การสร้างแบบจำลองเครือข่ายทางรถไฟด้วยไทมด์คัลเลอร์เพทริเน็ต

นายนิติพัฒน์ ทรงวิโรจน์

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมซอฟต์แวร์ ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2561
ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

Railway Network Modeling Using Timed Coloured Petri Nets

Mr. Nitipat Songwiroj

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Science in Software Engineering
Department of Computer Engineering
Faculty of Engineering
Chulalongkorn University
Academic Year 2018

Copyright of Chulalongkorn University

หัวข้อวิทยานิพนธ์	การสร้างแบบจำลองเครือข่ายทางรถไฟด้วยไทมด์คัลเลอร์เพทริเน็ต
โดย	นายนิติพัฒน์ ทรงวิโรจน์
สาขาวิชา	วิศวกรรมซอฟต์แวร์
อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก	รองศาสตราจารย์ ดร.วิวัฒน์ วัฒนาวุฒิ
อาจารย์ที่ปรึกษาวิทยานิพนธ์ร่วม	ผู้ช่วยศาสตราจารย์ ดร.สมศักดิ์ วาณิชอนันต์ชัย

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต

..... คณบดีคณะวิศวกรรมศาสตร์
(รองศาสตราจารย์ ดร.สุพจน์ เตชวรสินสกุล)

คณะกรรมการสอบวิทยานิพนธ์

..... ประธานกรรมการ
(ศาสตราจารย์ ดร.ประภาส จงสถิตย์วัฒนา)

..... อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก
(รองศาสตราจารย์ ดร.วิวัฒน์ วัฒนาวุฒิ)

..... อาจารย์ที่ปรึกษาวิทยานิพนธ์ร่วม
(ผู้ช่วยศาสตราจารย์ ดร.สมศักดิ์ วาณิชอนันต์ชัย)

..... กรรมการ
(รองศาสตราจารย์ ดร.พรศิริ หมื่นไชยศรี)

..... กรรมการภายนอกมหาวิทยาลัย
(ดร.เดชานุชิต กตัญญูทวีทิพย์)

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

นิติพัฒน์ ทรงวิโรจน์ : การสร้างแบบจำลองเครือข่ายทางรถไฟด้วยไทมด์คัลเลอร์เพทริเน็ต. (Railway Network Modeling Using Timed Coloured Petri Nets) อ.ที่ปรึกษาหลัก : รศ. ดร.วิวัฒน์ วัฒนาวุฒิ, อ.ที่ปรึกษาร่วม : ผศ. ดร.สมศักดิ์ วาณิชอนันต์ชัย

การทวนสอบเชิงรูปนัยสามารถนำไปใช้เป็นทางเลือกในการทวนสอบแบบจำลองต้นแบบที่ออกแบบให้ทราบถึงข้อผิดพลาดหรือคุณลักษณะด้านความปลอดภัย แต่อย่างไรก็ตาม การออกแบบแบบจำลองเครือข่ายทางรถไฟขนาดใหญ่ที่มีความซับซ้อนค่อนข้างมากอาจเกิดข้อผิดพลาดขึ้นได้ง่าย

วิทยานิพนธ์นี้จึงได้นำเสนอทางเลือกในการสร้างแบบจำลองเครือข่ายทางรถไฟโดยใช้มอดูลที่ถูกสร้างขึ้นจากแบบจำลองไทมด์คัลเลอร์เพทริเน็ตซึ่งเป็นแบบจำลองเชิงรูปนัยแทนการสร้างแบบจำลองไทมด์คัลเลอร์เพทริเน็ตแบบทั่วไป โดยวิทยานิพนธ์ได้นำเสนอมอดูลมาตรฐานที่แทนส่วนประกอบในเครือข่ายทางรถไฟ คือ สถานีรถไฟ และ รางรถไฟ รองรับประเภทรถไฟโดยสาร 3 ประเภทและการควบคุมการเดินรถไฟโดยใช้ตารางเวลารถไฟที่สามารถกำหนดได้ พร้อมทั้งนำเสนอกฎและเงื่อนไขในการต่อประสานมอดูลเหล่านั้นเพื่อประกอบกันเป็นเครือข่ายทางรถไฟขนาดใหญ่ได้ โดยมีเครื่องมือที่ถูกพัฒนาในวิทยานิพนธ์ช่วยเหลือผู้ใช้ในการสร้างและแปลงเครือข่ายทางรถไฟที่ถูกออกแบบไปเป็นไทมด์คัลเลอร์เพทริเน็ต

ผลลัพธ์การจำลองแสดงผลผ่านโปรแกรมเครื่องมือ ซีพีเอ็น เพื่อตรวจหาความถูกต้องและความปลอดภัยของแบบจำลองเครือข่ายทางรถไฟและตารางรถไฟที่นำเข้ามาตรวจสอบ โดยยกตัวอย่างกรณีศึกษาสำหรับเครือข่ายทางรถไฟจำนวน 8 สถานี พร้อมทำการจำลองเหตุการณ์ความล่าช้าที่เกิดขึ้นที่เครือข่ายทางรถไฟ เพื่อตรวจสอบผลกระทบที่เกิดขึ้น

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

สาขาวิชา วิศวกรรมซอฟต์แวร์ ลายมือชื่อนิสิต .....
ปีการศึกษา 2561 ลายมือชื่อ อ.ที่ปรึกษาหลัก .....
ลายมือชื่อ อ.ที่ปรึกษาร่วม .....

5971013021 : MAJOR SOFTWARE ENGINEERING

KEYWORD: Formal Verification, Petri Nets, Coloured Petri Nets, Timed Coloured Petri Nets, Railway Network, Railway, Modeling

Nitipat Songwiroj : Railway Network Modeling Using Timed Coloured Petri Nets. Advisor: Assoc. Prof. Wiwat Vatanawood, Ph.D. Co-advisor: Asst. Prof. Somsak Vanit-anunchai, Ph.D.

Formal verification is likely an alternative to check safety property of the system model. However, to design an abstract model of huge railway network manually may cause errors because of its complexity.

This thesis proposes an alternative to generate a formal railway network model from the module of Timed Coloured Petri Nets. Instead of constructing the formal model of a railway network model from scratch using CPN Tools, we provide a set of predefined modules of the high-level railway network components, including a train station module, a rail module, etc. A set of mapping rules is provided to generate the high-level railway network in terms of modules, into Timed Coloured Petri Nets automatically by the developed tool.

The resulting formal model is verified using CPN Tools to ensure the correctness, safety, and liveness of the formal railway network model.

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

Field of Study: Software Engineering Student's Signature .....

Academic Year: 2018 Advisor's Signature .....

Co-advisor's Signature .....

กิตติกรรมประกาศ

งานวิจัยนี้สำเร็จลุล่วงได้ด้วยดีด้วยความช่วยเหลือและได้รับคำแนะนำที่ดีจาก รศ.ดร.วิวัฒน์ วัฒนาวุฒิ อาจารย์ที่ปรึกษาและ ผศ.ดร.สมศักดิ์ วาณิชอนันต์ชัย อาจารย์ที่ปรึกษาร่วมของข้าพเจ้า อีกทั้งยังได้รับความรู้ ข้อมูลที่จำเป็นที่เกี่ยวกับระบบเครือข่ายทางรถไฟที่ใช้ในประเทศไทยจาก การรถไฟแห่งประเทศไทย

ขอบคุณเพื่อนๆ สาขาวิศวกรรมซอฟแวร์และครอบครัวในการสนับสนุนช่วยเหลือในการทำวิจัยในด้านต่างๆ

ขอบคุณประธานและคณะกรรมการสอบโครงร่างวิทยานิพนธ์สำหรับคำแนะนำที่เป็นประโยชน์ในด้านต่างๆของงานวิจัย

นิติพัฒน์ ทรงวิโรจน์

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

สารบัญ

หน้า

..... ค

บทคัดย่อภาษาไทย ..... ค

..... ง

บทคัดย่อภาษาอังกฤษ ..... ง

กิตติกรรมประกาศ ..... จ

สารบัญ ..... ฉ

สารบัญตาราง ..... ญ

สารบัญรูปภาพ ..... ฎ

บทที่ 1 บทนำ 1

1.1 ที่มาและความสำคัญของปัญหา 1

1.2 วัตถุประสงค์ 2

1.3 ขอบเขตการดำเนินงาน 3

1.4 ประโยชน์ที่คาดว่าจะได้รับ 3

1.5 ขั้นตอนการดำเนินงาน 4

1.6 บทความที่ตีพิมพ์จากงานวิจัย 5

บทที่ 2 ทฤษฎีที่เกี่ยวข้อง 6

2.1 ภาษาเอสเอ็มแอล (Standard Metalanguage หรือ SML) 6

2.2 ภาษาโปรแกรมซีพีเอ็นเอ็มแอล (CPN ML) 7

2.3 ไทมด์ คัลเลอร์ เพทริเน็ต แบบมีลำดับชั้น (Hierarchical Timed Coloured Petrinet) 8

2.4 งานวิจัยที่เกี่ยวข้อง 16

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

2.4.1 งานวิจัยของ Christine Choppy, Laure Petrucci, and Gianna Reggio หัวข้อ A Modelling Approach With Coloured Petri Nets. ในปี 2008 16
2.4.2 งานวิจัยของ Davide Giglio and Nicola Sacco หัวข้อ A Petri net model for analysis, optimisation, and control of railway networks and train schedules. ในปี 2016 18
2.4.3 งานวิจัยของ T. Sakaguchi and N. Tomii หัวข้อ A train traffic model based on coloured Petri Nets and its application to a train schedule planning system ในปี 1996 20
2.4.4 งานวิจัยของ Deesukying, J. and Vatanawood, W. หัวข้อ Generating of business rules for Coloured Petri Nets. ในปี 2016 21
บทที่ 3 การออกแบบและการสร้าง CPN 24
3.1 ข้อมูลประกอบการสร้างเครือข่ายทางรถไฟ 25
3.2 การออกแบบโครงสร้างเครือข่ายทางรถไฟ 26
3.2.1 กำหนดข้อมูลสำหรับสร้างเครือข่ายทางรถไฟ 26
3.2.2 ออกแบบมอดูลมาตรฐานสำหรับส่วนประกอบในเครือข่ายทางรถไฟ 27
3.2.3 กำหนดกฎการต่อประสานด้วยวิธีการซ้อนทับ (Superimposing Rules) 34
3.2.4 กำหนดกฎการสร้างที่พักรถไฟก่อนเข้าสถานีต้นทาง 35
3.2.5 กำหนดกฎการเพิ่มเติมสำหรับการสร้างแบบจำลองเครือข่ายทางรถไฟ 38
3.3 การกำหนดข้อมูลและโครงสร้างไฟล์เอ็กซ์เอ็มแอลสำหรับจัดเก็บเครือข่ายทางรถไฟ 42
3.3.1 ส่วนบันทึกแบบจำลองเครือข่ายทางรถไฟที่ถูกออกแบบ 43
3.3.2 ส่วนบันทึกตารางเวลารถไฟเข้าออกตามแต่ละสถานี 46
3.3.3 ส่วนบันทึกค่าพารามิเตอร์ของขบวนรถไฟ 47
3.4 กฎการแปลงเครือข่ายทางรถไฟไปเป็น CPN 48
3.4.1 เพลส 48
3.4.2 ทรานซิชัน 48

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

3.4.3 อาร์คหรือเส้นเชื่อม 51
3.4.3 ข้อมูลตารางเวลารถไฟ 52
3.4.4 ฟังก์ชันที่ถูกเรียกใช้งาน 53
3.5 เครื่องมือช่วยเหลือในการแปลงเครือข่ายทางรถไฟไปเป็นไทมด์คัลเลอร์เพทริเน็ต 55
3.5.1 คุณสมบัติของเครื่องมือ 56
3.5.2 ข้อจำกัดเครื่องมือ 57
3.5.3 แผนภาพยูสเคส (Use Case Diagram) 58
3.5.4 แผนภาพคลาส (Class Diagram) 61
3.5.5 แผนภาพลำดับการทำงาน (Sequence Diagrams) 61
3.6 อธิบายตัวอย่างแบบจำลองและตัวอย่างเอกสารนำออกที่ถูกบันทึกจากแบบจำลองไทมด์คัลเลอร์เพทริเน็ต 62
3.6.1 ตัวอย่างแบบจำลองไทมด์คัลเลอร์เพทริเน็ต 62
3.6.2 ตัวอย่างเอกสารนำออกที่ถูกบันทึกจากแบบจำลองไทมด์คัลเลอร์เพทริเน็ต 64
บทที่ 4 ผลการจำลองแบบจำลองเครือข่ายทางรถไฟ 69
4.1 ข้อมูลที่ใช้ในการสร้างแบบจำลองเครือข่ายทางรถไฟ 69
4.2 แบบจำลองเครือข่ายทางรถไฟ 71
4.3 ผลการจำลองแบบจำลองเครือข่ายทางรถไฟ 76
4.4 ผลการทดลองหลังปรับแต่งแบบจำลองเครือข่ายทางรถไฟ 76
4.5 สรุปผลการทดลอง 79
บทที่ 5 สรุปผลและข้อเสนอแนะ 80
5.1 สรุปผลการวิจัย 80
5.2 ข้อเสนอแนะ 82
บรรณานุกรม 83
ประวัติผู้เขียน 86

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

สารบัญตาราง

หน้า

ตารางที่ 2.1 ตารางส่วนประกอบและคำอธิบายของไทมด์คัลเลอร์เพทริเน็ต 10
ตารางที่ 2.2 ตารางคีอิลิเมนต์ที่สำคัญในเอกสารนามสกุล .CPN 14
ตารางที่ 2.3 ตารางสรุปเปรียบเทียบงานวิจัยต่างๆ 23
ตารางที่ 3.1 ตารางส่วนประกอบมอดูลสถานีและมอดูลรางรถไฟ 33
ตารางที่ 3.2 ตารางสัญลักษณ์มอดูล 43
ตารางที่ 3.3 ตารางอิลิเมนต์สำหรับหนึ่งมอดูล 44
ตารางที่ 3.4 ตารางอิลิเมนต์สำหรับตารางรถไฟแต่ละขบวน 46
ตารางที่ 3.5 ตารางอิลิเมนต์สำหรับจัดเก็บค่าพารามิเตอร์ของรถไฟ 47
ตารางที่ 3.6 ตารางอธิบายอิลิเมนต์ต่างๆที่สำคัญของเพลส 49
ตารางที่ 3.7 ตารางอธิบายอิลิเมนต์ต่างๆที่สำคัญของทรานซิชัน 50
ตารางที่ 3.8 ตารางอธิบายอิลิเมนต์ต่างๆที่สำคัญของอาร์ค 51
ตารางที่ 3.9 ตารางยูสเคสการเพิ่มมอดูลสถานีในแบบจำลอง (Add A Station Model) 59
ตารางที่ 3.10 ตารางยูสเคสการสร้างแฟ้มเอกสารนำออก CPN (Generate CPN File) 60
ตารางที่ 4.1 ตารางเวลารถไฟขบวนเลขที่ 259 70
ตารางที่ 4.2 ตารางเวลารถไฟหลังการแปลงเริ่มเวลานาทีที่ 0 71
ตารางที่ 4.3 ตารางเวลารถไฟใหม่ของรถไฟขบวนที่ 254 77
ตารางที่ 4.4 ตารางเวลารถไฟใหม่ขบวนที่ 254 (ล่าช้า) 78

Chulalongkorn University

สารบัญรูปภาพ

หน้า

รูปที่ 2.1 โครงสร้างตัวแปรประเภทบูลีน [2] 6

รูปที่ 2.2 ตารางค่าความสัมพันธ์และเครื่องหมายดำเนินการ [3] 7

รูปที่ 2.3 รูปโครงสร้างตัวแปรประเภทโครงสร้างคิว (Queue) [1] 7

รูปที่ 2.4 แสดงส่วนประกอบของฟังก์ชัน 8

รูปที่ 2.5 ภาพแสดงก่อนการยิง 11

รูปที่ 2.6 ภาพแสดงหลังการยิง 11

รูปที่ 2.7 ตัวอย่างแบบจำลองเพทริเน็ตทั่วไป, แบบจำลองไทม์เพทริเน็ต, และแบบจำลองไทม์คัลเลอร์เพทริเน็ต [5] 13

รูปที่ 2.8 ตัวอย่างการประกาศคัลเลอร์เซตแบบพื้นฐานเทียบโครงสร้างเอ็กซ์เอ็มแอลไฟล์ .cpn [5] 14

รูปที่ 2.9 มุมมองแบบจำลองไฮบริดของตัวอย่างการให้บริการในโรงพยาบาล [7] 15

รูปที่ 2.10 แบบจำลองขั้นสูงของระบบรางรถไฟของเล่น [8] 16

รูปที่ 2.11 แบบจำลองส่วนย่อยของรางรถไฟ (a) และ ส่วนย่อยจุดสลับราง (b) [8] 17

รูปที่ 2.12 แบบจำลองส่วนย่อยใหม่เพื่อแก้ปัญหาการติดตายในระบบราง [8] 17

รูปที่ 2.13 แบบจำลองเครือข่ายทางรถไฟสำหรับระบบรางคู่จำนวน 7 สถานี 8 เส้นทางรถไฟ [9].. 18

รูปที่ 2.14 รูปแสดงตารางเวลาเข้าออกของรถไฟ 11 ขบวนในแต่ละสถานี [9] 18

รูปที่ 2.15 ตัวอย่างไทมด์เพทริเน็ตของส่วนประกอบจริงของหน่วยรางรถไฟ (Block Section) 3 หน่วยที่เรียงต่อกัน [9] 19

รูปที่ 2.16 แบบจำลองรางรถไฟที่กำหนดที่ประกอบไปด้วยส่วนประกอบจริงและส่วนประกอบที่เป็นตรรกะ [9] 20

รูปที่ 2.17 ตัวอย่างการจัดการจราจรเพื่อใช้งานรางรถไฟ [10] 21

รูปที่ 2.18 ขั้นตอนภาพรวมงานวิจัย [11] 22

รูปที่ 3.1 แนวคิดในการทำวิจัยในภาพรวม 24

รูปที่ 3.2 แผนภาพแสดงการเชื่อมต่อสถานีต่างๆ 27
รูปที่ 3.3 รูปแสดงตารางเวลารถไฟจำลอง 27
รูปที่ 3.4 มอดูลสถานีที่สร้างมาจากไทมด์คัลเลอร์เพทริเน็ต 27
รูปที่ 3.5 มอดูลสถานีที่มีรางขาเข้า 1 ราง และ รางขาออก 2 ราง 30
รูปที่ 3.6 มอดูลรางที่สร้างมาจากไทมด์คัลเลอร์เพทริเน็ต 30
รูปที่ 3.7 แสดงมอดูลรางรถไฟประเภทรางคู่ 32
รูปที่ 3.8 ตัวอย่างการประสานมอดูลสถานีและมอดูลรางรถไฟด้วยวิธีการซ้อนทับ 34
รูปที่ 3.9 ตัวอย่างการประสานมอดูลรางรถไฟและรางรถไฟด้วยวิธีการซ้อนทับ 35
รูปที่ 3.10 การเพิ่มเพลสสำหรับสถานีต้นทางขาเข้า 36
รูปที่ 3.11 การเพิ่มเพลสสำหรับสถานีต้นทางขาออก 37
รูปที่ 3.12 การเพิ่มส่วนประกอบไทมด์คัลเลอร์เพทริเน็ตสำหรับสถานีชุมทาง 38
รูปที่ 3.13 แสดงการประกอบมอดูลสถานีและรางเดี่ยว 39
รูปที่ 3.14 แสดงการต่อประสาน 2 สถานีด้วยรางคู่ 40
รูปที่ 3.15 ตัวอย่างการประสานมอดูลรางรถไฟเดี่ยวและรางรถไฟคู่ (จุดแยกราง) 41
รูปที่ 3.16 ตัวอย่างการประสานมอดูลรางรถไฟคู่และรางเดี่ยว (จุดรวมราง) 42
รูปที่ 3.17 ตัวอย่างหลังการแปลงแบบจำลองเครือข่ายทางรถไฟเป็นไทมด์คัลเลอร์เพทริเน็ต 42
รูปที่ 3.18 มอดูลสถานีในรูปแบบเอ็กซ์เอ็มแอล 45
รูปที่ 3.19 มอดูลรางรถไฟในรูปแบบเอ็กซ์เอ็มแอล 45
รูปที่ 3.20 แสดงตัวอย่างเอ็กซ์เอ็มสำหรับเก็บค่าตารางเวลารถไฟ 46
รูปที่ 3.21 แสดงตัวอย่างค่าพารามิเตอร์ความเร็วของรถไฟประเภทต่างๆ 47
รูปที่ 3.22 แสดงอิลิเมนต์ที่จัดเก็บข้อมูลที่สำคัญของเพลส 48
รูปที่ 3.23 แสดงอิลิเมนต์ที่จัดเก็บข้อมูลที่สำคัญของทรานซิชัน 48
รูปที่ 3.24 แสดงอิลิเมนต์ที่จัดเก็บข้อมูลที่สำคัญของอาร์ค 51
รูปที่ 3.25 แสดงตัวอย่างการจัดเก็บข้อมูลตารางรายการรถไฟที่ถูกกำหนดให้ใช้เส้นทางรอง 52

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

รูปที่ 3.26 แสดงตัวอย่างการจัดเก็บข้อมูลตารางเวลารถไฟขบวนที่ 99 ในเอกสาร .cpn 52
รูปที่ 3.27 แสดงตัวอย่างตารางรถไฟรวมของทุกขบวนที่ถูกจัดเก็บใน ttReList 53
รูปที่ 3.28 แสดงตัวอย่างตารางเวลารถไฟเข้าที่สถานีต้นทางที่ถูกจัดเก็บใน inTrList 53
รูปที่ 3.29 ฟังก์ชัน traveltime 53
รูปที่ 3.30 ฟังก์ชัน isHeadOut 53
รูปที่ 3.31 ฟังก์ชัน isHeadIn 54
รูปที่ 3.32 ฟังก์ชัน isStop 54
รูปที่ 3.33 ฟังก์ชัน isDepartureTime 54
รูปที่ 3.34 ฟังก์ชัน trainInSch 55
รูปที่ 3.35 ฟังก์ชัน trainOutSch 55
รูปที่ 3.36 เครื่องมือที่ถูกพัฒนาในงานวิจัย 56
รูปที่ 3.37 แผนภาพยูสเคสของเครื่องมือ Railway Network Modeling Tool 58
รูปที่ 3.38 แผนภาพคลาสของเครื่องมือ Railway Network Modeling Tool 61
รูปที่ 3.39 แผนภาพลำดับการเพิ่มมอดูลสถานีในขั้นตอนการออกแบบ 61
รูปที่ 3.40 แผนภาพลำดับการแปลงเครือข่ายทางรถไฟเป็นไทมด์คัลเลอร์เพทริเน็ต 62
รูปที่ 3.41 แบบจำลองไทมด์คัลเลอร์เพทริเน็ตที่ประกอบด้วย 2 มอดูลสถานีและ 1 มอดูลรางรถไฟ 63
รูปที่ 3.42 แสดงแบบจำลองไทมด์คัลเลอร์เพทริเน็ตสำหรับมอดูลสถานี 65
รูปที่ 3.43 แสดงโครงสร้างเอกสารนำออกที่ถูกสร้างมาจากมอดูลสถานี (ส่วนที่ 1) 65
รูปที่ 3.44 แสดงโครงสร้างเอกสารนำออกที่ถูกสร้างมาจากมอดูลสถานี (ส่วนที่ 2) 66
รูปที่ 3.45 แบบจำลองไทมด์คัลเลอร์เพทริเน็ตสำหรับมอดูลรางรถไฟมาตรฐาน 67
รูปที่ 3.46 แสดงโครงสร้างเอกสารนำออกที่ถูกสร้างมาจากมอดูลรางรถไฟ 68
รูปที่ 4.1 แสดงรูปตัวอย่างเครือข่ายทางรถไฟจากข้อมูลในหัวข้อ 4.1 72
รูปที่ 4.2 แบบจำลองเครือข่ายทางรถไฟที่สร้างมาจากสถานีจำนวน 8 สถานี 72

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

รูปที่ 4.3 รูปแสดงตารางเวลารถไฟโดยเริ่มต้นที่เวลา 14.00 เป็นนาทีที่ 0 ในแบบจำลอง 72

รูปที่ 4.4 ข้อมูลความเร็วรถไฟประเภทต่างๆที่จะนำมาใช้ในการสร้างแบบจำลอง 73

รูปที่ 4.5 แบบจำลองไทมด์คัลเลอร์เพทริเน็ตจากแบบจำลองเครือข่ายทางรถไฟ 8 สถานี (ส่วนที่ 1) 74

รูปที่ 4.6 แบบจำลองไทมด์คัลเลอร์เพทริเน็ตจากแบบจำลองเครือข่ายทางรถไฟ 8 สถานี (ส่วนที่ 2) 75

รูปที่ 4.7 Time Graph จากแบบจำลองและจากตารางเวลา..... 76

รูปที่ 4.8 Time Graph แสดงผลกระทบหลังปรับเวลาตารางรถไฟขบวนที่ 254 77

รูปที่ 4.9 การเกิด Deadlock ในแบบจำลองไทมด์เพทริเน็ต 79

รูปที่ 4.10 Time Graph การเกิด Deadlock 79

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

บทที่ 1
บทนำ

1.1 ที่มาและความสำคัญของปัญหา

ในปัจจุบันการโดยสารด้วยระบบขนส่งประเภทรางได้กำลังมีบทบาทมากขึ้นในสังคมการใช้ชีวิตของทั้งคนในเมืองใหญ่และต่างจังหวัดของประเทศไทย จะเห็นได้จากโครงการภาครัฐที่ใช้งบประมาณหลายแสนล้านบาทเพื่อพัฒนาระบบรางทั้งในกรุงเทพมหานครและปริมณฑล และตามหัวเมืองต่างๆ ที่ได้เริ่มก่อสร้างขึ้น โดยจะแล้วเสร็จในอนาคตอันใกล้ โดยโครงการเหล่านี้จะทำให้การจราจรในระบบขนส่งมวลชนประเภทรางมีความหนาแน่นมากขึ้น ประชาชนอาจให้ความสนใจในการใช้บริการเพิ่มมากขึ้นเพราะระบบขนส่งประเภทรางมีค่าใช้จ่ายต่ำกว่าประเภทอื่น อีกทั้งยังมีความปลอดภัยสูงกว่า ส่งผลให้ต้องมีการบริหารจัดการระบบต่างๆ ที่ต้องดีขึ้นกว่าเก่า เพื่อลดและป้องกันปัญหาความปลอดภัยในชีวิตและทรัพย์สินของผู้โดยสาร อีกทั้งยังส่งผลต่อความเชื่อมั่นของผู้ใช้งาน เมื่อรถไฟเกิดเหตุความล่าช้าซึ่งอาจสืบเนื่องจากปัญหาเหตุสุดวิสัย แต่หากมีเครื่องมือที่ช่วยเหลือผู้ใช้งานระบบให้สามารถเรียนรู้และแก้ไขปัญหาได้อย่างรวดเร็วก็จะช่วยลดผลกระทบที่อาจเกิดขึ้นได้

โดยปัจจุบัน มีโปรแกรมต่างๆ ที่สามารถช่วยเหลือผู้ใช้งานให้สามารถสร้างแบบจำลองเครือข่ายทางรถไฟที่นิยมใช้กันในท้องตลาด เช่น OpenTrack RailSys เป็นต้น โปรแกรมเหล่านี้สามารถสร้างแบบจำลองระบบราง อุปกรณ์และส่วนประกอบต่างๆ หรือสร้างตารางเวลาเข้าออกของรถไฟตามสถานีได้ ทำให้ผู้ใช้งานโปรแกรมสามารถทดสอบการใช้งานระบบโดยการสร้างแบบจำลองระบบ เพื่อให้มั่นใจถึงความปลอดภัยและถูกต้องของระบบในระดับหนึ่งก่อนนำระบบไปใช้ทดสอบจริง อีกทั้งยังสามารถช่วยเหลือผู้ใช้งานในการออกแบบและตรวจสอบตารางเวลาการเดินรถได้ แต่โปรแกรมเหล่านั้นมีค่าใช้จ่ายที่สูง อีกทั้งการจำลองระบบเป็นเพียงการจำลองระบบรางรถไฟและทดสอบการใช้งานเท่านั้น มิใช่เพื่อจุดประสงค์ตรวจสอบหรือวิเคราะห์เหตุผิดพลาดที่อาจจะเกิดขึ้นในระบบ

ภาษาเชิงรูปนัย (Formal Language) เป็นที่รู้จักกันดีในการใช้งานสำหรับการตรวจสอบความปลอดภัย ความถูกต้องและข้อผิดพลาดของระบบที่มีโอกาสเกิดขึ้นได้ การใช้งานภาษาเชิงรูปนัยในระบบรางนี้สามารถนำไปประยุกต์ใช้สำหรับการวิเคราะห์และพิสูจน์คุณสมบัติด้านต่างๆ ของแบบจำลองเครือข่ายทางรถไฟและส่วนควบคุม ซึ่งจะช่วยเหลือผู้สร้างแบบจำลองให้สามารถตรวจสอบปัญหาต่างๆ ได้ดีและครบถ้วนกว่าการสร้างแบบจำลองโดยใช้โปรแกรมทั่วไป เมื่อเครือข่ายทางรถไฟที่มีความซับซ้อน

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ยิ่งขึ้นการสร้างแบบจำลองทั่วไปอาจไม่สามารถตรวจสอบเหตุการณ์หรือสถานการณ์บางอย่างได้ครบถ้วนสมบูรณ์ ภาษาเชิงรูปนัยจึงมีคุณสมบัติในการใช้งานที่ครบถ้วนกว่าสำหรับการสร้างแบบจำลองใด

แต่ปัญหาในการใช้งานภาษาเชิงรูปนัยแบบดั้งเดิม คือ ความยุ่งยากในการใช้งาน เนื่องจากเป็นภาษาที่คล้ายคลึงกับการเขียนโปรแกรม ผู้ใช้งานต้องศึกษาถึงรูปแบบการใช้งาน เมื่อทำการวิเคราะห์หรือทำความเข้าใจกับรูปแบบของภาษานี้ ทำให้คนทั่วไปสามารถเข้าถึงได้ยาก งานวิจัยนี้ได้ให้ความสนใจกับเพทริเน็ต (Petri Nets) ซึ่งมีความสามารถจำลองภาพแบบจำลองให้ผู้ใช้สามารถเข้าใจได้ง่ายกว่าเพียงการเขียนภาษาเชิงรูปนัยแบบดั้งเดิม โดยได้นำเอาคัลเลอร์เพทริเน็ต (Coloured Petri Nets) และ โปรแกรมเครื่องมือซีพีเอ็นหรือ ซีพีเอ็นทูลส์ (CPN Tools) มาใช้ในการสร้างแบบจำลองและพิสูจน์คุณสมบัติของแบบจำลองต้นแบบคัลเลอร์เพทริเน็ต โดยจะเพิ่มเติมคุณสมบัติในด้านของเวลาในงานวิจัยนี้ โปรแกรมซีพีเอ็นทูลส์ยังรองรับภาษาเอ็มแอล (ML) ใช้สำหรับเขียนชุดคำสั่ง (Script) เพื่อเพิ่มความสามารถให้แก่แบบจำลอง เพื่อใช้แก้ปัญหาต่างๆ ที่ซับซ้อนได้ ซึ่งจะเรียกว่าภาษาโปรแกรมซีพีเอ็นหรือซีพีเอ็น เอ็มแอล (CPN ML) ในการนำมาประกอบการใช้งานในแบบจำลอง

งานวิจัยนี้จะพัฒนาเครื่องมือสำหรับช่วยเหลือผู้ใช้งานออกแบบแบบจำลองเครือข่ายทางรถไฟ โดยสามารถบันทึกแบบจำลองเครือข่ายทางรถไฟในรูปแบบเอ็กซ์เอ็มแอลที่มีรูปแบบเป็นไปตามที่เครื่องมือกำหนด และสามารถแปลงแบบจำลองดังกล่าวไปเป็นไทมล์คัลเลอร์เพทริเน็ต ผู้ใช้งานสามารถกำหนดค่าตัวแปรสำคัญที่เกี่ยวข้อง คือ ชื่อสถานี ความยาวของรางรถไฟ ประเภทรถไฟและความเร็วของรถไฟประเภทต่างๆ เวลาเข้าและออกจากสถานีของขบวนรถไฟ เพื่อนำแบบจำลองเครือข่ายทางรถไฟไปทดสอบด้วยโปรแกรมเครื่องมือซีพีเอ็น ในการตรวจสอบความปลอดภัย (Safety) และตรวจสอบการปล่อยรถไฟไปตามเส้นทางที่กำหนด รวมถึงการตรวจสอบเวลาเข้าออกรถไฟตามสถานีต่างๆ จากแบบจำลองว่าเป็นไปตามที่ผู้ใช้งานกำหนดในตารางเวลาการเดินรถไฟหรือไม่ ผู้ใช้งานสามารถปรับแก้ตารางเวลาการเดินรถไฟให้สอดคล้องกับผลลัพธ์จากการทดสอบด้วยไทมด์คัลเลอร์เพทริเน็ต เพื่อนำไปประกอบในการใช้งานจริงได้ต่อไป

1.2 วัตถุประสงค์

1) ออกแบบแบบจำลองเครือข่ายทางรถไฟด้วยไทมด์คัลเลอร์เพทริเน็ต
2) พัฒนาเครื่องมือสำหรับสร้างแบบจำลองเครือข่ายทางรถไฟและแปลงเป็นไทมด์คัลเลอร์เพทริเน็ต

1.3 ขอบเขตการดำเนินงาน

1) มอดูลที่ออกแบบจะครอบคลุม 2 ประเภท
    a. มอดูลสถานี ที่สามารถระบุจำนวนรถไฟที่รองรับได้
    b. มอดูลรางรถไฟ โดยแบ่งเป็น 2 ประเภท คือ รางเดียว และรางคู่เท่านั้น สามารถระบุความยาวของรางได้
2) เครือข่ายทางรถไฟรองรับประเภทรถไฟ 3 ประเภท คือ รถด่วนพิศษ รถเร็ว และรถธรรมดา
3) การออกแบบเครือข่ายทางรถไฟโดยใช้สัญลักษณ์จากมอดูลที่ถูกกำหนดไว้แล้วเท่านั้น
4) เครื่องมือสร้างแบบจำลองเครือข่ายทางรถไฟ มีขีดความสามารถดังต่อไปนี้
    a) สามารถนำมอดูลมาประกอบกันเป็นเครือข่ายทางรถไฟได้
    b) สามารถนำเข้าข้อมูลตารางเวลาเข้าออกสถานีของรถไฟแต่ละขบวนโดยตั้งค่าจากหน้าจอได้
    c) สามารถบันทึกแบบจำลองเครือข่ายทางรถไฟที่ผู้ใช้งานสร้าง โดยแปลงเป็นข้อมูลนำออกที่มีรูปแบบภาษาเอ็กซ์เอ็มแอลที่โปรแกรมกำหนด
    d) สามารถนำเข้าข้อมูลแบบจำลองเครือข่ายทางรถไฟที่อยู่ในรูปแบบภาษาเอ็กซ์เอ็มแอลที่มีรูปแบบเป็นไปตามที่แอปพลิเคชันกำหนดได้
    e) สามารถแปลงแบบจำลองเครือข่ายทางรถไฟไปเป็นไทมด์คัลเลอร์เพทริเน็ตโดยอ้างอิงจากกฎการแปลงที่ถูกสร้างไว้
    f) สามารถสร้างข้อมูลนำออกในรูปแบบภาษาเอ็กซ์เอ็มแอลที่สามารถนำไปใช้งานกับโปรแกรมเครื่องมือซีพีเอ็น ได้
    g) มอดูลสำหรับระบบรถไฟถูกกำหนดไว้ ไม่สามารถเปลี่ยนแปลงหรือเพิ่มเติมได้
5) ผลลัพธ์ของการทดสอบแบบจำลองเปลี่ยนแปลงได้ตามการยิงของโทเคนในแบบจำลองคัลเลอร์เพทริเน็ต โดยงานวิจัยมีตัวอย่างผลลัพธ์ของการจำลองการทำงานของแบบจำลอง
6) ข้อมูลของระบบรางรถไฟและรถไฟจะอ้างอิงมาจากการรถไฟแห่งประเทศไทยเท่านั้น
7) ใช้กรณีทดสอบเส้นทางระบบรางรถไฟระหว่างจังหวัด ไม่เกิน 8 สถานี

1.4 ประโยชน์ที่คาดว่าจะได้รับ

1) ได้เครื่องมือสนับสนุนการออกแบบแบบจำลองเครือข่ายทางรถไฟ
2) ได้ทางเลือกในการออกแบบและสร้างแบบจำลองเครือข่ายทางรถไฟด้วยไทมด์คัลเลอร์เพทริเน็ต

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

1.5 ขั้นตอนการดำเนินงาน

1) ศึกษาการทำงานของเกี่ยวกับระบบรางรถไฟ ข้อจำกัด และกฎพื้นฐานในการสร้างตารางการเดินรถของการรถไฟแห่งประเทศไทย
2) ศึกษาแนวทางในการออกแบบและสร้างแบบจำลองไทม์คัลเลอร์เพทริเน็ต
3) จัดทำข้อเสนอโครงการ
4) คัดเลือก รูปแบบสถานี ระบบราง ประเภทรถไฟ ตารางสำหรับกำหนดเวลารถไฟเข้าออกแต่ละสถานี และข้อมูลสถานี ที่จะนำมาสร้างไทมด์คัลเลอร์เพทริเน็ตเครือข่ายทางรถไฟ
5) สร้างแบบจำลองเครือข่ายทางรถไฟด้วยไทมด์คัลเลอร์เพทริเน็ต
6) ออกแบบมอดูลสำหรับเครือข่ายทางรถไฟจากแบบจำลองไทมด์คัลเลอร์เพทริเน็ต
7) สร้างกฎพื้นฐานสำหรับการการแปลงมอดูลสำหรับเครือข่ายทางรถไฟไปเป็นแบบจำลองไทมด์คัลเลอร์เพทริเน็ตพร้อมกฎการประสานมอดูลต่างๆ เพื่อสร้างแบบจำลองเครือข่ายทางรถไฟ
8) สร้างเครื่องมือสำหรับช่วยเหลือผู้ใช้งานในสามารถสร้างแบบจำลองเครือข่ายทางรถไฟและแปลงแบบจำลองไปเป็นไทมด์คัลเลอร์เพทริเน็ต
9) สร้างแบบจำลองเครือข่ายทางรถไฟจากมอดูลที่กำหนดพร้อมกำหนดค่าตารางเวลารถไฟสำหรับแต่ละสถานีและทดสอบการแปลงแบบจำลองไปเป็นไทมด์คัลเลอร์เพทริเน็ต และวิเคราะห์ผลลัพธ์
10) ปรับแบบจำลองเครือข่ายทางรถไฟและตารางเวลารถไฟโดยใช้ข้อมูลนำเข้าข้อมูลรถไฟระหว่างจังหวัดจำนวน 8 สถานี จากการรถไฟแห่งประเทศไทย พร้อมบันทึกผลลัพธ์
11) สรุปผลการทดลองจากการปรับแบบจำลอง และภาพรวมงานวิจัย
12) จัดทำบทความสำหรับตีพิมพ์
13) จัดทำรูปเล่มวิทยานิพนธ์

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

1.6 บทความที่ตีพิมพ์จากงานวิจัย

ส่วนหนึ่งจากงานวิจัยนี้ได้รับการตีพิมพ์ในรูปแบบบทความวิชาการ ดังนี้

“Railway Network Modeling Using Building Block Of Timed Coloured Petri Nets” โดยนิติพัฒน์ ทรงวิโรจน์, วิวัฒน์ วัฒนาวุฒิ, และ สมศักดิ์ วาณิชอนันต์ชัย ในงานประชุมวิชาการ 2018 4th IEEE International Conference on Computer and Communications (ICCC) จัดขึ้นโดย International Conference on Computer and Communications (ICCC) ระหว่างวันที่ 7-10 ธันวาคม 2561 ณ เมืองเฉิงตู ประเทศสาธารณรัฐประชาชนจีน

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

บทที่ 2
ทฤษฎีที่เกี่ยวข้อง

2.1 ภาษาเอสเอ็มแอล (Standard Metalanguage หรือ SML)

ภาษาเอสเอ็มแอล เป็นภาษาฟังก์ชัน (Functional Programming Language) ประเภทไทป์เซฟ (Type-Safe) กล่าวคือ มีความสามารถในตรวจสอบความถูกต้องในขั้นตอนการประมวลผล (Compile) ทำให้ทราบถึงข้อผิดพลาด (Error) ของโปรแกรม โดยยังเป็นภาษาที่รวมเอานวัตกรรมทางด้านแนวความคิดการออกแบบต่างๆเข้าด้วยกัน ทำให้มีความสามารถค่อนข้างหลากหลายและมีประสิทธิภาพในการใช้งาน มีการจัดการในส่วนของการเขียนโปรแกรมได้ดี ทำให้มีการใช้ทรัพยากรอย่างคุ้มค่าในด้านการเขียนโปรแกรม มีความสามารถในการจัดการข้อผิดพลาด (Error Handling) ได้หลากหลาย อีกทั้งยังรอบรับการทำงานสำหรับโปรแกรมขนาดใหญ่ด้วย และด้วยความสามารถในด้านการโอนย้ายไปยังโปรแกรมอื่นๆ ทำให้การใช้งานภาษานี้เป็นที่นิยมในการนำไปใช้งานเป็นมาตรฐาน [1] โดยทั่วไปแล้ว ภาษาเอสเอ็มแอลมี 2 ส่วน

1) ส่วนภาษาคอร์ (Core Language) เป็นส่วนที่เป็นพื้นฐานของภาษาที่ถูกกำหนดมาเป็นแกนหลัก ที่ไม่มีความซับซ้อนใด ไม่ว่าจะเป็นประเภทตัวแปร (Type) หรือเครื่องหมายดำเนินการต่างๆ (Operand) เช่น ประเภทตัวแปรบูลีน(Boolean) ดังตัวอย่างใน รูปที่ 2.1 และตารางในรูป 2.2

```
datatype bool = false | true
val not : bool -> bool
val toString : bool -> string
val scan       : (char, 'a) StringCvt.reader
                   -> (bool, 'a) StringCvt.reader
val fromString : string -> bool option
```

รูปที่ 2.1 โครงสร้างตัวแปรประเภทบูลีน [2]

Basic operators on the Booleans

op	:	*type*	*form*	*precedence*
andalso	:	bool × bool → bool	infix	3
orelse	:	bool × bool → bool	infix	2
not	:	bool → bool	prefix	
=	:	bool × bool → bool *	infix	4
<>	:	bool × bool → bool *	infix	4

(* the exact type will be defined later)

Truth table

A	B	A **andalso** B	A **orelse** B
true	**true**	**true**	**true**
true	**false**	**false**	**true**
false	**true**	**false**	**true**
false	**false**	**false**	**false**

รูปที่ 2.2 ตารางค่าความสัมพันธ์และเครื่องหมายดำเนินการ [3]

2) ส่วนภาษามอดูล (Module Language) เป็นส่วนที่มีความซับซ้อนมากขึ้นขึ้น เนื่องจากมีสร้างโครงสร้างการทำงานที่ซับซ้อนเพื่อรองรับการใช้งานตามวัตถุประสงค์ที่ซับซ้อนยิ่งขึ้น โดยนำเอาตัวแปรประเภทแรก มาเรียกใช้งานในส่วนนี้ก็สามารถทำได้ ดังตัวอย่างในรูปที่ 2.3

```
structure Queue =
  struct
    type 'a queue = 'a list * 'a list
    exception Empty
    val empty = (nil, nil)
    fun insert (x, (b,f)) = (x::b, f)
    fun remove (nil, nil) = raise Empty
      | remove (bs, nil) = remove (nil, rev bs)
      | remove (bs, f::fs) = (f, (bs, fs))
  end
```

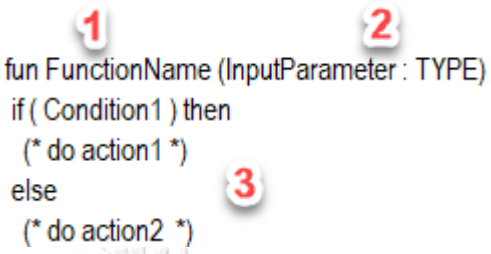
รูปที่ 2.3 รูปโครงสร้างตัวแปรประเภทโครงสร้างคิว (Queue) [1]

2.2 ภาษาโปรแกรมซีพีเอ็นเอ็มแอล (CPN ML)

เป็นภาษาที่ขยายมาจากภาษาเอสเอ็มแอล โดยเพิ่มความสามารถในการรองรับชนิดของข้อมูลหรือคัลเลอร์เซ็ต (Color Set) ซึ่งใช้ทำงานร่วมกับแบบจำลองคัลเลอร์เพทริเน็ตในโปรแกรมเครื่องมือซีพีเอ็นการกำหนดประเภทของโทเคนต่างๆจะใช้การสืบทอดประเภทข้อมูลมาจากคัลเลอร์เซ็ตพื้นฐานที่เคยกำหนดไว้ก่อน [4]

การใช้งานซีพีเอ็นเอ็มแอลจะอยู่ในรูปแบบฟังก์ชัน หรือ fun ที่ถูกกำกับไว้ในส่วนต่างๆของแบบจำลองคัลเลอร์เพทริเน็ต เช่น การ์ด (Guard) ของทรานซิชันหรือเรียกว่า การ์ดอินสคริปชัน เพื่อตรวจสอบประเภทหรือคัลเลอร์เซ็ตของโทเคนเพื่อทำการเอ็นเนเบิลหรือไม่ เป็นต้น ทำให้แบบจำลองมีความสามารถในการจัดการคัลเลอร์เซ็ตต่างๆ ลดการเขียนนิพจน์ซ้ำกันหลายๆที่ ผู้ใช้สามารถอ่าน

แบบจำลองได้ดีขึ้นและง่ายต่อการแก้ไข อีกทั้งทำให้ได้ขนาดของแบบจำลองที่เล็กลงด้วย โดยฟังก์ชันประกอบด้วยกัน 3 ส่วนหลัก คือ การรับข้อมูลนำเข้า การวัดผล และการแสดงผลลัพธ์ที่คำนวณมาได้ โดยแสดงดังรูปที่ 2.4



รูปที่ 2.4 แสดงส่วนประกอบของฟังก์ชัน

โดยมีรายละเอียดดังนี้

1) ชื่อฟังก์ชัน
2) พารามิเตอร์นำเข้า โดยประกอบด้วย ชื่อพารามิเตอร์และประเภท
3) ส่วนของนิพจน์ต่างๆ

2.3 ไทมด์ คัลเลอร์ เพทริเน็ต แบบมีลำดับชั้น (Hierarchical Timed Coloured Petrinet)

เพทริเน็ตมีหลายประเภทในปัจจุบัน โดยเพทริเน็ตพื้นฐานจะถูกเรียกว่า เพทริเน็ตระดับล่าง (Low-level Petrinet) หรือ พีทีเน็ต (P/T nets) ซึ่งจะประกอบไปด้วยเพลส (Place) ทรานซิชัน (Transition) และเส้นเชื่อมหรืออาร์ค (Arc) แต่เพทริเน็ตระดับสูง (High-level Petrinet) มักเป็นที่นิยมกว่า เนื่องจากมีการเพิ่มเติมความสามารถในหลายๆด้าน เพื่อตอบสนองการทำงานที่ซับซ้อนยิ่งขึ้น เช่น คัลเลอร์ริ่ง (Colouring) หรือการกำหนดสีหรือประเภทของโทเคน, ระบุเวลา, รองรับอาร์คชนิดพิเศษ, การกำหนดลำดับความสำคัญ (Priority) ของทรานซิชัน และการสร้างลำดับชั้นการทำงาน (Hierarchy) เป็นต้น [5] การกำหนดสี การระบุเวลา และการทำสร้างลำดับชั้นการทำงานนี้ถือเป็นส่วนหลักของการใช้งานคัลเลอร์เพทริเน็ต ซึ่งถูกเรียกว่า HTCPN (Hierarchical Timed Coloured Petri Net) ในงานวิจัยของ Andrzej Bozek, 2012 [5]

โดยมีงานวิจัย [6] ได้นิยามแบบจำลองคัลเลอร์เพทริเน็ตแบบปกติ ประกอบด้วย 9-สิ่งลำดับ (9-Tuple) หรือ CPNT = (P, T, A, **Σ**, V, C, G, E, I) โดยที่

P คือ เซตจำกัดของเพลส

T คือ เซตจำกัดของทรานซิชัน โดย P □ T = □.

A คือ เซตของเส้นเชื่อมแบบมีทิศทาง โดย A □ P × T □ T × P.

Σ คือ เซตจำกัดของคัลเลอร์เซต จะระบุเวลาหรือไม่ระบุก็ได้

V คือ เซตจำกัดของตัวแปร

C คือ คัลเลอร์เซตฟังก์ชันของเพลส C : P □ **Σ**

G คือ เซตของการ์ดฟังก์ชันบนทรานซิชัน G : T □ EXPRV.

E คือ เซตของนิพจน์บนเส้นเชื่อม E : A □ EXPRV โดย

- Type[E(a)] = C(p)MS ถ้า p ไม่ระบุเวลา (MS หรือ multi set),
- Type[E(a)] = C(p)TMS ถ้า p ระบุเวลา (TMS หรือ timed multi set),

โดย เพลส p คือเพลสต่ออยู่กับเส้นเชื่อม a.

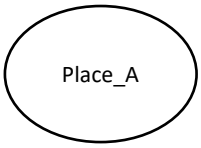
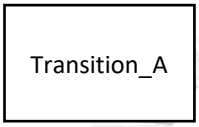
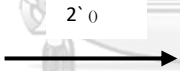
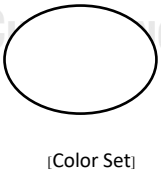
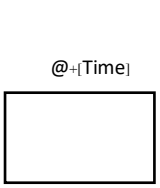
I คือ ฟังก์ชันการกำหนดค่าเริ่มต้นในแต่ละเพลส I : P □ EXPR□

Chulalongkorn University

- Type[I(p)] = C(p)MS ถ้า p ไม่ระบุเวลา
- Type[I(p)] = C(p)TMS ถ้า p ระบุเวลา

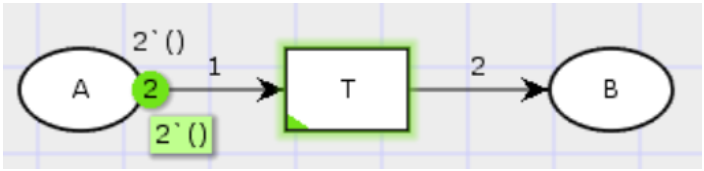
งานวิจัยได้ระบุว่า ความหมายของ HTCPN จะค่อนข้างซับซ้อน แต่โดยหลักแล้วจะอ้างอิงโดยใช้แนวคิดของไทม์มัลติเซต [5] ส่วนประกอบหลักของไทมด์คัลเลอร์เพทริเน็ตสามารถแสดงตามตารางที่ 2.1

ตารางที่ 2.1 ตารางส่วนประกอบและคำอธิบายของไทมด์คัลเลอร์เพทริเน็ต

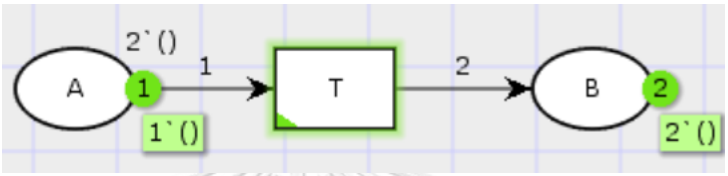
ลำดับ	ชื่อ	สัญลักษณ์	คำอธิบาย
1	**เพลส (Place)**		มีสัญลักษณ์เป็นรูปวงรี ที่มีชื่อที่กำกับไว้ต้องไม่ซ้ำกันเพลสอื่นๆ ใช้เป็นที่พักหรือที่อยู่สำหรับโทเคน ใดๆ ใช้ในการนำเสนอสถานะปัจจุบันของระบบ
2	**ทรานซิชัน (Transition)**		มีสัญลักษณ์เป็นรูปสี่เหลี่ยมผืนผ้าใดๆ ที่มีชื่อกำกับไว้ต้องไม่ซ้ำกันกับทรานซิชันอื่นๆ ใช้ในการนำเสนอการเปลี่ยนสถานะของระบบ โดยสถานะจะถูกเปลี่ยนเมื่อเกิดการยิง (Firing) โดยสามารถเพิ่มเติมเงื่อนไขบนการ์ดเรียกว่า การ์ดอินสคริปชัน (Guard Inscription) และเวลาเรียกว่า ไทม์อินสคริปชัน (Time Inscription) ก่อนเกิดการยิงได้
3	**โทเคน (Token)**		เพทริเน็ตทั่วไปโทเคนจะมีสัญลักษณ์เป็นจุดสีดำ แต่ในคัลเลอร์เพทริเน็ตโทเคนจะมีชนิดหรือคัลเลอร์เซ็ตกำกับไว้ อาจแสดงเป็นตัวอักษรหรือเลขแทนก็สามารถทำได้ โทเคนสามารถย้ายจากเพลสใดๆ ไปสู่อีกเพลสหนึ่งตามเส้นเชื่อมได้ เมื่อมีการยิง ทำให้สถานะระบบเปลี่ยนไป
4	**อาร์ค (Arc)**		อาร์คหรือเส้นเชื่อมที่เชื่อมระหว่างเพลสและทรานซิชัน เมื่อเกิดการยิงโทเคนจะถูกดึงออกจากเพลสเริ่มต้นหรือเพิ่มจำนวนโทเคนที่เพลสปลายหัวลูกศรตามจำนวนที่ถูกระบุไว้บนเส้นเชื่อมนั้นๆ
5	**คัลเลอร์เซ็ต (Color Set)**		ชนิดของข้อมูลที่ถูกระบุไว้สำหรับเพลสและโทเคนใดๆ โดยในแบบจำ-ลองคัลเลอร์เพทริเน็ต เพลสและโทเคนใดๆ ต้องมีการระบุชนิดของ ข้อมูลที่กำหนดไว้แล้วให้ตรงกัน
6	**เวลา (Time)**		เวลาที่กำหนดในแบบจำลองหรือเรียกว่า ดีเลย์ เมื่อโทเคนถูกยิง โทเคนจะถูกเพิ่มเวลาตามจำนวนที่ระบุไว้บนไทม์อินสคริปชันของทรานซิชัน เรียกว่า Timestamp ใช้ในการหน่วงเวลาในการยิง โดยเวลาจะอ้างอิงจากเวลาหลัก หรือ Global Clock

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

การยิง (Firing) คือ การนำโทเคนออกจากเพลสตั้งต้นที่ระบุ (Pre-Place) ตามจำนวนที่ระบุไว้บนเส้นเชื่อมที่ต่อไปยังทรานซิชัน และเพิ่มจำนวนโทเคนในเพลสปลายทางของทรานซิชันนั้น (Post-Place) ตามจำนวนที่ถูกระบุไว้บนเส้นเชื่อมจากทรานซิชันนั้นสู่เพลส ดังแสดงในรูปที่ 2.5 และรูปที่ 2.6



รูปที่ 2.5 ภาพแสดงก่อนการยิง



รูปที่ 2.6 ภาพแสดงหลังการยิง

ตัวอย่าง เมื่อเกิดการยิง โทเคนจะถูกดึงออกจากเพลส A จำนวน 1 ตัว ตามจำนวนที่ถูกระบุไว้บนเส้นเชื่อมระหว่างเพลส A ไปทรานซิชัน T และเพิ่มโทเคจำนวน 2 ตัว ไปยังเพลส B ตามจำนวนที่ถูกระบุไว้บนเส้นเชื่อมระหว่างทรานซิชัน T ไปยังเพลส B ดังรูปที่ 6 โดยจะเห็นได้ว่า การที่เพลส A ในรูปที่ 6 นั้นจะสามารถทำการยิงได้ เพลส A จะต้องเอ็นเนเบิล (Enable) ก่อน การเอ็นเนเบิลของทรานซิชันนั้น ก็ต่อเมื่อเพลสต้นทาง (Pre-place) ของทรานซิชัน มีจำนวนโทเคนเท่ากับหรือมากกว่าจำนวนที่ถูกระบุไว้บนเส้นเชื่อมระหว่างเพลสและทรานซิชัน และเงื่อนไขการ์ดของทรานซิชันจะต้องมีค่าความจริงเป็นจริงก่อน ถ้าหากมีการระบุเงื่อนไขการ์ด

ในกรณีของ HTCPN เวลาจะถูกเพิ่มเติมเข้าไปที่เพลสและโทเคน ทำให้ HTCPN มีส่วนประกอบเพิ่มเติมคือ โทเคนไทม์แสตมป์ (Timestamp หรือ TT) และ โกลบอลคล็อค (Global Clock หรือ GC) ในเพลสปกติยังคงรองรับทั้งโทเคนประเภทที่ระบุเวลาและไม่ระบุเวลาอยู่ แต่โทเคนที่อยู่บนเพลสที่มีประเภทระบุเวลาจะถูกเติมค่าไทม์แสตมป์ต่อท้ายหลังโทเคนนั้น โดยจะอยู่ในรูปของ @+tt โดย tt คือค่าไทม์แสตมป์ของโทเคน มีผลทำให้การเอ็นเนเบิลของทรานซิชันจะถูกเพิ่มด้วยเงื่อนไขของเวลาด้วย โดยกำหนดว่าทรานซิชันจะถูกเอ็นเนเบิลก็ต่อเมื่อโทเคนมีค่าไทม์แสตมป์เท่ากับหรือน้อยกว่าเวลาของโกลบอลคล็อค [5] เมื่อค่าโกลบอลคล็อคเป็นเวลามาตรฐานที่ใช้ร่วมกันของทั้งแบบจำลองและมีค่าเพิ่มมากขึ้นเรื่อยๆเมื่อทำการจำลอง (Simulation) และหากเวลาถูกระบุไว้ที่ไทม์สอินสคริปชันของทรานซิชันในที่นี้จะเรียกว่า ดีเลย์ ทำให้ทรานซิชันไม่สามารถเอ็นเนเบิลได้ก่อนครบเวลาที่ระบุ เมื่อมีการยิงโทเคนเข้า

ไปในเพลสหลังทรานซิชันนั้น โทเคนจะมีไทม์แสตมป์เท่ากับ a+b โดยให้ a คือ เวลาไทมป์แสตมป์ของโทเคนก่อนการยิง และ b คือ เวลาดีเลย์บนทรานซิชันนั้น

กำหนดทฤษฎีการเอ็นเนเบิลของไทม์เพทริเน็ต [5]

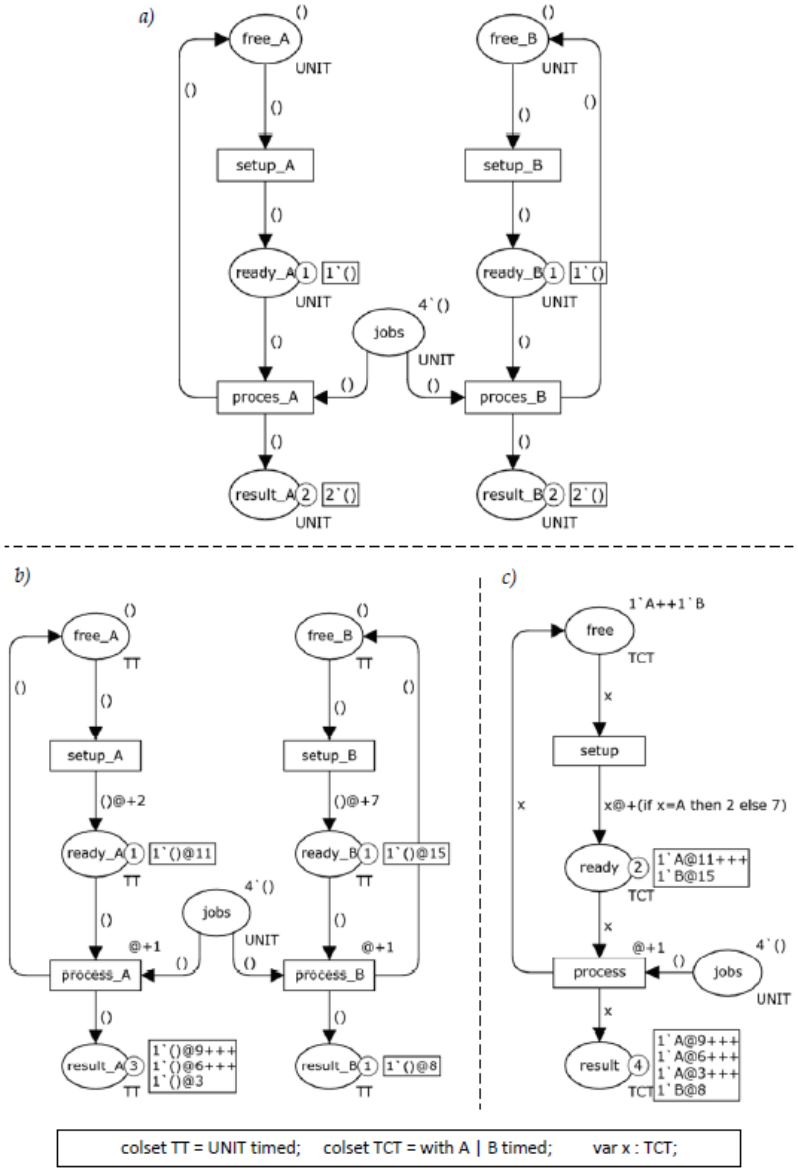
1) ตั้งค่าโกลบอลคล็อคเริ่มต้นที่ 0
2) ถ้ามีทรานซิชันที่สามารถเอ็นเนเบิลได้ ให้ไปข้อ 4. ถ้าไม่ไปข้อ 3.
3) เพิ่มค่าโกลบอลคล็อคที่น้อยที่สุด ที่สามารถทำให้ทรานซิชันเอ็นเนเบิลได้ ถ้าเป็นไปไม่ได้เลย ให้จบการจำลอง
4) สุ่มเลือกซับเซ็ตของทรานซิชันใดๆที่เอ็นเนเบิลได้
5) ปฏิบัติการ(Execute)บนทรานซิชันที่เลือกแล้ว
6) กลับไปข้อ 2.

จากรูปที่ 2.7a เนื่องจากเป็นแบบจำลองแบบไม่ระบุเวลา ทำให้ proces_A และ process_B สามารถทำงานได้พร้อมกันด้วยเวลาเท่ากัน เมื่อมีงานหรือโทเคนเข้ามา

จากรูปที่ 2.7b มีการระบุดีเลย์ในการทำงานของ setup_A คือ @+2 และ setup_B คือ @+7 ทำให้ไทม์แสตมป์ของงานหรือโทเคนที่เข้ามาผ่านการทำงานที่ setup_A จะถูกบวกเพิ่มเป็นจำนวน 2 หน่วย และไทม์แสตมป์ของงานหรือโทเคนที่ผ่านการทำงานที่ setup_B จะถูกบวกเพิ่มเป็นจำนวน 7 หน่วย และถูกบวกเพิ่มอีก 1 หน่วยจากการทำงานที่ process_A หรือ process_B โดยรวมจะเห็นเวลาที่ใช้ในการทำงานสำหรับส่วนของงาน A มีค่าเป็น 3 หน่วยต่อหนึ่งงาน และส่วนของงาน B จะมีค่าเป็น 8 หน่วย ซึ่งเมื่อเริ่มจำลองการทำงาน จะเห็นได้ว่าการทำงานของส่วนงาน A จะรวดเร็วกว่าทำให้มีจำนวนโทเคนมากกว่า และไทม์แสตมป์ของโทเคนของ result_A มีค่าเป็น @3, @6, @9 และ result_B มีค่าเป็น @8 เพียงตัวเดียวเท่านั้น

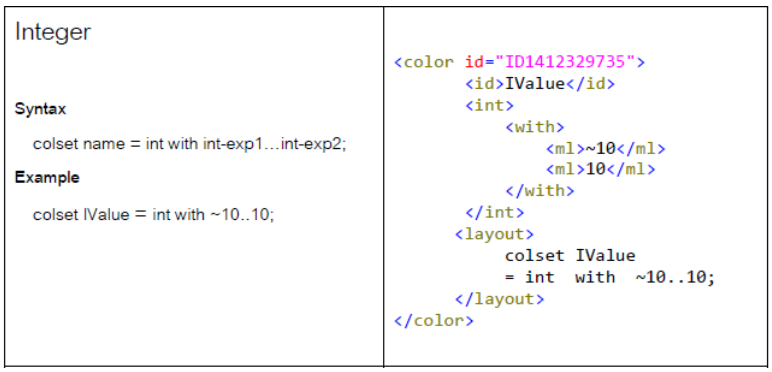
จากรูปที่ 2.7c ใช้หลักการเดียวกันกับรูปที่ 2.7b แต่ทำการเพิ่มคัลเลอร์เซ็ตแก่โทเคน ทำให้ผลลัพธ์สุดท้ายมีโทเคนที่เป็น คัลเลอร์เซ็ต A 3 ตัว และ คัลเลอร์เซ็ต B 1 ตัว ตามดีเลย์ที่กำหนดไว้บนทรานซิชันต่างๆ

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY



รูปที่ 2.7 ตัวอย่างแบบจำลองเพทริเน็ตทั่วไป, แบบจำลองไทม์เพทริเน็ต, และแบบจำลองไทม์คัลเลอร์เพทริเน็ต [5]

งานวิจัยนี้ได้เลือกใช้โปรแกรมเครื่องมือซีพีเอ็น ในการทดสอบแบบจำลองไทมด์คัลเลอร์เพทริเน็ต เนื่องจากเครื่องมือนี้รองรับการใช้งานคัลเลอร์เพทริเน็ตได้เป็นอย่างดีและสนับสนุนการใช้เงื่อนไขเวลาในแบบจำลองด้วย ซึ่งเครื่องมือนี้สามารถติดตั้งได้บนระบบปฏิบัติการหลัก ทั้งวินโดวส์ ลีนุกส์และแมคโอเอส โดยแบบจำลองที่ผู้ใช้งานออกแบบจจะถูกบันทึกเป็นเอกสารที่มีนามสุกลเป็น .CPN ที่มีโครงสร้างเป็นเอ็กซ์เอ็มแอลที่เป็นมาตรฐานของภาษาซีพีเอ็นเอ็มแอล (CPN ML) ในโปรแกรมเครื่องมือซีพีเอ็น ดังรูปที่ 2.8 ผู้ใช้งานสามารถเรียกดูข้อมูล และแสดงผลในรูปแบบกราฟฟิคได้



รูปที่ 2.8 ตัวอย่างการประกาศคัลเลอร์เซตแบบพื้นฐานเทียบโครงสร้างเอ็กซ์เอ็มแอลไฟล์ .cpn [5]

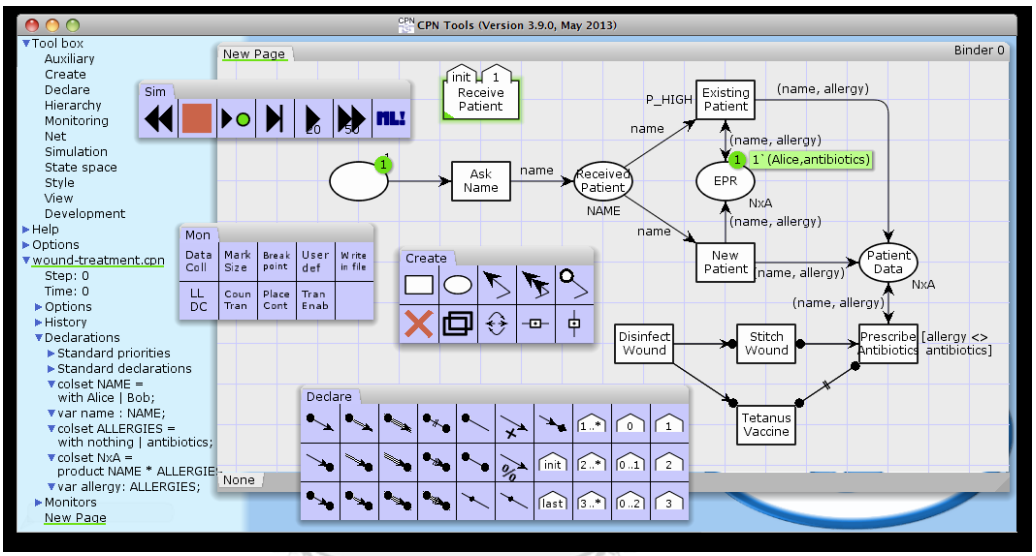
ข้อมูลในเอกสารนามสกุล .CPN จะประกอบด้วยส่วนที่เป็นคีย์อิลิเมนต์ที่สนใจในงานวิจัยนี้ เพื่อใช้สร้างแบบจำลองไทมด์คัลเลอร์เพทริเน็ตอัตโนมัติ ดังตารางที่ 2.2 เมื่ออิลิเมนต์ใดๆถูกสร้าง ข้อมูลจะถูกจัดเก็บไว้ในคีอิลิเมนต์ประเภทนั้นๆเสมอ

ตารางที่ 2.2 ตารางคีอิลิเมนต์ที่สำคัญในเอกสารนามสกุล .CPN

ชื่อส่วนประกอบ	**คีอิลิเมนต์**	**คำอธิบาย**
คัลเลอร์เซ็ต	<color>...</color>	เก็บข้อมูลคัลเลอร์เซต ประกอบด้วย ไอดี, ชื่อ, ประเภทและคำสั่ง
ตัวแปร	<var> ... </var>	เก็บข้อมูลตัวแปร ประกอบด้วย ไอดี, ชื่อ, ประเภทและคำสั่ง
ฟังก์ชัน	<ml> ... </ml>	เก็บข้อมูลฟังก์ชันและคำสั่งที่ถูกเขียนไว้เรียกใช้งาน
เพลส	<place>...</place>	เก็บข้อมูลเพลส ประกอบด้วย ไอดี, ชื่อ, ประเภทและข้อมูลโทเคน
ทรานซิชัน	<trans> ... /trans>	เก็บข้อมูลทรานซิชัน ประกอบด้วย ไอดี, ชื่อ, การ์ดอินคริปชันและไทม์อินสคริปชัน
อาร์ค	<arc> ... </arc>	เก็บข้อมูลอาร์ค ประกอบด้วย ไอดี, ทิศทาง, เพลสไอดีและทรานซิชันไอดีที่เชื่อมต่อ, และข้อมูลบนอาร์ค

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ปัจจุบันโปรแกรมเครื่องมือซีพีเอ็น เป็นเวอร์ชัน 4 ซึ่งได้ขยายความสามารถในด้านต่างๆเพื่อรองรับการใช้งานที่กว้างขึ้น ซึ่งมีภาษาซีพีเอ็นเอ็มเอลที่เป็นภาษาที่ใช้ในโปรแกรมอยู่แล้วซึ่งอาจจะมีข้อจำกัดของการเป็นภาษาประเภทดีแคเลทีฟ (Declarative Language) อยู่บ้าง โดยเพิ่มส่วนต่อขยายต่างๆ อาธิเช่น การแสดงผลแบบไฮเลเวล (High-Level Visualization) ที่สามารถสร้าง แผนภาพลำดับ (Sequence Diagram) และแผนภูมิทางสถิติ (Statistic Graph) จากแบบจำลองไฮบริดคัลเลอร์เพทริเน็ตในรูปที่ 2.9 เป็นต้น



รูปที่ 2.9 มุมมองแบบจำลองไฮบริดของตัวอย่างการให้บริการในโรงพยาบาล [7]

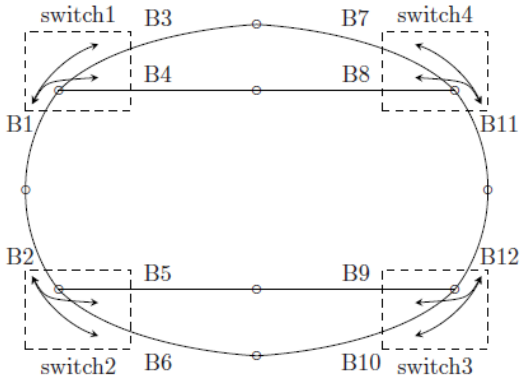
จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

2.4 งานวิจัยที่เกี่ยวข้อง

มีงานวิจัยเกี่ยวกับภาษาเชิงรูปนัยและเพทริเน็ตประเภทต่างๆ และประยุกต์ใช้งานกับระบบรถไฟมากมายที่เกิดขึ้นมาก่อนหน้า โดยบางงานวิจัยได้ทำการศึกษาการออกแบบตารางเวลา, ปัญหาการดีเลย์ของรถไฟ, การจัดการที่ตัวสถานี, หรือตรวจสอบการควบคุมอุปกรณ์ความปลอดภัยต่างๆ รวมถึงงานวิจัยที่เกี่ยวข้องในงานวิจัยนี้ ดังเช่น

2.4.1 งานวิจัยของ Christine Choppy, Laure Petrucci, and Gianna Reggio หัวข้อ A Modelling Approach With Coloured Petri Nets. ในปี 2008

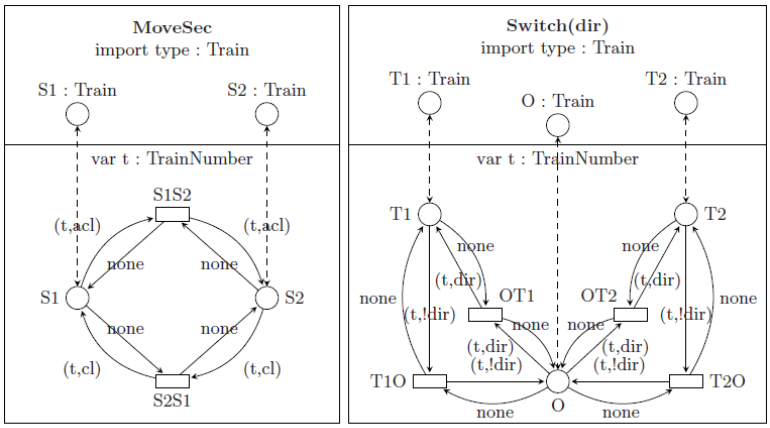
[8] ได้ใช้แบบจำลองฟอมอลในการวิเคราะห์และออกแบบเพื่อแก้ปัญหาความปลอดภัยสำหรับฮาร์ดแวร์และซอฟแวร์ต่างๆ โดยทดลองสร้างแบบจำลองคัลเลอร์เพทริเน็ตสำหรับระบบราง ซึ่งโจทย์ คือ ระบบรางรถไฟของเล่นที่ประกอบไปด้วย 12 รางรถไฟ แต่ละรางมีความยาว 15 เมตร และ มี 4 จุดสลับราง ในร้านค้าแห่งหนึ่งดังรูปที่ 2.10



รูปที่ 2.10 แบบจำลองขั้นสูงของระบบรางรถไฟของเล่น [8]

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

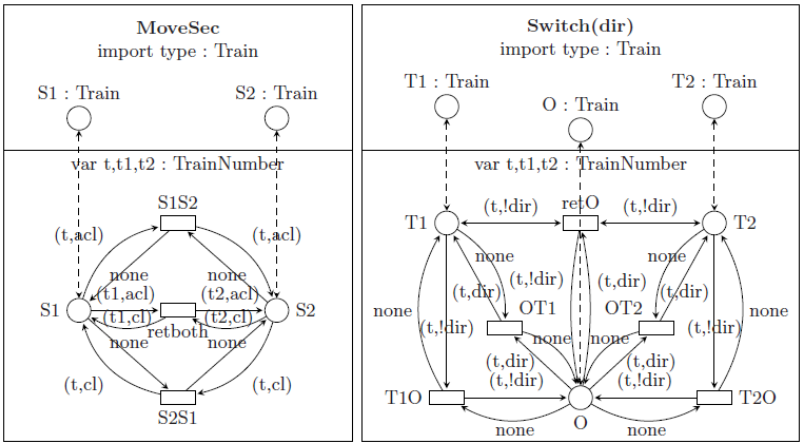
โดยมีข้อกำหนดว่า รถไฟจะวิ่งไปในทางเดิมแต่สามารถวิ่งกลับทิศได้ถ้าหากไม่สามารถไปต่อได้ ความเร็วขบวนรถไฟไม่สามารถเปลี่ยนแปลงได้ แต่สามารถหยุดได้ รถไฟมีหลายขบวนวิ่งพร้อมกัน แต่ต้องไม่ชนกัน โดยจากข้อกำหนดข้างต้นทำให้ผู้วิจัยสามาถทำการออกแบบแบบจำลองคัลเลอร์เพทริเน็ตสำหรับ 2 ส่วนย่อยจากแบบจำลองใหญ่ คือ ส่วนย่อยประเภทรางรถไฟ และส่วนย่อยประเภทจุดสลับราง ดังรูปที่ 2.11



(a) The `MoveSec` module (b) The `Switch(dir)` module

รูปที่ 2.11 แบบจำลองส่วนย่อยของรางรถไฟ (a) และ ส่วนย่อยจุดสลับราง (b) [8]

แต่อย่างไรก็ตาม ผู้วิจัยได้ตรวจพบปัญหาการติดตาย (Deadlock) หลังจากจำลองการทำงานเมื่อประกอบแบบจำลองส่วนย่อยทั้งหมดเป็นแบบจำลองเครือข่ายทางรถไฟ จึงทำให้ผู้วิจัยได้เสนอการปรับแบบจำลองส่วนย่อยทั้ง 2 เพื่อแก้ไขปัญหา ดังรูปที่ 2.12 โดยเพิ่ม ทรานซิชันชื่อ Retboth ในแบบจำลองส่วนย่อยราง เพื่อให้รถไฟสามารถวิ่งกลับทิศไปในทิศทางเดิมหากรถไฟไม่สามารถแล่นผ่านรางตำแหน่งนั้นได้ เนื่องจากมีรถไฟอีกขบวนหนึ่งที่กำลังแล่นสวนมา ในทำนองเดียวกันกับแบบจำลองส่วนย่อยจุดสลับราง ผู้วิจัยได้ทำการเพิ่มทรานซิชันชื่อ RetO เพื่อแก้ไขปัญหาติดตายเช่นกัน



(a) The new `MoveSec` module (b) The new `Switch(dir)` module

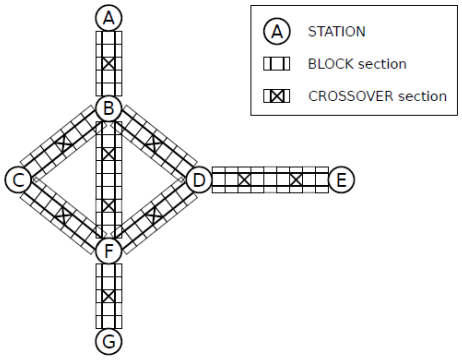
รูปที่ 2.12 แบบจำลองส่วนย่อยใหม่เพื่อแก้ปัญหาการติดตายในระบบราง [8]

โดยสรุปงานวิจัยนี้ได้เสนอแบบจำลองเครือข่ายทางรถไฟและทดสอบแบบจำลองด้วยคัลเลอร์เพทริเน็ต ซึ่งแบบจำลองที่ใช้ในงานวิจัยนี้ค่อนข้างเรียบง่าย โดยไม่สามารถใช้งานกรณีรถไฟมีหลายประเภทที่วิ่งด้วยความเร็วต่างกัน หรือรองรับกรณีมีจุดจอดที่เป็นสถานีต่างๆ และไม่มีปัญหาด้านเวลามา

เกี่ยวข้อง แต่สามารถนำงานวิจัยนี้มาปรับใช้แก่งานวิจัยที่มีสลับซับซ้อนขึ้นได้ตามแนวทางที่ผู้วิจัยเดิมได้นำเสนอในการสร้างแบบจำลองส่วนย่อยรางรถไฟ

2.4.2 งานวิจัยของ Davide Giglio and Nicola Sacco หัวข้อ A Petri net model for analysis, optimisation, and control of railway networks and train schedules. ในปี 2016

[9] ได้ออกแบบและวิเคราะห์เครือข่ายทางรถไฟประเภทรางคู่โดยใช้ไทม์เพทริเน็ตในการสร้างแบบจำลองเครือข่ายทางรถไฟ โดยจำลองประกอบด้วย 7 สถานี 8 เส้นทางรถไฟ และ 11 ขบวนรถไฟ ที่วิ่งระหว่างสถานีหนึ่งถึงอีกสถานีหนึ่งดังรูปที่ 2.13 โดยมีตารางเวลาที่กำหนดให้สำหรับกำหนดเส้นทาง เวลาเข้าและออกจากแต่ละสถานีตามรูปที่ 2.14



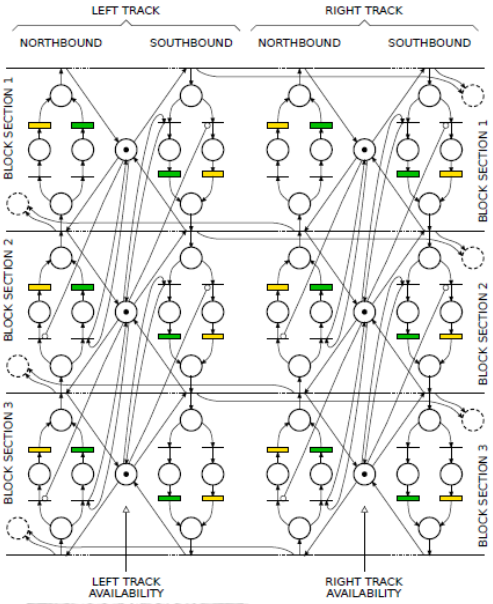
รูปที่ 2.13 แบบจำลองเครือข่ายทางรถไฟสำหรับระบบรางคู่จำนวน 7 สถานี 8 เส้นทางรถไฟ [9]

Train	Line	Stations													
		A		B		C		D		E		F		G	
		a	d	a	d	a	d	a	d	a	d	a	d	a	d
1	I	–	0	7	9	19	21					31	33	41	–
2	I	–	30	37	39	49	51					61	63	71	–
3	II	–	15	22	24							38	40	47	–
4	II	–	45	52	54							68	70	77	–
5	III			25	27	–	15	37	39	52	–				
6	IV	40	–	30	32			18	20	–	5				
7	V					67	–	43	45	–	30	55	57		
8	VI	51	–	41	43	29	31					17	19	–	10
9	VII	57	–	48	50							32	34	–	25
10	VII	87	–	78	80							62	64	–	55
11	VIII							59	61	75	–	47	49	–	40

รูปที่ 2.14 รูปแสดงตารางเวลาเข้าออกของรถไฟ 11 ขบวนในแต่ละสถานี [9]

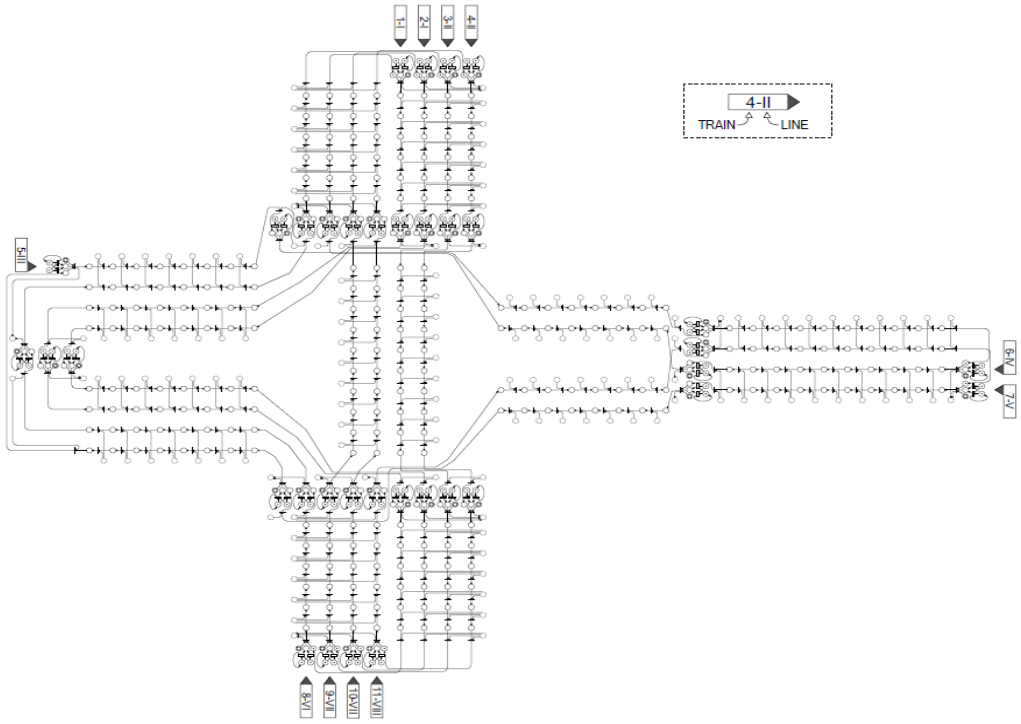
งานวิจัยได้นำเสนอไทม์เพทริเน็ตโดยได้แบ่งส่วนประกอบด้วยกัน 2 ส่วน คือ ส่วนประกอบจริงของแบบจำลอง (Physical Part) ที่ออกแบบเป็นส่วนประกอบแยกส่วน (Modular) ต่างๆ เพื่อนำไปประกอบกัน ดังตัวอย่างในรูปที่ 2.15 และ ส่วนประกอบที่เป็นตรรกะ (Logical Part) ที่มาจากการ

เชื่อมต่อส่วนประกอบแยกส่วนต่างๆเข้าด้วยกัน โดยจัดวางส่วนประกอบจริงต่างๆนั้นให้ได้แบบจำลองตามที่ต้องการ แล้วเชื่อมต่อส่วนประกอบที่เป็นตรรกะเพื่อความคุมหรือตรวจสอบสถานะของแบบจำลอง เช่น การปล่อยรถไปตามเส้นทางที่ถูกต้อง และเป็นไปตามตารางเวลา และป้องกันปัญหาต่างๆที่อาจจะเกิดขึ้นในระบบ เช่น การติดตายของขบวนรถไฟในแบบจำลอง การจัดลำดับความสำคัญของขบวนรถไฟในกรณีมีเหตุจำเป็น เป็นต้น ดังรูปที่ 2.16



รูปที่ 2.15 ตัวอย่างไทมด์เพทริเน็ตของส่วนประกอบจริงของหน่วยรางรถไฟ (Block Section) 3 หน่วยที่เรียงต่อกัน [9]

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY



รูปที่ 2.16 แบบจำลองรางรถไฟที่กำหนดที่ประกอบไปด้วยส่วนประกอบจริงและส่วนประกอบที่เป็นตรรกะ [9]

2.4.3 งานวิจัยของ T. Sakaguchi and N. Tomii หัวข้อ A train traffic model based on coloured Petri Nets and its application to a train schedule planning system ในปี 1996

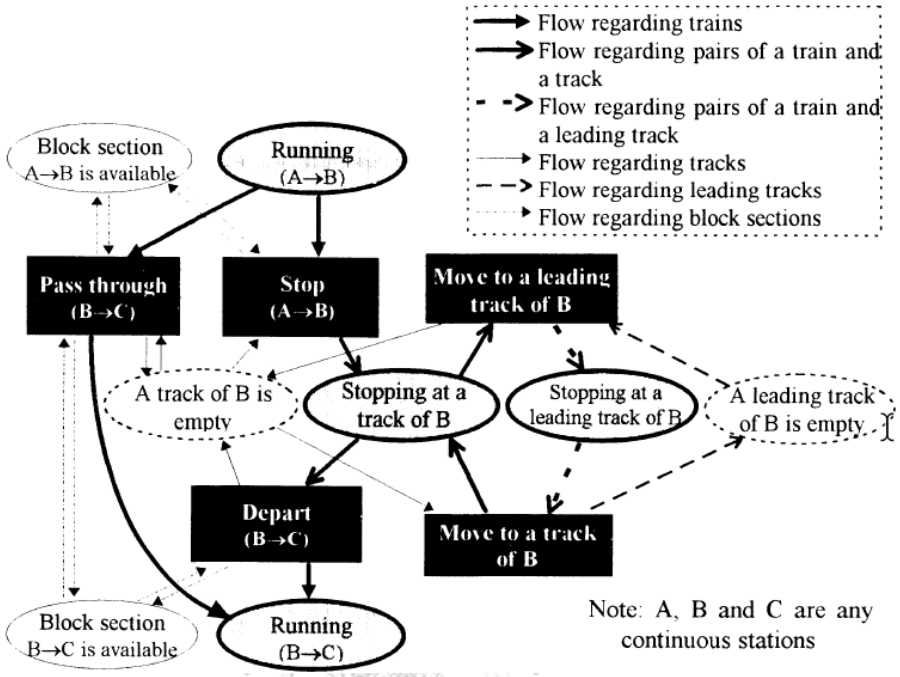
[10] ได้กล่าวถึงเหตุการณ์ที่อาจจะเกิดขึ้นในระบบการเดินรถ ซึ่งอาจต้องมีการกำกับเงื่อนไขเพื่อแก้ไขปัญหา ดังนี้

1). เงื่อนไขการปล่อยรถจากสถานี เมื่อรถไฟพร้อมที่จะออกสถานี รางรถไฟถัดไปที่อยู่ติดกับสถานีจะต้องมีสถานะ “ว่าง” เท่านั้น รถไฟถึงจะออกจากสถานีได้

2). เงื่อนไขการเข้าสถานี รถไฟขบวนใดๆสามารถเข้าสถานีก่อนเวลาที่ถูกกำหนดไว้ในตารางเวลาได้ แต่สถานีต้องมีรางใดรางหนึ่งมีสถานะ “ว่าง” เท่านั้น

3). เงื่อนไขการคงไว้ชั่วขณะใดขณะหนึ่ง (Temporal condition) ซึ่งโดยปกติแล้ว ต้องระมัดระวังชั่วขณะที่รถไฟสามารถขอใช้อุปกรณ์หรือส่วนประกอบบางอย่างในเวลาเดียวกันได้ เช่น เมื่อมีรถไฟขบวนหนึ่งกำลังวิ่งเข้าสู่จุดสลับรางในระหว่างที่รถไฟอีกขบวนหนึ่งกำลังออกจากจุดสลับรางต้องไม่เกิดขึ้น โดยมีเงื่อนไขในการตรวจสอบการใช้งานอุปกรณ์หรือส่วนประกอบนั้นเสร็จสิ้นก่อนที่จะอนุญาตให้อีกขบวนหนึ่งใช้งานได้.

โดยสามารถสรุปสถานะของระบบรางรถไฟในช่วงเวลาใดๆ ดังตัวอย่างรูปที่ 2.17



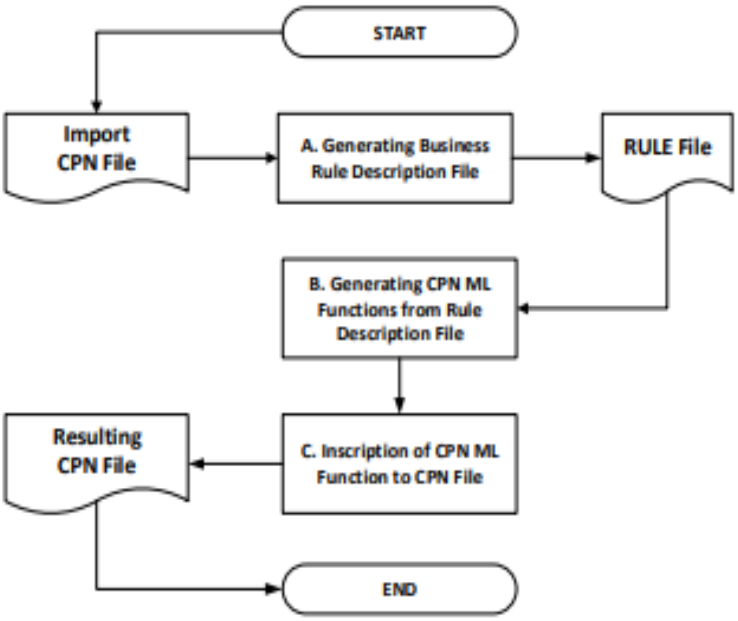
รูปที่ 2.17 ตัวอย่างการจัดการจราจรเพื่อใช้งานรางรถไฟ [10]

สามารถนำแนวคิดและข้อเสนอจากงานวิจัยข้างต้นมาปรับใช้ในงานวิจัยนี้ เพื่อปรับปรุงและแก้ปัญหาที่อาจเกิดขึ้นเมื่อสร้างแบบจำลอง

2.4.4 งานวิจัยของ Deesukying, J. and Vatanawood, W. หัวข้อ Generating of business rules for Coloured Petri Nets. ในปี 2016

[11] ได้เสนอแนวทางในการแปลงกฎทางธุรกิจที่อยู่ในรูป IF-THEN ELSE ไปเป็นภาษาซีพีเอ็นเอ็มแอล โดยที่ผู้ใช้งานไม่จำเป็นต้องมีทักษะการเขียนโปรแกรม เพื่อช่วยเหลือผู้ใช้ให้สามารถใช้งานโปรแกรมเครื่องมือซีพีเอ็น ช่วยในการวิเคราะห์กฎธุรกิจใหม่ว่ามีความถูกต้องหรือตรวจสอบปัญหาต่างๆได้ โดยงานวิจัยได้เสนอต้นแบบ (CPN ML Template) ที่ใช้ในการแปลง อธิบายตัวแปรต่างๆที่สำคัญในการแปลงคัลเลอร์เพทริเน็ตที่มีรูปแบบของภาษาซีพีเอ็นเอ็มแอลเป็นส่วนประกอบ เมื่อมีกฎธุรกิจใหม่ที่ถูกสร้างและต้องการเพิ่มไปยังแบบจำลองคัลเลอร์เพทริเน็ตเดิม ผู้ใช้งานสามารถนำเข้าแบบจำลองคัลเลอร์เพทริเน็ตเดิมและกฎธุรกิจใหม่ที่อยู่ในรูปเอ็กซ์เอ็มแอล แปลงเป็นข้อมูลนำออกที่อยู่ในรูปแบบภาษาซีพีเอ็นเอ็มแอล แบบจำลองคัลเลอร์เพทริเน็ตใหม่นี้จะถูกปรับปรุงในส่วนของรายละเอียดเงื่อนไข หรืออินสคริปชัน ตามกฎธุรกิจใหม่ ที่มีความสอดคล้องกับรูปแบบที่ใช้งานใน

โปรแกรมเครื่องมือซีพีเอ็น ทำให้โปรแกรมสามารถแสดงผลลัพธ์ให้ผู้ใช้งานได้ โดยมีขั้นตอนการทำงานดังรูปที่ 2.18



รูปที่ 2.18 ขั้นตอนภาพรวมงานวิจัย [11]

งานวิจัยนี้จะนำเอาแนวคิดการแปลงกฎธุรกิจที่อยู่ในรูปแบบเอ็กซ์เอ็มแอลมาสร้างเป็นข้อมูลนำออกใหม่ที่สามารถใช้งานด้วยโปรแกรมเครื่องมือซีพีเอ็น อัตโนมัติ มาใช้งาน

โดยสามารถสรุปงานวิจัยที่เกี่ยวข้องต่างๆได้ดังตารางที่ 2.3 นี้

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

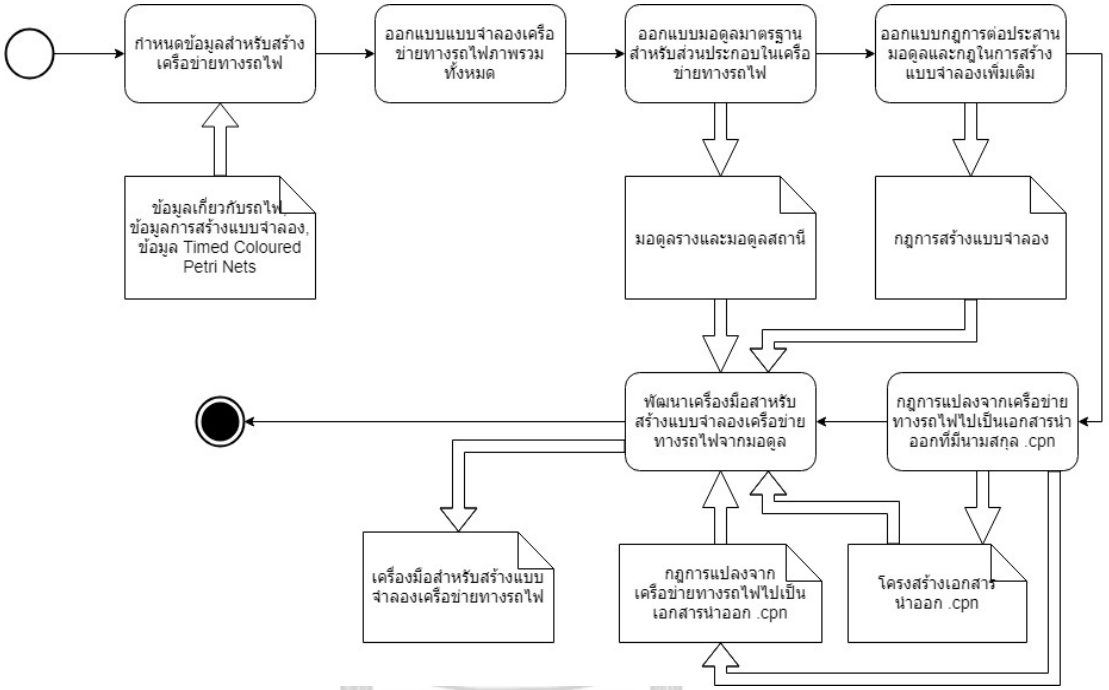
ตารางที่ 2.3 ตารางสรุปเปรียบเทียบงานวิจัยต่างๆ

งานวิจัย	รายละเอียดงานวิจัย	ประเภทเพทริเน็ต	ความแตกต่าง	ส่วนที่นำมาใช้ในงานวิจัย
2.4.1	ทำการทดลองสร้างแบบจำลองคัลเลอร์เพทริเน็ตสำหรับระบบรางรถไฟทีกำหนด โดยได้เสนอแบบจำลองย่อย MoveSec และ Switch ที่แก้ไขปัญหาติดตายในระบบรางได้	คัลเลอร์เพทริเน็ต	ไม่สนใจเรื่องของเวลา, ไม่มีตารางเวลา, รถไฟไม่จอดสถานี, ขรถไฟทุกประเภทวิ่งด้วยความเร็วเท่ากัน	การออกแบบแบบจำลองย่อยสำหรับรางรถไฟและจุดสลับราง
2.4.2	สร้างแบบจำลองไทมด์เพทริเน็ตสำหรับระบบรางคู่ที่มี 7 สถานี 8 เส้นทางและ 11 ขบวนรถไฟ และตารางเวลาที่กำหนด ได้เสนอส่วนประกอบ 2 ส่วน คือ ส่วนประกอบจริงของแบบจำลองที่ออกแบบเป็นส่วนประกอบแยกส่วน (Modular) และส่วนประกอบที่เป็นตรรกะที่ใช้ควบคุมการเดินรถไฟ	ไทม์เพทริเน็ต	มีรถไฟประเภทเดียว, ไม่ใช้เลขขบวนในการตรวจสอบเส้นทางการเดินรถ, ใช้การออกแบบทิศทางขึ้นเหนือ (North Bound) และล่องใต้ (South Bound) แยกกันอย่างอิสระ	การออกแบบแบบจำลองย่อยสำหรับรางรถไฟและแนวคิดการสร้างส่วนควบคุม
2.4.3	เสนอแนวคิดและวิธีการในการออกแบบระบบรางรถไฟด้วยคัลเลอร์เพทริเน็ต สามารถสร้างแบบจำลองระบบรางรถไฟเพื่อนำไปจำลองตารางรถไฟ โดยผลลัพธ์การจำลองมีค่าใกล้เคียงตารางเวลาจริงของรถไฟ	คัลเลอร์เพทริเน็ต	ไม่ได้ลงรายละเอียดวิธีการสร้างแบบจำลอง	แนวคิด ข้อจำกัด และเหตุการณ์ที่ต้องระบุเพื่อจัดการอุปกรณ์ในระบบรางรถไฟ
2.4.4	เสนอการสร้างคัลเลอร์เพทริเน็ตและแก้ไขอินสคริปชันจากกฎธุรกิจใหม่ที่กำหนดขึ้นที่อยู่ในรูปแบบเอ็กซ์เอ็มแอล	คัลเลอร์เพทริเน็ต	-	การแปลงเอ็กซ์เอ็มแอลไปเป็นคัลเลอร์เพทริเน็ต

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

บทที่ 3
การออกแบบและการสร้าง CPN

งานวิจัยนี้ได้มีแนวคิดการทำวิจัยในภาพรวมทั้งหมด ดังที่แสดงในรูปที่ 3.1



รูปที่ 3.1 แนวคิดในการทำวิจัยในภาพรวม

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

โดยได้เริ่มจากการศึกษาหาข้อมูลที่ต้องใช้และกำหนดข้อมูลพื้นฐานที่จะใช้ประกอบการสร้างแบบจำลองเครือข่ายทางรถไฟ ในหัวข้อ 3.1 โดยนำเข้าข้อมูลทั้งหมดเพื่อสร้างแบบจำลองไทม์คัลเลอร์เพทริเน็ตจากนั้นทำการกำหนดส่วนโครงสร้างมอดูล เพื่อใช้เป็นต้นแบบมอดูลมาตรฐานไทม์คัลเลอร์เพทริเน็ต สำหรับสถานีและรางรถไฟ พร้อมกำหนดการต่อประสานมอดูลมาตรฐานและกฎเพิ่มเติมสำหรับการสร้างแบบจำลองไทด์คัลเลอร์เพทริเน็ต ดังจะแสดงในหัวข้อ 3.2 จากนั้นกำหนดข้อมูลและโครงสร้างที่ใช้ในการสร้างเอกสารนำออกที่มีนามสกุล .cpn ในหัวข้อ 3.3 พร้อมทั้งกฎการแปลงจากเครือข่ายทางรถไฟไปเป็นเอกสารนำออกที่มีนามสกุล .cpn ในหัวข้อ 3.4 ซึ่งทั้งหมดจะเรียกใช้งานผ่านเครื่องมือช่วยเหลือในการแปลงเครือข่ายทางรถไฟเป็นไทม์คัลเลอร์เพทริเน็ตในหัวข้อ 3.5

3.1 ข้อมูลประกอบการสร้างเครือข่ายทางรถไฟ

จากการศึกษาหาข้อมูลที่เกี่ยวข้องกับการรถไฟและเครือข่ายทางรถไฟ เช่น ประเภทของราง สถานี ตารางเวลารถไฟ การปล่อยขบวนรถเข้าออกสถานี เป็นต้น รวมถึงกฎการเดินรถที่เกี่ยวข้อง เพื่อนำมาใช้ประกอบการสร้างแบบจำลองไทมด์คัลเลอร์เพทริเน็ตสำหรับเครือข่ายทางรถไฟให้มีความสามารถตอบสนองกับความต้องการจริง โดยใช้แหล่งข้อมูลจากการรถไฟแห่งประเทศไทย [12] และคำนึงถึงกฎการเดินรถพื้นฐานต่างๆ คือ รถไฟสามารถเข้าสถานีก่อนเวลาที่ระบุไว้ในตารางเวลารถไฟได้ แต่รถไฟห้ามออกจากสถานีก่อนเวลาที่ระบุไว้ในตารางเวลารถไฟของสถานีนั้นๆ รถไฟไม่สามารถวิ่งสวนกันบนรางเดียวกันในเวลาเดียวกันได้ โดยงานวิจัยนี้จะอ้างอิงเงื่อนไขของงานวิจัย [10] ในการจัดการปล่อยขบวนรถไฟเข้าออกตามสถานีต่างๆ พร้อมทั้งศึกษาข้อมูลโครงสร้างของข้อมูลนำออกที่อยู่ในรูปแบบภาษาซีพีเอ็นเอ็มแอล [11] เพื่อเป็นข้อมูลประกอบการแปลงข้อมูลนำเข้าที่อยู่ในรูปแบบเอ็กซ์เอ็มแอลไปเป็นไทมด์คัลเลอร์เพทริเน็ต

โดยการออกแบบเครือข่ายทางรถไฟภายใต้ข้อกำหนดการออกแบบตารางของการไฟแห่งประเทศไทยมีดังนี้

1) กรณีรางรถไฟที่เชื่อมต่อสถานีเป็นรางเดี่ยว จะมีเพียงรถไฟขบวนหนึ่งขบวนใดที่ได้รับอนุญาตให้ใช้งานรางนั้นในช่วงเวลานั้นเท่านั้น ขบวนอื่นที่ต้องการใช้งานรางนั้นต้องจอดรอที่สถานีหรือจุดพักรถไฟ จนกว่ารางรถไฟที่เชื่อมต่อสถานีนั้นมีสถานะเป็นว่างเสียก่อน
2) กรณีรางรถไฟที่เชื่อมต่อสถานีเป็นรางคู่ รางหนึ่งรางใดจะถูกกำหนดให้เฉพาะรถไฟเส้นทางขาล่องเดียวกันใช้งานเท่านั้น (One-way)
3) เวลาจอดรถไฟในแต่ละสถานี เพื่อรับส่งผู้โดยสาร จะใช้ค่าคงที่ซึ่งเป็นเวลามาตรฐาน 1 นาที
4) ความเร็วรถไฟกำหนดเป็นความเร็วเฉลี่ยของรถไฟแต่ละประเภทในช่วงสถานการณ์ปกติ
5) รถไฟที่นำมาใช้ประกอบการจำลองในงานวิจัย เฉพาะรถไฟสำหรับโดยสารเท่านั้น โดยจะแบ่งประเภทรถไฟดังนี้
    a. รถไฟชนิดรถด่วนหรือด่วนพิเศษ เลขขบวนตั้งแต่ 1 – 99
    b. รถไฟชนิดรถเร็ว เลขขบวนตั้งแต่ 100 – 199
    c. รถไฟชนิดรถธรรมดาหรือชานเมือง เลขขบวนตั้งแต่ 200 – 499
6) ทิศทางขาล่องถูกกำหนดตามเลขของรถไฟ
    a. ทิศทางขาล่องเข้าสถานีกรุงเทพหรือขาเข้า เลขขบวนคู่
    b. ทิศทางขาล่องออกจากสถานีกรุงเทพหรือขาออก เลขขบวนคี่

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

7) ค่าพารามิเตอร์เริ่มจากข้อมูลเบื้องต้นที่ผู้ใช้งานสามารถปรับแต่งได้มีดังต่อไปนี้

a. สถานี

i. จำนวนชานชาลาที่แสดงถึงจำนวนรถไฟที่รองรับได้ในสถานีนั้น

b. รางรถไฟรางเดี่ยวและรางคู่

i. ความยาวรางรถไฟ

c. รถไฟ

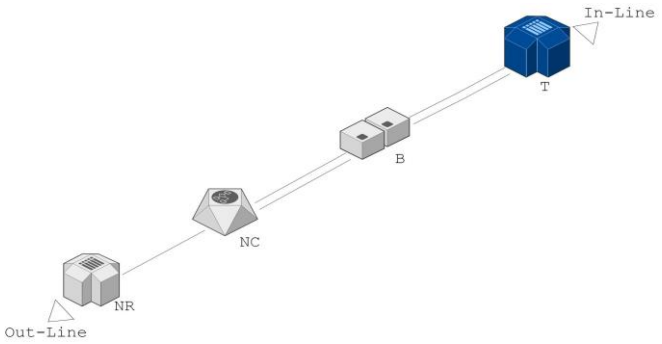
i. ความเร็วรถไฟแต่ละประเภท

3.2 การออกแบบโครงสร้างเครือข่ายทางรถไฟ

3.2.1 กำหนดข้อมูลสำหรับสร้างเครือข่ายทางรถไฟ

กำหนดข้อมูลตั้งต้น คือ ตำแหน่งสถานี ความยาวรางรถไฟ และตำแหน่งการต่อรางและสถานีในแบบจำลอง เพื่อประกอบการสร้างแบบจำลองไทมด์คัลเลอร์เพทริเน็ตสำหรับเครือข่ายทางรถไฟ พร้อมจำลองตารางรถไฟ เพื่อใช้เป็นข้อมูลนำเข้าในการสร้างแบบจำลองเครือข่ายทางรถไฟให้รองรับการทำงานและควบคุมการเดินรถไปยังสถานีต่างๆตามตารางเวลาที่ได้กำหนดไว้ โดยมีแผนภาพแสดงตำแหน่งสถานีดังรูปที่ 3.2 และ รูปที่ 3.3 แสดงตารางเวลารถไฟออกจากสถานีต่างๆ โดยเริ่มต้นที่นาทีที่ 0 โดยเวลานับต่อไปทีละ 1 นาที ต่อ 1 หน่วยยูนิตในตารางเวลา ในกรณีที่ตารางกำหนดให้รถไฟจอดที่สถานีนั้นๆ เวลาที่แสดงบนตารางจะรวมเวลาจอดรับผู้โดยสารในสถานี เป็นเวลา 1 นาทีแล้ว หากในกรณีที่รถไฟไม่จอดรับผู้โดยสารในบางสถานี ตารางเวลาจะไม่แสดงเลขกำกับ โดยจะแบ่งตารางเวลาออกเป็น 2 ตาราง คือ สำหรับรถไฟทิศทางขาล่องเข้าสถานีกรุงเทพหรือขาเข้า (IN-Direction) และสำหรับรถไฟทิศทางขาล่องออกจากสถานีกรุงเทพหรือขาออก (OUT-Direction) เช่น รถไฟขบวนที่ 99 ควรจะเข้าจอดที่ T ที่นาทีที่ 0 และควรออกจากสถานี T ที่นาทีที่ 1 จากนั้นรถไฟขบวนนี้ควรจอดที่สถานี B ที่นาทีที่ 7 และควรออกจากสถานี B ที่นาทีที่ 8 โดยรถไฟขบวนนี้จะผ่านสถานี NC โดยไม่จอดรับผู้โดยสาร และเข้าสู่สถานี NR ที่นาทีที่ 14 และออกจากสถานี NR ที่นาทีที่ 15 เพื่อไปยังสถานีปลายทาง

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY



รูปที่ 3.2 แผนภาพแสดงการเชื่อมต่อสถานีต่างๆ

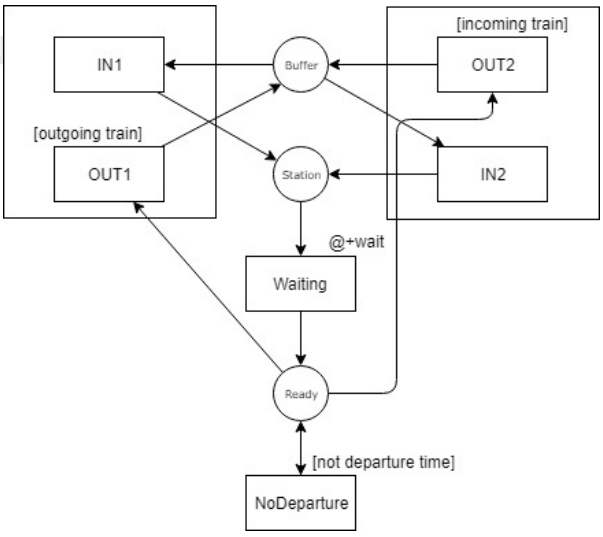
ขาเข้า					ขาออก				
สถานี\ขบวนรถ	20	112	212	402	สถานี\ขบวนรถ	99	111	211	407
NR	11	13	1	16	NR	15	25	28	22
NC			6	21	NC			23	17
B		20	11	26	B	8	18	18	12
T	24	27	20	35	T	1	11	9	3

รูปที่ 3.3 รูปแสดงตารางเวลารถไฟจำลอง

3.2.2 ออกแบบมอดูลมาตรฐานสำหรับส่วนประกอบในเครือข่ายทางรถไฟ

จากนั้นสร้างแบบจำลองไทมด์คัลเลอร์เพทริเน็ตสำหรับเครือข่ายทางรถไฟจากข้อมูลเบื้องต้น พร้อมทดสอบและปรับปรุงแบบจำลองเพื่อให้ตรงกับข้อจำกัดและความต้องการที่ถูกกำหนดไว้ จากนั้นทำการสกัดส่วนแบบจำลองไทมด์คัลเลอร์เพทริเน็ตเป็นส่วนย่อยหรือมอดูล (Module) เพื่อแสดงเป็นส่วนประกอบในเครือข่ายทางรถไฟ และใช้เป็นตัวแทนในการนำไปใช้งาน จากการศึกษาในงานวิจัยนี้สามารถกำหนดมอดูลสำหรับแบบจำลองเครือข่ายทางรถไฟได้ 2 ประเภท คือ มอดูลสถานี และ มอดูลรางรถไฟ

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University



รูปที่ 3.4 มอดูลสถานีที่สร้างมาจากไทมด์คัลเลอร์เพทริเน็ต

1) มอดูลสถานี

ได้ออกแบบไว้สำหรับใช้งานแทนแบบจำลองสถานี เพื่อควบคุมการเดินรถไฟเข้าและออกจากสถานี โดยมีตารางเวลากำกับสำหรับรถไฟที่ต้องจอดพักที่สถานี โดยมอดูลสถานีจะต้องประกอบไปด้วยส่วนสำหรับติดต่อประสานงานกับมอดูลอื่น ซึ่งใช้ในการรับรถไฟเข้ามาเพื่อส่งต่อไปยังจุดจอดรถไฟหรือชานชาลา โดยเมื่อมีรถไฟที่ถูกกำหนดให้จอดเพื่อรับส่งผู้โดยสาร ณ สถานีนั้นๆแล้ว แบบจำลองต้องมีหน้าที่ตรวจสอบเวลาออกของรถไฟให้เป็นไปตามตารางเวลาที่กำหนดด้วย อีกทั้งแบบจำลองต้องสามารถกำหนดจำนวนรถไฟที่สามารถเข้าใช้งานสถานีในช่วงเวลาหนึ่งๆพร้อมกันได้ เมื่อมีรถไฟใช้งานสถานีนั้นเต็มความจุของสถานี แบบจำลองต้องไม่อนุญาตให้มีรถไฟขบวนอื่นเข้ามาใช้งานอีกจนกว่าจะมีรถไฟถูกปล่อยออกจากแบบจำลองมอดูลสถานีเสียก่อน

โดยจากรูปที่ 3.4 แสดงมอดูลสถานีมาตรฐานที่มีรายละเอียดดังนี้

1.1) ส่วนประสานกับมอดูลอื่น โดย

a) ส่วนประสานด้านซ้ายเรียกว่า ส่วนประสานขาเข้า (IN-direction) เพื่อรับรถไฟทิศทางขาเข้าสู่สถานีและปล่อยรถไฟทิศทางขาออกออกจากสถานี โดยประกอบด้วย ทรานซิชันในการยิงโทเคนเข้ามาในมอดูล (IN1) และ ทรานซิชันในการยิงโทเคนออกจากมอดูล (OUT1) โดยมีฟังก์ชันตรวจสอบประเภทรถไฟที่เป็นขาออก (isHeadOut(train) โดย train คือเลขรถไฟที่ต้องการตรวจสอบ) เท่านั้น จึงจะยิงออกจากมอดูลได้โดยจะถูกกำกับไว้บนการ์ดอินสคริปชัน

b) ส่วนประสานด้านขวาเรียกว่า ส่วนประสานขาออก (OUT-direction) เพื่อรับรถไฟทิศทางขาออกเข้าสู่สถานีและปล่อยรถไฟทิศทางขาเข้าออกจากสถานี ประกอบด้วย ทรานซิชันในการยิงโทเคนออกจากมอดูล (OUT2) และ ทรานซิชันในการยิงโทเคนเข้ามาในมอดูล (IN2) โดยจะมีฟังก์ชันตรวจสอบประเภทรถไฟที่เป็นขาเข้า (isHeadIn(train) โดย train คือเลขรถไฟที่ต้องการตรวจสอบ) เท่านั้นจึงจะยิงออกจากมอดูลได้โดยจะกำกับไว้บนการ์ดอินสคริปชัน

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

1.2) ส่วนพักโทเคนที่เข้ามาในมอดูลสถานี ใช้เพลสที่มีชื่อตามชื่อสถานีและไม่ซ้ำกับเพลสอื่นๆ (Station) ใช้เป็นที่พักโทเคนหรือรถไฟที่ถูกยิงเข้ามาในสถานี

1.3) ส่วนกำหนดปริมาณรถไฟในสถานี โดยใช้ เพลส (Buffer) ที่ระบุจำนวนโทเคนเริ่มต้นซึ่งจะมีเป็นคนละชนิดกับโทเคนที่เป็นตัวแทนของรถไฟตามจำนวนชานชาลาที่สถานีนั้นๆมี เพื่อรอบรับรถไฟที่เข้ามาจอดรอรับผู้โดยสาร

1.4) ส่วนหน่วงเวลาในสถานี โดยใช้ค่าที่กำหนดบนไทม์อินสคริปชัน (@+ waiting) บนทรานซิชัน (Waiting) เป็นตัวแทนเวลาในการจอดรถไฟ โดยจะใช้ 1 หน่วยยูนิต แทนการจอดรถไฟ 1 นาที

1.5) ส่วนเตรียมพร้อมปล่อยรถออกจากสถานี ใช้ทรานซิชัน (NoDeparture) และการ์ดฟังก์ชันบนทรานซิชันตรวจสอบเวลาสำหรับการปล่อยรถออกจากสถานี (isDeparetureTime(train, [ชื่อสถานี]) โดย train คือเลขรถไฟที่ต้องการตรวจสอบและชื่อสถานีที่ต้องการตรวจสอบ) และประเภทรถไฟ

1.6) หากยังไม่ถึงเวลาปล่อยรถได้ การ์ดฟังก์ชันบนทรานซิชัน NoDeparture จะมีค่าความจริงเป็นจริง ทำให้ทรานซิชันนี้ถูกเอ็นเนเบิล เมื่อโทเคนถูกยิงโดยทรานซิชันนี้ เวลาโทเคนไทม์แสตมป์จะถูกเพิ่มขึ้น 1 หน่วย เนื่องจากทรานซิชันนี้มีค่าไทมด์อินสคริปชันเป็น 1 กำกับไว้

1.7) หากถึงเวลาปล่อยรถและรางที่ต่อกับสถานีมีสถานะว่าง การ์ดฟังก์ชันบนทรานซิชัน OUT จะมีค่าความจริงเป็นจริง จึงจะตรวจสอบชนิดของโทเคนที่แสดงถึงประเภทรถไฟเป็นลำดับถัดไป

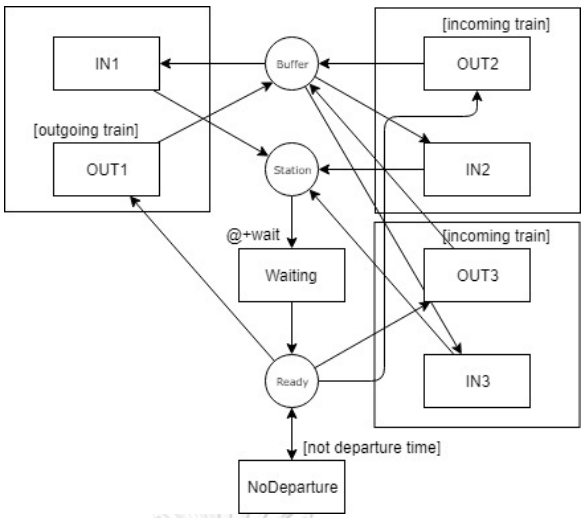
a) รถไฟขาเข้า ฟังก์ชันสำหรับตรวจสอบรถไฟขาเข้าบนการ์ดของทรานซิชัน OUT2 จะมีค่าความจริงเป็นจริง

b) รถไฟขาออก ฟังก์ชันสำหรับตรวจสอบรถไฟขาออกบนการ์ดของทรานซิชัน OUT1 จะมีค่าความจริงเป็นจริง

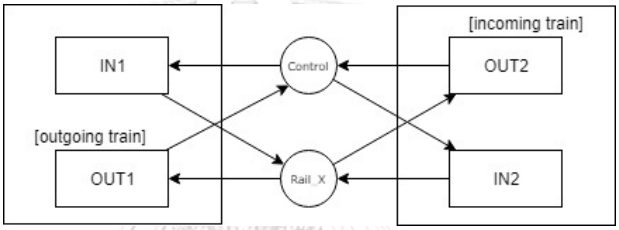
เมื่อทั้งสองฟังก์ชันมีค่าความจริงเป็นจริง จะทำให้ทรานซิชันนั้นๆ มีสถานะเป็นเอ็นเนเบิล โทเคนจะถูกยิงไปด้วยทรานซิชันนั้นๆ เพื่อแสดงการเคลื่อนรถไฟออกจากสถานี

มอดูลสถานีสามารถมีส่วนประสานมอดูลอื่นได้มากกว่า 1 ในแต่ละด้าน ตามจำนวนรางที่เข้าสู่สถานี เช่น กรณีเป็นรางคู่สำหรับขาออก จะมีจำนวนส่วนต่อประสานมอดูลอื่นด้านขวา 2 ส่วน ดังตัวอย่างในรูปที่ 3.5

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University



รูปที่ 3.5 มอดูลสถานีที่มีรางขาเข้า 1 ราง และ รางขาออก 2 ราง



รูปที่ 3.6 มอดูลรางที่สร้างมาจากไทมด์คัลเลอร์เพทริเน็ต

2) มอดูลรางรถไฟ

เป็นมอดูลของแบบจำลองที่ออกแบบสำหรับแทนส่วนประกอบรางรถไฟของแบบจำลองเครือข่ายทางรถไฟ โดยแบบจำลองมอดูลต้องมีคุณสมบัติการใช้ประสานงานกับมอดูลอื่นเพื่อรองรับรถไฟเข้าและรถไฟออก ในการใช้งานช่วงเวลาหนึ่งๆ ดังเช่นมอดูลสถานี แต่โดยปกติแล้วรางรถไฟจะถูกหนดให้ในหนึ่งช่วงเวลาใดๆจะมีรถไฟเข้าใช้งานได้เพียงหนึ่งขบวนเท่านั้น เพื่อป้องกันเหตุที่อาจจะเกิดขึ้น ฉะนั้นแบบจำลองมอดูลรางรถไฟต้องมีความสามารถในการใช้งานต้องเป็นไปตามกฎเบื้องต้น เมื่อมีรถไฟขบวนใดใช้งานแบบจำลองมอดูลรางรถไฟนี้อยู่ ต้องไม่อนุญาตให้ขบวนอื่นๆเข้ามาใช้งานส่วนหนึ่งส่วนใดของแบบจำลองมอดูลนี้อีกจนกว่ารถไฟขบวนนั้นจะเคลื่อนออกจากแบบจำลองมอดูรางรถไฟแล้วท่านั้น

โดยมอดูลรางรถไฟมาตรฐานจะเป็นดังรูปที่ 3.6 มีรายละเอียดดังต่อไปนี้

2.1) ส่วนประสานกับมอดูลอื่น โดย

a) ส่วนประสานด้านซ้ายมือเรียกว่า ส่วนประสานขาเข้า (IN-direction) เพื่อรับรถไฟขาเข้าสู่รางและปล่อยรถไฟขาออกออกจากราง ประกอบด้วยทรานซิชันในการยิงโทเคนเข้ามาในมอดูล (IN1) และ ทรานซิชันในการยิงโทเคนออกจากมอดูล (OUT1) โดยมีฟังก์ชันตรวจสอบประเภทรถไฟที่เป็น

ขาออก (isHeadOut(train) โดย train คือเลขรถไฟที่ต้องการตรวจสอบ) เท่านั้นจึงจะยิงออกจากมอดูลได้บนการ์ดอินสคริปชัน

b) ส่วนประสานด้านขวามือเรียกว่า ส่วนประสานขาออก (OUT-direction) เพื่อรับรถไฟออกเข้าสู่รางและปล่อยรถไฟขาเข้าออกจากราง ประกอบด้วยทรานซิชันในการยิงโทเคนออกจากมอดูล (OUT2) และทรานซิชันในการยิงโทเคนเข้ามาในมอดูล (IN2) โดยจะมีฟังก์ชันตรวจสอบประเภทรถไฟที่เป็นขาเข้า (isHeadIn(train) โดย train คือเลขรถไฟที่ต้องการตรวจสอบ) เท่านั้นจึงจะยิงออกจากมอดูลได้บนการ์ดอินสคริปชัน

2.2) ส่วนพักโทเคนที่เข้ามาในมอดูลรางรถไฟ ใช้เพลสที่มีชื่อเรียกเป็นชื่อรางรถไฟนั้นและไม่ซ้ำกับเพลสอื่นๆ (Rail_X) ใช้แสดงสถานะของรางรถไฟนั้นว่ากำลังถูกใช้งาน

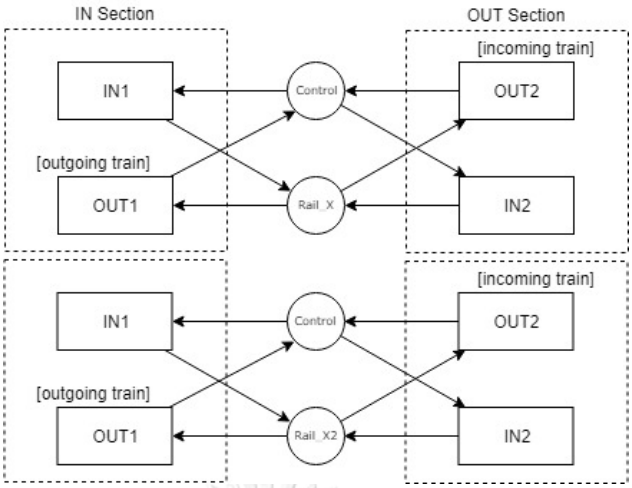
2.3) ส่วนกำหนดปริมาณรถไฟบนราง โดยใช้ เพลส (Control) ที่ระบุจำนวนโทเคนเริ่มต้น 1 ตัว ซึ่งเป็นคนละชนิดกับโทเคนที่เป็นตัวแทนของขบวนรถไฟ เพื่อกำหนดการใช้งานรางรถไฟนั้นๆ สามารถใช้งานรถไฟได้ทีละหนึ่งขบวนเท่านั้น

2.4) ส่วนหน่วงเวลาบนรางรถไฟ โดยใช้ค่าที่กำหนดในไทมด์อินสคริปชันบนทรานซิชันออก (OUT1 และ OUT2) โดยทั้ง 2 จะมีค่าเท่ากัน เพื่อแสดงว่าที่รถไฟใช้ในการเดินทางผ่านรางนั้นๆ โดยเมื่อทรานซิชันถูกยิง โทเคนที่เป็นตัวแทนของรถไฟ จะถูกเพิ่มค่าไทม์แสตมป์ตามจำนวนที่ระบุไว้

จำกัดความ ขาเข้า (IN-direction) ใช้บอกทิศทางการวิ่งของรถไฟเข้าสู่สถานีศูนย์กลางหรือสถานีกรุงเทพ โดยในที่นี้คือทิศทางรถไฟที่วิ่งจากซ้ายไปขวา

จำกัดความ ขาออก (OUT-direction) ใช้บอกทิศทางการวิ่งของรถไฟออกจากสถานีศูนย์กลางหรือสถานีกรุงเทพ โดยในที่นี้คือทิศทางที่รถไฟวิ่งจากขวาไปซ้าย

ในกรณีมอดูลรางประเภทรางคู่จะมีลักษณะคล้ายคลึงกับมอดูลรางรถไฟมาตรฐานหรือรางเดี่ยว โดยจะกำหนดให้มอดูลรางรถไฟ 2 รางขนานกันไป โดยไม่มีส่วนใดเชื่อมต่อกัน แต่ในภาวะปกติรางใดรางหนึ่งจะถูกกำหนดให้สำหรับรถไฟต่างขาล่องวิ่งได้ใช้งานเท่านั้น (One-way) โดยจะอธิบายเพิ่มเติมในหัวข้อ 3.2.5 ต่อไป ตัวอย่างมอดูลรถไฟรางคู่ดังรูปที่ 3.7



รูปที่ 3.7 แสดงมอดูลรางรถไฟประเภทรางคู่

ตารางสรุปส่วนประกอบมอดูลดังแสดงในตารางที่ 3.1

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

ตารางที่ 3.1 ตารางส่วนประกอบมอดูลสถานีและมอดูลรางรถไฟ

ส่วนประกอบ \ มอดูล	สถานี	รางรถไฟ
ส่วนต่อประสาน - ขาเข้า (IN-direction)	ทรานซิชันเข้า (IN1) และ ทรานซิชันออก (OUT1) ที่มีฟังก์ชันตรวจสอบรถไฟที่ เป็นขบวนขาออกเท่านั้น	ทรานซิชันเข้า (IN1) และทรานซิชันออก (OUT1) ที่มีฟังก์ชันตรวจสอบรถไฟที่เป็นขบวนขาออกเท่านั้น
ส่วนต่อประสาน - ขาออก (OUT-direction)	ทรานซิชันเข้า (IN2) และ ทรานซิชันออก (OUT2) ที่มีฟังก์ชันตรวจสอบรถไฟที่ เป็นขบวนขาเข้าเท่านั้น	ทรานซิชันเข้า (IN2) และทรานซิชันออก (OUT2) ที่มีฟังก์ชันตรวจสอบรถไฟที่เป็นขบวนขาเข้าเท่านั้น
ส่วนพักโทเคน	เพลสที่เป็นชื่อสถานี (Station)	เพลสที่เป็นชื่อราง (Rail_X)
ส่วนกำหนดปริมาณรถไฟ	เพลส (Buffer) พร้อมโทเคนเริ่มต้นตามจำนวนชานชาลาของสถานีนั้น	เพลส (Control) พร้อมโทเคนเริ่มต้น จำนวน 1 ตัว
ส่วนหน่วงเวลาในมอดูล	กำหนดค่าบนไทมด์อินสคริปชัน (@+wait) บนทรานซิชัน (Waiting) สำหรับดีเลย์ เพื่อแสดงเวลาในการหยุดรอผู้โดยสาร	กำหนดค่าบนไทมด์อินสคริปชันบนทรานซิชันออก (OUT1 และ OUT2)
ส่วนเตรียมพร้อมปล่อยรถออกจากมอดูล	เพลส (Read) ที่เชื่อมต่อกับทรานซิชัน (NoDeparture) ที่มีฟังก์ชันตรวจสอบว่าถึงเวลาปล่อยรถไฟหรือไม่ ถ้าไม่ ดีเลย์จะถูกบวกเพิ่มขึ้น 1 หน่วย	-

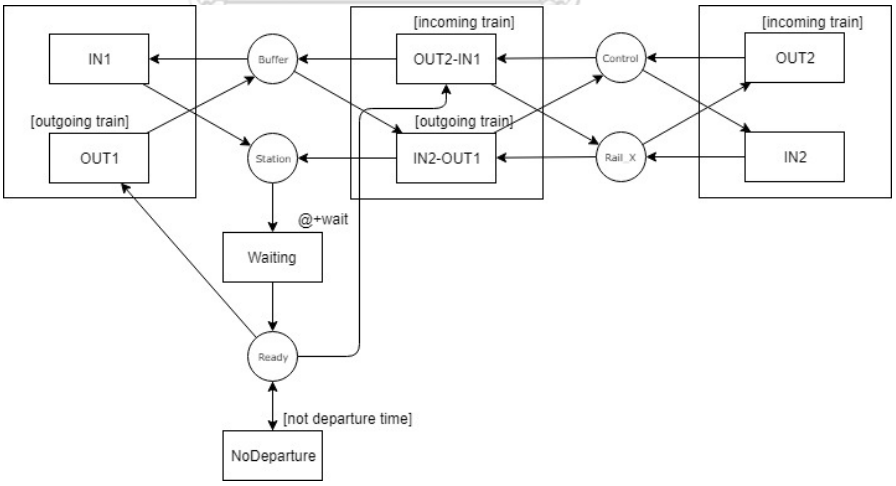
จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

3.2.3 กำหนดกฎการต่อประสานด้วยวิธีการซ้อนทับ (Superimposing Rules)

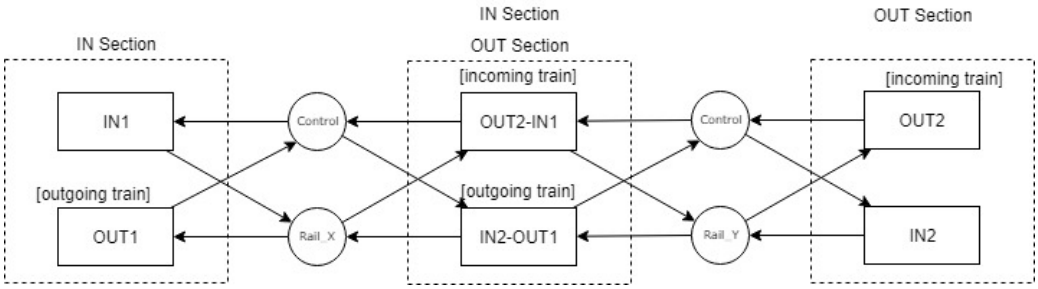
เนื่องจากกฎการใช้งานแบบจำลองเพทริเน็ตทุกประเภทได้กำหนดว่า เพลสต้องเชื่อมต่อกับทรานซิชันเท่านั้นโดยใช้เส้นเชื่อมที่ระบุทิศทาง จากรูปแบบมอดูลที่ระบุไว้ก่อนหน้า เราไม่สามารถนำมอดูล 2 มอดูลใดๆเชื่อมต่อกันด้วยเส้นเชื่อมได้ในทันที เนื่องจากปลายทั้งสองด้านของมอดูลใดๆ คือ ทรานซิชัน จึงต้องกำหนดกฎเพิ่มเติมในการต่อประสานมอดูล โดยใช้วิธีการซ้อนทับ หรือ Superimposing มีรายละเอียดดังต่อไปนี้

1) เมื่อต้องการเชื่อมต่อมอดูล 2 มอดูลเข้าด้วยกัน จะใช้วิธิการซ้อนทับในส่วนประสานที่ประกอบด้วยทรานซิชัน IN และ ทรานซิชัน OUT ที่อยู่ขอบซ้ายและขวาของมอดูลเท่านั้น โดยการซ้อนทับจะทำทีละส่วนประสานเท่านั้น
2) เมื่อซ้อนทับกันสำเร็จ ต้องกำหนดชื่อทรานซิชันใหม่โดยใช้ชื่อของทรานซิชันเดิมทั้ง 2 ทรานซิชันเรียงต่อกัน เพื่อไม่ให้มีชื่อซ้ำกันในแบบจำลอง
3) ทรานซิชันเดิมใดที่ถูกซ้อนทับและมีการ์ดอินสคริปชันกำกับไว้ ต้องคงการ์ดอินสคริปชันนั้นไว้เช่นเดิม

ตัวอย่างหลังการซ้อนทับกันของ 2 มอดูล สถานี Station และ รางรถไฟ Rail_X โดยรางรถไฟ Rail_X ถูกกำหนดให้ต่อประสานกับสถานี Station ในส่วนประสานขาออกด้านขวา ดังรูปที่ 3.8



รูปที่ 3.8 ตัวอย่างการประสานมอดูลสถานีและมอดูลรางรถไฟด้วยวิธีการซ้อนทับ



รูปที่ 3.9 ตัวอย่างการประสานมอดูลรางรถไฟและรางรถไฟด้วยวิธีการซ้อนทับ

จากรูปที่ 3.9 แสดงการซ้อนทับของมอดูลรางรถไฟ 2 มอดูลตามกฎการซ้อนทับปกติ

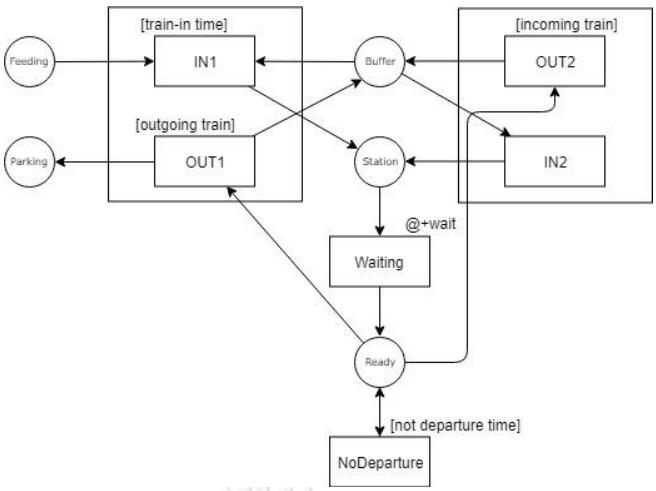
3.2.4 กำหนดกฎการสร้างที่พักรถไฟก่อนเข้าสถานีต้นทาง

เนื่องจากการใช้งานตารางรถไฟกำกับการเดินรถในเวลาต่างๆ โทเคนหรือรถไฟทั้งหมดไม่สามารถถูกกำหนดให้เริ่มต้นที่สถานีต้นทางเมื่อเริ่มการจำลอง และเมื่อรถไฟออกจากสถานีสุดท้ายที่แสดงในแบบจำลองต้องมีเพลสเพิ่มเติมสำหรับพักโทเคนหรือรถไฟ เพื่อไม่ให้กระทบกับการจัดการในสถานีนั้น โดยมีรายละเอียดดังนี้

1) สำหรับสถานีต้นทางทิศทางขาเข้า
   a) เพิ่มเพลส Feeding โดยเชื่อมต่อกับทรานซิชัน IN1 ที่ส่วนประสานขาเข้าของสถานี โดยเพลสนี้จะมีโทเคนเริ่มต้นที่เป็นตัวแทนของรถไฟที่มีทิศทางขาเข้าทุกขบวน
   b) เพิ่มการ์ดฟังก์ชันสำหรับตรวจสอบเวลารถไฟเข้าสถานีที่ทรานซิชัน IN1 เพื่ออนุญาตเฉพาะรถไฟที่ถึงเวลาเข้าสถานีแล้วเท่านั้นจึงจะเอ็นเนเบิลทรานซิชัน และทำการยิงเพื่อเข้าสู่สถานี
   c) เพิ่มเพลส Parking สำหรับพักโทเคนหรือรถไฟที่มีทิศทางขาออกที่วิ่งออกจากสถานีนี้ซึ่งเป็นสถานีปลายทางสำหรับแบบจำลองนี้แล้ว โดยเชื่อมต่อกับทรานซิชัน OUT1

ตัวอย่างการเพิ่มเพลสสำหรับสถานีต้นทางขาเข้าดังรูปที่ 3.10

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University



รูปที่ 3.10 การเพิ่มเพลสสำหรับสถานีต้นทางขาเข้า

2) สำหรับสถานีต้นทางทิศทางขาออก

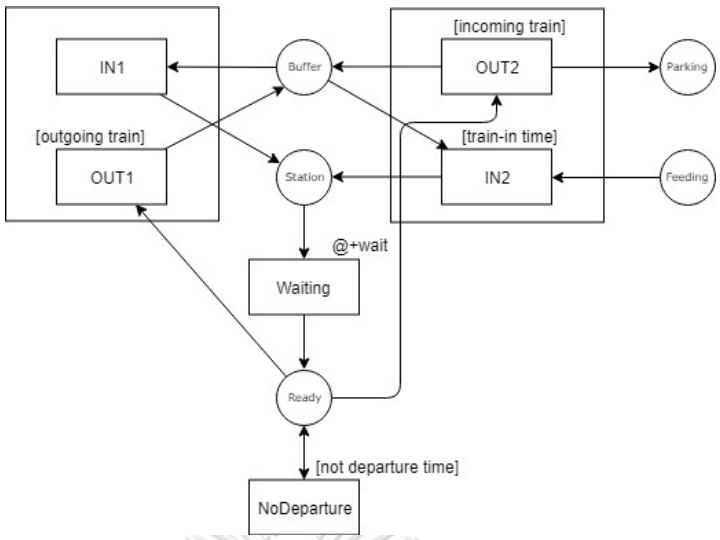
   a) เพิ่มเพลส Feeding โดยเชื่อมต่อกับทรานซิชัน IN2 ที่ส่วนประสานขาออกของสถานี โดยเพลสนี้จะมีโทเคนเริ่มต้นที่เป็นตัวแทนของรถไฟที่มีทิศทางขาออกทุกขบวน

   b) เพิ่มการ์ดฟังก์ชันสำหรับตรวจสอบเวลารถไฟเข้าสถานีที่ทรานซิชัน IN2 เพื่ออนุญาตเฉพาะรถไฟที่ถึงเวลาเข้าสถานีแล้วเท่านั้นจึงจะเอ็นเนเบิลทรานซิชัน และทำการยิงเพื่อเข้าสู่สถานี

   c) เพิ่มเพลส Parking สำหรับพักโทเคนหรือรถไฟที่มีทิศทางขาเข้าที่วิ่งออกจากสถานีนี้ซึ่งเป็นสถานีปลายทางสำหรับแบบจำลองนี้แล้ว โดยเชื่อมต่อกับทรานซิชัน OUT2

ตัวอย่างการเพิ่มเพลสสำหรับสถานีต้นทางขาออกดังรูปที่ 3.11

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY



รูปที่ 3.11 การเพิ่มเพลสสำหรับสถานีต้นทางขาออก

3) สำหรับสถานีชุมทาง

สถานีชุมทางมีลักษณะสถานีที่เป็นจุดรวมหรือแยกเส้นทางรถไฟตั้งแต่ 2 เส้นทางขึ้นไป โดยงานวิจัยจะใช้ตัวอย่างการออกแบบสถานีชุมทางที่รวมเส้นทาง 2 เส้นทาง โดยเส้นทางที่แสดงในแบบจำลองจะเรียกว่า เส้นทางหลัก และอีกเส้นทางหนึ่งที่ไม่แสดงในแบบจำลองเรียกว่า เส้นทางรอง เมื่อรถไฟเข้าสู่สถานีจากเส้นทางรองจะเสมือนรถไฟวิ่งเข้าที่สถานีต้นทาง ในทำนองเดียวกันเมื่อรถไฟออกจากสถานีชุมทางโดยใช้เส้นทางรองจะเสมือนรถไฟวิ่งออกที่สถานีปลายทาง โดยมีกฎการเพิ่มส่วนประกอบดังรูปที่ 3.12 และรายละเอียดดังต่อไปนี้

a) เพิ่มทรานซิชัน IN3 สำหรับเชื่อมต่อกับเพลสสถานีและเพลส Buffer เพื่อรองรับการยิงโทเคนออกจากเพลสใหม่ของเส้นทางรอง ในส่วนประสานใหม่ (New Section) โดยจะใช้การ์ดฟังก์ชันสำหรับตรวจสอบเวลารถไฟเข้าที่สถานีเช่นเดิม

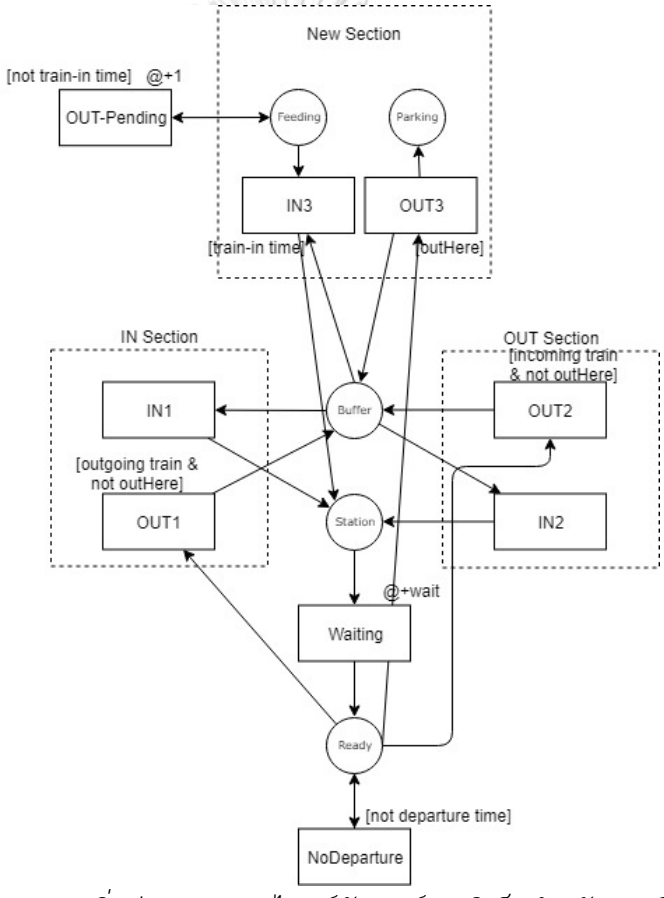
b) เพิ่มทรานซิชัน OUT3 สำหรับเชื่อมต่อกับเพลสสถานีและเพลส Buffer เพื่อรองรับการยิงโทเคนออกจากเพลส Ready ของสถานีไปยังเพลสใหม่ในเส้นทางรองโดยจะใช้การ์ดฟังก์ชันสำหรับตรวจสอบเลขขบวนรถไฟที่จะถูกยิ่งออกในเส้นทางรองนี้

c) เพิ่มเพลส Feeding โดยเชื่อมต่อกับทรานซิชัน IN3 โดยเพลสนี้จะมีโทเคนเริ่มต้นที่เป็นตัวแทนของรถไฟที่ใช้เส้นทางรองทุกขบวน

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

d) เพิ่มเพลส Parking สำหรับพักโทเคนหรือรถไฟที่มีถูกยิงออกจากสถานีในเส้นทางรองซึ่งถือเป็นสถานีปลายทางสำหรับแบบจำลองนี้แล้ว โดยเชื่อมต่อกับทรานซิชัน OUT3

e) เพิ่มการ์ดฟังก์ชันสำหรับเลขขบวนรถไฟจะสามารถออกในเส้นทางปกติได้ก็ต่อเมื่อเลขขบวนนั้นไม่ได้กำหนดให้วิ่งออกในเส้นทางรอง โดยจะใช้การ์ดฟังก์ชัน “not outHere” เพิ่มเติมที่การ์ดอินสคริปชันเดิมของทุกทรานซิชัน OUT ในทุกส่วนต่อประสานเดิม โดยในที่นี้คือ OUT1 และ OUT2



รูปที่ 3.12 การเพิ่มส่วนประกอบไทมด์คัลเลอร์เพทริเน็ตสำหรับสถานีชุมทาง

3.2.5 กำหนดกฎการเพิ่มเติมสำหรับการสร้างแบบจำลองเครือข่ายทางรถไฟ

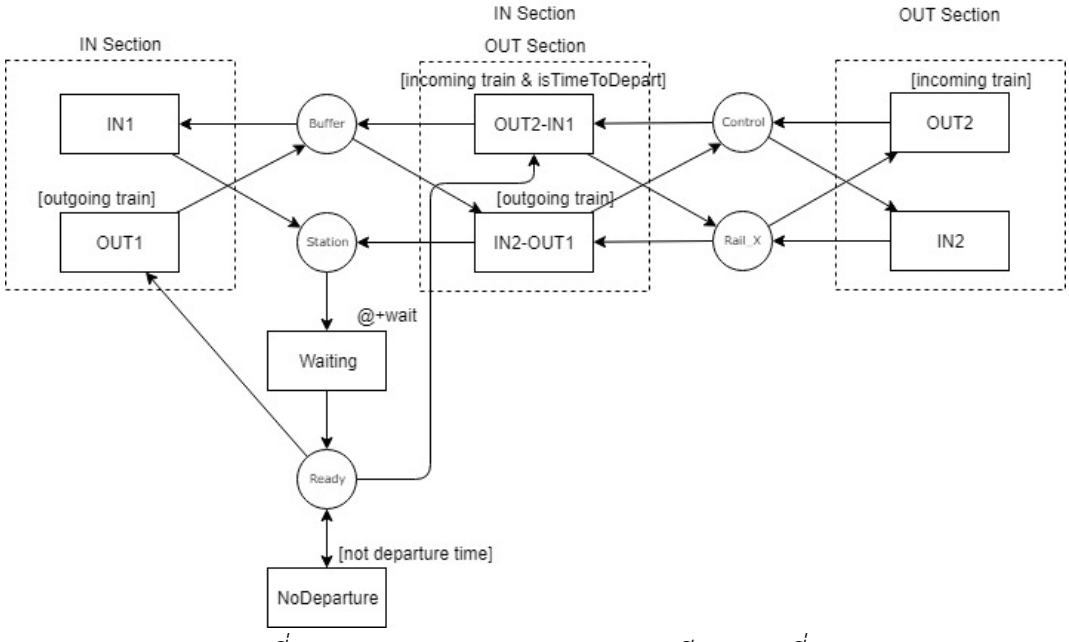
โดยหลังจากนำมอดูลมาตรฐานประกอบกันเพื่อสร้างเครือข่ายทางรถไฟ โดยมีตารางเวลารถไฟเข้ามาเกี่ยวข้องพบว่า ส่วนประกอบมอดูลบางส่วนต้องถูกปรับแต่ง เพื่อให้ได้ผลลัพธ์เป็นไปตามเงื่อนไขที่ระบุไว้ดังนี้

1) กฎการสร้างแบบจำลองเครือข่ายทางรถไฟ

แบบจำลองเครือข่ายทางรถไฟที่มีสถานีและรางเป็นส่วนประกอบนี้ถูกกำหนดให้สถานีต้องเชื่อมต่อกับรางรถไฟ โดยเมื่อผู้ใช้สร้างมอดูลสถานีแล้ว มอดูลที่อยู่ติดกันต้องเป็นมอดูลรางรถไฟเท่านั้น ยกเว้นมอดูลรางรถไฟสามารถเชื่อมต่อกับมอดูลรางรถไฟอื่นๆได้ จะทำให้ในหนึ่งช่วงเวลาใดๆ สามารถมีรถไฟใช้งานรางระหว่างสถานีได้มากกว่า 1 ขบวน แต่มีข้อควรระวังการใช้งานเมื่อผู้ใช้ออกแบบให้มีมอดูลรางรถไฟเชื่อมต่อกับมอดูลรางรถไฟอื่นๆ อาจทำให้เกิดปัญหาการติดตายได้ หากปล่อยรถไฟออกจากสถานีผิดพลาด ข้อกำหนดในการออกแบบเครือข่ายทางรถไฟในภาวะปกติจึงใช้เพียงมอดูลรางรถไฟมอดูลเดียวเท่านั้นในการเชื่อมต่อสถานี โดยมอดูลนั้นมีพารามิเตอร์สำหรับกำหนดความยาวรางตามระยะระหว่างสถานีไว้แล้ว

2) กฎการปล่อยรถไฟออกจากสถานี

โดยกฎการเดินรถทั่วไปได้ห้ามรถไฟออกจากสถานีก่อนเวลาที่กำหนดไว้ตามตารางเวลา ทำให้การต่อประสานด้วยวิธีการซ้อนทับเบื้องต้นจะเห็นได้ว่าไม่เพียงสำหรับการตรวจสอบเวลารถไฟว่าต้องออกจากสถานีแล้วหรือไม่ เพราะการ์ดอินสคริปชันนั้นมีเพียงการตรวจสอบชนิดของทิศทางขาล่องรถไฟเท่านั้น ฉะนั้นมอดูลสถานีเมื่อถูกประกอบเข้ากับรางใดๆแล้วต้องเพิ่มการ์ดอินสคริปชันของทรานซิชัน OUT ของส่วนประสานนั้นๆโดยเพิ่มฟังก์ชันสำหรับตรวจสอบเวลา isDeparturetime(train,[ชื่อสถานี]) ดังรูปที่ 3.13 ด้วย

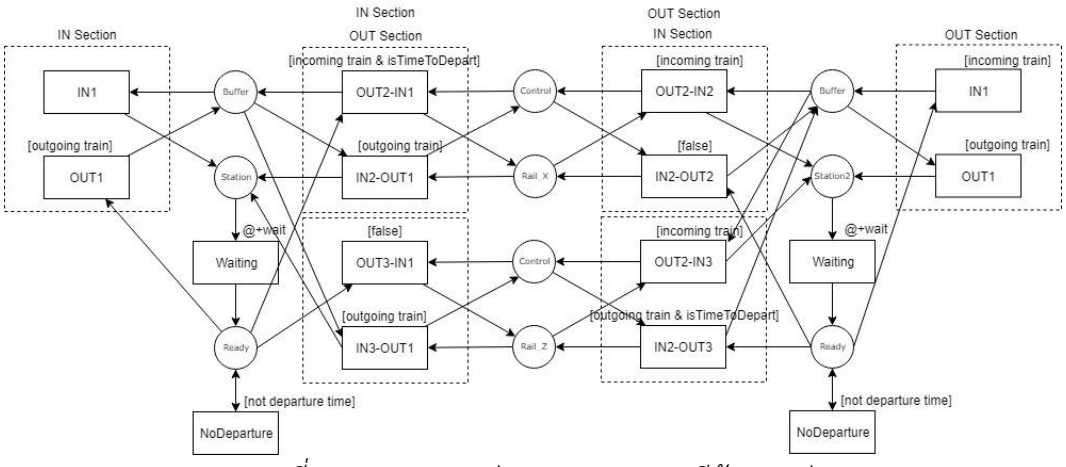


รูปที่ 3.13 แสดงการประกอบมอดูลสถานีและรางเดี่ยว

จากรูปที่ 3.13 เมื่อสถานี Station ถูกประกอบเข้ากับราง Rail_X แล้ว ส่วน OUT section ของมอดูล Station จะถูกซ้อนทับกับส่วน IN section ตามกฎการต่อประสานข้อ 3.2.3 โดยการ์ดอินสคริปชันสำหรับตรวจสอบรถไฟขาเข้าจะถูกคงไว้ที่ทรานซิชัน OUT2-IN1 และเพิ่มการตรวจสอบเวลาออกจากสถานีของรถไฟขบวนนั้นๆเข้าไปด้วย ส่วนทรานซิชัน IN2-OUT1 ยังคงการ์ดอินสคริปชันไว้ตามเดิม

3) กฎการควบคุมการใช้งานรางประเภทรางคู่ระหว่างสถานี

รถไฟทิศทางขาล่องใดๆจะถูกกำหนดให้ใช้งานรางหนึ่งรางใดบนรางคู่ระหว่างสถานีนั้นๆ เท่านั้น โดยข้อกำหนดดังกล่าว ทำให้รถไฟทิศทางขาล่องหนึ่งถูกอนุญาตให้ใช้งานรางรถไฟรางใดรางหนึ่งเท่านั้นในกรณีระหว่างสถานีเป็นรางคู่ ถ้าหากรถไฟทิศทางขาเข้าได้รับอนุญาตให้ใช้รางที่ 1 รถไฟทิศทางขาออกจะถูกกำหนดให้ใช้รางที่ 2 เสมอ โดยจะไม่มีการสลับรางวิ่งในเวลาปกติ ยกเว้นมีเหตุสุดวิสัยที่จำเป็นต้องใช้รางร่วมกันเท่านั้น จากข้อกำหนดข้อนี้ ทำให้เงื่อนไขบนการ์ดอินสคริปชันส่วนหนึ่งของส่วนต่อประสานของมอดูลรางรถไฟที่กำหนดให้เป็นรางคู่ต้องถูกแก้ไข



รูปที่ 3.14 แสดงการต่อประสาน 2 สถานีด้วยรางคู่

จากรูปที่ 3.14 แสดงแบบจำลองหลังจากการต่อประสานสถานีด้วยรางคู่ที่มี Rail_X (รางบน) ถูกกำหนดให้ใช้งานสำหรับรถไฟขาเข้าและ Rail_Z ถูกกำหนดให้ใช้งานสำหรับรถไฟขาออกเท่านั้น ทำให้เมื่อมอดูลสถานี Station, Rail_X, Rail_Z และ Station2 ถูกต่อประสานเข้าด้วยกันต้องทำการแก้ไขการ์ดอินสคริปชันดังนี้

a. ส่วนประสานขาออกของสถานีต้นทาง กำหนดให้การ์ดอินสคริปชันบนทรานซิชัน OUT ของส่วนที่เชื่อมต่อกับรางที่กำหนดสำหรับรถไฟขาออกเท่านั้น ต้องเปลี่ยนเป็น “false” เพื่อไม่ให้รถไฟขาเข้าหรือโทเคนถูกยิงไปยังรางนั้น โดยจากรูปที่ 32 การ์ดอินสคริปชันบนทรานซิชัน OUT3-IN1 ซึ่งเป็นทรานซิชัน OUT ของส่วนประสานขาออกของสถานี Station ที่ถูกต่อประสานกับรางล่าง

(Rail_Z) ที่ถูกกำหนดให้ใช้งานสำหรับรถไฟขาออกเท่านั้น จะต้องถูกเปลี่ยนเป็น [false]

b. ส่วนประสานขาเข้าของสถานีปลายทาง กำหนดให้การ์ดอินสคริปชันบนทรานซิชัน OUT ของส่วนที่เชื่อมต่อกับรางที่กำหนดสำหรับรถไฟขาเข้าเท่านั้น ต้องเปลี่ยนเป็น “false” เพื่อไม่ให้รถไฟขาออกหรือโทเคนถูกยิงไปยังรางนั้น โดยจากรูปที่ 32 การ์ดอินสคริปชันบนทรานซิชัน IN2-OUT2 ซึ่งเป็นทรานซิชัน OUT ของส่วนประสานขาเข้าของสถานี Station2 ที่ถูกต่อประสานกับรางบน (Rail_X) ที่ถูกกำหนดให้ใช้งานสำหรับรถไฟขาเข้าเท่านั้น จะต้องถูกเปลี่ยนเป็น [false]

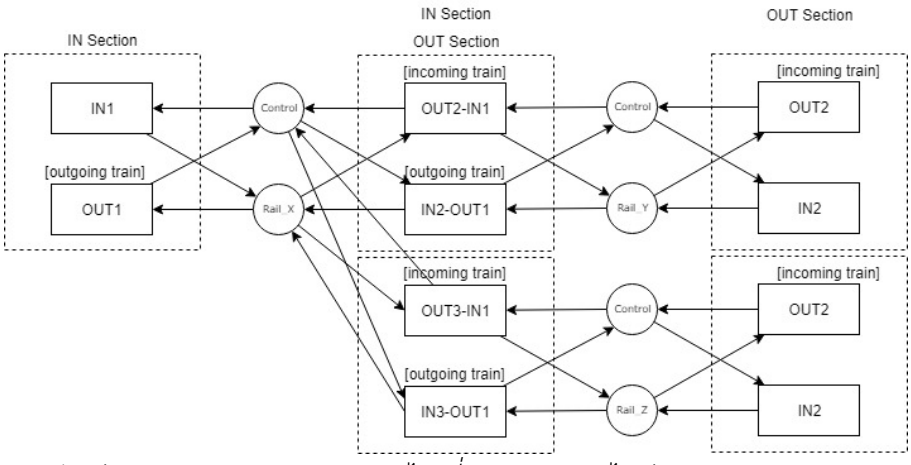
4) กฎการต่อประสานด้วยวิธีการซ้อนทับรางต่างชนิดกัน

ในกรณีการประสานรางรถไฟต่างชนิดกันจะเสมือนการสร้างจุดแยกรางหรือจุดรวมราง โดยมีการกำหนดกฎเพิ่มเติมเพื่อความสมบูรณ์ของแบบจำลองดังนี้

4.1) การซ้อนทับรางเดี่ยวและรางคู่

4.1.1) นำส่วนประสานขอออกของรางเดี่ยวซ้อนทับส่วนประสานขาเข้าของรางบนของรางคู่โดยใช้กฎการต่อประสานด้วยวิธีการซ้อนทับข้อ 3.2.3

4.1.2) สร้างเส้นเชื่อมระหว่างเพลสของสถานี และ Control เข้ากับรางล่างของรางคู่ตามการออกแบบของมอดูลรางรถไฟประเภทรางเดี่ยว ดังตัวอย่างรูปที่ 3.15

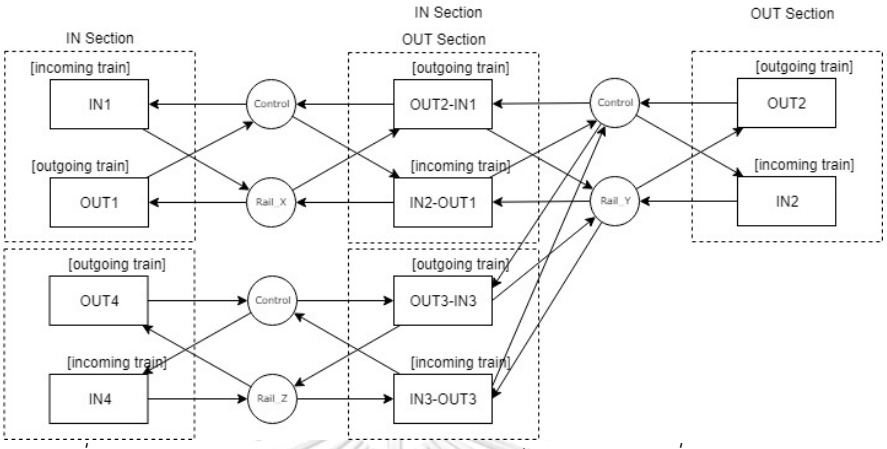


รูปที่ 3.15 ตัวอย่างการประสานมอดูลรางรถไฟเดี่ยวและรางรถไฟคู่ (จุดแยกราง)

4.2) การซ้อนทับรางคู่และรางเดี่ยว

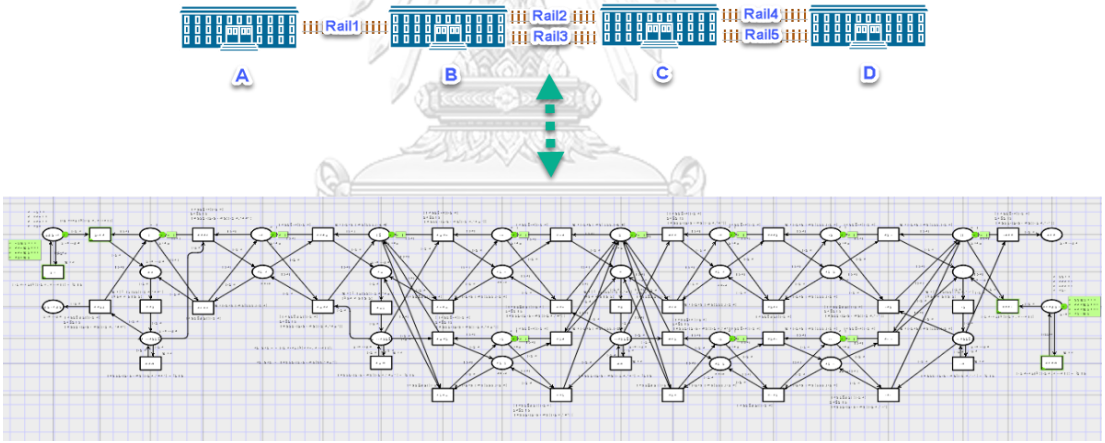
4.2.1) นำส่วนประสานขอออกรางบนของมอดูลรางคู่ซ้อนทับส่วนประสานขาเข้าของมอดูลรางเดี่ยวโดยใช้กฎการต่อประสานด้วยวิธีการซ้อนทับข้อ 3.2.3

4.2.2) สร้างเส้นเชื่อมระหว่างเพลสของสถานี และ Control เข้ากับรางล่างของรางคู่ตามการออกแบบของมอดูลรางรถไฟประเภทรางเดี่ยว ดังตัวอย่างรูปที่ 3.16



รูปที่ 3.16 ตัวอย่างการประสานมอดูลรางรถไฟคู่และรางเดี่ยว (จุดรวมราง)

จากกฎการแปลงที่ถูกเพิ่มเข้ามา ทำให้สามารถแปลงแบบจำลองเครือข่ายทางรถไฟไปเป็นไทมด์คัลเลอร์เพทริเน็ตดังแบบจำลองตัวอย่างในรูปที่ 3.17



รูปที่ 3.17 ตัวอย่างหลังการแปลงแบบจำลองเครือข่ายทางรถไฟเป็นไทมด์คัลเลอร์เพทริเน็ต

3.3 การกำหนดข้อมูลและโครงสร้างไฟล์เอ็กซ์เอ็มแอลสำหรับจัดเก็บเครือข่ายทางรถไฟ

จากมอดูลที่ได้สร้างมาจากหัวข้อ 3.2 ทำให้สามารถแทนการใช้งานมอดูลเหล่านั้นด้วยสัญลักษณ์สำหรับสร้างเครือข่ายทางรถไฟตามตารางที่ 3.2 ดังนี้

ตารางที่ 3.2 ตารางสัญลักษณ์มอดูล

	สัญลักษณ์
สถานี	
ราง รถไฟ	

หลังการออกแบบเครือข่ายทางรถไฟด้วยเครื่องมือที่จะกล่าวถึงในหัวข้อ 3.5 ผู้ใช้งานสามารถบันทึกเครือข่ายทางรถไฟเป็นเอกสารนำออกในรูปแบบเอ็กซ์เอ็มแอลที่มีโครงสร้างที่กำหนดไว้ได้ โดยจะแบ่งเอกสารนำออกเป็น 3 ส่วน สำหรับจัดเก็บข้อมูลต่างชนิดกันดังนี้

3.3.1 ส่วนบันทึกแบบจำลองเครือข่ายทางรถไฟที่ถูกออกแบบ

ส่วนนี้สำหรับบันทึกแบบจำลองเครือข่ายทางรถไฟที่ผู้ใช้งานได้ออกแบบไว้แล้ว โดยจะทำการแปลงแบบจำลองในรูปแบบสัญลักษณ์ที่เชื่อมต่อกันไปเป็นเอกสารนำออกที่อยู่ในรูปแบบเอ็กซ์เอ็มแอล โดยมีอิลิเม้นต์ที่สำคัญดังตารางที่ 3.3

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

ตารางที่ 3.3 ตารางอิลิเมนต์สำหรับหนึ่งมอดูล

อิลิเมนต์	คำอธิบาย
<RailUnit>...</RailUnit>	อิลิเมนต์หลักที่เป็นตัวแทนของมอดูลที่มีพารามิเตอร์ มี name (ชื่อสถานี) , type (station หรือ rail) , x (ตำแหน่งในแนวแกน X) , และ y (ตำแหน่งในแนวแกน Y)
<UnitParam>...</UnitParam>	อิลิเมนต์ลูกของ RailUnit สำหรับเก็บรายการพารามิเตอร์ โดยพารามิเตอร์ต่างๆจะถูกเก็บอยู่ในอิลิเมนต์ลูก หรือ Param
<Param>...</Param>	ค่าพารามิเตอร์ของมอดูลนั้น เช่น จำนวนชานชาลาหรือ Train Buffer (สำหรับมอดูลสถานี) หรือ ความยาวรางรถไฟหรือ Rail Length (สำหรับมอดูลรางรถไฟ)
<Connector>...</Connector>	อิลิเมนต์ลูกของ RailUnit สำหรับเก็บรายการมอดูลอื่นที่เชื่อมต่อกัน โดยชื่อของมอดูลอื่นจะถูกเก็บอยู่ในอิลิเมนต์ลูกหรือ ConnectedUnit
<ConnectedUnit>... </ConnectedUnit>	รายการมอดูลอื่นที่เชื่อมต่ออยู่กับ RailUnit โดยมีพารามิเตอร์ name สำหรับระบุชื่อมอดูลนั้นๆ และ Dir ระบุทิศทางการเชื่อมต่อ (IN หรือ OUT)
<TrainIns>..</TrainIns>	อิลิเมนต์หลักที่มีรายการเลขขบวนรถที่จะเข้าสถานี ในกรณี RailUnit นี้เป็นสถานีชุมทาง (Interchange Station)
<TrainIn>..</TrainIn>	รายการเลขขบวนรถที่จะเข้าสถานี ในกรณี RailUnit นี้เป็นสถานีชุมทาง โดยจะมี trNo สำหรับเก็บเลขขบวนรถไฟและ time สำหรับเก็บเวลาที่รถไฟขบวนนั้นเข้าสถานีในเส้นทางรอง โดยแยกออกมาจากตารางเวลารถไฟปกติ
<TrainOuts>..</TrainOuts>	อิลิเมนต์หลักที่มีรายการเลขขบวนรถที่จะออกจากสถานีนี้ในเส้นทางรอง ในกรณี RailUnit นี้เป็นสถานีชุมทาง
<TrainOut>..</TrainOut>	รายการเลขขบวนรถที่จะออกจากสถานีในเส้นทางรองหรือเรียกว่าสถานีปลายทาง ในกรณี RailUnit นี้เป็นสถานีชุมทาง โดยจะมี trNo สำหรับเก็บเลขขบวนรถไฟ และ time ซึ่งไม่มีการใช้งานโดยจะมีค่าเป็น 0 เสมอ

```
<RailUnit name="STATION" type="Station" x="0" y="0">
  <UnitParam>
    <Param name="Train Buffer" value="2" />
  </UnitParam>
  <Connector>
    <ConnectedUnit name="rail1" dir="OUT" />
  </Connector>
  <TrainIns>
    <TrainIn trNo="0" time="0" />
  </TrainIns>
  <TrainOuts>
    <TrainOut trNo="0" time="0" />
  </TrainOuts>
</RailUnit>
```

รูปที่ 3.18 มอดูลสถานีในรูปแบบเอ็กซ์เอ็มแอล

```
<RailUnit name="rail1" type="Rail" x="0" y="0">
  <UnitParam>
    <Param name="Lenght" value="300" />
  </UnitParam>
  <Connector>
    <ConnectedUnit name="STATION" dir="IN" />
    <ConnectedUnit name="STATION2" dir="OUT" />
  </Connector>
  <TrainIns>
    <TrainIn trNo="0" time="0" />
  </TrainIns>
  <TrainOuts>
    <TrainOut trNo="0" time="0" />
  </TrainOuts>
</RailUnit>
```

รูปที่ 3.19 มอดูลรางรถไฟในรูปแบบเอ็กซ์เอ็มแอล

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

จากตัวอย่างสามารถสร้างกฎการแปลงแบบจำลองจากสัญลักษณ์ไปเป็นมอดูลในรูปแบบเอ็กซ์เอ็มแอลได้ โดยกำหนดให้ ชื่อของ RailUnit ใด ที่ปรากฏที่พารามิเตอร์ name ของ ConnectedUnit หนึ่ง RailUnit นั้นจะต้องมี พารามิเตอร์ name ของ ConnectedUnit เป็นชื่อ RailUnit นั้นในทิศทางตรงกันข้าม (IN/OUT) เสมอ จากรูปที่ 3.18 และ 3.19 RailUnit สถานีชื่อ STATION มี ConnectedUnit คือ RailUnit รางรถไฟชื่อ rail1 ในทำนองเดียวกัน รางรถไฟ rail1 จะมี ConnectedUnit คือ RailUnit สถานีชื่อ STATION ด้วย ส่วนค่าพารามิเตอร์ที่สำคัญที่ระบุไว้จะต้องถูกกำหนดโดยผู้สร้างแบบจำลองก่อนการแปลง เพื่อส่งต่อค่าพารามิเตอร์เหล่านั้นไปยังแบบจำลองไทมด์คัลเลอร์เพทริเน็ตหลังแปลงสำเร็จ

3.3.2 ส่วนบันทึกตารางเวลารถไฟเข้าออกตามแต่ละสถานี

เอกสารนำออกนี้จะบันทึกข้อมูลตารางรถไฟแต่ละขบวนโดยจะบันทึกเวลาออกของรถไฟที่รวมเวลาจอด 1 นาที ในแต่ละสถานีไว้แล้ว โดยมีอิลิเมนต์ที่สำคัญดังตารางที่ 3.4

ตารางที่ 3.4 ตารางอิลิเมนต์สำหรับตารางรถไฟแต่ละขบวน

อิลิเมนต์	คำอธิบาย
<TrainList>...</TrainList>	อิลิเมนต์หลักทำหน้าที่เก็บข้อมูลตารางรถไฟ โดยมีอิลิเมนต์ลูกเป็นตารางรถไฟแต่ละขบวน
<Train>...</Train>	อิลิเมนต์ลูกของ TrainList สำหรับเก็บรายการตารางเวลารถไฟแต่ละสถานีของขบวนนั้นๆ โดยมีพารามิเตอร์เป็น trNo ซึ่งเก็บค่าเลขขบวนรถไฟ
<TrainTime>...</TrainTime>	อิลิเมนต์ย่อยที่เก็บรายละเอียดของเวลาออกจากสถานีใดๆ โดย 1 อิลิเมนต์ จะเก็บค่าเวลาออกของ 1 สถานี ประกอบด้วย 2 พารามิเตอร์ คือ stationName สำหรับเก็บชื่อสถานี และ outTime สำหรับเก็บเวลาออกจากสถานี

```
<TrainList>
  <Train trNo="99">
    <TrainTime stationName="NC" outTime="3" />
    <TrainTime stationName="s5" outTime="5" />
  </Train>
  <Train trNo="112">
    <TrainTime stationName="NC" outTime="1" />
    <TrainTime stationName="s5" outTime="3" />
  </Train>
  <Train trNo="300" />
</TrainList>
```

รูปที่ 3.20 แสดงตัวอย่างเอ็กซ์เอ็มสำหรับเก็บค่าตารางเวลารถไฟ

จากรูปที่ 3.20 แสดงตัวอย่างเอ็กซ์เอ็มแอลที่ใช้เก็บค่าตารางเวลารถไฟสำหรับขบวนต่างๆ โดยสามารถอธิบายรายละเอียดได้ดังนี้ รถไฟเลขขบวนที่ 99 ควรเข้าสู่สถานี NC ในเวลานาทีที่ 2 ของการจำลอง เพื่อทำการจอดรับส่งผู้โดยสารเป็นเวลา 1 นาที หรือ 1 หน่วยยูนิตเวลาของการจำลอง โดยพร้อมออกจากสถานี NC ตั้งแต่เวลานาทีที่ 3 ของการจำลอง และออกเดินทางต่อไปยัง

สถานี s5 ถึงสถานี s5 ในเวลานาทีที่ 4 จากนั้นหยุดจอดรับส่งผู้โดยสาร และพร้อมออกจากสถานี s5 ตั้งแต่เวลานาทีที่ 5 ของการจำลองเป็นต้นไป เพื่อไปยังสถานีปลายทางโดยไม่จอดสถานีใดอีก

โดยจะเห็นได้ว่ารถไฟขบวนเลขที่ 300 ไม่มีอิลิเมนต์ลูกภายใต้ อิลิเมนต์หลัก Train เลย แสดงให้เห็นว่าผู้ใช้ยังไม่ได้กำหนดการจอดรถไฟสำหรับขบวน 300 สำหรับสถานีใดๆ เป็นต้น

3.3.3 ส่วนบันทึกค่าพารามิเตอร์ของขบวนรถไฟ

โดยในเบื้องต้นได้กำหนดค่าพารามิเตอร์สำหรับรถไฟเพียงความเร็วเท่านั้น โดยมีการจำแนกประเภทรถไฟเป็น 3 ประเภทด้วยความเร็วที่ต่างกัน

1) รถด่วนพิเศษหรือรถด่วน คือ รถไฟที่มีเลขขบวนตั้งแต่ 1-99 จะใช้พารามิเตอร์ Express Train Speed ในการกำหนดความเร็วเฉลี่ยในการเคลื่อนจากสถานีหนึ่งสู่สถานีหนึ่ง
2) รถเร็ว คือ รถไฟที่มีเลขขบวนตั้งแต่ 100-199 จะใช้พารามิเตอร์ Rapid Train Speed ในการกำหนดความเร็วเฉลี่ยในการเคลื่อนจากสถานีหนึ่งสู่สถานีหนึ่ง
3) รถธรรมดาหรือรถชานเมือง คือ รถไฟที่มีเลขขบวนตั้งแต่ 200-499 จะใช้พารามิเตอร์ Normal Train Speed ในการกำหนดความเร็วเฉลี่ยในการเคลื่อนจากสถานีหนึ่งสู่สถานีหนึ่ง

โดยเอกสารเอ็กซ์เอ็มแอลจะมีอิลิเมนต์สำหรับจัดเก็บข้อมูลดังกล่าวตามตารางที่ 3.5

ตารางที่ 3.5 ตารางอิลิเมนต์สำหรับจัดเก็บค่าพารามิเตอร์ของรถไฟ

อิลิเมนต์	คำอธิบาย
<TrainParam>...</TrainParam>	อิลิเมนต์หลักทำหน้าที่เก็บประเภทพารามิเตอร์ต่างๆ ของรถไฟ
<trainParam>...</trainParam>	อิลิเมนต์ลูกของ TrainParam สำหรับเก็บรายการพารามิเตอร์ของรถไฟ โดยมี name สำหรับเก็บชื่อพารามิเตอร์ และ value สำหรับเก็บค่าพารามิเตอร์ โดย 1 trainParam มีจัดเก็บค่า 1 พารามิเตอร์

```
<TrainParam>
  <trainParam name="Express Train Speed (km/hr)" value="100" />
  <trainParam name="Rapid Train Speed (km/hr)" value="90" />
  <trainParam name="Normal TrainSpeed (km/hr)" value="80" />
</TrainParam>
```

รูปที่ 3.21 แสดงตัวอย่างค่าพารามิเตอร์ความเร็วของรถไฟประเภทต่างๆ

จากรูปที่ 3.21 แสดงถึงความเร็วของรถไฟประเภทต่างๆ โดยรถด่วนพิเศษจะมีความเร็วตามพารามิเตอร์ Express Train Speed (km/hr) คือ 100 รถเร็วจะมีความเร็วตามพารามิเตอร์ Rapid

Train Speed (km/hr) คือ 90 และรถธรรมดาจะมีความเร็วตามตัวแปล Normal Train Speed (km/hr) คือ 80

3.4 กฎการแปลงเครือข่ายทางรถไฟไปเป็น CPN

เมื่อได้เอกสารนำออกที่อยู่ในรูปแบบเอ็กซ์เอ็มแอลที่มาจากการออกแบบเครือข่ายทางรถไฟที่ประกอบด้วยมอดูลต่างๆจากหัวข้อ 3.3 แล้ว ต่อมาให้ทำการแปลงมอดูลในรูปแบบเอ็กซ์เอ็มแอลไปเป็นแบบจำลองไทมด์คัลเลอร์เพทริเน็ต ซึ่งจะเป็นข้อมูลนำออกที่มีรูปแบบสอดคล้องกับรูปแบบมาตรฐานของภาษาโปรแกรมซีพีเอ็นเอ็มแอลและรูปแบบการจัดเก็บข้อมูลนำออกของโปรแกรมเครื่องมือซีพีเอ็น ที่มีนามสกุลเป็น .cpn ทำให้สามารถเปิดใช้งานได้ด้วยโปรแกรมเครื่องมือซีพีเอ็น โดยจากการวิเคราะห์รูปแบบข้อมูลนำออกที่มีนามสกุลเป็น .cpn มีรายการอิลิเมนต์ที่สำคัญของส่วนประกอบของแบบจำลองไทมด์คัลเลอร์เพทริเน็ต ดังต่อไปนี้

3.4.1 เพลส

ข้อมูลของเพลสจะถูกจัดเก็บไว้ในอิลิเมนต์ <place>...</place> ดังรูปที่ 3.22 และมีรายละเอียดดังตารางที่ 3.6

```
<place id="ID0250794453">
  <posattr x="-1388" y="210" />
  <text>STARTST</text>
  <ellipse w="60" h="40" />
  <type id="ID0609599480">
    <posattr x="-1388" y="180" />
    <text tool="CPN Tools" version="4.0.1">STATION</text>
  </type>
  <initmark id="ID0445979193">
    <posattr x="30" y="-15" />
    <text tool="CPN Tools" version="4.0.1">1`112++
1`400++
1`300</text>
  </initmark>
</place>
```

รูปที่ 3.22 แสดงอิลิเมนต์ที่จัดเก็บข้อมูลที่สำคัญของเพลส

3.4.2 ทรานซิชัน

ข้อมูลของทรานซิชันมีการจัดเก็บต่างจากเพลสดังรูปที่ 3.23 มีรายละเอียดตามตารางที่ 3.7

```
<trans id="ID0804784220" explicit="false">
  <posattr x="-1388" y="100" />
  <text>STARTWaiting</text>
  <box w="60" h="40" />
  <cond id="ID0163589247">
    <posattr x="-1388" y="140" />
    <text tool="CPN Tools" version="4.0.1">trainInSch(train,inTrList) = false</text>
  </cond>
  <time id="ID0999968959">
    <posattr x="-1343" y="120" />
    <text tool="CPN Tools" version="4.0.1">@+1</text>
  </time>
</trans>
```

รูปที่ 3.23 แสดงอิลิเมนต์ที่จัดเก็บข้อมูลที่สำคัญของทรานซิชัน

ตารางที่ 3.6 ตารางอธิบายอิลิเมนต์ต่างๆที่สำคัญของเพลส

อิลิเมนต์	คำอธิบาย
<place>...</place>	อิลิเมนต์หลักของเพลสที่ประกอบด้วย id ที่ไม่ซ้ำกัน และอิลิเมนต์ลูกต่างๆสำหรับจัดเก็บค่าพารามิเตอร์ที่สำคัญ
<posattr>...</posattr>	อิลิเมนต์ลูกของเพลสสำหรับเก็บตำแหน่งที่จะแสดงบนหน้าจอเมื่อแสดงผลในโปรแกรมเครื่องมือซีพีเอ็น โดยจะเก็บในพารามิเตอร์ x และ y
<text>...</text>	อิลิเมนต์ลูกของเพลสสำหรับจัดเก็บชื่อของเพลสนั้นๆ ต้องมีชื่อที่ไม่ซ้ำกันกับเพลสอื่นๆ
<ellipse>..</ellipse>	อิลิเมนต์ลูกของเพลส ใช้ระบุลักษณะของเพลสที่เป็นวงรี โดยมีขนาดกว้างและสูงตามพารามิเตอร์ w และ h ที่กำหนด
<type>...</type>	อิลิเมนต์ลูกของเพลสใช้ระบุชนิดของเพลส โดยจะมีพารามิเตอร์ id ที่ไม่ซ้ำกัน และมีอิลิเมนต์ลูก คือ posattr สำหรับระบุตำแหน่งที่จะแสดงผลและ text สำหรับจัดเก็บชื่อประเภทพารามิเตอร์ โดยประเภทตัวแปรต้องตรงกับชนิดของพารามิเตอร์หรือคัลเลอร์เซ็ตที่ถูกประกาศไว้ก่อนหน้าแล้วเท่านั้น
<initmark>...</initmark>	อิลิเมนต์ลูกของ place ที่ใช้จัดเก็บโทเคนเริ่มต้นหากเพลสนั้นมีการระบุให้มีโทเคนเริ่มต้นไว้ก่อน โดยจะมี posattr จัดเก็บตำแหน่งที่จะแสดงผลโทเคนเริ่มต้นและ text สำหรับจัดเก็บรายการโทเคนเริ่มต้น โดยต้องอยู่ในรูปแบบที่กำหนดไว้เท่านั้น เช่น 1`init ++ 2`initsecond คือ มีการกำหนดโทเคนเริ่มต้นให้มี โทเคนชื่อ init 1 โทเคน และ โทเคนชื่อ initsecond 2 โทเคน ในเพลสนี้

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ตารางที่ 3.7 ตารางอธิบายอิลิเมนต์ต่างๆที่สำคัญของทรานซิชัน

อิลิเมนต์	คำอธิบาย
<trans>...</trans>	อิลิเมนต์หลักของทรานซิชัน ที่ประกอบด้วย id ที่ไม่ซ้ำกัน และ explicit ที่มีค่าเป็น false และอิลิเมนต์ลูกต่างๆสำหรับจัดเก็บค่าพารามิเตอร์ที่สำคัญ
<posattr>...</posattr>	อิลิเมนต์ลูกของทรานซิชันสำหรับเก็บตำแหน่งที่จะแสดงบนหน้าจอเมื่อแสดงผลในโปรแกรมเครื่องมือซีพีเอ็น โดยจะเก็บในพารามิเตอร์ x และ y
<box>..</box>	อิลิเมนต์ลูกของทรานซิชัน ใช้ระบุลักษณะของทรานซิชันที่เป็นกล่องสี่เหลี่ยมผืนผ้า โดยมีขนาดกว้างและสูงตามพารามิเตอร์ w และ h ที่กำหนด
<text>...</text>	อิลิเมนต์ลูกของทรานซิชันสำหรับจัดเก็บชื่อ และต้องไม่ซ้ำกัน
<cond>...</cond>	อิลิเมนต์ลูกของทรานซิชันใช้จัดเก็บเงื่อนไขหรือการ์ดอินสคริปชัน โดยจะมีพารามิเตอร์ id ที่ไม่ซ้ำกัน และมีอิลิเมนต์ลูก คือ posattr สำหรับระบุตำแหน่งที่จะแสดงผลและ text สำหรับจัดเก็บเงื่อนไขต้องตรงหลักของภาษา CPN ML หรือถ้าหากจัดเก็บเป็นชื่อฟังก์ชัน ฟังก์ชันนั้นต้องเคยถูกประกาศไว้ก่อนหน้าแล้ว
<time>...</ time>	อิลิเมนต์ลูกของทรานซิชันใช้จัดเก็บเงื่อนไขเวลาหรือไทม์อินสคริปชัน โดยจะมีพารามิเตอร์ id ที่ไม่ซ้ำกัน และมีอิลิเมนต์ลูก คือ posattr สำหรับระบุตำแหน่งที่จะแสดงผลและ text สำหรับจัดเก็บเงื่อนไขเวลา ต้องตรงหลักของภาษา CPN ML หรือถ้าหากจัดเก็บเป็นชื่อฟังก์ชัน ฟังก์ชันนั้นต้องเคยถูกประกาศไว้ก่อนหน้าแล้ว และต้องนำหน้าด้วย @ เสมอ

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

3.4.3 อาร์คหรือเส้นเชื่อม

ข้อมูลของอาร์คจะมีการจัดเก็บดังรูปที่ 3.24 และมีรายละเอียดตามตารางที่ 3.8

```
<arc id="ID0553958727" orientation="PtoT" order="1">
  <posattr x="0" y="0" />
  <arrowattr headsize="1.2" currentcyckle="2" />
  <transend idref="ID0344603605" />
  <placeend idref="ID0250794453" />
  <annot id="ID0912763753">
    <posattr x="0" y="500" />
    <text tool="CPN Tools" version="4.0.1">train</text>
  </annot>
</arc>
```

รูปที่ 3.24 แสดงอิลิเมนต์ที่จัดเก็บข้อมูลที่สำคัญของอาร์ค

ตารางที่ 3.8 ตารางอธิบายอิลิเมนต์ต่างๆที่สำคัญของอาร์ค

อิลิเมนต์	คำอธิบาย
<arc>...</arc>	อิลิเมนต์หลักของอาร์ค ที่ประกอบด้วย id ที่ไม่ซ้ำกัน ทิศทางหรือ orientation สำหรับระบุทิศทางของเส้นเชื่อมว่าวิ่งเข้าหาเพลส(TtoP) หรือวิ่งออกจากเพลส(PtoT) เท่านั้น order คือลำดับเส้นในที่นี้มีค่าเป็น 1 เสมอ และอิลิเมนต์ลูกต่างๆสำหรับจัดเก็บค่าพารามิเตอร์ที่สำคัญ
<posattr>...</posattr>	อิลิเมนต์ลูกของอาร์คสำหรับเก็บตำแหน่งที่จะแสดงบนหน้าจอเมื่อแสดงผลในโปรแกรมเครื่องมือซีพีเอ็น โดยจะเก็บในพารามิเตอร์ x และ y
<arrowattr>..</arrowattr>	อิลิเมนต์ลูกของอาร์ค ใช้ระบุลักษณะการแสดงผล โดยกำหนดขนาด headsize มีค่าคงที่เป็น 1.2 และ currentcyckle มีค่าคงที่เป็น 2
<transend>...</transend>	อิลิเมนต์ลูกของอาร์ค ใช้สำหรับจัดเก็บทรานซิชันที่อาร์คเชื่อมอยู่ โดยมี idref ระบุตรงกับ id ของทรานซิชันที่เคยจัดเก็บไว้แล้วเท่านั้น
<placeend>...</placeend>	อิลิเมนต์ลูกของอาร์ค ใช้สำหรับจัดเก็บเพลสที่อาร์คเชื่อมอยู่ โดยมี idref ระบุตรงกับ id ของเพลสที่เคยจัดเก็บไว้แล้วเท่านั้น
<annot>...</annot>	อิลิเมนต์ลูกของอาร์คใช้จัดเก็บประเภทหรือตัวแปรที่โทเคนสามารถใช้งานได้ โดยจะมีพารามิเตอร์ id ที่ไม่ซ้ำกัน และมีอิลิเมนต์ลูก คือ posattr สำหรับระบุตำแหน่งที่จะแสดงผลและ text สำหรับจัดเก็บประเภทหรือตัวแปรของโทเคนที่เคยถูกประกาศไว้ก่อนหน้าแล้ว

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

3.4.3 ข้อมูลตารางเวลารถไฟ

ข้อมูลต่างๆจะถูกจัดเก็บในอิลิเมนต์ <ml>...</ml> เสมือนการประกาศตัวแปรหรือชนิดข้อมูลในเอกสารนามสกุล .cpn ทั่วไป โดยจะมีพารามิเตอร์ id ไม่ซ้ำกัน และมี innertext หรือข้อความภายในเป็นรูปแบบการประกาศตัวแปรชนิด list และมีอิลิเมนต์ลูกคือ <layout>..</layout> ใช้สำหรับแสดงผลบนโปรแกรมเครื่องมือซีพีเอ็น ซึ่งโดยปกติแล้วจะกำหนดรูปแบบเหมือนกันกับ innertext

เพื่อการเรียกใช้งานของที่สะดวกขึ้น จึงมีแบ่งข้อมูลตารางเวลารถไฟออกเป็น 4 รายการย่อยในการจัดเก็บในเอกสารนามสกุล .cpn ดังนี้

1) ตารางรายการเลขขบวนรถไฟสำหรับเส้นทางรอง

เป็นตารางรายการเลขขบวนรถไฟสำหรับเส้นทางรอง เมื่อถึงสถานีที่กำหนด รายการจะถูกแยกตามชื่อสถานีที่เป็นสถานีชุมทาง (Interchange Station) โดยมีการกำหนดรูปแบบชื่อตารางที่ขึ้นต้นด้วย trOut ตามด้วยชื่อสถานี ดังรูปที่ 3.25

```
<ml id="ID0000600960">val trOutInterchange1 = [115]:TrOutList;<layout>val trOutInterchange1 = [115]:TrOutList;</layout></ml>
<ml id="ID0359405987">val trOutInterchange2 = [300]:TrOutList;<layout>val trOutInterchange2 = [300]:TrOutList;</layout></ml>
```

รูปที่ 3.25 แสดงตัวอย่างการจัดเก็บข้อมูลตารางรายการรถไฟที่ถูกกำหนดให้ใช้เส้นทางรอง

2) ตารางเวลารถไฟของแต่ละขบวน

เป็นตารางที่ใช้งานสำหรับตรวจสอบเวลาออกของรถไฟขบวนนั้นๆในทุกสถานี โดยจะมีชื่อขึ้นต้นด้วย re และตามด้วยเลขขบวน เช่น re99 จะจัดเก็บตารางเวลารถไฟของรถไฟขบวนเลขที่ 99 โดยมีชื่อสถานีและเวลาที่สามารถออกจากสถานีนั้นได้ ดังรูปที่ 3.26

```
<ml id="ID0401363416">val re99 = [{sName="NC", t=1},
{sName="s5", t=3}]:TSList;<layout>val re99 = [{sName="NC", t=1},
{sName="s5", t=3}]:TSList;</layout></ml>
```

รูปที่ 3.26 แสดงตัวอย่างการจัดเก็บข้อมูลตารางเวลารถไฟขบวนที่ 99 ในเอกสาร .cpn

3) ตารางเวลารถไฟรวมของทุกบวน

เป็นตารางรวมสำหรับการสืบค้นตารางรถไฟแต่ละขบวน โดยจะเก็บตารางเวลารถไฟแต่ละขบวนในตัวแปร ttReList ซึ่งจะถูกประกาศไว้ในเอกสารต้นแบบ และมีตารางเวลารถไฟขบวนต่างๆเป็นข้อมูลที่จะถูกจัดเก็บเพิ่มเติมต่อไป ดังรูปที่ 3.27

```
<ml id="ID0509342950">val ttReList = [{trainNo=99,schedule=re99},
{trainNo=112,schedule=re112},
{trainNo=400,schedule=re400},
{trainNo=211,schedule=re211},
{trainNo=399,schedule=re399},
{trainNo=300,schedule=re300}]:TschList;<layout>val ttReList = [{trainNo=99,schedule=re99},
{trainNo=112,schedule=re112},
{trainNo=400,schedule=re400},
{trainNo=211,schedule=re211},
{trainNo=399,schedule=re399},
{trainNo=300,schedule=re300}]:TschList;</layout></ml>
```

รูปที่ 3.27 แสดงตัวอย่างตารางรถไฟรวมของทุกขบวนที่ถูกจัดเก็บใน ttReList

4) ตารางเวลารถไฟเข้าสถานีต้นทางของทุกขวน

เป็นตารางเวลาสำหรับตรวจสอบเวลาเข้าสถานีต้นทางของรถไฟแต่ละขบวน โดยจะเก็บเวลารถไฟเข้าสถานีต้นทางแต่ละขบวนในตัวแปร inTrList ซึ่งจะถูกประกาศไว้ในเอกสารต้นแบบมีเลขขบวนรถไฟและมีเวลารถไฟเข้าที่สถานีต้นทางของขบวนต่างๆที่ตรงกับตารางเวลาที่ถูกประกาศไว้แล้วเป็นข้อมูลที่จะถูกจัดเก็บเพิ่มเติมต่อไป ดังรูปที่ 3.28

```
<ml id="ID0345722663">val inTrList = [{TrNo=99,TrIn=3,sOut="s5"},
{TrNo=112,TrIn=1,sOut="s5"},
{TrNo=400,TrIn=2,sOut="s5"},
{TrNo=211,TrIn=3,sOut="s5"},
{TrNo=399,TrIn=3,sOut="s5"},
{TrNo=300,TrIn=0,sOut=""}]:TrInSchList;<layout>val inTrList = [{TrNo=99,TrIn=3,sOut="s5"},
{TrNo=112,TrIn=1,sOut="s5"},
{TrNo=400,TrIn=2,sOut="s5"},
{TrNo=211,TrIn=3,sOut="s5"},
{TrNo=399,TrIn=3,sOut="s5"},
{TrNo=300,TrIn=0,sOut=""}]:TrInSchList;</layout></ml>
```

รูปที่ 3.28 แสดงตัวอย่างตารางเวลารถไฟเข้าที่สถานีต้นทางที่ถูกจัดเก็บใน inTrList

3.4.4 ฟังก์ชันที่ถูกเรียกใช้งาน

ฟังก์ชันต่างๆที่ถูกเรียกใช้งานจะถูกประกาศไว้ในเอกสารต้นแบบและจะไม่ถูกแก้ไขเปลี่ยนแปลง โดยมีฟังก์ชันดังต่อไปนี้

1) ฟังก์ชันคำนวณเวลาที่ใช้ในการเดิน หรือ traveltime

ใช้สำหรับคำนวณเวลารถไฟแต่ละประเภทใช้ในการเดินทางผ่านรางที่มีความยาวต่างกัน โดย l คือข้อมูลนำเข้าความยาวราง และ train คือข้อมูลนำเข้าเลขขวบนรถไฟ ดังรูปที่ 3.29

```
<ml id="ID1416094317">fun traveltime (l,train) = if train &lt; 100
then (l div 120)
else if train &lt; 200
then (l div 100)
else (l div 75)
<layout>fun traveltime (l,train) = if train &lt; 100
then (l div 120)
else if train &lt; 200
then (l div 100)
else (l div 75)</layout></ml>
```

รูปที่ 3.29 ฟังก์ชัน traveltime

2) ฟังก์ชันตรวจสอบชนิดรถไฟขาออกหรือ isHeadOutTrain

ใช้สำหรับตรวจสอบว่ารถไฟนี้เป็นขบวนขาออกหรือไม่ ดังรูปที่ 3.30

```
<ml id="ID1415267187">fun isHeadOut train =
(train mod 2 &lt;&gt; 0);
<layout>fun isHeadOut train =
(train mod 2 &lt;&gt; 0);</layout></ml>
```

รูปที่ 3.30 ฟังก์ชัน isHeadOut

3) ฟังก์ชันตรวจสอบชนิดรถไฟขาเข้าหรือ isHeadInTrain

ใช้สำหรับตรวจสอบว่ารถไฟนี้เป็นขบวนขาเข้าหรือไม่ ดังรูปที่ 3.31

```
<ml id="ID1415257808">fun isHeadIn train =
(train mod 2 = 0 );
<layout>fun isHeadIn train =
(train mod 2 = 0 );</layout></ml>
```

รูปที่ 3.31 ฟังก์ชัน isHeadIn

4) ฟังก์ชันตรวจสอบการหยุดรถไฟที่สถานี หรือ isStop

ใช้ตรวจสอบว่ารถไฟเลขขบวนนี้ ถูกกำหนดให้หยุดจอดรับผู้โดยสารหรือไม่ เมื่อถึงสถานีที่ระบุ ดังรูปที่ 3.32

```
<ml id="ID1416417606">fun isStop (trNo, station) =
let val dptTime = getDPT(trNo,station) :INT
in if dptTime &lt;&gt; 999
then  true
else  false
end
<layout>fun isStop (trNo, station) =
let val dptTime = getDPT(trNo,station) :INT
in if dptTime &lt;&gt; 999
then  true
else  false
end</layout></ml>
```

รูปที่ 3.32 ฟังก์ชัน isStop

5) ฟังก์ชันตรวจสอบเวลาออกรถไฟ หรือ isDepartureTime

ใช้ตรวจสอบเวลาออกรถไฟเทียบกับเวลาระหว่างการจำลองระบบ โดยนำเข้าค่าเลขขบวนและชื่อสถานี ถ้าเวลาจำลองระบบขณะนั้นมีค่าเท่ากับหรือมากกว่าเวลาที่ระบุไว้ในตารางเวลา จะคืนค่าผลลัพธ์เป็นจริง ดังรูปที่ 3.33

```
<ml id="ID1416873152">fun isDepartureTime (trNo,station) =
let val dptTime = getDPT(trNo,station);
in
if isStop(trNo,station) = false
orelse dptTime  = 999
orelse time() &gt;= IntInf.fromInt(dptTime)
then true
else false
end
<layout>fun isDepartureTime (trNo,station) =
let val dptTime = getDPT(trNo,station);
in
if isStop(trNo,station) = false
orelse dptTime  = 999
orelse time() &gt;= IntInf.fromInt(dptTime)
then true
else false
end</layout></ml>
```

รูปที่ 3.33 ฟังก์ชัน isDepartureTime

6) ฟังก์ชันตรวจสอบเวลารถไฟเข้าสถานีต้นทาง หรือ trainInSch

เป็นฟังก์ชันสำหรับตรวจสอบเวลาที่จะยิงโทเคนหรือเคลื่อนรถไฟเข้าที่สถานีต้นทางหรือสถานีชุมทาง โดยรับค่าเป็นเลขขบวนรถไฟและรายการเวลารถไฟเข้าสู่สถานีต้นทาง โดยจะเทียบกับเวลาจำลองปัจจุบัน หากเวลาจำลองมีค่าเท่ากับหรือมากกว่าเวลาที่ถูกระบุไว้ในรายการเวลารถไฟเข้าสู่สถานีต้นทาง จะคืนค่าผลลัพธ์เป็นจริง ดังรูปที่ 3.34

```
<ml id="ID1417378016">fun trainInSch(trainNo,[]:TrInSchList) = false
| trainInSch(trainNo, rec_TrInSch::sel_TrInSchList) =
if (trainNo = #TrNo rec_TrInSch) then
 if time() &gt;= IntInf.fromInt(#TrIn rec_TrInSch)
 then true
 else false
else trainInSch(trainNo, sel_TrInSchList)
<layout>fun trainInSch(trainNo,[]:TrInSchList) = false
| trainInSch(trainNo, rec_TrInSch::sel_TrInSchList) =
if (trainNo = #TrNo rec_TrInSch) then
 if time() &gt;= IntInf.fromInt(#TrIn rec_TrInSch)
 then true
 else false
else trainInSch(trainNo, sel_TrInSchList)</layout></ml>
```

รูปที่ 3.34 ฟังก์ชัน trainInSch

7) ฟังก์ชันตรวจสอบรายการรถออกที่สถานีชุมทาง หรือ trainOutSch

ฟังก์ชันสำหรับตรวจสอบรถไฟว่าจะต้องใช้เส้นทางรองหรือไม่เมื่อถึงสถานีชุมทาง โดยจะส่งค่าเลขขบวนรถไฟและรายการเลขขบวนรถไฟสำหรับเส้นทางรองของสถานีนั้น หากเลขขบวนถูกระบุในรายการ ฟังก์ชันจะคืนค่าผลลัพธ์เป็นจริง ดังรูปที่ 3.35

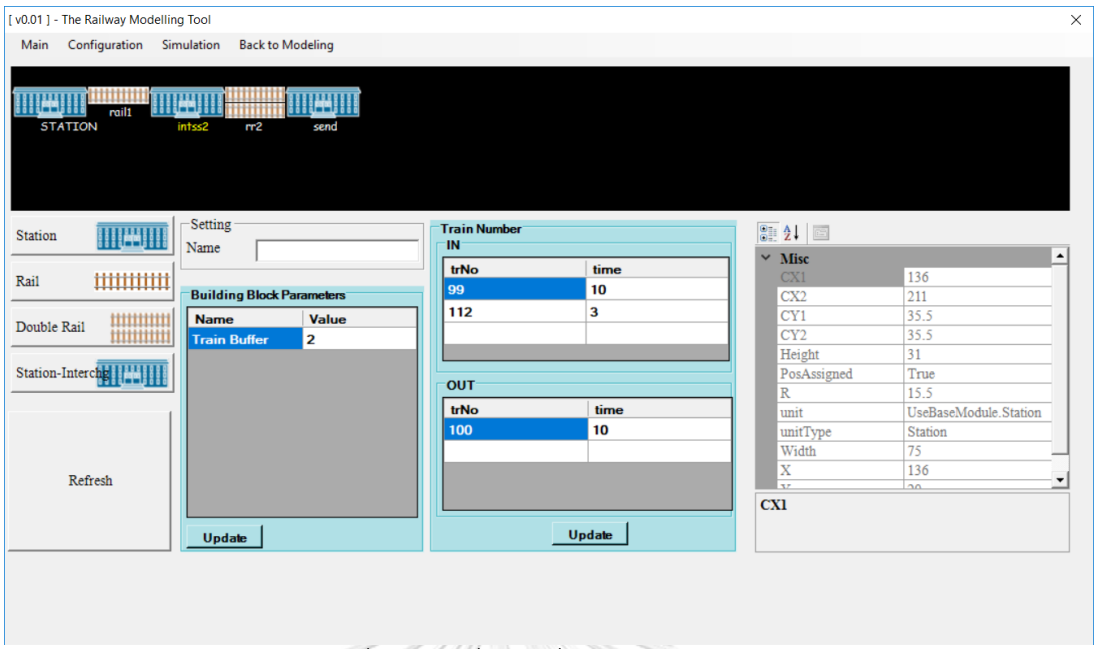
```
| trainOutSch(trainNo, trNum::sel_TrOutList) =
if (trainNo = trNum)
then true
else trainOutSch(trainNo,sel_TrOutList)
<layout>fun trainOutSch(trainNo, []:TrOutList) = false
| trainOutSch(trainNo, trNum::sel_TrOutList) =
if (trainNo = trNum)
then true
else trainOutSch(trainNo,sel_TrOutList)</layout></ml>
```

รูปที่ 3.35 ฟังก์ชัน trainOutSch

รายการทั้งหมดจะถูกจัดเก็บในอิลิเมนต์ <Block>...</Block> ต่างกัน โดยแยกออกจากอิลิเมนต์ Block ของส่วนประกอบคัลเลอร์เพทริเน็ต

3.5 เครื่องมือช่วยเหลือในการแปลงเครือข่ายทางรถไฟไปเป็นไทม์คัลเลอร์เพทริเน็ต

งานวิจัยนี้ได้มีการพัฒนาเครื่องมือ เพื่อช่วยเหลือผู้ใช้งานในการสร้างแบบจำลองคัลเลอร์เพทริเน็ตโดยใช้กฎพื้นฐาน กฎการสร้างเครือข่ายทางรถไฟ และมอดูลที่ถูกกำหนดขึ้นในงานวิจัยนี้ในหัวข้อที่ 3 มาเป็นส่วนประกอบในการพัฒนาเครื่องมือดังแสดงในรูปที่ 3.36 โดยมีรายละเอียดดังต่อไปนี้



รูปที่ 3.36 เครื่องมือที่ถูกพัฒนาในงานวิจัย

3.5.1 คุณสมบัติของเครื่องมือ

เครื่องมือที่ถูกพัฒนานี้ใช้สำหรับเป็นส่วนต่อประสานกับผู้ใช้ โดยให้ผู้ใช้งานสามารถสร้างแบบจำลองเครือข่ายทางรถไฟจากหน้าจอผู้ใช้งาน และสามารถแก้ไขค่าพารามิเตอร์ในส่วนการจัดการข้อมูลรถไฟ เช่น ความยาวรางรถไฟ ความเร็วรถไฟ และตารางเวลารถไฟ เป็นต้น โดยมีรายละเอียดดังต่อไปนี้

1) เครื่องมือได้กำหนดมอดูลไว้ 2 ชนิด สำหรับผู้ใช้งาน คือ
   a. มอดูลสถานี ใช้แทนสถานีรถไฟในแบบจำลอง ซึ่งแบ่งเป็น 2 ประเภท
      i. สถานีทั่วไป หรือ Station ใช้แทนสถานีปกติที่ผู้ใช้งานสามารถกำหนดจำนวนรางรถไฟหรือชานชาลาในแต่ละสถานีได้
      ii. สถานีชุมทาง หรือ Interchange Station ใช้แทนสถานีชุมที่ผู้ใช้สามารถกำหนดรายการรถไฟที่จะใช้งานในเส้นทางรอง โดยจะจัดการนำขบวนรถไฟเข้าหรือออกจากเส้นทางหลักที่สถานีนี้
   b. มอดูรางรถไฟ ใช้แทนรางรถไฟในแบบจำลอง ซึ่งแบ่งเป็น 2 ประเภท
      i. รางเดี่ยว หรือ Rail ใช้แทนรางรถไฟ 1 รางที่อนุญาตให้รถไฟทิศทางขาล่องใดก็ได้ใช้งาน
      ii. รางคู่ หรือ Double Rail ใช้แทนรางรถไฟทางคู่ ที่จะมีรางบนสำหรับรถไฟขาเข้าสถานีกรุงเทพ (วิ่งจากซ้ายไปขวา) และรางล่างสำหรับ

รถไฟขาออกจากสถานีกรุงเทพ (วิ่งจากขวาไปซ้าย) สามารถใช้งานได้เท่านั้น

2) ผู้ใช้งานสามารถนำมอดูลออกแบบไว้ประกอบกันเป็นแบบจำลองเครือข่ายทางรถไฟได้ผ่านหน้าจอได้
3) ผู้ใช้งานสามารถลบมอดูลที่ผู้ใช้งานได้ออกแบบไว้ได้จากมอดูลที่อยู่ขวาสุดก่อนเท่านั้น
4) แบบจำลองที่ถูกออกแบบต้องมีมอดูลสถานีเป็นมอดูลเริ่มต้นและมอดูลสุดท้ายต้องเป็นมอดูลสถานีด้วยเท่านั้น
5) ผู้ใช้งานสามารถกำหนดรายการขบวนรถไฟ โดยการใส่เลขขบวนรถไฟไปโดยมีข้อกำหนด
    a. เลขคู่แทนรถไฟทิศทางขาล่องเข้าสถานีกรุงเทพ
    b. เลขคี่แทนรถไฟทิศทางขาล่องออกจากสถานีกรุงเทพ
6) ผู้ใช้งานสามารถกำหนดตารางเวลารถไฟได้ หลังจากผู้ใช้งานสร้างเครือข่ายทางรถไฟและกำหนดเลขขบวนรถไฟแล้ว
7) ผู้ใช้งานสามารถปรับเปลี่ยนความรถไฟแต่ละชนิดได้
8) ผู้ใช้งานสามารถบันทึกแบบจำลองเครือข่ายทางรถไฟในรูปแบบภาษาเอ็กซ์เอ็มแอลที่มีรูปแบบเป็นไปตามเครื่องมือกำหนด
9) ผู้ใช้งานสามารถแปลงแบบจำลองเครือข่ายทางรถไฟไปเป็นไทมด์คัลเลอร์เพทริเน็ต โดยเครื่องมือจะสร้างเอกสารนำออกที่มีนามสกุล .cpn โดยผู้ใช้งานสามารถนำไปจำลองด้วยโปรแกรมเครื่องมือซีพีเอ็น ได้

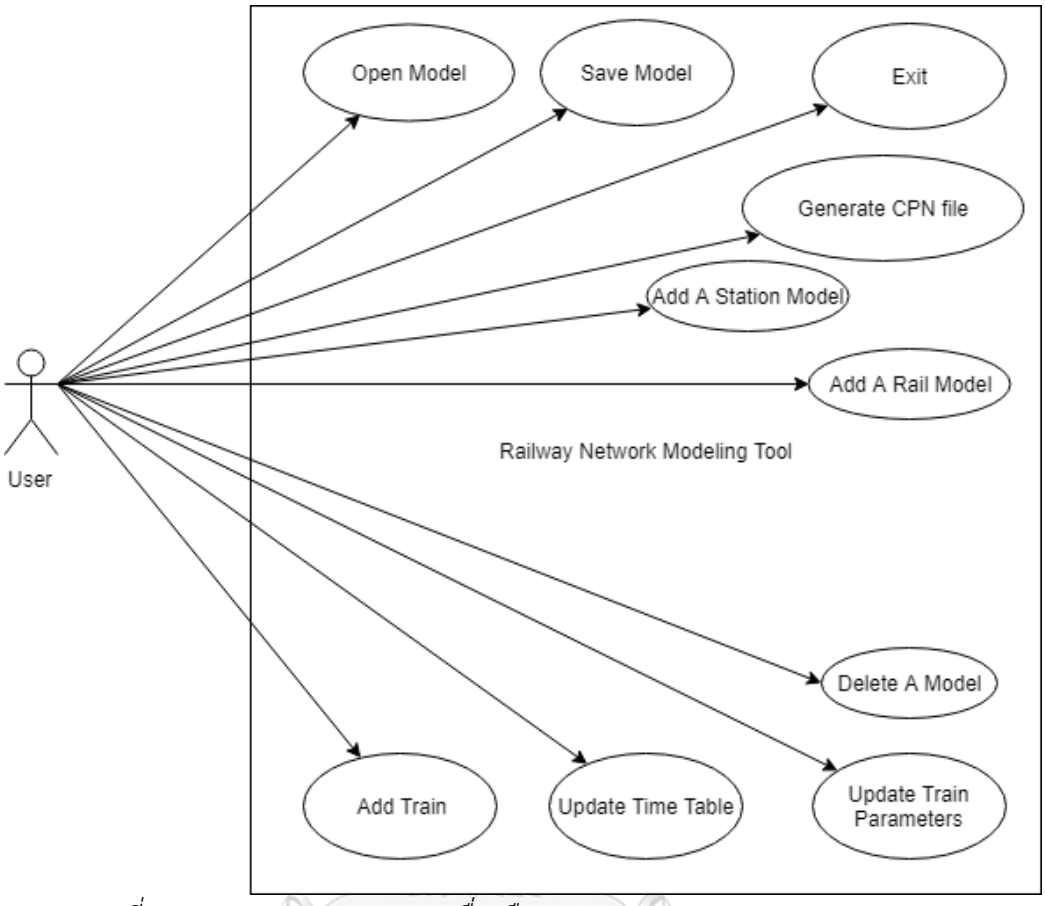
จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

3.5.2 ข้อจำกัดเครื่องมือ

ข้อจำกัดการใช้งานเครื่องมือมีดังต่อไปนี้

1) เครื่องมือรันบนระบบปฏิบัติการ Windows เท่านั้น
2) แบบจำลองที่สร้างเป็นการต่อสถานีและรางในลักษณะเส้นตรงเท่านั้น หรือเรียกว่าเส้นทางหลัก โดยในกรณีสถานีชุมทางที่มีเส้นทางรอง ผู้ใช้สามารถกำหนดรถไฟขบวนที่จะเข้าหรือออกจากสถานีที่สถานีนั้นได้ แต่จะไม่สามารถสร้างแบบจำลองของเส้นทางรองได้
3) ผู้ใช้งานไม่สามารถเพิ่มหรือแก้ไขมอดูลได้
4) ตารางเวลารถไฟแสดงในหน่วยจำนวนเต็ม ผู้ใช้งานต้องทำการแปลงตารางเวลารถไฟที่จะนำเข้าเป็นรูปแบบจำนวนเต็ม โดยมีหน่วยเป็นนาที และผู้ใช้ต้องกำหนดเวลาเริ่มต้นของแบบจำลองหรือนาทีที่ 0 ของแบบจำลองก่อน

3.5.3 แผนภาพยูสเคส (Use Case Diagram)



รูปที่ 3.37 แผนภาพยูสเคสของเครื่องมือ Railway Network Modeling Tool

เครื่องมือมีความสามารถในการใช้งานดังแสดงในแผนภาพยูสเคสดังรูปที่ 3.37 โดยมีตัวอย่างตารางยูสเคสของเครื่องมือดังตารางที่ 3.9 สำหรับยูสเคสการเพิ่มมอดูลสถานีในแบบจำลอง (Add A Station Model) และตารางที่ 3.10 สำหรับยูสเคสการสร้างแฟ้มเอกสารนำออก CPN (Generate CPN File)

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

ตารางที่ 3.9 ตารางยูสเคสการเพิ่มมอดูลสถานีในแบบจำลอง (Add A Station Model)

Use Case ID:	UC001
Use Case Name:	เพิ่มมอดูลสถานีในแบบจำลอง (Add A Station Model)
Actor:	ผู้ใช้งาน (User)
Description:	เพิ่มมอดูลสถานีเข้าไปในแบบจำลองทางด้านขวาสุดของแบบจำลอง
Precondition:	มอดูลที่อยู่ด้านขวาสุดในแบบจำลองเป็นมอดูลประเภทรางเท่านั้น
Postcondition:	-
Step:	1. ผู้ใช้งานใส่ชื่อสถานีที่ต้องการโดยไม่ซ้ำกับชื่อสถานีอื่น 2. ผู้ใช้งานแก้ไขพารามิเตอร์สำหรับสถานีบนหน้าจอ 3. ผู้ใช้งานกดเพิ่มสถานี (Station) 4. ระบบทำการเรียก ModellingController เพื่อสร้างอ็อบเจกต์สถานี (Station) โดยมีชื่อและพารามิเตอร์ตามที่ผู้ใช้งานกำหนด 5. ระบบเพิ่มอ็อบเจกต์สถานีในตารางพารามิเตอร์(ParamList) 6. ระบบเพิ่มอ็อบเจกต์สถานีที่สร้างไปยังรายการสถานี (StationList) บน RailwayNetworkController 7. ระบบสร้างอ็อบเจกต์มอดูลสถานี (createAndAddBuildingBlock) พร้อมตั้งค่าตำแหน่ง (SetUnitPosition) 8. ระบบจัดเก็บอ็อบเจกต์มอดูลสถานีที่ถูกสร้างจากข้อ 7 ไปยังรายการ BuildingBlockList ใน ModellingController 9. ระบบจำลองภาพแบบจำลองใหม่ผ่านฟังก์ชัน drawModel() ซึ่งจะเรียกใช้งาน drawRecursive() ในการสร้างแบบจำลอง
Exceptions conditions:	

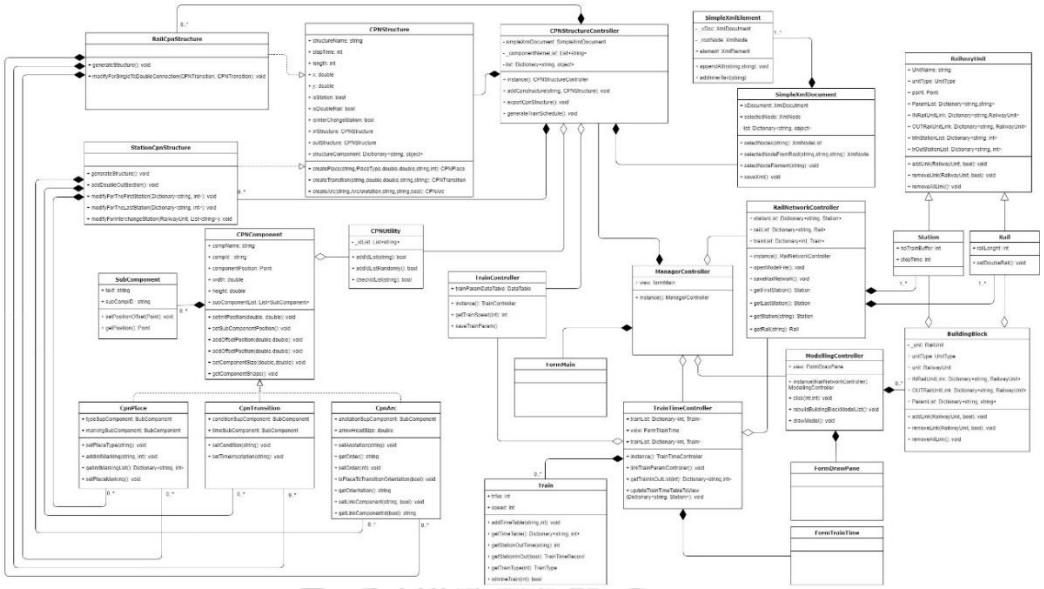
จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ตารางที่ 3.10 ตารางยูสเคสการสร้างแฟ้มเอกสารนำออก CPN (Generate CPN File)

Use Case ID:	UC002
Use Case Name:	สร้างแฟ้มเอกสารนำออก CPN (Generate CPN File)
Actor:	ผู้ใช้งาน (User)
Description:	สร้างแฟ้มเอกสารนำออกที่มีนามสกุล .cpn และมีโครงสร้างตามที่โปรแกรมเครื่องมือซีพีเอ็นกำหนด
Precondition:	แบบจำลองเครือข่ายทางรถไฟได้ถูกสร้างพร้อมทั้งกำหนดค่าพารามิเตอร์ของเครือข่ายทางรถไฟไว้เรียบร้อยแล้ว
Postcondition:	-
Step:	1. ผู้ใช้งานเลือกฟังก์ชันสร้างแฟ้มเอกสารนำออก (Generate CPN File) ผ่านหน้าจอ 2. ระบบทำการเรียก ManagerController เพื่อเตรียมการสร้างแฟ้มเอกสารนำออกผ่านฟังก์ชัน generateToCPNML() 3. ระบบค้นหามอดูลสถานีในแบบจำลองผ่าน RailwayNetworkController 3.1 กรณีเป็นสถานีแรก ระบบจะค้นหาสถานีผ่านฟังก์ชัน getFirstStation() 3.2 กรณีเป็นสถานีถัดไป ระบบจะใช้ชื่อสถานีที่ติดกันค้นหาผ่านฟังก์ชัน getStation 4. ระบบสร้างอ็อบเจกต์ StationCpnStructure เพื่อสร้างโครงสร้างไทมด์คัลเลอร์เพทริเน็ตของมอดูลสถานีนั้น 5. ระบบอ่านตารางเวลารถไฟทั้งหมดที่ผู้ใช้งานกรอก (getTrainInOutList) 6. ระบบแก้ไข StationCpnStructure สำหรับสถานีต้นทาง พร้อมกำหนดตารางรถไฟเริ่มต้นที่จะเข้าที่สถานีต้นทาง 7. ระบบบันทึกอ็อปเจกต์ StationCpnStructure ที่ถูกสร้างลงใน CPNStructureController 8. วนกลับไปทำข้อ 3 จนกระทั่งไม่มีสถานีถัดไป จึงข้ามไปทำข้อ 9 9. ระบบเรียกฟังก์ชัน exportCpnStructure บน CPNStructureController เพื่อสร้างเอกสารนำออกจากรายการ StationCpnStructure ทั้งหมดที่ถูกสร้างก่อนหน้า
Exceptions conditions:	

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

3.5.4 แผนภาพคลาส (Class Diagram)

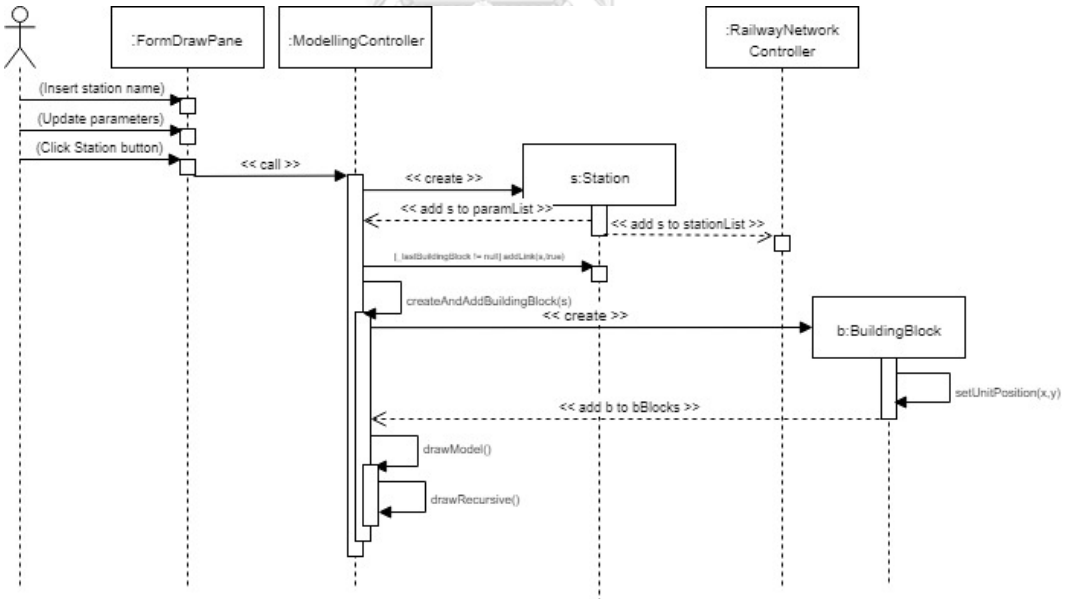


รูปที่ 3.38 แผนภาพคลาสของเครื่องมือ Railway Network Modeling Tool

3.5.5 แผนภาพลำดับการทำงาน (Sequence Diagrams)

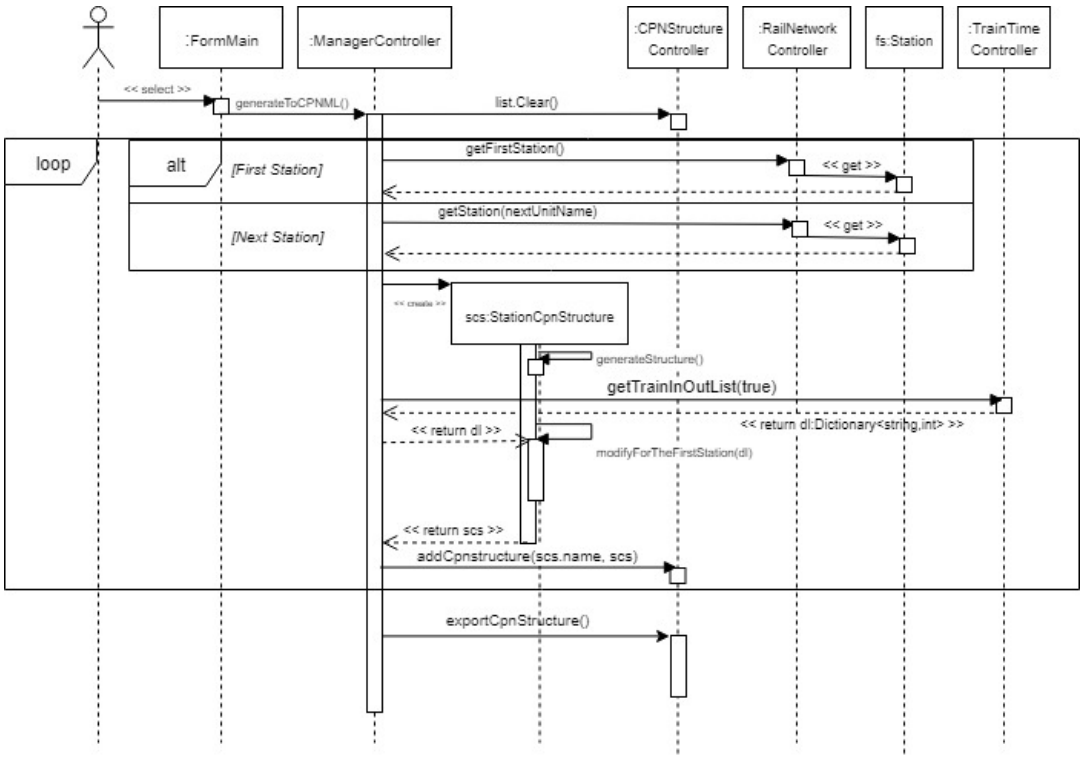
แผนภาพลำดับการทำงานจะยกตัวอย่างโดยประกอบด้วย 2 แผนภาพ

1) แผนภาพลำดับขั้นตอนการเพิ่มมอดูลสถานีในขั้นตอนการออกแบบ



รูปที่ 3.39 แผนภาพลำดับการเพิ่มมอดูลสถานีในขั้นตอนการออกแบบ

2) แผนภาพลำดับการแปลงเครือข่ายทางรถไฟเป็นไทมด์คัลเลอร์เพทริเน็ต



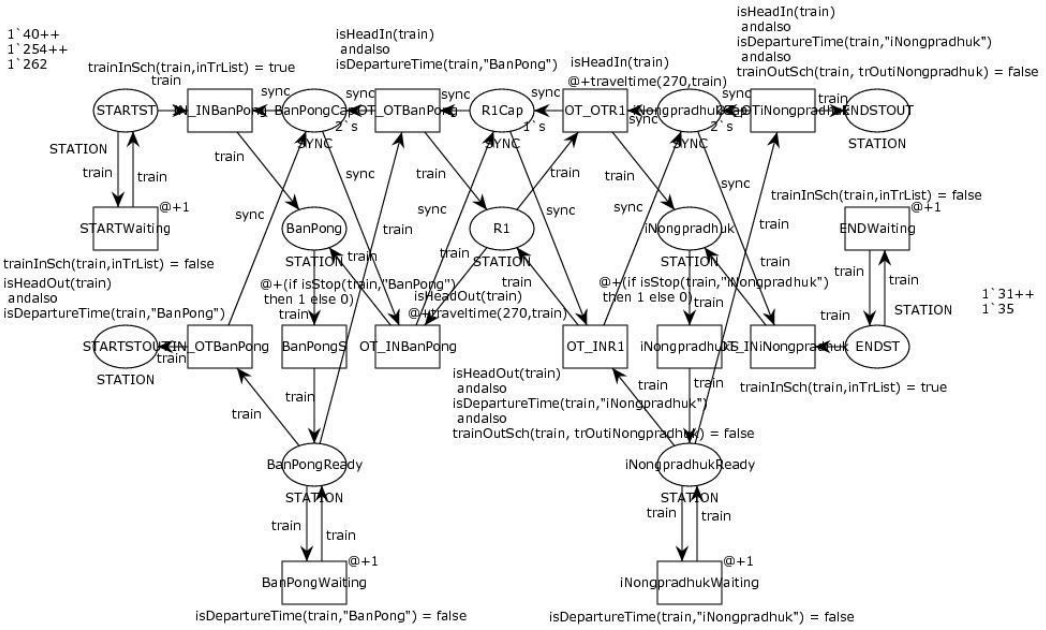
รูปที่ 3.40 แผนภาพลำดับการแปลงเครือข่ายทางรถไฟเป็นไทมด์คัลเลอร์เพทริเน็ต

3.6 อธิบายตัวอย่างแบบจำลองและตัวอย่างเอกสารนำออกที่ถูกบันทึกจากแบบจำลองไทมด์คัลเลอร์เพทริเน็ต

3.6.1 ตัวอย่างแบบจำลองไทมด์คัลเลอร์เพทริเน็ต

เมื่อนำมอดูลมาตรฐานประกอบด้วย 2 มอดูลสถานี ซึ่งเชื่อมต่อด้วยมอดูลรางรถไฟด้วยกฎที่ถูกกำหนดขึ้นในงานวิจัย สามารถแสดงตัวอย่างแบบจำลองที่ได้จากมอดูลข้างต้นดังรูปที่ 3.41

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY



รูปที่ 3.41 แบบจำลองไทมด์คัลเลอร์เพทริเน็ตที่ประกอบด้วย 2 มอดูลสถานีและ 1 มอดูลรางรถไฟ

จากแบบจำลองในรูปที่ 3.41 สถานีเริ่มต้น คือ สถานี BanPong ซึ่งถูกเชื่อมต่อด้วยรางรถไฟ R1 เพื่อไปยังสถานี iNongpraduk ซึ่งจะเป็นสถานีปลายทางสำหรับรถไฟทิศทางขาเข้า โดยจะมีการเพิ่มเพลสสำหรับพักรถไฟเริ่มต้น (STARTST) เนื่องจากสถานี BanPong เป็นสถานีต้นทางสำหรับรถไฟขาเข้าซึ่งจะมีขบวนรถไฟหมายเลข 40, 254 และ 262 อย่างละ 1 ขบวน ถูกกำหนดเป็นโทเคนเริ่มต้นที่เพลส และเพิ่มเพลสสำหรับพักรถไฟเริ่มต้นที่สถานีต้นทางทิศทางขาออกที่อยู่ติดกับสถานี iNongpraduk (ENDST) โดยมีรถไฟทิศทางขาออกหมายเลข 31 และ 35 อย่างละ 1 ขบวน โดยเมื่อทำการจำลองแบบจำลอง โปรแกรมจะทำการเพิ่มเวลา (Global Clock) ขึ้นที่ละ 1 หน่วยเวลา โดยรถไฟแต่ละขบวนได้ถูกกำหนดตารางเวลาไว้ตั้งแต่ขั้นตอนการสร้างแบบจำลอง โปรแกรมจะตรวจสอบเวลาของรถไฟ หากขบวนใดถึงกำหนดเวลาเข้าที่สถานีต้นทาง การ์ดอินสคริปชันที่ถูกกำหนดไว้บนทรานซิชัน INBanPong หรือ INiNongpraduk โดยมีฟังก์ชัน trainInSch ถูกเรียกใช้งานอยู่จะมีค่าความจริงเป็นจริง โทเคนที่เป็นตัวแทนของขบวนรถไฟจะถูกยิงผ่านทรานซิชัน พร้อมลดโทเคน s ที่เป็นประเภท SYNC ที่ถูกเก็บในเพลส Cap (BanPongCap หรือ iNongpradukCap) ลง 1 ตัว เพื่อจำกัดปริมาณรถฟที่เข้าใช้งานสถานี เมื่อเพลส Cap ไม่มีโทเคนเหลืออยู่ แสดงถึงสถานีนั้นไม่มีชานชาลาว่างเหลืออยู่ จะไม่มีรถไฟขบวนใดเข้ามาใช้งานได้อีก โดยทั้งนี้จำนวนโทเคน s เริ่มต้นขึ้นอยู่กับค่าพารามิเตอร์ Tran Buffer ที่ผู้ใช้กำหนด โดยกำหนดค่าเริ่มต้น (Default Value) เป็น 2

จากนั้นโทเคนจะถูกยิงต่อผ่านทรานซิชัน S ที่เชื่อมต่อกันเพลส โดยมีไทมด์อินคริปชันที่เรียกฟังก์ชัน isStop สำหรับตรวจสอบรถไฟที่จอดสถานี หากรถไฟขบวนนั้นถูกระบุให้จอดสถานี เวลาของโทเคนจะเพิ่มขึ้น 1 หน่วยเวลา โทเคนจะถูกยิงมาเก็บที่ Ready (BangPongReady หรือ

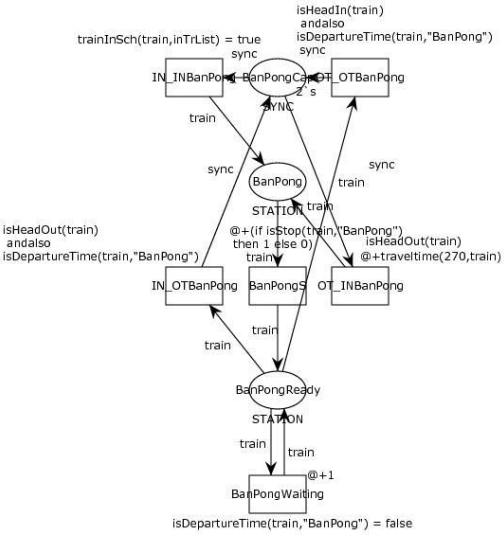
iNongpradukReady) เพื่อเตรียมพร้อมออกจากสถานีเมื่อถึงเวลาที่ระบุในตาราง โดยจะมีฟังก์ชัน isDepartureTime ในการตรวจสอบบนการ์ดอินสคริปชัน หากยังไม่ถึงเวลาที่ต้องออก โทเคนจะถูกยิงด้วยทรานซิชัน Waiting เพื่อเพิ่มเวลา 1หน่วยเวลาให้แก่รถไฟขบวนนั้นเสมือนการจอดรอที่สถานี เมื่อถึงเวลาที่กำหนด โทเคนจะถูกยิงผ่านทรานซิชัน OUT ใด (IN_OT หรือ OT_OT) โดยจะทำการตรวจสอบทิศทางขารถไฟก่อน หากเป็นทิศทางขาเข้า ทรานซิชัน OT_OT จะถูกเอ็นเนเบิล หากเป็นทางขาออก IN_OT จะถูกเอ็นเนเบิลจากการ์ดอินสคริปชันที่ประกอบด้วย isDepartureTime และ isHeadIn หรือ isHeadOut

หากรถไฟทิศทางขาเข้า เช่น ขบวน 40 ถูกยิงจากสถานี BanPong โทเคนจะผ่านทรานซิชัน OT_OTBanPong เข้าสู่ เพลส R1 โดยเพิ่มโทเคน s 1 ตัว กลับไปที่เพลส BanPongCap เพื่อคืนสถานะการใช้งานสถานีเพื่อให้สถานีสามารถรองรับรถไฟขบวนต่อไปได้ พร้อมลดโทเคน s บน R1Cap ลง 1 ตัว แสดงสถานะราง R1 ถูกใช้งาน เนื่องจาก R1Cap จะไม่เหลือโทเคน s อยู่เลย จนกว่าโทเคนนั้นจะถูกยิงผ่าน ทรานซิชัน OT_OTR1 โทเคน s จะถูกคืนกลับไปที่ R1Cap 1 ตัว พร้อมเพิ่มหน่วยเวลาให้แก่โทเคน 40 จำนวนหนึ่ง ทั้งนี้ขึ้นอยู่กับความยาวรางรถไฟที่ถูกระบุและความเร็วรถไฟประเภทต่างๆ หากรางรถไฟยาว 1 กิโลเมตร ความเร็วรถไฟด่วนพิเศษ (ขบวนรถไฟที่ 40 จัดเป็นรถไฟด่วนพิเศษ) ถูกกำหนดให้วิ่งด้วยความเร็วเฉลี่ย 60 กิโลเมตรต่อชั่วโมง ฉะนั้นเวลาจะสำหรับโทเคน 40 จะถูกเพิ่มขึ้น 1 หน่วย ซึ่งจะถูกคำนวณด้วยฟังก์ชัน traveltime บนไทมด์อินสคริปชัน

จากนั้นโทเคนจะถูกยิงเข้าสู่ส่วนของสถานีถัดไป (iNongpraduk) และทำการตรวจสอบเวลาออกดังเช่นกระบวนการที่เกิดขึ้นที่สถานี BanPong แต่เนื่องจากสถานี iNongpraduk ถือเป็นสถานีปลายทางของตัวอย่างแบบจำลองนี้ โทเคนเมื่อถูกยิงออกไปจากสถานีนี้ จะถูกเก็บเข้าเพลส Parking (ENDSTOUT) พร้อมด้วยเวลาสุดท้ายที่โทเคนออกไปจากแบบจำลอง

3.6.2 ตัวอย่างเอกสารนำออกที่ถูกบันทึกจากแบบจำลองไทมด์คัลเลอร์เพทริเน็ต

แสดงตัวอย่างมอดูลไทมด์คัลเลอร์เพทริเน็ตสำหรับสถานีที่ถูกออกแบบดังรูปที่ 3.42 เมื่อถูกบันทึกเป็นเอกสารนำออกนามสกุล .cpn จะมีโครงสร้างดังรูปที่ 3.43 และ 3.44



รูปที่ 3.42 แสดงแบบจำลองไทม์คัลเลอร์เพทริเน็ตสำหรับมอดูลสถานี

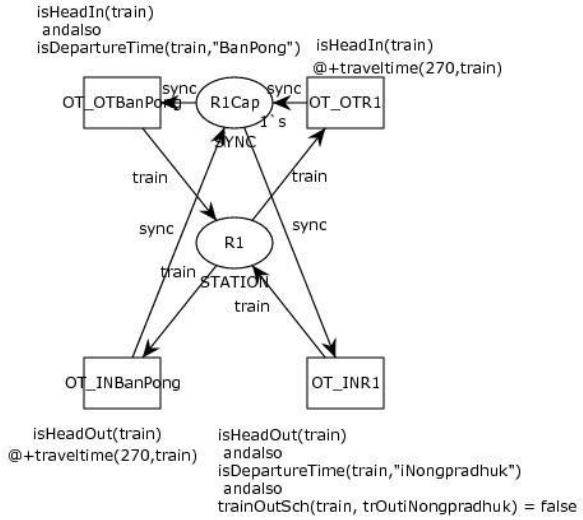
```
<cpnet>
  <page id="ID6">
    <pageattr name="Main"/>
    <place id="ID0506781480">
      <posattr x="-1212.000000" y="210.000000"/>
      <text>BanPongCap</text>
      <ellipse w="60.000000" h="40.000000"/>
      <marking x="0.000000" y="0.000000" hidden="false"> </marking>
      <type id="ID0865586507">
        <posattr x="-1212.000000" y="179.000000"/>
        <text tool="CPN Tools" version="4.0.1">SYNC</text>
      </type>
      <initmark id="ID0701966220">
        <posattr x="-1182.000000" y="195.000000"/>
        <text tool="CPN Tools"
              version="4.0.1">2`s</text>
      </initmark>
    </place>
    <place id="ID0060771248">
      <posattr x="-1212.000000" y="100.000000"/>
      <text>BanPong</text>
      <ellipse w="60.000000" h="40.000000"/>
      <marking x="0.000000" y="0.000000" hidden="false">
      </marking>
      <type id="ID0419576274">
        <posattr x="-1212.000000" y="69.000000"/>
        <text tool="CPN Tools" version="4.0.1">STATION</text>
      </type>
      <initmark id="ID1418002713">
        <posattr x="-1155.000000" y="123.000000"/>
        <text tool="CPN Tools" version="4.0.1"/>
      </initmark>
    </place>
    <place id="ID0451140727">
      <posattr x="-1212.000000" y="-120.000000"/>
      <text>BanPongReady</text>
      <ellipse w="60.000000" h="40.000000"/>
      <marking x="0.000000" y="0.000000" hidden="false">
      </marking>
      <type id="ID0809945754">
        <posattr x="-1212.000000" y="-151.000000"/>
        <text tool="CPN Tools" version="4.0.1">STATION</text>
      </type>
      <initmark id="ID1418002718">
        <posattr x="-1155.000000" y="-96.000000"/>
        <text tool="CPN Tools" version="4.0.1"/>
      </initmark>
    </place>
    <trans id="ID0203617207"
           explicit="false">
      <posattr x="-1300.000000" y="210.000000"/>
      <text>IN_INBanPong</text>
      <box w="60.000000" h="40.000000"/>
      <binding x="7.200000" y="-3.000000"/>
      <cond id="ID0562422234">
        <posattr x="-1326.000000" y="250.000000"/>
        <text tool="CPN Tools"
              version="4.0.1">trainInSch(train,inTrList) = true</text>
      </cond>
      <time id="ID1418002698">
        <posattr x="-1255.500000" y="241.000000"/>
        <text tool="CPN Tools" version="4.0.1"/>
      </time>
    </trans>
    <trans id="ID0398801947" explicit="false">
      <posattr x="-1300.000000" y="-10.000000"/>
      <text>IN_OTBanPong</text>
      <box w="60.000000" h="40.000000"/>
      <binding x="7.200000" y="-3.000000"/>
      <cond id="ID0757606973">
        <posattr x="-1395.000000" y="35.000000"/>
        <text tool="CPN Tools" version="4.0.1">isHeadOut(train)
 andalso
isDepartureTime(train,&amp;quot;BanPong&amp;quot;)</text>
      </cond>
      <time id="ID1418002702">
        <posattr x="-1255.500000" y="21.000000"/>
        <text tool="CPN Tools" version="4.0.1"/>
      </time>
    </trans>
    <trans id="ID0116412001"
           explicit="false">
      <posattr x="-1125.000000" y="210.000000"/>
      <text>OT_OTBanPong</text>
      <box w="60.000000" h="40.000000"/>
      <binding x="7.200000" y="-3.000000"/>
      <cond id="ID0433746656">
        <posattr x="-1085.000000" y="267.000000"/>
        <text tool="CPN Tools" version="4.0.1">isHeadIn(train)
 andalso
isDepartureTime(train,&amp;quot;BanPong&amp;quot;)</text>
      </cond>
      <time id="ID1418002706">
        <posattr x="-1080.500000" y="241.000000"/>
        <text tool="CPN Tools"
              version="4.0.1"/>
      </time>
    </trans>
    <trans id="ID0311596741" explicit="false">
      <posattr x="-1125.000000" y="-10.000000"/>
      <text>OT_INBanPong</text>
      <box w="60.000000" h="40.000000"/>
      <binding x="7.200000" y="-3.000000"/>
      <cond id="ID0270126369">
        <posattr x="-1066.500000" y="37.000000"/>
        <text tool="CPN Tools" version="4.0.1">isHeadOut(train)</text>
      </cond>
      <time id="ID0628931396">
        <posattr x="-1049.500000" y="21.000000"/>
        <text tool="CPN Tools" version="4.0.1">@+traveltime(270,train)</text>
      </time>
    </trans>
    <trans id="ID0259595987" explicit="false">
      <posattr x="-1212.000000" y="-10.000000"/>
      <text>BanPong$</text>
      <box w="60.000000" h="40.000000"/>
      <binding x="7.200000" y="-3.000000"/>
      <time id="ID0614761014">
        <posattr x="-1170.000000" y="41.000000"/>
        <text tool="CPN Tools" version="4.0.1">
@+(if isStop(train,&amp;quot;BanPong&amp;quot;) then 1 else 0)</text>
      </time>
    </trans>
    <trans id="ID0168750781" explicit="false">
      <posattr x="-1212.000000" y="-230.000000"/>
      <text>BanPongWaiting</text>
      <box w="60.000000" h="40.000000"/>
      <binding x="7.200000" y="-3.000000"/>
      <cond id="ID0005130494">
        <posattr x="-1193.000000" y="-261.000000"/>
```

รูปที่ 3.43 แสดงโครงสร้างเอกสารนำออกที่ถูกสร้างมาจากมอดูลสถานี (ส่วนที่ 1)

รูปที่ 3.44 แสดงโครงสร้างเอกสารนำออกที่ถูกสร้างมาจากมอดูลสถานี (ส่วนที่ 2)

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

มอดูลรางรถไฟที่ถูกออกแบบด้วยไทมด์คัลเลอร์เพทริเน็ตดังรูปที่ 3.45 เมื่อถูกบันทึกเป็นเอกสารนำออกนามสกุล .cpn จะมีโครงสร้างเอกสารดังรูปที่ 3.46



รูปที่ 3.45 แบบจำลองไทมด์คัลเลอร์เพทริเน็ตสำหรับมอดูลรางรถไฟมาตรฐาน

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

รูปที่ 3.46 แสดงโครงสร้างเอกสารนำออกที่ถูกสร้างมาจากมอดูลรางรถไฟ

บทที่ 4
ผลการจำลองแบบจำลองเครือข่ายทางรถไฟ

4.1 ข้อมูลที่ใช้ในการสร้างแบบจำลองเครือข่ายทางรถไฟ

จากข้อมูลเครือข่ายทางรถไฟเส้นทางต่างๆ งานวิจัยได้คัดเลือกสถานีจำนวน 8 สถานี ในเส้นทางสายใต้ที่มีโครงสร้างรางรถไฟทางผสมระหว่างรางเดี่ยวและรางคู่ โดยมีรายการข้อมูลสถานีดังนี้

1) สถานีท่าแฉลบ มีจำนวน 3 ชานชาลา
   ระยะทางระหว่างสถานี 4.6 กิโลเมตร ทางรถไฟประเภทรางคู่
2) สถานีต้นสำโรง มีจำนวน 4 ชานชาลา
   ระยะทางระหว่างสถานี 3.5 กิโลเมตร ทางรถไฟประเภทรางคู่
3) สถานีนครปฐม มีจำนวน 3 ชานชาลา
   ระยะทางระหว่างสถานี 2 กิโลเมตร ทางรถไฟประเภทรางเดี่ยว
4) สถานีพระราชวังสนามจันทร์ มีจำนวน 1 ชานชาลา
   ระยะทางระหว่างสถานี 5 กิโลเมตร ทางรถไฟประเภทรางเดี่ยว
5) สถานีโพรงมะเดื่อ มีจำนวน 2 ชานชาลา
   ระยะทางระหว่างสถานี 3.7 กิโลเมตร ทางรถไฟประเภทรางเดี่ยว
6) สถานีคลองบางดาล มีจำนวน 2 ชานชาลา
   ระยะทางระหว่างสถานี 4.2 กิโลเมตร ทางรถไฟประเภทรางเดี่ยว
7) ชุมทางหนองปลาดุก มีจำนวน 5 ชานชาลา
   ระยะทางระหว่างสถานี 4.5 กิโลเมตร ทางรถไฟประเภทรางเดี่ยว
8) สถานีบ้านโป่ง มีจำนวน 4 ชานชาลา เป็นสถานีปลายทางของแบบจำลอง

โดยเส้นทางดังกล่าวได้มีปริมาณการใช้งานเครือข่ายทางรถไฟหนาแน่นในช่วงเวลา 14:00 – 20:00 โดยมีขบวนรถไฟทั้งสิ้น 16 ขบวน โดยแบ่งเป็น

1) รถไฟขาล่องออกจากสถานีกรุงเทพ โดยมีสถานีต้นทางเป็นสถานีกรุงเทพและสถานีธนบุรี 11 ขบวน ประกอบด้วยขบวนเลขที่ 31, 37/45 (ขบวนร่วม), 83, 167, 169, 171, 173, 251, 259, 351, และ 355
2) รถไฟขาล่องเข้าสถานีกรุงเทพ โดยมีปลายทางเป็นสถานีกรุงเทพและสถานีธนบุรี 4 ขบวน ประกอบด้วยขบวนเลยที่ 40, 254, 258 และ 262
3) รถไฟขาล่องเข้าสถานีกรุงเทพ โดยมีสถานีปลายทางเป็นชุมทางหนองปลาดุก 1 ขบวน คือ ขบวนเลขที่ 486

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

โดยใช้ตารางเวลารถไฟ ณ วันที่ 1 พฤศจิกายน 2561 จากเวปไซด์การรถไฟแห่งประเทศไทย โดยนำเวลาที่ระบุในตารางรถไฟแปลงเป็นหน่วยนาที โดยนับเวลาเริ่มที่ 14.00 เป็นนาทีที่ 0 ของแบบจำลอง และคัดเลือกใช้งานตารางเวลารถไฟเฉพาะสถานีที่ระบุเท่านั้น

ตัวอย่างตารางที่ 4.1 แสดงตัวอย่างตารางรถไฟขบวนเลขที่ 259 (ทิศทางขาล่องออก)

ตารางที่ 4.1 ตารางเวลารถไฟขบวนเลขที่ 259

รถธรรมดา เลขที่ 259 : ธนบุรี - น้ำตก			
ลำดับ	สถานีที่หยุด	ถึง	ออก
1	ธนบุรี	-	13:55
2	จรัลสนิทวงศ์	13:58	13:59
3	บางระมาด	14:04	14:05
4	ชุมทางตลิ่งชัน	14:09	14:10
5	บ้านฉิมพลี	14:13	14:14
6	ศาลาธรรมสพน์	14:19	14:20
7	ศาลายา	14:24	14:25
8	วัดสุวรรณ	14:29	14:30
9	คลองมหาสวัสดิ์	14:34	14:35
10	วัดงิ้วราย	14:39	14:40
11	นครชัยศรี	14:44	14:45
12	ท่าแฉลบ	14:49	14:50
13	ต้นสำโรง	14:54	14:55
14	นครปฐม	14:59	15:03
15	โพรงมะเดื่อ	15:10	15:11
16	คลองบางตาล	15:15	15:16
17	ชุมทางหนองปลาดุก	15:21	15:22

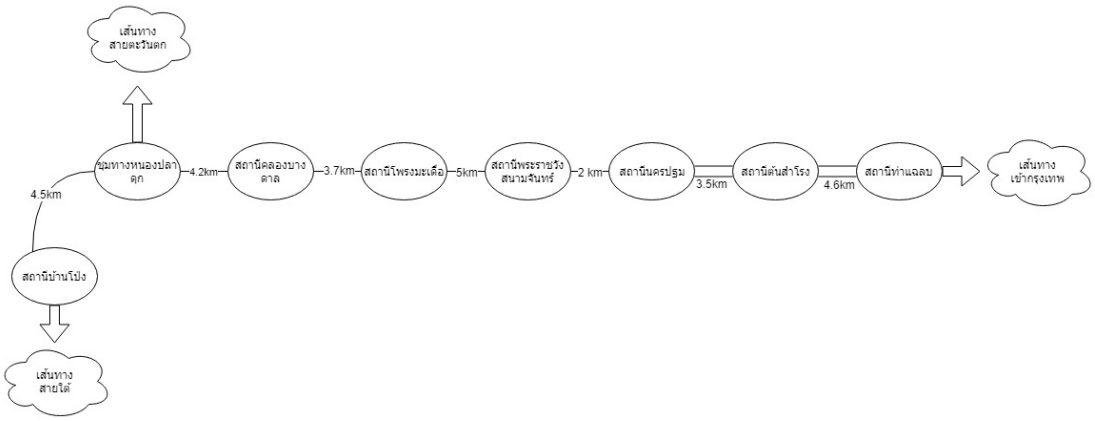
โดยจากตารางเวลารถไฟดังกล่าว สามารถแปลงตารางเวลารถไฟเพื่อใช้เป็นข้อมูลนำเข้าในแบบจำลอง โดยจะใช้ชุดข้อมูลเฉพาะสถานีที่เลือกเท่านั้น จะได้ตารางเวลาใหม่หลังทำการแปลงเป็นหน่วยนาทีของการจำลองแล้วดังตารางที่ 4.2

ตารางที่ 4.2 ตารางเวลารถไฟหลังการแปลงเริ่มเวลานาทีที่ 0

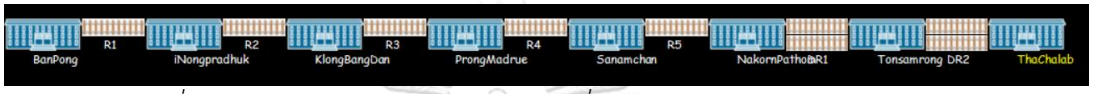
รถธรรมดา เลขที่ 259 : ธนบุรี - น้ำตก			
ลำดับ	สถานีที่หยุด	ถึง	ออก
1	ธนบุรี	-	-
2	จรัลสนิทวงศ์	-	-
3	บางระมาด	-	-
4	ชุมทางตลิ่งชัน	-	-
5	บ้านฉิมพลี	-	-
6	ศาลาธรรมสพน์	-	-
7	ศาลายา	-	-
8	วัดสุวรรณ	-	-
9	คลองมหาสวัสดิ์	-	-
10	วัดงิ้วราย	-	-
11	นครชัยศรี	-	-
12	ท่าแฉลบ	49	50
13	ต้นสำโรง	54	55
14	นครปฐม	59	63
15	โพรงมะเดื่อ	70	71
16	คลองบางตาล	75	76
17	ชุมทางหนองปลาดุก	81	82

4.2 แบบจำลองเครือข่ายทางรถไฟ

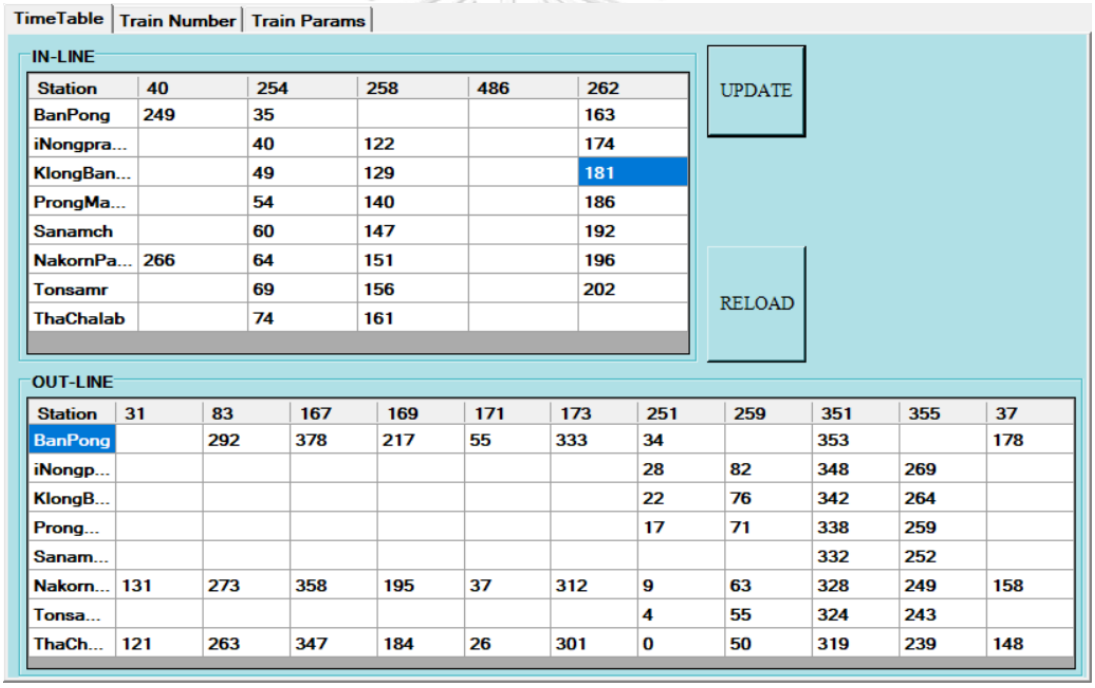
จากข้อมูลประกอบการสร้างแบบจำลองในหัวข้อ 4.1 สามารถแสดงได้ดังรูปที่ 4.1 โดยสามารถสร้างแบบจำลองโดยใช้เครื่องมือที่พัฒนาในงานวิจัย ทำการออกแบบแบบจำลองเป็นไปตามข้อมูลเบื้องต้นดังแสดงในรูปที่ 4.2 และนำเข้าข้อมูลตารางเวลารถไฟขบวนต่างๆที่เกี่ยวข้องกับแบบจำลองโดยกำหนดให้เริ่มต้น นาทีที่ 0 เวลา 14.00 น. ชุดข้อมูลตารางเวลารถไฟทั้งหมดดังแสดงในรูปที่ 4.3



รูปที่ 4.1 แสดงรูปตัวอย่างเครือข่ายทางรถไฟจากข้อมูลในหัวข้อ 4.1



รูปที่ 4.2 แบบจำลองเครือข่ายทางรถไฟที่สร้างมาจากสถานีจำนวน 8 สถานี

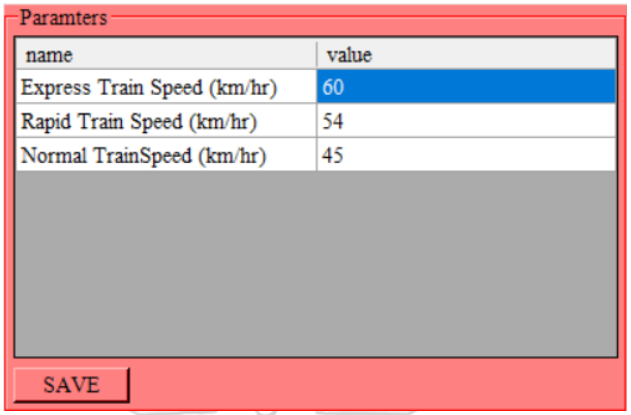


Station	40	254	258	486	262
BanPong	249	35			163
iNongpra...		40	122		174
KlongBan...		49	129		181
ProngMa...		54	140		186
Sanamch		60	147		192
NakornPa...	266	64	151		196
Tonsamr		69	156		202
ThaChalab		74	161		

Station	31	83	167	169	171	173	251	259	351	355	37
BanPong		292	378	217	55	333	34		353		178
iNongp...							28	82	348	269	
KlongB...							22	76	342	264	
Prong...							17	71	338	259	
Sanam...									332	252	
Nakorn...	131	273	358	195	37	312	9	63	328	249	158
Tonsa...							4	55	324	243	
ThaCh...	121	263	347	184	26	301	0	50	319	239	148

รูปที่ 4.3 รูปแสดงตารางเวลารถไฟโดยเริ่มต้นที่เวลา 14.00 เป็นนาทีที่ 0 ในแบบจำลอง

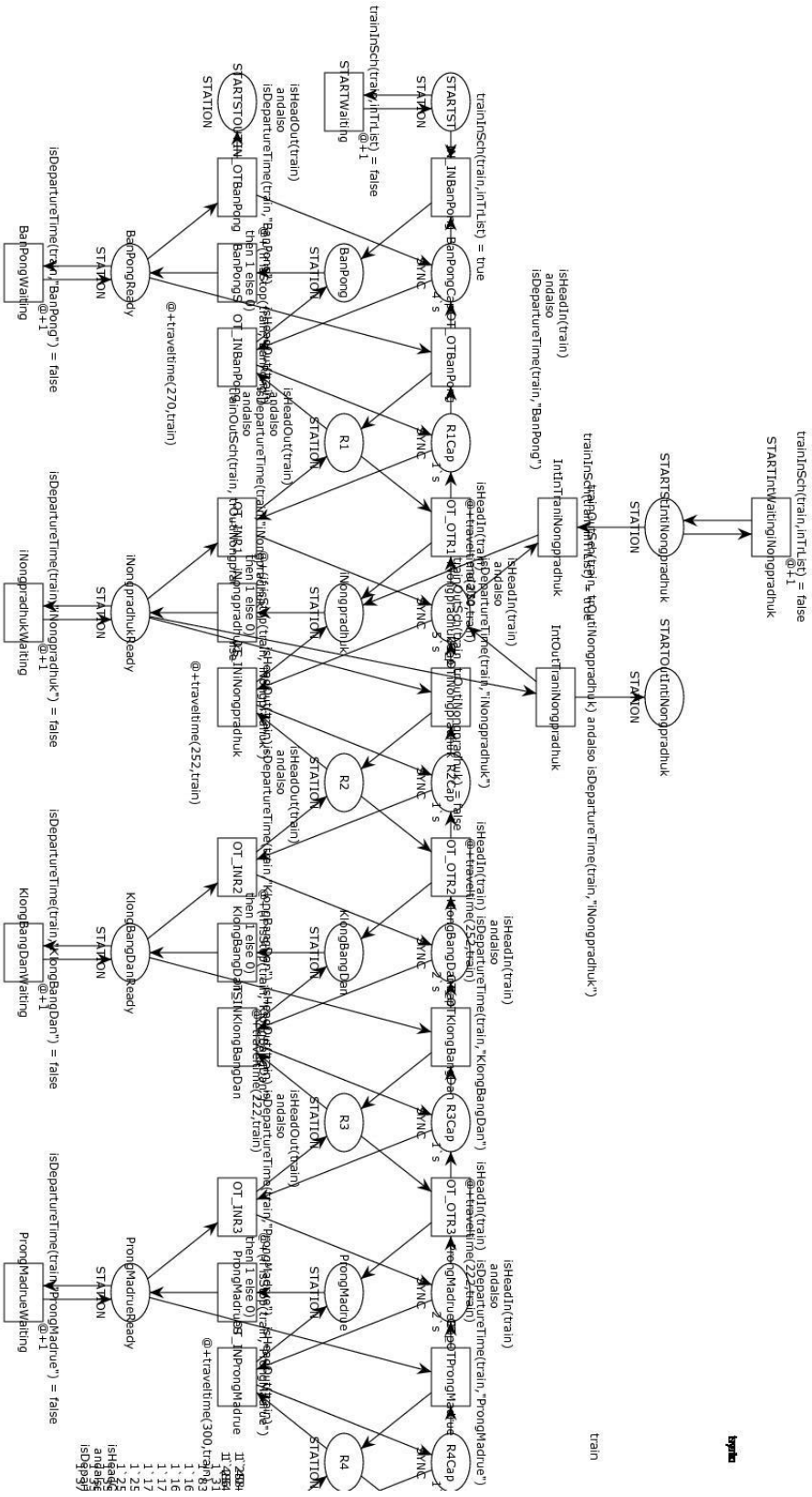
พร้อมกำหนดความเร็วรถไฟเฉลี่ยของแต่ละประเภทเริ่มต้นดังข้อมูลตารางในรูปที่ 4.4 โดยความเร็วเฉลี่ยรถไฟแต่ละประเภทได้ถูกกำหนดภายใต้สมมุติฐานว่า เมื่อนำความเร็วรถไฟแต่ละประเภทไปใช้ในการจำลองสามารถให้ผลลัพธ์ใกล้เคียงกับเวลาที่ระบุไว้บนตารางเวลามากที่สุด



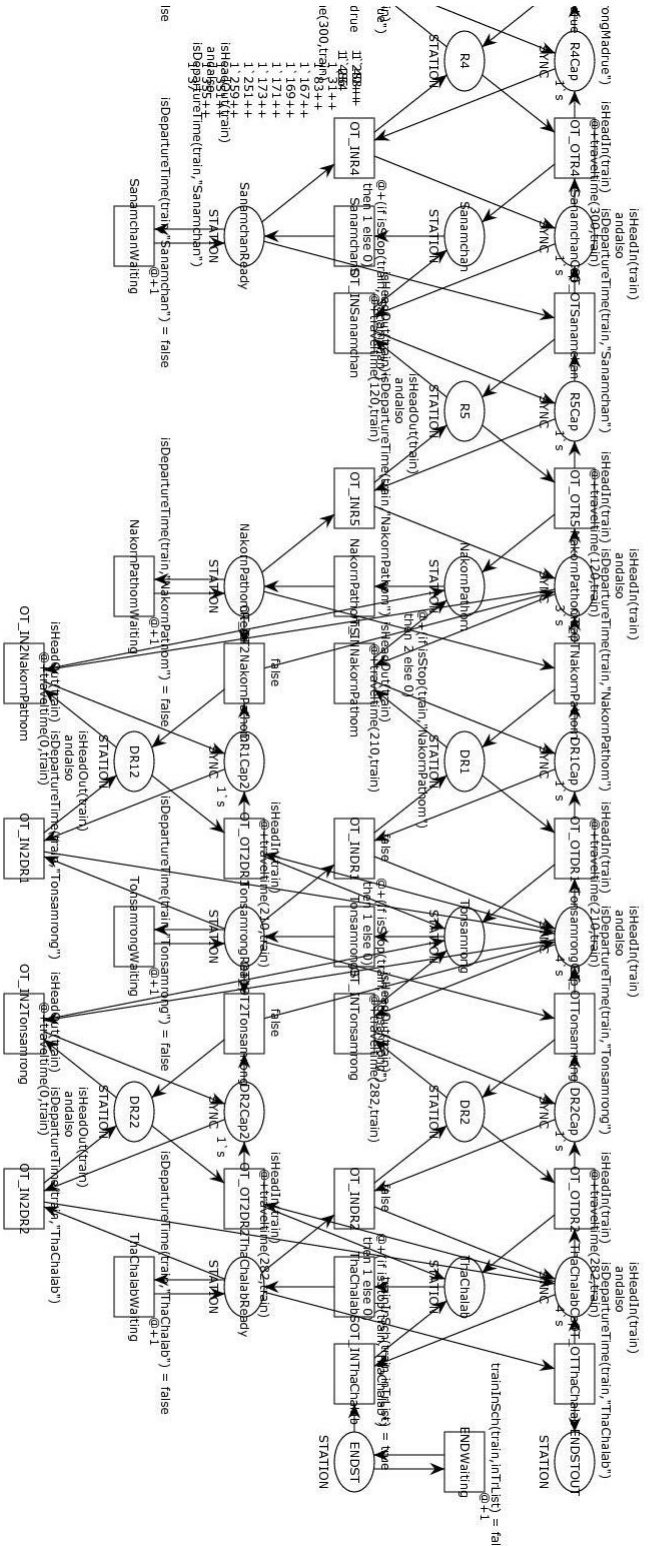
รูปที่ 4.4 ข้อมูลความเร็วรถไฟประเภทต่างๆที่จะนำมาใช้ในการสร้างแบบจำลอง

เมื่อใส่ข้อมูลทั้งหมดลงในเครื่องมือและเลือกฟังก์ชันแปลงแบบจำลองเป็นไทมด์คัลเลอร์เพทริเน็ต เครื่องมือจะทำการสร้างเอกสารนำออกนามสกุล .cpn โดยเมื่อนำไปใช้งานกับโปรแกรมเครื่องมือซีพีเอ็น สามารถแสดงผลลัพธ์แบบจำลองที่ถูกจากสร้างเครื่องมือได้ดังรูปที่ 4.5 และ 4.6 โดยมีฟังก์ชันที่ใช้ควบคุมการเคลื่อนที่ของรถไฟหรือโทเคนและข้อมูลตารางรถไฟทั้งหมด

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University



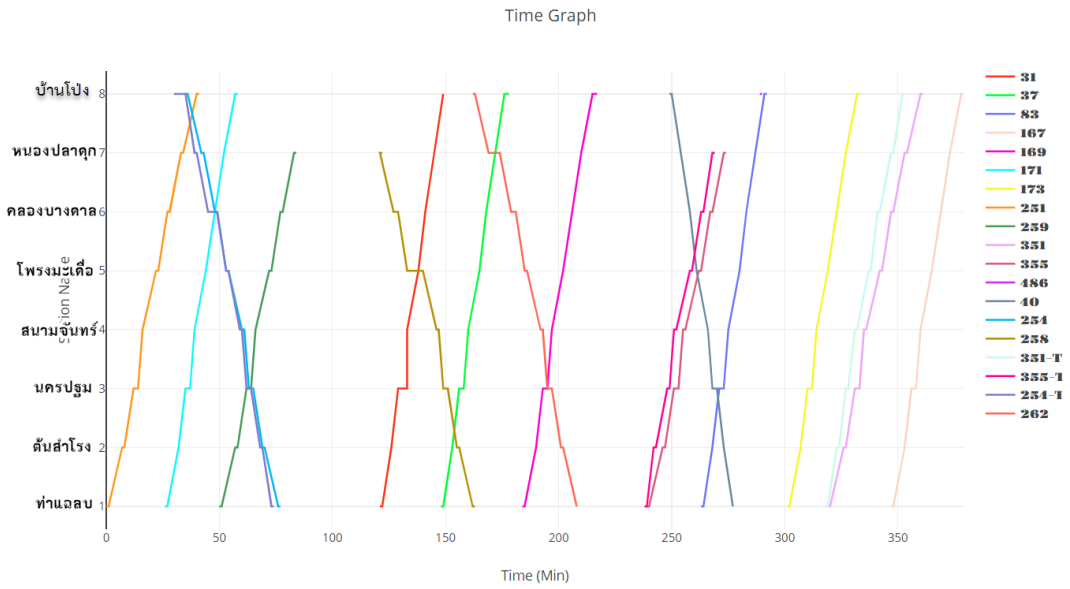
รูปที่ 4.5 แบบจำลองไทม์คัลเลอร์เพทริเน็ตจากแบบจำลองเครือข่ายทางรถไฟ 8 สถานี (ส่วนที่ 1)



รูปที่ 4.6 แบบจำลองไทม์คัลเลอร์เพทริเน็ตจากแบบจำลองเครือข่ายทางรถไฟ 8 สถานี (ส่วนที่ 2)

4.3 ผลการจำลองแบบจำลองเครือข่ายทางรถไฟ

จากการเก็บผลทดสอบจากการจำลองด้วยแบบจำลองไทมด์เพทริเน็ตจำนวน 5 รอบ ได้ผลลัพธ์ซึ่งสะท้อนถึงเวลาที่รถไฟใช้เข้าออกสถานีต่างๆได้เท่ากัน โดยเมื่อนำเข้าข้อมูลเวลาที่ได้นี้ระบุลงบนเส้นเวลาหรือ Time Graph สามารถแสดงได้ดังรูป 4.7



รูปที่ 4.7 Time Graph จากแบบจำลองและจากตารางเวลา

โดยจากผลทดสอบ พบเส้นเวลาของรถไฟ 3 ขบวนที่แตกต่างกัน เมื่อเทียบกับเส้นเวลาที่สร้างจากตารางเวลารถไฟ คือ ขบวนที่ 254, 351 และ 355 โดยมีเส้นที่สร้างจากตารางเวลารถไฟ คือ 254-T สำหรับขบวนที่ 254, 351-T สำหรับขบวนที่ 351 และ 355-T สำหรับขบวนที่ 355 ส่วนขบวนอื่นๆ เส้นเวลาของรถไฟค่อยข้างใกล้เคียงกับเวลาที่ระบุไว้ในตารางรถไฟ

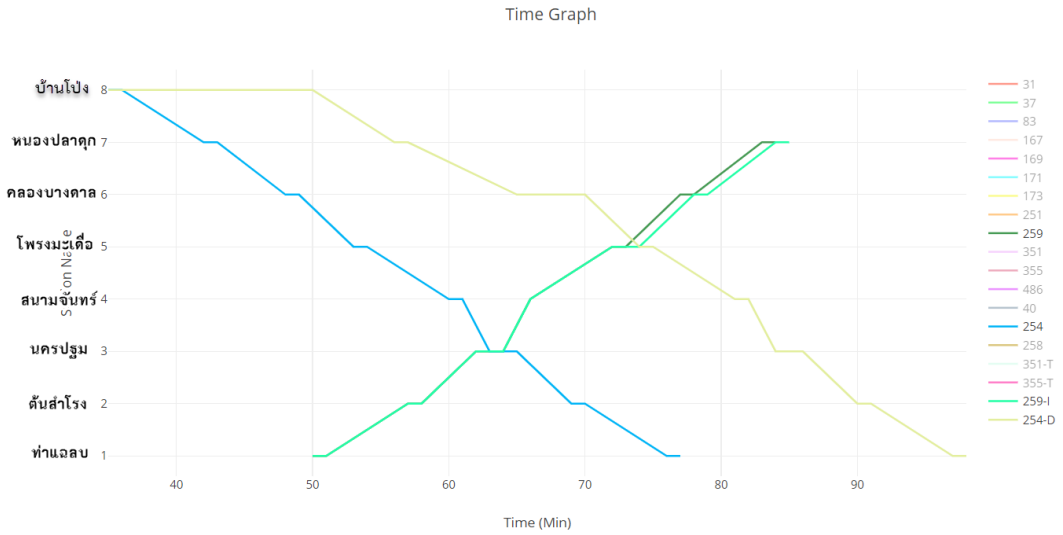
4.4 ผลการทดลองหลังปรับแต่งแบบจำลองเครือข่ายทางรถไฟ

4.4.1 ผลการปรับแต่งตารางเวลารถไฟครั้งที่ 1

เมื่อทำการทดลองปรับแต่งเวลาตารางรถไฟ โดยเลือกรถไฟขบวนที่ 254 กำหนดให้มีความล่าช้าเกิดขึ้นเพื่อทดสอบผลกระทบที่เกิดขึ้นในแบบจำลอง โดยตารางใหม่ที่ได้แสดงดังตารางที่ 4.3

ตารางที่ 4.3 ตารางเวลารถไฟใหม่ของรถไฟขบวนที่ 254

รถธรรมดา เลขที่ 254 : หลังสวน - ธนบุรี				เวลาเดิม
ลำดับ	สถานีที่หยุด	ถึง	ออก	ถึง
73	บ้านโป่ง	14:45	14:50	14:30
74	ชุมทางหนองปลาดุก	14:59	15:00	14:39
75	คลองบางตาล	15:09	15:10	14:45
76	โพรงมะเดื่อ	15:14	15:15	14:53
77	พระราชวังสนามจันทร์	15:20	15:21	14:59
78	นครปฐม	15:23	15:25	15:02
79	ต้นสำโรง	15:29	15:30	15:08
80	ท่าแฉลบ	15:34	15:35	15:13



รูปที่ 4.8 Time Graph แสดงผลกระทบหลังปรับเวลาตารางรถไฟขบวนที่ 254

เมื่อนำเข้าข้อมูลตารางใหม่ในแบบจำลองและทำการจำลองผล ทำให้ได้ผลลัพธ์เส้นเวลาใหม่เทียบเส้นเวลาเดิมก่อนปรับตารางเวลา ดังแสดงในรูปที่ 4.8 โดยเส้นเวลาใหม่ 254-D แสดงเส้นเวลาของผลลัพธ์การจำลองจากตารางเวลารถไฟใหม่ของขบวนที่ 254 จะพบว่า รถไฟขบวนที่ 259 จะได้รับผลกระทบเล็กน้อยจากการปรับแบบจำลอง โดยจะทำให้เกิดความล่าช้าขึ้นบริเวณสถานีพระราชวังสนามจันทร์ รถไฟขบวนอื่นๆไม่พบการเปลี่ยนแปลงหรือผลกระทบจากการปรับแบบตารางเวลารถไฟใหม่นี้

4.4.2 ผลการปรับแต่งตารางเวลารถไฟครั้งที่ 2

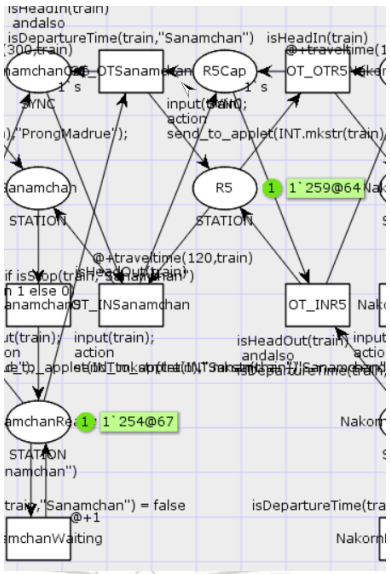
ทดลองปรับตารางเวลารถไฟขบวนที่ 254 อีกครั้ง โดยลดเวลาล้าช้าลงเล็กน้อยจากตารางเวลารถไฟปกติ จะได้ตารางรถไฟใหม่ดังตารางที่ 4.4

ตารางที่ 4.4 ตารางเวลารถไฟใหม่ขบวนที่ 254 (ล่าช้า)

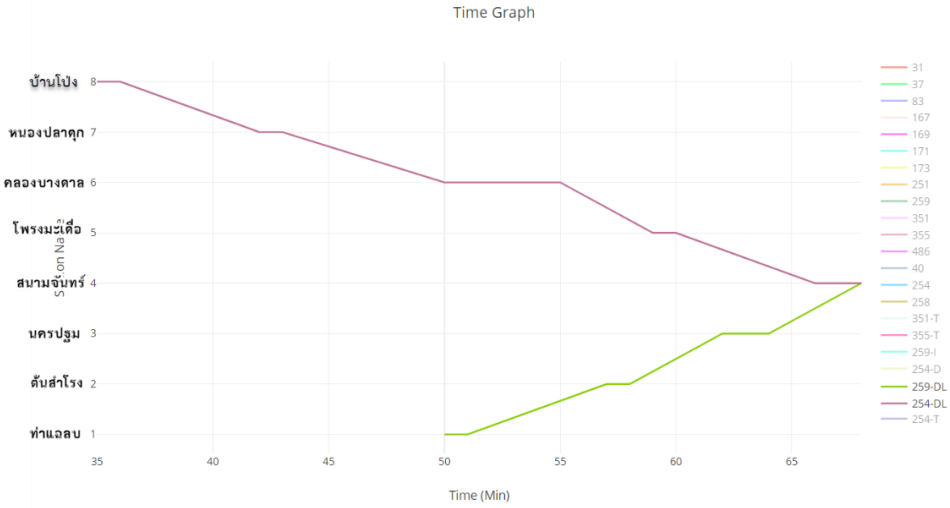
รถธรรมดา เลขที่ 254 : หลังสวน - ธนบุรี				เดิม
ลำดับ	สถานีที่หยุด	ถึง	ออก	ถึง
73	บ้านโป่ง	14:30	14:35	14:30
74	ชุมทางหนองปลาดุก	14:44	14:45	14:39
75	คลองบางตาล	14:51	14:55	14:45
76	โพรงมะเดื่อ	14:59	15:00	14:53
77	พระราชวังสนามจันทร์	15:05	15:06	14:59
78	นครปฐม	15:08	15:10	15:02
79	ต้นสำโรง	15:14	15:15	15:08
80	ท่าแฉลบ	15:19	15:20	15:13

โดยผลลัพธ์ที่ได้จะพบการติดตาย (Deadlock) ในแบบจำลองเกิดขึ้น การติดตายนี้เกิดขึ้นระหว่างรางรถไฟประเภทรางเดี่ยวจากสถานีนครปฐมและสถานีพระราชวังสนามจันทร์ ซึ่งสถานีพระราชวังสนามจันทร์นี้อยู่ห่างจากสถานีนครปฐม 2 กิโลเมตร ซึ่งไม่มีจุดสลับรางและสถานีมีเพียงหนึ่งชานชาลา โดยรถเมื่อรถไฟขบวนที่ 254 ที่กำลังมุ่งหน้าเข้าสถานีกรุงเทพ ได้เข้าจอดที่สถานีพระราชวังสนามจันทร์และพร้อมจะออกจากสถานีแล้วในช่วงนาทีที่ 66-67 ของการจำลอง แต่ไม่สามารถออกจากสถานีได้ เนื่องจากราง R5 ที่อยู่ติดกับสถานีถูกใช้งานด้วยขบวนรถไฟเลขที่ 259 ซึ่งกำลังมุ่งหน้าเข้าสู่สถานีพระราชวังสนามจันทร์ในทิศทางขาล่องออก ตั้งแต่ช่วงนาทีที่ 64 ของการจำลอง และไม่สามารถเข้าสู่สถานีพระราชวังสนามจันทร์ได้ เนื่องจากสถานีถูกใช้งานอยู่ด้วยรถไฟขบวนที่ 254 เช่นกัน ทำให้เกิดการติดตายขึ้นได้ในแบบจำลองดังรูปที่ 4.9 และ 4.10

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University



รูปที่ 4.9 การเกิด Deadlock ในแบบจำลองไทมด์เพทริเน็ต



รูปที่ 4.10 Time Graph การเกิด Deadlock

4.5 สรุปผลการทดลอง

จากการทดลองจำลองผลด้วยตารางข้อมูลปกติ ได้ผลลัพธ์จากการทดลองที่ใกล้เคียงกับตารางปัจจุบัน โดยความต่างของเส้นเวลาจากตารางรถไฟทั้ง 3 ขบวน มาจากตารางเวลารถไฟจริงที่กำหนดให้ระยะเวลาที่ใช้การเคลื่อนรถไฟจากสถานีหนึ่งถึงสถานีถัดไปสั้นกว่าตารางเวลารถไฟของขบวนอื่นๆ แสดงถึงความเร็วของขบวนรถที่ใช้การเคลื่อนที่ระหว่างสถานีนั้นต่างจากความเร็วรถไฟเฉลี่ยของขบวนอื่นที่ใช้ในการจำลองอย่างมีนัยยะหากรถไฟขบวนนั้นเข้าถึงสถานีต่างๆ ได้โดยไม่มีการเสียเวลาและเมื่อปรับแต่งแบบจำลองโดยการปรับตารางเวลารถไฟดังกล่าว แบบจำลองสามารถตรวจสอบผลกระทบดังแสดงในเส้นเวลาที่เปลี่ยนไปและสามารถตรวจสอบปัญหาที่อาจจะเกิดขึ้นในเครือข่ายทางรถไฟดังเช่น การติดตายที่สถานีพระราชวังสนามจันทร์

บทที่ 5
สรุปผลและข้อเสนอแนะ

5.1 สรุปผลการวิจัย

งานวิจัยได้เสนอแนวทางในออกแบบเครือข่ายทางรถไฟในมุมมองขั้นสูงด้วยไทมด์คัลเลอร์เพทริเน็ต สำหรับช่วยเหลือในการตรวจสอบความถูกต้องของตารางรถไฟและปัญหาของเครือข่ายทางรถไฟที่ออกแบบมาโดยไม่สนใจโอกาสในการเกิดความล่าช้าจากปัจจัยภายนอก เช่น เวลาปิดเปิดไม้กั้นทางรถไฟ เวลาล่าช้าจากปริมาณผู้โดยสาร เวลาล่าช้าจากขบวนรถไฟมีปัญหา เป็นต้น

งานวิจัยได้เสนอการจำลองส่วนประกอบย่อยของเครือข่ายทางรถไฟหรือเรียกว่า มอดูล เป็นมอดูลมาตรฐานไว้ 2 ประเภท คือ มอดูลสถานีและมอดูลรางรถไฟ ซึ่งมอดูลสถานีสามารถปรับแต่งค่าพารามิเตอร์จำนวนชานชาลาสำหรับรถไฟเข้าจอดได้ รวมถึงปรับแต่งจากมอดูลสถานีมาตรฐานเป็นสถานีชุมทางเพื่อรองรับรถไฟจากเส้นทางรองอื่นๆได้อีกด้วย สำหรับมอดูลรางรถไฟได้ถูกออกแบบให้รองรับรางเดี่ยวและรางคู่ แต่ในกรณีของรางคู่ข้อกำหนดการใช้งานจะถูกปรับปรุงเป็นเส้นทางเดินรถทางเดียว (Oneway) ในแต่ละราง โดยจะปรับในส่วนของเงื่อนไขหรือการ์ดอินสคริปชันให้รองรับข้อจำกัดดังกล่าว จากนั้นงานวิจัยได้นำเสนอกฎการประสานมอดูลด้วยวิธีการซ้อนทับ เพื่อประสานมอดูลที่ถูกออกแบบไว้เมื่อนำมอดูลเหล่านั้นมาเรียงต่อกันเพื่อสร้างแบบจำลองเครือข่ายทางรถไฟทั้งระบบ เพื่อให้ได้เครือข่ายทางรถไฟที่สามารถทำงานได้ตามข้อกำหนดพื้นฐานของการเดินรถและควบคุมการเคลื่อนรถไฟเข้าและออกจากสถานีต่างๆตามตารางรถไฟที่กำหนดในเบื้องต้น รวมถึงกฎการใช้งานคัลเลอร์เพทริเน็ตที่มีภาษาซีพีเอ็น เอ็มแอลกำกับไว้อยู่ด้วย

โดยเมื่อได้แบบจำลองไทมด์คัลเลอร์เพทริเน็ตที่ออกแบบสำเร็จ ได้ทำการทดสอบการทำงานของเครือข่ายทางรถไฟและตารางรถไฟด้วยโปรแกรมเครื่องมือซีพีเอ็น ผลลัพธ์การจำลองได้แสดงให้เห็นถึงเวลาที่รถไฟขบวนต่างๆเคลื่อนที่ในแบบจำลองด้วยข้อกำหนด พารามิเตอร์ต่างๆและตารางเวลาที่ระบุ แสดงผลลัพธ์ผ่านทางเส้นเวลารถไฟหรือเรียกว่า Time Graph

ผลการทดลองครั้งที่ 1 ด้วยเวลาปกติตามที่ระบุไว้บนตารางเวลารถไฟ แสดงให้เห็นรถไฟส่วนใหญ่สามารถเคลื่อนที่ได้ตามที่ระบุไว้บนตารางเวลารถไฟ มีเพียงรถไฟ 3 ขบวนที่มีเส้นเวลาที่รถไฟใช้จากแบบจำลองแตกต่างจากเส้นเวลาตารางรถไฟที่กำหนด โดยความต่างของเส้นเวลานี้เนื่องมาจากเวลาที่ถูกกำหนดเพื่อใช้ในการเดินทางของรถไฟระหว่างสถานีในตารางเวลารถไฟของขบวนเหล่านั้นสั้นกว่าปกติ ซึ่งเมื่อแบบจำลองได้กำหนดให้ใช้ความเร็วเฉลี่ยของขบวนรถแต่ละประเภท ทำให้ขบวนรถไฟดังกล่าวต้องใช้ในการเดินทางมากขึ้นจากตารางรถไฟที่กำหนดไว้เมื่อทำการจำลอง โดยหากในความเป็นจริงแล้วรถไฟขบวนดังกล่าวใช้เวลาเดินทางใกล้เคียงกับค่าความเร็วเฉลี่ยของแบบจำลองรถไฟขบวนเหล่านั้นก็อาจมีโอกาสเสียเวลาเนื่องจากการเดินทางล่าช้ากว่าที่ตารางกำหนดได้ อีกทั้ง

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

แบบจำลองสามารถแสดงผลกระทบดังแสดงด้วยเส้นเวลาที่เปลี่ยนไปของขบวนรถที่ล่าช้าและขบวนรถอื่นที่ได้รับผลกระทบจากการเสียเวลาของรถไฟขบวนนั้น ดังเช่นผลการทดลองการปรับแบบจำลองครั้งที่ 1 ด้วยการแก้ไขตารางเวลารถไฟของรถไฟขบวนเลขที่ 254 ให้เกิดความล่าช้าอย่างมากจากตารางเวลาปกติในหลายสถานี ทำให้เส้นเวลารถไฟถูกเลื่อนไปทางด้านขวามือและกระทบกับเส้นเวลารถไฟของรถไฟขบวนเลขที่ 259 ที่ต้องจอดรอที่สถานีโพรงมะเดื่อเนื่องจากรางรถไฟถูกใช้งานโดยรถไฟขบวนเลขที่ 254 อยู่ ซึ่งหากการจราจรในระบบรางมีความหนาแน่นมากกว่าตัวอย่างที่ยกมาอาจจะทำให้เกิดผลกระทบต่อรถไฟอีกหลายขบวนหรือปัญหาอื่นๆก็สามารถเกิดขึ้นได้เช่นกัน

ผลการทดลองการปรับแบบจำลองครั้งที่ 2 แสดงให้ถึงปัญหาในเครือข่ายทางรถไฟของตัวอย่างการทดลอง ซึ่งเมื่อปรับตารางเวลารถไฟขบวนเลขที่ 254 ให้มีความล่าช้าเล็กน้อยจากตารางปกติ ส่งผลให้นอกจากรถไฟขบวนเลขที่ 254 ล่าช้าแล้ว ยังเกิดปัญหาการติดตายกับรถไฟขบวนเลขที่ 259 ที่บริเวณสถานีพระราชวังสนามจันทร์ เนื่องจากสถานีนี้เป็นสถานีที่ตั้งอยู่บนรางรถไฟรางเดี่ยวและไม่มีชานชาลาสำหรับรองรับหรือจุดพักรถไฟ ทำให้หากปล่อยรถไฟให้ใช้งานปกติโดยขาดการติดตามและควบคุม เมื่อมีรถไฟ 2 ขบวนที่มีทิศทางขาล่องต่างกัน ขอเข้าใช้งานสถานีนี้ในเวลาไล่เลี่ยกันจะทำให้การติดตายเกิดขึ้นบริเวณดังกล่าวได้ จึงต้องมีการพิจารณาตารางเวลารถไฟสำหรับสถานีนี้หรือสถานีที่อยู่ติดกันเป็นพิเศษ หรือทำการปรับแก้ข้อกำหนดการเดินรถในบริเวณสถานีดังกล่าวและสถานีที่อยู่ติดกัน เพื่อป้องปัญหาที่อาจจะเกิดขึ้นได้ สุดท้ายงานวิจัยได้พัฒนาเครื่องมือเพื่อแปลงแบบจำลองเครือข่ายทางรถไฟซึ่งประกอบด้วยมอดูลที่ออกแบบในงานวิจัยโดยแสดงเป็นสัญลักษณ์ที่นำมาเรียงต่อกันเป็นเส้นตรง เพื่อสร้างแบบจำลองและทำการจำลองเครือข่ายทางรถไฟสำหรับเส้นทางหลักที่ระบุไปเป็นไทมด์คัลเลอร์เพทริเน็ต โดยเครื่องมือจะปรับเปลี่ยนแบบจำลองที่ออกแบบ, ตารางเวลารถไฟทุกขบวน และพารามิเตอร์พื้นฐานที่ถูกกำหนดในเครื่องมือทั้งหมดไปเป็นข้อมูลนำเข้าสำหรับสร้างเอกสารนำออกที่มีนามสกุลเป็น .cpn โดยเอกสารนำออกนี้มีรูปแบบพื้นฐานเอ็กซ์เอ็มแอลที่มีโครงสร้างตามข้อกำหนดของโปรแกรมเครื่องมือซีพีเอ็น และสามารถทดสอบผลได้ด้วยโปรแกรมเครื่องมือซีพีเอ็น โดยผู้ใช้ไม่ต้องออกแบบเพิ่มเติมอีก

โดยสรุปงานวิจัยได้เสนอแนวทางในการสร้างแบบจำลองเครือข่ายทางรถไฟด้วยไทมด์คัลเลอร์เพทริเน็ต ซึ่งแบบจำลองดังกล่าวสามารถช่วยเหลือในการตรวจสอบตารางเวลาที่ถูกออกแบบมาว่าสามารถใช้งานได้จริงหรือไม่ รวมถึงแบบจำลองดังกล่าวยังสามารถแสดงจุดที่อาจจะปัญหาในเครือข่ายทางรถไฟ เพื่อให้ผู้ออกใช้งานอย่างระมัดระวังหรือแก้ปัญหาดังกล่าวก่อนนำไปใช้งานอีกด้วย และเครื่องมือที่พัฒนาในงานวิจัยสามารถช่วยเหลือผู้ใช้งานในการสร้างแบบจำลองเครือข่ายทางรถไฟและแปลงแบบจำลองดังกล่าวเป็นไทมด์คัลเลอร์เพทริเน็ตได้

5.2 ข้อเสนอแนะ

ในงานวิจัยได้ออกแบบและจำลองส่วนประกอบหรือมอดูลของเครือข่ายทางรถไฟไว้เพียง 2 ประเภท คือ สถานีและรางรถไฟ โดยมอดูลสถานีได้กำหนดให้มีความสามารถรองรับทั้งสถานีธรรมดาและสถานีชุมทางในการสร้างแบบจำลองไทมด์คัลเลอร์เพทริเน็ตเพื่อตรวจสอบการใช้งานเครือข่ายทางรถไฟ มอดูลที่ถูกออกแบบมานั้นสามารถปรับปรุงให้ขยายความสามารถในการสร้างแบบจำลองได้ทั้งในเส้นทางหลักและเส้นทางรองๆหลายเส้นทาง โดยสามารถปรับแต่งที่มอดูลสถานีได้ แต่เครื่องมือที่ถูกพัฒนาสามารถสร้างแบบจำลองในลักษณะเส้นทางตรง จึงทำให้แบบจำลองที่ได้จะมุ่งเน้นถึงการใช้งานในเส้นทางหลักในกรณีที่มีสถานีชุมทางเข้ามาเกี่ยวข้อง ขบวนรถไฟที่มาจากเส้นทางรองจะเริ่มการจำลองด้วยเวลารถไฟตามตารางรางรถไฟที่เข้าถึงสถานีชุมทางแล้วเท่านั้น โดยไม่ได้สนใจเหตุการณ์ที่เกิดขึ้นในระหว่างเส้นทางก่อนถึงสถานีชุมทางของขบวนรถไฟนั้นเลย

ในอนาคต งานวิจัยนี้สามารถปรับปรุงแบบจำลองให้รองรับการออกแบบในกรณีหลากหลายเส้นทางได้ และปรับเปลี่ยนการออกแบบเครือข่ายไทมด์คัลเลอร์เพทริเน็ตเพื่อให้อ่านได้ง่ายขึ้น โดยจะเห็นว่าแบบจำลองค่อนข้างมีความซับซ้อน เนื่องจากแบบจำลองที่ออกในงานวิจัยไม่รองรับการใช้งาน Substitution Transitions และการประสานมอดูลต่างๆ ต้องผ่านฎการซ้อนทับเพื่อสร้างแบบจำลองไทมด์คัลเลอร์เพทริเน็ต รวมถึงการพัฒนาเครื่องมือให้มีความสามารถรองรับการออกแบบที่หลากหลายด้วยเช่นกัน และมอดูลของส่วนประกอบรถไฟยังสามารถพัฒนาหรือเพิ่มเติมได้อีก เพื่อรองรับอุปกรณ์และส่วนประกอบอื่นของเครือข่ายทางรถไฟที่ยังไม่กล่าวถึง เป็นต้น

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

บรรณานุกรม

1. Harper, R., ed. *Programming in Standard ML*. Version 2 ed. 2001.
2. sml-family.org. *The Standard ML Basis Library*. 2002 June 6, 2002 [cited 2018 August 26, 2018]; Available from: http://sml-family.org/Basis/.
3. Fletner, P., *ML Standard Chapter 2*, in *ML Standard*. 2004: IT Department, Sweden's First University.
4. CPN-Tools. *CPN Tools*. 2016 [cited 2018 August 26, 2018]; CPN Tools]. Available from: http://cpntools.org/
5. Bozek, A., *Using Timed Coloured Petri Nets for Modelling, Simulation and Scheduling of Production Systems*. 2012. p. 207-230.
6. Jensen, K., L.M. Kristensen, and L. Wells, *Coloured Petri Nets and CPN Tools for modelling and validation of concurrent systems.* Software Tools for Technology Transfer, 2007. **9**(3-4): p. 213-254.
7. Westergaard, M. and T. Slaats, *CPN Tools 4: A Process Modeling Tool Combining Declarative and Imperative Paradigms*, in *11th International Conference on BPM2013*. 2013: Beijing, China.
8. Choppy, C., L. Petrucci, and G. Reggio, *A Modelling Approach with Coloured Petri Nets*, in *International Conference on Reliable Software Technologies*. 2008: Venice, Italy. p. 73-86.
9. Giglio, D. and N. Sacco, *A Petri net model for analysis, optimisation, and control of railway networks and train schedules*, in *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*. 2016: Rio de Janeiro, Brazil.
10. Sakaguchi, T. and N. Tomi. *A train traffic model based on coloured Petri Nets and its application to a train schedule planning system*. 1996.
11. Deesukying, J. and W. Vatanawood, *Generating of business rules for Coloured Petri Nets*, in *IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS)*. 2016: Okayama, Japan.
12. *State Railway of Thailand Website*. 2018 [cited 2018 July 4, 2018]; Available from:

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

http://www.railway.co.th/.

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ประวัติผู้เขียน

ชื่อ-สกุล	นายนิติพัฒน์ ทรงวิโรจน์
วัน เดือน ปี เกิด	05 มิถุนายน 2528
สถานที่เกิด	กรุงเทพมหานคร
วุฒิการศึกษา	ปริญญาตรี คณะวิศวกรรมศาสตร์ สาขาคอมพิวเตอร์ มหาวิทยาลัยมหิดล
ที่อยู่ปัจจุบัน	263/58 หมู่1 ตำบลลำผักกูด อำเภอธัญบุรี จังหวัดปทุมธานี 12110
ผลงานตีพิมพ์	ส่วนหนึ่งจากงานวิจัยได้รับการตีพิมพ์ในรูปแบบบทความวิชาการในหัวข้อเรื่อง “Railway Network Modeling Using Building Block Of Timed Coloured Petri Nets” โดยนิติพัฒน์ ทรงวิโรจน์, วิวัฒน์ วัฒนาวุฒิ, และ สมศักดิ์ วาณิชอนันต์ชัย ในงานประชุมวิชาการ 2018 4th IEEE International Conference on Computer and Communications (ICCC) จัดขึ้นโดย International Conference on Computer and Communications (ICCC) ระหว่างวันที่ 7-10 ธันวาคม 2561 ณ เมืองเฉิงตู ประเทศสาธารณรัฐประชาชนจีน

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University