

การแก้ปัญหาการบรรจุผลิตภัณฑ์ในสามมิติโดยขั้นตอนวิธีพันธุกรรมดัดแปลง

Solving 3-Dimensional Bin Packing by Modified Genetic Algorithms

จิระเดช พลสวัสดิ์¹ และ ประภาส จงสถิตย์วัฒนา²

Jiradej Ponsawat¹ and Prabhas Chongstitvatana²

Department of Computer Engineering

Chulalongkorn University, Bangkok 10330, Thailand

e-mail: jiradej.p@student.chula.ac.th¹, prabhas@chula.ac.th²

บทคัดย่อ

บทความนี้นำเสนอวิธีการบรรจุผลิตภัณฑ์ในสามมิติโดยใช้ขั้นตอนวิธีพันธุกรรมร่วมกับการค้นหาเฉพาะที่และผลเปรียบเทียบผลระหว่างวิธีที่นำเสนอกับวิธีขยายและกั้นเขตซึ่งมีผู้ทำวิจัยไว้ก่อนหน้านี้ จากผลการทดลองพบว่าความสามารถในการหาคำตอบของวิธีที่นำเสนอจะดีกว่าเมื่อปัญหามีขนาดใหญ่ ทั้งในด้านโอกาสที่หาคำตอบเจอและเวลาที่ใช้

Abstract

This work proposes an algorithm to solve a three-dimensional Bin Packing Problem using Genetic Algorithm with local search. Comparison between the result of the proposed algorithm and an earlier work that used Branch and Bound Algorithm is also reported. The result shows that the number of solved problems and the run time of the proposed algorithm is superior when the problem size is large.

Key-Words: Bin Packing, Genetic Algorithm, Local Search

1. บทนำ

ปัญหาการบรรจุผลิตภัณฑ์ (Bin Packing Problem) เป็นปัญหาที่มีความสำคัญในหลาย ๆ ด้าน ทั้งด้านวิชาการคอมพิวเตอร์ ด้านการวิจัย และในเชิงธุรกิจ จะเห็นว่ามีการนำไปใช้แก้ปัญหาหลายอย่าง เช่น การจัด

ตารางงาน การจัดแบ่งทรัพยากร การแยกส่วนวัสดุ การบรรจุสินค้าสำหรับขนส่ง การจัดวางตำแหน่งโต๊ะทำงาน การจัดวางไอซีในวงจร เป็นต้น เนื่องจากปัญหาการบรรจุผลิตภัณฑ์เป็นปัญหาที่ไม่สามารถหาคำตอบได้ในเวลาโพลิโนเมียล จัดอยู่ในกลุ่ม NP-HARD รวมทั้งปัญหาในระดับ 1 มิติและ 2 มิติด้วย โดยเฉพาะในปัญหาระดับ 3 มิติจะยิ่งมีความยากมากขึ้น งานวิจัย [3] แสดงให้เห็นว่าสามารถปรับขั้นตอนวิธีแก้ปัญหา 2 มิติเป็นปัญหา 3 มิติด้วยการดัดแปลงเพียงเล็กน้อย แต่ในงานวิจัย [2] ซึ่งทำการแก้ปัญหาจัดตารางงาน ระบุว่ากรณีวิเคราะห์และปรับวิธีการในหนึ่งมิติให้ใช้ได้หลายมิติจะทำได้ไม่ดี เพราะขาดความรู้เกี่ยวกับปัญหาในมิติอื่น ๆ

ในงานวิจัย [1] ได้เสนอวิธีการสำหรับบรรจุผลิตภัณฑ์ในสามมิติโดยใช้ขั้นตอนวิธีพันธุกรรมแบบ Co-operative Co-evolutionary แต่ว่าการจัดเรียงเป็นแบบตั้งฉากและตัดแบ่งด้วยเส้นตรงได้ (orthogonal and guillotineable) ผลลัพธ์ที่ได้ดีกว่าการใช้ขั้นตอนวิธีพันธุกรรมอย่างธรรมดา (Simple GA)

งานวิจัย [3] เสนอวิธีแบบศึกษาสำนึก (heuristic approach) คือการขยายและกั้นเขตสำหรับแก้ปัญหาการบรรจุผลิตภัณฑ์สามมิติ โดยนำเสนอขอบเขตล่างแบบใหม่สามารถใช้หาคำตอบได้ดีในปัญหาที่มีขนาดเล็ก แต่เมื่อปัญหามีจำนวนชิ้นที่จะบรรจุเพิ่มมากขึ้นจะใช้เวลาเพิ่มขึ้นอย่างมาก ดังนั้นบทความนี้จึงเสนอขั้นตอนวิธีพันธุกรรม

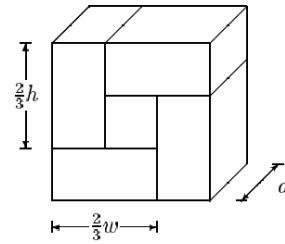
มาใช้แก้ปัญหาและศึกษาผลกระทบทางด้านเวลาที่ใช้ในการหาคำตอบ

2. ปัญหาและข้อมูลในการทดลอง

กำหนดให้มีกล่องสี่เหลี่ยม n กล่อง แต่ละกล่องมีขนาดกว้าง w_j สูง h_j และหนา d_j ($j \in J = \{1, \dots, n\}$) และมีคอนเทนเนอร์ไม่จำกัดจำนวนขนาดเท่ากันโดยมีความกว้าง W สูง H และหนา D ($w_j \leq W, h_j \leq H, d_j \leq D$) ปัญหาการบรรจุผลิตภัณฑ์ในสามมิติเป็นการบรรจุกล่องทั้งหมดเข้าไปในคอนเทนเนอร์โดยใช้คอนเทนเนอร์จำนวนน้อยที่สุด ในที่นี้จะต้องวางให้ขอบของคอนเทนเนอร์ขนานกับขอบกล่อง ไม่มีการหมุนของแต่ละกล่อง ข้อมูลที่ใช้ในการทดลองได้จากการสร้างโดยใช้ขั้นตอนวิธีในงานวิจัย [3] ข้อมูลที่ใช้ในการทดลองแบ่งเป็น 9 กลุ่ม แต่ละกลุ่มจะทำการทดลองที่จำนวนชิ้นตั้งแต่ 10 ชิ้นจนถึง 90 ชิ้น แต่ละขนาดของปัญหาจะมีข้อมูลแต่ละกลุ่ม 10 ชุด ขนาดของคอนเทนเนอร์ของทุกกลุ่มเป็นรูปทรงลูกบาศก์มีความกว้าง ความสูงและความหนาเท่ากันทุกด้านคือ 100 หน่วย ยกเว้นกลุ่มที่ 6 และ 7 มีขนาด 10 และ 40 หน่วย ตามลำดับ ปัญหาในกลุ่มที่ 1 2 และ 3 มีลักษณะคล้ายกันคือมีความยาวทางด้านหนึ่งน้อยกว่าครึ่งหนึ่งของคอนเทนเนอร์ อีกสองด้านที่เหลือมีความยาวมากกว่า 2 ใน 3 ของคอนเทนเนอร์ โดยด้านที่สั้นจะอยู่ทางด้านกว้าง สูง และหนา ของปัญหาที่ 1 2 และ 3 ตามลำดับ ปัญหาในกลุ่มที่ 4 เป็นปัญหาที่มีผลิตภัณฑ์มีขนาดใหญ่ กลุ่ม 5 ผลิตภัณฑ์มีขนาดเล็ก กลุ่ม 6 7 และ 8 ผลิตภัณฑ์มีทุก ๆ ขนาด ส่วนกลุ่มสุดท้ายเป็นปัญหาพิเศษคือมีคำตอบเท่ากับ 3 คอนเทนเนอร์ โดยจะได้จากการตัดแบ่งคอนเทนเนอร์ออกเป็นส่วนย่อย ๆ ซึ่งไม่สมมาตรมีลักษณะดังรูปที่ 1 แต่ละคอนเทนเนอร์ถูกแบ่งออกเป็นจำนวนชิ้นประมาณ $n/3$ ชิ้น เมื่อ n เป็นขนาดของปัญหา

ปัญหาในกลุ่มที่ 4 เป็นปัญหาที่ง่าย ปัญหาในกลุ่ม 1-3 เป็นปัญหาขากปานกลาง ปัญหาในกลุ่มที่ 5-8 เป็นปัญหาค่อนข้าง

ยากเพราะของมีขนาดคละ ปัญหาในกลุ่ม 9 เป็นปัญหาขากเพราะมีข้อจำกัดทางด้านขนาด



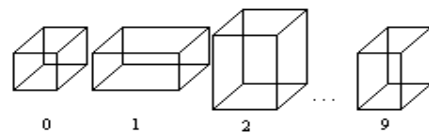
รูปที่ 1 การตัดแบ่งกล่องสำหรับข้อมูลปัญหาในกลุ่มที่ 9

3. ขั้นตอนวิธีพันธุกรรม

ขั้นตอนวิธีพันธุกรรม [4] เป็นวิธีการค้นหาผลเฉลยโดยการเลียนแบบวิวัฒนาการตามธรรมชาติ โดยจะแทนคำตอบของปัญหาให้อยู่ในรูปของโครโมโซมและวัดคุณภาพของแต่ละโครโมโซมเพื่อทำการคัดเลือกคำตอบที่ดีไว้สำหรับทำการพัฒนาไปสู่คำตอบที่ดียิ่งขึ้น

3.1 โครโมโซมและการเข้ารหัส

คำตอบจะถูกแทนด้วยโครโมโซมซึ่งเป็นสายอักขระของตัวเลข เช่น ปัญหาที่มีจำนวนสิ่งของ 10 ชิ้น จะกำหนดให้ของแต่ละชิ้นมีหมายเลขประจำตัวแน่นอนและไม่เปลี่ยนแปลง คำตอบจะอยู่ในรูปสายอักขระตัวเลขดังรูปที่ 2 จากรูปโครโมโซมจะแทนคำตอบที่จัดเรียงกล่องหมายเลข 0 เข้าก่อนจากนั้นจัด 1 5 6 3 ... 2 เข้าตามลำดับ



| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 5 | 6 | 3 | 4 | 7 | 9 | 8 | 2 |
|---|---|---|---|---|---|---|---|---|---|

รูปที่ 2 ตัวอย่างการเข้ารหัสคำตอบ

3.2 พารามิเตอร์ของขั้นตอนวิธีพันธุกรรม

การไขว้เปลี่ยนใช้วิธี Partially Matched Crossover (PMX) [4] ค่าความน่าจะเป็นในการไขว้เปลี่ยนเท่ากับ 0.9 และความน่าจะเป็นของการกลายพันธุ์เท่ากับ 0.001 เพื่อ

เลือกว่าจะทำการสุ่มเลือก 2 ตำแหน่งแล้วทำการสลับลำดับหรือไม่

ในการทดลองใช้ขนาดประชากรในแต่ละรุ่นเท่ากับ 2000 สำหรับกรณีที่ปัญหาขนาดเล็กลงกว่า 20 ชั้น สามารถลดขนาดประชากรลงเหลือเพียง 200 เพื่อช่วยลดเวลาที่ใช้ แต่จำนวนรุ่นของประชากรที่เจอคำตอบจะเพิ่มขึ้น จากการทดลองจึงพบว่าเวลาที่ใช้ไม่ได้ต่างกันสืบเท่าตามขนาดประชากรแต่ต่างกันไม่เกิน 2 เท่า เนื่องจากเป็นปัญหาขนาดเล็กเวลาที่ใช้จริงต่างกันไม่ถึงหนึ่งวินาที ในกรณีปัญหาขนาดใหญ่จำนวนประชากรมากขึ้นจะช่วยให้สามารถหาคำตอบได้มากขึ้น ในการทดลองจะทำงานจนกระทั่งพบคำตอบแล้วจึงหยุด ในกรณีที่ไม่สามารถหาคำตอบได้จะหยุดเมื่อจำนวนรุ่นเท่ากับ 200

3.3 การประเมินค่าโครโมโซม

ฟังก์ชันสำหรับวัดคุณภาพของแต่ละโครโมโซมแบ่งออกเป็นสองส่วนคือส่วนที่คิดจากจำนวนคอนเทนเนอร์ที่ใช้ และส่วนที่คิดจากความยาวของการบรรจุในคอนเทนเนอร์ใบสุดท้าย โดยจะให้ความสำคัญกับจำนวนคอนเทนเนอร์มากกว่าดังสมการ

$$f = \left(0.9 \times \frac{12}{N} \right) + \left(0.1 \times \frac{1}{1 + L/L_0} \right)$$

N คือจำนวนคอนเทนเนอร์

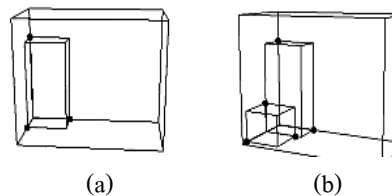
L_0 คือความกว้างของคอนเทนเนอร์

L คือความยาวการบรรจุในคอนเทนเนอร์ใบสุดท้าย

3.4 การจัดเรียงผลิตภัณฑ์

การจัดเรียงผลิตภัณฑ์เป็นแบบ Next Fit คือจัดเรียงตามลำดับที่ปรากฏในโครโมโซมเมื่อไม่สามารถบรรจุชิ้นต่อไปได้จะสร้างกล่องใหม่ขึ้นมาเพื่อบรรจุและปิดกล่องที่ผ่านมาโดยไม่ย้อนกลับ ไปบรรจุกล่องเดิมอีก

จากรูปที่ 3 แสดงให้เห็นการจัดเรียงกล่องเข้าคอนเทนเนอร์ เมื่อวางกล่องหนึ่งกล่องจะเกิดมุมสำหรับวางกล่องถัดไปเป็นจุดสี่ค่าเพิ่มเข้ามา 3 มุมและจุดที่ถูกทับจะหายไป การจัดเรียงเริ่มจากมุมด้านในสุด จุดสี่ค่าเข้มจะเป็นจุดในการพิจารณาวางกล่องถัดไป โดยจะเลือกจุดทางด้านซ้ายสุด ลึกสุดและต่ำสุดและไม่ทำให้เกิดการเหลื่อมซ้อนกัน ก่อนอื่นจะหาขนาดความกว้าง สูง และหนาน้อยที่สุดเพื่อใช้ประโยชน์ในการตัดบางตำแหน่งออกจากการพิจารณาว่าสามารถวางได้หรือไม่ หากตำแหน่งนั้นอยู่ห่างจากผนังคอนเทนเนอร์น้อยกว่าขนาดที่เล็กสุดของผลิตภัณฑ์ก็จะไม่นำมาพิจารณาอีก



รูปที่ 3 การจัดเรียงกล่องเข้าคอนเทนเนอร์

4. การค้นหาเฉพาะที่ (Local search)

เนื่องจากขั้นตอนวิธีพันธุกรรมเป็นกระบวนการค้นหาแบบครอบคลุม(global search) การค้นหาคำตอบสามารถที่จะเปลี่ยนจากบริเวณค้นหาคำตอบ(search space) ที่หนึ่งไปยังอีกแห่งหนึ่งซึ่งไม่ใกล้เคียงกันเป็นไปได้โดยง่าย จึงสามารถหาคำตอบที่ดีกว่าโดยรวมได้ง่าย แต่จะมีผลให้การค้นหาคำตอบที่ดีที่สุดเป็นไปได้ช้า วมทั้งคำตอบที่ดีย่อมมีองค์ประกอบที่ตีรวมอยู่ด้วย ดังนั้นการใช้การค้นหาเฉพาะที่จะช่วยในการปรับปรุงคุณภาพคำตอบที่ได้อยู่แล้วให้เป็นคำตอบที่ดียิ่งขึ้น

การค้นหาเฉพาะที่มีความสำคัญในการพัฒนาคำตอบอย่างมากและถูกใช้ในหลายงานวิจัย เช่น งานวิจัย [5] นำเสนอวิธีการปรับปรุงคุณภาพคำตอบของปัญหาบรรจุผลิตภัณฑ์หนึ่งมิติที่ได้จากการใช้ขั้นตอนวิธี Best Fit Decreasing (BFD) โดยทำการลดจำนวนกล่องลงเรื่อยๆ จากคำตอบที่เคยเจอ โดยการกระจายผลิตภัณฑ์จากคอนเทนเนอร์ที่ถูก

ตัดทิ้งออกไปยังคอนเทนเนอร์อื่น ๆ ที่เหลือ ถ้าเป็นคำตอบที่ถูกต้องก็จะพยายามลดจำนวนคอนเทนเนอร์ไปเรื่อย ๆ ระหว่างการลดจำนวนคอนเทนเนอร์มักจะทำให้เกิดคำตอบที่เป็นไปไม่ได้ เมื่อเจอคำตอบที่เป็นไปไม่ได้จะทำการค้นเฉพาะที่ โดยพิจารณาคอนเทนเนอร์ที่สั้นหนึ่งคู่และทำการแลกเปลี่ยนผลิตภัณฑ์เพื่อให้มีน้ำหนักรวมเท่า ๆ กัน ทำไปจนกระทั่งได้คำตอบที่เป็นไปไม่ได้ ถ้ายังไม่เจอคำตอบที่เป็นไปได้อาจจะหาจนกระทั่งครบทุกคู่ที่เป็นไปได้

นอกจากนี้ยังมีงานวิจัยที่ใช้ขั้นตอนวิธีพันธุกรรมเป็นจำนวนมากที่ใช้การค้นหาเฉพาะที่ช่วยในการปรับปรุงคุณภาพคำตอบที่เจอ เช่น [6] และ [7] ใช้ขั้นตอนวิธีพันธุกรรมร่วมกับการค้นเฉพาะที่เพื่อแก้ปัญหา TSP (Traveling Saleman Problem) ซึ่งเป็นปัญหาแนวเดียวกับปัญหาบรรจุผลิตภัณฑ์คือคำตอบที่ใช้จะอยู่ในรูปการเรียงสับเปลี่ยนลำดับของตัวเลข (combinatorial optimization) การแก้ปัญหานี้โดย GA มักจะหนีไม่พ้นการค้นหาเฉพาะที่ [8]

งานวิจัย [9] เพิ่มความเร็วให้กับขั้นตอนวิธีพันธุกรรมด้วยการค้นเฉพาะที่ซึ่งเป็นการโปรแกรมไม่เชิงเส้น (nonlinear programming) ไม่เพียงแต่สามารถทำให้เร็วขึ้นยังทำให้คำตอบที่ได้ถูกต้องมากยิ่งขึ้น

ในกระบวนการของขั้นตอนวิธีพันธุกรรมที่นำเสนอจะเพิ่มกระบวนการค้นหาเฉพาะที่เข้าไปในระหว่างที่ทำการประเมินค่าโครโมโซม โดยจะสุ่มเพื่อสลับตำแหน่งหมายเลขหนึ่งคู่แล้ววัดผล ทำอย่างนี้ห้าครั้งสำหรับทุก ๆ คำตอบ ถ้าพบว่าการสลับลำดับหนึ่งคู่แบบไหนให้คำตอบที่ดีกว่าจะใช้เป็นคำตอบแทนคำตอบเดิม

ได้ทดลองเปรียบเทียบระหว่างขั้นตอนวิธีพันธุกรรมที่ใช้และไม่ใช้การค้นหาเฉพาะที่ โดยวัดคุณภาพการแก้ปัญหาด้วยจำนวนปัญหาที่สามารถหาคำตอบได้ (คำตอบเทียบกับคำตอบที่ดีที่สุดด้วยวิธีของ [3]) จากตารางที่ 1 และตารางที่ 2 จะเห็นได้ว่า ขั้นตอนวิธีพันธุกรรมที่ใช้การค้นหาเฉพาะที่แก้จำนวนปัญหาได้มากกว่า

ตารางที่ 1. จำนวนปัญหาที่สามารถหาคำตอบได้ของ GA

| ชั้น | กลุ่ม | | | | | | | | |
|------|-------|----|----|----|----|----|----|----|----|
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 |
| 15 | 10 | 9 | 10 | 10 | 10 | 10 | 10 | 10 | 4 |
| 20 | 8 | 9 | 9 | 10 | 9 | 10 | 10 | 10 | 4 |
| 25 | 6 | 5 | 6 | 10 | 9 | 10 | 10 | 7 | - |
| 30 | 6 | 2 | 3 | 10 | 7 | 10 | 10 | 4 | - |

ตารางที่ 2. จำนวนปัญหาที่สามารถหาคำตอบได้ของ GA + local search

| ชั้น | กลุ่ม | | | | | | | | |
|------|-------|----|----|----|----|----|----|----|----|
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 |
| 15 | 10 | 9 | 10 | 10 | 10 | 10 | 10 | 10 | 6 |
| 20 | 9 | 9 | 10 | 10 | 10 | 10 | 10 | 10 | 3 |
| 25 | 7 | 6 | 8 | 10 | 9 | 10 | 10 | 8 | - |
| 30 | 9 | 6 | 6 | 10 | 8 | 10 | 10 | 6 | - |

5. ผลการทดลอง

การเปรียบเทียบวิธี [3] กับวิธีที่เสนอทำภายใต้สภาวะเดียวกันคือ เครื่องคอมพิวเตอร์ Pentium III ความเร็วหน่วยประมวลผลกลาง 1 GHz หน่วยความจำหลัก 128 Mbyte ระบบปฏิบัติการ Linux Mandrake 9.0 การทดลองของงานวิจัย [3] จะให้ผลเวลาทำกันทุกครั้งสำหรับปัญหาใดปัญหาหนึ่ง จึงทำการทดลองเพียงรอบเดียว ส่วนขั้นตอนวิธีที่นำเสนอเป็นขั้นตอนวิธีเชิงคู่การทำงานแต่ละครั้งใช้เวลาไม่คงที่ จึงทำการทดลองกับแต่ละปัญหาเป็นจำนวน 5 ครั้ง แล้วเลือกเวลาที่ดีสุดจาก 5 ครั้ง เนื่องจากขั้นตอนวิธีในงานวิจัย [3] ใช้เวลาในการค้นหาบางกรณีนานมาก ๆ บางปัญหาใช้เวลามากกว่าชั่วโมง เมื่อทำการกำหนดเวลาที่ใช้ค้นหาคำตอบไว้ที่ 3000 วินาทีต่อหนึ่งปัญหา ตารางที่ 4 แสดงเวลาโดยเฉลี่ยที่ใช้ในการหาคำตอบเฉพาะปัญหาที่สามารถหาคำตอบได้ ตามวิธี [3] (3DBPP) จากผลการทดลองในตารางที่ 3 และ 5 แสดงให้เห็นว่าความสามารถในการหาคำตอบของขั้นตอนวิธีพันธุกรรมร่วมกับการค้นเฉพาะที่สามารถหาคำตอบได้ดีพอ ๆ กับวิธีแบบศึกษาสำนักของงานวิจัย[3] ในปัญหาขนาดเล็ก และดีกว่าในปัญหาขนาดใหญ่ ในขณะที่เวลาที่

ใช้ของชั้นตอนวิธีพันธุกรรมจะมากกว่าในกรณีที่มีปัญหาขนาดเล็ก แต่ก็สามารถหาคำตอบได้เร็วกว่าในปัญหาที่มีขนาดใหญ่

มีข้อสังเกตว่า เมื่อปัญหาขนาดใหญ่ขึ้น วิธีที่เสนอมีอัตราการเพิ่มของเวลาน้อยกว่าวิธี 3DBPP จะเห็นได้ชัดในปัญหาทุกกลุ่ม เช่นเปรียบเทียบโดยปัญหากลุ่ม 2 และ 8 ซึ่งทั้งสองวิธีสามารถแก้ปัญหาได้ในจำนวนที่พอๆ กัน แนวโน้มเชิงความเร็วในการแก้ปัญหาของวิธีที่เสนอนี้จึงน่าจะดีกว่าวิธี 3DBPP ในปัญหาขนาดใหญ่

6. สรุป

เวลาที่ใช้ในการหาคำตอบของปัญหาบรรจุผลิตภัณฑ์เป็นอุปสรรคสำคัญอย่างมากของขั้นตอนวิธีโดยทั่วไป การใช้ขั้นตอนวิธีเชิงพันธุกรรมร่วมกับการค้นหาเฉพาะที่สามารถแก้ปัญหาบรรจุผลิตภัณฑ์ในสามมิติที่มีขนาดของปัญหาใหญ่ได้ดีกว่าวิธีแตกกิ่งและจำกัดเขตโดยใช้ขอบเขตต่างที่นำเสนอโดยงานวิจัย [3]

ปัญหาต่อไปที่ควรวิจัย คือ ปรับปรุงวิธีการค้นหาเฉพาะที่โดยเพิ่มวิธีการศึกษาสำนักหลาย ๆ แบบ ที่มีผู้เสนอใช้กับปัญหาอื่น ๆ น่าจะให้ผลดียิ่งขึ้น กว่าวิธีค้นหาเฉพาะที่แบบง่ายที่ใช้ในการทดลองนี้

7. เอกสารอ้างอิง

- [1] C. Pimpawat, and N. Chaiyaratana, "Using a cooperative co-evolutionary genetic algorithm to solve a three-dimensional container loading problem", Proc. of IEEE on Evolutionary Computation, 2001, pp. 1197-1204.
- [2] W. Leinberger, G. Karypis, and V. Kumar, "Multi-capacity bin packing algorithms with applications to job scheduling under multiple constraints", Int. Conf. on Parallel Processing, 1999, pp. 404-412.
- [3] S. Martello, D. Pisinger, D. Vigo, "The three-dimensional bin packing problem", Operations Research, 2000, pp. 256-267.
- [4] Goldberg D. E., Genetic Algorithm in Search Optimization and Machine Learning, Addison Wesley, New York, 1989.
- [5] A.C. Alvim, F. Glover, C.C. Ribeiro, and D.J. Aloise, "Local Search For The Bin Packing Problem", Extended Abstracts of the 3rd Metaheuristics Int. Conf., Angra dos Reis, 1999, pp. 7-12.
- [6] B. Freisleben, P. Merz, "A Genetic Local Search Algorithm for Solving Symmetric and Asymmetric Traveling Salesman Problems", Proc. of IEEE Int. Conf. on Evolutionary Computation, May 1996, pp. 616-621.
- [7] P. Merz, B. Freisleben, "Genetic local search for the TSP: new results", Proc. of IEEE Int. Conf. on Evolutionary Computation, 1997, pp. 159-164.
- [8] T. Yamada, C.R. Reeves, "Permutation Flowshop Scheduling by Genetic Local Search", 2nd Int. Conf. On Genetic Algorithms In Engineering Systems: Innovations And Applications, 1997, pp. 232 -238.
- [9] C.K. Chak, G. Feng, "Accelerated Genetic Algorithms: Combined with Local Search Techniques for Fast and Accurate Global Search", IEEE Int. Conf. on Evolutionary Computation, 1995, pp. 378.

ตารางที่ 3. จำนวนปัญหาที่สามารถหาคำตอบได้ของ 3DBPP

| ชั้น | กลุ่ม | | | | | | | | | รวม |
|------|-------|----|----|-----|----|-----|----|----|----|-----|
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | |
| 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 90 |
| 15 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 90 |
| 20 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 90 |
| 25 | 10 | 10 | 10 | 10 | 10 | 10 | 9 | 10 | 10 | 89 |
| 30 | 10 | 10 | 10 | 10 | 10 | 10 | 8 | 10 | 10 | 88 |
| 35 | 10 | 10 | 10 | 10 | 6 | 10 | 6 | 7 | 10 | 79 |
| 40 | 10 | 9 | 8 | 10 | 6 | 10 | 2 | 10 | 8 | 73 |
| 45 | 10 | 7 | 4 | 10 | 9 | 10 | 1 | 8 | 1 | 60 |
| 50 | 7 | 4 | 8 | 10 | 3 | 10 | 3 | 6 | - | 51 |
| 60 | 1 | 2 | 1 | 9 | 1 | 10 | - | 6 | - | 30 |
| 70 | - | 1 | - | 6 | 3 | 7 | 2 | 5 | - | 24 |
| 80 | - | - | 1 | 4 | - | 5 | - | 4 | - | 14 |
| 90 | - | - | - | 2 | - | 7 | - | 2 | - | 11 |
| รวม | 88 | 83 | 82 | 111 | 78 | 119 | 61 | 98 | 69 | 789 |

ตารางที่ 4. ค่าเฉลี่ยเวลาในการหาคำตอบโดยวิธี 3DBPP

| ชั้น | กลุ่ม | | | | | | | | |
|------|--------|---------|--------|--------|--------|--------|---------|---------|--------|
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 10 | 0.01 | 0.01 | 0.01 | < 0.01 | 0.03 | 0.02 | 0.03 | 0.03 | < 0.01 |
| 15 | 0.01 | 0.01 | 0.01 | < 0.01 | 0.06 | 0.04 | 0.06 | 0.06 | < 0.01 |
| 20 | 0.04 | 0.02 | 0.02 | 0.01 | 0.11 | 0.08 | 0.09 | 0.08 | 0.03 |
| 25 | 0.24 | 0.06 | 0.40 | 0.01 | 2.16 | 0.17 | 1.45 | 0.20 | 0.07 |
| 30 | 16.99 | 1.92 | 2.11 | 0.03 | 0.27 | 3.97 | 155.80 | 0.38 | 10.08 |
| 35 | 38.45 | 46.24 | 10.50 | 0.04 | 0.48 | 1.77 | 58.30 | 322.53 | 94.51 |
| 40 | 125.29 | 136.67 | 60.63 | 0.04 | 100.30 | 7.94 | 0.22 | 170.94 | 553.56 |
| 45 | 810.06 | 450.99 | 8.72 | 0.07 | 1.72 | 12.56 | 0.29 | 87.56 | 92.21 |
| 50 | 491.11 | 826.12 | 752.70 | 6.36 | 56.17 | 6.00 | 1099.41 | 280.36 | - |
| 60 | 16.13 | 662.77 | 841.01 | 5.76 | 2.30 | 154.68 | - | 269.66 | - |
| 70 | - | 1032.12 | - | 3.46 | 443.71 | 236.25 | 498.02 | 309.75 | - |
| 80 | - | - | 0.90 | 1.39 | - | 162.47 | - | 1167.54 | - |
| 90 | - | - | - | 72.29 | - | 111.35 | - | 204.55 | - |

ตารางที่ 5. จำนวนปัญหาที่สามารถหาคำตอบได้ของ GA + local search

| ชั้น | กลุ่ม | | | | | | | | | รวม |
|------|-------|----|----|-----|-----|----|-----|----|----|-----|
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | |
| 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 90 |
| 15 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 90 |
| 20 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 90 |
| 25 | 10 | 8 | 10 | 10 | 10 | 10 | 9 | 9 | 7 | 83 |
| 30 | 10 | 10 | 9 | 10 | 9 | 9 | 9 | 10 | - | 76 |
| 35 | 6 | 7 | 7 | 10 | 10 | 5 | 7 | 9 | - | 61 |
| 40 | 4 | 6 | 8 | 10 | 9 | 3 | 10 | 6 | 2 | 58 |
| 45 | 3 | 8 | 6 | 10 | 9 | 2 | 10 | 4 | 9 | 61 |
| 50 | 4 | 4 | 4 | 10 | 9 | - | 9 | 5 | 10 | 55 |
| 60 | 5 | 5 | 2 | 10 | 10 | - | 8 | 4 | 8 | 52 |
| 70 | 4 | 3 | - | 10 | 7 | - | 7 | 4 | 9 | 44 |
| 80 | 1 | 3 | 1 | 10 | 9 | - | 4 | 4 | 9 | 41 |
| 90 | 1 | 1 | - | 9 | 7 | - | 6 | - | 9 | 33 |
| รวม | 78 | 85 | 77 | 129 | 119 | 59 | 109 | 85 | 93 | 834 |

ตารางที่ 6. ค่าเฉลี่ยเวลาในการหาคำตอบโดยวิธี GA + local search

| ชั้น | กลุ่ม | | | | | | | | |
|------|-------|-------|-------|------|--------|-------|--------|-------|-------|
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 10 | 0.30 | 0.32 | 0.32 | 0.28 | 0.48 | 0.35 | 0.45 | 0.35 | 0.28 |
| 15 | 0.34 | 0.45 | 0.40 | 0.31 | 0.73 | 0.42 | 0.61 | 0.45 | 0.78 |
| 20 | 0.42 | 0.50 | 0.64 | 0.33 | 1.35 | 0.52 | 1.13 | 0.53 | 3.83 |
| 25 | 1.47 | 4.04 | 0.94 | 0.35 | 2.40 | 5.72 | 5.55 | 2.01 | 8.59 |
| 30 | 2.16 | 5.95 | 3.53 | 0.45 | 3.74 | 6.58 | 6.14 | 6.63 | - |
| 35 | 9.40 | 6.38 | 8.24 | 0.52 | 4.76 | 9.45 | 3.26 | 14.04 | - |
| 40 | 11.43 | 8.29 | 12.71 | 0.94 | 11.27 | 19.59 | 3.64 | 24.89 | 1.33 |
| 45 | 22.11 | 11.92 | 7.93 | 0.98 | 16.02 | 30.99 | 15.75 | 22.87 | 6.12 |
| 50 | 18.81 | 24.13 | 27.27 | 2.54 | 18.89 | - | 29.65 | 19.48 | 2.34 |
| 60 | 34.84 | 48.21 | 50.40 | 1.98 | 39.64 | - | 41.55 | 30.64 | 4.31 |
| 70 | 39.22 | 74.59 | - | 4.42 | 79.80 | - | 87.18 | 71.92 | 10.82 |
| 80 | 67.69 | 70.69 | 89.47 | 5.92 | 91.39 | - | 100.10 | 85.16 | 10.32 |
| 90 | 78.22 | 80.93 | - | 8.62 | 174.61 | - | 161.77 | - | 13.70 |