# บทที่ 5

# การทดสอบและผลการทดสอบ

ในบทนี้จะนำเสนอการทดสอบความถูกต้องในการทำงานของส่วนประกอบสำหรับจำลองการทำงานของเว็บแคช โดยเปรียบเทียบจากบทความของงานวิจัยที่เกี่ยวข้องกับขั้นตอนวิธีการนั้นๆ และทดสอบการทำงานของส่วนประกอบหลายๆ ส่วนร่วมกัน การทดสอบแบ่งออกเป็น 3 ส่วน คือ การทดสอบขั้นตอนวิธีการแทนที่ การทดสอบความต้องกันของเว็บแคช และการทดสอบสถาปัตยกรรมของการแคช ซึ่งผลการทดสอบต่างๆ และการวิเคราะห์ผลจะรวมอยู่ในบทนี้ด้วย

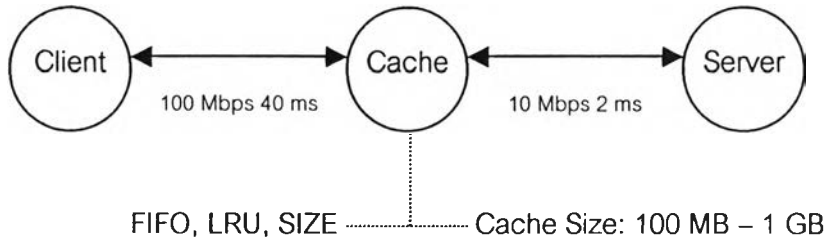
## 5.1 การทดสอบขั้นตอนวิธีการแทนที่

เกณฑ์สำหรับการทดสอบความถูกต้องในการทำงานของขั้นตอนวิธีการแทนที่ คือ การเปรียบเทียบค่าฮิตเรท และค่าไบต์ฮิตเรท โดยค่าฮิตเรทแสดงสัดส่วนของปริมาณการร้องขอที่แคชส่งเอกสารจากแคชไปยังไคลแอนท์ และค่าไบต์ฮิตเรทแสดงสัดส่วนของปริมาณข้อมูลของเอกสารที่แคชส่งเอกสารจากแคชไปยังไคลแอนท์ ซึ่งโดยทั่วไปแล้วขั้นตอนวิธีการแทนที่แบบ SIZE จะให้ค่าฮิตเรทที่สูงที่สุด รองลงมาคือ ขั้นตอนวิธีการแทนที่แบบ LRU และขั้นตอนวิธีการแทนที่แบบ FIFO จะให้ค่าฮิตเรทที่ต่ำที่สุด สำหรับค่าไบต์ฮิตเรท ขั้นตอนวิธีการแทนที่แบบ LRU จะให้ค่าไบต์ฮิตเรทที่สูงที่สุด รองลงมาคือ ขั้นตอนวิธีการแทนที่แบบ FIFO และขั้นตอนวิธีการแทนที่แบบ SIZE จะให้ค่าไบต์ฮิตเรทที่ต่ำที่สุดจนถึงจุดๆ หนึ่งขั้นตอนวิธีการแทนที่แบบ SIZE จะให้ค่าไบต์ฮิตเรทที่มากกว่าขั้นตอนวิธีการแทนที่แบบ FIFO ทั้งนี้ขึ้นอยู่กับปริมาณข้อมูล และขนาดของเว็บแคชด้วย

### 5.1.1 วิธีการทดสอบ และข้อมูลที่ใช้ในการทดสอบ

การทดสอบทำโดยเขียนสคริปต์ไฟล์สำหรับจำลองการทำงานของเว็บแคช ซึ่งประกอบด้วย 1 ไคลแอนท์ 1 แคช และ 1 เซิร์ฟเวอร์ โดยลิงค์ที่เชื่อมต่อระหว่างไคลแอนท์กับแคชเป็นลิงค์แบบสื่อสารสองทาง (Duplex-link) มีแบนด์วิดท์ 100 เมกะบิตต่อวินาที ดีเลย์ 40 มิลลิวินาที ลิงค์ที่เชื่อมต่อระหว่างแคชกับเซิร์ฟเวอร์เป็นลิงค์แบบสื่อสารสองทางมีแบนด์วิดท์ 10 เมกะบิตต่อวินาที ดีเลย์ 2 มิลลิวินาที ใช้ข้อมูลการใช้เว็บจริงวันที่ 21 กันยายน 2539 ของ DEC[12] โดยมีจำนวนการร้องขอ 222,834 ครั้ง ปริมาณข้อมูลทั้งหมด 2,335 MB ขั้นตอนวิธีการแทนที่ที่ใช้ทดสอบมีทั้งหมด 3 วิธี คือ FIFO, LRU และ SIZE ทดสอบโดยการเปลี่ยนแปลงขนาดของแคชจาก 100 เมกะ

ไบต์จนถึง 1 กิกะไบต์ โดยเพิ่มทีละ 100 เมกะไบต์ ดังรูปที่ 5.1 ซึ่งตัวอย่างสคริปต์ที่ใช้ในการทดสอบสามารถดูได้ในภาคผนวก




รูปที่ 5.1 รูปแสดงการเชื่อมต่อในการทดสอบขั้นตอนวิธีการแทนที่

5.1.2 ผลการทดสอบและการวิเคราะห์ผล

จากการทดสอบขั้นตอนวิธีการแทนที่แบบ FIFO, LRU และ SIZE เปรียบเทียบค่าฮิตเรทและค่าไบต์ฮิตเรทกับขนาดของเว็บแคชที่แตกต่างกันตั้งแต่ 100 เมกะไบต์จนถึง 1 กิกะไบต์ โดยเพิ่มทีละ 100 เมกะไบต์ ผลการทดสอบเป็นดังนี้คือ ขั้นตอนวิธีการแทนที่แบบ SIZE จะให้ค่าฮิตเรทสูงที่สุด รองลงมาคือ LRU และ FIFO จะให้ค่าฮิตเรทที่น้อยที่สุด ในส่วนของค่าไบต์ฮิตเรทขั้นตอนวิธีการแทนที่แบบ LRU จะให้ค่าไบต์ฮิตเรทที่สูงสุด รองลงมาคือ FIFO และ SIZE จะให้ค่าไบต์ฮิตเรทที่น้อยที่สุดเมื่อเว็บแคชมีขนาดเล็ก แต่เมื่อเว็บแคชมีขนาดใหญ่ขึ้นค่าไบต์ฮิตเรทของ SIZE จะเพิ่มขึ้นอย่างต่อเนื่องและจะให้ค่าไบต์ฮิตเรทที่สูงกว่า FIFO ในที่สุด ดังตารางที่ 5.1

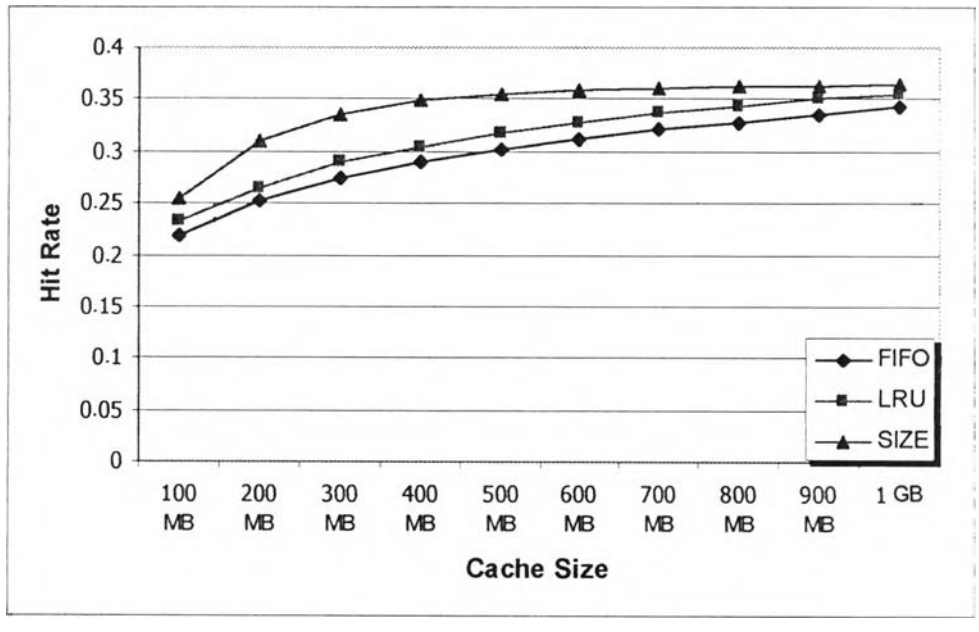
การเปรียบเทียบผลการทดสอบขั้นตอนวิธีการแทนที่นี้เปรียบเทียบกับงานวิจัยที่มีการทดสอบขั้นตอนวิธีการแทนที่ทั้งแบบ FIFO, LRU และ SIZE [15][16] โดยดูจากลักษณะความชันของกราฟ และค่าฮิตเรท ค่าไบต์ฮิตเรทของขั้นตอนวิธีการแทนที่ทั้ง 3 วิธีตามขนาดของเว็บแคชที่แตกต่างกัน ดังรูปที่ 5.2 ถึงรูปที่ 5.7

ตารางที่ 5.1 เปรียบเทียบค่าฮิตเรทและไบต์ฮิตเรทจากการทดสอบขั้นตอนวิธีการแทนที่

| DEC | FIFO | | LRU | | SIZE | |
|---|---|---|---|---|---|---|
| Cache Size | Hit Rate | Byte Hit Rate | Hit Rate | Byte Hit Rate | Hit Rate | Byte Hit Rate |
| 100 MB | 0.218 | 0.183 | 0.232 | 0.191 | 0.254 | 0.091 |
| 200 MB | 0.252 | 0.207 | 0.265 | 0.215 | 0.309 | 0.144 |
| 300 MB | 0.274 | 0.224 | 0.289 | 0.232 | 0.335 | 0.188 |
| 400 MB | 0.289 | 0.235 | 0.304 | 0.244 | 0.349 | 0.219 |
| 500 MB | 0.302 | 0.245 | 0.317 | 0.253 | 0.355 | 0.237 |
| 600 MB | 0.312 | 0.252 | 0.328 | 0.261 | 0.359 | 0.252 |

I20306362

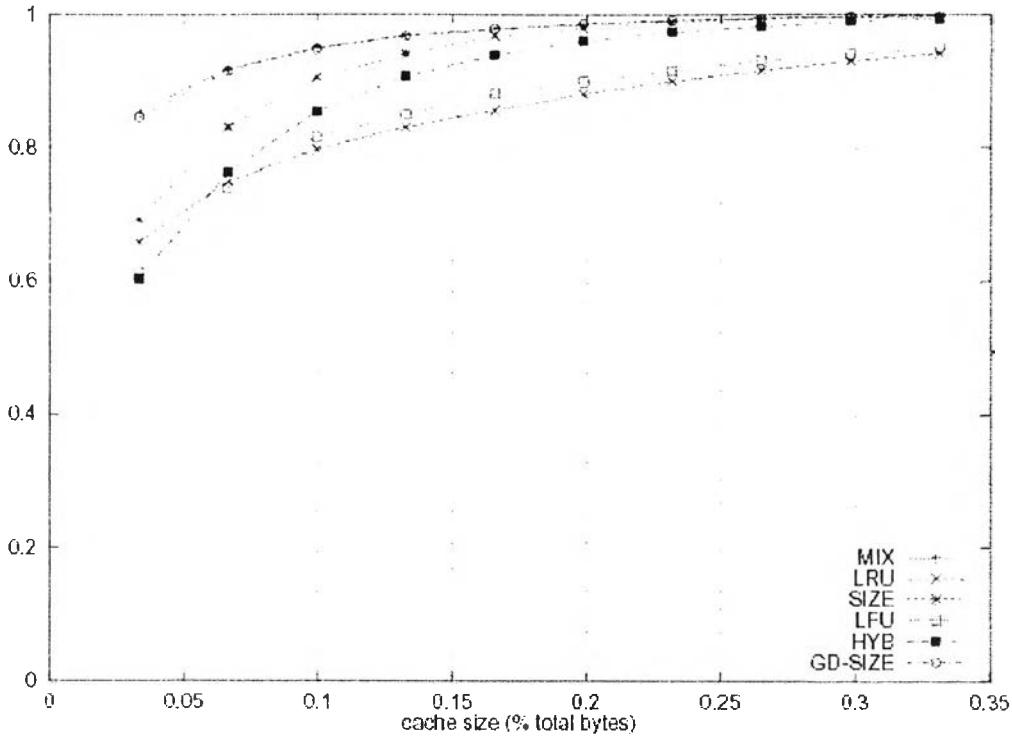
| DEC | FIFO | | LRU | | SIZE | |
|---|---|---|---|---|---|---|
| Cache Size | Hit Rate | Byte Hit Rate | Hit Rate | Byte Hit Rate | Hit Rate | Byte Hit Rate |
| 700 MB | 0.321 | 0.258 | 0.336 | 0.267 | 0.361 | 0.262 |
| 800 MB | 0.328 | 0.263 | 0.343 | 0.272 | 0.363 | 0.270 |
| 900 MB | 0.335 | 0.269 | 0.350 | 0.277 | 0.363 | 0.276 |
| 1 GB | 0.343 | 0.274 | 0.355 | 0.281 | 0.364 | .0.281 |



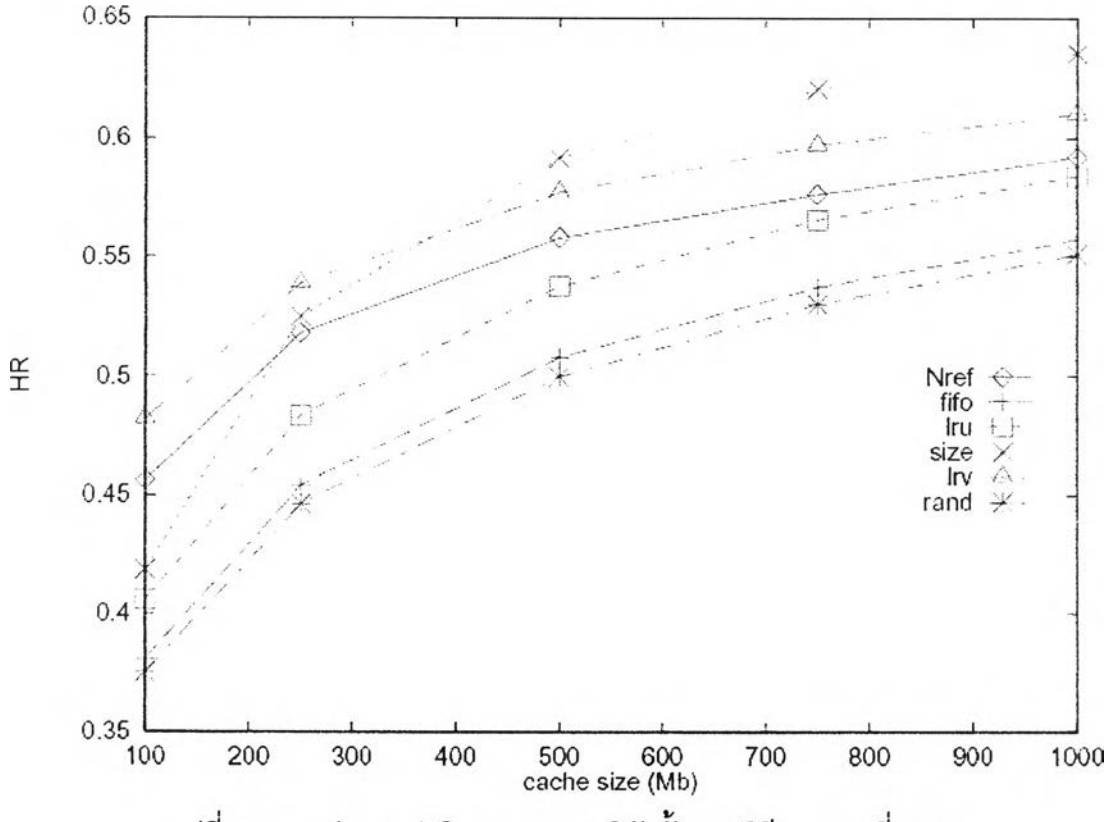

รูปที่ 5.2 กราฟแสดงค่าฮิตเรทจากการทดสอบขั้นตอนวิธีการแทนที่

จากรูปที่ 5.2 แกน X แสดงขนาดของแคชที่ใช้ในการทดสอบ แกน Y แสดงค่าฮิตเรท โดยผลการทดสอบที่ได้เป็นไปดังที่ได้กล่าวมาแล้ว ซึ่งมีผลการทดสอบตามเกณฑ์การทดสอบการทำงานของขั้นตอนวิธีการแทนที่พื้นฐานทั่วไป

สำหรับรูปที่ 5.3 เป็นกราฟแสดงค่าฮิตเรทจากงานวิจัยขั้นตอนวิธีการแทนที่ [15] ในงานวิจัยนี้ได้เสนอขั้นตอนวิธีการแทนที่แบบใหม่ คือ MIX ซึ่งพิจารณาจากความล่าช้าของระบบเครือข่าย ขนาดของเอกสาร ความถี่ในการเรียกขอ และระยะเวลาที่ผ่านไปตั้งแต่มีการอ้างอิงถึงเอกสารภายในแคช ทำการทดสอบโดยจำลองการทำงานของแคชจากข้อมูลการใช้เว็บของ DEC [12], BU, NLANR และ INRIA เปรียบเทียบค่าฮิตเรท ไบต์ฮิตเรท และอัตราความล่าช้า ซึ่งรูปที่ 5.3 เป็นกราฟแสดงค่าฮิตเรทที่ได้จากการทดสอบขั้นตอนวิธีการแทนที่จากข้อมูลการใช้เว็บของ DEC



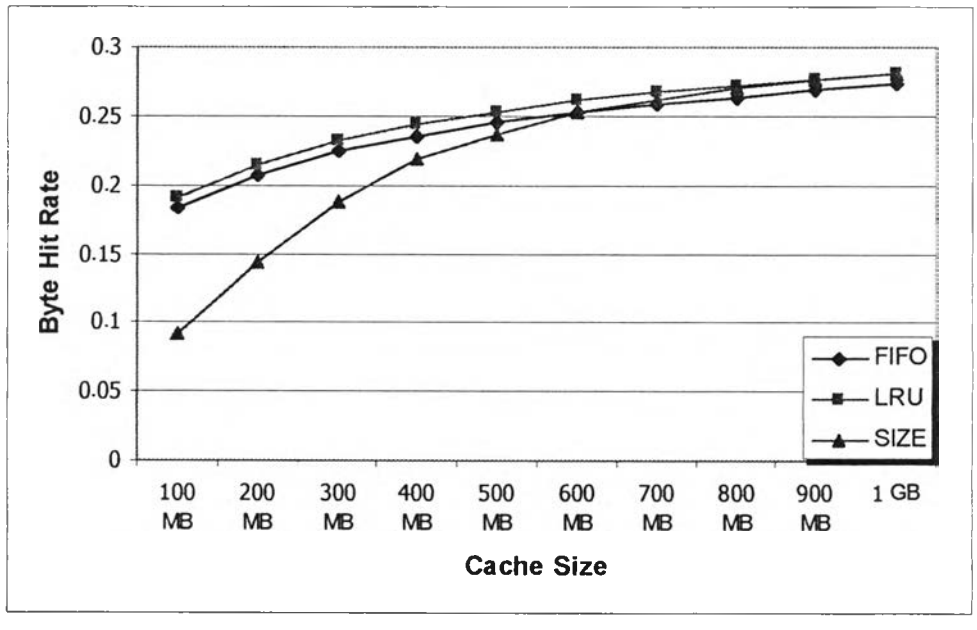

รูปที่ 5.3 กราฟแสดงค่าฮิตเรทจากงานวิจัยขั้นตอนวิธีการแทนที่ [15]




รูปที่ 5.4 กราฟแสดงค่าฮิตเรทจากงานวิจัยขั้นตอนวิธีการแทนที่ [16]

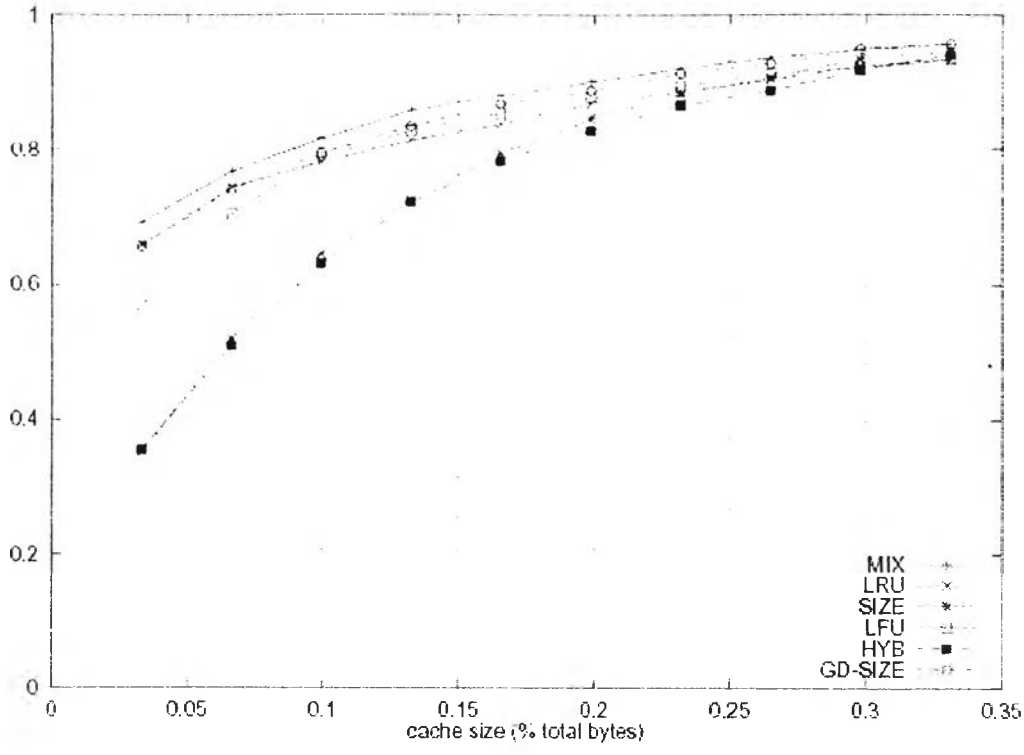
ในส่วนรูปที่ 5.4 เป็นกราฟแสดงค่าฮิตเรทจากงานวิจัยขั้นตอนวิธีการแทนที่ [16] ในงานวิจัยนี้ได้วิเคราะห์ถึงลักษณะการใช้เว็บของเว็บพร็อกซี โดยดูพารามิเตอร์ทางสถิติต่างๆ เพื่อที่จะนำมาออกแบบเป็นขั้นตอนวิธีการแทนที่เอกสารภายในแคช และได้นำเสนอขั้นตอนวิธีการแทนที่แบบใหม่ คือ LRV (Lowest Relative Value) โดยเปรียบเทียบกับขั้นตอนวิธีการแทนที่แบบ LRU และขั้นตอนวิธีการแทนที่แบบอื่นๆ เช่น FIFO และ SIZE

เมื่อเปรียบเทียบรูปที่ 5.2 กับรูปที่ 5.3 และรูปที่ 5.4 จะเห็นว่ากราฟแสดงค่าฮิตเรทของขั้นตอนวิธีการแทนที่ทั้ง 3 วิธีมีแนวโน้มเดียวกันตามผลการทดสอบและเกณฑ์ที่ใช้ในการทดสอบ คือ ขั้นตอนวิธีการแทนที่แบบ SIZE จะให้ค่าฮิตเรทที่สูงที่สุด รองลงมาคือ LRU และ FIFO จะให้ค่าฮิตเรทที่ต่ำที่สุด

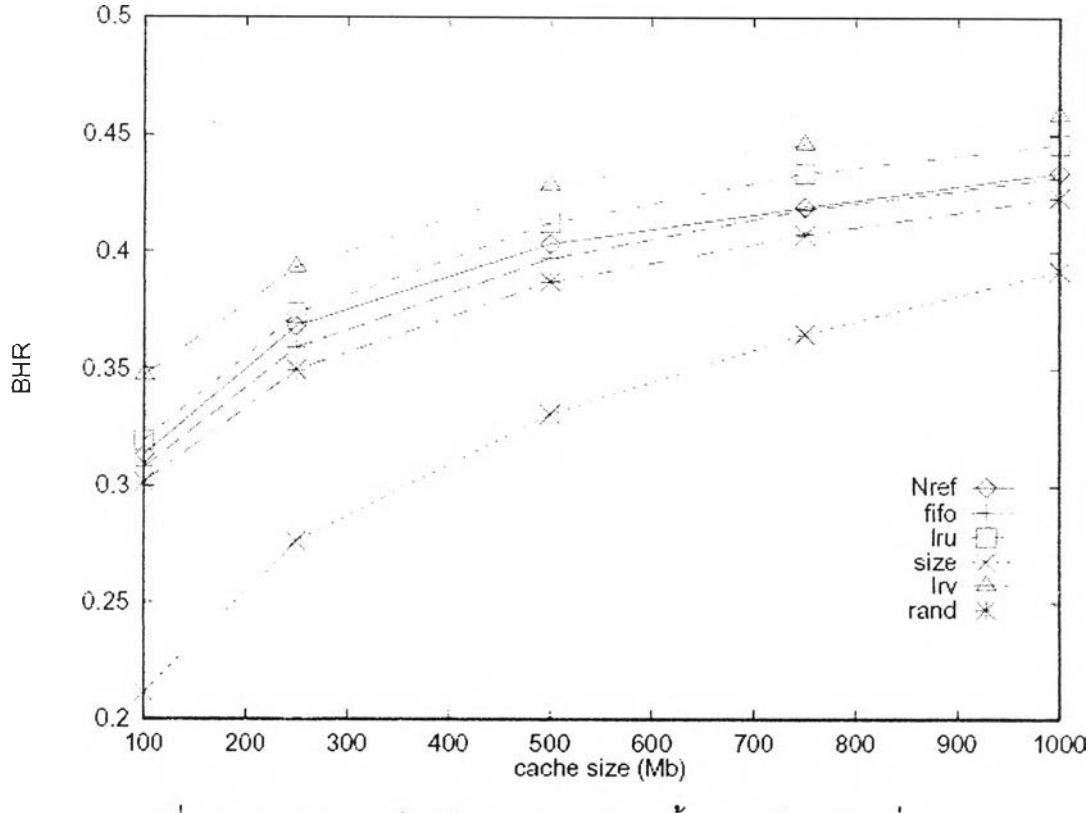



รูปที่ 5.5 กราฟแสดงค่าไบต์ฮิตเรทจากการทดสอบขั้นตอนวิธีการแทนที่

จากรูปที่ 5.5 แกน X แสดงขนาดของแคชที่ใช้ในการทดสอบ แกน Y แสดงค่าไบต์ฮิตเรท โดยผลการทดสอบที่ได้เป็นไปดังที่ได้กล่าวมาแล้ว คือ ขั้นตอนวิธีการแทนที่แบบ LRU จะให้ค่าไบต์ฮิตเรทที่สูงสุด รองลงมาคือ FIFO และขั้นตอนวิธีการแทนที่แบบ SIZE จะให้ค่าไบต์ฮิตเรทที่น้อยที่สุดเมื่อเว็บแคชมีขนาดเล็ก แต่เมื่อเว็บแคชมีขนาดใหญ่ขึ้นค่าไบต์ฮิตเรทของ SIZE จะเพิ่มขึ้นอย่างต่อเนื่องและจะให้ค่าไบต์ฮิตเรทที่สูงกว่า FIFO ในที่สุด ซึ่งมีผลการทดสอบตามเกณฑ์การทดสอบการทำงานของขั้นตอนวิธีการแทนที่พื้นฐานทั่วไป




รูปที่ 5.6 กราฟแสดงค่าไบต์ฮิตเรทจากงานวิจัยขั้นตอนวิธีการแทนที่ [15]




รูปที่ 5.7 กราฟแสดงค่าไบต์ฮิตเรทจากงานวิจัยขั้นตอนวิธีการแทนที่ [16]

สำหรับรูปที่ 5.6 เป็นกราฟแสดงค่าไบต์ฮิตเรทจากงานวิจัยขั้นตอนวิธีการแทนที่ [15] ในงานวิจัยนี้ได้เสนอขั้นตอนวิธีการแทนที่แบบใหม่ คือ MIX ดังรายละเอียดที่ได้กล่าวไปแล้ว ซึ่งรูปที่ 5.6 เป็นกราฟแสดงค่าไบต์ฮิตเรทที่ได้จากการทดสอบขั้นตอนวิธีการแทนที่จากข้อมูลการใช้เว็บของ DEC

ในส่วนรูปที่ 5.7 เป็นกราฟแสดงค่าไบต์ฮิตเรทจากงานวิจัยขั้นตอนวิธีการแทนที่ [16] ในงานวิจัยนี้ได้เสนอขั้นตอนวิธีการแทนที่แบบใหม่ คือ LRV ดังรายละเอียดที่ได้กล่าวไปแล้ว โดยเปรียบเทียบกับขั้นตอนวิธีการแทนที่แบบ LRU และขั้นตอนวิธีการแทนที่แบบอื่นๆ เช่น FIFO และ SIZE

เมื่อเปรียบเทียบรูปที่ 5.5 กับรูปที่ 5.6 และรูปที่ 5.7 ซึ่งเป็นกราฟแสดงค่าไบต์ฮิตเรทของขั้นตอนวิธีการแทนที่ทั้ง 3 วิธี จะเห็นว่ากราฟทั้ง 3 รูปมีแนวโน้มเดียวกันตามผลการทดสอบและเกณฑ์ที่ใช้ในการทดสอบเช่นกัน คือ ขั้นตอนวิธีการแทนที่แบบ LRU จะให้ค่าไบต์ฮิตเรทที่สูงที่สุด รองลงมาคือ FIFO และขั้นตอนวิธีการแทนที่แบบ SIZE จะให้ค่าไบต์ฮิตเรทที่ต่ำที่สุดจนถึงจุดๆ หนึ่งขั้นตอนวิธีการแทนที่แบบ SIZE จะให้ค่าไบต์ฮิตเรทที่มากกว่าขั้นตอนวิธีการแทนที่แบบ FIFO

จากการทดสอบขั้นตอนวิธีการแทนที่ทั้งหมดที่ผ่านมาแสดงให้เห็นว่าเอ็นเอสทูสามารถจำลองการทำงานของเว็บแคชในส่วนขั้นตอนวิธีการแทนที่ทั้ง 3 วิธีได้อย่างถูกต้อง

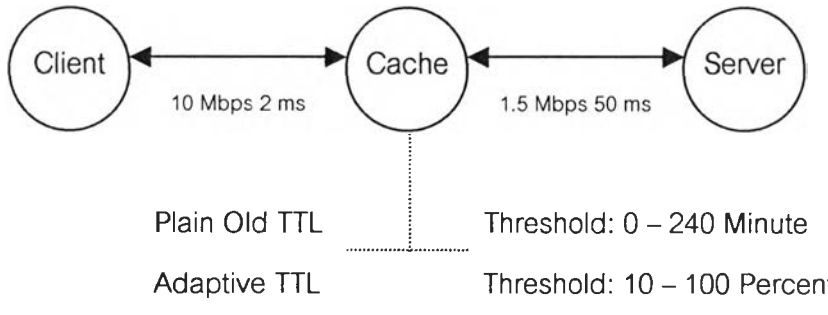
## 5.2 การทดสอบความต้องกันของเว็บแคช

เกณฑ์สำหรับการทดสอบความถูกต้องในการทำงานของความต้องกันของเว็บแคช คือ การเปรียบเทียบแนวโน้มการเปลี่ยนแปลงของค่ามิสเรท (Miss rate) และค่าสเทลฮิตเรท (Stale hit rate) โดยค่ามิสเรทแสดงสัดส่วนของปริมาณการร้องขอที่แคชต้องส่งต่อไปยังเซิร์ฟเวอร์เนื่องจากไม่พบเอกสารที่มีการร้องขอภายในแคชหรือเอกสารที่ถูกร้องขอมีค่า TTL ที่หมดอายุ และค่าสเทลฮิตเรทแสดงสัดส่วนของปริมาณการร้องขอที่ไคลแอนท์ได้รับเอกสารเก่าที่ยังไม่มีการเปลี่ยนแปลงจากแคชทั้งๆ ที่เอกสารมีการเปลี่ยนแปลงแล้วโดยเซิร์ฟเวอร์ ซึ่งโดยทั่วไปแล้ว ค่ามิสเรทของ Plain Old TTL จะต่ำกว่าของ Adaptive TTL และค่าสเทลฮิตเรทของ Plain Old TTL จะสูงกว่าของ Adaptive TTL สำหรับค่าขีดแบ่งในช่วงเดียวกัน

### 5.2.1 วิธีการทดสอบ และข้อมูลที่ใช้ในการทดสอบ

การทดสอบทำโดยเขียนสคริปต์ไฟล์สำหรับจำลองการทำงานของเว็บแคช ซึ่งประกอบด้วย 1 ไคลแอนท์ 1 แคช และ 1 เซิร์ฟเวอร์ โดยลิงค์ที่เชื่อมต่อระหว่างไคลแอนท์กับแคชเป็นลิงค์แบบสื่อสารสองทาง (Duplex-link) มีแบนด์วิดท์ 10 เมกะบิตต่อวินาที ดีเลย์ 2 มิลลิวินาที ลิงค์ที่เชื่อมต่อระหว่างแคชกับเซิร์ฟเวอร์เป็นลิงค์แบบสื่อสารสองทางมีแบนด์วิดท์ 1.5 เมกะบิตต่อวินาที

ดีเลย์ 50 มิลลิวินาที ใช้ข้อมูลการใช้เว็บที่สร้างขึ้น โดยมีจำนวนการร้องขอประมาณ 144,000 ครั้ง โดยใช้การกระจายการร้องขอแบบเอ็กซ์โพเนนเชียล ค่าเฉลี่ยอยู่ที่ 0.1 วินาทีต่อหนึ่งการร้องขอ เวลาสิ้นสุดของการจำลองการทำงานคือ 4 ชั่วโมง (14,400 วินาที) ขนาดของเอกสารกระจายตัวแบบเอ็กซ์โพเนนเชียลโดยมีค่าเฉลี่ย 4096 ไบต์ เลขประจำเอกสารกระจายตัวแบบเอ็กซ์โพเนนเชียลโดยมีค่าเฉลี่ย 300 อายุของเอกสารกระจายตัวแบบเอ็กซ์โพเนนเชียลโดยมีค่าเฉลี่ย 14,400 วินาที ความต้องกันของเว็บแคชที่ใช้ทดสอบมีทั้งหมด 2 วิธี คือ Plain Old TTL(Fixed TTL) และ Adaptive TTL(Alex) สำหรับ Plain Old TTL ทดสอบโดยการเปลี่ยนค่าขีดแบ่ง (Threshold) จาก 0 ถึง 14,400 วินาที (0-240 นาที) เพิ่มขึ้นทีละ 1,440 วินาที (24 นาที) และAdaptive TTL ทดสอบโดยการเปลี่ยนค่าขีดแบ่งจาก 0 ถึง 1 (10-100 เปอร์เซ็นต์) โดยเพิ่มขึ้นทีละ 0.1 ดังรูปที่ 5.8 ซึ่งตัวอย่างสคริปต์ที่ใช้ในการทดสอบสามารถดูได้ในภาคผนวก




รูปที่ 5.8 รูปแสดงการเชื่อมต่อในการทดสอบความต้องกันของเว็บแคช

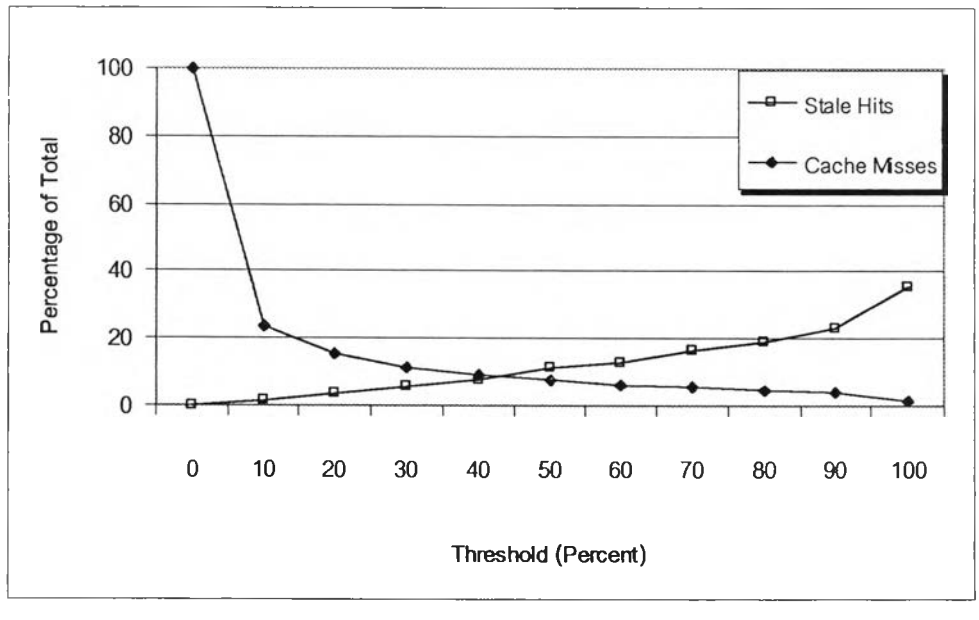
5.2.2 ผลการทดสอบและการวิเคราะห์ผล

จากการทดสอบความต้องกันของเว็บแคชทั้ง 2 วิธี คือ Plain Old TTL และ Adaptive TTL สำหรับ Plain Old TTL ทดสอบโดยการเปลี่ยนค่าขีดแบ่งจาก 0 ถึง 14,400 วินาที (0-240 นาที) เพิ่มขึ้นทีละ 1,440 วินาที (24 นาที) และAdaptive TTL ทดสอบโดยการเปลี่ยนค่าขีดแบ่งจาก 0 ถึง 1 (10-100 เปอร์เซ็นต์) โดยเพิ่มขึ้นทีละ 0.1 ผลการทดสอบเป็นดังนี้คือ ค่ามิสเรทของ Plain Old TTL จะต่ำกว่าของ Adaptive TTL แต่จะเกิดค่าสเทลฮิตเรทที่สูงกว่าสำหรับค่าขีดแบ่งในช่วงเดียวกัน ดังตารางที่ 5.2

การเปรียบเทียบผลการทดสอบความต้องกันของเว็บแคชนี้เปรียบเทียบกับงานวิจัยที่มีการทดสอบความต้องกันของเว็บแคช [13] โดยดูจากลักษณะความชันของกราฟ และค่ามิสเรท ค่าสเทลฮิตเรทของความต้องกันของเว็บแคชทั้ง 2 วิธีตามค่าขีดแบ่งที่แตกต่างกัน ดังรูปที่ 5.9 ถึงรูปที่ 5.12

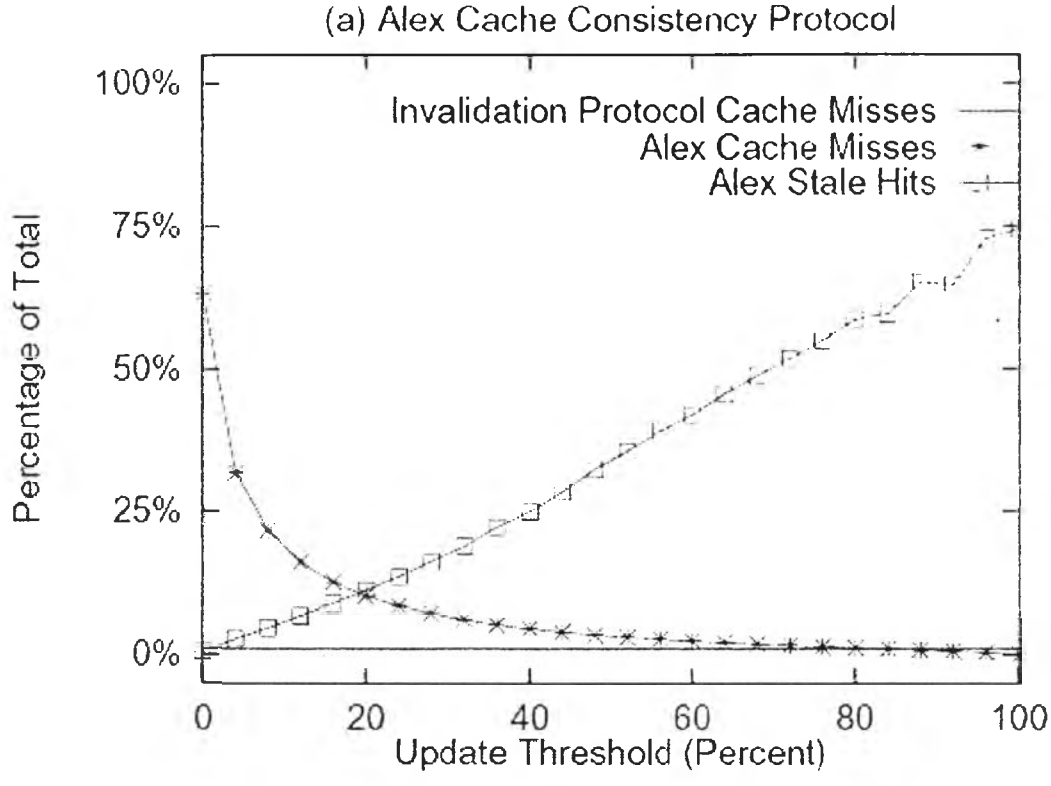
ตารางที่ 5.2 เปรียบเทียบค่ามิสเรทและสเทลฮิตเรทจากการทดสอบความต้องกันของเว็บแคช

| Adaptive TTL | | | Plain Old TTL | | |
|---|---|---|---|---|---|
| Threshold | Miss Rate | Stale Hit Rate | Threshold | Miss Rate | Stale Hit Rate |
| 0 | 100.00 | 0.00 | 0 | 100.00 | 0.00 |
| 0.1 | 23.77 | 1.59 | 24 | 8.38 | 4.28 |
| 0.2 | 15.28 | 3.69 | 48 | 4.96 | 8.57 |
| 0.3 | 11.38 | 5.69 | 72 | 3.94 | 11.98 |
| 0.4 | 9.09 | 7.73 | 96 | 3.22 | 14.99 |
| 0.5 | 7.51 | 11.26 | 120 | 2.44 | 20.38 |
| 0.6 | 6.35 | 12.81 | 144 | 2.39 | 21.10 |
| 0.7 | 5.44 | 16.28 | 168 | 2.32 | 23.23 |
| 0.8 | 4.65 | 18.89 | 192 | 2.24 | 26.19 |
| 0.9 | 3.94 | 23.34 | 216 | 2.10 | 30.52 |
| 1 | 1.40 | 35.65 | 240 | 1.40 | 35.65 |




รูปที่ 5.9 กราฟแสดงค่าจากการทดสอบของ Adaptive TTL

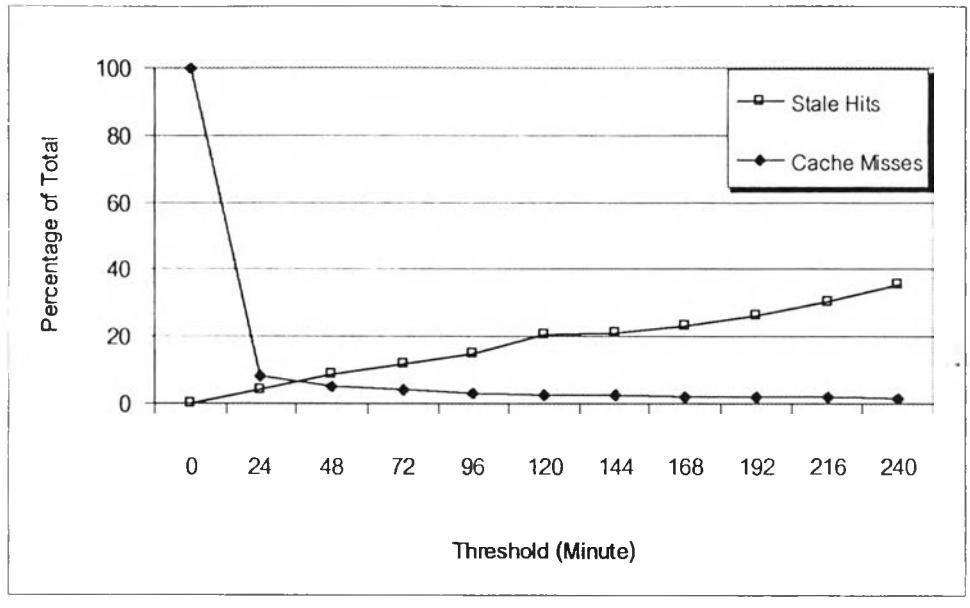
จากรูปที่ 5.9 แกน X แสดงค่าขีดแบ่งที่ใช้ในการทดสอบ แกน Y แสดงค่ามิสเรท และค่าสเทลฮิตเรท โดยผลการทดสอบที่ได้เป็นไปดังที่ได้กล่าวมาแล้ว ซึ่งมีผลการทดสอบตามเกณฑ์การทดสอบการทำงานของความต้องกันของเว็บแคชพื้นฐานทั่วไป



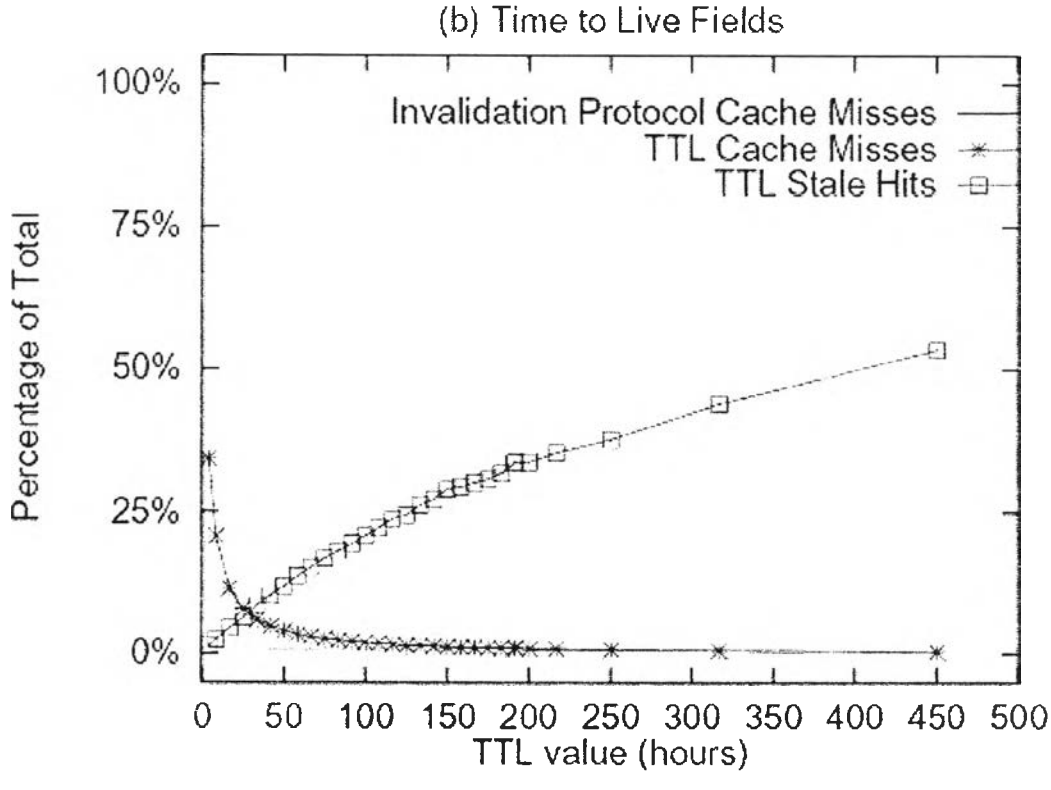

รูปที่ 5.10 กราฟแสดงค่าจากงานวิจัย [13] ของ Adaptive TTL

สำหรับรูปที่ 5.10 เป็นกราฟแสดงค่ามิสเรท และค่าสเทลฮิตเรทจากงานวิจัย [13] ในงานวิจัยนี้ได้เสนอการทดสอบการทำงานของความต้องกันของเว็บแคช 3 แบบคือ Invalidate Protocol, Alex TTL (Adaptive TTL) และ TTL (Fixed TTL) โดยเปรียบเทียบแบนด์วิดท์เซิร์ฟเวอร์โหลด ค่ามิสเรท และค่าสเทลฮิตเรท ที่เกิดขึ้นเมื่อค่าขีดแบ่งมีขนาดแตกต่างกัน ซึ่งรูปที่ 5.10 เป็นกราฟแสดงค่ามิสเรท และค่าสเทลฮิตเรทของ Adaptive TTL

เมื่อเปรียบเทียบรูปที่ 5.9 กับรูปที่ 5.10 ซึ่งเป็นกราฟแสดงค่ามิสเรท และค่าสเทลฮิตเรทของ Adaptive TTL จะเห็นว่ากราฟทั้ง 2 รูป แสดงค่ามิสเรทที่มีแนวโน้มเหมือนกันตามผลการทดสอบ ส่วนค่าสเทลฮิตเรทมีแนวโน้มที่คล้ายคลึงกันโดยค่าสเทลฮิตเรทจะเพิ่มขึ้นอย่างต่อเนื่องเมื่อค่าขีดแบ่งมีค่าเพิ่มขึ้น แต่จะมีความแตกต่างกันของรูปกราฟเพราะระยะเวลาที่ใช้ในการทดสอบการจำลองการทำงาน และปริมาณการร้องขอที่แตกต่างกัน รวมทั้งลักษณะของข้อมูลการใช้เว็บที่ไม่เหมือนกัน ซึ่งทำให้ในการทดสอบเมื่อค่าขีดแบ่งเป็น 0 จะเกิดค่ามิสเรท 100 เปอร์เซ็นต์ ในขณะที่ค่ามิสเรทจากงานวิจัย [13] มีค่าประมาณ 63 เปอร์เซ็นต์เท่านั้น รวมถึงมีข้อจำกัดในการจำลองการทำงานของเอ็นเอสทูซึ่งจะกล่าวถึงในบทสุดท้าย




รูปที่ 5.11 กราฟแสดงค่าจากการทดสอบของ Plain Old TTL




รูปที่ 5.12 กราฟแสดงค่าจากงานวิจัย [13] ของ Plain Old TTL

จากรูปที่ 5.11 แกน X แสดงค่าขีดแบ่งที่ใช้ในการทดสอบ แกน Y แสดงค่ามิสเรท และค่าสเทลฮิตเรท โดยผลการทดสอบที่ได้เป็นไปดังที่ได้กล่าวมาแล้ว คือ ค่ามิสเรทของ Plain Old TTL จะต่ำกว่าของ Adaptive TTL แต่จะเกิดค่าสเทลฮิตเรทที่สูงกว่าสำหรับค่าขีดแบ่งในช่วงเดียวกัน ซึ่งมีผลการทดสอบตามเกณฑ์การทดสอบการทำงานของความต้องกันของเว็บแคชพื้นฐานทั่วไป

สำหรับรูปที่ 5.12 เป็นกราฟแสดงค่ามิสเรท และค่าสเทลฮิตเรทจากงานวิจัย [13] ในงานวิจัยนี้ได้เสนอการทดสอบการทำงานของความต้องกันของเว็บแคช 3 แบบ ดังที่ได้กล่าวมาแล้ว ซึ่งรูปที่ 5.12 เป็นกราฟแสดงค่ามิสเรท และค่าสเทลฮิตเรทของ Plain Old TTL

เมื่อเปรียบเทียบรูปที่ 5.11 กับรูปที่ 5.12 ซึ่งเป็นกราฟแสดงค่ามิสเรท และค่าสเทลฮิตเรทของ Plain Old TTL จะเห็นว่ากราฟทั้ง 2 รูป แสดงค่ามิสเรทที่มีแนวโน้มเหมือนกันตามผลการทดสอบ ส่วนค่าสเทลฮิตเรทมีแนวโน้มที่คล้ายคลึงกันโดยค่าสเทลฮิตเรทจะเพิ่มขึ้นแบบลดลงเมื่อค่าขีดแบ่งมีค่าเพิ่มขึ้น แต่จะมีความแตกต่างกันของรูปกราฟเพราะระยะเวลาที่ใช้ในการทดสอบการจำลองการทำงาน และปริมาณการร้องขอที่แตกต่างกัน รวมทั้งลักษณะของข้อมูลการใช้เว็บที่ไม่เหมือนกัน ซึ่งทำให้ในการทดสอบเมื่อค่าขีดแบ่งเป็น 0 จะเกิดค่ามิสเรท 100 เปอร์เซ็นต์ ในขณะที่ค่ามิสเรทจากงานวิจัย [13] มีค่าประมาณ 37 เปอร์เซ็นต์เท่านั้น รวมถึงมีข้อจำกัดในการจำลองการทำงานของเอ็นเอสทูซึ่งจะกล่าวถึงในบทสุดท้าย

ซึ่งโดยพื้นฐานแล้วจะพบว่าเอ็นเอสทูสามารถจำลองการทำงานได้อย่างถูกต้อง คือ เมื่อค่าขีดแบ่งมีค่าน้อย เอกสารที่มีการร้องขอจะมีค่า TTL ที่หมดอายุอยู่เป็นจำนวนมาก คือ เกิดค่ามิสเรทที่สูง โดยเว็บแคชจะทำการส่งข้อความการร้องขอแบบไอเอ็มเอสไปยังเซิร์ฟเวอร์ปลายทางที่เก็บเอกสารนั้น เพื่อตรวจสอบความถูกต้องของเอกสาร ถ้าเอกสารที่อยู่ในแคชไม่ตรงกับเอกสารที่เก็บอยู่บนเซิร์ฟเวอร์ปลายทาง เซิร์ฟเวอร์ปลายทางจะทำการส่งเอกสารใหม่กลับมายังเว็บแคช และเว็บแคชจะเก็บเอกสารใหม่แทนที่เอกสารเดิม แล้วจัดส่งเอกสารให้กับผู้ร้องขอต่อไป ซึ่งทำให้เกิดค่าสเทลฮิตเรทที่ต่ำ และเมื่อค่าขีดแบ่งมีค่าเพิ่มขึ้น ค่ามิสเรทก็จะลดลง โดยจะเกิดค่าสเทลฮิตเรทที่สูงขึ้น ดังนั้นจะเห็นได้ว่าเอ็นเอสทูสามารถจำลองการทำงานในส่วนความต้องกันของเว็บแคชได้อย่างถูกต้อง

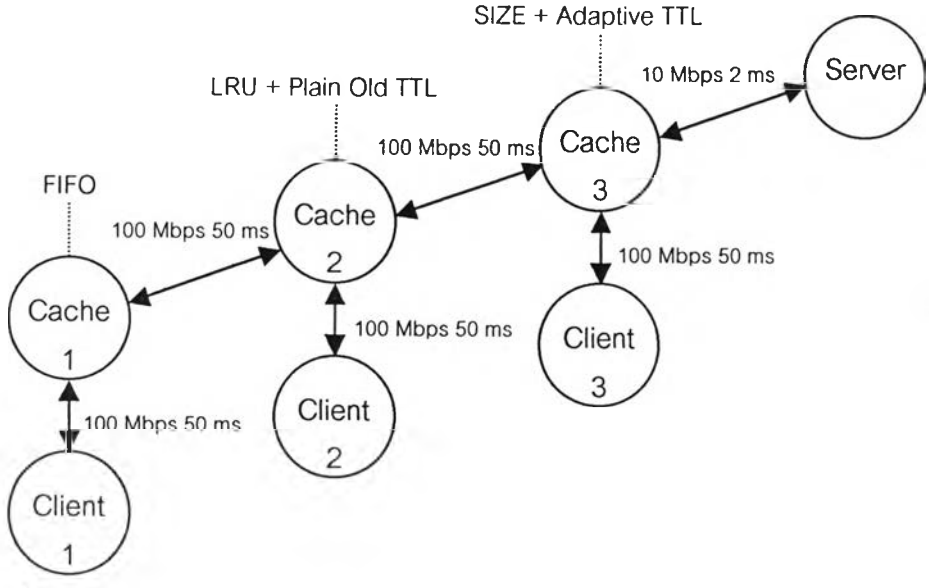
### 5.3 การทดสอบสถาปัตยกรรมของการแคช

การทดสอบในส่วนนี้ทดสอบเพื่อแสดงให้เห็นว่าแคชมากกว่าหนึ่งตัวสามารถทำงานร่วมกันได้ทั้งในแบบพ่อลูก(ลำดับชั้น) และแบบพี่น้อง(ระดับเดียวกัน) แคชแต่ละตัวสามารถใช้ขั้นตอนวิธีการแทนที่ และความต้องกันของเว็บแคชแบบใดก็ได้ โดยเกณฑ์สำหรับการทดสอบความถูกต้องในการทำงานของสถาปัตยกรรมของการแคช คือ การพิจารณาค่าฮิตเรท ค่าไบต์ฮิตเรท จำนวนการร้องขอไปยังเซิร์ฟเวอร์ และความสามารถในการทำงานร่วมกันของส่วนประกอบ

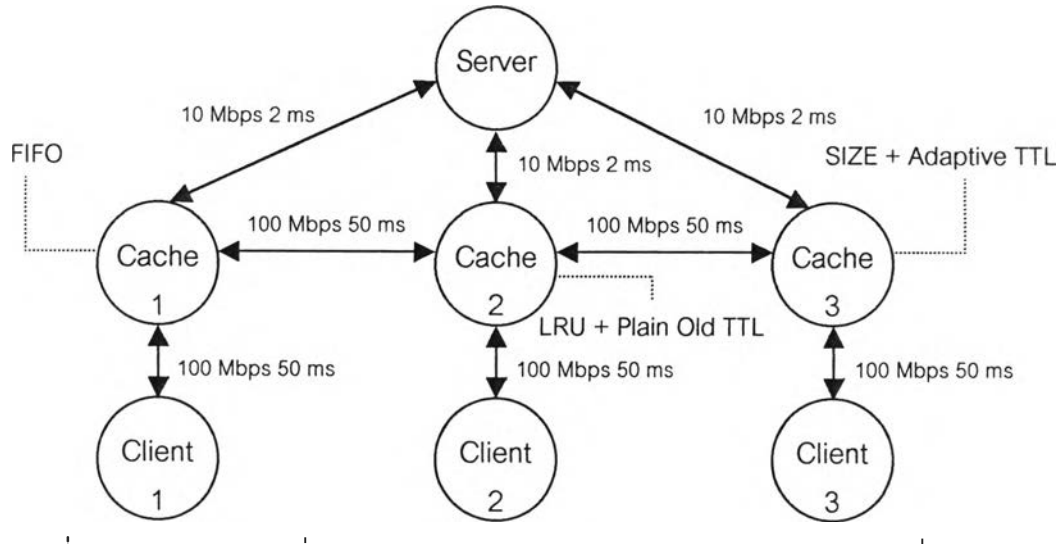
ต่างๆ โดยค่าฮิตเรทแสดงสัดส่วนของปริมาณการร้องขอที่แคชส่งเอกสารจากแคชไปยังไคลแอนท์ และค่าไบต์ฮิตเรทแสดงสัดส่วนของปริมาณข้อมูลของเอกสารที่แคชส่งเอกสารจากแคชไปยังไคลแอนท์ จำนวนการร้องขอไปยังเซิร์ฟเวอร์แสดงปริมาณการร้องขอไปยังเซิร์ฟเวอร์เมื่อไม่พบเอกสารที่ถูกร้องขอภายในแคชหรือเอกสารที่ถูกร้องขอมีค่า TTL ที่หมดอายุ ซึ่งโดยทั่วไปแล้วสถาปัตยกรรมของการแคชแบบพี่น้อง จะให้ค่าฮิตเรท และค่าไบต์ฮิตเรทที่สูงกว่าสถาปัตยกรรมของการแคชแบบพ่อลูก ในขณะเดียวกันก็จะเกิดการร้องขอไปยังเซิร์ฟเวอร์ที่น้อยกว่าสถาปัตยกรรมของการแคชแบบพ่อลูก ทั้งนี้ขึ้นอยู่กับปริมาณข้อมูล ขนาดของเว็บแคช และจำนวนไคลแอนท์ด้วย

5.3.1 วิธีการทดสอบ และข้อมูลที่ใช้ในการทดสอบ

การทดสอบทำโดยเขียนสคริปต์ไฟล์สำหรับจำลองการทำงานของเว็บแคช ซึ่งประกอบ 3 ไคลแอนท์ 3 แคช และ 1 เซิร์ฟเวอร์ โดยลิงค์ที่เชื่อมต่อระหว่างไคลแอนท์กับแคชเป็นลิงค์แบบสื่อสารสองทาง (Duplex-link) มีแบนด์วิดท์ 100 เมกะบิตต่อวินาที ดีเลย์ 50 มิลลิวินาที ลิงค์ที่เชื่อมต่อระหว่างแคชกับแคชเป็นลิงค์แบบสื่อสารสองทางมีแบนด์วิดท์ 100 เมกะบิตต่อวินาที ดีเลย์ 50 มิลลิวินาที และลิงค์ที่เชื่อมต่อระหว่างแคชกับเซิร์ฟเวอร์เป็นลิงค์แบบสื่อสารสองทางมีแบนด์วิดท์ 10 เมกะบิตต่อวินาที ดีเลย์ 2 มิลลิวินาที ใช้ข้อมูลการใช้เว็บจริงจากสำนักเทคโนโลยีสารสนเทศของจุฬาลงกรณ์มหาวิทยาลัย เวลาสิ้นสุดของการจำลองการทำงานคือ 10,000 วินาที โดยมีจำนวนการร้องขอ 124,372 ครั้ง แคชตัวที่ 1 ใช้ขั้นตอนวิธีการแทนที่แบบ FIFO ไม่มีความต้องกันของเว็บแคช แคชตัวที่ 2 ใช้ขั้นตอนวิธีการแทนที่แบบ LRU และความต้องกันของเว็บแคชแบบ Plain Old TTL ค่าขีดแบ่งเท่ากับ 100 วินาที แคชตัวที่ 3 ใช้ขั้นตอนวิธีการแทนที่แบบ SIZE และความต้องกันของเว็บแคชแบบ Adaptive TTL ค่าขีดแบ่งเท่ากับ 0.1 (10 เปอร์เซ็นต์) อายุของเอกสารกระจายตัวแบบเอ็กซ์โพเนนเชียลโดยมีค่าเฉลี่ย 5 วินาที สำหรับเอกสารที่มีการเปลี่ยนแปลงบ่อยและ 1000 วินาทีสำหรับเอกสารในส่วนที่เหลือ ทดสอบโดยการเปลี่ยนแปลงขนาดของแคชจาก 200 เมกะไบต์จนถึง 1 กิกะไบต์ โดยเพิ่มทีละ 200 เมกะไบต์ และเชื่อมต่อแคชใน 2 ลักษณะคือ แบบพ่อลูก และแบบพี่น้อง ดังรูปที่ 5.13 และรูปที่ 5.14 ซึ่งตัวอย่างสคริปต์ที่ใช้ในการทดสอบสามารถดูได้ในภาคผนวก




รูปที่ 5.13 รูปแสดงการเชื่อมต่อในการทดสอบสถาปัตยกรรมของการแคชแบบพ่อลูก




รูปที่ 5.14 รูปแสดงการเชื่อมต่อในการทดสอบสถาปัตยกรรมของการแคชแบบพี่น้อง

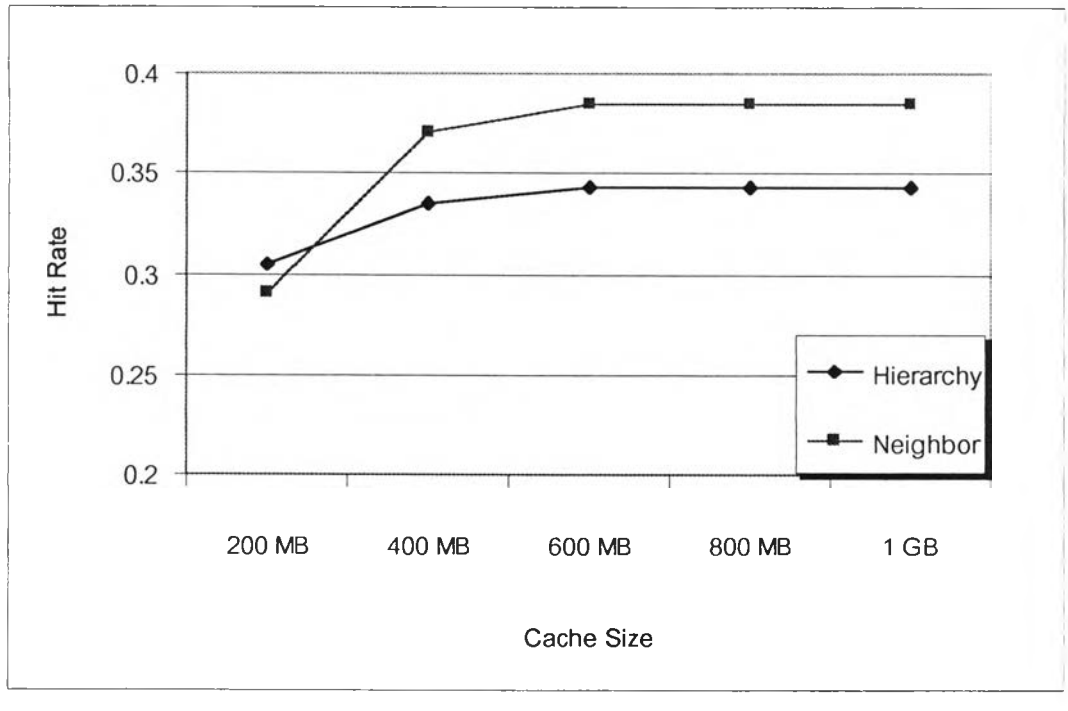
5.3.2 ผลการทดสอบและการวิเคราะห์ผล

จากการทดสอบสถาปัตยกรรมของการแคช ซึ่งใช้แคช 3 ตัว โดยแคชตัวที่ 1 ใช้ขั้นตอนวิธีการแทนที่แบบ FIFO ไม่มีความต้องกันของเว็บแคช แคชตัวที่ 2 ใช้ขั้นตอนวิธีการแทนที่แบบ LRU และความต้องกันของเว็บแคชแบบ Plain Old TTL แคชตัวที่ 3 ใช้ขั้นตอนวิธีการแทนที่แบบ SIZE และความต้องกันของเว็บแคชแบบ Adaptive TTL ทดสอบโดยการเปลี่ยนแปลงขนาดของแคชจาก 200 เมกะไบต์จนถึง 1 กิกะไบต์ โดยเพิ่มทีละ 200 เมกะไบต์ และเชื่อมต่อแคชใน 2 ลักษณะ คือ แบบพ่อลูก และแบบพี่น้อง ผลการทดสอบเป็นดังนี้คือ ค่าฮิตเรทและค่าไบต์ฮิตเรทของ

สถาปัตยกรรมการแคชแบบพี่น้องจะมีค่าที่มากกว่าสถาปัตยกรรมการแคชแบบพ่อลูก และจำนวนการเรียกขอไปยังเซิร์ฟเวอร์ของสถาปัตยกรรมการแคชแบบพี่น้องจะน้อยกว่าสถาปัตยกรรมการแคชแบบพ่อลูก ดังตารางที่ 5.3 และรูปที่ 5.15 ถึงรูปที่ 5.17 ซึ่งเป็นไปตามหลักการพื้นฐานและเกณฑ์การทดสอบการทำงานของสถาปัตยกรรมของการแคช

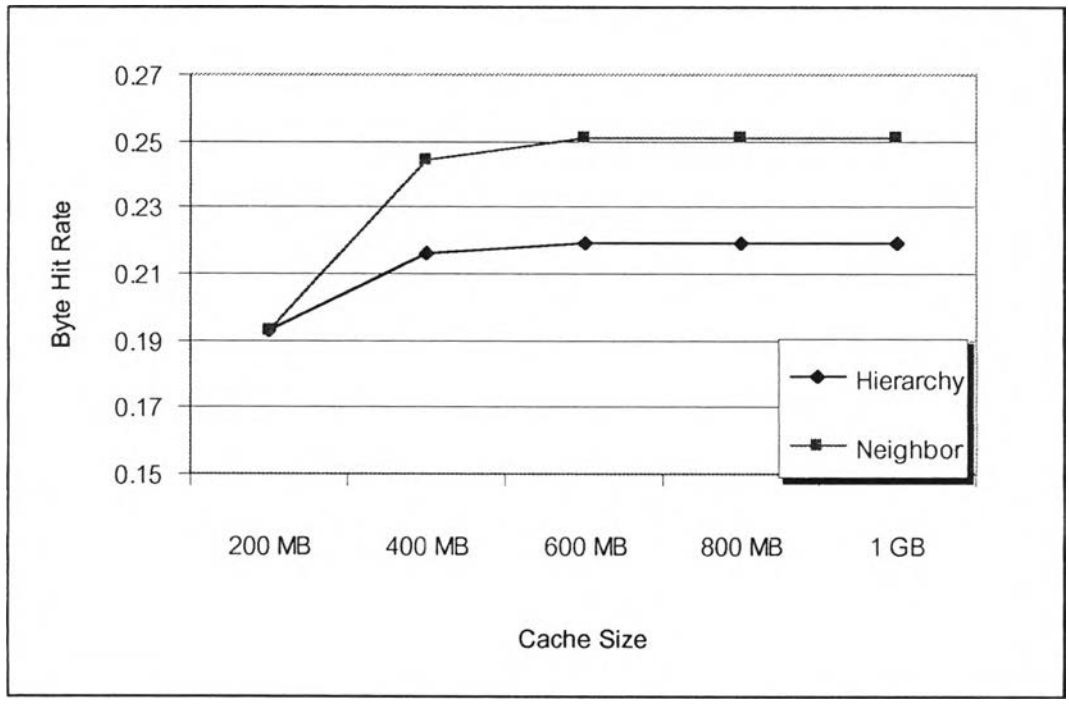
ตารางที่ 5.3 เปรียบเทียบค่าฮิตเรท ไบต์ฮิตเรทและจำนวนการร้องขอไปยังเซิร์ฟเวอร์จากการทดสอบสถาปัตยกรรมของการแคช

| | Hierarchy | | | Neighbor | | |
|---|---|---|---|---|---|---|
| Cache Size | Hit Rate | Byte Hit Rate | Server Request | Hit Rate | Byte Hit Rate | Server Request |
| 200 MB | 0.305 | 0.193 | 98541 | 0.290 | 0.193 | 75071 |
| 400 MB | 0.335 | 0.216 | 96806 | 0.371 | 0.244 | 73404 |
| 600 MB | 0.343 | 0.219 | 96486 | 0.385 | 0.251 | 73336 |
| 800 MB | 0.343 | 0.219 | 96507 | 0.385 | 0.251 | 73336 |
| 1 GB | 0.343 | 0.219 | 96507 | 0.385 | 0.251 | 73336 |



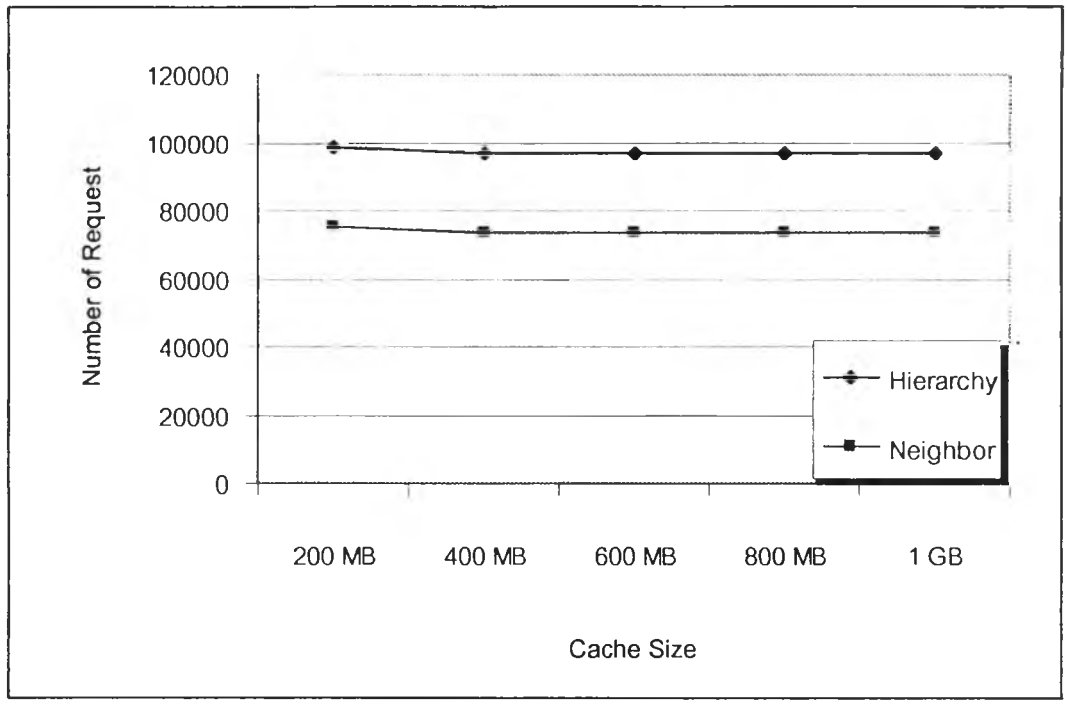

รูปที่ 5.15 กราฟแสดงค่าฮิตเรทจากการทดสอบสถาปัตยกรรมของการแคช

จากรูปที่ 5.15 แกน X แสดงขนาดของแคชที่ใช้ในการทดสอบ แกน Y แสดงค่าฮิตเรท ซึ่งเป็นรูปที่แสดงผลการทดสอบการเปรียบเทียบค่าฮิตเรทระหว่างสถาปัตยกรรมของการแคชแบบพ่อลูกกับสถาปัตยกรรมแบบพี่น้อง โดยทำการเปลี่ยนแปลงขนาดของแคชจาก 200 เมกะไบต์จนถึง 1 กิกะไบต์ โดยเพิ่มทีละ 200 เมกะไบต์ จากกราฟแสดงให้เป็นว่าค่าฮิตเรทของสถาปัตยกรรมการแคชแบบพี่น้องจะมีค่าที่มากกว่าสถาปัตยกรรมการแคชแบบพ่อลูก ซึ่งผลการทดสอบที่ได้เป็นไปตามเกณฑ์การทดสอบการทำงานของสถาปัตยกรรมของการแคชพื้นฐานทั่วไป




รูปที่ 5.16 กราฟแสดงค่าไบต์ฮิตเรทจากการทดสอบสถาปัตยกรรมของการแคช

จากรูปที่ 5.16 แกน X แสดงขนาดของแคชที่ใช้ในการทดสอบ แกน Y แสดงค่าไบต์ฮิตเรท ซึ่งเป็นรูปที่แสดงผลการทดสอบการเปรียบเทียบค่าไบต์ฮิตเรทระหว่างสถาปัตยกรรมของการแคชแบบพ่อลูกกับสถาปัตยกรรมแบบพี่น้อง โดยทำการเปลี่ยนแปลงขนาดของแคชจาก 200 เมกะไบต์จนถึง 1 กิกะไบต์ โดยเพิ่มทีละ 200 เมกะไบต์ จากกราฟแสดงให้เป็นว่าค่าไบต์ฮิตเรทของสถาปัตยกรรมการแคชแบบพี่น้องจะมีค่าที่มากกว่าสถาปัตยกรรมการแคชแบบพ่อลูก ซึ่งผลการทดสอบที่ได้เป็นไปตามเกณฑ์การทดสอบการทำงานของสถาปัตยกรรมของการแคชพื้นฐานทั่วไป




รูปที่ 5.17 กราฟแสดงจำนวนการร้องขอไปยังเซิร์ฟเวอร์จากการทดสอบสถาปัตยกรรมของการแคช

จากรูปที่ 5.17 แกน X แสดงขนาดของแคชที่ใช้ในการทดสอบ แกน Y แสดงจำนวนการร้องขอไปยังเซิร์ฟเวอร์ ซึ่งเป็นรูปที่แสดงผลการทดสอบการเปรียบเทียบจำนวนการร้องขอไปยังเซิร์ฟเวอร์ระหว่างสถาปัตยกรรมของการแคชแบบพ่อลูกกับสถาปัตยกรรมแบบพี่น้อง โดยทำการเปลี่ยนแปลงขนาดของแคชจาก 200 เมกะไบต์จนถึง 1 กิกะไบต์ โดยเพิ่มทีละ 200 เมกะไบต์ จากกราฟแสดงให้เป็นว่าจำนวนการร้องขอไปยังเซิร์ฟเวอร์ของสถาปัตยกรรมการแคชแบบพี่น้องจะมีค่าที่น้อยกว่าสถาปัตยกรรมการแคชแบบพ่อลูก ซึ่งผลการทดสอบที่ได้เป็นไปตามเกณฑ์การทดสอบการทำงานของสถาปัตยกรรมของการแคชพื้นฐานทั่วไป

จากการเปรียบเทียบค่าฮิตเรท ค่าไบต์ฮิตเรท และจำนวนการร้องขอไปยังเซิร์ฟเวอร์ของสถาปัตยกรรมแบบพ่อลูกกับสถาปัตยกรรมแบบพี่น้อง พบว่าเอ็นเอสทูสามารถจำลองการทำงานได้อย่างถูกต้องตามเกณฑ์การทดสอบการทำงานของสถาปัตยกรรมของการแคช นอกจากนี้ยังแสดงให้เห็นว่าในการจำลองการทำงานของเว็บแคช เว็บแคชแต่ละตัวสามารถใช้ขั้นตอนวิธีการต่างๆ ที่แตกต่างกันได้ โดยแคชยังคงสามารถทำงานร่วมกันได้อย่างถูกต้อง