บทที่ 3

มัลติยูสเซอร์ดีเทกชันแบบป้อนกลับที่ใช้กระบวนการปรับอัตโนมัติชนิดบอด

เนื่องจากความซับซ้อนในการคำนวณเมตริกซ์ของมัลติยูสเซอร์ดีเทกชันแบบป้อนกลับจึงมีผู้เสนอให้ใช้กระบวนการปรับอัตโนมัติโดยมีการส่งเทรนนิ่งซีเควนซ์ช่วยดังที่กล่าวมาแล้วในบทที่ 2 แต่อย่างไรก็ตาม ปกติแล้วระบบ DS-CDMA เป็นระบบที่ไม่มีการส่งเทรนนิ่งซีเควนซ์ ดังนั้น การเพิ่มเทรนนิ่งซีเควนซ์เข้าไปในระบบจะต้องมีการเปลี่ยนแปลงเฟรมในการส่งข้อมูล นอกจากนั้นการที่ผู้ใช้ทุกคนในระบบต้องทำการส่งเทรนนิ่งซีเควนซ์ให้สถานีฐานทุกๆครั้งที่มีการเปลี่ยนแปลงในระบบ เช่นกรณีที่มีผู้ใช้เข้าออกระบบ จะก่อให้เกิดความสูญเปล่าในการรับส่งข้อมูล วิทยานิพนธ์ฉบับนี้จึงได้เสนอมัลติยูสเซอร์ดีเทกชันแบบป้อนกลับซึ่งมีกระบวนการปรับค่าน้ำหนักถ่วงแบบบอด(ไม่ต้องส่งเทรนนิ่งซีเควนซ์)แทน และจะเรียกเครื่องรับนี้สั้นๆว่ามัลติยูสเซอร์ดีเทกชันแบบป้อนกลับชนิดบอดตลอดทั้งวิทยานิพนธ์นี้

เนื้อหาในบทนี้จะกล่าวถึงการนำอัลกอริทึม bootstrap ซึ่งเป็นอัลกอริทึมปรับอัตโนมัติแบบบอดซึ่งถูกเสนอใช้ในดีคอรีเลเตอร์มาใช้ในกระบวนการปรับอัตโนมัติของวงจรกรองป้อนไปข้างหน้า และกล่าวถึงการประยุกต์ใช้อัลกอริทึม ISIC (ISI Cancellation) ซึ่งเป็นกระบวนการปรับอัตโนมัติแบบบอดซึ่งถูกเสนอใช้ในการแก้ไขปัญหา ISI (Intersymbol Interference) ในส่วนป้อนกลับของเครื่องรับแบบผู้ใช้เดียว (single user) มาใช้ในการปรับน้ำหนักถ่วงของวงจรกรองป้อนกลับของมัลติยูสเซอร์ดีเทกชันแบบป้อนกลับเพื่อกำจัด MAI ส่วนที่เหลือ

3.1 อัลกอริทึม bootstrap

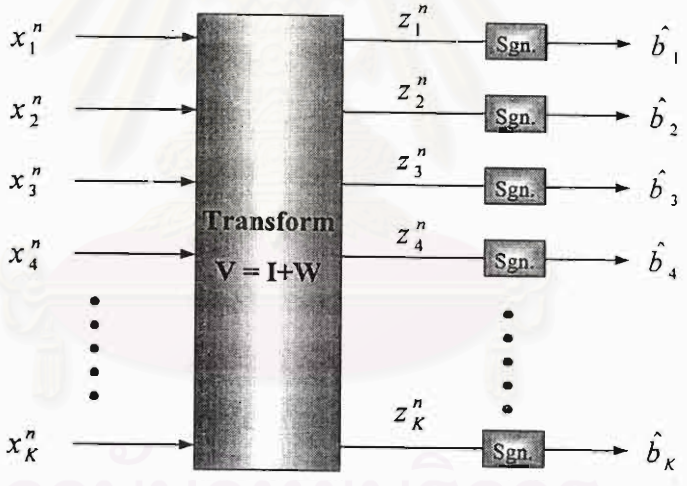
Y. Bar-Ness ได้เสนอการนำอัลกอริทึม bootstrap มาใช้กับเครื่องรับชนิดดีคอรีเลเตอร์เพื่อกำจัดสัญญาณแทรกสอดระหว่างผู้ใช้ (MAI) ในระบบ CDMA [17, 24, 25] โดยอัลกอริทึม bootstrap เป็นกระบวนการปรับอัตโนมัติแบบบอดที่ใช้ในการแยกแยะข้อมูลซึ่งเป็นอิสระจากกันแต่อยู่ปะปนกันออกจากกัน อัลกอริทึมนี้จะทำการกำจัดค่าสหสัมพันธ์ระหว่างข้อมูลที่ไม่ต้องการ(สัญญาณแทรกสอด)กับสัญญาณออก ทำให้ในที่สุดสัญญาณออกจะเป็นอิสระจากสัญญาณแทรกสอดนั้น

3.1.1 หลักการของอัลกอริทึม bootstrap

หากสัญญาณที่ป้อนเข้าเครื่องรับเกิดจากข้อมูลที่ต้องการและข้อมูลที่ไม่ต้องการซึ่งเป็นอิสระจากกันมาประกอบกันด้วยค่าสหสัมพันธ์ค่าหนึ่ง จะทำให้สัญญาณที่เข้ามาในเครื่องรับนั้นไม่เป็นอิสระจากสัญญาณที่ไม่ต้องการ หลักการของ bootstrap คือพยายามลดค่าสหสัมพันธ์ระหว่างสัญญาณออกของเครื่องรับนี้กับข้อมูลที่ไม่ต้องการทั้งหมด เพื่อให้สัญญาณออกเหลือแต่ข้อมูลที่ต้องการเท่านั้น ซึ่งทั้งหมดนี้สามารถเขียนอธิบายเป็นสมการได้ดังนี้

เมื่อให้ระบบมีผู้ใช้ K คน ข้อมูลของผู้ใช้ทุกคนที่ส่งออกมา ณ เวลา n เป็นเวกเตอร์ B_n ขนาด K×1 ซึ่งสมาชิก b_k, k = 1, ..., K แต่ละตัวเป็นอิสระต่อกัน, ค่าสหสัมพันธ์ระหว่างข้อมูลของผู้ใช้แต่ละคนเป็นเมตริกซ์ G ขนาด K×K ที่มีสมาชิก g_{ij}ในแนวเส้นทะแยงมุมเป็นหนึ่ง และถูกรบกวนด้วยสัญญาณรบกวนเกาส์เซียนแบบขาว M_n จะสามารถเขียนเวกเตอร์ของสัญญาณ X_n ขนาด K×1 ก่อนเข้าเครื่องรับได้เป็น

$$X_n = GB_n + M_n \tag{3-1}$$



รูปที่ 3.1 เครื่องแยกสัญญาณ (separator) ที่ใช้กระบวนการปรับอัตโนมัติโดยอัลกอริทึม bootstrap

หลักจากนั้นนำสัญญาณ X_n นี้ผ่านเข้าเครื่องรับ V ขนาด K×K ซึ่งโครงสร้างของเครื่องรับจะเป็นเมตริกซ์ I และ W บวกกันอยู่ตามรูปที่ 3.1 โดยแต่ละแถวของเมตริกซ์ V จะเสมือนเครื่องรับของผู้ใช้แต่ละคน นั่นคือเครื่องรับของผู้ใช้แต่ละคนขนาด 1×K จำนวน K เครื่องประกอบกันเป็นเครื่องรับ V และจะสามารถเขียนสมการหลังจากสัญญาณผ่านออกจากเครื่องรับได้ดังนี้

$$Z_n = VX_n = (I + W)X_n \tag{3-2}$$

โดยที่

$$\mathbf{V}=\begin{bmatrix} 1 & v_{1,2} & v_{1,3} & \cdots & v_{1,K} \\ v_{2,1} & 1 & v_{2,3} & & v_{2,K} \\ v_{3,1} & v_{3,2} & 1 & \ddots & \vdots \\ \vdots & & \ddots & \ddots & v_{K-1,K} \\ v_{K,1} & \cdots & \cdots & v_{K,K-1} & 1 \end{bmatrix}, \tag{3-3}$$

$$\mathbf{W}=\begin{bmatrix} 0 & w_{1,2} & w_{1,3} & \cdots & w_{1,K} \\ w_{2,1} & 0 & w_{2,3} & & w_{2,K} \\ w_{3,1} & w_{3,2} & 0 & \ddots & \vdots \\ \vdots & & \ddots & \ddots & w_{K-1,K} \\ w_{K,1} & \cdots & \cdots & w_{K,K-1} & 0 \end{bmatrix} \tag{3-4}$$

และ I เป็นเมตริกซ์เอกลักษณ์

จะสามารถเขียนสัญญาณออกจากเครื่องรับลำดับที่ k (แต่ละแถวของเมตริกซ์แทนเครื่องรับของผู้ใช้แต่ละคน)ได้เป็น

$$z_k^n = x_k^n + W_{out_k} X_{out_k}^n \tag{3-5}$$

โดยที่สัญลักษณ์ของ z_k^n และ x_k^n หมายถึงสมาชิกตัวที่ k ของเวกเตอร์ Z_n และ X_n ตามลำดับ ส่วนสัญลักษณ์ของ $Z_{out_k}^n$ และ $X_{out_k}^n$ นั้นหมายถึงเวกเตอร์ซึ่งประกอบด้วยสมาชิกทุกตัวของเวกเตอร์ Z_n และ X_n ซึ่งไม่มี z_k^n และ x_k^n อยู่ตามลำดับ และเวกเตอร์ W_{out_k} หมายถึงแถวที่ k ของเมตริกซ์ W ที่ไม่มีสมาชิก $w_{k,k}$ รวมอยู่ด้วย

จากตอนต้นที่กล่าวไปแล้วว่าหลักการของอัลกอริทึม bootstrap คือจะทำให้สัญญาณที่ออกจากเครื่องรับที่ต้องการ เป็นอิสระจากข้อมูลที่ไม่ต้องการทั้งหมด นั่นคืออัลกอริทึม bootstrap จะทำการลดค่าสหสัมพันธ์ระหว่างสัญญาณออกกับข้อมูลที่ไม่ต้องการทั้งหมด หมายความว่าถ้าพิจารณาให้ผู้ใช้ลำดับที่ k เป็นผู้ใช้ที่สนใจจะหมายถึงทำการลดค่าสหสัมพันธ์เพื่อให้ได้

$$E[z_k^n B_{out_k}^n] = 0, \qquad k = 1, \ldots, K \tag{3-6}$$

โดยที่ $B_{out_k}^n = [b_1^n, b_2^n, \cdots, b_{k-1}^n, b_{k+1}^n, b_{k+2}^n, \cdots, b_K^n]^T$ เป็นเวกเตอร์ข้อมูลของผู้ใช้ทุกคนที่ไม่ใช่คนที่ k หรือเป็นเวกเตอร์ B_n ที่ไม่มีสมาชิก b_k^n นั่นเอง อย่างไรก็ตามค่าของ $B_{out_k}^n$ จริงๆนั้นจะไม่สามารถทราบได้ อัลกอริทึม bootstrap จะประมาณค่า $B_{out_k}^n$ ด้วยบิตที่ตรวจจับได้จากสัญญาณออกจากเครื่องรับนี้ของผู้ใช้คนอื่นๆคือ $\operatorname{sgn}(Z_{out_k}^n)$ นั่นคือจะเป็นการปรับลด $E[z_k^n \operatorname{sgn}(Z_{out_k}^n)]$แทนการปรับลด $E[z_k^n B_{out_k}^n]$และเมื่อใช้สมการ steepest descent ในการ

ปรับลด $E[z_k^n \operatorname{sgn}(Z_{out_k}^n)]$ โดยใช้ค่าประมาณที่ขณะใดๆแทนค่าเฉลี่ยในที่สุดก็จะได้กระบวนการปรับอัตโนมัติ bootstrap เป็น

$$W_{out_k}^{n+1} = W_{out_k}^n - \mu z_k^n \operatorname{sgn}(Z_{out_k}^n)^T, \qquad k = 1, \ldots, K \tag{3-7}$$

3.1.2 ดีคอรีเลเตอร์แบบปรับตัวเองโดยใช้อัลกอริทึม bootstrap (Adaptive Bootstrap Decorrelator)

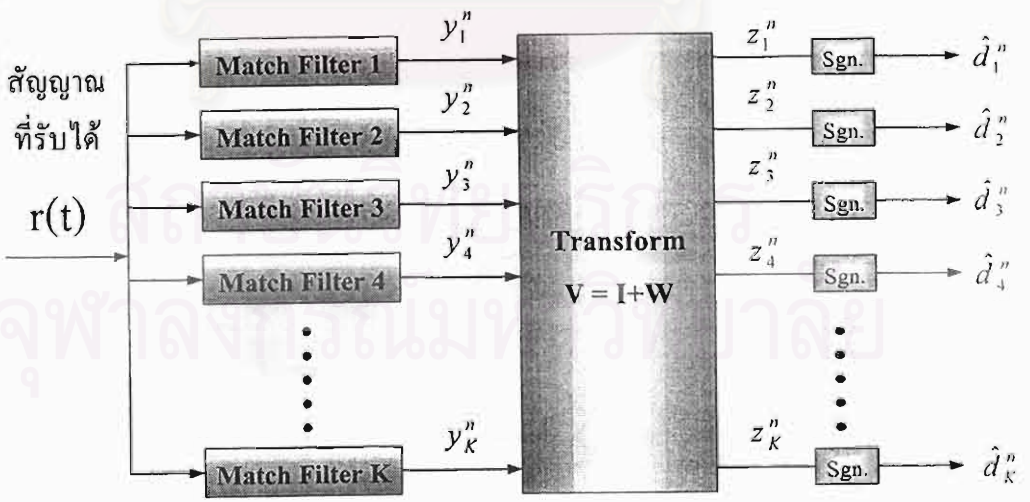
ดีคอรีเลเตอร์ของ Y. Bar-Ness ซึ่งใช้อัลกอริทึม bootstrap ในการปรับนั้น จะใช้สัญญาณออกจากเครื่องรับแบบแมตช์ของผู้ใช้ทุกๆคนเป็นสัญญาณเข้าของดีคอรีเลเตอร์ เนื่องจากสัญญาณออกของเครื่องรับแบบแมตช์จากสมการ (2-4) นั้นสามารถเขียนได้เป็น

$$Y_n = \Gamma \cdot \mathbf{A}_n \cdot D_n + M_n \tag{3-8}$$

ซึ่งจะพบว่าเลียนแบบอยู่กับสมการ (3-1)

$$X_n = \mathbf{G}B_n + M_n \tag{3-9}$$

ซึ่งเป็นสมการของสัญญาณเข้าที่จะใช้อัลกอริทึม bootstrap ทั้งนี้ Y. Bar-Ness พิจารณาว่าสัญญาณที่ต้องการคือ D_n และมองเมตริกซ์ความสัมพันธ์ G เป็น $\Gamma \cdot \mathbf{A}_n$ ดังนั้นกระบวนการปรับอัตโนมัติเพื่อลด $E[z_k^n D_{out_k}^n]$ จะเป็นตามสมการ (3-7) เช่นกัน ซึ่งเครื่องรับดีคอรีเลเตอร์แบบนี้สามารถแสดงได้ดังรูปที่ 3.2 และกระบวนการปรับอัตโนมัติให้กับน้ำหนักถ่วงของผู้ใช้ลำดับที่ k แสดงอยู่ในรูปที่ 3.3



รูปที่ 3.2 ดีคอรีเลเตอร์ที่ใช้กระบวนการปรับอัตโนมัติโดยอัลกอริทึม bootstrap

เมื่อกำลังของผู้ใช้ที่มารบกวนไม่สูงหรือกรณี SNR ของผู้ใช้ที่มารบกวนมีค่าต่ำ เครื่องรับนี้จะมีสมรรถนะดีกว่าดีคอรีเลเตอร์ของ Lupas ในบทที่ 2 ซึ่งเป็นดีคอรีเลเตอร์ที่ใช้ค่าน้ำหนักถ่วงคงที่ เนื่องจากในกรณีที่ค่า SNR ของผู้ใช้ที่มารบกวนต่ำนั้นเครื่องรับจะกำจัด MAI ได้ไม่หมดแต่สัญญาณรบกวนที่เกิดจากเครื่องรับนี้จะลดลง [17] ดังนั้นสมรรถนะจึงดีขึ้นเพราะเมื่อกำลังของผู้ใช้ที่มารบกวนต่ำความผิดพลาดส่วนใหญ่จะเกิดจากสัญญาณรบกวน และเครื่องรับนี้จะมีสมรรถนะคู่เข้าสู่ดีคอรีเลเตอร์ของ Lupas ในกรณีที่ SNR ของผู้ใช้ที่มารบกวนมีค่าสูงๆ ซึ่ง Y. Bar-Ness ได้พิสูจน์ไว้ใน [25] ว่าในสภาวะอยู่ตัวกรณีที่ผู้ใช้คนอื่นๆที่มารบกวนมี SNR สูงจนทำให้ประมาณค่า $E[z_k^n D_{out_k}^n]$ ด้วย $E[z_k^n \operatorname{sgn}(Z_{out_k}^n)]$ ได้แม่นยำ ค่าน้ำหนักถ่วงของเครื่องรับนี้จะคู่เข้าสู่ค่า

$$W_{out_k} = -(\Gamma_{out_k}\Gamma_{out_k}^{-1}) \tag{3-10}$$

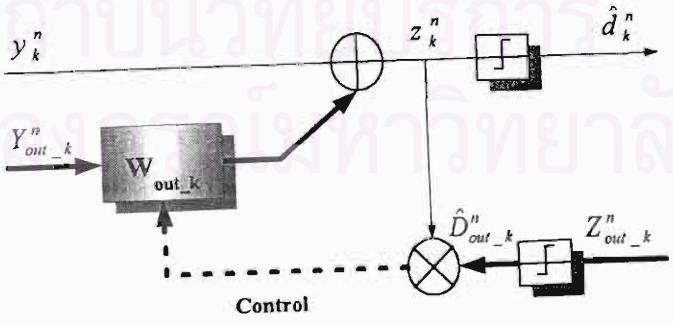
โดยที่ Γ_{out_k} หมายถึงเมตริกซ์ Γ ซึ่งไม่มีสมาชิกในแถวที่ k และคอลัมน์ที่ k อยู่ ในขณะที่ Γ_{out_k} เป็นแถวที่ k ของเมตริกซ์ Γ ซึ่งไม่มีสมาชิกตัวที่ k อยู่

สัญญาณออกจากดีคอรีเลเตอร์ที่สภาวะอยู่ตัวของผู้ใช้ลำดับที่ k สามารถเขียนได้เป็น

$$z_k^n = a_k^n(\Gamma_{k,k} - \Gamma_{out_k}\Gamma_{out_k}^{-1}\Gamma_{out_k}^T)d_k^n + \varsigma_k^n \tag{3-11}$$

โดยที่ ς_k^n หมายถึงสัญญาณรบกวนที่ผ่านออกมาจากเครื่องรับของผู้ใช้ลำดับที่ k

พบว่าในสมการ (3-11) จะเหลือเฉพาะข้อมูลที่ผู้ใช้ลำดับที่ k ส่งมาคือ d_k^n เท่านั้นไม่มีข้อมูลของผู้ใช้อื่นๆปนมา ซึ่งก็หมายถึงดีคอรีเลเตอร์กำจัด MAI ได้หมด และในกรณีที่ผู้ใช้คนอื่นๆมี SNR ต่ำๆนั้นการประมาณ $E[z_k^n D_{out_k}^n]$ จะถูกต้องน้อยลงทำให้เหลือ MAI หรือพจน์ที่มี $D_{out_k}^n$ อยู่บางส่วน แต่จะให้สัญญาณรบกวนน้อยกว่าดีคอรีเลเตอร์ของ Lupas ทำให้ดีคอรีเลเตอร์ที่ใช้อัลกอริทึม bootstrap แม้จะหักล้าง MAI ได้ไม่หมดในกรณีที่ SNR ของผู้ใช้อื่นๆที่มากวนมีค่าต่ำ ก็ยังคงให้ความผิดพลาดเนื่องจากผลรวมของการรบกวนจาก MAI ที่เหลือนี้กับสัญญาณรบกวนต่ำกว่าดีคอรีเลเตอร์ของ Lupas



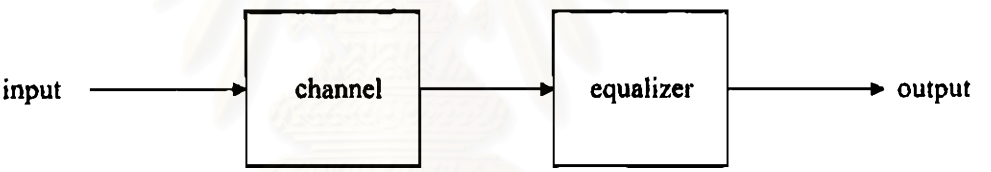
รูปที่ 3.3 กระบวนการปรับอัตโนมัติให้กับน้ำหนักถ่วงของดีคอรีเลเตอร์ของผู้ใช้ลำดับที่ k โดยอัลกอริทึม bootstrap

3.2 อัลกอริทึม ISIC (ISI Cancellation Algorithm) และหลักการลด MOE (Mean Output Energy)

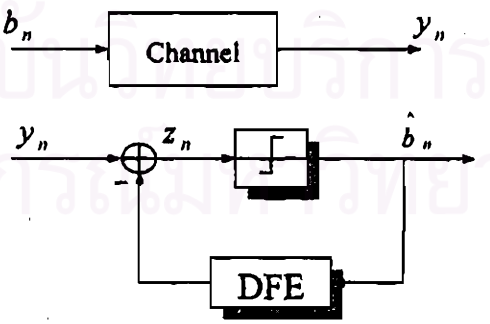
อัลกอริทึม ISIC เป็นวิธีการปรับอัตโนมัติซึ่งถูกเสนอขึ้นมาใช้กับอิควอไลเซอร์แบบป้อนกลับชนิดผู้ใช้เดียว (single user) โดย Sylvie [26] เมื่อปีค.ศ. 1995 เพื่อกำจัดสัญญาณแทรกสอดระหว่างสัญลักษณ์ในส่วนที่เกิดจากสัญลักษณ์ที่ตรวจจับได้แล้ว (causal Intersymbol Interference, causal ISI) อัลกอริทึมนี้ใช้หลักการในการลดค่าเฉลี่ยกำลังสองของสัญญาณออกหรือเรียกว่า MOE เพื่อกำจัด ISI และกระบวนการปรับอัตโนมัตินี้ไม่ต้องอาศัยเทรนนิ่งซีเควนซ์

3.2.1 อิควอไลเซอร์และ ISI

ISI เกิดจากช่องสัญญาณส่งผ่านความถี่ต่าง ๆ ด้วยความเร็วและเฟสต่างกัน หรือเกิดในช่องสัญญาณหลายวิถี ทำให้เมื่อส่งข้อมูลไปแล้วเกิดการทับซ้อน (overlap) ของบิตข้างเคียงกันทำให้สัญญาณที่รับได้ผิดพลาดไป วิธีการที่ใช้ในการแก้ไขปัญหานี้คือนำสัญญาณที่รับได้มาผ่านอิควอไลเซอร์ก่อนตรวจจับบิตข้อมูล ดังรูปที่ 3.4



รูปที่ 3.4 แผนภาพในการนำอิควอไลเซอร์มาแก้ปัญหา ISI



รูปที่ 3.5 อิควอไลเซอร์แบบป้อนกลับ

อิควอไลเซอร์เป็นวงจรกรองแบบปรับตัวได้ซึ่งสามารถนำอัลกอริทึมต่าง ๆ เช่น LMS หรือ ISIC มาปรับค่าน้ำหนักถ่วงภายในเพื่อชดเชยหรือหักล้าง ISI สำหรับโครงสร้างของอิควอไลเซอร์แบบป้อนกลับ (Decision Feedback Equalizer, DFE) ของ Sylvie ซึ่งใช้ในการหักล้าง ISI ที่เกิดจากบิตก่อนหน้าบิตปัจจุบันโดยนำเอาบิตข้อมูลที่ตรวจจับได้แล้ว ณ เวลาก่อนๆ ป้อนกลับไปเพื่อหักล้างการรบกวนเนื่องจากบิตข้อมูลเหล่านั้นต่อบิตข้อมูลปัจจุบัน แสดงอยู่ในรูป 3.5

3.2.2 หลักการของอัลกอริทึม ISIC

Sylvie ได้เสนอหลักการของอัลกอริทึม ISIC ไว้ว่า เมื่อตั้งสมมุติฐานให้การส่งสัญญาณแต่ละบิตในระบบดิจิตอลเป็นอิสระจากกันและค่าเฉลี่ยของบิตเป็น 0 จะพบว่าสัญญาณเข้าส่วนป้อนกลับซึ่งคือบิตข้อมูลในอดีตจะเป็นอิสระจากบิตข้อมูลที่ต้องการ ณ เวลาปัจจุบัน ดังนั้น ถ้าให้สัญญาณหลังการหักล้าง ISI เป็น z และบิตข้อมูลที่ส่งมาเป็น b ซึ่งมีค่าเป็น 1 หรือ –1 จะพบว่าการปรับลด MSE (Mean Square Error) ของสัญญาณออกจากอิควอไลเซอร์แบบป้อนกลับ $E[(b_n - z_n)^2]$ จะเหมือนกับการปรับลดค่าเฉลี่ยกำลังสองของสัญญาณออกจากอิควอไลเซอร์แบบป้อนกลับซึ่งเรียกว่า Mean Ouput Energy (MOE) $E[z_n^2]$ เนื่องจาก

$$\begin{aligned} MSE &= E[(b_n - z_n)^2] \\ &= E(b_n)^2 - 2E(b_n z_n) + E(z_n)^2 \\ &= 1 - 2E(b_n(y_n - G\hat{B}_{before_n})) + E(z_n)^2 \end{aligned} \tag{3-12}$$

โดยที่ $\hat{B}_{before_n} = [\hat{b}_{n-1}\ \hat{b}_{n-2} \cdots \hat{b}_{n-N}]^T$ เป็นสัญญาณเข้าส่วนป้อนกลับซึ่งเป็นเวกเตอร์ของบิตข้อมูลที่ตรวจจับได้แล้วของบิตที่ก่อนหน้าบิตปัจจุบันทั้งหมด N บิต และ G เป็นเวกเตอร์ของค่าน้ำหนักถ่วงในส่วนป้อนกลับมีขนาด 1×N ประกอบด้วยสมาชิก g_k, k = 1, ..., N โดยที่ N เป็นจำนวนน้ำหนักถ่วงของส่วนป้อนกลับทั้งหมดซึ่งมักจะมีค่าไม่เกินจำนวนเส้นทางของช่องสัญญาณลบหนึ่ง

$$\begin{aligned} \frac{\partial MSE}{\partial G} &= \frac{\partial}{\partial G}\{1 - 2E(b_n(y_n - GB_{before_n})) + E(z_n)^2\} \\ \frac{\partial MSE}{\partial G} &= 0 - 2E(b_n B_{before_n}) + \frac{\partial E(z_n)^2}{\partial G} \end{aligned} \tag{3-13}$$

และจากตอนต้นสัญญาณเข้าส่วนป้อนกลับ $\hat{B}_{before_n}$ เป็นอิสระจากข้อมูลที่ต้องการ b_n ดังนั้น

$E(b_n B_{before_n}) = 0$

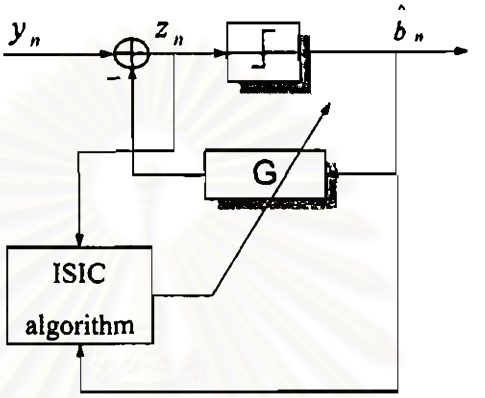
จะได้

$$\frac{\partial MSE}{\partial G} = \frac{\partial E(z_n)^2}{\partial G} = \frac{\partial MOE}{\partial G} \tag{3-14}$$

ดังนั้นการปรับลด MSE ก็เหมือนการปรับลด MOE ในกรณีที่สัญญาณเข้าส่วนป้อนกลับเป็นอิสระจากข้อมูลที่ต้องการ และเมื่อใช้สมการ stochastic gradient descent ในการปรับลดค่า MOE นี้ ในที่สุดก็จะได้กระบวนการปรับอัตโนมัติ ISIC ออกมาตามสมการ (3-15)

$$g_k^{n+1} = g_k^n + \mu(z_n \cdot \hat{b}_{n-k}) \tag{3-15}$$

บล็อกไดอะแกรมของอิควอไลเซอร์แบบป้อนกลับที่ใช้กระบวนการปรับอัตโนมัติโดยอัลกอริทึม ISIC แสดงอยู่ในรูปที่ 3.6



รูปที่ 3.6 บล็อกไดอะแกรมของอิควอไลเซอร์ที่ใช้กระบวนการปรับอัตโนมัติโดยอัลกอริทึม ISIC

สำหรับอัลกอริทึม ISIC นี้ สัญญาณเข้าส่วนป้อนกลับจะเป็นค่าตัดสินบิตของบิตข้อมูลซึ่งถูกส่งมาก่อนบิตที่จะถูกตรวจจับในปัจจุบัน แต่สำหรับกระบวนการปรับที่เสนอใช้กับมัลติยูสเซอร์ดีเทกชันแบบป้อนกลับในวิทยานิพนธ์นี้จะป้อนค่าตัดสินบิตของผู้ใช้ลำดับก่อนหน้าผู้ใช้ที่พิจารณาซึ่งถูกส่งมา ณ เวลาเดียวกันกับบิตที่จะตรวจจับในปัจจุบันเข้าส่วนป้อนกลับแทนโดยอาศัยการหน่วงเวลารอ ดังนั้นโครงสร้างในการใช้กระบวนการปรับอัตโนมัติจะต่างออกไป รวมทั้งน้ำหนักถ่วงจะเป็นเมตริกซ์ 2 มิติแทนที่จะเป็นเวกเตอร์มิติเดียวดังกรณีอิควอไลเซอร์แบบป้อนกลับ ซึ่งจะได้กล่าวในหัวข้อต่อไป อย่างไรก็ตามยังคงใช้หลักการที่ว่าเมื่อสัญญาณเข้าในส่วนป้อนกลับเป็นอิสระจากข้อมูลที่ต้องการแล้วจะสามารถปรับค่าน้ำหนักถ่วงเพื่อให้ได้ข้อมูลที่ต้องการโดยการปรับลด MOE ได้

3.3 มัลติยูสเซอร์ดีเทกชันแบบป้อนกลับชนิดบอด

(Blind Adaptive Decorrelating Decision Feedback Multiuser Detection)

วิทยานิพนธ์นี้เสนอการนำกระบวนการปรับอัตโนมัติแบบบอดมาใช้แก้ไขปัญหาความซับซ้อนในการหา cholesky decomposition การอินเวอร์สเมตริกซ์ รวมถึงกำจัดการใช้ค่าประมาณขนาดของสัญญาณในวงจรกรองป้อนกลับ

นั่นคือวิทยานิพนธ์นี้ใช้หลักการในการแก้ปัญหาด้วยการปรับค่าน้ำหนักถ่วงของวงจรกรองทั้งสองโดยอัตโนมัติเช่นเดียวกับ P. Seite ในบทที่แล้ว แต่กระบวนการในการปรับอัตโนมัติของวิทยานิพนธ์นี้จะเป็นเงื่อนไขและวิธีการที่ต่างออกไปเพื่อให้สามารถปรับน้ำหนักถ่วงโดยไม่อาศัยเทรนนิ่งซีเควนซ์ได้ สำหรับ P. Seite จะทำการปรับอัตโนมัติน้ำหนักถ่วงของวงจรกรองทั้งสองเพื่อลดค่าเฉลี่ยกำลังสองของค่าผิดพลาดระหว่างข้อมูลจริงกับสัญญาณออกที่ได้ตามสมการ (2-27) โดยใช้อัลกอริทึม LMS แต่ในวิทยานิพนธ์นี้จะใช้กระบวนการปรับอัตโนมัติที่ต่างกันสำหรับวงจรกรองป้อนไปข้างหน้าและป้อนกลับ โดยจะปรับวงจรกรองป้อนไปข้างหน้าเพื่อลดค่าสหสัมพันธ์ระหว่างบิตข้อมูลที่ประมาณได้ของผู้ใช้ที่มีลำดับสูงกว่าผู้ใช้ที่พิจารณากับสัญญาณออกของผู้ใช้ที่พิจารณาโดยใช้อัลกอริทึม bootstrap และปรับวงจรกรองป้อนกลับเพื่อลดค่าเฉลี่ยกำลังของสัญญาณออก ซึ่งเป็นหลักการเดียวกันกับอัลกอริทึม ISIC

3.3.1 แนวคิด

พิจารณาโครงสร้างของมัลติยูสเซอร์แบบป้อนกลับนี้พบว่าประกอบด้วย 2 ส่วนคือส่วนวงจรกรองป้อนไปข้างหน้ากับส่วนวงจรกรองป้อนกลับ ซึ่งวงจรกรองทั้งสองทำหน้าที่แยกกัน วงจรกรองป้อนไปข้างหน้ากำจัด MAI จากผู้ใช้ที่มีลำดับสูงกว่าผู้ใช้ที่พิจารณา ส่วนวงจรกรองป้อนกลับกำจัด MAI ส่วนที่เหลือทั้งหมด ดังนั้นในการพิจารณาจะพิจารณาแยกกัน

3.3.1.1 วงจรกรองป้อนไปข้างหน้า

วงจรกรองป้อนไปข้างหน้าจะทำหน้าที่กำจัดการรบกวนจากผู้ใช้ที่มีลำดับสูงกว่าผู้ใช้ที่พิจารณา หมายถึงถ้าวงจรกรองได้รับการปรับอย่างถูกต้องสัญญาณออกจะต้องไม่มีการรบกวนจากผู้ใช้ที่มีลำดับสูงกว่าผู้ใช้ที่พิจารณา และถ้าพิจารณาความจริงที่ว่าข้อมูลของผู้ใช้แต่ละคนไม่ขึ้นแก่กัน จะหมายความว่าถ้าปรับค่าน้ำหนักถ่วงของวงจรกรองในส่วนนี้ถูกต้องสัญญาณออกของผู้ใช้ที่ต้องการจะต้องไม่มีค่าสหสัมพันธ์กับข้อมูลของผู้ใช้ที่มีลำดับสูงกว่า ทำให้สามารถใช้เงื่อนไข

ค่าสหสัมพันธ์นี้ในการปรับวงจรกรองป้อนไปข้างหน้าได้ และกระบวนการแบบบอดที่ใช้ในการปรับให้ค่าสหสัมพันธ์นี้เป็น 0 คืออัลกอริทึม bootstrap ดังที่ได้กล่าวมาในหัวข้อที่แล้ว ดังนั้นจะสามารถนำอัลกอริทึมนี้มาใช้ปรับวงจรกรองป้อนไปข้างหน้าได้ โดยอัลกอริทึม bootstrap ที่ถูกเสนอใช้ในมัลติยูสเซอร์ดีเทกชันแบบดีดอรีเลเตอร์จะกำจัดค่าสหสัมพันธ์ระหว่างสัญญาณที่ต้องการกับข้อมูลของผู้ใช้คนอื่นทุกคน แต่ในกรณีนี้เป็นการนำมาใช้ในการกำจัดค่าสหสัมพันธ์ของสัญญาณที่ต้องการกับข้อมูลของผู้ใช้ที่มีลำดับสูงกว่าเท่านั้น

ดังนั้นจะนำเอาสัญญาณออกจากเครื่องรับแบบแมตช์หลังจากผ่านการเรียงลำดับตามความแรงที่มาถึงแล้วมาผ่านเข้าเครื่องรับ V ซึ่งโครงสร้างของเครื่องรับจะเป็นเมตริกซ์ I และ W บวกกันอยู่ตามรูปที่ 3.7 ซึ่งแถวแต่ละแถวของเมตริกซ์ V นั้นสามารถมองได้ว่าเป็นวงจรกรองป้อนไปข้างหน้าของผู้ใช้แต่ละคน หรือกล่าวได้ว่าเมตริกซ์ V ประกอบขึ้นจากวงจรกรองป้อนไปข้างหน้าทั้งหมด K ชุด และจะสามารถเขียนสมการหลังจากสัญญาณผ่านออกจากเครื่องรับได้ดังนี้

$$Z_n = \mathbf{V}Y_n = (\mathbf{I} + \mathbf{W})Y_n \tag{3-16}$$

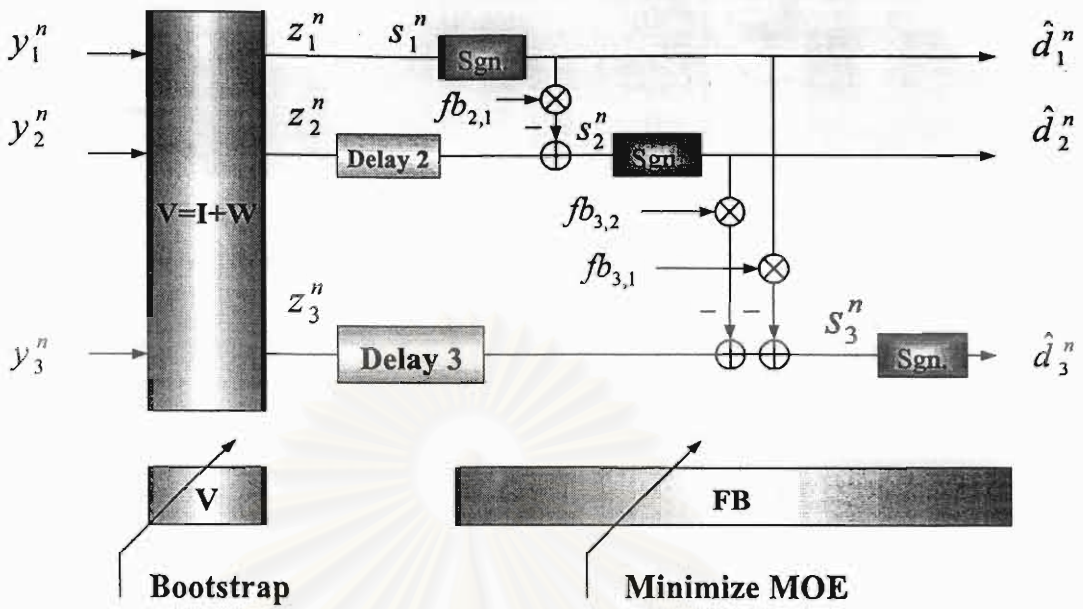
โดยที่

$$\mathbf{V} = \begin{bmatrix} 1 & w_{1,2} & w_{1,3} & \cdots & w_{1,K} \\ 0 & 1 & w_{2,3} & & w_{2,K} \\ 0 & 0 & 1 & \ddots & \vdots \\ \vdots & & \ddots & \ddots & w_{K-1,K} \\ 0 & \cdots & \cdots & 0 & 1 \end{bmatrix}, \tag{3-17}$$

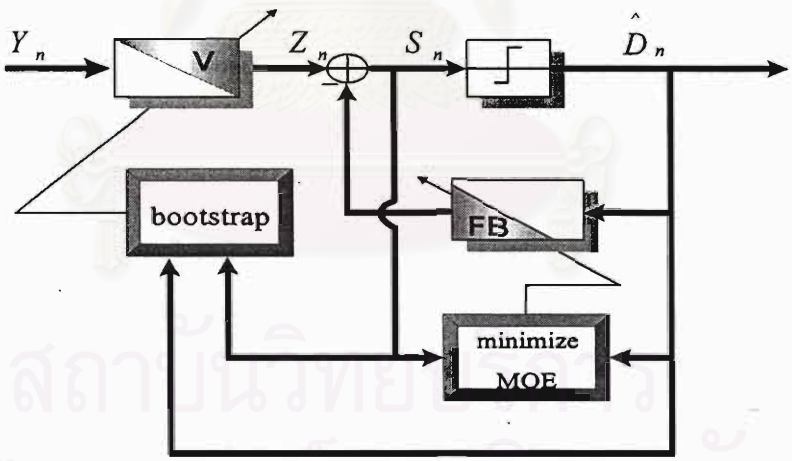
$$\mathbf{W} = \begin{bmatrix} 0 & w_{1,2} & w_{1,3} & \cdots & w_{1,K} \\ 0 & 0 & w_{2,3} & & w_{2,K} \\ 0 & 0 & 0 & \ddots & \vdots \\ \vdots & & \ddots & \ddots & w_{K-1,K} \\ 0 & \cdots & \cdots & 0 & 0 \end{bmatrix} \tag{3-18}$$

และ I เป็นเมตริกซ์เอกลักษณ์

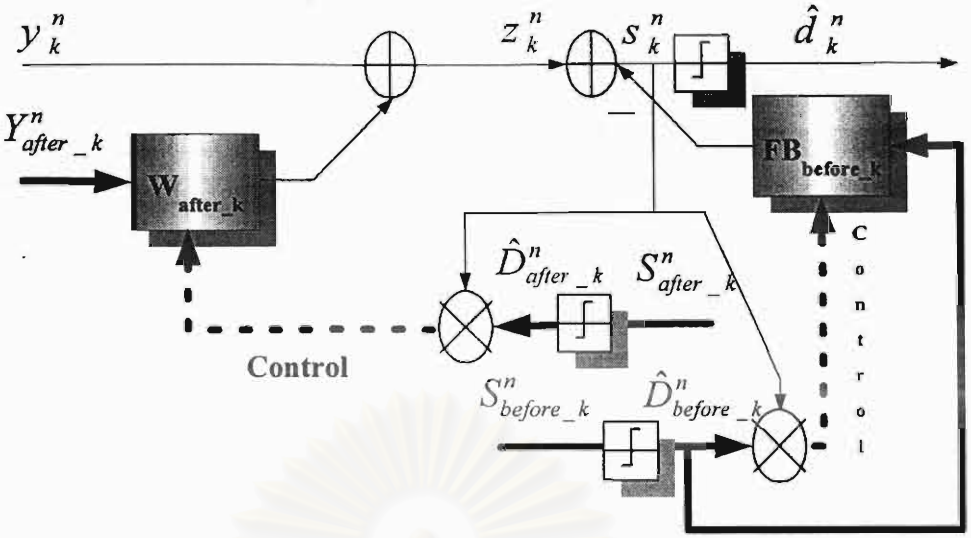
สังเกตว่าเนื่องจากในกรณีนี้ bootstrap กำจัดเฉพาะ MAI ซึ่งเกิดจากผู้ใช้ที่มีลำดับสูงกว่าผู้ใช้ที่พิจารณาเท่านั้น ดังนั้นเมตริกซ์ W ในสมการ (3-18) จึงมีสมาชิกเฉพาะเหนือแนวเส้นทะแยงมุมเท่านั้นซึ่งต่างจากกรณีดีดอรีเลเตอร์ของ Y. Bar-Ness ซึ่ง W ในสมการ (3-4) มีสมาชิกทั่วทั้งเมตริกซ์ยกเว้นแนวทะแยงมุม หรือกล่าวได้ว่าสำหรับในกรณีนี้สัญญาณที่ใช้ใน V จะเป็นสัญญาณออกจากเครื่องรับแบบแมตช์ของผู้ใช้ที่พิจารณาและของผู้ใช้ที่มีลำดับสูงกว่าผู้ใช้ที่พิจารณาเท่านั้น



รูปที่ 3.7 โครงสร้างของมัลติยูสเซอร์ดีเทกชันแบบป้อนกลับชนิดบอดในกรณีผู้ใช้ 3 คน



รูปที่ 3.8 โครงสร้างของกระบวนการปรับอัตโนมัติ
ของมัลติยูสเซอร์ดีเทกชันแบบป้อนกลับชนิดบอด



รูปที่ 3.9 กระบวนการปรับน้ำหนักถ่วงของผู้ใช้ลำดับที่ k
สำหรับมัลติยูสเซอร์ดีเทกชันแบบป้อนกลับชนิดบอด

สัญญาณออกจากวงจรกรองป้อนไปข้างหน้าของผู้ใช้ลำดับที่ k จะเขียนได้เป็น

$$z_k^n = y_k^n + W_{after_k} Y_{after_k}^n \tag{3-19}$$

โดยที่สัญลักษณ์ z_k^n และ y_k^n หมายถึง สมาชิกตัวที่ k ของเวกเตอร์ Z_n และ Y_n ตามลำดับ ส่วนสัญลักษณ์ $Z_{after_k}^n$ และ $Y_{after_k}^n$ หมายถึง เวกเตอร์ซึ่งประกอบด้วยสมาชิกทุกตัวของเวกเตอร์ Z_n และ Y_n หลังจากตัวที่ k เป็นต้นไป และเวกเตอร์ W_{after_k} หมายถึงแถวที่ k ของเมตริกซ์ W โดยที่ไม่มีสมาชิกตั้งแต่ตัวที่ $w_{k,1}$ ถึงตัวที่ $w_{k,k}$ รวมอยู่ด้วย

จากตอนต้นที่กล่าวไปแล้วว่าหลักการของอัลกอริทึม bootstrap จะทำให้สัญญาณที่ออกจากเครื่องรับนั้นเป็นอิสระจากข้อมูลที่ไม่ต้องการ นั่นคืออัลกอริทึม bootstrap จะทำการลดค่าสหสัมพันธ์ระหว่างสัญญาณออกกับข้อมูลที่ไม่ต้องการทั้งหมด ถ้าพิจารณาผู้ใช้ลำดับที่ k ข้อมูลที่ไม่ต้องการในขณะนี้คือข้อมูลจากผู้ใช้ที่มีลำดับสูงกว่าผู้ใช้ลำดับที่ k ทั้งหมด ดังนั้นในกรณีนี้คือทำการลดค่าสหสัมพันธ์เพื่อให้ได้

$$E[z_k^n D_{after_k}^n] = 0, \qquad k = 1,\ldots,K \tag{3-20}$$

โดยที่ $D_{after_k}^n = [d_{k+1}^n d_{k+2}^n \cdots d_K^n]^T$

แต่เนื่องจากเครื่องรับนี้ประกอบไปด้วยวงจรกรอง 2 ตัวด้วยกันและสัญญาณออกของมัลติยูสเซอร์ดีเทกชันจริงๆจะเป็นส่วนที่หักลบด้วยส่วนป้อนกลับไปแล้วคือ s_k^n ไม่ใช่ z_k^n ดังนั้นจะทำการปรับเพื่อให้ได้

$$E[s_k^n D_{after_k}^n] = 0, \qquad k = 1, ..., K \tag{3-21}$$

และทำนองเดียวกันการประมาณ $D_{after_k}^n$ ด้วย $\text{sgn}(Z_{after_k}^n)$ จะถูกเปลี่ยนเป็นประมาณ $D_{after_k}^n$ ด้วย $\text{sgn}(S_{after_k}^n)$ ซึ่งจะเชื่อถือได้มากกว่าเนื่องจาก S นั้นเป็นสัญญาณที่มีการหักล้าง MAI ทั้งหมดแต่ Z เป็นสัญญาณที่มีการหักล้าง MAI บางส่วนเท่านั้น สรุปแล้วก็คือจะใช้การปรับลด $E[s_k^n \text{sgn}(S_{after_k}^n)]$ แทน $E[z_k^n D_{after_k}^n]$ และเมื่อทำการปรับลด $E[s_k^n \text{sgn}(S_{after_k}^n)]$ ด้วยสมการ steepest descent โดยประมาณค่าเฉลี่ยด้วยค่าที่ขณะใดๆ ในที่สุดก็จะได้วิธีการปรับอัตโนมัติ bootstrap สำหรับวงจรกรองป้อนไปข้างหน้าของผู้ใช้ลำดับที่ k เป็น

$$W_{after_k}^{n+1} = W_{after_k}^n - \mu s_k^n \text{sgn}(S_{after_k}^n)^T, \qquad k = 1, ..., K \tag{3-22}$$

เมื่อพิจารณาที่สภาวะอยู่ตัวในกรณีที่ผู้ใช้ลำดับสูงกว่าผู้ใช้ที่พิจารณามีค่า SNR สูงพอสมควรซึ่งทำให้ $E[s_k^n \text{sgn}(S_{after_k}^n)] = E[s_k^n D_{after_k}^n] = 0$ นั้น จะสามารถพิสูจน์ได้ว่าค่าน้ำหนักถ่วงของวงจรกรองป้อนไปข้างหน้าจะลู่เข้าสู่ค่าตามสมการ (3-23) (สำหรับรายละเอียดของการพิสูจน์ได้แสดงอยู่ในภาคผนวก ค)

$$W_{after_k} = -(\Gamma_{after_k} \Gamma_{after_k}^{-1}) \tag{3-23}$$

โดยที่ Γ_{after_k} หมายถึง เมตริกซ์ซึ่งประกอบด้วยสมาชิกหลังจากแถวที่ k และคอลัมน์ที่ k ของเมตริกซ์ Γ เป็นต้นไป ในขณะที่ Γ_{after_k} เป็นแถวที่ k ของเมตริกซ์ Γ ซึ่งมีสมาชิกหลังจากตัวที่ k เป็นต้นไป

และจากภาคผนวก ค สัญญาณออกจากวงจรกรองป้อนไปข้างหน้าที่สภาวะอยู่ตัวของผู้ใช้ลำดับที่ k สามารถเขียนได้เป็น

$$\begin{aligned} z_k^n = {} & a_k^n (\Gamma_{k,k} - \Gamma_{after_k} \Gamma_{after_k}^{-1} \Gamma_{after_k}^T) d_k^n \\ & + (\Gamma_{before_k} - \Gamma_{after_k} \Gamma_{after_k}^{-1} \Gamma_{after_k \times before_k}) \mathbf{A}_{before_k}^n D_{before_k}^n \\ & + \xi_k^n \end{aligned} \tag{3-24}$$

โดยที่ ξ_k^n หมายถึงสัญญาณรบกวนที่ผ่านออกมาจากวงจรกรองป้อนไปข้างหน้าของผู้ใช้ลำดับที่ k ซึ่งมีค่าสหสัมพันธ์ตัวเองเป็น $E(\xi_k^2) = (\Gamma_{k,k} - \Gamma_{after_k} \Gamma_{after_k}^{-1} \Gamma_{after_k}^T)\sigma^2$

และ $\Gamma_{after_k \times before_k}$ หมายถึงเมตริกซ์ Γ ซึ่งมีสมาชิกเฉพาะหลังแถวที่ k และก่อนหน้าคอลัมน์ที่ k

จากสมการ (3-24) พบว่า MAI จากผู้ใช้ที่มีลำดับสูงกว่าผู้ใช้ที่พิจารณาหรือเทอมที่มี $d_j^n, j > k$ นั้นถูกกำจัดไปแล้วเมื่อผ่านออกมาจากวงจรกรองป้อนไปข้างหน้า (เหลือแต่ส่วนของข้อมูลผู้ใช้ที่ต้องการ d_k^n และส่วนของข้อมูลผู้ใช้ที่มีลำดับต่ำกว่าผู้ใช้ที่ต้องการ $D_{before_k}^n$ เท่านั้น)

3.3.1.2 วงจรกรองป้อนกลับ

จากหลักการของมัลติยูสเซอร์ดีเทกชันแบบป้อนกลับของ A. Duel-Hallen นั้นจะใช้ส่วนป้อนกลับนี้เป็นส่วนกำจัด MAI ซึ่งมาจากผู้ใช้ที่มีลำดับต่ำกว่าผู้ใช้ที่ต้องการ โดยนำข้อมูลที่ตัดสินบิตไปแล้วของผู้ใช้ที่มีลำดับต่ำกว่าป้อนกลับมาหักล้าง MAI ให้กับผู้ใช้ที่กำลังพิจารณาอยู่ และจากสมการ (3-24) ซึ่งเป็นสมการสัญญาณออกจากวงจรกรองป้อนไปข้างหน้าเมื่อหักล้าง MAI จากผู้ใช้คนที่มีลำดับสูงกว่าหมดไปเหลือแต่ MAI จากผู้ใช้ที่มีลำดับต่ำกว่า พบว่าถ้าส่วนป้อนกลับสามารถหักล้าง MAI ส่วนที่เหลือได้หมด สัญญาณที่ผ่านส่วนป้อนกลับแล้วจะเหลือเฉพาะพจน์แรกของสมการ (3-24) เท่านั้น คือจะเหลือเป็นค่าคงที่คูณกับข้อมูลที่ส่งมา ดังนี้

$$s_k'' = a_k^n(\Gamma_{k,k} - \Gamma_{after_k}\Gamma_{after_k}^{-1}\Gamma_{after_k}^{T})d_k^n + \xi_k'' \tag{3-25}$$

ดังนั้น ในการปรับน้ำหนักถ่วงของวงจรกรองป้อนกลับจะใช้หลักในการปรับให้ scaled mean-square error (SMSE) ซึ่งมีนิยามเป็นค่าเฉลี่ยกำลังสองของค่าผิดพลาดระหว่างผลคูณของค่าคงที่กับข้อมูลกับสัญญาณออกก่อนเข้าตัดสินบิต (E{ ค่าคงที่×ข้อมูลที่ส่งมา – สัญญาณออกก่อนเข้าตัดสินบิต }^{2}) มีค่าต่ำสุด หรือทำให้สัญญาณออกใกล้เคียงสมการ (3-25) มากที่สุดนั่นเอง

ถ้ากำหนดให้

$$\alpha_k = (\Gamma_{k,k} - \Gamma_{after_k}\Gamma_{after_k}^{-1}\Gamma_{after_k}^{T}) \tag{3-26}$$

จะสามารถเขียนสมการของค่าคงที่ได้เป็น

$$\beta_k = \alpha_k a_k^n \tag{3-27}$$

ดังนั้น สามารถเขียนสมการ SMSE ของผู้ใช้ลำดับที่ k ได้เป็น

$$SMSE_k = E[(\beta_k d_k^n - s_k^n)^2] \tag{3-28}$$

อย่างไรก็ตามค่าของข้อมูลที่ส่งมานั้นจะสามารถทราบได้เมื่อมีการส่งเทรนนิ่งซีเควนซ์เท่านั้น แต่ในที่นี้ไม่ต้องการส่งเทรนนิ่งซีเควนซ์ ดังนั้นจึงไม่ทราบค่าข้อมูลนี้ และไม่สามารถปรับลดค่า SMSE โดยใช้สมการ (3-28) ในการปรับค่าน้ำหนักถ่วงได้โดยตรง ซึ่งจากหลักการของอัลกอริทึม ISIC ที่ใช้ในอิควอไลเซอร์แบบป้อนกลับหัวข้อที่แล้วพบว่าโครงสร้างแบบป้อนกลับมีคุณสมบัติที่สำคัญอย่างหนึ่งคือ ถ้าสัญญาณเข้าในแต่ละตัวของวงจรกรองป้อนกลับเป็นอิสระจากข้อมูลที่ต้องการ การลด SMSE ก็จะเหมือนการลดค่าเฉลี่ยกำลังของสัญญาณออก (MOE: mean output energy) และในที่นี้สัญญาณเข้าแต่ละตัวของวงจรกรองป้อนกลับคือบิตข้อมูลที่ตรวจจับได้ของผู้ใช้ที่มีลำดับต่ำกว่าผู้ใช้ที่พิจารณาซึ่งเป็นอิสระจากบิตข้อมูลของผู้ใช้ที่พิจารณา ฉะนั้นในกรณีนี้จะ

สามารถปรับค่าน้ำหนักถ่วงในส่วนป้อนกลับโดยการลด MOE ได้เช่นเดียวกับอัลกอริทึม ISIC ซึ่งจะได้พิสูจน์ต่อไป

การลด MOE นี้ไม่จำเป็นต้องอาศัยค่าเทรนนิ่งซีเควนซ์ จะใช้เฉพาะค่าสัญญาณก่อนเข้าตัดสินบิตเท่านั้น และสามารถหาสมการที่ใช้ในการลด MOE ได้จากสมการ stochastic gradient descent เช่นเดียวกับสมการ ISIC ดังนี้

เริ่มจากกระจายสมการ (3-28) จะได้

$$SMSE_k = (\beta_k)^2 E(d_k^n)^2 - 2\beta_k E(d_k^n s_k^n) + E(s_k^n)^2 \tag{3-29}$$

$$SMSE_k = (\beta_k)^2 E(d_k^n)^2 - 2\beta_k E[d_k^n (V_k^n \cdot Y_n - FB_k^n \cdot \hat{D}_{k-1}^n)] + E(s_k^n)^2 \tag{3-30}$$

โดยที่ $\hat{D}_{k-1}^n = [\hat{d}_1^n \cdots \hat{d}_{k-1}^n \; 0 \cdots 0]^T$,

เมตริกซ์ $\mathbf{FB}_n$ เป็นน้ำหนักถ่วงของวงจรกรองป้อนกลับที่เวลา n ซึ่งแต่ละแถวของเมตริกซ์ $\mathbf{FB}_n$ จะแทนวงจรกรองป้อนกลับของผู้ใช้แต่ละคน หรือกล่าวได้ว่าวงจรกรองป้อนกลับ K ชุดประกอบกันเป็นเมตริกซ์ $\mathbf{FB}_n$

$$\mathbf{FB}_n = \begin{bmatrix} 0 & \cdots & \cdots & \cdots & 0 \\ fb_{2,1}^n & 0 & \cdots & \cdots & 0 \\ fb_{3,1}^n & fb_{3,2}^n & 0 & \cdots & 0 \\ \vdots & & & \ddots & 0 \\ fb_{K,1}^n & \cdots & \cdots & fb_{K,K-1}^n & 0 \end{bmatrix} = \begin{bmatrix} FB_1^n \\ FB_2^n \\ FB_3^n \\ \vdots \\ FB_K^n \end{bmatrix} \tag{3-31}$$

และ

$$s_k^n = V_k^n \cdot Y_n - FB_k^n \cdot \hat{D}_{k-1}^n \tag{3-32}$$

จากคุณสมบัติที่ข้อมูลของผู้ใช้แต่ละคนเป็นอิสระต่อกัน และส่งเป็นบิต +1 หรือ -1 โดยมีค่าเฉลี่ยเป็น 0 หรือเขียนได้ว่า $E(d_j^n d_k^n) = 0$, $E(d_j^n \hat{d}_k^n) = 0$ เมื่อ $j \neq k$ และ $E(d_k^n d_k^n) = 1$ จะทำให้ได้

$$\frac{\partial SMSE}{\partial FB_k^n} = \frac{\partial}{\partial FB_k^n}\{(\beta_k)^2 E[(d_k^n)^2] - 2\beta_k E[d_k^n (V_k^n \cdot Y_n - FB_k^n \cdot \hat{D}_{k-1}^n)] + E[(s_k^n)^2]\}$$

$$\frac{\partial SMSE}{\partial FB_k^n} = 0 + 2\beta_k E(d_k^n \hat{D}_{k-1}) + \frac{\partial E[(s_k^n)^2]}{\partial FB_k^n} \tag{3-33}$$

เนื่องจากสัญญาณเข้าส่วนป้อนกลับคือ $\hat{D}_{k-1}^n$ เป็นอิสระจากข้อมูลที่ต้องการ d_k^n จึงทำให้พจน์กลางในสมการ (3-33) มีค่าเป็น 0 เหลือแต่พจน์สุดท้ายซึ่งเป็นส่วนของ MOE ดังนี้

$$\frac{\partial SMSE}{\partial FB_k^n} = \frac{\partial E[(s_k^n)^2]}{\partial FB_k^n} = \frac{\partial MOE}{\partial FB_k^n} \tag{3-34}$$

ซึ่งสมการ (3-34) หมายถึงการลด SMSE เหมือนกับการลดค่าเฉลี่ยกำลังของสัญญาณออกจากมัลติยูสเซอร์ดีเทกชันแบบป้อนกลับ (MOE) และจะหากระบวนการปรับลด MOE ได้จาก

$$\frac{\partial MOE}{\partial FB_k^n} = -2s_k^n \hat{D}_{k-1}^{n\ T} \tag{3-35}$$

ดังนั้นเมื่อใช้สมการ stochastic gradient adaptation rule จะสามารถหากระบวนการปรับน้ำหนักถ่วงแบบบอดของวงจรกรองป้อนกลับของผู้ใช้ลำดับที่ k ได้เป็น [27]

$$FB_k^{n+1} = FB_k^n + \mu(s_k^n \cdot \hat{D}_{k-1}^{n\ T}) \tag{3-36}$$

ซึ่งจะพบว่าเป็นสมการที่มีลักษณะคล้าย ISIC algorithm เพียงแต่มี มิติต่างกัน, สัญญาณเข้าที่ใช้ขึ้นกับลำดับที่ของผู้ใช้ และสัญญาณเข้าเป็นข้อมูลที่ได้จากสัญญาณ ณ เวลาเดียวกันกับสัญญาณที่จะถูกหักล้างด้วยเพียงแต่สัญญาณนั้นถูกหน่วงเวลารอไว้

สัญญาณก่อนจะเข้าไปตัดสินใจที่เวลา n สามารถเขียนได้ดังนี้

$$s_k^n = V_k^n \cdot Y_n - FB_k^n \cdot \hat{D}_{k-1}^n \tag{3-37}$$

และจากภาคผนวก ค สามารถเขียนค่าน้ำหนักถ่วงของวงจรกรองป้อนกลับในกรณีอุดมคติได้เป็น

$$FB_{before_k} = (\Gamma_{before_k} - \Gamma_{after_k}\Gamma_{after_k}^{-1}\Gamma_{after_k \times before_k})\mathbf{A}_{before_k}^n \tag{3-38}$$

โดยที่ FB_{before_k} เป็นแถวที่ k ของเมตริกซ์ **FB** ซึ่งมีสมาชิกทุกตัวก่อนหน้าตัวที่ k หรือกล่าวได้ว่าเป็นค่าของน้ำหนักถ่วงของวงจรกรองป้อนกลับที่ไม่เป็น 0 ของผู้ใช้ลำดับที่ k นั่นเอง

สรุปได้ว่า ในส่วนของวงจรกรองป้อนไปข้างหน้า จะถูกปรับน้ำหนักถ่วงเพื่อลดค่าสหสัมพันธ์ระหว่างสัญญาณออกของผู้ใช้ที่ต้องการกับข้อมูลของผู้ใช้ที่มีลำดับสูงกว่าโดยใช้อัลกอริทึม boostrap ตามสมการ (3-22) ส่วนที่วงจรกรองป้อนกลับ จะปรับน้ำหนักถ่วงเพื่อลด MOE โดยหลักการเดียวกับอัลกอริทึม ISIC ตามสมการ (3-36) บล็อกไดอะแกรมของกระบวนการที่เสนอขึ้นทั้งหมดแสดงอยู่ในรูปที่ 3.7 ถึง 3.9

3.3.2 สมรรถนะที่สภาวะอยู่ตัวในกรณีอุดมคติ

เมื่อพิจารณาที่สภาวะอยู่ตัวกรณีอุดมคติ โดยตั้งสมมุติฐานว่าการประมาณ $D_{after_k}^n$ ด้วย $\text{sgn}(S_{after_k}^n)$ ถูกต้อง จะทำให้วงจรกรองป้อนไปข้างหน้าสามารถกำจัด MAI ของผู้ใช้ที่มีลำดับสูงกว่าได้หมดที่สภาวะอยู่ตัว รวมทั้งตั้งสมมุติฐานว่า บิตข้อมูลที่ตรวจจับไปแล้วของผู้ใช้คนอื่นๆ

ถูกต้องทุกคนทำให้วงจรกรองป้อนกลับสามารถกำจัด MAI ที่เหลือได้หมด ภายใต้สมมุติฐานเหล่านี้จะพบว่าสัญญาณออกจากมัลติยูสเซอร์ดีเทกชันแบบป้อนกลับชนิดบอดของผู้ใช้ลำดับที่ k จะเขียนได้เป็น

$$s_k^n = \alpha_k a_k^n d_k^n + \xi_k^n \tag{3-39}$$

โดยที่ $E(\xi_k^2) = \alpha_k \sigma^2$

ดังนั้นผู้ใช้ลำดับที่ k มีค่าความน่าจะเป็นที่จะเกิดความผิดพลาดในการตรวจจับบิตข้อมูลเป็น

$$Pe_k\,(blind\ DDFMD) = Q\left(\sqrt{\frac{(\alpha_k a_k^n)^2}{\alpha_k \sigma^2}}\right) = Q\left(\sqrt{\frac{\alpha_k (a_k^n)^2}{\sigma^2}}\right) \tag{3-40}$$

และพบว่า $f_{k,k}$ ซึ่งหาได้จากการทำ cholesky decomposition จะเป็นค่าเดียวกันกับรากที่สองของ α_k ในสมการ (3-26) ซึ่งตัวอย่างการเปรียบเทียบค่า $(f_{k,k})^2$ และ α_k ได้แสดงอยู่ในตารางที่ ง.1 ในภาคผนวก ง ดังนั้นจากสมการ (3-40) และสมการ (2-22) พบว่าสมรรถนะที่สภาวะอยู่ตัวในกรณีอุดมคติของมัลติยูสเซอร์ดีเทกชันแบบป้อนกลับชนิดบอดและชนิดค่าน้ำหนักถ่วงคงที่จะมีค่าเท่ากัน แต่อย่างไรก็ตามในกรณีนี้เป็นกรณีอุดมคติ ดังนั้นในกรณีทั่วไปค่า BER จะสูงกว่าสมการนี้ และในกรณีของมัลติยูสเซอร์ดีเทกชันแบบป้อนกลับชนิดค่าน้ำหนักถ่วงคงที่นั้นการที่ข้อมูลที่ตรวจจับได้ของผู้ใช้คนก่อนหน้าไม่ถูกต้องจะส่งผลให้หักล้าง MAI ผิดไป แต่ในกรณีของมัลติยูสเซอร์ดีเทกชันแบบป้อนกลับชนิดบอดนั้นนอกจากจะส่งผลให้หักล้าง MAI ผิดไปแล้วยังส่งผลให้กระบวนการปรับอัตโนมัติมีความผิดพลาดด้วยเนื่องจากในกระบวนการปรับอัตโนมัติมีการใช้ค่าบิตข้อมูลที่ตัดสินไปแล้วของผู้ใช้คนอื่นๆในกระบวนการปรับค่าน้ำหนักถ่วงด้วย

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย