

การหาค่าเหมาะสมที่สุดที่มีหลาย วัตถุประสงค์ด้วยขั้นตอน วิธีด้านวิวัฒนาการ Multi-Objective Optimization using Evolutionary Algorithms

ภรณ์ยา อัมมฤครัตน์ (Paranya Ammaruekarat)*

และ พยุง มีสัจ (Phayung Meesad)**

บทคัดย่อ

โดยทั่วไปปัญหามากมายในโลกแห่งความจริงเป็นปัญหาหลายวัตถุประสงค์ ขั้นตอนวิธีด้านวิวัฒนาการมีลักษณะหลายอย่างที่เหมาะสมสำหรับปัญหาแบบหลายวัตถุประสงค์ กลยุทธ์ในการค้นหาที่ใช้สำหรับการหาค่าเหมาะสมที่สุดที่มีหลายวัตถุประสงค์มีมานานกว่าสิบปี และการประยุกต์ใช้ขั้นตอนวิธีวิวัฒนาการในการหาค่าเหมาะสมที่สุดที่มีหลายวัตถุประสงค์ได้รับความสนใจจากนักวิจัยเป็นอย่างมาก และในบทความนี้ได้แสดงภาพรวมเกี่ยวกับทฤษฎีและวิธีการในการหาค่าเหมาะสมที่สุดที่มีหลายวัตถุประสงค์ ได้แก่ หลักการพื้นฐานของการหาค่าเหมาะสมที่สุดที่มีหลายวัตถุประสงค์ และขั้นตอนวิธีวิวัฒนาการซึ่งจะนำเสนอในด้านแนวคิดของขั้นตอนวิธี อย่างเช่น การกำหนดค่าความแข็งแรงสำหรับปัญหาการหาค่าเหมาะสมที่สุดที่มีหลายวัตถุประสงค์

คำสำคัญ: การหาค่าเหมาะสมที่สุดที่มีหลายวัตถุประสงค์ ขั้นตอนวิธีด้านวิวัฒนาการ

Abstract

In general, many problems in the real world are multi-objective problems. Evolutionary algorithms possess several characteristics that are desirable for this type of problem, search strategies has been used for multi-objective

optimization for more than a decade. The application of evolutionary algorithms in multi-objective optimization is receiving very high interest from researchers. In this article presents overview of evolutionary algorithms for multi-objective optimization with the focus on methods and theory, including the basic principles of multi-objective optimization and evolutionary algorithms are presented, and algorithm concepts such as fitness assignment for multi-objective optimization problem.

Keywords: Multi-Objective Optimization, Evolutionary Algorithms.

1. บทนำ

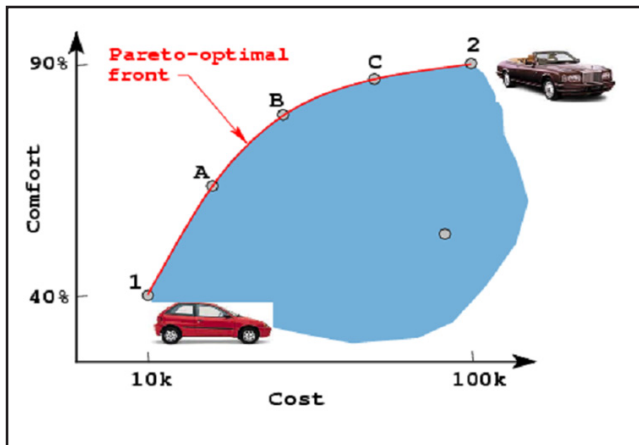
หลายปัญหาในโลกแห่งความจริงไม่ว่าจะเป็นทางด้านอุตสาหกรรมหรือแวดวงวิชาการนั้นจะเป็นปัญหาแบบหลายวัตถุประสงค์ และแต่ละวัตถุประสงค์มักก็มีความขัดแย้งกันอยู่ภายใต้เงื่อนไขเฉพาะที่ต้องการคำตอบหลายคำตอบ แต่ละคำตอบอาจจะเป็นคำตอบที่เป็นคู่แข่งกัน ซึ่งรูปแบบคำตอบที่ดีที่สุดของปัญหานั้น จะมีลักษณะพิเศษ คือ เป็นกลุ่มคำตอบที่ดีที่สุด โดยที่คำตอบแต่ละคำตอบนั้นเป็นคำตอบที่ดีที่สุดของปัญหา และไม่มีคำตอบอื่นครอบงำ (Dominate) เลย หรือที่เรียกว่า ฟาเรโดฟรอน (Pareto front) ตัวอย่างปัญหาแบบหลายวัตถุประสงค์ เช่น ปัญหาการจัดการผลิต ปัญหาการมอบหมายงาน หรือตัวอย่างปัญหาแบบหลายวัตถุประสงค์ง่ายๆ ที่พบในชีวิตประจำวัน เช่น ปัญหาในการเลือกซื้อรถยนต์ ตัวอย่างดังภาพที่ 1 หากต้องการซื้อรถยนต์ ส่วนใหญ่จะพิจารณาในด้านของราคากับสมรรถนะของรถยนต์ ซึ่งสามารถสรุปวัตถุประสงค์ได้ 2 วัตถุประสงค์ คือ ต้องการราคาต่ำ ในขณะที่ต้องการสมรรถนะของรถสูง ซึ่งคำตอบที่ได้ไม่ได้มีเพียงแค่คำตอบเดียว เป็นต้น

จากงานวิจัยที่ผ่านมาพบว่าขั้นตอนวิธีด้านวิวัฒนาการ (Evolutionary Algorithms) ได้รับการยอมรับว่าเหมาะสมสำหรับการแก้ปัญหาการหาค่าที่เหมาะสมที่สุดที่มีหลายวัตถุประสงค์ งานวิจัยที่ถูกพัฒนาขึ้นเพื่อแก้ปัญหาการหาค่าที่เหมาะสมที่สุดที่มีหลายวัตถุประสงค์ เป็นที่สนใจจากนักวิจัยเพิ่มขึ้น ไม่ว่าจะเป็นการพัฒนาและปรับปรุงอัลกอริทึม

* คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยราชภัฏนครปฐม

** คณะเทคโนโลยีสารสนเทศ มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ

ด้านวิวัฒนาการ จนได้กลายเป็นเทคโนโลยีที่สำคัญของการแก้ปัญหาการหาค่าที่เหมาะสมที่สุดที่มีหลายวัตถุประสงค์ ที่รู้จักกันทั่วไปในยุคแรกๆ เช่น Vector Evolution Genetic Algorithm (VEGA) [2] Multi-Objective Genetic Algorithm (MOGA) [3] และ Non-dominated Sorting Genetic Algorithm (NSGA) [4] หลังจากนั้นได้มีการนำเสนอ Niched Pareto Genetic Algorithm (NPGA) [5], [6] Pareto Achieved Evolution Strategy (PAES) [7] Strength Pareto Evolutionary Algorithm (SPEA) [8] Particle Swarm Optimizer (PSO) [9] เป็นต้น นอกจากนี้วิธีการที่กล่าวมานี้ยังคงมีอีกหลายวิธีที่ใช้งานในลักษณะเดียวกัน หรือ แม้กระทั่งการปรับปรุงวิธีการเดิมเพื่อเพิ่มสมรรถนะ เช่น SPEA2 [10] และ NSGAI [11] เป็นต้น บทความนี้จะนำเสนอทฤษฎีและวิธีการ หลักการที่ใช้ในการหาค่าเหมาะสมที่สุดที่มีหลายวัตถุประสงค์ เพื่อให้ผู้ที่สนใจนำไปเป็นแนวทางในการทำงาน การแก้ปัญหาให้เหมาะสมยิ่งขึ้น



ภาพที่ 1 ปัญหาการเลือกซื้อรถยนต์ [1]

2. ปัญหาการหาค่าเหมาะสมที่สุดที่มีหลายวัตถุประสงค์

ในการแก้ปัญหาการหาค่าเหมาะสมที่สุดที่มีหลายวัตถุประสงค์ ควรต้องมีการศึกษาทฤษฎีของการหาค่าเหมาะสมที่สุดที่มีหลายวัตถุประสงค์ ในส่วนนี้จะสรุปหัวข้อหลัก ๆ ที่ต้องทำการศึกษาไว้ 3 หัวข้อ ได้แก่ หลักการพื้นฐานของการหาค่าเหมาะสมที่สุด ลักษณะของปัญหาการหาค่าเหมาะสมที่สุดที่มีหลายวัตถุประสงค์ และการวัดสมรรถนะของกลุ่มคำตอบที่ดีที่สุด รายละเอียดดังนี้

2.1 หลักการพื้นฐานของการหาค่าเหมาะสมที่สุด

การหาค่าเหมาะสมที่สุด (Optimization) เป็นวิธีการที่ใช้

ในการหาคำตอบที่ดีที่สุดของปัญหาภายใต้เงื่อนไขหรือข้อจำกัดที่กำหนดขึ้น การหาค่าเหมาะสมที่สุดถือว่าเป็นสิ่งที่ช่วยในการแก้ปัญหาในด้านวิทยาการคอมพิวเตอร์ ปัญญาประดิษฐ์ การวิจัยการดำเนินงาน และสาขาอื่นที่เกี่ยวข้องได้เป็นอย่างดี [12] ปัญหาการหาค่าเหมาะสมที่สุด แบ่งออกเป็น 2 รูปแบบตามการพิจารณาฟังก์ชันวัตถุประสงค์หรือฟังก์ชันเป้าหมาย คือ การหาค่าเหมาะสมที่สุดที่พิจารณาฟังก์ชันวัตถุประสงค์เพียงวัตถุประสงค์เดียว (Single Objective Optimization Problem) ส่วนปัญหาที่พิจารณาฟังก์ชันวัตถุประสงค์มากกว่า 1 ฟังก์ชันวัตถุประสงค์พร้อมๆ กันในรูปแบบปัญหาลักษณะนี้อาจมีฟังก์ชันวัตถุประสงค์ที่มีความขัดแย้งกันหรือเป็นไปในแนวทางเดียวกัน และเรียกว่าเป็นปัญหาการหาค่าเหมาะสมที่สุดที่มีหลายวัตถุประสงค์ (Multi-Objective Optimization Problem) หรือปัญหาการหาค่าเหมาะสมที่สุดที่พิจารณาหลายเกณฑ์ (Multi-Criteria Optimization Problem) หรือปัญหาการหาค่าเหมาะสมที่สุดแบบเวกเตอร์ (Vector Optimization Problem) ซึ่งการหาค่าเหมาะสมที่สุดนี้จะประกอบด้วย เวกเตอร์ตัวแปรตัดสินใจ (Vector of Decision Variable) ข้อจำกัด (Constraints) และ เวกเตอร์ฟังก์ชัน (Vector Function) ที่สามารถเรียกว่าเป็น ฟังก์ชันวัตถุประสงค์ (Objective Function) โดยมากฟังก์ชันวัตถุประสงค์จะเกี่ยวข้องกับการหาค่ามากที่สุด (Maximization) หรือน้อยที่สุด (Minimization) ของฟังก์ชันวัตถุประสงค์นั้น

2.2 ลักษณะของปัญหาการหาค่าเหมาะสมที่สุดที่มีหลายวัตถุประสงค์

การแก้ปัญหาการหาค่าเหมาะสมที่สุดที่มีหลายวัตถุประสงค์ เป็นการค้นหาเซตคำตอบภายในพื้นที่ของคำตอบที่เป็นไปได้ เพื่อต้องการหาค่าที่ต่ำที่สุด หรือค่าสูงสุดของฟังก์ชันวัตถุประสงค์ในแต่ละฟังก์ชันพร้อมๆ กัน โดยผลลัพธ์ที่ได้จากการแก้ปัญหา เรียกว่า เซตกลุ่มคำตอบที่ดีที่สุด ปัญหาแบบหลายวัตถุประสงค์ เป็นปัญหาการออกแบบที่มีหลายวัตถุประสงค์ ประกอบด้วย m วัตถุประสงค์ และตัวแปรตัดสินใจ [13] 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 $i = 1, 2, \dots, m$ ดังนั้นรูปแบบปัญหาการหาค่าเหมาะสมที่สุดที่มีหลาย

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

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

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

ปกติโดยทั่วไปรูปแบบปัญหาการหาค่าเหมาะสมที่สุดที่มีหลายวัตถุประสงค์ มี 3 รูปแบบ ที่เป็นไปได้ ดังนี้

- 1) ทุกฟังก์ชันวัตถุประสงค์ต้องการหาคำตอบที่น้อยที่สุด
- 2) ทุกฟังก์ชันวัตถุประสงค์ต้องการหาคำตอบที่มากที่สุด
- 3) แบบขัดแย้งกัน คือบางฟังก์ชันวัตถุประสงค์ต้องการหาคำตอบที่น้อยที่สุด บางฟังก์ชันวัตถุประสงค์ต้องการหาคำตอบที่มากที่สุด

ในการหาค่าเหมาะสมที่สุดที่มีหลายวัตถุประสงค์สามารถแปลงฟังก์ชันวัตถุประสงค์ให้มีรูปแบบเป็นการหาคำตอบที่น้อยที่สุดหรือมากที่สุดได้ ตัวอย่างเช่น การแปลงทุกฟังก์ชันวัตถุประสงค์ที่มีการหาคำตอบที่น้อยที่สุดให้เป็นรูปแบบการหาคำตอบที่มากที่สุด เป็นต้น

สำหรับตัวอย่างคำตอบจากปัญหาการหาค่าฟังก์ชันวัตถุประสงค์ในที่นี้เป็นการหาคำตอบที่น้อยที่สุด จากสมการที่ (4) ซึ่งการค้นหาคำตอบจะถูกกำหนดจากเวกเตอร์ของตัวแปรตัดสินใจ ภายใต้ข้อจำกัดที่จะเป็นสิ่งที่ใช้ในการกำหนดขอบเขตที่น้อยที่สุด

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

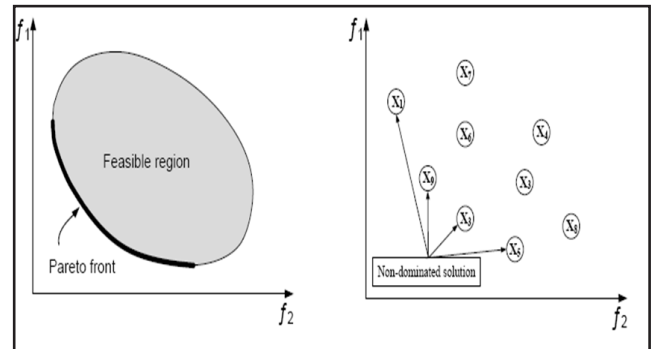
ข้อจำกัด $g(x) \leq 0$

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) หรือเซตคำตอบที่ไม่ถูกครอบงำจากทุกคำตอบ (Non-dominated Set) ดังแสดงในภาพที่ 2



ภาพที่ 2 พารेटอฟรอนของฟังก์ชันวัตถุประสงค์ f_1 และ f_2 [14]

Fonseca and Fleming [15] ได้ให้นิยามของการครอบงำแบบพารेट (Pareto Dominance) และคำตอบที่เหมาะสมที่สุดแบบพารेट (Pareto Optimality) ไว้ดังนี้

นิยาม 1 : การครอบงำแบบพารेट
ให้เวกเตอร์วัตถุประสงค์ $u = (u_1, \dots, u_m)$ ซึ่งครอบงำ $v = (v_1, \dots, v_m)$ ก็ต่อเมื่อ u น้อยกว่า v บางส่วน ดังสมการที่ (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 : คำตอบที่เหมาะสมที่สุดแบบพารेट
โซลูชัน $x_u \in [a, b]$ เป็นคำตอบที่เหมาะสมที่สุดแบบพารेट ถ้าไม่สามารถหาคำตอบ $x_v \in [a, b]$ ซึ่ง $v = f(x_v)$ ครอบงำ $u = f(x_u)$

2.3 การวัดสมรรถนะของกลุ่มคำตอบที่ดีที่สุด

หนึ่งในความแตกต่างของการแก้ปัญหาแบบหลายวัตถุประสงค์ กับปัญหาแบบวัตถุประสงค์เดียว ก็คือคำตอบที่ได้จากการค้นหา เนื่องจากผลลัพธ์ที่ได้จากการแก้ปัญหาแบบ วัตถุประสงค์เดียวจะได้คำตอบเพียงคำตอบเดียว ส่วนปัญหาแบบหลายวัตถุประสงค์ จะได้ขอบเขตกลุ่มคำตอบที่ดีที่สุด ดังนั้นขอบเขตกลุ่มคำตอบที่ได้นั้นจะยอมรับได้ว่าเป็นขอบเขตกลุ่มคำตอบที่ดีก็ต่อเมื่อสามารถวัดสมรรถนะของคำตอบว่าใกล้เคียงกับขอบเขตกลุ่มคำตอบที่ดีที่สุดที่แท้จริง (True Pareto Optimal) ดังนั้นการวัดสมรรถนะของกลุ่มคำตอบที่ดีที่สุดเป็นการวัดคุณภาพของคำตอบที่หาได้จากขั้นตอนวิธี เช่น การลู่เข้าสู่กลุ่มคำตอบที่แท้จริง และการกระจาย

ของกลุ่มคำตอบที่แท้จริง เป็นต้น สามารถดูรายละเอียดตัววัดสมรรถนะแบบอื่นได้จากเอกสารงานวิจัย [16] ในบทความนี้จะกล่าวถึงรายละเอียดของตัววัดสมรรถนะสองด้าน ดังนี้

2.3.1 การวัดสมรรถนะของคำตอบด้านการเข้าสู่กลุ่มคำตอบที่แท้จริง (Convergence Measurement) [16] เป็นตัวชี้วัดการเข้าสู่ของกลุ่มคำตอบโดยเปรียบเทียบระยะทางจากสมาชิกของกลุ่มคำตอบทุกคำตอบที่หาได้ (Obtained Solution Set) กับกลุ่มคำตอบที่แท้จริง (True Pareto Set) ดังสมการ (6) และ (7)

$$convergence(A) = \frac{\sum_{i=1}^{|A^*|} dt_i}{|A^*|} \quad (6)$$

$$dt_i = \min_{j=1}^{|A^*|} \sqrt{\sum_{k=1}^k \left[\frac{f_k(x) - f_k(y)}{f_k^{\max} - f_k^{\min}} \right]^2} \quad (7)$$

เมื่อ $convergence(A)$ คือ ตัวชี้วัดการเข้าสู่ของกลุ่มคำตอบที่แท้จริง

A^* คือ เซตคำตอบที่แท้จริง

$f_k^{\min}$ คือ ค่าฟังก์ชันวัตถุประสงค์ k^{th} ที่มีค่าน้อยที่สุด

$f_k^{\max}$ คือ ค่าฟังก์ชันวัตถุประสงค์ k^{th} ที่มีค่ามากที่สุด

k คือ จำนวนฟังก์ชันวัตถุประสงค์

$|A^*|$ คือ จำนวนคำตอบที่แท้จริงในเซต A

dt_i คือ ระยะห่างระหว่างคำตอบที่ไม่ถูกรอบงำ i^{th} จากขอบของกลุ่มคำตอบที่แท้จริง (y) และโซลูชันที่ได้ (x)

ถ้าค่าสมรรถนะชนิดนี้มีค่าเข้าใกล้ศูนย์ ถือว่ากลุ่มคำตอบขั้นตอนวิธีนั้นเป็นกลุ่มคำตอบที่เข้าสู่ใกล้กลุ่มคำตอบที่แท้จริง

2.3.2 การวัดสมรรถนะของคำตอบด้านการกระจายของกลุ่มคำตอบที่หาได้ (Spread Measurement) [16] เป็นตัวชี้วัดที่บอกการกระจายของกลุ่มคำตอบดังสมการ (8) และสมการที่ (9)

$$spread(A) = \frac{s d_f + s d_l + \sum_{i=1}^{|A|-1} \|s d_i - s d\|}{s d_f + s d_l + (|A|-1) s d} \quad (8)$$

$$s d_i = \sqrt{\sum_{k=1}^k \left[\frac{f_k(x_i) - f_k(x_{i+1})}{f_k^{\max} - f_k^{\min}} \right]^2} \quad (9)$$

เมื่อ $spread(A)$ คือ ตัวชี้วัดที่บอกการกระจายของกลุ่มคำตอบ

$s d_f$ และ $s d_l$ คือ ระยะห่างของคำตอบปลายสุดทั้งสองด้านของเส้นขอบเขตกลุ่มคำตอบที่หาได้

$s d_i$ คือ ระยะของคำตอบที่อยู่ต่อเนื่องกันในเซตคำตอบที่ดีที่สุดที่หาได้

$|A|$ คือ จำนวนคำตอบที่หาได้

$s d$ คือ ค่าเฉลี่ยของระยะทาง $s d_i$

ถ้าค่าสมรรถนะชนิดนี้ มีค่าเข้าใกล้ศูนย์ ถือว่ากลุ่มคำตอบขั้นตอนวิธีนั้นจะมีคำตอบที่มีลักษณะการกระจายที่มีลักษณะสม่ำเสมอตลอดเส้นขอบเขตของกลุ่มคำตอบที่ดีที่สุด

3. การแก้ปัญหาการหาค่าเหมาะสมที่สุดที่มีหลายวัตถุประสงค์ด้วยขั้นตอนวิธีด้านวิวัฒนาการ

ขั้นตอนวิธีด้านวิวัฒนาการได้นำมาประยุกต์ใช้ในปัญหาการหาค่าเหมาะสมที่สุดที่มีหลายวัตถุประสงค์ โดยที่มีวัตถุประสงค์เพื่อค้นหาขอบเขตกลุ่มคำตอบที่ดีที่สุดที่มีลักษณะการกระจายแบบสม่ำเสมอและใกล้เคียงกับขอบเขตกลุ่มคำตอบที่ดีที่สุดที่แท้จริง ซึ่งถือว่ามีความซับซ้อนมากกว่าปัญหาการหาค่าเหมาะสมที่สุดแบบวัตถุประสงค์เดียว ซึ่งในส่วนนี้จะกล่าวถึง ขั้นตอนวิธีด้านวิวัฒนาการ และการแก้ปัญหาการหาค่าเหมาะสมที่สุดที่มีหลายวัตถุประสงค์รายละเอียดดังนี้

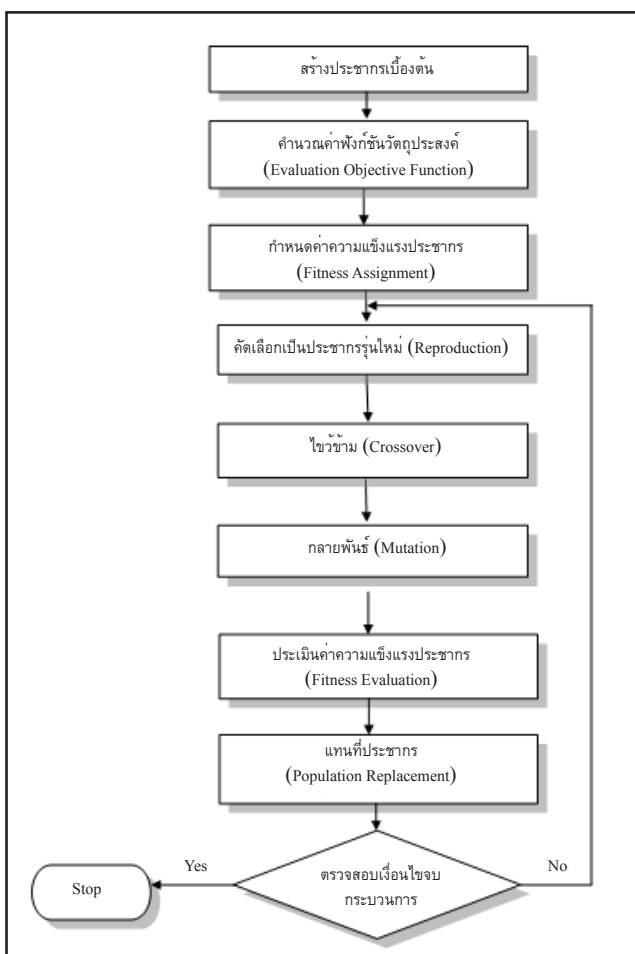
3.1 ขั้นตอนวิธีด้านวิวัฒนาการ

ขั้นตอนวิธีด้านวิวัฒนาการ [17] มีขั้นตอนในการทำงานหลัก คือการสร้าง กลุ่มของตัวค้นหาผลลัพธ์ (Population of Solutions) แล้วใช้หลักเกณฑ์ที่ต่างกันในการค้นหาแบบสุ่ม (Stochastic Search) ที่ทำการลอกเลียนแบบการพัฒนาตัวเองตามธรรมชาติ เช่น การที่มดพยายามหาเส้นทางที่สั้นที่สุดไปสู่แหล่งอาหาร หรือการที่นกพยายามหาจุดหมายปลายทางระหว่างการอพยพ ซึ่งขั้นตอนดังกล่าว จะถูกทำซ้ำไปซ้ำมาจนกระทั่งได้ตามเงื่อนไขของการหยุดการทำงาน โดยขั้นตอนวิธีเชิงวิวัฒนาการนั้นใช้กระบวนการที่ได้รับแรงบันดาลใจจากการวิวัฒนาการทางชีววิทยา [18] ในการคัดเลือกประชากรที่เหมาะสมตามสภาพแวดล้อมที่กำหนดไว้ ขั้นตอนวิธีด้านวิวัฒนาการนี้มักจะใช้ได้ดีสำหรับการหาผลเฉลยของปัญหาในทุกๆ ด้านเนื่องจากสามารถพัฒนาผลเฉลยที่มีไปยังผลเฉลยที่ถูกต้องได้อย่างรวดเร็วทำให้ประสบความสำเร็จในหลายๆ ด้านของปัญหา เช่น วิศวกรรม ศิลปกรรม ชีวภาพ เศรษฐศาสตร์ การตลาด การค้นคว้าวิจัย การออกแบบหุ่นยนต์วิทยาศาสตร์ ด้านสังคม ฟิสิกส์ รัฐศาสตร์ และเคมี เป็นต้น [19], [13]

ขั้นตอนวิธีด้านวิวัฒนาการ ได้แบ่งออกเป็น 2 ด้าน คือ วิวัฒนาการด้านพันธุกรรม เช่น ขั้นตอนวิธีเชิงพันธุกรรม

(Genetic Algorithms: GAs) ขั้นตอนวิธีมีมีติก (Memetic Algorithms : MAs) เป็นต้น และวิวัฒนาการด้านพฤติกรรม เช่น การหาค่าเหมาะสมด้วยวิธีอนุภาคเคลื่อนที่ (Particle Swarm Optimization : PSO) การหาค่าเหมาะสมที่สุดแบบอาณานิคมมด (Ant Colony Optimization : ACO) ขั้นตอนวิธีซฟเฟิล ฟรอกลีปิง (Shuffle Frog Leaping Algorithms : SFL) เป็นต้น

ซึ่งบทความนี้ได้นำเสนอขั้นตอนวิธีด้านวิวัฒนาการเน้นในด้านวิวัฒนาการด้านพันธุกรรม โดยโครงสร้างพื้นฐานวิวัฒนาการด้านพันธุกรรม แสดงดังภาพที่ 3 จากภาพอธิบายขั้นตอนการทำงานอย่างคร่าวๆ ได้ดังนี้



ภาพที่ 3 โครงสร้างพื้นฐานของขั้นตอนวิธีด้านวิวัฒนาการ

ขั้นตอนที่ 1: ทำการสร้างประชากรคำตอบเบื้องต้น (Initialization Population)

ขั้นตอนที่ 2: คำนวณค่าฟังก์ชันวัตถุประสงค์ของประชากรคำตอบ (Evaluation Objective Function)

ขั้นตอนที่ 3: การกำหนดค่าความแข็งแรง (Fitness

Assignment) ซึ่งส่วนนี้เป็นส่วนสำคัญของการแก้ปัญหาการหาค่าเหมาะสมที่สุดที่มีหลายวัตถุประสงค์

ขั้นตอนที่ 4: การคัดเลือกคำตอบ (Selection) มาคัดเลือกโดยดูจากค่าความแข็งแรง (Fitness Value) ของสตริงคำตอบแต่ละตัว

ขั้นตอนที่ 5: การแลกเปลี่ยนพันธุกรรม (Crossover) เป็นกระบวนการที่นำสตริงคำตอบที่ถูกจับคู่ไว้มาแลกเปลี่ยนส่วนซึ่งกันและกัน เพื่อให้เกิดสตริงคำตอบใหม่ขึ้น

ขั้นตอนที่ 6: กระบวนการกลายพันธุ์ (Mutation) เพื่อให้เกิดสตริงคำตอบใหม่ขึ้น

ขั้นตอนที่ 7: การประเมินค่าความแข็งแรงประชากร (Fitness Evaluation) ทำการคัดเลือกโครโมโซมที่มีความแข็งแรงสูง

ขั้นตอนที่ 8: การแทนที่ประชากร (Population Replacement) โดยทำการรวมโครโมโซมคำตอบจากโครโมโซมเริ่มต้น และโครโมโซมรุ่นลูก จะนำมาทำการรวมกันและทำการเก็บค่าที่ดีที่สุดเท่ากับจำนวนโครโมโซมเริ่มต้น

ขั้นตอนที่ 9: ดูว่าการคำนวณนั้นครบจำนวนสูงสุดของคำตอบที่ต้องการ และคำนวณค่าฟังก์ชันหรือจำนวนเงื่อนไขที่กำหนดไว้หรือไม่ ถ้าน้อยกว่าให้ทำขั้นตอนที่ 4-8 ถ้าไม่ใช่ ให้หยุด กระบวนการค้นหาคำตอบ (Stop)

3.2 การแก้ปัญหาการหาค่าเหมาะสมที่สุดที่มีหลายวัตถุประสงค์

จากขั้นตอนการทำงานของขั้นตอนวิธีด้านวิวัฒนาการ สิ่งที่สำคัญสำหรับปัญหาการหาค่าเหมาะสมที่สุดที่มีหลายวัตถุประสงค์ คือ การกำหนดค่าความแข็งแรงให้กับสมาชิกของกลุ่มประชากรแต่ละตัวได้อย่างเหมาะสมและสอดคล้องกับกลุ่มคำตอบที่ดีที่สุด ซึ่งวิธีการกำหนดค่าความแข็งแรงในปัญหาที่มีหลายวัตถุประสงค์ สามารถสรุปหลักๆ ได้ 4 วิธีดังต่อไปนี้

1) วิธีการคำนวณค่าแบบเวกเตอร์ (Vector Evaluation Approach) เป็นวิธีการกำหนดค่าแข็งแรงวิธีแรกที่ขยายผลให้ขั้นตอนวิธีเชิงพันธุกรรม สามารถแก้ปัญหาการหาค่าเหมาะสมที่สุดที่มีหลายวัตถุประสงค์ได้ ในที่ชื่อว่า Vector Evolution Genetic Algorithm (VEGA) [2]

2) วิธีโปรแกรมมิ่ง (Goal Programming Approach) วิธีการนี้อาศัยการจัดอันดับเพื่อประเมินความ สามารถของสมาชิกคำตอบ ได้นำมาใช้ในขั้นตอนวิธีด้านวิวัฒนาการที่

ชื่อว่า Multi-Objective Goal Programming (MOGP) ตัวอย่างจากงานวิจัยของ Lee และ Clayton [20] Singh และ Agarwal [21] Premchandra [22] Lyu และคณะ [23]

3) วิธีการรวมฟังก์ชันโดยอาศัยการให้น้ำหนัก (Weighted Sum Approach) เป็นวิธีการที่ไม่ยุ่งยากและง่ายต่อการนำไปใช้ วิธีนี้อาศัยการให้น้ำหนักกับฟังก์ชันวัตถุประสงค์แต่ละอย่าง แล้วนำเอาน้ำหนักและฟังก์ชันวัตถุประสงค์ทั้งหมดมารวมกัน เพื่อสร้างค่าวัตถุประสงค์เป็นค่าเดียว แต่ปัญหาของวิธีนี้คือ ถ้าหากไม่มีข้อมูลเพียงพอก็จะเกิดความยุ่งยากในการกำหนดน้ำหนักของแต่ละวัตถุประสงค์ ตัวอย่างงานวิจัยที่ใช้วิธีนี้ เช่น Paulo และคณะ [24] Ishibuchi และคณะ [25]

4) วิธีเชิงกลุ่มที่ดีที่สุด (Pareto-based Approach) วิธีนี้จะใช้การจัดอันดับแบบพาเรโต ในการสร้างความสัมพันธ์ระหว่างค่าฟังก์ชันวัตถุประสงค์และค่าความแข็งแรง โดยใช้หลักการครอบงำแบบพาเรโต เพื่อคำนวณค่าความแข็งแรงสำหรับวิธีการกำหนดค่าความแข็งแรงแบบนี้จะทำให้คำตอบที่ดีที่สุดของปัญหามากกว่าหนึ่งคำตอบ และอยู่ในรูปแบบที่เป็นเซตหรือกลุ่มคำตอบที่ดีที่สุด ตัวอย่างงานวิจัยที่ใช้วิธีนี้ เช่น Ammaruekarat และ Meesad [26], [27] ภรณ์ยา และพยุ่ง [28] Lara และคณะ [29]

จากวิธีการกำหนดค่าความแข็งแรงที่กล่าวมาข้างต้น วิธีการรวมฟังก์ชันโดยอาศัยน้ำหนัก กับวิธีเชิงกลุ่มที่ดีที่สุด เป็นวิธีที่ได้รับความนิยมจากนักวิจัยนำมาแก้ปัญหาการหาค่าเหมาะสมที่สุดที่มีหลายวัตถุประสงค์เป็นจำนวนมาก แต่เมื่อเปรียบเทียบ 2 วิธีนี้ จะพบว่าวิธีการรวมฟังก์ชันโดยอาศัยน้ำหนัก ยังถือว่าไม่ตอบสนองกับรูปแบบปัญหาดังกล่าวมากนัก เพราะค่าวัตถุประสงค์ได้มีการรวมเป็นค่าเดียว และคำตอบที่ได้จากการรวมวัตถุประสงค์จะมีเพียงคำตอบเดียวที่ให้ค่าเหมาะสมที่สุดระหว่างแต่ละวัตถุประสงค์ ส่วนวิธีเชิงกลุ่มที่ดีที่สุด จะให้ผลลัพธ์ที่ได้ตอบสนองกับผู้ตัดสินใจได้ดีกว่า เนื่องจากคำตอบที่ได้มีลักษณะเป็นกลุ่มคำตอบที่หลากหลายกว่า

4. สรุป

เนื้อหาที่กล่าวไว้ในบทความนี้เป็นการแก้ปัญหาการหาค่าเหมาะสมที่สุดที่มีหลายวัตถุประสงค์ด้วยขั้นตอนวิธีด้านวิวัฒนาการ โดยความแตกต่างของปัญหาที่พิจารณาจาก

ฟังก์ชันวัตถุประสงค์เดียวและจากหลายฟังก์ชันวัตถุประสงค์จะต่างกันในด้านของผลลัพธ์ของคำตอบที่ได้ และในส่วนขั้นตอนวิธีด้านวิวัฒนาการที่นำมาใช้ในการแก้ปัญหาสิ่งที่สำคัญสำหรับปัญหาการหาค่าเหมาะสมที่สุดที่มีหลายวัตถุประสงค์คือ การกำหนดค่าความแข็งแรงให้กับสมาชิกของกลุ่มประชากรแต่ละตัวได้อย่างเหมาะสมและสอดคล้องกับกลุ่มคำตอบที่ดีที่สุด จึงจะทำให้ผลลัพธ์ที่ได้ตอบสนองกับผู้ตัดสินใจมากขึ้น เนื่องจากคำตอบที่ได้มีลักษณะเป็นกลุ่มคำตอบที่หลากหลายและสามารถเลือกคำตอบนั้นภายใต้สถานการณ์นั้นๆ อีกด้วย

5. เอกสารอ้างอิง

- [1] K. Deb, *Multi-Objective Optimization using Evolutionary Algorithms*, Wiley 2001.
- [2] 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.
- [3] 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.
- [4] N. Srinivas, and K. Deb, "Multiobjective Optimization Using Non-Dominated Genetic Algorithms," *Evolutionary Computation*, vol. 2(3), pp. 221-248, 1994.
- [5] J. Horn, and N. Nafpliotis, *Multiobjective Optimization Using Niche Pareto Genetic Algorithm*, Tech. Report, I11iGA1 Report 93005, UIUC, 1993.
- [6] 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.
- [7] 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.
- [8] 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.
- [9] 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.
- [10] 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.
- [11] 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.
- [12] ปารเมศ ชุติมา, เพ็ญพักตร์ ปิ่นภูมิรี และ พัฒนพงษ์ แสงหัตถ์วัฒนา, “การประยุกต์ใช้เมมเมติกอัลกอริทึมสำหรับปัญหาการจัดลำดับสายการประกอบแบบผลิตภัณฑ์ผสมที่มีหลายวัตถุประสงค์ในระบบผลิตแบบทันเวลาพอดี,” *การประชุมวิชาการช่างงานวิศวกรรมอุตสาหกรรม*, pp. 72-77, 2550.
- [13] 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.
- [14] Srisomporn, S. and Bureerat, S., “Geometrical Design of Plate-Fin Heat Sinks Using Hybridization of MOEA and RSM,” *IEEE Transactions on Components and Packaging Technologies* 2, pp. 351-359, 2008.
- [15] 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 January, 1998.
- [16] R. Kumar, PK. Singh, “Pareto evolutionary Algorithm hybridized with local search for biobjective TSP,” In: C Grosan, A braham, H Ishibuchi (eds) *Hybrid Evolutionary Algorithms*, Springer, Berlin, pp. 361-98, 2007.
- [17] G.S. Hornby, and J.B. Pollack, “Creating high-level components with a generative representation for body-brain evolution,” *Artificial Life*. 3, pp. 223-246, 2002.
- [18] J. Clune, B. Beckmann, C. Ofria, and R. Pennock, “Evolving Coordinated Quadruped Gaits with the HyperNEAT Generative Encoding,” *Proceedings of the IEEE Congress on Evolutionary Computing Special Section on Evolutionary Robotics*, Trondheim, Norway, 2009.
- [19] 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, *LNCs*, Springer, pp. 358-367, 2008.
- [20] S. M. Lee, and E. R. Clayton, “A goal programming model for academic resource allocation,” *Management Science*, Vol. 18, No. 8, pp. 395-408, 1972.
- [21] N. Singh, and S. K. Agarwal, “Optimum design of an extended octogonal ring by goal programming,” *International Journal of Production Research*, Vol. 21, No. 6, pp. 891-898, 1983.
- [22] I. M. Premchandra, “A goal programming model for activity crashing in project networks,” *International Journal of Operations and Production Management*, Vol. 13, No. 6, pp. 79-85, 1993.
- [23] J. Lyu, A. Gunasekaran, C. Y. Chen, and C. Kao, “A goal programming model for the coal blending problem,” *Computers and Industrial Engineering*, Vol. 28, No. 4, pp. 861- 868, 1995.
- [24] V.W. Paulo, T. W. Radtke, and S. Robert, “A Multi-objective Memetic Algorithm for Intelligent Feature Extraction,” *Evolutionary Multi-Criterion Optimization*, pp. 767–781, 2005.
- [25] H. Ishibuchi, Y. Hitotsuyanagi, N. Tsukamoto, “Implementation of Multiobjective Memetic Algorithms for Combinatorial Optimization Problems : A Knapsack Problem Case Study,” *Multi-objective Memetic*

- Algorithms*, pp. 27-49, 2009.
- [26] P. Ammaruekarat, and P. Meesad, "Multi-Objective Chaos Memetic Algorithm for DTLZ Problems," *Advanced Materials Research*, Vol. 403-408, pp. 3676-3681, 2012.
- [27] 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.
- [28] ภรณ์ยา อัมฤครัตน์ และพยุ่ง มีสัจ, "การหาค่าที่เหมาะสมที่สุดด้วยขั้นตอนวิธีเคออสมีมติกแบบหลายวัตถุประสงค์." *วารสารเทคโนโลยีสารสนเทศ*, ฉบับที่ 1, หน้า 54-61, มกราคม - มิถุนายน 2555.
- [29] A. Lara, G. Sanchez, A. C. C. Coello, and O. Schütze, "HCS : A New Local Search Strategy for Memetic Multiobjective Evolutionary Algorithms," *IEEE Transactions on Evolutionary Computation*, Vol.14, 2010.
-