การบริหารจัดการระบบโครงข่ายด้วยวิธีรีอินฟอร์สเมนท์เลิร์นนิ่ง

นายอรุณรัตน์ นูพลกรัง

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมโทรคมนาคม
มหาวิทยาลัยเทคโนโลยีสุรนารี
ปีการศึกษา 2549
ISBN 974-533-581-9

NETWORK MANAGEMENT USING REINFORCEMENT LEARNING

Arunratn Noophoolkrang

A Thesis Submitted in Partial Fulfillment of the Requirements for the

Degree of Master of Engineering in Telecommunication Engineering

Suranaree University of Technology

Academic Year 2006

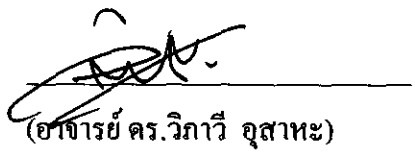
ISBN 974-533-581-9

การบริหารจัดการระบบโครงข่ายด้วยวิธีรีอินฟอร์สเมนท์เลิร์นนิ่ง

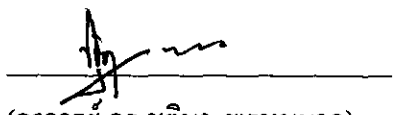
มหาวิทยาลัยเทคโนโลยีสุรนารี อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญามหาบัณฑิต

คณะกรรมการสอบวิทยานิพนธ์

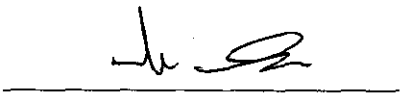
(อาจารย์ ดร.รังสรรค์ ทองทา)
ประธานกรรมการ



(อาจารย์ ดร.วิภาวี อุสาหะ)
กรรมการ (อาจารย์ที่ปรึกษาวิทยานิพนธ์)



(อาจารย์ ดร.ชุติมา พรหมมาก)
กรรมการ



(รศ. ดร.เสาวณีย์ รัตนพานี)
รองอธิการบดีฝ่ายวิชาการ

(รศ. น.อ. ดร.วรพจน์ ขำพิศ)
คณบดีสำนักวิชาวิศวกรรมศาสตร์

อรุณรัตน์ นูพลกรัง : การบริหารจัดการระบบโครงข่ายด้วยวิธีรีอินฟอร์สเมนท์เลิร์นนิ่ง
(NETWORK MANAGEMENT USING REINFORCEMENT LEARNING)
อาจารย์ที่ปรึกษา : อาจารย์ ดร.วิภาวี อุสาหะ, 72 หน้า.ISBN 974-533-581-9

ในปัจจุบันระบบโครงข่ายคอมพิวเตอร์มีความสำคัญมาก ดังนั้นเสถียรภาพของโครงข่ายคอมพิวเตอร์จึงเป็นสิ่งที่มีความสำคัญเป็นอย่างยิ่ง จึงจำเป็นต้องมีระบบบริหารจัดการโครงข่าย (Network Management System หรือ NMS) มาทำหน้าที่ในการติดตาม ตรวจสอบการทำงานของระบบโครงข่าย ซึ่งประกอบด้วย 5 ส่วนที่สำคัญคือ 1. ระบบบริหารจัดการอุปกรณ์ขัดข้อง (fault management) 2. ระบบบริหารจัดการสมรรถนะ (performance management) 3. ระบบบริหารจัดการการตั้งค่าของอุปกรณ์ (configuration management) 4. ระบบบริหารจัดการบัญชีผู้ใช้ (accounting management) 5. ระบบบริหารจัดการความปลอดภัย (security management) ซึ่งในงานวิจัยนี้ได้มุ่งเน้นพิจารณาในส่วนของการบริหารจัดการอุปกรณ์ขัดข้อง และการบริหารจัดการสมรรถนะ โดยวิธีการเฝ้าตรวจสถานะของโครงข่าย (network monitoring) ซึ่งเป็นการตรวจสอบสถานะการทำงานของอุปกรณ์ รวมทั้งการค้นหาตำแหน่งของอุปกรณ์ที่ขัดข้อง (fault localization) การทำงานของระบบดังกล่าวนั้น จะใช้วิธีในการโพลล์ (polling) ไปยังอุปกรณ์ที่อยู่ในระบบโครงข่ายและรอรับผลการรายงานสถานะกลับมา และต้องทำการโพลล์เพื่อตรวจสอบสถานะลักษณะนี้อยู่ตลอดเวลา ซึ่งกิจกรรมนี้ทำให้เกิด โพลล์ลิ่งโอเวอร์เฮด (polling overhead) มีผลให้แบนด์วิดท์ (bandwidth) บางส่วนของโครงข่ายต้องถูกใช้ไปมากกว่าที่จำเป็น นอกจากนั้นข้อมูลที่ได้จากการโพลล์อาจไม่ถูกต้องครบถ้วนหรือสิ่งที่ได้จากการสังเกตไม่มีความชัดเจน (partial observability) จึงไม่เพียงพอในการใช้ตัดสินใจถึงสถานะที่แท้จริงของอุปกรณ์นั้นงานวิจัยนี้จึงมีจุดประสงค์ที่จะพัฒนาขั้นตอนวิธี (algorithm) ที่ใช้สำหรับการเฝ้าตรวจสถานะโครงข่ายที่มีโพลล์ลิ่งโอเวอร์เฮดต่ำและสามารถค้นหาจุดขัดข้องได้อย่างถูกต้อง ภายใต้สภาวะที่มีข้อมูลที่ได้จากการสังเกตไม่ชัดเจน ทั้งยังมีความซับซ้อนต่ำ (low complexity) ด้วยการประยุกต์ใช้วิธีรีอินฟอร์สเมนท์เลิร์นนิ่ง (Reinforcement Learning หรือ RL) วิธีหนึ่งที่เรียกว่าออนโพลิซีมอนติคาร์โล (On-policy Monte Carlo หรือ ONMC) ที่สามารถเรียนรู้วิธีการตัดสินใจที่ดีในสภาวะแวดล้อมที่มีข้อมูลอยู่เพียงบางส่วน โดยเริ่มทำการศึกษากรณีที่โครงข่ายมีขนาดเล็ก จากผลการทดลองด้วยโปรแกรมจำลองแบบ (simulation) พบว่าวิธีการดังกล่าวสามารถลดปริมาณโพลล์ลิ่งโอเวอร์เฮดได้ระหว่าง 33 % ถึง 86 % เมื่อเปรียบเทียบกับการใช้วิธีการแบบโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ (proactive network management) และทำการศึกษากรณีที่

โครงข่ายที่มีขนาดใหญ่ขึ้น พบว่าวิธีออนโพลิซีมอนติคาร์โลยังคงสามารถลดปริมาณโพลล์ลิ่งโอเวอร์เฮดได้ถึง 88 % รวมทั้งสามารถลดความซับซ้อนของขั้นตอนวิธีลงได้ เมื่อเปรียบเทียบกับแบบโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์

สาขาวิชาวิศวกรรมโทรคมนาคม ลายมือชื่อนักศึกษา _____

ปีการศึกษา 2549 ลายมือชื่ออาจารย์ที่ปรึกษา _____

ARUNRATN NOOPHOOLKRANG: NETWORK MANAGEMENT USING REINFORCEMENT LEARNING. THESIS ADVISOR: WIPAWEE USAHA, Ph.D. 72 PP. ISBN 974-533-581-9

REINFORCEMENT LEARNING/NETWORK MONITORING/FAULT LOCALIZATION

Computer networks currently play an important role and therefore require high stability. Network management is thus needed to control its stability. Network management consists of five elements which are fault management, performance management, configuration management, accounting management and security management. In this thesis, the emphasis is placed on fault management and performance management which involves two tasks, namely, network monitoring and fault localization. Network monitoring requires polling the target device and receiving response from the device. The process is repeated continuously. As a result, a considerable amount of traffic is generated, called polling overhead. Bandwidth which would otherwise be used for actual data transfer, is wasted on accommodating polling overhead. Furthermore, due to possible delays and malfunctioning devices, the information received from the polling process may be incomplete, obsolete or even incorrect. Thus the information obtained from polling can be considered as only a partial observation of the network status. Based on such incomplete information, the network management must decide if the network status is normal or abnormal. If the latter case is observed, the network management must decide whether to repair the device or poll for more information in locating the faulty device.

Therefore, the underlying aim of this thesis is to develop a network monitoring and fault localization algorithm which performs well and requires low polling overhead and low complexity, under a partial observable environment. The proposed method is based on a reinforcement learning (RL) technique called the on-policy Monte Carlo (ONMC) method. In a recent research, it is shown that this method is able to learn to make good decisions under partial observation scenarios. Numerical results are obtained for the small network and large network cases. Simulation results from the two cases show that ONMC can reduce polling overhead between 33 % to 86 % and up to 88 %, respectively, when compared with existing polling schemes.

School of Telecommunication Engineering Student's Signature Arunrat n

Academic Year 2006 Advisor's Signature

กิตติกรรมประกาศ

วิทยานิพนธ์นี้สำเร็จลุล่วงด้วยดี เนื่องจากได้รับความช่วยเหลืออย่างดียิ่ง ทั้งด้านวิชาการและด้านการดำเนินงานวิจัย จากบุคคลและกลุ่มบุคคลต่างๆ ได้แก่

อาจารย์ ดร.วิภาวี อุสาหะ อาจารย์ประจำสาขาวิชาวิศวกรรมโทรคมนาคม มหาวิทยาลัยเทคโนโลยีสุรนารี อาจารย์ที่ปรึกษาวิทยานิพนธ์ ที่ให้โอกาสทางการศึกษา ให้คำแนะนำปรึกษา ช่วยแก้ปัญหา และให้กำลังใจแก่ผู้วิจัยมาโดยตลอด รวมทั้งช่วยตรวจทาน และแก้ไขวิทยานิพนธ์เล่มนี้จนเสร็จสมบูรณ์

ผู้ช่วยศาสตราจารย์ ดร.รังสรรค์ วงศ์สรรค์ ผู้ช่วยอธิการบดี มหาวิทยาลัยเทคโนโลยีสุรนารี ที่ให้โอกาสในการศึกษา คำปรึกษาด้านวิชาการ และให้กำลังใจแก่ผู้วิจัยมาโดยตลอด

อาจารย์ ดร.รังสรรค์ ทองทา หัวหน้าสาขาวิศวกรรมโทรคมนาคม มหาวิทยาลัยเทคโนโลยีสุรนารี ที่ให้คำปรึกษาอันเป็นประโยชน์ต่องานวิจัย และให้กำลังใจแก่ผู้วิจัยมาโดยตลอด

อาจารย์ ดร.ชุติมา พรหมมาก อาจารย์ ดร.ชาญชัย ทองโสภา และอาจารย์ ปิยาภรณ์ กระฉอดนอก อาจารย์ประจำสาขาวิชาวิศวกรรมโทรคมนาคม มหาวิทยาลัยเทคโนโลยีสุรนารี ที่กรุณาให้คำปรึกษาด้านวิชาการ และให้กำลังใจมาโดยตลอด

ขอขอบคุณ คุณประพล จาระตะคุ วิศวกรศูนย์เครื่องมือ ที่ช่วยอำนวยความสะดวกทางด้านเครื่องมือและอุปกรณ์ ขอขอบคุณ คุณปิติพงศ์ ชาญโลหะ คุณกาญจน์กมล มณีนิล คุณวิภาดา นฤพิพัฒน์ คุณพิชญ์นันท์ สุขวัชนี และพี่น้องบัณฑิตศึกษาทุกท่าน ที่ให้คำปรึกษาด้านวิชาการและให้กำลังใจมาโดยตลอด

ขอขอบคุณ คุณวิมลรัตน์ นูพลกรัง พี่ชาย และคุณศราวุฒิ นูพลกรัง น้องชาย ที่ช่วยดูแลครอบครัวในระหว่างที่ศึกษา และให้กำลังใจมาโดยตลอด

ขอขอบคุณ ธนาคารเพื่อการเกษตรและสหกรณ์การเกษตร ที่ให้โอกาสในการลาศึกษาต่อและสนับสนุนค่าใช้จ่ายระหว่างศึกษา จนสำเร็จการศึกษาด้วยดี ขอขอบคุณ สถาบันวิจัยและพัฒนา มหาวิทยาลัยเทคโนโลยีสุรนารี ที่ให้ทุนสนับสนุนในการจัดทำวิทยานิพนธ์

สำหรับคุณงามความดีอันใดที่เกิดจากวิทยานิพนธ์เล่มนี้ ผู้วิจัยขอมอบให้กับบิดา มารดา ซึ่งเป็นที่รักและเคารพยิ่ง ตลอดจนครูอาจารย์ที่เคารพทุกท่าน ที่ได้ประสิทธิ์ประสาทวิชาความรู้และถ่ายทอดประสบการณ์ที่ดีให้แก่ผู้วิจัยตลอดมา จนทำให้ประสบความสำเร็จในชีวิต

อรุณรัตน์ นูพลกรัง

สารบัญ

หน้า

สารบัญ (ต่อ)

หน้า

สารบัญ (ต่อ)

หน้า

สารบัญตาราง

ตารางที่ **หน้า**

สารบัญรูป

รูปที่ **หน้า**

คำอธิบายสัญลักษณ์และคำย่อ

NMS	=	network management system
SNMP	=	simple network management protocol
RL	=	reinforcement learning
ONMC	=	on-policy Monte Carlo
MP	=	Markov property
MDP	=	Markov decision process
POMDP	=	partially observable Markov decision process
s	=	state
a	=	action
r	=	reward
s'	=	next state
a'	=	next action
A	=	action space
S	=	state space
o	=	observation
O	=	set of observation
E	=	expect value
V^*	=	optimal state-value function
Q^*	=	optimal action-value function
π	=	policy π
t	=	time
g	=	reward function
R	=	long term reward
H	=	history
B	=	set of all the distributions over state space S
$\bar{\mathbf{b}}$	=	belief state vector
$\bar{b}$	=	belief state

คำอธิบายสัญลักษณ์และคำย่อ (ต่อ)

ε	=	greedy policy
a^*	=	action with arg max
$Q(s,a)$	=	value function state action pair
$Q(o,a)$	=	value function observation action pair
$Q^{\pi}(o,a)$	=	value function observation action pair with policy π

บทที่ 1
บทนำ

1.1 ความสำคัญและที่มาของปัญหาการวิจัย

ในปัจจุบันมีการประยุกต์ใช้ระบบสารสนเทศ ให้เข้ามามีบทบาทในชีวิตประจำวันของเรามากขึ้น ทำให้ระบบโครงข่ายคอมพิวเตอร์ซึ่งเป็นโครงสร้างพื้นฐานที่รองรับการทำงานของระบบสารสนเทศมีความสำคัญมากยิ่งขึ้นเช่นกัน โครงข่ายเหล่านี้ได้รับการติดตั้งแพร่ขยายไปยังพื้นที่ต่างๆ ขยายวงกว้างออกไปอย่างไม่หยุดยั้ง (Han, Ahn, and Chung, 2001) และรองรับระบบงานที่มีความสำคัญ เช่น ระบบทะเบียนราษฎร์ ระบบฐานข้อมูลของกรมตำรวจ ระบบการเงินการธนาคาร เป็นต้น

จากความสำคัญของระบบโครงข่ายคอมพิวเตอร์ที่ได้กล่าวมาข้างต้นแล้ว เสถียรภาพของโครงข่ายคอมพิวเตอร์จึงเป็นสิ่งที่มีความสำคัญเป็นอย่างยิ่ง จึงจำเป็นต้องนำเอาระบบบริหารจัดการโครงข่าย (Network Management System หรือ NMS) (He, 2003) มาทำหน้าที่ในการบริหารจัดการติดตามตรวจสอบสถานภาพการทำงานของโครงข่าย ซึ่งระบบบริหารจัดการโครงข่ายนี้จะประกอบด้วย 5 ส่วนที่สำคัญคือ (He, 2003), (Su, 2002)

1. ระบบบริหารจัดการอุปกรณ์ขัดข้อง (fault management) เป็นระบบที่มีความสำคัญมาก โดยจะประกอบด้วยหน้าที่ ที่สำคัญอยู่ 2 ส่วนคือ ทำหน้าที่เฝ้าตรวจสถานะของโครงข่าย (network monitoring) เพื่อให้ทราบสถานะของอุปกรณ์อยู่โดยตลอดซึ่งหากพบอุปกรณ์ที่ขัดข้องจะได้ดำเนินการซ่อมหรือแก้ไขได้ในทันทีและการทำหน้าที่ค้นหาตำแหน่งของอุปกรณ์ที่ขัดข้อง (fault localization) ซึ่งต้องสามารถระบุตำแหน่งที่มีอุปกรณ์ขัดข้องเพื่อดำเนินการแก้ไขได้อย่างถูกต้อง งานวิจัยนี้ได้มุ่งเน้นถึงความสำคัญของทั้งสองส่วนนี้

2. ระบบบริหารจัดการสมรรถนะ (performance management) ทำหน้าที่ควบคุมสมรรถนะในการทำงานของโครงข่าย เช่น การรับรองคุณภาพการให้บริการ (quality of service) การจัดลำดับความสำคัญของกิจกรรมที่เกิดขึ้นบนโครงข่าย (class) หรือการรับรองประสิทธิภาพในการให้บริการ (service level agreement) ที่ต้องรับประกันระยะเวลาในการให้บริการให้ได้ตามที่ตกลงไว้กับผู้รับบริการซึ่งในงานวิจัยนี้จะเป็นการพัฒนาให้ได้สมรรถนะของระบบโครงข่าย ตามเป้าหมายที่วางไว้ด้วยการค้นหาตำแหน่งอุปกรณ์ที่ขัดข้องและทำการแก้ไขได้อย่างรวดเร็ว

3. ระบบบริหารจัดการการตั้งค่าของอุปกรณ์ (configuration management) ทำหน้าที่อำนวยความสะดวกในการตั้งค่าของอุปกรณ์ต่างๆ ที่ติดตั้งอยู่ภายในโครงข่ายเพื่อให้อุปกรณ์สามารถเชื่อมต่อ

ถึงกันและสามารถให้บริการได้ตามที่ต้องการ

4. ระบบบริหารจัดการบัญชีผู้ใช้ (accounting management) ทำหน้าที่บริหารจัดการข้อมูลผู้ใช้ เช่น ปริมาณการถ่ายโอนข้อมูล สิทธิ์ในการใช้งานและเข้าถึงข้อมูลของผู้ใช้งานในโครงข่ายแต่ละบุคคลแต่ละกลุ่มหรือแต่ละองค์กร

5. ระบบบริหารจัดการความปลอดภัย (security management) ทำหน้าที่ดูแลและปกป้องอุปกรณ์ของโครงข่าย และข้อมูลที่ถ่ายโอนบนโครงข่ายเพื่อไม่ให้โครงข่ายถูกทำลายถูกบุกรุก หรือถูกรบกวนจากบุคคลอื่น

ในงานวิจัยนี้ได้มุ่งเน้นพิจารณาในส่วนของ ระบบบริหารจัดการอุปกรณ์ขัดข้อง (fault management) และระบบบริหารจัดการสมรรถนะ (performance management) เนื่องจากระบบทั้งสองเป็นส่วนสำคัญมากกับเสถียรภาพของโครงข่าย หากมีอุปกรณ์ในโครงข่ายขัดข้องขึ้นจะส่งผลกระทบต่อผู้ใช้ในทันที และหากเป็นอุปกรณ์ที่เชื่อมต่อในตำแหน่งที่มีความสำคัญสูงเมื่ออุปกรณ์เกิดขัดข้องขึ้นย่อมส่งผลกระทบต่อผู้ใช้เป็นจำนวนมาก ในทางกลับกันหากอุปกรณ์ดังกล่าวสามารถทำงานได้ตลอดเวลาจะสามารถให้บริการได้อย่างต่อเนื่อง ดังนั้น การเฝ้าตรวจสถานะของโครงข่ายและการค้นหาตำแหน่งที่อุปกรณ์ขัดข้อง จึงมีความสำคัญต่อการดำรงอยู่ได้ของระบบโครงข่ายโดยมีผู้ที่ให้ความสำคัญในการศึกษาวิจัยเพื่อให้ได้ระบบการเฝ้าตรวจสถานะของโครงข่ายที่มีประสิทธิภาพ เช่น การศึกษาเกี่ยวกับประสิทธิภาพของแบบจำลองที่ใช้ในการบริหารจัดการโครงข่าย (Vishnu, Mamidala, Hyun-Wook, and Panda, 2005)

ระบบบริหารจัดการโครงข่ายโดยทั่วไปแล้วจะทำงานบนพื้นฐานของโพรโทคอล SNMP (Simple Network Management Protocol หรือ SNMP) (Zhu, Chen, and Liu, 2001) และระบบบริหารจัดการอุปกรณ์ขัดข้อง จะอาศัยการเฝ้าตรวจสถานะของโครงข่าย (network monitoring) เพื่อทำหน้าที่ตรวจสอบสถานะ ของอุปกรณ์ในโครงข่ายว่าอยู่ในสถานะที่สามารถให้บริการเป็นปกติหรือไม่รวมทั้งการตรวจสอบและค้นหาตำแหน่งของอุปกรณ์ที่ขัดข้อง (fault localization) ซึ่งใช้โพรโทคอล SNMP เช่นกัน (Zhu, Chen, and Liu, 2001) (Su, 2002) โดยมีการทำงานที่สำคัญคือการส่งสัญญาณร้องขอไปที่อุปกรณ์ (get request) และรับสัญญาณที่ตอบกลับมาจากอุปกรณ์ (get response) ซึ่งเรียกกระบวนการทำงานในลักษณะนี้ว่าการโพลล์ (polling) หรือเป็นการส่งข้อมูลเพื่อสอบถามเข้าไปยังอุปกรณ์ที่อยู่ในระบบโครงข่าย เพื่อที่จะตรวจสอบสถานะการทำงานของอุปกรณ์โดยพิจารณาจากการส่งข้อมูลตอบกลับมา และกิจกรรมดังกล่าวต้องกระทำอยู่อย่างต่อเนื่อง ดังนั้นข้อมูลในส่วนนี้จึงก่อให้เกิดโพลล์ลิ่งโอเวอร์เฮด (polling overhead) หรือข้อมูลส่วนเกินที่เกิดจากการโพลล์ (polling) มีผลให้แบนด์วิดท์ (bandwidth) ส่วนหนึ่งต้องถูกใช้ไปกับโอเวอร์เฮดในส่วนนี้ (Zhu, Chen, and Liu, 2001)

ในกระบวนการของการเฝ้าตรวจสถานะของโครงข่าย ข้อมูลที่ได้รับกลับมาอาจเป็นข้อมูลที่ไม่ชัดเจนหรือมีข้อมูลของโครงข่ายเพียงบางส่วน (partial observability) (Steinder and Sethi, 2004) เช่น ข้อมูลอาจจะสูญหาย (packet loss) หรืออาจเกิดจากความผิดพลาดของการ รับ-ส่ง ข้อมูล (error) หรือเกิดจากลักษณะของความขัดข้องซึ่งไม่สามารถที่จะตรวจสอบได้อย่างชัดเจนจึงเป็นการยากที่จะนำมาใช้ในการตัดสินใจที่ถูกต้อง

อย่างไรก็ตามการเฝ้าตรวจสถานะของโครงข่าย และการบริหารจัดการอุปกรณ์ขัดข้อง เป็นสิ่งที่มีความสำคัญต่อเสถียรภาพของระบบโครงข่าย ดังนั้นจึงมีผู้ให้ความสำคัญในเรื่องดังกล่าวโดยทำการศึกษาวิจัยเพื่อให้ได้ระบบเฝ้าตรวจสถานะของโครงข่าย ที่มีประสิทธิภาพเพิ่มมากขึ้น ในด้านต่างๆ ดังนี้

การลดปริมาณโพลล์ลิ่งโอเวอร์เฮด เพื่อทำให้ปริมาณการใช้งานแบนด์วิดท์ ในส่วนที่เป็นโอเวอร์เฮดลดลง ซึ่งเป็นการเพิ่มประสิทธิภาพของโครงข่าย เช่น งานวิจัยที่เกี่ยวกับการตรวจสอบปริมาณข้อมูลที่เกิดขึ้นบนระบบโครงข่ายด้วยวิธีการตรวจสอบประวัติการ รับ-ส่ง ข้อมูล (passive network monitoring) (Marcia, and Bruce, 2004) โดยไม่ต้องทำการส่งการโพลล์ออกไปจึงไม่เป็นการเพิ่มแบนด์วิดท์ให้กับโครงข่ายซึ่งมีข้อเสียคือผลที่ได้จะเป็นสถานะของระบบโครงข่ายที่ไม่เป็นปัจจุบัน (non-real-time) ในงานวิจัย (Yuri, Feodor, and Hassan, 2004) ได้ทำการศึกษาถึงการโพลล์ผ่านเส้นทางที่สั้นที่สุด (shortest path tree) ในงานวิจัยของ (Xiaojiang, 2004) มีการวิจัยการใช้การเฝ้าตรวจสถานะของโครงข่ายแบบกระจาย (distributed) ซึ่งใช้วิธีแบ่งเป็นโครงข่ายย่อย (subnetwork) แต่ทั้งใน (Yuri, Feodor, and Hassan, 2004) และ (Xiaojiang, 2004) จำเป็นที่จะต้องติดตั้งอุปกรณ์เฝ้าตรวจสถานะของโครงข่าย ที่ทำหน้าที่ในการโพลล์เป็นจำนวนมาก แม้จะสามารถลดปริมาณโพลล์ลิ่งโอเวอร์เฮดลงได้ในปริมาณหนึ่งแต่ยังคงถือได้ว่าปริมาณโพลล์ลิ่งโอเวอร์เฮดยังคงอยู่ในระดับที่สูง และใน (Yoshihara, Sugiyama, Horiuchi, and Obana, 1999) ได้นำเสนอการใช้วิธีการกำหนดช่วงเวลาของการโพลล์ ในแต่ละรอบให้เปลี่ยนแปลงได้ โดยทำการตรวจสอบปริมาณแบนวิดท์ที่ใช้ในการเฝ้าตรวจสถานะของโครงข่ายให้ไม่เกิน 5 % ของแบนวิดท์ทั้งหมด หากพบว่ามีการใช้งานแบนวิดท์ที่เกิดจากการเฝ้าตรวจสถานะของโครงข่ายเกินกว่าที่กำหนด จะทำการเพิ่มระยะเวลาที่จะทำการโพลล์ครั้งถัดไปให้ห่างออกไป ซึ่งผลเสียคือจะทำให้ตรวจพบปัญหาที่เกิดขึ้นได้ช้าลงด้วย

การค้นหาตำแหน่งที่อุปกรณ์ขัดข้อง ซึ่งเป็นส่วนที่มีความสำคัญมากในการบริหารจัดการโครงข่าย ได้มีผู้วิจัยและเสนอแนวคิดต่างๆ เช่น การสร้างมอดูลที่ใช้ในการโพลล์สำหรับโครงข่ายขนาดใหญ่ ที่มีอุปกรณ์เชื่อมต่ออยู่จำนวนมากซึ่งมีเหตุการณ์ (event) ที่เกิดขึ้นจำนวนมากด้วย จึงเลือกใช้วิธีการที่ทำการวิเคราะห์และเลือกสนใจเฉพาะในส่วนของอุปกรณ์ขัดข้อง แต่มิได้นำเสนอ

ในส่วนของการค้นหาตำแหน่งของอุปกรณ์ที่ขัดข้อง (Sreedhar, Hill, and Stanley, 2000) การใช้การบริหารจัดการโครงข่ายแบบกระจายโดยทำการแยกการบริหารจัดการเป็นโดเมน (domain) เพื่อต้องการที่จะค้นหาตำแหน่งที่อุปกรณ์ขัดข้องที่แท้จริงได้อย่างรวดเร็ว ซึ่งพบว่าวิธีการแบบกระจายนี้จะให้ผลที่ดีกว่าแบบรวมศูนย์ (centralized) แต่มิได้นำเสนอถึงวิธีการที่ใช้ในการค้นหาตำแหน่งที่อุปกรณ์ขัดข้องที่เกิดขึ้นในแต่ละครั้ง (Bouloutas, Calo, and Katzela, 1995) และการรวบรวมงานวิจัยที่เกี่ยวข้องกับการค้นหาตำแหน่งที่อุปกรณ์ขัดข้องโดย (Steinder, and Sethi, 2004) ซึ่งพบว่ามีงานวิจัยจำนวนมากที่ทำการวิจัยในเรื่องนี้

การทำงานภายใต้สภาวะกรณีที่สิ่งที่ได้จากการสังเกตไม่ชัดเจน (partially observable) ซึ่งโดยปกติแล้วข้อมูลที่ได้จากการตรวจสอบสถานะของโครงข่ายจะเป็นข้อมูลที่ไม่ชัดเจน (Steinder, and Sethi, 2004) ซึ่งการที่มีข้อมูลที่ไม่ชัดเจนนี้เป็นปัญหาหนึ่งและส่งผลต่อระบบโครงข่ายหลายๆ อย่าง เช่น การที่ไม่ทราบตำแหน่งที่ชัดเจนของโหนด (node) จะส่งผลกระทบในการหาเส้นทาง (routing) และในบางครั้งอาจจำเป็นที่จะต้องตัดโหนดนี้ทิ้งไปเพื่อพยายามรักษาคุณภาพของการให้บริการไว้ รวมทั้งมีการเสนอวิธีแก้ไขปัญหาโดยการสร้างกระบวนการหาเส้นทางในกรณีที่มีข้อมูลไม่ชัดเจน (Goldenberg, et al., 2005) หรือการพยายามแก้ปัญหาที่เกิดขึ้นจากการเฝ้าตรวจสถานะของโครงข่ายโดยการสร้างลำดับขั้นของการโพลล์เพื่อทำการยืนยันสถานะที่แท้จริง (He, 2003)

เมื่อโครงข่ายมีขนาดใหญ่ขึ้น มีความซับซ้อนมากขึ้น มีอุปกรณ์ที่เชื่อมต่ออยู่ในโครงข่ายมากขึ้นอีกทั้งยังมีการเชื่อมต่อโครงข่ายที่มีลักษณะแตกต่างกันเข้าด้วยกันทั้งในแง่ของระบบบริหารจัดการอุปกรณ์หรือโครงรูปของโครงข่าย (network topology) ดังนั้นระบบบริหารจัดการโครงข่ายที่ดีต้องสามารถทำการวิเคราะห์เหตุการณ์ความผิดปกติที่เกิดขึ้นได้อย่างถูกต้อง และรวดเร็วจึงจำเป็นต้องอาศัยระบบบริหารจัดการโครงข่ายที่มีประสิทธิภาพ เพื่อให้โครงข่ายสามารถตอบสนองความต้องการการใช้งานดังกล่าวได้อย่างต่อเนื่อง (Sreedhar, Hill, and Stanley, 2000)

ดังนั้นในงานวิจัยนี้ได้นำเสนอวิธีรีอินฟอร์สเมนท์เลิร์นนิ่ง (Reinforcement Learning หรือ RL) เข้ามาช่วยในการลดปริมาณโพลล์ลิ่งโอเวอร์เฮดที่เกิดจากการเฝ้าตรวจสถานะของโครงข่ายให้มีปริมาณที่ลดลง ซึ่งการใช้วิธีรีอินฟอร์สเมนท์เลิร์นนิ่งนี้มีผู้นำมาใช้ในการแก้ปัญหาต่างๆ เช่น ควบคุมการดร็อปในการแฮนด์โอเวอร์ของการใช้งานมัลติมีเดีย ในโครงข่ายสื่อสารแบบไร้สาย (Alexandri, Martinez, and Zeghlache, 2002) ใช้ในการจัดสรรทรัพยากรในระบบโครงข่ายดาวเทียมวงโคจรต่ำ (Usaha, 2004) ในด้านของการบริหารจัดการโครงข่ายได้มีการนำมาใช้กับการบริหารจัดการโครงข่ายแบบโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ (proactive network management) (He, 2003) ซึ่งเป็นวิธีที่นำเสนออยู่ในปัจจุบัน ด้วยวิธีการสร้างลำดับขั้นในการโพลล์ ในการพิจารณาหาตำแหน่งที่อุปกรณ์ขัดข้องโดยการสร้างขั้นตอนวิธี (algorithm) เข้ามาช่วยในการกำหนดลำดับการโพลล์ซึ่งมีผลทำให้จำนวนการโพลล์ลดลงได้และถือได้ว่าสามารถลดโพลล์ลิ่งโอเวอร์เฮด

ลงได้ในระดับหนึ่ง แต่ด้วยการทำงานของขั้นตอนวิธีที่มีการทำงานที่ซับซ้อนต้องใช้การคำนวณหลายขั้นตอนโดยในวิทยานิพนธ์นี้ได้นำเสนอวิธีการในการลดปริมาณโพลล์ลิ่งโอเวอร์เฮดที่เกิดจากการเฝ้าตรวจสถานะของโครงข่ายให้มีปริมาณที่ลดลง ด้วยวิธีรีอินฟอร์สเมนท์เลิร์นนิ่งแบบออนโพลิซีมอนติคาร์โล (On-policy Monte Carlo หรือ ONMC) โดยทำการศึกษากับกรณีโครงข่ายขนาดเล็กและกรณีที่โครงข่ายมีขนาดใหญ่

1.2 วัตถุประสงค์ของการวิจัย

1.2.1 เพื่อศึกษาการใช้การบริหารจัดการอุปกรณ์ขัดข้องที่สามารถนำมาใช้ เพื่อลดปริมาณโพลล์ลิ่งโอเวอร์เฮดที่เกิดจากการเฝ้าตรวจสถานะของโครงข่าย

1.2.2 เพื่อศึกษาถึงวิธีการที่จะนำขั้นตอนวิธี สำหรับใช้ในการค้นหาอุปกรณ์ที่ขัดข้องที่มีความถูกต้อง ภายใต้สภาวะที่ข้อมูลที่พิจารณาไม่ชัดเจนหรือสิ่งที่ได้จากการสังเกตไม่ชัดเจน โดยพิจารณาจากการเลือกการกระทำที่เกิดขึ้นและการเลือกการกระทำได้ถูกต้อง

1.2.3 เพื่อศึกษาขั้นตอนวิธีการส่งข้อมูลการโพลล์ ที่มีความซับซ้อนของการคำนวณต่ำและใช้หน่วยความจำน้อย

1.3 สมมุติฐานของการวิจัย

1.3.1 การเฝ้าตรวจสถานะของโครงข่ายที่ดีควรมีโพลล์ลิ่งโอเวอร์เฮดต่ำ

1.3.2 การบริหารจัดการอุปกรณ์ขัดข้องที่ดีต้องสามารถที่จะตรวจพบความผิดปกติที่เกิดขึ้นได้ครบทุกเหตุการณ์

1.3.3 วิธีรีอินฟอร์สเมนท์เลิร์นนิ่ง สามารถที่จะนำมาใช้ในการแก้ปัญหาโพลล์ลิ่งโอเวอร์เฮดของการบริหารจัดการโครงข่าย ภายใต้สภาวะที่มีข้อมูลของกการสังเกตจากโครงข่ายเพียงบางส่วนได้

1.3.4 ระบบโครงข่ายมีคุณสมบัติแบบมาร์คอฟ (Markovian stochastic environment) ซึ่งทำให้การเปลี่ยนสถานะของโครงข่ายไปยังสถานะถัดไปจะขึ้นอยู่กับสถานะในปัจจุบันเท่านั้น และวิธีรีอินฟอร์สเมนท์เลิร์นนิ่ง สามารถทำงานภายใต้คุณสมบัติแบบมาร์คอฟนี้ได้ (รายละเอียดในบทที่ 2)

1.4 ขอบเขตของการวิจัย

1.4.1 ดำเนินการศึกษาเทคนิคการลดปริมาณโพลล์ลิ่งโอเวอร์เฮด ภายใต้สภาวะที่ข้อมูลที่พิจารณาไม่ชัดเจนหรือสิ่งที่ได้จากการสังเกตไม่ชัดเจนของการเฝ้าตรวจสถานะของโครงข่าย ในระบบโครงข่ายขนาดเล็กและระบบโครงข่ายขนาดใหญ่

1.4.2 ดำเนินการศึกษาเทคนิคการตรวจสอบและค้นหาตำแหน่งที่อุปกรณ์ขัดข้อง ภายใต้สภาวะที่ข้อมูลที่พิจารณาไม่ชัดเจนหรือสิ่งที่ได้จากการสังเกตไม่ชัดเจนของการเฝ้าตรวจสถานะของโครงข่ายในระบบโครงข่ายขนาดเล็กและระบบโครงข่ายขนาดใหญ่

1.4.3 ดำเนินการทดลองเพื่อเปรียบเทียบปริมาณโพลล์ลิ่งโอเวอร์เฮด ด้วยการใช้วิธีรีอินฟอร์สเมนท์เลิร์นนิ่ง เปรียบเทียบกับวิธีที่มีอยู่เดิม เช่น โปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ (He, 2003) และวิธีการโพลล์ที่คำนึงถึงโครงรูปโครงข่าย (Han, Ahn, and Chung, 2001)

1.4.4 ดำเนินการทดลองเพื่อเปรียบเทียบจำนวนขั้นตอนที่ใช้ในการตรวจสอบ และค้นหาตำแหน่งที่อุปกรณ์ขัดข้อง ด้วยการใช้วิธีรีอินฟอร์สเมนท์เลิร์นนิ่งเปรียบเทียบกับวิธีที่มีอยู่เดิม เช่น โปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ (He, 2003) และวิธีการโพลล์ที่คำนึงถึงโครงรูปโครงข่าย (Han, Ahn, and Chung, 2001)

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1.5.1 ได้โปรแกรมจำลองระบบการเฝ้าตรวจสถานะของโครงข่ายและการค้นหาตำแหน่งที่อุปกรณ์ขัดข้อง

1.5.2 ได้ข้อสรุปอันเป็นประโยชน์เกี่ยวกับกระบวนการในการตัดสินใจที่สามารถลดปริมาณโพลล์ลิ่งโอเวอร์เฮด และวิธีการสำหรับการตรวจสอบและค้นหาตำแหน่งที่อุปกรณ์ขัดข้องของระบบบริหารจัดการโครงข่ายด้วยวิธีรีอินฟอร์สเมนท์เลิร์นนิ่ง

1.5.3 ได้โปรแกรมจำลองระบบเพื่อนำไปใช้ในการพัฒนาระบบบริหารจัดการโครงข่าย

1.6 การจัดรูปเล่มวิทยานิพนธ์

วิทยานิพนธ์ฉบับนี้ประกอบด้วย 5 บท และ 2 ภาคผนวก บทที่ 1 เป็นบทนำกล่าวถึงความสำคัญของปัญหาและสาเหตุแห่งปัญหา รวมทั้งงานวิจัยที่ได้มีผู้นำเสนอไว้แล้วเพื่อเป็นแนวทางและข้อพิจารณาในการศึกษาวิจัย วัตถุประสงค์ ข้อตกลงเบื้องต้น ขอบเขตของการวิจัย และประโยชน์ที่คาดว่าจะได้รับจากงานวิจัย รวมทั้งแนะนำเนื้อหาเบื้องต้นของวิทยานิพนธ์ฉบับนี้ ส่วนบทอื่นๆ ประกอบด้วยเนื้อหาดังนี้

บทที่ 2 กล่าวถึงทฤษฎีและหลักการเบื้องต้นของคุณสมบัติแบบมาร์คอฟ ทฤษฎีวิธีรีอินฟอร์สเมนท์เลิร์นนิ่ง โดยมีคุณสมบัติและองค์ประกอบการนำมาใช้ในการแก้ปัญหากรณีที่สิ่งที่ได้จากการสังเกตไม่ชัดเจน และขั้นตอนวิธีที่ได้นำเสนอในวิทยานิพนธ์นี้ ซึ่งใช้วิธีรีอินฟอร์สเมนท์เลิร์นนิ่งแบบออนโพลิซีมอนติคาร์โล

บทที่ 3 เป็นการทดสอบสมมุติฐานในการนำวิธีออนโพลิซีมอนติคาร์โลมาใช้ลดโพลล์ลิ่งโอเวอร์เฮด โดยทำการศึกษาในโครงข่ายขนาดเล็ก ที่ประกอบด้วยอุปกรณ์โครงข่ายจำนวนน้อย

และมีการเชื่อมต่อที่ไม่ซับซ้อน ซึ่งจะกล่าวถึงการดำเนินการวิจัยในการลดโพลล์ลิ่งโอเวอร์เฮด และการหาตำแหน่งที่อุปกรณ์ขัดข้องที่ทำการศึกษาในโครงข่ายขนาดเล็ก ด้วยการใช้วิธีรีอินฟอร์สเมนท์เลิร์นนิ่งแบบออนโพลิซีมอนติคาร์โล เปรียบเทียบกับวิธีการแบบโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ (He, 2003) โดยจะแสดงผลของการดำเนินการวิจัย เช่น จำนวนการกระทำที่เกิดขึ้น เมื่อความน่าจะเป็นของการเกิดสัญญาณเตือนที่ผิดพลาด (false alarm) เพิ่มสูงขึ้น หรือสิ่งที่ได้จากการสังเกตมีความไม่ชัดเจนเพิ่มสูงขึ้นและข้อสังเกตต่างๆ ในเชิงของการเปรียบเทียบ

บทที่ 4 กล่าวถึงการดำเนินการวิจัยในการลดโพลล์ลิ่งโอเวอร์เฮด และการหาตำแหน่งที่อุปกรณ์ขัดข้องในโครงข่ายขนาดใหญ่ที่ประกอบด้วยจำนวนอุปกรณ์ของโครงข่ายจำนวนมาก และการเชื่อมต่อที่มีความซับซ้อน ซึ่งส่งผลให้มีจำนวนสถานะของโครงข่ายเพิ่มมากขึ้น และมีโอกาสในการเกิดกรณีที่สิ่งที่ได้จากการสังเกตไม่ชัดเจนสูงขึ้นด้วย โดยจะแสดงผลของการดำเนินการวิจัยและข้อสังเกตต่างๆ เปรียบเทียบกับวิธีการแบบโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ (He, 2003) และวิธีการโพลล์ที่คำนึงถึงโครงรูปโครงข่าย (Han, Ahn, and Chung, 2001)

บทที่ 5 เป็นส่วนของบทสรุปของการกำเนินการวิจัย ปัญหาและข้อเสนอแนะที่เกิดขึ้นในการดำเนินการวิจัยและงานวิจัยในอนาคต

ภาคผนวก ก ค่าความแปรปรวนในการจำลองแบบ

ภาคผนวก ข บทความวิชาการที่ได้รับการตีพิมพ์เผยแพร่ในขณะศึกษา

บทที่ 2
ทฤษฎีวิธีรีอินฟอร์สเมนท์เลิร์นนิ่ง

2.1 กล่าวนำ

ในการเฝ้าตรวจสถานะของโครงข่าย โดยใช้วิธีการโพลล์ไปยังอุปกรณ์ภายในโครงข่ายนั้น เพื่อให้ได้สมรรถนะของโครงข่ายที่ดี จึงต้องการให้มีจำนวนโพลล์ลิ่งโอเวอร์เฮดที่ต่ำ ไม่ว่าจะในโครงข่ายที่มีขนาดเล็กหรือโครงข่ายที่มีขนาดใหญ่

โดยการเกิดข้อขัดข้องของโครงข่ายจะเป็นการเปลี่ยนสถานะ จากสถานะปัจจุบันไปยังสถานะถัดไปและไม่ขึ้นกับสถานะในอดีตที่ผ่านมา เช่น กรณีที่สถานะของโครงข่ายที่อุปกรณ์ทุกอุปกรณ์ทำงานเป็นปกติเมื่อเวลาผ่านไปปรากฏว่ามีอุปกรณ์ขัดข้องเกิดขึ้น จะเห็นว่าในการเปลี่ยนสถานะนี้จะเป็นการเปลี่ยนสถานะจากการที่อุปกรณ์ทุกตัวทำงานเป็นปกติ มาเป็นสถานะที่มีอุปกรณ์ขัดข้องเกิดขึ้นโดยไม่มีผลที่มาจากสถานะในอดีตที่ผ่านมาก่อนหน้านี้ ดังนั้นพฤติกรรมของระบบโครงข่ายจึงมีคุณสมบัติของการเป็นมาร์คอฟ (Markov property) (He, 2003)

ในการเฝ้าตรวจสถานะของโครงข่าย เมื่อพบว่ามีอุปกรณ์ขัดข้องเกิดขึ้นจะเลือกตัดสินใจที่จะเลือกการกระทำใดการกระทำหนึ่ง เช่น ทำการซ่อมอุปกรณ์หรือทำการพิสูจน์ข้อขัดข้องที่เกิดขึ้นและจะพบว่า ผลของการกระทำนั้นจะส่งผลมาจาก การที่อยู่ในสถานะปัจจุบันและเลือกการกระทำซึ่งคุณสมบัติเช่นนี้จะถือได้ว่ามีคุณสมบัติของ กระบวนการตัดสินใจแบบมาร์คอฟ (Markov decision process) (He, 2003)

นอกจากนั้นการทำงานของการเฝ้าตรวจสถานะของโครงข่ายยังทำงานภายใต้สภาวะที่สิ่งที่ได้จากการสังเกตไม่ชัดเจน (partially observable) ซึ่งเกิดจากข้อมูลอาจจะสูญหาย (packet loss) หรืออาจเกิดจากความผิดพลาดของการ รับ-ส่ง ข้อมูล (error) หรือกรณีระบบโครงข่ายมีขนาดใหญ่ที่มีเหตุการณ์ (event) เกิดขึ้นเป็นจำนวนมากและมักประกอบด้วยสัญญาณเตือนที่ถูกต้อง (false) และสัญญาณเตือนที่ผิดพลาด (false alarm) ด้วยสาเหตุดังกล่าวจึงส่งผลต่อการตัดสินใจในการกระทำของการเฝ้าตรวจสถานะของโครงข่าย เนื่องจากการไม่ทราบสถานะที่แท้จริงของโครงข่ายจึงไม่สามารถที่จะเลือกการกระทำที่ดีที่สุดสำหรับสถานะนั้นได้ (He, 2003) (Steinder, and Sethi, 2004)

จึงถือได้ว่าการทำงานของการเฝ้าตรวจสถานะของโครงข่าย เป็นการทำงานภายใต้สภาวะที่สิ่งแวดล้อมมีคุณสมบัติของมาร์คอฟและมีกระบวนการตัดสินใจแบบมาร์คอฟ ทั้งยังเป็นการทำงานในกรณีที่สิ่งที่ได้จากการสังเกตจากสิ่งแวดล้อมมีความไม่ชัดเจน ซึ่งเรียกว่ากระบวนการตัดสินใจ

แบบมาร์คอฟกรณีที่สิ่งที่ได้จากการสังเกตไม่ชัดเจน (Partially Observable Markov Decision Process หรือ POMDP) (He, 2003) (Usaha, 2004)

โดยงานวิจัยนี้ได้เลือกใช้วิธีรีอินฟอร์สเมนท์เลิร์นนิ่ง (Reinforcement Learning หรือ RL) เข้ามาช่วยในการลดโพลล์ลิ่งโอเวอร์เฮดลง ซึ่งวิธีรีอินฟอร์สเมนท์เลิร์นนิ่ง นี้มีผู้นำมาใช้ในการแก้ปัญหาต่างๆ เช่น ควบคุมการดร็อปในการแฮนด์โอเวอร์ของการใช้งานมัลติมีเดียในโครงข่ายสื่อสารแบบไร้สาย (Alexandri, Martinez, and Zeghlache, 2002) ใช้ในการจัดสรรทรัพยากรในระบบโครงข่ายดาวเทียมวงโคจรต่ำ (Usaha, 2004)

ในการนำเอาวิธีรีอินฟอร์สเมนท์เลิร์นนิ่งมาประยุกต์ใช้ ในการเฝ้าตรวจสถานะของโครงข่าย นี้ได้นำเสนอวิธีรีอินฟอร์สเมนท์เลิร์นนิ่งแบบออนโพลิซีมอนติคาร์โล (On-policy Monte Carlo หรือ ONMC) ทำการศึกษากับโครงข่ายขนาดเล็กและโครงข่ายขนาดใหญ่ โดยเนื้อหาในบทนี้เป็นการแนะนำทฤษฎีพื้นฐานของวิธีและเทคนิคต่างๆ ที่กล่าวไว้ข้างต้นโดยประกอบไปด้วย หัวข้อ 2.2 กล่าวถึงคุณสมบัติแบบมาร์คอฟซึ่งเป็นคุณสมบัติที่ใช้พิจารณาการเปลี่ยนสถานะของสิ่งแวดล้อม เช่น สถานะของระบบโครงข่าย หัวข้อ 2.3 ทฤษฎีวิธีรีอินฟอร์สเมนท์เลิร์นนิ่ง โดยมีรายละเอียดและองค์ประกอบและการนำมาใช้ในการแก้ปัญหา หัวข้อ 2.4 จะกล่าวถึงกระบวนการตัดสินใจแบบมาร์คอฟกรณีที่สิ่งที่ได้จากการสังเกตไม่ชัดเจน สำหรับการทำงานในกรณีที่สิ่งที่ได้จากการสังเกตจากสิ่งแวดล้อมมีความไม่ชัดเจน หัวข้อ 2.5 จะกล่าวถึงหลักการของวิธีออนโพลิซีมอนติคาร์โล หัวข้อ 2.6 จะกล่าวถึงหลักการของวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ และหัวข้อ 2.7 เป็นส่วนของบทสรุป

2.2 คุณสมบัติแบบมาร์คอฟและกระบวนการตัดสินใจแบบมาร์คอฟ

การเกิดข้อขัดข้องของโครงข่ายซึ่งส่งผลให้สถานะของโครงข่ายเปลี่ยนแปลงไป เช่นกรณีที่สถานะของโครงข่ายที่อุปกรณ์ทุกอุปกรณ์ทำงานเป็นปกติเมื่อเวลาผ่านไปมีอุปกรณ์ขัดข้องเกิดขึ้น จะเห็นว่าในการเปลี่ยนสถานะนี้ จะเป็นการเปลี่ยนสถานะจากการที่อุปกรณ์ทุกตัวทำงานเป็นปกติมาเป็นสถานะที่มีอุปกรณ์ขัดข้องเกิดขึ้น หรือในทำนองเดียวกันกรณีที่อุปกรณ์อยู่ในสถานะขัดข้องและได้รับการแก้ไขให้กลับมาอยู่ในสถานะปกติ จะเป็นการเปลี่ยนสถานะที่มาจากสถานะก่อนหน้านี้เพียงหนึ่งสถานะเท่านั้นโดยไม่มีผลที่มาจากสถานะในอดีตที่ผ่านมา หรือเป็นการเปลี่ยนสถานะจากสถานะปัจจุบันไปยังสถานะถัดไปโดยไม่ขึ้นกับสถานะในอดีตที่ผ่านมา ดังนั้นพฤติกรรมของการเปลี่ยนสถานะของระบบโครงข่าย จึงจัดได้ว่ามีคุณสมบัติของการเป็นมาร์คอฟ (Markov property) (He, 2003)

2.2.1 คุณสมบัติแบบมาร์คอฟ

ถ้าสิ่งแวดล้อมที่พิจารณามีลักษณะของการเปลี่ยนสถานะ โดยที่สถานะในอนาคตไม่ขึ้นกับสถานะในอดีตที่ผ่านมาแต่จะขึ้นอยู่กับสถานะปัจจุบันเพียงสถานะเดียวเท่านั้น โดยจะเรียกสิ่ง

แวดล้อมดังกล่าวว่ามีคุณสมบัติแบบมาร์คอฟ (Markov Property หรือ MP) ซึ่งทำให้การวิเคราะห์การเปลี่ยนแปลงสถานะของสิ่งแวดล้อมทำได้ง่ายขึ้นเนื่องจากไม่จำเป็นต้องรู้ถึงการเปลี่ยนแปลงในอดีตทั้งหมดที่ผ่านมาเพราะสถานะถัดไปจะขึ้นอยู่กับสถานะในปัจจุบันเท่านั้น

2.2.2 กระบวนการตัดสินใจแบบมาร์คอฟ

เมื่อพิจารณาสิ่งแวดล้อมที่มีคุณสมบัติแบบมาร์คอฟ ซึ่งหากสิ่งแวดล้อมนั้นมีการเปลี่ยนสถานะจากสถานะปัจจุบันและมีการเลือกการกระทำใดการกระทำหนึ่ง ส่งผลให้สิ่งแวดล้อมนั้นเปลี่ยนไปยังสถานะถัดไปโดยไม่ขึ้นกับสถานะอื่นๆ ก่อนหน้านี้ จะเรียกสิ่งแวดล้อมที่มีการเลือกการกระทำเช่นนี้ว่ามีคุณสมบัติของกระบวนการตัดสินใจแบบมาร์คอฟ (Markov Decision Process หรือ MDP) (Sutton, and Barto, 1998) และหากสิ่งแวดล้อมนั้นมีจำนวนสถานะที่มีขอบเขต (a finite set of states) และมีจำนวนของการกระทำแบบมีขอบเขต (a finite set of actions) เราจะเรียกกระบวนการตัดสินใจเช่นนี้ว่า กระบวนการตัดสินใจแบบมาร์คอฟที่มีขอบเขตจำกัด (finite Markov Decision Process หรือ finite MDP) โดยในงานวิจัยนี้จะใช้กระบวนการตัดสินใจแบบมาร์คอฟที่มีขอบเขตจำกัดนี้เป็นหลัก เนื่องจากในระบบโครงข่ายนั้นจะประกอบด้วยอุปกรณ์ในจำนวนที่นับได้จึงถือว่ามีขอบเขตที่จำกัด และจำนวนของการกระทำที่เป็นไปได้จะมีจำนวนที่นับได้ซึ่งถือว่ามีขอบเขตที่จำกัดเช่นกัน

ดังนั้นถ้าให้ s_t แทนสถานะในเวลา t และ r_t เป็นผลรางวัลที่ได้รับ เมื่อสิ่งแวดล้อมอยู่ในสถานะ s_t และเลือกการกระทำ a_t จะได้ประวัติของกระบวนการดังสมการ (2.1) (Sutton, and Barto, 1998)

$$\Pr\left\{s_{t+1}=s^{'},r_{t+1}=r \mid s_t,a_t,r_t,s_{t-1},a_{t-1},...,r_1,s_0a_0\right\} \tag{2.1}$$

เมื่อ s' เป็นสถานะถัดไปและ r เป็นผลรางวัลที่ได้จากการกระทำ a ดังนั้นจะได้ค่าที่ผ่านมาในอดีตเป็น $s_t,a_t,r_t,s_{t-1},a_{t-1},...,r_1,s_0,a_0$ ซึ่งหากสิ่งแวดล้อมนี้มีคุณสมบัติแบบมาร์คอฟ จะสามารถเขียนได้ดังสมการที่ (2.2) (Sutton, and Barto, 1998) ซึ่งสรุปได้ว่าการเปลี่ยนไปยังสถานะถัดไปของสิ่งแวดล้อมจะขึ้นอยู่เพียง สถานะในปัจจุบันและการกระทำที่กระทำเมื่ออยู่ในสถานะปัจจุบันเท่านั้น

$$\Pr\left\{s_{t+1}=s^{'},r_{t+1}=r \mid s_t,a_t\right\} \tag{2.2}$$

เมื่อ s' เป็นสถานะถัดไป r เป็นผลรางวัลที่ได้รับเมื่อสิ่งแวดล้อมอยู่ในสถานะ s_t และเลือกการกระทำ a_t ดังนั้นหากสมการ (2.1) เท่ากับสมการ (2.2) จะถือได้ว่าสิ่งแวดล้อมนี้มีคุณสมบัติแบบมาร์คอฟ และคุณสมบัติแบบมาร์คอฟนี้จะเป็นคุณสมบัติที่สำคัญมากของวิธีรีอินฟอร์สเมนท์เลิร์นนิ่ง

ที่ช่วยให้สามารถที่จะทราบสถานะถัดไปได้หากทราบสถานะและการกระทำในปัจจุบัน (Sutton, and Barto, 1998)

2.3 ทฤษฎีวิธีรีอินฟอร์สเมนท์เลิร์นนิ่ง

วิธีรีอินฟอร์สเมนท์เลิร์นนิ่ง (Reinforcement Learning หรือ RL) จะเป็นการเรียนรู้จากจุดมุ่งหมายโดยตรงแล้วจึงตัดสินใจในการทำตามจุดมุ่งหมายนั้นโดยโครงสร้างพื้นฐานในการเรียนรู้ด้วยการตัดสินใจจากสภาวะแวดล้อมในลักษณะของ สถานะ (state) การกระทำ (action) และผลรางวัล (reward) เพื่อที่จะทำการเรียนรู้ในการเชื่อมโยงสถานะปัจจุบัน ไปสู่การเลือกการกระทำในลำดับถัดไป หรือเป็นการเรียนรู้เพื่อที่จะทำการตัดสินใจว่า หากสภาวะแวดล้อมอยู่ในสถานะหนึ่งแล้วควรจะเลือกการกระทำใดต่อไปที่จะก่อให้เกิดผลของการกระทำที่ดี โดยมีวงรอบการทำงานดังรูปที่ 2.1 ซึ่งเริ่มจากเมื่อตัวกระทำการตัดสินใจของระบบ (agent) ได้รับสถานะและผลรางวัลที่เกิดขึ้น ตัวกระทำการตัดสินใจจะทำการตัดสินใจที่จะเลือกกระทำการอย่างใดอย่างหนึ่ง จากนั้นตัวกระทำการตัดสินใจจะรับทราบถึงการเปลี่ยนไปของสถานะและผลรางวัลที่ได้รับกลับมา ระบบจะดำเนินไปและได้รับผลรางวัลกลับมาอยู่โดยตลอด จนกระทั่งตัวกระทำการตัดสินใจเกิดการเรียนรู้ว่าเมื่ออยู่ในสถานะใดจะต้องเลือกการกระทำใด จึงจะได้ผลรางวัลระยะยาวสูงสุด (long-term expect reward)

Reward r_t คือ ผลรางวัลในระยะยาวที่ได้จากการเลือกการกระทำ ณ เวลา t

State s_t คือ สถานะของระบบ ณ เวลา t

Action a_t คือ การกระทำที่ตัวกระทำการตัดสินใจเลือกกระทำ ณ เวลา t

Reward r_{t+1} คือ ผลรางวัลที่ได้จากการกระทำในเวลา t+1

State s_{t+1} คือ สถานะของสิ่งแวดล้อมในเวลา t+1

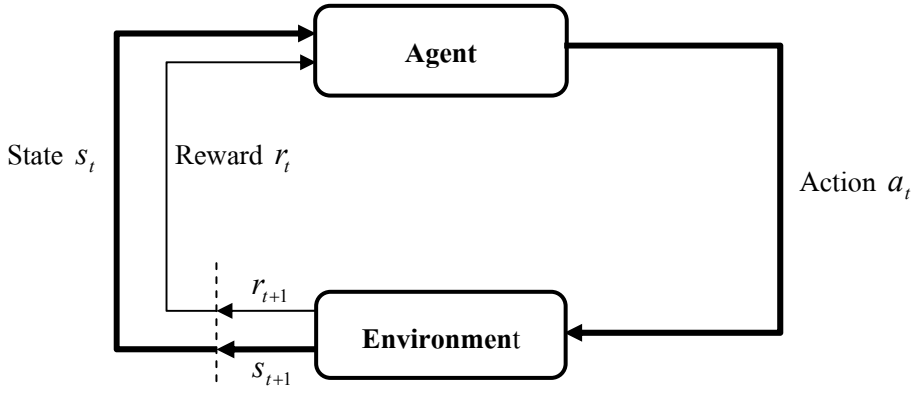
เมื่อ s เป็นสถานะของสิ่งแวดล้อม S เป็นเซตสถานะทั้งหมดที่เป็นไปได้ โดย $s_t \in S$ และ $A(s_t)$ เป็นเซตของการกระทำทั้งหมดที่เป็นไปได้ในแต่ละสถานะ โดย $a_t \in A(s_t)$ และ $r_{t+1} \in \Re$ เมื่อ t คือเวลาในแต่ละขั้นโดย $t = 0,1,2,3,....$

2.3.1 องค์ประกอบของวิธีรีอินฟอร์สเมนท์เลิร์นนิ่ง

โดยทั่วไปแล้ววิธีรีอินฟอร์สเมนท์เลิร์นนิ่งจะประกอบด้วย 4 องค์ประกอบหลัก (Sutton, and Barto, 1998) คือ

2.3.1.1 กฎควบคุม (policy) คือตัวกำหนดแนวทางของการเรียนรู้และการตัดสินใจในการเลือกการกระทำในลำดับถัดไป เพื่อให้ได้ผลรางวัลเฉลี่ยในระยะยาวที่สูงสุด ถ้าให้กฎควบคุมแทนด้วย π ที่ได้มาจากการหาค่าสูงสุดของผลเฉลี่ยรางวัลสะสมที่เรียกว่าคิวแวลู่แทนด้วย $Q(s,a)$ เพื่อนำไปใช้เป็นฟังก์ชันในการเลือกการกระทำดังสมการ (2.3) (Sutton, and Barto, 1998)

$$\pi(s) = \arg\max_{a} Q(s,a) \tag{2.3}$$



รูปที่ 2.1 วงรอบการทำงานของวิธีรีอินฟอร์สเมนท์เลิร์นนิ่ง

2.3.1.2 ฟังก์ชันผลรางวัล (reward function) คือฟังก์ชันที่ใช้คำนวณผลรางวัลในระยะยาวของระบบที่ได้ดำเนินอยู่ ณ ขณะนั้น โดยผลรางวัลนี้จะเกิดจากการตัดสินใจในการเลือกการกระทำเมื่อระบบอยู่ในสถานะต่างๆ โดยหากเป็นการตัดสินใจที่ถูกต้องจะได้ผลรางวัลในระดับที่สูง แต่หากเป็นการตัดสินใจที่ถูกต้องน้อยกว่าหรือเป็นการตัดสินใจที่ผิดจะได้รับผลรางวัลในระดับที่ต่ำลดหลั่นกันลงมา ซึ่งผลรางวัลนี้จะเป็นตัวแสดงถึงความสามารถของตัวตัดสินใจว่าสามารถทำการตัดสินใจได้ดีหรือไม่ในช่วงเวลา ณ ขณะนั้น จุดมุ่งหมายหลักของการแก้ปัญหาด้วยวิธีรีอินฟอร์สเมนท์เลิร์นนิ่งคือ ต้องการการเรียนรู้เพื่อให้ได้กฎควบคุมในการตัดสินใจที่ส่งผลให้ได้ผลรางวัลในระยะยาวที่สูงที่สุด ซึ่งจะแสดงถึงการตัดสินใจที่ดีของวิธีรีอินฟอร์สเมนท์เลิร์นนิ่ง หากพิจารณาผลรางวัลที่เกิดขึ้นภายหลังช่วงเวลา t กำหนดเป็น

$$g(s_t, a_t) + g(s_{t+1}, a_{t+1}) + g(s_{t+2}, a_{t+2}) + \ldots$$

เมื่อผลรวมของผลรางวัลในระยะยาวแทนด้วย R_t จะได้ค่าผลรวมของผลรางวัลที่เกิดขึ้นตั้งแต่เวลา t จนถึงเวลา T ดังสมการ (2.4) (Sutton, and Barto, 1998) ซึ่งวิธีรีอินฟอร์สเมนท์เลิร์นนิ่ง มีเป้าหมายต้องการที่จะให้ได้ R_t ที่สูงที่สุด

$$R_t = g(s_t, a_t) + g(s_{t+1}, a_{t+1}) + g(s_{t+2}, a_{t+2}) + \ldots + g(s_{T-1}, a_{T-1}) \tag{2.4}$$

เมื่อ R_t เป็นผลรวมของผลรางวัลของช่วงเวลาตั้งแต่เวลา t จนถึงเวลา T

2.3.1.3 แวลู่ฟังก์ชัน (value function) คือฟังก์ชันที่ใช้คำนวณหาผลรางวัลในระยะยาวที่คาดว่าจะได้รับหากเลือกการกระทำนั้นๆ ภายใต้กฎควบคุมเดียวกัน ในการจำลองการกระทำซึ่งตัวกระทำการตัดสินใจ จะใช้ผลรางวัลนี้ในการตัดสินใจเลือกการกระทำในลำดับถัดไป โดยถ้าให้กฎควบคุม π เป็นกฎควบคุมที่ทำการเชื่อมโยงระหว่างสถานะ s โดย $s \in S$ และการกระทำ a โดย $a \in A(s)$ และดำเนินไปภายใต้กฎควบคุม π นี้ ดังสมการที่ (2.5) (Sutton, and Barto, 1998) ซึ่งเป็นสมการ สเตท-แวลู่ฟังก์ชัน สำหรับกฎควบคุม π

$$V^{\pi}(s) = E_{\pi}\{R_t \mid s_t = s\} = E_{\pi}\left\{\sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \mid s_t = s\right\} \tag{2.5}$$

เมื่อ $E_{\pi}\{\ \}$ จะเป็นค่าคาดหวังเมื่อตัวกระทำการตัดสินใจกระทำตามกฎควบคุม π และ γ เป็นอัตราการลดทอน (discount rate) ที่มีค่าระหว่าง $0 \le \gamma \le 1$

2.3.1.4 แบบจำลองของสภาวะแวดล้อม (model of environment) คือแบบจำลองของสิ่งแวดล้อมที่เราจะนำวิธีรีอินฟอร์สเมนท์เลิร์นนิ่งเข้าไปประยุกต์ใช้ ซึ่งต้องมีความสามารถที่จะแสดงพฤติกรรมได้เหมือนกับสภาวะแวดล้อมจริงที่เราจะนำไปประยุกต์ใช้ โดยถ้าให้ $P^a_{ss'}$ เป็นความน่าจะเป็นที่จะเกิดการเปลี่ยนสถานะจากการที่สภาวะแวดล้อมอยู่ที่สถานะ s จะเปลี่ยนสถานะเป็น s' เมื่อเลือกการกระทำ a ดังสมการที่ (2.6) (Sutton, and Barto, 1998)

$$P^a_{ss'} = \Pr\{s_{t+1} = s' \mid s_t = s, a_t = a\} \tag{2.6}$$

การนำวิธีรีอินฟอร์สเมนท์เลิร์นนิ่งมาใช้ในการแก้ปัญหานั้น จะเป็นการหากฎควบคุมที่จะก่อให้เกิดผลรางวัลสะสมระยะยาวสูงที่สุด โดยจะทำการปรับปรุงกฎควบคุมนี้ให้ดีขึ้นอยู่โดยตลอดดังนั้นกฎควบคุมใหม่จึงดีกว่าหรือเท่ากับกฎควบคุมเดิม ซึ่งทำให้ได้กฎควบคุมอย่างน้อยหนึ่งกฎควบคุมที่ดีกว่าหรือเท่ากับกฎควบคุมอื่นๆ ซึ่งเรียกว่ากฎควบคุมที่เหมาะสม (optimal policy) และเมื่อให้ระบบดำเนินไปตามกฎควบคุมนี้ จะได้สเตตแวลู่ฟังก์ชั่นที่เป็นสเตตแวลู่ฟังก์ชั่นที่เหมาะสม (optimal state-value function) ซึ่งแทนด้วย $V*$ ดังสมการ (2.7) (Sutton, and Barto, 1998)

$$V^*(s) = \max_{\pi} V^{\pi}(s) \tag{2.7}$$

เมื่อ ทุกๆ $s \in S$

เมื่อให้ระบบดำเนินไปตามกฎควบคุมที่เหมาะสมนี้ จะได้แอคชั่นแวลู่ฟังก์ชั่นที่เป็นแอคชั่นแวลู่ฟังก์ชั่นที่เหมาะสม (optimal action-value function) ซึ่งแทนด้วย Q^* ดังสมการ (2.8)

$$Q^*(s,a) = \max_{\pi} Q^{\pi}(s,a) \tag{2.8}$$

เมื่อ ทุกๆ $s \in S$ และ $a \in A(s)$

สำหรับในทุกๆ คู่ของสถานะและการกระทำ (state-action pair) (s,a) จะได้ฟังก์ชั่นของค่าคาดหวังที่จะได้รับเมื่อเลือกการกระทำ a และสิ่งแวดล้อมอยู่ในสถานะ s และดำเนินไปด้วยกฎควบคุมที่เหมาะสม จะได้ความสัมพันธ์ของแอคชั่นแวลู่ฟังก์ชั่นที่เหมาะสม Q^* และสเตตแวลู่ฟังก์ชั่นที่เหมาะสม V^* ดังสมการที่ (2.9) (Sutton, and Barto, 1998)

$$Q^*(s,a) = E\left\{ r_{t+1} + \gamma V^*(s_{t+1}) \mid s_t = s, a_t = a \right\} \tag{2.9}$$

2.4 กระบวนการตัดสินใจแบบมาร์คอฟกรณีที่สิ่งที่ได้จากการสังเกตไม่ชัดเจน

วิธีรีอินฟอร์สเมนท์เลิร์นนิ่งจะเป็นกระบวนการเรียนรู้ ซึ่งสามารถใช้ได้กับสิ่งแวดล้อมที่มีกระบวนการตัดสินใจแบบมาร์คอฟ ในสถานการณ์ที่ตัวกระทำการตัดสินใจทราบสถานะของสิ่งแวดล้อมได้อย่างชัดเจน (completely observable Markov decision process) แต่ในความเป็นจริงแล้วมีหลายกรณีที่ตัวกระทำการตัดสินใจ ไม่สามารถที่จะทราบสถานะของสิ่งแวดล้อมได้อย่างชัดเจน หรือมีข้อมูลของสิ่งแวดล้อมนั้นเพียงบางส่วนซึ่งการทำงานของการเฝ้าตรวจสถานะของโครงข่ายที่ถือได้ว่า เป็นการทำงานภายใต้สภาวะที่สิ่งที่ได้จากการสังเกตไม่ชัดเจนนี้เช่นกัน (Steinder and Sethi, 2004) โดยสามารถอธิบายลักษณะของการเกิดกรณีที่สิ่งที่ได้จากการสังเกตจากสิ่งแวดล้อมมีความไม่ชัดเจนโดยมีลักษณะดังรูปที่ 2.2 หากพิจารณาที่ช่วงเวลา $t-1$ สมมุติให้สิ่งแวดล้อมอยู่ในสภาวะ s_{t-1} ซึ่งเมื่อไม่ทราบสถานะที่แท้จริงทำให้ทราบเพียงข้อมูลที่สังเกตได้แทนด้วย o_{t-1} เมื่อตัวกระทำการตัดสินใจเลือกการกระทำ a_{t-1} ทำให้ได้รับผลรางวัล $g(s_{t-1}, a_{t-1})$ และสิ่งแวดล้อมจะเปลี่ยนเป็นสถานะถัดไปเป็น s_t อย่างไรก็ตามเมื่อพิจารณาในกรณีที่มีข้อมูลของสภาวะแวดล้อมไม่ชัดเจน จึงไม่สามารถที่จะทราบสถานะที่แท้จริงของสิ่งแวดล้อมว่าอยู่ในสถานะ s_t ได้ ดังนั้นตัวกระทำการตัดสินใจจะทราบเพียงสถานะที่สังเกตได้เป็น o_t เท่านั้น (Usaha, 2004) และเมื่อพิจารณากรณีที่สิ่งแวดล้อมนั้นมีคุณสมบัติของกระบวนการตัดสินใจแบบมาร์คอฟ (Markov Decision Process หรือ MDP) ซึ่งตัวกระทำการตัดสินใจจะต้องทำการตัดสินใจ ด้วยกระบวนการ

ตัดสินใจแบบมาร์คอฟกรณีที่สิ่งที่ได้จากการสังเกตไม่ชัดเจน (Partially Observable Markov Decision Process หรือ POMDP)

พิจารณาที่กระบวนการตัดสินใจแบบมาร์คอฟที่มีสถานะทั้งหมด $S=\{1,...,n\}$ มีการเปลี่ยนสถานะด้วยความน่าจะเป็น P^{π} ดำเนินไปภายใต้กฎควบคุม π เมื่ออยู่ภายใต้เงื่อนไขของกรณีสิ่งที่ได้จากการสังเกตไม่ชัดเจน ทำให้ตัวกระทำการตัดสินใจไม่สามารถทราบสถานะที่แท้จริงได้ซึ่งจะทราบเพียงสถานะที่ได้จากการสังเกต o ดังนั้นความสัมพันธ์ของสถานะและการกระทำจึงสามารถแสดงในรูปของความน่าจะเป็นได้ว่า $S\times A\rightarrow P_O$ โดยที่ S เป็นเซตของสถานะทั้งหมดที่เป็นไปได้ A เป็นเซตของการกระทำทั้งหมดที่เป็นไปได้ P_O คือความน่าจะเป็นที่กระจายอยู่ในเซตของสิ่งที่ได้จากการสังเกต O ส่วน o คือสิ่งที่ได้จากการสังเกตโดยที่ $o\in O$ และ a คือการกระทำที่ถูกเลือกภายใต้กฎควบคุม π โดยที่ $a\in A$ จะสามารถพิจารณาได้ว่าที่ช่วงเวลา t สมมุติให้สิ่งแวดล้อมอยู่ในสถานะ s_t เมื่อตัวกระทำการตัดสินใจเลือกการกระทำ a_t ทำให้ได้รับผลรางวัล $g(s_t,a_t)$ และสิ่งแวดล้อมจะเปลี่ยนเป็นสถานะถัดไปเป็น s_{t+1} แต่เมื่อพิจารณาในกรณีที่มีข้อมูลของสภาวะแวดล้อมไม่ชัดเจน จึงไม่สามารถที่จะทราบสถานะที่แท้จริงของสิ่งแวดล้อมได้ว่าอยู่ในสถานะ s_{t+1} ได้ ดังนั้นตัวกระทำการตัดสินใจจะทราบเพียงสถานะที่สังเกตได้เป็น o_{t+1} ดังรูปที่ 2.2

เนื่องจากสถานะที่ได้จากการสังเกตที่ไม่ชัดเจนนี้เอง จึงมีการแทนสถานะนี้ด้วยความน่าจะเป็นที่เรียกว่าบีลิฟสเตต (belief state) ด้วยการใช้ความน่าจะเป็นแทนความไม่ชัดเจนของสถานะที่สังเกตได้ซึ่งความน่าจะเป็นนี้จะบอกถึงความเป็นไปได้ที่สิ่งแวดล้อมจะอยู่ในสถานะนั้นๆ จึงสามารถเขียนประวัติแทนด้วย H ที่เกิดขึ้นตั้งแต่เวลาที่ 0 ถึงเวลาที่ t ได้ว่า $H_t=(\mathbf{b}_t,a_t,...,\mathbf{b}_0,a_0)$ เมื่อเชื่อมโยง H_t ให้อยู่ในรูปของบีลิฟสเตต จะได้ลำดับขั้นของสิ่งที่ได้จากการสังเกตและการกระทำได้ว่า $\bar{\mathbf{b}}_t=\left[\bar{b}_t(1),...,\bar{b}_t(n)\right]\in\mathbf{B}$ เมื่อ $\mathbf{B}$ เป็นเซตของการกระจายทั้งหมดบนปริภูมิสเตต (state space) แทนด้วย S สามารถเขียนได้ดังสมการ (2.10) (Usaha, 2004)

$$\bar{b}_t(j)=\Pr\left[s_t=j\mid o_t,...,o_0;a_{t-1},...,a_0;\bar{\mathbf{b}}_0\right] \tag{2.10}$$

เมื่อ $s_t\in S$

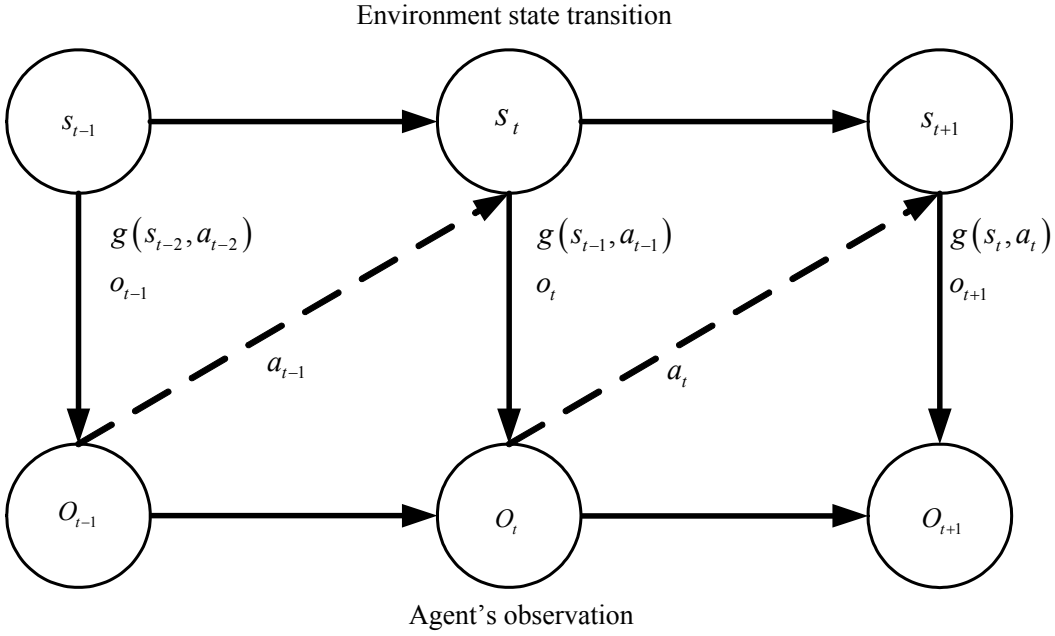
$\bar{\mathbf{b}}_0$ เป็นบิลีฟสเตตเริ่มต้น

$\mathbf{B}=\left\{\bar{\mathbf{b}}:\bar{\mathbf{b}}\in[0,1]^n,\sum_{\forall j\in S}\bar{b}(j)=1\right\}$ เป็นเซตของการกระจายทั้งหมดบนปริภูมิสเตต S

สำหรับทุกๆ การกระทำ a_t ที่ถูกเลือกกระทำเมื่ออยู่ในสถานะ $\bar{\mathbf{b}}_t$ และได้รับผลรางวัล $g(\bar{\mathbf{b}}_t,a_t)$

กระจายบนปริภูมิสเตต (state space) **B** และกฎควบคุมที่เหมาะสม π จะเป็นไปตามเงื่อนไขดังสมการ (2.11) (Usaha, 2004)

$$V_{\pi}\left(\overline{\mathbf{b}}\right)=\max_{\pi}\left\{E_{\pi}\left[\sum_{t=0}^{T-1}g\left(\overline{\mathbf{b}}_{t},a_{t}\right)|\overline{\mathbf{b}}_{0}=\overline{\mathbf{b}}\right]\right\}\ for\ all\ \overline{\mathbf{b}}\in B \tag{2.11}$$



รูปที่ 2.2 ลักษณะของการเกิดกรณีที่สิ่งที่ได้จากการสังเกตไม่ชัดเจน

เมื่อพิจารณาการทำงานของการเฝ้าตรวจสถานะของโครงข่ายจะพบว่า การโพลล์ในแต่ละครั้งนั้นสิ่งที่ตัวกระทำการตัดสินใจได้รับกลับมาเป็นเพียงสถานะที่สังเกตได้เท่านั้น ซึ่งเกิดจากข้อมูลอาจจะสูญหาย หรืออาจเกิดจากความผิดพลาดของการ รับ-ส่ง ข้อมูล หรือปัจจัยอื่นๆ ซึ่งทำให้ไม่ทราบสถานะที่แท้จริงของโครงข่าย และการทำงานของการเฝ้าตรวจสถานะของโครงข่ายยังมีลักษณะคล้ายกับคุณสมบัติของกระบวนการตัดสินใจแบบมาร์คอฟ เนื่องจากสถานะถัดไปของโครงข่ายที่สังเกตได้จะขึ้นอยู่กับสถานะของโครงข่ายในปัจจุบัน และการตัดสินใจของตัวกระทำการตัดสินใจเลือกการกระทำในขณะนั้นเท่านั้น ดังนั้นการทำงานของการเฝ้าตรวจสถานะของโครงข่ายจึงเป็นการทำงานภายใต้เงื่อนไขของกระบวนการตัดสินใจแบบมาร์คอฟ กรณีที่สิ่งที่ได้จากการสังเกตไม่ชัดเจน (Partially Observable Markov Decision Process หรือ POMDP) (He, 2003)

การใช้วิธีรีอินฟอร์สเมนท์เลิร์นนิ่งมาใช้กับกระบวนการตัดสินใจแบบมาร์คอฟกรณีที่สิ่งที่ได้จากการสังเกตไม่ชัดเจน โดยมีผู้นำมาใช้ในการแก้ปัญหาของระบบโครงข่ายสื่อสาร เช่น การนำมาใช้ในการหาเส้นทางแบบมีคุณภาพ ในการให้บริการในโครงข่ายแบบไร้สาย แบบแอดฮอค (quality-of-service routing in mobile ad hoc networks) (Usaha, 2004) นำมาใช้ในการบริหารจัดการอุปกรณ์ขัดข้องของโครงข่าย เอทีเอ็ม (ATM network) (He, 2003) การนำมาใช้ในการแก้ปัญหาของการเฝ้าตรวจสถานะของโครงข่ายที่มีความแม่นยำ และสามารถตรวจพบจุดที่อุปกรณ์ของโครงข่ายขัดข้องได้ด้วยความรวดเร็วโดยกำหนดสถานะของอุปกรณ์ในโครงข่าย (ขัดข้อง หรือ ปกติ) เป็นสถานะของสิ่งแวดล้อมและตัวกระทำการตัดสินใจจะทำการเรียนรู้ที่จะเลือกการกระทำที่สามารถวิเคราะห์จุดขัดข้องที่แท้จริงได้อย่างรวดเร็ว (He, 2003)

2.5 วิธีรีอินฟอร์สเมนท์เลิร์นนิ่งแบบออนโพลิซีมอนติคาร์โล

วิธีรีอินฟอร์สเมนท์เลิร์นนิ่งแบบออนโพลิซีมอนติคาร์โล (On-policy Monte Carlo หรือ ONMC) จะใช้วิธีการเรียนรู้โดยพิจารณาจากค่าเฉลี่ยของผลรางวัลที่ได้รับจากการจำลองแบบโดยมีการทำงานเป็นฉากหรือรอบ ตัวอย่างเช่น การค้นหาอุปกรณ์ที่ขัดข้องในโครงข่ายจากการเริ่มค้นหาด้วยการโพลล์ไปยังอุปกรณ์ต่างๆ จนพบอุปกรณ์ที่ขัดข้อง และดำเนินการแก้ไขให้สามารถใช้งานได้จะเป็นการทำงานหนึ่งฉากหรือหนึ่งรอบ ซึ่งเราเรียกว่าเอพพิโซด (episode) โดยจะทำการแบ่งประสบการณ์ของการเรียนรู้ออกเป็นหลายๆ เอพพิโซดและในทุกๆ ครั้งของการสิ้นสุดเอพพิโซดจะแสดงถึงการเลือกการกระทำที่สมควรที่จะกระทำ และในแต่ละเอพพิโซดจะนำค่าเฉลี่ยของผลรางวัลจากเอพพิโซดนั้นมาปรับปรุงกฎควบคุมซึ่งมีผลให้กฎควบคุมถูกปรับปรุงในทุกๆ เอพพิโซด เช่นในรูป 2.3 จะเริ่มการทำงานของเอพพิโซดแรกด้วยการใช้กฎควบคุม π_0 ระบบจะดำเนินไปเพื่อค้นหาการกระทำที่สมควรที่จะกระทำ และเมื่อได้การกระทำที่สมควรที่จะกระทำแล้วจะเป็นการสิ้นสุดเอพพิโซด และจะทำการเก็บผลรางวัลที่เกิดขึ้น ตั้งแต่เริ่มต้นเอพพิโซดจนจบเอพพิโซด และเก็บผลรางวัลนี้ในรูปของคิวแวลู่ภายใต้กฎควบคุม π_0 แทนด้วย $Q^{\pi 0}$ และนำค่าเฉลี่ยของผลรางวัลจากเอพพิโซดนั้นมาปรับปรุงกฎควบคุม ซึ่งมีผลให้กฎควบคุมถูกปรับปรุงในทุกๆ เอพพิโซด จากนั้นระบบจะเริ่มเอพพิโซดใหม่และมีการทำงานเช่นนี้ต่อๆ ไปจนกระทั่งได้กฎควบคุมที่เหมาะสมแทนด้วย π^* ซึ่งเป็นจุดประสงค์ของวิธีรีอินฟอร์สเมนท์เลิร์นนิ่งแบบออนโพลิซีมอนติคาร์โล

$$\pi_0 \xrightarrow{E} Q^{\pi 0} \xrightarrow{I} \pi_1 \xrightarrow{E} Q^{\pi 1} \xrightarrow{I} \cdots \xrightarrow{I} \pi^* \xrightarrow{E} Q^*$$

รูปที่ 2.3 ลักษณะของการทำงานเป็นเอพพิโซด

วิธีออนโพลิซีมอนติคาร์โลสามารถนำมาใช้กับ กระบวนการตัดสินใจแบบมาร์คอฟกรณีที่สิ่งที่ได้จากการสังเกตไม่ชัดเจนได้ (Usaha, 2004) โดยเมื่อพิจารณากระบวนการตัดสินใจแบบมาร์คอฟกรณีที่สิ่งที่ได้จากการสังเกตไม่ชัดเจน ด้วยวิธีออนโพลิซีมอนติคาร์โลซึ่งกำหนดให้ S แทนสถานะ A แทนการกระทำและ O แทนสิ่งที่ได้จากการสังเกตเมื่อพิจารณาสิ่งที่เกิดขึ้นในแต่ละเอพพิโซดเมื่อให้ o_{T-1} เป็นสถานะที่สิ้นสุดเอพพิโซดและผลรางวัล $g(o_{T-1}, a)$ มีค่าเท่ากับ 0 ในทุกๆ การกระทำ ถ้าให้เหตุการณ์ที่เกิดขึ้นคือ $\{o_0, a_0, g(o_0, a_0), ..., o_{T-1}, a_{T-1}, g(o_{T-1}, a_{T-1})\}$ เมื่อ T คือช่วงเวลาของเอพพิโซดที่เกิดขึ้นภายใต้กฎควบคุม $\pi : O \to A$ โดยเราจะพิจารณาเฉพาะคู่ของสิ่งที่ได้จากการสังเกตกับการกระทำที่ถูกเลือก (o, a) เมื่อ $o \in O$ และ $a \in A$

ถ้าให้ t เป็นจำนวนเอพพิโซด t^{th} เป็นเอพพิโซดที่มี (o, a) เกิดขึ้น ถ้า T_t คือจำนวนขั้นตอน (time steps) ที่เกิดขึ้นในเอพพิโซดที่ t และ $\tau_t(o, a)$ เมื่อ $0 \le \tau_t(o, a) \le T_t - 1$ เป็นจำนวนขั้นตอนที่พบ (o, a) เป็นครั้งแรก $Q^{\pi_t}(o, a)$ หมายถึงค่าผลรางวัลที่คาดหวังเมื่อเริ่มจากคู่ของสิ่งที่ได้จากการสังเกตและการกระทำ (o, a) ที่ดำเนินไปภายใต้กฎควบคุม π_t

ถ้าเริ่มต้นจากการใช้กฎควบคุม π_0 คิวฟังก์ชันของสิ่งที่ได้จากการสังเกตและการกระทำ $Q^{\pi_0}(o, a)$ เป็นค่าที่จุดเริ่มต้นเอพพิโซดดังนั้นในทุกๆ เอพพิโซด t จะมีการเลือกการกระทำและผลจากการกระทำเหล่านั้นจะถูกนำมาสร้างกฎควบคุม π_t ที่จุดสิ้นสุดของเอพพิโซด t ซึ่งสามารถประมาณค่าคิวฟังก์ชันของสิ่งที่ได้จากการสังเกตและการกระทำของ (o, a) ได้ดังสมการ (2.12) (Usaha, 2004)

$$Q^{\pi_t}(o, a) = Q^{\pi_{t-1}}(o, a) + \frac{1}{t}\left(\sum_{k=\tau_t(o,a)}^{T_t - 1} g(o_k, a_k) - Q^{\pi_{t-1}}(o, a) \right) \qquad (2.12)$$

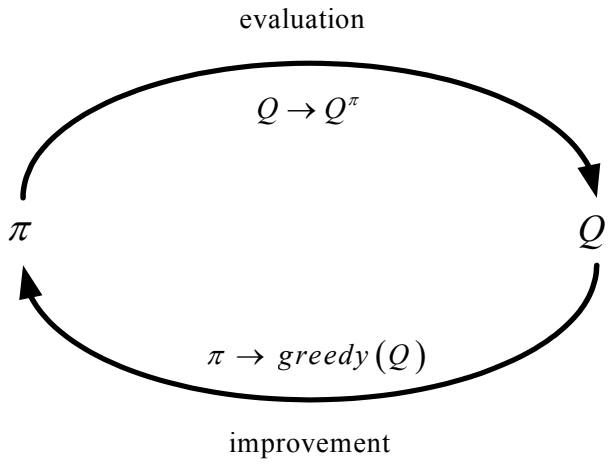
เมื่อพิจารณาในเทอมของผลรวมของผลรางวัลที่เกิดจาก (o, a) ในเอพพิโซด t จะได้การกระทำที่อาจจะถูกเลือก a^* หากใช้วิธีการสำหรับกฎควบคุมแบบกรีดี้ (ε) ดังสมการ (2.13) (Usaha, 2004)

$$a^* = \arg\max \left\{ Q^{\pi_t}(o, a) \right\} \qquad (2.13)$$

ในทำนองเดียวกันหากใช้กฎควบคุมแบบ $\varepsilon - greedy$ policy โดย $\varepsilon \in [0,1]$ จะสามารถปรับปรุงกฎควบคุมได้ดังสมการ (2.14) (Usaha, 2004)

$$\pi_{t+1}(o) = \begin{cases} a^* \text{ with probability } \varepsilon + \dfrac{(1-\varepsilon)}{|A|} \\ a \in A - \{a^*\} \text{ with probability } \dfrac{(1-\varepsilon)}{|A|} \end{cases} \tag{2.14}$$

เมื่อ $|A|$ คือขนาดของปริภูมิการกระทำ (action space) จึงส่งผลให้ขั้นตอนวิธีนี้สามารถปรับปรุงกฎควบคุมจนเข้าใกล้กฎควบคุมที่เหมาะสมได้ โดยจะมีวงรอบของการปรับปรุงกฎควบคุมดังรูป 2.4 และคิวฟังก์ชัน (Q) ที่ได้จากผลรวมของผลรางวัลที่เกิดขึ้นจากการดำเนินไปภายใต้กฎควบคุม π ซึ่งแทนด้วย Q^{π} จะถูกนำไปใช้เป็นส่วนที่ใช้ปรับปรุงกฎควบคุม π และส่งผลให้กฎควบคุมที่ได้รับการปรับปรุงใหม่นี้ดีกว่ากฎควบคุมเดิม และจะใช้กฎควบคุม π นี้ไปใช้ในการหาค่าคิวฟังก์ชันอีกครั้งและจะทำการปรับปรุงกฎควบคุมนี้หลายต่อหลายครั้งซึ่งส่งผลให้กฎควบคุมได้รับการปรับปรุงให้ดีขึ้นอยู่โดยตลอดจนได้กฎควบคุม π ที่เหมาะสม



รูปที่ 2.4 วงรอบของการปรับปรุงกฎควบคุม

ผังการทำงานของขั้นตอนวิธีแบบออนโพลิซีมอนติคาร์โล มีรายละเดียดของตัวแปรต่างๆ ดังตารางที่ 2.1 และมีลำดับขั้นของการทำงาน ของขั้นตอนวิธีตามผังการทำงานของขั้นตอนวิธีแบบออนโพลิซีมอนติคาร์โล

ตารางที่ 2.1 ความหมายและสัญลักษณ์ต่างๆ ของผังการทำงานของขั้นตอนวิธีออนโพลิซี มอนติคาร์โล

สัญลักษณ์	ความหมาย
o	สถานะที่ได้จากการสังเกต (observation)
a	การกระทำที่เลือกกระทำ (action)
$Q(o,a)$	เป็นคิวฟังก์ชันที่ใช้ในการเก็บผลรางวัลที่เกิดขึ้นในทุกๆ คู่ของสิ่งที่ได้จากการสังเกตและการกระทำ
$Returns(o,a)$	ใช้ในการเก็บผลรางวัลที่เกิดขึ้นในทุกๆ คู่ของสิ่งที่ได้จากการสังเกตและการกระทำในหนึ่งเอพพิโซดซึ่งในการเริ่มต้นจะถูกตั้งค่าเป็น 0
π	เป็นกฎควบคุมซึ่งใช้แบบ ε-greedy policy คือใช้การเลือกการกระทำที่มีค่า $Q(o,a)$ สูงสุดด้วยความน่าจะเป็นเท่ากับ $\varepsilon+\frac{(1-\varepsilon)}{\lvert A(o)\rvert}$ และเลือกการกระทำอื่นๆ ด้วยความน่าจะเป็นเท่ากับ $\frac{(1-\varepsilon)}{\lvert A(o)\rvert}$
$A(o)$	การกระทำในทุกสิ่งที่ได้จากการสังเกต
R	เป็นผลรางวัลที่ได้ซึ่งเริ่มจากการไปพบคู่ของสิ่งที่ได้จากการสังเกตและการกระทำที่พบเป็นครั้งแรก (first visit) และทำการเพิ่มเข้าสู่ $Returns(o,a)$
$a*$	เป็นการกระทำที่ก่อให้เกิดผลรางวัลสูงสุดที่ได้มาจาก $Q(o,a)$
$\pi(o,a)$	เป็นกฎควบคุมที่ใช้ในการเลือกการกระทำเมื่อพบสิ่งที่ได้จากการสังเกต o โดยจะเลือกการกระทำ $a*$ ด้วยความน่าจะเป็น $\varepsilon+\frac{(1-\varepsilon)}{\lvert A(o)\rvert}$ และเลือกการกระทำอื่นๆ ด้วยความน่าจะเป็น $\frac{(1-\varepsilon)}{\lvert A(o)\rvert}$
ε	กรีดี้ ของกฎควบคุม (greediness of the policy) โดยมีค่ามากกว่าหรือเท่ากับ 0 แต่น้อยกว่าหรือเท่ากับ 1

ผังการทำงานของขั้นตอนวิธีแบบออนโพลิซีมอนติคาร์โล

1. Initialize, for all $o \in O, a \in A(o)$:
2. $Q(o,a) \leftarrow$ arbitrary
3. $Returns(o,a) \leftarrow$ empty list
4. $\pi \leftarrow$ an arbitrary $\in -greedy$ policy

5. Repeat forever:
6. (a) Generate an episode using π
7. (b) For each pair o,a appearing in the episode:
8. $R \leftarrow$ return following the first occurrence of o,a
9. Append R to $Returns\left(o,a\right)$
10. $Q\left(o,a\right) \leftarrow \text{average}\left(Returns\left(o,a\right)\right)$
11. (c) For each o in the episode:
12. $a^* \leftarrow \arg\max_a Q\left(o,a\right)$
13. For all $a \in A\left(o\right)$:
14. $$\pi\left(o,a\right) \leftarrow \begin{cases} \varepsilon + \dfrac{\left(1-\varepsilon\right)}{\left|A\left(o\right)\right|} & \text{if } a = a^* \\ \dfrac{\left(1-\varepsilon\right)}{\left|A\left(o\right)\right|} & \text{if } a \neq a^* \end{cases}$$

2.6 วิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์

วิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ซึ่งนำเสนอโดย (He, 2003) มีจุดประสงค์ในการลดปริมาณโพลล์ลิ่งโอเวอร์เฮดที่เกิดจากการเฝ้าตรวจสถานะของโครงข่าย และทำการค้นหาตำแหน่งที่อุปกรณ์ขัดข้อง รวมทั้งทำการพิจารณาถึงกรณีที่สิ่งที่ได้จากกการสังเกตไม่ชัดเจน หรือการพยายามแก้ปัญหาที่เกิดขึ้นจากการเฝ้าตรวจสถานะของโครงข่ายด้วยการสร้างลำดับขั้นของการโพลล์ เพื่อทำการยืนยันสถานะที่แท้จริง โดยได้เลือกใช้วิธีรีอินฟอร์สเมนท์เลิร์นนิ่งในการแก้ปัญหาเช่นกัน และถือได้ว่าวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์นี้ เป็นงานวิจัยที่สามารถที่จะลดโพลล์ลิ่งโอเวอร์เฮด และค้นหาตำแหน่งที่อุปกรณ์ขัดข้อง ภายใต้กรณีสิ่งที่ได้จากการสังเกตไม่ชัดเจน ที่ให้ผลการทำงานที่ดีและสามารถลดโพลล์ลิ่งโอเวอร์เฮดลงได้ แต่อย่างไรก็ตามวิธีการนี้ยังคงมีปัญหาในส่วนของความยุ่งยากซับซ้อน ที่ใช้ในการคำนวณและต้องใช้ปริมาณหน่วยความจำในการเก็บข้อมูลจำนวนมากซึ่งมีหลักการเบื้องต้นดังนี้

โดยปกติแล้วการเรียนรู้ของวิธีรีอินฟอร์สเมนท์เลิร์นนิ่ง จะเป็นการเรียนรู้จากจุดเริ่มต้นภายใต้กฎการควบคุมแบบสุ่มด้วยการลองผิดลองถูก จนเกิดการเรียนรู้ขึ้นซึ่งหมายถึงว่าตัวกระทำการตัดสินใจจะต้องติดสินใจที่ผิดพลาดครั้งแล้วครั้งเล่า จนกว่าจะสามารถที่จะเรียนรู้ขึ้นมาได้ ซึ่งการกระทำผิดดังกล่าวจะเกิดขึ้นไม่ได้เลยเมื่อทำงานอยู่ในระบบจริง ดังนั้น (He, 2003) จึงใช้การจำลองแบบเพื่อสร้างการเรียนรู้โดยทำการแบ่งการเรียนรู้เป็นสองระยะ (two-phase learning) โดยระยะแรกจะเป็นการเรียนรู้ที่เกิดจากการจำลองแบบ โดยการสร้างรูปแบบการทำงานที่เหมือนกับระบบการทำงานจริง และกำหนดว่าทุกสิ่งที่ได้จากการสังเกตมีความถูกต้อง (completely-observed) เพื่อทำการสร้างกฎควบคุมขึ้นแล้วจึงนำกฎควบคุมนี้ไปใช้กับการทำงานในระยะที่สอง ซึ่งเป็นการทำงานกับระบบจริงซึ่งในส่วนนี้สิ่งที่ได้จากการสังเกตจะมีข้อมูลเพียงบางส่วนเท่านั้น (partially observed) หลังจากนั้นกฎควบคุมนี้จะถูกใช้และมีการปรับปรุงอยู่โดยตลอด เพื่อให้สามารถครอบคลุมถึงในส่วนที่ไม่สามารถจำลองแบบได้ จนกระทั่งได้กฎควบคุมที่ดีที่สามารถส่งผลได้ในทุกๆ คู่ของสถานะและการกระทำ (state-action pair) ซึ่งจากการสร้างการเรียนรู้แบบสองระยะนี้จะเป็นการลดความเสี่ยงของการเลือกการกระทำที่ผิดพลาด เพื่อลดความเสียหายที่จะเกิดขึ้นในการทำงานในระบบจริง (He, 2003)

วิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ ได้ทำการแบ่งการเรียนรู้เป็นสองระยะ (two-phase learning) โดยระยะแรกจะเป็นการเรียนรู้ที่เกิดจากการจำลองแบบที่กำหนดว่าทุกสิ่งที่ได้จากการสังเกตมีความถูกต้อง (completely-observed) และการทำงานในระยะที่สองซึ่งเป็นการทำงานในกรณีที่สิ่งที่ได้จากการสังเกตจะมีข้อมูลเพียงบางส่วนเท่านั้น (partially observed) ความหมายและตัวแปรต่างๆ ของผังการทำงานของขั้นตอนวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ระยะแรก และมีความหมายและสัญลักษณ์ต่างๆ ดังตารางที่ 2.2 และมีการทำงานตามผังการทำงานของขั้นตอนวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ระยะแรก

ตารางที่ 2.2 ความหมายและสัญลักษณ์ต่างๆ ของผังการทำงานของขั้นตอนวิธีโปรแอคทีฟเน็ตเวิร์คเน็ตเวิร์คมาเนจเมนท์ระยะแรก

สัญลักษณ์	**ความหมาย**
$Q(s,a)$	คิวแวลู่ของสถานะ s และการกระทำ a
$e(s,a)$	อิลิจิบิลิตี้เทรซ (eligibility trace) ของสถานะ s และการกระทำ a
$T(s,a,s')$	จำนวนของการกระทำ a ที่ถูกเลือกแล้วส่งผลให้สถานะเปลี่ยนจาก s เป็น s'

ตารางที่ 2.2 ความหมายและสัญลักษณ์ต่างๆ ของผังการทำงานของขั้นตอนวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ระยะแรก (ต่อ)

สัญลักษณ์	ความหมาย
$O(s',a,z')$	สิ่งที่ได้จากการสังเกตหลังจากเลือกการกระทำ a แล้วส่งผลให้สถานะเปลี่ยนเป็น s'และได้สถานะของสิ่งที่ได้จากการสังเกตเป็น z'
$Q(s',a')$	คิวแวลู่ของสถานะใหม่ s' และการกระทำที่เลือกกระทำครั้งใหม่ a'
s	สถานะเดิม
s'	สถานะใหม่
a	การกระทำที่เลือกกระทำ
a'	การกระทำที่เลือกกระทำครั้งใหม่
r	ผลรางวัลที่ได้รับ
γ	ปัจจัยการลดทอน (discount factor) ของอิลิจิบิลิตี้เทรซ โดยมีค่ามากกว่า 0 แต่น้อยกว่า 1
λ	ปัจจัยการลดทอนของอิลิจิบิลิตี้เทรซ (eligibility trace) โดยมีค่ามากกว่าหรือเท่ากับ 0 แต่น้อยกว่าหรือเท่ากับ 1
δ	ค่าผิดพลาดทีดี (TD error)
α	อัตราการเรียนรู้ (learning rate) โดยมีค่ามากกว่า 0 แต่น้อยกว่า 1
η	ปัจจัยการลดทอน (decay-factor) ของอัตราการเรียนรู้โดยมีค่ามากกว่า 0 แต่น้อยกว่า 1
ε	กรีดี้ของกฎควบคุม (greediness of the policy) โดยมีค่ามากกว่าหรือเท่ากับ 0 แต่น้อยกว่าหรือเท่ากับ 1

ผังการทำงานของขั้นตอนวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ระยะแรก

1. In a completely- observed simulator, initialize $Q(s,a)$ randomly;
   eligibility trace $e(s,a)=0$, system model $T(s,a,s')=0, O(s',a,z')=0.$
   $\forall s,a,s',z'$
2. repeat
3. Randomly choose state s, action a to generate a simulation trajectory.
4. repeat
5. Take action a with ε - *greedy*, receive reward r, reach next state s' and make observation z'
6. $T(s,a,s') := T(s,a,s')+1$
7. $O(s',a,z') := O(s',a,z')+1$
8. Take $a' = \arg\max_{\hat{a}} Q(s',\hat{a})$ with ε - *greedy*
9. Compute TD error $\delta = r+\gamma Q(s',a')-Q(s,a)$
10. Update eligibility trace $e(s,a) := \gamma\lambda e(s,a)+1$
11. for all s,a do
12. $e(s,a) := \gamma\lambda e(s,a)$
13. end for
14. for all s,a do
15. $Q(s,a) := Q(s,a)+\alpha\delta e(s,a)$
16. end for
17. $s := s'$, $a := a'$
18. until s is the terminal state or reach the step -limit
19. Decay learning rate: $\alpha=\eta\alpha$
20. until reach the episode-limit
21. Normalize $T(s,a,s')$ and $O(s',a,z')$ to make probability distributions.

การทำงานของขั้นตอนวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ระยะที่สอง ซึ่งเป็นการทำงานในกรณีที่สิ่งที่ได้จากการสังเกตจะมีข้อมูลเพียงบางส่วนเท่านั้น (partially observed) และใช้การระบุสถานะของระบบโดยแทนด้วยความน่าจะเป็นของสถานะ (belief state) หรือการใช้ความน่าจะเป็นแทนความไม่ชัดเจนของสถานะที่สังเกตได้ และความน่าจะเป็นนี้จะบ่งบอกถึงความเป็นไปได้ที่สิ่งแวดล้อมจะอยู่ในสถานะนั้น ความหมายและตัวแปรต่างๆ ของผังการทำงานของขั้นตอนวิธีแบบโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ระยะที่สองและมีความหมายและสัญลักษณ์ต่าง ดังตารางที่ 2.3 และมีการทำงานตามผังการทำงานของขั้นตอนวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ระยะที่สอง

ตารางที่ 2.3 ความหมายและสัญลักษณ์ต่างๆ ของผังการทำงานของขั้นตอนวิธีโปรแอคทีฟ เน็ตเวิร์คมาเนจเมนท์ระยะที่สอง

สัญลักษณ์	ความหมาย
$Q(s,a)$	คิวแวลู่ของสถานะ s และการกระทำ a
$e(s,a)$	อิลิจิบิลิตี้เทรซ (eligibility trace) ของสถานะ s และการกระทำ a
b	ความน่าจะเป็นของสถานะ (belief state)
b'	ความน่าจะเป็นของสถานะ (belief state) ที่ได้รับการปรับปรุง
a	การกระทำที่เลือกกระทำ
a'	การกระทำที่เลือกกระทำครั้งใหม่
$\tilde{Q}(\mathbf{b},a)$	คิวแวลู่ของสถานะ b และการกระทำ a ที่ได้จากการประมาณ
$\tilde{Q}(\mathbf{b}',a')$	คิวแวลู่ของสถานะ b และการกระทำ a ที่ได้จากการประมาณครั้งใหม่
r	ผลรางวัลที่ได้รับ
γ	ปัจจัยการลดทอน (discount factor) ของอิลิจิบิลิตี้เทรซ โดยมีค่ามากกว่า 0 แต่น้อยกว่า 1
λ	ปัจจัยการลดทอนของอิลิจิบิลิตี้เทรซ (eligibility trace) โดยมีค่ามากกว่าหรือเท่ากับ 0 แต่น้อยกว่าหรือเท่ากับ 1
δ	ค่าผิดพลาดทีดี (TD error)
α	อัตราการเรียนรู้ (learning rate) โดยมีค่ามากกว่า 0 แต่น้อยกว่า 1
η	ปัจจัยการลดทอน (decay-factor) ของอัตราการเรียนรู้โดยมีค่ามากกว่า 0 แต่น้อยกว่า 1
ε	กรีดี้ของกฎควบคุม (greediness of the policy) โดยมีค่ามากกว่าหรือเท่ากับ 0 แต่น้อยกว่าหรือเท่ากับ 1

ผังการทำงานของขั้นตอนวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ระยะที่สอง

1: Take estimated model T and O from phase-1; take $Q(s,a)$ from phase-1
as initial parameter $e(s,a)=0,\forall s,a$
2: repeat
3: repeat
4: Start from initial belief state **b**
5: Take action a with ε - *greedy* make observation z', receive reward r
6: Update belief state **b'** as in

$$b^{a}_{z'}(s') = P(s' \mid \mathbf{b}, a, z') = \frac{P(z' \mid s', a)P(s' \mid \mathbf{b}, a)}{P(z' \mid \mathbf{b}, a)}$$

using the model estimated from phase-1
7: Take $a' = \arg\max_{\tilde{a}} \sum_{s} Q(s,\hat{a})b'(s)$ with ε - *greedy*
8: Compute approximate $\tilde{Q}$:

$$\tilde{Q}(\mathbf{b},a) = \sum_{s} Q(s,a)b(s)$$

$$\tilde{Q}(\mathbf{b}',a') = \sum_{s} Q(s,a')b'(s)$$

9: Compute TD error $\delta = r + \gamma\tilde{Q}(\mathbf{b}',a') - \tilde{Q}(\mathbf{b},a)$
10: for all s and a do
11: $e(s,a) := \gamma\lambda e(s,a) + \mathbf{b}(s)$
12: $Q(s,a) := Q(s,a) + \alpha\delta e(s,a)$
13: end for
14: $\mathbf{b} := \mathbf{b}', a := a'$
15: until reach the step-limit
16: Decay learning rate $\alpha := \eta\alpha$
17: until reach the episode-limit

2.7 สรุป

การทำงานของการบริหารจัดการโครงข่ายนั้นในส่วนของการเฝ้าตรวจสถานะของโครงข่ายและการค้นหาตำแหน่งที่อุปกรณ์ขัดข้อง ซึ่งสามารถกำหนดให้อยู่ในรูปของการทำงานภายใต้สภาวะที่สิ่งที่ได้จากการสังเกตไม่ชัดเจน ซึ่งวิธีที่จะได้กฎควบคุมที่จะให้ผลรางวัลในระยะยาวที่สูงนั้นสามารถกระทำได้ โดยเลือกใช้วิธีรีอินฟอร์สเมนท์เลิร์นนิ่งที่มีการเรียนรู้เพื่อที่จะตัดสินใจเลือกการกระทำที่ดีซึ่งอยู่ในรูปของผลรางวัลเฉลี่ยในระยะยาว แม้ว่าลักษณะการทำงานของการเฝ้าตรวจ

สถานะของโครงข่ายจะมีข้อมูลที่ได้จากการสังเกตเพียงบางส่วน ซึ่งสามารถใช้กระบวนการตัดสินใจแบบมาร์คอฟกรณีที่สิ่งที่ได้จากการสังเกตไม่ชัดเจนช่วยในการแก้ปัญหาได้ โดยวิธีการที่เลือกใช้คือ แบบออนโพลิซีมอนติคาร์โล ที่มีการทำงานที่คล้ายกับการทำงานของการเฝ้าตรวจสถานะของโครงข่าย

ในบทที่ 3 จะนำเสนอการใช้วิธีออนโพลิซีมอนติคาร์โล เพื่อทำการสมมุติฐานในการนำวิธีออนโพลิซีมอนติคาร์โลมาใช้ลดโพลล์ลิ่งโอเวอร์เฮด โดยทำการศึกษาในโครงข่ายขนาดเล็ก และในบทที่ 4 จะนำเสนอกรณีที่โครงข่ายมีขนาดใหญ่ขึ้น โดยมีจุดประสงค์ที่จะลดโพลล์ลิ่งโอเวอร์เฮด เพื่อเป็นการลดปริมาณการใช้งานแบนด์วิดท์ และวิเคราะห์ข้อมูลของโครงข่ายที่เป็นข้อมูลที่สังเกตได้เพียงบางส่วนให้มีความถูกต้องรวดเร็วและมีการทำงานของขั้นตอนวิธีที่มีความซับซ้อนต่ำ เพื่อลดการใช้งานหน่วยความจำ (memory) ซึ่งจะเป็นการเพิ่มประสิทธิภาพของการบริหารจัดการอุปกรณ์ขัดข้อง ในด้านของความรวดเร็วในการค้นหาจุดที่ขัดข้อง และการบริหารจัดการสมรรถนะ ในด้านของการใช้งานแบนด์วิดท์ที่สามารถลดการสูญเสียที่เกิดจากโพลล์ลิ่งโอเวอร์เฮด

บทที่ 3
การดำเนินการวิจัยในโครงข่ายขนาดเล็ก

3.1 กล่าวนำ

เพื่อทำการทดสอบสมมุติฐาน ในการนำวิธีรีอินฟอร์สเมนท์เลิร์นนิ่งแบบออนโพลิซีมอนติคาร์โลมาใช้กับการบริหารจัดการโครงข่าย ในส่วนของการเฝ้าตรวจสถานะของโครงข่ายโดยมีวัตถุประสงค์เพื่อการลดปริมาณของโพลล์ลิ่งโอเวอร์เฮด (polling overhead) ที่เกิดขึ้นในโครงข่าย จึงได้ทำการวิจัยด้วยการจำลองแบบกับโครงข่ายขนาดเล็ก ด้วยการนำวิธีรีอินฟอร์สเมนท์เลิร์นนิ่งแบบออนโพลิซีมอนติคาร์โล เปรียบเทียบกับวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ (He, 2003) ที่ใช้วิธีสร้างลำดับขั้นของการโพลล์เพื่อลดโพลล์ลิ่งโอเวอร์เฮด และถือได้ว่าวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์นี้ สามารถที่จะลดปริมาณของโพลล์ลิ่งโอเวอร์เฮดได้ในระดับที่ดี ที่มีการนำเสนอในปัจจุบัน

เนื้อหาของบทนี้จะประกอบด้วย หัวข้อ 3.2 การนิยามปัญหาโดยจะกล่าวถึงโครงรูปของโครงข่ายที่ใช้ในการจำลองแบบ การกำหนดพารามิเตอร์ต่างๆ เช่น สถานะของโครงข่าย การกระทำที่เกิดขึ้นกับโครงข่าย สิ่งที่สามารถสังเกตได้ และการกำหนดผลรางวัล หัวข้อ 3.3 เป็นการทดลองการจำลองแบบกับโครงข่ายขนาดเล็ก โดยทำการจำลองแบบด้วยวิธีออนโพลิซีมอนติคาร์โลและวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ หัวข้อ 3.4 เป็นผลการจำลองของแบบจำลองของโครงข่ายขนาดเล็กโดยทำการเปรียบเทียบตัวชี้วัดที่สำคัญ เช่น ปริมาณของการโพลล์ที่เกิดขึ้น ปริมาณการแจ้งตำแหน่งที่อุปกรณ์ขัดข้องได้ถูกตำแหน่ง และจำนวนการกระทำที่ต้องใช้ในการวิเคราะห์หาจุดขัดข้อง หัวข้อ 3.5 เป็นส่วนของการวิเคราะห์ผลการจำลองของแบบจำลองของโครงข่ายขนาดเล็กและความซับซ้อนของขั้นตอนวิธีทั้งสองแบบ และหัวข้อ 3.6 จะเป็นในส่วนของบทสรุป

3.2 การนิยามปัญหา

กระทำการจำลองแบบ โดยนำวิธีรีอินฟอร์สเมนท์เลิร์นนิ่งแบบออนโพลิซีมอนติคาร์โลเปรียบเทียบกับวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ เพื่อทำการเปรียบเทียบค่าชี้วัดต่างๆ ที่เกิดขึ้น เช่น จำนวนครั้งของการกระทำ จำนวนผลรางวัลที่เกิดขึ้น โดยทำการจำลองแบบในโครงข่ายที่มีอุปกรณ์จำนวนน้อยและไม่ซับซ้อน

โครงรูปของโครงข่ายที่ใช้ในการจำลองแบบ ได้เลือกใช้แบบเดียวกับที่ใช้ในการจำลองแบบของวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ (He, 2003) โดยจะประกอบด้วยอุปกรณ์โครงข่าย ซึ่งในที่นี้หมายถึงสวิตซ์ (switch) จำนวน 6 สวิตซ์ (SW1-SW6) โดยมีลิงค์ (link) เชื่อมต่อถึงกันและสวิตซ์แต่ละสวิตซ์จะประกอบด้วยเอ็นโหนด (end node) เชื่อมต่ออยู่ 2 เอ็นโหนด (N11-N62) ดังรูปที่ 3.1

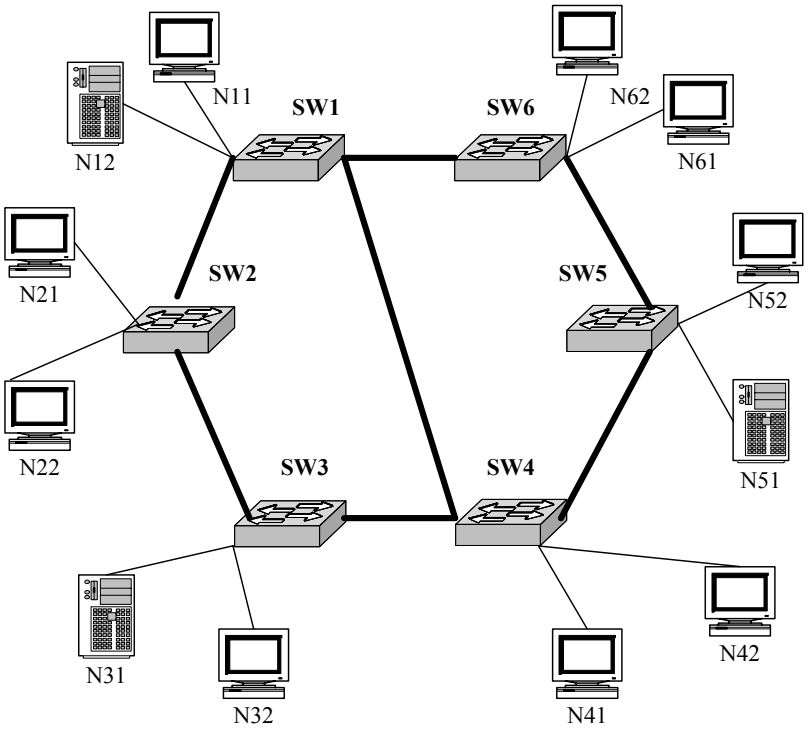
ในการจำลองแบบต้องการที่จะทำการตรวจสอบเฉพาะสถานะของสวิตซ์ทั้ง 6 สวิตซ์เท่านั้น โดยกำหนดเงื่อนไขว่าให้เกิดการขัดข้องได้ครั้งละหนึ่งสวิตซ์รวมทั้งกำหนดว่าเอ็นโหนดและลิงค์ทำงานเป็นปกติตลอดเวลาดังนั้นจึงสามารถกำหนดค่าพารามิเตอร์ต่างๆ ได้ดังนี้

3.2.1 เซตของสถานะที่เป็นไปได้ของโครงข่ายขนาดเล็ก

สถานะของสิ่งแวดล้อมซึ่งในที่นี้คือสถานะของระบบโครงข่าย ซึ่งโดยปกติแล้วเราต้องการให้ระบบโครงข่ายสามารถให้บริการได้ตลอดเวลา ดังนั้นอุปกรณ์ทุกอุปกรณ์จึงต้องสามารถให้บริการได้อย่างเป็นปกติ และหากอุปกรณ์นั้นขัดข้องจะถือได้ว่าอุปกรณ์นั้นอยู่ในสถานะที่ไม่ปกติ (abnormal) ดังนั้นสถานะที่เป็นไปได้ทั้งหมดของโครงข่ายที่มีโครงรูปโครงข่ายดังรูปที่ 3.1 จึงประกอบด้วยเซตของสถานะทั้งหมดที่เป็นไปได้หรือปริภูมิสเตต (state space) ทั้งหมดคือสถานะที่สวิตซ์ทั้งหมดทำงานเป็นปกติซึ่งในที่นี้แทนด้วย s_0 สถานะที่สวิตซ์ SW1 ขัดข้องแทนด้วย s_1 สถานะที่สวิตซ์ SW2 ขัดข้องแทนด้วย s_2 สถานะที่สวิตซ์ SW3 ขัดข้องแทนด้วย s_3 จนถึงสถานะที่สวิตซ์ SW6 ขัดข้องและแทนด้วยสถานะ s_6 ดังนั้นจึงสามารถเขียนสถานะทั้งหมดที่เป็นไปได้ว่า $S = \{s_0, s_1, s_2, ...s_N\}$ เมื่อ s_0 หมายถึงสถานะซึ่งสวิตซ์ทั้งหมดอยู่ในสภาวะปกติและ s_i หมายถึงสถานะซึ่งสวิตซ์ตัวที่ i ขัดข้อง

3.2.2 เซตของการกระทำที่เป็นไปได้ของโครงข่ายขนาดเล็ก

ในการเฝ้าตรวจสถานะของโครงข่าย เพื่อทำหน้าที่ในการตรวจสอบสถานะและค้นหาอุปกรณ์ที่ขัดข้อง โดยหลังจากที่พบว่าโครงข่ายอยู่ในสถานะใดสถานะหนึ่งจะต้องมีการเลือกการกระทำเพื่อที่จะจัดการกับสถานะที่ตรวจพบนั้น ซึ่งจะประกอบไปด้วยกรณีที่ตรวจพบว่าโครงข่ายอยู่ในสถานะปกติจะไม่เลือกการกระทำใดเพียงแจ้งว่าอยู่ในสถานะปกติ แต่หากพบว่ามีอุปกรณ์ขัดข้องเกิดขึ้นแต่ยังไม่ชัดเจนนัก จะต้องดำเนินการโพลล์ (polling)ไปที่เอ็นโหนด (end node) ที่เชื่อมต่ออยู่ที่สวิตซ์นั้นเพื่อยืนยันสถานะที่แท้จริง ซึ่งหากสวิตซ์นั้นขัดข้องจริงจะทำให้เอ็นโหนดนั้นขัดข้องด้วย แต่หากสวิตซ์นั้นปกติเอ็นโหนดนั้นจะปกติด้วยเช่นกัน หรือหากพบว่ามีอุปกรณ์ขัดข้องจริงจะต้องเลือกทำการเปลี่ยนหรือซ่อม (repair) สวิตซ์นั้นเพื่อให้ระบบโครงข่ายสามารถให้บริการได้ตามปกติ ดังนั้นเซตของการตัดสินใจที่เป็นไปได้ทั้งหมดของระบบ (action space) คือ $A = A^r \cup A^p$ เมื่อ $A^r = \{a_1^r, a_2^r, ...a_N^r\}$ และ $A^p = \{a_1^p, a_2^p, ...a_K^p\}$ เมื่อ a_i^r หมายถึงการเลือกที่จะทำการซ่อมสวิตซ์ ที่ i ส่วน a_j^p หมายถึงทำการโพลล์ไปที่เอ็นโหนดที่ j



รูปที่ 3.1 โครงรูปโครงข่ายที่ใช้ในการจำลองแบบโครงข่ายขนาดเล็ก

3.2.3 เซตของสิ่งที่ได้จากการสังเกตของโครงข่ายขนาดเล็ก

การทำงานของการเฝ้าตรวจสถานะของโครงข่าย ที่ต้องทำงานภายใต้สภาวะที่สิ่งที่ได้จากการสังเกตไม่ชัดเจน (partially observable) ซึ่งอาจเกิดจากข้อมูลสูญหาย (packet loss) หรืออาจเกิดจากความผิดพลาดของการ รับ-ส่ง ข้อมูล (error) จึงทำให้การทำงานของการเฝ้าตรวจสถานะของโครงข่ายจะทราบเพียงสถานะที่ได้จากการสังเกตเท่านั้น (He, 2003) ดังนั้นหากทำการโพลล์ไปที่เอ็นโหนดหนึ่งๆ แล้วพบว่าสวิตซ์ที่เอ็นโหนดนั้นเชื่อมต่ออยู่เสมือนอยู่ในสถานะที่ปกติหรือสถานะที่ผิดปกติตามลำดับโดยในความเป็นจริงแล้ว ข้อมูลที่ได้จากการโพลล์ดังกล่าวอาจให้ข้อมูลที่ไม่ถูกต้องก็เป็นได้ จึงสามารถเขียนเซตของเหตุการณ์ที่สังเกตได้ทั้งหมด (observation space) ได้ว่า $z = (normal, abnormal)$ โดย normal หมายถึงเหตุการณ์ที่สังเกตได้หลังจากทำการโพลล์ไปที่เอ็นโหนดหนึ่งๆ แล้วพบว่าสวิตซ์ที่เอ็นโหนดนั้นๆ เชื่อมต่ออยู่เสมือนอยู่ในสถานะปกติ และ abnormal หมายถึงเหตุการณ์ที่สังเกตได้หลังจากทำการโพลล์ไปที่เอ็นโหนดหนึ่งๆ แล้วพบว่าสวิตซ์ที่เอ็นโหนดนั้นเชื่อมต่ออยู่เสมือนอยู่ในสถานะผิดปกติ ซึ่งในความเป็นจริงแล้วข้อมูลจากการโพลล์ดังกล่าวอาจให้ข้อมูลที่ไม่ถูกต้องก็เป็นได้ จึงใช้ระดับของความน่าจะเป็นแทนความถูกต้องของข้อมูลนี้ ซึ่งเรียกว่าความน่าจะเป็นของการเกิดสัญญาณเตือนที่ผิดพลาด (false alarm) ที่มีค่าอยู่ระหว่าง 0.0 ถึง 1.0

3.2.4 โครงสร้างการกำหนดผลรางวัลของโครงข่ายขนาดเล็ก

วิธีรีอินฟอร์สเมนท์เลิร์นนิ่งจะใช้วิธีการเรียนรู้โดยพิจารณาจากผลรางวัลที่ได้รับ ดังนั้นจึงต้องทำการกำหนดผลรางวัลที่จะได้รับ เมื่อตัวกระทำการตัดสินใจทำการตัดสินใจเลือกการกระทำต่างๆ โดยมีหลักในการกำหนดผลรางวัลคือ เมื่อสามารถค้นหาจุดที่ขัดข้องพบและเลือกการกระทำที่จะทำการซ่อมอุปกรณ์ที่ขัดข้องนั้นจะถือว่าเป็นการเลือกการกระทำที่ถูกต้อง จะกำหนดค่าผลรางวัลที่มีค่าสูงสุดเท่ากับ c_1 หรือเมื่อสามารถค้นหาจุดที่ขัดข้องพบแต่ตัวกระทำการตัดสินใจยังไม่มั่นใจว่าเป็นตำแหน่งที่ถูกต้อง จึงตัดสินใจที่จะทำการโพลล์ไปที่เอ็นโหนดที่เชื่อมต่ออยู่ที่สวิตซ์นั้น จะกำหนดค่าผลรางวัลรองลงมามีค่าเท่ากับ c_2 แต่หากตัดสินใจที่จะทำการโพลล์ไปที่เอ็นโหนดอื่นๆ ที่ไม่ได้เชื่อมต่ออยู่ที่สวิตซ์นั้นซึ่งไม่ช่วยในการยืนยันสถานะของสวิตซ์ที่กำลังสังเกตอยู่จะถือว่าเป็นการกระทำที่ไม่ก่อให้เกิดประโยชน์ จึงกำหนดค่าผลรางวัลให้มีค่าต่ำรองลงมามีค่าเท่ากับ c_3 และหากมีการตัดสินใจที่ผิดพลาด เช่น ทำการตัดสินใจที่เลือกกระทำการซ่อมสวิตซ์ที่ไม่ขัดข้องที่ก่อให้เกิดความสูญเสียขึ้นจึงต้องทำการลงโทษในการตัดสินใจที่ผิดพลาดนี้ ด้วยการกำหนดให้ผลรางวัลมีค่าต่ำที่สุดให้มีค่าเท่ากับ c_4 ดังนั้นจะได้ความสัมพันธ์ของผลรางวัลทั้งหมดคือ $c_1 \gg c_2 > c_3 \gg c_4$

3.3 การทดลองการจำลองแบบของโครงข่ายขนาดเล็ก

การจำลองแบบกระทำเพื่อเป็นการทดสอบสมมุติฐานในการนำวิธีรีอินฟอร์สเมนท์เลิร์นนิ่งแบบออนโพลิซีมอนติคาร์โล มาใช้กับการบริหารจัดการโครงข่ายโดยใช้วิธีรีอินฟอร์สเมนท์เลิร์นนิ่งแบบออนโพลิซีมอนติคาร์โล เปรียบเทียบกับวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ (He, 2003) ทำการจำลองแบบโดยใช้โปรแกรมคอมพิวเตอร์ภาษาซี ในการจำลองแบบได้กำหนดให้สิ่งที่ได้จากการสังเกตคือสถานะของอุปกรณ์ในโครงข่าย และตัวกระทำการตัดสินใจคือตัวเฝ้าตรวจสถานะของโครงข่าย โดยทำการทดลองเริ่มจากทำการเลือกสถานะโดยการสุ่มแบบเสมอภาค (uniform) จากสถานะทั้งหมดที่เป็นไปได้เพื่อจำลองการเกิดข้อขัดข้องของสวิตซ์ และให้ตัวกระทำการตัดสินใจทำการตัดสินใจในการเลือกการกระทำที่เหมาะสมกับสถานะนั้น เพื่อสร้างการเรียนรู้ให้กับระบบ โดยกำหนดความน่าจะเป็นในการเกิดสัญญาณเตือนที่ผิดพลาด (false alarm) ซึ่งใช้แทนความไม่ชัดเจนของสิ่งที่ได้จากการสังเกตมีค่าเปลี่ยนแปลงเพิ่มขึ้นจาก 0.0 ถึง 1.0 เช่น ที่ความน่าเป็นของการเกิดสัญญาณเตือนที่ผิดพลาดเท่ากับ 0.2 จะหมายถึงสถานะที่สังเกตได้มีความเป็นไปได้ที่สัญญาณเตือนจะถูกต้องเท่ากับ 0.8 และมีความเป็นไปได้ที่จะเป็นสัญญาณเตือนที่ผิดพลาดเท่ากับ 0.2

ดังนั้นในการจำลองแบบด้วยโครงข่ายขนาดเล็กจึงประกอบด้วย สถานะของโครงข่ายทั้งหมด 7 สถานะมีการกระทำทั้งหมด 19 การกระทำ โดยทำการกำหนดผลรางวัลดังนี้คือ

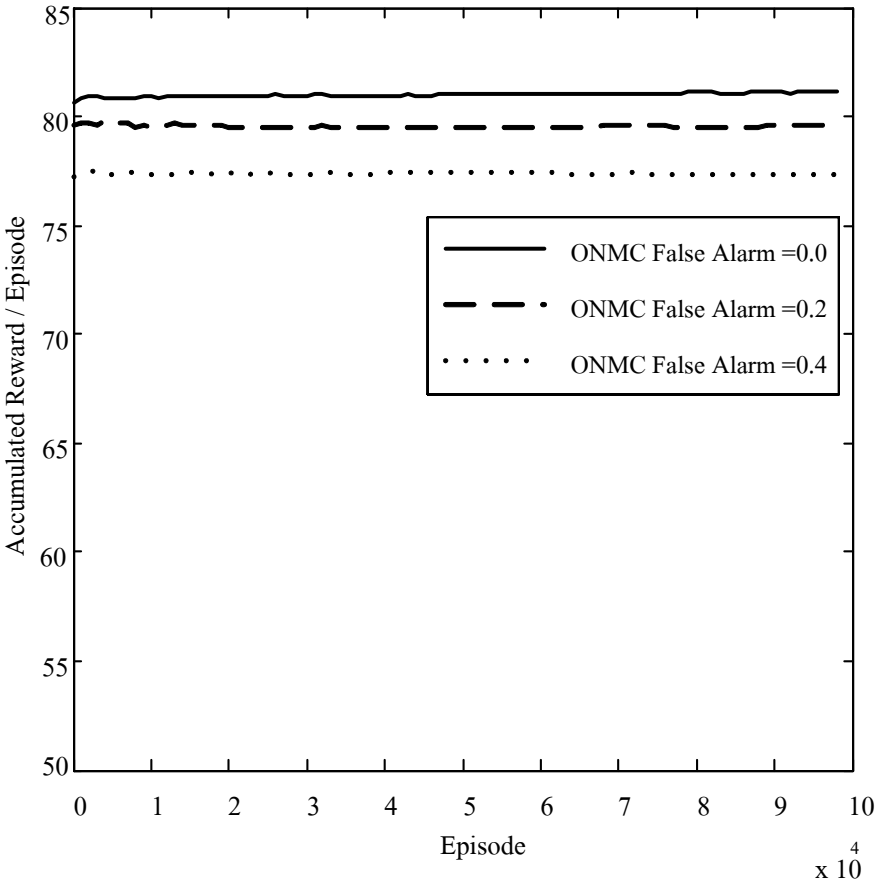
$c_1 = 100$, $c_2 = 1$, $c_3 = (-1)$, $c_4 = (-100)$ ซึ่ง $c_1 \gg c_2 > c_3 \gg c_4$ และสถานะของอุปกรณ์ที่สังเกตได้ประกอบด้วย 2 สถานะแทนด้วย $Z = \{0,1\}$ โดยกำหนดความน่าจะเป็นในการเกิดสัญญาณเตือนที่ผิดพลาด (false alarm) มีค่าเปลี่ยนแปลงเพิ่มขึ้นจาก 0.0 ถึง 1.0

3.3.1 การทดลองการจำลองแบบด้วยวิธีออนโพลิซีมอนติคาร์โล

การจำลองแบบด้วยวิธีออนโพลิซีมอนติคาร์โลโดยมีขั้นตอนดังหัวข้อที่ 2.5 และกำหนดให้ “1” แทนกรณีเกิดการขัดข้องที่สวิตซ์และ “0” แทนกรณีสวิตซ์อยู่ในสถานะปกติ ถ้า s แทนสถานะของโครงข่ายจะได้ว่า $S = \{s_0, s_1, s_2, ...s_N\}$ และ $s \in S$ กำหนดให้ s_o หมายถึงระบบอยู่ในสถานะปกติ s_1 จะหมายถึงสวิตซ์ SW1 ขัดข้องหรือ s_2 จะหมายถึงสวิตซ์ SW2 ขัดข้อง การกระทำของการเฝ้าตรวจสถานะของโครงข่ายจะประกอบไปด้วยการซ่อมสวิตซ์แต่ละสวิตซ์และการโพลล์ไปที่เอ็นโหนด เมื่อ a แทนการกระทำจะได้การกระทำทั้งหมดที่เป็นไปได้คือ คือ $A = A^r \cup A^p$ เมื่อ $A^r = \{a_1^r, a_2^r, ...a_N^r\}$ และ $A^p = \{a_0^p, a_1^p, a_2^p, ...a_K^p\}$ และ $a \in A$ เช่น a_1^r หมายถึงการเลือกการกระทำที่จะทำการซ่อมสวิตซ์ SW1 หรือ a_1^p หมายถึงการโพลล์ไปที่เอ็นโนดที่ N11 เพื่อทำการตรวจสอบสถานะของ SW1 และหากระบบอยู่ในสถานะปกติการกระทำที่แจ้งว่าระบบปกติ คือ a_0^p และเซตของการสังเกตที่ได้จะประกอบไปด้วย สถานะปกติและสถานะไม่ปกติแทนด้วย $Z = \{0,1\}$ โดยได้ทำการจำลองแบบจำนวน 50,000 ครั้งต่อค่าความน่าจะเป็นของการเกิดสัญญาณเตือนที่ผิดพลาดหนึ่งค่า

การจำลองแบบได้แบ่งเป็นสองระยะคือ ระยะแรกเป็นการสร้างการเรียนรู้ให้กับตัวกระทำการตัดสินใจเพื่อทำการปรับปรุงกฎควบคุมให้ได้กฎควบคุมที่ดีและในระยะที่สองจะเป็นการจำลองการใช้กับสถานการณ์จริง และนำกฎควบคุมที่ได้รับการปรับปรุงจากระยะแรกมาใช้เป็นกฎควบคุมในระยะที่สอง โดยพิจารณาผลของการตัดสินใจในรูปของผลรางวัลสะสมต่อเอพพิโซด ซึ่งแสดงผลรางวัลที่ได้รับหลังจากการตัดสินใจเลือกการกระทำดังรูปที่ 3.2 จะแสดงให้เห็นถึงผลรางวัลสะสมต่อเอพพิโซดที่เกิดขึ้นจากการจำลองแบบ ที่ค่าความน่าจะเป็นของการเกิดสัญญาณเตือนที่ผิดพลาดมีค่าเท่ากับ 0.0 ซึ่งหมายถึงข้อมูลที่สังเกตได้เป็นข้อมูลที่ถูกต้องทั้งหมดและที่ความน่าจะเป็นของการเกิดสัญญาณเตือนที่ผิดพลาดมีค่าเท่ากับ 0.2 ซึ่งหมายถึงข้อมูลที่สังเกตได้เป็นข้อมูลที่ถูกต้อง 0.8 และไม่ถูกต้อง 0.2 และที่ความน่าจะเป็นของการเกิดสัญญาณเตือนที่ผิดพลาดมีค่าเท่ากับ 0.4 ซึ่งพบว่าผลรางวัลสะสมต่อเอพพิโซดในกรณีที่ความน่าจะเป็นของการเกิดสัญญาณเตือนที่ผิดพลาดมีค่าเท่ากับ 0.0 จะสูงกว่าผลรางวัลสะสมต่อเอพพิโซดในกรณีที่ความน่าจะเป็นของการเกิดสัญญาณเตือนที่ผิดพลาดมีค่าเท่ากับ 0.2 และ 0.4 ซึ่งแสดงให้เห็นว่าวิธีออนโพลิซีมอนติคาร์โลสามารถตัดสินใจได้ถูกต้องจึงส่งผลให้ผลรางวัลสะสมต่อเอพพิโซดมีค่าสูง และเป็นไปในแนวทางที่ถูกต้องที่ประสิทธิภาพในการตัดสินใจจะลดลงเมื่อความน่าจะเป็นของการเกิดสัญญาณ

เตือนที่ผิดพลาดมีค่าเพิ่มมากขึ้น หรือข้อมูลที่ได้จากการสังเกตเป็นข้อมูลที่มีความผิดพลาดเพิ่มมากขึ้นนั่นเอง



รูปที่ 3.2 ผลรางวัลสะสมต่อเอพิโซด ของวิธีออนโพลิซีมอนติคาร์โลในโครงข่ายขนาดเล็ก

3.3.2 การทดลองการจำลองแบบด้วยวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์

วิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์เป็นขั้นตอนวิธีที่นำเสนอโดย (He, 2003) ซึ่งมีลำดับขั้นการทำงานของขั้นตอนวิธีตามหัวข้อ 2.6 ด้วยการนำเสนอการระบุสถานะของระบบในกรณีสิ่งที่ได้จากการสังเกตไม่ชัดเจนด้วยการใช้ความน่าจะเป็นแทนสถานะ ที่เรียกว่าบีลีฟสเตต (belief state) เช่น $\mathbf{b} = [0,1,0,0,0,0,0]$ หมายความถึงการเฝ้าตรวจสถานะของโครงข่ายมีความเชื่อว่าสวิตซ์ SW1 ขัดข้องจริง $\mathbf{b} = [0,0,0,0,0,0,1]$ หมายความถึงการเฝ้าตรวจสถานะของโครงข่ายมีความเชื่อว่า สวิตซ์ SW6 ขัดข้องจริง $\mathbf{b} = [0.1,0.5,0.1,0.1,0.1,0.1,0.0]$ หมายความถึง การเฝ้าตรวจสถานะของโครงข่ายมีความมั่นใจด้วยความน่าจะเป็น 0.1 ว่าไม่มีสวิตซ์ใดขัดข้อง 0.5 ว่า SW1 ขัดข้อง 0.1 ว่า SW2, SW3, SW4, SW5 ขัดข้อง และ 0.0 ว่า SW6 ขัดข้อง หรือ $\mathbf{b} = [0.1,0.05,0.05,0.1,0.1,0.1,0.5]$

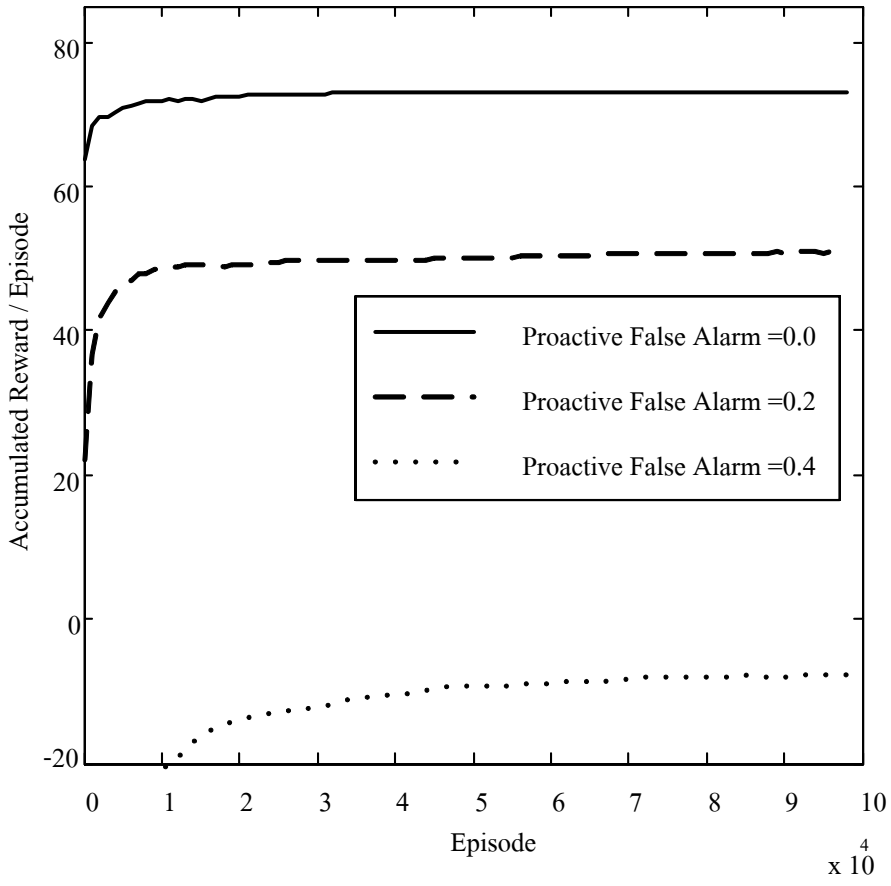
หมายความถึงการเฝ้าตรวจสถานะของโครงข่ายมีความเชื่อว่าสวิตซ์ SW6 ขัดข้องด้วยความน่าจะเป็น 0.5 และเชื่อว่าระบบอยู่ในสถานะอื่นๆ ด้วยความน่าจะเป็นลดหลั่นกันลงไป โดยในที่นี้ได้ทำการจำลองแบบโดยใช้บีลีฟสเตต 0.5 (proactive belief state 0.5) ซึ่งหมายถึงหากบีลีฟสเตตของสถานะที่สังเกตได้มีค่าตั้งแต่ 0.5 เป็นต้นไปจะตัดสินใจว่าโครงข่ายอยู่ในสถานะนั้นและบีลีฟสเตต 1.0 (proactive belief state 1.0) ซึ่งหมายถึงระบบจะทำการตัดสินใจว่าโครงข่ายอยู่ในสถานะนั้นก็ต่อเมื่อบีลีฟสเตตของสถานะที่สังเกตได้มีค่าเท่ากับ 1.0 เท่านั้น โดยได้ทำการจำลองแบบจำนวน 50,000 ครั้ง ต่อค่าความน่าจะเป็นของการเกิดสัญญาณเตือนที่ผิดพลาดหนึ่งค่า

การจำลองแบบได้แบ่งเป็นสองระยะคือ ระยะแรกเป็นการสร้างการเรียนรู้ให้กับตัวกระทำการตัดสินใจเพื่อทำการปรับปรุงกฎควบคุมให้ได้กฎควบคุมที่ดีและในระยะที่สองจะเป็นการจำลองการใช้กับสถานการณ์จริง โดยนำกฎควบคุมที่ได้รับการปรับปรุงจากระยะแรกมาใช้เป็นกฎควบคุมในระยะที่สองโดยพิจารณาผลของการตัดสินใจในรูปของผลรางวัลสะสมต่อเอพพิโซด ซึ่งแสดงผลรางวัลดังรูปที่ 3.3 แสดงให้เห็นถึงผลรางวัลสะสมต่อเอพพิโซดที่เกิดขึ้นจากการจำลองแบบที่ค่าความน่าจะเป็นของการเกิดสัญญาณเตือนที่ผิดพลาดมีค่าเท่ากับ 0.0 ซึ่งหมายถึงข้อมูลที่สังเกตได้เป็นข้อมูลที่ถูกต้องทั้งหมด และที่ความน่าจะเป็นของการเกิดสัญญาณเตือนที่ผิดพลาดมีค่าเท่ากับ 0.2 ซึ่งหมายถึงข้อมูลที่สังเกตได้เป็นข้อมูลที่ถูกต้อง 0.8 และไม่ถูกต้อง 0.2 และที่ความน่าจะเป็นของการเกิดสัญญาณเตือนที่ผิดพลาดมีค่าเท่ากับ 0.4 ซึ่งพบว่าผลรางวัลสะสมต่อเอพพิโซดในกรณีที่ ความน่าจะเป็นของการเกิดสัญญาณเตือนที่ผิดพลาดมีค่าเท่ากับ 0.0 จะสูงกว่าผลรางวัลสะสมต่อเอพพิโซดในกรณีที่ความน่าจะเป็นของการเกิดสัญญาณเตือนที่ผิดพลาดมีค่าเท่ากับ 0.2 และ 0.4 ซึ่งจะหมายถึงวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ สามารถตัดสินใจในกรณีที่มีข้อมูลที่ชัดเจนได้ดีกว่า แต่กลับพบว่าความสามารถในการตัดสินใจกลับลดลงเป็นอย่างมากเมื่อความไม่ชัดเจนของข้อมูลมีค่าสูงขึ้น

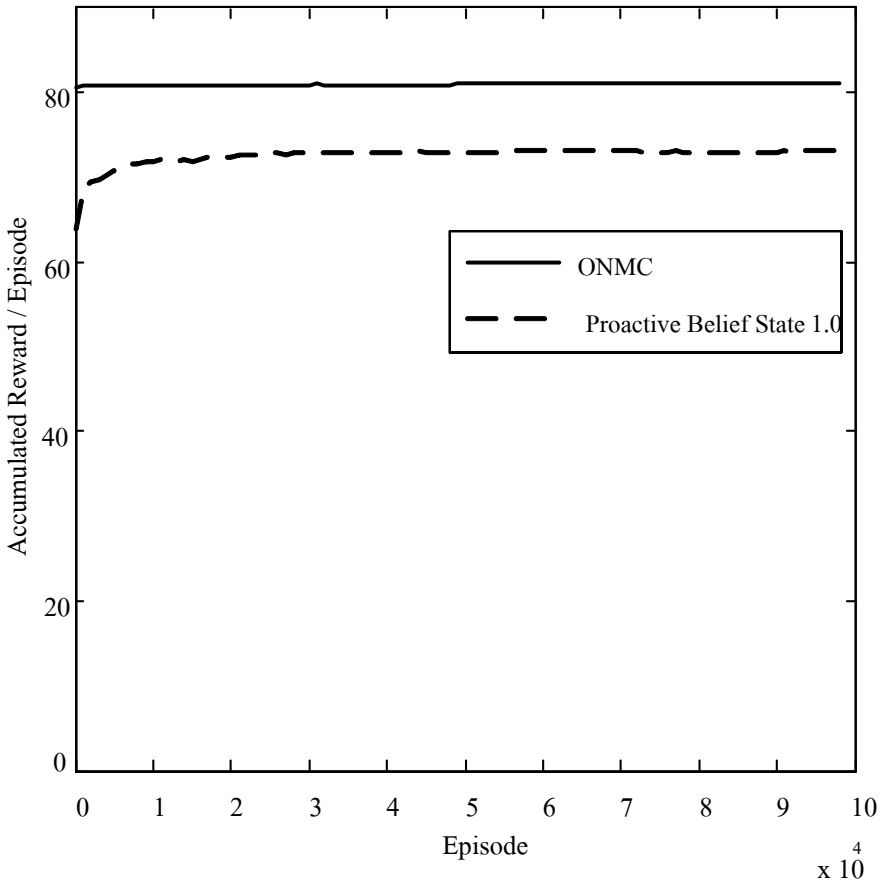
3.4 ผลการจำลองของแบบของโครงข่ายขนาดเล็ก

จากการจำลองแบบด้วยวิธีออนโพลิซีมอนติคาร์โลเปรียบเทียบกับวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ สามารถพิจารณาผลรางวัลสะสมต่อเอพพิโซด ซึ่งหากวิธีการใดให้ผลรางวัลสะสมต่อเอพพิโซดที่สูงกว่า จะหมายถึงวิธีการนั้น สามารถที่จะเลือกการกระทำที่เหมาะสมกับสถานะของสิ่งแวดล้อมได้ดีกว่า โดยสอดคล้องกับเงื่อนไขของการให้ผลรางวัลที่กำหนดว่าหากเป็นการตัดสินใจในการเลือกการกระทำที่ถูกต้องจะได้ผลรางวัลที่สูงที่สุด และหากเป็นการตัดสินใจที่ไม่ถูกต้องหรือก่อให้เกิดผลเสีย เช่น การสิ้นเปลืองทรัพย์ยากรหรือสิ้นเปลืองค่าใช้จ่ายจะ

กำหนดผลรางวัลในระดับที่ต่ำ ดังนั้นจึงทำการเปรียบเทียบผลรางวัลสะสมต่อเอพพิโซดระหว่างวิธีออนโพลิซีมอนติคาร์โลเปรียบเทียบกับวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ (โปรแอคทีฟ บีลีฟสเตต 1.0) ที่ได้จากการจำลองการใช้งานจริงโดยการนำวิธีการทั้งสองแบบไปสร้างการเรียนรู้เพื่อให้สามารถทำการตัดสินใจได้อย่างถูกต้อง โดยทำการจำลองแบบที่ความน่าจะเป็นของการเกิดสัญญาณเตือนที่ผิดพลาดมีค่าเท่ากับ 0.0 และทำการทดลองทั้งหมด 20 ครั้งแล้วจึงนำค่าเฉลี่ยมาทำการเปรียบเทียบ โดยในแต่ละครั้งได้ทำการทดลองถึง 100,000 เอพพิโซดเพื่อติดตามผลรางวัลเฉลี่ยที่เกิดจากการตัดสินใจในระยะยาว ซึ่งพบว่าวิธีออนโพลิซีมอนติคาร์โลจะให้ผลรางวัลสะสมต่อเอพพิโซดที่สูงกว่าโดยตลอดดังรูปที่ 3.4 เป็นการเปรียบเทียบผลรางวัลสะสมต่อเอพพิโซดของทั้งสองวิธี ซึ่งหมายถึงตัวกระทำการตัดสินใจของวิธีออนโพลิซีมอนติคาร์โลสามารถที่จะทำการตัดสินใจในการเลือกการกระทำที่ถูกต้องได้มากกว่า จึงส่งผลให้ผลรางวัลสะสมต่อเอพพิโซดมีค่าที่สูงกว่านอกจากนั้นยังคงมีแน้วโน้มที่จะให้ผลรางวัลเฉลี่ยในระยะยาวที่มีค่าสูงและอยู่ในระดับที่คงที่โดยตลอดซึ่งตรงกับที่เราต้องการผลรางวัลเฉลี่ยในระยะยาวที่มีค่าสูง

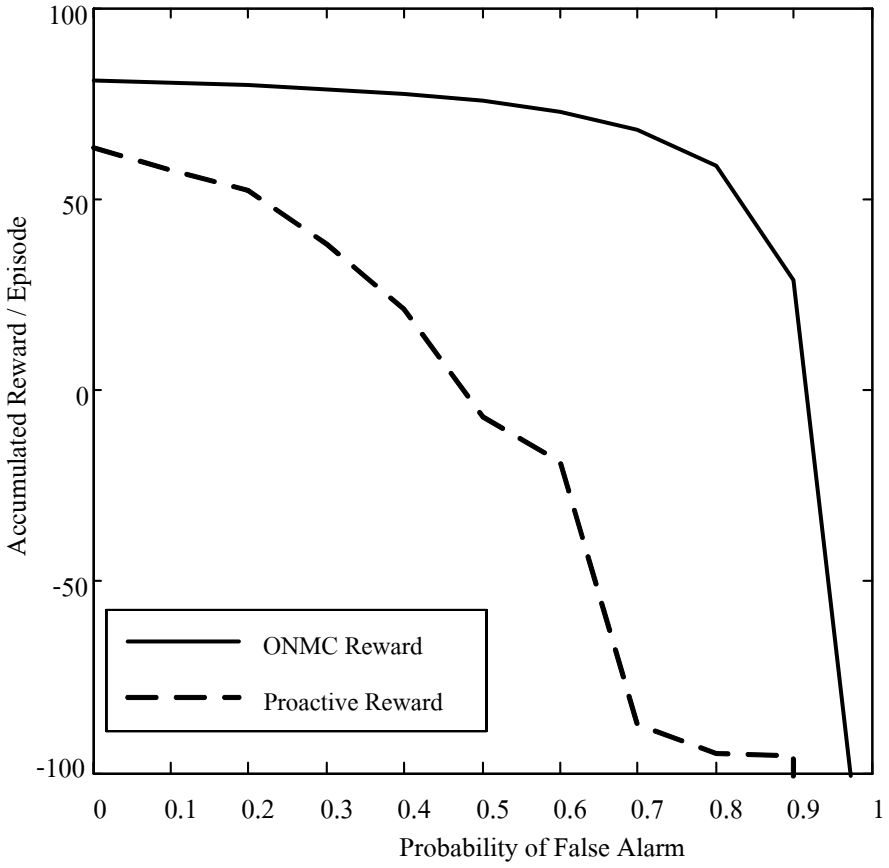


รูปที่ 3.3 ผลรางวัลสะสมต่อเอพพิโซดของวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์



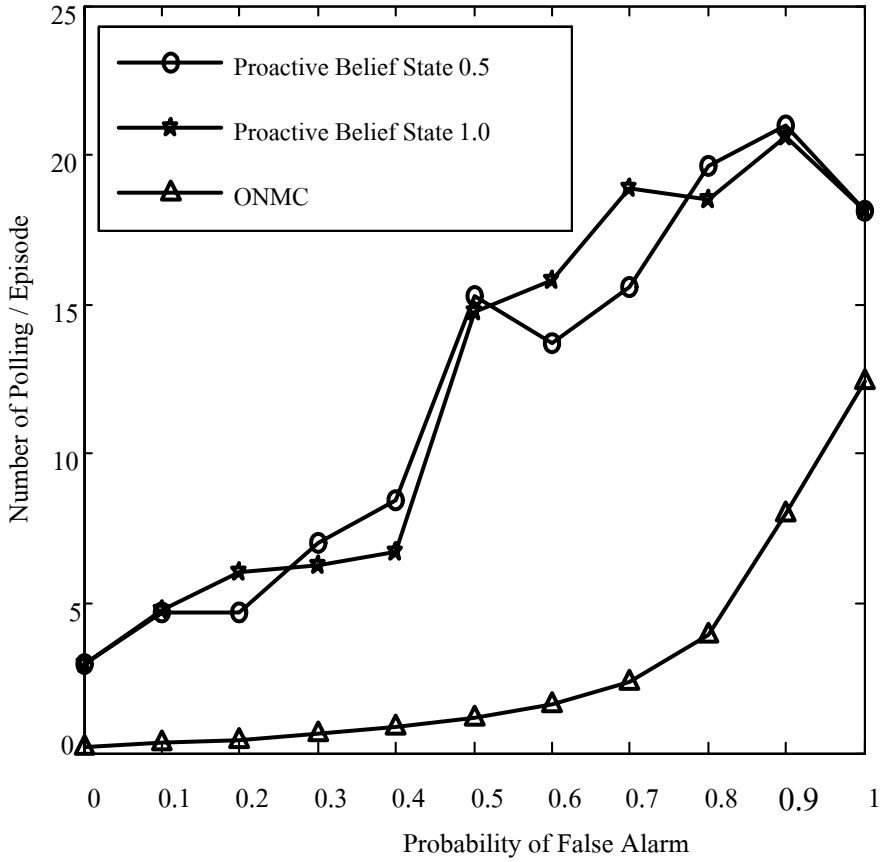
รูปที่ 3.4 ผลรางวัลสะสมต่อเอพพิโซดของวิธีออนโพลิซีมอนติคาร์โลเปรียบเทียบกับวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์

เมื่อพิจารณาผลการตัดสินใจของขั้นตอนวิธีทั้งสองแบบ ในกรณีที่สิ่งที่ได้จากการสังเกตมีความไม่ชัดเจนเพิ่มขึ้นหรือข้อมูลที่สังเกตได้มีโอกาสที่จะไม่ถูกต้องเพิ่มมากขึ้นโดยได้ผลการจำลองแบบดังรูปที่ 3.5 ซึ่งเป็นการเปรียบเทียบผลรางวัลสะสมต่อเอพพิโซดที่แทนด้วยแกนวาย (y) และความไม่ชัดเจนของสิ่งที่ได้จากการสังเกต ที่แทนด้วยความน่าจะเป็นของการเกิดสัญญาณเตือนที่ผิดพลาดในแกนเอ็กซ์ (x) โดยมีค่าเริ่มต้นที่ 0.0 และเพิ่มขึ้นจนถึง 1.0 ระหว่างวิธีออนโพลิซีมอนติคาร์โลและวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ โดยพบว่าวิธีออนโพลิซีมอนติคาร์โลจะมีผลรางวัลสะสมต่อเอพพิโซดที่สูงกว่าโดยตลอด จึงสรุปได้ว่าตัวกระทำการตัดสินใจของวิธีออนโพลิซีมอนติคาร์โลสามารถที่จะทำการตัดสินใจในการเลือกการกระทำที่ถูกต้องได้มากกว่า แม้ว่าสิ่งที่ได้จากการสังเกตจะมีความไม่ชัดเจนเพิ่มสูงขึ้นก็ตาม จึงส่งผลให้ผลรางวัลสะสมต่อเอพพิโซดของวิธีออนโพลิซีมอนติคาร์โลมีค่าที่สูงกว่าในทุกๆ ค่าของความน่าจะเป็นของการเกิดสัญญาณเตือนที่ผิดพลาด



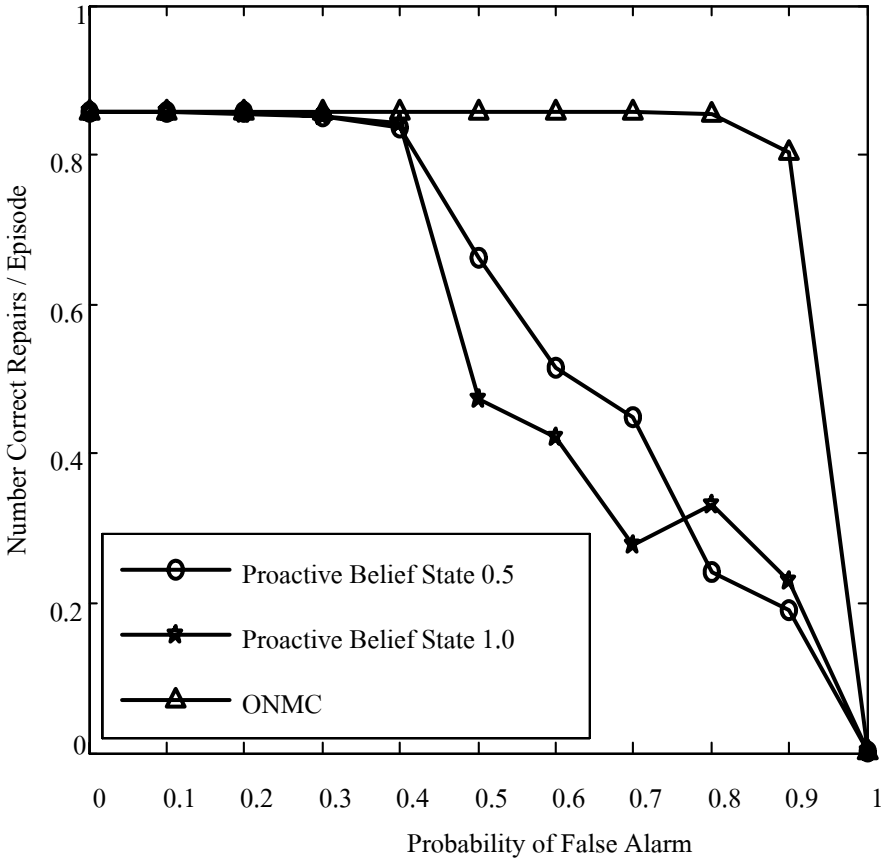
รูปที่ 3.5 ผลรางวัลสะสมต่อเอพพิโซดต่อความน่าจะเป็นของการเกิดสัญญาณเตือนที่ผิดพลาด

ในงานวิจัยนี้มีจุดประสงค์ที่จะลดโพลล์ลิ่งโอเวอร์เฮดที่เกิดขึ้นเนื่องจากการทำงานของการเฝ้าตรวจสถานะของโครงข่ายจึงได้ทำการวิจัย โดยทำการเปรียบเทียบปริมาณของการโพลล์ที่เกิดขึ้นระหว่างวิธีออนโพลิซีมอนติคาร์โลและวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ ซึ่งปริมาณของการโพลล์นี้จะเป็นตัวเปรียบเทียบที่ชัดเจน หากมีปริมาณของการโพลล์ที่สูงย่อมก่อให้เกิดโพลล์ลิ่งโอเวอร์เฮดที่สูงด้วยเช่นกัน โดยได้ทำการจำลองแบบที่ความน่าจะเป็นของการเกิดสัญญาณเตือนที่ผิดพลาดที่ค่าเริ่มต้นที่ 0.0 และเพิ่มขึ้นเรื่อยๆ จนถึง 1.0 ซึ่งพบว่าวิธีออนโพลิซีมอนติคาร์โลจะมีปริมาณการโพลล์ที่ต่ำกว่าในทุกๆ ค่าของการเกิดสัญญาณเตือนที่ผิดพลาดและสามารถลดปริมาณการโพลล์ลงได้ โดยสามารถลดปริมาณการโพลล์ระหว่าง 33 % ถึง 86 % ดังรูปที่ 3.6 ซึ่งการที่สามารถลดจำนวนการโพลล์ได้จะเป็นผลดีต่อระบบโครงข่ายเนื่องจากจะส่งผลให้ปริมาณโพลล์ลิ่งโอเวอร์เฮดที่เกิดขึ้นในโครงข่ายลดต่ำลง จึงเป็นการลดการสูญเสียแบนด์วิดท์ที่เกิดจากการเฝ้าตรวจสถานะของโครงข่ายลงได้



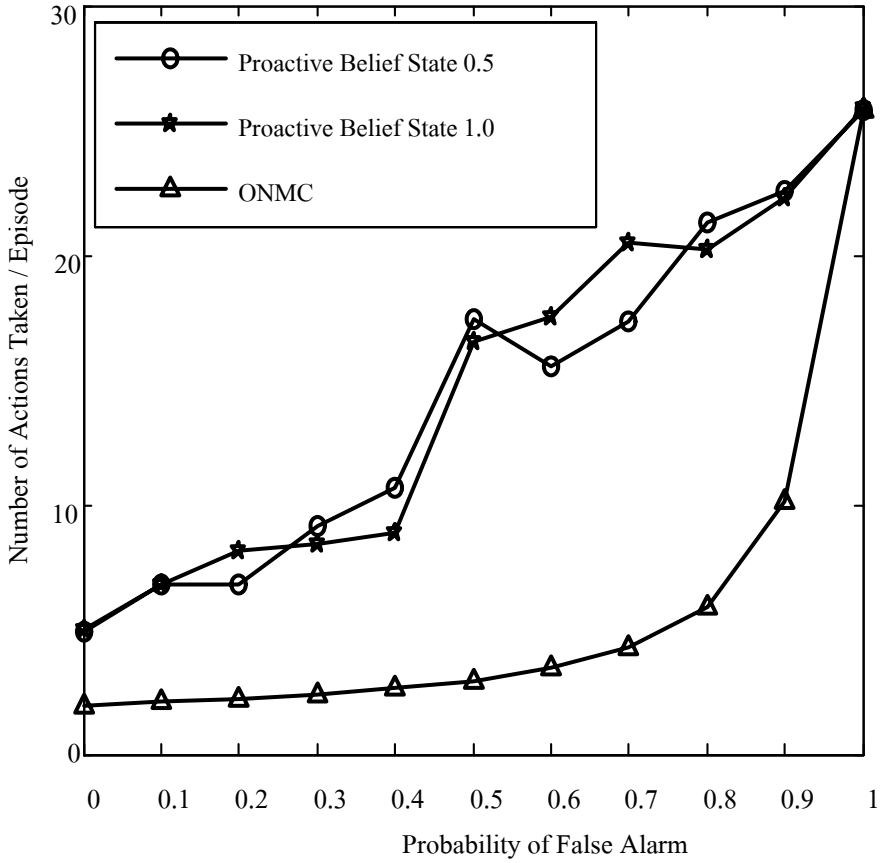
รูปที่ 3.6 ความสัมพันธ์ระหว่างจำนวนครั้งของการโพลล์ต่อเอพพิโซด

เมื่อพิจารณาในด้านของความถูกต้องในการแจ้งตำแหน่งที่อุปกรณ์ขัดข้อง ซึ่งเป็นค่าชี้วัดประสิทธิภาพของการเฝ้าตรวจสถานะของโครงข่ายในส่วนของการค้นหาตำแหน่งที่อุปกรณ์ขัดข้อง จึงได้ทำการเปรียบเทียบปริมาณการแจ้งตำแหน่งที่สวิตซ์ขัดข้องได้ถูกตำแหน่งต่อเอพพิโซดระหว่างวิธีออนโพลิซีมอนติคาร์โล และวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ ที่ค่าความน่าจะเป็นของการเกิดสัญญาณเตือนที่ผิดพลาดมีค่าระหว่าง 0.0 ถึง 1.0 ซึ่งพบว่า วิธีออนโพลิซีมอนติคาร์โล มีความสามารถ ในการระบุตำแหน่งของอุปกรณ์ที่ขัดข้อง ได้แม่นยำมากกว่าวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ถึงแม้ว่าที่ความน่าจะเป็นของการเกิดสัญญาณเตือนที่ผิดพลาดที่มีค่าระหว่าง 0.0 ถึง 0.4 จะมีค่าความถูกต้องที่ใกล้เคียงกัน แต่ที่ความน่าจะเป็นของการเกิดสัญญาณเตือนที่ผิดพลาดตั้งแต่ 0.4 เป็นต้นไปวิธีออนโพลิซีมอนติคาร์โลยังคงให้ความถูกต้องที่ดีกว่าอย่างชัดเจน โดยวิธีออนโพลิซีมอนติคาร์โลจะมีความถูกต้องมากกว่า วิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ 1.0 % ถึง 65 %ดังรูปที่ 3.7



รูปที่ 3.7 ความสัมพันธ์ระหว่างจำนวนครั้งที่ซ่อมได้ถูกต้องต่อเอพพิโซด

ทำการเปรียบเทียบให้เห็นถึงจำนวนการกระทำที่ต้องใช้ในการวิเคราะห์หาจุดขัดข้องในแต่ละเอพพิโซดเปรียบเทียบระหว่างวิธีออนโพลิซีมอนติคาร์โล กับวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ ภายใต้สมมุติฐานว่าการทำงานของการเฝ้าตรวจสถานะของโครงข่ายที่ดีควรมีโพลล์ลิ่งโอเวอร์เฮดต่ำทั้งยังต้องสามารถค้นหาจุดขัดข้องได้ด้วยความรวดเร็ว หรือมีจำนวนการกระทำที่ใช้ในการวิเคราะห์หาจุดขัดข้องต่ำจึงจะส่งผลให้มีโพลล์ลิ่งโอเวอร์เฮดต่ำด้วย จากผลการจำลองแบบดังรูปที่ 3.8 โดยทำการจำลองแบบที่ความน่าจะเป็นของการเกิดสัญญาณเตือนที่ผิดพลาด มีค่าตั้งแต่ 0.0 จนถึง 1.0 พบว่าการใช้วิธีออนโพลิซีมอนติคาร์โลสามารถลดจำนวนของการกระทำที่ใช้ในการค้นหาอุปกรณ์ขัดข้องในแต่ละครั้งลงได้ถึง 80 % ซึ่งตรงกับความต้องการที่จะลดโพลล์ลิ่งโอเวอร์เฮดของระบบโครงข่ายลง แม้ว่าที่ความน่าจะเป็นของการเกิดสัญญาณเตือนที่ผิดพลาด มีค่าเท่ากับ 1.0 ทั้งสามวิธีจะมีจำนวนการกระทำที่เท่ากันซึ่งเกิดจากการที่สิ่งที่ได้จากการสังเกตผิดพลาดทั้งหมดจึงไม่อาจที่จะนำมาใช้ในการตัดสินใจได้



รูปที่ 3.8 ความสัมพันธ์ระหว่างจำนวนครั้งของการกระทำต่อเอพพิโซด

3.5 วิเคราะห์ผลการจำลองของแบบจำลองของโครงข่ายขนาดเล็ก

จากผลการจำลองแบบโดยได้ทำการจำลองแบบด้วยวิธีออนโพลิซีมอนติคาร์โล เปรียบเทียบกับวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ โดยทำการเปรียบเทียบค่าชี้วัดต่างๆ ดังนี้

การเปรียบเทียบผลรางวัลสะสมต่อเอพพิโซด ในกรณีที่สิ่งที่ได้จากการสังเกตไม่ชัดเจนซึ่งแทนด้วยความน่าจะเป็นของการเกิดสัญญาณเตือนที่ผิดพลาด ที่ค่าเริ่มที่ 0.0 และเพิ่มขึ้นจนถึง 1.0 พบว่าวิธีออนโพลิซีมอนติคาร์โลจะมีผลรางวัลสะสมต่อเอพพิโซดที่สูงกว่า เนื่องจากตัวกระทำการตัดสินใจของวิธีออนโพลิซีมอนติคาร์โล สามารถที่จะทำการตัดสินใจในการเลือกการกระทำที่ถูกต้องได้มากกว่าจึงส่งผลให้ผลรางวัลสะสมต่อเอพพิโซดมีค่าที่สูงกว่า

การเปรียบเทียบปริมาณของการโพลล์ที่เกิดขึ้น ที่ความน่าจะเป็นของการเกิดสัญญาณเตือนที่ผิดพลาดที่ค่าเริ่มที่ 0.0 และเพิ่มขึ้นเรื่อยๆ จนถึง 1.0 ซึ่งปริมาณของการโพลล์นี้จะเป็นตัวเปรียบเทียบที่ชัดเจนที่แสดงถึงปริมาณโพลล์ลิ่งโอเวอร์เฮดที่เกิดขึ้น พบว่าวิธีออนโพลิซีมอนติคาร์โลจะมีปริมาณการโพลล์ที่ต่ำกว่าในทุกๆ ค่าของการเกิดสัญญาณเตือนที่ผิดพลาด

การเปรียบเทียบจำนวนการแจ้งตำแหน่งที่อุปกรณ์ขัดข้องได้ถูกตำแหน่งต่อเอพพิโซด ซึ่งเป็นค่าชี้วัดประสิทธิภาพของการเฝ้าตรวจสถานะของโครงข่าย ในส่วนของการค้นหาตำแหน่งที่อุปกรณ์ขัดข้อง ที่ค่าความน่าจะเป็นของการเกิดสัญญาณเตือนที่ผิดพลาดมีค่าระหว่าง 0.0 ถึง 1.0 พบว่าวิธีออนโพลิซีมอนติคาร์โล มีความสามารถในการระบุตำแหน่งของอุปกรณ์ที่ขัดข้องได้แม่นยำมากกว่าวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์

การเปรียบเทียบจำนวนการกระทำ ที่ต้องใช้ในการวิเคราะห์หาจุดขัดข้องในแต่ละเอพพิโซด ซึ่งจากผลการจำลองแบบ ที่ความน่าจะเป็นของการเกิดสัญญาณเตือนที่ผิดพลาดมีค่าตั้งแต่ 0.0 จนถึง 1.0 พบว่าการใช้วิธีออนโพลิซีมอนติคาร์โลสามารถลดปริมาณของการกระทำลงได้มากกว่าวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ นั่นคือสามารถลดโพลล์ลิ่งโอเวอร์เฮดลงได้

เมื่อพิจารณาปริมาณหน่วยความที่ใช้ในการเก็บข้อมูลในการทำงานของขั้นตอนวิธี ของวิธีออนโพลิซีมอนติคาร์โลเปรียบเทียบกับวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ โดยใช้ขั้นตอนวิธีตามหัวข้อ 2.5 และหัวข้อ 2.6 ตามลำดับ พบว่าขั้นตอนการทำงานของวิธีออนโพลิซีมอนติคาร์โลจะทำการเก็บแอคชั่นแวลู่ของทุกๆ คู่ของสิ่งที่ได้จากการสังเกตและการกระทำ (observation-action pair) ซึ่งในการจำลองแบบประกอบด้วยสิ่งที่ได้จากการสังเกตทั้งหมดเท่ากับ 7 สถานะและการกระทำทั้งหมดเท่ากับ 19 การกระทำ ดังนั้นจึงมีจำนวนพารามิเตอร์ทั้งหมดเท่ากับ 7 x 19 หรือเท่ากับ 133 พารามิเตอร์ในส่วนของวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์จะมีจำนวนพารามิเตอร์ที่ประกอบด้วยพารามิเตอร์ของ $Q(s,a), e(s,a), T(s,a,s'), O(s',a,z'), b(s)$ โดยในการจำลองแบบ s, s', b เท่ากับ 7 สถานะ a คือการกระทำทั้งหมดเท่ากับ 19 การกระทำและ z เท่ากับ 2 สถานะดังนั้นจึงมีจำนวนพารามิเตอร์ทั้งหมดเท่ากับ (7 x 19) +(7 x 19) + (7 x 19 x 7) + (7 x 19 x 2) +7 เท่ากับ 1,470 พารามิเตอร์

หากกำหนดว่าในหนึ่งพารามิเตอร์เป็นตัวแปรชนิดดับเบิล (double) ซึ่งต้องใช้หน่วยความจำในการเก็บข้อมูลขนาด 8 ไบต์ (byte) ดังนั้นในส่วนของขั้นตอนวิธีแบบออนโพลิซีมอนติคาร์โลจะใช้หน่วยความจำในการทำงานเท่ากับ 133 คูณ 8 ซึ่งเท่ากับ 1,064 ไบต์ ในส่วนของขั้นตอนวิธีแบบโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ จะใช้หน่วยความจำในการทำงานเท่ากับ 1,470 คูณ 8 ซึ่งเท่ากับ 11,760 ไบต์ ซึ่งเห็นได้ชัดเจนว่าในกรณีที่โครงข่ายขนาดเล็ก วิธีออนโพลิซีมอนติคาร์โลต้องการหน่วยความจำที่ใช้ในการเก็บข้อมูลน้อยกว่าวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์

3.6 สรุป

จากผลการจำลองแบบด้วยวิธีออนโพลิซีมอนติคาร์โล เปรียบเทียบกับวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ ในกรณีที่สิ่งที่ได้จากการสังเกตไม่ชัดเจนซึ่งแทนด้วยความน่าจะเป็นของการ

เกิดสัญญาณเตือนที่ผิดพลาดที่ค่าเริ่มต้นที่ 0.0 และเพิ่มขึ้นจนถึง 1.0 โดยทำการเปรียบเทียบค่าชี้วัดต่างๆ และได้ผลสรุปดังนี้

การเปรียบเทียบผลรางวัลสะสมต่อเอพพิโซด พบว่าวิธีออนโพลิซีมอนติคาร์โลมีผลรางวัลสะสมต่อเอพพิโซดที่สูงกว่าโดยตลอด

การเปรียบเทียบจำนวนของการโพลล์ที่เกิดขึ้น ซึ่งพบว่าวิธีออนโพลิซีมอนติคาร์โลจะมีจำนวนการโพลล์ที่ต่ำกว่าในทุกๆ ค่าของการเกิดสัญญาณเตือนที่ผิดพลาด และสามารถลดปริมาณการโพลล์ได้ระหว่าง 33 % ถึง 86 %

การเปรียบเทียบจำนวนการแจ้งตำแหน่งที่อุปกรณ์ขัดข้องได้ถูกตำแหน่งต่อเอพพิโซด ซึ่งพบว่าวิธีออนโพลิซีมอนติคาร์โล มีความสามารถในการระบุตำแหน่งของอุปกรณ์ที่ขัดข้องได้แม่นยำมากกว่าวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ได้ถึง 65 %

การเปรียบเทียบจำนวนการกระทำที่ต้องใช้ในการวิเคราะห์หาจุดขัดข้อง ในแต่ละเอพพิโซด พบว่าการใช้วิธีออนโพลิซีมอนติคาร์โล สามารถลดปริมาณของการกระทำลงได้มากกว่าวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ถึง 80 %

การเปรียบเทียบปริมาณหน่วยความที่ใช้ในการเก็บข้อมูล ในการทำงานของขั้นตอนวิธีแบบออนโพลิซีมอนติคาร์โลจะใช้หน่วยความจำในการทำงานเท่ากับ 1,064 ไบต์ ส่วนของขั้นตอนวิธีแบบโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์จะใช้หน่วยความจำในการทำงานเท่ากับ 11,760 ไบต์ ซึ่งวิธีออนโพลิซีมอนติคาร์โลต้องการหน่วยความจำที่ใช้ในการเก็บข้อมูลน้อยกว่า วิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์

หลังจากที่ได้ทำการจำลองแบบกับโครงข่ายขนาดเล็กซึ่งพบว่าวิธีออนโพลิซีมอนติคาร์โลสามารถลดโพลล์ลิ่งโอเวอร์เฮด ที่เกิดขึ้นจากการทำงานของการเฝ้าตรวจสถานะของโครงข่ายได้ และใช้หน่วยความจำที่ใช้ในการเก็บข้อมูลในการทำงานที่น้อยกว่าวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ ดังนั้นเพื่อเป็นการทดสอบกับโครงข่ายที่มีความซับซ้อนมากขึ้น จะได้ทำการจำลองแบบกับโครงข่ายที่มีขนาดใหญ่โดยจะทำการวิจัยในบทที่ 4 ต่อไป

บทที่ 4
การดำเนินการวิจัยในโครงข่ายขนาดใหญ่

4.1 กล่าวนำ

ระบบโครงข่ายที่ถูกนำมาใช้งานในหน่วยงานต่างๆ ด้วยจำนวนผู้ใช้ พื้นที่การให้บริการ รวมทั้งปริมาณข้อมูลที่ถ่ายโอนอยู่บนโครงข่าย ส่งผลให้โครงข่ายมีขนาดใหญ่ขึ้น มีความซับซ้อน และจำนวนอุปกรณ์ที่เชื่อมต่อร่วมอยู่ในโครงข่ายมากขึ้น อีกทั้งยังมีการเชื่อมต่อโครงข่ายที่มีลักษณะแตกต่างกันเข้าด้วยกัน ทั้งในแง่ของระบบบริหารจัดการอุปกรณ์ โครงรูปของโครงข่าย (network topology) อุปกรณ์หลายชนิดจากหลากหลายผู้ผลิต หรือแม้แต่การใช้งานซึ่งอาจต้องใช้งานที่โปรโตคอลแตกต่างกันอีกด้วย จากความแตกต่างเหล่านี้จึงหลีกเลี่ยงความสลับซับซ้อนที่เกิดขึ้นในโครงข่ายได้ยาก เมื่อโครงข่ายมีขนาดใหญ่ มีความซับซ้อนมากขึ้นและมีความสำคัญต่อการใช้งานที่สูงขึ้นด้วยเช่นกัน ทั้งยังเกิดความต้องการในเรื่องของประสิทธิภาพ และเสถียรภาพของโครงข่ายที่ต้องมีสูงขึ้นเช่นกัน ในโครงข่ายที่มีขนาดใหญ่จะประกอบด้วยอุปกรณ์โครงข่ายจำนวนมาก ดังนั้นจึงมีโอกาสที่จะเกิดความเสียหายกับอุปกรณ์เหล่านี้สูงขึ้นซึ่งอาจส่งผลให้ประสิทธิภาพของโครงข่ายลดลง ดังนั้นระบบบริหารจัดการโครงข่ายที่ดีต้องสามารถทำการวิเคราะห์เหตุการณ์ความผิดปกติที่เกิดขึ้นได้หลายรูปแบบอย่างมีประสิทธิภาพและถูกต้อง รวมทั้งต้องสามารถตรวจพบจุดขัดข้องที่เกิดขึ้นได้อย่างรวดเร็วอีกด้วย จึงจำเป็นต้องอาศัยระบบบริหารจัดการโครงข่ายที่มีประสิทธิภาพเพื่อตอบสนองความต้องการดังกล่าว (Sreedhar, Hill, and Stanley, 2000)

เนื้อหาของบทนี้จะประกอบด้วยหัวข้อ 4.2 การนิยามปัญหาโดยจะกล่าวถึงโครงรูปของโครงข่ายที่ใช้ในการจำลองแบบสำหรับโครงข่ายขนาดใหญ่ การกำหนดพารามิเตอร์ต่างๆ เช่น สถานะของโครงข่าย การกระทำทั้งหมดที่เป็นไปได้ สิ่งที่ได้จากการสังเกต และการกำหนดผลรางวัล หัวข้อ 4.3 เป็นการดำเนินการทดลองการจำลองแบบกับโครงข่ายขนาดใหญ่ โดยทำการจำลองแบบด้วยวิธีออนโพลิซีมอนติคาร์โล วิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ และวิธีคำนึงถึงโครงรูปของโครงข่าย หัวข้อ 4.4 จะเป็นในส่วนของผลการจำลองของแบบจำลองกรณีโครงข่ายขนาดใหญ่โดยทำการเปรียบเทียบตัวชี้วัดที่สำคัญ เช่น จำนวนครั้งของการโพลล์ที่เกิดขึ้น จำนวนการแจ้งตำแหน่งที่อุปกรณ์ขัดข้องได้ถูกตำแหน่ง และจำนวนครั้งของการกระทำที่ต้องใช้ในการวิเคราะห์หาจุดขัดข้อง หัวข้อ 4.5 เป็นส่วนของการวิเคราะห์ผลการจำลองของแบบจำลองกรณีโครงข่ายขนาดใหญ่ และความซับซ้อนของขั้นตอนวิธีที่ใช้ในการจำลองแบบ และหัวข้อ 4.6 จะเป็นในส่วนของบทสรุป

4.2 การนิยามปัญหา

ในโครงข่ายขนาดใหญ่ที่ประกอบด้วยอุปกรณ์จำนวนมาก ครอบคลุมพื้นที่บริเวณกว้าง มีโครงรูปของโครงข่ายที่สลับซับซ้อนนั้น การเฝ้าตรวจสถานะของโครงข่ายที่ประกอบด้วยอุปกรณ์จำนวนมากในโครงข่ายเดียวกันจะประสบปัญหาต่างๆ เช่น มีปริมาณข้อมูลที่เกิดจากการเฝ้าตรวจสถานะของโครงข่ายสูงเกินไป หรือปัญหาคอขวด (bottleneck) ที่มีปริมาณข้อมูลสูงเฉพาะที่จนเกิดการคับคั่งของข้อมูล (traffic jam) จึงได้มีผู้ทำการศึกษาการเลือกระบบบริหารจัดการโครงข่ายมาใช้งาน (Steinder and Sethi, 2004) และพบว่ามีสองแนวทางที่กระทำได้คือ แนวทางแรกเป็นการใช้ระบบบริหารจัดการโครงข่ายแบบรวมศูนย์ (centralize) ตรวจสอบสถานะของอุปกรณ์ในโครงข่ายทั้งหมดที่จุดๆ เดียวซึ่งพบว่าเกิดปัญหาของปริมาณข้อมูลที่เกิดจากการเฝ้าตรวจสถานะของโครงข่ายสูงเกินไปและปัญหาคอขวดที่มีปริมาณข้อมูลสูงเฉพาะที่ ส่วนแนวทางที่สองจะเป็นการใช้ระบบบริหารจัดการโครงข่ายแบบกระจาย (distributed) โดยทำการแบ่งอุปกรณ์โครงข่ายออกเป็นกลุ่มๆ หรือเป็นโครงข่ายย่อยที่มีขนาดเล็กลงและพอเหมาะต่อการทำงานของการเฝ้าตรวจสถานะของโครงข่ายแต่ละจุดซึ่งพบว่าวิธีการแบ่งโครงข่ายออกเป็นกลุ่มๆ หรือเป็นโครงข่ายย่อยนี้จะให้ประสิทธิภาพในด้านต่างๆ ที่ดีกว่าแบบรวมศูนย์ (Steinder and Sethi, 2004) ดังนั้นในการทดลองนี้จึงเลือกใช้ระบบบริหารจัดการโครงข่ายแบบกระจาย โดยทำการแบ่งโครงข่ายออกเป็นโครงข่ายย่อยๆ ที่มีขนาดเล็กลงและพอเหมาะต่อการทำงานของการเฝ้าตรวจสถานะของโครงข่ายแต่ละจุด และทำการจำลองแบบการทำงานของการเฝ้าตรวจสถานะของโครงข่ายในแต่ละโครงข่ายย่อยๆ นี้

ในการทดลองนี้ได้กระทำการจำลองแบบทั้งหมดสามแบบด้วยกันคือ แบบแรกคือวิธีรีอินฟอร์สเมนท์เลิร์นนิ่งแบบออนโพลิซีมอนติคาร์โล แบบที่สองคือวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ (He, 2003) และแบบที่สามคือวิธีคำนึงถึงโครงรูปโครงข่าย (Han, Ahn, and Chung, 2001) เพื่อทำการเปรียบเทียบค่าชี้วัดต่างๆ เช่น จำนวนครั้งของการกระทำ จำนวนผลรางวัลที่เกิดขึ้น

โครงรูปของโครงข่ายที่ใช้ในการจำลองแบบ ได้เลือกใช้แบบเดียวกับที่ใช้ในการจำลองแบบสำหรับการเฝ้าตรวจสถานะของโครงข่ายในกรณีที่โครงข่ายมีขนาดใหญ่ ที่นำเสนอโดย (Lin,Chan, 2002) โดยประกอบด้วยอุปกรณ์โครงข่ายซึ่งในที่นี้หมายถึงโหนด (node) จำนวน 30 โหนด โดยมีลิงค์ (link) เชื่อมโยงถึงกันและมีโหนด 0 โหนด 1 และโหนด 2 เป็นโหนดหลัก (core network) ตามรูปที่ 4.1

ในการจำลองแบบนั้นกำหนดเงื่อนไขว่า ต้องการที่จะทำการตรวจสอบเฉพาะสถานะของโหนดทั้ง 30 โหนดเท่านั้นและกำหนดให้ลิงค์ทำงานเป็นปกติตลอดเวลาดังนั้นจึงสามารถกำหนดค่าพารามิเตอร์ต่างๆ ได้ดังนี้

4.2.1 เซตของสถานะที่เป็นไปได้ของโครงข่ายขนาดใหญ่

สถานะของสิ่งแวดล้อมซึ่งในที่นี้คือ สถานะของโหนดที่เชื่อมต่ออยู่ในระบบโครงข่ายที่ประกอบด้วยสถานะที่ปกติ (normal) และสถานะที่ไม่ปกติ (abnormal) ดังนั้นสถานะที่เป็นไปได้ทั้งหมดของโครงข่ายที่มีโครงรูปโครงข่ายดังรูปที่ 4.1 จึงประกอบด้วยเซตของสถานะทั้งหมดที่เป็นไปได้หรือปริภูมิสเตต (state space) โดยกำหนดให้สถานะที่โหนด 0 ขัดข้องแทนด้วย s_0 สถานะที่โหนด 1 ขัดข้องแทนด้วย s_1 จนถึงสถานะที่โหนด 29 ขัดข้องแทนด้วย s_{29} และสถานะที่โหนดทั้งหมดทำงานเป็นปกติซึ่งในที่นี้แทนด้วย s_{30} ถ้าให้ S แทนสถานะทั้งหมดที่เป็นไปได้ N คือจำนวนโหนดจะได้ว่า $S = \{s_0, s_1, s_2, ...s_N\}$ เมื่อ s_i หมายถึงสถานะซึ่งโหนดตัวที่ i ขัดข้อง และ s_N หมายถึงสถานะซึ่งโหนดทั้งหมดอยู่ในสถานะปกติ

4.2.2 เซตของการกระทำที่เป็นไปได้ของโครงข่ายขนาดใหญ่

ในการเฝ้าตรวจสถานะของโครงข่าย เมื่อพบว่าโครงข่ายอยู่ในสถานะใดสถานะหนึ่งจะต้องมีการเลือกการกระทำเพื่อที่จะจัดการกับสถานะที่ตรวจพบนั้น โดยจะประกอบไปด้วยกรณีที่ตรวจพบว่ามีโหนดที่เชื่อมต่ออยู่ในโครงข่ายขัดข้องจะต้องตัดสินใจในการเปลี่ยนหรือซ่อม (repair) โหนดที่ขัดข้องนั้น แต่หากพบว่ามีความไม่ชัดเจนที่จะสรุปว่าโหนดขัดข้องหรือไม่จะต้องดำเนินการโพลล์ (polling) ไปที่โหนดนั้นหรือโหนดที่ต้องใช้โหนดนั้นเป็นเส้นทางในการส่งผ่านข้อมูล (transit node) เพื่อยืนยันสถานะที่แท้จริง ซึ่งหากโหนดนั้นขัดข้องจริงจะส่งผลให้โหนดที่ต้องใช้โหนดนั้นเป็นเส้นทางในการส่งผ่านข้อมูลไม่สามารถ รับ – ส่ง ข้อมูลได้ และหากพบว่ามีโหนดขัดข้องจริงจะต้องเลือกทำการเปลี่ยนหรือซ่อมโหนด (repair) ที่ขัดข้องนั้น เพื่อให้ระบบโครงข่ายสามารถให้บริการได้ตามปกติ หรือกรณีที่ตรวจพบว่าโครงข่ายอยู่ในสถานะปกติจะไม่ต้องเลือกการกระทำใด เพียงแจ้งว่าโครงข่ายอยู่ในสถานะปกติเท่านั้น ดังนั้นถ้าให้ A เป็นเซตของการตัดสินใจที่เป็นไปได้ทั้งหมด (action space) ของตัวกระทำการตัดสินใจจะได้ว่า $A = A^r \cup A^p$ เมื่อ $A^r = \{a_0^r, a_1^r, a_2^r, ...a_{N-1}^r\}$ และ $A^p = \{a_0^p, a_1^p, a_2^p, ...a_N^p\}$ เมื่อ a_i^r หมายถึงการเลือกที่จะทำการซ่อมโหนดที่ i ส่วน a_j^p หมายถึงทำการโพลล์ไปที่โหนดที่ j และ a_N^p หมายถึงการกระทำเมื่อโครงข่ายอยู่ในสถานะปกติ

4.2.3 เซตของสิ่งที่ได้จากการสังเกตของโครงข่ายขนาดใหญ่

การทำงานของการเฝ้าตรวจสถานะของโครงข่าย ที่ต้องทำงานภายใต้สภาวะที่สิ่งที่ได้จากการสังเกตไม่ชัดเจนดังที่ได้กล่าวมาแล้ว จึงทำให้การทำงานของการเฝ้าตรวจสถานะของโครงข่ายจะทราบเพียงสถานะที่ได้จากการสังเกตเท่านั้น (He, 2003) ดังนั้นหากทำการโพลล์ไปที่โหนดใดโหนดหนึ่งแล้วพบว่าโหนดนั้นๆ เสมือนอยู่ในสถานะปกติ (normal) หรือสถานะผิดปกติ (abnormal) ตามลำดับซึ่งในความเป็นจริงแล้วข้อมูลจากการโพลล์ดังกล่าวอาจให้ข้อมูลที่

ไม่ถูกต้องก็เป็นได้ จึงสามารถเขียนเซตของเหตุการณ์ที่สังเกตได้ทั้งหมด (observation space) ได้ว่า $z = (normal, abnormal)$ โดย normal หมายถึงเหตุการณ์ที่สังเกตได้หลังจากทำการโพลล์ไปที่โหนดหนึ่งๆ แล้วพบว่าโหนดนั้นเสมือนอยู่ในสถานะปกติและ abnormal หมายถึงเหตุการณ์ที่สังเกตได้หลังจากทำการโพลล์ไปที่โหนดหนึ่งๆ แล้วพบว่าโหนดนั้นเสมือนอยู่ในสถานะผิดปกติ ซึ่งในความเป็นจริงแล้ว ข้อมูลจากการโพลล์ดังกล่าวอาจให้ข้อมูลที่ไม่ถูกต้องก็เป็นได้ จึงใช้ระดับของความน่าจะเป็นแทนความถูกต้องของข้อมูลนี้ ซึ่งเรียกว่าความน่าจะเป็นของการเกิดสัญญาณเตือนที่ผิดพลาด (false alarm) ที่มีค่าอยู่ระหว่าง 0.0 ถึง 1.0

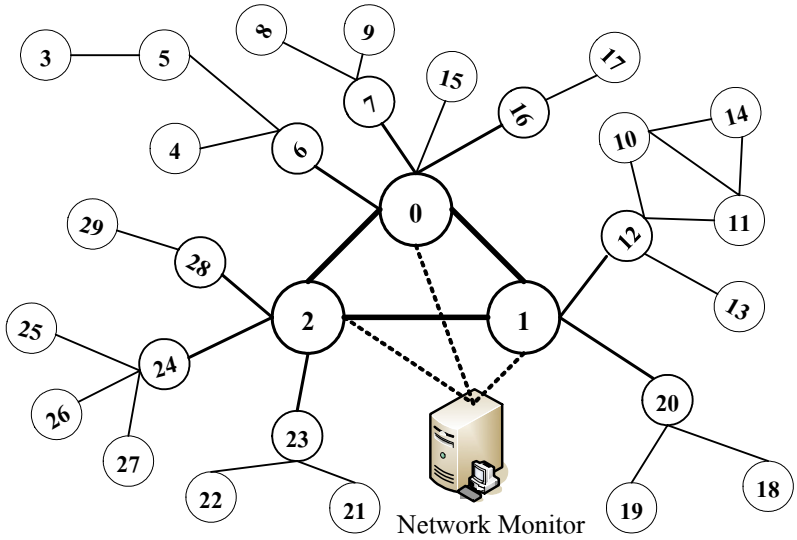
4.2.4 โครงสร้างการกำหนดผลรางวัลของโครงข่ายขนาดใหญ่

หลักในการกำหนดผลรางวัลจะคล้ายกับการจำลองในโครงข่ายขนาดเล็กคือ เมื่อสามารถค้นหาจุดที่ขัดข้องพบ และเลือกการกระทำที่ทำการซ่อมอุปกรณ์ที่ขัดข้องนั้นจะถือว่าเป็นการเลือกการกระทำที่ถูกต้อง จะกำหนดค่าผลรางวัลที่มีค่าสูงสุดเท่ากับ c_1 หรือเมื่อสามารถค้นหาจุดที่ขัดข้องพบแต่ตัวกระทำการตัดสินใจยังไม่มั่นใจว่าเป็นตำแหน่งที่ถูกต้องหรือไม่ จึงตัดสินใจที่จะทำการโพลล์ไปที่โหนดนั้นจะกำหนดค่าผลรางวัลรองลงมามีค่าเท่ากับ c_2 แต่หากตัดสินใจทำการโพลล์ไปที่โหนดอื่นๆ ที่ไม่ได้เกี่ยวข้องกับโหนดที่กำลังสังเกตอยู่จะถือว่าเป็นการกระทำที่ไม่ก่อให้เกิดประโยชน์ จึงกำหนดค่าผลรางวัลให้มีค่าต่ำรองลงมามีค่าเท่ากับ c_3 และหากมีการตัดสินใจที่ผิดพลาด เช่น ทำการตัดสินใจที่เลือกกระทำการซ่อมโหนดที่ไม่ขัดข้องซึ่งเป็นการตัดสินใจที่ผิดพลาด ที่ก่อให้เกิดความสูญเสียเกิดขึ้นจึงต้องทำการลงโทษในการตัดสินใจที่ผิดพลาดนี้ ด้วยการกำหนดให้ผลรางวัลต่ำที่สุดให้มีค่าเท่ากับ c_4 ดังนั้นจะได้ความสัมพันธ์ของผลรางวัลทั้งหมด คือ $c_1 \gg c_2 > c_3 \gg c_4$

4.3 การดำเนินการทดลองของแบบจำลองโครงข่ายขนาดใหญ่

ในการทดลองนี้ได้เลือกใช้ขั้นตอนวิธีในการบริหารจัดการโครงข่าย 3 แบบได้แก่ แบบแรกคือวิธีออนโพลิซีมอนติคาร์โลที่ใช้วิธีพิจารณาผลรางวัลของทุกๆ คู่ของสิ่งที่ได้จากการสังเกตและการกระทำมาสร้างกฎควบคุมที่ใช้ในการตัดสินใจ แบบที่สองคือวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ที่ใช้วิธีสร้างลำดับขั้นของการโพลล์ในการค้นหาจุดขัดข้อง และแบบที่สามคือวิธีคำนึงถึงโครงรูปโครงข่ายซึ่งเป็นวิธีการที่นำเสนอโดย (Han, Ahn, and Chung, 2001) ที่ใช้วิธีพิจารณาโครงรูปของโครงข่ายเพื่อสร้างเงื่อนไขของการโพลล์ โดยได้ทำการทดลองทั้งสามแบบและนำผลการทดลองมาเปรียบเทียบค่าชี้วัดต่างๆ เช่น จำนวนครั้งของการกระทำ จำนวนผลรางวัลที่ได้รับต่อเอพิโซด และปริมาณหน่วยความจำที่ต้องใช้ในการทำงานของขั้นตอนวิธีทั้ง 3 แบบ

ในการจำลองแบบกรณีโครงข่ายขนาดใหญ่ ด้วยโครงข่ายที่มีโครงรูปและการเชื่อมต่อดังรูป 4.1 ซึ่งประกอบด้วยสถานะของโครงข่ายทั้งหมด 31 สถานะ มีการกระทำทั้งหมด 61 การกระทำ และกำหนดผลรางวัลดังนี้คือ $c_1 = 1000$, $c_2 = 10$, $c_3 = (-100)$, $c_4 = (-1000)$ ซึ่ง $c_1 \gg c_2 > c_3 \gg c_4$ โดยการกำหนดผลรางวัลจะแตกต่างจากการจำลองในโครงข่ายขนาดเล็ก เนื่องจากในโครงข่ายที่มีขนาดใหญ่ขึ้น จะประกอบด้วยจำนวนสถานะและการกระทำทั้งหมดที่เป็นไปได้ ที่เพิ่มขึ้นตามจำนวนของโหนดและการกระทำที่เป็นไปได้จึงต้องกำหนดผลรางวัลให้เหมาะสมในส่วนของสถานะของอุปกรณ์ที่สังเกตได้ประกอบด้วย 2 สถานะเช่นกันคือ $z = \{0,1\}$ และกำหนดความน่าจะเป็นในการเกิดสัญญาณเตือนที่ผิดพลาด (false alarm) มีค่าเปลี่ยนแปลงเพิ่มขึ้นจาก 0.0 ถึง 1.0



รูปที่ 4.1 โครงรูปโครงข่ายที่ใช้ในการจำลองแบบโครงข่ายขนาดใหญ่

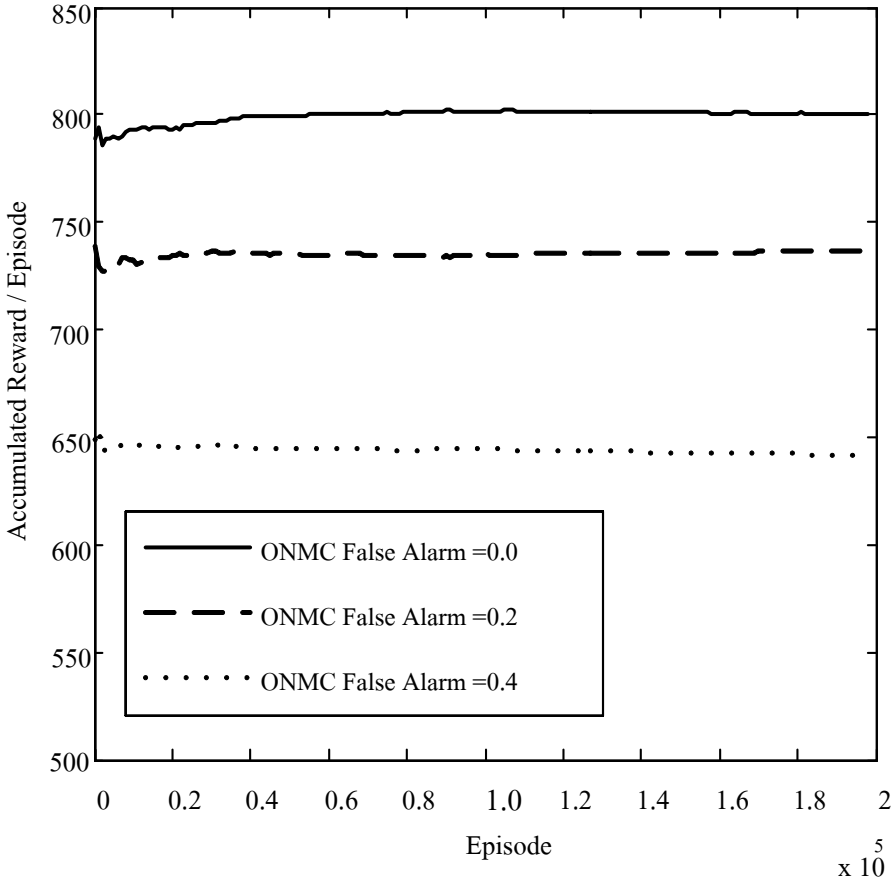
4.3.1 การทดลองการจำลองแบบวิธีออนโพลิซีมอนติคาร์โล

การจำลองแบบด้วยวิธีออนโพลิซีมอนติคาร์โล ที่มีขั้นตอนดังหัวข้อที่ 2.5 และกำหนดให้ “1” แทนกรณีเกิดการขัดข้องที่โหนดนั้นๆ และ “0” แทนกรณีที่โหนดอยู่ในสถานะปกติ ถ้า S คือสถานะของโครงข่ายทั้งหมดที่เป็นไปได้ และ s แทนสถานะของโหนดในโครงข่ายจะได้ว่า $S = \{s_0, s_1, s_2, ...s_N\}$ และ $s \in S$ โดยกำหนดให้ s_o หมายถึงระบบอยู่ในสถานะที่โหนด 0 ขัดข้อง ส่วน s_1 จะหมายถึงระบบอยู่ในสถานะที่โหนด 1 ขัดข้อง และ s_N จะหมายถึงทุกโหนดในระบบโครงข่ายอยู่ในสถานะปกติ

การกระทำของการเฝ้าตรวจสถานะของโครงข่าย จะประกอบไปด้วยการซ่อมโหนดแต่ละโหนดและการโพลล์ไปที่โหนด เมื่อ A คือการกระทำทั้งหมดที่เป็นไปได้และ a แทนการกระทำ จะได้การกระทำทั้งหมดที่เป็นไปได้คือ $A = A^r \cup A^p$ เมื่อ $A^r = \left\{a_0^r, a_1^r, a_2^r, ...a_{N-1}^r\right\}$ และ $A^p = \left\{a_0^p, a_1^p, a_2^p, ...a_N^p\right\}$ และ $a \in A$ เช่น a_1^r หมายถึงการเลือกการกระทำที่จะทำการซ่อมโหนดที่ 1 หรือ a_1^p หมายถึงการโพลล์ไปที่โหนดที่ 1 เพื่อทำการตรวจสอบสถานะของโหนดที่ 1 และหากระบบอยู่ในสถานะปกติการกระทำที่แจ้งว่าระบบปกติ คือ a_N^p และเซตของการสังเกตที่ได้จะประกอบไปด้วย สถานะปกติและสถานะไม่ปกติ แทนด้วย $Z = \{0,1\}$

ในการจำลองแบบเนื่องจากในโครงข่ายขนาดใหญ่ที่ประกอบด้วยอุปกรณ์จำนวนมากซึ่งส่งผลให้จำนวนสถานะและการกระทำที่เป็นไปได้มีขนาดใหญ่ด้วย จึงต้องทำการจำลองแบบเป็นจำนวนครั้งที่มากกว่าโครงข่ายขนาดเล็ก โดยได้ทำการจำลองแบบจำนวน 200,000 ครั้งต่อค่าความน่าจะเป็นของการเกิดสัญญาณเตือนที่ผิดพลาดหนึ่งค่า

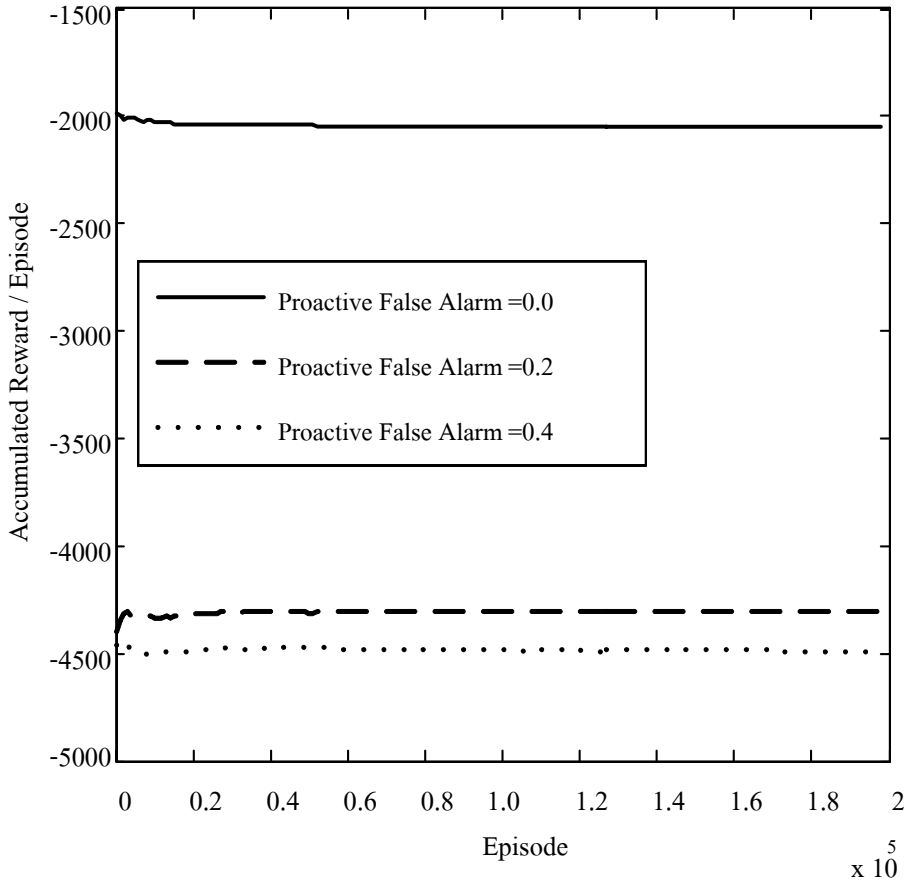
การจำลองแบบได้แบ่งเป็นสองระยะเช่นกันคือ ระยะแรกเป็นการสร้างการเรียนรู้ให้กับตัวกระทำการตัดสินใจเพื่อทำการปรับปรุงกฎควบคุมให้ได้กฎควบคุมที่ดี และในระยะที่สองจะเป็นการจำลองการใช้กับสถานการณ์จริง โดยนำกฎควบคุมที่ได้รับการปรับปรุงจากระยะแรกมาใช้เป็นกฎควบคุมในระยะที่สอง จากผลการจำลองแบบโดยพิจารณาที่ผลของการตัดสินใจในรูปของผลรางวัลสะสมต่อเอพพิโซด ดังรูปที่ 4.2 โดยที่แกนตั้งคือผลรางวัลสะสมที่เกิดขึ้นในแต่ละเอพพิโซดและแกนนอนเป็นจำนวนเอพพิโซดที่ทำการจำลองแบบ พบว่าผลรางวัลสะสมต่อเอพพิโซดที่เกิดขึ้นจากการจำลองแบบที่ค่าความน่าจะเป็นของการเกิดสัญญาณเตือนที่ผิดพลาดมีค่าเท่ากับ 0.0 เท่ากับ 0.2 และเท่ากับ 0.4 ซึ่งพบว่าการตัดสินใจของตัวกระทำการตัดสินใจจะมีการตัดสินใจที่ดีเนื่องจากผลรางวัลที่ได้อยู่ในช่วงที่เป็นบวก สอดคล้องกับการกำหนดผลรางวัลที่กำหนดให้ผลรางวัลมีค่าเป็นบวกเมื่อมีการตัดสินใจที่ถูกต้อง ผลรางวัลสะสมต่อเอพพิโซดในกรณีที่ความน่าจะเป็นของการเกิดสัญญาณเตือนที่ผิดพลาดมีค่าเท่ากับ 0.0 จะสูงกว่าผลรางวัลสะสมต่อเอพพิโซดในกรณีที่ความน่าจะเป็นของการเกิดสัญญาณเตือนที่ผิดพลาดมีค่าเท่ากับ 0.2 และ 0.4 ซึ่งหมายถึงประสิทธิภาพในการตัดสินใจของตัวกระทำการตัดสินใจจะลดลง เมื่อความน่าจะเป็นของการเกิดสัญญาณเตือนที่ผิดพลาดหรือความไม่ชัดเจนของสิ่งที่ได้จากการสังเกตมีค่าเพิ่มมากขึ้น จึงไม่ชัดเจนเพียงพอที่จะนำมาใช้ในการตัดสินใจให้ถูกต้อง



รูปที่ 4.2 ผลรางวัลสะสมต่อเอพพิโซดของวิธีออนโพลิซีมอนติคาร์โล

4.3.2 การทดลองการจำลองแบบวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์

การจำลองแบบวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์นี้ ถูกนำเสนอโดย (He, 2003) ดังที่ได้กล่าวมาแล้วในหัวข้อ 2.6 และ 3.3.2 โดยทำการแทนสถานะของโครงข่ายด้วยความน่าจะเป็น (belief state) และสร้างขั้นตอนวิธีในการปรับค่าความน่าจะเป็นและใช้ความน่าจะเป็นนี้แสดงถึงสถานะที่แท้จริง เช่น $\mathbf{b} = [0.8, 0.0, 0.1, 0.1, ..., 0.0]$ หมายความถึงการเฝ้าตรวจสถานะของโครงข่ายมีความเชื่อว่า โหนดหมายเลข 0 อาจจะขัดข้องด้วยความน่าจะเป็น 0.8 และเชื่อว่าโหนดหมายเลข 2 และ 3 อาจจะขัดข้องด้วยความน่าจะเป็น 0.1 โดยมีขั้นตอนวิธีในการทำงานดังหัวข้อ 2.6 ที่ได้กล่าวมาแล้ว จากผลการจำลองแบบพบว่าผลรางวัลสะสมต่อเอพพิโซดจะลดลงเมื่อสิ่งที่ได้จากการสังเกตมีความไม่ชัดเจนเพิ่มสูงขึ้นดังรูปที่ 4.3 ซึ่งแสดงผลรางวัลที่เกิดจากการตัดสินใจในการเลือกการกระทำ ซึ่งหมายถึงการตัดสินใจของตัวกระทำการตัดสินใจจะมีความสามารถในการตัดสินใจที่ลดลงเป็นอย่างมาก เมื่อสิ่งที่ได้จากการสังเกตมีความไม่ชัดเจนเพิ่มมากขึ้น

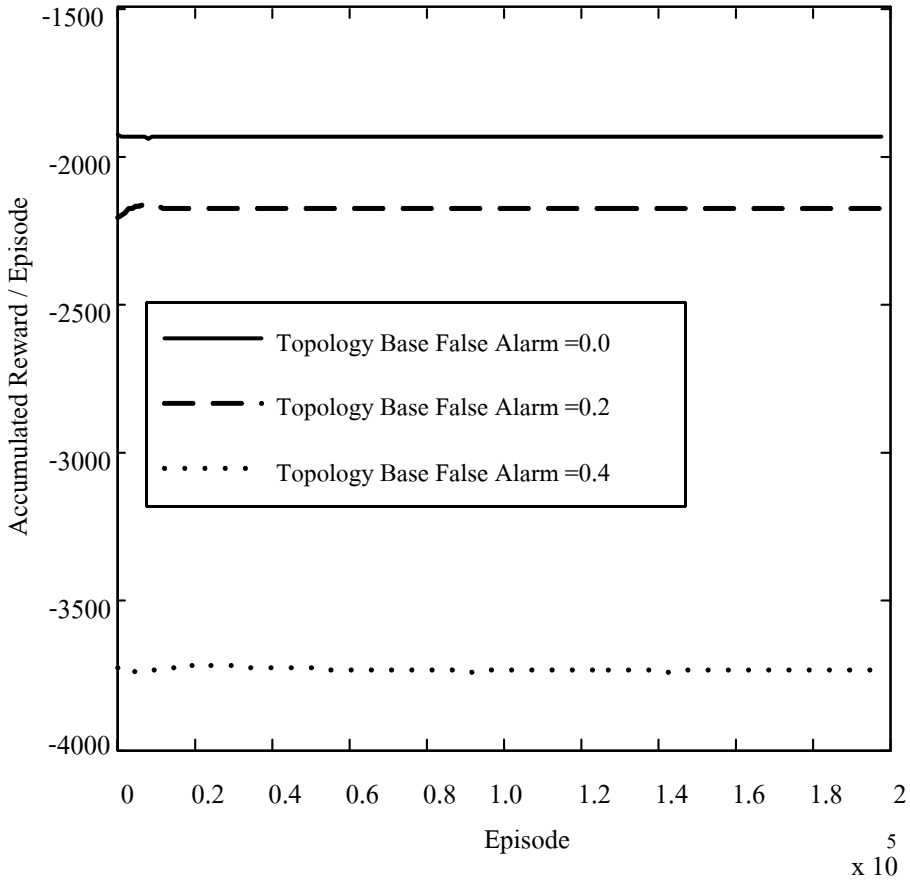


รูปที่ 4.3 ผลรางวัลสะสมต่อเอพิโซดของวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์

4.3.3 การทดลองการจำลองแบบวิธีคำนึงถึงโครงรูปโครงข่าย

เป็นการจำลองแบบที่นำเสนอโดย (Han, Ahn, and Chung, 2001) มีหลักการคือ ระบบจะทำการโพลล์ไปยังโหนดต่างๆ ในโครงข่ายทุกๆ โหนดตามลำดับของโหนด เมื่อพบว่าโหนดใดไม่มีข้อมูลตอบกลับมาระบบจะทำการโพลล์ซ้ำไปอีกถ้ายังไม่มีข้อมูลตอบกลับมาระบบจะทำการโพลล์ซ้ำเป็นครั้งที่สาม หากยังไม่มีข้อมูลตอบกลับมาระบบจะสรุปว่าโหนดนั้นมีความขัดข้องเกิดขึ้น นอกจากเงื่อนไขดังกล่าวแล้วระบบยังคำนึงถึงโครงรูปของโครงข่ายด้วย โดยจะไม่ทำการโพลล์ไปยังโหนดที่ต้องใช้เส้นทางในการส่งผ่านข้อมูลผ่านไปทางโหนดที่ขัดข้องอยู่ เพราะเมื่อโหนดนั้นขัดข้องจะส่งผลให้ไม่สามารถส่งข้อมูลไปถึงโหนดเหล่านั้นได้ วิธีการนี้จึงทำให้ลดจำนวนครั้งของการโพลล์ลงได้มาก เช่น จากโครงรูปที่ใช้ในการจำลองแบบ (รูปที่ 4.1) หากมีการโพลล์ไปยังโหนดหมายเลข 7 แต่ไม่ได้รับการตอบกลับมาระบบจะทำการโพลล์ไปที่โหนดหมายเลข 7 ซ้ำอีก 2 ครั้งหากยังไม่ได้รับข้อมูลตอบกลับมา จะทำการสรุปว่าโหนดหมายเลข 7 ขัดข้องถัดจากนั้น ระบบจะไม่ทำการโพลล์ไปยังโหนดหมายเลข 8 และ 9 เพราะถ้าพิจารณาจากโครงรูปของ

โครงข่ายจะพบว่าหากโหนดหมายเลข 7 ขัดข้องจะไม่สามารถส่งข้อมูลไปยังโหนดที่ 8 และ 9 ได้ แต่วิธีการนี้ยังมีจุดด้อยคือ ในกรณีที่โหนดที่ขัดข้องเป็นโหนดที่อยู่ท้ายสุดของโครงข่าย เช่นโหนดหมายเลข 8 หรือโหนดหมายเลข 9 จะไม่สามารถลดจำนวนการโพลล์ที่เกิดขึ้นบนโครงข่ายลงได้ ซึ่งจากผลการจำลองแบบที่กำหนดผลรางวัลเช่นเดียวกับวิธีออนโพลิซีมอนติคาร์โล และวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ พบว่าผลรางวัลสะสมต่อเอพพิโซด จะลดลงเมื่อความไม่ชัดเจนของสิ่งที่ได้จากการสังเกตมีความไม่ชัดเจนเพิ่มสูงขึ้น ดังรูปที่ 4.4 ซึ่งหมายถึงการตัดสินใจของตัวกระทำการตัดสินใจจะมีความสามารถในการตัดสินใจที่ถูกต้องน้อยกว่าเนื่องจากผลรางวัลที่ได้รับจะเป็นค่าที่ติดลบแม้แต่ในกรณีที่สิ่งที่ได้จากการสังเกตถูกต้องทั้งหมด (false alarm เท่ากับ 0.0)

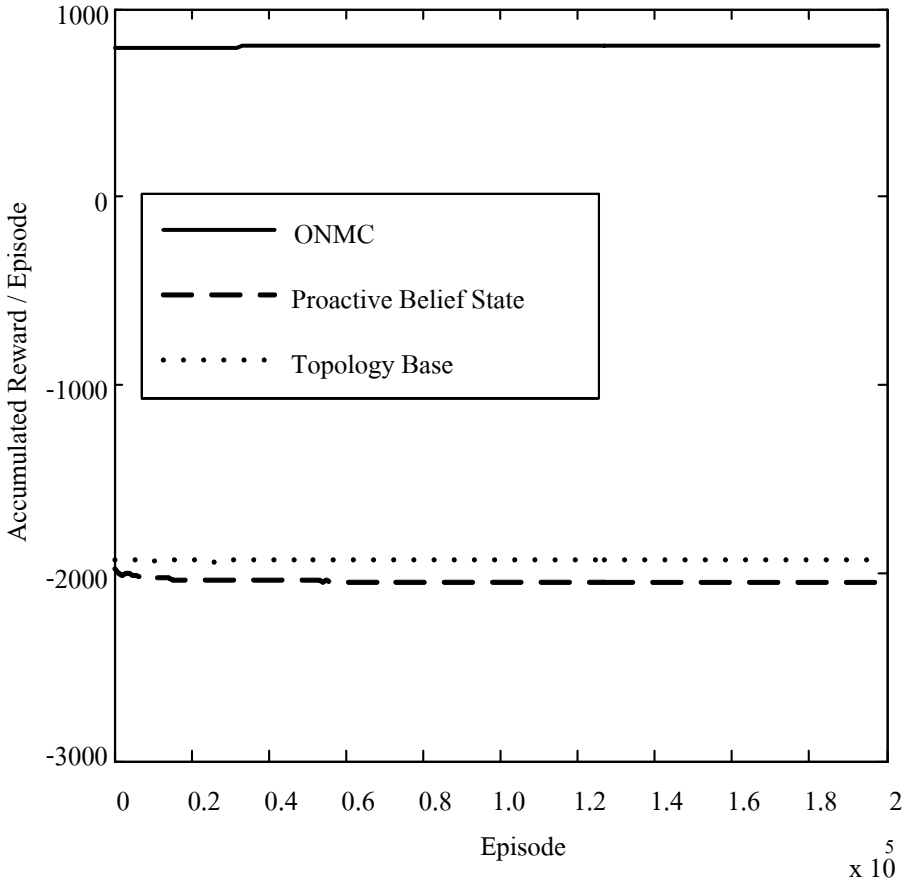


รูปที่ 4.4 ผลรางวัลสะสมต่อเอพพิโซดของวิธีคำนึงถึงโครงรูปโครงข่าย

4.4 ผลการทดลองของแบบจำลองโครงข่ายขนาดใหญ่

จากผลการจำลองแบบโดยพิจารณาผลรางวัลสะสมต่อเอพพิโซดที่เกิดขึ้น ดังรูปที่ 4.5 เป็นการเปรียบเทียบผลรางวัลสะสมต่อเอพพิโซด ที่ความน่าจะเป็นของการเกิดสัญญาณเตือนที่

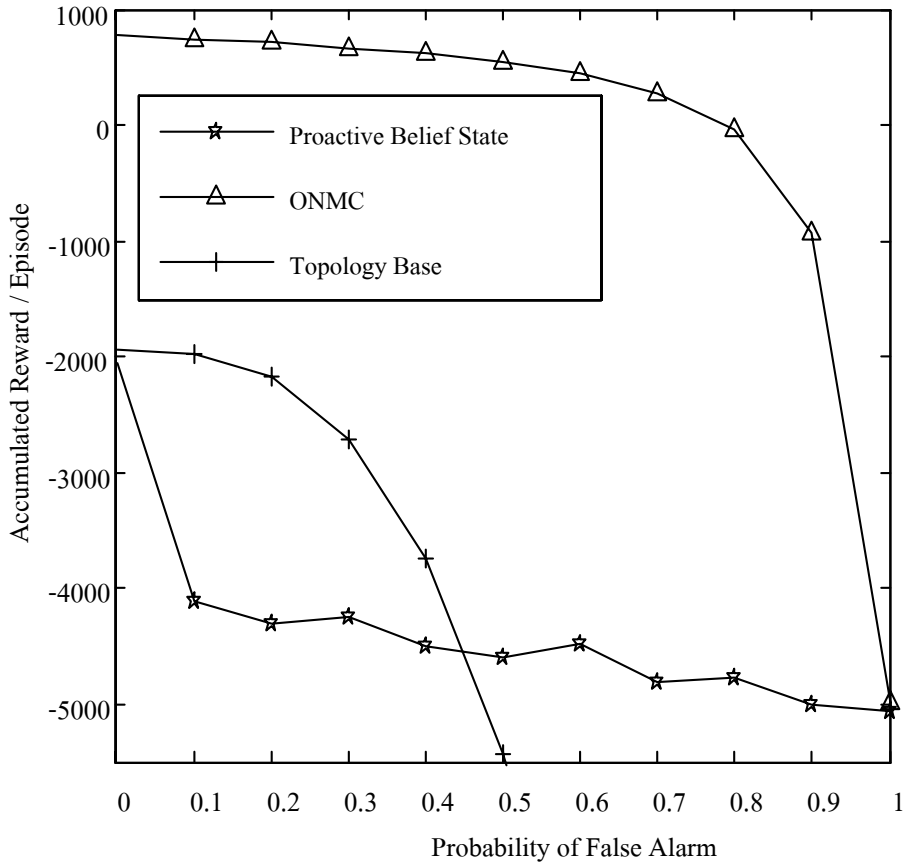
ผิดพลาดมีค่าเท่ากับ 0.0 ของวิธีออนโพลิซีมอนติคาร์โล วิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ และวิธีคำนึงถึงโครงรูปโครงข่าย ซึ่งพบว่าตัวกระทำการตัดสินใจของวิธีออนโพลิซีมอนติคาร์โล สามารถตัดสินใจได้ดีกว่าทั้งสองวิธีจึงส่งผลให้ได้รับผลรางวัลสะสมต่อเอพพิโซดที่สูงกว่า



รูปที่ 4.5 เปรียบเทียบผลรางวัลสะสมต่อเอพพิโซดของวิธีออนโพลิซีมอนติคาร์โล วิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์และวิธีคำนึงถึงโครงรูปโครงข่าย

ในส่วนของผลรางวัลสะสมต่อเอพพิโซดที่เกิดขึ้น เมื่อสิ่งที่ได้จากการสังเกตมีความไม่ถูกต้องเพิ่มสูงขึ้นหรือเมื่อความน่าจะเป็นของการเกิดสัญญาณเตือนที่ผิดพลาดมีค่าเพิ่มขึ้น ซึ่งได้ผลการจำลองแบบดังรูปที่ 4.6 โดยพบว่าวิธีออนโพลิซีมอนติคาร์โลมีผลรางวัลสะสมต่อเอพพิโซดที่สูงกว่าซึ่งแสดงให้เห็นว่าสามารถที่จะตัดสินใจในการเลือกการกระทำได้ถูกต้องมากกว่า แม้ว่าสิ่งที่ได้จากการสังเกตจะมีความไม่ชัดเจนเพิ่มสูงขึ้นก็ตาม ส่วนวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์จะให้ผลรางวัลสะสมต่อเอพพิโซดที่ลดลงเป็นอย่างมาก ระหว่างค่าความน่าจะเป็นของการเกิดสัญญาณเตือนที่ผิดพลาดมีค่าระหว่าง 0.0 ถึง 0.1 ซึ่งแสดงให้เห็นถึงความถูกต้องในการตัดสินใจจะลดลงเมื่อ

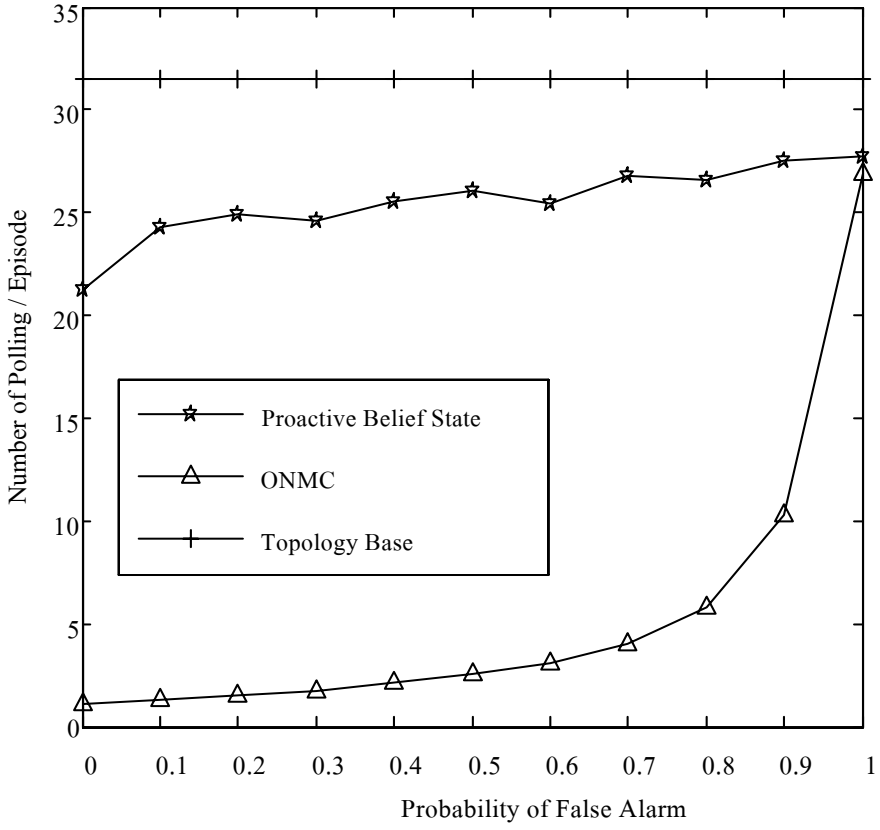
สิ่งที่ได้จากการสังเกตมีความไม่ชัดเจนเกิดขึ้น และวิธีคำนึงถึงโครงรูปโครงข่ายจะให้ผลรางวัลสะสมต่อเอพพิโซดที่ลดลงเป็นอย่างมากเมื่อสิ่งที่ได้จากการสังเกตมีความไม่ชัดเจนเพิ่มมากขึ้น ซึ่งหมายถึงการไม่ส่งผลดีต่อการเฝ้าตรวจสถานะของโครงข่ายในกรณีที่สิ่งที่ได้จากการสังเกต มีความไม่ชัดเจนเพิ่มมากขึ้น



รูปที่ 4.6 ผลรางวัลสะสมต่อเอพพิโซดต่อความน่าจะเป็นของการเกิดสัญญาณเตือนที่ผิดพลาด

ผลการจำลองแบบในส่วนของจำนวนการโพลล์ที่เกิดขึ้นดังรูปที่ 4.7 โดยที่แกนนอนจะแทนความไม่ชัดเจนของสิ่งที่ได้จากการสังเกต โดยที่จุด 0.0 จะหมายถึงสิ่งที่ได้จากการสังเกตถูกต้องทั้งหมดและความไม่ชัดเจนของสิ่งที่ได้จากการสังเกตจะเพิ่มขึ้นเรื่อยๆ จนถึงจุด 1.0 ที่หมายถึงสิ่งที่ได้จากการสังเกตไม่ถูกต้องทั้งหมด และแกนตั้งคือจำนวนการโพลล์ที่เกิดขึ้นในการค้นหาจุดที่อุปกรณ์ขัดข้องหนึ่งครั้ง จากผลการจำลองแบบพบว่าวิธีออนโพลิซีมอนติคาร์โลจะมีจำนวนการโพลล์ที่น้อยกว่า ซึ่งหมายถึงจะมีปริมาณโพลล์ลิ่งโอเวอร์เฮดที่ต่ำกว่าด้วย โดยที่วิธีออน

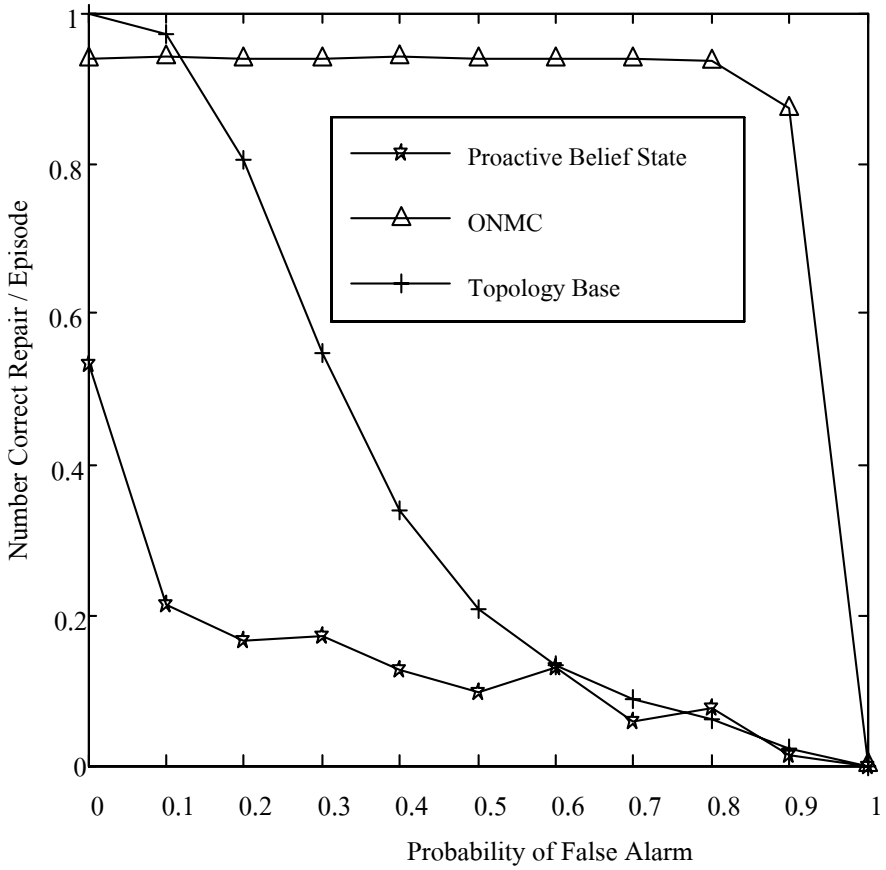
โพลิซีมอนติคาร์โล สามารถลดปริมาณโพลล์ลิ่งโอเวอร์เฮดได้ถึง 88 % เมื่อเปรียบเทียบกับวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ และ 15 % ถึง 94 % เมื่อเปรียบเทียบกับวิธีคำนึงถึงโครงรูปโครงข่าย



รูปที่ 4.7 ความสัมพันธ์ระหว่างจำนวนครั้งของการโพลล์ต่อเอพพิโซด

เมื่อพิจารณาในด้านของความถูกต้องในการแจ้งตำแหน่งที่อุปกรณ์ขัดข้อง ซึ่งเป็นค่าชี้วัดประสิทธิภาพของการเฝ้าตรวจสถานะของโครงข่าย โดยทำการเปรียบเทียบจำนวนการแจ้งตำแหน่งที่อุปกรณ์ขัดข้องได้ถูกตำแหน่งต่อเอพพิโซด ระหว่างวิธีออนโพลิซีมอนติคาร์โลวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์และวิธีคำนึงถึงโครงรูปโครงข่าย ดังรูปที่ 4.8 พบว่าที่ค่าความน่าจะเป็นของการเกิดสัญญาณเตือนที่ผิดพลาดมีค่า 0.0 หรือกรณีที่สิ่งที่ได้จากการสังเกตถูกต้องทั้งหมด วิธีคำนึงถึงโครงรูปโครงข่ายสามารถแจ้งตำแหน่งที่อุปกรณ์ขัดข้องได้ถูกต้องมากที่สุด เนื่องจากวิธีการนี้เป็นการทำงานโดยการส่งสัญญาณการโพลล์ไปที่โหนดทุกๆ โหนดจึงสามารถตรวจพบจุดที่อุปกรณ์ขัดข้องได้อย่างถูกต้อง แต่เมื่อความน่าจะเป็นของการเกิดสัญญาณเตือนที่ผิดพลาดมีค่าเพิ่มขึ้น กลับมีจำนวนการแจ้งตำแหน่งที่อุปกรณ์ขัดข้องได้ถูกตำแหน่งต่อเอพพิโซดลดลงเป็นอย่าง

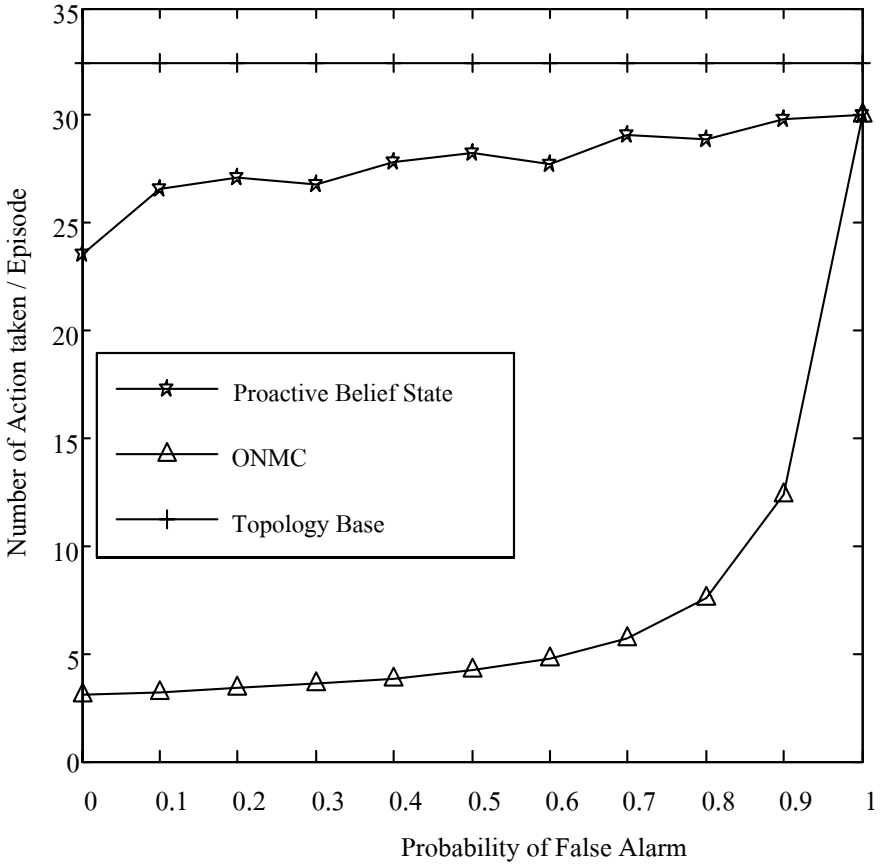
มากต่างจากวิธีออนโพลิซีมอนติคาร์โล ที่สามารถแจ้งตำแหน่งที่อุปกรณ์ขัดข้องได้ถูกตำแหน่งต่อเอพพิโซดอยู่ในระดับที่สูงกว่าทั้งสองขั้นตอนวิธี โดยที่ความน่าจะเป็นของการเกิดสัญญาณเตือนที่ผิดพลาดตั้งแต่ 0.2 เป็นต้นไปวิธีออนโพลิซีมอนติคาร์โลยังคงให้ความถูกต้องที่ดีกว่าอย่างชัดเจน โดยวิธีออนโพลิซีมอนติคาร์โล จะมีความถูกต้องมากกว่าวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์และวิธีคำนึงถึงโครงรูปโครงข่ายถึง 80 %



รูปที่ 4.8 ความสัมพันธ์ระหว่างจำนวนครั้งที่ซ่อมได้ถูกต้องต่อเอพพิโซด

เมื่อทำการเปรียบเทียบให้เห็นถึงจำนวนการกระทำที่ต้องใช้ในการวิเคราะห์หาจุดขัดข้องในแต่ละเอพพิโซด เปรียบเทียบระหว่างวิธีออนโพลิซีมอนติคาร์โลวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์และวิธีคำนึงถึงโครงรูปโครงข่าย จากผลการจำลองแบบดังรูปที่ 4.9 โดยทำการจำลองแบบที่ความน่าจะเป็นของการเกิดสัญญาณเตือนที่ผิดพลาดมีค่าตั้งแต่ 0.0 จนถึง 1.0 พบว่าการใช้วิธีออนโพลิซีมอนติคาร์โล สามารถลดจำนวนของการกระทำที่ใช้ในการค้นหาตำแหน่งที่อุปกรณ์ขัดข้องในแต่ละครั้งลงได้ถึง 88 % เมื่อเปรียบเทียบกับวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ และ 12 % ถึง

87 % เมื่อเปรียบเทียบกับวิธีคำนึงถึงโครงรูปโครงข่าย ซึ่งตรงกับความต้องการที่จะลดโพลล์ลิ่งโอเวอร์เฮดของระบบโครงข่ายลง



รูปที่ 4.9 ความสัมพันธ์ระหว่างจำนวนครั้งของการกระทำต่อเอพพิโซด

4.5 การวิเคราะห์ผลการจำลองของแบบจำลองของโครงข่ายขนาดใหญ่

การเปรียบเทียบผลรางวัลสะสมต่อเอพพิโซดที่ความน่าจะเป็นของการเกิดสัญญาณเตือนที่ผิดพลาดที่ค่าเริ่มที่ 0.0 และเพิ่มขึ้นจนถึง 1.0 วิธีออนโพลิซีมอนติคาร์โลจะมีผลรางวัลสะสมต่อเอพพิโซดที่สูงกว่า เนื่องจากตัวกระทำการตัดสินใจของวิธีออนโพลิซีมอนติคาร์โลสามารถที่จะทำการตัดสินใจในการเลือกการกระทำที่ถูกต้องได้มากกว่า จึงส่งผลให้ผลรางวัลสะสมต่อเอพพิโซดมีค่าที่สูงกว่า

การเปรียบเทียบจำนวนของการโพลล์ที่เกิดขึ้น ที่ความน่าจะเป็นของการเกิดสัญญาณเตือนที่ผิดพลาดเริ่มที่ 0.0 และเพิ่มขึ้นเรื่อยๆ จนถึง 1.0 พบว่าวิธีออนโพลิซีมอนติคาร์โลจะมีปริมาณการโพลล์ที่ต่ำกว่าในทุกๆ ค่าของการเกิดสัญญาณเตือนที่ผิดพลาด

การเปรียบเทียบจำนวนการแจ้งตำแหน่งที่โหนดขัดข้องได้ถูกตำแหน่งต่อเอพพิโซด ซึ่งเป็นค่าชี้วัดประสิทธิภาพของการเฝ้าตรวจสถานะของโครงข่าย ในส่วนของการค้นหาตำแหน่งที่อุปกรณ์ขัดข้อง พบว่าวิธีออนโพลิซีมอนติคาร์โลมีความสามารถในการระบุตำแหน่งของอุปกรณ์ขัดข้องได้ถูกต้องมากกว่าวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์

การเปรียบเทียบจำนวนการกระทำที่ต้องใช้ในการวิเคราะห์หาจุดขัดข้อง ในแต่ละเอพพิโซด ซึ่งจากผลการจำลองแบบพบว่าการใช้วิธีออนโพลิซีมอนติคาร์โลสามารถลดปริมาณของการกระทำได้มากกว่าวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์

เมื่อพิจารณาหน่วยความจำที่ต้องใช้ในการทำงานของขั้นตอนวิธี ของวิธีออนโพลิซีมอนติคาร์โลเปรียบเทียบกับวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ โดยใช้ขั้นตอนวิธีตามหัวข้อ 2.5 และหัวข้อ 2.6 ตามลำดับ พบว่าขั้นตอนการทำงานของวิธีออนโพลิซีมอนติคาร์โลจะทำการเก็บแอคชั่นแวลู่ของทุกๆ คู่ของสิ่งที่ได้จากการสังเกตและการกระทำ (observation-action pair) จากการจำลองแบบประกอบด้วยสิ่งที่ได้จากการสังเกตทั้งหมดเท่ากับ 31 สถานะ และการกระทำทั้งหมดเท่ากับ 61 การกระทำ ดังนั้นจึงมีจำนวนพารามิเตอร์ทั้งหมดเท่ากับ 31 x 61 หรือเท่ากับ 1,891 พารามิเตอร์ ในส่วนของวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ จะมีจำนวนพารามิเตอร์ที่ประกอบด้วยพารามิเตอร์ของ $Q(s,a), e(s,a), T(s,a,s'), O(s',a,z'), b(s)$ โดยในการจำลองแบบ s, s', b เท่ากับ 31 สถานะ a คือการกระทำทั้งหมดเท่ากับ 61 การกระทำ และ z เท่ากับ 2 สถานะ ดังนั้นจึงมีจำนวนพารามิเตอร์ทั้งหมดเท่ากับ (31 x 61) + (31 x 61) + (31 x 61 x 31) + (31 x 61 x 2) + 31 เท่ากับ 66,216 พารามิเตอร์ ส่วนวิธีคำนึงถึงโครงรูปโครงข่ายโดยใช้ขั้นตอนการทำงานตามหัวข้อ 4.3.3 ซึ่งขึ้นอยู่กับจำนวนอุปกรณ์ที่ต้องการเฝ้าตรวจสถานะและจำนวนครั้งในการโพลล์ซ้ำ วิธีการนี้ใช้การตรวจสอบที่สถานะของอุปกรณ์โดยตรงจึงต้องการหน่วยความจำเท่ากับจำนวนอุปกรณ์คือเท่ากับ 30 อุปกรณ์คูณด้วยจำนวนครั้งในการโพลล์ซ้ำ 3 ครั้ง เท่ากับ 90 พารามิเตอร์ หรือเท่ากับ $retry|N|$ เมื่อ $retry$ คือจำนวนครั้งในการโพลล์ซ้ำ $|N|$ คือจำนวนอุปกรณ์ที่ต้องการเฝ้าตรวจสถานะ แต่เนื่องจากการทำงานของวิธีนี้อาศัยโครงรูปของโครงข่ายเป็นหลักแต่มิได้นำเสนอในส่วนของการทำงานเพื่อกำหนดโครงรูปของโครงข่าย ซึ่งเป็นขั้นตอนหนึ่งที่มีความยุ่งยากและต้องใช้หน่วยความจำเพื่อเรียนรู้โครงรูปของโครงข่ายเป็นจำนวนมาก จึงไม่นำมาเปรียบเทียบในการทดลองนี้

หากกำหนดว่าในหนึ่งพารามิเตอร์เป็นตัวแปรชนิดดับเบิล (double) ซึ่งต้องใช้หน่วยความจำในการเก็บข้อมูลขนาด 8 ไบต์ (byte) ดังนั้นในส่วนของขั้นตอนวิธีแบบออนโพลิซีมอนติคาร์โลจะใช้หน่วยความจำในการทำงานเท่ากับ 1,891 คูณ 8 ซึ่งเท่ากับ 15,128 ไบต์ ในส่วนของขั้นตอนวิธี

แบบโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ จะใช้หน่วยความจำในการทำงานเท่ากับ 66,216 คูณ 8 ซึ่งเท่ากับ 529,728 ไบต์ ในส่วนของวิธีคำนึงถึงโครงรูปโครงข่ายนั้นจะใช้หน่วยความจำในการทำงานเท่ากับ 90 คูณ 8 ซึ่งเท่ากับ 72 ไบต์ ซึ่งยังไม่รวมกับใส่วนที่ต้องใช้ในส่วนของการทำงานเพื่อกำหนดโครงรูปของโครงข่ายจึงไม่นำมาเปรียบเทียบ ดังนั้นจึงเห็นได้ชัดเจนว่าในกรณีที่โครงข่ายมีขนาดใหญ่ วิธีออนโพลิซีมอนติคาร์โลใช้หน่วยความจำในการทำงานน้อยกว่าวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ เช่นเดียวกับการจำลองแบบในโครงข่ายขนาดเล็ก

4.6 สรุป

จากผลการจำลองแบบด้วยวิธีออนโพลิซีมอนติคาร์โล เปรียบเทียบกับวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์และวิธีคำนึงถึงโครงรูปโครงข่ายในโครงข่ายขนาดใหญ่ โดยทำการเปรียบ เทียบค่าชี้วัดต่างๆ และได้ผลสรุปดังนี้

การเปรียบเทียบผลรางวัลสะสมต่อเอพพิโซด พบว่าวิธีออนโพลิซีมอนติคาร์โล มีผลรางวัลสะสมต่อเอพพิโซดที่สูงกว่าโดยตลอด

การเปรียบเทียบจำนวนของการโพลล์ที่เกิดขึ้น ซึ่งพบว่าวิธีออนโพลิซีมอนติคาร์โลจะมีจำนวนการโพลล์ที่ต่ำกว่าในทุกๆ ค่าของการเกิดสัญญาณเตือนที่ผิดพลาดและสามารถลดปริมาณการโพลล์ถึง 88 % เมื่อเปรียบเทียบกับวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ และ 15 % ถึง 94 % เมื่อเปรียบเทียบกับวิธีคำนึงถึงโครงรูปโครงข่าย

การเปรียบเทียบจำนวนการแจ้งตำแหน่งที่อุปกรณ์ขัดข้องได้ถูกตำแหน่ง ต่อเอพพิโซด ซึ่งพบว่า วิธีออนโพลิซีมอนติคาร์โล มีความสามารถในการระบุตำแหน่งของอุปกรณ์ที่ขัดข้องได้แม่นยำมากกว่าวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์และวิธีคำนึงถึงโครงรูปโครงข่ายถึง ได้ถึง 80 %

การเปรียบเทียบจำนวนการกระทำที่ต้องใช้ในการวิเคราะห์หาจุดขัดข้อง ในแต่ละเอพพิโซด พบว่าการใช้วิธีออนโพลิซีมอนติคาร์โล สามารถลดปริมาณของการกระทำลงได้มากกว่าวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ถึง 88 % เมื่อเปรียบเทียบกับวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ และ 12 % ถึง 87 % เมื่อเปรียบเทียบกับวิธีคำนึงถึงโครงรูปโครงข่าย

การเปรียบเทียบปริมาณหน่วยความที่ใช้ในการเก็บข้อมูล โดยกำหนดว่าในหนึ่งพารามิเตอร์เป็นตัวแปรชนิดดับเบิล (double) ซึ่งต้องใช้หน่วยความจำในการเก็บข้อมูลขนาด 8 ไบต์ (byte) ในการทำงานของขั้นตอนวิธีของขั้นตอนวิธีแบบออนโพลิซีมอนติคาร์โล จะใช้หน่วยความจำในการ

ทำงานเท่ากับ 15,128 ไบต์ ส่วนของขั้นตอนวิธีแบบโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ จะใช้หน่วยความจำในการทำงานเท่ากับ 529,728 ไบต์ ดังนั้นในกรณีที่โครงข่ายมีขนาดใหญ่ วิธีออนโพลิซีมอนติคาร์โลต้องการหน่วยความจำที่ใช้ในการเก็บข้อมูลน้อยกว่าวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์

บทที่ 5
สรุปผลการวิจัยและข้อเสนอแนะ

งานวิจัยวิทยานิพนธ์นี้มีจุดประสงค์ในการพยายามลดโพลล์ลิ่งโอเวอร์เฮด ที่เกิดจากระบบบริหารจัดการโครงข่ายในส่วนของการเฝ้าตรวจสถานะของโครงข่าย และการค้นหาตำแหน่งที่อุปกรณ์ขัดข้อง ด้วยการนำเสนอวิธีรีอินฟอร์สเมนท์เลิร์นนิ่งแบบออนโพลิซีมอนติคาร์โล ซึ่งเป็นวิธีการหนึ่งของวิธีรีอินฟอร์สเมนท์เลิร์นนิ่ง ที่สามารถเรียนรู้วิธีการตัดสินใจที่ดี ในสภาวะแวดล้อมที่มีข้อมูลอยู่เพียงบางส่วนและมีการดำเนินไปของกระบวนการ ในลักษณะเป็นเอพพิโซด ซึ่งตรงกับลักษณะการทำงานของการเฝ้าตรวจสถานะของโครงขาย

5.1 สรุปผลการวิจัยในการทดลองกับโครงข่ายขนาดเล็ก

ในการจำลองแบบด้วยโครงข่ายขนาดเล็กโดยได้ทำการจำลองแบบด้วยวิธีรีอินฟอร์สเมนท์เลิร์นนิ่ง แบบออนโพลิซีมอนติคาร์โล เปรียบเทียบกับวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ เพื่อทำการเปรียบเทียบค่าชี้วัดต่างๆ ซึ่งพบว่าวิธีออนโพลิซีมอนติคาร์โล จะมีปริมาณการโพลล์ที่ต่ำกว่าในทุกๆ ค่าของการเกิดสัญญาณเตือนที่ผิดพลาด และสามารถลดปริมาณการโพลล์ได้ได้ระหว่าง 33 % ถึง 86 % ซึ่งตรงกับความต้องการที่จะลดโพลล์ลิ่งโอเวอร์เฮดของระบบโครงข่ายลง

เมื่อพิจารณาปริมาณหน่วยความที่ใช้ในการเก็บข้อมูลในการทำงานของขั้นตอนวิธี ของวิธีออนโพลิซีมอนติคาร์โล เปรียบเทียบกับ วิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ โดยใช้ขั้นตอนวิธีตามหัวข้อ 2.5 และหัวข้อ 2.6 ตามลำดับ พบว่า ขั้นตอนการทำงานของวิธีออนโพลิซีมอนติคาร์โล จะทำการเก็บแอคชั่นแวลู่ของทุกๆ คู่ของสิ่งที่ได้จากการสังเกตและการกระทำ (observation-action pair) ซึ่งในการจำลองแบบประกอบด้วยสิ่งที่ได้จากการสังเกตทั้งหมดเท่ากับ 7 สถานะและการกระทำทั้งหมดเท่ากับ 19 การกระทำดังนั้นจึงมีจำนวนพารามิเตอร์ทั้งหมดเท่ากับ 7 x 19 หรือเท่ากับ 133 พารามิเตอร์ ในส่วนของวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์จะมีจำนวนพารามิเตอร์ที่ประกอบด้วยพารามิเตอร์ของ $Q(s,a)$, $e(s,a)$, $T(s,a,s')$, $O(s',a,z')$, $b(s)$ โดยในการจำลองแบบ s,s',b เท่ากับ 7 สถานะ a คือการกระทำทั้งหมดเท่ากับ 19 การกระทำ และ z เท่ากับ 2 สถานะ ดังนั้นจึงมีจำนวนพารามิเตอร์ทั้งหมดเท่ากับ (7 x 19) + (7 x 19) + (7 x 19 x 7) + (7 x 19 x 2) + 7 เท่ากับ 1,470 พารามิเตอร์ ซึ่งสรุปได้ว่าวิธีออนโพลิซีมอนติคาร์โลต้องการหน่วยความที่ใช้ในการเก็บข้อมูลน้อยกว่าวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์

หากกำหนดว่าในหนึ่งพารามิเตอร์เป็นตัวแปรชนิดดับเบิล ที่ใช้หน่วยความจำในการเก็บข้อมูลขนาด 8 ไบต์ ขั้นตอนวิธีแบบออนโพลิซีมอนติคาร์โล จะใช้หน่วยความจำในการทำงานเท่ากับ 1,064 ไบต์ และขั้นตอนวิธีแบบโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ จะใช้หน่วยความจำในการทำงานเท่ากับ 11,760 ไบต์ ดังนั้นในกรณีที่โครงข่ายขนาดเล็ก วิธีออนโพลิซีมอนติคาร์โล ต้องการหน่วยความจำที่ใช้ในการเก็บข้อมูลน้อยกว่า วิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์

5.2 สรุปผลการวิจัยในการทดลองกับโครงข่ายขนาดใหญ่

จากการจำลองแบบในโครงข่ายขนาดใหญ่ โดยทำการเปรียบเทียบให้เห็นถึงจำนวนการกระทำที่ต้องใช้ในการวิเคราะห์หาจุดขัดข้องในแต่ละเอพพิโซด เปรียบเทียบระหว่างวิธีออนโพลิซีมอนติคาร์โลวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์และวิธีคำนึงถึงโครงรูปโครงข่าย จากผลการจำลองแบบพบว่าการใช้วิธีออนโพลิซีมอนติคาร์โล สามารถลดปริมาณการโพลล์ได้ถึง 88 % ซึ่งตรงกับความต้องการที่จะลดโพลล์ลิ่งโอเวอร์เฮดของระบบโครงข่าย

เมื่อพิจารณาหน่วยความจำที่ต้องใช้ในการทำงานของขั้นตอนวิธี ของวิธีออนโพลิซีมอนติคาร์โลเปรียบเทียบกับวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ โดยใช้ขั้นตอนวิธีตามหัวข้อ 2.5 และหัวข้อ 2.6 ตามลำดับ พบว่าขั้นตอนการทำงานของวิธีออนโพลิซีมอนติคาร์โลจะทำการเก็บแอคชั่นแวลู่ของทุกๆ คู่ของสิ่งที่ได้จากการสังเกตและการกระทำ จากการจำลองแบบประกอบด้วยสิ่งที่ได้จากการสังเกตทั้งหมดเท่ากับ 31 สถานะ และการกระทำทั้งหมดเท่ากับ 61 การกระทำ ดังนั้นจึงมีจำนวนพารามิเตอร์ทั้งหมดเท่ากับ 31 x 61 หรือเท่ากับ 1,891 พารามิเตอร์ ในส่วนของวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์จะมีจำนวนพารามิเตอร์ที่ประกอบด้วย พารามิเตอร์ของ $Q(s,a)$, $e(s,a)$, $T(s,a,s')$, $O(s',a,z')$, $b(s)$ โดยในการจำลองแบบ s,s',b เท่ากับ 31 สถานะ a คือการกระทำทั้งหมดเท่ากับ 61 การกระทำ และ z เท่ากับ 2 สถานะ ดังนั้นจึงมีจำนวนพารามิเตอร์ทั้งหมดเท่ากับ (31 x 61) + (31 x 61) + (31 x 61 x 31) + (31 x 61 x 2) + 31 เท่ากับ 66,216 พารามิเตอร์ ส่วนวิธีคำนึงถึงโครงรูปโครงข่ายโดยใช้ขั้นตอนการทำงานตามหัวข้อ 4.3.3 ซึ่งขึ้นอยู่กับ จำนวนอุปกรณ์ที่ต้องการเฝ้าตรวจสถานะและจำนวนครั้งในการโพลล์ซ้ำ วิธีการนี้ใช้การตรวจสอบที่สถานะของอุปกรณ์โดยตรง จึงต้องการหน่วยความจำเท่ากับจำนวนอุปกรณ์คือเท่ากับ 30 อุปกรณ์คูณด้วยจำนวนครั้งในการโพลล์ซ้ำ 3 ครั้ง เท่ากับ 90 พารามิเตอร์ แต่เนื่องจากการทำงานของวิธีนี้อาศัยโครงรูปโครงข่ายเป็นหลัก แต่มิได้นำเสนอในส่วนของการทำงานเพื่อเรียนรู้โครงรูปโครงข่าย (Han, Ahn, and Chung, 2001) ซึ่งเป็นขั้นตอนหนึ่งที่มีความยุ่งยาก และต้องใช้หน่วยความจำในการกำหนดโครงรูปโครงข่ายเป็นจำนวนมาก จึงไม่นำมาเปรียบเทียบในการทดลองนี้

หากกำหนดว่าในหนึ่งพารามิเตอร์เป็นตัวแปรชนิดดับเบิล ที่ใช้หน่วยความจำในการเก็บข้อมูลขนาด 8 ไบต์ ดังนั้นขั้นตอนวิธีแบบออนโพลิซีมอนติคาร์โลจะใช้หน่วยความจำในการทำงานเท่ากับ 15,128 ไบต์ ในส่วนของขั้นตอนวิธีแบบโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์จะใช้หน่วยความจำในการทำงานเท่ากับ 529,728 ไบต์ ดังนั้นในกรณีที่โครงข่ายมีขนาดใหญ่วิธีออนโพลิซีมอนติคาร์โลใช้หน่วยความจำในการทำงานน้อยกว่าวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ เช่นเดียวกับการจำลองแบบในโครงข่ายขนาดเล็ก

เมื่อพิจารณาปริมาณหน่วยความจำที่ใช้ในการทำงานของขั้นตอนวิธี พบว่าวิธีออนโพลิซีมอนติคาร์โลจะทำการเก็บแอคชั่นแวลู่ของทุกๆ คู่ของ สิ่งที่ได้จากการสังเกตและการกระทำ หรือเท่ากับ $|O||A|$ เมื่อ $|O|$ คือขนาดของสิ่งที่ได้จากการสังเกตที่เป็นไปได้ทั้งหมด และ $|A|$ คือขนาดของการกระทำที่เป็นไปได้ทั้งหมด (Usaha, 2004) ในขณะที่วิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์จะต้องใช้จำนวนพารามิเตอร์ในการคำนวณเท่ากับ $|S||A|+|S||A|+|S||A||S|+|S||A||Z|+|S|$ เมื่อ $|S|$ คือขนาดของสิ่งที่ได้จากการสังเกตที่เป็นไปได้ทั้งหมด $|A|$ คือขนาดของการกระทำที่เป็นไปได้ และ $|Z|$ คือขนาดของสถานะ ซึ่งเมื่อเปรียบเทียบแล้วจะพบว่าวิธีออนโพลิซีมอนติคาร์โล ใช้ปริมาณหน่วยความจำในการเก็บข้อมูลที่น้อยกว่าวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์

ส่วนวิธีคำนึงถึงโครงรูปโครงข่าย ขึ้นอยู่กับจำนวนอุปกรณ์ที่ต้องการเฝ้าตรวจสถานะและจำนวนครั้งในการโพลล์ซ้ำ หรือเท่ากับ $retry|N|$ เมื่อ $retry$ คือจำนวนครั้งในการโพลล์ซ้ำ $|N|$ คือจำนวนอุปกรณ์ที่ต้องการเฝ้าตรวจสถานะ แต่วิธีการนี้ต้องใช้หน่วยความจำในการกำหนดโครงรูปโครงข่ายเป็นจำนวนมากจึงไม่นำมาเปรียบเทียบในการทดลองนี้

เมื่อพิจารณาความซับซ้อนของการคำนวณในการตัดสินใจเลือกการกระทำพบว่า วิธีออนโพลิซีมอนติคาร์โลใช้การอ่านค่าจากตาราง $Q(o,a)$ โดยตรง ในขณะที่วิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์ต้องใช้การคำนวณจาก $\sum_s Q(s,a)b(s)$ ซึ่งในการตัดสินใจแต่ละครั้งจะต้องทำการคำนวณให้เสร็จสิ้นก่อนแล้วจึงกระทำการตัดสินใจ ส่วนวิธีคำนึงถึงโครงรูปโครงข่ายนั้นต้องใช้การคำนวณในการกำหนดโครงรูปโครงข่าย ซึ่งมิได้นำเสนอในส่วนนี้ (Han, Ahn, and Chung, 2001) จึงไม่นำมาเปรียบเทียบในการทดลองนี้ ดังนั้นเมื่อเปรียบเทียบความซับซ้อนของขั้นตอนวิธีแล้วพบว่าวิธีออนโพลิซีมอนติคาร์โลมีความซับซ้อนของการคำนวณน้อยกว่าวิธีโปรแอคทีฟเน็ตเวิร์คมาเนจเมนท์

5.3 ปัญหาและข้อเสนอแนะ

ในการศึกษาโพลล์ลิ่งโอเวอร์เฮด ซึ่งส่งผลกระทบโดยตรงต่อประสิทธิภาพของโครงข่าย แต่ถึงอย่างไรก็ตาม วิธีการโพลล์ยังคงเป็นวิธีที่จำเป็นต้องใช้ในการเฝ้าตรวจสถานะของโครงข่าย

ซึ่งปัญหาของโพลล์ลิ่งโอเวอร์เฮดนี้เป็นปัญหาในเชิงของปริมาณที่เหมาะสม ที่ชี้ชัดหรือระบุจำนวนที่ชัดเจนได้ยาก เนื่องจากปริมาณการโพลล์จะสัมพันธ์กับความถูกต้องและความรวดเร็วของการเฝ้าตรวจสถานะของโครงข่าย

การทำงานของวิธีออนโพลิซีมอนติคาร์โล ซึ่งต้องทำการเก็บแอคชั่นแวลู่ของทุกๆ คู่ของสิ่งที่ได้จากการสังเกตและการกระทำซึ่งอาจอยู่ในรูปของตาราง ดังนั้นเมื่อโครงข่ายมีขนาดที่ใหญ่มากๆ ย่อมทำให้ต้องใช้พื้นที่ในการเก็บข้อมูลจำนวนมากและทำให้ตารางมีขนาดใหญ่ จึงอาจเป็นการไม่สะดวกที่จะทำการค้นหาค่าคิวแวลู่จากตารางที่มีขนาดใหญ่นี้

5.4 งานวิจัยในอนาคต

นอกจากวิธีการต่างๆ ที่นำเสนอในวิทยานิพนธ์นี้ยังอาจมีวิธีการอื่นๆ ที่อาจนำมาประยุกต์ใช้กับการเฝ้าตรวจสถานะของโครงข่าย เช่น

5.4.1 กรณีที่โครงข่ายมีขนาดใหญ่และประกอบด้วยอุปกรณ์โครงข่ายจำนวนมาก ซึ่งการใช้วิธีเก็บผลรางวัลในทุกๆ คู่ของสถานะและการกระทำอาจส่งผลให้เกิดการทำงานที่ล่าช้าจึงอาจใช้วิธีการประมาณค่าในการทำงานแทน เช่น การใช้ขั้นตอนวิธีแบบแอคเตอร์คริติคบีลีฟสเตต (Actor-Critic Belief State หรือ ACBS) ซึ่งเป็นวิธีรีอินฟอร์สเมนท์เลิร์นนิ่ง แบบหนึ่งที่ใช้การประมาณค่าในการทำงาน แทนการอ่านค่าทั้งหมดจากตาราง จึงอาจทำงานได้ดีในกรณีที่ระบบโครงข่ายมีขนาดที่ใหญ่ โดยมีผู้นำมาใช้ในการแก้ปัญหาด้านต่างๆ มาก่อนจึงอาจนำมาประยุกต์ใช้กับการบริหารจัดการโครงข่ายได้เช่นกัน รวมทั้งกรณีที่มีอุปกรณ์ขัดข้องในเวลาเดียวกันหลายอุปกรณ์ ซึ่งอาจจำเป็นที่จะต้องทำการปรับเปลี่ยนการนิยามปัญหาหรือการพัฒนาขั้นตอนวิธีที่เหมาะสมยิ่งขึ้น

5.4.2 ในโครงข่ายขนาดใหญ่ที่ประกอบด้วยอุปกรณ์โครงข่ายจำนวนมาก ซึ่งต้องแบ่งโครงข่ายออกเป็นโครงข่ายย่อยๆ และต้องติดตั้งตัวเฝ้าตรวจสถานะของโครงข่ายหลายๆ ตำแหน่ง ซึ่งการศึกษาถึงตำแหน่งที่เหมาะสม ในการจัดวางตัวเฝ้าตรวจสถานะของโครงข่าย ที่จะก่อให้เกิดโพลล์ลิ่งโอเวอร์เฮดต่ำ และสามารถค้นหาตำแหน่งที่อุปกรณ์ขัดข้องได้อย่างรวดเร็ว รวมทั้งทำการศึกษาถึงการแลกเปลี่ยนข้อมูลของตัวเฝ้าตรวจสถานะของโครงข่ายด้วยกันเอง เพื่อการทำงานเป็นกลุ่มในการแบ่งเบาภาระงานหรือสามารถทำงานทดแทนกันได้

5.4.3 การเพิ่มความสามารถในการเรียนรู้ เพื่อให้สามารถที่จะทำการปรับปรุงการเรียนรู้ตามโครงรูปของโครงข่ายที่เปลี่ยนแปลงไปได้ ซึ่งจะเป็นประโยชน์ในกรณีที่มีการเปลี่ยนแปลงโครงรูปของโครงข่ายซึ่งไม่จำเป็นต้องสร้างการเรียนรู้ใหม่ทั้งหมด

5.4.4 การนำมาใช้งานในระบบโครงข่ายจริง เช่นการนำมาใช้ร่วมกับโปรโทคอล SNMP ซึ่งเป็นโปรโคคอลพื้นฐานและมีการใช้งานอย่างแพร่หลาย รวมทั้งการพัฒนาให้เกิดความเหมาะสม

ในกรณีที่มีการเฝ้าตรวจสถานะโครงข่ายในระยะใกล เช่น การเฝ้าตรวจสถานะผ่านโครงข่ายอินเทอร์เน็ตหรือโครงข่ายที่ใช้กับระบบสัญญาณเตือนภัยที่ติดตั้งในทะเลหรือบนภูเขาสูง ซึ่งหากสามารถทำงานร่วมกับซอฟแวร์ที่มีใช้งานอยู่ในปัจจุบันจะเป็นการเพิ่มประสิทธิภาพ ในการตรวจสอบสถานะของโครงข่ายให้ดียิ่งขึ้น

รายการอ้างอิง

He, Q. (2003). **Proactive Network Management : Reinforcement learning Approach.** PhD Thesis University of Maryland.

Usaha, W. (2004). **Resource Allocation in Networks with Dynamic Topology**. PhD Thesis,University of London, London, U.K.

Marcia, Z., and Bruce B.L.(2004). Using Passive Traces of Application Traffic in a NetworkMonitoring System. **High performance Distributed Computing. Proceedings. IEEE International Symposium**. 13: 77-86

Yuri B., Feodor D., and Hassan G. (2004). Effective Network Monitoring. **Computer Communications and Networks Proceedings International Conference**. 13: 394-399

Xiaojiang, D. (2004). Toward Efficient Distributed Network Monitoring. **Performance, Computing, and Communications IEEE International Conference.** (pp. 87 – 94)

Yoshihara, k.,Sugiyama, k.,Horiuchi, h., and Obana, S.(1999).Dynamic polling scheme based on time variation of network management information values. **Integrated Network Management. Distributed Management for the Networked Millennium. Proceedings of the Sixth IFIP/IEEE International Symposium**. (pp. 141 – 154)

Sutton, R.S., and Barto, A.G.(1998). **Reinforcement Learning: An introduction**. The MIT Press.

Alexandri, E., Martinez, G., and Zeghlache, D.(2002).A distributed reinforcement learning approach to maximize resource utilization and control handover dropping in multimedia wireless networks. **Personal, Indoor and Mobile Radio Communications. The IEEE International Symposium**. 13: 2249–2253

Goldenberg, D.K., Krishnamurthy, A., Maness, W.C., Yang, Y.R., Young, A., Morse, A.S., and Savvides, A.(2005). Network localization in partially localizable networks. **INFOCOM. Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE.** 24: 313-326

Sreedhar, R., Hill, T.D., and Stanley, G.M.(2000). Intelligent fault management for large networks. **Network Operations and Management Symposium. IEEE/IFIP.** (pp. 959 – 960)

Steinder, M., and Sethi, A.S.(2004). A Survey of Fault Localization Techniques in Computer Networks. **Science of Computer Programming, Special Edition on Topics in System Administration.** (pp. 165-194)

Han, J-S.,Ahn, S-J., and Chung, J-W.(2001). A new approach of polling efficiency based on network topology. **International Journal of Network Management.** 11(4): 243-251

Anthony, V- E., and Robert J-W. (2005).Fault Management: A Functional View of Root Cause Analysis and Correlation. Tavve software Company. [online]Available: http://www.tavve.com/EW_White_Paper.pdf

Vishnu, A., Mamidala, A.R., Hyun-Wook Jin., and Panda, D.K.(2005) .Performance Modeling of Subnet Management on Fat Tree InfiniBand Networks using OpenSM. **Parallel and Distributed Processing Symposium Proceedings. IEEE International.** 19.

Zhu, Y., Chen, T., and Liu, S.(2001), Models and analysis of trade-offs in distributed network management approaches. **Integrated Network Management Proceedings IEEE/IFIP International Symposium**. (pp. 391 – 404)

Bouloutas A.T., Calo S.B., Finkel A., and Katzela I.(1995) Distributed fault identification in telecommunication networks. **Journal of Network and Systems Management.** 3(3):295-312

Yow-Jian Lin and Mun Choon Chan (2002).A Scalable Monitoring Approach Based on Aggregation and Refinement.**IEEE journal on seleced areas in communications.** 20: 677-690

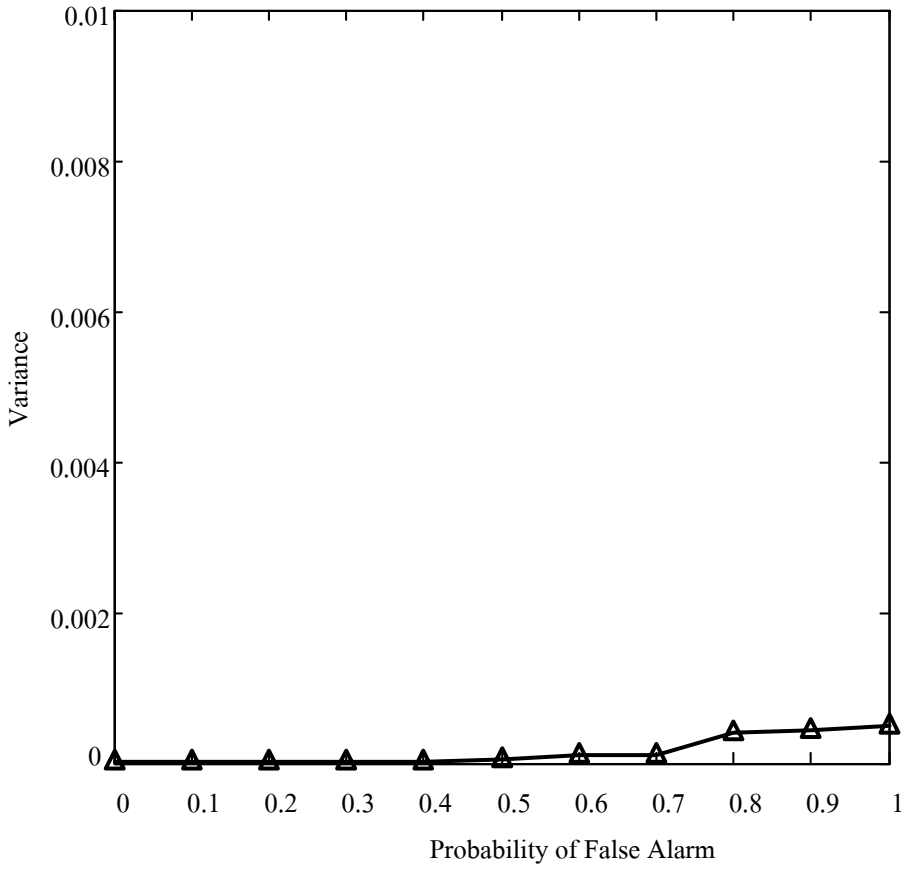
Ming-Shan Su.(2002) .**Multilevel distributed diagnosis and a design of a distributed network fault detection system based on the SNMP protocol.** PhD Thesis University of Oklahoma Graduate College.

ภาคผนวก ก

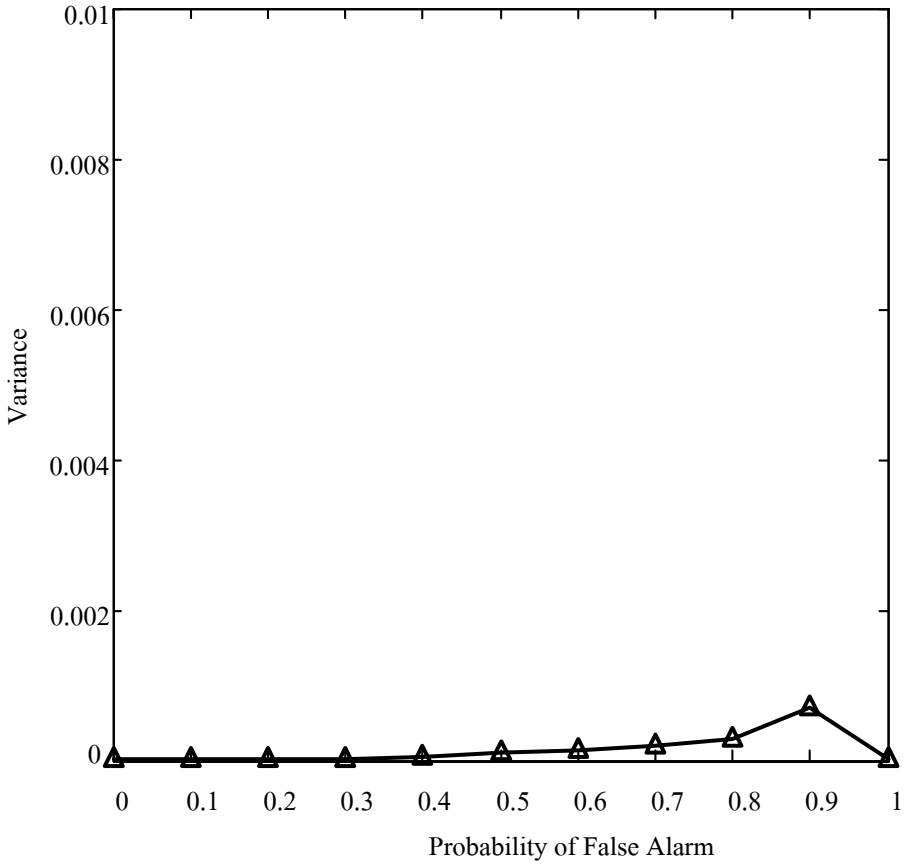
ค่าความแปรปรวนในการจำลองแบบ

ค่าความแปรปรวนในการจำลองแบบ

ในการจำลองแบบเพื่อให้ผลที่ได้มีความถูกต้อง ซึ่งต้องมีจำนวนครั้งในการทดลองที่มากเพียงพอ โดยเมื่อพิจารณาค่าที่ได้จากผลการทดลองในแต่ละครั้งผลที่ได้ไม่ควรที่จะห่างจากค่าเฉลี่ย (mean) ซึ่งแสดงออกมาในรูปของค่าความแปรปรวน (variance) จากการจำลองแบบในโครงข่ายขนาดเล็กซึ่งทำการทดลอง จำนวน 50,000 เอพพิโซด ทำการทดลองทั้งหมด 20 ครั้ง โดยมีค่าความแปรปรวนดังรูปที่ ก-1 และในการจำลองแบบในโครงข่ายขนาดใหญ่ซึ่งทำการทดลอง จำนวน 200,000 เอพพิโซดทำการทดลองทั้งหมด 20 ครั้ง โดยมีค่าความแปรปรวนดังรูปที่ ก-2



รูปที่ ก-1 ค่าความแปรปรวนของการจำลองแบบในโครงข่ายขนาดเล็ก



รูปที่ ก-2 ค่าความแปรปรวนของการจำลองแบบในโครงข่ายขนาดใหญ่

ภาคผนวก ข

บทความทางวิชาการที่ได้รับการตีพิมพ์เผยแพร่ในระหว่างศึกษา

รายชื่อบทความที่ได้รับการตีพิมพ์เผยแพร่ในขณะศึกษา

1. “การลดปริมาณของการโพลล์ในการเฝ้าตรวจสถานะโครงข่าย ด้วยการเรียนรู้แบบรีอินฟอร์สเมนท์” การประชุมวิชาการทางวิศวกรรมไฟฟ้า ครั้งที่ 28 (EECON-28) ตุลาคม 2548 มหาวิทยาลัยธรรมศาสตร์ หน้า 71-74

ประวัติผู้เขียน

นายอรุณรัตน์ นูพลกรัง เกิดเมื่อวันที่ 17 กรกฎาคม พ.ศ. 2513 เกิดที่อำเภอเมือง จังหวัดนครราชสีมา สำเร็จการศึกษาระดับมัธยมศึกษาตอนปลาย จากโรงเรียนโคราชพิทยาคม จังหวัดนครราชสีมา และสำเร็จการศึกษาระดับปริญญาตรี จากสถาบันเทคโนโลพีพระจอมเกล้า พระนครเหนือ จังหวัดกรุงเทพมหานคร เมื่อ พ.ศ. 2541 เคยรับราชการที่กรมอิเล็กทรอนิกส์ทหารเรือ และกรมสื่อสารทหารเรือ ปัจจุบันปฏิบัติงานในตำแหน่ง พนักงานระบบเครือข่ายสื่อสารที่ธนาคารเพื่อการเกษตรและสหกรณ์การเกษตร และได้รับทุนศึกษาต่อในระดับปริญญาโท ขณะกำลังศึกษาระดับปริญญาโทได้รับเงินอุดหนุนจากกองทุนวิจัยและพัฒนาเพื่อทำวิทยานิพนธ์ ระดับบัณฑิตศึกษา ประจำปีงบประมาณ พ.ศ. 2549 จากสถาบันวิจัยและพัฒนา มหาวิทยาลัยเทคโนโลยีสุรนารี