บทที่ 3

แผนการดำเนินงาน

3.1 หลักเกณฑ์และข้อพิจารณาในการสร้างตัวเลขสุ่ม

โดยปรกติแล้วการสร้างตัวเลขสุ่มนั้น เราสร้างได้หลายวิธีต่าง ๆ กัน แต่วิธีที่เป็นที่นิยมใช้กันมาก คือวิธี Congruence [1] ซึ่งมี Congruence นี้มีวิธีการสร้างที่ต้องพิจารณาถึงหลายแฟคเตอร์ เพื่อให้สามารถสร้างตัวเลขสุ่มได้ยาวที่สุด และมีความสุ่มมากที่สุด เราจะต้องพิจารณาถึง

3.1.1 การเลือกค่าเริ่มต้น (Initial value)

การเลือกค่าเริ่มต้นของการสร้างตัวเลขสุ่มมีความสำคัญต่อความยาวของชุดลำดับ เพราะถ้าเราเลือกไม่ดีแล้ว จะทำให้เราได้ชุดลำดับของตัวเลขสุ่มสั้นเกินไป การเลือกค่าเริ่มต้นเรามีหลักทั่ว ๆ ไป คือ

1. ตัวเลขที่เป็นค่าเริ่มต้นต้องเป็นเลขคี่
2. ต้องเป็นตัวเลขที่หารด้วย 5 ไม่ลงตัว
3. ควรจะใช้ตัวเลข 6 หลัก หรือมากกว่านั้น

3.1.2 การเลือกตัว Modulus

โดยปรกติการเลือกตัว modulus มีความสำคัญมากต่อความยาวของเลขชุดลำดับ ที่เราจะได้ เพราะค่า modulus หรือ m ที่เราเลือกนั้นจะมีผลต่อความยาวคาบของตัวเลขสุ่ม ซึ่งความยาวของตัวเลขสุ่มนี้ จะมีได้ไม่เกิน m นอกจากนี้การเลือกค่า m ยังมีผลต่อความเร็วในการสร้างตัวเลขสุ่มอีกด้วย เพราะปรกติแล้วการหารนั้นคอมพิวเตอร์ทำได้ช้ากว่าการบวก [3]

---

1. Donald E. Knuth, The Art of Computer Programming, Vol 2. page
2. Joe H. Mize and J. Grady Cox, Essentials of Simulation page 38 - 39
3. Donald E. Knuth, The Art of Computer Programming, Vol 2 page 11 - 15

ถ้าเรากำหนดให้ w เป็น word size ของคอมพิวเตอร์แล้ว การทำ modulus w เราจะสามารถทำด้วยวิธีบวกแทนวิธีการหารและการคูณ mod w ก็กระทำได้ง่ายเช่นเดียวกัน ดังนั้นการกำหนดให้ w เท่ากับ word size ของคอมพิวเตอร์จะช่วยลดระยะเวลาทำงานของเครื่องคอมพิวเตอร์ลง ทำให้การสร้างตัวเลขสุ่มทำได้รวดเร็วยิ่งขึ้น [4]

นอกจากนี้ เราต้องใช้ความรู้เกี่ยวกับแฟคเตอร์เฉพาะของ w เพื่อเลือก multiplier, a ที่ถูกต้อง ตารางที่ 2 นี้แสดงการแยกตัวประกอบของ $w \pm 1$ เป็นตัวเฉพาะเกือบทุกค่า word size ที่เรารู้จัก ถ้าต้องการขยายตารางนี้ออกไปอีก เราสามารถใช้วิธีแยกตัวประกอบของ $y^n \pm 1$ [5]

เหตุที่เราใช้ $m = w-1$ ทั้ง ๆ ที่ $m = w$ นั้นสดวกกว่า เพราะเมื่อใช้ $m = w$ ตัวเลขของ X_n ทางซีกขวามือจะสุ่มน้อยกว่าตัวเลขทางซ้ายมือ

ถ้า d เป็นตัวหารของ m และ

$$Y_n = X_n \bmod d \qquad \ldots\ldots\ldots\ldots\ldots\ldots (1)$$

จะเห็นได้ว่า

$$Y_{n+1} = (a\,Y_n + c) \bmod d \qquad \ldots\ldots\ldots\ldots\ldots\ldots (2)$$

เพราะว่าถ้า

$$Y_n = X_n \bmod d$$

$$Y_{n+1} = (a\,X_n + c) \bmod d$$

$$= (a\,Y_n + c) \bmod d \qquad Y_n + X_n$$

เนื่องจาก $X_{n+1} = a\,X_n + c - g\,m$ สำหรับ integer g บางค่า เมื่อใส่ mod d เข้าทั้งสองข้างจะทำให้ $g\,m$ หายไป ถ้า d เป็นแฟคเตอร์ของ m ดังนั้น

---

4. Shan S. Kuo, Computer Applications of Numerical Methods page 337-342

5. A.J.C. Cunningham and H. J. Woodall, Factorization of $y^n \pm 1$

ตารางที่ 2

ตัวประกอบเฉพาะของ $\omega \pm 1$

PRIME FACTORIZATIONS OF $w \pm 1$

$2^e - 1$	e	$2^e + 1$
$7 \cdot 31 \cdot 151$	15	$3^2 \cdot 11 \cdot 331$
$3 \cdot 5 \cdot 17 \cdot 257$	16	65537
131071	17	$3 \cdot 43691$
$3^2 \cdot 7 \cdot 19 \cdot 73$	18	$5 \cdot 13 \cdot 37 \cdot 109$
524287	19	$3 \cdot 174763$
$3 \cdot 5^2 \cdot 11 \cdot 31 \cdot 41$	20	$17 \cdot 61681$
$7^2 \cdot 127 \cdot 337$	21	$3^2 \cdot 43 \cdot 5419$
$3 \cdot 23 \cdot 89 \cdot 683$	22	$5 \cdot 397 \cdot 2113$
$47 \cdot 178481$	23	$3 \cdot 2796203$
$3^2 \cdot 5 \cdot 7 \cdot 13 \cdot 17 \cdot 241$	24	$97 \cdot 257 \cdot 673$
$31 \cdot 601 \cdot 1801$	25	$3 \cdot 11 \cdot 251 \cdot 4051$
$3 \cdot 2731 \cdot 8191$	26	$5 \cdot 53 \cdot 157 \cdot 1613$
$7 \cdot 73 \cdot 262657$	27	$3^4 \cdot 19 \cdot 87211$
$3 \cdot 5 \cdot 29 \cdot 43 \cdot 113 \cdot 127$	28	$17 \cdot 15790321$
$233 \cdot 1103 \cdot 2089$	29	$3 \cdot 59 \cdot 3033169$
$3^3 \cdot 7 \cdot 11 \cdot 31 \cdot 151 \cdot 331$	30	$5^2 \cdot 13 \cdot 41 \cdot 61 \cdot 1321$
2147483647	31	$3 \cdot 715827883$
$3 \cdot 5 \cdot 17 \cdot 257 \cdot 65537$	32	$641 \cdot 6700417$
$7 \cdot 23 \cdot 89 \cdot 599479$	33	$3^2 \cdot 67 \cdot 683 \cdot 20857$
$3 \cdot 43691 \cdot 131071$	34	$5 \cdot 137 \cdot 953 \cdot 26317$
$31 \cdot 71 \cdot 127 \cdot 122921$	35	$3 \cdot 11 \cdot 43 \cdot 281 \cdot 86171$
$3^2 \cdot 5 \cdot 7 \cdot 13 \cdot 19 \cdot 37 \cdot 73 \cdot 109$	36	$17 \cdot 241 \cdot 433 \cdot 38737$
$223 \cdot 616318177$	37	$3 \cdot 1777 \cdot 25781083$
$3 \cdot 174763 \cdot 524287$	38	$5 \cdot 229 \cdot 457 \cdot 525313$
$7 \cdot 79 \cdot 8191 \cdot 121369$	39	$3^2 \cdot 2731 \cdot 22366891$
$3 \cdot 5^2 \cdot 11 \cdot 17 \cdot 31 \cdot 41 \cdot 61681$	40	$257 \cdot 4278255361$
$13367 \cdot 164511353$	41	$3 \cdot 83 \cdot 8831418697$
$3^2 \cdot 7^2 \cdot 43 \cdot 127 \cdot 337 \cdot 5419$	42	$5 \cdot 13 \cdot 29 \cdot 113 \cdot 1429 \cdot 14449$
$431 \cdot 9719 \cdot 2099863$	43	$3 \cdot 2932031007403$
$3 \cdot 5 \cdot 23 \cdot 89 \cdot 397 \cdot 683 \cdot 2113$	44	$17 \cdot 353 \cdot 2931542417$
$7 \cdot 31 \cdot 73 \cdot 151 \cdot 631 \cdot 23311$	45	$3^3 \cdot 11 \cdot 19 \cdot 331 \cdot 18837001$
$3 \cdot 47 \cdot 178481 \cdot 2796203$	46	$5 \cdot 277 \cdot 1013 \cdot 1657 \cdot 30269$
$2351 \cdot 4513 \cdot 13264529$	47	$3 \cdot 283 \cdot 165768537521$
$3^2 \cdot 5 \cdot 7 \cdot 13 \cdot 17 \cdot 97 \cdot 241 \cdot 257 \cdot 673$	48	$193 \cdot 65537 \cdot 22253377$
$179951 \cdot 3203431780337$	59	$3 \cdot 2833 \cdot 37171 \cdot 1824726041$
$3^2 \cdot 5^2 \cdot 7 \cdot 11 \cdot 13 \cdot 31 \cdot 41 \cdot 61 \cdot 151 \cdot 331 \cdot 1321$	60	$17 \cdot 241 \cdot 61681 \cdot 4562284561$
$7^2 \cdot 73 \cdot 127 \cdot 337 \cdot 92737 \cdot 649657$	63	$3^3 \cdot 19 \cdot 43 \cdot 5419 \cdot 77158673929$
$3 \cdot 5 \cdot 17 \cdot 257 \cdot 641 \cdot 65537 \cdot 6700417$	64	$274177 \cdot 67280421310721$

$10^e - 1$	e	$10^e + 1$
$3^3 \cdot 7 \cdot 11 \cdot 13 \cdot 37$	6	$101 \cdot 9901$
$3^2 \cdot 239 \cdot 4649$	7	$11 \cdot 909091$
$3^2 \cdot 11 \cdot 73 \cdot 101 \cdot 137$	8	$17 \cdot 5882353$
$3^4 \cdot 37 \cdot 333667$	9	$7 \cdot 11 \cdot 13 \cdot 19 \cdot 52579$
$3^2 \cdot 11 \cdot 41 \cdot 271 \cdot 9091$	10	$101 \cdot 3541 \cdot 27961$
$3^2 \cdot 21649 \cdot 513239$	11	$11^2 \cdot 23 \cdot 4093 \cdot 8779$
$3^3 \cdot 7 \cdot 11 \cdot 13 \cdot 37 \cdot 101 \cdot 9901$	12	$73 \cdot 137 \cdot 99990001$
$3^2 \cdot 11 \cdot 17 \cdot 73 \cdot 101 \cdot 137 \cdot 5882353$	16	$353 \cdot 449 \cdot 641 \cdot 1409 \cdot 69857$

$$X_{n+1} = a X_n + c - g m$$

$$(X_{n+1}) \bmod d = (a X_n + c - q m) \bmod d$$

$$= (a X_n + c) \bmod d - (qm) \bmod d$$

ถ้า d เป็นแฟคเตอร์ของ m

$\therefore (q m) \bmod d = 0$ จึงตัด $q m$ ออกไปได้

สภาพการณ์ที่ ตัวเลขของ X_n ทางขีกขวามือจะสุ่มน้อยกว่าตัวเลขทางซ้ายมือจะไม่เกิดขึ้น เมื่อ $m = w \pm 1$ เพราะในกรณีเช่นนี้ bits ที่มีลำดับต่ำของ X_n จะมีลักษณะเป็น random เท่า ๆ กับ bits ในลำดับสูง ๆ ของ X_n เช่น ถ้า $w = 2^{35}$ และ $m = 2^{35} - 1$ จำนวนชุดลำดับจะไม่ใคร่สุ่ม ถ้าเราพิจารณาแต่เพียงเศษของ mod 31, mod 71, mod 127 หรือ mod 122921 เท่านั้น (ดูตาราง 2) เพราะ $2^{35} - 1$ แยกแฟคเตอร์ได้เป็นดังนี้ $2^{35} - 1 = 31.71.127.172921$ แต่ bits ที่มีลำดับต่ำกว่านี้ (ซึ่งแสดงจำนวนเลขของลำดับที่ใช้ mod 2) จะให้ค่าที่สุ่มมากกว่า

ในการนำไปใช้ ส่วนใหญ่แล้ว bits ทางลำดับต่ำมักไม่มีนัยสำคัญ และการเลือกให้ $m = w$ จะเหมาะมากถ้าคนเขียนโปรแกรมของเครื่องคอมพิวเตอร์ใช้เลขสุ่มนี้อย่างฉลาดจริง ๆ

3.1.3 การเลือกตัว Multiplier

การเลือกตัว multiplier, a นั้น เราจะต้องมีวิธีเลือกที่ดี จึงจะได้ตัว multiplier ที่ทำให้มีความยาวคาบยาวที่สุด และตัวเลขสุ่มที่ได้มีความสุ่ม เพราะปรกติแล้วการเลือกตัว multiplier ที่จะทำให้มีชุดลำดับของการสร้างตัวเลขสุ่มยาวที่สุด คือ m นั้น เราหาได้ง่าย เช่น $a = c = 1$ เราแทนค่าลงในสมการจะได้

$$X_{n+1} = (X_n + 1) \bmod m$$

ลำดับตัวเลขที่ได้จะมีความยาวคาบเป็น m แต่ตัวเลขสุ่มที่ได้มีความเป็นสุ่มไม่สมบูรณ์เพียงพอ

ค่า m มีได้หลายค่าต่าง ๆ กัน ความยาวของคาบที่เกิดจากการสร้างก็จะต่างกันออกไปด้วยซึ่งขึ้นอยู่กับค่า m ด้วย การเลือกตัว a และ c ก็มีผลทำให้คาบที่เกิดขึ้นมีความยาว

มากที่สุดหรือไม่

ดังนั้นเราจะพิจารณาค่าที่เป็นไปได้ทั้งหมดของ a และ c ซึ่งจะทำให้คาบมีความยาว m (ข้อสังเกตุ ความยาวคาบเป็น m จำนวนต่าง ๆ ก็จะเกิดขึ้นจาก 0 ถึง m - 1 ใน 1 คาบ)

ทฤษฎีบท A[6]

ชุดลำดับ Congruential เชิงเส้นมีความยาวคาบเป็น m ได้ก็ต่อเมื่อ

I) C เป็นตัวเฉพาะเมื่อมุ่งต่อ m (หารกันได้ลงตัว)

II) b = a - 1 เป็นตัวประกอบของ P สำหรับทุกค่าของตัวเฉพาะ p ที่ใช้หาร m ได้

III) b เป็นตัวคูณประกอบของ 4 ถ้า m เป็นตัวคูณประกอบของ 4

วิธีพิสูจน์ ทฤษฎีบท A นั้น M. Greenberger ได้ทำขึ้นในกรณีที่ $m = 2^e$[7] ส่วน sufficient condition (I), (II) และ (III) ในกรณีทั่ว ๆ ไปนั้น Hull และ Dobel ได้พิสูจน์ไว้แล้ว[8] จากการพิสูจน์ทฤษฎีบท A เราพอจะสรุปเป็นทฤษฎีบท B ได้

ทฤษฎีบท B

เมื่อ C = 0 ความยาวที่สุดที่เป็นไปได้ เราจะใช้สัญลักษณ์ (m)

(m) จะมีความยาวคาบยาวที่สุดก็ต่อเมื่อ

I) X_o คือ ตัวเฉพาะที่มุ่งต่อ m

II) a คือ primitive element modulo m

สังเกตุได้ว่า เราจะได้คาบยาว m - 1 ถ้า m คือตัวเฉพาะ (prime) ซึ่งเรา

---

6. Donald E. Knuth, The Art of Computer Programming, Vol page 15

7. JACM 8 (1961) Page 383 - 389

8. SIAM REVIEW 4 (1962) page 230 - 254

ได้คาบที่มีความยาวน้อยกว่าความยาวมากที่สุดเพียง 1

ปัญหาจึงขึ้นอยู่กับการหาค่าของ primitive element modulo m ซึ่งเราหาค่าได้โดยอาศัยความจริง ตามทฤษฎีบท c ที่จะกล่าวต่อไป

ทฤษฎีบท C

a จะเป็น primitive element modolo p^e ก็ต่อเมื่อ

I) $p^e = 2$ a คือเลขคี่ หรือ $p^e = 4$, a mod 4 = 3

หรือ $p^e = 8$, a mod 8 = 3, 5, 7

หรือ $p = 2$, $e \geq 4$, a mod 8=3 หรือ 5 หรือ

II) p คือเลขคี่ $e = 1$, $a \not\equiv 0$ (modolo p)

และ $a^{(p-1)/g} \not\equiv 1$ (modolo p) สำหรับค่าใด ๆ ของตัวหาร

เฉพาะ q ของ p - 1 หรือ

III) p คือเลขคี่, $e \geq 1$, a เหมาะสมกับ II และ

$a^{p-1} \not\equiv 1$ (modolo p^2)

เงื่อนไขข้อที่ II และ III ของทฤษฎีบทนี้สามารถทดสอบโดยอาศัยคอมพิวเตอร์ สำหรับค่าของ p ที่มีค่ามาก การใช้คอมพิวเตอร์ช่วยในการทดสอบ เราอาจใช้วิธีวิเคราะห์ powers ที่สามารถวิเคราะห์ได้อย่างมีประสิทธิภาพ คือ Evaluating of Power method[9] ซึ่งวิธีนี้เราสามารถคำนวณหาค่า X^n ได้ง่ายและเหมาะกับการนำมาใช้กับเครื่องคอมพิวเตอร์

ในที่สุด ถ้ากำหนดให้ a_j เป็น Primitive modulo $p_j^{e_j}$ เราสามารถคำนวณหาค่าเพียงค่าเดียวของ a ซึ่ง

$$a = a_j \text{ (modulo } p_j^{e_j}) \quad i \leq j \leq t,$$

โดยอาศัย "Chinese remainder algorithm[10]

---

9. Donald E. Knuth, The Art of Computer Programming, Vol 2 page 398 - 401

10. Ibid p. 249 - 254

ดังนั้น a จะเป็น primitive modulo $p_1^{e_1} \ldots\ldots\ldots p_t^{e_t}$ นี่เป็นวิธีที่จะช่วยให้เราสามารถสร้าง multipliers ที่เหมาะสมกับกรณีของทฤษฎีบท B สำหรับค่า m ที่กำหนดขึ้น แต่ในกรณีทั่ว ๆ ไปแล้วทำอย่างไร เราจึงจะสามารถคำนวณหาความยาวได้

ในกรณีที่ $m = 2^e, e \geq 4$ จากทฤษฎีบท C จะเห็นได้ว่า $a = 3$ หรือ 5 (modulo 8) ในกรณีนี้ $\frac{1}{4}$ ของ multi iers ที่เป็นไปได้จะให้คาบที่ยาวที่สุด

ทฤษฎีบท D[11]

ถ้า $m = 2^e$, $e \geq 5$, $C=0$ และ X_o ไม่ใช่ตัวประกอบของ 2 หรือ 5 คาบของ Congruence Sequence เชิงเส้น คือ $5 \times 10^{e-2}$ ก็ต่อเมื่อ a mod 200 เท่ากับตัวใดตัวหนึ่งของตัวเลข 32 ตัวข้างล่างนี้

3	11	13	19	21	27	29	37	53	59
61	67	69	77	83	91	109	117	123	131
133	139	141	147	163	171	173	179	181	187
189	197								

โดยทั่วไปแล้ว การเลือกตัว multiplier มีวิธีเลือกที่เป็นอิสระจากกันมาก แต่การเลือกตัว multiplier เราต้องระวังหลีกเลี่ยงแบบง่าย ๆ ของ multiplier digits

---

11. Donald E. Knuth, The Art of Computer Programming Vol 2. page 19

3.2 วิธีดำเนินการวิจัย

จากการศึกษาวิธีสร้างตัวเลขสุ่ม และการเลือกค่าต่าง ๆ พบว่า

3.2.1 วิธีสร้างตัวเลขสุ่มที่มีใช้กันอยู่ มีประมาณ 4 วิธี คือ

1. วิธี Middle - square
2. วิธี Power - Residue
3. วิธี Multiplicative
4. วิธี Mixed

3.2.2 ในวิธีการต่าง ๆ เหล่านี้ วิธี Multiplicative เป็นวิธีที่ดีที่สุด

3.2.3 ปัญหาในเรื่อง Multiplicative ที่จะต้องคำนึงถึง คือ

1. ค่าเริ่มต้น
2. ค่าตัว Modulus
3. ค่าตัวคูณ (Multiplier)

แต่ค่าตัว Multiplier เป็นค่าที่เราควรต้องสนใจมากที่สุด และได้มีผู้กำหนดหลักในการหาค่า Multiplier (a) ไว้ดังได้กล่าวไว้แล้ว

3.2.4 ค่า Multiplier (a) ที่หาได้ตามกฎการหาค่า Multiplier นั้น ก็ยังไม่แน่ว่าค่าตัว a ค่าใดที่ให้ผลดีที่สุด โดยเฉพาะอย่างยิ่ง เมื่อนำมาใช้กับเครื่อง NEAC 2200/200 ซึ่งมีขนาดหน่วยความจำ 24 K Characters

3.2.5 เพื่อที่จะหาค่า a ที่ดีที่สุด เราจึงเลือกค่า a เป็นไปตามกฎการเลือกค่า a จำนวน 10 ค่า ดังนี้

a	ค่าที่เลือก
1	32761
2	32763
3	32765
4	32767
5	32769
6	32771
7	32773
8	32775
9	32777
10	32779

3.2.6 เขียนโปรแกรมให้เครื่องคอมพิวเตอร์สร้างตัวเลขสุ่ม โดยกำหนดค่าต่าง ๆ คือ

ค่าเริ่มต้น (Initial value) = 89745369

ค่า Modulus = 2^{30} = 1073741824

ค่า Multiplier 10 ค่า ต่าง ๆ กัน

เราจะได้เลขสุ่มออกมา 10 ชุด โดยเราให้เครื่องสร้างตัวเลขแต่ละค่ามี 8 หลัก จำนวน 10,000 ค่า

3.2.7 ทำการทดสอบการกระจายแบบคงตัว

1. ทดสอบการกระจายแบบคงตัว (Uniform)
2. ทำการทดสอบ Series

3.2.8 รวบรวมผลการทดสอบ แสดงเป็นตารางและกราฟเพื่อหาค่า a ที่ดีที่สุด ที่จะทำให้ได้ชุดตัวเลขสุ่มที่สุ่มมากที่สุด