



โครงการ

เรื่อง

การประเมินประสิทธิภาพของโปรแกรมแก้ปัญหาคำหนดการเชิงเส้น:
กรณีศึกษาการกระจายพนักงานเพื่อเพิ่มยอดขายแพ็คเกจอินเทอร์เน็ต

Performance Evaluation of Linear Programming Solvers:
A Case Study on Workforce Allocation to Maximize Internet
Package Sales

นำเสนอโดย

นางสาว นวดี สิบตะ รหัสนักศึกษา 6604105309

นางสาว นวพร เมืองน้อย รหัสนักศึกษา 6604105310

นางสาว นวพร เลร้รักษา รหัสนักศึกษา 6604105311

รายงานฉบับนี้เป็นส่วนหนึ่งของรายวิชา 10305492 โครงการ

ภาคเรียนที่ 2 ปีการศึกษา 2568

สาขาวิชาคณิตศาสตร์ คณะวิทยาศาสตร์

มหาวิทยาลัยแม่โจ้

โครงงาน

เรื่อง

การประเมินประสิทธิภาพของโปรแกรมแก้ปัญหาคำหนดการเชิงเส้น:
กรณีศึกษาการกระจายพนักงานเพื่อเพิ่มยอดขายแพ็คเกจอินเทอร์เน็ต

Performance Evaluation of Linear Programming Solvers:
A Case Study on Workforce Allocation to Maximize Internet
Package Sales

นำเสนอโดย

นางสาว นวดี สิบตะ รหัสนักศึกษา 6604105309

นางสาว นวพร เมืองน้อย รหัสนักศึกษา 6604105310

นางสาว นวพร เลร้รักษา รหัสนักศึกษา 6604105311


..... อาจารย์ที่ปรึกษา
(ผู้ช่วยศาสตราจารย์ ดร.จิรวรรณ พังขรประกิติ)


..... อาจารย์ที่ปรึกษา
(อาจารย์ ดร.ธวัชชัย เพชรธราทิพย์)

ชื่อ-สกุล	นางสาวนวลี สิบตะ	รหัสนักศึกษา 6604105309
	นางสาวนพร เมืองน้อย	รหัสนักศึกษา 6604105310
	นางสาวนพร เล็กษา	รหัสนักศึกษา 6604105311
อาจารย์ที่ปรึกษา	ผู้ช่วยศาสตราจารย์ ดร.จิรพรรณ พชรประกิตติ	
	อาจารย์ ดร.ธวัชชัย เพชรธราทิพย์	
หัวข้อโครงการ	การประเมินประสิทธิภาพของโปรแกรมแก้ปัญหาคำหนดการเชิงเส้น: กรณีศึกษาการกระจายพนักงานเพื่อเพิ่มยอดขายแพ็คเกจอินเทอร์เน็ต Performance Evaluation of Linear Programming Solvers: A Case Study on Workforce Allocation to Maximize Internet Package Sales	

บทคัดย่อ

เป้าหมายของบทความนี้คือการหาวิธีเพิ่มยอดขายแพ็คเกจอินเทอร์เน็ตให้ได้มากที่สุด ภายใต้งบประมาณการตลาดของบริษัท โดยใช้แบบจำลองกำหนดการเชิงเส้น (Linear Programming) และแก้ปัญหาคำหนดการเชิงเส้นด้วยวิธีซิมเพล็กซ์ (Simplex Method) ซึ่งเป็นวิธีที่สามารถคำนวณหาค่าที่เหมาะสมที่สุดของตัวแปรตัดสินใจหลายตัวแปรได้อย่างมีประสิทธิภาพ งานวิจัยนี้กำหนดตัวแปรตัดสินใจจำนวน 3 ตัวแปร ได้แก่ จำนวนแพ็คเกจที่พนักงานการตลาดอาวุโสขาย (x_1) จำนวนแพ็คเกจที่พนักงานการตลาดรุ่นเยาว์ขาย (x_2) และจำนวนแพ็คเกจที่นักศึกษาฝึกงานด้านการตลาดขาย (x_3) โดยมีเงื่อนไขข้อจำกัดด้านทรัพยากรและงบประมาณ รวมถึงกำหนดให้ยอดขายรวมต้องไม่เกิน 3000 แพ็คเกจ

ในการศึกษาครั้งนี้ได้ทำการเปรียบเทียบการคำนวณผลลัพธ์ด้วยโปรแกรมคอมพิวเตอร์จำนวน 5 โปรแกรม ได้แก่ Microsoft Excel, Scilab, Python, LINGO และ GLPK เพื่อทดสอบความถูกต้องและประสิทธิภาพของการหาคำตอบจากแบบจำลองเดียวกัน ผลการคำนวณจากทุกโปรแกรมให้คำตอบที่สอดคล้องกัน โดยพบว่าค่าที่เหมาะสมที่สุดคือ พนักงานการตลาดอาวุโสสามารถขายได้ 1200 แพ็คเกจ พนักงานการตลาดรุ่นเยาว์สามารถขายได้ 1200 แพ็คเกจ และนักศึกษาฝึกงานด้านการตลาดสามารถขายได้ 600 แพ็คเกจ ซึ่งทำให้ได้ยอดขายรวมสูงสุดภายใต้ข้อจำกัดที่กำหนด

ผลการศึกษาชี้ให้เห็นว่าการใช้เทคนิคกำหนดการเชิงเส้นร่วมกับโปรแกรมคอมพิวเตอร์หลายแพลตฟอร์มสามารถช่วยสนับสนุนการวางแผนทางการตลาดและการตัดสินใจขององค์กรได้อย่างมีประสิทธิภาพอย่างไรก็ตาม โปรแกรม Excel Solver จึงถือว่าเหมาะสมที่สุดเนื่องจาก เป็นปัญหาคำหนดการเชิงเส้นที่มีขนาดไม่ใหญ่มาก และต้องการความสะดวกในการใช้งานและการแสดงผลที่เข้าใจง่ายอีกทั้งยังสามารถนำไปประยุกต์ใช้ในงานจริงได้สะดวก โดยเฉพาะสำหรับ ผู้ใช้ที่ไม่มีพื้นฐานการเขียนโปรแกรมมากนัก

กิตติกรรมประกาศ

โครงการคณิตศาสตร์ เรื่อง การประเมินประสิทธิภาพของโปรแกรมแก้ปัญหาคำหนดการเชิงเส้น: กรณีศึกษาการกระจายพนักงานเพื่อเพิ่มยอดขายแฟกเกจอินเทอร์เน็ต สำเร็จลุล่วงได้ด้วยความรู้และความช่วยเหลือจากหลายฝ่าย ผู้จัดทำขอกราบขอบพระคุณอาจารย์ผู้สอนที่ได้ให้คำแนะนำ คำปรึกษา และข้อเสนอแนะที่เป็นประโยชน์ตลอดระยะเวลาการดำเนินงานวิจัย ตั้งแต่การกำหนดหัวข้อ การพัฒนาแบบจำลองกำหนดการเชิงเส้น ตลอดจนการตรวจสอบและปรับปรุงเนื้อหาให้มีความถูกต้องและสมบูรณ์มากยิ่งขึ้น

ขอขอบคุณผู้ที่ให้ความรู้และแหล่งข้อมูลต่าง ๆ ที่เกี่ยวข้องกับการประยุกต์ใช้วิธีซิมเพล็กซ์ รวมถึงการใช้โปรแกรมคอมพิวเตอร์ในการคำนวณ ได้แก่ Excel, Scilab, Python, LINGO และ GLPK ซึ่งมีส่วนสำคัญในการวิเคราะห์และเปรียบเทียบผลลัพธ์ของแบบจำลองที่ใช้ในงานวิจัยครั้งนี้

สุดท้ายนี้ ผู้จัดทำขอขอบคุณครอบครัวและเพื่อน ๆ ที่ให้กำลังใจ สนับสนุน และช่วยเหลือในด้านต่าง ๆ จนทำให้งานวิจัยฉบับนี้สำเร็จลุล่วงไปได้ด้วยดี หากมีข้อผิดพลาดประการใด ผู้จัดทำขอน้อมรับไว้และจะนำไปปรับปรุงแก้ไขในโอกาสต่อไป

1 บทนำ

1.1 ที่มาและความสำคัญของปัญหา

ในยุคที่การแข่งขันทางธุรกิจมีความเข้มข้นมากขึ้นทุกปี องค์กรต่าง ๆ จำเป็นต้องพัฒนากลยุทธ์ด้านการบริหารงานและการจัดสรรทรัพยากรอย่างมีประสิทธิภาพ เพื่อรักษาระดับผลกำไรและเพิ่มความสามารถในการแข่งขัน โดยเฉพาะอย่างยิ่งงานด้านการตลาด ซึ่งมีบทบาทสำคัญในการสร้างยอดขายและรายได้ให้กับบริษัท ดังนั้นความสามารถในการกำหนดเป้าหมายการขายและการจัดสรรบุคลากรอย่างเหมาะสมจึงเป็นสิ่งจำเป็นอย่างยิ่ง บริษัท PT. Bistion Ponsel ซึ่งเป็นผู้จำหน่ายแพ็คเกจอินเทอร์เน็ตในเขต Tembung ประเทศอินโดนีเซีย กำลังเผชิญกับปัญหาการจัดสรรพนักงานการตลาดจำนวน 20 คน ซึ่งแบ่งออกเป็นสามระดับ ได้แก่ พนักงานการตลาดอาวุโส พนักงานการตลาดรุ่นเยาว์ และนักศึกษาฝึกงานด้านการตลาด เพื่อให้สามารถบรรลุเป้าหมายยอดขายจำนวน 3,000 แพ็คเกจต่อเดือน ภายใต้ทรัพยากรที่มีอยู่อย่างจำกัด บริษัทจึงต้องการแนวทางในการบริหารจัดการที่สามารถช่วยกำหนดปริมาณยอดขายที่เหมาะสมสำหรับพนักงานแต่ละระดับ รวมทั้งควบคุมงบประมาณด้านค่าตอบแทนให้เกิดความคุ้มค่าสูงสุด การนำแนวคิดของโปรแกรมต่าง ๆ เช่น Excel Solver, Scilab, Python, GNU Linear Programming Kit (GLPK) และ LINGO มาประยุกต์ใช้ในการแก้ปัญหาที่กำหนดการเชิงเส้นด้วยวิธีซิมเพล็กซ์ ถือเป็นแนวทางสำคัญที่ช่วยให้บริษัทสามารถวิเคราะห์เงื่อนไขข้อจำกัดและตัวแปรที่เกี่ยวข้องได้อย่างเป็นระบบ ส่งผลให้สามารถค้นหาวิธีการจัดสรรภาระงานที่ให้ผลลัพธ์ที่ดีที่สุด ทั้งในด้านยอดขายและต้นทุนค่าตอบแทน โปรแกรมเหล่านี้ได้รับการยอมรับอย่างแพร่หลายในงานวิจัยด้านการจัดสรรทรัพยากร เช่น การวางแผนการผลิต การวางแผนงาน หรือการบริหารงานขาย

ในปี ค.ศ. 2013 Jared L. Gearhart, Kristin L. Adair, Richard J. Detry, Justin D. Durfee, Katherine A. Jones และ Nathaniel Martin ได้ทำการศึกษาเกี่ยวกับการเปรียบเทียบตัวแก้ปัญหาคำหนดการเชิงเส้นแบบโอเพนซอร์ส โดยมีวัตถุประสงค์เพื่อสำรวจและประเมินว่ามีตัวแก้ปัญหาคำหนดการเชิงเส้น (Linear Programming: LP) แบบโอเพนซอร์สที่สามารถใช้งานได้จริงหรือไม่ เพื่อใช้ทดแทนตัวแก้ปัญหาคณิตศาสตร์เชิงพีชคณิตที่มีข้อจำกัดด้านงบประมาณหรือใบอนุญาต งานวิจัยได้ทำการทดสอบตัวแก้ปัญหาลินีอาร์โปรแกรมมิง (LP) ได้แก่ GNU Linear Programming Kit (GLPK) และ lp solve ผลการศึกษาพบว่า โปรแกรมโอเพนซอร์สสามารถเป็นทางเลือกที่ดีสำหรับผู้ใช้ที่มีข้อจำกัดด้านงบประมาณ และสามารถใช้งานได้ดีในงานทั่วไปหรือการเรียนการสอน อย่างไรก็ตาม ในกรณีของงานระดับอุตสาหกรรมที่ต้องการความเร็วในการประมวลผลสูงและสามารถแก้ปัญหาขนาดใหญ่ได้อย่างมีประสิทธิภาพ โปรแกรมเชิงพีชคณิตยังคงมีความได้เปรียบมากกว่า

ในปี ค.ศ. 2014 Vidya M. Ayer, Sheila Miguez และ Brian H. Toby ได้ศึกษาเปรียบเทียบแนวทางการเขียนโปรแกรมสามประเภท โดยเน้นถึงเหตุผลที่ภาษาแบบสคริปต์ (Scripting Languages) เช่น Python เป็นที่นิยมในหมู่นักวิทยาศาสตร์ที่ไม่เชี่ยวชาญด้านการเขียนโปรแกรม นอกจากนี้ยังแนะนำแพ็คเกจเสริมของภาษา Python ที่มีประโยชน์สำหรับการคำนวณเชิงตัวเลข การสร้างกราฟทางวิทยาศาสตร์ และการพัฒนาส่วนติดต่อผู้ใช้ (GUI) อีกทั้งยังมีแหล่งข้อมูลการเรียนรู้จำนวนมากซึ่งช่วยให้งานต่าง ๆ สามารถดำเนินการได้สำเร็จด้วยโค้ดที่สั้นกว่าเมื่อเปรียบ

เทียบกับภาษาอื่น และยังสามารถประมวลผลการคำนวณที่ซับซ้อนได้ด้วยความเร็วที่ใกล้เคียงกับภาษา Fortran หรือ C โดยใช้เทคนิคการคำนวณแบบเมทริกซ์แทนการวนซ้ำ ในปี ค.ศ. 2020 Haroldo G. Santos และ Túlio A. M. Toffolo ได้พัฒนาแพ็คเกจที่ชื่อว่า Python-MIP ซึ่งเป็นเครื่องมือสำหรับสร้างแบบจำลองและแก้ปัญหาการกำหนดการเชิงเส้นจำนวนเต็มผสม (Mixed Integer Linear Programming: MILP) ในภาษา Python เครื่องมือนี้สามารถช่วยค้นหาคำตอบที่เหมาะสมที่สุดภายใต้ข้อจำกัดต่าง ๆ และสามารถนำไปประยุกต์ใช้กับปัญหาหลากหลายประเภท เช่น การจัดสรรทรัพยากร การจัดตารางเวลา และการวางแผนเส้นทางที่มีความซับซ้อน

ในปี ค.ศ. 2023 Mulkan Azhari, Syahril Efendi, Purwanto และ Halim Maulana ได้ศึกษาการแก้ปัญหาการกำหนดโปรแกรมเชิงเส้นเพื่อค้นหาแนวทางในการเพิ่มยอดขายแพ็คเกจอินเทอร์เน็ตให้ได้สูงสุด โดยยังคงรักษาค่าใช้จ่ายด้านการตลาดของบริษัทให้อยู่ในระดับที่เหมาะสม งานวิจัยนี้ใช้วิธีซิมเพล็กซ์และใช้โปรแกรม Solver ของ Microsoft Excel ในการคำนวณหาคำตอบที่เหมาะสมที่สุด ผลการศึกษาพบว่า โปรแกรมสามารถให้คำตอบที่มีประสิทธิภาพและสามารถนำไปใช้ได้จริง ช่วยให้บริษัทเห็นภาพการจัดสรรงานของพนักงานแต่ละกลุ่มได้อย่างชัดเจน เช่น ควรเพิ่มภาระงานให้กับกลุ่มใด หรือควรลดภาระงานจากกลุ่มใดเพื่อลดต้นทุน ส่งผลให้บริษัทสามารถวางแผนการจัดสรรบุคลากรและงบประมาณได้อย่างเหมาะสม

นอกจากนี้ ยังมีงานวิจัยที่พัฒนาสภาพแวดล้อมแบบโต้ตอบโดยใช้โปรแกรม Scilab ร่วมกับกล่องเครื่องมือ Metanet เพื่อใช้ในการแก้ปัญหาการปรับให้เหมาะสมเชิงเส้นขนาดใหญ่และแบบสแปร์ส โดยมีการรวมตัวแก้ปัญหาเชิงเส้นขนาดใหญ่จำนวนสามตัว ได้แก่ Lipsol, Hopdm และ lp solve เข้าไว้ในระบบ Scilab งานวิจัยดังกล่าวอธิบายถึงวิธีการนำโค้ดเหล่านี้มาใช้งานร่วมกับ Scilab และมีการเปรียบเทียบประสิทธิภาพของแต่ละตัวแก้ปัญหา ผลการศึกษาพบว่าสภาพแวดล้อมดังกล่าวช่วยให้สามารถจัดการปัญหาการปรับให้เหมาะสมเชิงเส้นขนาดใหญ่ได้อย่างมีประสิทธิภาพ และสามารถนำไปประยุกต์ใช้กับปัญหาการไหลของหลายสินค้า (Multicommodity Problems) ได้ รวมทั้งยังมีระบบการนำเข้าข้อมูลและการแสดงผลที่เป็นมิตรต่อผู้ใช้งานอินเทอร์เน็ตแบบกราฟิก

ในปี ค.ศ. 2025 Arif Agung Nugroho, Sahana Pramukti และ Ratih Setyaningrum ได้ทำการศึกษาโดยใช้วิธี Integer Linear Programming ได้แก่ วิธี Simplex และ Branch and Bound พร้อมทั้งเปรียบเทียบผลลัพธ์กับซอฟต์แวร์ LINGO ผลการศึกษาพบว่า การใช้ซอฟต์แวร์ LINGO สามารถให้ผลลัพธ์การเพิ่มประสิทธิภาพที่ดีที่สุด โดยได้กำไรสูงสุดจำนวน Rp. 85,000 จากการผลิตเปมเป็กคาปาลเซล์จำนวน 14 ชิ้น และเปมเป็กเลนเจอร์จำนวน 0 ชิ้น ซึ่งช่วยให้สามารถจัดสรรวัตถุดิบที่มีอยู่อย่างจำกัดได้อย่างเหมาะสมและสร้างผลกำไรสูงสุดให้กับธุรกิจ UMKM เปมเป็กจิก ลิน

ดังนั้น จากปัญหาและงานวิจัยที่กล่าวมาข้างต้น จึงสามารถสรุปได้ว่าการพัฒนาแบบจำลองกำหนดการเชิงเส้นและการประยุกต์ใช้เครื่องมือต่าง ๆ เพื่อแก้ปัญหาการจัดสรรทรัพยากร สามารถช่วยเพิ่มประสิทธิภาพในการวางแผนงานได้อย่างมีประสิทธิภาพ งานวิจัยนี้จึงมีวัตถุประสงค์เพื่อพัฒนาแบบจำลองกำหนดการเชิงเส้นสำหรับการจัดสรรพนักงานการตลาดให้เหมาะสมที่สุด เพื่อเพิ่มยอดขายแพ็คเกจอินเทอร์เน็ตของบริษัท ภายใต้ข้อจำกัดด้านงบประมาณและทรัพยากรที่มีอยู่ พร้อมทั้งเปรียบเทียบประสิทธิภาพของเครื่องมือแต่ละชนิดในการแก้ปัญหาในสถานการณ์จริง

1.2 วัตถุประสงค์ของการศึกษา

1. เพื่อเปรียบเทียบประสิทธิภาพของเครื่องมือคำนวณและวิเคราะห์ 5 ชนิด ได้แก่ Excel Solver, Scilab, Python, GNU Linear Programming Kit (GLPK) และ LINGO ในการแก้ปัญหการจัดสรรทรัพยากรและค้นหาคำตอบที่เหมาะสมที่สุด
2. เพื่อประเมินความเหมาะสม ความแม่นยำ และความสะดวกในการใช้งานของแต่ละเครื่องมือในการประยุกต์ใช้กับกรณีศึกษา

1.3 ประโยชน์ที่คาดว่าจะได้รับ

1. ได้ข้อมูลเปรียบเทียบประสิทธิภาพของเครื่องมือที่ใช้ในการคำนวณและแก้ปัญหาเชิงตัวเลขจำนวน 5 ชนิด ได้แก่ Excel Solver, Scilab, Python, GLPK และ LINGO
2. ผู้ศึกษาจะได้รับความรู้และทักษะในการสร้างแบบจำลองทางคณิตศาสตร์ และการประยุกต์ใช้ซอฟต์แวร์ด้านการหาค่าเหมาะที่สุด (Optimization) เพื่อแก้ปัญหาคัดสินใจในสถานการณ์จริง
3. ผลลัพธ์สำหรับไปประยุกต์ใช้กับงานวางแผนทรัพยากร การบริหารงานขาย และการจัดสรรทรัพยากรในหน่วยงานอื่น ๆ ได้ในอนาคต
4. ได้ศึกษาประสิทธิภาพของโปรแกรมแก้ปัญหาคำหนดการเชิงเส้น

1.4 ขอบเขตในการศึกษา

1. การศึกษานี้เน้นการวิเคราะห์กรณีของบริษัท PT. Bistion Ponsel ในการจัดสรรยอดขายรวม 3,000 แพ็กเกจอินเทอร์เน็ตต่อเดือน
2. การวิเคราะห์จำกัดอยู่ที่เครื่องมือด้าน Optimization 5 ชนิด คือ Excel Solver, Scilab, Python, GNU Linear Programming Kit (GLPK) และ LINGO
3. การศึกษามุ่งเน้นไปที่การหาคำตอบที่เหมาะสมที่สุดเท่านั้นไม่ได้วิเคราะห์ด้านพฤติกรรมพนักงานหรือคุณภาพงานขาย

1.5 สมมติฐานการศึกษา

มีเครื่องมืออย่างน้อย 1 โปรแกรมที่มีประสิทธิภาพในการแก้ปัญหาคำหนดการเชิงเส้น เครื่องมือที่ใช้ในการวิจัย มีดังนี้ Microsoft Excel Solver, Scilab, Python, GNU Linear Programming Kit (GLPK), LINGO, คอมพิวเตอร์

2 ความรู้พื้นฐาน

2.1 กำหนดการเชิงเส้น (Linear Programming; LP) [1],[10]

กำหนดการเชิงเส้น (Linear Programming) เป็นเทคนิคทางคณิตศาสตร์ที่มุ่งแก้ปัญหาการจัดสรรทรัพยากรที่มีอยู่อย่างจำกัดให้เกิดประสิทธิภาพสูงสุด โดยคำตอบของปัญหาจะอยู่ในรูปของค่าตัวแปรตัดสินใจซึ่งต้องอยู่ภายใต้เงื่อนไขข้อจำกัดที่กำหนด และเป็นคำตอบที่เหมาะสมที่สุดตามวัตถุประสงค์ของปัญหา ทั้งนี้วัตถุประสงค์อาจเป็นการเพิ่มกำไรหรือผลผลิตให้มากที่สุด หรือการลดต้นทุนให้ต่ำที่สุด ในขณะที่ยังสามารถตอบสนองความต้องการและใช้ทรัพยากรที่มีอยู่อย่างเพียงพอ

ปัญหาคำหนดการเชิงเส้นสามารถนำไปประยุกต์ใช้ได้หลายด้าน เช่น ปัญหาการวางแผนการผลิต ซึ่งอาจมีข้อจำกัดด้านต้นทุน วัตถุดิบ หรือกำลังการผลิต ตัวอย่างเช่น การผลิตสินค้าที่ต้องใช้วัตถุดิบหลายชนิดหรือมีหลายขั้นตอนในการผลิต นอกจากนี้ยังรวมถึงปัญหาการวางแผนด้านการตลาด เช่น การตัดสินใจเลือกปริมาณการผลิตและการจำหน่ายสินค้าให้สอดคล้องกับความต้องการของตลาดและต้นทุนสินค้า นอกจากนี้ กำหนดการเชิงเส้นยังถูกนำมาใช้ในการจัดการการลงทุน เช่น การบริหารเงินลงทุนที่มีอยู่อย่างจำกัดเพื่อการลงทุนในกิจกรรมต่าง ๆ ไม่ว่าจะเป็นการลงทุนในหุ้น การฝากเงินในธนาคาร หรือการสำรองเงินลงทุน เพื่อให้ได้ผลกำไรสูงสุด ภายใต้ข้อจำกัดที่กำหนด ปัญหาดังกล่าวล้วนมีลักษณะร่วมกันคือการแสวงหาวิธีจัดสรรทรัพยากรที่เหมาะสมที่สุด

องค์ประกอบของปัญหาคำหนดการเชิงเส้น

1. กำหนดตัวแปรตัดสินใจ (Decision Variables) คือ ตัวแปรที่ต้องพิจารณาเพื่อให้ได้คำตอบที่ดีที่สุด โดยแทนด้วย

$$x_1, x_2, \dots, x_n$$

2. ฟังก์ชันวัตถุประสงค์ (Objective Function) คือ ฟังก์ชันที่ใช้กำหนดเป้าหมายของปัญหา เช่น การหาค่าสูงสุด (Maximization) หรือการหาค่าต่ำสุด (Minimization) กรณีต้องการหาค่าสูงสุด

$$\text{Maximize } Z = c_1x_1 + c_2x_2 + \dots + c_nx_n$$

กรณีต้องการหาค่าต่ำสุด

$$\text{Minimize } Z = c_1x_1 + c_2x_2 + \dots + c_nx_n$$

เมื่อ $c_1, c_2, \dots, c_n$ เป็นสัมประสิทธิ์ของตัวแปร $x_1, x_2, \dots, x_n$ ตามลำดับ

3. เงื่อนไขบังคับ (Constraints) คือ ขอบเขตหรือข้อจำกัดของตัวแปรตัดสินใจ ซึ่งสามารถเขียนได้ทั้งในรูปสมการหรืออสมการ โดยใช้เครื่องหมาย $\leq$, $\geq$ หรือ $=$

4. บริเวณที่เป็นไปได้ (Feasible Region) คือ กลุ่มของคำตอบที่สอดคล้องกับเงื่อนไขข้อจำกัดทั้งหมดของปัญหาซึ่งเกิดจากการอินเตอร์เซกชัน (Intersection) ของข้อจำกัด (Constraints) ทั้งหมด โดยพิจารณาเฉพาะค่าที่ไม่เป็นลบรวมทั้งศูนย์

ตัวอย่างตัวแบบกำหนดการเชิงเส้น เช่น

ฟังก์ชันวัตถุประสงค์ คือ Maximize $Z = c_1x_1 + c_2x_2 + \cdots + c_nx_n$

เงื่อนไขบังคับ คือ $a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n \leq b_1$

$$a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n \leq b_2$$

$$a_{31}x_1 + a_{32}x_2 + \cdots + a_{3n}x_n \geq b_3$$

$$a_{41}x_1 + a_{42}x_2 + \cdots + a_{4n}x_n = b_4$$

$$\vdots$$

$$a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n \leq b_m$$

$$x_i \geq 0, i = 1, 2, \dots, n$$

รูปแบบมาตรฐานของกำหนดการเชิงเส้น

จากตัวแบบกำหนดข้างต้น สามารถเขียนให้อยู่ในรูปแบบมาตรฐานได้ดังนี้

$$\text{Maximize } Z = c_1x_1 + c_2x_2 + \cdots + c_nx_n + 0s_1 + 0s_2 + \cdots + 0s_m + 0a_1 + 0a_2 + \cdots$$

$$a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n + s_1 = b_1$$

$$a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n + s_2 = b_2$$

$$a_{31}x_1 + a_{32}x_2 + \cdots + a_{3n}x_n - s_3 + a_1 = b_2$$

$$a_{41}x_1 + a_{42}x_2 + \cdots + a_{4n}x_n + a_2 = b_3$$

$$\vdots$$

$$a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n + s_m = b_m$$

$$x_i \geq 0, i = 1, 2, \dots, n$$

เมื่อ

$c_1, c_2, \dots, c_n$	คือ สัมประสิทธิ์ของตัวแปรตัดสินใจ ($x_1, x_2, \dots, x_n$)
$x_1, x_2, \dots, x_n$	คือ ตัวแปรตัดสินใจที่มีค่าเป็นบวก
$a_{m1}, a_{m2}, \dots, a_{mn}$	คือ สัมประสิทธิ์ของตัวแปรตัดสินใจ
$b_1, b_2, \dots, b_m$	คือ ค่าคงที่ทางขวามือ
$s_1, s_2, \dots, s_m$	คือ ตัวแปรเสริม ใช้เปลี่ยนเครื่องหมาย $\leq$ ให้เป็นสมการ
$a_1, a_2, \dots$	คือ ตัวแปรเทียมใช้ช่วยให้มี Basic feasible solution เริ่มต้น

การแก้ปัญหากำหนดการเชิงเส้น เป็นการหาค่าของตัวแปรตัดสินใจ (Decision Variable) ที่ทำให้ค่าของฟังก์ชันวัตถุประสงค์ (Objective Function) เป็นค่าที่ดีที่สุด โดยคำตอบที่ได้ต้องอยู่ภายใต้เงื่อนไขขอบข่ายของปัญหา (Constraints) ที่ถูกกำหนด เนื้อหาในบทนี้กล่าวถึงวิธีการแก้ปัญหาด้วยกัน วิธีซิมเพล็กซ์ (Simplex Method) ดังนี้

2.2 วิธีซิมเพล็กซ์ [3],[12]

วิธีซิมเพล็กซ์อาศัยการคำนวณหาคำตอบที่ดีที่สุดจากตารางซิมเพล็กซ์ สามารถแก้ปัญหากำหนดการเชิงเส้นที่มีจำนวนตัวแปรตัดสินใจตั้งแต่ 2 ตัวขึ้นไป

หลักการ

1. กำหนดปัญหากำหนดการเชิงเส้น
2. แปลงปัญหาให้อยู่ในรูปมาตรฐาน (แสดงข้อจำกัดทุกข้อในรูปสมการเส้นตรง โดยอาจต้องเพิ่มตัวแปรเพื่อเปลี่ยนจากอสมการ ($\leq$ และ $\geq$) ให้เป็นสมการ)
3. สร้างตารางซิมเพล็กซ์เพื่อเก็บค่าตัวแปรฟังก์ชันวัตถุประสงค์และข้อจำกัด
4. ทำการดำเนินการตามแถวเพื่อหาตัวแปรเข้า และตัวแปรออก ที่ทำให้ได้คำตอบที่ดีที่สุดในแต่ละรอบของการคำนวณ โดยกระบวนการนี้เป็นหัวใจสำคัญของ วิธีซิมเพล็กซ์ (Simplex Method) ซึ่งใช้ในการแก้ปัญหาโปรแกรมเชิงเส้น (Linear Programming) การเลือก Entering Variable (ตัวแปรเข้า) จะพิจารณาจากแถว Z ซึ่งเป็นแถวของ ฟังก์ชันวัตถุประสงค์ (Objective Function) โดยในกรณีที่ปัญหาแบบ Maximize จะเลือกตัวแปรที่มีค่าสัมประสิทธิ์ติดลบมากที่สุด เนื่องจากตัวแปรนั้นมีศักยภาพในการเพิ่มค่าของฟังก์ชันวัตถุประสงค์ได้มากที่สุด คอลัมน์ของตัวแปรที่เลือกนี้เรียกว่า Pivot Column หลังจากนั้นจึงทำการเลือก Leaving Variable (ตัวแปรออก) เพื่อกำหนดว่าตัวแปรใดจะออกจากฐาน (Basis) โดยคำนวณจากการนำค่าในช่อง RHS (Right Hand Side) ของแต่ละแถว มาหารด้วยค่าที่อยู่ใน Pivot Column ของแถวนั้น ๆ โดยจะพิจารณาเฉพาะค่าที่เป็นบวกมากกว่าศูนย์เท่านั้น เพื่อให้ยังคงเงื่อนไขของคำตอบที่เป็นไปได้ (Feasible Solution) แถวที่ให้ผลหาร น้อยที่สุด จะถูกเลือกเป็นแถวหลักในการดำเนินการ เรียกว่า Pivot Row และตัวแปรพื้นฐานในแถวนั้นจะเป็นตัวแปรที่ต้องออกจากฐาน
5. ทำกระบวนการดังกล่าวไปเรื่อย ๆ โดยการทำ Pivot Operation เพื่อปรับตารางซิมเพล็กซ์ในแต่ละรอบ จนกระทั่งไม่สามารถปรับปรุงค่าของฟังก์ชันวัตถุประสงค์ให้ดีขึ้นได้อีก การตรวจสอบว่าพบคำตอบที่เหมาะสมที่สุดแล้วหรือยัง สามารถทำได้โดยกลับไปดูที่แถว Z อีกครั้ง หากค่าสัมประสิทธิ์ของตัวแปรตัดสินใจ (Decision Variables) ในแถวนี้นี้ไม่มีค่าติดลบเหลืออยู่เลย (ในกรณีของปัญหา Maximize) แสดงว่าการเพิ่มตัวแปรใด ๆ เข้าสู่ฐานจะไม่ทำให้ค่าฟังก์ชันวัตถุประสงค์ดีขึ้นอีกต่อไป ดังนั้นคำตอบที่ได้ในขณะนั้นจึงถือว่าเป็น คำตอบที่เหมาะสมที่สุด (Optimal Solution) ของปัญหา

ตัวอย่างการแก้ปัญหากำหนดการเชิงเส้นด้วยวิธีซิมเพล็กซ์

1. กำหนดฟังก์ชันวัตถุประสงค์ของปัญหาเพื่อหาค่าสูงสุดของกำไร โดยให้ x_1 และ x_2 เป็นตัวแปรตัดสินใจ ดังนั้นฟังก์ชันสามารถเขียนได้ดังนี้

$$\text{Maximize } Z = 3x_1 + 5x_2$$

โดยที่ Z แทนค่าผลลัพธ์ของฟังก์ชันวัตถุประสงค์ที่ต้องการหาค่าสูงสุด

2. กำหนดเงื่อนไขบังคับของปัญหา ซึ่งเป็นข้อจำกัดของทรัพยากรที่มีอยู่ โดยเขียนเป็นสมการอสมการได้ดังนี้

$$x_1 + 2x_2 \leq 8$$

$$3x_1 + 2x_2 = 12$$

$$x_1, x_2 \geq 0$$

3. เขียนให้อยู่ในรูปแบบมาตรฐานของฟังก์ชันวัตถุประสงค์ เพื่อให้สามารถแก้ปัญหาด้วยซิมเพล็กซ์ได้ จำเป็นต้องแปลงอสมการให้อยู่ในรูปสมการ โดยเพิ่ม ตัวแปร s_1 และ s_2 เข้าไปในสมการ ดังนั้นจะได้

$$\text{Maximize } Z = 3x_1 + 5x_2 + 0s_1 + 0a_1$$

และเงื่อนไขบังคับจะได้

$$x_1 + 2x_2 + s_1 = 8$$

$$3x_1 + 2x_2 + a_1 = 12$$

$$x_1, x_2 \geq 0$$

4. สร้างตารางซิมเพล็กซ์เริ่มต้น (Initial Simplex Tableau) เพื่อใช้ในการคำนวณหาคำตอบที่เหมาะสมที่สุดของปัญหา โดยตารางจะประกอบด้วยตัวแปรพื้นฐาน ตัวแปรตัดสินใจ ตัวแปรเสริม และค่าด้านขวาของสมการ

C_B	C_j	-3	-5	0	0	X_B	อัตราส่วน
	X_B	x_1	x_2	s_1	a_1		X_B/x_2
0	s_1	1	2	1	0	8	4
0	a_1	3	2	0	1	12	6
Z_j		0	0	0	0		
$Z_j - C_j$		3	5	0	0		

จากนั้นคำนวณค่า

$$Z_j - C_j$$

ได้ค่า (3, 5, 0, 0) โดยเลือกค่ามากที่สุดคือ 5 ที่คอลัมน์ x_2 ดังนั้น x_2 จะเป็น ตัวแปรเข้า
คำนวณอัตราส่วน

$$\frac{X_B}{x_2}$$

ได้ค่า

$$\frac{8}{2} = 4, \quad \frac{12}{2} = 6$$

เลือกค่าที่น้อยที่สุดคือ 4 ดังนั้น s_1 จะออกจากฐาน และตำแหน่ง Pivot คือ 2

C_i	C_j	-3	-5	0	0	X_B	อัตราส่วน
	X_B	x_1	x_2	s_1	a_1		X_B/x_1
-5	x_2	1/2	1	1/2	0	4	8
0	a_1	2	0	-1	1	4	2
Z_j		-2.5	-5	-2.5	0		
$Z_j - C_j$		1/2	0	-2.5	0		

หลังจากทำ Pivot ที่ค่า 2 จะได้ตัวแปรฐานใหม่คือ x_2 แทน s_1 แถวที่ 1 ถูกหารด้วย 2 เพื่อให้ Pivot เท่ากับ 1 และใช้กำจัดค่าในคอลัมน์ x_2 ของแถวอื่นให้เป็น 0
จากนั้นคำนวณ

$$Z_j - C_j$$

ได้ค่า

$$(0.5, 0, -2.5, 0)$$

ค่าที่เป็นบวกมากที่สุดคือ 0.5 ที่คอลัมน์ x_1 ดังนั้น x_1 เป็น ตัวแปรเข้า
คำนวณอัตราส่วน

$$\frac{4}{0.5} = 8, \quad \frac{4}{2} = 2$$

เลือกค่าที่น้อยที่สุดคือ 2 ดังนั้น a_1 ออกจากฐาน และ Pivot คือ 2

C_i	C_j	-3	-5	0	0	X_B
	X_B	x_1	x_2	s_1	a_1	
-5	x_2	0	1	0.75	-1/4	3
-3	x_1	1	0	-0.5	0.5	2
Z_j		-3	-5	-2.25	-0.25	21
$Z_j - C_j$		0	0	-2.25	-0.25	

ทำ Pivot ที่ค่า 2 ในคอลัมน์ x_1 ทำให้ตัวแปรฐานใหม่คือ x_1 และ x_2
 หลังปรับตารางแล้ว คำนวณค่า Z_j

$$Z_j = (-3, -5, -2.25, -0.25)$$

คำนวณ

$$Z_j - C_j$$

ได้ค่า

$$(0, 0, -2.25, -0.25)$$

เนื่องจากไม่มีค่าบวกในแถว $Z_j - C_j$ จึงสรุปได้ว่าได้คำตอบที่เหมาะสมที่สุดแล้ว
 ค่าคำตอบที่เหมาะสมที่สุดคือ

$$x_1 = 2, \quad x_2 = 3$$

และค่าฟังก์ชันวัตถุประสงค์คือ

$$Z = 21$$

เนื่องจากไม่มีค่าสัมประสิทธิ์บวกมากที่สุดในแถว $Z_j - C_j$ แล้ว จึงได้คำตอบที่เหมาะสมที่สุด
 (Optimal Solution):

คำตอบสุดท้าย: $x_1 = 2, \quad x_2 = 3, \quad Z_{\max} = 21$

2.3 โปรแกรมที่ใช้ในการแก้ปัญหากำหนดการแข่งขัน [2],[4]

2.3.1 Microsoft Excel Solver [2],[4]

Microsoft Excel Solver เป็นเครื่องมือเสริม (Add-in) ที่ทำงานร่วมกับโปรแกรม Microsoft Excel มีหน้าที่ช่วยในการแก้ปัญหาการหาค่าที่เหมาะสมที่สุด (Optimization) โดยเริ่มมีการพัฒนาและนำมาใช้งานตั้งแต่ช่วงปลายทศวรรษที่ 1980 พร้อมกับการพัฒนาโปรแกรม Microsoft Excel ในระยะแรกโปรแกรม Solver ถูกพัฒนาโดยบริษัท Frontline Systems และ Microsoft ได้นำมาใช้เป็นส่วนเสริมของ Excel เพื่อช่วยให้ผู้ใช้งานสามารถแก้ปัญหาทางคณิตศาสตร์ ธุรกิจ วิศวกรรม และการวิจัย ได้อย่างสะดวก โดยไม่จำเป็นต้องเขียนโปรแกรมที่ซับซ้อนในปัจจุบัน Microsoft Excel Solver ได้รับการพัฒนาอย่างต่อเนื่อง และเป็น Add-in มาตรฐานที่สามารถใช้งานได้ทั้งใน Excel หลายเวอร์ชัน ทั้งระบบปฏิบัติการ Windows และ macOS

ความสามารถของโปรแกรม Microsoft Excel Solver

Microsoft Excel Solver สามารถใช้ในการหาค่าสูงสุด (Maximize) ค่าต่ำสุด (Minimize) หรือค่าที่กำหนดให้เท่ากับค่าหนึ่ง (Value Of) ของเซลล์เป้าหมาย โดยอาศัยการปรับค่าของตัวแปรที่ผู้ใช้งานกำหนด ตัวอย่างการใช้งาน เช่น การหาค่าไรสูงสุด การลดต้นทุนการผลิต และการจัดสรรทรัพยากรให้เกิดประสิทธิภาพสูงสุด Solver สามารถกำหนดเงื่อนไขบังคับ (Constraints) ให้กับตัวแปรได้หลายรูปแบบ เช่น ค่ามากกว่า น้อยกว่า หรือเท่ากับค่าที่กำหนด รวมถึงการกำหนดให้ตัวแปรเป็นจำนวนเต็ม หรือเป็นค่าฐานสอง (0 และ 1)

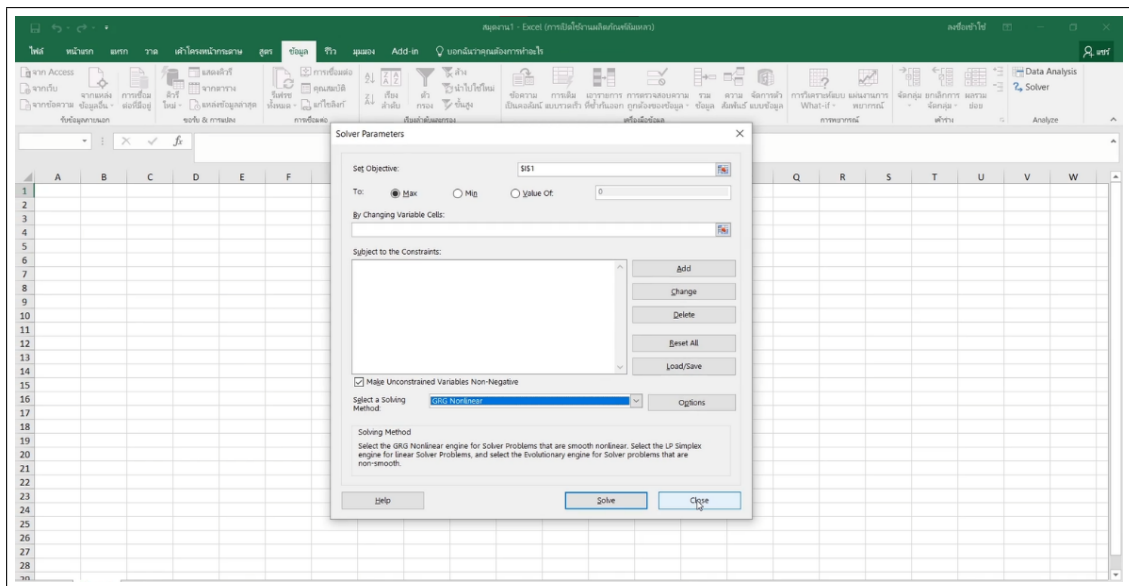
Microsoft Excel Solver รองรับการแก้ปัญหาหลายประเภท ได้แก่

- ปัญหาเชิงเส้น (Linear Programming)
- ปัญหาไม่เชิงเส้น (Nonlinear Programming)
- ปัญหาจำนวนเต็ม (Integer Programming)

Solver มีวิธีการคำนวณที่หลากหลาย เช่น

- Simplex LP สำหรับปัญหาเชิงเส้น
- GRG Nonlinear สำหรับปัญหาไม่เชิงเส้น
- Evolutionary สำหรับปัญหาที่มีความซับซ้อนสูง

Microsoft Excel Solver เป็นเครื่องมือที่ใช้งานง่าย ผู้ใช้สามารถกำหนดสูตรในเซลล์ Excel ได้ตามปกติ เลือกเซลล์เป้าหมาย กำหนดตัวแปรและเงื่อนไข จากนั้นให้โปรแกรมคำนวณคำตอบโดยอัตโนมัติ จึงเหมาะสำหรับนักเรียน นักศึกษา และผู้เริ่มต้นใช้งาน Microsoft Excel Solver สามารถนำไปใช้ในหลายด้าน เช่น การวางแผนการผลิต การวิเคราะห์ด้านการเงิน การจัดตารางเวลา การคำนวณต้นทุน และการทำโครงการหรือรายงานทางวิชาการ



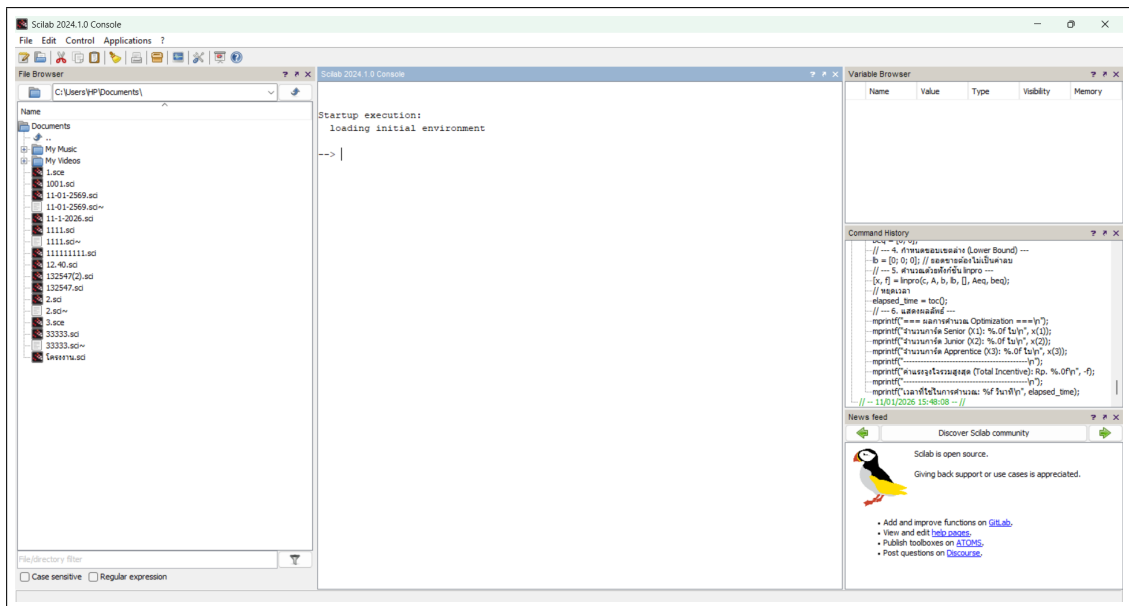
รูปที่ 1: หน้าหลักของโปรแกรม Microsoft Excel Solver

2.3.2 Scilab [2]

Scilab เป็นซอฟต์แวร์สำหรับการคำนวณเชิงตัวเลขและการวิเคราะห์ทางวิศวกรรม เริ่มพัฒนาขึ้นในปี ค.ศ. 1990 โดยนักวิจัยจากสถาบัน INRIA (Institut National de Recherche en Informatique et en Automatique) ประเทศฝรั่งเศส โดยมีวัตถุประสงค์เพื่อเป็นเครื่องมือด้านวิทยาศาสตร์ที่สามารถใช้งานได้ฟรี และเปิดเผยซอร์สโค้ดให้ผู้ใช้สามารถเข้าถึงได้ ต่อมา Scilab ได้รับการพัฒนาอย่างต่อเนื่อง และดูแลโดย Scilab Enterprises ปัจจุบัน Scilab เป็นซอฟต์แวร์แบบโอเพนซอร์ส สามารถใช้งานได้บนหลายระบบปฏิบัติการ เช่น Windows, Linux และ macOS และถูกนำไปใช้อย่างแพร่หลายในด้านการศึกษา การวิจัย และงานด้านวิศวกรรม

ความสามารถของโปรแกรม Scilab

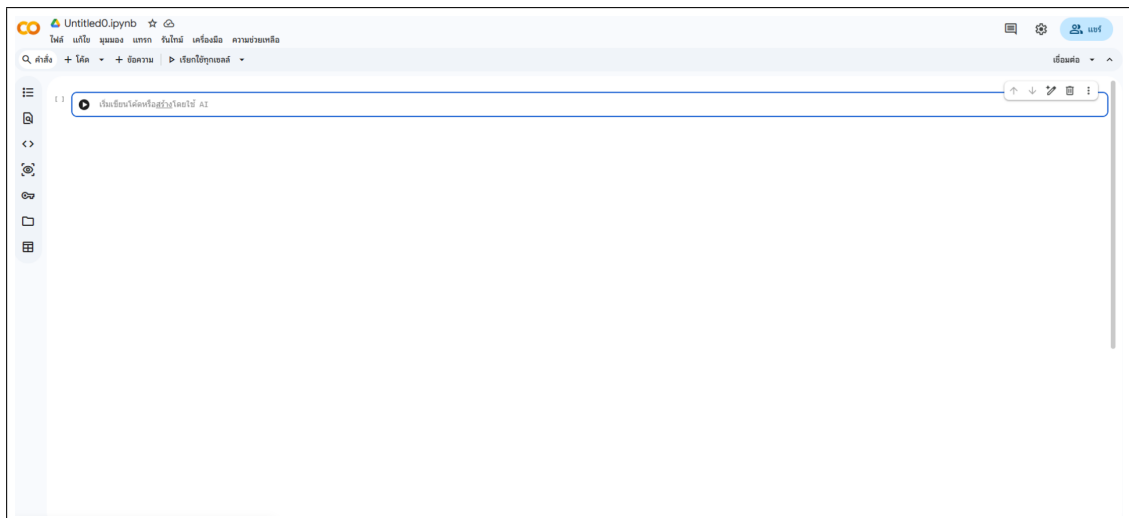
Scilab สามารถใช้ในการคำนวณเมทริกซ์ สมการเชิงเส้นและไม่เชิงเส้นรวมถึงการคำนวณอินทิกรัลอนุพันธ์และการคำนวณเชิงตัวเลขที่มีความซับซ้อน Scilab มีภาษาการเขียนโปรแกรมเฉพาะ ซึ่งมีรูปแบบคำสั่งคล้ายกับ MATLAB ผู้ใช้สามารถเขียนสคริปต์และสร้างฟังก์ชัน เพื่อแก้ปัญหาทางวิทยาศาสตร์และวิศวกรรมได้ Scilab สามารถสร้างกราฟสองมิติและสามมิติ เพื่อใช้ในการวิเคราะห์และนำเสนอข้อมูล ทำให้เข้าใจผลลัพธ์ได้อย่างชัดเจน Scilab ถูกนำไปใช้ในงานด้านวิศวกรรมไฟฟ้า วิศวกรรมเครื่องกล การควบคุมอัตโนมัติ การประมวลผลสัญญาณ รวมถึงการเรียนการสอนและงานวิจัยทางวิทยาศาสตร์



รูปที่ 2: หน้าหลักของโปรแกรม Scilab

2.3.3 Python [4]

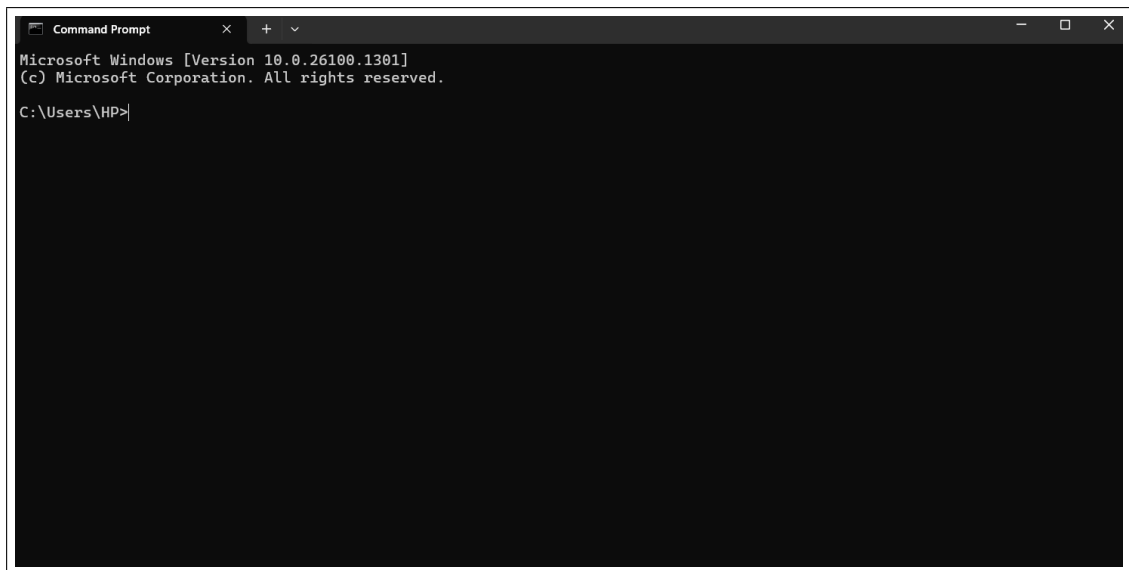
Python เป็นภาษาโปรแกรมระดับสูง พัฒนาโดย Guido van Rossum ในปี ค.ศ. 1991 โดยมีวัตถุประสงค์เพื่อสร้างภาษาที่อ่านง่าย มีโครงสร้างไม่ซับซ้อน และช่วยให้ผู้พัฒนาสามารถเขียนโปรแกรมได้อย่างรวดเร็ว ชื่อภาษา Python ได้รับแรงบันดาลใจมาจากรายการตลกของอังกฤษ ชื่อ Monty Python's Flying Circus ซึ่งสะท้อนแนวคิดของภาษาที่เน้นความเรียบง่าย และเป็นมิตรกับผู้ใช้ ปัจจุบัน Python เป็นภาษาโปรแกรมแบบโอเพนซอร์ส ได้รับความนิยมอย่างแพร่หลาย ในด้านการศึกษา การพัฒนาซอฟต์แวร์ งานด้านวิทยาศาสตร์ข้อมูล และงานวิจัยต่าง ๆ Python มีไวยากรณ์ที่เข้าใจง่าย โค้ดมีความกระชับ ทำให้เหมาะสำหรับผู้เริ่มต้นเรียนเขียนโปรแกรม รวมถึงการนำไปใช้ในงานโครงการทางการศึกษา Python รองรับการเขียนโปรแกรมหลายรูปแบบ เช่น Procedural Programming Object-Oriented Programming และ Functional Programming Python มีไลบรารีและแพ็คเกจจำนวนมาก ที่ช่วยอำนวยความสะดวกในการพัฒนาโปรแกรม เช่น NumPy, Pandas, Matplotlib รวมถึงไลบรารีด้านปัญญาประดิษฐ์และการเรียนรู้ของเครื่อง Python นิยมใช้ในการวิเคราะห์ข้อมูล การทำ Machine Learning และ Artificial Intelligence เนื่องจากมีเครื่องมือที่ช่วยจัดการข้อมูลขนาดใหญ่ได้อย่างมีประสิทธิภาพ Python สามารถนำไปใช้พัฒนาโปรแกรมทั่วไป เว็บไซต์ ระบบอัตโนมัติ รวมถึงการทำโครงการและงานวิจัยทางวิชาการ



รูปที่ 3: หน้าหลักของโปรแกรม Python บน Google Colab

2.3.4 GNU Linear Programming Kit (GLPK) [5]

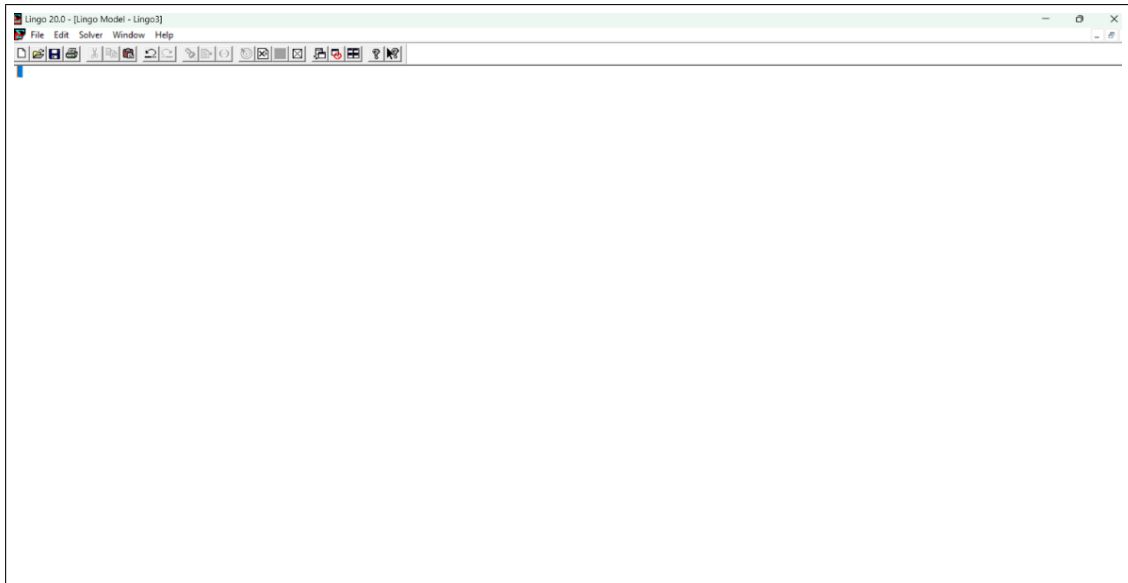
GNU Linear Programming Kit (GLPK) เป็นซอฟต์แวร์สำหรับการแก้ปัญหาการโปรแกรมเชิงเส้น พัฒนาโดย Andrew Makhorin ภายใต้โครงการ GNU โดยมีวัตถุประสงค์เพื่อเป็นเครื่องมือแบบโอเพนซอร์ส สำหรับการแก้ปัญหา Linear Programming, Integer Programming และ Mixed Integer Programming GLPK เริ่มพัฒนาตั้งแต่ช่วงทศวรรษที่ 1990 และถูกนำไปใช้อย่างแพร่หลายในด้านการศึกษาและงานวิจัยเนื่องจากสามารถใช้งานได้ฟรีและรองรับการทำงานบนหลายระบบปฏิบัติการ GLPK ทำงานโดยอาศัยแบบจำลองทางคณิตศาสตร์ ซึ่งประกอบด้วยฟังก์ชันวัตถุประสงค์ ตัวแปรตัดสินใจและเงื่อนไขข้อจำกัด จากนั้นโปรแกรมจะใช้ขั้นตอนวิธีทางคณิตศาสตร์เพื่อหาคำตอบที่เหมาะสมที่สุด GLPK รองรับการเขียนโมเดลด้วยภาษา MathProg ซึ่งช่วยให้สามารถกำหนดสมการและเงื่อนไขต่าง ๆ ได้อย่างชัดเจนเหมาะสำหรับงานวิจัยและการเรียนการสอน GLPK สามารถแก้ปัญหาเชิงเส้นและปัญหาคำนวณเต็มได้อย่างมีประสิทธิภาพแต่ยังมีข้อจำกัดด้านส่วนติดต่อผู้ใช้และต้องอาศัยความรู้พื้นฐานด้านคณิตศาสตร์ค่อนข้างมาก GLPK ถูกนำไปใช้ในงานด้านการวางแผนการจัดสรรทรัพยากรโลจิสติกส์และงานวิจัยทางวิศวกรรม



รูปที่ 4: หน้าหลักของโปรแกรม GNU Linear Programming Kit (GLPK)

2.3.5 LINGO [5]

LINGO เป็นซอฟต์แวร์เชิงพาณิชย์ พัฒนาโดยบริษัท LINDO Systems Inc. ออกแบบมาเพื่อใช้ในการสร้างแบบจำลองทางคณิตศาสตร์ และแก้ปัญหาการหาค่าที่เหมาะสมที่สุด (Optimization) LINGO เป็นโปรแกรมที่ต่อยอดมาจากแนวคิดของ LINDO ซึ่งเป็นซอฟต์แวร์ด้าน Linear Programming โดย LINGO ได้รับการพัฒนาให้รองรับปัญหาที่มีความซับซ้อนมากขึ้น และมีส่วนติดต่อผู้ใช้ที่สะดวกต่อการใช้งาน ปัจจุบัน LINGO ได้รับความนิยมอย่างมาก ทั้งในภาคการศึกษาและภาคธุรกิจ โดยเฉพาะด้านการจัดการ การวางแผน และการตัดสินใจเชิงกลยุทธ์ LINGO มีภาษาสำหรับสร้างโมเดลที่เข้าใจง่าย สามารถเขียนแบบจำลองได้กระชับ ลดความซับซ้อนในการเขียนสมการจำนวนมาก นอกจากนี้ LINGO ยังรองรับการทำงานกับข้อมูลจำนวนมาก และสามารถนำข้อมูลจากไฟล์ภายนอก เช่น Excel มาใช้ร่วมกับแบบจำลองได้ แก้ปัญหา Linear Programming แก้ปัญหา Non-linear Programming แก้ปัญหา Integer Programming มีส่วนติดต่อผู้ใช้แบบกราฟิกที่ใช้งานง่าย แสดงผลลัพธ์ในรูปแบบรายงานอย่างชัดเจน เหมาะสำหรับการวิเคราะห์และตัดสินใจเชิงธุรกิจ เป็นซอฟต์แวร์เชิงพาณิชย์ มีค่าใช้จ่าย เวอร์ชันทดลองมีข้อจำกัดด้านขนาดของปัญหา ไม่สามารถปรับแต่งระบบได้อิสระเท่าซอฟต์แวร์โอเพนซอร์ส LINGO ถูกนำไปใช้ในการวางแผนการผลิตและโลจิสติกส์ การจัดตารางงาน การวิเคราะห์ทางการเงิน การบริหารจัดการทรัพยากร การเรียนการสอนด้านการวิจัยดำเนินงาน



รูปที่ 5: หน้าหลักของโปรแกรม LINGO

3 เนื้อเรื่อง

3.1 ปัญหากำหนดการเชิงเส้นของการกระจายพนักงานเพื่อเพิ่มยอดขายแพ็คเกจอินเทอร์เน็ต [7],[11]

ปัจจุบันมีการแข่งขันธุรกิจที่รุนแรงมากขึ้น ส่งผลให้ยอดขายและผลประกอบการแต่ละองค์กรผันผวน ดังนั้นธุรกิจจึงจำเป็นต้องวางแผนเพื่อเพิ่มผลกำไร ด้วยการพิจารณาทั้งจุดแข็งและจุดอ่อนขององค์กร และทำการจำลองการเพิ่มประสิทธิภาพยอดขายโดยใช้เทคนิคต่าง ๆ เช่น การพัฒนาการทำงานของพนักงาน กลยุทธ์ส่งเสริมการขาย และการปรับปรุงบริการลูกค้า แผนการตลาดของบริษัทจำเป็นต้องได้รับการปรับปรุง โดยเฉพาะการตลาดที่เกี่ยวข้องโดยตรงกับลูกค้า เช่น ผ่านโซเชียลมีเดีย และการวิเคราะห์การตลาดโดยใช้วิธีการต่าง ๆ เช่น วิธีลำดับชั้นเชิงวิเคราะห์ ในภูมิภาค Tembung ประเทศอินโดนีเซีย มีบริษัท PT. Bistion Ponsel ซึ่งขายแพ็คเกจอินเทอร์เน็ตมือถือ โดยมีพนักงานฝ่ายการตลาดจำนวน 20 คน รับผิดชอบแคมเปญการตลาดในพื้นที่ต่าง ๆ เช่น Bandar Khalifah, Sei Rotan, Medan Estate และ Bandar Setia Regencies พนักงาน 20 คนนั้น แบ่งเป็น นักศึกษาฝึกงานด้านการตลาด 4 คน ได้ค่าตอบแทน 1,000 รูเปียห์ต่อหน่วย พนักงานการตลาดรุ่นเยาว์ 8 คน ได้ค่าตอบแทน 2,500 รูเปียห์ต่อหน่วย พนักงานการตลาดอาวุโส 8 คน ได้ค่าตอบแทน 5,000 รูเปียห์ต่อหน่วย เป้าหมายของบริษัทคือ ขายแพ็คเกจอินเทอร์เน็ตให้ได้ 3,000 แพ็คเกจต่อเดือน ด้วยทีมพนักงานจำนวนจำกัด ปัญหาของบริษัทคือจะต้องจัดสรรพนักงานอย่างไรให้สามารถขายได้ตามเป้าหมายและมีกำไรสูงสุดด้วยข้อจำกัดด้านจำนวนบุคลากร

เงื่อนไขที่สมาชิกแต่ละคนของฝ่ายการตลาดต้องปฏิบัติตามเพื่อให้บรรลุเป้าหมายมีดังนี้

1. จำนวนแพ็คเกจที่พนักงานการตลาดอาวุโสขาย ต้องมากกว่าหรือเท่ากับจำนวนแพ็คเกจที่พนักงานการตลาดรุ่นเยาว์ขายได้
 2. จำนวนแพ็คเกจที่พนักงานการตลาดอาวุโสขาย ต้องมีค่าเป็นสองเท่าของจำนวนแพ็คเกจที่นักศึกษาฝึกงานด้านการตลาดขายได้
 3. จำนวนแพ็คเกจที่พนักงานการตลาดรุ่นเยาว์ขาย ต้องมีค่าเป็นสองเท่าของจำนวนแพ็คเกจที่นักศึกษาฝึกงานด้านการตลาดขายได้
- Z คือยอดรวมเงินงูใจสูงสุดที่ต้องจ่ายต่อเดือน (หน่วยเป็น รูเปียห์)
- การกำหนดเงื่อนไขบังคับ

$$x_1 + x_2 + x_3 \leq 3000$$

ข้อจำกัดนี้มาจากจำนวนแพ็คเกจที่พนักงานทั้งสามระดับสามารถขายได้ ซึ่งมีจำนวนจำกัดที่ 3,000 แพ็คเกจ

$$x_1 \geq x_2$$

ข้อจำกัดนี้มาจากจำนวนแพ็คเกจที่พนักงานการตลาดอาวุโสขายได้ต้องมากกว่าหรือเท่ากับจำนวนแพ็คเกจที่พนักงานการตลาดรุ่นเยาว์ขายได้

$$x_1 = 2x_3$$

ข้อจำกัดนี้มาจากจำนวนแพ็คเกจที่พนักงานการตลาดอาวุโสขายได้ต้องเท่ากับสองเท่าของจำนวนแพ็คเกจที่นักศึกษาฝึกงานด้านการตลาดขายได้

$$x_2 = 2x_3$$

ข้อจำกัดนี้มาจากจำนวนแพ็คเกจที่พนักงานการตลาดรุ่นเยาว์ขายได้ต้องเท่ากับสองเท่าจำนวนแพ็คเกจที่นักศึกษาฝึกงานด้านการตลาดขายได้

และจำนวนแพ็คเกจต้องไม่เป็นค่าติดลบ

$$x_i \geq 0, i = 1, 2, 3$$

ดังนั้น สามารถเขียนรูปแบบจำลองกำหนดการเชิงเส้นของปัญหาดังกล่าว ได้ดังนี้

• ฟังก์ชันวัตถุประสงค์

$$\text{Maximize } Z = 5000x_1 + 2500x_2 + 1000x_3$$

• เงื่อนไขบังคับ

$$x_1 + x_2 + x_3 \leq 3000$$

$$x_1 \geq x_2$$

$$x_1 = 2x_3$$

$$x_2 = 2x_3$$

- เงื่อนไขที่ไม่เป็นลบ

$$x_1, x_2, x_3 \geq 0$$

สามารถเขียนตัวแบบกำหนดการเชิงเส้นให้อยู่ในรูปแบบมาตรฐานได้ดังนี้

- ฟังก์ชันวัตถุประสงค์

$$\text{Maximize } Z = 5000x_1 + 2500x_2 + 1000x_3 + 0s_1 + 0s_2 - Ma_1 - Ma_2$$

- เงื่อนไขบังคับ

$$x_1 + x_2 + x_3 + s_1 = 3000$$

$$-x_1 + x_2 + s_2 = 0$$

$$-x_1 + 2x_3 + a_1 = 0$$

$$-x_2 + 2x_3 + a_2 = 0$$

$$x_1, x_2, x_3 \geq 0$$

3.2 การแก้ปัญหาการกำหนดการเชิงเส้นโดยใช้โปรแกรม [8],[9]

ในส่วนนี้จะแสดงขั้นตอนการหาคำตอบโดยใช้ซอฟต์แวร์ช่วยคำนวณ ซึ่งช่วยเพิ่มความเร็วและลดข้อผิดพลาดจากการคำนวณด้วยมือ ดังต่อไปนี้

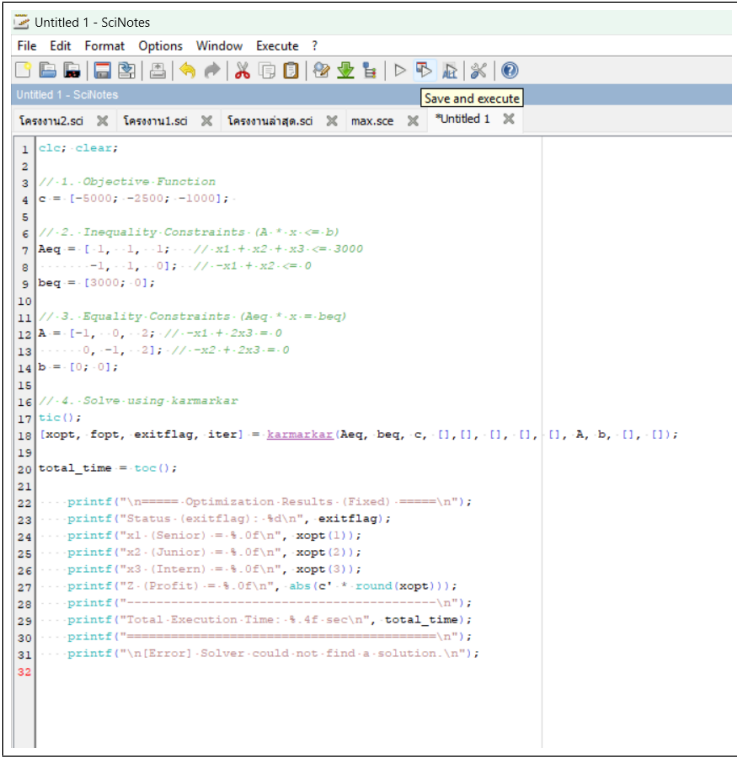
1. การแก้ปัญหาโดยโปรแกรม Microsoft Excel Solver

	x1	x2	x3	ค่าคงที่	จำกัด
ผลของกำไรต่อสินค้า	1200	1200	600		
ผลของกำไรต่อสินค้า	5000	2500	1000		
ผลของกำไรต่อสินค้า	9600000				
เงื่อนไข					
ข้อจำกัด	1	1	1	3000	3000
ข้อจำกัด	-1	1	0	0	0
ข้อจำกัด	-1	0	2	0	0
ข้อจำกัด	0	-1	2	0	0

รูปที่ 6: การตั้งค่าตัวแปรและเงื่อนไขบังคับใน Excel เพื่อแก้โจทย์โดยใช้ Solver

ในการหาคำตอบของปัญหาการโปรแกรมเชิงเส้นโดยใช้โปรแกรม Microsoft Excel สามารถใช้เครื่องมือ Solver เพื่อช่วยหาค่าที่เหมาะสมที่สุดของตัวแปรตัดสินใจ โดยเริ่มจากการกำหนดค่าตัวแปรตัดสินใจ x_1, x_2, x_3 ไว้ในเซลล์ที่กำหนด จากนั้นกำหนดค่าสัมประสิทธิ์ของฟังก์ชันวัตถุประสงค์ และใช้ฟังก์ชัน SUMPRODUCT ใน Excel เพื่อคำนวณค่าฟังก์ชันวัตถุประสงค์ $Z = 5000x_1 + 2500x_2 + 1000x_3$ ซึ่งเป็นผลรวมของผลคูณระหว่างค่าตัวแปรตัดสินใจกับค่าสัมประสิทธิ์ของแต่ละตัวแปร ต่อมาจึงกำหนดเงื่อนไขข้อจำกัดของปัญหาโดยจัดทำตารางสัมประสิทธิ์ของตัวแปรในแต่ละข้อจำกัด และคำนวณค่าด้านซ้ายของสมการข้อจำกัดโดยใช้ฟังก์ชัน SUMPRODUCT เช่นเดียวกัน หลังจากนั้นเปิดเครื่องมือ Solver จากเมนู Data แล้วกำหนดค่าในช่อง Set Objective เป็นเซลล์ที่เก็บค่าฟังก์ชันวัตถุประสงค์ กำหนดให้เป็นการหาค่าสูงสุด (Max) และกำหนด By Changing Variable Cells เป็นเซลล์ของตัวแปรตัดสินใจ x_1, x_2, x_3 จากนั้นเพิ่มเงื่อนไขข้อจำกัดของปัญหาในส่วน Subject to the Constraints ตามสมการที่กำหนด และกำหนดให้ตัวแปรตัดสินใจมีค่ามากกว่าหรือเท่ากับศูนย์ เมื่อกำหนดข้อมูลครบถ้วนแล้วให้เลือกวิธีการแก้ปัญหาเป็น Simplex LP และกดคำสั่ง Solve โปรแกรม Excel จะทำการคำนวณเพื่อหาค่าที่เหมาะสมที่สุดของตัวแปรตัดสินใจ ซึ่งจากการคำนวณพบว่า $x_1 = 1200, x_2 = 1200$ และ $x_3 = 600$ ทำให้ได้ค่าฟังก์ชันวัตถุประสงค์เท่ากับ $Z = 9,600,000$ ซึ่งเป็นค่ากำไรสูงสุดภายใต้เงื่อนไขข้อจำกัดที่กำหนด

2. การแก้ปัญหาโดยโปรแกรม Scilab



```

1 clc; clear;
2
3 //1.-Objective Function
4 c = [-5000; -2500; -1000];
5
6 //2.-Inequality Constraints (A.*-x.<=b)
7 Aeq = [1, -1, -1; ... // -x1 + x2 + x3 <= 3000
8 ... -1, -1, -1; ... // -x1 + x2 <= 0
9 beq = [3000; 0];
10
11 //3.-Equality Constraints (Aeq.*-x.=beq)
12 A = [-1, 0, 2; ... // -x1 + 2x3 = 0
13 ... 0, -1, -2; ... // -x2 + 2x3 = 0
14 b = [0; 0];
15
16 //4.-Solve using karmarkar
17 tic();
18 [xopt, fopt, exitflag, iter] = karmarkar(Aeq, beq, c, [], [], [], [], [], A, b, [], []);
19
20 total_time = toc();
21
22 printf("\n====Optimization Results (Fixed)====\n");
23 printf("Status (exitflag) : %d\n", exitflag);
24 printf("x1 (Senior) : %.0f\n", xopt(1));
25 printf("x2 (Junior) : %.0f\n", xopt(2));
26 printf("x3 (Intern) : %.0f\n", xopt(3));
27 printf("Z (Profit) : %.0f\n", abs(c' * round(xopt)));
28 printf("-----\n");
29 printf("Total Execution Time : %.4f sec\n", total_time);
30 printf("====\n");
31 printf("\n[Error] Solver could not find a solution.\n");
32

```

คำอธิบายการทำงานของโปรแกรม Scilab

- กำหนดเป้าหมาย (Objective Function)

บรรทัดที่ 4 เป็นการกำหนดตัวแปร c ซึ่งเป็นค่าสัมประสิทธิ์ของกำไรที่ได้จากพนักงานแต่ละตำแหน่ง โดยมีการใส่เครื่องหมายลบ (-) เพื่อแปลงปัญหาการหาค่าไรสูงสุด (Maximization) ให้อยู่ในรูปแบบการหาค่าต่ำสุด (Minimization) ตามรูปแบบที่โปรแกรม Scilab ใช้ในการคำนวณ

- กำหนดเงื่อนไขข้อจำกัด (Constraints)

โค้ดนี้มีการกำหนดข้อจำกัดเกี่ยวกับสัดส่วนและจำนวนพนักงาน โดยแบ่งเป็น 2 ประเภท ได้แก่

- ข้อจำกัดแบบอสมการ (Inequality Constraints): กำหนดผ่านตัวแปร A และ b
- ข้อจำกัดแบบสมการ (Equality Constraints): กำหนดผ่านตัวแปร Aeq และ beq

- การคำนวณหาจุดที่เหมาะสมที่สุด (Solving)

บรรทัดที่ 17-19 ใช้คำสั่ง `tic()`; และ `toc()`; เพื่อจับเวลาที่ใช้ในการคำนวณของโปรแกรม และเรียกใช้ฟังก์ชัน `karmarkar` สำหรับแก้ปัญหา Linear Programming

1. ข้อมูลที่ป้อนเข้าไปในโปรแกรม (Inputs)

ค่าที่อยู่ในวงเล็บ (...) หลังคำว่า `karmarkar` คือข้อมูลเงื่อนไขต่าง ๆ ที่ใช้ในการคำนวณ โดยมีลำดับดังนี้:

- Aeq , beq (ตำแหน่งที่ 1 และ 2) เป็นชุดข้อมูลที่เก็บเงื่อนไขแบบสมการ

- c (ตำแหน่งที่ 3) เป็นค่าสัมประสิทธิ์ของฟังก์ชันวัตถุประสงค์ ซึ่งแสดงกำไรที่ได้จากพนักงานแต่ละตำแหน่ง
- [], [], [], [], [] (ตำแหน่งที่ 4–8) เป็นการกำหนดให้โปรแกรมใช้ค่าเริ่มต้น (Default) ของระบบ
- A, b (ตำแหน่งที่ 9 และ 10) เป็นชุดข้อมูลที่เก็บเงื่อนไขแบบอสมการ
- [], [] (ตำแหน่งที่ 11 และ 12) เป็นการห้ามการกำหนดขอบเขตของตัวแปร

2 ผลลัพธ์ที่ได้จากโปรแกรม (Outputs)

ค่าที่อยู่ในวงเล็บเหลี่ยม [...] ด้านซ้ายของเครื่องหมายเท่ากับ คือผลลัพธ์ที่โปรแกรมคำนวณได้ ซึ่งประกอบด้วย

- xopt คือ คำตอบที่เหมาะสมที่สุด ซึ่งแสดงจำนวนพนักงานในแต่ละตำแหน่ง
- fopt คือ ค่ากำไรรวมที่ได้จากการคำนวณฟังก์ชันวัตถุประสงค์
- exitflag คือ สถานะของการแก้ปัญหา
- iter คือ จำนวนรอบที่คอมพิวเตอร์ใช้ในการคำนวณ

• การแสดงผลลัพธ์

บรรทัดที่ 22–31 เป็นส่วนที่โปรแกรมพิมพ์ผลลัพธ์ออกมาทางหน้าจอ โดยแสดงข้อมูลดังนี้

- จำนวนพนักงานที่เหมาะสมที่สุดในแต่ละตำแหน่ง ได้แก่ Senior (x_1), Junior (x_2) และ Intern (x_3)
- ค่า Z (Profit) ซึ่งเป็นกำไรรวมสูงสุดที่บริษัทจะได้รับ
- Total Execution Time ซึ่งเป็นเวลาที่คอมพิวเตอร์ใช้ในการประมวลผลทั้งหมด

```
Scilab 2024.1.0 Console

===== Optimization Results (Fixed) =====
Status (exitflag): 1
x1 (Senior) = 1200
x2 (Junior) = 1200
x3 (Intern) = 600
Z (Profit) = 9600000
-----
Total Execution Time: 0.0422 sec
=====
```

รูปที่ 7: ผลลัพธ์การทำงานของโปรแกรม Scilab จากการประมวลผลพบว่าค่าของตัวแปรตัดสินใจที่เหมาะสมที่สุดคือ $x_1 = 1200$, $x_2 = 1200$ และ $x_3 = 600$ ซึ่งทำให้ได้กำไรสูงสุดเท่ากับ 9,600,000 หน่วย โดยโปรแกรมใช้เวลาในการคำนวณประมาณ 0.0422 วินาที

3. การแก้ปัญหาโดยโปรแกรม Python บน Google Colab



```

colab โดรงงาน python
import numpy as np
from scipy.optimize import linprog
import time

c = [-5000, -2500, -1000]
start_time = time.time()
# เงื่อนไขบน
A_ub = [
    [1, 1, 1], # x1+x2+x3<=3000
    [-1, 1, 0] # -x1+x2<=0
]
b_ub = [3000, 0]

# เงื่อนไขล่าง
A_eq = [
    [-1, 0, 2], # -x1+2*x3=0
    [0, -1, 2] # -x2+2*x3=0
]
b_eq = [0, 0]

# เรียกใช้ linprog
res = linprog(c,
              A_ub=A_ub,
              b_ub=b_ub,
              A_eq=A_eq,
              b_eq=b_eq,
              bounds=(0, None),
              method='simplex')

end_time = time.time()

if res.success:
    x1, x2, x3 = res.x
    print("x1 =", round(x1, 4))
    print("x2 =", round(x2, 4))
    print("x3 =", round(x3, 4))
    print("Maximum Profit =", round(-res.fun, 2))
    print("Execution time =", round(end_time - start_time, 6), "seconds")
else:
    print("No solution found")
  
```

รูปที่ 8: การเขียนโค้ด Python บน Google Colab

คำอธิบายการทำงานของโปรแกรม Python

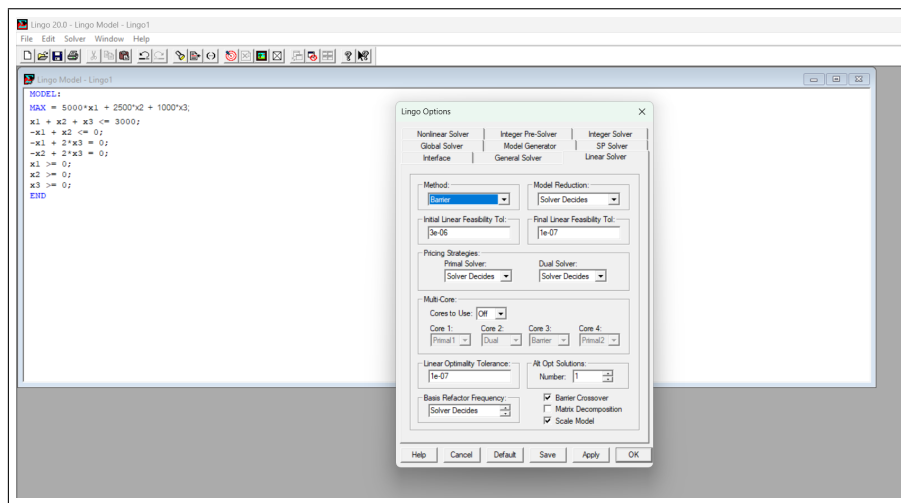
- `import numpy as np` ใช้สำหรับนำเข้าไลบรารี NumPy เพื่อช่วยจัดการข้อมูลเชิงตัวเลขและโครงสร้างข้อมูลแบบอาร์เรย์
- `from scipy.optimize import linprog` นำเข้าฟังก์ชัน `linprog` จากไลบรารี `scipy.optimize` ซึ่งใช้สำหรับแก้ปัญหาการหาค่าเหมาะที่สุดเชิงเส้น (Linear Programming)
- `import time` นำเข้าไลบรารี `time` เพื่อใช้วัดระยะเวลาในการทำงานของโปรแกรม
- `c = [-5000, -2500, -1000]` กำหนดค่าสัมประสิทธิ์ของฟังก์ชันวัตถุประสงค์ โดยใช้ค่าติดลบ เนื่องจากฟังก์ชัน `linprog` รองรับการหาค่าต่ำสุด (Minimization) ดังนั้นจึงต้องแปลงปัญหาการหาค่าสูงสุดให้เป็นการหาค่าต่ำสุดของค่าติดลบ
- `start_time = time.time()` บันทึกเวลาเริ่มต้นก่อนที่โปรแกรมจะเริ่มทำการคำนวณ
- `A_ub = [[1,1,1], [-1,1,0]]` กำหนดเมทริกซ์ของเงื่อนไขข้อจำกัดในรูปแบบสมการ $Ax \leq b$
- `b_ub = [3000,0]` กำหนดค่าด้านขวาของสมการสมการ ซึ่งสอดคล้องกับเงื่อนไขข้อจำกัดของปัญหา
- `A_eq = [[-1,0,2], [0,-1,2]]` กำหนดเมทริกซ์ของเงื่อนไขข้อจำกัดในรูปแบบสมการ $Ax = b$
- `b_eq = [0,0]` กำหนดค่าด้านขวาของสมการเท่ากับ

- `res = linprog(...)` เรียกใช้ฟังก์ชัน `linprog` เพื่อคำนวณหาคำตอบที่เหมาะสมที่สุดของปัญหา โดยกำหนดฟังก์ชันวัตถุประสงค์ เงื่อนไขข้อจำกัด และขอบเขตของตัวแปร
- `bounds=(0,None)` กำหนดขอบเขตของตัวแปรตัดสินใจให้มีค่าไม่ติดลบ กล่าวคือ $x_1, x_2, x_3 \geq 0$
- `method='simplex'` กำหนดให้โปรแกรมใช้วิธี Simplex Method ในการแก้ปัญหา Linear Programming
- `end_time = time.time()` บันทึกเวลาหลังจากโปรแกรมทำการคำนวณเสร็จสิ้น
- `if res.success:` ตรวจสอบว่าการคำนวณสามารถหาคำตอบได้สำเร็จหรือไม่
- `x1,x2,x3 = res.x` กำหนดค่าของตัวแปรตัดสินใจจากผลลัพธ์ที่ได้จากการคำนวณ
- `print(...)` ใช้สำหรับแสดงค่าของตัวแปร x_1, x_2, x_3 และค่ากำไรสูงสุดของฟังก์ชันวัตถุประสงค์
- `print("Execution Time")` แสดงระยะเวลาในการประมวลผลของโปรแกรม
- `else: print("No solution found")` หากโปรแกรมไม่สามารถหาคำตอบได้ จะแสดงข้อความว่าไม่พบคำตอบของปัญหา

```
... x1 = 1200.0
    x2 = 1200.0
    x3 = 600.0
    Maximum Profit = 9600000.0
    Execution Time = 0.024763 seconds
    /tmp/ipython-input-314/2428489956.py:23: DeprecationWarning:
      res = linprog(c,
```

รูปที่ 9: ผลลัพธ์การทำงานของโปรแกรม Python จากการประมวลผลพบว่าค่าของตัวแปรตัดสินใจที่เหมาะสมที่สุดคือ $x_1 = 1200$, $x_2 = 1200$ และ $x_3 = 600$ ซึ่งทำให้ได้ค่ากำไรสูงสุดเท่ากับ 9,600,000 หน่วย โดยโปรแกรมใช้เวลาในการคำนวณประมาณ 0.024763 วินาที

4. การแก้ปัญหาโดยโปรแกรม LINGO



รูปที่ 10: การเขียนแบบจำลองปัญหาในโปรแกรม LINGO โดยกำหนดฟังก์ชันวัตถุประสงค์ และเงื่อนไขข้อจำกัด (Constraints) ของปัญหา เพื่อให้โปรแกรมทำการประมวลผลและหาคำตอบที่เหมาะสมที่สุด

LINGO/WIN64 20.0.23 (5 Sep 2023), LINDO API 14.0.5099.295

Licensee info: Eval Use Only
License expires: 18 JUL 2026

Global optimal solution found.
Objective value: 9600000.
Infeasibilities: 0.000000
Total solver iterations: 0
Elapsed runtime seconds: 0.13

Model Class: LP

Total variables: 3
Nonlinear variables: 0
Integer variables: 0

Total constraints: 8
Nonlinear constraints: 0

Total nonzeros: 15
Nonlinear nonzeros: 0

Variable	Value	Reduced Cost
X1	1200.000	0.000000
X2	1200.000	0.000000
X3	600.0000	0.000000

Row	Slack or Surplus	Dual Price
1	9600000.	1.000000
2	0.000000	3200.000
3	0.000000	0.000000
4	0.000000	-1800.000
5	0.000000	700.0000
6	1200.000	0.000000
7	1200.000	0.000000
8	600.0000	0.000000

รูปที่ 11: ผลลัพธ์จากการประมวลผลของโปรแกรม LINGO พบว่าสามารถหาคำตอบที่เหมาะสมที่สุด (Global Optimal Solution) ได้ โดยค่าฟังก์ชันวัตถุประสงค์เท่ากับ 9,600,000 และค่าตัวแปรตัดสินใจที่เหมาะสมที่สุดคือ $x_1 = 1200$, $x_2 = 1200$ และ $x_3 = 600$

5. การแก้ปัญหาโดยโปรแกรม GNU Linear Programming Kit (GLPK)



```

model.mod
maxproblem.mod
File Edit View

var x1 >= 0;
var x2 >= 0;
var x3 >= 0;

maximize Z: 5000*x1 + 2500*x2 + 1000*x3;

s.t. c1: x1 + x2 + x3 <= 3000;
s.t. c2: -x1 + x2 <= 0;
s.t. c3: -x1 + 2*x3 = 0;
s.t. c4: -x2 + 2*x3 = 0;

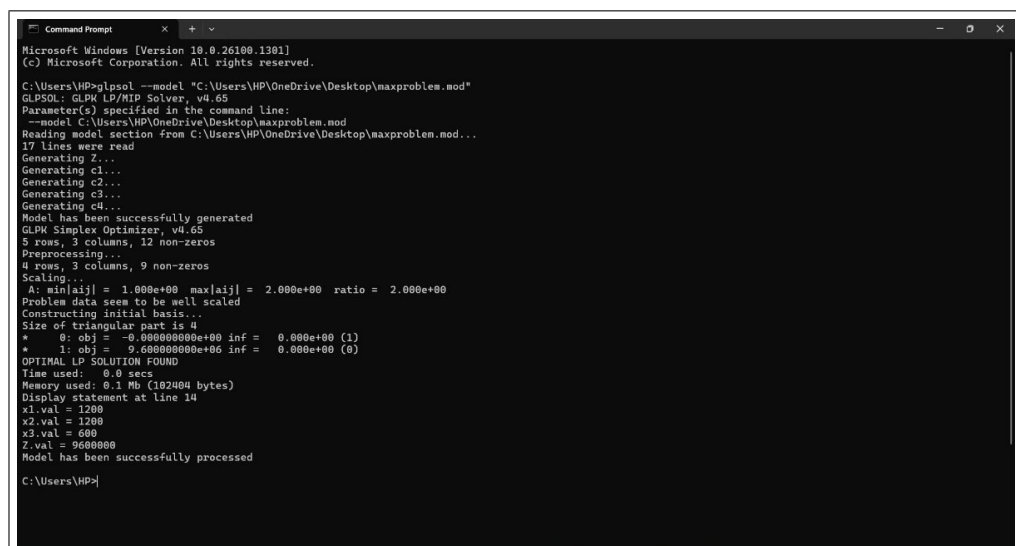
solve;

display x1, x2, x3, Z;

end;

```

รูปที่ 12: การเขียนโมเดลในรูปแบบไฟล์ .mod ด้วย Notepad



```

Command Prompt
Microsoft Windows [Version 10.0.26100.1381]
(c) Microsoft Corporation. All rights reserved.

C:\Users\HP>glpsol --model "C:\Users\HP\OneDrive\Desktop\maxproblem.mod"
GLPSOL: GLPK LP/MIP Solver, v4.65
Parameter(s) specified in the command line:
--model C:\Users\HP\OneDrive\Desktop\maxproblem.mod
Reading model section from C:\Users\HP\OneDrive\Desktop\maxproblem.mod...
17 lines were read
Generating Z...
Generating c1...
Generating c2...
Generating c3...
Generating c4...
Model has been successfully generated
GLPK Simplex Optimizer, v4.65
5 rows, 3 columns, 12 non-zeros
Preprocessing...
4 rows, 3 columns, 9 non-zeros
Scaling...
A: min|aij| = 1.000e+00 max|aij| = 2.000e+00 ratio = 2.000e+00
Problem data seem to be well scaled
Constructing initial basis...
Size of triangular part is 4
* 0: obj = -5.000000000e+00 inf = 0.000e+00 (1)
* 1: obj = 9.600000000e+06 inf = 0.000e+00 (0)
OPTIMAL LP SOLUTION FOUND
Time used: 0.0 secs
Memory used: 0.1 Mb (192484 bytes)
Display statement at line 14
x1.val = 1200
x2.val = 1200
x3.val = 600
Z.val = 9600000
Model has been successfully processed

C:\Users\HP>

```

รูปที่ 13: ผลลัพธ์การคำนวณผ่าน Command Prompt โดยใช้คำสั่ง glpsol

3.3 ประเมินประสิทธิภาพของโปรแกรม [6],[9]

สเปกซ์ของเครื่องที่ใช้ในการทดลอง ทั้งหมดดำเนินการบนเครื่องคอมพิวเตอร์โน้ตบุ๊กที่มีคุณสมบัติดังนี้

รุ่นเครื่อง: HP Laptop 15-fd0xxx หน่วยประมวลผล (CPU): 13th Gen Intel(R) Core(TM) i5-1334U ความเร็ว 1.30 GHz หน่วยความจำหลัก (RAM): 16 GB สถาปัตยกรรมระบบ: 64-bit operating system, x64-based processor ระบบปฏิบัติการ: Windows 11 Home Single Language เวอร์ชันระบบปฏิบัติการ: 24H2 OS Build: 26100.1301

เวอร์ชันของโปรแกรมที่ใช้ในการทดลอง คือ โปรแกรมที่ใช้ในการทดลองเพื่อเปรียบเทียบประสิทธิภาพในการแก้ปัญหา Linear Programming มีดังนี้

Microsoft Excel (Solver Add-in)

Scilab เวอร์ชัน 6.x

Python เวอร์ชัน 3.x

Lingo เวอร์ชัน 18.0

GLPK (GNU Linear Programming Kit)

ตารางที่1: ผลประเมินประสิทธิภาพของโปรแกรม

ประสิทธิภาพ	Excel	Scilab	Python	Lingo	GLPK
ความถูกต้อง (%)	100%	100%	100%	100%	100%
เวลา (วินาที)	0.0700	0.0064	0.0247	0.1300	0.0000
ความง่าย (1-5 คะแนน)	5	4	3.5	3.5	2.5
ความเหมาะสม	ดีมาก	ดี	ดี	ปานกลาง	พอใช้

จากตารางแสดงผลการประเมินประสิทธิภาพของโปรแกรม

1. ความถูกต้องของผลลัพธ์ทุกโปรแกรม ได้แก่ Excel Scilab Python Lingo และ GLPK ให้ผลลัพธ์ที่ถูกต้อง 100% เท่ากันทั้งหมด แสดงว่าทุกเครื่องมือสามารถแก้ปัญหาได้ถูกต้องเหมือนกัน
2. เวลาในการประมวลผล GLPK ใช้เวลาน้อยที่สุด (0.0000 วินาที) รองลงมาคือ Scilab (0.0064 วินาที) Python (0.0247 วินาที) Excel solver (0.0700 วินาที) และ Lingo ใช้เวลามากที่สุด (0.1300 วินาที)
3. ความง่ายในการใช้งาน (คะแนน 1-5) Excel solver ใช้งานง่ายที่สุด (5 คะแนน) รองลงมาคือ Scilab (4 คะแนน) Python (3.5 คะแนน) Lingo (3 คะแนน) และ GLPK ใช้งานยากที่สุดในกลุ่ม (2.5 คะแนน)
4. ความเหมาะสมโดยรวม Excel solver อยู่ในระดับดีมาก Scilab Python อยู่ในระดับดี Lingo อยู่ในระดับปานกลาง และ GLPK อยู่ในระดับพอใช้

สรุปผลการประเมินประสิทธิภาพของโปรแกรม

จากการประเมินประสิทธิภาพของโปรแกรมที่ใช้ในการแก้ปัญหาเชิงเส้น (Linear Programming) ได้แก่ Excel solver, Scilab, Python, Lingo และ GLPK ให้ผลลัพธ์ที่มีความถูกต้องเท่ากันที่ร้อยละ 100 ในด้านเวลาในการคำนวณ พบว่าโปรแกรม GLPK ใช้เวลาในการคำนวณน้อยที่สุด รองลงมาคือ Scilab และ Python ตามลำดับ ส่วนโปรแกรม Lingo ใช้เวลามากที่สุดเมื่อเปรียบเทียบกับโปรแกรมอื่น แสดงให้เห็นว่าโปรแกรมที่พัฒนามาเพื่อแก้ปัญหาเชิงเส้นโดยเฉพาะ สามารถให้ผลลัพธ์ได้รวดเร็วกว่าในบางกรณี สำหรับความง่ายในการใช้งาน โปรแกรม Excel solver ได้คะแนนสูงสุด เนื่องจากมีรูปแบบการใช้งานที่ไม่ซับซ้อนและเหมาะสำหรับผู้เริ่มต้น ขณะที่ Scilab และ Python มีความยืดหยุ่นและเหมาะสำหรับผู้ที่มีพื้นฐานด้านการเขียนโปรแกรม ส่วนโปรแกรม Lingo และ GLPK แม้จะมีความง่ายในการใช้งานน้อยกว่า แต่มีความเหมาะสมสำหรับงานด้านการหาคำตอบเชิงเหมาะสมโดยตรง

สำหรับกรณีศึกษา ซึ่ง เป็นปัญหากำหนดการเชิงเส้นที่มีขนาดไม่ใหญ่มาก และต้องการความสะดวกในการใช้งานและการแสดงผลที่เข้าใจง่าย โปรแกรม Excel Solver จึงถือว่าเหมาะสมที่สุด เนื่องจากสามารถตั้งค่าแบบจำลองและวิเคราะห์ผลลัพธ์ได้ง่าย โครงการนี้ได้แสดงให้เห็นถึงประสิทธิภาพของวิธีซิมเพล็กซ์ ในการช่วยบริษัทการตลาดกำหนดการกระจายเป้าหมายการขาย แพ็กเกจอินเทอร์เน็ต เพื่อเพิ่มรายได้ของธุรกิจผ่านการมอบแรงจูงใจเพิ่มเติมให้กับพนักงานการตลาด ผลการวิจัยชี้ให้เห็นว่า การกระจายเป้าหมายการขายที่เหมาะสมที่สุดคือ

- พนักงานการตลาดอาวุโส: 150 แพ็กเกจ พร้อมเงินจูงใจ 750,000 รูเปียห์ต่อเดือน
- พนักงานการตลาดรุ่นเยาว์: 150 แพ็กเกจ พร้อมเงินจูงใจ 275,000 รูเปียห์ต่อเดือน
- นักศึกษาฝึกงานด้านการตลาด: 150 แพ็กเกจ พร้อมเงินจูงใจ 150,000 รูเปียห์ต่อเดือน

โดยรวมแล้ว บริษัทได้จัดสรรเงินจูงใจ 9,600,000 รูเปียห์ต่อเดือน และสร้างกำไรรวม 30,000,000 รูเปียห์ต่อเดือน

ดังนั้น การเลือกใช้โปรแกรมในการแก้ปัญหาควรพิจารณาจากลักษณะของปัญหา ขนาดของข้อมูล ระยะเวลาในการคำนวณ และความสะดวกในการใช้งาน เพื่อให้เหมาะสมกับวัตถุประสงค์ของงานมากที่สุด

ข้อดีของโปรแกรม

- Excel Solver เป็นโปรแกรมที่ใช้งานง่าย เหมาะสำหรับผู้เริ่มต้น ใช้เวลาคำนวณน้อย และเหมาะกับปัญหาขนาดเล็กถึงปานกลาง
- Scilab มีความเหมาะสมกับงานด้านคณิตศาสตร์และการคำนวณเชิงตัวเลข ให้ผลลัพธ์ที่มีความถูกต้อง และเป็นซอฟต์แวร์ฟรี
- Python มีความยืดหยุ่นสูง สามารถประยุกต์ใช้กับปัญหาได้หลากหลาย รองรับไลบรารีด้านคณิตศาสตร์และการคำนวณจำนวนมาก
- LINGO เป็นโปรแกรมที่ออกแบบมาเฉพาะสำหรับการแก้ปัญหาเชิงเหมาะสม (Optimization) ให้ผลลัพธ์ที่มีความถูกต้องสูงและใช้เวลาคำนวณน้อย
- GNU มีประสิทธิภาพในการคำนวณสูง เป็นซอฟต์แวร์ฟรี และเหมาะสำหรับงานด้านวิศวกรรมและคณิตศาสตร์

ข้อเสียของโปรแกรม

- Excel Solver ไม่เหมาะกับปัญหาที่มีความซับซ้อนหรือมีขนาดใหญ่ และมีข้อจำกัดในการสร้างแบบจำลองทางคณิตศาสตร์
- Scilab การใช้งานอาจไม่สะดวกสำหรับผู้เริ่มต้น และมีชุมชนผู้ใช้น้อยเมื่อเทียบกับภาษาอื่น
- Python ใช้เวลาคำนวณมากกว่าบางโปรแกรม และจำเป็นต้องมีพื้นฐานด้านการเขียนโปรแกรม
- LINGO เป็นโปรแกรมที่มีค่าใช้จ่ายในเวอร์ชันเต็ม และมีความยืดหยุ่นน้อยกว่าภาษาเขียนโปรแกรมทั่วไป
- GNU การใช้งานค่อนข้างซับซ้อนสำหรับผู้เริ่มต้น และมีเอกสารภาษาไทยค่อนข้างจำกัด

ข้อเสนอแนะ

จากผลการประเมินสามารถเลือกใช้โปรแกรมให้เหมาะสมกับลักษณะของงานได้ดังนี้ หากเป็น ผู้เริ่มต้นหรือทำการคำนวณพื้นฐาน แนะนำให้ใช้ Excel Solver สำหรับงานคำนวณเชิงคณิตศาสตร์เชิงตัวเลข เหมาะกับการใช้ Scilab หรือ GNU ในกรณีที่ต้องการความยืดหยุ่นและการต่อยอดในระยะยาว Python จะเป็นทางเลือกที่เหมาะสม และหากเป็นปัญหาเชิงเหมาะสมโดยตรง โปรแกรม LINGO จะให้ประสิทธิภาพที่ดีที่สุด

4 ภาคผนวก

4.1 การแก้ปัญหาการกำหนดการเชิงเส้นด้วยวิธีซิมเพล็กซ์ [3],[12]

	C_j	5000	2500	1000	0	0	-M	-M			
C_B	X_B	x_1	x_2	x_3	s_1	s_2	a_1	a_2	X_B	X_B/x_3	ดำเนินการตามแถว
0	s_1	1	1	1	1	0	0	0	3000	3000	$R_1 - \frac{1}{2}R_3$
0	s_2	-1	1	0	0	1	0	0	0	-	R_2
-M	a_1	-1	0	2	0	0	1	0	0	0	$\frac{1}{2}R_3$
-M	a_2	0	-1	2	0	0	0	1	0	0	$R_4 - R_3$
Z_j		M	M	-4M	0	0	-M	-M			
$C_j - Z_j$		5000-M	2500-M	1000+4M	0	0	0	0			

ตารางเริ่มต้นของวิธีซิมเพล็กซ์โดยใช้เทคนิค Big-M เพื่อแก้ปัญหาการหาค่ามากที่สุดของฟังก์ชันวัตถุประสงค์ ภายใต้ข้อจำกัดที่กำหนด โดยมีการเพิ่มตัวแปรส่วนเกิน s_1, s_2 และตัวแปรเทียม a_1, a_2 เพื่อให้สมการข้อจำกัดอยู่ในรูปมาตรฐาน จากนั้นคำนวณค่า Z_j และ $C_j - Z_j$ เพื่อพิจารณาเลือกตัวแปรเข้าสู่ฐาน (Entering variable) ซึ่งพบว่าคอลัมน์ x_3 มีค่ามากที่สุด จึงเลือก x_3 เป็นตัวแปรเข้าสู่ฐาน และทำการทดสอบอัตราส่วน X_B/x_3 เพื่อหาตัวแปรออกจากฐาน (Leaving variable)

	C_j	5000	2500	1000	0	0	-M	-M			
C_B	X_B	x_1	x_2	x_3	s_1	s_2	a_1	a_2	X_B	X_B/x_1	ดำเนินการตามแถว
0	s_1	1.5	1	0	1	0	-0.5	0	3000	2000	$R_1 - 1.5R_4$
0	s_2	-1	1	0	0	1	0	0	0	0	$R_2 + R_4$
1000	x_3	-0.5	0	1	0	0	0.5	0	0	0	$R_3 + 0.5R_4$
-M	a_2	1	-1	0	0	0	-1	1	0	0	$R_4(\text{new})$
Z_j		-500 - M	M	1000	0	0	500 + M	-M	0		
$C_j - Z_j$		5500 + M	2500 - M	0	0	0	-500 - 2M	0			

ตารางที่ 2 ตารางนี้เป็นผลลัพธ์หลังจากทำการ Pivot โดยให้ตัวแปร x_3 เข้าสู่ฐานแทนตัวแปร a_1 จากนั้นทำการปรับค่าของแถวต่าง ๆ ด้วยกระบวนการแปลงแถว (Row operation) เพื่อให้คอลัมน์ของตัวแปร x_3 อยู่ในรูปเวกเตอร์เอกลักษณ์ แล้วคำนวณค่า Z_j และ $C_j - Z_j$ ใหม่ เพื่อพิจารณาเลือกตัวแปรเข้าสู่ฐานในรอบถัดไป

	C_j	5000	2500	1000	0	0	-M	-M			
C_B	X_B	x_1	x_2	x_3	s_1	s_2	a_1	a_2	X_B	X_B/x_2	ดำเนินการตามแถว
0	s_1	0	2.5	0	1	0	1	-1.5	3000	1200	$R_1(new) = \frac{1}{2.5}R_1$
0	s_2	0	0	0	0	1	-1	1	0	-	$R_2(new) = R_2$
1000	x_3	0	-0.5	1	0	0	0	0.5	0	0	$R_3 + 0.5R_1(new)$
5000	x_1	1	-1	0	0	0	-1	1	0	0	$R_4 + R_1(new)$
Z_j		5000	-5500	1000	0	0	-5000	5500	0		
$C_j - Z_j$		0	8000	0	0	0	5000-M	-5500-M			

ตารางที่ 3 ตารางนี้แสดงผลของการดำเนินการซิมเพล็กซ์ในรอบถัดไป โดยพิจารณาค่า $C_j - Z_j$ อีกครั้ง พบว่าคอลัมน์ x_1 มีค่ามากที่สุด จึงเลือก x_1 เป็นตัวแปรเข้าสู่ฐาน และคำนวณค่าอัตราส่วน X_B/x เพื่อหาตัวแปรที่จะออกจากฐาน ซึ่งได้ตัวแปร s_1 เป็นตัวแปรออกจากฐาน จากนั้นทำการ Pivot เพื่อปรับค่าของตารางใหม่

	C_j	5000	2500	1000	0	0	-M	-M	
C_B	X_B	x_1	x_2	x_3	s_1	s_2	a_1	a_2	X_B
2500	x_2	0	1	0	0.4	0	0.4	-0.6	1200
0	s_2	0	0	0	0	1	-1	1	0
1000	x_3	0	0	1	0.2	0	0.2	0.2	600
5000	x_1	1	0	0	0.4	0	-0.6	0.4	1200
Z_j		5000	2500	1000	3200	0	-1800	700	9600000
$C_j - Z_j$		0	0	0	-3200	0	1800-M	-700-M	

จากตารางที่ 4 เป็นผลการทำซ้ำครั้งที่ 3 (Iteration 3) ซึ่งเป็นตารางคำตอบที่เหมาะสมที่สุด (Optimal Solution) ของวิธีซิมเพล็กซ์ เนื่องจากค่าของ $C_j - Z_j$ ในทุกคอลัมน์มีค่าเป็นศูนย์ ซึ่งเป็นไปตามเงื่อนไขเหมาะสมที่สุดของปัญหาการหาค่าสูงสุด

จากตารางที่ 4 พบว่าการจ่ายเงินจูงใจ (Incentive) สูงสุดสำหรับพนักงานแผนการตลาดอยู่ที่ 9,600,000 รูปีต่อเดือน

การกระจายเป้าหมายการขายมีดังนี้:

- พนักงานการตลาดอาวุโส : $1200 \div 8 = 150$ แพ็กเกจต่อคน
- พนักงานการตลาดรุ่นเยาว์ : $1200 \div 8 = 150$ แพ็กเกจต่อคน
- นักศึกษาฝึกงานด้านการตลาด : $600 \div 4 = 150$ แพ็กเกจต่อคน

พนักงานการตลาดแต่ละระดับจะได้รับเงินจูงใจตามแผนการกระจาย ดังนี้:

- พนักงานการตลาดอาวุโส: $(150 \times 5,000) = 750,000$ รูเปียห์ต่อเดือน
- พนักงานการตลาดรุ่นเยาว์: $(150 \times 2,500) = 375,000$ รูเปียห์ต่อเดือน
- นักศึกษาฝึกงานด้านการตลาด: $(150 \times 1,000) = 150,000$ รูเปียห์ต่อเดือน

ซึ่งแสดงให้เห็นว่า เพื่อให้บริษัท PT. Bistion Ponsel สามารถบรรลุเป้าหมายการขายอินเทอร์เน็ต 3,000 แพ็กเกจต่อเดือน บริษัทจำเป็นต้องจัดสรรเงินจูงใจจำนวน 9,600,000 รูเปียห์ต่อเดือน ให้กับทีมการตลาดที่อยู่ใน 4 เขต

หากบริษัทสามารถขายอินเทอร์เน็ตได้ 3,000 แพ็กเกจใน 1 เดือน และมีกำไรสุทธิ รูเปียห์ต่อเดือน 10,000 ต่อแพ็กเกจ กำไรรวมต่อเดือนจะเท่ากับ $(10,000 \times 3,000) = 30,000,000$ รูเปียห์ต่อเดือน

เอกสารอ้างอิง

- [1] Anti, A. R., & Sudrajat, A. (2021). Optimization of profits using linear programming simplex method. *Jurnal Manajemen*, 13, 188–194. <https://journal.feb.unmul.ac.id/index.php/JURNALMANAJEMEN/article/view/9124/1318>
- [2] Apriansyah, & Baysha, M. H. (2018). Pengaruh media pembelajaran matrix laboratory (MATLAB) terhadap hasil belajar siswa. *Jurnal Teknologi Pendidikan*, 3, 10–20. <https://e-journal.undikma.ac.id/index.php/jtp/article/view/1235/1033>
- [3] Aprilyanti, S., Pratiwi, I., & Basuki, M. (2018). Optimasi keuntungan produksi kemplang panggang menggunakan linear programming melalui metode simpleks. *Seminar dan Konferensi Nasional IDEC*, 1–11. <https://idec.ft.uns.ac.id/wp-content/uploads/2018/05/ID038.pdf>
- [4] Busrah, Z. (2019). *Buku Ajar Matematika Komputasi Berbasis Pemrograman MATLAB*. Parepare: KAAFFAH.
- [5] Clacier, R., Fitriani, R., & Wahyudin. (2023). Optimalisasi keuntungan menggunakan program linier dengan metode simpleks dan POM-QM pada produksi tahu. *Jurnal Serambi Engineering*, 8, 5162–5169. <https://doi.org/10.32672/jse.v8i2.5721>
- [6] Dali, D., Lesnussa, Y. A., & Ilwaru, V. Y. I. (2022). Optimalisasi keuntungan menggunakan metode branch and bound pada produksi spring bed. *Jurnal Matematika*, 12, 78–88. <https://pdfs.semanticscholar.org/7120/71df3d517f61dd655605b884a7e503e6c9aa.pdf>
- [7] Dharmawan, D., & Arifin, J. (2022). Optimalisasi keuntungan tempat pencucian sepatu dengan metode grafik dan metode simpleks. *Jurnal Rekayasa Sistem Industri*, 8, 53–57. <https://doi.org/10.33884/jrsi.v8i1.6516>
- [8] Ghaliyah, S. F., Harahap, E., & Badruzzaman, F. H. (2021). Optimalisasi keuntungan produksi sambal menggunakan metode simpleks berbantuan software QM. *Bandung Conference Series: Mathematics*, 2, 9–16. <https://doi.org/10.29313/bcsm.v2i1.1388>
- [9] Handayani, S., Rinaldi, A., & Andriani, S. (2022). Optimalisasi keuntungan digital printing menggunakan branch and bound serta cutting plane berbasis R software. *EULER: Jurnal Ilmiah Matematika, Sains dan Teknologi*, 10, 303–313. <https://doi.org/10.34312/euler.v10i2.16960>

- [10] Hani, N., & Harahap, E. (2021). Optimasi produksi T-Shirt menggunakan metode simpleks. *Jurnal Matematika*, 20, 27–32. <https://journals.unisba.ac.id/index.php/matematika/article/view/1081/524>
- [11] Hidayah, A. A., Harahap, E., & Badruzzaman, F. H. (2022). Optimasi keuntungan bisnis bakery menggunakan program linear metode simpleks. *Jurnal Matematika*, 21, 77–83. <https://journals.unisba.ac.id/index.php/matematika/article/view/1556/786>
- [12] Hussain, M. R., Qayyum, M., & Hussain, M. E. (2019). Effect of seven steps approach on simplex method to optimize the mathematical manipulation. *International Journal of Recent Technology and Engineering (IJRTE)*, 7, 34–43. <https://www.ijrte.org/wp-content/uploads/papers/v7i5/E1948017519.pdf>
- [13] Rois, M. A. (2018). Metode simpleks program linear pada optimalisasi pengelolaan lahan parkir FST UIN Walisongo dengan konsep “Ukhuwah”. *Jurnal MIPA*, 41, 51–57. <https://journal.unnes.ac.id/nju/JM/article/view/16382/8346>