

บทที่ 6

การทดลองและวิเคราะห์หาประสิทธิภาพของแต่ละอัลกอริทึม

6.1 สภาพแวดล้อมในการทดลอง

การจำลองสภาพแวดล้อมของเครือข่ายที่นำมาใช้ในการทดลองภายในงานวิจัย เราได้อ้างอิงข้อมูลและตัวแปรต่างๆ ในการสร้างเครือข่าย รูปแบบของกระแสข้อมูล รวมถึงกำหนดต่างๆ จาก [6] และทำการทดลองกับเครือข่ายที่ทำการสุ่มสร้างขึ้นมา 100 เครือข่าย รวมถึงกระแสข้อมูลที่ใช้อยู่ในแต่ละเครือข่าย โดยมีรายละเอียดต่างๆ ดังต่อไปนี้

การจำลองเครือข่าย

เครือข่าย 100 เครือข่ายที่ใช้ในการทดสอบอัลกอริทึมต่างๆ เราจะกำหนดค่าต่างๆ สำหรับจำนวนบอเดอร์เราเตอร์ภายในโพลีโทปเอเอสมีค่าเป็น X จำนวนของเครือข่ายเอเอสข้างเคียงมีค่าเป็น H และจำนวนหมายเลขเครือข่ายต้องส่งผ่านข้อมูลผ่านเป็น K สำหรับรูปแบบเครือข่ายที่ได้มีการสร้างขึ้นมามีรายละเอียดต่างๆ ดังนี้

1. สร้างเซตของท่อส่งข้อมูล b_i
2. สร้างเซตของหมายเลขเครือข่าย P_k
3. ค่าต้นทุนระยะทางระหว่างบอเดอร์เราเตอร์ จะสุ่มค่าระหว่าง $[10...100]$ โดยที่จะให้ค่าของ $d(i, j) = d(j, i)$
4. ค่าจำนวนท่อส่งข้อมูลของบอเดอร์เราเตอร์ สุ่มค่าระหว่าง $[1...3]$ จากเซตของท่อส่งข้อมูล ซึ่งแต่ละบอเดอร์เราเตอร์จะต้องได้ท่อส่งข้อมูลที่ไม่ซ้ำกัน
5. สำหรับแต่ละเครือข่ายเอเอสข้างเคียงหรือ A_h มีเซตของ $In(h)$ หรือเซตของท่อส่งข้อมูลขาเข้าของแต่ละ A_h ซึ่งสุ่มจากเซตของท่อส่งข้อมูลระหว่าง $[1...3]$ ท่อส่งข้อมูล
6. สำหรับแต่ละหมายเลขเครือข่าย P_k มีเซตของ $Out(k)$ จะสุ่มเลือกท่อส่งข้อมูลขาออกจำนวน $[2...5]$ จากเซตของท่อส่งข้อมูล

การจำลองกระแสข้อมูล

โดยสมมติให้ทุกๆ เครือข่ายเอเอสข้างเคียงมีการส่งข้อมูลไปยังทุกหมายเลขเครือข่ายปลายทาง โดยที่มีขนาดของข้อมูล เป็นการสุ่มค่าจาก $[0...20]$ หน่วย โดยที่การทดลองไม่อนุญาตให้มีลูป และถ้าเครือข่ายเอเอสข้างเคียงมีการประกาศค่าหมายเลขเครือข่ายปลายทาง P_k มา เครือข่ายนั้นจะไม่สามารถส่งข้อมูลไปหา P_k ผ่านโพลีโทปเอเอสได้

6.2 การทดลองในปัญหาการเลือกเส้นทางแบบใช้ทางออกทางเดียว

ภายในงานวิจัยในงานการเลือกเส้นทางแบบใช้ทางออกทางเดียวของบีจีพี ได้ทำการทดลองโดย สุ่มสร้างโทโปโลยีของเครือข่ายโคคอลลเอเอส เป็นจำนวน 100 เครือข่าย เพื่อนำมาใช้ในการทดลอง โดยทำการทดลองกับอัลกอริทึมต่างๆ ดังต่อไปนี้

6.2.1 อัลกอริทึมบีทีเอฟ (Biggest Traffic First: BTF)

สำหรับหลักการทำงานของอัลกอริทึมบีทีเอฟ มีการทำงานอย่างง่ายๆ โดยที่ จะเรียงกระแสข้อมูลจากทุกๆ เอเอสจากปริมาณมากไปหาน้อย จากนั้นทำการกำหนดเส้นทางโดยฟังก์ชัน $f(A_h, b, P_k)$ โดยกระแสข้อมูล $t(h, i, k)$ ที่มีปริมาณมาก จะได้สิทธิในการเลือกท่อนส่งข้อมูลโดยเลือกท่อนส่งข้อมูลที่ใกล้ที่สุดสำหรับ P_k ก่อน และเมื่อมีการเลือกท่อนส่งข้อมูลขาออกไปแล้ว กระแสข้อมูลอื่นที่ตามมาเพื่อที่จะไปหมายเลขเครือข่าย P_k จะต้องเลือกท่อนส่งข้อมูลเดิมเสมอ ไม่สามารถเลือกท่อนส่งข้อมูลอื่นได้ การทำงานของอัลกอริทึมบีทีเอฟแสดงอยู่ในรูปที่ 6.1

Biggest Traffic First Algorithm

Inputs: $t(h, i, k)$ for all h, i, k ; $d(i, j)$ for all i, j

Outputs: f

1: For all j , set $U_j = 0$. /* U_j records the amount of traffic assigned to b_j . */

2: Sort $t(h, i, k)$ in an ordered list T .

3: For all k , set $f(k) = 0$. /* Initialize assignment to null. */

4: For each $t(h, i, k)$ in T .

5: if ($f(k) \neq 0$) then

6: Sort $j \in \text{Out}(k)$ in non-decreasing order of $\sum_{h=1}^H \sum_{i=1}^I d(i, j) t(h, i, k)$ to form an ordered

list J .

$h=1 \quad i=1$

7: For each j in the ordered list J

8: if ($U_j + t(h, i, k) \leq C_j$) then /* Check for capacity violation */

9: Set $f(k) = j$. /* Egress link b_j is selected */

10: Set $U_j = U_j + t(h, i, k)$

11: If $f(k) = 0$, quit program.

12: Else

13: if ($U_j + t(h, i, k) \leq C_j$) then

14: Set $U_j = U_j + t(h, i, k)$

15: Else quit program.

16: End

รูปที่ 6.1 แสดงขั้นตอนการทำงานของอัลกอริทึมบีทีเอฟ

6.2.2 อัลกอริทึมเอ็มพีพีเอฟ (Most Popular Prefix First:MPPF)

การทำงานของอัลกอริทึมเอ็มพีพีเอฟในการทดลองปัญหาการเลือกเส้นทางแบบใช้ทางออกทางเดียวของบีจีพี เป็นการนำวิธีฮิวริสติกมาใช้แก้ปัญหาอีกวิธีหนึ่ง โดยเริ่มต้นจากการรวมค่าแบนวิดท์แต่ละ P_k เข้าด้วยกัน จากนั้นเรียงลำดับจากมากไปน้อย โดยให้กลุ่มของกระแสข้อมูลที่จะไป P_k เดียวกันและปริมาณแบนวิดท์สูงเลือกท่อนส่งข้อมูลที่ใช้ต้นทุนต่ำที่สุดก่อน ซึ่งเหมือนกับเป็นสิทธิพิเศษกับหมายเลขเครือข่ายที่มีปริมาณการส่งข้อมูลมากเลือกก่อน วิธีนี้จึงถูกเรียกว่า Most Popular Prefix First (MPPF) ตัวอย่างการทำงานของอัลกอริทึมเอ็มพีพีเอฟแสดงอยู่ในรูป 6.2

Most Popular Prefix First Algorithm

Inputs: $t(h, i, k)$ for all h, i, k ; $d(i, j)$ for all i, j

Outputs: f

1: For all j , set $U_j = 0$. /* U_j records the amount of traffic assigned to b_j . */

2: Compute

3:
$$p_k = \sum_{h=1}^H \sum_{i=1}^I t(h, i, k) \text{ for all } k \in [1, K]$$

4: Sort k in non-increasing order of p_k to form an ordered list K .

5: For all k , set $f(k) = 0$. /* Initialize assignment to null. */

6: For each k in the ordered list K

7: Sort $j \in \text{Out}(k)$ in non-decreasing order of $\sum_{h=1}^H \sum_{i=1}^I d(i, j) t(h, i, k)$ to form an ordered list J .

8: For each j in the ordered list J

9: if ($U_j + p_k \leq C_j$) then /* Check for capacity violation */

11: Set $f(k) = j$. /* Egress link b_j is selected */

12: Set $U_j = U_j + p_k$

13: If $f(k) = 0$, quit program.

14: End

รูปที่ 6.2 แสดงขั้นตอนการทำงานของอัลกอริทึมเอ็มพีพีเอฟ

6.2.3 อัลกอริทึมจินติกแบบง่าย (SGA)

อัลกอริทึมจินติกที่ใช้ในการทดลองปัญหาการเลือกเส้นทางแบบใช้ทางออกทางเดียวของบีจีพี แบบอัลกอริทึมจินติกแบบง่ายโดย การครอสโอเวอร์ โดยให้ค่า $P_c = 0.8$ ค่า $P_m = 0.02$ และจำนวนประชากรเริ่มต้นที่ 50 สำหรับฟังก์ชันในการสุ่มประชากรเริ่มต้นทำที่ 30000 ครั้งต่อประชากรหนึ่งตัวสำหรับการสุ่มประชากรเริ่มต้นโดยคำนึงถึงเงื่อนไขของปัญหา ซึ่งแทนด้วย SGA (S) และสุ่มประชากรเริ่มต้นเพียงครั้งเดียวกับวิธีการสุ่มประชากรเริ่มต้นโดยไม่คำนึงเงื่อนไขของปัญหา ซึ่งแทนด้วย SGA (L)

6.2.4 อัลกอริทึมจีเนติกและฮิวริสติก (HGA)

การปรับปรุงอัลกอริทึมจีเนติกที่ใช้ในการทดลองปัญหาการเลือกเส้นทางแบบใช้ทางออกทางเดียวของบีจีพี โดยใช้วิธีฮิวริสติกในการมิวเตชันและครอสโอเวอร์ โดยการเลือกค่าต้นทุนที่ต่ำที่สุด ในการปรับเปลี่ยนบิตในแถวของหมายเลขเครือข่าย P_k ไปยังจุดที่มีค่าต้นทุนรวมต่ำที่สุด โดยที่ไม่ขัดกับค่าจำกัดด้านปริมาณที่ส่งข้อมูล โดยให้ค่า $P_m = 0.02$ $P_c = 0.8$ และจำนวนประชากรเริ่มต้นที่ 50 สำหรับฟังก์ชันในการสุ่มประชากรเริ่มต้นทำที่ 30000 ครั้งต่อประชากรหนึ่งตัวสำหรับการสุ่มประชากรเริ่มต้นโดยคำนึงถึงเงื่อนไขของปัญหา ซึ่งแทนด้วย HGA (S) และสุ่มประชากรเริ่มต้นเพียงครั้งเดียวกับวิธีการสุ่มประชากรเริ่มต้นโดยไม่คำนึงเงื่อนไขของปัญหา ซึ่งแทนด้วย HGA (L)

6.3 การวิเคราะห์ผลการทดลองของปัญหาการเลือกเส้นทางแบบใช้ทางออกทางเดียว

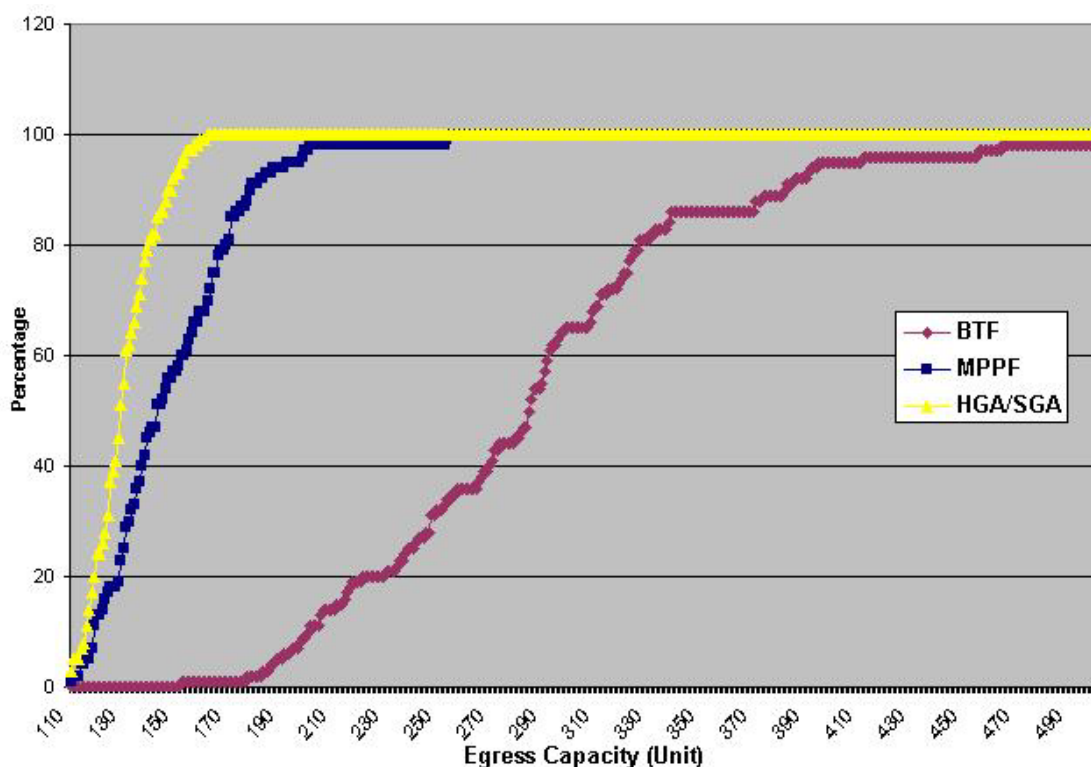
การทำการวิจัยและทดลองปัญหาการเลือกเส้นทางแบบใช้ทางออกทางเดียวของบีจีพี จะทำการวิเคราะห์และการเปรียบเทียบผลของแต่ละอัลกอริทึม โดยจะแบ่งออกเป็นสองลักษณะคือ

- การเปรียบเทียบด้านการค้นพบคำตอบ
- การเปรียบเทียบด้านต้นทุนรวมของคำตอบ
- การเปรียบเทียบด้านเวลาที่ใช้ในการค้นหาคำตอบ

6.3.1 การเปรียบเทียบด้านการค้นพบคำตอบ

เป็นการเปรียบเทียบความสามารถของแต่ละอัลกอริทึม ในการค้นหาคำตอบในแต่ละขนาดของข้อมูลขาออก สำหรับความหมายของคำตอบคือ เครือข่ายจะต้องส่งกระแสข้อมูลทั้งหมดที่วิ่งผ่านเครือข่ายไปยังเครือข่ายปลายทางได้ทั้งหมด ตามข้อกำหนดในสมการที่ (4.3) ดังนั้นอัลกอริทึมที่ดีควรจะคำนวณการส่งข้อมูล โดยใช้ขนาดของข้อมูลมีขนาดจำกัด ซึ่งเป็นขนาดที่พอดีหรือใกล้เคียงกับปริมาณข้อมูลที่ส่งผ่านเครือข่าย การที่สามารถใช้ขนาดของข้อมูลได้น้อยกว่าโดยที่ปริมาณข้อมูลที่ส่งเท่ากันนั้น แสดงว่าเครือข่ายสามารถลดต้นทุนด้านการเชื่อมต่อระหว่างไอเอสพี ซึ่งเป็นต้นทุนหลักในการดำเนินงานของไอเอสพี ในขณะที่เรายังสามารถให้บริการกับลูกค้าได้เท่าเดิม

เริ่มต้นจากการเปรียบเทียบผลการทดลองของอัลกอริทึม SGA (L) และ SGA(S) รวมถึง HGA (L) และ HGA (S) ซึ่งเป็นจีเนติกอัลกอริทึมจึงแทนด้วย SGA และ HGA สำหรับ BTS กับ MPPF ซึ่งเป็นอัลกอริทึมที่ได้ถูกพัฒนามาก่อนในเอกสาร [6] และใช้เป็นพื้นฐานในการเปรียบเทียบกับอัลกอริทึมที่เราได้พัฒนาขึ้นมาทั้งหมด ผลการทดลองที่ได้ด้านการค้นพบคำตอบทั้งหมดของ 100 เครือข่ายที่ใช้ในการทดลองแสดงอยู่ในรูปที่ 6.3



รูปที่ 6.3 แสดงผลการทดลองในการเปรียบเทียบด้านการค้นหาคำตอบ

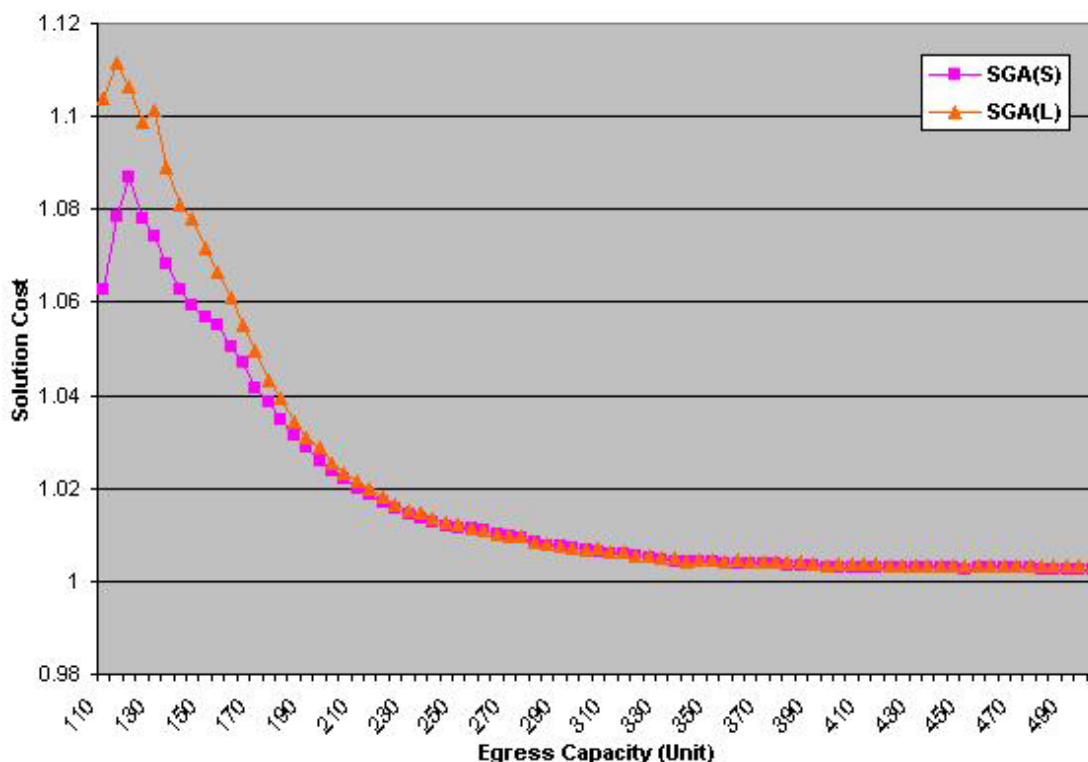
ในรูปที่ 6.3 แสดงผลการทดลองในการเปรียบเทียบด้านการค้นหาคำตอบ โดยแกนแนวนอนแทนค่าขนาดของท่อส่งข้อมูล ซึ่งมีหน่วยเป็นยูนิต ซึ่งถ้านำมาเทียบเคียงกับในสภาวะปัจจุบัน อาจแทนค่าเป็นเมกะบิต (Mbps) โดยเริ่มต้นจาก 110 หน่วยซึ่งเป็นค่าขนาดท่อส่งข้อมูลแรกที่มีการพบคำตอบ จนถึง 500 หน่วยซึ่งเป็นค่าขนาดท่อส่งข้อมูลที่ทุกอัลกอริทึมสามารถหาคำตอบได้ทั้งนี้ร้อยโทโปโลยีเครือข่าย ส่วนแกนในแนวตั้งแทนค่าเปอร์เซ็นต์การค้นพบคำตอบ ซึ่ง การที่อัลกอริทึมค้นพบคำตอบ 100% แสดงให้เห็นว่าสามารถหาคำตอบได้ครบทั้ง 100 เครือข่าย จากการทดลองโดยทำการเปรียบเทียบระหว่าง อัลกอริทึม SGA(S) , SGA(L) ซึ่งแทนด้วย SGA และ HGA(S) , HGA(L) ซึ่งแทนด้วย HGA และ MPPF กับ BTF ซึ่งผลการทดลองจะเห็นได้ว่าอัลกอริทึมจินตริก SGA และ HGA สามารถค้นหาคำตอบได้ 100% ในขณะที่ใช้ขนาดของท่อส่งข้อมูลที่ 162 หน่วย ซึ่งจินตริกอัลกอริทึมที่ใช้ในการทดสอบนี้ ยังประกอบย่อยๆ ไปด้วย SGA และ HGA แบบการสุ่มประชากรเริ่มต้นโดยคำนึงและไม่คำนึงถึงเงื่อนไขของปัญหาซึ่งทั้งหมดพบคำตอบที่เหมือนกัน แต่สำหรับ MPPF จะพบคำตอบ 100% ในขณะที่ใช้ขนาดของท่อส่งข้อมูลที่ 254 หน่วยและ BTF ที่เจอคำตอบทั้ง 100 เครือข่ายที่ 464 ซึ่งเมื่อทำการเปรียบเทียบเป็นเปอร์เซ็นต์แล้ว จะทำให้เห็นว่าอัลกอริทึม SGA และ HGA ทำงานได้ดีกว่า MPPF ถึง 36% และยังสามารถค้นหาคำตอบได้ดีกว่า BTF ถึง 65% สรุปผลการทดลองในด้านการค้นหาคำตอบแสดงให้เห็นว่า อัลกอริทึม SGA และ HGA สามารถหาค้นพบคำตอบได้ในขณะที่ปริมาณท่อส่งข้อมูลมีขนาดใกล้เคียงกับปริมาณข้อมูลที่ส่ง

สำหรับ SGA (S) และ HGA (S) แบบการสุ่มประชากรเริ่มต้นโดยคำนึงถึงเงื่อนไขของปัญหาสามารถหาคำตอบได้ถูกต้องจากการทดลองทั้งหมดจากทั้ง 100 เครือข่ายทำการทดลองเริ่มตั้งแต่ขนาดท่อนส่งข้อมูล 100 หน่วยไปจนถึง 500 หน่วย และแต่ละขนาดท่อนส่งข้อมูลทำการทดลอง 100 ครั้ง ส่วน SGA (L) และ HGA (L) แบบการสุ่มประชากรเริ่มต้นโดยไม่คำนึงถึงเงื่อนไขของปัญหานั้น มีความผิดพลาดในการหาคำตอบ 1.73% ซึ่งความผิดพลาดในส่วนนี้ เกิดจากการที่ทำการทดลอง 100 ครั้งใน 1 ขนาดท่อนส่งข้อมูลของเครือข่าย ไม่สามารถหาคำตอบได้ครบทั้ง 100 ครั้ง

6.3.2 การเปรียบเทียบด้านต้นทุนรวมของคำตอบ

เป็นการเปรียบเทียบด้านประสิทธิภาพของอัลกอริทึมในด้านค่าของคำตอบที่แต่ละอัลกอริทึมหามาได้นั้น อัลกอริทึมใดสามารถหาค่าต้นทุนรวมของการส่งข้อมูลผ่านเครือข่ายได้ต่ำกว่ากัน สำหรับการกำหนดค่าต้นทุนการเดินทางผ่านเครือข่าย เป็นการหมายถึงการกำหนดค่าความสำคัญให้กับทรัพยากรภายในเครือข่าย เช่น โหลดของอุปกรณ์ที่ข้อมูลวิ่งผ่าน จำนวนพอร์ต ปริมาณโหลดของท่อนส่งข้อมูลภายใน กระแสข้อมูลที่เดินทางผ่านอุปกรณ์เหล่านั้นหมายถึงการใช้ทรัพยากรภายในเครือข่าย ดังนั้นการที่อัลกอริทึมไหนที่สามารถหาค่าต้นทุนรวมที่ข้อมูลวิ่งผ่านเครือข่ายได้ต่ำที่สุด แสดงให้เห็นว่า การส่งข้อมูลผ่านเครือข่ายใช้ทรัพยากรของเครือข่ายได้มีประสิทธิภาพมากที่สุดเช่นกัน

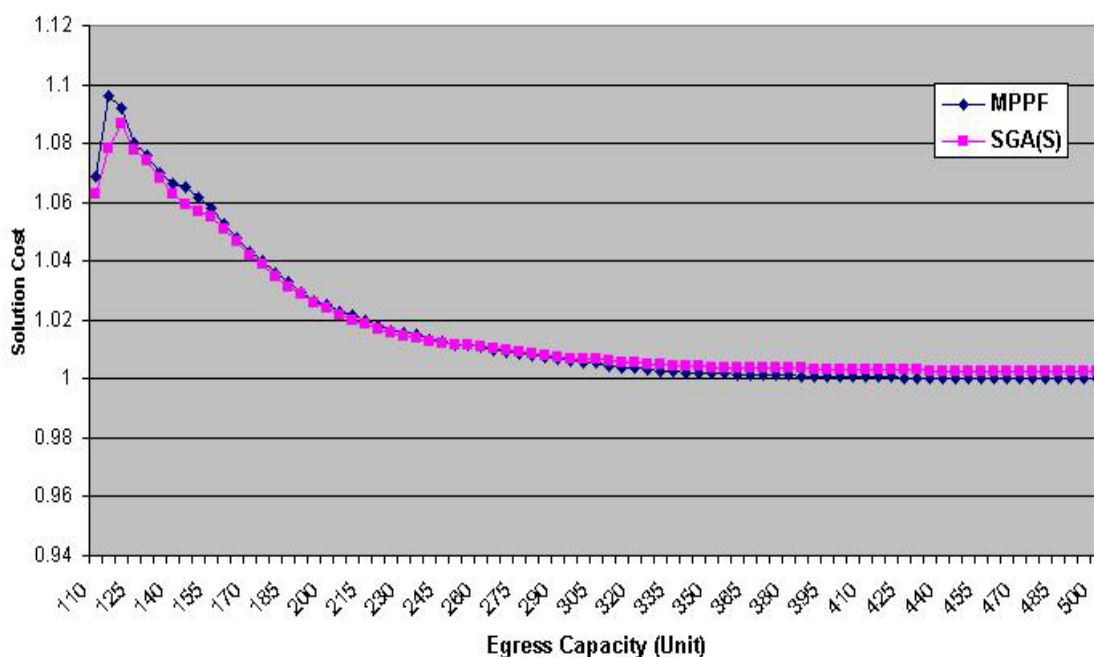
การเริ่มต้นของการทดลองเพื่อที่จะได้อัลกอริทึมจินตึก ที่สามารถหาค่าต้นทุนมีประสิทธิภาพมากที่สุด หรือกล่าวอีกนัยหนึ่งคือการหาว่าจินตึกอัลกอริทึมใดสามารถหาค่าต้นทุนได้ต่ำที่สุด โดยเริ่มต้นเปรียบเทียบกันระหว่าง SGA (S) แบบการสุ่มประชากรเริ่มต้นโดยคำนึงถึงเงื่อนไขของปัญหา และ SGA (L) แบบการสุ่มประชากรเริ่มต้นโดยไม่คำนึงถึงเงื่อนไขของปัญหา ซึ่งผลการทดลองด้านต้นทุนรวมระหว่าง SGA (L) และ SGA (S) สำหรับการเปรียบเทียบด้านต้นทุนรวมระหว่าง SGA (L) และ SGA (S) จะทำการทดลองกับเครือข่าย 100 โดยเปรียบเทียบกันในการหาค่าต้นทุนต่ำสุด ซึ่งการที่คำตอบที่ได้มีค่าเข้าใกล้ 1 มากเท่าไรแสดงว่าคำตอบนั้นเป็นจุดที่เหมาะสมที่สุดในเครือข่านั้น



รูปที่ 6.4 แสดงผลการเปรียบเทียบค่าต้นทุนรวมระหว่าง SGA (S) และ SGA (L)

จากรูปที่ 6.4 แสดงให้เห็นถึงผลการทดลองของ SGA (S) และ SGA (L) โดยที่แกนแนวนอนแทนด้วยขนาดท่อส่งข้อมูล และแกนแนวตั้งแสดงค่าต้นทุนรวม โดยที่ค่าต้นทุนรวมที่ดีที่สุดของปัญหาถูกแทนที่ด้วย 1 จากผลการทดลองการคำนวณหาค่าต้นทุนต่ำสุด จะเห็นได้ว่า SGA (S) ซึ่งเป็นแบบการสุ่มประชากรเริ่มต้นโดยคำนึงถึงเงื่อนไขของปัญหา สามารถหาต้นทุนได้ต่ำกว่าแบบ SGA (L) ซึ่งเป็นแบบการสุ่มประชากรเริ่มต้นโดยไม่คำนึงถึงเงื่อนไขของปัญหา

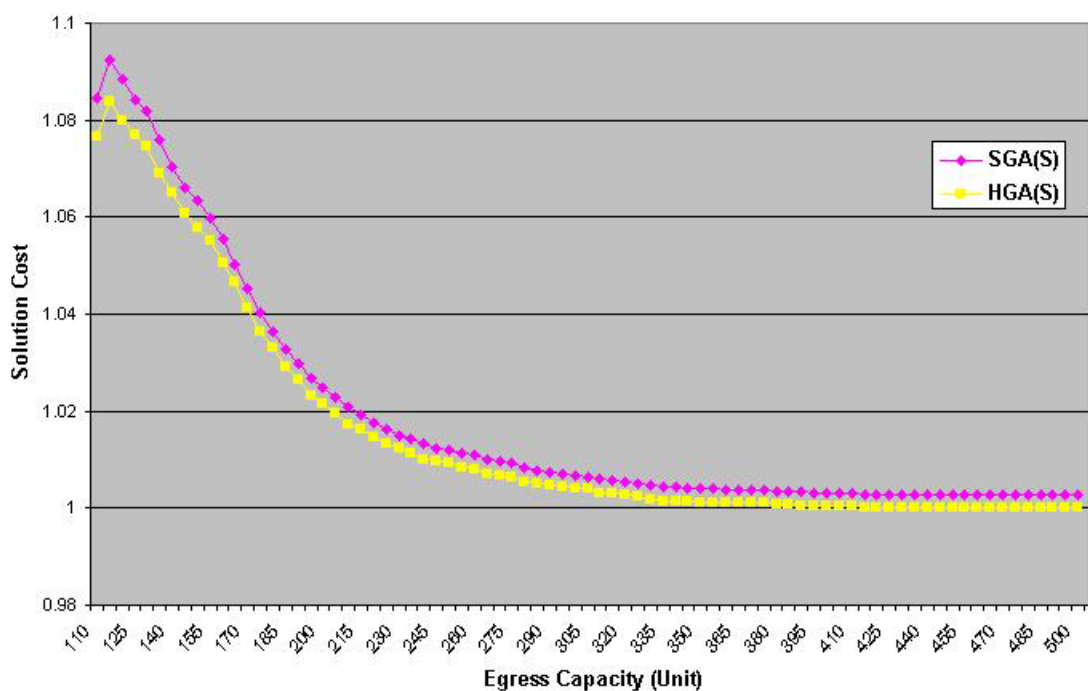
ดังนั้นการทดลองขั้นต่อไป จะเป็นการเปรียบเทียบเฉพาะจีเนติกอัลกอริทึมแบบการสุ่มประชากรเริ่มต้นโดยคำนึงถึงเงื่อนไขของปัญหา กับอัลกอริทึมอื่นๆ สำหรับการเปรียบเทียบด้านต้นทุนรวมระหว่าง SGA (S) และ MPPF ซึ่งทำการทดลองกับเครือข่าย 100 เครือข่าย และทำการเปรียบเทียบเฉพาะเครือข่ายที่ทั้ง SGA (S) และ MPPF สามารถค้นหาคำตอบได้เท่านั้น ผลของการทดลองแสดงอยู่ในรูปที่ 6.5



รูปที่ 6.5 แสดงผลการเปรียบเทียบค่าต้นทุนรวมระหว่าง SGA (S) และ MPPF

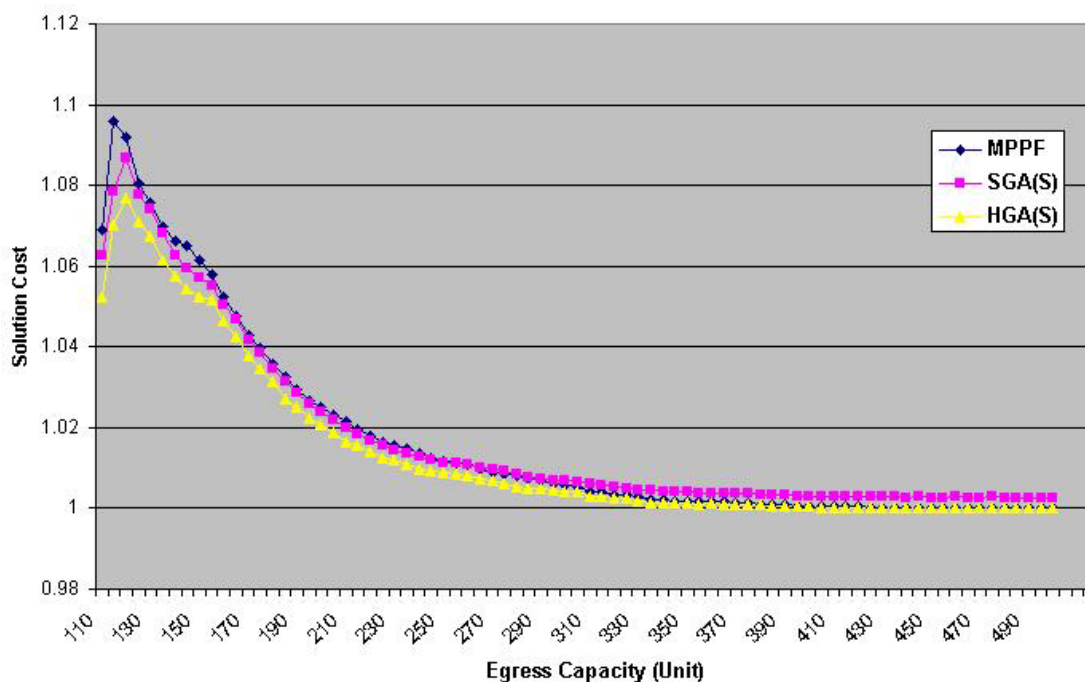
จากรูปที่ 6.5 แสดงผลการเปรียบเทียบค่าต้นทุนรวมระหว่าง SGA (S) และ MPPF โดยที่แกนแนวนอนแทนด้วยขนาดท่อส่งข้อมูล และแกนแนวตั้งแสดงค่าต้นทุนรวม โดยที่ค่าต้นทุนรวมที่ดีที่สุดของปัญหาถูกแทนที่ด้วย 1 จากผลการทดลองอัลกอริทึม SGA (S) สามารถหาค่าต้นทุนรวมในการเดินทางผ่านเครือข่ายได้ดีกว่า ในขณะที่ขนาดของท่อส่งข้อมูลอยู่ระหว่าง 110 หน่วย ถึง 255 หน่วยซึ่งสามารถหาค่าต้นทุนรวมได้ต่ำกว่า MPPF 0.25% สำหรับการเปรียบเทียบระหว่าง SGA (S) และ MPPF จะเห็นได้ว่า ผลลัพธ์ที่ได้นั้นมีขนาดใกล้เคียงกัน และในกรณีที่ขนาดท่อส่งข้อมูลมีขนาดกว้างขึ้นเรื่อยๆ ทำให้ข้อจำกัดด้านขนาดท่อส่งข้อมูลลดลง แต่ผลที่ได้จะเห็นได้ว่า SGA (S) ให้คำตอบที่สูงกว่า MPPF ในช่วงดังกล่าว ดังนั้นจึงเป็นเหตุผลให้มีการปรับปรุงการทำงานของ SGA (S) ให้มีประสิทธิภาพมากยิ่งขึ้น โดยการนำวิธีฮิวริสติกเข้ามาช่วย

อัลกอริทึมที่เกิดจากการปรับปรุง SGA (S) โดยการใช้วิธีครอสโอเวอร์และมิวเทชันแบบฮิวริสติกเข้ามาช่วยคือ HGA (S) ผลที่ได้ของ HGA (S) สามารถแก้ไขจุดบกพร่องของ SGA (S) ในด้าน การหาค่าต้นทุนรวม และความเร็วในการหาคำตอบ รูปที่ 6.6



รูปที่ 6.6 แสดงผลการเปรียบเทียบค่าต้นทุนรวมระหว่าง SGA (S) และ HGA (S)

รูปที่ 6.6 เป็นการแสดงผลการเปรียบเทียบค่าต้นทุนรวมระหว่าง SGA (S) และ HGA (S) โดยเปรียบเทียบในขณะที่เครือข่ายสามารถหาคำตอบของปัญหาได้ โดยที่ใช้ขนาดท่อส่งข้อมูลเท่ากัน ซึ่งทั้งสองอัลกอริทึมนี้สามารถหาคำตอบได้เท่ากัน แตกต่างกันเฉพาะผลลัพธ์ค่าต้นทุนรวมของเครือข่าย จากผลการทดลองแสดงให้เห็นว่า อัลกอริทึม HGA (S) ที่ได้รับการปรับปรุงแล้วนั้นสามารถหาค่าต้นทุนรวม ตั้งแต่ขนาดท่อส่งข้อมูล 110 หน่วยจนถึง 500 หน่วยได้ต่ำกว่า SGA (S) โดยเฉลี่ยแล้ว 0.34% และเมื่อนำทั้ง 3 อัลกอริทึมมาเปรียบเทียบกันแล้วผลที่ได้แสดงอยู่ในรูปที่ 6.7



รูปที่ 6.7 แสดงผลการเปรียบเทียบค่าต้นทุนรวมระหว่าง SGA (S), HGA (S) และ MPPF

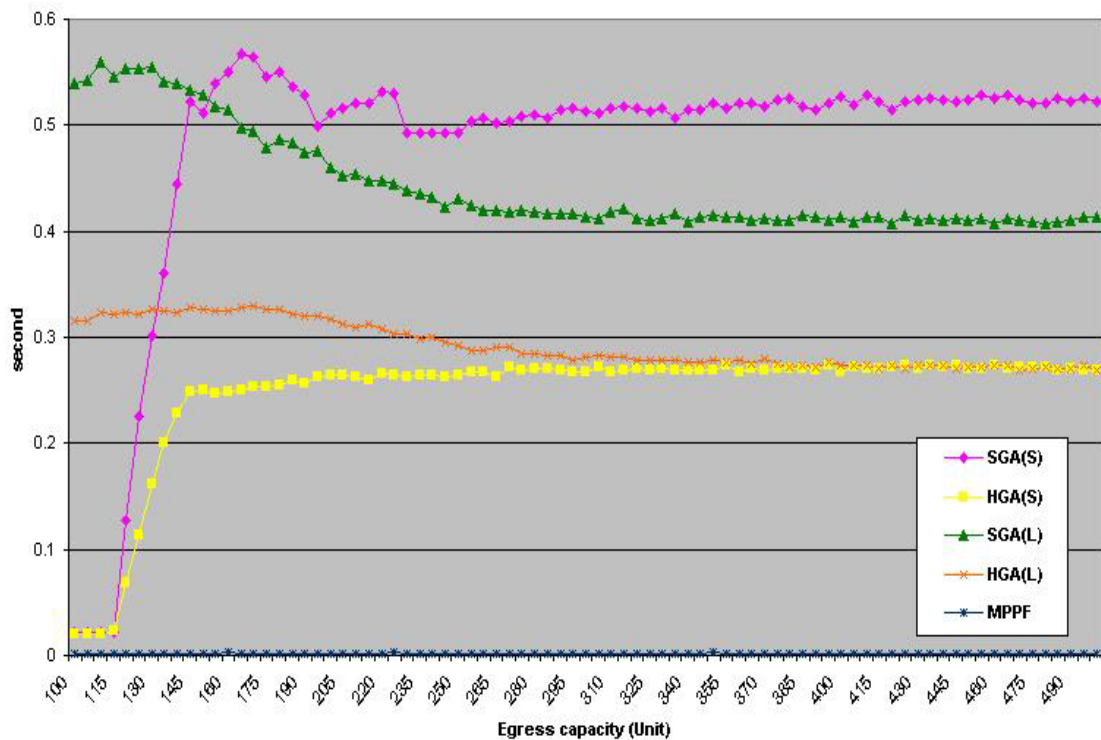
สำหรับการเปรียบเทียบค่าต้นทุนรวมของทั้ง 3 อัลกอริทึม จะมีการเปรียบเทียบเฉพาะเครือข่ายที่สามารถค้นพบคำตอบ โดยที่ใช้ทอส่งข้อมูลขนาดเท่ากัน ซึ่งจากผลการทดลองแสดงให้เห็นว่า ในกรณีที่ขนาดของทอส่งข้อมูลมีขนาดเล็ก เช่นขนาดทอส่งข้อมูลตั้งแต่ 110 หน่วยถึง 250 หน่วยค่าต้นทุนรวมของเครือข่ายที่เป็นคำตอบของอัลกอริทึม HAG (S) มีค่าต่ำกว่า MPPF 0.67% แต่เมื่อขนาดทอส่งข้อมูลเพิ่มไปเรื่อยๆ จะเห็นได้ว่าทุกอัลกอริทึมจะค้นพบคำตอบที่เข้าใกล้ค่าที่เหมาะสมที่สุดเหมือนกัน ซึ่งจะเห็นได้ว่าอัลกอริทึม MPPF และ HGA (S) ให้ค่าที่เท่ากันเมื่อในกรณีนี้ แต่ SGA (S) จะให้ค่าสูงกว่าเล็กน้อย แต่สำหรับการเพิ่มขนาดของทอส่งข้อมูลมากขึ้นเท่าไร เครือข่ายก็จะต้องมีต้นทุนการดำเนินงานเพิ่มมากขึ้นตามไปด้วย

6.3.3 การเปรียบเทียบด้านเวลาที่ใช้ในการค้นหาคำตอบ

สำหรับการเปรียบเทียบด้านเวลาที่ใช้ในการค้นหาคำตอบ จะเป็นการเปรียบเทียบด้านประสิทธิภาพของอัลกอริทึม ดังนั้นการทดลองจึงต้องทำอยู่บนสภาวะแวดล้อมเดียวกัน โดยได้ทำการทดลองบนเครื่องคอมพิวเตอร์ ซึ่งมีรายละเอียดของเครื่องดังนี้

1. ซีพียู Intel Pentium4 2.6 GHz
2. หน่วยความจำ 512MB
3. ฮาร์ดดิสก์ 40GB
4. ระบบปฏิบัติการ Windows XP

อัลกอริทึมที่ใช้ในการเปรียบเทียบได้แก่ MPPF BTf SGA (S) SGA (L) และ HGA (S) HGA (L) ผลการทดลองแสดงอยู่ในรูปที่ 6.8



รูปที่ 6.8 แสดงเวลาการทำงานของแต่ละอัลกอริทึม

จากรูปที่ 6.8 แสดงผลการทดลองด้านเวลาที่ใช้ในการทำงานของอัลกอริทึม MPPF และจินติก ซึ่งประกอบไปด้วย SGA (S) และ HGA (S) แบบการสุ่มประชากรเริ่มต้นโดยคำนึงถึงเงื่อนไขของปัญหา และ SGA (L) และ HGA (L) แบบการสุ่มประชากรเริ่มต้นโดยไม่คำนึงถึงเงื่อนไขของปัญหา แกนแนวนอนแทนขนาดท่อส่งข้อมูล แกนแนวตั้งแทนเวลาการทำงานซึ่งมีหน่วยเป็นวินาที จากผลการทดลองจะเห็นได้ว่าอัลกอริทึม MPPF ใช้เวลาในการทำงานเพียง 0.002 วินาที แต่อัลกอริทึม SGA (S) ใช้เวลาโดยเฉลี่ย 0.5 วินาที ส่วน SGA (L) ใช้เวลา 0.4 วินาที และอัลกอริทึม HGA (S) ซึ่งได้ทำการปรับปรุงประสิทธิภาพการทำงานจาก SGA แล้วทำให้เวลาการทำงานโดยเฉลี่ย 0.25 วินาที และ HGA (L) ที่ใช้เวลาใกล้เคียงกับ HGA (S)

สำหรับข้อแตกต่างกันระหว่างจินติกอัลกอริทึมแบบการสุ่มประชากรเริ่มต้นโดยคำนึงถึงเงื่อนไขของปัญหาและไม่คำนึงถึงเงื่อนไขของปัญหานั้นจะเห็นได้ว่า ในช่วงที่ไม่สามารถหาคำตอบได้ คือช่วงที่ท่อส่งข้อมูลมีขนาด 100 ถึง 115 จะเห็นจากผลการทดลองได้ว่า การที่สุ่มประชากรเริ่มต้น 30000 ครั้งสำหรับการค้นหาประชากรที่ถูกต้องในเรื่องของการส่งข้อมูลไม่ให้เกินขนาดท่อส่งข้อมูล ใช้เวลาน้อยกว่า การที่เริ่มต้นจากประชากรที่ไม่ถูกต้องที่สุ่มแบบให้การส่งข้อมูลสามารถเกินขนาดท่อส่งข้อมูลได้ถึง 0.5 วินาที ซึ่งสรุปได้ว่า จินติกแบบแบบการสุ่มประชากรเริ่มต้นโดยคำนึงถึงเงื่อนไขของปัญหานั้นใช้เวลาการทำงานน้อยกว่าแบบแบบการสุ่มประชากรเริ่มต้นโดยไม่คำนึงถึงเงื่อนไขของปัญหา