

การเพิ่มความสามารถในการค้นหาพื้นที่ใกล้เคียงของ เจเนติก อัลกอริทึม โดยใช้เทคนิค พาติเคิล สวอม ออปติไมเซชัน Improve Neighbor Search Ability of Genetic Algorithms Using Particle Swarm Optimization Technique

อรรถนิติ วงศ์จักร (Auttanit Wongjak)* กฤตวัน ศิริบุญ (Kritawan Siriboon)* และ
บุญธีร์ เครือตราชู (Boontee Kruatrachue)*

บทคัดย่อ

การในบทความฉบับนี้ ผู้วิจัยได้ประยุกต์ใช้วิธีการทางวิวัฒนาการแบบผสมระหว่าง เจเนติก อัลกอริทึม (Genetic Algorithms, GAs) ร่วมกับวิธีการพาติเคิล สวอม ออปติไมเซชัน (Particle Swarm Optimization, PSO) โดยใช้ชื่อเรียกวิธีการแบบผสมนี้ว่า Swarm Genetic Algorithms (SGA) วิธีการนี้ได้รวมเอาข้อดีของ GAs เข้ากับ PSO เพื่อทำการลดโอกาสการติดทั้งประชากรที่มีโอกาสเข้าสู่คำตอบที่ดีขึ้นก่อนเวลาอันควรรวมทั้งข้อบกพร่องของสวอมเพื่อให้สามารถเข้าถึง โลคอล ออปติมา (local optima) ได้เร็วขึ้นกว่า GAs แบบเดิม โดย SGA จะถูกนำไปประยุกต์ใช้ในปัญหาการเลือกโปรโตไทป์ ประสิทธิภาพของ SGA จะนำไปเปรียบเทียบกับผลลัพธ์กับ GAs ในปัญหาข้างต้นโดยทำการเทียบผลที่ได้ในหลายๆ เซตทดลอง ซึ่งผลการทดลองแสดงให้เห็นว่า SGA สามารถค้นหาคำตอบได้เร็วและดีกว่า GAs แบบเดิม

คำสำคัญ: เจเนติก อัลกอริทึม พาติเคิล สวอม ออปติไมเซชัน การเลือกโปรโตไทป์

Abstract

A new evolutionary learning method based on a hybrid of genetic algorithms (GAs) and particle swarm optimization (PSO) is proposed in this paper, The proposed algorithms is called Swarm Genetic Algorithms (SGA). SGA combining the strengths of GAs with local neighbor search ability of

PSO. PSO enhances GAs in the deselection of immature population and faster local optima search. SGA is applied to the minimal consistent prototype subset identification (MCSI) problem. The performance of SGA is compared to GAs in several MCSI test sets to demonstrate its improvement. The result shows that SGA can find answers faster and better than conventional GAs.

Keyword: Genetic algorithms, Particle swarm optimization, Prototype selection.

1. บทนำ

เจเนติก อัลกอริทึม และ พาติเคิล สวอม ออปติไมเซชัน เป็นวิธีการทางวิทยาศาสตร์ที่ใช้ในการหาผลลัพธ์ด้วยการประมาณค่าเพื่อให้ได้คำตอบที่ดีที่สุด (Optimization) ซึ่งวิธีการทั้งสองนี้สามารถนำไปประยุกต์ใช้ในการแก้ปัญหาต่างๆ ได้มากมายขึ้นอยู่กับการนำไปใช้งาน เนื่องจากแต่ละวิธีการมีจุดเด่นและข้อจำกัดที่แตกต่างกันดังนั้นงานวิจัยนี้จึงได้นำเสนอแนวทางประยุกต์ทั้งสองวิธีการรวมเข้าด้วยกันเป็นวิธีการแบบผสม สวอม เจเนติก อัลกอริทึม (SGA) ที่นำ PSO เข้ามาช่วยลดข้อจำกัดของ GAs ในเรื่องการค้นหาคำตอบบริเวณ โลคอล ออปติมา เพื่อเพิ่มความรวดเร็วและแม่นยำในการทำงานให้มากยิ่งขึ้น

การทำงานของ GAs มีลักษณะเป็นการประมาณค่าทุกขั้นตอนและมีเพียงแค่วิธีการกลายพันธุ์เท่านั้นที่สามารถ

* ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ทำให้เกิดการเปลี่ยนแปลงค่าเพื่อค้นหาคำตอบที่อยู่บริเวณใกล้เคียงได้ทว่าอัตราการเกิดการกลายพันธุ์มีความเป็นไปได้ต่ำอีกทั้งยังไม่สามารถระบุได้ว่าการกลายพันธุ์นั้นจะทำให้เกิดการเปลี่ยนแปลงที่ดีขึ้น ส่วนมากแล้วค่าความเหมาะสมที่ได้ภายหลังการกลายพันธุ์มักมีค่าต่ำกว่าค่าเดิม ดังนั้นแม้ว่า GAs จะประมาณค่าคำตอบออกมาได้ใกล้เคียงคำตอบที่ต้องการหรือคำตอบที่ดีที่สุดเพียงใดก็ตามแต่ยังคงต้องใช้วิธีการแบบสุ่มค้นหาคำตอบต่อไปเพราะไม่สามารถค้นหาคำตอบในพื้นที่รอบๆ ด้วยการเคลื่อนที่เข้าหาคำตอบนั้นอย่าง PSO ทำได้ นั่นจึงเป็นที่มาของการใช้วิธี PSO เพื่อช่วยค้นหาคำตอบในพื้นที่ใกล้เคียงจากการสุ่มของ GAs

วิธีการแบบผสมได้รวมเอาการดำเนินการอย่าง การคัดเลือก (Selection) การผสมพันธุ์ (Crossover) และ การกลายพันธุ์ (Mutation) จาก GAs ผสมเข้ากับแนวคิดเรื่องความเร็วและการเปลี่ยนแปลงตำแหน่งของ PSO ใน SGA นั้นประชากรที่ผ่านการคัดเลือกเพื่อเข้าสู่การทำงานรอบใหม่นั้นจะไม่ได้ถูกสร้างขึ้นจากกระบวนการผสมพันธุ์ และ กลายพันธุ์ เพียงอย่างเดียวแต่ยังรวมถึงกระบวนการเคลื่อนที่ของ PSO ด้วย นั่นคือภายหลังเสร็จสิ้นกระบวนการกลายพันธุ์แล้ว ประชากรทั้งหมดจะถูกนำไปเปลี่ยนแปลงตำแหน่งด้วยวิธีการ PSO โดยที่กลุ่มของประชากรจะถือว่าเป็นสวอมทั้งหมดและโครโมโซมแต่ละตัวจะนับเป็นอนุภาคหนึ่งตัว หลังจากผ่านกระบวนการทำงานของ PSO แล้วจึงค่อยทำการคัดเลือกเพื่อคัดประชากรที่มีค่าความเหมาะสมมากที่สุดครึ่งหนึ่งเป็นประชากรเริ่มต้นของการทำงานรอบต่อไป

รายละเอียดของงานวิจัยฉบับนี้ในส่วนของหัวข้อที่ 2 ได้กล่าวถึงเจเนติก อัลกอริทึมและพาติเคิล สวอม ออปติไมเซชัน วิธีการแบบผสม SGA เสนอในหัวข้อที่ 3 ในหัวข้อที่ 4 จะกล่าวถึงการทดลองและสุดท้ายบทสรุปงานวิจัยจะกล่าวถึงในหัวข้อที่ 5

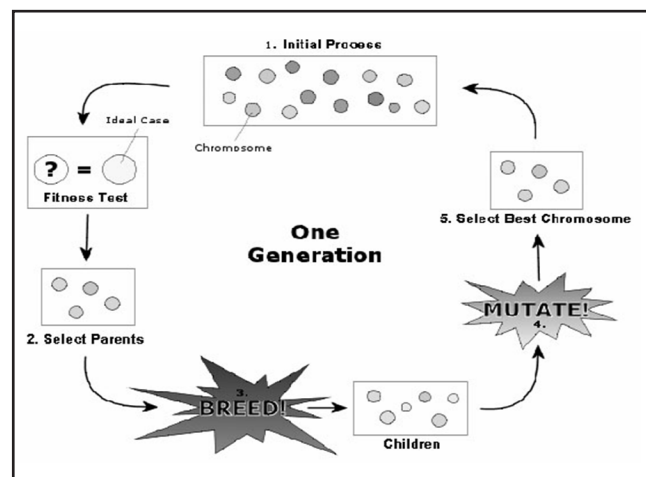
2. งานวิจัยที่เกี่ยวข้อง

2.1 เจเนติกอัลกอริทึม

เจเนติก อัลกอริทึมเป็นวิธีการทางวิวัฒนาการที่ถูกคิดค้นขึ้นมาโดย จอห์น ฮอลแลนด์ (John Holland) [1] ในช่วงต้นของยุค 1960s โดยอ้างอิงหลักการทำงานพื้นฐานเลียนแบบระบบวิวัฒนาการของพันธุกรรมในสิ่งมีชีวิตทั่วไปตามธรรมชาติ หรือเรียกอีกอย่างว่า “ทฤษฎีวิวัฒนาการทางพันธุกรรม”

ของ ชาลส์ ดาร์วิน (Charles Darwin) [2] ซึ่งเป็นทฤษฎีอธิบายเกี่ยวกับวัฏจักรของพันธุกรรมที่ได้รับการคัดสรรทางธรรมชาติ สิ่งมีชีวิตที่สามารถปรับตัวให้เข้ากับสภาพแวดล้อมในปัจจุบันจะมีโอกาสรอดมากกว่าสิ่งมีชีวิตที่ไม่สามารถปรับตัวได้ และเมื่อสิ่งมีชีวิตเหล่านั้นทำการผสมพันธุ์กันก็จะเกิดลูกหลานที่ได้รับการถ่ายทอดความสามารถในการเอาตัวรอดจากพ่อแม่พันธุ์สืบต่อมา

GAs เป็นวิธีการค้นหาคำตอบที่สามารถนำไปประยุกต์ใช้ในการแก้ปัญหาต่างๆ ได้มากมาย เช่น การค้นหาค่าต่ำสุดของสมการคณิตศาสตร์บางตัวหรือปัญหาอย่าง Traveling Sell Man โดยวิธีการทำงานของ GAs นั้นเหมาะสมกับปัญหาที่ยากต่อการคิดหาคำตอบหรือไม่มีหลักวิธีการคิดหาคำตอบที่ตายตัวทว่าทราบถึงวิธีการวัดค่าคำตอบแบบใดเป็นคำตอบที่ใกล้เคียงกับสิ่งที่ต้องการมากกว่ากันทำให้สามารถสุ่มสร้างคำตอบที่ใกล้เคียงกับคำตอบที่ต้องการได้มากยิ่งขึ้นเรื่อยๆ รอบของการทำงาน ซึ่ง GAs จะมีขั้นตอนการทำงานพื้นฐานที่เหมือนกันดังแสดงในภาพที่ 1



ภาพที่ 1 แสดงสัญลักษณ์ของ Sequence Diagram

ขั้นตอนการทำงานของ GAs สามารถเขียนอธิบายได้ดังต่อไปนี้

- 1) สุ่มสร้างคำตอบขึ้นมาเพื่อใช้เป็นประชากรเริ่มต้น
- 2) นำประชากรที่ได้ไปคำนวณหาค่าความเหมาะสม
- 3) สุ่มเลือกประชากรขึ้นมาเป็นพ่อ-แม่พันธุ์ โดยอัตราส่วนความน่าจะเป็นในการสุ่มแปรผันตรงกับค่าความเหมาะสม จากนั้นนำโครโมโซมของพ่อ-แม่มาผสมพันธุ์กันจะได้โครโมโซมรุ่นลูกออกมา
- 4) โครโมโซมในรุ่นลูกมีโอกาสที่จะเกิดการกลายพันธุ์

5) ทำการคัดเลือกประชากรที่มีค่าความเหมาะสมมากที่สุดจากโครโมโซมของพ่อ-แม่-ลูก เพื่อใช้เป็นประชากรในข้อ 1. ของรอบวิวัฒนาการถัดไป

2.2 พาดิเคิล สวอม ออปติไมเซชัน

พาดิเคิล สวอม ออปติไมเซชัน เป็นวิธีการค้นหาคำตอบที่เหมาะสมที่สุด ถูกคิดค้นขึ้นมาโดย อีเบอร์ฮาร์ท (Eberhart) และ เคนเนดี้ (Kennedy) [3], [4] ในปี 1995 โดยมีแรงบันดาลใจในการพัฒนามาจากการสังเกตการเคลื่อนไหวของฝูงนกที่มีลักษณะการเคลื่อนที่สอดคล้องกันในเวลาออกหาอาหาร ฝูงนกเหล่านั้นจะมีการส่งสัญญาณเพื่อสื่อสารกันให้ทราบถึงตำแหน่งที่มีอาหารอยู่และทำการเคลื่อนที่ไปยังแหล่งอาหารที่ได้รับข้อมูลมา เมื่อได้แนวคิดดังนั้น PSO จึงใช้วิธีการค้นหาคำตอบด้วยการใช้อนุภาค (Particles) จำนวนมาก เคลื่อนที่ไปบนพื้นที่ที่ต้องการค้นหา (Search Space) เพื่อค้นหาคำตอบที่ดีที่สุด

นกซึ่งเคลื่อนที่เปรียบได้กับอนุภาคหนึ่งตัวและอนุภาคแต่ละตัวจะจดจำตำแหน่งในปัจจุบันของตนเองเอาไว้ (x_i) พร้อมกับทิศทางและความเร็วในการเคลื่อนที่ของอนุภาค (v_i) PSO เหมือนกับ GAs ตรงที่ทุกๆ รอบการทำงานจะต้องมีสมการที่ใช้วัดค่าความเหมาะสมซึ่งรับค่าตำแหน่งปัจจุบันของอนุภาคเข้าไปเป็นข้อมูล เพื่อวัดหาค่าความเหมาะสมของตำแหน่งที่อนุภาค นั้นอยู่ว่าใกล้เคียงกับคำตอบที่ต้องการเพียงใด ด้วยแนวคิดในการเคลื่อนที่ของฝูงนกยามเมื่อหาอาหารเปรียบได้ดังพื้นที่ที่มีค่าความเหมาะสมมากที่สุดคือพื้นที่ที่มีอาหารอุดมสมบูรณ์และอนุภาคควรเคลื่อนที่โดยคำนึงถึงข้อมูลตรงส่วนนั้น จึงทำให้อนุภาคทุกๆ ตัวจะต้องเก็บข้อมูลตำแหน่งที่ดีที่สุดของตนเอง (PBest) ที่สามารถหาได้ตลอดช่วงระยะเวลาการทำงานพร้อมกับข้อมูลของตำแหน่งที่ดีที่สุดเท่าที่สามารถหาได้จากอนุภาคทุกๆ ตัว (GBest) เอาไว้

ทุกๆ รอบการทำงานเวลา t ความเร็วของการเคลื่อนที่จะถูกเปลี่ยนแปลงโดยใช้ข้อมูลของตำแหน่งที่ดีที่สุดของอนุภาคแต่ละตัว ($\bar{p}_i$), ตำแหน่งที่ดีที่สุดของอนุภาคทั้งหมด ($\bar{p}_g$) ความเร็วของอนุภาคแต่ละตัวที่เปลี่ยนไปจะสามารถคำนวณได้ด้วยสมการที่ (1)

$$\bar{v}_i(t+1) = \bar{v}_i(t) + c_1\phi_1(\bar{p}_i(t) - \bar{x}_i(t)) + c_2\phi_2(\bar{p}_g(t) - \bar{x}_i(t)) \quad (1)$$

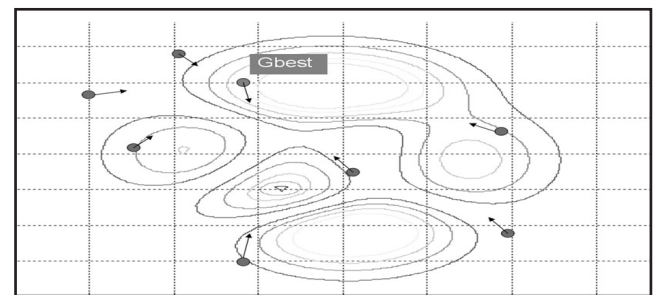
โดยค่า C_1 และ C_2 เป็นค่าคงที่ที่มีค่าเป็นบวก ค่า ϕ_1 และ ϕ_2 เป็นตัวเลขที่ทำการสุ่มขึ้นมาที่มีค่าอยู่ในช่วง $[0, 1]$ ค่าความเร็วจะถูกกำหนดเอาไว้ไม่ให้เกินค่าสูงสุดที่เป็นไปได้ ($\pm v_{max}$) เพื่อจำกัดความเร็วที่มากเกินไปรอบการทำงานอนุภาคจะมีการเปลี่ยนตำแหน่งด้วยความเร็วที่มีอยู่ดังสมการที่ (2)

$$\bar{x}_i(t+1) = \bar{x}_i(t) + \bar{v}_i(t) \quad (2)$$

ด้วยสมการการเคลื่อนที่ของ PSO ทำให้อนุภาคแต่ละตัวมีข้อมูลที่เชื่อมโยงถึงกันแม้ว่าอนุภาคแต่ละตัวจะเคลื่อนที่ไปในทิศทางที่แตกต่างกันก็ตาม สิ่งนี้เป็นผลดีที่ทำให้ PSO มีความสมดุลในการค้นหาคำตอบระหว่างแบบใกล้เคียงและโดยรวม แต่ข้อเสียสำคัญคือ หากความเร็วในการเคลื่อนที่มีค่าน้อยการเข้าสู่คำตอบที่ดีจะทำได้ค่อนข้างช้า

ขั้นตอนการทำงานของ PSO สามารถอธิบายได้ดังนี้

- 1) สร้างอนุภาคโดยสุ่มตำแหน่งเริ่มต้นให้กับอนุภาค
- 2) คำนวณค่าความเหมาะสม ถ้าหากว่าค่าความเหมาะสมที่ได้มีค่ามากกว่าค่าเดิม (PBest) ให้เปลี่ยนไปใช้ค่าที่มากกว่า
- 3) เลือกค่าความเหมาะสมที่ดีที่สุดจากอนุภาคทุกตัวแล้วใช้เป็นค่าที่ดีที่สุด (GBest)
- 4) คำนวณค่าความเร็วใหม่สำหรับทุกๆ อนุภาค
- 5) เคลื่อนที่อนุภาคไปยังตำแหน่งใหม่ ทำวนซ้ำจนได้คำตอบหรือครบจำนวนรอบที่กำหนด ลักษณะการเคลื่อนที่ของพาดิเคิลจะแสดงในภาพที่ 2



ภาพที่ 2 ลักษณะการเคลื่อนที่ของพาดิเคิล

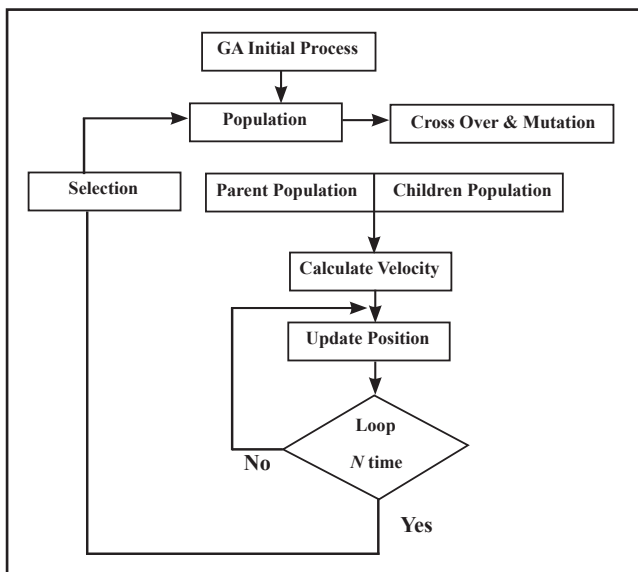
PSO สามารถทำการค้นหาคำตอบด้วยการเคลื่อนที่ไปในบริเวณใกล้เคียงตนเองเท่านั้น ถ้าพื้นที่ๆ ต้องการค้นหามีขนาดใหญ่หรือคำตอบที่ต้องการอยู่ห่างจากตำแหน่งของอนุภาค การค้นหาคำตอบก็จะทำได้ช้าลง ข้อจำกัดส่วนนี้แตกต่างกับ GAs ที่สามารถประมาณค่าคำตอบที่อยู่ห่างออกไปได้อย่างรวดเร็ว แต่ไม่มีวิธีในการค้นหาคำตอบที่อยู่ไกลกับตัวมันแบบ PSO

3. ระบบผสม SGA

อัลกอริทึมแบบผสม SGA จะมีการทำงานหลักอยู่สามส่วนคือ การผสมพันธุ์ การกลายพันธุ์และการทำงานของ PSO โดยทั้ง GAs และ SGA จะใช้วิธีการเข้ารหัสโครโมโซมแบบเดียวกัน ระบบการทำงานของ SGA จะใช้วิธีการทำงานของ GAs เป็นหลักและกระบวนการ PSO จะถูกนำมาใช้กับประชากรทั้งหมดเป็นจำนวน N รอบ จากนั้นจึงค่อยเข้าสู่กระบวนการคัดเลือกประชากรต่อไป

ขั้นตอนการทำงานของ SGA ดังภาพที่ 3 สามารถอธิบายได้ดังนี้

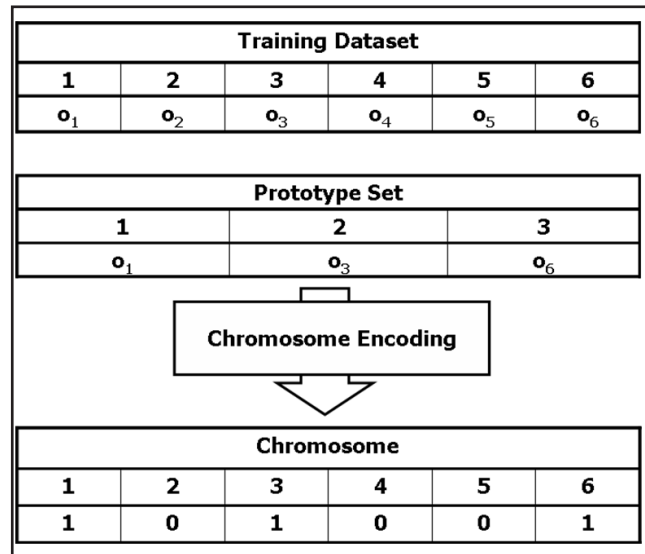
- 1) สุ่มสร้างประชากรขึ้นมาด้วยวิธีการเดียวกับ GAs
- 2) ทำกระบวนการผสมพันธุ์และกลายพันธุ์
- 3) นำประชากรพ่อ-แม่-ลูกไปคำนวณหาค่าความเร็วของประชากรแต่ละตัว
- 4) ทำการเปลี่ยนแปลงตำแหน่งของประชากรทั้งหมด
- 5) วนการทำงานในข้อ 3. จนครบรอบที่กำหนด จากนั้นนำประชากรทั้งหมดเข้ากระบวนการคัดเลือก เพื่อคัดประชากรที่มีค่าความเหมาะสมมากที่สุดครั้งหนึ่งเป็นประชากรเริ่มต้นในข้อ 1. ของรอบการทำงานต่อไป



ภาพที่ 3 วิธีการ SGA

หลักในการนำ GAs รวมถึง PSO มาประยุกต์ใช้ร่วมกันคือต้องใช้การเข้ารหัสแบบเดียวกัน วิธีการเข้ารหัสอ้างอิงมาจากงานวิจัย [5] ที่ทำการแทนโปรโตไทป์ให้อยู่ในรูปของไบนารีสตริง (Binary String) ที่มีความยาวเท่ากับจำนวนข้อมูลในชุดข้อมูลต้นแบบ โดยมีข้อกำหนดว่าหากเลือกข้อมูลตำแหน่งใดในชุดข้อมูลต้นแบบไปเป็นโปรโตไทป์ก็

ทำการกำหนดให้บิตในตำแหน่งนั้นมีค่าเป็น 1 ส่วนตำแหน่งอื่นๆที่ไม่ได้เลือกให้เป็นโปรโตไทป์ก็กำหนดให้มีค่าเป็น 0 ดังแสดงตัวอย่างในภาพที่ 4



ภาพที่ 4 การเข้ารหัสของปัญหาการเลือกโปรโตไทป์

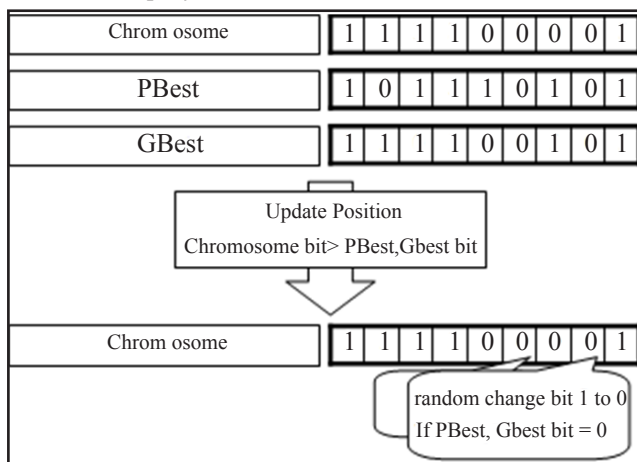
เนื่องด้วยวิธีการกลายพันธุ์และการเคลื่อนที่ของ GAs และ PSO มีลักษณะการทำงานไม่ต่างกัน ถ้ามองว่าการเปลี่ยนค่าในโครโมโซมของ GAs เท่ากับการเปลี่ยนตำแหน่งของ PSO สิ่งที่แตกต่างกันจะมีเพียงวิธีการคำนวณความเร็วที่นำมาใช้เปลี่ยนแปลงค่าเหล่านั้น ทำให้สิ่งสำคัญอีกอย่างในการนำวิธีการทั้งสองมาทำงานร่วมกันคือการกำหนด search space และวิธีการเคลื่อนที่ภายใน search space ให้กับประชากรของ GAs ซึ่งทั้งนี้ขึ้นอยู่กับลักษณะของโครโมโซมด้วย ในปัญหาการเลือกโปรโตไทป์ ถ้าให้ search space คือความเป็นไปได้ทั้งหมดในการเลือกโปรโตไทป์การที่โครโมโซมมีลักษณะเป็นไบนารีสตริง ทำให้ทิศทางการเคลื่อนที่สามารถเป็นไปได้อย่างสองค่าคือจาก 0 เป็น 1 และจาก 1 เป็น 0 แต่จะมีแกนมากเท่ากับจำนวนข้อมูลทั้งหมด

การทำงานของ SGA เริ่มต้นด้วยกระบวนการกำหนดค่าเริ่มต้นแบบเดียวกับ GAs ซึ่งเป็นวิธีการสุ่มกำหนดค่าให้กับประชากรเริ่มต้นเป็นลักษณะโครโมโซมดังที่ได้กล่าวไป จากนั้นจึงนำประชากรเริ่มต้นเหล่านั้นไปผ่านกระบวนการทำงานของ GAs ตามปกติทั้งการผสมพันธุ์และการกลายพันธุ์ตามลำดับ โดยใช้วิธีการผสมพันธุ์แบบ Uniform Crossover ที่จะทำให้การไล่ค่าที่ละบิตระหว่างโครโมโซม พ่อ-แม่ จากนั้นสุ่มสลับค่าบิตด้วยความน่าจะเป็น P_c การกลายพันธุ์จะใช้วิธีการแบบ Inverse-Bit Mutation ซึ่งเป็นการไล่ค่าที่ละบิตใน

โครโมโซมรุ่นลูก จากนั้นจึงสุ่มเปลี่ยนค่าบิตในโครโมโซมโดยเปลี่ยนค่าจาก 0 เป็น 1 หรือจาก 1 เป็น 0 ด้วยค่าความน่าจะเป็น P_m ตามที่ได้กำหนดไว้ล่วงหน้า

ภายในระบบผสม SGA ประชากร (หรืออนุภาค) ทั้งหมด จะทำการจดจำค่า PBest และความเร็วของตนเองไว้ในรอบการทำงานแรกค่า PBest ของประชากรเริ่มต้นคือค่าตำแหน่งของตัวมันเองส่วนประชากรที่เกิดจากการรวมกันของพันธุกรรมจะเปรียบเทียบค่าความเหมาะสมในตำแหน่งของตนเองกับ PBest ของพ่อแม่เพื่อเลือกตำแหน่งที่ดีที่สุดมาเป็น PBest ของตน ส่วนค่าความเร็วตั้งต้นของประชากรเริ่มต้นรวมทั้งประชากรที่เกิดจากการผสมพันธุกรรมจะมาจากการสุ่ม

การที่ความเร็วจะมีค่าเป็นบวกหรือลบขึ้นอยู่กับจำนวนบิต ณ ตำแหน่งปัจจุบันของประชากรเปรียบเทียบกับจำนวนบิตของตำแหน่ง PBest ของตนและ GBest แล้วนำไปบวกกับความเร็วเดิม โดยหากความเร็วมีค่าเป็นบวกก็จะทำการสุ่มกลับบิตที่ตำแหน่งปัจจุบันมีค่าเป็น 0 ให้เป็น 1 เหมือนกับค่าใน PBest หรือ GBest เช่นเดียวกันหากความเร็วมีค่าเป็นลบบิตที่ถูกสุ่มกลับค่าก็จะเป็นบิต 1 แทนดังภาพที่ 5



ภาพที่ 5 การเข้ารหัสของปัญหาการเลือกโปรโตไทป์

บางครั้งการเปลี่ยนแปลงตำแหน่งเพียงครั้งเดียวยังไม่สามารถทำให้ค่าความเหมาะสมของประชากรนั้นเปลี่ยนแปลงได้มากเพียงพอ จึงทำการวนซ้ำกระบวนการนี้เป็นจำนวน N ครั้งเพื่อให้สามารถค้นหาคำตอบบริเวณโดยรอบได้ดีมากยิ่งขึ้น หมายความว่าประชากรที่มีค่าความเหมาะสมต่ำเมื่อผ่านการทำงานของ PSO แล้วอาจมีค่าความเหมาะสมมากยิ่งขึ้นจนไม่ถูกคัดทิ้งก็เป็นได้

การวัดหาค่าความเหมาะสมของวิธีการ SGA ใช้ฟังก์ชันคำนวณค่าความเหมาะสมของประชากรเทียบกับเกณฑ์หรือ

คำตอบที่ต้องการ ซึ่งฟังก์ชันคำนวณความเหมาะสมที่ใช้จะแตกต่างกันไป ในงานวิจัยนี้จะอ้างอิงการใช้ฟังก์ชันคำนวณความเหมาะสมตามงานวิจัยที่ [6] ซึ่งสามารถแสดงเป็นสมการคณิตศาสตร์ดังนี้

$$F(S) = \frac{Accuracy(S) - \alpha \times Size(S)}{N_TrainingDataset} \quad (3)$$

$$\alpha = 1 / N_TraininigDataset \quad (4)$$

โดยที่ S คือโครโมโซมที่ถูกเข้ารหัสแทนโปรโตไทป์ $Accuracy(S)$ คือความถูกต้องในการจำแนกข้อมูลโดยใช้โปรโตไทป์ S $Size(S)$ คือจำนวนข้อมูลที่ถูกเลือกเป็นโปรโตไทป์ของ S $N_TrainingDataset$ คือจำนวนข้อมูลทั้งหมดของชุดข้อมูลต้นแบบ α คือค่าสัมประสิทธิ์ถ่วงน้ำหนักเพื่อถ่วงความสำคัญของพจน์ $Size(S)$ กับพจน์ $Accuracy(S)$ โดยในบทความนี้ได้กำหนดค่า α ดังสมการที่ 4

4. ผลการทดลอง

ขนาดของโปรโตไทป์จากวิธีการผสม SGA จะนำไปเปรียบเทียบกับ GAs โดยชุดข้อมูลที่ใช้ทำการทดสอบจะนำมาจากฐานข้อมูลมาตรฐานของ “UCI Machine Learning Repository” [6] ชุดข้อมูลจะถูกแบ่งออกเป็นชุดข้อมูลขนาดเล็กซึ่งประกอบด้วย IRIS, ECOLI และชุดข้อมูลขนาดใหญ่คือ CREDIT และ YEAST

ตารางที่ 1 รายละเอียดของชุดข้อมูล

ชุดข้อมูล	จำนวน	คุณลักษณะ	ชนิด
Iris	150	4	3
Ecoli	336	7	8
Credit	1000	24	2
Yeast	1484	8	10

การทดลองออกแบบให้วิธีการ SGA ทำการเลือกโปรโตไทป์จากเซตข้อมูลมาตรฐานในตารางที่ 1 แล้วทำการค้นหาโปรโตไทป์ที่มีค่าความเหมาะสมมากที่สุดตามวิธีการทำงานไปเรื่อยๆ

ตารางที่ 2 รายละเอียดค่าตัวแปรของวิธีการ GAs และ SGA

วิธีการ	N_{pop}	P_{ini}	P_m
GAs	40	0.8	0.01
SGA	20	0.8	0.01



เนื่องจากวิธีการของ GAs เป็นวิธีการแบบสุ่ม จึงทำการทดลองเป็นจำนวน 10 ครั้งแล้วนำมาหาค่าเฉลี่ย ทุกการทดลองใช้จำนวนรอบการทำงานทั้งหมด 10,000 รอบเพื่อทดสอบอัตราความเร็วของการเข้าสู่คำตอบด้วยรอบที่จำกัด โดยกำหนดค่าตัวแปรต่างๆดังตารางที่ 2 ซึ่ง N_{pop} คือจำนวนประชากรทั้งหมด P_{ini} คือความน่าจะเป็นในการสุ่มประชากรเริ่มต้นและ P_m คือความน่าจะเป็นของการกลายพันธุ์

ในการทดลองนี้จะเปรียบเทียบผลลัพธ์การทำงานของ เจเนติก อัลกอริทึมเดิมกับเจเนติกอัลกอริทึมที่นำเอาวิธีการ PSO มาช่วยในการแก้ปัญหา โดยใช้จำนวนผลลัพธ์ของชุดข้อมูลตัวอย่างที่ได้เทียบกับขนาดของข้อมูลตั้งต้น

ตารางที่ 3 รายละเอียดของชุดข้อมูล

Dataset	GAs		SGA	
	Average	Minimum	Average	Minimum
IRIS(150)	12.4	11	11.8	11
Ecoli (336)	102.4	97	96.0	89
Credit (1000)	560.8	542	466.6	458
Yeast (1484)	1451.5	1159	888.5	883

ในชุดข้อมูลขนาดเล็ก ผลการทดลองของ GAs และ SGA จะมีขนาดไม่แตกต่างกันมาก โดยเฉพาะชุดข้อมูล IRIS SGA จะให้ผลที่ดีกว่าภายใน 100 รอบการทำงานแรก หลังจากนั้นผลที่ได้จะเริ่มใกล้เคียงกัน เนื่องจากการทำงาน 10,000 รอบมากเกินไปจนความจำเป็นในการหาคำตอบของชุดข้อมูลนี้

ในชุดข้อมูลขนาดใหญ่ ผลลัพธ์ที่ได้จาก SGA จะแตกต่างกับ GAs อย่างชัดเจน โดยเฉพาะชุดข้อมูล YEAST 9 ใน 10 ของการทดลอง GAs ไม่สามารถหาคำตอบที่ดีกว่าชุดข้อมูลตั้งต้นได้ภายใน 10,000 รอบการทำงาน เนื่องจากขนาดข้อมูลที่ใหญ่รวมถึงความซับซ้อนของชนิดและคุณลักษณะ ทำให้ GAs ยากจะสร้างโปรโตไทป์ใหม่ที่ใช้แทนชุดข้อมูลตั้งต้นทั้ง 100% ทั้งนี้เพราะ GAs ไม่สามารถทดลองค้นหาคำตอบที่อยู่ใกล้เคียงแม้ว่าคำตอบนั้นจะมีบิตเพียงไม่กี่ตัวที่จำเป็นต้องทำการแก้ไขเพื่อให้ได้มาซึ่งคำตอบที่สามารถใช้แทนข้อมูลตั้งต้นทั้งหมดได้

5. สรุป

บทความนี้ได้นำเสนอระบบผสม SGA ที่นำเอา PSO มาเป็นส่วนช่วยในการทำงานของ GAs เราจะเห็นว่าหากให้ GAs ทำงานโดยไม่ใช้อัลกอริทึมอื่นช่วย ในปัญหาที่มีจำนวนข้อมูลน้อยคำตอบที่ได้จะใกล้เคียงกับ SGA และความแตกต่างจะมากขึ้นเมื่อจำนวนของข้อมูลมีขนาดเพิ่มขึ้น ทั้งนี้เนื่องจากชุดข้อมูลขนาดเล็กทั้ง GAs และ SGA สามารถหาคำตอบได้ด้วยจำนวนรอบการทำงานไม่มากนักแต่ในชุดข้อมูลขนาดใหญ่ GAs ถ้าเทียบกับ SGA ที่ใช้วิธีการค้นหาข้อมูลแบบพื้นที่ใกล้เคียงของ PSO มาช่วยเพิ่มประสิทธิภาพในการค้นหาคำตอบจะเห็นได้ว่า SGA จะมีอัตราการค้นหาคำตอบที่ดีในอัตราค่าเฉลี่ยที่ดีกว่า GAs

6. เอกสารอ้างอิง

- [1] Man K.F., K.S. Tang, and S. Kwong, "Genetic Algorithms: Concepts and Designs," Springer Verlag, 1999.
- [2] J.M. Smith, "Evolutionary Genetics", Second Edition, Oxford University Press Inc, 1998.
- [3] R. C. Eberhart, and J. Kennedy, "A new optimizer using particle swarm theory," *Proceedings of the Sixth International Symposium on Micromachine and Human Science*, Nagoya, Japan. pp. 39-43, 1995.
- [4] J. Kennedy, and R. C. Eberhart, "Particle swarm optimization," *Proceedings of IEEE International Conference on Neural Networks*, Piscataway, NJ. pp. 1942-1948, 1995.
- [5] L.I. Kuncheva and J.C. Bezdek, "Nearest Prototype Classification: Clustering, Genetic Algorithms, or Random Search?," *IEEE Transactions on Systems, Man and Cybernetics*, Vol. 28, No. 1, pp. 160-164, 1998.
- [6] Department of Information and Computer Science. University of California. "UCI Machine Learning Repository," [Online]. Available: <http://www.ics.uci.edu/~mllearn/MLRepository.html>. 1998.