บทที่ 5

การโปรแกรมพันธุการ

บทนี้จะอธิบายรายละเอียดต่างๆที่สำคัญของการโปรแกรมพันธุการ โดยกล่าวถึงหลักการทั่วๆไปของการโปรแกรมพันธุการที่มีการพัฒนามาจากขั้นตอนวิธีพันธุการ รายละเอียดของลำดับขั้นตอนการทำงานของการโปรแกรมพันธุการ ลักษณะของโครงสร้างของโปรแกรมคอมพิวเตอร์ที่ใช้ ลักษณะการทำงานของตัวปฏิบัติการทางพันธุการของการโปรแกรมพันธุการ และสรุปท้ายบท

5.1 <u>หลักการทั่ว ๆไป</u>

เมื่อพิจารณาการทำงานของขั้นตอนวิธีพันธุการ การที่กระบวนการตามขั้นตอนวิธีพันธุการสามารถทำงานกับปัญหาที่กำหนดได้เป็นอย่างดีนั้นเป็นเพราะว่า ขั้นตอนวิธีพันธุการใช้กระบวนการที่กระทำกับสัญลักษณ์ที่ใช้แทนกับปัญหาโดยตรง โดยที่การแทนปัญหานั้นสามารถที่จะกำจัดขอบเขตของระบบเพื่อที่จะปฏิบัติตามสภาพของปัญหานั้นๆได้ ปรกติแล้วขั้นตอนวิธีพันธุการจะปฏิบัติการกับการใช้สายอักขระที่จำกัดความยาวซึ่งสามารถใช้ในการแก้ปัญหาต่างๆได้มากมายแต่อยู่ในวงจำกัด ซึ่งไม่สามารถใช้กับปัญหาที่มีลักษณะการแก้ปัญหาที่มีลักษณะเป็นกระบวนการที่อาศัยการคำนวณที่ไม่ตายตัว มีกระบวนการทำงานที่มีลักษณะของการวนซ้ำ มีกระบวนการเวียนเกิด มีชุดคำสั่งหลักที่ใช้กับงานหลัก และมีชุดคำสั่งย่อยที่ใช้กับงานย่อย โดยมีขนาดและรูปร่างที่ไม่แน่นอน ซึ่งทั้งหมดนั้นคือลักษณะของโปรแกรมคอมพิวเตอร์นั้นเอง

Koza (1992) ได้เสนอแนวคิดของการโปรแกรมพันธุการ ดังต่อไปนี้

การโปรแกรมพันธุการมีลักษณะที่เหมือนกับขั้นตอนวิธีพันธุการ โดยยังคงใช้ลักษณะในการค้นหาคำตอบของปัญหาตามขบวนการของขั้นตอนวิธีพันธุการ แต่เพิ่มการใช้โครง-

สร้างที่ผ่านการทำให้เหมาะสมกับการแทนปัญหาแล้ว โครงสร้างที่ว่าจะมีลักษณะเป็นโปรแกรมคอมพิวเตอร์ที่มีลำดับชั้นของกระบวนการซึ่งมีขนาดและรูปร่างที่ไม่แน่นอน ทำให้การโปรแกรม-พันธุการสามารถใช้ได้กับปัญหาได้ทั่วๆไป ซึ่งในหลายๆปัญหาการแก้ปัญหาด้วยการใช้โปรแกรมคอมพิวเตอร์มีความเหมาะสมมากกว่าใช้การแทนปัญหาด้วยสายอักขระ ยกตัวอย่างเช่น การ-ประมวลผลทางภาษา(Language Processing) และการเรียนรู้ของเครื่อง(Machine Learning) เป็นต้น

การโปรแกรมพันธุการสามารถทำได้ในลักษณะเดียวกับขั้นตอนวิธีพันธุการโดยเปลี่ยนรูปแบบของการแทนค่าด้วยสายอักขระ เป็นรูปแบบของโปรแกรมคอมพิวเตอร์ แล้วทำการค้นหาโปรแกรมคอมพิวเตอร์ที่เหมาะสมที่สุดภายในขอบเขตที่เป็นไปได้ที่ประกอบด้วย ฟังก์ชัน (functions) และเทอร์มินอล(terminals) ที่เหมาะสมภายใต้ขอบเขตของปัญหา โดยสรุปแล้ว การ-โปรแกรมพันธุการเป็นวิธีการที่ใช้ค้นหาโปรแกรมคอมพิวเตอร์ที่เหมาะสมที่สุดสำหรับปัญหาที่กำ-หนดนั้นเอง

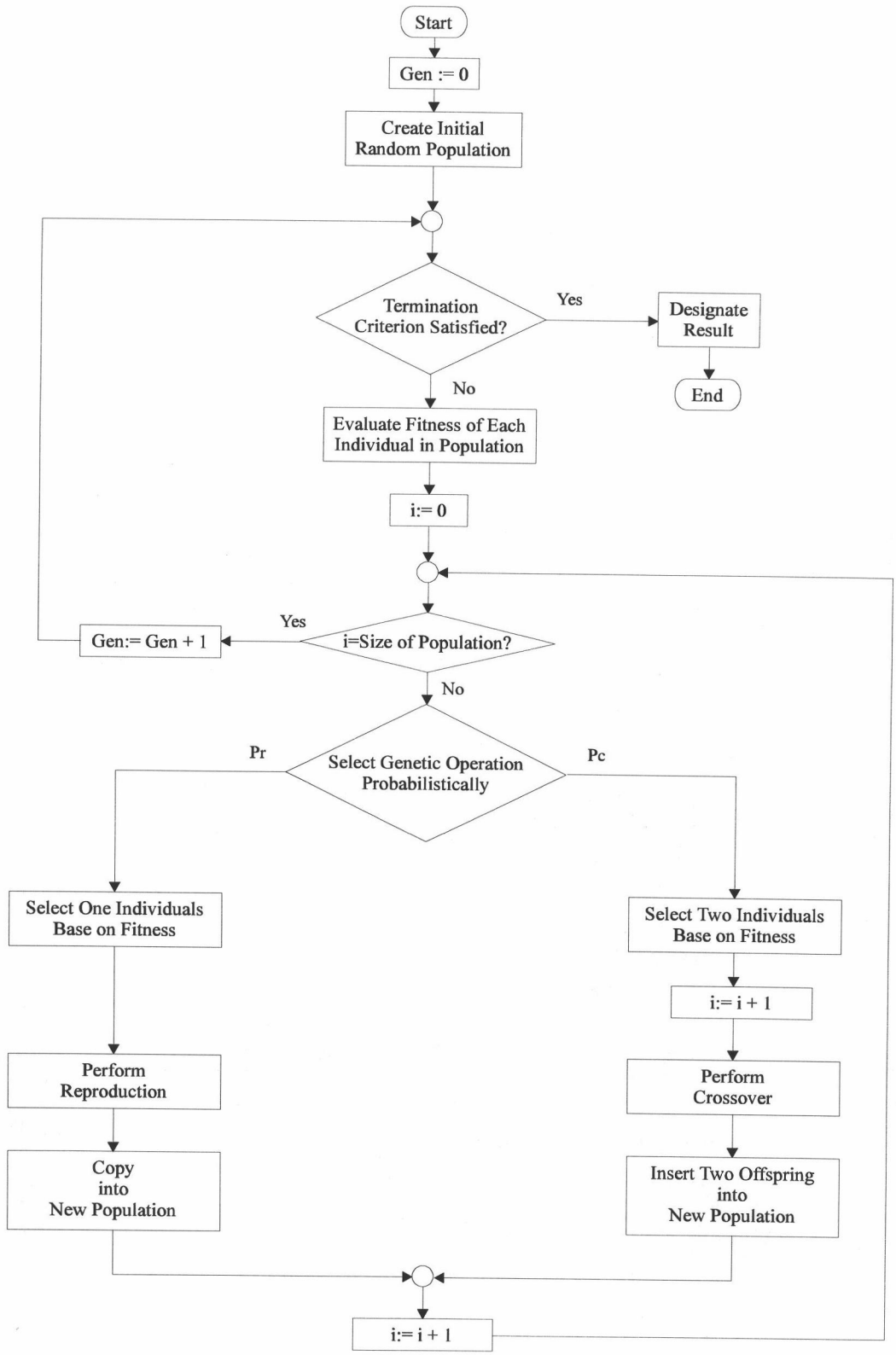
ในการโปรแกรมพันธุการ ประชากรของโปรแกรมคอมพิวเตอร์จะถูกดำเนินการตามขบวนการคัดเลือกและการสืบพันธุ์เลียนแบบตามวิธีการทางพันธุกรรมตามธรรมชาติที่เกี่ยว-ข้องกับการมีชีวิตรอดและการผสมพันธุ์ของสิ่งมีชีวิต โดยที่จะพิจารณาโปรแกรมคอมพิวเตอร์จำนวนหนึ่งที่มีความเหมาะที่สุดในประชากรทั้งหมด แล้วทำการจับคู่โปรแกรมคอมพิวเตอร์ เพื่อทำการผสมพันธุ์ ให้ได้ผลลัพธ์เป็นลูกหลานโปรแกรมคอมพิวเตอร์ในรุ่นถัดไป ในที่สุดแล้วจะได้โปรแกรมคอมพิวเตอร์ที่เหมาะสมสามารถใช้แก้ปัญหาที่กำหนดให้ได้ทั้งหมด(หรือโดยประมาณ)

5.2 <u>รายละเอียดของการโปรแกรมพันธุการ</u>

การโปรแกรมพันธุการ มีลักษณะของลำดับขั้นตอนการทำงานในลักษณะเดียวกับขั้นตอนวิธีพันธุการ โดยมีขบวนการสืบพันธุ์ด้วยวิธีทางพันธุการในประชากรของโปรแกรมคอมพิวเตอร์เพื่อหาคำตอบของปัญหาที่กำหนดไว้ ดังรูปที่ 5.1 โดยมีขั้นตอนดังนี้

(ก)สร้างกลุ่มของประชากรเริ่มต้นของการผสมรวมกันโดยอาศัยวิธีการสุ่มระหว่างฟังก์ชันและเทอร์มินอลของปัญหานั้นๆ(โปรแกรมคอมพิวเตอร์)

(ข)ปฏิบัติตามขั้นตอนต่อไปนี้ซ้ำไปเรื่อยๆจนกระทั่งถึงเป้าหมายที่ตั้งเงื่อน-ไขไว้



รูปที่ 5.1 แสดงแผนภูมิสายงานของการโปรแกรมพันธุการ

- ปฏิบัติการตามแต่ละโปรแกรมคอมพิวเตอร์ ในประชากรทั้งหมด และให้ค่าความเหมาะที่สอดคล้องกับความสามารถในการทำงานได้ดีเพียงใดต่อปัญหาที่กำหนดได้ ภายใต้สภาพแวดล้อมที่กำหนดให้

- สร้างกลุ่มประชากรของโปรแกรมคอมพิวเตอร์ขึ้นใหม่ บนพื้นฐานของค่าความเหมาะของแต่ละโปรแกรมคอมพิวเตอร์ที่ได้จากการปฏิบัติการ นำไปประยุกต์ใช้กับขบวนการทางพันธุการ 2 กระบวนการคือ

การกำเนิดใหม่ โดยการคัดลอกโปรแกรมคอมพิวเตอร์เดิมไปใช้ในกลุ่มประชากรใหม่

การผสมพันธุ์ โดยการสร้างโปรแกรมคอมพิวเตอร์ใหม่ขึ้นมาโดยอาศัยการแลกเปลี่ยนส่วนของโปรแกรมที่สุ่มเลือก ของสองโปรแกรมคอมพิวเตอร์เพื่อเป็นโปรแกรมคอมพิวเตอร์ที่นำไปใช้ในกลุ่มประชากรใหม่

(ค) เลือกโปรแกรมคอมพิวเตอร์ที่ดีที่สุดในประชากรทั้งหมดภายในเวลาที่ถึงเป้าหมายที่ตั้งเงื่อนไขไว้ โปรแกรมคอมพิวเตอร์ที่ได้จะเป็นคำตอบ หรือใกล้เคียงกับคำตอบของปัญหาที่ได้กำหนดไว้

โครงสร้างในการโปรแกรมพันธุการที่มีลักษณะเป็นโปรแกรมคอมพิวเตอร์ที่มีลำดับชั้นที่ประกอบไปด้วย เซตที่เป็นไปได้ทั้งหมดซึ่งประกอบขึ้นจากกระบวนการเวียนเกิดของเซตของฟังก์ชัน $F=\{f_1,f_2,f_3,...,f_{Nfunc}\}$ และเซตเทอร์มินอล $T=\{a_1,a_2,a_3,...,a_{Nterm}\}$

ฟังก์ชันที่ใช้ในหลายๆปัญหา อาจจะประกอบด้วย

-การดำเนินการคำนวณ (arithmetic operations) เช่น +, -, *, / เป็นต้น

-ฟังก์ชันทางคณิตศาสตร์ (mathematical functions) เช่น sin, cos, exp, log เป็นต้น

-การดำเนินการแบบบูล(boolean operations) เช่น AND, OR, NOT เป็นต้น

-การดำเนินการแบบมีเงื่อนไข (conditional branching operations) เช่น If-Then-Else เป็นต้น

-ฟังก์ชันที่ทำให้เกิดการวนซ้ำ (functions causing iteration) เช่น Do-Until เป็นต้น

-ฟังก์ชันที่ทำให้เกิดการเวียนเกิด (functions causing recursion)

-ฟังก์ชันเฉพาะงานอื่นๆ (domain-specific functions)

I16495642

ส่วนเทอร์มินอลนั้น จะแทนด้วยฟังก์ชันที่ไม่มีอาร์กิวเมนต์ เช่น คำสั่งพื้นฐานของหุ่นยนต์(primitive robotic actions) TL(Turn_Left), TR(Turn_Right), F(Forward), B (Backward) เป็นต้น หรือเป็นค่าคงที่ของระบบ เช่น NIL เป็นต้น เทอร์มินอลส่วนมากจะเป็นฟังก์ชันที่มีผลกระทบโดยตรงกับสถานะของระบบ

พิจารณาเซตของฟังก์ชัน

$F = \{ +, -, *, / \}$

และเซตของเทอร์มินอล

$T = \{N0, N1\}$

เมื่อ N0 และ N1 เป็นตัวแปรแบบตัวเลข (Number Variables) ทำหน้าที่เป็นอาร์กิวเมนต์ให้กับฟังก์ชัน

อย่างไรก็ตามเราอาจจะทำการรวมทั้งเซตของฟังก์ชัน และเซตของเทอร์มินอลเข้าด้วยกันเป็นเซต C ดังนี้

$C = F \cup T = \{ +, -, *, /, N0, N1\}$

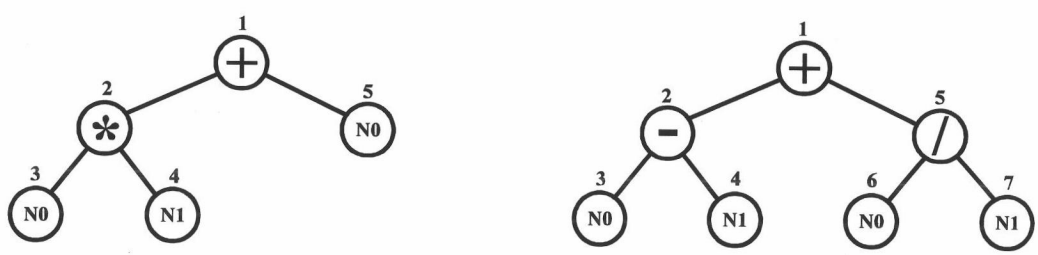
โดยที่เซต C เราสามารถแยกส่วนของเทอร์มินอลได้จากฟังก์ชันที่มีอาร์กิวเมนต์เป็นศูนย์ ซึ่งสมาชิกแต่ละตัวในเซต C มีอาร์กิวเมนต์เป็น 2, 2, 2, 2, 0 และ 0 ตามลำดับ

สำหรับรูปแบบของโปรแกรมคอมพิวเตอร์ที่ใช้สำหรับการโปรแกรมพันธุการ มีลักษณะเป็นภาษาลิสป์ (LISP language) ที่มีลักษณะเป็นนิพจน์เชิงสัญลักษณ์(The symbolic expressions) ซึ่งสามารถสร้างและดัดแปลงได้ง่าย สำหรับการผสมผสานระหว่างฟังก์ชัน และเทอร์มินอลของโปรแกรมคอมพิวเตอร์ ลักษณะนี้จะอาศัยโครงสร้างแบบต้นไม้ ซึ่งใช้กันมากในการสร้างตัวแปลโปรแกรมคอมพิวเตอร์ ตัวอย่างเช่น

(+ (* N0 N1) N0)

(+ (- N0 N1) (/ N0 N1))

สามารถแทนโปรแกรมคอมพิวเตอร์ทั้งสองข้างต้น เป็นต้นไม้ ดังรูปที่ 5.2



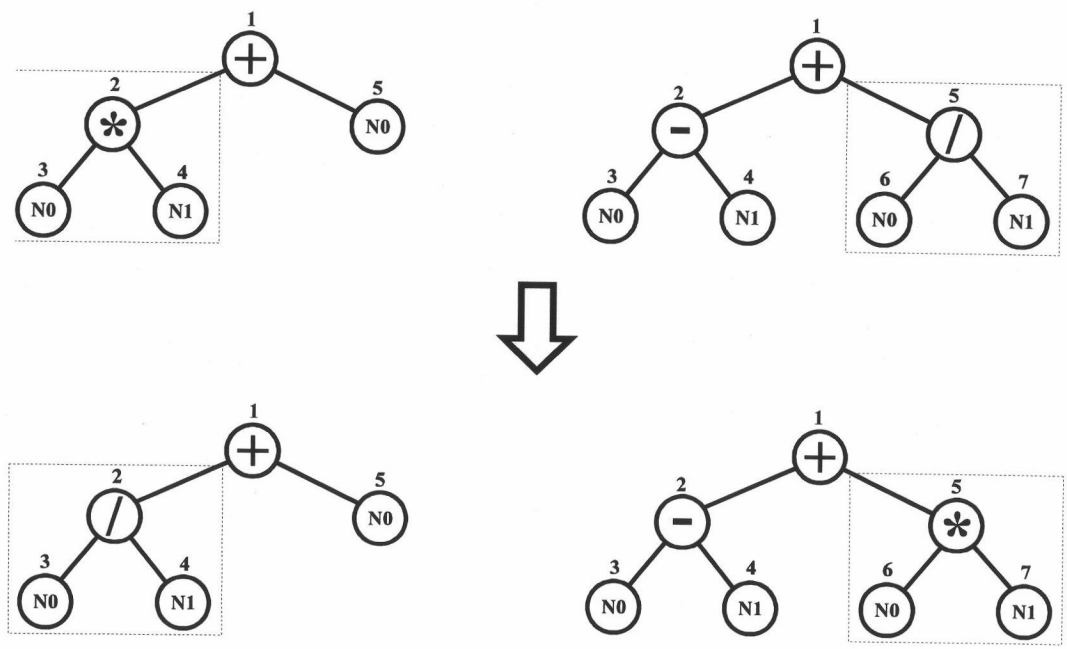
*รูปที่ 5.2 ต้นไม้ที่แทนโปรแกรมคอมพิวเตอร์ (+ (* N0 N1) N0) ซ้าย และ(+ (- N0 N1) (/ N0 N1)) ขวา*

5.3 ปฏิบัติการทางพันธุการของการโปรแกรมพันธุการ

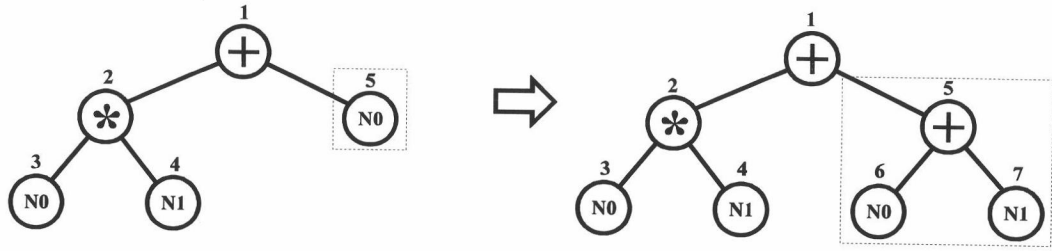
ในกระบวนการโปรแกรมพันธุการนั้น ปฏิบัติการทางพันธุการต่างๆจะอาศัยพื้นฐานของโครงสร้างแบบต้นไม้ดังกล่าว โดยถูกนำมาใช้แทนแต่ละโปรแกรมคอมพิวเตอร์ในประชากรทั้งหมด ซึ่งมีวิธีการที่แตกต่างไปจากการปฏิบัติการทางพันธุการของขั้นตอนวิธีพันธุการที่ใช้สายอักขระ ดังต่อไปนี้

5.3.1 การกำเนิดใหม่ เป็นการคัดลอกโปรแกรมคอมพิวเตอร์เดิมไปใช้ในกลุ่มประชากรใหม่ในรุ่นถัดไป โดยไม่มีการเปลี่ยนแปลงใดๆบนโปรแกรมคอมพิวเตอร์เดิม

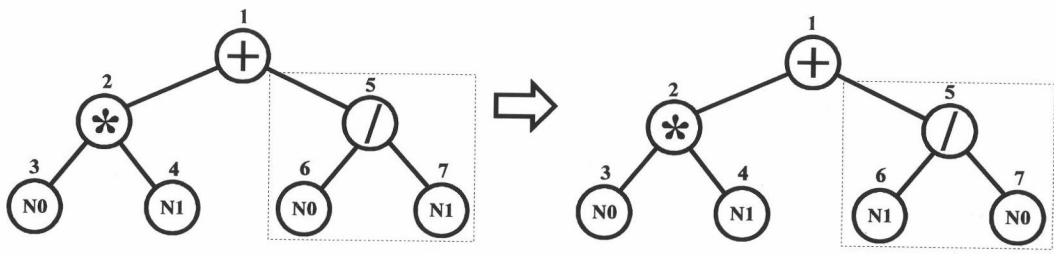
5.3.2 การผสมพันธุ์ เป็นการสร้างโปรแกรมคอมพิวเตอร์ขึ้นมาใหม่ โดยอาศัยการแลกเปลี่ยนกิ่งของต้นไม้โดยอาศัยวิธีการสุ่ม ทำให้เกิดต้นไม้ใหม่ที่ลักษณะร่วมกันของพ่อและแม่ ซึ่งจะถูกแทนเป็นโปรแกรมคอมพิวเตอร์ลูกหลานที่จะในการดำเนินการเพื่อแก้ปัญหาที่กำหนดในรุ่นถัดไป ตัวอย่างเช่น ในรูปที่ 5.3 แสดงการผสมพันธุ์ของโปรแกรมคอมพิวเตอร์ 2 โปรแกรม โดยที่ส่วนของกิ่งของโปรแกรมทั้งสองที่มีกรอบสี่เหลี่ยมล้อมรอบเป็นส่วนที่ถูกสุ่มเลือก จากนั้นจะทำการสลับกิ่งทั้งสอง เมื่อผ่านขบวนการผสมพันธุ์แล้วจะได้ต้นไม้ของโปรแกรมลูกที่มีลักษณะที่ร่วมกันของพ่อและแม่



รูปที่ 5.3 ตัวอย่างต้นไม้ที่เกิดจากการผสมพันธุ์ พ่อแม่(บน) และลูก(ล่าง)



รูปที่ 5.4 ตัวอย่างต้นไม้ในขบวนการกลายพันธุ์



รูปที่ 5.5 ตัวอย่างต้นไม้ในขบวนการเปลี่ยนรูป

นอกจากนี้ยังมีลักษณะการปฏิบัติการทางพันธุการของการโปรแกรมพันธุการในลักษณะอื่นๆอีก ดังต่อไปนี้

5.3.3 การกลายพันธุ์ เป็นปฏิบัติการที่ใช้การเปลี่ยนแปลงแบบสุ่มของต้นไม้ในประชากร โดยการเลือกจุดๆหนึ่งบนต้นไม้โดยอาศัยวิธีการสุ่ม ซึ่งจุดที่จะทำการกลายพันธุ์บนต้นไม้อาจจะเป็นฟังก์ชัน หรือ เทอร์มินอลก็ได้ ที่จุดนั้นจะทำการลบกิ่งของต้นไม้โดยลบตั้งแต่จุดที่เลือกลงมาจนถึงใบของต้นไม้ทั้งหมด แล้วจึงทำการสุ่มสร้างต้นไม้ย่อย(subtree)ที่จุดนั้นขึ้นแทน ตัวอย่างดังรูปที่ 5.4 โดยจุดที่ถูกสุ่มเลือกเพื่อการกลายพันธุ์เป็นจุด 5 (N0) และ (+ N0 N1)ถูกสุ่มสร้างที่จุด 5 ได้ต้นไม้ที่มีลักษณะใหม่

5.3.4 การเปลี่ยนรูป (permutation) เป็นปฏิบัติการที่ทำการสลับสับเปลี่ยนลำดับของปมภายใต้กิ่งของต้นไม้ที่ถูกสุ่มเลือก โดยการเลือกจุดๆหนึ่งบนต้นไม้โดยอาศัยวิธีการสุ่ม ซึ่งจุดที่จะทำการกลายพันธุ์บนต้นไม้จะต้องเป็นฟังก์ชัน ที่จุดนั้นจะทำการสลับสับเปลี่ยนลำดับของทุกๆปมภายใต้กิ่งของต้นไม้ตั้งแต่จุดที่เลือกลงมาจนถึงใบของต้นไม้ทั้งหมด ตัวอย่างดังรูปที่ 5.5 โดยจุดที่ถูกสุ่มเลือกเพื่อการเปลี่ยนรูปเป็นจุด 5 (/ N0 N1) และ ทำการสลับสับเปลี่ยนลำดับภายใต้จุด 5 ได้เป็น (/ N1 N0) ทำให้ต้นไม้เปลี่ยนรูปไป

5.3.5 การแก้ไข (editing) เป็นปฏิบัติการที่ใช้ตรวจสอบเพื่อแก้ไขในบางส่วนของโปรแกรมคอมพิวเตอร์ ที่ไม่มีผลกระทบกับกระบวนการทำงานของโปรแกรม เพื่อทำให้เกิดความกระชับของโปรแกรมคอมพิวเตอร์ เช่น

(+ 1 2) --> 3

(AND X X) --> X

(OR X X) --> X

(NOT (NOT X)) --> X เป็นต้น

5.4 สรุปท้ายบท

โดยสรุปแล้ว การโปรแกรมพันธุการ เป็นแนวคิดที่ว่าถึงการสร้างระบบที่มีการแลกเปลี่ยนซึ่งกันและกันของส่วนประกอบ(ส่วนประกอบที่กล่าวถึงนี้อาจหมายถึง โปรแกรม หรือ กฎเกณฑ์ต่างๆ) ภายในระบบเป็นจำนวนมาก ด้วยกรรมวิธีการสุ่ม โดยใช้เงื่อนไขภายนอกบางอย่างในการประเมินค่าส่วนประกอบนั้นๆ ถ้าส่วนนั้นๆดีพอแก่การให้รางวัลโดยให้ส่วนประกอบดังกล่าว ยังคงอยู่ ซึ่งสามารถนำไปจับคู่ทำการผลิตใหม่ เพื่อการขยายพันธุ์ต่อไป ส่วนที่ประเมินค่าว่าไม่ดีพอ ก็จะถูกทำลายทิ้ง

โดยหลักการทั่วๆไปแล้ว การโปรแกรมพันธุการ เริ่มต้นด้วย กฎ หรือ โปรแกรมจำนวนมาก(ประมาณ 1000 - 10000 โปรแกรม) แต่ละส่วนจะถูกนำไปทดสอบความสามารถในการแก้ปัญหาที่ได้กำหนดไว้ โดยใช้วิธีการสุ่มในการเลือก ถ้ากฎหรือโปรแกรมนั้นประสพความสำเร็จในการแก้ปัญหา จะถูกเก็บไว้ ซึ่งอาจนำไปจับคู่กับอีกกฎหรือโปรแกรมอื่น โดยถูกแปรสภาพเป็นกฎหรือโปรแกรมรุ่นต่อไป และจะถูกนำไปเก็บเพื่อทดสอบความสามารถในการแก้ปัญหาต่อไป ซึ่งขบวนการนี้เรียกว่า การวิวัฒนาการ หลังจากหลายๆรุ่นของการวิวัฒนาการ กฎหรือโปรแกรมที่คงสภาพอยู่ได้จะเป็นกฎหรือโปรแกรมที่มีคุณภาพสูงในการแก้ปัญหาที่กำหนดมานั้นๆ