

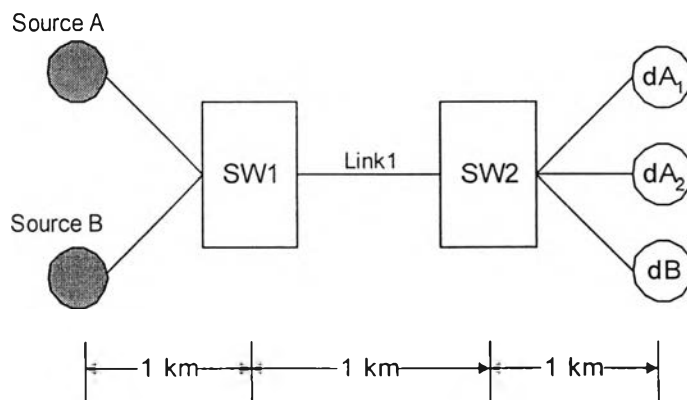
บทที่ 5

ผลการจำลองแบบ

5.1 ผลการจำลองสำหรับ One hop configuration

5.1.1 รูปแบบโครงข่ายและค่ากำหนดต่างๆ

โครงข่ายลักษณะนี้เป็นโครงข่ายรูปแบบที่ง่ายที่สุด และเป็นพื้นฐานในการทดสอบวิธีการใหม่ ๆ โครงข่ายประกอบด้วยแหล่งกำเนิด 2 แหล่ง ได้แก่ แหล่งกำเนิด A ซึ่งเป็นแหล่งกำเนิดข้อมูลสำหรับการส่งจากหนึ่งจุดไปสู่หลายจุด โดยจะส่งข้อมูลให้แก่ปลายทาง dA_1 และ dA_2 ตามลำดับ และแหล่งกำเนิด B ซึ่งส่งข้อมูลให้แก่ปลายทาง dB โดยทำหน้าที่เป็นแหล่งกำเนิดสำหรับการส่งจากหนึ่งจุดไปสู่หนึ่งจุด รูปแบบโครงข่ายจะเป็นดังรูปที่ 5.1



รูปที่ 5.1 One hop configuration

ค่าตัวแปรต่าง ๆ ที่ใช้ในรูปแบบโครงข่ายนี้ได้แก่

Source A : PCR = 149.76 Mbps , ICR = 149.76 Mbps, MCR= 0 Mbps, RIF= 1 , RDF = 0.0125 และ Nrm = 32

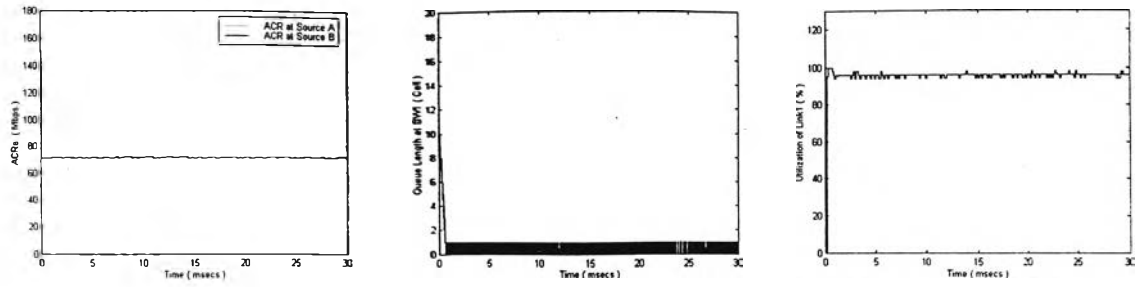
Source B : PCR = 149.76 Mbps , ICR = 149.76 Mbps, MCR= 0 Mbps, RIF= 1 , RDF = 0.0125 และ Nrm = 32

SW1 : Maximum queue length = 1,000

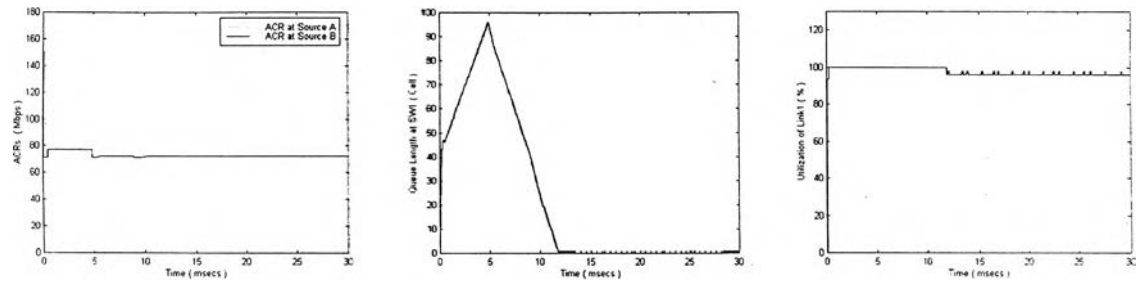
SW2 : Maximum queue length = 100

5.1.2 สมรรถนะของวิธี ERICA

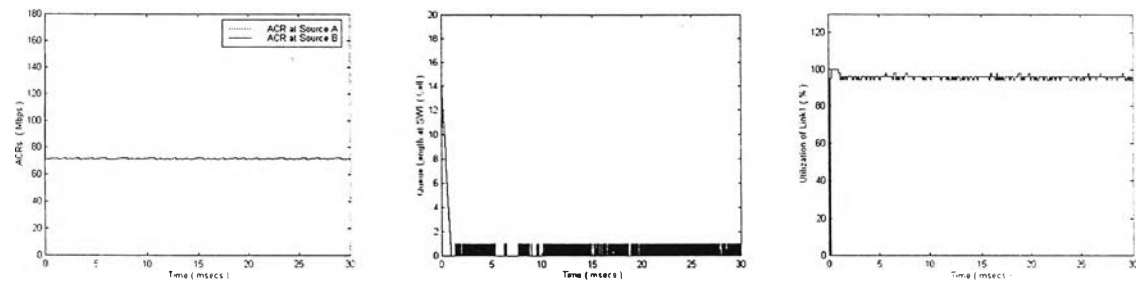
เมื่อใช้ ERICA ร่วมกับวิธีรวบรวมความคับคั่งต่างๆ จะได้ผลดังรูปที่ 5.2 ถึง 5.8



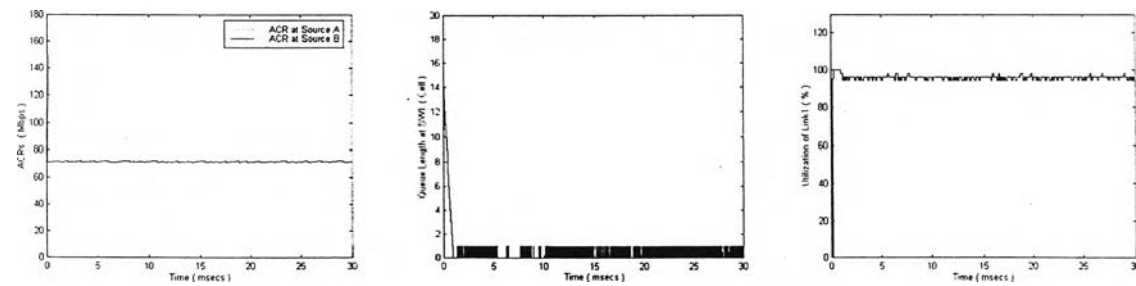
รูปที่ 5.2 ผลการจำลองเมื่อใช้วิธี ERICA ร่วมกับ Robert Algorithm สำหรับ one hop configuration



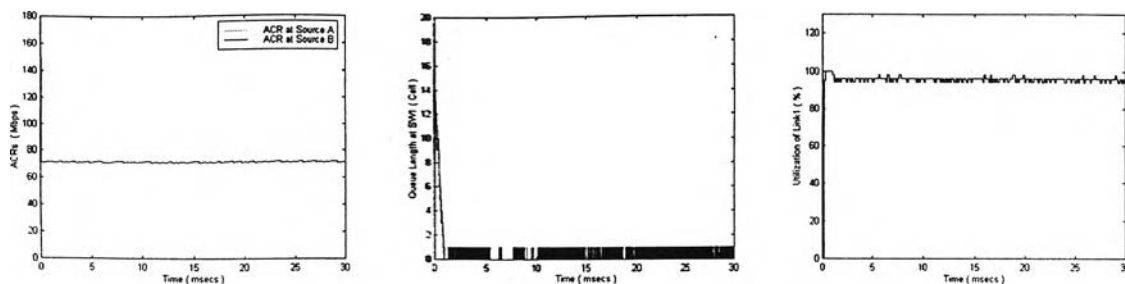
รูปที่ 5.3 ผลการจำลองเมื่อใช้วิธี ERICA ร่วมกับ TS Algorithm สำหรับ one hop configuration



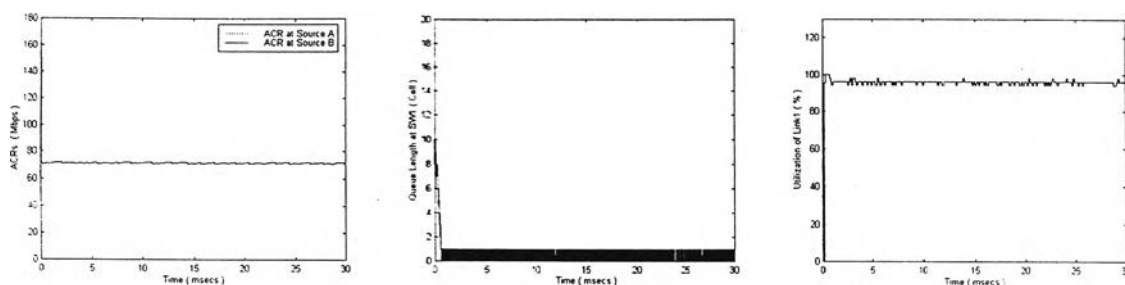
รูปที่ 5.4 ผลการจำลองเมื่อใช้วิธี ERICA ร่วมกับ RSS Algorithm สำหรับ one hop configuration



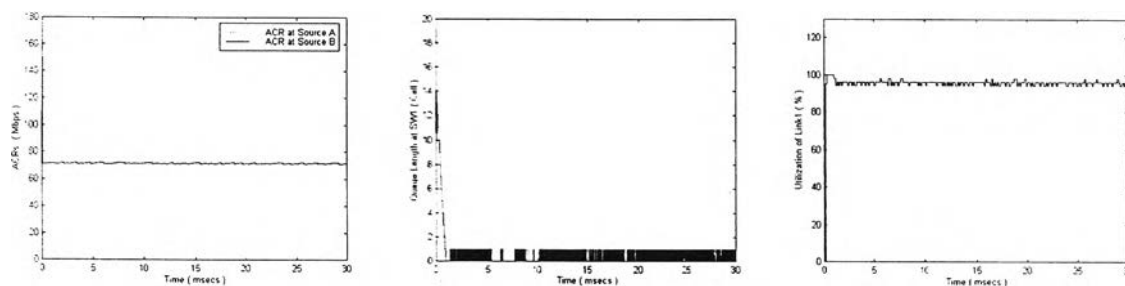
รูปที่ 5.5 ผลการจำลองเมื่อใช้วิธี ERICA ร่วมกับ Wait-for-all Algorithm สำหรับ one hop configuration



รูปที่ 5.6 ผลการจำลองเมื่อใช้วิธี ERICA ร่วมกับ Immediate Rate Calculation Algorithm สำหรับ one hop configuration



รูปที่ 5.7 ผลการจำลองเมื่อใช้วิธี ERICA ร่วมกับวิธีการที่เสนอแบบที่ 1 สำหรับ one hop configuration



รูปที่ 5.8 ผลการจำลองเมื่อใช้วิธี ERICA ร่วมกับวิธีการที่เสนอแบบที่ 2 สำหรับ one hop configuration

จุดประสงค์หลักของการทดสอบกับรูปแบบโครงข่ายนี้ เพื่อแสดงถึงความสามารถของวิธีรวบรวมข่าวสารความคับคั่งแต่ละวิธีว่าสามารถใช้งานวิธีการ ERICA และ FMMRA บนโครงข่าย ATM แบบง่ายที่สุดโดยไม่ก่อให้เกิดปัญหาใด ๆ

จากรูปที่ 5.2 สังเกตได้ว่าเมื่อใช้ ERICA กับ Robert algorithm จะเห็นได้ว่าค่า ACR ที่แหล่งกำเนิดทั้งสองนั้นมีค่าลดลงจากค่า ICR คือ 149.76 Mbps ลงมาสู่ fairshare อย่างรวดเร็ว ทั้งนี้เนื่องจากโครงข่ายมีขนาดเล็ก และมี propagation delay น้อย ทำให้เวลาที่ใช้ในการป้อนกลับมีค่าน้อย แหล่งกำเนิดสามารถปรับอัตราส่งให้เหมาะสมที่ fairshare ได้อย่างรวดเร็ว โดยค่า fairshare ของแต่ละแหล่งกำเนิดสำหรับรูปแบบโครงข่ายนี้คือ $149.76 / 2 = 74.88$ Mbps อย่างไรก็ตาม ERICA นั้นมีการเพื่อการใช้งานสายสัญญาณไว้ที่ค่า target utilization ซึ่งในรูปแบบโครงข่ายนี้ตั้งค่าไว้ที่ 95% ERICA จะนำค่าดังกล่าวมาทำการคำนวณแบนวิดท์ที่ใช้ได้จริงก่อนการคำนวณ fairshare และทำให้ค่า fairshare สำหรับแต่ละแหล่งกำเนิดเป็น $149.76 \times 0.95 / 2 = 71.14$ Mbps และเมื่อโครงข่ายอยู่ในสภาวะอยู่ตัวค่าอัตราส่งที่แหล่งกำเนิดมีค่างที่ค่าประมาณ fairshare นั้นเอง

สำหรับขนาดหน่วยความจำสูงสุดที่ SW1 นั้นมีค่าเพียง 10 เซลล์ ในช่วงแรกนั้น โครข่ายจะอยู่ในสถานะเกิดความคับคั่งขึ้นเนื่องจาก link1 ไม่สามารถรองรับข้อมูลที่ส่งมาจากแหล่งกำเนิดทั้งสองรวมกันได้ ทั้งนี้เนื่องจากแหล่งกำเนิดทั้งสองนั้นมีค่า ICR มากเกินไปนั่นเอง อย่างไรก็ตามการที่ขนาดหน่วยความจำสูงสุดที่ SW1 มีค่าน้อยนั้นเกิดจากการป้อนกลับที่รวดเร็ว และแหล่งกำเนิดสามารถปรับอัตราส่งได้อย่างรวดเร็ว ทำให้โครข่ายหลุดพ้นจากสถานะความคับคั่งได้รวดเร็ว

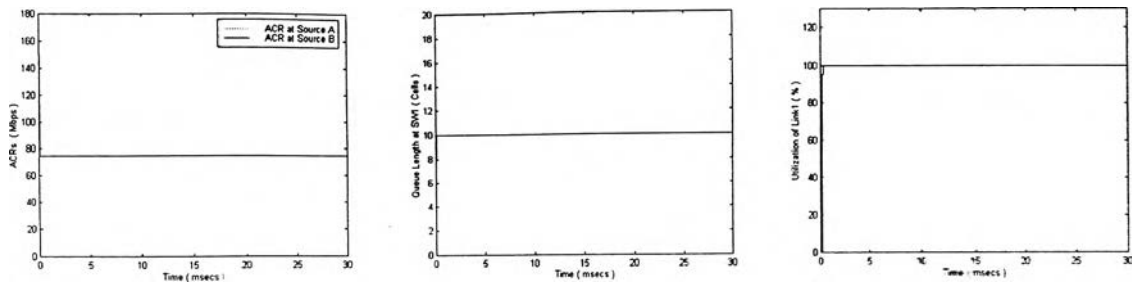
Robert Algorithm นั้นมีเวลาในการป้อนกลับรวดเร็วที่สุดในบรรดาวิธีรวบรวมข่าวสารความคับคั่ง ด้วยความสามารถในการป้อนกลับข่าวสารในเซลล์ RM ทันทีที่ได้รับเซลล์ FRM ทำให้เมื่อ SW1 ได้รับเซลล์ FRM จาก Source A จะทำการป้อนกลับข่าวสารความคับคั่งทันที และแหล่งกำเนิด A นั้นจะปรับอัตราส่งได้อย่างรวดเร็ว ทำให้โครข่ายออกจากสถานะคับคั่งได้รวดเร็ว ดังนั้นขนาดหน่วยความจำสูงสุดจึงมีค่าน้อยที่สุดด้วย ส่วนค่าการใช้งานสายสัญญาณนั้นจะมีค่าเป็น 1 ในช่วงแรก เนื่องจากในช่วงแรกแหล่งกำเนิดทั้ง 2 นั้นส่งข้อมูลที่อัตราส่งที่รวมกันสูงกว่าที่สายสัญญาณจะรองรับได้ ทำให้มีการใช้งานสายสัญญาณได้เต็มที่ โดยส่วนที่สายสัญญาณไม่สามารถรองรับได้นั้นจะนำไปเก็บไว้ในหน่วยความจำที่สวิตช์ และเมื่อสวิตช์ไม่เกิดความคับคั่ง เพราะกลไกการควบคุมความคับคั่ง ทำให้การใช้งานสายสัญญาณตกลงมาที่ประมาณ 95% ซึ่งเป็นค่า target utilization ที่ ERICA ตั้งไว้นั่นเอง

เมื่อใช้ ERICA กับ RSS algorithm, Wait-for-all algorithm, Immediate rate calculation algorithm และ วิธีการที่เสนอแบบที่ 2 จะได้ผลตามรูปที่ 5.4, 5.5, 5.6 และ 5.8 ตามลำดับ จะเห็นว่าผลการจำลองที่ได้นั้นคล้ายคลึงกับเมื่อใช้ Robert algorithm เพียงแต่ขนาดหน่วยความจำสูงสุดที่ใช้นั้นมีค่า 14 เซลล์ เนื่องจากกลไกการป้อนกลับข่าวสารความคับคั่งของวิธีรวบรวมข่าวสารความคับคั่งต่าง ๆ ข้างต้น มีการทำงานคล้ายกันในกรณีนี้ โดย SW1 จำเป็นต้องรอให้มี BRM เข้ามาก่อนจึงสามารถป้อนกลับข่าวสารความคับคั่งได้ ซึ่งทำให้เวลาที่ใช้ในการป้อนกลับครั้งแรกนั้นช้ากว่า Robert algorithm โครข่ายจะอยู่ในสถานะคับคั่งนานกว่า และทำให้ขนาดหน่วยความจำสูงสุดที่ใช้นั้นมากกว่า อย่างไรก็ตามจะเห็นว่าขนาดหน่วยความจำสูงสุดที่ใช้นั้นยังมีค่าน้อยมากเพียง 14 เซลล์ ซึ่งถือว่าเป็นค่าที่ยอมรับได้

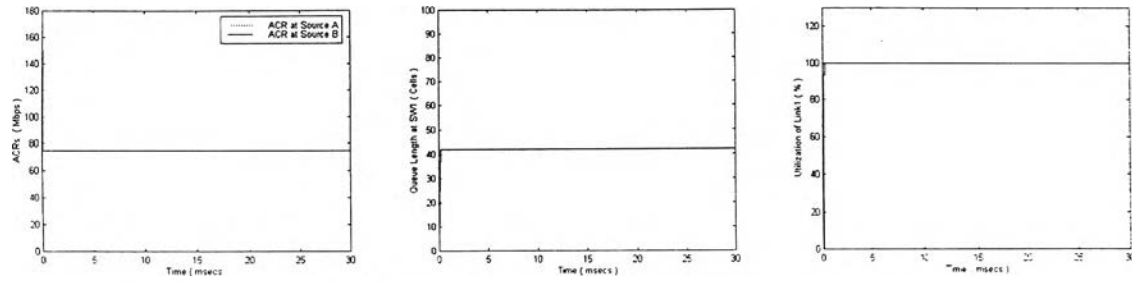
สำหรับวิธีการที่เสนอแบบที่ 1 นั้นจะได้ผลตามรูปที่ 5.7 ซึ่งจะเห็นว่าผลการจำลองนั้นคล้ายคลึงกับ Robert algorithm มาก เนื่องจากวิธีนี้ใช้เงื่อนไขในการตอบกลับข่าวสารความคับคั่งแบบเดียวกับ Robert algorithm ทำให้ผลการทดสอบสำหรับโครข่ายง่าย ๆ เช่นนี้มีผลลัพธ์ที่ไม่แตกต่างกัน

5.1.3 สมรรถนะของวิธี FMMRA

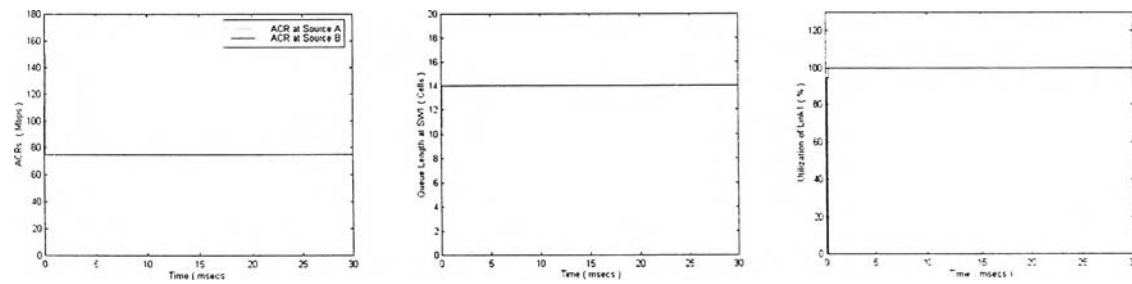
เมื่อใช้ FMMRA ร่วมกับวิธีรวบรวมความคับคั่งต่าง ๆ จะได้ผลดังรูปที่ 5.9 ถึง 5.15



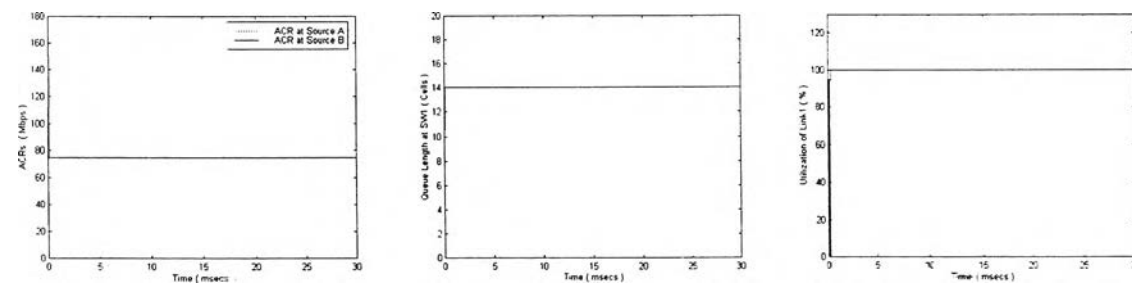
รูปที่ 5.9 ผลการจำลองเมื่อใช้วิธี FMMRA ร่วมกับ Robert Algorithm สำหรับ one hop configuration



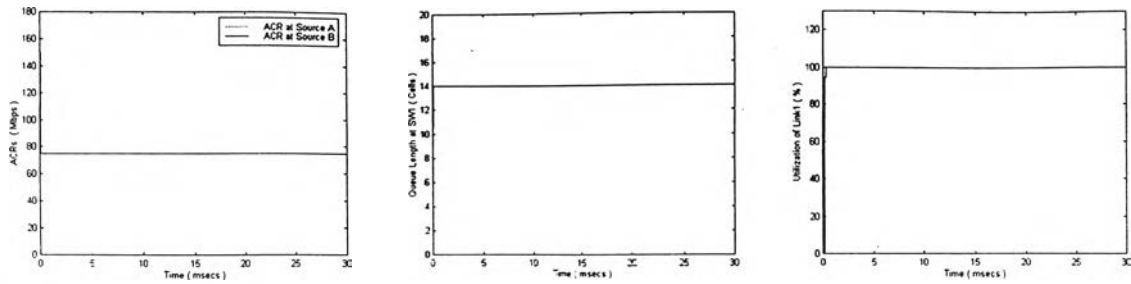
รูปที่ 5.10 ผลการจำลองเมื่อใช้วิธี FMMRA ร่วมกับ TS Algorithm สำหรับ one hop configuration



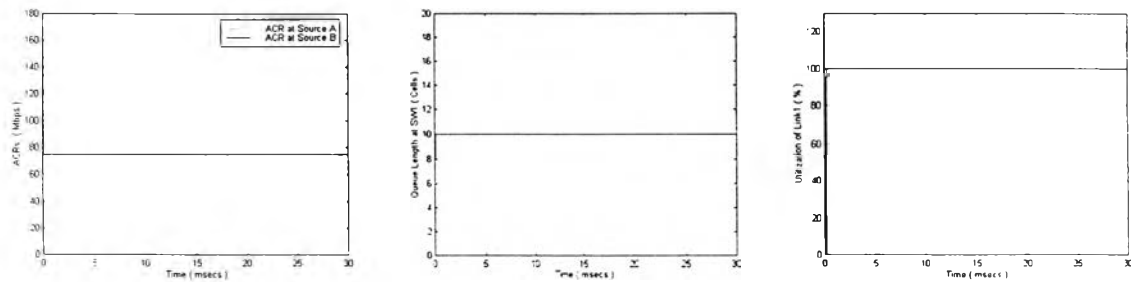
รูปที่ 5.11 ผลการจำลองเมื่อใช้วิธี FMMRA ร่วมกับ RSS Algorithm สำหรับ one hop configuration



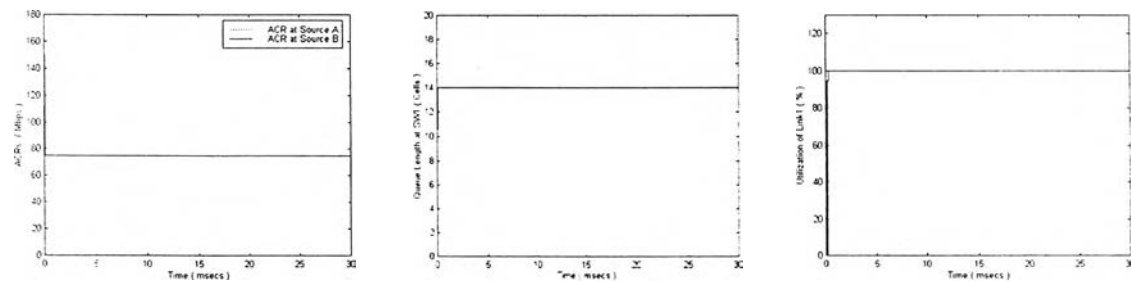
รูปที่ 5.12 ผลการจำลองเมื่อใช้วิธี FMMRA ร่วมกับ Wait-for-all Algorithm สำหรับ one hop configuration



รูปที่ 5.13 ผลการจำลองเมื่อใช้วิธี FMMRA ร่วมกับ Immediate Rate Calculation Algorithm สำหรับ one hop configuration



รูปที่ 5.14 ผลการจำลองเมื่อใช้วิธี FMMRA ร่วมกับวิธีการที่เสนอแบบที่ 2 สำหรับ one hop configuration



รูปที่ 5.15 ผลการจำลองเมื่อใช้วิธี FMMRA ร่วมกับวิธีการที่เสนอแบบที่ 1 สำหรับ one hop configuration

จากรูปที่ 5.9 และ 5.15 สังเกตได้ว่าเมื่อใช้ FMMRA กับ Robert algorithm และ วิธีการที่เสนอแบบที่ 1 ซึ่งให้ผลการจำลองที่เหมือนกันในรูปแบบโครงข่ายนี้ จะเห็นได้ว่าค่า ACR ที่แหล่งกำเนิดทั้งสองนั้นมีค่าลดลงจากค่า ICR คือ 149.76 Mbps ลงมาสู่ fairshare อย่างรวดเร็วเช่นเดียวกับกรณีที่ใช้ ERICA ทั้งนี้เนื่องจากโครงข่ายมีขนาดเล็ก และมี propagation delay น้อย โดยค่า fairshare ของแต่ละแหล่งกำเนิดสำหรับรูปแบบโครงข่ายนี้คือ $149.76 / 2 = 74.88$ Mbps และเนื่องจาก FMMRA นั้นไม่มีการตั้งค่าการใช้งานสายสัญญาณที่ต้องการไว้ ทำให้ค่า fairshare นั้นอยู่ที่ 74.88 Mbps และ SW1 นำค่าดังกล่าวมาใช้ในการป้อนกลับไปสู่แหล่งกำเนิด ดังนั้นแหล่งกำเนิดจึงส่งข้อมูลด้วยอัตราส่งดังกล่าวตลอดเวลา ซึ่งหากพิจารณาที่ SW1 จะเห็นว่าปริมาณทราฟฟิกขาเข้ารวมจากทั้งสองแหล่งกำเนิด และปริมาณความจุขาออกไป link1 นั้นมีค่าเท่ากัน ทำให้มีการใช้งานสายสัญญาณ link1 ได้เต็มที่ตามรูปที่ 5.9 ในขณะที่ขนาดหน่วยความจำที่สวิตช์นั้นมีค่าเพิ่มขึ้นในช่วงแรก เนื่องจากสถานะคับคั่งในช่วงแรกของ SW1 และมีค่าคงที่ที่ 10 เซลล์ตลอด ทั้งนี้เนื่องจากปริมาณทราฟฟิกขาเข้าและขาออกนั้นมีปริมาณเท่ากัน ส่วนขนาดหน่วยความจำสูงสุดนั้นมีค่าเป็น 10 เซลล์ เช่นเดียวกับในกรณี ERICA

สำหรับ TS algorithm จะได้ผลการจำลองตามรูปที่ 5.10 สังเกตได้ว่าค่าขนาดหน่วยความจำสูงสุดที่ SW1 นั้นมีค่าสูงกว่าเมื่อใช้ Robert algorithm ทั้งนี้เนื่องจากในช่วงแรกนั้น โครงข่ายอยู่ในสถานะคับคั่ง และ TS algorithm นั้นมีเงื่อนไขในการตอบกลับข่าวสารความคับคั่งในเซลล์ RM คือ ต้องได้รับเซลล์ FRM หลังจากได้รับเซลล์ BRM แล้ว ซึ่งทำให้ SW1 ไม่สามารถตอบกลับข่าวสารความคับคั่งได้ ในขณะที่ได้รับเซลล์ FRM ตัวแรก และ SW1 จำเป็นต้องรอเซลล์ FMR ตัวที่สองจากแหล่งกำเนิด A ทำให้โครงข่ายอยู่ในสถานะคับคั่งนานขึ้น ส่งผลให้ขนาดหน่วยความจำสูงสุดมีค่ามากขึ้นไปด้วย อย่างไรก็ตาม ขนาดหน่วยความจำยังคงมีค่าคงที่ที่ประมาณ 40 เซลล์

เมื่อใช้ FMMRA กับ RSS algorithm, Wait-for-all algorithm, Immediate rate calculation algorithm และ วิธีการที่เสนอแบบที่ 2 จะได้ผลตามรูปที่ 5.11, 5.12, 5.13 และ 5.15 ตามลำดับ จะเห็นว่า ได้ผลคล้ายคลึงกับเมื่อใช้ Robert algorithm เพียงแต่ขนาดหน่วยความจำสูงสุดที่ใช้มีค่า 14 เซลล์ เนื่องจากกลไกการป้อนกลับข่าวสารความคับคั่งของวิธีรวบรวมข่าวสารความคับคั่งต่าง ๆ ข้างต้น มีการทำงานคล้ายกันในกรณีนี้

5.1.4 เปรียบเทียบประสิทธิภาพระหว่าง ERICA และ FMMRA

ทั้ง FMMRA และ ERICA นั้นสามารถปรับอัตราส่งที่แหล่งกำเนิดไปสู่อัตราส่งที่โครงข่ายต้องการได้อย่างยุติธรรม หากแต่ ERICA นั้นมีการตั้งค่าการใช้งานสายสัญญาณที่ต้องการไว้ ทำให้สามารถลดระดับหน่วยความจำที่สวิตช์ได้ แต่ก็เป็นการทำให้การใช้งานสายสัญญาณ link1 นั้นเป็นไปได้ไม่เต็มที่ 100% ในขณะที่ FMMRA นั้นไม่มีการตั้งค่าดังกล่าว ทำให้ปริมาณทราฟฟิกขาเข้าและความจุขาออกมีค่าเท่ากัน ส่งผลให้ขนาดของหน่วยความจำที่สวิตช์นั้นมีค่าคงที่ ซึ่งเป็นผลเสีย ในกรณีที่มีการเชื่อมต่อใหม่อื่น ๆ ที่เข้ามาสู่โครงข่าย จะทำให้โครงข่ายเข้าสู่สถานะคับคั่งอีก ในขณะที่ไม่สามารถลดระดับหน่วยความจำได้

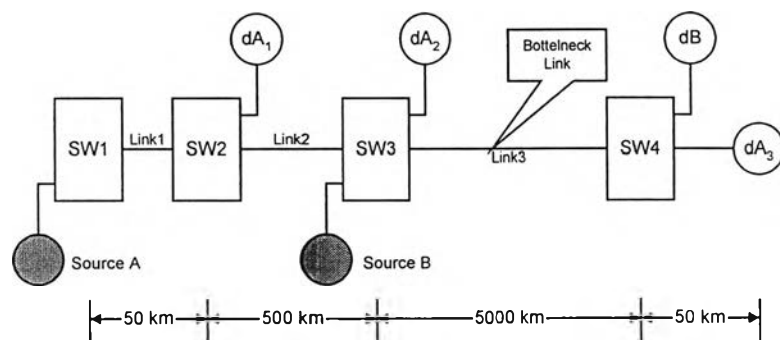
5.1.5 เปรียบเทียบประสิทธิภาพระหว่างวิธีรวบรวมข่าวสารความคับคั่งต่าง ๆ

ในรูปแบบโครงข่ายนี้วิธีรวบรวมข่าวสารความคับคั่งต่าง ๆ นั้นให้ผลที่ไม่แตกต่างกันมากนัก ทั้งนี้เนื่องจากเป็นโครงข่ายที่ง่าย และมีขนาดเล็ก ทำให้ผลการจำลองต่าง ๆ มีลักษณะคล้าย ๆ กัน มีเพียง TS algorithm เท่านั้นที่ให้ผลที่แย่ที่สุด โดยมีขนาดหน่วยความจำที่สวิตช์สูงที่สุด

5.2 ผลการจำลองสำหรับ Chain configuration

5.2.1 รูปแบบโครงข่ายและค่ากำหนดต่าง ๆ

รูปแบบโครงข่ายนี้เป็นโครงข่ายที่เหมาะสมสำหรับการทดสอบปัญหา Consolidation noise ได้เป็นอย่างดี ประกอบด้วย VC 2 ชุด ได้แก่ VC1 ของแหล่งกำเนิด A ซึ่งเป็น point-to-multipoint connection และ VC2 ของแหล่งกำเนิด B ซึ่งเป็น point-to-point connection โดยกราฟฟิกจะถูกรวมกันที่สวิตช์ 3 ดังนั้นจะมีจุดที่อาจเกิดปัญหาความคับคั่งได้เพียงจุดเดียวคือ link3 เมื่อระบบเข้าสู่สถานะอยู่ตัวค่า ACR ที่เหมาะสมสำหรับทั้งสอง VC ควรจะมีค่าประมาณเท่ากับครึ่งหนึ่งของความจุของ link3 โดย link3 มีความจุ 149.76 Mbps จะได้ค่า fairshare เป็น $149.76/2 = 74.88$ Mbps เพื่อให้การใช้งานสายสัญญาณมีประสิทธิภาพสูงสุดและมีความเท่าเทียมกันโดยไม่เกิดความคับคั่งขึ้น รูปแบบโครงข่ายจะเป็นดังรูปที่ 5.16



รูปที่ 5.16 Chain configuration

ค่าตัวแปรต่าง ๆ ที่ใช้ในรูปแบบโครงข่ายนี้ได้แก่

Source A : PCR = 149.76 Mbps , ICR = 149.76 Mbps, MCR= 0 Mbps, RIF= 1 , RDF = 0.0125 และ Nrm = 32

Source B : PCR = 149.76 Mbps , ICR = 149.76 Mbps, MCR= 0 Mbps, RIF= 1 , RDF = 0.0125 และ Nrm = 32

SW1 : Maximum queue length = 40,000

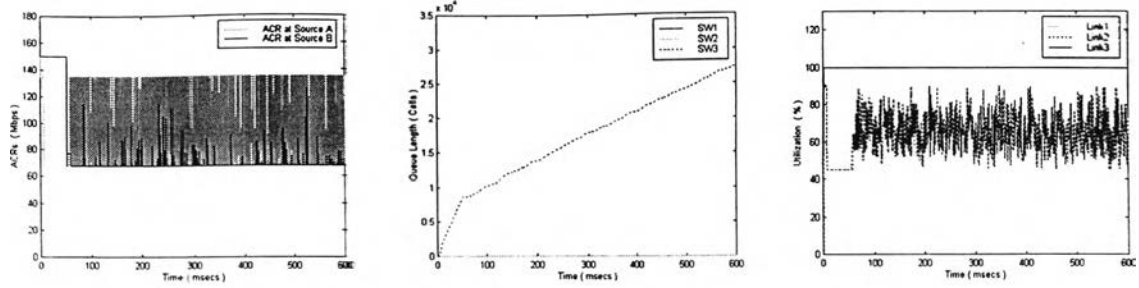
SW2 : Maximum queue length = 40,000

SW3 : Maximum queue length = 40,000

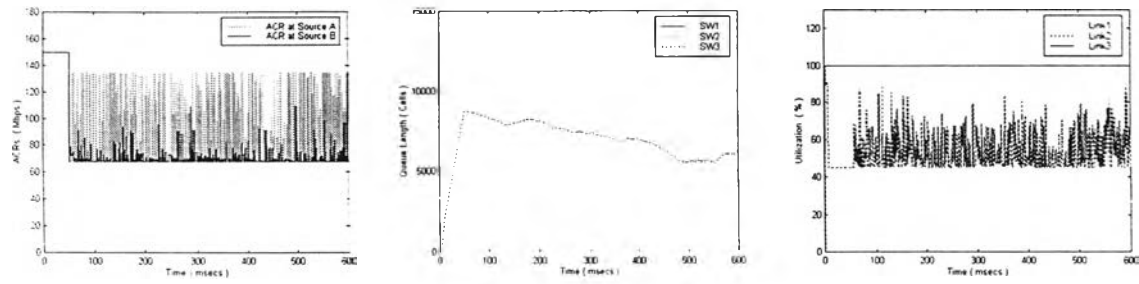
SW4 : Maximum queue length = 40,000

5.2.2 สมรรถนะของวิธี ERICA

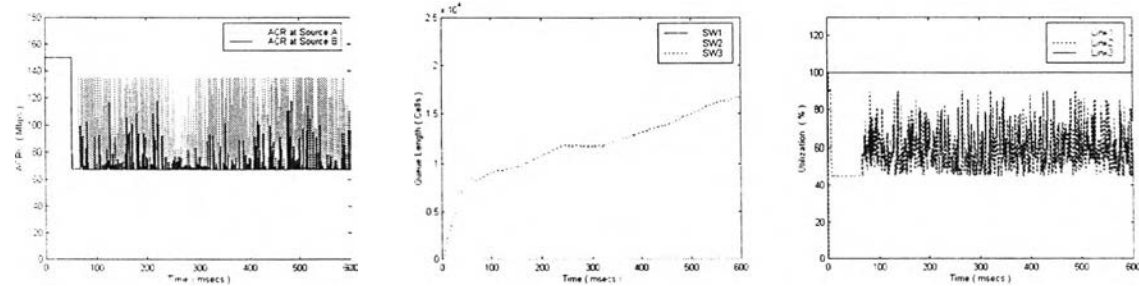
เมื่อใช้ ERICA ร่วมกับวิธีรวบรวมความคับคั่งต่าง ๆ จะได้ผลดังรูปที่ 5.17 ถึง 5.23



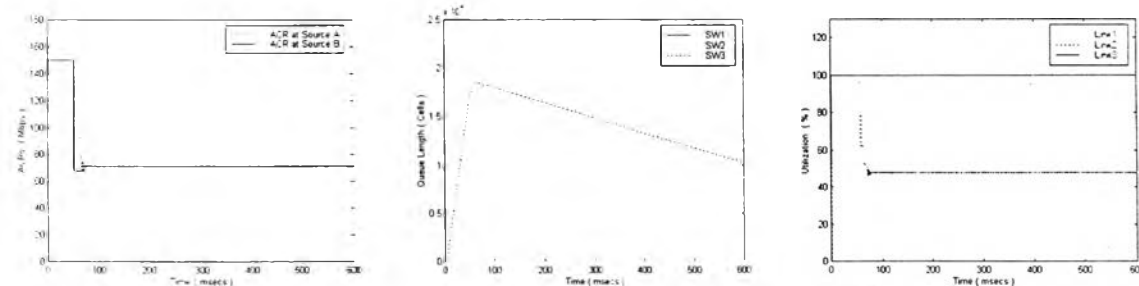
รูปที่ 5.17 ผลการจำลองเมื่อใช้วิธี ERICA ร่วมกับ Robert Algorithm สำหรับ Chain configuration



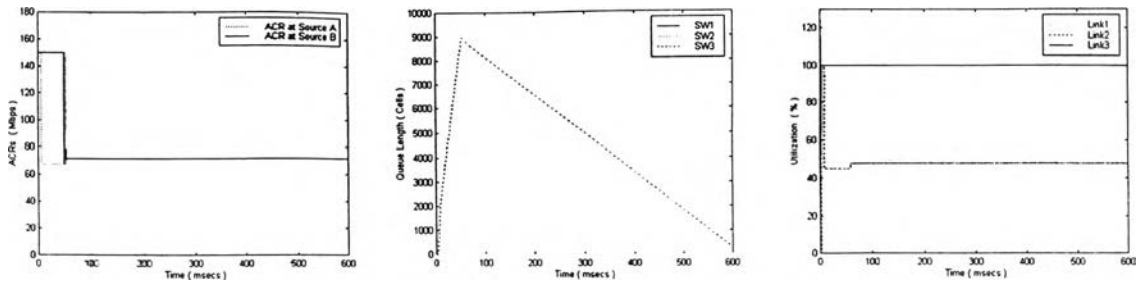
รูปที่ 5.18 ผลการจำลองเมื่อใช้วิธี ERICA ร่วมกับ TS Algorithm สำหรับ Chain configuration



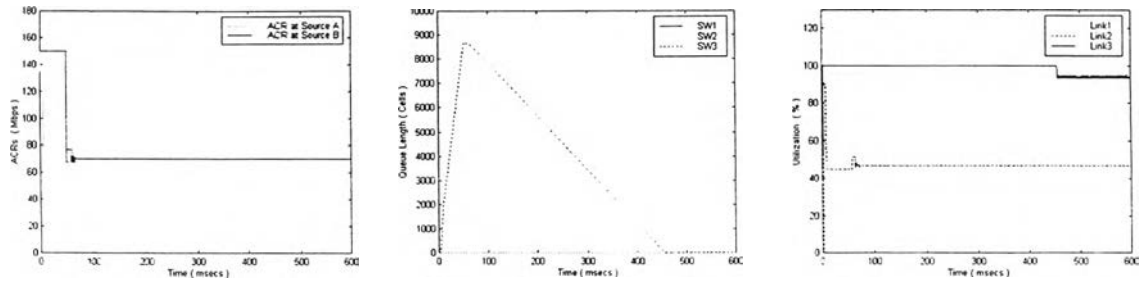
รูปที่ 5.19 ผลการจำลองเมื่อใช้วิธี ERICA ร่วมกับ RSS Algorithm สำหรับ Chain configuration



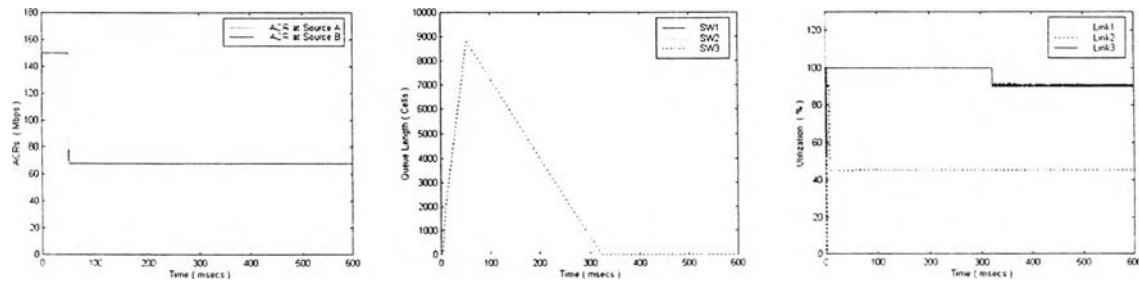
รูปที่ 5.20 ผลการจำลองเมื่อใช้วิธี ERICA ร่วมกับ Wait-for-all Algorithm สำหรับ Chain configuration



รูปที่ 5.21 ผลการจำลองเมื่อใช้วิธี ERICA ร่วมกับ Immediate Rate Calculation Algorithm สำหรับ Chain configuration



รูปที่ 5.22 ผลการจำลองเมื่อใช้วิธี ERICA ร่วมกับวิธีการที่เสนอแบบที่ 1 สำหรับ Chain configuration



รูปที่ 5.23 ผลการจำลองเมื่อใช้วิธี ERICA ร่วมกับวิธีการที่เสนอแบบที่ 2 สำหรับ Chain configuration

ค่า fairshare สำหรับแต่ละแหล่งกำเนิดที่ควรจะเป็นจะมีค่าเท่ากัน 74.88 Mbps แต่เนื่องจาก ERICA มีการกำหนดค่า target utilization เป็น 0.9 ทำให้ค่าอัตราส่งที่เท่าเทียมกันจะมีค่าเป็น 67.39 Mbps (0.9×74.88 Mbps) แต่อย่างไรก็ตามเนื่องจาก ERICA มีการทำงานโดยใช้ค่า z ที่อยู่ระหว่าง 1-1.1 ดังนั้นอัตราส่งที่เกิดขึ้นจริงจะมีค่าสูงกว่าค่าอัตราส่งเท่าเทียมกันเล็กน้อย

เมื่อพิจารณาค่า ACR ในรูปที่ 5.17 จะเห็นว่าผลตอบของ Robert algorithm นั้นมีผลตอบชั่วคราวที่รวดเร็วมาก ดังสังเกตได้จากค่า ACR ของแหล่งกำเนิด A ซึ่งเป็นการเชื่อมต่อแบบหนึ่งจุดไปสู่หลายจุด ในช่วงแรกค่า ACR มีค่า 149.76 Mbps ตามค่า ICR ของแหล่งกำเนิดนั้น เมื่อสวิตช์ 1 ได้รับเซลล์ FRM แล้ว ก็ทำการตอบกลับข่าวสารความคับคั่งทันที ด้วยการส่งกลับเซลล์ BRM ที่มีค่า ER ประมาณ 134.78 Mbps (149.76 Mbps คูณกับ target utilization) และ เมื่อแหล่งกำเนิดได้รับข่าวสารนี้จึงปรับค่า ACR ให้ลดลงมาเป็นประมาณ 134.78 Mbps ขั้นตอนนี้ใช้เวลารวดเร็วมากเนื่องจากระยะระหว่างแหล่งกำเนิด A กับ SW1 นั้นน้อยมาก และเมื่อ SW3 ได้รับเซลล์ FRM ตัวแรก ด้วยเงื่อนไขของ Robert algorithm ทำให้ SW3 ต้องตอบกลับเซลล์ BRM ทันที ซึ่งค่า ER ที่ส่งไปนั้นจะมีค่าประมาณ 67.39

Mbps โดย SW3 เป็นสวิตช์ที่มีทราฟฟิกมากที่สุดและเป็นสวิตช์ที่เกิดความคับคั่งขึ้นและจะเป็นสวิตช์ที่บ่งบอกถึงอัตราส่งที่แท้จริงที่แหล่งกำเนิดควรจะส่ง เมื่อแหล่งกำเนิด A ได้รับเซลล์ BRM ดังกล่าวก็จะทำการปรับลด ACR ลงมา โดยใช้เวลาในการป้อนกลับจากแหล่งกำเนิด A ไปสู่ SW3 และจาก SW3 กลับสู่ แหล่งกำเนิด A นั้นสอดคล้องกับเวลาที่ได้จากผลการจำลองคือประมาณ 6 ms โดยระยะทางที่เซลล์ RM ใช้ในเส้นทางไปและกลับจากแหล่งกำเนิดไปสู่ SW3 และจาก SW3 กลับไปสู่แหล่งกำเนิดรวมกันทั้งสิ้น 1,200 km ซึ่งคิดเป็นเวลาประมาณ 6 ms ส่วนแหล่งกำเนิด S4 นั้นมีผลตอบชั่วคราวที่ช้าเนื่องจากการเชื่อมต่อแบบหนึ่งจุดไปสู่หลายจุดทำให้ไม่จำเป็นต้องใช้วิธีรวบรวมข่าวสารความคับคั่งดังนั้นแหล่งกำเนิดจำเป็นต้องรอให้เซลล์ RM เดินทางไปสู่ปลายทาง และ เดินทางกลับมาจนถึงแหล่งกำเนิดเสียก่อนจึงสามารถปรับค่า ACR ได้

จากนั้นเมื่อแหล่งกำเนิด B เริ่มได้รับเซลล์ BRM ตัวแรกก็จะทำการปรับค่า ACR ให้ลดลง จะมีผลทำให้ค่า ACR ของ Source A เกิดการแกว่งตัวอย่างรุนแรงตามรูปที่ 5.17 โดยเรียกว่าเกิด consolidation noise ขึ้น โดย noise ที่เกิดขึ้นนั้นเกิดจากการรวบรวมข่าวสารความคับคั่งที่ไม่ดีพอ ข่าวสารความคับคั่งนั้นจะเก็บรวบรวมไว้ในตัวแปร MER และจะถูกนำมาเป็นค่า ER ใน BRM ทันทีเมื่อมีการป้อนกลับข่าวสารความคับคั่ง และค่า MER จะถูกกลบฝังให้กลายเป็นค่า ER ที่อยู่ใน เซลล์ FMR นั้นเอง ซึ่งโดยปกติแล้วค่าดังกล่าวจะมีค่าเป็น PCR ในระหว่างนั้นหากมีข่าวสารจาก branch ใด branch หนึ่งก็จะนำมาเปรียบเทียบกับค่า MER โดยเลือกเก็บค่าน้อยที่สุดไว้ใน MER หากข่าวสารที่ถูกต้องการจาก SW3 ยังเดินทางมาไม่ถึง SW1 และ SW2 ก่อนถึงกำหนดการป้อนกลับข่าวสารครั้งต่อไป ก็จะทำให้ SW1 และ SW2 นั้นจำเป็นต้องนำข่าวสารใน MER ไปป้อนกลับใน เซลล์ BRM ตัวใหม่ ซึ่งค่าดังกล่าวยังมีค่าสูงอยู่ทำให้แหล่งกำเนิด A นั้นปรับอัตราส่งเพิ่มขึ้นเกือบเท่ากับค่า PCR และ หากข่าวสารที่ถูกต้องการจาก SW3 เดินทางมาถึง SW1 และ SW2 ก็จะทำให้ข่าวสารที่จะป้อนกลับครั้งต่อไปนั้นมีความถูกต้อง ส่งผลให้แหล่งกำเนิด A นั้นจะปรับอัตราส่งไปมาจนเกิดการแกว่งขึ้น ซึ่งใน Robert Algorithm นั้นจะทำการตอบกลับข่าวสารในเซลล์ BRM ทันที เมื่อได้รับเซลล์ FRM หากสวิตช์ 1 และ 2 ได้รับเซลล์ FRM ในขณะที่ยังไม่ได้รับข่าวสารที่ถูกต้องการจากสวิตช์ 3 จะทำให้ข่าวสารที่ต้องตอบกลับทันทีในเซลล์ BRM นั้นผิดพลาด และทำให้เกิดการแกว่งของค่า ACR ขึ้น การแกว่งของค่า ACR ทำให้ Source B นั้นแกว่งตามไปด้วย และการแกว่งนั้นมีเพียงเล็กน้อย เนื่องจาก ERICA มีค่านับที่ค่า ER ที่ได้ส่งไปในเซลล์ RM ที่สุดที่ไว้ในช่วงเวลาก่อน และหากค่า z น้อยกว่า 1.1 ก็จะนำค่านีมาใช้ในการคำนวณด้วย ซึ่งค่านี้จะมีค่าสูง เนื่องจาก SW3 มีการคำนวณค่า ER จากค่า CCR ของแหล่งกำเนิด A ซึ่งมีค่าสูง ทำให้ค่า ER ที่บันทึกไว้มีค่าสูงตามไปด้วย และส่งผลให้ แหล่งกำเนิด B ได้รับผลกระทบจากค่า ER ที่บันทึกไว้ด้วย

อย่างไรก็ตาม การแกว่งตัวของค่า ACR ของแหล่งกำเนิด A นั้นจะไม่เกิดขึ้นหากว่าปริมาณเซลล์ FRM ที่ได้รับที่ SW1 และ SW2 มีค่าเท่า ๆ กับจำนวนเซลล์ BRM ที่ได้รับจาก branch ที่มีปัญหา

ทั้งนี้เนื่องจาก SW1 และ SW2 นั้นจะได้รับเซลล์ BRM จาก branch ที่มีปัญหา ก่อนการได้รับ FRM เสมอ ดังนั้นข่าวสารในเซลล์ BRM จาก branch ที่มีปัญหา จะทำให้ข่าวสารที่ตอบกลับใน BRM จาก SW1 และ SW2 จะเป็นค่าที่ถูกต้องจาก MER เสมอ การแกว่งตัวตามรูปที่ 5.17 นั้นเกิดจากการที่เมื่อแหล่งกำเนิด B เริ่มปรับค่า ACR ให้ลดลง ทำให้โครงข่ายเริ่มอยู่ในสภาวะไม่ติดขัด ทำให้ค่า z มีค่าน้อยกว่า 1 จากนั้น ERICA นั้นจะปรับอัตราส่งของแหล่งกำเนิด A และ B ขึ้นเล็กน้อยเพื่อปรับให้ค่า z มีค่าระหว่าง 1-1.1 ในจุดนี้เองที่แหล่งกำเนิด A ปรับค่า ACR เพิ่มขึ้นนั้นจะส่งผลให้มีปริมาณ FRM ในโครงข่ายมากขึ้นด้วย ทำให้ปริมาณของ FRM ที่ SW1 และ SW2 ได้รับนั้นมากขึ้น จนมากกว่า BRM และก่อให้เกิดการตอบกลับค่า ER ที่ผิดพลาดขึ้นในครั้งแรก และเหตุการณ์นั้นจะดำเนินเป็นวัฏจักรเช่นนี้ไปเรื่อย ๆ จนกลายเป็นการแกว่งตัวของค่า ACR ดังกล่าว

ขนาดหน่วยความจำที่ SW1 และ SW2 นั้นจะมีค่าเป็น 0 ดังรูปที่ 5.17 เนื่องจากปริมาณทราฟฟิกขาเข้านั้นน้อยกว่าที่สามารถรองรับได้ ส่วนหน่วยความจำที่สวิตช์ 3 นั้นจะมีค่าสูง เนื่องจากต้องรองรับทราฟฟิกสำหรับ VC ทั้ง 2 ชุดในช่วงแรก สังเกตเห็นได้ว่าขนาดของหน่วยความจำนั้นพุ่งสูงขึ้นอย่างรวดเร็ว มีสาเหตุมาจาก ค่าอัตราส่งเริ่มต้นที่แหล่งกำเนิดทั้งสองนั้นมีค่าสูง โครงข่ายจะอยู่ในสภาวะคับคั่งที่แต่เริ่มต้น ทำให้ link3 ไม่สามารถรองรับความคับคั่งได้ และ ส่วนที่รองรับไม่ได้นั้นจะนำมาใส่ไว้ในหน่วยความจำที่ SW3 จนถึงที่เวลาประมาณ 50 ms นั้น จะเห็นว่าอัตราการเพิ่มขึ้นของขนาดหน่วยความจำที่สวิตช์นั้นลดลง เนื่องจากแหล่งกำเนิดทั้งสองได้ปรับอัตราส่งให้ลดลง ทำให้มีข้อมูลส่งน้อยลง แต่อย่างไรก็ตามการเกิด consolidation noise นั้นทำให้ค่า ACR เกิดการแกว่งที่มากกว่าปริมาณทราฟฟิกที่สวิตช์สามารถรองรับได้ ทำให้ขนาดหน่วยความจำนั้นยังคงเพิ่มขึ้นเรื่อย ๆ จนไม่มีที่สิ้นสุด

สังเกตได้ว่าการใช้งาน link1 และ link2 นั้นเกิดการแกว่งขึ้น ทั้งนี้เนื่องจากเกิด consolidation noise ขึ้นนั่นเอง ทำให้ค่าอัตราส่งที่แหล่งกำเนิดนั้นแกว่งตัว ทั้ง ๆ ที่ค่าที่ควรจะเป็นสำหรับการใช้งานสายสัญญาณทั้งสองควรจะคงที่ที่ประมาณ 45% เนื่องจากสายสัญญาณรองรับได้ 149.76 Mbps แต่แหล่งกำเนิดควรสามารถส่งได้ 67.48 Mbps ส่วน link3 นั้นมีค่าใช้งาน 100% ตลอด โดยมีผลจากขนาดหน่วยความจำที่สวิตช์ยังมีอยู่ ทำให้มีข้อมูลส่งเข้าสู่ link3 ตลอดเวลาและ ทำให้การใช้งานเป็น 100 %

สังเกตได้จากรูปที่ 5.18 เมื่อนำ ERICA มาใช้ร่วมกับ TS algorithm แล้วจะเห็นว่า consolidation noise ลดลง โดยสังเกตได้จากการแกว่งตัวของค่า ACR ที่ Source A นั้นน้อยลงอย่างเห็นได้ชัด ในขณะที่ผลตอบชั่วคราวนั้นรวดเร็วเช่นเดียวกับ Robert algorithm ที่เป็นเช่นนี้เนื่องจากวิธีที่สองนั้นคล้ายกับ Robert algorithm แต่ได้เพิ่มความถูกต้องให้กับการรวบรวมข่าวสารการป้อนกลับ โดยให้รับเซลล์ RM ให้ได้อย่างน้อย 1 ตัวก่อนจึงสามารถป้อนกับเซลล์ BRM ได้ อย่างไรก็ตาม consolidation noise ก็ยังคงมีผลต่อขนาดหน่วยความจำที่สวิตช์ ที่ยังคงมีค่าแกว่งตัวขึ้น ๆ ลง โดยมีแนวโน้มที่จะเพิ่มขึ้นเรื่อย ๆ ในขณะที่ การใช้งาน link 1 และ link2 ยังคงแกว่งตัวตามค่า ACR ของ Source A

ผลการจำลองเมื่อใช้ RSS algorithm จะเป็นดังรูปที่ 5.19 ซึ่งกลับจะเห็นว่า ค่า ACR ที่ Source A นั้นมีการแกว่งตัวมากขึ้น แสดงว่าเกิด consolidation noise เพิ่มขึ้น โดยระดับการแกว่งตัวของค่า ACR นั้นจะพอ ๆ กับเมื่อใช้ Robert algorithm ที่เป็นเช่นนี้เนื่องจาก RSS algorithm นี้ใช้หลักการเดียวกับ Robert algorithm เพียงแต่ลดความซับซ้อนในการตอบกลับเซลล์ RM ลง แต่ไม่ได้พัฒนาด้านความถูกต้องของข่าวสาร ทำให้ผลตอบชั่วคราวอยู่ในระดับที่เร็วในขณะที่ขนาดหน่วยความจำที่ SW3 นั้นมีค่าเพิ่มขึ้นเรื่อย ๆ อันเนื่องมาจาก consolidation noise ที่มาก เช่นเดียวกับการใช้งาน link1 และ link2 ที่ยังคงแกว่งตัวตามค่า ACR ของ Source A

สำหรับ Wait-for-all algorithm นั้นสามารถขจัด consolidation noise ได้ สังเกตได้จากรูปที่ 5.20 ซึ่งค่า ACR ของ Source A ที่สถานะอยู่ตัวนั้นไม่เกิดการแกว่งตัว และคงที่ที่ประมาณ 70 Mbps ส่งผลให้ค่า ACR ของ Source B มีค่าคงที่ตามไปด้วย ที่เป็นเช่นนี้เนื่องจาก Wait-for-all algorithm นี้เน้นความถูกต้องโดยรวมเซลล์ RM จากทุกกิ่งให้ครบก่อนจึงตอบกลับเซลล์ RM ทำให้ข้อมูลที่ป้อนกลับนี้ถูกต้องแน่นอน อย่างไรก็ตามผลตอบชั่วคราวของวิธีนี้นั้นจะช้ามาก โดยสังเกตได้จากค่า ACR ที่ลดลงช้ามาก เนื่องจากต้องรอเซลล์ BRM จากกิ่งที่อยู่ไกลที่สุดนั่นคือกิ่งที่ส่งไปยัง d_A ผลตอบชั่วคราวของวิธีการนี้จึงขึ้นอยู่กับ branch ที่ไกลที่สุด ส่งผลให้โครงข่ายอยู่ในสถานะคับคั่งในช่วงเริ่มต้นนานขึ้น ซึ่งแสดงได้จากขนาดหน่วยความจำที่ SW3 ที่พุ่งสูงชันเรื่อย ๆ จนถึงจุดที่ ACR นั้นลดลงมา ขนาดหน่วยจำสูงสุดมีขนาดประมาณ 17000 ซึ่งถือว่ามีความสูงมาก อย่างไรก็ตามค่า ACR ในช่วงสถานะอยู่ตัวนั้นมีค่าคงที่ที่ค่าเหมาะสมโดยไม่เกิด consolidation noise ขึ้น ทำให้ขนาดหน่วยความจำสามารถลดลงมาได้

รูปที่ 5.21 แสดงผลการจำลองเมื่อใช้ ERICA ร่วมกับ Immediate rate calculation algorithm ซึ่งพัฒนามาจาก Wait-for-all algorithm จะเห็นได้ว่าวิธีนี้สามารถขจัด consolidation noise ไปได้เช่นกัน ในขณะที่มีผลตอบชั่วคราวที่เร็วมาก สังเกตได้จากค่า ACR ของ Source A ซึ่งลดลงอย่างรวดเร็วในช่วงแรก และคงที่ที่ประมาณ 70 Mbps ที่สถานะอยู่ตัว ทั้งนี้เนื่องจากวิธีนี้ได้เพิ่ม Overload detect ซึ่งเพิ่มความรวดเร็วในการตรวจจับความคับคั่งให้กับ Wait-for-all algorithm และทำการตอบกลับเซลล์ RM ทันทีที่เกิดสถานะ overload ซึ่งจะเห็นว่าในช่วงแรกนั้นโครงข่ายอยู่ในสถานะติดขัด และ SW3 นั้นจะอยู่ในสถานะ overload คือมีปริมาณทราฟฟิกขาเข้ามากกว่าทราฟฟิกที่รองรับได้ ดังนั้น SW3 จะทำการส่งกลับเซลล์ BRM พิเศษทันที ส่งผลให้มีการป้อนกลับที่รวดเร็ว และ แล่งกำเนิด A สามารถปรับอัตราส่งให้ลดลงอย่างรวดเร็ว เพื่อลดความคับคั่งของโครงข่ายได้ ด้วยเหตุนี้ขนาดหน่วยความจำสูงสุดที่สวิตช์ 3 จึงมีค่าไม่สูงมากนัก ในขณะที่การใช้งานสายสัญญาณ link1 link2 และ link3 นั้น มีการใช้งานได้อย่างเต็มที่เท่าที่เป็นไปได้

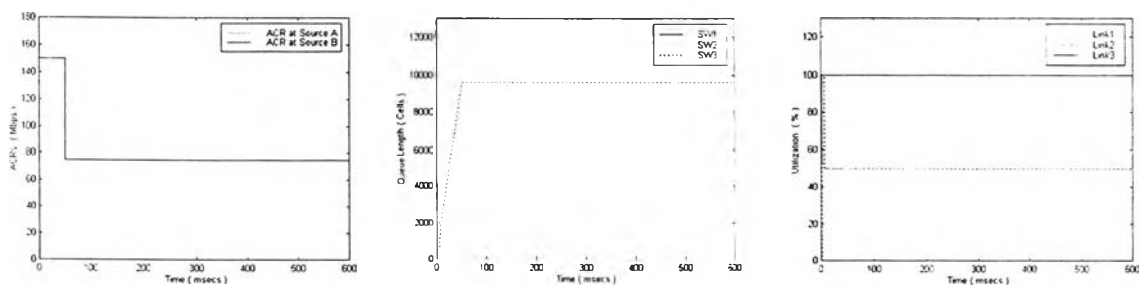
สำหรับวิธีการที่เสนอแบบที่ 1 จะเห็นได้จากรูปที่ 5.22 ว่าวิธีนี้สามารถขจัด consolidation noise ได้ ในขณะที่มีผลตอบชั่วคราวที่เร็วมากเช่นเดียวกัน เนื่องจากวิธีนี้นั้นจัดเก็บค่า MER เป็นแบบ per-VC และ แบบ per-branch ทำให้ข่าวสารความคับคั่งที่เก็บไว้ใน MER มีค่าที่ถูกต้องเสมอ ดังนั้นจึงไม่

เกิดการผิดพลาดของข่าวสารที่ใช้ในการป้อนกลับ และ ไม่เกิด consolidation noise ขึ้น นอกจากนี้ยังใช้เงื่อนไขในการตอบกลับข่าวสารในเซลล์ BRM เหมือนกับ Robert algorithm ทำให้สามารถตอบกลับข่าวสารในเซลล์ BRM ได้รวดเร็วที่สุดด้วย สังเกตได้จากขนาดหน่วยความจำสูงสุดที่ใช้มีค่าน้อยกว่า Immediate rate calculation เล็กน้อย ในขณะที่การใช้งานสัญญาณมีค่าเต็มที่เท่าที่จะเป็นไปได้

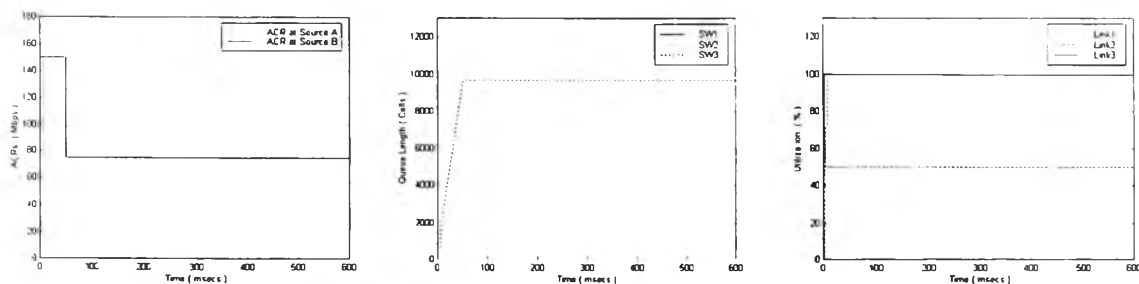
ส่วนวิธีการที่เสนอแบบที่ 2 นั้นให้ผลที่ใกล้เคียงกับวิธีการที่เสนอแบบที่ 1 สังเกตได้จากรูปที่ 5.23 โดยค่า ACR ของ source A นั้นไม่เกิดการแกว่งตัว แสดงว่าไม่เกิด consolidation noise ขึ้น ในขณะที่มีผลตอบชั่วคราวที่รวดเร็ว ทั้งนี้เนื่องจากวิธีตัดแปลงมาจาก RSS algorithm หากแต่ไม่มีการลบค่า MER ที่เก็บไว้ให้เป็นค่า PCR แต่จะตั้งค่า MER ให้เป็นค่าที่เพิ่งได้รับจากเซลล์ RM ตัวนั้น ซึ่งเป็นการลดปัญหาการลบค่า MER ให้เป็นค่าสูง ซึ่งเป็นต้นกำเนิดของการแกว่งตัวของค่า ACR นั่นเอง ส่วนผลตอบชั่วคราวมีผลตอบชั่วคราวที่รวดเร็วพอ ๆ กับ RSS algorithm เช่นกัน สังเกตได้จากค่า ACR ของ source A นั้นลดลงจากค่า ICR มาอยู่ที่ fairshare อย่างรวดเร็ว ในขณะที่ขนาดหน่วยความจำสูงสุดที่ SW3 นั้นมีค่าใกล้เคียงกับ Immediate rate calculation และ วิธีการที่เสนอแบบที่ 2 และมีการใช้งานสายสัญญาณได้อย่างเต็มที่

5.2.3 สมรรถนะของวิธี FMMRA

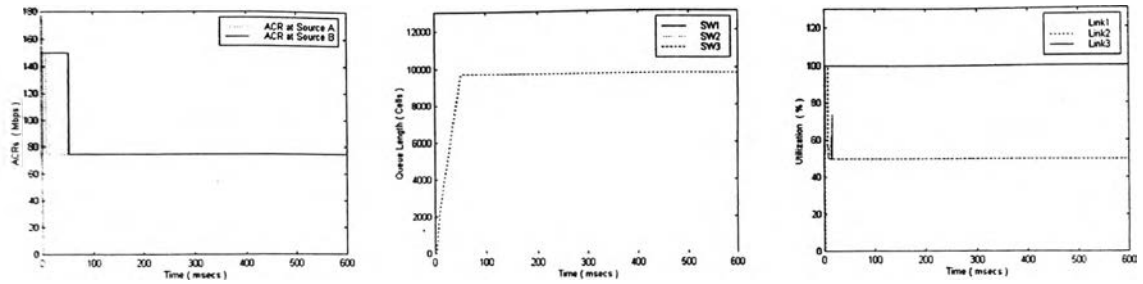
เมื่อใช้ FMMRA ร่วมกับวิธีรวบรวมความคับคั่งต่าง ๆ จะได้ผลดังรูปที่ 5.24 ถึง 5.30



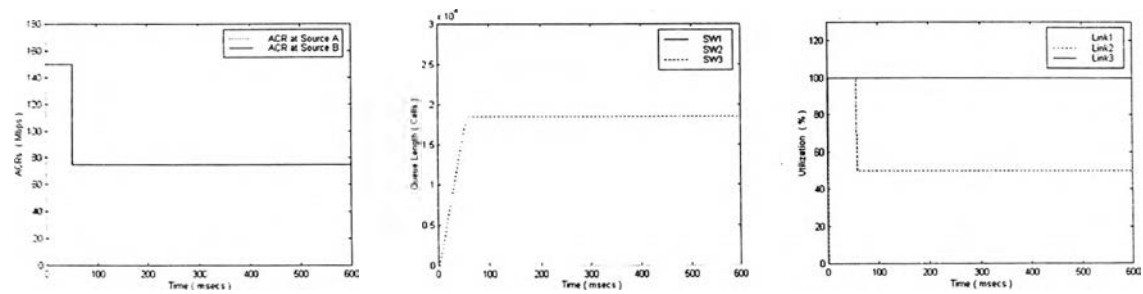
รูปที่ 5.24 ผลการจำลองเมื่อใช้วิธี FMMRA ร่วมกับ Robert Algorithm สำหรับ Chain configuration



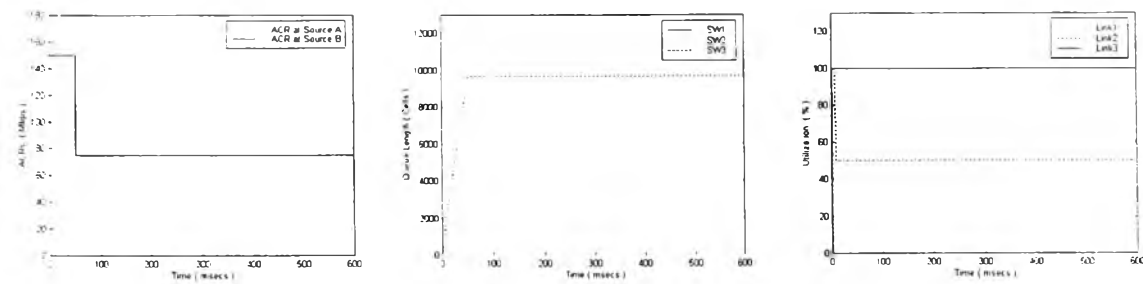
รูปที่ 5.25 ผลการจำลองเมื่อใช้วิธี FMMRA ร่วมกับ TS Algorithm สำหรับ Chain configuration



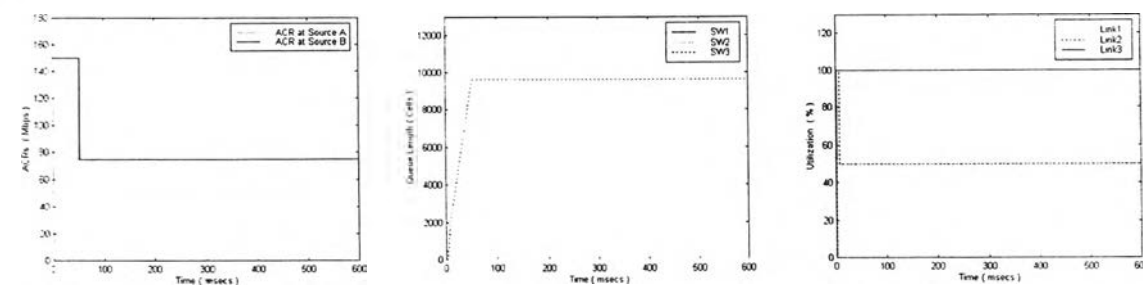
รูปที่ 5.26 ผลการจำลองเมื่อใช้วิธี FMMRA ร่วมกับ RSS Algorithm สำหรับ Chain configuration



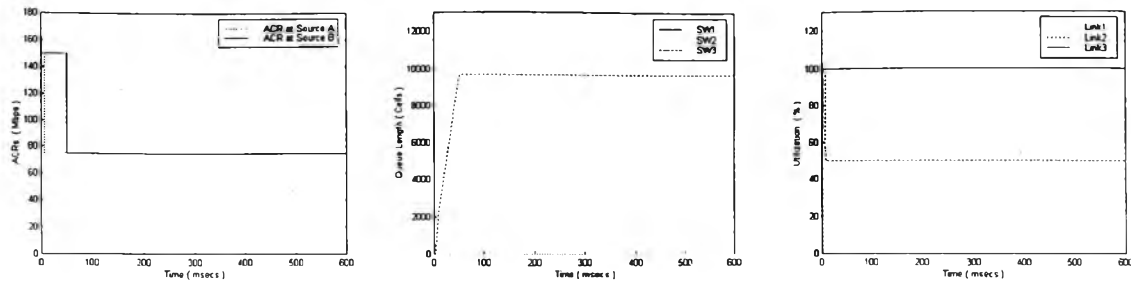
รูปที่ 5.27 ผลการจำลองเมื่อใช้วิธี FMMRA ร่วมกับ Wait-for-all Algorithm สำหรับ Chain configuration



รูปที่ 5.28 ผลการจำลองเมื่อใช้วิธี FMMRA ร่วมกับ Immediate Rate Calculation Algorithm สำหรับ Chain configuration



รูปที่ 5.29 ผลการจำลองเมื่อใช้วิธี FMMRA ร่วมกับวิธีการที่เสนอแบบที่ 1 สำหรับ Chain configuration



รูปที่ 5.30 ผลการจำลองเมื่อใช้วิธี FMMRA ร่วมกับวิธีการที่เสนอแบบที่ 2 สำหรับ Chain configuration

สังเกตได้จากรูปที่ 5.24, 5.25 และ 5.26 จะเห็นว่าเมื่อใช้ Robert algorithm, TS algorithm และ RSS algorithm ตามลำดับนั้นมีผลตอบชั่วครู่ดีมาก ซึ่งสังเกตได้จากค่า ACR ของ source A ที่ลดลงมาอยู่ที่ fairshare อย่างรวดเร็ว ในขณะที่ไม่มี consolidation noise เลย แสดงได้จากค่า ACR ของ source A ที่ไม่เกิดการแกว่งตัวเลย ทั้งนี้เนื่องจาก FMMRA นั้นให้ผลตอบชั่วครู่ที่รวดเร็วมาก และไม่มีการปรับเพิ่มค่าอัตราส่งของ source A หลังจากที่ source B นั้นปรับอัตราส่งลดลงมาที่ fairshare อันเป็นเหตุให้ปริมาณ FRM มากกว่า BRM และเป็นสาเหตุให้เกิดการแกว่งตัวของค่า ACR ขึ้น ดังนั้นกลไก FMMRA สามารถขจัดปัญหา consolidation noise ในรูปแบบโครงข่ายนี้ได้ และช่วยให้โครงข่ายปราศจากความคับคั่ง ซึ่งสังเกตได้จากขนาดหน่วยความจำที่ SW3 ซึ่งมีค่าไม่มากนัก จะเห็นว่าค่า ACR ของ source A ของวิธีรวบรวมความคับคั่งทั้ง 3 นั้นมีผลตอบที่คล้ายคลึงกันมาก ยังผลให้ขนาดหน่วยความจำที่ SW3 และ การใช้งานสายสัญญาณต่างๆ นั้นมีผลตอบคล้าย ๆ กัน

ขนาดหน่วยความจำที่สวิตช์ 3 นั้นจะเพิ่มขึ้นในช่วงแรกเนื่องจากมีสถานะโหลดเกินจากการที่แหล่งกำเนิดทั้งสองส่งข้อมูลที่อัตราส่งมากกว่าที่ link3 สามารถรองรับได้ หลังจากนั้นขนาดหน่วยความจำที่สวิตช์ 3 เริ่มมีขนาดคงที่ เนื่องจาก FMMRA นั้นทำให้โครงข่ายปรับตัวเข้าสู่สถานะอยู่ตัว โดยที่ปรับอัตราส่งให้พอดีกับปริมาณทราฟฟิกที่รองรับได้ ทำให้ปริมาณทราฟฟิกขาเข้าเท่ากับทราฟฟิกขาออกที่สวิตช์ ดังนั้นขนาดหน่วยความจำจึงมีขนาดคงที่ ซึ่งสังเกตได้จากการใช้งานสายสัญญาณของ link1 และ link2 ที่มีปริมาณ 50 % ในขณะที่การใช้งาน link3 เป็น 100%

จากรูปที่ 5.27 จะเห็นว่า Wait-for-all algorithm นั้นให้ผลตอบชั่วครู่ที่ช้าเช่นเดียวกันในกรณีที่ใช้ ERICA เนื่องจากกลไกการทำงานของ Wait-for-all algorithm ทำให้แหล่งกำเนิดต้องรอให้เซลล์ BRM เดินทางกลับมาจากปลายทางที่ไกลที่สุดก่อน จึงสามารถปรับอัตราส่งให้ลดลงได้ ส่งผลให้โครงข่ายอยู่ในสถานะคับคั่งเป็นเวลานาน และ ขนาดหน่วยความจำที่ SW3 มีค่าสูงตามไปด้วย อย่างไรก็ตาม การใช้งานสายสัญญาณยังคงเป็นไปได้อย่างเต็มที่

Immediate rate calculation algorithm นั้นปรับปรุงผลตอบชั่วครู่ของ Wait-for-all algorithm ได้เป็นอย่างดี สังเกตได้จากรูปที่ 5.28 ซึ่งค่า ACR ของ source A นั้นลดลงมาอยู่ที่ fairshare อย่างรวดเร็วใน

ช่วงแรก ทำให้โครงข่ายหลุดจากสถานะคับคั่งอย่างรวดเร็ว สังเกตว่าผลจากจำลองที่ได้ไม่แตกต่าง Robert algorithm มากนัก

เมื่อใช้วิธีการที่เสนอแบบที่ 1 และ วิธีการที่เสนอแบบที่ 2 ร่วมกับ FMMRA จะได้ผลการจำลองดังรูปที่ 5.29 ซึ่งจะสังเกตได้ว่าได้ผลคล้ายคลึงกับ Robert algorithm มาก โดยไม่เกิด consolidation noise และมีผลตอบชั่วคราวที่รวดเร็วมาก

5.2.4 เปรียบเทียบประสิทธิภาพระหว่าง ERICA และ FMMRA

ทั้ง FMMRA และ ERICA นั้นสามารถปรับอัตราส่งที่แหล่งกำเนิดไปสู่อัตราส่งที่โครงข่ายต้องการได้อย่างยุติธรรมที่สุด แต่สำหรับ ERICA นั้นเกิดปัญหา consolidation noise กับวิธี Robert algorithm, TS algorithm และ RSS algorithm ทั้งนี้เนื่องจาก ERICA มีการปรับอัตราส่งเพิ่มขึ้น หรือลดลงเล็กน้อย ทำให้เกิดปัญหาในการรวบรวมข่าวสารความคับคั่ง ดังที่ได้อธิบายไปแล้ว

FMMRA นั้นสามารถลด consolidation noise ได้ เนื่องจาก FMMRA นั้นสามารถปรับอัตราส่งให้กับแหล่งกำเนิดได้อย่างรวดเร็ว และทำให้ระบบอยู่ในสถานะอยู่ตัวอย่างรวดเร็ว

หากวิเคราะห์ด้านความซับซ้อนในการนำมาใช้ในการส่งแบบหนึ่งจุดไปสู่หลายจุดแล้วจะเห็นว่าวิธี FMMRA นั้นมีความซับซ้อนมากกว่าวิธี ERICA เนื่องจากต้องเพิ่มจำนวนตัวแปรสำหรับ β และ λ เข้าไป ทำให้มีความซับซ้อนคำนวณเพิ่มขึ้นด้วย

5.2.5 เปรียบเทียบประสิทธิภาพระหว่างวิธีการรวบรวมข่าวสารความคับคั่งต่าง ๆ

Robert algorithm, TS algorithm และ RSS algorithm นั้นยังคงมีปัญหา consolidation noise เมื่อใช้กับ ERICA มีซึ่งวิธีแก้ไขคือ เพิ่มความถูกต้องของข้อมูลที่เก็บรวบรวมไว้ เช่นการรอข่าวสารให้ได้ครบจากทุก branch ก่อนการป้อนกลับ ดังเช่น Wait-for-all algorithm ซึ่งจะให้มีผลตอบชั่วคราวที่ช้าลงตามไปด้วย Immediate rate calculation จึงเสนอมาเพื่อแก้ปัญหานี้ โดยหลักการทำงานจะมีลักษณะคล้ายคลึงกับ Wait-for-all algorithm เพียงแต่เพิ่มความสามารถในการตรวจจับสถานะ overload ขึ้นมา

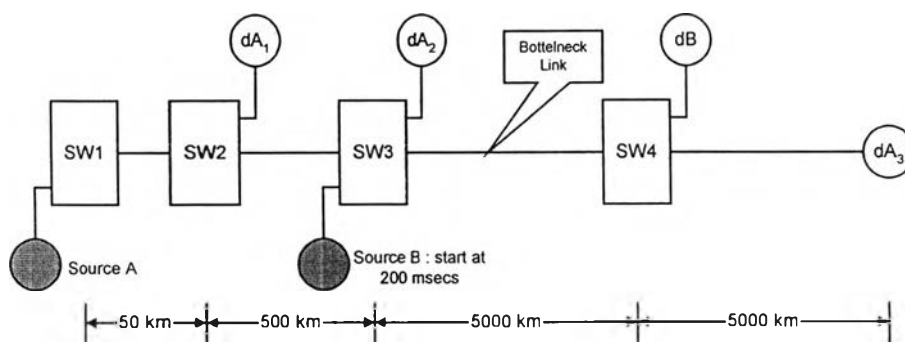
วิธีการอื่นที่ใช้แก้ปัญหาได้แก่ วิธีที่เสนอทั้งสองวิธีนั่นเอง ในวิธีการที่เสนอแบบที่ 1 นั้นมีการแก้ปัญหาโดยการเพิ่มปริมาณตัวแปรที่ใช้เก็บข่าวสารความคับคั่งให้เป็นแบบ per-VC และ per-branch ทำให้ข้อมูลที่จะป้อนกลับนั้นถูกต้องเสมอ อีกทั้งยังมีเงื่อนไขในการป้อนกลับเซลล์ BRM ที่เหมือนกับ Robert algorithm ทำให้มีผลตอบชั่วคราวที่รวดเร็วที่สุด อย่างไรก็ตามการแก้ปัญหาคด้วยวิธีจะเพิ่มความซับซ้อนให้กับสวิทช์ เนื่องจากต้องมีการเพิ่มปริมาณตัวแปรขึ้นมา

ส่วนในวิธีการที่เสนอแบบที่ 2 นั้น มีวิธีแก้ปัญหาที่ง่ายกว่า โดยการไม่ลบล้างข่าวสารความคับคั่งให้มีค่าสูง หากแต่ลบล้างให้กลายเป็นค่าที่เหมาะสมจาก BRM ที่ได้รับนั่นเอง วิธีนี้สามารถลดโอกาสการเกิด consolidation noise ได้อย่างมาก

5.3 ผลการจำลองสำหรับ Chain configuration with low ICR

5.3.1 รูปแบบโครงข่ายและค่ากำหนดต่าง ๆ

รูปแบบโครงข่ายนี้พัฒนามาจาก chain configuration โดยปรับอัตราส่งเริ่มต้นของแหล่งกำเนิด A มีค่าน้อย คือประมาณ 25 Mbps และ แหล่งกำเนิด B ซึ่งเป็น point-to-point connection นั้นเริ่มส่งข้อมูลที 200 msec ทั้งนี้เพื่อทดสอบผลตอบสนองชั่วคราวในสภาวะโหลดต่ำ รูปแบบโครงข่ายจะเป็นดังรูปที่ 5.31



รูปที่ 5.31 Chain configuration with low ICR

ค่าตัวแปรต่าง ๆ ที่ใช้ในรูปแบบโครงข่ายนี้ได้แก่

Source A : PCR = 149.76 Mbps , ICR = 25 Mbps, MCR= 0 Mbps, RIF= 1 . RDF = 0.0125 และ Nrm = 32

Source B : PCR = 149.76 Mbps , ICR = 149.76 Mbps, MCR= 0 Mbps, RIF= 1 . RDF = 0.0125. Nrm = 32 และ แหล่งกำเนิดเริ่มส่งข้อมูลที่ 200 msec

SW1 : Maximum queue length = 30,000

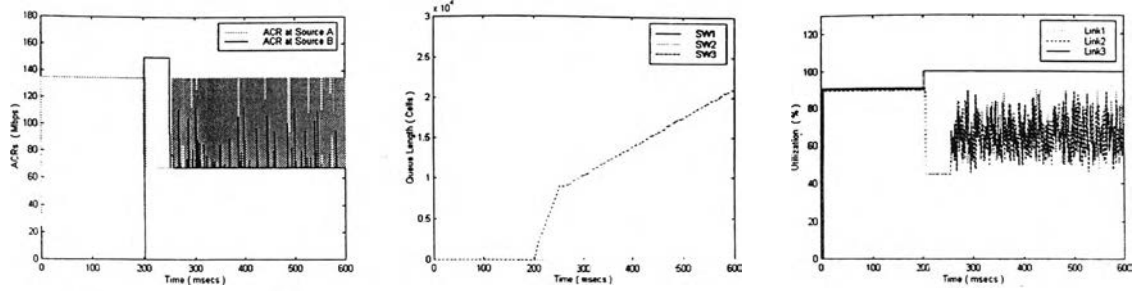
SW2 : Maximum queue length = 30,000

SW3 : Maximum queue length = 30,000

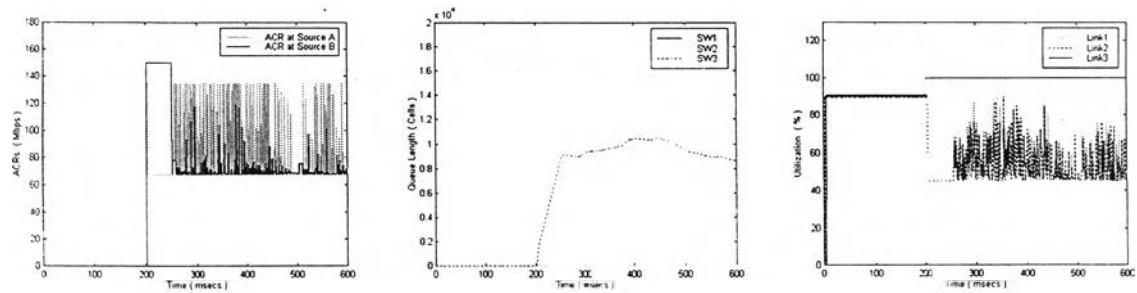
SW4 : Maximum queue length = 30,000

5.3.2 สมรรถนะของวิธี ERICA

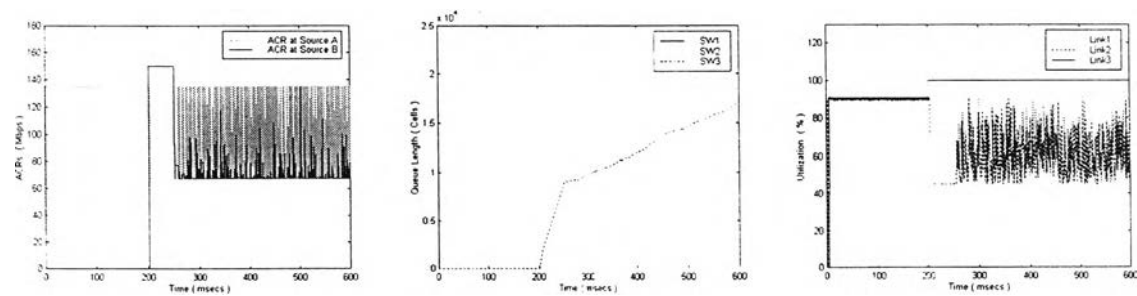
เมื่อใช้ ERICA ร่วมกับวิธีรวบรวมความคับคั่งต่าง ๆ จะได้ผลดังรูปที่ 5.32 ถึง 5.38



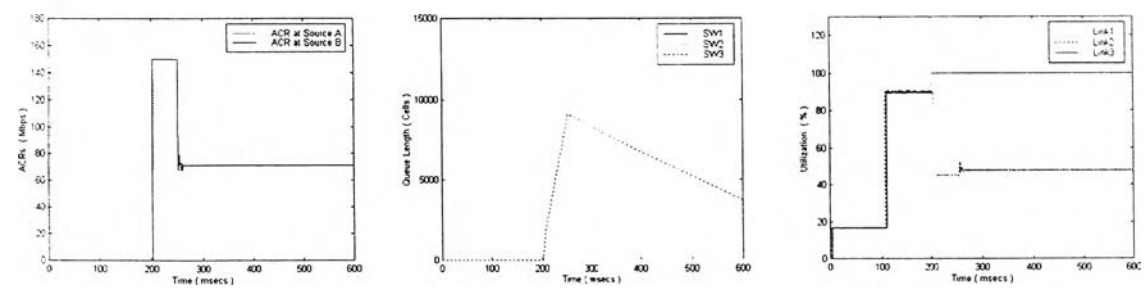
รูปที่ 5.32 ผลการจำลองเมื่อใช้วิธี ERICA ร่วมกับ Robert Algorithm สำหรับ Chain configuration with low ICR



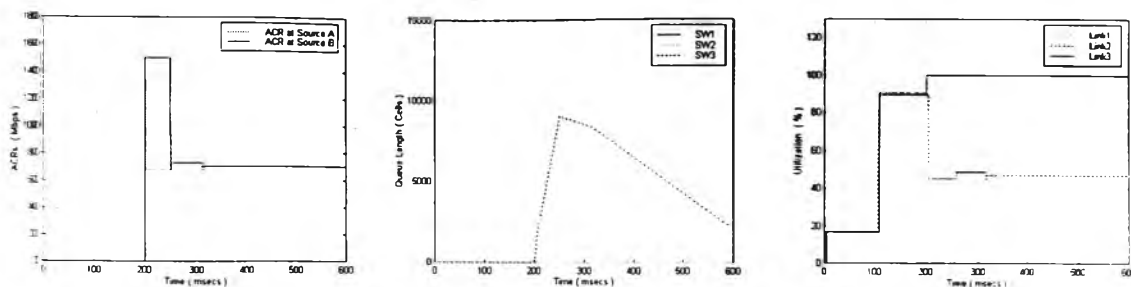
รูปที่ 5.33 ผลการจำลองเมื่อใช้วิธี ERICA ร่วมกับ TS Algorithm สำหรับ Chain configuration with low ICR



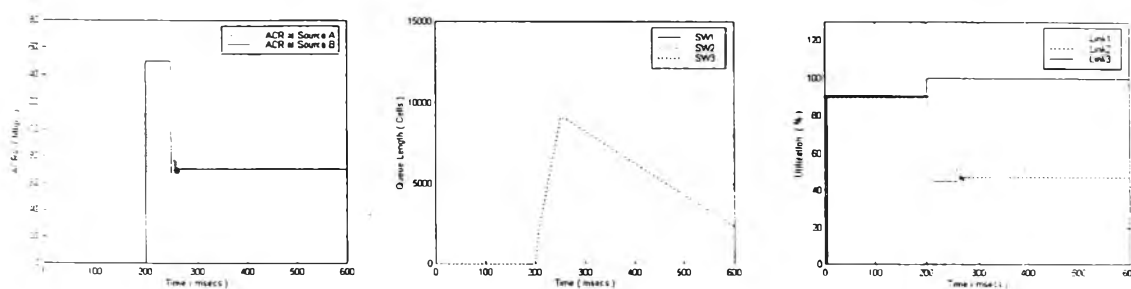
รูปที่ 5.34 ผลการจำลองเมื่อใช้วิธี ERICA ร่วมกับ RSS Algorithm สำหรับ Chain configuration with low ICR



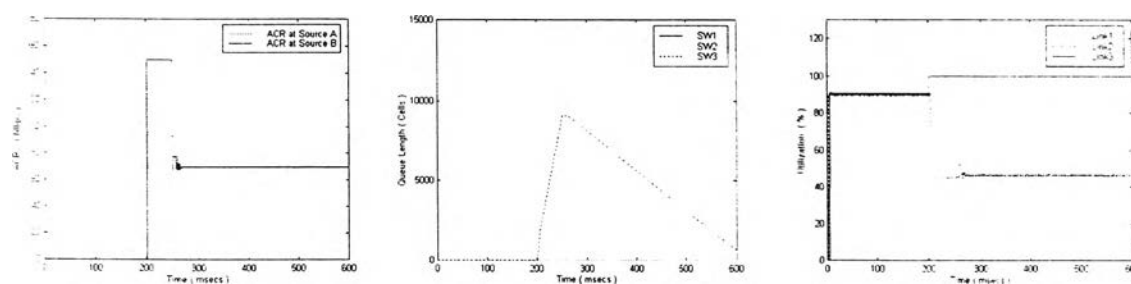
รูปที่ 5.35 ผลการจำลองเมื่อใช้วิธี ERICA ร่วมกับ Wait-for-all Algorithm สำหรับ Chain configuration with low ICR



รูปที่ 5.36 ผลการจำลองเมื่อใช้วิธี ERICA ร่วมกับ Immediate Rate Calculation Algorithm สำหรับ Chain configuration with low ICR



รูปที่ 5.37 ผลการจำลองเมื่อใช้วิธี ERICA ร่วมกับวิธีการที่เสนอแบบที่ 1 สำหรับ Chain configuration with low ICR



รูปที่ 5.38 ผลการจำลองเมื่อใช้วิธี ERICA ร่วมกับวิธีการที่เสนอแบบที่ 2 สำหรับ Chain configuration with low ICR

รูปแบบโครงข่ายนี้ถูกออกแบบมา เพื่อทดสอบสมรรถนะทางด้านผลตอบชั่วคราวของระบบนั่นเอง สังเกตได้จากรูปที่ 5.32 จะเห็นว่าเมื่อนำ Robert algorithm มาใช้งานร่วมกับ ERICA นั้นจะมีผลตอบชั่วคราวที่รวดเร็วมาก สังเกตได้จากค่า ACR ของแหล่งกำเนิด A ที่สามารถปรับตัวจากค่า ICR ที่มีค่าที่ 25 Mbps ไปสู่ค่าที่โครงข่ายสามารถรองรับได้คือที่ประมาณ 135 (150 x target utilization) Mbps นั่นเอง ที่เป็นเช่นนี้เนื่องจากเมื่อ SW1 ได้รับเซลล์ FRM ตัวแรกจากแหล่งกำเนิด A จะทำการตอบกลับเซลล์ BRM ที่บรรจุอัตราส่งที่โครงข่ายสามารถรองรับได้ในขณะนั้น และส่งกลับไปยังแหล่งกำเนิด ซึ่งขั้นตอนนี้จะใช้เวลาเร็วมาก อย่างไรก็ตามเมื่อแหล่งกำเนิด B เริ่มส่งข้อมูล ซึ่งมีผลทำให้แหล่งกำเนิด A ต้องปรับลดอัตราส่งเพื่อให้เหมาะสมกับโครงข่าย และต่อมาจึงเกิด consolidation noise ขึ้นอย่างหนักเช่นเดียวกับ chain configuration

TS algorithm ให้ผลตอบชั่วคราวที่รวดเร็วเช่นกัน เนื่องจาก SW1 มีการส่งเซลล์ BRM กลับไปหาแหล่งกำเนิดทันทีที่ได้รับเซลล์ FRM หลังจากได้รับเซลล์ BRM จาก branch ใด ๆ อย่างน้อย 1 ตัว ดังนั้น SW1 จะรอเซลล์ BRM จากปลายทางที่ใกล้ที่สุดคือปลายทาง dA, ก่อน แล้วจึงรอให้มีเซลล์ FRM มาถึง

จึงสามารถสร้างเซลล์ BRM ส่งกลับไปสู่แหล่งกำเนิดได้ จะเห็นว่าปลายทาง dA_1 นั้นใกล้เคียงกับแหล่งกำเนิดพอสมควร ทำให้ผลตอบชั่วคราวนั้นรวดเร็วมาก อย่างไรก็ตามหลังจากแหล่งกำเนิด B เริ่มส่งข้อมูล แหล่งกำเนิด A นั้นก็เกิด consolidation noise เช่นเดียวกับ Robert's algorithm แต่ระดับของการแหว่งน้อยกว่า Robert's algorithm พอสมควร

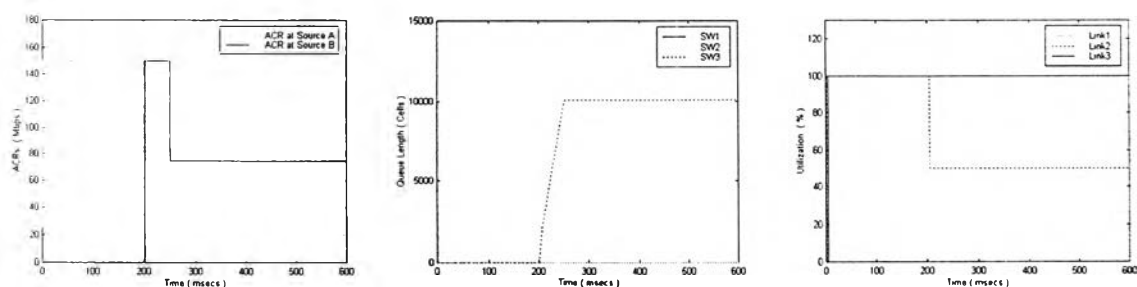
สำหรับ RSS algorithm นั้นสามารถปรับอัตราส่งไปสู่ค่าที่โครงข่ายสามารถรองรับได้อย่างรวดเร็ว ซึ่งจะเห็นว่าในช่วงแรก SW1 นั้นจะตอบกลับเซลล์ BRM เมื่อได้รับเซลล์ BRM จาก dA_1 นั้นเอง ทำให้แหล่งกำเนิดสามารถปรับอัตราส่งได้รวดเร็วเช่นเดียวกับ TS algorithm อย่างไรก็ตาม เมื่อแหล่งกำเนิด B เริ่มส่งข้อมูล ก็ยังคงมี consolidation noise เช่นเดียวกับในกรณี Robert algorithm

เมื่อใช้ Wait-for-all algorithm และ Immediate rate calculation algorithm กับรูปแบบโครงข่ายนี้ จะเห็นว่าได้ผลการจำลองที่เหมือนกัน เนื่องจากในสถานะที่โครงข่ายไม่อยู่ในสถานะไหลคดเกินนั้น Immediate rate calculation algorithm จะทำงานเหมือนกับ Wait-for-all algorithm นั้นเอง สังเกตได้จากรูปที่ 5.35 และ 5.36 ว่าวิธีการทั้งสองนั้นสามารถขจัด consolidation noise ไปได้ แต่อย่างไรก็ตาม ผลตอบชั่วคราวของวิธีการทั้งสองนั้นช้ามาก สังเกตได้จากค่า ACR ที่ไม่สามารถเพิ่มขึ้นจนถึงเวลาประมาณ 100 msec ทั้งนี้เนื่องจากแหล่งกำเนิดต้องรอให้เซลล์ BRM เดินทางกลับมาจากปลายทางที่ไกลที่สุด ซึ่งได้แก่ ปลายทาง dA_3 ส่งผลให้มีการใช้งานสัญญาณต่าง ๆ อย่างไม่คุ้มค่า สังเกตได้จากค่าการใช้งานสายสัญญาณสายสัญญาณ link1, link2 และ link3 ที่มีค่าประมาณ 20 % ในช่วง 0-100 msec นั้นเอง

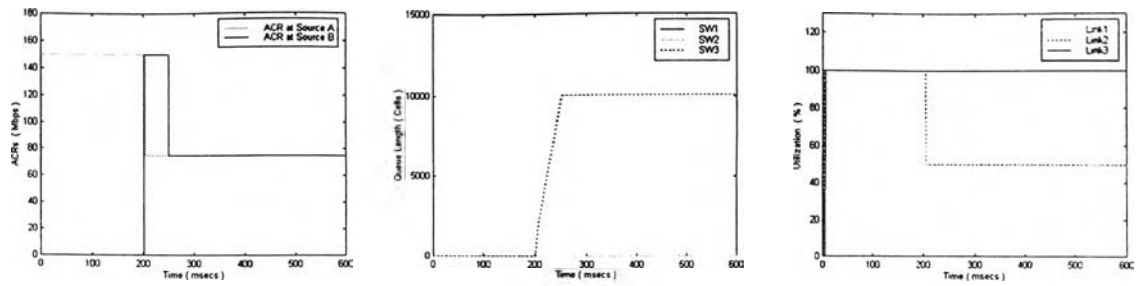
จากรูปที่ 5.37 และ 5.38 ซึ่งแสดงผลการจำลองสำหรับวิธีการที่เสนอแบบที่ 1 และ 2 นั้น จะเห็นว่าให้สมรรถนะที่ใกล้เคียงกัน โดยสามารถขจัด consolidation noise ไปได้ ในขณะที่มีผลตอบชั่วคราวที่รวดเร็ว สำหรับวิธีการที่เสนอแบบที่ 1 นั้นใช้เงื่อนไขในการตอบกลับเซลล์ BRM เหมือนกับ Robert algorithm ทำให้มีผลตอบสนองชั่วคราวที่รวดเร็วมาก ส่วนวิธีการที่เสนอแบบที่ 2 นั้นใช้เงื่อนไขการตอบกลับเซลล์ BRM เหมือนกับ RSS algorithm แต่เพิ่มการป้องกันไม่ให้เกิด consolidation noise ขึ้น ทำให้ได้สมรรถนะที่ดีมากทั้งสองวิธี

5.3.3 สมรรถนะของวิธี FMMRA

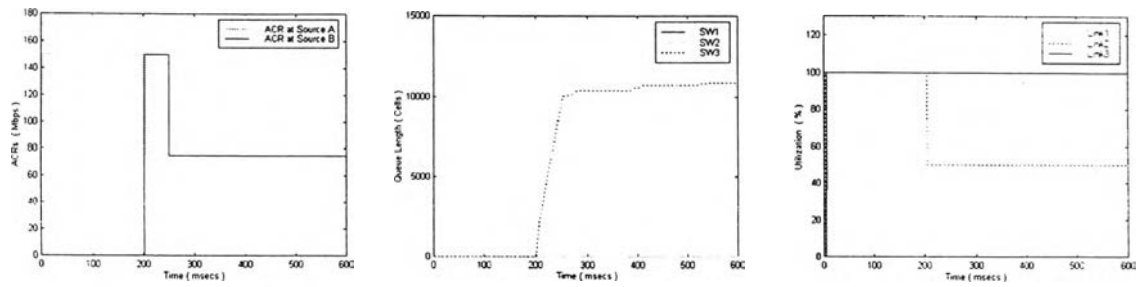
เมื่อใช้ FMMRA ร่วมกับวิธีรวบรวมความคับคั่งต่าง ๆ จะได้ผลดังรูปที่ 5.39 ถึง 5.45



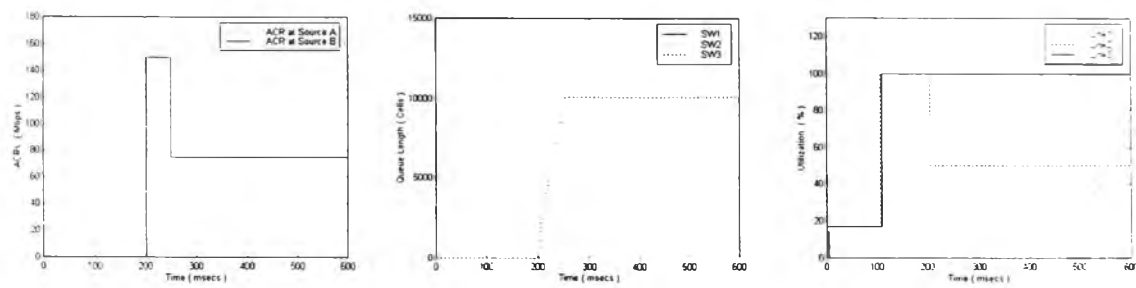
รูปที่ 5.39 ผลการจำลองเมื่อใช้วิธี FMMRA ร่วมกับ Robert Algorithm สำหรับ Chain configuration with low ICR



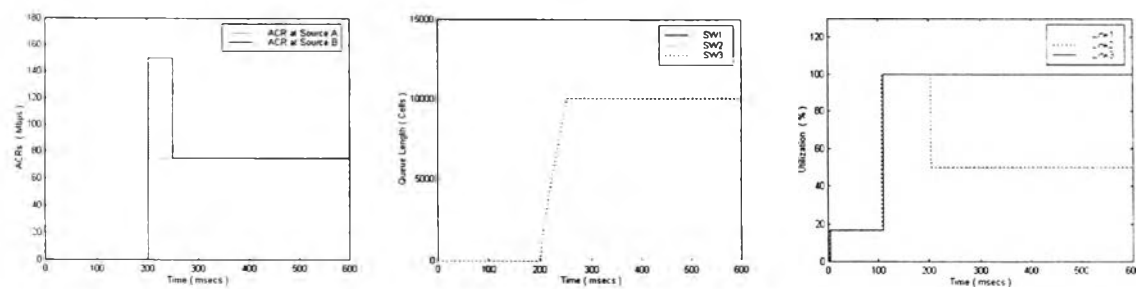
รูปที่ 5.40 ผลการจำลองเมื่อใช้วิธี FMMRA ร่วมกับ TS Algorithm สำหรับ Chain configuration with low ICR



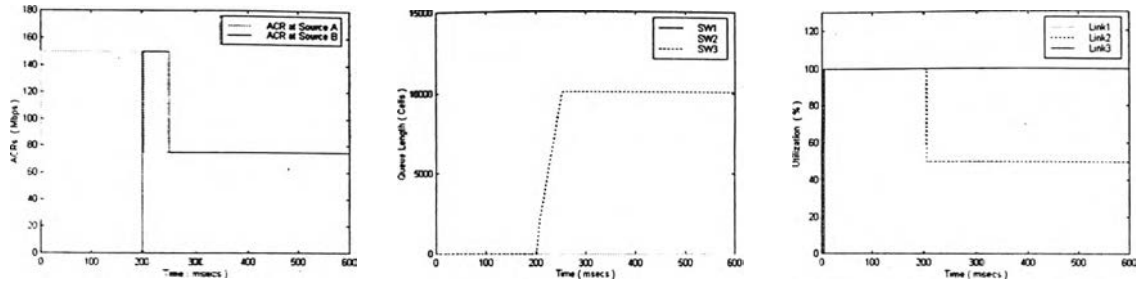
รูปที่ 5.41 ผลการจำลองเมื่อใช้วิธี FMMRA ร่วมกับ RSS Algorithm สำหรับ Chain configuration with low ICR



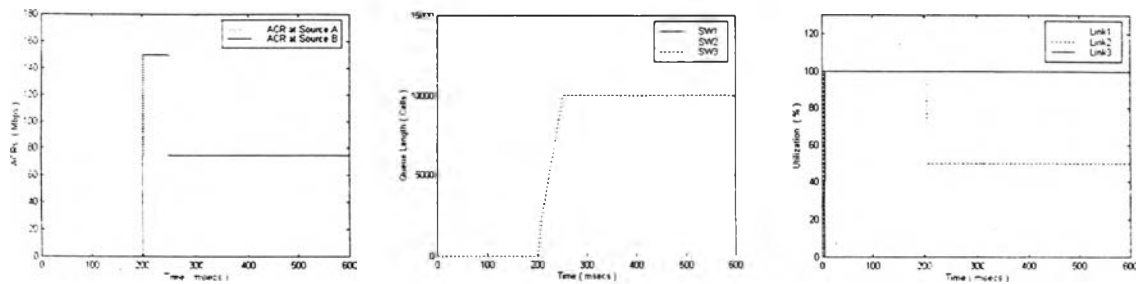
รูปที่ 5.42 ผลการจำลองเมื่อใช้วิธี FMMRA ร่วมกับ Wait-for-all Algorithm สำหรับ Chain configuration with low ICR



รูปที่ 5.43 ผลการจำลองเมื่อใช้วิธี FMMRA ร่วมกับ Immediate Rate Calculation Algorithm สำหรับ Chain configuration with low ICR



รูปที่ 5.44 ผลการจำลองเมื่อใช้วิธี FMMRA ร่วมกับวิธีการที่เสนอแบบที่ 1 สำหรับ Chain configuration with low ICR



รูปที่ 5.45 ผลการจำลองเมื่อใช้วิธี FMMRA ร่วมกับวิธีการที่เสนอแบบที่ 2 สำหรับ Chain configuration with low ICR

สังเกตได้จากรูปที่ 5.39 - 5.41 ได้ว่าเมื่อใช้ Robert algorithm, TS algorithm และ RSS algorithm ร่วมกับ FMMRA แล้วปรากฏว่าให้สมรรถนะที่ไม่แตกต่างกัน โดยที่มีผลตอบสนองชั่วคราวที่รวดเร็วในขณะที่สามารถขจัด consolidation noise ได้ ทั้งนี้เนื่องจาก FMMRA นั้นสามารถขจัด consolidation noise ไปได้ ในขณะที่ผลตอบสนองชั่วคราวของทั้งสามวิธีนั้นมีความรวดเร็วมาก สังเกตได้จากค่า ACR ของแหล่งกำเนิด A ที่ปรับเพิ่มขึ้นได้อย่างรวดเร็ว

เมื่อใช้ Wait-for-All algorithm และ Immediate rate calculation algorithm ร่วมกับ วิธี FMMRA จะให้ผลตอบที่คล้ายคลึงกัน เนื่องจากโครงข่ายอยู่ในสถานะโหลดต่ำ ทำให้วิธี Immediate rate calculation algorithm จะทำงานเหมือนกับ Wait-for-all algorithm จะเห็นว่าวิธีการทั้งสองนั้นมีผลตอบสนองชั่วคราวที่ช้ามาก ค่า ACR ของแหล่งกำเนิด A นั้นมีค่าคงที่ที่ ICR (25 Mbps) โดยไม่สามารถปรับตัวเข้าสู่อัตราส่งที่โครงข่ายสามารถรองรับได้คือที่ค่า 150 Mbps ทั้งนี้เนื่องจากวิธีการทั้งสองทำให้ แหล่งกำเนิดต้องรอให้เซลล์ BRM นั้นเดินทางกลับจากปลายทางที่ไกลที่สุดก่อนจึงสามารถปรับอัตราส่งให้เหมาะสมได้ อย่างไรก็ตาม การรอให้เซลล์ BRM เดินทางมาจากทุก branch ก่อนการป้อนกลับเซลล์ BRM นั้นสามารถขจัด consolidation noise ไปได้ โดยค่า ACR ที่แหล่งกำเนิด A นั้นไม่เกิดการแกว่งตัวขึ้น

จากรูปที่ 5.44 และ 5.45 สังเกตได้ว่าวิธีการที่เสนองานทั้งสองวิธีมีผลลัพธ์ที่คล้ายคลึงกัน และคล้ายคลึงกับเมื่อใช้ Robert algorithm โดยที่ไม่มี consolidation noise เลย ในขณะที่มีผลตอบสนองชั่วคราวที่รวดเร็วมาก

5.3.4 เปรียบเทียบประสิทธิภาพระหว่าง ERICA และ FMMRA

ทั้ง FMMRA และ ERICA นั้นสามารถปรับอัตราส่งที่แหล่งกำเนิดไปสู่อัตราส่งที่โครงข่ายต้องการได้อย่างยุติธรรมที่สุด แต่สำหรับ ERICA นั้นเกิดปัญหา consolidation noise กับวิธี Robert algorithm, TS algorithm และ RSS algorithm สำหรับวิธีควบคุมความคับคั่งอื่น ๆ นั้นไม่เกิด consolidation noise ได้หากแต่วิธี Wait-for-all algorithm และ Immediate rate calculation algorithm นั้นมีผลตอบสนองชั่วคราวที่ช้ามาก

เมื่อใช้ FMMRA นั้นไม่มีปัญหาเรื่อง consolidation noise เลย เนื่องจาก FMMRA นั้นสามารถปรับอัตราส่งให้กับแหล่งกำเนิดได้อย่างรวดเร็ว และทำให้ระบบอยู่ในสภาวะอยู่ตัวอย่างรวดเร็ว

หากวิเคราะห์ด้านความซับซ้อนในการนำมาใช้ในการส่งแบบหนึ่งจุดไปสู่หลายจุดแล้วจะเห็นว่าวิธี FMMRA นั้นมีความซับซ้อนมากกว่าวิธี ERICA เนื่องจากต้องเพิ่มจำนวนตัวแปรสำหรับ β และ λ เข้าไป ทำให้มีความซับซ้อนคำนวณเพิ่มขึ้นด้วย

5.3.5 เปรียบเทียบประสิทธิภาพระหว่างวิธีรวบรวมข่าวสารความคับคั่งต่าง ๆ

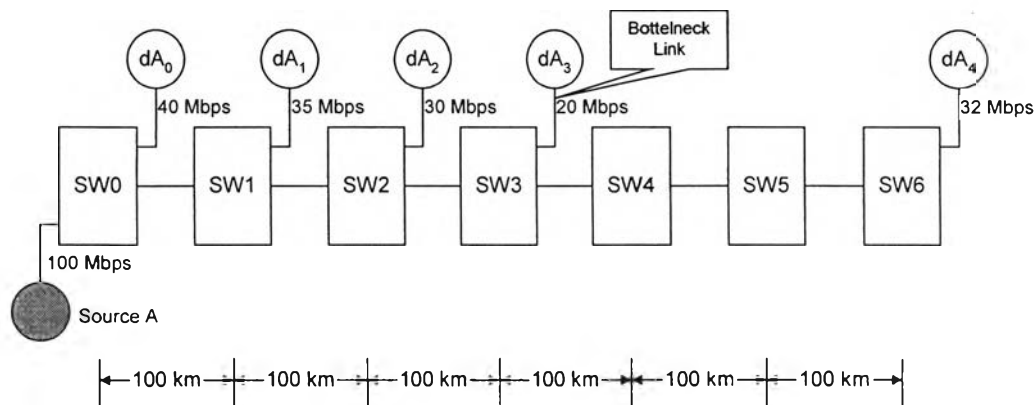
Robert algorithm, TS algorithm และ RSS algorithm นั้นยังคงมีปัญหา consolidation noise เมื่อใช้กับ ERICA ในขณะที่วิธี Wait-for-all algorithm, Immediate rate calculation algorithm, วิธีการที่เสนอแบบที่ 1 และ วิธีการที่เสนอแบบที่ 2 นั้นสามารถขจัด Consolidation noise ไปได้

ส่วนสมรรถนะทางด้านผลตอบสนองชั่วคราวนั้นสามารถแยกเป็นสองประเด็นได้แก่ ในสภาวะที่โครงข่ายอยู่ในภาวะโหลดเกินนั้น วิธี Immediate rate calculation algorithm สามารถปรับอัตราส่งของแหล่งกำเนิดได้อย่างรวดเร็ว แต่เมื่อโครงข่ายอยู่ในสภาวะโหลดต่ำนั้น วิธีนี้จะมีผลตอบสนองที่ช้ามาก อย่างไรก็ตาม วิธี Robert algorithm, TS algorithm, RSS algorithm, วิธีการที่เสนอแบบที่ 1 และ วิธีการที่เสนอแบบที่ 2 นั้นมีผลตอบสนองชั่วคราวที่เร็วมากทั้งในสภาวะโหลดเกินและโหลดต่ำ ส่วน Wait-for-all algorithm นั้นมีผลตอบสนองชั่วคราวที่ช้าในทุกสภาวะ

5.4 ผลการจำลองสำหรับ Jiang configuration

5.4.1 รูปแบบโครงข่ายและค่ากำหนดต่าง ๆ

Jiang ได้เสนอรูปแบบโครงข่ายขึ้นมาเพื่อทดสอบประสิทธิภาพระหว่างวิธีการรวบรวมความคับคั่งต่าง ๆ นั้นในการปรับค่าอัตราส่งของแหล่งกำเนิดที่สามารถปรับตัวตามค่าที่โครงข่ายสามารถรองรับได้หรือไม่ โดยโครงข่ายประกอบด้วยแหล่งกำเนิด A ที่ส่งข้อมูลให้กับปลายทาง dA_0 , dA_1 , dA_2 , dA_3 และ dA_4 ตามลำดับ และ สายสัญญาณที่ส่งถึงปลายทางทั้ง 5 นั้นมีความจุ 40, 35, 30, 20 และ 32 Mbps ตามลำดับ นอกจากนี้สำหรับวิธี ERICA ที่ใช้ในโครงข่ายนี้จะตั้งค่า target utilization เป็น 1 เพื่อให้แหล่งกำเนิดสามารถใช้งานแบนวิธที่มีอยู่ของโครงข่ายได้อย่างเต็มที่ รูปแบบโครงข่ายจะเป็นดังรูปที่ 5.46



รูปที่ 5.46 Jiang configuration

ค่าตัวแปรต่าง ๆ ที่ใช้ในรูปแบบโครงข่ายนี้ได้แก่

Source A : PCR = 5 Mbps , ICR = 149.76 Mbps, MCR = 0 Mbps, RIF = 1 , RDF = 0.0125 และ $N_{rm} = 32$

SW0 : Maximum queue length = 1,000

SW1 : Maximum queue length = 1,000

SW2 : Maximum queue length = 1,000

SW3 : Maximum queue length = 1,000

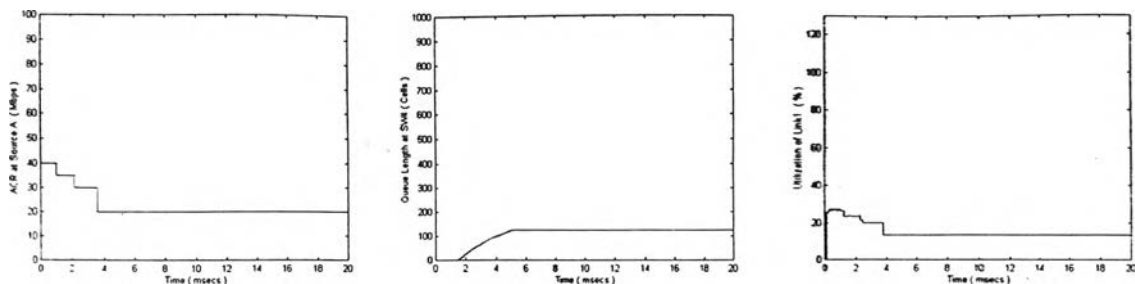
SW4 : Maximum queue length = 1,000

SW5 : Maximum queue length = 1,000

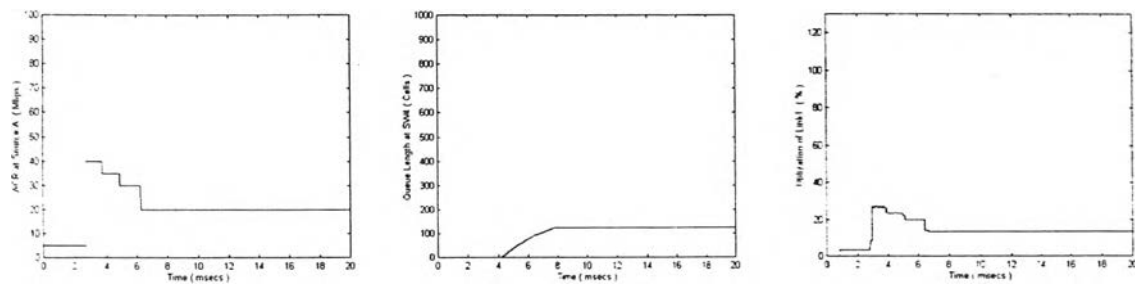
SW6 : Maximum queue length = 1,000

5.4.2 สมรรถนะของวิธี ERICA

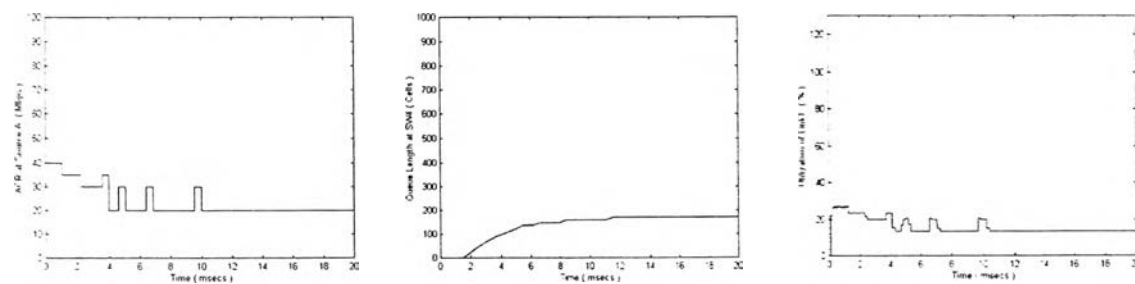
เมื่อใช้ ERICA ร่วมกับวิธีการรวบรวมความคับคั่งต่าง ๆ จะได้ผลดังรูปที่ 5.47 ถึง 5.53



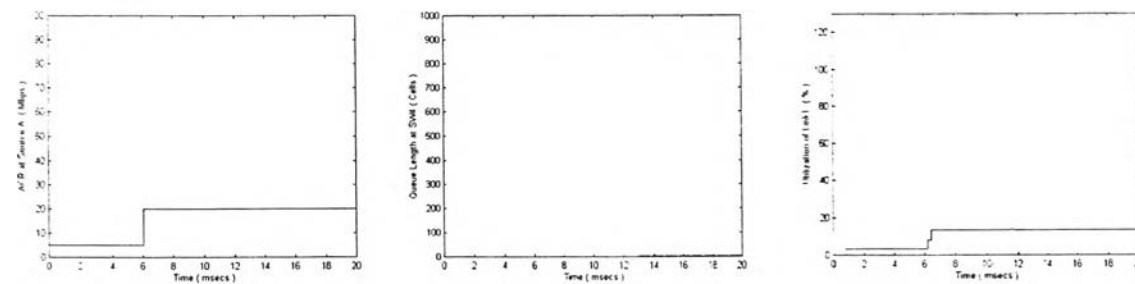
รูปที่ 5.47 ผลการจำลองเมื่อใช้วิธี ERICA ร่วมกับ Robert Algorithm สำหรับ Jiang configuration



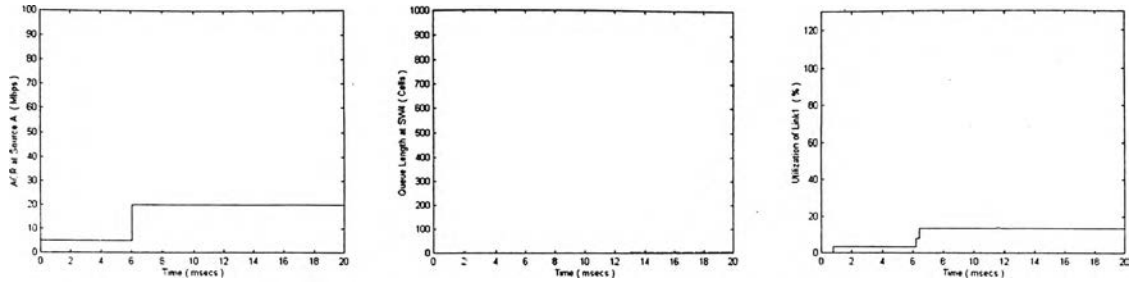
รูปที่ 5.48 ผลการจำลองเมื่อใช้วิธี ERICA ร่วมกับ TS Algorithm สำหรับ Jiang configuration



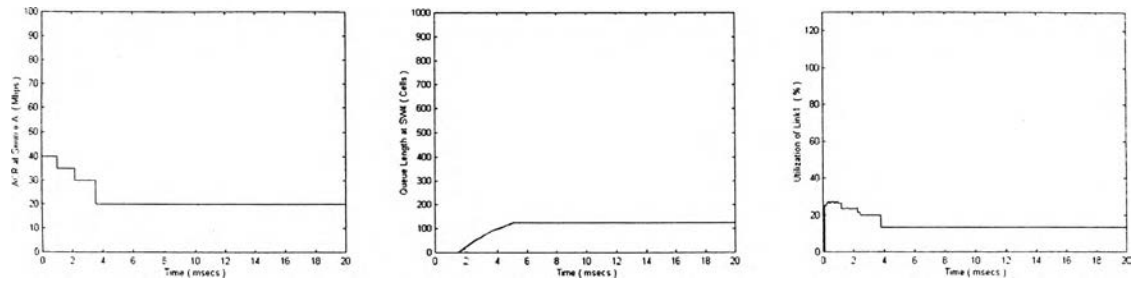
รูปที่ 5.49 ผลการจำลองเมื่อใช้วิธี ERICA ร่วมกับ RSS Algorithm สำหรับ Jiang configuration



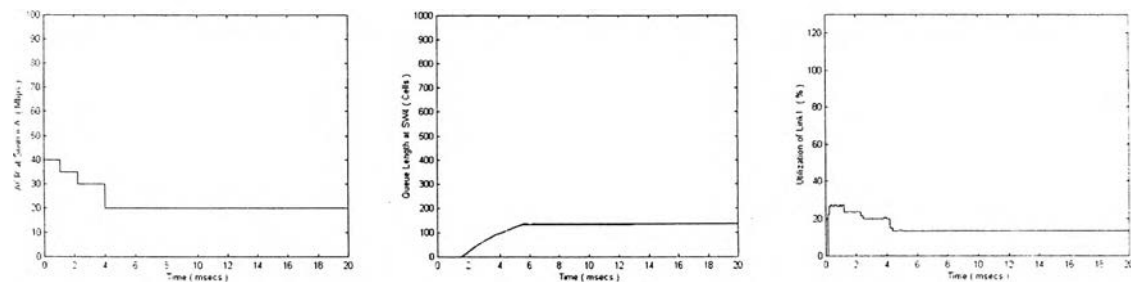
รูปที่ 5.50 ผลการจำลองเมื่อใช้วิธี ERICA ร่วมกับ Wait-for-all Algorithm สำหรับ Jiang configuration



รูปที่ 5.51 ผลการจำลองเมื่อใช้วิธี ERICA ร่วมกับ Immediate Rate Calculation Algorithm สำหรับ Jiang configuration



รูปที่ 5.52 ผลการจำลองเมื่อใช้วิธี ERICA ร่วมกับวิธีการที่เสนอแบบที่ 1 สำหรับ Jiang configuration



รูปที่ 5.53 ผลการจำลองเมื่อใช้วิธี ERICA ร่วมกับวิธีการที่เสนอแบบที่ 2 สำหรับ Jiang configuration

ในการจำลองการทำงานของโครงข่ายนี้ใช้ target utilization เป็น 100% ทำให้สวิตช์สามารถแบ่งแบนวิธของสายสัญญาณต่าง ๆ ให้กับแหล่งกำเนิดต่าง ๆ ได้อย่างเต็มประสิทธิภาพ สายสัญญาณที่มีจะเกิดความคับคั่งนั้นได้แก่ สายสัญญาณที่อยู่ระหว่าง SW3 และ ปลายทาง dA, โดยค่าอัตราส่งที่แหล่งกำเนิดควรจะส่งนั้นอยู่ที่ 20 Mbps

เมื่อพิจารณาอัตราส่งที่แหล่งกำเนิด A ตามรูปที่ 5.47 จะสังเกตได้ว่า Robert algorithm นั้นสามารถปรับอัตราส่งให้สามารถเพิ่มขึ้นเป็น 40 Mbps ได้อย่างรวดเร็ว จากนั้นที่เวลาประมาณ 1, 2 และ 3 ms จึงลดลงมาที่ 35, 30 และ 20 ตามลำดับ ทั้งนี้เนื่องจากแหล่งกำเนิดนั้นได้รับเซลล์ BRM ที่ตอบกลับมาจาก SW0, SW1, SW2 และ SW3 ตามลำดับ จึงสามารถปรับอัตราส่งให้เหมาะสมกับค่าที่สวิตช์ต่าง ๆ ดังกล่าวสามารถรองรับได้ สังเกตได้ว่าเวลาที่แหล่งกำเนิดใช้ปรับตัวตามอัตราส่งที่สวิตช์ต่าง ๆ สามารถรองรับได้นั้นรวดเร็วมาก แสดงให้เห็นถึงผลตอบชั่วคราวที่รวดเร็วของ Robert algorithm อย่างไรก็ตามการเพิ่มอัตราส่งในช่วงแรกนั้นส่งผลให้หน่วยความจำที่ SW4 นั้นมีค่าเพิ่มขึ้นในช่วงแรกจนถึงประมาณ 120 เซลล์ และมีค่าคงที่เนื่องจากค่า target utilization เป็น 100% ทำให้สวิตช์สามารถใช้งาน

สายสัญญาณได้อย่างเต็มที่ และ ปริมาณทราฟฟิกขาออกมีปริมาณเท่า ๆ กับปริมาณทราฟฟิกขาเข้า ดังนั้นจำนวนเซลล์ในหน่วยความจำจึงมีค่าคงที่

TS algorithm ในรูปที่ 5.48 นี้ มีผลตอบสนองชั่วคราวที่ช้ากว่า Robert algorithm อยู่ประมาณ 1 ms สังเกตได้จากค่าอัตราส่งของแหล่งกำเนิด A ที่เริ่มปรับตัวจากค่า ICR คือ 10 Mbps ไปสู่ 40 Mbps ที่เวลา 1 msec จากนั้นจึงปรับตัวลดลงมาที่ 35, 30 และ 20 Mbps ตามลำดับเช่นเดียวกับในกรณี Robert algorithm สาเหตุที่วิธีการนี้มีผลตอบสนองชั่วคราวที่ช้ากว่า Robert algorithm นั้น เนื่องจาก TS algorithm มีเงื่อนไขในการตอบกลับเซลล์ BRM โดยพิจารณาจากการได้รับเซลล์ FRM ในขณะที่เคยได้รับเซลล์ BRM มาแล้วอย่างน้อยหนึ่งตัวหลังจากตอบกลับเซลล์ BRM ไปแล้ว ทำให้เมื่อสวิตช์ได้รับเซลล์ FRM ตัวแรกนั้นจะยังไม่ตอบกลับเซลล์ BRM ในทันที แต่จะรอให้ได้รับเซลล์ BRM จากปลายทาง และได้รับเซลล์ FRM ตัวที่สองก่อนซึ่งช่วงเวลานี้ใช้เวลานานมากเพราะค่าอัตราส่งที่แหล่งกำเนิดมีค่าน้อย ทำให้ปริมาณเซลล์ที่สร้างขึ้นมามีน้อยส่งผลให้ปริมาณเซลล์ FRM มีน้อยตามไปด้วย แหล่งกำเนิดต้องใช้เวลา นานกว่าสามารถให้กำเนิดเซลล์ FRM ตัวที่สองได้ ทำให้การป้อนกลับเซลล์ BRM ของสวิตช์นั้นช้าตามไปด้วย

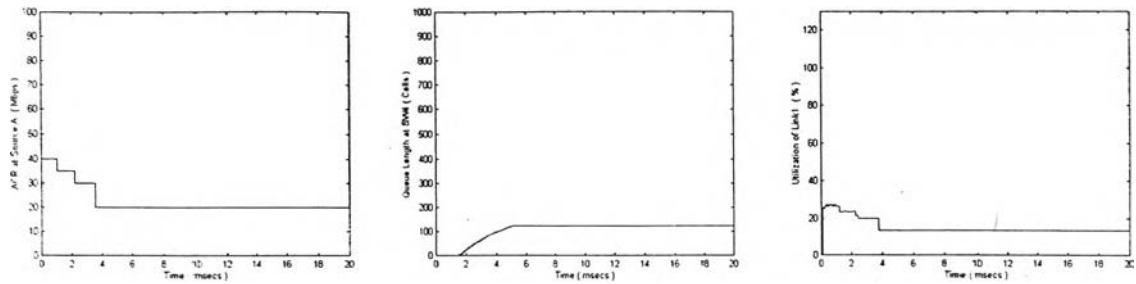
ผลการจำลองสำหรับ RSS algorithm แสดงอยู่ในรูปที่ 5.49 ซึ่งจะสังเกตเห็นว่าวิธีการนี้มีผลตอบสนองชั่วคราวที่รวดเร็วกว่า เนื่องจากวิธีการนี้รอเพียงให้ได้รับเซลล์ BRM หลังเพิ่งจากได้รับเซลล์ FRM จึงสามารถป้อนกลับเซลล์ BRM สู่อุปกรณ์ได้ ทำให้สวิตช์ต่าง ๆ นั้นสามารถป้อนกลับเซลล์ BRM ได้อย่างรวดเร็ว

Wait-for-all algorithm และ Immediate rate calculation นั้นมีผลการจำลองที่เหมือนกัน สังเกตได้จากรูปที่ 5.50 และ 5.51 โดยวิธีการทั้งสองนั้นให้ผลตอบสนองชั่วคราวที่ช้ามาก สังเกตได้จากค่าอัตราส่งของแหล่งกำเนิด A ที่ไม่สามารถปรับเพิ่มขึ้นจากค่า ICR ได้จนกระทั่งที่เวลาประมาณ 6 ms ทั้งนี้เนื่องจากในวิธีการ Wait-for-all algorithm นั้นแหล่งกำเนิดต้องรอให้เซลล์ BRM นั้นเดินทางกลับมาจากปลายทางที่ไกลที่สุดก่อนนั่นคือปลายทาง d_A นั่นเอง ซึ่งเวลาที่แหล่งกำเนิดต้องรอนั้นสอดคล้องกับระยะทางที่เซลล์ BRM ใช้ในการเดินทางไปและกลับจากปลายทาง d_A คือ 1,200 กิโลเมตร นั่นเอง สาเหตุที่ Immediate rate calculation มีผลการจำลองที่เหมือนกับ Wait-for-all algorithm นั้น เนื่องจากโครงข่ายไม่อยู่ในสภาวะโหลดเกิน ทำให้ overload detect technique นั้นไม่ทำงาน ดังนั้นวิธีดังกล่าวทำงานเหมือน Wait-for-all algorithm นั่นเอง

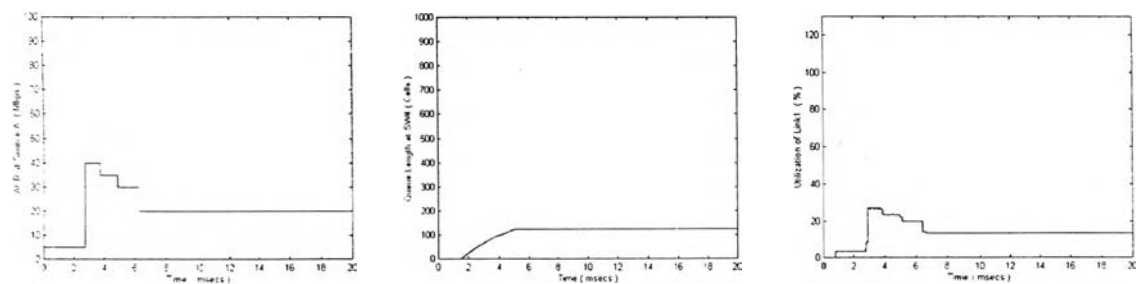
ผลการจำลองของวิธีการที่เสนอแบบที่ 1 และ วิธีการที่เสนอแบบที่ 2 มีลักษณะที่เหมือนกัน และเหมือนกัน Robert algorithm และ RSS algorithm เนื่องจากวิธีการที่เสนอแบบที่ 1 นั้นใช้เงื่อนไขในการตอบกลับเซลล์ BRM เหมือน Robert algorithm และ วิธีการที่เสนอแบบที่ 2 นั้นใช้เงื่อนไขในการตอบกลับเซลล์ BRM เหมือน RSS algorithm โดยสังเกตได้ว่าทั้งสองวิธีนี้มีผลตอบสนองชั่วคราวที่รวดเร็ว โดยสามารถปรับอัตราส่งให้เหมาะสมกับค่าที่โครงข่ายสามารถรองรับได้เป็นอย่างดี

5.4.3 สมรรถนะของวิธี FMMRA

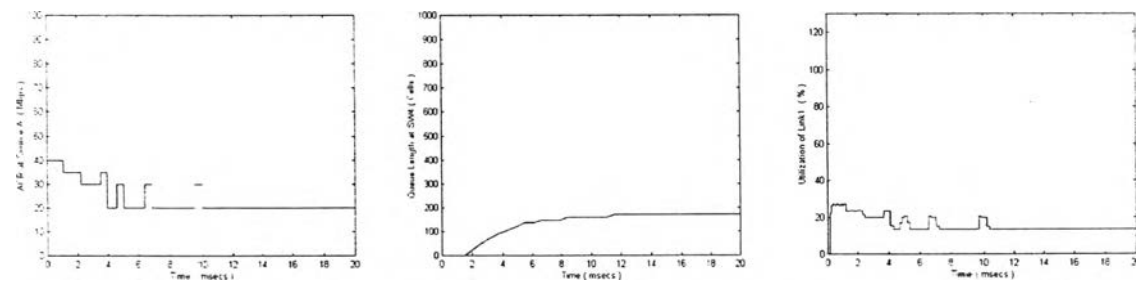
เมื่อใช้ FMMRA ร่วมกับวิธีรวบรวมความคับคั่งต่าง ๆ จะได้ผลดังรูปที่ 5.54 ถึง 5.60



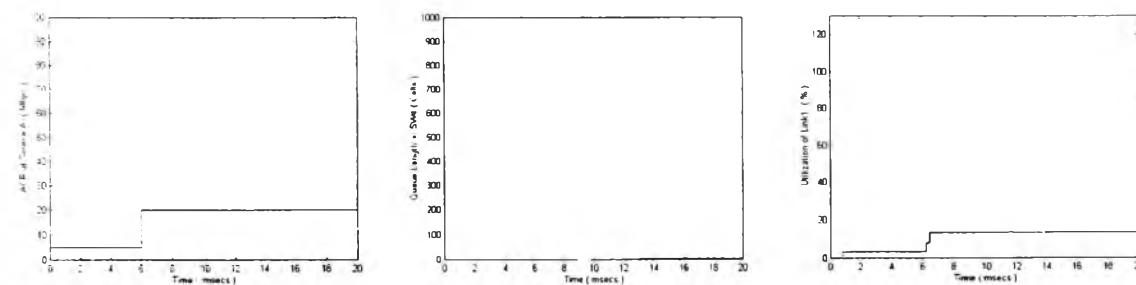
รูปที่ 5.54 ผลการจำลองเมื่อใช้วิธี FMMRA ร่วมกับ Robert Algorithm สำหรับ Jiang configuration



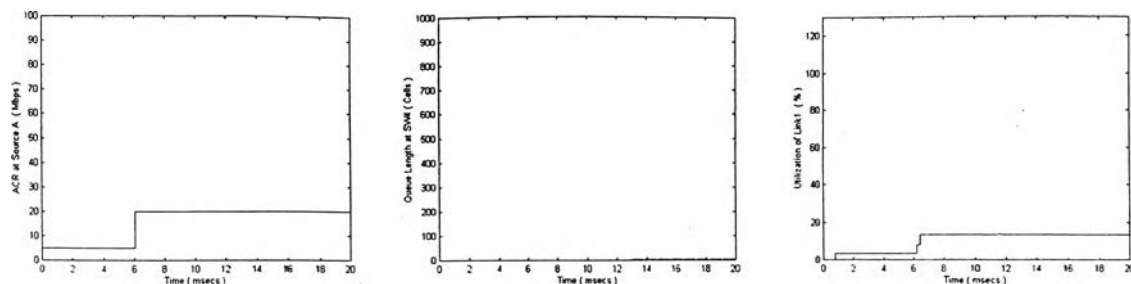
รูปที่ 5.55 ผลการจำลองเมื่อใช้วิธี FMMRA ร่วมกับ TS Algorithm สำหรับ Jiang configuration



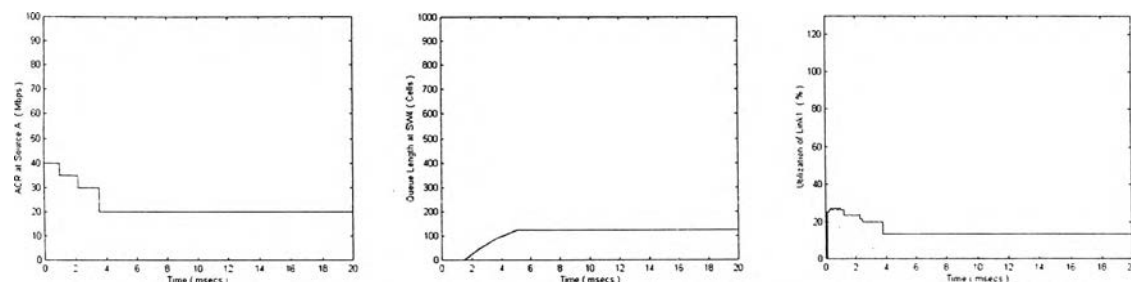
รูปที่ 5.56 ผลการจำลองเมื่อใช้วิธี FMMRA ร่วมกับ RSS Algorithm สำหรับ Jiang configuration



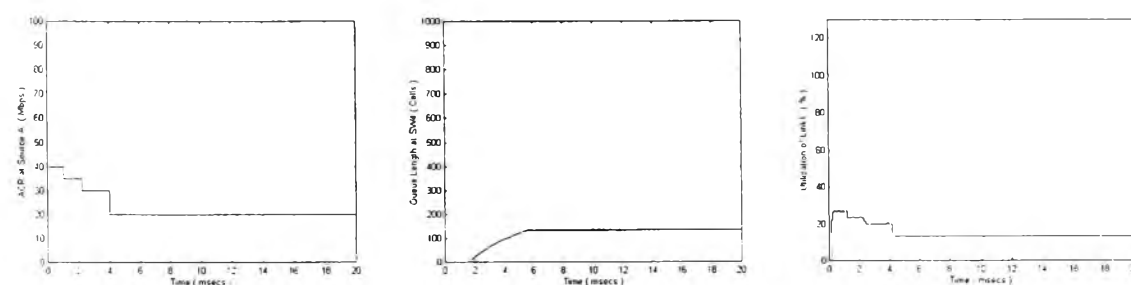
รูปที่ 5.57 ผลการจำลองเมื่อใช้วิธี FMMRA ร่วมกับ Wait-for-all Algorithm สำหรับ Jiang configuration



รูปที่ 5.58 ผลการจำลองเมื่อใช้วิธี FMMRA ร่วมกับ Immediate Rate Calculation Algorithm สำหรับ Jiang configuration



รูปที่ 5.59 ผลการจำลองเมื่อใช้วิธี FMMRA ร่วมกับวิธีการที่เสนอแบบที่ 1 สำหรับ Jiang configuration



รูปที่ 5.60 ผลการจำลองเมื่อใช้วิธี FMMRA ร่วมกับวิธีการที่เสนอแบบที่ 2 สำหรับ Jiang configuration

จะเห็นว่าผลการจำลองสำหรับวิธีรวบรวมข่าวสารความคับคั่งแต่ละแบบร่วมกับ FMMRA นั้นให้ผลที่เหมือนกับเมื่อนำมาใช้ร่วมกับ ERICA ทั้งนี้เนื่องจาก รูปแบบโครงข่ายนี้เป็นรูปแบบที่ง่ายไม่ซับซ้อน โดยมีแหล่งกำเนิดเพียงแหล่งเดียวเท่านั้น อีกทั้ง ERICA ยังมีการปรับค่า target utilization เป็น 100% ทำให้มีการใช้งานสายสัญญาณได้เต็มที่เช่นเดียวกับ FMMRA

5.4.4 เปรียบเทียบประสิทธิภาพระหว่าง ERICA และ FMMRA

ERICA และ FMMRA นั้นให้ผลการจำลองที่ไม่แตกต่างกัน เนื่องจากรูปแบบโครงข่ายนี้เป็นรูปแบบโครงข่ายที่ง่าย และ ERICA นั้นมีการตั้งค่า target utilization เป็น 100% นั้นเอง

5.4.5 เปรียบเทียบประสิทธิภาพระหว่างวิธีรวบรวมข่าวสารความคับคั่งต่าง ๆ

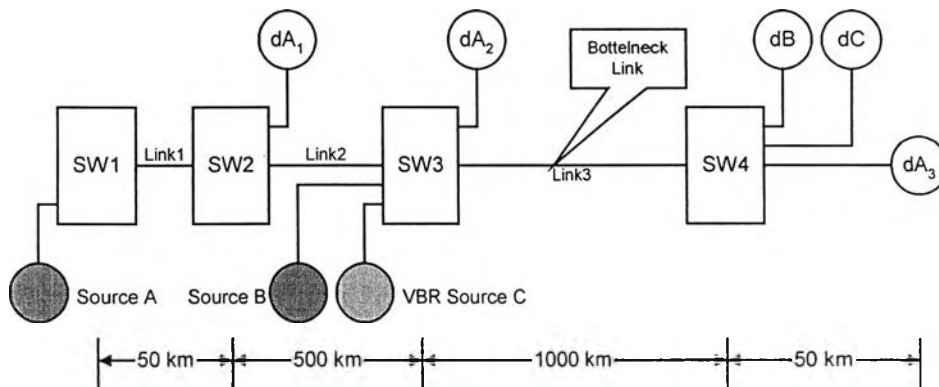
รูปแบบโครงข่ายนี้สามารถทดสอบผลตอบสนองชั่วคราวของวิธีรวบรวมข่าวสารความคับคั่งได้เป็นอย่างดี โดยวิธี Robert algorithm, RSS algorithm, วิธีการที่เสนอแบบที่ 1 และ 2 นั้นมีผลตอบสนองชั่วคราวที่รวดเร็ว โดยสามารถปรับอัตราส่งของแหล่งกำเนิดได้ตามสถานะของโครงข่าย ในขณะที่วิธี TS

algorithm นั้นสามารถปรับอัตราส่งได้ช้ากว่า ด้วยเงื่อนไขของการตอบกลับเซลล์ BRM ไม่ดีนั่นเอง ส่วน Wait-for-all และ Immediate rate calculation นั้นมีผลตอบสนองชั่วคราวที่ช้ามาก

5.5 ผลการจำลองสำหรับ VBR Chain configuration

5.5.1 รูปแบบโครงข่ายและค่ากำหนดต่าง ๆ

การส่งข้อมูลในโครงข่าย ATM จริงนั้น จะมีทราฟฟิกประเภท VBR ส่งร่วมกับทราฟฟิกประเภท ABR ด้วย ดังนั้นรูปแบบโครงข่ายนี้มีไว้เพื่อทดสอบสมรรถนะภาพในการส่งข้อมูลของวิธีควบคุมความคับคั่ง และ วิธีรวบรวมข่าวสารความคับคั่ง ภายใต้สภาวะที่ทราฟฟิก VBR นั้นมีการเปลี่ยนแปลงอย่างสูง อันจะทำให้แหล่งกำเนิดของ ABR นั้นต้องสามารถปรับอัตราส่งให้สอดคล้องกับทราฟฟิกที่โครงข่ายรองรับได้ โครงข่ายมีลักษณะคล้ายคลึงกับ chain configuration หากแต่เพิ่มแหล่งกำเนิด VBR ขึ้นมาเป็น source C โดย source C นี้เป็น on/off source ที่ทราบอัตราส่งแน่นอน รูปแบบโครงข่ายจะเป็นดังรูปที่ 5.61



รูปที่ 5.61 VBR chain configuration

ค่าตัวแปรต่าง ๆ ที่ใช้ในรูปแบบโครงข่ายนี้ได้แก่

Source A : PCR = 149.76 Mbps , ICR = 149.76 Mbps, MCR= 0 Mbps. RIF= 1 , RDF = 0.0125 และ Nrm = 32

Source B : PCR = 149.76 Mbps , ICR = 149.76 Mbps, MCR= 0 Mbps. RIF= 1 , RDF = 0.0125 และ Nrm = 32

Source C : เป็น on/off source ที่รู้ค่าอัตราส่งแน่นอน โดยช่วงมีช่วง on เป็น 100 ms และ ช่วง off เป็น 100 ms เช่นเดียวกัน โดยในช่วง on มีอัตราส่งที่ 130 Mbps และ หยุดส่งข้อมูลในช่วง off แหล่งกำเนิดอยู่ในสภาวะ ON ตั้งแต่เริ่มต้น

SW1 : Maximum queue length = 40,000

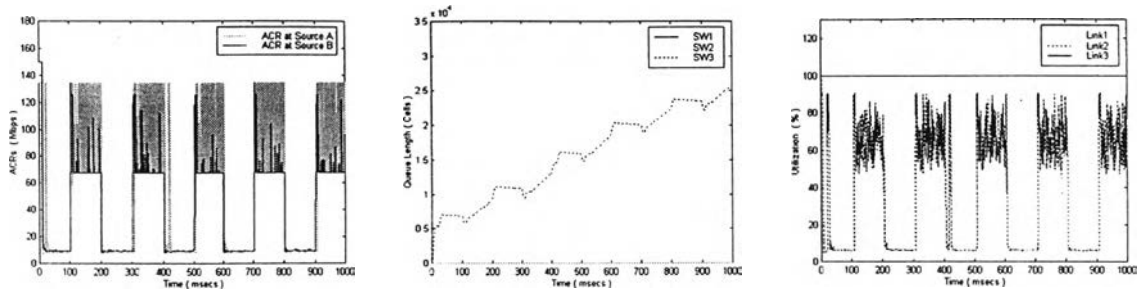
SW2 : Maximum queue length = 40,000

SW3 : Maximum queue length = 40,000

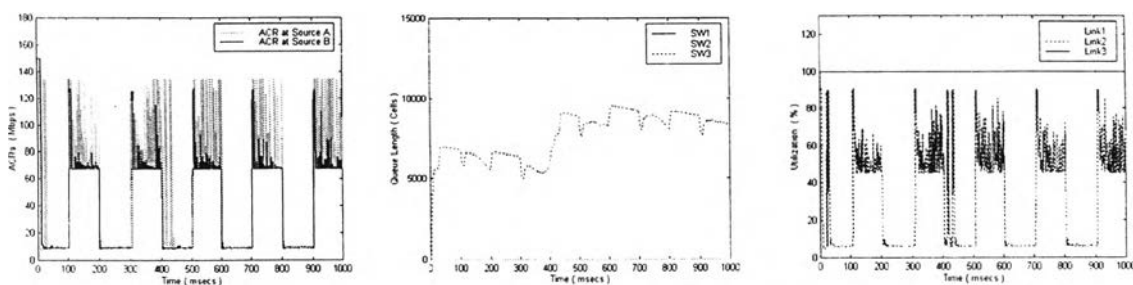
SW4 : Maximum queue length = 40,000

5.5.2 สมรรถนะของวิธี ERICA

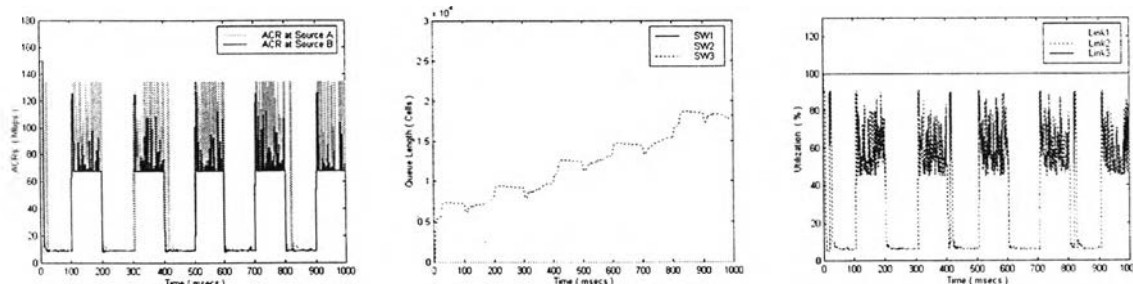
เมื่อใช้ ERICA ร่วมกับวิธีรวบรวมความคับคั่งต่าง ๆ จะได้ผลดังรูปที่ 5.62 ถึง 5.68



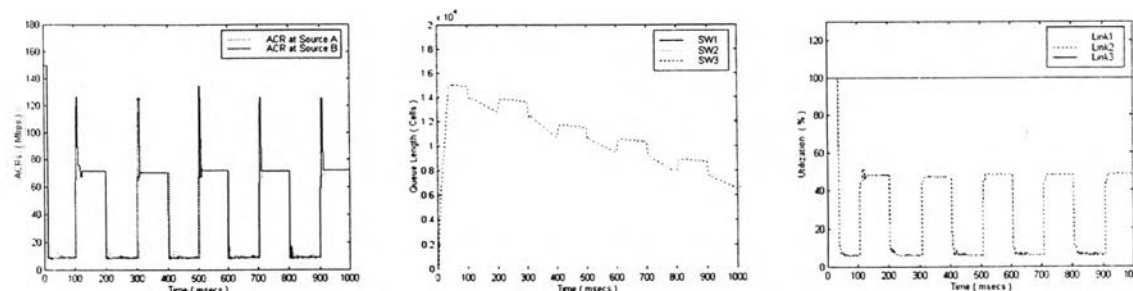
รูปที่ 5.62 ผลการจำลองเมื่อใช้วิธี ERICA ร่วมกับ Robert Algorithm สำหรับ VBR chain configuration



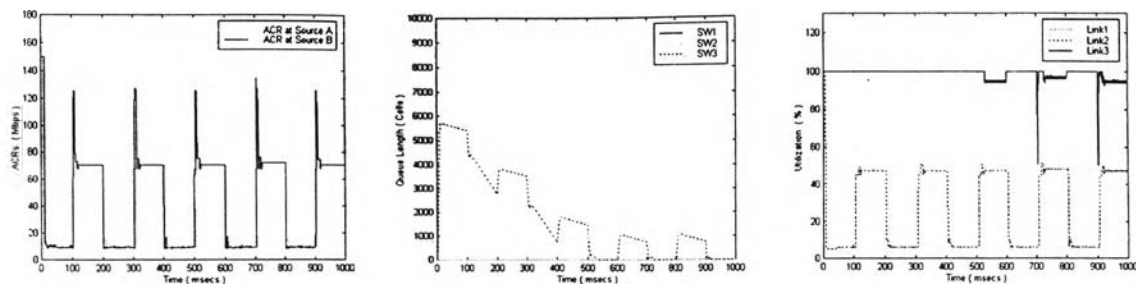
รูปที่ 5.63 ผลการจำลองเมื่อใช้วิธี ERICA ร่วมกับ TS Algorithm สำหรับ VBR chain configuration



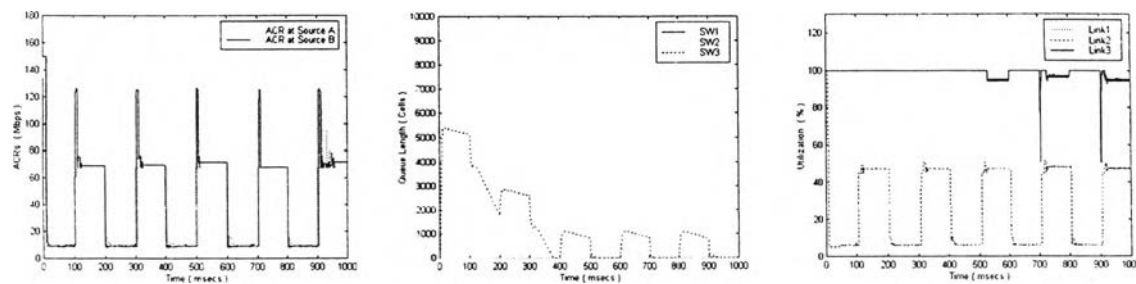
รูปที่ 5.64 ผลการจำลองเมื่อใช้วิธี ERICA ร่วมกับ RSS Algorithm สำหรับ VBR chain configuration



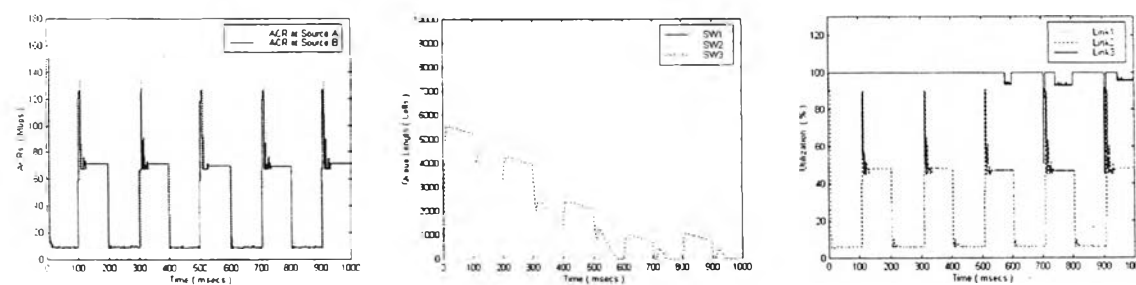
รูปที่ 5.65 ผลการจำลองเมื่อใช้วิธี ERICA ร่วมกับ Wait-for-all Algorithm สำหรับ VBR chain configuration



รูปที่ 5.66 ผลการจำลองเมื่อใช้วิธี ERICA ร่วมกับ Immediate Rate Calculation Algorithm สำหรับ VBR chain configuration



รูปที่ 5.67 ผลการจำลองเมื่อใช้วิธี ERICA ร่วมกับวิธีการที่เสนอแบบที่ 1 สำหรับ VBR chain configuration



รูปที่ 5.68 ผลการจำลองเมื่อใช้วิธี ERICA ร่วมกับวิธีการที่เสนอแบบที่ 2 สำหรับ VBR chain configuration

เนื่องจากโครงข่ายนี้มี VBR ที่เป็น on/off source อยู่ด้วย ทำให้ค่า fairshare สำหรับแต่ละทุกแหล่งกำเนิดสามารถแบ่งได้เป็นสองสภาวะ โดยในสภาวะที่แหล่งกำเนิด VBR นั้น on อยู่ นั่น ค่า fairshare จะเป็น 9.88 Mbps ส่วนในสภาวะ off จะมีค่าเป็น 74.88 Mbps อย่างไรก็ตามเนื่องจาก ERICA มีการทำงานโดยใช้ค่า z ที่อยู่ระหว่าง 1-1.1 ดังนั้นอัตราส่งที่แหล่งกำเนิดได้รับอนุญาตจริงจะมีค่าสูงกว่าค่าอัตราส่งเท่าเทียมกันเล็กน้อย

เมื่อพิจารณาค่า ACR ของแหล่งกำเนิด A ในรูปที่ 5.62 จะเห็นว่าผลตอบของ Robert algorithm นั้นมีผลตอบชั่วคราวที่รวดเร็วมาก โดยสามารถปรับค่า ACR ให้ลดลงสู่ fairshare ได้อย่างรวดเร็ว อย่างไรก็ตามวิธีการนี้ยังก่อให้เกิด consolidation noise อย่างหนัก โดยเฉพาะในสภาวะที่แหล่งกำเนิด VBR นั้นเปลี่ยนสถานะจากช่วง ON ไปสู่ช่วง OFF ซึ่งจะทำให้แหล่งกำเนิด ABR ทั้งสองนั้นสามารถปรับอัตราส่งของตนเองขึ้นมาได้ จะเห็นว่า ERICA นั้นสามารถปรับอัตราส่งของแหล่งกำเนิดไปสู่ค่า fairshare ได้อย่างรวดเร็ว การปรับค่าอัตราส่งของแหล่งกำเนิดขึ้นนี้เองที่เป็นสาเหตุของการเกิด consolidation noise ในลักษณะที่คล้ายคลึงกับในกรณี Chain configuration เมื่อแหล่งกำเนิดเพิ่มอัตราส่งขึ้นอย่างรวดเร็วจะ

ทำให้ปริมาณเซลล์ FRM ที่ส่งออกจากแหล่งกำเนิดในช่วงเวลาหนึ่งเพิ่มขึ้นด้วย และทำให้สวิตช์นั้นได้รับเซลล์ FRM เพิ่มขึ้น ในขณะที่จำนวนเซลล์ BRM ที่ได้รับนั้นมีปริมาณเท่าเดิม แต่เนื่องจาก Robert algorithm นั้นใช้เงื่อนไขในการตอบกลับเซลล์ BRM ด้วยการตรวจสอบจากการรับเซลล์ FRM และจะทำการลบค่า MER ที่รวบรวมไว้ให้เป็นค่าที่ได้รับจาก FRM ซึ่งมักจะเป็นค่า PCR ที่มีค่าสูงนั่นเอง ด้วยเหตุนี้สวิตช์นั้นต้องตอบกลับเซลล์ BRM มากขึ้น และต้องลบค่า MER บ่อยขึ้น ในขณะที่ปริมาณข่าวสารการป้อนกลับที่ได้รับนั้นมีปริมาณเท่าเดิม ทำให้เกิดความผิดพลาดในการรวบรวมข้อมูลในตัวแปร MER นั่นเอง และก่อให้เกิดการแกว่งตัวของค่า ACR ที่แหล่งกำเนิด A ขึ้น Consolidation noise ที่เกิดขึ้นนั้นทำให้ขนาดหน่วยความจำที่ SW3 นั้นเพิ่มขึ้นในระดับที่ไม่สามารถควบคุมได้

ผลการจำลองของ TS algorithm ดังรูปที่ 5.63 สังเกตได้ว่าผลตอบชั่วคราวนั้นมีความรวดเร็วเช่นเดียวกับ Robert algorithm แต่ยังคงเกิด consolidation noise ขึ้น แต่อยู่ในระดับที่น้อยกว่า Robert algorithm เนื่องจากวิธีดังกล่าวตรวจสอบการได้รับเซลล์ BRM ก่อนการส่งเซลล์ BRM กลับสู่แหล่งกำเนิดทันทีหลังจากที่ได้รับเซลล์ FRM ซึ่งสามารถลด consolidation noise ได้ แต่อย่างไรก็ตาม consolidation noise ที่เกิดก่อให้เกิดความคับคั่งอย่างรุนแรงในระดับที่ยอมรับไม่ได้ สังเกตได้จากขนาดหน่วยความจำที่ SW3 มีการแกว่งตัวตลอด และมีแนวโน้มที่จะเพิ่มขึ้นเรื่อย ๆ

RSS algorithm ตามรูปที่ 5.64 นั้น ยังคงเกิด consolidation noise ในช่วงที่แหล่งกำเนิด VBR นั้น OFF อยู่ โดยระดับการเกิด consolidation noise นั้นจะมีเท่า ๆ กับ Robert algorithm ส่วนสาเหตุการเกิด consolidation noise นั้นก็เหมือนกับที่เกิดใน Robert algorithm การเกิด consolidation noise ขึ้นนั้นทำให้หน่วยความจำที่ SW3 มีปริมาณเพิ่มขึ้นตลอด แสดงให้เห็นถึงความคับคั่งที่เกิดขึ้นในโครงข่ายส่วน Wait-for-all ตามรูปที่ 5.65 นั้นสามารถขจัด consolidation noise ไปได้ แต่ก็มีผลตอบสนองชั่วคราวที่ช้ามาก ซึ่งสังเกตได้จากค่า ACR ของแหล่งกำเนิด A ซึ่งปรับตัวช้ากว่า Robert algorithm เนื่องจากแหล่งกำเนิดต้องรอให้เซลล์ BRM เดินทางกลับมาจากปลายทางที่ไกลที่สุดก่อน การปรับตัวของค่า ACR ที่ช้าไปนี้เองทำให้โครงข่ายอยู่ในสภาวะติดขัดเป็นเวลานาน ส่งผลให้ขนาดหน่วยความจำที่ SW3 นั้นเพิ่มสูงขึ้นมาก

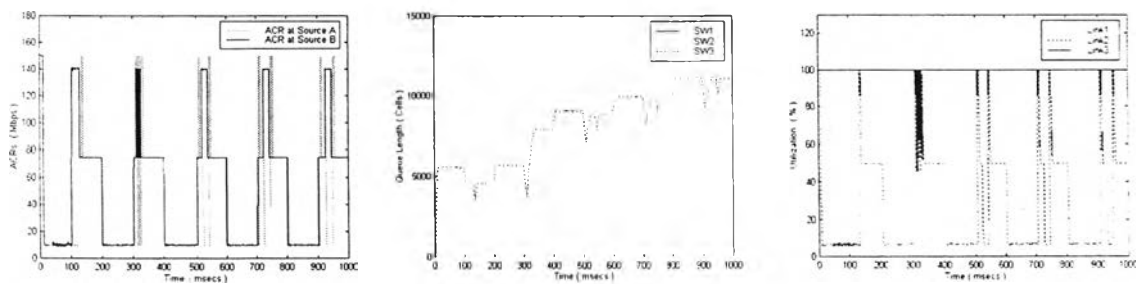
จากรูปที่ 5.66 จะเห็นว่า Immediate rate calculation algorithm นั้นสามารถขจัด consolidation noise ไปได้ ในขณะที่มีผลตอบสนองชั่วคราวที่รวดเร็ว เนื่องจากโครงข่ายอยู่ในสภาวะโหลดเกิน ทำให้กลไก overload detected ของ Immediate rate calculation algorithm นั้นทำงาน โดยการส่งเซลล์ BRM กลับไปสู่แหล่งกำเนิดทันที ส่วน Consolidation noise ถูกจำกัดไปได้เนื่องจากวิธีนี้ใช้หลักการเดียวกับ Wait-for-all algorithm โดยการรวบรวมข่าวสารความคับคั่งให้ครบจากทุก branch ก่อน

วิธีการที่เสนอแบบที่ 1 นั้นสามารถขจัด consolidation noise ได้เช่นเดียวกัน ในขณะที่มีผลตอบสนองชั่วคราวที่รวดเร็ว ด้วยข้อดีที่เก็บรวบรวมข่าวสารความคับคั่งจากทุก branch ไว้ ทำให้สามารถขจัด consolidation noise ไปได้ และการใช้เงื่อนไขในการตอบกลับเซลล์ BRM ที่เหมือนกับที่ใช้ใน Robert

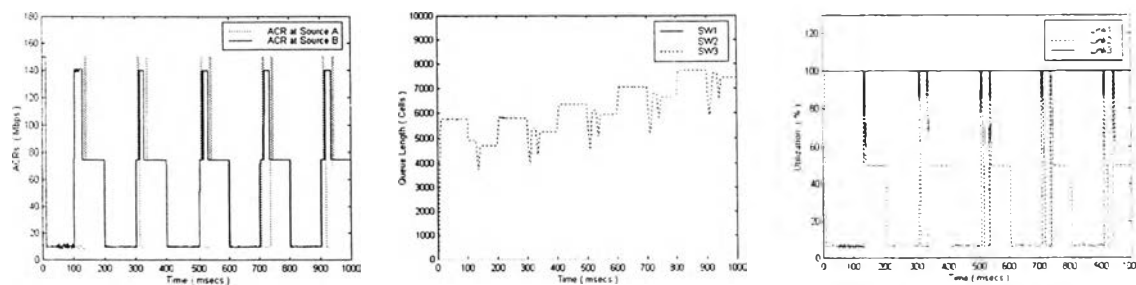
algorithm ทำให้มีผลตอบชั่วคราวที่รวดเร็ว ส่งผลให้ขนาดหน่วยความจำที่ SW3 นั้นมีค่าไม่มากเท่าไรนัก ในขณะที่การใช้งานสายสัญญาณนั้นมีประสิทธิภาพเต็มที่ ส่วนวิธีการที่เสนอแบบที่ 2 ตามรูปที่ 5.68 นั้นสามารถลดการเกิด consolidation noise ให้อยู่ในระดับที่น้อยมาก และสามารถยอมรับได้สังเกตได้จากค่า ACR ของแหล่งกำเนิด A ในระหว่างที่แหล่งกำเนิด VBR นั้นปรับการสถานะ ON ไปสู่ OFF นั้น จะเกิด consolidation noise ในช่วงแรก แต่ระดับการเกิด consolidation noise จะค่อย ๆ น้อยลง เนื่องจากวิธีการนี้ไม่ทำการลบล้างค่า MER ที่เก็บรวบรวมไว้ให้มีค่าเป็นค่า PCR นั้นเอง หากแต่ตั้งให้เป็นค่า ER ที่เพิ่งได้รับจากเซลล์ BRM นั้นเอง วิธีการนี้ยังไม่สามารถขจัด consolidation noise ไปได้ แต่ก็ป้องกัน consolidation noise ที่เกิดขึ้นนั้นไม่ให้เกิดต่อเนื่องไปเรื่อย ๆ จนโครงข่ายเกิดความคับคั่งมากเกินไป จะเห็นว่าวิธีการนี้สามารถควบคุม consolidation noise ให้อยู่ในระดับที่สามารถควบคุมได้ โดยที่ขนาดหน่วยความจำที่สวิตช์นั้นมีค่าไม่มากนัก ในขณะที่การใช้งานสายสัญญาณนั้นมีประสิทธิภาพดี

5.5.3 สมรรถนะของวิธี FMMRA

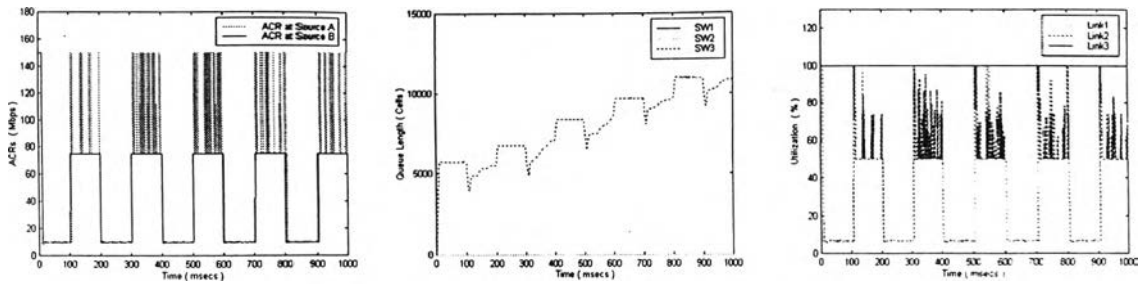
เมื่อใช้ FMMRA ร่วมกับวิธีรวบรวมความคับคั่งต่าง ๆ จะได้ผลดังรูปที่ 5.69 ถึง 5.75



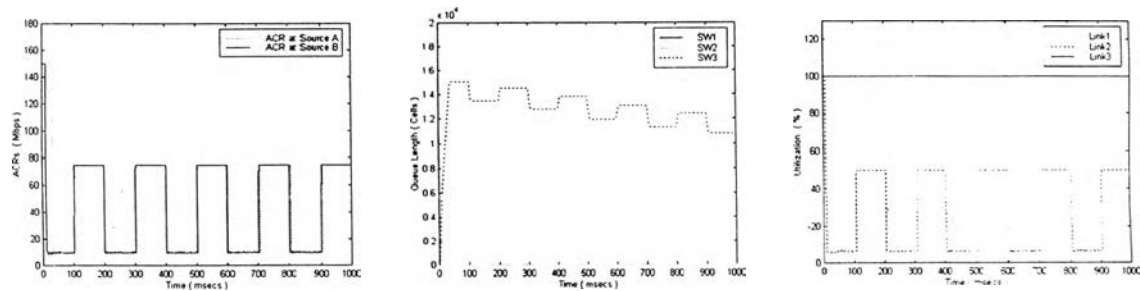
รูปที่ 5.69 ผลการจำลองเมื่อใช้วิธี FMMRA ร่วมกับ Robert Algorithm สำหรับ VBR chain configuration



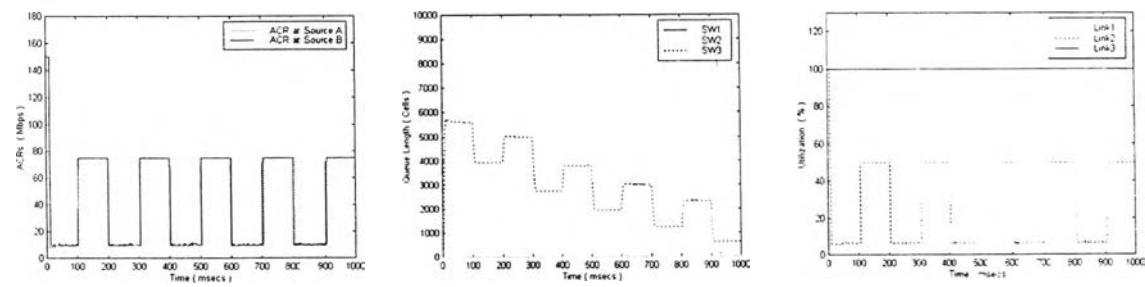
รูปที่ 5.70 ผลการจำลองเมื่อใช้วิธี FMMRA ร่วมกับ TS Algorithm สำหรับ VBR chain configuration



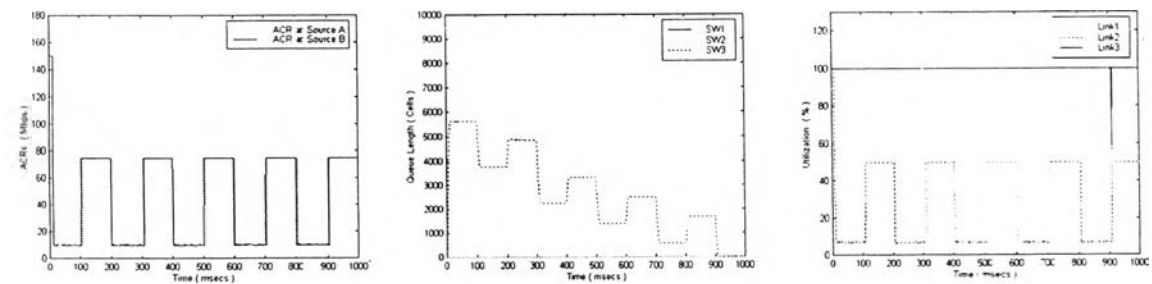
รูปที่ 5.71 ผลการจำลองเมื่อใช้วิธี FMMRA ร่วมกับ RSS Algorithm สำหรับ VBR chain configuration



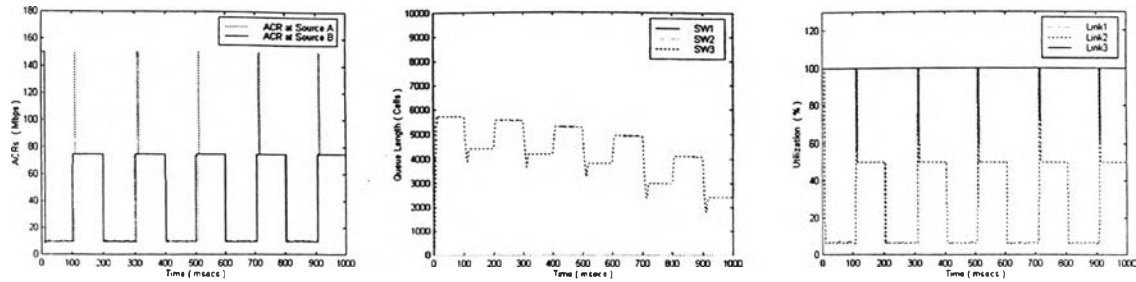
รูปที่ 5.72 ผลการจำลองเมื่อใช้วิธี FMMRA ร่วมกับ Wait-for-all Algorithm สำหรับ VBR chain configuration



รูปที่ 5.73 ผลการจำลองเมื่อใช้วิธี FMMRA ร่วมกับ Immediate Rate Calculation Algorithm สำหรับ VBR chain configuration



รูปที่ 5.74 ผลการจำลองเมื่อใช้วิธี FMMRA ร่วมกับวิธีการที่เสนอแบบที่ 1 สำหรับ VBR chain configuration



รูปที่ 5.75 ผลการจำลองเมื่อใช้วิธี FMMRA ร่วมกับวิธีการที่เสนอแบบที่ 2 สำหรับ VBR chain configuration

จากรูปที่ 5.59 สังเกตได้ว่าเมื่อนำ Robert algorithm มาใช้งานร่วมกับวิธี FMMRA นั้น จะเกิด consolidation noise ขึ้น โดยเฉพาะเมื่อ VBR source เปลี่ยนจากสถานะ ON ไปสู่สถานะ OFF นอกจากนี้ยังทำให้ FMMRA เกิดความผิดพลาดในการคำนวณค่า ER ที่แบ่งให้กับแต่ละการเชื่อมต่อในโครงข่าย สังเกตได้จากค่าอัตราส่งของแหล่งกำเนิด A นั้นมีการแกว่งตัวในลักษณะผิดปกติ โดยมีแหล่งกำเนิด B ซึ่งเป็น point-to-multipoint connection นั้นสามารถปรับอัตราส่งได้มากกว่าปกติ ส่วนแหล่งกำเนิด A ซึ่งเป็น point-to-multipoint connection นั้นไม่สามารถปรับอัตราส่งเพิ่มขึ้นได้ ทั้งนี้เนื่องจาก FMMRA นั้นมีการคำนวณค่า Bottleneck status ผิดพลาดอันเนื่องมาจากการรวบรวมข่าวสารความคับคั่งที่ผิดพลาดนั่นเอง ส่วนสาเหตุที่ทำให้เกิด consolidation noise และความผิดพลาดของข่าวสารความคับคั่งที่รวบรวมไว้ที่สวิตช์นั้น คือสาเหตุเช่นเดียวกับในกรณีที่ใช้ ERICA นั่นเอง หากแต่ ERICA นั้นคำนวณค่า ER จากค่า Fairshare และ ค่า Vcshare ที่เก็บไว้นั่นเอง ส่วนในกรณีของ FMMRA นั้น มีการนำค่า ER ที่ได้จากเซลล์ RM มามีส่วนในการคำนวณหาค่า ER ที่จะทำการป้อนกลับ วิธี Robert algorithm นี้ต้องมีการสร้างเซลล์ RM ขึ้นมา ดังนั้นค่า ER ในเซลล์ RM ที่สร้างขึ้นนี้ก็คือค่า MER ที่เก็บรวบรวมไว้จากการรับเซลล์ BRM จาก branch ต่าง ๆ นั่นเอง ด้วยเหตุนี้หากค่า ER ที่ได้จากเซลล์ RM นั้นผิดพลาดจากการรวบรวมข่าวสารความคับคั่งที่ไม่ดีพอ ก็จะทำให้ FMMRA นั้นคำนวณค่า ER ผิดพลาดตามไปด้วย

ความผิดพลาดในการคำนวณของ FMMRA และการเกิด consolidation noise นั้น ทำให้โครงข่ายไม่อยู่ในสถานะอยู่ตัว และ ค่าอัตราส่งที่แหล่งกำเนิดนั้นเกิดการแกว่งตัวอย่างรุนแรง ซึ่งนำไปสู่ความคับคั่งของโครงข่าย ดังสังเกตได้จากขนาดหน่วยความจำที่สวิตช์ที่เพิ่มขึ้นเรื่อย ๆ จนอยู่ในระดับที่ไม่สามารถควบคุมได้ ในขณะที่การใช้งานสายสัญญาณนั้นก็มีการแกว่งตัวในลักษณะคล้าย ๆ กับค่า ACR ที่แหล่งกำเนิดนั่นเอง

สำหรับ TS algorithm ในรูปที่ 5.70 นั้นมีผลการจำลองที่คล้ายคลึงกับ กรณีของ Robert algorithm โดยที่ FMMRA นั้นยังคงคำนวณค่า ER ที่ผิดพลาด และยังคงเกิด consolidation noise ขึ้น เนื่องจากวิธีนี้มีการสร้างเซลล์ BRM ก่อนการตอบกลับ แล้วนำค่า MER บรรจุเป็นค่า ER ในเซลล์ RM ก่อนที่จะให้ FMMRA คำนวณค่า ER ที่ควรจะเป็น ซึ่งหากค่า MER ที่รวบรวมไว้นั้นมีความผิดพลาดจาก consolidation noise ก็จะทำให้ FMMRA นั้นมีการคำนวณค่า ER ที่ผิดพลาดตามไปด้วย อย่างไรก็ตาม

การแกว่งตัวของค่า ACR ที่แสดงถึง consolidation noise นั้นมีน้อยลง ซึ่งมีสาเหตุจากการที่ TS algorithm ตรวจสอบการรับเซลล์ BRM ก่อนการสร้างเซลล์ BRM เพื่อการตอบกลับนั่นเอง อย่างไรก็ตามการแกว่งตัวของค่า ACR รวมถึงความผิดพลาดในการคำนวณค่า ER ของ FMMRA นั้นทำให้โครงข่ายเกิดความคับคั่งขึ้น ซึ่งสังเกตได้จากขนาดหน่วยความจำที่สวิตช์ที่เพิ่มขึ้นเรื่อย ๆ จนอยู่ในระดับที่ไม่สามารถควบคุมได้

RSS algorithm ตามรูปที่ 5.71 นั้น จะเห็นว่าเกิด Consolidation noise ขึ้นในสถานะที่แหล่งกำเนิด VBR นั้น OFF เช่นเดียวกับ Robert algorithm หากแต่ไม่เกิดความผิดพลาดในการคำนวณจาก FMMRA algorithm ทั้งนี้เนื่องจาก วิธีการนี้ไม่มีการสร้างเซลล์ BRM ในการตอบกลับ หากแต่นำเซลล์ BRM ที่ได้รับตัวแรกหลังจากได้รับเซลล์ FRM มาใช้ในการตอบกลับ ทำให้สามารถใช้ค่า ER ในเซลล์ BRM ที่ได้รับนั้นมาคำนวณหาค่า ER โดยใช้ FMMRA ก่อนนำไปเปรียบเทียบกับค่า MER ข้อมูลที่ FMMRA ใช้ในการคำนวณนั้นไปค่าที่ได้รับจากโครงข่ายจริง ทำให้ไม่เกิดความผิดพลาดในการคำนวณค่า ER อย่างไรก็ตาม วิธีการนี้ยังคงเกิด consolidation noise ขึ้นด้วยสาเหตุเช่นเดียวกับในกรณีเมื่อใช้ ERICA ซึ่งการเกิด consolidation noise นี้เอง ทำให้โครงข่ายอยู่ในสถานะคับคั่ง สังเกตได้จากขนาดหน่วยความจำที่มีค่าเพิ่มขึ้นเรื่อย ๆ นั่นเอง

ผลการจำลอง ของ Wait-for-all algorithm นั้น แสดงในรูปที่ 5.72 ซึ่งจะเป็นว่าวิธีนี้สามารถขจัด consolidation noise ไปได้ เนื่องจากวิธีการนี้ต้องรวบรวมข่าวสารให้ครบถ้วนก่อนการป้อนกลับเซลล์ BRM นั่นเอง อย่างไรก็ตามวิธีการนี้จะมีผลตอบสนองชั่วคราวที่ช้ามาก โดยโครงข่ายในช่วงแรกนั้นอยู่ในสถานะคับคั่ง โดยแหล่งกำเนิดสมควรลดอัตราส่งมาที่ fairshare โดยเร็วที่สุด วิธีการ Wait-for-all algorithm นั้นไม่สามารถปรับอัตราส่งให้ลดลงมาที่ fairshare ได้อย่างรวดเร็ว เนื่องจากแหล่งกำเนิดนั้นต้องรอให้เซลล์ BRM เดินทางกลับมาจากปลายทางที่ไกลที่สุดก่อน ด้วยเหตุนี้โครงข่ายจึงอยู่ในสถานะคับคั่งเป็นเวลานานในช่วงเริ่มต้น ทำให้ขนาดหน่วยความจำที่สวิตช์นั้นมีค่ามากตามไปด้วย ส่วนการใช้งานสายสัญญาณนั้นก็มีประสิทธิภาพดีตามที่สมควรได้รับ

ผลการจำลองของวิธี Immediate rate calculation ตามรูปที่ 5.73 และ วิธีการที่เสนอแบบที่ 1 ตามรูปที่ 5.74 นั้นให้ผลที่คล้ายคลึงกันมาก โดยสามารถขจัด consolidation noise ไปได้ ในขณะที่มีผลตอบสนองชั่วคราวที่รวดเร็ว วิธี Immediate rate calculation นั้นมีการทำงานที่คล้ายคลึงกับ Wait-for-all algorithm ทำให้สามารถขจัด consolidation noise ไปได้ และ โครงข่ายอยู่ในสถานะไหลตึงตั้งแต่เริ่มต้น ทำให้เทคนิค overload detect ของวิธีการนี้เกิดผล ทำให้มีการส่งเซลล์ BRM กลับไปที่ และทำให้แหล่งกำเนิดปรับอัตราส่งได้ทันที่ ส่วนวิธีการที่เสนอแบบที่ 1 นั้น ถึงแม้จะมีการสร้างเซลล์ BRM เพื่อการตอบกลับเช่นเดียวกับ Robert algorithm แต่ด้วยความถูกต้องของข้อมูลที่เก็บรวบรวมไว้ นั้น ทำให้วิธีการนี้ไม่ก่อให้เกิด consolidation noise และไม่ทำให้เกิดความผิดพลาดในการคำนวณค่า ER ของวิธี FMMRA ด้วย ในขณะที่มีผลตอบสนองชั่วคราวที่รวดเร็วเช่นเดียวกับ Robert algorithm ซึ่ง

จะเห็นว่าวิธีการ Immediate rate calculation และ วิธีการที่เสนอแบบที่ 1 นั้นทำให้แหล่งกำเนิดสามารถปรับอัตราส่งให้เหมาะสมกับโครงข่ายได้เป็นอย่างดี ทำให้โครงข่ายไม่อยู่ในสถานะคับคั่ง ส่งผลให้ขนาดหน่วยความจำสูงสุดที่สวิทช์มีค่าไม่มากนัก ส่วนการใช้งานสายสัญญาณทุกเส้นทางนั้นสามารถทำได้อย่างมีประสิทธิภาพที่สุด

สังเกตจากรูปที่ 5.75 จะเห็นว่าผลการจำลองสำหรับวิธีการที่เสนอแบบที่ 2 เมื่อใช้งานร่วมกับวิธี FMMRA นั้น ยังคงมี consolidation noise ในช่วงแรกที่แหล่งกำเนิด VBR นั้นเปลี่ยนจากสถานะ ON ไปสู่สถานะ OFF โดย consolidation noise ที่เกิดขึ้นนั้น เกิดขึ้นเฉพาะในเริ่มต้นการเปลี่ยนสถานะเท่านั้น และวิธีนี้สามารถควบคุม consolidation noise ให้อยู่ในระดับที่สามารถควบคุมได้ในที่สุด ในขณะที่มีผลตอบสนองชั่วคราวที่รวดเร็วมาก ซึ่งสังเกตได้จากการปรับค่าอัตราส่งของแหล่งกำเนิดที่สามารถปรับสู่ fairshare ได้อย่างรวดเร็วในช่วงแรก ทำให้โครงข่ายไม่อยู่ในสถานะคับคั่ง และขนาดหน่วยความจำสูงสุดที่สวิทช์นั้นมีค่าน้อย ในขณะที่การใช้งานสายสัญญาณทุกเส้นทางนั้นมีประสิทธิภาพดี

5.5.4 เปรียบเทียบประสิทธิภาพระหว่าง ERICA และ FMMRA

วิธี ERICA และ FMMRA นั้นสามารถตอบสนองต่อปริมาณทราฟฟิกที่เปลี่ยนแปลงไปของแหล่งกำเนิด VBR โดยสามารถปรับอัตราส่งของแหล่งกำเนิดไปสู่ fairshare ได้อย่างรวดเร็ว แต่วิธี FMMRA นั้นสามารถปรับอัตราส่งได้รวดเร็วกว่าวิธีของ ERICA หากแต่ยังมีความผิดพลาดในการคำนวณค่า ER ในกรณีเมื่อใช้วิธีรวบรวมข่าวสารความคับคั่งคือ Robert algorithm และ TS algorithm

5.5.5 เปรียบเทียบประสิทธิภาพระหว่างวิธีรวบรวมข่าวสารความคับคั่งต่าง ๆ

Robert algorithm, TS algorithm และ RSS algorithm ยังคงมีปัญหาในการเกิด consolidation noise อย่างหนัก และนำไปสู่การคำนวณค่า ER ที่ผิดพลาดเมื่อใช้วิธี FMMRA ร่วมกับ Robert algorithm และ TS algorithm ในขณะที่วิธีทั้งสามนั้นมีผลตอบสนองชั่วคราวที่ดีมาก ซึ่งทำให้โครงข่ายสามารถหลุดจากสถานะคับคั่งในช่วงแรกได้อย่างรวดเร็ว วิธี wait-for-all algorithm นั้นสามารถขจัด consolidation noise ไปได้ แต่มีปัญหาเรื่องผลตอบสนองชั่วคราวที่ช้ามาก ในขณะที่วิธี Immediate rate calculation algorithm และ วิธีการที่เสนอแบบที่ 1 นั้นให้สมรรถนะที่ดีที่สุด โดยสามารถขจัด consolidation noise ไปได้ ในขณะที่มีผลตอบสนองชั่วคราวที่รวดเร็วมาก วิธีการที่เสนอแบบที่ 2 นั้นมีผลตอบสนองชั่วคราวที่รวดเร็วเช่นกัน แต่ยังไม่สามารถขจัด consolidation noise ให้หมดไปอย่างสิ้นเชิงได้ แต่สามารถลด consolidation noise ให้น้อยมากจนอยู่ในระดับที่สามารถรับได้