สำนักงานวิทยทรัพยากร จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

ในบทนี้จะแบ่งเนื้อหาออกเป็น 2 ส่วน โดยส่วนแรกเป็นทฤษฎีที่เกี่ยวข้องจะอธิบายทฤษฎีที่ใช้ในงานวิจัย ส่วนที่สองเป็นส่วนของงานวิจัยที่เกี่ยวข้องจะอธิบายงานวิจัยที่ข้องเกี่ยวกับการสร้างมัลติคาสต์ทรี

2.1 ทฤษฎีที่เกี่ยวข้อง

ทฤษฎีที่เกี่ยวข้องกับงานวิจัย ประกอบไปด้วย ทฤษฎีสื่อสตรีมมิง แบบจำลองล็อกพี การสื่อสารในเครือข่าย มัลติคาสต์ทรี และงานวิจัยที่เกี่ยวข้อง

2.1.1 สื่อสตรีมมิง (Streaming Media)

สื่อสตรีมมิง (Streaming Media) เป็นเทคโนโลยีการส่งมัลติมีเดียซึ่งประกอบด้วยภาพและเสียงผ่านเครือข่ายอินเตอร์เน็ตไปยังผู้ชมที่อยู่ในเครือข่ายในเวลาเรียลไทม์ (Realtime) ทำให้ผู้ชมสามารถรับชมมัลติมีเดียได้ทันที และสามารถเลื่อนไปยังตำแหน่งที่ต้องการชมได้ โดยไม่ต้องรอให้การดาวน์โหลดเสร็จสิ้น นอกจากนี้ยังประหยัดพื้นที่ในฮาร์ดดิสก์เพราะไฟล์ที่รับชมไม่ได้ถูกจัดเก็บไว้ในเครื่อง ข้อมูลมัลติมีเดียที่ส่งจะมีลักษณะการส่งอย่างต่อเนื่องแบบทยอยส่งเรื่อยๆ (Streaming) การแพร่ภาพหรือกระจายเสียงในลักษณะนี้เกี่ยวพันกับเวลา (Time Based Media) การรับส่งจะต้องกระทำให้พร้อมกันหรือในเวลาใกล้เคียงกันที่สุด แต่อินเทอร์เน็ตเป็นระบบการส่งถ่ายข้อมูลที่ไม่ประสานเวลา (Asynchronous) จึงอาจเกิดปัญหาจิทเตอร์ดีเลย์ ซึ่งเป็นช่วงเวลาดีเลย์ที่แตกต่างกันของแต่ละแพ็กเก็ตในขณะส่งมัลติมีเดียไปยังปลายทาง จิทเตอร์ดีเลย์จึงเป็นเวลาที่เกิดขึ้นตามสภาวะของทราฟฟิกในขณะนั้น การที่แพ็กเก็ตถูกเราเตอร์ส่งแยกกันไปอาจทำให้แพ็กเก็ตถูกส่งไปยังผู้รับด้วยเส้นทางที่ต่างกัน ทำให้เกิดดีเลย์ (Delay) ที่แตกต่างกันไปด้วย ซึ่งอาจส่งผลให้คุณภาพในการแสดงผลมัลติมีเดียแย่ลง หากแบนด์วิดท์ของเครือข่ายมีความเร็วไม่คงที่ อาจจะพบปัญหาการบัฟเฟอร์ (Buffer) ข้อมูลบ่อยหรือนานขึ้นก่อนการแสดงผลและยังทำให้การตอบสนองของแอพพลิเคชันทำงานล่าช้าตามไปด้วย

การส่งข้อมูลมัลติมีเดียแบบสตรีมมิงนิยมใช้โพรโตคอลยูดีพี (User Datagram Protocol: UDP) ซึ่งมีจุดเด่นที่แพ็กเก็ตมีขนาดเล็กและไม่มีการตรวจสอบความถูกต้องของข้อมูล จึงไม่ต้องส่งข้อมูลซ้ำหรือคำนวณอัตราการส่งข้อมูล ทำให้สามารถส่งข้อมูลได้เร็วขึ้นและยังมี

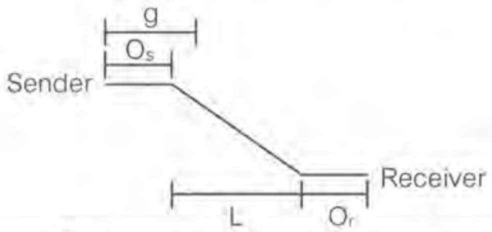
ความเหมาะสมกับการส่งข้อมูลแบบเรียลไทม์ที่ข้อมูลสามารถสูญหายได้ โดยทั่วไปการแสดงผลข้อมูลแบบสตรีมมิงจะต้องมีความต่อเนื่องในการรับข้อมูลเป็นสำคัญ ดังนั้นหากแพ็กเก็ตที่ถูกส่งออกไปก่อน ไปถึงผู้รับหลังจากที่แพ็กเก็ตถัดมาถูกแสดงผลแล้ว แพ็กเก็ตที่มาถึงที่หลังจึงไม่สามารถแสดงผลได้ ซึ่งเทียบแพ็กเก็ตสูญหาย นอกจากโพรโตคอลยูดีพีที่นิยมใช้ในการรับส่งสื่อสตรีมมิงแล้ว ยังมีโพรโตคอลที่ถูกออกแบบมาใช้เฉพาะสำหรับการรับส่งสื่อสตรีมมิง เช่น โพรโตคอลเรียลไทม์สตรีมมิง (Realtime Streaming Protocol: RTSP) เป็นต้น โดยทั่วไปสื่อสตรีมมิงมี 2 ลักษณะด้วยกัน คือ การถ่ายทอดสด และไฟล์ออนดีมานด์

ก) การถ่ายทอดสด (Live Broadcasting) เป็นการถ่ายทอดเหตุการณ์ที่เกิดขึ้น ณ เวลานั้น โดยผู้ชมได้รับชมเหตุการณ์ต่างๆ ได้เป็นปัจจุบันและทันท่วงที ด้วยวิธีการแปลงสัญญาณจากกล้องวิดีโอเป็นข้อมูลดิจิตอล ซึ่งเครื่องเซิร์ฟเวอร์จะทำการถ่ายทอดไปยังเครื่องผู้ชมปลายทาง ผู้ชมสามารถเรียกใช้งานพร้อมๆ กันเป็นจำนวนมากได้ ตัวอย่างเช่น การแข่งขันกีฬา รายงานข่าว รายการทีวี รายการวิทยุ เป็นต้น

ข) ไฟล์ออนดีมานด์ (On-Demand, Video Streaming) เป็นการให้บริการมัลติมีเดียโดยส่งสัญญาณไปยังเครื่องผู้รับ ทำให้ผู้ชมสามารถควบคุมฟังก์ชั่นการทำงานเกี่ยวกับมัลติมีเดียนั้นได้อย่างอิสระ เช่น เริ่มการแสดงผล (Play) หยุดการแสดงผล (Pause) เลือกไปข้าง หน้าอย่างรวดเร็ว (Fast forward) ย้อนกลับ (Rewind) และแสดงผลซ้ำ (Replay) ได้ทันที เพราะไฟล์เหล่านี้ถูกเข้ารหัสในรูปแบบที่เหมาะสมต่อการแสดงผลแบบสตรีมมิง และถูกจัดเก็บไว้ในเซิร์ฟเวอร์ เพื่อรอส่งถ่ายข้อมูลในรูปแบบโปรเกรสซีฟดาวน์โหลด (Progressive Download) ซึ่งเป็นการผสมผสานการส่งข้อมูลแบบสตรีมมิงและการดาวน์โหลดเข้าด้วยกัน โดยใช้บัฟเฟอร์เป็นที่พักข้อมูล นิยมใช้กับไฟล์มัลติมีเดียที่ไม่ใหญ่มากนัก

2.1.2 **แบบจำลองล็อกพี** (LogP Model) [7]

เป็นแบบจำลองที่กำหนดความสัมพันธ์และให้ค่าเวลาในการติดต่อสื่อสารที่ตรงกับที่เกิดขึ้นในการส่งข้อมูลจริง ในแบบจำลองนี้ประกอบด้วยพารามิเตอร์ที่สำคัญ 4 ตัว คือ ลาเทนซี (Latency: L) โอเวอร์เฮด (Overhead: O) แก็ป (Gap: g) และจำนวนโพรเซสเซอร์ (P)



รูปที่ 2.1 เวลาที่ใช้ในการส่งแพ็กเก็ตจากต้นทางไปยังปลายทาง

ลาเทนซีเป็นช่วงเวลาที่แพ็กเก็ตใช้เดินทางจากโหนดหนึ่งไปอีกโหนดหนึ่ง ซึ่งเป็นช่วงเวลาตั้งแต่แพ็กเก็ตถูกส่งออกจากผู้ส่งจนกระทั่งไปถึงผู้รับ โอเวอร์เฮดเป็นช่วงเวลาที่หน่วยประมวลผลใช้ในการส่งและรับข้อมูล ซึ่งเป็นซอฟต์แวร์โอเวอร์เฮดเท่านั้น และโอเวอร์เฮดนี้จะเกิดขึ้นทั้งฝั่งผู้ส่งและผู้รับ โอเวอร์เฮดที่ฝั่งผู้ส่งจะเป็นเวลาตั้งแต่เริ่มส่งแพ็กเก็ตจนกระทั่งแพ็กเก็ตถูกส่งออกไปในเครือข่าย ส่วนโอเวอร์เฮดที่ฝั่งผู้รับจะเป็นเวลาที่ตั้งแต่แพ็กเก็ตมาถึงจนกระทั่งได้เป็นข้อมูลที่สามารถส่งให้หน่วยประมวลผลได้ แก็ปเป็นระยะเวลาห่างระหว่างแพ็กเก็ตซึ่งเกิดขึ้นตั้งแต่ส่งแพ็กเก็ตออกไปจนกระทั่งผู้ส่งสามารถส่งแพ็กเก็ตถัดไปได้อีกครั้ง จำนวนโพรเซสเซอร์แทนจำนวนโพรเซสเซอร์และจำนวนเมมโมรีโมดูล จากรูปที่ 2.1 แสดงให้เห็นว่าเวลาทั้งที่ใช้ส่งแพ็กเก็ตจากโหนดหนึ่งไปอีกโหนดหนึ่ง (End-to-End Delay) แทนด้วย t_{sr} ประกอบ ด้วยเวลาที่โหนดผู้ส่งใช้ส่งแพ็กเก็ต (o_s) เวลาที่แพ็กเก็ตเดินทางในเครือข่าย (L) และเวลาโหนดผู้รับได้รับข้อมูลเรียบร้อยแล้ว (o_r) ดังสมการ

$$t_{sr} = o_s + L + o_r \tag{2.1}$$

2.1.3 การสื่อสารในเครือข่าย

การสื่อสารในเครือข่ายแบ่งได้ 3 แบบ คือ ยูนิคาสต์ บรอดคาสต์ และมัลติคาสต์

ก) การส่งแบบยูนิคาสต์ (Unicast) เป็นการส่งแบบเจาะจงผู้รับในลักษณะจุดต่อจุด (Point-to-Point) การส่งแต่ละครั้งจึงมีผู้ส่ง 1 โหนดและมีผู้รับ 1 โหนด หากมีจำนวนผู้รับมากกว่าหนึ่งโหนด ผู้ส่งจะต้องส่งข้อมูลเดิมซ้ำตามจำนวนผู้รับที่เพิ่มขึ้น (Multiple unicast) แม้ว่าข้อมูลนั้นจะถูกส่งออกไปในเส้นทางเดียวกันก็ตาม ในกรณีนี้ทำให้เกิดการสูญเสียแบนด์วิดท์ตามปริมาณข้อมูลที่ส่งออกไป

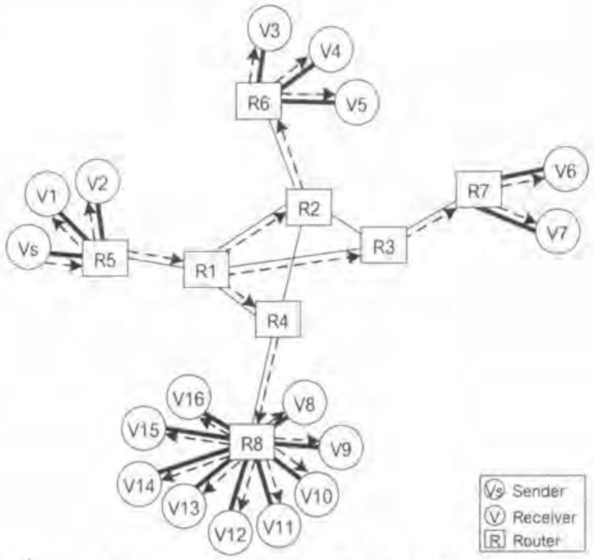
ข) การส่งแบบบรอดคาสต์ (Broadcast) เป็นการส่งข้อมูลเพียงครั้งเดียวไปยังทุกเครื่องที่ต่อเชื่อมอยู่ในเครือข่ายพร้อมๆ กัน โดยอาศัยอุปกรณ์ในเครือข่ายช่วยทำสำเนาและกระจายข้อมูลออกไป

ค) การส่งแบบมัลติคาสต์ (Multicast) เป็นการส่งแบบเจาะจงผู้รับเฉพาะกลุ่มที่ได้ทำการติดต่อหรือเชื่อมโยงไว้ เป็นการส่งที่มีผู้ส่ง 1 โหนด และมีผู้รับเป้าหมายจำนวน N โหนด การส่งข้อมูลแบบมัลติคาสต์มี 2 ประเภท คือ ไอพีมัลติคาสต์ และมัลติคาสต์ระดับชั้นแอพพลิเคชัน

1) ไอพีมัลติคาสต์ (IP Multicast) เป็นการติดต่อสื่อสารในระดับชั้นเครือข่าย (Network Level) โหนดในไอพีมัลติคาสต์ หมายถึง โหนดผู้ส่ง-ผู้รับและตัวกลางในการส่งข้อมูล เช่น เราเตอร์ เป็นต้น การกระจายข้อมูลด้วยไอพีมัลติคาสต์จำเป็นต้องใช้เราเตอร์ที่เข้าใจ

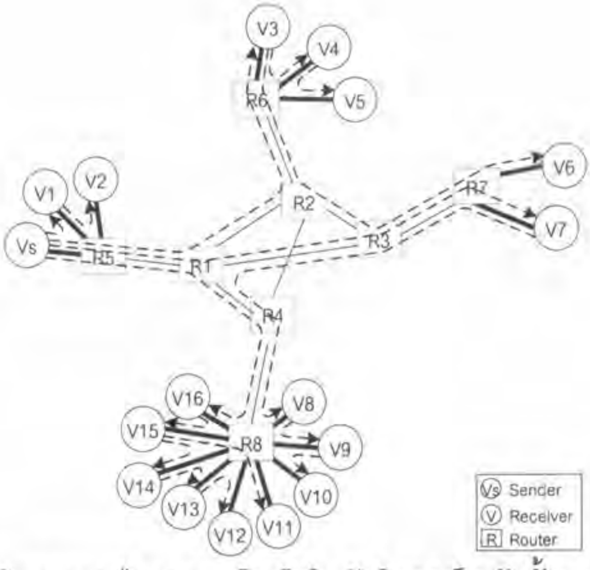
โพรโตคอลมัลติคาสต์ (Multicast Protocol) การส่งข้อมูลที่อาศัยเราเตอร์เป็นตัวกลางเป็นการส่งข้อมูลจะใช้แบนด์วิดท์ได้อย่างมีประสิทธิภาพ เพราะสามารถลดโหลดในการส่งข้อมูลที่โหนดผู้ส่งและสามารถลดปริมาณทราฟฟิกในเครือข่ายได้ นอกจากนี้การส่งในลักษณะนี้ยังช่วยให้ผู้ส่งสามารถส่งข้อมูลเพียงครั้งเดียวไปยังโหนดผู้รับหลายโหนดในเวลาใกล้เคียงกันได้ ไอพีมัลติคาสต์ถูกใช้กับแอพพลิเคชันต่างๆ มากมาย เช่น การประชุมทางจอภาพ (Video Conferencing) การสื่อสารในองค์กร (Corporate Communications) การเรียนทางไกล (Distance Learning) การกระจายการใช้งานซอฟต์แวร์ (Distribution of Software) การดูหุ้นแบบออนไลน์ และการส่งข่าวสาร เป็นต้น

2) มัลติคาสต์ระดับชั้นแอพพลิเคชัน (Application-Level Multicast) เป็นการส่งข้อมูลโดยให้โหนดต้นทาง (Source) และโหนดผู้รับเป็นผู้ส่งข้อมูลไปยังโหนดผู้รับอื่นๆ ภายในกลุ่มมัลติคาสต์ (Multicast Group) ด้วยวิธีการส่งแบบยูนิคาสต์ การส่งแบบนี้ไม่จำเป็นต้องใช้เราเตอร์เป็นตัวกลางในการกระจายข้อมูล เราเตอร์ในเครือข่ายจึงไม่จำเป็นต้องถูกแก้ไขเพิ่มเติมความสามารถให้รองรับโพรโตคอลมัลติคาสต์



รูปที่ 2.2 เส้นทางการไหลของแพ็กเก็ตในไอพีมัลติคาสต์

รูปที่ 2.2 และรูปที่ 2.3 เปรียบเทียบเส้นทางการไหลของแพ็กเก็ตในไอพีมัลติคาสต์และมัลติคาสต์ระดับชั้นแอพพลิเคชัน จากรูปที่ 2.2 พบว่าตลอดเส้นทางการส่งข้อมูลจะต้องอาศัยเราเตอร์ทำสำเนาและส่งแพ็กเก็ตต่อไปยังโหนดต่างๆ ที่อยู่ในกลุ่มเป้าหมาย

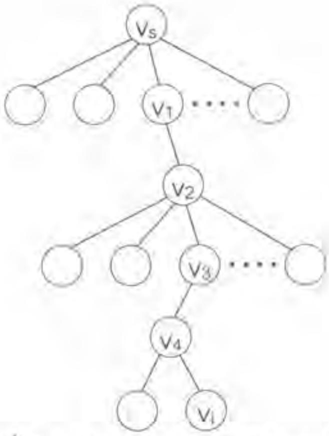


รูปที่ 2.3 เส้นทางการไหลของแพ็กเก็ตในมัลติคาสต์ระดับชั้นแอพพลิเคชัน

ในขณะที่รูปที่ 2.3 แสดงให้เห็นถึงการกระจายข้อมูลที่เกิดจากการส่งของโหนดภายในเครือข่ายด้วยการสื่อสารแบบยูนิคาสต์ที่ใช้ลิงค์กายภาพร่วมกัน ทำให้เกิดการส่งข้อมูลเดิมซ้ำหลายครั้งเพื่อให้ข้อมูลนั้นไปถึงผู้รับแต่ละโหนด ด้วยวิธีการส่งข้อมูลแบบนี้จะมีข้อเสียในเรื่องการใช้แบนด์วิดท์ และยังเป็นสาเหตุหนึ่งที่ทำให้เกิดข้อมูลซ้ำในเครือข่าย เพราะมีแพ็กเก็ตเดียวกันถูกส่งผ่านลิงค์เดียวกันมากกว่าหนึ่งครั้ง ตัวอย่างเช่น การส่งจาก v_s ไปยัง v_3 และ v_6 ซึ่งเป็นการส่งข้อมูลเดียวกันแต่แยกการส่งเป็นสองครั้ง และเมื่อพิจารณาลิงค์ (R_5, R_1) จะพบข้อมูลซ้ำซ้อนเกิดขึ้น เป็นต้น ลักษณะเช่นนี้ เรียกว่า แพ็กเก็ตซ้ำ หรือลิงค์สเตรส (Link Stress)

2.1.4 **มัลติคาสต์ทรี** (Multicast tree)

การติดต่อสื่อสารแบบมัลติคาสต์จำเป็นต้องสร้างมัลติคาสต์ทรี เพื่อกำหนดเส้นทางการส่งข้อมูลจากผู้ส่งไปยังผู้รับหรือที่เรียกว่า พาธ (Path) ซึ่งเป็นลำดับของโหนดเริ่มตั้งแต่โหนดต้นทางไปจนกระทั่งถึงโหนดผู้รับปลายทาง ในมัลติคาสต์ทรี โหนดต้นทางจะทำหน้าที่เป็นโหนดรูต (Root Mode) และโหนดอื่นที่เหลือทั้งหมดจะเป็นโหนดผู้รับ ข้อมูลจะถูกส่งจากโหนดรูตไปหาโหนดต่างๆ ซึ่งเป็นการส่งจากบนลงล่าง (Top Down) จากตัวอย่างมัลติคาสต์ทรีในรูปที่ 2.4 เส้นทางการส่งข้อมูลหรือพาธระหว่าง v_s ไปยัง v_i คือ $\{v_s, v_1, v_2, v_3, \ldots, v_i\}$



รูปที่ 2.4 ตัวอย่างของมัลติคาสต์ทรี

มัลติคาสต์ทรีที่สร้างขึ้นมี 2 แบบ คือ สร้างขึ้นเพื่อใช้เฉพาะเจาะจงต้นทางเพียงโหนดเดียว (Source-Based Tree) และสร้างขึ้นเพื่อให้โหนดต้นทางหลายโหนดสามารถใช้ทรีร่วมกัน (Group Shared Tree) โดยมัลติคาสต์ทรีที่ถูกสร้างขึ้นมาเฉพาะสำหรับผู้ส่งโหนดใดๆ จะเป็นทรีที่มีประสิทธิภาพเหมาะสำหรับผู้ส่งนั้นมากกว่ามัลติคาสต์ทรีที่ถูกสร้างขึ้นมาเพื่อใช้ร่วมกัน ตัวอย่างของอัลกอริทึมที่นิยมนำมาใช้สร้างมัลติคาสต์ทรี ได้แก่ ทรีที่มีทางเดินสั้นที่สุด (Shortest Path Tree: SPT) และทรีแบบทอดข้าม (Minimum Spanning Tree: MST) โดยทรีที่มีทางเดินสั้นที่สุดจะเป็นมัลติคาสต์ทรีที่ถูกสร้างขึ้นเพื่อใช้เฉพาะกับผู้ส่งโหนดใดโหนดหนึ่งเท่านั้น หากมีผู้ส่งหลายโหนดจะต้องสร้างมัลติคาสต์ทรีตามจำนวนโหนดผู้ส่ง ทำให้เสียทรัพยากรในการสร้างทรีเป็นจำนวนมาก แต่จะได้ประสิทธิภาพในการกระจายข้อมูลที่ดี อัลกอริทึมที่ใช้สร้างมัลติคาสต์ทรีที่มีทางเดินสั้นที่สุดถูกใช้มากทั้งในไอพีมัลติคาสต์และมัลติคาสต์ระดับชั้นแอพพลิเคชัน เพราะเป็นอัลกอริทึมที่หาเส้นทางที่ดีที่สุดจากโหนดรูตไปยังโหนดต่างๆ โดยพิจารณาจากเวลาที่น้อยที่สุดที่ใช้ส่งข้อมูลจากโหนดรูตไปยังโหนดนั้นๆ มัลติคาสต์ทรีชนิดนี้สามารถลดเวลาที่ใช้ส่งข้อมูลหรือดีเลย์ได้มาก ซึ่งตรงข้ามกับมัลติคาสต์ทรีแบบทอดข้ามที่ไม่ได้สร้างขึ้นเฉพาะเจาะจงกับโหนดรูตใด ทรีแบบทอดข้ามจึงมีความยืดหยุ่นมากกว่าเพราะสามารถใช้ได้กับทุกโหนด ยกเว้นกรณีที่โหนดในเครือข่ายมีการเปลี่ยนแปลง มัลติคาสต์ทรีแบบทอดข้ามจะถูกสร้างขึ้นอีกครั้ง

2.2 งานวิจัยที่เกี่ยวข้อง

งานวิจัยที่นำเสนอการสร้างเครือข่ายซ้อนทับสำหรับการมัลติคาสต์ระดับชั้นแอพพลิเคชันสามารถจัด แบ่งได้สองประเภท คือ แบบทรี (Tree-Based) และแบบเมช (Mesh-Based) ซึ่งงานวิจัยที่ใช้เมชเป็นพื้นฐานส่วนใหญ่เป็นการสร้างมัลติคาสต์ทรีที่ใช้งานร่วมกัน ตัวอย่างงานวิจัยที่ใช้โครงสร้างพื้นฐานแบบเมช เช่น นาราดา (Narada) [8], ยอยด์ (Yiod)[9], บาเยอ

(Bayeux) [10], สไคบ์ (Scribe) [11], การหาพื้นที่ขนาดเล็ก (Delaunay triangulation) [12], โดเน็ต/คูลสตรีมมิง (DONet/CoolStreaming) [13], มิวชวลคาสต์ (Mutualcast) [14], ไอโอโอ (IOO) [2] และเอ็มเอ็สเอ็มที/เอ็มบีเอ็มที [5] เป็นต้น

โดเน็ต/คูลสตรีมมิง [13] นำเสนออัลกอริทึมที่ใช้สร้างมัลติคาสต์ทรีโดยใช้โครงสร้างพื้นฐานแบบเมชและใช้การตรวจสอบแบนด์วิดท์ของลิงค์ในปรับปรุงมัลติคาสต์ทรีให้ใช้เวลาในการส่งข้อมูลน้อยลง ซึ่งช่วยลดการกระโดดในแสดงผล (Skipping During Playback) ระหว่างให้บริการสตรีมมิง ทำให้การให้สามารถให้บริการสตรีมมิงแอพพลิเคชันที่ต้องการแบนด์วิดท์สูงถึงปานกลางได้ด้วยโอเวอร์เฮดที่ต่ำลง

ไอโอโอนำ [2] เสนออัลกอริทึม ที่ช่วยเพิ่มประสิทธิภาพการสื่อสารระหว่างเครือข่ายซ้อนทับด้วยการสร้างเส้นทางที่มีประสิทธิภาพ เพื่อเชื่อมเครือข่ายซ้อนทับเข้าด้วยกันและปรับปรุงการใช้ทรัพยากรในเครือข่ายให้มีประสิทธิภาพยิ่งขึ้น โครงสร้างของอัลกอริทึมนี้จะมีการปรับสมดุลของโหลดระหว่างกลุ่มโหนด โดยใช้การต่อเชื่อมกับโหนดที่อยู่ใกล้ที่สุดเพื่อใช้ดีเลย์และทรัพยากรในเครือข่ายโลคอล และใช้บันทึกการใช้งาน (Log Data) ที่พัฒนามาจากโครงสร้างของเอนีซี [15] มาวัดผลและวิเคราะห์ประสิทธิภาพเทียบ

เอ็มเอสเอ็มทีและเอ็มบีเอ็มที [5] เป็นอัลกอริทึมที่สร้างมัลติคาสต์ทรีโดยมีจุดประสงค์เพื่อต้องการเพิ่มแบนด์วิดท์ด้วยการตัดลิงค์ที่มีการใช้แบนด์วิดท์สูงออกและแทนที่ลิงค์นั้นด้วยลิงค์ที่มีการใช้แบนด์วิดท์น้อยกว่า ทั้งสองอัลกอริทึมจะมีกระบวนการสร้างมัลติคาสต์ทรีที่คล้ายคลึงกัน คือการสร้างทรีแบบทอดข้ามจากค่าดีเลย์ แล้วจึงทำการปรับปรุงมัลติคาสต์ทรีด้วยการคัดลิงค์ที่เป็นคอขวดออก ซึ่งความแตกต่างกันของทั้งสองอัลกอริทึมจะอยู่ในกระบวนการปรับปรุงมัลติคาสต์ทรี ซึ่งเกิดจากข้อมูลอินพุตที่นำมาใช้สร้างมัลติคาสต์ทรีแตกต่างกัน โดยเอ็มเอสเอ็มทีเป็นอัลกอริทึมที่ไม่ต้องทราบแบนด์วิดท์ในขณะปรับปรุงมัลติคาสต์ทรี การปรับปรุงมัลติคาสต์ทรีจะทำโดยการตัดลิงค์ที่มีจำนวนแพ็กเก็ตซ้ำสูงสุดออกแล้วเลือกลิงค์ที่มีจำนวนแพ็กเก็ตซ้ำต่ำกว่าเข้ามาแทน ส่วนเอ็มบีเอ็มทีเป็นอัลกอริทึมที่จำเป็นต้องทราบข้อมูลแบนด์วิดท์ เนื่องจากการปรับปรุงมัลติคาสต์ทรีจะทำโดยการตัดลิงค์ที่มีอัตราการส่งข้อมูลสูงสุดออกแล้วแทนที่ด้วยลิงค์ที่มีอัตราการส่งข้อมูลที่ต่ำกว่าโดยไม่เกิดลูป (Loop) ในการส่งข้อมูลทราบและไม่ทราบข้อมูลแบนด์วิดท์ ถึงแม้ว่าอัลกอริทึมทั้งสองนี้จะสามารถสร้างมัลติคาสต์ทรีที่ลดการใช้แบนด์วิดท์ลงได้ แต่อย่างไรก็ตามอัลกอริทึมทั้งสองนี้ก็มีข้อด้อยตรงกระบวนการสร้างทรีแบบทอดข้ามที่จะต้อง

ทราบข้อมูลลิงค์ในเครือข่ายทั้งหมดเสียก่อน ซึ่งเป็นเรื่องยาก จึงทำให้อัลกอริทึมทั้งสองไม่สามารถนำไปใช้จริงได้

ส่วนตัวอย่างงานวิจัยที่มีโครงสร้างแบบทรี เช่น โอเวอร์คาสต์ (Overcast) [16], ไนซ์ [6], ซิกแซก [1], สปริตสตรีม (SplitStream) [17], เอนีซี (Anysee) [15], เอสเอ็ชเอ็ม (SHM) [4], สแต็จ (STAG) [3], คูปเน็ต (Coopnet) [18] เป็นต้น ซึ่งงานวิจัยเหล่านี้มีบางงานเป็นการสร้างทรีโดยคำนึงถึงข้อมูลโครงข่าย (Topology-Aware) เช่น แบนด์วิดท์ โหนดที่อยู่ในตำแหน่งใกล้เคียง และข้อมูลตำแหน่งที่ตั้ง (Location Information) เป็นต้น ตัวอย่างงานวิจัยที่คำนึงถึงข้อมูลโครงข่าย เช่น โอเวอร์คาสต์ [16] ซึ่งเป็นอัลกอริทึมที่นำเสนอการสร้างมัลติคาสต์ทรีที่รองรับแอพพลิเคชันที่มีผู้ส่งเดียว โดยมีจุดประสงค์เพื่อให้มีแบนด์วิดท์ที่โหนดรูตและโหนดผู้ส่งแต่ละโหนดให้มากที่สุด ในการเข้าเป็นสมาชิกของมัลติคาสต์ทรี โหนดใหม่แต่ละโหนดจะเชื่อมต่อกับโหนดแม่ซึ่งทำหน้าที่เป็นเสมือนโหนดพร็อกซี่ที่เชื่อมต่อกันไปเป็นลำดับชั้นจากโหนดหนึ่งถึงโหนดหนึ่งด้วยการพิจารณาจากค่าแบนด์วิดท์ที่มากที่สุด โดยไม่คำนึงถึงดีเลย์จึงทำให้มัลติคาสต์ทรีที่ได้มีดีเลย์มากเกินไป

คูปเน็ต [18] นำเสนออัลกอริทึมที่ใช้สร้างมัลติคาสต์ทรี โดยมีโหนดรูตเป็นโหนดผู้ส่งซึ่งเป็นโหนดที่เก็บรวบรวมข้อมูลทั้งหมดในการสร้างและปรับแต่งมัลติคาสต์ทรี จึงมีลักษณะเป็นแบบรวมศูนย์ซึ่งจะมีประสิทธิภาพมากในการสร้างและการส่งข้อมูลในมัลติคาสต์ทรี แต่โครงสร้างแบบรวมศูนย์นี้จำเป็นต้องอาศัยประสิทธิภาพของโหนดรูตเพียงอย่างเดียว

ไนซ์ [6] นำเสนอมัลติคาสต์ทรีที่มีลักษณะเป็นโครงสร้างลำดับชั้นของคลัสเตอร์ (Cluster Hierarchy) แต่ละคลัสเตอร์ประกอบด้วยโหนดหลายโหนด ไนซ์เป็นโครงสร้างแบบกระจายศูนย์ จึงสามารถลดโหลดในการส่งข้อมูลของโหนดรูตลงได้ ในมัลติคาสต์ทรีชนิดนี้โหนดรูตหรือโหนดผู้ส่งจะเป็นโหนดที่อยู่ชั้นบนสุดของมัลติคาสต์ทรี ส่วนโหนดที่เหลือทุกโหนดจะถูกจัดอยู่ชั้นเดียวกันคือชั้นล่างสุดของมัลติคาสต์ทรี ทำให้ทุกโหนดในมัลติคาสต์ทรีมีจำนวนเน็ตเวิร์คฮ็อปจากโหนดผู้ส่งเท่ากัน ไนซ์ถูกออกแบบมาให้รองรับเครือข่ายที่มีแบนด์วิดท์จำกัดและให้ใช้ดีเลย์ในการส่งข้อมูลในเครือข่ายขนาดใหญ่ลดลง แต่ละคลัสเตอร์จึงถูกจำกัดจำนวนสมาชิก เพื่อป้องกันการเกิดปัญหาคอขวดที่โหนดที่เป็นผู้กระจายข้อมูล การจัดกลุ่มโหนดนี้ช่วยลดโอเวอร์เฮด (Control Overhead) ที่เกิดขึ้นในขณะที่โหนดใหม่เข้าเป็นสมาชิกของมัลติคาสต์ทรี และการกระจายข้อมูลในแต่ละคลัสเตอร์ไม่ขึ้นต่อกัน แต่ละคลัสเตอร์จึงสามารถกระจายข้อมูลได้พร้อมกัน

ซิกแซก [1] เป็นงานวิจัยที่นำเสนอการสร้างมัลติคาสต์ทรีที่มีลักษณะเป็นโครงสร้างลำดับชั้นที่พัฒนามาจากโครงสร้างมัลติคาสต์ทรีของไนซ์ โดยแต่ละคลัสเตอร์จะเพิ่มโหนดกระจายข้อมูลเพื่อให้ทำหน้าที่ส่งข้อมูลให้โหนดต่างๆ ภายในคลัสเตอร์แทนที่โหนดหัวหน้า ที่แต่เดิมเป็นโหนดที่ทำหน้าที่กระจายข้อมูลให้โหนดภายในคลัสเตอร์ การเพิ่มโหนดกระจายข้อมูลช่วยลดแบนด์วิดท์โอเวอร์เฮดที่โหนดผู้ส่งและลดโอเวอร์เฮดในการกระจายการส่งข้อมูล นอกจากนี้ยังช่วยลดโอเวอร์เฮดของโหนดใหม่ในการเข้าร่วมเป็นสมาชิกของมัลติคาสต์ทรี รวมทั้งยังลดโอเวอร์เฮดในการออกจากทรีเช่นเดียวกัน และถึงแม้ว่าโครงสร้างของซิกแซกจะมีประสิทธิภาพในการกระจายข้อมูลในกลุ่มย่อยๆ แต่โครงสร้างของซิกแซกมีประสิทธิภาพเฉพาะสำหรับกับเครือข่ายที่ทุกโหนดมีดีเลย์และแบนด์วิดท์ใกล้กัน จึงทำให้โครงสร้างของซิกแซกไม่เหมาะสมในการนำไปใช้จริงกับเครือข่ายอินเตอร์เน็ตในปัจจุบันที่มีความแตกต่างระหว่างแบนด์วิดท์อย่างชัดเจน

สแต็จ [3] นำเสนอการสร้างมัลติคาสต์ทรีที่คำนึงถึงโครงสร้างการเชื่อมต่อด้วยการค้นหาเส้นทาง (Path Finding) โดยใช้เทรซรูท [13] เป็นเครื่องมือค้นหาโหนดที่ใกล้ที่สุด เพื่อให้เป็นโหนดแม่สำหรับโหนดใหม่ที่จะเข้าเป็นสมาชิกของมัลติคาสต์ทรี แต่อัลกอริทึมนี้ไม่ช่วยทำให้เกิดการใช้แบนด์วิดท์อย่างมีประสิทธิภาพ เพราะการสร้างมัลติคาสต์พิจารณาโหนดที่อยู่ใกล้ที่สุดโดยไม่ได้พิจารณาถึงแบนด์วิดท์ของลิงค์นั้น ทำให้มัลติคาสต์ทรีที่ได้ไม่สามารถใช้แบนด์วิดท์ของเครือข่ายได้อย่างมีประสิทธิภาพเท่าที่ควร

จากการศึกษาวิจัยที่นำเสนอการสร้างมัลติคาสต์ทรีที่รองรับการทำมัลติคาสต์ระดับชั้นแอพพลิเคชันพบว่า งานวิจัยส่วนใหญ่เน้นการลดดีเลย์ในการส่งข้อมูล (Delay Optimization) แต่มีงานวิจัยหลายชิ้นที่นำเสนอการสร้างมัลติคาสต์ทรีที่มีจุดประสงค์เพื่อใช้แบนด์วิดท์อย่างมีประสิทธิภาพโดยลดการใช้แบนด์วิดท์ที่ลิงค์คอขวด แต่อัลกอริทึมเหล่านั้นยังไม่สามารถใช้ประสิทธิภาพเครือข่ายได้อย่างเต็มที่ เนื่องจากกระบวนการสร้างมัลติคาสต์ทรีพิจารณาเฉพาะแบนด์วิดท์ที่มีให้หรือแบนด์วิดท์ที่ใช้ไปเพียงค่าใดค่าเดียว และบางอัลกอริทึมจำเป็นจะต้องทราบข้อมูลทั้งเครือข่ายเสียก่อนจึงจะสามารถสร้างมัลติคาสต์ทรีได้ ซึ่งการเก็บข้อมูลทั้งเครือข่ายก่อนการสร้างมัลติคาสต์ทรีเป็นเรื่องที่ทำได้ยาก จึงทำให้มัลติคาสต์ทรีในเหล่านี้รองรับผู้ใช้ได้จำนวนจำกัดและไม่สามารถใช้แบนด์วิดท์ที่มีอยู่ได้อย่างมีประสิทธิภาพ