บทที่ 5

การวิเคราะห์และผลการทดลอง

สำหรับบทนี้ก็จะกล่าวถึงค่าใช้จ่ายในการสร้างดัชนีบิตแมปแบบต่าง ๆ โดยวิธีการวิเคราะห์และจากผลการทดลอง โดยจะพิจารณาในเรื่องพื้นที่ที่ใช้ในการจัดเก็บดัชนี เวลาที่ใช้ในการค้นหาข้อมูล และความสัมพันธ์ระหว่างพื้นที่ที่ใช้ในการจัดเก็บดัชนีและเวลาที่ใช้ในการค้นหาข้อมูล

เป็นที่ทราบกันดีอยู่แล้วว่า ในการสร้างดัชนีแบบบิตแมปขึ้นมาเพื่อรองรับการสอบถามข้อมูลนั้น ปัจจัยหนึ่งที่เราจะต้องคำนึงถึงก็คือ ค่าใช้จ่าย โดยในงานวิทยานิพนธ์นี้ค่าใช้จ่ายที่เกิดขึ้นในการสร้างดัชนีบิตแมปแต่ละแบบนั้น สามารถพิจารณาได้ 2 วิธีด้วยกัน คือ จากการวิเคราะห์และจากผลการทดลอง โดยค่าใช้จ่ายที่ได้จากแต่ละวิธีนั้น จะพิจารณาในประเด็นของพื้นที่ที่ใช้ในการจัดเก็บดัชนีและเวลาที่ใช้ในการค้นหาข้อมูล

5.1 ค่าใช้จ่ายจากการวิเคราะห์ (Analytical Method)

ค่าใช้จ่ายที่ใช้ในการสร้างดัชนีบิตแมปแบบต่าง ๆ นั้น สามารถวิเคราะห์ได้เป็น 2 ประเด็น คือ พื้นที่ที่ใช้ในการจัดเก็บดัชนีและเวลาที่ใช้ในการค้นหาข้อมูล

5.1.1 พื้นที่ที่ใช้ในการจัดเก็บดัชนี

กำหนดให้ ตารางเชิงความสัมพันธ์มีจำนวนเรคอร์ดทั้งหมดเท่ากับ N เรคอร์ด แอทริบิวต์ A ที่จะนำมาสร้างดัชนีมีคาร์ดินอลิตี้เท่ากับ C ดังนี้

$T = \{t_0, t_1, \ldots, t_{N-1}\}$

$A = \{a_0, a_1, \ldots, a_{C-1}\}$

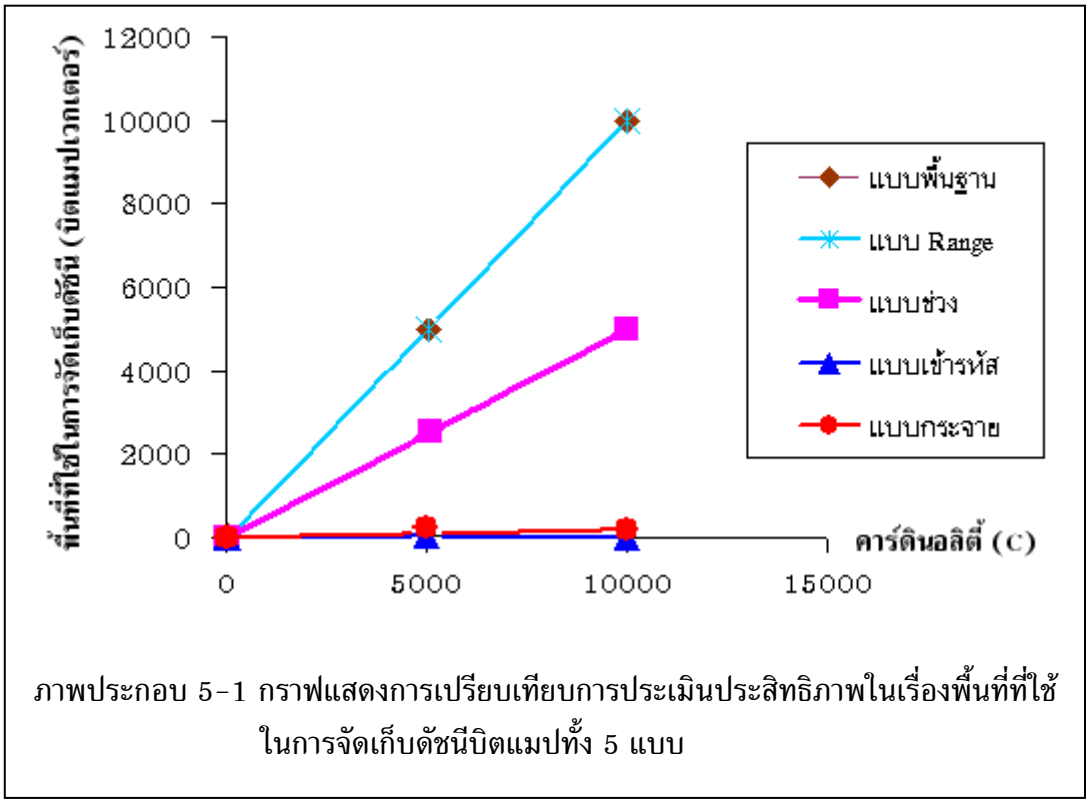
ดังนั้นจะได้ว่า ดัชนีบิตแมปแบบพื้นฐาน แบบ Range แบบช่วง แบบเข้ารหัส และแบบกระจาย จะใช้พื้นที่ในการจัดเก็บดัชนี เท่ากับ CN, $(C-1)N$, $\left\lceil \frac{C}{2} \right\rceil N$, $\lceil \log_2 C \rceil N$ และ $\lceil 2\sqrt{C} \rceil N$ ตามลำดับ ดังตาราง 5-1

ตาราง 5-1 การวิเคราะห์ค่าใช้จ่ายในเรื่องพื้นที่ที่ใช้ในการจัดเก็บดัชนีบิตแมปทั้ง 5 แบบ

แบบพื้นฐาน	แบบ Range	แบบช่วง	แบบเข้ารหัส	แบบกระจาย
CN	$(C-1)N$	$\left\lceil \frac{C}{2} \right\rceil N$	$\lceil \log_2 C \rceil N$	$\lceil 2\sqrt{C} \rceil N$

จากตาราง 5-1 จะเห็นได้ว่า พื้นที่ที่ใช้ในการจัดเก็บดัชนีจะแปรผันโดยตรงกับค่าคาร์ดินอลิตี้ของแอทริบิวต์ที่นำมาสร้างดัชนี เมื่อเปรียบเทียบพื้นที่ที่ใช้ในการจัดเก็บดัชนีบิตแมปทั้ง 5 แบบ จะเห็นได้ว่า มีการใช้พื้นที่เรียงตามลำดับจากมากไปน้อย คือ แบบพื้นฐาน แบบ Range แบบช่วง แบบกระจาย และแบบเข้ารหัส

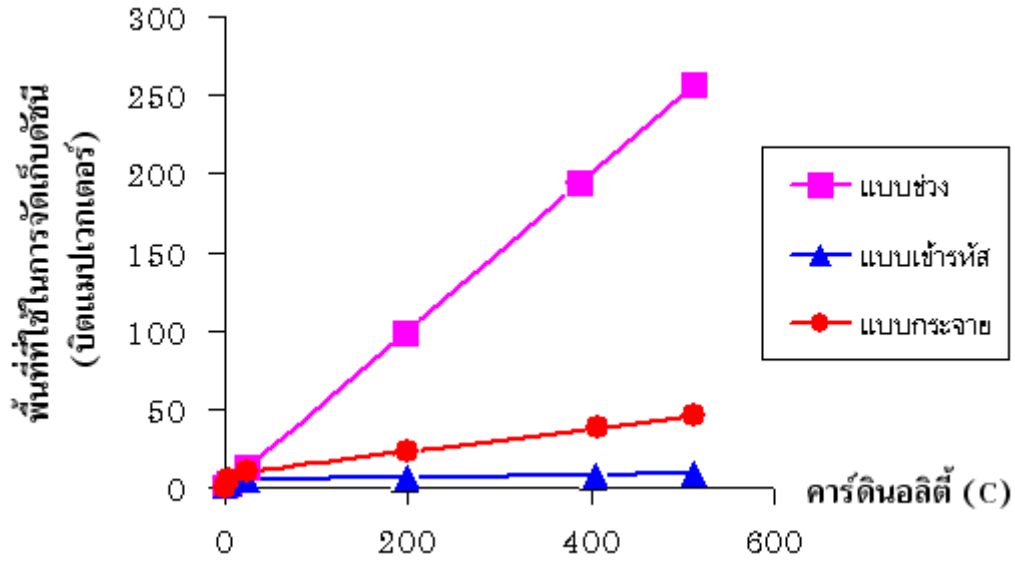
ดังนั้นเพื่อให้เห็นภาพชัดเจนยิ่งขึ้นในการเปรียบเทียบให้เห็นถึงพื้นที่ที่ใช้ในการจัดเก็บดัชนีทั้ง 5 แบบ จึงขอยกตัวอย่างดังภาพประกอบ 5-1



ภาพประกอบ 5-1 กราฟแสดงการเปรียบเทียบการประเมินประสิทธิภาพในเรื่องพื้นที่ที่ใช้ในการจัดเก็บดัชนีบิตแมปทั้ง 5 แบบ

จากภาพประกอบ 5-1 เป็นกราฟที่สร้างขึ้นเพื่อให้เห็นว่า พื้นที่ที่ใช้ในการจัดเก็บดัชนีจะแปรผันโดยตรงกับค่าคาร์ดินอลิตี้ของแอทริบิวต์ที่นำมาสร้างดัชนี ซึ่งค่าคาร์ดินอลิตี้อยู่ในช่วง 0 ถึง 10,000 เมื่อเปรียบเทียบพื้นที่ที่ใช้ในการจัดเก็บดัชนีบิตแมปทั้ง 5 แบบ จะเห็นได้ว่า มีการใช้พื้นที่ในการจัดเก็บดัชนีเรียงตามลำดับจากมากไปน้อย คือ ดัชนีบิตแมปแบบพื้นฐาน แบบ Range แบบช่วง แบบกระจาย และแบบเข้ารหัส ตัวอย่างเช่น แอทริบิวต์ A ที่มีค่า C = 10,000 จะใช้พื้นที่ในการจัดเก็บของดัชนีบิตแมปแบบพื้นฐาน แบบ Range แบบช่วง แบบเข้ารหัส และแบบกระจาย มีค่าเท่ากับ 10000, 9999, 5000, 14 และ 200 บิตแมปเวกเตอร์ตามลำดับ ซึ่งจะสังเกตเห็นได้ว่า ดัชนีบิตแมปแบบช่วง แบบเข้ารหัส และแบบกระจาย จะใช้พื้นที่น้อยมากเมื่อเปรียบเทียบกับแบบพื้นฐานและแบบ Range ดังนั้นจึงขอเน้นจุดความสนใจเฉพาะ

แบบช่วง แบบเข้ารหัส และแบบกระจาย โดยมีค่าคาร์ดินอลิตี้อยู่ในช่วง 0 ถึง 500 ดังภาพประกอบ 5-2



ภาพประกอบ 5-2 กราฟแสดงการเปรียบเทียบการประเมินประสิทธิภาพในเรื่องพื้นที่ที่ใช้ในการจัดเก็บดัชนีบิตแมปแบบช่วง แบบเข้ารหัส และแบบกระจาย

จากภาพประกอบ 5-2 จะเห็นได้ว่า ถ้าแอทริบิวต์ที่นำมาสร้างดัชนีมีค่าคาร์ดินอลิตี้เท่ากับ 500 พื้นที่ที่ใช้ในการจัดเก็บของดัชนีบิตแมปแบบช่วง แบบเข้ารหัส และแบบกระจายจะมีค่าเท่ากับ 250, 9 และ 46 บิตแมปเวกเตอร์ ตามลำดับ ซึ่งเป็นไปตามความคาดหมายกล่าวคือ ดัชนีบิตแมปแบบกระจายใช้พื้นที่ในการจัดเก็บดัชนีอยู่ระหว่างดัชนีบิตแมปแบบช่วงและแบบเข้ารหัส

5.1.2 เวลาที่ใช้ในการค้นหาข้อมูล

วิธีการหนึ่งในการวิเคราะห์เวลาที่ใช้ในการค้นหาข้อมูล คือ พิจารณาจากจำนวนตัวดำเนินการตรรกะกับบิตแมป (Bitmap Operator) และจำนวนบิตแมปที่ถูกอ่าน (Bitmap Scan) ดัชนีบิตแมปแบบพื้นฐานเหมาะสำหรับการสอบถามแบบค่าเท่ากันมากที่สุด เพราะใช้แค่ 1 บิตแมปเวกเตอร์ และไม่มีการดำเนินการตรรกะใด ๆ เกิดขึ้น รองลงมา คือ ดัชนีบิตแมปแบบกระจาย เพราะใช้แค่ 2 บิตแมปเวกเตอร์ และตัวดำเนินการตรรกะเพียง 1 ตัวดำเนินการตรรกะเท่านั้น คือ ตัวดำเนินการตรรกะ AND ส่วนดัชนีบิตแมปแบบช่วงใช้ 2 บิตแมปเวกเตอร์ และตัวดำเนินการตรรกะ 1 หรือ 2 ตัวดำเนินการตรรกะ ส่วนดัชนีบิตแมปแบบ Range ใช้ 2 บิตแมป

เวกเตอร์ และ 5 ตัวดำเนินการตรรกะ แล้วแต่กรณี ดัชนีบิตแมปแบบเข้ารหัส ใช้ $\lceil \log_2 C \rceil$ บิตแมปเวกเตอร์ แล้วใช้การเทียบค่า ดังตาราง 5-2

ตาราง 5-2 การวิเคราะห์เวลาที่ใช้ในการค้นหาข้อมูลแบบค่าเท่ากันของดัชนีบิตแมปทั้ง 5 แบบ

พิจารณา	แบบพื้นฐาน	แบบ Range	แบบช่วง	แบบเข้ารหัส	แบบกระจาย
ตัวดำเนินการตรรกะกับบิตแมป	0	5	2	0 (ใช้การเทียบค่า)	1
จำนวนบิตแมปที่ถูกอ่าน	1	2	2	$\lceil \log_2 C \rceil$	2

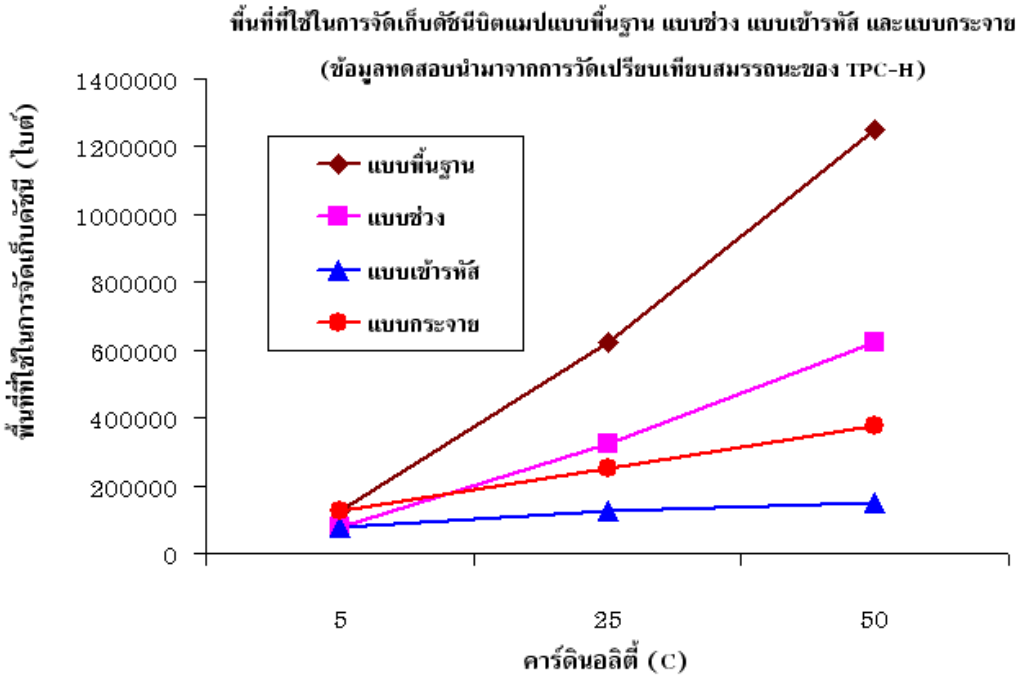
5.2 ค่าใช้จ่ายจากผลการทดลอง (Experimental Method)

นอกจากเราจะสามารถเปรียบเทียบการประเมินประสิทธิภาพ ของการสร้างดัชนีบิตแมปแบบต่าง ๆ ในเรื่องพื้นที่ที่ใช้ในการจัดเก็บดัชนีและเวลาที่ใช้ในการค้นหาข้อมูลแบบค่าเท่ากันโดยวิธีการวิเคราะห์ค่าใช้จ่ายแล้วนั้น เรายังสามารถที่จะเปรียบเทียบการประเมินประสิทธิภาพของการสร้างดัชนีได้อีกวิธีหนึ่ง ซึ่งก็คือ จากผลการทดลอง

สำหรับงานวิทยานิพนธ์ชิ้นนี้ ได้ทำการทดลองกับดัชนีบิตแมปแบบพื้นฐาน แบบช่วง แบบเข้ารหัส และแบบกระจาย โดยทำการประเมินประสิทธิภาพของการสร้างดัชนีในเรื่องพื้นที่ที่ใช้ในการจัดเก็บดัชนีและเวลาที่ใช้ในการค้นหาข้อมูล โดยมีผลการทดลองดังนี้

5.2.1 พื้นที่ที่ใช้ในการจัดเก็บดัชนี

สำหรับงานวิทยานิพนธ์ชิ้นนี้ ได้ทำการทดลองประเมินประสิทธิภาพของดัชนีบิตแมป 4 แบบด้วยกัน คือ ดัชนีบิตแมปแบบพื้นฐาน แบบช่วง แบบเข้ารหัส และแบบกระจาย ทั้งนี้เนื่องจากดัชนีบิตแมปแบบ Range ใช้พื้นที่ในการจัดเก็บดัชนีใกล้เคียงดัชนีบิตแมปแบบพื้นฐานมาก คือ เท่ากับ (C-1) บิตแมปเวกเตอร์ จึงไม่ได้ทำการทดลองดัชนีบิตแมปแบบ Range ซึ่งจากการนำข้อมูลทดสอบจากการวัดเปรียบเทียบสมรรถนะของ TPC-H [34] ตาราง customer แอทริบิวต์ c_mktsegment (ค่า C = 5), ตาราง part แอทริบิวต์ p_brand (ค่า C = 25) และแอทริบิวต์ p_size (ค่า C = 50) จำนวน 200,000 เรคอร์ดมาทดลอง สามารถแสดงพื้นที่ที่ใช้ในการจัดเก็บดัชนี ได้ดังภาพประกอบ 5-3



ภาพประกอบ 5-3 กราฟแสดงพื้นที่ที่ใช้ในการจัดเก็บดัชนีบิตแมปทั้ง 4 แบบ (ข้อมูลทดสอบนำมาจากการวัดเปรียบเทียบสมรรถนะของ TPC-H)

จากภาพประกอบ 5-3 จะสังเกตเห็นได้ว่า

- พื้นที่ที่ใช้ในการจัดเก็บดัชนี จะแปรผันโดยตรงกับค่าคาร์ดินอลิตี้ของแอทริบิวต์ที่นำมาสร้างดัชนี
- ดัชนีบิตแมปแบบช่วง แบบกระจาย และแบบเข้ารหัส จะใช้พื้นที่น้อยมากเมื่อเปรียบเทียบกับแบบพื้นฐาน และดัชนีบิตแมปแบบกระจายใช้พื้นที่อยู่ระหว่างดัชนีบิตแมปแบบช่วงและแบบเข้ารหัส ซึ่งเป็นทำนองเดียวกันกับภาพประกอบ 5-1

5.2.2 เวลาที่ใช้ในการค้นหาข้อมูล

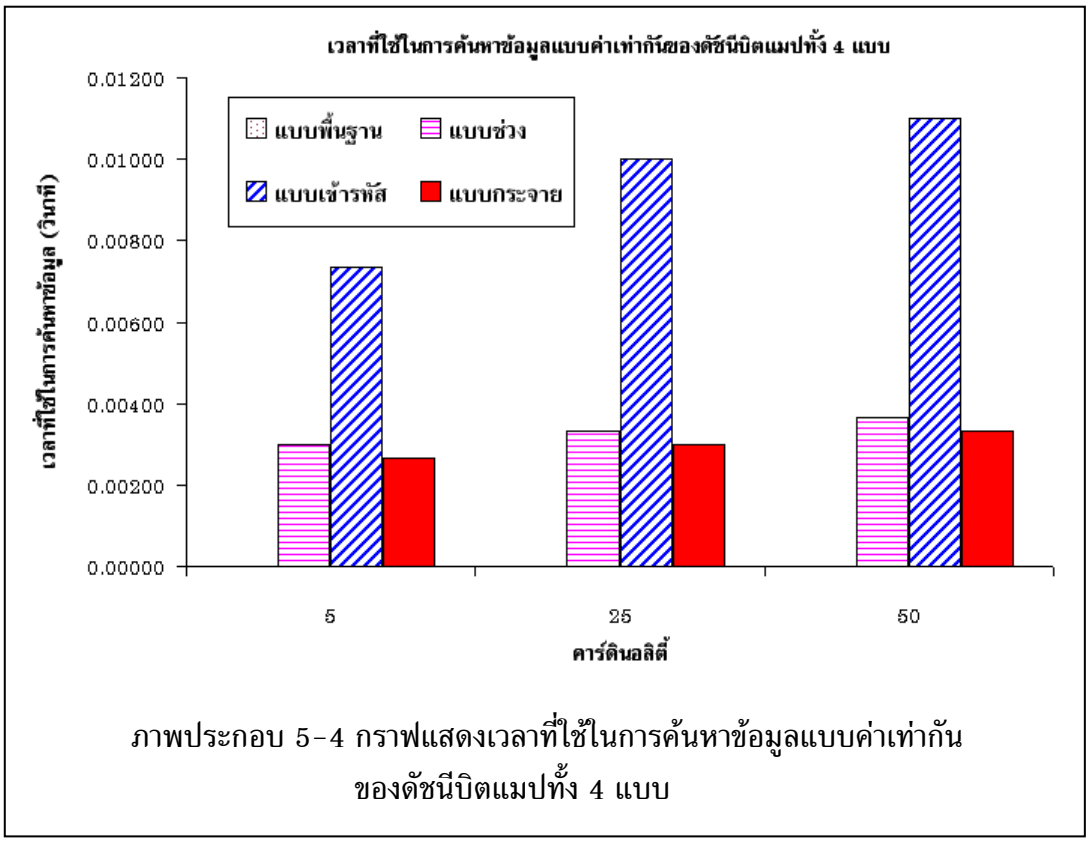
การพิจารณาค่าใช้จ่ายในการสร้างดัชนีบิตแมปแบบต่าง ๆ นั้น นอกจากจะพิจารณาในเรื่องพื้นที่ที่ใช้ในการจัดเก็บดัชนีแล้วนั้น ปัจจัยหนึ่งซึ่งสำคัญที่จะต้องพิจารณาด้วย ก็คือ เวลาที่ใช้ในการค้นหาข้อมูล สำหรับงานวิทยานิพนธ์ชิ้นนี้ได้ทำการทดลองเพื่อประเมินประสิทธิภาพของดัชนีบิตแมปทั้ง 4 แบบ คือ ดัชนีบิตแมปแบบพื้นฐาน แบบช่วง แบบเข้ารหัส และแบบกระจาย ในเรื่องของเวลาที่ใช้ในการค้นหาข้อมูล โดยแบ่งเป็น 2 กรณี คือ

- การค้นหาข้อมูลแบบค่าเท่ากัน
- การค้นหาข้อมูลแบบความเป็นสมาชิก

โดยทำการทดลองบนเครื่องคอมพิวเตอร์ รุ่น Celeron ที่มีหน่วยประมวลผลกลางขนาด 1.69 GHz หน่วยความจำขนาด 384 MB. ได้ผลดังนี้

5.2.2.1 การค้นหาข้อมูลแบบค่าเท่ากัน

จากผลการทดลองกับแอทริบิวต์ที่นำมาสร้างดัชนี ซึ่งมีค่าคาร์ดินอลิตี้เท่ากับ 5, 25 และ 50 สามารถแสดงเวลาที่ใช้ในการค้นหาข้อมูลแบบค่าเท่ากัน ได้ดังภาพประกอบ 5-4



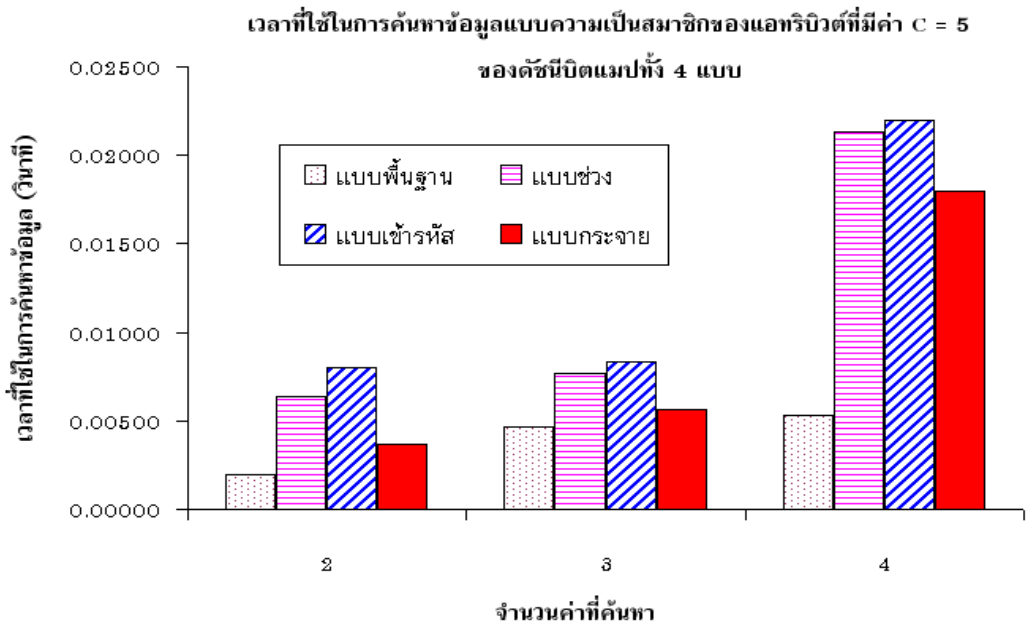
ภาพประกอบ 5-4 กราฟแสดงเวลาที่ใช้ในการค้นหาข้อมูลแบบค่าเท่ากัน ของดัชนีบิตแมปทั้ง 4 แบบ

จากผลการทดลองที่แสดงดังภาพประกอบ 5-4 จะเห็นได้ว่า ดัชนีบิตแมปแบบพื้นฐานจะเหมาะสำหรับการค้นหาข้อมูลแบบค่าเท่ากันมากที่สุด จะใช้เวลาน้อยมากจนไม่สามารถแสดงออกมาให้เห็นเป็นแท่งกราฟได้ รองลงมาคือดัชนีบิตแมปแบบกระจาย ซึ่งสอดคล้องกับตาราง 5-2 ซึ่งแสดงการวิเคราะห์ค่าใช้จ่ายในเรื่องเวลาที่ใช้ในการค้นหาข้อมูลแบบค่าเท่ากัน

5.2.2.2 การค้นหาข้อมูลแบบความเป็นสมาชิก

งานวิทยานิพนธ์นี้ ได้ทำการทดลองการค้นหาข้อมูลแบบความเป็นสมาชิกกับแอทริบิวต์ที่นำมาสร้างดัชนี ซึ่งมีค่าคาร์ดินอลิตี้เท่ากับ 5, 25 และ 50

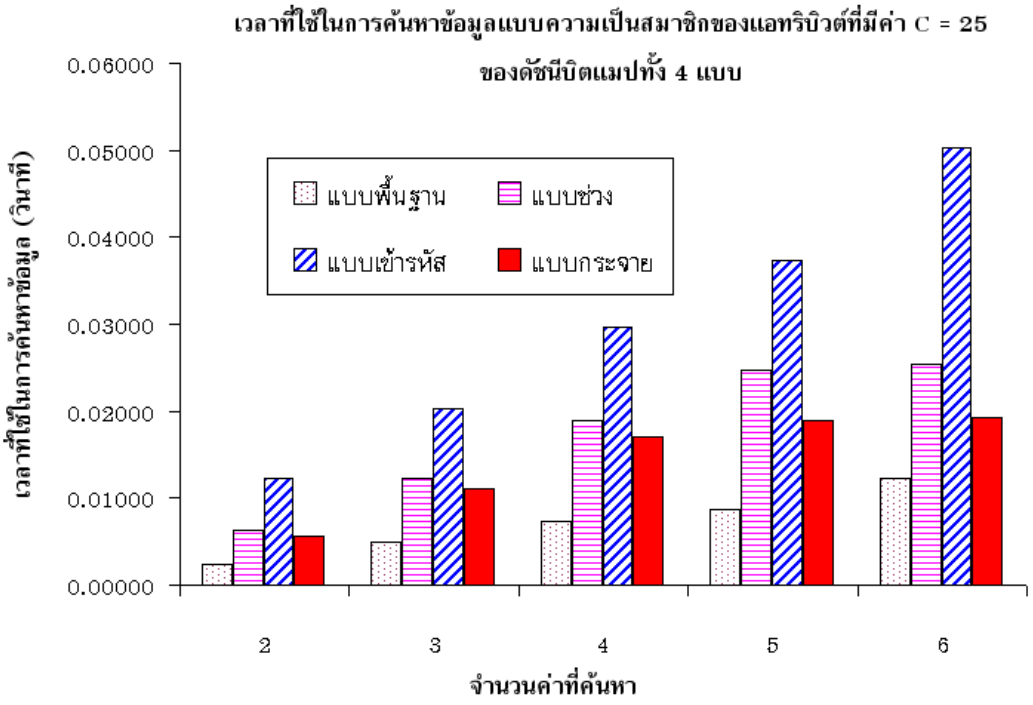
จากผลการทดลองกับแอทริบิวต์ที่นำมาสร้างดัชนี ซึ่งมีค่าคาร์ดินอลิตี้ เท่ากับ 5 สามารถแสดงเวลาที่ใช้ในการค้นหาข้อมูลแบบความเป็นสมาชิก โดยจำนวนค่าที่ค้นหา เท่ากับ 2, 3 และ 4 ค่า ตามลำดับ ได้ดังภาพประกอบ 5-5



ภาพประกอบ 5-5 กราฟแสดงเวลาที่ใช้ในการค้นหาข้อมูลแบบความเป็นสมาชิก ของแอทริบิวต์ที่มีค่า C = 5 ของดัชนีบิตแมปทั้ง 4 แบบ

จากภาพประกอบ 5-5 จะเห็นได้ว่า ดัชนีบิตแมปแบบพื้นฐานจะเหมาะสำหรับการค้นหาข้อมูลแบบความเป็นสมาชิกมากที่สุด เพราะใช้เวลาน้อยที่สุดในการค้นหาข้อมูล รองลงมา คือ ดัชนีบิตแมปแบบกระจาย เพราะใช้เวลาน้อยรองลงมาจากดัชนีบิตแมปแบบพื้นฐานในการค้นหาข้อมูล สำหรับดัชนีบิตแมปแบบเข้ารหัส จะใช้เวลามากที่สุดในการค้นหาข้อมูลแบบความเป็นสมาชิก ทั้งในกรณีที่จำนวนค่าที่ค้นหา เท่ากับ 2, 3 และ 4 ค่า

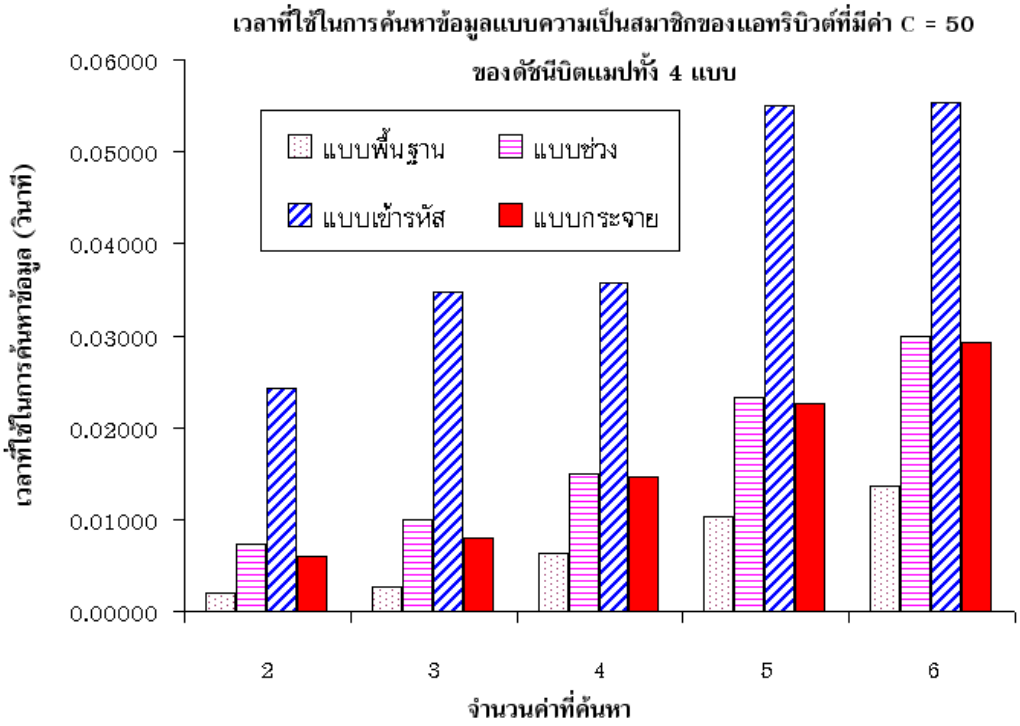
จากผลการทดลองกับแอทริบิวต์ที่นำมาสร้างดัชนี ซึ่งมีค่าคาร์ดินอลิตี้ เท่ากับ 25 สามารถแสดงเวลาที่ใช้ในการค้นหาข้อมูลแบบความเป็นสมาชิก โดยจำนวนค่าที่ค้นหา เท่ากับ 2, 3, 4, 5 และ 6 ค่า ตามลำดับ ได้ดังภาพประกอบ 5-6



ภาพประกอบ 5-6 กราฟแสดงเวลาที่ใช้ในการค้นหาข้อมูลแบบความเป็นสมาชิก ของแอทริบิวต์ที่มีค่า C = 25 ของดัชนีบิตแมปทั้ง 4 แบบ

จากภาพประกอบ 5-6 จะเห็นได้ว่า ดัชนีบิตแมปแบบพื้นฐานจะเหมาะสำหรับการค้นหาข้อมูลแบบความเป็นสมาชิกมากที่สุด เพราะใช้เวลาน้อยที่สุดในการค้นหาข้อมูล รองลงมา คือ ดัชนีบิตแมปแบบกระจาย เพราะใช้เวลาน้อยรองลงมาจากดัชนีบิตแมปแบบพื้นฐานในการค้นหาข้อมูล สำหรับดัชนีบิตแมปแบบเข้ารหัส จะใช้เวลามากที่สุดในการค้นหาข้อมูลแบบความเป็นสมาชิก ทั้งในกรณีที่จำนวนค่าที่ค้นหา เท่ากับ 2, 3, 4, 5 และ 6 ค่า

จากผลการทดลองกับแอทริบิวต์ที่นำมาสร้างดัชนี ซึ่งมีค่าคาร์ดินอลิตี้ เท่ากับ 50 สามารถแสดงเวลาที่ใช้ในการค้นหาข้อมูลแบบความเป็นสมาชิก โดยจำนวนค่าที่ค้นหา เท่ากับ 2, 3, 4, 5 และ 6 ค่า ตามลำดับ ได้ดังภาพประกอบ 5-7



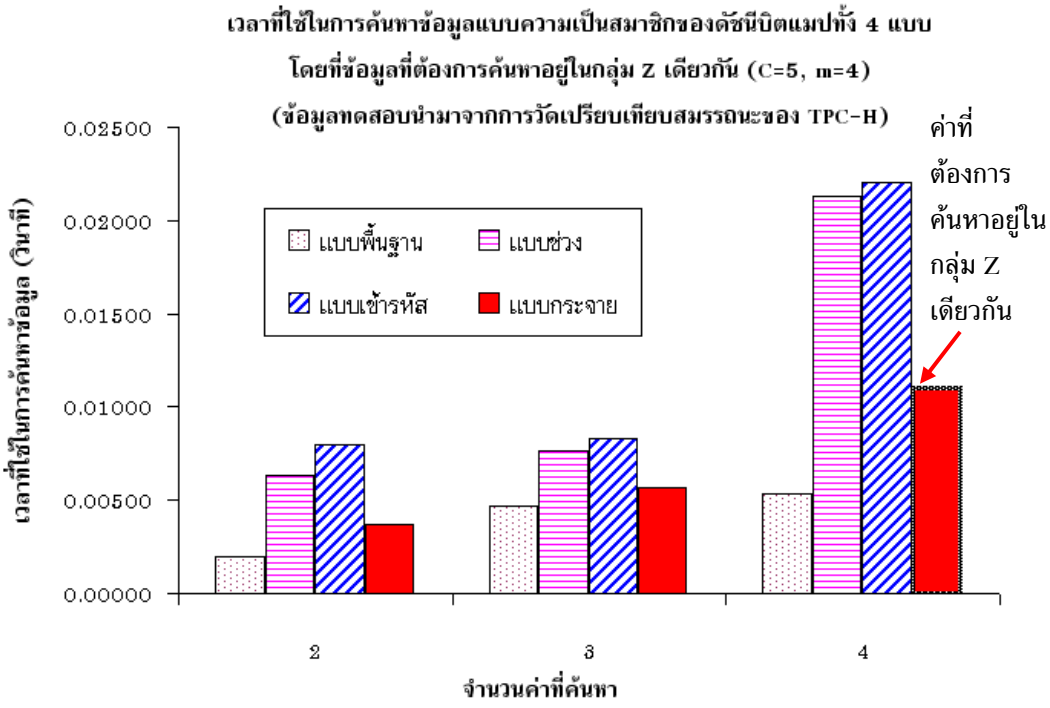
ภาพประกอบ 5-7 กราฟแสดงเวลาที่ใช้ในการค้นหาข้อมูลแบบความเป็นสมาชิก ของแอทริบิวต์ที่มีค่า C = 50 ของดัชนีบิตแมปทั้ง 4 แบบ

จากภาพประกอบ 5-7 จะเห็นได้ว่า ดัชนีบิตแมปแบบพื้นฐานจะเหมาะสำหรับการค้นหาข้อมูลแบบความเป็นสมาชิกมากที่สุด เพราะใช้เวลาน้อยที่สุดในการค้นหาข้อมูล รองลงมา คือ ดัชนีบิตแมปแบบกระจาย เพราะใช้เวลาน้อยรองลงมาจากดัชนีบิตแมปแบบพื้นฐานในการค้นหาข้อมูล สำหรับดัชนีบิตแมปแบบเข้ารหัส จะใช้เวลามากที่สุดในการค้นหาข้อมูลแบบความเป็นสมาชิก ทั้งในกรณีที่จำนวนค่าที่ค้นหา เท่ากับ 2, 3, 4, 5 และ 6 ค่า

จากผลการทดลองข้างต้น ดังภาพประกอบ 5-5, 5-6 และ 5-7 จะเห็นได้ว่า ดัชนีบิตแมปแบบพื้นฐาน จะมีประสิทธิภาพในการค้นหาข้อมูลแบบความเป็นสมาชิกมากที่สุด กล่าวคือ ใช้เวลาน้อยที่สุด เพราะสามารถดำเนินการตรรกะ OR ได้ทันที หลังจากค้นหาบิตแมปเวกเตอร์ของแต่ละค่า แต่ดัชนีบิตแมปแบบช่วง ต้องเสียเวลาในการดำเนินการตรรกะ AND ยกเว้นค่าที่จะค้นหาเท่ากับ C-1 จะดำเนินการตรรกะ OR และ NOT แทน ส่วนดัชนีบิตแมปแบบกระจาย ต้องเสียเวลาในการดำเนินการตรรกะ AND และดัชนีบิตแมปแบบเข้ารหัส จะต้องเสียเวลามากขึ้นในการอ่านค่าที่จะค้นหานั้น จากตารางการเทียบค่าก่อนว่ามีการเข้ารหัสบิตแมปเวกเตอร์ในรูปแบบอะไร แล้วไปอ่านบิตแมปเวกเตอร์ทั้ง $\lceil \log_2 C \rceil$ บิตแมปเวกเตอร์จากตารางดัชนี เพื่อดึงแต่ละค่าที่จะค้นหาก่อนดำเนินการตรรกะ OR

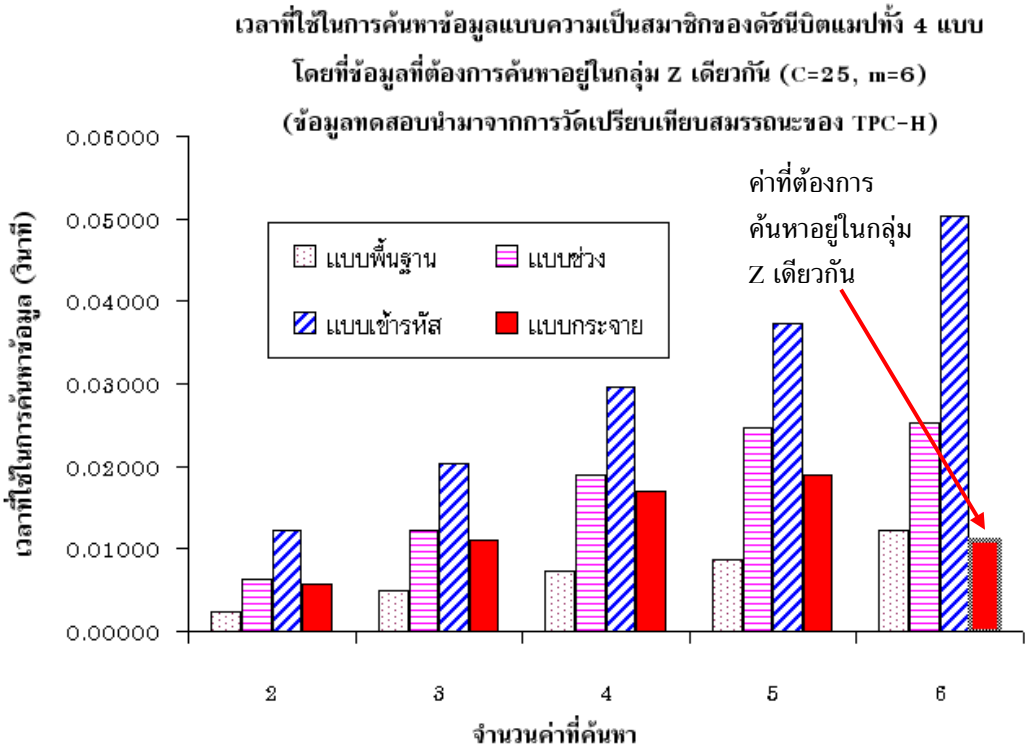
5.2.2.3 การค้นหาข้อมูลแบบความเป็นสมาชิก โดยข้อมูลที่จะค้นหาอยู่ภายในกลุ่ม Z หรือ L เดียวกัน และจำนวนค่าที่ค้นหาเท่ากับจำนวนสมาชิกภายในกลุ่ม Z หรือ L

แต่ถ้าหากค่าที่จะค้นหาเป็นข้อมูลอยู่ภายในกลุ่ม Z หรือ L เดียวกัน และจำนวนค่าที่จะค้นหาเท่ากับจำนวนสมาชิกภายในกลุ่ม Z หรือ L ก็จะทำให้ดัชนีบิตแมปแบบกระจายมีประสิทธิภาพขึ้น (ใช้เวลาน้อยลงมาก) เพราะจะดึงแค่ 1 บิตแมปเวกเตอร์ในการตอบคำถาม ดังภาพประกอบ 5-8 และ 5-9



ภาพประกอบ 5-8 กราฟแสดงเวลาที่ใช้ในการค้นหาข้อมูลแบบความเป็นสมาชิกของดัชนีบิตแมปทั้ง 4 แบบ โดยที่ข้อมูลที่ต้องการค้นหาอยู่ในกลุ่ม Z เดียวกัน (C = 5, m = 4)

จากภาพประกอบ 5-8 จะเห็นได้ว่า เมื่อแอทริบิวต์มีค่าคาร์ดินอลิตี้เท่ากับ 5 จำนวนค่าที่ต้องการค้นหาเท่ากับ 4 และจำนวนสมาชิกภายในกลุ่ม Z เท่ากับ 4 (m = 4) โดยที่ค่าที่ต้องการค้นหานั้นอยู่ภายในกลุ่ม Z เดียวกัน จะใช้เวลาน้อยลงกว่าเดิมมากในการค้นหาข้อมูล กล่าวคือ ใช้เวลาประมาณ 0.011 วินาที แต่จะใช้เวลามากกว่าดัชนีบิตแมปแบบพื้นฐาน (ใช้เวลาประมาณ 0.005 วินาที) ทั้งนี้เพราะดัชนีบิตแมปแบบกระจายจะเสียเวลาในการตรวจสอบว่า ค่าที่ต้องการค้นหาเหล่านั้นอยู่ในกลุ่ม Z เดียวกันหรือไม่ ก่อนที่จะอ่าน 1 บิตแมปเวกเตอร์ ซึ่งการใช้เวลาในการตรวจสอบและการอ่าน 1 บิตแมปเวกเตอร์ของดัชนีบิตแมปแบบกระจายจะใช้เวลามากกว่าดัชนีบิตแมปแบบพื้นฐาน



ภาพประกอบ 5-9 กราฟแสดงเวลาที่ใช้ในการค้นหาข้อมูลแบบความเป็นสมาชิกของดัชนีบิตแมปทั้ง 4 แบบ โดยที่ข้อมูลที่ต้องการค้นหาอยู่ในกลุ่ม Z เดียวกัน (C = 25, m = 6)

จากภาพประกอบ 5-9 จะเห็นได้ว่า เมื่อแอทริบิวต์มีค่าคาร์ดินอลิตี้เท่ากับ 25 จำนวนค่าที่ต้องการค้นหาเท่ากับ 6 และจำนวนสมาชิกภายในกลุ่ม Z เท่ากับ 6 (m = 6) โดยที่ค่าที่ต้องการค้นหานั้นอยู่ภายในกลุ่ม Z เดียวกัน จะใช้เวลาน้อยลงกว่าเดิมมากในการค้นหาข้อมูล กล่าวคือ ใช้เวลาประมาณ 0.011 วินาที และใช้เวลาน้อยกว่าดัชนีบิตแมปแบบพื้นฐาน (ใช้เวลาประมาณ 0.012 วินาที) ด้วย เพราะการใช้เวลาในการตรวจสอบว่าค่าที่ต้องการค้นหาเหล่านั้นอยู่ในกลุ่ม Z เดียวกันหรือไม่และเวลาที่ใช้ในการอ่าน 1 บิตแมปเวกเตอร์ของดัชนีบิตแมปแบบกระจาย น้อยกว่าการใช้เวลาในการอ่านแต่ละบิตแมปเวกเตอร์ของดัชนีบิตแมปแบบพื้นฐานมาดำเนินการตรรกะ OR กัน

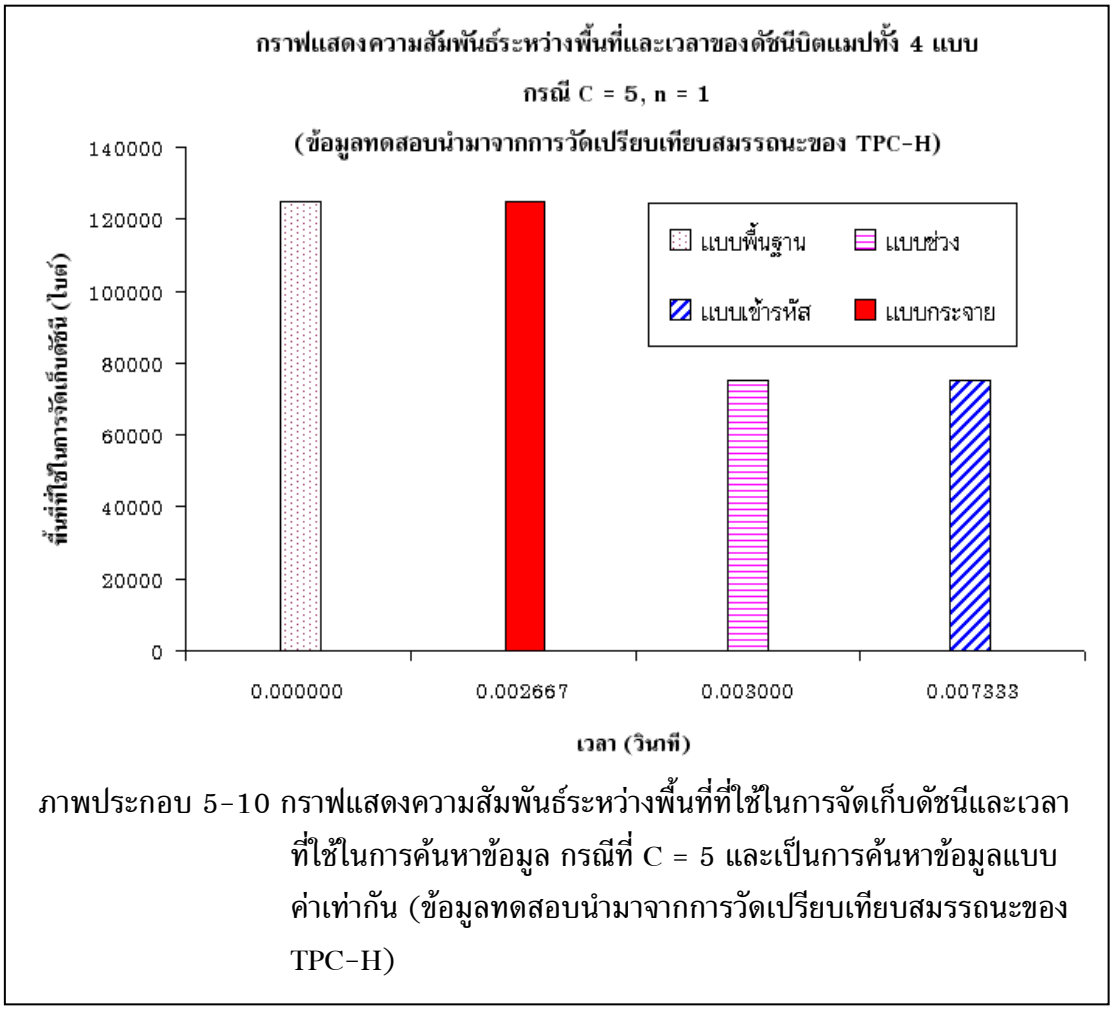
5.2.3 ความสัมพันธ์ระหว่างพื้นที่ที่ใช้ในการจัดเก็บดัชนีและเวลาที่ใช้ในการค้นหาข้อมูล

ในการประเมินเปรียบเทียบประสิทธิภาพของการสร้างดัชนีบิตแมปแบบต่าง ๆ นั้น เราควรพิจารณาค่าใช้จ่ายทั้งในเรื่องพื้นที่ในการจัดเก็บดัชนีและเวลาที่ใช้ในการค้นหาข้อมูลควบคู่กันไป โดยการหาความสัมพันธ์ระหว่างพื้นที่ที่ใช้ในการจัดเก็บดัชนีและเวลาที่ใช้ในการ

ค้นหาข้อมูล ซึ่งในงานวิทยานิพนธ์นี้แบ่งเป็น 2 กรณี คือ กรณีที่เป็นการค้นหาข้อมูลแบบค่าเท่ากันและแบบความเป็นสมาชิก

5.2.3.1 การค้นหาข้อมูลแบบค่าเท่ากัน

ความสัมพันธ์ระหว่างพื้นที่ที่ใช้ในการจัดเก็บดัชนีและเวลาที่ใช้ในการค้นหาข้อมูลแบบค่าเท่ากัน ดังภาพประกอบ 5-10, 5-11 และ 5-12

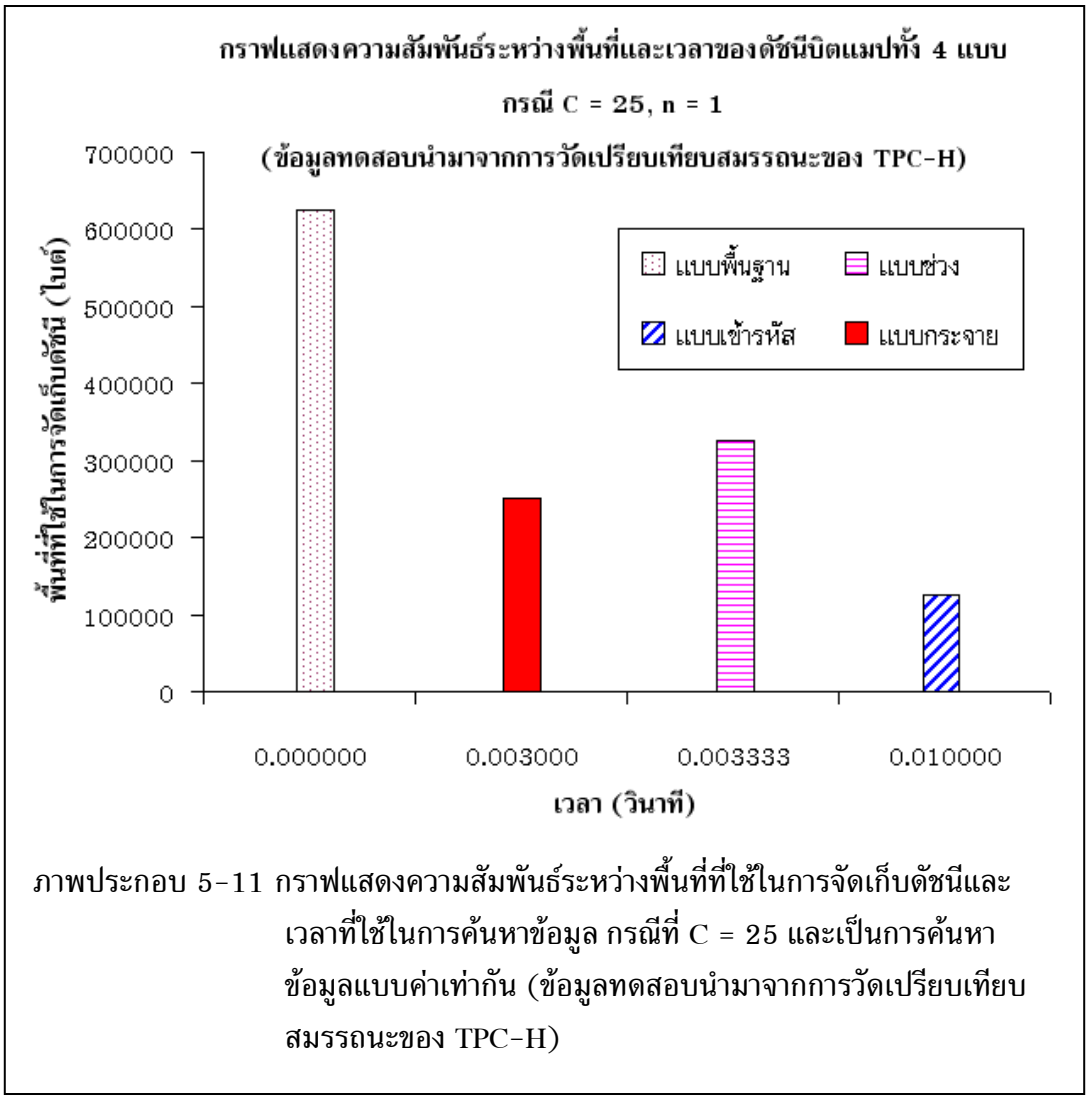


ภาพประกอบ 5-10 กราฟแสดงความสัมพันธ์ระหว่างพื้นที่ที่ใช้ในการจัดเก็บดัชนีและเวลาที่ใช้ในการค้นหาข้อมูล กรณีที่ C = 5 และเป็นการค้นหาข้อมูลแบบค่าเท่ากัน (ข้อมูลทดสอบนำมาจากการวัดเปรียบเทียบสมรรถนะของ TPC-H)

ภาพประกอบ 5-10 เป็นการแสดงความสัมพันธ์ระหว่างพื้นที่ที่ใช้ในการจัดเก็บดัชนีและเวลาที่ใช้ในการค้นหาข้อมูล กรณีที่แอทริบิวต์ที่นำมาสร้างดัชนีมีค่าคาร์ดินอลิตี้เท่ากับ 5 และเป็นการค้นหาแบบค่าเท่ากัน ซึ่งก็คือ จำนวนค่าที่ต้องการค้นหาเท่ากับ 1 (n=1) จะเห็นได้ว่า

- ดัชนีบิตแมปแบบพื้นฐาน และดัชนีบิตแมปแบบกระจาย จะใช้พื้นที่มากที่สุดในการจัดเก็บดัชนี
- ดัชนีบิตแมปแบบช่วง และดัชนีบิตแมปแบบเข้ารหัส จะใช้พื้นที่น้อยที่สุดในการจัดเก็บดัชนี

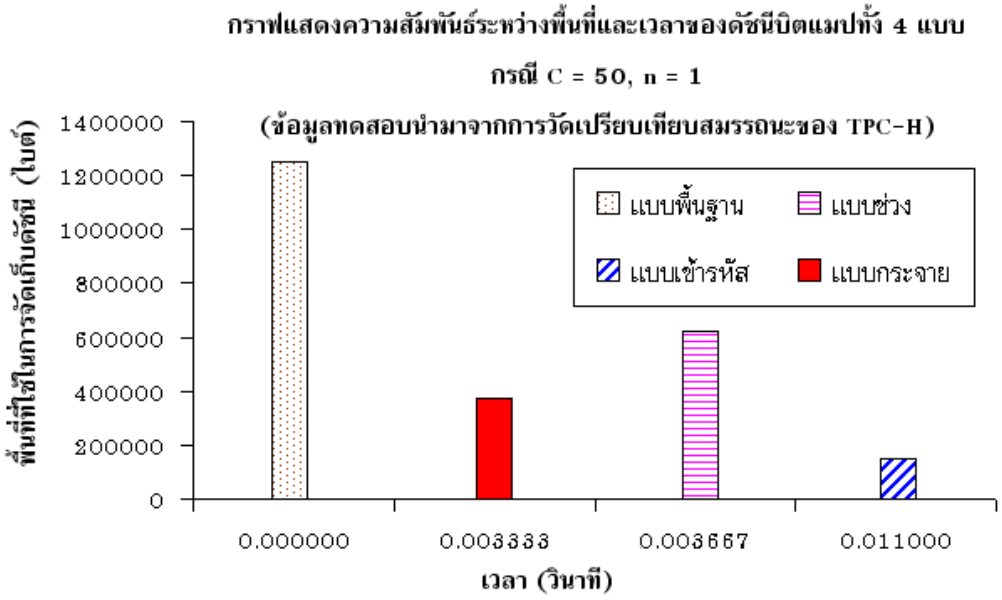
- ในกรณีที่แอทริบิวต์ที่นำมาสร้างดัชนีมีค่าคาร์ดินอลิตี้ที่ต่ำมาก ๆ เช่น ในกรณีนี้ แอทริบิวต์ที่นำมาสร้างดัชนีมีค่าคาร์ดินอลิตี้เท่ากับ 5 จะทำให้ดัชนีบิตแมปแบบกระจายใช้พื้นที่มากกว่าดัชนีบิตแมปแบบช่วง
- ดัชนีบิตแมปแบบพื้นฐาน ใช้เวลาน้อยที่สุดในการค้นหาข้อมูล
- ดัชนีบิตแมปแบบเข้ารหัส ใช้เวลามากที่สุดในการค้นหาข้อมูล
- ดัชนีบิตแมปแบบกระจาย ใช้เวลาน้อยกว่าดัชนีบิตแมปแบบช่วงและแบบเข้ารหัสในการค้นหาข้อมูล



ภาพประกอบ 5-11 กราฟแสดงความสัมพันธ์ระหว่างพื้นที่ที่ใช้ในการจัดเก็บดัชนีและเวลาที่ใช้ในการค้นหาข้อมูล กรณีที่ C = 25 และเป็นการค้นหาข้อมูลแบบค่าเท่ากัน (ข้อมูลทดสอบนำมาจากการวัดเปรียบเทียบสมรรถนะของ TPC-H)

ภาพประกอบ 5-11 และ 5-12 เป็นการแสดงความสัมพันธ์ระหว่างพื้นที่ที่ใช้ในการจัดเก็บดัชนีและเวลาที่ใช้ในการค้นหาข้อมูล กรณีที่แอทริบิวต์ที่นำมาสร้างดัชนีมีค่าคาร์ดินอลิตี้เท่ากับ 25 และ 50 ตามลำดับ และเป็นการค้นหาแบบค่าเท่ากัน ซึ่งก็คือ จำนวนค่าที่ต้องการค้นหาเท่ากับ 1 (n = 1) จะเห็นได้ว่า

- ดัชนีบิตแมปแบบพื้นฐาน จะใช้พื้นที่มากที่สุดในการจัดเก็บดัชนี แต่ใช้เวลาน้อยที่สุดในการค้นหาข้อมูล
- ดัชนีบิตแมปแบบเข้ารหัส จะใช้พื้นที่น้อยที่สุดในการจัดเก็บดัชนี แต่ใช้เวลามากที่สุดในการค้นหาข้อมูล
- ดัชนีบิตแมปแบบกระจาย จะใช้พื้นที่ในการจัดเก็บดัชนีและเวลาในการค้นหาข้อมูลน้อยกว่าดัชนีบิตแมปแบบช่วง



ภาพประกอบ 5-12 กราฟแสดงความสัมพันธ์ระหว่างพื้นที่ที่ใช้ในการจัดเก็บดัชนีและเวลาที่ใช้ในการค้นหาข้อมูลกรณีที่ C = 50 และเป็นการค้นหาข้อมูลแบบค่าเท่ากัน (ข้อมูลทดสอบนำมาจากการวัดเปรียบเทียบสมรรถนะของ TPC-H)

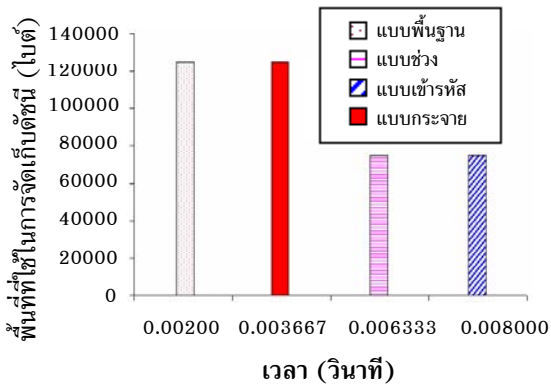
5.2.3.2 การค้นหาข้อมูลแบบความเป็นสมาชิก

ความสัมพันธ์ระหว่างพื้นที่ที่ใช้ในการจัดเก็บดัชนีและเวลาที่ใช้ในการค้นหาข้อมูลแบบความเป็นสมาชิก ดังภาพประกอบ 5-13, 5-14 และ 5-15

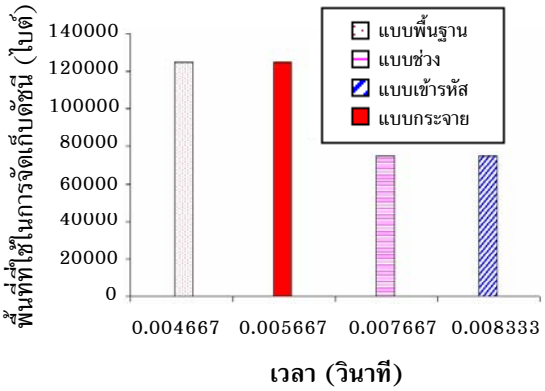
ภาพประกอบ 5-13 เป็นการแสดงความสัมพันธ์ระหว่างพื้นที่ที่ใช้ในการจัดเก็บดัชนีและเวลาที่ใช้ในการค้นหาข้อมูล กรณีที่แอทริบิวต์ที่นำมาสร้างดัชนีมีค่าคาร์ดินอลิตี้เท่ากับ 5 และเป็นการค้นหาแบบความเป็นสมาชิก โดยภาพประกอบ 5-13(a), 5-13(b) และ 5-13(c) เป็นการค้นหาในกรณีที่จำนวนค่าที่ต้องการค้นหาเท่ากับ 2, 3 และ 4 ค่า ตามลำดับ

ภาพประกอบ 5-14 เป็นการแสดงความสัมพันธ์ระหว่างพื้นที่ที่ใช้ในการจัดเก็บดัชนีและเวลาที่ใช้ในการค้นหาข้อมูล กรณีที่แอทริบิวต์ที่นำมาสร้างดัชนีมีค่าคาร์ดินอลิตี้เท่ากับ 25 และเป็นการค้นหาแบบความเป็นสมาชิก โดยภาพประกอบ 5-14(a), 5-14(b), 5-14(c), 5-14(d) และ 5-14(e) เป็นการค้นหาในกรณีที่จำนวนค่าที่ต้องการค้นหาเท่ากับ 2, 3, 4, 5 และ 6 ค่า ตามลำดับ

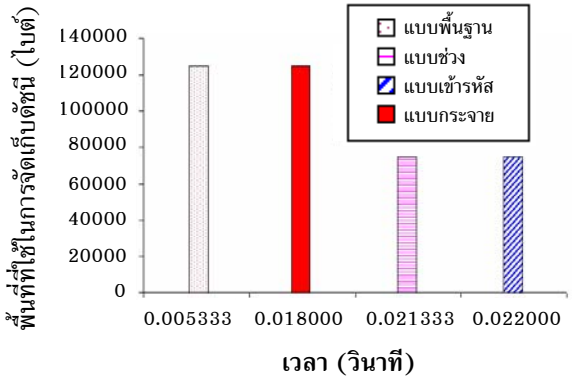
ภาพประกอบ 5-15 เป็นการแสดงความสัมพันธ์ระหว่างพื้นที่ที่ใช้ในการจัดเก็บดัชนีและเวลาที่ใช้ในการค้นหาข้อมูล กรณีที่แอทริบิวต์ที่นำมาสร้างดัชนีมีค่าคาร์ดินอลิตี้เท่ากับ 50 และเป็นการค้นหาแบบความเป็นสมาชิก โดยภาพประกอบ 5-15(a), 5-15(b), 5-15(c), 5-15(d) และ 5-15(e) เป็นการค้นหาในกรณีที่จำนวนค่าที่ต้องการค้นหาเท่ากับ 2, 3, 4, 5 และ 6 ค่า ตามลำดับ



(a) กรณีจำนวนค่าที่ต้องการค้นหา = 2 (n=2)

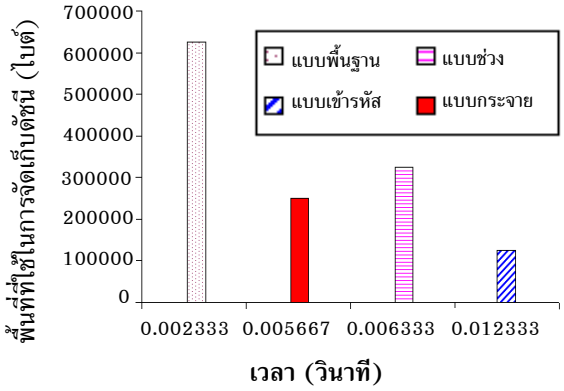


(b) กรณีจำนวนค่าที่ต้องการค้นหา = 3 (n=3)

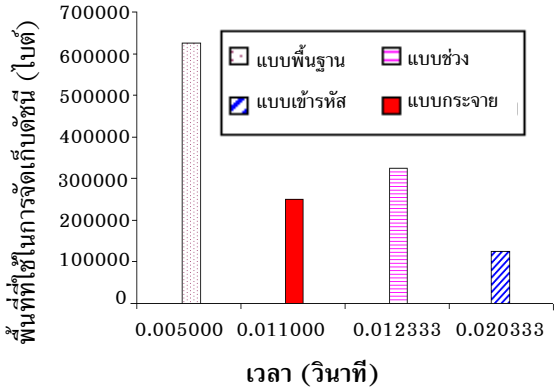


(c) กรณีจำนวนค่าที่ต้องการค้นหา = 4 (n=4)

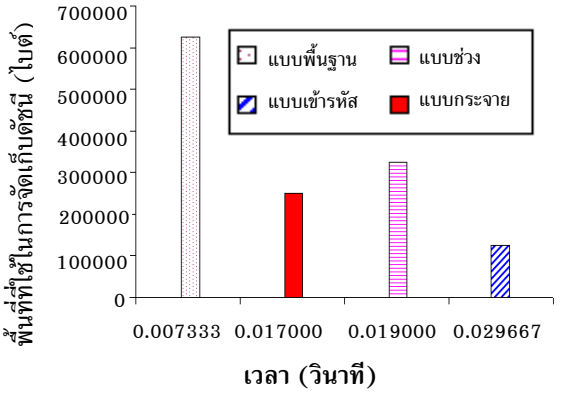
ภาพประกอบ 5-13 กราฟแสดงความสัมพันธ์ระหว่างพื้นที่ที่ใช้ในการจัดเก็บดัชนีและเวลาที่ใช้ในการค้นหาข้อมูล กรณี C = 5 การสอบถามข้อมูลแบบความเป็นสมาชิก โดย n คือ จำนวนค่าที่ต้องการค้นหา (ข้อมูลทดสอบนำมาจากการวัดสมรรถนะของ TPC-H)



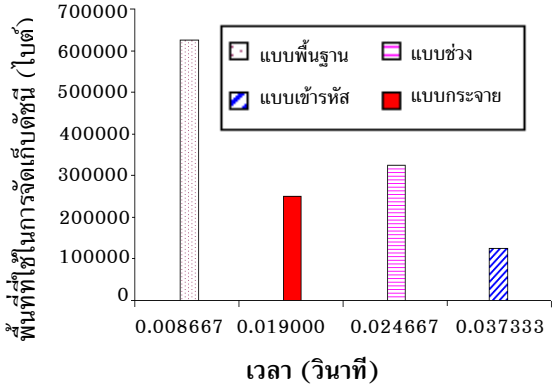
(a) กรณีจำนวนค่าที่ต้องการค้นหา = 2 (n=2)



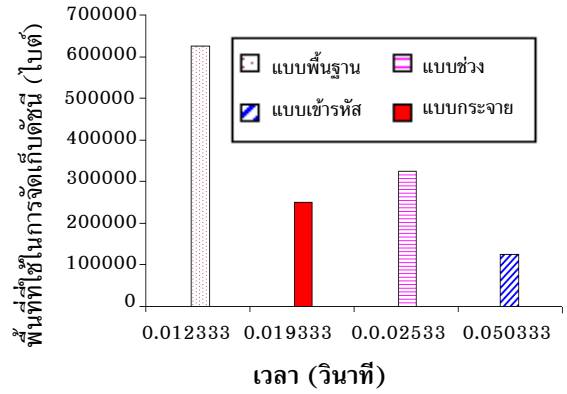
(b) กรณีจำนวนค่าที่ต้องการค้นหา = 3 (n=3)



(c) กรณีจำนวนค่าที่ต้องการค้นหา = 4 (n=4)

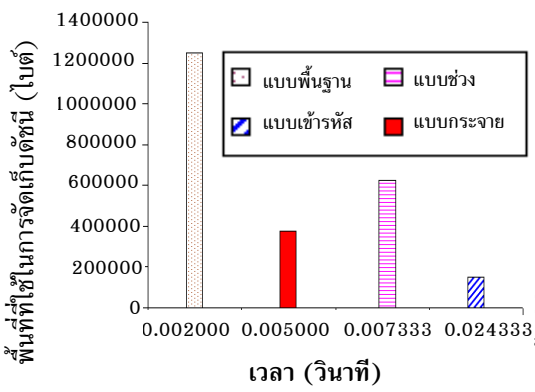


(d) กรณีจำนวนค่าที่ต้องการค้นหา = 5 (n=5)

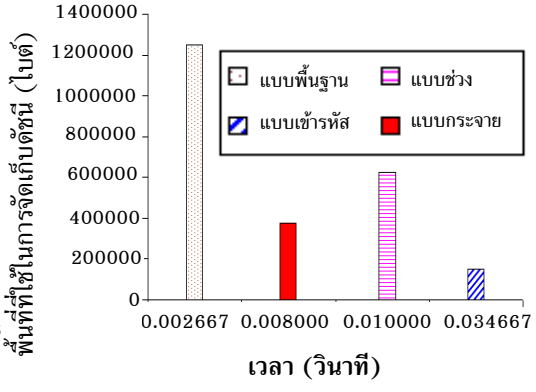


(e) กรณีจำนวนค่าที่ต้องการค้นหา = 6 (n=6)

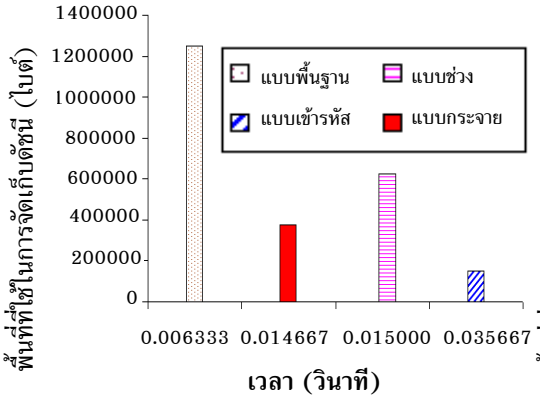
ภาพประกอบ 5-14 กราฟแสดงความสัมพันธ์ระหว่างพื้นที่ที่ใช้ในการจัดเก็บดัชนี และเวลาที่ใช้ในการค้นหาข้อมูล กรณี C = 25 การสอบถามข้อมูลแบบความเป็นสมาชิก โดย n คือ จำนวนค่าที่ต้องการค้นหา (ข้อมูลทดสอบนำมาจากการวัดสมรรถนะของ TPC-H)



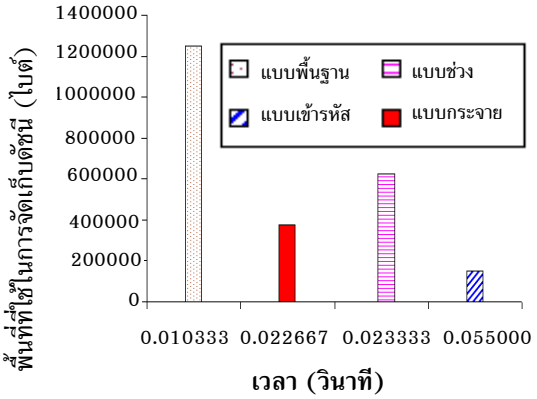
(a) กรณีจำนวนค่าที่ต้องการค้นหา = 2 (n=2)



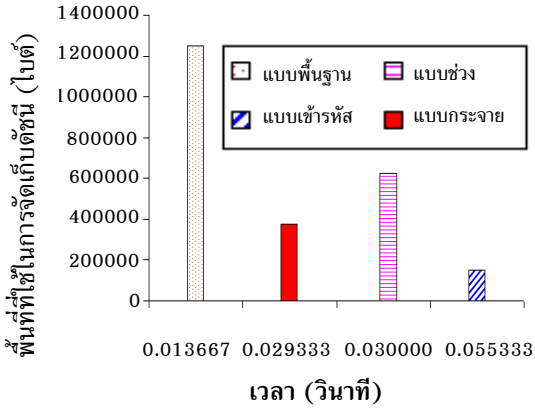
(b) กรณีจำนวนค่าที่ต้องการค้นหา = 3 (n=3)



(c) กรณีจำนวนค่าที่ต้องการค้นหา = 4 (n=4)



(d) กรณีจำนวนค่าที่ต้องการค้นหา = 5 (n=5)



(e) กรณีจำนวนค่าที่ต้องการค้นหา = 6 (n=6)

ภาพประกอบ 5-15 กราฟแสดงความสัมพันธ์ระหว่างพื้นที่ที่ใช้ในการจัดเก็บดัชนี และเวลาที่ใช้ในการค้นหาข้อมูล กรณี C = 50 การสอบถามข้อมูลแบบความเป็นสมาชิก โดย n คือ จำนวนค่าที่ต้องการค้นหา (ข้อมูลทดสอบนำมาจากการวัดสมรรถนะของ TPC-H)

จากภาพประกอบ 5-13 จะเห็นได้ว่า

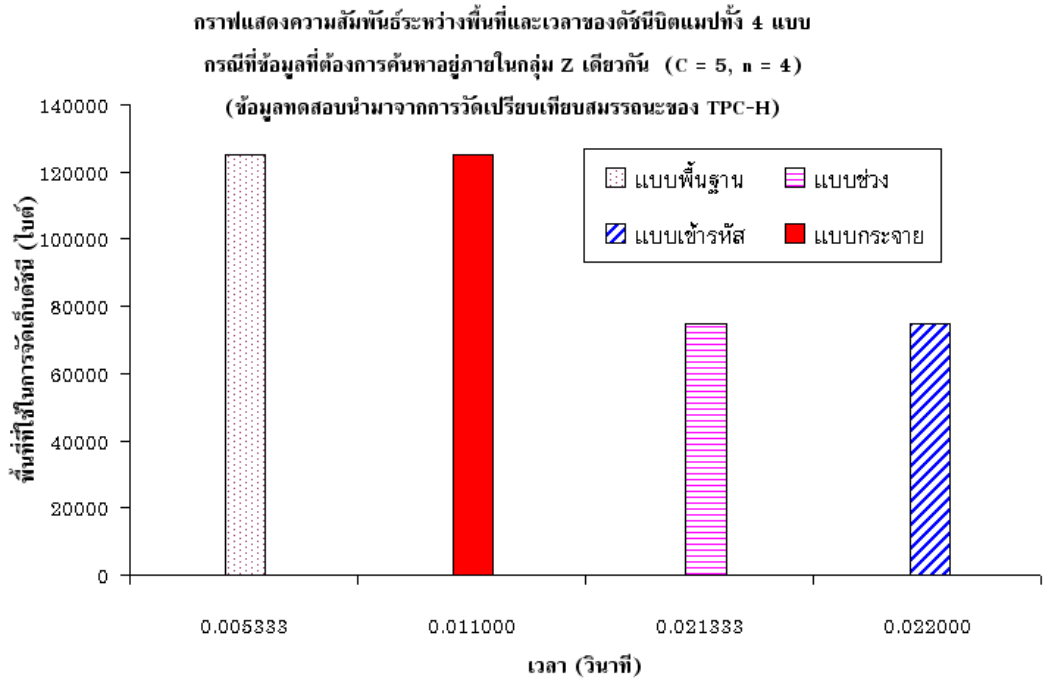
- ดัชนีบิตแมปแบบพื้นฐาน และดัชนีบิตแมปแบบกระจาย จะใช้พื้นที่มากที่สุดในการจัดเก็บดัชนี
- ดัชนีบิตแมปแบบช่วง และดัชนีบิตแมปแบบเข้ารหัส จะใช้พื้นที่น้อยที่สุดในการจัดเก็บดัชนี
- ในกรณีที่แอทริบิวต์ที่นำมาสร้างดัชนีมีค่าคาร์ดินอลิตี้ที่ต่ำมาก ๆ เช่น ในกรณีนี้ แอทริบิวต์ที่นำมาสร้างดัชนีมีค่าคาร์ดินอลิตี้เท่ากับ 5 จะทำให้ดัชนีบิตแมปแบบกระจายใช้พื้นที่ในการจัดเก็บดัชนีมากกว่าดัชนีบิตแมปแบบช่วง
- ดัชนีบิตแมปแบบพื้นฐาน ใช้เวลาน้อยที่สุดในการค้นหาข้อมูล
- ดัชนีบิตแมปแบบเข้ารหัส ใช้เวลามากที่สุดในการค้นหาข้อมูล
- ดัชนีบิตแมปแบบกระจาย ใช้เวลาในการค้นหาข้อมูลน้อยกว่าดัชนีบิตแมปแบบช่วงและแบบเข้ารหัส

จากภาพประกอบ 5-14 และ 5-15 จะเห็นได้ว่า

- ดัชนีบิตแมปแบบพื้นฐาน จะใช้พื้นที่มากที่สุดในการจัดเก็บดัชนี แต่ใช้เวลาน้อยที่สุดในการค้นหาข้อมูล
- ดัชนีบิตแมปแบบเข้ารหัส จะใช้พื้นที่น้อยที่สุดในการจัดเก็บดัชนี แต่ใช้เวลามากที่สุดในการค้นหาข้อมูล
- ดัชนีบิตแมปแบบช่วง จะใช้พื้นที่ในการจัดเก็บดัชนีมากกว่าดัชนีบิตแมปแบบเข้ารหัสและแบบกระจาย แต่ใช้เวลาในการค้นหาข้อมูลน้อยกว่าดัชนีบิตแมปแบบเข้ารหัส
- ดัชนีบิตแมปแบบกระจาย จะใช้พื้นที่ในการจัดเก็บดัชนีน้อยกว่าดัชนีบิตแมปแบบพื้นฐาน แต่ใช้เวลาในการค้นหาข้อมูลมากกว่าดัชนีบิตแมปแบบพื้นฐาน เมื่อเปรียบเทียบดัชนีบิตแมปแบบกระจายกับดัชนีบิตแมปแบบช่วงและแบบเข้ารหัส ดัชนีบิตแมปแบบกระจายจะใช้พื้นที่ในการจัดเก็บดัชนีน้อยกว่าดัชนีบิตแมปแบบช่วง แต่ใช้พื้นที่ในการจัดเก็บดัชนีมากกว่าดัชนีบิตแมปแบบเข้ารหัส และใช้เวลาในการค้นหาข้อมูลน้อยกว่าดัชนีบิตแมปแบบช่วงและแบบเข้ารหัส

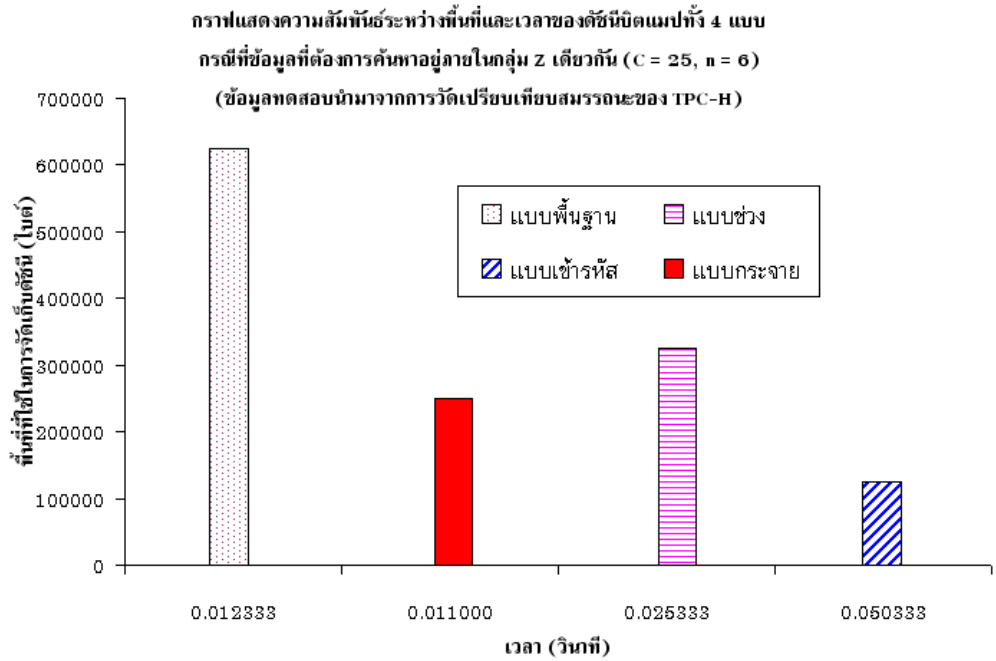
5.2.3.3 การค้นหาข้อมูลแบบความเป็นสมาชิก โดยข้อมูลที่จะค้นหาอยู่ภายในกลุ่ม Z หรือ L เดียวกันและจำนวนค่าที่ต้องการค้นหาเท่ากับจำนวนสมาชิกภายในกลุ่ม Z หรือ L

ความสัมพันธ์ระหว่างพื้นที่ที่ใช้ในการจัดเก็บดัชนี และเวลาที่ใช้ในการค้นหาข้อมูลแบบความเป็นสมาชิก ในกรณีที่ข้อมูลที่ต้องการค้นหาอยู่ภายในกลุ่ม Z หรือ L เดียวกัน และจำนวนค่าที่ต้องการค้นหาเท่ากับจำนวนสมาชิกภายในกลุ่ม Z หรือ L สามารถแสดงความสัมพันธ์ได้ ดังภาพประกอบ 5-16 และ 5-17



ภาพประกอบ 5-16 กราฟแสดงความสัมพันธ์ระหว่างพื้นที่ที่ใช้ในการจัดเก็บดัชนีและเวลาที่ใช้ในการค้นหาข้อมูลแบบความเป็นสมาชิก กรณีที่ค่าที่ต้องการค้นหาอยู่ภายในกลุ่ม Z เดียวกัน (C = 5, n = 4) (ข้อมูลทดสอบนำมาจากการวัดสมรรถนะของ TPC-H)

ภาพประกอบ 5-16 เป็นการแสดงความสัมพันธ์ระหว่างพื้นที่ที่ใช้ในการจัดเก็บดัชนีและเวลาที่ใช้ในการค้นหาข้อมูล กรณีที่แอทริบิวต์ที่นำมาสร้างดัชนีมีค่าคาร์ดินอลิตี้เท่ากับ 5 และเป็นการค้นหาแบบความเป็นสมาชิก จำนวนค่าที่ต้องการค้นหาเท่ากับ 4 (n = 4) และจำนวนสมาชิกภายในกลุ่ม Z เท่ากับ 4 (m = 4) โดยค่าที่ต้องการค้นหานั้นอยู่ภายในกลุ่ม Z เดียวกัน ซึ่งจะเห็นได้ว่าเป็นทำนองเดียวกันกับภาพประกอบ 5-13(c) แต่เวลาที่ใช้ในการค้นหาข้อมูลของดัชนีบิตแมปแบบกระจายลดลง



ภาพประกอบ 5-17 กราฟแสดงความสัมพันธ์ระหว่างพื้นที่ที่ใช้ในการจัดเก็บดัชนีและเวลาที่ใช้ในการค้นหาข้อมูลแบบความเป็นสมาชิก กรณีที่ค่าที่ต้องการค้นหาอยู่ภายในกลุ่ม Z เดียวกัน (C = 25, n = 6) (ข้อมูลทดสอบนำมาจากการวัดสมรรถนะของ TPC-H)

ภาพประกอบ 5-17 เป็นการแสดงความสัมพันธ์ระหว่างพื้นที่ที่ใช้ในการจัดเก็บดัชนีและเวลาที่ใช้ในการค้นหาข้อมูล กรณีที่แอทริบิวต์ที่นำมาสร้างดัชนีมีค่าคาร์ดินอลิตี้เท่ากับ 25 และเป็นการค้นหาแบบความเป็นสมาชิก จำนวนค่าที่ต้องการค้นหาเท่ากับ 6 (n = 6) และจำนวนสมาชิกภายในกลุ่ม Z เท่ากับ 6 (m = 6) โดยค่าที่ต้องการค้นหานั้นอยู่ภายในกลุ่ม Z เดียวกัน ซึ่งจะเห็นได้ว่าเป็นทำนองเดียวกันกับภาพประกอบ 5-14(e) แต่เวลาที่ใช้ในการค้นหาข้อมูลของดัชนีบิตแมปแบบกระจายลดลง

จากภาพประกอบข้างต้น แสดงความสัมพันธ์ระหว่างพื้นที่ที่ใช้ในการจัดเก็บดัชนีและเวลาที่ใช้ในการค้นหาข้อมูล จะเห็นว่า เมื่อเปรียบเทียบดัชนีบิตแมปแบบกระจายกับแบบพื้นฐานเป็นการลดพื้นที่ แต่เพิ่มเวลา เมื่อเปรียบเทียบดัชนีบิตแมปแบบกระจายกับแบบเข้ารหัสเป็นการเพิ่มพื้นที่ แต่ลดเวลา และเมื่อเปรียบเทียบดัชนีบิตแมปแบบกระจายกับแบบช่วง เป็นการลดทั้งพื้นที่และเวลาด้วย นอกจากนี้ในกรณีที่ต้องการค้นหาข้อมูลแบบความเป็นสมาชิก โดยที่จำนวนค่าที่ต้องการค้นหาเท่ากับจำนวนสมาชิกภายในกลุ่ม Z (ค่า m) หรือ L และค่าที่ต้องการค้นหานั้นอยู่ภายในกลุ่ม Z หรือ L เดียวกัน ก็จะช่วยลดเวลาในการค้นหาข้อมูลลงได้อีก

จากผลการทดลองในเรื่องพื้นที่ที่ใช้ในการจัดเก็บดัชนี จะสอดคล้องตรงกันกับการวิเคราะห์ค่าใช้จ่ายในเรื่องพื้นที่ที่ใช้ในการจัดเก็บดัชนีบิตแมป ในตาราง 5-1 กล่าวคือ ดัชนีบิตแมปแบบพื้นฐานจะใช้พื้นที่มากที่สุดในการจัดเก็บดัชนี รองลงมา คือ ดัชนีบิตแมปแบบช่วง ส่วนดัชนีบิตแมปแบบเข้ารหัสจะใช้พื้นที่น้อยที่สุด และดัชนีบิตแมปแบบกระจาย จะใช้พื้นที่อยู่ระหว่างดัชนีบิตแมปแบบช่วง และแบบเข้ารหัส สำหรับผลการทดลองในเรื่องเวลาที่ใช้ในการค้นหาข้อมูลแบบค่าเท่ากัน จะสอดคล้องตรงกันกับการวิเคราะห์ในเรื่องเวลาที่ใช้ในการค้นหาข้อมูลแบบค่าเท่ากัน ในตาราง 5-2 กล่าวคือ ดัชนีบิตแมปแบบพื้นฐานจะมีประสิทธิภาพมากที่สุดในการค้นหาข้อมูลแบบค่าเท่ากัน คือ ใช้เวลาน้อยที่สุดในการค้นหาข้อมูล เพราะจำนวนบิตแมปที่ถูกอ่าน เท่ากับ 1 และไม่มีการดำเนินการตรรกะใด ๆ เกิดขึ้น รองลงมา คือ ดัชนีบิตแมปแบบกระจาย เพราะจำนวนบิตแมปที่ถูกอ่าน เท่ากับ 2 และมีการดำเนินการตรรกะเกิดขึ้นเพียง 1 ตัวดำเนินการเท่านั้น ดัชนีบิตแมปแบบช่วง จะใช้เวลามากกว่าดัชนีบิตแมปแบบกระจาย เพราะจำนวนบิตแมปที่ถูกอ่าน เท่ากับ 2 และมีการดำเนินการตรรกะเกิดขึ้น 2 ตัวดำเนินการ สำหรับดัชนีบิตแมปแบบเข้ารหัส จะมีประสิทธิภาพน้อยที่สุดในการค้นหาข้อมูลแบบค่าเท่ากัน คือ ใช้เวลามากที่สุดในการค้นหาข้อมูล เพราะจะต้องเสียเวลาในการอ่านค่าที่จะค้นหานั้น จากตารางการเทียบค่าก่อนว่ามีการเข้ารหัสบิตแมปเวกเตอร์ในรูปแบบอะไร แล้วไปอ่านบิตแมปเวกเตอร์ทั้ง $\lceil \log_2 C \rceil$ บิตแมปเวกเตอร์จากตารางดัชนี

สรุปได้ว่า ดัชนีบิตแมปแบบกระจายเหมาะสมกับแอทริบิวต์ที่มีคาร์ดินอลิตี้สูง เพราะใช้พื้นที่น้อยในการจัดเก็บดัชนีและเวลาในการค้นหาข้อมูลเหมาะสม ในกรณีที่เป็นการสอบถามข้อมูลแบบค่าเท่ากันและแบบความเป็นสมาชิก