บทที่ 6
บทสรุปและข้อเสนอแนะ

6.1 เปรียบเทียบประสิทธิภาพระหว่าง ERICA และFMMRA

วิธี ERICA และ FMMRA นั้นมีข้อดีและข้อเสียแตกต่างกันไป โดยเฉพาะเมื่อนำมาใช้ในการส่งจากหนึ่งจุดไปสู่หลายจุด โดยหัวข้อที่เราสนใจเปรียบเทียบประสิทธิภาพระหว่างสองวิธีนี้เมื่อนำมาใช้กับการส่งจากหนึ่งจุดไปสู่หลายจุด ได้แก่ Max-Min fairness, ผลตอบสนองชั่วครู่, Consolidation noise, ความซับซ้อนในการพัฒนา และ ความเหมาะสมในการใช้งานกับวิธีรวบรวมข่าวสารความคับคั่งต่าง ๆ

จากผลการจำลองสามารถสรุปได้ว่า วิธีควบคุมความคับคั่งนั้นมีผลต่อการทำงานของวิธีรวบรวมข่าวสารความคับคั่ง โดยที่วิธี FMMRA นั้นสามารถลดการเกิด consolidation noise ได้ในบางกรณีเมื่อเทียบกับวิธี ERICA ที่ยังคงมี consolidation noise ในสถานะการณ์เดียวกัน

6.1.1 Max-Min fairness

ในการศึกษาที่ผ่านมานั้น ได้มีการศึกษาด้าน Max-Min fairness ในการส่งจากหนึ่งจุดไปสู่หลายจุด ซึ่งผลการวิเคราะห์แสดงว่า Max-Min fairness นั้นสามารถนำมาใช้กับการส่งจากหนึ่งจุดไปสู่หลายจุดได้ จากการจำลองที่ผ่านมาได้แสดงให้เห็นถึงความสามารถในการปรับอัตราส่งให้เข้าสู่ Max-Min fairness ของทั้งวิธี ERICA และ FMMRA ได้เป็นอย่างดี

6.1.2 ผลตอบสนองชั่วครู่

ในการส่งแบบหนึ่งจุดไปสู่หลายจุดนั้น วิธี FMMRA นั้นสามารถปรับอัตราส่งของแหล่งกำเนิดให้เข้าสู่ค่า fairshare ได้อย่างรวดเร็วกว่าวิธี ERICA ทำให้ FMMRA นั้นมีผลตอบสนองชั่วครู่ที่รวดเร็วกว่าวิธี ERICA อย่างไรก็ตาม วิธี ERICA นั้นมีผลตอบสนองชั่วครู่ที่รวดเร็วเพียงพอต่อปรับอัตราส่งของแหล่งกำเนิดให้เข้ากับสภาวะของโครงข่ายได้เป็นอย่างดี ถึงแม้อยู่ในสภาวะที่เลวร้ายที่สุด

6.1.3 Consolidation noise

วิธี FMMRA นั้นไม่มีปัญหาเรื่อง Consolidation noise มากนัก เนื่องจากวิธี FMMRA นั้นสามารถปรับอัตราส่งที่แหล่งกำเนิดให้เข้าสู่ fairshare ได้อย่างรวดเร็ว และถูกต้อง ในขณะที่ ERICA นั้นเกิดปัญหา Consolidation noise ขึ้น โดยเฉพาะเมื่อใช้งานกับ Robert algorithm, TS algorithm และ RSS algorithm ทั้งนี้เนื่องจาก ERICA มีการปรับอัตราส่งเพิ่มขึ้นจาก fairshare เล็กน้อย ทำให้เกิด consolidation noise ขึ้น

Consolidation noise ที่เกิดขึ้นนั้นจะเกิดขึ้นเฉพาะในช่วงที่อัตราส่งมีการปรับอัตราส่งเพิ่มขึ้นเท่านั้น ดังแสดงได้ใน VBR Chain Configuration ซึ่งจะเห็นว่าวิธี FMMRA นั้นยังคงมีปัญหา

Consolidation noise เช่นเดียวกับวิธี ERICA เนื่องจากมีการปรับอัตราส่งเพิ่มขึ้นเนื่องจากแหล่งกำเนิด VBR หยุดส่งไป ทำให้แหล่งกำเนิดแบบ ABR สามารถเพิ่มอัตราส่งได้ และทำให้เกิด Consolidation noise ขึ้น

6.1.4 ความซับซ้อนในการพัฒนา

วิธี ERICA และ FMMRA นั้นมีความซับซ้อนในการคำนวณระดับ O(1) ทั้งคู่ โดยการคำนวณทุกครั้งจะทำการคำนวณรอบเดียวสำหรับการเชื่อมต่อนั้นเท่านั้น ทำให้วิธีการทั้งสองมีความซับซ้อนในระดับที่ใกล้เคียงกัน อย่างไรก็ตามวิธี FMMRA นั้นไม่สามารถนำมาใช้งานบนการส่งจากหนึ่งจุดไปสู่หลายจุดได้ทันที เนื่องจากตัวแปรต่าง ๆ ที่วิธี FMMRA ใช้นั้นเป็นตัวแปรเฉพาะการเชื่อมต่อ โดยแต่ละการเชื่อมต่ออาจมีเส้นทางขาออกมากกว่าหนึ่งเส้นทาง ทำให้สถานะของการเชื่อมต่อที่เก็บไว้ไม่เพียงพอต่อการคำนวณสำหรับหลายเส้นทางขาออก จึงจำเป็นต้องมีการเพิ่มตัวแปรต่าง ๆ สำหรับทุกการเชื่อมต่อให้เพิ่มขนาดเป็นเก็บทุกเส้นทางขาออกด้วย ทำให้ FMMRA นั้นมีความซับซ้อนมากขึ้น และมากกว่าวิธี ERICA

6.1.5 ความเหมาะสมในการใช้งานกับวิธีรวบรวมข่าวสารความคับคั่งต่าง ๆ

วิธี ERICA นั้นสามารถทำงานร่วมกับวิธีการต่าง ๆ ได้เป็นอย่างดี โดยไม่เกิดปัญหาในขบวนการคำนวณแต่อย่างใดแต่อาจเกิดปัญหา ในขณะที่วิธี FMMRA นั้นจะเกิดปัญหาในการคำนวณขึ้นโดยเฉพาะใน VBR chain configuration โดยเมื่อใช้งานร่วมกับ Robert algorithm และ TS algorithm จะเกิดความผิดพลาดในการตรวจสอบสถานะความคับคั่งของการเชื่อมต่อ อย่างไรก็ตามความผิดพลาดที่เกิดขึ้นนั้น มีสาเหตุมาจากความผิดพลาดในการคำนวณค่าการป้อนกลับของวิธีรวบรวมข่าวสารความคับคั่งด้วย

ตารางที่ 6.1 เปรียบเทียบประสิทธิภาพระหว่าง ERICA และ FMMRA

Algorithms	ERICA	FMMRA
Provide Max-Min fairness	Yes	Yes
ผลตอบสนองชั่วครู่	Fast	Very Fast
Consolidation noise	High	Low
ความซับซ้อน	Low	Medium
ความเหมาะสมในการใช้งานกับวิธีรวบรวมข่าวสารความคับคั่งต่าง ๆ	Good	Fair

6.2 เปรียบเทียบประสิทธิภาพระหว่างวิธีรวบรวมข่าวสารความคับคั่ง

ในการเปรียบเทียบประสิทธิภาพระหว่างวิธีรวบรวมข่าวสารความคับคั่งนั้น จะสนใจในเรื่อง Consolidation noise, ผลตอบสนองชั่วครู่, ความซับซ้อนในการพัฒนา, RM cell overhead, Change to non responsive branch และ Change to newly joined branch

6.2.1 Consolidation noise

วิธี Robert algorithm, TS algorithm และ RSS algorithm นั้นยังคงมีปัญหาเรื่อง consolidation noise ในระดับที่ไม่สามารถรับได้เมื่อใช้งานกับวิธี ERICA ถึงแม้ว่าเมื่อนำมาใช้งานกับวิธี FMMRA นั้นจะสามารถขจัด consolidation noise ได้ก็ตาม อย่างไรก็ตาม การเกิด consolidation noise นั้นเกิดได้ในเฉพาะบางกรณีเท่านั้น โดยเฉพาะเมื่อค่า RIF มีค่าสูง ทั้งนี้เนื่องจากเมื่อค่า RIF มีค่าสูงจะทำให้แหล่งกำเนิดนั้นสามารถเพิ่มอัตราส่งขึ้นได้อย่างรวดเร็ว อันจะทำให้เกิดการแกว่งตัวอย่างรุนแรงของค่า ACR และก่อให้เกิด consolidation noise ขึ้น อย่างไรก็หากค่า RIF ไม่เท่ากับ 1 นั้นจะทำให้แหล่งกำเนิดไม่สามารถปรับเพิ่มอัตราส่งได้อย่างรวดเร็ว และจะทำให้การใช้งานสายสัญญาณมีค่าน้อยตามไปด้วย วิธี RSS algorithm นั้นสามารถลด consolidation noise ได้ โดยการเพิ่มความถูกต้องของข่าวสารความคับคั่งที่ทำการป้อนกลับสู่แหล่งกำเนิด โดยไม่ทำการป้อนกลับข่าวสารความคับคั่งหากว่ายังไม่ได้รับเซลล์ BRM จาก branch ใด branch หนึ่ง อย่างไรก็ตามวิธีการดังกล่าวยังคงมีปัญหา consolidation noise อยู่เมื่อใช้งานกับวิธี ERICA

วิธี Wait-for-all algorithm, Immediate Rate Calculation algorithm และ วิธีการที่เสนอแบบที่ 1 นั้นสามารถขจัด Consolidation noise ไปได้ โดย Wait-for-all algorithm และ Immediate Rate Calculation algorithm นั้นรอให้มีการรับเซลล์ BRM จากทุก branch ก่อนการป้อนกลับเซลล์ BRM ทำให้ข่าวสารที่เก็บรวบรวมไว้ที่สวิตช์นั้นถูกต้องเสมอ ส่วนวิธีการที่เสนอแบบที่ 1 นั้นมีการจัดเก็บข่าวสารความคับคั่งที่ดี ทำให้มีความถูกต้องของข่าวสารที่รวบรวมไว้เสมอ และ ไม่เกิด consolidation noise ขึ้น

ส่วนวิธี วิธีการที่เสนอแบบที่ 2 นั้นสามารถลด Consolidation noise ให้อยู่ในระดับที่น้อยมาก ถึงแม้ว่าจะยังคงมี consolidation noise อยู่บ้างในบางกรณีเท่านั้น แต่ผลของ consolidation noise ที่เกิดขึ้นนั้นมีผลต่อโครงข่ายน้อยมาก โดยไม่ก่อให้เกิดความคับคั่งขึ้นในโครงข่าย ด้วยความสามารถในการหยุดยั้งไม่ให้ Consolidation noise ที่เกิดขึ้นมานั้น เกิดต่อเนื่องไปเรื่อย ๆ จนทำให้โครงข่ายอยู่ในสภาวะที่คับคั่งในที่สุด การไม่ลบล้างค่า MER ที่เก็บรวบรวมไว้หลังจากป้อนกลับเซลล์ BRM ไปแล้ว ทำให้ข่าวสารที่รวบรวมไว้มีความถูกต้องมากขึ้น และสามารถลด Consolidation noise ได้

6.2.2 ผลตอบสนองชั่วครู่ในสภาวะโหลดเกิน

ผลตอบสนองชั่วครู่ในสภาวะเริ่มต้นของโครงข่ายนั้นมีความสำคัญมาก เนื่องจากในช่วงเริ่มต้นนั้นหากโครงข่ายอยู่ในสภาวะคับคั่งเนื่องจากโหลดเกิน วิธีรวบรวมข่าวสารความคับคั่งที่มีผลตอบสนองชั่วครู่ที่ช้านั้นจะทำให้โครงข่ายอยู่ในสภาวะคับคั่งเป็นเวลานาน ซึ่งสามารถก่อให้เกิดการสูญเสียเซลล์ข้อมูลได้

วิธี Robert algorithm และ วิธีการที่เสนอแบบที่ 1 นั้น ให้ผลตอบสนองชั่วครู่ในสภาวะโหลดเกินได้รวดเร็วที่สุด เนื่องจากสามารถสร้างเซลล์ BRM ขึ้นมาเพื่อป้อนกลับได้ทันทีหลังจากรับเซลล์ FRM ตัวแรก การที่วิธีทั้งสองนั้นสามารถป้อนกลับเซลล์ BRM ได้อย่างรวดเร็ว ทำให้โครงข่ายหลุดออกจากสภาวะคับคั่งได้อย่างรวดเร็วตามไปด้วย

สำหรับ TS algorithm, RSS algorithm และ วิธีการที่เสนอแบบที่ 2 นั้นมีผลตอบสนองชั่วครู่ที่รวดเร็วเช่นเดียวกัน แต่การที่วิธีการทั้งสามนั้นต้องรอให้มีเซลล์ BRM เดินทางกลับมาจากปลายทางอย่างน้อย 1 ตัวก่อนจึงสามารถป้อนกลับเซลล์ BRM ได้ ทำให้วิธีการทั้งสามนั้นมีผลตอบสนองชั่วครู่ที่ช้ากว่า Robert algorithm

วิธี Immediate Rate Calculation algorithm นั้นมีผลตอบสนองชั่วครู่ในสภาวะโหลดเกินที่รวดเร็ว ทั้งนี้เนื่องจากวิธีดังกล่าวใช้เทคนิค Overload detected ซึ่งสามารถตรวจจับสภาวะโหลดเกินได้อย่างรวดเร็ว และทำให้สามารถป้อนกลับเซลล์ BRM กลับไปสู่แหล่งกำเนิดได้ทันทีในสภาวะโหลดเกิน ดังนั้นแหล่งกำเนิดจึงสามารถปรับอัตราส่งมาอยู่ที่ค่าที่โครงข่ายสามารถรองรับได้อย่างรวดเร็ว ทำให้ผลตอบสนองชั่วครู่ในสภาวะโหลดเกินนั้นรวดเร็วขึ้นด้วย อย่างไรก็ตามผลตอบสนองที่รวดเร็วนี้ขึ้นอยู่กับการตรวจจับสภาวะโหลดเกิน ซึ่งหากค่า threshold ที่ตั้งไว้เพื่อตรวจจับสภาวะโหลดเกินนั้นไม่ดีพอ จะทำให้ไม่สามารถตรวจจับสภาวะโหลดเกินได้ในบางกรณีไม่ได้

Wait-for-all algorithm นั้นให้ผลตอบสนองชั่วครู่ในสภาวะโหลดเกินที่ช้ามาก ทั้งนี้วิธีนี้ต้องรอให้เซลล์ BRM จากทุก brach เดินทางกลับมาถึงสวิตช์ก่อน ดังนั้นในสภาวะที่ยังไม่มีเซล์ RM ในโครงข่ายมากนักเช่นในช่วงเริ่มต้นนั้น สวิตช์จะต้องรอให้เซลล์ BRM จากปลายทางที่ไกลที่สุดเดินทางมาถึงก่อน จึงสามารถป้อนกลับเซลล์ BRM กลับไปได้ ซึ่งหากว่าปลายทางที่ไกลที่สุดนั้นมีเวลาประวิงสูงมาก ก็จะทำให้ผลตอบสนองชั่วครู่ในสภาวะโหลดเกินนั้นช้าตามไปด้วย

6.2.3 ผลตอบสนองชั่วครู่ในสภาวะโหลดต่ำ

ในสภาวะโหลดน้อยนั้น วิธีควบคุมข่าวสารความคับคั่งต้องสามารถปรับอัตราส่งของแหล่งกำเนิดให้เพิ่มขึ้นได้ เพื่อให้โครงข่ายสามารถใช้งานสายสัญญาณได้อย่างคุ้มค่าที่สุด ซึ่งหากว่าวิธีรวบรวมข่าวสารความคับคั่งแบบใดที่ให้ผลตอบสนองชั่วครู่ในสภาวะโหลดต่ำช้า ก็จะทำให้เกิดการใช้งานสัญญาณที่ไม่คุ้มค่า โดยวิธี Robert algorithm และ วิธีการที่เสนอแบบที่ 1 นั้นยังคงมีผลตอบสนองใน

สภาวะโหลดน้อยที่รวดเร็วที่สุดเช่นเดียวกับในกรณีโหลดเกิน ด้วยสาเหตุเดียวกันคือ วิธีการทั้งสองสามารถตอบกลับเซลล์ BRM ทันทีที่ได้รับเซลล์ FRM ตัวแรก ในขณะที่วิธี TS algorithm, RSS algorithm และ วิธีการที่เสนอแบบที่ 2 นั้นยังคงมีผลตอบสนองชั่วครู่ที่รวดเร็ว โดยในสภาวะที่ไม่มีเซลล์ RM ในโครงข่าย วิธีการดังกล่าวสามารถป้อนกลับเซลล์ BRM ได้ทันทีที่สวิตช์ได้รับเซลล์ BRM ตัวแรก

สำหรับ Wait-for-all algorithm และ Immediate Rate Calculation algorithm นั้นมีผลตอบสนองชั่วครู่ที่ช้ามาก โดยเฉพาะเมื่อไม่มีเซลล์ BRM ในโครงข่าย เช่นในสภาวะเริ่มต้นการส่งข้อมูล ทั้งนี้เนื่องจากว่าวิธี Wait-for-all algorithm ทำให้สวิตช์จำเป็นต้องรอให้เซลล์ BRM เดินทางกลับมาจากปลายทางทุก branch ก่อน ซึ่งหาก branch หนึ่งที่มีเวลาประวิงมาก ก็จะทำให้เวลาที่ใช้ในการรอการตอบกลับนั้นนานตามไปด้วย ส่วนวิธี Immediate Rate Calculation algorithm นั้นจะทำงานเหมือนกับวิธี Wait-for-all algorithm เมื่อโครงข่ายอยู่ในสภาวะโหลดต่ำ ดังนั้นผลตอบสนองชั่วครู่ในสภาวะโหลดต่ำจึงเหมือนกับ Wait-for-all algorithm

6.2.3 ความซับซ้อนในการพัฒนา

วิธีรวบรวมข่าวสารความคับคั่งที่ดีไม่ควรที่จะมีความซับซ้อนมากนัก เพื่อความรวดเร็ว, ง่าย และ ประหยัด ในการพัฒนาสวิตช์ที่สามารถใช้งานได้จริง ความซับซ้อนในการพัฒนาขึ้นอยู่กับปริมาณตัวแปรที่ใช้เก็บข้อมูลสำหรับแต่ละวิธี, การเพิ่มฟังก์ชั่นในการทำงานของสวิตช์ และ ลำดับขั้นตอนในการคำนวณ

ในวิธี Robert algorithm, TS algorithm และ วิธีการที่เสนอแบบที่ 1 นั้นมีความซับซ้อนในการพัฒนาสูง เนื่องจากว่าวิธีการทั้งสองนั้นต้องการให้สวิตช์เพิ่มฟังก์ชั่นในการสร้างเซลล์ BRM เพื่อการตอบกลับขึ้นมา หลักการนี้คล้ายคลึงกับหลักการของ Virtual Source/Virtual Destination (VS/VD) แต่อย่างไรก็ตามก็ถือได้ว่าเป็นค่าเพิ่มความซับซ้อนให้แก่สวิตช์อย่างมาก

วิธี RSS algorithm และ วิธีการที่เสนอแบบที่ 2 มีความซับซ้อนอยู่ในระดับที่น้อยที่สุดโดยสังเกตจากจำนวนปริมาณตัวแปรที่ต้องการใช้ที่เก็บไว้ที่สวิตช์ นั้นมีน้อยที่สุด และมีลำดับขั้นตอนในการทำงานน้อยที่สุดด้วย ในขณะที่ไม่ต้องมีการสร้างเซลล์ BRM เพื่อการตอบกลับ ส่วนวิธี Wait-for-all algorithm นั้นมีความซับซ้อนมากกว่าทั้งสองวิธีข้างต้น เนื่องจากต้องมีการเพิ่มตัวแปรในการตรวจจับการรับเซลล์ BRM จากแต่ละ branch และ ตัวแปรในการเก็บปริมาณเซลล์ BRM ที่รับได้ ในขณะที่เพิ่มลำดับขั้นตอนการทำงานขึ้นมา

Immediate Rate Calculation algorithm มีความซับซ้อนในระดับที่มากกว่า Wait-for-all algorithm พอสมควร จากการเพิ่มเทคนิค Overload detected โดยต้องเพิ่มตัวแปรในการจดจำค่า Last_ER เพื่อนำมาเปรียบเทียบในเทคนิค Overload detected นอกจากนี้ยังเพิ่มขั้นตอนการทำงานขึ้นมา

พอสมควร โดยเฉพาะการคำนวณโดยใช้วิธีควบคุมความคับคั่งทุกครั้งในการรับเซลล์ BRM ทุกตัว โดยไม่คำนวณเฉพาะตอนที่กำลังจะตอบกลับเซลล์ BRM

6.2.4 RM cell Overhead

RM cell Overhead คือ การสร้างเซลล์ BRM ที่เกินความจำเป็นขึ้นมาในโครงข่าย โดยปริมาณเซลล์ FRM ที่แหล่งกำเนิดสร้างขึ้นนั้นควรมีค่าประมาณเท่า ๆ กับ ปริมาณเซลล์ BRM ที่แหล่งกำเนิดได้รับ โดยปริมาณดังกล่าวแหล่งกำเนิดนั้นจะนับปริมาณเซลล์ FRM ที่สร้างขึ้นในขณะที่ยังไม่ได้รับเซลล์ BRM ตอบกลับมาจากเซลล์ FRM ตัวนั้น ซึ่งหากค่าดังกล่าวมีค่ามากกว่าค่า CRM แหล่งกำเนิดจะปรับอัตราส่งให้ลดลงเป็นค่า ACR * CDF ซึ่งหากว่าวิธีรวบรวมข่าวสารความคับคั่งใด มีการผลิตเซลล์ BRM มากกว่าจำนวนเซลล์ FRM ที่ได้รับ อาจเป็นสาเหตุให้เกิดความผิดพลาดในการนับจำนวนดังกล่าว และนำไปสู่ความผิดพลาดในการควบคุมความคับคั่งได้ การตรวจสอบการป้อนกลับเซลล์ BRM ของวิธีรวบรวมข่าวสารความคับคั่งนั้นสามารถแสดงได้จากค่าอัตราส่วนระหว่างเซลล์ FRM ที่รับได้ต่อ BRM ที่สวิตช์สร้างขึ้น

วิธี Robert algorithm และ วิธีการที่เสนอแบบที่ 1 นั้นมีการป้อนกลับเซลล์ BRM ทันทีที่ได้รับเซลล์ FRM ทำให้อัตราส่วนระหว่างเซลล์ FRM ต่อ เซลล์ BRM นั้นมีค่าเป็นหนึ่งเสมอ วิธี TS algorithm นั้นต้องรอให้มีรับเซลล์ BRM ก่อนจึงสามารถป้อนกลับเซลล์ BRM ทันทีที่ได้รับเซลล์ FRM ทำให้อัตราส่งนั้นน้อยกว่า 1 เสมอ เนื่องจากเซลล์ FRM ตัวแรกนั้นจะไม่ได้รับการตอบกลับแน่นอน เนื่องจากยังไม่มีเซลล์ BRM ในโครงข่าย ส่วน RSS algorithm และ วิธีการที่เสนอแบบที่ 2 นั้นจะป้อนกลับเซลล์ BRM ทันทีที่ได้รับเซลล์ BRM ตัวแรกหลังจากได้รับเซลล์ FRM แล้ว ซึ่งถือว่าเป็นเงื่อนไขที่มีการทำงานคล้ายคลึงกับกรณีการส่งแบบหนึ่งจุดไปสู่หนึ่งจุด ซึ่งโดยรวมแล้วจะทำให้อัตราส่วนระหว่างเซลล์ FRM ต่อเซลล์ BRM นั้นมีค่าน้อยกว่าหนึ่งในช่วงที่ได้รับเซลล์ FRM แล้วแต่ยังไม่ได้รับเซลล์ BRM และมีค่าเท่ากับหนึ่งหลังจากป้อนกลับเซลล์ BRM ไปแล้วจนถึงก่อนได้รับเซลล์ FRM ตัวต่อไป

สำหรับ Wait-for-all algorithm นั้นมีอัตราส่วนระหว่างเซลล์ FRM ต่อเซลล์ BRM เป็นค่าน้อยกว่าหรือเท่ากับ 1 เนื่องจากการรอให้ได้รับเซลล์ BRM จากทุก branch ก่อนการป้อนกลับนั่นเอง ส่วนวิธี Immediate Rate Calculation algorithm นั้นหากว่าโครงข่ายอยู่ในสภาวะโหลดเกิน จะมีการป้อนกลับเซลล์ BRM เพิ่มขึ้น ดังนั้นจะทำให้อัตราส่วนระหว่างเซลล์ FRM และ เซลล์ BRM มีค่ามากกว่า 1 ในบางสภาวะ อย่างไรก็ตาม วิธีการดังกล่าวมีการแก้ไขข้อผิดพลาดนี้โดยการนับจำนวนเซลล์ BRM ที่ส่งไปด้วยเทคนิค Overload detected และจะทำการงดการป้อนกลับเซลล์ BRM ในสภาวะปกติ เป็นจำนวนเท่ากับจำนวนเซลล์ BRM ที่ส่งเกินไปนั่นเอง วิธีการนี้สามารถควบคุมอัตราส่วนระหว่างเซลล์ FRM และ เซลล์ BRM ให้มีค่าเป็นน้อยกว่า หรือเท่ากับ 1 ในสภาวะอยู่ตัวได้ อย่างไรก็ตามค่าดังกล่าวอาจมีค่ามากกว่า 1 ในช่วงสภาวะโหลดเกินได้

6.2.5 Change to non responsive branch

สำหรับการส่งจากหนึ่งจุดไปสู่หลายจุดนั้น หากมี branch ใด branch หนึ่งที่เกิดความผิดพลาดและไม่สามารถตอบสนองต่อเซลล์ FRM ที่ส่งไปได้ วิธีการรวบรวมข่าวสารความคับคั่งควรที่สามารถทำงานต่อไปได้โดยปกติ วิธี Robert algorithm, TS algorithm, RSS algorithm, วิธีการที่เสนอแบบที่ 1 และ วิธีการที่เสนอแบบที่ 2 นั้นมีการเปลี่ยนแปลงต่อกรณีที่ branch ใด branch หนึ่งไม่ตอบสนองที่ต่ำมาก เนื่องจากวิธีการดังกล่าวนั้นใช้เงื่อนไขในการรับเซลล์ FRM เป็นตัวตัดสินใจหลักในการป้อนกลับเซลล์ BRM ทำให้หากขาดเซลล์ BRMจาก branch ใด branch หนึ่ง ก็ยังสามารถที่จะทำงานต่อไปได้โดยอาศัยข่าวสารการป้อนกลับจาก branch ที่เหลือแทน

ส่วน วิธี Wait-for-all algorithm และ Immediate Rate Calculation algorithm นั้นจะเกิดปัญหาในกรณีนี้ โดยที่วิธีการทั้งสองมีพื้นฐานในการที่ต้องรอให้มีเซลล์ BRM ตอบกลับมาจากทุก branch ก่อน ซึ่งหากว่าไม่มีเซลล์ BRM ตอบกลับมาจาก branch ใด branch หนึ่งนั้น จะทำให้วิธีทั้งสองนั้นไม่สามารถที่จะป้อนกลับเซลล์ BRM ได้ นอกจากมีวิธีการเพิ่มเติมในการตรวจสอบกรณีดังกล่าวยกตัวอย่างเช่น การจับเวลารอสำหรับทุก branch ซึ่งจะทำให้สวิตช์มีความซับซ้อนเพิ่มขึ้นไปอีก

6.2.6 Change to newly joined branch

ในการใช้งานจริงนั้น ต้องการมีการใช้งานที่สามารถเพิ่มเต็มปลายทางข้ามาใน multicast tree นั้น ดังนั้นวิธีการรวบรวมข่าวสารความคับคั่งที่ดีนั้นควรมีการตอบสนองที่รวดเร็วต่อการเพิ่ม branch ใหม่นี้ โดยวิธี Robert algorithm, TS algorithm, RSS algorithm, วิธีการที่เสนอแบบที่ 1 และ วิธีการที่เสนอแบบที่ 2 ยังคงมีผลตอบสนองต่อเหตุการณ์ดังกล่าวได้เป็นอย่างดี โดยที่สามารถป้อนกลับเซลล์ BRM ได้ เนื่องจากวิธีการดังกล่าวนั้นมีเงื่อนไขในการป้อนกลับเซลล์ BRM โดยใช้เซลล์ FRM ถึงแม้ว่าจะมี branch ใหม่เพิ่มเติมเข้ามา ก็ยังคงสามารถป้อนกลับเซลล์ BRM ไปได้อย่างรวดเร็วเป็นปกติ ในขณะที่ Wait-for-all algorithm และ Immediate Rate Calculation algorithm จะมีการเปลี่ยนแปลงในการทำงานอย่างมากหากมี branch ใหม่เพิ่มเติมเข้ามา เนื่องจากในช่วงเริ่มต้นการเพิ่ม branch ใหม่นั้น สวิตช์ต้องรอให้เซลล์ FRM นั้นเดินทางไปและกลับมาจาก branch ใหม่ก่อน จึงสามารถป้อนกลับเซลล์ BRM ตัวต่อไปได้ ซึ่งหากว่า propagation delay ระหว่างสวิตช์ไปสู่ปลายทางใหม่นั้นมีค่าสูงจะทำให้วิธีทั้งสองนั้นต้องรอการตอบกลับจากปลายทางดังกล่าวเป็นเวลานานตามไปด้วย ซึ่งในช่วงนี้ หากโครงข่ายมีการเปลี่ยนแปลงก็จะทำให้แหล่งกำเนิดไม่สามารถตอบสนองต่อการเปลี่ยนแปลงนี้ได้ เนื่องจากไม่ได้รับเซลล์ BRM จากโครงข่ายนั่นเอง

ตารางที่ 6.2 เปรียบเทียบประสิทธิภาพระหว่างวิธีรวบรวมข่าวสารความคับคั่งต่าง ๆ

Algorithms	Robert	TS	RSS	Wait-for-all	Immediate rate calculate	วิธีการที่เสนอแบบที่ 1	**วิธีการที่เสนอแบบที่ 2**
Consolidation noise	Very High	High	Very High	No	No	No	Low
Overload Transient Response	**Very Fast**	**Fast**	**Fast**	Medium	**Fast**[1]	**Very Fast**	**Fast**
Underload Transient Response	Very Fast	Fast	Fast	Medium	Medium	Very Fast	Fast
Transient Response at initial state with low ICR	**Very Fast**	**Fast**	**Fast**	Slow	Slow	**Very Fast**	**Fast**
Transient Response at initial state with high ICR	Very Fast	Fast	Fast	Slow	Fast	Very Fast	Fast
Implimentation Complexity	High	High	**Low**	Medium	Medium	High	**Low**
Ratio of BRM cells to FRM cells	≥ 1	< 1	≤ 1	≤ 1	Limit = 1[2]	1	≤ 1
Reset stored feedback information at the end of cycle	Yes	Yes	Yes	Yes	Yes	No	No
Feedback information storing method	Per VC	Per VC	Per VC	Per VC	Per VC	Per VC and Per Branch	Per VC
BRM cell returning condition	Wait for FRM	Wait for FRM after BRM received	Wait for BRM after FRM received	Wait for BRM from all branches	Wait for BRM from all branches or Overload detected	Wait for FRM	Wait for FRM after BRM received
Generate BRMs	Yes	No	No	No	No	Yes	No
Change to non responsive branch	**Low**	**Low**	**Low**	High	High	**Low**	**Low**
Change to newly joined branch	Low	Low	Low	High	High	Low	Low

Note

1. Depend on threshold
2. May be > 1 or <1 but =1 in steady state

6.3 ข้อเสนอแนะ

1. การเปรียบเทียบระหว่างวิธีควบคุมความคับคั่งเพียง 2 วิธีคือ ERICA และ FMMRA นั้น สามารถสรุปผลของวิธีการควบคุมความคับคั่งที่มีผลต่อประสิทธิภาพด้าน consolidation noise และ consolidation delay ที่เกิดขึ้นในการส่งจากหนึ่งจุดไปสู่หลายจุดได้ชัดเจนพอสมควร แต่ควรเพิ่มเติมในการทดสอบสำหรับวิธีควบคุมความคับคั่งแบบอื่น ๆ ต่อไป เพื่อสามารถสรุปผลดังกล่าวเพิ่มเติมได้

2. การทำงานร่วมกับระหว่างวิธีรวบรวมข่าวสารความคับคั่งด้วยกันเอง เป็นสิ่งที่ควรได้รับการศึกษา เนื่องจากหากในโครงข่ายประกอบด้วยสวิตช์ที่ใช้วิธีรวบรวมข่าวสารความคับคั่งที่ต่างกัน อาจทำให้เกิดการผิดพลาดหรือมีผลต่อประสิทธิภาพในการส่งข้อมูลอย่างมาก

3. วิธีการรวบรวมความคับคั่งนั้นได้มีการพัฒนาขึ้นมาใหม่ จึงควรมีการศึกษาเพิ่มเติมเพื่อศึกษาให้ครอบคลุมทุกวิธีการต่อไป