
การปรับเพิ่มความยาวโครโมโซมในขั้นตอนวิธีเชิงพันธุกรรมวัตถุประสงค์เดียว
Increment of Chromosome Length in Single-Objective Genetic Algorithms

กิตติพงษ์ บัญโล่ง*

ภาควิชาวิศวกรรมเครื่องกล คณะวิศวกรรมศาสตร์ มหาวิทยาลัยบูรพา

Kittipong Boonlong*

Department of Mechanical Engineering, Faculty of Engineering, Burapha University

บทคัดย่อ

บทความนี้นำเสนอการปรับเพิ่มความยาวโครโมโซมให้กับขั้นตอนวิธีเชิงพันธุกรรมสำหรับปัญหาการหาค่าเหมาะที่สุดวัตถุประสงค์เดียวเพื่อให้ได้คำตอบที่ดีที่สุด แนวคิดนี้หาคำตอบจากกรณีตัวแปรตัดสินใจเข้ารหัสด้วยเลขฐานสองจำนวนบิตน้อยซึ่งจำนวนคำตอบที่เป็นไปได้ทั้งหมดในปริภูมิค้นหาซึ่งง่ายในการหาคำตอบที่ดี หลังจากทำการหาคำตอบจนกระทั่งคำตอบเข้าสู่จึงเพิ่มจำนวนบิตที่แทนแต่ละตัวแปรตัดสินใจจนกระทั่งถึงจำนวนบิตที่กำหนด แนวคิดนี้ยังสามารถประยุกต์ใช้กับขั้นตอนวิธีเชิงพันธุกรรมที่เข้ารหัสด้วยเลขจำนวนจริง โดยคำตอบที่ได้จากแนวคิดนี้นำไปเปรียบเทียบกับคำตอบที่ได้จากการให้ค่าจำนวนบิตต่อตัวแปรตัดสินใจ ปัญหาที่ใช้ทดสอบมีทั้งหมด 7 ปัญหาซึ่งประกอบด้วย 5 ปัญหามาตรฐาน และ 2 ปัญหาการหาค่าเหมาะที่สุดทางด้านวิศวกรรม หลังจากการจำลองพบว่าคำตอบที่ได้จากการปรับเพิ่มความยาวโครโมโซมดีกว่าคำตอบกรณีความยาวโครโมโซมคงที่ทั้งกรณีคำตอบเข้ารหัสเป็นเลขฐานสองและเลขจำนวนจริง โดยคำตอบที่ได้จากแนวคิดนี้ค่อนข้างใกล้เคียงกับคำตอบที่เหมาะสมที่สุดแท้จริงโดยเฉพาะการเข้ารหัสด้วยเลขจำนวนจริง

คำสำคัญ : ขั้นตอนวิธีเชิงพันธุกรรม การหาค่าเหมาะที่สุดวัตถุประสงค์เดียว การปรับเพิ่มความยาวโครโมโซม

Abstract

This paper proposed increment of chromosome length embedded into a genetic algorithm (GA) to solve single-objective optimization problems in order to achieve better solutions. For this idea, the GA initially searches solutions of which a decision variable is encoded by binary string with a few binary bits. Therefore, the total number of possible solutions in search space is small so that it is easy to search for good solutions. After solutions are converged, the number of represented binary bits is gradually increased towards a defined number. The proposed idea can also be particularly embedded in real-coded GAs. Solutions obtained from this idea have been compared to solutions from fixed numbers of represented binary bits. There are 7 tested problems which are 5 benchmark problems and 2 engineering optimization problems. After simulation, solutions from the increment of chromosome length are obviously better than those from fixed chromosome length for both binary and real coding. Especially, solutions obtained by the proposed idea are very close to exact solutions of the tested problems with real coding.

Keywords : Genetic Algorithm, Single-objective Optimization, Increment of Chromosome Length

*E-mail: kittipong@buu.ac.th

บทนำ

การหาคำตอบที่เหมาะสมที่สุดทั้งแบบวัตถุประสงค์เดียวและหลายวัตถุประสงค์นั้น ความยากง่ายของการหาคำตอบขึ้นกับหลายปัจจัย เช่น ลักษณะของปัญหา ลักษณะของปริภูมิค้นหา (Search Space) จำนวนค่าที่เหมาะสมที่สุดเฉพาะที่ (Local Optima) รวมถึงขนาดของปริภูมิค้นหา งานวิจัยจำนวนมากที่เกี่ยวกับการปรับปรุงคำตอบสำหรับปัญหาการหาคำตอบที่เหมาะสมที่สุดวัตถุประสงค์เดียวเป็นการพัฒนาขั้นตอนวิธีในการหาคำตอบหรือการพัฒนาเฉพาะสำหรับปัญหาใดๆ (Shimodaira, 1997; Liang & Leung, 2011; Shi *et al.*, 2012; Gonzalez-Longatt *et al.*, 2012) ซึ่งเป็นประโยชน์มาก หากมีแนวทางที่สามารถประยุกต์กับขั้นตอนวิธีเชิงวิวัฒนาการทุกแบบ ทั้งขั้นตอนวิธีเชิงพันธุกรรม (Genetic Algorithm, GA) ขั้นตอนวิธีจำลองการอบอุ่น (Simulated Annealing, SA) กลยุทธ์วิวัฒนาการ (Evolution Strategy, ES) และขั้นตอนวิธีเชิงวิวัฒนาการชนิดอื่น

งานวิจัยนี้จะนำเสนอแนวคิดที่พิจารณาขนาดของปริภูมิค้นหา (Search Space) ซึ่งประยุกต์กับขั้นตอนวิธีเชิงวิวัฒนาการทุกแบบ โดยแนวคิดนี้มาจากกรณีการเข้ารหัสตัวแปรเป็น เลขฐานสองที่เรียงกันซึ่งเรียกว่า “โครโมโซม” โดยตัวอย่าง หากจำนวนบิตเลขฐานสองซึ่งเรียกว่า “ความยาวโครโมโซม” ของตัวแปรเพิ่มจาก 5 เป็นสองเท่าเป็น 10 ขนาดปริภูมิค้นหาซึ่งเท่ากับจำนวนคำตอบที่เป็นได้ทั้งหมดจะเพิ่มจาก $2^5 = 32$ เป็น $2^{10} = 1024$ ซึ่งจะเพิ่มขึ้นอย่างมาก งานวิจัยก่อนหน้านี้ได้ทำการศึกษาการปรับเพิ่มความยาวโครโมโซมในปัญหาการหาคำตอบโพลีต่อเนืองที่เหมาะสมที่สุดทั้งแบบวัตถุประสงค์เดียวและแบบหลายวัตถุประสงค์ หลักการปรับเพิ่มความยาวโครโมโซมนี้จะเริ่มต้นด้วยคำตอบซึ่งแทนด้วยโครโมโซมเลขฐานสองที่มีจำนวนบิตน้อยซึ่งแสดงว่าเริ่มต้นด้วยการหาคำตอบด้วยปริภูมิค้นหาที่มีคำตอบที่เป็นไปได้จำนวนน้อย จากนั้นค่อยเพิ่มความยาวโครโมโซมจนกระทั่งถึงความยาวโครโมโซมที่กำหนด ซึ่งหลักการนี้ใช้ได้โดยตรงสำหรับปัญหาการหาคำตอบโพลีต่อเนืองที่เหมาะสมที่สุดคำตอบซึ่งเป็นการจัดวางตัวของโครงสร้างเข้ารหัสได้โดยตรงจากเลขฐานสอง (Kim & de Weck, 2005; Boonlong *et al.*, 2010) แต่ยากในการประยุกต์ใช้กับปัญหาการหาคำตอบที่เหมาะสมที่สุดทั่วไปซึ่งคำตอบเป็นเซตของจำนวนจริงจำนวนหนึ่งไม่ได้เข้ารหัสด้วยเลขฐานสองโดยตรง

แนวคิดการปรับเพิ่มความยาวโครโมโซมในงานวิจัยนี้จะเริ่มหาคำตอบจากกรณีตัวแปรตัดสินใจเข้ารหัสเป็นโครโมโซมเลขฐานสองที่มีความยาวน้อย ทำการหาคำตอบจนคำตอบเข้าสู่ จากนั้นก็ทำการเพิ่มความยาวโครโมโซมเรื่อยๆ จนกระทั่งคำตอบ

ของความยาวโครโมโซมสุดท้ายเข้าสู่ ทั้งนี้งานวิจัยนี้ได้นำเสนอการปรับเพิ่มความยาวโครโมโซมนี้กับกรณีตัวแปรตัดสินใจเข้ารหัสเป็นเลขจำนวนจริงอีกด้วย เนื่องจากปัจจุบันงานวิจัยนิยมเข้ารหัสตัวแปรด้วยจำนวนจริง (Ming-Shyan *et al.*, 2012; Stach *et al.*, 2012; Ciomei & Kyriakides, 2012) ซึ่งแนวคิดนี้สามารถประยุกต์ใช้ได้กับขั้นตอนวิธีเชิงวิวัฒนาการต่างๆ ได้ ซึ่งขั้นตอนวิธีหาคำตอบที่เหมาะสมที่สุดที่ใช้ในงานวิจัยนี้คือขั้นตอนวิธีเชิงพันธุกรรม ซึ่งถือเป็นขั้นตอนวิธีวิวัฒนาการชนิดหนึ่ง ขณะที่ปัญหาที่มาทดสอบมีทั้งหมด 7 ปัญหา โดย 5 ปัญหาแรกเป็นปัญหาการหาคำตอบที่เหมาะสมที่สุดแบบวัตถุประสงค์เดียวมาตรฐาน ขณะที่ 2 ปัญหาสุดท้ายเป็นปัญหาการหาคำตอบที่เหมาะสมที่สุดทางด้านวิศวกรรมศาสตร์แบบวัตถุประสงค์เดียว ทั้งนี้หลักการเพิ่มความยาวโครโมโซมนี้น่าจะมีประโยชน์สำหรับนักวิจัยที่ใช้ขั้นตอนวิธีเชิงวิวัฒนาการในการหาคำตอบที่เหมาะสมที่สุดทั้งงานทางด้านวิศวกรรมศาสตร์และวิทยาศาสตร์ โดยแนวคิดนี้สามารถใช้ได้ดีกับปัญหาที่มีตัวแปรตัดสินใจต่อเนื่อง แต่ก็ยังมีข้อจำกัดกับการหาคำตอบที่เหมาะสมที่สุดแบบไม่ต่อเนื่อง (Discrete Optimization) ซึ่งมีตัวแปรตัดสินใจไม่ต่อเนื่อง เช่น ปัญหาการเดินทางของพนักงานขาย (Traveling Salesman Problem) และปัญหาถุงกระสอบ (Knapsack Problem) เป็นต้น

วัตถุประสงค์และวิธีการวิจัย

บทความนี้ใช้ขั้นตอนวิธีเชิงพันธุกรรมในการทดสอบประสิทธิภาพของแนวคิดการปรับเพิ่มความยาวโครโมโซม ดังนั้นจึงแบ่งออกเป็น 2 ส่วนคือขั้นตอนวิธีเชิงพันธุกรรม และการปรับเพิ่มความยาวโครโมโซม

1. ขั้นตอนวิธีเชิงพันธุกรรม

ขั้นตอนวิธีเชิงพันธุกรรม (Genetic Algorithm, GA) ถูกอธิบายอย่างละเอียดใน (Goldberg, 1989) ขั้นตอนการทำงานของ GA เป็นไปตามขั้นตอนดังนี้

- 1) สร้างกลุ่มของประชากรของคำตอบเริ่มต้นโดยการสุ่ม โดยคำตอบจะเข้ารหัสเป็นตัวเลข เช่น เข้ารหัสเป็นเลขฐานสอง คำตอบหนึ่งแทนด้วย กลุ่มตัวเลขที่ต่อกัน ซึ่งจะเรียกว่าโครโมโซม ตัวอย่างคำตอบ x เป็นเวกเตอร์ของตัวแปร $\{x_1, x_2, x_3\}$ โดย $x_1 \in [-1, 1]$, $x_2 \in [0, 1]$, $x_3 \in [1, 5]$ โดยที่ x_1, x_2 และ x_3 แทนด้วยเลขฐานสองที่มีความยาว 4, 5 และ 6 บิตตามลำดับ (โดยทั่วไปแล้วแต่ละตัวแปรแทนด้วยจำนวนบิตที่เท่ากัน) ดังนั้นโครโมโซมของคำตอบ x เข้ารหัสด้วยเลขฐานสองที่มีจำนวนบิตเท่ากับ $4+5+6 = 15$ บิต ตัวอย่างเช่นหาก x เข้ารหัสเป็น 110001011100101 โดยโครโมโซมนี้อาจแบ่งได้เป็น 3 ส่วน

โดยที่แต่ละส่วนแทน x_1, x_2 และ x_3 ตามลำดับ ดังสมการต่อไปนี้

$$x = \underbrace{11000}_{x_1} \underbrace{1011}_{x_2} \underbrace{1100101}_{x_3} \quad (1)$$

กำหนดให้ V_1, V_2 และ V_3 แทนจำนวนเต็มของเลขฐานสองในส่วนที่แทน x_1, x_2 และ x_3 ตามลำดับ โดยสามารถคำนวณได้โดยที่ $V_1 = (1100)_2 = 1(2^3)+1(2^2)+0(2^1)+0(2^0) = 12$ ในทำนองเดียวกัน $V_2 = (01011)_2 = 11$ และ $V_3 = (100101)_2 = 37$ จากนั้น x_i สามารถถอดรหัสได้ดังสมการต่อไปนี้

$$x_i = x_i^l + \frac{V_i (x_i^u - x_i^l)}{2^{L_i} - 1} \quad (2)$$

โดย x_i^l, x_i^u และ L_i คือค่าต่ำสุด ค่าสูงสุด และความยาวของเลขฐานสองที่แทน x_i จากสมการดังกล่าวจะได้ว่า x_1, x_2 และ x_3 มีค่าเท่ากับ 0.6, 0.3548 และ 3.3492 ตามลำดับ ส่วนการเข้ารหัสด้วยจำนวนจริง ค่าตอบ x แทนด้วยโครโมโซมจำนวนจริงที่มีความยาวเท่ากับจำนวนตัวแปรดังนี้

$$x = \underbrace{0.6}_{x_1} \underbrace{0.3548}_{x_2} \underbrace{3.3492}_{x_3} \quad (3)$$

2) ทำการถอดรหัสของโครโมโซมทั้งหมดในกลุ่มประชากรเพื่อที่จะได้กลุ่มคำตอบของปัญหาที่พิจารณา

3) หลังจากได้คำตอบจากการถอดรหัส ทำการหาค่าฟังก์ชันวัตถุประสงค์ของแต่ละคำตอบในกลุ่มประชากรทั้งหมด

4) ทำการคำนวณค่าความแข็งแรงของแต่ละคำตอบโดยใช้ค่าวัตถุประสงค์ที่ได้มาจากข้อ 3)

5) เลือกกลุ่มประชากรรุ่นพ่อแม่มาจากกลุ่มประชากรรุ่นปัจจุบัน โดยในการเลือกนั้นจะใช้ค่าความแข็งแรงเป็นตัวบ่งชี้ว่าจะเลือกคำตอบใดจากประชากรรุ่นปัจจุบันนี้

6) ทำการเปลี่ยนแปลงกลุ่มประชากรรุ่นพ่อแม่โดยใช้ตัวดำเนินการพันธุกรรม ซึ่งคือ การสลับสายพันธุ์ (Crossover) และการกลายพันธุ์ (Mutation) เป้าหมายของสลับสายพันธุ์คือทำการแลกเปลี่ยนของประชากรรุ่นพ่อแม่ที่คัดเลือกมาแล้ว โดยถือได้ว่าประชากรที่คัดมาดังกล่าวเป็นคำตอบที่ดีมีค่าบิตที่ดีเมื่อทำการแลกเปลี่ยนกันก็จะมีโอกาสได้คำตอบที่ดีขึ้น โดยความน่าจะเป็นของการสลับสายพันธุ์นั้นจะมีค่าสูงมีค่าประมาณ 0.8-1.0 การสลับสายพันธุ์แต่ละครั้งกระทำกับคำตอบคู่หนึ่งซึ่งจะได้คำตอบใหม่มาอีกคู่หนึ่ง ตัวอย่างการสลับสายพันธุ์แบบจุดเดียว (One-Point Crossover) สำหรับโครโมโซมเลขฐานสองแสดงดังภาพที่ 1 โดยตำแหน่งสลับสายพันธุ์ซึ่งได้จากการสุ่มแสดงโดยเส้นประ โดยจะมีการแลกเปลี่ยนบิตหลังตำแหน่งนี้ ส่วนบิตก่อนเส้นประไม่มีการแลกเปลี่ยนในคำตอบคู่ใหม่

$$\begin{array}{ccc} 110:0001 & \longrightarrow & 110:1100 \\ 011:1100 & & 011:0001 \end{array}$$

ภาพที่ 1 การสลับสายพันธุ์แบบจุดเดียว

ขณะที่เป้าหมายของการกลายพันธุ์คือทำการเปลี่ยนแปลงคำตอบเพียงเล็กน้อยเพื่อเพิ่มความหลากหลายของคำตอบในประชากรปัจจุบัน ความน่าจะเป็นของการกลายพันธุ์นั้นจะมีค่าต่ำมีค่าประมาณ 0.001-0.05 แตกต่างจากการสลับสายพันธุ์แต่ละครั้งของการกลายพันธุ์กระทำกับคำตอบเพียงหนึ่งคำตอบซึ่งจะได้ผลลัพธ์หนึ่งคำตอบใหม่ ภาพที่ 2 แสดงตัวอย่างการสลับสายพันธุ์แบบกลับค่าบิต (Flip-bit Mutation) สำหรับโครโมโซมเลขฐานสอง โดยความน่าจะเป็นที่สุ่มของบิตตำแหน่งที่ 4 น้อยกว่าความน่าจะเป็นการกลายพันธุ์ที่ตั้งไว้ ค่าบิตตำแหน่งนี้ก็จะเปลี่ยนค่าไปดังแสดงในภาพ

$$1100001 \longrightarrow 1101001$$

ภาพที่ 2 การสลับสายพันธุ์แบบกลับค่าบิต

สำหรับกรณีโครโมโซมจำนวนจริงนั้นได้การนำเสนอการสลับสายพันธุ์และการกลายพันธุ์หลายแบบ โดยงานวิจัยนี้จะใช้การสลับสายพันธุ์ Simulated Binary Crossover (SBX) (Deb & Agrawal, 1995) ซึ่งประยุกต์มาจากการ One-Point Crossover ในโครโมโซมฐานสอง ส่วนการกลายพันธุ์ใช้ Variable-Wise Polynomial Mutation (VPM) (Deb, 1997)

7) กลับไปทำซ้ำขั้นตอนที่ 2 จนกระทั่งคำตอบลู่อูเข้าหรือจนกระทั่งครบจำนวนการทำซ้ำที่กำหนด โดย 1 รอบของขั้นตอนที่ 2 ถึง 6 ก็คือ 1 รุ่น (Generation) ของการหาคำตอบด้วย GA คำตอบดีที่สุดของประชากรรุ่นสุดท้ายก็คือผลลัพธ์ของ GA นอกจากนั้นอาจจะมีการเพิ่มตัวดำเนินการที่เรียกว่า อภิชนนิยม (Elitism) หลังจากขั้นตอนที่ 4) โดย Elitism เป็นกระบวนการซึ่งนำคำตอบดีที่สุดรุ่นปัจจุบันไปเป็นคำตอบในรุ่นถัดไปเลยโดยไม่ต้องผ่านการเปลี่ยนแปลงด้วยการสลับสายพันธุ์และการกลายพันธุ์

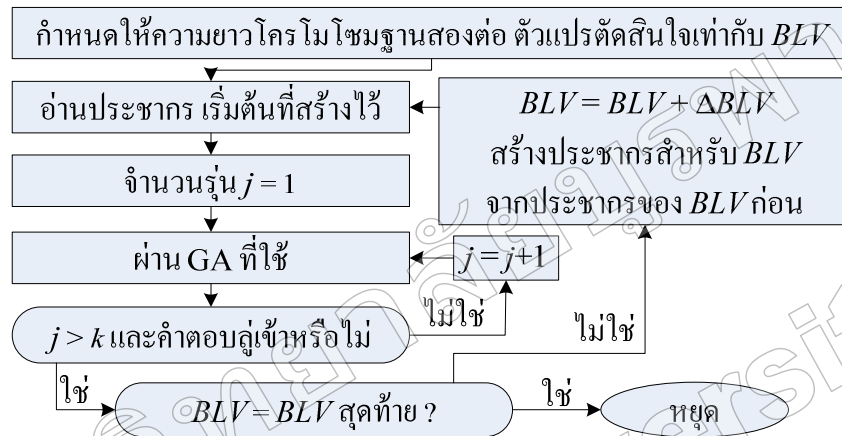
2. การปรับเพิ่มความยาวโครโมโซม

การหาคำตอบพร้อมด้วยการปรับเพิ่มความยาวโครโมโซมแสดงดังภาพที่ 3 กระบวนการเริ่มต้นด้วยความยาวโครโมโซมฐานสองต่อตัวแปรตัดสิน (Binary Chromosome Length per Variable; BLV) เริ่มต้น จากนั้นก็หาคำตอบด้วย GA ที่ใช้เมื่อจำนวนรุ่น (Number of Generations) มากกว่า k ซึ่งเป็น

ค่าที่กำหนดขึ้น ค่าตอบที่ดีที่สุดของรุ่นปัจจุบันก็จะถูกนำไปเปรียบเทียบกับค่าตอบที่ดีที่สุดของ k รุ่นที่แล้ว หากค่าตอบที่ดีที่สุดปัจจุบันมีค่าวัตถุประสงค์เท่ากับหรือต่ำกว่า ก็ถือเป็นการสิ้นสุดการหาค่าตอบของ BLV ขณะนั้น จากนั้นก็ทำการเพิ่ม BLV จากนั้นหาค่าตอบซ้ำอีก จนกระทั่งค่าตอบของ BLV สุดท้ายลู่เข้า ค่าตอบที่ดีที่สุดของ BLV สุดท้ายคือผลลัพธ์ของกระบวนการหาค่าตอบนี้

เนื่องจากการปรับเพิ่มความยาวโครโมโซมดังนั้นโครโมโซมที่แทนด้วยคำตอบหนึ่งก็ต้องเปลี่ยนไปด้วยสำหรับกรณีคำตอบ

เข้ารหัสเป็นเลขฐานสอง ส่วนกรณีคำตอบเข้ารหัสเป็นเลขจำนวนจริงนั้น เนื่องจากความยาวของโครโมโซมซึ่งเท่ากับจำนวนตัวแปรตัดสินใจคงที่ ช่วงเปลี่ยนค่า BLV เทียบเท่า จึงไม่ต้องพิจารณาอะไรเป็นพิเศษ แต่จำเป็นต้องมีการปรับเปลี่ยนค่าของตัวแปรตัดสินใจหลังการสลับสายพันธุ์และการกลายพันธุ์ สำหรับ BLV เทียบเท่าค่าใดๆ การประยุกต์ใช้การปรับเพิ่มความยาวโครโมโซมร่วมกับ GA จะอธิบายได้เป็นสองกรณีตามชนิดของการเข้ารหัสดังต่อไปนี้



ภาพที่ 3 การหาค่าตอบของ GA ร่วมกับการปรับเพิ่มความยาวโครโมโซม

2.1 กรณีคำตอบเข้ารหัสเป็นเลขฐานสอง

การเข้ารหัสด้วยเลขฐานสองซึ่งใช้ตัวเลข '0' และ '1' แบ่งได้เป็น 2 แบบคือรหัสฐานสองแบบปกติ (Binary Code) และรหัสฐานสองแบบเกรย์ (Gray Code) โดยเริ่มต้น ตัวแปรตัดสินใจแต่ละตัวเข้ารหัสด้วยเลขฐานสองซึ่งมีความยาวโครโมโซมเท่ากับ BLV ซึ่งก็คือจำนวนของบิต 0 หรือ 1 ซึ่งแทนตัวแปรนั้น โดยเมื่อ BLV มีค่าเพิ่มขึ้น จะได้โครโมโซมใหม่สำหรับ BLV ที่เพิ่มขึ้น

หลักการปรับเปลี่ยนโครโมโซมคือ สัดส่วนของเลขฐานสิบซึ่งแทนโครโมโซมส่วนที่แทนตัวแปรตัดสินใจที่พิจารณาต่อสอง ยกกำลังความยาวโครโมโซมต้องคงที่ตลอด ดังนั้นหากมีการเพิ่มความยาวโครโมโซมจาก m เป็น $m + n$ ซึ่งจะได้ว่า

$$\frac{V_m}{2^m} = \frac{V_{m+n}}{2^{m+n}} \quad (4)$$

โดย V_m และ V_{m+n} เป็นเลขฐานสิบซึ่งแทนโครโมโซมยาว m และ $m + n$ ของส่วนที่แทนตัวแปรตัดสินใจที่พิจารณาตามลำดับ ซึ่งจะได้ว่า

$$V_{m+n} = 2^n \times V_m \quad (5)$$

โดยหลักการนี้จะได้แสดงตัวอย่างของโครโมโซมที่เปลี่ยนไปสำหรับโครโมโซมรหัสฐานสองแบบปกติ (Binary Code) สำหรับกรณี BLV เพิ่มขึ้น 1 และ 2 บิต เป็นดังภาพที่ 4 และภาพที่ 5 ตามลำดับ

จากภาพที่ 4 และภาพที่ 5 แสดงให้เห็นว่าโครโมโซมรหัสฐานสองนั้นหลังจากเพิ่มไม่ว่าด้วยจำนวนบิตเท่าใดบิตที่เพิ่มขึ้นจะเป็น '0' เพียงค่าเดียว ซึ่งทำให้การหาค่าตอบของ GA ยากขึ้นเนื่องจากไม่มีความหลากหลายของบิตที่เพิ่ม ซึ่งอาจทำให้เกิดการลู่เข้าก่อนกำหนด ดังนั้นการเข้ารหัสฐานสองแบบปกติจึงไม่เหมาะสมสำหรับการหาค่าตอบแบบปรับเพิ่มความยาวโครโมโซม แต่เมื่อพิจารณาการเข้ารหัสฐานสองแบบเกรย์ดังภาพที่ 6 และภาพที่ 7 พบว่าเมื่อเพิ่มโครโมโซมขึ้นทีละ 1 บิต บิตที่เพิ่มขึ้นจะมีทั้ง '0' และ '1' แต่เมื่อเพิ่มโครโมโซมเพิ่มขึ้นทีละ 2 บิต ก็จะพบว่าบิตสุดท้ายเป็นศูนย์เพียงอย่างเดียว เนื่องด้วยเงื่อนไขดังสมการ (5) จึงทำการบิตที่ 2 เป็น '1' ไม่ได้ตามหลักการทั่วไปของการเข้ารหัสฐานสองแบบเกรย์ ดังนั้นรหัสฐานสองแบบเกรย์โดยที่เพิ่ม BLV ขึ้นทีละ 1 น่าจะเหมาะสมที่สุดในการใช้กับการหาค่าตอบพร้อมด้วยการปรับเพิ่มความยาวโครโมโซมเนื่องจากบิตที่เพิ่มขึ้นยังคงความหลากหลายอยู่จะทำให้สามารถป้องกันการลู่เข้าก่อนกำหนดได้

100	→	100 <u>0</u>
0110	→	0110 <u>0</u>
11011	→	11011 <u>0</u>
011010	→	011010 <u>0</u>

ภาพที่ 4 โครโมโซมรหัสฐานสองแบบปกติที่เปลี่ยนไปเมื่อความยาวเพิ่มขึ้น 1 บิต

100	→	100 <u>00</u>
0110	→	0110 <u>00</u>
11011	→	11011 <u>00</u>
011010	→	011010 <u>00</u>

ภาพที่ 5 โครโมโซมรหัสฐานสองแบบปกติที่เปลี่ยนไปเมื่อความยาวเพิ่มขึ้น 2 บิต

100	→	100 <u>1</u>
0110	→	0110 <u>0</u>
11011	→	11011 <u>0</u>
011010	→	011010 <u>1</u>

ภาพที่ 6 โครโมโซมรหัสฐานสองแบบเกรย์ที่เปลี่ยนไปเมื่อความยาวเพิ่มขึ้น 1 บิต

100	→	100 <u>10</u>
0110	→	0110 <u>00</u>
11011	→	11011 <u>00</u>
011010	→	011010 <u>10</u>

ภาพที่ 7 โครโมโซมรหัสฐานสองแบบเกรย์ที่เปลี่ยนไปเมื่อความยาวเพิ่มขึ้น 2 บิต

2.2 กรณีคำตอบเข้ารหัสเป็นเลขจำนวนจริง

เนื่องด้วยการเข้ารหัสแบบจำนวนจริง ความยาวของโครโมโซมจะคงที่ซึ่งเท่ากับจำนวนตัวแปรตัดสินใจ โดยการจะใช้หลักการของเลขฐานมาประยุกต์ใช้ ตัวแปรตัดสินใจซึ่งแทนด้วยค่าของแต่ละบิตจะต้องปรับเปลี่ยนเพื่อให้เป็นค่าเปรียบเสมือนของเลขฐานสอง ซึ่งหลักการเปลี่ยนก็ง่ายๆ ก็คือเปลี่ยนตามช่วงของค่าที่ได้ ตัวอย่างเช่น หาก BLV เปรียบเสมือนของตัวแปรตัดสินใจลำดับที่ i เท่ากับ L_i ภายหลังจากผ่านการสลับสายพันธุและการกลายพันธุ์ได้ผลเป็น $x_{i,cur}$ จากนั้นคำนวณหาเลขฐานสิบเทียบเท่า $V_{i,cur}$ ซึ่งแทน $x_{i,cur}$ ได้โดยประยุกต์สมการที่ (2) จะได้ $V_{i,cur}$ เป็นดังสมการดังต่อไปนี้

$$V_{i,cur} = \left(\frac{x_{i,cur} - x_i^l}{x_i^u - x_i^l} \right) 2^{L_i} \quad (6)$$

จากนั้นทำการเปลี่ยนเลขฐานสิบเทียบเท่า $V_{i,cur}$ ซึ่งเป็นเลขทศนิยมเป็นตัวเลขฐานสิบเทียบเท่าที่เป็นจำนวนเต็มแทนด้วย V_i ซึ่งคำนวณง่ายๆ จากค่า $V_{i,cur}$ ดังสมการ (7) จากนั้นคำนวณค่าตัวแปรตัดสินใจที่เปลี่ยนแปลงแล้วซึ่งแทนด้วย x_i ได้ดังสมการ (8) หลังจากนั้น x_i ที่ทำการเปลี่ยนแล้วคือตัวแปรตัดสินใจของคำตอบรุ่นถัดไป

$$V_i = \begin{cases} 0, & \text{if } V_{i,cur} \in [0,1] \\ 1, & \text{if } V_{i,cur} \in [1,2] \\ \vdots & \\ 2^{L_i-1} - 2, & \text{if } V_{i,cur} \in [2^{L_i-1} - 1, 2^{L_i-1} - 1] \\ 2^{L_i-1} - 1, & \text{if } V_{i,cur} \in [2^{L_i-1} - 1, 2^{L_i}] \end{cases} \quad (7)$$

$$x_i = x_i^l + \frac{V_i (x_i^u - x_i^l)}{2^{L_i} - 1} \quad (8)$$

ในการหาคำตอบที่มีการปรับเพิ่มความยาวโครโมโซมนั้น จะมีการกำหนดความยาวโครโมโซมฐานสองต่อตัวแปรตัดสินใจ (BLV) มีค่าตั้งแต่ BLV_{min} ถึง BLV_{max} โดยเพิ่มขึ้นทีละ ΔBLV โดยความยาวโครโมโซมตั้งแต่ BLV_{min} จนถึง $BLV_{max} - \Delta BLV$ ก็จะมีการเปลี่ยนแปลงค่าของตัวแปรตัดสินใจหลังจากการสลับสายพันธุและการกลายพันธุ์ดังแสดงก่อนหน้านี้ แต่เมื่อความยาวโครโมโซมเทียบเท่านี้เท่ากับค่าสุดท้าย BLV_{max} ก็จะไม่มีการเปลี่ยนแปลงอีกต่อไปเพื่อให้การหาคำตอบเหมือนกับการหาคำตอบของโครโมโซมเลขจำนวนจริงทั่วไป

ผลการวิจัยและวิจารณ์ผล

ปัญหาที่ทำการทดสอบมีทั้งหมด 7 ปัญหา โดย 5 ปัญหาแรก (Yao et al., 1999) เป็นปัญหาวัตถุประสงค์เดียวมาตรฐาน ส่วน

2 ปัญหาสุดท้าย (Cagnina *et al.*, 2008) เป็นปัญหาการหาค่าเหมาะที่สุดทางด้านวิศวกรรมแบบวัตถุประสงค์เดียว โดยปัญหาทั้งหมดเป็นปัญหาการหาค่าน้อยที่สุด (Minimization Problem) รายละเอียดของปัญหาแสดงดังต่อไปนี้

ปัญหาที่ 1 (f_1) Schwefel's Problem 1.2 :

$$f_1(x) = \sum_{i=1}^{30} \left(\sum_{j=1}^i x_j \right)^2 \quad (9)$$

โดย $-100 \leq x_i \leq 100, i = 1, \dots, 30$ คำตอบที่แท้จริง $x = \{0, 0, \dots, 0\}$ โดยวัตถุประสงค์ที่ดีที่สุด $f_1(x) = 0$

ปัญหาที่ 2 (f_2) Schwefel's Problem 2.21 :

$$f_2(x) = \max_i \{|x_i|, 1 \leq i \leq 30\} \quad (10)$$

โดย $-100 \leq x_i \leq 100, i = 1, \dots, 30$ คำตอบที่แท้จริง $x = \{0, 0, \dots, 0\}$ โดยวัตถุประสงค์ที่ดีที่สุด $f_2(x) = 0$

ปัญหาที่ 3 (f_3) Generalized Rosenbrock's Function :

$$f_3(x) = \sum_{i=1}^{29} [100(x_{i+1} - x_i^2) + (x_i - 1)^2] \quad (11)$$

โดย $-30 \leq x_i \leq 30, i = 1, \dots, 30$ คำตอบที่แท้จริง $x = \{1, 1, \dots, 1\}$ โดยวัตถุประสงค์ที่ดีที่สุด $f_3(x) = 0$

ปัญหาที่ 4 (f_4) Quartic Function พร้อมด้วย Noise :

$$f_4(x) = \sum_{i=1}^{30} |x_i|^4 + \text{random}(0,1) \quad (12)$$

โดย $-1.28 \leq x_i \leq 1.28, i = 1, \dots, 30$ คำตอบที่แท้จริง $x = \{1, 1, \dots, 1\}$ โดยวัตถุประสงค์ที่ดีที่สุด $f_4(x) = 0$

ปัญหาที่ 5 (f_5) Generalized Griewank Function :

$$f_5(x) = \frac{1}{4,000} \sum_{i=1}^{30} x_i^2 - \prod_{i=1}^{30} \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1 \quad (13)$$

โดย $-600 \leq x_i \leq 600, i = 1, \dots, 30$ คำตอบที่แท้จริง $x = \{0, 0, \dots, 0\}$ โดยวัตถุประสงค์ที่ดีที่สุด $f_5(x) = 0$

ปัญหาที่ 6 ปัญหาการออกแบบคานเชื่อมเหมาะที่สุด (Welded Beam Design Optimization Problem) ปัญหานี้ ออกแบบคานเชื่อมติดเพื่อให้ต้นทุนการผลิตน้อยที่สุด ภาพที่ 8 แสดงคานที่ต้องการเชื่อมซึ่งประกอบด้วยคาน A และแนวเชื่อมซึ่งยึดคานกับชิ้นส่วน B โดยปัญหามี 4 ตัวแปรออกแบบ x_1, x_2, x_3 และ x_4 โดยมีเงื่อนไขบังคับของความเค้นเฉือน (τ) ความเค้นดัดของคาน (σ) ภาระ Bucking บนบาร์ (P_c) และระยะแอนตัวของปลายคาน δ และเงื่อนไขขนาดด้านข้าง วัตถุประสงค์ของปัญหานี้แสดงดังสมการต่อไปนี้

$$f_6(x) = 1.10471x_1^2x_2 + 0.04811x_3x_4(14.0+x_2) \quad (14)$$

โดยมีเงื่อนไขบังคับ 7 เงื่อนไข (g_1-g_7) ดังนี้

$$\begin{aligned} g_1(x) &= \tau(x) - \tau_{\max} \leq 0, \\ g_2(x) &= \sigma(x) - \sigma_{\max} \leq 0, \\ g_3(x) &= x_1 - x_4 \leq 0 \\ g_4(x) &= 0.10471x_1^2 + 0.04811x_3x_4(L+x_2) - 5.0 \leq 0, \\ g_5(x) &= 0.125 - x_1 \leq 0 \\ g_6(x) &= \delta(x) - \delta_{\max} \leq 0 \text{ และ} \\ g_7(x) &= P - P_c(x) \leq 0 \end{aligned}$$

โดยที่

$$\tau(x) = \sqrt{(\tau')^2 + (2\tau'\tau'')\frac{x_2}{2R} + (\tau'')^2}$$

$$\tau' = \frac{P}{\sqrt{2}x_1x_2}, \tau'' = \frac{MR}{J}$$

$$M = P\left(L + \frac{x_2}{2}\right), R = \sqrt{\left(\frac{x_2}{2}\right)^2 + \left(\frac{x_1+x_2}{2}\right)^2},$$

$$J = 2\left\{x_1x_2\sqrt{2}\frac{x_2^2}{12} + \left(\frac{x_1+x_3}{2}\right)^2\right\},$$

$$\alpha(x) = \frac{6PL}{x_1x_3^2}, \sigma(x) = \frac{4PL^3}{Ex_3^3x_4} \text{ และ}$$

$$P_c(x) = \frac{4.013E\sqrt{\frac{x_3^2x_4^6}{36}}}{L^2} \left(1 - \frac{x_3}{2L}\sqrt{\frac{E}{4G}}\right)$$

โดยที่ $0.1 \leq x_1, x_4 \leq 2.0, 0.1 \leq x_2, x_3 \leq 10.0$, คำตอบที่แท้จริง $x = \{0.205730, 3.470489, 9.036624, 0.205729\}$

โดยวัตถุประสงค์ที่ดีที่สุด $f_6(x) = 1.724852$

ปัญหาที่ 7 ปัญหาการออกแบบสปริงเหมาะที่สุด (Spring Design Optimization Problem) ปัญหานี้มีวัตถุประสงค์เพื่อให้น้ำหนักของสปริงน้อยที่สุด โดยมีเงื่อนไขบังคับของระยะเคลื่อนตัวของสปริง ความเค้นเฉือน ความถี่ธรรมชาติของสปริง (Surge frequency) และขอบเขตเส้นผ่านศูนย์กลางรอบนอก โดยจะมีตัวแปรที่ใช้ในการออกแบบ 3 ตัวแปร ได้แก่ เส้นผ่านศูนย์กลางของลวดสปริง x_1 เส้นผ่านศูนย์กลางของขดสปริง x_2 และจำนวนขดสปริง x_3 โดยวัตถุประสงค์ของปัญหานี้แสดงดังสมการต่อไปนี้

$$f_7(x) = (x_3 + 2)x_2x_1^2 \quad (15)$$

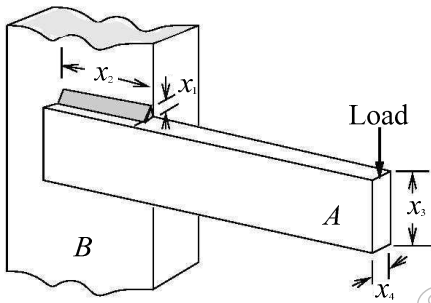
โดยมีเงื่อนไขบังคับ 4 เงื่อนไข ($g_1 - g_4$) ดังนี้

$$g_1(x) = 1 - \frac{x_2^2 x_3}{71,785 x_1^4} \leq 0,$$

$$g_2(x) = 1 - \frac{4x_2^2 - x_1 x_2}{12,566 x_2 x_1^3 - x_1^4} + \frac{1}{5,108 x_1^4} - 1 \leq 0,$$

$$g_3(x) = 1 - \frac{140.45 x_1}{x_2^2 x_3} \leq 0, \text{ และ}$$

$$g_4(x) = \frac{x_2 + x_1}{1.5} - 1 \leq 0$$

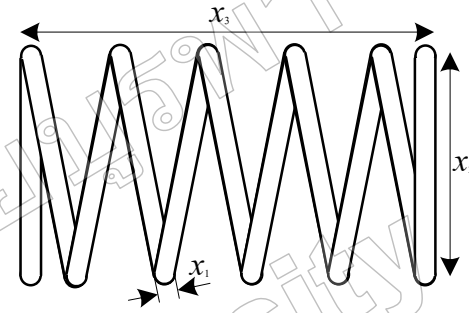


ภาพที่ 8 คานเชื่อมที่ใช้ออกแบบ

ในการหาคำตอบของ GA ร่วมกับการปรับเพิ่มความยาวโครโมโซมแบ่งเป็น 2 กรณี คือกรณีโครโมโซมเลขฐานสองแบบเกรย์ และกรณีโครโมโซมเลขจำนวนจริง ความยาวโครโมโซมฐานสองต่อตัวแปรตัดสิน (BLV) เปรียบเสมือน (ภาพที่ 3) กำหนดให้เท่ากับ 5 ถึง 30 โดย ΔBLV ซึ่งเป็นค่าที่เพิ่มขึ้นของ BLV เท่ากับ 1 โดยคำตอบที่ได้จากการปรับเพิ่มความยาวโครโมโซม (Increased Chromosome Length; ICL) จะนำไปเปรียบเทียบกับกรณีการกำหนดความยาวโครโมโซมคงที่ (Fixed Chromosome Length; FCL) ด้วยการหาคำตอบที่มีจำนวนรุ่น (Number of Generations) ในการคำตอบแต่ละครั้งเท่ากัน ดังนั้นในการคำตอบจะต้องหาคำตอบกรณีการปรับเพิ่มความยาวโครโมโซมก่อน จากนั้นจึงใช้จำนวนรุ่นแต่ละครั้งของการคำตอบของแต่ละปัญหาไปใช้ในการกำหนดจำนวนรุ่นของกรณีการกำหนดความยาวโครโมโซม (FCL) โดยจะกำหนด BLV ที่ใช้ในงานวิจัยนี้เท่ากับ 5, 10, 15, 20 และ 30 สำหรับโครโมโซมเลขจำนวนจริง กรณี FCL เนื่องจากต้องการเปรียบเทียบกับกรณีการคำตอบแบบทั่วไป ดังนั้นจึงกำหนดให้กรณี BLV มากสุดซึ่งเท่ากับ 30 หาคำตอบเหมือนกับการคำตอบแบบทั่วไปของการเข้ารหัสแบบจำนวนจริง ส่วนกรณี BLV เท่ากับ 5, 10, 15 และ 20 เนื่องด้วยเพราะว่าโครโมโซมเลขจำนวนจริงนั้น

โดยที่ $0.05 \leq x_1 \leq 2.0$, $0.25 \leq x_2 \leq 2.0$, $0.25 \leq x_3 \leq 15.0$ คำตอบที่แท้จริง $x = \{0.051690, 0.356750, 11.287126\}$ โดยให้ค่าที่ดีที่สุด $f_7(x) = 0.012665$

ทั้งนี้ปัญหาที่ 6 และ 7 เป็นปัญหาการหาค่าเหมาะที่สุดแบบมีเงื่อนไขบังคับ (Constrained Optimization Problems) โดยปัญหาทั้งสองจะใช้ Self Adaptive Penalty Function ร่วมกับ GA ในการหาคำตอบ ซึ่งใช้ได้ทั้งปัญหาแบบมีเงื่อนไขบังคับวัตถุประสงค์เดียว (Tessema & Yen, 2006) และปัญหาแบบมีเงื่อนไขบังคับหลายวัตถุประสงค์ (Woldesenbet et al., 2009)



ภาพที่ 9 สปริงที่ใช้ออกแบบ

ตัวแปรตัดสินใจมีค่าต่อเนื่องภายหลังการสลับสายพันธุและการกลายพันธุ์ แต่เนื่องจากตัวแปรตัดสินสำหรับ BLV ดังกล่าวเป็นตัวแปรไม่ต่อเนื่อง ดังนั้นจึงจำเป็นต้องมีการเปลี่ยนค่าของตัวแปรตัดสินใจภายหลังการสลับสายพันธุและการกลายพันธุ์ดังสมการ (6)-(8) ขณะที่ตัวแปรตัดสินใจสำหรับโครโมโซมเลขฐานสองทั้งก่อนและหลังกระบวนการสลับสายพันธุและการกลายพันธุ์มีค่าไม่ต่อเนื่องทั้งคู่จึงไม่จำเป็นต้องมีการเปลี่ยนแปลงค่าของตัวแปร

แต่ละปัญหาจะมีการเปรียบเทียบคำตอบจาก 6 แบบ ก็คือ ICL และ 5 ค่าของ BLV ของกรณี FCL ตารางที่ 1 แสดงพารามิเตอร์ที่ใช้ในการคำตอบ โดยค่าน้อยที่สุด ค่าเฉลี่ย ค่ามากที่สุดของจำนวนรุ่นที่ใช้ของแต่ละปัญหาแสดงในตารางที่ 2 ขณะที่ค่าวัตถุประสงค์ของคำตอบที่ดีที่สุดที่ได้ของกรณีโครโมโซมเลขฐานสองและเลขจำนวนจริงแสดงโดย ตารางที่ 3 และ ตารางที่ 4 ตามลำดับ โดยในตารางทั้งสองแต่ละปัญหาตัวเลขหน้าคือวัตถุประสงค์เฉลี่ยที่ดีที่สุด ส่วนเลขขีดเส้นใต้คือวัตถุประสงค์เฉลี่ยดีเป็นอันดับสอง จากตารางที่ 2 พบว่าโดยส่วนใหญ่จำนวนรุ่นที่ใช้ของโครโมโซมเลขจำนวนจริงมากกว่าโครโมโซมเลขฐานสอง ส่วนค่าวัตถุประสงค์ที่ได้ของกรณีโครโมโซมเลขฐานสองดังตารางที่ 3 พบว่า คำตอบจาก ICL ให้ค่าวัตถุประสงค์ที่ดีที่สุดสำหรับ

ตารางที่ 1 พารามิเตอร์ที่ใช้ในการหาคำตอบ

พารามิเตอร์	โครโมโซมเลขฐานสอง	โครโมโซมเลขจำนวนจริง
จำนวนประชากร	100	
ความน่าจะเป็นในการสลับสายพันธุ	1.0	
ชนิดการสลับสายพันธุ	One-point	SBX
ความน่าจะเป็นในการกลายพันธุ	0.025	1/NV
ชนิดการสลับกลายพันธุ	Flip-bit	Variable-wise polynomial
รุ่นที่ k ก่อนหน้าในการเปรียบเทียบการลู่เข้าของคำตอบ	$k = 20$ สำหรับปัญหา f_1 ถึง f_5 , $k = 200$ สำหรับปัญหา f_6 และ f_7	
ความยาวโครโมโซมฐานสองต่อตัวแปรตัดสินใจ (BLV)	5, 6, 7, ..., 30 สำหรับ ICL, 5, 10, 15, 20, 30 สำหรับกรณี FCL	
จำนวนการหาคำตอบซ้ำของแต่ละปัญหา	30	

โดยที่ NV คือจำนวนตัวแปรตัดสินใจแต่ละปัญหา

ปัญหา f_1, f_3, f_5, f_6 และ f_7 แม้ว่าปัญหา f_2 และ f_4 คำตอบที่ได้จาก ICL ไม่ใช่คำตอบที่ดีที่สุด แต่ก็พบว่าให้ค่าที่ใกล้เคียงกับค่าที่ดีที่สุดค่อนข้างมากเช่น f_2 ค่าวัตถุประสงค์เฉลี่ย 0.5539 ขณะที่กรณีที่ดีที่สุดคือกรณีที่กำหนดจำนวนบิตคงที่ 20 บิต (Fix-20) มีค่าวัตถุประสงค์เฉลี่ย 0.5423 ซึ่งใกล้เคียงกัน เช่นเดียวกันกับปัญหา f_4 ค่าวัตถุประสงค์เฉลี่ยมีค่าเป็น 0.0095 (ICL) และ 0.0085 (Fix-5) ซึ่งถือได้ว่าใกล้เคียงกัน เมื่อเทียบกับปัญหาซึ่ง ICL ให้ค่าวัตถุประสงค์เฉลี่ยที่ดีที่สุด ซึ่ง ICL ให้ค่าวัตถุประสงค์ดีกว่ากรณี FCL ค่อนข้างมาก โดยรวมแล้วถือได้ว่ากรณีโครโมโซมเลขฐานสอง ICL

ดีกว่า FCL อย่างชัดเจน ส่วนกรณีโครโมโซมเลขจำนวนจริงดังตารางที่ 4 ก็พบว่าค่าวัตถุประสงค์เฉลี่ยของ ICL ดีกว่า FCL เกือบทุกปัญหายกเว้น f_2 เพียงปัญหาเดียวที่ Fix-20 (0.1006) ดีกว่า ICL (0.1038) เพียงเล็กน้อย ซึ่งเมื่อเปรียบเทียบกันก็จะพบว่ากรณีโครโมโซมเลขจำนวนจริง ICL ให้คำตอบที่ดีกว่า FCL ค่อนข้างมาก รวมถึงว่าเมื่อพิจารณาเปรียบเทียบค่าวัตถุประสงค์ที่ได้จาก ICL กับค่าวัตถุประสงค์ของคำตอบที่แท้จริง (Exact Solution) ก็จะพบว่าคำตอบที่ได้จาก ICL ให้ค่าวัตถุประสงค์ใกล้เคียงกับค่าวัตถุประสงค์ของคำตอบที่แท้จริงค่อนข้างมาก

ตารางที่ 2 จำนวนรุ่นที่ใช้ของโครโมโซมเลขฐานสองและโครโมโซมเลขจำนวนจริง

ปัญหา	จำนวนรุ่นการหาคำตอบ					
	โครโมโซมเลขฐานสอง			โครโมโซมเลขจำนวนจริง		
	ค่าน้อยสุด	ค่าเฉลี่ย	ค่ามากที่สุด	ค่าน้อยสุด	ค่าเฉลี่ย	ค่ามากที่สุด
f_1	1,474	1,666.9	2,068	10,039	21,276.3	28,621
f_2	1,683	1,855.0	2,101	2,679	3,212.6	4,048
f_3	1,395	1,563.6	1,825	3,661	7,956.6	11,182
f_4	951	1,104.6	1,198	874	945.8	1,022
f_5	1,111	1,212.6	1,377	3,760	4,138.9	4,599
f_6	6,315	9,193.8	13,207	5,460	6,967.4	9,371
f_7	6,029	7,117.1	11,197	6,186	7,201.5	8,465

ตารางที่ 3 ค่าวัดผลกระทบเฉลี่ยของคำตอบที่ดีที่สุดที่ได้ของกรณีโครโมโซมเลขฐานสอง

ปัญหา	ค่าสถิติ	คำตอบที่แท้จริง	5 บิต	10 บิต	15 บิต	20 บิต	30 บิต	ICL
f_1	Average	0.0000	156.0874	33.9363	35.0067	35.9250	<u>32.3744</u>	11.7743
	SD		0.0000	12.0001	14.9473	13.1390	10.7347	6.5501
f_2	Average	0.0000	5.5914	0.9319	0.5453	<u>0.5423</u>	0.5408	0.5539
	SD		3.1621	0.1535	0.0505	0.0730	0.0699	0.0371
f_3	Average	0.0000	<u>2.8563</u>	92.3156	89.5969	84.6194	103.4911	0.4994
	SD		0.0000	42.3517	35.5120	29.9107	45.2471	0.3404
f_4	Average	0.0000	0.0085	0.0096	0.0100	0.0100	0.0106	<u>0.0095</u>
	SD		0.0025	0.0029	0.0029	0.0022	0.0037	0.0009
f_5	Average	0.0000	3.8096	0.8188	<u>0.8130</u>	0.8252	0.8365	0.0293
	SD		0.0000	0.0954	0.0807	0.0890	0.0794	0.0144
f_6	Average	1.7249	1.7956	1.7847	<u>1.7778</u>	1.8337	1.7960	1.7639
	SD		0.0000	0.0427	0.0244	0.0479	0.0341	0.0152
f_7	Average	0.01267	0.12188	0.01318	<u>0.01315</u>	0.01301	0.01339	0.01279
	SD		0.00000	0.00018	0.00035	0.00021	0.00038	0.00019

ตารางที่ 4 ค่าวัดผลกระทบเฉลี่ยของคำตอบที่ดีที่สุดที่ได้ของกรณีโครโมโซมเลขจำนวนจริง

ปัญหา	ค่าสถิติ	คำตอบที่แท้จริง	5 บิต	10 บิต	15 บิต	20 บิต	30 บิต	ICL
f_1	Average	0.0000	156.0874	0.5154	0.1346	<u>0.1261</u>	0.1486	0.0705
	SD		0.0000	0.5110	0.3987	0.1461	0.1565	0.0323
f_2	Average	0.0000	100.0000	0.7625	0.1790	0.1006	0.1064	<u>0.1038</u>
	SD		0.0000	0.1215	0.0432	0.0315	0.0356	0.0236
f_3	Average	0.0000	2.8563	45.1921	36.9810	<u>32.9629</u>	34.1726	0.0004
	SD		0.0000	31.5039	24.3074	23.5132	21.6764	0.0007
f_4	Average	0.0000	0.0116	<u>0.0100</u>	0.0107	0.0104	0.0100	0.0097
	SD		0.0037	0.0030	0.0032	0.0031	0.0025	0.0020
f_5	Average	0.0000	3.8096	0.3682	0.0111	<u>0.0071</u>	0.0079	0.0048
	SD		0.0000	0.0141	0.0095	0.0060	0.0067	0.0021
f_6	Average	1.7249	<u>1.8373</u>	1.9235	2.4285	2.2738	2.2422	1.7356
	SD		0.0925	0.3883	0.5987	0.6885	0.5545	0.0061
f_7	Average	0.01267	0.12188	<u>0.01267</u>	0.01268	0.01268	0.01267	0.01267
	SD		0.00000	0.00000	0.00000	0.00000	0.00000	0.00000

สรุปผลการวิจัย

บทความนี้นำเสนอการปรับเพิ่มความยาวโครโมโซมให้ขั้นตอนวิธีเชิงพันธุกรรมสำหรับปัญหาการหาค่าเหมาะที่สุดวัตถุประสงค์เดียว โดยมีเป้าหมายเพื่อปรับปรุงคำตอบให้ดีขึ้นจากผลการจำลองพบว่าคำตอบที่ได้จากการปรับเพิ่มความยาวโครโมโซม (*ICL*) ดีกว่าคำตอบที่ได้จากการกำหนดความยาวโครโมโซมคงที่ (*FCL*) ทั้งกรณีโครโมโซมเลขฐานสองและเลขจำนวนจริง รวมถึงคำตอบจาก *ICL* ให้ค่าวัตถุประสงค์ค่อนข้างใกล้เคียงกับค่าวัตถุประสงค์ของคำตอบที่แท้จริง ดังนั้นการปรับเพิ่มความยาวโครโมโซมซึ่งเป็นแนวคิดที่ลดขนาดของปริภูมิค้นหาในตอนแรกของการหาค่าตอบแล้วค่อยเพิ่มขนาดขึ้นสามารถปรับปรุงประสิทธิภาพการหาค่าตอบของขั้นตอนวิธีเชิงพันธุกรรมวัตถุประสงค์เดียวได้ และยังใช้ได้ทั้งกับโครโมโซมเลขฐานสองและจำนวนจริง แนวคิดนี้จึงน่าจะสามารถนำไปประยุกต์ใช้กับขั้นตอนวิธีเชิงวิวัฒนาการตัวอื่นได้ โดยไม่ต้องไปเปลี่ยนกระบวนการหาค่าตอบของขั้นตอนวิธีนั้น

กิตติกรรมประกาศ

งานวิจัยนี้ได้รับทุนอุดหนุนการวิจัยและพัฒนา จากคณะวิศวกรรมศาสตร์ มหาวิทยาลัยบูรพา เลขที่ 14/2555 ผู้วิจัยใคร่ขอขอบพระคุณผู้สนับสนุนทุนวิจัยเป็นอย่างสูง

เอกสารอ้างอิง

- Boonlong, K., Maneeratana, K., & Chaiyaratana, N. (2010). Progressive Refinement Solution Search for Multi-objective Continuum Topology Optimization Problems. In *The 24th conference of the Mechanical Engineering Network of Thailand*, Ubon Ratchathani, Thailand.
- Cagnina, L. C., Esquivel, S. C., & Coello, C. A. C. (2008). Solving Engineering Optimization Problems with the Simple Constrained Particle Swarm Optimizer. *Informatica*, 32, 319-326.
- Ciornei, I., & Kyriakides, E. (2012). A GA-API Solution for the Economic Dispatch of Generation in Power System Operation. *IEEE Transactions on Power Systems*, 27(1), 233-242.
- Deb, K. (1997). Mechanical component design using genetic algorithms. In Dasgupta, D., & Michalewicz, Z. (eds.), *Evolutionary Algorithms in Engineering Applications*, 495-512. New York : Springer.
- Deb, K., & Agrawal, R. B. (1995). Simulated binary crossover for continuous search space. *Complex Systems*, 9(2), 115-148.
- Gonzalez-Longatt, F. M., Wall, P., Regulski, P., & Terzija, V. (2012). Optimal Electric Network Design for a Large Offshore Wind Farm Based on a Modified Genetic Algorithm Approach. *IEEE Systems Journal*, 6(1), 164-172.
- Kim, I. Y., & de Weck, O. (2005). Variable chromosome length genetic algorithm for progressive refinement in topology optimization. *Structural and Multidisciplinary Optimization*. 29(6), 445-456.
- Liang, Y., & Leung, K. W. (2011). Genetic Algorithm with adaptive elitist-population strategies for multimodal function optimization, *Applied Soft Computing*, 11(2), 2017-2034.
- Ming-Shyan, H., Hsu, Y. L., & Fung, R. F. (2012). Minimum-Energy Point-to-Point Trajectory Planning for a Motor-Toggle Servomechanism. *IEEE/ASME Transactions on Mechatronics*, 17(2), 337-344.
- Shi, L., Deg, Y., Sun, H., Wang, R., Ai, J., & Yan, H. (2012). An Improved Real-Coded Genetic Algorithm for the Beam Forming of Spaceborne SAR. *IEEE Transactions on Antennas and Propagation*, 60(6), 3034-3040.
- Shimodaira, H. (1997). DCGA: A Diversity Control Oriented Genetic Algorithm. In *The Second International Conference on Genetic Algorithms in Engineering Systems: Innovation and Application*, (pp. 444-449). UK:Glasgow.
- Stach, W., Pedrycz, W., & Kurgan, L.A. (2012). Learning of Fuzzy Cognitive Maps Using Density Estimate. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 42(3), 900-912.

Tessema, B., & Yen, G. G. (2006) A Self Adaptive Penalty Function Based Algorithm for Constrained Optimization. In *IEEE Congress on Evolutionary Computation*, Sheraton Vancouver Wall Centre Hotel, Vancouver, BC, Canada.

Woldesenbet, Y.G., Yen, G.G., & Tessema, B.G. (2009) Constraint Handling in Multiobjective Evolutionary Optimization. *IEEE Transactions on Evolutionary Computation*, 13(3), 514-525.

Yao, X., Liu, Y., & Lin, G. (1999) Evolutionary Programming Made Faster. *IEEE Transactions on Evolutionary Computation*, 3(2), 82-102.

มหาวิทยาลัยบูรพา
Burapha University