

ภาคผนวก ก.
จีเนติกอัลกอริธึม

จีเนติกอัลกอริทึม

จุดเริ่มแรกของพันธุศาสตร์เริ่มขึ้นเมื่อปี ค.ศ. 1859 เมื่อ ชาร์ล ดาร์วิน (Charles Darwin) ได้เสนอทฤษฎีซึ่งทำให้เกิดข้อถกเถียงอย่างมากในชื่อหนังสือ “พื้นฐานของสิ่งมีชีวิตโดยการคัดสรรทางธรรมชาติหรือการปรับตัวเพื่อรักษาเผ่าพันธุ์” (On the origin of species by means of natural selection, or the preservation of favored races in the struggle for life) หรือที่รู้จักกันในชื่อ “พื้นฐานของสิ่งมีชีวิต” (The origin of species) ในหนังสือนั้นได้นำเสนอการค้นคว้าของเขาโดยเปรียบเทียบระหว่างสิ่งมีชีวิตสายพันธุ์เดียวกันแต่ถิ่นที่อยู่อาศัยคนละที่ มีผลให้เกิดการปรับตัวให้เหมาะสมกับการดำรงชีวิต

และในเวลาใกล้เคียงกัน เกออร์ก เม็นเดล (Gregor Mendel) ได้ทำการค้นคว้าในเรื่องของลักษณะเฉพาะของตัวบุคคล โดยได้ทำการทดลองกับต้นถั่ว และได้ค้นพบยีน (Gene) แต่การค้นพบของไมเมนเดลนั้นไม่เป็นที่แพร่หลายและไม่มีคนสนใจมากนัก แต่การค้นพบของเมนเดลนี้ก็เป็พื้นฐานสำหรับการศึกษาพันธุศาสตร์ในเวลาต่อมา ภายหลังเมนเดลเสียชีวิตไปประมาณ 30 ปี วอลเตอร์ ซัทตัน (Walter Sutton) ทำการทดลองและค้นพบว่ายีนนั้นเป็นส่วนประกอบของโครโมโซม (Chromosomes) ซึ่งโครโมโซมนั้นจะอยู่ในนิวเคลียส (Nucleus) อีกทีหนึ่ง โดยโครโมโซมนั้นจะเป็นตัวกำหนดรูปแบบลักษณะของสิ่งมีชีวิต

โครโมโซมนั้นประกอบด้วยชิ้นเป็น DNA (Deoxyribonucleic Acid) โดยพันธะเคมีของ DNA นั้นจะเรียกว่านิวคลีโอไทด์ (Nucleotides) โดยประกอบขึ้นจากสาร 4 ชนิด อันได้แก่ อะดีนีน (Adenine : A) , กวานีน (Guanine : G) , ไซโตซีน (Cytosine : C) และ ไทมีน (thymine : T) โดยกระบวนการจับคู่กันแบบพันธะทางเคมีพบว่า การจับคู่กันของนิวคลีโอไทด์ 3 ตัวเรียกว่า โคดอน (Codon) และการที่โคดอนหลายๆตัวจับรวมกันเป็นสายเราเรียกว่า สายของโคดอนหรือยีน ขึ้นต่อไปของกระบวนการคือการจับรวมกันเป็นสายของยีนเพื่อถ่ายทอดลักษณะทางพันธุกรรมที่เราเรียกว่า DNA

จีเนติกอัลกอริทึม นั้นถูกคิดค้นโดยจอห์น ฮอลแลนด์ (John Holland) ซึ่งสนใจในทฤษฎีวิวัฒนาการทางธรรมชาติในการกำเนิดประชากรของสิ่งมีชีวิตรุ่นต่อไป ซึ่งประกอบด้วยการคัดเลือกทางธรรมชาติ คือสิ่งมีชีวิตใดมีความแข็งแกร่งกว่าย่อมมีโอกาสในการอยู่รอดมากกว่า หรืออาจกล่าวอีกนัยหนึ่งว่า ยีนใดๆซึ่งประกอบด้วยชิ้นจากโครโมโซมที่มีลักษณะดีกว่ายีนอื่นๆ ย่อมมีโอกาสอยู่รอดสูงกว่า และยีนที่สามารถอยู่รอดได้ก็จะสามารถถ่ายทอดลักษณะเด่นไปยังลูกหลานได้ จากที่กล่าวมานี้ จอห์น ฮอลแลนด์และคณะผู้ร่วมงานในมหาวิทยาลัยมิชิแกนได้ทดลองนำมาใช้กับการคำนวณทางคณิตศาสตร์จนเกิดเป็น จีเนติก อัลกอริทึม เมื่อปี ค.ศ. 1960 โดยเริ่มแรกนั้นถูกออกแบบให้เป็นแบบจำลองที่ใช้อธิบายการเปลี่ยนแปลงของธรรมชาติ (Computer Base Model for Natural Genetic) ซึ่งใช้ในทางชีววิทยา (Biological) ทุกวันนี้จีเนติกอัลกอริทึมกลายเป็นเทคนิคทาง

คณิตศาสตร์มากกว่าที่จะเป็นแบบจำลองทางชีววิทยา ต่อมาในปี ค.ศ. 1995 กอสรี่และเยน (Ghoshry & Yen) ได้เสนอวิธีที่มีประสิทธิภาพมากขึ้นเรียกว่า MGA (Modified Genetic Algorithm) ซึ่งได้รับการพัฒนามาจากการผสมกันระหว่าง จีเนติกอัลกอริธึมแบบปกติ (Simple Genetic Algorithm : SGA) และ SA (Simulated Annealing) ขั้นตอนการคำนวณที่ได้เสนอมานั้นทั้งหมดเป็นการทำงานแบบวิวัฒนาการพันธุศาสตร์ แต่ได้พัฒนาเป็นฟังก์ชันชุดใหม่ ซึ่งใช้หลักของ SA ผลที่ได้แสดงให้เห็นว่า MGA มีประสิทธิภาพในการค้นหาคำตอบที่ดีที่สุดในรูปแบบต่าง ๆ ด้วยเวลาน้อยลงเมื่อเปรียบเทียบกับวิธีเดิม และผลที่ได้มีความถูกต้อง นอกจากนี้ ผลที่ได้ยังแสดงให้เห็นว่า MGA มีความคลุมเครือลดลง ต่อมาในปี ค.ศ. 1997 ชิโมไดระ (Shimodaira) ได้นำเสนอวิธีเชิงพันธุศาสตร์แบบใหม่ เรียกว่า DCGA (Diversity Control oriented Genetic Algorithm) ซึ่งสามารถป้องกันการเกิดปัญหาการจับกลุ่มของผลลัพธ์ (Local Optimum Solution) โดยโครงสร้างประชากรรุ่นใหม่ในวิธี DCGA นั้นได้รับการคัดเลือกจากการรวมกันของประชากรในรุ่นก่อนและรุ่นต่อมาด้วยวิธีที่เรียกว่า CPSS (Cross-generational Probabilistic Survival Selection) ซึ่งเป็นการคำนวณโดยใช้ระยะทาง ระหว่างโครงสร้างใหม่และโครงสร้างที่ดีที่สุด พบว่าการทำงานของ DCGA ดีกว่าวิธีเชิงพันธุศาสตร์แบบธรรมดาตรงในการหาผลลัพธ์ที่ดีที่สุด (Global Optimum Solution) แต่ว่าใช้เวลาในการคำนวณในขั้นตอนการคัดเลือกมากกว่า

ก.1 ตัวดำเนินการทางจีเนติกอัลกอริธึม (Operator of genetic algorithms)

แบบจำลองทางพันธุกรรมที่นำมาใช้ในจีเนติกอัลกอริธึมถูกเรียกว่าโครโมโซม (chromosome) ในโครโมโซมประกอบด้วยบิตจำนวน n บิต ค่าของบิตในของแต่ละโครโมโซมนิยามให้มี 3 แบบคือ

1. บิต 0 หรือบิต 1 (binary coding)
2. บิตจำนวนเต็ม (integer coding)
3. บิตจำนวนจริง (real number coding)

ค่าบิตในแต่ละแบบมีข้อดีข้อเสียที่แตกต่างกัน การเลือกใช้ขึ้นอยู่กับลักษณะของงานค่าบิต 0 หรือบิต 1 เป็นทฤษฎีที่ยอมรับและถูกออกแบบมานานแล้ว ข้อดีแบบบิต 0 หรือ บิต 1 คือลดขนาดช่วงการหาค่าที่เหมาะสม (optimization space) ลงแต่การใช้งานในบางอย่างทำให้เกิดการสูญเสียค่าความละเอียดของผลลัพธ์ที่ได้ ซึ่งถ้าค่าบิตเป็นจำนวนเต็มหรือจำนวนจริง ขนาดช่วงการหาค่าที่เหมาะสม (optimization space) จะมีขนาดใหญ่กว่าและมีความละเอียดของผลลัพธ์ที่ได้มากกว่า

บิตในโครโมโซมของจีเนติกอัลกอริธึมนั้นคือค่าของชุดตัวแปรที่จะนำไปทดลองแทนในสมการเพื่อดูผลลัพธ์ที่ได้ โดยนิยามว่าถูกเรียงกันอยู่ในแนวนอน

ความมีประสิทธิภาพของจีเนติกอัลกอริทึมขึ้นอยู่กับตัวดำเนินการทางจีเนติกอัลกอริทึมซึ่งตัวดำเนินการจีเนติกอัลกอริทึมประกอบด้วย

1. การคัดเลือก
2. การสร้างโครโมโซมลูก
3. การกลายพันธุ์
4. ปัจจัยที่มีผลต่อการทำงานของจีเนติกอัลกอริทึม

ก.2 การคัดเลือก (reproduction or selection)

การคัดเลือกเป็นกระบวนการที่จะตัดสินใจว่าโครโมโซมใดจะคงอยู่ในรุ่น (generation) เพื่อนำไปคัดเลือกต่อไป โดยจะตัดสินใจจากความเหมาะสมของแต่ละโครโมโซมโดยค่าความเหมาะสมของแต่ละโครโมโซมนั้นหาได้จากการฟังก์ชันความเหมาะสม โดยโครโมโซมที่มีค่าตัวแปรที่ทำให้เกิดผลลัพธ์ได้ใกล้เคียงกับฟังก์ชันความเหมาะสมมาก ก็จะมีค่าความเหมาะสมสูง และมีโอกาสที่จะถูกคัดเลือกสูง ในบางกรณีฟังก์ชันเป้าหมายและฟังก์ชันความเหมาะสมอาจเป็นฟังก์ชันเดียวกัน โดยโครโมโซมที่ถูกคัดเลือกนั้นแทนด้วย $P_{select(i)}$ หาได้จากสมการที่ (ก.1)

$$P_{select(i)} = \frac{f_i}{\sum_{i=1}^k f_i} \quad (ก.1)$$

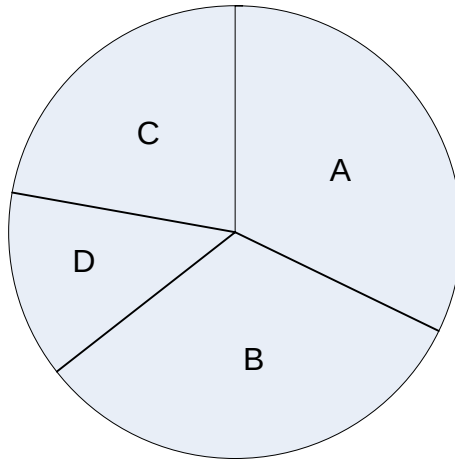
โดยที่ f_i คือค่าความเหมาะสมของแต่ละโครโมโซม

ส่วนจำนวนโครโมโซมที่ถูกคัดเลือก (num_i) เพื่อนำไปจับคู่และสร้างโครโมโซมลูกสามารถหาได้จากสมการที่ (ก.2)

$$num_i = N * P_{select(i)} \quad (ก.2)$$

โดยที่ N คือจำนวนโครโมโซมทั้งหมด (population size)

กระบวนการที่ตัดสินใจว่าโครโมโซมใดจะยังอยู่เพื่อนำไปคัดเลือกในรุ่นต่อไปโดยตัดสินใจจากความน่าจะเป็นของแต่ละโครโมโซม ถ้าเป็นแบบไบนารีจะใช้เทคนิควงล้อรูเล็ตและบิตเป็นจำนวนจริงหรือจำนวนเต็มจะใช้ normalized geometric การคัดเลือกโครโมโซมโดยเทคนิควงล้อรูเล็ตแสดงดังรูปที่ ก.1



รูปที่ ก.1 การคัดเลือกโครโมโซมโดยเทคนิควงล้อรูเล็ต

เทคนิคของวงล้อรูเล็ตนี้เป็นวิธีการคัดเลือกตามอัตราส่วน โดยพิจารณาค่าความเหมาะสมของโครโมโซมแต่ละตัว โดยจะทำการแบ่งพื้นที่วงล้อตามสัดส่วนของค่าความเหมาะสมของโครโมโซมแต่ละตัว โดยให้ผลรวมของวงล้อรูเล็ตคือผลรวมของค่าความเหมาะสมของโครโมโซมทั้งหมด จากนั้นจะทำการสุ่มค่าตั้งแต่ 0 ถึง ค่าผลรวมความเหมาะสม ถ้าตัวเลขที่สุ่มไปหยุดบนพื้นที่ของโครโมโซมใด โครโมโซมตัวนั้นจะถูกเลือก ดังนั้น โครโมโซมที่มีค่าความเหมาะสมสูงย่อมมีโอกาสถูกเลือกมากกว่าโครโมโซมที่มีค่าความเหมาะสมต่ำ

ตัวอย่างการทำงานของวงล้อรูเล็ต สมมติให้โครโมโซมที่เข้าสู่กระบวนการคัดเลือกมี 4 โครโมโซม อันได้แก่ A, B, C และ D ดังรูปที่ ก.1 โดยโครโมโซมแต่ละโครโมโซมมีค่าความเหมาะสมดังนี้

$$A = 0.9926, \quad B = 0.9214, \quad C = 0.7352 \text{ และ } D = 0.3463$$

ผลรวมของค่าความเหมาะสมทั้งหมด (Sum of Fitness) = 2.9955 เมื่อนำมาจัดอัตราส่วนแล้ว ค่า A จะมีพื้นที่ในวงกลม 33% ค่าภายในพื้นที่ตั้งแต่ 0 ถึง 0.9926, ค่า B จะมีพื้นที่ในวงกลม 30.75% ค่าภายในพื้นที่ตั้งแต่ 0.9927 ถึง 1.914, ค่า D จะมีพื้นที่ในวงกลม 11.56% ค่าภายในพื้นที่ตั้งแต่ 1.915 ถึง 2.2603 และ ค่า D จะมีพื้นที่ในวงกลม 24.54% ค่าภายในพื้นที่ตั้งแต่ 2.2603 ถึง 2.9955 จากนั้นจะทำการสุ่มค่าคงที่ขึ้นมา 1 ค่า โดยค่าคงที่นั้นมีค่าอยู่ระหว่างอยู่ระหว่าง 0 ถึง 2.9955 ถ้าค่าคงที่ที่สุ่มขึ้นมามีค่าเท่ากับค่าภายในพื้นที่ของโครโมโซมใด โครโมโซมนั้นจะถูกเลือก

ในกระบวนการของการคัดเลือกนี้บางครั้งอาจมีวิธีการบางส่วนที่แตกต่างกันไป เช่น จำนวนโครโมโซมที่ถูกการคัดเลือก (num_i) ซึ่งในงานวิจัยของเรนดี้ (Randy L.Haupt) ได้ใช้วิธีการคัดเอา 50% ของจำนวนโครโมโซมจากโครโมโซมที่มีค่าฟังก์ชันความเหมาะสมที่ดีที่สุดของทั้งหมดมาเป็นโครโมโซมพ่อแม่ ส่วนงานวิจัยของโคจิ,ยาซุฮิซะและโทชิโอะ (Koji Shimojima ,

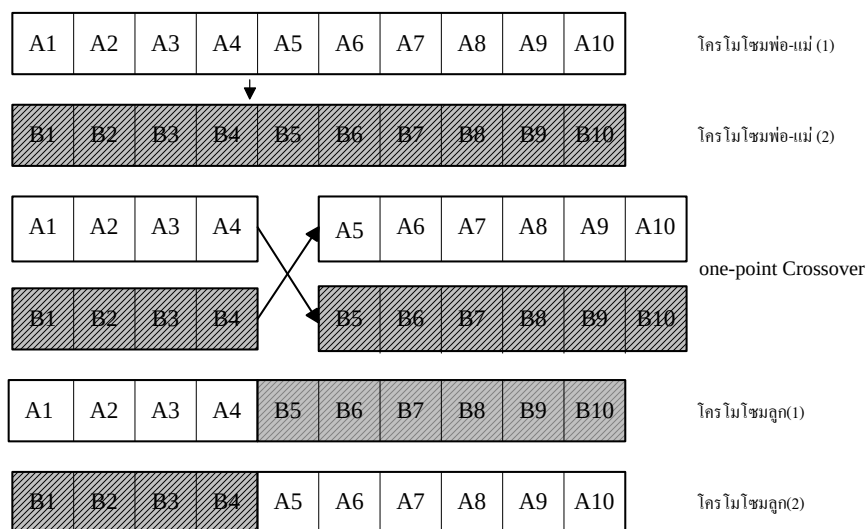
Yasuhisa Hasegawa & Toshio Fukuda) โดยใช้วิธีการคัดเอาโครโมโซมที่มีค่าฟังก์ชันสูงกว่าค่าฟังก์ชันที่กำหนดไว้เท่านั้น แล้วจึงทำการสุ่มคัดเอาเพื่อเป็นโครโมโซมพ่อแม่เป็นต้น

ก.3 การสร้างโครโมโซมลูก (crossover)

กระบวนการนี้เป็นกระบวนการพื้นฐานขั้นต่อมาของจีเนติก อัลกอริทึม หลังจากกระบวนการคัดเลือกได้โครโมโซม พ่อ-แม่แล้วขั้นตอนต่อมาคือการสร้างโครโมโซมลูก หรือการ (crossover) โดยโครโมโซมลูกที่เกิดขึ้นนั้นจะมีลักษณะของโครโมโซมพ่อและแม่ปะปนกันอยู่ กระบวนการขั้นต่อไปซึ่งโดยทั่วไปมี 3 วิธี อันได้แก่ one-point crossover, two-point crossover และ uniform crossover

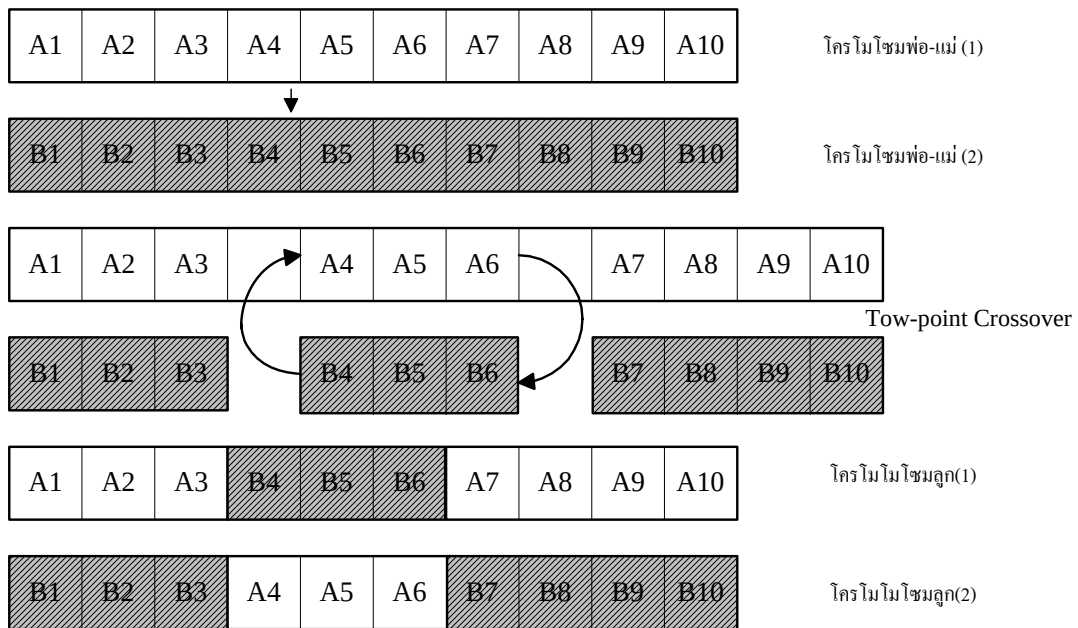
วิธีของ one-point crossover คือการสุ่มค่าตั้งแต่ 1 ถึง $n-1$ เพื่อหาตำแหน่งในการ crossover โดยที่ n คือค่าจำนวนบิตทั้งหมดในโครโมโซมดังตัวอย่างรูปที่ ก.2

จากรูปที่ 3-9 โครโมโซม พ่อ-แม่ (1) และโครโมโซม พ่อ-แม่ (2) นำมาสร้างโครโมโซมลูกแบบ one-point crossover โดยมีตำแหน่งสำหรับการครอสโอเวอร์ที่สุ่มได้คือ 4 ทำให้โครโมโซมลูก 2 โครโมโซมที่มีการสลับกันของโครโมโซม พ่อ-แม่ ณ ตำแหน่งบิตที่ 4



รูปที่ ก.2 การสร้างโครโมโซมลูกแบบ one-point crossover

วิธี two-point crossover คือการสุ่มค่า 2 ค่า ตั้งแต่ 1 ถึง $n-1$ เพื่อดำเนินการ 2 ตำแหน่งในการครอสโอเวอร์แล้วทำการสลับบิต โดยการเอาบิตที่อยู่ระหว่างค่าตำแหน่งบิตที่สุ่มได้ 2 ตำแหน่งมาสลับกันระหว่างโครโมโซม พ่อ-แม่ (1) กับโครโมโซม พ่อ-แม่ (2) เพื่อให้ได้โครโมโซมลูก 2 ดังรูปที่ ก.3

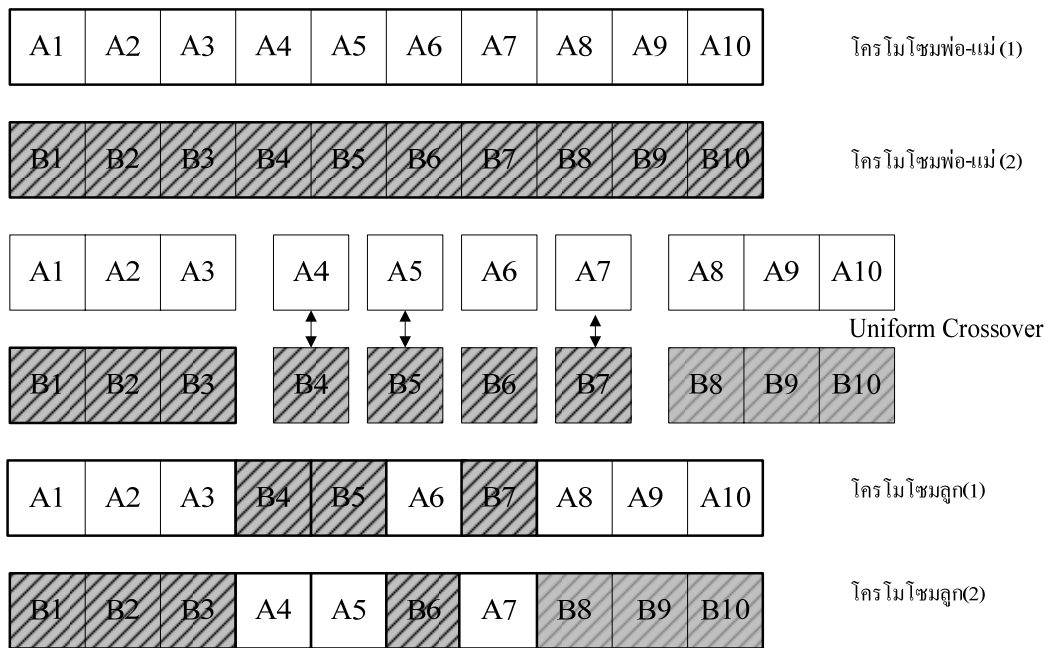


รูปที่ ก.3 การสร้างโครโมโซมลูกแบบ two-point crossover

จากตัวอย่างรูปที่ ก.3 เป็นการสร้างโครโมโซมลูกแบบ two-point crossover ค่าตำแหน่งบิตที่สุ่มได้คือ 3 และ 6 โดยการสลับค่าบิตที่อยู่ระหว่าง 3 กับ 6 ของโครโมโซม พ่อ-แม่ ทั้ง 2 โครโมโซมจะได้โครโมโซมลูก 2 โครโมโซม

วิธี uniform crossover คือกำหนดจำนวนบิตที่ต้องการแลกเปลี่ยนกันระหว่างโครโมโซม พ่อ-แม่ (1) กับโครโมโซม พ่อ-แม่ (2) แล้วสุ่มจำนวนบิตที่ได้โดยทั่วไปจำนวนบิตที่แลกเปลี่ยนกันจะมีค่าอยู่ระหว่าง 0% ถึง 5 % ดังตัวอย่างรูปที่ ก.4

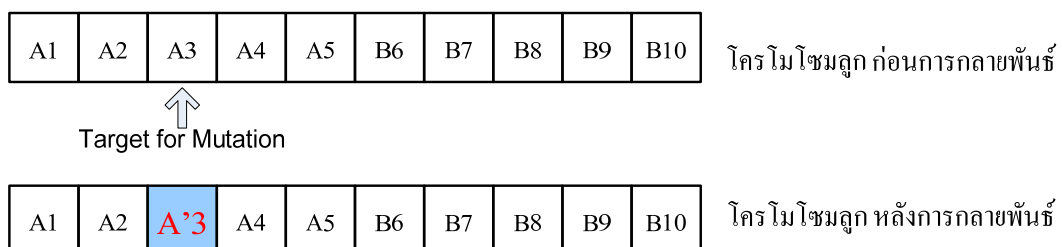
จากรูปที่ ก.4 กำหนดให้จำนวนบิตที่ต้องการแลกเปลี่ยนมีค่าเท่ากับ 30% ดังนั้น 30% ของจำนวนบิตทั้งหมดคือ 10 บิต มีค่าเท่ากับ 3 บิต แล้วสุ่มตำแหน่งบิตที่ 3 ตำแหน่งโดยมีค่าที่สุ่มได้คือ 4, 5, 7 ดังนั้น โครโมโซมลูกที่ได้จะมีการแลกเปลี่ยนบิตกันระหว่างบิตที่ 4, 5, 7 ของโครโมโซม พ่อ-แม่ (1) กับโครโมโซม พ่อ-แม่ (2)



รูปที่ ก.4 การสร้างโครโมโซมลูกแบบ uniform crossover

ก.4 การกลายพันธุ์ (mutation)

ในธรรมชาติ กระบวนการกลายพันธุ์นั้นเกิดจากการที่สาร 4 สารที่ประกอบตัวขึ้นเป็น DNA นั้น เกิดการแทนที่กันผิดตำแหน่ง ทำให้เกิดโครงสร้างทางพันธุกรรมแบบใหม่ขึ้นมา ซึ่งสายพันธุ์ใหม่ที่เกิดขึ้นนี้อาจจะดีขึ้นหรือแย่ลงก็ได้ และในจีเนติกอัลกอริทึม จะกำหนดให้มีกระบวนการกลายพันธุ์ขึ้นมา เพื่อเพิ่มความหลากหลายของโครโมโซม แต่จะกำหนดอัตราของการกลายพันธุ์ไว้ต่ำ โดยมีวิธีการคือ ทำการสุ่มโครโมโซมลูกแล้วสุ่มตำแหน่งบิตในโครโมโซมนั้นเพื่อเปลี่ยนค่าของบิตในโครโมโซมดังกล่าวเนื่องจากโครโมโซมลูกที่ได้จากการครอสโอเวอร์ของโครโมโซม พ่อ-แม่นั้น มีส่วนประกอบของพันธุกรรมที่ไม่มีการเปลี่ยนแปลง การกลายพันธุ์จึงสามารถป้องกันข้อบกพร่องดังกล่าวได้ วิธีการกลายพันธุ์ของโครโมโซมลูกแสดงตัวอย่างดังรูป ก.5



รูปที่ ก.5 การกลายพันธุ์(mutation)

จากรูปที่ ก.5 ตำแหน่งที่ถูกสุ่มให้มีการกลายพันธุ์คือตำแหน่งที่ 3 โดยมีค่าบิตเปลี่ยนจาก A3 เป็น A'3 โดยที่ค่าของบิตโครโมโซมที่ใช้เป็นแบบบิต 0 หรือ 1 ค่าของ B'3เปลี่ยนจาก 1 เป็น 0 หรือ 0 เป็น 1 ทั้งนี้ขึ้นอยู่กับค่าของ A3 แต่สำหรับค่าบิตที่เป็นแบบจำนวนเต็มหรือจำนวนจริง ค่าของ A'3 จะเป็นค่าที่เกิดจากการสุ่มค่าจำนวนเต็มสำหรับโครโมโซมมีค่าบิตเป็นจำนวนเต็มและจำนวนจริงสำหรับโครโมโซมที่มีค่าบิตเป็นจำนวนจริง

ก.5 ปัจจัยที่มีผลต่อการทำงานของจินตિકอัลกอริธึม

ในการใช้จินตિકอัลกอริธึม ซึ่งประกอบด้วยตัวดำเนินการต่างๆตามที่ได้อธิบายมาแล้ว ตัวดำเนินการที่ถือว่าเป็นตัวดำเนินการหลักของจินตિકอัลกอริธึมได้แก่ตัวดำเนินการครอสโอเวอร์ และตัวดำเนินการกลายพันธุ์ เนื่องจากการครอสโอเวอร์ เป็นการสร้างโครโมโซมรุ่นใหม่ โดยจากโครโมโซมที่ถูกคัดเลือกแล้วส่วนการกลายพันธุ์เป็นการเพิ่มโอกาสในเปลี่ยนค่าของบิต ดังนั้นปัจจัยที่เป็นตัวกำหนดผลการทำงานของจินตિકอัลกอริธึมได้แก่จำนวนโครโมโซมทั้งหมด (population size, $Popsiz$), ความน่าจะเป็นจากการสร้างโครโมโซมลูก (crossover rate or crossover probability, P_c) และ ค่าความน่าจะเป็นของการกลายพันธุ์ (mutation rate or mutation probability, P_m)

ก.5.1 จำนวนโครโมโซมทั้งหมด (population size, $Popsiz$)

จำนวนโครโมโซมหรือ $Popsiz$ คือค่าจำนวนโครโมโซมทั้งหมดที่กำหนดขึ้นในกระบวนการของจินตિકอัลกอริธึมยังมีค่า $Popsiz$ มากจะทำให้มีตัวเลือกมากขึ้นด้วยแต่ทั้งนี้การกำหนดค่า $Popsiz$ ขึ้นอยู่กับค่าที่กำหนดผู้ใช้เองหรือความซับซ้อนของงานที่จะนำมาค้นหาคำตอบ (optimization)

ก.5.2 ค่าความน่าจะเป็นของการสร้างโครโมโซมลูก (crossover probability, P_c)

ค่าความน่าจะเป็นของการสร้างโครโมโซมลูกหรือค่า P_c คือค่าความน่าจะเป็นที่จะทำให้เกิดการครอสโอเวอร์ของโครโมโซม พ่อ-แม่ โดยทั่วไปจะกำหนดให้ P_c มีค่าอยู่ในช่วง 0.75 ถึง 0.95 ได้มีการเสนอแนะการหาจำนวนโครโมโซมที่จะถูกครอสโอเวอร์ ไว้ดังสมการที่ (ก.3)

$$N_x = P_c * Popsiz \quad (ก.3)$$

โดยที่	N_x	คือจำนวนโครโมโซมที่จะถูกครอสโอเวอร์
	$Popsiz$	คือจำนวนโครโมโซมทั้งหมด

หรืออีกวิธีหนึ่งของการคัดเลือกโครโมโซมที่ใช้ค่า P_c เป็นแนวทางโดยในแต่ละโครโมโซมจะสุ่มค่า r โดยที่ r มีค่าอยู่ระหว่าง 0 ถึง 1 โดยมีเงื่อนไขว่า $r < P_c$ โครโมโซมนั้นจะถูกคัดออกแล้วทำการจับคู่โครโมโซมแบบสุ่มเพื่อทำการคัดออก

ก.5.3 ค่าความน่าจะเป็นของการกลายพันธุ์ (mutation probability, P_m)

ค่าความน่าจะเป็นของการกลายพันธุ์หรือค่า P_m เป็นปัจจัยอีกค่าหนึ่งที่ทำให้เกิดการกลายพันธุ์ซึ่งทุกบิตในโครโมโซมจะถูกสุ่มค่า r โดยที่ r มีค่าอยู่ระหว่าง 0 ถึง 1 โดยมีเงื่อนไขว่า $r < P_m$ โครโมโซมนั้นจะถูกกลายพันธุ์โดยทั่วไปจะกำหนดให้ P_m มีค่าอยู่ในช่วง 0.001 ถึง 0.01 และค่า P_m มักจะกำหนดให้มีค่าน้อยๆเพื่อให้โครโมโซมที่ดีมีโอกาสถูกทำลายน้อยลง Dejong ได้เสนอแนะว่าค่า P_m ควรมีค่าแปรผกผันกับ $Popsiz$ e และ H essner และ Manner ได้เสนอแนวทางการหาค่า P_m โดยประมาณดังสมการที่ (ก.4) เพื่อใช้เป็นแนวทางในการเลือกค่า P_m

$$P_m = \frac{1}{Popsiz \times \sqrt{m}} \quad (ก.4)$$

โดยที่ m คือจำนวนบิตใน 1 โครโมโซม

นอกจากนี้ยังมีข้อเสนอแนะในจำนวนบิตที่จะถูกกลายพันธุ์ดังสมการที่ (ก.5)

$$N_m = P_m \times m \times Popsiz \quad (ก.5)$$

โดยที่ N_m คือจำนวนบิตทั้งหมดที่ถูกกลายพันธุ์ m คือจำนวนบิตใน 1 โครโมโซม

เมื่อได้ค่า P_m แล้วในแต่ละโครโมโซมจะถูกสุ่มค่า r โดยที่ r มีค่าอยู่ระหว่าง 0 ถึง 1 โดยมีเงื่อนไขว่าถ้า $r < P_m$ โครโมโซมนั้นจะถูกเลือกให้มีบิตที่ถูกกลายพันธุ์

จากที่กล่าวมาแล้วว่า ตัวดำเนินการคัดออกเป็นตัวดำเนินการหลักของจีเนติกอัลกอริทึมการคัดออกจึงถือเป็นกระบวนการหลักที่ทำให้จีเนติกอัลกอริทึมเป็นวิธีการหาค่าที่เหมาะสมที่มีประสิทธิภาพวิธีหนึ่ง ดังนั้น ค่า P_c จึงมักถูกกำหนดให้มีค่าสูงแต่วิธีการคัดออกจะนำเอาโครโมโซมพ่อ-แม่ ที่อยู่ติดกันมาคัดออกหรือจะกระทำโดยการสุ่มเอาโครโมโซมพ่อ-แม่มาคัดออก ก็ได้และเช่นเดียวกันกับการกลายพันธุ์ ผู้ใช้อาจสุ่มเอาโครโมโซมลูกขึ้นมาเพื่อทำการกลายพันธุ์ หรือสุ่มจากค่า r ของแต่ละโครโมโซมเพื่อเปรียบเทียบกับค่า P_m ทั้งนี้ขึ้นอยู่กับข้อกำหนดผู้ใช้เอง L.Haupt ได้เสนอแนะการใช้จีเนติกอัลกอริทึมเพื่อให้เกิดการเข้าสู่คำตอบ (algorithms convergence) อย่างได้ผลมากขึ้นโดยได้มีข้อเสนอแนะดังนี้คือ

1. เพิ่มจำนวนการกลายพันธุ์
2. เพิ่มจำนวนโครโมโซม ($Popsiz$ e)
3. เพิ่มเดิมเงื่อนไขหรือบังคับ (constrains) ของปัญหาที่ต้องวิเคราะห์

จากข้อเสนอแนะดังกล่าวอาจช่วยให้สามารถหาคำตอบได้ดีขึ้นหรือรวดเร็ว แต่ Randy L. Haupt ได้กล่าวถึงปัจจัยบางประการที่มีผลต่อการเข้าสู่คำตอบ convergence เช่นจำนวนบิตโครโมโซม ถ้ายิ่งจำนวนบิตมากค่าความถูกต้องแม่นยำ (accuracy) ของคำตอบจะมีค่ามากการหาค่าที่เหมาะสม (optimization) จะมีตัวเลือกเพื่อให้ใกล้คำตอบมากขึ้นแต่จะทำให้แต่ถ้าเกิดการเข้าสู่คำตอบ convergence จะช้า และต้องใช้หน่วยความจำในการทำงานของจินดิกอัลกอริธึมดังกล่าวมาก