

การเพิ่มประสิทธิภาพขั้นตอนวิธี ด้านวิวัฒนาการแบบหลาย วัตถุประสงค์ด้วยทฤษฎี ความอลวน Multi-objective Evolutionary Algorithm Optimization using Chaos Theory

ภรณ์ยา ปาลวิสุทธิ (Paranya Palwisut)*

บทคัดย่อ

ขั้นตอนวิธีด้านวิวัฒนาการมีลักษณะหลายประการที่เหมาะสมสำหรับปัญหาแบบหลายวัตถุประสงค์ การประยุกต์ใช้ขั้นตอนวิธีด้านวิวัฒนาการในการหาค่าเหมาะสมที่สุดที่มีหลายวัตถุประสงค์ได้รับความสนใจจากนักวิจัยอย่างมาก และได้มีการพัฒนาประสิทธิภาพกันอย่างต่อเนื่อง ดังนั้นบทความนี้จึงได้นำเสนอแนวทางในการเพิ่มประสิทธิภาพการทำงานขั้นตอนวิธีด้านวิวัฒนาการแบบหลายวัตถุประสงค์ด้วยทฤษฎีความอลวนบนพื้นฐานลอจิสติกแมป

คำสำคัญ: ขั้นตอนวิธีด้านวิวัฒนาการแบบหลายวัตถุประสงค์
ทฤษฎีความอลวน ลอจิสติกแมป

Abstract

Evolutionary algorithms possess several characteristics that are desirable for Multi-Objective Optimization Problem. The application of evolutionary algorithms in multi-objective optimization is receiving very interest from researchers and continuous improvement. This article presents an approach to enhance performance of multi-objective evolutionary algorithms with chaos theory based on Logistic Map.

* คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยราชภัฏนครปฐม

Keywords: Multi-Objective Evolutionary Algorithms, Chaos Theory, Logistic Map.

1. บทนำ

ปัญหาต่างๆ จากหลากหลายสาขา ไม่ว่าจะเป็นทางด้านการขนส่ง ด้านการผลิต หรือด้านวิศวกรรม ทุกปัญหาล้วนมีวัตถุประสงค์ในการแก้ปัญหาเพื่อให้ได้ผลลัพธ์ที่ดีที่สุด เช่น ปัญหาด้านการผลิตสินค้าให้ได้ปริมาณมากที่สุดในเวลาที่จำกัด เป็นปัญหาที่เราต้องการให้เกิดค่าสูงสุด (Maximum) หรือปัญหาด้านต้นทุนในการผลิตสินค้าเป็นปัญหาที่เราต้องการให้เกิดค่าต่ำสุด (Minimum) เป็นต้น ปัญหาเหล่านั้นสามารถกำหนดเป็นสมการรูปแบบปัญหาการหาค่าเหมาะสมที่สุด (Optimization Problem) ซึ่งอยู่ในรูปของฟังก์ชันวัตถุประสงค์ (Objective Function) ภายใต้เงื่อนไขหรือข้อบังคับ (Constraint) เป็นการสร้างแบบจำลองปัญหาเพื่อหาโซลูชันที่น้อยที่สุดหรือมากที่สุด หรือโซลูชันที่ดีที่สุด (Optimal Solution) สิ่งที่สำคัญของการแก้ปัญหาแบบนี้คือการหาคำตอบหรือค่าที่ดีที่สุดที่แท้จริง (Global Optimum) ของปัญหานั้นๆ

การแก้ปัญหาการหาค่าเหมาะสมที่สุดมีทั้งแบบวัตถุประสงค์เดียว (Single Objective Optimization Problem) และแบบหลายวัตถุประสงค์ (Multi-Objective Optimization Problem) และส่วนใหญ่ปัญหาต่างๆ มักเป็นแบบหลายวัตถุประสงค์ ตัวอย่างปัญหาแบบหลายวัตถุประสงค์ง่ายๆ ที่พบในชีวิตประจำวัน เช่น ปัญหาในการเลือกซื้อรถยนต์ [1] หากต้องการซื้อรถยนต์ ลูกค้าย่อมจะพิจารณาในด้านของราคากับสมรรถนะของรถยนต์ สามารถสรุปวัตถุประสงค์ได้ 2 วัตถุประสงค์ คือ ต้องการราคาที่ไม่สูง ในขณะที่ต้องการสมรรถนะของรถสูง ซึ่งคำตอบที่ได้ไม่ได้มีเพียงแค่คำตอบเดียวจะมีลักษณะเป็นกลุ่มคำตอบที่ดีที่สุด เป็นต้น แต่สิ่งที่ยากที่สุดในการแก้ปัญหาการหาค่าเหมาะสมที่สุดก็คือการแยกค่าที่ดีที่สุดที่แท้จริงออกจากค่าดีที่สุดเฉพาะที่ (Local Optimum) เนื่องจากฟังก์ชันที่ต้องการหาค่าที่สูงที่สุดหรือค่าที่ต่ำที่สุดนั้น อาจจะมีหลายจุดที่คล้ายกับจุดที่สูงที่สุดหรือจุดที่ต่ำที่สุด ที่เราเรียกว่าค่าดีที่สุดเฉพาะที่นั่นเอง

ขั้นตอนวิธีด้านวิวัฒนาการ (Evolutionary Algorithms) นับเป็นวิธีหนึ่งที่ได้รับคามนิยมนำมาประยุกต์ใช้กับการแก้ปัญหาการหาค่าเหมาะสมที่สุด โดยทำการลอกเลียน

แบบการพัฒนาตัวเองตามธรรมชาติ และใช้หลักเกณฑ์ที่ต่างกันในการค้นหาแบบเฟ้นสุ่ม (Stochastic Search) [2] จากงานวิจัยที่ผ่านมาพบว่าขั้นตอนวิธีด้านวิวัฒนาการได้รับการยอมรับว่าเหมาะสมสำหรับการแก้ปัญหาการหาค่าที่เหมาะสมที่สุดที่มีหลายวัตถุประสงค์ เช่น Multi-Objective Genetic Algorithm (MOGA) [3] และ Non-dominated Sorting Genetic Algorithm (NSGA) [4] เป็นต้น และในปัจจุบันขั้นตอนวิธีด้านวิวัฒนาการยังเป็นที่สนใจจากนักวิจัยในการพัฒนาและปรับปรุงเพิ่มประสิทธิภาพอย่างต่อเนื่อง เพื่อให้ได้ผลลัพธ์ที่ดีขึ้นโดยสามารถแยกค่าที่ดีที่สุดที่แท้จริงออกจากค่าที่ดีที่สุดเฉพาะที่ได้โดยมีประสิทธิภาพและเพื่อให้ได้เทคโนโลยีที่ดีในการแก้ปัญหาการหาค่าที่เหมาะสมที่สุด

บทความนี้ได้นำเสนอแนวทางในการเพิ่มประสิทธิภาพขั้นตอนวิธีด้านวิวัฒนาการแบบหลายวัตถุประสงค์ด้วยทฤษฎีความอลวน (Chaos Theory) ซึ่งทฤษฎีความอลวนมีประโยชน์คือใช้ในการควบคุม สร้างความเสถียรให้กับระบบสามารถใช้หลักการของทฤษฎีความอลวนช่วยให้การหาค่าที่ดีที่สุดของฟังก์ชันหนึ่งให้ได้ค่าที่ดีที่สุดที่แท้จริงได้ง่ายขึ้นเพราะความอลวนสามารถช่วยให้หลีกเลี่ยงการได้ค่าที่ดีที่สุดเฉพาะที่ได้ [5] และสามารถช่วยให้หลีกเลี่ยงการได้ค่าที่ดีที่สุดเฉพาะที่ได้ดีกว่าขั้นตอนวิธีการหาค่าเหมาะสมที่สุดแบบเฟ้นสุ่ม (Stochastic Optimization) [6] จากที่ผู้เขียนบทความได้ศึกษาและได้ทำการวิจัยโดยใช้ลอจิสติกแมป (Logistic Map) เป็นตัวแทนของความอลวนมาใช้ดังเอกสารงานวิจัย [7–12] พบว่าผลลัพธ์ที่ได้เป็นในทิศทางเดียวกัน คือทฤษฎีความอลวนช่วยเพิ่มประสิทธิภาพให้กับขั้นตอนวิธีด้านวิวัฒนาการแบบหลายวัตถุประสงค์ได้เป็นอย่างดี โดยในบทความนี้ผู้เขียนจะแบ่งเนื้อหาออกเป็น 5 หัวข้อ ประกอบด้วย 1) บทนำ 2) ขั้นตอนวิธีด้านวิวัฒนาการแบบหลายวัตถุประสงค์ 3) ทฤษฎีความอลวน 4) ตัวอย่างการประยุกต์ใช้ทฤษฎีความอลวนในขั้นตอนวิธีด้านวิวัฒนาการแบบหลายวัตถุประสงค์ และ 5) สรุป

2. ขั้นตอนวิธีด้านวิวัฒนาการแบบหลายวัตถุประสงค์

2.1 ขั้นตอนวิธีด้านวิวัฒนาการ

Konak และคณะ [2] ได้กล่าวว่า ขั้นตอนวิธีด้านวิวัฒนาการมีขั้นตอนในการทำงานหลัก คือการสร้างกลุ่มของตัวค้นหาผลลัพธ์ (Population of Solutions) แล้วใช้หลักเกณฑ์ที่ต่างกัน

ในการค้นหาแบบเฟ้นสุ่ม ที่ทำการลอกเลียนแบบการพัฒนาตัวเองตามธรรมชาติ เช่น การที่มดพยายามหาเส้นทางที่สั้นที่สุดไปสู่แหล่งอาหาร หรือการที่นกพยายามหาจุดหมายปลายทางระหว่างการอพยพ ซึ่งขั้นตอนดังกล่าว จะถูกทำซ้ำไปซ้ำมาจนกระทั่งได้ตามเงื่อนไขของการหยุดการทำงาน Clune และคณะ [13] ได้กล่าวถึง ขั้นตอนวิธีด้านวิวัฒนาการว่าเป็นหนึ่งในเรื่องของ การคำนวณเชิงวิวัฒนาการ (Evolutionary Computation) ที่ใช้ฐานประชากรโดยทั่วไปของขั้นตอนวิธีแบบเมตาฮิวริสติกที่เหมาะสมที่สุด (Metaheuristic Optimization Algorithm) โดยขั้นตอนวิธีเชิงวิวัฒนาการนั้นใช้กระบวนการที่ได้รับแรงบันดาลใจมาจากการวิวัฒนาการทางชีววิทยาได้แก่การสืบพันธุ์ (Reproduction) การกลายพันธุ์ (Mutation) การแลกเปลี่ยนยีน (Recombination) และการคัดเลือก (Selection) โดยจะมีผลเฉลยที่สามารถเลือกได้ (Candidate Solution) แทนประชากรและฟังก์ชันคุณภาพ (Quality Function) ในการคัดเลือกประชากรที่เหมาะสมตามสภาพแวดล้อมที่กำหนดไว้ [14, 15] ขั้นตอนวิธีเชิงวิวัฒนาการนี้มักจะใช้ได้ดีสำหรับการหาผลเฉลยของปัญหาในทุกๆ ด้านเนื่องจากสามารถพัฒนาผลเฉลยที่มีไปยังผลเฉลยที่ต้องการได้อย่างรวดเร็วทำให้ประสบความสำเร็จในหลายๆ ด้านของปัญหา เช่น วิศวกรรม ศิลปกรรม ชีวภาพ เศรษฐศาสตร์ การตลาด พันธุศาสตร์ การค้นคว้าวิจัย การออกแบบหุ่นยนต์ วิทยาศาสตร์ ด้านสังคม ฟิสิกส์ รัฐศาสตร์ และเคมี เป็นต้น ซึ่งขั้นตอนวิธีด้านวิวัฒนาการ ได้แบ่งออกเป็น 2 ด้าน คือ วิวัฒนาการด้านพันธุกรรม เช่น ขั้นตอนวิธีเชิงพันธุกรรม (Genetic Algorithms: GAs) ขั้นตอนวิธีมีมติก (Memetic Algorithms: MAs) เป็นต้น และวิวัฒนาการด้านพฤติกรรม เช่น การหาค่าเหมาะสมด้วยวิธีอนุภาคเคลื่อนที่ (Particle Swarm Optimization: PSO) การหาค่าเหมาะสมที่สุดแบบอาณานิคมมด (Ant Colony Optimization: ACO) ขั้นตอนวิธีชัฟเฟิลฟรอกลิปปีง (Shuffle Frog Leaping Algorithms: SFL) เป็นต้น

2.2 ขั้นตอนวิธีด้านวิวัฒนาการแบบหลายวัตถุประสงค์

ขั้นตอนวิธีด้านวิวัฒนาการแบบหลายวัตถุประสงค์ได้นำมาประยุกต์ใช้ในปัญหาการหาค่าเหมาะสมที่สุดที่มีหลายวัตถุประสงค์ โดยขั้นตอนวิธีนี้ได้มีวัตถุประสงค์เพื่อค้นหาขอบเขตกลุ่มคำตอบที่ดีที่สุดที่มีลักษณะการกระจายแบบสม่ำเสมอ และใกล้เคียงกับขอบเขตกลุ่มคำตอบที่ดีที่สุด

ที่แท้จริง ซึ่งถือว่ามีความซับซ้อนมากกว่าปัญหาการหาค่าเหมาะสมที่สุดแบบวัตถุประสงค์เดียว งานวิจัยที่นำเสนอขั้นตอนวิธีด้านวิวัฒนาการที่ถูกพัฒนาขึ้นเพื่อแก้ปัญหาหลายวัตถุประสงค์ โดยในช่วงเริ่มแรกการแก้ปัญหาการหาค่าเหมาะสมที่สุดที่มีหลายวัตถุประสงค์เป็นที่สนใจจากนักวิจัยเพิ่มมากขึ้น ไม่ว่าจะเป็นการแนะนำและปรับปรุงขั้นตอนวิธีด้านวิวัฒนาการให้ดีขึ้น จนได้กลายเป็นเทคโนโลยีที่สำคัญของการแก้ปัญหาแบบหลายวัตถุประสงค์ ที่รู้จักกันทั่วไป เช่น Vector Evolution Genetic Algorithm (VEGA) นำเสนอโดย Schaffer [16] Multi-Objective Genetic Algorithm (MOGA) นำเสนอโดย Fonseca and Fleming [3] และ Non-dominated Sorting Genetic Algorithm (NSGA) นำเสนอโดย Srinivas และ Deb [4] หลังจากนั้นได้มีการนำเสนอ Niched Pareto Genetic Algorithm (NPGA) โดย Horn และ Nafpliotis [17] และ Horn และคณะ [18] Pareto Achieved Evolution Strategy (PAES) นำเสนอโดย Knowles and Corne [19] Strength Pareto Evolutionary Algorithm (SPEA) นำเสนอโดย Zitzler and Thiele [20] และ Multi-Objective Particle Swarm Optimizer (PSO) นำเสนอโดย Reyes-Sierra และ Coello [21] เป็นต้น

การแก้ปัญหาการหาค่าเหมาะสมที่สุดที่มีหลายวัตถุประสงค์ เป็นการค้นหาเซตคำตอบภายในพื้นที่ของคำตอบที่เป็นไปได้ เพื่อต้องการหาค่าที่ต่ำที่สุด หรือค่าสูงสุดของฟังก์ชันวัตถุประสงค์ในแต่ละฟังก์ชันพร้อมๆ กัน โดยผลลัพธ์ที่ได้จากการแก้ปัญหา เรียกว่า เซตกลุ่มคำตอบที่ดีที่สุดปัญหาแบบหลายวัตถุประสงค์ เป็นปัญหาการออกแบบที่มีหลายวัตถุประสงค์ ประกอบด้วย m วัตถุประสงค์ และตัวแปรตัดสินใจ [2] เขียนอยู่ในรูปทั่วไปดังสมการที่ (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

เมื่อ $i = 1, 2, \dots, m$

ดังนั้นรูปแบบปัญหาการหาค่าเหมาะสมที่สุดที่มีหลายวัตถุประสงค์ จะเป็นการค้นหาเวกเตอร์คำตอบ x ภายใต้ m ข้อจำกัดแบบสมการดังสมการที่ (2) หรือภายใต้ n ข้อจำกัดแบบสมการดังสมการที่ (3) ซึ่งจะเป็นการกำหนดขอบเขตพื้นที่คำตอบที่เป็นไปได้

$$g_i(x) \leq 0, i = 1, 2, \dots, m \quad (2)$$

$$h_i(x) = 0, i = 1, 2, \dots, n \quad (3)$$

ปกติโดยทั่วไปรูปแบบปัญหาการหาค่าเหมาะสมที่สุดที่มีหลายวัตถุประสงค์มี 3 รูปแบบ ที่เป็นไปได้ ดังนี้ 1) ทุกฟังก์ชันวัตถุประสงค์ต้องการหาค่าตอบที่น้อยที่สุด 2) ทุกฟังก์ชันวัตถุประสงค์ต้องการหาค่าตอบที่มากที่สุด 3) แบบขัดแย้งกัน คือบางฟังก์ชันวัตถุประสงค์ต้องการหาค่าตอบที่น้อยที่สุด บางฟังก์ชันวัตถุประสงค์ต้องการหาค่าตอบที่มากที่สุด

ในการหาค่าเหมาะสมที่สุดที่มีหลายวัตถุประสงค์สามารถแปลงฟังก์ชันวัตถุประสงค์ให้มีรูปแบบเป็นการหาค่าตอบที่น้อยที่สุดหรือมากที่สุดได้ ตัวอย่างเช่น การแปลงทุกฟังก์ชันวัตถุประสงค์ที่มีการหาค่าตอบที่น้อยที่สุดให้เป็นรูปแบบการหาค่าคำตอบที่มากที่สุด เป็นต้น

สำหรับตัวอย่างคำตอบจากปัญหาการหาค่าฟังก์ชันวัตถุประสงค์ในที่นี้เป็นการหาค่าตอบที่น้อยที่สุด จากสมการที่ (4) ซึ่งการค้นหาคำตอบจะถูกกำหนดจากเวกเตอร์ของตัวแปรตัดสินใจ ภายใต้ข้อจำกัดที่จะเป็นสิ่งที่ใช้ในการกำหนดขอบเขตที่น้อยที่สุด

$$\text{Minimize} : \{f_1(x), f_2(x), \dots, f_m(x)\} \quad (4)$$

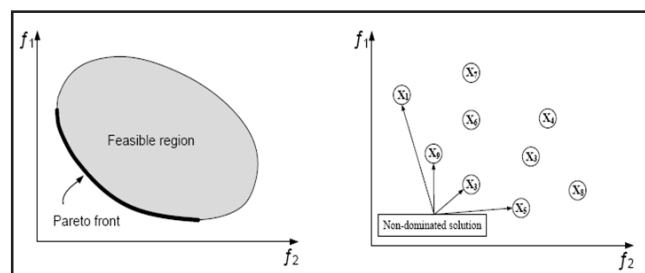
ข้อจำกัด $g(x) \leq 0$

โดยที่ x คือ เวกเตอร์ของตัวแปรตัดสินใจ

$f_i(x)$ คือ ฟังก์ชันวัตถุประสงค์ที่ i

$g_i(x)$ คือ เวกเตอร์ข้อจำกัดที่ i

ซึ่งคำตอบในพื้นที่คำตอบที่เป็นไปได้ เป็นคำตอบที่ไม่มีคำตอบใดดีกว่า หรือไม่มีคำตอบใดที่สามารถครอบงำชุดคำตอบนี้ได้ จะเรียกว่าคำตอบนี้เป็น กลุ่มคำตอบที่เหมาะสมที่สุด (Pareto Optimal) และเรียกสมาชิกคำตอบทุกคำตอบที่อยู่ในกลุ่มคำตอบที่เหมาะสมที่สุดว่า เซตกลุ่มคำตอบที่เหมาะสมที่สุด (Pareto Optimal Set) หรือเซตคำตอบที่ไม่ถูกรอบงำจากทุกคำตอบ (Non-dominated Set) ดังแสดงในภาพที่ 1



ภาพที่ 1 พาเรโตฟรอนของฟังก์ชันวัตถุประสงค์ f_1 และ f_2 [22]

Fonseca และ Fleming [23] ได้ให้คำนิยามของการครอบงำแบบพาเรโต (Pareto Dominance) และคำตอบที่เหมาะสมที่สุดแบบพาเรโต (Pareto Optimality) ไว้ดังนี้

นิยาม 1 : การครอบงำแบบพาเรโต

ให้เวกเตอร์วัตถุประสงค์ $u = (u_1, \dots, u_m)$ ซึ่งครอบงำ $v = (v_1, \dots, v_m)$ ก็ต่อเมื่อ u น้อยกว่า v บางส่วนดังสมการที่ (5)

$$\forall_i \in \{1, \dots, m\}, u_i \leq v_i \wedge \exists_i \in \{1, \dots, m\} : u_i < v_i \quad (5)$$

นิยามเกี่ยวกับการครอบงำดูจากค่าของจุดประสงค์เวกเตอร์ u ครอบงำเวกเตอร์ v ก็ต่อเมื่อในแต่ละจุดประสงค์ค่าของ u ดีกว่าหรือเท่ากับค่าของ v และ อย่างน้อยหนึ่งกรณีที่ค่าของ u มีค่าดีกว่า v

นิยาม 2 : คำตอบที่เหมาะสมที่สุดแบบพาเรโต

โซลูชัน $x_u \in [a, b]$ เป็นคำตอบที่เหมาะสมที่สุดแบบพาเรโต ถ้าไม่สามารถหาคำตอบ $x_v \in [a, b]$ ซึ่ง $v = f(x_v)$ ครอบงำ $u = f(x_u)$

ในด้านการวัดประสิทธิภาพของเทคนิคต่าง ๆ ที่นำใช้ในการแก้ปัญหาการหาค่าเหมาะสมที่สุดที่มีหลายวัตถุประสงค์ จะทำโดยการวัดสมรรถนะของกลุ่มคำตอบที่หาได้ ตัวอย่างการวัดสมรรถนะที่นิยมใช้ เช่น 1) การวัดสมรรถนะของคำตอบด้านการเข้าสู่กลุ่มคำตอบที่แท้จริง (Convergence Measurement) เป็นตัวชี้วัดการเข้าสู่ของกลุ่มคำตอบโดยเปรียบเทียบระยะทางจากสมาชิกของกลุ่มคำตอบทุกคำตอบที่หาได้ (Obtained Solution Set) กับกลุ่มคำตอบที่แท้จริง (True Pareto Set) และ 2) การวัดสมรรถนะของคำตอบด้านการกระจายของกลุ่มคำตอบที่หาได้ (Spread Measurement) เป็นตัวชี้วัดที่บอกการกระจายของกลุ่มคำตอบ สามารถดูรายละเอียดตัววัดสมรรถนะได้จากเอกสารหมายเลข [12]

3. ทฤษฎีความอลวน (Chaos Theory)

ทฤษฎีความอลวน [24] เป็นทฤษฎีที่อธิบายถึงลักษณะพฤติกรรมของระบบพลวัต (คือระบบที่มีการเปลี่ยนแปลง เช่น เปลี่ยนแปลงตามเวลาที่เปลี่ยนไป) โดยลักษณะการเปลี่ยนแปลงของระบบที่เรียกว่าอลวนนี้ จะมีลักษณะที่ปั่นป่วนจนดูคล้ายว่า การเปลี่ยนแปลงนั้นเป็นแบบสุ่มหรือไร้ระเบียบ (Random/Stochastic) แต่จริงๆ แล้ว ระบบอลวนนี้เป็นระบบแบบไม่สุ่ม หรือระบบที่สามารถกำหนดหรือระบุความเป็นระเบียบได้ (Deterministic) และสามารถแทนในรูป

สมการทางคณิตศาสตร์ได้เช่นกัน

3.1 ลักษณะของความอลวน

1) มีคุณสมบัติแบบไม่เป็นเชิงเส้น คือ ผลลัพธ์ทั้งหมดของระบบไม่เท่ากับผลรวมของผลลัพธ์ที่เกิดจากส่วนย่อย ๆ ของระบบรวมกัน

2) ไม่ใช่เกิดแบบสุ่ม หรือเรียกได้ว่าในระบบอลวน เหตุการณ์ทั้งหลายมักเกิดขึ้นภายใต้กฎเกณฑ์ที่แน่นอนตายตัว เพื่อป้องกันความสับสนระหว่าง “ความอลวน” และ “การสุ่ม” จึงเรียก Chaos ว่า “Deterministic Chaos”

3) ไวต่อสภาวะเริ่มต้น นั่นคือถ้าระบบใดๆ ที่เริ่มต้นจากสภาวะที่แตกต่างกันเพียงเล็กน้อยเมื่อระบบได้มีการเปลี่ยนแปลงไปสักระยะหนึ่งสภาวะของระบบทั้งสองนั้นจะแตกต่างกันอย่างเห็นได้ชัด

4) ไม่สามารถทำนายล่วงหน้าในระยะยาวได้ เพราะไม่สามารถรู้ได้ว่าจะมีเหตุปัจจัยใดที่กระทบ ส่งผลให้เกิดการเปลี่ยนแปลง

ทฤษฎีความอลวนมีประโยชน์ [5] คือ ใช้ในการวิเคราะห์ระบบและทำนายอนาคตและใช้ในการควบคุม-สร้างความเสถียรให้กับระบบสามารถใช้หลักการของทฤษฎีความอลวนช่วยให้การหาค่าที่เหมาะสมที่สุด ของฟังก์ชันหนึ่งได้ค่าดีที่สุดที่แท้จริงได้ง่ายขึ้น เพราะความอลวนสามารถช่วยให้หลีกเลี่ยงการได้ค่าดีที่สุดเฉพาะบริเวณได้

3.2 ลอจิสติกแมป (Logistic Map)

ลอจิสติกแมปเป็นระบบอลวนที่นิยมใช้ ชนิดทางสองแพร่ง (Bifurcation) [25] จะมีลักษณะเหมือนตัววาย (Y) เกิดขึ้นเมื่อความไร้ระเบียบได้พัฒนาตัวเองออกไปจากจุดสมดุลจนสุดขอบเขต ทำให้ระบบมีความยุ่งเหยิงซับซ้อนที่สุด จุดที่ไกลจากสมดุลที่สุดนี้ ตัวควบคุมหรือตัวดึงดูดจะเปลี่ยนสภาพโดยสิ้นเชิงไปสู่ตัวควบคุมตัวใหม่ที่จะทำหน้าที่กำหนดและควบคุมระบบที่ปรากฏขึ้นใหม่หลังทางแพร่งนี้ ทั้งที่ไม่เหมือนเดิมและซับซ้อนยิ่งกว่าเดิม แต่จะคงสาระและคุณสมบัติเดิมเป็นเนื้อในอยู่ในระบบใหม่ นอกจากนั้นยังมีตัวชี้วัดที่แสดงความเป็นระบบอลวนที่นิยมใช้กันโดยทั่วไป อีกคือ แฟรคทัล (Fractal) และเลขชี้กำลังของลอเรนซ์ (Lyapunov exponent) ที่รู้จักกันในนามตัวดึงดูดของลอเรนซ์ (Lorenz Attractor) สามารถดูรายละเอียดตัวชี้วัดความเป็นระบบอลวนได้จากเอกสารหมายเลข [25]

ลอจิสติกแมป เป็นตัวอย่างของระบบพลวัตไม่เป็นเชิงเส้นอย่างง่ายที่สามารถแสดงพฤติกรรมความอลวนได้เป็นที่รู้จักจากผลงานตีพิมพ์ของนักชีววิทยา โรเบิร์ตเมย์ (Robert May) [26] แรกเริ่มถูกสร้างขึ้นโดย ปีแอร์ ฟรองซัวเวอฮัลสต์ (Pierre François Verhulst) เพื่อเป็นแบบจำลองการกระจายปริมาณประชากรมนุษย์ ต่อมาถูกนำไปใช้สำหรับการเพิ่มปริมาณประชากรของสปีชีส์อื่นๆ ภายใต้สภาวะแวดล้อมจำกัด เช่น อาหาร โรค และอื่นๆ ซึ่งแบบจำลองจะมีพฤติกรรมจากผลของ

- การสืบพันธุ์ คือ จำนวนประชากรจะเพิ่มขึ้นด้วยอัตราที่แปรผันตามจำนวนประชากรในขณะนั้น

- การขาดอาหารคือ จำนวนประชากรจะลดลงด้วยอัตราที่แปรผันตามจำนวนประชากรที่สภาพแวดล้อมนั้นสามารถรองรับได้ในทางทฤษฎี ลบออกด้วยค่าจำนวนประชากรในขณะนั้น ซึ่งสามารถเขียนในรูปสมการทางคณิตศาสตร์ดังสมการที่ (6)

$$z^{k+1} = \mu z^k (1 - z^k) \quad k = 0, 1, \dots \quad (6)$$

โดยที่ $z^k \in [0, 1]$ คือตัวแปรความอลวน (Chaos Variable) ในการทำซ้ำจำนวน k ครั้ง (ใช้หมายถึงปริมาณประชากรที่ปี k และ z^0 หมายถึงปริมาณประชากรเริ่มต้น (ที่ปี 0))

เมื่อ μ คือพารามิเตอร์ที่สามารถควบคุมความอลวนของระบบ โดยให้

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

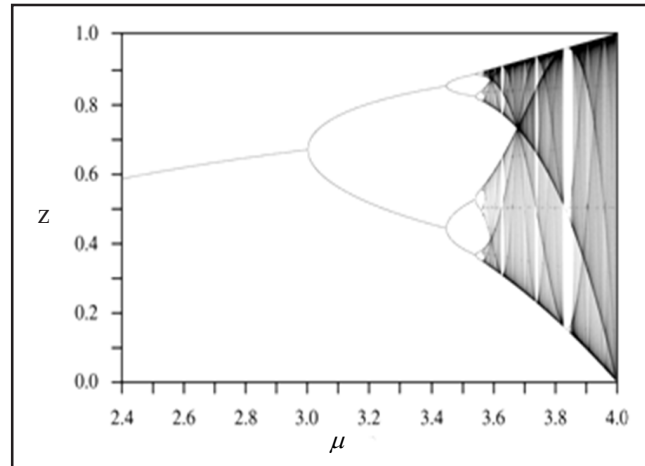
(μ เป็นจำนวนบวก หมายถึง ค่าผลรวมของอัตราการสืบพันธุ์ และการขาดอาหาร)

ถ้ากำหนดให้ $z^0 = 0.6$ และ μ คือ 0 – 4 ในขณะที่ $\mu < 3.0$ ฟังก์ชันไม่เปลี่ยนแปลงและในขณะที่ $\mu > 3.0$ จะเกิดเป็นทางแยกขึ้น ซึ่งเมื่อ μ เพิ่มขึ้นจะเกิดทางแยกปรากฏขึ้นใน 2^n รอบ แต่เมื่อ $\mu = 3.7$ ฟังก์ชันเริ่มเข้าสู่ความอลวนดังภาพที่ 2

เมื่อพิจารณาจากภาพที่ 2 เมื่อ $\mu = 3.7$ และ z^0 คือ 0.3 และ 0.301 จากการแยกกันเราสามารถหาค่า z^k ที่แตกต่างกันหลังจาก k ครั้ง โดยมีลักษณะที่สำคัญ คือ ความไวต่อสภาวะเริ่มต้น กล่าวคือ ความแตกต่างเพียงเล็กน้อยในสถานะเบื้องต้น จะนำมาซึ่งความแตกต่างที่มากมายของผลลัพธ์จะเห็นได้จากค่า z^0

ในการเริ่มต้นหากมีการเปลี่ยนแปลงเพียงเล็กน้อย

ฟังก์ชันจะนำเสนอปรากฏการณ์ความอลวนที่แตกต่างกันอย่างเห็นได้ชัด



ภาพที่ 2 ความไวต่อสภาวะเริ่มต้นของความอลวน

3.3งานวิจัยที่นำทฤษฎีความอลวนมาเพิ่มประสิทธิภาพให้กับขั้นตอนวิธีด้านวิวัฒนาการ

จากตัวอย่างงานวิจัยที่ได้ศึกษากรณีนำทฤษฎีความอลวนมาเพิ่มประสิทธิภาพในการทำงาน เช่น Zelinka I. [27] ได้นำมาทดลองกับ 3 ขั้นตอนวิธีคือ differential evolution, self-organizing migrating algorithm และ GA กับปัญหา class of CML (coupled map lattices) ได้ทดลองซ้ำจำนวน 50 ครั้ง ในทุกการทดลอง ผลลัพธ์แสดงให้เห็นว่า chaos control สามารถแก้ปัญหา class ของปัญหานี้ได้และทั้งสามขั้นตอนวิธีก็ให้ผลลัพธ์ที่ดี Zhenyu G. และคณะ [28] ได้นำเสนอ DE (Differential Evolution) แบบ Self-adaptive Chaos Differential Evolution (SACDE) ด้วย Chaos mutation ซึ่งใช้ในการเพิ่มช่วงของการ mutation เพื่อหลีกเลี่ยงการได้ค่าที่ดีที่สุดเฉพาะบริเวณ ซึ่งผลลัพธ์ที่ได้คือ สามารถเพิ่มความเร็วและความแม่นยำของ DE และเวลาที่ลดลงก็แสดงให้เห็นว่า Chaos mutation เพิ่มประสิทธิภาพให้กับ DE ส่วน Stefania และ Member [29] ได้นำเอาความอลวนมาพัฒนาประสิทธิภาพการทำงานของขั้นตอนวิธีเชิงพันธุกรรมที่ใช้ร่วมกับโครงข่ายประสาทเทียมแบบพืชีสำหรับปัญหาการวินิจฉัยโรค Ying และคณะ [30] ได้ใช้การค้นหาแบบอลวนซึ่งเป็นการค้นหาแบบเฉพาะที่มีความสามารถในการค้นพบที่ดีพัฒนาประสิทธิภาพของโครงสร้างของการหาค่าเหมาะสมด้วยวิธีอนุภาคเคลื่อนที่ Leifu Gao และ Xuwang Liu [31] ได้ใช้ความอลวนร่วมกับ PSO โดยใช้ Chaos

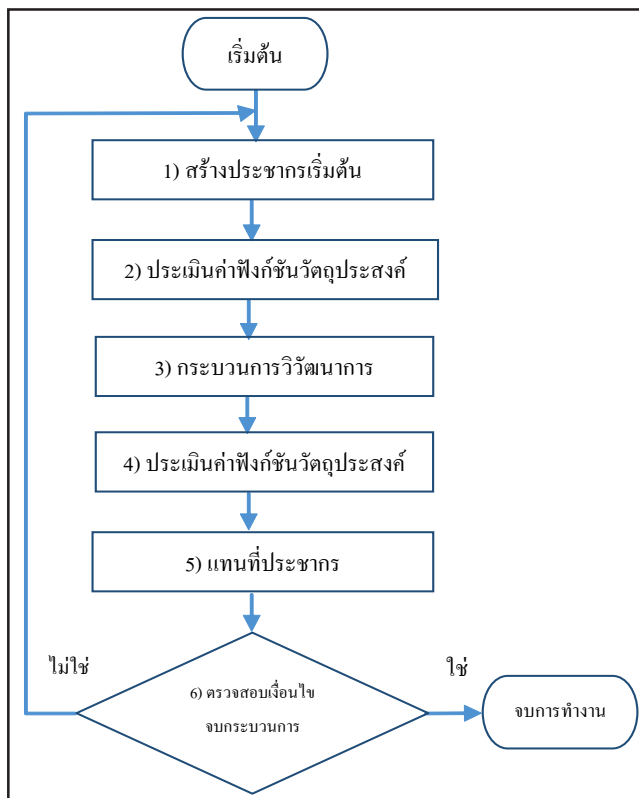
Optimization Algorithms (COAs) ซึ่งก็คือ การค้นหาแบบอลวน (Chaos Search) มาช่วยในการหลีกเลี่ยงการได้ค่าที่ดีที่สุดเฉพาะบริเวณ เป็นต้น ซึ่งผลการวิจัยที่ปรากฏแสดงให้เห็นว่าทฤษฎีความอลวนนั้นสามารถนำมาเพิ่มประสิทธิภาพให้ขั้นตอนวิธีต่างๆ ได้เป็นอย่างดีไม่ว่าในด้านการหลีกเลี่ยงการได้ค่าที่ดีที่สุดเฉพาะบริเวณ ได้การเพิ่มความเร็วและความแม่นยำให้กับขั้นตอนวิธีที่นำมาใช้

4. ตัวอย่างการประยุกต์ใช้ทฤษฎีความอลวนในขั้นตอนวิธีด้านวิวัฒนาการแบบหลายวัตถุประสงค์

บทความนี้ได้นำเสนอขั้นตอนวิธีด้านวิวัฒนาการแบบหลายวัตถุประสงค์โดยยกตัวอย่างจากขั้นตอนวิธีด้านวิวัฒนาการด้านพันธุกรรมและใช้ลอจิสติกแมปเป็นตัวแทนของความอลวนโดยมีรายละเอียด ดังนี้

4.1 โครงสร้างพื้นฐานวิวัฒนาการด้านพันธุกรรม

หลักการเบื้องต้นของขั้นตอนวิธีด้านวิวัฒนาการด้านพันธุกรรมคืออนุญาตให้คำตอบที่ได้จากการพัฒนาคำตอบเบื้องต้นมีการวิวัฒนาการตนเองสู่คำตอบในแต่ละรุ่นด้วยกระบวนการกลายพันธุ์และแลกเปลี่ยนพันธุกรรม (Mutation and Crossover Process) ซึ่งมีวิธีการหลักๆ แสดงดังภาพที่ 3



ภาพที่ 3 โครงสร้างพื้นฐานของขั้นตอนวิธีด้านวิวัฒนาการด้านพันธุกรรม

1) สร้างประชากรเริ่มต้น เรียกว่า เวกเตอร์เริ่มต้นหรือโครโมโซมเริ่มต้น

2) ประเมินค่าฟิตเนสหรือค่าฟังก์ชันวัตถุประสงค์ของแต่ละเวกเตอร์หรือโครโมโซมในประชากร

3) ดำเนินการวิวัฒนาการด้วยกระบวนการต่อไปนี้ ตามจำนวนรอบที่ตั้งไว้

-การคัดเลือก (Selection) เป็นการคัดเลือกโครโมโซมหรือเวกเตอร์ต้นแบบโดยดูจากค่าความแข็งแรง (Fitness Value)

-สร้างประชากรรุ่นใหม่ด้วยกระบวนการแลกเปลี่ยนพันธุกรรม (Crossover) การกลายพันธุ์ (Mutation) หรือการปรับเปลี่ยนค่าองค์ประกอบของโครโมโซมหรือค่าในพิกัดในเวกเตอร์

4) ประเมินค่าฟิตเนสหรือค่าฟังก์ชันวัตถุประสงค์

5) ทำการแทนที่โครโมโซมหรือเวกเตอร์เก่าด้วยเวกเตอร์ใหม่ที่มีค่าฟิตเนสหรือฟังก์ชันวัตถุประสงค์ที่ดีกว่า

6) ทำการตรวจสอบการสิ้นสุดการค้นหาผลลัพธ์

4.2 ตัวอย่างเทคนิคความอลวนที่ได้นำมาใช้ในขั้นตอนวิธีด้านวิวัฒนาการหลายวัตถุประสงค์ด้านพันธุกรรม

จากที่ผู้เขียนบทความได้ทำการวิจัย [7-10] โดยนำเทคนิคความอลวนมาใช้พัฒนาขั้นตอนวิธีชื่อว่า Multi-Objective Chaos Memetic Algorithm (MOCMA) นำมาทดสอบประสิทธิภาพกับชุดปัญหา DTLZ [32] แล้วเปรียบเทียบกับประสิทธิภาพกับขั้นตอนวิธี Non-dominated Sorting Genetic Algorithm II (NSGAI) [33] ผลการวิจัยพบว่าในการวัดสมรรถนะของคำตอบ ด้านการลู่เข้าสู่คำตอบที่แท้จริงจะมีอัตราการลู่เข้าสู่คำตอบที่แท้จริงดีกว่าและด้านการวัดสมรรถนะของคำตอบด้านการกระจายของกลุ่มคำตอบมีการกระจายที่สม่ำเสมอมากกว่าขั้นตอนวิธี NSGAI ซึ่งตัวอย่างเทคนิคความอลวนที่นำมาใช้ เช่น

1) การค้นหาแบบอลวน (Chaos Search) [8], [10] เป็นการค้นหาแบบเฉพาะที่ (Local Search) มักจะถูกใช้หลังจากกระบวนการทางพันธุกรรม ด้วยหวังที่จะเพิ่มประสิทธิภาพของวิธีทางพันธุกรรม โดยนำการค้นหาแบบอลวนมาใช้ภายหลังขั้นตอนการสร้างประชากรและหลังขั้นตอนการกลายพันธุ์

กระบวนการของการค้นหาแบบอลวน อธิบายได้ดังนี้
ขั้นตอนที่ 1: กำหนดเวกเตอร์ตัดสินใจ (Decision Vectors)

$$x_{ij}^* = (x_{i1}^*, x_{i2}^*, \dots, x_{in}^*)$$



$i = 1, 2, \dots, NN$ คือ จำนวนของโครโมโซม

$j = 1, 2, \dots, nn$ คือ จำนวนของยีน

ทำการกำหนดเวกเตอร์ตัดสินใจ ตัวอย่างในภาพที่ 4

0.24	0.18	0.95	0.56	0.73	0.50
X_{11}	X_{12}	X_{13}	X_{14}	X_{15}	X_{16}

ภาพที่ 4 ตัวอย่างการกำหนดเวกเตอร์ตัดสินใจ (โครโมโซม 1)

ขั้นตอนที่ 2: กำหนดโดเมนของแต่ละโครโมโซม

$$\{(x_{ij}^* - \varepsilon, x_{ij}^* + \varepsilon), \dots, (x_{Nn}^* - \varepsilon, x_{Nn}^* + \varepsilon)\}$$

โดยที่ ε คือ ระบุไว้สำหรับกำหนดรัศมีของพื้นที่ใกล้เคียง (เช่น 0.01, 0.05) ตัวอย่างกำหนด $\varepsilon = 0.05$ เพื่อเข้าสู่กระบวนการค้นหาแบบอลวนจะได้ดังภาพที่ 5

0.24-0.05 , 0.24+0.05	0.18-0.05 , 0.18+0.05	0.95-0.05 , 0.95+0.05
X_{11}	X_{12}	X_{13}
0.56-0.05 , 0.56+0.05	0.73-0.05 , 0.73+0.05	0.50-0.05 , 0.50+0.05
X_{14}	X_{15}	X_{16}

ภาพที่ 5 กำหนดโดเมนของแต่ละโครโมโซมในการค้นหาแบบอลวน

ขั้นตอนที่ 3: จากนิยามของคำตอบที่เหมาะสมที่สุดแบบพาเรโต โดยที่ $[a, b]$ เป็นคำตอบที่เหมาะสมที่สุดแบบพาเรโต

$$\text{เมื่อ } x_{ij}^* - \varepsilon < a_j, x_{ij}^* + \varepsilon > b_j$$

$$\text{ดังนั้นให้ } x_{ij}^* - \varepsilon = a_j, x_{ij}^* + \varepsilon = b_j$$

ดำเนินการแมปสมการความอลวนโดยใช้สมการที่ (6) ได้ดังสมการที่ (7)

$$cx_{ij}^{k+1} = 4cx_{ij}^k(1 - cx_{ij}^k) \quad (7)$$

$$i = 1, \dots, N, \quad j = 1, \dots, n$$

จากตัวอย่างมีจำนวน 6 ตัว กำหนด $k = 6$ สุ่มค่าเริ่มต้น $cx_{ij}^{k+1} = 0.321$ จะได้ดังภาพที่ 6

0.87	0.45	0.99	0.05	0.17	0.57
cx_{11}	cx_{12}	cx_{13}	cx_{14}	cx_{15}	cx_{16}

ภาพที่ 6 การแมปสมการความอลวนเพื่อหาค่า Chaos Variable

ขั้นตอนที่ 4: นำค่าของ Chaos Variable cx_{ij}^{k+1} ที่ได้แมปลงในช่วงของ $[a, b]$ ในสมการที่ (8) เมื่อแทนค่าจะได้ดังสมการที่ (9)

$$x_{ij}^{k+1} = a_j + (b_j - a_j)cx_{ij}^{k+1} \quad (8)$$

$$x_{ij}^{k+1} = x_{ij}^* - \varepsilon + 2cx_{ij}^{k+1} \quad (9)$$

ได้ดังภาพที่ 7 และได้ผลดังภาพที่ 8

0.24-0.05 + (2 × 0.05 × 0.87)	0.18-0.05 + (2 × 0.05 × 0.45)	0.95-0.05 + (2 × 0.05 × 0.99)
X_{11}	X_{12}	X_{13}
0.56-0.05 + (2 × 0.05 × 0.05)	0.73-0.05 + (2 × 0.05 × 0.17)	0.50-0.05 + (2 × 0.05 × 0.57)
X_{14}	X_{15}	X_{16}

ภาพที่ 7 การแมป Chaos Variable ลงในช่วง Optimization Valuable

0.28	0.18	1	0.52	0.7	0.51
X_{11}	X_{12}	X_{13}	X_{14}	X_{15}	X_{16}

ภาพที่ 8 ผลการแมป Chaos Variable ลงในช่วง Optimization Valuable

ขั้นตอนที่ 5: คำนวณเวกเตอร์ฟังก์ชันวัตถุประสงค์ของแต่ละประชากร

ขั้นตอนที่ 6: ทำการเปรียบเทียบค่าฟังก์ชันวัตถุประสงค์ว่าตรงตามกฎการยอมรับหรือไม่ กฎการยอมรับ คือคำตอบที่ได้หลังจากค้นหาเฉพาะที่จะยอมรับก็ต่อเมื่อคำตอบที่ได้หลังจากการค้นหาฟังก์ชันวัตถุประสงค์น้อยกว่าคำตอบดั้งเดิม

2) การสร้างประชากรเบื้องต้นด้วยความอลวน (Chaos Initialization)

ประชากรเริ่มต้นถูกสร้างขึ้นโดยลำดับความอลวนที่มีลักษณะของการแพร่กระจายที่มากกว่าการสุ่ม ความไวต่อสภาวะเริ่มต้น และข้อมูลซึ่งกระจายอยู่ในพื้นที่ที่แก้ปัญหาอย่างสม่ำเสมอ ดังนั้นขั้นตอนวิธีนี้สามารถเอาชนะข้อมูลของลำดับการสุ่มที่ซ้ำซ้อนได้ ซึ่งประชากรจะถูกสร้างขึ้นตามสมการที่ (10)

$$CsC = \{X_{ij}(C_j)\} = \begin{bmatrix} x_{11}(c_1) & x_{12}(c_1) & \dots & x_{1v}(c_1) \\ x_{21}(c_2) & x_{22}(c_2) & \dots & x_{2l}(c_2) \\ \vdots & \vdots & \ddots & \vdots \\ x_{m1}(c_1) & x_{m2}(c_1) & \dots & x_{mn}(c_n) \end{bmatrix} \quad (10)$$

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

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

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

c_j คือ ค่าที่คำนวณได้จากสมการที่ (6)

จากข้างต้น โครโมโซม CsC ถูกสร้างขึ้นโดย c ซึ่งเป็นค่าเริ่มต้นสำหรับการจับคู่ของสมการ โครโมโซมเริ่มต้นที่ได้กำหนดในทุกุ่นจะแตกต่างกัน สามารถลดความซ้ำซ้อนของโครโมโซม หลังจากที่มีการสร้างโครโมโซมจะได้ว่าโครโมโซมของทุกุ่นมีความซ้ำซ้อนของข้อมูลที่ลดลงซึ่งส่งผลสมรรถนะของคำตอบด้านการเข้าสู่คำตอบที่แท้จริงและด้านการกระจายของกลุ่มคำตอบมีการกระจายที่สม่ำเสมอ ผลการทดลองสามารถดูได้จากงานวิจัย [7-9]

5. สรุป

เนื้อหาที่กล่าวไว้ในบทความนี้เป็น การเพิ่มประสิทธิภาพการทำงานให้กับขั้นตอนวิธีด้านวิวัฒนาการแบบหลายวัตถุประสงค์ด้วยทฤษฎีความอลวนและหาประสิทธิภาพโดยพิจารณาในประเด็น 2 ประเด็น คือ การเข้าสู่กลุ่มคำตอบที่แท้จริงหรือไม่ และการกระจายของกลุ่มคำตอบที่หาได้เหมาะสมหรือไม่ จากตัวอย่างเทคนิคความอลวนที่ยกมา ได้แก่ การค้นหาแบบอลวนและการสร้างประชากรเบื้องต้นด้วยความอลวนนั้น ได้ส่งผลให้สมรรถนะของคำตอบด้านการเข้าสู่คำตอบที่แท้จริงและด้านการกระจายของกลุ่มคำตอบมีการกระจายที่สม่ำเสมอเพิ่มขึ้น และนอกเหนือจากที่นำเสนอในบทความ ยังมีเทคนิคที่นำความอลวนมาใช้อีก เช่น การคัดเลือกแบบอลวน (Chaos Selection) การกลายพันธุ์แบบอลวน (Chaos Mutation) เป็นต้น ซึ่งผู้วิจัยที่สนใจสามารถนำไปประยุกต์ใช้กับขั้นตอนวิธีต่างๆ ที่ใช้แก้ปัญหาการหาค่าเหมาะสมที่สุดได้

สุดท้ายการตัดสินใจว่าจะคัดเลือกขั้นตอนวิธีวิวัฒนาการแบบใดมาประยุกต์ใช้กับปัญหานั้น ไม่ได้มีคำตอบที่ตายตัว แต่หากผู้วิจัยเข้าใจการทำงานของขั้นตอนวิธีที่เลือกใช้ได้ อย่างถ่องแท้ ทำการปรับค่าพารามิเตอร์ต่างๆ ที่เกี่ยวข้องได้อย่างเหมาะสม หรือหาเทคนิคใหม่ๆ มาประยุกต์เพื่อช่วย

เพิ่มประสิทธิภาพการทำงานให้กับขั้นตอนวิธีนั้นๆ นอกจากจะสามารถแก้ปัญหาที่มีความซับซ้อนได้แล้ว ยังสามารถสร้างความรู้ใหม่ให้กับงานวิจัยในด้านนี้อีกด้วย

6. เอกสารอ้างอิง

- [1] K. Deb. *Multi-Objective Optimization using Evolutionary Algorithms*, Wiley 2001.
- [2] 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.
- [3] C. M. Fonseca and P. J. Fleming. "Genetic Algorithms for Multiobjective Optimization: Formulation Discussion and Generalization." *In Proceedings of the 5th International Conference on Genetic Algorithms*, pp. 416-423, 1993.
- [4] N. Srinivasand and K. Deb. "Multiobjective Optimization Using Non-Dominated Genetic Algorithms." *Evolutionary Computation*, Vol. 2, No. 3, pp. 221-248, 1994.
- [5] สมเกียรติ ตั้งกิจวานิชย์. *ทฤษฎีความอลวน*. Available online at www.tdri.or.th/reports/unpublished/chaos-theory.pdf. (วันที่ค้นคว้าข้อมูล 10 ธันวาคม 2555).
- [6] B. Li and W. Jiang. "Optimization of complex functions by chaos search." *International Journal of Cybernetics and Systems*, pp. 409-419, 1998.
- [7] ภรณ์ยา อามฤรัตน์ และ พยุง มีสัจ. "การหาค่าที่เหมาะสมที่สุดด้วยขั้นตอนวิธีเคออสมีมีติกแบบหลายวัตถุประสงค์." *วารสารเทคโนโลยีสารสนเทศ*, ฉบับที่ 15 หน้า 54-61, มกราคม - มิถุนายน 2555.
- [8] P. Ammaruekarat and P. Meesad. "Multi-Objective Chaos Memetic Algorithm for DTLZ Problems." *Advanced Materials Research*, Vol. 403-408, pp. 3676-3681, 2012.
- [9] P. Ammaruekarat and P. Meesad. "A Multi-Objective Memetic Algorithm Based on Chaos Optimization." *Applied Mechanics and Materials*, Vol. 130-134, pp. 725-729, 2012.

- [10] P. Ammaruekarat and P. Meesad. "A Chaos Search for Multi-Objective Memetic Algorithm." *International Conference on Information and Electronics Engineering (ICIEE)*, Bangkok, Thailand, pp. 140-144, 2011.
- [11] ภรณ์ยา อามฤครัตน์ และ พยุง มีสัจ. "เคออสมีมิติ อัลกอริทึมแบบหลายวัตถุประสงค์สำหรับปัญหาการจัดตารางการทำงานพยาบาลประจำการ." *การประชุมทางวิชาการ NCIT2008*, ครั้งที่ 4, เพชรบุรี, หน้า 147-152, 2555.
- [12] ภรณ์ยา อามฤครัตน์ และ พยุง มีสัจ. "การหาค่าที่เหมาะสมที่สุดที่มีหลายวัตถุประสงค์ด้วยขั้นตอนวิธีด้านวิวัฒนาการ." *วารสารเทคโนโลยีสารสนเทศ*, ฉบับที่ 2, หน้า 73-80, กรกฎาคม-ธันวาคม 2555.
- [13] J. Clune, C. Ofria and R. T. Pennock. "How a generative encoding fares as problem-regularity decreases." in PPSN (G. Rudolph, T. Jansen, S. M. Lucas, C. Poloni, and N. Beume, eds.), Vol. 5199, *Lecture Notes in Computer Science*, Springer, pp. 358-367, 2008.
- [14] G.S. Hornby and J.B. Pollack. "Creating high-level components with a generative representation for body-brain evolution." *Artificial Life*, pp. 223-246, 2002.
- [15] J. Clune, C. Ofria and R. T. Pennock. "How a generative encoding fares as problem-regularity decreases." in PPSN (G. Rudolph, T. Jansen, S. M. Lucas, C. Poloni, and N. Beume, eds.), Vol. 5199, *Lecture Notes in Computer Science*, Springer, pp. 358-367, 2008.
- [16] J. D. Schaffer. "Multiobjective Optimization with Vector Evaluated Genetic Algorithms." In *Proceedings of 1st International Conference on Genetic Algorithms: In GAs and their Application*, pp. 93-100, 1985.
- [17] J. Horn and N. Nafpliotis. *Multiobjective Optimization Using Niche Pareto Genetic Algorithm*, Technical Report, I11iGA1 Report 93005, UIUC, 1993.
- [18] J. Horn, N. Nafpliotis and D. E. Goldberg. "A Niche Pareto Genetic Algorithm for Multiobjective Optimization." In *Proceedings of the 1st IEEE Conference on Evolutionary Computation*, pp. 82-87, 1994.
- [19] J.D. Knowles and D.W. Corne. "Approximating the Non-Dominated Front Using the Pareto Archive Evolution Strategy." *Evolutionary Computation*, Vol. 8, No. 2, pp. 149-172, 2000.
- [20] E. Zitzler and L. Thiele. "Multiobjective Evolutionary algorithms: A Comparative Case Study and the Strength Pareto Approach." *Transactions on Evolutionary Computation*, Vol. 3, No. 4, pp. 257-271, 1999.
- [21] M. Reyes-Sierra and C. A. Coello Coello. "Multi-Objective Particle Swarm Optimizers: A Survey of the State-of-the-Art." *International Journal of Computational Intelligence Research*, Vol. 2, No. 3, pp. 287-308, 2006.
- [22] S. Srisomporn and S. Bureerat. "Geometrical Design of Plate-Fin Heat Sinks Using Hybridization of MOEA and RSM." *IEEE Transactions on Components and Packaging Technologies*, Issue. 2, pp. 351-359, 2008.
- [23] C. M. Fonseca and P. J. Fleming. "Multiobjective Optimization and Multiple Constraint Handling with Evolutionary Algorithms Part 1: A Unified Formulation." *IEEE Trans. Syst. Man Cybern. A*, Vol. 28, pp. 26-37, 1998.
- [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] ศศิธร บัจจุโส. "ทฤษฎีความอลวนกับอุตุนิยมวิทยา." *วารสารวิทยาศาสตร์มข*, ปีที่ 40 ฉบับที่ 1, หน้า 66-74, 2555.
- [26] D. Yang, G. Li and G. Cheng. "On the efficiency of chaos optimization algorithms for global optimization." *Chaos Solitons Fractals*, Vol. 34, pp. 45-53, 2007.
- [27] I. Zelinka. "Investigation on realtime deterministic chaos control by means of evolutionary algorithms." In *Proceedings of 1st IFAC Conference on Analysis and Control of Chaotic Systems*, Reims, France, pp. 211-217, 2006.
- [28] G. Zhenyu, B. Cheng, Y. Min and B. Cao. "Self-Adaptive Chaos Differential Evolution." In *Proceedings of International Conference on Natural*



- Computation*, pp. 972-975, 2006.
- [29] G. Stefania and I. Member. "Neuro-Fuzzy Learning and Genetic Algorithm Approach with Chaos Theory Principles Applying for Diagnostic Problem Solving." *In Proceedings of the World Congress on Engineering*, London, U.K. Vol. 1, 2009.
- [30] Y. Wang, J. Zhou, Y. Lu, H. Qin and Y. Zhang. "Chaos Cultural Particle Swarm Optimization and Its Application." *Advances in Neural Networks – ISNN*, pp. 30-40, 2009.
- [31] L. Gao and X. Liu. "A Resilient Particle Swarm Optimization Algorithm Based on Chaos And Applying it to Optimize." *The Fermentation Process International Journal of Information and System Sciences*, Vol. 5, pp. 380-391, 2009.
- [32] K. Deb, L. Thiele, M. Laumanns and E. Zitzler. "Scalable Multi-Objective Optimization Test Problems." *CEC 2002, IEEE Press*, pp. 825-830, 2002.
- [33] K. Deb, A. Pratap, S. Agarwal and T. Meyarivan. "A fast and elitist multiobjective genetic algorithm: NSGAII." *IEEE Transactions On Evolutionary Computation*, Vol. 6, No. 2, pp. 182-197, 2002.