บทที่ 2

หลักการทำงานเบื้องต้นของโปรแกรมคอมพิวเตอร์

ในบทนี้จะกล่าวถึงการทำงานของโปรแกรม โดยแบ่งออกเป็น 2 หัวข้อใหญ่ คือ วิธีการเก็บบันทึกข้อมูล และวิธีการทำงานของโปรแกรม

2.1 หลักการในการเก็บบันทึกข้อมูลของโปรแกรม

ข้อมูลใดๆ จะเก็บไว้ในรูปของเมทริดซ์ ขนาดของเมทริดซ์ที่ใช้จะขึ้นอยู่กับความสำคัญของข้อมูลนั้นๆ เช่น ถ้าเป็นข้อมูลที่ใช้ในการกำหนดขอบเขตของงานที่พิจารณาจะใช้เมทริดซ์ขนาด 2 มิติ คือ ค่าพิกัดทางแนวนอนทางด้านแกน x , y เท่านั้น หรือ ถ้าเป็นข้อมูลสำหรับเส้นศูนย์กลางถนนซึ่งเป็นแนวสำคัญของถนนจะใช้เมทริดซ์ขนาด 6 มิติ ประกอบด้วยพิกัดทางด้านแกน x, y, z และค่าพารามิเตอร์อีก 3 ค่าคือ ระยะตามแนวถนน (CHAINAGE) มุมแบริ่ง (BEARING) และ ค่ารัศมี

ข้อมูลของเมทริดซ์แต่ละชุดถูกแบ่งแยกประเภทด้วยการกำหนดชื่อของชุดข้อมูลนั้นๆ ชื่อของชุดข้อมูลจะประกอบด้วยตัวอักษร 4 ตัว เช่น ชุดข้อมูลของเส้นศูนย์กลางถนนในช่วงหลัก กม. ที่ 100 + 00 ถึง 105 + 00 ใช้ชื่อว่า M105 เมื่อใดที่ต้องการเรียกใช้หรืออ้างอิงถึงข้อมูลชุดนี้ ก็สามารถเรียกโดยใช้ชื่อ M105 ได้ทันที

เนื่องจากงานทางด้านวิศวกรรมการทาง การเก็บบันทึกข้อมูลงานสำรวจ งานออกแบบ และการแสดงผลทางภาพจะกระทำในลักษณะเป็นเส้นเป็นแนว ดังนั้นชุดข้อมูลที่กล่าวไปแล้วนั้นก็จะถูกจัดเก็บในลักษณะเป็นเส้นเป็นแนวด้วยเพื่อความสะดวกในการทำงาน ชุดข้อมูลที่จัดเป็นแนวเส้นเช่นนี้เราเรียกว่าเส้นข้อมูล (STRING)

กล่าวโดยสรุปสำหรับการบันทึกข้อมูลของโปรแกรมก็คือมันจะเก็บข้อมูลไว้ในแบบจำลองตัวเลข (DIGITAL MODEL) โดยที่ชื่อของแบบจำลองจะมีได้ไม่เกิน 32 ตัวอักษรรวมช่องว่าง แต่ละแบบจำลองประกอบด้วยเส้นข้อมูลจำนวนหนึ่ง เส้นข้อมูลแต่ละเส้นจะถูกแยกแยะตามลักษณะการใช้งานโดยการกำหนดชื่อของเส้นข้อมูลนั้นด้วยตัวอักษร 4 ตัว และในแต่ละเส้นข้อมูลประกอบด้วยจุดข้อมูลจำนวนหนึ่งโดยที่จุดข้อมูลแต่ละจุดนั้นประกอบด้วยค่าพิกัดของจุดนั้นๆ ค่าพิกัดของจุดข้อมูลอาจมีตั้งแต่ 2-3 มิติและค่าพารามิเตอร์อีก 2-3 ค่าตามความสำคัญของเส้นข้อมูลนั้นๆ รูปที่ 2.1 ได้แสดงถึงลำดับขั้นของการบันทึกข้อมูล

2.1.2 <u>ประเภทของแบบจำลอง</u> สามารถแบ่งออกได้ตามลักษณะการใช้งานได้ 4 ประเภทคือ

2.1.2.1 แบบจำลองลักษณะพื้นผิว (SURFACE MODELS) ตัวอย่างเช่น

ก. แบบจำลองพื้นผิวดินเดิม
ข. แบบจำลองทางธรณีวิทยา
ค. แบบจำลองของโครงการ เช่น ถนน ทางรถไฟ เขื่อน ฯลฯ

2.1.2.2 แบบจำลองรายละเอียด (FEATURE MODELS) ตัวอย่างเช่น

ก. แบบจำลองรายละเอียดของถนน เช่น เสาไฟฟ้า รั้วกันตก
ข. แบบจำลองรายละเอียดของพื้นดินเดิม เช่น แนวต้นไม้ คลอง

2.1.2.3 แบบจำลองโครงข่าย (NETWORK MODELS) ตัวอย่างเช่น

ก. แบบจำลองโครงข่ายของท่อน้ำประปา แก๊ส โทรศัพท์

2.1.2.4 แบบจำลองนามธรรม (ABSTRACT MODELS) ตัวอย่างเช่น

ก. แบบจำลองของมลภาวะต่างๆ เช่น เสียง กลิ่น ไอพิษ

2.1.3 <u>ประเภทของเส้นข้อมูล</u> สามารถแบ่งออกได้ตามลักษณะการใช้งานเป็น 7 ประเภท คือ

2.1.3.1 เส้นข้อมูลชั้นความสูง (CONTOUR STRINGS) ประกอบด้วยค่าพิกัดในแนวแกน x, y เท่านั้น ความสูง (LEVEL) ของเส้นข้อมูลจะกำหนดเพียงครั้งเดียวภายใต้ชื่อของเส้นข้อมูลนี้ เส้นข้อมูลชั้นความสูงอาจป้อนเข้าสู่ระบบโดยใช้ TABLE DIGITIZER หรือสร้างขึ้นจากการคำนวณโดยอาศัยข้อมูลจากแบบจำลองลักษณะพื้นผิว

2.1.3.2 เส้นข้อมูลสามมิติ (THREE DIMENSIONAL STRINGS) ประกอบด้วยค่าพิกัดสามมิติ คือค่าในแนวแกน x, y, z เส้นข้อมูลนี้ใช้ในการกำหนดรูปร่างลักษณะโดยทั่วไปของแบบจำลอง เช่น เส้นขอบถนน แนวรั้ว

2.1.3.3 เส้นข้อมูลหลัก (MASTER ALIGMENT STRINGS) ประกอบด้วยค่าพิกัด 3 มิติ คือค่าพิกัดในแนวแกน x, y, z พร้อมพารามิเตอร์อีก 3 ค่า คือ ระยะตามแนวถนน มุมแบริ่ง และค่ารัศมีเส้นข้อมูลหลักจะถูกสร้างขึ้นโดยโปรแกรม ในส่วนการออกแบบแนวทางหลักของถนน (HIGHWAY MASTER ALIGNMENT OPTIONS)

2.1.3.4 เส้นข้อมูลรูปตัด (SECTION STRINGS) ประกอบด้วยค่าพิกัด 3 มิติ คือค่าพิกัดในแนวแกน x, y, z พร้อมพารามิเตอร์อีก 2 ค่า คือ ค่าพิกัดฉากจากเส้นข้อมูลที่ใช้อ้างอิง และชื่อของเส้นข้อมูลใดๆที่เส้นข้อมูลรูปตัดตัดผ่าน เส้นข้อมูลรูปตัดโดยปกติจะถูกสร้างขึ้นโดยโปรแกรมในส่วนของการสร้างรูปตัด (SECTION OPTIONS)

2.1.3.5 เส้นข้อมูลแนวสัมผัส (INTERFACE STRINGS) ประกอบด้วยค่าพิกัด 3 มิติ คือ พิกัดในแนวแกน x, y, z พร้อมพารามิเตอร์อีก 2 ค่าคือ ค่าพิกัดฉากจากเส้นข้อมูลที่ใช้อ้างอิงและมุมแบริ่ง เส้นข้อมูลแนวสัมผัสถูกสร้างโดยโปรแกรม ในส่วนของการคำนวณแนวสัมผัส (INTERFACE OPTION) ตัวอย่างของเส้นข้อมูลนี้ได้แก่แนวสัมผัสหรือจุดตัดของไหล่ทาง กับพื้นดินเดิมเป็นต้น

2.1.3.6 เส้นข้อมูลจุด (POINT STRINGS) ประกอบด้วยค่าพิกัด 3 มิติ คือ ค่าพิกัดในแนวแกน x, y และ z เช่นเดียวกับเส้นข้อมูลสามมิติ เส้นข้อมูลแบบอื่นๆ จะมีความต่อเนื่องโดยถือว่าจุดข้อมูลแต่ละจุดบนเส้นข้อมูลนั้นถูกเชื่อมต่อด้วยเส้นตรง เส้นข้อมูลจุดจะไม่มีความต่อเนื่องดังกล่าว แต่ละจุดข้อมูลถือว่าเป็นจุดโดดๆ ไม่มีความสัมพันธ์กับจุดอื่นใด เส้นข้อมูลจุดมีประโยชน์มากในงานสำรวจหรือกำหนดจุดสำคัญเช่น ต้นไม้ ก้อนหินใหญ่ ฯลฯ อย่างไรก็ดีเส้นข้อมูลจุดไม่สามารถนำไปใช้กับส่วนการคำนวณใดๆ ที่ต้องการความต่อเนื่องของเส้นข้อมูล เช่น การออกแบบแนวทางหลัก การคำนวณเส้นชั้นความสูง ฯลฯ

2.1.3.7 เส้นข้อมูลตัวอักษร (TEXT STRINGS) ประกอบด้วยค่าพิกัด 2 มิติคือ ค่าพิกัดในแนวแกน x, y พร้อมพารามิเตอร์อีก 3 ค่าคือความสูงของตัวอักษร มุมของตัวอักษรเทียบกับแกน x และตัวอักษรที่ต้องการพิมพ์เส้นข้อมูลตัวอักษรมีไว้เพื่อกำหนดตัวอักษรที่ต้องการพิมพ์ทางเครื่อง PLOTER หรือจอภาพ มิได้นำมาใช้ในการคำนวณแต่อย่างใด

2.1.4 <u>ลักษณะพิเศษของเส้นข้อมูล</u> การออกแบบแนวทางหลักของถนน ต้องเริ่มจากการออกแบบแนวทางนอนก่อนแนวทางดิ่ง แนวทางนอนที่ออกแบบไว้ในครั้งแรกจะยังไม่มีค่าระดับโปรแกรม จะกำหนดให้เส้นข้อมูลนี้เป็นแบบไม่มีค่าระดับ (NULL LEVEL) อย่างไรก็ตามในกรณีที่เส้นข้อมูลแบบไม่มีค่าระดับถูกเรียกใช้โดยส่วนของโปรแกรมที่ต้องการใช้ค่าระดับด้วย เช่น การสร้างเส้นชั้นความสูง เส้นข้อมูลแบบไม่มีค่าระดับจะไม่ถูกนำมาพิจารณาในการคำนวณ

2.1.5 <u>การกำหนดชื่อของเส้นข้อมูล</u> เส้นข้อมูลทุกตัวใน โปรแกรมต้องถูกกำหนดชื่อโดยประกอบด้วยตัวอักษร 4 ตัว และห้ามใช้ช่องว่างเป็นส่วนหนึ่งของชื่อเส้นข้อมูล โปรแกรมได้ตั้งอักษรตัวแรกให้ผู้ใช้เลือกใช้งานตามสภาพงานที่มีอยู่ และอักษรตัวแรกที่ตั้งไว้จะทำให้การแสดงผลทาง PLOTER หรือจอภาพได้สะดวกขึ้น เนื่องจากมันได้กำหนดลักษณะของเส้นที่จะใช้ตามตัวอักษรตัวแรกที่ใช้เป็นชื่อของเส้นข้อมูล

ตัวอักษรตัวแรกที่แนะนำให้ใช้สำหรับชื่อของเส้นข้อมูล แบ่งตามลักษณะการใช้งาน มีดังนี้

B อาคาร (BUILDING)
C ถนน (CARRIAGEWAY)
D คู คลอง (DITCH)
F แนวรั้วที่มนุษย์สร้าง (FENCE)
H แนวรั้วธรรมชาติ (HEDGE)
I แนวสัมผัส (INTERFACE)
L ค่าระดับ (LEVEL)
M แนวทางหลัก (MASTER ALIGNMENT)
P จุด (POINT)
V ขอบทาง (VERGE)
W แนวน้ำ (WATER LINE)
* ตัวอักษร (TEXT)

ตัวอย่างการกำหนดชื่อของเส้นข้อมูล

COCL แนวถนนด้านนอกของฝั่งซ้ายคลอง (CARRIAGEWAY OUTER CHANNEL LEFT)
CICR แนวถนนด้านในของฝั่งขวาคลอง (CARRIAGEWAY INNER CHANNEL RIGHT)
VLO1 ขอบถนนทางด้านซ้าย (LEFT VERGE)
BOO1 อาคาร (BUILDING

MCL1 แนวทางหลักเส้นศูนย์กลางถนน (MASTER ALIGNMENT OF CENTER LINE)

MSR1 แนวทางหลักของถนนที่แยกออก หมายเลข 1 (MASTER ALIGNMENT OF SLIP ROAD, No1.)

HOO1 แนวรั้วธรรมชาติ (HEDGE)

FOO1 แนวรั้วมนุษย์สร้าง (FENCE)

2.1.6 ประเภทของจุดข้อมูล บนเส้นข้อมูลใดๆ ต้องประกอบด้วยจุดข้อมูลจำนวนหนึ่งเสมอ จุดข้อมูลถูกแบ่งออกเป็น 2 ประเภท โดยแบ่งตามลักษณะการเกิดหรือการสร้างจุดข้อมูลนั้นๆ

2.1.6.1 จุดข้อมูลแบบตำแหน่งแน่นอน (EXACT POINTS) หมายถึง จุดข้อมูลที่มีค่าพิกัดเก็บบันทึกไว้ในเส้นข้อมูล โดยมีลำดับที่ของจุดข้อมูลแจ้งออกมาด้วยในการรายงานผล

2.1.6.2 จุดข้อมูลแบบไม่มีตำแหน่งแน่นอน (NON EXACT POINTS) หมายถึงจุดที่อยู่บนเส้นข้อมูล แต่อยู่ระหว่าง exact points สองจุด

ถ้าผู้ใช้กำหนด non exact point โดยที่จุดนั้นมิได้อยู่บนเส้นที่เชื่อมระหว่าง exact point MOSS จะทำการหาค่าพิกัดโดยลากเส้นจากจุดดังกล่าวให้ลงมาตั้งฉากกับเส้นเชื่อมระหว่าง exact point และใช้ค่าพิกัดของจุดตั้งฉากนั้นเป็นค่าพิกัดของจุด

2.1.7 ค่าพิกัด หน่วย และเครื่องหมายที่ใช้ โปรแกรมได้จำแนกวิธีการใช้ค่าต่างๆ ดังนี้

2.1.7.1 ระบบพิกัด (CO - ORDINATE) ค่าพิกัดทางแกน x หมายถึงค่าพิกัดทางทิศตะวันออก และ ค่าพิกัดทางแกน y หมายถึงค่าพิกัดทางทิศเหนือ โดยที่ค่าทั้งสองแกนนี้ต้องเป็นบวกเสมอ ค่าระดับความสูง (แกน z) สามารถเป็นได้ทั้งทางบวกและลบ ในกรณีที่เป็นเส้นข้อมูลแบบ

ไม่มีระดับความสูง (NULL LEVEL) การป้อนข้อมูลหรือแสดงผลจะกำหนดให้มีค่าระดับเท่ากับ -999.0

2.1.7.2 ระบบพิกัดฉาก (OFFSETS) พิกัดฉากที่วัดจากทางขวามือของเส้นอ้างอิงมีค่าเป็นบวก วัดจากทางซ้ายมือของเส้นอ้างอิง มีค่าเป็นลบ

2.1.7.3 หน่วยที่ใช้ (UNITS) สามารถใช้ได้ทั้ง 2 ระบบ คือระบบ เมตริกและระบบอังกฤษ โดยปกติโปรแกรม จะถูกตั้งให้ทำงานโดยใช้หน่วยเมตริก เว้นเสียแต่จะระบุให้เป็น หน่วยอังกฤษ

2.1.7.4 หน่วยวัดมุม (ANGULAR NOTATION) สามารถใช้ได้ 5 แบบคือ

ก. องศา ลิบดา และฟิลิปดา ค่าฟิลิปดาสามารถกำหนดได้ถึงทศนิยมตำแหน่งที่สอง
ข. องศา และทศนิยมขององศา
ค. ใช้ได้รวมกันทั้งแบบ ก. และ ข.
ง. GRADS
จ. RADIANS

โปรแกรมสามารถตั้งให้ใช้หน่วยวัดมุมแบบใดก็ได้ โดยอาจทำการป้อนข้อมูลแบบหนึ่งและการแสดงผลเป็นอีกแบบหนึ่ง การป้อนข้อมูลและแสดงผลของหน่วยวัดมุม เขียนเหมือนกับที่ใช้กันตามปกติ ยกเว้นแบบ ก. ต้องเขียนติดกันและครบทุกตัวเช่น 27° 3' 9.27" ต้องเขียนเป็น 27103972 เป็นต้น

2.1.7.5 ความลาดเอียง (CROSSFALL) ความลาดเอียงมีค่าเป็นบวกในกรณีที่จุดที่ต้องการอยู่สูงกว่าจุดอ้างอิง และมีค่าเป็นลบถ้าอยู่ต่ำกว่าจุดอ้างอิง

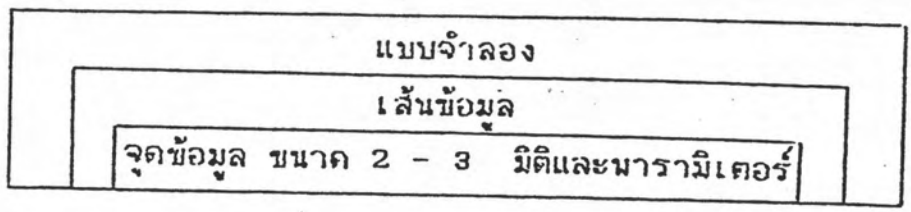
2.1.7.6 รัศมี (RADIUS) โค้งซ้ายมีค่าเป็นลบ โค้งขวามีค่าเป็นบวก ในกรณีที่เป็นเส้นตรง เมื่อแสดงผลจะกำหนดให้มีค่ารัศมีเท่ากับ 999999.900

2.1.8 หลักการณ์ในการเรียกข้อมูลมาใช้ แบบจำลองที่ใช้ในงานวิศวกรรมการทางจักประกอบด้วยค่าพิกัดของข้อมูลเป็นจำนวนมหาศาล ดังนั้นในการประมวลผลใดๆ บนแบบจำลอง มันจำเป็นต้องมีวิธีในการที่จะดึงข้อมูลเท่าที่จำเป็นเท่านั้นมาใช้ มิใช่ดึงข้อมูลทั้งหมดของแบบจำลองออกมาซึ่งจะเป็นการสิ้นเปลืองหน่วยความจำและเวลาในการประมวลผลโดยเปล่าประโยชน์

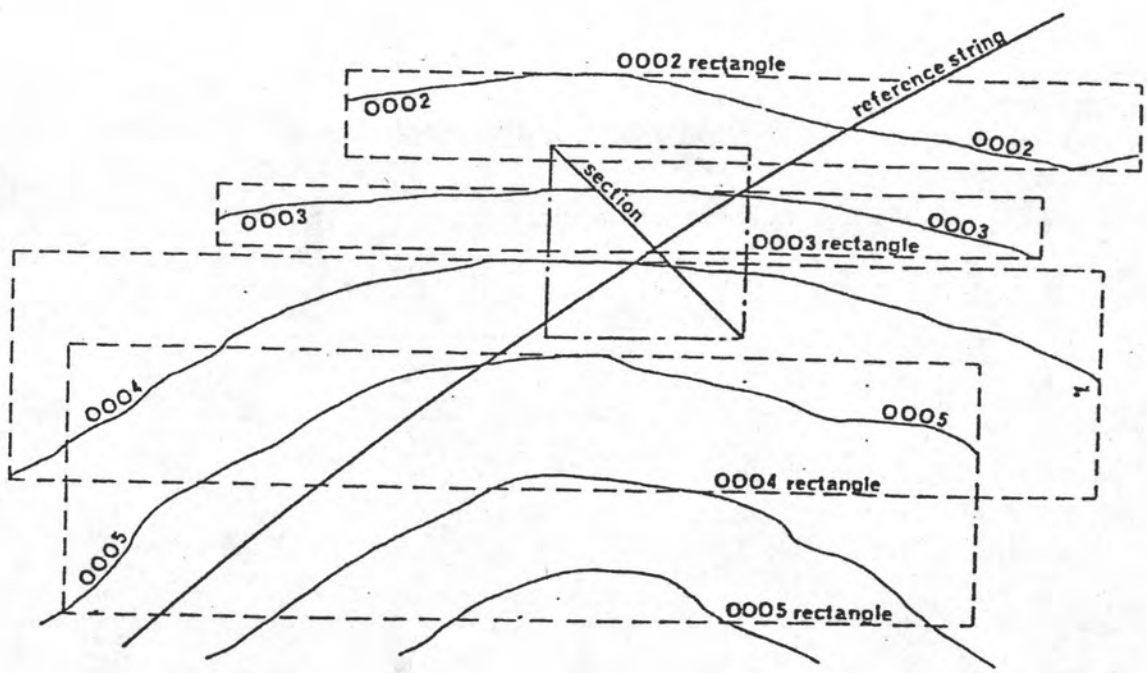
เส้นข้อมูลใดๆ ต้องมีค่าพิกัดสูงสุดและต่ำสุดทางแนวนอนของมันอยู่ จากค่าพิกัดทั้งสองค่านี้ โปรแกรมจะสร้างเป็นกล่องสี่เหลี่ยมที่บรรจุเส้นข้อมูลนี้อยู่ โดยที่กล่องบรรจุเส้นข้อมูลแต่ละกล่องนี้มีพื้นที่ซ้อนทับกันได้ เมื่อเรากำหนดขอบเขตของพื้นที่เราต้องการประมวลผล เช่น ต้องการสร้างรูปตัดของลักษณะพื้นดินเดิม โดยรูปตัดนี้ตั้งฉากออกจากเส้นข้อมูลที่ใช้อ้างอิง ดังแสดงในรูปที่ 2.2 MOSS จะนำค่าพิกัดสูงสุดและต่ำสุดทางแนวนอนของแนวรูปตัดที่ต้องการมาสร้างเป็นกล่องสี่เหลี่ยมของพื้นที่ที่พิจารณา หลังจากนั้นมันจะทำการค้นหาว่าในขอบเขตของกล่องที่พิจารณานั้นซ้อนทับกับกล่องบรรจุเส้นข้อมูลใดบ้าง จากรูปที่ 2.2 พบว่ากล่องที่พิจารณาซ้อนกับกล่องบรรจุเส้นข้อมูล 0002 0003 และ 0004 ดังนั้นมันจะดึงเฉพาะข้อมูลจากเส้นข้อมูล 0002 0003 และ 0004 เท่านั้นมาทำการประมวลผล ซึ่งจะเป็นการประหยัดหน่วยความจำและเวลาในการประมวลผลเป็นอย่างมาก

2.1.9 ประสิทธิภาพในการทำงานของโปรแกรม ตามทฤษฎีโปรแกรมสามารถสร้างและเก็บบันทึกเส้นและจุดข้อมูลได้โดยไม่จำกัดขนาด แต่ในการปฏิบัติเมื่อคำนึงถึงประสิทธิภาพและความถูกต้องในการทำงานผู้ใช้ควรสังวรณ์ถึงกฎในการปฏิบัติ 2 ข้อ ดังนี้

ก. จุดข้อมูลบนเส้นข้อมูล ไม่ควรมีจำนวนมากกว่า 100 จุด
ข. เส้นข้อมูลไม่ควรมีความยาวเกิน 1 กม.



รูปที่ 2.1 ลำดับขั้นของการเก็บข้อมูล



รูปที่ 2.2 การเรียกข้อมูลมาใช้

อย่างไรก็ดี กฎทั้งสองข้อนี้มิใช่ขีดจำกัดของมัน แต่เป็นข้อเสนอแนะในการใช้งานโดยคำนึงถึงประสิทธิภาพของมัน

2.2 การทำงานของโปรแกรม

2.2.1 ชนิดของคำสั่งที่ใช้ คำสั่งที่ใช้ในโปรแกรม จะแบ่งความสำคัญออกเป็น 2 ลำดับ คือกลุ่มของคำสั่ง ทางเลือกหลัก (MAJOR OPTIONS) และทางเลือกรอง (MINOR OPTIONS)

2.2.1.1 ทางเลือกหลัก (MAJOR OPTIONS) ใช้ในการกำหนดการทำงานบนแบบจำลองที่เป็นหลักใหญ่ของโปรแกรม เช่น การสร้างแบบจำลองใหม่ การปรับปรุงแก้ไขแบบจำลอง การคำนวณหาพื้นที่ของแบบจำลอง การออกแบบทางเรขาคณิตของโครงการ ฯ

คำสั่งที่ใช้ในทางเลือกหลักประกอบขึ้นจากตัวอักษรไม่เกิน 8 ตัวอักษร เช่น COPY CREATE INTERFAC(E) ฯ ทางเลือกหลักสามารถเลือกแบบจำลองมาทำงานพร้อมกันได้ไม่เกิน 2 แบบจำลอง เช่น ใช้เส้นข้อมูลจากแบบจำลองหนึ่งเป็นเส้นอ้างอิง เพื่อนำไปสร้างเส้นข้อมูลใหม่ในอีกแบบจำลองหนึ่งได้

ทางเลือกหลักแต่ละตัวมีขอบเขตในการทำงานดังนี้

ก. ปฏิบัติการบนแบบจำลอง
ข. ปฏิบัติการบนแฟ้มข้อมูล
ค. กำหนดแบบจำลองที่ต้องใช้ในการปฏิบัติการ

2.2.1.2 ทางเลืองรอง (MINOR OPTIONS) เป็นกลุ่มของคำสั่งที่ถูกเรียกใช้ในแต่ละทางเลือกหลัก ในแต่ละทางเลือกหลักจะมีกลุ่มคำสั่งทางเลือกรองของมันเองยกเว้นทางเลือกรองบางตัวที่ใช้ในการกำหนด

014163

พารามิเตอร์ต่างๆ สามารถใช้ได้กับทางเลือกหลักทุกตัว ทางเลือกรองจะถูกเรียกใช้เพื่อทำการปฏิบัติการบนเส้นข้อมูล เช่น ทำการปรับปรุงแก้ไขเส้นข้อมูล สร้างเส้นข้อมูลใหม่

คำสั่งที่ใช้ในทางเลือกรองจะประกอบขึ้นจากตัวเลข 3 ตัว โดยที่ทางเลือกรองแต่ละตัวมีขอบข่ายในการทำงานดังนี้

ก. ปฏิบัติการบนเส้นข้อมูล
ข. ปฏิบัติการบนจุดข้อมูล ของแต่ละเส้นข้อมูล
ค. จัดเตรียมวิธีการทำงานและข้อมูลสำหรับทางเลือกหลัก

2.2.2 <u>การป้อนข้อมูลและคำสั่งให้กับโปรแกรม</u> โปรแกรมคอมพิวเตอร์ทุกโปรแกรมต้องมีรูปแบบ (FORMAT) ที่แน่นอนสำหรับใช้ในการป้อนข้อมูลและคำสั่ง ใน MOSS ผู้ใช้ต้องทำการกำหนดทางเลือกหลักก่อนเสมอ เพื่อให้มันทราบว่าผู้ใช้ต้องการทำงานชนิดใดๆ และบนแบบจำลองใด หลังจากนั้นจึงทำการกำหนดทางเลือกรองเพื่อบอกลักษณะการทำงานในรายละเอียดของทางเลือกหลัก

รูปแบบการป้อนข้อมูลและคำสั่ง มี 2 รูปแบบคือ

ก. การป้อนข้อมูลแบบมาตรฐาน วิธีนี้ผู้ใช้ต้องทำการป้อนข้อมูลตามเขตข้อมูล (FIELD) โดยที่แต่ละเขตข้อมูลต้องมีตำแหน่งตามคอลัมน์ (COLUMN) ที่กำหนด

ข. การป้อนข้อมูลแบบอิสสระ วิธีนี้ผู้ใช้ต้องทำการป้อนข้อมูลตามเขตข้อมูลที่กำหนดเช่นกัน เพียงแต่เขตข้อมูลแต่ละตัวไม่จำเป็นต้องอยู่ในคอลัมน์ใดๆ ตายตัว เขตข้อมูลแต่ละชุดจะถูกคั่นด้วยเครื่องหมาย " , " ซึ่งเป็นการสะดวกในการทำงานมาก

2.2.3 ค่าที่ยอมให้ (TOLERANCE) เพื่อหลีกเลี่ยงการคำนวณค่าพิกัดของจุดข้อมูลที่มากเกินความจำเป็น หรือมิให้เกิดการซ้อนทับกันของจุดข้อมูล หรือเพื่อการหาจุดตัดกันบนเส้นข้อมูล มันจำเป็นที่จะต้องมีค่าที่ยอมให้เพื่อสามารถกำหนดแยกแยะจุดต่างๆ บนเส้นข้อมูล

ค่าที่ยอมให้ที่ถูกใช้ในโปรแกรม แบ่งออกเป็น 2 แบบคือ system tolerance และ location tolerance system tolerance จะใช้ในการคำนวณเปรียบเทียบข้อมูลทางเรขาคณิตทั้งหมดโดยจะถูกกำหนดให้มีค่าตายตัวเท่ากับ 0.001 หน่วย location tolerance ใช้ในการกำหนดค่าพิกัดในแกน x , y หรือ chainage เพื่อกำหนดจุดที่ต้องการ โดยที่โปรแกรมจะทำการกำหนดให้เท่ากับ 0.01 หน่วย โดยอัตโนมัติ เว้นเสียแต่ว่าจะกำหนดเป็นอย่างอื่น แต่ห้ามน้อยกว่า 0.001 หน่วย

วิธีการกำหนดใช้ค่าที่ยอมให้นี้มีหลายแบบดังต่อไปนี้

2.2.3.1 ค่าที่ยอมให้ในการสร้างเส้นข้อมูล โปรแกรมมีสมมุติฐานว่าระหว่างจุดข้อมูลใดๆ มันจะใช้เส้นตรงเชื่อมจุดข้อมูลเข้าด้วยกันเพื่อผลทางด้านความต่อเนื่องของเส้นข้อมูล ดังนั้นมันจำเป็นที่ต้องสร้างจุดข้อมูลให้เพียงพอบนเส้นข้อมูล ในกรณีของแนวทางนอน จำนวนของจุดบนเส้นข้อมูลจะถูกควบคุมด้วยระยะห่างที่มากสุดระหว่าง chord ถึง arc โดยที่ค่านี้จะกำหนดมีค่าเท่ากับ 0.1 หน่วยโดยอัตโนมัติ แต่ผู้ใช้สามารถแก้ไขได้ตามต้องการ

ในการสร้างเส้นข้อมูลในแนวนอน โปรแกรมจะทำการสร้างจุดตามช่วงระยะของ chainage ที่กำหนด แต่มันจะทำการตรวจสอบระยะห่างระหว่าง chord ถึง arc ด้วย ถ้าค่านี้เกินกว่าที่กำหนด (0.1 หน่วย) มันจะทำการสร้างจุดเพิ่มเพื่อให้ค่านี้อยู่ในเกณฑ์ที่ยอมได้ ดังแสดงในรูปที่ 2.3 จุดที่เพิ่มขึ้นเหล่านี้จะไม่แสดงออกในการรายงานผล (OUT PUT) แต่จะถูกเก็บบันทึกไว้ในเส้นข้อมูล

โดยปกติโค้งแนวดิ่งจะมีรัศมีโค้งมากกว่าโค้งแนวนอน ดังนั้น มันไม่จำเป็นต้องตรวจสอบค่านี้อีกเพราะเพียงพอแล้วในแนวทางนอน แต่อย่างไรก็ดีถ้ามีสิ่งผิดสังเกตขึ้นในแนวทางดิ่งที่ออกแบบไว้ ผู้ใช้สามารถตรวจสอบและแก้ไขโดยการทำแนวทางนอนซ้ำอีกครั้ง โดยลดค่าที่ยอมให้นี้น้อยลงไปอีกแต่ต้องไม่น้อยกว่า 0.001 หน่วย

2.2.3.2 ค่าที่ยอมให้ในการกำหนดตำแหน่งของจุดข้อมูล จุดข้อมูลบนเส้นข้อมูลต้องไม่มีค่าพิกัดซ้ำกัน จุดข้อมูลแต่ละจุดสามารถอยู่ใกล้กันได้ไม่เกิน(system tolerance 0.001 หน่วย) ไม่ว่าจะเป็นแกน x,y หรือ chainage การทำงานของทางเลือกหลักเกือบทั้งหมดสามารถประกันได้ว่าจะไม่เกิดผลเช่นนี้ขึ้น ยกเว้นในการสอดแทรกจุดข้อมูลโดยผู้ใช้ต้องระมัดระวังมิให้เกิดการซ้อนทับของจุดข้อมูลขึ้น

การค้นหาจุดใดๆ บนเส้นข้อมูล ไม่ว่าจะเป็นแกน x, y หรือ chainage จะถูกควบคุมโดย location tolerance ระยะ location tolerance จะถูกขยายออกจากจุดข้อมูลไปตามทิศทางของเส้นข้อมูลทั้งสองด้าน ดังแสดงในรูปที่ 2.4

ถ้าจุดข้อมูลอยู่ในช่วงของ location tolerance เดียวกัน มันจะทำการค้นหาและเลือกจุดแรกที่พบมาใช้ เช่น บนเส้นข้อมูลหนึ่งมีจุดข้อมูลแบบ exact point ที่ระยะ chainage 41.630, 41.638 และ 41.640 ถ้าผู้ใช้กำหนดว่าต้องการ chainage ที่ 41.638 มันจะได้ผลลัพธ์เป็นที่ 41.630 หรือถ้าต้องการ 41.640 ผลลัพธ์จะเป็น 41.638 วิธีแก้ไขคือเปลี่ยน location tolerance เป็น 0.001 หน่วย หรือกำหนดจุดโดยใช้ลำดับที่ของจุดข้อมูลแทนการใช้ chainage

2.2.3.3 ค่าที่ยอมให้ในการหาจุดตัดหรือชนกันของเส้นข้อมูล การคำนวณหาจุดตัดหรือชนระหว่างเส้นข้อมูล 2 เส้น หรือ ระหว่างเส้นข้อมูลกับเส้นตรงใดๆ ที่กำหนดขึ้น ใช้หลักการดังต่อไปนี้

ก. การหาจุดที่ชนกัน ดังแสดงในรูปที่ 2.5 ถ้าระยะ d มีค่าน้อยกว่า 0.001 มันจะพิจารณาว่า PQ จะชนกับ AB ที่จุด A

ข. การหาจุดที่ตัดกัน ดังแสดงในรูปที่ 2.6 ถ้าระยะ BI มีค่าน้อยกว่า location tolerance มันจะพิจารณาว่า RS ตัด BC ที่จุด B โดยที่ B เป็น exact point หรือถ้า BI มากกว่า location loterance มันจะพิจารณาว่าจุด I เป็น non exact point

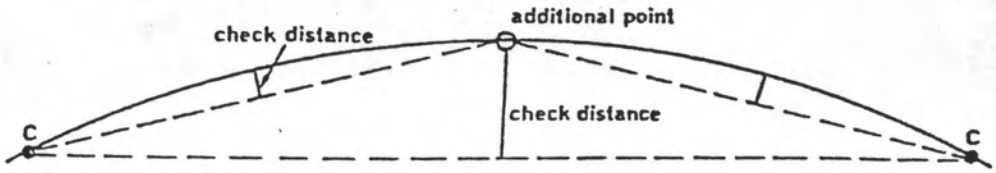
อย่างไรก็ดีพึงระลึกไว้ว่า ที่จุดหัวและท้ายของเส้นข้อมูล มันจะขยายออกไปเท่ากับ system tolerance เท่านั้น ดังแสดงในรูปที่ 2.7

2.2.4 <u>การกำหนดขอบเขตเส้นข้อมูลที่ใช้อ้างอิง</u> (Standard Point Reference Data) เมื่อต้องการสร้างเส้นข้อมูลใหม่โดยอ้างอิงจากเส้นข้อมูลเดิมที่มีอยู่ (reference string) ผู้ใช้ต้องระบุตำแหน่งของจุดที่หัวและท้ายของเส้นข้อมูลนั้น เพื่อกำหนดขอบเขตที่ต้องการจุดที่กำหนดอาจจะเป็น exact point หรือ non exact point ก็ได้ การกำหนดจุดที่หัวและท้ายของเส้นข้อมูลที่ใช้อ้างอิงมี 3 วิธีคือ

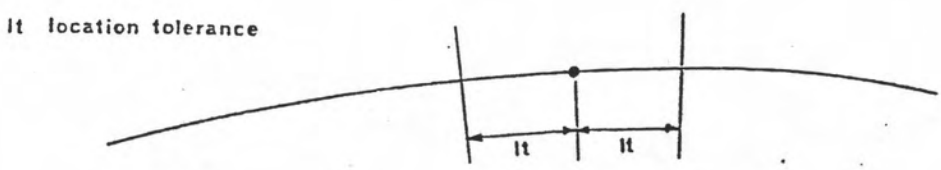
2.2.4.1 กำหนดโดยใช้ลำดับที่ของจุดข้อมูล บนเส้นข้อมูลใดๆ จะบอกลำดับที่ของจุดข้อมูลไว้โดยเรียงลำดับจากน้อยไปมาหามาก ลำดับที่ของจุดข้อมูลสามารถดูได้จากผลลัพธ์ (OUTPUT) ที่แสดงในการรายงานผล พึงจำไว้ว่าถ้ามีการลบหรือเพิ่มจุดข้อมูลลงบนเส้นข้อมูล ลำดับที่ของจุดข้อมูลจะทำการเรียงลำดับใหม่ทันทีโดยอัตโนมัติ

2.2.4.2 กำหนดโดยใช้ค่าพิกัดในแกน x, y ในกรณีที่ค่าพิกัดที่กำหนดอยู่ห่างจาก exact point ไม่เกิน location tolerance จุดข้อมูลที่กำหนดจะได้ผลเช่นเดียวกับหัวข้อ 2.2.4.1

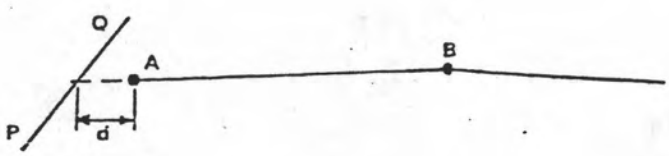
ในกรณีที่ค่าพิกัดที่กำหนดอยู่ห่างจาก exact point เกินกว่า location tolerance MOSS จะทำการลากเส้นตั้งฉาก



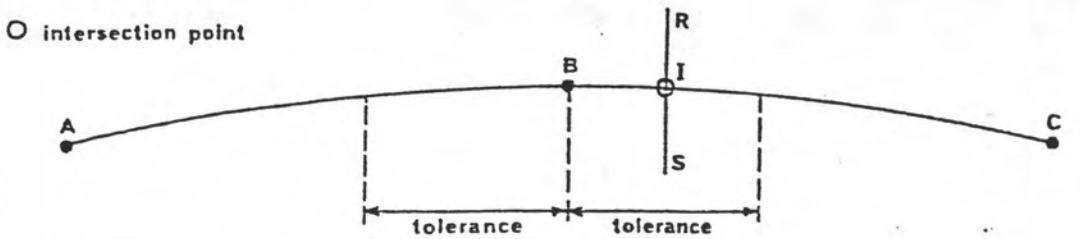
รูปที่ 2.3 การตรวจสอบระยะห่างระหว่าง chord ถึง arc



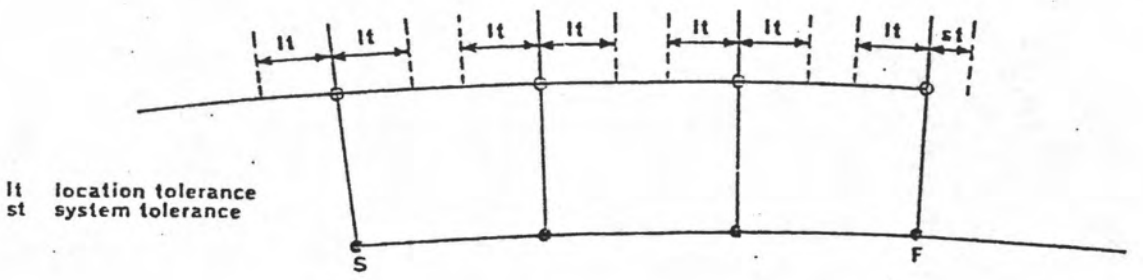
รูปที่ 2.4 แสดงขอบเขตของ location tolerance



รูปที่ 2.5 การหาจุดที่ชนกันระหว่างเส้นข้อมูล



รูปที่ 2.6 การหาจุดตัดกันของเส้นข้อมูล



รูปที่ 2.7 ระยะที่ขอมให้ที่หัวและท้ายของเส้นข้อมูล

หอสมุดกลาง สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

จากเส้นข้อมูลนั้นผ่านจุดที่กำหนด และใช้ค่าพิกัด ณ จุดตั้งฉากบนเส้นข้อมูลนั้นเป็นค่าพิกัดที่นำไปใช้

2.2.4.3 กำหนดโดยใช้ระยะ Chainage ใช้ได้เฉพาะการใช้เส้นข้อมูลหลักเป็นเส้นอ้างอิงเนื่องจากเส้นข้อมูลหลักประกอบด้วยค่าพิกัดถึง 3 มิติและพารามิเตอร์ซึ่งรวมถึง chainage ด้วย ดังนั้นถ้าใช้เส้นข้อมูลหลักเป็นเส้นอ้างอิงจึงสามารถใช้ระยะ chainage ได้ด้วย

2.2.5 <u>การแสดงผลของโปรแกรม</u> โปรแกรมสามารถแสดงผลได้ 2 แบบคือ แสดงผลด้วยรูปภาพและแสดงผลด้วยข้อมูล การแสดงผลด้วยรูปภาพสามารถแสดงผลทางเครื่อง PLOTTER หรือจอภาพแบบกราฟฟิก (GRAPHIC TERMINAL) สำหรับการแสดงผลด้วยข้อมูลสามารถแสดงผลทางเครื่องพิมพ์ (PRINTER) หรือ จอภาพ (TERMINAL)

2.2.5.1 การแสดงผลด้วยรูปภาพ การเก็บข้อมูลในลักษณะเส้นข้อมูลทำให้โปรแกรมได้เปรียบโปรแกรมอื่นอย่างมากตรงที่มันทำงานในลักษณะเดียวกับการเขียนแบบของมนุษย์ ดังนั้นมันจึงสามารถแสดงผลทางรูปภาพได้อย่างมีประสิทธิภาพ

ผู้ใช้ต้องกำหนดชื่อของเส้นข้อมูลที่พิจารณาและขอบเขตของพื้นที่ที่ต้องการ ผลลัพธ์สามารถแสดงได้หลายแบบทั้ง รูปแปลน รูปตัดขวาง และรูปสามมิติ (PERSPECTIVE VIEWS) สำหรับรูปแปลนโปรแกรมจะใช้ชนิดของเส้นต่างชนิดกันตามชื่อของเส้นข้อมูลที่แบ่งไว้ดังหัวข้อ 2.1.5 ในกรณีที่ต้องการชนิดของเส้นที่แตกต่างไปจากนี้สามารถกำหนดขึ้นใหม่ได้เช่นกัน

2.2.5.2 การแสดงผลด้วยข้อมูล การแสดงผลด้วยรูปภาพไม่สามารถบอกความสัมพันธ์ทางเรขาคณิตและข้อมูลของเส้นข้อมูลที่แน่นอนได้ ดังนั้นมันต้องสามารถแสดงผลของข้อมูลบนแบบจำลองได้อย่างแน่นอนด้วย การแสดงผลโดยข้อมูลแบ่งออกแเป็น 2 ประเภทใหญ่ คือ

ก. รายงานสถานภาพของแบบจำลองและเส้นข้อมูล เช่น ชื่อของแบบจำลองที่มีอยู่ในแฟ้มข้อมูล ข้อมูลของเส้นข้อมูลชนิดต่างๆ ตามที่ได้กำหนดไว้ในหัวข้อ 2.1.3 จำนวนและชื่อของเส้นข้อมูลที่มีอยู่ในแบบจำลอง

ข. รายงานสภาพทางเรขาคณิต ใช้ในการแสดงผลจุดตัดของเส้นข้อมูลในแต่ละแบบจำลอง

2.2.6 <u>การรายงานความผิดพลาดที่เกิดขึ้น</u> ความผิดพลาดอันอาจเกิดจาก การป้อนข้อมูล การทำงานที่ผิดจากปกติ หรือเนื่องมาจากความบกพร่องของมันเอง ต้องสามารถตรวจสอบและรายงานให้ผู้ใช้ทราบได้ โปรแกรมแบ่งชนิดของการรายงานความผิดพลาดที่เกิดขึ้นเป็น 3 แบบคือ

2.2.6.1 การเตือนว่าผิดปกติ รหัสที่ใช้แจ้งการเตือนแบบนี้ขึ้นต้นด้วยตัวอักษร W ตามด้วยตัวเลขอีก 3 ตัว เมื่อมีการเตือนเกิดขึ้นมันจะยังไม่หยุดการทำงาน แต่เป็นเพียงแจ้งให้ผู้ใช้ทราบว่ามีการทำงานที่ผิดปกติ เช่น เมื่อผู้ใช้ทำการตั้งค่าพารามิเตอร์ใหม่ หรือโปรแกรมทำการขยายขอบเขตการทำงานโดยอัตโนมัติ

2.2.6.2 การแจ้งว่าเกิดความผิดพลาดขึ้นในโปรแกรม รหัสที่ใช้จะขึ้นต้นด้วยตัวอักษา E ตามด้วยตัวเลขอีก 3 ตัว เมื่อมีการแจ้งว่าเกิดความผิดพลาดขึ้น โปรแกรมจะหยุดการทำงานทั้งหมด ยกเว้นในกรณีที่ผู้ใช้สั่งให้มันทำการรายงานผล มันจะข้ามการทำงานอื่นใดทั้งหมด โดยทำเฉพาะส่วนของรายงานผลเท่านั้น ถ้าความผิดพลาดนั้นมิได้อยู่ในส่วนของการรายงานผล

2.2.6.3 การแจ้งว่าเกิดความผิดพลาดขึ้นในระบบคอมพิวเตอร์ รหัสที่ใช้คือ E99 เมื่อมีการแจ้งดังกล่าว ทุกส่วนของโปรแกรมจะหยุดการทำงาน จำเป็นที่จะต้องทำการปรับปรุงแก้ไขระบบก่อนที่จะให้มันทำงานต่อไปได้ เช่น จัดรูปแบบของแฟ้มข้อมูลใหม่เป็นต้น