

## บทที่ 4

### การสร้างแบบจำลองและศึกษาความเป็นไปได้

ในบทนี้ จะอธิบายถึงการสร้างแบบจำลองและศึกษาความเป็นไปได้ของงานวิจัย ซึ่งเนื้อหาจะประกอบไปด้วยการกำหนดนิยาม การกำหนดค่าดัชนีและตัวแปรในแบบจำลอง การกำหนดสมการเป้าหมาย และสมการบังคับ และการกำหนดรหัสจำลอง

#### 4.1 การกำหนดนิยาม

การกำหนดนิยามของปัญหาในแบบจำลองนี้จะได้กำหนดโดย ขยะพลาสติกทั้งหมด จะถูกจัดเก็บมายังโรงงานเก็บรวบรวมและคัดแยก จากนั้นขยะพลาสติกตามเปอร์เซ็นต์เรียก กลับคืนตามที่กำหนดจะถูกขนส่งไปทำการรีไซเคิลที่โรงงานรีไซเคิล ส่วนขยะที่เหลือทั้งหมดจะถูก นำไปกำจัดโดยการฝังกลบ โดยที่

ขยะทั้งหมดจะถูกส่งไปยังโรงงานเก็บรวบรวมและคัดแยก ซึ่งจำนวนโรงงานที่เปิดทำการ จะเปิดดำเนินการตามปริมาณของขยะที่สามารถรองรับได้ในเวลาที่ดำเนินการ

เมื่อขยะถูกคัดแยกแล้ว ขยะพลาสติกที่เรียกคืนตามเปอร์เซ็นต์ที่กำหนดจะถูกขนส่ง ลำเลียงไปยังโรงงานรีไซเคิล ซึ่งจำนวนโรงงานรีไซเคิลจะเปิดดำเนินการตามปริมาณของขยะที่ สามารถรองรับได้ในเวลาที่ดำเนินการ

ขยะพลาสติกที่ไม่ได้ถูกนำไปรีไซเคิล จะถูกนำไปกำจัดด้วยการฝังกลบที่บ่อฝังกลบ ซึ่งจำนวนบ่อฝังกลบที่เปิดทำการ จะเปิดดำเนินการตามปริมาณของขยะที่สามารถรองรับได้ใน ปริมาณความจุตามที่กำหนดไว้ในช่วงเวลา ซึ่งเมื่อรองรับขยะจนกระทั่งเต็มความจุแล้ว บ่อฝังกลบ จะปิดตัวลง และบ่อฝังกลบแห่งใหม่จะเปิดดำเนินการ ในขณะเดียวกันกรณีที่มีปริมาณขยะที่มาก ในช่วงเวลาเดียวกันบ่อฝังกลบสามารถเปิดดำเนินการมากกว่า 1 แห่ง ได้

ทั้งนี้ปริมาณของยานพาหนะและค่าใช้จ่ายที่ใช้ในการขนส่งจะเพิ่มขึ้นตามปริมาณ ของขยะและระยะทางที่ต้องถูกลำเลียงไปยังบ่อฝังกลบ และโรงงานต่างๆ

## 4.2 การกำหนดค่าดัชนี และตัวแปรในแบบจำลอง

ในขั้นตอนนี้จะได้อธิบายถึงการกำหนดค่าดัชนีที่จำเป็น ค่าพารามิเตอร์ ตลอดจนตัวแปรต่างๆ ที่นำมาประกอบการสร้างแบบจำลอง ดังแสดงในตารางที่ 1, 2 และ 3 ตามลำดับ

ตารางที่ 4.1

ค่าดัชนีในแบบจำลอง MILP

ดัชนี	คำอธิบาย	ดัชนี	คำอธิบาย
$S$	จังหวัด	$i, j$	ตำแหน่งของที่ตั้ง
$i_p^l$	ที่ตั้ง(บ่อฝังกลบ)	$i_p$	ที่ตั้งที่กำลังดำเนินการ
$C$	ที่ตั้งลูกค้า	$m$	รูปแบบในการขนส่ง
$k, p$	ชนิดของวัสดุขุด	$P/p$	ประเภทหลัก/รอง ของกระบวนการ

ตารางที่ 4.2

ตัวแปรที่ใช้ในการตัดสินใจในแบบจำลอง MILP

ตัวแปร	นิยาม และคำอธิบาย
$y_{it}^{(oper)}$	= 1 ถ้าที่ตั้ง $i$ เปิดและดำเนินการ ; 0 ถ้าเป็นอย่างอื่น
$y_{it}^{(open)}$ , $y_{it}^{(close)}$ , $\bar{y}_{it}^{(close)}$	= 1 ถ้า ที่ตั้ง $i$ เพิ่งเปิดทำการซึ่งเริ่มที่ช่วงเวลา $t$ ; 0 ถ้าเป็นอย่างอื่น = 1 ถ้า ที่ตั้ง $i$ เพิ่งปิดทำการซึ่งหยุดที่ช่วงเวลา $t-1$ , ตัวแบบผืนแปร
$y_{ijmt}^{(tr)}$	จำนวนยานพาหนะที่ต้องการเพื่อส่งถ่ายวัสดุขุดจากที่ตั้ง $i$ ไปยัง $j$ โดยรูปแบบการขนส่ง $m$ ณ ช่วงเวลา $t$
$y_{iPt}^{(pr)}$	จำนวนเครื่องจักรของกระบวนการหลัก $P$ ที่ต้องการ ณ ที่ตั้ง $i$ ที่ช่วงเวลา $t$
$x_{ikj}^{(tr)}$	จำนวนของวัสดุขุด $k$ ที่ส่งถ่ายไปยังที่ตั้ง $i$ ที่ช่วงเวลา $t$
$x_{ipt}^{(pr)}$ , $x_{ipPt}^{(tr)}$	จำนวนของวัสดุขุดในกระบวนการโดยกระบวนการย่อย $p$ ณ ที่ตั้ง $i$ ที่ช่วงเวลา $t$ , ปริมาณของวัสดุขุดในกระบวนการโดยกระบวนการหลัก $P$ , กระบวนการย่อย $p$ ณ ที่ตั้ง $i$ ที่ช่วงเวลา $t$
$ICAP_{it}$	= ปริมาณวัสดุขุดทั้งหมดในกระบวนการ ณ ที่ตั้ง $i$ ที่ดำเนินการได้ที่ช่วงเวลา $t$

ตารางที่ 4.3  
ตัวแปรต่างๆในแบบจำลอง MILP

ตัวแปร	นิยาม และคำอธิบาย
$F_{it}^{(oper)}$	กำหนดต้นทุนในการดำเนินการที่ $i$ ที่ช่วงเวลา $t$
$F_{it}^{(oper)}$ , $F_{it}^{(close)}$	กำหนดต้นทุนในการเปิดทำการและปิดทำการ ณ ที่ตั้ง $i$ ที่ช่วงเวลา $t$
$F_{ijmt}^{(tr)}$ , $F_{icmt}^{(tr)}$	กำหนดต้นทุนต่อยานพาหนะในการขนส่งวัตถุดิบจากที่ตั้ง $i$ ไปยัง $j$ /ลูกค้า $c$ ด้วยรูปแบบการขนส่ง $m$ ที่ช่วงเวลา $t$
$F_{iPt}^{(pr)}$	กำหนดต้นทุนหนึ่งหน่วยของกระบวนการหลัก $P$ ณ ที่ตั้ง $i$ ที่ช่วงเวลา $t$
$V_{ipt}^{(pr)}$	ต้นทุนการดำเนินการต่อหน่วยมาตรฐานสำหรับการดำเนินการย่อย $p$ ณ ที่ตั้ง $i$ ที่ช่วงเวลา $t$
$V_{ijmt}^{(tr)}$ , $V_{icmt}^{(tr)}$	ต้นทุนการขนส่งต่อหน่วยมาตรฐานของวัตถุดิบที่ส่งผ่านไปยังที่ตั้ง $i$ ไปยัง $j$ /ลูกค้า $c$ ที่ช่วงเวลา $t$
$\rho_{kp}$ , $\rho'_{kp}$	สัดส่วนของวัตถุดิบชนิด $k$ ที่ใช้โดยกระบวนการ $p$ สัดส่วนของวัตถุดิบชนิด $k$ ที่สร้างโดยกระบวนการ $p$
$C_{ikj}^{tr}$	ปริมาณสูงสุดของวัตถุดิบ $k$ ที่ยานพาหนะหนึ่งคันสามารถขนส่งได้ในช่วงเวลาจากที่ตั้ง $i$ ไปยัง $j$ ที่ช่วงเวลา $t$
$C_{pP}^{(pr)}$	ปริมาณสูงสุดของวัตถุดิบที่หนึ่งเครื่องจักรของกระบวนการ $P$ กระบวนการย่อย $p$ สามารถดำเนินการได้ในช่วงเวลา
$H_{(ijmt)}^{(tr)}$	จำนวนยานพาหนะสูงสุดในรูปแบบการขนส่ง $m$ ที่ส่งถ่ายวัตถุดิบจากที่ตั้ง $i$ ไปยัง $j$ ณ ช่วงเวลา $t$
$H_{iPt}^{(pr)}$	จำนวนเครื่องจักรสูงสุดของกระบวนการหลัก $P$ ณ ที่ตั้ง $i$ ที่ช่วงเวลา $t$
$G_{ijmt}^{(tr)}$	= 1 ถ้ายานพาหนะในการขนส่งรูปแบบ $m$ ที่ส่งถ่ายวัตถุดิบจากที่ตั้ง $i$ ไปยัง $j$ ต้องได้ประโยชน์ ณ เวลา $t$ ; 0 หากเป็นอย่างอื่น
$G_{iPt}^{(pr)}$	= 1 ถ้าเครื่องจักรของกระบวนการหลัก $P$ ณ ที่ตั้ง $i$ ต้องได้ประโยชน์ ณ เวลา $t$ ; 0 หากเป็นอย่างอื่น
$A_{ijmt}^{(tr)}$	= 1 ถ้ายานพาหนะในการขนส่งรูปแบบ $m$ ที่ส่งวัตถุดิบจากที่ตั้ง $i$ ไปยัง $j$ ยอมใช้ประโยชน์; 0 หากเป็นอย่างอื่น
$A_{iPt}^{(tr)}$	= 1 ถ้าเครื่องจักรของกระบวนการหลัก $P$ ณ ที่ตั้ง $i$ ยอมใช้ประโยชน์; 0 หากเป็นอย่างอื่น
$S_{skt}$ , $D_{kct}$	= จำนวนอุปทานของวัตถุดิบ $k$ ที่จังหวัด $s$ ณ ช่วงเวลา $t$ = จำนวนอุปสงค์ของวัตถุดิบ $k$ จากลูกค้า $c$ ณ ช่วงเวลา $t$
$CAP_i$	= ปริมาณวัตถุดิบสูงสุดทั้งหมดที่ดำเนินการ ณ ที่ตั้ง $i$ ที่สามารถดำเนินการได้ในระยะแรก

ตารางที่ 4.3 (ต่อ)  
ตัวแปรต่างๆ ในแบบจำลอง MILP

ตัวแปร	นิยาม และคำอธิบาย
$LCAP_i$	= ปริมาณความจุวัตถุดิบที่ต่ำที่สุดทั้งหมดที่บ่อฝังกลบ $i$ สามารถเก็บไว้ได้ก่อนที่จะปิดตัวลง
$M$	= ตัวเลขจำนวนมากที่เป็นบวก
$b_t$	= อัตราของเปอร์เซ็นต์ที่ต่ำที่สุดในการนำกลับมาของพลาสติก ณ ช่วงเวลา $t$

### 4.3 การกำหนดสมการเป้าหมาย และสมการบังคับ

การกำหนดสมการเป้าหมายและสมการบังคับ (Define Objective Function and Set Constrain) นั้น เมื่อพิจารณาจากตัวแปรต่างๆ ที่ประกาศไว้ซึ่งจะทำให้แบบจำลอง MILP สามารถที่จะตอบสนองความต้องการกับผลิตภัณฑ์ที่มีความหลากหลาย อีกทั้งกระบวนการต่างๆ ที่มีหลายขั้นตอน และช่วงเวลาที่มีความแตกต่างกัน เพื่อที่จะนำมาออกแบบเพื่อใช้ในการแก้ปัญหาจึงสามารถกำหนดสมการเป้าหมาย และสมการบังคับ ได้ดังนี้

#### สมการเป้าหมายต่ำสุด

$$\begin{aligned} & \sum_{i,t} F_{it}^{(oper)} y_{it}^{(oper)} + \sum_{i,t} F_{it}^{(open)} y_{it}^{(open)} + \sum_{i,t} F_{it}^{(close)} y_{it}^{(close)} + \sum_{i,j \neq i,m,t} F_{ijmt}^{(tr)} y_{ijmt}^{(tr)} + \sum_{i,j \neq i,k,m,t} V_{ijmt}^{(tr)} x_{ikjmt}^{(tr)} \\ & + \sum_{i,P,t} F_{iPt}^{(pr)} y_{iPt}^{(pr)} + \sum_{i,p,t} V_{ipt}^{(pr)} x_{ipt}^{(pr)} \end{aligned}$$

#### สมการบังคับ

$$(ก) \quad \sum_{i,j \neq i,k \in Plastics,m} x_{jkimt}^{(tr)} \geq b_t \sum_{i,k \in Plastics} S_{ikt} \quad \forall i \in i_p, t \quad (1)$$

$$\begin{aligned} (ข) \quad & \sum_{j \neq i,m} x_{jkimt}^{(tr)} - \sum_{j \neq i,m} x_{ikjmt}^{(tr)} - \sum_{c,m} x_{ikcmt}^{(tr)} \\ & + \sum_{p,P} \rho'_{kp} x_{ipP}^{(pr)} - \sum_{p,P} \rho_{kp} x_{ipPt}^{(pr)} = 0 \quad \forall i \in i_p, k, t \quad (2) \end{aligned}$$

$$(ค) \quad x_{ipt}^{(pr)} = \sum_{P:p \in P} x_{ikpPt}^{(pr)} \quad \forall i, p, t \quad (3)$$

$$(a) \sum_k \frac{x_{ikjmt}^{(tr)}}{C_{ikjmt}^{(tr)}} \leq y_{ijmt}^{(tr)} \quad \forall i, j \neq i, m, t \quad (4)$$

$$(a) \sum_{p \in P} \frac{x_{ipPt}^{(pr)}}{C_{pP}^{(pr)}} \leq y_{iPt}^{(pr)} \quad \forall i, P, t \quad (5)$$

$$(a) \sum_{jkmt} x_{jkimt}^{(tr)} \leq CAP_{it} \quad \forall i, t \quad (6)$$

$$(a) \sum_{ikjm} x_{ikjmt}^{(tr)} = S_{ikt} \quad \forall i \in i_s, k, t \quad (7)$$

$$(a) \sum_{i,m} x_{ijcmt}^{(Tr)} \leq D_{ckt} \quad \forall, k, c, t \quad (8)$$

$$(a) ICAP_{i0} = CAP_i \quad \forall i \in i_p^l \quad (9)$$

$$(a) ICAP_{it} = ICAP_{t-1} - \sum_{jkm} x_{jkim,t-1}^{(tr)} \quad \forall i \in i_p^l, t : t > 1 \quad (10)$$

$$y_{i,t+1}^{(close)} - 1 \leq M \bar{y}_{i,t+1}^{(close)} \quad \forall i \in i_p^l, t \quad (11)$$

$$(a) LCAP_i - ICAP_{it} < M(1 - \bar{y}_{i,t+1}^{(close)}) \quad \forall i \in i_p^l, t \quad (12)$$

$$(a) y_{i,t-1}^{(close)} \leq y_{it}^{(close)} \quad \forall i, t : t > 1 \quad (13)$$

$$(a) y_{it}^{(oper)} - y_{i,t-1}^{(oper)} \leq y_{it}^{(open)} \quad \forall i, t : t > 1 \quad (14)$$

$$(a) y_{i,t-1}^{(oper)} - y_{it}^{(oper)} \leq y_{it}^{(close)} \quad \forall i, t : t > 1 \quad (15)$$

$$(a) y_{it}^{(oper)} \leq y_{it}^{(open)} \quad \forall i, t = 1 \quad (16)$$

$$(a) G_{ijmt}^{(tr)} \leq y_{ijmt}^{(tr)} \leq H_{ijmt}^{tr} A_{ijmt}^{tr} y_{it}^{oper} \quad \forall i, j \neq i, m, t \quad (17)$$

$$(a) G_{ijmt}^{(tr)} \leq H_{ijmt}^{tr} A_{ijmt}^{tr} y_{it}^{oper} \quad \forall i, j \neq i, m, t \quad (18)$$

$$(a) G_{iPt}^{(pr)} \leq y_{iPt}^{(pr)} \leq H_{iPt}^{(pr)} A_{iPt}^{(pr)} y_{it}^{(oper)} \quad \forall i, P, t \quad (19)$$

$$(a) x_{ikjmt}^{(tr)}, x_{ipt}^{(pr)}, x_{ipPt}^{(pr)} \geq 0 \quad \forall i, k, j \neq i, t, p, P, m \quad (20)$$

$$(a) y_{it}^{(oper)}, y_{it}^{(open)}, y_{it}^{(close)} \in \{0,1\} \quad \forall i, t \quad (21)$$

$$(a) y_{ijmt}^{(tr)} \in \{0,1,2,\dots, H_{ijmt}^{(tr)}\} \quad \forall i, j \neq i, m, t \quad (22)$$

$$(๓) \quad y_{ip_t}^{(pr)} \in \{0,1,2,\dots, H_{ip_t}^{(pr)}\} \quad \forall i, P, t \quad (23)$$

จากสมการเป้าหมายข้างต้น ได้แสดงถึงการหาต้นทุนที่ต่ำที่สุดรวมไปถึงต้นทุนคงที่ และต้นทุนที่ใช้ในการดำเนินการที่เกิดขึ้นในโครงข่ายระหว่างการดำเนินการ ของโลจิสติกส์ ย้อนกลับในแบบจำลองนี้ โดยในส่วนของสมการบังคับนั้น ในสมการที่ (1) เป็นสมการที่กำหนด เพื่อให้แน่ใจว่าที่ตั้งที่เปิดดำเนินการอยู่ จะสามารถรองรับพลาสติกได้ไม่น้อยกว่าเปอร์เซ็นต์ที่กำหนดไว้ โดยในสมการที่ (2) และ (3) เป็นสมการบังคับให้เกิดความสมดุลของการไหลเข้าและไหลออกของวัตถุดิบในแต่ละที่ตั้งทุกช่วงเวลา ส่วนสมการที่ (4) และ (5) จะเป็นการกำหนดว่า ความสามารถของเครื่องจักร และยานพาหนะที่ใช้ในการขนส่งในแต่ละที่ตั้งจะไม่เกิน ความสามารถของที่ตั้งนั้น ส่วนในสมการที่ (7) และ (8) เป็นการสร้างความมั่นใจได้ว่าได้ทำการแพร่กระจายวัตถุดิบไปดำเนินการทั้งหมด ส่วนในสมการที่ (9)-(12) จะเป็นการบังคับให้บ่อฝังกลบที่เต็มแล้วปิดตัวลงและจะไม่เปิดทำการอีกในช่วงเวลาถัดไป ส่วนในสมการที่ (13)-(16) นั้น จะเป็นการควบคุมความสัมพันธ์เชิงตรรกะของตัวแปรทวิภาค ที่แสดงถึงที่กำลังปิดทำการและเปิดทำการ และในสมการที่ (17)-(23) นั้น จะเป็นการบังคับความสัมพันธ์เชิงตรรกะระหว่างตัวแปรต่อเนื่องและตัวแปรที่ไม่เป็นค่าลบ ในแบบจำลองนี้จะได้กล่าวถึงส่วนที่สำคัญสองเรื่อง คือ การแสดงถึงปัญหาของเปอร์เซ็นต์ในการนำกลับคืนตามที่ต้องการ และบ่อฝังกลบที่กำลังปิดตัวลง ซึ่งหากต้องการเพิ่มเปอร์เซ็นต์ในการเรียกกลับคืนพลาสติกเป็นจำนวนที่แน่นอน ในแต่ละช่วงเวลา โดยจะทำให้เกิดผลดังกล่าวได้ด้วยการบังคับใช้สมการ (1) บังคับ ในลำดับถัดมาสิ่งที่เราต้องพิจารณาอย่างรอบคอบ คือ ความจุของบ่อฝังกลบในแต่ละบ่อโดยเฉพาะส่วนที่ระบุถึงที่ตั้งที่กำลังดำเนินการอยู่ โดยพิจารณาบ่อฝังกลบที่มีความจุสูงสุดเป็นจำนวนเท่าใดจวบจนกระทั่งปิดตัวลง ซึ่งทางผู้วิจัยได้ทำการปรับแต่งแบบจำลองนี้เพื่อรองรับกรณีดังกล่าว โดยได้ทำการสมมติให้บ่อฝังกลบนี้สามารถที่จะรองรับวัตถุดิบได้ทุกชนิด ซึ่งสมการนี้ได้แสดงไว้ในสมการที่ (9)-(12)

#### 4.4 การกำหนดรหัสจำลอง

รหัสจำลอง (Pseudo code) หรือ รหัสเทียม เป็นเครื่องมือที่ช่วยในการแก้ปัญหาในรูปแบบอัลกอริทึม ซึ่งจะทำการเขียนเป็นคำอธิบายที่มีโครงสร้างที่ชัดเจน ทั้งนี้จะไม่ขึ้นกับภาษาใดภาษาหนึ่ง ซึ่งในการแสดงขั้นตอนการแก้ปัญหา แต่จะมีการใช้คำเฉพาะ (Reserve words) ที่มีอยู่ในภาษาโปรแกรมมาช่วยในการเขียน โครงสร้างของรหัสจำลองจึงมีส่วนที่คล้ายกับการเขียน

โปรแกรมคอมพิวเตอร์มาก ในที่นี้เราสามารถนำรูปแบบวิธีการเขียนมากำหนดค่าต่างๆ ลงใน GAMS เพื่อแก้ปัญหาตามที่ต้องการ โดยรหัสจำลองที่นำมาใช้ในการเขียนโปรแกรม GAMS มีดังนี้

#### 4.4.1 การกำหนดทางเลือก (Set Option)

ในเขียนรหัสจำลองเพื่อสร้างแบบจำลองด้วยโปรแกรม GAMS นี้ จำเป็นต้องกำหนดทางเลือกในการแก้ปัญหา ซึ่งเป็นค่าตั้งต้นของระบบในการจำลองการแก้ปัญหา โดยในที่นี้กำหนดให้

จำนวนแถวที่แสดงในแต่ละช่วงสมการ = 20 แถว

การหาทวนค่าภายในขอบเขตที่กำหนด = 1,000,000 ครั้ง

เปอร์เซ็นต์ยอมรับในการหาค่าตอบกำหนดการเชิงเส้นตรง = 10 เปอร์เซ็นต์

เวลาที่อนุญาตให้ทำการแก้ปัญหา = 24 ชั่วโมง หรือ 86,400 วินาที

นำมาเขียนรหัสจำลองใน GAMS ได้ดังนี้

```
option limrow =20;
```

```
option ITERLIM = 1000000;
```

```
option OPTCR = 0.10;
```

```
option reslim = 86400;
```

#### 4.4.2 การกำหนดแฟ้มข้อมูลเพื่ออ่านค่าที่ได้ (Set file to read)

ในการจำลองการแก้ปัญหาแต่ละครั้งนั้นโปรแกรม GAMS จะนำค่าที่ได้ไปทำการเก็บไว้โดยเขียนรหัสจำลองใน GAMS ได้ดังนี้

```
file thesismodel /C:\Thesismodel\thesismodel.txt/;
```

#### 4.4.3 การกำหนดวิธีการหาค่าที่ต้องการ (Set Method)

ข้อมูลที่น่าเข้ามา นั้น ใช้วิธีการกำหนดการเชิงเส้นมาใช้ในการหาค่าตอบการซึ่งหากใช้วิธี Simplex ธรรมดาอาจไม่สามารถหาผลลัพธ์ได้ จำเป็นที่จะต้องใช่วิธีที่ซับซ้อนขึ้น ในงานวิจัยนี้เลือกที่จะใช้วิธี Big-M Method ซึ่งการหาค่าต่ำสุดจะต้องทำให้ค่าสัมประสิทธิ์ของตัวแปรเทียม (M) และในสมการเป้าหมายสูงๆ ในปัญหาการหาค่าสูงสุดจะต้องทำให้ค่าสัมประสิทธิ์ของตัวแปรเทียม (-M) และในสมการเป้าหมายต่ำๆ โดยที่ M แทนด้วยตัวเลขมากๆ ดังนั้นจึงทำการแทนค่าในโปรแกรมให้ M = 999,999,999 โดยเขียนรหัสจำลองใน GAMS ได้ดังนี้

```
scalar bigM;
```

```
bigM = 999999999;
```

#### 4.4.4 การกำหนดตัวแปร (Set parameter)

ในการกำหนดตัวแปรซึ่งจะแทนสถานที่ตั้งของจังหวัดต่างๆ ทั้ง 76 จังหวัด ซึ่งเป็นแหล่งที่มาของขยะโดยแทนด้วยค่า i1- i76 ทั้งนี้กำหนดให้ไม่มีจังหวัดใดที่เป็นดัมมี่ โดยที่จะมีจังหวัดที่ถูกเลือกมาเป็นทำเลที่จะสร้างโรงงานเก็บรวบรวมและคัดแยก ซึ่งแทนด้วยค่า i77 - i94 จังหวัดที่ถูกเลือกมาเป็นทำเลที่จะสร้างโรงงานรีไซเคิลซึ่งแทนด้วยค่า i95 - i104 และจังหวัดที่ถูกเลือกมาเป็นทำเลที่จะสร้างบ่อฝังกลบซึ่งแทนด้วยค่า i105 - i110 และบ่อฝังกลบนี้จะมี 2 แบบ คือ บ่อแบบแรกเป็นบ่อขนาดซึ่งจะเปิดดำเนินการอยู่แล้ว และบ่อแบบที่สองจะเป็นบ่อที่เตรียมที่จะเปิดดำเนินการด้วยโปรแกรม ตลอดถึงตัวแปรที่เป็นการกำหนดลูกค้า ประเภทของวัตถุบิ วัตถุบิที่รวบรวมไปยังแหล่งรวบรวม ประเภทของการขนส่ง และช่วงเวลาในการแก้ปัญหา โดยเขียนรหัสจำลองใน GAMS ได้ดังนี้

```

set
$onempty
dum 'dummy site' / i0 /
s 'source counties' /i1*i76/
i 'potential nodes of sites' /i1*i110/
is(i) 'potential source sites' /i1*i76/
ic(i) 'potential collection sites' /i77*i94/
ip(i) 'potential processing sites' /i95*i104/
il(i) 'potential landfill sites' /i105*i110/
il1(i) 'landfill site type 1' /i105/
il2(i) 'landfill site type 2' /i106*i109/
il3(i) /i110/

c 'customers' /c1/
k 'material types' /k1/
kcol(k) 'materials collected at collection sites' /k1/
m 'transportation type' /1/
t 'time periods' /t1*t3/;
alias(i,j),(k,q);

```

#### 4.4.5 การอ่านข้อมูลจากภายนอก (Read Data)

การสร้างแบบจำลองครั้งนี้ได้นำข้อมูลระยะทางระหว่างที่ตั้ง และจำนวนประชากร ซึ่งได้บันทึกไว้ในโปรแกรมไมโครซอฟท์ ออฟฟิศ เอ็กเซล (MS Excel) ซึ่งเป็นโปรแกรมสำเร็จรูปง่ายต่อการจัดเก็บและแก้ไขข้อมูลเข้ามาทำการวิเคราะห์ในการแก้ปัญหาด้วย จึงต้อง เชื่อมต่อข้อมูลโดยในที่นี้กำหนดชื่อแฟ้มข้อมูล คือ “PlasticDataBig1” สามารถเขียนรหัสจำลองใน GAMS ได้ดังนี้

```
$onecho > allt.txt
i=%system.fp%PlasticDataBig1.xls
r1=pop!a2:pop!b77
o1=pop.inc
r2=popn!a2:popn!b77
o2=popn.inc
r3=rho1a!a1:rho1a!d5
o3=rho1.inc
r4=rho2a!a1:rho2a!d5
o4=rho2.inc
r5=dist!a1:dist!dg111
o5=dist.inc

$offecho

$call =xls2gms @allt.txt

table
d(i,j)
$include dist.inc
;
parameter dist(i,j);

loop(i,
    loop(j,
        dist(i,j) = d(i,j);
    );
);
```

#### 4.4.6 การกำหนดเปอร์เซ็นต์ในการรีไซเคิล (Define Recycle Percentage)

ในการสร้างแบบจำลอง เราสามารถกำหนดเปอร์เซ็นต์ในการรีไซเคิลเพื่อจำลองการแก้ปัญหาในค่าที่แตกต่างกันได้ เช่น หากกำหนดเปอร์เซ็นต์ในการรีไซเคิลในระยะเวลาทั้งสามปีไว้ “xxx” เปอร์เซ็นต์ จะสามารถเขียนรหัสจำลองใน GAMS ได้ดังนี้

```
parameter b(t);
b('t1') = xxx;
b('t2') = xxx;
b('t3') = xxx;
display b;
```

#### 4.4.7 การกำหนดจำนวนพลาสติก (Define Amount of Plastic)

เราสามารถกำหนดปริมาณพลาสติกเพื่อเป็นแหล่งข้อมูลในการแก้ปัญหาทั้งสามปีได้ โดยหากกำหนดให้มีปริมาณปีละ “xxx” ตัน จะสามารถเขียนรหัสจำลองใน GAMS ได้ดังนี้

```
parameter plastic(t);
plastic('t1') = xxx ;
plastic('t2') = xxx ;
plastic('t3') = xxx ;
```

#### 4.4.8 การแทนค่าในตัวแปรต่างๆ (Setup Parameter)

ลำดับต่อไปจะได้อธิบายถึงการแทนค่าลงในตัวแปรต่างๆ ซึ่งเป็นค่าที่มีส่วนสำคัญในการสร้างแบบจำลองเพื่อทำการแก้ปัญหา ซึ่งสมมติให้แทนค่า = “xxx” โดยแยกย่อยอธิบายได้ดังต่อไปนี้

- ต้นทุนในการเปิดโรงงาน และบ่อฝังกลบ ซึ่งสามารถนำมาเขียนรหัสจำลองใน GAMS ได้ดังนี้

```
parameter Fop(i,t) 'fixed cost to open sites =office/working space
leasing,forklift and driver,Fop(i,t)';
Fop(ic,t)= xxx;
Fop(ip,t)= xxx;
Fop(il,t)= xxx;
display Fop;
```

- ต้นทุนหลักในการดำเนินการของโรงงาน และบ่อฝังกลบ สามารถเขียนรหัสจำลองใน GAMS ได้ดังนี้

```
parameter Fjopen(i,t);
Fjopen(ic,t)= xxx;
Fjopen(ip,t)= xxx;
Fjopen(il,t)= xxx;
```

- ต้นทุนในการปิดโรงงาน และบ่อฝังกลบ สามารถนำมาเขียนเป็นรหัสจำลองใน GAMS ได้ดังนี้

```
parameter Fjclose(i,t);
Fjclose(ic,t)= xxx;
Fjclose(ip,t)= xxx;
Fjclose(il,t)= xxx;
```

- ต้นทุนในการจ้างแรงงาน สามารถเขียนรหัสจำลองใน GAMS ได้ดังนี้

```
parameter Fco(ic,t) 'fixed cost per worker = hire a warehouse
worker,Fco(icp,t)';
Fco(ic,t)= xxx;
display Fco;
```

- ต้นทุนในการขนส่งของพาหนะ 1 คัน สามารถเขียนรหัสจำลองใน GAMS ได้ดังนี้

```
parameter Ftrs(i,j,m,t) 'fixed cost per vehicle = buy a truck&trailer + hire
a driver,Ftrs(i,j,m,t)';
Ftrs(i,j,m,t)= xxx;
Ftrs(is,il,m,t)= xxx;
Ftrs(is,ic,m,t)= xxx;
Ftrs(ic,ip,m,t)= xxx;
Ftrs(ip,il,m,t)= xxx;
display Ftrs;
```

- ต้นทุนในการจ้างแรงงานซึ่งปฏิบัติงานอยู่ภายในกระบวนการต่างๆ สามารถเขียนรหัสจำลองใน GAMS ได้ดังนี้

parameter Fpr(i,t) 'fixed cost per main-process(Inspector/repair & disassembly worker),Fpr(i,Pm,t)';

Fpr(ic,t) = xxx;

Fpr(ip,t) = xxx;

Fpr(il,t) = xxx;

display Fpr;

- ต้นทุนในการขนส่งของขยะต่อหน่วย ซึ่งพาหนะ 1 คัน สามารถรองรับได้ ในการขนส่งขยะไปยังปลายทาง สามารถเขียนรหัสจำลองใน GAMS ได้ดังนี้

parameter Vtrs(i,j,m,t) 'transportation cost per ton of materials per vehicle of shipping mode m,Vtrs(i,j,m,t)';

Vtrs(is,j,m,t)= xxx\*dist(is,j);

Vtrs(ic,j,m,t)= xxx\*dist(ic,j);

Vtrs(ip,j,m,t)= xxx\*dist(ip,j);

Vtrs(i,is,m,t)= 0;

display Vtrs;

- ต้นทุนในการจัดการขยะต่อหน่วยในกระบวนการย่อยภายในโรงงาน หรือบ่อฝังกลบ สามารถเขียนรหัสจำลองใน GAMS ได้ดังนี้

parameter Vpr(i, t) 'processing cost per ton of materials in sub-process p,Vpr(i,p,t)';

Vpr(ic, t) = xxx;

Vpr(ip, t) = xxx;

Vpr(il, t) = xxx;

- การกำหนดนัยของจำนวนประชากรลงในแบบจำลองซึ่งกำหนดให้มีอัตราเพิ่มขึ้น 3 เปอร์เซ็นต์ในแต่ละปี สามารถเขียนรหัสจำลองใน GAMS ได้ดังนี้

parameter

Pop1(i) 'population in county s at the first time period,Pop1(s,t)'

\$include pop.inc

/;

scalar gamma 'percent of population growth per time period'/0.03/;

```

parameter Pop(i,t) 'population in county s at time period t';
loop(t,
    Pop(i,t)=power(1+gamma,ord(t)-1)*Pop1(i);
);
display Pop;
parameter Popn1(i) 'population at location n/'
$include pop.inc
/;
*display popn1;
parameter Popnt(i,t)'population at location n, at time period t';
loop(t,
    Popnt(i,t)= power(1+gamma,ord(t)-1)*Popn1(i);
);
scalar dummy;
parameter sumpop(t);
parameter NSupply(i,t) 'total supply of all products at time period t';

loop (t,
    sumpop(t) = 0;
    loop(i,
        sumpop(t) = sumpop(t) + Popnt(i,t);
    );
    dummy = sumpop(t);
    NSupply(i,t)= Popnt(i,t)*plastic(t)/dummy ;
);

```

- การกำหนดอุปทานของขยะ ณ ช่วงเวลาใด และทำเลที่ตั้งใด ลงในแบบจำลอง สามารถเขียนรหัสจำลองใน GAMS ได้ดังนี้

```

parameter Sup(i,k,t)'Supply of material k, at time t, at location n';
Sup(i,'k1',t)=NSupply(i,t);

```

- การกำหนดอุปสงค์ของขยะ ณ ช่วงเวลาใด และทำเลที่ตั้งใด (โดยหลักแล้วต้องกำหนดให้มีปริมาณมากกว่าอุปทานการไหลของขยะในระบบจึงจะไม่หยุดชะงัก)ลงในแบบจำลองสามารถเขียนรหัสจำลองใน GAMS ได้ดังนี้

parameter Dem(k,c,t) 'demand of material k at customer k at time period t (in lbs per period),Dem(k,c,t)';

Dem('k1',c,t)=xxx;

- ปริมาณขยะที่โรงงานเก็บรวบรวมและคัดแยกสามารถรองรับได้ ในช่วงเวลาที่กำหนด สามารถเขียนรหัสจำลองใน GAMS ได้ดังนี้

parameter Lc(i,t) 'amount of material (tons) that a collection site can receive,Lc(ic,t)';

Lc(ic,t)=xxx;

- ปริมาณขยะที่โรงงานรีไซเคิลสามารถรองรับได้ ในช่วงเวลาที่กำหนด สามารถเขียนรหัสจำลองใน GAMS ได้ดังนี้

parameter Lp(i,t) 'amount of material (tons) that a processing site can receive,Lp(ip,t)';

Lp(ip,t)=xxx;

- ปริมาณขยะที่คนงานสามารถรองรับได้ ในช่วงเวลาที่กำหนด สามารถเขียนรหัสจำลองใน GAMS ได้ดังนี้

parameter Cco(i,k) 'amount of material k (tons) that a worker can handle per period (tons per period),Cco(ic,k)';

Cco(i,'k1')=xxx;

- ปริมาณขยะที่พาหนะสามารถบรรทุกได้ ในช่วงเวลาที่กำหนด สามารถเขียนรหัสจำลองใน GAMS ได้ดังนี้

parameter Ctrs(i,k,j,m) 'amount of material k (tons) that a vehicle can ship per time period (tons per truck per period),Ctrs(i,k,j,m)';

ctrs(i,k,j,m) = xxx;

- ปริมาณขยะที่บ่อฝังกลบแต่ละแบบสามารถรองรับได้ สามารถเขียนรหัสจำลองใน GAMS ได้ดังนี้

```
parameter LCAP(i) 'Landfill Cap at Site i';
LCAP(il1) = xxx;
LCAP(il2) = xxx;
```

- ปริมาณขยะที่บ่อฝังกลบมีขยะค้างอยู่ในบ่อ สามารถนำมาเขียนเป็นรหัสจำลองใน GAMS ได้ดังนี้

```
parameter CAP(i) 'Current Cap at Site i';
CAP(il1) = xxx;
CAP(il2) = xxx;
CAP(il3) = xxx;
```

#### 4.4.9 การประกาศตัวแปร (Declare Variable)

ในขั้นตอนการประกาศตัวแปรนี้ จะนำรหัสจำลองที่ได้กำหนดค่ามาแล้วข้างต้นมาทำการกำหนดว่าเป็นตัวแปรชนิดใดเพื่อนำไปคำนวณหาผลเฉลยในลำดับต่อไป ซึ่งในกระบวนการนี้จะประกาศตัวแปรที่เป็นจำนวนเต็มบวก ตัวแปรที่เป็นแบบทวิภาค (ไบนารี) ตัวแปรที่เป็นเลขจำนวนเต็ม และตัวแปรอิสระ โดยสามารถเขียนรหัสจำลองใน GAMS ได้ดังนี้

```
variable
TCost;
positive variable
xtrs(i,k,j,m,t) 'amount of material shipped between sites(i to j or j to i),
xtrs(i,k,j,m,t)'
xtrc(i,k,c,m,t) 'amount of material shipped to customers(i to c),
xtrc(i,k,c,m,t)'
xprm(i,t) 'the amount of material processed by main process Pm, sub-
process p, xprm(i,p,Pm,t)'
xprs(i,t) 'the amount of material in sub-process p, xprs(i,p,t)'
ICAP(i,t) 'inventory capacity'

binary variables
yop(i,t) 'open/close a node, yop(i,t) = yoper'
```

```

yjopen(i,t) 'just opened site i at the beginning of time period t,yjopen(i,t)'
yjclose(i,t) 'just closed site i at the end of time period t-1,yjclose(i,t)'
yjclose_bar(i,t) 'just closed site i at the end of time period t-1,yjclose(i,t)'

```

```
integer variables
```

```

ytrs(i,j,m,t) " #vehicles between sites, ytrs(i,j,m,t)"
ypr(i,t) " #machines, ypr(i,Pm,t)"
yco(i,t) " #workers at collection sites, yco(icp,t)"
ytrc(i,c,m,t) " #vehicles to customers, ytrc(i,c,m,t)"

```

```
;
```

```
free variables
```

```
TotalSiteOpeningCost
```

```
;
```

#### 4.4.10 การกำหนดสมการ (Set Equation)

ในขั้นตอนการกำหนดสมการนี้จะกำหนดอักษรย่อ หรืออักษรแทนในการกำหนดตัวแปรต่างๆ เพื่อความสะดวกและความสมบูรณ์ในการนำไปเขียนลงในสมการเป้าหมายและสมการบังคับในลำดับต่อไป โดยสามารถเขียนรหัสจำลองใน GAMS ได้ดังนี้

```
equation
```

```
obj 'objective function
```

```
minrecycle 'minimum recycle each year'
```

```
balancecollect 'flow balance for permanent collection sites'
```

```
balanceprocess 'flow balance for processing sites'
```

```
balancelandfill 'flow balance for landfill sites'
```

```
balance_amt_process 'calculate amt process = amt sent to customer'
```

```
balance_amt_collect
```

```
balance_amt_landfill
```

```
close1
```

```
close2
```

```
close3
```

```
close4
```

close5

close6

close7

supply 'supply constraint at physical site location'

demand 'demand constraint'

justopen 'just opened a site at the beginning of current time period'

justclose 'just closed a site at the end of previous time period'

firstopen 'open a site at the first time period'

collectCap 'collection Cap

processCap 'processing Cap

allowcol 'allow to collect

allowpro 'allow to process

allowlf 'allow to open landfill

allow\_ytrs

allow\_ypr

allow\_ypr1

allow\_ypr2

;

#### 4.4.11 การกำหนดสมการเป้าหมาย (Objective Function)

ในขั้นตอนการกำหนดสมการเป้าหมายนี้เราจะนำเอารหัสจำลอง และตัวแปรที่ประกาศไว้แล้วข้างต้นมาแทนค่าลงในสมการเป้าหมายต่ำสุดในหัวข้อที่ 4.3 คือ สมการ

$$\sum_{i,t} F_{it}^{(oper)} y_{it}^{(oper)} + \sum_{i,t} F_{it}^{(open)} y_{it}^{(open)} + \sum_{i,t} F_{it}^{(close)} y_{it}^{(close)} + \sum_{i,j \neq i,m,t} F_{ijmt}^{(tr)} y_{ijmt}^{(tr)} + \sum_{i,j \neq i,k,m,t} V_{ijmt}^{(tr)} x_{ikjmt}^{(tr)} + \sum_{i,P,t} F_{iPt}^{(pr)} y_{iPt}^{(pr)} + \sum_{i,p,t} V_{ipt}^{(pr)} x_{ipt}^{(pr)}$$

ในรูปแบบของรหัสจำลองในโปรแกรม GAMS ได้ดังนี้

$$\begin{aligned} \text{obj.. TCost} = & \text{sum}((ic,t)\$Fop(ic,t),Fop(ic,t)*yop(ic,t)) \\ & + \text{sum}((ic,t)\$Fjopen(ic,t),Fjopen(ic,t)*yjopen(ic,t)) \\ & + \text{sum}((ic,t)\$Fjclose(ic,t),Fjclose(ic,t)*yjclose(ic,t)) \end{aligned}$$

$$\begin{aligned}
& +\text{sum}((ip,t)\$Fop(ip,t),Fop(ip,t)*yop(ip,t)) \\
& +\text{sum}((ip,t)\$Fjopen(ip,t),Fjopen(ip,t)*yjopen(ip,t)) \\
& +\text{sum}((ip,t)\$Fjclose(ip,t),Fjclose(ip,t)*yjclose(ip,t)) \\
\\
& +\text{sum}((il,t)\$Fop(il,t),Fop(il,t)*yop(il,t)) \\
& +\text{sum}((il,t)\$Fjopen(il,t),Fjopen(il,t)*yjopen(il,t)) \\
& +\text{sum}((il,t)\$Fjclose(il,t),Fjclose(il,t)*yjclose(il,t)) \\
\\
& +\text{sum}((i,j,m,t),Ftrs(i,j,m,t)*ytrs(i,j,m,t)) \\
& +\text{sum}((i,k,j,m,t),Vtrs(i,j,m,t)*xtrs(i,k,j,m,t)) \\
\\
& +\text{sum}((ip,k,c,m,t),Vtrc(ip,c,m,t)*xtrc(ip,k,c,m,t)) \\
\\
& +\text{sum}((i,t)\$Fpr(i,t),Fpr(i,t)*ypr(i,t)) \\
& +\text{sum}((i,t)\$Vpr(i,t),Vpr(i,t)*xprs(i,t));
\end{aligned}$$

#### 4.4.12 การกำหนดสมการบังคับ (Set Constrain)

ในขั้นตอนการกำหนดสมการบังคับนี้สามารถแยกย่อยอธิบายได้ดังนี้

- การกำหนดเพื่อให้แน่ใจว่าที่ตั้งที่เปิดดำเนินการอยู่จะสามารถรองรับพลาสติกได้ไม่น้อยกว่าเปอร์เซ็นต์ที่กำหนดไว้ ซึ่งได้กำหนดไว้ในสมการ (1) ในหัวข้อที่ 4.3 สามารถเขียนรหัสจำลองใน GAMS ได้ดังนี้

$$\begin{aligned}
& \text{minrecycle}(t).. \\
& \text{sum}((ip,k,c,m),xtrc(ip,k,c,m,t)) =g= b(t)* \text{plastic}(t);
\end{aligned}$$

- การบังคับให้เกิดความสมดุลของการไหลเข้าและไหลออกของวัตถุดิบในแต่ละที่ตั้งทุกช่วงเวลา ซึ่งได้กำหนดไว้ในสมการ (2) และ (3) ในหัวข้อที่ 4.3 สามารถเขียนรหัสจำลองใน GAMS ได้ดังนี้

$$\begin{aligned}
& \text{balancecollect}(i(ic),k,t).. \\
& \text{sum}((j(is),m),xtrs(j,k,i,m,t)) =e= \\
& \text{sum}((j(ip),m),xtrs(i,k,j,m,t)) + \text{sum}((j(il),m),xtrs(i,k,j,m,t));
\end{aligned}$$

$$\begin{aligned}
& \text{balanceprocess}(i(ip),k,t).. \\
& \text{sum}((j(ic),m),xtrs(j,k,i,m,t)) =e= \\
& \text{sum}((c,m),xtrc(i,k,c,m,t)) + \text{sum}((j(il),m),xtrs(i,k,j,m,t));
\end{aligned}$$

$$\begin{aligned} & \text{balance\_landfill}(i(il),k,t).. \\ & \text{sum}((j(is),m),xtrs(j,k,i,m,t)) + \text{sum}((j(ic),m),xtrs(j,k,i,m,t)) + \\ & \text{sum}((j(ip),m),xtrs(j,k,i,m,t)) =e= \\ & \text{sum}((c,m),xtrc(i,k,c,m,t)); \end{aligned}$$

$$\begin{aligned} & \text{balance\_amt\_process}(i(ip),t).. \\ & \text{sum}((c,k,m),xtrc(i,k,c,m,t)) =e= \text{xprs}(i,t); \end{aligned}$$

$$\begin{aligned} & \text{balance\_amt\_collect}(i(ic),t).. \\ & \text{sum}((j(is),k,m),xtrs(j,k,i,m,t)) =e= \text{xprs}(i,t); \end{aligned}$$

$$\begin{aligned} & \text{balance\_amt\_landfill}(i(il),t).. \\ & \text{sum}((c,k,m),xtrc(i,k,c,m,t)) =e= \text{xprs}(i,t); \end{aligned}$$

- การบังคับให้ความสามารถของเครื่องจักรและยานพาหนะที่ใช้ในการขนส่งในแต่ละที่ตั้งจะไม่เกินความสามารถที่ตั้งนั้น ซึ่งได้กำหนดไว้ในสมการ (4) และ (5) ในหัวข้อที่ 4.3 สามารถเขียนรหัสจำลองใน GAMS ได้ดังนี้

$$\begin{aligned} & \text{supply}(i(is),k,t).. \text{sum}((j(ic),m),xtrs(i,k,j,m,t)) =e= \text{Sup}(i,k,t); \\ & \text{demand}(k,c,t).. \text{sum}((i(ip),m),xtrc(i,k,c,m,t)) + \text{sum}((i(il),m),xtrc(i,k,c,m,t)) = \\ & \text{Dem}(k,c,t); \end{aligned}$$

- การบังคับให้วัตถุดิบแพร่กระจายเข้าสู่กระบวนการดำเนินการทั้งหมด ซึ่งได้กำหนดไว้ในสมการ (7) และ (8) ในหัวข้อที่ 4.3 สามารถเขียนรหัสจำลองใน GAMS ได้ดังนี้

$$\begin{aligned} & \text{collectCap}(i(ic),t).. \text{sum}((j(is),k,m),xtrs(j,k,i,m,t)) =l= \text{Lc}(i,t); \\ & \text{processCap}(i(ip),t).. \text{sum}((j(ic),k,m),xtrs(j,k,i,m,t)) =l= \text{Lp}(i,t); \end{aligned}$$

- การบังคับให้บ่อฝังกลบที่เต็มแล้วปิดตัวลง และจะไม่เปิดทำการซ้ำอีกในช่วงเวลาถัดไป ซึ่งได้กำหนดไว้ในสมการ (9) - (12) ในหัวข้อที่ 4.3 สามารถเขียนรหัสจำลองใน GAMS ได้ดังนี้

$$\begin{aligned} & \text{close1}(i(il),t)\$(\text{ord}(t)= 1).. \text{ICAP}(i,t) =e= \text{CAP}(i); \\ & \text{close2}(i(il),t)\$(\text{ord}(t)>1).. \text{ICAP}(i,t) =e= \text{ICAP}(i,t-1) + \\ & \text{sum}((j,k,m),xtrs(j,k,i,m,t)); \\ & \text{close3}(i(il),t)\$(\text{ord}(t)< 3).. \text{yclose}(i,t+1) - 1 =l= \text{bigM}*\text{yclose\_bar}(i,t+1); \end{aligned}$$

```

close4(i(il),t)$(ord(t)>1).. LCAP(i) - ICAP(i,t) =| bigM*(1-
yjclose_bar(i,t+1));
close5(i,t)$(ord(t)>1).. yjclose(i,t-1) =| yjclose(i,t);
close6(i,t)$(ord(t)>1).. yjopen(i,t+1) + yjclose(i,t) =| 1;
close7(i).. yjopen(i,'t1') + yjopen(i,'t2') + yjopen(i,'t3') =| 1;

justopen(i,t)$(ord(t)>1).. yop(i,t)-yop(i,t-1) =| yjopen(i,t);
justclose(i,t)$(ord(t)>1).. yop(i,t-1)-yop(i,t) =| yjclose(i,t);
firstopen(i,t).. yop(i,'t1') =| yjopen(i,'t1');

```

- การควบคุมความสัมพันธ์เชิงตรรกะของตัวแปรทวิภาคที่แสดงถึงที่ตั้งที่กำลังปิดทำการและเปิดทำการ ซึ่งได้กำหนดไว้ในสมการ (13) - (16) ในหัวข้อที่ 4.3 สามารถเขียนรหัสจำลองใน GAMS ได้ดังนี้

```

allowcol(ic,t).. sum((j(is),k,m),xtrs(j,k,ic,m,t)) =| bigM*yop(ic,t);
allowpro(ip,t).. sum((j(ic),k,m),xtrs(j,k,ip,m,t)) =| bigM*yop(ip,t);
allowlf(il,t).. sum((j(is),k,m),xtrs(j,k,il,m,t)) + sum((j(ic),k,m),xtrs(j,k,il,m,t))+
sum((j(ip),k,m),xtrs(j,k,il,m,t)) =| bigM*yop(il,t);

```

- การบังคับความสัมพันธ์เชิงตรรกะของตัวแปรต่อเนื่องและตัวแปรที่ไม่เป็นค่าลบ ซึ่งได้กำหนดไว้ในสมการ (17) - (23) ในหัวข้อที่ 4.3 สามารถเขียนรหัสจำลองใน GAMS ได้ดังนี้

```

allow_ytrs (i,k,j,m,t).. xtrs(i,k,j,m,t) =| bigM*ytrs(i,j,m,t);
allow_ypr(ip,t).. xprs(ip,t) =| bigM*ypr(ip,t);
allow_ypr1(ic,t).. xprs(ic,t) =| bigM*ypr(ic,t);
allow_ypr2(il,t).. xprs(il,t) =| bigM*ypr(il,t);
yjopen.lo(il1,'t1') = 1;
yop.lo(il1,'t1') = 1;

```

#### 4.4.13 การแก้ปัญหา (Solve)

ในขั้นตอนนี้จะแสดงถึงการแก้ปัญหาโดยใช้ต้นทุนต่ำที่สุด และการแสดงค่าผลเฉลยทางหน้าจอมอนิเตอร์ ซึ่งสามารถเขียนรหัสจำลองใน GAMS ได้ดังนี้

```
solve finalmodel using mip min TCost;
display xtrs.l;
display xtrc.l;
display xprs.l;
display yop.l;
display yjopen.l;
display yjclose.l;
display TCost.l;
```

#### 4.4.14 การคำนวณต้นทุนที่ใช้ (Cost Compute)

ในขั้นตอนนี้จะนำค่าที่ได้จากการกำหนดตัวแปร และผลเฉลยในสมการที่กล่าวมาแล้วมาทำการคำนวณต้นทุนในกระบวนการเก็บรวบรวมและคัดแยก กระบวนการรีไซเคิล และกระบวนการฝังกลบ พร้อมทั้งแสดงผลที่ได้ทางมอนิเตอร์ ซึ่งสามารถเขียนรหัสจำลองใน GAMS ได้ดังนี้

```
parameter CostCollectOpening;
CostCollectOpening =sum((ic,t),Fop(ic,t)*yop.l(ic,t))
                    + sum((ic,t),Fjopen(ic,t)*yjopen.l(ic,t))
                    + sum((ic,t),Fjclose(ic,t)*yjclose.l(ic,t));
parameter CostProcessOpening;CostProcessOpening
= sum((ip,t),Fop(ip,t)*yop.l(ip,t))
  + sum((ip,t),Fjopen(ip,t)*yjopen.l(ip,t))
  + sum((ip,t),Fjclose(ip,t)*yjclose.l(ip,t));

parameter CostLandFill;CostLandFill =sum((il,t),Fop(il,t)*yop.l(il,t))
  + sum((il,t),Fjopen(il,t)*yjopen.l(il,t))
  + sum((il,t),Fjclose(il,t)*yjclose.l(il,t));

parameter Transportation;
Transportation =sum((i,j,m,t),Ftrs(i,j,m,t)*ytrs.l(i,j,m,t))
```

$$+\text{sum}((i,k,j,m,t),V\text{trs}(i,j,m,t)*x\text{trs}.l(i,k,j,m,t));$$

\*parameter TransportationCust;

$$*\text{TransportationCust} = \text{sum}((i,k,c,m,t),V\text{trc}(i,c,m,t)*x\text{trc}.l(i,k,c,m,t));$$

parameter CostProcessC;

$$\text{CostProcessC} = \text{sum}((ic,t),F\text{pr}(ic,t)*y\text{pr}.l(ic,t)) \\ + \text{sum}((ic,t),V\text{pr}(ic,t)*x\text{prs}.l(ic,t));$$

parameter CostProcessP;

$$\text{CostProcessP} = \text{sum}((ip,t),F\text{pr}(ip,t)*y\text{pr}.l(ip,t)) \\ + \text{sum}((ip,t),V\text{pr}(ip,t)*x\text{prs}.l(ip,t));$$

parameter CostProcessL;

$$\text{CostProcessL} = \text{sum}((il,t),F\text{pr}(il,t)*y\text{pr}.l(il,t)) \\ + \text{sum}((il,t),V\text{pr}(il,t)*x\text{prs}.l(il,t));$$

display CostCollectOpening, CostProcessOpening,

CostLandFill,Transportation,CostProcessC, CostProcessP, CostProcessL;

การกำหนดรหัสจำลอง หรือ รหัสเทียม ที่กล่าวมาข้างต้นนั้น เป็นส่วนที่สำคัญอย่างยิ่งที่ผู้ดำเนินการวิจัยจะต้องทำการกำหนดรหัสอย่างระมัดระวัง และรอบคอบ เพราะหากเรากำหนดรหัสผิดพลาด นั่นก็หมายความว่าค่า หรือพารามิเตอร์ต่างๆ ที่เราแทนลงไปแบบจำลองนั้นมีความคลาดเคลื่อนไปจากความเป็นจริง อันจะส่งผลให้การแก้ปัญหาที่มีความผิดพลาดทำให้ไม่สามารถแก้ปัญหาที่แท้จริงได้ ซึ่งการกำหนดรหัสที่ดีจะต้องเขียนโดยมีคำอธิบายที่มีโครงสร้างที่ชัดเจน และแยกการวิเคราะห์ในแต่ละส่วนให้ชัดเจน ทั้งนี้ก็เพื่อง่ายต่อการตรวจสอบ และแก้ไข ดังนั้นโครงสร้างของรหัสจำลองจึงมีส่วนที่สำคัญอย่างมาก ที่จะส่งผลให้งานวิจัยซึ่งอาศัยการแก้ปัญหาด้วยรูปแบบทางคณิตศาสตร์โดยใช้คอมพิวเตอร์ในการประมวลผลนั้น มีความสมบูรณ์ และแม่นยำมากยิ่งขึ้น