

การวางแผนล่วงหน้าในการกำหนดตำแหน่งที่เหมาะสมให้กับ สินค้าในคลังสินค้า กรณีศึกษา ผู้ให้บริการโลจิสติกส์ครบวงจร

Planning in Advance for determine appropriate location for product in the warehouse

วิวัฒน์ เจียรวัชรมงคล และ ตรีวิทย์ เยาวสุวรรณไชย

มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี

E-mail : wittawat_jien@hotmail.com

บทคัดย่อ

งานวิจัยนี้เป็นการพัฒนาแบบจำลองคณิตศาสตร์เพื่อช่วยในการวางแผนการเลือกตำแหน่ง (Location) ในการจัดเก็บสินค้าที่เหมาะสมภายในคลังสินค้าของบริษัทกรณีศึกษา โดยมีวัตถุประสงค์เพื่อลดระยะทางรวมในการเคลื่อนย้ายสินค้าเข้าออกจากคลังสินค้าซึ่งในปัจจุบัน คลังสินค้าของบริษัทกรณีศึกษามีจุดอ่อนในการเลือก Location ให้กับสินค้าใน 2 ด้านคือ (1) ไม่ได้เลือก Location ที่ใกล้ประตูให้กับสินค้าที่มีอัตราการหมุนเวียนสูง และ (2) ไม่มีการวางแผนล่วงหน้าในการเลือก Location ให้กับสินค้า เป็นเพียงการเลือก Location แบบวันต่อวัน ซึ่งอาจทำให้สินค้าที่มีอัตราการหมุนเวียนสูงแต่เข้ามาคลังสินค้าในวันหลังๆ อาจถูกจัดวางใน Location ที่ไกลประตู เพราะมีจัดวางสินค้าที่มีอัตราการหมุนเวียนต่ำแต่เข้ามาคลังสินค้าในวันต้นๆ ใน Location ที่ใกล้ประตูไปแล้ว

ดังนั้น เพื่อแก้จุดอ่อนใน 2 ด้านนี้ ผู้วิจัยจึงได้พัฒนาแบบจำลองทางคณิตศาสตร์ที่สามารถช่วยจัดวางสินค้าตามอัตราการหมุนเวียน และเป็นการวางแผนล่วงหน้า โดยใช้วิธี Linear Programming และแบบจำลองทางคณิตศาสตร์จำนวน 2 แบบที่ผู้วิจัยได้พัฒนาขึ้นมาในการวางแผนการเลือก Location ให้กับสินค้าได้ โดยแบบจำลองที่ 1 เป็นปัญหาการจับคู่ 3 มิติ (Three Dimensional Assignment Problem) และแบบจำลองที่ 2 เป็นการดัดแปลงปัญหาการจับคู่ 3 มิติเป็นปัญหาการจับคู่ 2 มิติ นอกจากนี้ยังทดสอบและเปรียบเทียบประสิทธิภาพของแบบจำลองทั้ง 2 แบบ โดยจำลองปัญหาทดสอบ (Test Problem) จำนวน 35 ปัญหา และใช้โปรแกรม Lingo สำหรับการแก้ปัญหา Linear Programming นอกจากนี้ยังพัฒนาโปรแกรมด้วยภาษา Visual Basic for Application ในโปรแกรม Microsoft Excel เพื่อคำนวณหาค่าของตัวแปรนำเข้าโดยอัตโนมัติ ผลการทดสอบประสิทธิภาพพบว่า แบบจำลองที่ 2 สามารถหาคำตอบที่ดีที่สุดได้เร็วกว่าแบบจำลองที่ 1 แต่ว่าแบบจำลองที่ 2 มีอุปสรรคในการแก้ปัญหาขนาดใหญ่ เพราะแบบจำลองที่ 2 ต้องใช้หน่วยความจำของโปรแกรม Lingo และเครื่องคอมพิวเตอร์ปริมาณมากในการแก้ปัญหา

คำสำคัญ :

แบบจำลองคณิตศาสตร์ โปรแกรมเชิงเส้น ปัญหาการจับคู่ 3 มิติ

Abstract

This research is to develop a mathematical model to assist in planning and selecting

appropriate locations for a set of products in a warehouse of the case study company. The objective is to minimize total distance of product moved inside the warehouse. Two major weaknesses of the current practice at the warehouse are: (1) the products with a high turnover rate are not placed in close proximity to the door; and (2) there is no advance planning. That is, the location selecting process is implemented on a daily basis.

To solve the two weaknesses, we have developed two mathematical models that can help locate the placement according to the turnover rate and plan proactively plan ahead. By using the linear programming method, both mathematical models can reach the same solution. Model 1 is similar to the Three Dimensional Assignment problem, and Model2 is to transform the problem of matching from 3 dimensions to 2 dimensions. To test and compare the performance of the both models, we constructed 35 test problems and used Lingo program for solving linear programming. We also developed a program with Visual Basic Application in Microsoft Excel to automatically calculate the value of the input variables. Test results showed that Model 2 can find the optimal solutions faster than Model 1, but Model 2 has a difficulty in solving large-sized problems because it requires the larger amount of computer memory

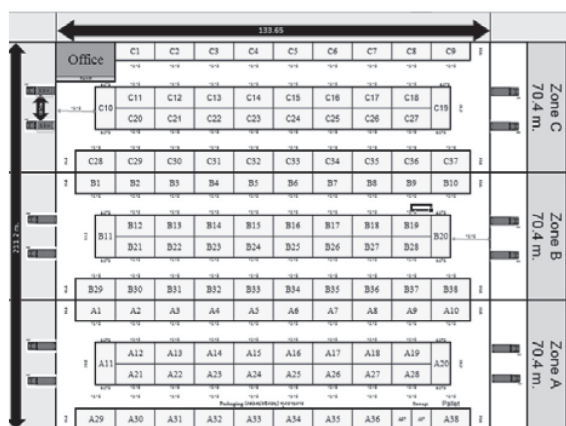
Keywords: >>>

mathematical model, linear programming, Three Dimensional Assignment problems

1. บทนำ

การจัดการคลังสินค้า[1]เป็นกิจกรรมด้านโลจิสติกส์ที่มีความสำคัญต่อธุรกิจ และมีความจำเป็นอย่างมากที่ต้องพัฒนาการจัดการคลังสินค้าให้มีประสิทธิภาพยิ่งขึ้น เพื่อตอบสนองความต้องการของลูกค้าที่เปลี่ยนแปลงอย่างรวดเร็วได้ การเลือก “ตำแหน่ง” หรือ “Location” ในการจัดเก็บสินค้า หากสามารถเลือกได้เหมาะสมก็จะช่วยเพิ่มประสิทธิภาพของคลังสินค้าได้ จากนั้นไปในบทความนี้ ผู้วิจัยขอใช้คำว่า Location แทนคำว่าตำแหน่ง ผู้วิจัยได้เข้าไปศึกษากระบวนการทำงานในบริษัทกรณีศึกษาแห่งหนึ่งซึ่งเป็นผู้ให้บริการด้านโลจิสติกส์ครบวงจรรายใหญ่ของประเทศไทย โดยทางบริษัทมีคลังสินค้าขนาดใหญ่จำนวนหลายแห่งทั่วประเทศ โดยผู้วิจัยได้เข้าไปศึกษาคลังสินค้าของบริษัทกรณีศึกษาในจังหวัดระยอง คลังสินค้าแห่งนี้ มีพื้นที่ในการจัดเก็บ 21,470 ตารางเมตร และมีการจัดเก็บสินค้าแบบวางซ้อนกันบนพื้น (Block Stacking) โดยที่คลังสินค้ามีโซนในการจัดเก็บสินค้าแบ่งออกเป็น 3 โซนคือ โซน A, B และ C มีจำนวน Location ทั้งสิ้น

113 Location และในแต่ละ Location มีพื้นที่เท่ากับ 190 ตารางเมตร มีประตูสำหรับให้รถบรรทุกจอดเพื่อขึ้นลงของอยู่ทั้ง 2 ด้านของคลัง แต่ละด้านมี 6 ประตูตามแผนผังในภาพที่ 1



ภาพที่ 1 คลังสินค้าของบริษัทกรณีศึกษา

จากการศึกษาพบว่า ทางคลังสินค้ามีจุดอ่อนในการเลือก Location ให้กับสินค้าที่ควรปรับปรุงใน 2 ด้าน ด้านแรกคือทางคลังสินค้าไม่ได้เลือก Location ให้



กับสินค้าตามแนวทาง Fastest Mover Closest to the Door [2] นั่นคือ ทางคลังสินค้าไม่ได้พยายามจัดวางสินค้าที่มีอัตราหมุนเวียนสูงไว้ใน Location ที่ใกล้ประตูทางเข้าออกมากที่สุด แต่พบว่าได้จัดวางสินค้าใน Location แบบสุ่มคือวางสินค้าใน Location ใดก็ได้ และไม่มีผลการพิจารณาปัจจัยใดในการจัดวางในด้านที่สองพบว่า ทางคลังสินค้าไม่มีการวางแผนล่วงหน้าในการเลือก Location ให้กับสินค้า ทั้งๆ ที่จากการศึกษาพบว่า ทางคลังสินค้าแห่งนี้มีข้อมูลของวันที่ที่สินค้าจะเข้าและจะออกจากคลังสินค้าล่วงหน้าอย่างน้อย 1 เดือน แต่ทางคลังสินค้าไม่ได้ใช้ประโยชน์จากข้อมูลดังกล่าวในการเลือก Location ในการจัดเก็บสินค้า ปัจจุบันทางคลังสินค้าจะกำหนด Location ให้กับสินค้าที่เข้าคลังสินค้าแบบวันต่อวัน ทำให้ใช้ระยะทางในการเคลื่อนย้ายสินค้าภายในคลังสินค้ามากเกินไปจนเกิดความจำเป็น ผู้วิจัยพบว่าข้อมูลดังกล่าวจะเป็นประโยชน์ในการวางแผนการเลือก Location ให้กับสินค้าได้ เนื่องจากการวางแผนการจัดเก็บสินค้าล่วงหน้าสามารถทำให้มีการสำรอง Location ที่ใกล้ประตูไว้สำหรับสินค้าที่มีอัตราการหมุนเวียนสูง แต่มีกำหนดเข้าคลังสินค้าในวันหลังๆ หรือวันถัดๆ ไป นอกจากนี้การวางแผนจัดเก็บสินค้าล่วงหน้าจะช่วยเพิ่มการใช้ประโยชน์ของพื้นที่จัดเก็บได้มากขึ้น

ดังนั้น เพื่อแก้ไขจุดอ่อนทั้ง 2 ด้าน ผู้วิจัยจึงพัฒนาวิธีการวางแผนล่วงหน้าเพื่อเพิ่มประสิทธิภาพในการบริหารจัดการคลังสินค้าให้สูงขึ้น โดยการวางแผนการจัดเก็บสินค้าล่วงหน้าจะทำให้คลังสินค้าสามารถสำรอง Location สำหรับสินค้าที่มีอัตราการหมุนเวียนของสินค้าสูง เพื่อให้ระยะทางรวมในการเคลื่อนย้ายสินค้าต่ำที่สุดด้วยแนวคิดการจัดกลุ่มสินค้า ABC (ABC Analysis) [3] และแนวทาง Fastest Mover Closest to the Door [2] โดยอาศัยเครื่องมือด้านโปรแกรมเชิงเส้น (Linear Programming) [4] ดังนั้นผู้วิจัยจึงได้พัฒนาแบบจำลองคณิตศาสตร์เพื่อสามารถใช้วางแผนล่วงหน้าในการเลือก Location ที่เหมาะสมตามอัตราหมุนเวียนของสินค้า ซึ่งผู้วิจัยได้พัฒนาออกมาเป็น 2 รูปแบบ โดยแบบจำลองรูปแบบแรกสามารถจัดอยู่ในกลุ่มปัญหา Three-Dimensional Assignment Problem [5] นั้น

คือเป็นปัญหาการจับคู่ 3 มิติ ระหว่าง Location สินค้า และวันที่ แบบจำลองรูปแบบที่ 2 ได้พัฒนาต่อยอดมาจากแบบจำลองคณิตศาสตร์ของซนิกานต์ กมลสุข [6] ซึ่งเป็นการแปลงปัญหาการจับคู่ในรูปแบบ 3 มิติ คือระหว่าง Location สินค้า และวันที่ ให้อยู่ในรูปแบบของการจับคู่ 2 มิติแทน คือเป็นการจับคู่ระหว่าง Location กับสินค้าเท่านั้น และใช้สมการที่มีตัวแปรนำเข้า $S_{j|t}$ เข้ามาจัดการกับมิติของวันที่แทนในส่วนท้ายของบทความนี้ ผู้วิจัยได้เปรียบเทียบประสิทธิภาพในการหาคำตอบของแบบจำลองทางคณิตศาสตร์ทั้ง 2 รูปแบบ โดยใช้โปรแกรม Lingo Version 14 [7] และเครื่องมือ Visual Basic for Applications (VBA) ในโปรแกรมไมโครซอฟท์เอ็กเซล (Microsoft Excel) ในการทดสอบ

2. ปัญหาจำลองอย่างง่าย

ในการอธิบายเพื่อช่วยให้เข้าใจลักษณะปัญหาและแบบจำลองทางคณิตศาสตร์ที่ผู้วิจัยได้พัฒนาขึ้นมานั้น ผู้วิจัยขอใช้ปัญหาจำลองอย่างง่ายอธิบายดังต่อไปนี้คลังสินค้าแห่งหนึ่งต้องการวางแผนการเลือก Location ให้กับสินค้าใน 7 วันล่วงหน้า โดยในวันที่ 0 มีสินค้าเกาแดงค้างอยู่ในคลังสินค้า 2 รายการ (คือสินค้า ก และ ข) และตั้งแต่วันที่ 1-7 จะมีสินค้าเข้ามาใหม่อีก 10 รายการ (ได้แก่สินค้า A, B, C, D, E, F, G, H, I และ J) โดยข้อมูลของวันที่ที่สินค้าแต่ละรายการจะเข้าและออกจากคลังสินค้าจะแสดงในตารางที่ 1 ในคลังสินค้ามี Location ให้วางสินค้าได้อยู่เพียง 5 Location (ได้แก่ Location 1, 2, 3, 4, และ 5) สมมติให้คลังสินค้ามีประตูเข้าออกประตูเดียวและระยะทางจากประตูไปยังแต่ละ Location ดังแสดงในตารางที่ 2 (มีหน่วยเป็นเมตร) สำหรับสินค้า ก และ ข นั้นคงค้างอยู่ใน Location 2 และ 4 ดังตารางที่ 3 และแต่ละสินค้ามีอัตราหมุนเวียนดังในตารางที่ 4 และมีเงื่อนไขเพิ่มเติมในการจัดวางสินค้าคือ

- ในแต่ละวัน สินค้าที่ออกจะออกก่อนสินค้าเข้าเสมอ
- ในแต่ละวัน ในแต่ละ Location สามารถเลือกวางได้เพียงสินค้าเดียว

- สินค้าเมื่อถูกจัดวางใน Location ไหนแล้วจะต้องอยู่ที่ Location นั้นจนถึงวันที่ออก ไม่สามารถย้ายไป Location อื่นได้

ตารางที่ 1 ข้อมูลการเข้า-ออกของสินค้าในแต่ละวันของปัญหาจำลองอย่างง่าย

วันที่ (w)	สินค้าที่ค้างใน Location	เข้า	ออก
1	ก, ข	A, B, C	
2		D	ข
3		E	ก, B
4		F, G	C
5		H, I	A, D
6		J	E, G
7			F, I
			H, J

ตารางที่ 2 ระยะทาง (เมตร) ในการเคลื่อนย้ายสินค้าจากประตูไปยังแต่ละ Location

Location(i)	(i) = 1	(i) = 2	(i) = 3	(i) = 4	(i) = 5
ระยะทาง	5	10	15	20	25
(เมตร)					

ตารางที่ 3 สินค้าที่ค้างอยู่ใน Location ในวันที่ 0

Location (i)				
(i) = 1	(i) = 2	(i) = 3	(i) = 4	(i) = 5
	ก		ข	

ตารางที่ 4 อัตราการหมุนเวียนของแต่ละสินค้า

สินค้า	A	B	C	D	E	F	G	H	I	J
อัตราหมุนเวียน	5	6	7	8	9	10	11	12	13	14

หากลองแก้ปัญหาจำลองอย่างง่ายนี้ตามแนวทาง Fastest Mover Closest to the Door แต่ไม่ได้วางแผนล่วงหน้า 7 วัน แต่เป็นการพิจารณาแบบวันต่อวัน (คือไม่ได้ใช้ข้อมูลในตารางที่ 1 มาประกอบการจัดวาง) นั่นคือในแต่ละวัน เลือกสินค้าที่เข้ามาในวันนั้นมีอัตราหมุนเวียนสินค้ามากที่สุด ให้มาสินค้านั้นไปวาง

ใน Location ที่มีระยะทางต่ำที่สุดก่อน ยกตัวอย่างเช่น ในวันที่ 1 มีสินค้าเข้าคือ A, B, และ C และมีอัตราหมุนเวียนคือ 5, 6, และ 8 ตามลำดับ และในวันที่ 1 มี Location วางอยู่ 3 Location คือ Location 1, 3, และ 5 ดังนั้นอาศัยแนวทาง Fastest Mover Closest to the Door ในวันแรกสามารถจัดวางได้คือ สินค้า C วางใน Location 1 สินค้า B วางใน Location 3 และสินค้า A วางใน Location 5 ดังนั้นเมื่อสิ้นวันที่ 1 จะมีสินค้าอยู่ครบทุก Location ในวันที่ 2 มีสินค้าออกสินค้าเดียวคือสินค้า ข จึงทำให้ Location ที่ 4 วางและมีสินค้าเข้าสินค้าเดียวคือสินค้า D ดังนั้น สินค้า D จึงถูกวางใน Location 4 โดยปริยาย หากทำตามขั้นตอนนี้ไปเรื่อยๆ คือพิจารณาไปที่ละวันจนครบ 7 วัน ก็จะได้คำตอบดังในตารางที่ 5 และมีผลรวมระยะทางในการเคลื่อนย้ายสินค้าเท่ากับ 5,560 เมตร

ตารางที่ 5 ผลการจัดวางสินค้าตามแนวทาง Fastest Mover Closest to the Door แต่พิจารณาแบบวันต่อวัน (มีระยะทางรวม 5,560 เมตร)

วันที่ (w)	Location(i)				
	(i) = 1	(i) = 2	(i) = 3	(i) = 4	(i) = 5
1	C	ก	B	ข	A
2	C	ก	B	D	A
3	C	E		D	A
4	G	E	F	D	A
5	G	E	F	I	H
6	J		F	I	H
7	J				H

3. แบบจำลองคณิตศาสตร์ที่ 1

จากผลการจัดวางสินค้าตามตารางที่ 5 ยังไม่ได้ นำข้อมูลวันที่สินค้าจะเข้าและออกในตารางที่ 1 มาใช้ประกอบในการพิจารณาเลือก Location ให้กับสินค้า ดังนั้นผู้วิจัยจึงได้พัฒนาแบบจำลองคณิตศาสตร์แบบแรกที่สามารถจัดวางสินค้าโดยอาศัยข้อมูลวันเข้าออกสินค้าและใช้แนวทาง Fastest Mover Closest



to the Door ซึ่งแบบจำลองนี้ถือว่าเป็นปัญหาการจับคู่ 3 มิติ หรือ Three-Dimensional Assignment Problem นั่นคือการจับคู่ระหว่าง Location สินค้า และ วันที่ ซึ่งสามารถแสดงเป็นแบบจำลองคณิตศาสตร์ได้ ดังนี้ Notationsของแบบจำลองที่ 1:

- f_i อัตราการหมุนเวียนของสินค้า j
 D_i ระยะทางจากประตูไปยัง Location i (หน่วย: เมตร)
 X_{wij} ตัวแปรตัดสินใจในการเลือก Location i ของสินค้า j ณ วันที่ w
 ถ้า X_{wij} เท่ากับ 0 แสดงว่าไม่วางสินค้า j ที่ Location i ณ วันที่ w
 ถ้า X_{wij} เท่ากับ 1 แสดงว่าวางสินค้า j ที่ Location i ณ วันที่ w
 Y_{ij} ตัวแปรตัดสินใจในการเลือก Location i ของสินค้า j
 ถ้า Y_{ij} เท่ากับ 0 แสดงว่าไม่วางสินค้า j ที่ Location i

- ถ้า Y_{ij} เท่ากับ 1 แสดงว่าวางสินค้า j ที่ Location i
 h_{wi} ตัวแปรนำเข้า เพื่อบอกสถานะของ Location i ณ วันที่ w ว่ามีสินค้าเก่าค้างหรือไม่
 ถ้า h_{wi} เท่ากับ 0 แสดงว่า Location i ณ วันที่ w ไม่มีสินค้าเก่าวางอยู่
 ถ้า h_{wi} เท่ากับ 1 แสดงว่า Location i ณ วันที่ w มีสินค้าเก่าวางอยู่
 R_{wj} ตัวแปรนำเข้า เพื่อบอกถึงความต้องการ Location ของแต่ละสินค้า j ในวันที่ w ถ้า R_{wj} เท่ากับ 0 แสดงว่า สินค้า j ไม่ต้องการ Location ให้วาง ณ วันที่ w
 ถ้า R_{wj} เท่ากับ 1 แสดงว่า สินค้า j ต้องการ Location ให้วาง ณ วันที่ w
 m จำนวนทั้งหมดของ Location ($i=1,...,m$)
 n จำนวนทั้งหมดของสินค้า ($j=1,...,n$)
 d จำนวนวันที่จะวางแผนล่วงหน้า ($w=1,...,d$)

แบบจำลองที่ 1:

$$Min \sum_{i=1}^m \sum_{j=1}^n 4f_j D_i y_{ij}$$

Subject to

$$\sum_{i=1}^n x_{wij} + h_{wi} \leq 1, \quad \text{for } w=1,...,d, \text{ for } i=1,...,m, \quad (1)$$

$$\sum_{i=1}^m X_{wij} = R_{wj}, \quad \text{for } w=1,...,d, \text{ for } j=1,...,n, \quad (2)$$

$$\sum_{i=1}^m y_{ij} = 1, \quad \text{for } j=1,...,n \quad (2)$$

$$X_{wij} \leq y_{ij}, \quad \text{for } w=1,...,d, \text{ for } i=1,...,m, \text{ for } j=1,...,n, \quad (3)$$

$$X_{wij} \in \{0,1\}, \quad \text{for } w=1,...,d, \text{ for } i=1,...,m, \text{ for } j=1,...,n,$$

$$y_{ij} \in \{0,1\}, \quad \text{for } i=1,...,m, \text{ for } j=1,...,n.$$

โดยแต่ละสมการสามารถอธิบายได้ดังนี้สมการที่ (1) เป็นสมการเป้าหมายเพื่อต้องการให้ได้ระยะทางรวมในการเคลื่อนย้ายสินค้าที่ต่ำที่สุด (หน่วย: เมตร) สำหรับค่าสัมประสิทธิ์เท่ากับ 4 นั้นหมายถึง การเดินไปมาระหว่างประตูกับ Location ใดๆ จำนวน 4 ครั้ง โดย 2 ครั้งแรกเป็นการนำของเข้าคลังสินค้า ขาไปขน

สินค้าไป Location ส่วนขากลับเดินเที่ยวเปล่ากลับมาที่ประตู ส่วน 2 ครั้งหลังเป็นการไปนำสินค้าออกจากคลัง ขาไป Location จากประตูเดินเที่ยวเปล่า ขากลับขนสินค้ากลับมาที่ประตูด้วย สมการที่ (2) กำหนดให้ ในแต่ละวัน สามารถจัดวางสินค้าได้ในเฉพาะ Location ที่ว่างอยู่เท่านั้น ซึ่งตัวแปรนำเข้า h_{wi} จะเป็น

ตัวบ่งบอกสถานะของ Location i ว่ามีสินค้าเก่าคงค้างอยู่ ณ วันที่ w หรือไม่สมการที่ (3) กำหนดว่า หากสินค้าใดต้องการ Location ในวันไหน ก็ต้องสามารถหา Location เพื่อจัดวางสินค้านั้นได้ R_{wj} คือตัวแปรนำเข้าไปบอกถึงความต้องการ Location ของแต่ละสินค้า j ในวันที่ w สมการที่ (4) กำหนดให้ แต่ละสินค้าต้องถูกจัดวางอยู่ใน Location เดียว ไม่สามารถวางใน 2 Location หรือมากกว่าได้ สมการที่ (5) กำหนดความสัมพันธ์ระหว่างตัวแปรตัดสินใจ x_{wij} และ y_{ij} นั่นคือ ถ้าไม่ได้เลือกให้สินค้า j ถูกจัดวางใน Location i แล้ว (หรือ $y_{ij} = 0$) ก็ไม่สามารถวางสินค้า j ลงใน Location i ในวันใดๆ ได้เลย (นั่นคือ x_{wij} จะถูกบังคับให้มีค่าเป็นศูนย์ทั้งหมด) และสมการที่ (6) และ (7) กำหนดให้ตัวแปรตัดสินใจทุกตัวมีค่าเป็น 0 หรือ 1 ได้เท่านั้น

จากโจทย์ของปัญหาจำลองอย่างง่ายในหัวข้อ 2 สามารถกำหนดค่าให้กับตัวแปรนำเข้า h_{wi} และ R_{wj} ดังในตารางที่ 6 และ 7 และเมื่อใช้วิธี Linear Programming หาคำตอบของแบบจำลองที่มีสมการ (1)-(7) สามารถได้ผลการจัดวางสินค้าดังในตารางที่ 8 ที่มีผลรวมระยะทางในการเคลื่อนย้ายสินค้าเท่ากับ 5,480 เมตร ซึ่งน้อยกว่าระยะทางที่ได้จากการจัดวางสินค้าตามแนวทาง Fastest Mover Closest to the Door แต่พิจารณาแบบวันต่อวันดังในตารางที่ 5 (หรือ 5,560 เมตร) อยู่ 80 เมตร จะเห็นได้ว่าการนำข้อมูลวันที่สินค้าจะเข้าและออกมาพิจารณาเพื่อเลือก Location ให้กับสินค้านั้น สามารถลดระยะทางรวมลงไปได้ เพราะจะมีการเลือก Location ที่มีระยะทางจากประตูที่ต่ำลงให้กับสินค้าที่อัตราหมุนเวียนที่สูงขึ้นแต่เข้ามาทีหลัง (หรือมีวันที่สินค้าจะเข้ามากกว่า) นั่นคือมีการสำรอง Location ที่ดีไว้สำหรับสินค้าที่มีอัตราการหมุนเวียนสูงแต่มีกำหนดเข้าคลังสินค้าในวันหลังๆ หรือวันถัดๆ ไป

ตารางที่ 6 ค่าของตัวแปรนำเข้า h_{wi} ซึ่งแสดงสถานะของแต่ละ Location ในแต่ละวันว่ามีสินค้าคงค้างวางอยู่ หรือไม่และจนถึงวันไหน

	Location (i)				
วันที่(w)	i=1	i=2	i=3	i=4	i=5
1	0	1	0	1	0
2	0	1	0	0	0
3	0	0	0	0	0

	Location (i)				
วันที่(w)	i=1	i=2	i=3	i=4	i=5
4	0	0	0	0	0
5	0	0	0	0	0
6	0	0	0	0	0
7	0	0	0	0	0

ตารางที่ 7 ค่าของตัวแปรนำเข้า R_{wj} ซึ่งแสดงถึงความต้องการ Location ของสินค้า j ในแต่ละวันที่ w ของปัญหาจำลองอย่างง่าย

	สินค้า (j)									
วันที่(w)	A	B	C	D	E	F	G	H	I	J
1	1	1	1	0	0	0	0	0	0	0
2	1	1	1	1	0	0	0	0	0	0
3	1	0	1	1	1	0	0	0	0	0
4	1	0	0	1	1	1	1	0	0	0
5	0	0	0	0	1	1	1	1	1	0
6	0	0	0	0	0	1	0	1	1	1
7	0	0	0	0	0	0	0	1	0	1

ตารางที่ 8 ผลการจัดวางสินค้าตามแนวทาง Fastest Mover Closest to the Door และมีการวางแผนล่วงหน้าทั้ง 7 วันด้วยแบบจำลองสมการ 1-7 (มีระยะทางรวม 5,480 เมตร)

	Location (i)				
วันที่(w)	i=1	i=2	i=3	i=4	i=5
1	C	ก	A	ข	B
2	C	ก	A	D	B
3	C		A	D	E
4	G	F	A	D	E
5	G	F	I	H	E
6	J	F	I	H	
7	J			H	

4. แบบจำลองคณิตศาสตร์ที่ 2

จากงานวิจัยของชนิกานต์ กมลสุข [6] เรื่อง “การพัฒนาแบบจำลองคณิตศาสตร์เพื่อช่วยในการกำหนดตำแหน่งในการจัดเก็บสินค้าที่เหมาะสม กรณีศึกษา บริษัทเครื่องแก้วสำเร็จรูป” สามารถดัดแปลงปัญหาการจับคู่ในรูปแบบ 3 มิติ คือ Location สินค้า และวันที่ ให้อยู่ในรูปแบบของการจับคู่ 2 มิติ คือ Location และสินค้า เท่านั้น ส่วนมิติของวันนั้นจะสร้างสมการที่มีตัวแปรนำเข้า $S_{j_1j_2}$ เข้ามาทดแทน โดยตัวแปรนำเข้า $S_{j_1j_2}$ จะระบุว่าคู่สินค้า j_1, j_2 ใดๆ มีวันเข้าและออกทับซ้อนกันหรือไม่ ยกตัวอย่างเช่น ในปัญหาจำลองอย่างง่ายของเรา หากดูในตารางที่ 1 จะเห็นว่า สินค้า D จะเข้าในวันที่ 2 และออกในวันที่ 5 ส่วนสินค้า E จะเข้าในวันที่ 3 และออกในวันที่ 6 ดังนั้น สินค้า D และ E จะไม่สามารถถูกจัดวางใน Location เดียวกันได้ เพราะช่วงเวลาของวันเข้าออกมี การทับซ้อนกัน ดังนั้นค่า $S_{DE} = 0$ (หรือ $S_{ED} = 0$) แต่ถ้าดูสินค้า A (เข้าวันที่ 1 ออกวันที่ 5) และสินค้า J (เข้าวันที่ 6 ออกหลังจากวันที่ 7) สินค้าสองชนิดนี้ถ้าจะเลือกวางใน Location

Model ของชนิกานต์ กมลสุข [6]:

$$\text{Min } \sum_{i=1}^m \sum_{j=1}^n C_i x_{ij} \quad (4)$$

Subject to

$$\sum_{i=1}^m x_{ij} = 1, \quad \text{for } j=1, \dots, n, \quad (5)$$

$$x_{ij_1} + x_{ij_2} \leq 1 + S_{j_1j_2}, \quad \text{for } j_1=1, \dots, n, \text{ for } j_2=1, \dots, n \text{ where } j_1 < j_2, \\ \text{for } i=1, \dots, m, \quad (6)$$

$$x_{ij} \in \{0, 1\}, \quad \text{for } j=1, \dots, n, \text{ for } i=1, \dots, m. \quad (7)$$

โดยแต่ละสมการ สามารถอธิบายได้ดังนี้ สมการที่ (8) คือสมการเป้าหมายที่ต้องการคำตอบที่มีระยะเวลารวมต่ำที่สุดในการเคลื่อนย้ายสินค้า (หน่วย: นาที) สมการที่ (9) กำหนดให้ในแต่ละสินค้าต้องมี Location ให้วางเท่ากับ 1 Location พอดีสมการที่ (10) กำหนดว่า หากสินค้า j_1 และ j_2 มีวันที่เข้าและวันที่ออกทับซ้อนกัน จะไม่สามารถวางอยู่ใน Location เดียวกันได้จะต้องเลือกวางสินค้าใดสินค้าหนึ่งเท่านั้น และสมการที่

เดียวกันก็สามารถวางได้เพราะวันเข้าออกไม่มีการทับซ้อนกัน ดังนั้นค่า $S_{AJ} = 1$ (หรือ $S_{JA} = 1$)

แบบจำลองคณิตศาสตร์งานวิจัยของชนิกานต์ กมลสุข [6] มีรายละเอียดดังต่อไปนี้

C_i ระยะเวลาในการเข้าและออกของ Location i (หน่วย: นาที)

x_{ij} ตัวแปรตัดสินใจในการเลือกตำแหน่งสินค้า j ของ Location i

ถ้า x_{ij} เท่ากับ 0 แสดงว่าสินค้า j ไม่เลือก Location i

ถ้า x_{ij} เท่ากับ 1 แสดงว่าสินค้า j เลือก Location i

$S_{j_1j_2}$ ตัวแปรนำเข้าที่ระบุว่าคู่สินค้า j_1, j_2 ใดๆ มีวันเข้าและออกทับซ้อนกันไหม

ถ้า $S_{j_1j_2}$ เท่ากับ 0 แสดงว่าในวันเข้าและวันออกของสินค้า j_1 และสินค้า j_2 ทับซ้อนกัน

ถ้า $S_{j_1j_2}$ เท่ากับ 1 แสดงว่าในวันเข้าและวันออกของสินค้า j_1 และสินค้า j_2 ไม่ทับซ้อนกัน

m จำนวนทั้งหมดของ Location ($i=1, \dots, m$)

n จำนวนทั้งหมดของสินค้า ($j=1, \dots, n$)

(11) กำหนดให้ตัวแปรตัดสินใจมีค่าเป็นได้แค่ 0 หรือ 1 เท่านั้น อย่างไรก็ตามวิธีคิดแบบจำลองคณิตศาสตร์ของชนิกานต์ กมลสุข [6] ยังไม่สามารถแก้ปัญหาของตัววิจัยได้ เพราะว่ายังไม่ได้คำนึงถึงสินค้าเก่าที่มีคงค้างอยู่ใน Location (เช่น สินค้า ก และ ข ในปัญหาจำลองอย่างง่าย) และสมการเป้าหมายยังเป็นเวลารวมไม่ใช่ระยะทางรวม ดังนั้นผู้วิจัยจึงปรับเปลี่ยนแบบจำลองคณิตศาสตร์ของชนิกานต์ กมลสุข [6] ให้สามารถรองรับปัญหาของ

ผู้วิจัยได้จัดเป็นแบบจำลองดังในสมการ (12) - (16)

Notations ของแบบจำลองที่ 2:

f_j อัตราการหมุนเวียนของสินค้า j
 D_i ระยะทางในการเคลื่อนย้ายสินค้าเข้าและออก
 ของ Location i (หน่วย: เมตร)
 X_{ij} ตัวแปรตัดสินใจในการเลือกตำแหน่งสินค้า j
 ของ Location i
 ถ้า x_{ij} เท่ากับ 0 แสดงว่าสินค้า j ไม่เลือก
 Location i
 ถ้า x_{ij} เท่ากับ 1 แสดงว่าสินค้า j เลือก Location i
 $S_{j_1j_2}$ ตัวแปรนำเข้าที่ระบุว่าคู่สินค้า j_1, j_2 ใดๆ มีวันเข้า
 และออกทับซ้อนกันไหม
 ถ้า $S_{j_1j_2}$ เท่ากับ 0 แสดงว่าในวันเข้าและวันออก
 ของสินค้า j_1 และสินค้า j_2 ทับซ้อนกัน
 ถ้า $S_{j_1j_2}$ เท่ากับ 1 แสดงว่าในวันเข้าและวันออก

ของสินค้า j_1 และสินค้า j_2 ไม่ทับซ้อนกัน
 m จำนวนทั้งหมดของ Location ($j=1, \dots, m$)
 n จำนวนของสินค้าที่เข้ามาใหม่ก่อนหาคำตอบ
 ($J=1, \dots, n$)
 P จำนวนสินค้าเก่าที่ตกค้างอยู่ในคลังสินค้าก่อน
 หาคำตอบ
 k ดัชนีของสินค้าเก่าที่ตกค้างอยู่ในคลังสินค้าก่อน
 หาคำตอบ ($k = n+1, \dots, n+P$)
 a_k ตัวแปรนำเข้าระดับดัชนีของ Location ที่สินค้า
 เก่า k ที่ตกค้างอยู่ใน Location ก่อนหาคำตอบ
 ยกตัวอย่างเช่น ในปัญหาจำลองอย่างง่ายมีสินค้า
 เก่า ก, ข ดังนั้น $p=2, n=10, k=11, 12$, ดังนั้น
 ตัวแปรนำเข้า a_{11} คือ Location ที่สินค้า ก อยู่
 เพราะฉะนั้น $a_{11}=2$ หรือ $a_{12}=4$

Model ของแบบจำลองที่ 2:

$$\text{Min} \sum_{i=1}^m \sum_{j=1}^{n+p} 4f_j D_i x_{ij} \quad (8)$$

Subject to

$$\sum_{i=1}^m x_{ij} = 1, \quad \text{for } j=1, \dots, n+P, \quad (9)$$

$$x_{ij_1} + x_{ij_2} \leq 1 + S_{j_1j_2}, \quad \text{for } j_1=1, \dots, n+P, \text{ for } j_2=1, \dots, n+P \quad (10)$$

where $j_1 < j_2$ for $i=1, \dots, m$,

$$x_{a_k k} = 1, \quad \text{for } k=n+1, \dots, n+P, \quad (11)$$

$$x_{ij} \in \{0, 1\}, \quad \text{for } i=1, \dots, m, \text{ for } j=1, \dots, n+P. \quad (12)$$

โดยแต่ละสมการอธิบายได้ดังนี้สมการที่ (12) คือ
 สมการเป้าหมายที่ต้องการหาคำตอบที่มีระยะทางรวมที่
 ต่ำที่สุดในการเคลื่อนย้ายสินค้า (หน่วย: เมตร) สมการที่
 (13) กำหนดให้ในแต่ละสินค้าต้องมี Location ให้วาง
 เท่ากับ 1 Location พอดี สมการที่ (14) กำหนดว่าถ้า
 หากสินค้า j_1 และ j_2 มีวันที่เข้าและวันที่ออกทับซ้อนใน
 วันเดียวกัน จะไม่สามารถวางอยู่ใน Location เดียวกัน
 ได้ สมการที่ (15) เป็นสมการที่กำหนดให้ตัวแปร
 ตัดสินใจของสินค้าเก่าที่ตกค้างอยู่ในคลังสินค้าจะต้องเลือก
 Location ที่สินค้านั้นคงค้างอยู่ใน Location ก่อนเริ่ม
 หาคำตอบ และสมการที่ (16) กำหนดให้ตัวแปรตัดสินใจ
 มีค่าเป็นได้แค่ 0 หรือ 1 เท่านั้น

สำหรับปัญหาจำลองอย่างง่าย สามารถนำข้อมูลวัน
 ที่เข้าและออกของสินค้าต่างๆ ที่แสดงในตารางที่ 1 มา
 หาค่าตัวแปรนำเข้า $s_{j_1j_2}$ ได้ดังในตารางที่ 9 โดยผู้วิจัย

ได้พัฒนาโปรแกรม Visual Basic for Applications
 (VBA) บน Microsoft Excel ที่สามารถคำนวณหาค่า
 ของ $s_{j_1j_2}$ ได้โดยอัตโนมัติ และเมื่อแก้ปัญหาจำลองอย่าง
 ง่ายด้วยวิธี Linear Programming ของแบบจำลอง
 ที่ 2 (สมการ 12-16) จะได้คำตอบเดียวกับคำตอบที่
 ได้จากแบบจำลองที่ 1 คือได้ผลรวมระยะทางเท่ากับ
 5,480 เมตร เช่นเดียวกัน ดังในตารางที่ 8

ตารางที่ 9 ค่าของตัวแปรนำเข้า $s_{j_1j_2}$ ที่ระบุว่าคู่สินค้า
 j_1, j_2 ใดๆ มีวันเข้าและออกทับซ้อนกันหรือไม่ของปัญหา
 จำลองอย่างง่าย

		สินค้า (j_2)											
สินค้า (j_1)		A	B	C	D	E	F	G	H	I	J	ก	ข
	A	0	0	0	0	0	1	1	0	1	1	0	0
	B	0	0	0	0	0	0	0	0	1	0	1	1



		สินค้า (j_2)											
		A	B	C	D	E	F	G	H	I	J	ก	ข
สินค้า (j_1)	C	0	0	0	0	0	1	1	0	1	1	0	1
	D	0	0	0	0	1	0	0	1	0	0	1	1
	E	0	0	0	1	0	1	1	0	1	1	0	0
	F	1	0	1	0	1	0	0	1	0	0	1	1
	G	1	0	1	0	1	0	0	1	0	0	1	1
	H	0	0	0	1	0	1	1	0	1	1	0	0
	I	1	1	1	0	1	0	0	1	0	0	1	1
	J	1	0	1	0	1	0	0	1	0	0	1	1
	ก	0	1	0	1	0	1	1	0	1	1	0	0
	ข	0	1	1	1	0	1	1	0	1	1	0	0

5. การทดสอบเปรียบเทียบประสิทธิภาพของแบบจำลองคณิตศาสตร์ทั้ง 2 แบบจำลอง

เพื่อศึกษาประสิทธิภาพและความสามารถในการหาคำตอบของแบบจำลองคณิตศาสตร์ที่ 1 (สมการ 1 - 7) และแบบจำลองที่ 2 (สมการ 12-16) ผู้วิจัยจึงได้สร้างปัญหาทดสอบ (Test Problem) จำนวน 35 ปัญหา โดยใน 35 ปัญหานี้มีจำนวนวัน (ค่า d) ในช่วง 7-20 วัน จำนวนสินค้า (ค่า n) อยู่ในช่วง 10-200 สินค้า และจำนวน Location (ค่า m) อยู่ในช่วง 5 - 120 Location ในแต่ละปัญหา ค่าอัตราหมุนเวียนของแต่ละสินค้า (fi) ระยะทางของแต่ละ Location (Di) และวันที่เข้าและวันที่ออกจากคลังสินค้าของแต่ละสินค้าจะถูกสุ่มอย่างอิสระ ส่วนจำนวนสินค้าเก่าที่คงค้างอยู่ในคลังสินค้ามีค่าอยู่ในช่วง 2-103 สินค้า

เครื่องคอมพิวเตอร์ที่ผู้วิจัยใช้ในการทดสอบคือรุ่น AsusIntel(R) Core i5 มีความเร็ว 2.4GHz มีหน่วยความจำ 4 GB สำหรับโปรแกรมคอมพิวเตอร์ที่ผู้วิจัย

$$= \frac{\text{จำนวนตัวแปรของปัญหาทดสอบที่ 1 ของแบบจำลองที่ 2} - \text{จำนวนตัวแปรของปัญหาทดสอบที่ 1 ของแบบจำลองที่ 1}}{\text{จำนวนตัวแปรของปัญหาทดสอบที่ 1 ของแบบจำลองที่ 1}} \times 100$$

$$\% dif = \frac{82-407}{407} \times 100 = -79.8\% \sim -80.0\%$$

ใช้ในการหาคำตอบของวิธี Linear Programming คือโปรแกรม LingoVersion 14.0 และผู้วิจัยยังเขียนโปรแกรมที่สามารถหาค่า R_{wj} (สำหรับแบบจำลองที่ 1) และค่า $S_{j_{ij_2}}$ (สำหรับแบบจำลองที่ 2) ได้โดยอัตโนมัติ โดยโปรแกรมดังกล่าวถูกพัฒนามาบน Visual Basicfor Applications (VBA) ของโปรแกรม MicrosoftExcel รุ่น 2010

ในการทดสอบประสิทธิภาพของแต่ละแบบจำลองคณิตศาสตร์ ผู้วิจัยได้บันทึกค่าสมการเป้าหมาย (Objective Function) ซึ่งก็คือระยะทางรวมในการเคลื่อนย้ายสินค้าในคลังสินค้า (มีหน่วยเป็นเมตร) จำนวนครั้ง (Iteration) ที่โปรแกรม Lingo เรียกใช้ Simplex Algorithm ในกระบวนการ Branch-and-Bound เวลาในการประมวลผลหาคำตอบของโปรแกรม Lingo (หน่วยเป็นวินาที) จำนวนขีดจำกัด (Constraints) และจำนวนตัวแปร (Variables) ที่โปรแกรม Lingo สร้างเป็น Model ของตัวเองขึ้นมา รวมทั้งเวลา (มีหน่วยเป็นวินาที) ที่โปรแกรม VBA ใน MicrosoftExcel ใช้ในการสร้างค่าของตัวแปรนำเข้า 2 ตัวแปรคือค่า R_{wj} (สำหรับแบบจำลองที่ 1) และค่า $S_{j_{ij_2}}$ (สำหรับแบบจำลองที่ 2) ผลการทดสอบเปรียบเทียบประสิทธิภาพของทั้ง 2 แบบจำลองแสดงอยู่ในตารางที่ 10 และ 11 ตารางที่ 11 แสดงเปอร์เซ็นต์ความแตกต่าง (% dif) โดยจะนำค่าที่สนใจจะเปรียบเทียบของแบบจำลองที่ 2 ลบด้วยค่าของแบบจำลองที่ 1 และหารด้วยค่าของแบบจำลองที่ 1 แล้วแปลงค่าเป็นเปอร์เซ็นต์ด้วยการคูณ 100 ยกตัวอย่างเช่น เปอร์เซ็นต์ความแตกต่างของ “จำนวนตัวแปร” ของปัญหาทดสอบที่ 1 มีค่าเท่ากับ 80% ซึ่งคำนวณมาจากสูตรดังนี้

ถ้าค่า % *dif* ตีกลับแสดงว่า แบบจำลองที่ 2 มีค่าที่นำมาเปรียบเทียบน้อยกว่าของแบบจำลองที่ 1

จากการทดสอบประสิทธิภาพด้วยปัญหาเพื่อการทดสอบจำนวน 35 ปัญหา พบว่า แบบจำลองทางคณิตศาสตร์ที่ 1 สามารถหาคำตอบที่ดีที่สุด (Optimal Solution) ได้ครบทั้ง 35 ปัญหา แต่แบบจำลองที่ 2 สามารถหาคำตอบที่ดีที่สุดได้เพียงแค่ 28 ปัญหาแรกเท่านั้น ส่วนในปัญหาที่ 29-35 โปรแกรม Lingo ไม่สามารถหาคำตอบได้ เนื่องจากโปรแกรม Lingo มีหน่วยความจำไม่เพียงพอ (Out of Memory หรือ mem-out) ในการสร้างตัวแบบ (Model Generation) ซึ่งเป็นไปได้ว่าถ้าใช้เครื่องคอมพิวเตอร์ที่มีหน่วยความจำมากกว่านี้ หรือปรับค่าหน่วยความจำในระบบปฏิบัติการ Windows ให้โปรแกรม Lingo มีหน่วยความจำมากขึ้นก็อาจทำให้ในแบบจำลองที่ 2 สามารถหาคำตอบที่ดีที่สุดได้ครบทั้ง 35 ปัญหาได้

สำหรับจำนวนตัวแปร (Variable) พบว่า แบบจำลองที่ 2 ใช้จำนวนตัวแปรน้อยกว่าแบบจำลองที่ 1 เสมอตลอดทั้ง 35 ปัญหา และเมื่อวิเคราะห์อัตราเติบโตของจำนวนตัวแปรเมื่อปัญหาทดสอบมีขนาดใหญ่ขึ้นพบว่าแบบจำลองที่ 1 มีอัตราเติบโตที่สูงกว่าแบบจำลองที่ 2 มาก แต่อย่างไรก็ดีแม้ปัญหาจะมีขนาดเท่าไรก็ตาม พบว่า % *dif* ของจำนวนตัวแปรจะอยู่คงที่ในช่วง 74% ถึง 95%

สำหรับจำนวนขีดจำกัด (Constraints) พบว่ามีลักษณะที่ซับซ้อนกว่า นั่นคือ แบบจำลองที่ 2 จะใช้จำนวนขีดจำกัดน้อยกว่าแบบจำลองที่ 1 ใน 14 ปัญหาแรก (% *dif* อยู่ในช่วง 4% ถึง 69%) ส่วนในปัญหาที่ 15-35 แบบจำลองที่ 2 จะใช้จำนวนขีดจำกัดมากกว่าแบบจำลองที่ 1 และมีอัตราเติบโตของจำนวนขีดจำกัดที่มากกว่า (% *dif* อยู่ในช่วง 32% ถึง 1028%) สาเหตุหนึ่งที่แบบจำลองที่ 2 ตั้งแต่ปัญหาที่ 29-35 ไม่สามารถหาคำตอบได้เนื่องจากมีหน่วยความจำไม่เพียงพอ นั้นสามารถวิเคราะห์ได้ว่าเป็นเพราะจำนวนขีดจำกัดที่เพิ่ม

ขึ้นของแบบจำลองที่ 2 นั่นคือเมื่อขีดจำกัดมีจำนวนเพิ่มขึ้นจะใช้หน่วยความจำมากกว่าจำนวนตัวแปรที่เพิ่มขึ้นของแบบจำลองที่ 1

สำหรับจำนวนครั้ง (Iteration) ที่โปรแกรม Lingo เรียกใช้ Simplex Algorithm ในกระบวนการ Branch-and-Bound พบว่า แบบจำลองที่ 2 จะใช้จำนวน Iteration น้อยกว่าแบบจำลองที่ 1 เสมอ โดยค่า % *dif* อยู่ในช่วง 12% ถึง 100%

สำหรับระยะเวลาที่ใช้หาคำตอบของโปรแกรม Lingo ที่สามารถนำมาเปรียบเทียบได้คือ ตั้งแต่ปัญหาที่ 11-28 เท่านั้น เนื่องจากปัญหาที่ 1-10 ไม่สามารถเปรียบเทียบได้เพราะเวลาที่ทั้งสองแบบจำลองใช้มีค่าไม่ถึง 1 วินาที หรือเกือบจะเป็น 0 วินาทีเป็นส่วนใหญ่ในการหาคำตอบ ทำให้ไม่สามารถคำนวณหาค่า % *dif* ได้ซึ่งได้แสดงด้วยอักษรย่อ “n/a” แทน และปัญหาที่ 29-35 แบบจำลองที่ 2 ไม่สามารถหาคำตอบได้ เนื่องจากโปรแกรม Lingo มีหน่วยความจำไม่เพียงพอในส่วนที่เปรียบเทียบได้ จะเห็นได้ว่า แบบจำลองที่ 2 ใช้เวลาในการหาคำตอบด้วยโปรแกรม Lingo น้อยกว่าแบบจำลองที่ 1 เกือบเท่าตัวคือมี % *dif* อยู่ในช่วง 96% ถึง 100% หรือสามารถกล่าวได้ว่าระยะเวลาที่ใช้หาคำตอบในโปรแกรม Lingo แปรผันตามจำนวน Iteration ของโปรแกรม Lingo

สำหรับระยะเวลาที่ใช้ในการเตรียมค่าของตัวแปรนำเข้า นั่นคือ ค่า (สำหรับแบบจำลองที่ 1) และค่า (สำหรับแบบจำลองที่ 2) พบว่า ตามค่า % *dif* แบบจำลองที่ 2 ใช้เวลามากกว่าแบบจำลองที่ 1 มาก นั่นคือ % *dif* อยู่ในช่วง 700% ถึง 3900% อย่างไรก็ตามแบบจำลองที่ 1 ไม่เคยใช้เวลาเกิน 2 วินาทีในทั้ง 35 ปัญหา ส่วนแบบจำลองที่ 2 ใช้เวลาไม่เกิน 40 วินาทีในปัญหาที่ใหญ่ที่สุด สำหรับในปัญหา 1-13 ไม่สามารถหา % *dif* ได้เพราะเวลาที่ใช้ในแบบจำลองที่ 1 มีค่าน้อยมากคือไม่เกิน 1 วินาทีหรือแทบจะเป็น 0 วินาที



ตารางที่ 10 ผลการทดสอบเปรียบเทียบประสิทธิภาพการหาคำตอบของทั้ง 2 แบบจำลอง

	ปัญหาทดสอบ Test Problem					แบบจำลองที่ 1						แบบจำลองที่ 2					
	จำนวนวัน (d)	จำนวนสินค้า(n)	จำนวน Location (m)	จำนวนสินค้าเก่าที่ตั้งค้างอยู่ใน Location	ระยะทางรวม (Objective function value)	จำนวนรอบ (Iteration)	จำนวนตัวแปร(Variable)	จำนวนเงื่อนไข (Constraint)	เวลาประมวลผลด้วยโปรแกรม Lingo (วินาที)	เวลาประมวลผลด้วยโปรแกรม VBA (วินาที)	ระยะทางรวม (Objective function value)	จำนวนรอบ (Iteration)	จำนวนตัวแปร(Variable)	จำนวนเงื่อนไข (Constraint)	เวลาประมวลผลด้วยโปรแกรม Lingo (วินาที)	เวลาประมวลผลด้วยโปรแกรม VBA (วินาที)	
1	7	10	5	2	5,480	45	407	466	0	0	5,480	29	82	343	0	1	
2	7	10	5	2	5,680	41	407	466	0	0	5,680	-	82	343	0	1	
3	11	12	5	2	372	41	731	860	0	0	372	36	88	343	0	1	
4	11	12	5	2	1,420	56	731	860	0	0	1,420	28	96	470	0	1	
5	11	12	5	3	4,160	27	731	860	0	0	4,160	-	102	541	0	1	
6	15	15	5	2	5,160	86	1,215	1,441	0	0	5,160	47	117	798	0	1	
7	15	15	5	3	5,780	81	1,215	1,441	0	0	5,780	42	123	784	0	1	
8	15	15	5	3	8,040	71	1,215	1,441	0	0	8,040	37	123	784	0	2	
9	15	20	5	3	11,380	99	1,615	1,896	0	0	11,380	66	158	1,289	0	2	
10	20	20	5	2	18,680	161	2,120	2,521	0	0	18,680	77	123	784	0	2	
11	20	30	10	4	44,440	40,050	6,320	6,831	7	0	44,440	202	404	5,645	0	7	
12	20	30	10	2	46,700	104,299	6,320	6,831	15	0	46,700	163	382	4,993	0	7	
13	20	40	10	2	63,040	132,691	8,420	9,041	24	0	63,040	295	502	8,653	0	7	
14	20	40	10	2	47,600	93,504	8,420	9,041	16	1	47,600	338	502	8,653	0	8	
15	20	50	15	4	98,440	237,247	15,770	16,351	59	1	98,440	507	914	21,520	1	8	
16	20	50	15	10	104,740	379,087	15,770	16,351	82	1	104,740	798	1,010	26,611	1	8	
17	20	50	20	4	178,680	206,702	21,020	21,451	71	1	178,680	945	1,184	28,675	1	8	
18	20	50	20	4	181,440	204,727	21,020	21,451	86	1	181,440	788	1,184	28,675	1	8	
19	20	60	20	4	220,120	411,411	25,220	25,661	175	1	220,120	1,324	1,404	40,385	2	8	
20	20	60	20	4	314,000	159,050	25,220	25,661	83	1	314,000	1,295	1,404	40,385	2	8	
21	20	60	30	4	189,820	448,424	37,820	37,861	195	1	189,820	1,724	2,044	60,545	3	9	
22	20	60	30	7	139,760	321,511	37,820	37,861	152	1	139,760	1,860	2,137	66,398	3	9	
23	20	80	40	28	364,120	224,323	67,220	66,481	197	2	364,120	1,917	4,508	231,229	9	14	
24	20	80	40	35	258,140	319,502	67,220	66,481	309	2	258,140	1,715	4,795	262,316	13	18	
25	20	90	50	14	447,960	308,956	94,520	92,891	356	2	447,960	2,452	5,394	267,905	14	14	
26	20	90	50	14	399,380	237,884	94,520	92,891	331	2	399,380	3,063	5,394	267,905	14	14	

27	20	100	60	31	575,880	551,052	126,060	123,301	925	1	575,880	442	8,091	511,032	24	20
28	20	120	60	30	888,540	374,158	151,220	147,721	948	2	888,540	500	9,270	670,651	34	24
29	20	140	70	59	549,260	573,958	205,820	200,341	1769	2	mem-out	mem-out	14,269	1,379,270	mem-out	28
30	20	160	80	27	1,481,420	231,308	268,820	260,961	1553	2	mem-out	mem-out	15,307	1,391,468	mem-out	27
31	20	160	90	73	125,748	283,223	302,420	293,161	2232	1	mem-out	mem-out	21,363	2,432,754	mem-out	34
32	20	160	120	15	367,260	332,817	403,220	389,761	3673	0	mem-out	mem-out	21,335	1,827,176	mem-out	25
33	20	160	120	15	653,600	473,654	403,220	389,761	7523	1	mem-out	mem-out	21,335	1,827,176	mem-out	25
34	20	200	120	52	1,424,240	599,208	504,020	486,601	10753	1	mem-out	mem-out	94,724	3,795,425	mem-out	36
35	20	200	120	103	1,415,860	490,151	504,020	486,601	7604	1	mem-out	mem-out	129,404	5,490,767	mem-out	40

*mem-out หมายถึง โปรแกรม Lingo ไม่สามารถหาคำตอบได้เนื่องจากหน่วยความจำของโปรแกรมไม่เพียงพอในแบบจำลองที่ 2

ตารางที่ 11 เปอร์เซนต์แตกต่าง (% dif) ของตัวแปรต่างๆ ของทั้ง 2 แบบจำลอง

ปัญหาทดสอบ (Test Problem)				จำนวนครั้ง (Iteration)	จำนวนตัวแปร (Variable)	จำนวนข้อจำกัด (Constraint)	เวลาประมวลผลด้วย โปรแกรม Lingo (วินาที)	เวลาประมวลผลด้วย โปรแกรม VBA (วินาที)
	จำนวนวัน (d)	จำนวนสินค้า (n)	จำนวน Location (m)					
1	7	10	5	-36%	-80%	-26%	n/a	n/a
2	7	10	5	-100%	-80%	-26%	n/a	n/a
3	11	12	5	-12%	-88%	-60%	n/a	n/a
4	11	12	5	-50%	-87%	-45%	n/a	n/a
5	11	12	5	-100%	-86%	-37%	n/a	n/a
6	15	15	5	-45%	-90%	-45%	n/a	n/a
7	15	15	5	-48%	-90%	-46%	n/a	n/a
8	15	15	5	-48%	-90%	-46%	n/a	n/a
9	15	20	5	-33%	-90%	-32%	n/a	n/a
10	20	20	5	-52%	-94%	-69%	n/a	n/a
11	20	30	10	-99%	-94%	-17%	-100%	n/a
12	20	30	10	-100%	-94%	-27%	-100%	n/a
13	20	40	10	-100%	-94%	-4%	-100%	n/a
14	20	40	10	-100%	-94%	-4%	-100%	700%
15	20	50	15	-100%	-94%	32%	-98%	700%
16	20	50	15	-100%	-94%	63%	-99%	700%
17	20	50	20	-100%	-94%	34%	-99%	700%
18	20	50	20	-100%	-94%	34%	-99%	700%
19	20	60	20	-100%	-94%	57%	-99%	700%

20	20	60	20	-99%	-94%	57%	-98%	700%
21	20	60	30	-100%	-95%	60%	-98%	800%
22	20	60	30	-99%	-94%	75%	-98%	800%
23	20	80	40	-99%	-93%	248%	-95%	600%
24	20	80	40	-99%	-93%	295%	-96%	800%
25	20	90	50	-99%	-94%	188%	-96%	600%
26	20	90	50	-99%	-94%	188%	-96%	600%
27	20	100	60	-100%	-94%	314%	-97%	1900%
28	20	120	60	-100%	-94%	354%	-96%	1100%
29	20	140	70	mem-out	-93%	588%	mem-out	1300%
30	20	160	80	mem-out	-94%	433%	mem-out	1250%
31	20	160	90	mem-out	-93%	730%	mem-out	3300%
32	20	160	120	mem-out	-95%	369%	mem-out	n/a
33	20	160	120	mem-out	-95%	369%	mem-out	2400%
34	20	200	120	mem-out	-81%	680%	mem-out	3500%
35	20	200	120	mem-out	-74%	1028%	mem-out	3900%

*mem-out หมายถึง โปรแกรม Lingo ไม่สามารถหาคำตอบได้เนื่องจากหน่วยความจำของโปรแกรมไม่เพียงพอในแบบจำลองที่ 2

*n/a หมายถึง ไม่สามารถคำนวณหาค่า % dif ได้เนื่องจากตัวหารหรือค่าของแบบจำลองที่ 1 มีค่าเท่ากับศูนย์

6. สรุปผลการวิจัย

งานวิจัยนี้ได้เสนอวิธีการวางแผนการเลือกตำแหน่งหรือ Location ให้กับสินค้าในคลังสินค้า โดยพิจารณาตามแนวทางของ Fastest Mover Closest to the Door คือพยายามจัดวางสินค้าที่มีอัตราหมุนเวียนมากไว้ใน Location ที่มีระยะทางน้อยหรืออยู่ใกล้ประตูเข้าออกของคลังสินค้ามากที่สุด นอกจากนี้ยังเป็นวิธีการวางแผนล่วงหน้าที่อาศัยข้อมูลวันที่สินค้าจะเข้าและออกมาใช้เลือก Location ให้กับสินค้า ทำให้สามารถสำรอง Location ที่เหมาะสมให้กับสินค้าที่มีอัตราหมุนเวียนสูงกว่าแต่มีกำหนดเข้าคลังช้ากว่าสินค้าที่มาก่อน

วิธีการเลือก Location โดยการสร้างแบบจำลองทางคณิตศาสตร์จำนวน 2 แบบแบบจำลองที่ 1 มีลักษณะเป็นปัญหาการจับคู่ 3 มิติ (Three-Dimensional Assignment Problem) ส่วนแบบจำลองที่ 2 เป็นการลดรูปปัญหาจับคู่ 3 มิติให้เหลือแค่ 2 มิติ อย่างไรก็ตามทั้ง 2 แบบจำลองนี้จะให้คำตอบที่เหมือนกันถ้านำไปแก้ปัญหาเดียวกัน

การทดสอบเปรียบเทียบประสิทธิภาพของแบบจำลองทั้ง 2 ใช้ปัญหาทดสอบ (Test Problem) จำนวน 35 ปัญหา และใช้โปรแกรม Lingo ซึ่งสามารถแก้ปัญหา Linear Programming ได้ หาค่าตัวแปรนำเข้าของทั้งสองแบบจำลอง จากการทดสอบเปรียบเทียบพบว่า แบบจำลองที่ 2 จะสามารถ

หาคำตอบที่ดีที่สุด (Optimal Solution) ได้เร็วกว่าแบบจำลองที่ 1 แต่ว่าแบบจำลองที่ 2 มีอุปสรรคในการแก้ปัญหาขนาดใหญ่ เพราะแบบจำลองที่ 2 ใช้จำนวนขีดจำกัด (Constraints) มากกว่าแบบจำลองที่ 1 ซึ่งส่งผลให้ต้องใช้หน่วยความจำของโปรแกรม Lingo หรือเครื่องคอมพิวเตอร์มาก จนทำให้มีหน่วยความจำไม่เพียงพอในการหาคำตอบของปัญหาขนาดใหญ่ด้วยแบบจำลองที่ 2 ดังนั้นจึงสามารถสรุปแนวทางการแก้ปัญหาได้ว่า แบบจำลองที่ 2 เหมาะกับการแก้ปัญหาขนาดเล็กหรือกลางเพราะสามารถหาคำตอบได้เร็วกว่า ส่วนแบบจำลองที่ 1 เหมาะกับการแก้ปัญหาขนาดใหญ่เพราะใช้หน่วยความจำของคอมพิวเตอร์น้อยกว่า

กิตติกรรมประกาศ

ผู้วิจัยขอขอบคุณบริษัทกรณีศึกษาที่ให้ความร่วมมือในการให้ข้อมูลและความช่วยเหลืออย่างเต็มที่มาโดยตลอด และขอขอบคุณผู้ที่เกี่ยวข้องทุกท่านที่ทำให้งานวิจัยนี้สามารถดำเนินการสำเร็จเป็นอย่างดี

เอกสารอ้างอิง

[1] ไชยยศ ไชยมั่นคง และ มยุขพันธุ์ ไชยมั่นคง. (2550). การจัดการคลังสินค้า ในกลยุทธ์โลจิสติกส์ และซัพพลายเชนเพื่อแข่งขันในตลาดโลก. (หน้า 390-406). นนทบุรี: บริษัท วิชั่น พรีเมียร์ จำกัด.

[2] Tompkins, J. A., and Harmelink, D., 2004, "The Supply Chain Handbook", Tompkins Press, 2004

[3] James and Jerry, 1998 "The Warehouse Management Handbook", second edition, Tompkins Press. 823-848

[4] กิตติ ภัคดีวัฒนกุล และ พนิดา พานิชกุล. (2554). การหาค่าผลลัพธ์ที่ดีที่สุดด้วยโปรแกรมเชิงเส้น (Optimization Linear Programming). ในการวิเคราะห์เชิงปริมาณเพื่อการตัดสินใจ (หน้า 10-12). กรุงเทพฯ: บริษัท เคทีพี คอมพ์ แอนด์ คอนซัลท์ จำกัด.

[5] William P., 1967 "The Multidimensional Assignment Problem," Technical Memorandum Vol.16 No 2, Operations Research. 422-431

[6] ชนิกันต์ กมลสุข (2554) การพัฒนาแบบจำลองคณิตศาสตร์เพื่อช่วยกำหนดตำแหน่งการจัดวางสินค้าที่เหมาะสม: กรณีศึกษาบริษัทผลิตเครื่องแก้วสำเร็จรูป. กรุงเทพมหานคร: การประชุมวิชาการด้านการวิจัยดำเนินงานแห่งชาติ ประจำปี 2554 วันที่ 8-9 กันยายน 2554

[7] Lindo System Inc., 2014 "Lingo 14 – Optimization Modeling Software for Linear, Nonlinear and Integer Programming", Available: <http://www.lindo.com> [29 May 2014]