

## ระบบ MC-CDMA ที่ประยุกต์ใช้อัลกอริทึม V-BLAST และอัลกอริทึมที่ถูกปรับปรุงเพื่อลดความซับซ้อน

ในบทนี้ได้กล่าวถึงรายละเอียดของการนำอัลกอริทึม V-BLAST มาประยุกต์ใช้ร่วมกับระบบ MC-CDMA ซึ่งได้นำเสนอขึ้นเพื่อเพิ่มความจุและอัตราเร็วในการรับส่งข้อมูลของระบบ MC-CDMA ที่มีอยู่เดิมและนอกจากนี้ เนื้อหาในบทนี้จะได้อธิบายถึง อัลกอริทึมที่ถูกนำเสนอขึ้นเพื่อลดความซับซ้อนของอัลกอริทึม V-BLAST ที่มีอยู่เดิม โดยได้นำเสนอทั้งสิ้น 3 วิธี ได้แก่ วิธีการละทิ้งผลของ MAI วิธีการหักล้างสัญญาณแบบผสม และวิธีการหักล้างสัญญาณแบบผสมที่มีการใช้วิธีการ multiple slicing ร่วมด้วย

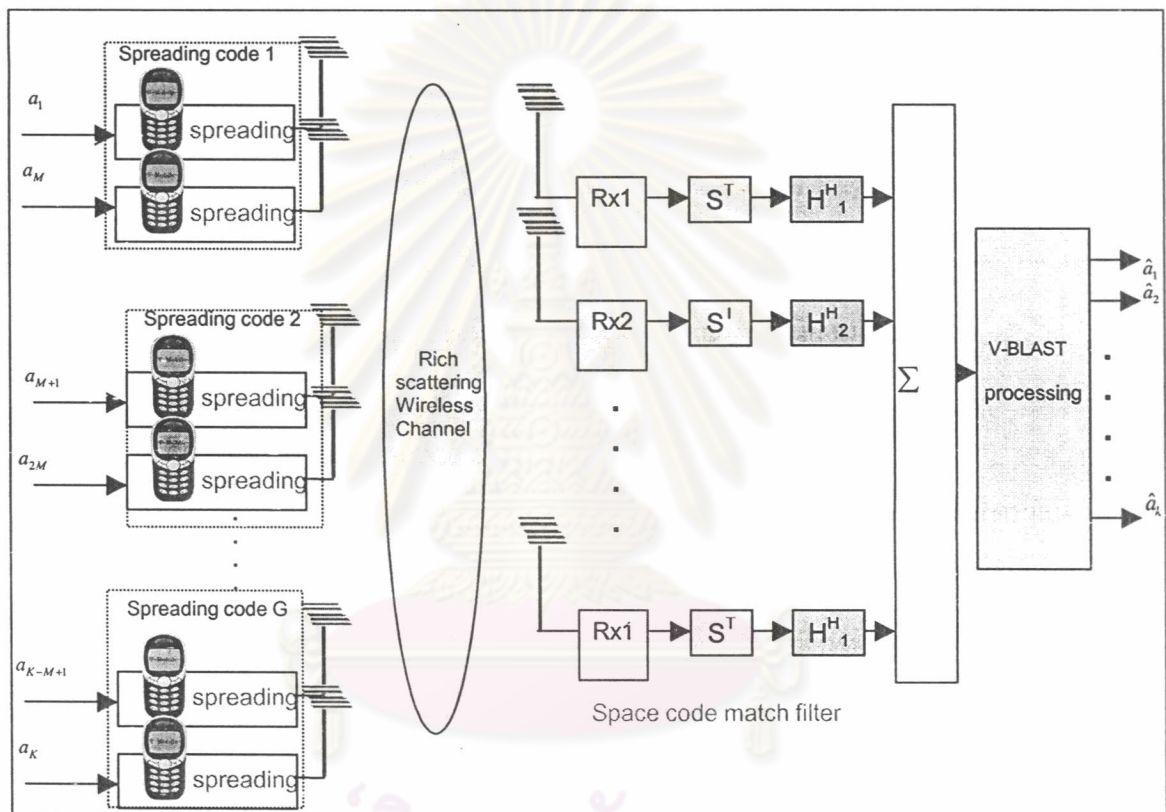
### 3.1. การดัดแปลงสถาปัตยกรรม V-BLAST เพื่อใช้ในระบบโทรศัพท์ไร้สาย MC-CDMA

เพื่อให้ได้มาซึ่งความสามารถในการรับส่งข้อมูลด้วยอัตราที่สูงมากขึ้นสำหรับการสื่อสารของโทรศัพท์ไร้สายในอนาคต วิธีการที่น่าสนใจคือการรวมวิธีการรับส่งของระบบ MC-CDMA เข้ากับสถาปัตยกรรม V-BLAST ด้วยเหตุผลที่ทำให้การส่งข้อมูลกระทำได้เร็วยิ่งขึ้น เนื่องจากผู้ใช้โทรศัพท์ไร้สายสามารถที่จะใช้รหัสแม่ที่ซ้ำกันได้ เป็นผลให้ไม่จำเป็นต้องใช้รหัสแม่ที่มีความยาวสูง

วิทยานิพนธ์นี้จะเลือกพิจารณาเฉพาะการส่งข้อมูลทางข่ายเชื่อมโยงขาขึ้น (up link) เท่านั้น โดยการส่งข้อมูลของสายอากาศแต่ละตัวในทางฝั่งส่งข้อมูลของระบบ V-BLAST ที่ได้กล่าวมาข้างต้นจะถูกแทนที่ด้วยการส่งข้อมูลจากผู้ใช้บริการโทรศัพท์ไร้สายแต่ละคน ส่วนทางฝั่งรับหรือสถานีฐานจะประกอบไปด้วยสายอากาศหลายตัวเช่นกัน ทั้งนี้ผู้ใช้บริการโทรศัพท์ไร้สายแต่ละคนสามารถที่จะส่งข้อมูลโดยใช้รหัสแม่ที่เหมือนกันได้ ซึ่งในระบบ MC-CDMA นั้นจะมีคุณสมบัติของโดเวอร์ซิติทิวตี้ ทำให้ข้อมูลในแต่ละชิพของผู้ใช้แต่ละคนจะถูกลดทอนโดยเฟดดิ้งของช่องสัญญาณที่แตกต่างกัน และจะอาศัยความแตกต่างนี้เพื่อมาใช้ในการแยกแยะข้อมูลของผู้ใช้ที่ใช้รหัสแม่รหัสเดียวกันออกมา ส่วนข้อมูลจากผู้ใช้ที่ใช้รหัสแม่ที่แตกต่างกันก็จะสามารถแยกแยะออกจากกันได้โดยอาศัยคุณสมบัติตั้งฉากของชุดรหัสแม่ตามหลักการของระบบ CDMA ทั่วไป

ในการใช้ระบบ MC-CDMA ร่วมกับวิธี V-BLAST นี้ จะมีสัญญาณแทรกสอดที่เราต้องพิจารณา 2 ประเภท ที่ส่งผลต่อสมรรถนะของระบบ ประเภทแรก คือ สัญญาณแทรกสอด

ระหว่างผู้ใช้ที่ใช้อรหัสแม่ที่ต่างกัน (MAI) โดยจะเกิดจากการที่ชุดรหัสแม่สูญเสียสมบัติความตั้งฉากไปเมื่อผ่านช่องสัญญาณ ซึ่งสัญญาณแทรกสอดประเภทนี้จะเกิดขึ้นในระบบ CDMA ทุกระบบอยู่แล้ว เครื่องรับแมตซ์ฟิลเตอร์ที่มีการใช้วิธีการปรับเท่าแบบ maximum ratio combining (MRC) จะถูกนำมาใช้ในการลดผลของ MAI นี้ ส่วนสัญญาณแทรกสอดอีกประเภทคือ สัญญาณแทรกสอดระหว่างผู้ใช้ที่ใช้อรหัสแม่ที่เดียวกัน (CAI) ซึ่งเกิดจากสายข้อมูลที่ถูกส่งจากผู้ใช้รายอื่นที่ใช้อรหัสแม่เหมือนกัน ซึ่งอัลกอริทึม V-BLAST จะถูกนำมาใช้ในการหักล้างสัญญาณแทรกสอดประเภทนี้



รูปที่ 3-1 แบบจำลองของระบบ MC-CDMA ที่ใช้ร่วมกับสถาปัตยกรรม V-BLAST

แบบจำลองการรับส่งข้อมูลจะเป็นดังรูปที่ 3-1 โดยจะพิจารณาการส่งข้อมูลทางสายเชื่อมโยงขาขึ้น จากผู้ใช้บริการโทรศัพท์ที่ไร้สายจำนวน  $K$  คน ไปยังสถานีฐานที่ประกอบไปด้วยกลุ่มสายอากาศจำนวน  $P$  ตัว โดยผู้ใช้บริการทั้งหมดจะถูกจัดแบ่งออกเป็นกลุ่ม แต่ละกลุ่มจะใช้อรหัสแม่เดียวกัน โดยสมมติให้แบ่งออกเป็นกลุ่มละ  $M$  คน จำนวน  $G$  กลุ่มดังนั้นจะต้องใช้อรหัสแม่ที่แตกต่างกันทั้งสิ้นจำนวน  $G$  รหัส

ส่วนทางภาครับข้อมูลนั้นจะแบ่งออกเป็น 2 ขั้นตอน ในขั้นตอนแรก จะนำข้อมูลที่ได้จากสายอากาศแต่ละตัวมาผ่านแมตริกซ์ฟิลเตอร์ที่ใช้วิธีการปรับเท่าแบบ MRC เพื่อแผ่ข้อมูลกลับซึ่งจะต้องคูณด้วยเฟดดิ้งการลดทอนจากช่องสัญญาณที่ประมาณได้ ซึ่งข้อมูลจากผู้ใช้แต่ละกลุ่มจะถูกแยกออกจากกันในขั้นตอนนี้ด้วยคุณสมบัติความตั้งฉากของซูดรหัสแม่ จากนั้นขั้นตอนที่ 2 คือนำเอาสัญญาณทั้งหมดจากสายอากาศแต่ละตัวมารวมกันแล้วนำไปผ่านกระบวนการ V-BLAST เพื่อกำจัดสัญญาณรบกวนระหว่างผู้ใช้ที่ใช้อรหัสแม่เดียวกันออกไป

โดยจะใช้  $Y$  ซึ่งเป็นตัวแปรสถิติที่ถูกนำมาใช้ผ่านกระบวนการ V-BLAST ทั้งนี้ค่า  $Y$  นี้จะได้มาหลังจากนำสัญญาณที่รับมาได้จากสายอากาศแต่ละตัวมาผ่านแมตริกซ์ฟิลเตอร์เพื่อคูณเข้ากับรหัสแม่และเฟดดิ้งการลดทอนของช่องสัญญาณ ดังสมการ

$$Y^{(0)} = \sum_{p=1}^P H_p^H \otimes S^T \bar{r}_p = \tilde{R} \bar{a} + \eta \quad \dots\dots\dots(3-1)$$

เมื่อ 
$$\tilde{R} = \sum_{p=1}^P H_p^H \otimes S^T S \otimes H_p = \sum_{p=1}^P \psi_p^H \psi_p \quad \dots\dots\dots(3-2)$$

เมื่อ สมาชิกของเมตริกซ์  $\psi_p$  ซึ่งมีขนาด  $N \times K$  คือ  $\psi_{p,i,j}$  นิยามโดย

$$\psi_{p,i,j} = c_j(i) H_{p,j}(i) \quad \dots\dots\dots(3-3)$$

$\tilde{R}$  คือ เมตริกซ์ space-code cross correlation ซึ่งมีขนาด  $K \times K$

$S = [S_1 \ S_2 \ \dots \ S_K]$  คือ เมตริกซ์ของซูดรหัสแม่ขนาด  $N \times K$  เมื่อ  $N$  คือ อัตราแม่

$\bar{r}_p$  คือ เวกเตอร์ขนาด  $N \times 1$  ของสัญญาณที่รับได้ทางภาครับหลังจากผ่านกระบวนการ FFT

$H_p$  คือ เมตริกซ์ฟังก์ชันถ่ายโอนของช่องสัญญาณที่ผ่านมายังสายอากาศภาครับตัวที่  $p$

$H_{p,j}(i)$  คือ สมาชิกของเมตริกซ์  $H_p$  ซึ่งเป็นสัมประสิทธิ์ของช่องสัญญาณ ของคลื่นพหุย่อยที่  $i$  ของผู้ใช้คนที่  $j$  มายังสายอากาศภาครับที่  $p$

$c_j(i)$  คือ ชิพที่  $i$  ของรหัสของผู้ใช้คนที่  $j$

$p$  คือ ดรรชนีแสดงหมายเลขของสายอากาศทางภาครับซึ่งมีทั้งสิ้น  $P$  สายอากาศ

$N$  คือ ความยาวของรหัสแม่ที่ใช้

และ  $K$  คือ จำนวนผู้ใช้ทั้งหมดในระบบ

และให้ 
$$\eta = \sum_{p=1}^P \psi_p^H n_p \quad \dots\dots\dots(3-4)$$

$n_p$  คือ เวกเตอร์ของสัญญาณรบกวนแบบเกาส์ของสายอากาศภาครับตัวที่  $p$

สัญลักษณ์  $^H$  แทนการทำ conjugate transpose และ สัญลักษณ์  $\otimes$  แทนการทำ Hadamard product

ให้ตัวแปร  $k_i$  เป็นตัวแปรที่ใช้ในการบ่งบอกถึงลำดับ ที่ใช้ในการตัดสินใจตัดสินบนิตข้อมูล โดย  $k_i$  มีความหมายว่าข้อมูลที่ถูกลัดออกมาเป็นลำดับที่  $i$  คือ ข้อมูลจากผู้ใช้คนที่  $k_i$

ในขั้นตอนการทำกระบวนการ V-BLAST นั้นจะเป็นไปดังที่อธิบายในหัวข้อที่ 2-3 เพียงแต่จะใช้ตัวแปรทางสถิติ  $Y^{(i)}$  แทน  $\tilde{r}$  โดยจะแบ่งออกเป็น 3 ขั้นตอนย่อย

ขั้นตอนที่ 1 จะทำการหา pseudo inverse ของเมตริกซ์ space-code cross correlation  $\tilde{R}$

$$G_i = [\tilde{R}^{(i)}]^+ \quad \dots\dots\dots(3-5)$$

แล้วทำการจัดลำดับหาผู้ใช้คนที่  $k_i$  ที่มี post detection SNR มากที่สุด

$$k_i = \arg_j \min \| (G_i)_j \|^2 \quad : j \notin \{k_1 \dots k_{i-1}\} \quad \dots\dots\dots(3-6)$$

จากนั้นจะเลือกนัลลิ่งกเวกเตอร์ที่สอดคล้องกับผู้ใช้คนนี้จากแถวที่  $k_i$  ของเมตริกซ์  $G_i$

$$w_{k_i} = (G_i)_{k_i} \quad \dots\dots\dots(3-7)$$

ขั้นตอนที่ 2 ทำการตัดสินใจตัดสินบนิตข้อมูล

$$\hat{a}_{k_i} = \Phi[w_{k_i}^T Y^{(i)}] \quad \dots\dots\dots(3-8)$$

เมื่อ  $\Phi$  คือ slicing operation ซึ่งขึ้นอยู่กับเทคนิคการมอดูเลตที่ใช้

ขั้นตอนที่ 3 นำบิตที่ตัดสินใจไปแล้วไปหักล้างออกจาก เวกเตอร์  $Y_{MU}$

$$Y^{(i+1)} = Y^{(i)} - (\tilde{R}^{(i)})^{k_i} \hat{a}_{k_i} \quad \dots\dots\dots(3-9)$$

เมื่อ  $(\tilde{R}^{(i)})^{k_i}$  คือ หลักที่  $k_i$  เมตริกซ์  $\tilde{R}^{(i)}$

โดย  $\tilde{R}^{(i+1)}$  ได้มาจากการแทนหลักที่  $k_i$  และแถวที่  $k_i$  ของเมตริกซ์  $\tilde{R}^{(i)}$  ด้วย 0 ทั้งหมด

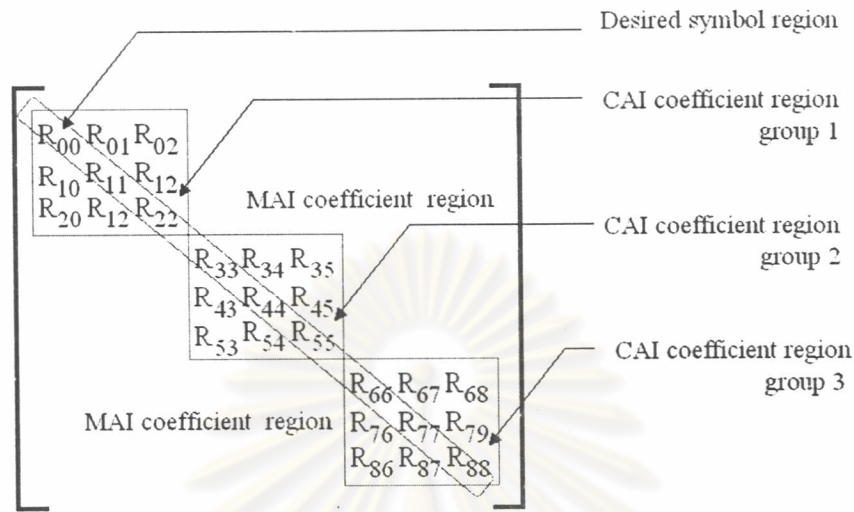
จากนั้นทำซ้ำขั้นตอนที่ 1 -3 จนตัดสินใจข้อมูลจากผู้ใช้ทุกคนครบทั้งหมด

### 3.2 การปรับปรุงอัลกอริทึมของระบบที่นำเสนอด้วยวิธีการละทิ้งผลของ MAI

จากที่ได้บรรยายมาข้างต้นจะเห็นได้ว่าระบบที่นำเสนอจะมีสัญญาณแทรกสอด 2 ประเภท ที่จำเป็นต้องนำมาพิจารณา นั่นคือสัญญาณแทรกสอด MAI และสัญญาณแทรกสอด CAI โดยสัญญาณ MAI นั้นคือสัญญาณแทรกสอดระหว่างกลุ่มผู้ใช้ที่ใช้รหัสที่แตกต่างกัน ซึ่งเป็นผลมาจากสัญญาณรบกวนและผลจากเฟดดิ้งชนิดเลือกความถี่ทำให้รหัสแต่ละสัญญาณเสียความตั้งใจไป ส่วนสัญญาณ CAI นั้น คือ สัญญาณแทรกสอดระหว่างผู้ใช้ภายในกลุ่มเดียวกันหรือระหว่างผู้ใช้ที่ใช้รหัสเหมือนกันนั่นเอง

ผลกระทบของสัญญาณ MAI และ CAI ที่มีผลต่อสัญญาณที่รับได้ทางภาครับนั้น สามารถที่จะอธิบายได้อย่างชัดเจนโดยอาศัยเมตริกซ์ space-code cross correlation ( $\tilde{R}$ ) ที่ได้

กล่าวถึงไว้สมการ (3-2) โดยจะสามารถจะแสดงรายละเอียดของแต่ละสมาชิกของเมตริกซ์นี้ดังรูปที่ 3-2

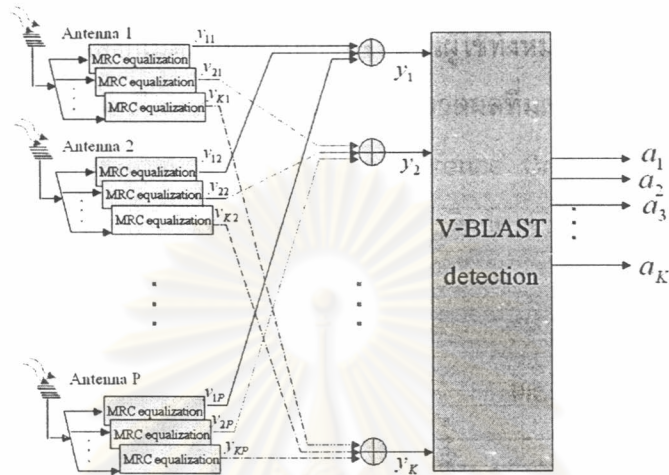


รูปที่ 3-2 แสดงสมาชิกของเมตริกซ์ space-code cross correlation ซึ่งมีจำนวนผู้ใช้ทั้งสิ้น 9 คน แบ่งออกเป็น 3 กลุ่ม กลุ่มละ 3 คน

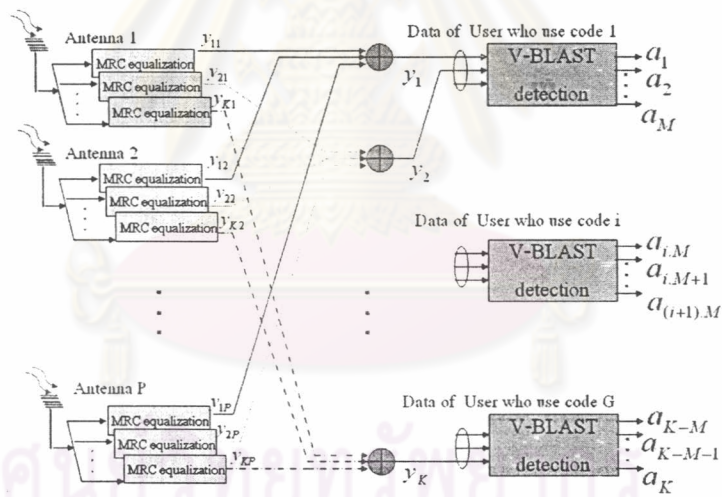
เมื่อพิจารณาสมการที่ (3-1) ควบคู่ไปด้วยจะเห็นได้ว่าทุก ๆ สมาชิกในแถวที่  $i$  ของเมตริกซ์ space-code cross correlation นั้นจะเป็นค่าสัมประสิทธิ์ของทุก ๆ สัญลักษณ์ข้อมูลของผู้ใช้ทุก ๆ คนมาประกอบเป็นสัญญาณ  $y_i$  ซึ่งเป็นสมาชิกตัวที่  $i$  ของเวกเตอร์  $Y$  โดย  $\tilde{R}_{ii}$  คือค่าสัมประสิทธิ์ของสัญญาณที่ต้องการของผู้ใช้คนที่  $i$  ส่วน  $\tilde{R}_{ij}$  ที่อยู่ในบริเวณ CAI ดังแสดงในรูปที่ 3-2 นั้นจะเป็นค่าสัมประสิทธิ์ของสัญญาณของผู้ใช้รายอื่นที่แทรกสอดเดียวกับผู้ใช้คนที่  $i$  ส่วน  $\tilde{R}_{ij}$  ที่อยู่ในบริเวณ MAI นั้นจะเป็นสัมประสิทธิ์ของผู้ใช้รายอื่นที่แทรกสอดที่แตกต่างกันออกไป โดยจะเรียกค่าแอมพลิจูดของสัมประสิทธิ์ทั้ง 2 ประเภทนี้ว่าค่าสัมประสิทธิ์ CAI และค่าสัมประสิทธิ์ MAI ตามลำดับ ซึ่งจะเห็นได้ว่าค่าแอมพลิจูดของสัมประสิทธิ์เหล่านี้จะขึ้นอยู่กับลักษณะสมบัติของช่องสัญญาณเป็นสำคัญ

เพื่อต้องการที่จะลดความซับซ้อนของอัลกอริทึมของระบบที่นำเสนอ ในกรณีที่มีระบบมี MAI น้อย เราอาจจะละทิ้งผลของ MAI นี้ลงไปได้ โดยลดขนาดของเมตริกซ์  $\tilde{R}$  ที่จะนำมาใช้ในการประมวลผลลง โดยขนาดของเมตริกซ์  $\tilde{R}'$  ใหม่ที่ได้จะเป็นเมตริกซ์จัตุรัสและมีขนาดเล็กลง โดยกระบวนการที่เปลี่ยนแปลงไปได้แสดงไว้ในรูปที่ 3-3 โดยเมตริกซ์  $\tilde{R}'$  นี้มีเพียงเฉพาะส่วน CAI และส่วนของสัญญาณที่ต้องการของเมตริกซ์  $\tilde{R}$  เดิมในรูปที่ 3-2 เท่านั้น อย่างไรก็ตาม

ตามจำเป็นที่จะต้องพึงสังวรไว้ว่าวิธีการลดความซับซ้อนในรูปแบบนี้จะใช้ได้เฉพาะในกรณีที่ระบบมี สัญญาณ MAI ในปริมาณที่น้อยเท่านั้น ตัวอย่างเช่น ในกรณีที่ระบบมีการส่งข้อมูลที่ SNR สูง หรือในสภาพแวดล้อมที่มีเฟดดิ้งแบบเรียบ



รูปที่ 3-3(ก) แสดงเครื่องรับที่ใช้อัลกอริทึม V-BLAST



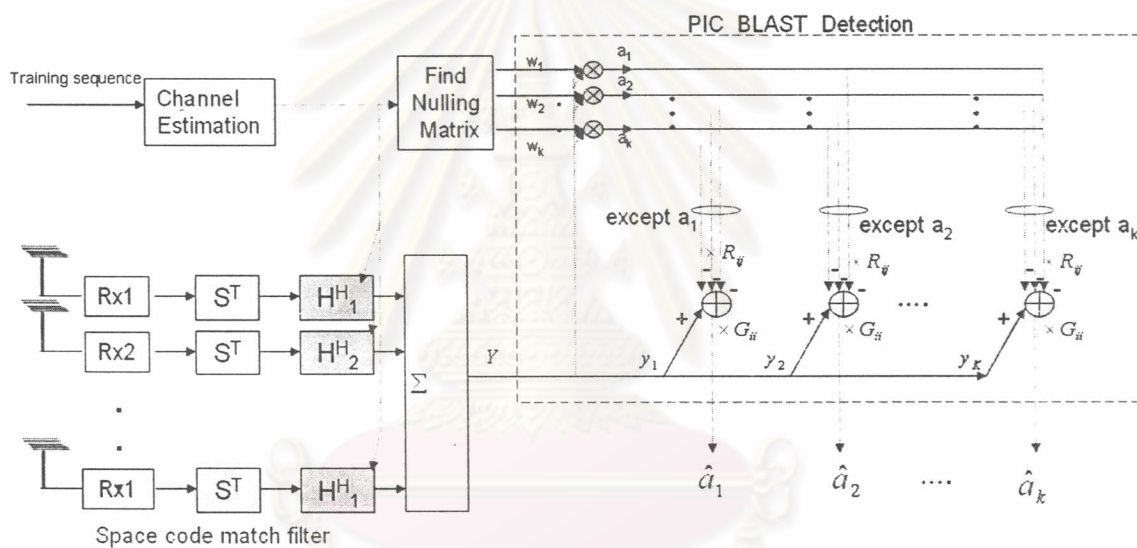
รูปที่ 3-3(ข) แสดงเครื่องรับที่ใช้วิธีการละทิ้งผลของ MAI

### 3.3 การปรับปรุงอัลกอริทึมของระบบที่นำเสนอโดยวิธีการหักล้างสัญญาณแบบผสม

เราสามารถที่จะมองได้ว่าอัลกอริทึม V-BLAST แบบเดิมนั้นมีลักษณะที่คล้ายคลึงกับการหักล้างสัญญาณแบบต่อเนื่อง (Successive Interference Cancellation: SIC) โดยหลังจากที่มีการหาค่าผลคูณไขว้ของเมตริกซ์ space-code cross correlation แล้วจะทำการตัดสินใจหาสัญญาณข้อมูลของผู้ใช้ที่น่าเชื่อถือที่สุดออกมา หลังจากนั้นจะนำเอาสัญญาณข้อมูลที่ตัดสินใจแล้วนี้ไปหักล้างออกจากตัวแปรสถิติ  $Y$  ที่มีผลของบิตข้อมูลของผู้ใช้ทุกคนรวมกันอยู่

และพร้อมกันนั้นจะลบผลของผู้ใช้ที่ตัดลีนิตไปแล้วนี้ออกจากเมตริกซ์ space-code cross correlation ด้วยเพื่อนำไปใช้ประมวลผลในรอบถัดไปด้วย

เนื่องจากเครื่องรับของระบบที่ใช้สถาปัตยกรรม V-BLAST นั้น จะมีความซับซ้อนที่สูงโดยเฉพาะอย่างยิ่งในขั้นตอนการหาค่าลิ่งเมตริกซ์ และในอัลกอริทึมของ V-BLAST นั้น จำนวนครั้งในการหาค่าลิ่งเมตริกซ์นี้จะเท่ากับจำนวนผู้ใช้ทั้งหมด จึงมีผลทำให้ต้องการเครื่องรับที่มีความซับซ้อนสูงและจะเกิดความล่าช้าในการประมวลผลที่มาก ดังนั้นจึงนำเอาแนวคิดของการหักล้างสัญญาณรบกวนแบบขนาน (Parallel Interference Cancellation: PIC) มาประยุกต์ใช้เพื่อแก้ไขข้อเสียที่ได้กล่าวมานี้ โดยจะทำให้มีการหาค่าลิ่งเมตริกซ์เพียงครั้งเดียวเท่านั้น โดยรูปแบบของอัลกอริทึม จะเป็นดังรูปที่ 3-4



รูปที่ 3-4 แบบจำลองภาครับของระบบ MC-CDMA ที่ใช้สถาปัตยกรรม V-BLAST ที่ประยุกต์ใช้แนวคิดของการหักล้างสัญญาณรบกวนแบบขนาน

โดยตัวแปรทางสถิติ  $Y$  และเมตริกซ์ space-code cross correlation ที่ใช้ในอัลกอริทึมที่ถูกปรับปรุงนี้จะเหมือนในระบบเดิมตามสมการที่ (3-1) และ (3-2) โดยขั้นตอนทั้งหมดจะแบ่งออกเป็น 3 ขั้นตอนย่อย

ขั้นตอนที่ 1 จะทำการหา pseudo inverse ของเมตริกซ์ space-code cross correlation  $\tilde{R}$  เมื่อนำลิ่งเมตริกซ์  $G$  ออกมา

$$G = [\tilde{R} (i)]^+ \dots\dots\dots(3-9)$$

ซึ่งนัลลิ่งก์เมตริกซ์นี้จะถูกใช้ตลอดในการตัดสินใจถอดสัญญาณของผู้ใช้ทุกคนออกมา จนกว่าจะมาทำการประมาณช่องสัญญาณครั้งถัดไป โดยนัลลิ่งก์เวกเตอร์ที่สอดคล้องกับของผู้ใช้คนที่  $i$  จะตรงกับแถวที่  $i$  ของนัลลิ่งก์เมตริกซ์นี้เอง

$$\bar{w}_i = (G)_i \quad \dots\dots\dots(3-10)$$

ขั้นตอนที่ 2 ทำการตัดสินใจถอดสัญญาณโดยเริ่มแรกจะใช้นัลลิ่งก์เวกเตอร์ที่ได้นี้ทำการหาตัวแปร  $a'$  ของผู้ใช้ทุกคนออกมาก่อนโดย

$$a'_i = \Phi[w_i, Y] \quad \dots\dots\dots(3-11)$$

โดยในการประมวลผลการหักล้างสัญญาณแบบขนานนี้ บิตข้อมูลของผู้ใช้คนที่ต้องการจะไม่ได้ทำการหาออกมาจากเวกเตอร์ของตัวเองโดยตรง แต่จะทำการหาออกมาจากสัญญาณที่เหลือในตัวแปรสถิติ  $Y$  หลังจากหักล้างผลของบิตข้อมูลของผู้อื่นออกไปจนหมดแล้วดังสมการ

$$\bar{c}^{(i)} = \begin{bmatrix} \alpha a'_1 \\ \alpha a'_2 \\ \vdots \\ (1-\alpha)a'_i \\ \vdots \\ \alpha a'_K \end{bmatrix} \quad \dots\dots\dots(3-12)$$

$$y'_i = \left[ y_i - \sqrt{\frac{E}{M}} \tilde{R}_{(i)} \cdot \bar{c}^{(i)} \right] \quad \dots\dots\dots(3-13)$$

โดย  $i$  เป็นดรรชนีแสดงถึงผู้ใช้คนที่  $i$

และ  $y_i$  ซึ่งเป็นสมาชิกตัวที่  $i$  ของเวกเตอร์  $Y$

$\tilde{R}_{(i)}$  คือแถวที่  $i$  ของเมตริกซ์  $\tilde{R}$

$$\hat{a}_i = \Phi[G_{ii} y'_i] \quad \dots\dots\dots(3-14)$$

โดยค่า  $\alpha$  ที่ปรากฏอยู่ในสมการที่ (3-12) นี้ จะเป็นกรณีที่ต้องการให้มีการหักล้างข้อมูลของผู้ใช้รายอื่นแต่เพียงบางส่วน หรือตามหลักการของการหักล้างสัญญาณแทรกสอดแบบขนานบางส่วน (Partial Parallel Interference Cancellation : PPIC) นั้นเอง โดยค่า  $\alpha$  นี้จะเป็นค่าคงที่มีค่าอยู่ระหว่าง 0 ถึง 1 นั้นเอง [31]

การหาค่าบิตข้อมูล  $\hat{a}_i$  ของผู้ใช้แต่ละคนที่ได้นี้จะสามารถกระทำขนานไปพร้อม ๆ กันได้ทำให้เวลาที่จำเป็นต้องใช้ในการประมวลผลที่ได้นี้มีค่าลดลง

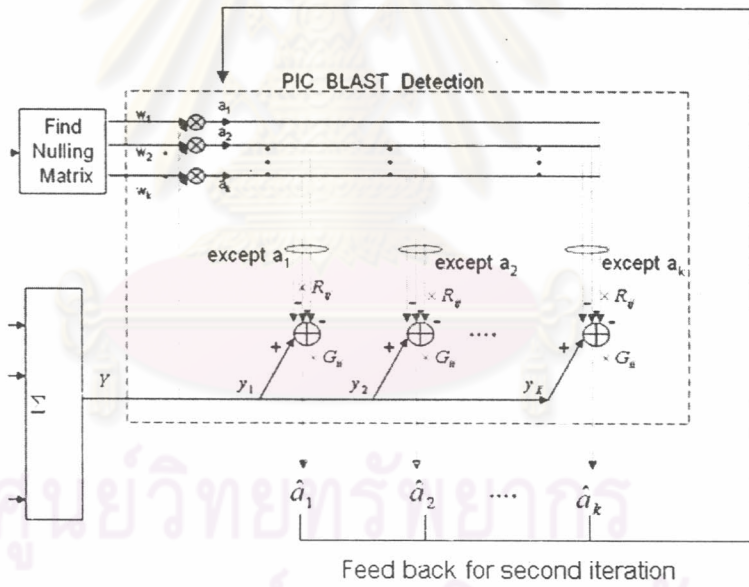


ขั้นตอนที่ 3 ถึงแม้จะได้ค่าสัญลักษณ์ข้อมูลของผู้ใช้ทุกคนออกมาแล้ว แต่การประมวลผลในการหักล้างสัญญาณแบบขนานนี้สามารถที่จะทำการวนซ้ำ (iteration) ในขั้นตอนที่ 2 ได้อีกเพื่อที่จะเพิ่มความน่าเชื่อถือของบิตข้อมูลที่ได้โดยค่าเวกเตอร์  $\vec{c}^{(i)}$  ที่จะใช้ในรอบถัดไปจะมีค่าดังนี้

$$\vec{c}^{(i)} = \begin{bmatrix} \alpha \hat{a}_1 \\ \alpha \hat{a}_2 \\ \vdots \\ (1-\alpha) \hat{a}_i \\ \vdots \\ \alpha \hat{a}_K \end{bmatrix} \dots\dots\dots(3-15)$$

โดยค่า  $\hat{a}_i$  เหล่านี้คือค่าบิตข้อมูลที่เป็นผลลัพธ์ที่ได้จากการประมวลผลในรอบที่แล้ว (ขั้นตอนที่ 2)

ค่า  $\alpha$  ที่ปรากฏในสมการที่ (3-15) นี้สามารถปรับเปลี่ยนได้ในการทำซ้ำในรอบถัดไปโดยกำหนดให้ค่า  $\alpha$  นี้มีค่าที่เข้าใกล้ 1 มากยิ่งขึ้น ทั้งนี้เพราะ  $\hat{a}_i$  ที่ใช้ในการทำซ้ำในรอบถัดไปจะมีความน่าเชื่อถือมากยิ่งขึ้นเมื่อเทียบกับ  $\hat{a}_i$  ที่ใช้ในรอบก่อนหน้านี้



รูปที่ 3-5 แบบจำลองภาครับของระบบ MC-CDMA ที่ใช้สถาปัตยกรรม V-BLAST ที่ประยุกต์ใช้แนวคิดของการหักล้างสัญญาณรบกวนแบบขนานที่มีการวนซ้ำ

อย่างไรก็ตามอัลกอริทึมที่ถูกปรับปรุงนี้จะยังไม่สามารถทำให้ระบบมีสมรรถนะที่ดีเพียงพอ จึงต้องปรับปรุงอัลกอริทึม V-BLAST ที่ใช้หลักการหักล้างสัญญาณรบกวนแบบขนานนี้เพิ่มเติมโดยกำหนดให้ในรอบการทำซ้ำเดียวกันนั้นมีการป้อนไปข้างหน้าและใช้บิตข้อมูลของผู้ใช้คนที่ตัด

สิ้นไปแล้วในการตัดสินใจข้อมูลของผู้อื่นต่อไปซึ่งเป็นการเปลี่ยนแปลงในขั้นตอนที่ 2 ของอัลกอริทึมข้างต้น

ให้ตัวแปร  $k_i$  เป็นตัวแปรที่ใช้ในการบ่งบอกถึงลำดับที่ใช้ในการตัดสินใจข้อมูล โดย  $k_i$  มีความหมายว่าข้อมูลที่ถูกระบุออกมาเป็นลำดับที่  $i$  คือข้อมูลจากผู้ใช้คนที่  $k_i$  การป้อนไปข้างหน้าจะแสดงไว้ดังสมการที่ (3-16) และ (3-17) โดยจะมีการเปลี่ยนแปลงเมตริกซ์  $\bar{c}^{(i)}$  ในสมการที่ (3-15) ดังนี้

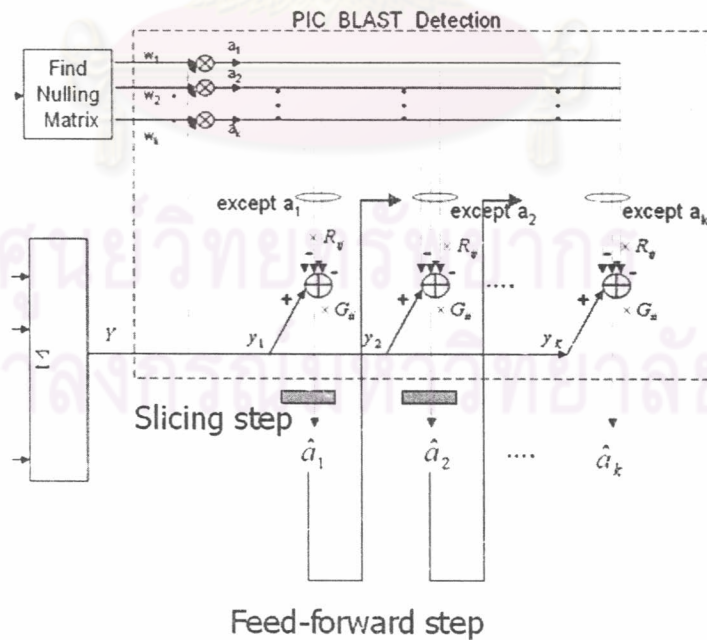
$$\bar{c}^{(k_{i+1})} = \bar{c}^{(k_i)} \tag{3-16}$$

และ 
$$a'_{k_i} = \hat{a}_{k_i} \tag{3-17}$$

$$y'_{k_{i+1}} = \left[ y_{k_{i+1}} - \sqrt{\frac{E}{M}} \bar{R}_{(k_{i+1})} \cdot \bar{c}^{(k_{i+1})} \right] \tag{3-18}$$

$$\hat{a}_{k_{i+1}} = \Phi \left[ G_{k_{i+1}k_{i+1}} y'_{k_{i+1}} \right] \tag{3-19}$$

ซึ่งจะเห็นได้ว่าอัลกอริทึมที่มีการป้อนไปข้างหน้าจะมีหลักการที่คล้ายคลึงกับลักษณะของการหักล้างสัญญาณแบบต่อเนื่อง (Successive Interference Cancellation : SIC) ดังนั้นจึงเรียกอัลกอริทึมการหักล้างสัญญาณแบบขนานที่มีการป้อนไปข้างหน้าว่าอัลกอริทึมที่ใช้วิธีการหักล้างสัญญาณแบบผสม (Hybrid Interference Cancellation :HIC) ซึ่งจะแสดงได้ดังรูปที่ 3-6



รูปที่ 3-6 แบบจำลองภาครับของระบบ MC-CDMA ที่ใช้สถาปัตยกรรม V-BLAST ที่ประยุกต์ใช้แนวคิดของการหักล้างสัญญาณรบกวนแบบผสม

โดยทั้งนี้อัลกอริทึมที่กล่าวมาข้างต้นนี้จำเป็นต้องได้รับการจัดลำดับค่าตัวแปร  $k_1, k_2, \dots, k_i, \dots, k_K$  ที่ใช้ในการประมวลผลที่มีการป้อนกลับเพื่อให้มีสมรรถนะที่ดีที่สุดดังนี้

$$k_i = \arg_j \max \|(G)_j\|^2 \quad : j \notin \{k_1, \dots, k_{i-1}\} \dots \dots \dots (3-20)$$

จากการจัดเรียงลำดับนี้จะช่วยลดความไม่น่าเชื่อถือของบิตข้อมูลของผู้ใช้รายอื่นที่จะนำมาใช้ในการตัดสินใจบิตข้อมูลของผู้ใช้คนที่ต้องการ โดยการจัดเรียงตามสมการที่ (3-20) จะเป็นการจัดเรียงลำดับตามค่านอร์มของนัลลิงก์เวกเตอร์  $\mathbf{h}_i$  หรือเป็นการจัดเรียงตามค่า post detection SNR ตามที่แสดงไว้ในสมการที่ (2-14) นั่นเอง โดยจะกำหนดให้มีการตัดสินใจบิตข้อมูลของผู้ใช้เริ่มจากคนที่มีค่านอร์มของนัลลิงก์เวกเตอร์มากที่สุด หรือมีค่า post detection SNR ต่ำที่สุดก่อน นั่นคือจะเริ่มจากผู้ใช้ที่มีค่าความน่าเชื่อถือของบิตข้อมูลที่ตัดสินใจออกมาต่ำ ซึ่งจะตรงกันข้ามกับกรณีของอัลกอริทึม V-BLAST เดิม ทั้งนี้เพราะในกระบวนการของอัลกอริทึมที่ถูกปรับปรุงนี้การตัดสินใจบิตข้อมูลของผู้ใช้แต่ละคนนั้นจะอาศัยบิตข้อมูลของผู้ใช้รายอื่นที่ได้จากการหานัลลิงก์เมตริกซ์ในขั้นตอนแรก ดังนั้นจึงเป็นการเพิ่มความถูกต้องยิ่งขึ้นถ้าบิตข้อมูลของผู้ใช้รายอื่นมีความน่าเชื่อถือมากกว่า นั่นคือค่า post detection SNR ของผู้ใช้คนที่ต้องการนั้นควรมีค่าที่ต่ำสุดนั่นเอง

### 3.4 การปรับปรุงอัลกอริทึมของระบบที่นำเสนอด้วยวิธี multiple slicing ร่วมกับวิธีการหักล้างสัญญาณแบบผสม

อัลกอริทึมที่ได้เสนอนี้มีการผนวกอัลกอริทึมวิธีการหักล้างสัญญาณแบบผสมเข้ากับกรรมวิธี multiple slicing ซึ่งแนวคิดของวิธี multiple slicing นี้ได้ถูกนำเสนอโดย Ashish Bhargave, Rui J.P.de Figueiredo และ Torbjorn Eltoft ในปี ค.ศ. 2001 [21] โดยวิธีการ multiple slicing นี้แต่เดิมนั้นได้ถูกนำเสนอขึ้นเพื่อเพิ่มสมรรถนะของอัลกอริทึม V-BLAST โดยจะส่งผลเสียให้มีความซับซ้อนที่สูงขึ้นยิ่งกว่าของอัลกอริทึม V-BLAST ดั้งเดิมที่ถูกเสนอโดย Foschini จึงเป็นที่ชัดเจนว่าอัลกอริทึม multiple slicing เดิมก็ไม่มี ความเหมาะสมที่จะนำไปใช้ในทางปฏิบัติจริง

โดยอัลกอริทึม V-BLAST ที่ประยุกต์ใช้วิธีการหักล้างสัญญาณแบบผสมที่ได้นำเสนอในหัวข้อที่แล้วจะมีค่าความซับซ้อนที่ต่ำลงมากเมื่อเทียบกับอัลกอริทึม V-BLAST เดิม ดังนั้นในวิทยานิพนธ์ฉบับนี้จึงได้นำเอาหลักการของวิธีการ multiple slicing มาประยุกต์ใช้เพื่อที่จะทำให้อัลกอริทึมที่ใช้วิธีการหักล้างสัญญาณแบบผสมนั้นมีสมรรถนะที่ดีขึ้น โดยแลกกับความซับซ้อน

ที่เพิ่มขึ้นไม่มากนัก เพื่อให้ได้มาซึ่งอัลกอริทึมที่มีความเหมาะสมทั้งในแง่ที่ทำให้ระบบมีสมรรถนะที่ดีและในแง่ที่มีความซับซ้อนที่เหมาะสมเพียงพอที่จะนำไปใช้ได้จริงในทางปฏิบัติ

เงื่อนไขการตรวจวัดแบบเหมาะสมที่สุด (optimal detection) [21] จะใช้เงื่อนไขของ maximum likelihood โดยการหาค่าระยะความผิดพลาด (error distance) ในสมการที่ (3-21) ให้มีค่าน้อยสุด

$$\sum_{j=1}^K \sigma_j^{-2} (y_i - \sum_{i=1}^K \tilde{R}_{ji} a_i)^2 \dots\dots\dots(3-21)$$

เมื่อกำหนดให้สัญญาณรบกวนเป็นตัวแปรสุ่มแบบเกาส์ที่เป็นอิสระต่อกันและมีความแปรปรวนเท่ากับ  $\sigma^2$  โดยอัลกอริทึมที่นำเสนอจะคล้ายคลึงกับกรรมวิธีการตรวจรับแบบเหมาะสมที่สุดโดยการตัดสินใจสัญลักษณ์ข้อมูลของผู้ใช้รายหนึ่ง จะมีพื้นฐานอยู่บนระยะของความผิดพลาดในการตัดสินใจสัญลักษณ์ข้อมูลของผู้ใช้รายอื่นในลำดับถัดไปด้วย หลักการสำคัญสำหรับอัลกอริทึมนี้จะมี 2 ประการ

1. ตามเงื่อนไขของ maximum likelihood ในสมการที่ (3-21) นั้น การตัดสินใจสัญลักษณ์ข้อมูล 1 ครั้งจะอาศัยค่าระยะความผิดพลาดของผู้ใช้ทุกคน (K คน) ในการตัดสินใจ นั่นคือจะรวมการพิจารณาค่าความผิดพลาดของผู้ใช้คนที่จะถูกนำมาตัดสินใจต่อไปในอนาคตทุกคนที่เหลือด้วย แต่ในอัลกอริทึมที่เสนอนี้จะนำมาใช้เป็นพื้นฐานเพียง T คนเท่านั้น ( $T \leq M$ ) นั่นคือเสมือนเรามองผลรวมของค่าความผิดพลาดของผู้ใช้ล่วงหน้าไปเพียง T รายเท่านั้น ดังนั้น  $\sum_j$  ในสมการที่ (3-21) จะเป็นผลบวกของ T เท่านั้น ดังนั้นค่าความซับซ้อนที่ได้จะมีค่าที่ลดลงมาก
2. แทนที่จะพิจารณาเพียงสัญลักษณ์ที่น่าจะเป็นไปได้ที่สุดเพียงตัวเดียวเท่านั้นเหมือนดังกรณีอัลกอริทึม V-BLAST เดิม เราจะพิจารณาสัญลักษณ์จำนวน S สัญลักษณ์ที่น่าจะเป็นไปได้แทน โดยจะเลือกจากสัญลักษณ์ที่มีค่าระยะความผิดพลาดน้อยสุด S ตัวแรกเท่านั้นเอง

จุฬาลงกรณ์มหาวิทยาลัย

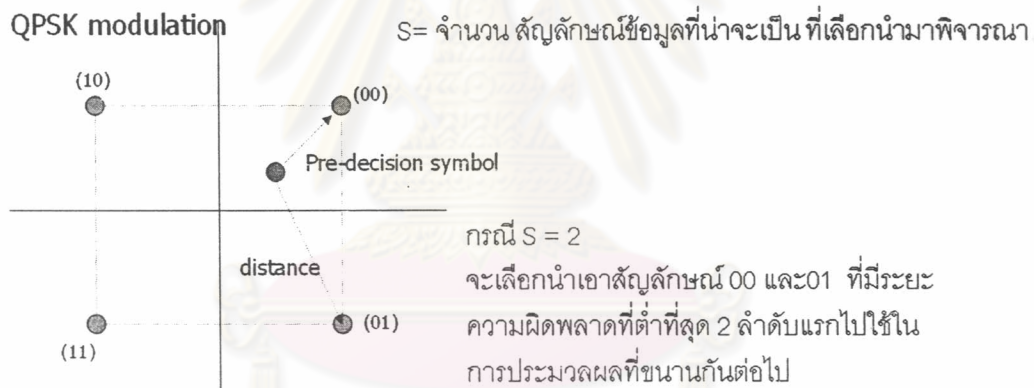
จะเห็นได้ว่าตัวแปร S และ T จะเป็นพารามิเตอร์สำคัญในการปรับเปลี่ยนอัลกอริทึมนี้ กรณีที่ T เท่ากับจำนวนผู้ใช้ที่ใช้รหัสซ้ำกันและ S เท่ากับจำนวนสัญลักษณ์ข้อมูลทั้งหมดที่น่าจะเป็นไปได้ (constellation size) นั้น ก็จะเป็นกรณีเดียวกันกับกรณีเงื่อนไข maximum likelihood ส่วนในกรณี V-BLAST แบบเดิมก็จะเหมือนในกรณีที่ตัวแปร T เท่ากับ 1 และตัวแปร S เท่ากับ 1 นั่นเอง

จะเห็นได้ว่าแทนที่เราจะทำการตัดสินใจสัญลักษณ์ข้อมูลในทันทีที่เราจะมองล่วงหน้าถึงผลการตัดสินใจในหลาย ๆ ทางที่เป็นไปได้สำหรับผู้ใช้แต่ละคน โดยการตัดสินใจที่แตกต่างกันนี้จะถูกนำมาสร้างโครงสร้างแบบต้นไม้ขึ้น ทำยสุดท้ายสัญลักษณ์ข้อมูลที่เป็นไปได้จะถูกตัดสินใจหลังจากพิจารณาค่าของระยะความผิดพลาดของผู้ใช้จนครบ T คนถัดไปแล้ว

เพื่อให้ง่ายต่อการทำความเข้าใจเราจะยกตัวอย่างในกรณี S เท่ากับ 2 และ T เท่ากับ 2 ในกรณี T เท่ากับ 2 นั้นเราจะรอการตัดสินใจสัญลักษณ์ข้อมูลไว้จนกว่าจะถึงรอบของผู้ใช้คนที่ถัดไป ส่วนกรณี S เท่ากับ 2 นั้นจะแสดงว่าจะเลือกสัญลักษณ์ที่น่าจะเป็นไปได้ซึ่งมีระยะค่าความผิดพลาดน้อยที่สุด 2 สัญลักษณ์ ดังแสดงในรูปที่ 3-7 มาใช้ในการประมวลผลถัดไปโดยเราจะเรียกกระบวนการนี้ว่า "multiple slicing" โดยจะเขียนให้อยู่ในรูปสมการดังนี้

$$\Phi(y_1) = (a_{1,1}, a_{1,2}) \dots\dots\dots(3-22)$$

เมื่อ  $\Phi(y_1)$  แสดงถึง slicing operation ที่สอดคล้องกับเทคนิคการมอดูเลตที่ใช้และได้ผลลัพธ์เป็นตัวเลือกที่น่าจะเป็นไปได้  $a_{1,1}$  และ  $a_{1,2}$



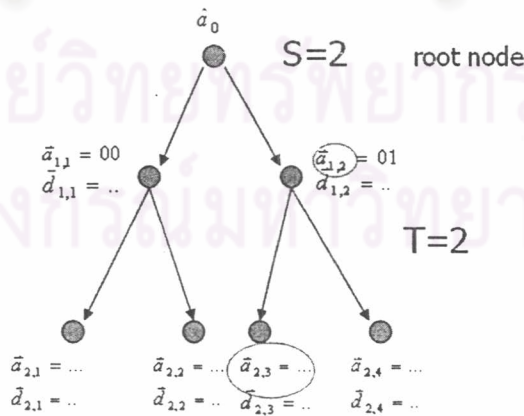
รูปที่ 3-7 กระบวนการ multiple slicing

ตัวอย่างของโครงสร้างแบบต้นไม้ของกรณี S เท่ากับ 2 และ T เท่ากับ 2 ได้แสดงไว้ดังรูปที่ 3-8 โดยแต่ละ node ยกเว้น root node หรือ node บนสุดจะแสดงถึงสัญลักษณ์ตัวเลือกที่เป็นไปได้ ทุก ๆ node ที่อยู่ระดับเดียวกันจะแสดงถึงสัญลักษณ์ตัวเลือกที่น่าจะเป็นไปได้สำหรับสัญลักษณ์ข้อมูลของผู้ใช้ 1 คน โดย root node จะแสดงถึงสัญลักษณ์ข้อมูลตัวล่าสุดที่ถูกตัดสินใจเรียบร้อยแล้ว เราจะให้ตัวแปร  $\hat{a}_i$  แสดงถึงสัญลักษณ์ข้อมูลที่ได้ตัดสินใจไปแล้ว และ  $\hat{a}_{i,j}$  แสดงถึงสัญลักษณ์ข้อมูลตัวเลือกที่ j ที่น่าจะเป็นไปได้ โดยกรณี i แสดงถึงหมายเลขของลำดับชั้นในโครงสร้างต้นไม้หรือหมายเลขรอบในการตัดสินใจครั้งหนึ่ง ๆ และกรณี j แสดงถึงหมายเลขตัวเลือกที่น่าจะเป็นไปได้ของรอบที่ i นั้น ๆ ทั้งนี้ในแต่ละชั้น หมายเลขตัวเลือกที่น่าจะเป็นไปได้จะถูกเรียงลำดับโดยเริ่มจาก 1 ดังแสดงในรูปที่ 3-8 ดังนั้นถ้าเริ่มต้นจาก node  $\hat{a}_0$  node ของชั้นถัด

มาคือ  $a_{1,1}$   $a_{1,2}$  และ node ของชั้นถัดลงไปอีกชั้นหนึ่งคือ  $a_{2,1}$   $a_{2,2}$  .....  $a_{2,4}$  โดยสัญลักษณ์ที่ถูกตัดสินใจก่อนหน้านี้จะสามารถหาได้ในเส้นทางจาก root node ไปยัง node ที่ต้องการพิจารณานั้นเอง ตัวอย่างเช่น  $a_{2,3}$  จะเป็นตัวเลือกที่เป็นไปได้ก็ต่อเมื่อสัญลักษณ์ที่ถูกตัดสินใจของคนก่อนหน้านี้คือ  $a_{1,2}$  นอกจากนี้ความหมายของ  $d_{i,j}$  ก็จะเป็นไปในทำนองเดียวกันด้วย โดย  $d_{i,j}$  เป็นค่าของระยะความผิดพลาดรวมตลอดทุกสัญลักษณ์ที่ถูกตัดสินใจก่อนหน้านี้รวมทั้งระยะความผิดพลาดของสัญลักษณ์ที่สอดคล้องกับ node นั้นด้วย ส่วนตัวแปร  $\bar{c}^{(i,j)}$  นั้นเป็นเวกเตอร์ที่ใช้ในการป้อนไปยังผู้ใช้คนถัดไปของอัลกอริทึมที่ใช้วิธีการหักล้างสัญญาณแบบผสม ซึ่งเมื่อนำมาใช้ร่วมกับวิธี multiple slicing นี้ ค่า  $\bar{c}^{(i,j)}$  ที่ใช้ในการป้อนไปตัดสินใจผู้ใช้คนถัดไปจะแปรเปลี่ยนไปตามสัญลักษณ์ตัวเลือกที่น่าจะเป็นไปได้ของผู้ใช้ที่ถูกตัดสินใจก่อนหน้านี้ในเส้นทางของโครงสร้างแบบต้นไม้จาก root node มาถึง node ที่พิจารณา นั่นคือจะขึ้นอยู่กับดรรชนี  $i$  และ  $j$  ที่ได้กล่าวไว้ข้างต้นเป็นหลัก โดยเมื่อมีการตัดสินใจเลือกสัญลักษณ์ข้อมูลของผู้ใช้คนที่  $k$  ก็จะต้องเปลี่ยนแปลงค่าสมาชิกของเวกเตอร์  $\bar{c}$  ให้สอดคล้องกับค่าสัญลักษณ์ที่ถูกเลือกดังสมการที่ (3-23)

$$\bar{c}_k^{(i,j)} = (1-\alpha)\hat{a}_k \quad \dots\dots\dots(3-23)$$

ความกว้างและความลึกของโครงสร้างต้นไม้จะถูกควบคุมโดยพารามิเตอร์  $S$  และ  $T$  ตามลำดับ ดังนั้นค่าความซับซ้อนในการคำนวณและสมรรถนะของอัลกอริทึมที่ได้จะขึ้นกับพารามิเตอร์ทั้ง 2 นี้ โดยในขั้นตอนแต่ละรอบเมื่อสัญลักษณ์ของผู้ใช้ 1 คนถูกตัดสินใจแล้วสัญลักษณ์นั้นจะถูกนำไปใช้เป็น root node ในขั้นตอนเริ่มแรกของรอบถัดไปแทนและตัวเลือกที่น่าจะเป็นไปได้ใหม่จะถูกเพิ่มเข้ามาใหม่ในชั้นล่างสุด ขณะที่กิ่งที่ไม่ต่อเนื่องจากสัญลักษณ์ที่ไม่ถูกเลือกและ node ต่าง ๆ ในกิ่งนั้นจะถูกทิ้งไป

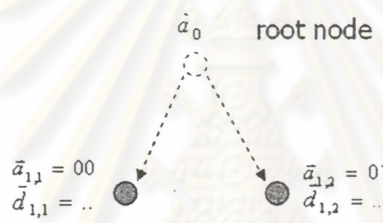


รูปที่ 3-8 โครงสร้างต้นไม้ กรณี  $S = 2$  และ  $T = 2$

เพื่อความเข้าใจอย่างสมบูรณ์ จะแสดงตัวอย่างกรณี  $S$  เท่ากับ 2 และ  $T$  เท่ากับ 2 นี้ในเชิงคณิตศาสตร์ควบคู่กับโครงสร้างแบบต้นไม้ที่สอดคล้องกันและเพื่อที่จะทำความเข้าใจได้

ง่ายยิ่งขึ้นในการอธิบายนี้จึงได้ทำการละทิ้งขั้นตอนในการจัดลำดับของผู้ใช้ที่จะต้องถูกตัดสิน  
 สัญลักษณ์ข้อมูลก่อนหลังออก โดยสมมติให้ตรวจรับที่มีการระบุถึงหมายเลขผู้ใช้จาก 1 ถึง K นั้น มี  
 การเรียงลำดับที่เหมาะสมจากก่อนไปหลังอยู่แล้ว โดยลำดับที่เหมาะสมนี้จะปฏิบัติตามเงื่อนไข  
 ของการหักล้างสัญญาณแบบผสมดังแสดงในหัวข้อที่ 3.3 คือจะเรียงจากผู้ใช้ที่มี post detection  
 SNR จากน้อยไปยังมาก

การนำเอาอัลกอริทึมที่ได้กล่าวข้างต้นนี้ มาประยุกต์ใช้ร่วมกับอัลกอริทึมที่อาศัย  
 วิธีการหักล้างสัญญาณแบบผสมนั้น จะสามารถทำได้โดยจะเริ่มหลังจากที่มีการหักล้างสัญญาณ  
 ของผู้ใช้รายอื่นออกจากสมการที่ (3-18) ในขั้นตอนการตัดสินบิตข้อมูลในสมการที่ (3-19) นั้นจะ  
 ทำกระบวนการ multiple slicing แทน โดยจะยังไม่ทำการตัดสินสัญญาณข้อมูล  $\hat{d}_k$  โดยทันทีแต่  
 จะเลือกสัญญาณที่เป็นตัวเลือกที่น่าจะเป็นไปได้ออกมา 2 ตัวแทน (กรณี S=2)



รูปที่ 3-9 กระบวนการ multiple slicing ในรอบที่ 1 กรณี S = 2 และ T = 2

$$(a_{1,1}, a_{1,2}) = \Phi(G_1 y_1') \dots\dots\dots(3-24)$$

จากสัญญาณตัวเลือกที่น่าจะเป็นไปได้ 2 ตัวนี้จะสามารถนำมาหาค่าระยะ  
 ความผิดพลาดที่สอดคล้องกันกับแต่ละตัวเลือก  $d_{1,1}$  และ  $d_{1,2}$  ดังนี้

$$d_{1,1} = \frac{|y_1 - a_{1,1}|^2}{\|w_1^T\|^2} \dots\dots\dots(3-25)$$

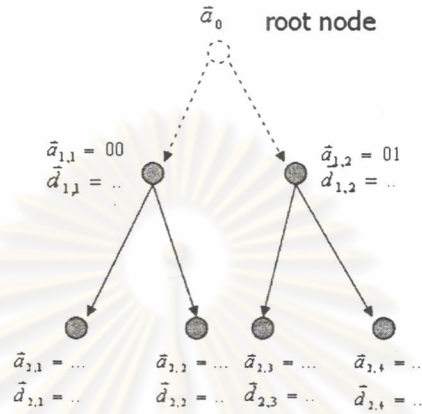
$$d_{1,2} = \frac{|y_1 - a_{1,2}|^2}{\|w_1^T\|^2} \dots\dots\dots(3-26)$$

หลังจากนั้นจะต้องทำการเปลี่ยนแปลงค่าสมาชิกของเวกเตอร์  $\bar{c}$  ให้สอดคล้อง  
 กับสัญญาณตัวเลือกที่น่าจะเป็นไปได้ด้วยตามสมการต่อไปนี้

$$\bar{c}_1^{(1,1)} = (1 - \alpha) a_{1,1} \dots\dots\dots(3-27)$$

$$\bar{c}_1^{(1,2)} = (1 - \alpha) a_{1,2} \dots\dots\dots(3-28)$$

จากนั้นจะนำมาหาสัญลักษณ์ตัวเลือกที่เป็นไปได้ของผู้ใช้คนถัดไป ซึ่งในกรณีนี้จะมีทั้งสิ้น 4 สัญลักษณ์ โดยที่ 2 สัญลักษณ์แรกเกิดจากกรณีที่สัญลักษณ์ของผู้ใช้รายแรก  $a_1 = a_{1,1}$  และ 2 สัญลักษณ์หลังเกิดจากกรณีที่  $a_1 = a_{1,2}$



รูปที่ 3-10 กระบวนการ multiple slicing ในรอบที่ 2 กรณี S = 2 และ T = 2

จากนั้นจะนำมาหาค่าระยะความผิดพลาดของสัญลักษณ์ตัวเลือกที่น่าจะเป็นไปได้ทั้ง 4 ตัวนี้ โดยที่ระยะความผิดพลาดของ node ในระดับชั้นบนที่สอดคล้องกับ node ในระดับชั้นล่างจะถูกนำมารวมเข้าด้วยกันกล่าวคือ

$$y'_{2,1} = \left[ y_2 - \sqrt{\frac{E}{M}} \bar{R}_2 \cdot \bar{c}^{(1,1)} \right] \dots\dots\dots(3-29)$$

$$(a_{2,1}, a_{2,2}) = \Phi(G_{11} y'_{2,1}) \dots\dots\dots(3-30)$$

$$y'_{2,2} = \left[ y_2 - \sqrt{\frac{E}{M}} \bar{R}_2 \cdot \bar{c}^{(1,2)} \right] \dots\dots\dots(3-31)$$

$$(a_{2,3}, a_{2,4}) = \Phi(G_{11} y'_{2,2}) \dots\dots\dots(3-32)$$

$$d_{2,1} = d_{1,1} + \frac{|y_2 - a_{2,1}|^2}{\|w_2^T\|^2} \dots\dots\dots(3-33)$$

$$d_{2,2} = d_{1,1} + \frac{|y_2 - a_{2,2}|^2}{\|w_2^T\|^2} \dots\dots\dots(3-34)$$

$$d_{2,3} = d_{1,2} + \frac{|y_2 - a_{2,3}|^2}{\|w_2^T\|^2} \dots\dots\dots(3-35)$$



$$d_{2,4} = d_{1,2} + \frac{|y_2 - a_{2,4}|^2}{\|w_2^T\|^2} \dots\dots\dots(3-36)$$

และหาเวกเตอร์  $\bar{c}$  ที่สอดคล้องกับสัญลักษณ์ตัวเลือกที่น่าจะเป็นไปได้แต่ละตัว ด้วยเพื่อนำไปใช้ในการป้อนไปยังผู้ใช้คนถัดไปดังสมการที่ (3-37) ถึง (3-40)

$$\bar{c}_2^{(2,1)} = (1-\alpha)a_{2,1} \dots\dots\dots(3-37)$$

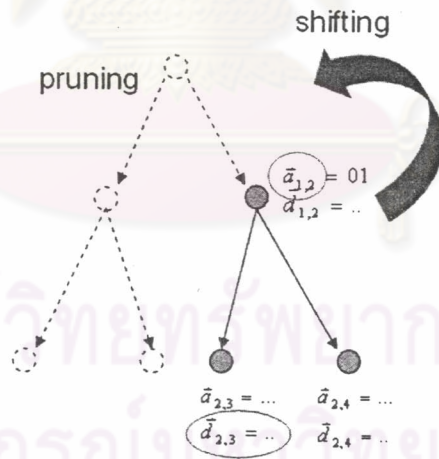
$$\bar{c}_2^{(2,2)} = (1-\alpha)a_{2,2} \dots\dots\dots(3-38)$$

$$\bar{c}_2^{(2,3)} = (1-\alpha)a_{2,3} \dots\dots\dots(3-39)$$

$$\bar{c}_2^{(2,4)} = (1-\alpha)a_{2,4} \dots\dots\dots(3-40)$$

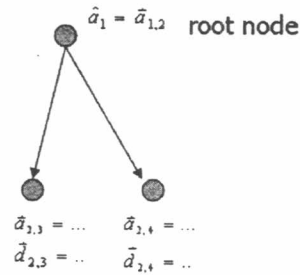
ซึ่งสมาชิกของเวกเตอร์  $\bar{c}$  ตัวอื่นนี้ก็จะต้องสอดคล้องกับ node ในระดับชั้นข้างบนของโครงสร้างแบบต้นไม้ด้วย

จากนั้นจะนำมาหา node ที่มีระยะความผิดพลาดต่ำสุด ในที่นี้สมมติว่าเป็น node  $a_{2,3}$  ดังนั้น node ที่อยู่ในระดับชั้นข้างบนของ node นี้ คือ node  $a_{1,2}$  จะเป็นสัญลักษณ์ข้อมูลที่ถูกตัดสินใจเลือกสำหรับผู้ใช้ที่สอดคล้องกับระดับชั้นนั้น ๆ หลังจากนั้นก็จะทำการทิ้งข้อมูลต่าง ๆ ของ node  $a_{1,1}$  ไปและ node ต่าง ๆ ที่อยู่ในกิ่งที่ต่อจาก node  $a_{1,1}$  นี้ออกไปด้วย



รูปที่ 3-11 การเล็มกิ่ง (pruning) ของ node ที่ไม่ต้องการทิ้ง

หลังจากเล็มกิ่ง (pruning) ของ node ที่ไม่ต้องการทิ้ง node  $a_{1,2}$  ที่ถูกตัดสินใจเลือกจะถูกนำมาเป็น root node ใหม่แทน



รูปที่ 3-12 การนำ node ที่เลือกมาเป็น root node ใหม่

โดยที่สัญลักษณ์ตัวเลือกที่นำจะเป็นไปได้ทั้ง 2 ตัว  $a_{2,3}$  และ  $a_{2,4}$  ที่เหลืออยู่ พร้อมทั้งระยะความผิดพลาด  $d_{2,3}$  และ  $d_{2,4}$  และเวกเตอร์  $\bar{c}$  ที่ใช้ในการป้อนไปข้างหน้า  $c_{2,3}$  และ  $c_{2,4}$  จะใช้ค่าเดิมที่ใช้ในการตัดสินใจสัญลักษณ์ข้อมูลในรอบที่แล้ว จึงไม่จำเป็นที่จะต้องนำมาคำนวณใหม่อีกครั้ง ซึ่งกระบวนการจะกลับมาเหมือนกับขั้นตอนแรกเริ่มในรูปที่ 3-9 การประมวลผลสำหรับตัดสินใจสัญลักษณ์ของผู้ใช้คนต่อ ๆ ไปก็จะเป็นไปในทำนองเดียวกันกับที่กล่าวมาข้างต้น จนกระทั่งเมื่อเหลือผู้ใช้จำนวน  $T-1$  คนที่ยังไม่ได้ทำการตัดสินใจสัญลักษณ์ข้อมูลออกมา ซึ่งจะมีจำนวนผู้ใช้เหลืออยู่ไม่เพียงพอตั้งที่ใช้ในกระบวนการข้างต้น ก็จะเลือกสัญลักษณ์ที่เหลือในเส้นทางจาก root node มายัง node ที่มีผลรวมระยะความผิดพลาดที่ต่ำที่สุดมาเป็นสัญลักษณ์ข้อมูลของผู้ใช้ที่เหลือจำนวน  $T-1$  คน [21]

ศูนย์วิทยทรัพยากร  
จุฬาลงกรณ์มหาวิทยาลัย