

บทที่ 3

โปรแกรมคอมพิวเตอร์ในการควบคุมกำลังด้านย้อนกลับของระบบ DS/CDMA

การสร้างตัวควบคุมที่ใช้หลักการฟัซซีลอจิกและเจเนติกอัลกอริทึม จะอาศัยโปรแกรม MATLAB ซึ่งเป็นภาษาคอมพิวเตอร์ที่เหมาะสมกับการคำนวณ โดยที่ในโปรแกรม MATLAB มีฟังก์ชันของฟัซซีลอจิก มากพอสมควร ดังนั้นจึงสะดวกที่จะนำมาใช้ในงานวิจัย ในบทนี้ได้อธิบายการใช้คำสั่งของฟัซซีลอจิกเอาไว้ เพื่อสะดวกต่อการศึกษา

3.1 การออกแบบตัวควบคุม Fuzzy GA Controller (FGa)

ระบบควบคุมที่ใช้ฟัซซีลอจิกมีความเร็วในการตัดสินใจขึ้นอยู่กับระดับของกฎที่มีอยู่ ซึ่งสามารถนำมาใช้สร้างระบบควบคุมให้ทำงานกับระบบที่ไม่เป็นเชิงเส้น อย่างเช่นกรณีปัญหาการควบคุมกำลังด้านย้อนกลับของระบบ DS/CDMA ได้

เนื่องจากระบบควบคุมแบบ PID ดั้งเดิมสามารถทำงานควบคุมได้ดี แต่ว่าเมื่อเกิดสัญญาณแทรกสอดขึ้น ระบบควบคุมแบบ PID จะให้สมรรถนะไม่เป็นที่น่าพอใจโดยต้องเลือกระหว่างความต้องการให้ตอบสนองเร็วแต่มีการแกว่งของสัญญาณ (overshoot) มาก หรือตอบสนองช้า แต่มีลักษณะการควบคุมที่เรียบกว่า การควบคุมที่ใช้ฟัซซีลอจิกจะสามารถกำจัดสัญญาณแทรกสอดได้ ซึ่งเป็นการควบคุมที่ง่ายแต่ให้ความมั่นคงกับระบบ และยังครอบคลุมพารามิเตอร์ช่วงกว้างกว่า PID อีกด้วย นอกจากนี้การใช้การควบคุมโดยใช้ฟัซซีลอจิก จะเป็นการเชื่อมโยงระหว่างผู้ใช้กับผู้ออกแบบโปรแกรม เพราะการควบคุมใช้คำสั่งทางภาษาทำให้สามารถเข้าใจได้ง่าย ทำให้ลดระยะเวลาในการทำความเข้าใจกับระบบควบคุมและลดเวลาในการพัฒนาระบบควบคุม [7]

ในรูปที่ 3.1 เป็นการแสดงการทำงานของระบบการควบคุมกำลัง ซึ่งกำลังที่สถานีฐานจะรับหาได้จากสมการ (3.1)

$$P_r = P_T + P_F + P_I - L_p \quad (3.1)$$

- เมื่อ P_r คือ กำลังที่สถานีฐานได้รับจากโทรศัพท์เคลื่อนที่ ที่กำลังพิจารณา [W]
 P_T คือ กำลังส่งจากโทรศัพท์เคลื่อนที่ที่พิจารณา [W]
 P_F คือ กำลังของโทรศัพท์เคลื่อนที่ ที่เปลี่ยนแปลงไปเนื่องจากช่องสัญญาณเกิดเฟดดิ้ง [W]
 P_I คือ กำลังที่เกิดจากการแทรกสอดจากผู้ใช้คนอื่นและเกาซ์เซียนนอยส์ [W]
 L_p คือ กำลังของโทรศัพท์เคลื่อนที่ ที่ลดลงเนื่องจากระยะทาง [W]

เพื่อความสะดวกต่องานวิจัย จึงได้กำหนดให้เงื่อนไขของแบบจำลองระบบ DS/CDMA ให้กำลังส่งที่ออกจากโทรศัพท์เคลื่อนที่เริ่มต้น ทุกเครื่องเป็น 1 W เท่ากันทุกเครื่อง เมื่อเดินทางผ่านช่องสัญญาณที่มีการเฟดดิ้ง ทำให้สัญญาณที่สถานีฐานได้รับเปลี่ยนไป และถูกแทรกสอดด้วยสัญญาณจากโทรศัพท์เคลื่อนที่อื่น ๆ ที่อยู่ในเซลล์ข้างเคียง ซึ่งมีทั้งหมด 18 เซลล์ และจากโทรศัพท์เคลื่อนที่ ที่อยู่ในเซลล์เดียวกันด้วย ดังนั้นสัญญาณที่สถานีฐานได้รับ จึงมีค่าตามสมการที่ (3.1)

ในงานวิจัยนี้ จึงได้เสนอแนวทางการควบคุมกำลังด้านย้อนกลับของระบบบรีงคิง DS/CDMA เพื่อควบคุมให้ กำลังที่สถานีฐานได้รับจากโทรศัพท์เคลื่อนที่ มีค่าคงที่เท่ากับ 1 W โดยการใช้ตัวควบคุมแบบฟัซซีลอจิก โดยใช้เจเนติกอัลกอริทึมให้ทำการค้นหาพารามิเตอร์ของฟัซซีลอจิก เพื่อให้ได้ตัวควบคุมที่เหมาะสมที่สุด

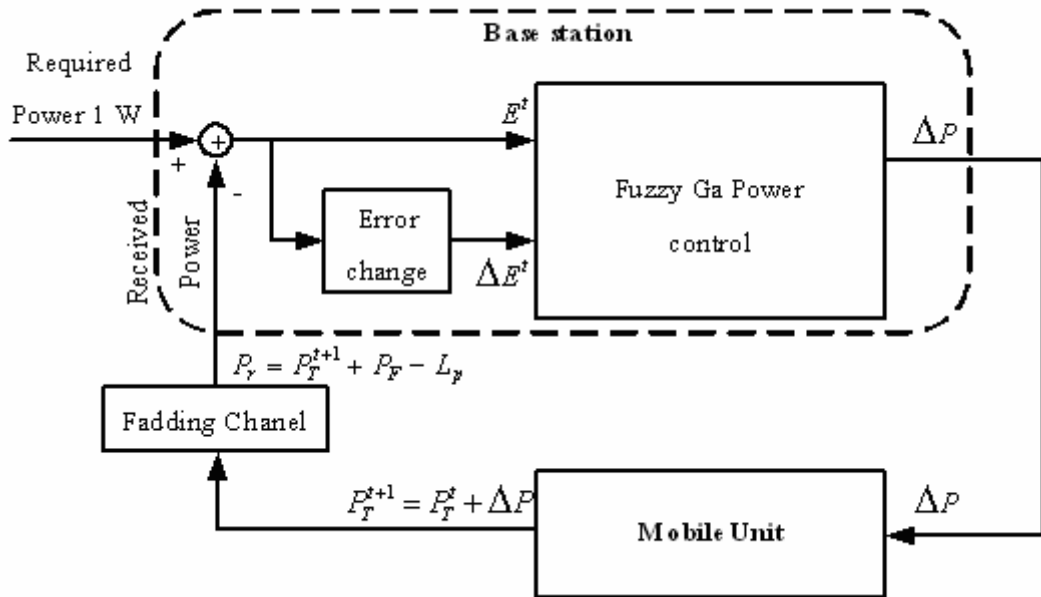
ตัวควบคุม FGa (fuzzy genetic controller) มีอินพุตสองตัวคือ ค่าความผิดพลาด และการเปลี่ยนแปลงของค่าความผิดพลาด และจะให้ค่าเอาต์พุตเป็นกำลังส่วนที่เพิ่มหรือลด (ΔP) ซึ่งจะอธิบายเป็นสมการทางคณิตศาสตร์ได้ดังสมการ (3.2) และ (3.3)

$$P_T^{t+1} = P_T^t + \Delta P \quad (3.2)$$

$$\Delta P = FGa(E^t, \Delta E^t) \quad (3.3)$$

เมื่อ P_r^t คือ กำลังส่งของโทรศัพท์เคลื่อนที่ [W]

ΔP คือ กำลังที่เพิ่ม หรือลด ที่ได้จากตัวควบคุม FGa [W]



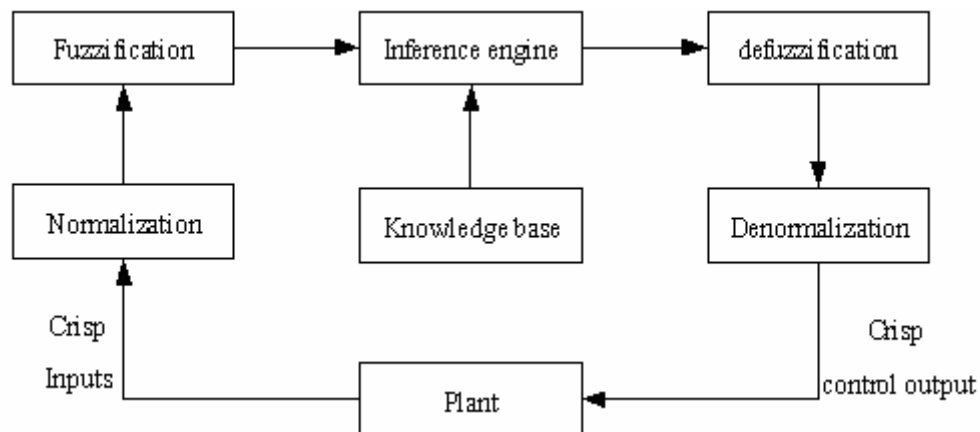
รูปที่ 3.1 โครงสร้างระบบควบคุมกำลังด้านย้อนกลับของระบบรับส่ง DS/CDMA ที่ใช้การควบคุมแบบ Fuzzy Genetic Power Control

โดยที่ E^t เป็นค่าผิดพลาดที่ได้จากกำลังของสัญญาณที่ได้รับที่สถานะ t ลบด้วยกำลังของสัญญาณที่ต้องการ ซึ่งในที่นี้ กำลังที่ต้องการได้รับกำหนดให้เป็น 0 dB และ ΔE^t ได้จากค่าผิดพลาดของกำลังที่ได้รับในขณะปัจจุบันลบด้วย ค่าผิดพลาดของกำลังที่ได้รับก่อนหน้านี้ ดังสมการที่ (3.4)

$$\Delta E^t = E^t - E^{t-1} \quad (3.4)$$

3.2 โปรแกรมคอมพิวเตอร์ฟัซซีลอจิก

หลักของการควบคุมแบบฟัซซีลอจิก ประกอบด้วย 5 ส่วนประกอบหลัก ดังรูปที่ 3.2



รูปที่ 3.2 ระบบควบคุมแบบฟัซซีลอจิก

1) ส่วนแปลงค่าคริปส์อินพุตให้อยู่ในเอกภพสัมพัทธ์ (normalization)

ทำหน้าที่ในการวัดค่าของตัวแปรอินพุต และทำการแปลงค่าของตัวแปรอินพุตให้มาอยู่ในเอกภพสัมพัทธ์ที่ต้องการ เช่น ในกรณีที่ ค่าอินพุตของฟัซซี ที่กำหนดให้มีค่าอยู่ระหว่าง $[-18, 18]$ เมื่อรับค่าอินพุตจากระบบ จะต้องทำการแปลงให้อยู่ระหว่างค่าที่กำหนดให้ เช่น ในกรณีที่รับค่าอินพุตน้อยกว่า -18 ก็อาจกำหนดให้เป็น -18 เป็นต้น

2) ส่วนแปลงเป็นฟัซซี (fuzzification)

ทำหน้าที่แปลงค่าคริปส์อินพุตที่อยู่ในเอกภพสัมพัทธ์ที่ต้องการแล้วให้เป็นค่าตัวแปรเชิงภาษา (linguistic values) เช่น ถ้าค่าอินพุตที่รับมาหลังจากทำการแปลงให้อยู่ในขอบเขตของตัวแปรอินพุตของฟัซซีแล้ว มีค่าเป็น -18 ซึ่งจะกำหนดให้เป็นค่าตัวแปรเชิงภาษาเป็น NL (negative large) เป็นต้น

3) ส่วนดำเนินการอนุมานกฎ (inference engine)

ส่วนดำเนินการอนุมานกฎ จะมีฐานความรู้ (knowledge base) ซึ่งในฐานความรู้จะประกอบด้วย ฐานข้อมูล คือ ความรู้เกี่ยวกับระบบ และส่วนกฎควบคุมเชิงภาษา ทำหน้าที่เก็บกฎต่างๆ ที่ผู้เชี่ยวชาญในระบบนั้นทำการป้อนข้อมูลเอาไว้ เมื่อจะทำการอนุมานก็จะต้องใช้ความรู้จากฐานความรู้เพื่อทำการตัดสินใจ เช่น

If input1 is NL and input2 is NL then output is PL

4) ส่วนแปลงค่าฟัซซีกลับ (defuzzification)

ทำหน้าที่แปลงค่าเอาต์พุตเชิงภาษา ให้กลับเป็นค่าคริสป์เอาต์พุตที่สามารถนำไปใช้งานจริงได้ เช่น ถ้าค่าเอาต์พุตที่ได้เป็น PL (positive large) จะให้ค่าเป็น 18 เป็นต้น

5) ส่วนแปลงค่าในเอกภพสัมพัทธ์กลับเป็นค่าคริสป์เอาต์พุต (denormalization)

ทำหน้าที่แปลงค่าฟัซซีเอาต์พุตให้เป็นค่าคริสป์เอาต์พุตส่งกลับไปยังระบบที่ถูกควบคุมอยู่ให้ทำงานตามกระบวนการควบคุมดังกล่าว ในงานวิจัยนี้จะใช้ค่าที่ได้จาก กระบวนการแปลงค่าฟัซซีกลับ มาเป็นค่าเอาต์พุตที่ใช้ควบคุมโดยตรง เนื่องจากขอบเขตของค่าเอาต์พุตที่กำหนดในตัวควบคุมฟัซซีลอจิก มีขอบเขตที่เหมาะสมต่อการใช้งานอยู่แล้ว

กระบวนการทั้งหมดที่กล่าวมา จะเริ่มต้นตั้งแต่ ขั้นตอนที่ 1) จนถึง 5) หลังจากนั้น กระบวนการทั้งหมดก็จะเริ่มต้นวัฏจักรใหม่

เนื่องจากการใช้ฟัซซีลอจิกในโปรแกรม MATLAB มีอยู่ด้วยกันสองแบบ แบบแรก เป็นการใชฟัซซีลอจิกโดยผ่านส่วนที่เรียกว่า GUI (Graphic User Interface) แบบที่สอง เป็นการใช้ฟังก์ชันที่มีอยู่ใน MATLAB ในที่นี้จะใช้แบบที่สอง

3.2.1 การออกแบบตัวควบคุมฟัซซีลอจิก

1) การออกแบบส่วนบรรทัดฐาน (Normalization)

สำหรับงานวิจัยนี้ใช้ตัวควบคุมฟัซซีลอจิกแบบ MISO (Multi input single output) ค่าผลต่างระหว่างกำลังที่สถานีฐานต้องการกับกำลังที่สถานีฐานได้รับ (E) และค่าเปลี่ยนแปลงของผลต่างดังกล่าว (ΔE) เป็นตัวแปรอินพุตสำหรับฟัซซีลอจิก โดยจะกำหนดให้ขอบเขตของ E อยู่ระหว่าง $[-18,18]$ และ ΔE อยู่ระหว่าง $[-12,12]$ และ ให้ค่าเอาต์พุตเป็นการลดลงหรือเพิ่มขึ้นของ

กำลัง (ΔP) ซึ่งมีขอบเขตอยู่ระหว่าง $[-6,6]$ ดังนั้นค่า E ที่เข้ามามีค่าเกินกว่า -18 ก็จะถูกกำหนดค่าเป็น -18 ทำนองเดียวกันกับ ΔE

2) การออกแบบส่วนแปลงเป็นค่าฟัซซี (Fuzzification)

เมื่อค่าคริปส์อินพุตทั้งสอง ($E, \Delta E$) ถูกทำให้อยู่ในเอกภพสัมพัทธ์แล้วจะถูกแปลงเป็นค่าฟัซซีอินพุต ซึ่งในงานวิจัยนี้จะใช้กลไกการอนุมานแบบแมมดानी (mamdani) และใช้ตัวดำเนินการค่าต่ำสุด (min Implication) ดังสมการที่ 3.5

$$\mu_C(W) = \min\{\mu_A(U), \mu_B(V)\} \quad (3.5)$$

เมื่อ A เป็นค่าเชิงภาษาที่อยู่ในเอกภพสัมพัทธ์ U โดยที่ $\mu_A(U)$ เป็นค่าของฟังก์ชันการเป็นสมาชิกของตัวแปรภาษา A ซึ่งมีขอบเขตอยู่ในเอกภพสัมพัทธ์ U ซึ่งเป็นของตัวแปรฟัซซีอินพุต

B เป็นค่าเชิงภาษาที่อยู่ในเอกภพสัมพัทธ์ V โดยที่ $\mu_B(V)$ เป็นค่าของฟังก์ชันการเป็นสมาชิกของตัวแปรภาษา B ซึ่งมีขอบเขตอยู่ในเอกภพสัมพัทธ์ V ซึ่งเป็นของตัวแปรฟัซซีอินพุต

C เป็นค่าเชิงภาษาที่อยู่ในเอกภพสัมพัทธ์ W โดยที่ $\mu_C(W)$ เป็นค่าของฟังก์ชันการเป็นสมาชิกของตัวแปรภาษา C ซึ่งมีขอบเขตอยู่ในเอกภพสัมพัทธ์ W ซึ่งเป็นของตัวแปรฟัซซีเอาต์พุต

ตามปกติการสรุปกฎอาจมีได้หลายผลลัพธ์เนื่องจากค่าอินพุตอาจถูกตีความหมายได้มากกว่า 1 ค่า เช่นเมื่อ $E = 0.5$ จะตีความหมายเป็นค่าเชิงภาษาได้ 2 ค่า คือ ZE และ PS และเช่นเดียวกัน $\Delta E = 5$ จะตีความหมายเป็นค่าเชิงภาษาได้ 2 ค่า เช่นกันคือ PS และ PM ดังนั้นเมื่อจะดำเนินการสรุปกฎ จะมีค่าเชิงภาษาที่ถูกเรียกใช้งานจริง 4 กฎคือ

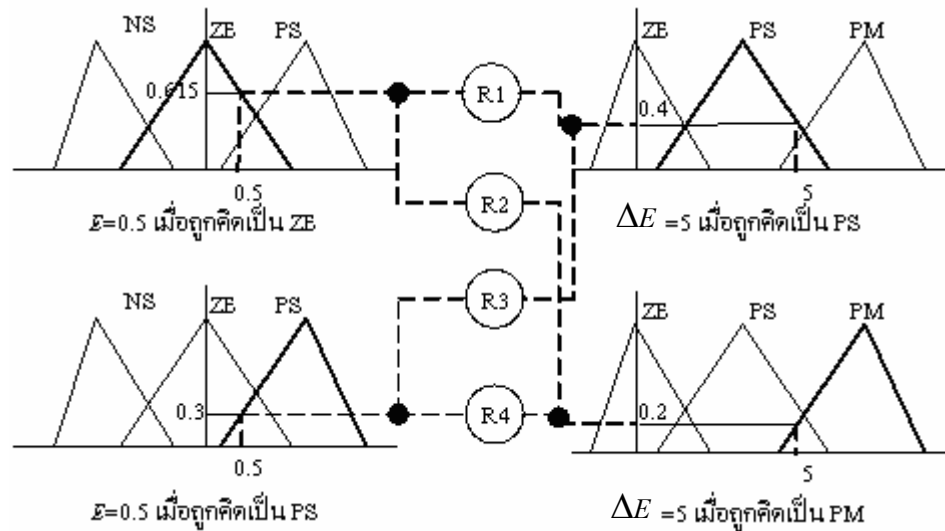
R1 : if (E is ZE) and (ΔE is PS) then (ΔP is NM)

R2 : if (E is ZE) and (ΔE is PM) then (ΔP is NS)

R3 : if (E is PS) and (ΔE is PS) then (ΔP is NS)

R4 : if (E is PS) and (ΔE is PM) then (ΔP is ZE)

การสรุปกฎขั้นตอนแรกค่าเชิงภาษาทั้งสี่ จะถูกดำเนินการด้วยตัวดำเนินการค่าต่ำสุด (Min Implication) รูปที่ 3.3 เป็นกระบวนการแปลงค่าคริปส์อินพุตเป็นค่าทางภาษาของฟัซซีอินพุต



รูปที่ 3.3 กฎสี่กฎที่ถูกใช้งานเมื่อ $E = 0.5$ และ $\Delta E = 5$

ยกตัวอย่างการหาค่าความเป็นสมาชิก (MF) จากแกน y จะใช้ความสัมพันธ์ของสมการเส้นตรง ดังสมการที่ (3.6)

$$y = ax + b \quad (3.6)$$

ยกตัวเช่น ถ้ามีข้อมูลบนเส้นตรงของค่าเชิงภาษา ZE ในด้านขวามือ ของตัวแปรอินพุต E อยู่ในช่วง $[0, 1.3]$ จะมีจุดที่ทราบพิกัด (x, y) สองจุด คือ $(0, 1)$ กับจุด $(1.3, 0)$ พิกัดทั้งสองเมื่อนำไปแทนในสมการที่ (3.6) เราจะสามารถหาค่า a และ b ได้ทำให้สามารถเขียนสมการเส้นตรงได้ดังสมการที่ (3.7)

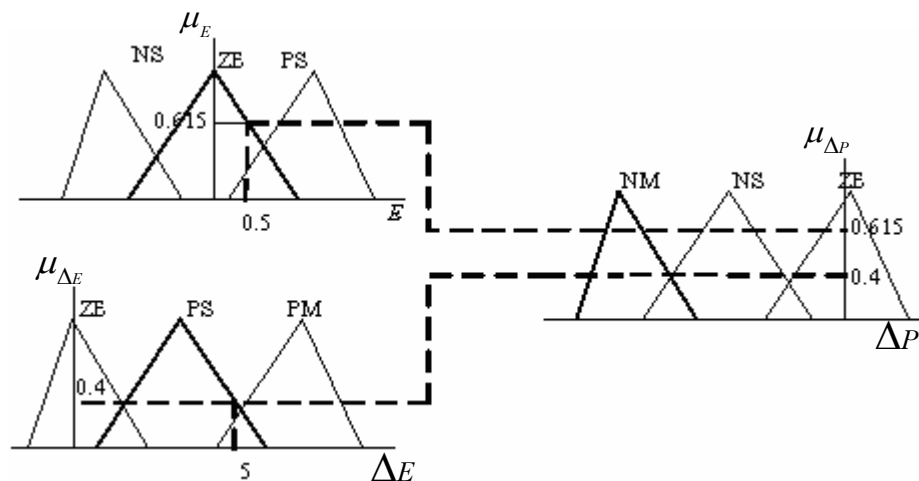
$$y = -0.77x + 1 \quad (3.7)$$

เมื่อค่าคริปส์อินพุตมีค่าเป็น 0.5 เราจะหาค่าความเป็นสมาชิกที่อยู่บนแกน y ได้เป็น 0.615 ในการหาค่าความเป็นสมาชิกของค่าคริปส์อินพุตอื่น ก็จะดำเนินการเช่นเดียวกัน

เมื่อได้ค่าความเป็นสมาชิกแล้ว ก็จะดำเนินการหาค่าความเป็นสมาชิกของฟัซซีเอาต์พุต ΔP โดยการประยุกต์ใช้สมการดังต่อไปนี้

$$\begin{aligned}\mu_{\Delta P} &= \min\{E, \Delta E\} \\ &= \min\{0.615, 0.4\} \\ &= 0.4\end{aligned}$$

การอนุมานค่าฟัซซีอินพุต โดยใช้ตัวดำเนินการค่าสุด กฎที่ใช้เป็นตัวอย่างคือ กฎข้อที่ 1 ทำให้ได้ค่าความเป็นสมาชิก ($\mu_{\Delta P}$) มีค่าเป็น 0.4 แสดงดังรูปที่ 3.4



รูปที่ 3.4 การหาค่าความเป็นสมาชิกของตัวแปรฟัซซีเอาต์พุต ด้วยตัวดำเนินการค่าสุด

จากที่กล่าวมาเป็นเพียงการสรุปกฎเพียงกฎเดียว ซึ่งมีกฎที่ถูกเรียกใช้งานถึง 4 กฎ และมีกฎที่สรุปแล้วให้ตัวแปรทางภาษาเดียวกันคือ NS แต่มีค่าความเป็นสมาชิกต่างกันคือ กฎข้อที่ 2 และ 3 ซึ่งให้ค่าความเป็นสมาชิกของตัวแปรฟัซซีเอาต์พุต คือ 0.2 และ 0.3 ตามลำดับ ซึ่งจะต้องมีการเลือกว่าจะใช้ค่าความเป็นสมาชิกเพียงค่าใดค่าหนึ่ง ซึ่งกระบวนการดังกล่าวเรียกว่าเป็น การแจงเหตุสุด ผล โดยในงานวิจัยนี้เลือกวิธีแจงเหตุสุดแบบตัวดำเนินการสูงสุดแบบทั่วไป (general max aggregation) ซึ่งมีการหาค่าเป็นดังสมการที่ (3.8)

$$\mu_{\Delta P} = \max[\mu_{\Delta P1}, \dots, \mu_{\Delta Pn}] \quad (3.8)$$

ดังนั้นค่าความเป็นสมาชิกของตัวแปรฟัซซีเอาต์พุต ΔP ที่มีตัวแปรทางภาษาเป็น NS จะเป็นหาได้จาก

$$\mu_{\Delta P}(\text{NS}) = \max[\min\{0.615, 0.2\}, \min\{0.3, 0.4\}] = 0.3$$

3) การออกแบบส่วนแปลงค่าฟัซซิกลับ (Defuzzification)

หลังจากผ่านกระบวนการแปลงค่าฟัซซีแล้ว ค่าเชิงภาษาของตัวแปร ΔP แต่ละตัวจะมีค่าความเป็นสมาชิกที่ถูกสรุปมากกว่าหนึ่งครั้ง โดยค่าเชิงภาษาที่ถูกสรุปมากกว่าหนึ่งครั้งจะใช้ค่าความเป็นสมาชิกที่สูงที่สุด ส่วนกฎที่ไม่ถูกใช้งานจะมีค่าความเป็นสมาชิกเป็นศูนย์ การแปลงค่าฟัซซีเอาต์พุตกลับให้เป็นค่าคริปส์เอาต์พุตนั้นมีด้วยกันหลายวิธีที่นิยมใช้กัน ได้แก่ วิธีหาจุดศูนย์ถ่วง (COG : center of gravity), วิธีหาค่าเฉลี่ยบริเวณที่สูงที่สุด (mean of maxima) และวิธีพิจารณาความสูง (height method) เป็นต้น สำหรับงานวิจัยนี้เลือกใช้วิธีหาจุดศูนย์ถ่วง ซึ่งเป็นวิธีที่นิยมใช้ วิธีการหาจุดศูนย์ถ่วงแสดงดังสมการที่ (3.9)

$$u^* = \frac{\sum_{i=1}^M u_i \cdot \mu(u_i)}{\sum_{i=1}^M \mu(u_i)} \quad (3.9)$$

เมื่อ u^* เป็นค่าคริปส์เอาต์พุต

u_i เป็นตัวแปรภาษา

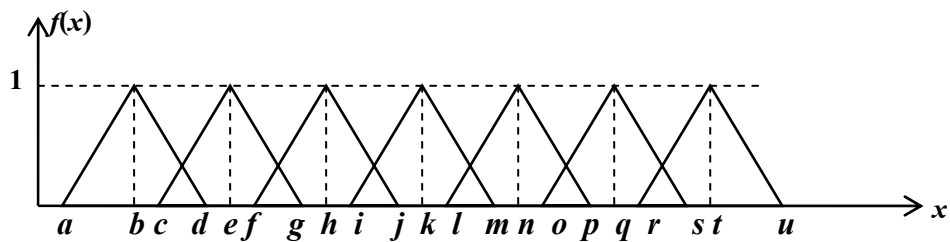
$\mu(u_i)$ เป็นค่าความเป็นสมาชิกของตัวแปรภาษา u_i

$\sum_{i=1}^M u_i \cdot \mu(u_i)$ เป็นการหาพื้นที่ใต้กราฟของตัวแปรภาษา โดยที่ M คือ

จำนวนตัวแปรภาษาของค่าความเป็นสมาชิก ในงานวิจัยนี้มีตัวแปรภาษาทั้งหมด 7 ตัวแปร คือ $M=7$

3.2.2 การหาตำแหน่งของฟังก์ชันความเป็นสมาชิก

ในงานวิจัยนี้ ได้ทำการสร้างฟังก์ชันเพื่อทำการสุ่มตัวเลขที่จะนำมาใช้เป็นตำแหน่งของฟังก์ชันการเป็นสมาชิก สำหรับฟังก์ชันการเป็นสมาชิกรูปสามเหลี่ยม จะต้องใช้ตำแหน่งเพื่อสร้างจุด สามจุด ต่อหนึ่งฟังก์ชันการเป็นสมาชิก เนื่องจากมีทั้งหมด 7 ฟังก์ชันการเป็นสมาชิก ดังนั้นจำนวนจุดที่ต้องทำการสุ่มทั้งหมดจะมี 21 จุด



รูปที่ 3.5 ฟังก์ชันการเป็นสมาชิกรูปสามเหลี่ยม

ในรูปที่ 3.5 เป็นฟังก์ชันการเป็นสมาชิกรูปสามเหลี่ยมจำนวน 7 ฟังก์ชัน ในการหาจุด $[a \ b \ c \dots u]$ จะได้จากการสุ่มหาจุดบนแกน x ที่มีความละเอียด (Resolution) บนแกน x หาได้จาก

$$resolution = \frac{L_{right} - L_{left}}{|L| - 1} \quad (3.10)$$

โดยที่ L_{right} เป็นขอบเขตบนของตัวแปรฟัซซีใด ๆ
 L_{left} เป็นขอบเขตล่างของตัวแปรฟัซซีใด ๆ
 $|L|$ เป็นจำนวนจุดบนแกน x ที่เราต้องการ

เนื่องจากเราต้องการให้จำนวนยีนของโครโมโซมในส่วนของ GA แต่ละโครโมโซม มีจำนวนเท่ากัน เราจึงได้กำหนดให้ค่า $|L| = 361$ ซึ่งเราจะหาตำแหน่งบนแกน x ได้จาก

$$A = L_{left} + \frac{L_{right} - L_{left}}{|L| - 1} \times p \quad (3.11)$$

โดยที่ A เป็นจุดบนแกน x

p เป็นลำดับของจำนวนจุดบนแกน x (โดยในที่นี้มีจุดตั้งแต่ลำดับที่ 0 ถึง 360)

ยกตัวอย่างเช่น ลำดับที่ 0, ลำดับที่ 1 และลำดับที่ 360 ของตัวแปรฟuzzy E อยู่ที่จุดบนแกน x คือ

$$A_0 = -18 + \frac{18 - (-18)}{361 - 1} \times 0 = -18$$

$$A_1 = -18 + \frac{18 - (-18)}{361 - 1} \times 1 = -17.9$$

$$A_{360} = -18 + \frac{18 - (-18)}{361 - 1} \times 360 = 18$$

3.2.3 รูปแบบคำสั่งของฟuzzyลอจิกในโปรแกรม MATLAB

การใช้ฟังก์ชันฟuzzyลอจิกที่อยู่ในโปรแกรม MATLAB มีรูปแบบการใช้ดังนี้ ฟังก์ชัน newfis ทำการสร้างเมตริก FIS ที่ชื่อ cdma ฟังก์ชัน addvar ใช้สำหรับสร้างตัวแปรและกำหนดขอบเขตให้กับเมตริก FIS ในที่นี้ได้มีการสร้างตัวแปรสองตัวคือ E , EC และ PC รูปแบบคำสั่งคือ

```
a=newfis('cdma');
a=addvar(a,'input','E',[-18 18]);
a=addvar(a,'input','EC',[-12 12]);
a=addvar(a,'output','PC',[-6 6]);
```

คำสั่งที่ใช้ในการกำหนดชื่อให้กับฟังก์ชันการเป็นสมาชิกตัวที่ 1 ของอินพุตตัวที่ 1 กำหนดให้ชื่อว่า NL มีรูปแบบคำสั่งคือ

```
a.input(1).mf(1).name='NL';
```

คำสั่งที่ใช้ในการกำหนดรูปร่างให้กับฟังก์ชันการเป็นสมาชิกตัวที่ 1 ของอินพุตตัวที่ 1 กำหนดให้เป็นรูปร่างสามเหลี่ยม มีรูปแบบคำสั่งคือ

```
a.input(1).mf(1).type='trimf';
```

คำสั่งที่ใช้กำหนดจุดเพื่อนำไปสร้างเป็นฟังก์ชันการเป็นสมาชิกตัวที่ 1 ของอินพุตที่ 1 โดยในที่นี้จะรับค่าจากตัวแปรชื่อ point ซึ่งสามารถเปลี่ยนแปลงค่าไปได้ ตามการสุ่มของหลักวิธีเจเนติก อัลกอริทึม มีรูปแบบคำสั่งคือ

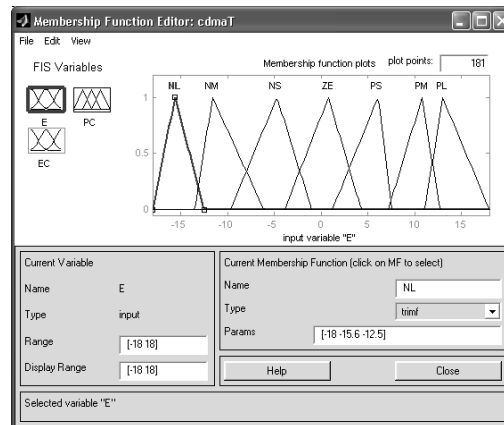
```
a.input(1).mf(1).params=[point(1,g(1)) point(1,g(2))point(1,g(4))];
```

คำสั่งที่ใช้ในการกำหนดกฎให้กับเมตริก FIS โดยเริ่มต้นจะกำหนดให้กฎเป็นตัวแปรแบบ โครงสร้าง (Structure) จากนั้นนำกฎที่ได้จากการสุ่มไปใส่ให้กับเมตริก FIS โดยการใช้คำสั่ง addrule มีรูปแบบคำสั่งคือ

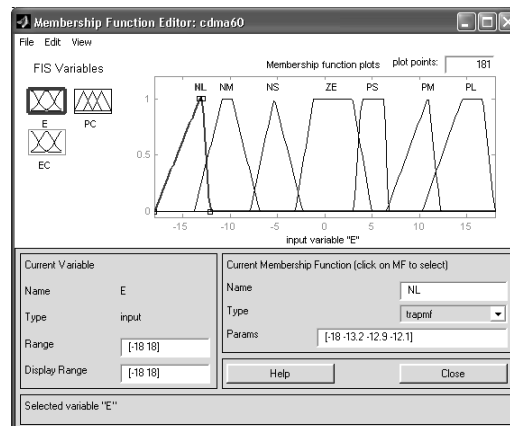
```
a.rule=struct([ ]);  
ruleList=round(rand(7,7)*6+1);  
a= addrule(a,ruleList);
```

ในงานวิจัยนี้ได้ทำการสร้างเมตริก FIS ให้มีรูปแบบของฟังก์ชันการเป็นสมาชิกแตกต่างกัน สามรูปแบบคือ รูปสามเหลี่ยม (triangle) รูปสี่เหลี่ยมคางหมู (trapezoidal) และรูปเกาส์เซียน (gaussian) โดยลักษณะการเขียนโปรแกรมคล้ายกัน แตกต่างกันตรงจำนวนจุดของฟังก์ชันการเป็น สมาชิก โดยรูปสี่เหลี่ยมคางหมูจะมีจำนวนจุด 4 จุดต่อหนึ่งฟังก์ชันการเป็นสมาชิก และรูปเกาส์ เซียนจะมีจำนวนจุด 1 จุดต่อหนึ่งฟังก์ชันการเป็นสมาชิก

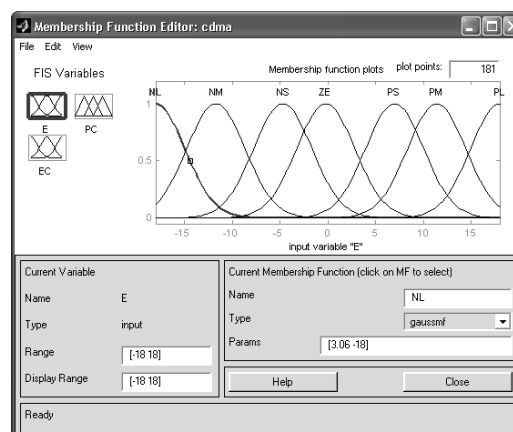
รูปที่ 3.6 ถึง 3.8 เป็นฟังก์ชันการเป็นสมาชิกรูปสามเหลี่ยม สี่เหลี่ยมคางหมู และเกาส์เซียน ที่ สร้างจากการสุ่มตำแหน่งจากหลักวิธีเจเนติกอัลกอริทึม โดยตำแหน่งของฟังก์ชันความเป็นสมาชิก “NL” และ “PL” จะถูกกำหนดให้ตำแหน่งสุดท้ายของฟังก์ชันอยู่ตรงกับค่าขอบต่ำสุด และขอบ สูงสุดของตัวแปรฟัซซีแต่ละตัว



รูปที่ 3.6 ฟังก์ชันการเป็นสมาชิกรูปสามเหลี่ยม ที่ได้จากการสุ่มจุด 3 จุด



รูปที่ 3.7 ฟังก์ชันการเป็นสมาชิกรูปคางหมู ที่ได้จากการสุ่มจุด 4 จุด



รูปที่ 3.8 ฟังก์ชันการเป็นสมาชิกรูปเกาส์เซียน ที่ได้จากการสุ่มจุด 1 จุด

3.3 โปรแกรมคอมพิวเตอร์เลียนแบบวิธีเจเนติกอัลกอริทึม

วิธีเจเนติกอัลกอริทึม หรือ GA เป็นวิธีการค้นหาแบบหนึ่ง ที่มีพื้นฐานเลียนแบบกลไกของการคัดเลือกตามธรรมชาติ โดยที่พันธุ์ดีและเหมาะสมกว่า จะถูกคัดเลือกให้คงอยู่ หลักวิธี GA จึงมักถูกนำไปใช้ในการแก้ปัญหาการหาค่าความเหมาะสม เพื่อหาคำตอบที่ดีที่สุดภายใต้เงื่อนไข และฟังก์ชันจุดประสงค์ที่กำหนด โดยเริ่มต้นจากการสุ่มค่าคำตอบขึ้นมาจำนวนหนึ่ง เรียกว่าประชากร โดยที่แต่ละประชากรเรียกว่าโครโมโซม ในแต่ละโครโมโซมประกอบด้วยยีน หลายๆ ยีนเรียงต่อกัน โดยมากจะแทนแต่ละยีนด้วยเลขฐานสอง เลขจำนวนจริง หรือเลขจำนวนเต็ม จากนั้นจึงพิจารณาคุณภาพประชากรทั้งหมดว่ามีความเหมาะสมเพียงใด โดยการนำไปแทนในฟังก์ชันจุดประสงค์ ซึ่งผลลัพธ์แสดงถึงค่าความเหมาะสมของประชากรนั้น เมื่อทราบค่าความเหมาะสมแล้ว จึงทำการคัดเลือกพ่อพันธุ์ และแม่พันธุ์ โดยอาจจะใช้วิธีหมุนวงล้อ (roulette wheel) ซึ่งเป็นการนำโครโมโซมรุ่นพ่อและแม่ ถ่ายทอดไปสู่รุ่นลูกโดยตรง หรืออาจจะใช้วิธีอื่นๆ ก็ได้

หลังจากที่ได้พ่อพันธุ์และแม่พันธุ์แล้ว จะทำการสุ่มเลือกพ่อและแม่เพื่อมาผลิตรุ่นลูกออกมาเป็นประชากรรุ่นใหม่ ที่เกิดขึ้นมาด้วยวิธีการมิวเตชัน (mutation) และการครอสโอเวอร์ (crossover)

การมิวเตชัน เป็นการเปลี่ยนค่าของยีนในโครโมโซมตามตำแหน่งที่สุ่มขึ้นมา มีเพียงพ่อหรือแม่ 1 ตัว ให้กำเนิดลูก 1 ตัว โดยที่ถ้าค่าของยีนในกรณีที่เป็นเลขฐานสอง ในตำแหน่งที่สุ่มขึ้นมามีค่าเป็น 0 จะถูกเปลี่ยนเป็น 1

การครอสโอเวอร์ เป็นการสลับยีนหรือชุดของยีนระหว่างโครโมโซม ตามตำแหน่งที่สุ่มขึ้นมา จะได้โครโมโซมที่เป็นรุ่นลูก ที่มีส่วนของพ่อและแม่ผสมกัน คือมีพ่อและแม่ ให้กำเนิดลูกออกมา 2 ตัว

เมื่อผ่านกระบวนการเหล่านี้แล้ว จะได้ประชากรรุ่นใหม่ ซึ่งเกิดจากการนำโครโมโซมที่ดีไปแทนที่รุ่นพ่อรุ่นแม่ที่ด้อยกว่า ทำกระบวนการเช่นนี้ไปเรื่อยๆ จนถึงยุคสุดท้ายซึ่งโครโมโซมที่ได้จะเป็นคำตอบที่ดีที่สุดของปัญหานั้นเอง

ในงานวิจัยนี้ ได้นำเอาโครโมโซมที่ใช้ในงานวิจัยของ [21] ใส่ไว้ในขั้นตอนที่สุ่มสร้างโครโมโซมด้วย เนื่องจากต้องการทดสอบว่าโครโมโซมจากงานวิจัยนี้ จะเป็นโครโมโซมที่ดีที่สุดหรือไม่ และเพื่อลดระยะเวลาในการหาโครโมโซมที่ดีที่สุด

การเขียนโปรแกรมคอมพิวเตอร์เลียนแบบวิธีเจเนติกอัลกอริทึมประกอบด้วยส่วนต่างๆ ดังต่อไปนี้

3.3.1 การแทนค่าตัวแปร (Representation)

ตัวแปรอินพุตจะประกอบด้วยค่าผิดพลาด (E) และค่าผิดพลาดที่เปลี่ยนไป (ΔE) ตัวแปรเอาต์พุต เป็นกำลังที่เปลี่ยนไป (ΔP) เราทำการเข้ารหัสโดยใช้ตัวเลขจำนวนเต็ม และ จำนวนจริง ดังต่อไปนี้

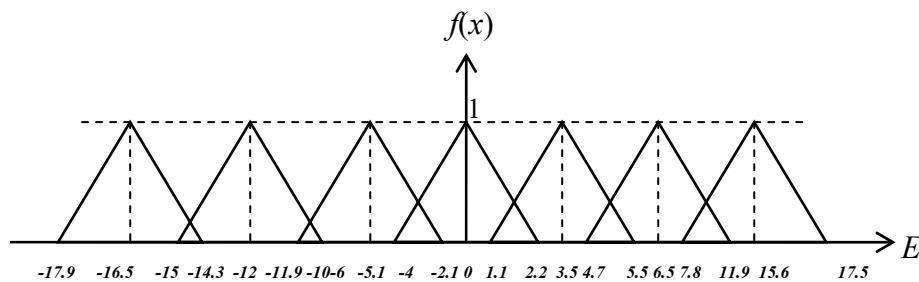
ตัวเลขจำนวนเต็ม 1, 2, 3,...,7 กำหนดให้ใช้แทนกฎ ดังนี้

- 1 แทน NL (Negative Large)
- 2 แทน NM (Negative Medium)
- 3 แทน NS (Negative Small)
- 4 แทน ZE (Zero)
- 5 แทน PS (Positive Small)
- 6 แทน PM (Positive Medium)
- 7 แทน PL (Positive Large)

ตัวเลขจำนวนเต็ม 8, 9 และ 10 แทนรูปร่างของการเป็นสมาชิก ดังนี้

- 8 แทนการเป็นสมาชิกรูปสามเหลี่ยม
- 9 แทนการเป็นสมาชิกรูปสี่เหลี่ยมคางหมู
- 10 แทนการเป็นสมาชิกรูปเกาซ์เซียน

เราใช้ตัวเลขจำนวนจริงเป็นตำแหน่งบนแกน x โดยมีขอบเขตดังนี้ สำหรับ E จะเป็นตัวเลขจำนวนจริงที่อยู่ระหว่าง $(-18,18)$ สำหรับ ΔE จะเป็นตัวเลขจำนวนจริงที่อยู่ระหว่าง $(-12,12)$ และ ΔP จะเป็นตัวเลขจำนวนจริงที่อยู่ระหว่าง $(-6,6)$ [16] โดยเมื่อทำการสุ่มรูปร่างและตำแหน่งของจุดของการเป็นสมาชิกได้แล้ว จะถูกนำไปสร้างฟังก์ชันการเป็นสมาชิกในส่วน of โปรแกรมฟิชชี ซึ่งจะได้เป็นลักษณะดังรูปที่ 3.8



รูปที่ 3.9 การเข้ารหัสของฟังก์ชันการเป็นสมาชิกรูปสามเหลี่ยม สำหรับตัวแปรอินพุต E

ในรูปที่ 3.9 เป็นการแทนจุดสามจุดของการเป็นสมาชิกรูปสามเหลี่ยมโดยใช้ตัวเลขจำนวนจริง จุดที่เป็นจุดยอด และฐานของสามเหลี่ยมจะถูกแทนด้วยตัวเลขจำนวนจริงบนแกน x ส่วนจุดอื่นจะถูกแทนด้วยศูนย์ โดยจำนวนบิตสำหรับตัวแปรอินพุตหนึ่งตัวจะเท่ากับ 361 บิต ซึ่งประกอบด้วยเลขจำนวนจริง 21 ตัว ที่เหลือเป็นศูนย์ ในการสร้างโปรแกรม จะทำการสุ่มหาตัวเลขที่ไม่ซ้ำกันทั้งหมด 21 ตัวระหว่าง 1 ถึง 361 แล้วจึงนำตัวเลขที่ได้มาหาตัวเลขจำนวนจริง ซึ่งเป็นตำแหน่งบนแกน x จากสมการที่ 3.7

ในวิทยานิพนธ์นี้ได้กำหนดใน 1 โครโมโซมประกอบไปด้วย ตัวเลขที่ใช้แทนรูปร่างของการเป็นสมาชิก ซึ่งได้จากการสุ่มตัวเลข ระหว่าง 8-10, ตำแหน่งของการเป็นสมาชิกบนแกน x และกฎ ซึ่งมีทั้งหมด 49 กฎ ดังรูป 3.10

Shape MFs	Point of MFs Input E	Point of MFs Input EC	Point of MFs output PC	Rules
-----------	-------------------------	--------------------------	---------------------------	-------

รูปที่ 3.10 องค์ประกอบของโครโมโซม

รูปร่างของฟังก์ชันการเป็นสมาชิกแต่ละแบบจะถูกเข้ารหัสดังนี้ รูปสามเหลี่ยมถูกเข้ารหัสโดยอาศัยจุดยอดทั้งสามจุด เราใช้จำนวนการเป็นสมาชิกทั้งหมด 7 แบบ [14] ทำให้เราได้จุดทั้งหมดเป็น 21 จุด ส่วนที่เหลือเป็นค่าศูนย์ รูปสี่เหลี่ยมคางหมู ถูกเข้ารหัสโดยอาศัยจุดสี่จุด ทำให้ได้จุดทั้งหมดเป็น 28 จุด นอกนั้นเป็นศูนย์ และรูปเกาส์เซียน ถูกเข้ารหัส โดยใช้ค่าความชัน และจุดที่ได้จากการโปรเจกต์จุดกึ่งกลางของรูปเกาส์เซียนลงบนแกน x ทำให้ได้จำนวนจุดทั้งหมดเป็น 14 จุด ที่เหลือเป็นศูนย์

ในการสร้างกฎฟัซซีโดยการสุ่มตามหลักการเจเนติกอัลกอริทึม ต้องไม่ให้เกิดเหตุการณ์ที่ขัดแย้งกัน หมายถึงต้องไม่มี ส่วนสาเหตุและส่วนผลลัพธ์ที่ซ้ำกัน ยกตัวอย่างเช่น เหตุการณ์ที่เกิดการขัดแย้งกันดังนี้

สมมติให้ทำการสุ่มกฎโดยหลักการของเจเนติกอัลกอริทึมได้เป็น

$$E = 1 \text{ (แปลว่ามี error เป็น NL)}$$

$$\Delta E = 2 \text{ (แปลว่ามี error change เป็น NM)}$$

$$\Delta P = 3 \text{ (แปลว่ามี power change เป็น NS)}$$

เมื่อทำการสุ่มต่อไปพบว่า

$$E = 1$$

$$\Delta E = 2$$

$$\Delta P = 4 \text{ (แปลว่ามี การเปลี่ยนแปลงกำลังเป็น ZE)}$$

จากการสุ่มสองชุด ให้ผลลัพธ์เอาต์พุต ΔP ที่ต่างกัน ทำให้เกิดการขัดแย้งกัน ดังนั้นจึงไม่ควรทำการสุ่มทั้งส่วนของสาเหตุและผลลัพธ์ แต่สุ่มเฉพาะส่วนที่เป็นผลลัพธ์เท่านั้น โดยทำการสุ่มทั้งหมด 49 ตัว เนื่องจากตัวแปรอินพุตทั้งสองอินพุต มีฟังก์ชันการเป็นสมาชิก 7 ฟังก์ชัน ทำให้กฎที่เป็นไปได้ มีทั้งหมดเท่ากับ 49 กฎ โดยกฎจะถูกสร้างมาจากการสุ่ม ตัวอย่างการเข้ารหัสกฎแสดงดังตารางที่ 3.1

ตารางที่ 3.1 ตัวอย่างการเข้ารหัสกฎ

$\begin{matrix} E \\ \Delta E \end{matrix}$	NL(1)	NM(2)	NS(3)	ZE(4)	PS(5)	PM(6)	PL(7)
NL(1)	1	2	4	3	7	5	6
NM(2)	4	3	1	2	5	7	5
NS(3)	3	1	7	6	5	4	1
ZE(4)	6	4	5	7	3	1	2
PS(5)	7	4	3	2	1	5	4
PM(6)	5	4	3	1	2	7	6
PL(7)	1	1	2	4	5	7	6

โปรแกรมจำลองการทำงานของเจเนติกอัลกอริทึม ทำการสุ่มตัวเลขเพื่อนำมาใช้เป็นกฎ ได้เมตริกซ์ที่มีขนาด 7×7 โปรแกรมควบคุมโดยฟิชชิลอจิก จะเรียกใช้งานกฎโดยอ่านจากซ้ายไปขวา และจากบนลงล่าง จากตารางที่ 3.1 กฎของฟิชชิลที่ทำการสุ่มได้ จะถูกนำมาใช้ในโปรแกรมควบคุมฟิชชิลอจิกโดยมีลำดับของกฎเป็นดังต่อไปนี้

กฎที่ 1 : 1, 1, 1	กฎที่ 4 : 4, 1, 3	กฎที่ 7 : 7, 1, 6
กฎที่ 2 : 2, 1, 2	กฎที่ 5 : 5, 1, 7	:
กฎที่ 3 : 3, 1, 4	กฎที่ 6 : 6, 1, 5	กฎที่ 49 : 7, 7, 6

จากตารางที่ 3.1 สามารถเขียนกฎของฟิชชิลอจิกในรูปของข้อความได้เป็น

ถ้า E เป็น NL และ EC เป็น NL แล้ว PC เป็น NL

จำนวนยีนทั้งหมดในโครโมโซมจะเป็น

$$\text{จำนวนยีน} = 1+361+361+361+49 = 1133$$

3.3.2 การประเมินค่า (Evaluation)

การหาคำตอบเป็นกระบวนการหาคำของฟังก์ชันความเหมาะสมของโครโมโซม ในที่นี้ได้ใช้ฟังก์ชันความเหมาะสม เป็นค่าความผิดพลาดที่หาได้จาก

$$\text{ค่าผิดพลาดเฉลี่ย (E)} = \frac{\text{กำลังจากลูกข่าย} - \text{กำลังสถานีฐานต้องการ}}{\text{จำนวนสัญลักษณ์ (symbol)}}$$

กำลังที่สถานีฐานต้องการในงานวิจัยนี้กำหนดให้เป็น 0 dB หรือ 1 W เวลาที่ใช้ในการสุ่ม หาได้จากอัตราการส่งข้อมูล ซึ่งได้ถือว่าการส่งข้อมูลด้วยอัตราสูงสุดของมาตรฐาน IS-95 คือ 9600 bps ในงานวิจัยได้จำลองให้มีการส่งข้อมูลออกไปทั้งหมด 1000 สัญลักษณ์ ซึ่งเวลาที่ใช้ในการส่งข้อมูลออกไปจะเท่ากับ 0.1042 sec

เนื่องจากฟังก์ชันความเหมาะสมที่ใช้ เป็นการหาค่าความผิดพลาด ดังนั้นโครโมโซมที่ให้ค่าความผิดพลาดเฉลี่ยน้อยที่สุด จะเป็นโครโมโซมที่เหมาะสมที่สุด

ในกระบวนการหาคำของฟังก์ชันความเหมาะสม จะทำการแปลงโครโมโซมให้เป็นเมตริกซ์ FIS จากนั้นจึงทำการจำลองระบบ DS/CDMA เพื่อทดลองให้ได้ผลจากการควบคุม แล้วจึงนำไปหาค่าความผิดพลาดเฉลี่ย

3.3.3 การเลือก (Selection)

โดยส่วนใหญ่ มักใช้วิธีหมุนวงล้อ (Roulette Wheel) ในการเลือกโครโมโซม เพื่อนำไปสร้างประชากรรุ่นถัดไป วิธีนี้เป็นการเลือกตามค่าความเหมาะสม ซึ่งขึ้นอยู่กับความน่าจะเป็น การเลือกด้วยวิธีหมุนวงล้อนี้ จะได้ประชากรในรุ่นต่อไปซ้ำกัน กระบวนการเลือกแบบหมุนวงล้อ มีขั้นตอนดังนี้

- (1) หาค่าความเหมาะสม $eval(v_k)$ สำหรับแต่ละโครโมโซม (v_k)

$$eval(v_k) = fitness\ function \quad k = 1, 2, \dots, pop_size$$

- (2) หาค่ารวมของความเหมาะสมของประชากรทั้งหมด

$$F = \sum_{k=1}^{pop_size} eval(v_k)$$

- (3) หาค่าความน่าจะเป็นของการถูกเลือก p_k ของแต่ละโครโมโซม (v_k)

$$p_k = \frac{eval(v_k)}{F} \quad k = 1, 2, \dots, pop_size$$

- (4) หาค่าความน่าจะเป็นสะสม (cumulative probability) q_k ของโครโมโซม (v_k)

$$q_k = \sum_{j=1}^k p_j \quad k = 1, 2, \dots, pop_size$$

กระบวนการเลือกจะทำการหมุนวงล้อ เป็นจำนวนครั้งเท่ากับจำนวนประชากร ในแต่ละครั้งจะได้โครโมโซมหนึ่งอัน มาเป็นประชากรในรุ่นต่อไป กระบวนการเลือกมีขั้นตอนดังนี้

- (1) สุ่มตัวเลข (r) ที่มีค่าระหว่าง $[0,1]$

(2) ถ้าตัวเลขที่สุ่ม (r) มีค่าน้อยกว่า q_1 จะเลือกโครโมโซม v_1 เป็นประชากรรุ่นต่อไป ถ้าตัวเลขที่สุ่มมา มีค่าอยู่ระหว่าง q_1 และ q_2 ($q_1 < r \leq q_2$) จะทำการเลือกโครโมโซมที่ v_2 เป็นประชากรรุ่นต่อไป

เนื่องจากต้องการให้โครโมโซมที่ให้ค่าผิดพลาดน้อยที่สุด มีความน่าจะเป็นที่จะถูกเลือก (p_k) มากที่สุด ในการสร้างโปรแกรมที่ทำหุนวงล้อ เมื่อหาค่าผิดพลาดซึ่งเป็นฟังก์ชันจุดประสงค์ได้แล้ว จึงนำไปหาส่วนกลับ เพื่อทำการบวกให้ได้ค่ารวมของความเหมาะสม (Fitness) ของประชากรทั้งหมด ดังต่อไปนี้

$$F = \sum_{k=1}^{pop\ size} \frac{1}{E_k}$$

ดังนั้นค่าความน่าจะเป็นของการถูกเลือก (p_k) ของแต่ละโครโมโซม หาได้จาก

$$p_k = \frac{1/E_k}{F}$$

3.3.4 การครอสโอเวอร์ (Crossover)

การครอสโอเวอร์ เป็นการสร้างโครโมโซมใหม่ โดยใช้โครโมโซมเก่าสองตัวนำมาสลับตำแหน่งกัน โดยจะใช้วิธีที่เรียกว่า One-cut-point คือสุ่มตำแหน่งที่จะทำการสลับมา 1 จุดจากนั้นทำการสลับยีนที่อยู่ด้านขวามือของจุดนั้น จะได้ลูกขึ้นมาใหม่จำนวนสองตัว โดยที่จำนวนประชากรที่จะถูกทำครอสโอเวอร์จะถูกกำหนดโดย ค่าความน่าจะเป็นของการครอสโอเวอร์ (Crossover Probability) คือ เมื่อทำการสุ่มตัวเลขที่มีค่าอยู่ระหว่าง $[0,1]$ เป็นจำนวนเท่ากับจำนวนประชากรทั้งหมดแล้ว จะนำค่าที่ได้มาเปรียบเทียบกับค่าความน่าจะเป็นของการครอสโอเวอร์ ถ้าค่าที่สุ่มได้ มีค่าน้อยกว่า หรือเท่ากับ ก็จะเลือกเอาโครโมโซมที่ลำดับเดียวกันนั้นมาเป็นพ่อและแม่ในการทำครอสโอเวอร์ ในงานวิจัยนี้จะใช้การครอสโอเวอร์ระหว่างโครโมโซมที่มีรูปร่างของฟังก์ชันการเป็นสมาชิกเหมือนกัน และจะทำการครอสสองครั้ง คือครั้งแรก ทำการครอสจุดที่ใช้สำหรับการสร้างฟังก์ชันการเป็นสมาชิก และครั้งที่สองคือ ทำการครอสกันที่กฎ

ยกตัวอย่างเช่น มีประชากรที่มีรูปร่างของฟังก์ชันการเป็นสมาชิกเป็นรูปสามเหลี่ยมจำนวน 10 ตัว ในการครอสโอเวอร์จะสุ่มตัวเลขที่อยู่ระหว่าง $[0,1]$ จำนวน 10 ตัว สมมติว่าได้ค่าเป็นดังนี้

$$\begin{aligned} r1=0.9501 \quad r2=0.2311 \quad r3=0.6068 \quad r4=0.4860 \quad r5=0.8913 \\ r6=0.7621 \quad r7=0.4565 \quad r8=0.0185 \quad r9=0.8214 \quad r10=0.4447 \end{aligned}$$

กำหนดให้ค่าความน่าจะเป็นของการครอสโอเวอร์เป็น 0.25 ดังนั้น โครโมโซมที่จะนำมาใช้เป็นพ่อและแม่ จึงเป็นโครโมโซมที่ 2 และ 8 สมมติว่าโครโมโซมที่ 2 และ 8 เป็นดังนี้

$$\begin{aligned} v_2 &= [800 \quad -17.70000 \quad -17.20000000 \quad -16.4000 \quad -16....2475....5] \quad 1133 \text{ ยีน} \\ v_8 &= [80000 \quad -17.500000 \quad -16.9000 \quad -16.5000 \quad -16.10....7345....2] \end{aligned}$$

จากโครโมโซมที่ได้ เราจะนำไปสร้างฟังก์ชันการเป็นสมาชิกโดยใช้ตำแหน่งที่บิตนั้นมีค่าไม่เป็นศูนย์ เฉพาะจุดที่จะนำมาสร้างฟังก์ชันการเป็นสมาชิกที่จะเอามาทำ ครอสโอเวอร์เป็นดังนี้

$$\begin{aligned} v_2^* &= [-17.7 \quad -17.2 \quad -16.4 \quad -16.0 \quad2475....5] \quad 112 \text{ ยีน} \\ v_8^* &= [-17.5 \quad -16.9 \quad -16.5 \quad -16.1 \quad7345....2] \end{aligned}$$

จากนั้นจึงทำการสุ่มจุดที่จะทำครอสโอเวอร์ครั้งที่หนึ่ง โดยจะต้องเป็นตัวเลขที่ข้ามทีละ 3 ตัว คือ 1, 3, 5, 7, 9, ..., 63 สมมติว่าสุ่มได้ 3 คือจะทำการครอสกันที่ตำแหน่งที่ 3 โดยการสลับกันตั้งแต่ตำแหน่งที่ 4 เป็นต้นไป ดังนั้นจะได้โครโมโซมรุ่นลูกตัวใหม่ เป็นดังนี้

$$\begin{aligned} v_2^{**} &= [-17.7 \quad -17.2 \quad -16.4 \quad \boxed{-16.1 \quad} \quad 2475....5] \quad 112 \text{ ยีน} \\ v_8^{**} &= [-17.5 \quad -16.9 \quad -16.5 \quad \boxed{-16.0 \quad} \quad 7345....2] \end{aligned}$$

จากนั้นจึงทำการสุ่มจุดที่จะทำครอสโอเวอร์ครั้งที่สอง เพื่อทำการครอสกฎ หลังจากนั้นจะได้โครโมโซมในรุ่นถัดไปเป็นดังนี้

$$\begin{aligned} v_2^{**} &= [-17.7 \quad -17.2 \quad -16.4 \quad \boxed{-16.1 \quad} \quad 24 \quad \boxed{45....2}] \quad 112 \text{ ยีน} \\ v_8^{**} &= [-17.5 \quad -16.9 \quad -16.5 \quad \boxed{-16.0 \quad} \quad 73 \quad \boxed{75....5}] \end{aligned}$$

นำโครโมโซมที่ได้ใหม่ทั้งสองตัว กลับเข้าไปแทนโครโมโซมพ่อและแม่

3.3.5 การมิวเตชัน (Mutation)

การทำมิวเตชัน เป็นลักษณะของการกลายพันธุ์ ซึ่งหมายถึงมีการเปลี่ยนค่าของยีนไปจากเดิม ทุกยีนสามารถถูกทำมิวเตชันได้ ไม่แน่นอนว่าเป็นตำแหน่งใด ดังนั้นในการเขียนโปรแกรม จึงต้องสุ่มตำแหน่งของยีนที่จะทำมิวเตชัน โดยมีจำนวนการสุ่มเท่ากับจำนวนยีนทั้งหมดรวมกันทุกโครโมโซม

โครโมโซมที่ใช้ในงานวิจัยนี้เหมาะแก่การใช้มิวเตชันแบบไม่สม่ำเสมอ (Non uniform) เนื่องจากการเปลี่ยนตำแหน่งของฟังก์ชันการเป็นสมาชิก ซึ่งสามารถเปลี่ยนได้แบบไม่สม่ำเสมอ โดยสมมติให้ สุ่มการทำมิวเตชันได้ที่ x_k สำหรับโครโมโซมที่ v_k โครโมโซมใหม่ที่ได้จากการมิวเตชันจะเป็น

$$v_k = [x_1, \dots, x'_k, \dots, x_{1133}]$$

เมื่อ x' เป็นค่าที่สุ่ม จากตัวเลือกสองตัวเลือกดังต่อไปนี้

$$\begin{aligned} x'_1 &= x_k + \Delta(t, x_k^U - x_k) \\ x'_2 &= x_k + \Delta(t, x_k - x_k^L) \end{aligned} \quad (3.12)$$

โดยที่ x_k^U และ x_k^L เป็นยีนก่อนหน้า และหลังของ x_k ตามลำดับ ฟังก์ชัน $\Delta(t, y)$ ให้ค่าที่อยู่ในช่วง $[0, y]$ ค่าของฟังก์ชัน $\Delta(t, y)$ จะเข้าใกล้ศูนย์เมื่อ t เพิ่มขึ้น โดยที่ t เป็นรอบของการสร้างโครโมโซม ฟังก์ชัน $\Delta(t, y)$ หาได้จากสมการต่อไปนี้

$$\Delta(t, y) = y \cdot r \cdot \left(1 - \frac{t}{T}\right) \quad (3.13)$$

เมื่อ r เป็นค่าสุ่มจาก $[0, 1]$
 T เป็นรอบสูงสุดของการสร้างโครโมโซม

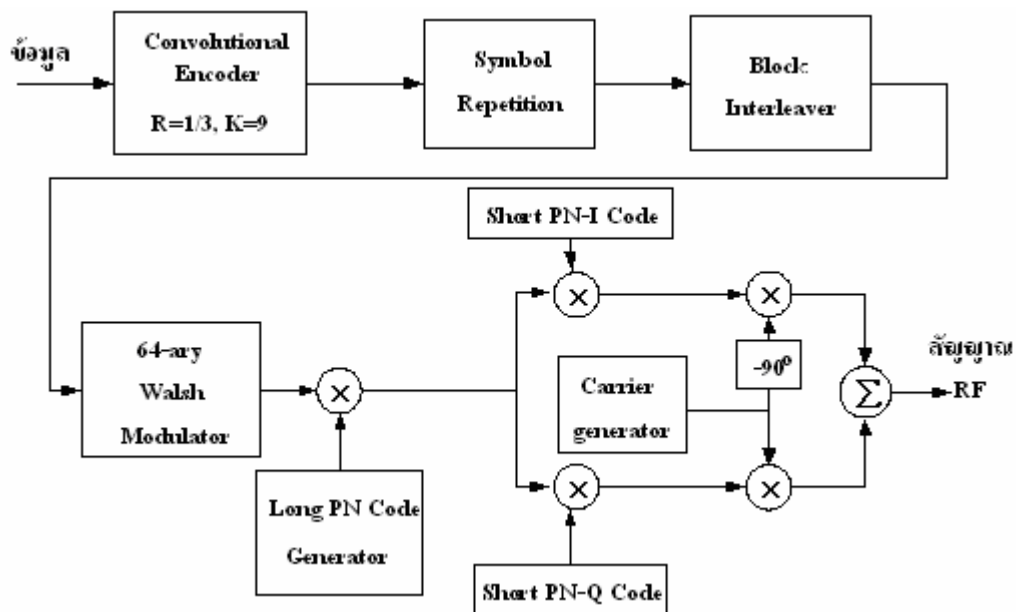
ในการเขียนโปรแกรมเมื่อทำการมิวเตชัน จะนำยีนของโครโมโซมทุกอันมาเรียงต่อกัน โดยตัดยีนที่เป็นศูนย์ ยีนส่วนของกฎและยีนส่วนรูปร่างออกไปก่อน และกำหนดตัวแปรใหม่ ให้แทน

ยื่นทั้งหมดที่นำมาเรียงต่อกัน ซึ่งกำหนดให้มีความยาวมากที่สุดแต่ละช่วงโครโมโซมเท่ากับ 84 บิต เนื่องจากเมื่อนำไปเขียนโปรแกรมหาอินที่ถูกรหัสเป็นโครโมโซมที่เท่าใด จะทำได้ง่ายขึ้น

เมื่อทำการรหัสในส่วนของฟังก์ชันการเป็นสมาชิกแล้ว ต่อไปทำการรหัสของส่วนกฎ โดยใช้โครโมโซมเดียวกับที่ทำรหัสฟังก์ชันการเป็นสมาชิก ตำแหน่งของอินที่ถูกรหัสส่วนกฎ ได้มาจากการสุ่มตัวเลขระหว่าง 0 – 1 แล้วเปรียบเทียบกับอัตราการรหัส ถ้าค่าที่สุ่มได้มีค่าน้อยกว่าจะทำการเปลี่ยนค่าของอินนั้น โดยค่าที่เปลี่ยนไปของอิน ได้มาจากการสุ่มค่า ตั้งแต่ 1 ถึง 7

3.4 โปรแกรมคอมพิวเตอร์จำลองระบบ DS/CDMA ในด้านย้อนกลับ

3.4.1 ขั้นตอนการสร้างสัญญาณในระบบ DS/CDMA ด้านย้อนกลับ



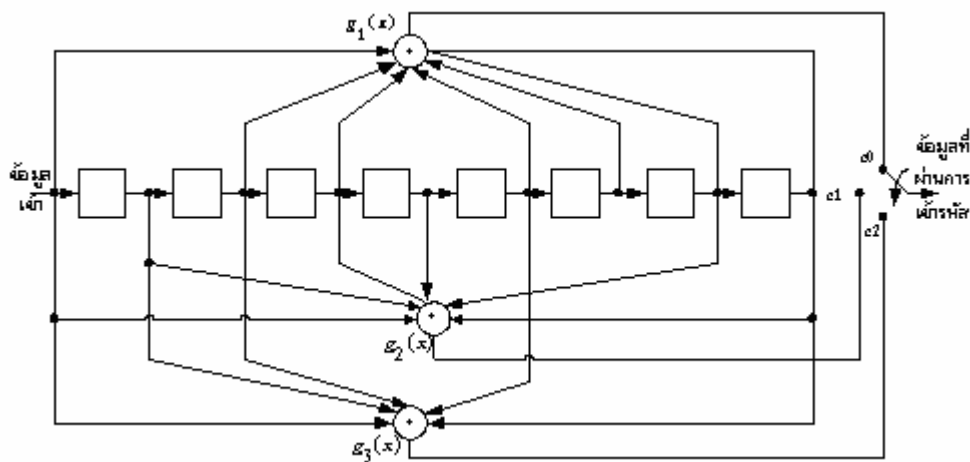
รูปที่ 3.11 ขั้นตอนในการสร้างสัญญาณในทิศทางจากโทรศัพท์ไปยังสถานีฐาน

ขั้นตอนในการสร้างสัญญาณสำหรับส่งผ่านจากโทรศัพท์เคลื่อนที่ไปยังสถานีฐานในงานวิจัยนี้ เริ่มต้นจาก นำข้อมูลมาผ่านกระบวนการเข้ารหัส แบบคอนวอลูชันที่มีอัตราการเข้ารหัส (R) เป็น 1/3 คือข้อมูลหนึ่งบิต แทนด้วยสัญลักษณ์ (symbol) 3 บิต และใช้จำนวนชุดของข้อมูลที่เข้ารหัสในแต่ละรอบ (constraint length : K) ทั้งหมด 9 ตัว จากนั้นนำข้อมูลซึ่งเพิ่มขึ้นเป็นสามเท่า

ไปผ่านกระบวนการทำซ้ำ (symbol repetition) ทำให้ได้ข้อมูลเพิ่มขึ้นอีกเป็น 2 เท่า แล้วนำไปผ่านกระบวนการอินเตอร์ลีฟเวอร์เพื่อลดการผิดพลาดแบบเบิร์ส จากนั้นนำไปเข้ารหัสโดยการมอดูเลตกับโค้ด Walsh 64 แบบ แล้วนำไปมอดูเลตกับโค้ด PN แบบยาว แล้วนำไปมอดูเลตกับแบบ QPSK กับคลื่นพาห์ แล้วจึงทำการส่งออกไปยังสถานีฐาน โดยผ่านช่องสัญญาณที่มีการเฟดดิ้ง และมี AWGN ขั้นตอนการสร้างสัญญาณแสดงดังรูปที่ 3.11

1) การเข้ารหัสคอนโวลูชันนอล (Convolutional code)

ในการเข้ารหัสคอนโวลูชันของการส่งสัญญาณในด้านย้อนกลับ มีวงจรเข้ารหัสเป็นดังรูปที่ 3.12



รูปที่ 3.12 วงจรเข้ารหัสแบบคอนโวลูชัน

การเข้ารหัสแบบคอนโวลูชัน มีค่าพารามิเตอร์ทั้งหมด 3 ค่า คือ n , k และ K โดยที่ K คือ จำนวนชุดของสัญลักษณ์ที่ใช้ในการเข้ารหัสแต่ละรอบ (constraint length) k คือ จำนวนของบิตต่อหนึ่งสัญลักษณ์ เช่น ถ้าข้อมูลเข้าเป็นเลขฐานสอง ค่า $k=1$ และถ้าข้อมูลเข้าเป็นอะเรียรี 4 บิต ค่า $k=2$ kK คือ จำนวนชิฟต์รีจิสเตอร์ที่ต้องใช้ n คือ จำนวนบิตข้อมูลที่ผ่านกระบวนการเข้ารหัสแล้ว k/n คือ อัตราการเข้ารหัส

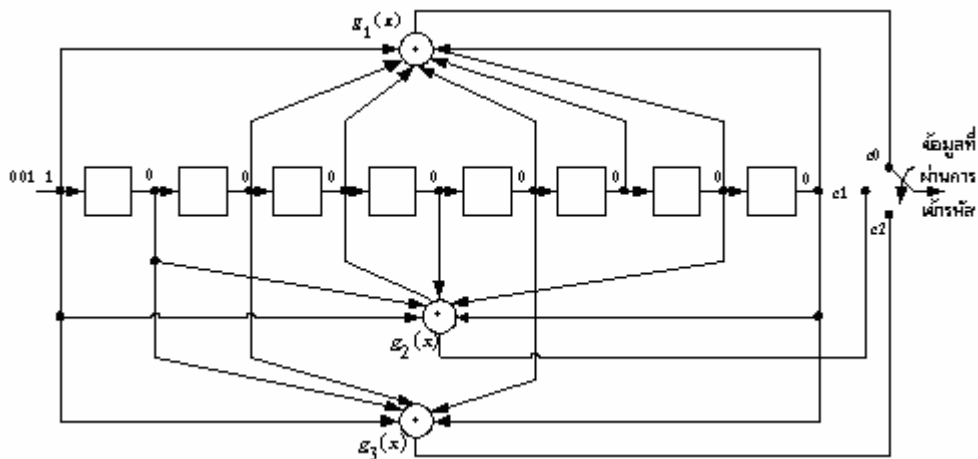
ในงานวิจัยนี้ เป็นการเข้ารหัสโดยใช้ $k=1$ และใช้ชิฟต์รีจิสเตอร์ 8 ตัว มีอัตราการเข้ารหัสเท่ากับ $1/3$ เนื่องจาก $k=1$ ดังนั้นจำนวนบิตที่จะเลื่อนเข้าสำหรับการเข้ารหัสแต่ละครั้งมีค่าเท่ากับ 1 บิต จากรูปที่ 3.12 ประกอบไปด้วยวงจรกำเนิดโพลิโนเมียลสามชุดคือ $g_1(x)$, $g_2(x)$ และ $g_3(x)$ โดยที่เขียนเป็นสมการโพลิโนเมียลได้ดังนี้

$$g_1(x) = 1 + x^2 + x^3 + x^5 + x^6 + x^7 + x^8$$

$$g_2(x) = 1 + x + x^3 + x^4 + x^7 + x^8$$

$$g_3(x) = 1 + x + x^2 + x^5 + x^8$$

โปรแกรมคอนโวลูชันจะทำการเลื่อนบิตข้อมูลเข้าไปที่ละหนึ่งบิต จากซ้ายไปขวา ในรูปที่ 3.12 แล้วทำการบวกแบบมอดูโล 2 คือ $1+1 = 0$, $0+1 = 1$ และ $1+0 = 1$ ยกตัวอย่างเช่น ข้อมูลที่ต้องการเข้ารหัสเป็น 1100 มีทั้งหมด 4 บิต เนื่องจากใช้การเข้ารหัสคอนโวลูชันแบบ $1/3$ ดังนั้น 1 บิตข้อมูล จะถูกแทนด้วยสัญลักษณ์ 3 บิต หลังจากการเข้ารหัสแล้วจะได้ข้อมูลทั้งหมด 12 บิต ตามขั้นตอนดังนี้



รูปที่ 3.13 (ก) การเข้ารหัสของบิตข้อมูลตัวแรก ‘1’

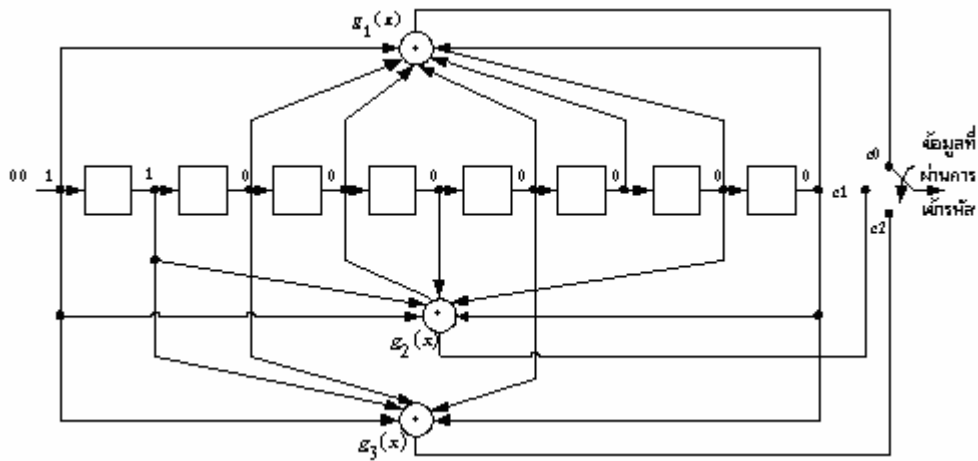
$$g_1(x) = 1 + 0 + 0 + 0 + 0 + 0 + 0 = 1$$

$$g_2(x) = 1 + 0 + 0 + 0 + 0 + 0 = 1$$

$$g_3(x) = 1 + 0 + 0 + 0 + 0 = 1$$

ข้อมูลบิตแรกคือ 1 เมื่อผ่านการเข้ารหัสแล้วจะได้สัญลักษณ์แทน 1 เป็น 111 ดังรูปที่

3.13 (ก)



รูปที่ 3.13 (ข) การเข้ารหัสของบิตข้อมูลตัวที่สอง '1'

$$g_1(x) = 1 + 0 + 0 + 0 + 0 + 0 + 0 = 1$$

$$g_2(x) = 1 + 1 + 0 + 0 + 0 + 0 + 0 = 0$$

$$g_3(x) = 1 + 1 + 0 + 0 + 0 + 0 + 0 = 0$$

ข้อมูลบิตที่สองคือ 1 เมื่อผ่านการเข้ารหัสแล้วจะได้สัญลักษณ์แทน 1 เป็น 100 ดังรูป

ที่ 3.13 (ข)

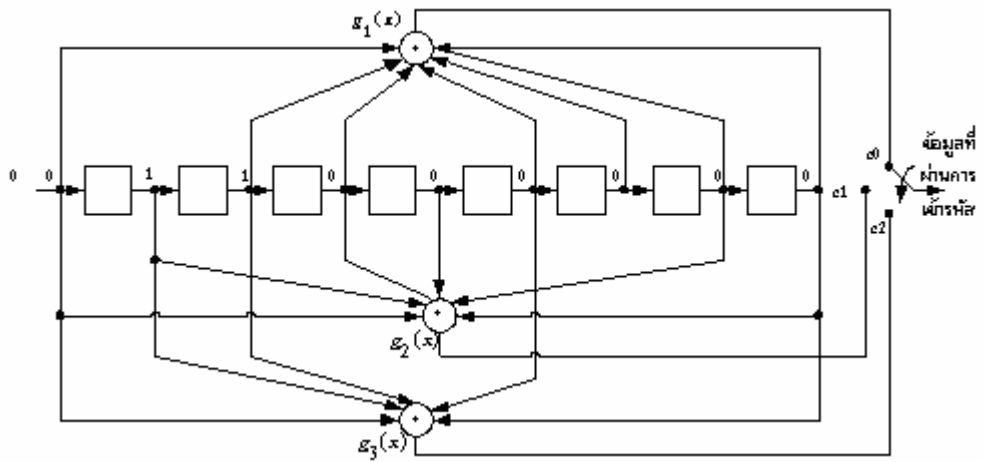
ข้อมูลบิตที่สามคือ 0 เมื่อผ่านการเข้ารหัสแล้วจะได้สัญลักษณ์แทน 0 เป็น 110 ดังรูป

ที่ 3.13 (ค)

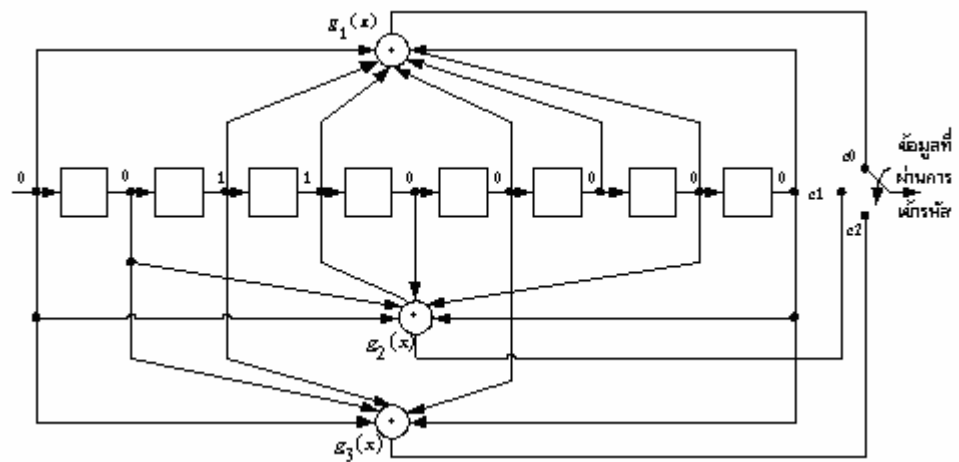
$$g_1(x) = 0 + 1 + 0 + 0 + 0 + 0 + 0 = 1$$

$$g_2(x) = 0 + 1 + 0 + 0 + 0 + 0 + 0 = 1$$

$$g_3(x) = 0 + 0 + 1 + 1 + 0 + 0 + 0 = 0$$



รูปที่ 3.13 (ค) การเข้ารหัสของบิตข้อมูลตัวที่สาม '0'



รูปที่ 3.13 (ง) การเข้ารหัสของบิตข้อมูลตัวที่สี่ '0'

$$g_1(x) = 0 + 1 + 1 + 0 + 0 + 0 + 0 = 0$$

$$g_2(x) = 0 + 0 + 1 + 0 + 0 + 0 = 1$$

$$g_3(x) = 0 + 0 + 1 + 0 + 0 = 1$$

ข้อมูลบิตที่สี่คือ 0 เมื่อผ่านการเข้ารหัสแล้วจะได้สัญลักษณ์แทน 0 เป็น 011 ดังรูปที่ 3.13 (ค) ดังนั้นข้อมูลสี่บิตที่ต้องการส่งออกไป คือ 1100 เมื่อทำการเข้ารหัสคอนโวลูชันแล้วจะได้ข้อมูลที่ออกไปเป็น 111 100 110 011

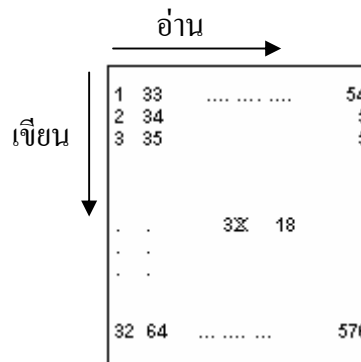
2) การทำซ้ำ (symbol repetition)

เป็นการส่งบิตข้อมูลเดิมออกไปทั้งหมดสองครั้งติดต่อกัน แบบบิตต่อบิต ยกตัวอย่างเช่น ถ้าบิตข้อมูลที่ออกมาจากการเข้ารหัสคอนโวลูชันเป็น 111 100 110 011 เมื่อผ่านโปรแกรมการทำซ้ำ จะได้ข้อมูลที่ส่งออกไป เป็นดังนี้

111111 110000 111100 001111

3) การทำอินเตอร์ลีฟเวอร์ (block interleaver)

การทำอินเตอร์ลีฟเวอร์ ก็เพื่อจะลดการผิดพลาดของข้อมูล โดยนำข้อมูลทีละ 576 บิต มาจัดเป็น 32 แถว และ 18 คอลัมน์ ดังในรูปที่ 3.14 ในการจัดรูปแบบจะกระทำในแนวตั้ง แต่ในการอ่านข้อมูล เป็นการอ่านทีละแถวในแนวนอน



รูปที่ 3.14 การทำอินเตอร์ลีฟเวอร์

4) การเข้ารหัส (coding)

ในระบบวังผึ้งที่ใช้การส่งข้อมูลแบบ DS/CDMA ด้านย้อนกลับ มีการเข้ารหัส สองแบบ คือแบบแรกเป็นการเข้ารหัสโดยใช้รหัส Walsh แบบที่สองคือการเข้ารหัสโดยใช้รหัส PN แบบสั้น และ PN แบบยาว รายละเอียดทั้งสองแบบ มีดังนี้

4.1) การเข้ารหัสกับรหัส Walsh

หลังจากข้อมูลผ่านกระบวนการอินเทอร์ลีฟวิ่ง ข้อมูลจะถูกส่งออกไปเป็นแบบแถวทีละ 6 บิต ซึ่งจะถูกคำนวณเป็นค่าเลขฐานสิบ เพื่อใช้เลขฐานสิบในการระบุรหัส Walsh ที่ใช้แทนข้อมูลทั้ง 6 บิตนั้น เนื่องจากความเป็นไปได้ของข้อมูล 6 บิต มีทั้งหมด $2^6 = 64$ ดังนั้นเราจึงสามารถแทนข้อมูล 6 บิตด้วยรหัส Walsh ซึ่งมีทั้งหมด 64 ชุด (W0-W64) ยกตัวอย่างเช่น ถ้าข้อมูลที่ส่งออกมาจากอินเทอร์ลีฟวิ่ง เป็น $000\ 011_2 = 3_{10}$ ดังนั้นรหัส Walsh ที่เลือกใช้คือรหัส W3 ซึ่งมีลำดับบิตดังต่อไปนี้

1100 1100 1100 1100 1100 1100 1100 1100 1100 1100 1100 1100 1100 1100 1100 1100

4.2) การเข้ารหัสกับรหัส PN

เมตริกซ์กำเนิด (generate metric) ของรหัส PN แบบยาว คือ

$$P = [1\ 1\ 1\ 0\ 1\ 1\ 1\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 0\ 1\ 1\ 0\ 0\ 1\ 1\ 1\ 0\ 0\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 1]$$

รหัส PN แบบยาว ซึ่งมีทั้งหมด 42 บิต จะถูกนำไปกระทำลอจิก and กับ รหัสของผู้ใช้งานแต่ละคน ซึ่งมีทั้งหมด 42 บิต เราเรียกรหัสของผู้ใช้นี้ว่าเป็นรหัสมาสก์แบบยาว (long code mask) จากนั้นจะนำรหัสที่ได้ไปคูณ กับข้อมูลที่ส่งมาจากกระบวนการเข้ารหัส Walsh

จากนั้น ข้อมูลที่ผ่านกระบวนการเข้ารหัส Walsh แล้วจะนำมาคูณกับเมตริกซ์กำเนิดของรหัส PN แบบสั้น ชุด I ซึ่งมี 15 บิต มีลำดับบิตคือ

$$I = [0\ 0\ 0\ 0\ 1\ 0\ 1\ 1\ 1\ 0\ 0\ 0\ 1\ 0\ 1]$$

และข้อมูลชุดเดียวกัน ที่ผ่านกระบวนการเข้ารหัส Walsh มานี้จะถูกนำไปคูณ กับเมตริกซ์กำเนิดของรหัส PN แบบสั้น ชุด Q ซึ่งมีจำนวน 15 บิต มีลำดับบิตดังนี้

$$Q = [0\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 0\ 1\ 1\ 1\ 0\ 0\ 1]$$

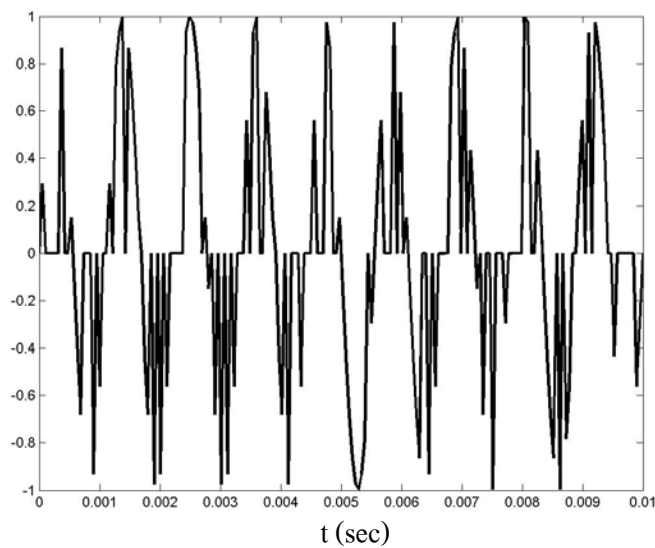
เมื่อผ่านกระบวนการนี้แล้ว ทำให้ได้ข้อมูลที่แตกต่างกันสองชุด คือชุดที่คูณกับ I และชุดที่คูณกับ Q

5) มอดูเลชัน (modulation)

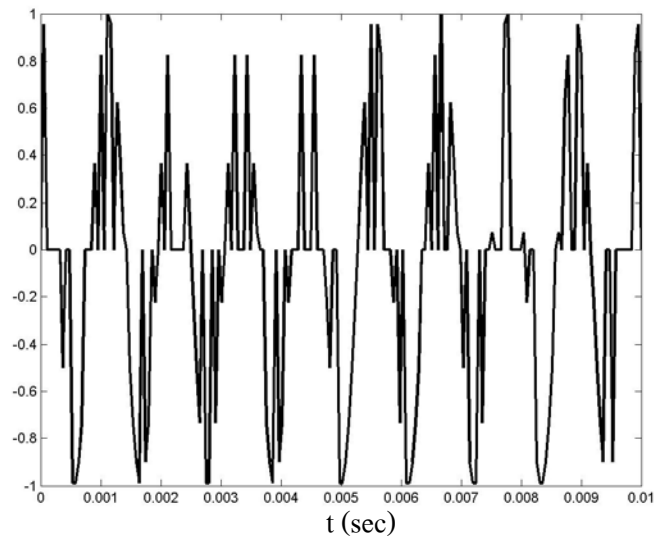
ตามมาตรฐาน IS-95 การทำมอดูเลชัน ในด้านย้อนกลับของระบบรังสีที่ใช้ DS/CDMA จะใช้การมอดูเลชันกับสัญญาณพาห์ แบบ QPSK (quadrature phase shift keying) โดยนำสัญญาณที่ผ่านการคูณกับรหัส PN ชุด I มาคูณกับสัญญาณโคไซน์ ที่กำหนดให้ความถี่คลื่นพาห์เป็น 900 MHz และสัญญาณที่ผ่านการคูณกับรหัส PN ชุด Q มาคูณกับสัญญาณไซน์ที่มีความถี่ 900 MHz ถ้ามีข้อมูลที่จะทำการมอดูเลชันเป็น

```
00010001111010101101100000010001010100011101000001001100011111100010100110010
00110100001111101011000110001011100000001110100010110001001010011010110111011
01100100011000100100100111110011000011
```

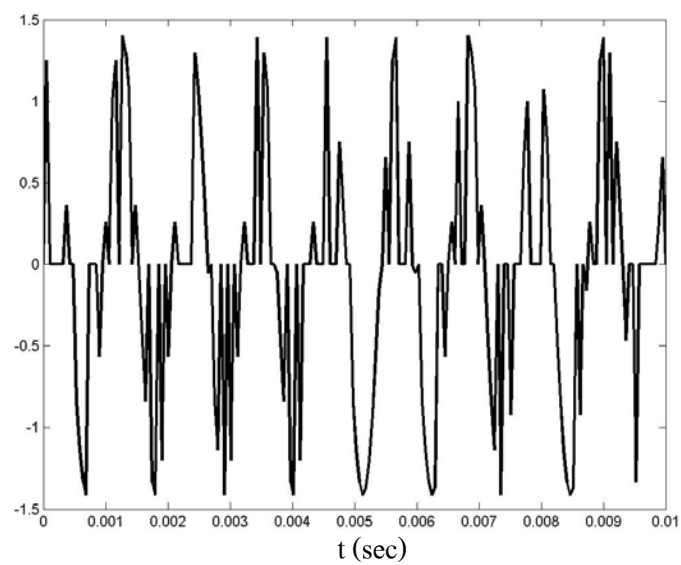
เมื่อนำข้อมูลทำการมอดูเลตกับสัญญาณไซน์ และโคไซน์ จะได้สัญญาณดังรูปที่ 3.15 และ รูปที่ 3.16 ตามลำดับ



รูปที่ 3.15 สัญญาณที่ได้หลังจากมอดูเลตกับสัญญาณไซน์ที่มีความถี่ 900 MHz



รูปที่ 3.16 สัญญาณที่ได้หลังจากมอดูเลตกับสัญญาณ โคไซน์ที่มีความถี่ 900 MHz



รูปที่ 3.17 รูปสัญญาณที่ส่งออกจากโทรศัพท์เคลื่อนที่

ขั้นตอนสุดท้ายก่อนทำการส่งสัญญาณออกไป คือการนำสัญญาณที่ได้จากการมอดูเลตกับสัญญาณไซน์และโคไซน์มาบวกกัน ซึ่งจะได้รูปสัญญาณที่ออกไปจากโทรศัพท์เคลื่อนที่ที่เป็นดังรูปที่ 3.17

3.4.2 แบบจำลองช่องสัญญาณเฟดดิ้งแบบเรย์ลีเฟดดิ้ง

ช่องสัญญาณที่เกิดการเฟดดิ้งตามธรรมชาติ เกิดขึ้นเนื่องจากสภาพแวดล้อมที่มีการเปลี่ยนแปลงไป ยกตัวอย่างเช่น การเดิน หรือการที่มีวัตถุเคลื่อนที่ผ่าน เช่น รถยนต์ สามารถแทนได้ด้วยการจำลองในหลายรูปแบบ รูปแบบหนึ่งที่มีความใกล้เคียงกับเฟดดิ้งในธรรมชาติ และเป็นที่ยอมรับใช้ในการสร้างแบบจำลองของช่องสัญญาณคือ เรย์ลีเฟดดิ้ง ซึ่งในการสร้างแบบจำลองนี้สามารถทำได้ โดยนำจำนวนเชิงซ้อนที่สุ่มแบบเกาส์เซียน มาผ่านบัตเตอร์เวิร์ธฟิลเตอร์ (butterworth filter)

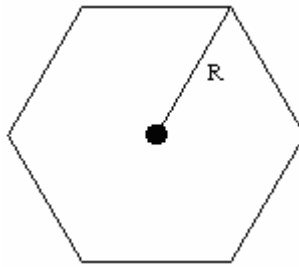
3.4.3 แบบจำลองการกระจายตัวของโทรศัพท์เคลื่อนที่

แบบจำลองจำนวนเซลล์ที่ใช้ช่วงความถี่เดียวกัน มีทั้งหมด 19 เซลล์แสดงดังรูปที่ 3.18 แต่ละเซลล์มีสถานีฐานตั้งอยู่ตรงกลางเซลล์ พื้นที่ (Area) ของแต่ละเซลล์รูปร่างสี่เหลี่ยมสามารถคำนวณได้จากสมการที่ (3.14)

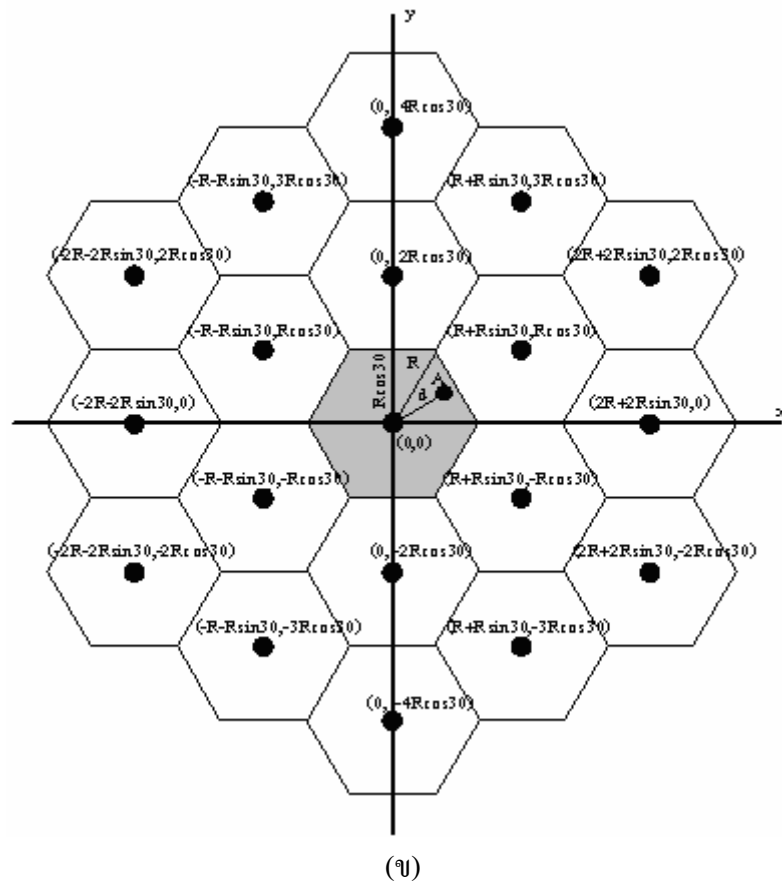
$$\text{Area} = 2.6R^2 \quad (3.14)$$

รูปที่ 3.18 (ก) แสดงเส้นรัศมี (R) ของรูปเซลล์หกเหลี่ยม รูปที่ 3.18 (ข) ตำแหน่งของสถานีฐานอยู่ที่พิกัด (0,0) ตำแหน่งของโทรศัพท์เคลื่อนที่ ได้มาจากการสุ่มให้อยู่ในรูปหกเหลี่ยมสี่เท่า แล้วทำการส่งตำแหน่งไปยังเซลล์ต่างๆ โดยการบวกตำแหน่งที่สุ่มได้ด้วยตำแหน่งของสถานีฐานแต่ละเซลล์เข้าไป

ตำแหน่ง A ในเซลล์สี่เท่า แทนตำแหน่งของโทรศัพท์เคลื่อนที่ที่กำลังสนใจ ซึ่งอยู่ห่างจากสถานีฐาน เป็นระยะทาง d สัญญาณจากโทรศัพท์เคลื่อนที่อื่นๆ จะถูกพิจารณาว่าเป็นสัญญาณแทรกสอด



(ก) การวัดรัศมีของเซลล์รูปหกเหลี่ยม



(ข)

รูปที่ 3.18 (ก) การวัดรัศมีของเซลล์รูปหกเหลี่ยม (ข) แบบจำลองการแบ่งพื้นที่เซลล์ และตำแหน่งของสถานีฐานแต่ละเซลล์

3.4.4 แบบจำลองการลดทอนสัญญาณเนื่องจากระยะทาง

แบบจำลองการลดทอนกำลังเนื่องจากระยะทางจากผู้ใช้ถึงสถานีฐาน และสัญญาณเดินทางตกกระทบและสะท้อนในหลายทิศทาง ใช้แบบจำลองของ Hata ซึ่งเป็นที่ยอมรับให้เป็น

มาตรฐานในการคำนวณกำลังที่ลดลง แบบจำลองของ Hata ดังแสดงในสมการที่ (2.2) งานวิจัยนี้ใช้แบบจำลองความสูงของเสาอากาศที่สถานีฐานโดยเฉลี่ยเป็น 40 m และ ความสูงของเสาอากาศโทรศัพท์เคลื่อนที่วัดจากพื้นโดยเฉลี่ยคือ 3 m กำลังของสัญญาณที่ลดทอนลงไปจึงคำนวณได้จากการแทนสมการดังนี้

$$\begin{aligned} L_p &= 69.55 + 26.16 \log_{10}(900 \times 10^6) + (44.9 - 6.55 \log_{10}(40)) \log_{10}(d) \\ &\quad - 13.82 \log_{10}(40) - 3.2[\log_{10}(11.75 \times 3)]^2 - 4.97 \\ &= 112.0627 + [34.4065 \times \log_{10}(d)] \end{aligned} \quad (3.15)$$

เมื่อ L_p เป็นกำลังที่สูญเสียไปเนื่องจากการเดินทาง [dB]

ระยะทาง d สามารถหาได้จากการสุ่มตำแหน่งของโทรศัพท์เคลื่อนที่ในแต่ละเซลล์ แล้วทำการคำนวณหากำลังที่สูญเสียไป กำลังที่สถานีฐานจะได้รับจะหาได้จากสมการที่ (3.16)

$$\text{กำลังที่สถานีฐานได้รับผิดพลาดไปจากที่ต้องการ} = P_T - (P_F + L_p) \quad (3.16)$$

เมื่อ P_T เป็นกำลังที่โทรศัพท์เคลื่อนที่ส่งออกมา กำหนดให้เป็น 1 [W]

P_F เป็นกำลังที่เปลี่ยนไปเนื่องจากการเกิดเฟดดิ้ง [W]

L_p เป็นกำลังที่เปลี่ยนไปเนื่องจากการสูญเสียเนื่องจากการเดินทาง [W]

กำลังที่สถานีฐานรับได้จากโทรศัพท์เคลื่อนที่ที่อยู่รอบๆ ทำให้เกิดสัญญาณแทรกสอดซึ่งจะมารวมกันที่สถานีฐานของเซลล์สีเทา จะนำมาเปรียบเทียบเป็นอัตราส่วนกับกำลังที่สถานีฐานได้รับเนื่องจากโทรศัพท์เคลื่อนที่ตัวที่กำลังพิจารณา (จุด A) เพื่อหา SIR (signal to interference ratio) จากสมการที่ (3.17)

$$SIR = \frac{S}{I} \quad (3.17)$$

เมื่อ S แทนกำลังของโทรศัพท์เคลื่อนที่ที่กำลังพิจารณา [W]

I แทนกำลังของโทรศัพท์เคลื่อนที่ที่ทำให้เกิดสัญญาณแทรกสอด [W]

ค่า SIR ที่ได้ นำมาเปลี่ยนหน่วยเป็น dB แล้วทำการเปรียบเทียบกับ -15 dB ถ้าค่า SIR มีค่าน้อยกว่า -15 dB แสดงว่าสัญญาณของโทรศัพท์เคลื่อนที่ที่กำลังพิจารณามีค่าอ่อนกว่าสัญญาณแทรกสอดจากโทรศัพท์เคลื่อนที่ ที่อยู่บริเวณรอบๆ มากกว่าค่าที่ยอมรับได้ นั่นคือ คุณภาพของการสื่อสารลดลง

จำนวนครั้งที่ SIR มีค่าน้อยกว่า -15 dB สามารถนำมาใช้หาความน่าจะเป็นของความน่าจะเป็นที่จะใช้งานไม่ได้ (outage probability) เนื่องจากจำนวนครั้งในการส่งสัญญาณมีทั้งหมด 1,000 ครั้ง ดังนั้นความน่าจะเป็น จึงมีสมการเป็นดังต่อไปนี้

$$\text{Outage probability} = \frac{N_{SIR < -15dB}}{N_A} \quad (3.18)$$

เมื่อ N_{SIR} เป็นจำนวนครั้งที่ SIR มีค่าน้อยกว่า -15 dB
 N_A เป็นจำนวนข้อมูลทั้งหมด

3.5 โปรแกรมคอมพิวเตอร์ทดสอบตัวควบคุม FGa

ในส่วนนี้เป็นการทำงานร่วมกันระหว่างโปรแกรมควบคุมฟuzzyลอจิก กับโปรแกรมเจเนติกอัลกอริทึม โดยรับสัญญาณอินพุตมาจากแบบจำลองของ DS/CDMA ด้านย้อนกลับ ซึ่งมีสองอินพุต คือ ค่าความผิดพลาดของกำลังที่อุปกรณ์โทรศัพท์เคลื่อนที่ส่งออกมาเทียบกับกำลังที่สถานีฐานต้องการได้รับที่เวลาปัจจุบัน และ ค่าความแตกต่างระหว่างค่าผิดพลาดของกำลังส่งที่เวลาก่อนหน้าเทียบกับค่าผิดพลาดที่เวลาปัจจุบัน จากนั้นจะนำค่าผิดพลาดที่ได้ มาตัดสินใจโดยใช้กฎของฟuzzy ซึ่งได้มาจากการสุ่มตามหลักการของเจเนติกอัลกอริทึม แล้วส่งกำลังที่ต้องการให้โทรศัพท์เคลื่อนที่เพิ่มหรือลดกำลังส่งออกไปเป็นเอาต์พุต ฟังก์ชันความเหมาะสมของเจเนติกอัลกอริทึม ในงานวิจัยนี้ได้ใช้ความผิดพลาดเฉลี่ยของแต่ละรอบ เพื่อนำมาเปรียบเทียบกัน แล้วหาโครโมโซมชุดที่ให้ค่าผิดพลาดน้อยที่สุด หลังจากนั้นจะนำโครโมโซมชุดที่ได้ มาทำการครอสโอเวอร์ และมิวเตชันเพื่อให้ได้โครโมโซมใหม่ แล้วนำโครโมโซมใหม่ไปทดลองกับแบบจำลองระบบ DS/CDMA ด้านย้อนกลับ ทำเช่นนี้ไปเรื่อยๆ จนกระทั่งครบจำนวนรอบที่กำหนดไว้ ซึ่งกำหนดเป็น 1,000 รอบ หรือจนกว่าจะมีค่าผิดพลาดเฉลี่ยน้อยกว่า 4 หรือจนกว่าโครโมโซมจะไม่สามารถเปลี่ยนไปได้อีก

3.6 เงื่อนไขของแบบจำลองที่ใช้ในงานวิจัย

แบบจำลองที่ใช้ในงานวิจัยนี้ กำหนดให้ใช้ค่าต่างๆ เหมือนกับ [21] เพื่อทำการเปรียบเทียบผลการควบคุมที่ได้ แต่สำหรับค่าพารามิเตอร์ที่ไม่ได้กำหนดไว้ จะใช้ค่าสมมติดังต่อไปนี้

- ความสูงของเสาอากาศที่สถานีฐานวัดจากพื้นดิน 40 m
- ความสูงของเสาอากาศที่โทรศัพท์เคลื่อนที่ 3 m
- พื้นที่การให้บริการอยู่ในบริเวณชุมชนเมือง โดยมีขนาดของพื้นที่เป็น 4 ตารางกิโลเมตร ต่อหนึ่งเซลล์
- เพื่อความสะดวก งานวิจัยนี้ ใช้กำลังส่งของผู้ใช้ทุกๆ คนที่ออกจากโทรศัพท์เคลื่อนที่ก่อนเดินทางผ่านช่องสัญญาณ มีค่าคงที่เป็น 1 W เนื่องจากเป็นกำลังที่สถานีฐานต้องการได้รับจากโทรศัพท์เคลื่อนที่แต่ละเครื่องที่อยู่ภายในเซลล์ แต่กำลังนี้ถูกเปลี่ยนไปเนื่องจากช่องสัญญาณแบบเฟดดิ้ง และ AWGN
- แบบจำลองการใช้ตัวควบคุม FGa เป็นการควบคุมกำลังของโทรศัพท์เคลื่อนที่ 1 เครื่อง ในระยะเวลาของการควบคุม 1 รอบการทำงาน เนื่องจากเป็นกรณีที่กำลังที่สถานีฐานได้รับจากโทรศัพท์ที่กำลังพิจารณา เกิดสัญญาณแทรกสอดกันที่สถานีฐานมากที่สุด

การสุ่มหาตำแหน่งของผู้ใช้แต่ละคน จะได้มาจากการสุ่ม ทำการบวกตำแหน่งที่สุ่มได้ ด้วยตำแหน่งของสถานีฐานของแต่ละเซลล์ เพื่อหาตำแหน่งของโทรศัพท์เคลื่อนที่ทั้งหมดใน 19 เซลล์ แล้วนำระยะทางไปคำนวณหาค่ากำลังที่สูญเสียไปเนื่องจากการเดินทาง ตามสมการที่ (3.15)

ระยะของผู้ใช้แต่ละคนในเซลล์วงรอบที่ 1 ที่ใกล้ที่สุดระหว่างผู้ใช้นับสถานีฐานของเซลล์ตรงกลาง (A) ไม่เกิน $d_1 = R\sqrt{3}$ และระยะห่างที่ไกลที่สุดคือ $d_2 = R\sqrt{7}$ ระยะของผู้ใช้แต่ละคนในเซลล์วงรอบที่ 2 ที่ใกล้ที่สุดระหว่างผู้ใช้นับสถานีฐาน A ไม่เกิน $d_3 = 2R\sqrt{3}$ และระยะห่างที่ไกลที่สุดคือ $d_4 = R\sqrt{19}$ ระยะของผู้ใช้แต่ละคนในเซลล์วงรอบที่ 3 ที่ใกล้ที่สุดไม่เกิน $d_5 = 3R\sqrt{3}$ และระยะห่างที่ไกลที่สุดคือ $d_6 = R\sqrt{37}$ การวัดระยะห่างแสดงดังรูปที่ 3.19 สมมติให้ $R = 1$ km เพื่อหาค่ากำลังที่ลดลงเนื่องจากระยะทางตามสมการที่ (3.15) ได้ดังนี้

$$S(d_1) = 112.0627 + [34.4065 \times \log_{10}(\sqrt{3})] = 120.2757 \quad \text{dBm}$$

$$S(d_2) = 112.0627 + [34.4065 \times \log_{10}(\sqrt{7})] = 126.6061 \quad \text{dBm}$$

$$S(d_3) = 112.0627 + [34.4065 \times \log_{10}(2\sqrt{3})] = 130.6331 \quad \text{dBm}$$

$$S(d_4) = 112.0627 + [34.4065 \times \log_{10}(\sqrt{19})] = 134.0664 \quad \text{dBm}$$

$$S(d_5) = 112.0627 + [34.4065 \times \log_{10}(3\sqrt{3})] = 136.6868 \quad \text{dBm}$$

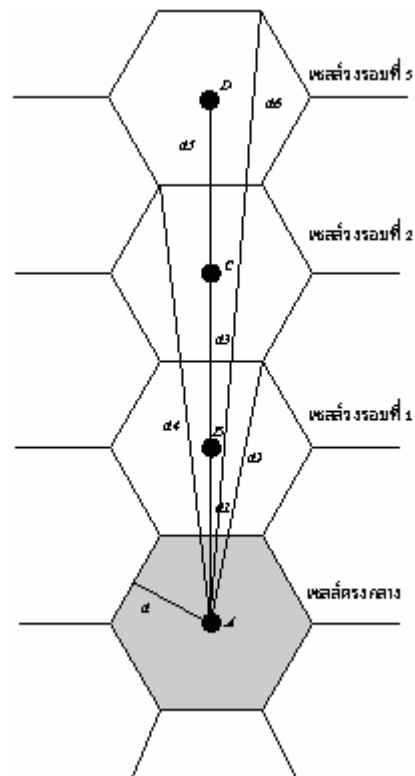
$$S(d_6) = 112.0627 + [34.4065 \times \log_{10}(\sqrt{37})] = 139.0409 \quad \text{dBm}$$

สมมติว่าผู้ใช้แต่ละคนส่งกำลังมายังสถานีฐานด้วยกำลัง 10 W รัศมีของเซลล์ (R) เท่ากับ 1 km ดังนั้นกำลังที่สถานีฐาน A ได้รับจากผู้ใช้ในเซลล์วงรอบที่ 1 จะอยู่ระหว่าง -80.2757 dBm ถึง -86.6061 dBm และกำลังที่สถานีฐาน A ได้รับจากผู้ใช้ในเซลล์วงรอบที่ 2 อยู่ระหว่าง -90.6331 dBm ถึง -94.0664 dBm กำลังที่สถานีฐาน A ได้รับจากเซลล์วงรอบที่ 1 มีค่าลดลงอยู่ในช่วงกว้าง ในขณะที่กำลังที่ A ได้รับจากเซลล์วงรอบที่ 2 มีค่าลดลงมาก และกำลังที่ได้รับยิ่งลดลงมากขึ้นเมื่อผู้ใช้อยู่ไกลจากเซลล์วงรอบที่ 2 ดังนั้นในงานวิจัยนี้ จึงใช้เซลล์รอบๆ เพียง 2 วงรอบเท่านั้น เนื่องจากเมื่อระยะไกลมากขึ้น กำลังที่สถานีฐานของเซลล์ตรงกลางได้รับมีค่าน้อยลง กำลังที่ลดลงเนื่องจากระยะทางแสดงดังรูป 3.19 (ข)

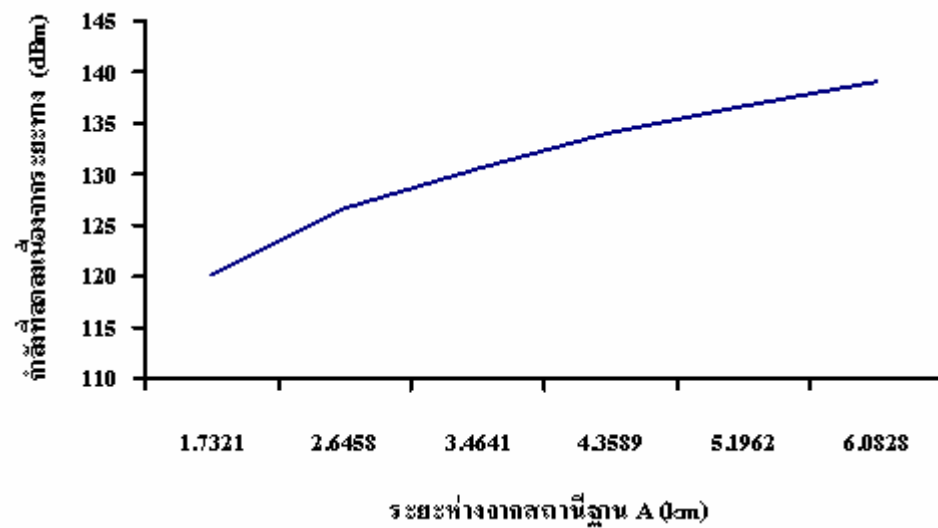
ผลจากตัวควบคุม FGa จะนำมาเปรียบเทียบผล กับตัวควบคุมที่ใช้วิธี FNN [21] ซึ่งใช้แบบจำลองของระบบที่มีเงื่อนไขดังนี้

- ความถี่คลื่นพาห้ 900 MHz
- แบนด์วิดท์ 1.25 MHz
- อัตราส่วนที่ยอมรับได้ระหว่างสัญญาณที่ต้องการ ต่อสัญญาณแทรกสอด $SIR_{Th} = -15$ dB
- แบบจำลองของช่องสัญญาณ มีการเฟดดิ้งของสัญญาณเป็นแบบเรย์ลีเฟดดิ้ง
- พื้นที่ที่พิจารณาเป็นโครงสร้างรูปหกเหลี่ยม ทั้งหมด 19 เซลล์ โดยพิจารณาควบคุมกำลัง

อุปกรณ์โทรศัพท์เคลื่อนที่ ที่อยู่ในเซลล์ตรงกลาง



(ก) ระยะที่ใกล้ และไกลที่สุดของเซลล์แต่ละวงรอบเมื่อวัดจากเซลล์กลาง



(ข) กำลังที่ลดลงเนื่องจากระยะทางของเซลล์ในวงรอบที่ 1, 2 และ 3

รูปที่ 3.19 กำลังที่ลดลงเนื่องจากระยะทางจากแบบจำลองของ Hata