



การพัฒนาโปรแกรมคอมพิวเตอร์สำหรับการออกแบบ
ข่ายงานเครื่องแลกเปลี่ยนความร้อนที่มีความยืดหยุ่น


โดย
มนตรี วงศ์ศรี

โครงการวิจัยเลขที่ 118-ERD-2533
ทุนส่งเสริมการวิจัยวิศวกรรมศาสตร์
ปี 2534

สถาบันวิจัยและพัฒนาคณะวิศวกรรมศาสตร์
คณะวิศวกรรมศาสตร์
จุฬาลงกรณ์มหาวิทยาลัย
กรุงเทพฯ

สิงหาคม 2541

จพ
วศ 15
007282



สถาบันวิจัยและพัฒนาของคณะวิศวกรรมศาสตร์ไม่รับผิดชอบ
ต่อผลเสียใดๆ อันอาจเกิดจากการนำความคิดเห็นในเอกสาร
ฉบับนี้ไปใช้ ความคิดเห็นที่ปรากฏในเอกสารเป็นความคิดเห็น
ของผู้เขียนซึ่งไม่จำเป็นต้องเป็นความคิดเห็นของสถาบันฯ

จุฬาลงกรณ์มหาวิทยาลัย

รายงานวิจัยฉบับสมบูรณ์

เรื่อง



การพัฒนาโปรแกรมคอมพิวเตอร์สำหรับการออกแบบ
ข่ายงานเครื่องแลกเปลี่ยนความร้อนที่มีความยืดหยุ่น

โดย

มนตรี วงศ์ศรี

สถาบันวิทยบริการ

จุฬาลงกรณ์มหาวิทยาลัย

สิงหาคม 2541

บทคัดย่อ

RHEN เป็นซอฟต์แวร์ทางกราฟฟิคสำเร็จรูป สำหรับออกแบบข่ายงานเครื่องแลกเปลี่ยนความร้อนที่มีความยืดหยุ่น ซอฟต์แวร์นี้สามารถออกแบบข่ายงานเครื่องแลกเปลี่ยนความร้อนที่สามารถให้การดึงพลังงานกลับคืนมากที่สุดและ มีจำนวนหน่วยเครื่องแลกเปลี่ยนความร้อนที่น้อยที่สุด ข่ายงานที่ออกแบบสามารถทำงานได้ที่สภาวะที่ออฟติมัมแม้อุณหภูมิและอัตราไหลของกระแสกระบวนการเข้ามีการผันแปร การออกแบบใช้วิธีการกระสวนการจับคู่ และวิธีการส่งผ่านความแปรปรวนที่พัฒนาโดย Wongsri (1990) RHEN เขียนด้วยภาษา C++ และทำงานภายใต้ปฏิบัติการ DOS ลักษณะและฟังก์ชันที่สำคัญอื่นๆ ที่มีอยู่ใน RHEN คือ ตารางข้อมูล ตารางปัญหา ไดอะแกรมกริด เส้นโค้งคอมโพสิต เส้นโค้งแกรนด์คอมโพสิต และค่าใช้จ่ายของเครื่องแลกเปลี่ยนความร้อน นอกจากนี้ RHEN สามารถหาค่า ΔT_{\min} ที่ประหยัดค่าใช้จ่ายของข่ายงานมากที่สุด และหาคำตอบของข่ายงานที่เป็นไปได้เกือบทุกคำตอบ ข่ายงานคำตอบที่ได้ มีความยืดหยุ่นโดยไม่จำเป็นต้องดัดแปลงโครงสร้างของข่ายงานในภายหลังอีก

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

Abstract

RHEN is a graphical software package for synthesis resilient heat exchanger networks. The software uses optimal resilient heat exchanger network design methods developed by Wongsri (1990), i.e., the Match Pattern Design Method and the Disturbance Propagation Method. RHEN is able to design a heat exchanger network that delivers and maintains Maximum Energy Recovery and Minimum Number of Unit despite the variations of temperature and flowrate of input streams. It is written in C++ language, and run on a DOS machine. Other features and functions are data input table, problem table, grid diagram, composite curves, grand composite curves and the cost of heat exchanger calculation. RHEN can also optimize ΔT_{\min} for capital and operating costs. It can find almost all-possible network solutions. The solution networks obtained are resilient without further evolutionary development.



สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

กิตติกรรมประกาศ

ผู้วิจัยใคร่ขอแสดงความขอบคุณ ฝ่ายวิจัยคณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ที่ได้ให้ความสนับสนุนทุนวิจัย จำนวน 75,000 บาท และขอขอบคุณ ผู้ช่วยวิจัย นายณัฐพร ทรงศิริ ที่ได้ช่วยเขียนโปรแกรมคอมพิวเตอร์ จนเป็นผลงานที่สมบูรณ์ และนำเข้าประกวดโครงการสิ่งประดิษฐ์ ของจุฬาลงกรณ์มหาวิทยาลัย เมื่อ 14-18 ธันวาคม พ.ศ. 2538 และได้นำผลงานวิจัยนี้ไปเสนอในการประชุมทางวิชาการ (Joint Sixth National Chemical Engineering Conference and the Second International Chemical Engineering Conference :ซึ่งจัดที่กรุงเทพ เมื่อเดือนตุลาคม ปี 2538 ในชื่อเรื่อง RHEN: Resilient Heat Exchanger Network Designer และ ขอแสดงความขอบคุณ ฝ่ายวิจัยคณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อีกครั้งที่ไม่ยุติโครงการนี้



จุฬาลงกรณ์มหาวิทยาลัย

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	ก
บทคัดย่อภาษาอังกฤษ	ข
กิตติกรรมประกาศ	ค
สารบัญ	ง
บทที่ 1 บทนำ	1
1.1 คำนำ	1
1.2 วัตถุประสงค์	2
1.3 ขอบเขตงานวิจัย	2
1.4 ประโยชน์ที่ได้รับ	2
1.5 โครงสร้างของรายงาน	2
บทที่ 2 ผลงานที่ผ่านมา	3
2.1 วิธีการออกแบบข่ายงานเครื่องแลกเปลี่ยนความร้อนแบบไม่ยัดหุ่่น	3
2.2 การออกแบบข่ายงานเครื่องแลกเปลี่ยนความร้อนแบบยัดหุ่่น	5
บทที่ 3 การออกแบบข่ายงานเครื่องแลกเปลี่ยนความร้อน	8
3.1 อีวริสติกสำหรับการวิเคราะห์ข่ายงานเครื่องแลกเปลี่ยนความร้อน	8
3.2 กระบวนการจับคู่ (Match Patterns)	10
3.3 กระบวนการจับคู่ของข่ายงานแบบยัดหุ่่น	14
3.4 ชนิดของความแปรปรวน	17
3.5 สภาวะการออกแบบ	20
3.6 พารามิเตอร์ที่ใช้ในการคำนวณ	21
3.7 วิธีการส่งผ่านความแปรปรวน (Disturbance Propagation Method, DPM)	24
3.8 การหาค่า ΔT_{min} ที่เหมาะสมกับข่ายงาน	27
บทที่ 4 การออกแบบโปรแกรมและการใช้งาน	39
4.1 ภาษาเชิงวัตถุ	39

4.2 การออกแบบคลาสในข่ายงาน	41
4.3 โฟลว์ชาร์ตการทำงานของโปรแกรม	48

บทที่ 5 การสาธิตการใช้โปรแกรม	53
-------------------------------	----

บทที่ 6 สรุปผลและวิจารณ์ผลงานวิจัย	66
6.1 สรุปผลการวิจัย	66
6.2 วิจารณ์ผลงานวิจัย	67

รายการอ้างอิง	68
---------------	----

ภาคผนวก ก บทความ เรื่อง RHEN: Resilient Heat Exchanger Network Designer ที่ได้นำเสนอในการประชุมทางวิชาการ	71
--	----

ภาคผนวก ข ตัวอย่างการออกแบบข่ายงานเครื่องแลกเปลี่ยนความร้อน ตัวอย่างการทดสอบโปรแกรม	84
--	----

ภาคผนวก ค สำเนาประกาศนียบัตร ในการเข้าร่วมงานสิ่งประดิษฐ์ ของจุฬาลงกรณ์มหาวิทยาลัย เมื่อ 14-18 ธันวาคม พ.ศ. 2538	105
---	-----

สถาบันวิทยบริการ

จุฬาลงกรณ์มหาวิทยาลัย

เลขหมู่	ค
	ค 15
เลขทะเบียน	007282
วัน,เดือน,ปี	19 พ.ค. 42

บทที่ 1

บทนำ



1.1 คำนำ

โดยทั่วไป ในกระบวนการผลิตของอุตสาหกรรมกระบวนการ เช่น โรงงานปิโตรเคมี โรงกลั่นน้ำมัน กระแสกระบวนการจะมีการเพิ่มอุณหภูมิหรือลดอุณหภูมิ โดยแลกเปลี่ยนความร้อนระหว่างกระแสกระบวนการด้วยกัน หรือใช้กระแสยูทิลิตี การแลกเปลี่ยนความร้อนระหว่างกระแสกระบวนการที่ต้องการเพิ่มความร้อนกับกระแสกระบวนการที่ต้องการลดความร้อด้วยตัวเอง แทนการกระแสยูทิลิตี เป็นการช่วยประหยัดพลังงาน และลดค่าใช้จ่ายทางด้านการปฏิบัติการ เครื่องแลกเปลี่ยนความร้อนหลายๆ เครื่องที่มีการเชื่อมโยงให้มีการแลกเปลี่ยนความร้อนระหว่างกระแสกระบวนการด้วยกัน เรียกว่า *ข่ายงานเครื่องแลกเปลี่ยนความร้อน* การออกแบบข่ายงานของเครื่องแลกเปลี่ยนความร้อนที่เหมาะสมนี้ เป็นหัวข้อวิจัยทางวิศวกรรมกระบวนการนับตั้งแต่กลางทศวรรษที่ 1970 เป็นต้นมา จนถึงปัจจุบัน

ในปัจจุบันได้มีซอฟต์แวร์ใช้สำหรับการออกแบบข่ายงานเครื่องแลกเปลี่ยนความร้อนโดยใช้วิธีการปกติไมเซชัน เช่น โปรแกรม MAGNETS (Floudas, 1986) โปรแกรมโดย Saboo et al. (1987) แต่วิธีการออกแบบข่ายงานเครื่องแลกเปลี่ยนความร้อนนี้ มิได้คำนึงถึงสภาวะการใช้งานจริงในอุตสาหกรรม ซึ่งอัตราการไหลและอุณหภูมิขาเข้าของกระแสกระบวนการมีค่าแปรปรวนไปจากค่าปกติเสมอ ทำให้บางครั้งข่ายงานเครื่องแลกเปลี่ยนความร้อนไม่สามารถทำให้อุณหภูมิขาออกของกระแสถึงค่าอุณหภูมิเป้าหมายที่ต้องการได้ ต่อมาจึงมีผู้พัฒนาวิธีการออกแบบข่ายงานแลกเปลี่ยนความร้อนแบบยืดหยุ่นขึ้น (Cerda et al., 1990 และ Wongsri, 1990) ข่ายงานแบบยืดหยุ่นนี้ สามารถทำงานได้อย่างมีประสิทธิภาพที่สามารถรักษาอุณหภูมิขาออกของกระแสทุกกระแสไว้ที่ค่าอุณหภูมิเป้าหมายได้ตลอดเวลา แม้ว่าอุณหภูมิขาเข้าและอัตราการไหลจะมีความแปรปรวนออกไปจากค่าปกติที่ใช้ในการออกแบบตั้งแต่แรกก็ตาม และนอกเหนือไปจากนั้นข่ายงานที่ออกแบบในวิธีนี้ยังคงใช้พลังงานจากยูทิลิตีน้อยที่สุดอีกด้วย

งานวิจัยนี้ ได้นำวิธีการออกแบบข่ายงานเครื่องแลกเปลี่ยนความร้อนแบบยืดหยุ่นของ Wongsri (1990) มาสร้างเป็นโปรแกรมคอมพิวเตอร์ ที่สามารถทำงานบนเครื่องคอมพิวเตอร์แบบบุคคลได้ และใช้ระบบปฏิบัติการ DOS ทั้งนี้เพื่อให้ผู้ใช้สามารถออกแบบข่ายงานเครื่องแลกเปลี่ยนความร้อนที่ประหยัดพลังงานดีที่สุดได้อย่างถูกต้อง รวดเร็ว และมีประสิทธิภาพ

1.2 วัตถุประสงค์

วัตถุประสงค์ของงานวิจัยนี้ เพื่อออกแบบและเขียนโปรแกรมการสร้างข่ายงานเครื่องแลกเปลี่ยนความร้อนแบบยัดหุ่่นที่สามารถทำงานได้ดี แม้ในสภาวะที่มีความแปรปรวนของอัตราการไหล และอุณหภูมิขาเข้า โดยใช้วิธีการออกแบบข่ายงานเครื่องแลกเปลี่ยนความร้อนของ Wongsri (1990) และพัฒนาโปรแกรมด้วยภาษาคอมพิวเตอร์แนววัตถุ C++ ที่สามารถทำงานบนเครื่องคอมพิวเตอร์แบบบุคคลได้ และใช้ระบบปฏิบัติการ DOS

1.3 ขอบเขตงานวิจัย

- ความแปรปรวนของอุณหภูมิขาเข้า หรือ ความแปรปรวนของอัตราการไหลของกระแส อยู่ในช่วงที่ทำให้อุณหภูมิพินช์ (Pinch) มีค่าเปลี่ยนแปลงอย่างต่อเนื่องเท่านั้น กล่าวคือไม่มีการกระโดดของค่าอุณหภูมิพินช์ (Pinch)
- กระแสกระบวนการไม่มีการเปลี่ยนสถานะ (phase)
- อุณหภูมิขาออก (อุณหภูมิเป้าหมาย) และความจุความร้อนจำเพาะของกระแสกระบวนการมีค่าคงที่

1.4 ประโยชน์ที่ได้รับ

โปรแกรมคอมพิวเตอร์สำเร็จรูป สำหรับการวิเคราะห์และออกแบบข่ายงานเครื่องแลกเปลี่ยนความร้อนแบบยัดหุ่่นทไคน ผู้ใช้สามารถออกแบบข่ายงานเครื่องแลกเปลี่ยนความร้อนที่เหมาะสมต่อกระบวนการผลิตของตนเองได้ในเวลารวดเร็ว และง่ายต่อการนำไปใช้งาน

1.5 โครงสร้างงานวิจัย

ในด้านเนื้อหาของงานวิจัย ได้แบ่งออกเป็น 5 บท บทที่ 1 กล่าวถึงคำนำ วัตถุประสงค์ ขอบเขตงานวิจัย ประโยชน์ที่ได้รับ และโครงสร้างงานวิจัย บทที่ 2 เป็นผลงานวิจัยข่ายงานเครื่องแลกเปลี่ยนความร้อนในอดีตที่ผ่านมา บทที่ 3 อธิบายถึงกระบวนการจับคู่ (matching patterns) ที่พัฒนาโดย Wongsri (1990) และ หลักการส่งผ่านความแปรปรวน (disturbance propagation technique) ที่พัฒนาโดย Wongsri (1990) และวิธีการหาค่าความแตกต่างของอุณหภูมิน้อยที่สุดที่เหมาะสมสำหรับการแลกเปลี่ยนความร้อน (ΔT_{min} ที่ออปติ멈ของเครื่องแลกเปลี่ยนความร้อนแบบสวนทาง) สำหรับบทที่ 4 กล่าวถึงการออกแบบโปรแกรมคอมพิวเตอร์ และตัวอย่างการใช้งาน บทที่ 5 กล่าวถึงบทสรุปและวิจารณ์ผลงานวิจัย ภาคผนวก ก เป็นบทความ เรื่อง RHEN: Resilient Heat Exchanger Network Designer ที่ได้นำเสนอในการประชุมทางวิชาการ ภาคผนวก ข เป็น ตัวอย่างการออกแบบข่ายงานเครื่องแลกเปลี่ยนความร้อน และ ภาคผนวก ค เป็น ประกาศนียบัตร ในการเข้าร่วมงานสิ่งประดิษฐ์ของจุฬาลงกรณ์มหาวิทยาลัย เมื่อ 14-18 ธันวาคม พ.ศ. 2538

บทที่ 2

ผลงานวิจัยที่ผ่านมา

ในบทนี้จะกล่าวถึงผลงานวิจัยที่ผ่านมาเกี่ยวกับวิธีการออกแบบข่ายงานเครื่องแลกเปลี่ยนความร้อน สำหรับวิธีการออกแบบข่ายงานเครื่องแลกเปลี่ยนความร้อนแบบยัดหุ่่นของ Wongsri (1990) ที่ใช้ในงานวิจัยนี้ จะได้กล่าวไว้ในบทที่ 3

2.1 วิธีการออกแบบข่ายงานเครื่องแลกเปลี่ยนความร้อนแบบไม่ยัดหุ่่น

ขั้นตอนของการออกแบบข่ายงานเครื่องแลกเปลี่ยนความร้อนแบบไม่ยัดหุ่่น แบ่งออกได้เป็น 2 ขั้นตอน คือ การวิเคราะห์เป้าหมายของข่ายงาน และการออกแบบข่ายงาน

2.1.1 การวิเคราะห์เป้าหมายของข่ายงาน

ขั้นตอนนี้จะกำหนดเป้าหมายสำหรับการออกแบบข่ายงานก่อน คือ ค่าพลังงานกลับคืนมากที่สุด (The Maximum Energy Recovery, MER) และจำนวนเครื่องแลกเปลี่ยนความร้อนน้อยที่สุด (The Minimum Number of Units, MNU)

ค่าพลังงานกลับคืนมากที่สุด (MER) คือ พลังงานความร้อนมากที่สุดที่กระแสน้ำในกระบวนการผลิตใช้แลกเปลี่ยนความร้อนกันเอง สามารถหาได้จากไดอะแกรมอุณหภูมิและเอนทาลปี (Hohmann, 1971), วิธีตารางปัญหา (Problem Table Method) พัฒนาโดย Hohmann (1971), และ Linnhoff et al. (1982) หรือเทคนิคการโปรแกรมทางคณิตศาสตร์ (Cerda และ Westerberg, 1983) สำหรับวิธีเหล่านี้ นอกจากหาค่า MER ได้แล้ว ยังสามารถหาอุณหภูมิพินช์ได้ด้วย อุณหภูมิพินช์ มีความสำคัญในการแก้ปัญหาเพื่อให้ได้ MER โดยแบ่งการแก้ข่ายงานของปัญหาออกเป็นสองข่ายงานย่อย หรือมากกว่า ข่ายงานย่อยหนึ่งจะกระแสน้ำให้ความร้อน (heating section) และอีกข่ายงานย่อยหนึ่งจะใช้กระแสน้ำที่ดึงความร้อนออกมา (cooling section) การแก้ปัญหาของข่ายงานย่อยทั้งสองจะเป็นอิสระต่อกัน ข่ายงานที่สมบูรณ์จะได้จากการนำข่ายงานย่อยทั้งสองมารวมกัน การออกแบบข่ายงานโดยแบ่งการแก้ปัญหาออกเป็นข่ายงานเหนือจุดพินช์และข่ายงานใต้จุดพินช์ สามารถทำให้ใช้พลังงานจากยูทิลิตี้น้อยที่สุดได้ตามที่คำนวณไว้ (แต่ไม่จำเป็นที่ทุก ๆ ข่ายงานที่ออกแบบจะมีอุณหภูมิพินช์เสมอไป)

จำนวนเครื่องแลกเปลี่ยนความร้อนน้อยที่สุด (MNU) จำนวนเครื่องแลกเปลี่ยนความร้อนน้อยที่สุดในที่นี้จะรวมฮีตเตอร์และคูลเลอร์ในข่ายงานด้วย สามารถหาได้โดยสมการ (Hohmann, 1971)

$$N_{min} = N_h + N_c + N_{hu} + N_{cu} - 1 \quad (2.1)$$

เมื่อ N_h และ N_c คือ จำนวนของกระแสน้ำร้อน และกระแสน้ำเย็นในกระบวนการผลิต

N_{hu} และ N_{cu} คือ จำนวนกระแสนอน และกระแสนอนจากหน่วยปฏิบัติการ

สำหรับข่ายงานที่มีอุณหภูมิพินช์ สมการดังกล่าวสามารถใช้ได้ทั้งข่ายงานเหนือและใต้จุดพินช์

(Linnhoff et. al., 1982)

ในงานวิจัยนี้จะใช้วิธีตารางปัญหาและสมการ 2.1 ในการวิเคราะห์เป้าหมายของข่ายงาน

2.1.2 การหาโครงสร้างของข่ายงาน

การออกแบบข่ายงานเครื่องแลกเปลี่ยนความร้อน แบ่งได้เป็นสองวิธี คือ วิธีออปติไมเซชัน (Optimization method) และวิธีฮิวริสติก (Heuristic method)

ก. วิธีออปติไมเซชัน

วิธีนี้ข้อมูล เงื่อนไขและปัญหาการออกแบบ จะถูกแปลงเป็นโมเดลทางคณิตศาสตร์ ซึ่งในบางปัญหาที่ซับซ้อนจะทำได้ยาก หรืออาจทำไม่ได้เลย ตัวอย่างของวิธีนี้ ได้แก่ Transportation problem และ Transshipment models

ข. วิธีฮิวริสติก

เนื่องจากยังไม่มีทฤษฎีที่สมบูรณ์ใดๆ สำหรับหาโครงสร้างของข่ายงานเครื่องแลกเปลี่ยนความร้อน วิธีฮิวริสติกเป็นวิธีการหาที่ใช้ความรู้ประสบการณ์ หรือ สหัชญาณ (intuition) ของผู้ออกแบบ ไม่จำเป็นต้องเป็นความรู้ที่ครบถ้วนสมบูรณ์ แต่ต้องใช้แก้ปัญหาได้ผลในระดับที่พอใจ วิธีนี้จึงมีข้อดีที่ไม่จำเป็นต้องทราบทฤษฎีที่สมบูรณ์ หรือแปลปัญหาการออกแบบเป็นสมการคณิตศาสตร์ในการเหมือนวิธีออปติไมเซชัน และนอกจากนั้นโดยการอาศัยเทคนิคการค้นหาที่พัฒนายังสามารถเขียนด้วยภาษาคอมพิวเตอร์ขั้นสูงได้โดยตรง จึงทำให้การค้นหาโครงสร้างของข่ายงานที่ประหยัดพลังงานมากที่สุดเป็นไปได้อย่างรวดเร็ว และมีประสิทธิภาพกว่า ตัวอย่างของวิธีฮิวริสติก เช่น Graphs หรือ Diagrams, Temperature Interval method, Evolutionary design methods, Pinch method, Enumeration methods และ Ruled-based methods เป็นต้น อย่างไรก็ตาม ข่ายงานแบบไม่ยึดหยุ่นที่ออกแบบโดยวิธีเหล่านี้ จะไม่สามารถรักษาอุณหภูมิขาออกของกระแสต่างๆ ให้อยู่ที่อุณหภูมิเป้าหมายได้ เพราะเมื่อมีความแปรปรวนของอุณหภูมิขาเข้าและอัตราการไหล แม้ว่าจะแก้ไขแบบโดยการเพิ่มพื้นที่ของเครื่องแลกเปลี่ยนความร้อนให้มีค่าเปลี่ยนไปมากเพียงใดก็ตาม ก็ไม่สามารถแก้ปัญหาเรื่องอุณหภูมิเป้าหมายได้ แต่อัลกอริทึมที่พัฒนาโดย Wongsri (1990) สามารถแก้ปัญหานี้ได้ โดยอัลกอริทึมที่พัฒนานี้สามารถเลือกโครงสร้างของข่ายงานที่เหมาะสมได้ตั้งแต่แรก ข่ายงานที่มีคุณสมบัติทำงานได้ในภาวะที่มีความแปรปรวนดังกล่าวนี้เรียกว่า ข่ายงานเครื่องแลกเปลี่ยนความร้อนแบบยืดหยุ่น (Resilient Heat Exchanger Network หรือ RHEN)

โดยปกติทั่วไปแล้ว ผู้ออกแบบจะออกแบบโครงสร้างของข่ายงานเครื่องแลกเปลี่ยนความร้อนได้หลายโครงสร้างด้วยกัน และนำโครงสร้างทั้งหมดมาทดสอบว่า โครงสร้างใดที่สามารถทำงานได้ในภาวะความแปรปรวนของอุณหภูมิขาเข้า และอัตราการไหลในช่วงที่กำหนด ถ้าโครงสร้างใดสามารถทำให้อุณหภูมิขาออกของกระแส ถึงค่าอุณหภูมิเป้าหมายที่ต้องการได้ ก็จะเลือกโครงสร้างนั้นไปใช้งาน

จะสังเกตได้ว่า ผู้ออกแบบไม่สามารถกำหนดหาโครงสร้างที่มีความยืดหยุ่นได้ตั้งแต่เริ่มออกแบบ แต่จะหาโครงสร้างข้างานปกติมาหลายๆ โครงสร้าง แล้วนำมาคัดเลือกและทดสอบความยืดหยุ่น และตัดปลงเพิ่มเติมในภายหลัง ซึ่งเป็นการแก้ปัญหาแบบแอดฮอค (Ad hoc method) ไม่สามารถแก้ปัญหาได้ทุกกรณี

2.2 การออกแบบข้างานเครื่องแลกเปลี่ยนความร้อนแบบยืดหยุ่น

ในปี 1983 Saboo และ Morari ได้แบ่งปัญหาการออกแบบข้างานแบบยืดหยุ่นออกเป็นสองประเภท ตามลักษณะความแปรปรวนที่ส่งผลต่อตำแหน่งของอุณหภูมิพินซ์ สำหรับกรณีที่มีความแปรปรวนของอุณหภูมิขาเข้าเพียงเล็กน้อย และทำให้การเปลี่ยนแปลงตำแหน่งของพินซ์เปลี่ยนแปลงไปในลักษณะที่ต่อเนื่อง จะเรียกว่าข้างานนั้นเป็นปัญหาแบบคลาส I (Class I) แต่ในกรณีที่มีความแปรปรวนของอุณหภูมิขาเข้า หรือความแปรปรวนเนื่องจากอัตราการไหลมาก อุณหภูมิพินซ์จะเปลี่ยนแปลงไปอย่างไม่ต่อเนื่อง และจะเรียกว่าเป็นปัญหาแบบคลาส II (Class II)

Saboo และ Morari (1985) ได้เสนอการวัดความยืดหยุ่นของข้างานที่ออกแบบโดยใช้ดัชนีความยืดหยุ่น (Resiliency Index, RI) ซึ่งจะบอกถึงปริมาณความแปรปรวนสูงสุดที่ข้างานสามารถรับได้โดยไม่มีผลกระทบต่อเป้าหมายของข้างาน

$$RI = \min\{L_{i,max}\} \quad (2.2)$$

$$L_{i,max} = \min\{L_o, R_o\} \quad (2.3)$$

$$L_i = W_i \Delta T_i^s \quad (2.4)$$

เมื่อ W_i คือ ผลคูณของอัตราการไหลกับความจุความร้อนจำเพาะ, ΔT_i^s คือ ความแปรปรวนของอุณหภูมิขาเข้าจากค่าที่ภาวะปกติ, L_o และ R_o คือค่าความแปรปรวน และค่าพารามิเตอร์ความยืดหยุ่น (Resiliency Parameter) ตามลำดับ

Swaney และ Grossmann (1985) ได้เสนอพารามิเตอร์ตัวใหม่สำหรับวัดความยืดหยุ่นของข้างาน คือดัชนีความอ่อน โยน (Flexibility Index, FI) ซึ่งจะวัดขนาดหรือขอบเขตของภาวะที่ข้างานยังสามารถทำงานได้ตามปกติ

Calandranis และ Stephanopoulos (1986) ได้เสนอการวิเคราะห์วิธีการออกแบบโดยตรวจสอบ Nonconvexity ของข้างาน สำหรับกรณีพินซ์ไม่ต่อเนื่อง และความแปรปรวนของอุณหภูมิ เป็นต้น

Saboo และ Morari (1987) ใช้วิธีการโปรแกรมทางคณิตศาสตร์ในการวิเคราะห์ความยืดหยุ่นของข้างาน โดยใช้โปรแกรมเชิงเส้นและ โปรแกรมเชิงเส้นหลายตัวแปร

2.1.1 การออกแบบข้างานด้วยวิธีการรวมข้างาน (Combination Design)

สำหรับการออกแบบข้างานแบบยืดหยุ่นด้วยวิธีนี้ จะคำนึงถึงภาวะการทำงานในกรณีที่ความแปรปรวนที่จุดขีดสุด (Extreme points) โดยการหาค่าตอบจะนำพารามิเตอร์ความยืดหยุ่นไปทดสอบข้างานที่ทำงานในภาวะดังกล่าวด้วย ยกตัวอย่างวิธีเหล่านี้เช่น Worst cases design, Corner point

theorem, Multiperiod Operation Design, Superstructure Design, Computer design package, Downpath technique และ Sensitivity table เป็นต้น วิธีต่างๆ เหล่านี้มีข้อเสียที่ ในบางกรณีเราไม่สามารถหาสถานะการทำงานที่แย่ที่สุดเพื่อใช้ทดสอบได้, เสียเวลาในการลองผิดลองถูกค่อนข้างมาก และวิธีนี้ไม่รับรองว่าจะให้ช่างงานมีจำนวนเครื่องที่น้อยที่สุด และได้ MER หรือไม่

2.2.2 วิธีการออกแบบช่างงานแบบยืดหยุ่นโดยตรง

วิธีนี้จะทดสอบความยืดหยุ่นของช่างงานในขณะออกแบบโครงสร้างได้โดยตรง โดยช่างงานที่ได้ในขั้นตอนสุดท้ายไม่ต้องนำมาทดสอบความยืดหยุ่นอีก สำหรับตัวอย่างวิธีทางออปติไมเซชันที่สำคัญ คือ

การออปติไมซ์หลายวัตถุประสงค์ (Multiple-objective Optimization) วิธีนี้แบ่งการวิเคราะห์ได้เป็นสองขั้นตอน ดังนี้

- (1) การโปรแกรมไม่เชิงเส้นแบบไม่จำกัด (Infinite nonlinear programming) สำหรับการส่งผ่านความร้อนที่มีค่าเปลี่ยนแปลงจากแหล่งให้ความร้อนไปยังแหล่งรับความร้อน
- (2) วิธีการออปติไมซ์แบบมีวัตถุประสงค์หลายอย่างหรือรวมวัตถุประสงค์ (Multiple objective optimization methods) สำหรับวัดความยืดหยุ่นและสร้างเป็นฟังก์ชันเพื่อทำการออปติไมซ์ รวมทั้งการทดสอบทางด้านค่าใช้จ่ายของช่างงานด้วย

ค่าใช้จ่ายในการติดตั้งหรือการสร้างช่างงานเครื่องแลกเปลี่ยนความร้อน, ค่าใช้จ่ายในการดำเนินการผลิต และ การทดสอบความยืดหยุ่นนั้น จะถูกนำมาเขียนในรูป Multi-objective function ได้ดังนี้

$$\max [f_1(x), f_2(x), f_3(x)] \text{ subject to: } g_i(x) \leq 0, i = 1, 2, \dots, m \quad (2.5)$$

Cerda และคณะ (1990) ได้คิดค้นเทคนิคนี้ เพื่อที่จะรวมการทดสอบความยืดหยุ่นเข้าไปในโมเดลคณิตศาสตร์ของการแก้ปัญหาการออกแบบด้วยเทคนิคออปติไมเซชัน

เทคนิคการแก้ปัญหาการออกแบบช่างงานที่มีความยืดหยุ่น

กระแสในกระบวนการผลิตจะถูกแบ่งเป็นสองประเภทคือ

1. กระแสถาวร เป็นกระแสที่เข้ามามีอุณหภูมิขาเข้าต่ำที่สุด โดยไม่คำนึงถึงความแปรปรวน
2. กระแสชั่วคราว เป็นกระแสที่เข้ามาโดยมีอุณหภูมิขาเข้าที่เปลี่ยนแปลงด้วย

จากนั้นทำการแก้ปัญหาโดยใช้หลักดังนี้

1. แลกเปลี่ยนความร้อนระหว่างกระแสถาวรให้มากที่สุด

2. ส่งผ่านความร้อนจากกระแสร้อนชั่วคราว ให้ไปยังกระแสเย็นถาวรที่เหลือน้อยให้มากที่สุด หลังจากนั้นก็กำหนดกระแสเย็นถาวรที่ปริมาณความร้อนเหลือน้อยไปเป็นกระแสชั่วคราว
3. แลกเปลี่ยนความร้อนระหว่างกระแสชั่วคราวให้มากที่สุด

การรวมเป้าหมายทั้งสามเข้าด้วยกัน จะให้นำหนักต่างๆกันตามความสำคัญ สำหรับวิธีทางฮิวริสติกของ Wongsri (1990) จะกล่าวไว้ในบทที่ 3 ต่อไป

งานวิจัยของจุฬาลงกรณ์มหาวิทยาลัยที่ผ่านมา คือของ ประทีป อรุณวัฒนมงคล (1992) ได้ใช้กระบวนการจับคู่ของ Wongsri (1990) เขียนโปรแกรมการออกแบบข่ายงานเครื่องแลกเปลี่ยนความร้อนแบบไม่ยึดหยุ่น โดยไม่ได้คำนึงถึงความแปรปรวนของอุณหภูมิขาเข้าและ อัตราการไหลของกระแสนอกจากนี้โปรแกรมเขียนด้วยภาษาเบสิก ซึ่งมีข้อจำกัดด้านโครงสร้างของข้อมูล และทำงานได้ไม่เร็วนัก

ในงานวิจัยนี้ ใช้วิธีกระบวนการจับคู่ และวิธีส่งผ่านความแปรปรวนของ Wongsri (1990) ออกแบบข่ายงานเครื่องแลกเปลี่ยนความร้อนแบบยึดหยุ่น โดยพิจารณาในกรณีช่วงความแปรปรวนของอัตราการไหล และอุณหภูมิขาเข้าของกระแสค่าที่ไม่ทำให้การเปลี่ยนแปลงของอุณหภูมิพินซ์เกิดการกระโดด หรือเป็นปัญหาแบบคลาส I

โปรแกรมคอมพิวเตอร์ที่ได้นี้ ช่วยให้ผู้ใช้สามารถออกแบบข่ายงานเครื่องแลกเปลี่ยนความร้อนแบบยึดหยุ่นได้อย่างรวดเร็ว โดยภาษาคอมพิวเตอร์สำหรับการออกแบบโปรแกรมที่ใช้คือ ภาษา C++ ซึ่งเป็นภาษาที่รวมเอาความสามารถของภาษา C ที่มีความคล่องตัว, ความเร็วสูง และความสามารถใหม่เพิ่มขึ้นมาคือ ความเป็นโปรแกรมเชิงวัตถุ (Object-oriented program) ซึ่งเป็นการออกแบบโครงสร้างการเขียนโปรแกรมที่มีประสิทธิภาพสูง สามารถแก้ไขหรือนำมาพัฒนาต่อไปในภายหลังได้ง่าย โดยการออกแบบโปรแกรมเชิงวัตถุนี้จะกล่าวไว้ในบทที่ 4

จุฬาลงกรณ์มหาวิทยาลัย



บทที่ 3

วิธีการออกแบบข่ายงานเครื่องแลกเปลี่ยนความร้อนแบบยัดหยุ่น

หลักการสำคัญในการออกแบบข่ายงานแบบยัดหยุ่นโดยตรงตามวิธีของ Wongsri (1990) คือ การใช้กระสวนการจับคู่ (Match patterns) และวิธีการส่งผ่านความแปรปรวน (Disturbance propagation Method)

3.1 ฮีริสติกสำหรับการวิเคราะห์ข่ายงานเครื่องแลกเปลี่ยนความร้อน

ผลงานวิจัยเกี่ยวกับฮีริสติกสำหรับการออกแบบข่ายงานเครื่องแลกเปลี่ยนความร้อน ได้ถูกตีพิมพ์มาเป็นจำนวนมาก โดยมีวัตถุประสงค์เพื่อใช้พลังงานจากหน่วยยูนิตให้ให้น้อยที่สุด และประหยัดค่าใช้จ่ายการสร้างข่ายงานเครื่องแลกเปลี่ยนความร้อนให้มากที่สุด เช่น Masso และ Rudd (1969), Ponton และ Donalson (1974), Rathore และ Powers (1975), Linnhoff และ Hindmarsh (1983), Jezowski และ Hahne (1986), Metha และ Fan (1988) เป็นต้น

ตัวอย่างฮีริสติกที่สำคัญในการออกแบบข่ายงานมีดังนี้

ฮีริสติกสำหรับการประหยัดค่าใช้จ่ายในการสร้างเครื่องแลกเปลี่ยนความร้อนในข่ายงาน

C.1 ให้การจับคู่กระแสเพื่อแลกเปลี่ยนความร้อนทุกครั้งนั้น สามารถกำจัดกระแสใดกระแสหนึ่งหรือทั้งสองกระแสออกไปเสมอ เพื่อให้ข่ายงานแลกเปลี่ยนความร้อนมีจำนวนเครื่องแลกเปลี่ยนความร้อนน้อยที่สุด

C.2 ถ้าเป็นปัญหาทางด้านความร้อน ให้แลกเปลี่ยนความร้อนทางด้านคู่กระแสที่ให้กระแสมีความร้อนหลงเหลืออยู่ที่ปลายด้านอุณหภูมิต่ำ และถ้าเป็นปัญหาทางด้านความเย็น ให้แลกเปลี่ยนความร้อนระหว่างคู่ของกระแส โดยให้กระแสมีความร้อนหลงเหลืออยู่ที่ปลายด้านอุณหภูมิสูง การแลกเปลี่ยนความร้อนลักษณะนี้เรียกว่า การจับคู่แบบที่ 1 (First Type Match โดย Rathore และ Powers, 1975)

สำหรับการจับคู่แบบที่ 2 (Second Type Match) นั้น สำหรับปัญหาทางด้านความร้อนจะให้ความร้อนหลงเหลืออยู่ที่ปลายด้านอุณหภูมิสูง กรณีเป็นปัญหาทางด้านความเย็นจะให้ความร้อนหลงเหลืออยู่ที่ปลายด้านอุณหภูมิต่ำ

C.3 ให้แลกเปลี่ยนความร้อนระหว่างคู่กระแสที่มีความร้อนสูงก่อน

ฮีริสติกสำหรับทำให้ข่ายงานประหยัดพลังงานความร้อน

E.1 แบ่งการออกแบบข่ายงาน (เมื่อมีจุดพินช์) ออกเป็นปัญหาย่อยๆ และแก้ปัญหาย่อยๆ อย่างอิสระ (Linnhoff และ Hindmarsh, 1983)

E.2 ไม่ส่งผ่านความร้อนข้ามจุดพินช์

E.3 ไม่ใช่คูเลเตอร์สำหรับการออกแบบเหนือจุดพินช์ (ปัญหาทางด้านความร้อน)

E.4 ไม่ใช่ฮีตเตอร์สำหรับการออกแบบใต้จุดพินช์ (เป็นปัญหาทางด้านความเย็น)

E.5 ควรใช้การจับคู่แบบที่ 2 เนื่องจากให้กระแสที่มีความร้อนหลงเหลือมีอุณหภูมิต่ำ ทำให้ประหยัดการใช้พลังงานจากหน่วยยูทิลิตีมากกว่า (Ponton และ Donaldson, 1974)

E.6 สำหรับปัญหาด้านความร้อน ถ้ามีกระแสร้อนกระแสเดียวที่ทำให้กระแสเย็นมีอุณหภูมิถึงค่าเป้าหมายได้ ให้แลกเปลี่ยนความร้อนระหว่างกระแสคู่นี้ก่อนเป็นอันดับแรก ส่วนถ้าเป็นปัญหา ทางด้านความเย็นจะเป็นในลักษณะทางตรงกันข้าม

อิวิริตติกสำหรับการแลกเปลี่ยนความร้อนระหว่างคู่กระแส

N.1 สำหรับปัญหาทางด้านความร้อนนั้น ควรเลือกจับคู่กระแสที่ความร้อนของกระแสเย็นมีค่ามากกว่า ความร้อนของกระแสร้อน

N.2 สำหรับปัญหาทางด้านความเย็น ควรเลือกจับคู่กระแสที่ความร้อนของกระแสร้อนมีค่ามากกว่า ความร้อนของกระแสเย็น

N.3 สำหรับปัญหาด้านความร้อน ควรเลือกการแลกเปลี่ยนความร้อนระหว่างคู่กระแสสำหรับ กระแส ร้อนที่มีความร้อนหลงเหลือน้อยกว่า หรือเท่ากับค่าความเย็นที่มีค่าน้อยที่สุดที่จะใช้จากหน่วยยูทิลิตี

N.4 สำหรับปัญหาด้านความเย็น ควรเลือกการแลกเปลี่ยนความร้อนระหว่างคู่กระแสสำหรับกระแสเย็น ที่มีความร้อนหลงเหลือน้อยกว่า หรือเท่ากับค่าความร้อนที่มีค่าน้อยที่สุดที่จะใช้จากหน่วยยูทิลิตี

กฎทางเทอร์โมไดนามิกส์

T.1 สำหรับปัญหาทางด้านความร้อน ให้จับคู่ระหว่างกระแสร้อน และกระแสเย็น ถ้าอุณหภูมิเป้าหมาย ของกระแสร้อนต่างจากอุณหภูมิขาเข้าของกระแสเย็นมากกว่า หรือเท่ากับ ΔT_{min} หรือสำหรับกรณีที่ค่า ผลคูณของอัตราการไหล และความจุความร้อนจำเพาะของกระแสร้อนน้อยกว่า หรือเท่ากับค่าดังกล่าว ของกระแสเย็น

T.2 สำหรับปัญหาทางด้านความเย็น ให้จับคู่ระหว่างกระแสร้อน และกระแสเย็น ถ้าอุณหภูมิขาเข้า ของ กระแสร้อนต่างจากอุณหภูมิเป้าหมายของกระแสเย็นมากกว่า หรือเท่ากับ ΔT_{min} หรือสำหรับกรณีที่ค่า ผลคูณของอัตราการไหล และความจุความร้อนจำเพาะของกระแสร้อนมากกว่า หรือเท่ากับค่าดังกล่าว ของกระแสเย็น

T.3 สำหรับการจับคู่ที่มีเงื่อนไขต่างจากข้างต้น ให้ตรวจสอบค่าอุณหภูมิที่แตกต่างในการแลกเปลี่ยนความร้อนระหว่างคู่กระแส ให้มีค่ามากกว่า ΔT_{min} เสมอ

3.2 กระบวนการจับคู่ (Match patterns)

จากฮิวริสติกที่กล่าวมา จะสามารถนำมาออกแบบเป็นกระบวนการจับคู่ เพื่อใช้สำหรับสร้างข่ายงานแลกเปลี่ยนความร้อนได้ง่ายขึ้น และเนื่องจากการแลกเปลี่ยนความร้อนระหว่างคู่ของกระแส มีโอกาสแลกเปลี่ยนกันได้สี่แบบด้วยกัน จึงแบ่งแบบจับคู่ออกเป็นคลาสต่างๆ ได้สี่คลาส ดังนี้

คลาสเอ (Class A) - ปริมาณความร้อนของกระแสเย็นมีค่าสูงกว่าค่าของกระแสร้อน และการแลกเปลี่ยนความร้อนจะทำให้ปลายกระแสที่มีอุณหภูมิต่ำ โดยหลังการแลกเปลี่ยนความร้อนแล้วความร้อนของกระแสร้อนจะหมด

คลาสบี (Class B) - ปริมาณความร้อนของกระแสเย็นมีค่าต่ำกว่าค่าของกระแสร้อน และการแลกเปลี่ยนความร้อนทำให้ปลายกระแสที่มีอุณหภูมิสูง โดยหลังการแลกเปลี่ยนความร้อนแล้วความร้อนของกระแสเย็นจะหมด

คลาสซี (Class C) - ปริมาณความร้อนของกระแสเย็นมีค่าต่ำกว่าค่าของกระแสร้อน และการแลกเปลี่ยนความร้อนทำให้ปลายกระแสที่มีอุณหภูมิต่ำ

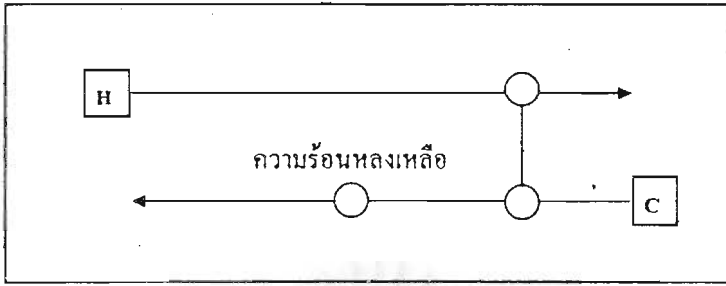
คลาสดี (Class D) - ปริมาณความร้อนของกระแสเย็นมีค่าสูงกว่าค่าของกระแสร้อน และการแลกเปลี่ยนความร้อนทำให้ปลายกระแสที่มีอุณหภูมิสูง โดยหลังการแลกเปลี่ยนความร้อนแล้ว ความร้อนของกระแสร้อนจะหมดไป (ดูรูปที่ 3.1 และตารางที่ 3.1)

จากฮิวริสติก N3 และ N4 ยังสามารถแบ่งคลาสออกเป็นคลาสย่อยได้อีกคือ

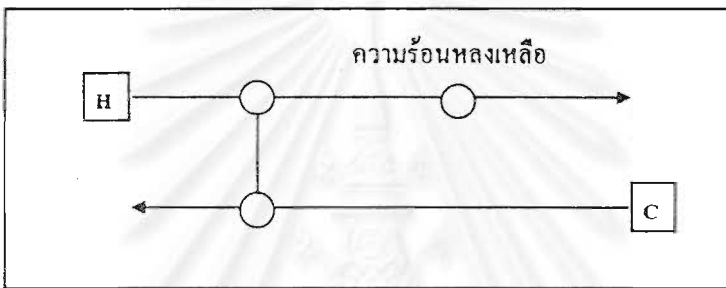
1. คลาสย่อย AH ใช้สำหรับปัญหาทางด้านความร้อนเมื่อความร้อนหลงเหลือของกระแสเย็นในการแลกเปลี่ยนหลังการจับคู่มีค่าน้อยกว่า หรือเท่ากับค่าความร้อนน้อยที่สุดที่ต้องการรับจากหน่วย ยูทิลิตี้ $(Q_{H,min})$
2. คลาสย่อย BC ใช้สำหรับปัญหาทางด้านความเย็น เมื่อความร้อนหลงเหลือของกระแสร้อนในการแลกเปลี่ยนหลังการจับคู่ มีค่าน้อยกว่าหรือเท่ากับค่าความร้อนน้อยที่สุดที่ต้องการให้แก่อนหน่วย ยูทิลิตี้ $(Q_{C,min})$

เมื่อความร้อนหลงเหลือของกระแสหนึ่งๆ หลังการจับคู่แลกเปลี่ยนความร้อน มีค่าน้อยกว่าความร้อนที่ต้องการรับ (หรือให้) จากหน่วย ยูทิลิตี้ แล้วจะสามารถสร้างเป็นฮีตเตอร์ (หรือคูลเลอร์) ได้ทันทีดังนั้นจึงควรให้กระบวนการจับคู่หรือแพทเทิร์น AH หรือ BC เป็นอันดับแรก นอกจากนี้ยังสามารถใช้ความแตกต่างระหว่างผลคูณของอัตราการไหลและความจุความร้อนจำเพาะของกระแส ออกแบบกระบวนการจับคู่เพิ่มขึ้นได้อีกเช่น จากฮิวริสติกที่กล่าวมา สำหรับปัญหาทางด้านความร้อน ควรจับคู่ระหว่างกระแสที่ผลคูณของอัตราการไหลและความจุความร้อนจำเพาะของกระแสเย็นมีค่ามากกว่าหรือเท่ากับค่าของกระแสร้อน ตามเงื่อนไขจะได้แพทเทิร์น A[H] แต่ถ้าค่าของกระแสเย็นน้อยกว่าของกระแสร้อนจะให้แพทเทิร์น A[C] สำหรับปัญหาทางด้านความเย็น จากฮิวริสติกกล่าวมาควรจับคู่ระหว่างกระแสที่ผลคูณของอัตราการไหลและความจุความร้อนจำเพาะของกระแสร้อนมีค่ามากกว่าหรือเท่ากับค่าของกระแสเย็นซึ่งก็คือ แพทเทิร์น B[C] แต่ถ้าค่าดังกล่าวของกระแสร้อนน้อยกว่าค่า

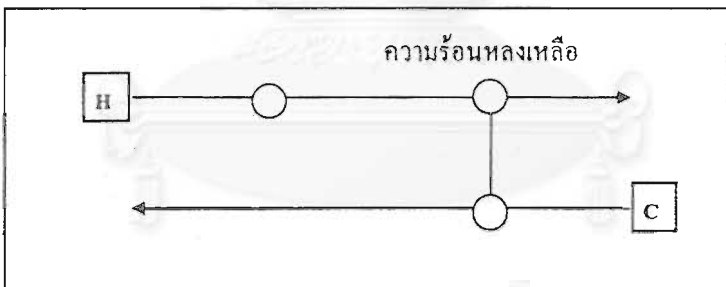
กลาสเอ



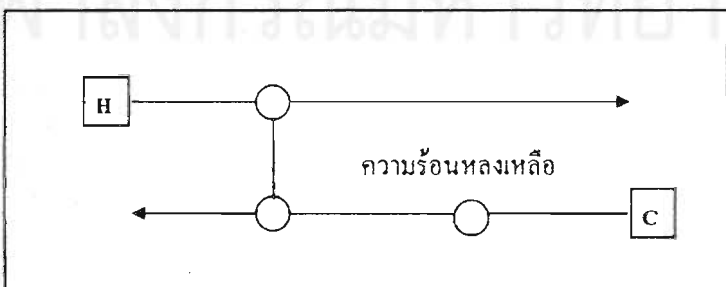
กลาสบี



กลาสซี

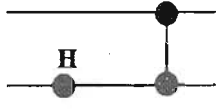
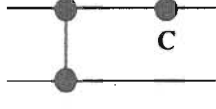
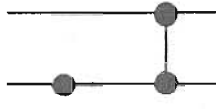
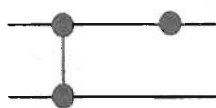
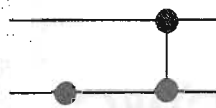
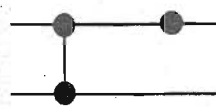


กลาสดี

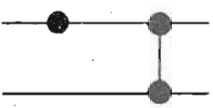
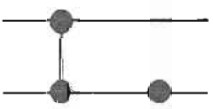
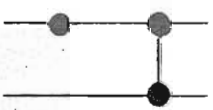



รูปที่ 3.1 แสดงลักษณะการแลกเปลี่ยนความร้อนกลาสต่าง ๆ

ตารางที่ 3.1 โอเปอเรเตอร์ของกระบวนการจับคู่คลาสเอ และ คลาสบี

โอเปอเรเตอร์	เงื่อนไข	ค่าที่คำนวณใหม่
 <p>แพทเทิร์น AH</p>	$T_H^i \geq T_C^s, L_H \leq L_C$ $T_H^s \geq T_C^s + L_H W_C^{-1}$ $L_C - L_H \leq Q_{min}^{heating}$	$T_C^s = T_C^s + L_H W_C^{-1}$ $L_C = L_C - L_H$
 <p>แพทเทิร์น BC</p>	$T_H^s \geq T_C^i, L_H \geq L_C$ $T_C^s \leq T_H^s - L_C W_H^{-1}$ $L_H - L_C \leq Q_{min}^{cooling}$	$T_H^s \leq T_H^s - L_C W_H^{-1}$ $L_H = L_H - L_C$
 <p>แพทเทิร์น A[H]</p>	$T_H^i \geq T_C^s$ $L_H \leq L_C$ $W_H \leq W_C$	$T_C^s = T_C^s + L_H W_C^{-1}$ $L_C = L_C - L_H$
 <p>แพทเทิร์น B[C]</p>	$T_H^s \geq T_C^i$ $L_H \geq L_C$ $W_H \geq W_C$	$T_H^s \leq T_H^s - L_C W_H^{-1}$ $L_H = L_H - L_C$
 <p>แพทเทิร์น A[C]</p>	$T_H^i \geq T_C^s, L_H \leq L_C$ $W_H > W_C$ $T_H^s \geq T_C^s + L_H W_C^{-1}$	$T_C^s = T_C^s + L_H W_C^{-1}$ $L_C = L_C - L_H$
 <p>แพทเทิร์น B[H]</p>	$T_H^s \geq T_C^i, L_H \geq L_C$ $W_H < W_C$ $T_C^s \leq T_H^s - L_C W_H^{-1}$	$T_H^s \leq T_H^s - L_C W_H^{-1}$ $L_H = L_H - L_C$

ตารางที่ 3.1 (ต่อ) โอเปอเรเตอร์ของกระบวนการจับคู่คลาสซี และ คลาสดี

โอเปอเรเตอร์	เงื่อนไข	ค่าที่คำนวณใหม่
 <p>แพทเทิร์น C H</p>	$T_H^t \geq T_C^s$ $L_H > L_C$ $W_H \leq W_C$	$T_H^t = T_H^t + L_C W_H^{-1}$ $L_H = L_H - L_C$
 <p>แพทเทิร์น D C</p>	$T_H^s \geq T_C^t$ $L_H < L_C$ $W_H \geq W_C$	$T_C^t = T_C^t - L_H W_C^{-1}$ $L_C = L_C - L_H$
 <p>แพทเทิร์น C C</p>	$T_H^t \geq T_C^s, L_H > L_C$ $W_H > W_C$ $T_C^t \leq T_H^t + L_C W_H^{-1}$	$T_H^t = T_H^t + L_C W_H^{-1}$ $L_H = L_H - L_C$
 <p>แพทเทิร์น D</p>	$T_H^s \geq T_C^t, L_H < L_C$ $W_H < W_C$ $T_H^t \geq T_C^t - L_H W_C^{-1}$	$T_C^t = T_C^t - L_H W_C^{-1}$ $L_C = L_C - L_H$

T^t = อุณหภูมิเป้าหมาย, T^s = อุณหภูมิขาเข้า

W = ผลคูณของอัตราการไหลและความจุความร้อนจำเพาะ

L หรือ Q = ความร้อนที่ใช้ในการแลกเปลี่ยน

อุณหภูมิของกระแสเย็นในที่นี้ (T_C) จะถูกลบออกไปด้วย ΔT_{min} แล้ว

กระแสนี้จะให้มันเป็นแพทเทิร์น B[H] สำหรับการพิจารณาในกระบวนการจับคู่ของคลาสซี และคลาสซีก็พิจารณาค้างๆกัน

สรุปสำหรับปัญหาทางด้านความร้อน จะใช้กระบวนการจับคู่เรียงตามลำดับดังนี้ AH, A[H], B[C], A[C], B[H], C[H], D[C], C[C] และ D[H] และปัญหาทางด้านความเย็นจะใช้ลำดับกระบวนการจับคู่เป็น BC, B[C], A[H], B[H], A[C], D[C], C[H], D[H] และ C[C]

3.3 กระบวนการจับคู่ของข่ายงานแบบยัดหยุ่น

กระบวนการจับคู่ที่ใช้กับข่ายงานแบบยัดหยุ่นนั้น ต้องสามารถส่งความแปรปรวนของกระแสทิ้งไปให้อีกกระแสหนึ่งซึ่งมีความร้อนหลงเหลือได้ นอกจากนี้ จะต้องมีการวัดหรือทดสอบค่าความยัดหยุ่นของกระแสและความยัดหยุ่นของเครื่องแลกเปลี่ยนความร้อนด้วย

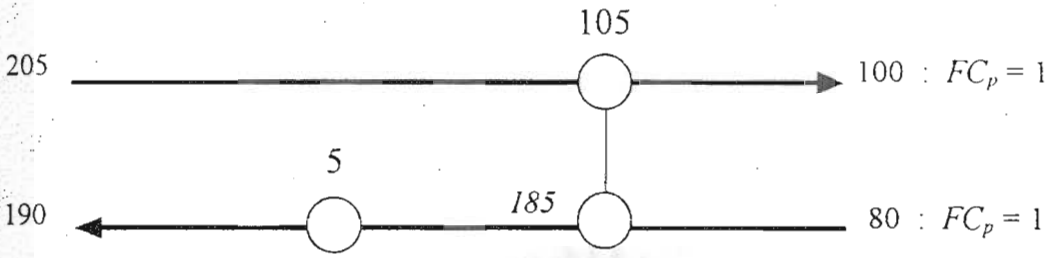
สำหรับกระแสการจับคู่คลาสเอและคลาสบี ความแปรปรวนของทั้งสองกระแสสามารถถูกส่งไปให้กระแสที่มีความร้อนหลงเหลือได้ ดังนั้น กระบวนการจับคู่ทั้งสองนี้ถือว่าเป็นกระบวนการจับคู่แบบยัดหยุ่น ส่วนคลาสซี พบว่ามีเพียงความแปรปรวนของกระแสร้อนหรือสำหรับคลาสดี จะมีเพียงความแปรปรวนของกระแสนี้ที่สามารถถูกจัดการได้เท่านั้น กระบวนการจับคู่ทั้งสองนี้ ไม่สามารถจัดการกับความแปรปรวนที่เกิดขึ้นในสองกระแสพร้อมกันได้ ดังนั้นจึงถือว่า คลาสซีและคลาสดีเป็นกระบวนการจับคู่สำหรับการออกแบบข่ายงานที่มีค่านิ่งถึงความยัดหยุ่นเท่านั้น (อย่างไรก็ตาม ถ้าต้องการใช้คลาสซีและดีในการออกแบบข่ายงานแบบไม่ยัดหยุ่นแล้ว จะต้องมีการควบคุมที่เหมาะสมร่วมด้วย)

พิจารณารูปที่ 3.2 เป็นตัวอย่างการแลกเปลี่ยนความร้อนระหว่างกระแสร้อนและกระแสน้ำรวมสองกระแส ในที่นี้ใช้กระบวนการจับคู่คลาสเอ โดยให้ ΔT_{min} เท่ากับ 10°

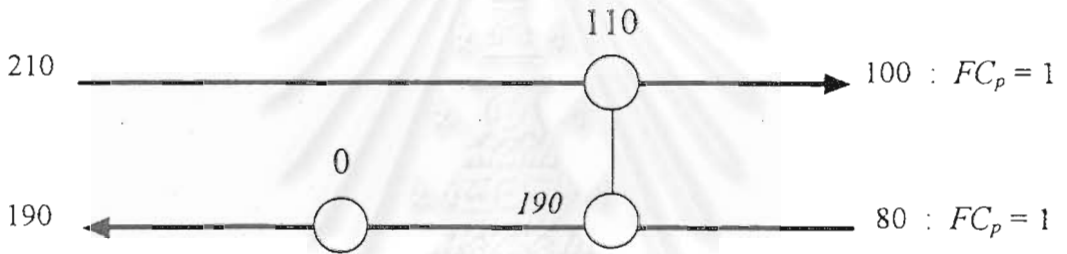
กระแสทั้งสองใช้ความร้อนในการแลกเปลี่ยนกันเองเท่ากับ 105 หน่วย กระแสน้ำจะเหลือความร้อนอยู่อีก 5 หน่วย ซึ่งจะใช้ความร้อนจากยูทิลิตี้ (ฮีตเตอร์) หรือนำไปแลกเปลี่ยนความร้อนกับกระแสร้อนกระแสอื่น เพื่อเพิ่มอุณหภูมิจาก 185° ให้ถึงอุณหภูมิเป้าหมายคือ 190°

รูปที่ 3.3 อุณหภูมิขาเข้ากระแสร้อนมีค่าเพิ่มขึ้นไปอีก 5° อุณหภูมิขาเข้าจึงเป็น 210° คิดเป็นความแปรปรวน หรือ ความร้อนเนื่องจากอุณหภูมิที่เปลี่ยนไป $FC_p \Delta T = 1 * (210 - 205) = 5$ หน่วยความร้อนในกระแสร้อนเพิ่มจาก 105 เป็น 110 หน่วย โดยที่ความร้อนของกระแสร้อนนี้เท่ากับความร้อนของกระแสน้ำ จึงใช้แลกเปลี่ยนกันได้พอดี ซึ่งหมายความว่า อุณหภูมิขาเข้าของกระแสร้อนในที่นี้สามารถเพิ่มขึ้นได้ถึง 5° โดยที่ไม่ทำให้อุณหภูมิเป้าหมายของกระแสน้ำเปลี่ยนไปจาก 190°

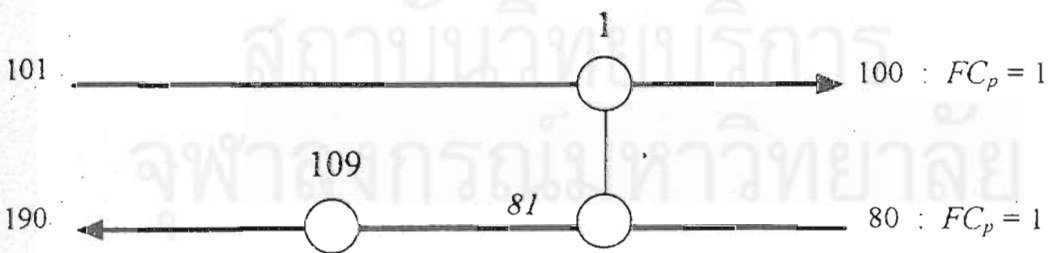
รูปที่ 3.4 แสดงอุณหภูมิขาเข้าของกระแสร้อนเมื่อลดลงไปถึง 101° ในที่นี้กระแสร้อนจะเหลือความร้อนอยู่เพียง 1 หน่วย ดังนั้น การที่จะทำให้อุณหภูมิขาออกของกระแสน้ำเท่ากับอุณหภูมิเป้าหมาย 190° ให้ได้จึงตกเป็นภาระของฮีตเตอร์ทั้งหมด โดยที่ความร้อนเท่ากับ 109 หน่วย (ความร้อน



รูปที่ 3.2 ตัวอย่างการแลกเปลี่ยนความร้อนระหว่างสองกระแส



รูปที่ 3.3 เมื่ออุณหภูมิขาเข้าของกระแสร้อนเพิ่มขึ้น 5°

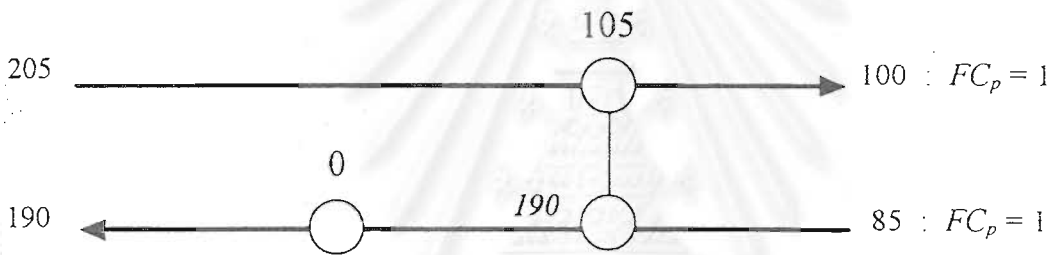


รูปที่ 3.4 เมื่ออุณหภูมิขาเข้าของกระแสร้อนลดลงเหลือ 101°

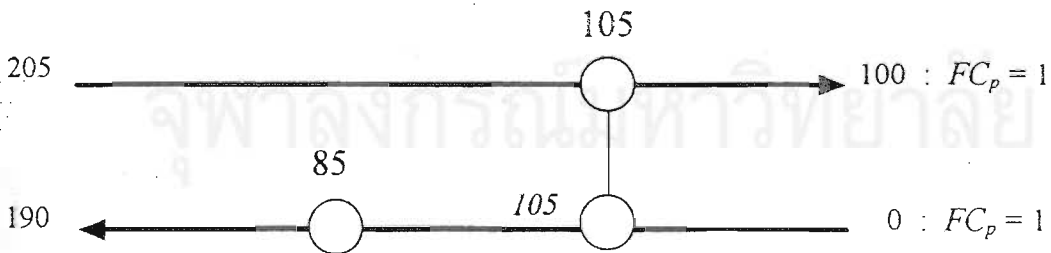
จำนวนนี้ อาจถูกมองว่าเป็นปริมาณความร้อนที่หลงเหลือของกระแสน้ำซึ่งอาจถูกส่งไปแลกเปลี่ยนความร้อนกับกระแสน้ำอื่นๆที่ไม่ใช่กระแสน้ำจากยูทิลิตี้ก็ได้)

รูปที่ 3.5 เมื่อเพิ่มอุณหภูมิขาเข้าของกระแสน้ำจาก 80° เป็น 85° ความร้อนจากทั้งสองกระแสน้ำจะเท่ากันพอดี จึงไม่ต้องใช้ความร้อนจากฮีตเตอร์อีก เช่นเดียวกับจากรูป 3.2 การเพิ่มอุณหภูมิขาเข้าของกระแสน้ำให้มีค่าสูงกว่า 85° จะทำให้อุณหภูมิขาออกของกระแสน้ำสูงกว่าอุณหภูมิ 190° ดังนั้นในที่นี้อุณหภูมิ 85° จึงเป็นอุณหภูมิขาเข้าสูงสุดของกระแสน้ำที่เป็นไปได้

รูปที่ 3.6 จากรูป อุณหภูมิขาเข้าของกระแสน้ำสามารถลดลงได้ไม่มีขีดจำกัด โดยภาระทั้งหมดจะถูกผลักไปให้แก่ฮีตเตอร์แทน ในที่นี้แสดงตัวอย่างเมื่ออุณหภูมิขาเข้าของกระแสน้ำลดลงไปถึง 0°



รูปที่ 3.5 เมื่ออุณหภูมิขาเข้าของกระแสน้ำเพิ่มขึ้น 5°



รูปที่ 3.6 เมื่ออุณหภูมิขาเข้าของกระแสน้ำลดลงเหลือ 0°

ต่อไปนี้เป็นพิจารณาผลของการเพิ่มหรือลดอุณหภูมิ สำหรับการแลกเปลี่ยนความร้อน กระบวนการจับคู่คลาสติ

รูปที่ 3.7 เป็นการแลกเปลี่ยนความร้อนที่ภาวะปกติที่ปลายด้านอุณหภูมิสูง โดยกระแสน้ำจะเหลือความร้อนเท่ากับ 5 หน่วย ความร้อนจำนวนนี้อาจมองเป็นฮีตเตอร์ตัวหนึ่ง หรือเป็นความร้อนที่จะถูกแลกเปลี่ยนความร้อนกับกระแสน้ำอื่นๆ ต่อไป หรือเป็นความร้อนที่ได้ถูกแลกเปลี่ยนความร้อนกับกระแสน้ำไว้แล้วก็ได้ ซึ่งถ้ามองในกรณีหลังนี้ จะพบว่ากระบวนการจับคู่คลาสตินี้ไม่มีความยืดหยุ่นในบางกรณี ดังที่จะกล่าวต่อไปนี้

รูปที่ 3.8 เมื่ออุณหภูมิขาเข้าของกระแสน้ำเพิ่มขึ้นเท่ากับ 1° จะทำให้อุณหภูมิขาออกของกระแสน้ำเพิ่มจาก 190° เป็น 191° ซึ่งเกินค่าอุณหภูมิเป้าหมายคือ 190° ไป

รูปที่ 3.9 เมื่ออุณหภูมิขาเข้าของกระแสน้ำลดลงขึ้นเท่ากับ 1° จะทำให้อุณหภูมิขาออกของกระแสน้ำลดลงจาก 190° เป็น 189° ซึ่งน้อยกว่าค่าอุณหภูมิเป้าหมาย 190°

รูปที่ 3.10 และ 3.11 พิจารณาเช่นเดียวกับคลาสเอ ซึ่งได้ผลคล้ายๆ กัน

สรุปก็คือ เปรียบเทียบระหว่างคลาสเอและคลาสดีแล้ว คลาสเอสามารถจัดการกับความแปรปรวนได้ทั้งจากกระแสน้ำและกระแสน้ำ ส่วนคลาสดีจัดการได้เฉพาะความแปรปรวนของกระแสน้ำเท่านั้น ดังนั้น คลาสเอจึงเหมาะเป็นกระบวนการจับคู่สำหรับช่างงานแบบยืดหยุ่นมากกว่าคลาสดี กรณีการเปรียบเทียบระหว่างคลาสบีและคลาสซีก็เช่นเดียวกัน จะพบว่าควรใช้คลาสบีเป็นกระบวนการจับคู่แบบยืดหยุ่นมากกว่าคลาสซี และสำหรับพารามิเตอร์ทดสอบช่างงานแบบยืดหยุ่นที่ใช้ร่วมกับกระบวนการจับคู่นี้ จะได้กล่าวในภายหลัง

3.4 ชนิดของความแปรปรวน

การออกแบบช่างงานเครื่องแลกเปลี่ยนความร้อนแบบยืดหยุ่น ความแปรปรวนที่ถูกใช้นำมาพิจารณามีสองแบบ คือ ความแปรปรวนเนื่องจากอุณหภูมิขาเข้า และความแปรปรวนเนื่องจากอัตราการไหล

$$\text{ความแปรปรวนเนื่องจากอุณหภูมิ} = W(T_{\text{supply,max}} - T_{\text{supply,min}}) \quad (3.1)$$

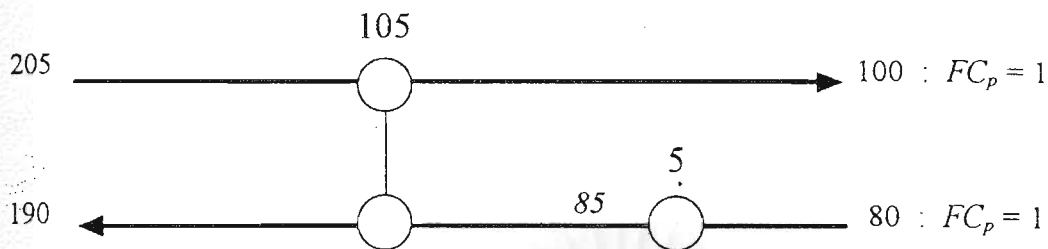
$$\text{ความแปรปรวนเนื่องจากอัตราการไหล} = (W_{\text{max}} - W_{\text{min}})(T_{\text{target}} - T_{\text{supply}}) \quad (3.2)$$

เมื่อ W (หรือ FC_p) คือ ผลคูณของอัตราการไหลและความจุความร้อนจำเพาะ

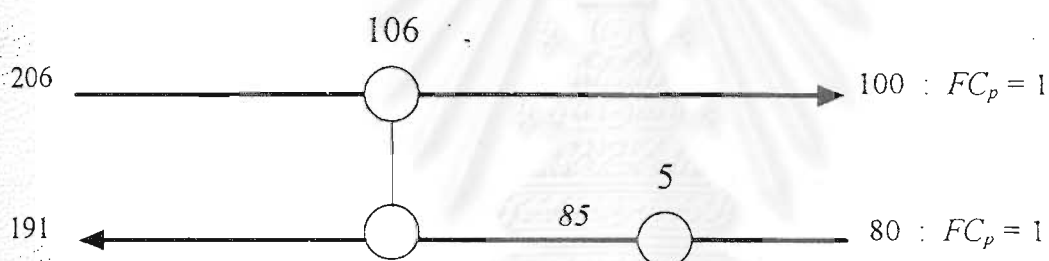
T_{supply} และ T_{target} คืออุณหภูมิขาเข้าและอุณหภูมิเป้าหมายตามลำดับ

ความแปรปรวนดังกล่าวนี้ ยังสามารถแบ่งได้อีกเป็นสองประเภท คือ ความแปรปรวนบวก หรือความแปรปรวนลบ ดังนี้

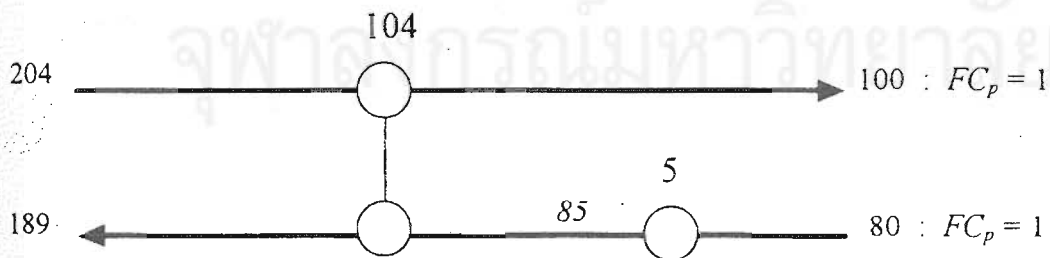
ความแปรปรวนบวก คือ ความแปรปรวนที่ทำให้ความร้อนกระแสน้ำมีค่าเพิ่มขึ้น เช่น กรณีอุณหภูมิขาเข้าของกระแสน้ำถูกรบกวนโดยมีค่าสูงกว่าที่ภาวะปกติ หรืออุณหภูมิขาเข้าของกระแสน้ำมีค่าที่เย็นต่ำกว่าค่าปกติ ปริมาณความร้อนที่เพิ่มขึ้นเนื่องจากความแปรปรวนนี้ สามารถที่จะถูกกำจัดไปได้ โดยส่ง



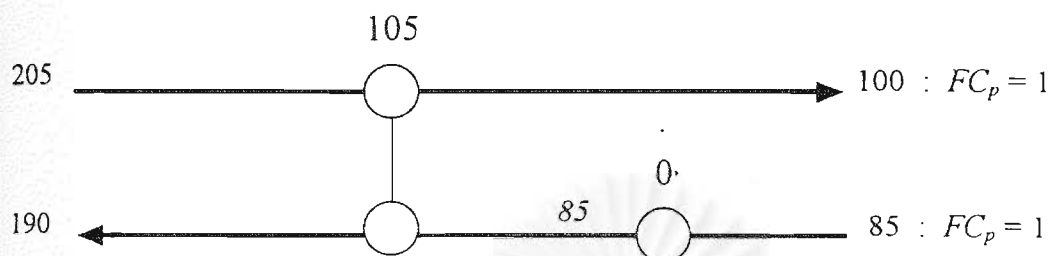
รูปที่ 3.7 การแลกเปลี่ยนความร้อนที่ภาวะปกติ



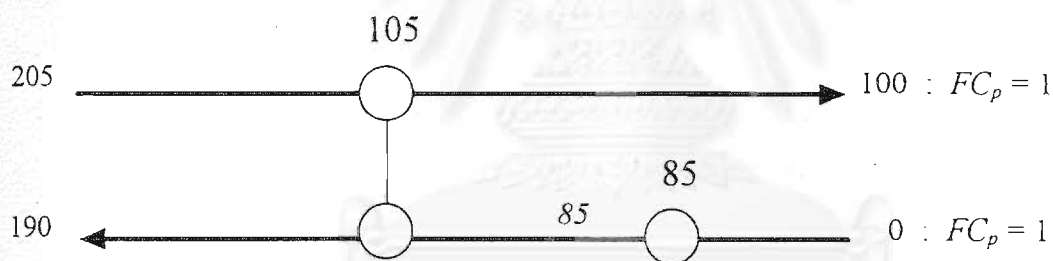
รูปที่ 3.8 เมื่ออุณหภูมิขาเข้าของกระแสน้ำเพิ่มขึ้น 1°



รูปที่ 3.9 เมื่ออุณหภูมิขาเข้าของกระแสน้ำลดลง 1°



รูปที่ 3.10 เมื่ออุณหภูมิขาเข้าของกระแสเย็นเพิ่มขึ้น 5°



รูปที่ 3.11 เมื่ออุณหภูมิขาเข้าของกระแสเย็นลดลง

ไปที่กระแสน้ำซึ่งมีเครื่องแลกเปลี่ยนความร้อนจากหน่วยผลิตบริการอยู่ (กระแสน้ำที่ติดตั้งฮีตเตอร์ หรือ คูลเลอร์) ถ้าเกิดความแปรปรวนบวกในกระแสน้ำร้อน จะทำให้เพิ่มปริมาณความร้อนที่ใช้ในการแลกเปลี่ยนของคูลเลอร์ แต่จะลดปริมาณความร้อนที่ใช้ในการแลกเปลี่ยนของฮีตเตอร์ ในขณะที่ความแปรปรวนบวกของกระแสเย็น จะทำให้เพิ่มปริมาณความร้อนที่ใช้ของฮีตเตอร์ แต่จะลดปริมาณความร้อนของคูลเลอร์

ความแปรปรวนลบ คือความแปรปรวนที่ทำให้ความร้อนกระแสมีค่าลดลงเช่น กรณีอุณหภูมิขาเข้าของกระแสร้อนถูกรบกวนโดยมีค่าต่ำกว่าที่ภาวะปกติ หรืออุณหภูมิขาเข้าของกระแสเย็นมีค่าที่เย็นสูงกว่าค่าปกติ ความแปรปรวนลบถ้าเกิดในกระแสร้อน จะทำให้ลดปริมาณความร้อนของคอนดิวเตอร์ลง แต่จะเพิ่มปริมาณความร้อนของฮีตเตอร์ ในขณะที่ความแปรปรวนลบของกระแสเย็น จะทำให้ลดปริมาณความร้อนที่ใช้ของฮีตเตอร์ แต่จะเพิ่มปริมาณความร้อนของคอนดิวเตอร์ให้สูงขึ้น

3.5 สถานะการออกแบบ

จากการที่ข่างานถูกรบกวนโดยค่าของอุณหภูมิ และอัตราการไหลซึ่งแปรปรวนออกไปจากค่าที่ได้กำหนดไว้จากสภาวะปกตินั้น การรบกวนดังกล่าวจะแบ่งได้สี่สถานะ (คิดที่สภาวะสุดขีด หรือ Extream conditions) คือ

สภาวะโหลดความร้อนสูงสุด (Maximum heat load condition) เป็นสภาวะที่ทุกกระแสมีค่าปริมาณความร้อนมากที่สุด เช่นอุณหภูมิขาเข้าของกระแสร้อนมีค่าสูงที่สุดจากช่วงการแปรปรวน และอุณหภูมิขาเข้าของกระแสเย็นมีค่าต่ำที่สุดจากช่วงการแปรปรวน

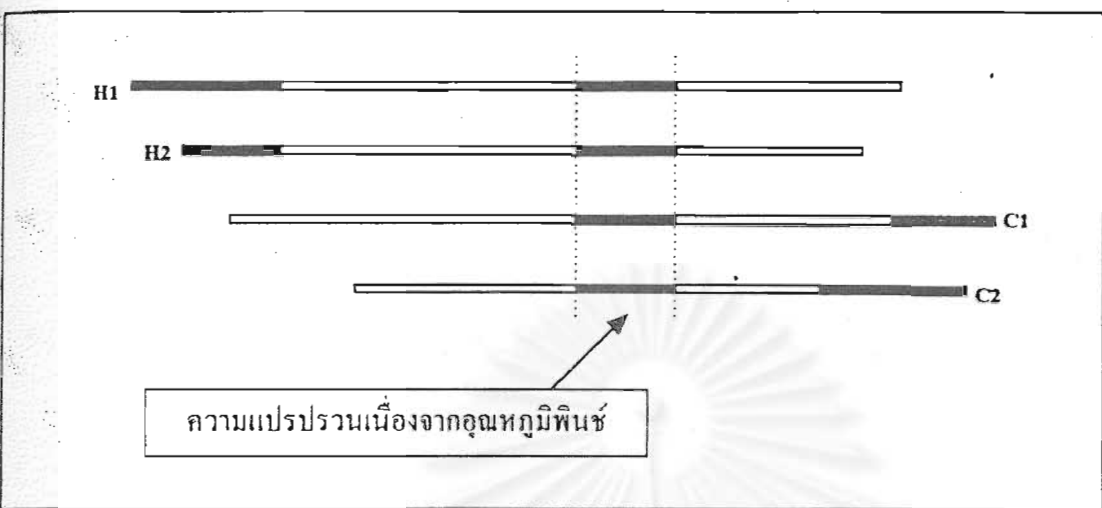
สภาวะโหลดความร้อนต่ำสุด (Minimum heat load condition) เป็นสภาวะที่ทุกกระแสมีค่าปริมาณความร้อนน้อยที่สุด เช่นอุณหภูมิขาเข้าของกระแสร้อนมีค่าต่ำที่สุดจากช่วงการแปรปรวน และอุณหภูมิขาเข้าของกระแสเย็นมีค่าสูงที่สุดจากช่วงการแปรปรวน

สภาวะดึงความร้อนสูงสุด (Maximum cooling condition) เป็นสภาวะที่กระแสร้อนมีค่ามากที่สุด และกระแสเย็นมีความร้อนน้อยที่สุด เช่นอุณหภูมิขาเข้าของกระแสร้อนมีค่าสูงสุด และของกระแสเย็นมีค่าต่ำสุด

สภาวะให้ความร้อนสูงสุด (Minimum heating condition) เป็นสภาวะที่กระแสร้อนมีความร้อนน้อยที่สุด และกระแสเย็นมีความร้อนมากที่สุด เช่นอุณหภูมิขาเข้าของกระแสร้อนมีค่าต่ำสุด และของกระแสเย็นมีค่าสูงสุด

สำหรับการออกแบบข่างานนั้น ถ้าทำการวิเคราะห์ข่างานด้วยความแปรปรวนเพียงชนิดเดียวแล้ว (ความแปรปรวนบวก หรือลบ) จะทำให้สามารถพิจารณาทดสอบความยืดหยุ่นเพื่อจับคู่แลกเปลี่ยนความร้อนระหว่างกระแสร้อนและเย็นได้มากขึ้น ซึ่งในที่นี้ก็คือ ที่สภาวะโหลดความร้อนของกระแสมีค่าสูงสุด หรือต่ำสุด โดยที่สภาวะโหลดความร้อนสูงสุด จะทำให้เกิดความแปรปรวนลบชนิดเดียวทั้งในกระแสเย็นและกระแสร้อน ส่วนสภาวะโหลดความร้อนต่ำสุด จะทำให้เกิดความแปรปรวนบวกชนิดเดียวเช่นกัน นอกจากนี้ที่สภาวะทั้งสอง ความแตกต่างระหว่างกระแสร้อน และกระแสเย็นจะเป็นที่สภาวะขีดสุดด้วย ซึ่งที่สภาวะอื่นๆจะไม่ใช่

และสำหรับปัญหาที่มีอุณหภูมิพินช์ ถ้าสภาวะการรบกวนทำให้เกิดความแปรปรวน เนื่องจากอุณหภูมิพินช์เพิ่มขึ้นมา ความแปรปรวนจากอุณหภูมิพินช์นี้ จะถูกนำมาพิจารณาพร้อมกันไปด้วย (ดูรูปที่ 3.12)



รูปที่ 3.12 ความแปรปรวนเนื่องจากอุณหภูมิพินช์ในข่ายงานแบบยี่ดหุ่่น

3.6 พารามิเตอร์ที่ใช้ในการคำนวณ

ในการคำนวณ และทดสอบความยืดหยุ่น จะมีพารามิเตอร์ที่ใช้ร่วมกับกระบวนการจับคู่ดังนี้
 โหลดของเครื่องแลกเปลี่ยนความร้อน (Heat Exchanger Load, L_{E_i}) คือความร้อนที่ใช้ในการแลกเปลี่ยนของเครื่องแลกเปลี่ยนความร้อน

โหลดของกระแสกระบวนการ (Process Stream Load, L_{S_i}) คือปริมาณความร้อนของกระแสพารามิเตอร์ความยืดหยุ่นของเครื่องแลกเปลี่ยนความร้อน (Heat Exchanger Resiliency Parameter, E) คือความแตกต่างของอุณหภูมิในการแลกเปลี่ยนความร้อน (ΔT) กับ ΔT_{min}

$$E_{ij} = W_j (\Delta T_i - \Delta T_{min}) \quad \text{เมื่อ } L_j > L_i \quad (3.3)$$

ความยืดหยุ่นของกระแส (Stream Resiliency, R_{ij}) คือความร้อนหลงเหลือของกระแส

พารามิเตอร์ความยืดหยุ่นของกระแส (Stream Resiliency Parameter, S) นิยามโดย

$$S_i = (W_j - W_i)(T_i^1 - T_i^2) \quad (3.4)$$

เมื่อ T_i^1 และ T_i^2 คือ อุณหภูมิของกระแสปลายด้านอุณหภูมิสูงและต่ำตามลำดับ

ความแปรปรวนจากอุณหภูมิ (Temperature Disturbance, D^θ) เป็นค่าความแปรปรวนเนื่องจากอุณหภูมิขาเข้า

$$D^\theta = W (T_{Supply,max} - T_{Supply,min}) \quad (3.5)$$

ความแปรปรวนจากอัตราการไหล (Flowrate Disturbance, D^w) คือค่าความแปรปรวนของผลคูณของอัตราการไหลกับความจุความร้อนจำเพาะของกระแส

$$D_i^w = (W_{i,max} - W_{i,min})(T_{i,max}^1 - T_{i,min}^2) \quad (3.6)$$

เมื่อ $T_{i,max}^1$ คือ อุณหภูมิปลายกระแสด้านอุณหภูมิสูงที่มีค่ามากที่สุดจากช่วงการแปรปรวน และ

$T_{i,min}^2$ คือ อุณหภูมิปลายกระแสด้านอุณหภูมิต่ำที่มีค่าน้อยที่สุดจากช่วงการแปรปรวน

ความแปรปรวนเริ่มต้น (Original Disturbance, D) คือความแปรปรวนรวมซึ่งมีค่าเท่ากับ

$$D_i = D_i^o + D_i^w \quad (3.7)$$

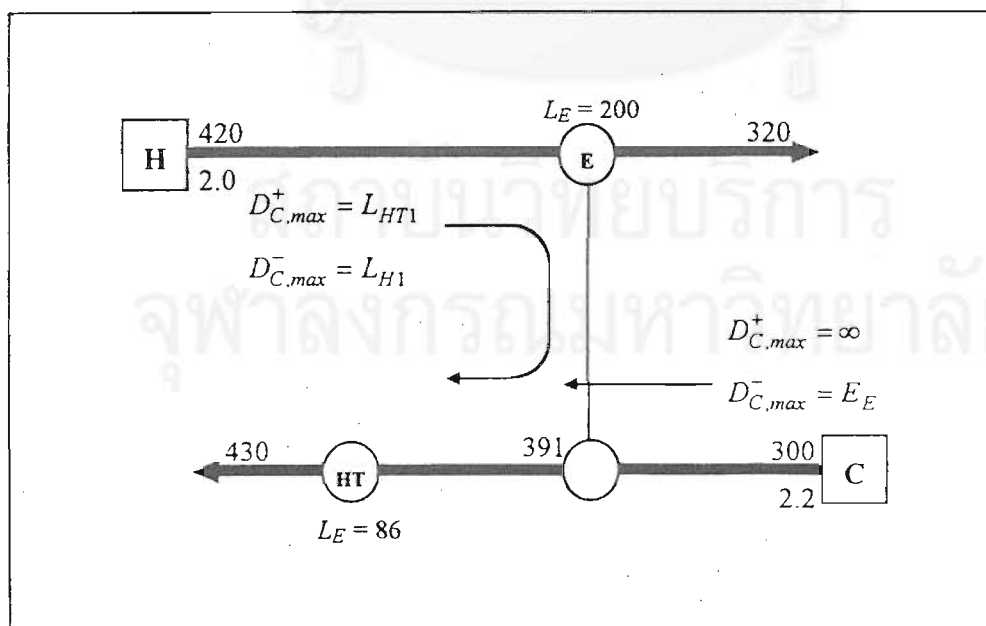
ความแปรปรวนจากอุณหภูมิพินช์ (Pinch Induced Disturbance, D^p) นิยามโดย

$$D_i^p = W_{i,max}(T_{i,max}^p - T_{i,min}^p) \quad (3.8)$$

สำหรับเงื่อนไขในการพิจารณาพารามิเตอร์เหล่านี้ที่ใช้กระแสการจับคู่ เพื่อออกแบบขำยงานเครื่องแลกเปลี่ยนความร้อนแบบยัดหยุน ได้แสดงไว้ในตารางที่ 3.2

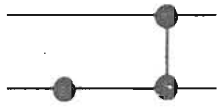
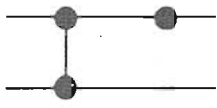
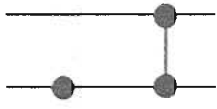
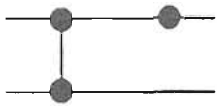
ต่อไปนี้เป็นอีกตัวอย่างหนึ่งของการพิจารณาความแปรปรวน ที่เกิดขึ้นระหว่างการแลกเปลี่ยนความร้อนระหว่างกระแส ในที่นี้จะแสดงความแปรปรวนของอุณหภูมิเท่านั้น

จากรูปที่ 3.13 มีกระแสร้อน (H) และกระแสนเย็น (C) นำมาแลกเปลี่ยนความร้อนกันในแบบการจับคู่คลาสเอ โดยความร้อนที่เหลือของกระแสนเย็น C จะถูกให้ความร้อนด้วยฮีตเตอร์ (HT) ให้ ΔT_{min} เท่ากับ 10° การพิจารณามีดังนี้



รูปที่ 3.13 แสดงตัวอย่างการพิจารณาความแปรปรวนของกระแส

ตารางที่ 3.2 โอเปอเรเตอร์กระสวนการจับคู่แบบยี่ดหุ่่น

โอเปอเรเตอร์	เงื่อนไข	การตรวจสอบความยี่ดหุ่่น
 <p>แพทเทิร์น A[H]</p>	$T_H^i \geq T_C^s$ $L_H \leq L_C$ $W_H \leq W_C$	$D^\theta + D^\omega \leq R_{C,H}$ $D^\omega \leq E_{C,H} + S_{C,H}$
 <p>แพทเทิร์น B[C]</p>	$T_H^s \geq T_C^i$ $L_H \geq L_C$ $W_H \geq W_C$	$D^\theta + D^\omega \leq R_{H,C}$ $D^\omega \leq E_{H,C} + S_{H,C}$
 <p>แพทเทิร์น A[C]</p>	$T_H^i \geq T_C^s$ $L_H \leq L_C$ $W_H > W_C$ $-S_{C,H} \leq E_{C,H}$	$D^\theta + D^\omega \leq \min \{R_{C,H}, E_{C,H}\}$ $D^\omega \leq E_{C,H} + S_{C,H}$
 <p>แพทเทิร์น B[H]</p>	$T_H^s \geq T_C^i$ $L_H \geq L_C$ $W_H < W_C$ $-S_{H,C} \leq E_{H,C}$	$D^\theta + D^\omega \leq \min \{R_{H,C}, E_{H,C}\}$ $D^\omega \leq E_{H,C} + S_{H,C}$

D^θ = ความแปรปรวนเนื่องจากอุณหภูมิ (Temperature disturbance)

D^ω = ความแปรปรวนเนื่องจากค่าผลคูณของอัตราการไหลและความจุความร้อนจำเพาะ (Heat capacity flowrate disturbance)

R = ความยี่ดหุ่่นของกระแส (Stream resiliency)

E = พารามิเตอร์ความยี่ดหุ่่นของเครื่องแลกเปลี่ยนความร้อน (Exchanger resiliency parameter)

S = พารามิเตอร์ความยี่ดหุ่่นของกระแส (Stream resiliency parameter)

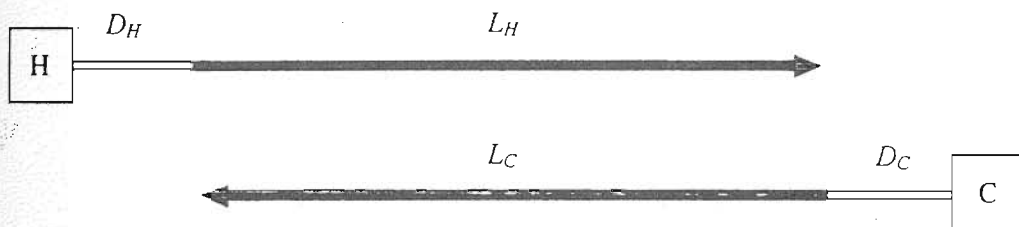
1. กรณีอุณหภูมิเพิ่มขึ้น ความแปรปรวนบวกสูงสุดของกระแส H จะมีค่าเท่ากับความร้อนที่ใช้ในการแลกเปลี่ยนความร้อนของฮีตเตอร์ HT (หรืออุณหภูมิขาเข้าของกระแส H ที่จะเป็นไปได้ คือไม่สูงกว่า 463°) เพราะถ้าความแปรปรวนบวกมีค่ามากกว่านี้ จะทำให้อุณหภูมิขาออกของกระแส H หลังจากผ่านเครื่องแลกเปลี่ยนความร้อน E ไม่ได้ค่าเท่ากับอุณหภูมิเป้าหมาย คือค่าจะไม่ต่ำลงถึง 320° หรืออีกกรณีหนึ่ง อุณหภูมิขาออกของกระแส C จะมีค่าสูงกว่า 430° สำหรับการพิจารณาเมื่ออุณหภูมิขาเข้าของกระแส C เพิ่มขึ้นจะต้องพิจารณาที่ต่างกันออกไปจากกรณีสำหรับกระแส H เพราะว่าความแตกต่างของอุณหภูมิต่ำที่สุดในการแลกเปลี่ยนความร้อนของกระแสในเครื่องแลกเปลี่ยนความร้อน ถูกจำกัดด้วยค่า ΔT_{min} เท่ากับ 10° โดยความแปรปรวนสูงสุดของกระแส C ที่มีได้นั้น มีค่าเท่ากับพารามิเตอร์ความยืดหยุ่นของเครื่องแลกเปลี่ยนความร้อน (E_E)

$$\begin{aligned} E_E &= \Delta T_S W_C \\ &= (20 - 10) 2.2 \\ &= 22 \text{ kW} \end{aligned}$$

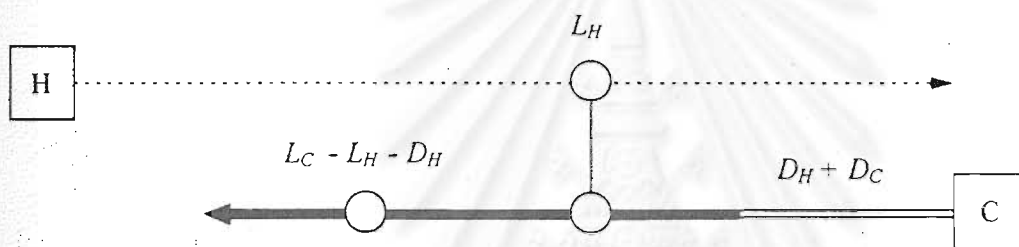
2. กรณีอุณหภูมิลดลง ความแปรปรวนลบสำหรับกระแส H นั้นจะถูกจำกัดด้วยค่า เนื่องจากอุณหภูมิขาเข้าของกระแส H สามารถลดลงได้เท่ากับ อุณหภูมิเป้าหมายของกระแสเอง โดยกระแส C ยังสามารถใช้ฮีตเตอร์ให้ความร้อน โดยมีอุณหภูมิขาออกค่าเท่ากับอุณหภูมิเป้าหมายได้ (ชัดเจนความร้อนจากกระแส H ที่น้อยลงไป) กรณีถ้าอุณหภูมิของกระแส C ลดลงนั้นจะเป็นความแปรปรวนบวก ซึ่งสามารถลดลงได้อย่างไม่จำกัด เนื่องจากสามารถผลักภาระความร้อนส่วนเกินไปให้ฮีตเตอร์ได้ทั้งหมด

3.7 วิธีการส่งผ่านความแปรปรวน (Disturbance Propagation Method หรือ DPM)

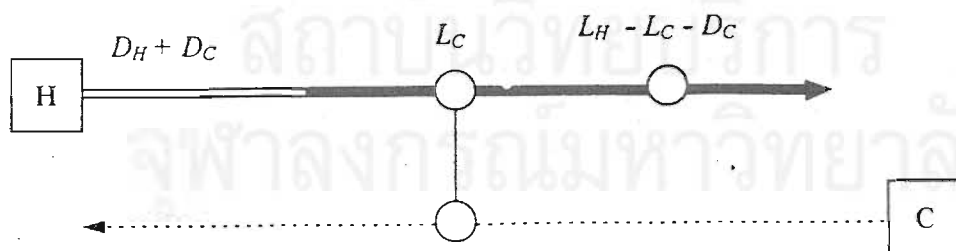
โดยการเลือกภาวะที่กระแสมีโหลดความร้อนต่ำสุด (Minimum heat load) เมื่อมีการจับคู่ระหว่างกระแส ความแปรปรวนของกระแสหนึ่งจะถูกส่งไปที่กับกระแสที่มีความร้อนหลงเหลือหลังการจับคู่ ซึ่งความแปรปรวนของกระแสที่มีความร้อนหลงเหลือนี้ จะเท่ากับผลรวมความแปรปรวนของกระแสทั้งสอง ดังนั้นแม้ว่าอีกกระแสหนึ่งไม่มีความแปรปรวนจะถูกจับคู่กับอีกกระแสหนึ่งซึ่งมีความแปรปรวน (แต่มีปริมาณความร้อนน้อยกว่า) ผลสุดท้ายความแปรปรวนของกระแสที่สองก็จะถูกส่งไปให้กระแสแรก ดูรูปที่ 3.14 - 3.16



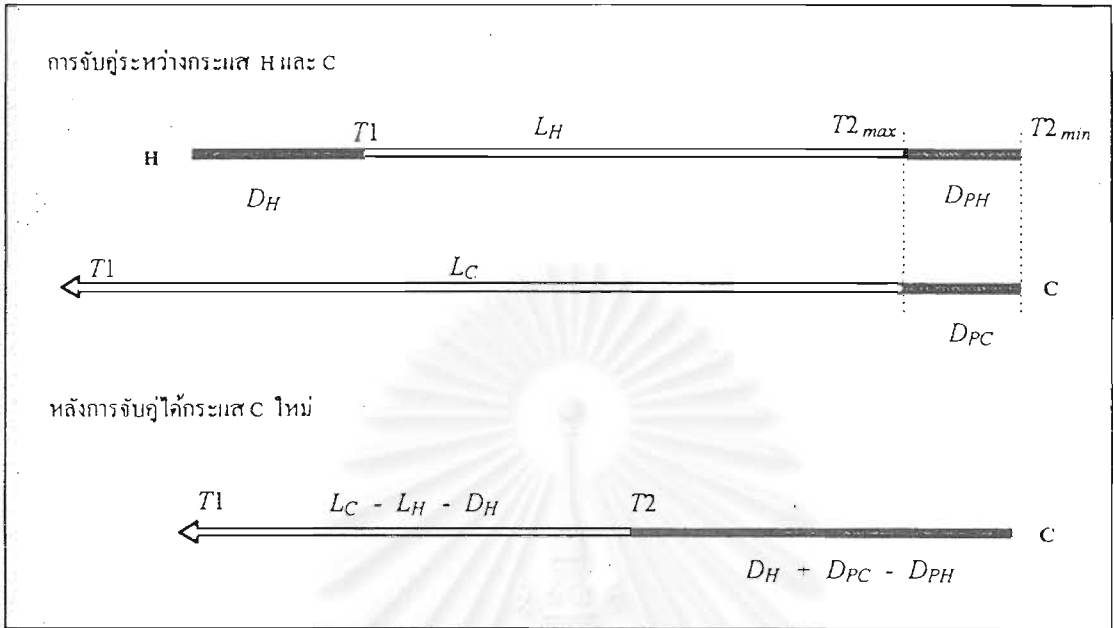
รูปที่ 3.14 แสดงความร้อนและความแปรปรวนของกระแส
ก่อนการแลกเปลี่ยนความร้อน



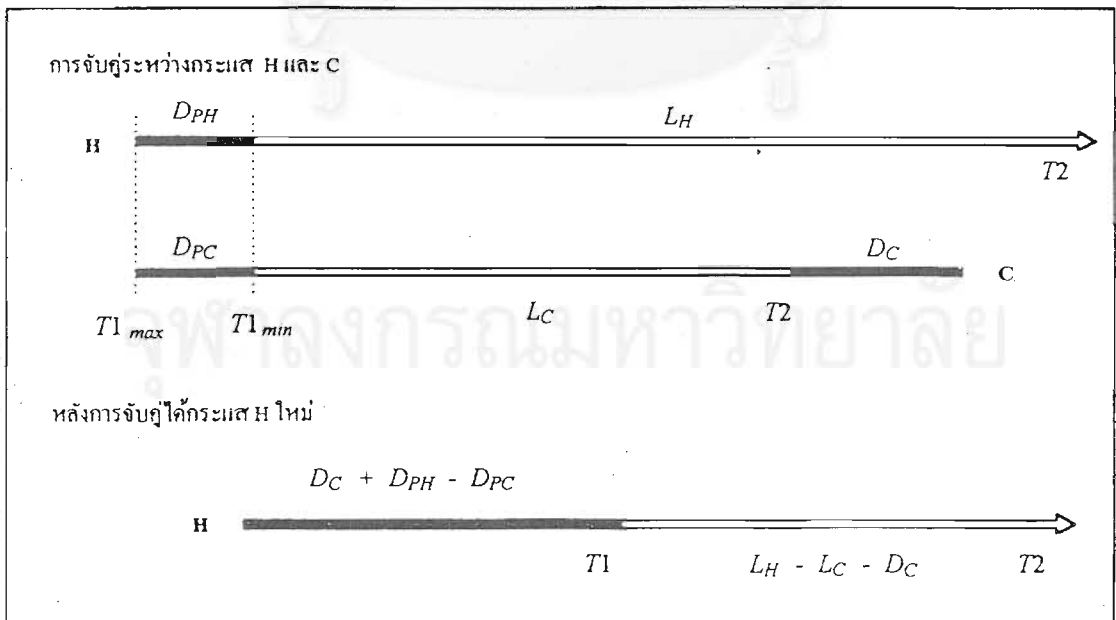
รูปที่ 3.15 การแลกเปลี่ยนความร้อนเมื่อ $L_C > L_H$ (คลาสสิก)



รูปที่ 3.16 การแลกเปลี่ยนความร้อนเมื่อ $L_C < L_H$ (คลาสสิก)



รูปที่ 3.17 กรณี $L_C > L_H$ (คลาสเอ)



รูปที่ 3.18 กรณี $L_H > L_C$ (คลาสบี)

$$D_H^\omega = (1 - 0.8)(250 - 120) = 26$$

$$D_H^\theta + D_H^\omega = 30 < R_{C,H}$$

3. พารามิเตอร์ความยืดหยุ่นของเครื่องแลกเปลี่ยนความร้อน, $E_{C,H} = 2(20 - 10) = 20$

พารามิเตอร์ความยืดหยุ่นของกระแส, $S_{C,H} = (2 - 0.8)(250 - 120) = 156$

$$D_H^\omega = 26 < E_{C,H} + S_{C,H}$$

4. ความแปรปรวนทั้งหมดของกระแส $H = 26 + 4 = 30$ จะถูกส่งไปให้กระแส C และสำหรับกระแส C มีความแปรปรวนเนื่องจากอุณหภูมิอย่างเดียวโดย $D_C^\theta = 2(100 - 90) = 20$ ดังนั้นความแปรปรวนรวมในกระแส C จะมีค่าเท่ากับ $20 + 30 = 50$ และปริมาณความร้อนที่เหลือเท่ากับ 50 นั้น จะนำไปจับคู่กับกระแสอื่นๆ หรือสร้างเป็นฮีตเตอร์ต่อไปก็ได้ ดังรูปที่ 3.21

3.8 การหาค่า ΔT_{min} ที่เหมาะสมกับข่างาน

ในหัวข้อนี้จะกล่าวถึงการหาค่า ΔT_{min} ที่ทำให้ค่าใช้จ่ายทางด้านพลังงานและค่าใช้จ่าย การสร้างข่างานเครื่องแลกเปลี่ยนความร้อน (โดยวิธีสมดุลพลังงาน และสมดุลมวลสาร) มีค่าน้อยที่สุด โดยจะสามารถหาค่า ΔT_{min} นี้ได้ก่อนการออกแบบหาโครงสร้างของข่างานจริง

3.8.1 จำนวนเครื่องแลกเปลี่ยนความร้อนที่น้อยที่สุด

จากที่เคยกล่าวมาแล้วในบทที่ 2 จำนวนของเครื่องแลกเปลี่ยนความร้อนที่น้อยที่สุดในข่างานนี้สามารถหาค่าได้โดย

$$N_{UNITS} = S - 1 \quad (3.9)$$

เมื่อ N_{UNITS} คือ จำนวนเครื่องแลกเปลี่ยนความร้อนที่น้อยที่สุดในข่างาน

S คือ จำนวนกระแสในข่างานทั้งหมดซึ่งจะรวมกระแสจากยูทิลิตี้ด้วย

ในกรณีถ้าข่างานมีอุณหภูมิพินช์ สมการที่ 3.9 จะเปลี่ยนเป็น

$$N_{UNITS} = (S_{ABOVE PINCH} - 1) + (S_{BELOW PINCH} - 1) \quad (3.10)$$

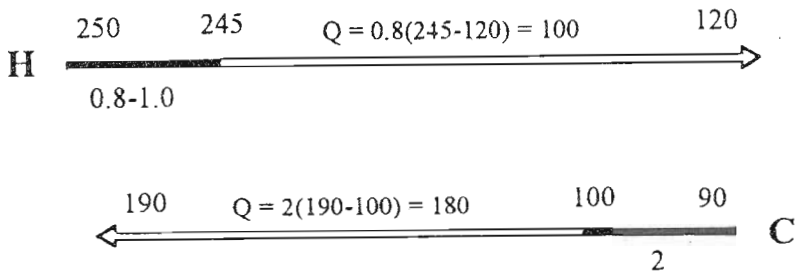
เมื่อ $S_{ABOVE PINCH}$ และ $S_{BELOW PINCH}$ คือ จำนวนกระแสทั้งหมดในข่างานทั้งหมดเหนือ และใต้จุดพินช์

ตัวอย่างที่ 3.2 สำหรับข่างานที่มีอุณหภูมิพินช์ตามรูปที่ 3.22 นี้ให้หา N_{UNITS} กำหนดให้มีกระแสยูทิลิตี้ร้อน และเย็นอย่างละหนึ่งกระแส

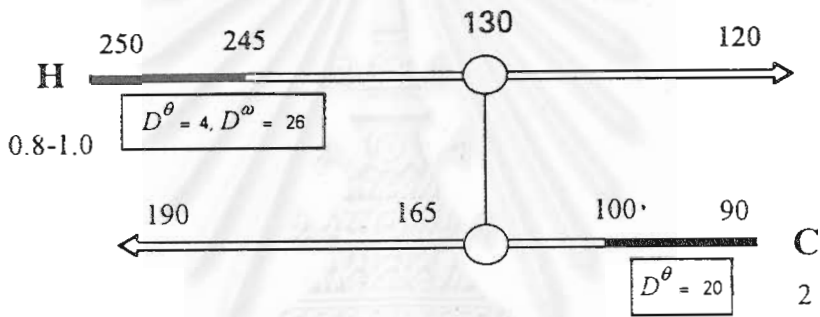
ที่เหนือจุดพินช์จำนวนกระแสร้อนมี 4 กระแส, กระแสเย็น 4 กระแส และกระแสยูทิลิตี้ร้อนอีก 1 กระแส รวมทั้งหมดเท่ากับ 9 กระแส

ใต้จุดพินช์จำนวนกระแสร้อนมี 5 กระแส, กระแสเย็น 4 กระแส และกระแสยูทิลิตี้ร้อนอีก 1 กระแส รวมทั้งหมดเท่ากับ 10 กระแส

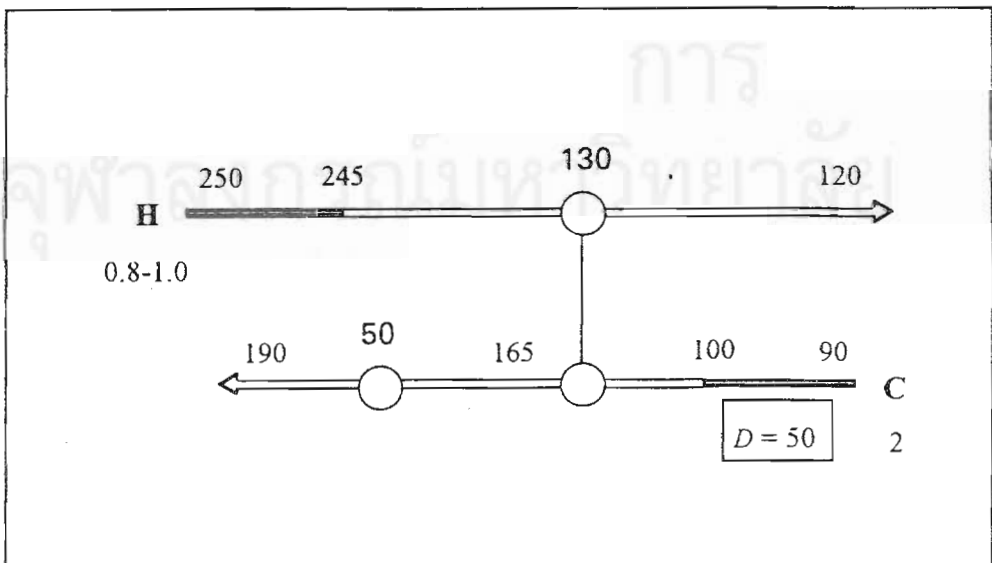
ดังนั้น $N_{UNITS} = (9 - 1) + (10 - 1) = 17$



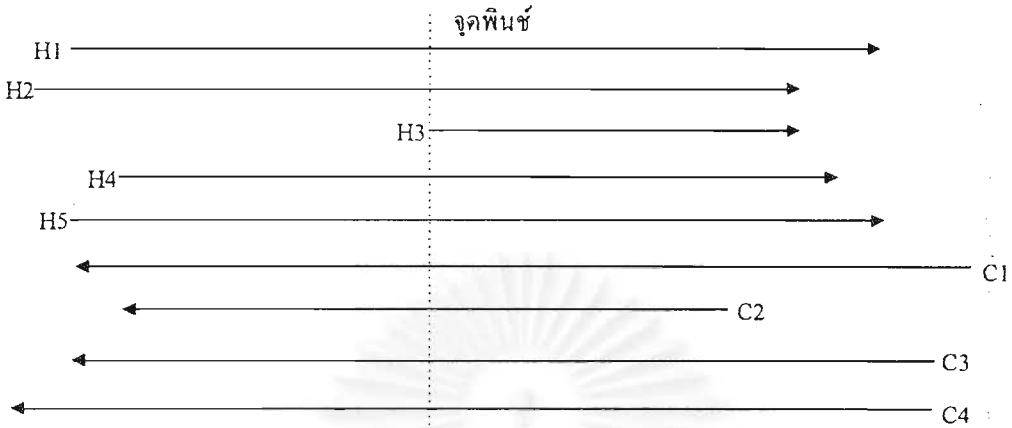
รูปที่ 3.19 แสดงข้อมูลของกระแสในตัวอย่างที่ 3.1



รูปที่ 3.20 แสดงค่าความแปรปรวนของกระแสที่คำนวณได้



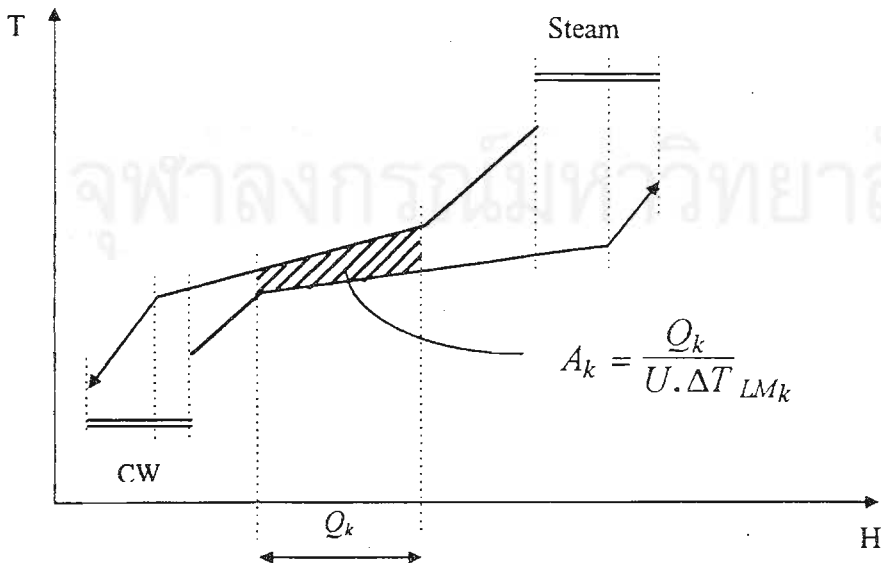
รูปที่ 3.21 แสดงการแลกเปลี่ยนความร้อนของกระแสทั้งสองที่สมบูรณ์แล้ว



รูปที่ 3.22 โคอะแกรมกริดสำหรับตัวอย่างที่ 3.2

3.8.2 พื้นที่ในการแลกเปลี่ยนความร้อน

การหาพื้นที่แลกเปลี่ยนความร้อนทั้งหมดของข้างงานนี้ สามารถหาได้โดยใช้วิธีเส้นโค้งคอมโพสิต แต่จะต้องรวมกระแสจากยูทิลิตี้เข้าไปด้วย และเรียกชื่อใหม่ว่า เส้นโค้งคอมโพสิตสมดุล (Balanced Composite Curves) ซึ่งเส้นโค้งคอมโพสิตสมดุลนี้ จะถูกแบ่งเป็นช่วงเอนทัลปีแนวตั้ง (Vertical Enthalpy Intervals) ดังแสดงในรูป 3.23



รูปที่ 3.23 เส้นโค้งคอมโพสิตสมดุลสำหรับการหาพื้นที่แลกเปลี่ยนความร้อนของข้างงาน

โดยให้การแลกเปลี่ยนความร้อนเป็นแบบสวนทาง (Counter Current) พื้นที่ในการแลกเปลี่ยนความร้อนในช่วง k ใดๆ สำหรับการถ่ายเทความร้อนแนวตั้ง (Vertical Heat Transfer) จะเท่ากับ

$$A_{NETWORKk} = \frac{\Delta H_k}{U \cdot \Delta T_{LMk}}$$

เมื่อ $A_{NETWORKk}$ = พื้นที่แลกเปลี่ยนความร้อนสำหรับการถ่ายเทความร้อนแนวตั้งในช่วง k ใดๆ

ΔH_k = เอนทาลปีในช่วง k

ΔT_{LMk} = อุณหภูมิเฉลี่ยลอการิทึม (Logarithmic Mean Temperature) ในช่วง k

U = สัมประสิทธิ์การถ่ายเทความร้อนรวม (Overall Heat Transfer Coefficient)

ดังนั้นการหาพื้นที่เครื่องแลกเปลี่ยนความร้อนรวมทั้งหมดในข่างาน จะได้จากการรวมพื้นที่ในแต่ละช่วง k เข้าด้วยกัน

$$A_{NETWORKk} = \frac{1}{U} \sum_k^{INTERVALSK} \frac{\Delta H_k}{\Delta T_{LMk}}$$

เมื่อ $A_{NETWORKk}$ = พื้นที่รวมของข่างานในการแลกเปลี่ยนความร้อนแนวตั้ง

K = จำนวนของช่วงเอนทาลปี (Enthalpy Intervals)

จากสมการการหาพื้นที่ที่ได้กล่าวมานี้ ค่า U ได้ถูกสมมติให้คงที่ทุกๆช่วงการคำนวณ แต่ในความเป็นจริงแล้วไม่ใช่ และสามารถคำนวณใหม่ได้จากค่าสัมประสิทธิ์การถ่ายเทความร้อนผ่านฟิล์ม (Film Transfer Coefficient, h) ดังนี้

$$A_{NETWORKk} = \sum_k^{INTERVALSK} \frac{1}{\Delta T_{LMk}} \left[\sum_i^{HOTSTREAMSI} \frac{q_i}{h_i} + \sum_j^{COLDSTREAMSJ} \frac{q_j}{h_j} \right] \quad (3.13)$$

เมื่อ q_i = ปริมาณความร้อนของกระแสร้อน i ในช่วงเอนทาลปีที่ k

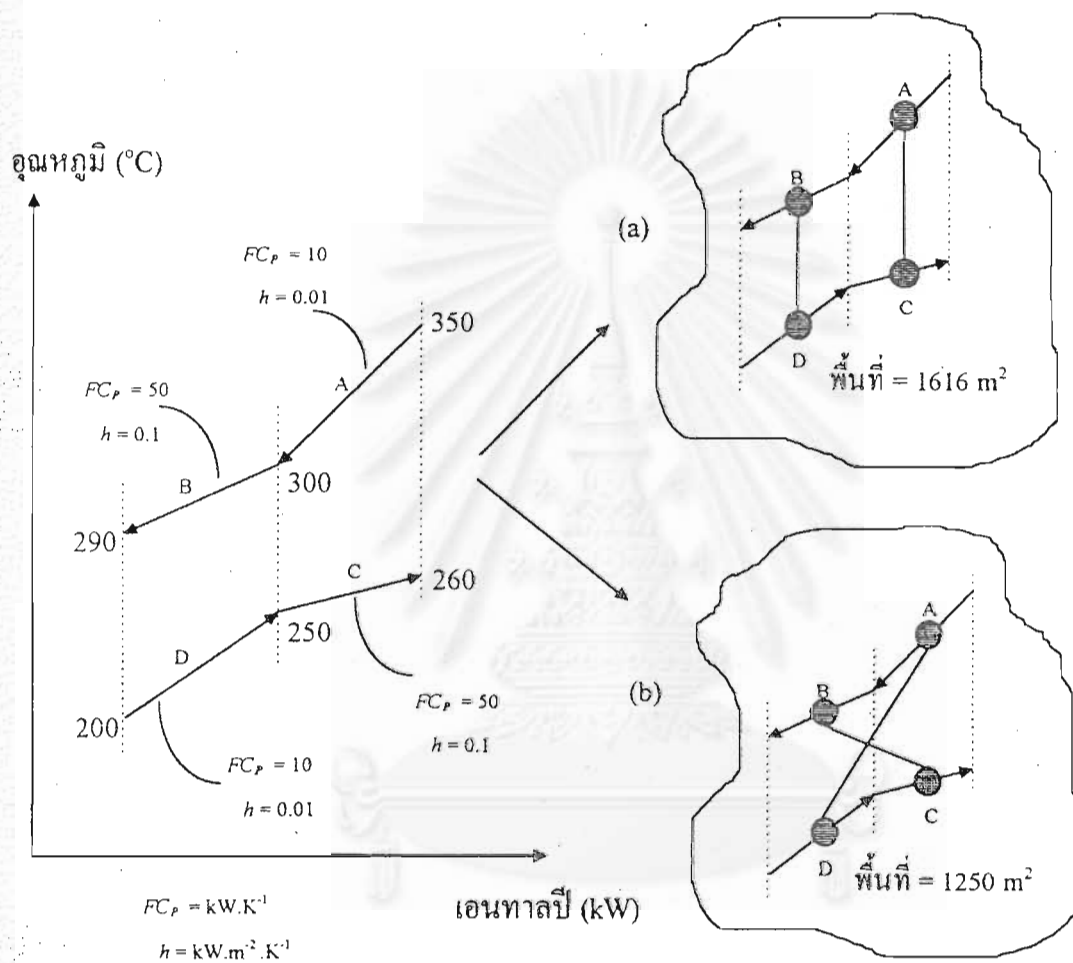
q_j = ปริมาณความร้อนของกระแสเย็น j ในช่วงเอนทาลปีที่ k

h_i, h_j = สัมประสิทธิ์การถ่ายเทความร้อนผ่านฟิล์ม สำหรับกระแสร้อน i และกระแสเย็น j ซึ่งจะรวมความต้านทานจากผนัง (Wall Resistance) และสิ่งสกปรกภายใน (Fouling Resistance) ด้วย

I, J = จำนวนกระแสร้อนและกระแสเย็นทั้งหมดในช่วงเอนทาลปีที่ k

K = จำนวนช่วงเอนทาลปีทั้งหมด

สมการที่ 3.13 จะใช้ในกรณีที่ h ของกระแสร้อนและกระแสเย็นไม่มีค่าต่างกันมากนัก จึงใช้โมเดลการแลกเปลี่ยนความร้อนแนวตั้งได้ แต่ในกรณีที่ค่า h ของกระแสร้อนและกระแสเย็น ในช่วงเอนทาลปีหนึ่งๆ มีค่าต่างกันมาก จะใช้การแลกเปลี่ยนความร้อนไม่ใช่แนวตั้งแทน (Non-vertical Heat Transfer) ดังรูปที่ 3.24 ซึ่งจะทำให้ได้พื้นที่ของข่างานน้อยที่สุด



รูปที่ 3.24 ผลของการแลกเปลี่ยนแบบแนวตั้ง (a) และไม่ใช่แนวตั้ง (b)

3.8.3 ค่าใช้จ่ายของข่ายงาน

ค่าใช้จ่ายของข่ายงานเครื่องแลกเปลี่ยนความร้อนนั้นแบ่งได้สองประเภทคือ ค่าใช้จ่ายด้านพลังงานขณะทำงาน (Operating Cost) และค่าใช้จ่ายในการสร้างเครื่องแลกเปลี่ยนความร้อนของข่ายงาน (Capital Cost)

ค่าใช้จ่ายในการทำงานจะมาจากหน่วยยูทิลิตี้ เช่น ถ้ากำหนดให้ไอน้ำมีค่าใช้จ่าย 2 ล้านบาท/ปี. เมกกะวัตต์ และน้ำหล่อเย็นมีค่าใช้จ่าย 3 แสนบาท/ปี. เมกกะวัตต์ สมมติว่าข่ายงานหนึ่งใช้พลังงานไอน้ำ 5 เมกกะวัตต์/ปี ใช้พลังงานจากน้ำหล่อเย็น 10 เมกกะวัตต์/ปี ค่าใช้จ่ายรวมทั้งหมดจะเท่ากับ $2*5 + 0.3*10 = 13$ ล้านบาท/ปี

สรุปเป็นสมการสำหรับค่าใช้จ่ายดำเนินการด้านพลังงานที่ใช้จากหน่วยยูทิลิตี้ดังนี้

$$\text{ค่าใช้จ่ายจากพลังงาน} = C_H * Q_{H,min} + C_C * Q_{C,min} \quad (3.14)$$

เมื่อ C_H, C_C = สัมประสิทธิ์ค่าใช้จ่ายของยูทิลิตี้ในการให้ความร้อน และยูทิลิตี้ในการดึงความร้อนคิดต่อ 1 ปี ต่อหน่วยความร้อน

$Q_{H,min}, Q_{C,min}$ = พลังงานความร้อนที่ใช้ให้น้อยที่สุดสำหรับให้ความร้อนแก่ข่ายงาน และดึงความร้อนออกจากข่ายงาน

สำหรับค่าใช้จ่ายในการสร้างเครื่องแลกเปลี่ยนความร้อน 1 เครื่อง ซึ่งมีพื้นที่ในการแลกเปลี่ยนความร้อนเท่ากับ A นั้นเป็นไปตามสมการ

$$\text{ค่าใช้จ่ายของเครื่องแลกเปลี่ยนความร้อน} = a + bA^c \quad (3.15)$$

ค่า a, b และ c เป็นค่าคงที่ (Cost law constants) ซึ่งมีค่าต่างๆกันไปตามชนิดวัสดุที่ใช้สร้าง, ความดัน และชนิดของเครื่องแลกเปลี่ยนความร้อน

สำหรับการประมาณค่าใช้จ่ายในการสร้างข่ายงานเครื่องแลกเปลี่ยนความร้อน N เครื่อง (Network Capital Cost) คำนวณได้จากสมการ 3.16

$$\text{ค่าใช้จ่ายสร้างข่ายงานรวม} = N[a + b(A_{NETWORK} / N)^c] \quad (3.16)$$

ถ้าคิดถึงจำนวนปี (n) ที่จะต้องจ่ายงานให้กับธนาคาร โดยมีดอกเบี้ยอัตราเท่ากับ i แล้วค่าใช้จ่ายสร้างข่ายงานรวมคิดเป็นรายปี (Annualized Capital Cost) จะคำนวณได้จากสมการที่ 3.17

$$\text{ค่าใช้จ่ายสร้างข่ายงานรวมรายปี} = \text{ค่าใช้จ่ายสร้างข่ายงานรวม} \times \frac{i(1+i)^n}{(1+i)^n - 1} \quad (3.17)$$

ตัวอย่างที่ 3.3 สำหรับข้อมูลของกระแสดำเนินการที่ 3.3 ให้คำนวณหา ΔT_{min} ที่ทำให้ข่างานเครื่องแลกเปลี่ยนความร้อนประหยัดค่าใช้จ่ายมากที่สุด

กำหนดให้ ใช้น้ำมีอุณหภูมิ 240 °C และน้ำหล่อเย็นมีอุณหภูมิ 20 °C

ค่าใช้จ่ายของไอน้ำ = 120,000 (\$·MW⁻¹/year)

ค่าใช้จ่ายของน้ำหล่อเย็น = 10,000 (\$·MW⁻¹/year)

ค่าใช้จ่ายในการสร้างเครื่องแลกเปลี่ยนความร้อน = 40,000 + 500*A*

จากสมการ 3.17 ให้ $\epsilon = 0.1$ และ $n = 5$

การคำนวณต่อไปนี้จะใช้ตัวอย่าง $\Delta T_{min} = 10^\circ\text{C}$ ซึ่งจะได้ค่า $Q_{H,min} = 7.5$ และ $Q_{C,min} = 10$ MW จำนวนอุณหภูมิขาออกของไอน้ำได้ 239 °C และอุณหภูมิขาออกของน้ำหล่อเย็นได้ 30 °C ดังแสดงข้อมูลไว้ในตารางที่ 3.3

วิธีทำ

ในขั้นแรกจะสร้างเส้นโค้งคอมโพสิตสมมูล โดยใช้วิธีเช่นเดียวกับการสร้างเส้นโค้งคอมโพสิต และจะรวมข้อมูลของกระแสไอน้ำ และน้ำหล่อเย็นเข้าไปด้วย

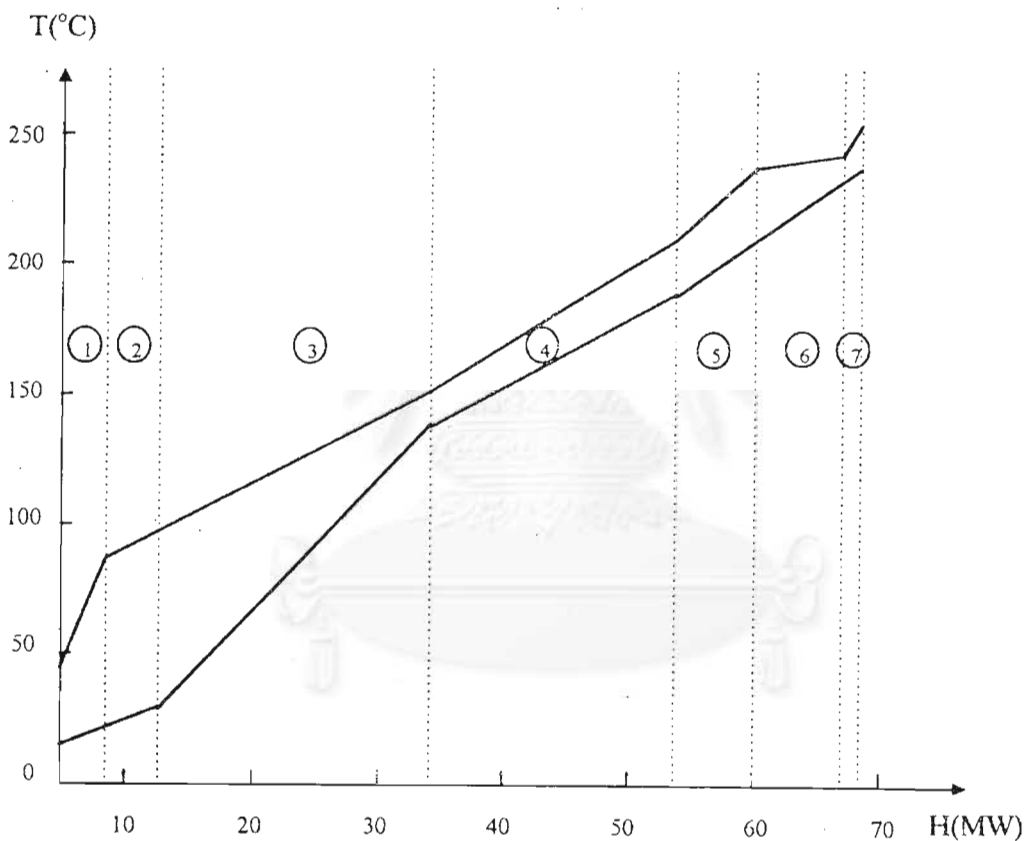
จากรูปที่ 3.25 สังเกตได้ว่า ช่วงเอนทาลปีนั้นแบ่งได้จากการเปลี่ยนแปลงความชันของเส้นกราฟเส้นใดเส้นหนึ่ง รูปที่ 3.26 แสดงโคอะแกรมของแต่ละช่วงเอนทาลปีในโคอะแกรมกริด ซึ่งจะแบ่งขอบเขต หรือช่วงในกระแสต่างๆ ได้ชัดเจนขึ้น

ตารางที่ 3.3 ข้อมูลในการหาพื้นที่แลกเปลี่ยนความร้อนของข่างานของตัวอย่างที่ 3.3

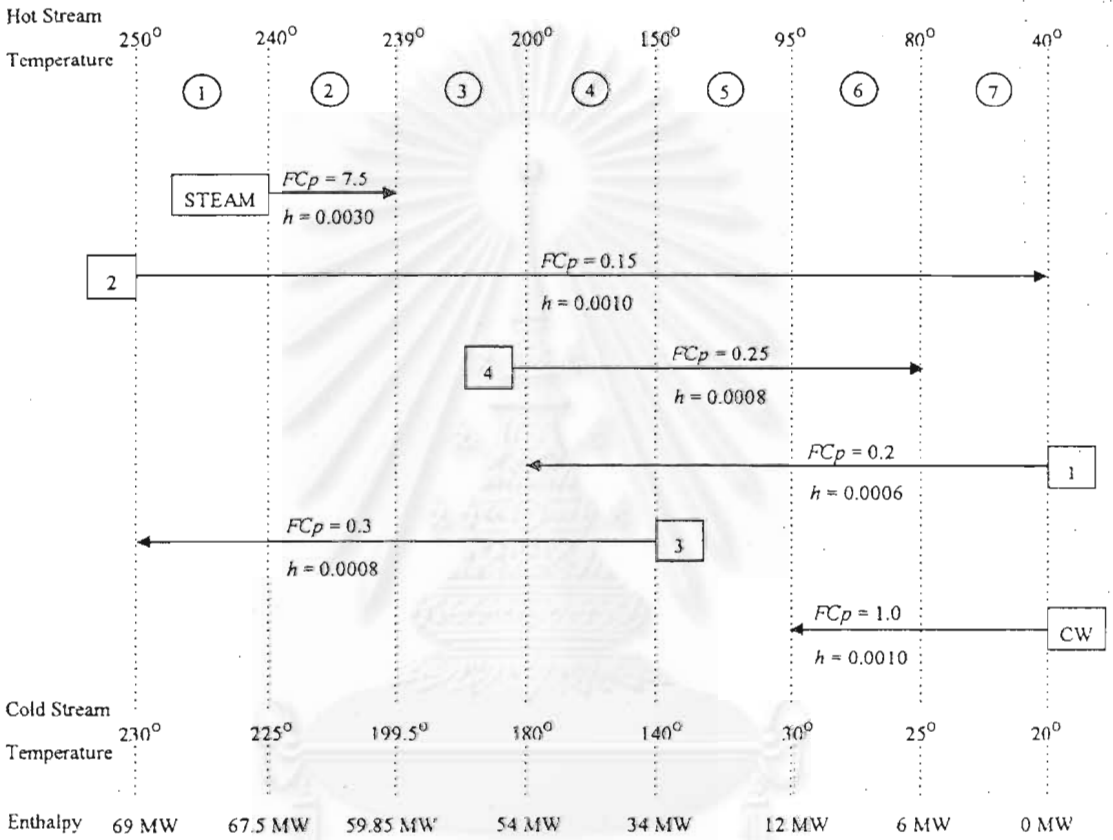
Stream	Supply Temp (°C)	Target Temp (°C)	ΔH (MW)	FC_p (MW·°C ⁻¹)	h (MW·m ² ·°C)
1. Reactor1 feed	20	180	-32.0	0.2	0.0006
2. Reactor1 product	350	40	-31.5	0.15	0.0010
3. Reactor 2 feed	140	230	27.0	0.3	0.0008
4. Reactor2 product	200	80	-30.0	0.25	0.0008
5. Steam	240	239	-7.5	7.5	0.0030
6. Cooling water	20	30	10.0	1.0	0.0010

ผลของการคำนวณหาพื้นที่รวมน้อยที่สุดของข่ายงานเครื่องแลกเปลี่ยนความร้อนสำหรับ $\Delta T_{min} = 10^\circ\text{C}$ เท่ากับ นี้ ได้แสดงไว้ในตารางที่ 3.4 สำหรับการคำนวณหาพื้นที่ และค่าใช้จ่ายของช่วง ΔT_{min} ตั้งแต่ $2-14^\circ\text{C}$ ได้แสดงผลออกมาไว้ในตารางที่ 3.5

เมื่อนำค่าใช้จ่ายของการทำงานและ การสร้างข่ายงานมารวมกัน จะได้ค่าใช้จ่ายรวม (Total Cost) นำค่าใช้จ่ายรวมมาพล็อตกับค่าของ ΔT_{min} จะได้กราฟดังรูปที่ 3.27



รูปที่ 3.25 เส้นโค้งคอมโพสิตสมดุลของตัวอย่างที่ 3.3 โดยให้ $\Delta T_{min} = 10^\circ$



รูปที่ 3.26 แสดงการแบ่งช่วงเอนทัลปีในโคอะเคอเรนทรีค

จุฬาลงกรณ์มหาวิทยาลัย

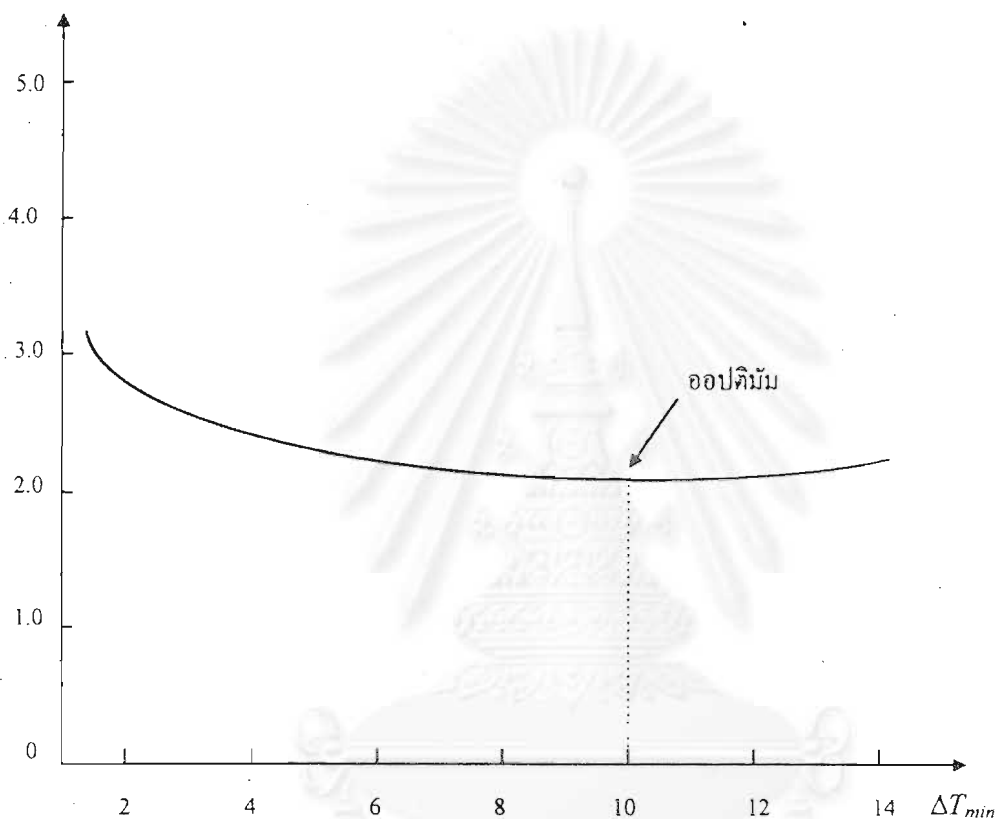
ตารางที่ 3.4 แสดงข้อมูลในการคำนวณที่ $\Delta T_{min} = 10^\circ$

Enthalpy intervals	ΔT_{LMk}	Hot streams $\Sigma(q_i/h_i)_k$	Cold Streams $\Sigma(q_j/h_j)_k$	A_k
1	17.38	1,500	1,875.0	194.2
2	25.30	2,650	9,562.5	482.7
3	28.65	5,850	7,312.5	459.4
4	14.43	23,125	28,333.3	3566.1
5	29.38	25,437.5	36,666.7	2113.8
6	59.86	6,937.5	6,666.7	227.3
7	34.60	6,000	6,666.7	366.1
			$\Sigma A_k =$	7,409.6

ตารางที่ 3.5 เปรียบเทียบค่าใช้จ่ายของข่ายงานที่ ΔT_{min} ต่างต่าง ๆ

ΔT_{min}	$Q_{H,min}$ (MW)	$Q_{C,min}$ (MW)	Annual hot utility cost (10^6 \$/year)	Annual Cold utility cost (10^6 \$/year)	$A_{NETWORK}$ (m^2)	N_{UNITS}	Annualized capital cost (10^6 \$/year)	Annualized total cost (10^6 \$/year)
2	4.3	0.516	6.8	0.068	15,519	7	2.121	2.705
4	5.1	0.612	7.6	0.076	11,677	7	1.614	2.302
6	5.9	0.708	8.4	0.084	9,645	7	1.346	2.138
8	6.7	0.804	9.2	0.092	8,336	7	1.173	2.069
10	7.5	0.900	10.0	0.100	7,410	7	1.051	2.051
12	8.3	0.996	10.8	0.108	6,716	7	0.960	2.064
14	9.1	1.092	11.6	0.116	6,174	7	0.888	2.096

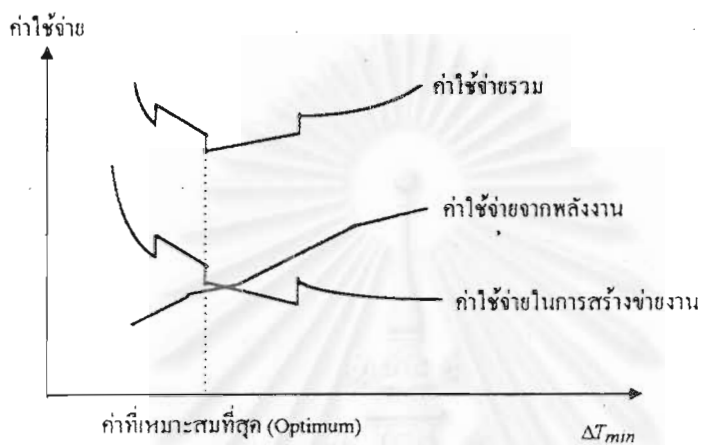
ค่าใช้จ่ายรวม
($\$ \cdot 10^6 / \text{year}$)



รูปที่ 3.27 แสดงค่า ΔT_{min} ในข่างานตัวอย่าง 3.3 ที่ประหยัดค่าใช้จ่ายมากที่สุด

จากรูปจะเห็นว่าค่า ΔT_{min} ที่ทำให้ข่างานประหยัดค่าใช้จ่ายมากที่สุดคือ 10°C สังเกตได้จากจุดต่ำที่สุดของกราฟ สำหรับการออกแบบปัญหาที่ซับซ้อนกว่านี้ เส้นกราฟของค่าใช้จ่ายนั้นจะเปลี่ยนแปลงไปเป็นแบบขั้น (Step Changes) ดังที่แสดงไว้ในรูปที่ 3.28 ที่เป็นเช่นนี้เนื่องมาจากการเปลี่ยนแปลงของค่า N_{UNITS} มีค่าไม่เท่ากันตลอดในช่วง ΔT_{min}

สรุปแล้วสำหรับการหาค่า ΔT_{min} ที่ทำให้ข่างานประหยัดค่าใช้จ่ายมากที่สุดนี้ เราสามารถหาค่าได้ก่อนการออกแบบโครงสร้างของข่างาน โดยใช้ค่าพลังงานจากหน่วยยูนิตที่คำนวณได้จากตารางปัญหากับค่าพื้นที่การแลกเปลี่ยนความร้อนที่น้อยที่สุดของข่างานทางทฤษฎี (Minimum Area) จากเส้นโค้งคอมโพสิตสมดุล ซึ่งจะหาค่าผิดพลาดในทางปฏิบัติไม่เกิน 5 %



รูปที่ 3.28 ค่าใช้จ่ายของช่างานเปลี่ยนแปลงแบบขั้นบันได

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 4

การออกแบบและการสร้างโปรแกรม



ในบทนี้จะกล่าวถึงการออกแบบโปรแกรมในส่วนที่สำคัญต่อการสร้างข่ายงานเครื่องแลกเปลี่ยนความร้อนตามที่ได้กล่าวมาในบทที่แล้ว และตัวอย่างการใช้โปรแกรมด้วย อย่างไรก็ตามเนื่องจากโปรแกรมการออกแบบข่ายงานนี้เขียนด้วยภาษาเชิงวัตถุ ดังนั้น จึงขอกล่าวความสำคัญของภาษาเชิงวัตถุก่อนที่จะกล่าวถึงการออกแบบโปรแกรมต่อไป

4.1 ภาษาเชิงวัตถุ

การเขียนโปรแกรมสมัยก่อนนั้น ผู้เขียนโปรแกรมจะใช้ภาษาเครื่องอันประกอบไปด้วยเลข 0 และ 1 เท่านั้น การพัฒนาโปรแกรมจะประสบปัญหาที่ย่างยากมากมาย ต่อมาภาษาแอสเซมบลี ข้อดีคือสามารถทำงานได้รวดเร็ว และติดต่อกับระดับฮาร์ดแวร์ของเครื่องได้สะดวกแต่ก็ยังยากต่อผู้ใช้ทั่วไปที่จะเข้าใจได้ ต่อมาเริ่มเกิดภาษาชั้นสูงซึ่งใช้คำสั่งประกอบด้วยคำพูดหรือสัญลักษณ์ที่ใช้กันในชีวิตประจำวันและคนทั่วไปเข้าใจง่ายขึ้น เช่น ภาษาฟอร์แทรน ซึ่งมีผู้นิยมใช้กันมากแต่ก็เหมาะกับงานเฉพาะด้านการคำนวณเท่านั้น ในที่สุดได้มีผู้พัฒนาภาษาที่ใช้งานได้คล่องตัวและกว้างขึ้นกว่าเดิม ตัวอย่างเช่น ภาษาเบสิก ปาสคาล และภาษา C เป็นต้น ซึ่งประสบความสำเร็จมากและยังเป็นภาษาที่นิยมใช้กันจนถึงปัจจุบันนี้ ภาษาชั้นสูงเหล่านี้เป็นภาษาที่เน้นโครงสร้าง การออกแบบโปรแกรมจะเน้นไปที่วิธีการและหน้าที่ส่วนต่างๆ ของโปรแกรม (ฟังก์ชัน หรือ สับโปรแกรม) โดยจะแตกโปรแกรมขนาดใหญ่เป็นโปรแกรมขนาดเล็กๆ ที่จัดการได้ง่าย จากนั้นจึงค่อยมาคำนึงถึงตัวแปรที่ใช้ อันเป็นลักษณะการออกแบบโปรแกรมจากบนลงล่าง ถ้าโปรแกรมมีขนาดเล็ก การออกแบบโปรแกรมแบบนี้จะไม่มีปัญหาอันใด แต่เมื่อโปรแกรมมีขนาดใหญ่และซับซ้อนขึ้น การพัฒนาและการแก้ไขปรับปรุงโปรแกรมจะทำได้ลำบาก ประสิทธิภาพในการทำงานจะลดลงอย่างมาก

ต่อมาได้มีการปรับปรุงการเขียนโปรแกรมแนวใหม่ที่เรียกว่า การเขียนโปรแกรมเชิงวัตถุ (Object-Oriented Program หรือ OOP) การเขียนโปรแกรมแบบนี้ไม่เน้นอัลกอริธึม แต่จะให้ความสำคัญแก่ลักษณะข้อมูลและตัวแปรก่อน ข้อมูลจะถูกใช้เป็นหลักในการแบ่งโปรแกรมออกเป็นส่วนย่อยๆ การออกแบบโปรแกรมเชิงวัตถุจะเกี่ยวข้องกับการนิยามคลาส (class) และวัตถุ (Object) เท่านั้น ซึ่งในส่วนการทำงานของโค้ดจะยังไม่ถูกคำนึงถึงเลยในการออกแบบโดยเป้าหมายหลักของการพัฒนาโปรแกรมเชิงวัตถุ คือ

- ทำให้การพัฒนาโปรแกรมใช้เวลาสั้นลง และต้นทุนต่ำลงโดยการใช้คลาสที่สามารถนำกลับมาใช้ได้อีก และสร้างสับคลาส (Subclass) ขึ้นมาเพื่อใช้ในการแก้ปัญหา
- ทำให้ต้นทุนในการบำรุงรักษาโปรแกรมต่ำลง เพราะสามารถหาจุดที่ต้องการเปลี่ยนแปลงในโปรแกรมได้ และการเปลี่ยนแปลงไม่ทำให้เกิดผลกระทบไปยังภายนอกคลาสเลย

ระบบโปรแกรมเชิงวัตถุมีความน่าเชื่อถือมาก เพราะการประกอบส่วนต่างๆ ของโปรแกรมขึ้นเป็นระบบได้ทำในระดับสูงตั้งแต่การออกแบบ ซึ่งสามารถตรวจสอบได้ก่อนที่รายละเอียดส่วนใหญ่ในระดับต่ำจะถูกเขียนขึ้น สิ่งเหล่านี้จึงเป็นตัวพิสูจน์ความน่าเชื่อถือเป็นอย่างดี ต่อไปจะได้กล่าวถึงสิ่งสำคัญอันเป็นคุณสมบัติของโปรแกรมเชิงวัตถุ

4.1.1 คลาสและวัตถุ

คลาสเป็นข้อกำหนดที่ระบุรายละเอียดของข้อมูล (Type) ให้แก่วัตถุ ซึ่งวัตถุคือ ข้อมูลจริงที่ถูกสร้างขึ้นตามข้อกำหนดนั้น โดยทั่วไป นอกจากที่คลาสจะกำหนดลักษณะข้อมูลที่ใช้เพื่อสร้างเป็นวัตถุแล้ว ยังสามารถกำหนดวิธีปฏิบัติสำหรับดำเนินงานกับข้อมูลนั้นอีกด้วย ยกตัวอย่างเช่น ถ้าเราต้องการสร้างโปรแกรมวาดภาพกล่องสี่เหลี่ยมสามมิติขึ้นมา สิ่งแรกที่ต้องทำก็คือ จะต้องสร้างคลาสขึ้นมาคลาสหนึ่ง โดยคลาสนี้จะกำหนดรายละเอียดต่างๆ ที่จำเป็นต่อการวาดรูปไว้ ส่วนที่เป็นข้อมูลอาจประกอบไปด้วยข้อมูลของตำแหน่ง ความสูง ความกว้าง ความยาว สี ทิศทาง หรือมุมมอง เป็นต้น ส่วนข้อกำหนดที่เป็นการปฏิบัติจะประกอบไปด้วยวิธีการเคลื่อนที่ การเปลี่ยนขนาด การวาด หมุน เปลี่ยนสี และคัดลอกรูป เป็นต้น จากนั้นจึงจะสร้างโปรแกรมขึ้นมาเพื่อวาดรูปกล่อง โปรแกรมจะสร้างวัตถุที่มีลักษณะตามที่ได้กำหนดไว้ในคลาสนี้ขึ้นมา โดยวัตถุนั้นจะมีข้อมูลของรูปครบบตามที่กำหนดไว้ในคลาส ถ้าต้องการวาดรูปกล่องสองรูปที่เป็นอิสระต่อกัน โปรแกรมจะสร้างวัตถุขึ้นมาสองตัวสำหรับกล่องแต่ละรูป โดยใช้นิยามคลาสเดียวกัน

4.1.2 เอนแคปซูเลชัน (Encapsulation)

คุณสมบัติเอนแคปซูเลชันของคลาสที่กำหนดให้วัตถุ คือ

1. กำหนดขอบเขตที่ชัดเจนให้แก่วัตถุ
2. กำหนดส่วนการติดต่อภายนอกกับวัตถุตัวอื่นๆ

หมายความว่า เราสามารถกำหนดให้ข้อมูลหรือฟังก์ชันในวัตถุหนึ่ง สามารถที่จะให้โปรแกรมภายนอกหรือวัตถุอื่นๆเรียกใช้งานหรือไม่ก็ได้ ซึ่งจะป้องกันการใช้และแก้ไขข้อมูลที่ไม่ถูกต้อง

4.1.3 อินเฮริเทนซ์ (Inheritance)

อินเฮริเทนซ์ หมายถึง การที่เราสามารถสร้างคลาสใหม่โดยสืบทอดคุณสมบัติจากคลาสเก่า เช่น เราสร้างคลาสของจักรวาลเป็นคลาสเริ่มต้น ต่อมาสร้างคลาสใหม่เป็นคลาสเกี่ยวกับระบบสุริยะสืบทอดจากคลาสจักรวาล และในที่สุดอาจสร้างคลาสของดาวเคราะห์สืบทอดจากคลาสระบบสุริยะ เพื่อแบ่งรายละเอียดต่อไปอีก เป็นต้น

4.1.4 โพลิมอร์ฟิซึม (Polymorphism)

โพลิมอร์ฟิซึม ในการเขียนโปรแกรมก็คือ ผู้ใช้สามารถใช้ชื่อต่างๆซ้ำกันได้ แต่จุดประสงค์ของการใช้นั้นจะต่างกันไป ตัวอย่างเช่น สร้างฟังก์ชันสำหรับวาดเส้นชื่อ *LINE* ฟังก์ชันนี้จะใช้วาดเส้นตรงบนผิวระนาบใดๆ ต่อมาในกรณีที่ต้องสร้างฟังก์ชันวาดเส้นตรงบนผิวทรงโค้งใดๆ เช่น ทรงกลม กรณีดังกล่าวนี้ก็ยังสามารถสร้างฟังก์ชันใหม่ให้มีชื่อว่า *LINE* เช่นเดียวกันได้ แม้ฟังก์ชันทั้งสองจะมีชื่อซ้ำกัน แต่ตัวแปรหรือข้อมูลภายในจะต่างกันซึ่งจะทำให้คอมพิวเตอร์สามารถเลือกฟังก์ชันให้ผู้เขียนโปรแกรมได้ถูกต้อง

4.2 การออกแบบคลาสในข่ายงาน

ในโปรแกรมจะมีคลาสสองคลาสใหญ่ๆ คือคลาสการออกแบบเบื้องต้น และคลาสการออกแบบโครงสร้าง คลาสแรกมีหน้าที่ในการเตรียมหาค่า $Q_{H,min}$ $Q_{C,min}$ และอุณหภูมิพินช์แสดงผลของตารางปัญหา ไดอะแกรมกริด เส้นโค้งคอมโพสิต เส้นโค้งคอมโพสิตสมดุค เส้นโค้งแกรนด์คอมโพสิต เลือกโหมดการคำนวณแบบยึดหยุ่น หรือแบบไม่ยึดหยุ่น และหาค่า ΔT_{min} ที่เหมาะสมกับข่ายงานเป็นต้น คลาสที่สองทำหน้าที่หาโครงสร้างของข่ายงานที่เป็นไปได้ทั้งหมดออกมาและแสดงผล รวมทั้งหาข่ายงานที่ประหยัดค่าใช้จ่ายที่สุดในกรณีที่ได้คำตอบหลายโครงสร้างด้วยกัน สำหรับการทำงานของโปรแกรมโดยรวมให้ดูรูปที่ 4.1

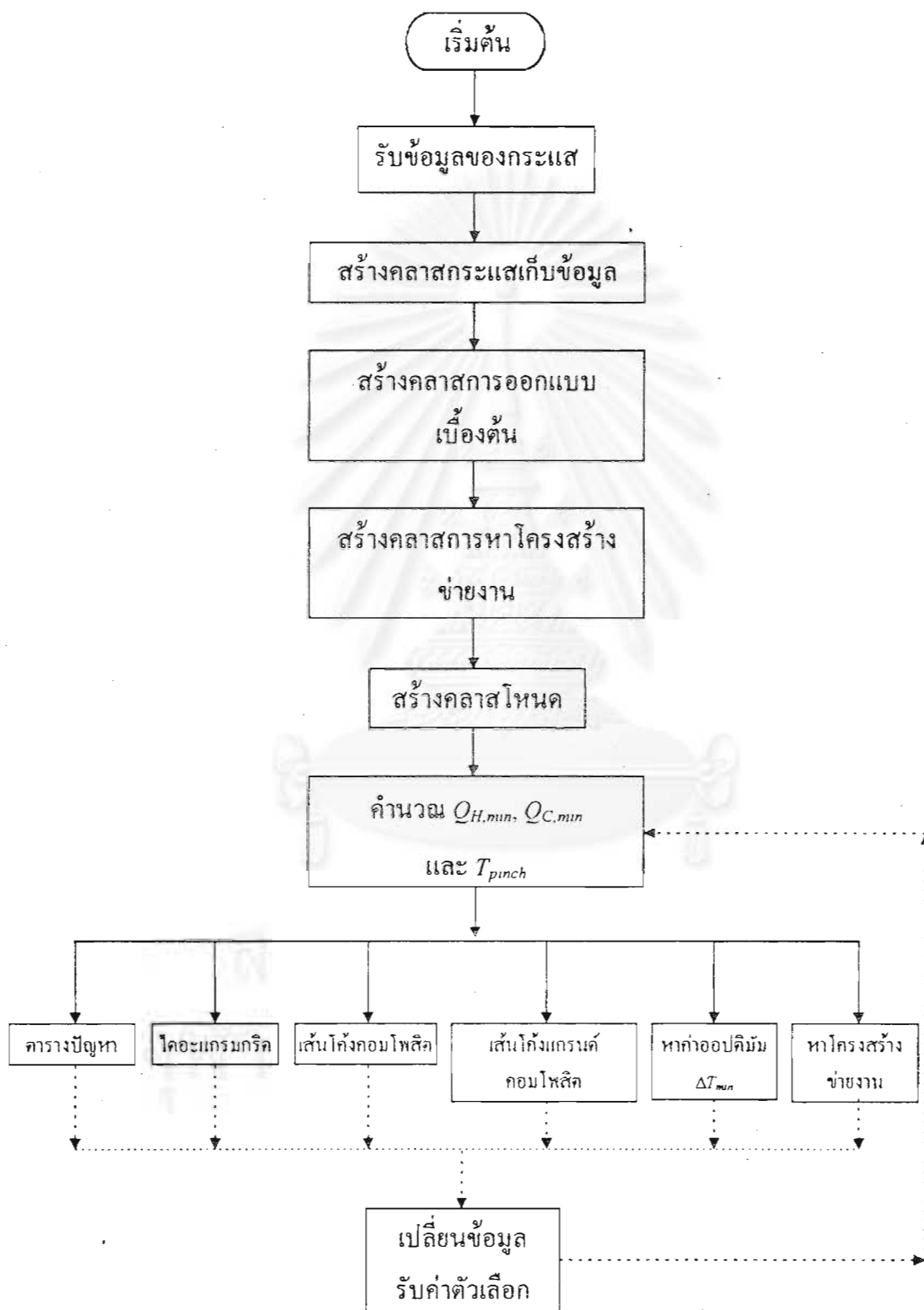
นอกจากนี้ ยังมีคลาสที่สำคัญอีกสองคลาส คือ คลาสกระแส และ คลาสโหนด สำหรับคลาสกระแส เป็นคลาสที่ใช้เก็บข้อมูลที่จำเป็นของกระแสไว้ เช่น อุณหภูมิขาเข้า อุณหภูมิขาออก ความแปรปรวน ความร้อน และผลคูณของอัตราการไหลและความจุความร้อนจำเพาะเป็นต้น ซึ่งเวลาใช้งานจะใช้คลาสนี้สร้างวัตถุกระแสร้อน และวัตถุกระแสเย็นตามนิยามของคลาสกระแสขึ้นมา

สำหรับคลาสโหนด เป็นคลาสที่เกิดสภาวะของการแลกเปลี่ยนความร้อนในข่ายงานไว้หรือเรียกว่าเป็นคลาสของเครื่องแลกเปลี่ยนความร้อนก็ได้ นอกจากจะมีพารามิเตอร์ที่สำคัญหลายตัวแล้ว ยังมีคลาสกระแสเป็นสมาชิกอยู่ด้วย ซึ่งก่อนที่ได้จะอธิบายต่อไป จะขอกล่าวถึงหลักการการออกแบบคลาสโหนดก่อน

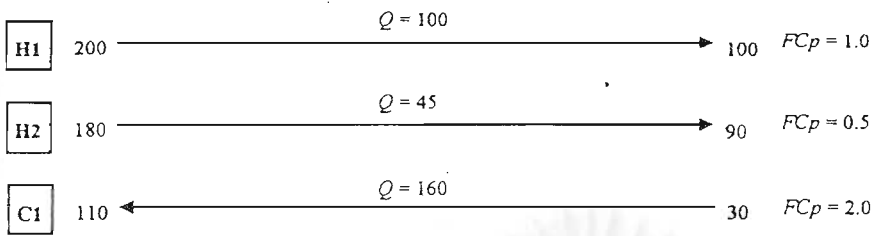
ตัวอย่างที่ 4.1 ให้พิจารณาการแลกเปลี่ยนความร้อนระหว่างกระแสที่มีข้อมูลดังรูปที่ 4.2

การออกแบบจะมองว่า ณ สภาวะการแลกเปลี่ยนความร้อนหนึ่งๆหรือเมื่อใดก็ตามที่ได้ใช้กระบวนการจับคู่สร้างเครื่องแลกเปลี่ยนความร้อน, ฮีตเตอร์ หรือคูลเลอร์ขึ้น ที่สภาวะนั้นๆ ถือเป็นโหนดๆ หนึ่ง และจะใช้คลาสโหนดเก็บข้อมูลของทุกกระแสที่สภาวะนั้นไว้ โดยโหนดที่ 0 จะเป็นโหนดเก็บข้อมูลเริ่มต้นของทุกกระแสก่อนการออกแบบ และจะสร้างโหนดที่ 1, 2, 3 เป็นลำดับต่อไป สำหรับเก็บข้อมูลของทุกกระแสหลังการใช้กระบวนการจับคู่ต่างๆ

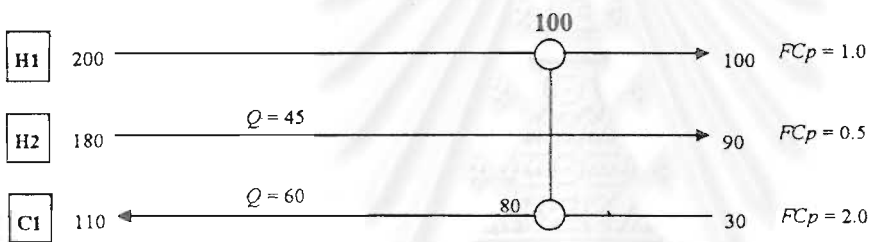
ในที่นี้ข่างานเป็นปัญหาทางด้านความร้อน (สร้างฮีตเตอร์ในข่างาน) ให้ ΔT_{min} มีค่าเท่ากับ 10° เริ่มการจับคู่แลกเปลี่ยนความร้อนระหว่างกระแส H1 และ C1 โดยใช้แพทเทิร์นที่ 2 คือแพทเทิร์น A[H] (ในที่นี้ แพทเทิร์นที่ 1 ไม่สามารถใช้ได้) ดังรูปที่ 4.3 สถานะของกระแสที่ถูกจับแลกเปลี่ยนความ



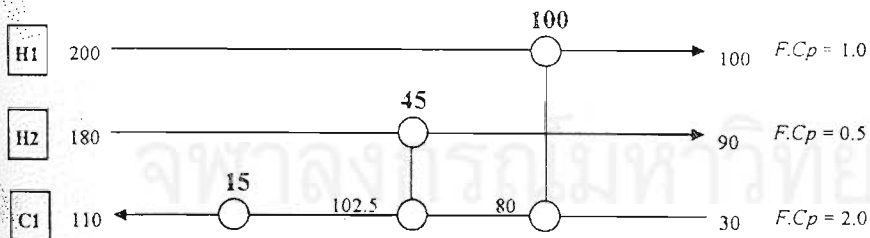
รูปที่ 4.1 แสดงการทำงานของโปรแกรมโดยรวม



รูปที่ 4.2 ข้อมูลกระแสสำหรับสร้างข่ายงานแลกเปลี่ยนความร้อน



รูปที่ 4.3 โหนดที่ 1 ใช้แพทเทิร์น A[H]

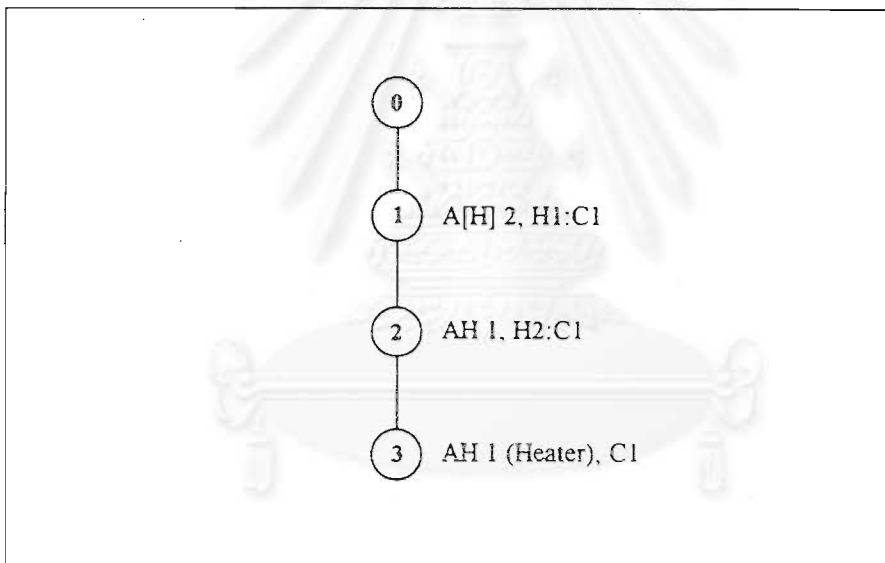


รูปที่ 4.4 แสดงโครงสร้างที่สมบูรณ์ของข่ายงาน

ร้อนจะเปลี่ยนไปนั่นคือ กระแส H1 ความร้อนเหลือ = 0 , กระแส C1 ความร้อนเหลือ = 140 และ อุณหภูมิขาเข้าคำนวณใหม่ได้เป็น 80°

ที่สถานะนี้จะสร้างโหนดที่ 1 เก็บข้อมูลต่างๆไว้ โดยมีโหนด 0 เป็นโหนดแม่ และถ้าสามารถสร้างโหนดที่ 2 ได้ โหนดที่ 1 ก็จะมีสถานะเป็นโหนดแม่ของโหนดที่ 2 อีก และโดยการใช้แพทเทิร์น AH ทำการแลกเปลี่ยนความร้อนระหว่างกระแส H2 และกระแส C1 จะได้โหนดที่ 2 และโหนดที่ 3 ดังรูปที่ 4.4

เนื่องจากโหนดจะเก็บสถานะของกระแสต่างๆไว้ โดยปกติจะใช้ 1 โหนดต่อ 1 แพทเทิร์นการจับคู่ แต่จะสังเกตได้ว่าแพทเทิร์น AH (หรือ BH) นี้ พิเศษกว่าแพทเทิร์นอื่นๆที่มีการสร้างฮีตเตอร์ (หรือคูลเลอร์) ด้วยภายในตัวแพทเทิร์นเอง แต่ในที่นี้การออกแบบโปรแกรมจะทำการสร้างโหนดเมื่อได้สร้างเครื่องแลกเปลี่ยนความร้อนใดๆขึ้น (รวมทั้งฮีตเตอร์หรือคูลเลอร์) ดังนั้นในแพทเทิร์นดังกล่าว จะสร้างโหนดไว้สองโหนดต่อ 1 แพทเทิร์น ขณะที่แพทเทิร์นอื่นๆ จะสร้างโหนดเพียงโหนดเดียว สำหรับโหนดที่ได้สร้างมาทั้งสี่โหนดนี้สามารถสร้างเป็นไดอะแกรมได้ดังรูปที่ 4.5



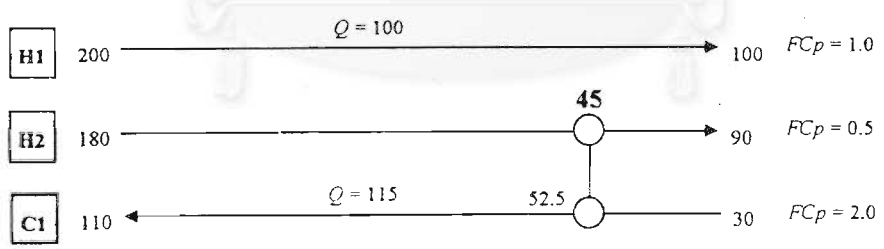
รูปที่ 4.5 แสดงลำดับและสถานะของโหนดต่างๆ โดยตัวเลข 0 ถึง 3 แสดงหมายเลขประจำโหนด หมายเลขหลังแพทเทิร์นหมายถึงลำดับแพทเทิร์นตามที่ได้กล่าวไว้ในบทที่ 3

ในการสร้างโหนดใหม่ (คิดจากบนลงล่าง) สำหรับการจับคู่ระหว่างกระแสน้ำร้อนและกระแสน้ำเย็น คู่ไหนคู่ใต้นั้น จะใช้แพทเทิร์นเริ่มต้นจาก AH หรือจากลำดับที่ 1 ไปเสมอ แต่ถ้าเป็นการสร้างโหนดใหม่โดยย้อนกลับไปหาโหนดเก่าโหนดใดๆ (หรือโหนดแม่) ที่มีอยู่แล้ว จะใช้แพทเทิร์นและลำดับ

กระแสนี้ไปจากโหนดลูกล่าสุดของโหนดแม่ นั่นๆ (ที่ทำเช่นนี้เพื่อหาโครงสร้างของข่ายงานอื่นๆที่เป็นไปได้ก็เนื่องจากข่ายงานอาจมีได้หลายคำตอบ).

ตัวอย่างเช่น สำหรับโหนดที่พบคำตอบของข่ายงานที่สมบูรณ์แล้ว หรือสำหรับโหนดที่ไม่สามารถหาคำตอบต่อไปได้อีก (และจะถือว่าโหนดๆ นั้นตายแล้ว) ซึ่งโหนดในลักษณะดังกล่าวไม่สามารถสร้างโหนดในทิศทางจากบนลงล่างได้ ให้ทำการย้อนทิศทางการสร้างโหนดโดยย้อนกลับจากล่างขึ้นบนจากโหนดดังกล่าวไปที่โหนดแม่ (Back tracking) และที่โหนดแม่นี้จะเป็นโหนดเริ่มในการสร้างโหนดใหม่อีกครั้ง โดยจะใช้แพทเทิร์นและลำดับของคู่กระแสถัดไปจากโหนดลูกล่าสุด โดยปกติจะย้อนกลับเพียง 1 โหนดไปที่โหนดแม่ แต่ถ้าโหนดที่กำลังจะย้อนนั้นเป็นฮีดเตอร์หรือ คูลเลอร์ให้ทำการย้อนโหนดไปสองโหนด นอกจากนี้ถ้าโหนดที่เป็นคำตอบนั้น เป็นคำตอบที่ซ้ำกับตอบซ้ำ และโหนดที่ไม่สามารถให้คำตอบได้อีกนี้ จะต้องถูกลบออกไปก่อนที่จะสร้างโหนดใหม่ต่อไป เพื่อประหยัดหน่วยความจำคอมพิวเตอร์

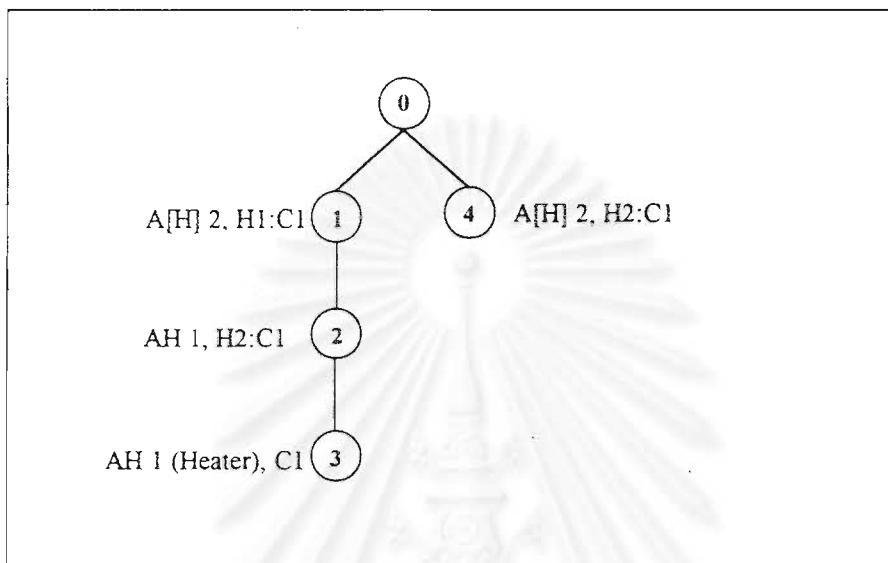
รูปที่ 4.5 จะทำการย้อนจากโหนดที่ 3 (ฮีดเตอร์) ไปยังโหนดที่ 1 และใช้แพทเทิร์นที่สอง (A[H]) จับคู่กระแส ในที่นี้ให้คำตอบซ้ำและไม่สามารถหาโครงสร้างใหม่ได้อีก จึงย้อนกลับจากโหนดที่ 1 ไปยังโหนดที่ 0 แล้วทำการค้นหาใหม่ และเนื่องจากโหนด 0 มีโหนดที่ 1 เป็นโหนดลูกล่าสุด ดังนั้นการสร้างโหนดที่ 4 จะทำการค้นหาแพทเทิร์นและลำดับคู่กระแสถัดไปจากโหนดที่ 1 โดยโหนดที่ 1 นี้สังเกตได้ว่าลำดับคู่ของกระแสยังไม่หมด เพราะเป็นการจับคู่ระหว่าง H1 และ C1 ยังเหลือ H2 และ C1 อีก จึงใช้แพทเทิร์นที่ 2 จับคู่กระแส H2 และ C1 ในการสร้างโหนดที่ 4 ซึ่งเป็นโหนดลูกของโหนด 0 ดังรูปที่ 4.6



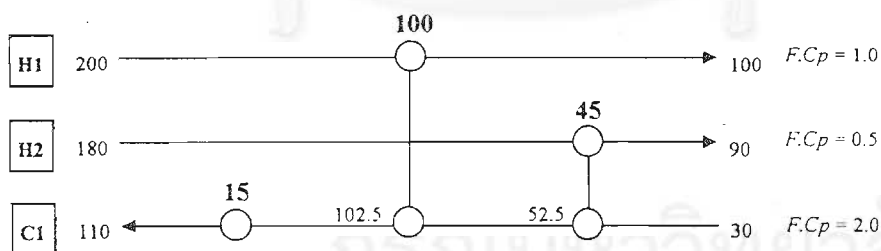
รูปที่ 4.6 โหนดที่ 4 ใช้แพทเทิร์น A[H]

รูปที่ 4.7 แสดงไดอะแกรมของโหนด จะสังเกตได้ว่าโหนดที่ 4 เป็นโหนดลูกของโหนด 0 และลำดับของกระสวนการจับคู่ (หรือแพทเทิร์น) และลำดับคู่ของกระแสจะใช้ต่อจากโหนดลูกของโหนด 0 ล่าสุดที่มีอยู่ ในที่นี้คือโหนดที่ 1

การสร้างโหนดใหม่ต่อจากโหนดที่ 4 ต่อไปนั้น จะเริ่มจากการใช้แพทเทิร์นที่ 1 (AH) แลกเปลี่ยนความร้อนระหว่างกระแส H1 กับ C1 ซึ่งจะสร้างโหนดที่ 5 และ 6 ได้คำตอบดังรูปที่ 4.8 แสดงไดอะแกรมของโหนดในรูปที่ 4.9

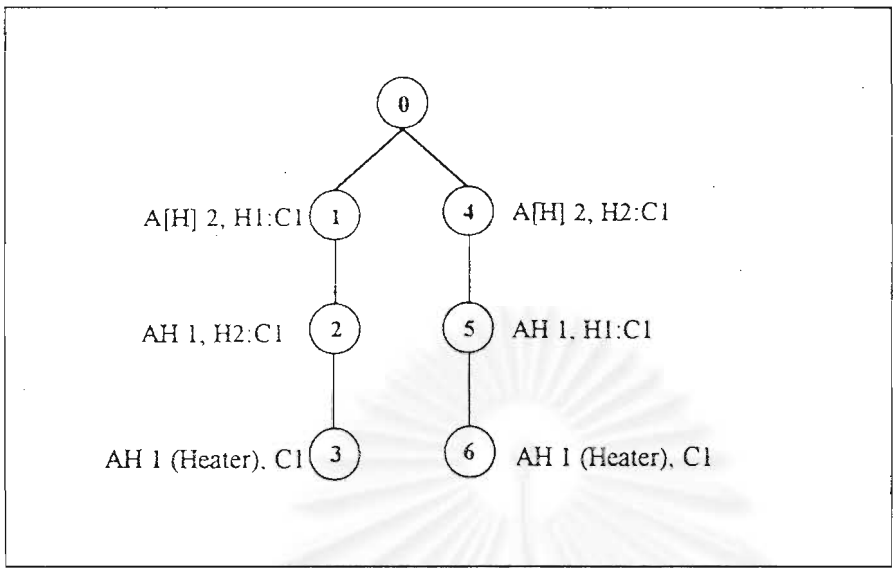


รูปที่ 4.7 แสดงโหนดที่ 4 ซึ่งเป็นโหนดลูกของโหนด 0

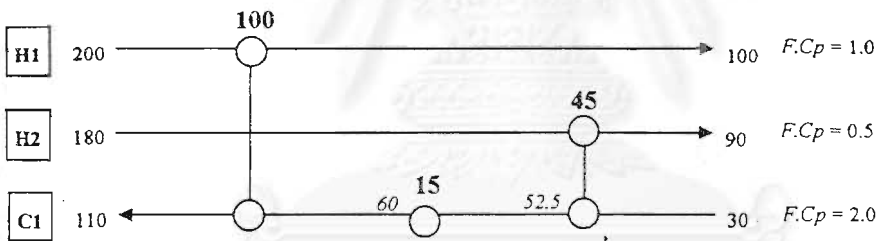


รูปที่ 4.8 แสดงขำงานคำตอบที่สอง (เทียบคำตอบแรกในรูปที่ 4.4)

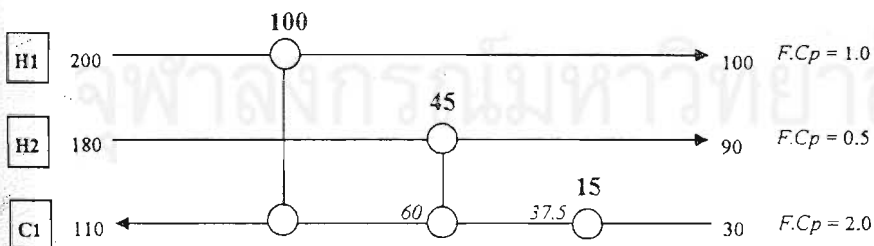
โดยวิธีเช่นนี้ จะได้คำตอบเพิ่มขึ้นมาอีกสองคำตอบ ดังรูปที่ 4.10 และ 4.11 โดยมีโหนดไดอะแกรมที่สมบูรณ์ในรูปที่ 4.12



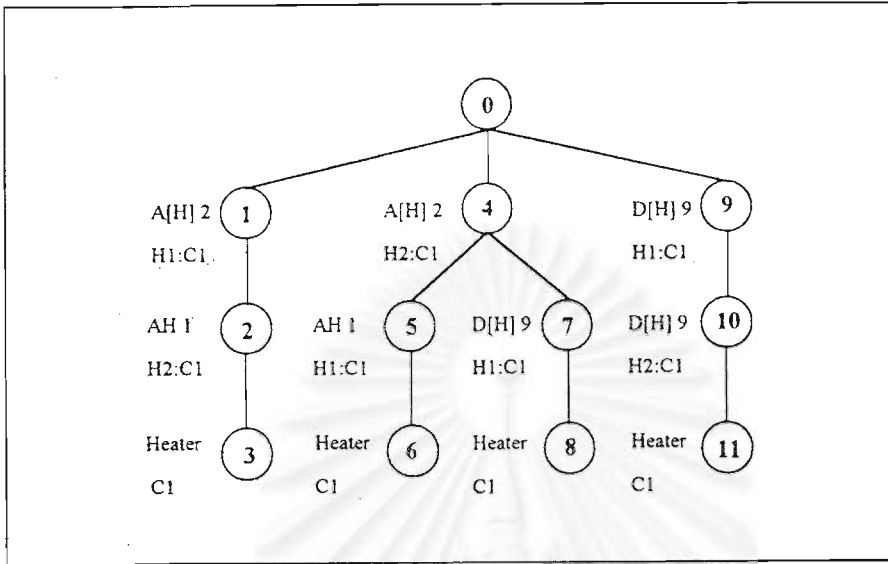
รูปที่ 4.9 ไคอะแกรมของโหนดคำตอบข้างงานคำตอบที่ 2 ที่เพิ่มขึ้น



รูปที่ 4.10 แสดงข่ายงานคำตอบที่สาม



รูปที่ 4.11 แสดงข่ายงานคำตอบที่สี่



รูปที่ 4.12 โค้ดแกรมของโหนดทั้งสี่คำตอบที่สมบูรณ์ในตัวอย่างที่ 4.1

4.3 โฟลว์ชาร์ตการทำงานของโปรแกรม

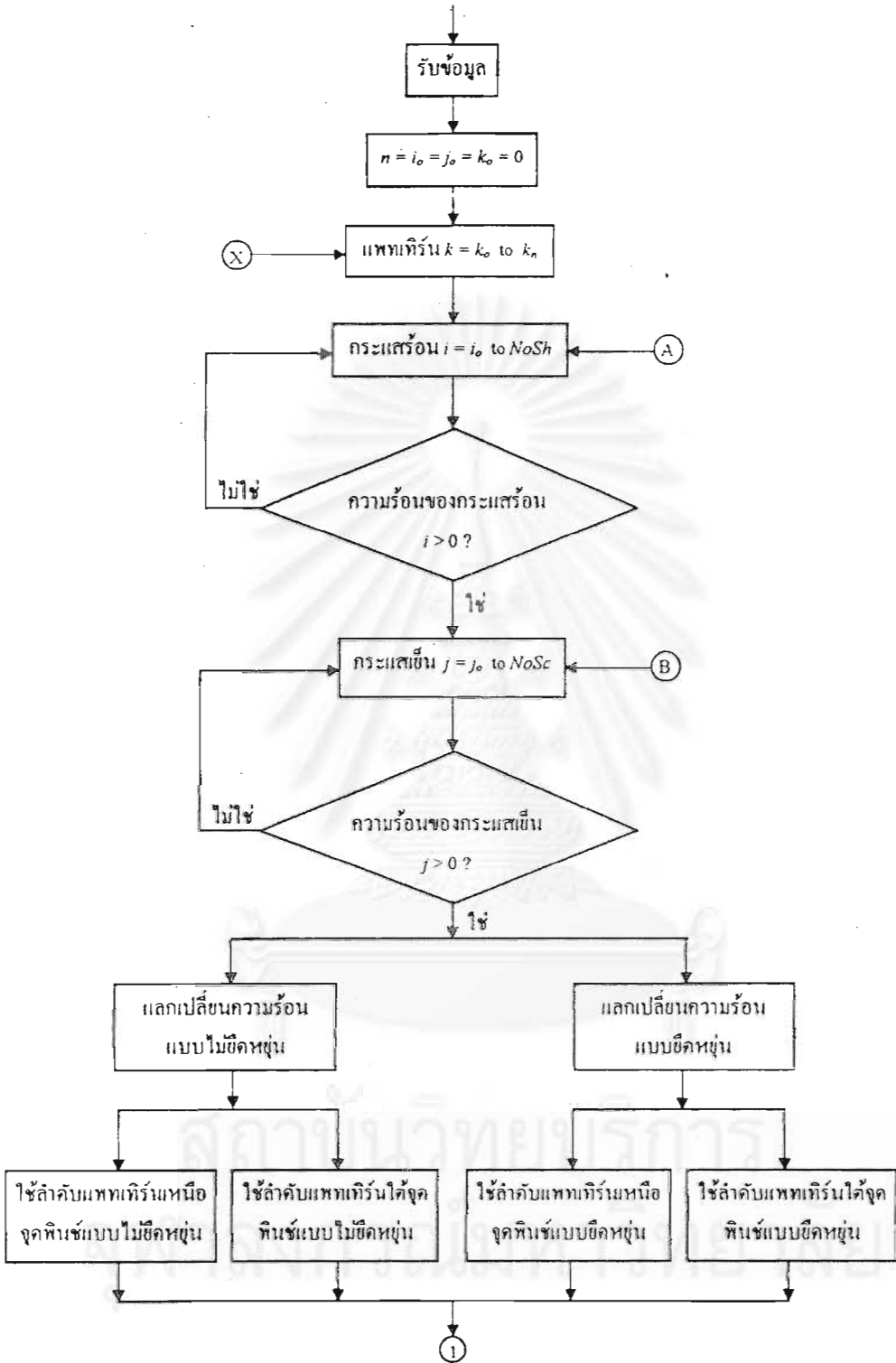
โปรแกรมส่วนการหาโครงสร้างข้างงานนี้ จะให้ความสำคัญแก่คลาสโหนดและคลาสกระแสด เป็นอย่างมาก เพราะว่าจะใช้ประโยชน์ในการคำนวณและการค้นหาคำตอบ (Search) โดยตรง และสำหรับวิธีการค้นหาคำตอบดังกล่าว จะต้องหาให้ครอบคลุมความน่าจะเป็นของโอกาสทั้งหมดที่โครงสร้างข้างงานสามารถจะมีได้ด้วย โดยไล่คู่อันดับของกระสวนการจับคู่และกระแสที่เป็นไปได้ให้ครบ ตรวจสอบคำตอบที่ซ้ำและลบโหนดที่ตายออกไป เพื่อประหยัดหน่วยความจำของคอมพิวเตอร์ หรือเมื่อมีการสร้างโหนดใหม่ไม่ว่าแบบจากบนลงล่างหรือล่างขึ้นบนก็ตาม จะต้องตั้งค่าแพทเทิร์นและลำดับของกระแสต่างๆ สำหรับการคำนวณครั้งใหม่ต่อไปให้ถูกต้อง เพื่อให้โปรแกรมทำงานได้อย่างมีประสิทธิภาพและเร็วที่สุด

จากรูป.4.13 แสดงถึงการไล่ลำดับกระสวนการจับคู่และกระแส เพื่อนำมาแลกเปลี่ยนความร้อน จากโฟลว์ชาร์ตจะเป็นว่า ได้เลือกอันดับกระสวนการจับคู่ก่อน (k) จากนั้นเลือกกระแสร้อน (i) และกระแสน้ำเย็น (j) ตามลำดับ เมื่อได้ข้อมูลของกระสวนการจับคู่และกระแสที่จะนำมาทดสอบการแลกเปลี่ยน

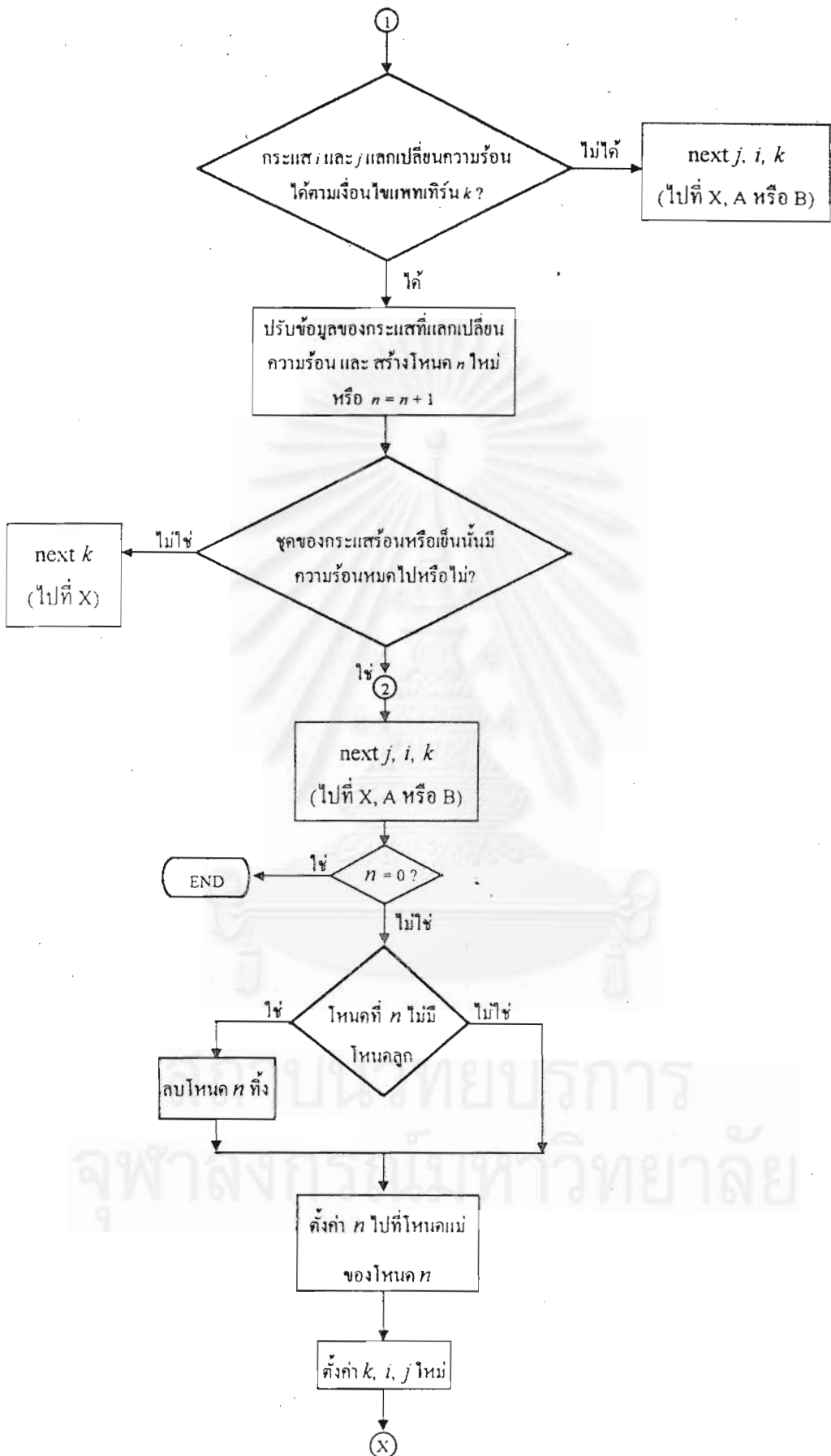
เปลี่ยนความร้อนแล้ว ก็จะตรวจสอบว่าขำยงานที่กำลังออกแบบนี้เป็นขำยงานแบบยึดหยุ่นหรือไม่ยึดหยุ่น และเป็นขำยงานเหนือจุดพินช์หรือใต้จุดพินช์ หรือไม่มีจุดพินช์ โปรแกรมก็จะทำการเลือกฟังก์ชันที่จะใช้ตามความเหมาะสม

รูปที่ 4.13 (ต่อ) แสดงการตรวจสอบการแลกเปลี่ยนความร้อนระหว่างกระแสร้อนและเย็นในกระบวนการจับคู่ที่เลือกไว้ ถ้าไม่สามารถแลกเปลี่ยนกันได้จะใช้ลำดับการแลกเปลี่ยนถัดไป เช่น ถ้ายังใช้กระแสเย็นไม่หมด ก็จะนำกระแสเย็นอื่นๆ มาจับคู่กับกระแสร้อนให้หมดก่อน เมื่อกระแสเย็นถูกเลือกมาหมดแล้วและยังไม่ได้ตามเงื่อนไขอีก ก็จะเลือกอันดับกระแสร้อนถัดไป แล้ววนรอบการเลือกกระแสเย็นใหม่ และถ้าเลือกกระแสร้อนหมดแล้วและยังไม่ได้ ก็จะใช้กระบวนการจับคู่ถัดไป แล้วจะตั้งต้นเลือกกระแสร้อนและเย็นใหม่ เป็นต้น แต่ถ้าการแลกเปลี่ยนความร้อนของคู่กระแสในกระบวนการจับคู่นั้นๆ สามารถเป็นไปตามเงื่อนไขได้แล้ว ก็จะทำการสร้างโหนดใหม่ เพื่อเก็บข้อมูลของกระแสที่ทำการแลกเปลี่ยนความร้อนล่าสุดนี้ไว้ แล้วตรวจสอบว่า ความร้อนของกระแสร้อนทั้งหมดหรือกระแสเย็นทั้งหมด ยังมีเหลือไว้แลกเปลี่ยนความร้อนได้ต่อไปอีกหรือไม่ ถ้าความร้อนของกระแสชุดใดหมดลง แสดงว่าสิ้นสุดการหาโครงสร้างขำยงานนั้นๆ แล้ว และความร้อนของกระแสชุดที่เหลือจะถูกจัดการโดยหน่วยยูทิลิตี จากนั้น โปรแกรมจะเริ่มหาคู่ลำดับใหม่สำหรับการหาโครงสร้างขำยงานอื่นๆ ที่เป็นไปได้ต่อไปเรื่อยๆ โปรแกรมในส่วนนี้จะหยุดการทำงาน เมื่อสถานะของโหนดนั้นอยู่ที่โหนด 0 และได้ใช้คู่ลำดับของกระบวนการจับคู่และคู่กระแสครบหมดแล้ว

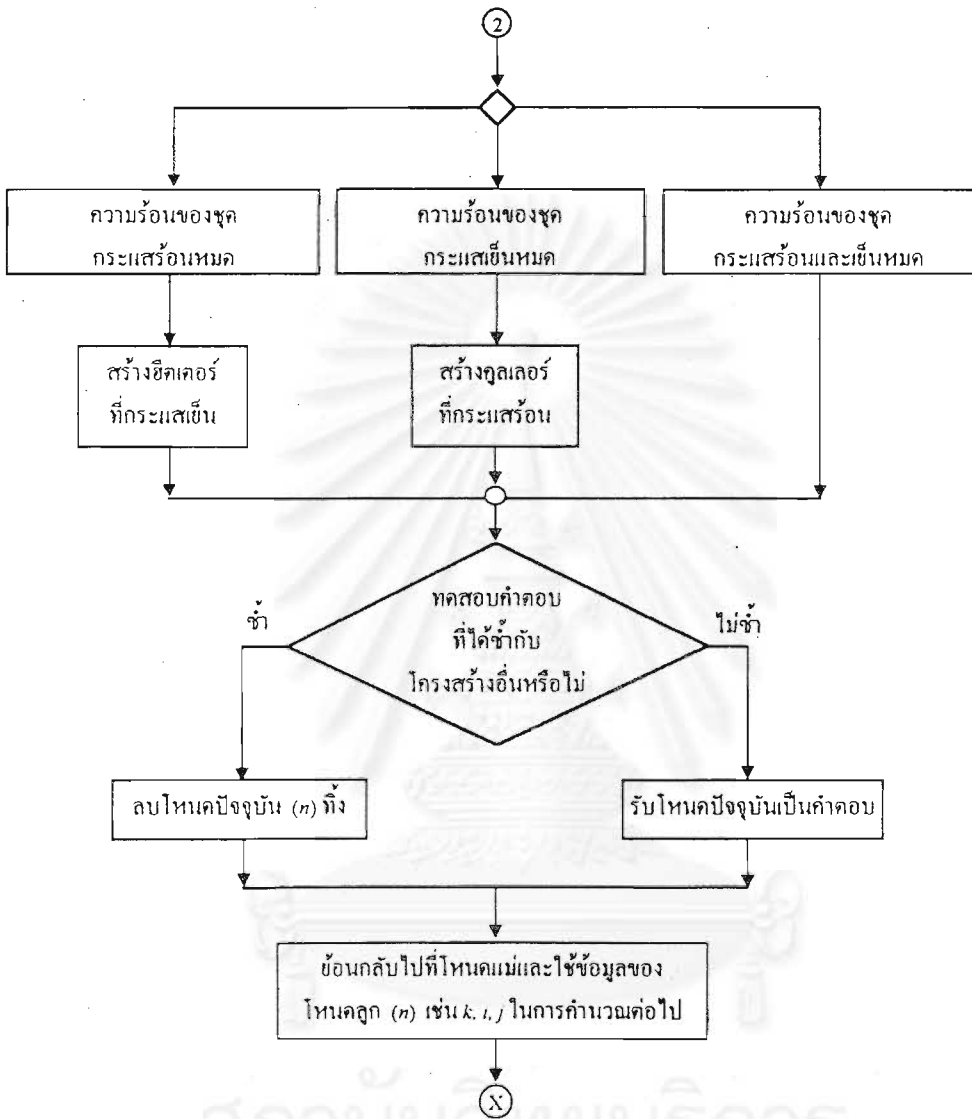
ตัวอย่างเช่น ในการแลกเปลี่ยนความร้อนเหนือจุดพินช์ สมมติว่าความร้อนของกระแสร้อนทุกกระแสถูกใช้หมดแล้ว การค้นหาลำดับการจับคู่กระแสก็จะหยุดลงชั่วคราว (ถ้ายังไม่หมด ก็จะยังทำการค้นคู่ลำดับต่อไป) จากนั้นโปรแกรมจะไปที่ส่วนของการสร้างหน่วยยูทิลิตีซึ่งในที่นี้จะเป็นการสร้างฮีตเตอร์ ณ กระแสเย็นที่มีความร้อนหลงเหลือ ดูเงื่อนไขการสร้างฮีตเตอร์และคูเลอร์ได้ในรูปที่ 4.14 ซึ่งเมื่อได้โครงสร้างของขำยงานออกมาแล้ว ยังต้องตรวจสอบว่า โครงสร้างที่หามาได้นี้ซ้ำกับโครงสร้างเดิมที่เคยมีมาก่อนหรือไม่ โดยตรวจสอบจากอุณหภูมิขาเข้าและขาออกของกระแสร้อนและกระแสเย็นในเครื่องแลกเปลี่ยนความร้อนของขำยงานทุกเครื่อง ซึ่งถ้าไปเหมือนกับของขำยงานอื่นแล้ว ก็จะทำการลบโหนดปัจจุบันทิ้ง (ไม่ได้ลบโหนดทั้งหมดซึ่งประกอบกันคำตอบที่ซ้ำ) แล้วทำการย้อนโหนดไปที่โหนดแม่ของโหนดปัจจุบัน และจะเริ่มการคำนวณใหม่ แต่ถ้าโครงสร้างขำยงานนี้ไม่ซ้ำกับโครงสร้างอื่นๆ โปรแกรมก็จะทำการรับโหนดปัจจุบันเป็นคำตอบ จากนั้นโปรแกรมจะวนหาโครงสร้างอื่นๆ ต่อไปจนครบรอบและจบการทำงาน



รูปที่ 4.13 แสดงโฟลวชาร์ตการทำงานของโปรแกรม
ในการหาโครงสร้างของข่ายงาน



รูปที่ 4.13 (ต่อ)



รูปที่ 4.14 แสดงส่วนของการสร้างหน่วยยู่ทลลลตีและการตรวจสอบค่าคอบ

บทที่ 5

การสาธิตการใช้โปรแกรม

เมื่อผู้ใช้เปิดโปรแกรมขึ้นมา (ดูรูปที่ 5.1) จะพบว่าโปรแกรมมีไอคอน (icon) สำหรับการทำงาน 11 ไอคอน เป็นแถวยาวทางด้านบนซ้ายของเมนู ไอคอนต่างๆ จะมีหน้าที่ดังนี้

ไอคอนที่ 1: เมื่อกด จะปรากฏตารางรับข้อมูลจำนวนกระแสร้อน, กระแสเย็น และค่า ΔT_{min} สำหรับค่า ΔT_{min} นี้ ผู้ออกแบบสามารถสมมติค่าก่อนได้ โดยตัวโปรแกรมได้ตั้งค่า ΔT_{min} ไว้ก่อนเท่ากับ 10°

ไอคอนที่ 2: เป็นตารางรับข้อมูลอุณหภูมิ (และความแปรปรวนของอุณหภูมิ ถ้าผู้ใช้ต้องการออกแบบขำงานแบบยืดหยุ่น) และผลคูณของอัตราการไหลกับความจุความร้อนจำเพาะของกระแสต่างๆ

ไอคอนที่ 3: เมื่อกดจะแสดงผลของตารางปัญหา

ไอคอนที่ 4: เมื่อกดจะแสดงผลของเส้นโค้งคอมโพสิต

ไอคอนที่ 5: เมื่อกดจะแสดงไคอะแกรมกริดของกระแสร้อนและกระแสเย็น ถ้าขำงานมีอุณหภูมิพินช์ ก็จะต้องแสดงไว้ในไคอะแกรมด้วย

ไอคอนที่ 6: แสดงโครงสร้างขำงานเครื่องแลกเปลี่ยนความร้อนตามที่คำนวณได้ออกมา

ไอคอนที่ 7: สลับโหมดการคำนวณสำหรับขำงานแบบยืดหยุ่นและไม่ยืดหยุ่น

ไอคอนที่ 8: รับค่าความแปรปรวนของผลคูณอัตราการไหลและความจุความร้อนจำเพาะ

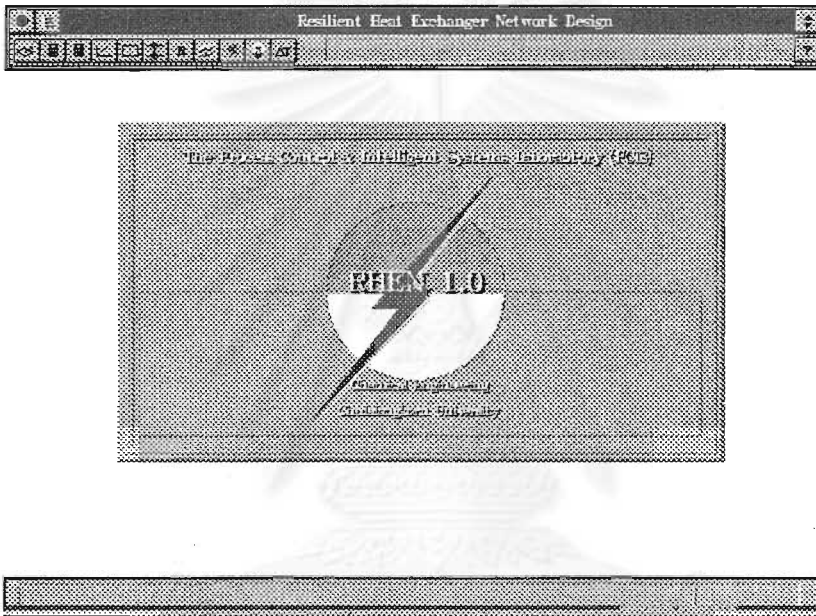
ไอคอนที่ 9: แสดงเส้นโค้งแกรนด์คอมโพสิต

ไอคอนที่ 10: สำหรับรวมขำงานเหนือจุดพินช์และใต้จุดพินช์เข้าด้วยกัน ถ้าคำตอบของขำงานมีหลายโครงสร้าง ก็จะมีไอคอนภายในสำหรับหาโครงสร้างที่ให้ค่าใช้จ่ายน้อยที่สุดด้วย

ไอคอนที่ 11: สำหรับหาค่า ΔT_{min} ที่ทำให้ได้ขำงานที่ประหยัดพลังงานและค่าใช้จ่ายมากที่สุด โดยจะต้องรับข้อมูลของกระแสเสียทิลิตี (ผู้ใช้สามารถใส่ข้อมูลของกระแสเสียทิลิตีได้ตั้งแต่ตารางรับข้อมูลในไอคอนที่ 2), อัตราดอกเบี้ยและจำนวนปีในการใช้คืนเงินลงทุน และสมการค่าใช้จ่ายในการสร้างเครื่องแลกเปลี่ยนความร้อน และค่าใช้จ่ายทางพลังงานความร้อน

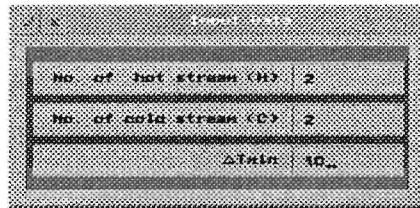
ไอคอนขวามือรูปสามเหลี่ยมสองรูปหงายและคว่ำ มีไว้สำหรับเลิกการทำงานของโปรแกรม และสำหรับไอคอนรูปเครื่องหมายคำถาม (Help) ผู้ใช้สามารถใช้เมาส์กดได้ตลอดเวลา โปรแกรมจะมีคำแนะนำเพิ่มเติมในสถานะการทำงานของผู้ใช้ขณะนั้นๆ

หลังจากนั้น ให้ผู้ใช้กดคีย์ใดๆ เพื่อให้โปรแกรมลบหน้าจอและพร้อมที่จะทำงานต่อไป โดยใช้เมาส์กดไอคอนที่ 1 (หรือใช้คีย์บอร์ด Alt-b) เพื่อที่จะรับข้อมูลของจำนวนกระแสและ ΔT_{min} ดังรูปที่ 5.2 ในที่นี้จะแสดงตัวอย่างของกระแสร้อนและเย็นอย่างละ 2 กระแส และ $\Delta T_{min} = 10^\circ$



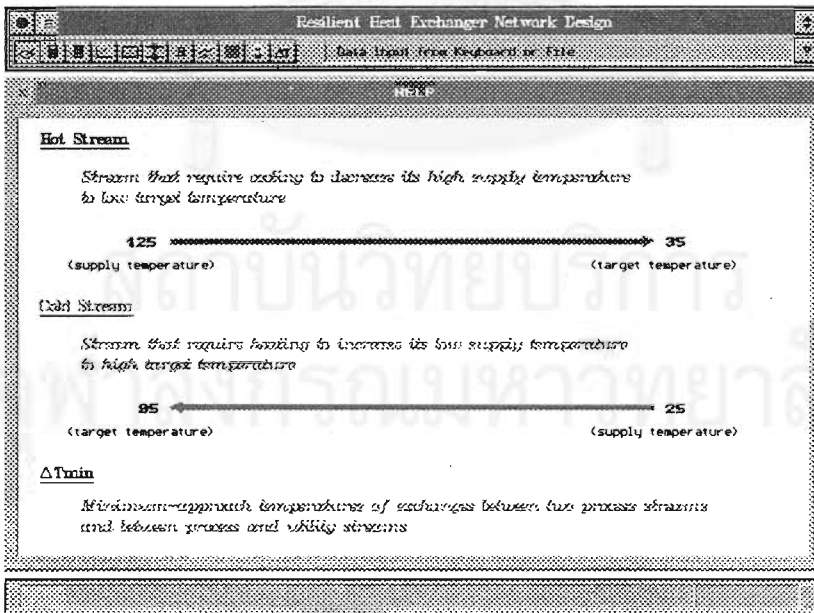
รูปที่ 5.1 แสดงโปรแกรมเมื่อเริ่มเปิดใช้งาน

ให้สังเกตว่าเมื่อเมาส์ได้ชี้ไปที่ไอคอนใดๆ ใน 11 ไอคอนนั้น โปรแกรมจะแสดงคำอธิบายสั้นๆ สำหรับไอคอนนั้นๆ ว่ามีหน้าที่สำหรับทำอะไรตลอดเวลา โดยจะแสดงไว้ที่ส่วนแสดงผลข้อความด้านบนขวา



รูปที่ 5.2 ตารางสำหรับรับข้อมูลของจำนวนกระแสและค่า ΔT_{min}

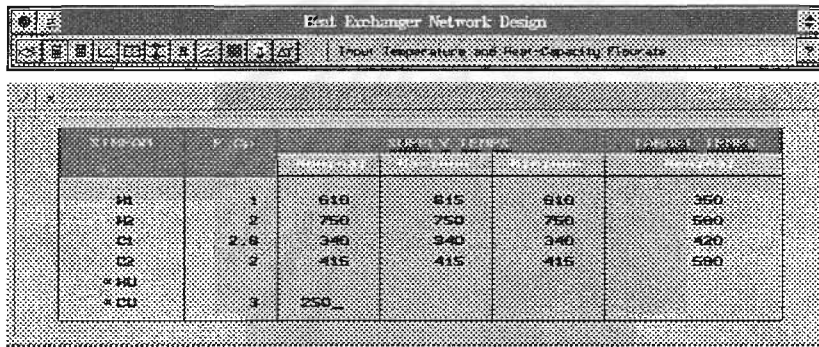
ในที่นี้ ถ้าผู้ใช้ต้องการดูความหมายของของคำว่า Hot Stream, Cold stream และ ΔT_{min} ที่แสดงไว้ในโปรแกรม ให้ใช้เมาส์กดที่ไอคอนรูปเครื่องหมายคำถาม ได้ ดังที่แสดงไว้ในรูปที่ 5.3



รูปที่ 5.3 ตัวอย่างแสดงการช่วยเหลือของโปรแกรมเมื่อใช้ไอคอนรูปเครื่องหมายคำถาม ที่ตารางรับข้อมูลของจำนวนกระแสและค่า ΔT_{min}

เมื่อใส่จำนวนข้อมูลของกระแสและ ΔT_{min} แล้ว ต่อไปให้ใช้เมาส์คลิกไอคอนที่ 2 (คีย์บอร์ด Alt-i) เพื่อใส่รายละเอียดข้อมูลของแต่ละกระแสรวมถึงกระแสขุทธิลิตีด้วย (ดูรูปที่ 5.4) โดย H แทนกระแสร้อนและ C แทนกระแสนเย็น HU แทนกระแสร้อนจากขุทธิลิตี และ CU แทนกระแสนเย็นจากขุทธิลิตี ในที่นี้ HU ไม่ได้ใส่ข้อมูลเข้าไป เพราะว่าปัญหาที่ออกแบบเป็นปัญหาทางด้านความเย็น (สร้างคูลเลอร์สำหรับกระแสร้อน)

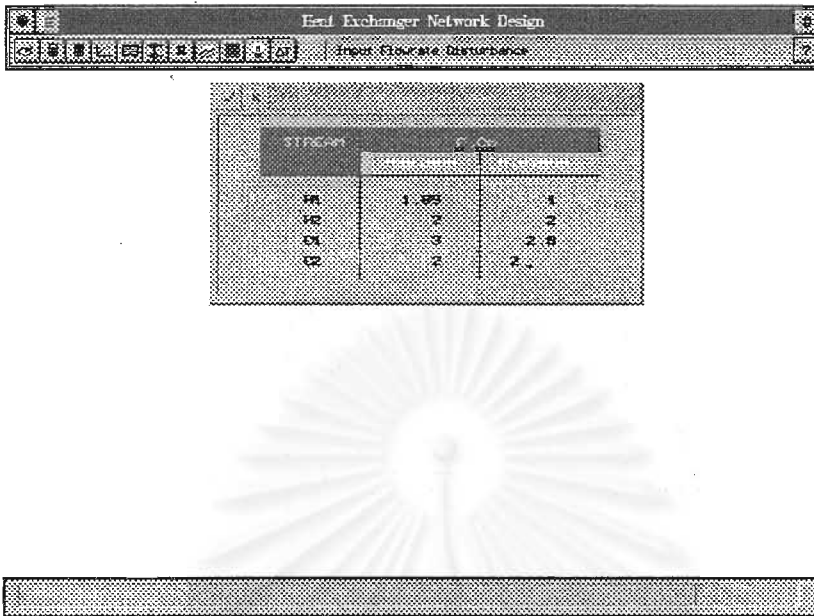
ถ้าจะใส่ข้อมูลความแปรปรวนอัตราการไหลของกระแส ให้ใช้เมาส์กดที่ไอคอนที่ 8 (คีย์บอร์ด Alt-f) ตารางรับข้อมูลจะรองรับค่าของผลคูณอัตราการไหลและความจุความร้อนจำเพาะสูงสุดและค่าต่ำสุดจากค่าที่ภาวะปกติ ดูรูปที่ 5.5



Stream	Flow	Hot Inlet	Hot Outlet	Cold Inlet	Cold Outlet
H1	1	610	815	610	350
H2	2	750	750	750	500
C1	2.6	340	340	340	420
C2	2	415	415	415	500
*HU					
*CU	3	250			

รูปที่ 5.4 ตารางสำหรับรับข้อมูลอุณหภูมิและค่าของผลคูณของอัตราการไหล กับความจุความร้อนจำเพาะของกระแส

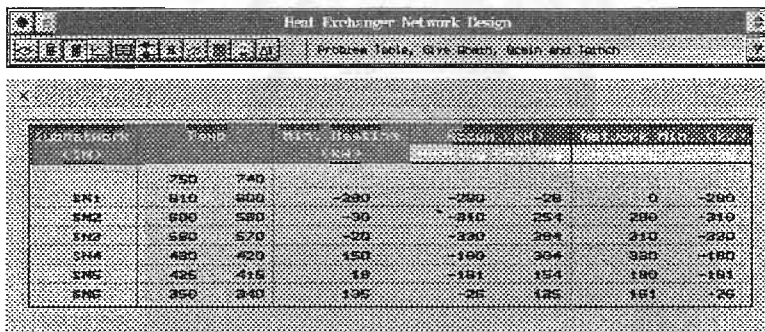
รูปที่ 5.6 แสดงตารางปัญหา โดยใช้เมาส์กดที่ไอคอนที่ 3 (คีย์บอร์ด Alt-z) ซึ่งจะให้ค่า $Q_{H,min}$, $Q_{C,min}$ หรืออุณหภูมิที่จุดพินช์ของข้างงาน แต่สำหรับปัญหานี้ไม่มีจุดพินช์ ค่า $Q_{H,min} = 0$ และ $Q_{C,min} = 26$ เป็นปัญหาทางด้านความเย็น (Cooling problem) โดยจะสร้างคูลเลอร์ที่กระแสร้อน ไม่มีฮีตเตอร์



Heat Exchanger Network Design
Input: Clean and Disturbance

STREAM	Flow	Temp
H1	1.000	1
H2	2	2
C1	3	2.5
C2	2	2

รูปที่ 5.5 รับช่วงการแปรปรวนของผลคูณอัตราการใช้กับความจุความร้อนจำเพาะ

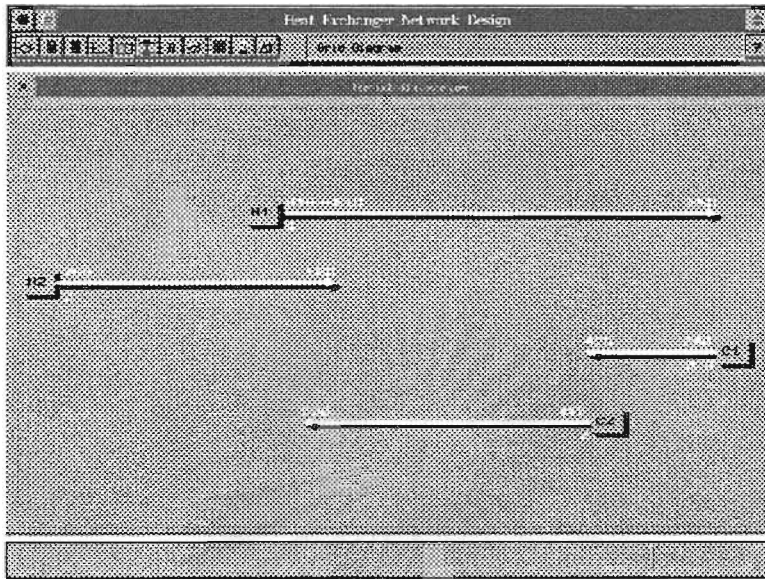


Heat Exchanger Network Design
Problem Table, Give stream, design and match

Stream	Flow	Temp	Temp	Temp	Temp	Temp	Temp
H1	750	740	-280	-280	-28	0	-280
H2	800	580	-30	-310	254	280	210
H3	580	570	-20	-200	204	210	-200
C1	480	420	150	-100	204	380	-180
C2	425	415	10	-181	154	180	-181
C3	350	340	135	-25	125	181	-25

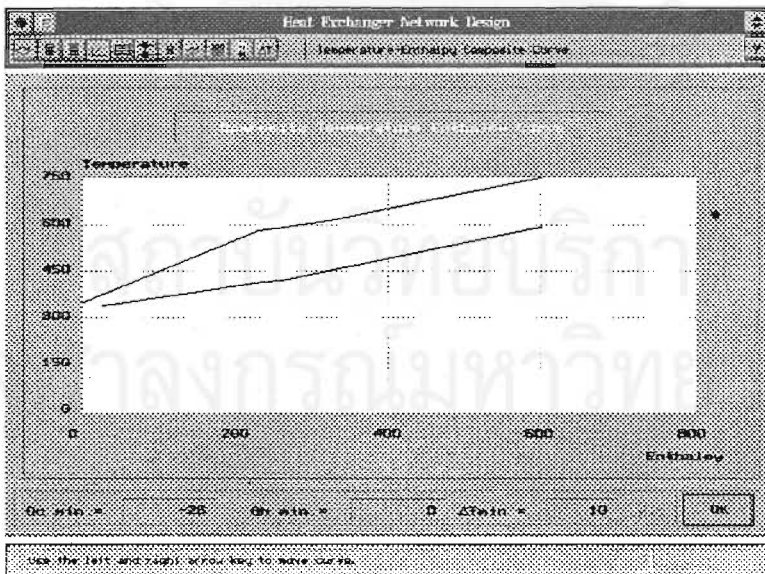
รูปที่ 5.6 แสดงผลของตารางปัญหา

รูปที่ 5.7 แสดงไดอะแกรมกริดของกระแสน้ำร้อนและกระแสน้ำเย็น โดยใช้เมาส์กดที่ไอคอนที่ 5 หรือใช้คีย์บอร์ด Alt-s ตัวเลขข้างบนของเส้นกริดกระแส คืออุณหภูมิ โดยที่ต้นกระแสเป็นอุณหภูมิขาออกหรืออุณหภูมิเป้าหมาย สำหรับตัวเลขข้างล่างต้นเส้นกริดกระแส คือค่าผลคูณของอัตราการใช้และความจุความร้อนจำเพาะ



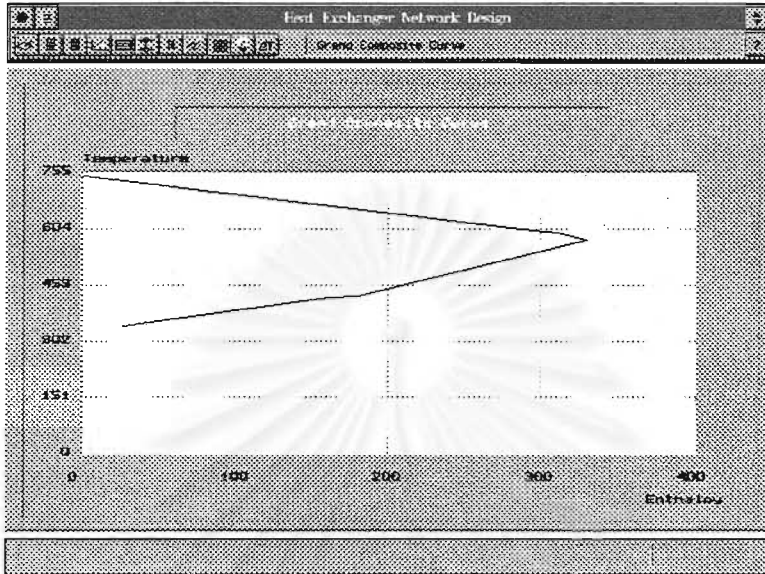
รูปที่ 5.7 ไดอะแกรมกริด

รูปที่ 5.8 เมื่อผู้ใช้กดเมาส์ที่ไอคอนที่ 4 (ใช้คีย์บอร์ด Alt-g) โปรแกรมจะแสดงเส้นโค้งคอมโพสิต ซึ่งผู้ใช้สามารถเพิ่มหรือลด ΔT_{min} ได้ ที่ปุ่มลูกศรซ้ายขวาบนตัวกราฟ (ข้างซ้ายของปุ่ม OK) หรือใช้คีย์บอร์ด PageUp และ PageDown ก็ได้



รูปที่ 5.8 เส้นโค้งคอมโพสิตที่ ΔT_{min} เท่ากับ 10° .

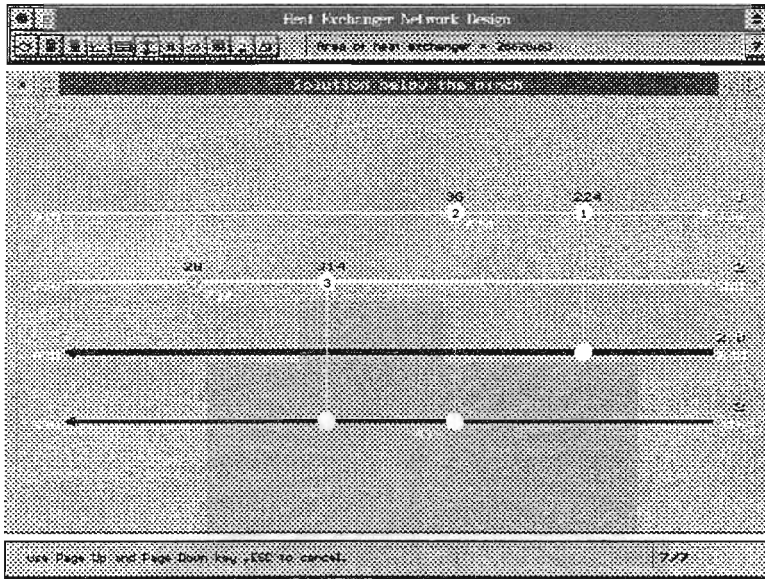
รูปที่ 5.9 แสดงเส้นโค้งแกรนด์คอมโพสิต โดยใช้เมาส์กดไอคอนที่ 9 หรือคีย์บอร์ด Alt-d



รูปที่ 5.9 เส้นโค้งแกรนด์คอมโพสิต

รูปที่ 5.10 โดยใช้เมาส์กดที่ไอคอนที่ 6 (ใช้คีย์บอร์ด Alt-m) แสดงโครงสร้างของข่ายงาน ในที่นี้มีทั้งหมด 7 โครงสร้าง ผู้ใช้สามารถกดเมาส์ที่ไอคอนรูปสามเหลี่ยมหันทางซ้าย และทางขวา หรือใช้คีย์บอร์ด PageUp และ PageDown เพื่อดูโครงสร้างทั้งหมดก็ได้ แต่ละโครงสร้างที่ส่วนแสดงข้อความข้างบนจะแสดงพื้นที่แลกเปลี่ยนความร้อนทั้งหมดของข่ายงานด้วย จากรูปได้แสดงเฉพาะโครงสร้างที่ 7

สำหรับวงกลมสองวงซึ่งเชื่อมระหว่างกระแสร้อนและกระแสนั้น หมายถึงเครื่องแลกเปลี่ยนความร้อนหนึ่งเครื่อง และวงกลมวงเดียวที่กระแสร้อน คือคูลเลอร์ (ถ้าเป็นวงกลมวงเดียวแต่อยู่ที่กระแสนั้น ก็จะหมายถึงฮีตเตอร์) ตัวเลขข้างบนของวงกลมหมายถึง ปริมาณความร้อนที่เครื่องแลกเปลี่ยนความร้อนใช้ ตัวเลขข้างล่างเส้นกริดกระแสตกลงมาจากวงกลมหมายถึง อุณหภูมิขาเข้าหรือขาออกจากเครื่องแลกเปลี่ยนความร้อนนั้นๆ สำหรับปัญหานี้มีเครื่องแลกเปลี่ยนความร้อนรวมทั้งหมด (รวมคูลเลอร์ด้วย) เท่ากับ 4 เครื่อง



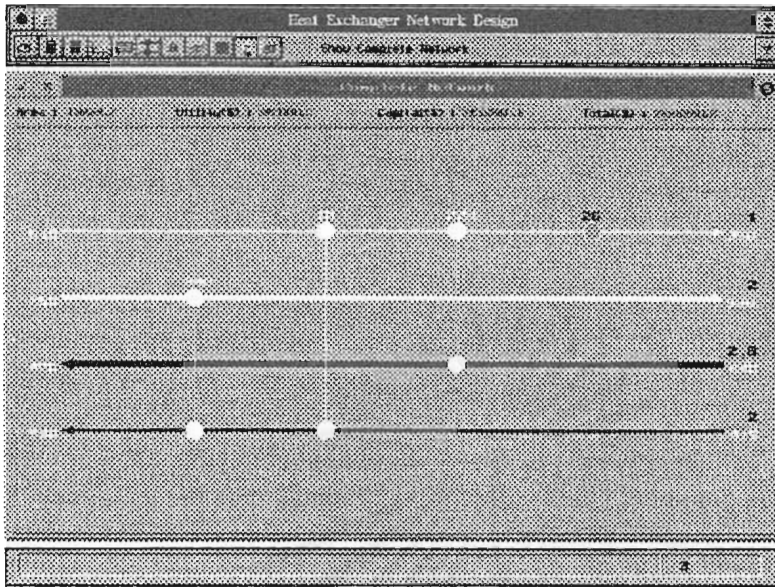
รูปที่ 5.10 แสดงข่ายงานที่หาได้ 7 โครงสร้างในที่นี้แสดงโครงสร้างที่ 7

ในที่นี้แม้ว่าข่ายงานไม่มีอุณหภูมิพินช์ก็ตาม ก็ยังสามารถใช้เมตริกที่ไอคอนที่ 10 และใช้ไอคอนภายในหาโครงสร้างที่ประหยัดค่าใช้จ่ายมากที่สุดได้อีกด้วย จาก 7 โครงสร้างที่ได้นั้นพบว่าโครงสร้างที่ 3 ประหยัดที่สุด (รูปที่ 5.11)

สำหรับไอคอนที่ 11 ใช้หา ΔT_{\min} ที่ทำให้ข่ายงานประหยัดค่าใช้จ่ายที่สุด (Optimum ΔT_{\min}) โดยผู้ใช้จะใส่ข้อมูลสำหรับการคำนวณดังแสดงในรูปที่ 5.12 นอกจากนี้ยังสามารถเลือกให้ข่ายงานมีค่าสัมประสิทธิ์การถ่ายเทความร้อนรวม (U) คงที่หรือไม่ก็ได้ โดยถ้าค่าดังกล่าวไม่คงที่ ผู้ใช้ต้องใส่ค่า Film transfer coefficient (h) ของแต่ละกระแสน้ำด้วย จากรูปที่ 5.13 แสดงการใส่ค่า U เมื่อผู้ใช้เลือกให้ค่า U คงที่ ในที่นี้ให้ U มีค่าเท่ากับ 0.00004

จากรูปที่ 5.14 เป็นกราฟสำหรับหาค่า optimum ΔT_{\min} แกนตั้งเป็นค่าใช้จ่ายรวมของข่ายงานซึ่งประกอบไปด้วยค่าใช้จ่ายสำหรับการสร้างข่ายงาน และค่าจ่ายด้านพลังงานสำหรับแกนนอนคือค่า ΔT_{\min} ในที่นี้ได้แสดงในช่วง 0 ถึง 20° ถ้าผู้ใช้ต้องการเปลี่ยนช่วงของ ΔT_{\min} ให้กว้างหรือแคบกว่านี้ก็สามารถเปลี่ยนได้โดยใส่ในช่องรับค่า ΔT_{\min} range

จากรูป สังเกตได้ว่าเส้นกราฟให้จุดต่ำสุด หรือค่าใช้จ่ายต่ำที่สุดที่ ΔT_{\min} เท่ากับ 11°



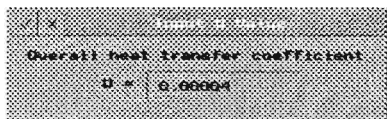
รูปที่ 5.11 โครงสร้างข่ายงานที่ประหยัดค่าใช้จ่ายมากที่สุด

The screenshot shows the 'Data for Optimization' dialog box in the 'Heat Exchanger Network Design' software. The dialog box contains the following fields and labels:

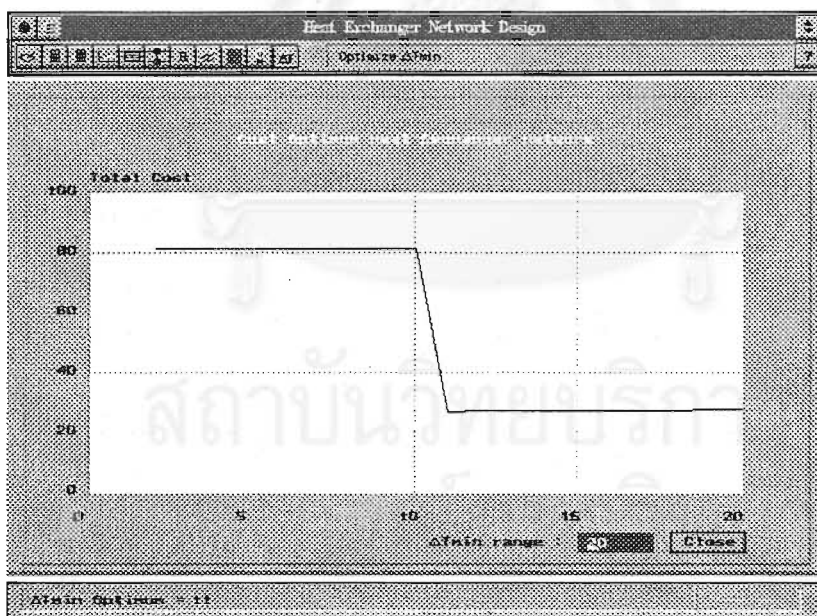
- Hot Utility Stream: Supply Temp. F, Cp
- Cold Utility Stream: Supply Temp. F, Cp
- Fractional interest rate per year: number of years
- Decaying Cost (%/year, G): Heating, Cooling
- Heat Exchanger Cost (\$/ft²): A Area

At the bottom of the dialog box, there are three buttons: 'All vars constant', 'Ok', and 'Cancel'.

รูปที่ 5.12 รับข้อมูลเพื่อที่จะหาค่า optimum ΔT_{min}

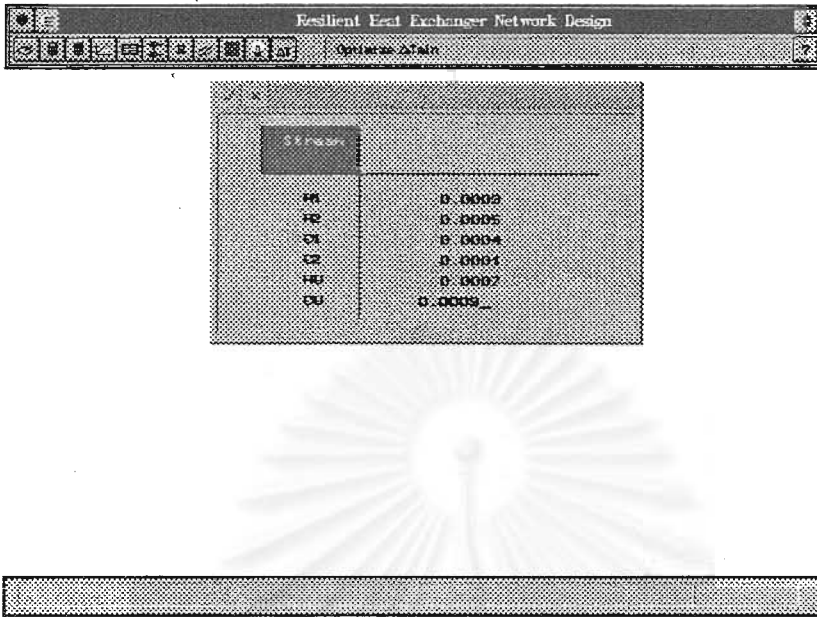


รูปที่ 5.13 เมื่อให้ U คงที่ โปรแกรมจะให้ผู้ผู้ใช้ใส่ค่า ในที่นี้ให้ U เท่ากับ 0.00004



รูปที่ 5.14 แสดง ΔT_{min} ที่เหมาะสมกับข่างานซึ่งเท่ากับ 11° (เมื่อให้ U คงที่)

ถ้าผู้ใช้ให้ค่า U ไม่คงที่ โปรแกรมจะแสดงตารางรับค่าสัมประสิทธิ์การถ่ายเทความร้อนผ่านฟิล์ม (h) ของแต่ละกระแสน้ำ ดังรูปที่ 5.15



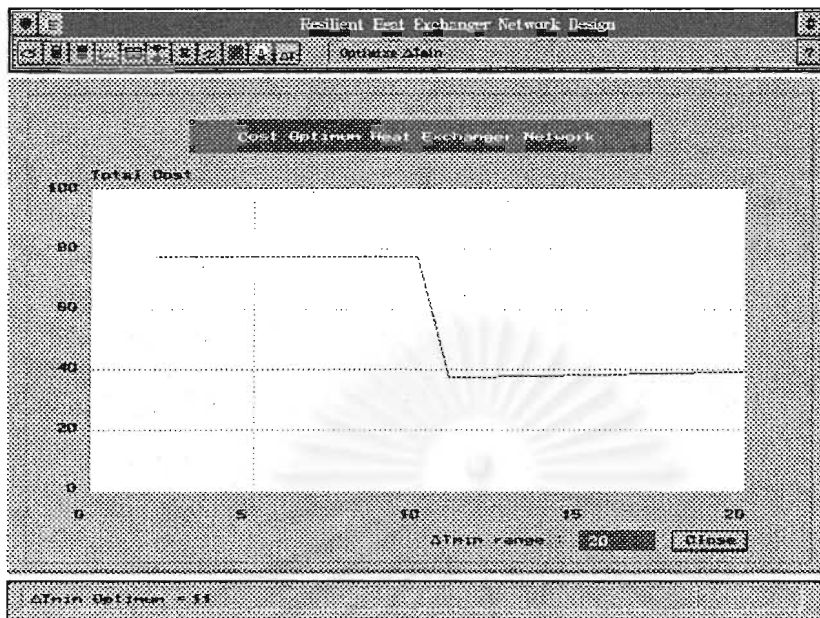
The screenshot shows a software window titled "Resilient East-Exchanger Network Design" with a sub-window "Optimize ATMin". Inside, a table lists values for 'h' corresponding to different 'U' values. The table is as follows:

U	h
U1	0.0002
U2	0.0005
U3	0.0004
U4	0.0004
U5	0.0007
U6	0.0009

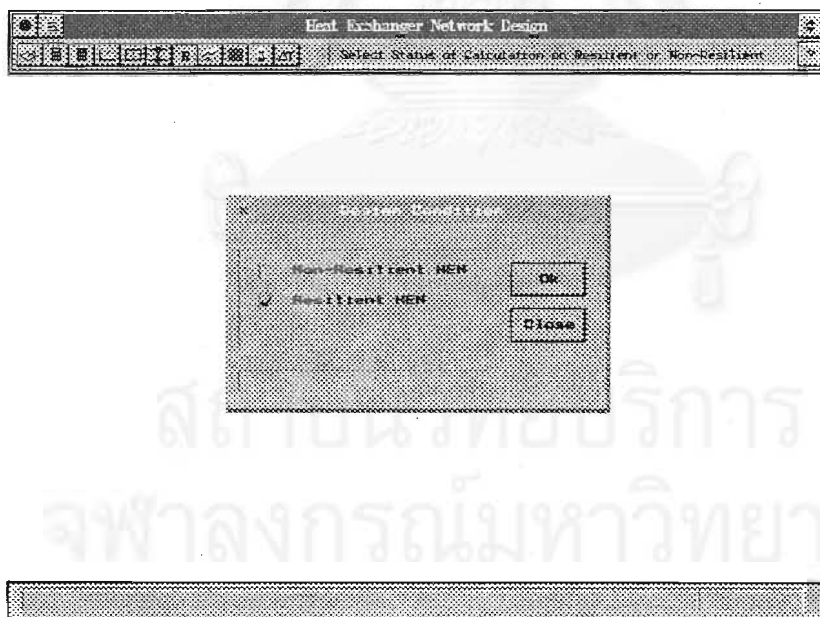
รูปที่ 5.15 ตารางรับค่า h เมื่อ U ในข่ายงานไม่คงที่

และผลการคำนวณเมื่อใช้ค่า h แทนค่า U ซึ่งไม่คงที่ จะให้ค่าออกปดิม ΔT_{min} เท่ากับ 11° เช่นกัน (รูปที่ 5.16)

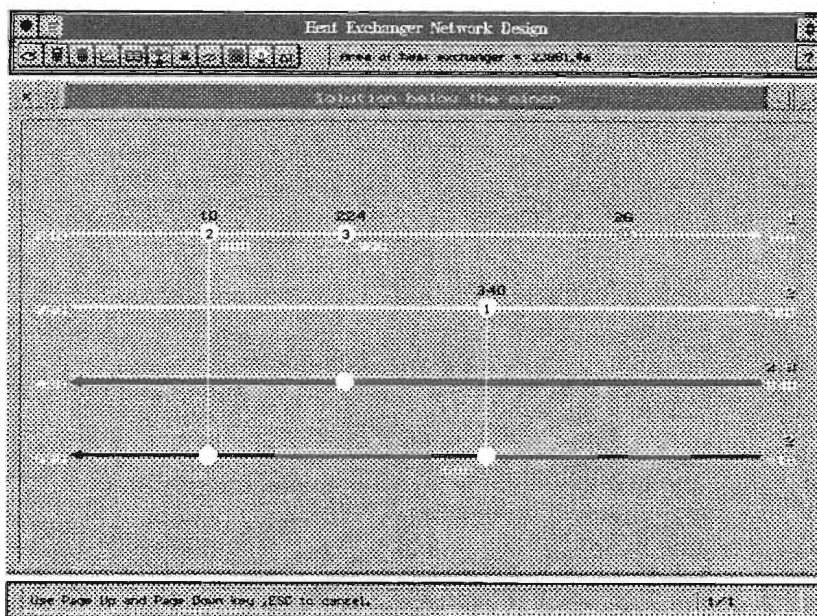
กรณีที่ผู้ใช้ต้องการหาโครงสร้างที่ยืดหยุ่นในช่วงของความแปรปรวนที่ได้กำหนดไว้ ก่อนอื่นต้องเปลี่ยนโหมดการคำนวณก่อน โดยใช้เมาส์คลิกที่ไอคอนที่ 7 (คีย์บอร์ด Alt-r) จากรูปที่ 5.17 จะมีตัวเลือกให้ผู้ใช้สองข้อ ก็คือ Non-resilient (หาโครงสร้างข่ายงานแบบไม่ยืดหยุ่น ไม่คำนึงถึงความแปรปรวน) และ Resilient (หาโครงสร้างข่ายงานแบบยืดหยุ่น โดยคำนึงถึงความแปรปรวน) ส่วนของช่องแสดงข้อความด้านล่างจะแสดงอุณหภูมิพินช์ (ถ้ามี) ในที่นี้เลือก Resilient จากนั้นกลับไปหาโครงสร้างข่ายงานอีกครั้ง จะได้คำตอบของข่ายงานแบบยืดหยุ่นออกมาเพียงโครงสร้างเดียว ดังรูปที่ 5.18



รูปที่ 5.16 แสดง ΔT_{min} ที่เหมาะสมกับข่ายงานซึ่งเท่ากับ 11°
เมื่อให้ U ไม่คงที่และคำนวณจากค่า h แทน



รูปที่ 5.17 เลือกโหมดโครงสร้างข่ายงานแบบยืดหยุ่น



รูปที่ 5.18 แสดงข่ายงานแบบขีดหยุ่นที่โปรแกรมหาได้

คำสั่งอื่นๆ

การใช้งานโปรแกรมยังมีคำสั่งที่ผู้ใช้สามารถสั่งผ่านคีย์บอร์ดได้อีก ดังนี้

Alt-h : ขอความช่วยเหลือจากฐานข้อมูลในโปรแกรม

Alt-p : เปลี่ยนแปลงจำนวนการจองพื้นที่ของหน่วยความจำของโน้ต

Alt-u : รับค่าสัมประสิทธิ์การถ่ายเทความร้อนรวม (U)

Alt-l : คู่มือของโปรแกรม

Alt-c : เกี่ยวกับผู้พัฒนาโปรแกรม

Alt-n : แสดงเมนูหลักของโปรแกรม

Alt-t : รับค่า ΔT_{min}

จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 6

สรุปและวิจารณ์ผลงานวิจัย

6.1 สรุปผลการวิจัย

โปรแกรมการออกแบบข่ายงานเครื่องแลกเปลี่ยนความร้อนแบบยัดหุ่่นนี้ (RHEN) ใช้วิธีการตารางปัญหาสำหรับการหาค่าพลังงานที่ใช้้น้อยที่สุดจากหน่วยยูลิติติ ($Q_{H,min}$ และ $Q_{C,min}$) และอุณหภูมิพินซ์ สำหรับการหาโครงสร้างข่ายงานใช้้กระสวนการจับคู่ (Match patterns) กับวิธีส่งผ่านความแปรปรวน (Disturbance propagation method) ของ Wongsri (1990) โดยวิธีนี้ข่ายงานจะมีจำนวนเครื่องแลกเปลี่ยนความร้อนและใช้พลังงานจากหน่วยยูลิติติได้น้อยที่สุด ถ้าข่ายงานมีอุณหภูมิพินซ์ จะแบ่งการออกแบบเป็น การออกแบบข่ายงานเหนือจุดพินซ์ และใต้จุดพินซ์ ข่ายงานที่สมบูรณ์จะได้มาจากการรวมข่ายงานย่อยทั้งสองเข้าด้วยกัน

สำหรับการวิเคราะห์ข่ายงานโปรแกรมจะมีตารางปัญหา โคอะแกรมกริด เส้นโค้งคอมโพสิต เส้นโค้งแกรนด์คอมโพสิต และเส้นโค้งคอมโพสิตสมดุค โปรแกรมยังสามารถหา ΔT_{min} ที่อปกติ้ม (ΔT_{min} ที่ทำให้ข่ายงานเครื่องแลกเปลี่ยนความร้อนประหัยพลังงาน และค่าใช้จ่ายในการสร้างมากที่สุด) โดยใช้วิธีการแลกเปลี่ยนความร้อนแนวตั้งของเส้นโค้งคอมโพสิตสมดุค (Linnhoff และ Ahmad, 1990) นอกจากนี้ถ้าคำตอบของการออกแบบข่ายงานเครื่องแลกเปลี่ยนความร้อนมีหลายโครงสร้างด้วยกัน โปรแกรมยังสามารถเลือกโครงสร้างที่ประหัยค่าใช้จ่ายที่สุดออกมาให้ได้อีกด้วย

การออกแบบโปรแกรมจะใช้ภาษาเชิงวัตถุ C++ ซึ่งในที่นี้จะใช้ของ Borland C++ 4.0 โดยเลือกให้โปรแกรมที่ได้ทำงานบนระบบปฏิบัติการ DOS สำหรับเครื่องคอมพิวเตอร์ที่ใช้ควรเป็นเครื่องมีไมโครโปรเซสเซอร์ 80386 ขึ้นไป หน่วยความจำหลัก (RAM) ควรมากกว่า 4 เมกะไบต์ โดยขนาดของโปรแกรมมีเนื้อที่หน่วยความจำรวม 0.75 เมกะไบต์

การออกแบบคลาสของโปรแกรมจะแบ่งได้เป็นสองคลาสใหญ่ๆ คือคลาสการออกแบบเริ่มต้น มีหน้าที่สำหรับจัดเตรียมข้อมูลและแสดงผลการวิเคราะห์ข่ายงานในรูปแบบของ ตาราง โคอะแกรมกริด และกราฟ คลาสที่สอง คือ คลาสการหาโครงสร้างข่ายงาน ซึ่งมีหน้าที่สำหรับหาข่ายงานเครื่องแลกเปลี่ยนความร้อนและแสดงผล นอกจากนี้ยังมีคลาสที่สำคัญอีกสองคลาสคือ คลาสกระแส และคลาสโหนด

สำหรับคลาสกระแสมีไว้สำหรับสร้างวัตถุกระแสร้อน และวัตถุกระแสเย็นขึ้นเพื่อเก็บข้อมูลต่างๆของกระแส เช่น อุณหภูมิ อัตราการไหล เป็นต้น คลาสโหนด เป็นคลาสที่เก็บข้อมูลต่างๆของกระแสต่างๆ ก่อนหรือหลังการใช้้กระสวนการจับคู่สร้างเครื่องแลกเปลี่ยนความร้อน คำนวณพื้นที่แลกเปลี่ยน

เปลี่ยนความร้อนของเครื่องแลกเปลี่ยนความร้อน เก็บค่าอุณหภูมิขาเข้า, ขาออก และความร้อนที่ใช้แลกเปลี่ยน อาจถือว่าคลาสโหนดเป็นคลาสของเครื่องแลกเปลี่ยนความร้อนก็ได้ คลาสโหนดมีความสำคัญสำหรับการค้นหา (Search) โครงสร้างข่ายงานต่างๆออกมาให้มากที่สุดเท่าที่จะเป็นไปได้

6.2 วิจารณ์ผลงานวิจัย

โปรแกรม RHEN เป็นโปรแกรมที่ให้ความสะดวกแก่วิศวกรสำหรับออกแบบข่ายงานเครื่องแลกเปลี่ยนความร้อน เนื่องจากมีเครื่องมือในการวิเคราะห์ข่ายงาน คือ กราฟ กริดไดอะแกรม ตาราง และส่วนช่วยเหลือ (Help) ให้แก่ผู้ใช้ โปรแกรมสามารถหาข่ายงานแบบยึดหยุ่นทั้งหมดออกมาได้อย่างรวดเร็ว

อย่างไรก็ตาม โปรแกรมจะใช้สำหรับปัญหาข่ายงานที่มีความแปรปรวนเล็กน้อยซึ่งให้การเปลี่ยนแปลงอุณหภูมิพินช์มีค่าต่อเนื่อง (เรียกว่าปัญหาแบบคลาส I) ดังนั้น การพัฒนาโปรแกรมเพื่อใช้งานกับปัญหาต่างๆให้ได้กว้างขึ้น จึงควรปรับปรุงโปรแกรมให้สามารถแก้ปัญหาในกรณีที่กระแสมีความแปรปรวนมากๆได้ (ปัญหาแบบคลาส II หรือในลักษณะที่อุณหภูมิพินช์เปลี่ยนแปลงแบบไม่ต่อเนื่อง) และควรเพิ่มเงื่อนไขการคำนวณในกรณี สารที่นำมาแลกเปลี่ยนความร้อนกันนั้นเกิดการเปลี่ยนสถานะด้วย

นอกจากนี้โปรแกรมใช้ทำงานภายใต้ระบบปฏิบัติการ DOS ซึ่งมีข้อจำกัดสำหรับการอ้างถึงหน่วยความจำคอมพิวเตอร์ จึงทำให้โปรแกรมใช้ออกแบบข่ายงานสำหรับกระแสร้อนและเย็นทั้งหมดประมาณไม่เกิน 15 กระแสเท่านั้น การพัฒนาต่อไปจึงควรย้ายระบบการทำงานของโปรแกรมให้ไปทำงานบน WINDOWS ซึ่งในอนาคตจะใช้งานได้กว้างกว่า และมีหน่วยความจำให้โปรแกรมได้อ้างถึงได้มากกว่าด้วย

รายการอ้างอิง

- Arunwatanamongkol P., "A Computer Program for Heat Exchanger Network Design Using Match Patterns Approach", Master's Thesis, Chulalongkorn University, (1992).
- Calandranis, J. and Stephanopoulos, G., "Structural Operability Analysis of Heat Exchanger Network", Chem. Eng Res. Des., 64, 347 (1986).
- Cerda, J., A.W. Westerberg, D. Mason and B. Linnhoff, "Minimum Utility Usage in Heat Exchanger Network Synthesis - A Transportation Problem", Chem.Eng. Sci., 38(3), 373 (1983a).
- Cerda, J. and A.W. Westerberg, "Synthesizing Heat Exchanger Networks having Restricted Stream/Stream Matches using Transportation Problem Formulations", Chem. Eng. Sci., 38(10), 1723 (1983b).
- Cerda, J., M.R. Galli, N. Camussi and M.A. Isla, "Synthesis of Flexible Heat Exchanger Networks-I. Convex Networks", Comp. Chem. Engng., 14(2), 197-211 (1990a).
- Cerda, J. and M.R. Galli, "Synthesis of Flexible Heat Exchanger Networks-II. Nonconvex Networks with Large Temperature Variations", Comp. Chem. Eng., 14(2), 213-225 (1990b).
- Chen, B., J. Shen, Q. Sun, and S.Hu, "Development of Expert System for Synthesis of Heat Exchanger Networks", Comput. Chem. Eng., 13(11/12), 1221 (1989).
- Floudas, C.A., A.R. Ciric and I.E. Grossmann, "Automatic Synthesis of Optimum Heat Exchanger Network Configurations", AIChE J., 32(2), 276(1986).
- Floudas, C.A., and I.E. Grossmann, "Synthesis of Flexible Heat Exchanger Network for Multiperiod Operation", Comput. Chem. Eng., 10(2), 153 (1986).
- Floudas, C.A., and I.E. Grossmann, "Synthesis of Flexible Heat Exchanger Network with Uncertain Flowrates and Temperatures", Comput. Chem. Eng., 11(4), 319-336 (1987).
- Hohmann, E. C. Optimum Networks for Heat Exchanger, Ph.D. Dissertation, University of South California, (1971).
- Jezowski, J. and E. Hahne, "Heat Exchanger Network Synthesis by a Depth-First Search Method-A Case Study", Chem. Eng. Sci., 41(12), 2989 (1986).

- Linnhoff, B., et. al., User Guide on Process Integration for the Efficient Use of Energy, Inst. of Chem. Engrs. UK (1982).
- Linnhoff, B., and Ahmad, S., Cost Optimum Heat Exchanger Networks-1. Minimum Energy and Capital Using Simple Models for Capital Cost, *Comput. Chem.Engng*, 14(7), 729-750 (1990).
- Linnhoff, B., and J.R.Flower, "Synthesis of Heat Exchanger Networks-I. Systematic Generation of Energy Optimal Networks", *AIChE J.*, 24(4), 633-642 (1978a).
- Linnhoff, B., and J.R.Flower, "Synthesis of Heat Exchanger Networks-II. Evolutionary Generation of Networks with Various Criteria of Optimality", *AIChE J.*, 24(4), 642-654 (1978b).
- Linnhoff, B., and Hindmarsh, E., "The Pinch Design Method for Heat Exchanger Networks", *Chem. Eng. Sci.*, 38(5), 745-763(1983).
- Linnhoff, B., and E. Kotjabasakis, "Design of Operable Heat Exchanger Networks", First U.K. National Heat Transfer Conference, I. Chem. E. Symp. Ser. No.86, Vol1, 599 (1984).
- Linnhoff, B., and E. Kotjabasakis, "Downstream Paths for Operable Process Design", *CEP*, 82 (5),23(1986).
- Marselle, D.F., M.Morai and D.F. Rudd, "Design of Resilient Processing Plants-II: Design and Control of Energy and Management System", *Chem. Eng. Sci.*, 37(2), 259(1982).
- Masso, A.H. and D.F. Rudd, "The Synthesis of System Designs-II. Heuristic Structuring", *AIChE J.*, 15, 10-18 (1969).
- Mehta. C.D. and L.T.Fan, "Heat Exchanger Network Synthesis: A Knowledge Engineering Approach", Paper presented at AIChE annual meeting, New York City, November (1987).
- Papoulias, S.A. and I.E. Grossmann "A Structural Optimization Approach in Process Synthesis-II. Utility Systems", *Comput. Chem. Engng.*, 7, 707-722 (1983).
- Pehler, F.A. and Y.A. Liu, "Efficiency and Costing: Second Law Analysis of Processes", ACS Symposium Series No.135, ACS, Washington D.C. (1983).
- Pho, T.K. and L. Lapidus, "Topics in Computer Aides Design-II. Synthesis of Optimal Heat Exchanger Networks by Tree Search Algorithms", *AIChE J.*, 19(6), 1182(1973).
- Ponton, J.W. and R.A.B. Donaldson, "A Fast Method for Synthesis of Optimal Heat Exchanger Networks", *Chem. Eng. Sci.*, 29, 2375-2377 (1974).
- Rathore, R.N.S. and G.J. Powers, "A Forward Branching Scheme for the Synthesis of Energy Recovery System", *Ind. Eng. Chem. Proc. Des. Dev.*, 14, 175(1975).

- Saboo, A.K. and M. Morari, "Design of Resilient Processing Plants-IV: Some New Results on Heat Exchanger Network Synthesis", *Chem. Eng. Sci.*, 39(3), 579 (1984).
- Saboo, A.K. and M. Morari and D.C. Woodcock, "Design of Resilient Plants-VIII: A Resilience Index for Heat Exchanger Networks", *Chem. Eng. Sci.*, 40, 1553 (1985).
- Saboo, A.K., M. Morari and R.D. Colberg, "Resilience Analysis of Heat Exchanger Networks-I. Temperature Dependent Heat Capacities", *Comput. Chem. Engng.*, 11(4), 339 (1987a).
- Saboo, A.K., M. Morari and R.D. Colberg, "Resilience Analysis of Heat Exchanger Networks-II. Stream Splits and Flowrate Variations", *Comput. Chem. Engng.*, 11(5), 457 (1987b).
- Smith, R., "Heat Exchanger Network and Utilities: Capital and Total Cost Targets", *Chemical Process Design*, McGraw-Hill, New York, (1995).
- Su, J.L. and R. L. Motard, "Evolutionary Synthesis of Heat Exchanger Networks", *Comput. Chem. Engng.*, 8 67 (1984).
- Swaney, R.E. and I.E. Grossmann, "An Index for Operational Flexibility in Chemical Process Design, Part I&II", *AIChE J.*, 31(4), 621 (1985).
- Wistler, A.M. "Heat Exchanger as Money Makers", *Pet. Refiner.*, 27, 83 (1948).
- Wongsri, M., Resilient Heat Exchanger Network Design, Doctoral Dissertation, Washington University, (1990).

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ภาคผนวก ก บทความนำเสนอในการประชุมทางวิชาการ

RHEN: Resilient Heat Exchanger Network Designer

Montree Wongsri, and Nattaporn Songsiri

Department of Chemical Engineering
Chulalongkorn University, Bangkok 10330



สถาบันวิทยบริการ

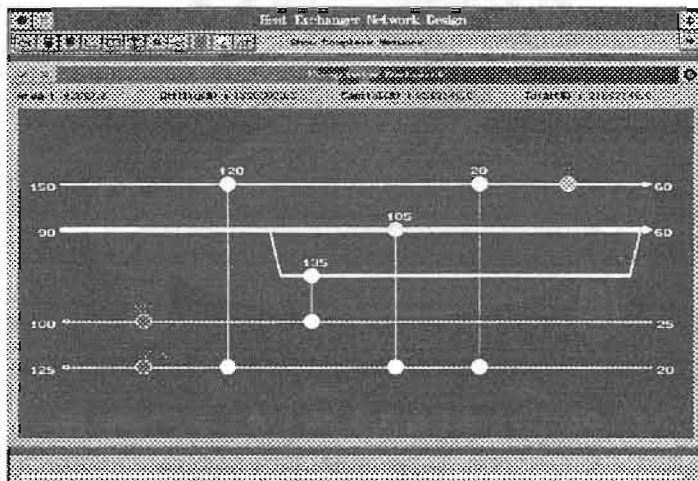
Presented at the Joint Sixth National Chemical Engineering Conference
and
the Second International Chemical Engineering Conference, Bangkok,
Thailand October, 1995.

RHEN: Resilient Heat Exchanger Network Designer

Montree Wongsri*, Nattaporn Songsiri

Department of Chemical Engineering, Chulalongkorn University, Bangkok 10330

A large number of heat exchanger network (HEN) design papers have been published in the past two decades. The existing HEN design approaches use heuristics and optimization. Most of the recent automated HEN synthesis programs have been developed by using optimization techniques. Although the optimization approach is more rigorous, it is limited by knowledge and insight. This paper presents a programming tool for automated HEN design by the heuristic approach. The program, RHEN, is a graphical software package for synthesis of resilient heat exchanger networks. The software uses optimal resilient heat exchanger network design methods developed by Wongsri (1990), i.e., the Match Pattern (MP) design method and the Disturbance Propagation (DP) method. RHEN is written in C++ language, and run on a DOS machine. Other features and functions are data input table, problem table, grid diagram, composite curves, grand composite curves and the cost of heat exchangers. RHEN can also optimize ΔT_{min} for capital and operating costs. It can find almost all possible network solutions. The solution networks obtained are resilient without further evolutionary development and feature maximum energy recovery.



INTRODUCTION

Typical chemical process industries like oil refineries and petrochemical plants have one common feature, i.e. they all use large amounts of energy. Reducing energy consumption leads to large cost savings and consequently the search for this reduction attracts the attention of engineers throughout the world. Another important character of chemical process industries is the fluctuation in operating conditions. To reduce the energy consumption in heating and cooling, the energy recovery network or heat exchanger network must be devised. The network designs must not only feature the economic optimum but also the resiliency characteristics namely, the ability to cope

* Correspondence concerning this paper should be addressed to M. Wongsri.

with fluctuations in operating conditions while still maintaining acceptable performance.

Good reviews covering HEN design are found in Nishida (1981) and Gundersen and Naess (1988). Two fundamentally different approaches for HEN synthesis are optimization technique and heuristic. Several computer programmed HEN design systems using optimization technique have been developed, e.g. MAGNETS by Floudas et al. (1986) and RESHEX by Saboo et al. (1987). But this approach does not utilize domain-specific problem-solving knowledge. In optimization, the problem data and design constraints are difficult to express mathematically, if not possible. Furthermore, there is no insight on how the solutions are obtained. It is also difficult to make the system interact with the designer during a problem solving process.

For a heuristic approach, there have been attempts to develop a computer programmed HEN design system by using a heuristic model. To automate synthesis we can use a rule-based system. A rule-based system offers an attractive feature that the knowledge in the rules can be built incrementally. Metha and Fan (1986) use a rule-based program on the Xerox LOOP system to synthesize a HEN. A more extensive use of the accumulated HEN synthesis heuristic knowledge-based system will make a rule-based system more powerful, and broaden the scope of problems that can be solved. Grimes et al. (1982), employ two match types to find a network with the aid of the search matrix for bookkeeping. Their method and heuristics used in written in OPS3X, and early rule-based language. However, the disadvantage of a rule-based program is that it quite takes much time to find more than one solutions.

RESILIENT HEAT EXCHANGER NETWORK DESIGN

In order to develop a resilience target, we need a definition of Resilience for HEN. Consider an uncertainty range of uncertain parameters (supply temperatures and flowrates) over which a HEN is expected to operate. A HEN structure is resilient in the giving uncertainty range if for every combination of supply temperatures and flowrates in the uncertainty range is: (1) achieves the specified target temperatures of each streams; (2) satisfies the specified minimum approach temperature (ΔT_{min}) in each exchanger; (3) satisfying a specified utility consumption constraint (minimum utility consumption, or some relaxation factors times minimum utility consumption).

Note that we are interested in the effects of HEN structure (topology) upon resilience before sizing individual exchangers, we define HEN resilience with respect to ΔT_{min} rather than area. Since resiliency is a property of a network structure, it can not be added to a design by merely increasing the size of number of components of a structure, but only by generating a proper structure. Resiliency must be considered at the beginning of a structure generating phase.

The resilient HEN synthesis methods presented by Marselle et al. (1982), Saboo et al. (1985), Flodas et al (1986) and Cerda et al (1990). Marselle et al. (1982) identify heuristically the extreme conditions to design a HEN and the network solution is obtained by combining the networks designed at the specified conditions. Saboo et al. (1985) improve the combination method by testing the feasibility of a solution network at all corner points. Flodas et al. (1986) use the combination method (multiperiod MILP transshipment model) to derive the superstructure multiperiod model and test its feasibility at the specified parameter points. Cerda et al. (1990) eliminated the trial and error nature of these methods by including the resilient requirement into the optimization model. The energy recovery is set up in tree levels with different priority.

Wongsri (1990) developed the heuristics and procedures for resilient heat exchanger network synthesis. The heuristics are used to developed basic and derived

match patterns which were classified according to their (1) resiliency (2) chances that they are in solutions and (3) the matching rules like the pinch method, and the thermodynamics law, etc. The match patterns are also ranked to these properties. Furthermore the same author developed the procedure for synthesizing heat exchanger network called the Disturbance Propagation Method to be used together with the match patterns. The match pattern concept is used with the developed disturbance load propagation technique to facilitate the bookkeeping of resiliency and the propagated disturbance loads. This method will find a resilient network structure directly from the resiliency requirement and also feature minimum number of units (MNU) and maximum energy recovery (MER).

In this work, the computer program for resilient heat exchanger network synthesis (RHEN, Release 1) is developed using the MP and DP methods of Wongsri (1990). RHEN Release 1 can solve Class I problem (Saboo and Morari, 1983), i.e. the case where small variations in inlet temperatures do not affect the pinch temperature location. The program RHEN is written in C++ and has graphical user interface. It runs on a DOS machine. A new version that can solve Class II problem, i.e. the case where large changes in inlet temperatures or flowrate variations cause the discrete changes in pinch temperature locations, is planned to be developed.

THE RHEN DESIGN METHOD

Usually, heat exchanger network synthesis is divided into 2 steps:

1. Network targeting
2. Network invention.

In the targeting step, the following important properties are determined before the actual network is designed. They are used as the targets for a design. (1) The maximum energy recovery (or the minimum utilities). The minimum utilities can be calculated by constructing the problem table (Linnhoff and Flower, 1978a). The values depend on the minimum approach temperature, ΔT_{min} , (2) The minimum number of matches (or units). The minimum number of matches is calculated from

$$N_{match,min} = N_{HotStream} + N_{ColdStream} - 1$$

If a problem is separated by pinch, this equation must be applied separately to each separated problem.

In the network invention step, RHEN uses the design methods developed by Wongsri (1990). They are summarized as follows:

HEURISTICS

Several HEN matching rules have been presented (Masso and Rudd, 1969, Ponton and Donalson, 1974, Rathore and Powers, 1975, Linnhoff and Hindmarsh, 1983, Jezowski and Hahne, 1986, Huang, Metha and Fan, 1988, etc.), They are, for example:

- *Heuristic C1*. To generate a network featuring the minimum number of heat transfer units, let each match eliminate at least one of the two streams; a tick-off rule (Hohmann, 1971).
- *Heuristic C2*. Prefer the matches that will leave a residual stream at its cold end for a heating problem, or its hot end for a cooling problem. A match of this type will feature the maximum temperature difference.
- *Heuristic C3*. Prefer matching large heat load streams together. The significance of this rule is that the control problem (a capital cost) of a match of this type (whether it is implemented by one or many heat exchangers) should be less than that of heating or cooling a large stream with many small streams.

- *Heuristic E1*. Divide the problem at the pinch into subproblems, one a heat sink (heating subproblem or hot end problem) and the other a heat source (cooling subproblem or cold end problem), and solve them separately (Linnhoff and Hindmarsh, 1983).
- *Heuristic E2*. Do not transfer heat across the pinch.
- *Heuristic E3*. Do not cool above the pinch.
- *Heuristic E4*. Do not heat below the pinch.
- *Heuristic T1*. In a heating problem, if a supply temperature of a cold stream is less than a target temperature of a hot stream by ΔT_{min} or more and the heat capacity flowrate of a hot stream is less than or equal to the heat capacity flowrate of a cold stream, the match between these two streams is feasible. Immediately above the pinch temperature, the heat capacity flowrate of a cold stream must be greater than or equal to that of a hot stream.
- *Heuristic T2*. In a cooling problem, if a supply temperature of a hot stream is greater than a target temperature of a cold stream by ΔT_{min} or more and the heat capacity flowrate of a hot stream is greater than or equal to the heat capacity flowrate of a cold stream, the match between these two streams is feasible. Immediately below the pinch temperature, the heat capacity flowrate of a hot stream must be greater than or equal to that of a cold stream.

Position of a match. One heuristic prefers a match at the cold end and another prefers a match at the hot end. Pinch heuristics prefers a match at the cold end in a heating subproblem and a match at the hot end in a cooling subproblem. However, there are other possibilities.

- By using the tick-off heuristic, there are four ways that two streams can match. This leads to the basic four match patterns. (Wongsri, 1990)

Heat load between hot and cold streams. The heuristics that concern heat load state that one must match large heat load hot and cold streams first. This leads to two additional heuristics:

- *Heuristic N1*. For a heating subproblem, a match where the heat load of a cold stream is greater than that of a hot stream should be given higher priority than the other. The reason is that the net heat load in heating subproblem is in deficit. The sum of heat loads of cold streams is greater than that of hot streams. The purposed match will likely be part of a solution, (Wongsri, 1990).
- *Heuristic N2*. Conversely, we prefer a match where the heat load of a hot stream is greater than that of a cold in a cooling subproblem, (Wongsri, 1990).

Residual heat load. No heuristics for this quantity have thus far appeared in the literature. Two new heuristics are introduced. For a match in a heating subproblem that satisfies the heat load preference,:

- *Heuristic N3*. We prefer a match where the residual heat load is less than or equal to the minimum heating requirement, (Wongsri, 1990).

For a match in a cooling subproblem that satisfies the heat load preference or heuristic N.2 :

- *Heuristic N4*. We prefer a match where the residual heat load is less than or equal to the minimum cooling requirement, (Wongsri, 1990).

The reasoning behind the above two heuristics N3 and N4 is that the residual may be matched to a utility stream. One has the possibility of eliminating two streams at once.

MATCH PATTERNS

A number of match patterns or substructure descriptions are generated and ranked according to the heuristics. A match pattern is a new concept used as a fundamental selection unit in a combinatorial HENs problem. There are four possible configurations that two streams can be match according to the comparative heat loads between hot and cold streams and the positions of a match. We then classified match patterns into four classes and simply call them Class A, Class B, Class C and Class D. See Figure 1.

Class A. A match of this class is a first type match; a match at cold end position and the heat load of the cold stream is greater than that of the hot stream. This is an upstream match. For a heating subproblem, a Class A match is favored, because it leaves a cold process stream (N1) at the hot end and follows the pinch heuristics.

Class B. A match of this class is a second type match; a hot end match and the heat load of the hot stream is greater than that of the cold stream. This is an upstream match. For a cooling subproblem, a Class B match is favored, because it leaves a hot process stream at the cold end (N2) and also follows the pinch heuristics.

Class C. A match of this class is a first type match; a cold end match and the heat load of the hot stream is greater than that of the cold stream. This is a downstream match.

Class D. A match of this class is a second type match; a hot end match and the heat load of the cold stream is greater than that of the hot stream. This is a downstream match.

A match of Class A or C will leave a residual at the hot end, while a match of Class B or D will leave a residual at the cold end. Heuristics N.3 and N.4 will be use heuristics to further subclassify matches of Class A and B into matches of high priority.

Derivative Match Patterns

Subclass AH. A match of this subclass is a member of Class A, a heating subproblem where the residual is less than or equal to the minimum heating requirement. See Figure 2.

Subclass BK. A match of this subclass is a member of Class B, a cooling subproblem where the residual is less than or equal to the minimum cooling requirement. See Figure 2.

A match of subclasses AH in a heating subproblem and BK in a cooling subproblem have the highest priorities. We further discriminate match patterns according to heat capacity flowrate. By following pinch heuristics, in a heating subproblem, we prefer a match where the heat capacity flowrate of a cold stream is greater than or equal to that of a hot stream. For example, A[H] is a match in which the heat capacity flowrate of the cold stream and the residual of the cold stream is matched to the heating utility. Similarly in a cooling subproblem, we prefer a match where the heat capacity flowrate of the hot stream is greater or equal to that of the cold stream. For example, B[C] is a match in which the heat capacity flowrate of the hot stream is greater than that of the cold stream and the residual of the hot stream is matched to the cooling utility.

For example, the rankings of the match patterns in a heating subproblem are AH, A[H], B[C], A[C], B[H], C[H], D[C], C[C] and D[H]; for a cooling subproblem BK, B[C], A[H], B[H], A[C], D[C], C[H], D[H] and C[C]. There are more derivative match patterns, see Wongsri (1990).

Resilient Match Patterns

We are now in a position to discuss the resiliency of the four match pattern classes. Resiliency of a match pattern can be achieved if the disturbances in input

conditions of the hot and cold streams can be transferred to the active stream (a residual portion). In effect, the transferred disturbances are the responsibility of and will be managed by the resiliency of the active stream whose resiliency is yet to be determined. So, the degree of resiliency depends on how much the disturbances can be passed to the residual. In general, resiliency of any individual match pattern depends on (1) the resiliency of the active residual stream, (2) the resiliency of the heat exchanger.

For Class A and Class B match patterns, the disturbance of a member stream can be transfer to the residual. So, they are considered to be potential resilient match patterns. For the Class C and Class D match pattern, we can see that only the disturbances of a hot stream in Class C and of a cold stream in Class D can be managed but neither a cold stream in Class C nor a hot stream in Class D. Since these two classes cannot handle disturbance of one of their streams, they are considered non-resilient match patterns.

DISTURBANCE PROPAGATION DESIGN METHOD

In order for a stream to be resilient with a specified disturbance load, the disturbance load must be transferred to heat sinks or heat sources within the network. With the use of the heuristic : *To generate a heat exchanger network featuring the minimum number of heat transfer units, let each match eliminate at least one of the two streams.*

We can see that in a match of two heat load variable streams, the variation in heat load of the smaller stream S1 (see Figure 3) will cause a variation to the residual of the larger stream S2 by the same degree: in effect the disturbance load of S1 is shifted to the residual of S2. If the residual stream S2 is matched to S3 which has larger heat load, the same situation will happen. The combined disturbance load of S1 and S2 will cause the variation in the heat load to the residual S3. Hence, it is easy to see that the disturbance load in residual S3 is the combination of its own disturbance load and those obtained from S1 and S2. Or, if S2 is matched to a smaller heat load stream S4, the new disturbance load of residual S2 will be the sum of the disturbance loads of S1 and S4. From this observation, in order to be resilient, a smaller process stream with specified disturbance load must be matched to a larger stream that can tolerate its disturbance. In other words, the propagated disturbance will not overshoot the target temperature of the larger process stream.

However, the amount of disturbance load that can be shifted from one stream to another depends upon the type of match patterns and the residual heat load. Hence, in design we must choose a pattern that yields the maximum resiliency. We can state that the resiliency requirement for a match pattern selection is that the entire disturbance load from a smaller heat load stream must be tolerated by a residual stream. Otherwise, the target temperature of the smaller stream will fluctuate by the unshifted disturbance. Of course, the propagated disturbance will be finally handled by utility exchangers. In short, the minimum heat load value of a larger stream must be less than a maximum heat load value of a smaller stream. The propagated disturbance will proportionally cause more temperature variation in the residual stream and the range of temperature variation of the residual stream will be larger than its original range. See Figure 4.

Hence, a stream with no original variation in heat load will be subjected to variation in heat load if it is matched to a stream with disturbance. Another design consideration is that the disturbance load travel path should be as short as possible, i.e. the least number of streams involved. Otherwise, the accumulated disturbance will be at high level. From the control point of view, it is difficult to achieve good control if the order of the process and the transportation lag are high. From the design viewpoint, one may not find heat sinks or sources that can handle the large amount of propagated disturbance.

RHEN MANUAL

The following section is to show how RHEN is run.

When you start RHEN, you will see the main menu as shown in Figure 5.

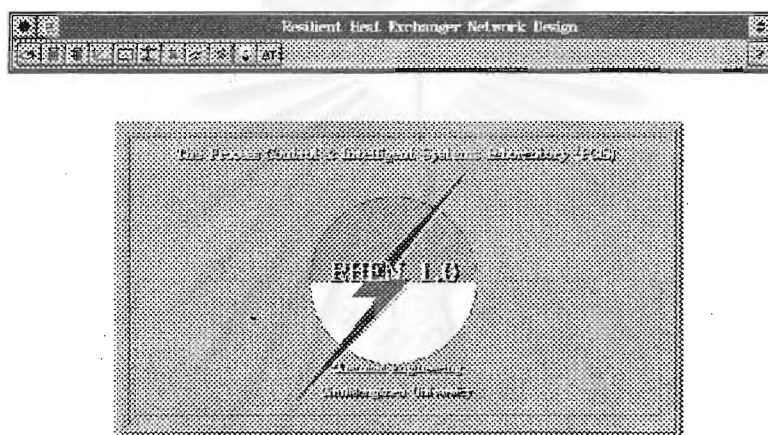


Figure 5. Main menu of program, RHEN.

Now RHEN is ready to solve a problem. Click the first icon to input number of hot streams, number of cold streams and ΔT_{min} . See Figure 6.

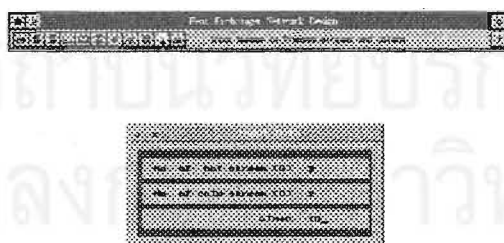


Figure 6. Table to input number of hot streams, number of cold streams and ΔT_{min} .

Then, click the second icon, RHEN will display a table for inputting temperatures and heat capacity flowrates of all streams, including of utility streams. (Figure 7).

STREAM	F. Cp	SUPPLY TEMPS			TARGET TEMPS
		Hot/Cold	Hot/Cold	Minimum	Hot/Cold
H1	1	610	615	610	350
H2	2	750	750	750	500
C1	2.8	340	340	340	420
C2	2	415	415	415	500
HU					
CU	3	250			

Figure 7. Input temperatures and heat capacity flowrates.

If the streams have flowrate disturbances. Click The *heat capacity flowrate disturbance* icon. (Figure 8).

STREAM	F. Cp	
	Min Flowrate	Max Flowrate
H1	1.85	1
H2	2	2
C1	3	2.8
C2	2	2

Figure 8. Input heat capacity flowrate variations.

Figure 9. shows the problem table. The problem table shows the utility requirement and the pinch temperature and figure 10 shows the grid diagram. Figure 11 shows temperature-enthalpy composite curves. Figure 12 shows grand composite curves.

Heat Exchanger Network Design

Problem Table, Grid Matrix, Grand and Target

Stream	Flow	Temp	Temp	Temp	Temp	Temp	Temp
H-1	750	700	-200	-200	-100	0	-200
C-1	800	300	-20	-20	-100	200	-20
C-2	200	270	-20	-20	200	240	20
H-2	400	400	100	-100	300	300	100
C-3	120	410	10	-10	100	100	10
H-3	350	300	100	-100	100	100	100

Figure 9. The Problem table.

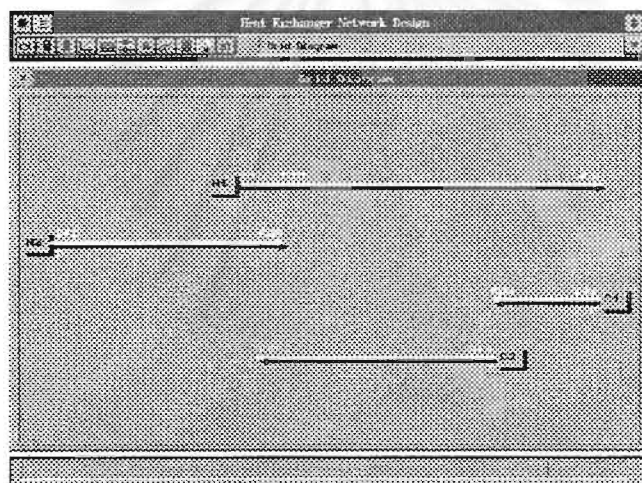


Figure 10. Grid Diagram.

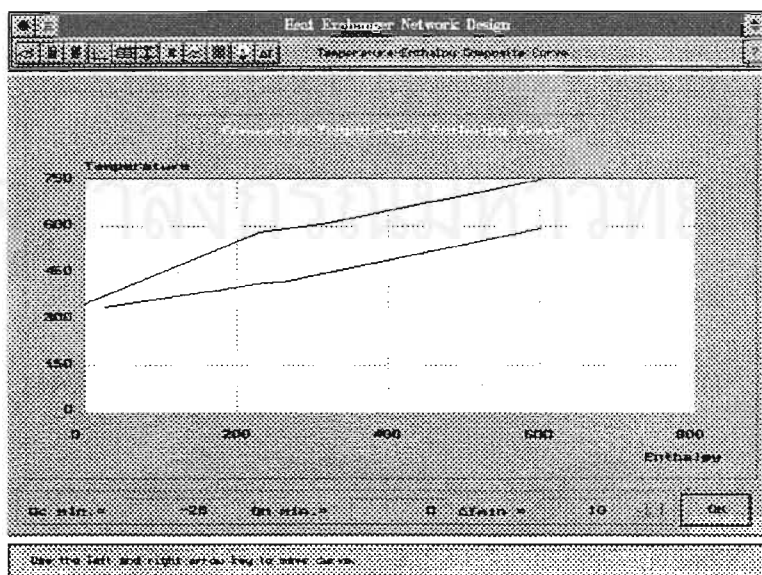


Figure 11. Composite curves.

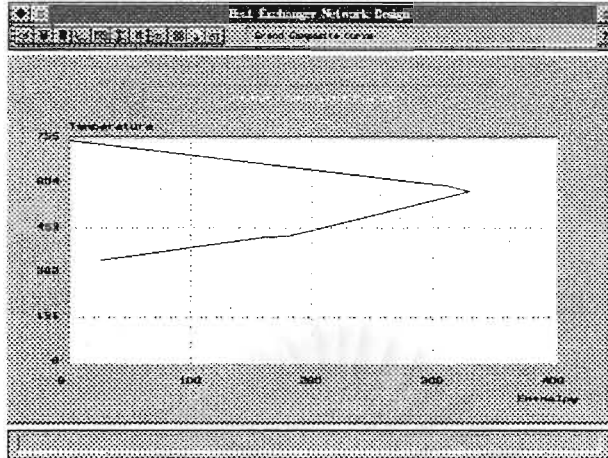


Figure 12. Grand composite curve.

To find the solutions of the problem, Click the *match streams* icon. For this particular problem, seven solutions are found. Figure 13. show the 7th solution.

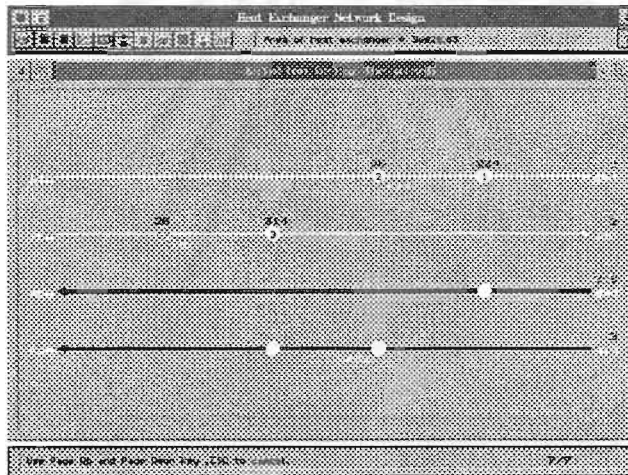


Figure 13. The seventh solution.

If there are many solutions like this problem and you want to find what the solution is minimum cost, Click the 10th icon. The result is the solution No. 3. See Figure 14 .

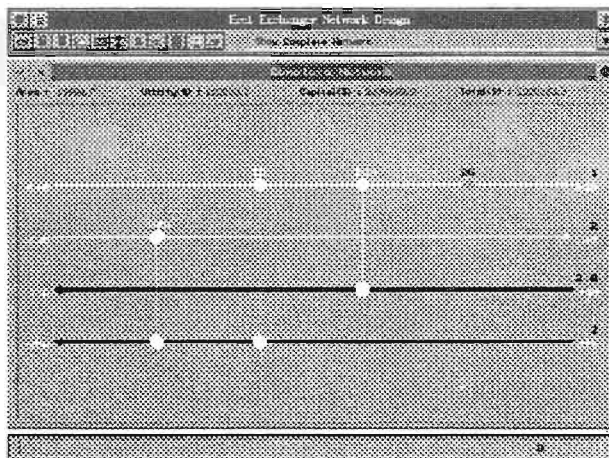


Figure 14. The solution No. 3 is minimum cost.

To optimize ΔT_{min} for minimum capital and operating cost. Click the ΔT icon. See Figure 15. Optimum ΔT_{min} of this problem is 11° .

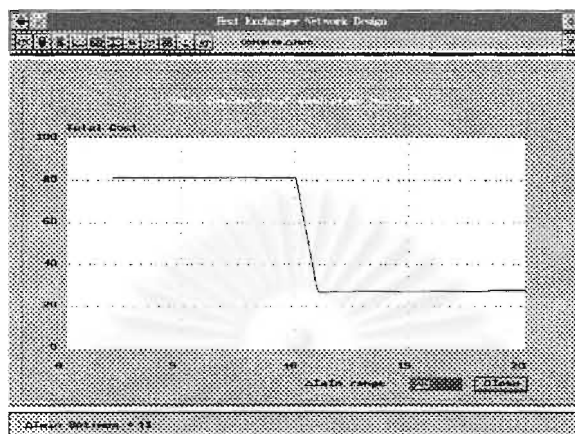


Figure 15. The optimum ΔT_{min} is 11° .

To find resilient structure, Click the *R* icon to change from non-resilient to resilient mode.

Figure 16 shows the resilient structure that can operate in the specified variations range.

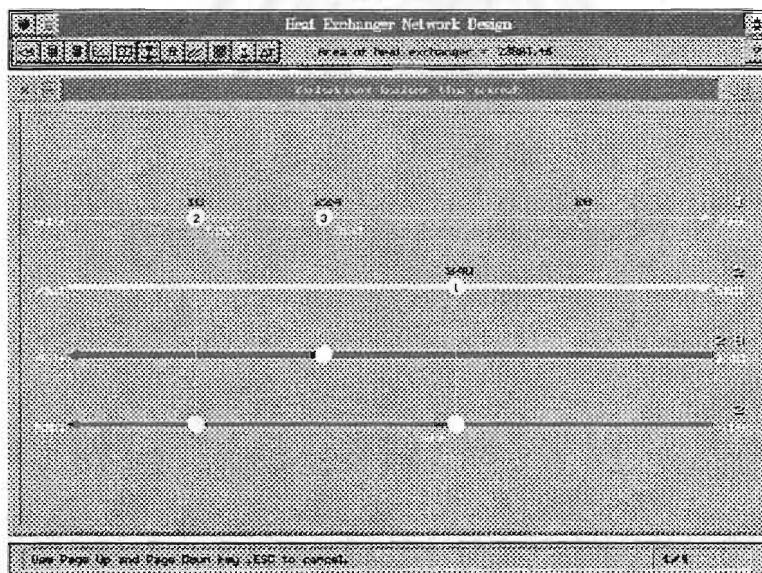


Figure 16. Resilient network structure.

SUMMARY

RHEN is a computer program for resilient heat exchanger network written in C++. HEN uses the heuristics and the design procedures developed by Wongsri (1990). RHEN has a graphical user interface running on a personal computer. The program is easy to use featuring the problem table, the grand composite curve, the grid diagram, ΔT_{min} optimization, resilient network, minimum cost network.

REFERENCES

1. Montree Wongsri, "Resilient Heat Exchanger Network Design", Washington University, 1990.
2. R.D. Colberg and M. Morari, "A Resilience Target For Heat Exchanger Network Synthesis", Computers Chem. Engng., Vol.13, No. 7, 1989.
3. Prateep Arunwatanamongkol, "A computer Program for Heat Exchanger Network Design Using Match Pattern Approach", Chulalongkorn University, 1992.



สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ภาคผนวก ข

ตัวอย่างการทดสอบโปรแกรม

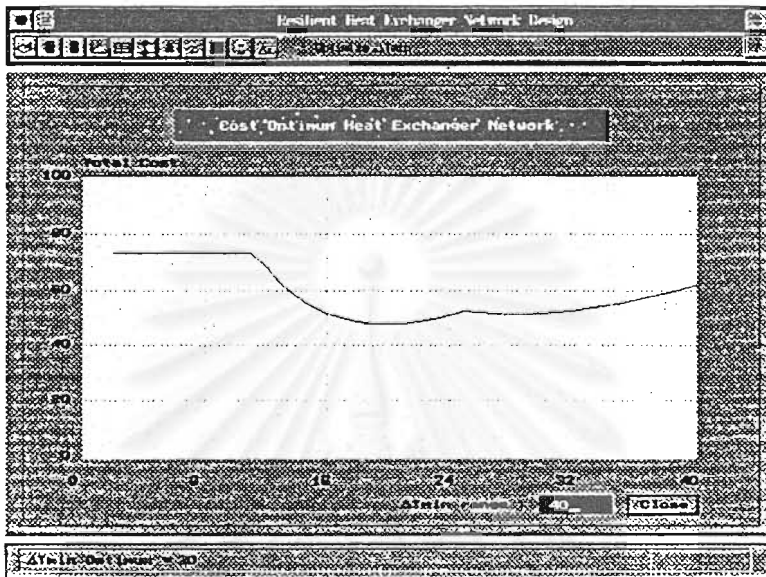
ตัวอย่างทดสอบโปรแกรม ข.1 ตารางข้างล่างนี้แสดงข้อมูลของกระแส สำหรับข้อมูลค่าใช้จ่ายของช่างงานในตัวอย่างนี้และตัวอย่างต่อไป ให้ใช้ข้อมูลตามตัวอย่างที่ 3.3

กระแส	W (kW/°C)	อุณหภูมิขาเข้า (°C)	อุณหภูมิขาออก (°C)	h (kW/m ² ·°C)
H1	3.0	150	60	0.00002
H2	8.0	90	60	0.00003
C1	3.0	25	100	0.00080
C2	2.5	20	125	0.00060
HU	7.5	175	-	0.00002
CU	10.0	35	-	0.00002

ปัญหานี้ ทำการออปติไมซ์ ΔT_{min} ก่อน ซึ่งจะให้ค่า ΔT_{min} ที่ออปติไม้มเท่ากับ 20 °C

(รูปที่ ข.1) จากการวิเคราะห์ด้วยวิธีตารางปัญหาจะได้อุณหภูมิพินช์เท่ากับ 90 °C, $Q_{H,min}$ เท่ากับ 47.5 kW และ $Q_{C,min}$ เท่ากับ 70 kW (รูปที่ ข.3) จากนั้นดูการแบ่งช่างงานออกเป็นช่างงานย่อยสองช่างงานด้วยอุณหภูมิพินช์ในรูปที่ ข.4 สำหรับรูปที่ ข.5-ข.7 แสดงคำตอบของช่างงาน

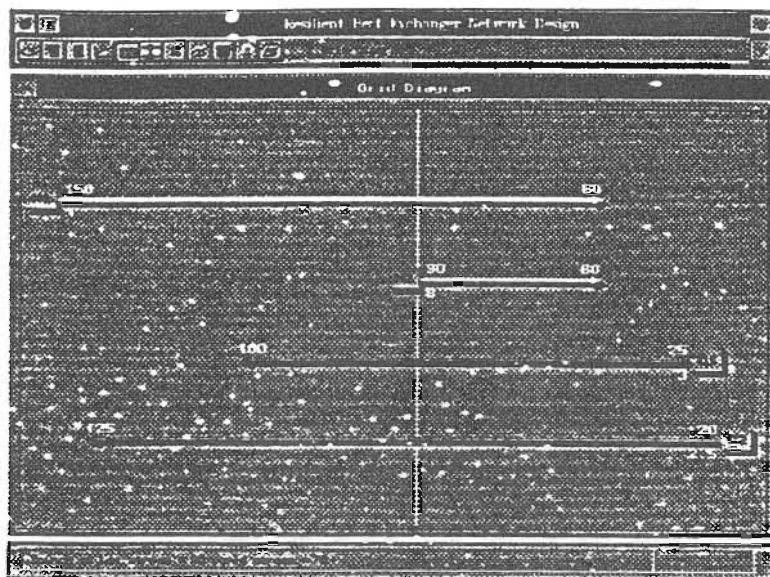
เหนือจุดพินช์ซึ่งมีทั้งหมดสามคำตอบ และรูปที่ ข.8-ข.21 แสดงคำตอบของช่างงานใต้จุดพินช์ มีทั้งหมด 14 คำตอบ ปัญหาสำหรับการทดสอบการแยกกระแส (Stream splitting)



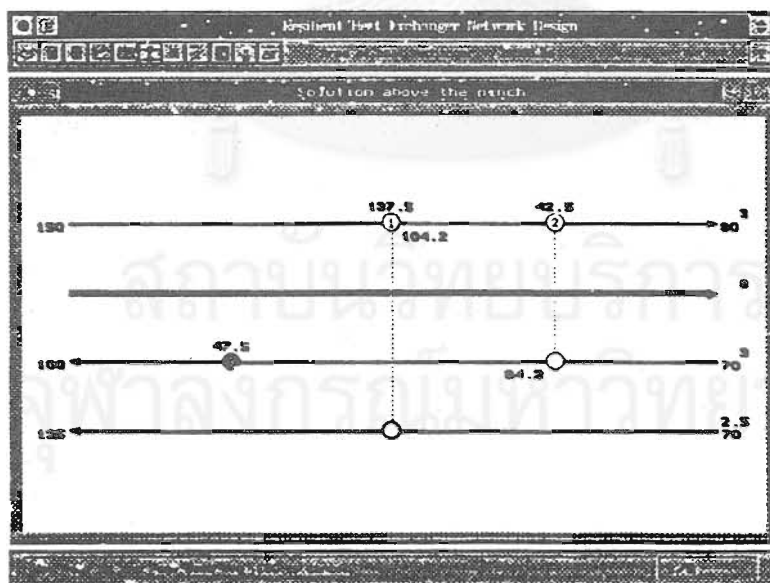
รูปที่ ข.1 แสดง ΔT_{min} ที่ออกแบบเท่ากับ 20°C

Subnetwork (SN)	Temp	Min. Heating (kM)	Accum. (kM)		Network Min. (kM)	
			Heating	Cooling	Heating	Cooling
SW1	15	15				
SW1	145	125	15	22.5	47.5	62.5
SH2	120	100	12.5	27.5	62.5	75
SH3	80	70	7.5	47.5	75	0
SH4	60	40	18.5	117.5	70	185
SH5	45	25	82.5	25	185	62.5
SH6	40	20	117.5	22.5	62.5	70

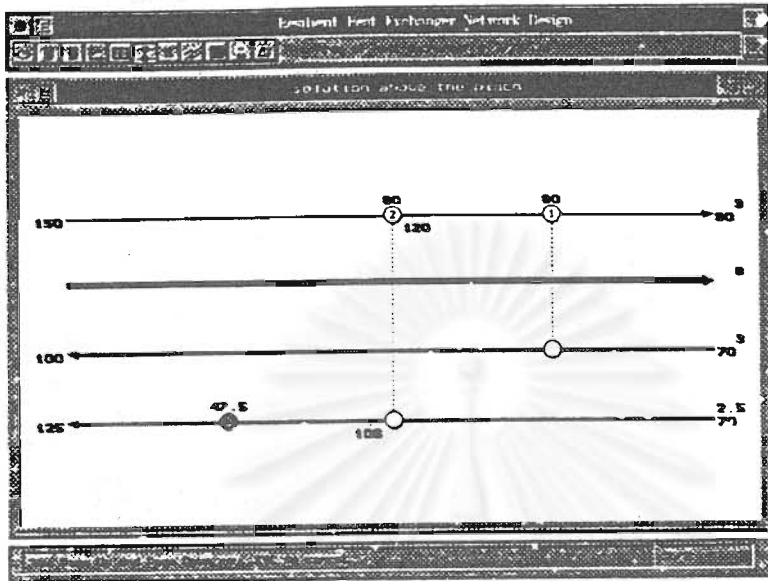
รูปที่ ข.2 ตารางปัญหา



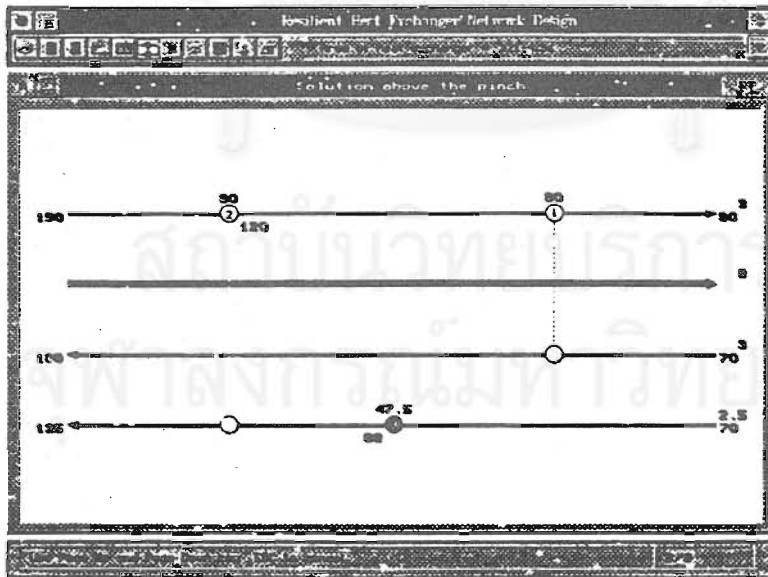
รูปที่ ข.3 Grid diagram ของตัวอย่างที่ ข.1.



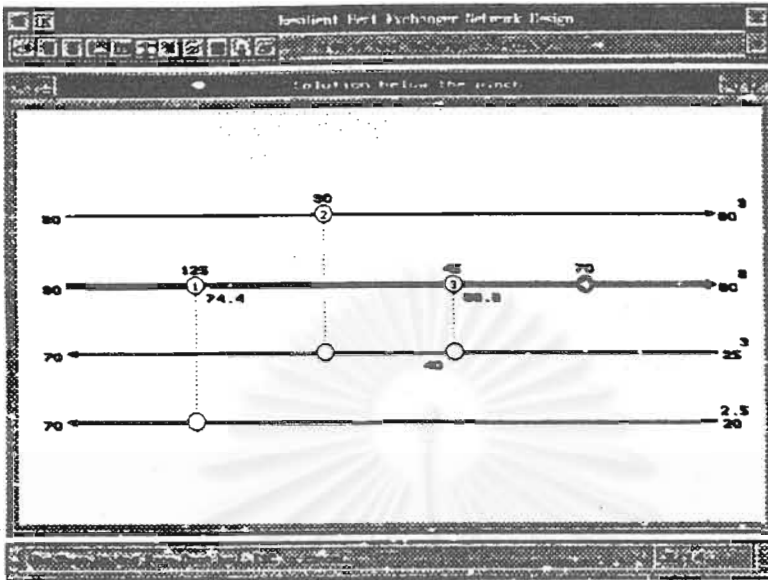
รูปที่ ข.4 ข่ายงานเหนือจุดพินช์คำตอบที่ 1/3.



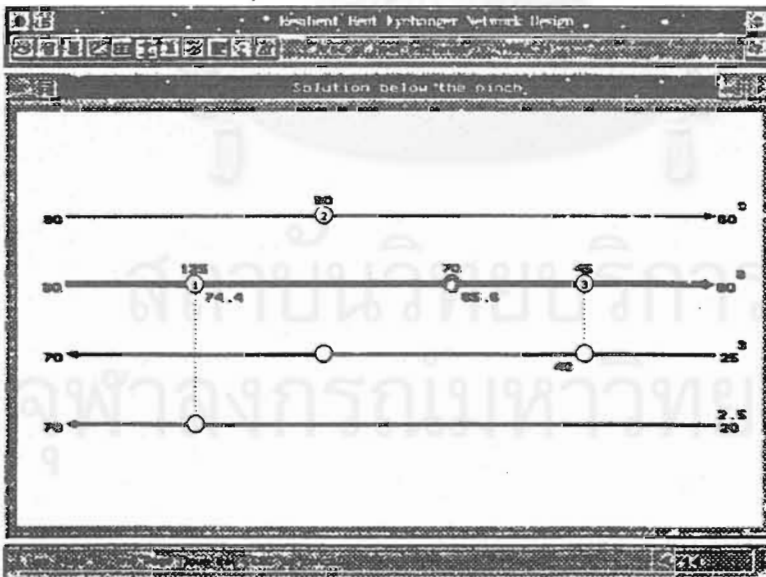
รูปที่ ข.5 ข่ายงานเหนือจุดพินช์คำตอบที่ 2/3.



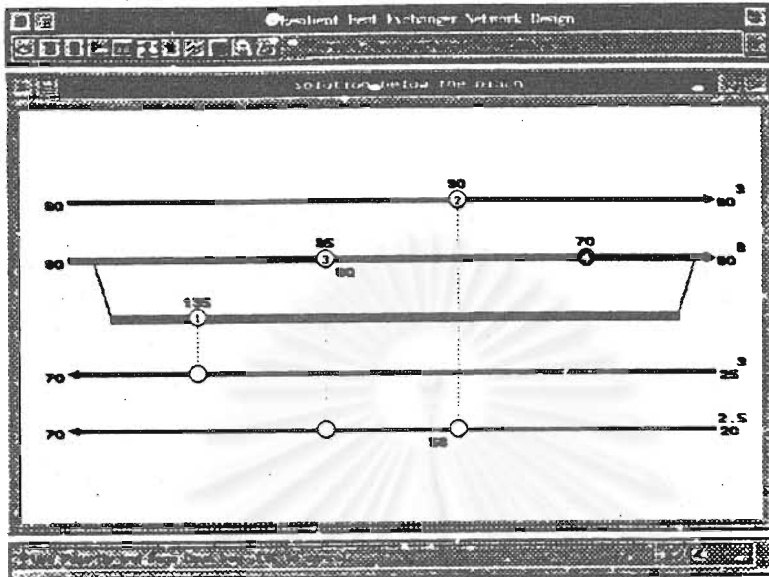
รูปที่ ข.6 ข่ายงานเหนือจุดพินช์คำตอบที่ 3/3.



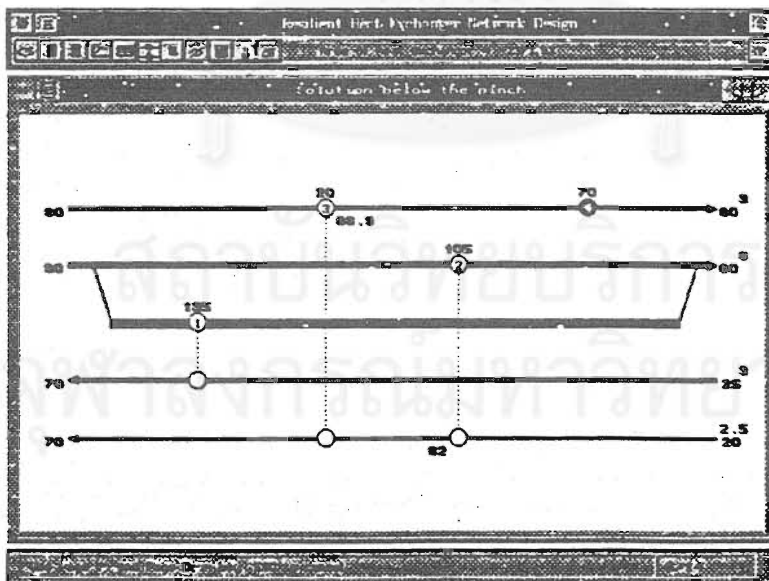
รูปที่ ข.9 ข่ายงานได้จุดพินช์คำตอบที่ 3/14.



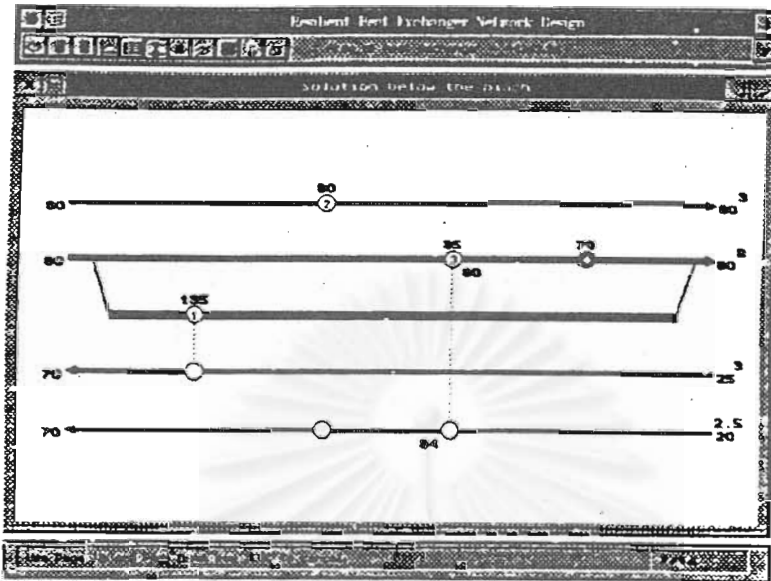
รูปที่ ข.10 ข่ายงานได้จุดพินช์คำตอบที่ 4/14.



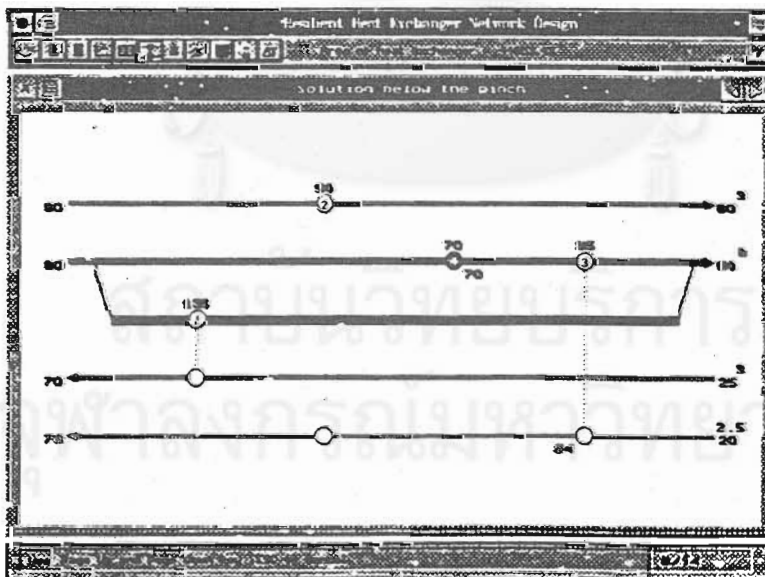
รูปที่ ข.11 ข้างงานใต้จุดพินซ์ค่าตอบที่ 5/14.



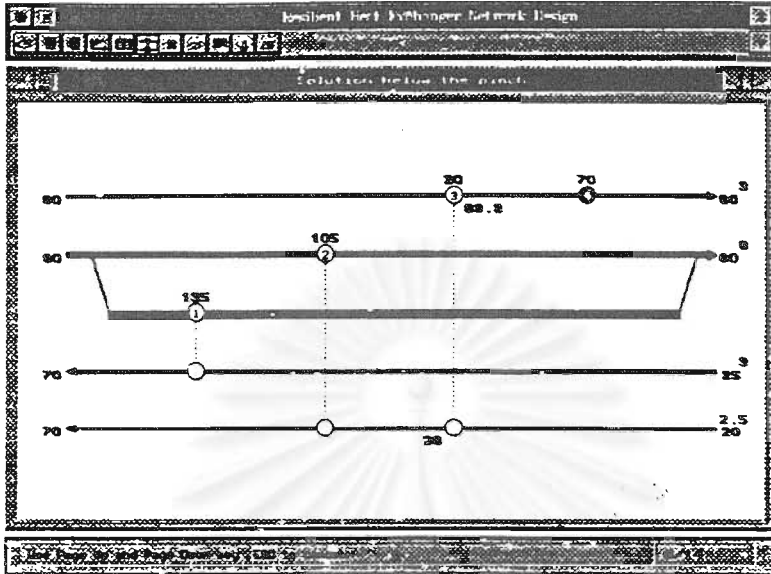
รูปที่ ข.12 ข้างงานใต้จุดพินซ์ค่าตอบที่ 6/14.



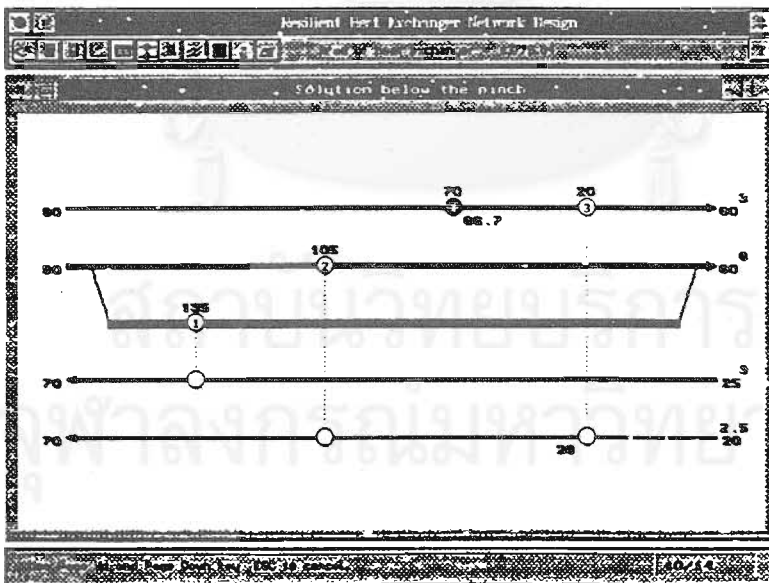
รูปที่ ข.13 ข่ายงานไต้จุดพินช์คำตอบที่ 7/14.



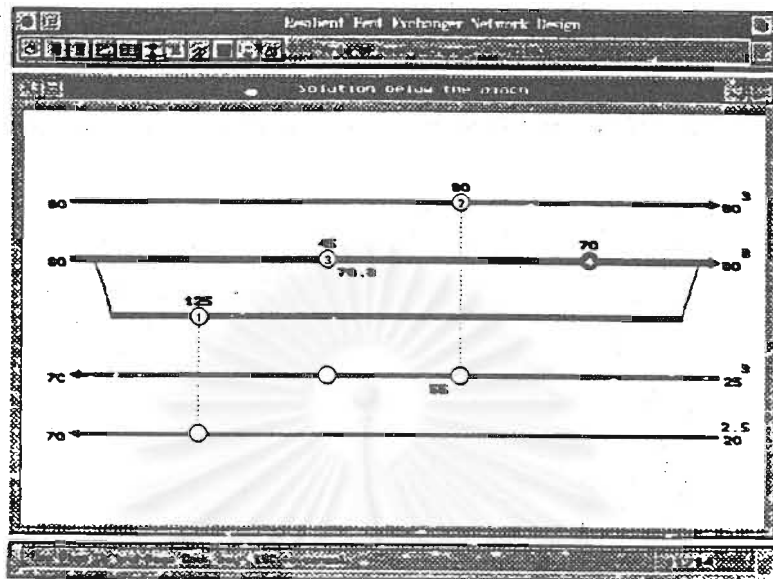
รูปที่ ข.14 ข่ายงานไต้จุดพินช์คำตอบที่ 8/14.



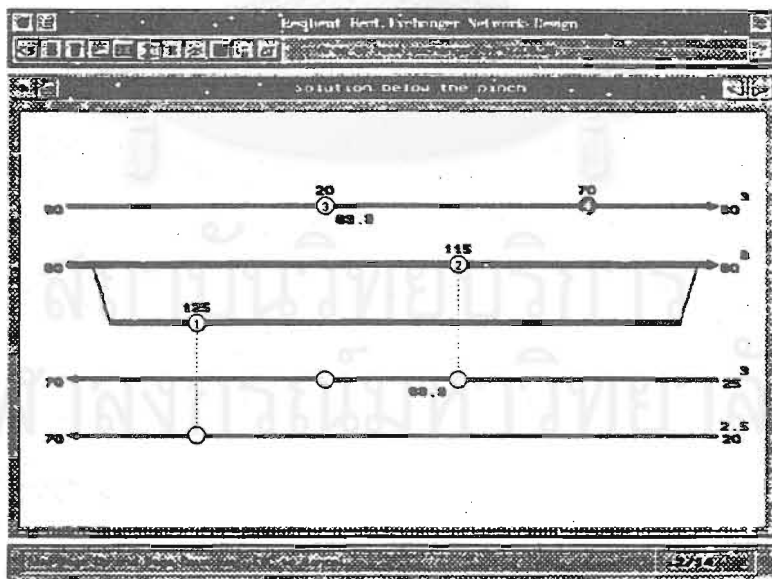
รูปที่ ข.15 ข่ายงานใต้จุดพินช์คำตอบที่ 9/14.



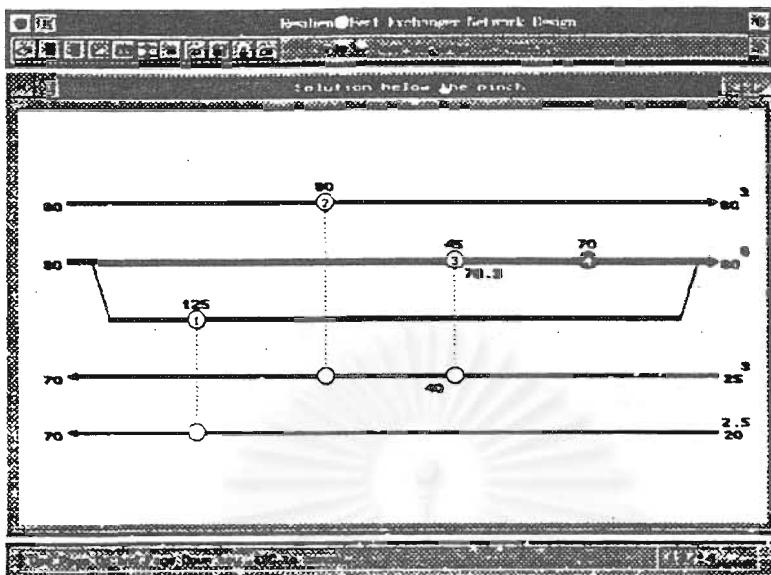
รูปที่ ข.16 ข่ายงานใต้จุดพินช์คำตอบที่ 10/14.



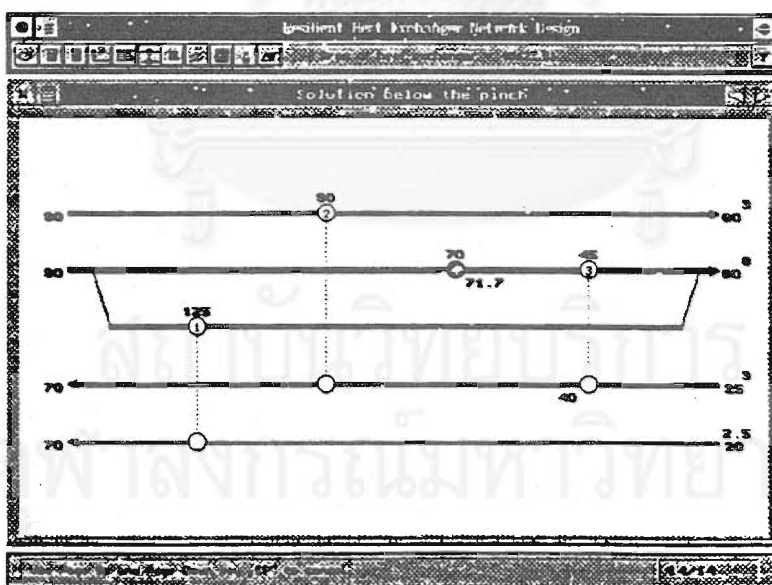
รูปที่ ข.17 ข่ายงานได้จุดพินช์คำตอบที่ 11/14.



รูปที่ ข.18 ข่ายงานได้จุดพินช์คำตอบที่ 12/14.



รูปที่ ข.19 ข่ายงานได้จุดพินช์คำตอบที่ 13/14.



รูปที่ ข.20 ข่ายงานได้จุดพินช์คำตอบที่ 14/14.

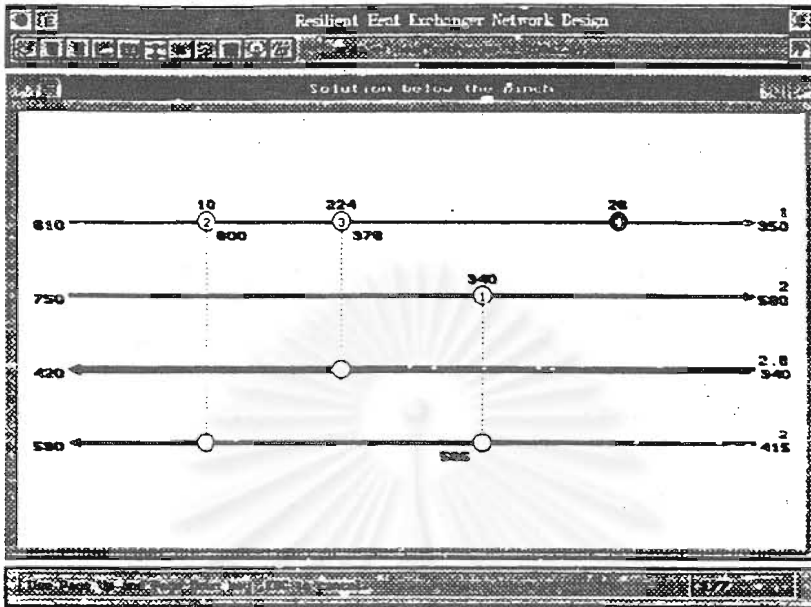
ตัวอย่างทดสอบโปรแกรม ข.2 กำหนด $\Delta T_{min} = 10\text{ }^{\circ}\text{C}$ (จากตัวอย่างของบทที่ 4)

กระแส	W (kW/ $^{\circ}\text{C}$)		อุณหภูมิขาเข้า ($^{\circ}\text{C}$)		อุณหภูมิขาออก ($^{\circ}\text{C}$)
	สูงสุด	ต่ำสุด	สูงสุด	ต่ำสุด	
H1	1.85	1.00	615	610	350
H2	-	2.00	-	750	580
C1	3.00	2.80	-	340	420
C2	-	2.00	-	415	590

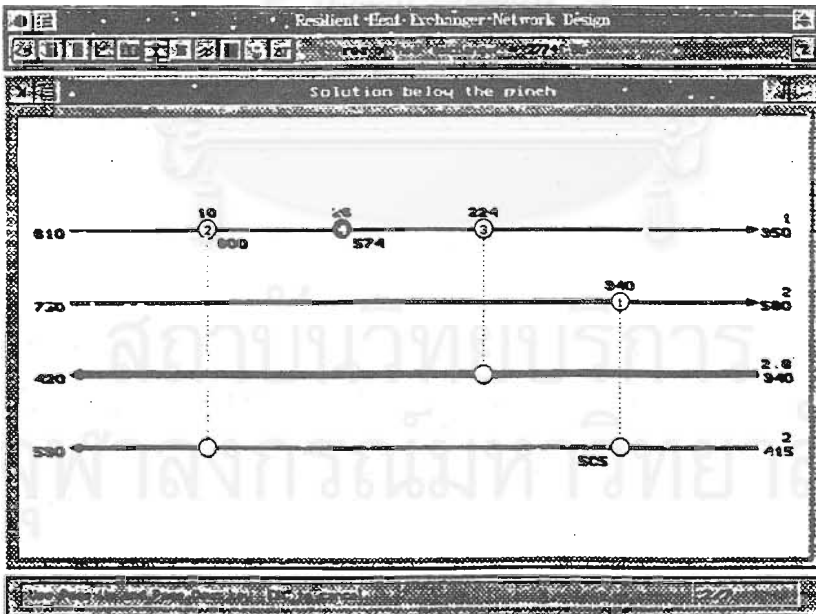
ปัญหานี้ไม่มีอุณหภูมิพินช์ ค่า $Q_{H,min} = 0$ โดย $Q_{C,min} = 26\text{ kW}$ หรือเรียกว่าเป็นปัญหาเกี่ยวกับทางด้านความเย็นอย่างเดียว (สร้างकुलเลอร์ในข่ายงานที่กระแสร้อน ไม่มีฮีตเตอร์) สำหรับตารางปัญหาให้ดูรูปที่ ข.22 และโครงสร้างข่ายงานเมื่อไม่คิดความแปรปรวนหรือเป็นข่ายงานแบบไม่ยืดหยุ่นนั้น จะได้คำตอบทั้งหมด 7 โครงสร้างด้วยกัน (รูปที่ ข.23-ข.29) และเมื่อใช้เงื่อนไขข่ายงานแบบยืดหยุ่นแล้วจะได้โครงสร้างเพียง 1 โครงสร้างเท่านั้น (รูปที่ ข.30)

Subnetwork (SN)	Temp	Min. Heating (kW)	Accum (kW)	Network Min. (kW)
H1	750 740	280	280	0 280
H2	800 590	80	310 251	290 310
H3	580 570	20	830 284	310 330
C1	480 420	150	180 304	330 180
C2	425 415	18	181 154	180 181
C3	350 340	185	28 185	181 28

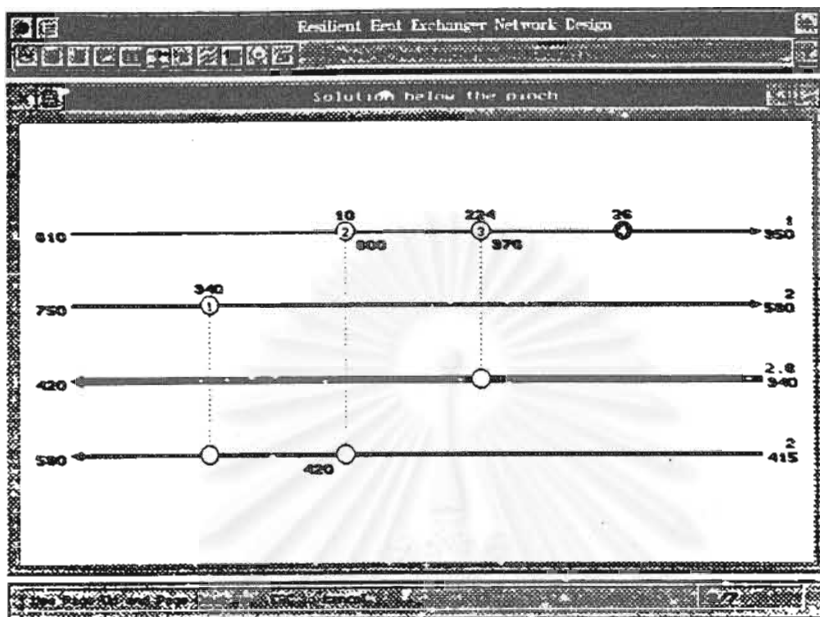
รูปที่ ข.22 แสดงตารางปัญหา



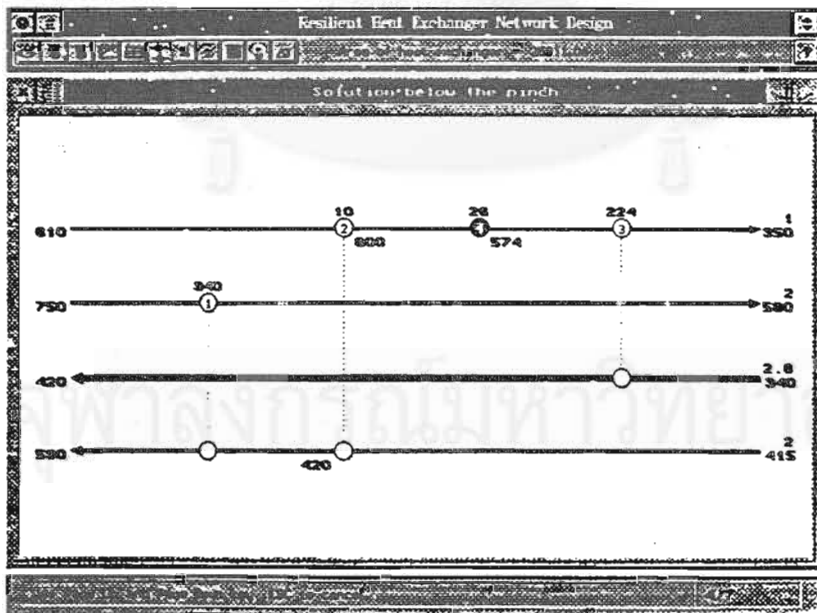
รูปที่ ข.23 ข่างงานค่าตอบที่ 1/7



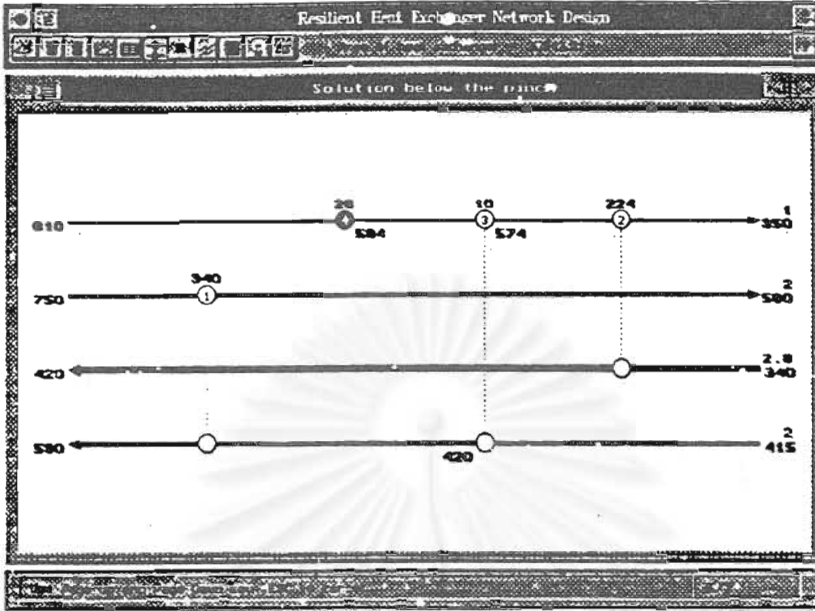
รูปที่ ข.24 ข่างงานค่าตอบที่ 2/7



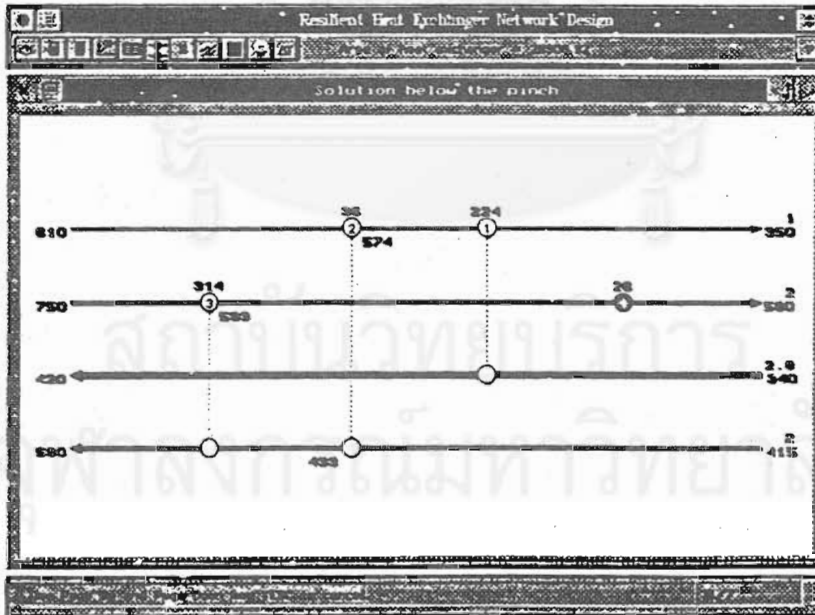
รูปที่ ข.25 ข่า่งานคำตอบที่ 3/7



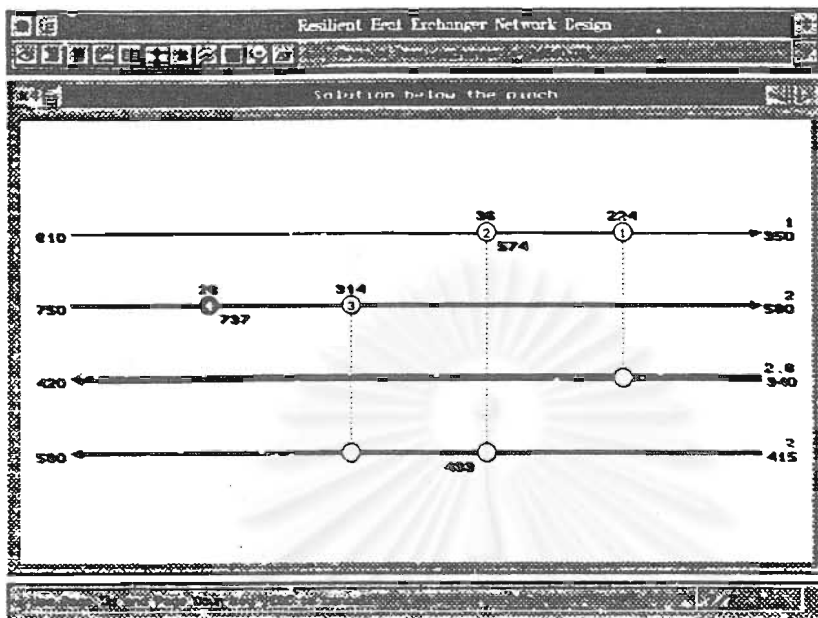
รูปที่ ข.26 ข่า่งานคำตอบที่ 4/7



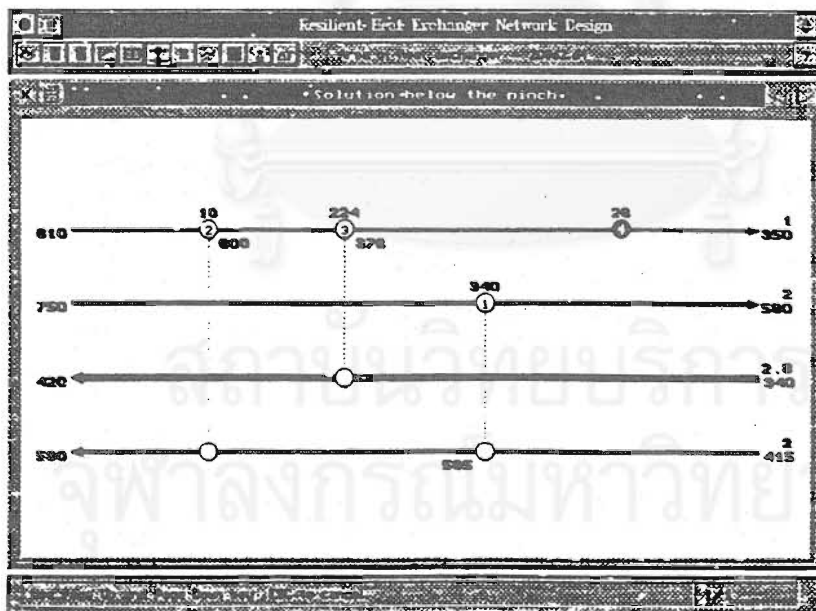
รูปที่ ข.27 ข่ายงานค่าตอบที่ 5/7



รูปที่ ข.28 ข่ายงานค่าตอบที่ 6/7



รูปที่ ข.29 ขำงานค่าตอบที่ 7/7

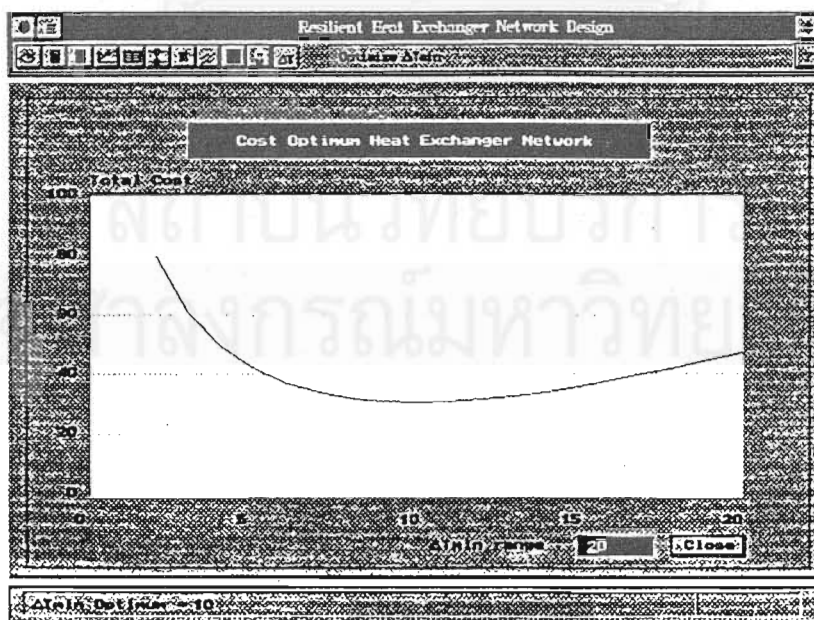


รูปที่ ข.30 แสดงขำงานแบบฮีดหุ่ไน นที่นี้ไ้เพียงโครงสร้างเดียว

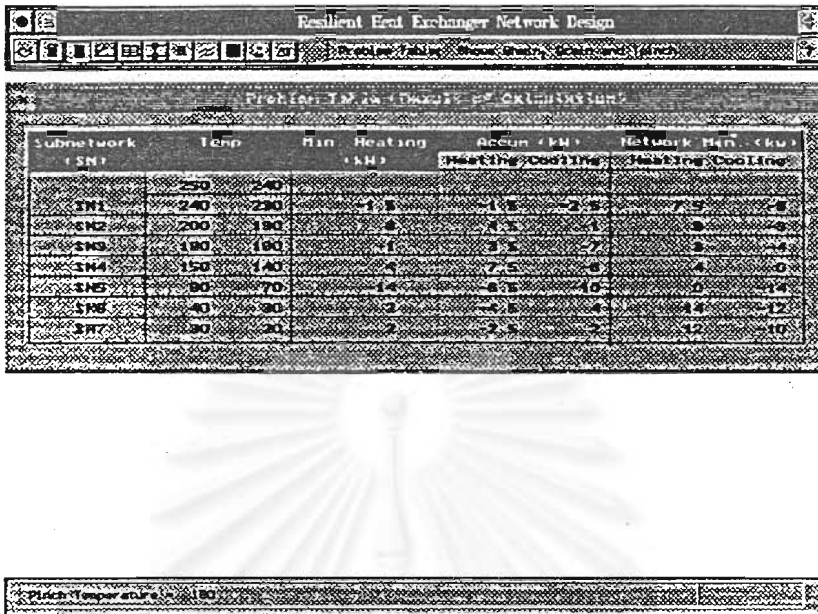
ตัวอย่างทดสอบโปรแกรม ข.3 (จากตัวอย่างที่ 3.3)

กระแส	W (kW/°C)	อุณหภูมิขาเข้า (°C)	อุณหภูมิขาออก (°C)	h (kW/m ² ·°C)
H1	0.15	250	40	0.0010
H2	0.25	200	80	0.0008
C1	0.20	20	180	0.0006
C2	0.30	140	230	0.0008
HU	7.5	240	-	0.0030
CU	1.0	20	-	0.0010

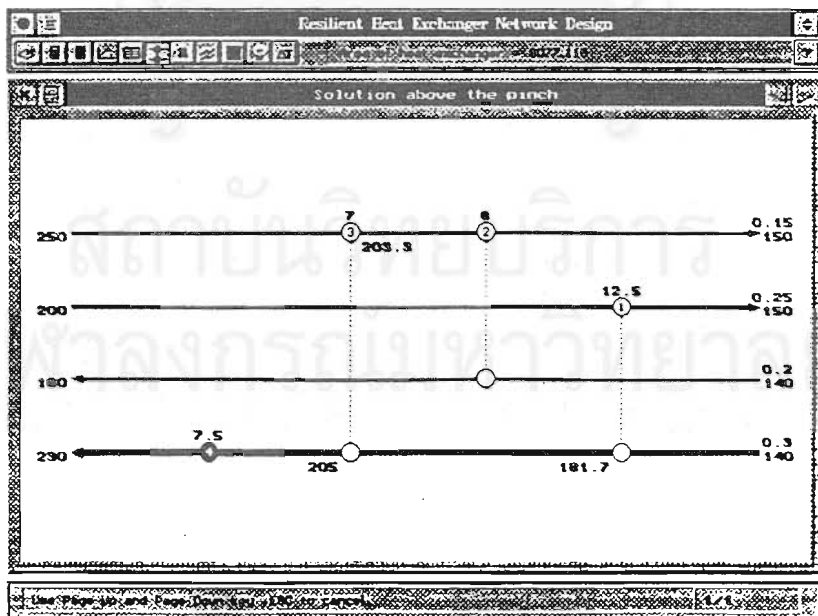
รูปที่ ข.31 แสดงการออปติไมซ์ ΔT_{min} ในเส้นโค้งคอมโพสิตสมมูล ซึ่งจะได้ค่าออกมาที่ 10 °C จากวิธีตารางปัญหา (รูปที่ ข.32) ได้ $Q_{H,min}$ เท่ากับ 7.5 kW, $Q_{C,min}$ เท่ากับ 10 kW และ อุณหภูมิพินช์เท่ากับ 150 °C รูปที่ ข.33-ข.36 แสดงข่ายงานเหนือจุดพินช์และได้จุดพินช์ที่หาได้ รูปที่ ข.37 แสดงข่ายงานที่ประหยัดค่าใช้จ่ายมากที่สุดเหนือจุดพินช์และได้จุดพินช์เข้าด้วยกัน



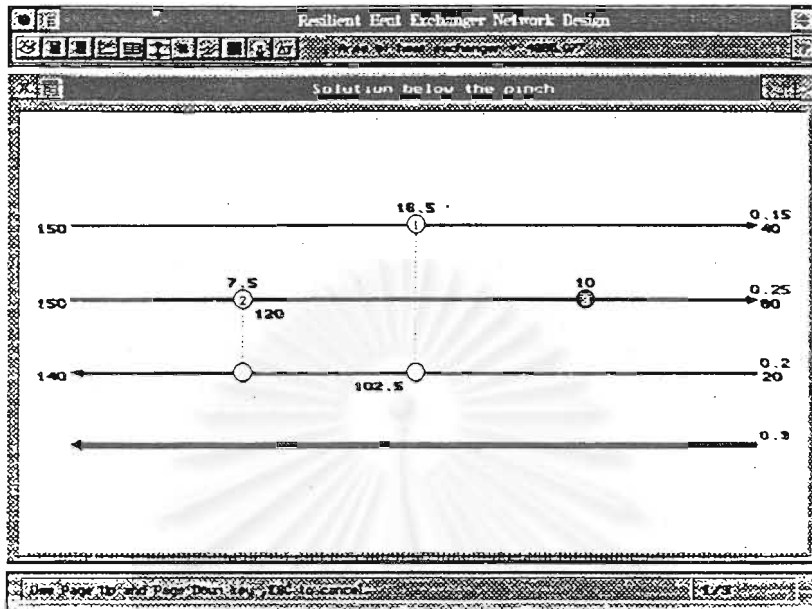
รูปที่ ข.31 ได้ค่า ΔT_{min} ที่ออปติไม้มเท่ากับ 10 °C



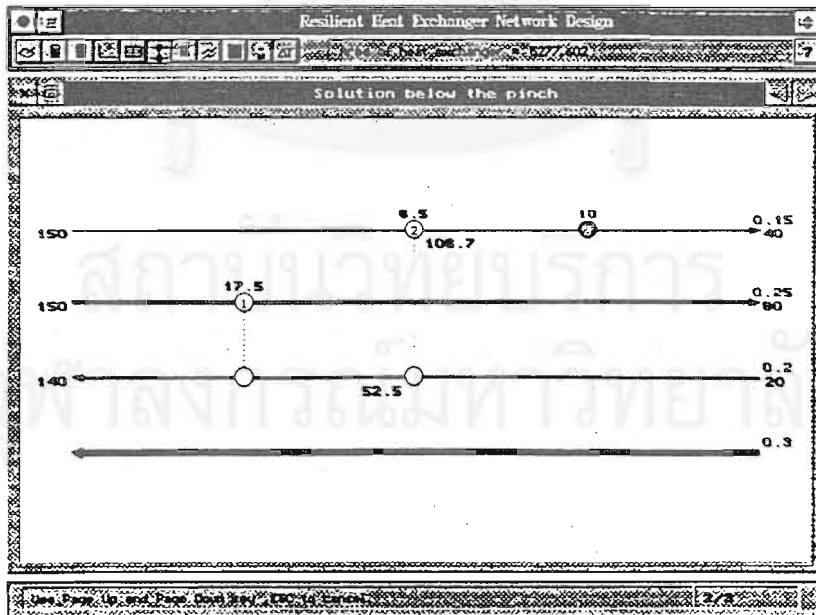
รูปที่ ข.32 ตารางปัญหา



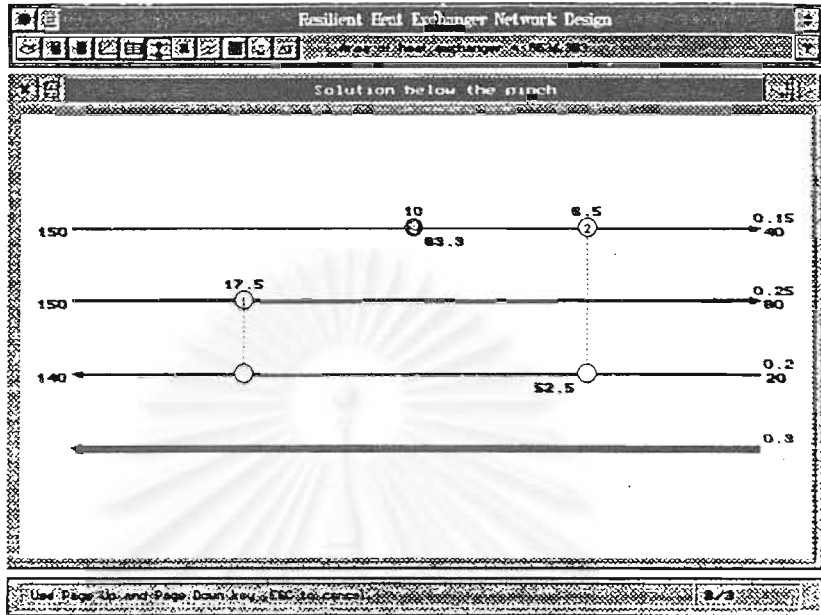
รูปที่ ข.33 แสดงข่ายงานเหนือจุดพินช์



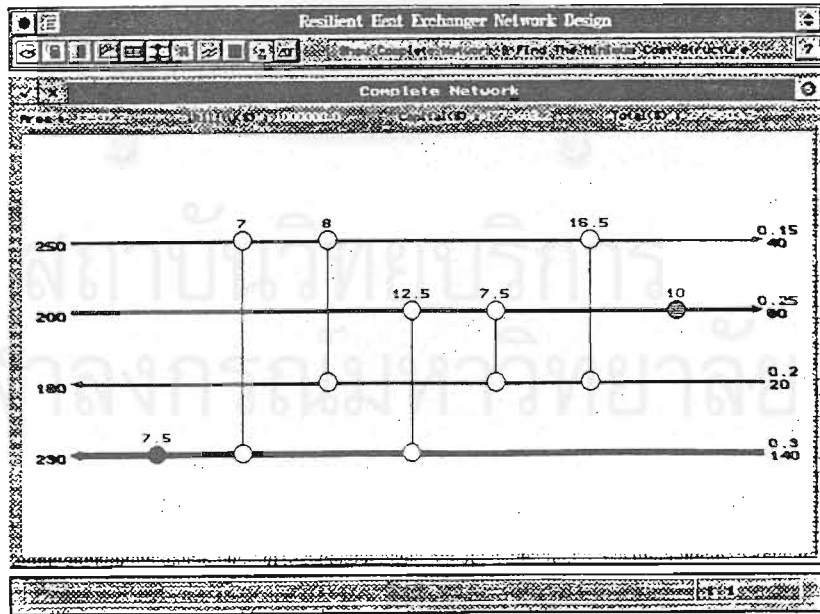
รูปที่ ข.34 แสดงข่ายงานใต้จุดพินช์คำตอบที่ 1/3



รูปที่ ข.35 แสดงข่ายงานใต้จุดพินช์คำตอบที่ 2/3



รูปที่ ข.36 แสดงข่ายงานใต้จุดพินช์คำตอบที่ 3/3



รูปที่ ข.37 แสดงข่ายงานที่สมบูรณ์และประหยัดค่าใช้จ่ายมากที่สุด

จุฬาลงกรณ์มหาวิทยาลัย

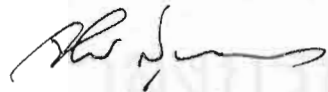
ขอขอบคุณ อาจารย์ ดร. มนต์รี วงศ์ศิริ, นายณัฐพร ทรงศิริ

ที่ได้ร่วมมือเสนอสิ่งประดิษฐ์ โปรแกรมคอมพิวเตอร์สำหรับการออกแบบช่างภาพ
เครื่องแตกเนื้อคนความถี่จนแนบชิดหุ่น

เพื่อแสดงในงานนิทรรศการสิ่งประดิษฐ์ของจุฬาลงกรณ์มหาวิทยาลัย

ระหว่างวันที่ ๑๔ - ๑๘ ธันวาคม พุทธศักราช ๒๕๓๘

ขออำนวยการให้ท่านประสบแต่ความสุขความเจริญตลอดไป



อธิการบดี