การวิเคราะห์หาลำดับความสัมพันธ์ของกลุ่มยีน และแสดงผลในรูปแบบของ Interval Graph โดยใช้ PC -Trees

รวินทร์ วิรัชพินทุ และ นพดล คีรีเพ็ชร

ฝ่ายวิจัยและพัฒนาเทคโนโลยีคอมพิวเตอร์เพื่อการคำนวณ
ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ
112 อุทยานวิทยาศาสตร์แห่งประเทศไทย ถนนพหลโยธิน ต. คลองหนึ่ง อ. คลองหลวง จ. ปทุมธานี 12120
โทรศัพท์ 025646900 โทรสาร 025646772

E-mail: rawin.viruchpintu@nectec.or.th, noppadon.khiripet@nectec.or.th

บทคัดย่อ – งานวิจัยด้านเทคโนโลยีชีวสารสนเทศในปัจจุบัน ในด้านของการวิเคราะห์สายข้อมูลพันธุกรรมเพื่อหาความสัมพันธ์ของยีนของสิ่งมีชีวิต ยังขาดวิธีการแสดงผลข้อมูลความสัมพันธ์กันของกลุ่มยีนที่ง่ายต่อนำไปใช้งาน งานวิจัยนี้ได้นำข้อมูลสายพันธุกรรมของแบคทีเรีย มาทำการเปรียบเทียบเพื่อหาลักษณะความคล้ายกันด้วยโปรแกรม BLAST แล้วนำผลมาจัดกลุ่มด้วยวิธี Maximal Clique Detection ผลลัพธ์ที่ได้จะอยู่ในรูปแบบเมทริกซ์ ความสัมพันธ์ของยีนกับกลุ่มยีน ที่มีค่า 0 หรือ 1 เพื่อให้ได้ข้อมูลเมทริกซ์ที่อยู่ในรูปแบบของ Interval Graph ที่ง่ายต่อนำไปใช้งาน งานวิจัยนี้ได้ใช้หลักการของ PC-Trees Algorithm ที่ใช้คุณสมบัติ Circular Ones Property (CROP) ของข้อมูล ในการหาลำดับของความสัมพันธ์ของแต่ละคอลัมน์ (กลุ่มยีน) ของข้อมูลเมทริกซ์ แล้วนำลำดับของกลุ่มยีนที่ได้มาเรียงข้อมูลเมทริกซ์ใหม่ ได้ข้อมูลที่เป็น Interval Graph โปรแกรมดังกล่าว ถูกพัฒนาด้วยภาษา Perl เนื่องจากมีความสามารถในการจัดการกับข้อมูลอักขระได้ดี มีโครงสร้าง hash table ที่ใช้สร้าง PC -Trees ได้อย่างรวดเร็วและมีประสิทธิภาพ Interval graph ที่ได้จากการทดลองกับข้อมูลพันธุกรรมของแบคทีเรีย ทำให้การศึกษายีนที่มีลักษณะแบบยีนร่วมของหลายๆ กลุ่มเป็นไปได้โดยง่าย

คำสำคัญ - interval graph, PC-Trees algorithm, circular ones property

1. บทนำ

ข้อมูลในงานวิจัยด้านชีวสารสนเทศศาสตร์ส่วนใหญ่ เป็นข้อมูลที่มีปริมาณมากและมีความสัมพันธ์ต่อกันที่ค่อนข้างซับซ้อน การนำเสนอข้อมูลในรูปแบบของตารางความสัมพันธ์ที่สื่อความหมายได้น้อย และยากต่อการนำไปใช้งานในเชิงวิเคราะห์ Interval graph เป็นการแสดงความสัมพันธ์ของข้อมูลให้เข้าใจง่ายรูปแบบหนึ่ง โดยจะแสดงข้อมูลอยู่ในรูปของเส้นตรงที่ต่อเนื่องกัน ตามขอบเขตที่ข้อมูลนั้นมีความเกี่ยวข้องกันอยู่จริง ทำให้สามารถมองเห็นข้อมูลที่อยู่ในหลายๆ กลุ่มได้ชัดเจน และรู้ขนาดของกลุ่มได้สะดวก

ในงานวิจัยด้านชีววิทยาของ Seymour Benzer [1] , Interval graph ถูกนำมาในการแสดงผลข้อมูล โดยอาศัย graph theory ในการหาลำดับคอลัมน์ที่เหมาะสมของชุดข้อมูล เพื่อนำไปเรียงใหม่ให้อยู่ในรูปของ Interval graph แต่ยังเป็นวิธีการที่ค่อนข้างซับซ้อนต่อการนำมาใช้งาน

ต่อมา Booth และ Lueker ได้เสนอวิธีการใหม่ในการสร้าง Interval graph โดยใช้การตรวจสอบคุณสมบัติ Consecutive ones property (COP) ของชุดข้อมูลเมทริกซ์ ด้วย PQ-Tree Algorithms เป็นโครงสร้างแบบต้นไม้ที่มีกิ่ง (node) สองรูปแบบ เพื่อแยกแยะความสำคัญของการเรียงลำดับใบ เพื่อช่วยในการจัดสลับเพื่อหาลำดับของข้อมูลที่

เหมาะสม ในรูปแบบของ Interval graph แต่พบปัญหาเนื่องจากมีขั้นตอนที่ซับซ้อนและต้องใช้รูปแบบ (template) จำนวนมากในขั้นตอนการปรับเปลี่ยนโครงสร้างของต้นไม้เพื่อหาคำตอบ ทำให้เข้าใจได้ยากและพัฒนาเป็นโปรแกรมได้ลำบาก

ต่อมา Wen-Lian Hsu และ Ross M. McConnell [2] ได้เสนอ PC-Tree Algorithms ซึ่งเป็นวิธีที่สามารถตรวจสอบคุณสมบัติ Circular-ones property [3] ของข้อมูลได้ และลดความซับซ้อนที่เกินความจำเป็นของ PQ-Tree Algorithms [4] โดยใช้เงื่อนไขในการลดรูปแทนรูปแบบ (template) ของ PQ-Tree ทำให้นำมาพัฒนาเป็นโปรแกรมได้ง่ายขึ้น

และเนื่องจาก Consecutive ones property (COP) เป็นคุณสมบัติที่เป็นสับเซตของ Circular-ones property (CROP) ดังนั้นจึงสามารถหา COP จาก CROP ได้โดยการเลือกหาจุดเริ่มต้นที่เหมาะสม และนำลำดับที่มี COP ไปจัดเรียงข้อมูลให้อยู่ในรูปแบบของ Interval graph ได้

คำจำกัดความ

Consecutive Ones Property (COP)

คุณสมบัติของข้อมูลเมทริกซ์ที่มีค่าหนึ่งในแต่ละแถวติดกันเป็นช่วงเดียวในแนวเส้นตรง

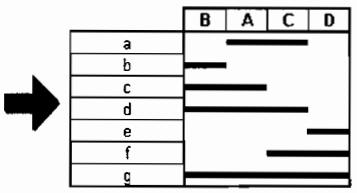
Circular Ones Property (CROP)

คุณสมบัติของข้อมูลเมทริกซ์ที่มีค่าหนึ่งในแต่ละแถวติดกันเป็นช่วงเดียวในแนววงกลม

Interval Graph

กราฟแสดงความสัมพันธ์ของข้อมูลที่อยู่ในรูปของเส้นตรงที่ต่อเนื่องกันโดยไม่ขาดช่วง ดังรูปที่ 1

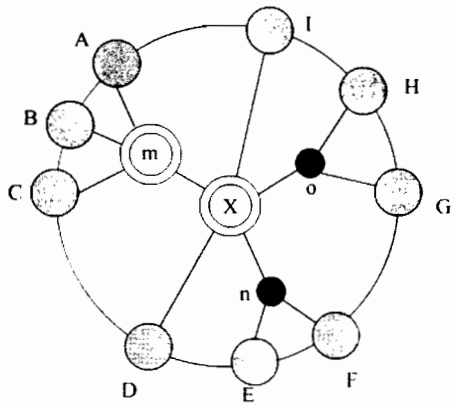
	B	A	C	D
a	0	1	1	0
b	1	0	0	0
c	1	1	0	0
d	1	1	1	0
e	0	0	0	1
f	0	0	1	1
g	1	1	1	1



รูปที่ 1 แสดงการเปลี่ยนจากข้อมูลตารางเป็น interval graph

PC-Tree

PC-Tree (Wen-Lian Hsu และ Ross M. McConnell [2]) เป็นโครงสร้างแบบต้นไม้ชนิดหนึ่งที่ไม่มีราก (root) ของต้นไม้ มีกิ่ง (node) 2 ชนิด คือ C node ซึ่งแทนด้วยสัญลักษณ์เป็นวงกลมสองวงซ้อนกันและ P node แทนด้วยสัญลักษณ์เป็นจุดดำ และลำดับของลูก (leaf node) ทำได้โดยการดูที่ปลายของต้นไม้หมุนวนเป็นวงกลม โดยการหมุนในทิศตามเข็ม หรือทวนเข็มนาฬิกาจนครบรอบวง ดังรูปที่ 2



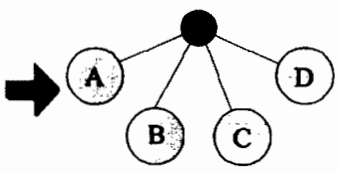
รูปที่ 2 แสดงตัวอย่าง C Tree

ในส่วนของคุณสมบัติของกิ่ง C node อนุญาตมีให้สลับที่ของสมาชิกภายใน C node ได้เฉพาะกรณีย้อนกลับลำดับ (reverse order) ของสมาชิกทุกตัวภายใน C node เท่านั้น แต่สำหรับ P node สามารถสลับตำแหน่งของสมาชิกได้โดยไม่ต้องสนลำดับภายใน P node ตัวเดียวกัน

การสร้าง PC-Tree

การกำหนด P node ขึ้นมา 1 จุด และนำคอลัมน์ของข้อมูลทั้งหมดมาสร้างเป็น leaf node ของ P node ตั้งต้น (โดยลำดับยังไม่สำคัญ) ดังรูปที่ 3

	A	B	C	D
a	1	0	1	0
b	0	1	0	0
c	1	1	0	0
d	1	1	1	0
e	0	0	0	1
f	0	0	1	1
g	1	1	1	1

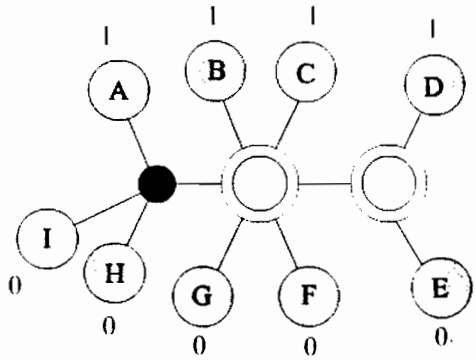


รูปที่ 3 แสดงสร้าง PC Tree จากข้อมูลเมทริกซ์

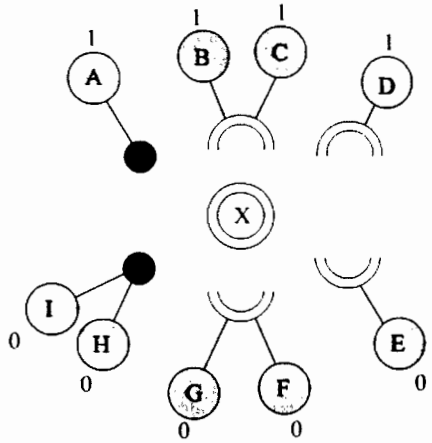
ขั้นตอนการปรับโครงสร้างของ PC-Tree ในแต่รอบ (แถวข้อมูล)

หลังจากนำข้อมูลทีละแถวเข้าใส่ใน leaf node ของต้นไม้ ซึ่งจะทำให้ leaf node มี 2 ค่า คือ full leaves (node ที่มีค่าเป็น 1) และ empty leaves (node มีค่าเป็น 0)

1. ทำการหาเส้นแบ่งของต้นไม้ (Terminal path) (รูปที่ 4) ที่สามารถแบ่ง full leaves และ empty leaves ออกจากกันด้วยเส้นเชื่อมระหว่างกิ่งหลายๆกิ่งเพียงเส้นทางเดียว ที่ไม่ซ้อนทับและวนซ้ำกัน โดยอาศัยการสลับลำดับของลูก (leaves) ถ้าหาก node ที่ทำการแบ่งเป็นชนิด
2. ทำการแบ่งกิ่งที่เป็น partial node (กิ่งที่มีลูกทั้ง 2 ชนิด คือ full leaves และ empty leaves) ออกเป็น 2 ส่วน โดยการสร้าง node ประเภทเดียวกันออกมา 2 กิ่ง และย้าย full leaves และ empty leaves ไปคนละกิ่งกัน ดังรูปที่ 5

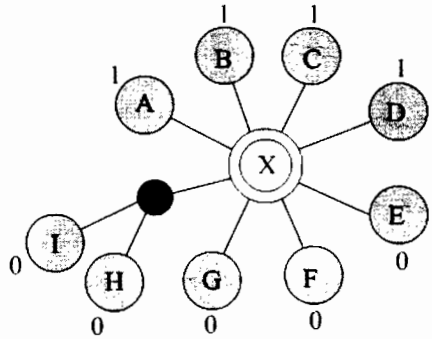


รูปที่ 4 แสดงแถบสีแสดง terminal path ที่แบ่ง leaves ออกเป็น 2 ด้าน

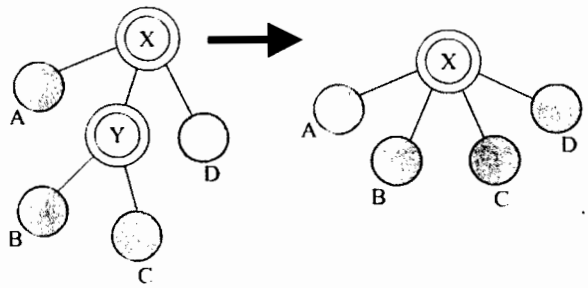


รูปที่ 5 แสดงการแบ่ง nodes บน terminal path ออกเป็น 2 ด้าน

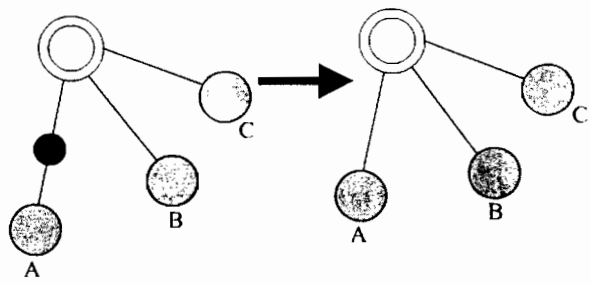
3. หลังจากทำการแบ่งกิ่งครบทั้ง Terminal path แล้ว (ลบกิ่งที่ถูกแบ่งทิ้ง) ให้ทำการสร้าง C node ขึ้นมาใหม่ 1 ตัว แล้วทำการเชื่อมโยงกับทุก node ที่ถูกทำการ Split โดยเรียงตามลำดับของเส้นทาง Terminal path ซึ่งผลจากการเชื่อมโยงจะต้องได้ต้นไม้ใหม่ที่แบ่ง full leaves กับ empty leave ออกจากกันคนละฝั่งของต้นไม้แล้ว
4. เมื่อทำการเชื่อมโยงลูกที่เกิดจากการแบ่งทั้งหมดเข้ากับ C node ตัวใหม่เสร็จแล้ว (รูปที่ 6) อาจมีการยุบกิ่งที่ไม่มีความสำคัญออกหรือลดรูปของต้นไม้ลง เพื่อให้อยู่ในโครงสร้างที่สั้นและกระชับ เช่น การยุบ C node ที่เป็นลูกของ C node เข้าด้วยกัน โดยยังคงลำดับของ leaves node ไว้ให้เหมือนเดิม (รูปที่ 7) หรือการยุบ node ที่ไม่จำเป็น (รูปที่ 8)



รูปที่ 6 แสดง PC Tree หลังการรวมต้นไม้



รูปที่ 7 แสดงตัวอย่างการยุบโหนด C ที่มีลูกเป็นโหนด C

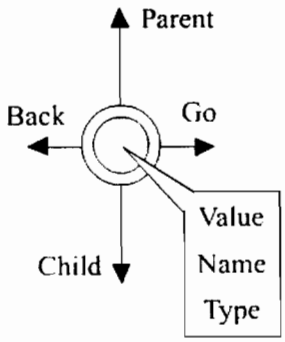


รูปที่ 8 แสดงตัวอย่างการยุบโหนด P ที่ไม่จำเป็น

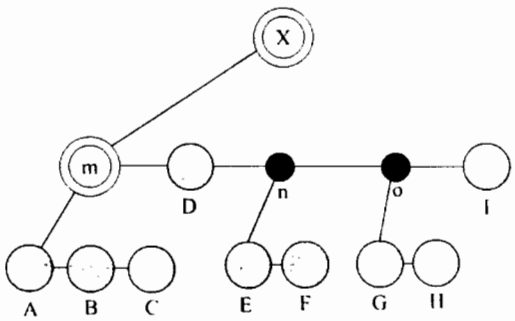
ทำวนซ้ำจนหมดข้อมูล ถ้าหากไม่มีการขัดเงื่อนไขของการสลับลูกใน C node จนกระทั่งครบหมดทั้งชุดข้อมูล หลังจากได้โครงสร้างสุดท้ายของต้นไม้แล้ว ลำดับของลูก (leaves node) สุดท้ายนั้นจะเป็นลำดับของคอลัมน์ที่ทำให้ข้อมูลมีค่าต่อเนื่องกัน ตามคุณสมบัติ Circular Ones Property

การพัฒนาโปรแกรม

ภาษาในใช้ในการพัฒนาในงานคือภาษา PERL เพราะตัวภาษาสนับสนุนการจัดการกับข้อมูลแบบอักขระได้ดี และมีโครงสร้างทางข้อมูลที่สะดวกต่อการนำไปใช้งาน เช่น hash table datastructure ทำให้สามารถพัฒนาได้เร็ว ส่วนโครงสร้างของข้อมูลที่ใช้ทำเป็นกิ่งและใบ (node และ leaf node) ของ PC-Tree ในครั้งนี้ ถูกออกแบบให้อยู่ในรูปแบบดังรูปที่ 9 และ 10



รูปที่ 9 แสดงโครงสร้างข้อมูลที่ใช้

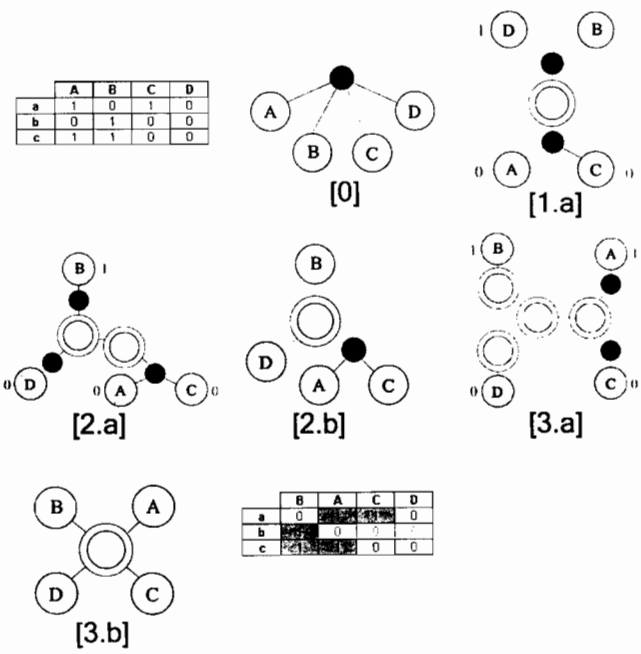


รูปที่ 10 แสดงโครงสร้างต้นไม้ที่สร้างจากข้อมูลรูปที่ 2

ลักษณะข้อมูลที่ใช้จะถูกจัดอยู่รูปแบบของตารางเมทริกซ์ 0-1 ขนาด m x n มิติ โดยกำหนดให้ m ที่เป็นจำนวนแถว คือชื่อข้อมูล และ n ที่เป็นคอลัมน์ของชุดข้อมูล คือกลุ่มของข้อมูล โดยใช้ค่า 1 แทนความหมายว่าข้อมูลนั้นอยู่ในกลุ่ม (คอลัมน์) นั้น และค่า 0 แทนความหมายว่าข้อมูลตัวนั้นไม่มีความสัมพันธ์กับกลุ่มในคอลัมน์นั้น ผลลัพธ์ที่ต้องการได้จาก PC-Tree คือ ลำดับของคอลัมน์ที่จะทำให้ค่าตัวเลข 1 และ 0 ของข้อมูลทั้งหมด ถูกเรียงต่อเนื่องกันแบบวงกลม (Circular ones property)

การเก็บข้อมูล เลือกใช้ hash Table datastructure เป็นโครงสร้างของ PC Tree เพราะในภาษา Perl hash table datastructure ถูกออกแบบมาให้ทำงานได้เร็วและมีความเร็วในการเข้าถึงข้อมูลขนาดใหญ่มากกว่าการเก็บข้อมูลแบบ array

ตัวอย่างแสดงขั้นตอนการทำงานของ PC Tree



รูปที่ 11 แสดงการเปลี่ยนแปลงของ PC Tree แต่ละขั้นตอน

จากรูปที่ 11 เป็นตัวอย่างการเปลี่ยนแปลงโครงสร้างของ PC Tree ทีละขั้นตอน โดยเริ่มจากการสร้าง Tree [0] จากคอลัมน์ของข้อมูล แล้วนำข้อมูลทีละแถวเข้าทำการเปลี่ยนแปลง tree ตามขั้นตอนที่กล่าวข้างต้น (1.a , 2.a , 3.a) และยุบ node ที่ไม่จำเป็น เช่นในรูป [2.b] และ [3.b] ทำซ้ำจนข้อมูลหมดทุกแถว ถ้าหากพบว่าไม่ขัดเงื่อนไขการสลับลำดับของ C node ก็จะได้โครงสร้างของ tree สุดท้ายเป็นลำดับของคอลัมน์ที่จะทำให้ข้อมูลมีคุณสมบัติ CROP ครบทั้งตาราง

ปัญหาและแนวทางในการพัฒนาต่อ

เนื่องจากโปรแกรมที่พัฒนาในเวอร์ชันนี้ถูกออกแบบให้แสดงผลลัพธ์สุดท้ายออกมาในชุดข้อมูลที่พร้อมสำหรับการแปลงไปเป็น Interval graph แต่ในขั้นตอนการเปลี่ยนแปลงที่ใช้กราฟในการวิเคราะห์ยังไม่ได้แสดงผลออกมาในรูปของกราฟิก ทำให้การตรวจสอบทีละขั้นตอนค่อนข้างยุ่งยาก และเพื่อให้ง่ายต่อการใช้งานมากขึ้น ในขั้นต่อไปจะพัฒนาส่วนติดต่อผู้ใช้ให้สามารถควบคุมการแสดงผลในแต่ละขั้นตอนได้

งานส่วนต่อไปคือ นำข้อมูลจริงที่ได้จากผลลัพธ์ของการจัดกลุ่ม Maximam Clique Detection ของข้อมูลยีนแบคทีเรีย ซึ่งข้อมูลที่ได้เป็นกลุ่มของยีนที่มีความสัมพันธ์กันทั้งกลุ่ม จำนวนหลายๆกลุ่มมาทำการทดสอบด้วย PC Tree algorithm และนำผลมาทำ Interval graph เพื่อหายีนที่มีความสัมพันธ์อยู่ในหลายๆกลุ่มจากความยาวของเส้นที่แสดงความสัมพันธ์ระหว่างกลุ่มของแต่ละยีน ซึ่งจะช่วยให้มองเห็นยีนของแบคทีเรียที่คล้ายกัน ในสายพันธุ์ที่มีความใกล้เคียงกันหลายๆ สายพันธุ์ได้สะดวกมากขึ้นและจะช่วยนำไปสู่การวิเคราะห์หายีนที่มีความสำคัญของแบคทีเรียหลายๆ สายพันธุ์อีกทางหนึ่ง

เอกสารอ้างอิง

[1] John R. Jungck, Dept. of Biology, Beloit College, Beloit, WI and Vince Streif, Deletion Mapping of Genetic "Fine Structure": Supplementing Ad Hoc Problem Solving Approaches with Algorithms and Heuristics, Computer Center, University of Wisconsin-Eau Clair, Eau Claire, WI

[2] Wen-Lian Hsu Ross M. McConnell , "PC Trees and circular-ones Arrangements", Theoretical Computer Science 296(1), (2003), 99-116.

[3] MohammadTaghi Hajiaghayi , "Consecutive Ones Property"

[4] Wen-Lian Hsu , "PC-Trees vs. PQ-Trees", Workshop on Graph Structures and Algorithms,also,appeared in Lecture Notes in Computer Science 2108, (2001),207-217.

รวินทร์ วิรัชพินทุ จบการศึกษาวิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมคอมพิวเตอร์ เกียรตินิยมอันดับ 2 จากสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง เมื่อ พศ. 2546 เข้าทำงานในศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ ในปี พ.ศ. 2547 ขอบเขตงานที่สนใจ คือ การพัฒนา visualization ในงานวิจัยเกี่ยวกับฐานข้อมูลของงาน Bioinformatics

นพดล คีรีเพ็ชร สำเร็จการศึกษาระดับปริญญาตรี สาขาฟิสิกส์จากมหาวิทยาลัยสงขลานครินทร์ จังหวัดสงขลาในปี 2536 และระดับปริญญาเอก ในสาขา Electrical and Computer Engineering จาก Georgia Institute of Technology ประเทศสหรัฐอเมริกาในปี 2544 ปัจจุบัน ดร. นพดล เป็นนักวิจัยสาขาชีวสารสนเทศศาสตร์ ศูนย์เทคโนโลยีอิเลคทรอนิกส์และคอมพิวเตอร์แห่งชาติ และมีความสนใจด้านการค้นหาความหมายของข้อมูลจีโนมเพื่องานประยุกต์ด้านการแพทย์และสาธารณสุข ด้วยเทคนิควิธีการทาง datamining, machine learning และ graph algorithms