

เจเนติกส์อัลกอริทึมกับปัญหาการวางแผนโรงงาน*

ปณิธาน พีรพัฒนา¹⁾

¹⁾อาจารย์ประจำ ภาควิชาวิศวกรรมอุตสาหการ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยขอนแก่น 40002

Email: panpee@kku.ac.th

บทคัดย่อ

บทความนี้นำเสนอการประยุกต์ใช้เจเนติกส์อัลกอริทึมกับปัญหาการวางแผนโรงงานซึ่งเป็นปัญหาแบบการหาค่าที่ดีที่สุด (Optimization Problem) โดยมีจุดประสงค์เพื่อให้ได้คำตอบที่ดีที่สุดซึ่งส่วนใหญ่จะเป็นการหาตำแหน่งที่ก่อให้เกิดค่าใช้จ่ายการขนถ่ายต่ำที่สุด เจเนติกส์อัลกอริทึมเป็นแนวคิดสมัยใหม่เป็นหลักการที่มีพื้นฐานมาจากการเปลี่ยนแปลงทางพันธุกรรมทางธรรมชาติ โดยการเลียนแบบการผสมยีนหรือโครโมโซมโดยที่สิ่งมีชีวิต (โครโมโซม) ที่มีความสมบูรณ์แข็งแรงเท่านั้นที่จะอยู่รอดได้ในธรรมชาติผสมผสานกับปัญหาการหาค่าที่ดีที่สุดมีเป้าหมาย คือหาสายพันธุ์หรือโครโมโซมที่ดีที่สุดซึ่งก็คือรูปแบบของคำตอบที่ก่อให้เกิดค่าของฟังก์ชันวัตถุประสงค์ที่ดีที่สุดโดยวิธีการปรับปรุงสายพันธุ์หรือคำตอบที่ดีขึ้นเรื่อยๆ โดยอาศัยกระบวนการพื้นฐานเช่นเข้ารหัส (Encoding) การประเมินค่า (Evaluation) การคัดเลือก (Selection) การข้ามสายพันธุ์ (Crossover) การผ่าเหล่า (Mutation) สามารถนำไปประยุกต์ใช้ได้อย่างมีประสิทธิภาพสูงกับหลากหลายสภาพปัญหาการวางแผนโรงงานทั้งแบบสถิตและพลวัต

คำสำคัญ : เจเนติกส์อัลกอริทึม การวางแผนโรงงาน ปัญหาการหาค่าที่ดีที่สุด

* รับผิดชอบเมื่อวันที่ 24 กรกฎาคม 2548 และได้รับบทความฉบับแก้ไขเมื่อวันที่ 26 กรกฎาคม 2549

Genetic Algorithms for solving The Facility Layout Problem*

Panitarn Peerapattana¹⁾

¹⁾Lecturer, Department of Industrial Engineering, Faculty of Engineering, Khon Kaen University 40002

Email: panpee@kku.ac.th

ABSTRACT

This paper introduces genetic algorithms for solving a facility layout problem which is an optimization problem. The objective generally for facility layout problem is to minimize the material flow cost. The concept of genetic algorithms is based on the process of natural selection in a biological system. The basic features of genetic algorithm are encoding, evaluation, selection, crossover, and mutation. Genetic algorithms can be efficiently applied to facility layout problems in both static and dynamic styles.

Keywords : genetic algorithms, facility layout problems, optimization problems

* Original manuscript submitted: July 24, 2006 and Final manuscript received: July 26, 2006

บทนำ

ในโลกปัจจุบันนี้มีวิธีการหาค่าหลายวิธีที่ใช้สำหรับปัญหาการหาค่าที่ดีที่สุด (Optimization Problems) นักวิจัยไม่ว่าจะในสาขาชีววิทยา เศรษฐศาสตร์ เกษตรศาสตร์ วิทยาศาสตร์ โดยเฉพาะอย่างยิ่งสาขา วิศวกรรมศาสตร์ต้องการเครื่องมือที่มีประสิทธิภาพสูงสำหรับปัญหาการหาค่าที่ดีที่สุด เนื่องจากความยุ่งยาก ของการกำหนดรูปแบบของระบบปัญหาในชีวิตจริงที่ค่อนข้างซับซ้อนไม่ว่าจะเป็น ปัญหาการหาค่าที่ดีที่สุดที่มีวัตถุประสงค์เดียวหรือหลายวัตถุประสงค์ในปัญหาเดียว (แต่ส่วนใหญ่จะนิยมใช้กับปัญหาที่มีวัตถุประสงค์เดียวเพราะง่ายต่อการแก้ปัญหามากกว่า) โดยที่ความซับซ้อนของ ปัญหาจะเพิ่มมากขึ้นไม่เฉพาะกับจำนวน ของวัตถุประสงค์ที่เพิ่มขึ้นเท่านั้น แต่ยังเพิ่มขึ้นเนื่องจากการมี ผลที่ตรงข้ามของวัตถุประสงค์เหล่านั้นอีกซึ่ง ทำให้จำนวนของคำตอบที่เป็นไปได้ของปัญหาการหาค่าที่ดีที่สุดนี้มีมากมายแต่จะมีคำตอบเพียงจำนวนน้อยเท่านั้นที่เป็นที่ต้องการ

เพื่อที่จะแก้ปัญหาการหาค่าที่ดีที่สุดได้อย่างเหมาะสมจึงมีการคิดค้นวิธีการที่มีประสิทธิภาพสูงที่ชื่อเจเนติกส์อัลกอริทึม (Holland, 1970) เจเนติกส์อัลกอริทึมเป็นแนวคิดสมัยใหม่สำหรับปัญหาการหาค่าที่ดีที่สุด เป็นหลักการที่มีพื้นฐานมาจากการเปลี่ยนแปลงทางพันธุกรรมทางธรรมชาติ (เลียนแบบการ ผสมยีนหรือโครโมโซมโดยที่สิ่งมีชีวิต (โครโมโซม) ที่มีความสมบูรณ์แข็งแรงเท่านั้นที่จะอยู่รอดได้ใน ธรรมชาติ) ผสมผสาน กับปัญหาการหาค่าที่ดีที่สุดมีเป้าหมายคือหาสายพันธุ์ (โครโมโซม) ที่ดีที่สุด ซึ่งก็คือรูปแบบของคำตอบ (โครโมโซม) ที่ก่อให้เกิดค่าของวัตถุประสงค์ที่ดีที่สุด โดยวิธีการปรับปรุงสายพันธุ์หรือคำตอบที่ดีขึ้นเรื่อยๆ ซึ่งสามารถนำไปประยุกต์ใช้ได้กับหลากหลายสภาพปัญหา

เจเนติกส์อัลกอริทึมเป็นวิธีการสำหรับแก้ปัญหาการหาค่าที่ดีที่สุดที่มีประสิทธิภาพสูง (Goldberg, 1989) ทั้งที่เป็นปัญหาการหาค่าที่ดีที่สุดแบบวัตถุประสงค์เดียวหรือหลายวัตถุประสงค์ (Single or Multi Objective Optimization Problems) (Murata, 1997) ในปัญหาเดียว

บทความนี้นำเสนอการประยุกต์ใช้เจเนติกส์อัลกอริทึมกับปัญหาการวางผังโรงงานซึ่งเป็น ปัญหา แบบการหาค่าที่ดีที่สุดโดยมีจุดประสงค์เพื่อให้ได้คำตอบที่ดีที่สุด (ส่วนใหญ่จะเป็นการหา ตำแหน่งของ หน่วยงานที่ก่อให้เกิดค่าใช้จ่ายการขนถ่ายวัสดุต่ำที่สุด) โดยมีการนำมาใช้กับปัญหา ทางด้านการวางผังโรงงานทั้งในรูปแบบสถิต (Static Layout Problems) และแบบพลวัต (Dynamic Layout Problems) อย่างมากมาย เช่น Tam (1992), Chan (1994), Conway (1994), Cheng (1995), Tate (1994), Suresh (1995), Jonies (1996), Hamamoto (1999), Peerapattana (2002), Peerapattana (2003), Peerapattana (2003), BalaKrishnan (2003), Peerapattana (2004), Adel El-Baz (2004), Perez (2004), และ Wang (2005) เจเนติกส์อัลกอริทึมเป็นวิธีการแบบฮิวริสติก (Heuristic) ที่ไม่สามารถ แก้ปัญหาการวางผังโรงงานได้โดยตรงแต่จะต้องนำมาใช้ร่วมกับวิธีการแบบฮิวริสติกพื้นฐานสำหรับวาง ผังโรงงานเช่นวิธีการปรับปรุงผัง (Improvement Algorithms) และวิธีการสร้างผัง (Construction Algorithms)

รูปแบบและแนวคิดพื้นฐานของเจเนติกส์อัลกอริทึมกับปัญหาการหาค่าที่ดีที่สุด

การประยุกต์ใช้เจเนติกส์อัลกอริทึมสำหรับแก้ปัญหาการหาค่าที่ดีที่สุดนั้น แต่ละค่าคำตอบของปัญหาที่ต้องการหาจะต้องถูกทำให้เป็นรหัส (Encoding) ซึ่งเรียกว่าการเข้ารหัส (Coding) ในรูปแบบการเรียงของตัวอักษร (String) ที่มีความยาวจำกัด อาจอยู่ในรูปของการเรียงสลับของตัวเลขทางพีชคณิต (Permutation Coding) เช่น 6 2 1 4 3 5 โดยที่ตัวเลขแต่ละตัวอาจจะแสดงเลขที่ของงานหน่วยงานของปัญหาหรือรหัส แบบตัวเลขฐานสอง (Binary Coding) เช่น 1 0 1 0 0 1 ซึ่งขึ้นอยู่กับลักษณะและความเหมาะสมของปัญหา

ส่วนประกอบที่สำคัญอื่นๆของเจเนติกส์อัลกอริทึมได้แก่โอเปอเรเตอร์พื้นฐานต่างๆเช่น การคัดเลือก (Selection or Reproduction) การข้ามสายพันธุ์ (Crossover) การผ่าเหล่า (Mutation) การเลือกโครโมโซม ที่โดดเด่นเพื่อรักษาไว้ในรุ่นถัดไป (Elitist Strategy) และพารามิเตอร์ต่างๆเช่นจำนวนประชากร (Population Size) ความน่าจะเป็นของการข้ามสายพันธุ์ การผ่าเหล่าหรือร้อยละของการเลือกโครโมโซมที่โดดเด่นเพื่อ รักษาไว้ในรุ่นถัดไป ซึ่งจะต้องถูกกำหนดให้เหมาะสมกับลักษณะของแต่ละปัญหาโดยการทดลองทางตัวเลขหลายๆครั้ง นอกจากนี้การเข้ารหัสที่ต่างกันจะทำให้การใช้โอเปอเรเตอร์พื้นฐานและค่าพารามิเตอร์เหล่านี้ แตกต่างกันด้วย โดยที่โอเปอเรเตอร์พื้นฐานต่างๆสามารถอธิบายได้ดังนี้

การเข้ารหัส (Coding)

การประยุกต์ใช้เจเนติกส์อัลกอริทึมสำหรับแก้ปัญหาการหาค่าที่ดีที่สุดนั้นแต่ละค่าคำตอบของปัญหาที่ต้องการหาจะต้องถูกทำให้เป็นรหัส (Encoded) ในรูปแบบการเรียงของตัวอักษรที่มีความยาวจำกัด (Strings) ซึ่งสามารถแบ่งออกเป็น 2 ประเภทคืออยู่ในรูปของรหัสการเรียงสลับของตัวเลขทางพีชคณิต (Permutation Coding) เช่น 6 2 1 4 3 5 และรหัสแบบตัวเลขฐานสอง (Binary Coding) เช่น 1 0 1 0 0 1 รหัสแบบตัวเลขฐานสองนิยมใช้กับปัญหาฟังก์ชันการหาค่าที่ดีที่สุดโดยการถอดรหัส (Decoding) จะออกมาอยู่ในรูปค่า จำนวนเต็ม (Integer) ส่วนรหัสการเรียงสลับของตัวเลขทางพีชคณิต มักจะใช้สำหรับปัญหาที่มีลักษณะของลำดับเข้ามาเกี่ยวข้องโดยใช้เลขจำนวนเต็ม 1 ถึง n เช่น ปัญหาการกำหนดการ (Scheduling) ปัญหาการวางแผนโรงงานหรือปัญหาการเดินทางของเซลแมน (Travelling Salesman Problems) โดยที่ตัวเลขแต่ละตัวจะ แสดงเลขที่ของงาน หน่วยงานหรือเมืองซึ่งขึ้นอยู่กับลักษณะและความเหมาะสมของปัญหา

การประเมินค่าความเหมาะสม (Evaluation)

แต่ละค่าคำตอบของปัญหาที่ผ่านการถอดรหัสจากสตริงในกระบวนการเจเนติกส์อัลกอริทึมจะถูกประเมินค่าความเหมาะสมสำหรับปัญหาแบบการหาค่าที่ดีที่สุด กระบวนการเจเนติกส์อัลกอริทึมจะหาสตริงที่มีค่า ความเหมาะสมที่ดีกว่า (ค่าที่ตอบสนองต่อฟังก์ชันวัตถุประสงค์ที่ดีกว่า) ตัวอย่างเช่นในปัญหาการวางแผนโรงงานที่มีฟังก์ชันวัตถุประสงค์คือการหาตำแหน่งของหน่วยงานที่ก่อให้เกิดค่าใช้จ่าย

การขนถ่ายวัสดุที่ต่ำที่สุดดังสมการที่ (1) โดยค่าความเหมาะสมมักจะเป็นค่าที่ตรงข้ามกับค่าของฟังก์ชันวัตถุประสงค์ดังสมการที่ (2)

$$\min z = \sum_{i=1}^m \sum_{j=1}^m f_{ij} c_{ij} d_{ij} , \tag{1}$$

f_{ij} = ปริมาณการขนถ่ายระหว่างหน่วยงาน i และ j

d_{ij} = ระยะทางระหว่างหน่วยงาน i และ j

c_{ij} = ค่าใช้จ่ายการขนถ่ายของ 1 หน่วยต่อ 1 หน่วยระยะทางระหว่างหน่วยงาน i และ j

$$f(x_i) = \frac{1}{z} \tag{2}$$

$f(x_i)$ = ค่าของฟังก์ชันความเหมาะสม (Fitness Value)

การคัดเลือก (Selection or Reproduction)

เป็นกระบวนการคัดเลือกสตริงพ่อแม่พันธุ์ (Parent Strings) สำหรับใช้ในกระบวนการสร้างสตริงรุ่น ใหม่ (Offspring) สตริงที่มีค่าความเหมาะสมที่สูงกว่าจะมีโอกาสที่จะถูกเลือกเป็นสตริงพ่อแม่พันธุ์มากกว่า สตริงที่มีค่าความเหมาะสมที่ต่ำกว่าโดยอาศัยวิธีการสุ่มคัดเลือกภายใต้หลักความน่าจะเป็นที่สิ่งที่ดีกว่าจะมี โอกาสถูกเลือกมากกว่า โดยทั่วไปมีหลายวิธีที่ใช้ในการคัดเลือกเช่น วิธีการจัดอันดับ (Ranking Based Method) และวิธีวงล้อรูเล็ตต์ (Roulette Wheel) เป็นต้น

$$P_s(x_i) = \frac{f(x_i)}{\sum_{j=1}^{N_{pop}} f(x_j)} , \text{ for } i = 1, 2, \dots, N_{pop} , \tag{3}$$

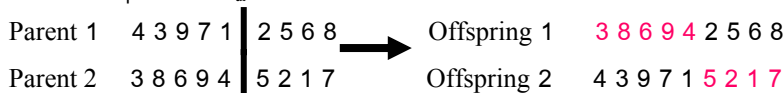
$P_s(x_i)$ ค่าความน่าจะเป็นในการเลือก

$f(x_i)$ ค่าของฟังก์ชันความเหมาะสม N_{pop} จำนวนประชากร

การข้ามสายพันธุ์ (Crossover)

เป็นกระบวนการสร้างสตริงลูกหลาน (Offspring) จากพ่อแม่พันธุ์ที่ถูกเลือกภายใต้ความน่าจะเป็น หรือเป็นการปรับปรุงสายพันธุ์โดยเป็นโอเปอเรเตอร์ที่ใช้ในการถ่ายถอดส่วนประกอบ (Elements) ของสตริง พ่อ-แม่พันธุ์ไปสู่สตริงรุ่นต่อไป โดยเริ่มจากการสุ่มเลือกจุดตัด (Cut Point: █) ซึ่งอาจจะใช้ 1 จุดหรือมากกว่า ก็ได้ขึ้นอยู่กับแต่ละปัญหาเพื่อทำการแลกเปลี่ยนส่วนประกอบกันมีหลายวิธีเช่น

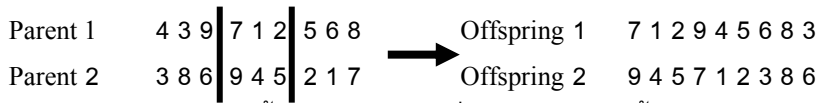
การข้ามสายพันธุ์แบบมาตรฐาน (Standard Crossover)



การข้ามสายพันธุ์แบบนี้จะเป็นการแลกเปลี่ยนส่วนประกอบทั้งหมด (ก่อนหรือหลังจุดตัด)

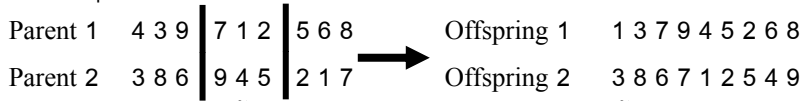
ระหว่าง พ่อ-แม่พันธุ์ โดยจะใช้เมื่อการซ้ำกันของส่วนประกอบในสตริงไม่มีผลต่อคำตอบ

การข้ามสายพันธุ์แบบลำดับ (Order Crossover)



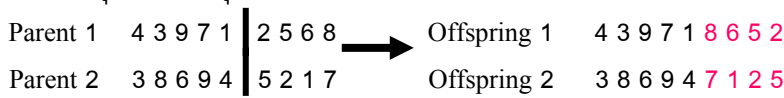
การข้ามสายพันธุ์แบบนี้จะเป็นการแลกเปลี่ยนส่วนประกอบทั้งหมด (ระหว่างจุดตัดทั้งสอง) ของพ่อ-แม่พันธุ์แล้วทำการจัดเรียงส่วนประกอบ โดยเริ่มจากส่วนประกอบระหว่างจุดตัดทั้งสองตามด้วยส่วนที่เหลือ ตามลำดับ เป็นวิธีการที่คำนึงถึงลำดับการเรียงของส่วนประกอบของพ่อ-แม่พันธุ์ในการสร้างสตริงรุ่นลูก แต่การซ้ำกันของส่วนประกอบในสตริงเดียวกันจะทำให้ได้คำตอบที่เป็นไปไม่ได้ (Infeasible Solution)

การข้ามสายพันธุ์แบบการจับกันบางส่วน (Partially Matched Crossover)



การข้ามสายพันธุ์แบบนี้จะเป็นการแลกเปลี่ยนส่วนประกอบทั้งหมด (ระหว่างจุดตัดทั้งสอง) ของพ่อ-แม่พันธุ์แล้วทำการพิจารณาส่วนประกอบที่จะเรียงเริ่มจากส่วนประกอบด้านซ้ายสุดโดยการตรวจสอบว่าซ้ำ กับส่วนประกอบระหว่างจุดตัดทั้งสองหรือไม่ ถ้าไม่ซ้ำก็คงไว้ แต่ถ้าซ้ำก็พิจารณาเลือกส่วนประกอบที่อยู่ใน ตำแหน่งตรงข้ามกับส่วนประกอบดังกล่าว เป็นวิธีการที่ไม่คำนึงถึงลำดับของส่วนประกอบของพ่อ-แม่พันธุ์ ในการสร้างสตริงรุ่นลูก ส่วนการซ้ำกันของส่วนประกอบจะทำให้ได้คำตอบที่เป็นไปไม่ได้

การข้ามสายพันธุ์แบบลำดับจุดเดียว (One Point Order Crossover)



การข้ามสายพันธุ์แบบนี้จะเป็นการแลกเปลี่ยนส่วนประกอบทั้งหมดหลังจุดตัดโดยพิจารณาลำดับการ เรียงของสตริงตรงข้าม เป็นวิธีการที่คำนึงถึงลำดับของส่วนประกอบของพ่อ-แม่พันธุ์ ในการสร้างสตริงรุ่นลูก ส่วนการซ้ำกันของส่วนประกอบจะทำให้ได้คำตอบที่เป็นไปไม่ได้การข้ามสายพันธุ์แบบลำดับจุดเดียวจะเห็นว่ากระบวนการข้ามสายพันธุ์จะช่วยให้การหาคำตอบไม่ตกเข้าไปในบริเวณที่เป็นกับดักของค่า คำตอบในลักษณะที่เป็น Locally Optimal แต่จะเป็นกระบวนการหาคำตอบในลักษณะกระจายตัวไปทั่วทุก บริเวณของพื้นที่คำตอบ ทำให้สามารถเข้าถึงคำตอบที่ใกล้เคียงคำตอบที่ดีที่สุด (Near Optimal Solutions) ได้

การผ่าเหล่า (Mutation)

เป็นโอเปอเรเตอร์ที่เปลี่ยนส่วนประกอบ (Elements) ของสตริงหลังจากกระบวนการข้ามสายพันธุ์ เป็นการค้นหาคำตอบในบริเวณใกล้เคียง (Neighborhood Search) ทำให้เกิดความหลากหลายของคำตอบ โดยที่กลไกของกระบวนการจะมีความแตกต่างเมื่อใช้กับการเข้ารหัสที่ต่างกันเช่นการผ่าเหล่าเมื่อใช้กับรหัส การเรียงสลับของตัวเลขทางพีชคณิตจะสามารถทำได้โดยการสุ่มเลือกสองส่วนประกอบของสตริงเพื่อทำการสลับตำแหน่งซึ่งกันและกัน (Arbitrary Two-Element Change) เช่นถ้าตำแหน่งที่เกิดการผ่าเหล่าคือตำแหน่งที่ 2 และ 6



ส่วนการผ่าเหล่าเมื่อใช้รหัสแบบตัวเลขฐานสองจะสามารถทำได้โดยการสุ่มเลือกส่วนประกอบของสตริงเพื่อทำการเปลี่ยนค่าส่วนประกอบนั้นเป็นค่าที่ตรงกันข้ามเช่นถ้าตำแหน่งที่เกิดการผ่าเหล่าคือตำแหน่งที่ 2 และ 6

1 0 1 0 0 1 0 \longrightarrow 1 1 1 0 0 0 0

การเลือกโครโมโซมที่โดดเด่นเพื่อรักษาไว้ในรุ่นถัดไป (Elitist Strategy)

เป็นวิธีการที่รักษาสตริงที่ดีที่สุดประชากรรุ่นหนึ่งเพื่อเก็บไว้สำหรับรุ่นต่อไปโดยไม่มีผลกระทบจากกระบวนการข้ามสายพันธุ์และการผ่าเหล่า สตริงที่ดีที่สุดเหล่านี้จะมีโอกาสสูงในการถูกเลือกเป็นพ่อแม่พันธุ์ ในรุ่นต่อไป ถือเป็นวิธีการเสริมที่ช่วยในการเข้าถึงคำตอบได้เร็วขึ้นซึ่งอาจไม่จำเป็นต้องใช้สำหรับทุกปัญหา โดยที่ร้อยละของการเลือกโครโมโซมที่โดดเด่นเพื่อรักษาไว้ในรุ่นถัดไปนั้นก็ขึ้นอยู่กับลักษณะและความเหมาะสมของแต่ละปัญหา

ลำดับขั้นตอนการประยุกต์ใช้เจเนติกส์อัลกอริทึม

- ขั้นตอน 1 การสร้างประชากรเริ่มต้น (Initialization)
- ขั้นตอน 2 การประเมินค่าความเหมาะสม
- ขั้นตอน 3 การคัดเลือก
- ขั้นตอน 4 การข้ามสายพันธุ์
- ขั้นตอน 5 การผ่าเหล่า
- ขั้นตอน 6 การเลือกโครโมโซมที่โดดเด่นเพื่อรักษาไว้ในรุ่นถัดไป
- ขั้นตอน 7 การทดสอบเงื่อนไขการหยุด (Termination Test)
- ขั้นตอน 8 การได้คำตอบที่ดีที่สุด

แนวทางการประยุกต์ใช้เจเนติกส์อัลกอริทึมกับปัญหาการวางผังโรงงาน

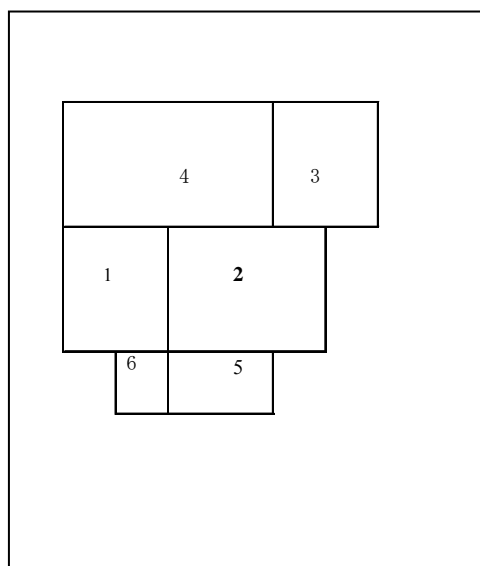
ปัญหาการวางผังโรงงานเป็นปัญหาแบบการหาค่าที่ดีที่สุดที่เรียกว่า NP Hard Problems ส่วนใหญ่มี จุดประสงค์เพื่อหาค่าตำแหน่งของ หน่วยงานที่ทำให้เกิดค่าใช้จ่ายการขนถ่ายวัสดุต่ำที่สุดซึ่งทำให้จำนวนของ คำตอบที่เป็นไปได้ของปัญหาการหาค่าที่ดีที่สุดนี้มีมากมาย แต่จะมีคำตอบเพียงจำนวนน้อยเท่านั้นที่เป็นที่ ต้องการ ยิ่งถ้าขนาดของปัญหาใหญ่ขึ้น (จำนวนของหน่วยงานที่ต้องการหาค่าตำแหน่งมีมากขึ้น) หรือเป็นปัญหา การวางผังที่หน่วยงานมีขนาดไม่เท่ากัน (Unequal Area) ในพื้นที่การวางผังแบบจำกัด (Fixed Layout Area) และแบบไม่จำกัด (Open Space) ทั้งใน กรณีปัญหาการวางผังแบบสถิต (Static Layout Problems) หรือโดยเฉพาะอย่างยิ่งในปัญหาการวางผังแบบพลวัต (Dynamic Layout Problems) ก็ยังจะทำให้การเข้า ถึงคำตอบที่ดีที่สุด (Optimal Solution) เป็นไปได้ยาก

เพื่อที่จะแก้ปัญหาการวางผังโรงงานได้อย่างเหมาะสมจึงมีการนำวิธีการที่มีประสิทธิภาพสูงที่ชื่อ เจเนติกส์อัลกอริทึมมาใช้ผสมผสานกับปัญหาการหาค่าที่ดีที่สุด (Mavridou et al., 1997) มีเป้าหมายคือ

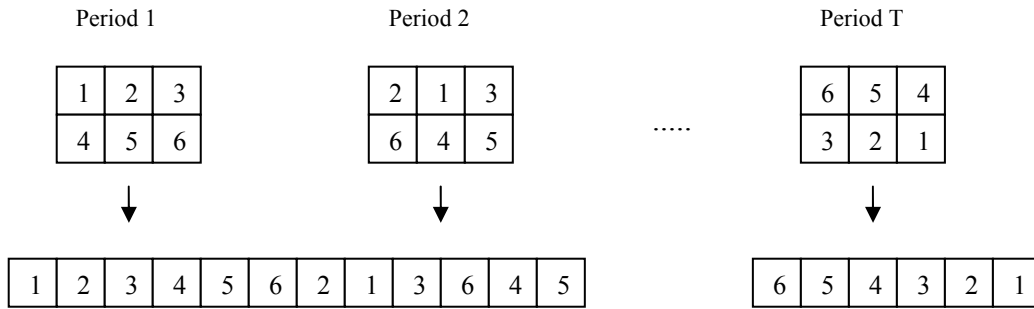
หารูป แบบของคำตอบที่ก่อให้เกิดค่าของวัตถุประสงค์ที่ดีที่สุด โดยวิธีการปรับปรุงคำตอบที่ดีขึ้นเรื่อยๆ ซึ่งสามารถ นำไปประยุกต์ใช้ ได้กับหลากหลายสภาพปัญหาทั้งที่เป็นปัญหาการวางแผนโรงงานแบบมี วัตถุประสงค์เดียว และหลาย วัตถุประสงค์ในปัญหาเดียวหรือทั้งในกรณีปัญหาการวางแผนโรงงานแบบ สถิตและแบบพลวัต

เจเนติกส์อัลกอริทึมเป็นวิธีการแบบฮิวริสติกที่ไม่สามารถแก้ปัญหาการวางแผนโรงงานได้ โดยตรง แต่จะต้องนำมาใช้ร่วมกับวิธีการแบบฮิวริสติกพื้นฐานของการวางแผนโรงงานสำหรับการ ถอดรหัสเช่นวิธี การปรับปรุงผัง (Improvement Algorithms) เช่นการสลับตำแหน่งหน่วยงาน (Pairwise Exchange Method) การปรับปรุงผังแบบ CRAFT และวิธีการสร้างผัง (Construction Algorithms) เช่น วิธีการสร้างไดอะแกรมความสัมพันธ์ (Relationship Diagramming Method), วิธีการสร้างผังแบบ CORELAP, วิธีการสร้างผังแบบการเติบโต (Growth Method) วิธีการสร้างผังแบบกราฟ (Graph-Based Method)

ปัญหาการวางแผนโรงงานที่ใช้เจเนติกส์อัลกอริทึมส่วนใหญ่จะนิยมเข้ารหัสแบบการเรียงสลับของ ตัวเลขทางพีชคณิตเพราะสะดวกในการถอดรหัสและเข้าใจง่ายมากกว่ารหัสแบบตัวเลขฐานสอง เช่น งานวิจัยของ Hamamoto (1999), Peerapattana (2002), Peerapattana (2003), Peerapattana (2003), Balakrishnan (2003), Peerapattana (2004) และ Wang (2005) จะใช้เลขจำนวน เต็ม 1 ถึง n กับปัญหา การวางแผนโรง งาน โดยที่ตัวเลขแต่ละตัวจะแสดงเลขที่ของหน่วยงานที่ต้องการหาตำแหน่งที่เหมาะสม ที่สุด เช่น 2 1 5 6 4 3 หน่วยงานที่ 2 จะถูกจัดเรียงแรกสุดตามด้วยหน่วยงาน 1 5 6 4 และ 3 ตามลำดับ ดังรูปที่ 1 ส่วนรูป ที่ 2 ก็คล้ายกันแต่เป็นปัญหาในแบบพลวัต



รูปที่ 1 ตัวอย่างการสร้างผังโดย Growth method ในปัญหาแบบสถิต (Hamamoto et al., 1999), โดยมีลำดับ การเรียงตามรหัส 2 1 5 6 4 3



รูปที่ 2 ตัวอย่างการสร้างผังแบบ QAP ในปัญหาแบบพลวัต (Balakrishnan et al., 2003) โดยมีลำดับการเรียงตามรหัส 1 2 3 4 5 6 2 1 3 6 4 5 6 5 4 3 2 1

กระบวนการเจเนติกส์อัลกอริทึมเริ่มต้นจากการสร้างประชากรเริ่มต้น (Initial Solutions) ในรูปของการเข้ารหัสโดยอาจจะใช้วิธีการสุ่มหรือการเลือกจากกลุ่มของคำตอบที่ดี แต่ละคำตอบของปัญหาที่ผ่านการถอดรหัสจากสตริงจะถูกประเมินค่าความเหมาะสมสำหรับปัญหาการวางผังโรงงาน กระบวนการเจเนติกส์อัลกอริทึมจะหาสตริงที่มีค่าความเหมาะสมที่ดีกว่า (ค่าที่ตอบสนองต่อฟังก์ชันวัตถุประสงค์ที่ดีกว่าซึ่ง ก็คือตำแหน่งของหน่วยงานที่ก่อให้เกิดค่าใช้จ่ายการขนถ่ายวัสดุที่ต่ำกว่า) โดยค่าความเหมาะสมมักจะเป็น ค่าที่ตรงข้ามกับค่าของฟังก์ชันวัตถุประสงค์ดั้งสมการที่ (1) และ (2)

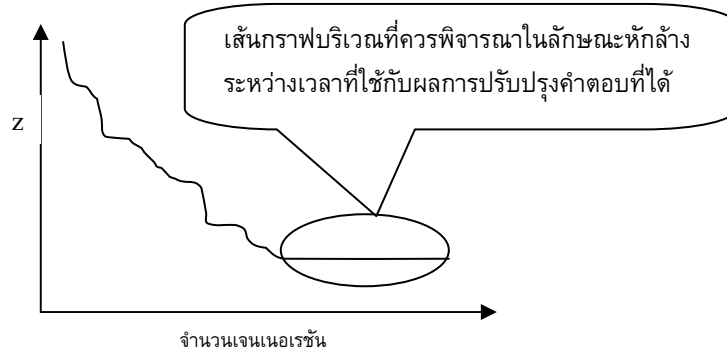
ต่อไปเป็นกระบวนการคัดเลือกสตริงพ่อแม่พันธุ์ สำหรับใช้ในกระบวนการสร้างสตริงรุ่นต่อไป สตริงที่มีค่าความเหมาะสมที่สูงกว่าจะมีโอกาสที่จะถูกเลือกเป็นสตริงพ่อแม่พันธุ์มากกว่า โดยอาศัยวิธีการสุ่มคัด เลือกภายใต้หลักความน่าจะเป็นที่สิ่งที่ดีกว่าจะมีโอกาสถูกเลือกมากกว่าดังสมการที่ (3) สำหรับปัญหาการ วางผังโรงงานนิยมใช้วิธีวงล้อรูเล็ตต์

ส่วนวิธีปรับปรุงคำตอบโดยการข้ามสายพันธุ์การข้ามสายพันธุ์แบบลำดับมากกว่าแบบอื่น เพราะรหัสจะเปลี่ยนแปลงน้อยกว่าแบบอื่นโดยเป็นการรักษาสภาพเดิมไว้ได้มากที่สุด ถ้าเป็นปัญหาที่ต้องการทำให้รหัสเปลี่ยนแปลงมากกว่าก็ควรใช้การข้ามสายพันธุ์แบบการจับกันบางส่วน แต่ถ้าเป็นปัญหาที่ไม่ ซับซ้อนมากนักวิธีการข้ามสายพันธุ์แบบธรรมดาหรือแบบมาตรฐานก็เหมาะสม

หลังจากกระบวนการข้ามสายพันธุ์ การผ่าเหล่าจะเกิดภายใต้ความน่าจะเป็นที่ไม่สูงนักเพื่อเป็นการ ค้นหาคำตอบในบริเวณใกล้เคียงและเป็นการสร้างความหลากหลายให้กับคำตอบ

การหาค่าพารามิเตอร์ต่างๆของเจเนติกส์อัลกอริทึมซึ่งได้แก่จำนวนประชากร (Population Size) จำนวนเจเนเนอเรชัน (Generation Number) ความน่าจะเป็นของการข้ามสายพันธุ์ (Crossover Probability) การผ่าเหล่า (Mutation Probability) ร้อยละของการเลือกโครโมโซมที่โดดเด่นเพื่อรักษาไว้ในรุ่นถัดไป (Elitist Percentage) จะได้จากการทดลองทางตัวเลขหลายๆครั้งเพื่อหาค่าพารามิเตอร์ที่เหมาะสมเช่นจำนวนประชากรก็อาจใช้ค่าตั้งแต่ 10 ถึง 50 จำนวนเจเนเนอเรชันก็อาจใช้ค่าตั้งแต่ 100 ถึง 5000 ความน่าจะเป็นของการข้ามสายพันธุ์ก็อาจใช้ค่าตั้งแต่ 10 ถึง 70% ความน่าจะเป็นของการผ่าเหล่าอาจ จะใช้ค่าตั้งแต่ 10 ถึง 50% ส่วนการทดสอบเงื่อนไขการหยุดนั้นจะต้องพิจารณาใน

ลักษณะหักล้าง (Tradeoff) ระหว่างเวลาที่ใช้กับผลการปรับปรุงคำตอบที่ดีขึ้นโดยอาจดูจากกราฟ บริเวณที่ทำเครื่องหมายระหว่างจำนวนเงินเนอเรนซ์กับค่าคำตอบที่ได้



รูปที่ 3 แสดงกราฟระหว่างจำนวนเงินเนอเรนซ์กับค่าคำตอบที่ได้

การวิเคราะห์

เจเนติกส์อัลกอริทึมเป็นวิธีการหาคำตอบที่มีประสิทธิภาพสูงสำหรับปัญหาการวางแผนโรงงานที่เหนือกว่าวิธีการฮิวริสติกในกลุ่มของ Global Optimization Heuristics ที่มีชื่อเสียงบางวิธีเช่น ซิมมูลเตด แอนนิลลิ่ง (Simulated Annealing) (Balakrishnan, 2003 และ Peerapattana, 2004) เหมาะกับปัญหาที่มีความซับซ้อนและมีจำนวนของคำตอบมากมายเช่นปัญหาการวางแผนโรงงานในกรณีที่หน่วยงานมีขนาดไม่เท่ากันในพื้นที่การวางแผนแบบจำกัดและแบบไม่จำกัดทั้งในกรณีปัญหาการวางแผนแบบสถิติหรือโดยเฉพาะอย่างยิ่งในปัญหาการวางแผนแบบพลวัตถึงแม้ว่าจะใช้เวลาในการคำนวณมากกว่าก็ตามเจเนติกส์อัลกอริทึม ช่วยให้สามารถเข้าถึงคำตอบที่ใกล้เคียงคำตอบที่ดีที่สุด (Near Optimum) ในเวลาที่เหมาะสม

เจเนติกส์อัลกอริทึมเป็นวิธีการหนึ่งของการหาคำตอบที่อาศัยการกระจายความน่าจะเป็น (Stochastic) ในลักษณะการค้นหาแบบขนาน (Parallel Search Methods) โดยที่แต่ละประชากร (คำตอบ) จะสร้างคำตอบที่เป็นไปได้หลากหลายในทิศทางค้นหา (Search Direction) เพื่อให้กระบวนการค้นหาคำตอบของเจเนติกส์อัลกอริทึมมีประสิทธิภาพสูงสุดการกำหนดค่าพารามิเตอร์ที่เหมาะสมของเจเนติกส์อัลกอริทึมซึ่งได้แก่จำนวนประชากรและจำนวนเงินเนอเรนซ์จะต้องไม่มากเกินไป เนื่องจากมีผลโดยตรงต่อเวลาที่ใช้ในการคำนวณ ส่วนความน่าจะเป็นของการข้ามสายพันธุ์ส่วนใหญ่จะอยู่ระหว่าง 0.2-0.7 เพราะถ้ามากเกินไปจะทำให้เข้าใกล้วิธีการสุ่ม (Randomized Search) การผ่าเหล่าส่วนใหญ่จะอยู่ระหว่าง 0.1-0.8 ส่วน ร้อยละของการเลือกโครโมโซมที่โดดเด่นเพื่อรักษาไว้ในรุ่นถัดไปไม่ควรจะมากเกินไปเพราะอาจจะทำให้มีโอกาสที่คำตอบจะตกอยู่ในบริเวณกับดักของคำตอบ (Locally Optimal Solutions) ได้ง่าย

สรุป

บทความนี้นำเสนอรูปแบบการประยุกต์ใช้เจเนติกส์อัลกอริทึมกับปัญหาการวางผังโรงงานซึ่งเป็นปัญหาแบบการหาค่าที่ดีที่สุด โดยเป็นวิธีการสำหรับแก้ปัญหาที่มีประสิทธิภาพสูงนิยมใช้กันอย่างแพร่หลายใน ปัญหาที่มีความซับซ้อนและมีจำนวนของคำตอบมาก เจเนติกส์อัลกอริทึมช่วยให้สามารถเข้าถึงคำตอบที่ใกล้เคียงคำตอบที่ดีที่สุด (Near Optimum) ในเวลาที่เหมาะสม

เอกสารอ้างอิง

- Balakrishnan, J., Cheng, C. H. and Wong, K. F. 2003. FACOPT: A user friendly facility layout optimization system. **Computers & Operations Research**. Vol.30, No.11: pp.1625-1641.
- Balakrishnan, J., Chun Hung Cheng, Daniel G. Conway, and Chun Ming Lau. 2003. A hybrid genetic algorithm for the dynamic plant layout problem. **Int. J. Production Economics**. Vol.86: pp. 107-120.
- Chan, K.C. and Tansri, H. 1994. A study of genetic crossover operations on the facilities layout problem. **Computers & Industrial Engineering**. Vol.26, No.3: pp.537-550.
- Cheng, R., Gen, M. and Tozawa, T. 1995. Genetic search for facility layout design under interflows uncertainty. **IEEE International Conference on Evolutionary Computation**. Vol.1: pp.400-405.
- Conway, D.G. and Venkataramanan, M.A. 1994. Genetic search and the dynamic facility layout problem. **Computers & Operations Research**. Vol.21, No.8: pp.955-960.
- Goldberg, D. 1989. **Genetic algorithms in search, optimization, and machine learning**. Addison-Wesley Reading MA.
- Hamamoto, S., Yih, Y., and Salvendy, G. 1999. Development and Validation of Genetic AlgorithmBased Facility Layout. **Int. J. Production. Research**. vol.37, no.4: pp.749-768.
- Hirabayashi, N., Peerapattana, P., and Nagasawa, N. 2000. A study on dynamic facility layout with alternative processing routes. **Proceedings of 2000 International Conference on Advanced Manufacturing Systems and Manufacturing Automation**. Guangzhou, China: pp. 678-690.
- Hirabayashi, N., Peerapattana, P., and Nagasawa, N. 2001. Dynamic layout in case of non-identical shapes and areas of facilities with alternative processing routes. **Proceedings of 16th International Conference on Production Research (ICPR-16)**. ISBN 80-02-01438-3, #0188, Prague, Czech Republic: pp. 1-10.
- Joines, J.A., Culbreth, C.T. and King, R.E. 1996. Manufacturing cell design: an integer programming model employing genetic algorithms. **IIE Transactions**. Vol.28, No.1: pp.69-85.
- Mavridou, T.D. and Pardalos, P.M. 1997. Simulated annealing and genetic algorithms for the facility layout problem: A survey. **Computational Optimization and Applications**. Vol.7, No.1: pp.111-126.
- Ming-Jaan Wang, Michael H. Hu and Meei-Yuh Ku. 2005. A solution to the unequal area facilities layout problem by genetic algorithm. **Computers in Industry**. Vol. 56: pp.207-220
- Murata, T. 1997. **Genetic algorithms for multi-objective optimization** [Doctoral Thesis at Osaka Prefecture University]. Osaka: Osaka Prefecture University.

- Peerapattana, P., Hirabayashi, N., and Nagasawa, N. 2002. Dynamic facility layout using genetic algorithm. **Proceedings of 3rd International ICSC-NAISO Congress on World Manufacturing (WMC-2001)**. Rochester, New York, USA. ISBN 3-906454-28-2, #1913-022: pp. 1-5.
- Peerapattana, P., Hirabayashi, N., and Nagasawa, N. 2003. Dynamic layout using genetic algorithm in case of different shapes and areas of facilities with alternative processing routes. **Proceedings of 17th International Conference on Production Research (ICPR-17)**. Blacksburg, Virginia, USA. ISBN 0-9721257-3-6, #0177: pp.1-7.
- Peerapattana, P., Hirabayashi, N., and Nagasawa, N. 2003. A multi-phase heuristic method in dynamic facility layout. **Transactions of the Institute of Systems, Control and Information Engineers in Japanese**. Vol. 17, No.4: pp. 178-186.
- Peerapattana, P. 2004. **Dynamic Facility Layout with Alternative Processing Routes** [Doctoral Thesis at Osaka Prefecture University]. Osaka: Osaka Prefecture University.
- Suresh, G., Vinod, V. V. and Sahu, S. 1995. A genetic algorithm for facility layout. **International Journal of Production Research**. Vol.33, No.12: pp.3411-3423.
- Tam, K.Y. 1992. Genetic algorithms, function optimization and facility layout design. **European Journal of Operational Research**. Vol.63, No.2: pp.322-346.
- Tate, D.M. and Smith, A.E. 1995. Unequal-area facility layout by genetic search. **IIE Transactions**. Vol.27, No.4: pp.465-472.
- Tompkins, White, Bozer, Frazelle, Tanchoco and Trevino. 1996. **Facilities planning**. second edition. John Willey & Sons Inc.
- Wu, Y. and Appleton, E. 2002. The optimization of block layout and aisle structure by a genetic algorithm. **Computers & Industrial Engineering**. Vol.41, No.4: pp.371-387.