

รายงานฉบับสมบูรณ์

ขั้นตอนวิธีแบบเพิ่มค่าสำหรับการค้นหาเว็บบนเครือข่ายเพียร์ทูเพียร์

A Peer-to-Peer-Based Incremental Web Searching Algorithm

รหัสโครงการ

SCH-NR2012-103

พฤษภ บัญมา

ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเชียงใหม่

รายงานฉบับสมบูรณ์

ขั้นตอนวิธีแบบเพิ่มค่าสำหรับการค้นหาเว็บบนเครือข่ายเพียร์ทูเพียร์

A Peer-to-Peer-Based Incremental Web Searching Algorithm

รายนามคณะผู้วิจัย

พฤษภ บัญมา

ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเชียงใหม่

วันที่เริ่มต้นโครงการ 1 กรกฎาคม 2556

วันที่สิ้นสุดโครงการ 1 กรกฎาคม 2557

(...../ตรวจสอบเนื้อหาแล้ว)

กิตติกรรมประกาศ

โครงการวิจัยนี้ได้รับการสนับสนุนจาก
ศูนย์ประสานงานนักเรียนทุนรัฐบาลทางด้านวิทยาศาสตร์และเทคโนโลยี
กระทรวงวิทยาศาสตร์และเทคโนโลยี
และสำนักงานพัฒนาวิทยาศาสตร์และเทคโนโลยีแห่งชาติ

ปี พ.ศ. ที่ได้รับทุน 2556

บทคัดย่อ

กูเกิลเพจแรงค์ (Google's PageRank) เป็นอัลกอริทึมที่สำคัญอันหนึ่งในการทำเครื่องจักรค้นหาเว็บ (Web search engine) โดยในกูเกิลเพจแรงค์ เว็บเพจทั้งหมดจะถูกแทนด้วยเว็บลิงก์กราฟ (Web-link graph) โดยโหนดของกราฟจะใช้แทนเว็บเพจ และเอ็ดจ์ของกราฟจะใช้แทนเว็บลิงค์ การคำนวณเพจแรงค์จะทำงานได้อย่างไม่ค่อยมีประสิทธิภาพหากเว็บลิงก์กราฟมีขนาดใหญ่มาและมีการเพิ่มขนาดเว็บลิงก์กราฟ เนื่องจากต้องทำการคำนวณใหม่ทั้งหมด เพื่อแก้ปัญหาดังกล่าว จึงมีผู้นำเสนอแนวคิดการคำนวณเพจแรงค์บนระบบเครือข่ายแบบเพียร์ทูเพียร์ โดยในแต่ละเพียร์จะเก็บเว็บลิงก์กราฟไว้บางส่วน ซึ่งเว็บลิงก์กราฟที่ถูกเก็บไว้ในแต่ละเพียร์ก็คือกราฟย่อยของเว็บลิงก์กราฟขนาดใหญ่ แล้วแต่ละเพียร์จะแยกกันคำนวณเพจแรงค์แบบกระจายศูนย์ จำนวนรอบที่ใช้ในการคำนวณเพจแรงค์จะขึ้นอยู่กับความหนาแน่นของกราฟย่อยที่ถูกเก็บไว้ในแต่ละเพียร์ โดยหากกราฟย่อยมีความหนาแน่นสูง จะทำให้จำนวนรอบในการคำนวณเพจแรงค์ก็จะสูงตามไปด้วย และจะส่งผลถึงการคำนวณเพจแรงค์ของเครือข่ายทั้งหมด เพราะในการคำนวณเพจแรงค์บนเครือข่ายเพียร์ทูเพียร์นั้นจะนับเวลาในการคำนวณเพจแรงค์จากเพียร์ที่ทำงานช้าที่สุด การทำให้เว็บลิงก์กราฟที่ถูกเก็บไว้ในแต่ละเพียร์มีค่าความหนาแน่นใกล้เคียงกันจะช่วยลดเวลาของการคำนวณเพจแรงค์บนเครือข่ายแบบกระจายศูนย์ลงได้ และเมื่อมีการเพิ่มขนาดเว็บลิงก์กราฟ และต้องทำการคำนวณเพจแรงค์ใหม่ ในงานวิจัยนี้จึงได้เสนอปัญหาแบ่งกราฟแบบความหนาแน่นสมดุลขึ้น เพื่อให้การคำนวณเพจแรงค์บนระบบแบบกระจายศูนย์มีประสิทธิภาพมากยิ่งขึ้น โดยสามารถใช้ในการแบ่งกราฟทั้งในการทำงานเริ่มต้น (Start-up) และการทำงานแบบเพิ่มค่า (Incremental) ซึ่งจากผลการทดลองแสดงให้เห็นว่า วิธีการที่นำเสนอทำให้สามารถแบ่งกราฟให้มีความหนาแน่นสมดุลมากขึ้น เมื่อเทียบกับการแบ่งแบบดั้งเดิม

Abstract

Google's PageRank is the most notable approach for web search ranking. In general, web pages are represented by web-link graph; a web-page is represented by a node, and a link between two pages is represented by an edge. In particular, it is not efficient both to perform PageRank of a large web-link graph in a single computer and when new web pages are added to the web-link graph. Distributed systems, such as P2P, are viable choices to address such limitation. In P2P-based PageRank, each computational peer contains a partial web-link graph, i.e., a sub-graph of the global web-link graph, and its PageRank is computed locally. The convergence time of a PageRank calculation is affected by the web-link graph density, i.e., the ratio of the number of edges to the number of nodes, such that if a web-link graph has high density, it will take longer time to converge. As the execution time to compute the P2P-based web ranking is influenced by the execution time of the slowest peer to compute the local ranking, the density-balanced local web-link graph partitioning can be highly desirable. This paper addresses a density-balanced partitioning problem and proposes an efficient algorithm for the problem that will benefit both for starting up and also when web pages are incrementally increased. The experiment results show that the proposed algorithm can effectively partition graph into density-balanced sub-graph with an acceptable cost.

	หน้า
1 บทนำ	1
2 วัตถุประสงค์ของงานวิจัย	2
3 การทบทวนวรรณกรรม	2
4 การออกแบบปัญหาในงานวิจัย	3
5 ขอบเขตของงานวิจัย	9
6 ระเบียบวิจัย	9
7 ผลการวิจัย	11
8 สรุปผล	16
9 ปัญหาและอุปสรรค	16

สารบัญภาพ

	หน้า
ภาพที่ 1 ตัวอย่าง Edges separator	6

ภาพที่ 2	ตัวอย่าง Binary tree decomposition	7
ภาพที่ 3	ขั้นตอนวิธี DBP	8
ภาพที่ 4	ตัวอย่างโครงสร้าง Random graph	9
ภาพที่ 5	ตัวอย่างโครงสร้าง Euclidian graph	9
ภาพที่ 6	แสดงผลของค่าคงที่ α ที่มีผลกับค่าความไม่สมดุลของความหนาแน่นในการแบ่ง Random graph	11
ภาพที่ 7	แสดงผลของค่าคงที่ α ที่มีผลกับจำนวนลักษณะการแบ่งกราฟย่อยที่ใช้งานได้ในการแบ่ง Random graph	11
ภาพที่ 8	แสดงประสิทธิภาพของขั้นตอนวิธี DBP ในแง่ของการแบ่งกราฟแบบความหนาแน่นสมดุลในการแบ่ง Random graph	13
ภาพที่ 9	แสดงประสิทธิภาพของขั้นตอนวิธี DBP ในแง่ของ Execution time ในการแบ่ง Random graph	13
ภาพที่ 10	แสดงผลของค่าคงที่ α ที่มีผลกับค่าความไม่สมดุลของความหนาแน่นในการแบ่ง Watts-Strogatz graph	14
ภาพที่ 11	แสดงผลของค่าคงที่ α ที่มีผลกับจำนวนลักษณะการแบ่งกราฟย่อยที่ใช้งานได้ในการแบ่ง Watts-Strogatz graph	14
ภาพที่ 12	แสดงประสิทธิภาพของขั้นตอนวิธี DBP ในแง่ของการแบ่งกราฟแบบความหนาแน่นสมดุลในการแบ่ง Watts-Strogatz graph	15
ภาพที่ 13	แสดงประสิทธิภาพของขั้นตอนวิธี DBP ในแง่ของ Execution time ในการแบ่ง Watts-Strogatz graph	15

1. บทนำ

กูเกิลเพจเรงก์ (Google's PageRank) [1][2] เป็นอัลกอริทึมที่สำคัญอันหนึ่งในการทำเครื่องจักรค้นหาเว็บ (Web Search engine) ในการทำงานของกูเกิลเพจเรงก์ เว็บเพจทั้งหมดจะถูกแทนด้วยเว็บลิงค์กราฟ (Web-link graph) โดยโหนดของกราฟจะใช้แทนเว็บเพจ และเอดจ์ของกราฟจะใช้แทนเว็บลิงค์ การคำนวณเพจเรงก์จะทำงานได้อย่างไม่ค่อยมีประสิทธิภาพหากเว็บลิงค์กราฟมีขนาดใหญ่มากๆ เนื่องจากเวลาที่ใช้การคำนวณเพจเรงก์ค่อนข้างสูง นอกจากนั้น เมื่อมีการเพิ่มจำนวนเว็บเพจเข้าไปยังเว็บลิงค์กราฟ ก็ต้องทำการคำนวณเพจเรงก์ใหม่ทั้งหมด

แนวคิดการคำนวณเพจเรงก์บนระบบเครือข่ายแบบเพียร์ทูเพียร์จึงถูกพัฒนาขึ้น แต่ละเพียร์จะเก็บเว็บลิงค์กราฟไว้บางส่วน ซึ่งเว็บลิงค์กราฟที่ถูกเก็บไว้ในแต่ละเพียร์ก็คือกราฟย่อยของเว็บลิงค์กราฟขนาดใหญ่ ซึ่งแต่ละเพียร์จะแยกกันคำนวณเพจเรงก์แบบกระจายศูนย์ จำนวนรอบที่ใช้ในการคำนวณ

เพลงแรงจะขึ้นอยู่กับความหนาแน่นของกราฟย่อยที่ถูกเก็บไว้ในแต่ละเพียร์ ซึ่งพิจารณาจากอัตราส่วนระหว่างจำนวนเอดจ์ของกราฟย่อยหารด้วยจำนวนโหนดของกราฟย่อยที่ถูกเก็บไว้ในแต่ละเพียร์ หากกราฟย่อยมีค่าความหนาแน่นสูงจำนวนรอบในการคำนวณเพลงแรงก็จะสูงตามไปด้วย นอกจากนั้น เมื่อมีการเพิ่มจำนวนเว็บเพจแบบเพิ่มค่า (Incremental) ก็ต้องทำการรวมกราฟและแยกกราฟเป็นกราฟย่อยในแต่ละเพียร์อีกครั้งหนึ่ง

ในการคำนวณเพลงแรงบนเครือข่ายเพียร์ทูเพียร์นั้นจะนับเวลาในการคำนวณเพลงแรงจากเพียร์ที่ทำงานช้าที่สุด การทำให้เว็บลิงค์กราฟที่ถูกเก็บไว้ในแต่ละเพียร์มีค่าความหนาแน่นใกล้เคียงกันจะช่วยลดเวลาของการคำนวณเพลงแรงบนเครือข่ายแบบกระจายศูนย์ลงได้ แต่การคำนวณเพลงแรงบนระบบแบบกระจายศูนย์จำเป็นต้องคำนึงถึงความผิดพลาดของค่าเพลงแรงด้วย เนื่องจากการพิจารณาเพลงแรงจากเว็บลิงค์กราฟเฉพาะบางส่วนอาจมีบางเว็บลิงค์ที่ไม่ถูกพิจารณา

ในงานวิจัยนี้จึงได้เสนอปัญหาแบ่งกราฟแบบความหนาแน่นสมดุลขึ้น เพื่อให้การคำนวณเพลงแรงบนระบบแบบกระจายศูนย์มีประสิทธิภาพมากยิ่งขึ้น โดยได้นำเสนอขั้นตอนวิธีที่ชื่อว่า DBP สำหรับแบ่งกราฟแบบความหนาแน่นสมดุล โดยขั้นตอนวิธี DBP นอกจากจะพิจารณาเฉพาะความสมดุลของขนาดกราฟย่อยแล้ว ยังพิจารณาค่าความไม่สมดุลของความหนาแน่นเข้ามาประกอบด้วย เพื่อให้ได้กราฟย่อยที่มีความหนาแน่นใกล้เคียงกัน รวมทั้งขั้นตอนวิธีนี้จะแบ่งกราฟย่อยให้มีเอดจ์ที่เชื่อมกันระหว่างกราฟย่อยน้อย เพื่อลดขนาดของความผิดพลาดของค่าเพลงแรง

ผลการทดลองขั้นตอนวิธี DBP มี 2 ส่วนคือ ส่วนแรกจะเป็นการหาค่าคงที่ค่าหนึ่งที่ส่งผลกระทบต่อค่าความไม่สมดุลของความหนาแน่นในการแบ่งกราฟแต่ละแบบ รวมทั้งหาค่าคงที่ค่าที่ส่งผลกระทบต่อจำนวนลักษณะของการแบ่งกราฟที่นำไปใช้งานได้ ส่วนที่สองเป็นการเปรียบเทียบประสิทธิภาพการแบ่งกราฟแบบความหนาแน่นสมดุลของขั้นตอนวิธี DBP กับขั้นตอนวิธีในการแบ่งกราฟสมดุลขั้นตอนวิธีอื่นๆ

2. วัตถุประสงค์ของโครงการวิจัย

เพื่อแบ่งกราฟย่อยจากกราฟใหญ่ที่มีความหนาแน่นแบบสมดุลในขณะที่เอดจ์ระหว่างกราฟย่อยมีจำนวนไม่มาก สำหรับหาหลักการแบ่งกราฟย่อยที่เหมาะสมสำหรับการเริ่มต้นการทำงาน และการแบ่งกราฟย่อยเมื่อมีการเพิ่มค่าของกราฟ ของขั้นตอนวิธีในการคำนวณเพลงแรงบนระบบแบบกระจายศูนย์อย่างมีประสิทธิภาพ

3. การทบทวนวรรณกรรม

ในกระบวนการค้นหาเว็บไซต์ที่ใช้ขั้นตอนวิธีเพจเรงก์ (PageRank Algorithm) [3] ที่นำเสนอโดย ลาร์รี เพจ (Larry Page) และ เซอร์เกย์ บริน (Sergey Brin) จากมหาวิทยาลัยสแตนฟอร์ดในปี คศ. 1996 นั้น จะอาศัยการหาคำน้้ำหนักของเว็บไซต์หนึ่งโดยดูจากความสัมพันธ์ระหว่างลิงค์ที่อ้างอิงมายัง เว็บไซต์นี้และลิงค์ที่อ้างอิงออกไปจากเว็บไซต์นี้ ซึ่งน้ำหนักของเว็บไซต์จะเป็นสิ่งกำหนดผลการค้นหาและ ลำดับของเว็บไซต์ในผลของการค้นหา

การหาคำน้้ำหนักของเว็บไซต์เป็นการประมวลผลตามแบบจำลองสโโทแคสติก (Stochastic Model) ซึ่งใช้แบบจำลองของมาร์คอฟ (Markov model) [4] บนการเชื่อมโยงเว็บไซต์แบบกราฟ (Graph) ซึ่งมีเว็บไซต์แต่ละหน้าเป็นโหนด (Node) ในกราฟ และลิงค์ที่เชื่อมโยงแต่ละเว็บไซต์คือเอ็ดจ์ (Edge) ในกราฟ คำน้้ำหนักของเว็บไซต์จะเป็นตัวเลขที่แสดงถึงโอกาสที่เว็บไซต์นั้นจะถูกเข้าชม หากมี ลิงค์เข้ามามากโอกาสที่จะถูกเข้าชมก็มาก หรือหากมีลิงค์เข้าน้อยโอกาสที่จะถูกเข้าชมก็น้อยตามไปด้วย ดังนั้น คำน้้ำหนักของเว็บไซต์จึงจะพิจารณาจากจำนวนลิงค์ที่เชื่อมเข้ามายังเว็บนั้น หน้าเว็บที่เป็นต้นทาง ของลิงค์เหล่านั้น ลิงค์ที่เชื่อมโยงออกไปยังเว็บอื่น ๆ และหน้าเว็บปลายทางลิงค์เหล่านั้น การคำนวณค่า น้ำหนักของเว็บไซต์ตามหลักการของโมเดลมาร์คอฟจะเป็นการคำนวณซ้ำเดิมเรื่อยๆ ที่สถานะ (State) จนกว่าผลรวมของความต่างระหว่างสถานะปัจจุบัน และสถานะก่อนหน้ามีค่าน้อยกว่าค่าขีดค้นหายุติ ทำงาน (Termination Threshold) ซึ่งเรียกการทำงานในลักษณะนี้ว่าสถานะลู่เข้า (Convergence State)

จากหลักการดังกล่าวเพจ และ บริน ได้คิดขั้นตอนวิธีในการคำนวณค่าน้้ำหนักของเว็บไซต์ขึ้นมา โดยการสร้างเป็นสมการในการหาคำน้้ำหนักของเว็บไซต์ และเรียกค่าน้้ำหนักของเว็บไซต์ที่คำนวณได้จาก สมการดังกล่าวว่าค่าเพจเรงก์ เพื่อใช้สำหรับจัดลำดับเว็บไซต์ในผลการค้นหาหน้าเว็บ การคำนวณค่าเพจ เรงก์นั้นจำเป็นต้องใช้เนื้อที่ในหน่วยความจำค่อนข้างสูงในการเก็บค่าเพจเรงก์ของแต่ละเว็บไซต์ และยัง กราฟของเว็บไซต์มีขนาดใหญ่ จะส่งผลให้การคำนวณค่าเพจเรงก์ เพื่อเข้าสู่สถานะลู่เข้าใช้จำนวนรอบ มากตามไปด้วย ซึ่งส่งผลให้ใช้เวลาในการคำนวณนานมากขึ้นตามไปด้วย

ขั้นตอนวิธี JXP [5] เป็นขั้นตอนวิธีสำหรับคำนวณค่าเพจเรงก์บนระบบเครือข่ายแบบเพียร์ทู เพียร์ โดยหลักการของ JXP คือในแต่ละเพียร์จะเก็บเว็บไซต์กราฟย่อยไว้ และมีการคำนวณค่าน้้ำหนัก ของเว็บไซต์ในแต่ละเพียร์ ซึ่งการคำนวณค่าน้้ำหนักของเว็บไซต์ในแต่ละเพียร์นี้จะช่วยลดการใช้ หน่วยความจำและลดจำนวนรอบในการคำนวณค่าน้้ำหนักของเว็บไซต์ลงได้ รวมทั้ง JXP ยังทำให้เกิดค่า ความผิดพลาดของน้ำหนักเว็บไซต์น้อยมาก เมื่อเทียบกับการคำนวณกราฟขนาดใหญ่ทั้งหมด

ในขั้นตอนวิธี JXP เว็บไซต์ในแต่ละเพียร์ถูกเก็บในแบบสุ่ม เนื่องจากเป็นการทำงานแบบอิสระต่อกันของครอว์เลอร์ (Crawler) ที่ทำหน้าที่เก็บข้อมูลเว็บไซต์จากเว็บไซต์ต่าง ๆ ในแต่ละเพียร์ ซึ่งการเก็บ แบบสุ่มนี้ ส่งผลให้เว็บไซต์ที่เก็บอยู่ในแต่ละเพียร์มีจำนวนเว็บและจำนวนลิงค์ที่เชื่อมโยงไม่เท่ากัน ซึ่ง ส่งผลให้ความหนาแน่นของกราฟย่อย ซึ่งคำนวณจากสัดส่วนของจำนวนลิงค์และจำนวนเว็บไม่เท่ากันไป

ด้วย การที่แต่ละเพียร์มีความหนาแน่นกราฟย่อยไม่เท่ากันจะส่งผลให้การคำนวณเพจแรงค์เพื่อให้เข้าสู่สถานะลู่เข้าใช้เวลาต่างกันมาก และถ้ามีลิงค์ที่เชื่อมโยงระหว่างเว็บลิงค์กราฟย่อยในแต่ละเพียร์จำนวนมาก ก็จะส่งผลให้การคำนวณเพจแรงค์มีความผิดพลาดสูง ผู้วิจัยจึงตั้งสมมติฐานว่า ถ้าสามารถแบ่งเว็บเพจไปไว้ในแต่ละเพียร์ โดยให้มีความหนาแน่นของเว็บเพจในแต่ละเพียร์ใกล้เคียงกัน และจำนวนลิงค์ที่เชื่อมโยงระหว่างเว็บลิงค์กราฟย่อยในแต่ละเพียร์มีจำนวนน้อย จะส่งผลถึงความเร็วในการทำงานของขั้นตอนวิธี JXP ที่เพิ่มขึ้น และยังสามารถลดความผิดพลาดในการคำนวณเพจแรงค์ได้ด้วย

ในงานวิจัยเดิมของผู้วิจัย [6] ยังได้พิจารณาถึงกรณีที่มีการเพิ่มจำนวนเว็บเพจเข้าไปยังเว็บลิงค์กราฟ ซึ่งในการทำงานของ JXP จะต้องทำการคำนวณเพจแรงค์ใหม่ทั้งหมด แต่ในงานวิจัยเดิม ผู้วิจัยได้ใช้ขบวนการแบบเพิ่มค่า (Incremental Method) ในการคำนวณเพจแรงค์ในแต่ละเพียร์เพื่อมิให้ต้องทำการเพจแรงค์ใหม่หมดทั้งเว็บลิงค์กราฟ แต่เมื่อมีการคำนวณแบบเพิ่มค่าแล้ว ก็ยังต้องทำการรวมกราฟและแยกกราฟระหว่างเพียร์ใหม่ตามขั้นตอนของ JXP ซึ่งในการแยกกราฟก็ต้องพิจารณาถึงความหนาแน่นของกราฟย่อยใหม่อีกด้วย ดังนั้นถ้ามีการปรับปรุงการแยกข้อมูลโดยใช้วิธีการที่น่าเสนอ ก็จะช่วยเพิ่มประสิทธิภาพในการทำงานของ JXP ได้เช่นกัน

4. การออกแบบการวิจัย

การบัญญัติปัญหาวิจัย

หลังจากทำการทบทวนวรรณกรรม และได้กำหนดลักษณะปัญหาที่จะศึกษาแน่นอนแล้ว จึงทำการบัญญัติปัญหาที่ต้องการศึกษา โดยเรียกว่าปัญหาที่ต้องการจะศึกษาว่า ปัญหาการแบ่งกราฟแบบความหนาแน่นสมดุล (Density-balanced Partitioning Problem) โดยทำการกำหนดว่า ถ้ามีกราฟระบุทิศทาง (Directed Graph) G ซึ่งประกอบไปด้วยเซตของโหนด V และเซตของเอดจ์ E หรือ $G = (V, E)$ แล้ว ความหนาแน่นของกราฟ (Graph Density) [7] ของกราฟ G ใด ๆ สามารถหาจากสัดส่วนของขนาดของ E และขนาดของ V ซึ่งสามารถเขียนเป็นสมการได้ดังนี้

$$d(G) = \frac{|E|}{|V|} \quad (1)$$

ดังนั้นจะได้ว่าปัญหาที่ต้องการศึกษา คือการแบ่งกราฟ (Graph Partitioning) ให้ได้กราฟย่อยเรียกว่า $G_i = (V_i, E_i)$ โดย $i \in I$, $i > 0$ และ k เป็นจำนวนกราฟย่อยทั้งหมด ซึ่งมีนิยามตามเงื่อนไขต่อไปนี้

$$(U_1^k V_i = V) \wedge (N_1^k V_i \subseteq V) \quad (2)$$

$$(U_1^k E_i \subseteq E) \quad (3)$$

จากเงื่อนไข (2) จะได้ว่าถ้ายูเนียนของเซต $V_i ; i \in I, i > 0$ จะเท่ากับ V ในกรณีที่อินเตอร์เซกชันของเซต V_i ทั้งหมดอาจจะเป็นเซตย่อยของ V หรือเท่ากับ V ก็ได้ ในทางตรงกันข้าม จากเงื่อนไข (3) ยูเนียนของเอดจ์ทั้งหมด จะเป็นซับเซตหรือเท่ากับ E โดยในกรณีที่ยูเนียนออกมาเป็นเซตย่อยก็หมายความว่า มีเอดจ์บางส่วนที่ไม่ได้อยู่ในกราฟย่อย แต่เป็นเอดจ์ที่เชื่อมระหว่างกราฟย่อยเหล่านั้น ซึ่งเรียกเซตของเอดจ์เหล่านี้ว่า เซตตัด (Cut Set)

ในการแบ่งกราฟ G ออกเป็นกราฟย่อยนั้น สามารถแบ่งกราฟย่อยได้หลายลักษณะ โดยในงานวิจัยนี้ลักษณะของการแบ่งกราฟย่อยที่ต้องการคือ ลักษณะการแบ่งกราฟที่มีค่าความไม่สมดุลของความหนาแน่น (Density-unbalanced values) น้อย โดยค่าความไม่สมดุลของความหนาแน่น จะพิจารณาจากผลต่างของความหนาแน่นระหว่างกราฟย่อยที่มีความหนาแน่นมากที่สุดกับกราฟย่อยที่มีความหนาแน่นน้อยที่สุด ดังแสดงในสมการ (4)

$$\text{density} - \text{unbalanced values} = \max_{V_i \subseteq V} d(G_i) - \min_{V_i \subseteq V} d(G_i) \quad (3)$$

หรือสามารถเขียนเป็นปัญหาการตัดสินใจ (Decision Problem) ได้เป็น

$$\text{density} - \text{unbalanced values} \leq \alpha \quad (4)$$

นั่นคือเป็นไปได้หรือไม่ ที่จะหาลักษณะของการแบ่งกราฟย่อยที่มีค่าความไม่สมดุลของความหนาแน่นไม่มากกว่าค่าคงที่ (α) ค่าคงที่ค่าหนึ่ง โดยมีเงื่อนไขบังคับ (constraint) คือ มีขนาดของเซตของเอดจ์ที่เชื่อมระหว่างกราฟย่อยทั้งสองน้อยกว่าค่าคงที่ค่าหนึ่ง หรือ

$$|\{e | e \in E \wedge e \notin E' \wedge e \notin E''\}| \leq \beta \quad (5)$$

จากสมการ (6) e คือเอดจ์ที่อยู่ระหว่างกราฟย่อยทั้งสอง โดยขนาดของเซตของเอดจ์ที่มีลักษณะดังกล่าว จะต้องไม่มากกว่าค่าคงที่ (β) ที่กำหนดไว้

จากการทบทวนวรรณกรรมในงานวิจัยต่างๆ [8] ที่ได้วิจัยเกี่ยวกับการแบ่งกราฟให้มีความสมดุล ผู้วิจัยได้พบว่าในงานของ Andreev และ Racke [9] ได้เสนอปัญหาการแบ่งกราฟแบบมีความสมดุลของขนาดกราฟย่อย (Size-balanced partitioning problem) โดยในปัญหานี้คือการแบ่งกราฟออกเป็นกราฟย่อยจำนวน k ตัว โดยแต่ละกราฟย่อยมีขนาดไม่เกิน $v \cdot \frac{|V|}{k}$ โดยในกรณีที่ $v \geq 2$ มีผู้ศึกษามาแล้วว่าสามารถแก้ปัญหาอย่างมีประสิทธิภาพด้วยขั้นตอนวิธีแบบประมาณ (Approximation Algorithm) [10] ด้วยเวลา $O(\log n)$ ดังนั้น Andreev และ Racke จึงได้พิจารณาเฉพาะกรณีที่ $v < 2$ โดยสนใจไปที่

ปัญหาการแบ่งกราฟย่อยภายใต้ค่าคงมี $\varepsilon \in (0,1)$ ปัญหานี้จะแบ่งแต่ละกราฟย่อยให้มีขนาดไม่เกิน $(1 + \varepsilon) \cdot \frac{|V_i|}{k}$ ซึ่งจะทำให้ได้ขนาดของกราฟย่อยที่ใกล้เคียงกัน ปัญหาของ Andreev และ Racke ได้รับการพิสูจน์แล้วว่าเป็นปัญหา NP-hard และสามารถแก้ปัญหาดังกล่าวได้อย่างมีประสิทธิภาพด้วยขั้นตอนวิธีแบบประมาณเรียกว่าขั้นตอนวิธีการแบ่งกราฟ $(k, 1 + \varepsilon) - \text{balanced}$

ในงานวิจัยนี้ผู้วิจัยได้ต่อยอดแนวคิดจากปัญหาของ Andreev และ Racke โดยเพิ่มการพิจารณาความหนาแน่นของกราฟย่อย และได้บัญญัติปัญหาใหม่ที่เรียกว่าปัญหาการแบ่งกราฟแบบความหนาแน่นสมดุล (Uniform Density Min-cut Graph Partition Problem) ซึ่งปัญหานี้ต้องการแบ่งกราฟใหญ่ออกเป็น k กราฟย่อยที่มีขนาดของกราฟย่อย โดยแต่ละกราฟย่อยมีขนาดไม่เกิน $(1 + \varepsilon) \cdot \frac{|V_i|}{k}$ และมีค่าความไม่สมดุลของความหนาแน่นในแต่ละลักษณะของการแบ่งกราฟย่อยไม่มากกว่าค่าคงที่ α

ปัญหา UDMGP เป็นปัญหาใหม่ที่ยังไม่มีการศึกษา โดยตัวปัญหาจะใกล้เคียงกับปัญหาการแบ่งกราฟย่อยแบบหนาแน่น (Dense Partition Graph) หรือเรียกโดยย่อว่า DGP แต่ปัญหาดังกล่าวพิจารณาความหนาแน่นของกราฟเพียงกราฟเดียว แต่ในปัญหา UDMGP ที่นำเสนอใหม่นี้ จะพิจารณาความหนาแน่นของกราฟย่อยสองกราฟที่แยกมาจากกราฟหลักเพียงกราฟเดียว ซึ่งปัญหา DGP เป็นปัญหาที่พิสูจน์แล้วว่าเป็นปัญหาที่มีลักษณะเป็น NP-Complete

สำหรับความซับซ้อนทางการคำนวณ สามารถพิจารณาได้จากปัญหาการ DGP ซึ่งมีนิยามว่า ถ้ามีกราฟ $G = (V, E)$ และ เลขจำนวนจริง D ใด ปัญหาการตัดสินใจของ DGP ก็คือ มีกราฟย่อย P ใด ๆ ใน G ที่มีความหนาแน่น $d(P)$ ไม่น้อยกว่า D หรือไม่ โดย $d(P)$ นิยามแบบเดียวกับในสมการ (1) ซึ่งปัญหา DGP ได้รับการพิสูจน์แล้วว่าเป็นปัญหาในกลุ่ม NP-complete โดยการลดรูปไปยังปัญหา Graph- k -colorability ที่เป็นปัญหา NP-complete เช่นเดียวกัน สำหรับความซับซ้อนทางการคำนวณของปัญหา UDMGP สามารถพิจารณาได้จากการลดรูปไปยังปัญหา DGP ตามขั้นตอนต่อไปนี้

ให้ $G=(V,E)$ เป็นกราฟใด ๆ โดย $G'=(V',E')$ และ $G''=(V'',E'')$ เป็นกราฟย่อยของ G ตามนิยามในสมการ (1) และ (2) จะได้ว่า ถ้าให้ D เป็นเลขจำนวนจริงใด โดยที่ $d(G') \geq D$ และ $d(G'') \geq D$ ด้วย ดังนั้น จะได้ว่า

$$d(G') = D + \alpha' \quad (6)$$

$$d(G'') = D + \alpha'' \quad (7)$$

เมื่อ α' และ α'' คือจำนวนจริงที่มากกว่าศูนย์ใด ๆ ดังนั้น เมื่อนำสมการ (7) และ (8) มาหาค่าความแตกต่างกัน จะได้ว่า

$$|d(G') - d(G'')| = |\alpha' - \alpha''| \quad (8)$$

ซึ่งถ้าให้ $\alpha \geq |\alpha' - \alpha''|$ เมื่อ α คือจำนวนจริงใด ๆ จะได้ว่า

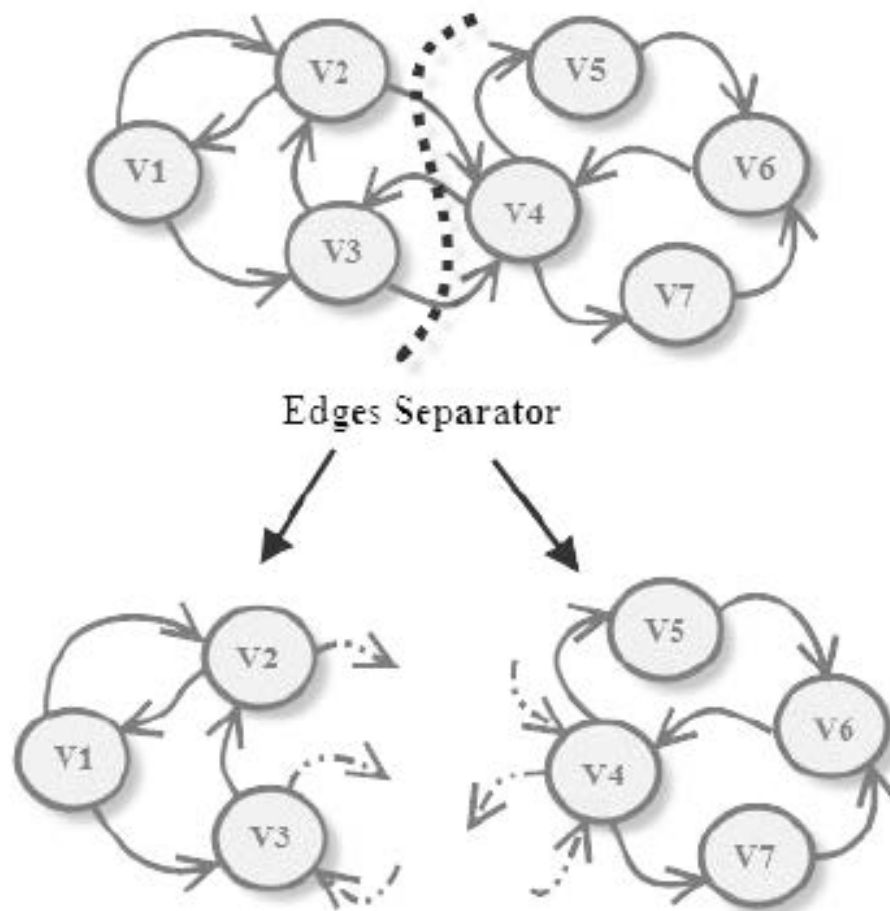
$$|d(G') - d(G'')| \leq \alpha' \quad (9)$$

ดังนั้น ถ้าสามารถหาคำตอบสำหรับปัญหา UDMGP โดยการหากราฟย่อย G' และ G'' ที่มีสมบัติตามสมการ (10) ได้ ก็จะสามารถหาค่า α' และ α'' ได้ ๆ ที่มีสมบัติตามสมการ (9) ซึ่งเมื่อแยกสมการ (9) ออกมาเป็นสมการ (7) และ (8) จะได้ว่า G' และ G'' เป็นคำตอบของปัญหา DGP จำนวนสองปัญหา (Instance) นั่นคือ ถ้าสามารถแก้ปัญห UDMGP ได้ ก็จะสามารถแก้ปัญห DGP ได้เช่นเดียวกัน ซึ่งแสดงว่า ปัญห UDMGP มีความยากไม่น้อยกว่าปัญห GDP และหมายความว่า ปัญห UDMGP เป็นปัญหาในกลุ่ม NP-Hard เช่นเดียวกัน และเนื่องจากปัญห UDMGP เป็นปัญหาเชิงการจัด (Combinatorial Problem) ซึ่งความซับซ้อนของปัญหจะเพิ่มขึ้นแบบฟังก์ชันเลขชี้กำลังตามขนาดของกราฟ แต่การพิสูจน์คำตอบของปัญห UDMGP ซึ่งก็คือการหาค่าความหนาแน่นของกราฟ สามารถทำได้ตามฟังก์ชันเส้นตรงตามขนาดของกราฟ ทำให้ปัญห UDMGP เป็นปัญหาในกลุ่ม NP และโดยสรุปปัญห UDMGP จึงเป็นปัญหาในกลุ่ม NP-complete นั่นเอง

แนวคิดแก้ปัญหา

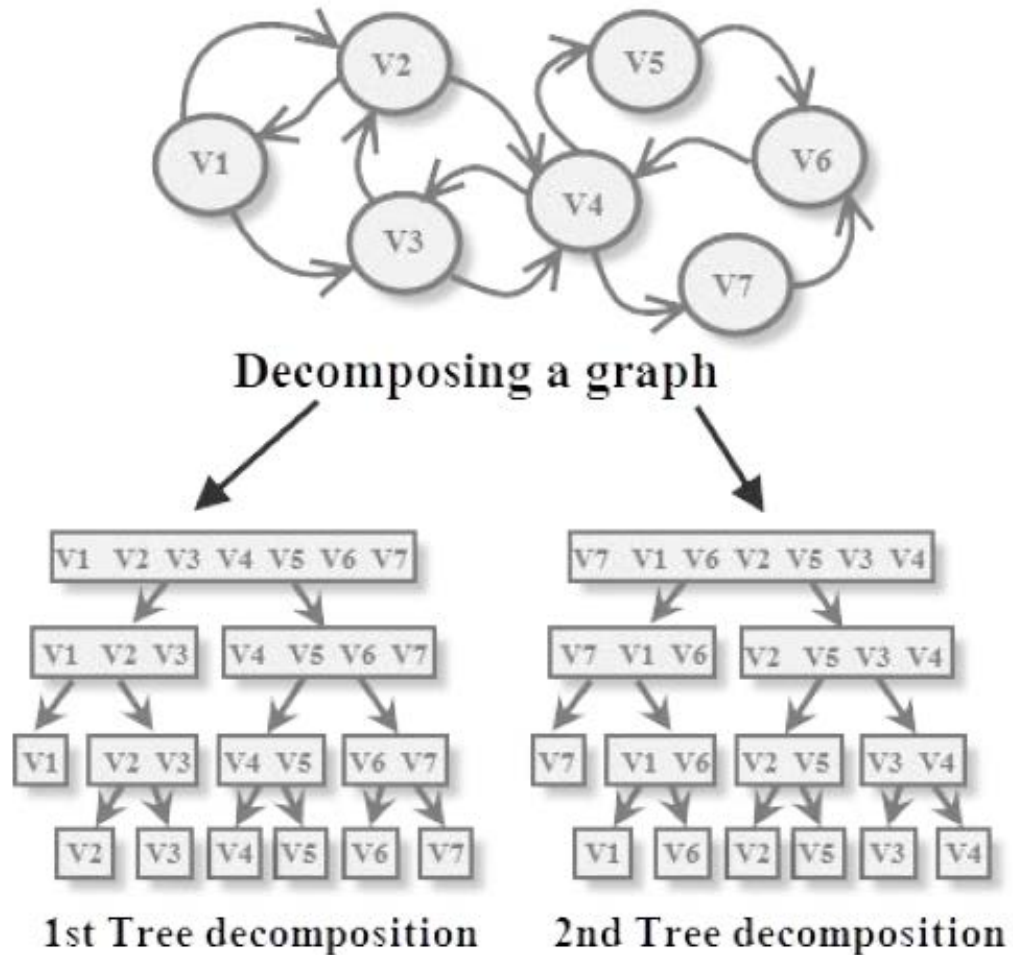
จากการบัญญัติปัญหาดังที่กล่าวมาข้างต้น ผู้วิจัยได้เสนอขั้นตอนวิธีแบ่งกราฟ $(k, 1 + \varepsilon, \alpha) - \text{balanced}$ หรือขั้นตอนวิธี DBP (Density-balanced Partitioning) เข้ามาแก้ปัญหการแบ่งกราฟแบบความหนาแน่นสมดุล โดยอาศัยหลักการของการย่อยสลายกราฟ (Decomposed graph) ด้วยวิธีการแบ่งครึ่งกราฟภายใต้เงื่อนไขที่กำหนดให้เซตของเอ็ดจ์ที่เชื่อมระหว่างกราฟย่อยทั้งน้อยกว่าค่าคงที่ (β) ที่กำหนดไว้ (ดูสมการ 6)

วิธีการตัดกราฟทีละครึ่งภายใต้เงื่อนไขที่กำหนดให้เซตของเอ็ดจ์ที่เชื่อมระหว่างกราฟย่อยทั้งน้อยกว่าค่าคงที่ค่าหนึ่งซึ่งอ้างตามสมการ (6) จะตัดกราฟ $G = (V, E)$ ให้เป็น $G1 = (V1, E1)$ และ $G2 = (V2, E2)$ ดังแสดงตัวอย่างของการตัดกราฟตามเงื่อนไขของสมการที่ 6 ในภาพที่ 1



ภาพที่ 1 ตัวอย่าง Edges separator

วิธีการย่อยสลายกราฟเป็นการเปลี่ยนโครงสร้างแบบกราฟให้อยู่ในรูปของโครงสร้างแบบต้นไม้ เพื่อลดความซับซ้อนลง การย่อยสลายกราฟจะได้โครงสร้างต้นไม้ของการย่อยสลาย (Tree decomposition) ซึ่งโครงสร้างต้นไม้ของการย่อยสลายมีได้หลายแบบขึ้นอยู่กับขั้นตอนวิธีการย่อยสลาย ในงานวิจัยนี้จะใช้ขั้นตอนการย่อยสลายด้วยการตัดกราฟทีละครั้งภายใต้เงื่อนไขที่กำหนดให้เซตของเอดจ์ที่เชื่อมระหว่างกราฟย่อยทั้งน้อยกว่าค่าคงที่ค่าหนึ่ง ซึ่งจะทำให้ได้โครงสร้างต้นไม้ของการย่อยสลายแบบทวิภาค (Binary tree decomposition) ดังแสดงตัวอย่างในภาพที่ 2



ภาพที่ 2 ตัวอย่าง Binary tree decomposition

การย่อยสลายแบบทวิภาคจากกราฟหนึ่ง จะสร้างเป็นโครงสร้างต้นไม้ได้หลายแบบ ซึ่งแต่ละแบบของโครงสร้างต้นไม้ของการย่อยสลายแบบทวิภาคถือเป็นแบ่งกราฟ 1 ลักษณะ โดยที่โหนดของโครงสร้างต้นไม้ของการย่อยสลายแบบทวิภาคจะแทนกราฟย่อย ดังตัวอย่างในภาพที่ 2 สำหรับกราฟใหญ่ $G = (V, E)$ ซึ่งแสดงตัวอย่างการย่อยสลายกราฟได้เป็นโครงสร้างต้นไม้สองแบบที่แตกต่างกัน โดยในโครงสร้างต้นไม้ของการย่อยสลายแบบทวิภาคแบบที่ 1 ดัวซ้ายมือ จะแบ่งกราฟที่ประกอบไปด้วยโหนดทั้งหมดเจ็ดโหนด ได้เป็นกราฟย่อยสองกราฟ กราฟย่อยแรกมีสามโหนดคือ $\{v1, v2, v3\}$ ส่วนกราฟย่อยที่สองมีสี่โหนดคือ $\{v4, v5, v6\}$ ในขณะที่โครงสร้างต้นไม้จากการย่อยสลายแบบทวิภาคแบบที่ 2 ด้านขวามือก็แบ่งกราฟออกเป็นกราฟย่อยสองกราฟเช่นเดียวกัน แต่กราฟย่อยแรกมีโหนดคือ $\{v7, v1, v6\}$ ในขณะที่กราฟย่อยที่สองมีโหนดคือ $\{v2, v5, v3, v4\}$

จากที่กล่าวมาข้างต้นการหาลักษณะของการแบ่งกราฟด้วยวิธีการย่อยสลายกราฟทำให้ได้ลักษณะการแบ่งกราฟย่อยจำนวนมาก โดยแต่ละลักษณะของการแบ่งกราฟย่อยอาจนำไปใช้งานจริงไม่ได้เนื่องจากค่าความไม่สมดุลของความหนาแน่นเกินกว่าค่าคงที่ α หรืออาจไม่มีความสมดุลของขนาดกราฟ

ย่อย ดังนั้นเมื่อย่อยสลายกราฟแล้วจึงต้องนำโครงสร้างต้นไม้ของการย่อยสลายแบบทวิภาคทั้งหมดมากรองด้วยค่าคงที่ α เพื่อไม่ให้ค่าความไม่สมดุลของความหนาแน่นเกินกว่าค่าที่กำหนด และกรองด้วยค่าคงที่ ϵ เพื่อไม่ให้ขนาดของกราฟย่อยเกินกว่าค่าที่กำหนด โดยขั้นตอนวิธีการทำงานแสดงตามภาพที่ 3

จากภาพที่ 3 ฟังก์ชัน BTD จะทำการสร้างโครงสร้างต้นไม้ของการย่อยสลายแบบทวิภาคที่เป็นไปได้ทั้งหมด ใส่ไว้ในเซต TreeList ในบรรทัดที่ 3 แล้วจึงนำเอาโครงสร้างต้นไม้แต่ละอันมาพิจารณา โดยในบรรทัดที่ 6-10 เป็นการตรวจสอบว่าขนาดของกราฟย่อยในแต่ละโหนดของโครงสร้างต้นไม้ที่กำลังพิจารณามีขนาดเกิน เงื่อนไขที่ระบุด้วยค่าคงที่ ϵ หรือไม่ ส่วนในบรรทัดที่ 12-16 เป็นการคำนวณหาค่าความแตกต่างสูงสุดระหว่างความหนาแน่นของกราฟย่อย ซึ่งถ้าโครงสร้างต้นไม้ได้มีค่าความแตกต่างของความหนาแน่นน้อยกว่าค่าคงที่ α และมีจำนวนกราฟย่อยเท่ากับ k ก็จะบรรจุโครงสร้างต้นไม้ไว้ในรายการการแบ่งกราฟย่อยที่ใช้งานได้

Algorithm 1 DBP

Input: Graph $G = (V, E)$, number of component k , size unbalanced constrain ε , density unbalanced constrain α .

Output: The density balanced partitions of G .

```

1:  $TreeList \leftarrow \emptyset$ 
2:  $PartitionList \leftarrow \emptyset$ 
3:  $TreeList \leftarrow BT D(G)$ 
4: for each  $T_i \in TreeList$  do
5:    $v_{T_i} \leftarrow V(T_i)$ 
6:   for each  $node_j \in v_{T_i}$  do
7:     if  $|node_j| \geq (1 + \varepsilon) \frac{|V(G)|}{k}$  then
8:        $v_{T_i} \leftarrow v_{T_i} \setminus node_j$ 
9:     end if
10:  end for
11:   $md \leftarrow 0$ 
12:  for each  $node_n \in v_{T_i}$  do
13:    for each  $node_m \in v_{T_i}, node_m \neq node_n$  do
14:       $md \leftarrow \max(md, |d(node_m) - d(node_n)|)$ 
15:    end for
16:  end for
17:  if  $|v_{T_i}| > 0 \wedge md \leq \alpha \wedge component(v_{T_i}) = k$  then
18:     $PartitionList \leftarrow PartitionList \cup \{T_i\}$ 
19:  end if
20: end for
21: return  $PartitionList$ 

```

5. ขอบเขตของการวิจัย

ตัวแปรต้น/ตัวแปรตาม

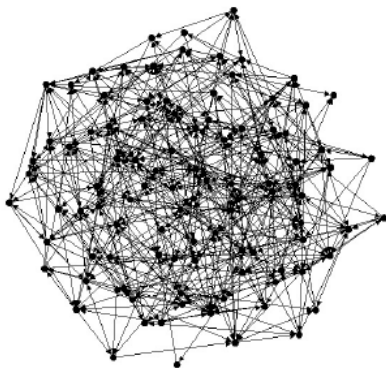
- ช่วงของค่าคงที่ α ที่มีผลกับค่าความไม่สมดุลของความหนาแน่นของการแบ่งกราฟในแต่ละลักษณะ และจำนวนลักษณะการแบ่งกราฟย่อยที่นำไปใช้งานได้
- ค่าความไม่สมดุลของความหนาแน่นของการแบ่งกราฟในแต่ละลักษณะ ที่ได้จากการแบ่งกราฟด้วยขั้นตอนวิธีแบ่งกราฟหลายๆขั้นตอนวิธี

ตัวแปรควบคุม

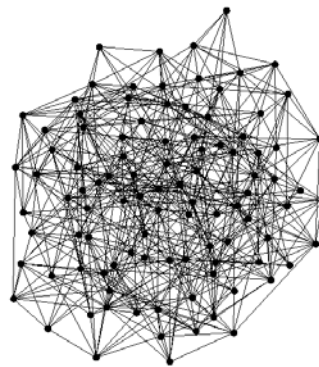
- ในแต่ละลักษณะของการแบ่งกราฟกำหนดให้กราฟย่อยมีแค่ 4 กราฟเท่านั้น
- กำหนดค่าคงที่ ϵ เป็น 0.5
- กำหนดค่าคงที่ β เป็น 20% ของจำนวนเอดจ์ทั้งหมด

6. ระเบียบวิธีวิจัย

แบ่งการทดลองออกเป็น 2 ชุดทดลองคือ ชุดทดลองที่ 1 ทดลองกับโครงสร้างกราฟแบบสุ่ม (Random graph) ดังแสดงตัวอย่างในภาพที่ 4 ชุดทดลองที่ 2 ทดลองกับโครงสร้างกราฟแบบสุ่มชนิดพิเศษที่จำลองความสัมพันธ์ของวัตถุในเครือข่ายสังคมตามแบบจำลองของ Watts และ Strogatz (Watts-Strogatz Model) [11] ดังแสดงในภาพที่ 5



ภาพที่ 4 ตัวอย่าง Random graph



ภาพที่ 5 ตัวอย่าง Watts-Strogatz graph

โดยทั้ง 2 ชุดทดลองจะแบ่งการทดลองย่อยออกเป็น 2 ส่วน ในส่วนแรกจะเป็นการหาผลกระทบของค่าคงที่ α ที่มีผลกับค่าความไม่สมดุลของความหนาแน่นในแต่ละลักษณะของการแบ่งกราฟย่อย และหาผลกระทบของค่าคงที่ α ที่มีผลกับจำนวนลักษณะของการแบ่งกราฟย่อยที่นำไปใช้งานได้ ในส่วนที่สองจะเป็นการเปรียบเทียบประสิทธิภาพของขั้นตอนวิธี DBP ในภาพที่ 3 กับขั้นตอนวิธี ABP ที่ถูกเสนอโดย

Andreev และ Racke และเทียบกับขั้นตอนวิธี GBP ซึ่งเป็นขั้นตอนวิธีแบบละโมบ (Greedy Algorithm) ที่ใช้ในงานวิจัยก่อนหน้านี้ของผู้วิจัย [6]

ชุดทดลองที่ 1 Random Graph

ในส่วนแรกจะเป็นการหาผลกระทบของค่าคงที่ α ที่มีผลกับค่าความไม่สมดุลของความหนาแน่นในแต่ละลักษณะของการแบ่งกราฟย่อย และหาผลกระทบของค่าคงที่ α ที่มีผลกับจำนวนลักษณะของการแบ่งกราฟย่อยที่นำไปใช้งานได้ ซึ่งในการทดลองส่วนแรกจะกำหนดขนาดของกราฟใหญ่อยู่ที่ 1000 โหนด จำนวนเอ็ดจ์เฉลี่ยต่อโหนดอยู่ที่ 10 เอ็ดจ์ รวมทั้งกำหนด ϵ ที่ 0.5 และจำนวนกราฟย่อยในการแบ่งแต่ละลักษณะมีเพียง 4 กราฟย่อยเท่านั้น โดยให้ค่า α อยู่ในช่วง 0 - 0.85 สังเกตค่าความไม่สมดุลของความหนาแน่นในแต่ละลักษณะของการแบ่งกราฟย่อย และสังเกตจำนวนลักษณะของการแบ่งกราฟย่อยที่นำไปใช้งานได้ ในแต่ละค่าของ α

ในส่วนที่สองจะเป็นการเปรียบเทียบประสิทธิภาพของขั้นตอนวิธี DBP กับขั้นตอนวิธี ABP และ GBP โดยกำหนดให้ขนาดของกราฟใหญ่อยู่ในช่วง 1,000 - 1,800 โหนด จำนวนเอ็ดจ์เฉลี่ยต่อโหนดอยู่ที่ 10 เอ็ดจ์ ใช้ α ที่ 0.15 และ ϵ ที่ 0.5 โดยที่จำนวนกราฟย่อยในการแบ่งแต่ละลักษณะมีเพียง 4 กราฟย่อยเท่านั้น สังเกตค่าความไม่สมดุลของความหนาแน่นในแต่ละลักษณะของการแบ่งกราฟย่อย ที่ทำการแบ่งกราฟย่อยด้วย 3 ขั้นตอนวิธีที่กล่าวมา

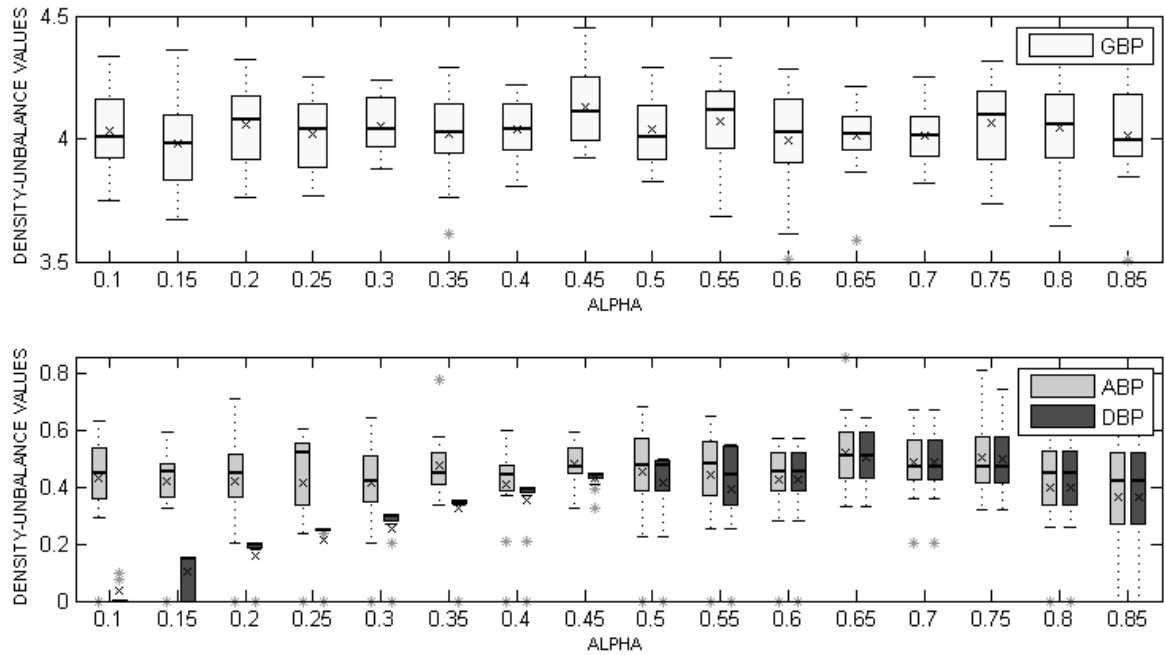
ชุดทดลองที่ 2 Watts-Strogatz Graph

ในส่วนแรกจะเป็นการหาผลกระทบของค่าคงที่ α ที่มีผลกับค่าความไม่สมดุลของความหนาแน่นในแต่ละลักษณะของการแบ่งกราฟย่อย และหาผลกระทบของค่าคงที่ α ที่มีผลกับจำนวนลักษณะของการแบ่งกราฟย่อยที่นำไปใช้งานได้ ซึ่งในการทดลองส่วนแรกจะกำหนดขนาดของกราฟใหญ่อยู่ที่ 1000 โหนด จำนวนเอ็ดจ์เฉลี่ยต่อโหนดอยู่ที่ 4 เอ็ดจ์ รวมทั้งกำหนด ϵ ที่ 0.5 และจำนวนกราฟย่อยในการแบ่งแต่ละลักษณะมีเพียง 4 กราฟย่อยเท่านั้น โดยให้ค่า α อยู่ในช่วง 0 - 0.85 สังเกตค่าความไม่สมดุลของความหนาแน่นในแต่ละลักษณะของการแบ่งกราฟย่อย และสังเกตจำนวนลักษณะของการแบ่งกราฟย่อยที่นำไปใช้งานได้ ในแต่ละค่าของ α

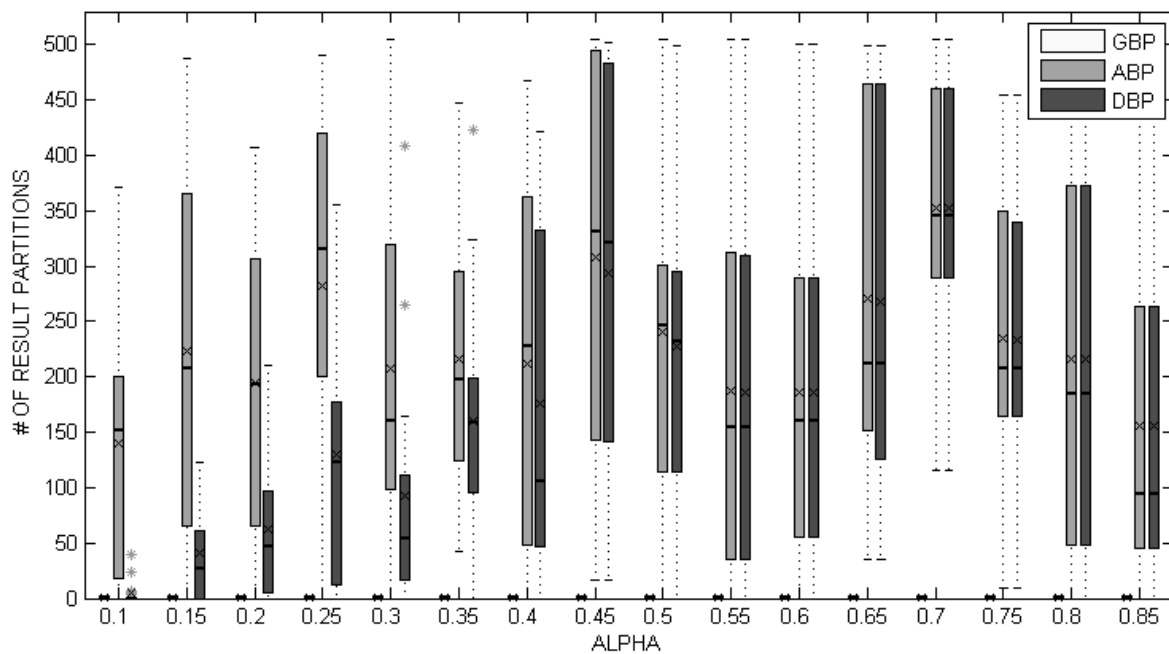
ในส่วนที่สองจะเป็นการเปรียบเทียบประสิทธิภาพของขั้นตอนวิธี DBP กับขั้นตอนวิธี ABP และ DBP โดยกำหนดให้ขนาดของกราฟใหญ่อยู่ในช่วง 1,000 - 1,800 โหนด จำนวนเอ็ดจ์เฉลี่ยต่อโหนดอยู่ที่ 4 เอ็ดจ์ ใช้ α ที่ 0.15 และ ϵ ที่ 0.5 โดยที่จำนวนกราฟย่อยในการแบ่งแต่ละลักษณะมีเพียง 4 กราฟย่อยเท่านั้น สังเกตค่าความไม่สมดุลของความหนาแน่นในแต่ละลักษณะของการแบ่งกราฟย่อย ที่ทำการแบ่งกราฟย่อยด้วย 3 ขั้นตอนวิธีที่กล่าวมา

7. ผลการวิจัย

ผลการทดลองที่ 1 Random Graph



ภาพที่ 6 แสดงผลของค่าคงที่ α ที่มีผลกับค่าความไม่สมดุลของความหนาแน่นในการแบ่ง Random graph



ภาพที่ 7 แสดงผลของค่าคงที่ α ที่มีผลกับจำนวนการแบ่งกราฟย่อยที่ใช้งานได้ใน การแบ่ง Random graph

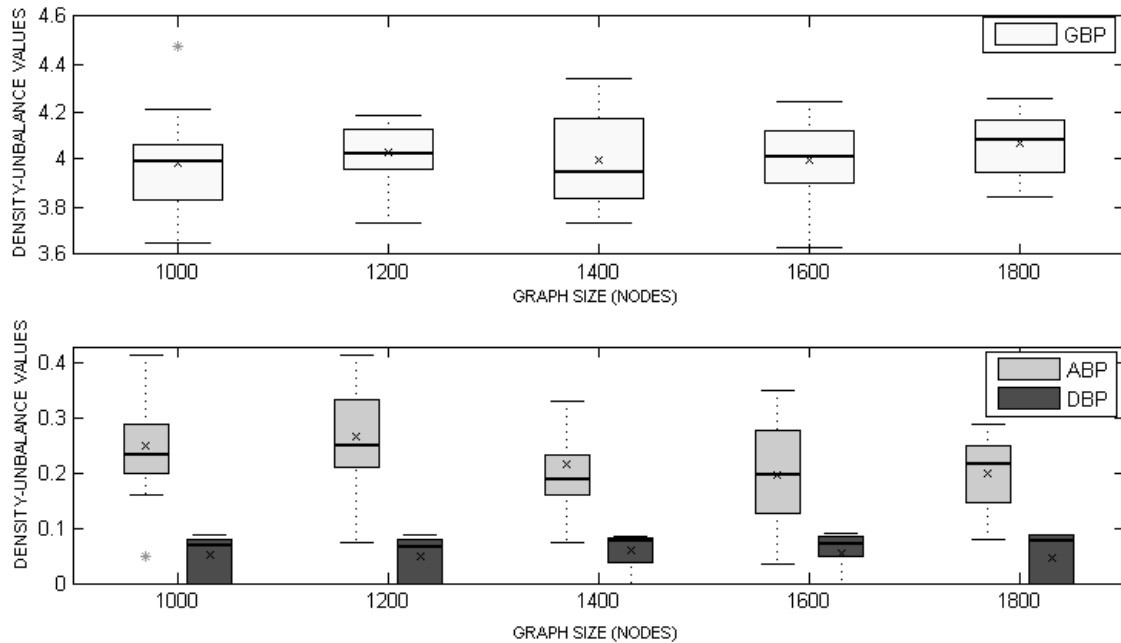
ภาพที่ 6 เป็นกราฟที่แสดงผลกระทบของค่าคงที่ α ที่มีผลกับค่าความไม่สมดุลของความหนาแน่น ในกราฟนี้เป็นการสังเกตค่า α ที่อยู่ระหว่างช่วง 0.1 – 0.85 ซึ่งแสดงตามแกน x และค่าความไม่สมดุลของความหนาแน่นแสดงตามแกน y เก็บผลการทดลอง 16 ครั้ง โดยใช้ Random graph ขนาด 1000 โหนด จะเห็นได้ว่าค่าคงที่ α ที่มีขนาดเล็กจะส่งผลให้ค่าความไม่สมดุลของความหนาแน่นน้อยตามไปด้วย ขั้นตอนวิธี DBP สามารถแบ่งกราฟให้ค่าความไม่สมดุลของความหนาแน่นน้อยมากจนเกือบเป็น 0 ได้ เมื่อเทียบกับ ABP และ GBP ซึ่งจากภาพที่ 6 จะเห็นได้ว่าค่าความไม่สมดุลของความหนาแน่นที่ได้จากการแบ่งกราฟย่อยด้วยขั้นตอนวิธี DBP เฉลี่ยประมาณ 2 อย่างไรก็ตามหากเพิ่มค่าคงที่ ค่าความไม่สมดุลของความหนาแน่นจะเพิ่มขึ้นตามไปด้วย แต่จะไม่มากไปกว่าการแบ่งกราฟย่อยด้วยขั้นตอนวิธี ABP และ GBP

ภาพที่ 7 แสดงผลของค่าคงที่ α ที่มีผลกับจำนวนการแบ่งกราฟย่อยที่นำไปใช้งานได้ ซึ่งหมายถึงการแบ่งกราฟย่อยที่เป็นไปตามเงื่อนไขของค่าคงที่ ϵ และ α โดยในกราฟนี้เป็นการกำหนดค่า α ที่อยู่ระหว่างช่วง 0.1 – 0.85 ซึ่งแสดงตามแกน x และจำนวนลักษณะการแบ่งกราฟย่อย แสดงตามแกน y เก็บผลการทดลอง 16 ครั้ง โดยใช้ Random graph ขนาด 1000 โหนด จะเห็นได้ว่าจำนวนลักษณะการแบ่งกราฟย่อยที่นำไปใช้งานได้ ที่แบ่งด้วยขั้นตอนวิธี GBP จะได้เพียง 1 ลักษณะเสมอ (ดังแสดงเป็นเส้นประในภาพที่ 7 โดยกราฟของ GBP จะอยู่ด้านซ้ายมือของกราฟ ABP) การแบ่งกราฟย่อยด้วยขั้นตอนวิธี ABP จะได้จำนวนลักษณะการแบ่งกราฟย่อยที่นำไปใช้งานได้ มากกว่าหรือเท่ากับกับการแบ่งกราฟย่อยด้วยขั้นตอนวิธี DBP เสมอ ในทางตรงกันข้ามการแบ่งกราฟด้วยขั้นตอนวิธี DBP อาจทำให้ไม่มีลักษณะการแบ่งกราฟย่อยที่นำไปใช้งานได้ จากค่าที่ α ที่มีขนาดเล็กมากๆ ทั้งนี้เนื่องมาจากเงื่อนไขของความหนาแน่นสมดุลที่นำมาพิจารณาในขั้นตอนวิธี DBP แต่ในกรณีที่ α ไม่ได้มีขนาดเล็กมาก ขั้นตอนวิธี DBP ก็จะสามารถหาคำตอบได้อย่างน้อยหนึ่งคำตอบ ซึ่งก็เพียงพอต่อการใช้งานแล้ว ดังเช่นในภาพที่ 7 จะเห็นว่า DBP สามารถหาคำตอบได้ในทุกค่าของ α

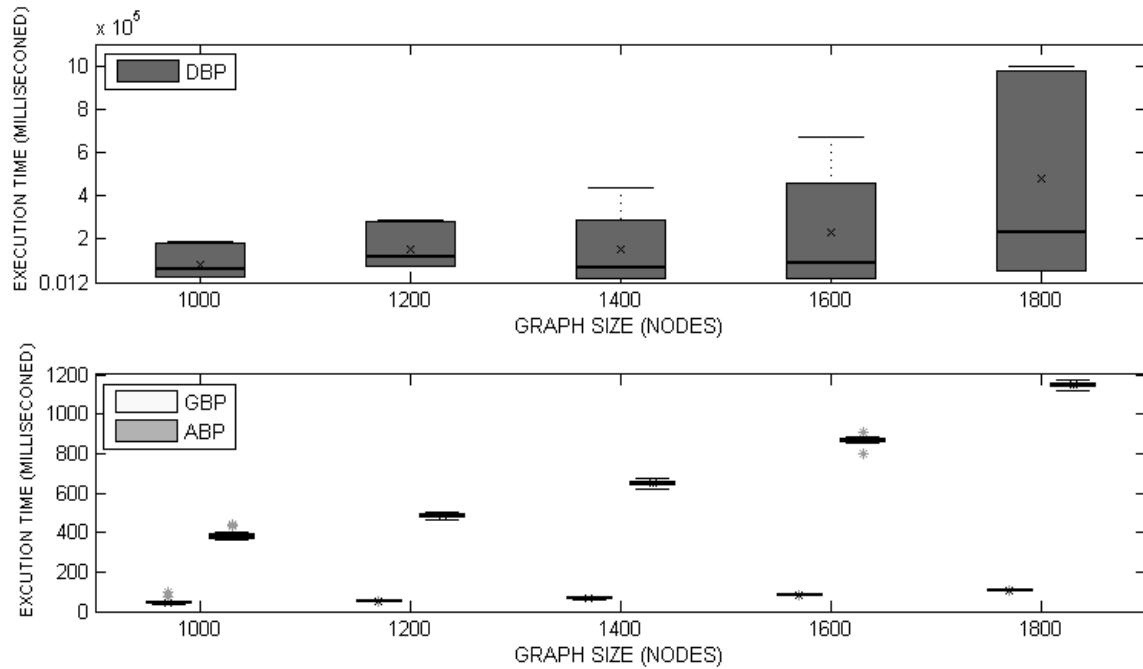
ภาพที่ 8 เป็นการทดสอบประสิทธิภาพของขั้นตอนวิธี DBP เทียบกับขั้นตอนวิธี ABP และ GBP โดยทดลอง 16 ครั้ง และใช้กราฟทดลองแบบสุ่มที่มีขนาด 1000 – 1800 โหนด หากพิจารณาในแง่ของการแบ่งกราฟแบบความหนาแน่นสมดุลแล้วจะเห็นว่าเมื่อนำขั้นตอนวิธี DBP มาแบ่งกราฟย่อยจะได้กราฟย่อยที่มีความหนาแน่นสมดุลมากกว่าขั้นตอนวิธี ABP และดีกว่าขั้นตอนวิธี GBP อย่างเห็นได้ชัดเจน แต่ในทางตรงกันข้ามในภาพที่ 9 เป็นการทดสอบ DBP เทียบกับขั้นตอนวิธี ABP และ GBP โดยทดลอง 16 ครั้ง และใช้กราฟทดลองแบบสุ่มที่มีขนาด 1000 – 1800 โหนด เช่นเดียวกับภาพที่ 8 แต่จะพิจารณาในแง่ของเวลาที่ใช้ในการทำงานจะเห็นได้ว่า DBP จะใช้เวลามากกว่า ABP และ GBP เนื่องจากขั้นตอนวิธี

DBP มีความซับซ้อนในเชิงคำนวณเป็น $O(n^3)$ แต่ในส่วนของการ Execution time นี้ไม่ได้มีผลกับความหนาแน่นสมดุลในการแบ่งกราฟย่อย และอยู่ในช่วงค่าที่สามารถยอมรับได้

ดังนั้นถ้าพิจารณาในแง่ของการแบ่งกราฟแบบความหนาแน่นสมดุลขั้นตอนวิธี DBP จะมีประสิทธิภาพในการแบ่งกราฟมากกว่าขั้นตอนวิธีอื่นๆ เนื่องจากขั้นตอนวิธี DBP พิจารณาค่าความไม่สมดุลของความหนาแน่นในการแบ่งกราฟเข้าไปด้วย ซึ่งการแบ่งกราฟแบบความหนาแน่นสมดุลนี้หากนำไปใช้ในการคำนวณพหุคูณระบบเครือข่ายแบบเพียร์ทูเพียร์จะทำให้จำนวนรอบของการคำนวณพหุคูณในแต่ละเพียร์ใช้เท่าๆกัน



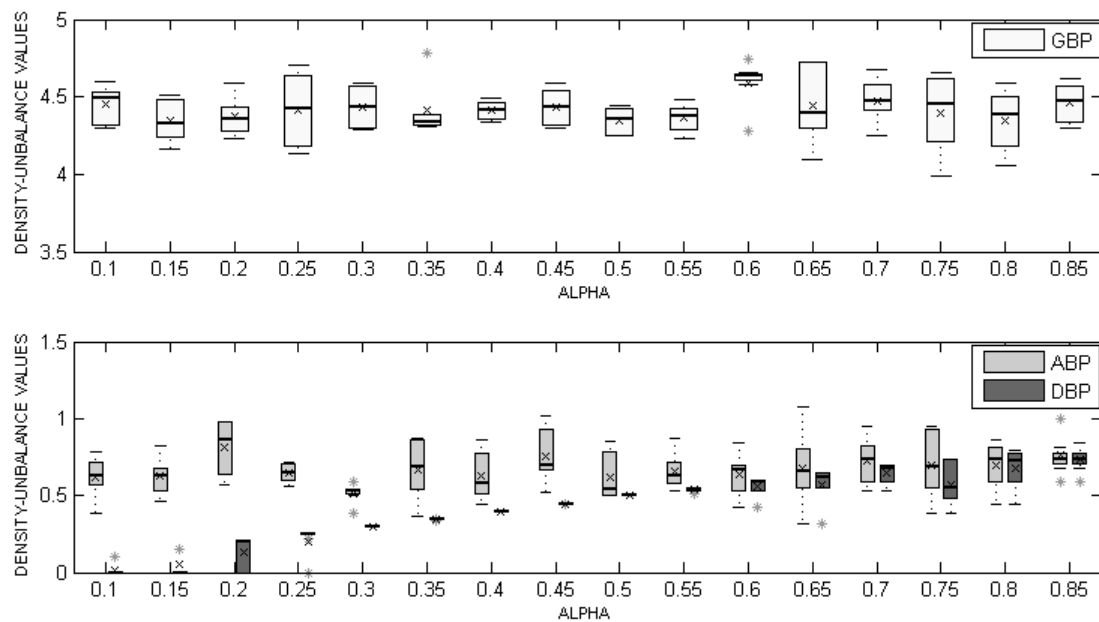
ภาพที่ 8 แสดงประสิทธิภาพของขั้นตอนวิธี DBP ในแง่ของการแบ่งกราฟแบบความหนาแน่นสมดุลในการแบ่ง Random graph



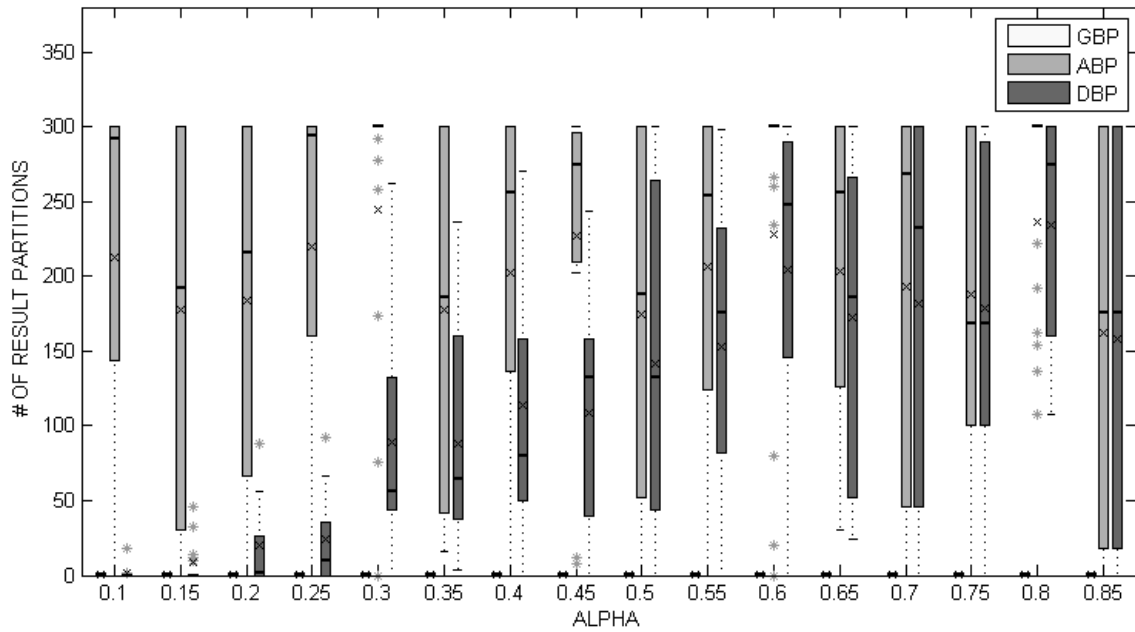
ภาพที่ 9 แสดงประสิทธิภาพของขั้นตอนวิธี DBP ในแง่ของ Execution time ในการแบ่ง Random graph

ผลการทดลองที่ 2 Watts-Strogatz Graph

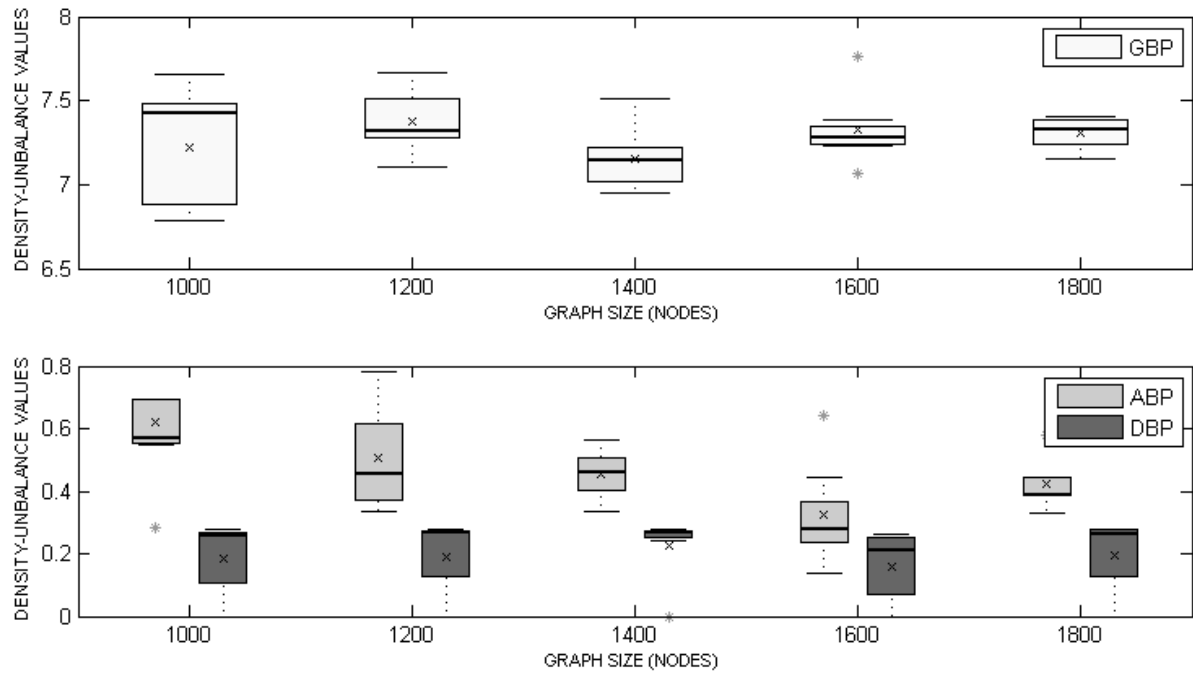
ภาพที่ 10, 11, 12, 13 เป็นการแบ่งกราฟที่สร้างโดยใช้แบบจำลอง Watts-Strogatz graph ซึ่งให้ผลการทดลองไปในทางเดียวกันกับการทดลองที่ 1



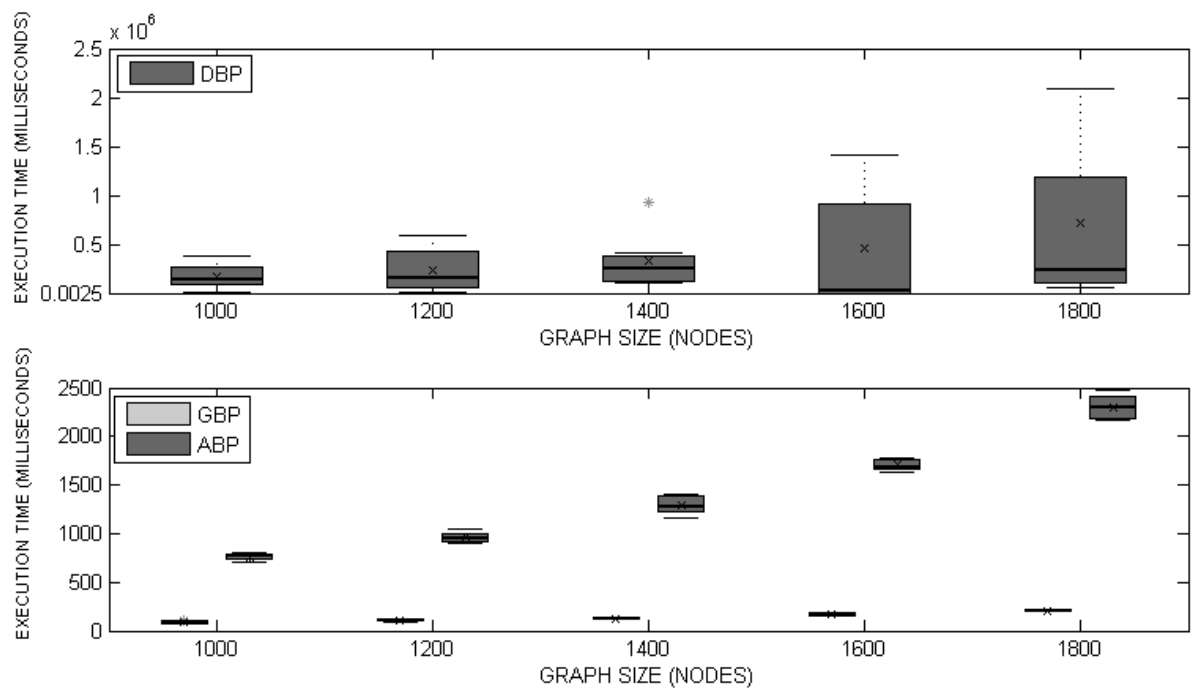
ภาพที่ 10 แสดงผลของค่าคงที่ α ที่มีผลกับค่าความไม่สมดุลของความหนาแน่น ในการแบ่ง Watts-Strogatz graph



ภาพที่ 11 แสดงผลของค่าคงที่ α ที่มีผลกับจำนวนการแบ่งกราฟย่อยที่ใช้งานได้ ในการแบ่ง Watts-Strogatz Graph



ภาพที่ 12 แสดงประสิทธิภาพของขั้นตอนวิธี DBP ในแง่ของการแบ่งกราฟแบบความหนาแน่นสมดุลในการแบ่ง *Watts-Strogatz graph*



ภาพที่ 13 แสดงประสิทธิภาพของขั้นตอนวิธี DBP ในแง่ของ Execution time ในการแบ่ง *Watts-Strogatz graph*

8. สรุปผล

จากการบัญญัติปัญหาในงานวิจัยนี้ ซึ่งเป็นปัญหาการแบ่งกราฟแบบความหนาแน่นสมดุล โดยแบ่งกราฟใหญ่ออกเป็น k กราฟย่อยที่มีขนาดของกราฟย่อยเกือบเท่าๆกัน โดยแต่ละกราฟย่อยมีขนาดไม่เกิน $(1 + \varepsilon) \cdot \frac{|V|}{k}$ และมีค่าความไม่สมดุลของความหนาแน่นในแต่ละลักษณะของการแบ่งกราฟย่อยไม่มากกว่าค่าคงที่ α ผู้วิจัยได้พัฒนาขั้นตอนวิธี $(k, 1 + \varepsilon, \alpha) - \text{balanced}$ หรือขั้นตอนวิธี DBP ในการแก้ปัญหา ซึ่งผลการทำงานของขั้นตอนวิธีดังกล่าว มีประสิทธิภาพเป็นอย่างมากในการแบ่งกราฟให้มีความหนาแน่นสมดุล เมื่อเทียบกับขั้นตอนวิธีการแบ่งกราฟแบบอื่นๆ

เมื่อเปรียบเทียบกับงานเดิม [9] งานวิจัยนี้ สามารถสร้างกราฟย่อยที่มีความสมดุลของความหนาแน่น (Density Balanced Partition) ได้ดีกว่า โดยต้องใช้เวลาในการประมวลผลมากกว่างานเดิม แต่ก็ยังถือว่าในระดับที่ยังยอมรับได้ (ความซับซ้อนในการคำนวณยังเป็นโพลิโนเมียล) งานวิจัยในอนาคตสำหรับโครงการนี้คือการปรับปรุงเวลาในการประมวลผล โดยจะปรับปรุงการย่อยสลายแบบทวิภาค ซึ่งในงานปัจจุบัน จะสร้างโครงสร้างต้นไม้ที่เป็นไปได้ทั้งหมด ซึ่งจะได้จำนวนของโครงสร้างต้นไม้จำนวนมาก เมื่อเทียบกับจำนวนโหนดในกราฟ ซึ่งผู้วิจัยมีแผนที่จะลดจำนวนโครงสร้างต้นไม้ที่ได้จากการย่อยสลายแบบทวิภาคโดยใช้วิธีการเลือกกลบบางส่วน (Pruning) เพื่อเลือกเอาโครงสร้างต้นไม้ที่มีลักษณะไม่เหมาะสมออกไป โดยผู้วิจัยมีแผนที่จะตีพิมพ์ผลการวิจัยอันนี้ในวารสารทางวิชาการต่อไป

งานวิจัยนี้ ได้รับการตอบรับให้ไปนำเสนอในงานประชุมวิชาการนานาชาติ The 9th International Conference on Digital Information Management (ตามจดหมายแนบ) โดยจดหมายเหตุการณ์ประชุม (Proceedings) ของการประชุมนี้ จะถูกตีพิมพ์โดยสำนักพิมพ์ IEEE และจะถูกจัดเก็บไว้ในฐานข้อมูล IEEE Xplore ฐานข้อมูล DBLP และ ฐานข้อมูล SCOPUS เป็นต้น

9. ปัญหาและอุปสรรค

มีปัญหาในเรื่องของ Execution Time ซึ่งจะปรับปรุงให้เร็วขึ้นในงานวิจัยต่อไป

10. บรรณานุกรม

- [1] S. Brin and L. Page, “The anatomy of a large-scale hypertextual web search engine,” *Computer Networks and ISDN Systems*, vol. 30, no. 1-7, pp. 107 – 117, 1998.
- [2] P. Berkhin, “A survey on pagerank computing,” *Internet Mathematics*, vol. 2, pp. 73–120, 2005.
- [3] L. Page, S. Brin, R. Motwani, and T. Winograd, “The pagerank citation ranking: Bringing order to the web,” *Stanford University, Technical Report*, 1998.
- [4] A. N. Langville and C. D. Meyer, “Deeper inside pagerank,” *Internet Mathematics*, vol. 1, p. 2004, 2004.
- [5] J. X. Parreira and G. Weikum, “JXP global authority scores in a P2P network,” in *Proceedings of the Eight International Work-shop on the Web and Databases (WebDB 2005)*, Baltimore, Maryland, USA, 2005, pp. 31–36.
- [6] S. Sangamuang, P. Boonma, and J. Natwichai, “A P2P-based incremental web ranking algorithm,” in *Proceedings of the 2011 International Conference on P2P, Parallel, Grid, Cloud and Internet Computing*, Oct 2011, pp. 123–127.
- [7] J. Darlay, N. Brauner, and J. Moncel, “Dense and Sparse Graph Partition,” *Discrete Appl. Math.*, vol. 160, no. 16–17, pp. 2389–2396, Nov. 2012.
- [8] M. Charikar, “Greedy Approximation Algorithms for Finding Dense Components in a Graph,” in *Proceedings of the Third International Workshop on Approximation Algorithms for Combinatorial Optimization*, London, UK, UK, 2000, pp. 84–95.
- [9] K. Andreev and H. Racke, “Balanced graph partitioning,” *Theory of Computing Systems*, vol. 39, pp. 929–939, November 2006.
- [10] G. Even, J. S. Naor, S. Rao, and B. Schieber, “Fast approximate graph partitioning algorithms,” in *Proceedings of the Eighth Annual ACMSIAM Symposium on Discrete Algorithms*. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics, 1997, pp. 639–648.

[11] Duncan J. Watts and Steven H. Strogatz. "Collective dynamics of 'small-world' networks". *Nature*, 393:440-442, 1998.

11. ภาคผนวก

ประวัติผู้วิจัย

ชื่อ ดร. พฤษภ บัญมา

ตำแหน่ง อาจารย์ประจำภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์
มหาวิทยาลัยเชียงใหม่

การศึกษา ปริญญาเอกด้านวิทยาการคอมพิวเตอร์จาก University of Massachusetts,
Boston ด้วยทุนกระทรวงวิทยาศาสตร์และเทคโนโลยี

ที่อยู่ ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเชียงใหม่ 50200