# บทที่ 4
# การทดลองและผลการทดลอง

งานวิจัยนี้เป็นการนำเสนอการเพิ่มสมรรถนะ (Performance) โดยการลดเวลาที่ใช้ในการติดต่อสื่อสารระหว่างหน่วยประมวลของการบีบอัดข้อมูลแบบขนานด้วยวิธีบีซิพทู และลดการใช้งานอุปกรณ์ อินพุทเอาท์พุทศูนย์กลาง (Centralized I/O) พร้อมกันของแต่ละหน่วยประมวลผล โดยวิธีต่างๆ 3 วิธี (MSBzip2, AWBzip2 และ AWWBzip2) ที่เสนอได้แสดงขั้นตอนวิธีไว้แล้วในบทที่ 3 ส่วนเนื้อหาในบทนี้ จะเป็นการเสนอผลการพัฒนาโปรแกรมการบีบอัดข้อมูลแบบขนานดังกล่าวเพื่อประมวลผลบนระบบพีซีคลัสเตอร์ ซึ่งเป็นระบบที่ได้รับความสนใจเป็นอย่างมากในปัจจุบัน เนื่องจากมีราคาไม่แพง และมีขั้นตอนการพัฒนาที่ไม่ยุ่งยากซับซ้อนมาก

## 4.1 ระบบคอมพิวเตอร์คลัสเตอร์ที่ใช้ในการทดลอง

ระบบคลัสเตอร์ที่ใช้ในการทดลอง เป็นระบบคลัสเตอร์แบบ “Homogenous Cluster” ที่มีส่วนประกอบ ประสิทธิภาพในการประมวลผล และภาระงานที่รับผิดชอบในแต่ละเครื่องมีจำนวนเท่ากัน นอกจากนั้นในขณะทำการทดลองระบบคลัสเตอร์ดังกล่าวจะไม่มีภาระงานอื่นใดประมวลผลอยู่ในระบบคลัสเตอร์ที่ใช้ในการทดลอง มีส่วนประกอบพื้นฐานของระบบซึ่งจะประกอบด้วยส่วนประกอบด้านฮาร์ดแวร์ ระบบปฏิบัติการและซอฟท์แวร์ ที่ทำงานในระดับของแกน (Kernel) ของระบบปฏิบัติการซึ่งเรียกซอฟท์แวร์นี้ว่า “คลัสเตอร์มิดเดิลแวร์” และสุดท้ายคือการเชื่อมต่อแต่ละโหนดข้อมูลเข้าด้วยกันผ่านทางเครือข่ายความเร็วสูง ดังนี้

**1) ส่วนประกอบด้านฮาร์ดแวร์**

โครงสร้างทางฮาร์ดแวร์ของระบบที่ใช้ในการทดลอง เป็นคลัสเตอร์ที่มีโครงสร้างเหมือนกัน (Homogenous Cluster) คือหน่วยประมวลผล ขนาดของหน่วยความจำ และส่วนประกอบอื่นๆ เหมือนกัน ระบบที่ใช้ในการทดลองเป็นระบบคลัสเตอร์ที่พัฒนาขึ้นโดยสำนักวิจัยและบริการคอมพิวเตอร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง โดยระบบในปัจจุบันนี้ ประกอบด้วยคอมพิวเตอร์ส่วนบุคคลจำนวน 5 เครื่อง โดยใช้เครื่องคอมพิวเตอร์ชนิดที่มีสองหน่วยประมวลผลในเครื่องเดียวกัน (Dual CPU) ซึ่งมีองค์ประกอบด้านฮาร์ดแวร์ที่จำเป็นในการพิจารณาเพื่อสร้างเป็นระบบคลัสเตอร์ต่างๆ ดังนี้

ก) แผงวงจรรวม (Mother Board) หรือ System Board

ข) หน่วยประมวลผล (CPUs) ในงานวิจัยนี้ใช้เครื่องคอมพิวเตอร์ชนิดที่มีสองหน่วยประมวลผลในเครื่องเดียวกัน (Dual CPUs) ยี่ห้ออินเทล (Intel Xeon) ความเร็วต่อรอบ 2.4 กิกะเฮิร์ต

ค) หน่วยความจำหลัก (RAM) ใช้ 1 กิกะไบต์

ง) หน่วยความจำสำรอง (Disk Storage) มีขนาดเท่ากับ 100 กิกะไบต์

จ) หน่วยความจำแคช (Cache Memory) ขนาด 512 kb

ฉ) เชื่อมต่อผ่านเครือข่ายความเร็ว 1000 เมกกะบิตต่อวินาที (Mbps)

**2) ระบบปฏิบัติการ**

ระบบปฏิบัติการจะเป็นส่วนสำคัญอย่างมากต่อการทำงานของคอมพิวเตอร์และระบบคลัสเตอร์เพราะคอมพิวเตอร์ในระบบคลัสเตอร์ต้องสามารถทำงานเองได้โดยอิสระไม่ขึ้นกับเงื่อนไขของเครื่องอื่นถึงแม้มีเครื่องใดเครื่องหนึ่งในระบบหยุดทำงาน ระบบคลัสเตอร์ก็ยังสามารถทำงานได้ดังนั้นระบบปฏิบัติการจึงมีความจำเป็นในส่วนที่จะทำให้คอมพิวเตอร์แต่ละเครื่องสามารถทำงานโดยอิสระต่อกันได้ และสามารถติดต่อสื่อสารกันได้ ระบบปฏิบัติการที่สามารถใช้กับระบบคลัสเตอร์แบบพีซี (Cluster of PCs) มีหลากหลาย เช่น Linux, SUSE, FreeBSD, HP-UX และระบบปฏิบัติการที่สามารถใช้กับระบบคลัสเตอร์แบบเวิร์กสเตชั่น (Cluster of Workstation) คือ Tru64 UNIX, Solaris เป็นต้น ซึ่งการเลือกใช้ระบบปฏิบัติการนั้นขึ้นอยู่กับระบบคลัสเตอร์ด้วยว่าเป็นแบบใด (PCs or Workstation) นอกจากนี้ยังขึ้นอยู่กับความสะดวก ความเชี่ยวชาญและอุปกรณ์ที่เลือกใช้ อีกอย่างหนึ่งคือจะขึ้นอยู่กับซอฟท์แวร์และชุดคำสั่งที่เลือกใช้ด้วย เช่น ถ้าเลือกใช้ OpenMosix ที่ทำงานได้บนระบบลีนุกซ์เท่านั้น ก็จำเป็นต้องเลือกใช้ลีนุกต์เป็นระบบปฏิบัติการอีกทั้งฮาร์ดแวร์ที่สนับสนุนด้วย โดยส่วนใหญ่ซอฟท์แวร์และชุดคำสั่งที่ถูกพัฒนาขึ้นมาเพื่อใช้ในระบบคลัสเตอร์นี้จะทำงานเข้ากันได้กับระบบปฏิบัติการตระกูลยูนิกซ์เกือบทุกชนิดอยู่แล้ว

ระบบคลัสเตอร์ที่ใช้ในงานวิจัยนี้เป็นระบบปฏิบัติลีนุกต์รุ่นซูซี (SUSE Version) ที่ใช้บนระบบคลัสเตอร์แบบพีซีเนื่องจากมีความสามารถและรองรับการทำงานได้หลากหลาย อีกทั้งยังมีความสามารถในการเฝ้าระวัง (Monitoring) ในส่วนงานของผู้ดูแลระบบ (Administrator) เป็นแบบออนไลน์ (Online) ทำให้เราสามารถตรวจสอบระบบคลัสเตอร์ผ่านทางอินเตอร์เน็ต (Internet) ได้ตลอดเวลาอีกด้วย

**3) คลัสเตอร์มิดเดิลแวร์**

คลัสเตอร์มิดเดิลแวร์ คือซอฟท์แวร์ที่ทำงานในระดับเดียวกับแกน (Kernel) ของระบบปฏิบัติการ มีหน้าที่ในการกระจายงาน (Process) จากโหนดหนึ่งไปยังโหนดอื่นๆ ที่อยู่ในระบบคลัสเตอร์เดียวกัน โดยส่วนใหญ่แล้วคลัสเตอร์มิดเดิลแวร์นี้จะเป็นซอฟท์แวร์ที่เขียนเพิ่มเติม

เข้าไปในแกนของระบบปฏิบัติการ เพื่อให้ทุกเครื่องที่อยู่ในระบบคลัสเตอร์เสมือนเป็นเครื่องคอมพิวเตอร์หลายหน่วยประมวลผลขนาดใหญ่ ที่เสมือนมีหน่วยความจำ หน่วยประมวลผลอยู่ที่เดียวกันเหมือนกับระบบคอมพิวเตอร์แบบ SMP (Symmetric Multi Processor) หรือแบบ MMP (Massive Multi Processor)

**4) การเชื่อมต่อเครือข่าย**

การสื่อสารระหว่างหน่วยประมวลผลในระบบคลัสเตอร์จะทำผ่านระบบเครือข่ายความเร็วสูงซึ่งอุปกรณ์เครือข่ายแต่ละชนิดจะมีความเร็วและราคาแตกต่างกันไป ตัวอย่างอุปกรณ์เครือข่ายที่นิยมนำมาใช้ในการสร้างระบบคลัสเตอร์ มีดังนี้

ก) Ethernet ในปัจจุบันอุปกรณ์ Ethernet นั้นได้ถูกพัฒนาให้มีความเร็วในการรับส่งข้อมูลสูงมากขึ้นจนถึงระดับกิกะบิตอีเทอร์เน็ต (Gigabit Ethernet) หรือ 10 กิกะบิตอีเทอร์เน็ต คือมีความเร็วในการส่งผ่านข้อมูลประมาณ 1-10 พันล้านบิตต่อวินาที ซึ่งเป็นแบบที่ใช้ในระบบคลัสเตอร์ที่ใช้ทำวิจัยนี้ด้วย

ข) Myrinet มีความเร็วในการส่งผ่านข้อมูลประมาณ 2 พันล้านบิตต่อวินาที และมีค่า Latency Time ต่ำกว่าเครือข่ายแบบ Ethernet มาก แต่มีราคาแพงกว่าอีเทอร์เน็ตสูงมาก

ค) Quadrics มีความเร็วในการส่งข้อมูลอยู่ที่ 340-900 MB/Second หรือประมาณ 2.65-7 กิกะบิตต่อวินาที และมีค่า Latency Time ต่ำมาก

ง) InfiniBand เป็นเทคโนโลยีที่มีความเร็วในการสื่อสารข้อมูลสูงมากถึง 5 กิกะบิตต่อวินาที และมีค่า Latency Time น้อยกว่า 10 ไมโครวินาที (Microsecond)

**5) เอ็มพีไอซีเอช (MPICH)**

เอ็มพีไอซีเอช (MPICH) เป็นการพัฒนาชุดคำสั่งตามมาตรฐานขึ้นมาใช้งานจริงมากที่สุดคือมาตรฐานการส่งผ่านข้อความ (Message Passing Interface: MPI) ซึ่งสามารถพัฒนาโปรแกรมภาษาต่างๆ เช่น โปรแกรมภาษาซี (C Language) โปรแกรมภาษาฟอร์แทน (Fortran Language) และโปรแกรมภาษาจาวา (Java Language) เป็นต้น ที่สนับสนุนการโปรแกรมแบบขนาน ซึ่งมีฟังก์ชันพื้นฐานต่างๆ รวมทั้งมีคำสั่งใช้งานเพื่อทำการคอมไพล์โปรแกรมที่เขียนขึ้น โดยเอ็มพีไอซีเอชนี้จะสนับสนุนการพัฒนาโปรแกรมภาษาซี และโปรแกรมภาษาฟอร์แทนเท่านั้น

การทดลองในงานวิจัยนี้ จะนำโปรแกรมที่ได้พัฒนาขึ้นไปประมวลผลบนระบบพีซีคลัสเตอร์ ของสำนักวิจัยและบริการคอมพิวเตอร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง โดยคำสั่งที่ใช้ในการทดลองมีรูปแบบดังนี้

mpirun –nolocal –np <P1> bzipC <P2> <P3> <P4>

เมื่อ <P1> คือ จำนวนหน่วยประมวลผล

<P2> คือ ชื่อแฟ้มข้อมูลที่ทำการบีบอัด

<P3> คือ วิธีการบีบอัดข้อมูล โดยมีค่าดังนี้ 1 แทนวิธีเอ็มเอสบีซิพทู (MSBzip2) 2 แทนวิธีเอดับเบิลยูบีซิพทู (AWBzip2) และ 3 แทนวิธีเอดับเบิลยูดับเบิลยูบีซิพทู (AWWBzip2)

<P4> คือ น้ำหนักของการแบ่งบล็อกข้อมูลย่อย สำหรับการบีบอัดข้อมูลแบบขนานด้วยวิธีเอดับเบิลยูดับเบิลยูบีซิพทู (AWWBzip2)

ผลการทดลองในงานวิจัยนี้ จะทำการบันทึกเวลาที่ใช้ในการประมวลผล โดยนำเวลาที่มากที่สุดที่ใช้ในการประมวลผล มาหาค่าเฉลี่ยโดยทำการทดลองซ้ำทั้งหมด 3 ครั้ง และทำการบันทึกขนาดของข้อมูลที่ผ่านการบีบอัดข้อมูลแล้ว ของทุกๆวิธีการบีบอัดข้อมูล และตรวจสอบความถูกต้องของการบีบอัดข้อมูลโดยนำแฟ้มข้อมูลที่ผ่านการบีบอัดแล้วไปคลายข้อมูลด้วยวิธีการคลายข้อมูลแบบอนุกรม และนำแฟ้มข้อมูลที่คลายข้อมูลแล้ว ตรวจสอบความถูกต้องกับแฟ้มข้อมูลต้นฉบับก่อนการบีบอัดข้อมูล ด้วยโปรแกรม WinDiff ซึ่งสามารถดาวน์โหลดได้จาก http://www.grigsoft.com/windiff.zip ซึ่งถ้าแฟ้มข้อมูลที่ผ่านการบีบอัดถูกคลายข้อมูล แล้วนำมาเปรียบเทียบกับแฟ้มข้อมูลต้นฉบับจะต้องมีข้อมูลที่เหมือนกัน

## 4.2 ลักษณะข้อมูล การเลือกข้อมูลและเหตุผลการเลือก

ข้อมูลที่ถูกใช้ในงานวิจัยต่างๆ ที่เกี่ยวกับการบีบอัดข้อมูลแบบที่ไม่มีการสูญเสียข้อมูล (Lossless Compression) แบ่งออกเป็น 4 กลุ่ม ได้แก่

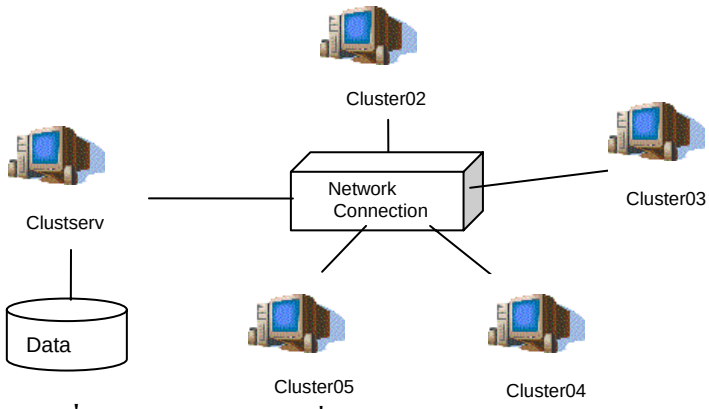
1) ชุดข้อมูลแคลกอรี่ (Calgary Corpus)
2) ชุดข้อมูลแคนเทอร์เบอรี่ (Canterbury Corpus)
3) ชุดข้อมูลโปรตีน (Protein Corpus)
4) ชุดข้อมูลซิลีเซีย (Silesia Corpus)

ชุดข้อมูลดังกล่าวถูกใช้ในการวัดสมรรถนะ (Performance Measurement) ของวิธีการบีบอัดข้อมูลแบบอนุกรมวิธีต่างๆ แต่งานวิจัยนี้เป็นการพัฒนาเพื่อเพิ่มสมรรถนะของการบีบอัดข้อมูลข้อมูลแบบขนาน ทำให้ชุดข้อมูลที่ใช้ในการวัดสมรรถนะของวิธีการบีบอัดข้อมูลแบบอนุกรม ไม่เหมาะสมสำหรับการวัดสมรรถนะของวิธีการบีบอัดข้อมูลแบบขนาน เนื่องจากชุดข้อมูลเหล่านี้มีขนาดเล็ก ดังนั้น งานวิจัยนี้ จึงทำการเพิ่มชุดข้อมูลที่มีขนาดใหญ่เพื่อใช้ในการวัดสมรรถนะ โดยชุด

ข้อมูลที่ใช้ในการวัดสมรรถนะประกอบด้วย แฟ้มข้อมูลต่างๆ ในชุดข้อมูลกูเทนเบริก (Gutenberg Corpus) ดังนี้

| ชื่อข้อมูล | ขนาดข้อมูล (ไบต์) | ที่มาของข้อมูล |
|---|---|---|
| 00ws110.txt | 4,651,867 | http://gutenberg.elib.com/gutenberg/etext00/ |
| 20hgp10.txt | 12,066,257 | http://gutenberg.elib.com/gutenberg/etext00/ |
| 22hgp11.txt | 34,707,288 | http://gutenberg.elib.com/gutenberg/etext00/ |
| 0xhgp10.txt | 57,923,382 | http://gutenberg.elib.com/gutenberg/etext00/ |
| 07hgp10.txt | 73,370,264 | http://gutenberg.elib.com/gutenberg/etext00/ |

งานวิจัยนี้ ข้อมูลที่ใช้ในการวัดสมรรถนะของการบีบอัดข้อมูลจะถูกจัดเก็บอยู่ในหน่วยความจำสำรองของหน่วยประมวลผลหลัก ซึ่งเรียกการเก็บข้อมูลแบบนี้ว่า “การเก็บข้อมูลแบบรวมศูนย์กลาง” (Centralized Data) ดังรูปที่ 4.1




**รูปที่ 4.1** ระบบคลัสเตอร์ที่มีการเก็บข้อมูลแบบรวมศูนย์กลาง

## 4.3 ผลการทดลอง

จากการพัฒนาโปรแกรมบีบอัดข้อมูลแบบขนาน ได้นำไปทดลองบนระบบพีซีคลัสเตอร์ของสำนักวิจัยและบริการคอมพิวเตอร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ซึ่งประกอบด้วย หน่วยประมวลผล 10 หน่วย ขนาด 2.4 GHz จำนวน 5 เครื่องหน่วยความจำหลัก 1 Gb หน่วยความจำสำรอง 100 Gb หน่วยความจำแคช (Cache memory) 512 Kb เชื่อมต่อกันด้วยความเร็วขนาด 1000 Mbps ระบบปฏิบัติการ Suse และโปรแกรม MPICH 1.2 ทำการวัดสมรรถนะของวิธีการบีบอัดข้อมูลแบบขนานทั้ง 3 วิธีที่เสนอ คือ

1) การบีบอัดข้อมูลแบบขนานด้วยวิธีเอ็มเอสบีซิพทู (Master-Slave Bzip2 : MSBzip2)
2) การบีบอัดข้อมูลแบบขนานด้วยวิธีเอดับเบิลยูบีซิพทู (All-Worker Bzip2 : AWBzip2)
3) การบีบอัดข้อมูลแบบขนานด้วยวิธีเอดับเบิลยูดับเบิลยูบีซิพทู (All-Worker Weight Bzip2 : AWWBzip2)

รวมถึงการบีบอัดข้อมูลแบบอนุกรมเพื่อนำมาเปรียบเทียบกัน โดยการวัดสมรรถนะประกอบด้วย เวลาที่ใช้ในการประมวลผล (Response Time) อัตราการเพิ่มขึ้นของความเร็ว (Speedup) และประสิทธิภาพ (Efficiency)

**<u>หมายเหตุ</u>**

วิธีการบีบอัดข้อมูลแบบขนานด้วยวิธีเอดับเบิลยูดับเบิลยูบีซิพทู (AWWBzip2) จะทำการทดลองซ้ำๆ โดยให้น้ำหนักที่กำหนดให้บล็อกข้อมูลย่อยมีขนาดแตกต่างกัน ($w$) ดังต่อไปนี้ 0.01%, 0.02%, 0.03%, …, 0.2% ของขนาดข้อมูลที่จะทำการบีบอัดข้อมูล เพื่อหาค่า $w$ ที่เหมาะสมกับระบบพีซีคลัสเตอร์ที่ใช้ในการทดลอง ซึ่งน้ำหนักที่กำหนดให้บล็อกข้อมูลย่อยมีขนาดแตกต่างกัน ($w$) ที่เหมาะสมกับระบบพีซีคลัสเตอร์ที่ใช้ในการทดลอง คือ 0.1 % ของขนาดข้อมูลที่จะทำการบีบอัดข้อมูล

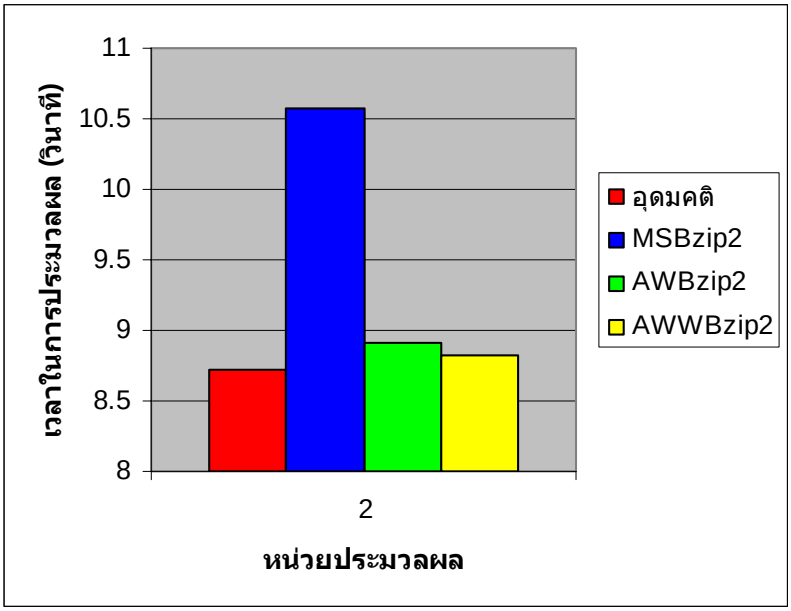
### 4.3.1 ผลการทดลองเปรียบเทียบกับเวลาในอุดมคติ

การทดลองเปรียบเทียบเวลาที่ประมวลผลจริงกับเวลาในอุดมคติ ซึ่งเป็นการวัดสมรรถนะของระบบ (System Performance) ที่ใช้ในการทดลอง โดยทำการทดลองบีบอัดข้อมูลแฟ้มข้อมูล 22hgp11.txt ซึ่งมีขนาดเท่ากับ 34,707,288 ไบต์ ประมวลผลด้วยหน่วยประมวลผล(P) 2 หน่วยประมวลผล มีผลการทดลองดังนี้ ตารางที่ 4.1 และรูปที่ 4.2 แสดงผลการเปรียบเทียบเวลาที่ใช้ในการประมวลผล ทั้ง 3 วิธี เปรียบเทียบกับเวลาที่ใช้ในการประมวลผลในอุดมคติ ($T_S/P$) ตารางที่ 4.2 และรูปที่ 4.3 แสดงอัตราการเพิ่มขึ้นของความเร็ว ทั้ง 3 วิธี ($S_P = T_S/T_P$) เปรียบเทียบกับอัตราการ

เพิ่มขึ้นของความเร็วในอุดมคติ ($S_P$=P) ตารางที่ 4.3 และรูปที่ 4.4 แสดงประสิทธิภาพในการบีบอัดข้อมูล ทั้ง 3 วิธี ($E_P = S_P/P$) เปรียบเทียบกับประสิทธิภาพในอุดมคติ ($E_P$ =1)

**ตารางที่ 4.1** เวลาที่ใช้ในการบีบอัดข้อมูล ทั้ง 3 วิธี เมื่อใช้หน่วยประมวลผล 2 หน่วยประมวลผล ประมวลผลบีบอัดแฟ้มข้อมูล 22hgp11.txt ซึ่งมีขนาดเท่ากับ 34,707,288 ไบต์

| **วิธีการบีบอัดข้อมูล** | **เวลาในการประมวลผล (วินาที)** |
|---|---|
| อุดมคติ | 8.7252 |
| ***วิธีแบบขนาน*** | |
| MSBzip2 | 10.5782 |
| AWBzip2 | 8.9070 |
| AWWBzip2 | 8.8217 |

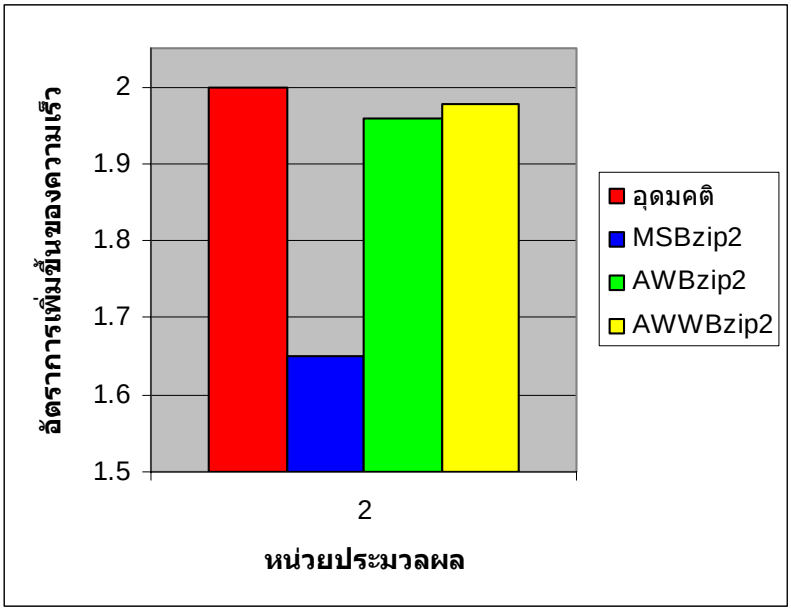



**รูปที่ 4.2** เปรียบเทียบเวลาที่ใช้ในการบีบอัดข้อมูล ทั้ง 3 วิธี กับเวลาในอุดมคติ เมื่อ P=2

จากรูปที่ 4.2 สามารถอธิบายได้ว่าเวลาที่ใช้ในการประมวลผลวิธี AWWBzip2 ที่เสนอในวิทยานิพนธ์ เมื่อใช้หน่วยประมวลผล 2 หน่วยประมวลผล ประมวลผลบีบอัดแฟ้มข้อมูล 22hgp11.txt ซึ่งมีขนาดเท่ากับ 34,707,288 ไบต์จะใช้เวลาในการประมวลผล 8.8217 วินาทีใกล้เคียงกับเวลาในอุดมคติ คือ 8.7252 วินาที ซึ่งคิดเป็น 98.9% และวิธี MSBzip2 เป็นวิธีที่ใช้เวลาในการประมวลผลมากที่สุด คือ 10.5782 วินาที

**ตารางที่ 4.2** อัตราการเพิ่มขึ้นของความเร็ว ทั้ง 3 วิธี เมื่อใช้หน่วยประมวลผล 2 หน่วยประมวลผล ประมวลผลบีบอัดแฟ้มข้อมูล 22hgp11.txt ซึ่งมีขนาดเท่ากับ 34,707,288 ไบต์

| วิธีการบีบอัดข้อมูล | อัตราการเพิ่มขึ้นของความเร็ว |
|---|---|
| อุดมคติ | 2 |
| ***วิธีแบบขนาน*** | |
| MSBzip2 | 1.6496 |
| AWMBzip2 | 1.9591 |
| AWWPBzip2 | 1.9781 |

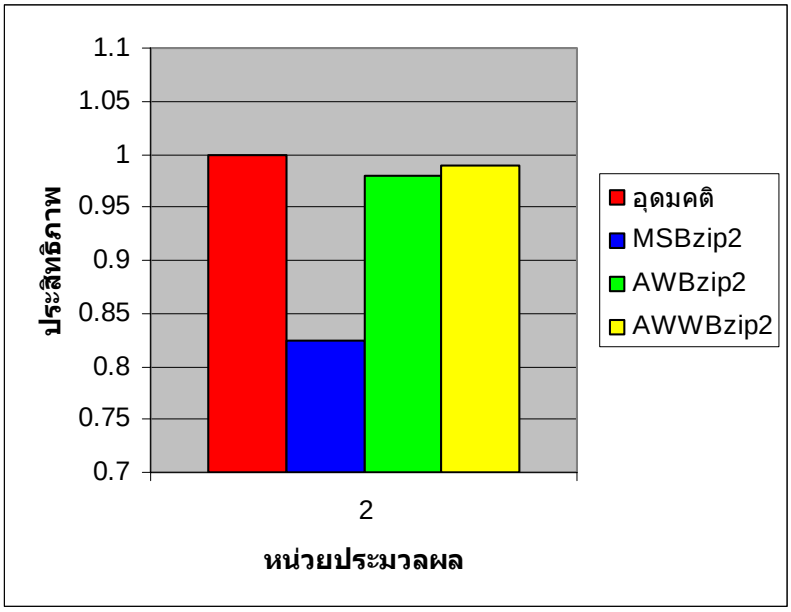



**รูปที่ 4.3** เปรียบเทียบอัตราการเพิ่มขึ้นของความเร็วทั้ง 3 วิธี กับอัตราการเพิ่มขึ้นของความเร็วในอุดมคติ เมื่อ P=2

จากรูปที่ 4.3 สามารถอธิบายได้ว่าอัตราการเพิ่มขึ้นของความเร็วของการบีบอัดข้อมูล เมื่อใช้หน่วยประมวลผล 2 หน่วยประมวลผล ประมวลผลบีบอัดแฟ้มข้อมูล 22hgp11.txt ซึ่งมีขนาดเท่ากับ 34,707,288 ไบต์ วิธี AWWBzip2 เท่ากับ 1.9781 ซึ่งเป็นวิธีที่ดีที่สุด ส่วนวิธี MSBzip2 มีอัตราการเพิ่มขึ้นของความเร็วเท่ากับ 1.6496 ซึ่งน้อยกว่าทุกๆวิธี

**ตารางที่ 4.3** ประสิทธิภาพ ทั้ง 3 วิธี เมื่อใช้หน่วยประมวลผล 2 หน่วยประมวลผล ประมวลผลบีบอัดแฟ้มข้อมูล 22hgp11.txt ซึ่งมีขนาดเท่ากับ 34,707,288 ไบต์

| วิธีการบีบอัดข้อมูล | ประสิทธิภาพ |
|---|---|
| อุดมคติ | 1 |
| ***วิธีแบบขนาน*** | |
| MSBzip2 | 0.8248 |
| AWMBzip2 | 0.9795 |
| AWWPBzip2 | 0.9890 |




**รูปที่ 4.4** เปรียบเทียบประสิทธิภาพทั้ง 3 วิธี กับประสิทธิภาพในอุดมคติ เมื่อ P=2

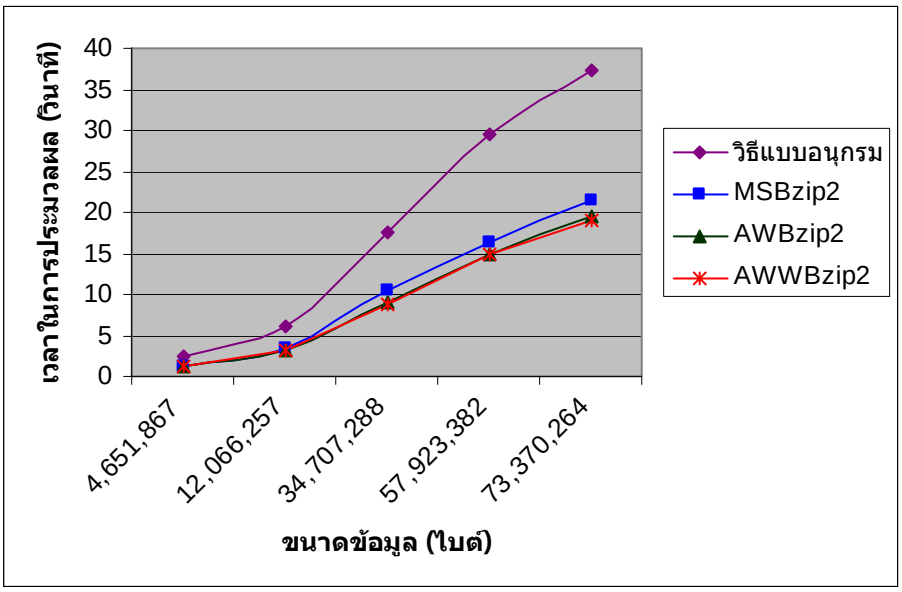
จากรูปที่ 4.4 แสดงประสิทธิภาพของการบีบอัดข้อมูลทั้ง 3 วิธี เมื่อใช้หน่วยประมวลผล 2 หน่วยประมวลผล ประมวลผลบีบอัดแฟ้มข้อมูล 22hgp11.txt ซึ่งมีขนาดเท่ากับ 34,707,288 ไบต์ จะเห็นว่าวิธี AWWBzip2 มีค่าประสิทธิภาพเท่ากับ 0.9890 จะให้ผลเข้าใกล้ 1 ส่วนวิธี MSBzip2 จะมีประสิทธิภาพเท่ากับ 0.8248 น้อยที่สุดเมื่อเทียบกับทุกๆวิธี

### 4.3.2 ผลการทดลองเปรียบเทียบเมื่อขนาดข้อมูลที่ใช้ทดลองต่างกัน

การทดลองเปรียบเทียบการบีบอัดข้อมูลเมื่อขนาดของข้อมูลที่ใช้ทดลองมีขนาดต่างกัน กำหนดให้จำนวนหน่วยประมวลผลที่ใช้ในการคำนวณเท่ากับ 2 หน่วยประมวลผล ทำการเปรียบเทียบวิธีการบีบอัดข้อมูลแบบอนุกรม (Sequential Compression) และวิธีการบีบอัดข้อมูลแบบขนานทั้ง 3 วิธี โดยบีบอัดแฟ้มข้อมูล ดังนี้ 00ws110.txt, 20hgp10.txt, 22hgp11.txt, 0xhgp10.txt และ 07hgp10.txt ซึ่งมีขนาด 4,651,867 ไบต์, 12,066,257 ไบต์, 34,707,288 ไบต์, 57,923,382 ไบต์ และ 73,370,264 ไบต์ ตามลำดับ ซึ่งผลการทดลองแสดงดังนี้ ตารางที่ 4.4 และรูปที่ 4.5 แสดงเวลาที่ใช้ในการประมวลผล นอกจากนี้เราสามารถนำเวลาที่ใช้ในการประมวลผลมาคำนวณหาอัตราการเพิ่มขึ้นของความเร็ว ดังแสดงในตารางที่ 4.5 และรูปที่ 4.6 จากนั้นนำค่าของอัตราการเพิ่มขึ้นของความเร็วมาคำนวณหาประสิทธิภาพ ดังแสดงในตารางที่ 4.6 และรูปที่ 4.7

**ตารางที่ 4.4** เวลาที่ใช้ในวิธีการบีบอัดข้อมูลทั้ง 3 วิธี เมื่อใช้หน่วยประมวลผล 2 หน่วยประมวลผล บีบอัดแฟ้มข้อมูล ดังนี้ 00ws110.txt, 20hgp10.txt, 22hgp11.txt, 0xhgp10.txt และ 07hgp10.txt ซึ่งมีขนาด 4,651,867 ไบต์, 12,066,257 ไบต์, 34,707,288 ไบต์, 57,923,382 ไบต์ และ 73,370,264 ไบต์ ตามลำดับ

| **ขนาดข้อมูล (ไบต์)** | **วิธีแบบอนุกรม** | **วิธีแบบขนาน** | | |
|---|---|---|---|---|
| | | MSBzip2 | AWBzip2 | AWWBzip2 |
| **4,651,867** | 2.4136 | 1.2826 | 1.2197 | 1.2157 |
| **12,066,257** | 6.1348 | 3.3675 | 3.1579 | 3.1059 |
| **34,707,288** | 17.4500 | 10.5782 | 8.9070 | 8.8217 |
| **57,923,382** | 29.4150 | 16.3253 | 14.9430 | 14.7927 |
| **73,370,264** | 37.2360 | 21.4778 | 19.4758 | 18.9934 |

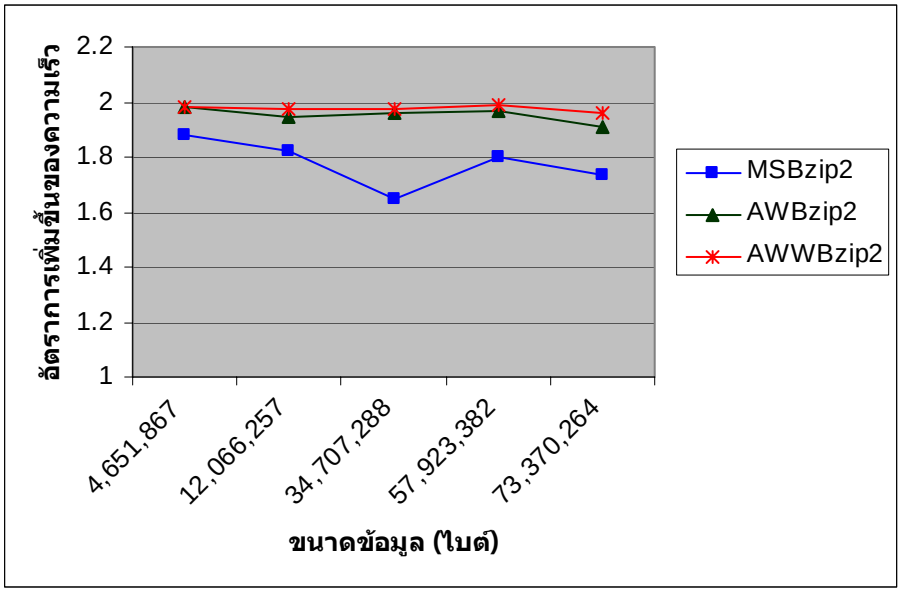



**รูปที่ 4.5** เปรียบเทียบเวลาที่ใช้ในการบีบอัดข้อมูล ทั้ง 3 วิธี เมื่อขนาดข้อมูลที่ใช้ทดลองต่างกัน และP = 2

จากรูปที่ 4.5 แสดงเวลาที่ใช้ในการบีบอัดข้อมูล โดยทำการเปรียบเทียบเวลาทั้ง 3 วิธี เมื่อใช้หน่วยประมวลผล 2 หน่วยประมวลผล บีบอัดแฟ้มข้อมูล ดังนี้ 00ws110.txt, 20hgp10.txt, 22hgp11.txt, 0xhgp10.txt และ 07hgp10.txt ซึ่งมีขนาด 4,651,867 ไบต์, 12,066,257 ไบต์, 34,707,288 ไบต์, 57,923,382 ไบต์ และ 73,370,264 ไบต์ ตามลำดับ จะเห็นว่าเมื่อข้อมูลมีขนาดใหญ่ขึ้นเวลาที่ใช้ในการบีบอัดข้อมูลนั้นจะเพิ่มขึ้นตามไปด้วย โดยวิธี AWWBzip2 ใช้เวลาในการประมวลผลน้อยที่สุด และวิธี MSBzip2 ใช้เวลาในการประมวลผลมากกว่าวิธีอื่นๆที่เสนอ

**ตารางที่ 4.5** อัตราการเพิ่มขึ้นของความเร็วทั้ง 3 วิธี เมื่อใช้หน่วยประมวลผล 2 หน่วยประมวลผลบีบอัดแฟ้มข้อมูล ดังนี้ 00ws110.txt, 20hgp10.txt, 22hgp11.txt, 0xhgp10.txt และ 07hgp10.txt ซึ่งมีขนาด 4,651,867 ไบต์, 12,066,257 ไบต์, 34,707,288 ไบต์, 57,923,382 ไบต์ และ 73,370,264 ไบต์ ตามลำดับ

| ขนาดข้อมูล (ไบต์) | วิธีแบบขนาน | | |
|---|---|---|---|
| | MSBzip2 | AWBzip2 | AWWBzip2 |
| **4,651,867** | 1.8817 | 1.9788 | 1.9853 |
| **12,066,257** | 1.8217 | 1.9426 | 1.9751 |
| **34,707,288** | 1.6496 | 1.9591 | 1.9781 |
| **57,923,382** | 1.8018 | 1.9684 | 1.9884 |
| **73,370,264** | 1.7336 | 1.9118 | 1.9604 |



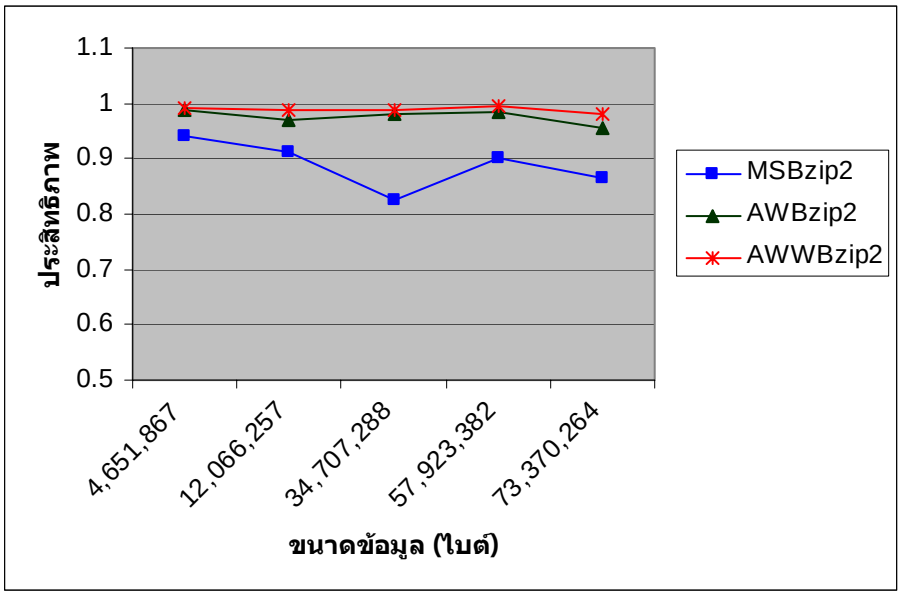

**รูปที่ 4.6** เปรียบเทียบอัตราการเพิ่มขึ้นของความเร็วทั้ง 3 วิธี เมื่อขนาดข้อมูลที่ใช้ทดลองต่างกัน และP = 2

จากรูปที่ 4.6 แสดงการเปรียบเทียบอัตราการเพิ่มขึ้นของความเร็วทั้ง 3 วิธี เมื่อใช้หน่วยประมวลผล 2 หน่วยประมวลผล บีบอัดแฟ้มข้อมูล ดังนี้ 00ws110.txt, 20hgp10.txt, 22hgp11.txt, 0xhgp10.txt และ 07hgp10.txt ซึ่งมีขนาด 4,651,867 ไบต์, 12,066,257 ไบต์, 34,707,288 ไบต์,

57,923,382 ไบต์ และ 73,370,264 ไบต์ ตามลำดับ จะเห็นว่าวิธี AWWBzip2 มีอัตราการเพิ่มขึ้นของความเร็วมากที่สุด ส่วนวิธี MSBzip2 มีอัตราการเพิ่มขึ้นของความเร็วน้อยที่สุด

**ตารางที่ 4.6** ประสิทธิภาพทั้ง 3 วิธี เมื่อใช้หน่วยประมวลผล 2 หน่วยประมวลผล บีบอัดแฟ้มข้อมูลดังนี้ 00ws110.txt, 20hgp10.txt, 22hgp11.txt, 0xhgp10.txt และ 07hgp10.txt ซึ่งมีขนาด 4,651,867 ไบต์, 12,066,257 ไบต์, 34,707,288 ไบต์, 57,923,382 ไบต์ และ 73,370,264 ไบต์ ตามลำดับ

| **ขนาดข้อมูล (ไบต์)** | **วิธีแบบขนาน** | | |
|---|---|---|---|
| | MSBzip2 | AWBzip2 | AWWBzip2 |
| **4,651,867** | 0.9408 | 0.9894 | 0.9926 |
| **12,066,257** | 0.9108 | 0.9713 | 0.9875 |
| **34,707,288** | 0.8248 | 0.9795 | 0.9890 |
| **57,923,382** | 0.9009 | 0.9842 | 0.9942 |
| **73,370,264** | 0.8668 | 0.9559 | 0.9802 |




**รูปที่ 4.7** เปรียบเทียบประสิทธิภาพ ทั้ง 3 วิธี เมื่อขนาดข้อมูลที่ใช้ทดลองต่างกัน และP = 2

จากรูปที่ 4.7 แสดงการเปรียบเทียบประสิทธิภาพทั้ง 3 วิธี เมื่อใช้หน่วยประมวลผล 2 หน่วยประมวลผล บีบอัดแฟ้มข้อมูล ดังนี้ 00ws110.txt, 20hgp10.txt, 22hgp11.txt, 0xhgp10.txt และ 07hgp10.txt ซึ่งมีขนาด 4,651,867 ไบต์, 12,066,257 ไบต์, 34,707,288 ไบต์, 57,923,382 ไบต์

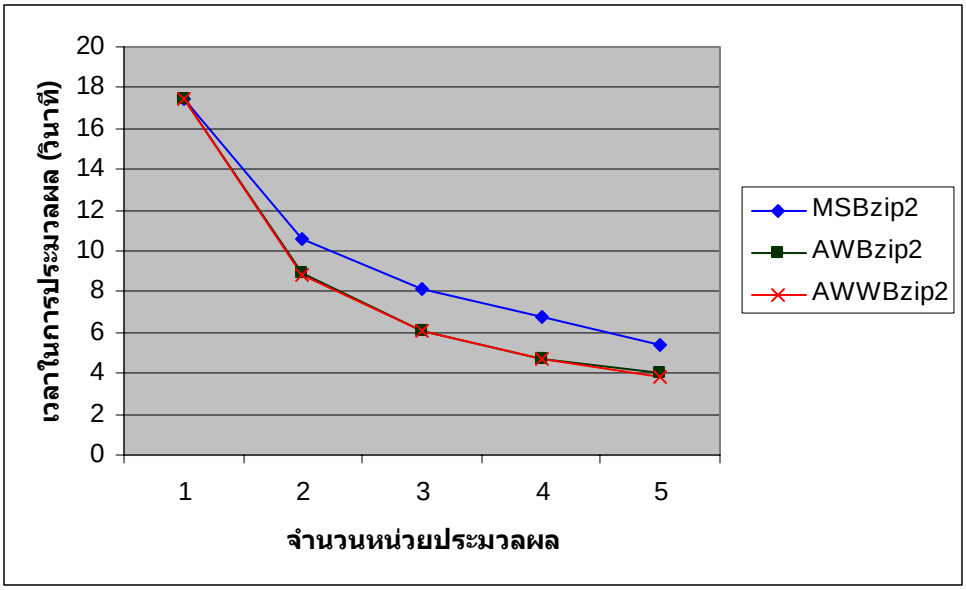
และ 73,370,264 ไบต์ ตามลำดับ จะเห็นว่าวิธี AWWBzip2 มีประสิทธิภาพมากที่สุด ส่วนวิธี MSBzip2 มีประสิทธิภาพน้อยที่สุด

### 4.3.3 ผลการทดลองเปรียบเทียบเมื่อจำนวนหน่วยประมวลผลที่ใช้ในการทดลองต่างกัน

ผลการทดลองเปรียบเทียบการบีบอัดข้อมูลเมื่อกำหนดแฟ้มข้อมูลคือ 22hgp11.txt ซึ่งมีขนาดเท่ากับ 34,707,288 ไบต์ และทำการเพิ่มจำนวนหน่วยประมวลผลตั้งแต่ 1 หน่วยประมวลผล 2 หน่วยประมวลผล 3 หน่วยประมวลผล 4 หน่วยประมวลผล และ 5 หน่วยประมวลผลตามลำดับ โดยจะพิจารณาเวลาทั้งหมดที่ใช้ในการประมวลผล ดังแสดงในตารางที่ 4.7 และรูปที่ 4.8 นอกจากนี้เราสามารถนำเวลาที่ใช้ในการประมวลผลมาคำนวณหาอัตราการเพิ่มขึ้นของความเร็ว ดังแสดงในตารางที่ 4.8 และรูปที่ 4.9 จากนั้นนำค่าของอัตราการเพิ่มขึ้นของความเร็วมาคำนวณหาประสิทธิภาพ ดังแสดงในตารางที่ 4.9 และรูปที่ 4.10

**ตารางที่ 4.7** เวลาที่ใช้ในการประมวลผลทั้ง 3 วิธี โดยใช้หน่วยประมวลผลตั้งแต่ 1, 2, 3, 4 และ 5 หน่วยประมวลผล ตามลำดับ ทำการบีบอัดแฟ้มข้อมูล 22hgp11.txt ซึ่งมีขนาดเท่ากับ 34,707,288 ไบต์

| **จำนวน หน่วยประมวลผล** | **วิธีแบบขนาน** | | |
|---|---|---|---|
| | MSBzip2 | AWBzip2 | AWWBzip2 |
| **1** | 17.4504 | 17.4504 | 17.4504 |
| **2** | 10.5782 | 8.9070 | 8.8217 |
| **3** | 8.0914 | 6.0438 | 6.0379 |
| **4** | 6.7806 | 4.7017 | 4.6970 |
| **5** | 5.3943 | 4.0363 | 3.8658 |

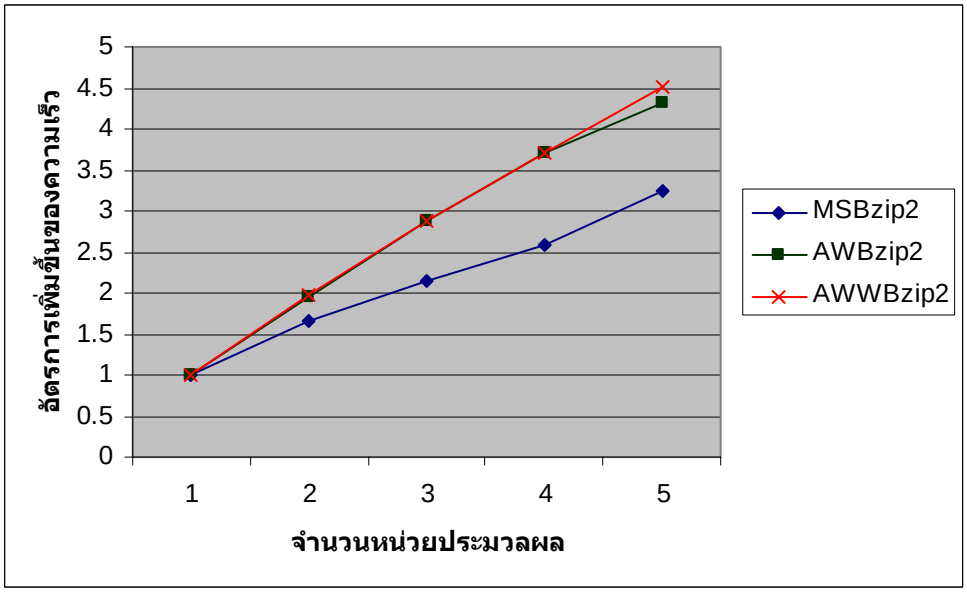



**รูปที่ 4.8** เปรียบเทียบเวลาที่ใช้ในการประมวลผล เมื่อจำนวนหน่วยประมวลผลที่ใช้ในการทดลองต่างกัน

จากรูปที่ 4.8 แสดงเวลาที่ใช้ในการประมวลผล ซึ่งจะเห็นได้ว่าเมื่อเพิ่มจำนวนหน่วยประมวลผล เวลาในการประมวลผลจะลดลง โดยวิธี AWWBzip2 ใช้เวลาในการประมวลผลน้อยที่สุด เท่ากับ 8.8217 วินาที, 6.0379 วินาที, 4.6970 วินาที และ 3.8658 วินาที เมื่อจำนวนหน่วยประมวลผลเท่ากับ 2, 3, 4 และ 5 หน่วยประมวลผลตามลำดับ และวิธี MSBzip2 ใช้เวลาในการประมวลผลมากกว่าทุกๆวิธี

**ตารางที่ 4.8** อัตราการเพิ่มขึ้นของความเร็ว ทั้ง 3 วิธี โดยใช้หน่วยประมวลผลตั้งแต่ 1, 2, 3, 4 และ 5 หน่วยประมวลผล ตามลำดับ ทำการบีบอัดแฟ้มข้อมูล 22hgp11.txt ซึ่งมีขนาดเท่ากับ 34,707,288 ไบต์

| **จำนวนหน่วยประมวลผล** | **วิธีแบบขนาน** | | |
|---|---|---|---|
| | MSBzip2 | AWBzip2 | AWWBzip2 |
| **1** | 1 | 1 | 1 |
| **2** | 1.6496 | 1.9591 | 1.9781 |
| **3** | 2.1566 | 2.8873 | 2.8901 |
| **4** | 2.5735 | 3.7114 | 3.7151 |
| **5** | 3.2349 | 4.3233 | 4.5139 |

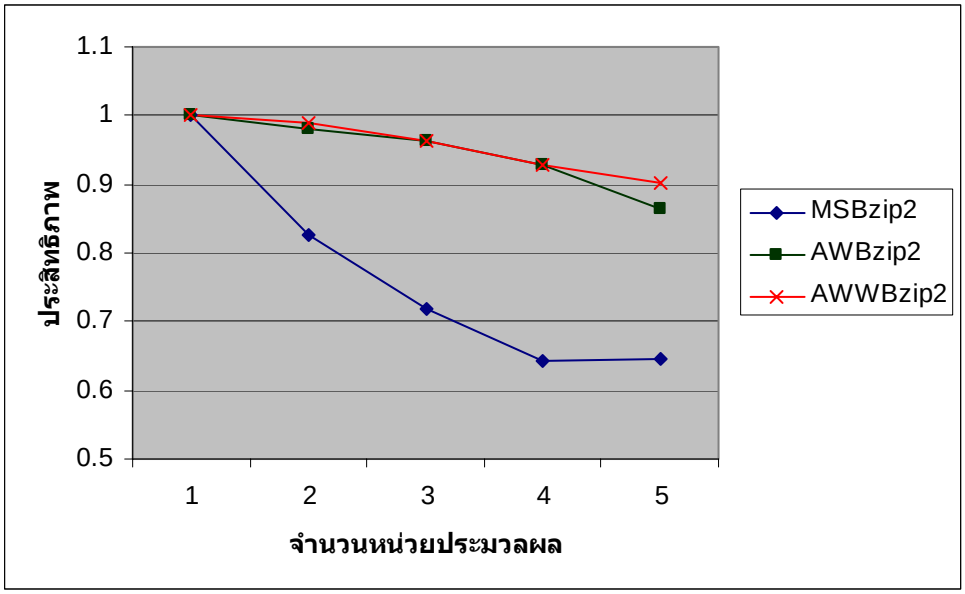



**รูปที่ 4.9** เปรียบเทียบอัตราการเพิ่มขึ้นของความเร็ว เมื่อจำนวนหน่วยประมวลผลที่ใช้ในการทดลองต่างกัน

จากรูปที่ 4.9 แสดงอัตราการเพิ่มขึ้นของความเร็ว จะเห็นได้ว่าเมื่อเพิ่มจำนวนหน่วยประมวลผล อัตราการเพิ่มขึ้นของความเร็วจะเพิ่มขึ้นใกล้เคียงกับจำนวนหน่วยประมวลผล โดยวิธี AWWBzip2 เท่ากับ 1.9781, 2.8901, 3.7151 และ 4.5139 เมื่อจำนวนหน่วยประมวลผลเท่ากับ 2, 3, 4 และ 5 หน่วยประมวลผลตามลำดับ และวิธี MSBzip2 มีอัตราการเพิ่มขึ้นของความเร็วน้อยกว่าทุกๆวิธี

**ตารางที่ 4.9** ประสิทธิภาพทั้ง 3 วิธี โดยใช้หน่วยประมวลผลตั้งแต่ 1, 2, 3, 4 และ 5 หน่วยประมวลผล ตามลำดับ ทำการบีบอัดแฟ้มข้อมูล 22hgp11.txt ซึ่งมีขนาดเท่ากับ 34,707,288 ไบต์

| **จำนวนหน่วยประมวลผล** | **วิธีแบบขนาน** | | |
|---|---|---|---|
| | MSBzip2 | AWBzip2 | AWWBzip2 |
| **1** | 1 | 1 | 1 |
| **2** | 0.8248 | 0.9795 | 0.9890 |
| **3** | 0.7188 | 0.9624 | 0.9633 |
| **4** | 0.6433 | 0.9278 | 0.9287 |
| **5** | 0.6469 | 0.8646 | 0.9027 |




**รูปที่ 4.10** เปรียบเทียบประสิทธิภาพ

จากรูปที่ 4.10 แสดงผลการเปรียบเทียบประสิทธิภาพ วิธี AWWBzip2 ให้ประสิทธิภาพเข้าใกล้ 1 เท่ากับ 0.9890, 0.9633, 0.9287 และ 0.9027 เมื่อหน่วยประมวลผลเท่ากับ 2, 3, 4 และ 5 หน่วยประมวลผล ตามลำดับ เนื่องจากต้องใช้ระยะเวลาในการติดต่อระหว่างหน่วยประมวลผลเพิ่มมากขึ้นเมื่อเพิ่มจำนวนหน่วยประมวลผล ส่งผลให้ประสิทธิภาพลดลง ส่วนวิธี MSBzip2 มีค่าประสิทธิภาพน้อยกว่าทุกๆ วิธี

### 4.3.4 ผลการทดลองเปรียบเทียบขนาดข้อมูลที่ผ่านการบีบอัดข้อมูลด้วยวิธีต่างๆ

ผลการทดลองเปรียบเทียบขนาดข้อมูลที่ผ่านการบีบอัดข้อมูลด้วยวิธีต่างๆ เมื่อบีบอัดแฟ้มข้อมูล ดังนี้ 00ws110.txt, 20hgp10.txt, 22hgp11.txt, 0xhgp10.txt และ 07hgp10.txt ซึ่งมีขนาด 4,651,867 ไบต์, 12,066,257 ไบต์, 34,707,288 ไบต์, 57,923,382 ไบต์ และ 73,370,264 ไบต์ ตามลำดับ โดยประมวลผลตั้งแต่ 2 หน่วยประมวลผล 3 หน่วยประมวลผล 4 หน่วยประมวลผล และ 5 หน่วยประมวลผล ตามลำดับ มีผลแสดงในตารางที่ 4.10

**ตารางที่ 4.10** ขนาดข้อมูลที่ผ่านการบีบอัดทั้ง 3 วิธี โดยใช้หน่วยประมวลผลตั้งแต่ 2, 3, 4 และ 5 หน่วยประมวลผล ตามลำดับ ทำการบีบอัดแฟ้มข้อมูล ดังนี้ 00ws110.txt, 20hgp10.txt, 22hgp11.txt, 0xhgp10.txt และ 07hgp10.txt ซึ่งมีขนาด 4,651,867 ไบต์, 12,066,257 ไบต์, 34,707,288 ไบต์, 57,923,382 ไบต์ และ 73,370,264 ไบต์ ตามลำดับ

| ชื่อแฟ้มข้อมูล | ขนาดข้อมูลก่อนการบีบอัด | ขนาดข้อมูลหลังการบีบอัด ด้วยวิธีต่าง | | |
|---|---|---|---|---|
| | | **msbzip2** | **awbzip2** | **awwbzip2** |
| **ประมวลผลการบีบอัดข้อมูลด้วย 2 หน่วยประมวลผล** | | | | |
| **00ws110.txt** | 4,651,867 | 1,370,442 | 1,370,442 | 1,370,546 |
| **20hgp10.txt** | 12,066,257 | 3,296,022 | 3,296,022 | 3,295,629 |
| **22hgp11.txt,** | 34,707,288 | 9,241,934 | 9,241,934 | 9,239,995 |
| **0xhgp10.txt** | 57,923,382 | 15,774,308 | 15,774,308 | 15,772,779 |
| **07hgp10.txt** | 73,370,264 | 20,007,150 | 20,007,150 | 20,009,326 |
| **ประมวลผลการบีบอัดข้อมูลด้วย 3 หน่วยประมวลผล** | | | | |
| **00ws110.txt** | 4,651,867 | 1,369,958 | 1,369,958 | 1,369,991 |
| **20hgp10.txt** | 12,066,257 | 3,296,072 | 3,296,072 | 3,296,145 |
| **22hgp11.txt,** | 34,707,288 | 9,240,212 | 9,240,212 | 9,239,808 |
| **0xhgp10.txt** | 57,923,382 | 15,771,424 | 15,771,424 | 15,770,780 |
| **07hgp10.txt** | 73,370,264 | 20,001,765 | 20,001,765 | 20,005,646 |
| **ประมวลผลการบีบอัดข้อมูลด้วย 4 หน่วยประมวลผล** | | | | |
| **00ws110.txt** | 4,651,867 | 1,382,695 | 1,382,695 | 1,382,646 |
| **20hgp10.txt** | 12,066,257 | 3,297,507 | 3,297,507 | 3,297,233 |
| **22hgp11.txt,** | 34,707,288 | 9,240,465 | 9,240,465 | 9,241,781 |
| **0xhgp10.txt** | 57,923,382 | 15,778,491 | 15,778,491 | 15,773,721 |
| **07hgp10.txt** | 73,370,264 | 20,006,693 | 20,006,693 | 20,003,106 |
| **ประมวลผลการบีบอัดข้อมูลด้วย 5 หน่วยประมวลผล** | | | | |
| **00ws110.txt** | 4,651,867 | 1,374,777 | 1,374,777 | 1,374,040 |
| **20hgp10.txt** | 12,066,257 | 3,296,830 | 3,296,830 | 3,297,506 |
| **22hgp11.txt,** | 34,707,288 | 9,240,722 | 9,240,722 | 9,238,140 |
| **0xhgp10.txt** | 57,923,382 | 15,768,194 | 15,768,194 | 15,769,714 |
| **07hgp10.txt** | 73,370,264 | 20,010,471 | 20,010,471 | 20,013,244 |

จากตารางที่ 4.10 พบว่า ขนาดข้อมูลที่ผ่านการบีบอัดข้อมูลด้วยวิธี MSBzip2 และการบีบอัดข้อมูลด้วยวิธี AWBzip2 มีขนาดข้อมูลที่ผ่านการบีบอัดแล้วเท่ากัน ส่วนวิธีการบีบอัดข้อมูลด้วยวิธี AWWBzip2 ขนาดข้อมูลที่ผ่านการบีบอัดแล้วอาจเพิ่ม หรือลดขนาดไปจากการบีบอัดข้อมูลวิธีอื่นๆโดยเฉลี่ย 0.0013 % เมื่อเทียบกับขนาดแฟ้มข้อมูลที่ทำการบีบอัด ขนาดที่เพิ่มหรือลดของการบีบอัดข้อมูลวิธี AWWBzip2 เกิดขึ้นเนื่องจาก การแบ่งขนาดบล็อกข้อมูลย่อยที่จะบีบอัดข้อมูลไม่เท่ากัน นั่นคือ แต่ละหน่วยประมวลผลลำดับเดียวกันของการบีบอัดข้อมูลวิธี AWWBzip2 และวิธีการบีบอัดข้อมูลวิธีอื่นๆ จะมีขนาดข้อมูลที่จะทำการบีบอัดไม่เท่ากัน ทำให้ข้อมูลที่ผ่านการบีบอัดแล้วของแต่ละหน่วยประมวลผลลำดับเดียวกันมีขนาดไม่เท่ากัน ส่งผลให้ผลลัพธ์ของการบีบอัดข้อมูลวิธี AWWBzip2 มีขนาดแตกต่างกับการบีบอัดข้อมูลวิธีอื่นๆ