นๆ และส่ง accept ไปยังเอาท์พุตตัวที่เลือก และทำการส่งเซล์ที่ได้รับเลือกจากอินพุตนั้นไปยังเอาท์พุตที่เลือก จากนั้นตัวชี้ตำแหน่งในตารางการจัดกำหนดการ round robin ของอินพุตที่ลำดับความสำคัญนั้นๆ จะเปลี่ยนแปลงโดยเพิ่มขึ้นทีละหนึ่งลำดับวนไปเรื่อยๆแบบโมดูโล แต่หากมีการแย่งชิงกันของเซลล์ที่มีลำดับความสำคัญที่แตกต่างกันแล้ว อินพุตจะทำการสุ่มเลือกลำดับความสำคัญโดยการสุ่มแบบถ่วงน้ำหนักของแต่ละลำดับความสำคัญ จากนั้นอินพุตจะเลือกเอาท์พุตตาม ตารางการจัดกำหนดการ round-robin ของลำดับความสำคัญที่เลือกได้ และส่ง accept ไปยังเอาท์พุตตัวที่เลือก จากนั้นจะทำการส่งเซล์ที่ได้รับเลือกจากอินพุตนั้นไปยังเอาท์พุตที่เลือก จากนั้นตัวชี้ตำแหน่งในตารางการจัดกำหนดการ round robin ของอินพุตที่ลำดับความสำคัญนั้นๆ จะเปลี่ยนแปลงโดยเพิ่มขึ้นทีละหนึ่งลำดับวนไปเรื่อยๆ แบบโมดูโล

7. งานส่วนที่จะดำเนินการต่อไป

ส่วนที่ 1

1) ศึกษาเทคนิคการบันทึกข้อมูลบนตัวกลางแม่เหล็กแบบตั้งฉาก (perpendicular recording) ที่สามารถเพิ่มความจุในการบันทึกข้อมูลได้สูง
2) ศึกษาการบันทึกข้อมูลบนตัวกลางแบบอื่น เช่น การบันทึกข้อมูลบนแผ่นซีดี และดีวีดี เป็นต้น

ส่วนที่ 2

1) ศึกษาอัลกอริทึมการจัดกำหนดการสำหรับการจัดสรรบริการในเซลล์สวิตช์ที่พิจารณาลำดับความสำคัญของเซลล์
2) เขียนโปรแกรมของอัลกอริทึมการจัดกำหนดการสำหรับการจัดสรรบริการในเซลล์สวิตช์ที่พิจารณาลำดับความสำคัญของเซลล์
3) ศึกษาวิธีแก้ปัญหาคอขวดในการสวิตช์ข้อมูลโดยใช้เทคนิค envelope

8. รายการอ้างอิง

[1] R.E. Jones, M.H. Kryder, "Storing magnetic data," *IEEE Potentials*, vol. 18, no. 4, pp. 17 – 20, Oct.-Nov. 1999.

[2] P. H. Siegel and J. K. Wolf, "Modulation and coding for information storage," *IEEE Commun. Magazine*, pp. 68–86, Dec. 1991.

[3] H. Kobayashi and D. Tang, "Application of partial response channel coding to magnetic recording systems," *IBM J. Res. Dev.*, pp. 368–375, July 1970.

[4] P. Kabal and S. Pasupathy, "Partial-response signaling," *IEEE Trans. Commun.*, vol. COM-23, no. 9, pp. 921-934, Sep. 1975.

[5] H. Kobayashi, "Application of Probabilistic Decoding to Digital Magnetic Recording Systems," *IBM J. Res. Dev.*, vol. 15, pp. 64–74, Jan. 1971.

[6] R. W. Wood and D. A. Petersen, "Viterbi detection of class IV partial response on a magnetic recording channel," *IEEE Trans. Commun.*, vol. COM-34, no. 5, pp. 454–461, May 1986.

[7] H. K. Thapar and A. M. Patel, "A class of partial response systems for increasing storage density in magnetic recording," *IEEE Trans. Magn.*, vol. MAG-23, no. 5, pp. 3666–3668, Sept. 1987.

[8] W. E. Ryan and B. Zafer, "A study of class I partial response signaling for magnetic recording," *IEEE Trans. Magn.*, vol. 33, no. 6, pp. 4543–4550, Nov. 1997.

[9] Y. Wu and J. R. Cruz, "Noise predictive turbo systems," *IEEE Trans. Magn.*, vol. 37, no. 2, pp. 742–747, Mar. 2001

[10] P. R. Chevillat, E. Eleftheriou, and D. Maiwald, "Noise-predictive partial-response equalizers and applications," *in Proc. IEEE Int. Conf. Commun. (ICC'92)*, vol. 2, pp. 942–947, June 1992.

[11] E. Eleftheriou and W. Hirt, "Improving Performance of PRML/EPRML through noise prediction," *IEEE Trans. Magn.*, vol. 22, no. 5, pp. 3968–3970, Sept. 1996.

[12] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: turbo codes," *in Proc. IEEE Int. Conf. Comm. (ICC'93)*, Geneva, Switzerland, May 1993, pp. 1064–1070.

[13] C. Berrou and A. Glavieux, "Near optimum error correcting coding and decoding: turbo-codes," *IEEE Trans. Commun.*, vol. 44, no. 10, pp. 1261-1271, Oct. 1996.

[14] W. L. Abbott, J. M. Cioffi, and H. K. Thapar, "Performance of digital magnetic recording with equalization and offtrack interference," *IEEE Trans. Magn.*, vol. 27, no. 1, Jan 1991.

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย