

การจินตทัศน์ข้อมูลในระหว่างการพัฒนาการของผลเฉลย
ที่ได้จากกำหนดการเชิงพีชคณิต

นาย ยอดธง รอดแก้ว



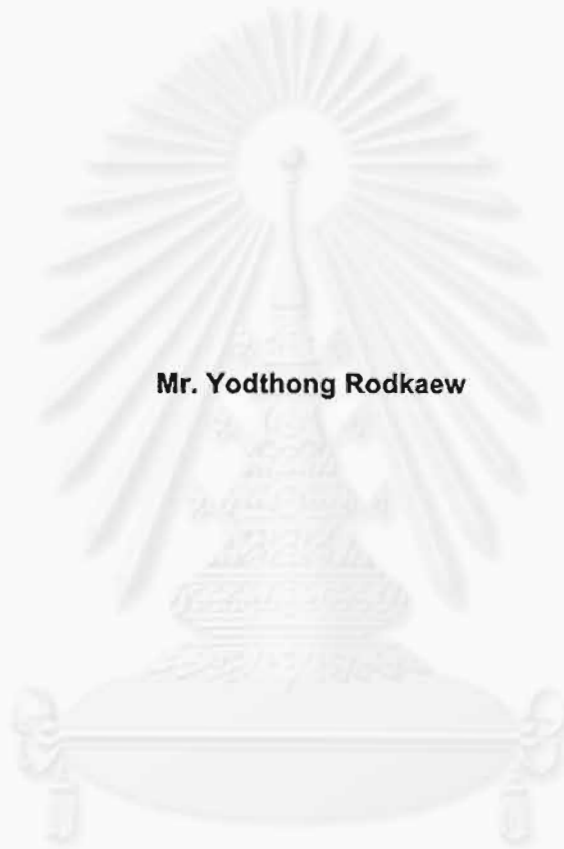
วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
สาขาวิชาวิทยาศาสตร์คอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

ปีการศึกษา 2542

ISBN 974-334-207-9

ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

**VISUALIZATION OF DATA DURING AN EVOLUTION OF SOLUTIONS
FROM GENETIC PROGRAMMING**



Mr. Yodthong Rodkaew

**A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Science in Computer Science**

Department of Computer Engineering

Faculty of Engineering

Chulalongkorn University

Academic Year 1999

ISBN 974-334-207-9

หัวข้อวิทยานิพนธ์	การจินตทัศน์ข้อมูลในระหว่างการพัฒนาการของผลเฉลย ที่ได้จากกำหนดการเชิงพันธุกรรม
โดย	นาย ยอดธง รอดแก้ว
ภาควิชา	วิศวกรรมคอมพิวเตอร์
อาจารย์ที่ปรึกษา	ผู้ช่วยศาสตราจารย์ ดร. ประภาส จงสถิตยวัฒน์

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้บัณฑิตวิทยาลัย
เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรบัณฑิต

..... *สมศักดิ์ ปัญญาแก้ว* คณะบดีคณะวิศวกรรมศาสตร์
(ศาสตราจารย์ ดร. สมศักดิ์ ปัญญาแก้ว)

คณะกรรมการสอบวิทยานิพนธ์

..... *สมชาย ประสิทธิ์จตุระกุล* ประธานกรรมการ
(ผู้ช่วยศาสตราจารย์ ดร. สมชาย ประสิทธิ์จตุระกุล)

..... *ประภาส จงสถิตยวัฒน์* อาจารย์ที่ปรึกษา
(ผู้ช่วยศาสตราจารย์ ดร. ประภาส จงสถิตยวัฒน์)

..... *นงลักษณ์ โควาวิสารัช* กรรมการ
(ผู้ช่วยศาสตราจารย์ นงลักษณ์ โควาวิสารัช)

..... *สืบสกุล พิภพมงคล* กรรมการ
(อาจารย์ ดร. สืบสกุล พิภพมงคล)

ยอดธง รอดแก้ว : การจินตทัศน์ข้อมูลในระหว่างการวิวัฒนาการของผลเฉลยที่ได้จาก
กำหนดการเชิงพันธุกรรม (VISUALIZATION OF DATA DURING AN
EVOLUTION OF SOLUTIONS FROM GENETIC PROGRAMMING)
อาจารย์ที่ปรึกษา ผศ.ดร.ประภาส จงสดีดีย์วัฒนา, 106 หน้า, ISBN 974-334-207-9

วิทยานิพนธ์ฉบับนี้เสนอการจินตทัศน์ข้อมูลในระหว่างการวิวัฒนาการของผลเฉลยที่ได้จากกำหนดการเชิงพันธุกรรม สามารถแสดงการจินตทัศน์ข้อมูลในรูปแบบต่างๆ ดังนี้ โครงสร้างคำตอบ การจำลองการทำงานของคำตอบ การวิวัฒนาการโครงสร้างคำตอบ ค่าเฉลี่ยค่าความเหมาะสมที่ได้จากตัวปฏิบัติการพันธุกรรมต่างๆ เส้นทางที่ถูกใช้งานภายในโครงสร้างต้นไม้ของคำตอบ โปรแกรมทำงานบน X-windows ภายใต้ระบบปฏิบัติการ Linux พัฒนาโดยใช้ภาษา C ร่วมกับคลังโปรแกรม GTK และ OpenGL การจินตทัศน์ด้วยโปรแกรมดังกล่าวช่วยให้ผู้ใช้งานมีความสะดวกในการนำข้อมูลกำหนดการเชิงพันธุกรรมมาแสดงผล มีระบบรายการเลือกช่วยให้การใช้งานง่ายและสามารถแสดงผลได้ดีในรูปแบบ 3 มิติ จากการนำข้อมูลจากงานวิจัยเดิมเรื่องเกี่ยวกับกำหนดการเชิงพันธุกรรมมาจินตทัศน์ทำให้เข้าใจผลเฉลยที่ได้จากกำหนดการเชิงพันธุกรรมดีขึ้น ยกตัวอย่างเช่น ตัวปฏิบัติการพันธุกรรม ชนิดต่างๆ มีผลต่อการวิวัฒนาการต่างกัน และส่วนของผลเฉลยบางส่วนมิได้ถูกใช้งาน ความถี่ของการทำงานของผลเฉลยสัมพันธ์กับโครงสร้างของคำตอบ เป็นต้น

ภาควิชา วิศวกรรมคอมพิวเตอร์ ลายมือชื่อนิสิตยอดธง รอดแก้ว.....
สาขาวิชา วิทยาศาสตร์คอมพิวเตอร์ ลายมือชื่ออาจารย์ที่ปรึกษา **ประภาส จงสดีดีย์วัฒนา**
ปีการศึกษา 2542 ลายมือชื่ออาจารย์ที่ปรึกษารวม-.....

YODTHONG RODKAEW : VISUALIZATION OF DATA DURING AN EVOLUTION OF SOLUTIONS FROM GENETIC PROGRAMMING. THESIS ADVISOR : ASSIST. PROF. PRABHAS CHONGSTITVATANA. Ph.D. 106 pp. ISBN 974-334-207-9

This thesis describes a visualization of data during an evolution of solutions from genetic programming. It can display the data in various forms: a structure of solution, a simulation of solution, an evolution of solution's structure, an averaging fitness value from genetic operators, useful path inside a tree structure of solution. The program works on X-windows under Linux operating system. The development uses C language with GTK and OpenGL library. The visualization from this program helps a user to present the data from genetic programming. It has a menu system that is easy to use and can present data in 3D. Using this program to visualize the data from a previous research related to genetic programming helps the understanding of the solutions, for example, different genetic operators affected the evolution differently, some part of solutions was not used, the frequency of the solution executions was related to its structure.



สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ภาควิชา	วิศวกรรมคอมพิวเตอร์	ลายมือชื่อนิสิต
สาขาวิชา	วิทยาศาสตร์คอมพิวเตอร์	ลายมือชื่ออาจารย์ที่ปรึกษา
ปีการศึกษา	2542	ลายมือชื่ออาจารย์ที่ปรึกษาร่วม



กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงไปได้ด้วยความช่วยเหลืออย่างดียิ่งของผู้ช่วยศาสตราจารย์ ดร.ประภาส จงสถิตย์วัฒนา อาจารย์ที่ปรึกษาวิทยานิพนธ์ ซึ่งท่านได้ให้คำแนะนำ ช้อปรับปรุ่ และข้อคิดเห็นต่างๆ ในการทำวิจัยด้วยดีมาตลอดมา และกรุณาให้คำแนะนำในเรื่องอื่นๆ ที่มีประโยชน์ในระหว่างการทำวิจัยอีกด้วย

ขอบคุณ คุณชนิษฐา หุตะเมขลิน ซึ่งได้ให้คำแนะนำสำหรับการแก้ไขปัญหาค้นสำเร็จลุล่วงไปได้ด้วยดี

ท้ายนี้ ผู้วิจัยขอกราบขอบพระคุณบิดาและมารดา ซึ่งสนับสนุนในด้านการเงินและขอใจน้อมที่ให้กำลังใจแก่ผู้วิจัยเสมอมาจนสำเร็จการศึกษา

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	ง
บทคัดย่อภาษาอังกฤษ.....	จ
กิตติกรรมประกาศ.....	ฉ
สารบัญ.....	ช
สารบัญภาพ.....	ฎ
สารบัญตาราง.....	ณ
บทที่	
1 บทนำ.....	1
1.1 ความสำคัญและความเป็นมาของปัญหา.....	1
1.2 งานวิจัยที่เกี่ยวข้อง.....	6
1.3 วัตถุประสงค์.....	7
1.4 ขอบเขตของงานวิจัย.....	7
1.5 ประโยชน์ที่คาดว่าจะได้รับ.....	7
1.6 เนื้อหาในวิทยานิพนธ์นี้.....	7
1.7 งานตีพิมพ์.....	8
2 กำหนดการเชิงพันธุกรรมสำหรับควบคุมแขนหุ่นยนต์.....	9
2.1 ข้อกำหนดของแขนหุ่นยนต์.....	9
2.1.1 แขนหุ่นยนต์.....	9
2.1.2 สนามทดลอง.....	10
2.2 ข้อกำหนดของกำหนดการเชิงพันธุกรรม.....	10

สารบัญ (ต่อ)

	หน้า
2.2.1 ส่วนฟังก์ชัน.....	11
2.2.2 ส่วนเทอมินอล.....	11
2.2.3 การวัดค่าความเหมาะสม.....	12
2.2.4 ส่วนพารามิเตอร์ของกำหนดการเชิงพันธุกรรม.....	13
3 การใช้งานระบบการจินตทัศน์ข้อมูล.....	15
3.1 เข้าสู่ระบบ GPVIS	15
3.2 แสดงข้อมูลคำตอบที่ได้จากกำหนดการเชิงพันธุกรรม.....	16
3.3 แสดงโครงสร้างของคำตอบ.....	17
3.4 แสดงความสัมพันธ์ของคำตอบเริ่มต้น.....	18
3.5 แสดงการเคลื่อนไหวแขนหุ่นยนต์	19
3.6 แสดงข้อมูลกำหนดการเชิงพันธุกรรมในรูปแบบ 3 มิติ.....	19
3.7 การจัดเก็บภาพ.....	20
3.8 ออกจากโปรแกรม	20
4 โปรแกรมจินตทัศน์ข้อมูลกำหนดการเชิงพันธุกรรม.....	21
4.1 โปรแกรม VGL.....	21
4.2 ส่วนต่างๆ ของโปรแกรม VGL	22
4.2.1 ส่วนที่ 1 ส่วนสร้างและจัดเก็บข้อมูลกำหนดการเชิงพันธุกรรม.....	22
4.2.2 ส่วนที่ 2 ส่วนเรียกข้อมูลสำหรับการจินตทัศน์.....	23
4.2.3 ส่วนที่ 3 ส่วนควบคุมและประมวลผลข้อมูลสำหรับการจินตทัศน์.....	24
4.2.4 ส่วนที่ 4 ส่วนการสร้างการจินตทัศน์.....	27
4.3 โปรแกรม VGLCON.....	28
4.4 โปรแกรม CONSOLE.....	29
5 การจินตทัศน์ข้อมูลกำหนดการเชิงพันธุกรรมด้วยระบบจินตทัศน์ข้อมูล GPVIS	31
5.1 การสร้างข้อมูลกำหนดการเชิงพันธุกรรม	31
5.2 การจินตทัศน์ข้อมูลกำหนดการเชิงพันธุกรรม	32

สารบัญ (ต่อ)

	หน้า
5.3 การเลือกสภาพแวดล้อม.....	33
5.4 การเลือกคำตอบที่ได้จากกำหนดการเชิงพันธุกรรม.....	34
5.4.1 การเลือกคำตอบจากโปรแกรม VGL.....	35
5.4.2 การเรียกใช้เมนูเลือกคำตอบกำหนดการเชิงพันธุกรรม.....	36
5.4.3 การแสดงรายละเอียดข้อมูลคำตอบที่ได้จากกำหนดการเชิงพันธุกรรม.....	37
5.5 การแสดงคำตอบที่ได้จากกำหนดการเชิงพันธุกรรม ในรูปแบบโครงสร้างต้นไม้.....	39
5.6 การจำลองการเคลื่อนไหวของแขนหุ่นยนต์จากคำตอบที่ได้จากกำหนดการเชิงพันธุกรรม	40
5.7 การแสดงและเลือกคำตอบจากความสัมพันธ์ระหว่างคำตอบที่เกิดจากการ วิวัฒนาการในกำหนดการเชิงพันธุกรรม.....	41
5.8 การแสดงกระบวนการวิวัฒนาการของคำตอบในกำหนดการเชิงพันธุกรรม.....	43
5.9 การแสดงข้อมูลกำหนดการเชิงพันธุกรรมในรูปแบบ 3 มิติ.....	44
5.10 การตั้งค่าสำหรับกรองข้อมูลสำหรับการแสดงผลกำหนดการเชิงพันธุกรรมในรูปแบบ 3 มิติ	47
5.10.1 การตั้งค่าสี	47
5.10.2 การกำหนดค่าสีแบบ Hierarchical	50
5.10.3 การตั้งค่าสำหรับการกรองข้อมูล	51
5.10.4 ค่าเฉลี่ยของข้อมูล.....	52
5.10.5 ค่ากำหนดรูปแบบการแสดงผล	53
5.11 การปรับเปลี่ยนสถานะการแสดงผล และ การเก็บภาพที่ปรากฏในระบบ GPVIS เป็นไฟล์.....	54
6 สิ่งที่น่าสนใจจากการจินตทัศน์ข้อมูลในระหว่างการพัฒนาการที่ได้มาจากกำหนดการเชิง พันธุกรรม.....	56
6.1 การจินตทัศน์ฟังก์ชันการพัฒนาการกำหนดการเชิงพันธุกรรม.....	56
6.2 โครงสร้างของคำตอบที่ได้จากกำหนดการเชิงพันธุกรรม.....	64
6.3 ต้นไม้และเส้นทางที่ถูกใช้งาน.....	67

สารบัญ (ต่อ)

	หน้า
6.4 การไขว้เปลี่ยนและการกลายพันธุ์.....	72
7 สรุป	77
7.1 เกี่ยวกับงานวิจัย	77
7.2 แนวทางในการพัฒนาโปรแกรมต่อไปสำหรับอนาคต	77
7.2.1 ข้อกำหนดของระบบ.....	77
7.2.2 ส่วนเพิ่มขยายระบบการจินตทัศน์.....	78
7.2.3 การจินตทัศน์กำหนดการเชิงพันธุกรรมอื่นๆ.....	78
7.2.4 การจินตทัศน์บน WEB-Browser	78
7.2.5 มาตรฐานสำหรับข้อมูลกำหนดการเชิงพันธุกรรม	78
รายการอ้างอิง.....	80
บรรณานุกรม.....	82
ภาคผนวก	
ก รายละเอียดเกี่ยวกับโปรแกรม	83
ก.1 แผ่นลัด	83
ก.2 การติดตั้งระบบ GPVIS จากแผ่น CD-ROM	84
ก.3 ระบบเพิ่มข้อมูลในระบบ GPVIS	84
ข วิธีการเพิ่มฟังก์ชันภายในระบบ GPVIS.....	86
ข.1 ข้อมูลสำหรับใช้ในการสร้างภาพ 3 มิติ	86
ข.2 การจัดวางข้อมูล	88
ข.3 การจัดเรียงข้อมูล	90
ข.4 ปัญหาเกี่ยวกับการสร้างเส้นเชื่อมข้อมูล.....	91
ข.5 การเฉลี่ยข้อมูล	91
ข.6 การเพิ่มฟังก์ชัน.....	96

สารบัญ (ต่อ)

	หน้า
ค โครงสร้างข้อมูลสำหรับการจินตทัศน์ในระบบ GPVIS	98
ค.1 โครงสร้างไฟล์ DATANN.VGP	98
ค.2 โครงสร้างกล่องข้อมูลรุ่น.....	99
ค.3 โครงสร้างกล่องข้อมูลกำหนดการเชิงพันธุกรรม.....	99
ค.4 โครงสร้างของข้อมูลรอบและรุ่น	99
ค.5 โครงสร้างของข้อมูลคุณสมบัติ	100
ค.6 โครงสร้างของข้อมูลโปรแกรม	101
ค.7 ตัวอย่างข้อมูลจากไฟล์.....	101
ค.8 ตัวโปรแกรม.....	102
ง คำศัพท์คอมพิวเตอร์ที่ใช้ในวิทยานิพนธ์.....	103
ประวัติผู้วิจัย.....	106

สารบัญภาพ

ภาพที่	หน้า
1 การเรียงลำดับตัวเลข.....	2
2 การเรียงลำดับตัวเลขโดยแสดงแบบเส้น.....	2
3 การวิวัฒนาการของประชากรและจำนวนสายพันธุ์.....	5
4 แสดงสภาพแวดล้อมที่ 1.....	10
5 หน้าต่างการใช้งานระบบ GPVIS.....	15
6 กล่องโต้ตอบสำหรับการเลือกคำตอบที่ได้จากกำหนดการเชิงพันธุกรรม.....	16
7 หน้าจอระบบโดยรวมเมื่อแสดงข้อมูลคำตอบ.....	17
8 โครงสร้างของคำตอบ.....	17
9 ความสัมพันธ์ของคำตอบ.....	18
10 การจำลองการเคลื่อนไหวแขนหุ่นยนต์.....	19
11 กล่องโต้ตอบข้อมูลกำหนดการเชิงพันธุกรรมในรูปแบบ 3 มิติ.....	19
12 แสดงความสัมพันธ์ข้อมูลกำหนดการเชิงพันธุกรรมในรูปแบบ 3 มิติ.....	20
13 ระบบ GPVIS.....	21
14 โปรแกรม VGL ส่วนที่ 1 ส่วนสร้างและจัดเก็บข้อมูลกำหนดการเชิงพันธุกรรม.....	22
15 โปรแกรม VGL ส่วนที่ 2 ส่วนเรียกข้อมูลสำหรับการจินตทัศน์.....	23
16 โปรแกรม VGL ส่วนที่ 3 ส่วนควบคุมและประมวลผลข้อมูลสำหรับการจินตทัศน์.....	25
17 โปรแกรม VGL ส่วนที่ 4 ส่วนการสร้างการจินตทัศน์.....	27
18 โปรแกรม VGLCON และส่วนเชื่อมต่อ.....	28
19 โปรแกรม CONSOLE และส่วนเชื่อมต่อ.....	29
20 แสดงหน้าจอการเลือกสภาพแวดล้อม.....	34
21 แสดงหน้าจอการเลือกคำตอบที่ได้จากกำหนดการเชิงพันธุกรรม.....	35
22 แสดงหน้าจอการเลือกคำตอบที่ได้จากกำหนดการเชิงพันธุกรรมโดยไม่เรียงลำดับค่า fitness ของคำตอบกำหนดการเชิงพันธุกรรมในแต่ละรุ่น.....	36
23 แสดงกล่องโต้ตอบสำหรับการเลือกคำตอบที่ได้จากกำหนดการเชิงพันธุกรรม.....	37
24 กล่องโต้ตอบรายละเอียดของคำตอบที่ได้จากกำหนดการเชิงพันธุกรรม.....	37
25 โครงสร้างของคำตอบกำหนดการเชิงพันธุกรรมที่ดีที่สุดที่รุ่นสุดท้าย ในสภาพแวดล้อมที่ 1180000.....	39

สารบัญญภาพ (ต่อ)

ภาพที่	หน้า
26 แสดงการเคลื่อนไหวของแขนหุ่นยนต์ (เรียงจากซ้ายไปขวา, บนลงล่าง) ควบคุมจากคำตอบกำหนดการเชิงพันธุกรรมที่ดีที่สุดในเรื่องสุดท้าย ในสภาพแวดล้อมที่ 1180000	41
27 ความสัมพันธ์ในระหว่างการวิวัฒนาการของคำตอบที่ดีที่สุด.....	42
28 ความสัมพันธ์ในระหว่างการวิวัฒนาการของคำตอบที่ดีที่สุด (หมายเลข 18) ในเรื่องสุดท้าย ในสภาพแวดล้อมที่ 1180000.....	42
29 กระบวนการวิวัฒนาการโครงสร้างคำตอบในกำหนดการเชิงพันธุกรรม	44
30 กราฟ 3 มิติแสดงความสัมพันธ์ของข้อมูลคำตอบจากกำหนดการเชิงพันธุกรรมระหว่าง Generation - Node - Fitness.....	45
31 (a) กล้องโต้ตอบข้อมูลกำหนดการเชิงพันธุกรรมในรูปแบบ 3 มิติ (ซ้าย) และ (b) ส่วนกำหนดประเภทของข้อมูลที่จะนำมาแสดงผลในแต่ละแกน (ขวา).....	46
32 การตั้งค่าสีในรูปแบบต่างๆ	49
33 ผลจากการกำหนดค่าสีแบบ Hierarchical.....	50
34 กำหนดค่าสำหรับการกรองข้อมูล	51
35 ผลลัพธ์ที่ได้จากการจินตทัศน์เมื่อกำหนดค่าสำหรับการกรองข้อมูลที่แตกต่างกัน	51
36 กำหนดค่าเฉลี่ยข้อมูล	52
37 ผลลัพธ์ที่ได้จากการจินตทัศน์เมื่อกำหนดค่าเฉลี่ยข้อมูลแตกต่างกัน.....	52
38 กำหนดรูปแบบการแสดงผล.....	53
39 ผลลัพธ์ที่ได้จากการจินตทัศน์เมื่อกำหนดรูปแบบการแสดงผลที่แตกต่างกัน	53
40 ความสัมพันธ์ของคำตอบระหว่างรุ่นของคำตอบ, ค่าความเหมาะสม และหมายเลขคำตอบ	57
41 การเลือกคำตอบหมายเลข 1 ถึง 40.....	57
42 หมายเลขคำตอบ 1 ถึง 40 บนแกนความสัมพันธ์ของคำตอบระหว่างรุ่นของคำตอบ, ค่าความเหมาะสม และหมายเลขคำตอบ.....	58
43 หมายเลขคำตอบ 41 ถึง 200 บนแกนความสัมพันธ์ของคำตอบระหว่างรุ่นของคำตอบ, ค่าความเหมาะสม และหมายเลขคำตอบ.....	58
44 หมายเลขคำตอบ 201 ถึง 300 บนแกนความสัมพันธ์ของคำตอบระหว่างรุ่นของคำตอบ, ค่าความเหมาะสม และหมายเลขคำตอบ	59

สารบัญภาพ (ต่อ)

ภาพที่	หน้า
45 หมายเลขคำตอบ 301 ถึง 400 บนแกนความสัมพันธ์ของคำตอบระหว่างรุ่นของคำตอบ, ค่าความเหมาะสม และหมายเลขคำตอบ	59
46 หมายเลขคำตอบ 1 ถึง 40 โดยเฉลี่ย.....	60
47 หมายเลขคำตอบ 41 ถึง 200 โดยเฉลี่ย	60
48 หมายเลขคำตอบ 201 ถึง 300 โดยเฉลี่ย	60
49 หมายเลขคำตอบ 301 ถึง 400 โดยเฉลี่ย	61
50 ผลของฟังก์ชันวิวัฒนาการเปรียบเทียบระหว่างภาพที่ 46-49	61
51 ความเพียรพยายามเชิงคำนวณเปรียบเทียบในกรณีที่ใช้และไม่ใช้การกลายพันธุ์แบบด้อยอด	62
52 การจินตทัศน์จำนวนลูกของคำตอบ.....	63
53 การจินตทัศน์โครงสร้างข้อมูลคำตอบในแบบต้นไม้เดิม(ซ้าย) กับแบบทรงกลม(ขวา)	65
54 การจินตทัศน์โครงสร้างข้อมูลคำตอบในแบบต้นไม้(ซ้าย) กับแบบทรงกลม(ขวา)	65
55 โครงสร้างต้นไม้แบบทรงกลมสมบูรณ์ (ซ้าย) เปรียบเทียบกับแบบไม่สมบูรณ์ (ขวา)	66
56 โครงสร้างต้นไม้ 3 มิติแบบสมบูรณ์(ซ้าย) และแบบไม่สมบูรณ์(ขวา).....	67
57 เส้นทางที่เกิดจากสถานะการเรียกใช้บัลลูน	67
58 เส้นทางที่ถูกใช้งานในต้นไม้จากคำตอบที่ 18 รุ่นที่ 10	71
59 แสดงตัวอย่างการไขว้เปลี่ยน (crossover) จากคำตอบ (ก) เป็นคำตอบ (ข).....	72
60 แสดงอัตราส่วนระหว่างบัฟฟังก์ชันและเทอมินอลจากคำตอบทั้งหมดในสภาพแวดล้อม 1180000	73
61 ความสัมพันธ์ในระหว่างการวิวัฒนาการของคำตอบที่ดีที่สุด (หมายเลข 18) ในรุ่นสุดท้าย ในสภาพแวดล้อมที่ 1180000	74
62 โครงสร้างคำตอบที่ 18 รุ่นที่ 10	75
63 สารบบ GPVIS.....	84
64 ลักษณะของเส้นเชื่อมระหว่างจุด	88
65 การจัดวางข้อมูลที่มีด้านยาว 4 จุด	89
66 การจัดวางข้อมูลที่มีด้านยาว 5 จุด	89
67 การจัดวางข้อมูลที่มีด้านยาว 4 จุดซึ่งไม่สัมพันธ์กับตัวแปร	89

สารบัญญภาพ (ต่อ)

ภาพที่	หน้า
68 การจัดเรียงข้อมูล	90
69 กำหนดค่าเฉลี่ยข้อมูล	91
70 การเฉลี่ยที่ไม่เหมาะสม	93
71 การเฉลี่ยที่กำหนดขอบตามความยาวด้าน	93
72 กำหนดค่าเฉลี่ยข้อมูล	96
73 ต้นไม้จากสายอักขระ oecdpijgk	102



สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

สารบัญตาราง

ตารางที่	หน้า
1 รายละเอียดของสภาพแวดล้อมที่ 1	10
2 ฟังก์ชันที่ใช้ในกำหนดการเชิงพันธุกรรมของชัยวัฒน์ เจษฎาปกรณ์	11
3 เทอมินอลที่ใช้ในกำหนดการเชิงพันธุกรรมของชัยวัฒน์ เจษฎาปกรณ์.....	12
4 ตัวแปรในการคำนวณค่าความเหมาะสมของชัยวัฒน์ เจษฎาปกรณ์.....	13
5 ค่าพารามิเตอร์กำหนดการเชิงพันธุกรรมที่ปรับปรุงจากงานวิจัยของชัยวัฒน์ เจษฎาปกรณ์	14
6 รายละเอียดของคำตอบที่ได้จากกำหนดการเชิงพันธุกรรม	38
7 สีของฟังก์ชันและเทอมินอล	39
8 แสดงค่าสีและความหมาย	48
9 การขึ้นต้นชื่อไฟล์สำหรับการจัดเก็บการแสดงผลการจินตทัศน์	54
10 ค่าสถานะของเส้นทางที่ถูกใช้งานในต้นไม้จากคำตอบที่ 18 รุ่นที่ 10.....	68
11 ความน่าจะเป็นโดยประมาณในการเกิดการไขว้เปลี่ยนของคำตอบที่บัพแต่ละประเภท	74
12 ตัวแปรสำหรับเก็บข้อมูลและความหมาย	87
13 โครงสร้างข้อมูลภายในไฟล์ dataNNN .vgp	98
14 โครงสร้างกล่องข้อมูลรุ่น.....	99
15 โครงสร้างกล่องข้อมูลกำหนดการเชิงพันธุกรรม.....	99
16 โครงสร้างของข้อมูลรอบและรุ่น.....	99
17 โครงสร้างของข้อมูลคุณสมบัติ	100
18 โครงสร้างของข้อมูลโครงสร้างบัพ	101
19 ตัวอย่างข้อมูลภายในไฟล์ data001.vgp.....	101
20 ลัญลักษณ์แทนบัพในสายอักขระ	102

บทที่ 1

บทนำ



1.1 ความสำคัญและความเป็นมาของปัญหา

การรับรู้โดยทั่วไปของมนุษย์ในด้านการมองเห็น ถูกจำกัดด้วยข้อจำกัดของตา มนุษย์ เราไม่สามารถมองเห็นทุกสิ่งทุกอย่างด้วยตาเปล่าได้ เช่น รั้งสีเหนือม่วง หรือรั้งสีใต้แดง จะอยู่ในช่วงที่ไม่สามารถมองเห็นได้ด้วยตาเปล่า อุณหภูมิ แรงดัน สนามแม่เหล็ก สนามไฟฟ้า ล้วนแต่เป็นสิ่งที่ไม่สามารถมองเห็นได้ทั้งสิ้น ด้วยเหตุนี้มนุษย์เราจึงเพียรพยายามขยายขอบเขตของการรับรู้โดยอาศัยอุปกรณ์ต่างๆ นานาชนิด เช่น กล้องจุลทรรศน์ กล้องโทรทรรศน์ เพื่อขยายขนาดของภาพ เครื่องเอกซเรย์เพื่อมองเห็นรั้งสีที่มีความถี่สูง เทอร์โมมิเตอร์เพื่อแสดงอุณหภูมิ เป็นต้น

การใช้อุปกรณ์ต่างๆ ขยายความสามารถในการมองเห็นสิ่งที่มองไม่ได้ด้วยตาเปล่า มีประโยชน์กับมนุษย์มากมาย เช่นการใช้กล้องจุลทรรศน์ส่องเพื่อศึกษาพฤติกรรมของแบคทีเรีย ใช้กล้องโทรทรรศน์เพื่อการศึกษาทางด้านดาราศาสตร์ ดังเช่นกาลิเลโอมองเห็นดวงจันทร์ของดาวพฤหัสบดี ซึ่งนำการโคจรมาเปรียบเทียบกับโคจรของโลกรอบดวงอาทิตย์ ซึ่งทำให้ความเชื่อเก่าๆ ที่เกิดมาจากการคาดเดาหมดไปและได้ข้อสรุปที่แน่ชัดจากการทดลอง

การจินตทัศน์ข้อมูลเป็นการขยายขีดความสามารถในการการรับรู้ในอีกรูปแบบหนึ่ง เราสามารถใช้การจินตทัศน์ข้อมูลเพื่อเพิ่มความเข้าใจในข้อมูลที่เราสนใจได้ เช่นการจินตทัศน์ข้อมูลของโปรแกรมเรียงลำดับตัวเลขจากน้อยไปมาก และใช้ขั้นตอนวิธี (Algorithm) ที่แตกต่างกัน ในการเรียงลำดับแบบควิกซอร์ต (Quick Sort) จะเรียงลำดับได้เร็วกว่าการเรียงลำดับแบบฟอง (Bubble Sort) จากการจับเวลาจากการทำงานของโปรแกรม แต่ถ้าหากอยากรู้ว่าโปรแกรมเรียงลำดับตัวเลขได้อย่างไร การแสดงผลลัพธ์ของการเรียงลำดับที่ละขั้นตอนด้วยตัวเลขเพียงอย่างเดียวจะมองเห็นวิธีการสลับตัวเลขได้ยาก (รูปที่ 1) วิธีที่ดี

หน้านี้หายไป ไม่มีในต้นฉบับที่นำมาสแกน

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

เทอมินอล คือบัพ (Node) ภายในต้นไม้ที่ไม่มีลูกใช้แทนการกระทำหรือตัวเลขที่ขึ้นอยู่กับลักษณะของปัญหา

ฟังก์ชัน คือบัพภายในต้นไม้ใช้เป็นตัวเชื่อมโยง เทอมินอล และ ฟังก์ชัน ประกอบกันเป็นต้นไม้

การทำงานของกำหนดการเชิงพันธุกรรมโดยทั่วไปสามารถอธิบายได้โดยสังเขปดังนี้ เริ่มต้นโดยสุ่มสร้างเทอมินอลและฟังก์ชันประกอบเป็นต้นไม้ จนได้ขนาดที่ต้องการ ต้นไม้จะมีจำนวนเท่าใดขึ้นอยู่กับข้อกำหนดขนาดประชากร จากนั้นจะเริ่มประเมินค่าความเหมาะสมโดยการแหว่ผ่าน (Traversal) ในต้นไม้เริ่มจากบัพราก ถ้าในต้นไม้มีส่วนประกอบที่สอดคล้องกับคำตอบของปัญหา จะได้ค่าความเหมาะสม (Fitness Value) มาก ต้นไม้ที่ดีจะมีค่าความเหมาะสมมากกว่าต้นไม้ที่ไม่ดี เลือกต้นไม้ที่ค่าความเหมาะสมมากในกลุ่มมาจำนวนหนึ่ง เพื่อใช้ในการสร้างต้นไม้รุ่นต่อไป

ในรุ่นต่อมา โปรแกรมจะนำต้นไม้ผ่านการคัดเลือกจากค่าความเหมาะสม และทำการสุ่มมา 2 ต้น ใช้วิธีการไขว้เปลี่ยน (Crossover) และจะได้ต้นไม้ใหม่อีก 2 ต้นซึ่งมีโครงสร้างคล้ายกับ 2 ต้นเดิม ด้วยการสุ่มจุดที่จะทำการไขว้เปลี่ยนทั้งสองต้น และนำส่วนของต้นไม้ทั้ง 2 ต้นมาสลับกัน เมื่อทำงานได้ต้นไม้ครบตามจำนวนที่ต้องการแล้ว ประเมินค่าความเหมาะสมของต้นไม้ เพื่อทำการคัดเลือกและสร้างต้นไม้รุ่นต่อไปเรื่อยๆ จนกว่าจะสร้างต้นไม้ครบรุ่นที่ต้องการ หรือต้นไม้ที่ดีที่สุดนั้นสามารถนำไปใช้แก้ปัญหาได้

โดยทั่วไปข้อมูลที่ได้มาจากกำหนดการพันธุกรรมมีปริมาณมาก ต้นไม้ที่สร้างจากกำหนดการเชิงพันธุกรรมอาจจะมีขนาดเล็กสลับซับซ้อนอาจถึงหลายพันบัพ ในแต่ละต้นจะมีข้อมูลต่างๆ กำกับอยู่ คือค่าความเหมาะสม, หมายเลขของต้นไม้ที่เรียงลำดับจากการสร้าง, หมายเลขรุ่นของต้นไม้, หมายเลขต้นไม้พ่อแม่รุ่นก่อน, กรรมวิธีในการสร้างต้นไม้, อัตราการไขว้เปลี่ยน (Crossover Rate), อัตราการกลายพันธุ์ (Mutation Rate), อัตราการสืบพันธุ์ (Reproduction Rate), วิธีการเลือกต้นไม้เพื่อสร้างต้นไม้ต้นใหม่, ข้อมูลของสภาพแวดล้อม, เลขสุ่มเริ่มต้น (Random Seed) ดังนั้นถ้าต้องการเก็บข้อมูลทั้งหมดที่ได้จากการหาคำตอบจะต้องเก็บต้นไม้ทุกต้น ในทุกๆ รุ่น จะต้องใช้เนื้อที่เก็บข้อมูลมากมาย

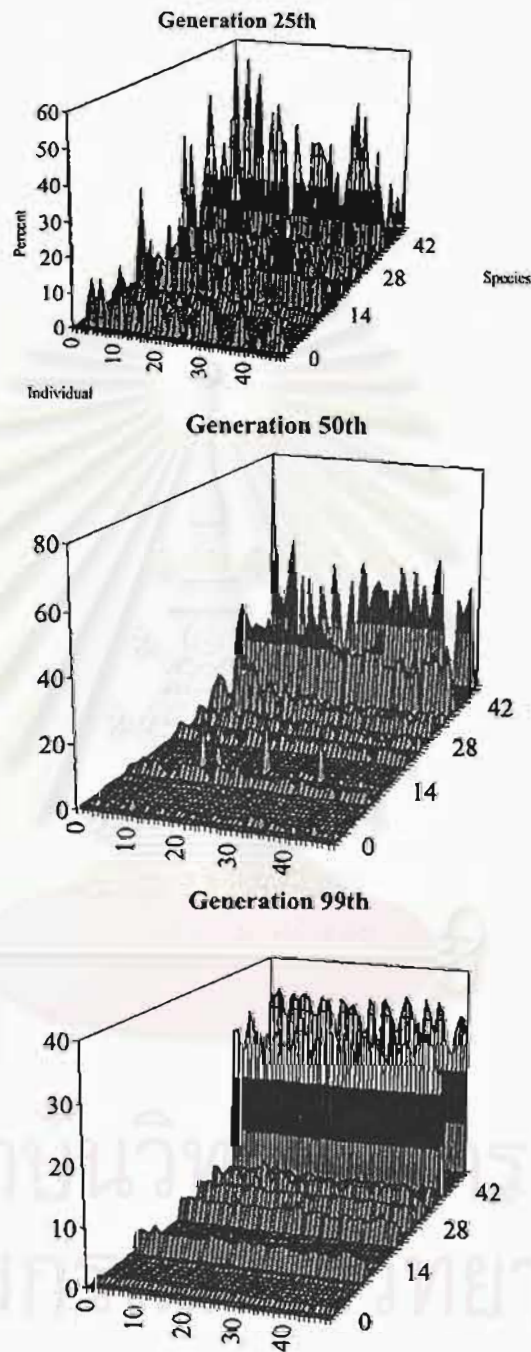
ในการดูโครงสร้างภายในคำตอบที่ได้มาจากกำหนดการเชิงพันธุกรรมนั้นจะเป็นไปได้อย่างยากเนื่องจากตัวคำตอบมีโครงสร้างแบบต้นไม้ซึ่งเมื่อนำมาเก็บในหน่วยความจำของโปรแกรมจำเป็นจะต้องเก็บข้อมูลโดยใช้รายการโยง (Link List) ที่เก็บอยู่ในแถวลำดับ (Array) ขนาดใหญ่ในการเก็บข้อมูล ข้อมูลของบัพที่เก็บค่าฟังก์ชันหรือเทอมมินอลและตัวชี้ (Pointer) ที่ชี้ไปยัง บัพลูก ในแต่ละบัพ โดยทุกๆ บัพที่เป็นฟังก์ชันจะมีตัวชี้สำหรับชี้ไปยัง บัพลูก 3 ถึง 4 ตำแหน่ง และคำตอบหนึ่งจะมีจำนวนบัพตั้งแต่ 40 บัพหรือมากกว่าขึ้นอยู่กับข้อกำหนดค่าในสร้างต้นไม้เริ่มต้น ในการหาคำตอบรุ่นหนึ่ง (Generation) ของกำหนดการเชิงพันธุกรรมจะต้องมีเนื้อที่สำหรับเก็บคำตอบจำนวน 400 คำตอบ สำหรับคำตอบรุ่นเดิม และเนื้อที่สำหรับเก็บคำตอบอีก 400 คำตอบสำหรับคำตอบในรุ่นถัดไป ซึ่งเกิดจากคำตอบรุ่นเดิมที่ผ่านกระบวนการวิวัฒนาการของกำหนดการเชิงพันธุกรรม ดังนั้นกำหนดการเชิงพันธุกรรมจึงมีความจำเป็นที่จะต้องใช้เนื้อที่ในการเก็บข้อมูลมากและทำให้เกิดความยากในการเข้าไปตรวจสอบข้อมูลที่มีเป็นจำนวนมากและเก็บอยู่ในรูปแบบรายการโยง

สิ่งที่น่าสนใจในกระบวนการวิวัฒนาการของกำหนดการเชิงพันธุกรรมก็คือ ความสัมพันธ์ระหว่างคำตอบที่เกิดจากรุ่นหนึ่งไปยังคำตอบรุ่นต่อมา คำตอบจะพัฒนาจากรุ่นแรกไปยังรุ่นที่สอง รุ่นที่สาม จนถึงรุ่นที่สิบ ความสัมพันธ์ที่เกิดจากคำตอบระหว่างคำตอบรุ่นแรกและคำตอบรุ่นที่สิบ ซึ่งในแต่ละรุ่นมีจำนวน 400 คำตอบนั้น เกิดจากการวิวัฒนาการในรูปแบบที่แตกต่างกัน การค้นหาว่าคำตอบในรุ่นที่สิบนั้น เกิดจากคำตอบใดบ้าง และเกิดด้วยการวิวัฒนาการแบบใด และได้เพิ่มเติมหรือเปลี่ยนแปลงโครงสร้างของคำตอบที่จุดใดอย่างไรบ้าง ซึ่งจะทำให้เห็นถึงการพัฒนาคำตอบของกำหนดการเชิงพันธุกรรมที่สามารถเปลี่ยนแปลงคำตอบเดิมแล้วได้คำตอบที่ดีขึ้น

ในงานวิจัย Visualization of evolutionary process in genetic programming [3] ได้นำการจินตทัศน์ข้อมูลมาทำการจินตทัศน์ปัญหาที่กำหนดการเชิงพันธุกรรมในรูปแบบที่เกี่ยวข้องกับการวิวัฒนาการโดยมุ่งเน้นความสนใจไปยังสายพันธุ์ โดยการกำหนดให้ประชากร (หรือผลเฉลย) เริ่มต้นมีป้ายระบุ (ID tag) ในแต่ละบัพ ซึ่งประชากรในรุ่นต่อไปที่เกิดจากการรวมกันของสายพันธุ์จะได้ป้ายระบุจากต้นกำเนิด หากประชากรเกิดจากการกลายพันธุ์จากเดิมจะได้ป้ายระบุอันใหม่

โดยการนับจำนวนบัพในประชากรแต่ละตัวและแสดงเป็นอัตราส่วนดังรูปที่ 3 การวิวัฒนาการของประชากรและจำนวนสายพันธุ์ ซึ่งนำมาจากภาพที่แสดงในงานวิจัย [3] การ

จินตทัศน์แสดงให้เห็นว่ามีเพียงบางสายพันธุ์ที่สามารถอยู่รอดได้ถึงการวิวัฒนาการรุ่นสุดท้ายและบางสายพันธุ์จะหายไป



รูปที่ 3 การวิวัฒนาการของประชากรและจำนวนสายพันธุ์ (จากงานวิจัย [3])

จากปัญหาต่างๆ ที่ได้กล่าวมาแล้วเบื้องต้น การนำการจินตทัศน์ข้อมูลมาใช้ จึงเป็นวิธีหนึ่งที่สามารถช่วยแสดงผลและอธิบายข้อมูลที่ได้มาจากกำหนดการเชิงพันธุกรรมได้

การนำการจินตทัศน์ข้อมูลมาศึกษากำหนดการเชิงพันธุกรรมที่ได้จากข้อมูลกำหนดการเชิงพันธุกรรมของชัยวัฒน์ เจริญภาพกรณ์ [4,5] มีสิ่งที่น่าสนใจหลายประการ เช่น ลักษณะโครงสร้างของต้นไม้ที่ดีมีลักษณะอย่างไร การไขว่เปลี่ยนโดยทั่วไปของต้นไม้แล้วได้ต้นไม้ที่ดีขึ้นเป็นอย่างไร และมีอะไรที่เป็นสาเหตุบ่งชี้ว่าเป็นต้นไม้ดีเป็นต้นไม้ที่ดีหรือไม่ ส่วนใดของต้นไม้ถูกใช้งานมากและส่วนใดของต้นไม้ไม่ได้ถูกใช้งาน เราจึงคาดหวังว่าโปรแกรมจินตทัศน์ข้อมูลจะเป็นสิ่งหนึ่งซึ่งช่วยให้การแสดงข้อมูลที่ได้จากกำหนดการเชิงพันธุกรรมเป็นไปโดยง่าย และช่วยอำนวยความสะดวกในการค้นหาข้อมูล แสดงโครงสร้างของคำตอบ แสดงความสัมพันธ์ระหว่างคำตอบ รวมทั้งข้อมูลอื่นๆ ที่เก็บอยู่ในกำหนดการเชิงพันธุกรรมมาแสดงผล การสร้างโปรแกรมจินตทัศน์ข้อมูลมีได้มีจุดมุ่งหมายในการสร้างเพื่อสามารถตอบคำถามต่างๆ ที่เกี่ยวกับกำหนดการเชิงพันธุกรรมได้ทั้งหมด แต่เป็นการสร้างโปรแกรมเพื่อให้ภาพที่ได้มาจากโปรแกรมจินตทัศน์ข้อมูลนั้นจะช่วยเพิ่มความเข้าใจเกี่ยวกับกำหนดการเชิงพันธุกรรม สำหรับผู้ที่ต้องการศึกษากำหนดการเชิงพันธุกรรมให้ดียิ่งขึ้น และจะเป็นสิ่งซึ่งช่วยในการแสดงผลลัพธ์ ของคำตอบที่ได้มาจากกำหนดการเชิงพันธุกรรมอีกทางหนึ่ง

1.2 งานวิจัยที่เกี่ยวข้อง

จุมพล พลวิชัย [6,7] ได้ทำการวิจัย เรื่องการเรียนรู้ของแขนหุ่นยนต์ โดยใช้วิธีกำหนดการเชิงพันธุกรรม เพื่อสังเคราะห์โปรแกรมควบคุมแขนหุ่นยนต์ ในการเคลื่อนที่หลบสิ่งกีดขวางไปยังเป้าหมาย โดยใช้สภาพจำลองบนเครื่องคอมพิวเตอร์ และได้นำมาทดลองบนสภาพใช้งานจริง

ต่อมา ประภาส จงสถิตย์วัฒนา [8] ได้ศึกษางานวิจัยดังกล่าวพบว่าผลจากกำหนดการเชิงพันธุกรรมที่ได้ยังไม่มี ความทนทาน เมื่อขยับตำแหน่งเพียงเล็กน้อยจะทำให้เกิดความล้มเหลวในการไปสู่เป้าหมายได้

C. Wongsiriprasert, P. Chongstitvatana และ S. Prasitjutrakul [3] ได้วิจัยการจินตทัศน์วิวัฒนาการข้อมูลกำหนดการเชิงพันธุกรรม ในปัญหาการนำร่องหุ่นยนต์ โดยนำเสนอในรูปแบบของกราฟ 2 มิติและ 3 มิติ ที่เปลี่ยนแปลงตามรุ่นของประชากร จำแนกตามสายพันธุ์ เป็นแบบอย่างในการนำการจินตทัศน์มาช่วยในการศึกษากำหนดการเชิงพันธุกรรม ดังที่ได้แสดงไว้ในเบื้องต้นแล้ว

ชัยวัฒน์ เจษฎาปกรณ์ [4] ได้วิจัยการลดทอนความเพียรพยายามเชิงคำนวณของวิธีการเรียนรู้แบบกำหนดการเชิงพันธุกรรม โดยนำกำหนดการเชิงพันธุกรรมมาแก้ปัญหาการเคลื่อนที่ของแขนหุ่นยนต์ จากงานวิจัยของ จุมพล พลวิชัย [6]

1.3 วัตถุประสงค์

ศึกษากระบวนการวิวัฒนาการของคำตอบโดยกำหนดการเชิงพันธุกรรม เพื่อทำการจินตทัศน์ข้อมูลในระหว่างการวิวัฒนาการของคำตอบที่ได้จากกำหนดการเชิงพันธุกรรม โดยสร้างโปรแกรมประมวลผลข้อมูลและแสดงผลกราฟิก

1.4 ขอบเขตของงานวิจัย

ทำการจินตทัศน์ข้อมูลกำหนดการเชิงพันธุกรรม โดยใช้ข้อมูลจากโปรแกรมกำหนดการเชิงพันธุกรรมของ ชัยวัฒน์ เจษฎาปกรณ์ [4] และโปรแกรมที่ทำการพัฒนาสามารถทำงานบนเครื่องคอมพิวเตอร์ส่วนบุคคลได้

1.5 ประโยชน์ที่คาดว่าจะได้รับ

ได้โปรแกรมสำหรับช่วยในการจินตทัศน์ข้อมูล ซึ่งจะช่วยให้เพิ่มความเข้าใจในข้อมูลที่ได้จากกำหนดการเชิงพันธุกรรม ซึ่งอาจเป็นแนวทางในการปรับปรุงและพัฒนา กำหนดการเชิงพันธุกรรมได้

1.6 เนื้อหาในวิทยานิพนธ์นี้

ในบทที่ 2 กล่าวถึงกำหนดการเชิงพันธุกรรมสำหรับควบคุมแขนหุ่นยนต์ ซึ่งประกอบไปด้วยข้อมูลกำหนดการเชิงพันธุกรรม ปัญหาแขนหุ่นยนต์ที่นำกำหนดการเชิงพันธุกรรมมาใช้ สนามทดลอง การวัดค่าความเหมาะสม พารามิเตอร์กำหนดการเชิงพันธุกรรม บทที่ 3 แนะนำการใช้งานระบบการจินตทัศน์ข้อมูล ก่อนแสดงรายละเอียดของระบบ ในบทที่ 4 ซึ่งกล่าวถึงโครงสร้างของระบบการจินตทัศน์ข้อมูลกำหนดการเชิงพันธุกรรมโดย

แยกออกเป็นส่วนๆ เช่น การรับส่งข้อมูลระหว่างกำหนดการเชิงพันธุกรรมของชัยวัฒน์ เจริญภาพกรณ์ [4] กับระบบจินตทัศน์ข้อมูล ส่วนรับข้อมูลจากผู้ใช้ ส่วนการจินตทัศน์ เป็นต้น ในบทที่ 5 กล่าวถึงความสามารถต่างๆ ของระบบการจินตทัศน์ที่ได้ออกแบบไว้ในตอนเริ่มงานวิจัย ซึ่งภายหลังจากการทดลองใช้งานได้เพิ่มส่วนเพิ่มเติมของระบบการจินตทัศน์ข้อมูลซึ่งได้กล่าวไว้ในบทที่ 6 และสรุปในบทที่ 7

ในบทต่างๆ ได้แสดงถึงการจินตทัศน์การวิวัฒนาการข้อมูลคำตอบที่ได้จากกำหนดการเชิงพันธุกรรมในหลายแง่มุมรวมทั้งระบบการจินตทัศน์ข้อมูลกำหนดการเชิงพันธุกรรม ซึ่งสิ่งที่ได้รับจากการใช้การจินตทัศน์ข้อมูลโดยตรงก็คือได้เห็นภาพของผลลัพธ์บางอย่างซึ่งไม่ตรงกับสิ่งที่ได้คาดเดาไว้ ซึ่งเป็นสิ่งที่จะอธิบายได้ด้วยวิธีจินตทัศน์ และนำเอาการจินตทัศน์มาเป็นส่วนช่วยในการสนับสนุนสมมติฐานหรือทฤษฎีต่างๆ ผู้วิจัยคาดหวังอย่างยิ่งว่าจะมีการนำการจินตทัศน์ข้อมูลมาใช้ประโยชน์ในงานส่วนที่เกี่ยวข้องกับข้อมูลอย่างอื่น ไม่เพียงแต่เฉพาะข้อมูลกำหนดการเชิงพันธุกรรมเท่านั้น

1.7 งานตีพิมพ์

ส่วนหนึ่งของงานวิจัยนี้ได้รับการตีพิมพ์เป็นโปสเตอร์ในงานประชุมเชิงวิชาการ National Computer Science and Engineering Conference, (NCSEC'99) ในชื่อเรื่อง A Program for Visualizing Data in the Evolution of Solutions from Genetic Programming โดย Rodkaew, Y. and Chongstitvatana, P. [9] ซึ่งแสดงโครงสร้างของระบบการจินตทัศน์ข้อมูลและผลลัพธ์บางส่วนที่ได้จากการจินตทัศน์ข้อมูลจากบทที่ 4 และบทที่ 5

ได้รับการตีพิมพ์ในงานประชุมเชิงวิชาการ Annual National Symposium on Computational Science and Engineering, (ANSCSE #4) ในชื่อเรื่อง How to reduce the memory requirement in genetic programming โดย Rodkaew, Y. and Chongstitvatana, P. [10] ซึ่งนำเสนอวิธีจัดเก็บและประมวลผลข้อมูลกำหนดการเชิงพันธุกรรมโดยประหยัดหน่วยความจำและใช้เวลาในการประมวลผลไม่ช้าลง

บทที่ 2

กำหนดการเชิงพันธุกรรมสำหรับควบคุมแขนหุ่นยนต์

ชัยวัฒน์ เจริญภาพกรณ์ [4] ได้ทำการวิจัยต่อจากงานวิจัยของ จุมพล พลวิชัย [6] โดยใช้วิธีกำหนดการเชิงพันธุกรรม เพื่อสังเคราะห์โปรแกรมควบคุมแขนหุ่นยนต์ให้มีความสามารถหลบหลีกสิ่งกีดขวางและบังคับให้แขนหุ่นยนต์เคลื่อนที่ไปยังเป้าหมายที่กำหนด งานวิจัยที่ได้ทำการวิจัยต่อมาเกี่ยวข้องโดยตรงกับการลดทอนความเพียรพยายามเชิงคำนวณเพื่อปรับปรุงกรรมวิธีให้กำหนดการเชิงพันธุกรรมสังเคราะห์คำตอบที่มีประสิทธิภาพเพิ่มขึ้น และมีโอกาสได้คำตอบที่ดีขึ้นกว่ากำหนดการเชิงพันธุกรรมเดิม ด้วยวิธีที่เรียกว่า ADFX ในบทนี้จะกล่าวถึงรายละเอียดในด้านของข้อกำหนดของแขนหุ่นยนต์ และส่วนของโปรแกรมกำหนดการเชิงพันธุกรรม ที่เกี่ยวข้องกับการจินตทัศน์ข้อมูลในงานวิจัยนี้เท่านั้น ข้อมูลต่างๆ ในบทนี้นำมาจากวิทยานิพนธ์ของชัยวัฒน์ เจริญภาพกรณ์ [4] สำหรับผู้ที่สนใจในรายละเอียดสามารถค้นคว้าได้จากงานวิจัยดังกล่าว

ในการจินตทัศน์ข้อมูลได้นำโปรแกรมของชัยวัฒน์ เจริญภาพกรณ์ [4] มาเป็นต้นแบบสำหรับการสร้างข้อมูลต่างๆ ทางด้านกำหนดการเชิงพันธุกรรม จึงทำให้โปรแกรมจินตทัศน์ข้อมูลใช้ข้อกำหนดเหมือนกับ โปรแกรมของชัยวัฒน์ เจริญภาพกรณ์ [4] ยกเว้นส่วนของ ADFX ไม่นำมาใช้ในการจินตทัศน์

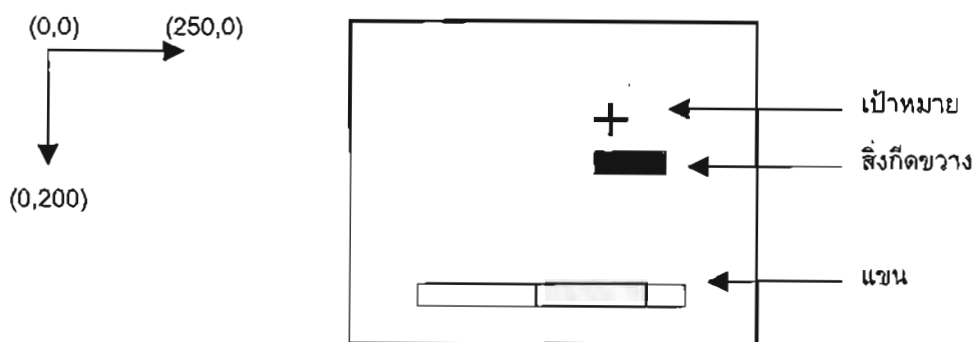
2.1 ข้อกำหนดของแขนหุ่นยนต์

2.1.1 แขนหุ่นยนต์

แขนหุ่นยนต์ได้ถูกออกแบบให้มีลักษณะคล้ายกับแขนของมนุษย์ซึ่งมีทั้งหมด 3 ส่วนคือ ไหล่ ศอก และข้อมือ โดยความยาวของส่วนไหล่ ส่วนศอก และส่วนมือเท่ากับ 74 จุดภาพ 68 จุดภาพ และ 24 จุดภาพ ตามลำดับ ความกว้างของแต่ละส่วนเท่ากับ 15 จุดภาพ และมืองศาในการหมุนจำกัดคือ ส่วนไหล่ -90 ถึง 90 องศา ส่วนศอก -20 ถึง 160 องศา และส่วนมือ -90 ถึง 90 องศา สามารถหมุนเพื่อปรับองศาเพิ่มหรือลดได้ครั้งละ 5 องศา

2.1.2 สนามทดลอง

แขนหุ่นยนต์จะมีพื้นที่ในการเคลื่อนไหวมีขนาดความกว้าง 200 จุดภาพ (Pixel) และมีความยาว 250 จุดภาพ โดยส่วนไหล่จะยึดอยู่ที่ตำแหน่ง (40,170) ในแต่ละสภาพแวดล้อมของการทดลอง สนามจะมีเป้าหมายและสิ่งกีดขวางวางไว้แตกต่างกัน ดังรูปที่ 4 แสดงสภาพแวดล้อมที่ 1 เป็นตัวอย่างสภาพแวดล้อมที่ 1 จากวิทยานิพนธ์ของชัยวัฒน์ เจริญภาพกรณ์



รูปที่ 4 แสดงสภาพแวดล้อมที่ 1

ตาราง 1 รายละเอียดของสภาพแวดล้อมที่ 1

ที่มา	สภาพแวดล้อมที่ยากที่สุดในกลุ่มของสภาพแวดล้อมที่มีสิ่งกีดขวาง 1 สิ่งกีดขวาง
สิ่งกีดขวาง	สี่เหลี่ยมผืนผ้าตำแหน่งมุมบนซ้าย (150,80) ความกว้าง 45 จุดภาพ ความสูง 15 จุดภาพ
เป้าหมาย	ตำแหน่ง (160,60) รัศมี 10 จุดภาพ

2.2 ข้อกำหนดของกำหนดการแข่งขันหุ่นยนต์

กำหนดการแข่งขันหุ่นยนต์สำหรับควบคุมแขนหุ่นยนต์ได้กำหนดในส่วนที่เป็นฟังก์ชันและเทอมินอล ไว้ดังนี้

2.2.1 ส่วนฟังก์ชัน

ได้กำหนดไว้ 3 ฟังก์ชันคือ IF-AND, IF-OR และ IF-NOT มีลักษณะการทำงานตามตาราง 2

ตาราง 2 ฟังก์ชันที่ใช้ในกำหนดการเชิงพันธุกรรมของชัยวัฒน์ เจริญภาพกรณ์

ฟังก์ชัน	จำนวนอาร์กิวเมนต์	ลักษณะการทำงาน
IF-AND	4	เรียกอาร์กิวเมนต์ที่ 1 และ 2 ทำงาน หากอาร์กิวเมนต์ที่ 1 และ 2 เป็นจริงทั้งคู่ เรียกอาร์กิวเมนต์ที่ 3 ทำงาน ถ้าหากอาร์กิวเมนต์ที่ 1 และ 2 อันใดอันหนึ่งไม่เป็นจริงหรือไม่เป็นจริงทั้งคู่ เรียกอาร์กิวเมนต์ที่ 4 ทำงาน
IF-OR	4	เรียกอาร์กิวเมนต์ที่ 1 และ 2 ทำงาน ถ้าหากอาร์กิวเมนต์ที่ 1 และ 2 อันใดอันหนึ่งเป็นจริงหรือเป็นจริงทั้งคู่ เรียกอาร์กิวเมนต์ที่ 3 ทำงาน หากอาร์กิวเมนต์ที่ 1 และ 2 ไม่เป็นจริงทั้งคู่ เรียกอาร์กิวเมนต์ที่ 4 ทำงาน
IF-NOT	3	เรียกอาร์กิวเมนต์ที่ 1 ทำงาน ถ้าหากอาร์กิวเมนต์ที่ 1 เป็นจริงเรียกอาร์กิวเมนต์ที่ 2 ทำงาน ถ้าหากอาร์กิวเมนต์ที่ 1 ไม่เป็นจริงเรียกอาร์กิวเมนต์ที่ 3 ทำงาน

ภายหลังจากฟังก์ชันทำงานแล้วจะส่งค่า จริง หรือ เท็จ ของอาร์กิวเมนต์ที่เรียกทำงานตัวสุดท้ายให้กับตัวที่เรียกฟังก์ชัน

2.2.2 ส่วนเทอมินอล

มีทั้งหมด 11 เทอมินอลตามตาราง 3

ตาราง 3 เทอมินอลที่ใช้ในกำหนดการเชิงพันธุกรรมของชัยวัฒน์ เจษฎาปกรณ์

เทอมินอล*	ลักษณะการทำงาน
S+	หมุนข้อต่อหัวไหล่ขึ้น 1 ชั้น (5 องศา)
S-	หมุนข้อต่อหัวไหล่ลง 1 ชั้น
E+	หมุนข้อต่อข้อศอกขึ้น 1 ชั้น
E-	หมุนข้อต่อข้อศอกลง 1 ชั้น
W+	หมุนข้อต่อข้อมือขึ้น 1 ชั้น
W-	หมุนข้อต่อข้อมือลง 1 ชั้น
HIT?	ทำหน้าที่ในการตรวจสอบสภาพการชนกับสิ่งกีดขวาง โดยจะให้ค่าเป็นจริง เมื่อส่วนใดส่วนหนึ่งของแขนหุ่นยนต์มีการชนกับสิ่งกีดขวาง และให้ค่าเป็นเท็จเมื่อไม่พบการชนกับสิ่งกีดขวาง
SEE?	ทำหน้าที่ในการตรวจสอบการมองเห็นเป้าหมาย โดยให้ค่าเป็นจริงเมื่อแนวเส้นตรงจากปลายแขนหุ่นยนต์ไปยังเป้าหมายไม่ถูกสิ่งกีดขวางบัง และให้ค่าเป็นเท็จเมื่อพบว่ามีสิ่งกีดขวางบังอยู่
INC?	ทำหน้าที่ตรวจสอบระยะทางที่เพิ่มขึ้นระหว่างปลายแขนหุ่นยนต์และเป้าหมาย โดยให้ค่าเป็นจริงเมื่อพบว่าปลายแขนหุ่นยนต์เคลื่อนที่ออกห่างจากเป้าหมาย และให้ค่าเป็นเท็จเมื่อพบว่าปลายแขนหุ่นยนต์เคลื่อนที่เข้าใกล้เป้าหมายมากขึ้น
DEC?	ทำหน้าที่ตรวจสอบระยะทางที่ลดลงระหว่างปลายแขนหุ่นยนต์และเป้าหมาย โดยให้ค่าเป็นจริงเมื่อพบว่าปลายแขนหุ่นยนต์เคลื่อนที่เข้าใกล้เป้าหมายและให้ค่าเป็นเท็จเมื่อพบว่าปลายแขนหุ่นยนต์เคลื่อนที่ออกห่างจากเป้าหมายมากขึ้น
OUT?	ทำหน้าที่ตรวจสอบการออกนอกพื้นที่ โดยให้ค่าเป็นจริงเมื่อพบว่ามีส่วนใดส่วนหนึ่งของแขนหุ่นยนต์เคลื่อนที่ออกนอกพื้นที่ และให้ค่าเป็นเท็จเมื่อไม่พบการออกนอกพื้นที่

2.2.3 การวัดค่าความเหมาะสม

ค่าความเหมาะสม (Fitness Value) ของคำตอบที่ได้จากกำหนดการเชิงพันธุกรรม สามารถวัดได้จากสมการตามที่ได้แสดงไว้ใน วิทยานิพนธ์ของชัยวัฒน์ เจษฎาปกรณ์ ดังนี้

* เมื่อเรียกใช้เทอมินอล S+,S-,E+,E-,W+,W- จะให้ค่าจริงทั้งหมด

$$\begin{aligned}
 \text{FitnessValue} = & -2000 \times \text{FinalDistance}/\text{InitialDistance} \\
 & - 100 \times \text{Sumdistance}/\text{InitialDistance} \\
 & - 1000 \times \text{notSee} \\
 & - 4000 \times \text{Die}
 \end{aligned}$$

ตาราง 4 ตัวแปรในการคำนวณค่าความเหมาะสมของซัพวัญณ์ เงื่อนไขปรกรณ์

ตัวแปร	ความหมาย
<i>InitialDistance</i>	ระยะทางระหว่างปลายแขนหุ่นยนต์และเป้าหมายก่อนการประมวลผลโปรแกรมหุ่นยนต์
<i>FinalDistance</i>	ระยะทางระหว่างปลายแขนหุ่นยนต์และเป้าหมายเมื่อประมวลผลตามโปรแกรมหุ่นยนต์จบแล้ว
<i>SumDistance</i>	ระยะทางรวมในการเคลื่อนแขนหุ่นยนต์จนจบการประมวลผลตามโปรแกรมหุ่นยนต์
<i>NotSee</i>	เมื่อประมวลผลตามโปรแกรมหุ่นยนต์จบแล้วพบว่าระหว่างปลายแขนหุ่นยนต์และเป้าหมายมีสิ่งกีดขวางบังอยู่
<i>Die</i>	โปรแกรมหุ่นยนต์สิ้นสุดการทำงานในลักษณะตายซึ่งเกิดจากระหว่างการประมวลผลโปรแกรมหุ่นยนต์พบว่ามีกชนระหว่างแขนหุ่นยนต์และสิ่งกีดขวางมากเกินไปที่กำหนด

ค่าความเหมาะสมที่ได้จากสมการดังกล่าวจะแสดงว่าคำตอบที่ได้มีประสิทธิภาพในการแก้ปัญหาดีหากมีค่ามากเมื่อเข้าใกล้ค่าศูนย์ และคำตอบจะไม่มีประสิทธิภาพในการแก้ปัญหาเมื่อค่าความเหมาะสมที่วัดได้มีค่าติดลบมาก

2.2.4 ส่วนพารามิเตอร์ของกำหนดการเชิงพันธุกรรม

ค่าพารามิเตอร์ต่างๆของกำหนดการเชิงพันธุกรรมแสดงดังตาราง 5 ในการใช้งานผู้ใช้จะสามารถปรับขนาดเริ่มต้นของประชากรได้และอัตราการกลายได้ขึ้นอยู่กับค่าของพารามิเตอร์เริ่มต้นสำหรับคำสั่งในการสร้างข้อมูลของโปรแกรมจินตทัศน์ข้อมูล และการใช้งานสำหรับการจินตทัศน์ได้ปรับจำนวนรอบ (Run) ในการทดลองลดลงเหลือ 10 รอบ จากเดิมที่กำหนดไว้ในงานวิจัยเดิม 1,000 รอบ

ตาราง 5 ค่าพารามิเตอร์กำหนดการเชิงพันธุกรรมที่ปรับปรุงจากงานวิจัยของชัยวัฒน์ เจริญภาพกรณ์

จำนวนประชากร	400 ตัว
ขนาดเริ่มต้นของประชากร	80 สัญลักษณ์ (1 สัญลักษณ์เท่ากับ 1 ฟังก์ชัน หรือ 1 เทอมินอล)
จำนวนรุ่นมากที่สุดในการทดลอง	10 รุ่น (Generation)
จำนวนรอบของการทดลอง	10 รอบ (Run)
อัตราการสืบพันธุ์	10 เปอร์เซ็นต์
อัตราการไขว้เปลี่ยน	40 เปอร์เซ็นต์
อัตราการกลายแบบต่อยอด	25 เปอร์เซ็นต์
อัตราการกลายแบบต่อปลาย	25 เปอร์เซ็นต์

สำหรับการลดจำนวนรอบในการทดลองลงเหลือเพียง 10 รอบในงานวิจัยนี้เนื่องจากการต้องการความรวดเร็วในการสร้างข้อมูลสำหรับการจินตทัศน์ ซึ่งการทำงานวิจัยเดิมใช้ 1000 รอบในการสร้างข้อมูลนั้นเพราะต้องการนำข้อมูลไปทำการคำนวณหาค่าความเพียรพยายามเชิงคำนวณ (Effort) ทำให้การสร้างข้อมูลครั้งหนึ่งใช้เวลาประมาณ 30 นาทีเมื่อให้โปรแกรมทำงานบนเครื่อง Pentium 133Mhz เป็นเวลาที่นานพอสมควรและโปรแกรมต้องการเนื้อที่เก็บข้อมูลสำหรับการจินตทัศน์ประมาณ 1 Mbytes ต่อ 1 รอบการทดลอง ซึ่งถ้ากำหนดเป็น 1000 รอบ ต้องมีเนื้อที่สำหรับเก็บประมาณ 1 Gbytes

สำหรับการจินตทัศน์ข้อมูลนั้นไม่ได้ออกแบบไว้สำหรับการแสดงค่าความเพียรพยายามเชิงคำนวณหรือการแสดงผลเปรียบเทียบระหว่างจำนวนรอบที่แตกต่างกัน แต่เน้นในรายละเอียดภายในข้อมูลการทดลองในแต่ละรอบ อย่างไรก็ตามหากผู้ใช้ต้องการก็สามารถปรับค่ากำหนดให้สร้าง 1000 รอบได้ตามปกติ

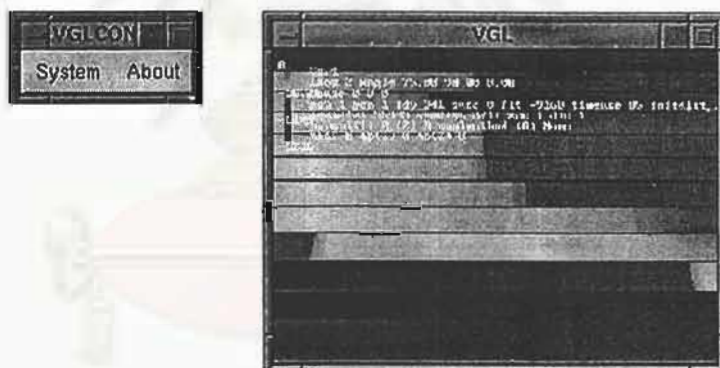
บทที่ 3

การใช้งานระบบการจินตทัศน์ข้อมูล

ระบบจินตทัศน์ข้อมูลซึ่งเป็นผลจากงานวิจัยนี้ มีชื่อว่า GPVIS ในบทนี้จะแนะนำการใช้งานระบบ GPVIS เพื่อแสดงให้เห็นโปรแกรมและวิธีการใช้งานก่อนที่จะแสดงถึงโครงสร้างของโปรแกรมในบทถัดไป ในบทนี้จะไม่เน้นในรายละเอียดในการใช้งานมากนัก เนื่องจากรายละเอียดต่างๆ ของฟังก์ชันในโปรแกรมได้กล่าวไว้ในบทที่ 5

3.1 เข้าสู่ระบบ GPVIS

เมื่อเริ่มต้นใช้งาน เรียกโปรแกรมโดยพิมพ์ Vglcon ที่หน้าต่างใน X-Windows โปรแกรมจะแสดง 2 หน้าต่างดังรูปที่ 5



รูปที่ 5 หน้าต่างการใช้งานระบบ GPVIS

หน้าต่าง VGLCON ทางด้านซ้ายเป็นส่วนรับข้อมูลจากผู้ใช้ ด้วยรายการเลือก (Menu) เมื่อผู้ใช้กดเมาส์ (Mouse) ที่แถบรายการเลือก (Menu bar) **System** โปรแกรมจะแสดงรายการต่างๆ เมื่อผู้ใช้เลือกรายการ โปรแกรมจะแสดงหน้าต่างกล่องโต้ตอบ (Dialog Box) ใหม่เพื่อรับข้อมูลจากผู้ใช้ และแสดงผลกราฟิกในหน้าต่าง VGL ด้านขวาคือแผ่นที่แสดงประชากรทั้งหมดเพื่อเอาไว้เลือกตัวผลเฉลยที่ต้องการ สีต่างๆ แสดงถึงค่าความเหมาะสมของผลเฉลยนั้นๆ เพื่อความรวดเร็วในการสั่งงาน รายการบางอย่างผู้ใช้สามารถ

ใช้แป้นลัด (Hot key) กับหน้าต่างทางด้านขวาได้โดยตรง โดยการเลื่อนเมาส์ไปยังหน้าต่างขวาและกดแป้นลัด โปรแกรมจะทำการปรับปรุงกราฟิกโดยอัตโนมัติ หน้าต่าง VGL ด้านขวาคือเป็นส่วนหนึ่งของโปรแกรม VGL ซึ่งเรียกใช้ OpenGL ในการประมวลผลกราฟิกดังนั้นการแสดงผลจึงสามารถแสดงอยู่ในรูป 3 มิติทั้งหมด

หน้าต่างทั้งสองเป็นโปรแกรมสองโปรแกรมซึ่งติดต่อกันและกันโดย PIPELINE ระบบได้ถูกออกแบบให้มีการปรับปรุงสถานะต่างๆ ของโปรแกรมทั้งสองตลอดเวลา ผู้ใช้งานจึงสามารถสั่งงานผ่านหน้าต่างทั้งสองสลับกันได้

รูปที่ 5 หน้าต่างการใช้งานระบบ GPVIS เป็นการแสดงค่าความเหมาะสมของคำตอบ(หรือประชากร)ที่ได้จากกำหนดการเชิงพันธุกรรมใน 1 รอบ แสดงแถบสีเรียงตามค่าความเหมาะสมตามรุ่นของคำตอบจากน้อยไปมากจากซ้ายไปขวา ค่าตอบรุ่นแรกแสดงด้านล่างสุด ค่าตอบรุ่นสุดท้ายแสดงไว้ด้านบนสุด รายละเอียดกล่าวไว้ในบทที่ 5

3.2 แสดงข้อมูลคำตอบที่ได้จากกำหนดการเชิงพันธุกรรม

เมื่อผู้ใช้เลือกรายการ "Individual" โปรแกรมจะแสดงกล่องโต้ตอบให้ผู้ใช้กรอกข้อมูล ณ ช่อง Run เป็น 1, Generation เป็น 10 และ Individual เป็น 18 ดังรูปที่ 6 กล่องโต้ตอบสำหรับการเลือกคำตอบที่ได้จากกำหนดการเชิงพันธุกรรมจากนั้นกดปุ่ม

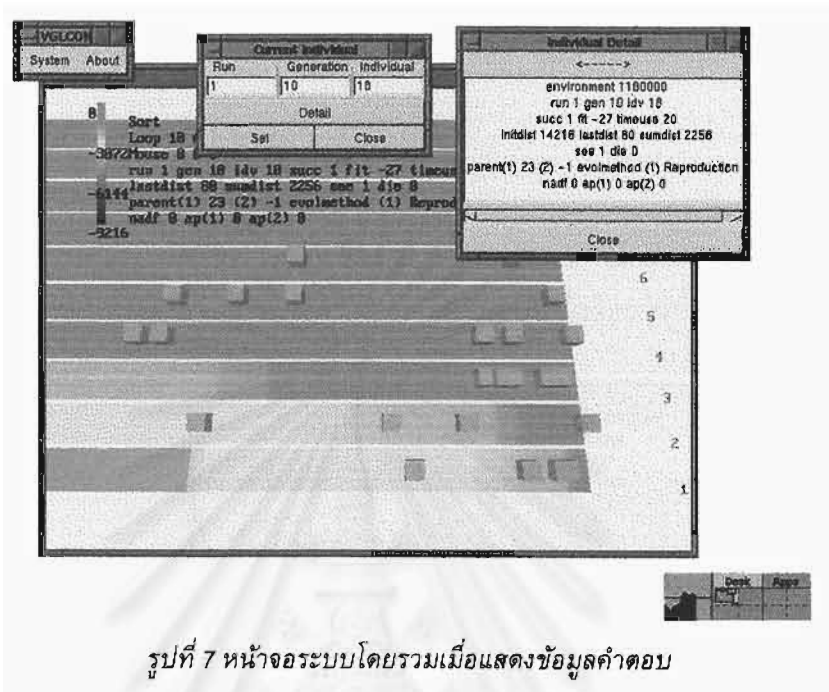
Run	Generation	Individual
1	10	18

Detail

Set Close

รูปที่ 6 กล่องโต้ตอบสำหรับการเลือกคำตอบที่ได้จากกำหนดการเชิงพันธุกรรม

หลังจากกดปุ่ม **Set** แล้ว ผู้ใช้งานจะสังเกตเห็นความเปลี่ยนแปลงในหน้าต่าง VGL ให้ผู้ใช้งานกดปุ่ม **Detail** โปรแกรมจะแสดงกล่องโต้ตอบรายละเอียดข้อมูลคำตอบปัจจุบัน ผู้ใช้งานจะเห็นหน้าจอดังรูปที่ 7

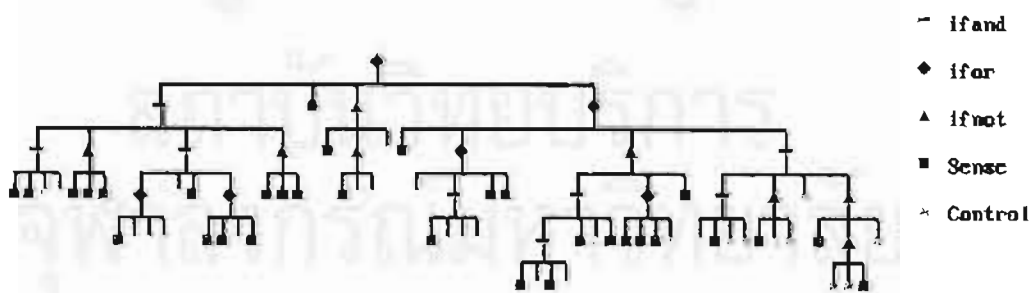


รูปที่ 7 หน้าจอระบบโดยรวมเมื่อแสดงข้อมูลคำตอบ

สำหรับรายละเอียดต่างๆ ในหน้าต่าง **Individual Detail** แสดงไว้ในบทที่ 5

3.3 แสดงโครงสร้างของคำตอบ

หลังจากที่ได้เลือกคำตอบที่สนใจแล้ว เลือกรายการ **System** และเลือก **"Individual Tree"** ระบบจะแสดงโครงสร้างของคำตอบดังรูปที่ 8

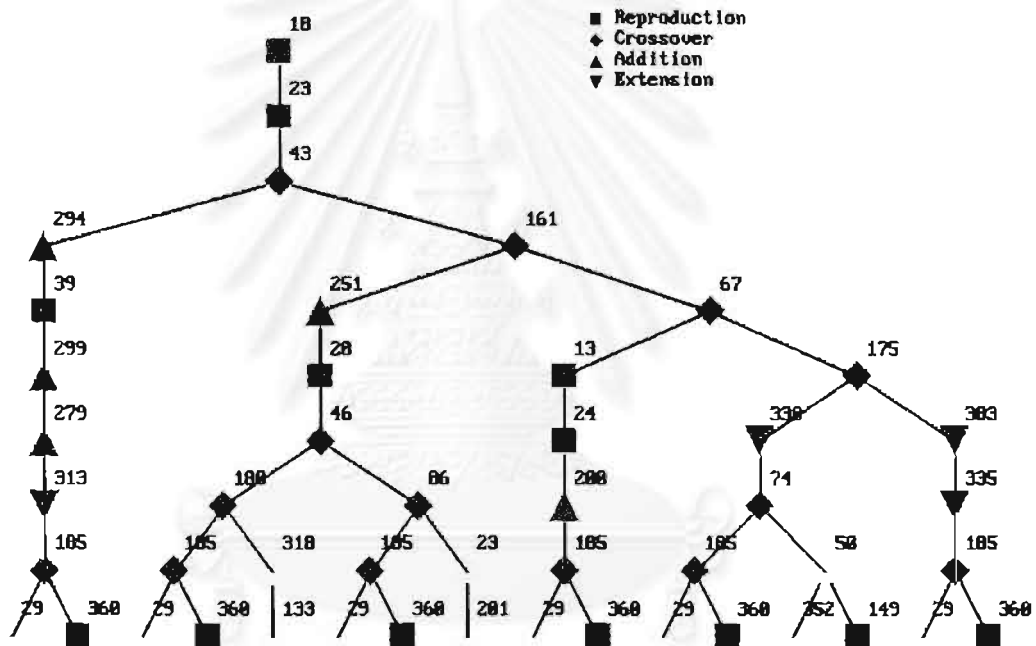


รูปที่ 8 โครงสร้างของคำตอบ

ระบบจะแสดงโครงสร้างของคำตอบที่เราสนใจบนหน้าจอโดยแสดงบัพ ใช้สีแทนบัพประเภทต่างๆ ในขั้นตอนนี้เราสามารถใช้น๊อคค (Arrow Key) เพื่อเลื่อนดูโครงสร้างพร้อมกับใช้แป้น - และ + สำหรับการย่อขยายภาพได้ จากนั้นกดแป้น ESC เพื่อกลับสู่การแสดงผลข้อมูลคำตอบที่ได้จากกำหนดการเชิงพันธุกรรม

3.4 แสดงความสัมพันธ์ของคำตอบเริ่มต้น

เลือกรายการ **System** และเลือก "Hierarchical Model" ระบบจะแสดงความสัมพันธ์ของคำตอบที่เลือกกับคำตอบเริ่มต้น ว่าคำตอบที่ได้เลือกไว้นั้นเกิดมาจากคำตอบใดบ้างด้วยการวิวัฒนาการแบบใด



รูปที่ 9 ความสัมพันธ์ของคำตอบ

ในการแสดงความสัมพันธ์นี้เราสามารถเลือกคำตอบที่สนใจภายในได้โดยการใช้แป้น PageUp และ PageDn เพื่อเลือกรุ่นที่ต้องการ และใช้แป้น < และ > เพื่อเลือกคำตอบทางด้านซ้ายและขวา จากนั้นสามารถใช้คำสั่ง "Individual Tree" เพื่อดูโครงสร้างของคำตอบ หรือเลือกคำตอบเพื่อใช้ทดลองควบคุมแขนหุ่นยนต์หรือ กดแป้น ESC เพื่อกลับสู่การแสดงผลข้อมูลคำตอบที่ได้จากกำหนดการเชิงพันธุกรรม

ทางด้านซ้ายและขวา จากนั้นสามารถใช้คำสั่ง "Individual Tree" เพื่อดูโครงสร้างของคำตอบ หรือเลือกคำตอบเพื่อใช้ทดลองควบคุมแขนหุ่นยนต์หรือ กดแป้น ESC เพื่อกลับสู่การแสดงผลข้อมูลคำตอบที่ได้จากกำหนดการเชิงพันธุกรรม

3.5 แสดงการเคลื่อนไหวแขนหุ่นยนต์

เลือกรายการ **System** และเลือก "Simulation" ระบบจะนำคำตอบที่ได้เลือกไว้มาทำงานแบบจำลอง (Simulation) เพื่อแสดงให้เห็นการนำคำตอบที่ได้ไปใช้ควบคุมแขนหุ่นยนต์ว่าจะเกิดการขยับอย่างไรและสามารถไปถึงเป้าหมายที่ได้กำหนดไว้หรือไม่



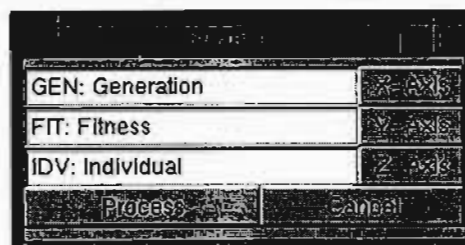
รูปที่ 10 การจำลองการเคลื่อนไหวแขนหุ่นยนต์

ในส่วนนี้เราสามารถบันทึกการเคลื่อนไหวของแขนหุ่นยนต์เป็นแฟ้มภาพได้ ซึ่งรายละเอียดได้แสดงไว้ในหัวข้อ 5.6

3.6 แสดงข้อมูลกำหนดการเชิงพันธุกรรมในรูปแบบ 3 มิติ

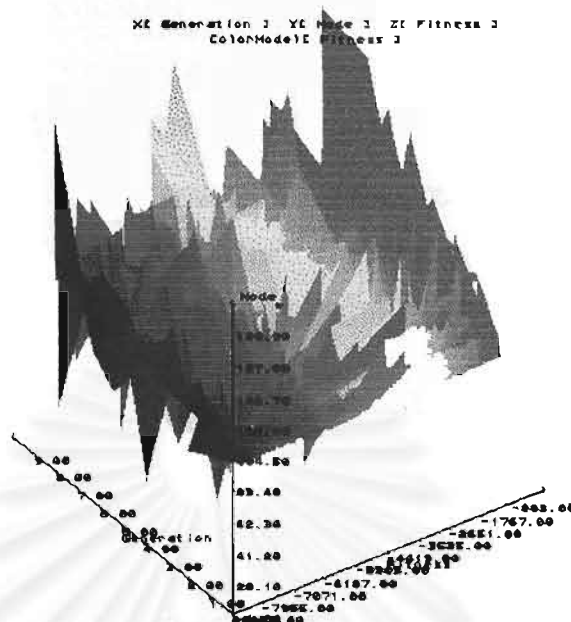
เลือกรายการ **System** และเลือก "Surface" ระบบจะแสดงกล่องโต้ตอบ ดังรูปที่

11



รูปที่ 11 กล่องโต้ตอบข้อมูลกำหนดการเชิงพันธุกรรมในรูปแบบ 3 มิติ

ให้ผู้ใช้งานกดแป้น **Process** ระบบจะแสดงการจินตทัศน์ดังรูปที่ 12



รูปที่ 12 แสดงความสัมพันธ์ข้อมูลกำหนดการเชิงพันธุกรรมในรูปแบบ 3 มิติ

สำหรับรายละเอียดเกี่ยวกับการจินตทัศน์และการปรับค่าต่างๆ ในการแสดงผลได้แสดงไว้ในหัวข้อ 5.9

3.7 การจัดเก็บภาพ

ผู้ใช้สามารถจัดเก็บภาพในหน้าต่าง VGL ด้วยแป้น F12 ระบบจะจัดเก็บเป็นแฟ้มภาพ TIFF หากต้องการนำภาพที่ได้ไปใช้ในงานพิมพ์ระบบสามารถปรับสีพื้นด้านหลังเป็นสีขาว หรือสีดำด้วยแป้น F11 รายละเอียดได้แสดงไว้ในหัวข้อ 5.11

3.8 ออกจากโปรแกรม

การออกจากโปรแกรมทำได้โดยเลือกรายการ **System** และเลือก "Exit" ระบบจะปิดหน้าต่างทั้งหมดและจบการทำงาน

บทที่ 4

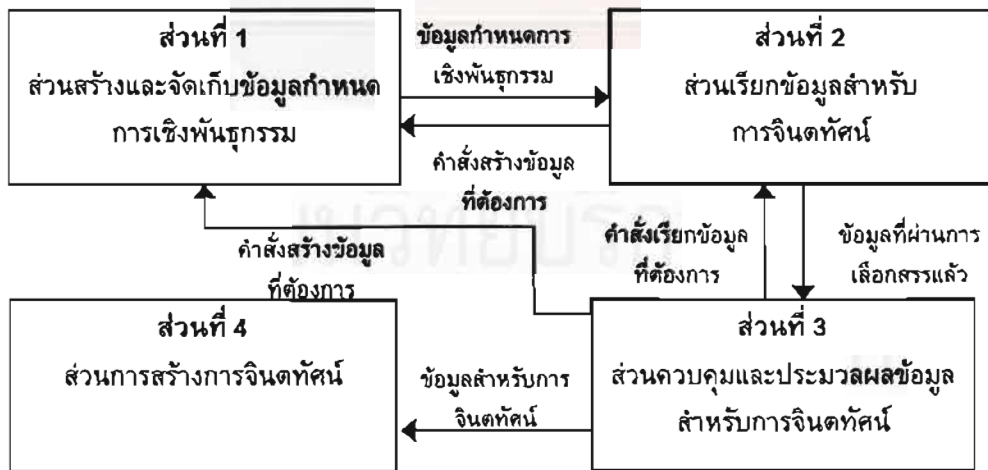
โปรแกรมจินตทัศน์ข้อมูลกำหนดการเชิงพันธุกรรม

ระบบ GPVIS ได้ถูกออกแบบให้ทำงานบน X-Windows ของระบบปฏิบัติการ Linux Redhat เวอร์ชัน 5.0 ใช้ GTK* [11] ในการสร้างเมนูและกล่องโต้ตอบต่างๆ (DialogBox) สำหรับเป็นส่วนต่อประสานกราฟิกกับผู้ใช้ (GUI) เพื่อรับข้อมูลจากผู้ใช้และเรียกใช้คลังโปรแกรม MESA [12] ซึ่งเป็น OpenGL† บนระบบปฏิบัติการ Linux สำหรับการแสดงผลต่างๆ ระบบ GPVIS จะนำข้อมูลกำหนดการเชิงพันธุกรรมที่ได้มาจากโปรแกรมกำหนดการเชิงพันธุกรรมสำหรับควบคุมแขนหุ่นยนต์ของชัยวัฒน์ เจริญภาพกรณ์ เป็นข้อมูลสำหรับมาใช้ในการจินตทัศน์

ระบบ GPVIS แบ่งโปรแกรมในการเรียกใช้งานออกเป็น 3 โปรแกรมคือ โปรแกรม VGL, โปรแกรม VGLCON และโปรแกรม CONSOLE

4.1 โปรแกรม VGL

เป็นส่วนใหญ่ของระบบทั้งหมด ซึ่งประกอบไปด้วย ส่วนต่างๆ 4 ส่วน คือ ส่วนสร้างและจัดเก็บข้อมูลกำหนดการเชิงพันธุกรรม ส่วนเรียกข้อมูลสำหรับการจินตทัศน์ ส่วนควบคุมและประมวลผลข้อมูลสำหรับการจินตทัศน์และส่วนการสร้างการจินตทัศน์ ดังรูปที่ 13



รูปที่ 13 ระบบ GPVIS

* GTK (GIMP Toolkit) เป็นส่วนโปรแกรมช่วยในการสร้างส่วนต่อประสานกราฟิกกับผู้ใช้บน X-Windows

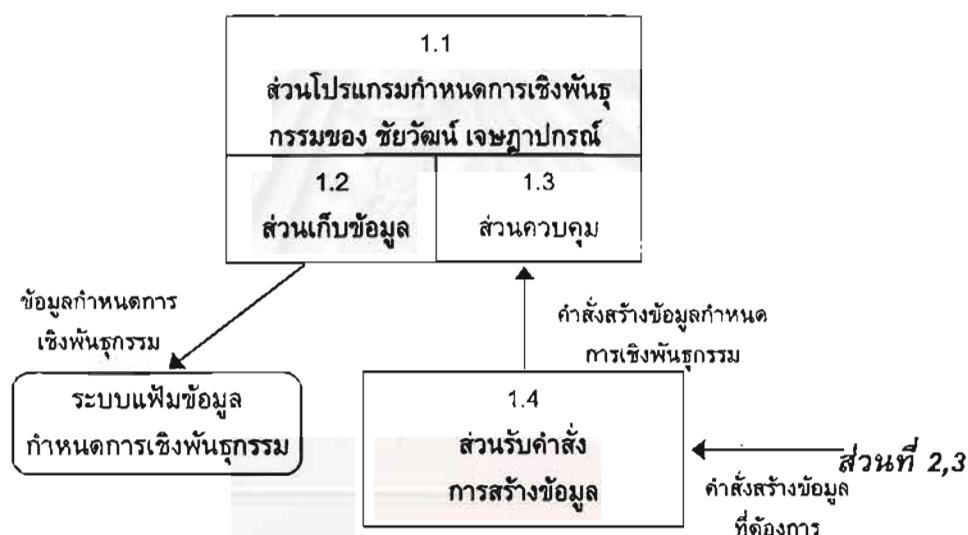
† OpenGL เป็นมาตรฐานของส่วนโปรแกรมช่วยแสดงผลด้านกราฟิก 3 มิติ

4.2 ส่วนต่างๆ ของโปรแกรม VGL

โปรแกรม VGL แบ่งออกเป็นส่วนต่างๆ ได้ดังนี้

4.2.1 ส่วนที่ 1 ส่วนสร้างและจัดเก็บข้อมูลกำหนดการแข่งขันฟุตบอล

ทำหน้าที่สร้างและจัดเก็บข้อมูลที่ได้จากโปรแกรมกำหนดการแข่งขันฟุตบอลของชัยวัฒน์ เจริญภาพกรณ์ โดยรับคำสั่งการสร้างข้อมูลจากส่วนที่ 2 และเรียกใช้ส่วนโปรแกรมของเดิม และคอยจัดเก็บข้อมูลที่เกิดขึ้นในระหว่างการพัฒนาการกำหนดการแข่งขันฟุตบอล โดยเพิ่มการจัดเก็บรายละเอียดต่างๆ ที่จำเป็นสำหรับการจินตทัศน์ ด้วย เช่น การพัฒนาการด้วยวิธีใด ที่จุดใด



รูปที่ 14 โปรแกรม VGL ส่วนที่ 1 ส่วนสร้างและจัดเก็บข้อมูลกำหนดการแข่งขันฟุตบอล

ส่วนที่ 1.1 ส่วนโปรแกรมกำหนดการแข่งขันฟุตบอลของชัยวัฒน์ เจริญภาพกรณ์ ซึ่งได้ถูกแก้ไขโปรแกรมบางส่วนเพื่อให้มีความสามารถในการจัดเก็บข้อมูลในระหว่างการทำงานผ่านขั้นตอนต่างๆ ของกระบวนการพัฒนาการในกำหนดการแข่งขันฟุตบอล โดยเรียกใช้ส่วนที่ 1.2

ส่วนที่ 1.2 ส่วนเก็บข้อมูลกำหนดการแข่งขันฟุตบอลที่ได้จากโปรแกรมกำหนดการแข่งขันฟุตบอลของชัยวัฒน์ เจริญภาพกรณ์ ทำหน้าที่ ในการเก็บข้อมูลรายละเอียดต่างๆ สำหรับการจินตทัศน์ข้อมูลอยู่ในรูปแฟ้มอักขระ (Text File)

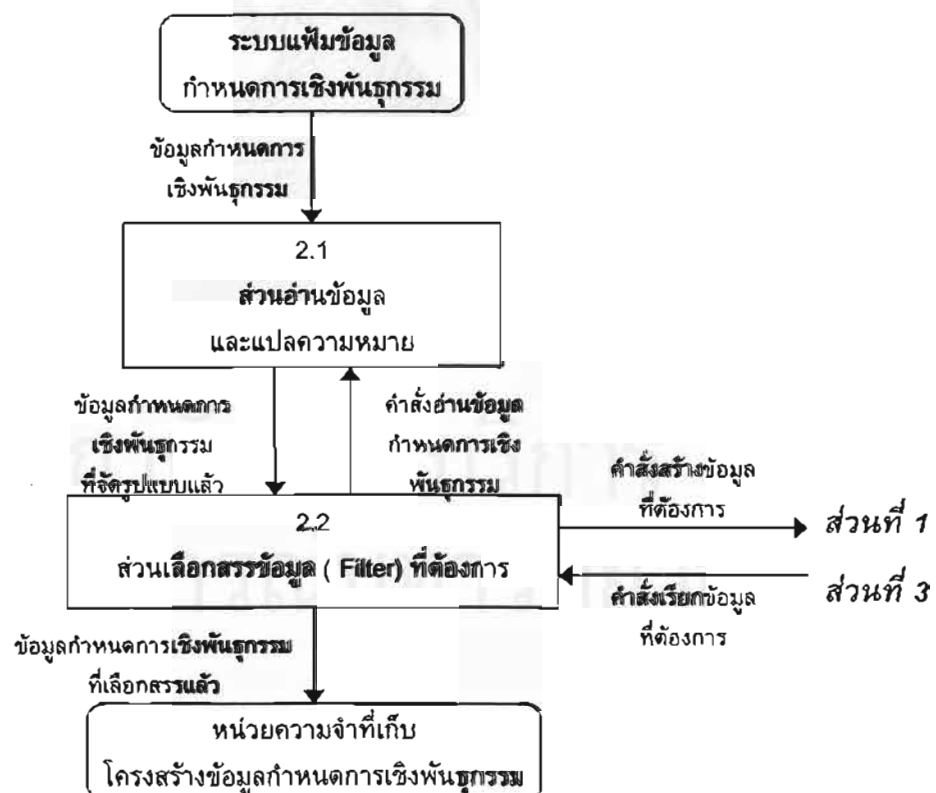
ส่วนที่ 1.3 ส่วนควบคุมทำหน้าที่ควบคุมและเรียกใช้งานในระดับฟังก์ชันย่อยต่างๆ ของโปรแกรมกำหนดการเชิงพันธุกรรมของชัยวัฒน์ เจริญปกรณ์ เพื่อให้โปรแกรมทำการวิวัฒนาการและสร้างข้อมูลตามที่ระบบจินตทัศน์ต้องการ

ส่วนที่ 1.4 ส่วนรับคำสั่งสร้างข้อมูล ทำหน้าที่ใช้เรียกส่วนที่ 3 ด้วยค่าของพารามิเตอร์ที่ได้รับมาจากระบบจินตทัศน์

สำหรับรายละเอียดเกี่ยวกับระบบเพิ่มข้อมูล สามารถดูได้จากภาคผนวก

4.2.2 ส่วนที่ 2 ส่วนเรียกข้อมูลสำหรับการจินตทัศน์

ทำหน้าที่ส่งคำสั่งสร้างข้อมูลไปยังส่วนที่ 1 ในกรณีที่ข้อมูลที่ร้องขอไม่ได้ถูกจัดเก็บไว้ และทำหน้าที่อ่านข้อมูลกำหนดการเชิงพันธุกรรมซึ่งได้ทำการจัดเก็บข้อมูลโดยส่วนที่ 1 มาয়งหน่วยความจำของระบบจินตทัศน์ ซึ่งจำแนกตามข้อมูลที่ต้องการ และทำการกรองข้อมูลที่ไม่ต้องการออกตามเงื่อนไขในโปรแกรมโดยจะนำเฉพาะข้อมูลที่มีความจำเป็นต่อการจินตทัศน์ในเวลานั้นมาเก็บในหน่วยความจำ



รูปที่ 15 โปรแกรม VGL ส่วนที่ 2 ส่วนเรียกข้อมูลสำหรับการจินตทัศน์

ส่วนที่ 2.1 ส่วนอ่านข้อมูลและแปลความหมาย ทำหน้าที่อ่านข้อมูลกำหนดการเชิงพันธุกรรมจากแฟ้มข้อมูลอักขระ (Text File) และเปลี่ยนอักขระที่อ่านขึ้นมาเป็นข้อมูลตัวเลข และจัดรูปแบบในลักษณะของระเบียบ (Record) และส่งข้อมูลให้ส่วนที่ 2.2

ส่วนที่ 2.2 ส่วนเลือกสรรข้อมูล ทำหน้าที่กำหนดให้ส่วนที่ 2.1 อ่านข้อมูลและรอรับข้อมูลที่จัดรูปแบบแล้วจากส่วนที่ 2.1 มาคัดเลือกให้ตรงกับเงื่อนไขตามที่ต้องการ และจัดเก็บลงในหน่วยความจำส่วนทั่วไปที่สามารถเข้าถึงได้จากทุกๆ ส่วนในโปรแกรม ซึ่งเป็นโครงสร้างข้อมูลกำหนดการเชิงพันธุกรรมที่เตรียมไว้เพื่อจะนำไปใช้ในการจินตทัศน์ต่อไป

4.2.3 ส่วนที่ 3 ส่วนควบคุมและประมวลผลข้อมูลสำหรับการจินตทัศน์

ทำหน้าที่รับคำสั่งจากผู้ใช้ จากนั้นจะนำคำสั่งที่ได้มาวิเคราะห์และเรียกใช้ส่วนต่างๆ ของโปรแกรมใช้งานและควบคุมการทำงานของระบบ GPVIS ในการตัดสินใจในการอ่านข้อมูลและการแสดงผลการจินตทัศน์ข้อมูลในรูปแบบต่างๆ ตามที่ผู้ใช้ต้องการ

ส่วนที่ 3.1 ส่วนรับคำสั่ง ทำหน้าที่รับคำสั่งจากผู้ใช้จากระบบ GPVIS โดยผ่านทาง 3 ทาง คือ

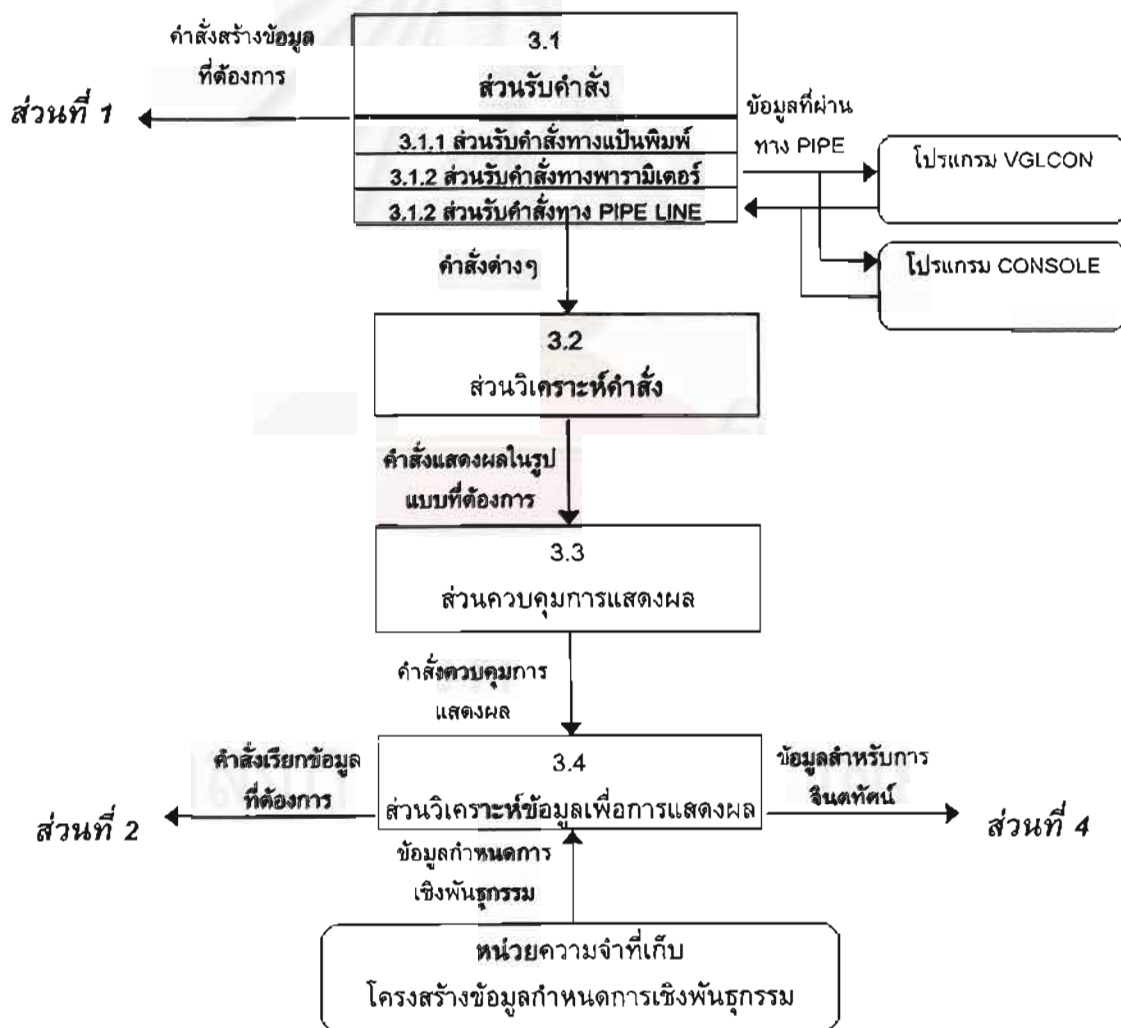
ส่วนที่ 3.1.1 ส่วนรับคำสั่งทางแป้นพิมพ์ จากหน้าจอ OpenGL ซึ่งจะรับเป็นตัวอักษร ตัวอักษรแต่ละตัวจะถูกตีความหมายเป็นแต่ละคำสั่งเฉพาะตามที่ได้กำหนดไว้ในโปรแกรม เช่น อักษร t หมายถึงให้โปรแกรมแสดงแผนภาพความสัมพันธ์ระหว่างการพัฒนาการตั้งแต่เริ่มต้นจนถึงคำตอบปัจจุบัน อักษร * ในทำการวาดหน้าจอใหม่ (Refresh) เป็นต้น*

ส่วนที่ 3.1.2 ส่วนรับคำสั่งทางพารามิเตอร์ เมื่อผู้ใช้เรียกโปรแกรมให้ทำงานและมีพารามิเตอร์ต่อท้าย ในส่วนนี้ใช้ในกรณีที่ผู้ใช้ต้องการสร้างข้อมูลกำหนดการเชิงพันธุกรรมเริ่มต้นสำหรับระบบ GPVIS ในสภาพแวดล้อมและเงื่อนไขที่ผู้ใช้กำหนด โดยโปรแกรมจะทำการสร้างข้อมูลตามที่ผู้ใช้กำหนดให้และออกจากโปรแกรมเมื่อโปรแกรมได้สร้างข้อมูลทั้งหมด โดยไม่เข้าระบบการจินตทัศน์

* รายละเอียดของคำสั่งควบคุมต่างๆ และเป็นลัดแสดงไว้ในภาคผนวก

ส่วนที่ 3.1.2 นี้ ในโปรแกรมจะเป็นส่วนของโปรแกรมที่ทำงานแยกจากส่วนที่ 3 กล่าวคือ ส่วนของโปรแกรมส่วนนี้จะถูกทำงานเพื่อตรวจสอบคำสั่งทางพารามิเตอร์ก่อนเป็นอันดับแรกเมื่อโปรแกรมถูกเรียกใช้งาน และจะไม่กลับมาเรียกส่วนนี้อีก เนื่องจากความเหมาะสมในการสร้างข้อมูลซึ่งใช้เวลาในการสร้างนาน แต่ได้รวมไว้ในส่วนที่ 3 เนื่องจากเป็นวิธีการรับคำสั่งจากผู้ใช้งานวิธีหนึ่ง

ส่วนที่ 3.1.3 ส่วนรับคำสั่งทาง PIPE LINE เมื่อเรียกใช้งานระบบโดยเรียกโปรแกรม VGLCON (หรือโปรแกรม CONSOLE) เมื่อโปรแกรมเริ่มต้นทำงาน โปรแกรมจะทำการ FORK PROCESS และเรียกโปรแกรม VGL มาทำงานพร้อมกันและโปรแกรมจะทำการสร้าง PIPE LINE เพื่อให้โปรแกรมสามารถสื่อสารกันได้ซึ่งกันและกัน และเมื่อผู้ใช้เลือกเมนูที่ต้องการ หรือเปลี่ยนแปลงค่าในกล่องโต้ตอบ โปรแกรม VGLCON จะส่งคำสั่งผ่านทาง PIPE เพื่อบอกให้โปรแกรมทำงาน



รูปที่ 16 โปรแกรม VGL ส่วนที่ 3 ส่วนควบคุมและประมวลผลข้อมูลสำหรับการจินตทัศน์

ส่วนที่ 3.2 ส่วนวิเคราะห์คำสั่ง คำสั่งที่ได้จากผู้ใช้งานจะถูกตีความและปรับสถานะ (Flag/Status) การแสดงผลการจินตทัศน์ตามคำสั่งนั้นๆ สำหรับคำสั่งที่ได้จากส่วนที่ 3.1.2 ส่วนรับคำสั่งทางพารามิเตอร์ โปรแกรมจะไม่เรียกใช้ในส่วนนี้ โปรแกรมจะส่งคำสั่งสร้างข้อมูลไปยังส่วนที่ 1 โดยตรง ตามรายละเอียดในส่วนที่ 3.1.2

ส่วนที่ 3.3 ส่วนควบคุมการแสดงผล เป็นส่วนที่สำคัญสำหรับระบบจินตทัศน์ข้อมูล เนื่องจากเป็นส่วนควบคุมการเรียกใช้งานส่วนการจินตทัศน์ตามรูปแบบต่างๆ ในโปรแกรมตามสถานะการแสดงผลจินตทัศน์ที่ได้กำหนดมาจากส่วนที่ 3.2 เมื่อมีการเปลี่ยนสถานะการแสดงผลจินตทัศน์ขึ้น ส่วนควบคุมการแสดงผลจะส่งคำสั่งควบคุมการแสดงผลไปยังส่วนที่ 3.4 และส่วนควบคุมการแสดงผลจะยกเลิก (Cancel) หรือจัดเก็บสถานะชั่วคราว (Save Visualize Status) ของการจินตทัศน์ที่แสดงผลอยู่ และปรับลักษณะการจินตทัศน์และสร้างข้อมูลของการจินตทัศน์ใหม่ หรือเรียกสถานะของการจินตทัศน์ที่ได้จัดเก็บไว้เดิม (Restore Visualize Status) มาแสดงผล ตัวอย่างเช่น เมื่อระบบแสดงผลการจินตทัศน์โครงสร้างของความสัมพันธ์ระหว่างคำตอบที่เกิดจากการวิวัฒนาการ เมื่อผู้ใช้ดูและคำตอบเลือกในความสัมพันธ์ดังกล่าว และกดแป้นพิมพ์ 'v' เพื่อให้โปรแกรมแสดงโครงสร้างของคำตอบในรูปแบบต้นไม้ ส่วนควบคุมการแสดงผลจะเก็บสถานะการจินตทัศน์โครงสร้างของความสัมพันธ์ระหว่างคำตอบที่เกิดจากการวิวัฒนาการไว้ (Save) และเริ่มต้นเรียกส่วนเริ่มต้น (Initialise)* ของจินตทัศน์โครงสร้างของคำตอบในรูปแบบต้นไม้ และส่งคำสั่งควบคุมการแสดงผลไปยังส่วนที่ 3.4 เพื่อให้สร้างข้อมูลที่ใช้สำหรับการจินตทัศน์โครงสร้างของคำตอบในรูปแบบต้นไม้ ส่วนควบคุมจะทำการวนลูปเพื่อเรียกส่วนแสดงผลการจินตทัศน์ และหลังจากที่ผู้ใช้ได้ดูการจินตทัศน์โครงสร้างของคำตอบในรูปแบบต้นไม้ แล้วเมื่อผู้ใช้ยกเลิกโดยกด 'ESC' ส่วนควบคุมการแสดงผลจะเรียกส่วนสิ้นสุด (Deinitialise)† ของการจินตทัศน์โครงสร้างของคำตอบในรูปแบบต้นไม้ และเรียกสถานะของการจินตทัศน์ที่ได้จัดเก็บไว้เดิม (Restore) จากนั้นผู้ใช้จะพบว่าโปรแกรมจะแสดงผลการจินตทัศน์โครงสร้างของความสัมพันธ์ระหว่างคำตอบที่เกิดจากการวิวัฒนาการด้วยข้อมูลชุดเดิม

ส่วนที่ 3.4 ส่วนวิเคราะห์ข้อมูลเพื่อการแสดงผล ส่วนนี้จะประกอบไปด้วยส่วนของการจินตทัศน์ในรูปแบบลักษณะต่างๆ กัน ซึ่งในการจินตทัศน์แต่ละแบบนั้นก็มีส่วนที่ทำหน้าที่ส่งคำสั่งเพื่อเรียกข้อมูลกำหนดการเชิงพันธุกรรมเฉพาะข้อมูลที่ต้องการตามสถานะการจินตทัศน์ในขณะนั้นจากส่วนที่ 2.2 และนำข้อมูลกำหนดการเชิงพันธุกรรมที่เก็บไว้ในหน่วย

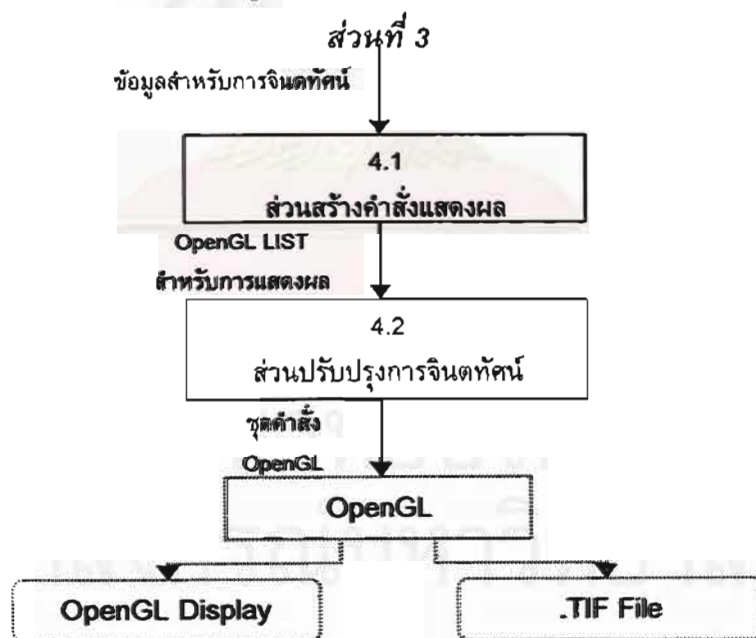
* ส่วนเตรียมการพิเศษสำหรับการตั้งค่าตัวแปรต่างๆ ให้พร้อมก่อนจะทำงานในส่วนต่อไป

† ส่วนเตรียมการพิเศษสำหรับการอ่านค่าสถานะที่ได้ภายหลังจากทำงานและปรับปรุ้งค่าตัวแปรต่างๆ ให้พร้อมก่อนจะทำงานในส่วนต่อไป

ความจำซึ่งได้มากจากการทำงานของส่วนที่ 2.2 มาวิเคราะห์ข้อมูลเพื่อสร้างข้อมูลเพิ่มเติมที่เหมาะสมกับการจินตทัศน์ในแต่ละแบบ เช่นการจินตทัศน์โครงสร้างของคำตอบในรูปแบบต้นไม้ คำตอบที่ 3 ในรุ่นที่ 1 ส่วนที่ 3.4 จะส่งคำสั่งเรียกรายละเอียดของข้อมูลคำตอบที่ 3 ในรุ่นที่ 1 ส่วนที่ 2.2 จะไปเรียกส่วนที่ 2.1 และเลือกข้อมูลคำตอบที่เป็นคำตอบที่ 3 ในรุ่นที่ 1 และส่งรายละเอียดคืนมาให้ ส่วนที่ 3.4 จากนั้นส่วนที่ 3.4 จะทำการสร้างข้อมูลแบบต้นไม้ในหน่วยความจำซึ่งประกอบด้วยประเภทของ Node ตำแหน่งของแต่ละ Node โดยใช้รายละเอียดของคำตอบที่ 3 เมื่อได้ข้อมูลแบบต้นไม้แล้วจะส่งต่อให้ส่วนที่ 4 เพื่อนำข้อมูลมาแสดงบนจอภาพ

4.2.4 ส่วนที่ 4 ส่วนการสร้างการจินตทัศน์

ทำหน้าที่รับคำสั่งและข้อมูลที่ต้องการสำหรับการจินตทัศน์จากส่วนที่ 3 จากนั้นนำข้อมูลที่ได้มาประมวลผล และแสดงการจินตทัศน์ในลักษณะที่ต้องการ ในรูปแบบต่างๆ ทั้ง 2 มิติ และ 3 มิติ ด้วยคำสั่งของ OpenGL เพื่อให้ OpenGL สร้างการจินตทัศน์ตามที่ได้ออกแบบไว้ หรือเก็บเป็นแฟ้มภาพ (Graphics File) ในฮาร์ดดิสก์ ซึ่งสามารถจัดเก็บภาพเดียวเมื่อผู้ใช้สั่งหรือเก็บเป็นแฟ้มภาพที่มีหมายเลขเรียงกันแบบต่อเนื่องได้ ในการจำลองการเคลื่อนไหวของแขนหุ่นยนต์ หากผู้ใช้ต้องการโดยอัตโนมัติ



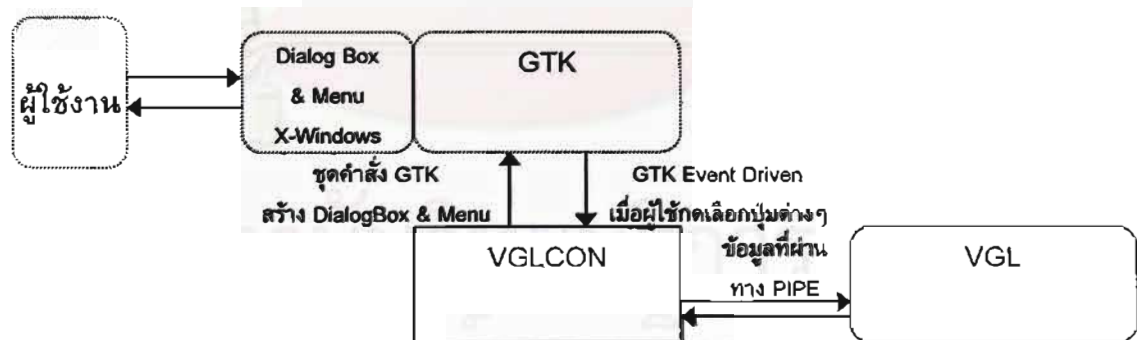
รูปที่ 17 โปรแกรม VGL ส่วนที่ 4 ส่วนการสร้างการจินตทัศน์

ส่วนที่ 4.1 ส่วนสร้างคำสั่งการแสดงผล ทำหน้าที่นำข้อมูลสำหรับการจินตทัศน์มาแสดงผลด้วยชุดคำสั่งของ OpenGL โดยใช้ความสามารถของ OpenGL ในการสร้างชุดคำสั่งเป็น LIST* โดยสร้างเพียงครั้งเดียวและเรียก LIST เมื่อต้องการแสดงผล

ส่วนที่ 4.2 ส่วนปรับปรุงการจินตทัศน์ เป็นส่วนที่แสดงผลเพิ่มเติมจากข้อมูล LIST ที่ได้จากส่วนที่ 4.1 โดยส่งคำสั่ง OpenGL ที่เรียกใช้ LIST ผสมกับคำสั่ง OpenGL ที่ใช้สำหรับวาดภาพ ซึ่ง LIST จะเป็นเพียงส่วนหนึ่งในการจินตทัศน์เท่านั้นเนื่องจากการใช้ LIST จะเหมาะกับการแสดงข้อมูลที่ไม่เปลี่ยนแปลงบ่อยๆ การแสดงสถานะต่างๆ บนจอภาพ จะเป็นหน้าที่ของส่วนที่ 4.2 รวมไปถึงการควบคุมการแสดงผลของ LIST เมื่อผู้ใช้ต้องการเลื่อนภาพ การขยายหรือการลดขนาด หรือหมุนดูข้อมูลในมุมที่ต่างจากเดิม

4.3 โปรแกรม VGLCON

เป็นส่วนที่รับคำสั่งจากผู้ใช้โดยใช้ GTK เป็นคลังโปรแกรม (Library) ในการแสดงรายการ และหน้าจอในการควบคุมโปรแกรม VGL ใช้ความสามารถของระบบ UNIX ในการ PIPE ข้อมูลระหว่าง 2 กระบวนการ (Process) คือโปรแกรม VGLCON และโปรแกรม VGL ในการรับส่งข้อมูล การส่งข้อมูลจะเป็นในลักษณะสองทางคือ เมื่อผู้ใช้เรียกเมนู ข้อมูลของชุดคำสั่งจะถูกส่งจากโปรแกรม VGLCON ไปยัง โปรแกรม VGL และจากนั้นโปรแกรม VGL จะรับคำสั่งไปประมวลผลเพื่อจินตทัศน์ตามคำสั่งนั้นๆ และจะส่งข้อมูลบางส่วนที่โปรแกรม VGLCON ต้องการกลับมา เพื่อให้โปรแกรม VGLCON นำมาแสดงในกล่องโต้ตอบผ่านทาง GTK

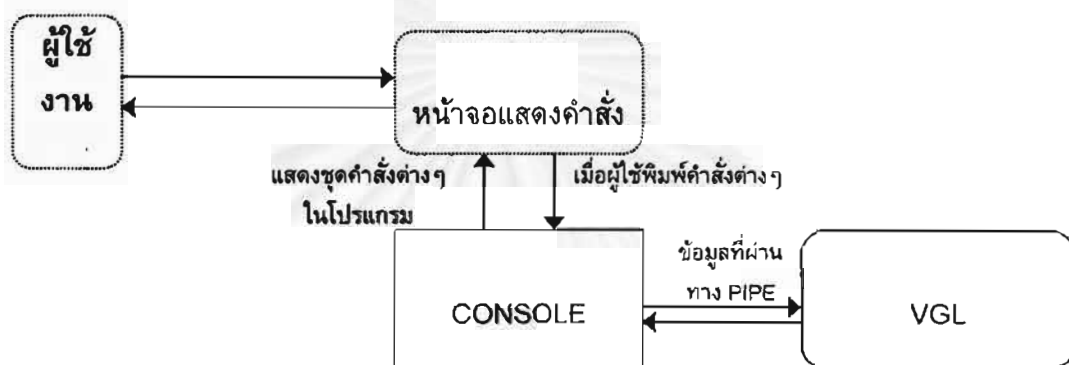


รูปที่ 18 โปรแกรม VGLCON และส่วนเชื่อมต่อ

* OpenGL มีความสามารถในการจัดเก็บชุดคำสั่งสร้างกราฟิกยาวๆ รวมกันเป็นคำสั่งเดียวไว้ในส่วนที่เรียกว่า LIST สำหรับ OpenGL และเมื่อต้องการใช้งานในภายหลังสามารถเรียกใช้ LIST แทนชุดคำสั่งนั้นได้ มีแนวคิดคล้ายกับการเรียกใช้กระบวนการ (Procedure) ในการเขียนโปรแกรมทั่วไป วิธีนี้มีข้อดีคือช่วยลดเวลาในการประมวลผลข้อมูลของ OpenGL จึงช่วยให้การแสดงผลรวดเร็วขึ้น

4.4 โปรแกรม CONSOLE

เป็นส่วนที่รับคำสั่งเช่นเดียวกับโปรแกรม VGLCON แต่ได้ออกแบบในลักษณะของชุดคำสั่งแบบ Command Line เนื่องจากพบว่าโปรแกรม VGLCON ไม่สามารถทำงานได้บนเครื่องที่ติดตั้งระบบปฏิบัติการที่ต่างรุ่น (Version) กันกับระบบปฏิบัติการที่ใช้พัฒนาโปรแกรม* ดังนั้นเพื่อความสะดวกในการควบคุมบางคำสั่งที่ไม่สามารถสั่งด้วยแป้นลัดได้ เช่น ระบบการเลือกแกนข้อมูลเพื่อการแสดงกราฟ หากผู้ใช้ไม่สามารถใช้งานโปรแกรม VGLCON ได้ผู้ใช้สามารถเรียกใช้โปรแกรม CONSOLE แทนได้ ด้วย Command Line



รูปที่ 19 โปรแกรม CONSOLE และส่วนเชื่อมต่อ

โปรแกรมทั้ง 3 โปรแกรมประกอบรวมกันเป็นระบบ GPVIS ซึ่งมีความสามารถต่างๆ ในการจินตทัศน์ข้อมูลกำหนดการเชิงพันธุกรรม เช่น การสร้างข้อมูลกำหนดการเชิงพันธุกรรมเพื่อจัดเก็บลงไฟล์ตามที่ผู้ใช้กำหนดค่าพารามิเตอร์ให้กับโปรแกรม การแสดงข้อมูลกำหนดการเชิงพันธุกรรมจากไฟล์ที่ได้จัดเก็บไว้ มีส่วนที่ผู้ใช้สามารถเลือกดูเฉพาะคำตอบที่ต้องการได้ และส่วนกรอง (Filter) ข้อมูล มีการจำลองการเคลื่อนไหวของแขนหุ่นยนต์ (Simulation) จากคำตอบที่ผู้ใช้เลือก มีการแสดงความสัมพันธ์ระหว่างคำตอบที่เกิดจากกระบวนการวิวัฒนาการ การแสดงกระบวนการวิวัฒนาการของคำตอบ การแสดงข้อมูลกำหนดการเชิงพันธุกรรมในรูปแบบ 3 มิติโดยผู้ใช้สามารถเลือกข้อมูลที่ต้องการดูในแต่ละแบบได้ตามต้องการ มีฟังก์ชันเก็บภาพที่ปรากฏในระบบ GPVIS เป็นเพิ่มภาพได้ และ

* ระบบปฏิบัติการที่มีรุ่นต่างกันจะมีคลังโปรแกรม (Library) ต่างรุ่นเช่นเดียวกัน บางคลังโปรแกรมจะปรับปรุงเพิ่มเติมส่วนฟังก์ชันใหม่และลดส่วนฟังก์ชันเดิมออก ซึ่งมีส่วนทำให้โปรแกรมที่แปลโปรแกรม (Compile) โดยใช้ข้อกำหนดของคลังโปรแกรมแบบเดิมไม่สามารถใช้งานได้

สามารถเก็บภาพแบบต่อเนื่องในส่วนของการจำลองการเคลื่อนไหวของแขนหุ่นยนต์ได้ และอื่นๆ ซึ่งเป็นส่วนของรายละเอียดในแต่ละหัวข้อที่จะกล่าวในบทต่อไป

บทที่ 5

การจินตทัศน์ข้อมูลกำหนดการเชิงพันธุกรรมด้วยระบบจินต ทัศน์ข้อมูล GPVIS

ระบบ GPVIS มีความสามารถในการจินตทัศน์ข้อมูลกำหนดการเชิงพันธุกรรมหลากหลายรูปแบบตามที่ผู้ใช้งานต้องการจากข้อมูลที่สร้างเก็บไว้ ดังนั้นจึงแบ่งการทำงานของระบบ GPVIS ออกเป็น 2 ส่วนใหญ่คือ ส่วนสร้างข้อมูลกำหนดการเชิงพันธุกรรม และ ส่วนแสดงผลข้อมูลกำหนดการเชิงพันธุกรรม ซึ่งเป็นส่วนสำหรับจินตทัศน์ข้อมูล

5.1 การสร้างข้อมูลกำหนดการเชิงพันธุกรรม

เป็นส่วนสำหรับสร้างข้อมูลกำหนดการเชิงพันธุกรรม เพื่อจัดเก็บลงไฟล์ตามที่ผู้ใช้งานกำหนดค่าพารามิเตอร์ให้กับโปรแกรม โดยที่ค่าพารามิเตอร์จะเป็นตัวกำหนดสภาพแวดล้อม ขนาดเริ่มต้นของคำตอบ รูปแบบของพารามิเตอร์ที่กำหนดให้กับโปรแกรมจะมีรูปแบบเช่นเดียวกับกับรูปแบบพารามิเตอร์ที่ใช้กับโปรแกรมของชัยวัฒน์ เจริญภาพกรณ์ ซึ่งเดิมปรากฏอยู่ในไฟล์ readme.txt และได้นำมาปรับปรุงให้สอดคล้องกับโปรแกรม VGL ดังนี้

รูปแบบ

vgl %1 %2 %3 %4 %5 %6

%1 = สภาพแวดล้อมที่ (1-14)

%2 = ใช้ initial population generator ที่ออกแบบใหม่หรือใช้ตัวเก่าที่ใช้ในการวิจัยเดิม(1=ใหม่, 0=เก่า)

ในกรณีที่ใช้ initial population generator เก่าจะทำการทดลองได้เฉพาะวิธี gp ปกติเท่านั้น

%3 = ขนาดของโปรแกรมเริ่มต้น

%4 = จำนวนฟังก์ชัน ADFs น้อยที่สุดที่ใช้

%5 = จำนวนฟังก์ชัน ADFs มากที่สุดที่ใช้ (หากใช้วิธี ADFs ปกติ %4 และ %5 จะเป็นค่าเดียวกันเสมอ)

%6 = วิธีการ crossover (0=cross ระหว่างฟังก์ชันเดียวกัน (ADFs), 1=cross ต่างฟังก์ชันได้ (ADFX), 2=cross แบบ Point typing (ADFs + Evolution method))

* พารามิเตอร์ที่ %4 %5 %6 จะไม่ใช้ในโปรแกรมจินตทัศน์การวิวัฒนาการกำหนดการแข่งขัน GPVIS ดังนั้นผู้ใช้ต้องใส่เป็นค่า 0 แทน รายละเอียดดูได้จากวิทยานิพนธ์ของชัยวัฒน์ เจริญภาพกรณ์ [4]

ตัวอย่าง

```
vgl 1 1 80 0 0 0
```

ทำการทดลองสภาพแวดล้อมที่ 1 ใช้ initial population generator ใหม่
ขนาดเริ่มต้น 80 สัญลักษณ์ ใช้วิธี gp ปกติ(ไม่ใช้วิธี ADFs)

```
vgl 12 1 40 0 0 0
```

ทำการทดลองสภาพแวดล้อมที่ 12 ใช้ initial population generator ใหม่
ขนาดเริ่มต้น 40 สัญลักษณ์ ใช้วิธี gp ปกติ(ไม่ใช้วิธี ADFs)

เมื่อเรียกโปรแกรม VGL ตามด้วยค่าพารามิเตอร์ โปรแกรมจะสร้างไดเรกทอรีใหม่ภายใต้ไดเรกทอรี data เช่นเมื่อเรียก vgl 1 1 80 0 0 0 โปรแกรมจะสร้างไดเรกทอรี data/1180000 จากนั้นโปรแกรมจะทำการสุ่มสร้างคำตอบกำหนดการแข่งขันพันธุกรรมเริ่มต้นตามขนาดที่กำหนด ในสภาพแวดล้อมที่กำหนด และให้กำหนดการแข่งขันพันธุกรรมทำการวิวัฒนาการคำตอบ และเก็บข้อมูลในระหว่างการวิวัฒนาการสำหรับการจินตทัศน์ไว้ในไดเรกทอรีที่ได้สร้างไว้

ระบบจินตทัศน์ข้อมูลไม่ได้ออกแบบไว้สำหรับการจินตทัศน์ข้อมูลที่เกี่ยวข้องกับ ADF และ ADFX รวมไปถึงสิ่งที่เกี่ยวข้องกับ Effort ในงานวิจัยของชัยวัฒน์ เจริญภาพกรณ์ ดังนั้นในการสร้างข้อมูลสำหรับการจินตทัศน์นั้น ผู้ใช้ต้องใส่ค่า 0 ให้กับพารามิเตอร์ในส่วนที่ควบคุม ADF และ ADFX เพื่อไม่ให้โปรแกรมทำการสร้างข้อมูลกำหนดการแข่งขันพันธุกรรมที่ใช้ ADF และ ADFX

5.2 การจินตทัศน์ข้อมูลกำหนดการแข่งขันพันธุกรรม

ในระบบ GPVIS ผู้ใช้สามารถเลือกดูการจินตทัศน์ในรูปแบบต่างๆ จากระบบจินตทัศน์นี้ได้ตามที่ได้อธิบายไว้ในบทที่ 3 สำหรับในบทนี้จะกล่าวถึงการใช้งานระบบ GPVIS

ด้วยโปรแกรม VGLCON กับ VGL หากผู้ใช้งานไม่สามารถเรียกใช้โปรแกรม VGLCON ได้ ขอให้ดูคำสั่งใช้งานโปรแกรม CONSOLE ซึ่งเป็นแบบ Command Line ที่สามารถใส่ทดแทนคำสั่งจากโปรแกรม VGLCON ได้ที่ภาคผนวก

สำหรับการจินตทัศน์ข้อมูลคำตอบที่ได้จากกำหนดการเชิงพันธุกรรมแบ่งการทำงานออกเป็นส่วนต่างๆ ซึ่งผู้ใช้สามารถสลับการทำงานในแต่ละส่วนได้ตามต้องการ แบ่งออกเป็นหัวข้อดังนี้

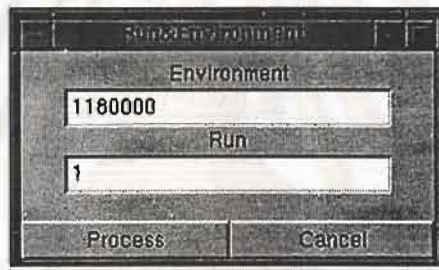
- การเลือกสภาพแวดล้อม
- การเลือกคำตอบที่ได้จากกำหนดการเชิงพันธุกรรม
- การแสดงรายละเอียดข้อมูลคำตอบที่ได้จากกำหนดการเชิงพันธุกรรม
- การแสดงคำตอบที่ได้จากกำหนดการเชิงพันธุกรรม ในรูปแบบโครงสร้างต้นไม้
- การจำลองการเคลื่อนไหวของแขนหุ่นยนต์จากคำตอบที่ได้จากกำหนดการเชิงพันธุกรรม
- การแสดงและเลือกคำตอบจากความสัมพันธ์ระหว่างคำตอบที่เกิดจากการวิวัฒนาการ ในกำหนดการเชิงพันธุกรรม
- การแสดงกระบวนการวิวัฒนาการของคำตอบในกำหนดการเชิงพันธุกรรม
- การตั้งค่าสำหรับกรองข้อมูลสำหรับการแสดงข้อมูลกำหนดการเชิงพันธุกรรมในรูปแบบ 3 มิติ
- การแสดงข้อมูลกำหนดการเชิงพันธุกรรมในรูปแบบ 3 มิติ
- การปรับเปลี่ยนสถานะการแสดงผล และ การเก็บภาพที่ปรากฏในระบบ GPVIS เป็นไฟล์

ในแต่ละหัวข้อมียละเอียดดังนี้

5.3 การเลือกสภาพแวดล้อม

สภาพแวดล้อมในระบบจินตทัศน์ข้อมูล หมายถึง สภาพแวดล้อมของสิ่งกีดขวางและเป้าหมาย การกำหนดจำนวนขนาดของคำตอบเริ่มต้น จากค่าพารามิเตอร์ที่ผู้ใช้กำหนดในการสร้างข้อมูลสำหรับการจินตทัศน์ ซึ่งค่าต่างๆ อ้างอิงจากงานวิจัยของชัยวัฒน์ เจริญภาพกรณ์ สภาพแวดล้อมในระบบจินตทัศน์ข้อมูลจะรวมไปถึงค่าของ run แต่ละ run ใน

สภาพแวดล้อมนั้นด้วย การเริ่มต้น run คือการสุ่มสร้างคำตอบเริ่มต้นและนำคำตอบที่ได้ผ่านกระบวนการวิวัฒนาการของกำหนดการเชิงพันธุกรรมจนถึงรุ่นที่ได้กำหนดไว้ (ในที่นี้กำหนดไว้ 10 รุ่น) เมื่อครบจำนวนรุ่นที่ได้กำหนดไว้ โปรแกรมจะเริ่มต้น run ใหม่คือการสุ่มสร้างคำตอบเริ่มต้นใหม่และนำคำตอบที่ได้ผ่านกระบวนการวิวัฒนาการจนถึงรุ่นที่ได้กำหนดไว้ ทำเช่นนี้ไปเรื่อยๆ ในสภาพแวดล้อมเดียวกันจนครบจำนวน run ที่ได้กำหนดไว้ในข้อมูลกำหนดการเชิงพันธุกรรมที่เก็บไว้ในแต่ละสภาพแวดล้อมหนึ่งจึงมี run หลายๆ run เนื่องจากกำหนดการเชิงพันธุกรรมมีการสุ่มเป็นพื้นฐานในการค้นหาคำตอบ ในสภาพแวดล้อมเดียวกันบาง run อาจจะได้คำตอบที่ดี หรือบาง run อาจจะไม่ได้อะไรเลยก็ได้ โดยแต่ละ run ข้อมูลกำหนดการเชิงพันธุกรรมจะไม่มีเกี่ยวข้องกัน ดังนั้นระบบจินตทัศน์ข้อมูลจะให้ผู้ใช้กำหนดว่าต้องการ run ที่เท่าไร จากสภาพแวดล้อมที่เลือก ข้อมูลที่จะนำมาใช้ในการจินตทัศน์จะอยู่ใน run และสภาพแวดล้อมที่กำหนดเท่านั้น เมื่อผู้ใช้งานกำหนดสภาพแวดล้อมที่ต้องการแล้ว และกด Process ระบบจินตทัศน์จะยกเลิกข้อมูลเดิม และอ่านข้อมูลกำหนดการเชิงพันธุกรรมตามสภาพแวดล้อมใหม่



รูปที่ 20 แสดงหน้าจอการเลือกสภาพแวดล้อม

5.4 การเลือกคำตอบที่ได้จากกำหนดการเชิงพันธุกรรม

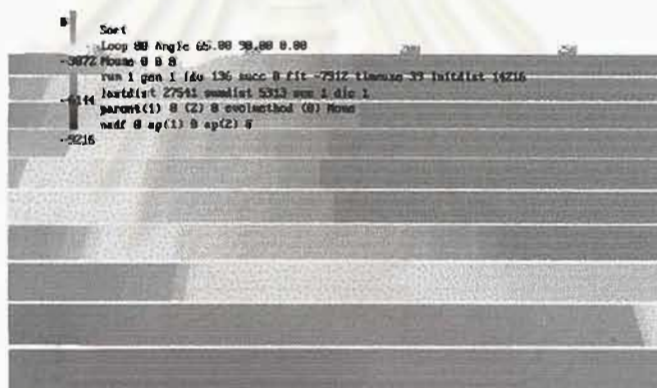
ระบบ GPVIS จะแสดงคำตอบที่ได้จากกำหนดการเชิงพันธุกรรมทั้งหมด จากสภาพแวดล้อมและหมายเลข run ที่กำหนดไว้ เมื่อระบบเริ่มต้นทำงาน ระบบจะแสดงการเลือกคำตอบที่ได้จากกำหนดการเชิงพันธุกรรมนี้เป็นสิ่งแรก โดยกำหนดค่าตั้งต้นให้แสดงที่สภาพแวดล้อมที่ 1180000* และ run ที่ 1

ในส่วนของการเลือกคำตอบนี้สามารถทำได้ 2 วิธีคือ เลือกคำตอบจากโปรแกรม VGL หรือเลือกโดยเรียกใช้เมนูจากโปรแกรม VGLCON ซึ่งจะแสดงกล่องโต้ตอบ

* หมายเลขของสภาพแวดล้อมเรียกตามค่าพารามิเตอร์การสร้างข้อมูลกำหนดการเชิงพันธุกรรมในหัวข้อ 5.1

5.4.1 การเลือกคำตอบจากโปรแกรม VGL

โปรแกรม VGL ใช้หน้าต่างของ OpenGL แสดงตัวคำตอบไว้เป็น แถวของแถบสีต่างๆ มีทั้งหมดจำนวน 10 แถว โดยแต่ละแถวแทนความหมายของแต่ละรุ่น (Generation) ของคำตอบ แถวที่อยู่ด้านหน้าสุดจะเป็นคำตอบรุ่นที่ 1 แถวที่อยู่ด้านหลังสุดจะเป็นคำตอบรุ่นที่ 10 ซึ่งเป็นคำตอบรุ่นสุดท้าย สีที่ใช้ในการแสดงผลในส่วนการเลือกคำตอบที่ได้จากกำหนดการเชิงพันธุกรรมนี้ จะใช้สีเขียวแทนคำตอบที่มีค่า fitness ที่ดี และไล่โทนสีไปยัง สีเหลือง สีส้ม สีแดง สีแดงเข้ม และสีดำ แทนค่าของ fitness ที่ลดลงขึ้นเรื่อยๆ ตามลำดับ โดยปรกติแล้วโปรแกรมจะเรียงลำดับค่า fitness ของคำตอบในแต่ละรุ่น ทำให้ผู้ใช้งานเห็นเป็นแถบสีไล่เรียงกันจาก สีดำ สีแดงเข้ม ไปยัง สีเขียว สังเกตได้ว่าในคำตอบรุ่นแรกๆ จะมีส่วนที่เป็นสีแดงและสีเหลืองมาก ส่วนคำตอบรุ่นหลังๆ จะมีส่วนที่เป็นสีเขียวเพิ่มมากขึ้นซึ่งแสดงให้เห็นว่าค่า fitness ของคำตอบในรุ่นหลังๆ จะมีค่าโดยรวมดีกว่าค่า fitness ของคำตอบในรุ่นแรกๆ ดังรูปที่ 21



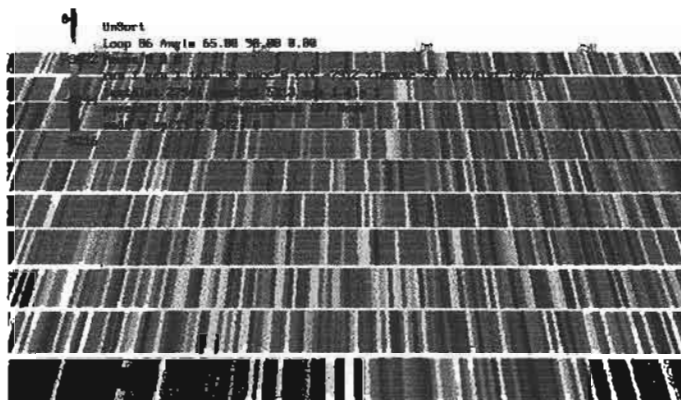
```

Soet
Loop 80 Angle 65.88 98.88 8.88
-3872 Mouse 8 8 8
run 1 gen 1 fit 136 succ 8 fit -7512 timmax 33 initfit 18216
-3144
parents (1) 8 (2) 8 evolmethod (8) Mouse
off 8 sp(1) 8 sp(2) 8
-2216
  
```

รูปที่ 21 แสดงหน้าจอการเลือกคำตอบที่ได้จากกำหนดการเชิงพันธุกรรม

ผู้ใช้งานสามารถเลือกไม่ให้โปรแกรมแสดงการเรียงลำดับค่า fitness ของคำตอบได้ด้วยการกดปุ่ม 's' โปรแกรมจะแสดงคำตอบแต่ละรุ่นโดยเรียงตามลำดับของหมายเลขคำตอบที่เป็นลำดับที่เกิดจากการสร้างคำตอบนั้นๆ การแสดงผลโดยไม่ให้โปรแกรมเรียงลำดับค่า fitness ของคำตอบนี้ จะเห็นการสลับของสีต่างๆ ซึ่งไม่แน่นอนดังรูปที่ 22 เป็นการจินตทัศน์จากสภาพแวดล้อมที่ 1180000 เช่นเดียวกับกับรูปที่ 21 คำตอบรุ่นที่ 1 ทั้งหมดจะเกิดจากการผสมสร้าง ส่วนที่เป็นคำตอบรุ่นอื่นๆ จะเกิดจากกระบวนการวิวัฒนาการโดยคำตอบหมายเลข 1-40 จะเกิดจากการ reproduction จากคำตอบรุ่นก่อนหน้านั้น คำตอบหมายเลข

41-200 จะเกิดจากการ crossover คำตอบที่ 201-300 จะเกิดจากการ addition และคำตอบที่ 301-400 จะเกิดจากการ addition จากคำตอบรุ่นก่อนหน้านั้น



รูปที่ 22 แสดงหน้าจอการเลือกคำตอบที่ได้จากกำหนดการเชิงพันธุกรรมโดยไม่เรียงลำดับค่า fitness ของคำตอบกำหนดการเชิงพันธุกรรมในแต่ละรุ่น

โปรแกรมจะแสดง "กล่อง" สีเหลี่ยมสี่มุม วางอยู่บนแถบสีที่เป็นตัวแทนของคำตอบปัจจุบันที่ต้องการเลือก ผู้ใช้งานสามารถเลื่อนกล่อง ไปยังคำตอบถัดไปในคำตอบรุ่นเดียวกันโดยใช้ปุ่ม 'w' หรือเลื่อนไปยังคำตอบก่อนหน้านั้นโดยใช้ปุ่ม 'd' และเลื่อนกล่องเพื่อเลือกคำตอบในรุ่นถัดไปจากคำตอบเดิมด้วยปุ่ม 'r' หรือกดปุ่ม 'e' เพื่อเลื่อนกล่องไปยังคำตอบรุ่นก่อนหน้านั้น เมื่อเลื่อนกล่องมาถึงขอบของแถบสี โปรแกรมจะนำกล่องไปวางไว้ที่ขอบด้านตรงข้ามของแถบสีโดยอัตโนมัติเพื่อความสะดวกในการเลือก

5.4.2 การเรียกใช้เมนูเลือกคำตอบกำหนดการเชิงพันธุกรรม

เมื่อผู้ใช้งานเลือกเมนู "Current Individual" จากโปรแกรม VGLCON โปรแกรมจะกล่องโต้ตอบดังรูปที่ 23 ซึ่งค่า Generation และ Individual คือค่า รุ่นของคำตอบ และหมายเลขของคำตอบ มีค่าเดียวกันกับตำแหน่งของกล่องในหัวข้อ 5.4.1 หากผู้ใช้งานเลื่อนกล่องโดยใช้ปุ่มควบคุมผ่านทางหน้าจอโปรแกรม VGL ค่าในกล่องโต้ตอบนี้จะปรับปรุงโดยอัตโนมัติเช่นเดียวกับการเปลี่ยนแปลงค่าในกล่องโต้ตอบและกด "set" โปรแกรม VGL จะปรับกล่องไปยังตำแหน่งใหม่

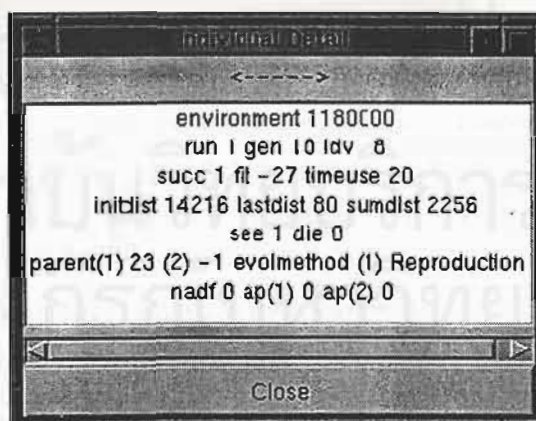
สำหรับค่า run ในกล่องโต้ตอบที่ผู้ใช้สามารถเปลี่ยนค่าจากส่วนนี้ได้และโปรแกรมจะเข้าสู่การเปลี่ยนค่าของ run ใหม่ ซึ่งมีผลเหมือนกับการเรียกหัวข้อ 5.4.1 ขึ้นมาใช้งานโดยไม่เปลี่ยนแปลงค่าสภาพแวดล้อม



รูปที่ 23 แสดงกล่องโต้ตอบสำหรับการเลือกค่าตอบที่ได้จากกำหนดการเชิงพันธุกรรม

5.4.3 การแสดงรายละเอียดข้อมูลค่าตอบที่ได้จากกำหนดการเชิงพันธุกรรม

โดยทั่วไปแล้ว หากโปรแกรมทำงานและแสดงผลอยู่ในส่วนที่ 5.4 การเลือกค่าตอบที่ได้จากกำหนดการเชิงพันธุกรรม โปรแกรมจะแสดงรายละเอียดข้อมูลของค่าตอบปัจจุบันโดยอัตโนมัติ หากผู้ใช้งานสลับให้โปรแกรมทำการจินตทัศน์กำหนดการเชิงพันธุกรรมในส่วนอื่น ที่โปรแกรมไม่แสดงรายละเอียดข้อมูลของค่าตอบ ผู้ใช้สามารถเรียกใช้งานกล่องโต้ตอบตามรูปที่ 23 และกดปุ่มคำสั่ง "Detail" โปรแกรมจะแสดงกล่องโต้ตอบที่มีรายละเอียดข้อมูลของค่าตอบดังรูปที่ 24



รูปที่ 24 กล่องโต้ตอบรายละเอียดของค่าตอบที่ได้จากกำหนดการเชิงพันธุกรรม

รายละเอียดต่างๆ มีความหมายตาราง 6 รายละเอียดของค่าตอบที่ได้จากกำหนดการเชิงพันธุกรรมดังนี้

ตาราง 6 รายละเอียดของค่าตอบที่ได้จากกำหนดการเชิงพันธุกรรม

รายละเอียด	ความหมาย
Environment	ค่าตอบนี้ถูกสร้างขึ้น ในสภาพแวดล้อมที่ 1180000
Run	เป็น run ที่ 1
Gen	เป็นค่าตอบรุ่นที่ 10
ldv	เป็นค่าตอบที่ 8
succ	เป็น 1 แสดงว่าค่าตอบกำหนดการเชิงพันธุกรรมนี้สามารถควบคุมให้แขนหุ่นยนต์เคลื่อนที่ไปสู่เป้าหมายได้ หากเป็น 0 แสดงว่าไม่สามารถควบคุมแขนเคลื่อนที่ไปสู่เป้าหมายได้
fit	ค่า fitness ของค่าตอบ มีค่าตั้งแต่ 0 (ดีที่สุด) ถึง -9216 (แย่ที่สุด)
timeuse	ใช้เวลาเท่าใด ในการทำงาน
initdist	ระยะห่างระหว่างปลายแขนกับเป้าหมาย เมื่อเริ่มการทำงาน
lastdist	ระยะห่างระหว่างปลายแขนกับเป้าหมาย เมื่อเริ่มสิ้นสุดการทำงาน
sumdist	ระยะห่างระหว่างปลายแขนกับเป้าหมาย โดยรวม
see	เป็น 1 เมื่อไม่มีสิ่งกีดขวาง (Obstacle) อยู่ระหว่างปลายแขนกับเป้าหมาย เป็น 0 เมื่อมีสิ่งกีดขวาง
die	เป็น 1 เมื่อแขนกระทบกับสิ่งกีดขวางจนชำรุด (ทำงานต่อไปไม่ได้) เป็น 0 เมื่อแขนหุ่นยนต์ยังใช้งานได้ หรือกระทบกับสิ่งกีดขวางเป็นจำนวนครั้งที่น้อยกว่าจำนวนที่ทำให้แขนเสียหาย
parent (1)	ค่าตอบนี้เกิดจาก (ต้นฉบับ) ค่าตอบหมายเลขที่ 23 ในรุ่นก่อนหน้า
parent (2)	เป็น -1 หมายความว่า ไม่ใช่ค่าตอบในรุ่นก่อนหน้า
evolmethod	แสดงวิธีการวิวัฒนาการของค่าตอบ reproduction ค่าตอบนี้เกิดจากการ reproduction จากค่าตอบหมายเลข 23 (parent1) สำหรับในกรณีอื่นๆ crossover ค่าตอบนี้เกิดจากการ crossover ระหว่าง parent1 กับ parent2 addition ค่าตอบนี้เกิดจากการ mutation แบบ addition จาก parent1 extension ค่าตอบนี้เกิดจากการ mutation แบบ extension จาก parent1 creation ค่าตอบนี้เกิดจากการสุ่มสร้างค่าตอบเริ่มต้น
nadf	จำนวน nadf ของค่าตอบ (ไม่ใช่ในการจินตทัศน์)
ap(1)	ตำแหน่งที่เกิดการเปลี่ยนแปลงของค่าตอบ parent1 ในการวิวัฒนาการครั้งนี้ (ค่า 0 หมายถึง ไม่มีความเปลี่ยนแปลง)
ap(2)	ตำแหน่งที่เกิดการเปลี่ยนแปลงของค่าตอบ parent2 ในการวิวัฒนาการครั้งนี้ (ค่า 0 หมายถึง ไม่มีความเปลี่ยนแปลง)

5.5 การแสดงคำตอบที่ได้จากกำหนดการเชิงพันธุกรรม ในรูปแบบโครงสร้างต้นไม้

การแสดงคำตอบในรูปแบบโครงสร้างของต้นไม้หรือ Tree Structure โดยแสดงความสัมพันธ์ระหว่างฟังก์ชันและเทอมินอลที่มีอยู่ในโครงสร้างคำตอบ และให้สีเพื่อการแบ่งแยกความแตกต่างระหว่างฟังก์ชันและเทอมินอลตามตาราง 7

ตาราง 7 สีของฟังก์ชันและเทอมินอล

ประเภทของบัพ	สี
If not	แดง
If or	น้ำเงิน
If and	เขียว
Inc?, Out?, Hit?, Dec? (Sense)	ฟ้า
w+, w-, e+, e-, s+, s- (Control)	เหลือง

ตัวอย่างแสดงดังรูปที่ 25 โครงสร้างของคำตอบกำหนดการเชิงพันธุกรรมที่ดีที่สุด ในรุ่นสุดท้าย ในสภาพแวดล้อมที่ 1180000 สำหรับความหมาย ของฟังก์ชันและเทอมินอล ในการควบคุมการเคลื่อนไหวแขนหุ่นยนต์สามารถดูรายละเอียดได้ในบทที่ 2



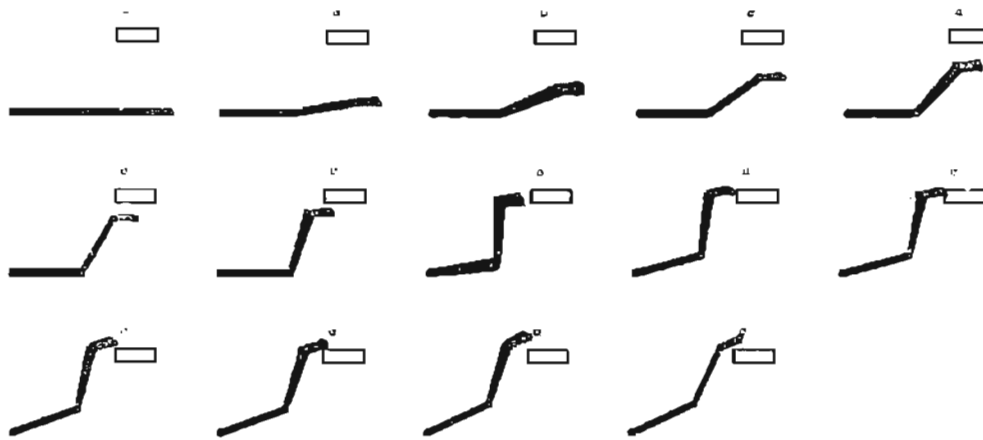
รูปที่ 25 โครงสร้างของคำตอบกำหนดการเชิงพันธุกรรมที่ดีที่สุด ในรุ่นสุดท้าย ในสภาพแวดล้อมที่ 1180000

ภาพโครงสร้างของคำตอบจะแสดงให้เห็นการเชื่อมต่อระหว่างฟังก์ชันและเทอมินอลเช่นบัพรากเป็นบัพ IF-NOT มีบัพลูกคือ control, sense เป็นเทอมินอล และมีบัพ IF-OR เป็นบัพลูกที่เป็นฟังก์ชัน เรากำหนดสีของบัพเพื่อความสะดวกในการดูว่าเป็นบัพประเภทใด เช่นหากในโครงสร้างมีบัพสีเหลืองมากแสดงว่าโครงสร้างนั้นมีเทอมินอลที่ควบคุมแขนหุ่นยนต์ (control) มาก หากในโครงสร้างมีบัพสีฟ้าแสดงว่าโครงสร้างนั้นมีเทอมินอลที่เป็นเงื่อนไขมาก (sense) ในรูปที่ 25 นี้มีบัพทั้งสองสีเป็นจำนวนใกล้เคียงกันแสดงว่ามีเทอมินอลที่ควบคุมแขนหุ่นยนต์มีจำนวนพอๆ กับเทอมินอลที่เป็นเงื่อนไข

ผู้ใช้งานสามารถเลือกดูโครงสร้างคำตอบกำหนดการเชิงพันธุกรรม คำตอบใดก็ได้ ด้วยวิธีการเลือกคำตอบกำหนดการเชิงพันธุกรรมในหัวข้อ 5.4 และกดปุ่ม “v” โปรแกรมจะสลับการจินตทัศน์มาแสดงผลโครงสร้างคำตอบ ซึ่งผู้ใช้สามารถกด “+” เพื่อขยายภาพ หรือกด “-” เพื่อลดขนาดของภาพ และใช้ปุ่มลูกศรเพื่อเลื่อนภาพตามต้องการ หรือกด “esc” เพื่อกลับไปเลือกคำตอบอื่นๆ ในหัวข้อ 5.5

5.6 การจำลองการเคลื่อนไหวของแขนหุ่นยนต์จากคำตอบที่ได้จากกำหนดการเชิงพันธุกรรม

ระบบจะอำนวยความสะดวกให้กับผู้ใช้โดยออกแบบให้มีส่วนของการจำลอง (Simulation) การเคลื่อนไหวของแขนหุ่นยนต์ อยู่ในโปรแกรม เมื่อผู้ใช้เลือกคำตอบที่ต้องการ และกดปุ่ม “o” (ตัวอักษรโอเล็ก) ระบบจะแสดงการเคลื่อนไหวของแขนหุ่นยนต์ตามสภาพแวดล้อมที่ได้กำหนดไว้ โดยนำคำตอบที่เลือกมาเป็นส่วนที่ควบคุมการเคลื่อนไหวของแขนหุ่นยนต์ ผู้ใช้งานสามารถสั่งให้ระบบเก็บภาพเป็นภาพต่อเนื่องบนฮาร์ดดิสก์ด้วยปุ่ม “O” (ตัวอักษรโอใหญ่) แทนการกดปุ่ม “o” ระบบจะจัดเก็บในรูปแบบของกราฟิกไฟล์ Compressed TIFF 5.0 (Tagged Image File Format) ทุกๆ การเคลื่อนไหวของแขนหุ่นยนต์ ซึ่งไฟล์ที่ได้จะมีส่วนขยายเป็น .TIF ผู้ใช้งานอาจนำไฟล์ TIFF เหล่านี้มาประยุกต์ใช้กับโปรแกรมอื่นๆ เช่น Adobe Premiere, Xing Mpeg Encoder, Quicktime ฯลฯ เพื่อสร้างเป็นไฟล์ภาพยนตร์ (Movie File) สำหรับใช้เป็นตัวอย่างในการศึกษาการเคลื่อนไหวของแขนหุ่นยนต์ในการควบคุมโดยคำตอบที่ได้จากกำหนดการเชิงพันธุกรรมได้อีกวิธีหนึ่ง

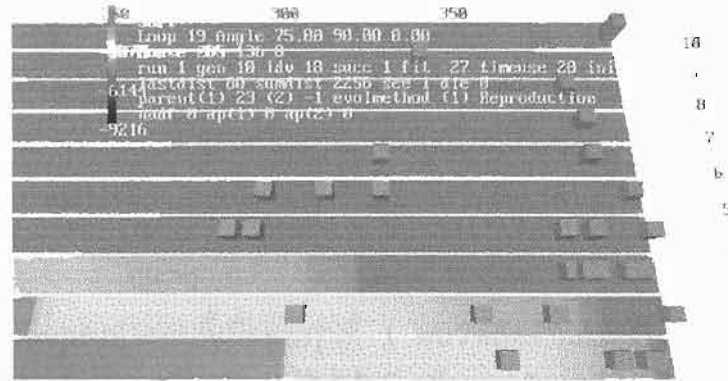


รูปที่ 26 แสดงการเคลื่อนไหวของแขนหุ่นยนต์ (เรียงจากซ้ายไปขวา, บนลงล่าง) ควบคุมจากคำตอบ กำหนดการชิงพันธุกรรมที่ดีที่สุดในวันสุดท้าย ในสภาพแวดล้อมที่ 1180000

จากภาพการเคลื่อนไหวของแขนหุ่นยนต์จะเห็นว่าแขนหุ่นยนต์จะหมุนแขนก่อน ดันและงอแขนท่อนกลางพร้อมกับงอแขนท่อนปลาย เมื่อระยะปลายแขนกับเป้าหมายไม่มี สิ่งกีดขวางแล้ว แขนหุ่นยนต์จะเริ่มยืดออกเพื่อเคลื่อนที่ไปยังเป้าหมาย การเคลื่อนไหวนี้ได้ แสดงให้เห็นถึงพฤติกรรม (Behavior) ของแขนหุ่นยนต์ซึ่งควบคุมจากคำตอบที่ได้จาก กำหนดการชิงพันธุกรรมผ่านการการจำลอง ซึ่งเราไม่สามารถบอกได้ว่าคำตอบจะมีพฤติกรรมเป็นอย่างไรหากดูโครงสร้างคำตอบเพียงอย่างเดียว

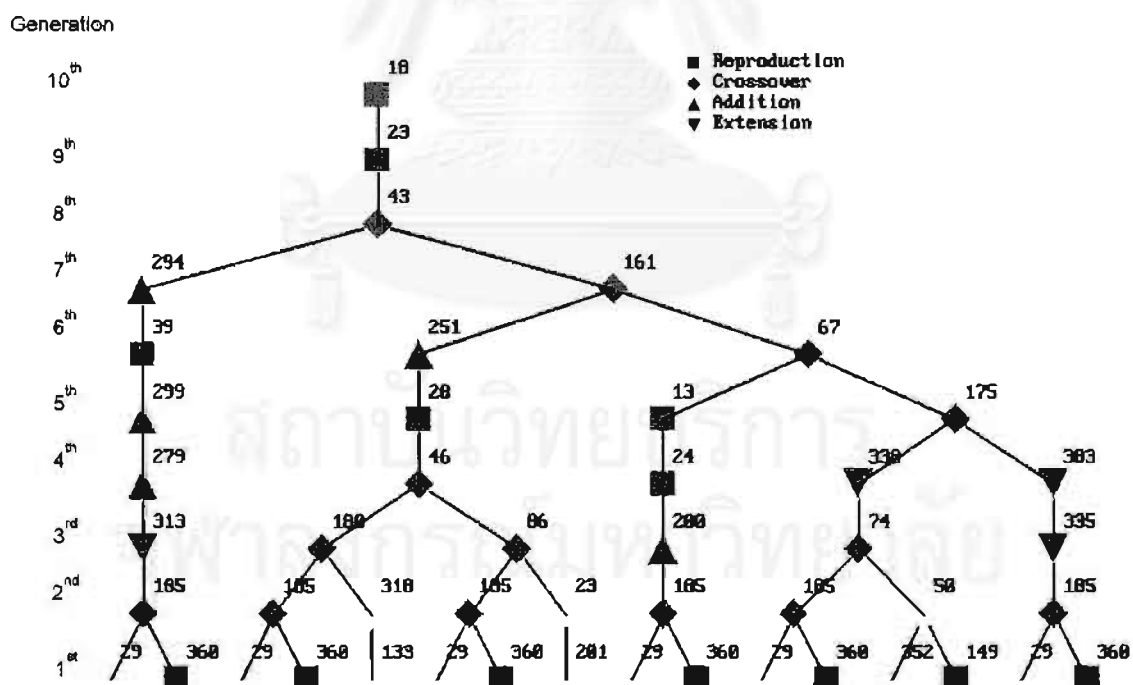
5.7 การแสดงและเลือกคำตอบจากความสัมพันธ์ระหว่างคำตอบที่เกิดจากการ วิวัฒนาการในกำหนดการชิงพันธุกรรม

คำตอบที่ผู้ใช้งานสนใจจะถูกเลือกด้วยขั้นตอนในหัวข้อ 5.4 จากนั้นเมื่อผู้ใช้งานกด ปุ่ม “a” โปรแกรมจะทำการคำนวณเพื่อหาความสัมพันธ์ของคำตอบ ว่าคำตอบที่ผู้ใช้งาน กำลังเลือกอยู่นั้นเกิดมาจากคำตอบใดบ้าง และแสดงผลเป็นกล่องสี่เหลี่ยมเพิ่มเติมขึ้นมา ในตำแหน่งแถบสีของคำตอบแต่ละรุ่น ดังรูปที่ 27 แสดงว่าคำตอบที่ถูกเลือกมากจาก คำตอบที่ดีเป็นส่วนใหญ่



รูปที่ 27 ความสัมพันธ์ในระหว่างการพัฒนาการของคำตอบที่ดีที่สุด

หลังจากที่ผู้ใช้ได้กดปุ่ม “a” เพื่อคำนวณความสัมพันธ์แล้ว และผู้ใช้งานกดปุ่ม “r” หรือเลือกเมนู Hierarchical Model ระบบจะแสดงความสัมพันธ์ในระหว่างการพัฒนาการว่าคำตอบนี้เกิดจากคำตอบใดบ้างและผ่านฟังก์ชันวิวัฒนาการอย่างไร ผู้ใช้สามารถเลือกดูคำตอบต่างๆ เพื่อเป็นแนวทางในการเปรียบเทียบระหว่างคำตอบเริ่มต้นกับคำตอบสุดท้ายได้ หรืออาจเลือกเส้นทางการวิวัฒนาการของคำตอบโดยให้ ระบบแสดงการพัฒนาการคำตอบในแต่ละขั้นตอนตั้งแต่คำตอบเริ่มแรกจนถึงคำตอบสุดท้าย ดังรูปที่ 28



รูปที่ 28 ความสัมพันธ์ในระหว่างการพัฒนาการของคำตอบที่ดีที่สุด (หมายเลข 18) ในรุ่นสุดท้าย ในสภาพแวดล้อมที่ 1180000

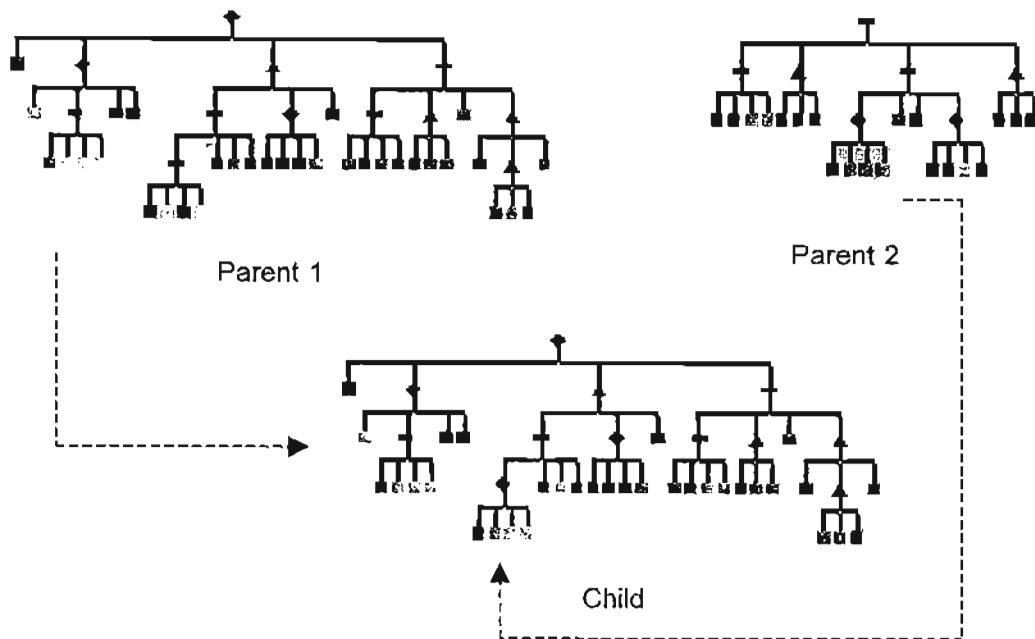
รูปที่ 28 แสดงให้เห็นว่าคำตอบหมายเลข 18 ในรุ่นที่ 10 เกิดจากการ reproduction คำตอบหมายเลข 23 รุ่นที่ 9 ซึ่งเกิดจากการ reproduction เช่นเดียวกันจากคำตอบหมายเลข 43 รุ่นที่ 8 และคำตอบนี้เกิดมาจากการ crossover ของคำตอบในรุ่นที่ 7 สองคำตอบคือคำตอบหมายเลข 94 และคำตอบหมายเลข 161 เป็นต้น

ต้นกำเนิดของคำตอบนี้เมื่อดูในรุ่นที่ 1 พบว่าเกิดจากคำตอบเริ่มต้นที่มีค่า fitness ไม่ค่อยดีจนถึงค่า fitness ที่ไม่ดีผสมกัน เป็นสีเหลืองและสีแดงที่มีค่าอยู่ระหว่าง -1500 ถึง ประมาณ -6600 และเริ่มต้นดีขึ้นจากการนำคำตอบมา crossover กันในรุ่นที่ 2

ในส่วนแสดงความสัมพันธ์ในระหว่างการพัฒนาการนี้ ผู้ใช้งานสามารถเลือกคำตอบใดๆ ที่ปรากฏในหน้าจอนี้ได้โดยการกดปุ่ม “ < ” หรือ ปุ่ม “ > ” เพื่อเลือกคำตอบในรุ่นที่ 1 โปรแกรมจะแสดงเส้นทางการพัฒนาการจากคำตอบที่กำลังเลือกซึ่งเป็นคำตอบเริ่มต้นหรือเรียกได้ว่าเป็นคำตอบต้นกำเนิด ไปยังคำตอบสุดท้ายที่อยู่ด้านบนสุด และกด “PageUp” เพื่อเลื่อนขึ้นไปเลือกคำตอบยังรุ่นถัดไปคือรุ่นที่ 2 ตามเส้นทางที่โปรแกรมได้แสดงไว้ หรือกด “PageDown” เพื่อเลื่อนลงมาเลือกรุ่นที่น้อยกว่า จากจุดนี้ผู้ใช้งานสามารถสลับการทำงานเพื่อเลือกดูโครงสร้างของคำตอบตั้งหัวข้อ 4.4 หรือดูการเคลื่อนไหวของแขนหุ่นยนต์ตามหัวข้อที่ 4.5 ได้เช่นเดียวกัน โดยโปรแกรมจะจดจำตำแหน่งต่างๆ เอาไว้ เมื่อผู้ใช้งาน กด “esc” เพื่อยกเลิกการดูโครงสร้างคำตอบหรือการเคลื่อนไหวของแขนหุ่นยนต์ โปรแกรมจะกลับมาจินตทัศน์ความสัมพันธ์ในระหว่างการพัฒนาการดังเดิม

5.8 การแสดงกระบวนการวิวัฒนาการของคำตอบในกำหนดการเชิงพันธุกรรม

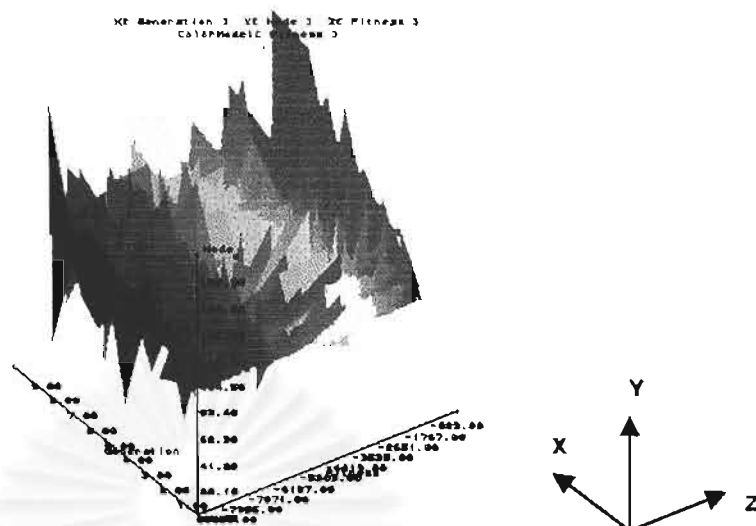
ระบบจินตทัศน์ข้อมูลมีความสามารถในการแสดงกระบวนการวิวัฒนาการเป็นลำดับขั้นตอนอย่างต่อเนื่องได้ ซึ่งผู้ใช้จะเห็นความเปลี่ยนแปลงอย่างต่อเนื่องในโครงสร้างคำตอบ จากคำตอบเริ่มต้น ไปยังคำตอบสุดท้ายที่เป็นผลลัพธ์ เมื่อผู้ใช้แสดงความสัมพันธ์ในระหว่างการพัฒนาการ และเลือกคำตอบโดยกด “ < ” หรือ “ > ” เมื่อโปรแกรมแสดงเส้นทางการพัฒนาการ เมื่อผู้ใช้กดปุ่ม “v” โปรแกรมจะแสดงโครงสร้างของคำตอบตั้งแต่เริ่มต้นจนถึงโครงสร้างของคำตอบสุดท้าย และสามารถสั่งให้ระบบจินตทัศน์เก็บเป็นภาพต่อเนื่องได้โดยใช้ปุ่ม “v” แทน รูปที่ 29 เป็นตัวอย่างจากการวิวัฒนาการ ซึ่งพัฒนาคำตอบด้วยวิธีการไขว้เปลี่ยน (crossover) จากโครงสร้างต้นไม้เริ่มต้น parent1 และ parent2 ไปยังต้นไม้ที่เป็นผลลัพธ์ child



รูปที่ 29 กระบวนการวิวัฒนาการโครงสร้างคำตอบในกำหนดการเชิงพันธุกรรม

5.9 การแสดงข้อมูลกำหนดการเชิงพันธุกรรมในรูปแบบ 3 มิติ

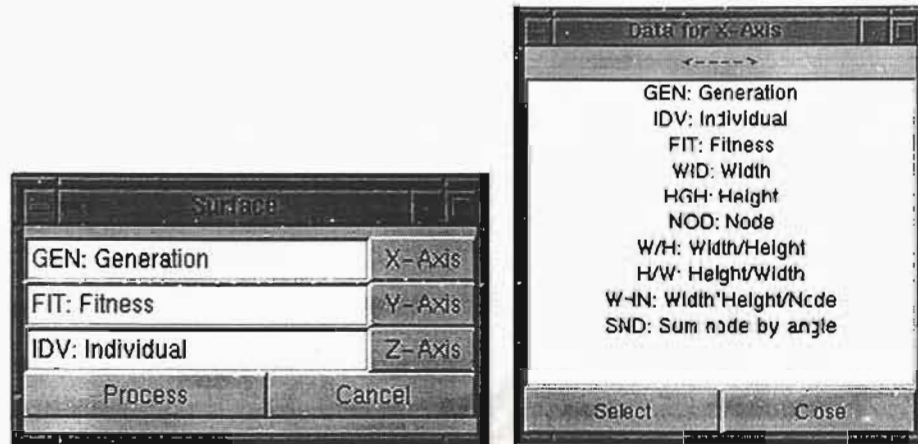
ผู้ใช้สามารถเลือกดูความสัมพันธ์ในรูปแบบต่างๆ ของข้อมูลที่สนใจโดยระบบได้ออกแบบให้ผู้ใช้สามารถเลือกข้อมูลได้อย่างอิสระ เพื่อมาสร้างแผนภาพความสัมพันธ์ระหว่างข้อมูลในรูปแบบ 3 มิติ โดยระบบจะเลือกข้อมูลตามเงื่อนไข (Filter) ตามหัวข้อ 5.10.3 ที่ได้กำหนดไว้มาแสดงผล จากรูปที่ 30 เป็นการแสดงความสัมพันธ์ระหว่างรุ่น (Generation) จำนวนบัพ (Node) และค่าความเหมาะสม (Fitness Value)



รูปที่ 30 กราฟ 3 มิติแสดงความสัมพันธ์ของข้อมูลค่าตอบจากกำหนดการเชิงพันธุกรรมระหว่าง
Generation - Node - Fitness

แกน X แทนข้อมูลรุ่น (Generation) ในกำหนดการเชิงพันธุกรรม ซึ่งมีค่าอยู่ระหว่างรุ่นที่ 1 ถึง รุ่นที่ 10 แกน Y แทนจำนวนบัพ (Node) ซึ่งถ้าดูจากภาพที่ขยายขนาดบนจอภาพแล้วจะมีค่ามากที่สุดประมาณ 200 บัพ และแกน Z แทนค่าความเหมาะสม (Fitness) ซึ่งมีค่าอยู่ระหว่าง 0 (ดีที่สุด) ถึง -9216 (ไม่ดี)

จากภาพที่ได้จะเห็นว่ารุ่นที่ 1 ซึ่งอยู่ด้านหน้าของภาพจะมีความสูงเท่ากันและค่อยๆ เปลี่ยนแปลงไปในรุ่นต่อไปซึ่งอยู่ด้านหลัง เพราะว่าเริ่มต้นค่าตอบได้ถูกกำหนดให้สร้างที่จำนวน 80 บัพ ต่อมาเกิดการเปลี่ยนแปลงแสดงว่าจำนวนบัพภายในค่าตอบเพิ่มขึ้นและลดลงในระหว่างการพัฒนาการ ในรุ่นหลังสุดจะเห็นว่าจำนวนบัพอยู่ในช่วงสีแดง สีเหลือง และสีเขียวเพิ่มขึ้นมาอย่างชัดเจน แสดงว่าค่าตอบที่มีค่าความเหมาะสมดีมีจำนวนบัพมากกว่าจำนวนบัพเริ่มต้น แต่ค่าตอบที่มีจำนวนบัพมากอาจมีค่าความเหมาะสมปานกลาง (สีเหลือง) หรือน้อย (สีแดง) ได้



รูปที่ 31 (a) กล่องโต้ตอบข้อมูลกำหนดการเชิงพันธุกรรมในรูปแบบ 3 มิติ (ซ้าย) และ (b) ส่วนกำหนดประเภทของข้อมูลที่จะนำมาแสดงผลในแต่ละแกน (ขวา)

เมื่อผู้ใช้งานต้องการให้ระบบทำการจินตทัศน์ โดยเลือกคำสั่ง "surface" จากเมนู โปรแกรมจะแสดงกล่องโต้ตอบ ดังรูปที่ 31(a) เป็นการเลือกข้อมูลเพื่อแสดงผลในแกนต่างๆ ผู้ใช้งานสามารถเปลี่ยนแปลงได้โดยกด ปุ่มที่อยู่ด้านหลัง เช่นปุ่ม "X-Axis" โปรแกรมจะแสดงกล่องโต้ตอบ รูปที่ 31 (b)

ข้อมูลที่สามารถนำมาแสดงผล มีดังนี้

Generation	รุ่นของคำตอบ
Individual	หมายเลขของคำตอบในแต่ละรุ่น
Fitness	ค่าความเหมาะสมของคำตอบ
Width	ความกว้างของคำตอบเมื่อจัดอยู่ในรูปโครงสร้างต้นไม้
Height	ความสูงของคำตอบเมื่อจัดอยู่ในรูปโครงสร้างต้นไม้
Node	จำนวนบัพของคำตอบเมื่อจัดอยู่ในรูปโครงสร้างต้นไม้
Width/Height	อัตราส่วนของความกว้างและความสูงของคำตอบเมื่อจัดอยู่ในรูปโครงสร้างต้นไม้
Height/Width	อัตราส่วนของความสูงและความกว้างของคำตอบเมื่อจัดอยู่ในรูปโครงสร้างต้นไม้
Width*Height/Node	อัตราส่วนความหนาแน่นของคำตอบเมื่อจัดอยู่ในรูปโครงสร้างต้นไม้

5.10 การตั้งค่าสำหรับกรองข้อมูลสำหรับการแสดงข้อมูลกำหนดการเชิงพันธุกรรมในรูปแบบ 3 มิติ

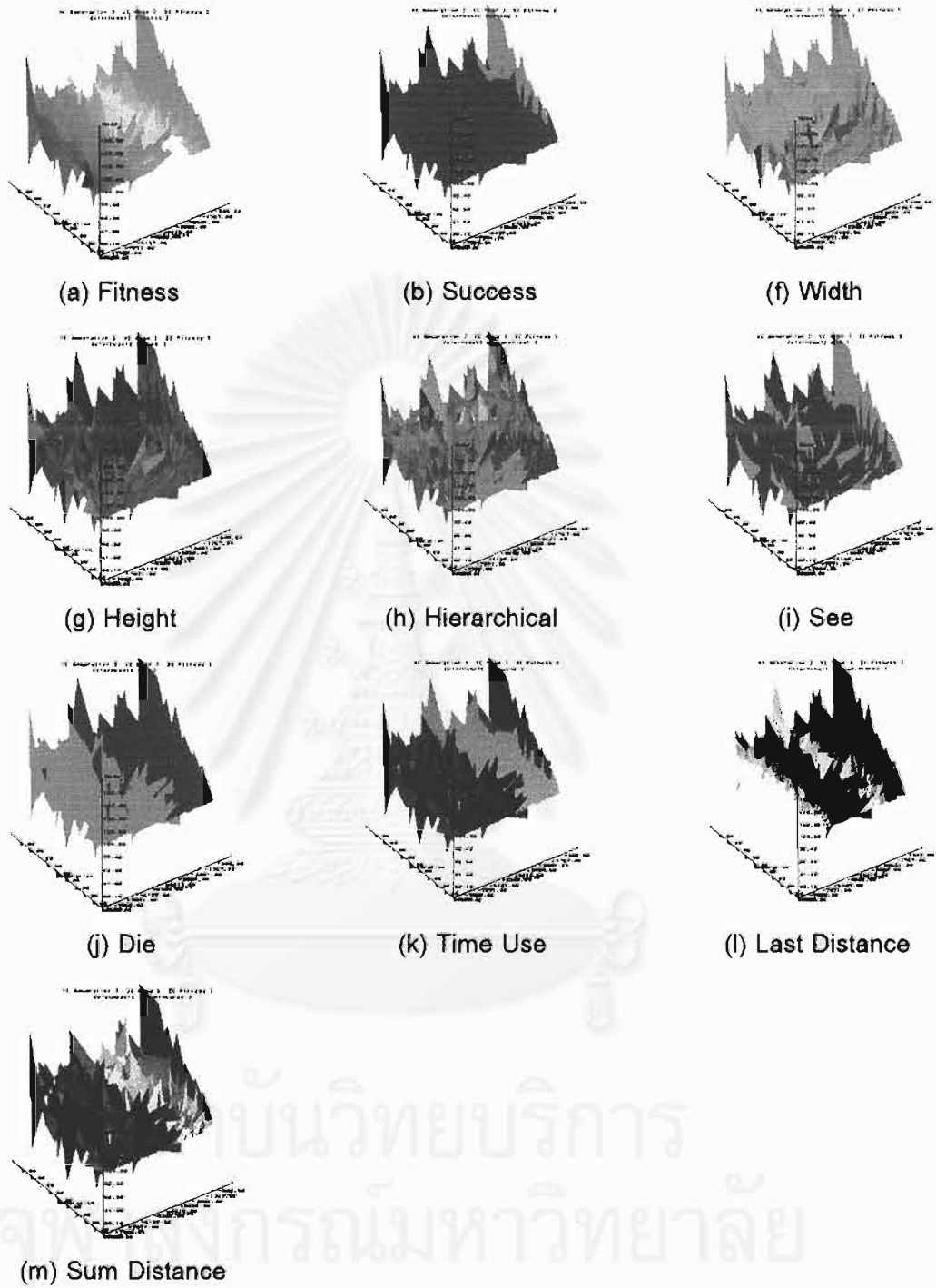
เป็นการตั้งค่าต่างๆ ไว้สำหรับการแสดงข้อมูลกำหนดการเชิงพันธุกรรมในรูปแบบ 3 มิติในหัวข้อ 5.9 การแสดงข้อมูลกำหนดการเชิงพันธุกรรมในรูปแบบ 3 มิติ ซึ่งจะมีผลเฉพาะการจินตทัศน์ในหัวข้อที่ 5.9 เท่านั้น ผู้ใช้งานควรจะต้องตั้งค่าในส่วนนี้ก่อนจะเรียกการทำงานในหัวข้อ 5.9 การตั้งค่าประกอบไป 4 ส่วนย่อย คือ ค่าเกี่ยวกับสี ค่า filter ค่าความ smooth หรือค่าเฉลี่ยของข้อมูล ค่ากำหนดรูปแบบการแสดงผล

5.10.1 การตั้งค่าสี

แสดงให้เห็นถึงตัวเลือกสีที่ผู้ใช้สามารถเลือกในการแสดงผลได้โดยใช้สีตามตาราง 8 ประกติเมื่อเริ่มต้นโปรแกรมจะแสดงค่าสีของคำตอบแบบ fitness เป็นการแสดงสีแบบเดียวกับรูปที่ 32 (a)

ตาราง 8 แสดงคำสีและความหมาย

คำสี	สี	ความหมาย
(a) Fitness	สี ดำ - แดง - ส้ม - เหลือง - เขียว	ค่าความเหมาะสมตั้งแต่ -9000 ถึง 0 ตามลำดับ
(b) Success	สีเขียว	สำเร็จ (1)
	สีน้ำเงิน	ไม่สำเร็จ (0)
(c) Node	สีน้ำตาลเข้ม - น้ำตาล	จำนวน node จากน้อยไปมาก
(d) Generation	สีชมพู, ม่วง, น้ำเงิน, น้ำเงินอมฟ้า, ฟ้า, เขียวอ่อน, เขียว, เหลือง, ส้ม	แทนรุ่นที่ 1 ถึง 10 ตามลำดับ
(e) Individual	สีดำ - สีเหลืองอ่อน	หมายเลขคำตอบที่ 0 ถึง 399 ตามลำดับ
(f) Width	สีชมพูเข้ม - สีชมพูอ่อน	แทนความกว้างของคำตอบเมื่อแสดงในรูปโครงสร้างสร้างต้นไม้จากน้อยไปมาก
(g) Height	สีฟ้าเข้ม - สีฟ้าอ่อน	แทนความสูงของคำตอบเมื่อแสดงในรูปโครงสร้างต้นไม้จากน้อยไปมาก
(h) Hierarchical	เริ่มจาก แดง, เขียว, น้ำเงิน	แสดงการเคลื่อนที่ของสีจากต้นกำเนิดโดยแบ่งต้นกำเนิดเป็นสามส่วนด้วยสี แดง, เขียว, น้ำเงิน สำหรับรายละเอียดได้แสดงไว้ในหัวข้อถัดไป
(i) See	สีเขียว	ไม่มีสิ่งกีดขวางระหว่างปลายแขนกับเป้าหมาย (1)
	สีน้ำเงิน	มีสิ่งกีดขวางระหว่างปลายแขนกับเป้าหมาย (0)
(j) Die	สีเขียว	แขนหุ่นยนต์เสีย (1)
	สีน้ำเงิน	แขนหุ่นยนต์ไม่เสีย (0)
(k) Time Use	สีดำ - สีขาว	แทนปริมาณการใช้เวลาในการขยับแขนหุ่นยนต์จากจุดเริ่มต้นไปยังเป้าหมายจากน้อยไปมาก
(l) Last Distance	สีดำ - สีขาว	แทนปริมาณระยะห่างระหว่างปลายแขนและเป้าหมายจากน้อยไปมาก
(m) Sum Distance	สีดำ - สีขาว	แทนปริมาณระยะห่างระหว่างปลายแขนและเป้าหมายรวมทั้งตั้งแต่เริ่มต้นจนถึงสิ้นสุดการทำงานจากน้อยไปมาก
(n) Evolution Method	สีขาว	คำตอบที่เกิดจากการสุ่มสร้าง
	สีเหลือง	คำตอบที่เกิดจากการ reproduction
	สีน้ำเงิน	คำตอบที่เกิดจากการ crossover
	สีชมพู	คำตอบที่เกิดจากการ addition
	สีฟ้า	คำตอบที่เกิดจากการ extension



รูปที่ 32 การตั้งค่าสีในรูปแบบต่างๆ

5.10.2 การกำหนดค่าสีแบบ Hierarchical

เป็นแนวคิดในการแสดงการเคลื่อนที่ของสีจากคำตอบที่เป็นต้นกำเนิดไปยังคำตอบรุ่นต่อไป โดยแบ่งต้นกำเนิดสีเป็นสามส่วนตามค่าความเหมาะสม ด้วยสี แดง, เขียว, น้ำเงิน ซึ่งมีความหมายดังนี้ คือ สีแดงแทนช่วงคำตอบที่มีค่าความเหมาะสมน้อย สีเขียวแทนช่วงคำตอบที่มีค่าความเหมาะสมปานกลาง และสีน้ำเงินแทนช่วงคำตอบที่มีค่าความเหมาะสมมาก

หากคำตอบในรุ่นต่อไปวิวัฒนาการด้วยการ **reproduction** จากคำตอบรุ่นก่อนจะมีสีเหมือนกับคำตอบรุ่นก่อน หากคำตอบในรุ่นต่อไปวิวัฒนาการด้วยการ **crossover** จะนำสีมาผสมกันเพื่อเป็นสีในคำตอบรุ่นต่อไป ดังนี้

$$\begin{aligned} \text{RED}_n &= (\text{RED}_{p1} + \text{RED}_{p2}) / 2 \\ \text{GREEN}_n &= (\text{GREEN}_{p1} + \text{GREEN}_{p2}) / 2 \\ \text{BLUE}_n &= (\text{BLUE}_{p1} + \text{BLUE}_{p2}) / 2 \\ (n = \text{new offspring}, \quad p1 = \text{parent1}, \quad p2 = \text{parent2}) \end{aligned}$$

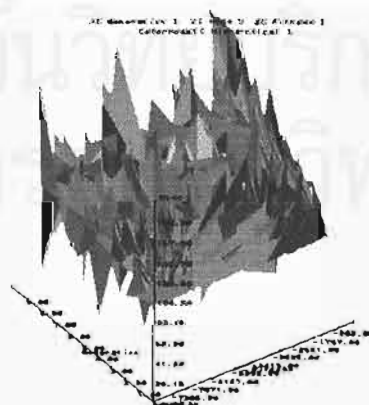
หากคำตอบในรุ่นต่อไปวิวัฒนาการด้วยการ **addition** จะผสมสีเดียวกับสีเดิมเพื่อเป็นสีในคำตอบรุ่นต่อไป ดังนี้

$$\begin{aligned} \text{RED}_n &= (\text{RED}_p + 0) / 2 \\ \text{GREEN}_n &= (\text{GREEN}_p + 0) / 2 \\ \text{BLUE}_n &= (\text{BLUE}_p + 0) / 2 \end{aligned}$$

หากคำตอบในรุ่นต่อไปวิวัฒนาการด้วยการ **extension** จะผสมสีขาวกับสีเดิมเพื่อเป็นสีในคำตอบรุ่นต่อไป ดังนี้

$$\begin{aligned} \text{RED}_n &= (\text{RED}_p + 255) / 2 \\ \text{GREEN}_n &= (\text{GREEN}_p + 255) / 2 \\ \text{BLUE}_n &= (\text{BLUE}_p + 255) / 2 \end{aligned}$$

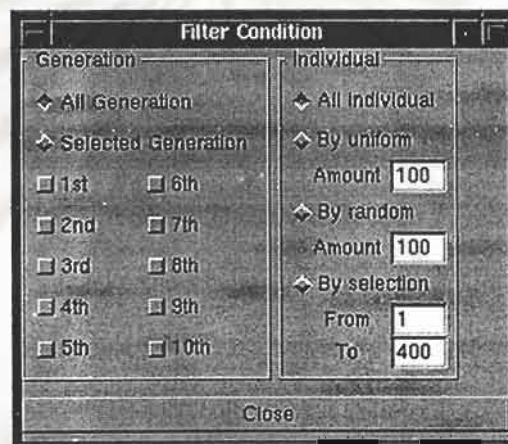
ผลที่ได้แสดงให้เห็นว่าคำตอบที่เกิดจากการวิวัฒนาการในรุ่นหลังนั้นมีการผสมรวมกันจากคำตอบรุ่นแรกไปยังรุ่นท้ายสุด โดยมีส่วนผสมของสีน้ำเงินมากแสดงว่า คำตอบส่วนใหญ่มาจากต้นกำเนิดที่มีค่าความเหมาะสมมาก



รูปที่ 33 ผลจากการกำหนดค่าสีแบบ Hierarchical

5.10.3 การตั้งค่าสำหรับการกรองข้อมูล

เป็นการบอกให้ระบบจินตทัศน์ข้อมูลทราบว่าควรจะนำข้อมูลมาใช้ในการจินตทัศน์อย่างไร หรือเฉพาะส่วนใด ผู้ใช้งานสามารถเลือกให้ระบบจินตทัศน์นำข้อมูลเฉพาะรุ่นที่กำหนดมาจินตทัศน์ก็ได้ โดยกดเลือกที่ Generation จาก "All Generation" เป็น "Selected Generation" และเลือกรุ่นที่ต้องการ หรือผู้ใช้งานไม่ต้องการให้ระบบจินตทัศน์นำข้อมูลคำตอบทั้ง 400 ในรุ่นนั้นมาใช้ ก็สามารถกำหนดให้ใช้ข้อมูลบางส่วนได้ โดยกดเลือกที่ Individual เปลี่ยนจาก "All individual" เป็น "By uniform" สำหรับข้อมูล 100 ใน "Amount" ตำแหน่งข้อมูลที่มีระยะห่างเท่าๆ กัน หรือเปลี่ยนเป็น "By random" โปรแกรมจะสุ่มคำตอบต่างๆ ในรุ่นมาเป็นจำนวน 100 ตัวซึ่งอาจมีการสุ่มซ้ำเกิดขึ้นได้ ส่วน "By selection" เป็นการกำหนดช่วงของหมายเลขคำตอบ



รูปที่ 34 กำหนดค่าสำหรับการกรองข้อมูล



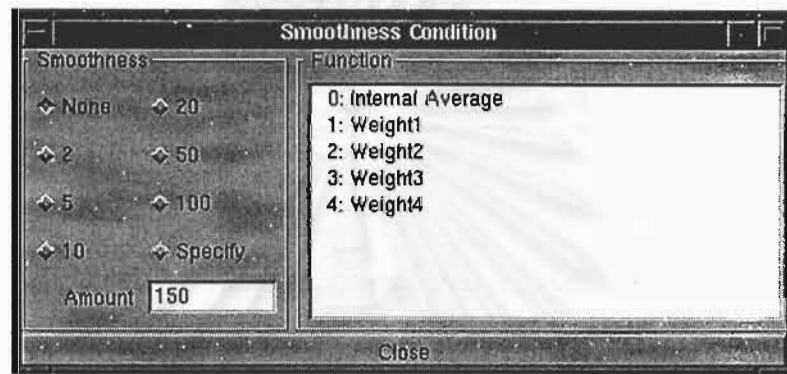
(a) 6th to 10th Generation Filtered

(b) 1st to 40th Individual Filtered

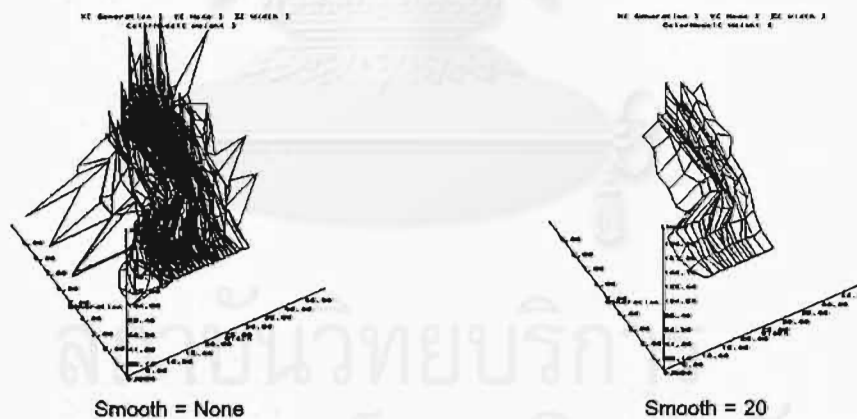
รูปที่ 35 ผลลัพธ์ที่ได้จากการจินตทัศน์เมื่อกำหนดค่าสำหรับการกรองข้อมูลที่แตกต่างกัน

5.10.4 ค่าเฉลี่ยของข้อมูล

หากผู้ใช้งานเลือก "Smoothness" เป็นค่าใดค่าหนึ่งที่ไม่ใช่ "None" และผู้ใช้งานเลือกฟังก์ชัน "Internal Average" โปรแกรมจะนำข้อมูลใกล้เคียงมาทำการหาค่าเฉลี่ยเป็นจำนวนตามที่คุณกำหนด ผู้ใช้สามารถเพิ่มเติมฟังก์ชันที่ต้องการได้เอง โดยการเขียนโปรแกรมเพิ่มเติมซึ่งสามารถอ่านรายละเอียดได้จากภาคผนวก



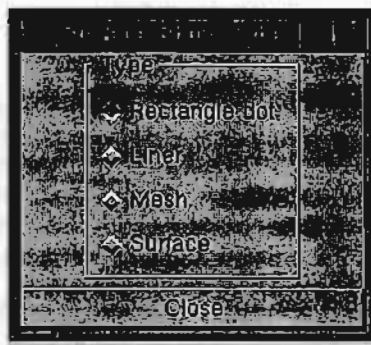
รูปที่ 36 กำหนดค่าเฉลี่ยข้อมูล



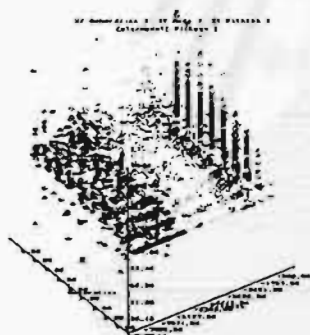
รูปที่ 37 ผลลัพธ์ที่ได้จากการจินตทัศน์เมื่อกำหนดค่าเฉลี่ยข้อมูลแตกต่างกัน

5.10.5 คำกำหนดรูปแบบการแสดงผล

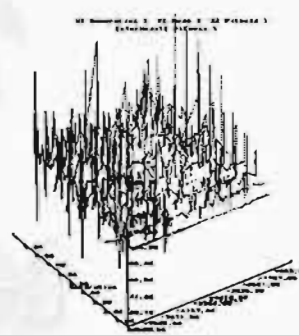
เพื่อความรวดเร็วในการแสดงผลระบบการจินตทัศน์ และการใช้งานในการจินตทัศน์ที่อาจต้องการรูปแบบที่แตกต่างกัน ผู้ใช้งานสามารถเลือกให้ระบบจินตทัศน์ทำการแสดงข้อมูลด้วยรูปแบบทั้งหมด 4 รูปแบบ ตามรูปที่ 38



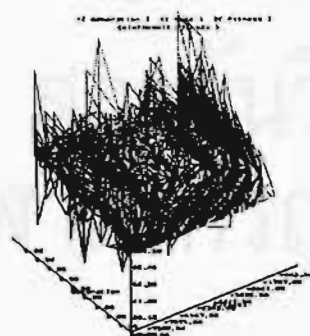
รูปที่ 38 กำหนดรูปแบบการแสดงผล



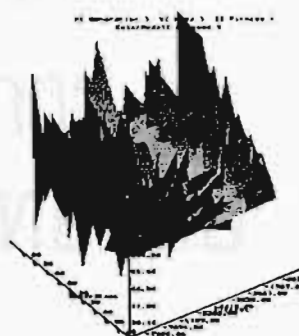
(a) แสดงแบบจุด (Rectangle Dot)



(b) แสดงแบบเส้น (Liner)



แสดงแบบโครงตาข่าย (Mesh)



(d) แสดงแบบพื้นผิว (Surface)

รูปที่ 39 ผลลัพธ์ที่ได้จากการจินตทัศน์เมื่อกำหนดรูปแบบการแสดงผลที่แตกต่างกัน

5.11 การปรับเปลี่ยนสถานะการแสดงผล และ การเก็บภาพที่ปรากฏในระบบ GPVIS เป็นไฟล์

โดยทั่วไปในการจินตทัศน์ข้อมูล ระบบจะแสดงผลด้วยสีพื้นซึ่งเป็นสีดำ หรือโทนสีมืด และแสดงตัวอักษรต่างๆ ด้วยสีขาว เมื่อผู้ใช้งานจัดเก็บเป็นไฟล์จะได้ภาพตามที่ระบบแสดงอยู่ ซึ่งจะสิ้นเปลืองหมึกในการนำภาพที่ได้ไปพิมพ์ ดังนั้นเมื่อผู้ใช้งานกดปุ่ม “F11” ระบบจะอำนวยความสะดวกในการปรับเปลี่ยนสถานะการแสดงผลจากพื้นสีดำเป็นพื้นสีขาว และเปลี่ยนสีตัวอักษรสีขาวเป็นพื้นสีดำ โดยสีต่างๆ ในส่วนของการจินตทัศน์จะไม่เปลี่ยนแปลง จากนั้นผู้ใช้งานจะกดปุ่ม “F12” เพื่อจัดเก็บไฟล์ได้ตามปรกติ

ผู้ใช้งานสามารถสั่งให้ระบบจินตทัศน์ข้อมูลจัดเก็บหน้าจอการจินตทัศน์ในขณะที่แสดงผลอยู่ ในการแสดงผลใดๆ ด้วยปุ่ม “F12” ระบบจะจัดเก็บไฟล์ภาพด้วยรูปแบบ Compress TIFF 5.0 ไฟล์ ในสารบบ vgl/snap/ โดยมีชื่อขึ้นต้น ตามด้วยหมายเลข 6 หลัก ที่นับภายในโปรแกรมเริ่มต้นจาก 0 เมื่อโปรแกรมเริ่มทำงาน ดังตาราง 9

ตาราง 9 การขึ้นต้นชื่อไฟล์สำหรับการจัดเก็บการแสดงผลการจินตทัศน์

ขึ้นต้นชื่อไฟล์	การแสดงผล	ตัวอย่าง
gni	การเลือกคำตอบที่ได้จากกำหนดการเชิงพันธุกรรม (หัวข้อ 5.4)	gni00018.tif
idv	การแสดงผลคำตอบในรูปแบบโครงสร้างของต้นไม้ (หัวข้อ 5.5)	idv000884.tif
sim	การจำลองการเคลื่อนไหวของแขนหุ่นยนต์จากคำตอบที่ได้จากกำหนดการเชิงพันธุกรรม (หัวข้อ 5.6)	sim000776.tif
tre	การแสดงผลและเลือกคำตอบจากความสัมพันธ์ระหว่างคำตอบที่เกิดจากการวิวัฒนาการ (หัวข้อ 0)	tre000581.tif
srf	การแสดงผลข้อมูลกำหนดการเชิงพันธุกรรมในรูปแบบ 3 มิติ (หัวข้อ 5.9)	srf000998.tif

ในบทนี้ได้แสดงถึงส่วนต่างๆ ของการจินตทัศน์การวิวัฒนาการข้อมูลกำหนดการเชิงพันธุกรรมตามระบบการจินตทัศน์ข้อมูลกำหนดการเชิงพันธุกรรม GPVIS สำหรับบทต่อไปจะกล่าวถึงการนำการจินตทัศน์เหล่านี้ไปใช้งาน และผลบางส่วนที่สังเกตได้จากการจินตทัศน์ ซึ่งมีผลทำให้เกิดความต้องการในการขยายระบบการจินตทัศน์เพิ่มเติมขึ้นมาในรูปแบบที่แตกต่างจากการจินตทัศน์ข้อมูลกำหนดการเชิงพันธุกรรมที่ได้ออกแบบไว้



สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 6

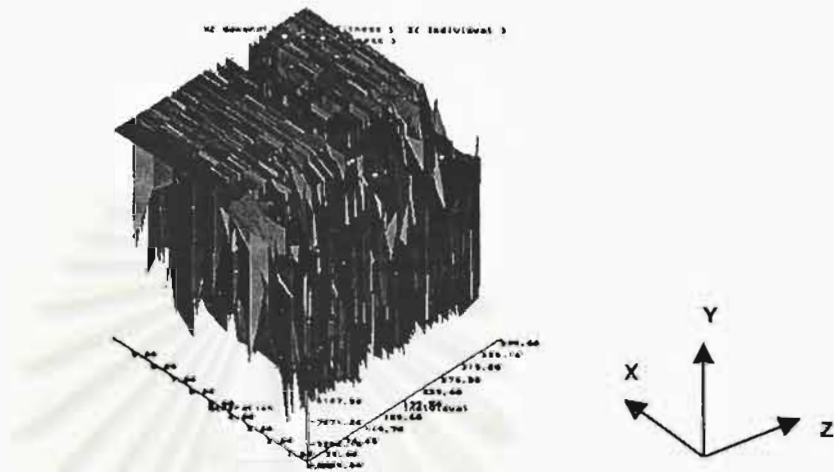
สิ่งที่น่าสนใจจากการจินตทัศน์ข้อมูลในระหว่างการพัฒนาการที่ได้มาจากกำหนดการเชิงพันธุกรรม

เนื่องจากมีการนำกำหนดการเชิงพันธุกรรมมาใช้ค้นหาคำตอบที่ต้องการอย่างแพร่หลาย และเป็นทางเลือกวิธีหนึ่งในการค้นหาคำตอบในกรณีที่ไม่มีขั้นตอนวิธี (Algorithm) หาคำตอบอื่นที่ดีพอในการนำไปใช้งาน ซึ่งเราทราบว่ากำหนดการเชิงพันธุกรรมเริ่มต้นจากการสุ่มสร้างคำตอบขึ้นมาเพียงเท่านั้น เราจึงเกิดข้อสงสัยว่ากำหนดการเชิงพันธุกรรมปรับปรุงคำตอบอย่างไร จึงทำให้คำตอบที่ได้ในรุ่นหลังมีประสิทธิภาพดีเพียงพอที่จะนำไปใช้งานได้ ผู้วิจัยจึงได้มุ่งเน้นไปยังกระบวนการวิวัฒนาการของคำตอบในกำหนดการเชิงพันธุกรรมเพื่อค้นหาบางสิ่งซึ่งช่วยอธิบายข้อสงสัยดังกล่าวด้วยเครื่องมือจินตทัศน์ที่มีอยู่

6.1 การจินตทัศน์ฟังก์ชันการวิวัฒนาการกำหนดการเชิงพันธุกรรม

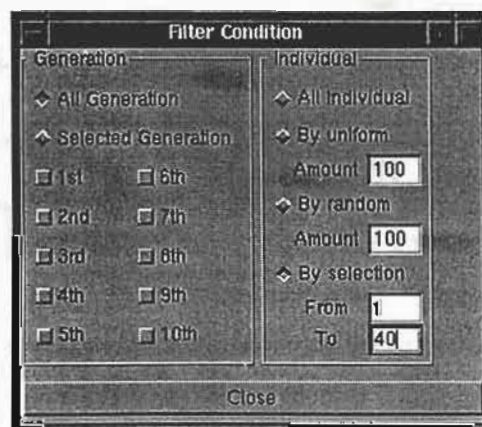
ในการนำกำหนดการเชิงพันธุกรรมไปใช้งานแก้ปัญหาต่างๆ นั้น หลายคนมีข้อสงสัยว่าควรจะปรับพารามิเตอร์ของกำหนดการเชิงพันธุกรรมอย่างไรจึงจะดีซึ่งทำให้กำหนดการเชิงพันธุกรรมพัฒนาคำตอบที่มีประสิทธิภาพหรือได้คำตอบที่รวดเร็ว ซึ่งพารามิเตอร์ของกำหนดการเชิงพันธุกรรมมีหลากหลายประการ เช่นควรใช้อัตราส่วนในการสืบพันธุ์เท่าใด อัตราส่วนในการไขว้เปลี่ยนเท่าใดจึงจะดีหรือควรจะมีมากกว่าการกลายพันธุ์หรือไม่ ทางด้าน Koza ได้กล่าวไว้ว่าการกลายพันธุ์ในกำหนดการเชิงพันธุกรรมนั้นไม่มีประโยชน์เนื่องจากกำหนดการเชิงพันธุกรรมนั้นไม่ได้ใช้ตำแหน่งของบัพในการทำงานเหมือนกับขั้นตอนเชิงพันธุกรรมอีกทั้งยังมีประเภทของบัพเป็นจำนวนน้อยซึ่งมีอยู่ในประชากรครบถ้วนไม่ได้หายไป จึงไม่จำเป็นต้องเพิ่มประเภทของบัพด้วยวิธีการกลายพันธุ์ และการไขว้เปลี่ยนในกรณีที่ดินไม้ทั้งสองต้นเกิดขึ้น ณ ตำแหน่งบัพใบนั้นก็เหมือนกับการกลายพันธุ์แบบจุด (Point Mutation) และหากการกลายพันธุ์แบบจุดเป็นสิ่งดีการไขว้เปลี่ยนก็ได้ทำหน้าที่นั้นแล้ว [Koza [1] pp 105-107] หันมาดูทางด้านงานวิจัยของชัยวัฒน์ ได้ใช้การกลายพันธุ์เป็นส่วนหนึ่งของการทดลองซึ่งใช้เป็นจำนวนมาก และยังแบ่งการกลายพันธุ์ออกเป็นสองแบบคือกลายแบบต่อยอดและกลายแบบต่อปลาย ผู้วิจัยจึงคิดว่าควรทดลองแสดง

การจินตทัศน์เพื่อแสดงให้เห็นว่าค่าความเหมาะสมของคำตอบที่ผ่านฟังก์ชันการวิวัฒนาการว่าแบบใดจะดีกว่ากัน โดยเริ่มต้นจากรูปที่ 40 ซึ่งแสดงความสัมพันธ์ของคำตอบระหว่าง รุ่นของคำตอบ (แกน X) กับค่าความเหมาะสม (แกน Y) และหมายเลขคำตอบ (แกน Z)



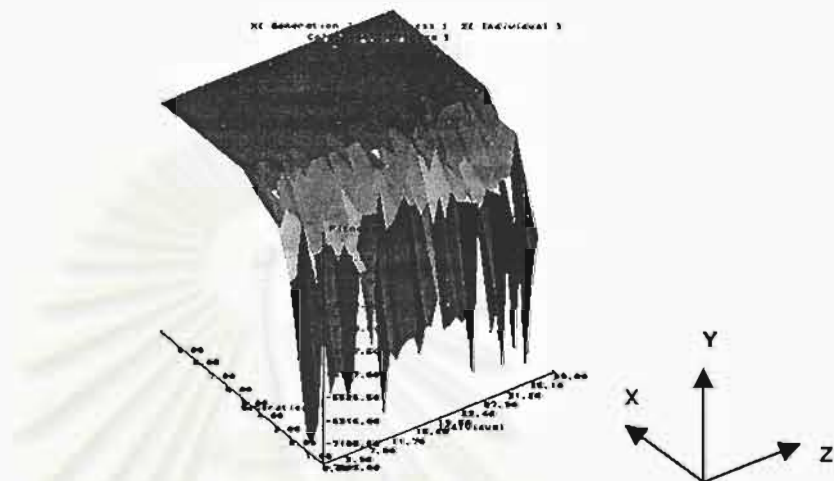
รูปที่ 40 ความสัมพันธ์ของคำตอบระหว่างรุ่นของคำตอบ, ค่าความเหมาะสม และหมายเลขคำตอบ

เราทราบว่าในรุ่นแรกคำตอบทุกตัวเกิดจากการสุ่มสร้าง แต่ในรุ่นถัดไปคำตอบทุกตัวที่มีหมายเลข 1-40 เกิดจากการสืบพันธุ์ (reproduction) หมายเลข 41-200 เกิดจากการไขว้เปลี่ยน (crossover) หมายเลข 201-300 เกิดจากการกลายพันธุ์แบบต่อยอด (addition) หมายเลข 301-400 เกิดจากการกลายพันธุ์แบบต่อปลาย (extension) เราต้องการทราบผลที่ได้จากฟังก์ชันการวิวัฒนาการ ว่าแบบใดให้ค่าความเหมาะสมเป็นอย่างไร จึงได้ใช้การกรองข้อมูลในหัวข้อ 5.10.3 เลือกคำตอบหมายเลข 1-40



รูปที่ 41 การเลือกคำตอบหมายเลข 1 ถึง 40

และให้โปรแกรมแสดงภาพความสัมพันธ์ใหม่ ซึ่งจะเป็นดังรูปที่ 42 สังเกตว่าหมายเลขคำตอบจะกระจายเต็มแกน Z และถ้ามองเห็นตัวเลขบนแกนจะมีค่าอยู่ระหว่าง 0 ถึง 40 เนื่องจากระบบ GPVIS ถูกออกแบบให้มีการปรับอัตราส่วนของแกนตามข้อมูลที่แสดงโดยอัตโนมัติ

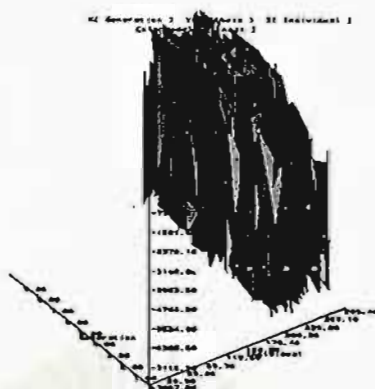


รูปที่ 42 หมายเลขคำตอบ 1 ถึง 40 บนแกนความสัมพันธ์ของคำตอบระหว่างรุ่นของคำตอบ, ค่าความเหมาะสม และหมายเลขคำตอบ

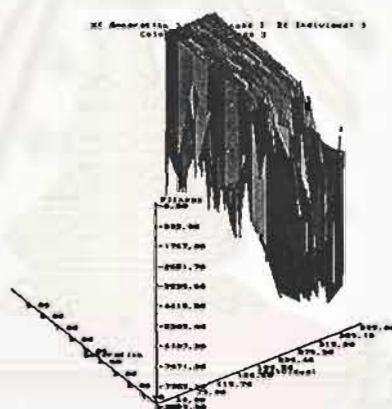
เราแสดงภาพคำตอบหมายเลข 41-200 หมายเลข 201-300 และหมายเลข 301-400 ด้วยวิธีเดียวกัน จะได้รูปที่ 43 รูปที่ 44 และรูปที่ 45 ตามลำดับ



รูปที่ 43 หมายเลขคำตอบ 41 ถึง 200 บนแกนความสัมพันธ์ของคำตอบระหว่างรุ่นของคำตอบ, ค่าความเหมาะสม และหมายเลขคำตอบ

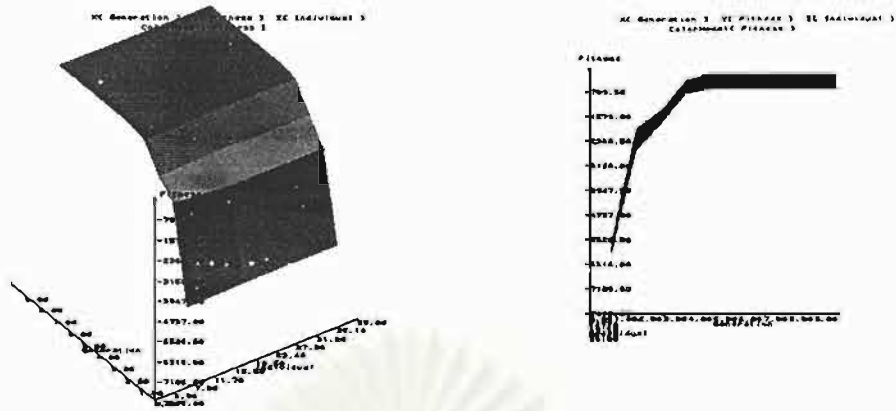


รูปที่ 44 หมายเลขคำตอบ 201 ถึง 300 บนแกนความสัมพันธ์ของคำตอบระหว่างรุ่นของคำตอบ, ค่าความเหมาะสม และหมายเลขคำตอบ

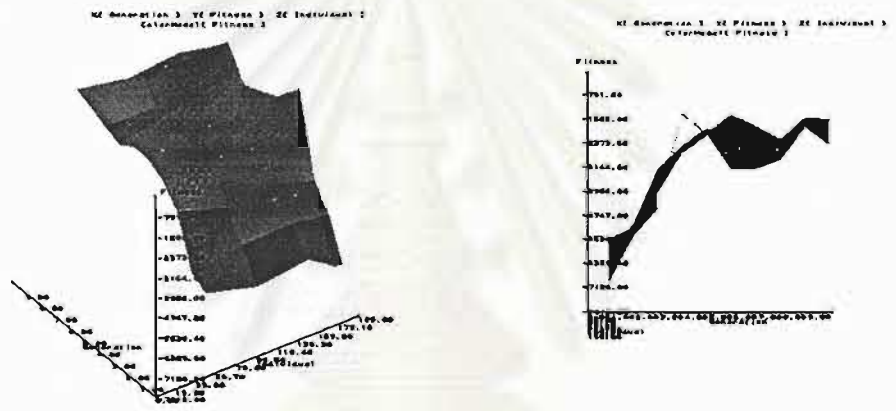


รูปที่ 45 หมายเลขคำตอบ 301 ถึง 400 บนแกนความสัมพันธ์ของคำตอบระหว่างรุ่นของคำตอบ, ค่าความเหมาะสม และหมายเลขคำตอบ

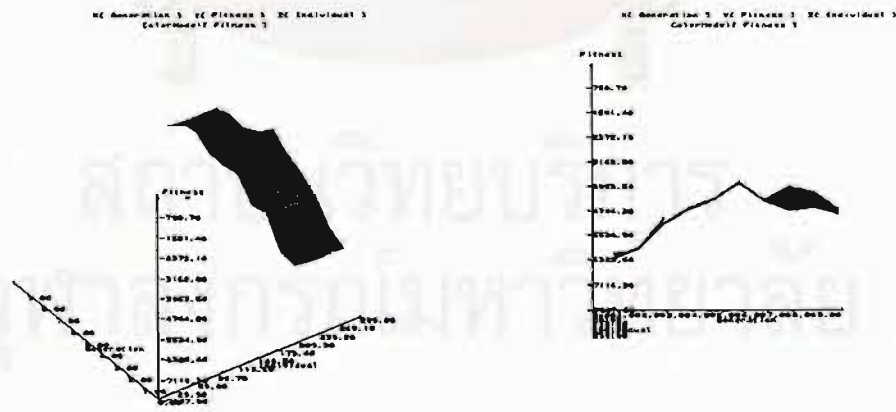
เมื่อใช้คุณสมบัติในการปรับการแสดงผลเปลี่ยนภาพดังกล่าวให้ดูง่ายขึ้นโดยปรับค่าเฉลี่ยในการแสดงผลไปที่ 20 จุด สำหรับคำตอบหมายเลข 1-40 คำตอบอื่นๆ ใช้การเฉลี่ย 50 จุด ผลที่ได้เป็นดังรูปที่ 46 ถึงรูปที่ 49 ด้านซ้าย ทางด้านขวาเป็นภาพเดียวกันที่ได้ปรับมุมมองใหม่ซึ่งแสดงความสัมพันธ์ระหว่างค่าความเหมาะสม (แกนตั้ง) กับรุ่น (แกนนอน)



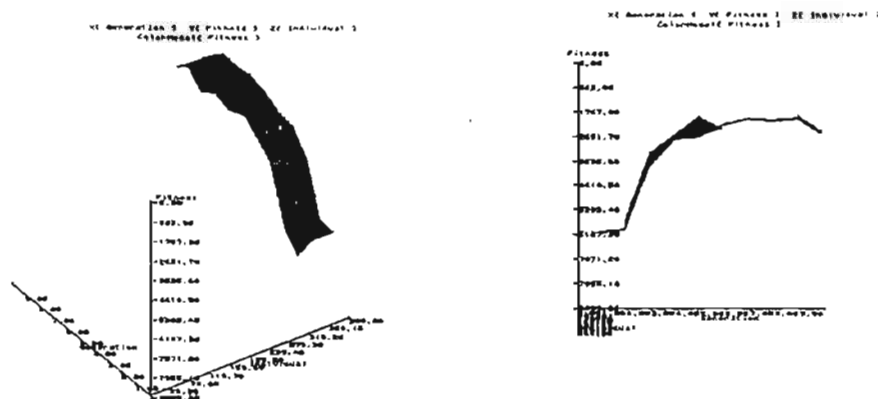
รูปที่ 46 หมายเลขคำตอบ 1 ถึง 40 โดยเฉลี่ย



รูปที่ 47 หมายเลขคำตอบ 41 ถึง 200 โดยเฉลี่ย

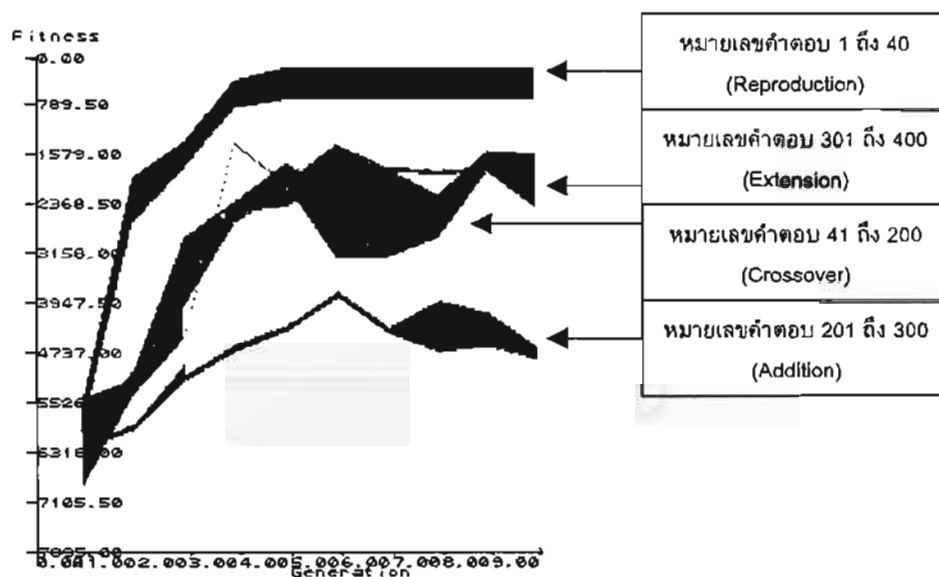


รูปที่ 48 หมายเลขคำตอบ 201 ถึง 300 โดยเฉลี่ย



รูปที่ 49 หมายเลขคำตอบ 301 ถึง 400 โดยเฉลี่ย

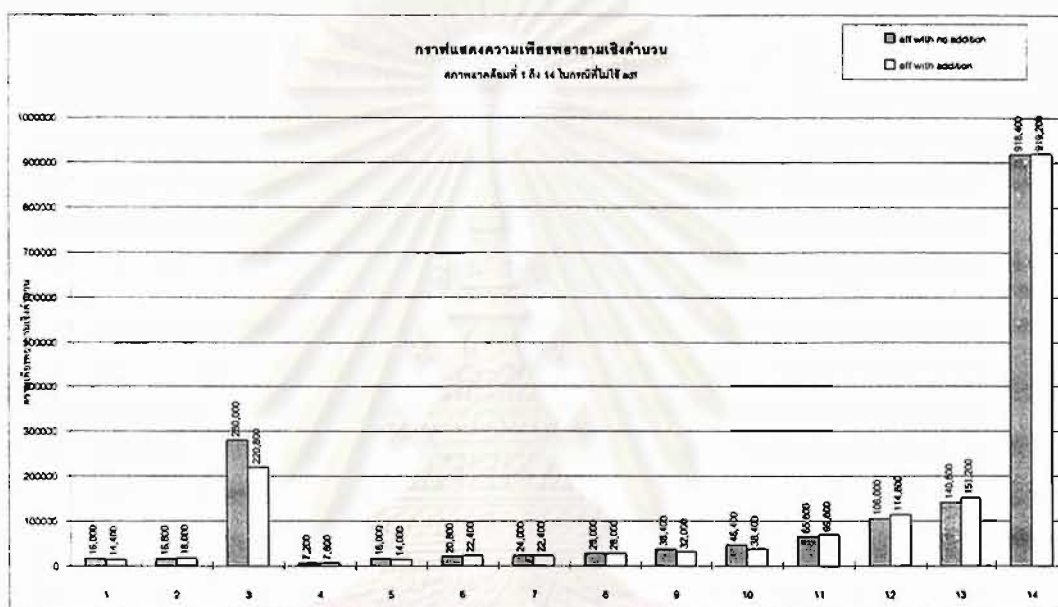
หลังจากได้ภาพที่แบ่งหมายเลขคำตอบทั้งหมดแล้ว นำรูปที่ 46 ถึงรูปที่ 49ด้านขวา
เมื่อนำมาเปรียบเทียบกัน ได้ตั้งรูปที่ 50



รูปที่ 50 ผลของฟังก์ชันวิวัฒนาการเปรียบเทียบระหว่างภาพที่ 46-49

จากรูปที่ 50 แสดงให้เห็นว่ากลุ่มหมายเลขคำตอบ 1-40 ซึ่งเกิดจากฟังก์ชันสืบพันธุ์ (reproduction) มีค่าความเหมาะสมโดยเฉลี่ยมากกว่ากลุ่มอื่น กลุ่มหมายเลขคำตอบคำตอบ 201-300 ซึ่งเกิดจากการกลายพันธุ์แบบต่อยอด (addition) มีค่าความเหมาะสมโดยเฉลี่ยน้อยกว่ากลุ่มอื่น ส่วนกลุ่มหมายเลขคำตอบ 41-200 ซึ่งเกิดจากการไขว้เปลี่ยน (crossover) และกลุ่มหมายเลขคำตอบ 301-400 ซึ่งเกิดจากการกลายพันธุ์แบบต่อปลาย (extension) มีค่าความเหมาะสมโดยเฉลี่ยใกล้เคียงกัน

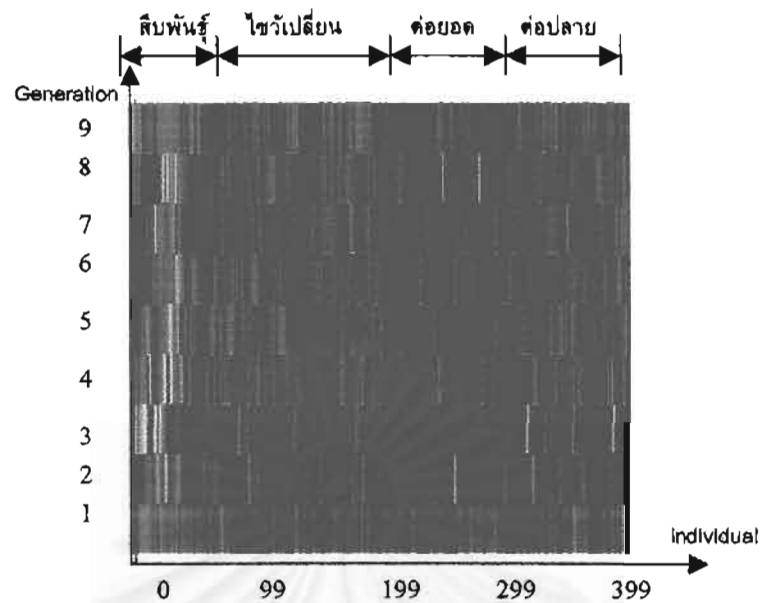
จากผลที่ได้จึงเกิดข้อสงสัยตามมาอีกว่าหากเป็นเช่นนี้แล้วควรใช้การกลายพันธุ์แบบด้อยอดดีหรือไม่เพราะได้ทำให้ค่าความเหมาะสมโดยเฉลี่ยน้อยถึง 1 ใน 4 ส่วน และคำตอบที่เกิดจากส่วนนี้ก็ไม่ช่วยในการแก้ปัญหาแขนหุ่นยนต์เลยหากเลิกใช้ควรจะทำให้กำหนดการเชิงพันธุกรรมได้คำตอบอย่างรวดเร็วกว่าเดิม ผู้วิจัยจึงได้ทำการทดลองโดยปรับการทดลองของชัยวัฒน์ใหม่ให้ใช้การไขว่เปลี่ยนแทนการกลายพันธุ์แบบด้อยอดทั้งหมด ได้ผลการทดลองดังรูปที่ 51



รูปที่ 51 ความเพียรพยายามเชิงคำนวณเปรียบเทียบในกรณีที่ใช้และไม่ใช้การกลายพันธุ์แบบด้อยอด

การใช้หรือไม่ใช้การกลายพันธุ์แบบด้อยอดไม่ทำให้ค่าความเพียรพยายามเชิงคำนวณแตกต่างกันมากนัก มีเพียงสภาพแวดล้อมที่ 3 ที่ใช้แล้วได้ผลดีกว่านอกจากนั้นแล้วได้ค่าใกล้เคียงกัน ซึ่งผิดจากที่ได้คาดเดาไว้ในตอนแรก ถ้าเป็นเช่นนี้แล้วการใช้หรือไม่ใช้การกลายพันธุ์แบบด้อยอดก็ไม่มีประโยชน์อย่างที่ Koza ได้กล่าวไว้เลยหรือ ถ้าผลที่ได้ไม่ต่างกันมากนักดังนั้นแล้วคำตอบที่ใช้แก้ปัญหาได้ในรุ่นสุดท้ายก็คงไม่เลือกใช้คำตอบที่ผ่านการกลายพันธุ์แบบด้อยอดซึ่งมีค่าความเหมาะสมน้อย ดังนั้นผู้วิจัยจึงเพิ่มการจินตทัศน์ โดยกำหนดให้โปรแกรมแสดงสีเพื่อของจำนวนลูกของคำตอบในโปรแกรมจินตทัศน์ข้อมูล

รูปที่ 52 สีดำแสดงว่าคำตอบนี้ไม่ได้ถูกนำไปเป็นต้นแบบเพื่อวิวัฒนาการเป็นคำตอบในรุ่นถัดไปหรือไม่มีลูกเลย สีน้ำเงินแสดงว่าคำตอบนี้มีลูกตั้งแต่ 1 ตัวตามระดับความสว่าง สีฟ้าแสดงว่าเกิดลูกเป็นจำนวนมากกว่า 10 ตัวโดยประมาณ



รูปที่ 52 การจินตทัศน์จำนวนลูกของคำตอบ

จากภาพที่ได้แสดงว่าคำตอบที่ถูกนำมาพัฒนาในรุ่นถัดไปเกิดในส่วนของคำตอบที่ มาจากการสืบพันธุ์และการไขว้เปลี่ยน การกลายพันธุ์แบบค่อยสลายบางส่วน และการกลาย พันธุ์แบบค่อยลดเป็นส่วนน้อย ในรุ่นที่ 3 เราจะเห็นแถบสีฟ้าอยู่บริเวณส่วนของคำตอบที่มา จากการสืบพันธุ์ แสดงว่าคำตอบส่วนใหญ่ในรุ่นที่ 4 มาจากส่วนนี้เกือบทั้งหมด และได้ข้อ สังเกตประการหนึ่งว่ามีแถบสีฟ้าเกิดขึ้นในบริเวณคำตอบที่เกิดจากการกลายพันธุ์ในบาง ตำแหน่ง จึงพบว่าการกลายพันธุ์ของคำตอบบางครั้งก็มีประโยชน์มากสามารถทำให้เกิดคำ ดอบที่ดีได้ ซึ่ง Luke S. และ Spector L. [13,14] ได้ศึกษาโดยเปรียบเทียบการใช้การไขว้ เปลี่ยนกับการ กลายพันธุ์โดยใช้ในปัญหาต่างๆ ใช้การสืบพันธุ์ 10% และการไขว้เปลี่ยน 90% กับการสืบพันธุ์ 10% และการกลายพันธุ์ 90% ผลที่ได้พบว่าการไขว้เปลี่ยนมีความ สำเร็จมากกว่าและหากมีจำนวนประชากรน้อยควรใช้การกลายพันธุ์จะดีกว่าใช้การไขว้ เปลี่ยน ดังนั้นผู้วิจัยเสนอว่าควรมีการศึกษาต่อไปว่าควรนำการกลายพันธุ์มาใช้เป็นอัตรา ส่วนเท่าใดถึงช่วยให้คำตอบที่ได้มีประสิทธิภาพไปยังเป้าหมายได้เร็วขึ้น หรือควรปรับปรุง การกลายพันธุ์อย่างไร เช่นให้มีรูปร่างที่แปลกกว่าเดิมมากๆ หรือสุ่มสร้างประชากรใหม่มา เพิ่มเลยจะดีหรือไม่ เป็นต้น

6.2 โครงสร้างของคำตอบที่ได้จากกำหนดการเชิงพันธุกรรม

ในการทดลองใช้โปรแกรมจินตทัศน์ข้อมูล ผู้วิจัยได้ให้โปรแกรมแสดงคำตอบในรูปของต้นไม้ และพบว่าคำตอบที่แสดงในรูปของต้นไม้โดยทั่วไปไม่สามารถแสดงถึงความแตกต่างได้อย่างชัดเจนเมื่อมีการเปลี่ยนแปลงโครงสร้างบางส่วน จึงได้ค้นหาวิธีที่จะแสดงโครงสร้างแบบอื่นเพื่อให้เห็นความแตกต่างได้ชัดเจนขึ้น หากบัพในส่วนต้นๆ มีความเปลี่ยนแปลงก็ควรจะเห็นการเปลี่ยนแปลงได้ทันที จึงได้แนวคิดว่าจะหากนำบัพมาจัดเรียงใหม่โดยให้ความสำคัญกับส่วนต้นก่อนส่วนปลายก็อาจจะแบ่งแยกได้ชัดเจนขึ้นโดยเห็นได้ว่าบัพในต้นไม้อาจจะไปรวมในแนวใดแนวหนึ่ง ดังนั้นผู้วิจัยจึงได้ทดลองสร้างการจินตทัศน์โครงสร้างในลักษณะทรงกลมโดยวิธีการดังนี้

(1) กำหนดจุดศูนย์กลางของโครงสร้าง (X_c, Y_c) กำหนดมุมตั้งต้น 360 องศา และวางบัพราก (Root Node) ไว้ตรงจุดศูนย์กลาง

(2) พิจารณาบัพลูก (Child Node) ของบัพรากว่ามี จำนวนเท่าใด จากนั้นนำไปหารมุมตั้งต้น เพื่อทำการแบ่งมุมตั้งต้นให้กับบัพลูกแต่ละบัพเช่น

มีบัพลูกจำนวน 4 บัพ นำ 4 ไปหาร 360 ได้ 90 องศา

ดังนั้นมุมตั้งต้นของบัพลูกที่ 1 มีองศาระหว่าง 0 ถึง 90

บัพที่ 2 มีองศาระหว่าง 90 ถึง 180

บัพที่ 3 มีองศาระหว่าง 180 ถึง 270

บัพที่ 4 มีองศาระหว่าง 270 ถึง 360

(3) จากมุมตั้งต้นที่ได้ใหม่นำมาหาร 2 เพื่อกำหนดเป็นจุดศูนย์กลางใหม่ มีระยะห่าง R จากจุดศูนย์กลางเดิม เช่นบัพที่ 3 เป็นมุม 225 องศา คำนวณจากระยะจุดศูนย์กลางเดิม ได้จุดศูนย์กลางใหม่เป็น $(X_n, Y_n) = (X_c + \cos(225) * R, Y_c + \sin(225) * R)$

(4) พิจารณาบัพลูกของบัพต่างๆ ดังข้อที่ 2 จนหมดโครงสร้างคำตอบ

ด้วยวิธีการนี้จะทำการแบ่งมุมให้มุมตั้งต้นของบัพที่ได้ใหม่จะค่อยเล็กลงเรื่อยๆ เช่นบัพที่ 2 มีองศาระหว่าง 90-179 หากบัพที่ 2 มี เริ่มตั้งแต่ 90 องศา คือ มีบัพลูก 4 บัพนำ 4 ไปหาร 0 ได้ 22.5 องศา โดยเริ่มจากมุมตั้งต้นที่ 90 องศา ดังนั้นมุมตั้งต้นของ

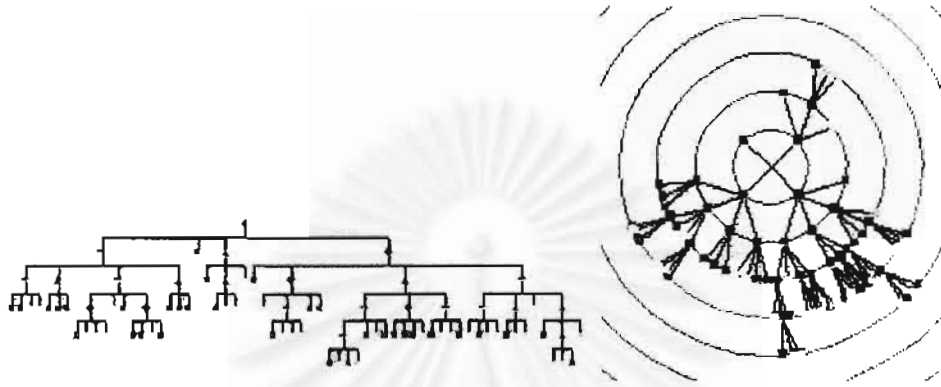
บัพที่ 1 มีองศาระหว่าง 90 ถึง $90+22.5 = 112.5$

บัพที่ 2 มีองศาระหว่าง 112.5 ถึง 135

บัพที่ 3 มีองศาระหว่าง 135 ถึง 157.5

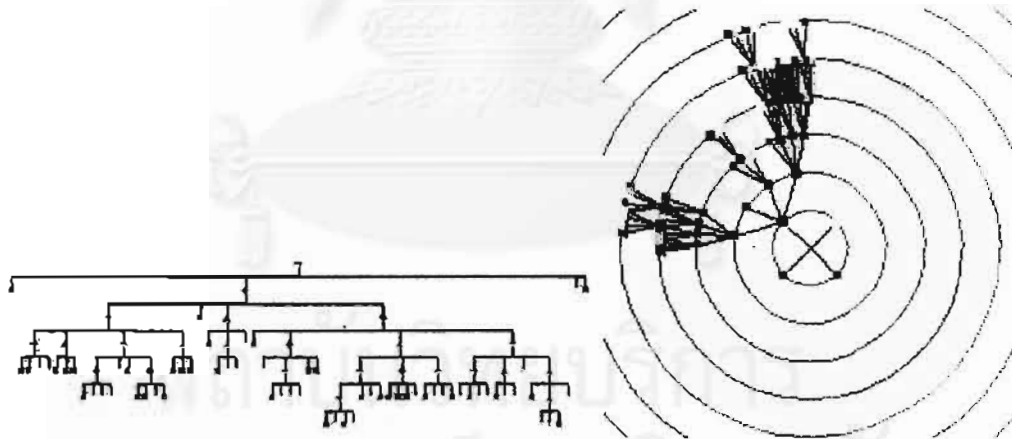
บัพที่ 4 มีองศาระหว่าง 157.5 ถึง 180

เมื่อคำนวณครบทั้งหมดแล้ว นำมาสร้างเป็นภาพได้ดังรูปที่ 53 (ขวา)



รูปที่ 53 การจินตทัศน์โครงสร้างข้อมูลคำตอบในแบบต้นไม้เดิม(ซ้าย) กับแบบทรงกลม(ขวา)

เมื่อมุมที่หารได้ลดลงมากๆ จะทำให้ node อยู่ชิดกันมาก ไม่สามารถแยกความแตกต่างบนจอภาพได้ ดังนั้นในโปรแกรมจึงได้กำหนดให้มุมลดลงได้ไม่เกิน 5 องศา หากน้อยกว่า 5 องศาโปรแกรมจะเพิ่มเป็น 5 องศา



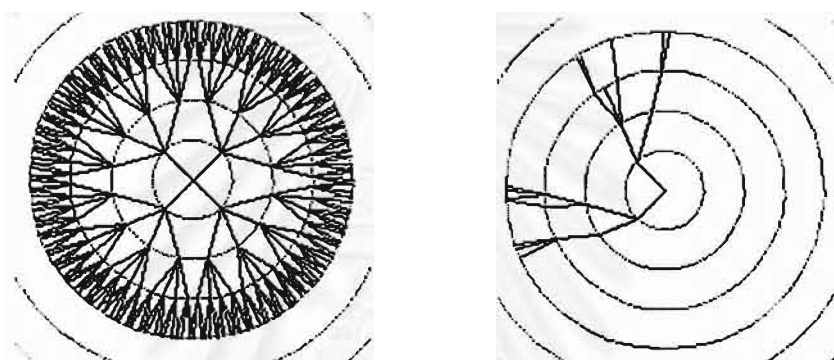
รูปที่ 54 การจินตทัศน์โครงสร้างข้อมูลคำตอบในแบบต้นไม้(ซ้าย) กับแบบทรงกลม(ขวา)

เมื่อผ่านการวิวัฒนาการ

เมื่อนำคำตอบที่แสดงไว้จากรูปที่ 53 ผ่านการวิวัฒนาการแบบการกลายพันธุ์แบบต่อยอดซึ่งโปรแกรมจะสุ่มสร้างต้นไม้ย่อยมาต่อทางด้านบัพแรก ทำให้โครงสร้างต้นไม้ส่วนบนเปลี่ยนแปลงไป ผลที่ได้สามารถแสดงดังรูปที่ 54 หากแสดงโครงสร้างเป็นรูปแบบเดิม

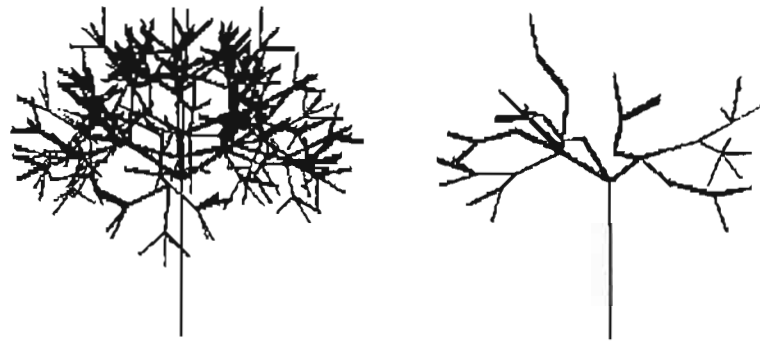
ก็จะไม่เห็นความแตกต่างมากนัก แต่เมื่อใช้การแสดงโครงสร้างข้อมูลคำตอบในแบบทรงกลมเมื่อผ่านการกลายพันธุ์แบบต่อยอดจะทำให้บัพมีลักษณะรวมกันอยู่ในทิศทางหนึ่ง

การแสดงโครงสร้างข้อมูลคำตอบในแบบทรงกลมแสดงให้เห็นถึงลักษณะการกระจายตัวของบัพภายในโครงสร้าง รูปที่ 54 (ขวา) แสดงว่าโครงสร้างต้นไม้มีบัพเรียงกันในแนวเส้นมากกว่าในแนวกว้าง หากบัพในต้นไม้มีลูกทุกตัวการแสดงแบบทรงกลมจะแสดงให้เห็นการกระจายตัวในอุดมคติหรือกระจายแบบสมบูรณ์ดังรูปที่ 55 ซึ่งในคำตอบที่ได้จากกำหนดการเชิงพันธุกรรมไม่มีคำตอบใดที่มีโครงสร้างสมบูรณ์แบบ



รูปที่ 55 โครงสร้างต้นไม้แบบทรงกลมสมบูรณ์ (ซ้าย) เปรียบเทียบกับแบบไม่สมบูรณ์ (ขวา)

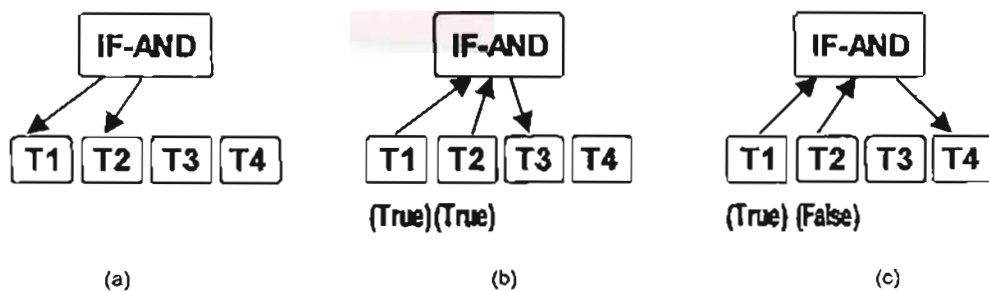
จากสิ่งที่ได้มาเรายังไม่สามารถสรุปได้ว่าการเชื่อมต่อแบบใดที่จะทำให้คำตอบดีหรือไม่ดี เหตุผลประการหนึ่งเป็นเพราะค่าความเหมาะสมขึ้นอยู่กับประเภทของบัพในต้นไม้เป็นหลัก การแสดงจินตทัศน์ในรูปแบบนี้จึงเป็นเพียงแนวทางหนึ่งซึ่งผู้วิจัยเสนอในการแสดงโครงสร้างต้นไม้เท่านั้น และผู้วิจัยได้ทดลองต่อไปโดยพัฒนาการแสดงโครงสร้างให้อยู่ในรูปแบบ 3 มิติซึ่งใช้แนวคิดการแบ่งมุมออกเป็นส่วนย่อยๆ เช่นเดียวกับ 2 มิติ ดังรูปที่ 56 โดยคาดหวังว่าจะสร้างเป็นป่า (Forest) ของคำตอบที่ได้จากกำหนดการเชิงพันธุกรรม แต่เนื่องจากในการแสดงผลช่วงทดลองผู้วิจัยพบว่าดูได้ยากทำให้ยังไม่สามารถสรุปอะไรได้มากนัก และใช้เวลาในการประมวลผลเพื่อการแสดงผลเป็นอย่างมาก จึงไม่ได้มีส่วนฟังก์ชันแสดงโครงสร้าง 3 มิติไว้ในโปรแกรม GPVIS



รูปที่ 56 โครงสร้างต้นไม้ 3 มิติแบบสมบูรณ์(ซ้าย) และแบบไม่สมบูรณ์(ขวา)

6.3 ต้นไม้และเส้นทางที่ถูกใช้งาน

หลายท่านเชื่อว่าต้นไม้ไม่มีเส้นทาง (Path) ภายในซึ่งผ่านบัพต่างๆ หากเริ่มต้นที่บัพรากไปจนถึงปลายด้วยการควบคุมของบัพซึ่งเป็นฟังก์ชัน จะได้เส้นทางย่อยๆ หลายเส้นทางที่แตกต่างกันตามแต่ละสถานะจากบัพลูกที่ส่งกลับมา เช่น บัพ IF-AND มีบัพลูก 4 บัพ และเรียกใช้ 2 บัพแรกก่อน รูปที่ 57 (a) จากนั้นจึงดูสถานะ 2 บัพว่าควรจะเรียกใช้บัพลูกตัวที่ 3 หรือที่ 4 หากบัพลูก 2 บัพแรกมีค่าเป็นจริงทั้งคู่ IF-AND จะเรียกใช้บัพลูกที่ 3 และไม่ใช้บัพลูกที่ 4 รูปที่ 57 (b) หากบัพลูก 2 บัพแรกมีค่าเป็นอื่น เช่น จริงและเท็จ IF-AND จะเรียกใช้บัพลูกที่ 4 ดังรูปที่ 57 (c)



รูปที่ 57 เส้นทางที่เกิดจากสถานะการเรียกใช้บัพลูก

ในคำตอบที่ได้จากกำหนดการเชิงพันธุกรรม บัพที่เป็นฟังก์ชันจะตัดสินใจเลือกบัพตัวใดตัวหนึ่งซึ่งพิจารณาจากบัพลูก 2 ตัวแรก หรือ 1 ตัวแรกในกรณีที่เป็นบัพฟังก์ชัน IF-NOT ดังนั้นผู้วิจัยคาดว่ามีความเป็นไปได้อย่างมากที่จะมีบางบัพซึ่งไม่ถูกเรียกใช้งานเลย

หากเราพิจารณาข้อกำหนดของปัญหาแขนหุ่นยนต์จะพบว่าเงื่อนไขที่ใส่ไว้ในปัญหามีทั้งหมด 5 อย่าง คือ inc, dec, see, hit และ die แต่ละอย่างจะให้ค่าจริงหรือเท็จขึ้นอยู่กับสภาพแวดล้อมและตำแหน่งของแขนหุ่นยนต์ในขณะนั้น และบัพอื่นจะให้ค่าเป็นจริงทั้งหมด* จากเงื่อนไขนี้เราสามารถแจกแจงได้ทั้งหมด 32 กรณีที่เป็นไปได้ ในการสลับค่าระหว่างจริงและเท็จทั้ง 5 เงื่อนไข ซึ่งทำให้เกิดกรณีที่เป็นไปได้ทั้งหมด 32 เส้นทางในคำตอบหนึ่งๆ หากคำตอบนั้นมีบัพเงื่อนไขครบทั้ง 5 อย่าง

การจินตทัศน์เส้นทางภายในต้นไม้คำตอบจึงได้ถูกเพิ่มเติมไว้ในโปรแกรมจินตทัศน์เพื่อแสดงว่าเส้นทางภายในเกิดความเปลี่ยนแปลงอย่างไรในขณะที่คำตอบนั้นนำไปใช้งานเริ่มแรกโปรแกรมจะกำหนดให้ค่าสถานะทั้งหมดเป็น 0 จากนั้นจึงปรับปรุงค่าสถานะต่างๆ ของแขนหุ่นยนต์ จนกระทั่งครบเงื่อนไขคือ ปลายแขนสัมผัสเป้าหมายภายในระยะที่กำหนดหรือเกิดความเสียหายจนแขนขยับไม่ได้ ซึ่งนับจากจำนวนครั้งที่ชน หรือ แขนหุ่นยนต์ขยับอยู่กับที่จนครบกำหนดเวลาหนึ่ง

เพื่อให้เห็นรายละเอียดเราจะมาพิจารณาการเคลื่อนไหวของแขนหุ่นยนต์ โดยนำข้อมูลมาจาก track log file ของโปรแกรมจินตทัศน์ข้อมูล ที่ควบคุมโดยคำตอบที่ 18 รุ่นที่ 10 ซึ่งเป็นคำตอบที่ดีที่สุด จาก track log file แสดงให้เห็นว่าคำตอบ ไม่จำเป็นต้องใช้ทั้ง 32 เส้นทางที่เป็นไปได้ จากตัวอย่างนี้ได้เรียกใช้ทั้งหมดเพียง 5 เส้นทางโดยเรียกทั้งหมดเป็นจำนวน 20 ครั้งคือ 0, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 4, 10, 26 และ 26 ตามลำดับจึงทำให้แขนถึงเป้าหมาย

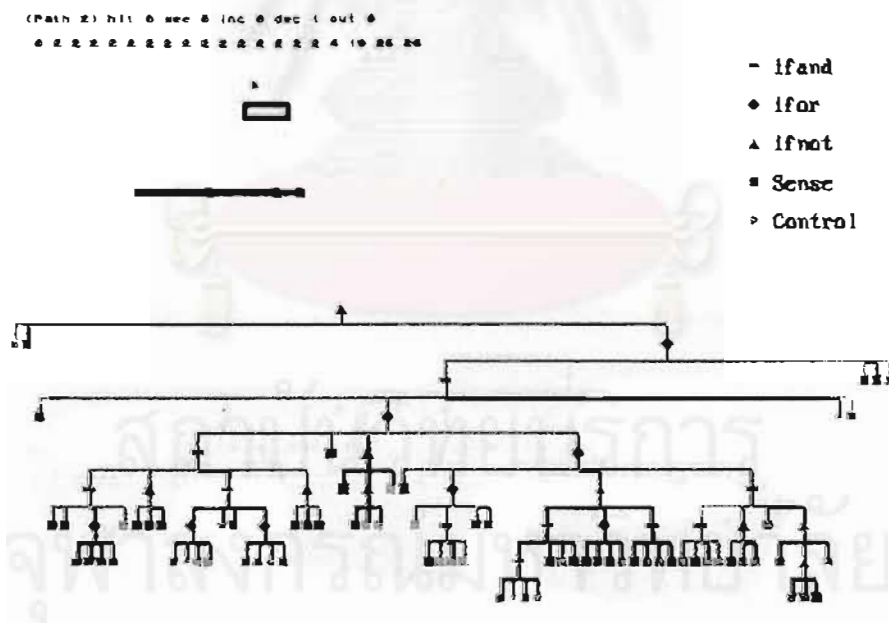
ตาราง 10 ค่าสถานะของเส้นทางที่ถูกใช้งานในต้นไม้จากคำตอบที่ 18 รุ่นที่ 10

ค่าสถานะ					หมายเลขเส้นทาง	จำนวนครั้งที่เรียกใช้เส้นทาง	จำนวนบัพที่ใช้งาน [†]
Hit	see	inc	dec	Out			
0	0	0	0	0	0	1	39
0	0	0	1	0	2	15	39
0	0	1	0	0	4	1	24
0	1	0	1	0	10	1	30
1	1	0	1	0	26	2	30

* ดูรายละเอียดได้จากหัวข้อ 2.2

[†] จำนวนบัพที่ใช้งานจะนับรวมบัพเทอมินอลและบัพฟังก์ชันด้วย

กล่าวได้ว่าเส้นทางที่ 2 ถูกเรียกใช้งานมากที่สุด และเมื่อเรียกใช้งานแล้วยังคงทำให้ค่าสถานะยังคงอยู่ในสถานะเดิมคือ $dec=1$ ซึ่งเป็นการลดระยะห่างระหว่างปลายแขนกับเป้าหมาย โดยที่มีสิ่งกีดขวางระหว่างปลายแขนกับเป้าหมาย ($see=0$) เมื่อใช้งานมากขึ้นทำให้ระยะปลายแขนกับเป้าหมายเพิ่มมากขึ้นจึงไปเรียกใช้เส้นทางที่ 4 หลังจากนั้นใช้เส้นทางที่ 10 และ 26 ไม่มีสิ่งกีดขวางระหว่างแขนกับเป้าหมาย ($see=1$) และระยะห่างลดลง ($dec=1$) เพื่อนำปลายแขนไปสู่เป้าหมาย สำหรับการเคลื่อนไหวของแขนหุ่นยนต์ที่สัมพันธ์กับเส้นทางต่างๆ ได้แสดงไว้ดังรูปที่ 58 (a)-(e) ตัวเลขในวงเล็บที่ตามหลังแสดงถึงหมายเลขของกรอบภาพ (Frame) ของการเคลื่อนไหวซึ่งในการเคลื่อนที่ไปยังเป้าหมายของแขนหุ่นยนต์ ในคำตอบนี้ตั้งแต่เริ่มต้นจนถึงเป้าหมายจะมีทั้งหมด 64 กรอบภาพ ภาพต้นไม้ซึ่งเป็นโครงสร้างของคำตอบ แสดงเส้นทางที่ใช้งานโดยเส้นสีเขียวเป็นเส้นทางที่ใช้งาน และเส้นสีดำแทนส่วนของเส้นทางที่ไม่ได้ใช้งาน

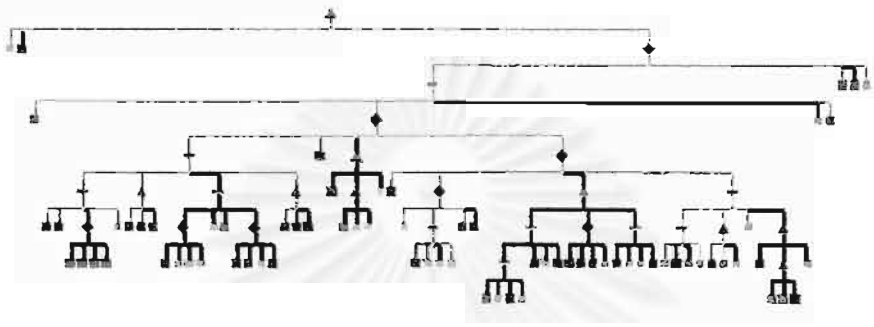


(a-2)

(Path 0) hit @ see @ inc @ dec @ out @
> 0 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 4 10 26 26



- ifand
- ifor
- ▲ ifnot
- Sense
- * Control

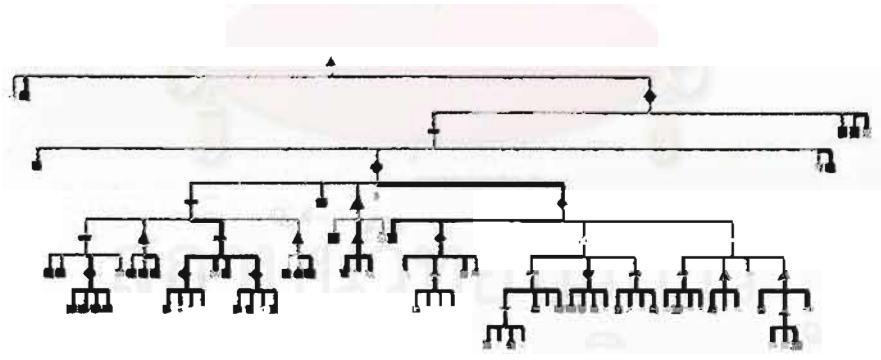


(b-4)

(Path 4) hit @ see @ inc @ dec @ out @
0 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 4 10 26 26



- ifand
- ifor
- ▲ ifnot
- Sense
- Control

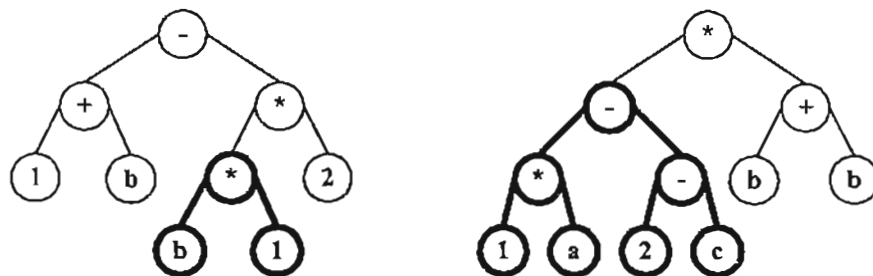


(c-35)

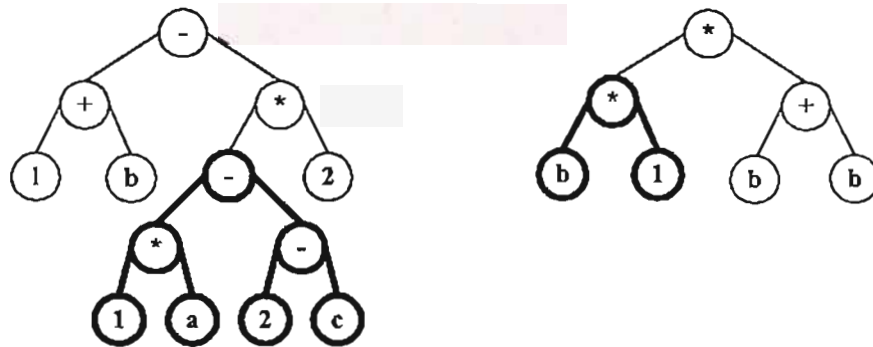
อยู่ในระยะที่ไม่มีสิ่งกีดขวางระหว่างปลายแขนกับเป้าหมายรูปที่ 58 (b) และรูปที่ 58 (c) ตั้งแต่ช่วงนี้โครงสร้างทางด้านขวาล่างจะไม่ถูกเรียกใช้งาน และมีส่วนของบัพที่ไม่เคยถูกเรียกใช้งานเลยเมื่อสังเกตตั้งแต่เริ่มจนถึงสิ้นสุดการทำงาน อยู่ในส่วนที่ได้ทำเครื่องหมายวงกลมไว้ในรูปที่ 58 (e) การจินตทัศน์เส้นทางภายในต้นไม้แสดงว่าต้นไม้มีบางส่วนที่ไม่ได้ถูกนำมาใช้ประโยชน์เลย

6.4 การไขว้เปลี่ยนและการกลายพันธุ์

การไขว้เปลี่ยนคือการนำเอาคำตอบ 2 คำตอบมาสลับตำแหน่งที่จะทำการไขว้เปลี่ยนในแต่ละคำตอบจากนั้นจะตัดส่วนปลายจากตำแหน่งที่สลับมาดังกล่าวเพื่อสลับกัน จะได้เป็นคำตอบใหม่ 2 คำตอบที่มีลักษณะคล้ายคำตอบเดิม



(ก)

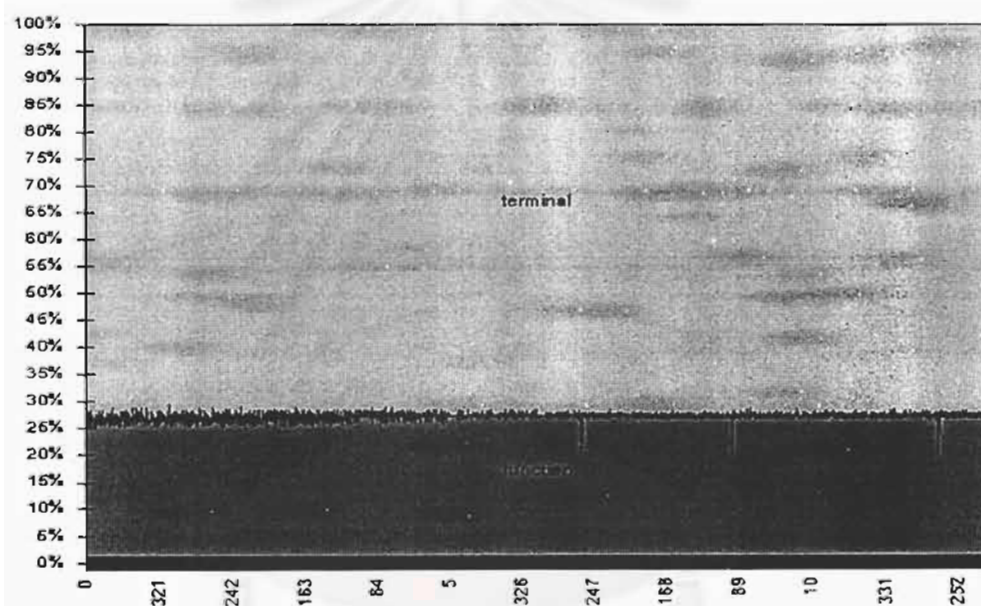


(ข)

รูปที่ 59 แสดงตัวอย่างการไขว้เปลี่ยน (crossover) จากคำตอบ (ก) เป็นคำตอบ (ข)^{*}

^{*} ภาพจาก ชัยวัฒน์ เจริญภาพกรณ์ [4] หน้า 7 รูปที่ 1.1

ในการไขว้เปลี่ยนโดยทั่วไปคำตอบไม่ได้เกิดตามที่แสดงไว้ตามรูปที่ 59 ดังกล่าวเสมอไป หลายครั้งที่การจินตทัศน์ข้อมูลได้แสดงการไขว้เปลี่ยนในกรณีที่ต้นไม้ทั้งสองต้นเกิดขึ้น ณ ตำแหน่งบัพโบนั้นก็เหมือนกับการกลายพันธุ์แบบจุด (Point Mutation) ดังที่ Koza ได้กล่าวไว้ ทำให้ผู้วิจัยเกิดความต้องการคำอธิบายที่เกี่ยวกับการเกิดปรากฏการณ์การไขว้เปลี่ยนแบบนี้ จากการตรวจสอบโปรแกรมและข้อมูลต่างๆ จึงได้ข้อสรุปว่าตำแหน่งที่สุ่มคำตอบในการไขว้เปลี่ยนนั้นมีความน่าจะเป็นเท่ากันทุกตำแหน่ง และจากรูปที่ 60 ซึ่งเป็นกราฟแสดงอัตราส่วนจำนวนบัพทั้งหมดของแต่ละคำตอบ จำนวนบัพที่เป็นฟังก์ชัน และจำนวนบัพที่เป็นเทอมินอลในสภาพแวดล้อมนี้ จะเห็นได้ว่าโอกาสที่การสุ่มตำแหน่งภายในแล้วได้ตำแหน่งบัพที่เป็นฟังก์ชันนั้น มีอัตราส่วนโดยประมาณ 25% และโอกาสที่สุ่มตำแหน่งได้บัพที่เป็นเทอมินอลมีโอกาสประมาณ 75%



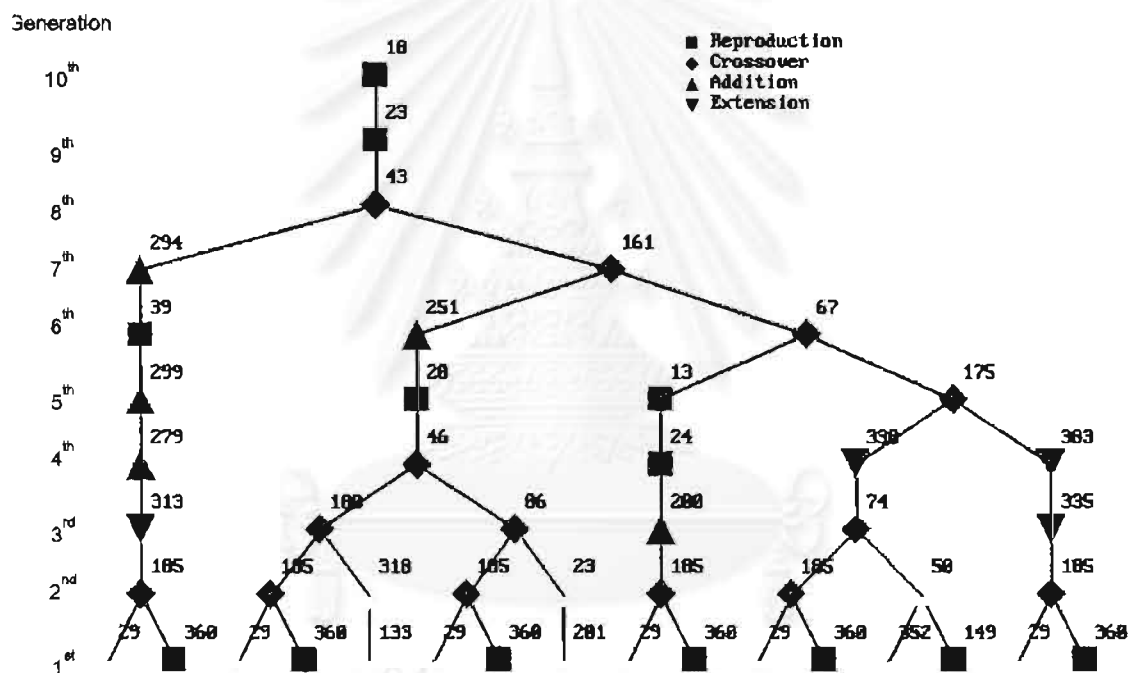
รูปที่ 60 แสดงอัตราส่วนระหว่างบัพฟังก์ชันและเทอมินอลจากคำตอบทั้งหมดในสภาพแวดล้อม 1180000

จากอัตราส่วนดังกล่าวจึงทำให้ การไขว้เปลี่ยนเกิดการสุ่มได้เทอมินอลมากกว่าฟังก์ชัน หากคิดในกรณีที่สุ่มตำแหน่งจากคำตอบทั้งสองแล้วได้เทอมินอลทั้งคู่ จึงเป็นการแลกเปลี่ยนกันเพียงบัพเดียว ซึ่งมีโอกาสถึง 0.5625 จากตาราง 11 ดังนั้นการไขว้เปลี่ยนจึงไม่ได้ก่อให้เกิดความเปลี่ยนแปลงกับโครงสร้างต้นไม้ของคำตอบใหม่อย่างที่ผู้วิจัยเคยคาดไว้มากนัก

ตาราง 11 ความน่าจะเป็นโดยประมาณในการเกิดการไขว้เปลี่ยนของคำตอบที่บัพแต่ละประเภท

	Function (0.25)	Terminal (0.75)
Function (0.25)	0.0625	0.1875
Terminal (0.75)	0.1875	0.5625

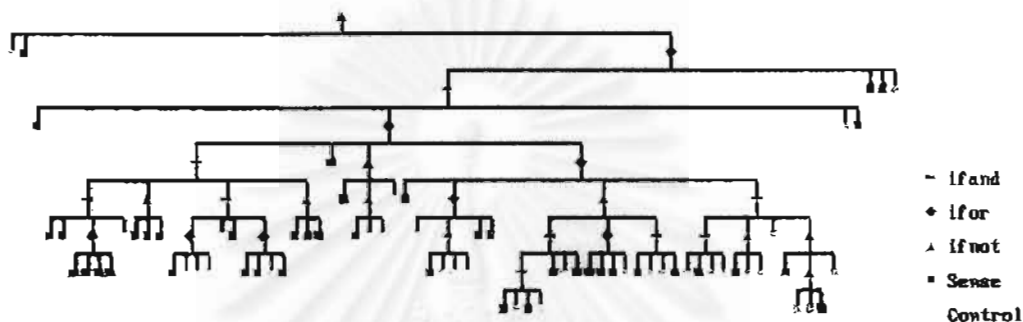
เรามาพิจารณาการวิวัฒนาการซึ่งแสดงต้นกำเนิดของคำตอบที่ 18 ในรูปที่ 61 คำตอบได้เกิดจากการไขว้เปลี่ยน การสืบพันธุ์และการกลายพันธุ์ทั้งสองแบบ ผสมรวมกันได้เป็นคำตอบที่ 18



รูปที่ 61 ความสัมพันธ์ในระหว่างการพัฒนาการของคำตอบที่ดีที่สุด (หมายเลข 18) ในรุ่นสุดท้าย ในสภาพแวดล้อมที่ 1180000

สิ่งหนึ่งซึ่งสังเกตได้จากรูปที่ 61 ก็คือในรุ่นที่ 2 มีคำตอบหมายเลข 185 หลายตัว เนื่องจากคำตอบนี้มีค่าความเหมาะสมมากและได้ถูกเลือกเพื่อพัฒนาเป็นคำตอบได้รุ่นต่อไป คำตอบที่ดีที่สุดนี้เกิดจากคำตอบเริ่มต้นเพียงไม่กี่ตัวเท่านั้น

เราจะเห็นส่วนที่เกิดการไขว้เปลี่ยนในหลายครั้งที่ค่าความเหมาะสมเท่ากับคำตอบตัวใดตัวหนึ่งในรุ่นเดิม เป็นสิ่งหนึ่งซึ่งอธิบายปรากฏการณ์การเกิดการกลายพันธุ์แบบจุดนี้ได้เป็นอย่างดี เช่นในรุ่นที่ 8 คำตอบหมายเลข 43 เกิดจากคำตอบหมายเลข 294 และคำตอบหมายเลข 161 ในรุ่นที่ 7 ได้ค่าความเหมาะสมเป็น -27 ดังนั้นผลของการไขว้เปลี่ยนควรจะมีส่วนผสมของคำตอบหมายเลข 294 มากกว่าหมายเลข 161 จึงทำให้ได้ค่าความเหมาะสมเป็นเช่นนั้น



รูปที่ 62 โครงสร้างคำตอบที่ 18 รุ่นที่ 10

หากเรามาดูทางด้านโครงสร้างของคำตอบที่ 18 ในรุ่นที่ 10 เราจะพบว่าโครงสร้างนี้ผ่านการกลายแบบต่อยอดถึง 3 ครั้ง เพราะว่าการกลายแบบต่อยอดนั้นโปรแกรมจะสุ่มสร้างต้นไม้ขนาดความสูง 1 บัพมาทำการเพิ่มที่ส่วนยอดของโครงสร้างหรือบัพราก เมื่อเรียงลำดับการกลายแบบต่อยอดของโครงสร้างนี้จะสามารถเรียงได้ดังนี้ ครั้งแรกเป็นบัพ IF-AND ครั้งที่สองเป็นบัพ IF-OR และครั้งสุดท้ายเป็นบัพ IF-NOT ตามลำดับ

ดังนั้นโครงสร้างในคำตอบนี้จึงเป็นการสนับสนุนว่าส่วนผสมของบัพได้มาจากคำตอบหมายเลข 294 ในรุ่นที่ 7 เกือบทั้งหมดยกเว้นเพียงบัพเดียว ซึ่งก่อนที่จะมาเป็นคำตอบหมายเลข 294 หากเราย้อนไปดูจะพบว่าคำตอบได้มีการกลายแบบต่อยอด 3 ครั้ง กลายแบบต่อยอด 1 ครั้ง และการไขว้เปลี่ยนเกิดเพียงครั้งเดียวในรุ่นที่ 2 จากจุดนี้แสดงให้เห็นว่าการวิวัฒนาการที่ทำให้เกิดคำตอบหมายเลข 161 ในรุ่นที่ 7 ที่มีเป็นจำนวนมากนั้นไม่ก่อให้เกิดความเปลี่ยนแปลงกับคำตอบที่ 18 ในรุ่นที่ 10 เลย เป็นการเสียเวลาโดยเปล่าประโยชน์

ต้นกำเนิดของคำตอบที่ 18 รุ่นที่ 10 โดยแท้จริงแล้วมากกว่าคำตอบเพียง 2 ตัวคือ คำตอบหมายเลข 29 และ 360 ในรุ่นที่ 1 นำมาไขว้เปลี่ยนกันเพียงครั้งเดียวที่ทำให้เกิดประโยชน์มากที่สุด การเกิดคำตอบในรูปแบบนี้ผู้วิจัยเสนอว่าควรนำมาศึกษาต่อไป ซึ่งอาจนำมาช่วยในการลดความเพียรพยายามเชิงคำนวณได้

ในบทนี้เราได้นำการจินตทัศน์มาเป็นเครื่องมือช่วยในการค้นหาและอธิบายข้อสงสัย และสมมติฐานต่างๆ ในด้านของการวิวัฒนาการคำตอบ โครงสร้างต้นไม้คำตอบ เส้นทางภายในต้นไม้ ส่วนที่มีประโยชน์หรือส่วนที่ไม่ได้ถูกใช้งาน รวมทั้งต้นกำเนิดของคำตอบ สิ่งต่างๆ เหล่านี้ได้อธิบายด้วยการจินตทัศน์ในหลากหลายรูปแบบ เมื่อนำการจินตทัศน์มาประกอบกับสมมติฐานแล้ว การจินตทัศน์จะมีส่วนช่วยสนับสนุนหรือโต้แย้งเพื่อการตัดสินใจว่าเราควรเชื่อถือแนวคิดใด



สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 7

สรุป

7.1 เกี่ยวกับงานวิจัย

ในงานวิจัยนี้ได้แสดงถึงการจินตทัศน์ข้อมูลกำหนดการเชิงพันธุกรรม และระบบการจินตทัศน์ข้อมูล GPVIS ที่ได้มาจากข้อมูลกำหนดการเชิงพันธุกรรมของชัยวัฒน์ เจริญภาพกรณ์ ผู้วิจัยได้กล่าวถึงโครงสร้างของระบบ GPVIS ความสามารถของระบบและได้นำระบบ GPVIS มาใช้ในการจินตทัศน์ข้อมูลในรูปแบบต่างๆ ดังนี้ โครงสร้างคำตอบ การจำลองการทำงานของคำตอบ การวิวัฒนาการโครงสร้างคำตอบ ค่าเฉลี่ยค่าความเหมาะสมที่ได้จากตัวปฏิบัติการพันธุกรรมต่างๆ เส้นทางที่ถูกใช้งานภายในโครงสร้างต้นไม้ของคำตอบ เพื่อเป็นสิ่งที่ช่วยตอบข้อสงสัยของผู้วิจัยเกี่ยวกับกำหนดการเชิงพันธุกรรมก่อนทำงานวิจัยชิ้นนี้ และทำให้เข้าใจกำหนดการเชิงพันธุกรรมมากขึ้นกว่าเดิม เช่นในเรื่องของการไขว้เปลี่ยน และเส้นทางภายในต้นไม้ เป็นต้น เราไม่อาจศึกษากำหนดการเชิงพันธุกรรมได้โดยง่ายเนื่องจากข้อมูลเป็นจำนวนมากของกำหนดการเชิงพันธุกรรม และความเกี่ยวข้องกันระหว่างข้อมูลต่างๆ ในหลากหลายรูปแบบ การนำการจินตทัศน์มาใช้ช่วยให้เราศึกษา กำหนดการเชิงพันธุกรรมได้ง่ายขึ้น

7.2 แนวทางในการพัฒนาโปรแกรมต่อไปสำหรับอนาคต

ปัญหาและแนวทางในการพัฒนาระบบจินตทัศน์ต่อไปนี้เป็นเพียงความคิดเห็นส่วนหนึ่งที่อาจปรับปรุงให้ระบบจินตทัศน์กำหนดการเชิงพันธุกรรมดีขึ้น

7.2.1 ข้อจำกัดของระบบ

เนื่องจากการจินตทัศน์ต้องการระบบคอมพิวเตอร์ที่มีความสามารถสูง มีหน่วยความจำที่มาก มีเนื้อที่ว่างในฮาร์ดดิสก์สำหรับเก็บข้อมูล และระบบฮาร์ดแวร์ช่วยเร่งความเร็วในการแสดงผล ปัญหาที่พบในระบบการจินตทัศน์ปัจจุบันคือความล่าช้าในการ

แสดงผล และข้อจำกัดในการแสดงผล ซึ่งเป็นข้อจำกัดข้อหนึ่งที่มีผลอย่างมากในการออกแบบจินตทัศน์ครั้งนี้

7.2.2 ส่วนเพิ่มขยายระบบการจินตทัศน์

ระบบจินตทัศน์อาจพัฒนาต่อโดยมีส่วนเพิ่มขยายระบบการจินตทัศน์ (GPVIS Plugin) เชื่อมต่อกับโปรแกรมส่วนอื่นผ่านทาง PIPE ซึ่งโปรแกรมที่เพิ่มเข้ามาอาจจะเป็นส่วนฟังก์ชันกำหนดการเชิงพันธุกรรมหรือฟังก์ชันเกี่ยวกับการจินตทัศน์ก็ได้ โดยระบบจินตทัศน์จะทำหน้าที่จินตทัศน์โดยรับข้อมูลจากโปรแกรมกำหนดการเชิงพันธุกรรมอื่น

7.2.3 การจินตทัศน์กำหนดการเชิงพันธุกรรมอื่นๆ

ระบบจินตทัศน์ได้ใช้ข้อมูลกำหนดการเชิงพันธุกรรมจากโปรแกรมกำหนดการเชิงพันธุกรรมของชัยวัฒน์ เจษฎาภรณ์ ระบบที่พัฒนาทั้งหมดจะผูกติดกับการใช้ข้อมูลกำหนดการเชิงพันธุกรรมในปัญหาแขนหุ่นยนต์เท่านั้น หากต้องการนำระบบที่ได้ไปใช้ในโปรแกรมกำหนดการเชิงพันธุกรรมปัญหาอื่นที่แตกต่างจากนี้ ต้องปรับปรุงโปรแกรมในส่วนต่างๆ เช่น ส่วนสร้างข้อมูล ส่วนอ่านข้อมูล รวมทั้งส่วนจำลองการทำงานด้วย

7.2.4 การจินตทัศน์บน WEB-Browser

ระบบจินตทัศน์เรียกใช้ library MESA [12] ในส่วนของการสร้างภาพด้วยคำสั่ง OpenGL ซึ่ง library นี้จะมีส่วนที่เรียกว่า OFF LINE RENDERER MODE ทำให้สามารถทำงานได้ไม่ต้องเรียกคำสั่ง Open Display ใน X-Window ส่งผลให้ไม่จำเป็นต้องให้โปรแกรมทำงานบน X-Window ดังนั้นหากมีการแก้ไขระบบการจินตทัศน์โดยเพิ่มเติมให้มีส่วนของการจินตทัศน์แบบ OFF LINE และจัดเก็บเป็นไฟล์ JPEG ในอนาคตผู้ใช้งานอาจควบคุมการจินตทัศน์ผ่านทาง WEB-Browser ด้วย CGI เพื่อเรียกระบบจินตทัศน์มาทำงานบน Server โดยไม่ต้องเปิด X-Window และส่งภาพที่ได้การจินตทัศน์กลับมาแสดงบน Browser ทำให้ผู้ใช้งานสามารถเรียกใช้การจินตทัศน์จากที่ใดหรือระบบปฏิบัติการใดก็ได้ที่ Browser สามารถใช้งานได้

7.2.5 มาตรฐานสำหรับข้อมูลกำหนดการเชิงพันธุกรรม

จากการเขียนโปรแกรมกำหนดการเชิงพันธุกรรมในงานต่างๆ พบว่าในโปรแกรมกำหนดการเชิงพันธุกรรมควรจะมีมาตรฐานบางอย่างสำหรับข้อมูลกำหนดการเชิงพันธุกรรมซึ่งสามารถนำมาใช้งานร่วมกันได้ในปัญหาต่างๆ และจะทำให้สามารถสร้างเป็น Component หรือ Module เบื้องต้นสำหรับกำหนดการเชิงพันธุกรรม ซึ่งอาจเป็นส่วนสร้าง

คำตอบเริ่มต้น ส่วนพัฒนาคำตอบด้วยการวิวัฒนาการแบบต่างๆ เป็นต้น ถ้ามาตรฐานสำหรับข้อมูลกำหนดการแข่งขันธุรกรรม เป็นสิ่งที่เป็นไปได้จะทำให้ช่วยอำนวยความสะดวกในการพัฒนาโปรแกรมเกี่ยวกับกำหนดการแข่งขันธุรกรรมอย่างยิ่ง

ในหนังสือเทคนิคการเขียนโปรแกรมภาษาแอสเซมบลีซึ่งเขียนโดย ปีเตอร์ นอร์ดัน และ จอห์น โซซา [15] มีอยู่ส่วนหนึ่งได้กล่าวไว้ว่า ไม่มีโปรแกรมใดที่จะสมบูรณ์ที่สุด ดังนั้นจึงไม่มีจุดสิ้นสุดของโปรแกรมในการพัฒนา ระบบการจินตทัศน์นี้ก็เช่นเดียวกัน เราจะสามารถพัฒนาโปรแกรมต่อไปได้อย่างไม่มีที่สิ้นสุด แม้ในปัจจุบันอาจจะยังไม่เห็นสิ่งที่เป็นไปได้ แต่สำหรับอนาคตจะมีแนวคิดและแรงบันดาลใจใหม่ๆ เกิดขึ้นในการจินตทัศน์อย่างแน่นอน



สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

รายการอ้างอิง

1. Koza, J.R. Genetic Programming: On the Programming of Computers by Natural Selection. Cambridge, MA: MIT Press.1992.
2. Koza, J.R. Genetic Programming II: Automatic Discovery of Reuseable Programs. Cambridge, MA: MIT Press.1996.
3. Wongsiriprasert, C., Chongstitvatana, P. and Prasitjutrakul, S. Visualization of evolutionary process in genetic programming. Proceedings of The Second Annual National Symposium on Computational Science and Engineering, pp. 249-254, Bangkok, March 25-27, 1998.
4. ชัยวัฒน์ เจษฎาปกรณ์. การลดทอนความเพียรพยายามเชิงคำนวณของวิธีการเรียนรู้แบบกำหนดการเชิงพันธุกรรม. วิทยานิพนธ์ปริญญาโทมหาบัณฑิต ภาควิชาวิศวกรรมคอมพิวเตอร์ บัณฑิตวิทยาลัย จุฬาลงกรณ์มหาวิทยาลัย, 2540.
5. Jassadapakorn, C. and Chongstitvatana, P. Reduction of Computational Effort in Genetic Programming by Subroutines. Proceedings of National Computer Science and Engineering Conference, Bangkok, Thailand, 1998.
6. จุมพล พลวิชัย. การเรียนรู้ของแขนหุ่นยนต์โดยการใช้การโปรแกรมพันธุกรรม. วิทยานิพนธ์ปริญญาโทมหาบัณฑิต ภาควิชาวิศวกรรมคอมพิวเตอร์ บัณฑิตวิทยาลัย จุฬาลงกรณ์มหาวิทยาลัย, 2539.
7. Polvichai, J. and Chongstitvatana, P. Visually-guided reaching by genetic programming. Proceedings of 2nd Asian Conference on Computer Vision. Singapore, 1995.
8. Chongstitvatana, P. and Polvichai, J. Learning a visual task by genetic programming. Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, Osaka, Japan, 1996.
9. Rodkaew, Y. and Chongstitvatana, P. A Program for Visualizing Data in the Evolution of Solutions from Genetic Programming. Proceedings of National Computer Science and Engineering Conference, Bangkok, Thailand, 1999.
10. Rodkaew, Y. and Chongstitvatana, P. How to reduce the memory requirement in genetic programming, Proceedings of 4th Annual National Symposium on Computational Science and Engineering, Bangkok, Thailand, 2000.

11. Gale, T. and Main, I. GTK v1.2 Tutorial. [cited 21 February 1999] Available from <http://www.gtk.org>; INTERNET.
12. Paul. B. MESA. [cited 15 November 1999] Available from <http://www.mesa3d.org>; INTERNET.
13. Luke, S. and Spector, L. A Comparison of Crossover and Mutation in Genetic Programming. Proceedings of the Second Annual Genetic Programming Conference, pp. 240-248, 1997.
14. Luke, S. and Spector, L. A Revised Comparison of Crossover and Mutation in Genetic Programming. Proceedings of the Third Annual Genetic Programming Conference, pp. 208-213, 1998.
15. ปีเตอร์ นอร์ดัน และ จอห์น โซซา. เทคนิคการเขียนโปรแกรมภาษาแอสเซมบลี. พิมพ์ครั้งที่ 2. เรียบเรียงโดย หัสรังษี ตีริวิมลวรรณ, อภิชาติ พัฒนไพโรจน์, อาทิตย์ จิตต์จุพานนท์ และ วีรศักดิ์ วิทวัสกุล. กรุงเทพมหานคร: สำนักพิมพ์ ซีเอ็ดยูเคชั่น, 2532. หน้า 255.

บรรณานุกรม

1. Goldberg, D.E. Genetic Algorithms in Search, Optimization, and Machine Learning. Addison Wesley. 1989.
2. Richard, S., Wright, Jr. and Sweet, S. OpenGL Superbible. Waite Group Press. 1996.

ภาคผนวก ก

รายละเอียดเกี่ยวกับโปรแกรม

ก.1 แป้นลัด

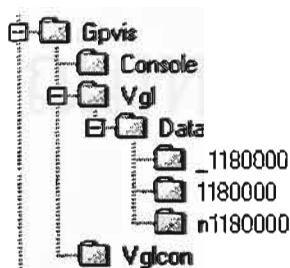
แป้นลัด	ความหมาย
การเลือกคำตอบ (Select mode) หัวข้อ 5.4	
q/w	เพิ่มหรือลดหมายเลขของคำตอบ
e/r	เพิ่มหรือลดหมายเลขรุ่นของคำตอบ
a	กำหนดการแสดงผลเป็นแบบ Hierarchy หรือยกเลิก
Arrow key	หมุนมุมมอง
[/]	เพิ่มหรือลดตำแหน่งของการมอง
t	ปรับการแสดงผลไปยัง Hierarchy mode
ESC	ออกจากโปรแกรม
การแสดงโครงสร้างคำตอบ (Individual mode) หัวข้อ 5.5	
+ -	ปรับอัตราการย่อขยาย
Arrow key	เลื่อนตำแหน่งภาพ
ESC	ออกจากส่วนนี้
ความสัมพันธ์ของคำตอบ (Hierarchy mode) หัวข้อ 5.7	
< >	เลือกเส้นทางของความสัมพันธ์
Page up	เลือกรุ่นคำตอบเพิ่มขึ้น
Page down	เลือกรุ่นคำตอบลดลง
h	ไปยังตำแหน่งกึ่งกลาง
+ -	ปรับอัตราการย่อขยาย
Arrow Key	เลื่อนตำแหน่งภาพ
i	เลิกการแสดงผลในแบบทรงกลมหรือแบบธรรมดา
O	ปรับการแสดงผลไปยัง Simulation mode

V	ปรับการแสดงผลไปยัง Individual mode
ESC	ออกจากส่วนนี้
การจำลองการเคลื่อนไหวแขนหุ่นยนต์ (Simulation mode) หัวข้อ 5.6	
+ -	ปรับอัตราการย่อขยาย
Arrow key	เลื่อนตำแหน่งภาพ
ESC	ออกจากส่วนนี้
ทั่วไป (All mode)	
*	ปรับปรุงหน้าจอ
F11	สลับสีพื้นระหว่างสีขาวกับสีดำสำหรับใช้ในงานพิมพ์
F12	จัดเก็บภาพภายในโปรแกรมเป็นแฟ้มที่มีนามสกุล .tif ในสารบบ snap/

ก.2 การติดตั้งระบบ GPVIS จากแผ่น CD-ROM

ใช้คำสั่งคัดลอกสารบบ GPVIS จากแผ่น CD-ROM ลงสู่ระบบ Linux และเปลี่ยนเป็น สารบบ GPVIS/VGLCON โดยใช้คำสั่ง `cd gpvis/vglcon` และเรียกใช้โปรแกรม VGLCON ได้ทันที หากมีความจำเป็นต้อง Compile ใหม่ ให้เรียกคำสั่ง `compile` ที่ สารบบ VGL, VGLCON และ CONSOLE ในการ Compile ใหม่มีความจำเป็นต้องใช้ Library X11, libcc5, GTK 1.2 และ MESA ด้วย ซึ่ง X11, libcc5 และ GTK โดยมากจะมีอยู่ในระบบปฏิบัติการแล้ว ส่วน MESA สามารถ Download ได้จาก <http://www.mesa3d.org>

ก.3 ระบบเพิ่มข้อมูลในระบบ GPVIS



รูปที่ 63 สารบบ GPVIS

ระบบแฟ้มจะแบ่งเป็นส่วนตามสารบบ (Directory) ดังรูปที่ 63 ซึ่งมีรายละเอียดดังนี้

- สารบบ GPVIS/VGL จัดเก็บไฟล์ โปรแกรม VGL
- สารบบ GPVIS/VGL/DATA จัดเก็บไฟล์ ข้อมูลสำหรับการจินตทัศน์ด้วยโปรแกรม VGL โดยแบ่งข้อมูลเป็นส่วนๆ จัดเก็บในสารบบตามสภาพแวดล้อม
- สารบบ GPVIS/VGLCON จัดเก็บไฟล์ โปรแกรม VGLCON
- สารบบ GPVIS/CONSOLE จัดเก็บไฟล์ โปรแกรม CONSOLE

ภาคผนวก ข

วิธีการเพิ่มฟังก์ชันภายในระบบ GPVIS

ในการใช้งานระบบ GPVIS ในส่วนของการเลือกฟังก์ชันเฉลี่ยข้อมูล หากผู้ใช้งานต้องการฟังก์ชันอื่นๆ แทนการเฉลี่ยข้อมูล ผู้ใช้งานสามารถเพิ่มฟังก์ชันใหม่ในส่วนดังกล่าวได้เอง โดยการเขียนโปรแกรมเพิ่มในส่วนที่ได้จัดเตรียมไว้ และคอมไพล์ไฟล์รวมกับระบบใหม่ ภายหลังจากการคอมไพล์แล้ว ผู้ใช้งานจะสามารถเลือกใช้ฟังก์ชันใหม่ได้ตามความต้องการ

ในส่วนแรกจะแสดงข้อมูลที่เกี่ยวข้องกับการสร้างภาพ 3 มิติภายในระบบ GPVIS เพื่อทำความเข้าใจกับข้อมูลก่อนที่จะแสดงตัวอย่างในการแก้ไขข้อมูลและวิธีเพิ่มฟังก์ชันต่อไป

ข.1 ข้อมูลสำหรับใช้ในการสร้างภาพ 3 มิติ

เมื่อผู้ใช้เลือกแกน X,Y,Z กับแกนข้อมูลที่ต้องการแล้ว ระบบจะอ่านข้อมูลจากสภาพแวดล้อมและรอบที่ได้เลือกไว้ และทำการกรองข้อมูลบางส่วนทิ้งไปตามเงื่อนไข จากนั้นคำนวณ ข้อมูลสำหรับการปรับค่าอัตราส่วนของแกนและการแสดงแกน และจัดเรียงข้อมูลใหม่ จากนั้นถ้าผู้ใช้งานเลือกฟังก์ชันเฉลี่ยข้อมูลไว้ ระบบจะเรียกฟังก์ชันเฉลี่ยข้อมูล เมื่อฟังก์ชันเฉลี่ยข้อมูลทำงานเสร็จแล้วส่งข้อมูลกลับมา ข้อมูลจะถูกปรับปรุงอีกครั้งหนึ่งตามค่าอัตราส่วนของแกนที่ได้คำนวณไว้ตอนแรก และนำข้อมูลมาสร้างเป็นภาพ 3 มิติ

ในระบบได้นิยามตัวแปรสำหรับเก็บข้อมูลเพื่อแสดงผลชื่อ surfacedata ไว้ดังนี้

```
#define MAXsurfacepoint 24000
#define _GLfloat float
#define _GLubyte unsigned char

typedef struct {
    int use;
    _GLfloat x,y,z;
    _GLfloat sx,sy,sz;
    _GLfloat tx,ty,tz;
```

```

    _GLubyte r,g,b;
    _GLubyte discount;
    int value;
} surfacedatastruct ;

surfacedatastruct surfacedata[MAXsurfacepoint];

```

ตัวแปร surfacedata เป็นตัวแปรแบบแถวลำดับขนาด 24,000 หน่วย ของโครงสร้างชื่อ surfacedatastruct ข้อมูล 1 หน่วยแทนข้อมูลสำหรับการสร้างภาพ 1 จุด

อาจมีข้อสงสัยว่าทำไมจึงได้กำหนดไว้ 24,000 หน่วย เพราะหากดูจำนวนรุ่นกับจำนวนคำตอบทั้งหมดในหนึ่งการทดลองจะพบว่า มีจำนวนรุ่นมากที่สุด 10 รุ่นและมีจำนวนคำตอบในแต่ละรุ่นมากที่สุด 400 ตัว เมื่อกำหนดแล้วควรจะใช้เพียง 4,000 หน่วย สาเหตุที่กำหนดไว้ 24,000 หน่วยนั้นเนื่องมาจากการแสดงข้อมูลบางอย่าง 1 คำตอบใช้เนื้อที่มากกว่า 1 หน่วย

รายละเอียดของโครงสร้าง surfacedatastruct มีดังนี้

ตาราง 12 ตัวแปรสำหรับเก็บข้อมูลและความหมาย

ตัวแปร	ความหมาย
use	แสดงการใช้งาน
x,y,z	ตำแหน่ง
sx,sy,sz	ตำแหน่งเคลื่อนย้ายสัมพัทธ์
tx,ty,tz	ขนาด
r,g,b	สี
discount	ความต่อเนื่องในการลากเส้น
value	สงวนไว้

ตัวแปร use เมื่อมีค่าเป็น 1 แสดงว่าข้อมูลนี้เป็นข้อมูลใช้งาน และเมื่อมีค่าเป็น 0 ข้อมูลนี้ไม่นำไปใช้งาน

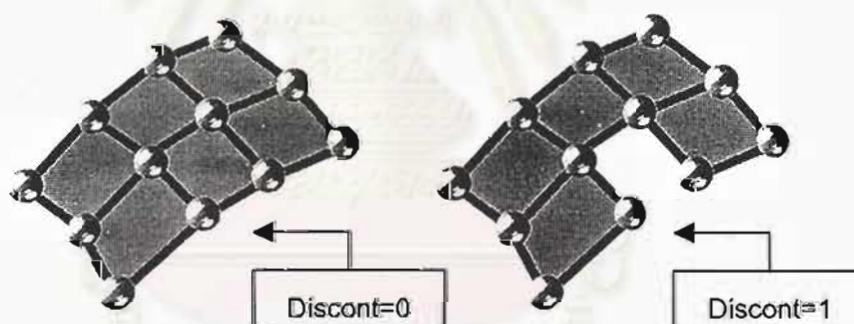
ตำแหน่งเคลื่อนย้ายสัมพัทธ์ มีความสัมพันธ์กับตำแหน่ง x,y,z ในกรณีทั่วไปข้อมูลตำแหน่งเคลื่อนย้ายสัมพัทธ์มีค่าเท่ากับ 0 หากผู้ใช้งานต้องการเขียนฟังก์ชันเพิ่มเติมเพื่อให้ข้อมูลมีการย้ายตำแหน่งไปจะเปลี่ยนค่าตัวแปรตำแหน่งโดยตรง หรือเพิ่มค่าตัวแปรตำแหน่งเคลื่อนย้ายสัมพัทธ์แทนก็ได้ เช่นต้องการให้ข้อมูล ณ ตำแหน่ง n เคลื่อนไป 3 หน่วย

ในแนวแกน X ก็ปรับค่า `surfacedata[n].x += 3`; หรือ `surfacedata[n].sx = 3`; ก็ได้ผลเช่นเดียวกัน แต่การใช้ตำแหน่งเคลื่อนย้ายสัมพัทธ์จะคงสภาพของข้อมูลเดิมไว้ดีกว่าการเปลี่ยนแปลงข้อมูลโดยตรง

ตัวแปรขนาดแสดงถึงขนาดของจุดข้อมูล จะถูกนำไปใช้เมื่อกำหนดในการแสดงผลในการจินตทัศน์เป็นแบบจุด โดยปรกติระบบจะกำหนดขนาดเป็น 5 เท่ากันทุกจุด หากต้องการเขียนฟังก์ชันเพื่อเน้นข้อมูลใดเป็นพิเศษอาจเพิ่มขนาดให้มากขึ้น และลดขนาดของข้อมูลที่ไม่สำคัญ

ตัวแปรสี `r,g,b` แทนค่าสีของข้อมูลคือ แดง, เขียว และน้ำเงินตามลำดับ มีค่าตั้งแต่ 0 ถึง 255

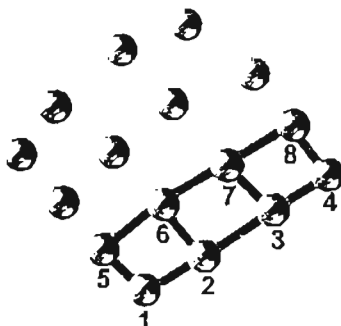
ตัวแปร `discont` เป็นตัวแปรแสดงความต่อเนื่องในการลากเส้นเชื่อมระหว่างจุดถัดไปในการแสดงผลแบบ `line`, แบบ `mesh` และแบบ `surface` หากเป็น 0 มีการลากเส้นเชื่อม หากเป็น 1 ไม่มีการลากเส้นเชื่อมกับจุดถัดไป ดังภาพ



รูปที่ 64 ลักษณะของเส้นเชื่อมระหว่างจุด

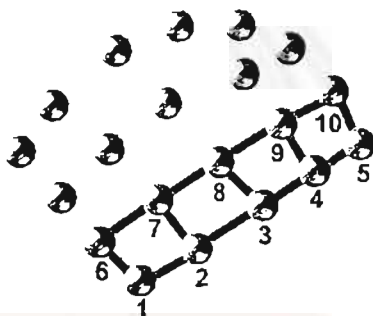
ข.2 การจัดวางข้อมูล

ในการสร้างเส้นเชื่อมระหว่างจุด โปรแกรมจำเป็นต้องทราบขนาดความยาวของด้านเพื่อเป็นตัวแบ่งข้อมูลแถวลำดับออกเป็น 2 ด้าน ตัวแปร `gensurfacedata_sidecount` ซึ่งเป็น `integer` เป็นตัวบอกขนาดดังกล่าว เพื่อใช้ในการคำนวณการเชื่อมจุด



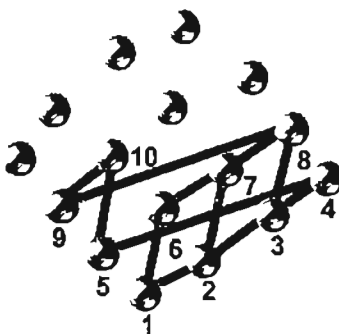
รูปที่ 65 การจัดวางข้อมูลที่มีด้านยาว 4 จุด

จากภาพ ข้อมูลจัดวางโดยมีด้านยาว 4 จุด ให้ `gensurfacedata_sidecount = 4` จุด
ที่ 1 ควรเชื่อมกับจุดที่ 2 และ 5 เพื่อให้เกิด surface ที่เกิดจากจุด 1, 2, 5, 6



รูปที่ 66 การจัดวางข้อมูลที่มีด้านยาว 5 จุด

จากภาพ ข้อมูลจัดวางโดยมีด้านยาว 5 จุด ให้ `gensurfacedata_sidecount = 5` จุด
ที่ 1 ควรเชื่อมกับจุดที่ 2 และ 6 เพื่อให้เกิด surface ที่เกิดจากจุด 1, 2, 6, 7



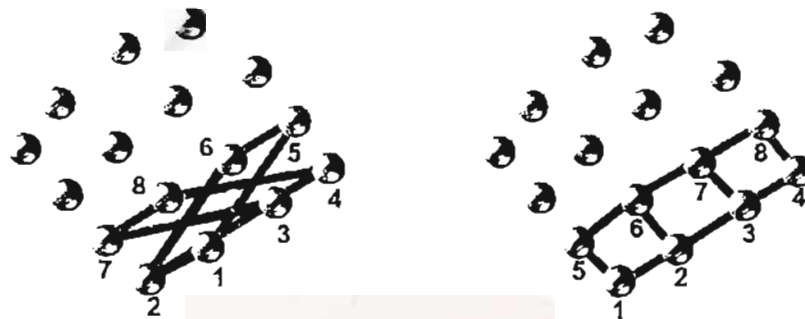
รูปที่ 67 การจัดวางข้อมูลที่มีด้านยาว 4 จุดซึ่งไม่สัมพันธ์กับตัวแปร

จากภาพ ข้อมูลจัดวางโดยมีด้านยาว 4 จุด แต่ค่า gensurfacedata_sidecount = 5 ซึ่งไม่สัมพันธ์กับตำแหน่งของข้อมูลทำให้ จุดที่ 1 ที่ควรเชื่อมกับจุดที่ 2 และ 6 เพื่อให้เกิด surface ที่เกิดจากจุด 1, 2, 6, 7 ขยับเลื่อนไป และส่งผลให้เกิดความสับสนกับ surface ที่เกิดจากจุด 4, 5, 9, 10 ในตอนท้าย

ดังนั้นตัวแปร gensurfacedata_sidecount จำเป็นต้องมีค่าสอดคล้องกับข้อมูลที่ถูกจัดวางไว้

ข.3 การจัดเรียงข้อมูล

เมื่อโปรแกรมอ่านข้อมูลจากสภาพแวดล้อมและรอบที่ได้เลือกไว้ และทำการกรองข้อมูลบางส่วนทิ้งไปตามเงื่อนไข โปรแกรมจำเป็นต้องจัดเรียงลำดับข้อมูลใหม่ ตามแกนข้อมูลที่กำหนดเพื่อให้สามารถสร้างเส้นเชื่อมระหว่างข้อมูล ได้อย่างถูกต้อง



รูปที่ 68 การจัดเรียงข้อมูล

จากภาพแสดงข้อมูลที่อยู่บนแกนที่ไม่ได้จัดเรียงลำดับ(ภาพซ้าย) และข้อมูลที่เรียงลำดับตามแกนข้อมูลแล้ว(ภาพขวา) การสร้างเส้นเชื่อมในภาพทางด้านซ้ายไม่ถูกต้องเนื่องจากลำดับข้อมูลไม่สัมพันธ์กับตำแหน่งของข้อมูลบนแกน ตัวอย่างเช่น ข้อมูลลำดับที่ 1 มีค่า 20 ข้อมูลที่ 2 มีค่า 10 ข้อมูลที่ 3 มีค่า 30 ข้อมูลที่ 4 มีค่า 40 เมื่อนำข้อมูลมาใช้ ข้อมูลที่ 2 จะมาก่อนข้อมูลที่ 1 และข้อมูลในแถวถัดไปคือ 7, 8, 6, 5 เส้นเชื่อมที่สร้างจากลำดับดังกล่าวจะไม่ถูกต้อง ดังนั้นจึงต้องจัดเรียงลำดับข้อมูลใหม่ตามค่าของข้อมูลในแถวแรกคือ สลับข้อมูลที่ 1 กับ 2 และต้องจัดเรียงลำดับแถวอื่นๆ ด้วย

ข.4 ปัญหาเกี่ยวกับการสร้างเส้นเชื่อมข้อมูล

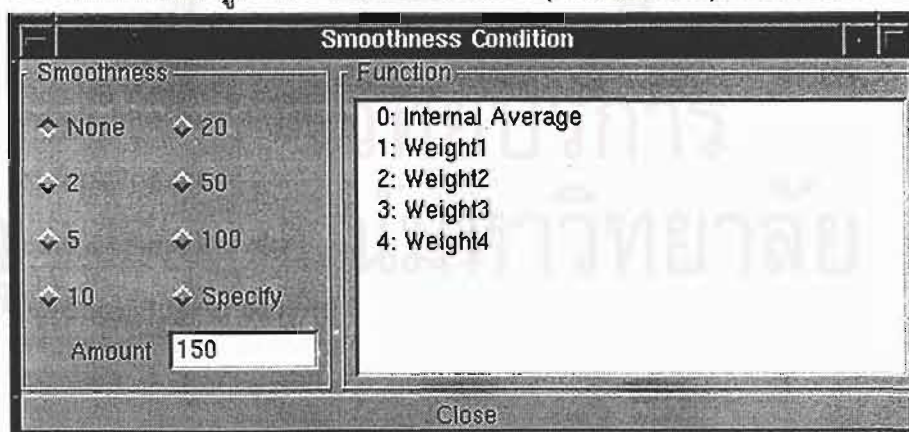
การสร้างเส้นเชื่อมข้อมูลนั้น ไม่สามารถทำกับแกนใดๆ ได้ทุกแกน เนื่องจากปัญหาทางด้านการจัดเรียงของลำดับข้อมูล หากแกน X ไม่ได้เป็นแกนที่แสดงรุ่น ข้อมูลจะไม่สามารถจัดเรียงได้อย่างถูกต้อง ทำให้การสร้างเส้นเชื่อมไม่ถูกต้องและภาพที่ได้ซ้อนทับกัน แต่ข้อมูลที่แสดงยังมีความถูกต้องซึ่งผู้ใช้งานควรเปลี่ยนไปใช้การแสดงผลแบบจุดแทน

ข้อมูลต้องมีขนาดความยาวของด้านเท่ากันทุกๆ แก้ว เป็นปัญหาอีกอย่างหนึ่งในการสร้างเส้นเชื่อมข้อมูล หากมีด้านยาวไม่เท่ากันในทุกแก้วจะพบว่าในบริเวณส่วนปลายแก้วไม่สามารถสร้างเส้นเชื่อมได้เนื่องจากบางแก้วยังเหลือข้อมูลอยู่แต่แก้วถัดไปไม่มีข้อมูลแล้ว การกำหนดให้ใช้ตัวแปร `gensurfacedata_sidecount` แทนความยาวของทุกๆ แก้ว เป็นการควบคุมไม่ให้เกิดปัญหาดังกล่าว

ข.5 การเฉลี่ยข้อมูล

ข้อมูลที่ถูกนำมาใช้ในการเฉลี่ยข้อมูล คือข้อมูลที่ผ่านการจัดเรียงข้อมูลตามแกนแล้ว โดยมีตัวแปร `gensurfacedata_sidecount` เป็นตัวบอกขนาดความยาวของด้าน และตัวแปร `nsurfacedata` เพื่อบอกว่าข้อมูลที่ส่งมาให้ภายใน `surfacedata[.]` มีจำนวนเท่าใด

ฟังก์ชันเฉลี่ยภายในระบบจากภาพ คือฟังก์ชันที่ 0: Internal Average เป็นฟังก์ชันที่ออกแบบให้เฉลี่ยค่าข้อมูลตามจำนวนของค่าเฉลี่ย (Smoothness) ที่กำหนดไว้



รูปที่ 69 กำหนดค่าเฉลี่ยข้อมูล

ในการทำงานของฟังก์ชันนี้ เมื่อผู้ใช้งานเลือกค่าเฉลี่ยเป็น 5 โปรแกรมจะให้ค่า smoothness = 5 กับฟังก์ชัน

ฟังก์ชันจะอ่านข้อมูลจาก surfacedata[...] ทั้งหมด เพื่อทำการเฉลี่ย โดยมีตัวแปร iscount เป็นตัวนับว่าครบตามค่า smoothness หรือไม่ หากยังไม่ครบก็จะทำการบวกข้อมูลใหม่เพิ่มเข้าไปที่ตัวแปรสะสมในที่นี้คือ x,y,z,r,g,b หากครบโปรแกรมจะหารตัวแปรสะสมด้วยค่า iscount ซึ่งมีค่าเท่ากับ smoothness ค่าที่ได้จะเป็นค่าเฉลี่ยของตัวแปรแต่ละตัว และนำกลับไปเก็บไว้ที่ตำแหน่งสุดท้าย ในกรณีนี้คือ 5 เนื่องจากการนำข้อมูลมาเฉลี่ย 5 ตัว ข้อมูลที่ 1 ถึง 5 เมื่อผ่านการเก็บข้อมูลเพื่อการเฉลี่ยแล้วจะไม่นำมาใช้อีก โปรแกรมก็จะให้ค่า surfacedata[i].use=0; เพื่อบอกกับตัวจัดการข้อมูลภายหลังว่าข้อมูลนี้ไม่ใช่ หลังจากนั้นนำข้อมูลเฉลี่ยเก็บไว้ที่ข้อมูลที่ 5 แล้วแก้ไขค่าของ surfacedata[i].use=1; เพื่อบอกว่าใช้ข้อมูลนี้ และเริ่มต้นนับใหม่ให้ iscount=0; และให้ตัวแปรสะสมทุกตัวมีค่าเป็น 0 จากนั้นโปรแกรมจะนำข้อมูลที่ 6 ถึง 10 มาเฉลี่ยและเก็บไว้ในข้อมูลที่ 10 เป็นเช่นนี้เรื่อย ๆ จนหมดข้อมูล ซึ่งก็คือเงื่อนไข $i < n_{surfacedata}$ เป็นจริง

```
void fnavg(void)
{
    int i;
    int x,y,z;
    int r,g,b;
    int iscount;

    iscount=0; x=y=z=0; r=g=b=0;

    for ( i=0; i< nsurfacedata; i++)
    {
        iscount++;

        x+=surfacedata[i].x;    y+=surfacedata[i].y; z+=surfacedata[i].z;
        r+=surfacedata[i].r;    g+=surfacedata[i].g; b+=surfacedata[i].b;

        surfacedata[i].use=0;

        if (iscount==Smoothness)
        {
            surfacedata[i].x=x/iscount;
            surfacedata[i].y=y/iscount;
            surfacedata[i].z=z/iscount;

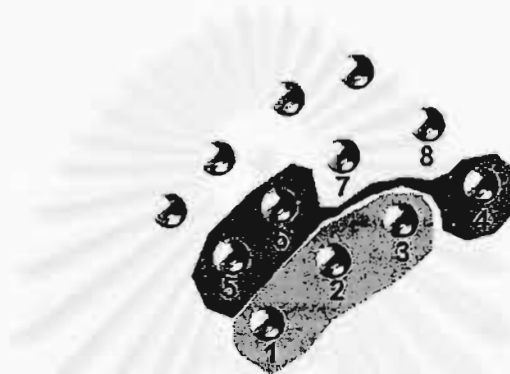
            surfacedata[i].r=r/iscount;
            surfacedata[i].g=g/iscount;
            surfacedata[i].b=b/iscount;
        }
    }
}
```

```

surfacedata[i].use=1;
iscount=0; x=y=z=0; r=g=b=0;
}
}

```

ในการเฉลี่ยข้อมูลข้างต้นไม่ได้คำนึงถึงความยาวของด้าน ซึ่งจะทำให้เกิดความผิดพลาดถ้าความยาวด้านมีจำนวนที่หารแล้วไม่ลงตัวกับค่าเฉลี่ยข้อมูล ดังภาพ



รูปที่ 70 การเฉลี่ยที่ไม่เหมาะสม

ข้อมูลจัดวางโดยมีด้านยาว 4 จุด และมีค่า gensurfacedata_sidecount = 4 ต้องการเฉลี่ยด้วยค่า Smoothness = 3 ข้อมูลกลุ่มแรกคือ 1-3 เฉลี่ยได้ ข้อมูลกลุ่มต่อมาคือ 4-6 ค่าเฉลี่ยที่ได้จะเกิดข้อผิดพลาด เราจึงต้องแก้ไขโปรแกรมข้างต้นใหม่โดยเพิ่มส่วนจัดการกับขอบของข้อมูล เพื่อให้กลุ่มของการเฉลี่ยเป็นไปดังภาพ



รูปที่ 71 การเฉลี่ยที่กำหนดขอบตามความยาวด้าน

ในส่วนของโปรแกรมได้เพิ่มตัวแปร rowsmooth เพื่อเป็นตัวนับว่าข้อมูลเริ่มจากแถวใด ตัวแปร minentry คือตัวแปรบอกข้อมูลตัวแรกของแถว (จากภาพตัวอย่างนี้คือ 1)

และตัวแปร maxentry คือตัวแปรบอกข้อมูลตัวสุดท้ายของแถว (จากภาพตัวอย่างนี้คือ 4) ได้เปลี่ยน for loop ของ i จากเดิม 0 ถึง nsurfacedata - 1 เป็น minentry ถึง maxentry และให้ตัวแปร rowsmooth เป็นตัวนับแทน จากภาพตัวอย่าง รอบแรก minentry มีค่า 1, maxentry มีค่า 4 รอบต่อมา minentry มีค่า 5, maxentry มีค่า 8 รอบต่อมา minentry มีค่า 9, maxentry มีค่า 12 เป็นเช่นนี้ไปจนหมดข้อมูล

```
void fnavg2(void)
{
    int rowsmooth;
    int minentry,maxentry;

    int i;
    int iscount,x,y,z,r,g,b;
    int count_sidecount;

    for (rowsmooth=0;
        rowsmooth<nsurfacedata;
        rowsmooth+=gensurfacedata_sidecount)
    {
        minentry = rowsmooth;
        maxentry = rowsmooth+gensurfacedata_sidecount-1;

        iscount=0; x=y=z=0; r=g=b=0;
        count_sidecount=0;

        for (i=minentry; i<=maxentry; i++)
        {

            iscount++;

            x+=surfacedata[i].x;
            y+=surfacedata[i].y;
            z+=surfacedata[i].z;

            r+=surfacedata[i].r;
            g+=surfacedata[i].g;
            b+=surfacedata[i].b;

            surfacedata[i].use=0;

            if (iscount==Smoothness)
            {

                surfacedata[i].x=x/iscount;
                surfacedata[i].y=y/iscount;
                surfacedata[i].z=z/iscount;

                surfacedata[i].r=r/iscount;
                surfacedata[i].g=g/iscount;
                surfacedata[i].b=b/iscount;
            }
        }
    }
}
```

```

        surfacedata[i].use=1;

        iscount=0; x=y=z=0; r=g=b=0;
        count_sidecount++;
    }

}

if (iscount>0)
{
    i=maxentry;

    surfacedata[i].x=x/iscount;
    surfacedata[i].y=y/iscount;
    surfacedata[i].z=z/iscount;

    surfacedata[i].r=r/iscount;
    surfacedata[i].g=g/iscount;
    surfacedata[i].b=b/iscount;

    surfacedata[i].use=1;

    count_sidecount++;
}
}

gensurfacedata_sidecount = count_sidecount;
}

```

เมื่อเฉลี่ยถึงขอบข้อมูลอาจจะเหลือเศษ ส่วนที่เพิ่มเติมในตอนท้าย if (iscount>0) เป็นส่วนตรวจสอบว่าเหลือเศษหรือไม่ หากเหลือเศษเหลือเท่าใดก็ตาม ภายในเงื่อนไขนี้ก็จะนำไปเฉลี่ยเท่าจำนวนที่เหลือ

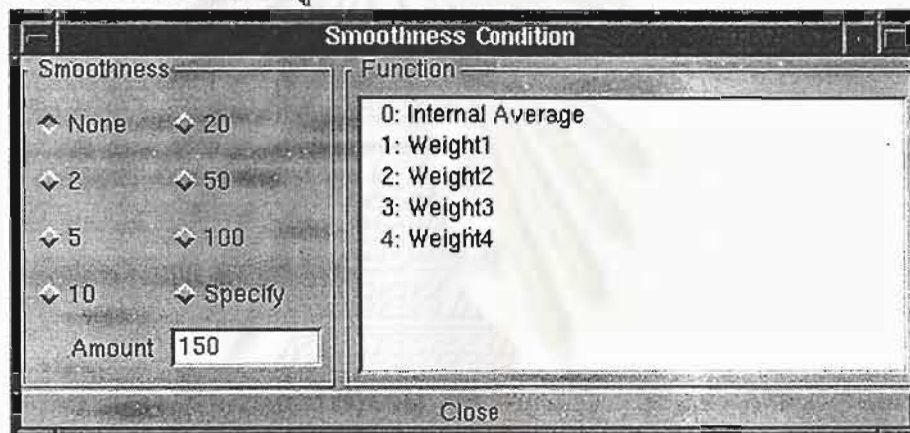
ส่วนที่มีความสำคัญอีกส่วนหนึ่งที่เพิ่มเข้ามาในฟังก์ชันนี้ก็คือตัวแปร count_sidecount เป็นตัวแปรสำหรับนับความยาวของด้านใหม่หลังจากการเฉลี่ย และนำค่าที่ได้ไปแก้ไขตัวแปร gensurfacedata_sidecount ส่วนนี้มีความจำเป็นอย่างมากที่จะต้องปรับปรุงค่าของตัวแปรใหม่ เนื่องจากในการเฉลี่ยนี้ เราได้ลดจำนวนของข้อมูลลง โดยการบอกว่าข้อมูลบางตัวไม่ใช่ ด้วยคำสั่ง surfacedata[i].use=0; ถ้าไม่แก้ไขตัวแปร gensurfacedata_sidecount ก็จะเกิดความผิดพลาดในการสร้างเส้นเชื่อมระหว่างข้อมูล

ข.6 การเพิ่มฟังก์ชัน

ไฟล์ `gpvisuser.h` และ `gpvisuser.c` เป็นไฟล์สำหรับผู้ใช้งานโดยมีฟังก์ชันที่ติดต่อกับระบบ 2 ฟังก์ชันคือ `GPVIS_GetUserFuncDesc` และ `GPVIS_ActUserFunc`

```
char *GPVIS_GetUserFuncDesc(int GPVIS_FuncNo) ;
int GPVIS_ActUserFunc(Int GPVIS_FuncNo,int Smoothness) ;
```

ฟังก์ชัน `GPVIS_GetUserFuncDesc` เป็นฟังก์ชันสำหรับแสดงชื่อฟังก์ชันที่เพิ่มเติมให้กับผู้ใช้งานได้เลือกใช้ เมื่อระบบ `GPVIS` แสดงกล่องโต้ตอบ `Smoothness Condition` ชื่อฟังก์ชันจะแสดงในส่วนของ "Function" ทางด้านขวามือ มีหมายเลขของฟังก์ชันตามด้วยเครื่องหมาย : อยู่ก่อนชื่อฟังก์ชัน



รูปที่ 72 กำหนดค่าเฉลี่ยข้อมูล

เมื่อเข้าสู่ระบบ ในส่วนของการสร้างกล่องโต้ตอบระบบจะเรียกฟังก์ชัน `GPVIS_GetUserFuncDesc` เพื่อสอบถามว่าฟังก์ชันใดบ้าง ระบบจะถามโดยเริ่มตั้งแต่ `GPVIS_FuncNo` หมายเลขที่ 1 ว่าชื่ออะไร ระบบจะหยุดเรียก `GPVIS_GetUserFuncDesc` เมื่อฟังก์ชันคืนค่าเป็น "" หรือจนถึงลำดับที่ 100

ในการแก้ไขฟังก์ชันนี้ ผู้ใช้สามารถเพิ่ม case ได้มากที่สุด 100 case และมีข้อความระวางก็คือ ตัวแปร `desctxt` ต้องไม่เกิน 128 ตัวอักษร

```
char *GPVIS_GetUserFuncDesc(Int GPVIS_FuncNo)
{
    static char desctxt[128];

    switch (GPVIS_FuncNo) {
        case 1: strcpy(desctxt,"Weight1"); break;
```



```

    case 2: strcpy(desctxt,"Weight2"); break;
    case 3: strcpy(desctxt,"Weight3"); break;
    case 4: strcpy(desctxt,"Weight4"); break;
    default: strcpy(desctxt,""); break;
  }
  return desctxt;
}

```

จากนั้นเมื่อผู้ใช้งานเลือกการแสดงผลแบบ 3 มิติ ระบบ GPVIS จะมาเรียกใช้ฟังก์ชัน ผ่าน GPVIS_ActUserFunc ซึ่งรับพารามิเตอร์คือเลือกฟังก์ชันหมายเลขใด (GPVIS_FuncNo) และค่าเฉลี่ย (Smoothness)

```

int GPVIS_ActUserFunc(int GPVIS_FuncNo,int Smoothness)
{
  int result;
  switch (GPVIS_FuncNo) {
    case 1: result = FnWeigth1(Smoothness); break;
    case 2: result = FnWeigth2(Smoothness); break;
    case 3: result = FnWeigth3(Smoothness); break;
    case 4: result = FnWeigth4(Smoothness); break;
    default: result = 1 ; break; /* tell GPVIS, failed */
  }

  return result ; /* 0 - success, other - failed (with code) */
}

```

ตัวอย่างเช่น เมื่อผู้ใช้เลือกใช้ฟังก์ชันชื่อ Weight1 จากกล่องโต้ตอบ โปรแกรมในส่วนนี้จะไปเรียก FnWeight1 และฟังก์ชัน FnWeight1 ก็จะคำนวณค่าเฉลี่ยหรือแก้ไขข้อมูลตาม que ผู้ใช้งานได้เขียนเพิ่มเติมไว้ สำหรับตัวอย่างการเพิ่มฟังก์ชันดูได้จากโปรแกรมต้นฉบับ gpvisuser.c

ภาคผนวก ค

โครงสร้างข้อมูลสำหรับการจินตทัศน์ในระบบ GPVIS

ระบบ GPVIS ใช้ข้อมูลกำหนดการแข่งขันจากรูปโปรแกรมในงานวิจัยของชัยวัฒน์ เจริญภาพกรณ์ [4] โปรแกรมจัดเก็บข้อมูลในขณะที่โปรแกรมทำงาน ข้อมูลมี 3 ส่วนคือ ข้อมูลสนามทดลอง ข้อมูลเกี่ยวกับการทดลอง และข้อมูลคำตอบจากกำหนดการแข่งขัน

- env.txt เป็นไฟล์ที่เก็บข้อมูลของสนามทดลอง
- stat.txt เป็นไฟล์เก็บข้อมูลที่เกี่ยวข้องกับการทดลอง
- dataNNN.vgp เป็นไฟล์เก็บข้อมูลคำตอบกำหนดการแข่งขันในแต่ละรอบ NNN เป็นตัวเลขของรอบ เช่น data010.vgp เป็นข้อมูลของรอบที่ 10 เป็นต้น

ไฟล์ env.txt และ stat.txt มีรูปแบบข้อมูลตามงานวิจัยของชัยวัฒน์ เจริญภาพกรณ์ [4] ดังนั้นจึงไม่ขอกล่าวถึงในที่นี้

ค.1 โครงสร้างไฟล์ dataNNN.vgp

ภายในไฟล์ dataNNN.vgp ซึ่งเป็นไฟล์อักขระ (text file) เก็บข้อมูลกำหนดการแข่งขันแต่ละรุ่นตามลำดับ ตั้งแต่รุ่นที่ 1 ถึง 10 ตามตาราง 13

ตาราง 13 โครงสร้างข้อมูลภายในไฟล์ dataNNN.vgp

ไฟล์ dataNNN.vgp
กล่องข้อมูล* รุ่น (รุ่นที่ 1)
กล่องข้อมูลรุ่น (รุ่นที่ 2)
กล่องข้อมูลรุ่น (รุ่นที่ 3)
..
กล่องข้อมูลรุ่น (รุ่นที่ 9)
กล่องข้อมูลรุ่น (รุ่นที่ 10)

*
Data block

ค.2 โครงสร้างกล่องข้อมูลรุ่น

ในกล่องข้อมูลรุ่นเก็บข้อมูลกำหนดการแข่งขันรายการหมายเลข 1 จนถึงหมายเลข 400 ตาม

ตาราง 14 โดยมีส่วนแสดงรอบและรุ่นนำหน้า

ตาราง 14 โครงสร้างกล่องข้อมูลรุ่น

กล่องข้อมูลรุ่น
แสดงรอบและรุ่น (รุ่น 1)
กล่องข้อมูลกำหนดการแข่งขันรายการ (หมายเลข 1)
กล่องข้อมูลกำหนดการแข่งขันรายการ (หมายเลข 2)
กล่องข้อมูลกำหนดการแข่งขันรายการ (หมายเลข 3)
..
กล่องข้อมูลกำหนดการแข่งขันรายการ (หมายเลข 400)

ค.3 โครงสร้างกล่องข้อมูลกำหนดการแข่งขันรายการ

เป็นส่วนเก็บข้อมูลกำหนดการแข่งขันรายการ 1 คำตอบแบ่งออกเป็น 2 ส่วนคือส่วนคุณสมบัติและข้อมูลโปรแกรม

ตาราง 15 โครงสร้างกล่องข้อมูลกำหนดการแข่งขันรายการ

กล่องข้อมูลกำหนดการแข่งขันรายการ
คุณสมบัติ
ข้อมูลโปรแกรม

ค.4 โครงสร้างของข้อมูลรอบและรุ่น

ตาราง 16 โครงสร้างของข้อมูลรอบและรุ่น

ระเบียบ	ความหมาย
Run	หมายเลขรอบ
Gen	หมายเลขรุ่น

ตัวอย่าง

run 1 gen 1

ค.5 โครงสร้างของข้อมูลคุณสมบัติ

ตาราง 17 โครงสร้างของข้อมูลคุณสมบัติ

ระเบียบ	ความหมาย
Id	Individual (หมายเลขคำตอบ)*
Sc	Success (ความสำเร็จ) 0 – ปลายแขนไม่ถึงเป้าหมาย 1 – ปลายแขนถึงเป้าหมาย
Ft	Fitness (ค่าความเหมาะสม)
Pl	Proc Len (ความยาวของโปรแกรม)
Ti	Time Use (เวลาที่ใช้ในการเคลื่อนที่)
Id	Initial Distance (ระยะห่างระหว่างปลายแขนหุ่นยนต์กับเป้าหมายเริ่มต้น)
Ld	Last Distance (ระยะห่างระหว่างปลายแขนหุ่นยนต์กับเป้าหมายเมื่อสิ้นสุดการทำงาน)
Sd	Sum Distance (ระยะทางการเคลื่อนที่ของปลายแขนหุ่นยนต์โดยรวม)
Se	See (สิ่งกีดขวางระหว่างปลายแขนกับเป้าหมาย) 0 – มีสิ่งกีดขวาง 1 – ไม่มีสิ่งกีดขวาง
Di	Die (แขนหุ่นยนต์ชำรุด) 0 – ไม่ชำรุด 1 – ชำรุด
P1	Parent Number 1 (หมายเลขของพ่อแม่ตัวที่ 1)
P2	Parent Number 2 (หมายเลขของพ่อแม่ตัวที่ 2)
Ev	Evolution Method (กรรมวิธีวิวัฒนาการของคำตอบ)
Ad	Number of ADF (จำนวน ADF – ไม่ใช่ในงานวิจัยนี้)
Al1	Alter-Point Parent1 (ตำแหน่งเปลี่ยนแปลงภายในโครงสร้างของพ่อแม่ตัวที่ 1)
Al2	Alter-Point Parent2 (ตำแหน่งเปลี่ยนแปลงภายในโครงสร้างของพ่อแม่ตัวที่ 2)

ตัวอย่าง

```
id 0 sc 0 ft -7192 pl 82 ti 45 id 14216 ld 22213 sd 9531 se 1 di 1 p1 0 p2 0 ev 0 ad 0 al1 0 al2 0
```

* หมายเลขคำตอบในไฟล์เริ่มตั้งแต่หมายเลข 0 ถึงหมายเลข 399 ส่วนในโปรแกรมเริ่มตั้งแต่หมายเลข 1 ถึงหมายเลข 400 หมายเลขคำตอบในไฟล์คือหมายเลขคำตอบในโปรแกรมลบด้วย 1

ก.6 โครงสร้างของข้อมูลโปรแกรม

ตาราง 18 โครงสร้างของข้อมูลโครงสร้างบัพ

ระเบียบ	ความหมาย
MAIN:ln	Proc Len (ความยาวของโปรแกรม)
[String]	ตัวโปรแกรมกำหนดการเชิงพันธุกรรม

ตัวอย่าง

```
MAIN:ln 82 pdmoefcchoenempdiioedgdeloeoeepdoenejiimpdijmmeoekkfmoeg  
gilloehpdifdpdenefloeldijkoefjncimcpdmmmdg
```

ก.7 ตัวอย่างข้อมูลจากไฟล์

ตาราง 19 ตัวอย่างข้อมูลภายในไฟล์ *data001.vgp*

```
Run 1 gen 1
id 0 sc 0 ft -7192 pl 82 ti 45 id 14216 ld 22213 sd 9531 se 1 di 1 p1 0 ..
MAIN:ln 82 pdmoefcchoenempdiioedgdeloeoeepdoenejiimpdijmmeoekkfmoegi ..
id 1 sc 0 ft -7723 pl 82 ti 89 id 14216 ld 26090 sd 7592 se 1 di 1 p1 0 ..
MAIN:ln 82 oecpdlnedioemndneffkklhneoejkeioehkoekjllghnenellpdpdiicfpd ..
id 2 sc 0 ft -7000 pl 80 ti 11 id 14216 ld 14216 sd 0 se 0 di 1 p1 0 p2 ..
MAIN:ln 80 nefnemjgipdnenedcdclkoefiikoecpdjkjgoecdlldcoenelldepdmnef ..
id 3 sc 0 ft -3336 pl 83 ti 100 id 14216 ld 16010 sd 11988 se 0 di 0 p1 ..
MAIN:ln 83 oepdkjdneioeioekoemikgfmioeicpdjkjcenecnedecmgmoenepdhhdhpdml ..
..
..
id 398 sc 0 ft -1646 pl 81 ti 100 id 14216 ld 3492 sd 22000 se 0 di 0 p1 ..
MAIN:ln 81 nepdoejjekjgcnepdgehlpdhfpdmoeppedcgfhleeoenepdjgoeklldoecglcc ..
id 399 sc 0 ft -5230 pl 83 ti 15 id 14216 ld 1602 sd 746 se 0 di 2 p1 0 ..
MAIN:ln 83 neineoeoenelhnepdhfjclmmoejffoejfgpdneinellnelhhfoejhlfpdjm ..
Run 1 gen 2
id 0 sc 0 ft -2489 pl 81 ti 100 id 14216 ld 10004 sd 11616 se 0 di 0 p1 ..
MAIN:ln 81 pdoeneicpdmhdgoeknelnejkhfncnekjoecfkmfknedgmcpdjoelednepdjh ..
id 1 sc 0 ft -2452 pl 82 ti 100 id 14216 ld 9665 sd 13223 se 0 di 0 p1 1 ..
MAIN:ln 82 oeoenelehfinemddhpdoekgpdmlcnekcoehgejioegnedepdnemfdcineg ..
id 2 sc 0 ft -665 pl 83 ti 100 id 14216 ld 4000 sd 14668 se 1 di 0 p1 12 ..
MAIN:ln 83 pdpcoeddpdpdjgjoepdfoegeoegikejfpdfmoejoefdneeidlmefoecgl ..
id 3 sc 0 ft -1686 pl 81 ti 100 id 14216 ld 4612 sd 5312 se 0 di 0 p1 29 ..
MAIN:ln 81 oeneneimhpepdijineoekeefejoelmcjpdlljkpdkpdjcdmoeioegnemhdhklp ..
..
..
Run 1 gen 10
..
..
id 399 sc 0 ft -2489 pl 81 ti 100 id 14216 ld 10004 sd 11616 se 0 di 0 p1 ..
MAIN:ln 81 pdoeneicpdmhdgoeknelnejkhfncnekjoecfkmfknedgmcpdjoelednepdjh ..
```

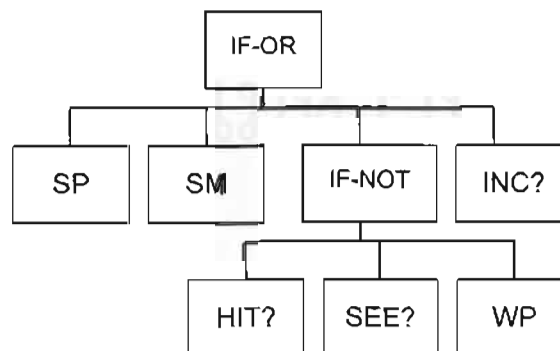
ค.8 ตัวโปรแกรม

โปรแกรมคือโครงสร้างกำหนดการเชิงพันธุกรรมจากเดิมเก็บไว้ในรายการโยงภายในหน่วยความจำมาเข้ารหัสอยู่ในรูปของสายอักขระ ด้วยการแวะผ่าน (traverse) ต้นไม้แบบก่อนลำดับ (preorder) ตัวอักขระ 1 ตัวแทน 1 บัพซึ่งเป็นบัพฟังก์ชัน ตามด้วยจำนวนลูก และบัพเทอมินอล ตามลำดับ

ตาราง 20 สัญลักษณ์แทนบัพในสายอักขระ

บัพ	สัญลักษณ์
SP	C
SM	D
EP	E
EM	F
WP	G
WM	H
HIT?	I
SEE?	J
INC?	K
DEC?	L
OUT?	M
IF-AND	N
IF-OR	O
IF-NOT	P

ตัวอย่างเช่นสายอักขระ `oecdpijgk` ตัว **O** ตัวแรกแทนบัพ IF-OR ตามด้วยจำนวนลูกคือ `e` ซึ่งแทนจำนวน 4 ตัว (ได้จากการนำ 'a' ไปลบจาก 'e' ตามรหัส ASCII) และ `cdpijgk` แทน SP, SM, IF-NOT, 3 (จำนวนลูก), HIT?, SEE?, WP, INC? ตามลำดับ ซึ่งนำมาสร้างต้นไม้ได้ดังนี้



รูปที่ 73 ต้นไม้จากสายอักขระ `oecdpijgk`

ภาคผนวก ง

คำศัพท์คอมพิวเตอร์ที่ใช้ในวิทยานิพนธ์

คำศัพท์ต่างๆ ที่ได้นำมาไว้ในส่วนนี้ ได้มาจากหนังสือศัพท์คอมพิวเตอร์ฉบับราชบัณฑิตยสถาน ซึ่งจัดพิมพ์ครั้งที่ 4 พ.ศ. 2540 เพื่อใช้สำหรับเป็นการอ้างอิงและเทียบเคียงระหว่างคำศัพท์คอมพิวเตอร์ภาษาอังกฤษ และภาษาไทยที่ใช้ในวิทยานิพนธ์ฉบับนี้

ภาษาอังกฤษ	ภาษาไทย
Algorithm	ขั้นตอนวิธี
Array	แถวลำดับ
Bubble sort	การเรียงลำดับแบบฟอง
Compile	แปลโปรแกรม
Compiler	1. ตัวแปลโปรแกรม 2. โปรแกรมแปลโปรแกรม, คอมไพเลอร์
Computer	คอมพิวเตอร์, คณิตกรณ์
Command-line Interface	ส่วนต่อประสานรายคำสั่ง
Directory	สารบบ
File	แฟ้ม, แฟ้มข้อมูล
Function	1. ฟังก์ชัน 2. โปรแกรมฟังก์ชัน
Generation	1. ยุค 2. รุ่น 3. การก่อกำเนิด
GUI	ส่วนต่อประสานกราฟิกกับผู้ใช้
Hard disk	จานบันทึกแบบแข็ง
Hardware	1. ส่วนเครื่อง, ฮาร์ดแวร์ 2. ส่วนอุปกรณ์, ฮาร์ดแวร์
Help	คำอธิบาย
Hot key	แป้นลัด
Leaf	ใบ

ภาษาอังกฤษ	ภาษาไทย
Leaf node	บัพใบ (มีความหมายเหมือนกับ exterior node และ terminal node)
Library; Program Library	คลัง(โปรแกรม)
Linked list	รายการโยง
Memory	หน่วยความจำ
Menu	รายการเลือก
Menu Bar	แถบรายการเลือก
Mouse	เมาส์
Node	บัพ, จุดต่อ, ปม, ข้อ
Parameter	1. พารามิเตอร์ 2. ตัวแปรเสริม
Parent Node	บัพแม่
Pixel	จุดภาพ
Pointer	ตัวชี้
Process	1. กรรมวิธี, กระบวนการ 2. ประมวลผล(ผล)
Procedure	1. กระบวนการ 2. กระบวนคำสั่ง
Program	1. โปรแกรม, ชุดคำสั่ง 2. สร้างโปรแกรม
Quicksort	ควิกซอร์ต
Record	1. ระเบียบ 2. บันทึก
Root	ราก
Software	ส่วนชุดคำสั่ง, ซอฟต์แวร์
Tag	ป้ายระบุ
Terminal	เครื่องปลายทาง*
Terminal node	บัพปลายทาง (มีความหมายเหมือนกับ exterior node และ leaf node)

* ในวิทยานิพนธ์นี้ใช้คำว่า เทอมินอล แทนคำว่า เครื่องปลายทาง

ภาษาอังกฤษ	ภาษาไทย
Thread	สายโยงใย
Tree	(รูป)ต้นไม้
Tree Structure	โครงสร้างรูปต้นไม้
Visualisation; Visualization	การสร้างภาพนามธรรม

คำศัพท์ที่ได้แสดงไว้ในส่วนต่อจากนี้ ไม่ปรากฏในหนังสือศัพท์คอมพิวเตอร์ฉบับราชบัณฑิตยสถาน จึงได้ใช้คำแปลเป็นภาษาไทย จากการสอบถามอาจารย์ที่ปรึกษา ผศ. ดร.ประภาส จงสภิตยวัฒนา และจากคำศัพท์ในวิทยานิพนธ์ของ ชัยวัฒน์ เจษฎาปกรณ์ [4] ซึ่งนอกจากการเพิ่มภาษาอังกฤษไว้ในวงเล็บท้ายคำที่ปรากฏในวิทยานิพนธ์แล้วผู้วิจัยเห็นว่าควรนำมาแสดงไว้ในส่วนนี้ด้วยเพื่อป้องกันความสับสน

ภาษาอังกฤษ	ภาษาไทย
Crossover	การไขว้เปลี่ยน
Dialog Box	กล่องโต้ตอบ
Effort	ค่าความเพียรพยายาม
Evolve; Evolution	การวิวัฒนาการ
Fitness	ค่าความเหมาะสม
Genetic Algorithm (GA)	ขั้นตอนวิธีเชิงพันธุกรรม
Genetic Programming (GP)	กำหนดการเชิงพันธุกรรม
Mutation	การกลายพันธุ์, การกลาย
Random Seed	เลขสุ่มเริ่มต้น
Reproduction	การสืบพันธุ์

* ในวิทยานิพนธ์นี้ใช้คำว่า การจินตทัศน์ข้อมูล แทนคำว่า การสร้างภาพนามธรรม

ประวัติผู้วิจัย

นายยอดธง รอดแก้ว เกิดวันที่ 29 มกราคม พ.ศ. 2518 ที่กรุงเทพมหานคร สำเร็จการศึกษาปริญญาตรีวิทยาศาสตร์บัณฑิต สาขาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยหอการค้าไทย ในปีการศึกษา 2538 และเข้าศึกษาต่อในหลักสูตรวิทยาศาสตรมหาบัณฑิต ที่จุฬาลงกรณ์มหาวิทยาลัย เมื่อ พ.ศ. 2540

