# การดำเนินการพื้นฐานเลขคณิตสำหรับระบบจำนวนมอดุลาร์ซ้ำซ้อน

นายสัพพชัย อยู่เย็น



วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
สาขาวิชาวิทยาศาสตร์คอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2551

# FUNDAMENTAL ARITHMETIC OPERATION IN REDUNDANT MODULAR NUMBER SYSTEM

Mr.Sappachai Yuyen



A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Science Program in Computer Science
Department of Computer Engineering
Chulalongkorn University
Academic Year 2008

| | |
|---|---|
| หัวข้อวิทยานิพนธ์ | การดำเนินการพื้นฐานเลขคณิตสำหรับระบบจำนวนมอดุลาร์ซ้ำซ้อน |
| โดย | นายสัพพชัย อยู่เย็น |
| สาขาวิชา | วิทยาศาสตร์คอมพิวเตอร์ |
| อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก | ผู้ช่วยศาสตราจารย์ ดร.อรรถสิทธิ์ สุรฤกษ์ |

---

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญามหาบัณฑิต

..... คณบดีคณะวิศวกรรมศาสตร์
(รองศาสตราจารย์ ดร.บุญสม เลิศหิรัญวงศ์)

คณะกรรมการสอบวิทยานิพนธ์

..... ประธานกรรมการ
(ศาสตราจารย์ ดร.ประภาส จงสถิตย์วัฒนา)

..... อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก
(ผู้ช่วยศาสตราจารย์ ดร.อรรถสิทธิ์ สุรฤกษ์)

..... กรรมการ
(ผู้ช่วยศาสตราจารย์ ดร.พิษณุ คนองชัยยศ)

..... กรรมการ
(ผู้ช่วยศาสตราจารย์ ดร.อานนท์ รุ่งสว่าง)

สัพพชัย อยู่เย็น : การดำเนินการพื้นฐานเลขคณิตสำหรับระบบจำนวนมอดุลาร์ซ้ำซ้อน (FUNDAMENTAL ARITHMETIC OPERATION IN REDUNDANT MODULAR NUMBER SYSTEM). อ.ที่ปรึกษาวิทยานิพนธ์หลัก : ผศ.ดร.อรรถสิทธิ์ สุรฤกษ์, 43 หน้า

ระบบจำนวนมอดุลาร์แบบพหุนามเหมาะสมสำหรับการคำนวณที่รวดเร็ว การดำเนินการพื้นฐานเลขคณิตสามารถดำเนินการโดยใช้รูปแบบความซ้ำซ้อนของพหุนามที่มีค่าเป็นศูนย์ แต่การบวกและการลบไม่สามารถรับประกันได้ว่าจะสิ้นสุดภายในเวลาคงที่ ความซับซ้อนเชิงเวลาได้รับการพิสูจน์ว่าแปรผันตรงตามจำนวนตัวเลข ในงานวิจัยนี้ เราได้เสนออัลกอริทึมใหม่สำหรับการบวก การลบ และการคูณ (อาจเรียกว่า การแปลงชุดตัวเลข) ซึ่งแนวคิดของเราสำหรับอัลกอริทึมนี้สนใจตัวทดทุกตัวที่เป็นไปได้ และระบุความสัมพันธ์ของตัวทดที่เป็นฟังก์ชันประกอบอย่างชัดเจน ผลลัพธ์ทางทฤษฎีแสดงให้เห็นว่าผลลัพธ์จากอัลกอริทึมของเราได้รูปแบบแทนจำนวนที่มีคุณสมบัติที่ต้องการ ความซับซ้อนเชิงเวลาในการคำนวณนั้นถูกแสดงให้เห็นว่าลดลง ซึ่งจะเป็นค่าคงที่เมื่อจำนวนหลักของรูปแบบแทนจำนวนเป็นค่าคงที่



ภาควิชา วิศวกรรมคอมพิวเตอร์ ลายมือชื่อนิสิต สัพพชัย อยู่เย็น
สาขาวิชา วิทยาศาสตร์คอมพิวเตอร์ ลายมือชื่ออ.ที่ปรึกษาวิทยานิพนธ์หลัก Athasit Surarerks
ปีการศึกษา 2551

## 4870511721 : MAJOR COMPUTER SCIENCE
KEYWORDS : NUMBER SYSTEM / MODULAR ARITHMETIC / LATTICE THEORY / MERSENNE NUMBER / CRYPTOGRAPHY

SAPPACHAI YUYEN : FUNDAMENTAL ARITHMETIC OPERATION IN REDUNDANT MODULAR NUMBER SYSTEM. ADVISOR: ASST. PROF. ATHASIT SURARERKS, Ph.D., 43 pp.

A polynomial modular number system is shown to be suitable for fast computation. Fundamental arithmetic operations can be performed using zero-polynomial redundant form. Unfortunately, addition and subtraction cannot be guaranteed to terminate within a constant time. Time complexity is proved to be linear on the number of digits. In this thesis, we propose a novel algorithm for addition, subtraction and multiplication (i.e., probably called a digit-set conversion) where our concept of the algorithm is to focus on all possible carries, and to describe their relationship which is expressed by a composite function. Theoretical result shows that the result obtained from our algorithm always satisfy the representation property. Computational time complexity is demonstrated to be decreased where the base of the system is fixed.



Department: Computer Engineering Student's Signature: .....
Field of Study: Computer Science Advisor's Signature: .....
Academic Year: 2008

# กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงไปได้ด้วยความเมตตาและความช่วยเหลืออย่างดียิ่งจาก ผู้ช่วยศาสตราจารย์ ดร.อรรถสิทธิ์ สุรฤกษ์ อาจารย์ที่ปรึกษา ซึ่งเป็นผู้ให้ข้อคิด แนวทางการแก้ปัญหาและคำปรึกษา ตลอดจนเป็นผู้ตรวจทานแก้ไขในส่วนที่บกพร่องต่างๆ ทำให้วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วง

ขอขอบพระคุณ ศาสตราจารย์ประภาส จงสถิตย์วัฒนา ประธานกรรมการสอบวิทยานิพนธ์ ผู้ช่วยศาสตราจารย์ ดร.พิษณุ คนองชัยยศ และผู้ช่วยศาสตราจารย์ ดร.อานนท์ รุ่งสว่าง กรรมการสอบวิทยานพนธ์ ที่ได้กรุณาให้คำแนะนำในการแก้ไขวิทยานิพนธ์ให้มีคุณภาพยิ่งขึ้น และขอขอบพระคุณคณาจารย์ในภาควิชาวิศวกรรมคอมพิวเตอร์ จุฬาลงกรณ์มหาวิทยาลัยทุกท่านที่ประสิทธิ์ประสาทความรู้อันมีค่ายิ่งแก่ผู้วิจัย

ขอกราบขอบพระคุณ บิดา มารดา และขอขอบคุณพี่ๆ เพื่อนๆ น้องๆ สมาชิกห้องปฏิบัติการ ELITE ทุกคน ที่ผลักดันและให้ความช่วยเหลือในทุกๆ ด้านจนผู้วิจัยสามารถทำวิทยานิพนธ์ฉบับนี้สำเร็จลุล่วง

ท้ายที่สุดนี้ ผู้วิจัยหวังเป็นอย่างยิ่งว่างานวิจัยนี้จะเป็นประโยชน์ต่อผู้ที่สนใจและผู้ที่เกี่ยวข้อง และหากมีข้อผิดพลาดประการใดผู้วิจัยขออภัยไว้ ณ ที่นี้ด้วย

# สารบัญ

หน้า

บทที่ หน้า

# สารบัญตาราง

# สารบัญภาพ

ภาพที่ หน้า

# บทที่ 1

# บทนำ

## 1.1 ความเป็นมาและความสำคัญของปัญหา

ความสามารถในการคำนวณพื้นฐานเลขคณิตของระบบวิทยาการรหัสลับ (cryptography) เป็นสิ่งสำคัญที่ต้องคำนึงถึง เนื่องจากการเข้ารหัสลับในปัจจุบันอาศัยคุณสมบัติของจำนวนเฉพาะที่มีขนาดใหญ่ [1-2] การคำนวณบนจำนวนขนาดใหญ่มักจะมีปัญหาด้านความล่าช้า งานวิจัยจำนวนหนึ่งได้มุ่งเน้นไปที่ระบบการคำนวณที่สามารถจำกัดขนาดของห่วงโซ่การทดได้ ซึ่งระบบจำนวนที่น่าสนใจและถูกนำมาพัฒนาใช้ในวิทยาการรหัสลับคือ ระบบจำนวนส่วนตกค้าง (residue number system : RNS) ซึ่งเป็นระบบจำนวนที่สามารถขจัดการทดได้อย่างมีประสิทธิภาพ ตัวอย่างของอัลกอริทึมการเข้ารหัสลับที่ใช้คุณสมบัติของระบบจำนวนส่วนตกค้าง ได้แก่ อัลกอริทึมที่ปรากฏในงานวิจัยของ ดิฟเฟอร์ และ เฮลแมน [3] และงานวิจัยของ ริเวสต์ ชาร์มี และ อเดลแมน [4] แนวคิดของระบบจำนวนส่วนตกค้างนั้น จะแบ่งจำนวนเต็มที่มีขนาดใหญ่ออกเป็นส่วนย่อยๆ ที่อิสระจากกัน ซึ่งส่วนย่อยแต่ละส่วนเป็นระบบจำนวนจำกัด การแบ่งจำนวนจะอาศัยเศษเหลือจากการหารหรือการสมภาคกันของมอดุลาร์ (ส่วนย่อยแต่ละส่วนเกิดจากการหารหรือการสมภาคกันของมอดุลาร์ของจำนวนเฉพาะแต่ละค่า) การคำนวณพื้นฐานเลขคณิตในคอมพิวเตอร์จะคำนวณบนระบบรูปแบบแทนจำนวนเลขฐาน $\beta$ ซึ่งต้องใช้จำนวนชุดตัวเลขเท่ากับ $\beta$ ดังนั้นแต่ละหลักของรูปแบบแทนจำนวนต้องใช้เนื้อที่สูงตามจำนวนของชุดตัวเลข แสดงว่าการแทนส่วนย่อยของระบบจำนวนส่วนตกค้างต้องใช้เนื้อที่สูงตามจำนวนชุดตัวเลข ต่อมามีงานวิจัยได้เสนอระบบรูปแบบแทนจำนวนของส่วนย่อยดังกล่าว โดยเรียกว่าระบบจำนวนมอดุลาร์ (modular number system : MNS) [5] ซึ่งเป็นระบบที่ลดจำนวนชุดตัวเลขลง แต่จะเพิ่มจำนวนหลักของรูปแบบแทนจำนวนแทน โดยอาศัยคุณสมบัติของการสมภาคกันของมอดุลาร์ นั่นคือจำนวนเต็มที่สมภาคกันของมอดุลาร์สามารถแทนด้วยรูปแบบแทนจำนวนเดียวกันได้ จากการลดจำนวนชุดตัวเลขทำให้แต่ละหลักจะใช้เนื้อที่ในการแทนค่าน้อยลงตามไปด้วย ดังนั้นรูปแบบแทนจำนวนของระบบจำนวนมอดุลาร์จะใช้เนื้อที่ที่ลดลงกว่าเดิม ในงานวิจัยของบาจาร์ด [6-7] ได้เสนออัลกอริทึมการคำนวณพื้นฐานเลขคณิตของระบบจำนวนมอดุลาร์ โดยใช้แนวคิดของความซ้ำซ้อนของศูนย์ ในกรณีของการคำนวณพื้นฐานทางเลขคณิตที่มีรูปแบบแทนจำนวนหลายจำนวน (เช่น การบวกกัน $k$ จำนวน) เวลาจะเพิ่มมากขึ้นตามจำนวนของรูปแบบแทนจำนวนที่นำมาคิดคำนวณ

ในวิทยานิพนธ์นี้ เสนอการดำเนินการพื้นฐานเลขคณิตของระบบจำนวนมอดุลาร์ นั่นคือ การดำเนินการการบวก การลบ และการคูณ โดยจะเรียกการดำเนินการนี้ว่า การดำเนินการพื้นฐานเลขคณิตของระบบจำนวนมอดุลาร์ซ้ำซ้อน ซึ่งพิจารณาในรูปของปัญหาการ

แปลงชุดตัวเลข แนวคิดในการแปลงชุดตัวเลข พิจารณาจากตัวทดที่เป็นไปได้ทั้งหมดตามคุณสมบัติของความซ้ำซ้อนของศูนย์ โดยเวลาที่ใช้ในการคำนวณจะขึ้นอยู่กับความยาวของรูปแบบแทนจำนวนที่ใช้ในระบบ

## 1.2 วัตถุประสงค์ของการวิจัย

ออกแบบการดำเนินการพื้นฐานเลขคณิตของระบบจำนวนมอดุลาร์ซ้ำซ้อน ได้แก่ การบวก การลบ และการคูณ พร้อมทั้งเสนออัลกอริทึมการแปลงชุดตัวเลขให้มีประสิทธิภาพทางด้านเวลาที่ดีขึ้น

## 1.3 ขอบเขตของการวิจัย

1.3.1 ออกแบบการดำเนินการพื้นฐานเลขคณิตของระบบจำนวนมอดุลาร์ซ้ำซ้อน ได้แก่ การบวก การลบ และการคูณ

1.3.2 เสนออัลกอริทึมการแปลงชุดตัวเลขให้มีประสิทธิภาพทางด้านเวลา

## 1.4 ขั้นตอนและวิธีดำเนินการวิจัย

1.4.1 ศึกษางานวิจัยทางด้านระบบจำนวนมอดุลาร์

1.4.2 วิเคราะห์ปัญหาของงานวิจัยที่มีความสอดคล้องกับงานวิจัยที่สนใจ

1.4.3 กำหนดขอบเขตงานวิจัย

1.4.4 ออกแบบการดำเนินการพื้นฐานเลขคณิตของระบบจำนวนมอดุลาร์ซ้ำซ้อน ได้แก่ การบวก การลบ และการคูณ

1.4.5 ออกแบบอัลกอริทึมการแปลงชุดตัวเลขของการบวก การลบ และการคูณ ที่มีประสิทธิภาพทางด้านเวลา

1.4.6 พิสูจน์อัลกอริทึมที่ได้ออกแบบไว้

1.4.7 วิเคราะห์ สรุปผล และจัดทำวิทยานิพนธ์

## 1.5 ประโยชน์ที่คาดว่าจะได้รับจากการวิจัย

1.5.1 ได้การดำเนินการพื้นฐานเลขคณิตของระบบจำนวนมอดุลาร์ซ้ำซ้อน ได้แก่ การบวก การลบ และการคูณ

1.5.2 ได้อัลกอริทึมการแปลงชุดตัวเลขของการบวก การลบ และการคูณ ที่มีประสิทธิภาพทางด้านเวลา

## 1.6 ผลงานที่ตีพิมพ์จากวิทยานิพนธ์

ส่วนหนึ่งของวิทยานิพนธ์นี้ได้รับการตีพิมพ์เป็นบทความทางวิชาการในหัวเรื่องดังต่อไปนี้

1.6.1 "Fast Addition and Subtraction in Redundant Modular Number System" โดย สัพพชัย อยู่เย็น และอรรถสิทธิ์ สุรฤกษ์ ในงานประชุมวิชาการ 12th Annual National Symposium on Computational Science and Engineering (ANSCSE2008) ณ มหาวิทยาลัยอุบลราชธานี จ.อุบลราชธานี ประเทศไทย ระหว่างวันที่ 27-29 มีนาคม พ.ศ. 2551(Abstract Submission)

1.6.2 "Digit-Set Conversion in Modular Number System" โดย สัพพชัย อยู่เย็น และอรรถสิทธิ์ สุรฤกษ์ ในงานประชุมวิชาการ 12th National Computer Science and Engineering Conference (NCSEC12) ณ Long Beach Garden Hotel and Spa พัทยา จ.ชลบุรี ประเทศไทย ระหว่างวันที่ 20-21 พฤศจิกายน พ.ศ. 2551

# บทที่ 2

# ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

## 2.1 ตัวดำเนินการมอดุลาร์ (modular operator)

ตัวดำเนินการมอดุลาร์ถูกเสนอโดย คาร์ล ฟรีดริช เกาส์ ในปี ค.ศ.1801 [8] เลขคณิตมอดุลาร์ (modular arithmetic) เป็นระบบเลขคณิตที่มีรากฐานมาจากระบบจำนวนเต็มทั่วไป แต่จำนวนในระบบนี้มีลักษณะเหมือนกับเข็มนาฬิกา คือ เข็มนาฬิกาจะชี้ไปที่เลข 1 ถึง 12 เมื่อเวลาเกินจากเลข 12 เข็มนาฬิกาก็จะวนกลับชี้ไปที่เลข 1 ซึ่งจะเรียกเลข 12 ว่า มอดุลัส กล่าวคือ ตัวเลขที่มีค่าเกินจากค่ามอดุลัสจะทำการลบออกจนกว่าจะมีค่าน้อยกว่าค่าของมอดุลัส โดยนิยามของมอดุลาร์ $a$ mod $b$ แสดงไว้ในสมการที่ (2.1)

$$a \bmod b = a - b\lfloor a / b \rfloor \tag{2.1}$$

และถ้า $a$ mod $b$ = $c$ mod $b$ จะเขียนได้อีกรูปแบบคือ $a \equiv c \pmod{b}$

**ตัวอย่างที่ 2.1** *แสดงการคำนวณมอดุลาร์ของ 16 mod 5 และ -16 mod 5*

**วิธีทำ** จากสมการที่ (2.1) สามารถคำนวณหาค่า 16 mod 5, -16 mod 5 และ16 mod -5 ได้ดังนี้

$$\begin{aligned} 16 \bmod 5 &= 16 - 5\lfloor 16 / 5 \rfloor \\ &= 16 - 5(3) \\ &= 1 \end{aligned}$$

ซึ่งสามารถเขียนใหม่ได้เป็น $16 \equiv 1 \pmod{5}$

และ
$$\begin{aligned} -16 \bmod 5 &= (-16) - 5\lfloor (-16) / 5 \rfloor \\ &= (-16) - 5(-4) \\ &= 4 \end{aligned}$$

ซึ่งสามารถเขียนใหม่ได้เป็น $-16 \equiv 4 \pmod{5}$ □

## 2.2 ระบบจำนวนส่วนตกค้าง (residue number system)

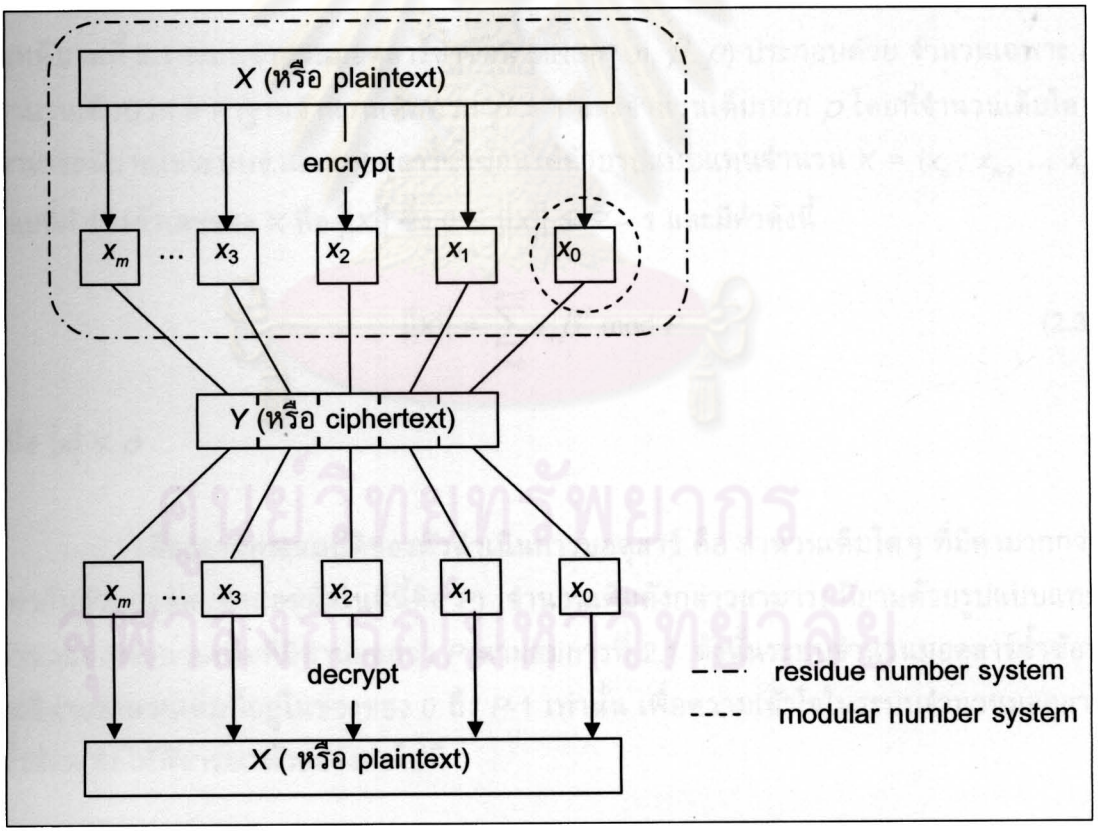
จำนวนเต็มสามารถแทนได้ด้วยระบบจำนวนส่วนตกค้าง โดยอาศัยคุณสมบัติของตัวดำเนินการมอดุลาร์ ซึ่งทุก ๆ จำนวนเต็ม $X$ สามารถแทนในระบบจำนวนส่วนตกค้างด้วยรูปแบบแทนจำนวน $Y$ โดยที่

$$Y = (x_m, x_{m-1}, x_{m-2}, \ldots, x_0)$$

เมื่อ

$$X \equiv x_0 \pmod{P_0} \text{ และ } 0 \leq x_0 < P_0$$
$$X \equiv x_1 \pmod{P_1} \text{ และ } 0 \leq x_1 < P_1$$
$$X \equiv x_2 \pmod{P_2} \text{ และ } 0 \leq x_2 < P_2$$
$$\vdots$$
$$X \equiv x_m \pmod{P_m} \text{ และ } 0 \leq x_m < P_m$$

โดยที่ $P_0\ P_1\ P_2 \ldots P_m$ เป็นจำนวนเฉพาะ




รูปที่ 2.1 แสดงการเข้ารหัสลับโดยระบบจำนวนส่วนตกค้าง

ระบบจำนวนส่วนตกค้างถูกนำไปใช้ในการเข้ารหัสลับ [9-10] (เพื่อความเข้าใจง่ายให้พิจารณารูปที่ 2.1) โดยพิจารณาข้อความต้นฉบับ (plaintext) อยู่ในรูปของจำนวนเต็ม $X$ ที่มี

ขนาดใหญ่ ซึ่งระบบจำนวนส่วนตกค้างจะแบ่งจำนวนเต็ม $X$ ออกเป็นส่วนย่อยๆ โดยคุณสมบัติของตัวดำเนินการมอดุลาร์ สำหรับข้อความเข้ารหัสลับ (ciphertext) หรือจำนวนเต็ม $Y$ คือ จำนวนเต็มย่อย $x_i$ เมื่อ $i = 0,\ldots,m$ โดยจะเรียกว่า ระบบจำนวนมอดุลาร์ ซึ่งจะได้ศึกษาในส่วนต่อไปนี้

## 2.3 ระบบจำนวนมอดุลาร์ซ้ำซ้อน (redundant modular number system)

จำนวนเต็มใดๆ สามารถนิยามในระบบจำนวนทั่วไปได้ด้วยรูปแบบแทนจำนวน $X = (x_{n-1}\ x_{n-2}\ \ldots\ x_0)$ โดยค่าเชิงตัวเลขของ $X$ มีค่าดังสมการที่ (2.2)

$$||X|| = \sum_{i=0}^{n-1} x_i \beta^i \tag{2.2}$$

เมื่อ $\beta$ เป็นเลขฐาน และ $n$ เป็นจำนวนหลักของรูปแบบแทนจำนวน ซึ่งในระบบจำนวนมอดุลาร์ซ้ำซ้อน [5] จะมีการนิยามในลักษณะที่คล้ายกัน โดยจะมีการเพิ่มตัวดำเนินการมอดุลาร์เข้าไป นิยามของระบบจำนวนมอดุลาร์ซ้ำซ้อนมีนิยามดังนี้

**บทนิยามที่ 2.1** ระบบจำนวนมอดุลาร์ซ้ำซ้อน MNS($P$, $n$, $\beta$, $\rho$) ประกอบด้วย จำนวนเฉพาะ $P$ จำนวนเต็มบวก $n$ ค่าฐานจำนวนเต็มบวก $\beta > 1$ และจำนวนเต็มบวก $\rho$ โดยที่จำนวนเต็มใดๆ สามารถนิยามในระบบจำนวนมอดุลาร์ซ้ำซ้อนได้ด้วยรูปแบบแทนจำนวน $X = (x_{n-1}\ x_{n-2}\ \ldots\ x_0)$ โดยที่ค่าเชิงตัวเลขของ $X$ คือ $||X||$ ซึ่ง $0 \le ||X|| \le P - 1$ และมีค่าดังนี้

$$||X|| = \sum_{i=0}^{n-1} x_i \beta^i \bmod P \tag{2.3}$$

เมื่อ $|x_i| < \rho$

เนื่องจากคุณสมบัติของตัวดำเนินการมอดุลาร์ คือ จำนวนเต็มใดๆ ที่มีค่ามากกว่าเท่ากับค่ามอดุลัส (ค่ามอดุลัสในที่นี้คือ $P$) จำนวนเต็มดังกล่าวสามารถนิยามด้วยรูปแบบแทนจำนวนของจำนวนเต็มที่มีค่าน้อยกว่า $P$ ตามสมการที่ 2.1 ดังนั้นระบบจำนวนมอดุลาร์ซ้ำซ้อนจะนิยามจำนวนเต็มที่อยู่ในช่วงของ 0 ถึง $P$-1 เท่านั้น เพื่อความเข้าใจในระบบจำนวนมอดุลาร์ซ้ำซ้อน ขอให้พิจารณาตัวอย่างต่อไปนี้

**ตัวอย่างที่ 2.2** *ระบบจำนวนมอดุลาร์ซ้ำซ้อน MNS(37, 4, 8, 2) ซึ่งสามารถนิยามจำนวนเต็มใดๆ ตั้งแต่ 0 ถึง 36 ได้ดังนี้*

**วิธีทำ** จาก $\rho = 2$ หมายความว่ารูปแบบแทนจำนวน $X = (x_{n-1}\ x_{n-2}\ x_{n-3}\ \ldots\ x_0)$ ใดๆ สามารถนิยาม $x_i$ ได้ 3 ค่าคือ -1 0 และ 1 เมื่อ $i = 0, \ldots, n - 1$ ระบบจำนวนมอดุลาร์ซ้ำซ้อนดังกล่าวสามารถนิยามทุกจำนวนเต็มตั้งแต่ 0 ถึง 36 ได้ดังตารางที่ 2.1

ตารางที่ 2.1 แสดงจำนวนเต็มที่นิยามในระบบ MNS(37, 4, 8, 2)

| 0 | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| 0 0 0 0 | 0 0 0 1 | 1 0 1 0 | -1 1 1 -1 | -1 1 1 0 | -1 1 1 1 |
| **6** | **7** | **8** | **9** | **10** | **11** |
| -1 0 0 0 | 0 0 1 -1 | 0 0 1 0 | 0 0 1 1 | 0 -1 0 0 | 0 -1 0 1 |
| **12** | **13** | **14** | **15** | **16** | **17** |
| 1 -1 1 0 | 1 -1 1 1 | 1 1 -1 1 | -1 -1 0 -1 | -1 -1 0 0 | -1 -1 0 1 |
| **18** | **19** | **20** | **21** | **22** | **23** |
| 0 1 -1 -1 | 0 1 -1 0 | 1 1 0 -1 | 1 1 0 0 | 1 1 0 1 | -1 -1 1 -1 |
| **24** | **25** | **26** | **27** | **28** | **29** |
| -1 -1 1 0 | -1 -1 1 1 | 0 1 0 -1 | 0 1 0 0 | 0 1 0 1 | 1 1 1 0 |
| **30** | **31** | **32** | **33** | **34** | **35** |
| 1 1 1 1 | 1 0 0 0 | 1 0 0 1 | -1 1 0 0 | -1 1 0 1 | 0 1 1 0 |
| **36** | | | | | |
| 0 1 1 1 | | | | | |

□

จากตารางที่ 2.1 แสดงค่าของจำนวนเต็ม $||X||$ ที่มีค่าตั้งแต่ 0 ถึง 36 (หรือ $P$-1) โดยส่วนล่างของตารางจะเป็นรูปแบบแทนจำนวนที่ใช้นิยามค่าของจำนวนเต็มตามช่องบน เช่น จำนวนเต็ม 25 สามารถแทนได้ด้วยรูปแบบแทนจำนวน (-1 -1 1 1) ซึ่งสามารถหาค่าเชิงตัวเลขของรูปแบบแทนจำนวนดังกล่าวได้ในสมการที่ 2.3 จะได้ดังนี้

$$\begin{aligned}((-1)8^3 + (-1)8^2 + (1)8^1 + (1)8^0) \bmod 37 &= ((-512) + (-64) + 8 + 1) \bmod 37 \\ &= (-567) \bmod 37 \\ &= (-567) - 37 \lfloor (-567) / 37 \rfloor \\ &= 25\end{aligned}$$

เมื่อพิจารณาความซ้ำซ้อนของรูปแบบแทนจำนวนจะได้ว่าจำนวนเต็ม 25 สามารถแทนได้ด้วยรูปแบบแทนจำนวนอีกรูปแบบหนึ่ง นั่นคือ (-1 1 -1 0) ซึ่งสามารถหาค่าเชิงตัวเลขได้ดังนี้

$$((-1)8^3 + (1)8^2 + (-1)8^1 + (0)8^0) \bmod 37 = ((-512) + 64 + (-8) + 0) \bmod 37$$
$$= (-456) \bmod 37$$
$$= (-456) - 37 \lfloor (-456) / 37 \rfloor$$
$$= 25$$

## 2.4 การดำเนินการเลขคณิตสำหรับระบบจำนวนมอดุลาร์แบบพหุนาม (arithmetic operations in the polynomial modular number system)

ปัจจุบันมีงานวิจัยมุ่งพัฒนาการดำเนินการเลขคณิตสำหรับระบบจำนวนมอดุลาร์ให้มีประสิทธิภาพทางด้านเวลา โดยได้นำเสนอการดำเนินการเลขคณิตสำหรับระบบจำนวนมอดุลาร์แบบพหุนาม [6-7] ดังนี้

### 2.4.1 การบวก (addition)

รูปแบบแทนจำนวน $A = (a_{n-1}\ a_{n-2}\ a_{n-3} \ldots a_0)$ และ $B = (b_{n-1}\ b_{n-2}\ b_{n-3} \ldots b_0)$ เป็นรูปแบบแทนจำนวนในระบบจำนวนมอดุลาร์ MNS($P$, $n$, $\beta$, $2^k$) (หรือ MNS($P$, $n$, $\beta$, $\rho$)) โดยที่ $|a_i| < 2^k$ และ $|b_i| < 2^k$ การบวกกันของ $A$ และ $B$ เป็นการคำนวณหารูปแบบแทนจำนวน $Z' = (z'_{n-1}\ z'_{n-2}\ z'_{n-3} \ldots z'_0)$ ที่สอดคล้องกับสมการที่ 2.4 ดังนี้

$$A + B = Z'$$
$$(||A|| + ||B||) \bmod P = ||Z'|| \bmod P \qquad (2.4)$$

เมื่อ $||A|| = \sum_{i=0}^{n-1} a_i \beta^i \bmod P$, $||B|| = \sum_{i=0}^{n-1} b_i \beta^i \bmod P$ และ $||Z'|| = \sum_{i=0}^{n-1} z'_i \beta^i \bmod P$ เป็นค่าเชิงตัวเลขของรูปแบบแทนจำนวน $A$ $B$ และ $Z'$ ตามลำดับ โดยที่ $|z'_i| < 2^k$

การบวกในระบบจำนวนมอดุลาร์แบบพหุนามสามารถคำนวณได้ดังรูปที่ 2.2

$$\begin{array}{lll} a_{n-1} \ldots\ a_3 \quad a_2 \quad a_1 \quad a_0 & ; & |a_i| < 2^k \\ + & & \\ b_{n-1} \ldots\ b_3 \quad b_2 \quad b_1 \quad b_0 & ; & |b_i| < 2^k \\ \hline z_{n-1} \ldots\ z_3 \quad z_2 \quad z_1 \quad z_0 & ; & |z_i| < 2^{k+1} \end{array}$$

รูปที่ 2.2 แสดงการคำนวณหารูปแบบแทนจำนวน $Z = A + B$

เมื่อ $i = 0, 1, \ldots, n\text{-}1$ เนื่องจาก $|z_i| < 2^{k+1}$ แสดงว่ารูปแบบแทนจำนวน $Z = (z_{n-1}\ z_{n-2}\ z_{n-3} \ldots z_0)$ ไม่เป็นรูปแบบแทนจำนวนในระบบจำนวนมอดุลาร์ MNS($P$, $n$, $\beta$, $2^k$) ดังนั้นจึงต้องทำการแปลงชุดตัวเลขเพื่อให้ได้รูปแบบแทนจำนวน $Z' = (z'_{n-1}\ z'_{n-2}\ z'_{n-3} \ldots z'_0)$ ที่สอดคล้องกับสมการที่ 2.4 ดังกล่าวข้างต้นและสมการที่ 2.5 ดังต่อไปนี้

$$||Z'|| \bmod P = ||Z|| \bmod P \tag{2.5}$$

โดยที่ $|z'_i| < 2^k$

### 2.4.2 การคูณ (multiplication)

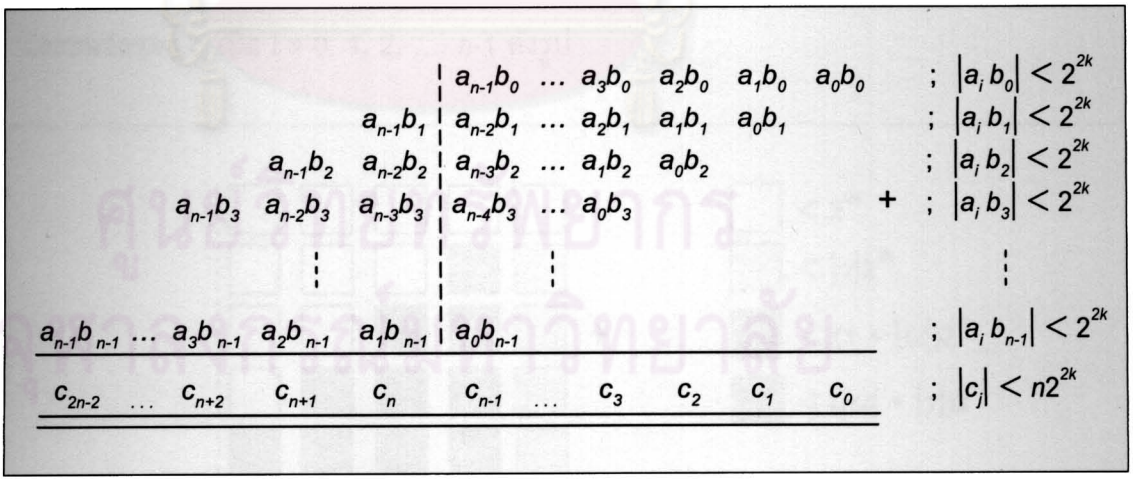
การคูณของ 2 รูปแบบแทนจำนวน $A$ และ $B$ เป็นการคำนวณหารูปแบบแทนจำนวน $C' = (c'_{n-1}\ c'_{n-2}\ c'_{n-3} \ldots c'_0)$ ที่สอดคล้องกับสมการที่ 2.6 ดังนี้

$$A \times B = C'$$

$$(||A|| \times ||B||) \bmod P = ||C'|| \bmod P \tag{2.6}$$

เมื่อ $||C'|| = \sum_{i=0}^{n-1} c'_i \beta^i \bmod P$ เป็นค่าเชิงตัวเลขของรูปแบบแทนจำนวน $C'$ โดยที่ $|c'_i| < 2^k$

การคูณในระบบจำนวนมอดุลาร์แบบพหุนามสามารถคำนวณได้ดังรูปที่ 2.3



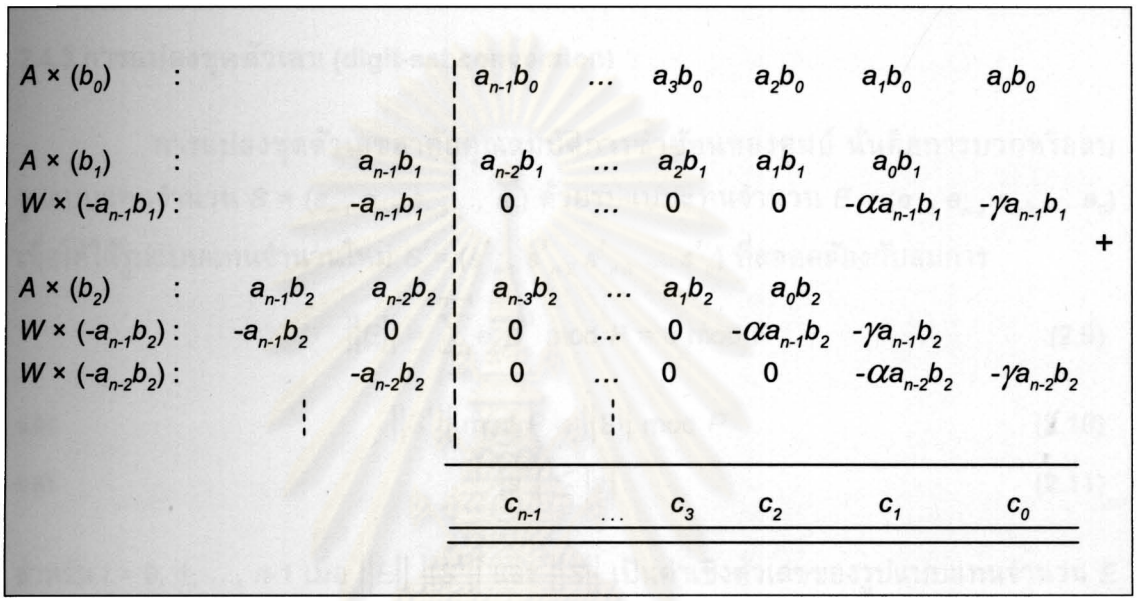

รูปที่ 2.3 แสดงการคำนวณหารูปแบบแทนจำนวน $C = A \times B$

เมื่อ $i = 0, 1, \ldots, n\text{-}1$ และ $j = 0, 1, \ldots, 2n\text{-}2$ เนื่องจากการคูณของ 2 รูปแบบแทนจำนวนทำให้จำนวนหลักของรูปแบบแทนจำนวน $C$ เพิ่มขึ้น แสดงว่ารูปแบบแทนจำนวน $C$ ไม่เป็นรูปแบบแทนจำนวนในระบบจำนวนมอดุลาร์ MNS($P$, $n$, $\beta$, $2^k$) ดังนั้นจึงต้องทำการลดจำนวนหลักลง

การลดจำนวนหลักของการคูณสำหรับระบบจำนวนมอดุลาร์แบบพหุนามจะใช้รูปแบบแทนจำนวน $W = (w_n\ w_{n-1}\ w_{n-2}\ w_{n-3}\ \ldots\ w_1\ w_0) = (1\ 0\ 0\ 0\ \ldots\ \alpha\ \gamma)$ ที่สอดคล้องกับสมการดังนี้

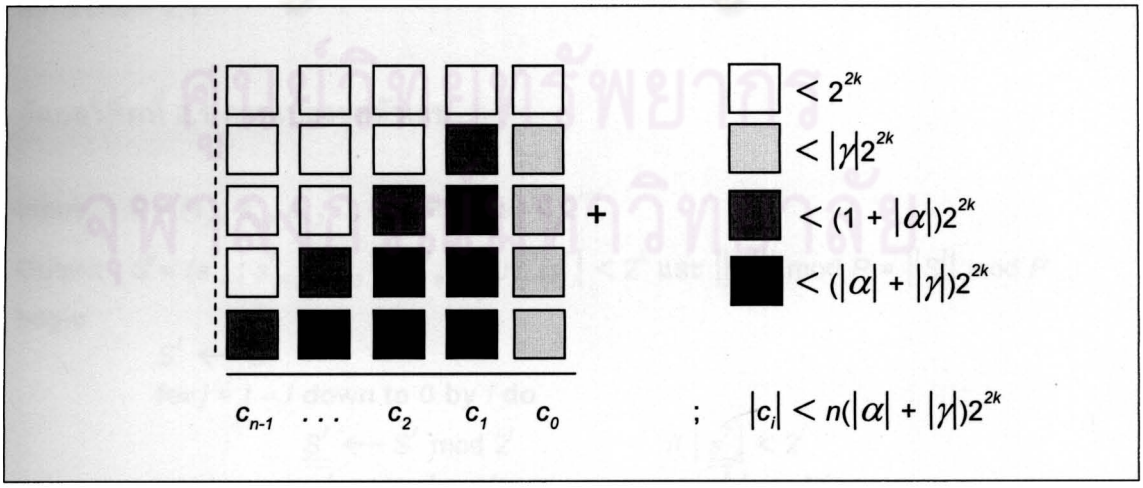
$$||W|| = \sum_{i=0}^{n} w_i \beta^i \bmod P = 0 \bmod P \qquad (2.7)$$

ซึ่งสามารถแสดงการลดจำนวนหลักได้ดังรูปต่อไปนี้




รูปที่ 2.4 แสดงการคำนวณการลดจำนวนหลักของรูปแบบแทนจำนวน $C$

และขนาดของ $c_i$ เมื่อ $i = 0, 1, 2, \ldots, n\text{-}1$ ดังรูป




รูปที่ 2.5 แสดงขนาดของ $c_i$

เนื่องจาก $|c_i| < n(|\alpha| + |\gamma|)2^{2k} = 2^{k+t}$ เมื่อ $t = \log_2 n(|\alpha| + |\gamma|) + k$ แสดงว่ารูปแบบแทนจำนวน $C = (c_{n-1}\ c_{n-2}\ c_{n-3}\ \ldots\ c_0)$ ไม่เป็นรูปแบบแทนจำนวนในระบบจำนวนมอดุลาร์ดังกล่าว ดังนั้นจึงต้องทำการแปลงชุดตัวเลขเพื่อให้ได้รูปแบบแทนจำนวน $C' = (c'_{n-1}\ c'_{n-2}\ c'_{n-3}\ \ldots\ c'_0)$ ที่สอดคล้องกับสมการที่ 2.6 ดังกล่าวข้างต้นและสมการ 2.8 ดังต่อไปนี้

$$||C'|| \bmod P = ||C|| \bmod P \tag{2.8}$$

โดยที่ $|c'_i| < 2^k$

### 2.4.3 การแปลงชุดตัวเลข (digit-set conversion)

การแปลงชุดตัวเลขอาศัยคุณสมบัติการซ้ำซ้อนของศูนย์ นั่นคือการบวกหรือลบรูปแบบแทนจำนวน $S = (s_{n-1}\ s_{n-2}\ s_{n-3}\ \ldots\ s_0)$ ด้วยรูปแบบแทนจำนวน $E = (e_{n-1}\ e_{n-2}\ e_{n-3}\ \ldots\ e_0)$ เพื่อให้ได้รูปแบบแทนจำนวนใหม่ $S' = (s'_{n-1}\ s'_{n-2}\ s'_{n-3}\ \ldots\ s'_0)$ ที่สอดคล้องกับสมการ

$$||E|| = \sum_{i=0}^{n-1} e_i \beta^i \bmod P = 0 \bmod P \tag{2.9}$$

และ

$$||S'|| \bmod P = ||S|| \bmod P \tag{2.10}$$

และ

$$|s'_i| \leq |s_i| \tag{2.11}$$

สำหรับ $i = 0, 1, \ldots, n-1$ เมื่อ $||E||$ $||S'||$ และ $||S||$ เป็นค่าเชิงตัวเลขของรูปแบบแทนจำนวน $E$ $S'$ และ $S$ ตามลำดับ

ปัจจุบันมีงานวิจัยที่พัฒนาการแปลงชุดตัวเลขภายใต้จำนวนรอบเท่ากับ $t$ เมื่อ $|s_i| < 2^{k+t}$ โดยที่รูปแบบแทนจำนวนที่เราต้องการคือ $S'$ เมื่อ $|s'_i| < 2^k$ ดังแสดงได้ตามอัลกอริทึมที่ 2.1

**อัลกอริทึมที่ 2.1** การแปลงชุดตัวเลข

**Input** : $S = (s_{n-1}\ s_{n-2}\ s_{n-3}\ \ldots\ s_0)$ โดยที่ $|s_i| < 2^{k+t}$

**Output** : $S' = (s'_{n-1}\ s'_{n-2}\ s'_{n-3}\ \ldots\ s'_0)$ โดยที่ $|s'_i| < 2^k$ และ $||S'|| \bmod P = ||S|| \bmod P$

**begin**

$S' \leftarrow S$

**for** $j = t - l$ **down to** 0 **by** $l$ **do**

$\underline{S'} \leftarrow S' \bmod 2^j$ // $|\underline{s'_i}| < 2^j$

$\overline{S'} \leftarrow (S' - \underline{S'}) / 2^j$ // $|\overline{s'_i}| < 2^{k+1}$

$\overline{S'} \leftarrow \overline{S'} - E$ // $|\overline{s'_i}| < 2^k$

$S' \leftarrow \underline{S'} + 2^j \overline{S'}$

**endfor**

**end**

จากอัลกอริทึม 2.1 การแปลงชุดตัวเลขของการดำเนินการเลขคณิตสำหรับระบบจำนวนมอดุลาร์แบบพหุนามจะต้องทำการคำนวณ $t$ รอบ ถ้า $\rho = 2^k$ และ $m = 2^t$ จะได้ว่าความซับซ้อนเชิงเวลาคือ $O(\log_2 m)$ เมื่อขนาดของดิจิตของรูปแบบแทนจำนวนก่อนการแปลงชุดตัวเลขคือ $m\rho$ และขนาดของดิจิตของรูปแบบแทนจำนวนหลังการแปลงชุดตัวเลขคือ $\rho$

## 2.5 ระบบจำนวนซ้ำซ้อน (redundant number system)

จำนวนเต็มใดๆ สามารถแทนในระบบจำนวนทั่วไปได้ด้วยรูปแบบแทนจำนวน $X = (x_{n-1}\ x_{n-2}\ x_{n-3}\ \ldots\ x_0)$ โดยค่าเชิงตัวเลขของรูปแบบแทนจำนวน $X$ มีค่าดังสมการที่ (2.12)

$$||X|| = \sum_{i=0}^{n-1} x_i \beta^i \qquad (2.12)$$

เมื่อ $\beta$ เป็นเลขฐาน และ $n$ เป็นจำนวนหลักของรูปแบบแทนจำนวน

ในระบบจำนวนดังกล่าว ถ้ามีรูปแบบแทนจำนวน $X_1$ และรูปแบบแทนจำนวน $X_2$ โดยที่ค่าเชิงตัวเลขของรูปแบบแทนจำนวน $X_1$ และ $X_2$ มีค่าเท่ากัน นั่นคือ $||X_1|| = ||X_2||$ แล้วจะเรียกระบบจำนวนนี้ว่า ระบบจำนวนซ้ำซ้อน (redundant number system) [11] ตัวอย่างเช่น รูปแบบแทนจำนวน $X_1$ = (0 1 0 1) และรูปแบบแทนจำนวน $X_2$ = (1 0 -1 -1) ในระบบจำนวนเลขฐาน 2 โดยที่รูปแบบแทนจำนวนมี 4 หลัก มีค่าเชิงตัวเลขของรูปแบบแทนจำนวน $X_1$ และ $X_2$ เท่ากัน คือ 5

# บทที่ 3

## การดำเนินการพื้นฐานเลขคณิต

การคำนวณพื้นฐานเลขคณิตของระบบจำนวนทั่วไปมีปัญหาด้านการคำนวณที่ล่าช้า จำนวนเต็มที่มีขนาดใหญ่จะเสียเวลามากในการคำนวณ งานวิจัยส่วนใหญ่มุ่งไปที่การจำกัดการทด ในขณะเดียวกันมีงานวิจัยส่วนหนึ่งที่มุ่งพัฒนาการขจัดการทด เช่น ระบบจำนวนส่วนตกค้าง ระบบจำนวนส่วนตกค้างเป็นระบบที่แบ่งจำนวนเต็มที่มีขนาดใหญ่ออกเป็นจำนวนเต็มย่อยๆ ที่อิสระต่อกัน ซึ่งจะเรียกจำนวนเต็มย่อยๆ แต่ละจำนวนว่า ระบบจำนวนมอดุลาร์ ในการคำนวณพื้นฐานเลขคณิตของระบบจำนวนส่วนตกค้างสามารถคำนวณจำนวนเต็มย่อยๆ แต่ละจำนวนไปพร้อมๆกันได้ ดังนั้นเวลาในการคำนวณพื้นฐานเลขคณิตของระบบจำนวนส่วนตกค้างจะขึ้นอยู่กับเวลาในการคำนวณพื้นฐานเลขคณิตของระบบจำนวนมอดุลาร์ การคำนวณพื้นฐานเลขคณิตของระบบจำนวนมอดุลาร์จะทำการคำนวณแบบไม่มีการทด โดยพิจารณาในรูปของปัญหาการแปลงชุดตัวเลข ซึ่งเดิมทีการแปลงชุดตัวเลขอาศัยคุณสมบัติความซ้ำซ้อนของศูนย์ นั่นคือจะอาศัยความซ้ำซ้อนของรูปแบบแทนจำนวนที่มีค่าเชิงตัวเลขเป็นศูนย์ในการแปลงชุดตัวเลข ในความเป็นจริงรูปแบบแทนจำนวนที่มีค่าเชิงตัวเลขเป็นศูนย์มีหลายรูปแบบและรูปแบบไหนที่เหมาะสมที่จะนำมาใช้ในการแปลงชุดตัวเลข ในงานวิจัยของบาจาร์ดได้เสนอการดำเนินการพื้นฐานเลขคณิตสำหรับระบบจำนวนมอดุลาร์แบบพหุนาม ซึ่งการแปลงชุดตัวเลขของการดำเนินการพื้นฐานเลขคณิตสำหรับระบบจำนวนมอดุลาร์แบบพหุนามได้กำหนดขอบเขตในการหารูปแบบแทนจำนวนที่มีค่าเชิงตัวเลขเป็นศูนย์ โดยรับประกันว่าในการแปลงชุดตัวเลขแต่ละครั้งจะทำให้ขนาดของชุดตัวเลขลดลงครึ่งหนึ่ง ดังนั้นเวลาในการแปลงชุดตัวเลขจะขึ้นอยู่กับขนาดของชุดตัวเลขก่อนการแปลงชุดตัวเลข ทั้งนี้ในกรณีที่มีการแปลงชุดตัวเลขหลายครั้ง แต่ละครั้งก็ต้องทำการคำนวณเพื่อหารูปแบบแทนจำนวนใหม่ทุกครั้ง

จากที่กล่าวมาข้างต้น เราจึงสนใจปัญหาการแปลงชุดตัวเลข โดยพิจารณาลักษณะของรูปแบบแทนจำนวนที่มีค่าเชิงตัวเลขเป็นศูนย์ที่ทำให้จำนวนครั้งในการแปลงชุดตัวเลขเป็นค่าคงที่ ซึ่งเวลาในการการแปลงชุดตัวเลขไม่อยู่ขึ้นกับขนาดของชุดตัวเลขก่อนการแปลงชุดตัวเลข ดังนั้นรูปแบบแทนจำนวนดังกล่าวจะต้องมีค่าที่สอดคล้องกับรูปแบบแทนจำนวนก่อนการแปลงชุดตัวเลข การคำนวณหารูปแบบแทนจำนวนที่มีค่าเชิงตัวเลขเป็นศูนย์สามารถคำนวณได้จากผลรวมของรูปแบบแทนจำนวน $U$ และรูปแบบแทนจำนวน $V$ เพื่อลดเวลาในการแปลงชุดตัวเลขดังนั้นจึงกำหนดให้รูปแบบแทนจำนวน $U$ มีค่าที่สอดคล้องกับรูปแบบแทนจำนวนก่อนการแปลงชุดตัวเลข และเนื่องจากต้องการรูปแบบแทนจำนวนที่มีค่าเชิงตัวเลขเป็นศูนย์ดังนั้นรูปแบบแทนจำนวน $V$ ต้องมีค่าเชิงตัวเลขเท่ากับผลลบของค่าเชิงตัวเลขของรูปแบบแทนจำนวน $U$ แสดงว่าในการแปลงชุดตัวเลขจะคำนวณหารูปแบบแทนจำนวน $V$ แทนรูปแบบ

แทนจำนวนที่มีค่าเชิงตัวเลขเป็นศูนย์ ในงานวิจัยนี้เสนอการดำเนินการพื้นฐานเลขคณิตสำหรับระบบจำนวนมอดุลาร์อีกรูปแบบหนึ่ง โดยเรียกว่า การดำเนินการพื้นฐานเลขคณิตสำหรับระบบจำนวนมอดุลาร์ซ้ำซ้อน (fundamental arithmetic operation in redundant modular number system) ได้แก่ การบวก การลบ และการคูณ และเสนออัลกอริทึมการแปลงชุดตัวเลขพร้อมทั้งบทพิสูจน์

## 3.1 การดำเนินการพื้นฐานเลขคณิตสำหรับระบบจำนวนมอดุลาร์ซ้ำซ้อน (fundamental arithmetic operation in redundant modular number system)

ในงานวิจัยเกี่ยวกับระบบแทนจำนวน สิ่งสำคัญที่ต้องพิจารณาคือเรื่องของการดำเนินการพื้นฐานเลขคณิตของระบบแทนจำนวน ซึ่งในงานวิจัยนี้ได้นำเสนอการดำเนินการพื้นฐานเลขคณิตสำหรับระบบจำนวนมอดุลาร์ซ้ำซ้อน ได้แก่ การบวก การลบ และการคูณ

### 3.1.1 การบวก (addition)

รูปแบบแทนจำนวน $A = (a_{n-1}\ a_{n-2}\ a_{n-3}\ \dots\ a_0)$ และ $B = (b_{n-1}\ b_{n-2}\ b_{n-3}\ \dots\ b_0)$ เป็นรูปแบบแทนจำนวนในระบบจำนวนมอดุลาร์ซ้ำซ้อน MNS($P$, $n$, $\beta$, $\rho$) โดยที่ $|a_i| < \rho$ และ $|b_i| < \rho$ การบวกกันของ $A$ และ $B$ เป็นการคำนวณหารูปแบบแทนจำนวน $Z' = (z'_{n-1}\ z'_{n-2}\ z'_{n-3}\ \dots\ z'_0)$ ที่สอดคล้องกับสมการที่ 3.1 ดังนี้

$$A + B = Z'$$
$$(||A|| + ||B||) \bmod P = ||Z'|| \bmod P \qquad (3.1)$$

เมื่อ $||A|| = \sum_{i=0}^{n-1} a_i \beta^i \bmod P$, $||B|| = \sum_{i=0}^{n-1} b_i \beta^i \bmod P$ และ $||Z'|| = \sum_{i=0}^{n-1} z'_i \beta^i \bmod P$ เป็นค่าเชิงตัวเลขของรูปแบบแทนจำนวน $A$ $B$ และ $Z'$ ตามลำดับ โดยที่ $|z'_i| < \rho$

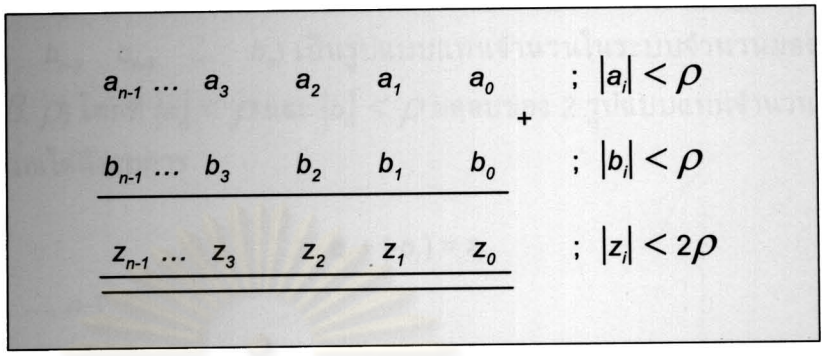
การบวกในระบบจำนวนมอดุลาร์ซ้ำซ้อนจะคำนวณในแต่ละหลักพร้อมกัน เพื่อความเร็วในการคำนวณ ซึ่งจะมีบทนิยามดังนี้

**บทนิยามที่ 3.1** รูปแบบแทนจำนวน $A = (a_{n-1}\ a_{n-2}\ a_{n-3}\ \dots\ a_0)$ และรูปแบบแทนจำนวน $B = (b_{n-1}\ b_{n-2}\ b_{n-3}\ \dots\ b_0)$ เป็นรูปแบบแทนจำนวนในระบบจำนวนมอดุลาร์ซ้ำซ้อน MNS($P$, $n$, $\beta$, $\rho$) โดยที่ $|a_i| < \rho$ และ $|b_i| < \rho$ ผลบวกของ 2 รูปแบบแทนจำนวน ($A + B = Z$) สามารถคำนวณได้ดังสมการ

$$a_i + b_i = z_i \qquad (3.2)$$

เมื่อ $i$ = **0**, **1**, ..., $n$-**1**

เพื่อความเข้าใจให้พิจารณารูปที่ 3.1




รูปที่ 3.1 แสดงการคำนวณการบวกของรูปแบบแทนจำนวน $Z = A + B$

**เนื่องจาก** $|z_i| < 2\rho$ แสดงว่ารูปแบบแทนจำนวน $Z = (z_{n-1}\ z_{n-2}\ z_{n-3}\ \ldots\ z_0)$ ไม่เป็นรูปแบบแทนจำนวนในระบบจำนวนมอดุลาร์ซ้ำซ้อน MNS($P$, $n$, $\beta$, $\rho$) ดังนั้นจึงต้องทำการแปลงชุดตัวเลขเพื่อให้ได้รูปแบบแทนจำนวน $Z' = (z'_{n-1}\ z'_{n-2}\ z'_{n-3}\ \ldots\ z'_0)$ ที่สอดคล้องกับสมการที่ 3.1 ดังกล่าวข้างต้นและสมการที่ 3.3 ดังต่อไปนี้

$$||Z'|| \bmod P = ||Z|| \bmod P \tag{3.3}$$

โดยที่ $|z'_i| < \rho$

### 3.1.2 การลบ (subtraction)

รูปแบบแทนจำนวน $A = (a_{n-1}\ a_{n-2}\ a_{n-3}\ \ldots\ a_0)$ และ $B = (b_{n-1}\ b_{n-2}\ b_{n-3}\ \ldots\ b_0)$ เป็นรูปแบบแทนจำนวนในระบบจำนวนมอดุลาร์ซ้ำซ้อน MNS($P$, $n$, $\beta$, $\rho$) โดยที่ $|a_i| < \rho$ และ $|b_i| < \rho$ การลบกันของ $A$ และ $B$ เป็นการคำนวณหารูปแบบแทนจำนวน $Z' = (z'_{n-1}\ z'_{n-2}\ z'_{n-3}\ \ldots\ z'_0)$ ที่สอดคล้องกับสมการที่ 3.4 ดังนี้

$$A - B = Z'$$

$$(||A|| - ||B||) \bmod P = ||Z'|| \bmod P \tag{3.4}$$

เมื่อ $||A|| = \sum_{i=0}^{n-1} a_i \beta^i \bmod P$, $||B|| = \sum_{i=0}^{n-1} b_i \beta^i \bmod P$ และ $||Z'|| = \sum_{i=0}^{n-1} z'_i \beta^i \bmod P$ เป็นค่าเชิงตัวเลขของรูปแบบแทนจำนวน $A$ $B$ และ $Z$ ตามลำดับ โดยที่ $|z'_i| < \rho$

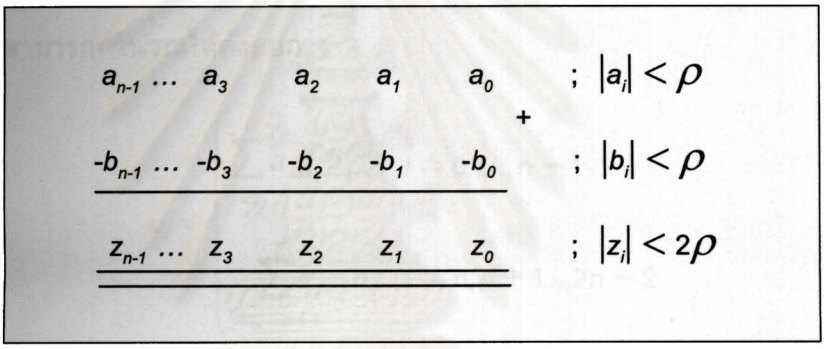
การลบสามารถคำนวณได้เช่นเดียวกับการบวก โดยการกลับเครื่องหมายของรูปแบบแทนจำนวน นั่นคือมีบทนิยามดังนี้

**บทนิยามที่ 3.2** รูปแบบแทนจำนวน $A = (a_{n-1}\ a_{n-2}\ a_{n-3}\ \dots\ a_0)$ และรูปแบบแทนจำนวน $B = (b_{n-1}\ b_{n-2}\ b_{n-3}\ \dots\ b_0)$ เป็นรูปแบบแทนจำนวนในระบบจำนวนมอดุลาร์ซ้ำซ้อน MNS($P$, $n$, $\beta$, $\rho$) โดยที่ $|a_i| < \rho$ และ $|b_i| < \rho$ ผลลบของ 2 รูปแบบแทนจำนวน ($A - B = Z$) สามารถคำนวณได้ดังสมการ

$$a_i + (-b_i) = z_i \tag{3.5}$$

เมื่อ $i = 0, 1, \dots, n-1$

เพื่อความเข้าใจให้พิจารณารูปที่ 3.2




รูปที่ 3.2 แสดงการคำนวณการลบของรูปแบบแทนจำนวน $Z = A - B$

เนื่องจาก $|z_i| < 2\rho$ แสดงว่ารูปแบบแทนจำนวน $Z = (z_{n-1}\ z_{n-2}\ z_{n-3}\ \dots\ z_0)$ ไม่เป็นรูปแบบแทนจำนวนในระบบจำนวนมอดุลาร์ซ้ำซ้อน MNS($P$, $n$, $\beta$, $\rho$) ดังนั้นจึงต้องทำการแปลงชุดตัวเลขเพื่อให้ได้รูปแบบแทนจำนวน $Z' = (z'_{n-1}\ z'_{n-2}\ z'_{n-3}\ \dots\ z'_0)$ ที่สอดคล้องกับสมการที่ 3.4 ดังกล่าวข้างต้นและสมการที่ 3.6 ดังต่อไปนี้

$$||Z'|| \bmod P = ||Z|| \bmod P \tag{3.6}$$

โดยที่ $|z'_i| < \rho$

### 3.1.3 การคูณ (multiplication)

การคูณของ 2 รูปแบบแทนจำนวน $A = (a_{n-1}\ a_{n-2}\ a_{n-3}\ \dots\ a_0)$ และรูปแบบแทนจำนวน $B = (b_{n-1}\ b_{n-2}\ b_{n-3}\ \dots\ b_0)$ เป็นการคำนวณหา $C' = (c'_{n-1}\ c'_{n-2}\ c'_{n-3}\ \dots\ c'_0)$ ที่สอดคล้องกับสมการที่ 3.7 ดังนี้

$$A \times B = C'$$

$$(||A|| \times ||B||) \bmod P = ||C'|| \bmod P \tag{3.7}$$

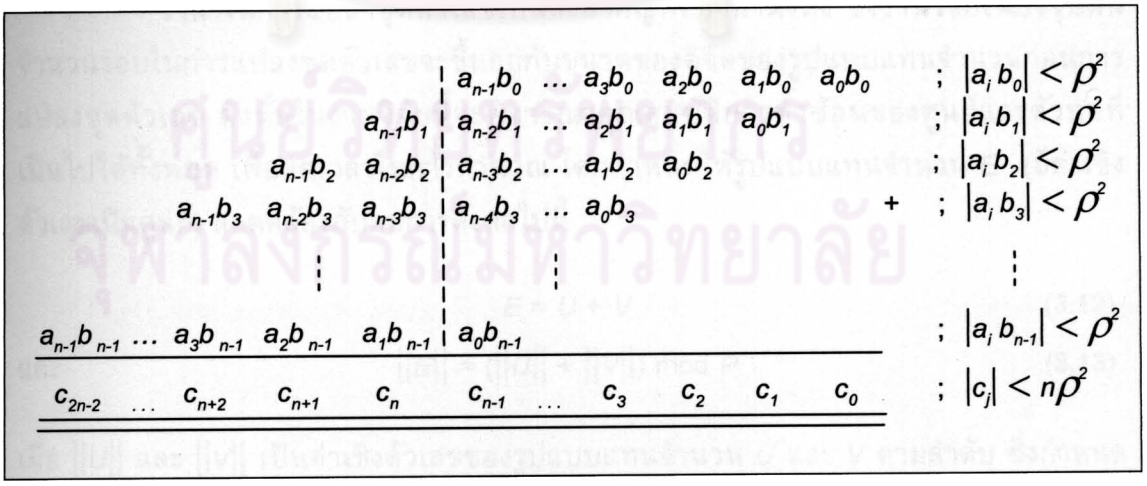
เมื่อ $||A|| = \sum_{i=0}^{n-1} a_i \beta^i \bmod P$, $||B|| = \sum_{i=0}^{n-1} b_i \beta^i \bmod P$ และ $||C'|| = \sum_{i=0}^{n-1} c'_i \beta^i \bmod P$ เป็นค่าเชิงตัวเลขของรูปแบบแทนจำนวน $A$ $B$ และ $C'$ โดยที่ $|c'_i| < \rho$

การคูณของการดำเนินการพื้นฐานเลขคณิตสำหรับระบบจำนวนมอดุลาร์ซ้ำซ้อนสามารถคำนวณได้ตามบทนิยามดังนี้

**บทนิยามที่ 3.3** รูปแบบแทนจำนวน $A = (a_{n-1}\ a_{n-2}\ a_{n-3}\ \dots\ a_0)$ และรูปแบบแทนจำนวน $B = (b_{n-1}\ b_{n-2}\ b_{n-3}\ \dots\ b_0)$ เป็นรูปแบบแทนจำนวนในระบบจำนวนมอดุลาร์ซ้ำซ้อน MNS($P$, $n$, $\beta$, $\rho$) โดยที่ $|a_i| < \rho$ และ $|b_i| < \rho$ ผลคูณของ 2 รูปแบบแทนจำนวน ($A \times B = C$) สามารถคำนวณได้ดังสมการ

$$c_i = \begin{cases} \sum_{j=0}^{i} a_{i-j} b_j & ; i = 0,1,..,n-1 \\ \sum_{j=i-n+1}^{n-1} a_{i-j} b_j & ; i = n, n+1,..,2n-2 \end{cases} \tag{3.8}$$

เพื่อความเข้าใจให้พิจารณารูปที่ 3.3




รูปที่ 3.3 แสดงการคำนวณการคูณของรูปแบบแทนจำนวน $C = A \times B$

เนื่องจาก $|c_i| < n\rho^2 = m\rho$ เมื่อ $m = n\rho$ และรูปแบบแทนจำนวน $C = (c_{2n-2}\ c_{2n-3}\ c_{2n-4}\ \ldots\ c_0)$ มีจำนวนหลักเพิ่มขึ้นเป็น $2n$-1 หลัก แสดงว่ารูปแบบแทนจำนวน $C$ ไม่เป็นรูปแบบแทนจำนวนในระบบจำนวนมอดุลาร์ซ้ำซ้อน MNS($P$, $n$, $\beta$, $\rho$) ดังนั้นจึงต้องทำการแปลงชุดตัวเลขเพื่อให้ได้รูปแบบแทนจำนวน $C' = (c'_{n-1}\ c'_{n-2}\ c'_{n-3}\ \ldots\ c'_0)$ (รูปแบบแทนจำนวน $C'$ มี $n$ หลัก) ที่สอดคล้องกับสมการที่ 3.7 ดังกล่าวข้างต้นและสมการที่ 3.9 ดังต่อไปนี้

$$||C'|| \bmod P = ||C|| \bmod P \tag{3.9}$$

โดยที่ $|c'_i| < \rho$

### 3.1.4 การแปลงชุดตัวเลข (digit-set conversion)

การแปลงชุดตัวเลขอาศัยคุณสมบัติการซ้ำซ้อนของศูนย์ นั่นคือการลบรูปแบบแทนจำนวน $S$ ด้วยรูปแบบแทนจำนวน $E$ เพื่อให้ได้รูปแบบแทนจำนวนใหม่ $S' = S - E$ ที่สอดคล้องกับสมการ

$$||E|| = 0 \bmod P \tag{3.10}$$

และ
$$||S'|| \bmod P = ||S|| \bmod P \tag{3.11}$$

เมื่อ $||E||$ $||S'||$ และ $||S||$ เป็นค่าเชิงตัวเลขของรูปแบบแทนจำนวน $E$ $S'$ และ $S$ ตามลำดับ ซึ่งรูปแบบแทนจำนวน $S'$ ต้องเป็นรูปแบบแทนจำนวนในระบบจำนวนมอดุลาร์ซ้ำซ้อน MNS($P$, $n$, $\beta$, $\rho$) นั่นคือ $S' = (s'_{n-1}\ s'_{n-2}\ s'_{n-3}\ \ldots\ s'_0)$ โดยที่ $|s'_i| < \rho$

เวลาในการแปลงชุดตัวเลขเป็นสิ่งสำคัญที่ต้องคำนึงถึง ซึ่งงานวิจัยในปัจจุบันนี้จำนวนรอบในการแปลงชุดตัวเลขจะขึ้นอยู่กับขนาดของดิจิตของรูปแบบแทนจำนวนก่อนการแปลงชุดตัวเลข ดังนั้นในงานวิจัยนี้จะพิจารณาคุณสมบัติการซ้ำซ้อนของศูนย์และตัวทดที่เป็นไปได้ทั้งหมด เพื่อลดเวลาในการคำนวณ โดยกำหนดให้รูปแบบแทนจำนวน $E$ (มีค่าเชิงตัวเลขเป็นศูนย์) สอดคล้องกับสมการดังต่อไปนี้

$$E = U + V \tag{3.12}$$

และ
$$||E|| = (||U|| + ||V||) \bmod P \tag{3.13}$$

เมื่อ $||U||$ และ $||V||$ เป็นค่าเชิงตัวเลขของรูปแบบแทนจำนวน $U$ และ $V$ ตามลำดับ ซึ่งกำหนดรูปแบบแทนจำนวน $U$ ให้สอดคล้องกับรูปแบบแทนจำนวนก่อนการแปลงชุดตัวเลข แต่ไม่มีการกำหนดรูปแบบแทนจำนวน $V$ ดังนั้นการแปลงชุดตัวเลขจะทำการหารูปแบบแทนจำนวน $V$ แทนรูปแบบแทนจำนวน $E$ การหารูปแบบแทนจำนวน $V$ จะต้องพิสูจน์ให้ได้ก่อนว่ามีรูปแบบแทนจำนวน $V$ ที่มีค่าเชิงตัวเลขอยู่ระหว่าง 0 ถึง $P - 1$ ซึ่งมีทฤษฎีบทต่อไปนี้

**ทฤษฎีบทที่ 3.1** ในระบบจำนวนมอดุลาร์ซ้ำซ้อน MNS($P$, $n$, $\beta$, $\rho$) จำนวนเต็มใดๆ ที่มีค่าตั้งแต่ **0** ถึง $P - 1$ จะสามารถหารูปแบบแทนจำนวน $Y = (y_{n-1}\ y_{n-2}\ y_{n-3}\ \dots\ y_0)$ ได้ ซึ่งมีค่าเชิงตัวเลขคือ $||Y|| = \sum_{i=0}^{n-1} y_i \beta^i \bmod P$ โดยที่ $0 \le y_i \le 2\rho - 2$

**พิสูจน์** เนื่องจากในระบบจำนวนมอดุลาร์ซ้ำซ้อน MNS($P$, $n$, $\beta$, $\rho$) จำนวนเต็มใดๆ ที่มีค่าตั้งแต่ **0** ถึง $P - 1$ สามารถหารูปแบบแทนจำนวน $X = (x_{n-1}\ x_{n-2}\ x_{n-3}\ \dots\ x_0)$ โดยค่าเชิงตัวเลขมีค่าเป็น

$$||X|| = \sum_{i=0}^{n-1} x_i \beta^i \bmod P$$

โดยที่ $|x_i| < \rho$ หรือ $-\rho + 1 \le y_i \le \rho - 1$ ได้

และกำหนดค่าคงที่ $$||C|| = \sum_{i=0}^{n-1} (\rho - 1)\beta^i$$

จะเห็นว่า $||X|| + ||C||$ เมื่อ $0 \le ||X|| \le P - 1$ เป็นจำนวนเต็มที่เรียงกัน $P$ จำนวน

ดังนั้น $(||X|| + ||C||) \bmod P$ จะเป็นจำนวนเต็มที่แตกต่างกันทั้งหมด $P$ จำนวน ซึ่งมีค่าตั้งแต่ 0 ถึง $P - 1$

กำหนดให้ $$||Y|| = (||X|| + ||C||) \bmod P$$

จะได้ว่า $$(||X|| + ||C||) \bmod P = \left(\sum_{i=0}^{n-1} x_i \beta^i + \sum_{i=0}^{n-1} (\rho - 1)\beta^i\right) \bmod P$$

$$= \left(\sum_{i=0}^{n-1} (x_i + (\rho - 1))\beta^i\right) \bmod P$$

ถ้ากำหนดให้ $y_i = x_i + (\rho - 1)$

จะได้ $$(||X|| + ||C||) \bmod P = \sum_{i=0}^{n-1} y_i \beta^i \bmod P$$

จาก $$-\rho + 1 \le x_i \le \rho - 1$$

จะได้ $$-\rho + 1 + (\rho - 1) \le x_i + (\rho - 1) \le \rho - 1 + (\rho - 1)$$

$$0 \le y_i \le 2\rho - 2$$ ■

จากทฤษฎีบทดังกล่าวจะได้ว่าสามารถหารูปแบบแทนจำนวน $V$ ที่มีค่าเชิงตัวเลขอยู่ระหว่าง 0 ถึง $P - 1$ ได้ โดยที่รูปแบบแทนจำนวน $V = (v_{n-1}\ v_{n-2}\ v_{n-3}\ \dots\ v_0)$ มีค่าเชิงตัวเลขคือ $||v|| = \sum_{i=0}^{n-1} v_i \beta^i \bmod P$ เมื่อ $0 \le v_i \le 2\rho - 2$ ซึ่งการหารูปแบบแทนจำนวน $V$ สามารถหาได้ดังอัลกอริทึมที่ 3.1 ต่อไปนี้

**อัลกอริทึมที่ 3.1** Transform($U$)

**Input** : $U = (u_{2n-2}\ u_{2n-3}\ u_{2n-4}\ \ldots\ u_0)$

**Output** : $V = (v_{n-1}\ v_{n-2}\ v_{n-3}\ \ldots\ v_0)$ โดยที่ $0 \le v_i \le 2\rho - 2$

**Begin**

$\|U\| \leftarrow \sum_{i=0}^{2n-2} u_i \beta^i \bmod P$

$\|V\| \leftarrow (-\|U\|) \bmod P$

$y \leftarrow \beta - (P \bmod \beta)$

$x \leftarrow (yP + 1) / \beta$

$q \leftarrow \lfloor (\sum_{i=0}^{n-1} (2\rho - 2)\beta^i - \|V\|) / P \rfloor$

$j \leftarrow 0$

**do**

$i \leftarrow 0$

**do**

**if** $i = 0$ **then**

$v_0 \leftarrow j$

$h \leftarrow y(\|V\| - v_0) \bmod \beta$

$r \leftarrow \lfloor (\|V\| + hP) / \beta \rfloor$

**if** $r \bmod \beta > 2\rho - 2$ **then**

$h \leftarrow h + \beta\{(yr \bmod \beta) \bmod y\}$

$r \leftarrow r + P\{(yr \bmod \beta) \bmod y\}$

**endif**

**else**

$v_i \leftarrow r \bmod \beta$

$r \leftarrow (r - v_i) / \beta$

**if** $r \bmod \beta > 2\rho - 2$ **then**

$f \leftarrow q$

**for** $l = 0$ **to** $2\rho - 2$ **do**

**if** $v_i - l \neq 0$ **and** $((v_i - l)x + r) \bmod \beta \le 2\rho - 2$ **then**

$f \leftarrow \min\{f, ((v_i - l) \bmod \beta) y\}$

**endfor**

$h \leftarrow h + \beta^i f$

$r \leftarrow \lfloor (\|V\| + hP) / \beta^{i+1} \rfloor$

**endif**

**endif**

$i \leftarrow i + 1$

**while** $i \le n - 1$ **and** $h \le q$ // do...while จะวนซ้ำไม่เกิน $n$-1 รอบ

$j \leftarrow j + 1$

**while** $j \le 2\rho - 2$ **and** $h > q$ // do...while จะวนซ้ำไม่เกิน $2\rho$ - 2 รอบ

$g \leftarrow \|V\| + hP$

**for** $i = 0$ **to** $n$-1 **do**

$v_i \leftarrow g \bmod \beta$

$g \leftarrow (g - v_i) / \beta$

**endfor**

**End**

**พิสูจน์** เนื่องจากต้องการรูปแบบแทนจำนวน $V = (v_{n-1}\ v_{n-2}\ v_{n-3}\ \ldots\ v_0)$ โดยที่ $0 \le v_i \le 2\rho - 2$ และสอดคล้องกับสมการที่ 3.10 3.12 และ 3.13 ดังนั้นจึงกำหนดให้

$$||V|| = (-||U||) \bmod P$$

และ
$$q = \lfloor (\sum_{i=0}^{n-1} (2\rho - 2)\beta^i - ||V||) / P \rfloor$$

แสดงว่า $||V|| = ||V|| \bmod P = ||V|| + hP$ โดยที่ $h \le q$ ในอัลกอริทึมนี้จะแบ่งการทำงานออกเป็น 2 ส่วน คือ การคำนวณหา $h$ กับการคำนวณหารูปแบบแทนจำนวน $V$

ส่วนของการคำนวณหา $h$ จะทำการคำนวณแบบวนซ้ำตามจำนวนหลักของรูปแบบแทนจำนวน นั่นคือจะคำนวณในหลักที่ 0 ถึงหลักที่ $n$-1 เมื่อ $n$ เป็นจำนวนหลักของรูปแบบแทนจำนวน โดยจะแบ่งเป็น 2 กรณี

กรณีที่ 1 คำนวณในหลักที่ 0 จะต้องกำหนดค่าเริ่มต้น $v_0 = j$ เมื่อ $j = 0, 1, \ldots, 2\rho - 2$ เพื่อคำนวณหา $h$ ที่สอดคล้องกับ $v_0$ โดย

$$h = y(||V|| - v_0) \bmod \beta$$

และจะได้
$$r = \lfloor (||V|| + hP) / \beta \rfloor$$

จากนั้นนำ $r$ ไปตรวจสอบ ถ้า $r \bmod \beta > 2\rho - 2$ จะต้องคำนวณค่า $h$ ใหม่ โดยสมการนี้

$$h_{new} = h + \beta((yr \bmod \beta) \bmod y)$$

และจะได้ $r$ ใหม่ โดย
$$r_{new} = r + P((yr \bmod \beta) \bmod y)$$

แต่ถ้า $r \bmod \beta \le 2\rho - 2$ ไม่ต้องคำนวณค่า $h$ ใหม่

กรณีที่ 2 คำนวณในหลักที่ไม่ใช่หลักที่ 0 โดย

$$v_i = r \bmod \beta$$

และจะได้
$$r_{new} = (r - v_i) / \beta$$

จากนั้นตรวจสอบ $r$ เช่นเดิม คือถ้า $r \bmod \beta > 2\rho - 2$ จะต้องคำนวณค่า $h$ ใหม่ ซึ่งจะคำนวณได้โดยสมการนี้

$$f_{new} = \min\{f, ((v_i - l) \bmod \beta) y\}$$

และ
$$h_{new} = h + \beta^i f$$

เมื่อ $l = 0, 1, \ldots, 2\rho - 2$

และจะได้ $r$ ดังสมการนี้ $\quad r = \lfloor (||V|| + hP) / \beta^{i+1} \rfloor$

โดยจะเลือก $f$ ที่น้อยที่สุด เพราะต้องการ $h \leq q$ แต่ถ้า $h > q$ จะกำหนดค่าเริ่มต้น $v_0 = j$ ในหลักที่ 0 ใหม่แล้วคำนวณหลักถัดไปเช่นเดิมจนกว่าจะคำนวณครบทุกหลักและ $h \leq q$

**เนื่อง**จากทฤษฎีบทที่ 3.1 ดังนั้นจะได้ว่ามี $h$ ที่ทำให้สามารถหารูปแบบแทนจำนวน $V = (v_{n-1}\ v_{n-2}\ v_{n-3} \ldots v_0)$ เมื่อ $0 \leq v_i \leq 2\rho - 2$ ได้ โดยที่มีค่าเชิงตัวเลขคือ

$$||V|| = (||V|| + hP) \bmod P$$

ส่วนของการคำนวณหารูปแบบแทนจำนวน $V$ จะทำการคำนวณเหมือนกับการคำนวณหารูปแบบแทนจำนวนในระบบเลขฐาน $\beta$ นั่นคือจะทำการคำนวณแบบวนซ้ำในแต่ละหลักโดยจะเริ่มจากหลักที่ 0 ถึงหลักที่ $n$-1

กำหนดให้ $\quad g = ||V|| + hP$

จะได้ว่า $\quad v_i = g \bmod \beta$ และ $g = (g - v_i) / \beta$

เมื่อ $i = 0, 1, \ldots, 2\rho - 2$ ■

**ตัวอย่างที่ 3.1** *ระบบจำนวนมอดุลาร์ซ้ำซ้อน MNS(P, n, $\beta$, $\rho$) = MNS(37, 4, 8, 2) หารูปแบบแทนจำนวน $V = (v_{n-1}\ v_{n-2}\ v_{n-3} \ldots v_0)$ โดยที่ $0 \leq v_i \leq 2\rho - 2$ และสอดคล้องกับสมการดังนี้*

$$||V|| = (-||U||) \bmod P$$

*เมื่อ $||V||$ และ $||U||$ เป็นค่าเชิงตัวเลขของรูปแบบแทนจำนวน $V$ และ $U = (1\ -1\ -2\ -3)$ ตามลำดับ*

**วิธีทำ** สามารถหารูปแบบแทนจำนวน $V = (v_{n-1}\ v_{n-2}\ v_{n-3} \ldots v_0)$ โดยที่ $0 \leq v_i \leq 2\rho - 2$ และสอดคล้องสมการ $||V|| = (-||U||) \bmod P$ ได้โดยอัลกอริทึม 3.1 ดังนี้

$||U|| = ((1)8^3 + (-1)8^2 + (-2)8 + (-3)) \bmod 37 = 22$

$||V|| = (-22) \bmod 37 = 15$

$y = 8 - (37 \bmod 8) = 3$

$x = (3(37)+1) / 8 = 14$

$q = ((2)8^3 + (2)8^2 + (2)8 + (2)) \bmod 37 = 31$

ส่วนของการคำนวณหา $h$ โดยที่ $h \leq q$ สามารถคำนวณได้ดังนี้

ที่ $j = 0$ และ $i = 0$

$v_0 = 0$

$h = 3(15 - 0) \bmod 8 = 5$

$r = \lfloor (15 + 5(37)) / 8 \rfloor = 25$

ตรวจสอบ $r$ โดย $r \bmod \beta = 25 \bmod 8 = 1 \leq 2 = 2\rho - 2$ ดังนั้นไม่ต้องเพิ่มค่า $h$

ที่ $j = 0$ และ $i = 1$

$v_1 = 25 \bmod 8 = 1$

$r = (25 - 1) / 8 = 3$

ตรวจสอบ $r$ โดย $r \bmod \beta = 3 \bmod 8 = 3 > 2$ ดังนั้นต้องเพิ่มค่า $h$ ดังนี้

$f = 31$

ที่ $l = 0$ จะได้ $v_i - l = 1 - 0 = 1 \neq 0$

และ $((v_i - l)x + r) \bmod \beta = ((1 - 0)14 + 3) \bmod 8 = 1 \leq 2 = 2\rho - 2$

ดังนั้น $f = \min\{31, 3\} = 3$

ที่ $l = 1$ จะได้ $v_i - l = 1 - 1 = 1 = 0$

ที่ $l = 2$ จะได้ $v_i - l = 1 - 2 \neq 0$

และ $((v_i - l)x + r) \bmod \beta = ((1 - 2)14 + 3) \bmod 8 = 5 > 2$

$h = 5 + (8)3 = 29$

$r = \lfloor (15 + 29(37)) / 8^2 \rfloor = 17$

ที่ $j = 0$ และ $i = 2$

$v_1 = 17 \bmod 8 = 1$

$r = (17 - 1) / 8 = 2$

ตรวจสอบ $r$ โดย $r \bmod \beta = 2 \bmod 8 = 2 \leq 2$ ดังนั้นไม่ต้องเพิ่มค่า $h$

ที่ $j = 0$ และ $i = 3$

$v_1 = 2 \bmod 8 = 2$

$r = (2 - 2) / 8 = 0$

ตรวจสอบ $r$ โดย $r \bmod \beta = 0 \bmod 8 = 0 \leq 2$ ดังนั้นไม่ต้องเพิ่มค่า $h$

เนื่องจากการคำนวณหา $h$ โดยครบทุกหลักของรูปแบบแทนจำนวน และ $h = 29 \leq 31 = q$ ดังนั้นจะได้รูปแบบแทนจำนวน $V$ โดยการคำนวณดังนี้

ส่วนของการคำนวณหารูปแบบแทนจำนวน $V$

$g = ||V|| + hP = 15 + 29(37) = 1088$

ที่ $i = 0$ จะได้ $v_0 = 1088 \bmod 8 = 0$ และ $g = (1088 - 0) / 8 = 136$

ที่ $i = 1$ จะได้ $v_1 = 136 \bmod 8 = 0$ และ $g = (136 - 0) / 8 = 17$

ที่ $i = 2$ จะได้ $v_2 = 17 \bmod 8 = 1$ และ $g = (17 - 1) / 8 = 2$

ที่ $i = 3$ จะได้ $v_3 = 2 \bmod 8 = 2$ และ $g = (2 - 2) / 8 = 0$

ดังนั้นจะได้รูปแบบแทนจำนวน $V = (2\ 1\ 0\ 0)$ □

จากตัวอย่างที่ 3.1 พิจารณาค่าเชิงตัวเลขของรูปแบบแทนจำนวน $V = (2\ 1\ 0\ 0)$ รูปแบบแทนจำนวน $U = (1\ -1\ -2\ -3)$ และรูปแบบแทนจำนวน $E = V + U$ ที่สอดคล้องกับสมการที่ 3.12 ดังนี้

$$\begin{array}{rl} U: & 1\ \ -1\ \ -2\ \ -3 \\ & \qquad\qquad + \\ V: & \underline{2\ \ 1\ \ 0\ \ 0} \\ E: & \underline{\underline{3\ \ 0\ \ -2\ \ -3}} \end{array}$$

ดังนั้นจะได้รูปแบบแทนจำนวน $E = (3\ 0\ -2\ -3)$ โดยค่าเชิงตัวเลขของรูปแบบแทนจำนวน $U$, $V$ และ $E$ สามารถคำนวณได้ดังนี้

$$\begin{aligned} ||U|| &= ((1)8^3 + (-1)8^2 + (-2)8^1 + (-3)) \bmod 37 \\ &= 429 \bmod 37 \\ &= 22 \\ ||V|| &= ((2)8^3 + (1)8^2 + (0)8^1 + (0)) \bmod 37 \\ &= 1088 \bmod 37 \\ &= 15 \\ ||E|| &= ((3)8^3 + (0)8^2 + (-2)8^1 + (-3)) \bmod 37 \\ &= 1517 \bmod 37 \\ &= 0 \end{aligned}$$

จากสมการที่ 3.13 จะได้ว่า

$$\begin{aligned} (||U|| + ||V||) \bmod 37 &= (22 + 15) \bmod 37 \\ &= 37 \bmod 37 \\ &= 0 \\ &= ||E|| \end{aligned}$$

ดังนั้นรูปแบบแทนจำนวน $E = (3\ 0\ -2\ -3)$ สอดคล้องกับสมการที่ 3.10

ความซับซ้อนเชิงเวลาของอัลกอริทึมที่ 3.1 (อัลกอริทึม Transform($U$)) คือ O($n$) เมื่อ $n$ เป็นจำนวนหลักของรูปแบบแทนจำนวนในระบบจำนวนมอดุลาร์ซ้ำซ้อน จากการคำนวณของอัลกอริทึมที่ 3.1 จะได้รูปแบบแทนจำนวน $V$ ที่ทำให้รูปแบบแทนจำนวน $E$ สอดคล้องกับสมการที่ 3.10 ซึ่งจะนำไปใช้ในการแปลงชุดตัวเลขต่อไป การแปลงชุดตัวเลขของการดำเนินการพื้นฐานเลขคณิตสำหรับระบบจำนวนมอดุลาร์ซ้ำซ้อนมีอัลกอริทึมดังนี้

**อัลกอริทึมที่ 3.2** การแปลงชุดตัวเลข

**Input** : $S = (s_{2n-2}\ s_{2n-3}\ s_{2n-4}\ \dots\ s_0)$ โดยที่ $|s_i| < m\rho$

**Output** : $S' = (s'_{n-1}\ s'_{n-2}\ s'_{n-3}\ \dots\ s'_0)$ โดยที่ $|s'_i| < \rho$ และ $||S'||\ \text{mod}\ P = ||S||\ \text{mod}\ P$

**Begin**

$U \leftarrow S - L$ // รูปแบบแทนจำนวน $L = (l_{n-1}\ l_{n-2}\ l_{n-3}\ \dots\ l_0)$ โดยที่ $l_i = \rho - 1$

$V \leftarrow$ Transform($U$)

$E \leftarrow U + V$

$S' \leftarrow S - E$

**End**

**พิสูจน์** จาก $U \leftarrow S - L$

$E \leftarrow U + V$

และ $S' \leftarrow S - E$

จะได้ $S' = S - (U + V) = (S - U) - V = L - V$

จากทฤษฎีบทที่ 3.1 จะสามารถหารูปแบบแทนจำนวน $V = (v_{n-1}\ v_{n-2}\ v_{n-3}\ \dots\ v_0)$ ได้

โดยที่ $0 \le v_i \le 2\rho - 2$ เมื่อ $i = 0, 1, \dots, n-1$

ดังนั้นจาก $0 \le v_i \le 2\rho - 2$

จะได้ $-2\rho + 2 \le -v_i \le 0$

$(\rho - 1) + (-2\rho + 2) \le l_i - v_i \le (\rho - 1) + 0$

ดังนั้น $-\rho + 1 \le s'_i \le \rho - 1$

ในกรณีที่ $i \ge n$ เนื่องจาก $l_i = v_i = 0$ จะได้ว่า $s'_i = l_i - v_i = 0 - 0 = 0$

ต่อไปจะพิสูจน์ว่า $||E|| = 0\ \text{mod}\ P$

จากอัลกอริทึม 3.1 $||V|| \leftarrow (-||U||)\ \text{mod}\ P$

และ $E \leftarrow U + V$

จะได้ว่า $||E|| = (||U|| + ||V||)\ \text{mod}\ P$

$= (||U|| + (-||U||)\ \text{mod}\ P)\ \text{mod}\ P$

$= (||U|| - ||U||)\ \text{mod}\ P$

$= 0\ \text{mod}\ P$ ■

**ตัวอย่างที่ 3.2** *หาผลบวกของรูปแบบแทนจำนวน A = (1 1 0 -1) และรูปแบบแทนจำนวน B = (1 -1 -1 -1) ของระบบจำนวนมอดุลาร์ซ้ำซ้อน MNS(P, n, $\beta$, $\rho$) = MNS(37, 4, 8, 2)*

**วิธีทำ** หารูปแบบแทนจำนวนของผลบวก $S = A + B$ โดยที่ $s_i = a_i + b_i$ เมื่อ $i$ = 0, 1, 2, 3 ดังนี้

$$\begin{array}{rr} A: & 1\ \ 1\ \ 0\ \ -1 \\ & \qquad\qquad + \\ B: & 1\ \ -1\ \ -1\ \ -1 \\ S: & 2\ \ 0\ \ -1\ \ -2 \end{array}$$

เมื่อได้รูปแบบแทนจำนวน $S$ = (2 0 -1 -2) ต่อไปก็นำไปคำนวณในอัลกอริทึมการแปลงรูปแบบแทนจำนวน (อัลกอริทึมที่ 3.2) ซึ่งจะได้ดังนี้

จาก $U \leftarrow S - L$

จะได้

$$\begin{array}{rr} S: & 2\ \ 0\ \ -1\ \ -2 \\ & \qquad\qquad - \\ L: & 1\ \ 1\ \ 1\ \ 1 \\ U: & 1\ \ -1\ \ -2\ \ -3 \end{array}$$

จาก $V \leftarrow$ Transform($U$) เมื่อ $U$ = (1 -1 -2 -3) และจากตัวอย่างที่ 3.1 จะได้รูปแบบแทนจำนวน $V$ = (2 1 0 0)

จาก $E \leftarrow U + V$

จะได้

$$\begin{array}{rr} U: & 1\ \ -1\ \ -2\ \ -3 \\ & \qquad\qquad + \\ V: & 2\ \ 1\ \ 0\ \ 0 \\ E: & 3\ \ 0\ \ -2\ \ -3 \end{array}$$

จาก $S' \leftarrow S - E$

จะได้

$$\begin{array}{rr} S: & 2\ \ 0\ \ -1\ \ -2 \\ & \qquad\qquad - \\ E: & 3\ \ 0\ \ -2\ \ -3 \\ S': & -1\ \ 0\ \ 1\ \ 1 \end{array}$$

ดังนั้น รูปแบบแทนจำนวน $S'$ = ( -1 0 0 1) □

เมื่อทำการตรวจคำตอบของรูปแบบแทนจำนวน $A$ = (1 1 0 -1) $B$ = (1 -1 -1 -1) และ $S'$ = (-1 0 1 1) จะได้ดังนี้

$||A|| = ((1)8^3 + (1)8^2 + (0)8^1 + (-1)) \bmod 37$

$= 575 \bmod 37$

$= 20$

$||B|| = ((1)8^3 + (-1)8^2 + (-1)8^1 + (-1)) \bmod 37$

$= 439 \bmod 37$

$= 32$

$||S'|| = ((-1)8^3 + (0)8^2 + (1)8^1 + (1)) \bmod 37$

$= -503 \bmod 37$

$= 15$

จาก $(||A|| + ||B||) \bmod 37 = (20 + 32) \bmod 37$

$= 52 \bmod 37$

$= 15$

$= ||S'||$

**ตัวอย่างที่ 3.3** *หาผลลบของรูปแบบแทนจำนวน A = (1 -1 -1 -1) และรูปแบบแทนจำนวน B = (-1 0 1 1) ของระบบจำนวนมอดุลาร์ซ้ำซ้อน MNS(P, n, $\beta$, $\rho$) = MNS(37, 4, 8, 2)*

**วิธีทำ** การหารูปแบบแทนจำนวนของผลลบ $S = A - B$ สามารถคำนวณเหมือนกับการบวกได้ โดย $s_i = a_i + (-b_i)$ เมื่อ $i = 0, 1, 2, 3$ ดังนี้

| | | |
|---|---|---|
| $A$ : | 1 | -1 -1 -1 |
| | | + |
| $-B$ : | 1 | 0 -1 -1 |
| $S$ : | 2 | -1 -2 -2 |

เมื่อได้รูปแบบแทนจำนวน $S = (2\ -1\ -2\ -2)$ ต่อไปก็นำไปคำนวณในอัลกอริทึมการแปลงรูปแบบแทนจำนวน นั่นคืออัลกอริทึมที่ 3.2 ซึ่งจะได้ดังนี้

จาก $U \leftarrow S - L$

จะได้

| | | |
|---|---|---|
| $S$ : | 2 | -1 -2 -2 |
| | | − |
| $L$ : | 1 | 1 1 1 |
| $U$ : | 1 | -2 -3 -3 |

จาก $V \leftarrow$ Transform($U$)

นั่นคือคำนวณหารูปแบบแทนจำนวน $V = (v_{n-1}\ v_{n-2}\ v_{n-3}\ \ldots\ v_0)$ ที่ทำให้รูปแบบแทนจำนวน $E$ สอดคล้องกับสมการที่ 3.12 และสมการที่ 3.13 โดยอัลกอริทึมที่ 3.1 (อัลกอริทึม Transform($U$)) เมื่อรูปแบบแทนจำนวน $U = (1\ -2\ -3\ -3)$ ซึ่งมีขั้นตอนดังนี้

$||U|| = 24$

$||V|| = 13$

$y = 3$

$x = 14$

$q = 31$

ที่ $j = 0$ และ $i = 0$

$v_0 = 0$

$h = 3(13 - 0) \bmod 8 = 7$

$r = \lfloor (13 + 7(37)) / 8 \rfloor = 34$

ตรวจสอบ $r$ โดย $r \bmod \beta = 34 \bmod 8 = 2 \leq 2 = 2\rho - 2$ ดังนั้นไม่ต้องเพิ่มค่า $h$

ที่ $j = 0$ และ $i = 1$

$v_1 = 34 \bmod 8 = 2$

$r = (34 - 2) / 8 = 4$

ตรวจสอบ $r$ โดย $r \bmod \beta = 4 \bmod 8 = 4 > 2$ ดังนั้นต้องเพิ่มค่า $h$ ดังนี้

$f = 31$

ที่ $l = 0$ จะได้ $v_i - l = 2 - 0 = 2 \neq 0$

และ $((v_i - l)x + r) \bmod \beta = ((2 - 0)14 + 4) \bmod 8 = 0 \leq 2$

ดังนั้น $f = \min\{31, 6\} = 6$

ที่ $l = 1$ จะได้ $v_i - l = 2 - 1 = 1 \neq 0$

และ $((v_i - l)x + r) \bmod \beta = ((2 - 1)14 + 4) \bmod 8 = 2 \leq 2$

ดังนั้น $f = \min\{6, 3\} = 3$

ที่ $l = 2$ จะได้ $v_i - l = 2 - 2 = 0$

$h = 7 + (8)3 = 31$

$r = \lfloor (13 + 31(37)) / 8^2 \rfloor = 18$

ที่ $j = 0$ และ $i = 2$

$v_1 = 18 \bmod 8 = 2$

$r = (18 - 2) / 8 = 2$

ตรวจสอบ $r$ โดย $r \bmod \beta = 2 \bmod 8 = 2 \leq 2$ ดังนั้นไม่ต้องเพิ่มค่า $h$

ที่ $j = 0$ และ $i = 3$

$v_1 = 2 \bmod 8 = 2$

$r = (2 - 2) / 8 = 0$

ตรวจสอบ $r$ โดย $r \bmod \beta = 0 \bmod 8 = 0 \leq 2$ ดังนั้นไม่ต้องเพิ่มค่า $h$

เนื่องจากการคำนวณหา $h$ โดยครบทุกหลักของรูปแบบแทนจำนวน และ $h = 31 \leq 31 = q$ ดังนั้นจะได้รูปแบบแทนจำนวน $V$ โดยการคำนวณดังนี้

$g = ||V|| + hP = 13 + 31(37) = 1160$

ที่ $i = 0$ จะได้ $v_0 = 1160 \bmod 8 = 0$ และ $g = (1160 - 0) / 8 = 145$

ที่ $i = 1$ จะได้ $v_1 = 145 \bmod 8 = 1$ และ $g = (145 - 1) / 8 = 18$

ที่ $i = 2$ จะได้ $v_2 = 18 \bmod 8 = 2$ และ $g = (18 - 2) / 8 = 2$

ที่ $i = 3$ จะได้ $v_3 = 2 \bmod 8 = 2$ และ $g = (2 - 2) / 8 = 0$

ดังนั้นจะได้รูปแบบแทนจำนวน $V = (2\ 2\ 1\ 0)$

จาก $E \leftarrow U + V$

จะได้

$U$ : 1 -2 -3 -3

\+

$V$ : 2 2 1 0

$E$ : 3 0 -2 -3

จาก $S' \leftarrow S - E$

จะได้

$S$ : 2 -1 -2 -2

\-

$E$ : 3 0 -2 -3

$S'$ : -1 -1 0 1

ดังนั้น รูปแบบแทนจำนวน $S' = (-1\ -1\ 0\ 1)$ □

เมื่อทำการตรวจคำตอบของรูปแบบแทนจำนวน $A = (1\ -1\ -1\ -1)$ $B = (-1\ 0\ 1\ 1)$ และ $S' = (-1\ -1\ 0\ 1)$ จะได้ดังนี้

$||A|| = ((1)8^3 + (-1)8^2 + (-1)8^1 + (-1)) \bmod 37$

$= 439 \bmod 37$

$= 32$

$||B||$ = $((-1)8^3 + (0)8^2 + (1)8^1 + (1))$ mod 37

= -503 mod 37

= 15

$||S'||$ = $((-1)8^3 + (-1)8^2 + (0)8^1 + (1))$ mod 37

= -575 mod 37

= 17

จาก $(||A|| - ||B||)$ mod 37 = (32 - 15) mod 37

= 17 mod 37

= 17

= $||S'||$

**ตัวอย่างที่ 3.4** *หาผลคูณของรูปแบบแทนจำนวน A = (1 1 -1 -1) และรูปแบบแทนจำนวน B = (1 1 1 1) ของระบบจำนวนมอดุลาร์ซ้ำซ้อน MNS(P, n, $\beta$, $\rho$) = MNS(37, 4, 8, 2)*

**วิธีทำ** หารูปแบบแทนจำนวนของผลคูณ $S = A \times B$ โดย

$$s_i = \begin{cases} \sum_{j=0}^{i} a_{i-j} b_j & ; i = 0,1,..,n-1 \\ \sum_{j=i-n+1}^{n-1} a_{i-j} b_j & ; i = n, n+1,..,2n-2 \end{cases}$$

จะได้ $s_0 = a_0 b_0 = (-1)(1) = -1$

$s_1 = a_1 b_0 + a_0 b_1 = (-1)(1) + (-1)(1) = -2$

$s_2 = a_2 b_0 + a_1 b_1 + a_0 b_2 = (1)(1) + (-1)(1) + (-1)(1) = -1$

$s_3 = a_3 b_0 + a_2 b_1 + a_1 b_2 + a_0 b_3 = (1)(1) + (1)(1) + (-1)(1) + (-1)(1) = 0$

$s_4 = a_3 b_1 + a_2 b_2 + a_1 b_3 = (1)(1) + (1)(1) + (-1)(1) = 1$

$s_5 = a_3 b_2 + a_2 b_3 = (1)(1) + (1)(1) = 2$

$s_6 = a_3 b_3 = (1)(1) = 1$

เมื่อได้รูปแบบแทนจำนวน $S$ = (1 2 1 0 -1 -2 -1) ต่อไปก็นำไปคำนวณในอัลกอริทึมการแปลงรูปแบบแทนจำนวน นั่นคืออัลกอริทึมที่ 3.2 ซึ่งจะได้ดังนี้

จาก $U \leftarrow S - L$

จะได้

$S$ : 1 2 1 0 -1 -2 -1

$L$ : 1 1 1 1

$U$ : 1 2 1 -1 -2 -3 -2

จาก $V \leftarrow$ Transform($U$)

นั่นคือคำนวณหารูปแบบแทนจำนวน $V = (v_{n-1}\ v_{n-2}\ v_{n-3}\ \ldots\ v_0)$ ที่ทำให้รูปแบบแทนจำนวน $E$ สอดคล้องกับสมการที่ 3.12 และสมการที่ 3.13 โดยอัลกอริทึมที่ 3.1 (อัลกอริทึม Transform($U$)) เมื่อรูปแบบแทนจำนวน $U$ = (1 2 1 -1 -2 -3 -2) ซึ่งมีขั้นตอนดังนี้

$||U|| = 34$

$||V|| = 3$

$y = 3$

$x = 14$

$q = 31$

ที่ $j = 0$ และ $i = 0$

$v_0 = 0$

$h = 3(3 - 0) \bmod 8 = 1$

$r = \lfloor (3 + 1(37)) / 8 \rfloor = 5$

ตรวจสอบ $r$ โดย $r \bmod \beta = 5 \bmod 8 = 5 > 2 = 2\rho - 2$ ดังนั้นต้องเพิ่มค่า $h$ ดังนี้

$h = 1 + 8((3(5) \bmod 8) \bmod 3) = 9$

$r = 5 + 37((3(5) \bmod 8) \bmod 3) = 42$

ที่ $j = 0$ และ $i = 1$

$v_1 = 42 \bmod 8 = 2$

$r = (42 - 2) / 8 = 5$

ตรวจสอบ $r$ โดย $r \bmod \beta = 5 \bmod 8 = 5 > 2$ ดังนั้นต้องเพิ่มค่า $h$ ดังนี้

$f = 31$

ที่ $l = 0$ จะได้ $v_i - l = 2 - 0 = 2 \neq 0$

และ $((v_i - l)x + r) \bmod \beta = ((2 - 0)14 + 5) \bmod 8 = 1 \leq 2$

ดังนั้น $f = \min\{31, 6\} = 6$

ที่ $l = 1$ จะได้ $v_i - l = 2 - 1 = 1 \neq 0$

และ $((v_i - l)x + r) \bmod \beta = ((2 - 1)14 + 5) \bmod 8 = 3 > 2$

ที่ $l = 2$ จะได้ $v_i - l = 2 - 2 = 0$

$h = 9 + (8)6 = 57$

$r = \lfloor (3 + 57(37)) / 8^2 \rfloor = 33$

เนื่องจาก $h = 57 > 31 = q$ ดังนั้นจึงทำการคำนวณหา $h$ ใหม่ ดังนี้

ที่ $j = 1$ และ $i = 0$

$v_0 = 1$

$h = 3(3 - 1) \bmod 8 = 6$

$r = \lfloor (3 + 6(37)) / 8 \rfloor = 28$

ตรวจสอบ $r$ โดย $r \bmod \beta = 28 \bmod 8 = 4 > 2$ ดังนั้นต้องเพิ่มค่า $h$

$h = 6 + 8((3(28) \bmod 8) \bmod 3) = 14$

$r = 28 + 37((3(28) \bmod 8) \bmod 3) = 65$

ที่ $j = 1$ และ $i = 1$

$v_1 = 65 \bmod 8 = 1$

$r = (65 - 1) / 8 = 8$

ตรวจสอบ $r$ โดย $r \bmod \beta = 8 \bmod 8 = 0 \leq 2$ ดังนั้นไม่ต้องเพิ่มค่า $h$

ที่ $j = 1$ และ $i = 2$

$v_1 = 8 \bmod 8 = 0$

$r = (8 - 0) / 8 = 1$

ตรวจสอบ $r$ โดย $r \bmod \beta = 1 \bmod 8 = 1 \leq 2$ ดังนั้นไม่ต้องเพิ่มค่า $h$

ที่ $j = 1$ และ $i = 3$

$v_1 = 1 \bmod 8 = 1$

$r = (1 - 1) / 8 = 0$

ตรวจสอบ $r$ โดย $r \bmod \beta = 0 \bmod 8 = 0 \leq 2$ ดังนั้นไม่ต้องเพิ่มค่า $h$

เนื่องจากการคำนวณหา $h$ โดยครบทุกหลักของรูปแบบแทนจำนวน และ $h = 14 \leq 31 = q$ ดังนั้นจะได้รูปแบบแทนจำนวน $V$ โดยการคำนวณดังนี้

$g = ||V|| + hP = 3 + 14(37) = 521$

ที่ $i = 0$ จะได้ $v_0 = 521 \bmod 8 = 1$ และ $g = (521 - 1) / 8 = 65$

ที่ $i = 1$ จะได้ $v_1 = 65 \bmod 8 = 1$ และ $g = (65 - 1) / 8 = 8$

ที่ $i = 2$ จะได้ $v_2 = 8 \bmod 8 = 0$ และ $g = (8 - 0) / 8 = 1$

ที่ $i = 3$ จะได้ $v_3 = 1 \bmod 8 = 1$ และ $g = (1 - 1) / 8 = 0$

ดังนั้นจะได้รูปแบบแทนจำนวน $V = (1\ 0\ 1\ 1)$

จาก $E \leftarrow U + V$

จะได้

$U$ : 1 2 1 -1 -2 -3 -2

$+$

$V$ : 1 0 1 1

$E$ : 1 2 1 0 -2 -2 -1

จาก $S' \leftarrow S - E$

จะได้

$S$ : 1 2 1 0 -1 -2 -1

$-$

$E$ : 1 2 1 0 -2 -2 -1

$S'$ : 0 0 0 0 1 0 0

ดังนั้น รูปแบบแทนจำนวน $S'$ = ( 0 1 0 0) □

เมื่อทำการตรวจคำตอบของรูปแบบแทนจำนวน $A$ = (1 1 -1 -1) $B$ = (1 1 1 1) และ $S'$ = (0 1 0 0) จะได้ดังนี้

$$||A|| = ((1)8^3 + (1)8^2 + (-1)8^1 + (-1)) \bmod 37$$
$$= 567 \bmod 37$$
$$= 12$$

$$||B|| = ((1)8^3 + (1)8^2 + (1)8^1 + (1)) \bmod 37$$
$$= 585 \bmod 37$$
$$= 30$$

$$||S'|| = ((0)8^3 + (1)8^2 + (0)8^1 + (0)) \bmod 37$$
$$= 64 \bmod 37$$
$$= 27$$

จาก
$$(||A|| \times ||B||) \bmod 37 = (12 \times 30) \bmod 37$$
$$= 360 \bmod 37$$
$$= 27$$
$$= ||S'||$$

## 3.2 สรุป

ในงานวิจัยนี้เสนอการแปลงชุดตัวเลขของการดำเนินการพื้นฐานเลขคณิตสำหรับระบบจำนวนมอดุลาร์ซ้ำซ้อน ได้แก่ การบวก การลบ และการคูณ โดยขั้นตอนการทำงานจะเริ่มจากอัลกอริทึมที่ 3.2 (อัลกอริทึมการแปลงชุดตัวเลข) และอัลกอริทึม 3.1 (อัลกอริทึม Transform($U$)) ตามลำดับ ความซับซ้อนเชิงเวลาคือ O($n$) เมื่อ $n$ คือ จำนวนหลักของรูปแบบแทนจำนวน (ซึ่งสามารถพิจารณาเป็นค่าคงที่ได้) เมื่อพิจารณาเวลาในรูปของ $m$ เมื่อ $m\rho$ เป็นขนาดของชุดตัวเลขของรูปแบบแทนจำนวนก่อนการแปลงชุดตัวเลข และ $\rho$ เป็นขนาดของชุดตัวเลขของรูปแบบแทนจำนวนหลังการแปลงชุดตัวเลข มีความซับซ้อนเชิงเวลาคือ O(1) ดังนั้นการดำเนินการพื้นฐานเลขคณิตสำหรับระบบจำนวนมอดุลาร์ซ้ำซ้อน ได้แก่ การบวก การลบ และการคูณ มีความซับซ้อนเชิงเวลาเท่ากันคือ O(1)

# บทที่ 4

## วิเคราะห์การดำเนินการพื้นฐานเลขคณิตสำหรับระบบจำนวนมอดุลาร์ซ้ำซ้อน

เนื่องจากการแปลงชุดตัวเลขของการบวก การลบ และการคูณ มีความซับซ้อนเชิงเวลาเท่ากัน นั่นคือ O(1) ดังนั้นการบวกหรือการลบกันแบบหลายจำนวนจะต้องมีความซับซ้อนเชิงเวลาคือ O(1) ด้วย ในการคำนวณการลบสามารถคำนวณได้เช่นเดียวกับการบวกโดยการกลับเครื่องหมายของดิจิตของรูปแบบแทนจำนวน ดังนั้นจะพิจารณาเฉพาะการบวกแบบหลายจำนวนดังนี้

### 4.1 การบวกแบบหลายจำนวน

รูปแบบแทนจำนวน $A_c = (a_{c,n-1}\ a_{c,n-2}\ a_{c,n-3}\ \ldots\ a_{c,0})$ เมื่อ $1 \le c \le s$ เป็นรูปแบบแทนจำนวน $s$ จำนวนในระบบจำนวนมอดุลาร์ซ้ำซ้อน MNS($P$, $n$, $\beta$, $\rho$) ผลบวกของรูปแบบแทนจำนวน $s$ จำนวนเป็นการคำนวณหารูปแบบแทนจำนวน $Z' = (z'_{n-1}\ z'_{n-2}\ z'_{n-3}\ \ldots\ z'_0)$ ที่สอดคล้องกับสมการที่ 4.1 ดังต่อไปนี้นี้

$$A_1 + A_2 + A_3 + \ldots + A_s = Z'$$
$$(||A_1|| + ||A_2|| + ||A_3|| + \ldots + ||A_s||) \bmod P = ||Z'|| \bmod P \qquad (4.1)$$

เมื่อ $||A_c|| = \sum_{i=0}^{n-1} a_i \beta^i \bmod P$ และ $||Z'|| = \sum_{i=0}^{n-1} z'_i \beta^i \bmod P$ เป็นค่าเชิงตัวเลขของรูปแบบแทนจำนวน $A_c$ และ $Z'$ ตามลำดับ เมื่อ $1 \le c \le s$ โดยที่ $|z'_i| < \rho$

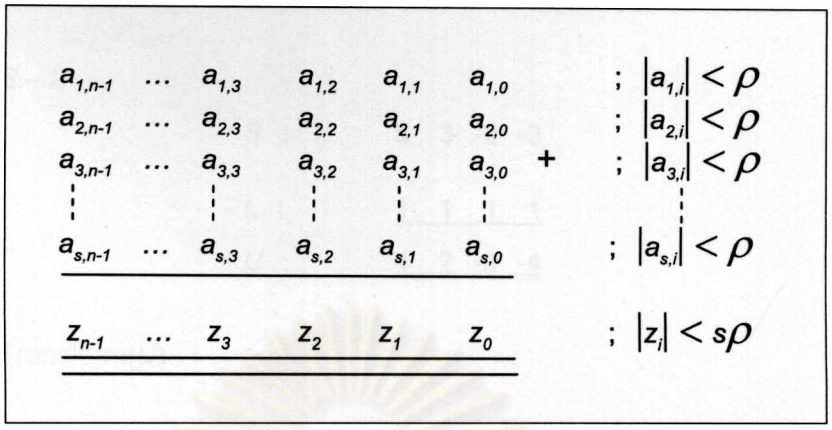
การบวกแบบหลายจำนวนในระบบจำนวนมอดุลาร์ซ้ำซ้อนจะคำนวณในแต่ละหลักพร้อมกัน เพื่อความเร็วในการคำนวณ ซึ่งจะมีบทนิยามดังนี้

**บทนิยามที่ 4.1** รูปแบบแทนจำนวน $A_c = (a_{c,n-1}\ a_{c,n-2}\ a_{c,n-3}\ \ldots\ a_{c,0})$ เมื่อ $1 \le c \le s$ เป็นรูปแบบแทนจำนวน $s$ จำนวนในระบบจำนวนมอดุลาร์ซ้ำซ้อน MNS($P$, $n$, $\beta$, $\rho$) โดยที่ $|a_{c,i}| < \rho$ ผลบวกของรูปแบบแทนจำนวน $s$ จำนวน ($A_1 + A_2 + A_3 + \ldots + A_s = Z$) สามารถคำนวณได้ดังสมการ

$$a_{1,i} + a_{2,i} + a_{3,i} + \ldots + a_{s,i} = z_i \qquad (4.2)$$

เมื่อ $i = 0, 1, \ldots, n\text{-}1$

เพื่อความเข้าใจให้พิจารณารูปที่ 4.1




รูปที่ 4.1 แสดงการคำนวณการบวกของรูปแบบแทนจำนวน $s$ จำนวน

เนื่องจาก $|z_i| < s\rho$ แสดงว่ารูปแบบแทนจำนวน $Z = (z_{n-1}\ z_{n-2}\ z_{n-3}\ \ldots\ z_0)$ ไม่เป็นรูปแบบแทนจำนวนในระบบจำนวนมอดุลาร์ซ้ำซ้อน MNS($P$, $n$, $\beta$, $\rho$) ดังนั้นจึงต้องทำการแปลงชุดตัวเลขเพื่อให้ได้รูปแบบแทนจำนวน $Z' = (z'_{n-1}\ z'_{n-2}\ z'_{n-3}\ \ldots\ z'_0)$ ที่สอดคล้องกับสมการที่ 4.1 ดังกล่าวข้างต้นและสมการที่ 4.3 ดังต่อไปนี้

$$||Z'|| \bmod P = ||Z|| \bmod P \qquad (4.3)$$

โดยที่ $|z'_i| < \rho$

ในการแปลงชุดตัวเลขของการบวกแบบหลายจำนวนสามารถคำนวณได้โดยอัลกอริทึมที่ 3.2 และอัลกอริทึมที่ 3.1 ตามลำดับ

**ตัวอย่างที่ 4.1** *หาผลบวกของรูปแบบแทนจำนวน 5 จำนวน คือ $A_1$ = (1 1 0 -1) $A_2$ = (1 0 -1 -1) $A_3$ = (1 1 -1 1) $A_4$ = (1 1 -1 -1) และ $A_5$ = (1 0 1 -1) ซึ่งถูกนิยามในระบบจำนวนมอดุลาร์ซ้ำซ้อน MNS($P$, $n$, $\beta$, $\rho$) = MNS(37, 4, 8, 2)*

**วิธีทำ** การหารูปแบบแทนจำนวนของผลบวก $S = A_1 + A_2 + A_3 + A_4 + A_5$ สามารถหาได้โดย $s_i = a_{1,i} + a_{2,i} + a_{3,i} + a_{4,i} + a_{5,i}$ เมื่อ $i$ = 0, 1, 2, 3 ซึ่งจะได้รูปแบบแทนจำนวน $Z$ ดังนี้

$A_1$ : 1 1 0 -1
$A_2$ : 1 0 -1 -1
$A_3$ : 1 1 -1 1 +
$A_4$ : 1 1 -1 -1
$A_5$ : 1 0 1 -1
$S$ : 5 3 -2 -3

เมื่อได้รูปแบบแทนจำนวน $S$ = (5 3 -2 -3) ต่อไปก็นำไปคำนวณในอัลกอริทึมการแปลงรูปแบบแทนจำนวน (อัลกอริทึมที่ 3.2) ซึ่งจะได้ดังนี้

จาก $U \leftarrow S - L$

จะได้

$S$ : 5 3 -2 -3

$L$ : 1 1 1 1

$U$ : 4 2 -3 -4

จาก $V \leftarrow$ Transform($U$)

นั่นคือคำนวณหารูปแบบแทนจำนวน $V = (v_{n-1}\ v_{n-2}\ v_{n-3}\ \ldots\ v_0)$ ที่ทำให้รูปแบบแทนจำนวน $E$ สอดคล้องกับสมการที่ 3.12 และสมการที่ 3.13 โดยอัลกอริทึมที่ 3.1 (อัลกอริทึม Transform($U$)) เมื่อรูปแบบแทนจำนวน $U$ = (4 2 -3 -4) ซึ่งมีขั้นตอนดังนี้

$||U|| = 2$

$||V|| = 35$

y = 3

x = 14

q = 3

ที่ $j = 0$ และ $i = 0$

$v_0 = 0$

$h = 3(35 - 0) \bmod 8 = 1$

$r = \lfloor (35 + 1(37)) / 8 \rfloor = 9$

ตรวจสอบ $r$ โดย r mod $\beta$ = 9 mod 8 = 1 $\leq$ 2 = 2$\rho$ - 2 ดังนั้นไม่ต้องเพิ่มค่า $h$

ที่ $j = 0$ และ $i = 1$

$v_1 = 9 \bmod 8 = 1$

$r = (9 - 1) / 8 = 1$

ตรวจสอบ $r$ โดย r mod $\beta$ = 1 mod 8 = 1 $\leq$ 2 ดังนั้นไม่ต้องเพิ่มค่า $h$

ที่ $j = 0$ และ $i = 2$

$v_1 = 1 \bmod 8 = 1$

$r = (1 - 1) / 8 = 0$

ตรวจสอบ $r$ โดย r mod $\beta$ = 0 mod 8 = 0 $\leq$ 2 ดังนั้นไม่ต้องเพิ่มค่า $h$

ที่ $j = 0$ และ $i = 3$

$v_1 = 0 \bmod 8 = 0$

$r = (0 - 0) / 8 = 0$

ตรวจสอบ $r$ โดย $r \bmod \beta = 0 \bmod 8 = 0 \leq 2$ ดังนั้นไม่ต้องเพิ่มค่า $h$

เนื่องจากการคำนวณหา $h$ โดยครบทุกหลักของรูปแบบแทนจำนวน และ $h = 1 \leq 31 = q$ ดังนั้นจะได้รูปแบบแทนจำนวน $V$ โดยการคำนวณดังนี้

$g = ||V|| + hP = 35 + 1(37) = 72$

ที่ $i = 0$ จะได้ $v_0 = 72 \bmod 8 = 0$ และ $g = (72 - 0) / 8 = 9$

ที่ $i = 1$ จะได้ $v_1 = 9 \bmod 8 = 1$ และ $g = (9 - 1) / 8 = 1$

ที่ $i = 2$ จะได้ $v_2 = 1 \bmod 8 = 1$ และ $g = (1 - 1) / 8 = 0$

ที่ $i = 3$ จะได้ $v_3 = 0 \bmod 8 = 0$ และ $g = (0 - 0) / 8 = 0$

ดังนั้นจะได้รูปแบบแทนจำนวน $V = (0\ 1\ 1\ 0)$

จาก $E \leftarrow U + V$

จะได้

| | | |
|---|---|---|
| $U$ : | 4 2 -3 -4 | |
| | | + |
| $V$ : | 0 1 1 0 | |
| $E$ : | 4 3 -2 -4 | |

จาก $S' \leftarrow S - E$

จะได้

| | | |
|---|---|---|
| $S$ : | 5 3 -2 -3 | |
| | | − |
| $E$ : | 4 3 -2 -4 | |
| $S'$ : | 1 0 0 1 | |

ดังนั้น รูปแบบแทนจำนวน $S' = (1\ 0\ 0\ 1)$ □

เมื่อทำการตรวจคำตอบของรูปแบบแทนจำนวน $A_1 = (1\ 1\ 0\ -1)$ $A_2 = (1\ 0\ -1\ -1)$ $A_3 = (1\ 1\ -1\ 1)$ $A_4 = (1\ 1\ -1\ -1)$ $A_5 = (1\ 0\ 1\ -1)$ และ $S' = (1\ 0\ 0\ 1)$ จะได้ดังนี้

$||A_1|| = ((1)8^3 + (1)8^2 + (0)8^1 + (-1)) \bmod 37$

$= 575 \bmod 37$

$= 20$

$||A_2|| = ((1)8^3 + (0)8^2 + (-1)8^1 + (-1)) \bmod 37$

$= 503 \bmod 37$

$= 22$

$||A_3|| = ((1)8^3 + (1)8^2 + (-1)8^1 + (1)) \bmod 37$

$= 569 \bmod 37$

$= 14$

$||A_4|| = ((1)8^3 + (1)8^2 + (-1)8^1 + (-1)) \bmod 37$

$= 567 \bmod 37$

$= 12$

$||A_5|| = ((1)8^3 + (0)8^2 + (1)8^1 + (-1)) \bmod 37$

$= 519 \bmod 37$

$= 1$

$||S'|| = ((1)8^3 + (0)8^2 + (0)8^1 + (1)) \bmod 37$

$= 513 \bmod 37$

$= 32$

จาก $(||A_1|| + ||A_2|| + ||A_3|| + ||A_4|| + ||A_5||) \bmod 37 = (20 + 22 + 14 + 12 + 1) \bmod 37$

$= 69 \bmod 37$

$= 32$

$= ||S'||$

**4.2 สรุป**

เวลาที่ใช้ในการคำนวณการแปลงชุดตัวเลขของการบวกหรือการลบแบบหลายจำนวนไม่ขึ้นกับขนาดของชุดตัวเลขของรูปแบบแทนจำนวนก่อนการแปลงชุดตัวเลข นั่นคือมีความซับซ้อนเชิงเวลาคือ O(1)

# บทที่ 5

## สรุปผลการวิจัยและข้อเสนอแนะ

### 5.1 สรุปผลการวิจัย

ระบบจำนวนมอดุลาร์เป็นระบบที่ไม่มีการทด ดังนั้นการคำนวณพื้นฐานเลขคณิตจะพิจารณาในรูปของปัญหาการแปลงชุดตัวเลข ซึ่งเวลาที่ใช้ในการคำนวณพื้นฐานเลขคณิตจะขึ้นอยู่กับเวลาในการแปลงชุดตัวเลข การแปลงชุดตัวเลขจะอาศัยคุณสมบัติของความซ้ำซ้อนของศูนย์ นั่นคืออาศัยการลบด้วยรูปแบบแทนจำนวนที่มีค่าเชิงตัวเลขเป็นศูนย์ เนื่องจากรูปแบบแทนจำนวนที่มีค่าเชิงตัวเลขเป็นศูนย์มีหลายรูปแบบ ผลที่ตามมาคือรูปแบบแทนจำนวนใดที่จะเหมาะสมโดยไม่ทำให้เกิดการวนซ้ำแบบไม่รู้จบของการแปลงชุดตัวเลข ซึ่งทำให้หาคำตอบของการแปลงชุดตัวเลขไม่ได้ ปัจจุบันมีงานวิจัยเกี่ยวกับการแปลงชุดตัวเลขของระบบจำนวนมอดุลาร์ ซึ่งจะทำการคำนวณวนซ้ำแบบรู้จบ และแต่ละรอบของการคำนวณจะต้องคำนวณหารูปแบบแทนจำนวนที่มีค่าเชิงตัวเลขเป็นศูนย์ใหม่ทุกครั้ง จำนวนรอบของการวนซ้ำจะขึ้นอยู่กับขนาดของชุดตัวเลขของรูปแบบแทนจำนวนก่อนการแปลงชุดตัวเลข แสดงว่าเวลาในการคำนวณการแปลงชุดตัวเลขขึ้นอยู่กับขนาดของชุดตัวเลขของรูปแบบแทนจำนวนก่อนการแปลงชุดตัวเลข โดยที่ความซับซ้อนเชิงเวลาเท่ากับ $O(\log_2 m)$ เมื่อขนาดของชุดตัวเลขของรูปแบบแทนจำนวนก่อนการแปลงชุดตัวเลขคือ $m\rho$ และขนาดของชุดตัวเลขของรูปแบบแทนจำนวนหลังการแปลงชุดตัวเลขคือ $\rho$

ในงานวิจัยนี้ได้เสนอการดำเนินการพื้นฐานเลขคณิตสำหรับระบบจำนวนมอดุลาร์ซ้ำซ้อน ได้แก่ การบวก การลบ และการคูณ และออกแบบรูปแบบแทนจำนวน $E$ ที่มีค่าเชิงตัวเลขเป็นศูนย์ โดยที่รูปแบบแทนจำนวน $E$ เท่ากับผลบวกของรูปแบบแทนจำนวน $U$ และรูปแบบแทนจำนวน $V$ เพื่อลดจำนวนรอบในการแปลงชุดตัวเลข จึงมีการกำหนดรูปแบบแทนจำนวน $U$ ให้สอดคล้องกับรูปแบบแทนจำนวนก่อนการแปลงชุดตัวเลข แต่ไม่มีการกำหนดรูปแบบแทนจำนวน $V$ ดังนั้นในการแปลงชุดตัวเลขจะทำการคำนวณหารูปแบบแทนจำนวน $V$ แทนการหารูปแบบแทนจำนวน $E$ ซึ่งได้เสนอทฤษฎีบทที่ 3.1 คือมีรูปแบบแทนจำนวน $V$ ที่ทำให้รูปแบบแทนจำนวนหลังการแปลงชุดตัวเลขเป็นรูปแบบแทนจำนวนในระบบจำนวนมอดุลาร์ซ้ำซ้อนพร้อมบทพิสูจน์ และเสนออัลกอริทึมที่ 3.1 เป็นอัลกอริทึมการหารูปแบบแทนจำนวน $V$ พร้อมบทพิสูจน์ ส่วนอัลกอริทึมการแปลงชุดตัวเลข ในงานวิจัยนี้ได้เสนออัลกอริทึมที่ 3.2 พร้อมทั้งบทพิสูจน์ ความซับซ้อนเชิงเวลาคือ $O(n)$ เมื่อ $n$ คือ จำนวนหลักของรูปแบบแทนจำนวน (ซึ่งสามารถพิจารณาเป็นค่าคงที่ได้) เมื่อพิจารณาเวลาในรูปของ $m$ เมื่อ $m\rho$ เป็นขนาดของชุดตัวเลขของรูปแบบแทนจำนวนก่อนการแปลงชุดตัวเลข และ $\rho$ เป็นขนาดของชุดตัวเลขของรูปแบบแทนจำนวนหลังการแปลงชุดตัวเลข มีความซับซ้อนเชิงเวลาคือ $O(1)$ ดังนั้นการ

ดำเนินการการบวก การลบ และการคูณสำหรับระบบจำนวนมอดุลาร์ซ้ำซ้อนมีความซับซ้อนเชิงเวลาเท่ากันคือ O(1)

## 5.2 ข้อเสนอแนะ

ในอัลกอริทึมที่ 3.1 (อัลกอริทึม Transform($U$)) การคำนวณหารูปแบบแทนจำนวน $V$ จะต้องกำหนดค่าเริ่มต้น $v_0$ เพื่อหา $h$ ที่สอดคล้องกับค่าเริ่มต้น เมื่อพิจารณาค่าของ $v_0$ ที่ต้องการคือมีค่าอยู่ในช่วง 0 ถึง $2\rho$ - 2 ซึ่งไม่สามารถระบุได้ว่าค่าของ $v_0$ ค่าไหนที่ทำให้ได้รูปแบบแทนจำนวน $V$ ที่ต้องการ ดังนั้นในส่วนของการกำหนดค่าเริ่มต้น $v_0$ จะกำหนดให้เท่ากับ 0 ถึง $2\rho$ - 2 โดยกำหนดทีละค่าแล้วคำนวณหา $h$ ในส่วนถัดไป ถ้า h ที่ได้มีค่ามากกว่าที่ต้องการจะทำการกำหนดค่าเริ่มต้น $v_0$ ใหม่และคำนวณหา $h$ ใหม่ต่อไป ดังนั้นในกรณีที่ใช้เวลาน้อยที่สุดคือ $n$ นั่นคือกำหนดค่าเริ่มต้น $v_0$ เพียงครั้งเดียวก็สามารถคำนวณหา $h$ ที่ต้องการได้ และกรณีที่ใช้เวลามากที่สุดคือ $(2\rho - 2)n$ นั่นคือกำหนดค่าเริ่มต้น $v_0$ เป็นจำนวน $2\rho$ - 2 ครั้งจึงจะสามารถคำนวณหา $h$ ที่ต้องการได้ ดังนั้นถ้าสามารถออกแบบอัลกอริทึมคำนวณหารูปแบบแทนจำนวน $V$ โดยไม่ต้องกำหนดค่าเริ่มต้น $v_0$ หรือกำหนดค่าเริ่มต้น $v_0$ เพียงครั้งเดียวโดยไม่ต้องกำหนดค่าเริ่มต้น $v_0$ ใหม่สำหรับทุกกรณีก็จะทำให้เวลาในการคำนวณลดลงด้วย

# รายการอ้างอิง


[1] J. Solinas. Generalized mersenne numbers. Research Report CORR. (1999).

[2] J. Chung and A. Hasan. More generalized mersenne numbers. In M. Matsui and R. Zuccherato, editors, Selected Areas in Cryptography – SAC. Vol, 3006 (2004): 335–347.

[3] W. Diffie and M. E. Hellman. New directions in cryptography. IEEE Transactions on Information Theory. (1976): 644–654.

[4] R. Rivest, A. Shamir, and L. Adleman. A method for obtaining digital signatures and public key cryptosystems. Communications of the ACM. (1978): 120–126.

[5] J.-C. Bajard, L. Imbert, and T. Plantard. Modular number systems: Beyond the Mersenne family. In Selected Areas in Cryptography: 11th International Workshop. Vol, 3357 (2005): 159–169.

[6] J.-C. Bajard, L. Imbert, and T. Plantard. Arithmetic operations in the polynomial modular number system. Research Report LIRMM - CNRS. 2004.

[7] J.-C. Bajard, L. Imbert, and T. Plantard. Arithmetic operations in the polynomial modular number system. IEEE Symposium on Computer Arithmetic. (2005): 1063-6889.

[8] C.-F. Gauss. Disquisitiones Arithmeticae. Lipsiae.(1801).

[9] J.-C. Bajard, L.-S. Didier, and P. Kornerup. An RNS montgomery modular multiplication algorithm. IEEE Transactions on Computers. (1997): 1063-6889.

[10] M. Ciet, M. Neve, E. Peeters, and J.-J. Quisquater. Parallel FPGA implementation of RSA with residue number systems. IEEE Midwest Symposium on Circuits and Systems. (2003).

[11] B. Parhami. Computer Arithmetic Algorithm and hardware designs. Oxford, New York: Oxford University Press. (2000).

# ประวัติผู้เขียนวิทยานิพนธ์

นายสัพพชัย อยู่เย็น เกิดเมื่อวันที่ 16 มิถุนายน พ.ศ. 2526 ที่จังหวัดประจวบคีรีขันธ์ สำเร็จการศึกษาระดับมัธยมศึกษาตอนปลายจากโรงเรียนพรหมานุสรณ์จังหวัดเพชรบุรี อำเภอเมือง จังหวัดเพชรบุรี เข้าศึกษาต่อในระดับปริญญาบัณฑิต สาขาวิชาคณิตศาสตร์ คณะวิทยาศาสตร์ มหาวิทยาลัยศิลปากร จนสำเร็จการศึกษาในปี พ.ศ. 2548 และศึกษาต่อในระดับปริญญามหาบัณฑิต สาขาวิชาวิทยาศาสตร์คอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย