

博士學位論文

MANET에서의
능동형 PDP 탐색 기법 기반의
정책 기반 네트워크 관리 시스템

濟州大學校 大學院

컴퓨터工學科

許智阮

2009年 12月

MANET에서의 능동형 PDP 탐색 기법 기반의 정책 기반 네트워크 관리 시스템

指導教授 宋 旺 晳

許 智 阮

이 論文을 工學 博士學位 論文으로 提出함

2009年 12月

許智阮의 工學 博士學位 論文을 認准함

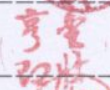
審査委員長

한 기 증



委員

김 광 령



委員

변 홍 용



委員

이 상 조

이 상 조

委員

송 왕 철



濟州大學校 大學院

2009年 12月

감사의 글

무언가를 이루려고 하는 마음이 살아있는 한 언젠가는 뜻을 이룰 수 있으리라 생각했지만, 어느덧 시간은 많이 흘러버린 듯 합니다. 오랜 세월을 고향을 떠나 살았고 이제 고향에서 어찌면 마지막일지 모르는 졸업을 하게 되었습니다.

이 논문이 나오기까지 많은 밤을 지새우고, 고민하고, 여러 가지 도움을 받았습니다. 아직도 무엇인가를 더 써야만 했던 것을 놓친 것이 없는지 많이 망설여지기는 하지만 그럼에도 저를 학자의 길로 이끌어 주신 송왕철 지도교수님께 제일 먼저 감사를 드립니다.

그리고 논문 심사에 시간을 할애하여 생각한 것을 구체화하고 그것을 표현하게 해 주신 안기중 교수님, 김장형 교수님, 변상용 교수님, 이상준 교수님께 감사드립니다. 박사과정을 보내면서 많은 깨달음을 주신 곽호영 교수님, 김도현 교수님과 변영철 교수님께도 감사의 마음과 함께 건강을 기원합니다.

별써 여러 해를 연구실에서 지내면서 여러 선·후배들께 도움을 받았습니다. 많은 시간을 같이하며 여러 가지 고민을 묵묵히 들어주셨던 김강석 학형에게 감사의 말을 전합니다. 학부과정에 있으면서도 스스로 학업에 열심이며 나이 많은 선배의 말에 잘 따라주는 은성, 승찬, 유석, 민협에게도 앞날에 뜻하는 바를 반드시 이루기를 빌겠습니다. 석사 후배이면서 이제는 사회에서 훌륭하게 생활하고 있는 정윤, 경진과 성수에게도 이 작은 결실에 그대들의 도움이 있었다는 것을 알고 있다고 말하고 싶습니다.

그리고 이 학교가 낫설지 않게 받아 준 후배 권훈과 노영민, 여러 가지 서류 처리를 해 주신 정은경 선생과 이정하 선생께도 감사드립니다.

여러모로 응원해주며 논문을 쓸 수 있도록 배려를 해 주신 제주한라대학의 정보통님과 교수님들에게도 감사의 말씀을 전합니다.

제 두 딸에게 이 논문으로 못난 아빠를 조금이나마 용서해 줄 수 있었으면 합니다. 이제 조금은 더 같이 시간을 보낼 수 있을거라는 약속도 합니다. 그리고 지치고 미련조차 남지 않은 제게 이 논문을 쓰게끔 해 준 호경에게 사랑을 전합니다. 두 분 누님께도 조금의 위안이 되기를 바랍니다.

미련한 아들을 믿어 주시고 말없이 지켜 주신 아버님과 이 세상에서 가장 현명한 제 어머님께 이 논문을 바칩니다. 항상 건강하세요.

마지막으로, 이 세상의 모든 오픈소스 개발자들에게 감사드립니다.

2009년 겨울, 연구실에서
허지완



목 차

그림목차	V
표 목차	VIII
국문초록	IX
영문초록	XI
약어표	XIII
I. 서 론	1
II. 관련 연구	5
1. 무선 애드 혹 네트워크	5
1) MANET	5
2) VANET	8
2. 정책 기반 네트워크 관리	10
1) 개요	10
2) 정책 기반 네트워크 관리의 구조	11
3) COPS와 COPS-PR 프로토콜	12
4) MANET에서의 정책 기반 네트워크 관리	14
3. QoS	15
1) 개요	15
2) TCP/IP 프로토콜에서의 QoS에 관한 문제점	16
3) QoS가 필요한 응용들	18
4) QoS의 구조	19
5) QoS 모델	21
4. 요약	22
III. MANET을 고려한 정책 기반 네트워크 관리 프레임워크의 설계	23
1. MANET을 위한 정책 기반 네트워크 관리의 전체 구조	23
2. 프로토타입의 분석	24
3. 정책 전송 영역 관리 구조	25
1) k-hop Cluster	25
2) 능동형 PDP 탐색	27

4. 시스템 접근	30
1) MANET의 정책 기반 네트워크 관리 프레임워크를 위한 구성 요소	31
2) MANET의 정책 기반 네트워크 관리 프레임워크를 위한 구현 요소	34
5. 요약	40
IV. 구현	41
1. 하드웨어 환경 구축	41
1) 하드웨어	41
2) 하드웨어 파라미터	43
2. 소프트웨어 환경 구축	45
1) 운영체제와 QoS 트래픽 제어	45
2) MANET 라우팅 프로토콜	46
3) COPS 프로토콜의 구현	48
4) 측정 및 분석 도구들	49
3. COPS-PR의 구현	49
1) COPS 프로토콜	49
4. 능동형 PDP 탐색 구현	52
1) 능동형 PDP 탐색 기법에 대한 고찰	52
2) KA(keep-Alive) 메시지의 확장	56
3) PREQ, PREP 에이전트	60
4) Management Node List의 작성	64
5. QoS 모델의 적용	66
1) 개요	66
2) DSMARK의 적용	66
3) 트래픽 처리 인터페이스의 개발	68
6. 정책 관리 도구	69
1) PIB의 확장	69
2) 정책 관리 도구의 구현	50
7. 요약	72
V. 구현 결과와 분석	74
1. 능동형 PDP 탐색 기법의 개선	74
1) PEP에 의한 광고메시지의 해결	74

2) Find Other PDP 메시지의 개선 가능성	74
3) PEP의 Connection Close 메시지	75
4) PREQ 메시지 헤더의 TTL 필드	76
2. 측정 시나리오	76
3. 결과의 분석	78
1) MANET 라우팅에서 광고메시지	78
2) 노드의 이동과 능동형 PDP 탐색의 동작	81
3) 비디오 스트리밍 실험	82
4. 요약	85
VI. 결론	87

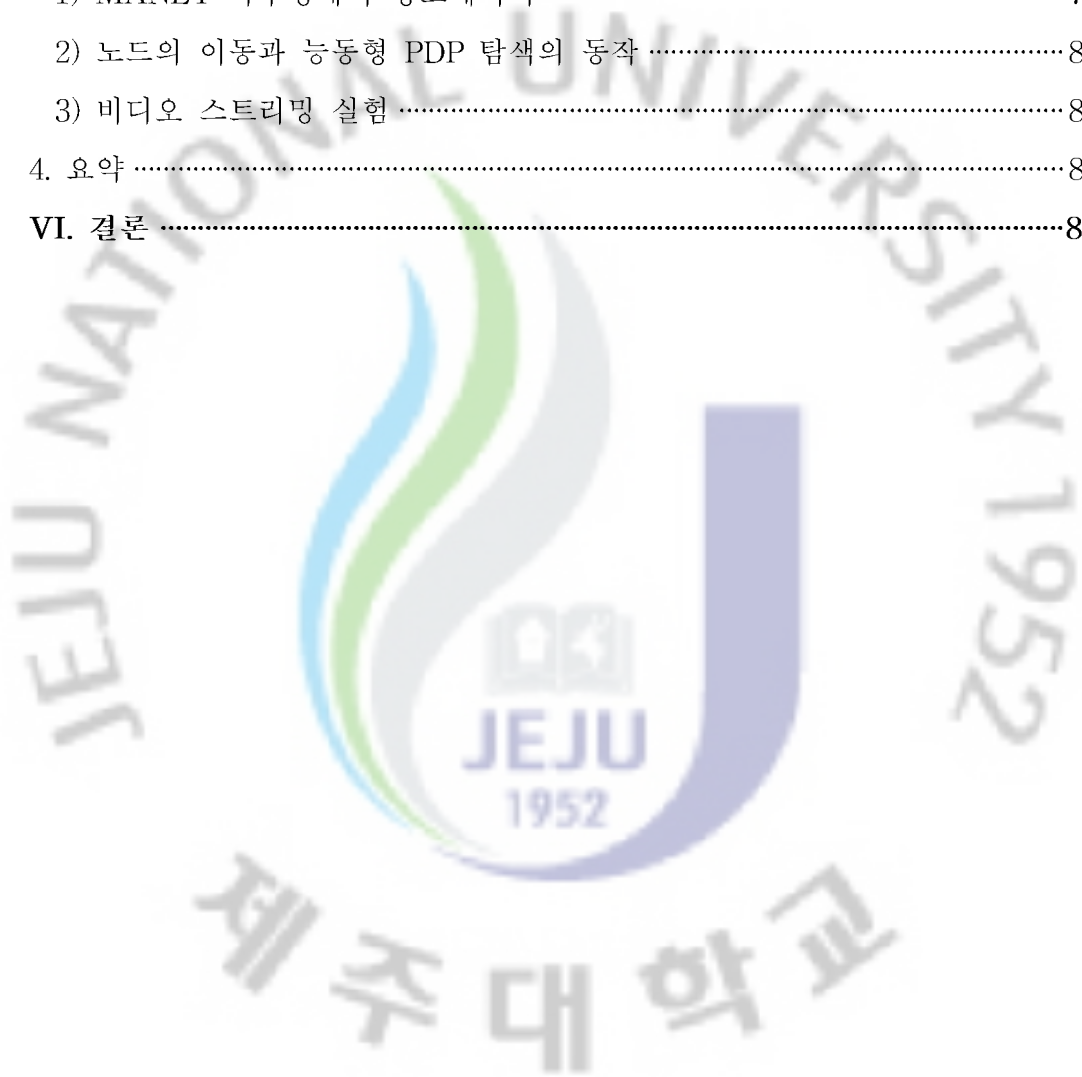


그림 목차

그림 1. 전체 연구 구성도	2
그림 2. MANET의 개념	6
그림 3. VANET의 개념	8
그림 4. 차량 안전을 위한 기술들	9
그림 5. 정책 기반 네트워크 관리의 구조	11
그림 6. COPS-RSVP 구조	13
그림 7. COPS-PR 구조	14
그림 8. IPv4의 헤더	16
그림 9. ToS 필드의 DSCP 필드 사용	19
그림 10. 네트워크 데이터 처리 과정	20
그림 11. VANET에서의 정책 기반 네트워크 관리 구현을 위한 개념도	23
그림 12. k=1 hop Cluster	26
그림 13. 파급 효과	27
그림 14. 능동형 PDP 탐색과 PDP 선택 과정	28
그림 15. 능동형 PDP 탐색의 PDP 동작	29
그림 16. 능동형 PDP 탐색에서 PEP 동작	30
그림 17. 단계별 구현 개념도	34
그림 18. COPS-PR과 능동형 PDP 탐색 구성도	35
그림 19. Class Based Queuing	38
그림 20. HTB에 의한 대역폭 분배	38
그림 21. DSMARK 동작 개념	39
그림 22. 알루미늄 포일로 전송거리를 줄인 노드들	42
그림 23. 랩톱PC의 NIC 모듈에서 안테나 제거 후 포일로 감싼 부분	43
그림 24. 드라이버의 설정 가능한 값	44
그림 25. 무선 NIC의 드라이버 기본 세팅	44
그림 26. Tx-Power 값을 조정 후 드라이버 설정 값	45
그림 27. OLSRD httpinfo plugin	46
그림 28. topology_view 스크립트에 의한 출력물	47

그림 29. 네트워크 토폴로지 변화의 측정	47
그림 30. COPS프로토콜에 의한 연결 과정	50
그림 31. COPS의 구조	52
그림 32. 능동형 PDP 탐색의 PDP 선택과정	53
그림 33. OLSR 라우팅에서 추출한 라우팅 정보	54
그림 34. COPS 프로토콜의 표준 KA(Keep-Alive) 메시지	57
그림 35. KA 메시지 확장을 위한 CALLBACK 함수 구현	58
그림 36. PEP에서 확장한 KA 메시지의 처리	59
그림 37. 확장한 KA(Keep-Alive) 메시지의 화면 출력	60
그림 38. PREQ의 헤더 구조	61
그림 39. PREP의 헤더 구조	61
그림 40. PREQ_agent	63
그림 41. PREP_agent	63
그림 42. PREP의 화면 출력	64
그림 43. PREP의 화면 출력 #2	64
그림 44. 라우팅에 의한 라우팅 정보	65
그림 45. 변환된 노드 리스트	65
그림 46. 리눅스 커널에서 QoS를 설정	67
그림 47. 리눅스 커널의 DSMARK 모듈 설정	67
그림 48. traffic control DSCP	68
그림 49. 외부 입력 파일에서 파라미터를 적용하게 한 확장된 PIB	70
그림 50. 정책 관리 도구의 웹 인터페이스	70
그림 51. PDP노드의 확장된 PIB와 PMT에 의해 정책 전송 화면	71
그림 52. PEP에서 정책 수신 화면	72
그림 53. 새로운 PDP로 연결하는 셸 스크립트	75
그림 54. PDP의 PEP 접속 종료 화면	76
그림 55. 측정 환경	77
그림 56. 2 홉 토폴로지에서 라우팅 테이블	78
그림 57. 전송 패킷의 분석	79
그림 58. MANET 라우팅에 의한 광고메시지와 ARP	80
그림 59. PEP#2 노드 이동 중에 PDP#2에서의 COPS, KA, Total Packets	81

그림 60. TCP 전송과 COPS의 KA메시지	82
그림 61. QoS Policy 적용에 따른 전송 패킷량 변화	83
그림 62. 정책 적용시의 트래픽 변화	84
그림 63. PDP#1로부터 정책 수신 된 PEP#2의 비디오 스트리밍	84
그림 64. PDP#2로부터 정책 수신 된 PEP#2의 비디오 스트리밍	85



표 목차

표 1. QoS 우선순위 계층	21
표 2. DSMARK의 클래스 분류	39
표 3. 표준 COPS 프로토콜의 PIB의 구조	52
표 4. PIB 테이블의 구조	69
표 5. 전송 패킷의 분석	80
표 6. QoS Policy 적용에 따른 측정	83



MANET에서의 능동형 PDP 탐색 기법 기반의 정책 기반 네트워크 관리 시스템

컴퓨터공학과 허지완
지도교수 송왕철

MANET(Mobile Ad hoc Network)은 그동안 많은 가능성에도 불구하고 군사 작전 네트워크, 모바일 게임 기기, 센서 네트워크와 같은 특정 분야로 연구가 집중되어 왔다. 하지만 최근 모바일 기기의 종류와 수가 많아지면서 새로운 서비스 모델들이 제시되고 있다. 특히 VANET(Vehicular Ad hoc Network)은 실질적인 MANET 기술의 적용 분야로 우리나라를 비롯한 세계 각국의 표준화에 대한 연구가 활발히 진행 중에 있다. 또한 MANET에 정책 기반 네트워크 관리(Policy Based Network Management)를 도입하고자 하는 연구는 유선 네트워크에서의 정책 기반 네트워크 관리와 마찬가지로 효율성과 신뢰성을 확보하기 위해 시작되었다. 하지만 대규모 유선 네트워크를 기반으로 표준이 제정된 정책 기반 네트워크 관리의 구조는 노드가 이동성이 있고 노드 간 자율적인 토폴로지를 구성하는 특성을 가진 MANET에 적용하기 위해서는 추가적인 연구가 필요하다. 특히 VANET은 도로 유실, 차량 사고 등 긴급한 상황에 대하여 즉시 후속 차량에 알려 안전성을 확보하는 것이 목표이므로 정책 기반 네트워크 관리의 도입이 가장 필요한 분야라 할 수 있다.

본 논문은 VANET으로의 MANET기술 적용에 관한 연구와 정책 기반 네트워크 관리 도입에 관한 연구를 위해 MANET의 정책 영역 관리 메커니즘으로 제안된 능동형 PDP 탐색 기법을 표준 COPS-PR 프로토콜을 확장하여 실제 네트워크에서 구현하였으며, 이 과정을 통해 능동형 PDP 탐색 기법의 설계를 검증하고, 실험에 사용된 MANET 라우팅 프로토콜을 이용하여 정책 전송 영역 관리에 있어서 네트워크에 대한 부하가 보다 적은 메커니즘으로 개선하였다.

주요어 : 정책 기반 망 관리, 이동 애드 혹 네트워크, 차량 간 애드 혹 네트워

크, 전송 품질 보장



ABSTRACT

A PBNM System with Active PDP Discovery Mechanism in MANET

HUH, JEE-WAN

Department of Computer Engineering

Graduate School

Jeju National University

MANET(Mobile Ad hoc Network), despite its many potentialities, has been focused only on specific areas such as military operation networks, mobile game instruments, and sensor networks. However, new service models are being presented with increasing kinds and number of mobile instruments. Especially, active studies are now under way among a variety of countries including Korea to standardize VANET(Vehicular Ad hoc Network) as a real appliance of MANET technology. Also, the study to introduce PBNM(Policy Based Network Management) to MANET, just like PBNM in wired networks, has been set off to secure efficiency and reliability. Nevertheless, there is a need for additional research to apply PBNM to MANET whose nodes have mobility and which has a characteristic of composing network topology autonomously among nodes, since PBNM is standardized based on large-scale wired networks. VANET is a particular area that needs introduction of PBNM the most since it aims at securing safety by informing following cars of urgent situations like road sweep or car accidents.

To study how to apply MANET technology to VANET and how to introduce PBNM, this paper implements Active PDP Discovery mechanism which is suggested as policy transfer cluster management mechanism of MANET in real networks by expanding standard COPS-PR protocol, verifies

the design of Active PDP Discovery mechanism through this process, and improves it to a mechanism with less network load by using MANET routing protocol which is employed in the test.

Keywords : Policy Based Network management, Mobile Ad hoc network, Vehicular Ad hoc Network, Quality of Service



약어표

ACC	Adaptive Cruise Control
AF PHB	Assured Forward Per Hop Behaviour
bps	bit per seconds
C2C_CC	Car to Car Communication Consortium
CBQ	Class Based Queuing
CC	Connection Close
CDMA	Code Division Multiple Access
COPS	Common Open Policy Service
COPS-PR	Common Open Policy Service-Provisioning
COPS-RSVP	Common Open Policy Service-Resource Reservation Protocol
CPU	Central Processing Unit
DiffServ	Differentiated Services
DMB	Digital Multimedia Broadcasting
DSCP	Differentiated Service Code Point
DSMARK	Differentiated Service Mark
EF PHB	Expedited Per Hop behaviour
FIFO	First IN First Out
FOP	Find Other PDP
FTP	File Transfer Protocol
GPS	Global positioning System
GREED	Generalized Random Early Detection
HTB	Hierarchical Token Bucket
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
IntServ	Integrated Services
IPTV	Internet Protocol Television
IPv4	Internet Protocol version 4
IPv6	Internet Protocol version 6

ISP	Internet Service Provider
IT	Information Technology
KA	Keep-Alive
LDAP	Lightweight Directory Access Protocol
LKS	Lane Keeping Support
MAC	Medium Access Control
MANET	Mobile Ad hoc Network
PCI	Peripheral Component Interconnect Bus
MNL	Management Node List
NIC	Network Interface Card
NoW	Network on Wheels
OLSR	Optimized Link State Routing
PBNM	Policy Based network Management
PCS	Pre-Crash Safety
PDP	Policy Decision Point
PEP	Policy Enforcement Point
PIB	Policy Information Base
PMT	Policy Management Tool
PREP	PDP Response
PREQ	PDP Request
RTP	Realtime Protocol
QoS	Quality of Service
RED	Random Early Detection
RF	Radio Frequency
RSVP	Resource Reservation Protocol
SDK	Software Development Kit
SFQ	Stochastic Fair Queuing
SMI	Structure of Management Protocol
TBF	Token Bucket Flow
ToS	Type of Service
V2I	Vehicle to Infra-structure

V2V	Vehicle to Vehicle
VANET	Vehicular Ad hoc Network
VLAN	Virtual Local Area Network
VMC	Vehicle Multi-hop Communication
VoIP	Voice over Internet Protocol
WAVE	Wireless Access Vehicular Environment
WFQ	Weighted Fair Queuing
WRR	Weighted Round Robin
XML	eXtended Markup Language



I. 서론

MANET(Mobile Ad hoc Network)[1]에 대한 연구는 최근 VANET(Vehicular Ad hoc Network)[2]이라는 실질적인 MANET기술의 적용 분야가 대두됨으로서 우리나라를 비롯한 세계 각국에서 표준화를 위한 연구[3]를 진행하는 등 관심이 집중되고 있다. 이러한 MANET에 정책 기반 네트워크 관리(Policy Based Network Management)[4]를 도입하고자 하는 연구[5]는 기존 유선 네트워크에서 표준으로 제정된 정책 기반 네트워크 관리 시스템을 수정하고 확장하여 무선 네트워크 환경에 효율성과 신뢰성을 확보하기 위해 연구되고 있다.

특히 정책 기반 네트워크 관리의 표준 메커니즘 중 정책 교환 프로토콜인 COPS(Common Open Policy Service)[6]는 노드가 이동성을 가지는 MANET에서 어떤 노드가 PDP(Policy Decision Point)가 될 것이며, 어떠한 PEP(Policy Enforcement Point)노드에게 정책을 배포할 것인지에 대한 연구가 필요하다. 정책 교환과 정책 전송 영역 관리에 관한 연구는 정책 전송 영역 관리 메커니즘 자체의 네트워크 부하를 줄이면서 정책을 수신하지 못하는 노드의 수를 줄이는 것을 목표로 하고 있다. 이에 대한 새로운 메커니즘으로 능동형 PDP 탐색(Active PDP Discovery)[7] 기법이 제안되었다.

본 논문은 능동형 PDP 탐색 기법을 실제 네트워크에서 구현하여 설계를 검증하고 정책 전송 영역 결정 과정에서 광고메시지가 발생하지 않는 구조로 개선하였다. 아울러, 표준 정책 기반 네트워크 관리의 모든 요소를 포함하고, VANET을 고려한 QoS모델에 대한 연구와 실제 트래픽에 적용하는 전체 프레임워크를 설계하고 구현하여 결과를 분석한다.

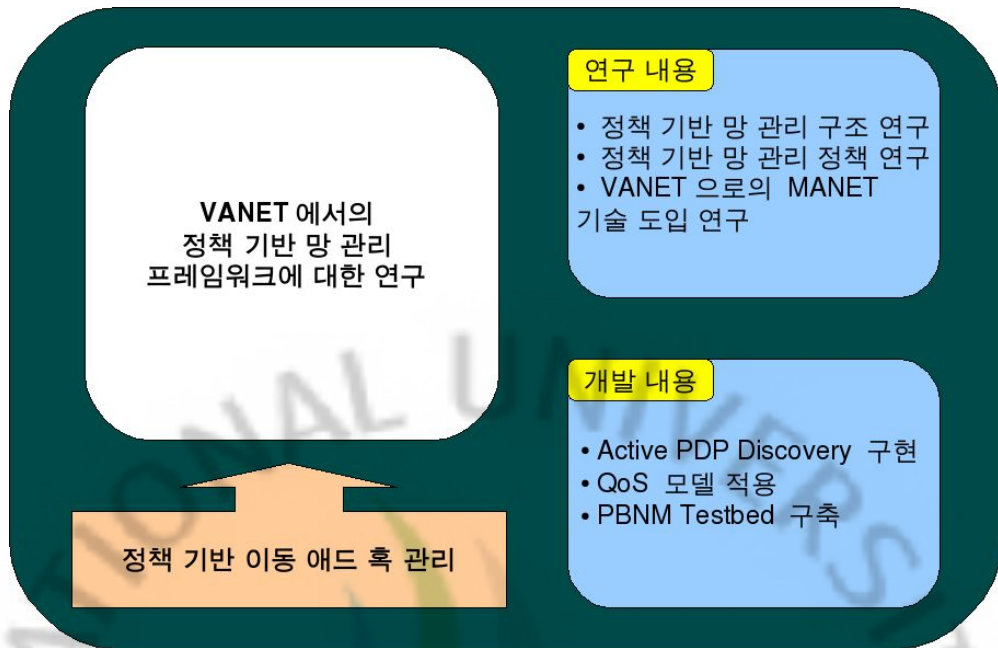


그림 2. 전체 연구 구성도

그림 2.은 본 논문의 전체적인 연구 영역을 도식하고 있다. 그동안 이루어진 정책 기반 MANET 관리에 관한 연구를 바탕으로 표준 정책 기반 네트워크 관리의 기본 구조를 확장하여 연구를 진행하였다. 표준 정책 기반 네트워크 관리의 구조는 유선 네트워크를 기반으로 각 노드의 풍부한 자원과 대역폭을 가지고 있다고 가정하며, 노드의 이동성은 고려하고 있지 않다. 이런 특성으로 인해 유선 네트워크에서는 크게 고려되고 있지 않은 정책 전송 영역 관리에 관한 연구는 MANET에 정책 기반 네트워크 관리를 적용하는데 필수적이라 할 수 있다.

그동안 인터넷 상의 QoS(Quality of Service)[8]에 관한 연구는 정책 기반 네트워크 관리 방식[9]이 관심을 끌어 왔다. 각 네트워크 엔티티를 개별적으로 구성하고 관리하는 전통적인 네트워크 관리 시스템과는 다르게, 정책 기반 네트워크 관리는 네트워크 보안, QoS와 같은 다양한 운용 특성을 전체적으로 구성, 제어 하면서 동시에 네트워크 운용자에게는 단순하고 논리적으로 집중화되어 있으면서 전체 네트워크에 대한 자동화된 제어방식을 제공하게 된다. 지금까지는 정책 기반 네트워크 관리는 주로 대규모의 유선 네트워크에 초점이 맞추어져 왔고, 엔터프라이즈 네트워크, 콘텐츠 제공 네트워크, ISP(Internet Service Provider) 네트워크 등이 그 대상이었다. 본 연구는 무선 네트워크이며, 노드가 이동하며, 노드 간 토폴로지를 자율적으로 구성하는 MANET 환경에서의 정책 기반 네트워

크 관리에 대한 연구이다.

본 연구에서는 이러한 기존 연구들을 배경으로 하여 MANET의 많은 특성에서 언급되듯이 각 이동 노드들의 자원이 풍부하지 않은 경우에 대한 고려가 충분히 되어야 할 것이다. 기본적으로는 가장 필수적인 기능인 자율-복구(self-healing) 및 자율-적응(self-adaptive) 기능을 제공할 수 있어야 하겠고, 제시하는 관리 기능들이 역동적인 네트워크 변화에 자치적으로 적응할 수 있어야 할 것이다. 본 연구에서 제시하는 네트워크 관리 시스템은 다음과 같은 특성을 가지도록 한다.

- 일반화와 변형가능성 : 자원이 풍부하지 않은 특성에 대해 충분히 고려해서 설계되어야 하겠으나, 기본적으로 모델은 일반화 되어야 하며, 이를 다시 특정 상황에 적용 가능할 수 있도록 변형 가능한 모델이 제시되도록 하겠다.
- 효율적인 대역폭 사용 : MANET에서 가장 고려되어야 할 부분이라고 생각한다. 접속이 단절되는 상황 역시 충분히 고려해야 하며, 대역폭을 효율적으로 관리하려면, 응용의 특성에 따라 효율적으로 관리할 수 있는 새로운 관리 방식이 모색되어야 하겠다.
- 완전한 분산 모델 : 분산기법은 기존의 정책 기반 네트워크 관리에서도 고려되고 있으나, MANET으로의 적용에 있어서는 더욱 중점을 두고 개발해야 할 것이다. 따라서 정책 서버들 간의 부하 분산(load balancing) 등을 고려함은 물론이고, 상황에 따른 역할의 전이 등이 고려된 분산모델을 제시해야 할 것이다.
- 자율성 : 이는 자율-복구 및 자율-구성(self-organizing)에 중점을 두고 있으나, 고립되어 있는 상황이 충분히 고려된 상황에서의 자치기능을 더욱 강화시킨 모델의 개발이 필요하리라 본다.
- 적용성 : 이는 MANET 관리 시스템에서의 필수적인 고려점이나, 본 연구에서는 위의 자치성과 연관된 모델의 개발에 좀 더 중점을 둔다.
- 견고성 : MANET 관리 시스템은 가장 절박한 상황을 고려해야 할 것이다.

본 논문은 1장에서 서론으로서 연구의 목적에 대하여 기술하고, 2장에서는 관련 연구를 기술하며, 3장에서는 능동형 PDP 탐색 기법을 도입한 정책 기반 네트워크 관리 프레임워크에 대하여 설명하고, 4장에서는 능동형 PDP 탐색 기법의 설계를 수정하고 개선한 내용에 대한 설명과 구현의 절차와 방법을 설명하였으

며, 5장에서는 능동형 PDP 탐색 기법에서 도출된 데이터를 분석하여 제안하는 프레임워크를 증명하고, 6장에서 결론으로 끝을 맺는다.



II. 관련 연구

이 장에서는 연구의 배경이 되는 관련 연구에 대해서 기술한다. 1절에서는 광의의 무선 애드 hoc 네트워크(Wireless Ad hoc Network)에서 MANET과 그 예로 VANET에 대하여 기술하고, 2절에서는 정책 기반 네트워크에 대하여 논하며, 3절에서는 QoS 모델에 대하여 기술한다.

1. 무선 애드 hoc 네트워크

무선 애드 hoc 네트워크는 무선 네트워크에 애드 hoc 토폴로지를 이루는 네트워크를 뜻한다. 그러므로 라우터나 액세스 포인트와 같은 인프라스트럭처가 없이 노드 간 토폴로지를 구성하는 네트워크를 의미한다. 이러한 네트워크에서 노드가 이동성을 가지는 네트워크를 MANET으로 분류한다.

1) MANET

MANET은 무선 애드 hoc 네트워크[10]에서 이동성이 있는 네트워크로 세분되고, 메시 네트워크와는 유·무선의 구분으로 분류된다. 랩톱PC의 일반화와 IEEE 802.11 기술의 발전으로 1990년대부터 관심을 끌기 시작한 MANET은 2007년 9월에는 스웨덴의 TerraNET AB에서 가입자 단말기 간 통화가 가능한 서비스를 제공[11]하기에 이르렀다.

애드 hoc 네트워크에서의 QoS에 대해 기존의 문헌들은 크게 두 그룹으로 나뉜다. 하나는 QoS 구조와 시그널링[11][12][13]을 다루고 있고, 다른 하나는 QoS-인지 라우팅[14][15][16][17]에 대해 논하고 있다.

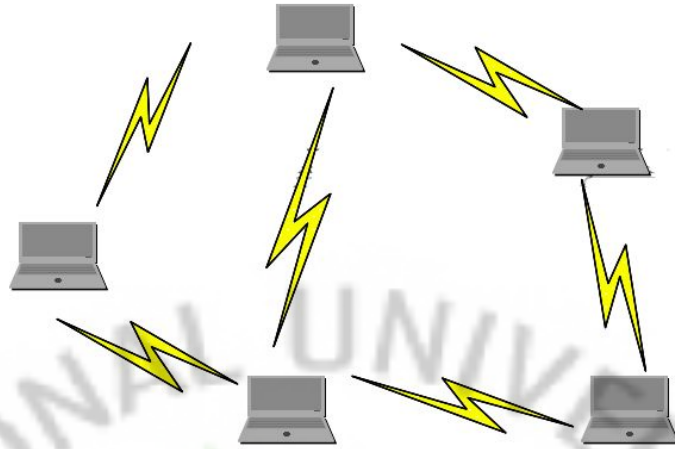


그림 3. MANET의 개념

그림 3.는 MANET의 개념도이다. MANET은 네트워크 구성 노드들이 이동하며, 상호간의 네트워크를 구성하는 특성을 가진 네트워크로써 노드 간의 통신을 무선 인터페이스를 사용하기 때문에 다음과 같은 특성들을 가진다.

낮은 대역폭과 가변 용량의 링크: 무선 링크는 유선보다 전형적으로 대역폭이 제한되어 있으며, 페이딩, 신호간섭, 재밍 등으로 일시적인 링크 고장이나, 채널의 에러율이 급변할 수 있다. 또한, 노드들의 다양한 특성과 사용되는 통신방식의 특이성들 때문에 멀티 홉 무선 네트워크에서 각 링크는 용량이 가변적일 수 밖에 없다.

- 역동적인 토폴로지: 애드 혹 네트워크의 토폴로지는, 구성 호스트들의 이동에 따른 변화는 물론 새로운 노드의 출현, 전파방해 등의 다양한 원인에 따른 네트워크 연결이 변하기 때문에 역동적으로 변하게 된다. 무선 애드 혹 환경에서, 관리 시스템이 빈번한 토폴로지 변화에 적절히 대응하는 것은 중요하다. 하지만, 토폴로지 정보를 자주 교환하게 되면, 시그널링 오버헤드가 발생하고, 낮은 대역폭의 무선 링크에 혼잡이 발생할 수 있으며, 해당 노드의 제한된 배터리 수명을 단축시킬 수 있다.
- 제한된 자원(배터리 수명, 저장용량, 처리용량): 이동 노드의 특성상, 경량화를 위해 자원의 크기나 무게 등에 대한 제한요건이 필수불가결하다.

다중역할: 네트워크 노드는 원천지/목적지 역할은 물론 라우터로서의 여러 역할을 수행한다.

- 이질성: 이는 채용되는 통신 기술의 다양한 특성은 물론, 노드들의 형태가 각각 다르기 때문에 고려되는 요소이다. 각 노드들은 센서 및 팜톱, 랩톱에서부터 배나 탱크, 비행기내에서의 이동 네트워크에 이르기 까지 다양하게 구성될 수 있다. 더욱이, 애드 혹 네트워크는 여러 조직들의 컨소시엄의 형태를 띠 수 있고, 그래서 관리 시스템에 대해서는 부가적인 상호운용성에 대한 문제를 고려해야 할 필요도 있다.
- 제한된 존속성(survivability): 애드 혹 네트워크를 이용함에 있어서 주요 극복과제는 제한된 존속성과 보안 공격에 약하다는 점이다. 군사 교전 지역의 네트워크와 같이 다양하면서도 적대적인 환경에 구축된 무선 애드 혹 네트워크인 경우 네트워크 요소들의 장애를 초래할 다양한 공격들을 고려해야 한다.

일시적이고 특별임무 중심의 구축방식: 애드 혹 네트워크의 구축은 일시적이며 특정 임무를 중심으로 이뤄질 수 있다. 군사용도 이거나 재난관리를 목적으로 구축될 수 있다.

2) VANET

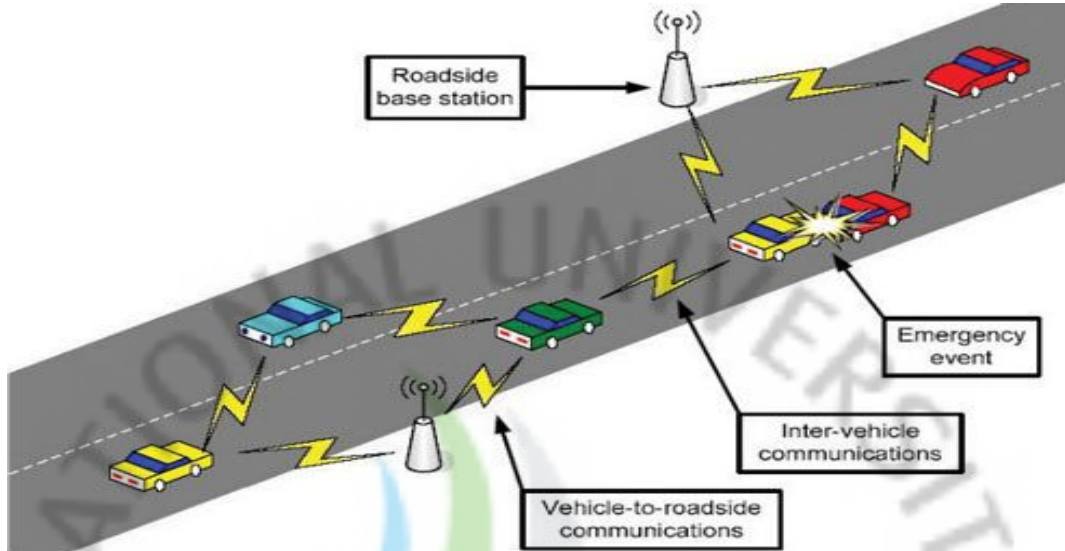


그림 4. VANET의 개념

그림 4.은 VANET의 개념을 도식한 것이다. VANET은 다수의 차량이 무선 환경에서 V2V(Vehicle to Vehicle : 차량과 차량 간) 또는 V2I(Vehicle to Infra-structure : 차량과 노변 장치 간)의 통신을 하는 네트워크이다. 이는 MANET의 한 형태이며 차량은 MANET의 노드의 역할을 하게 된다. 광의의 차량 간 통신은 기지국을 거쳐서 차량 간 통신을 수행하는 방식도 포함하지만 실질적으로는 이동 중이거나 정지 중인 차량들 간의 무선 통신을 말한다. 이러한 VANET은 국내외적으로 텔레매틱스 기술 고도화를 위해 차량에 IT 기술을 융합한 차량 정보 기반 융합 서비스로 그 중요성이 높아지고 있다. 원격차량진단, 긴급구난등과 같은 차량 텔레매틱스 서비스에서 요구되는 차량 정보의 범위도 차량 네트워크에서 추출된 정보를 비롯하여, GPS 기반의 위치정보, 차량 내외부의 부가적인 센서 정보, 멀티미디어 기기 정보, 차량 간 통신 정보까지 수용하여 점차 확장되고 있다.

VANET은 MANET과는 다른 특징을 가지고 있다. VANET은 일반적으로 각 노드의 이동성이 강하며, 이에 따른 네트워크 토폴로지의 변화와 구성된 토폴로지의 노드 밀집도의 변화율 또한 높다. 이로 인하여 네트워크의 단절과 짧은 링크 연결 시간, 높은 패킷 손실률, 무선 채널의 불안정성을 MANET과의 다른 특징으로 들 수 있다.

VANET의 목적은 운전자와 차량의 안전성의 확보에 있다. 도로의 유실, 차량 사고 등의 상황이 발생하였을 때, 이를 후방 차량에 신속히 알려 사고를 미연에 방지하는 데 있다. 이외의 다른 VANET의 부가적인 목적은 차량의 상태에 대한 원격 진단이나 일반적인 도로 상황, 교통 정보를 제공하는 것이다. 이러한 VANET의 목적은 1:1방식의 통신을 위주로 하기는 하나 멀티 홉을 지원해야 한다. 차량의 무선 통신 범위가 제한적일 수밖에 없으므로 후속 차량에 긴급한 상황에 대하여 신속히 알려 위험을 미연에 방지하기 위해서는 차량 간 멀티 홉 통신이 필수적이다. VANET에서는 반대편 차선에서 진행해 오는 차량이 긴급한 상황에 대한 신호를 전파하기 위해서도 멀티 홉 네트워킹이 필요하다.



그림 5. 차량 안전을 위한 기술들

그림 4.에서 현재 개발되어 있거나 개발 중인 차량 안전 기술들을 보인다.

VANET의 적용 기술은 IEEE 802.11기반의 WiFi뿐만 아니라 CDMA등 다양한 무선 기술이 활용될 수 있을 것으로 예상된다. 우리나라의 정부에 의한 스마트하이웨이 개발 계획[19], 미국의 차량 간 이동 통신 표준인 WAVE(Wireless Access Vehicular Environment)[20], 유럽의 i2020 비전에 따른 독일의 NoW(Network on Wheels) 프로젝트[21], 그리고 민간 컨소시엄인 유럽 자동차 메이커에서 결성한 C2C_CC(Car to Car Communication Consortium)[22]등이 국

내외 주요 VANET에 관한 표준화 동향이며 우리나라의 경우, 이에 관련한 기술 연구는 전자통신연구원에서 주로 진행되고 있다. 2004년부터 2005년까지 2년간 무선통신통합 기술 개발을 수행하여 텔레매틱스 서비스를 저렴하게 제공하기 위해 셀룰러, 무선랜, DMB를 지원하는 단말에서의 무선통합 프로토콜 구조 및 차량 간 통신기술을 연구하였다. 차량 간 통신은 기존의 2.4GHz 무선랜 기술의 RF와 모뎀 기술을 이용하여 차량 간 통신을 위한 MAC기술을 연구하였는데, 2.4GHz 무선 네트워크를 차량 간 통신에 적용하였을 때 이동 시에 2Mbps 정도의 패킷 전송이 가능함을 확인하였다. 또한, 동 연구소에서 2007년부터 4년간 VMC(Vehicle Multi-hop Communication)기술 개발을 현재 진행 중에 있으며, 응급 메시지를 CDMA기술을 통해 송신하는 특징을 가지고 있다. 미국에서 제정된 WAVE(Wireless Access in Vehicular Environment)는 차량 환경에서 근거리 무선 통신 규약의 표준이다. 이 표준에 의하면 5.9GHz band와 75MHz의 대역폭을 가지며, 1Km 전송 범위와 140Km/h 속도에서 차량 간 통신이 가능하다.

2. 정책 기반 네트워크 관리

1) 개요

인터넷이 급속도로 성장하면서 정책 기반 네트워킹의 필요성이 대두되었다. 이는 TCP/IP 프로토콜이 경쟁 기반의 트래픽 처리를 하고 있고 다양한 서비스와 상황에 인터넷이 사용되기 시작하면서 그에 따른 트래픽의 특성에 따른 차등 적용이 필요하게 된 것이다. 이에 대한 연구는 학계와 업계에서 공히 연구되기 시작했으며 1970년대 처음 새로운 아이디어[27]로 연구되기 시작했다. 트래픽 차등화는 실제 트래픽을 어떻게 처리하느냐에 따른 TCP 프로토콜의 처리로 이루어지며 이 분야의 연구는 많은 부분이 진척이 되어 있으며, 표준화 되었다. 하지만, 네트워크 관리자나 의사 결정자의 의도를 어떻게 정책으로 만들 것이며, 정의된 정책을 저장하는 것은 어떤 형태이며, 이 정책을 전송하는 방법과 실제로 트래픽을 어떻게 처리할 것인지에 대한 정책의 적용에 관한 프레임워크가 필요하게 되

었다. 그래서 정책 기반 네트워크의 연구는 정책에 대한 연구, 트래픽 차등화에 관한 연구, 네트워크 보안이나 IP 주소 관리에 대한 연구 분야에서 진행되어 왔다.

2) 정책 기반 네트워크 관리의 구조

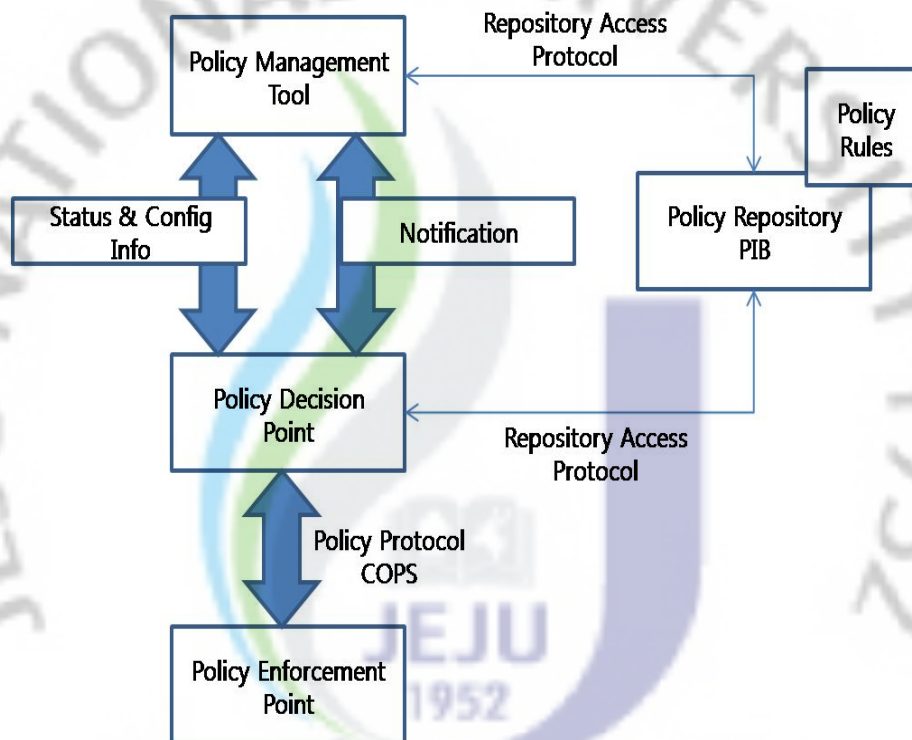


그림 6. 정책 기반 네트워크 관리의 구조

그림 6.는 정책 기반 네트워크의 구조를 보이고 있다. 네트워크 관리자나 조직의 의사 결정자에 의해서 정책이 정의되는 정책 관리 도구(Policy Management Tool)와 정의된 정책을 저장하는 정책 저장소(Policy repository), 정책 저장소로부터 정책을 배포하는 PDP(Policy Decision Point), PDP로 장비의 상태를 송신하고 그에 따른 정책을 수신하는 PEP(Policy Enforcement Point)가 있다.

위와 같은 구성 요소들 중 정책 저장소는 저장 형태와 PDP로의 프로토콜이 필요하다. 정책 저장소의 정책 저장은 PIB(Policy Information Base)[28]의 형태로 저장된다. PIB는 SMI(Structure of Management Information)[29]의 원형을

따르고 이는 널리 쓰이는 SNMP(Simple Network Management Protocol)[30]의 MIB(Management Information Base)[31]와 같다. 또 다른 정책 저장소의 정책 저장과 정책결정자사이의 프로토콜로는 LDAP(Lightweight Directory Access Protocol)[32]이 표준으로 제정되었다. 또한, 저장 구조를 XML형태로 하는 연구 [33]도 표준으로 제안되었다.

PDP는 일종의 정책 서버(Policy server)로 볼 수 있다. 정책 저장소의 정책을 전송받고 PEP로부터 전송받은 장비의 상태에 따라 정책을 전송한다. 그러므로 PEP는 정책 클라이언트(Policy client)로 볼 수 있다. 이 정책결정자와 정책수행자 사이의 프로토콜은 COPS(Common Open Policy Protocol)가 제안되어 있다. 이 프로토콜에 대해서 3항에서 별도로 다룬다.

3) COPS와 COPS-PR 프로토콜

PDP와 PEP간의 프로토콜로 COPS를 정의하고 있다. 본 항에서는 COPS프로토콜과 COPS-RSVP(Resource Reservation Protocol)[34], COPS-PR(COPS Usage for Policy Provisioning)[35]에 대하여 설명한다.

(1) RAP 워킹그룹과 COPS-PR

IETF RAP(resource allocation protocol) 워킹그룹은 QoS 정책 분야를 다루고 있다. 이 그룹에서 정책기반 승인 제어 프레임워크와 COPS, COPS-PR 프로토콜을 정의했다. COPS는 단순한 클라이언트-서버 프로토콜로서 정책 클라이언트와 정책 서버간의 통신을 가능하게 한다. 두 가지 정책 제어 모델이 정의되어 있는데, 한 가지는 아웃소싱으로 COPS를 COPS-RSVP로 확장하였고, 다른 하나는 프로비저닝 방식으로 COPS-PR로 확장되었다.

(2) COPS-RSVP 모델

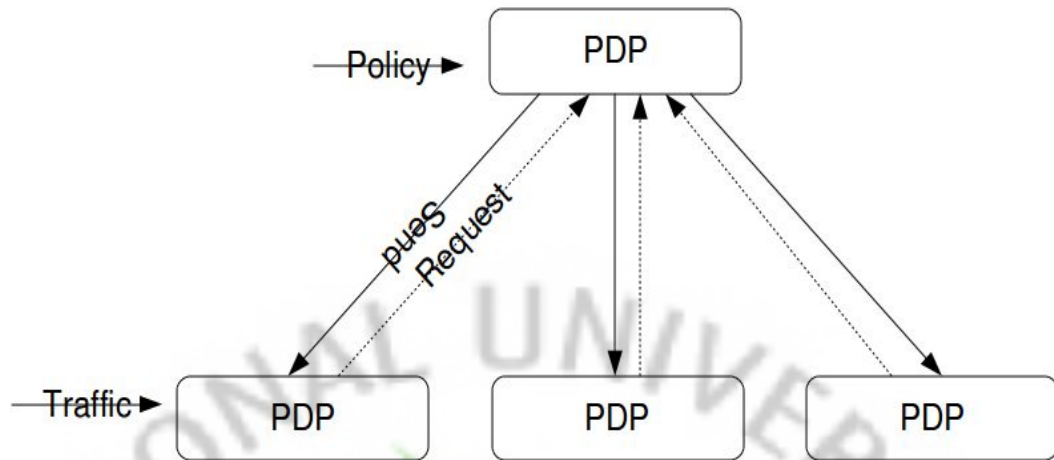


그림 7. COPS-RSVP 구조

COPS-RSVP는 PDP로 하여금 각 PEP에게 내려야 할 즉각적인 정책 결정을 대신 내릴 수 있도록 하는데 초점을 맞추고 있다. 그림 7.에서 각각의 PEP는 새로운 트래픽이 도달할 경우, 해당 트래픽이 장비를 통과하여도 되는지 PDP에게 요청 한다. PDP는 요청에 대한 결정을 내린 후 그 결과를 장비에게 전송한다. 이와 같이 PDP와 PEP간의 통신은 요구에 의한 방법(On-Demand)으로 개시되는 질의-응답(Request-Reply) 모델을 따른다. 이 모델은 COPS 프로토콜에서 간단히 구현이 가능하며, 모든 정책은 PDP에 저장되어야 하고, PEP가 트래픽을 처리할 때마다 PDP로부터 정책을 수신 받아야 한다. PDP 역시 PEP로부터의 트래픽에 대한 모든 관련 정보를 수신하고 이에 대한 정책을 PEP로 전송하여야 한다. 이 구조는 간단하지만 실제 동작에 있어서는 정책 전송 프로토콜이 효율적인 트래픽 처리를 위한 것임을 가만할 때, 정책 전송 프로토콜 자체가 네트워크에 부하를 줄 수 있다.

(3) COPS-PR 모델

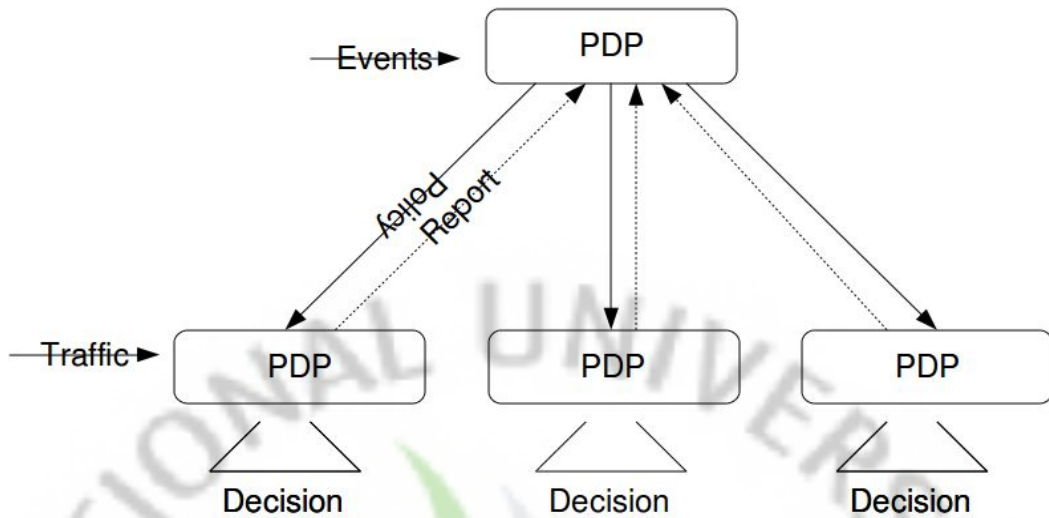


그림 8. COPS-PR 구조

COPS-PR은 PDP가 정책 관련 설정 정보를 비동기적으로 PEP(네트워크 장비)에게 전송하는데 초점을 맞추고 있으며 PDP와 PEP의 1:1 관계를 가정하지 않는다. PDP는 외부 이벤트나 PEP 이벤트 반응으로서 설정 정보를 PEP에 보낸다. 이와 같이 그림 8.은 PDP가 먼저 모든 PEP에게 정책 관련 설정 정보를 보내고 PEP는 이를 실행한 후 보고하는 전송-보고(Send-Report) 모델을 따른다. 그러므로 PEP는 처리할 트래픽에 대해 적용할 정책이 없을 때만 PDP에 정책을 요청한다. 그 외의 상황에서는 PEP는 자체적으로 수신한 정책에 따라 트래픽을 처리한다.

4) MANET에서의 정책 기반 네트워크 관리

MANET에서 정책 기반 네트워크 관리를 적용하기 위한 연구는 미래의 군사 작전 네트워크를 중심으로 시작되었다. 이러한 네트워크는 높은 이동성과 자율적인 네트워크 토폴로지 구성, 자율-복구 등이 필수적이므로 MANET의 특성과 동일하다. 그러므로 정책을 배포하고 배포된 정책은 빠르고 자동으로 네트워크에 적용되는 시스템이 필요하게 된다. 이러한 정책 교환과 수행에 관한 필요성으로

인해 MANET에 정책 기반 네트워크 관리를 도입하는 연구가 진행 중이다.

VANET에서의 정책 기반 네트워크 관리에 대한 연구는 찾을 수 없었으며, VANET으로의 MANET기술 적용에 대한 연구는 주로 전송 기술[4]과 VANET에 적합한 시그널링 기술에 관한 분야에 집중되어 있다. 이 외의 VANET에 대한 연구는 차량 통신 보안 및 프라이버시에 대한 연구[36]로 위협 모델의 정의와 보안 메커니즘에 대한 연구, VANET의 상황을 고려한 암호화 방법을 논하고 있다.

VANET 기술은 현재 초기 상태로 전송과 시그널링 기술에 집중되고 있고 일부 보안에 관련한 이슈에 대한 연구가 이루어지고 있으므로, TCP/IP 프로토콜의 발전과 비슷한 맥락으로 전송과 보안 기술이 정립된 후에 이를 기반으로 한 정책 기반 네트워크의 필요성이 대두될 것으로 예상할 수 있다.

3. QoS

1) 개요

QoS에 대한 연구는 사용자에게 더 나은 전송 품질과 차등화된 서비스를 인터넷에서 제공하기 위하여 연구되기 시작했다. 기초가 되는 아이디어는 사용자에게 따라서 네트워크 전송 품질에 대해 서로 다른 요구와 필요에 의해서 트래픽을 분류하고 처리하는 구조를 개발하는데 있다. 그러므로 QoS는 트래픽 엔지니어링의 한 분야이며 많은 부분 트래픽 엔지니어링의 연구를 참조하고 있다. 서로 다른 응용 프로그램, 사용자, 데이터 플로어에 의해 요구와 필요가 구분되며, 이것은 데이터 플로어의 성능 분류에 차등을 두어 전송 대역의 보장을 이루는 방식이다. VoIP(Voice over IP), 온라인 게임, IPTV와 실시간 스트리밍 멀티미디어 응용들이 주요한 QoS 적용이 필요한 분야[37]들이다.

네트워크나 프로토콜들은 연결과정을 끝나면 세션이 구성된다. 이 세션 연결 과정에서 네트워크 노드들이나 응용 프로그램에서 트래픽 처리에 대한 즉 전송될 데이터 플로어에 대한 대역폭, 딜레이 등 여러 가지 스케줄링에 대한 QoS에

대한 동의 과정을 맺게 된다.

전형적인 최선-노력(best-effort) 방식의 네트워크인 인터넷은 QoS에 대한 기능은 포함하고 있지 않았다. 그러므로 TCP/IP상에서 QoS는 최선-노력 방식의 네트워크 기반에서 프로비저닝 방식으로 특별히 고성능의 트래픽 처리를 요구하는 통신에 복잡한 QoS 메커니즘을 제공하여 처리된다.

패킷 스위칭 네트워크에서 QoS는 두 가지 분류로 나누어 볼 수 있다. '인간'과 '기술'이라는 분류[38]이다. '인간'의 분류에서 QoS는 서비스의 안정성, 서비스 가능성, 지연, 사용자 정보로 볼 수 있으며, '기술'의 분류에서 보면 신뢰성, 유연성, 효율성, 관리가능성, 서비스 계층화로 볼 수 있다.

2) TCP/IP 프로토콜에서의 QoS에 관한 문제점

TCP/IP 프로토콜 기반의 인터넷은 QoS가 라우팅 장치의 성능에 의해서 결정되는 구조이다. 즉, TCP/IP에서 기본적인 QoS 분류는 최선-노력방식이 된다. TCP 헤더에는 ToS(Type of Service)와 Precedence 비트 필드가 준비되어 있지만 이를 적용하여 처리되는 경우는 없으며, 해당 필드 정보는 무시되었다. 그림 9.은 IPv4의 헤더를 도식한 것이다. IPv4에는 ToS 필드가 있고, IPv6에서는 기본적으로 QoS를 지원하기 위하여 Traffic Class 필드가 있다.

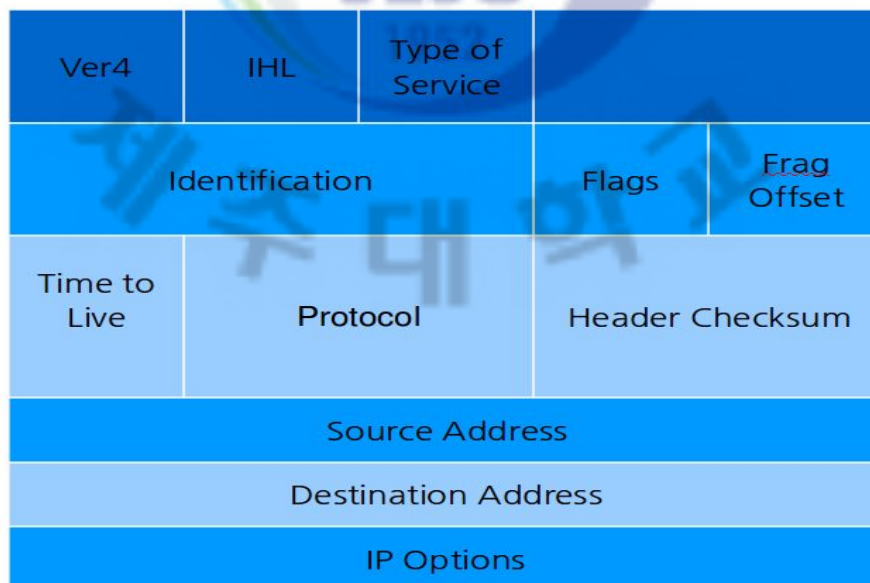


그림 9. IPv4의 헤더

송신지에서 목적지로 패킷이 전달될 때, 다음과 같은 문제점이 발생할 수 있다.

- 패킷 드롭 : 라우터에서 패킷이 드롭 되는 것을 말한다. 라우터의 버퍼가 가득 찼을 때 새롭게 들어오는 패킷은 드롭 될 것이다. 이것은 미리 예측하기 불가능한 면이 있으며 네트워크의 상태에 따라 드롭 되는 패킷은 부분적일 수도 여러 개일 수도 또는 전체일 수도 있다. 전송 메커니즘에 의해서 드롭 된 패킷을 재전송하게 되지만, 드롭 된 패킷이 여러 번에 걸쳐 반복되면 전송을 포기하게 된다.
- 지연 : 라우팅 구간의 길이에 따라, 또는 라우팅 장치의 긴 큐로 인해서 해당 장치가 패킷을 계속해서 잡고 있어 목적지까지 패킷이 전송되는 시간이 길어지는 것을 뜻한다. 이것은 VoIP나 온라인 게임과 같은 지연에 민감한 응용에 치명적이다.
- 지터 : 출발지에서 목적지까지 패킷은 서로 다른 지연값을 가지고 전송된다. 서로 다른 지연값을 가지고 도착한 패킷은 목적지의 큐에서 서로 다르게 위치하게 되고 이는 예측할 수 없다. 이 문제점을 지터라 한다. 지터는 스트리밍 오디오나 비디오와 같은 서비스에 영향이 있다.
- 전송 에러 : 패킷은 인터넷을 통해 서로 다른 패킷이 서로 다른 라우터들을 통과하게 된다. 라우터들은 서로 다른 지연값을 가지고 있으며, 이로 인해 패킷이 목적지에 도착하게 되면 여러 개의 라우터들로부터 도착한 패킷들을 재조립하게 된다. 이 과정에서 잘 못 된 패킷을 수신하거나 도착하지 못한 패킷 때문에 결국은 패킷을 처리할 수 없게 된다.
- 에러 : 라우터에서 패킷이 때때로 잘 못 된 경로로 전송되는 경우나 패킷이 손상되거나 하는 경우를 말한다. 수신측에서는 이런 패킷을 받았을 때, 패킷을 드롭 한다.

3) QoS가 필요한 응용들

QoS적용이 필요한 일례를 들자면 요즘 케이블TV방송국에서는 트리플서비스

라는 하나의 케이블망을 이용하여 전화, 케이블TV, 인터넷을 동시에 제공하고 있다. 이 세 가지 서비스는 각각 서로 다른 트래픽 특성을 가지고 있다. 전화는 필요한 대역폭이 적으나 지터나 딜레이에 민감하고, 케이블TV는 스트리밍 비디오 서비스이므로 역시 지터나 딜레이에 민감하면서 많은 대역폭을 요구한다. 그리고 인터넷은 수시로 대역폭이 변경되는 원래의 특성을 가지게 되므로 세 가지 트래픽 중 가장 지터, 딜레이, 대역폭에 영향을 적게 받는다고 할 수 있다. 이 경우, 각각의 우선순위를 두어 세 가지 서비스를 동시에 사용할 경우 케이블TV 트래픽 처리에 가장 많은 우선권을 주고 그 다음 전화와 인터넷 순으로 대역폭을 분배할 수 있다.

QoS가 필요한 응용들은 다음과 같이 정리할 수 있다.

- 스트리밍 멀티미디어 : 이 종류의 데이터들은 보장된 전송 대역이 필요하다.
- IPTV : IP기반의 TV 신호 전송 방법에서는 ISP에 QoS를 필요로 한다.
- VoIP : IP 기반의 음성 통화 기술에서는 적은 지터와 지연을 요구한다.
- Video Conferencing : 화상 회의용 소프트웨어도 적은 지터와 지연을 요구한다.
- 안전과 위급한 상황에 대처하는 응용 : 원격 진료나 위급한 상황에 발생하는 트래픽은 가장 높은 계층의 보장을 필요로 한다.
- 이외의 실시간을 요구하는 온라인 게임 등

4) QoS의 구조

QoS 제어 메커니즘은 네트워크의 트래픽이 최상위일 때를 가만하여 해당 네트워크상에서 프로비저닝을 제공하여 높은 품질의 통신을 제공하는 것을 목표로 하고 있다. 이러한 시스템 접근은 다양한 응용에 서로 다른 대역폭을 제공하며 때로 낮은 지터나 딜레이를 제공하기도 한다.

초기의 연구로 네트워크 자원을 예약하기 위한 메커니즘으로써 IntServ(Integrated Services)[39]가 있다. 이 메커니즘에서는 RSVP(Resource

Reservation Protocol)을 이용하여 자원을 요청(request)하고 예약(reservation)하는 순서로 되어 있다. IntServ가 동작하면 전송측과 수신측의 경로에 자원을 예약하여 전송을 시작한다. 이 메커니즘은 인터넷의 성장과 함께 거대 네트워크에서는 적합하지 않다는 것이 증명되었다. 왜냐하면 RSVP에 의해서 예약되는 경로의 개수가 너무 많아서 코어 라우터에서 그 많은 경로에 대한 예약을 처리하는 것은 불가능하기 때문이다.

두 번째의 접근은 DiffServ(Differentiated services)[40]이라 불리는 메커니즘이다. 이 모델에서는 패킷의 헤더 중 Type of Service 필드[41]를 마킹하고 이를 지원하는 네트워크 장비를 통해서 트래픽에 맞는 QoS를 제공하는 구조이다. 즉, 라우터나 스위치들은 서로 다른 큐잉 정책을 사용하게 된다. 헤더의 마킹 방법은 IP 계층에서는 DSCP(Differentiated Services Code Point)의 6비트 값을 마킹하여 차등화하며, MAC 계층에서 VLAN의 IEEE 802.1q[42]와 IEEE 802.1D에서 같은 정보를 탑재하여 전송되게 된다. 그림 10.은 IPv4의 ToS 필드를 이용하는 DSCP 마킹의 도식이다.

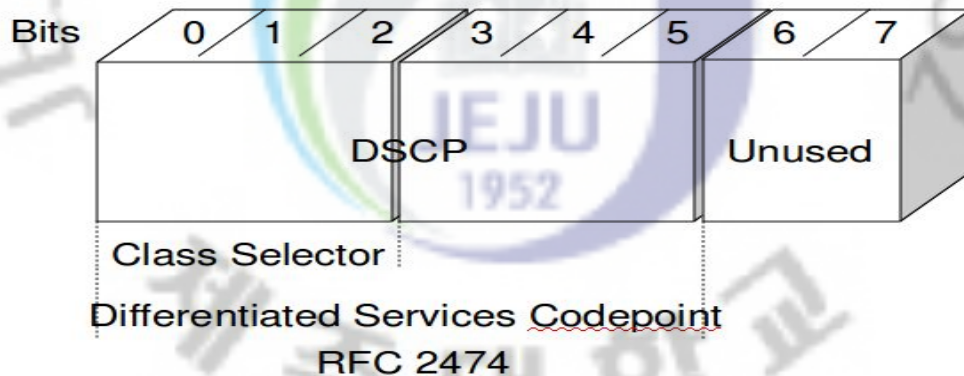


그림 10. ToS 필드의 DSCP 필드 사용

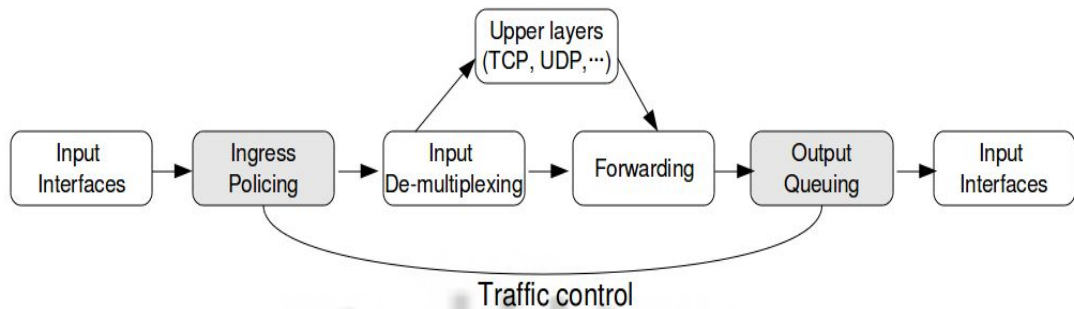


그림 11. 네트워크 데이터 처리 과정

그림 11.은 네트워크에서 데이터가 처리되는 과정을 나타낸 것이다. 그림 7에서 QoS는 들어오는 데이터 패킷에 적용되며, 이는 나가는 패킷 큐잉 정책에서 어떤 트래픽에 우선권을 줄 것인가를 결정되는 과정을 거친다. 그러므로 QoS의 실제 트래픽 처리 과정은 다음과 같은 요소가 필요하다.

큐잉 방법

- 클래스
- 필터링
- 정책

정책을 적용하여 필터링하고 필터링한 헤더에 맞게 클래스를 정의하고 실질적으로 큐잉에 영향을 주어 트래픽을 차등화 하는 여러 가지 모델이 있다. 이러한 모델들을 QoS 모델이라 한다.

5) QoS 모델

지금까지 연구된 주요 QoS 모델들은 트래픽 셰이핑, 스케줄링, 충돌 방지의 분류로 나눌 수 있으며 각 분류의 세부 모델들은 다음과 같다.

트래픽 셰이핑(Traffic shaping)

- 토큰 버킷(Token bucket)

- 리키 버킷(Leaky bucket)

스케줄링 알고리즘(Scheduling algorithm)

- CBQ(Class Based Queuing)
- WFQ(Weighted Fair Queuing)
- WRR(Weighted Round Robin)

충돌 방지(congestion avoidance)

- RED(Random Early Detection) Queuing
- GRED(Generalized RED) Queuing

제안된 QoS 우선순위 계층은 다음 표 1과 같다.

Priority Level	Traffic Type	Applications	Traffic Attribute
0 (lowest)	Best Effort		
1	Background		
2	Standard(Spare)		
3	Excellent Load	Business Critical	
4	Controlled Load	Streaming Multimedia	
5	Voice and Video	Interactive Media and Voice	Less than 100ms latency and jitter
6	Layer 3 Network Control Reserved Traffic		Less than 10ms latency and jitter
7 [highest]	Layer 2 Network Control Reserved Traffic		Lowest la- tency and Jitter

표 1. QoS 우선순위 계층

4. 요약

본 절에서는 연구의 배경이 되는 무선 애드 혹 네트워크에서 MANET과 VANET에 대하여 1절에서 설명하였고, 정책 기반 네트워크 관리의 표준화된 구조에 대하여 2절에서 설명하였다. 그리고 실제 트래픽 차등화 처리 기술인 QoS가 인터넷에서 필요하게 된 이유와 필요한 어플리케이션과 QoS 모델들에 대하여 3절에서 기술하였다. MANET에서의 연구는 상당 부분 VANET에서 적용할 수 있는 기술이지만 VANET은 목표가 뚜렷한 실질적인 MANET의 적용 분야이므로 MANET의 연구를 대부분 수용하지만, VANET에 맞게 추가적으로 연구해야 할 부분도 많다. 또한 COPS 프로토콜의 적용은 특히 VANET과 같이 노드간의 성능이 다른 네트워크에서 필수적으로 COPS-PR 메커니즘을 따라야 할 것이다. 그리고 QoS모델에 대한 연구에서도 VANET에 적용할 모델은 정상시에는 비디오 스트리밍이나 VoIP와 일반적인 인터넷을 통한 트래픽을 차등화하고 위급한 상황이 발생했을 경우 이에 대처할 수 있는 모델이 필요하다. 그러므로 VANET과 같은 노드의 이동성이 높으며, 노드의 밀집도가 높은 MANET에서 정책 기반 네트워크 관리를 적용하기 위해서는 이동하는 PEP 노드가 정책을 수신할 PDP 노드를 능동적으로 탐색하는 메커니즘이 필요하다. 본 연구는 3장에서 이러한 높은 이동성과 밀집도의 변화를 고려한 메커니즘을 기존 연구를 토대로 분석하고 프레임워크를 설계한다.

III. MANET을 고려한 정책 기반 네트워크 관리 프레임워크의 설계

이 장에서는 정책 기반 네트워크 관리 프레임워크에서 어떠한 기능이 필요한지 기술하고, 본 논문의 주요 논제인 표준 정책 기반 네트워크 관리의 구조 중 정책 전송 영역 관리에 관한 두 가지 메커니즘에 대하여 설명한다. 그리고 이러한 시스템에 접근하기 위한 구성 요소와 구현 요소를 정의한다.

1. MANET을 위한 정책 기반 네트워크 관리의 전체 구조

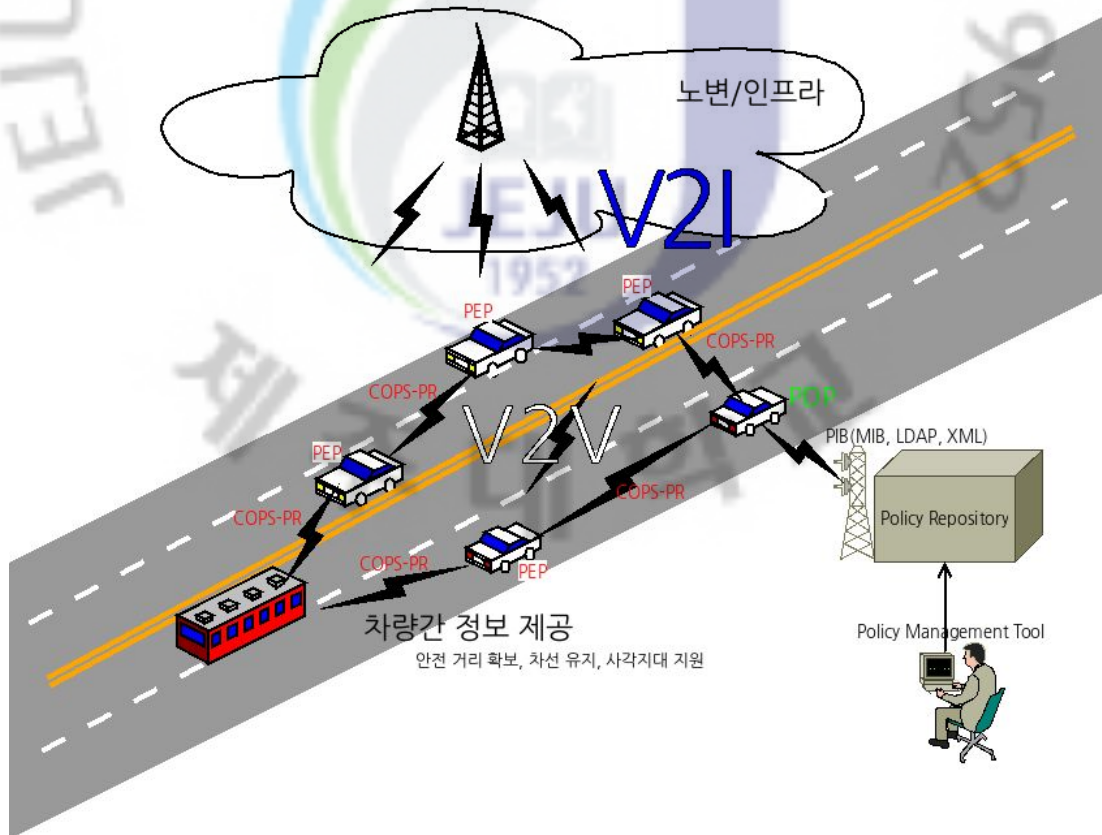


그림 12. VANET에서의 정책 기반 네트워크 관리 구현을 위한 개념도

그림 12.은 MANET의 실질적인 응용 분야인 VANET에 정책 기반 네트워크 관리를 적용할 때의 구성 요소를 예를 들어 표시한 것이다. VANET에서 차량은 MANET의 노드이며, 정책 기반 네트워크 관리의 구조로는 PDP나 PEP가 된다. 차량 간에는 PEP 노드인 차량에서 PDP 차량에게 상태를 보고하고 PDP 차량은 PEP 차량에게 현재 도로 상태나 제반 여건을 고려한 정책을 배포하게 된다. 정책을 수신한 PEP 차량은 수신한 정책 중 트래픽의 조건에 맞는 정책을 적용하여 수행한다.

2. 프로토타입의 분석

MANET의 주요 특징들을 고려하면서, 기존의 유선망에서 시스템 설계 시에 고려되던 것들을, 애드 혹 네트워크에서의 시스템 설계에는 새로운 관점에서 보완해야 한다. 더욱, 무선 애드 혹 네트워크상에서의 관리 프레임워크 개발을 위해서는 다음과 같은 운용적인 측면들이 고려되어야 할 것이다.

- 효율적인 시그널링 메커니즘: 대역폭 제한적인 무선 네트워크에서 시그널링 오버헤드를 최소화하는 것은 중요하다. 이는 링크들이 관리 트래픽으로 혼잡상황에 이르지 않도록 하기 위해서이다. 이러한 측면은 네트워크 관리 시스템에서의 모니터링, 구성 및 제어의 용도로 쓰이는 메커니즘이나 프로토콜의 선택에 있어서 아주 큰 영향을 미치는 요인이다.
- 경량의 프레임워크: MANET은 경량의 관리 프레임워크를 요한다. 자원제한적인 네트워크 노드에 과도한 처리 요구로 부담을 안기지 말아야 한다.
- 자동화, 지능화, 유연화: 애드 혹 네트워크의 역동적 특성을 고려하면, 네트워크 조건의 변화에 자동적으로 대응할 수 있는 적응형 관리 프레임워크가 요구된다. 이를 달성하기 위해서, 관리 시스템은 관련 노드들의 다양한 용량을 자동적으로 파악할 수 있어야 하고, 이 정보를 판별기준으로 이용해서 각기 다른 타입의 노드에 적절한 역할들을 부여할 수 있어야 한다. 정책기반 시스템의 경우, 사람의 개입을 최소화 하면서, 부여된 임무에 맞게 통신용량 및 자원들을 재평

가한 것을 기초로 하여, 네트워크 제어를 자동화할 수 있는 역동적 정책을 정의하고 개발할 필요가 있다.

- 견고함: 관리 프레임워크는 인증 받은 호스트들 사이의 관리 데이터를 안전하게 교환할 수 있게 해야 하고 네트워크의 전반적인 존속성을 향상시켜야 한다. 이는 사용자 및 호스트를 인증하고 권한을 부여하는 도구가 지원되어야 함을 의미하며, 최소한 간단한 암호화 메커니즘을 가지고 있어야 한다는 것이다. 또한, 중앙 집중식 관리 구조는 애드 혹 네트워크에 있어서, 특히 존속성의 관점에서 좋은 접근 방법이 아니다. 존속성을 향상시키기 위해서 분산 구조, 특히 더 가능하다면 계층구조의 관리 프레임워크가 필요하다.

3. 정책 전송 영역 관리 구조

1) k -hop Cluster

정책 기반 네트워크 관리의 구조에서 PDP가 어느 영역의 PEP들에게 정책을 전송할 것인가, 하는 문제에 대한 메커니즘으로 k -hop Cluster 메커니즘이 제안되어 있다. 이 메커니즘은 PDP측에서 정책을 전송하는 PEP가 연결가능한지 주기적으로 광고메시지를 전송하여 응답하는 PEP노드들 중 해당하는 홉 안의 PEP노드들을 선택하게 된다. k 값의 기본값은 1로 설정되어 1홉에 있는 PEP들이 정책을 전송 받는다.

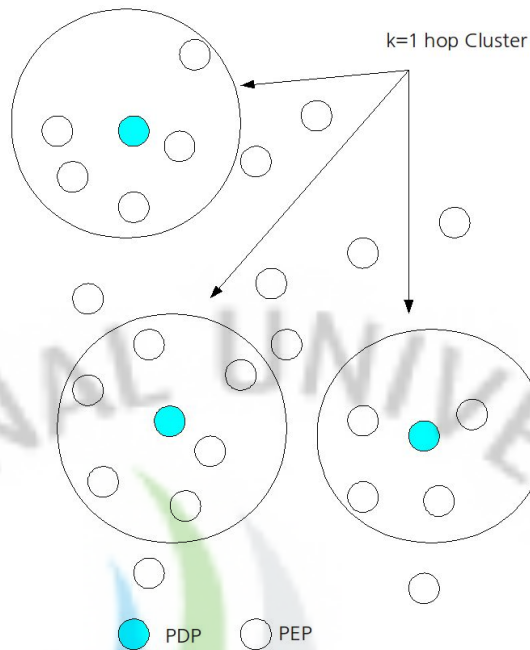


그림 13. $k=1$ hop Cluster

그림 13.는 $k=1$ 일 경우의 PDP가 선택한 PEP 노드들에 대한 개념도이다. k 의 값이 1일 경우 PDP는 홉이 1인 모든 PEP를 관리한다. 이 메커니즘을 VANET에 적용하는 데는 문제점이 있다. 노드가 이동성을 가지는 VANET에서 PDP가 어떻게 적절한 k 값을 정할 수 있는가에 대한 문제와 노드의 이동에 따라 노드의 밀집도 또한 수시로 변할 수 있다. 노드의 밀집도가 높아지면 해당하는 홉수에 있는 PEP노드들도 많아지게 되고 이로 인해 더 많은 노드들이 광고메시지를 수신하게 된다. 이것을 파급 효과(ripple effect)[43]라고 하며, 이로 인한 광고메시지에 의해 네트워크의 성능에 영향을 미치게 된다. 또한 밀집도가 떨어지면 많은 PEP 노드들이 k 홉 밖에 있게 되어 정책을 수신할 수 없게 된다. 또 다른 문제점으로는 밀집도가 높아지면 하나의 PDP에 연결되는 PEP의 수 또한 많아지는 문제가 있다. 이를 해결하기 위해 영역을 내부 영역(Inner region), 외곽 영역(Outer region), 외부 영역(Outside region)으로 나누어 관리하는 방법을 제안하고 있다. 외곽 영역은 파급 효과를 줄이기 위해 사용하고, 외부 영역은 밀집도에 따라 영역을 합치고 분리하는 영역으로 분류하는 것이다. 이에 대해서 2항에서 자세히 다룬다.



그림 14. 파급 효과

2) 능동형 PDP 탐색

능동형 PDP 탐색 기법은 MANET에 대한 정책기반 네트워크 관리를 위해, 애드 홀 네트워크에서 PDP와 PEP 사이의 정책 전달을 위한 방식으로 제안되었다. 이는 PEP가 PDP를 필요할 때 능동적으로 발견 하도록 하여 전체 네트워크에 부하를 줄일 수 있도록 한 것이다.

그림 15는 PEP 노드가 능동형 PDP 탐색을 이용하여 PDP를 발견하고 PDP 선택과정을 거쳐 PDP에 접속하는 절차를 보인다. 네트워크 전체에 분산되어 있는 PDP A, B, C가 있고 각각 PDP에 의해 관리되고 있는 영역은 점선으로 표시되어 있다. PDP정보를 필요 하는 PEP노드 4는 PREQ 메시지를 자신의 이웃 노드 1, 2, 3 에게 1 hop Broadcast 한다. PREQ 메시지를 받은 1, 2, 3 노드는 자신의 현재 정책 서비스를 받고 있는 PDP의 정보를 PREP 메시지에 담아 광고메시지로 보낸다. 노드 1, 2, 3으로부터 받은 PEP 노드 4는 PDP Temporary List 생성 과정을 통해 PDP 리스트가 만들어지고 그 중 우선 순위(Priority)가 가장 높은 PDP A에 COPS-OPN 메시지를 보내고 접속이 성공하면 현재 접속한 PDP 노드의 주소 A와 PDP hop 값 2를 저장 하고 PDP Temporary List는 제거한다.

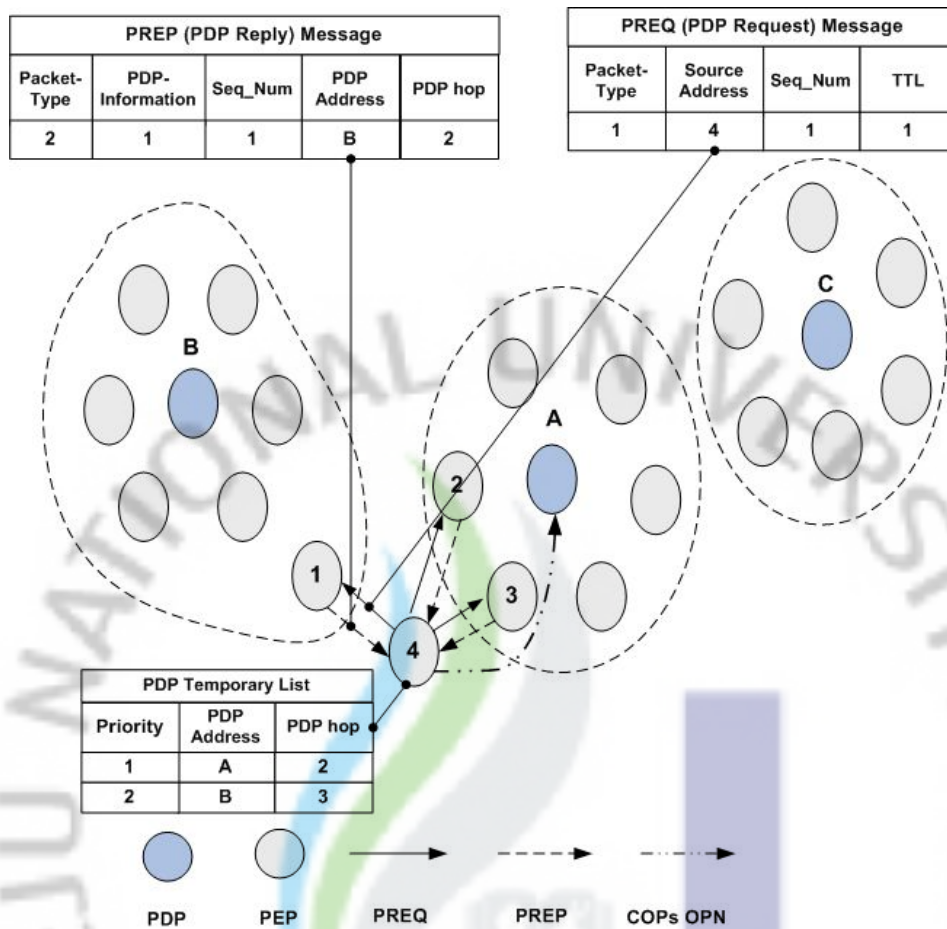


그림 15. 능동형 PDP 탐색과 PDP 선택 과정

이러한 정책기반 애드 혹 네트워크 관리 시스템의 메커니즘을 설계하기 위해 COPS-PR을 확장하고, PDP는 MNL(Management Node List)를 유지함으로써 자신에게 연결된 PEP를 관리하도록 하여 PEP의 이동을 관리할 수 있도록 한다.

```

Function HANDLE_KA(COPS_KA_MSG msg) {
    Pass Out: MNList File
    Update MNList Using msg's information
    SAVE MNList to File
    If (hop is increasing) {
        Send FOP To PEP
    }endIf
} end HANDLE_KA

Function HANDLE_CC(COPS_CC_MSG msg) {
    Delete PEP In MNList
    Call CLOSE_COPS_CONNECTION()
} end HANDLE_CC

```

그림 16. 능동형 PDP 탐색의 PDP 동작

그림 15.는 PDP에 의한 PEP관리와 PEP의 PDP 탐색 과정이다. PDP의 MNL은 PDP가 PEP를 관리하기 위해서 사용되고, 또한 PDP 정보 요청 프로토콜인 PREQ에 대한 응답으로 PREP를 보낼 때 PDP 정보를 제공하기 위해서도 사용된다.

```

Function HANDLE_FOP(COPS_FOP_MSG msg) {
    best_prep = NULL
    Do {
        Send PREQ To k-hop nodes
        Receive PREP From k-hop nodes
        best_prep = Choose best PDP Among PREP messages
    } while (best_prep.pdp_information == NULL)
    Call DO_COPS_CONNECT(best_prep.pdp_addr)
}end HANDLE_FOP

Function DO_COPS_CONNECT(char pdp_addr) {
    Send connect msg To pdp_addr
}end DO_COPS_CONNECT

```

그림 17. 능동형 PDP 탐색에서 PEP 동작

그림 16.에서 능동형 PDP 탐색의 PEP 동작을 보이고 있다. PEP는 FOP 메시지를 받으면 k 홉에 해당하는 노드들에게 PDP 정보 요청 프로토콜인 PREQ를 전송한다. 이에 응답하는 PREP를 다른 PEP나 PDP로부터 수신한 후 PDP Information 필드를 보고 PREP를 보낸 노드가 PDP인지 또는 PEP인지 알 수 있으며, PEP의 경우 COPS 연결을 맺고 있다면 해당 PDP 정보를 송신할 것이며, 연결된 PDP가 없을 경우에는 해당 필드로 NULL로 마킹하여 전송하게 된다. PREQ를 보낸 PEP는 PDP Information 필드가 0이거나 1일 경우에 해당 PDP로 접속하게 되며, NULL인 경우에는 다시 PREQ를 재전송하게 된다.

4. 시스템 접근

본 절에서는 MANET에서의 정책 기반 네트워크 관리의 도입에 대한 연구를 위한 시스템 접근 방법을 시스템의 구성 요소와 구현 요소로 나누어 기술한다.

구성 요소는 제안하는 시스템이 갖추어야 할 기능의 명세이며, 구현 요소는 구성 요소를 만족하기 위해 구현해야 하는 요소들이다.

1) MANET의 정책 기반 네트워크 관리 프레임워크를 위한 구성 요소

MANET의 정책 기반 네트워크 관리를 위해서 다음과 같은 요소를 가지고 있어야 한다.

(1) 정책의 정의

제안하는 시스템은 무선이며 노드가 이동성이 강하며, 노드 간 성능 차이가 있는 특성을 가진 네트워크이므로 PEP노드에서 필요한 정책을 요청할 때 전송하는 COPS-PR의 모델을 적용한다. 이 모델은 PEP의 상태에 따라 예상되는 정책을 미리 전송하여야 한다. 그러므로 COPS-PR 모델에 따른 정책은 PEP의 정책 요청에 따라 최소 두 개 이상의 정책을 전송할 수 있어야 한다.

(2) 정책의 구조와 배포 방법

● 정책의 구조

정책은 정책 관리 도구를 이용해 의사 결정자나 네트워크 관리자가 정의하고, 수정하고, 삭제할 수 있다. 결정된 정책은 별도의 전용 정책 서버에 있을 수도 있고, PDP자체가 가지고 있을 수도 있다. 본 논문에서는 별도의 정책 저장 서버를 두지 않고 PDP가 정책 저장소의 성격을 동시에 수행하는 것으로 한다. 저장되는 정책의 구조는 PIB의 구조에 따라 SMIV2의 형식을 따른 SNMP의 MIB형태로 저장한다.

● 정책 전송을 위한 프로토콜

정책 전송을 위한 프로토콜은 LDAP이나 SNMP를 이용할 수 있으나 본 시스

템에서는 PDP가 정책 저장소의 기능을 함께 수행하므로 별도의 정책 저장소 접근 프로토콜을 사용하지 않고 저장 구조를 직접 접근한다.

(3) 서비스 적용

- 클러스터링과 토폴로지 관리

능동형 PDP 탐색 기법은 PDP에서 PEP까지의 홉을 기준으로 영역을 세 가지로 분류로 나누어 관리한다. $k=1$ 인 홉 들을 내부 영역으로 하여 정책을 전송하고 있는 영역으로 구분하고, $k=2$ 까지의 홉을 가진 PEP 노드들을 외곽 영역으로 하여 영역을 분리하거나 합칠 수 있는 영역으로 구분하고, 그 이상의 영역을 외부 영역으로 분류하여 새로운 영역으로 PDP를 선출하여 관리할 영역으로 한다. 어떻게 PDP에서 PEP까지의 홉 수를 인지할 것인지에 대하여는 아래에서 자세하다.

- 서비스 범위의 탐색

2절의 정책 전송 영역 관리 구조에서 살펴보았듯이 k -hop Cluster나 능동형 PDP 탐색 모두 광고메시지를 이용한다. k -hop Cluster에서는 PDP가 주기적으로 광고메시지를 전송하여 주변 PEP까지의 홉을 알아내며, 능동형 PDP 탐색에서는 PEP가 필요할 때, 역시 광고메시지를 전송하여 주변의 PDP 정보를 알아낸다.

하지만, MANET은 자율적으로 토폴로지를 구축하는 특성을 가진 네트워크이며, TCP/IP 바탕위에 MAC 기반 라우팅을 하여 멀티-홉 토폴로지를 구성한다. 즉, IP 네트워크에서의 라우팅 개념은 MANET에서는 다른 의미이다. IP 네트워크에서 서브넷이 다를 때에 라우팅이 필요해 지지만, MANET에서는 같은 서브넷에 있는 노드들도 다른 노드를 통하여 연결할 수 있어야 하므로 IP 네트워크의 특성을 가지고 있으면서도 내부적으로 MAC 기반 라우팅을 한다. 그러므로 능동형 PDP 탐색 기법의 실제 구현에서 있어서는 MANET의 특성이 자율적인 토폴로지 구성 특성을 그대로 이용하여 주변 PDP정보를 알아내도록 하였다. 이에 대하여 4장 실험 환경의 구현과 5장 실험 결과와 분석에서 자세하다.

(4) 자원의 탐색

본 시스템에 적용할 라우팅 프로토콜은 멀티-홉 라우팅이 필요하며, 라우팅 프로토콜에 의해서 토폴로지 정보가 제공되어야 한다. 제안하는 프레임워크에서는 COPS의 연결 유지 메시지인 KA(Keep-Alive)메시지를 통해 토폴로지 정보가 주기적으로 교환되어야 함으로 이 정보가 필요하다.

(5) 정책 프로비저닝

정책 프로비저닝을 위해서 하나의 정책은 다음과 같은 속성을 가지고 있어야 한다.

- marking : 정책은 다른 트래픽과 구별될 수 있도록 표시할 수 있어야 한다.
- classification : 정책은 트래픽의 표시에 따라 분류될 수 있어야 한다.
- queuing : 분류된 정책을 처리하는 서로 다른 큐잉을 가지고 있어야 한다.
- policing : 상기한 세 가지 속성에 따라 일관된 처리를 할 수 있어야 한다.

(6) 정책 기반 라우팅

운영체제의 커널에서 정책에 따른 트래픽 처리가 가능하여야 한다. 이에 따라 (5)목의 정책 프로비저닝에서 거론한 속성에 대해 실제 처리를 할 수 있어야 한다.

(7) 정책 모니터링

트래픽이 정책에 의해서 처리되는 것을 모니터링 할 수 있어야 한다. 전송된 패킷의 갈무리와 실시간 트래픽을 모니터링 할 수 있어야 한다.

2) MANET의 정책 기반 네트워크 관리 프레임워크를 위한 구현 요소

MANET의 정책 기반 네트워크 관리를 위해서 다음과 같은 구현 요소들이 필요하다.

- 정책 관리 도구
- 정책 저장소
- COPS-PR 프로토콜
- 능동형 PDP 탐색
- QoS

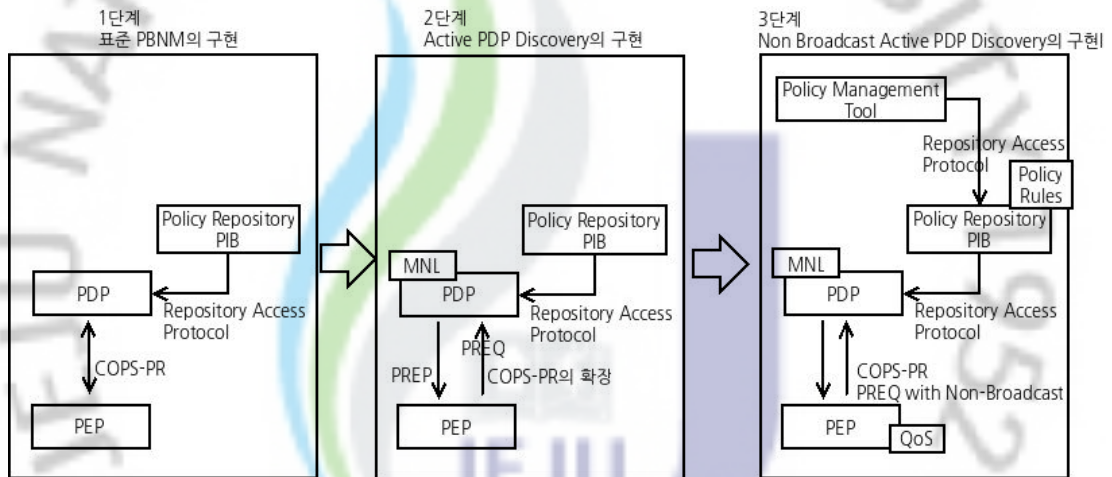


그림 18. 단계별 구현 개념도

그림 16.에서 본 연구에서 표준 정책 기반 네트워크 관리의 구성 요소 중 COPS-PR을 중심으로 단계적으로 구현하고 절차를 보인다. 표준 정책 기반 네트워크 관리에서 정의한 COPS-PR과 PIB를 먼저 구현하고, COPS-PR 연결 과정에서 MANET에서 필요한 PDP 탐색 알고리즘을 적용하여 능동형 PDP 탐색을 구현한다. 그리고 PEP의 PDP 탐색 과정에서 광고메시지를 이용하지 않고 PDP 정보를 요청하도록 하며, 이전 단계에서 구현한 PIB를 확장하여 외부에서 정책을 정의할 수 있도록 한다. 이를 위해 정책 관리 도구를 구현하고, 정의되고 전송된 정책에 의하여 실제 트래픽이 전송되어 질 수 있도록 QoS를 적용한다. 그러므로 각각의 구현 요소들은 다음과 같이 표준 정책 기반 네트워크 관리의 구성 요소를 구현한다.

(1) 정책 관리 도구

정책을 결정, 저장, 수정할 수 있도록 한다. 인터페이스는 커맨드 라인, 로컬 GUI용 어플리케이션 개발, 웹 인터페이스의 활용을 들 수 있다.

(2) 정책 저장소

정책 저장소에서는 별도의 서버를 두지 않고 PDP에서 PIB를 구성한다.

(3) COPS-PR 프로토콜

표준 COPS-PR 프로토콜 사양에 맞는 프로토콜로 특정 PDP에서 실행하고, PEP노드의 요청에 따라 접속을 허용하며, 접속된 PEP의 요청에 따라 정책을 전송한다.

(4) 능동형 PDP 탐색

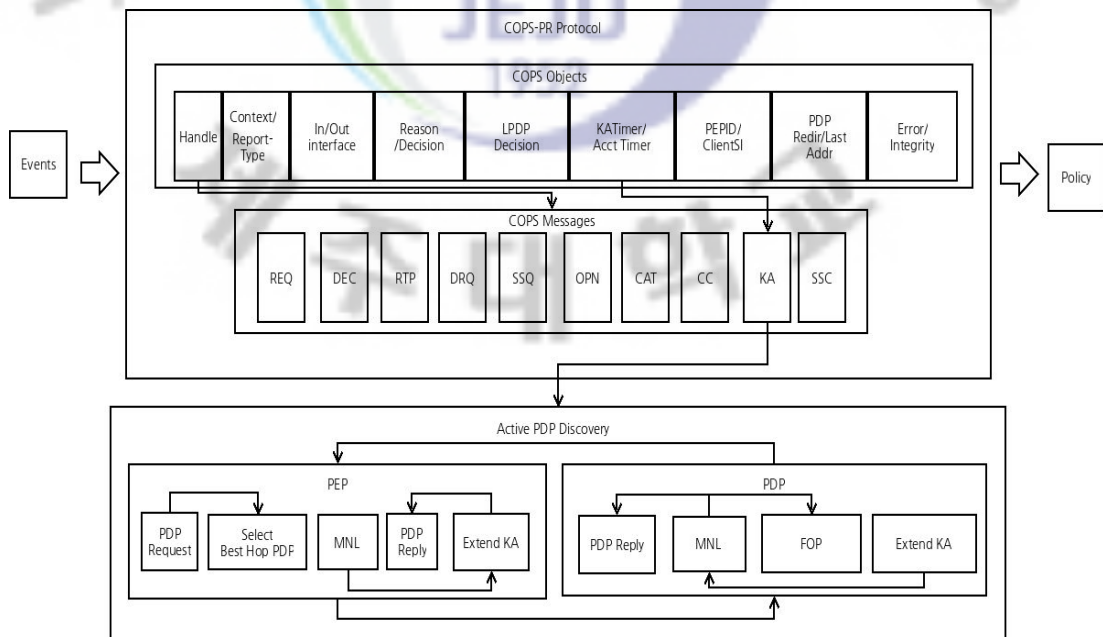


그림 19. COPS-PR과 능동형 PDP 탐색 구성도

그림 19.에서 COPS-PR과 능동형 PDP 탐색의 구성을 보인다. 표준 COPS-PR 프로토콜의 객체들은 COPS 메시지를 생성하며, 이 메시지 중 연결 유지 목적인 KA 메시지를 확장하여 PDP 정보를 송·수신하도록 하고, 전송된 정보를 MNL로 저장하여 PEP의 PDP 요청 프로토콜인 PREQ에 응답하는 PREP에 의해 전송되도록 한다. 능동형 PDP 탐색 기법의 구현 요소는 아래와 같다.

- COPS-PR 프로토콜의 확장

능동형 PDP 탐색 기법에서는 표준 COPS-PR의 KA 메시지를 확장하여 PDP와 PEP간의 홈 정보를 전송하여야 한다.

- PREQ, PREP

능동형 PDP 탐색 기법에서는 새롭게 네트워크에 참여하거나 기존 정책 전송 영역에서 떨어진 PEP노드가 기존 연결된 PDP로부터 FOP(Find Other PDP) 메시지를 받으면, 새로운 PDP노드를 탐색(PREQ)해야 한다.

이동 PEP 노드의 PREQ를 받은 PDP는 자신의 IP와 홈 정보를 담은 정보로 응답하여야 하며, PREQ를 받은 노드가 PEP라면 자신이 연결된 PDP정보를 전송하여야 한다.

PREP를 받은 이동 PEP 노드는 기 연결된 PDP 노드와 탐색 결과로 찾은 PDP간의 홈 정보를 비교하여 홈 정보가 작은 쪽으로 COPS 연결을 한다. 그리고 기존 연결되었던 PDP로는 CC(Connection Close)신호를 보낸다.

CC 메시지를 받은 PDP는 자신의 Management Node List에서 해당 노드를 삭제한다.

- MNL(Management Node List)

확장된 KA 메시지에 의해 전송된 PEP 노드 정보들은 PDP 노드에서 MNL로 저장되고 주기적으로 갱신된다. MNL에 저장된 PEP 노드들 중 이동을 감지(홈 수가 늘어나는 노드)한 노드에 대하여 FOP 메시지를 보낸다.

(5) QoS 기반의 트래픽 처리

운영체제 커널 차원에서 진입(Ingress) 트래픽에 대해 트래픽의 마킹이 이루어져야 하며, 발신(Outgoing) 큐에 대하여 정책이 적용되어 처리되어야 한다. 이를 위해 리눅스 운영체제 상에서 구현된 QoS 모델들을 이용한다.

리눅스 운영체제 구현된 QoS 모델들을 아래와 같다.

- FIFO(First IN First Out) Queuing
- CBQ(Class Based Queuing)
- TBF(Token Bucket Flow) Queuing
- SFQ(Stochastic Fair Queuing)
- RED(Random Early Detection) Queuing
- GRED(Generalized RED) Queuing
- HTB(Hierarchical Token Bucket) Queuing
- DSMARK(DiffServ Maker)

상기한 QoS 모델들은 운영체제 커널에서 모듈형태로 제공되며, IProute2 패키지[44]에 의하여 접근 가능하며, 실제적으로 tc(traffic control) 명령어가 커널에서 구현한 QoS API를 이용하여 컨트롤하게 된다.

QoS 모델들 중 MANET에 적용할 수 있는 것은 HTB(Hierarchical Token Bucket)와 DSMARK 모델을 들 수 있다. HTB 모델은 TCP port를 대상으로 트래픽의 분배를 총 대역폭에 대한 비율로 정할 수 있으며, 이것은 대상 네트워크 전체에 같은 비율로 분배되므로 정책 기반 망 관리 시스템의 구현에서 정책의 결정이 쉽고 전체 네트워크에 정책을 배포하기 쉬운 장점이 있다. 하지만, VANET과 같은 네트워크에서는 응급 재난 관리 등 평상시에는 트래픽에 특성에 따른 적절한 분배를 하고 특정 상황이 발생하는 경우에는 다른 트래픽보다 특별히 높은 우선권을 가지는 QoS모델이 필요하다.

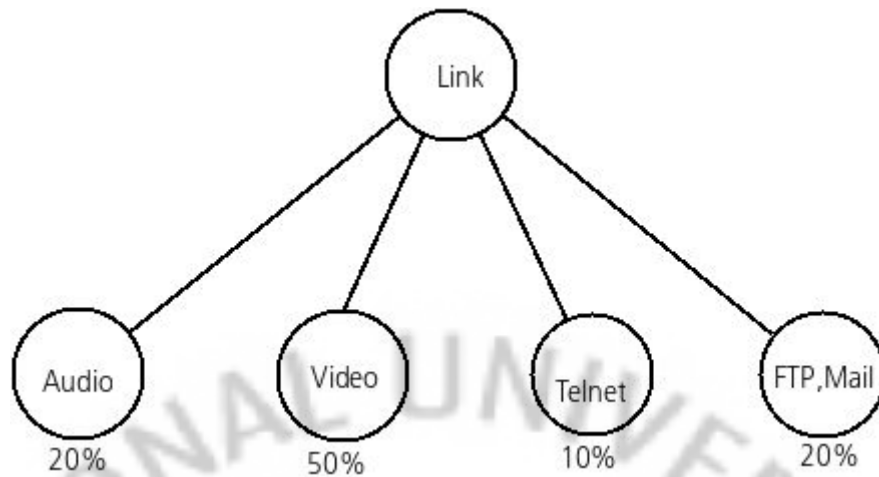


그림 20. Class Based Queuing

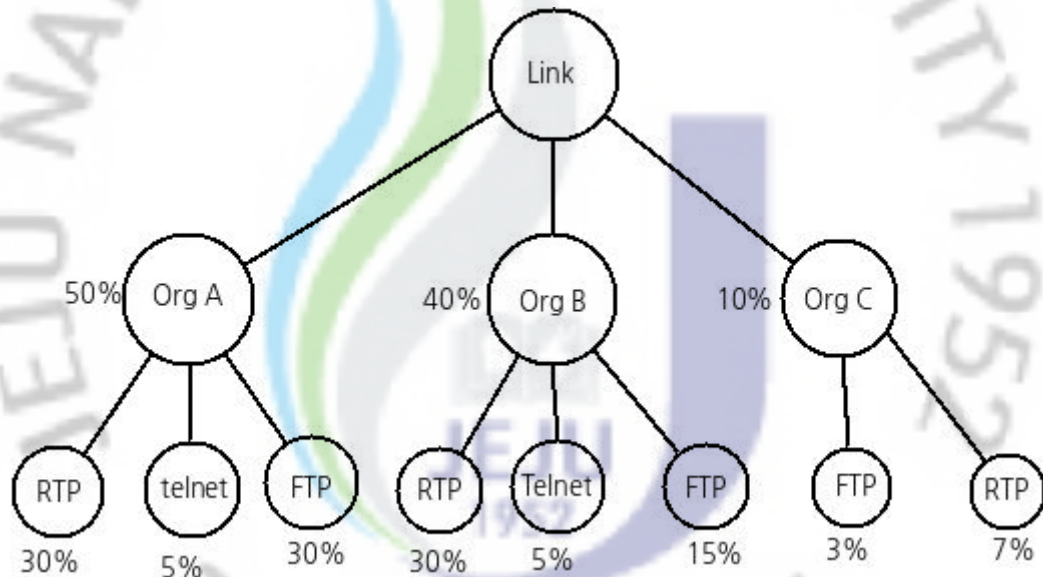


그림 21. HTB에 의한 대역폭 분배

그림 20.은 트래픽 특성에 따른 대역폭의 분배를 나타내는 CBQ이며, 그림 21.은 CBQ를 확장하여 HTB에 의해 하위 네트워크 전체에 결정된 정책에 따라 트래픽이 분배되는 것을 도식한 것이다.

DSMARK 모델은 TCP헤더의 DSCP 필드 마킹에 따라 몇 가지 클래스로 분류하고 그에 따른 전송 우선순위를 두는 방식이다.

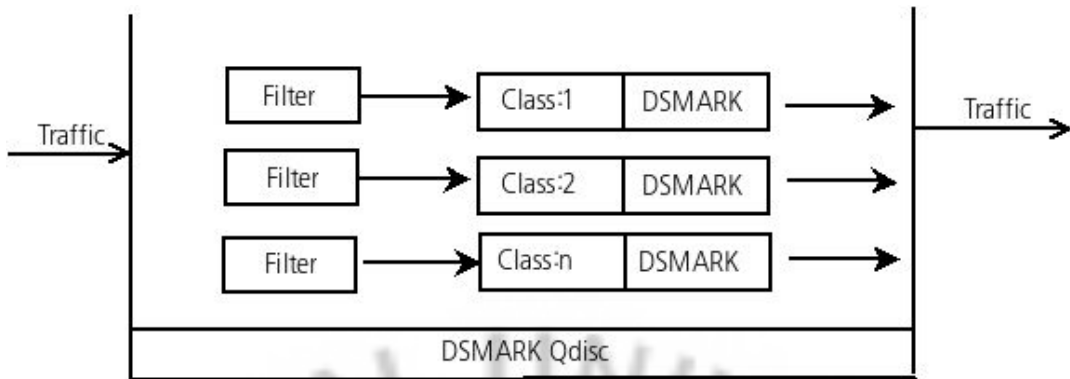


그림 22. DSMARK 동작 개념

그림 22.은 DSMARK의 동작 개념도이다. DSCP 필드에 의해 필터링 된 트래픽들은 미리 정의된 클래스에 의해 분류되고 정책에 의해 전송된다.

Drop	Class 1	Class 2	Class 3	Class 4	EF
Low	001010	010010	011010	100010	101110
	AF11	AF21	AF31	AF41	
	DSCP 10	DSCP 18	DSCP 26	DSCP 34	
Medium	001100	010100	011100	100100	
	AF12	AF 22	AF32	AF42	
	DSCP 12	DSCP 20	DSCP 28	DSCP 36	
High	001110	010110	011110	100110	
	AF13	AF23	AF33	AF43	
	DSCP 14	DSCP 22	DSCP 30	DSCP 38	

표 2. DSMARK의 클래스 분류

표 3에서 DSMARK의 표준 클래스를 보이고 있다. 클래스는 크게 AF(Assured Forward)과 EF(Expedited Forwarding) PHB(Per Hop Behavior)를 가진다. AF는 다시 세 단계로 나뉘고 나뉜 각 단계는 세부적으로 세 개의 클래스를 추가로 정의할 수 있다. 각 클래스는 비디오나 오디오, 일반적인 트래픽의 분류로 사용될 수 있고, 이 중 EF 클래스는 다른 트래픽을 드롭 시키고 우선 전송되는 특성을 가진 클래스이다. 즉, 다른 클래스들이 대역폭을 정의된 정책에 의하여 비율로 분배하지만, EF 마킹된 트래픽은 전체 대역폭의 30%를 그 즉시

사용하는 특성을 가지고 있다.

5. 요약

본 장에서는 표준 정책 기반 네트워크 관리의 구조에 대하여 설명하고, 그 구조에서 이 연구가 중점적으로 연구한 정책 전송 영역에 관한 두 가지 제안된 구조에 대하여 2절에서 설명하였다. 그리고 VANET에서 정책 기반 네트워크 관리 시스템을 구현하기 위한 구성 요소와 구현 요소에 대하여 어떠한 것이 있는지 정리하였다. 연구를 위한 시스템 접근으로 목적에 맞는 정책의 구조와 배포 방식을 가지고 있어야 하며, 서비스 적용과 자원의 탐색, 정책 프로비저닝과 정책 기반 라우팅, 정책 모니터링을 포함한 시스템이 필요하며, 이를 위한 구현 요소로써 정책 관리 도구, 정책 저장소, COPS와 그에 관련한 프로토콜과 정책 적용 트래픽의 처리에 관한 요소가 필요하다. 그리고 실제 트래픽 처리에 있어서는 리눅스 운영체제에서 구현된 QoS 모델들을 분석한 결과 VANET에 적합한 모델로 DSMARK로 결정하였다.

IV. 구현

본 장에서는 구현 과정을 밝힌다. 하드웨어와 소프트웨어 환경의 구축 과정을 1절과 2절에서 각각 다루며, 핵심 내용인 COPS-PR의 구현을 3절에서 다루고 구현한 COPS-PR을 확장하고 PREQ, PREP 에이전트를 개발하여 능동형 PDP 탐색 기법을 구현한 과정을 4절에서 설명하고 제안된 메커니즘의 설계의 개선의 이유와 각각의 동작과정을 보인다. 5절에서는 적용하려는 QoS 모델과 수신된 정책에 따라 해당하는 QoS 모델이 실제 동작하도록 하는 과정을 설명한다. 6절에서는 정책 관리 도구를 구현하는 과정을 보이고 7절에서 이 장을 요약한다.

1. 하드웨어 환경 구축

1) 하드웨어

실험 환경을 위한 하드웨어로는 5대의 랩톱 PC를 이용하여 구성하였다. 다섯 대의 랩톱 PC들은 각각 아래와 같은 사양을 가지고 있다. AMD64계열의 2.2GHz의 CPU, Broadcom사의 BCM4318 칩셋을 사용하는 IEEE 802.11/bg 두 대 Intel Atom CPU를 사용하는 1.6GHz, Broadcom사의 BCM 4312 칩셋을 사용하는 IEEE 802.11/bg 두 대 Intel Centrino CPU를 사용하는 1.4GHz, 두 대의 PC는 PDP로 이용하고 나머지 랩톱 PC들은 PEP로 구성하였다.



그림 23. 알루미늄 포일로 전송거리를 줄인 노트북들

하드웨어 벤더들이 리눅스 드라이버를 제공하지 않거나 드라이버의 제작 사양을 제공하지 않아 그 대안으로 역 엔지니어링(Reverse engineering)을 이용하거나 Microsoft사의 Windows® 운영체제의 드라이버를 사용하는 Ndiswrapper를 이용할 수 있다. 하지만, 이렇게 구현된 드라이버들을 표준 파라미터들 중 몇 가지를 지원하지 않는 경우가 많았다. 이는 일반적인 사용에는 문제가 없으나 구현하려는 환경은 드라이버에 매우 민감하므로 사용할 수 없었다. 우분투 리눅스의 커뮤니티 문서[45]와 기타 공개 문서를 참고하여 몇 가지의 무선 NIC를 실험하였다. zd1211rw 칩셋을 사용하는 A-Link사의 WLAN54USB, Broadcom사의 BCM4311의 Ndiswrapper 드라이버와 역 엔지니어링에 의해서 구현된 리눅스 네이티브 드라이버, Netgear사의 리얼텍 rtl8187 칩셋을 사용하는 WG11v3의 Ndiswrapper 드라이버, Intersil Prism 칩셋을 사용하는 무선 PCMCIA 모델 등이 실험되었다. 최종적으로 리눅스의 전용 드라이버가 지원되고 여러 가지 표준 파라미터가 모두 제공되며, 비교적 수급이 쉬운 제품은 Intel사의 PRO/Wireless2200BG 칩셋을 사용하는 Mini-PCI 형태의 제품을 찾을 수 있었다. 하지만, 하드웨어 인터페이스 형태가 Mini-PCI이므로 이를 지원하지 않는 랩톱PC에는 적용할 수 없었고, 실험 환경에서 이동 노트북으로 사용할 랩톱PC에 이

NIC으로 교체하였다.

2) 하드웨어 파라미터

실험 환경의 물리적인 범위를 축소시킬 필요가 있다. 이를 위해 그림 23와 같이 랩톱PC를 분해하여 무선 NIC 모듈에서 안테나를 분리하고 알루미늄 포일로 감싸고, 무선 NIC모듈의 분리가 불가능한 랩톱PC는 메인보드로부터 안테나가 연결된 부분까지 역시 알루미늄 포일로 감싸서 물리적인 전송 범위를 축소시켰다.



그림 24. 랩톱PC의 NIC 모듈에서 안테나 제거 후 포일로 감싼 부분

그림 24.은 전송 속도를 낮추기 위해 랩톱 PC의 NIC 모듈에서 안테나 연결을 해제하고 해당 부분을 알루미늄 포일로 감싼 모습이다.

다음은 하드웨어 드라이버의 설정값을 변경하여 전송 거리를 줄일 수 있는 부분이 있는지 조사하였다.

```

iwlist [interface] scanning [essid NNN] [last]
[interface] frequency
[interface] channel
[interface] bitrate
[interface] rate
[interface] encryption
[interface] keys
[interface] power
[interface] txpower
[interface] retry
[interface] ap
[interface] accesspoints
[interface] peers
[interface] event
[interface] auth
[interface] wpakeys
[interface] genie
[interface] modulation

```

그림 25. 드라이버의 설정 가능한 값

그림 25는 리눅스 운영체제에서 무선 NIC의 설정 가능한 파라미터들을 보여주고 있다. 여러 가지 설정 가능한 파라미터 값 들 중 전송 성능에 영향을 줄 수 있는 것은 Bit Rate, Tx-Power이며 Bit Rate 값은 무선 애드 혹 네트워크인 경우 상호 협상에 의해 낮은 전송 속도를 가지는 노드 쪽으로 맞추어 지므로 별도로 설정할 필요가 없었으며, 드라이버 설정 값을 조정하여 전송 범위를 줄이는 것은 결국 Tx-Power 값을 조정함으로써 원하는 결과를 얻을 수 있었다. Tx-Power는 기본 값이 20dBm으로 이 값은 드라이버마다 낮출 수 있는 최대치가 다르다. 본 연구를 위한 하드웨어 구성에서는 드라이버가 허용하는 최대치까지 값을 낮추었다.

```

wlan0 IEEE 802.11bg ESSID:"VANET"
Mode:Ad-Hoc Frequency:2.412 GHz Cell: 4E:B5:BB:44:4F:84
Tx-Power=20 dBm
Retry min limit:7 RTS thr:off Fragment thr=2352 B
Power Management:off
Link Quality:0 Signal level:0 Noise level:0
Rx invalid nwid:0 Rx invalid crypt:0 Rx invalid frag:0
Tx excessive retries:0 Invalid misc:0 Missed beacon:0

```

그림 26. 무선 NIC의 드라이버 기본 세팅

```
wlan0 IEEE 802.11bg ESSID:"VANET"
Mode:Ad-Hoc Frequency:2.432 GHz Cell: 3E:0B:03:F9:5B:4C
Tx-Power=-44 dBm
Retry min limit:7 RTS thr:off Fragment thr=2352 B
Power Management:off
Link Quality:0 Signal level:0 Noise level:0
Rx invalid nwid:0 Rx invalid crypt:0 Rx invalid frag:0
Tx excessive retries:0 Invalid misc:0 Missed beacon:0
```

그림 27. Tx-Power 값을 조정 후 드라이버 설정 값

드라이버가 낮출 수 있는 최대치는 드라이버마다 상이하므로 드라이버의 개발 문서가 있으면 그것을 참조하고 없다면 소스를 참고하거나 직접 설정값 조정을 반복하여 해당 설정 값이 최대치를 찾아 설정하였다. 그림 26.는 드라이버의 기본 세팅값을 보여주는 그림이며, 그림 27.은 Tx-Power 설정값을 조정한 후의 드라이버 상태에 대한 설정 값이다.

위의 두 가지의 작업으로 인해 해당 랩톱PC마다 NIC의 성능과 드라이버의 완성도에 차이가 있으므로 일정치는 않지만, 노드 간 전송 범위는 대략 5미터 내외가 되게 할 수 있었다.

2. 소프트웨어 환경 구축

1) 운영체제와 QoS 트래픽 제어

실험 환경의 모든 랩톱PC들은 리눅스 커널 2.6[46]기반으로 설치하였다. 리눅스 커널 버전 2.6은 2003년 12월에 발표되었으며, 2009년 11월 현재까지 안정 커널의 최신 버전이다. 이 커널은 대용량 시스템에서부터 임베디드 시스템에 이르는 다양한 하드웨어를 지원하고 하이퍼쓰레딩이나 인텔사의 PXE와 같은 하드웨어 확장을 지원한다. 리눅스는 어떤 행동에 대하여 응답에 대한 시간이 예측가능하지 않기 때문에 리얼타임 운영체제가 아니지만, 상호작용성과 응답성의 개선이 이루어져 있다.

리눅스 커널에서 제공하는 QoS 모듈과 이 모듈을 제어할 수 있는 명령어 패키지를 이용하였다. QoS에 관한 부분은 5절에서 자세히 설명한다.

2) MANET 라우팅 프로토콜

네트워크의 구성은 애드 혹 모드로 하고, MANET 전용의 라우팅 소프트웨어로 OLSRD(Optimized Link State Routing Daemon)[47]를 이용하였다. OLSR은 최초에 INRIA의 연구원에 의해서 시작되었으며, 400 노드, 600 노드에서 2,000 노드에 이르는 실제 네트워크에서 동작하는 예가 있다. 특징적인 기능으로 부가 기능들이 개발 가능하여 여러 가지 방법으로 라우팅 상태에 대한 모니터링이 가능하다.

[olsr.org OLSR daemon](http://olsr.org) 

Configuration		Routes		Links/Topology		All		About	
Links									
Local IP	Remote IP	Hysteresis		LinkCost					
192.168.0.101	192.168.0.102			0.00		(0.996/0.890) 1.128			
Neighbors									
IP Address	SYM	MPR	MPRS	Willingness	2 Hop Neighbors				
192.168.0.102	YES	NO	YES	3	IP ADDRESS ▾ (0)				
Topology Entries									
Destination IP	Last Hop IP		Linkcost						
192.168.0.102	192.168.0.101		(0.996/0.890) 1.128						
192.168.0.101	192.168.0.102		(0.890/0.984) 1.141						
MID Entries									
Main Address	Aliases								

그림 28. OLSRD httpinfo plugin

그림 28.는 OLSRD의 httpinfo 부가기능으로 간단한 웹서버 기능을 제공하고 전체적인 OLSRD의 모니터링 기능과 몇 가지의 파라미터 값을 변경할 수 있다. 또한 실시간 Link Quality를 이용한 네트워크 위상정보를 그림파일로 제작할 수 있는 부가 기능을 이용, 프로그램을 작성하여, 노드의 이동에 따른 위상 변화를 추적할 수 있도록 하였다.

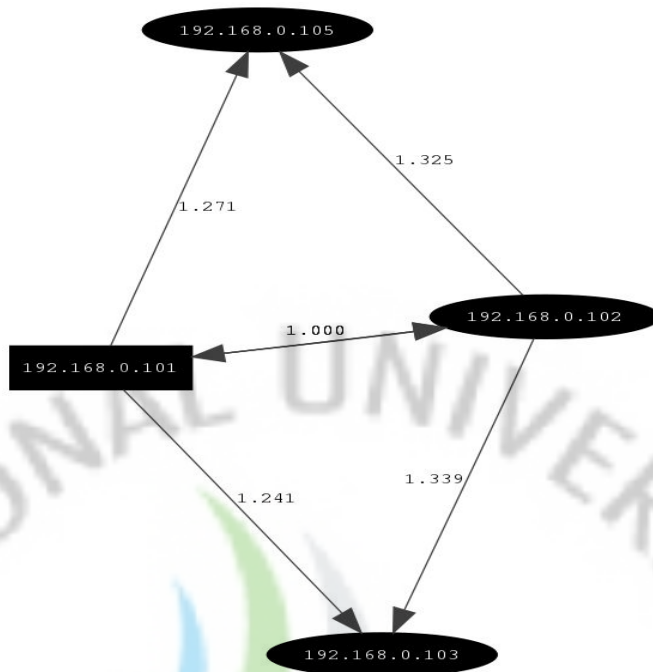


그림 29. topology_view 스크립트에 의한 출력물

그림 29는 IP 주소 192.168.0.101에서 본 연결된 다른 노드들을 보여주고 있다. 링크사이에 표시된 숫자는 연결된 신호의 세기이다.

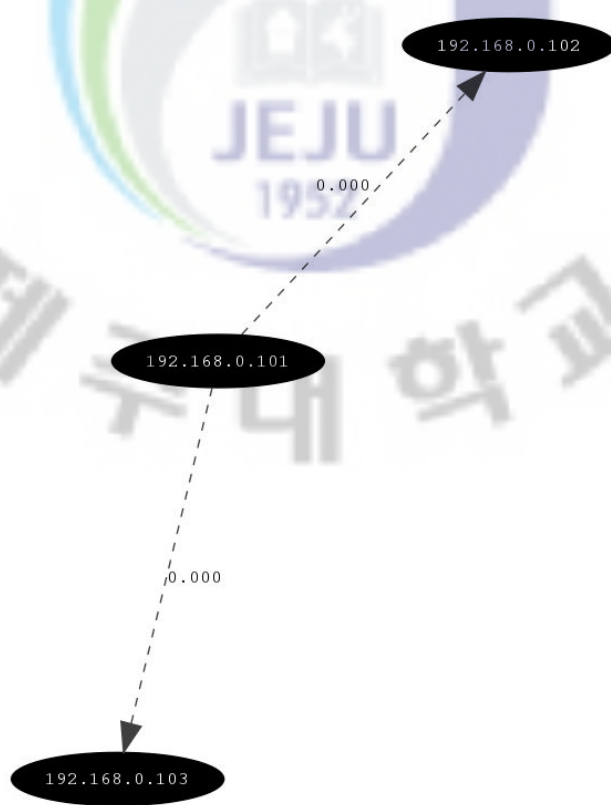


그림 30. 네트워크 토폴로지 변화의 측정

그림 30.에서 노드의 이동을 보여 주고 있다. 이 스크립트로 주기적으로 실행하여 노드의 이동에 따른 네트워크의 토폴로지 변화를 알아내었다.

3) COPS 프로토콜의 구현

COPS-PR의 구현에 있어서는 유선 네트워크상에서 구현된 것들로 아래와 같은 것들이 있다.

- Intel® COPS Client SDK[48]

인텔사의 엔지니어에 의해서 제작되었던 소프트웨어 SDK(Software Development Kit)으로써 COPS와 COPS-PR 프로토콜을 구현할 수 있도록 되어 있다고 한다. 하지만, 저작권의 문제로 사용할 수 없었다.

- Telia 연구소의 COPS 구현[49]

MANET에서의 정책 기반 네트워크 관리에 대한 최초 구현한 관련 연구[5]에서 사용된 것이다. 스웨덴의 Telia 연구소와 Luleå University of Technology에서 구현한 것이다. 하지만, 구현된 소스를 찾을 수 없었다.

- 그 외의 COPS 구현

다른 구현으로 University of New South Wales의 COPS-PR 구현[50]과 Vovida 연구소의 COPS 구현이 있다. 두 소스 모두 이미 삭제된 상태이며, 다른 곳에서 찾을 수 없었다. 특히 Vovida 연구소의 COPS 구현은 가장 최근의 COPS 구현으로 근래의 관련 연구[51]에서 사용되어 왔지만 더 이상 배포하지 않는 것으로 보인다. 이메일을 통해서 문의했지만, 답신은 오지 않았다.

- 실험 환경에 사용된 COPS 구현

실험에는 결국 핀란드의 Tampere University of Technology의 Faster project의 일환으로 구현한 COPS 프로토콜 소스[52]를 참조하였다. 이 구현은 표준에 충실하고 표준 C 프로그램을 제작되었으며, GPL 저작권으로 되어 있어 자유롭게 사용할 수 있다.

COPS-PR 프로토콜의 실제 구현에 관련한 부분은 3절 COPS-PR의 구현에서 자세히 다룬다.

4) 측정 및 분석 도구들

측정 및 분석 도구들로 다음과 같은 도구들을 사용하였다.

- Iperf[53] : 능동형 대역폭 측정 도구이다. 하지만, 본 연구에서는 트래픽 발생기로도 사용하였다.
- Wireshark[54] : 오랫동안 패킷의 분석 도구로 사용되어 왔던 ethereal에서 발전한 것으로 본 연구에서는 패킷 분석 및 그래프 생성 도구로도 사용하였다.
- VLC[55] : 비디오 스트리밍 서비스를 통해 구현된 테스트베드의 동작을 실험하기 위해 사용되었다. 비디오 재생기의 역할을 하면서도 스트리밍 서버와 클라이언트의 역할도 동시에 수행할 수 있다.

3. COPS-PR의 구현

이 절에서는 4장 2절 3항에서 설명하였던 Tampere University of Technology의 COPS 구현 소스를 이용하여 COPS-PR을 구현한 과정 중 제안하는 시스템에 연관된 부분에 대하여 설명한다.

1) COPS 프로토콜

(1) Keep-Alive

표준 COPS 프로토콜의 구조에서 PDP는 연결 대기 상태에서 시작하고 PEP가 연결 요청을 하면 COPS-OPN(OPEN) 메시지를 전송하여 연결하게 된다. 연결이 완성되면 PDP는 정책 요청을 기다릴 준비가 된 것이며, PEP가 정책 전송 요청을 하지 않고 계속 COPS 연결 상태에 있으면, KA(Keep-Alive) 메시지를 송신하여 PDP와의 연결 유지를 하게 된다. 이 과정을 도식하면 그림 31.와 같다. TCP 포트 2195번으로 PEP가 연결 요청을 하고 TCP 연결이 끝나면, COPS 연결을 그림 31.과 같이 이어가게 된다.

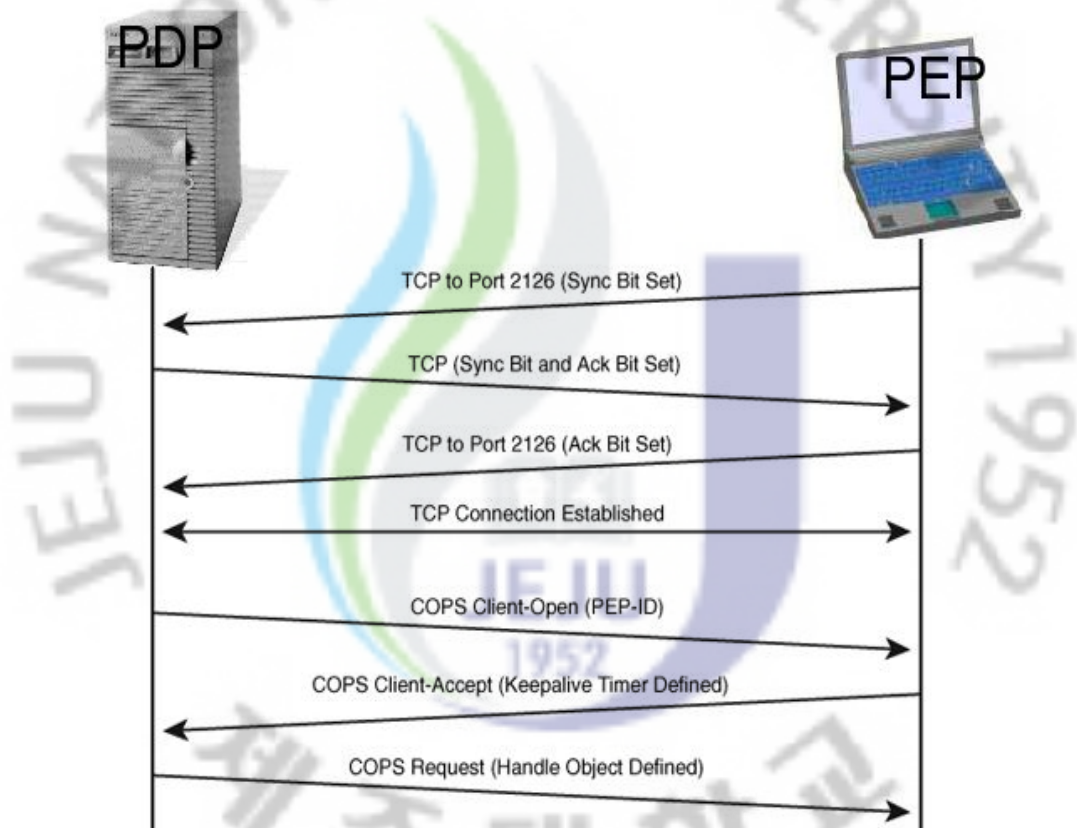
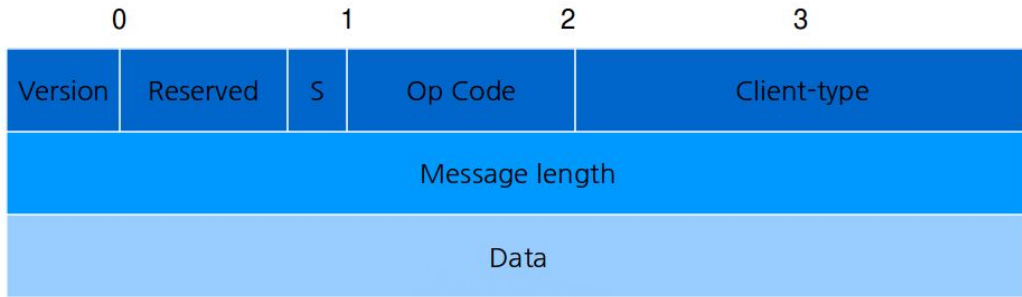


그림 31. COPS 프로토콜에 의한 연결 과정



Op Code : 8 bits
The COPS Operations :

- | | |
|---------------------------|-----|
| 1 – Request | REQ |
| 2 – Decision | DEC |
| 3 – Report State | RPT |
| 4 – Delete Request State | DRQ |
| 5 – Synchronize State Req | SSQ |
| 6 – Client-Open | OPN |
| 7 – Client-Accept | CAT |
| 8 – Client-Close | CC |
| 9 – Keep-Alive | KA |
| 10 – Synchronize Complete | SSC |

그림 32. COPS의 구조

그림 32.은 표준 COPS의 구조이다. Version 필드는 4비트로 COPS 프로토콜의 버전을 나타낸다. Reserved 필드는 0으로 세팅되며, S 필드는 1비트로 다른 COPS 메시지들을 호출하기 위한 필드이다. Op code는 그림과 같으며, Client-type 필드는 16비트의 길이를 가지며, 0x0001은 RSVP, 0x0002는 DiffServe QoS를 지시하고, 0x8001부터 0x8009까지는 엔터프라이즈 환경의 사양을 지시하는 플래그로 되어 있다. 이 메시지가 KA(Keep-Alive) 메시지로 쓰일 경우에는 Client-type은 항상 '0'으로 세팅하는데 그 이유는 KA메시지가 연결 유지의 목적으로 쓰이기 때문에 클라이언트의 종류를 명시할 필요가 없기 때문이다. Message length 필드는 총 메시지의 길이를 명시하고, Data 필드는 하나 이상의 정책 개체를 포함한다. Op code가 9인 경우 KA메시지는 COPS 프로토콜에 있어서 단순한 연결 유지를 확인하기 위한 용도로 사용된다.

PEP에서 KA메시지의 전송은 Timer를 두어 KA 메시지 전송 함수에서 주기적으로 호출하는 형태로 되어 있다.

(2) Policy Information Base

```
struct PIB_TABLE{
    DESTINATION_ADDR = IP_ADDRESS
    SOURCE_ADDR = IP_ADDRESS

    POLICY(
        DSCP;
        PROTOCOL;
        DESTINATION_PORT_MIN;
        DESTINATION_PORT_MAX;
        SOURCE_PORT_MIN;
        SOURCE_PORT_MAX;
    )
}end PIB_TABLE
```

표 3. 표준 COPS 프로토콜의 PIB의 구조

본 연구에서는 3장 3절 2항의 구현 요소에서 명시한대로 정책을 저장하는 정책 저장소를 별도로 구성하지 않고 PDP노드에 함께 구성하였다. 표 3.은 하나의 정책을 저장하는 구조체이다. 출발지와 목적지 IP 주소, 그리고 DSCP 값과 트래픽의 출발지와 목적지 TCP 포트의 최대값과 최소값 필드로 구성된다.

4. 능동형 PDP 탐색 구현

1) 능동형 PDP 탐색 기법에 대한 고찰

능동형 PDP 탐색 기법을 구현하기에 앞서 3장에서 설명한 능동형 PDP 탐색 기법에 대해 다시 한 번 고찰할 필요가 있다. 아래 그림 33.은 능동형 PDP 탐색 기법에 의한 이동 노드 PEP#4의 PDP 선택과정을 도식한 것이다. 새롭게 정책 전송 영역에 참여하거나 다른 PDP가 관리하는 영역에서 이동한 PEP #4는 광고메시지를 이용하여 주변의 PDP나 PEP에게 연결 가능한 PDP정보를 수신하게 된다.

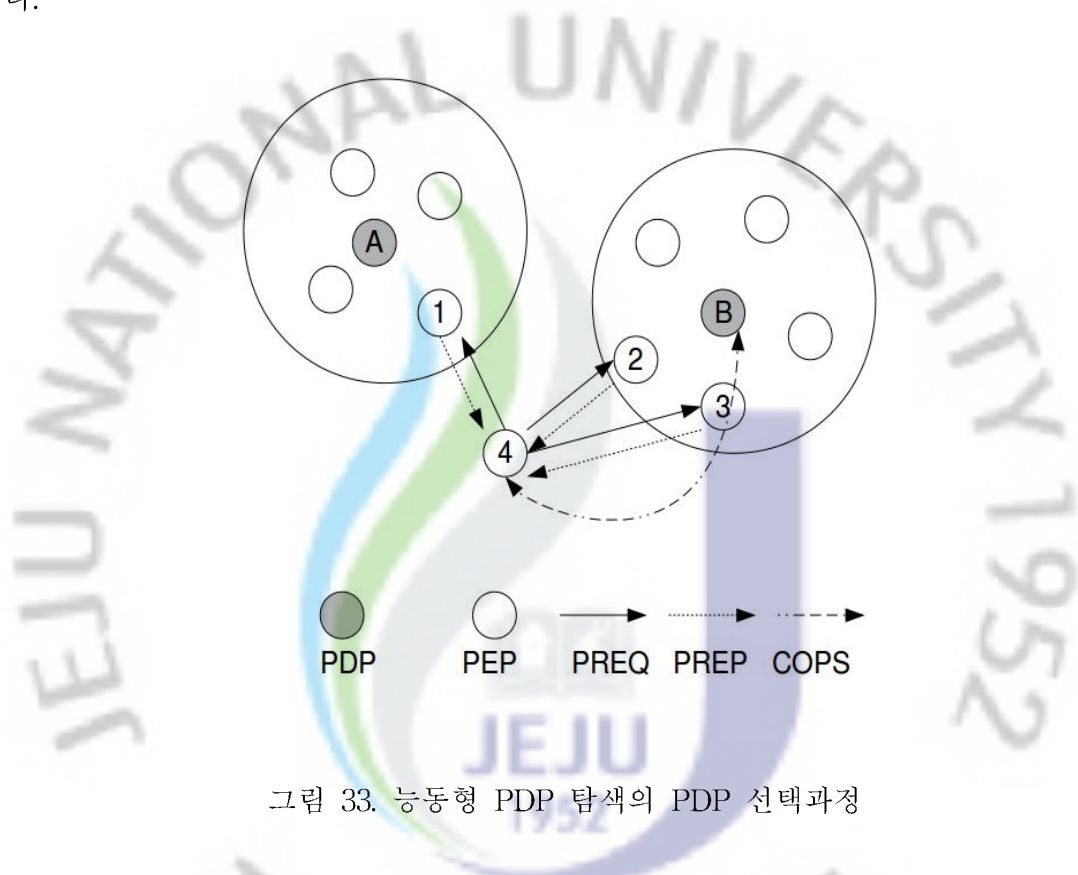


그림 33. 능동형 PDP 탐색의 PDP 선택과정

이동 PEP #4는 그림 33.에서 보이는 다른 PEP노드 #1, #2, #3으로부터 PDP A와 B의 정보, 즉 홉 수를 받게 된다. 하지만, 그 정보는 해당 정보를 전송한 PEP #1, #2, #3에서 연결한 PDP까지의 홉 수이며, 새롭게 참여한 이동 노드인 PEP #4는 자기 자신으로부터 PDP까지의 홉 수를 알 수 없다. 그러므로 제안된 메커니즘에서는 광고메시지의 범위를 1 홉으로 제한하여 PREQ를 보낸다. 이로써 PEP #4는 다른 PEP들로부터 받은 PDP 홉에 +1을 함으로써 자기 자신으로부터 PDP까지의 홉 수를 알아낼 수 있다.

Table: Links

Local IP	remote IP	Hysteresis	Linkcost
192.168.0.101	192.168.0.102	0.00 0.933/1.000	1.071
192.168.0.101	192.168.0.103	0.00 0.020/0.000	INFINITE

Table: Neighbors

IP address	SYM	MPR	MPRS	Willingness	2 Hop Neighbors
192.168.0.102	YES	NO	YES	3 0	
192.168.0.103	NO	NO	NO	3 1	

Table: Topology

Destination IP	Last hop IP	Linkcost
192.168.0.102	192.168.0.101	0.933/1.000 1.071
192.168.0.101	192.168.0.102	1.000/1.000 1.000
192.168.0.103	192.168.0.102	0.569/0.475 3.706
192.168.0.102	192.168.0.103	0.478/0.627 INFINITE

Table: HNA

Network Netmask Gateway

Table: MID

IP Aliases

Table: Routes

Destination	Gateway	Metric	ETX	Interface
192.168.0.102/32	192.168.0.102	1	1.071	wlan0
192.168.0.103/32	192.168.0.102	2	4.778	wlan0

그림 34. OLSR 라우팅에서 추출한 라우팅 정보

그림 34.은 실험에 사용한 OLSR MANET 라우팅 프로그램에서 추출한 노드 간 라우팅 정보이다. 192.168.0.101에서 192.168.0.102를 통하여 192.168.0.103으로 연결되어 있음을 보여준다. 즉, MANET라우팅은 2홉 토폴로지가 완성되면 중간 노드는 다른 노드의 게이트웨이로 라우팅 테이블에 등록된다. 그러므로 연결되는 즉시 자신으로부터 연결된 노드들의 홉 수를 알 수 있다는 뜻이다. MANET 라우팅에 의한 네트워크 토폴로지의 구성은 자율적으로 결정되므로 노드의 파워에 의한 전파 전송 범위안의 모든 노드는 자동으로 네트워크에 가입된다. 그러므로 이동 노드 PEP #4는 이미 해당 무선 네트워크에 노드로써 동작하고 있는 것이 된다. 이동 노드 PEP #4는 스스로 1 홉에 있는 이웃 노드들에게 유니캐스트로 PDP 정보를 요청하면 되고, 이 경우 광고메시지는 전혀 발생하지 않게 된다. 이는 결과적으로 정책 전송 영역을 결정하는 과정에서 광고메시지를

전혀 이용하지 않아 이에 대한 네트워크 부하 문제를 해결할 수 있다.

두 번째는 FOP 메시지에 대한 것이다. PDP는 자신이 정책을 전송하고 있는 PEP 노드들을 MNL을 이용하여 관리하고 있다. 이 MNL에서 자신으로부터 PEP 노드까지의 홉이 증가하는 PEP 노드에게 보내는 메시지로 PDP가 자신이 관리하고 있는 노드의 이동을 감지하고 이동한 PEP 노드에게 새로운 PDP를 탐색하라는 메시지이다.

FOP 메시지의 필요성에 대한 의문은 PDP에서 PEP의 이동을 감지하지 않더라도 PEP에서 스스로 PDP와 떨어지고 있다는 것을 알고 있다는 것이다. 이는 확장된 KA 메시지의 교환에 따라 PEP가 인지할 수 있다. 능동형 PDP 탐색 기법은 PDP가 아닌 PEP가 능동적으로 연결 가능한 PDP를 탐색한다. 새롭게 네트워크에 참여한 노드의 광고메시지에 대한 문제점을 제시한 것과 마찬가지로 PEP는 능동적으로 기 연결된 PDP와 홉이 늘어나는 것을 감지하고 새로운 PDP를 찾게 하면 FOP는 불필요하게 된다.

세 번째는 CC 메시지에 대한 필요성이다. 새로운 PDP의 탐색을 하는 PEP는 기존의 PDP와 연결을 지속할지 새로운 PDP와 연결을 할지를 결정하게 된다. 이때, 새로운 PDP와 연결을 하게 되면 기존의 PDP에 CC 메시지를 보내게 되고, 이를 수신한 PDP는 자신의 MNL에서 해당 노드들 삭제한다. 하지만, 새로운 PDP에 연결하기 위해서는 기존 연결된 PDP에 대한 연결을 끊게 된다. 이 때, 표준 COPS 프로토콜에 의해서 Client Close 메시지가 전송되어 실제 COPS 연결을 종료하고 PDP는 연결 대기 상태가 된다. 이것은 더 이상 확장한 KA 메시지가 전송되지 않음을 의미하며, KA 메시지에 의해서 MNL을 관리하는 PDP에서 자동으로 해당 PEP 정보는 삭제되게 된다. 이로써 PREQ에 의한 CC 메시지와 이를 수신한 PDP의 MNL에서 해당 PEP노드를 삭제하는 절차는 필요가 없다.

수정한 능동형 PDP 탐색 기법을 사용하는 MANET에서 정책 기반 네트워크 관리 실험환경의 구축에는 다음과 같은 요소로 정리할 수 있다.

- COPS-PR의 구현과 KA 메시지의 확장

3절에서 설명한 표준 COPS 프로토콜에서 KA 메시지를 능동형 PDP 탐색 프로토콜에 맞게 확장한다. 이에 대해 본 절의 2항에서 자세히 설명한다.

- MNL의 작성과 관리

확장한 KA 메시지에 의한 PDP의 MNL의 작성과 관리가 필요하며 본 절의 3항에 기술한다.

- PREQ, PREP 에이전트

이동 노드가 새로운 PDP를 탐색하기 위해 주변 노드들에게 전송할 PREQ메시지와 이에 응답할 PREP메시지이다. 이에 대해서는 본 절의 4항에서 설명한다.

- QoS 모델의 적용과 처리 프로그램의 구현

정책 기반 네트워크 관리는 QoS 모델의 발전에 따라 제안되었다. 이에 대해서는 2장 관련연구와 3장 시스템 접근 방법에서 결정된 DSMARK 모델을 사용하고 이를 PREQ, PREP 에이전트에 의해서 수신된 정책이 해당 QoS의 트래픽 셰이핑 모델을 따라 전송될 수 있도록 하는 프로그램이 필요하다. 이에 대한 사항은 본 장의 5절에서 상세히 다룬다.

- 정책 관리 도구의 구현

PMT의 구현에 있어서는 기존 C언어의 구조체로 되어 있는 PIB 형태를 확장하여 각 정책의 파라미터를 입력받아 저장하는 구조가 필요하다. 이에 대한 구현은 본 절의 4항과 본 장의 6절에 설명한다.

2) KA(keep-Alive) 메시지의 확장

COPS 프로토콜을 정의에 의하면, KA 메시지는 PEP에 의해 Common Header만 보내게 되어 있다. PDP는 COPS 프로토콜에서 호출되고, KA에 대해 응답 KA를 보내는 구조로 되어 있다.

KA 메시지의 확장은 COPS 프로토콜에서 PDP에 함수를 추가하여 기존 KA 함수에서 호출하도록 하여 구현하였다. 추가한 함수는 MNL의 내용을 호출하는 형태로 되어 있다.

PEP는 응답 KA를 받으면, common_header 뒤에 MNL을 오브젝트 파일로 저장한다. 표준 KA 메시지는 단순한 COPS 연결을 유지하는 용도로 사용되며, 제안된 능동형 PDP 탐색 Protocol에서는 이를 확장하여 연결된 PDP와 PEP 사이의 정보를 교환하게 되어 있다. 그림 35.은 표준 KA 메시지가 전송되는 화면을 보인다.

```
handle_packet: received COPS message: Keep-Alive (KA)
handle_packet: received COPS message: Keep-Alive (KA)
handle_packet: received COPS message: Keep-Alive (KA)
handle_packet: received COPS message: Keep-Alive (KA)
handle_packet: received COPS message: Keep-Alive (KA)
handle_packet: received COPS message: Keep-Alive (KA)
handle_packet: received COPS message: Keep-Alive (KA)
handle_packet: received COPS message: Keep-Alive (KA)
handle_packet: received COPS message: Keep-Alive (KA)
handle_packet: received COPS message: Keep-Alive (KA)
handle_packet: received COPS message: Keep-Alive (KA)
handle_packet: received COPS message: Keep-Alive (KA)
handle_packet: received COPS message: Keep-Alive (KA)
handle_packet: received COPS message: Keep-Alive (KA)
handle_packet: received COPS message: Keep-Alive (KA)
```

그림 35. COPS 프로토콜의 표준 KA(Keep-Alive) 메시지

```

Function FILL_MNL(void){
    FILE = "/SOMEWHERE/MNL"
    If (FILE is NULL) Than
        Display Standard_Error
    Else
        BUFFER = Read(FILE)
    Endif
}end FILL_MNL

Function COPS_SEND_KA_CALLBACK(integer COPS_MSG){
    IF (COPS_MSG == 1) Than
        Call FILL_MNL
    Else
        Display Standard_Error
    Endif
    //표준 COPS-PR 헤더 값
    OBJECT.CONTENTS_LEN = FILL_MNL(integer BUFFER)
    OBJECT.CONTENTS = FILL_MNL(BUFFER)
}end COPS_SEND_KA_CALLBACK

Function HANDLE_KA(integer COPS_MSG){
    Call COPS_SEND_KA_CALLBACK
    Pass Out COPS_MSG + COPS_SEND_KA_CALLBACK
}end HANDLE_KA

```

그림 36. KA 메시지 확장을 위한 CALLBACK 함수 구현

PDP에서 표준 COPS 프로토콜에서 KA 메시지 전송 함수에 두 개의 함수를 추가하여 PEP 관리 리스트인 MNL을 덧붙여 보내게 하였다. 그림 36.는 추가한 두개의 함수이다. COPS_SEND_KA_CALLBACK은 KA 메시지 전송 함수 내에 추가되는 함수이고, 이 함수는 FILL_MNL 함수를 다시 호출한다. FILL_MNL함

수는 실제의 MNL을 버퍼에 저장하고 있는 함수이다.

```
Function HANDLE_KA(interger COPS_MSG){  
    WRITE_MNL_FILE(char CONTENTS)  
    PRINT_MNL_FILE(char CONTENTS)  
}end HANDLE_KA  
  
Function WRITE_MNL_FILE(char CONTENTS){  
    FILE = "/SOMEWHERE/PDPINFO"  
    Write(FILE) = CONTENTS  
}end WRITE_MNL_FILE  
  
Function PRINT_MNL_FILE(char CONTENTS){  
    Display(CONTENTS)  
}end PRINT_MNL_FILE
```

그림 37. PEP에서 확장한 KA 메시지의 처리

그림 37은 PEP쪽에서 확장된 KA 메시지를 수신하고, 화면과 파일로 출력하는 과정이다. 이 과정에서 저장된 PDP 정보는 PREP 메시지를 전송할 때 자료 파일로 쓰인다.

```

0 | 192.168.0.101 | 5 | 192.168.0.115 |
7 | 192.168.0.101 | 6 | 192.168.0.116 |
-----
handle_packet: received COPS message: Keep-Alive (KA)
-----
Priority | PDP Addr. | PDP Hop | Neighbor Addr. |
-----
1 | 192.168.0.101 | 1 | 192.168.0.111 |
2 | 192.168.0.101 | 1 | 192.168.0.100 |
3 | 192.168.0.101 | 2 | 192.168.0.112 |
4 | 192.168.0.101 | 3 | 192.168.0.113 |
5 | 192.168.0.101 | 4 | 192.168.100.221 |
6 | 192.168.0.101 | 5 | 192.168.0.115 |
7 | 192.168.0.101 | 6 | 192.168.0.116 |
-----
handle_packet: received COPS message: Keep-Alive (KA)
-----
Priority | PDP Addr. | PDP Hop | Neighbor Addr. |
-----
1 | 192.168.0.101 | 1 | 192.168.0.111 |
2 | 192.168.0.101 | 1 | 192.168.0.100 |
3 | 192.168.0.101 | 2 | 192.168.0.112 |
4 | 192.168.0.101 | 3 | 192.168.0.113 |
5 | 192.168.0.101 | 4 | 192.168.100.221 |
6 | 192.168.0.101 | 5 | 192.168.0.115 |
7 | 192.168.0.101 | 6 | 192.168.0.116 |

```

그림 38. 확장한 KA(Keep-Alive) 메시지의 화면 출력

그림 38은 앞서 설명한 표준 COPS 메시지를 PDP와 PEP 쪽에서 확장하여 전송된 KA 메시지이다. 전송된 정보는 PDP Hop 필드를 기준으로 정렬하고 Priority 값을 주어 저장하도록 하였다. 이 정보는 다른 PEP의 PREQ에 응답하는 PREP의 PDP정보로 사용된다.

3) PREQ, PREP 에이전트

새로운 PDP를 탐색하기 위한 PREQ와 이에 응답하는 PREP는 에이전트로 구성하였다. PREQ는 필요할 때마다 실행하며, COPS 연결을 하고 있는 노드들은 PREP를 실행하여 특정 TCP Port에서 응답 대기하도록 작성하였다.

Packet-Type	Source Address	Seq_Num	TTL
-------------	----------------	---------	-----

그림 39. PREQ의 헤더 구조

Packet-Type	PDP-Information	Seq_Num	PDP Address	PDP hop
-------------	-----------------	---------	-------------	---------

그림 40. PREP의 헤더 구조

그림 39와 그림 40.는 능동형 PDP 탐색 Protocol에서 제안하는 PREQ와 PREP 헤더의 구조이다. 그림 39의 PREQ 헤더에서 Packet-Type이 1이면 PREQ 메시지임을 나타내며, Source Address는 PREQ 메시지를 보내는 PEP 노드 자신의 주소를 표시하며, Seq_Num은 패킷의 중복 수신을 막기 위해 사용한다. 그리고 TTL은 PEP의 광고 메시지의 도달 범위를 홉 수로 제한하기 위해 사용된다.

그림 40의 PREP 메시지는 PREQ 메시지에 대한 응답 메시지로 PDP가 수신했을 경우는 자신의 정보를 전송하고, PEP노드의 경우는 해당 PEP가 COPS 연결된 PDP 정보를 유니캐스트로 송신한다. Packet-Type의 값이 2이면 PREP 메시지이며, PDP-Information은 세 가지 경우의 수를 가지고 있는데 첫 번째로 자기 자신이 PDP이면 0을 세팅하고 PDP address 필드에는 자신의 주소를 표시한다. PEP이면 해당 노드로부터 연결된 PDP까지의 홉 수를 표시하고, 두 번째 경우는 수신한 노드가 PEP인데 연결된 PDP가 없는 경우에는 NULL 값을 표시하고 PDP-Information 필드에는 0을 세팅한다. 이 표시는 PDP에 대한 정보가 없는 것이므로 PREQ를 보낸 PEP는 새로운 PDP를 재탐색하게 된다. 세 번째 경우는 PREP를 보내는 노드가 PDP인 경우이다. 이 경우는 PDP-Information 필드를 1로 세팅하고, PDP address 필드의 값은 자신의 주소를 입력하고, PDP Hop 필드의 값은 0이다.

능동형 PDP 탐색에 의하면 PREQ의 TTL 필드에 홉을 기준으로 네트워크 전송 한계를 지정하였으나 구현한 실험 환경에서는 광고메시지를 전송하지 않고 유니캐스트로 목적지에 PREQ를 보내게 되므로 이 값은 홉 수에서 시간으로 바뀌었다. 기본값을 100ms으로 하여 해당하는 시간 내에 정해진 노드에 PREQ를 보내지 못하면 폐기된다.

PREP_HEADER 구조체는 PACKET_TYPE, PDP_INFO, SEQ_NUM, PDP_ADDR, HOP의 멤버를 가진다. 그리고 PACKET_TYPE PREQ 이면 1을 정의하고, PACKET_TYPE PREP이면 2를 정의하였다. PREP가 전송되는 노드가 PDP인지 PEP인지에 따라 PREP 헤더의 값이 달라지므로 PEP_REPLY 0과 PDP_REPLY 1을 선언하였다. PREP는 PACKET_TYPE이 2이면 PREP 패킷임을 의미하고, 나머지는 세 가지 경우로 나누어 생각해 볼 수 있다. PREP를 보내는 노드가 PDP인 경우와 PEP인 경우, 그리고 PEP이지만 PDP에 대한 접속 정보가 없는 경우이다.

PREP를 보내는 노드가 PEP이고 PDP에 접속되어 있다면, PDP_INFO에는 0을 세팅하고, PDP_ADDR에는 접속 중인 PDP 정보를 입력하고, HOP에는 해당 PDP까지의 홉을 명시한다.

PREP를 보내는 노드가 PDP라면 PDP_INFO 필드는 1로 세팅하고, PDP_ADDR에는 응답하고 있는 자신의 주소를 입력하고, HOP에는 0을 세팅한다.

PREP를 보내는 노드가 PEP이고 PDP에 연결되어 있지 않다면, PREP 메시지에는 0과 NULL을 세팅한다. 0은 PDP_INFO 필드에 설정하고, NULL은 PDP_INFO와 HOP에 설정한다. 이 경우 PREQ에 대한 응답으로 위의 값을 가진 PREP를 만나면 능동형 PDP 탐색 기법에서는 새로운 PDP를 탐색하기 위해 광고메시지를 보내 재탐색을 하게 되지만, 구현한 실험 환경에서는 해당 노드로 더 이상 PREQ 메시지를 보내지 않는다.

PREP를 보내는 노드가 PDP인 경우 자신으로부터 요청한 노드까지 라우팅 정보를 이용하여 PREQ를 요청한 PEP 노드까지 홉 정보를 제공하면 하고, PEP가 PREQ를 전송할 때는 KA 메시지에 의해서 교환된 PDP와 자신의 정보를 이용하여 전송한다. 만약 요청 노드까지의 정보를 찾지 못하면, 제안된 메커니즘의 방법대로 PDP에 연결되지 않은 PEP노드를 세팅하는 방법으로 헤더를 작성하여 전송한다.

PREP를 수신한 PEP 노드는 여러 노드에서 PDP 정보를 수신할 수 있다. 그러므로 PREP를 수신한 PEP 노드는 자신이 수신한 PDP 정보에서 최적의 PDP를 찾아 내야 한다.


```

gunsroses@ncl:~/PREQ_PREP_2$ ./PREQ_agent
Usage: ./PREQ_agent <Server IP:Port> <Server IP:Port>...]]
Example: ./PREQ_agent 192.168.0.1:5001 192.168.0.2:5001
gunsroses@ncl:~/PREQ_PREP_2$
gunsroses@ncl:~/PREQ_PREP_2$
gunsroses@ncl:~/PREQ_PREP_2$ ./PREQ_agent 192.168.100.221:5001 192.168.100.221:5002
-----
press Enter to find PDP

connected to PREP agent 192.168.100.221:5001
PREP agent = 192.168.100.221:5001 --> PDP Info : addr = 192.168.0.101, hop = 4, is_pdp = 0
connected to PREP agent 192.168.100.221:5002
PREP agent = 192.168.100.221:5002 --> PDP Info : addr = 192.168.0.102, hop = 1, is_pdp = 0
Found the best PDP 192.168.0.102 and save it to file '/home/policy/best_pdp.txt'
-----
press Enter to find PDP
█

```

그림 41. PREQ_agent

그림 41.은 PREQ 에이전트의 실행화면이다. 명령행에서 PREQ를 보낼 IP와 TCP 포트를 지정하여 실행하고, 접속한 PREP 에이전트로부터 PDP 정보를 전송 받아 저장한다.

```

gunsroses@ncl:~/PREQ_PREP_2$
gunsroses@ncl:~/PREQ_PREP_2$
gunsroses@ncl:~/PREQ_PREP_2$ ./PREP_agent
Jsage: ./PREP_agent <port> <MNList file path> <mode>
mode : 1 - PDP, 0 - otherwise
gunsroses@ncl:~/PREQ_PREP_2$
gunsroses@ncl:~/PREQ_PREP_2$
gunsroses@ncl:~/PREQ_PREP_2$ ./PREP_agent 5001 /home/policy/MNList.out 0
█

```

그림 42. PREP_agent

그림 42.은 PREP 에이전트의 실행화면이다. PREP 에이전트와 마찬가지로 명령행 인자를 가지고 실행하게 하였으며, 실행 TCP 포트와 제공한 PDP 정보 그리고 마지막 숫자는 자신이 PDP이면 0으로 실행하고, PEP이면 1로 설정하여 실행한다.

```

~~~~~
type   : 1
srcaddr : 117.17.102.208
seq_num : 1
ttl    : 10
~~~~~
Found the nearest PDP [192.168.0.101] to PREQ agent[192.168.100.221]. hop = 4
Found the nearest PDP [192.168.0.101] to me. hop = 4
Found the nearest PDP [192.168.0.101] with hop [4]
PDP information : addr = 192.168.0.101, hop = 4

```

그림 43. PREP의 화면 출력

그림 43은 PREQ를 받은 PREP의 화면이다. 자신이 연결된 PDP와 자기 자신으로부터 요청한 PEP까지의 정보를 바탕으로 PDP 정보를 송신한다.

```

gunsroses@ncl:~/PREQ_PREP_2$ ./PREP_agent 5002 /home/policy/MNList.out.1 0
client connected : 192.168.100.221
PREQ Header
~~~~~
type   : 1
srcaddr : 117.17.102.208
seq_num : 1
ttl    : 10
~~~~~
Found the nearest PDP [192.168.0.102] to PREQ agent[192.168.100.221]. hop = 1
Found the nearest PDP [192.168.0.102] to me. hop = 1
Found the nearest PDP [192.168.0.102] with hop [1]
PDP information : addr = 192.168.0.102, hop = 1

```

그림 44. PREP의 화면 출력 #2

그림 44는 PREQ를 두 대의 PEP에 전송했다고 가정했을 때, 두 번째 PREP 에이전트에 의한 소스이며, 첫 번째와 같은 동작을 한다.

4) Management Node List의 작성

능동형 PDP 탐색 기법에서는 MNL을 만들어 PDP가 정책 전송을 하는 PEP들을 관리하게 된다. 확장된 KA메시지에 의해 전송된 PDP로부터 PEP까지의 홉 정보를 이용하여 MNL을 생성하였다.

```

Table: Links
Local IP    remote IP    Hysteresis  Linkcost
192.168.0.101  192.168.0.102  0.00      1.000/1.000 1.000

Table: Neighbors
IP address  SYM MPR MPRS    Willingness 2 Hop Neighbors
192.168.0.112  YES YES YES 7 2
192.168.0.111  YES YES YES 7 1
192.168.0.113  YES YES YES 7 3
192.168.0.115  YES YES YES 7 5
192.168.0.114  YES YES YES 7 4
192.168.0.116  YES YES YES 7 6
192.168.0.100  YES YES YES 7 1

Table: Topology
Destination IP  Last hop IP Linkcost
192.168.0.102  192.168.0.101  1.000/1.000 1.000
192.168.0.101  192.168.0.102  1.000/1.000 1.000

Table: HNA
Network Netmask Gateway

Table: MID
IP Aliases

Table: Routes
Destination Gateway Metric  ETX Interface
192.168.0.102/32  192.168.0.102  1  1.000  wlan0

```

그림 45. 라우팅에 의한 라우팅 정보

그림 45.은 초기의 토폴로지 정보를 알기위한 OLSR 라우팅에 의해 도출한 정보이다. 이 정보를 가공하여 각 노드에서 PREQ 전송과 PREP로 응답한 정보로 이용한다.

```

gunsnroses@ncl:~/mk_mnlist$ ./mk_mnlist
Usage : ./mk_mnlist managed-node-file manage-node-list
gunsnroses@ncl:~/mk_mnlist$ ./mk_mnlist MNL
MNL.txt      MNLlist.out  MNLlist.out.2
gunsnroses@ncl:~/mk_mnlist$ ./mk_mnlist MNL.txt MNLlist_convert.txt
pdp_addr : 192.168.0.101
192.168.0.111 1
192.168.0.100 1
192.168.0.112 2
192.168.0.113 3
192.168.0.114 4
192.168.0.115 5
192.168.0.116 6
gunsnroses@ncl:~/mk_mnlist$ cat MNLlist_convert.txt
-----
Priority      |      PDP Addr.      |      PDP Hop |      Neighbor Addr.  |
-----
1            |      192.168.0.101  |      1        |      192.168.0.111   |
-----
2            |      192.168.0.101  |      1        |      192.168.0.100   |
-----
3            |      192.168.0.101  |      2        |      192.168.0.112   |
-----
4            |      192.168.0.101  |      3        |      192.168.0.113   |
-----
5            |      192.168.0.101  |      4        |      192.168.0.114   |
-----
6            |      192.168.0.101  |      5        |      192.168.0.115   |
-----
7            |      192.168.0.101  |      6        |      192.168.0.116   |
-----
gunsnroses@ncl:~/mk_mnlist$ █

```

그림 46. 변환된 노드 리스트

이 정보는 그림 46.과 같이 정리되고 초기에 PDP에 접속할 정보를 제공하고 PREQ 요청에 대한 PREP 응답 자료로 쓰인다. COPS 연결이 완성된 후에 KA 메시지에 의한 MNL 또한 유지하게 하였다.

5. QoS 모델의 적용

1) 개요

2장의 관련연구와 3장의 시스템 접근에서 VANET에 적합한 QoS 모델로 DSMARK를 결정하였다. 이는 일반적인 트래픽을 처리할 때는 트래픽의 특성에 맞는 큐잉을 적용할 수 있고, 위급한 메시지가 발생했을 경우 다른 트래픽을 드롭 시키고 신속히 대역폭을 확보할 수 있는 QoS 모델이 VANET에 필요하기 때문이다. DSMARK는 AF PHB의 분류를 통하여 비디오 스트리밍, VoIP, 통상적인 인터넷 트래픽을 처리할 수 있으며, EF PHB에 의하여 전체 대역폭의 30%를 즉시 확보할 수 있기 때문에 VANET에 가장 적합하다.

본 절에서는 리눅스 운영체제에서 DSMARK를 사용 가능하게 하고, 관련 명령어를 이용하여 수신된 정책에 의해 트래픽이 실제로 처리하게 하는 과정을 설명한다.

2) DSMARK의 적용

실험 환경에서 QoS 모델의 실제 적용을 위하여 필요한 사항은 다음과 같다.

- 리눅스 커널의 설정
- DSMARK에 의한 클래스의 분류를 처리하는 프로그램
- 수신된 정책을 분석하여 tc(traffic control) 명령어에 의한 처리하는 프로그램

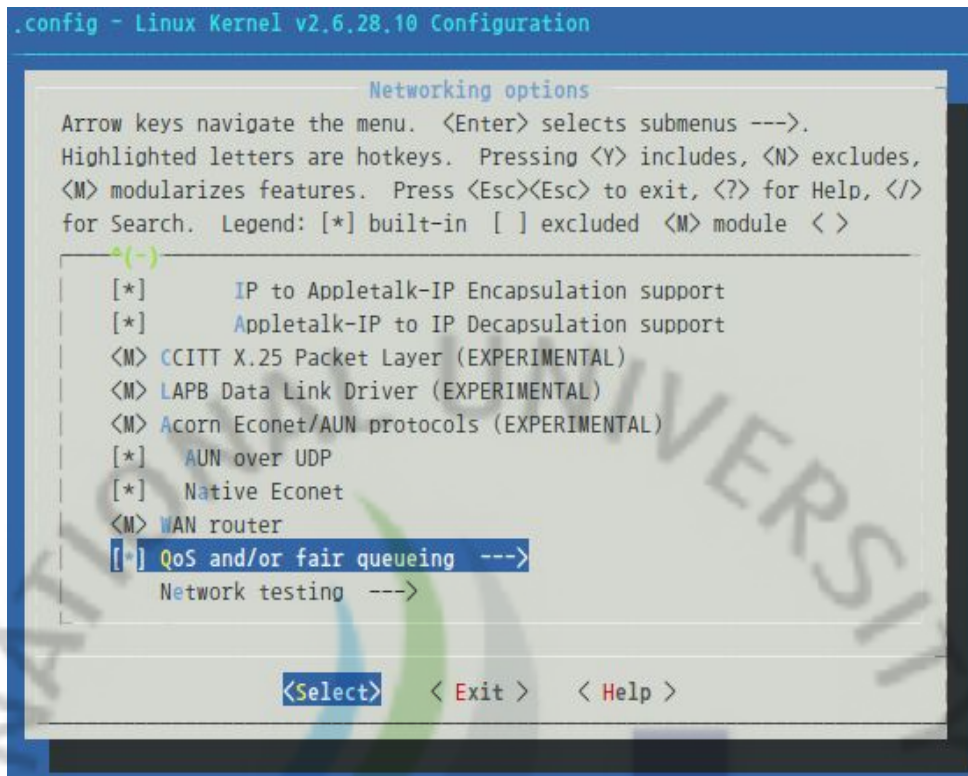


그림 47. 리눅스 커널에서 QoS를 설정

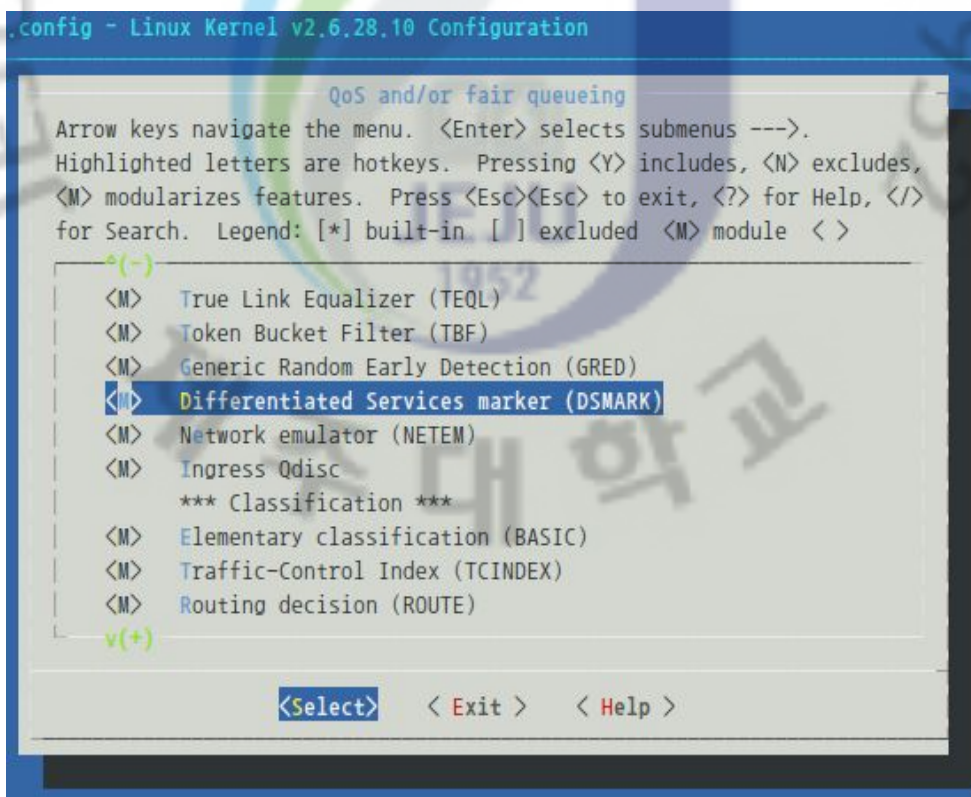


그림 48. 리눅스 커널의 DSMARK 모듈 설정

그림 47.와 그림 48.는 리눅스 커널의 QoS 큐잉 모듈을 사용가능하게 하는 그림이다. 생성된 QoS 모듈을 컴파일 하여 TCP 헤더의 DSCP 필드 마킹이 가능하게 한다.

```
tc qdisc del dev wlan1 root
tc qdisc add dev wlan1 handle 1:0 root dsmark indices 8
tc class change dev wlan1 classid 1:1 dsmark mask 0x0 value 0xb8
tc class change dev wlan1 classid 1:2 dsmark mask 0x3 value 0x58
tc class change dev wlan1 classid 1:3 dsmark mask 0xe3 value 0x10
tc class change dev wlan1 classid 1:4 dsmark mask 0x1f value 0x60
tc class change dev wlan1 classid 1:5 dsmark mask 0x0 value 0x30
tc class change dev wlan1 classid 1:6 dsmark mask 0x3 value 0x70
tc class change dev wlan1 classid 1:7 dsmark mask 0x0 value 0x0
tc filter add dev wlan1 parent 1:0 protocol ip prio 1 u32 match ip src 192.168.0
.101/24 flowid 1:1
```

그림 49. traffic control DSCP

그림 49.는 DSCP 클래스를 tc 명령어로 분류한 화면이다. 클래스 1:0은 EF PHB이며, 나머지는 AF PHB에 대한 클래스 정의이다. 마지막으로 tc filter 명령어에 의해 특정 IP로부터 들어오는 패킷에 대하여 EF 마킹을 하고 QoS 지원 모듈을 포함한 운영체제 커널을 통해 실제 트래픽 처리를 하게 된다.

3) 트래픽 처리 인터페이스의 개발

실제 트래픽 처리를 위한 프로그램은 본 장의 6절에서 설명할 정책 관리 도구에 의해서 웹 인터페이스로 지정할 수 있게 하였다. 저장된 정책은 PDP에 의해서 전송되며, PEP에서 정책에 따라 어떤 트래픽 분류에 넣을 것인지를 결정하고 처리하는 순서로 구현하였다. 실험을 위한 간단한 셸 스크립트를 작성하여 사용하였다.

6. 정책 관리 도구

1) PIB의 확장

정책을 정의하고 수정하기 위하여 기존의 PIB를 확장하여 웹 인터페이스를 통하여 정책을 정의할 수 있도록 하였다.

PIB의 원형은 C 언어의 구조체 형태로 되어 있으며, 해당하는 구조체를 확장하여 인자를 받아 전송되도록 하였다.

```
Struct PIB_TABLE{  
    POLICY(  
        char DESTINATION_ADDR  
        char SOURCE_ADDR  
        char DSCP;  
        char PROTOCOL;  
        integer DESTINATION_PORT_MIN;  
        integer DESTINATION_PORT_MAX;  
        integer SOURCE_PORT_MIN;  
        integer SOURCE_PORT_MAX;  
    )  
}end PIB_TABLE
```

표 4. PIB 테이블의 구조

```

Function POLICY_DECISION(struct PIB_FILE){
    PIB_FILE = "/SOMEWHERE/PIB_FILE"

    If PIB_FILE is NULL then
        Display Standard_Error
    Else
        Read(PIB_FILE)
    Endif
} end POLICY_DECISION

```

그림 50. 외부 입력 파일에서 파라미터를 적용하게 한 확장된 PIB

표 4.에서 확장한 PIB 테이블의 구조를 보이며, PIB의 정책 요소들을 모두 변수로 지정하고, 그림 50.에서 확장한 PIB 구조에 외부 파일을 읽어 정책 파라미터를 지정하는 부분이다. 이제 PEP의 요청에 PDP는 해당하는 정책 파일을 읽어 정책을 생성하여 보내게 된다. 정책 생성에 관한 정책 관리 도구로 웹 인터페이스를 만들어 사용하였다.

2) 정책 관리 도구의 구현

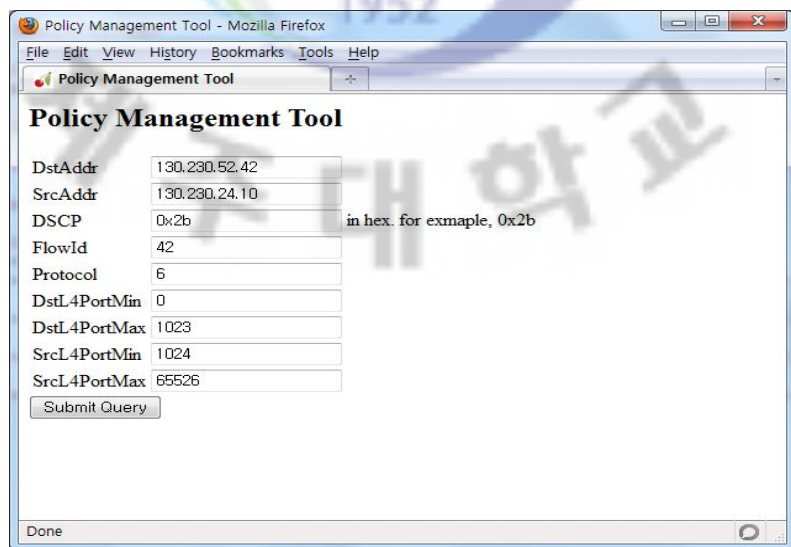


그림 51. 정책 관리 도구의 웹 인터페이스

그림 51.는 구현된 웹 인터페이스이다. 변경된 PIB 구조의 값들을 입력 받으면 이는 특정 파일로 저장하게 되고, PDP는 해당 값을 PEP에게 전송한다.

```
msg: Keep-Alive (KA)
msg: Request (REQ)
FrwkDeviceIdEntry.8: (1.2.1.3.1.8)
  frwkDeviceIdPrid=8
  frwkDeviceIdDescr=Linux router romukoppa
  frwkDeviceIdMaxMsg=2048
  frwkDeviceIdMaxContexts=250

FrwkPrcSupportEntry.7: (1.2.1.1.1.7)
  frwkPrcSupportPrid=7
  frwkPrcSupportSupportedPrc=frwkPrcSupportEntry (1.2.1.1.1)
  frwkPrcSupportSupportedAttrs=0x11, 0x22, 0x33, 0x44

fscanf ret = 9
create_t1 : 192.168.0.103 192.168.0.101 2b 42 6 0 1023 1024 65536
fscanf ret = 9
create_t1 : 192.168.0.103 192.168.0.101 2b 42 6 0 1023 1024 65536
msg: Keep-Alive (KA)
msg: Keep-Alive (KA)
```

그림 52. PDP노드의 확장된 PIB와 PMT에 의해 정책 전송 화면

그림 52.은 PDP 노드에서 PEP의 요청에 따라 확장된 정책을 전송하는 것을 보이고 있다.

```

send_req: the tables to send:
FrwkPrcSupportEntry.7: (1.2.1.1.1.7)
  frwkPrcSupportPrid=7
  frwkPrcSupportSupportedPrc=frwkPrcSupportEntry (1.2.1.1.1)
  frwkPrcSupportSupportedAttrs=0x11, 0x22, 0x33, 0x44
FrwkDeviceIdEntry.8: (1.2.1.3.1.8)
  frwkDeviceIdPrid=8
  frwkDeviceIdDescr=Linux router romukoppa
  frwkDeviceIdMaxMsg=2048
  frwkDeviceIdMaxContexts=250
handle_packet: received COPS message: Decision (DEC)
handle_dec: COPS-PR Decision with Command Code 1
FrwkIpFilterEntry.5: (1.2.3.2.1.5)
  frwkIpFilterAddrType=1
  frwkIpFilterDstAddr='192.168.0.103'
  frwkIpFilterDstPrefixLength=25
  frwkIpFilterSrcAddr='192.168.0.101'
  frwkIpFilterSrcPrefixLength=24
  frwkIpFilterDscp=43
  frwkIpFilterFlowId=42
  frwkIpFilterProtocol=6
  frwkIpFilterDstL4PortMin=0
  frwkIpFilterDstL4PortMax=1023
  frwkIpFilterSrcL4PortMin=1024
  frwkIpFilterSrcL4PortMax=65536

FrwkIpFilterEntry.5: (1.2.3.2.1.5)
  frwkIpFilterAddrType=1
  frwkIpFilterDstAddr='192.168.0.103'
  frwkIpFilterDstPrefixLength=25
  frwkIpFilterSrcAddr='192.168.0.101'
  frwkIpFilterSrcPrefixLength=24
  frwkIpFilterDscp=43
  frwkIpFilterFlowId=42
  frwkIpFilterProtocol=6
  frwkIpFilterDstL4PortMin=0
  frwkIpFilterDstL4PortMax=1023
  frwkIpFilterSrcL4PortMin=1024
  frwkIpFilterSrcL4PortMax=65536

```

그림 53. PEP에서 정책 수신 화면

그림 53은 PEP에서 수신한 확장된 PIB에 의한 정책 수신 화면이다. 각 필드는 웹 인터페이스에서 지정된 값을 전송한다.

7. 요약

4장에서 하드웨어와 소프트웨어 환경 구축과 COPS-PR의 구현과 이를 확장하여 능동형 PDP 탐색을 구현하고 표준 정책 기반 네트워크 관리의 정책 적용을 위한 PMT와 정책에 의한 실제 트래픽 처리를 위한 QoS 모델을 적용하였다. 이

과정에서 능동형 PDP 탐색 기법의 네트워크 부하를 줄이기 위해 광고메시지에 의한 PDP 정보 요청을 MANET 라우팅을 이용하여 유니캐스트로 전송하도록 구조를 개선하였다. MANET 환경은 멀티 홉 토폴로지를 구성하기 위하여 이미 많은 광고메시지를 이용하고 있다. 이러한 환경에서 광고메시지가 없는 정책 영역 관리 메커니즘은 MANET에 보다 적합할 것이다. 5장에서 이에 관련한 MANET에서의 광고메시지 측정과 노드 이동에 따른 개선된 메커니즘의 동작을 실험한다.



V. 구현 결과와 분석

본 장에서는 구현 결과를 정리하고 측정된 트래픽 데이터를 기반으로 한 그래프와 비디오 스트리밍 테스트 결과에 대하여 설명한다.

1. 능동형 PDP 탐색 기법의 개선

1) PEP에 의한 광고메시지의 해결

능동형 PDP 탐색 기법은 k -hop Cluster의 단점을 보완하기 위해 제안되었지만 여전히 광고메시지의 문제가 남아 있다. 이는 능동형 PDP 탐색 기법이 k -hop Cluster를 파급 효과와 노드 밀집도에 따른 하나의 PDP에 대한 PEP들의 과도한 집중을 막을 수 있다고 하지만, 이 메커니즘에 의하면 노드의 이동성이 높아지면 즉, PEP의 이동이 많아지면 역시 광고메시지가 같이 증대되며, 이로 인해 네트워크에 부하가 증대된다. MANET 라우팅의 특성상 광고메시지를 전송할 수 있다는 것은 이미 해당 네트워크의 노드로 참여했다는 것이다. 그러므로 MANET 라우팅 정보에 의해 1홉에 해당하는 이웃 노드들을 알 수 있기 때문에 광고메시지로 PDP를 탐색할 필요가 없어진다.

구현한 메커니즘은 MANET의 특성을 고려한 것으로 이동 PEP 노드가 주변의 노드를 탐색할 때 MANET 라우팅 정보에 의해 1홉에 해당하는 노드들에게 유니캐스트 메시지로 PDP 정보를 요청한다. 이는 정책 전송을 위한 PDP와 PEP 연결에 광고메시지가 완전히 배제될 수 있다는 것이 된다.

2) Find Other PDP 메시지의 개선 가능성

능동형 PDP 탐색 기법에서 FOP 메시지는 확장된 KA 메시지를 이용하여

PDP는 자신이 관리하고 있는 PEP 노드들 중 이동하는 노드들을 홉 수가 증가하는 것으로 인지하게 되고 이는 PDP로부터 멀어지고 있다는 것이므로 해당 PEP에 새로운 PDP를 탐색하라는 시그널링이다.

하지만, 확장된 KA 메시지는 PDP와 PEP 사이에서 교환하는 메시지이므로 PEP에서도 자신이 PDP로부터 멀어지고 있다는 것을 인지할 수 있다. 만약, 이를 인지한 PEP가 스스로 다른 PDP를 찾게 한다면, FOP는 불필요하게 된다.

추가적으로 FOP 메시지를 대체하여 네트워크의 관리가능성을 높이는 방법으로 PDP로부터 특정 홉만큼 멀어지면 다른 PDP를 탐색하라는 시그널링을 정책에 삽입하는 방법을 생각해 볼 수 있다. 해당 네트워크에 참여한 노드의 수를 대상으로 밀집도를 산출하고 이에 따라 새로운 PDP를 탐색하는 홉 수를 정책에 반영함으로써 정책 전송 영역을 유연하게 관리할 수 있을 것이다.

3) PEP의 Connection Close 메시지

능동형 PDP 탐색 기법의 새로운 PDP 탐색 알고리즘에서 새로운 PDP에 연결하는 PEP는 기존의 연결된 PDP에 CC 메시지를 전송하게 되고, 이를 수신한 PDP는 해당 PEP 정보를 MNL에서 삭제하게 된다. 이는 표준 COPS 프로토콜에 정의된 COPS-CC 메시지로 대체할 수 있다.

새로운 PDP이 더 낮은 홉을 가졌다면 PEP는 기존 PDP 연결을 끊게 된다. 이는 리눅스의 셸 스크립트로 구현하였다.

```
#!/bin/sh
killall pep
cat /home/policy/best_pdp.txt | xargs -t pep
```

그림 54. 새로운 PDP로 연결하는 셸 스크립트

새로운 최적의 PDP에 대한 정보는 그림 54의 best_pdp.txt에 들어 있다. 이 파일을 PREQ, PREP에 의해 수신된 PDP 정보 중 가장 작은 홉 수를 갖는 PDP 정보를 저장한 것이다. 이 새로운 PDP에 연결하기 위해서는 기존의 PEP 프로세스를 종료해야 한다. 그림 54의 세 번째 줄인 'killall pep' 명령은 시스템에서 실

행 중인 프로세스 중에 pep 이름을 가진 프로세스를 모두 종료한다. KA 메시지를 수신하지 못한 PDP는 해당 PEP의 접속을 종료한다.

```
msg: Client-Open (OPN)
msg: Keep-Alive (KA)
cops_read_msg: connection closed
■
```

그림 55. PDP의 PEP 접속 종료 화면

그림 55는 KA 메시지를 수신하지 못한 PDP가 해당 PEP의 연결을 종료하는 화면이다. PDP의 MNL은 확장된 KA 메시지에 의하여 유지되므로 접속이 끊긴 PEP로부터는 더 이상 KA 메시지를 수신할 수 없으므로 MNL에서 해당 PEP 정보는 삭제되게 되므로 능동형 PDP 탐색 기법 구조에서 CC 메시지는 필요가 없다.

4) PREQ 메시지 헤더의 TTL 필드

PREQ 메시지 헤더의 TTL 필드의 용도는 몇 개의 홉만큼 광고메시지를 전송할 것인가에 대한 값을 저장한다. 하지만, 본 절 1항에서 언급한대로 광고메시지를 이용하지 않으므로 해당 필드의 값이 무의미하다.

실제의 구현에서는 시간으로 값을 변경하여 작성하였다. 해당하는 시간 동안 목적지에 도달하지 못하면 폐기되게 된다. 기본값은 10ms으로 설정하였다.

2. 측정 시나리오

구현한 테스트베드를 실험하기 위하여 영역을 두 개로 나눌 필요가 있다. 제안하는 시스템은 하나의 네트워크에서도 동작 가능하지만 실제 네트워크를 대상으로 한 테스트베드의 특성상 많은 노드를 대상으로 할 수 없기 때문에 물리적으로 영역을 나누고 적어도 2 홉이 가능한 상태를 만들어야 한다.

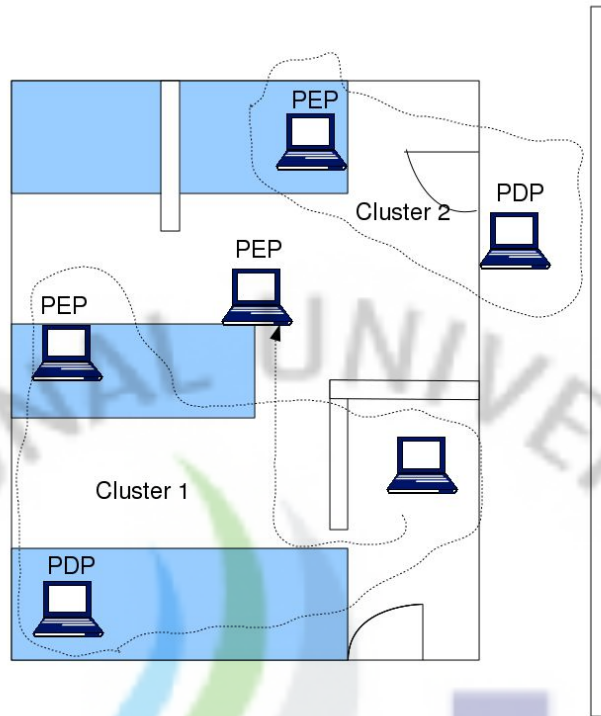


그림 56. 측정 환경

그림 56.은 실험한 연구실 환경의 도식이다. 5대의 랩톱PC로 이루어진 테스트 베드이므로 두 대는 PDP로 설정하였고 나머지는 PEP로 설정하였다. 이동 PEP 노드는 1홉으로 Cluster 1에 연결되어 있는 상태에서 이동하여 그림과 같이 Cluster 2의 방향으로 이동하게 된다. 여기서 이동한 PEP 노드는 PREQ를 보내게 되고 연결 가능한 이웃 노드들에서 PREP에 의한 PDP 연결 정보를 받게 된다. 새로운 Cluster 2의 PDP와 연결한 이동 PEP는 기존 Cluster 1의 연결을 끊고 새로운 Cluster 2의 PDP에 저장된 정책을 전송받게 된다.

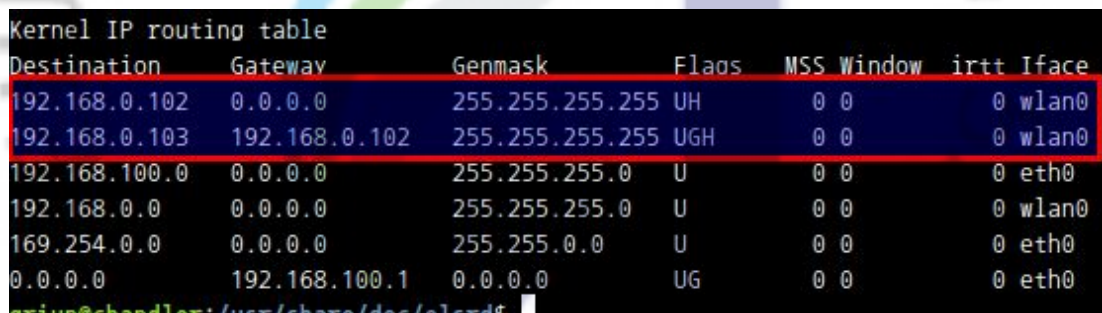
상기한 상황에서 이동 PEP 노드는 특정 서버에서 스트리밍 서비스를 받게 하여 트래픽을 측정하였다. Cluster 1의 PDP에서의 수신한 정책은 비디오 스트리밍을 보내는 서버에 대하여 아무런 정책을 가지고 있지 않으며, Cluster 2의 PDP에는 동일 IP의 트래픽을 EF PHB를 적용하여 전송하도록 정의하였다.

Iperf에 의해서 측정한 노드 간 대역폭은 편차가 크기는 하지만, 액티브 측정방법이므로 1.2Mbits/sec 정도로 측정되었다. 하지만, 실제 파일을 전송할 때의 노드 간의 차가 크기는 하지만, 대역폭은 40kbits/sec~80Kbits/sec정도였다.

3. 결과의 분석

1) MANET 라우팅에서 광고메시지

많은 MANET 라우팅 프로토콜에서 새로운 노드를 발견하고, 연결을 유지하기 위하여 광고메시지를 사용한다. MANET 라우팅의 특성상 모든 노드가 1홉으로 연결이 되었을 때는 각 노드로 직접 라우팅 테이블이 구성되며, 2 홉 토폴로지 발견되면 하나의 노드는 다른 노드의 게이트웨이로 라우팅 테이블에 등록된다. 중간의 중계 노드에서 실제의 트래픽 처리를 위해 IP forwarding 방식으로 해당 하는 노드에 전송하게 된다.



```
Kernel IP routing table
Destination      Gateway         Genmask        Flags   MSS Window  irtt Iface
192.168.0.102    0.0.0.0        255.255.255.255 UH      0 0      0 wlan0
192.168.0.103    192.168.0.102 255.255.255.255 UGH     0 0      0 wlan0
192.168.100.0    0.0.0.0        255.255.255.0  U      0 0      0 eth0
192.168.0.0      0.0.0.0        255.255.255.0  U      0 0      0 wlan0
169.254.0.0      0.0.0.0        255.255.0.0    U      0 0      0 eth0
0.0.0.0          192.168.100.1  0.0.0.0        UG      0 0      0 eth0
```

그림 57. 2 홉 토폴로지에서 라우팅 테이블

그림 57.은 2홉 토폴로지가 구성된 모습이다. 192.168.0.102에는 1홉으로 연결되어 있으며, 192.168.0.103의 경로는 192.168.0.102를 게이트웨이로 통신하게 된다. 이러한 과정들은 모두 광고메시지를 이용하므로 MANET 라우팅에서 광고메시지의 양은 유선 네트워크에 비해 그 비중이 크다고 예상할 수 있다.

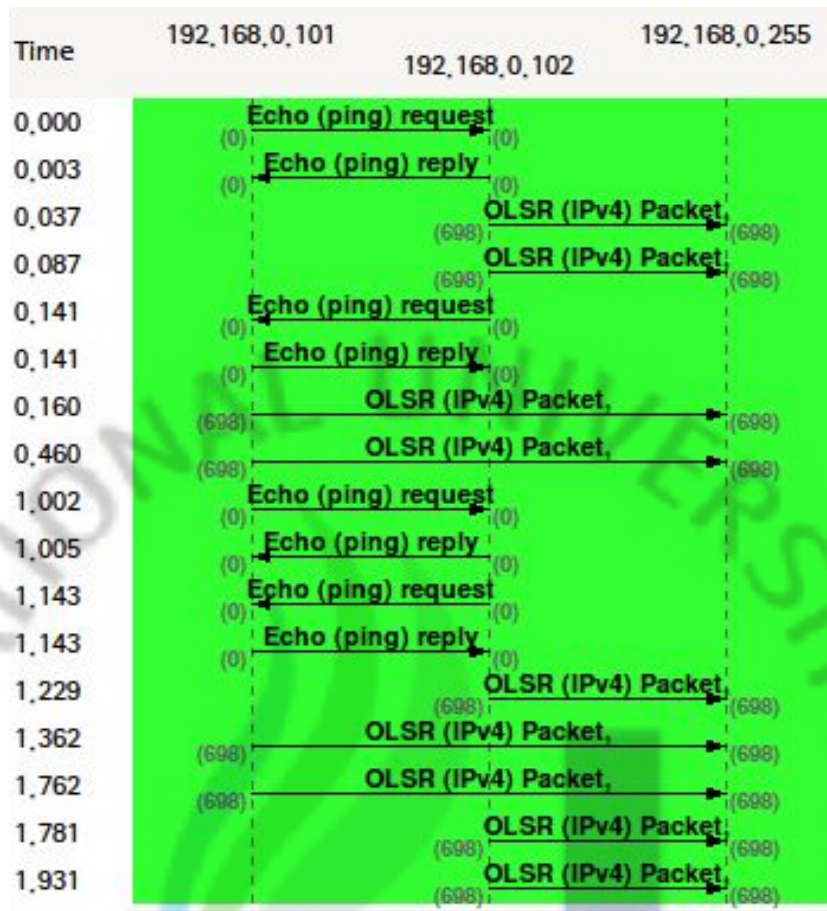


그림 58. 전송 패킷의 분석

그림 58.는 멀티 홉 라우팅 상태에서 전송 패킷을 분석한 것이다. 실험에 사용한 OLSR 라우팅 프로토콜에 의한 트래픽은 네트워크의 각 노드에서 발생하며, 모든 노드로 라우팅 제어 패킷이 전송된다. MANET 라우팅은 일반적인 유선 네트워크의 광고메시지 전송방식을 사용하지 못하는데, 그 이유는 MANET 라우팅이 구성되면서 하나의 노드는 하나씩의 노드를 가진 네트워크 비트 32의 서브넷으로 동작한다. 일반적으로 광고메시지는 서브넷을 통과하지 못하므로 MANET에 광고메시지를 전송하기 위해서는 유선 네트워크와는 다른 방법이 필요하다. 192.168.0.0/24 네트워크의 예를 들면, 해당 서브넷의 광고메시지 전송 주소는 0.0.0.255를 OR 연산한 192.168.0.255가 된다. 하지만, 이 서브넷 전송 주소 방식을 사용하면, MANET의 노드는 하나 하나가 서브넷이므로 다른 노드에 광고메시지를 전송할 수 없게 된다. 이를 해결하기 위해 또 다른 광고메시지 전송 주소 방식인 255.255.255.255를 사용한다. 이 주소는 OR 연산을 수행하면, 0.0.0.0이 되므로 원래의 의미는 인터넷에 연결된 모든 주소로 전송한다는 뜻이 된다. 하지

만, 이 주소는 종종 네트워크 장비에 의해 해당 서브넷으로 광고메시지의 전송 범위를 제한 당한다. MANET 라우팅은 255.255.255.255 주소를 이용한 광고메시지 전송 방식을 제안해 놓고 있다.

	Packets (%)	Bytes	Kbit/s
UDP(OLSRD)	67.53	52515346	5
TCP	26.08	9627161	1
ARP	5.84	1294440	0.1

표 5. 전송 패킷의 분석

표 5.는 전송된 패킷을 계층적으로 분석한 것이다. 구성된 실험 환경에서 주기적으로 노드 상호 간의 TCP 패킷을 전송한 후에 2 홉간의 UDP, TCP, ARP 트래픽을 분석하였다. UDP 패킷은 실험 환경에 이용한 MANET 라우팅 프로토콜인 OLSR의 트래픽이며, TCP는 패킷 발생기에 의해 주기적으로 전송된 트래픽이며, ARP는 유선 네트워크에서와 같은 트래픽이다.

총 패킷의 수가 100%일 때, MANET 라우팅 프로토콜에 의한 UDP 패킷은 65%정도의 비율을 보이고 있다.

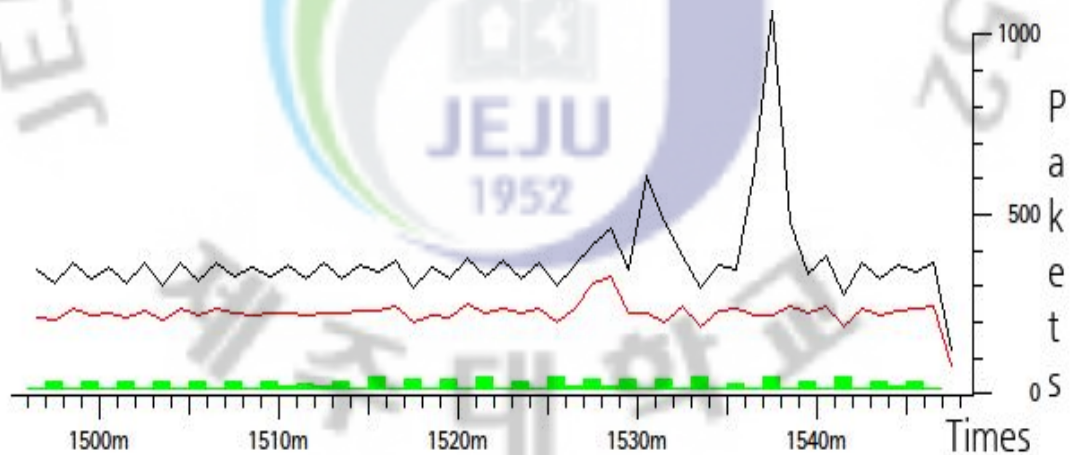


그림 59. MANET 라우팅에 의한 광고메시지와 ARP

그림 59.에서 상단에 있는 선 그래프는 전송된 총 패킷량을 나타내며, 중간에 있는 선 그래프는 MANET 라우팅 프로토콜의 광고메시지이고, 맨 아래 있는 막대 그래프는 ARP 트래픽을 나타낸다. ARP는 유선 네트워크에서도 동일하게 발생하는 트래픽이며, 어떤 주소로 빈번한 통신을 하면 캐시 메모리를 이용하여, ARP

광고메시지의 양을 줄일 수 있다.

하지만, MANET 라우팅에서 보내는 광고메시지는 이미 다른 트래픽에 상관없이 일정한 양으로 전체 네트워크에 전송되고 있다. 그리고 이 광고메시지는 캐시 메모리등을 이용할 수 없으며, 전체 MANET 노드가 발생시킨다. 이것은 이미 MANET에서 라우팅 프로토콜에 의한 광고메시지는 네트워크에 부하를 주고 있다는 의미가 된다. 또한 노드의 수가 늘어날수록 MANET 라우팅 프로토콜에 의한 광고메시지는 계속 증가할 것이다.

2) 노드의 이동과 능동형 PDP 탐색의 동작

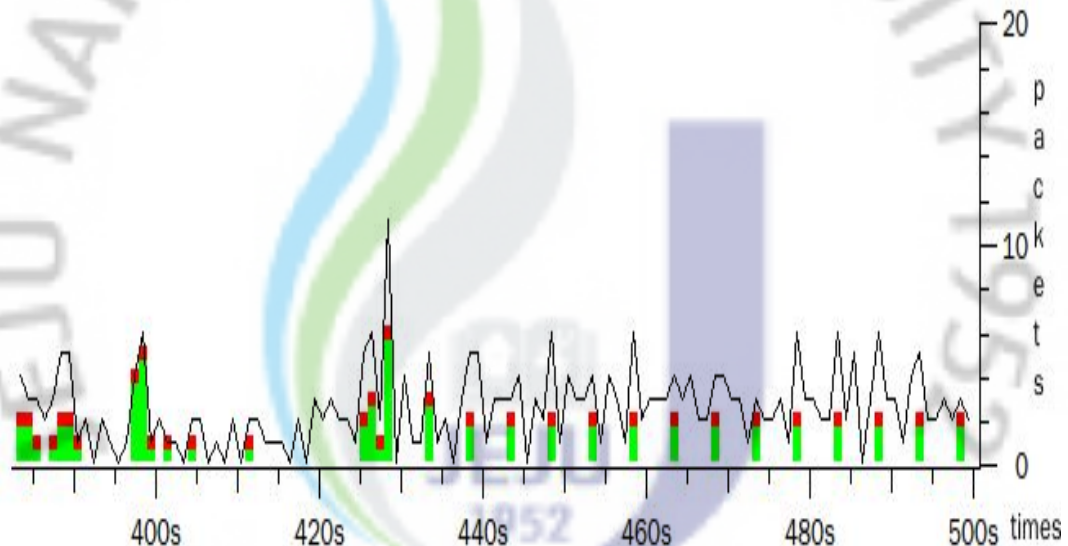


그림 60. PEP#2 노드 이동 중에 PDP#2에서의 COPS, KA, Total Packets

그림 60.는 이동 PEP 노드에서 전송되고 있는 트래픽을 분석한 것이다. 점으로 표시된 것은 KA 메시지이며, 막대그래프는 COPS 메시지, 선 그래프는 총 전송 패킷량을 나타내고 있다. 400초에서 425초 사이에서 해당 노드를 이동시켰을 때, 네트워크 도달 범위의 한계 거리까지 이동하게 되면 COPS 연결은 더 이상 일어나지 않지만, Cluster 2의 영역으로 가까워졌을 때 다시 COPS 연결이 이루어지고 있음을 볼 수 있다.

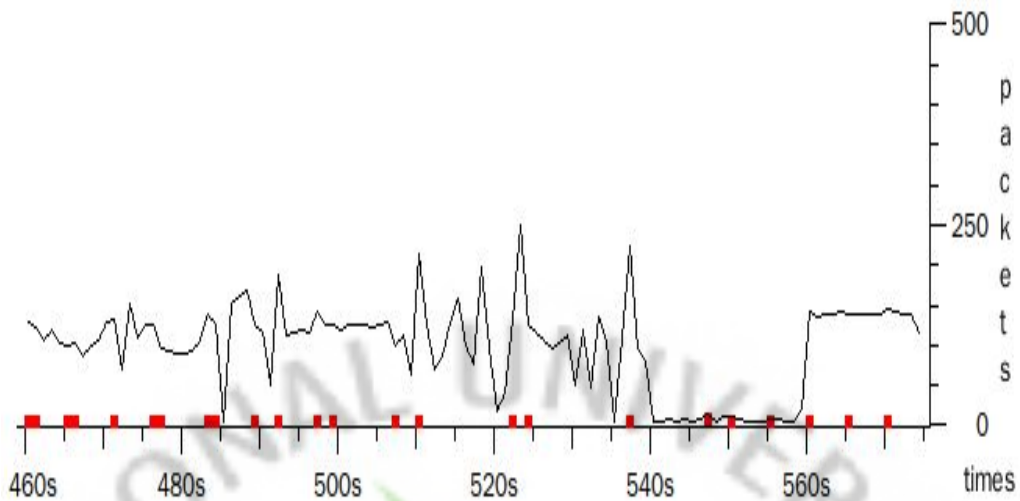


그림 61. TCP 전송과 COPS의 KA메시지

그림 61.는 이동 PEP에서 총 전송 패킷을 노드를 이동 시키며 수집한 결과이다. 540초까지의 전송 상태가 노드 PEP가 연결되었던 Cluster 1의 PDP으로부터 COPS 연결을 한 상태의 전송 상태이며 트래픽 발생기에 의한 트래픽과 하드웨어와 드라이버의 설정으로 전송효율을 최대한 낮춘 상태이므로 매우 불안정한 상태를 보여주고 있다. 540초에서 560초 사이는 해당 노드가 네트워크 범위의 한계 거리까지 이동한 모습이다. 이 노드를 Cluster 2의 방향으로 접근했을 때, PDP 탐색 메커니즘에 의한 새로운 PDP 선택 과정이 진행되고 560초 이후에 새로운 Cluster 2의 PDP로부터 COPS 연결을 맺고 정책을 수신한 상태이다. 트래픽 발생기에 의한 IP로의 동일한 트래픽을 전송 받고 있으나 Cluster 2의 PDP의 정책을 전송한 이후에 트래픽의 전송 상태는 보다 안정적이다. Cluster 2의 PDP의 정책은 트래픽을 전송하고 있는 IP의 전송 트래픽에 대하여 EF PHB 클래스를 적용하도록 되어 있다.

3) 비디오 스트리밍 실험

두 클러스터에서 PEP노드를 이동하여 2홉 토폴로지 상태에서 비디오 스트리밍을 전송하여 트래픽을 측정하였다.

	Total Packets(MB)	Bandwidth(Kbi ts/sec)	Delay(ms)
EF Traffic	9.23	774	0.091
Non EF Traffic	0.49	4.23	88.262

표 6. QoS Policy 적용에 따른 측정

표 6.는 비디오 스트리밍 실험에서 QoS Policy가 적용된 트래픽과 그렇지 않은 트래픽의 총 전송량, 대역폭, 지연값을 측정한 것이다. 상기한 값은 100초간의 전송한 값의 평균값이며, EF PHB로 처리된 트래픽이 EF PHB 트래픽과 그 이외의 트래픽의 값들을 비교한 것이다.

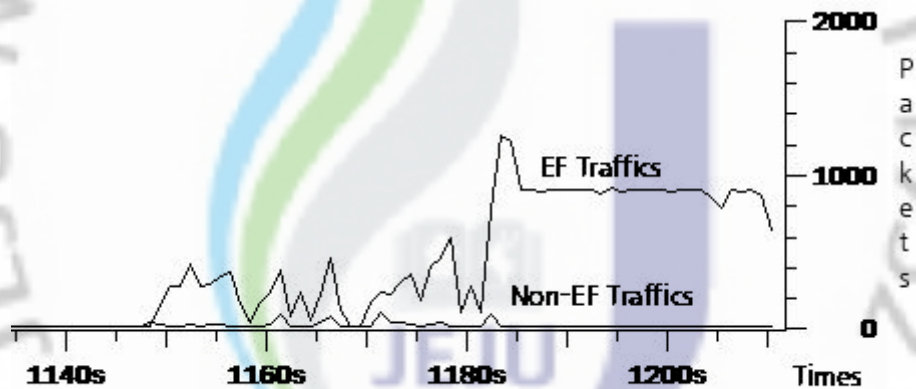


그림 62. QoS Policy 적용에 따른 전송 패킷량 변화

그림 61.은 비디오 스트리밍이 전송될 때의 EF PHB와 일반 트래픽의 전송 패킷량의 변화이다. EF PHB의 적용을 받은 트래픽은 1Mbps 정도의 전송을 보이고 있으며, 그 이외의 트래픽들은 EF 트래픽이 시작되면서 별도의 DSCP 마킹 값이 없으므로 그 이전에 비해 적은 패킷 전송량을 보이고 있다.

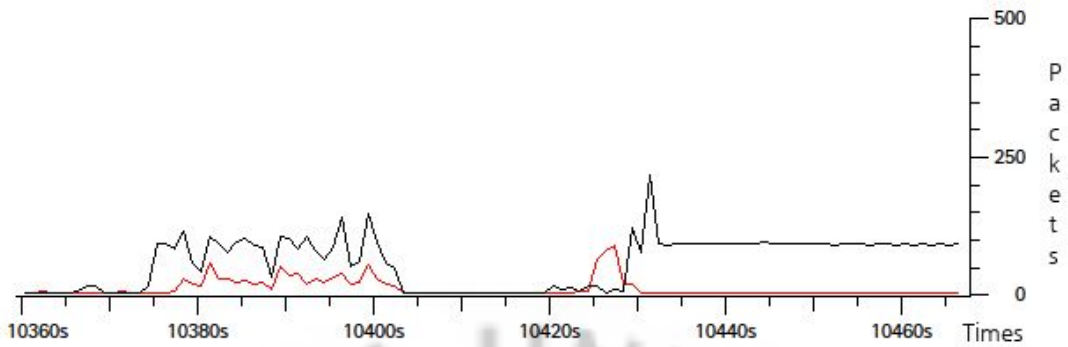


그림 63. 정책 적용시의 트래픽 변화

그림 62는 노드가 새로운 정책을 수신하여 트래픽에 적용했을 때의 변화를 나타낸다. 서로 다른 소스 어드레스에서 전송되는 두 가지의 트래픽은 10400초 부근까지 최선-노력방식으로 전송되지만, 정책을 수신한 이후에는 10430초 부근에서 해당 트래픽을 EF PHB로 처리하여 보다 안정적으로 전송하고 있다.

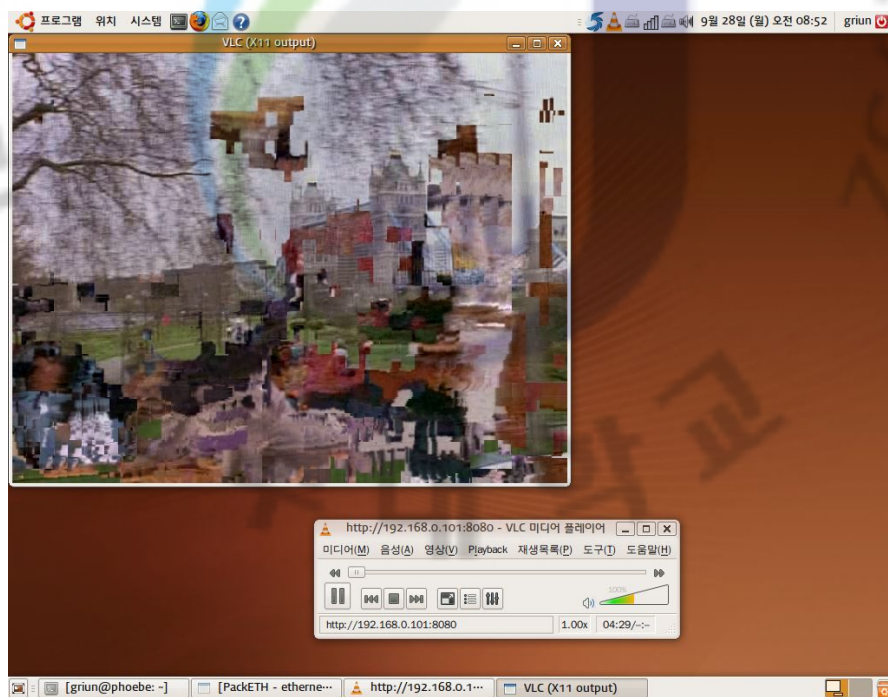


그림 64. PDP#1로부터 정책 수신 된 PEP#2의 비디오 스트리밍

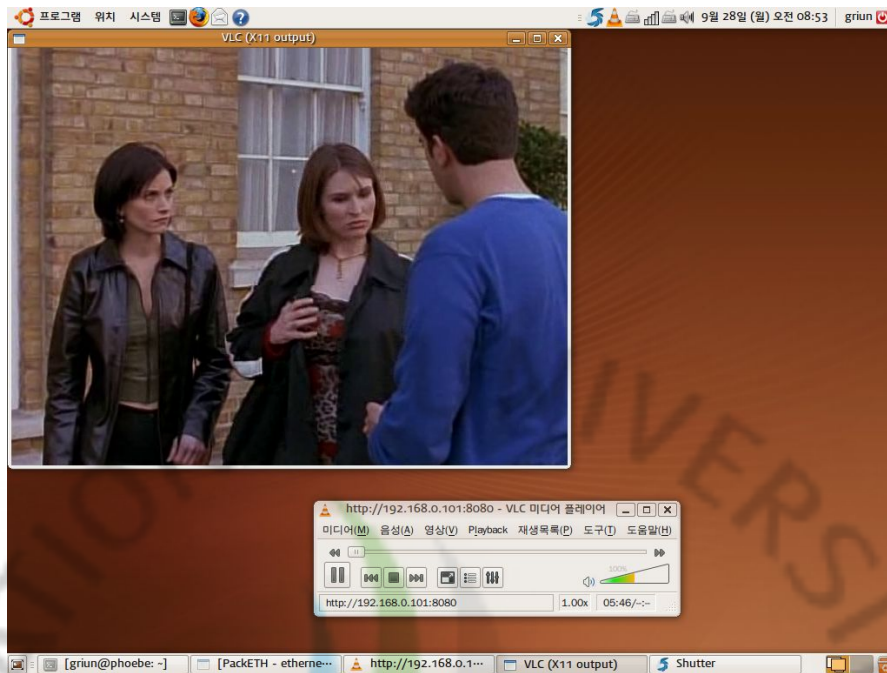


그림 65. PDP#2로부터 정책 수신 된 PEP#2의 비디오 스트리밍

그림 64.과 65는 동일한 소스 IP로부터 동영상 스트리밍으로 수신하게 한 이후에 PEP 노드를 이동 시켰을 때의 수신 상태이다. 그림 64.는 Cluster 1의 PDP에서 정책이 수신되고 있는 상태이며, 그림 65.는 Cluster 2의 PDP로부터 정책을 수신한 후의 전송상태이다. 테스트베드의 전송 능력이 4장 1절의 하드웨어 환경 구축에서 밝혔듯이 테스트베드의 물리적인 공간 축소를 위하여 최대한 전송 능력을 떨어뜨린 상태이므로 비디오 스트리밍 서비스는 Cluster 1에서 정책을 수신하거나 Cluster 2에 의해서 정책을 수신 받거나 둘 다 심하게 찌그러지기는 했으나 그림 65.에서 보듯이 비디오 스트리밍을 받는 해당 IP의 트래픽을 EF PHB로 처리하라는 정책을 적용한 상태에서는 보다 깨끗한 화면을 볼 수 있었다.

4. 요약

본 장에서는 구축 결과와 도출된 데이터를 근거로 결과를 기술하였다. 시뮬레이션에 의하여 증명된 능동형 PDP 탐색 기법은 실제의 구현에서 PEP의 광고메시지 문제, FOP 메시지의 사용 가능성, PREQ 구조에서 CC메시지가 불필요하며, PREQ의 헤더의 개선과 확장된 KA 메시지에 의한 PDP 정보 전송의 문제점

이 발견되었으며, 구현을 통해 해결점을 제시하였다. 그리고 제안하는 메커니즘이 실제로 동작함을 보였다.



VI. 결론

본 논문에서는 정책 기반 네트워크 관리를 MANET 환경에 도입하기 위한 연구를 수행하였다. 기존 네트워크들과 다른 여러 가지 특성을 가진 MANET에서 정책을 수신하지 못하는 노드의 수를 줄이고 정책 전송 영역 관리 메커니즘에 의한 네트워크의 부하를 개선하기 위하여 PEP가 능동적으로 PDP를 탐색하는 메커니즘을 실제 네트워크를 대상으로 구현하고 기존의 메커니즘보다 효율적임을 보였다.

이 메커니즘을 구현하기 위하여 표준 정책 기반 네트워크 관리의 구성 요소 중 정책 프로토콜인 COPS-PR을 확장하여 PEP의 요청을 받은 PEP나 PDP가 PDP 정보를 전송하도록 하여 정책을 수신하지 못하는 노드의 수를 줄였으며, PEP의 PDP정보 요청 과정에서 광고메시지를 사용하지 않도록 하여 정책 전송 영역 관리 메커니즘이 네트워크에 줄 수 있는 부하를 줄였다.

실험 환경에서 MANET 라우팅 프로토콜의 영향으로 총 패킷량을 100%로 볼 때, 라우팅 프로토콜이 새로운 노드 탐색과 연결 유지를 위해 전송한 광고메시지는 64.5%에 이르렀다. 가변적이기는 하지만 같은 실험환경에서 측정한 ARP 광고메시지의 수가 5.8%정도임을 고려한다면 유선 네트워크에 비해 매우 높은 수치의 광고메시지 사용을 보이고 있다. 이는 이미 MANET 라우팅 프로토콜에 의해 유선 네트워크에 비해 많은 부하가 있다는 것을 뜻한다. 그러므로 본 연구에서 이러한 부하를 줄이고자 능동형 PDP 탐색 기법을 개선한 부분은 다음과 같다.

- ① PEP의 광고메시지 없이 PREQ 전송 : MANET 라우팅의 특성 상 네트워크에 참여한 노드는 풀 메시 형태이거나 2홉 이상의 토폴로지를 형성하면 중간 노드가 다른 노드의 게이트웨이 형태로 포워딩하기 때문에 PEP에서는 k 홉의 값을 정하여 유니캐스트로 전송할 수 있도록 하였다.
- ② Find Other PDP 메시지 : PEP의 이동을 감지한 PDP가 해당 PEP에게 새로운 PDP를 탐색하라는 메시지이다. 이는 PEP도 이동을 감지할 수 있으므로

자율적으로 k 홉 밖으로 이동하면, 새로운 PDP를 탐색하도록 하였다.

③ PEP의 Connection Close 메시지 삭제 : 새로운 PDP와 연결할 PEP는 기존 PDP와 연결을 끊어야 하며, 더 이상 KA메시지를 전송하지 않게 된다. 능동형 PDP 탐색 기법은 표준 COPS-PR의 KA 메시지를 확장하여 이용하므로 표준 COPS-PR의 CC메시지와 중복되므로 삭제하였다.

④ PREQ 메시지 헤더의 TTL 필드 : 이 필드는 광고메시지를 어느 홉까지 전송할 것인지에 대한 값을 지정한다. 구현한 메커니즘에서는 광고메시지를 이용하지 않기 때문에 이 값을 시간으로 변환하여 지정한 시간이 지나도 응답하지 않으면, 해당 노드에 PREQ를 보내지 않도록 하였다.

이러한 MANET에서의 정책 전송 영역 관리 메커니즘에 대한 연구는 노드가 고정되어 있는 즉, PDP와 PEP가 정해져 있는 유선 네트워크에서는 불필요하지만 MANET에서는 반드시 필요한 영역이다. 특히 본 연구에서 구현하고 개선한 능동형 PDP 탐색 기법은 MANET의 실질적인 적용 분야로 대두되고 있는 VANET과 같이 노드의 이동성이 일반적인 MANET보다 강하며, 노드의 밀집도의 변화 또한 큰 네트워크에서 효용성과 효율성이 크리라 기대한다.

본 논문에서 구현한 능동형 PDP 탐색 기법은 Pro-Active방식의 MANET 라우팅 프로토콜을 이용하였다. 하지만, 수많은 MANET 라우팅을 위한 새로운 알고리즘들이 연구되어 발표되고 있다. MANET 라우팅은 기존의 Pro-Active, Reactive 방식뿐만 아니라, 여러 가지 메트릭 즉, 위치 기반 라우팅이나 파워 기반 라우팅 등이 연구되고 있다. 본 연구는 하나의 라우팅 프로토콜만을 대상으로 했으므로 다른 알고리즘을 가진 라우팅 프로토콜에 능동형 PDP 탐색 기법을 적용하기 위한 연구가 필요할 것이다. 이와 함께 향후 능동형 PDP 탐색 기법을 이용하여 다음과 같은 경우의 연구가 추가적으로 필요할 것이다.

① 클러스터 크기에 대한 연구 : 예전에는 클러스터의 크기가 경계값을 넘어가면 나누어지는 것에 주목하였다. 이 경계값은 다양할 수 있고 따라서 클러스터의 크기도 다양해 질 수 있다. 통제 이론적 기술(control-theoretic techniques)에 기반을 둔 클러스터 크기를 조정하는 연구가 가능할 것으로 보고 있다.

② PDP 노드가 이동하는 경우 : PDP가 다른 PDP가 관리하고 있는 영역으로 이동할 경우 해당 클러스터를 분리하는 동작이나 PDP가 이동한 클러스터에서 그 PDP에서 정책을 수신하던 PEP중 하나가 PDP가 되는 메커니즘을 연구할 필요가 있다. 이것은 클러스터를 나누는 연구와 유사하다.

③ Find Other PDP의 정책 적용에 관한 연구 : PDP에 의해서 정책이 전송되는 PEP 노드들에게 노드의 이동에 따라 자율적으로 새로운 PDP 노드를 탐색하도록 정책에 적용하여 처리할 수 있을 것이다. 이것은 MANET에 정책 기반 네트워크 관리를 도입함에 있어 자율적인 네트워크를 구축하는 MANET의 특성과 맞물려 유연성을 확보할 수 있을 것이다.

본 논문의 MANET에 정책 기반 네트워킹을 도입하기 위한 정책 기반 네트워크 관리에 대한 연구와 정책 전송 영역 관리 기법인 능동형 PDP 탐색 기법, 그리고 이에 관한 실제 네트워크를 대상으로 한 연구 방법이 앞으로 이어질 연구에 있어서 공헌을 하리라 기대한다.

참고 문헌

- [1] C. Siva Ram Murthy, B.S. Manoj, Ad Hoc Wireless Networks Architectures and Protocols, Prentice Hall PTR, New Jersey, pp.5-875, 2004년
- [2] A.K. Saha, D.B. Johnson, 'Modeling mobility for vehicular ad-hoc networks,' Proceedings of the 1st ACM international workshop on Vehicular ad hoc networks, ACM, Poster session, pp.91-92., 10월, 2004년
- [3] 이상선, 'VANET(Vehicle Ad-hoc Network)환경에서의 라우팅 기술 및 서비스 개발 동향', 한국정보과학회학술지, 22권, 1호, 3월, 2008년
- [4] D. Kosiur, Understanding Policy-Based Networking, first ed., Wiley, New York, 2001년
- [5] K. Phanse, L. Dasilva, 'Extending Policy-Based Management to Ad Hoc Networks', Virginia Polytechnic Institute and State University, IREAN Research Worksop-2003, 4월, 2003년
- [6] A. Westerinen, J. Schnizlein, J. Strassner, M. Scherling, B. Quinn, S. Herzog, A. Huynh, M. Carlson, J. Perry, S. Waldbusser, 'Terminology for policy-based management,' IEEE, Request for comments, 3198호, 2001년
- [7] Kyung-Jin Lee, Hanan Lutfiyya, Wang-Cheol Song, 'Management of PDP/PEP for PBNM in MANETs,' IEEE/IFIP NOMS 2006, pp.1-4, 4월, 2006년.
- [8] R. Chadha, H. Cheng, Y.H. Cheng, J. Chiang, A. Ghetie, G. Levin, H. Tanna, 'Policy-based mobile ad hoc network management', Proceedings of the Fifth IEEE Workshop on Policies for Distributed Systems and Networks (POLICY'04), 6월, 2004년
- [9] The IETF Policy Framework Working Group: Charter available at <http://www.ietf.org/html.charters/policy-charter.html>.
Tomas Krag, Sebastian Buettrich, Wireless Mesh Networking. O'Reilly Wireless Dev Center. USA, <http://www.oreillynet.com/pub/a/wire->

less/2004/01/22/wirelessmesh.html, 2004

BBC, 'Mobile system promises free calls', BBC, UK, <http://news.bbc.co.uk/2/hi/technology/6987784.stm>, 2007

[10] A. Parasuraman, V.A. Zeithaml, L.L. Berry, 'A conceptual model of service quality and its implications for future research,' *The Journal of Marketing*, 49권, pp.41 - 50, 1985년

[11] D. Chalmers, M. Sloman, 'A survey of Quality of Service in mobile computing environments', *IEEE Communications Survey*, Second Quarter, 1999년

[12] M. Mirhakkak, N. Schult, D. Thomson, 'Dynamic quality of service for mobile ad hoc networks', *Proceedings of the First Annual Workshop on Mobile and Ad Hoc Networking and Computing (MobiHOC)*, Poster session, pp.137 - 138, 2000년

[13] S. Lee, G. Ahn, X. Zhang, A. Campbell, 'INSIGNIA: an IP-based Quality of Service framework for mobile ad hoc networks', *Journal of Parallel and Distributed Computing*, 60권, 4호, pp.374 - 406, 5월, 2000년

[14] H. Xiao, W. Seah, A. Lo, K. Chua, 'A flexible Quality of Service model for mobile ad-hoc networks', *Proceedings of the IEEE Vehicular Technology Conference (VTC)*, 1권, pp.445 - 449, 6월, 2000년

[15] P. Sinha, R. Sivakumar, V. Bharghavan, 'CEDAR: a core-extraction distributed ad hoc routing algorithm', *IEEE Journal on Selected Areas in Communications (JSAC)*, 8권, 17호, pp.1454 - 1465, 1999년

[16] E. Elmallah, H. Hassanein, H. AboElFotoh, 'Supporting QoS routing in mobile ad hoc networks using probabilistic locality and load balancing', *The Proceedings of IEEE Global Telecommunications Conference (Globecom)*, 5권, 2001년

[17] 전자신문 '한국형 아우토파브 스마트 하이웨이 개발 착수', 전자신문, 8월, 2007년

[18] C. Laurendeau and M. Barbeau, 'Threats to Security in DSRC/WAVE', *ADHOC-NOW 2006, LNCS 4104*, pp. 266 - 79, 2006년

[19] A. Festag, G. Noecker, M. Strassberger, A. Lübke, B. Bochow, M.

- Torrent-Moreno, S. Schnauffer, R. Eigner, C. Patrinescu, J. Kunisch, , others, 'NoW-Network on Wheels: Project Objectives, Technology and Achievements,' Proc. 5th International Workshop on Intelligent Transportation, pp.211 - 216, 2008년
- [20] 오현서, 최혜옥, 조한벽, '차량 통신 네트워크 기술 동향', ETRI, 주간기술동향, 23권 5호, 10월, 2008년
- [21] D. Kidston, J. Robinson, 'Distributed network management for coalition deployments', Proceedings of the IEEE Military Communications Conference (MILCOM) 1권, 1호, pp.460 - 464, 2000년
- [22] J. Brand, G. Hartwig, 'Ad hoc network management with C2 data models', Proceedings of IEEE Southeast Conference, 1권, 1호, pp.100 - 104, 4월, 2001년
- [23] S. Lee, M. Gerla, 'Split multipath routing with maximally disjoint paths in ad hoc networks', Proceedings of IEEE International Conference on Communications (ICC), 10권, pp.3201 - 3205. 2001년
- [24] C. Lin, 'On-demand QoS routing in multihop mobile networks', Proceedings of IEEE INFOCOM, 3권, pp.1735 - 1744. 2001년
- [25] M. Marchese, QoS over heterogeneous networks, John Wiley & Sons, England, pp.1-13, 2007년
- [26] R. Sahita, S. Hahn, K. Chan, , K. McCloghrie, 'Framework policy information base,' IETF, Request For Comments (RFC), 3318호, 2003년
- [27] M. Rose , K. McCloghrie, 'Structure and Identification of Management Information for TCP/IP-based internets,' Request For Comments, 1155호, 5월, 1990년
- [28] D. Harrington, R. Presuhn, B. Wijnen, 'An architecture for describing simple network management protocol (SNMP) management frameworks', Request for Comments, 3411호, 11월, 2002년
- [29] K. McCloghrie , M. Rose, 'Management Information Base for network management of TCP/IP-based internets', MIB-II, STD 17, Request For Comments, 1213호, 3월, 1991년
- [30] W. Yeong , T. Howes, S. Kille, 'Lightweight Directory Access Protocol',

Request For Comments, 1777호, 3월, 1995년

- [31] K. Chan, R. Sahita, S. Hahn, , K. McCloghrie, 'RFC3317: Differentiated Services Quality of Service Policy Information Base,' IEEE, Request For Comments, 3317호, 3월, 2003년
- [32] D. Durham, J. Boyle, R. Cohen, S. Herzog, R. Rajan, , A. Sastry, 'The COPS (Common Open Policy Service) Protocol,' IEEE, Request For Comments, 2748호, 1월, 2000년
- [33] K. Chan, J. Seligson, D. Durham, S. Gai, K. McCloghrie, S. Herzog, F. Reichmeyer, R. Yavatkar, , A. Smith, 'RFC3084: COPS Usage for Policy Provisioning (COPS-PR),' IEEE, Request For Comments Editor, 3월, 2001년
- [34] M. Raya, P. Papadimitratos and JP. Hubaux, 'Securing Vehicular Communications', IEEE Wireless Communication Magazine, Special Issue on Inter-Vehicular Communications, 10월, 2006년
- [35] V.B. Iversen, 'Teletraffic Engineering Handbook', ITU-T Study Group 2, 2권, 16호, 2002년
- [36] Peuhkuri M., 'IP Quality of Service', Helsinki University of Technology, Laboratory of Telecommunications Technology, 1999.
- [37] R. Braden, D. Clark, S. Shenker, 'RFC1633: Integrated Services in the Internet Architecture: an Overview,' IEEE, Request For Comments, 1633호, 6월, 1994년
- [38] K. Nichols, S. Blake, F. Baker, D. Black, 'RFC2474: Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers,' IEEE, Request For Comments, 2474호, 12월, 1998년
- [39] K. Ramakrishnan, S. Floyd, D. Black, 'RFC3168: The Addition of Explicit Congestion Notification (ECN) to IP,' IEEE, Request For Comments, 3168호, 9월, 2001년
- [40] N. Ek, 'IEEE 802.1 P, Q-QoS on the MAC level', IEEE, Apr, 24권, pp. 3-6, 4월, 1999년
- [41] J. Yu, P. Chong, 'A survey of Clustering Schemes for Mobile Ad Hoc Networks', IEEE Communication Surveys & Tutorials, 7권, 1호, pp. 32-48, 2005년

- [42] B. Hubert, 'Linux advanced routing & traffic control HOWTO', setembro de, 2002년
- [43] Ubuntu Linux Community, 'WifiDocsWirelessCardsSupported', Ubuntu Linux Community Documents Wiki, <https://help.ubuntu.com/community/WifiDocs/WirelessCardsSupported>, 10월, 2009년
- [44] Linus Tovalds, Linux Kernel Organization. Inc, http://www.kernel.org/_11 월, 2009년
- [45] T. Clausen, et al., 'OLSR Routing Protocol' IETF, Request For Comments, 3626호, 10월, 2003년
- [46] Ravi Sahita, 'COPS Protocol Provides New Way of Delivering Services on the Network', Intel Developer UPDATE Magazine, pp. 1 - 6, 5월, 2002년
- [47] Torger, Anders; Lidén, Erik, 'Implementation and evaluation of the Common Open Policy Service (COPS) protocol and its use for policy provisioning', Luleå University of Technology Sweden, 4월, 2000년
- [48] H. Halim, M. Darmadi in their Honours thesis "Implementation of bandwidth broker using COPS-PR", University of New South Wales Australia, School Of Computing Science and Engineering, 11월, 2000년
- [49] 경북대학교 차세대 정보통신 연구소, 'BB DB 및 Admission Control 설계 및 연구', 한국전자통신연구원, 12월, 2002년
- [50] Department of Communications Engineering, 'Faster Pro and diffserv.', Tampere University of Tech. Finland, 9월, 2002년
- [51] A. Tirumala, F. Qin, J. Dugan, J. Ferguson, , K. Gibbs, 'Iperf The TCP/UDP bandwidth measurement tool,' <http://dast.nlanr.net/Projects/Iperf>, 2004년
- [52] Wireshark Foundation, 'Wireshark network protocol analyzer', <http://www.wireshark.org>, 9월, 2009년
- [53] VLC team, 'VLC media player', <http://www.videolan.org/vlc/>, 9월, 2009년