

การหาค่าที่เหมาะสมที่สุดด้วยขั้นตอนวิธีเคออสมีมติก แบบหลายวัตถุประสงค์

Optimization by Multi-Objective Chaos-Memetic Algorithm

ภรณ์ยา อัมมฤครัตน์ (Paranya Ammaruekarat)* และพยุ่ง มีสัจ (Phayung Meesad)*

บทคัดย่อ

บทความวิจัยนี้เป็นการนำเสนอวิธีการหาค่าที่เหมาะสมด้วยขั้นตอนวิธีเคออสมีมติกอัลกอริทึมแบบหลายวัตถุประสงค์ (Multi-Objective Chaos-Memetic Algorithms: MOCMA) เป็นการปรับปรุงมีมติกอัลกอริทึมด้วยทฤษฎีความอลวน (Chaos Theory) ในการปรับปรุงประชากรคำตอบให้ได้คำตอบที่ดีขึ้น ซึ่งในงานวิจัยนี้ได้นำขั้นตอนวิธีที่นำเสนอไปทดสอบกับปัญหาสมการแบบ 2 วัตถุประสงค์จากชุดข้อมูล DTLZ1-4 แล้วนำผลเฉลยที่ได้จากการทดสอบมาวัดสมรรถนะของกลุ่มคำตอบที่ดีที่สุด 2 ด้าน คือ ตัววัดสมรรถนะในด้านการเข้าสู่กลุ่มคำตอบที่แท้จริง และการกระจายของกลุ่มคำตอบที่หาได้ และได้ทำการเปรียบเทียบสมรรถนะ MOCMA กับ NSGAII (Non-dominated Sorting Genetic Algorithm II) พบว่า ตัววัดสมรรถนะของคำตอบด้านการเข้าสู่คำตอบที่แท้จริงของ MOCMA จะมีอัตราการเข้าสู่คำตอบที่แท้จริงดีกว่า NSGAII และในด้านการวัดสมรรถนะของคำตอบด้านการกระจายของกลุ่มคำตอบ MOCMA มีการกระจายที่สม่ำเสมอมากกว่า NSGAII ซึ่งจะเห็น MOCMA มีค่าเข้าใกล้ศูนย์ มากกว่า NSGAII

คำสำคัญ: มีมติกอัลกอริทึมแบบหลายวัตถุประสงค์ ทฤษฎีความอลวน

Abstract

In this research, we discuss the efficacy improvement for Multi-Objective Memetic Algorithms: MOMA by using Chaos theory. Chaos theory is used to improve the population

to achieve a better result. The technique can be referred to in abbreviation as MOCMA: Multi-Objective Chaos Memetic Algorithms. In this research, such technique is applied to solve Multi-Objective equation: 2 objective DTLZ1-4. The resulted outcomes will then be measured for their capabilities to find out the best outcome group in 2 aspects namely; the capability to converge to the true outcome and the capability to spread the outcome groups found. The capabilities of the technique will be compared with those of NSGAII (Non-dominated Sorting Genetic Algorithm II). The research shows that the Convergence Measurement of MOCMA process has a better rate of convergence than that of NSGAII and Measurement of MOCMA is scattered more evenly than that of NSGAII. It can be seen that the value of MOCMA's capability measurement in both aspects is closer to 0 than those of NSGAII.

Keyword: Multi-Objective Memetic Algorithms, Chaos Theory.

1. บทนำ

หลายๆ ปัญหาในโลกแห่งความจริงไม่ว่าจะเป็นทางด้านอุตสาหกรรมหรือแวดวงวิชาการนั้นจะเป็นปัญหาแบบหลายวัตถุประสงค์ และแต่ละวัตถุประสงค์ก็มักมีความขัดแย้งกันอยู่ภายใต้เงื่อนไขเฉพาะที่ต้องการคำตอบหลายๆ คำตอบ แต่ละคำตอบอาจจะเป็นคำตอบที่เป็นคู่แข่งกัน ซึ่งรูปแบบคำตอบที่ดีที่สุดของปัญหานั้นๆ จะมีลักษณะพิเศษ คือ

* ภาควิชาเทคโนโลยีสารสนเทศ คณะเทคโนโลยีสารสนเทศ มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ

เป็นกลุ่มคำตอบที่ดีที่สุด โดยที่คำตอบแต่ละคำตอบนั้นเป็นคำตอบที่ดีที่สุดของปัญหา และไม่มีคำตอบอื่นครอบงำ (Dominate) เลย หรือที่เรียกว่าพาเรโตฟรอนต์ (Pareto front)

งานวิจัยที่นำเสนออัลกอริธึมด้านวิวัฒนาการ (Evolutionary Algorithm) ที่ถูกพัฒนาขึ้นเพื่อแก้ปัญหาหลายวัตถุประสงค์ โดยในช่วงเริ่มแรก Multi-Objective Optimization เป็นที่สนใจจากนักวิจัยเพิ่มขึ้น ไม่ว่าจะเป็นการแนะนำและปรับปรุงอัลกอริธึมด้านวิวัฒนาการ จนได้กลายเป็นเทคโนโลยีที่สำคัญของการแก้ปัญหา Multi-Objective Optimization Problem ที่รู้จักกันทั่วไป คือ Vector Evolution Genetic Algorithm (VEGA) [1] Multi-Objective Genetic Algorithm (MOGA) [2] และ Non-dominated Sorting Genetic Algorithm (NSGA) [3] หลังจากนั้นได้มีการนำเสนอ Niched Pareto Genetic Algorithm (NPGA) [4], [5] Pareto Achieved Evolution Strategy (PAES) [6] Strength Pareto Evolutionary Algorithm (SPEA) [7] Particle Swarm Optimizer (PSO) [8] เป็นต้น นอกจากวิธีการที่กล่าวมานี้ยังคงมีอีกหลายวิธีที่ใช้งานในลักษณะเดียวกัน หรือแม้กระทั่งการปรับปรุงวิธีการเดิมเพื่อเพิ่มสมรรถนะ เช่น SPEA2 [9] และ NSGAII [10] เป็นต้น

Memetic Algorithm (MA) เป็นอีกอัลกอริธึมหนึ่งด้านวิวัฒนาการที่ได้รับความนิยมมากจากนักวิจัยที่นำไปประยุกต์แก้ปัญหาดังกล่าวต่าง ๆ ดังปรากฏในงานวิจัยเช่น [11], [12], [13] แต่การศึกษาและพัฒนา Memetic Algorithm สำหรับ Multi-Objective Optimization ยังมีอยู่น้อย [14] ซึ่ง Memetic Algorithm ที่ได้พัฒนาสำหรับ Multi-Objective ในเริ่มแรกเรียกว่า Multi-Objective Genetic Local Search (MOGLS) เป็นอัลกอริธึมที่นำเสนอโดย Ishibuchi และ Murata ในปี 1996 [15] ต่อมา Jaszkiewicz [16] ได้นำเสนอ MOGLS ที่มีความสามารถในการค้นหาที่สูงกว่า และได้มีการพัฒนา Multi-Objective Memetic Algorithm (MOMA) อื่นๆ ที่ประสิทธิภาพสูงขึ้น เช่น Memetic Pareto Archived Evolution Strategy (M-PAES) ซึ่งถูกนำเสนอโดย Knowles และ Corne [17] เป็นต้น

ในส่วนของการศึกษาและพัฒนาเพื่อเพิ่มประสิทธิภาพให้กับ MOMA นั้นได้มีการศึกษา ดังปรากฏในงานวิจัย เช่น [18], [19] ซึ่งผลจากงานวิจัยที่ผ่านมา พบว่าสามารถเพิ่มประสิทธิภาพของการประมาณค่าคำตอบที่ดีที่สุดของ

Memetic Algorithm ได้ดีขึ้น แต่ยังคงมีข้อเสียในเรื่องของการใช้เวลาในการค้นหานั้นเกินกว่าเดิมและบางอัลกอริธึมไม่เหมาะกับปัญหาที่มีตัวแปรหลายตัว

ดังนั้นขอบเขตของงานวิจัยนี้จึงมุ่งเน้นที่จะศึกษาการเพิ่มประสิทธิภาพของ MOMA โดยการใส่ทฤษฎีความอลวน ในการปรับปรุงประชากรคำตอบให้เกิดคำตอบที่ดีกว่า ซึ่งความอลวนมีความสามารถในการหาค่าที่ดีที่สุดของฟังก์ชันหนึ่งให้ได้ค่าที่ดีที่สุดที่แท้จริง (Global optimum) ได้ง่ายขึ้น เนื่องจากความอลวนสามารถช่วยให้หลีกเลี่ยงการได้ค่าที่ดีที่สุดเฉพาะบริเวณ (Local optimum) ได้ เพื่อปรับปรุงให้เกิดการค้นหาที่ให้คำตอบที่ดีกว่า และเพื่อให้เหมาะกับปัญหาที่มีหลายตัวแปร โดยได้ทำการทดลองเปรียบเทียบสมรรถนะของอัลกอริธึมนี้กับ NSGAII

2. ทบทวนวรรณกรรม

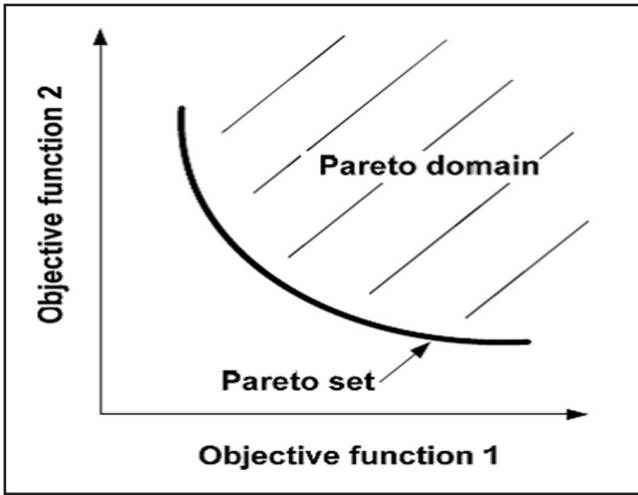
ขั้นตอนวิธีเชิงวิวัฒนาการ เป็นวิธีการที่ได้รับความนิยมในการนำไปแก้ปัญหามีหลายวัตถุประสงค์ ลักษณะของขั้นตอนวิธีเชิงวิวัฒนาการ [20] จะทำการลอกเลียนแบบการพัฒนาตัวเองตามธรรมชาติ เช่น การที่มดพยายามหาเส้นทางที่สั้นที่สุดไปสู่แหล่งอาหารที่เรียกว่า Ant-Colony Optimization (ACO) หรือ การที่นกพยายามหาจุดหมายปลายทางระหว่างกรอพยพ ที่เรียกว่า Particle Swarm Optimization (PSO) เป็นต้น

ปัญหาแบบหลายวัตถุประสงค์ [21] เป็นปัญหาการออกแบบที่มีหลายวัตถุประสงค์จะประกอบด้วย m วัตถุประสงค์ และตัวแปรตัดสินใจ (Decision Variables) ตัวเขียนอยู่ในรูปทั่วไปดังสมการที่ (1)

$$\text{Minimize (or maximize) : } \{f_1(x), f_2(x), \dots, f_m(x)\} \quad (1)$$

เมื่อ x คือ เวกเตอร์ของตัวแปรตัดสินใจ $f_i(x)$ คือ ฟังก์ชันวัตถุประสงค์ที่ $i, i = 1, 2, \dots, m$ ผลลัพธ์ที่ได้จากสมการเรียกว่าพาเรโตฟรอนต์ หรือ เซตของผลเฉลยที่ไม่ถูกครอบงำแสดงในภาพที่ 1

Multi-Objective Memetic Algorithm หรือ MOMA เป็นอัลกอริธึมด้านวิวัฒนาการอีกชนิดหนึ่งที่นิยมนำมาแก้ปัญหาหลายวัตถุประสงค์ ซึ่ง Memetic Algorithm ถูกคิดค้นโดย Merz และ Freisleben [23] และมีลักษณะคล้ายกับ Genetic Algorithm แต่มีลักษณะพิเศษกว่าตรงที่โครโมโซมสามารถ



ภาพที่ 1 พาราโดฟรอนของฟังก์ชันวัตถุประสงค์ f_1 และ f_2 [22]

เพิ่มประสบการณ์ผ่านการค้นหาแบบเฉพาะที่เพื่อเพิ่มประสิทธิภาพก่อนที่จะผ่านขบวนการทางพันธุกรรม

ทฤษฎีความอลวน (Chaos Theory) [24] เป็นทฤษฎีที่อธิบายถึง ลักษณะพฤติกรรมของระบบพลวัต (คือ ระบบที่มีการเปลี่ยนแปลง เช่น เปลี่ยนแปลงตามเวลาที่เปลี่ยนไป) โดยลักษณะการเปลี่ยนแปลงของระบบที่เรียกว่าอลวนนี้ จะมีลักษณะที่ปั่นป่วนจนดูคล้ายว่า การเปลี่ยนแปลงนั้นเป็นแบบสุ่มหรือไร้ระเบียบ (random/stochastic) แต่จริงๆ แล้วระบบอลวนนี้เป็นระบบแบบไม่สุ่ม หรือระบบที่มีระเบียบ (deterministic) ตัวอย่างของระบบที่แสดงความอลวน คือ เครื่องสร้างเลขสุ่มเทียม (psuedo-random number generator) ในเครื่องคอมพิวเตอร์จากงานจำลองสถานการณ์จริง (simulation) การที่คอมพิวเตอร์สามารถสร้างเลขสุ่ม (random number) ซึ่งอาจดูเหมือนการเกิดของตัวเลขสุ่มไม่มีแบบแผน เพราะเป็นเพียงเลขสุ่มเทียม (psuedo-random number) ซึ่งต่างจากเลขสุ่มแท้ที่เกิดจากการทอดลูกเต๋า เพราะเลขสุ่มของคอมพิวเตอร์เกิดขึ้นจากโปรแกรมง่ายๆ ระบบที่แสดงความอลวนจะต้องประกอบไปด้วยลักษณะดังต่อไปนี้

- 1) มีคุณสมบัติแบบไม่เป็นเชิงเส้น (nonlinearity)
- 2) ไม่ใช่เกิดแบบสุ่ม (deterministic ไม่ใช่ probabilistic) หรือเรียกว่าในระบบอลวน เหตุการณ์ทั้งหลายมักเกิดขึ้นภายใต้กฎเกณฑ์ที่แน่นอนตายตัว เพื่อป้องกันความสับสนระหว่าง “ความอลวน” และ “การสุ่ม” จึงเรียก chaos ว่า deterministic chaos
- 3) ไวต่อสภาวะเริ่มต้น (sensitivity to initial conditions)

คือ การเริ่มต้นที่ต่างกันเพียงนิดเดียวอาจส่งผลให้มันปลายต่างกันมาก

4) ไม่สามารถทำนายล่วงหน้าในระยะยาวได้ (long-term prediction is impossible)

ทฤษฎีความอลวนมีประโยชน์ คือใช้ในการวิเคราะห์ระบบและทำนายอนาคต และใช้ในการควบคุม-สร้างความเสถียรให้กับระบบ สามารถใช้หลักการของทฤษฎีความอลวนช่วยให้การหาค่าที่ดีที่สุด ของฟังก์ชันหนึ่งได้ค่าที่ดีที่สุดที่แท้จริง (global optimum) ได้ง่ายขึ้น เพราะความอลวนสามารถช่วยให้หลีกเลี่ยงการได้ค่าที่ดีที่สุดเฉพาะบริเวณ (local optimum) ได้

ความอลวนเป็นปรากฏการณ์ทั่วไปในระบบไม่เป็นเชิงเส้น (nonlinear system) ความอลวนนำมาใช้กับด้านการเพิ่มประสิทธิภาพระบบไม่เป็นเชิงเส้น โดยมีสมการดังนี้:

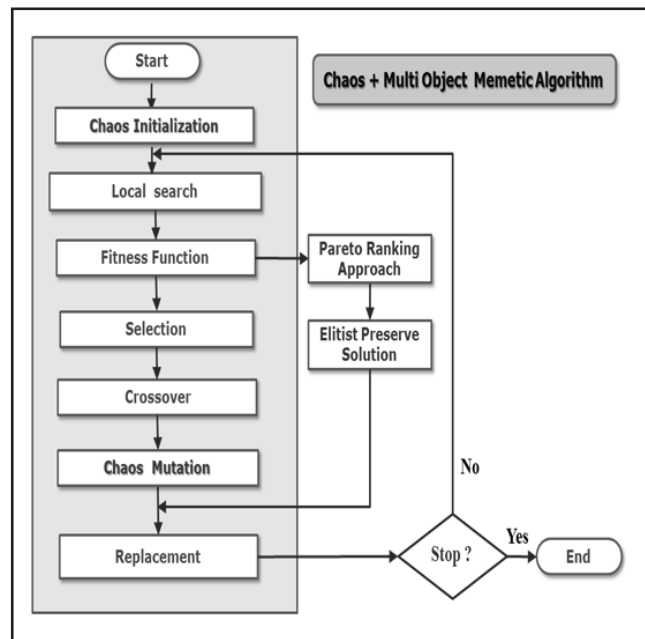
$$Z^{k+1} = \mu z^k (1 - z^k), k = 0, 1, \dots (2)$$

โดยที่ $z^{k+1} \in [0,1]$ คือ chaos variable ในการทำซ้ำจำนวน k ครั้ง

เมื่อ $\mu = 4$, $Z^0 \neq \{0, 0.25, 0.5, 0.75, 1\}$

3. ขั้นตอนการทำงาน

ขั้นตอนการทำงานของมีมิติเคออสที่มีใช้ในการวิจัย ดังภาพที่ 2



ภาพที่ 2 Multi-Objective Chaos Memetic Algorithm

จากขั้นตอนการทำงานของงานวิจัยนี้จะนำเสนอถึงเทคนิคใหม่ในการเพิ่มประสิทธิภาพของ Multi-Objective Memetic Algorithm ในส่วนของ Initialization Procedure และ Mutation Procedure ซึ่งได้ใช้ ทฤษฎีความอลวน ในการปรับปรุงประชากรคำตอบ โดยมีรายละเอียด ดังนี้

3.1 สร้างโครโมโซมเริ่มต้นด้วยความอลวน (Chaos Initialization)

โครโมโซมเริ่มต้นถูกสร้างขึ้นโดยลำดับความอลวนที่มีลักษณะของการแพร่กระจายที่มากกว่าการสุ่ม ความไวต่อสภาวะเริ่มต้น และข้อมูลซึ่งกระจายอยู่ในพื้นที่แก้ปัญหาอย่างสม่ำเสมอ ดังนั้นขั้นตอนวิธีนี้สามารถเอาชนะข้อจำกัดของการสุ่มที่เข้าซ้อนได้ ซึ่งโครโมโซมจะถูกสร้างขึ้นตามสมการที่ (3)

$$CsC = \{x_{ij}(c_j)\} = \begin{bmatrix} x_{11}(c_1) & x_{12}(c_1) & \cdots & x_{1l}(c_1) \\ x_{21}(c_2) & x_{22}(c_2) & \cdots & x_{2l}(c_2) \\ \vdots & \vdots & & \vdots \\ x_{n1}(c_1) & x_{n2}(c_1) & \cdots & x_{nl}(c_n) \end{bmatrix} \quad (3)$$

โดยที่ $x_{ij}(c_j)$ คือ โครโมโซมที่สร้างขึ้นด้วยความอลวน

$i = 1, 2, \dots, n$ n คือ จำนวนสูงสุดของ generation

$j = 1, 2, \dots, l$ l คือ จำนวนของโครโมโซม

c_j คือ ค่าเริ่มต้นจากสมการ (2)

จากข้างต้น โครโมโซม CsC ถูกสร้างขึ้นโดย c ซึ่งเป็นค่าเริ่มต้นสำหรับการจับคู่ของสมการ โครโมโซมเริ่มต้นที่ได้กำหนดในทุกุ่นจะแตกต่างกัน ซึ่งสามารถลดความซ้ำซ้อนของโครโมโซม หลังจากที่มีการสร้างโครโมโซม จะได้ว่าโครโมโซมของทุกุ่นมีความซ้ำซ้อนของข้อมูลที่ลดลง

3.2 การกลายพันธุ์แบบอลวน (Chaos Mutation)

การดำเนินการกลายพันธุ์ถูกใช้เพื่อสร้างโครโมโซมใหม่ ซึ่งการดำเนินการกลายพันธุ์นี้ได้ปรับตัวขึ้นตามกลไกของความอลวน ซึ่งใช้ความอลวนร่วมกันกับกระบวนการของการกลายพันธุ์ ซึ่งเรียกว่า chaos mutation

ขั้นแรก โครโมโซม Cm นำเข้าสู่สมการ (4) ที่มีค่าความอลวนเริ่มต้น x^0 ขั้นที่สอง x^0 ถูกวนรอบซ้ำ k ครั้งและจะได้เวกเตอร์ความอลวน x^k ขั้นที่สาม x^k ของการกลายพันธุ์ขึ้นอยู่กับสมการที่ (3) และจะได้เวกเตอร์การกลายพันธุ์แบบอลวน x^{k+1} ขั้นสุดท้าย ใส่รวมเข้าไปในระบบจะได้การกลายพันธุ์ของโครโมโซม Cm'.

$$x^{k+1} = \hat{x}(1 - x^k), \mu z^k \quad (4)$$

โดยที่ $\hat{x}$ คือโซลูชันที่ดีที่สุดในปัจจุบัน จากสมการ (4) สามารถพบได้ว่า ความอลวนมีลักษณะความไวต่อสภาวะเริ่มต้นความแตกต่างเพียงเล็กน้อยของ x^0 สามารถทำให้เกิดความแตกต่างที่สูงของ x^k ลักษณะนี้สามารถขยายช่วงการกลายพันธุ์และเพิ่มประสิทธิภาพการกลายพันธุ์

4. ตัวชี้วัดสมรรถนะของกลุ่มคำตอบ

ตัวชี้วัดสมรรถนะเพื่อรักษาเป้าหมายที่สำคัญสองด้านของปัญหาการหาค่าที่เหมาะสมที่สุดที่มีหลายวัตถุประสงค์ คือ การเข้าสู่กลุ่มคำตอบที่แท้จริง และการรักษาความหลากหลายภายในกลุ่มคำตอบที่หาได้ [3] จึงใช้ตัวชี้วัดสมรรถนะดังนี้

4.1 การวัดสมรรถนะของคำตอบด้านการเข้าสู่กลุ่มคำตอบที่แท้จริง (Convergence Measurement)

เป็นตัวชี้วัดการเข้าสู่ของกลุ่มคำตอบโดยเปรียบเทียบระยะทางจากสมาชิกของกลุ่มคำตอบทุกคำตอบที่หาได้ (Obtained Solution Set) กับกลุ่มคำตอบที่แท้จริง (True Pareto Set) ดังสมการที่ (5) และ (6)

$$convergence(A) = \frac{\sum_{i=1}^{|A^*|} dt_i}{|A^*|} \quad (5)$$

$$dt_i = \min_{j=1}^{|A^*|} \sqrt{\sum_{k=1}^K \left[\frac{f_k(x) - f_k(y)}{f_k^{\max} - f_k^{\min}} \right]^2} \quad (6)$$

เมื่อ A^* คือ เซตคำตอบที่แท้จริง $f_k^{\min}$ และ $f_k^{\max}$ คือ ค่าฟังก์ชันวัตถุประสงค์ k^{th} ที่มีค่ามากที่สุดและน้อยที่สุด ตามลำดับ k คือ จำนวนฟังก์ชันวัตถุประสงค์ และ $|A^*|$ คือ จำนวนคำตอบที่แท้จริงในเซต A

ถ้าค่าตัวชี้วัดสมรรถนะชนิดนี้มีค่าเข้าใกล้ศูนย์ ถือว่ากลุ่มคำตอบอัลกอริธึมนั้นเป็นกลุ่มคำตอบที่เข้าสู่ใกล้กลุ่มคำตอบที่แท้จริง

4.2 การวัดสมรรถนะของคำตอบด้านการกระจายของกลุ่มคำตอบที่หาได้ (Spread Measurement)

เป็นตัวชี้วัดที่บอกการกระจายของกลุ่มคำตอบดังสมการที่ (7) และ (8)

$$spread(A) = \frac{sd_f + sd_l + \sum_{i=1}^{|A|-1} \|sd_i - \overline{sd}\|}{sd_f + sd_l + (|A|-1)sd} \quad (7)$$

$$sd_i = \sqrt{\sum_{k=1}^K \left[\frac{f_k(x_i) - f_k(x_{i+1})}{f_k^{\max} - f_k^{\min}} \right]^2} \quad (8)$$

เมื่อ sd_f และ sd_l เป็นระยะห่างของคำตอบปลายสุดทั้งสองด้าน sd_l เป็นระยะของคำตอบที่อยู่ต่อเนื่องกันในชุดคำตอบที่ดีที่สุดที่หาได้ $|A|$ เป็นจำนวนคำตอบที่หาได้ sd คือค่าเฉลี่ยของระยะทาง sd_i

ถ้าค่าตัวสมรรถนะชนิดนี้ มีค่าเข้าใกล้ศูนย์ ถือว่ากลุ่มคำตอบอัลกอริธึมนั้นจะมีคำตอบที่มีลักษณะการกระจายที่มีลักษณะสม่ำเสมอตลอดเส้นขอบเขตของกลุ่มคำตอบที่ดีที่สุด

5. ผลการทดลอง

การทดสอบ MOCMA ที่ได้พัฒนาออกแบบขึ้นมาใหม่ โดยทำการเปรียบเทียบสมรรถนะของคำตอบกับ NSGAII

ปัญหาหลายวัตถุประสงค์ที่นำมาทดสอบในงานวิจัยนี้คือ ชุดปัญหา DTLZ (Suite of continuous test problems by Deb, Thiele, Laumanns, Zitzler) ซึ่งเป็นปัญหาที่นิยมใช้ในการทดสอบ Multi-Objective และสามารถปรับขนาดของฟังก์ชันวัตถุประสงค์ได้ โดยได้เลือกทำการทดสอบกับ DTLZ1-4 แบบ 2 วัตถุประสงค์ซึ่งรายละเอียดของปัญหาดังปรากฏใน ตารางที่ 1

ตารางที่ 1 รายละเอียดปัญหา DTLZ1-4 [25]

DTLZ	Var.	Objective functions
1	6	$f_1 = \frac{1}{2} x_1 (1+g)$ $f_2 = \frac{1}{2} (1-x_1)(1+g)$ $g = 100[5 + \sum_{i=2}^6 ((x_i - 0.5)^2 - \cos(20\pi(x_i - 0.5)))]$ $x_1 \in [0, 1], i \in \{1, \dots, 6\}$
2	11	$f_1 = (1+g) \cos(x_1 \frac{\pi}{2})$ $f_2 = (1+g) \sin(x_1 \frac{\pi}{2})$ $g = \sum_{i=2}^{11} (x_i - 0.5)^2$ $x_1 \in [0, 1], i \in \{1, \dots, 11\}$
3	11	$f_1 = (1+g) \cos(x_1 \frac{\pi}{2})$ $f_2 = (1+g) \sin(x_1 \frac{\pi}{2})$ $g = 100[10 + \sum_{i=2}^{11} ((x_i - 0.5)^2 - \cos(20\pi(x_i - 0.5)))]$ $x_1 \in [0, 1], i \in \{1, \dots, 11\}$
4	11	$f_1 = (1+g) \cos(x_1 \frac{\pi}{2})$ $f_2 = (1+g) \sin(x_1 \frac{\pi}{2})$ $g = \sum_{i=2}^{11} (x_i - 0.5)^2$ $\infty = 100$ $x_1 \in [0, 1], i \in \{1, \dots, 11\}$

โดยกำหนดวิธีที่ใช้ในการทดลองดังนี้

- Selection : Tournament Selection
- Crossover: Modified Order Crossover
- Mutation : Chaos Mutation
- Fitness Assignment: Pareto Ranking Approach
- Local search : Insert Procedure

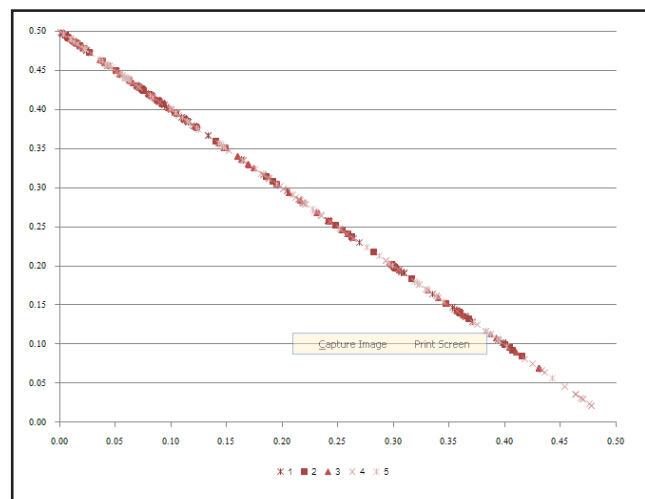
และพารามิเตอร์ที่ใช้ในการทดลอง โดยเลือกจากการทดลองที่ดีที่สุด ดังตารางที่ 2

ตารางที่ 2 พารามิเตอร์ที่ใช้ในการทดลอง

Parameter	Value
Population size	100
Number of Generation	200
Crossover probability	0.9
Mutation probability	0.8
Local search probability	0.5

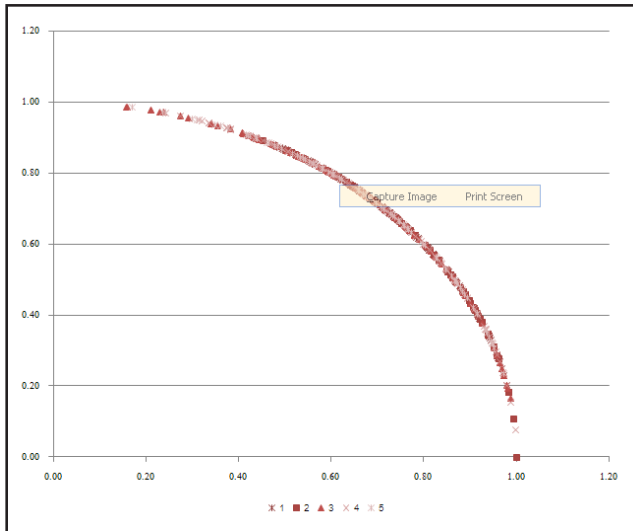
5.1 กราฟแสดงผลจากการทดสอบพารेटอฟรอน

จากการทดลองซ้ำ 5 ครั้ง ของแต่ละปัญหา จากอัลกอริธึม MOCMA ถูกนำมาเปรียบเทียบ ดังแสดงในภาพที่ 3, 4, 5 และ 6

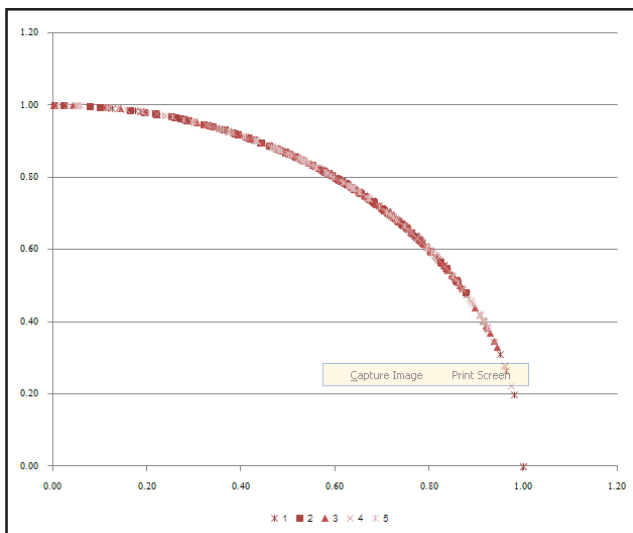


ภาพที่ 3 การเปรียบเทียบพารेटอฟรอนของ DTLZ1

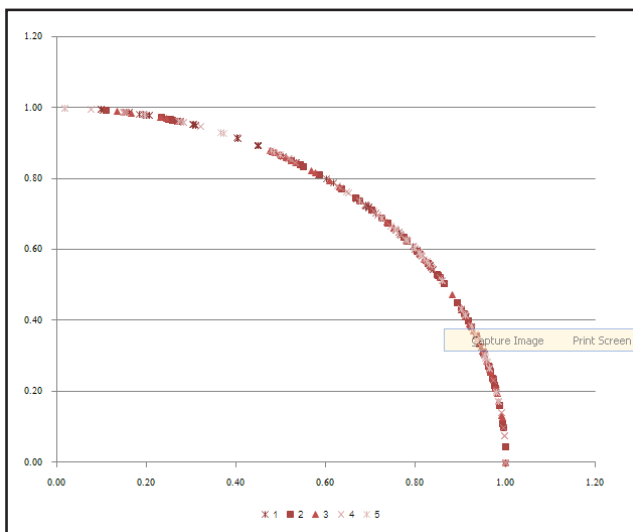
จากภาพที่ 3, 4, 5 และ 6 จะเห็นว่า จากพารेटอฟรอนของทั้ง 4 ปัญหา คือ DTLZ1-4 นั้น เมื่อทำการทดลองจากอัลกอริธึม MOCMA โดยทดลองซ้ำปัญหาละ 5 ครั้ง ซึ่งในแต่ละครั้งให้พารेटอฟรอนของทุกปัญหาไม่แตกต่างกัน แต่แตกต่างกันในด้านของการกระจายของกลุ่มคำตอบที่หาได้เล็กน้อย



ภาพที่ 4 การเปรียบเทียบพารेटอฟรอนของ DTLZ2

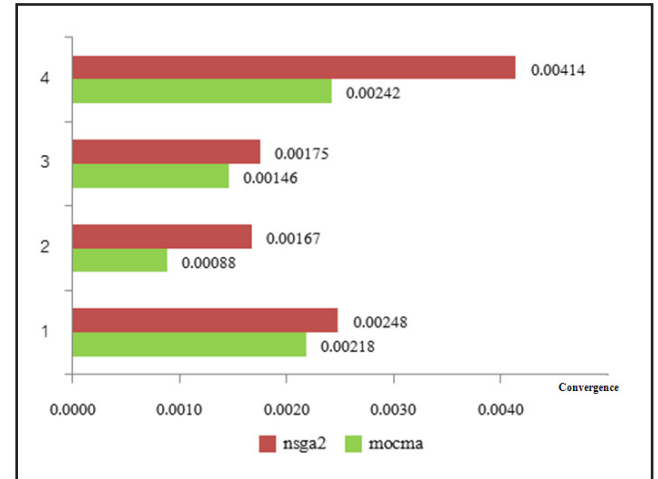


ภาพที่ 5 การเปรียบเทียบพารेटอฟรอนของ DTLZ3



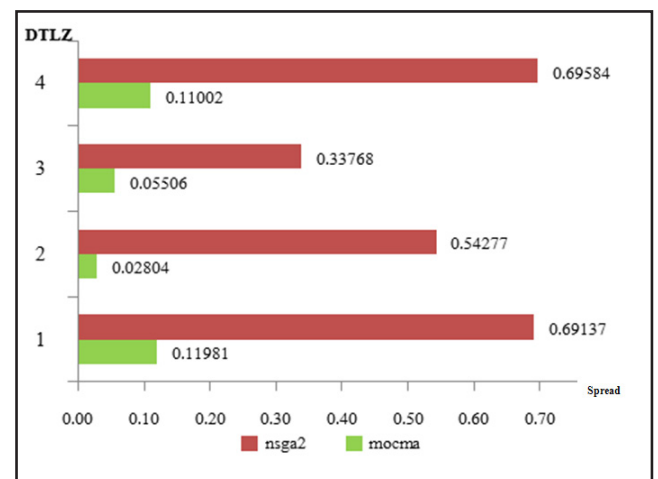
ภาพที่ 6 การเปรียบเทียบพารेटอฟรอนของ DTLZ4

5.2 ผลการเปรียบเทียบสมรรถนะของคำตอบด้านการลู่เข้าสู่กลุ่มคำตอบที่แท้จริง และการกระจายของกลุ่มคำตอบที่หาได้ โดยทำการเปรียบเทียบกับอัลกอริธึม NSGAII ดังแสดงในตารางที่ 2 และภาพที่ 7 และ 8



ภาพที่ 7 การเปรียบเทียบด้านการลู่เข้าสู่กลุ่มคำตอบที่แท้จริง

จากภาพที่ 7 แสดงค่าเฉลี่ยทางด้านการลู่เข้าสู่กลุ่มคำตอบที่แท้จริง โดยทำการเปรียบเทียบอัลกอริธึม MOCMA กับ อัลกอริธึม NSGAII พบว่า อัลกอริธึม MOCMA สามารถลู่เข้าสู่กลุ่มคำตอบที่แท้จริงได้ดีกว่าอัลกอริธึม NSGAII คือ มีค่าเข้าใกล้ศูนย์ (0) มากกว่าอัลกอริธึม NSGAII



ภาพที่ 8 การเปรียบเทียบด้านการกระจายของกลุ่มคำตอบ

จากภาพที่ 8 แสดงค่าเฉลี่ยทางด้านการกระจายของกลุ่มคำตอบที่หาได้ โดยทำการ เปรียบเทียบอัลกอริธึม MOCMA กับ NSGAII พบว่า อัลกอริธึม MOCMA สามารถกระจายของกลุ่ม

ตารางที่ 3 ผลการเปรียบเทียบสมรรถนะของคำตอบ

DTLZ	Algorithm	Performance	Convergence	Spread
1	MOCMA	mean	0.00218	0.11981
		variance	0.00093	0.09694
	NSGAII	mean	0.00248	0.69137
		variance	0.00145	0.24249
2	MOCMA	mean	0.00088	0.02804
		variance	0.00029	0.00694
	NSGAII	mean	0.00167	0.54277
		variance	0.00090	0.50476
3	MOCMA	mean	0.00146	0.05506
		variance	0.00030	0.03276
	NSGAII	mean	0.00175	0.33768
		variance	0.00072	0.43764
4	MOCMA	mean	0.00242	0.11002
		variance	0.00107	0.05978
	NSGAII	mean	0.00414	0.69584
		variance	0.00130	0.38014

คำตอบที่หาได้ดีกว่า อัลกอริธึม NSGAII คือมีค่าเข้าใกล้ศูนย์ (0) มากกว่าอัลกอริธึม NSGAII

6. บทสรุป

บทความวิจัยนี้ได้ทำการศึกษาพัฒนาประสิทธิภาพให้กับขั้นตอนวิธีที่มีมิติอัลกอริธึมแบบหลายวัตถุประสงค์ โดยการใช้ทฤษฎีความอลวนในส่วนของ Initialization Procedure และ Mutation Procedure ซึ่งได้ใช้ในการปรับปรุงประชากรคำตอบให้เกิดคำตอบที่ดีกว่าโดยนำขั้นตอนวิธีดังกล่าวไปทดสอบกับปัญหาสมการแบบหลายวัตถุประสงค์ DTLZ1-4 และนำผล เปรียบเทียบที่ได้จากการทดสอบมาวิเคราะห์หาสมรรถนะของคำตอบด้านการเข้าสู่กลุ่มคำตอบที่แท้จริง และการกระจายของกลุ่มคำตอบที่หาได้

พบว่าตัววัดสมรรถนะของคำตอบ ด้านการเข้าสู่คำตอบที่แท้จริงของอัลกอริธึม MOCMA จะมีอัตราการเข้าสู่คำตอบที่แท้จริงดีกว่าอัลกอริธึม NSGAII และในด้านการวัด

สมรรถนะของคำตอบด้านการกระจายของกลุ่มคำตอบ อัลกอริธึม MOCMA มีการกระจายที่สม่ำเสมอมากกว่าอัลกอริธึม NSGAII ซึ่งจะเห็นได้ว่าการวัดสมรรถนะทั้งสองด้าน อัลกอริธึม MOCMA มีค่าเข้าใกล้ศูนย์ มากกว่าอัลกอริธึม NSGAII และการศึกษาครั้งต่อไปจะทำการทดสอบกับชุดปัญหา DTLZ ที่กำหนดวัตถุประสงค์มากกว่านี้

7. เอกสารอ้างอิง

- [1] J. D. Schaffer, "Multiobjective Optimization with Vector Evaluated Genetic Algorithms" In *GAs and their Application: Proc. Of 1st Inter Conf. on GAs*, pp. 93-100, 1985.
- [2] C. M. Fonseca, and P. J Fleming., "Genetic Algorithms for Multiobjective Optimization: Formulation Discussion and Generalization," In: *Proc. of the 5th Inter. Conf. on GAs*, pp. 416-423, 1993.
- [3] N. Srinivas, and K. Deb, "Multiobjective Optimization Using Non-Dominated Genetic Algorithms," *Evolutionary Computation*, vol. 2(3), pp. 221-248, 1994.
- [4] J. Horn, and N. Nafpliotis, *Multiobjective Optimization Using Niche Pareto Genetic Algorithm*. Tech. Report, I11iGA1 Report 93005, UIUC, 1993.
- [5] J. Horn, N. Nafpliotis, and D. E. Goldberg., "A Niche Pareto Genetic Algorithm for Multiobjective Optimization," In: *the 1st IEEE Conf. on Evolutionary Computation*, pp. 82-87, 1994.
- [6] J.D. Knowles, and D.W. Corne., "Approximating the Non-Dominated Front Using the Pareto Archive Evolution Strategy," *Evolutionary Computation*, vol. 8(2), pp.149-172, 2000.
- [7] E. Zitzler, and L. Thiele., "Multiobjective Evolutionary algorithms: A Comparative Case Study and the Strength Pareto Approach" *Transactions on Evolutionary Computation*, vol. 3(4), pp. 257-271, 1999.
- [8] M. Reyes-Sierra, and C. A. Coello Coello., "Multi-Objective Particle Swarm Optimizers: A Survey of the State-of-the-Art" *Int.J. of Computational Intelligence*

- Research*, vol. 2(3), pp. 287-308, 2006.
- [9] E. Zitzler, M. Laumanns, and L. Thiele., "SPEA2 improving the strength pareto evolutionary algorithm for multiobjective optimization," in *Proc. Evolutionary Methods Design, Optimization Contr.*, Barcelona, Spain, pp. 95–100, 2002.
- [10] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan., "A Fast and Elitist Multiobjective Genetic Algorithm: NSGAII" *IEEE Trans. On Evolutionary Computation*, Vol. 6(2), pp. 182-197, 2002.
- [11] A. Alkan, and E. Özcan., *Memetic Algorithms for Time Tabling*, Yeditepe University, 34755 Kayisdagi-Istanbul/Turkey, 2003.
- [12] ปารเมศ ชูติมา เพ็ญพักตร์ ปิ่นภูมิรัมย์ และพัฒนพงษ์ แสงหัตถวัฒนา, "การประยุกต์ใช้เมมเมติกอัลกอริทึมสำหรับปัญหาการจัดลำดับสายการประกอบแบบผลิตภัณฑ์ผสมที่มีหลายวัตถุประสงค์ในระบบผลิตแบบทันเวลาพอดี" *การประชุมวิชาการข่ายงานวิศวกรรมอุตสาหกรรม*, หน้า 72-77, 2007.
- [13] Nguyen Quoc Viet Hung, Ta Quang Binh and Duong Tuan Anh., "A Memetic Algorithm for Timetabling" In: *Proceedings of 3rd Int ,Conf. RIVF'05 Research Informatics Vietnam-Francophony*, pp. 289-294, 2005.
- [14] H. Ishibuchi, Y. Hitotsuyanagi, and Y. N. Tsukamoto., "Implementation of Multiobjective Memetic Algorithms for Combinatorial Optimization Problems: A Knapsack Problem Case Study," *Multi-objective Memetic Algorithms*, pp. 27–49, 2009.
- [15] H. Ishibuchi, and T. Murata, "Multi-Objective Genetic Local Search Algorithm" In: *Proc. of 1996 IEEE International Conference on Evolutionary Computation*, pp. 119–124, 1996.
- [16] Jaszkiewicz, A., "Genetic Local Search for Multi-Objective Combinatorial Optimization," *European Journal of Operational Research*, 137, pp. 50–71, 2002.
- [17] J. D. Knowles, and D. Corne., "M-PAES: A Memetic Algorithm for Multiobjective Optimization," In: *Proc. of 2000 Congress on Evolutionary Computation*, pp. 325–332, 2002.
- [18] H. Duan, and X. Yu., "Hybrid Ant Colony Optimization Using Memetic Algorithm for Traveling Salesman Problem," In: *Proceedings of the IEEE Symposium on Approximate Dynamic Programming and Reinforcement Learning*, pp. 92-95, 2007.
- [19] สุภัคกานดา ชมพูมิ่ง, ประพล อิทธิพงษ์ และภูพงษ์ พงษ์เจริญ, "การประยุกต์ใช้เทคนิคซัพเฟลพรอกลิปปีงเพื่อเพิ่มประสิทธิภาพการทำงานของมีมติกอัลกอริทึม" *ORNET*, หน้า 162-174, 2007.
- [20] E. Emad, H. Tarek, and G. Donald., "Comparison among five evolutionary-based optimization algorithm," *Advanced Engineering Informatics*, pp. 43-53, 2005.
- [21] A. Konak, D.W. Coit, and E.S. Alice, "Multi-objective optimization using genetic algorithms: A tutorial". *Reliability Engineering and System Safety*, pp. 992-1007, 2006.
- [22] K. Kim, M. Wang, M. R. von Spakovsky, and D. J. Nelson., "Stochastic Modeling and Uncertainty Analysis with Multi-objective Optimization Strategies for the Synthesis/Design and Operation /Control of a PEMFC Fuel Processing Subsystem," In: *International Mechanical Engineering Congress and Exposition IMECE'2008*, ASME Paper No. IMECE2008-68065, N.Y., 2008.
- [23] P. Merz, B. Freisleben, "A genetic local search approach to the quadratic assignment problem," In: Back CT, editor. *Proceedings of the 7th international conference on genetic algorithms*, San Diego, CA: Morgan Kaufmann, pp. 465-72, 1997.
- [24] D. J. Jefferies, J. H. B. Deane and G. G. Johnstone, "An introduction to chaos," *Electronics and Communication Engineering Journal*, pp. 115-123, 1989.
- [25] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler., "Scalable Multi-Objective Optimization Test Problems," In: *CEC 2002 , IEEE Press*, pp. 825-830, 2002.