

---

# MIB-II 기반 계층적 네트워크 구조 분석 및 패킷우회 검출 알고리즘

김진천\*

MIB-II based Algorithm for hierarchical network analysis  
and detection of detour routing paths

Jin-chun Kim\*

---

이 논문은 2001학년도 경성대학교 학술지원연구비에 의하여 연구되었음

---

## 요약

네트워크 관리는 메일, 화상회의, 웹, 데이터베이스 등이 작동하는 네트워크 환경에서 더욱 중요한 문제가 되고 있다. 패킷의 적절한 경로를 발견하기 위하여 라우터간에 정보를 교환하는 라우팅 프로토콜이 사용되고 있으나 관리자의 실수로 만들어진 비정상적인 라우팅 테이블로 인하여 데이터를 전송하는 브리지 또는 라우터가 네트워크 디자인의 관점에서 비효율적인 경로로 전송할 수 있다. 이 논문에서는 네트워크상에서 발생할 수 있는 우회 경로를 발견하는 문제를 논의한다. SNMP, MIB로부터의 정보를 사용 계층적 네트워크 구조 분석을 기반으로 우회경로 패스를 발견하는 새로운 알고리즘을 제시하고, 제안된 알고리즘을 검증하기 위해 미리 선언된 데이터를 이용한 시뮬레이션을 수행하였다.

## ABSTRACT

Network Management become more and more important issue in the network environment in which many applications such as Mail, teleconferencing, WWW and database software are operated. It can be possible for The Bridge and Router forwarding data to select next hop device which results in routing incorrect path from the viewpoint of network design. In this paper we address the problem of finding the detour routing path due to incorrect setting on routing devices. We propose the new algorithm for finding detour routing path based on hierarchical network structure analysis using information from SNMP MIB. To prove the correctness of the proposed algorithm we have done simulation with predefined data. Simulation results show that the algorithm finds detour path correctly.

## 키워드

우회경로, MIB-II, SNMP, 계층적 네트워크

## 1. 서론

오늘날 정보화가 진행되어 컴퓨터 네트워크는 급속도로 발전하고 있다. 많은 회사나 공공단체

또는 학교에서 메일, 문서공유 등의 간단한 작업에서부터, 데이터베이스의 연동, 화상회의, 음성 및 데이터 통신 등 복잡한 작업에까지 많은 부분

---

\*경성대학교 컴퓨터공학과

접수일자 : 2003. 10. 13

의 업무를 네트워크에 의존하고 있어, 네트워크 관리의 필요성은 더욱 중요해 지고 있다.[1]

네트워크 구성 중 기능을 정의하여 정의된 각 기능별로 계층을 분리하여 구현한 계층적 네트워크 구성이 있다. 이는 네트워크 설치의 최소 비용, 관리의 분산 및 장애 국부화, 네트워크의 위상 변화적응에 용이한 장점이 있다. 또한 라우팅 프로토콜도 이를 토대로 구성되고 있는 추세이며 MIB 값 형태로 수집한 정보로 네트워크의 구성에 맞는 계층적 네트워크의 형태와 같은 트리를 생성할 수 있다.[2]

패킷의 적절한 경로를 발견하기 위해 라우터간에 정보를 교환하는 라우팅 프로토콜이 있으나 관리자의 관리 실수로 인해 만들어진 라우팅 테이블로 인하여 만들어진 패킷의 우회 경로가 존재할 수 있다.[3] 이는 한 세그먼트 내 몇몇 시스템의 네트워크 불능보다 발견하기 힘든 상황이다. 패킷의 최적 경로를 찾는 라우팅 프로토콜은 많이 존재하나 그러나 네트워크 디자인 분석을 토대로 패킷의 전송로를 예측하여 우회 경로를 발견하는 네트워크 관리 접근은 아직 없었다.

본 논문에서 제안하는 알고리즘은 계층적 네트워크 디자인에서 MIB-II의 필요한 정보를 수집하여 장비의 종류를 분석하고, 라우터로 판명된 정보를 가지고 네트워크 계층에 맞게 트리를 생성하여 패킷이 움직일 수 있는 경로를 예측한다. 그리고 이를 기반으로 우회 경로로 패킷을 라우팅하고 있는 라우터를 검출한다.

## II. 패킷 우회 경로 검출 알고리즘

### 2.1 개요

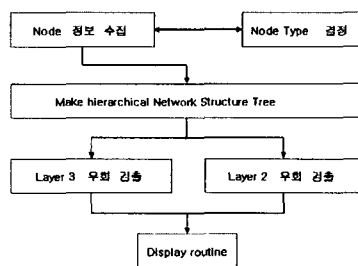


그림 1. 패킷의 우회 경로 발견 알고리즘의 순서  
Fig. 1 Sequence of detour routing path finding algorithm

그림1 은 패킷의 우회 경로 발견 알고리즘의 순서도를 나타내고 있다. 계층적 네트워크에서 패킷의 흐름을 이해하기 위한 첫 단계로 네트워크에 어떠한 장비로 구성되어 있는지를 조사한다.

조사된 각 노드들의 네트워크 상에서의 실질적인 사용형태를 조사해야 한다. Layer 3 이상의 스위치 장비는 Layer 2 스위칭 기능뿐만 아니라, 각 장비의 세팅에 따라서 라우팅 기능을 가질 수도 있어 스위치가 될 수도 있고 라우터로 인식 될 수도 있다.

노드들을 수집한 후 각 노드들의 연결 상태를 조사하기 위해 네트워크의 구조를 해석하여 트리 형태로 나타낸다. 트리 형태의 데이터를 근거로 하여, Layer 2의 경로가 적절한 구조인지를 수집된 Spanning 트리 값을 이용하여 조사하고, 각 장비들의 라우팅 테이블과 비교하여 트리 형태의 네트워크에서 모든 경로가 계층적 네트워크 구조에 적합한지를 조사함으로써, 적절한 Layer 3의 패킷의 흐름을 조사 할 수 있게 된다.

### 2.2 Node type 결정

각 장비들을 구별하는 방법은 그림 2의 순서도에 나타나있다. MIB-II의 System.sysService는 Node의 7 layer상의 기능을 말해 준다. 각 Layer를 L이라 두었을 때, sysService의 값은 2(L-1)의 합이 된다. 각 장비의 능력에 따라 라우터 또는 스위치 그리고 워크스테이션으로 세팅이 가능하다. System.sysService의 값만으로는 장비의 속성을 알 수가 없어 ip.ipForwarding의 값과 소유하는 segment의 수도 노드 타입 결정에 참여하게 된다.[4]

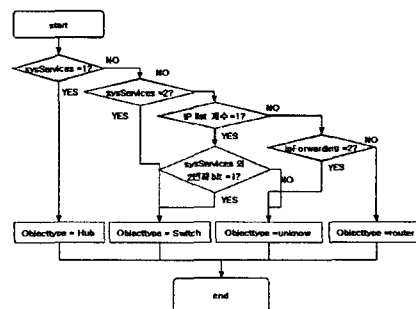


그림 2. 노드 타입의 결정 순서도  
Fig. 2 Sequence of determining Node Type

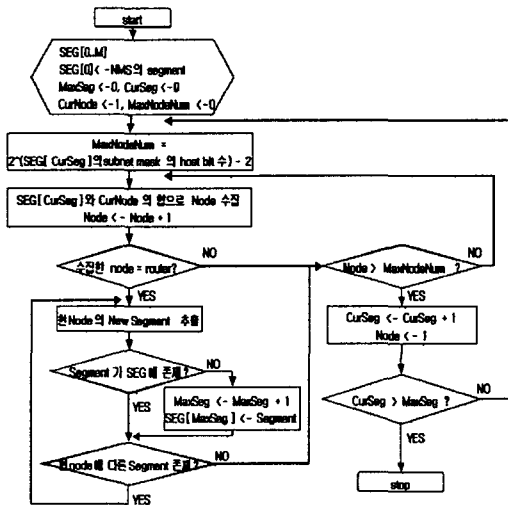


그림 3. 노드 수집 단계의 순서도  
Fig. 3 Sequence of Node collect process

### 2.3 Node collect process

그림 3의 순서도에서 초기값을 준 다음 차례로 노드의 정보를 수집한다. 노드가 라우터라면, Segment를 추출하여 MaxSeg를 증가하고, SEG에 추출한 Segment를 저장한다. 또한 CurNode의 값이 MaxNodeNum와 같다면 Segment의 모든 노드의 정보를 읽었으므로 다음 Segment로 확장한다. 만약 다음 세그먼트가 없다면 Node collect process는 끝난다.

Object type	Router, Switch, Hub, other	system.sysServices
Identifier	IP Address	ip.ipAddrTable ipAdEntAddr ipAdEntNetMask
Service	Node Services	
IP Address Table	IP Address & Netmask List	ip.ipRouteTable ipRouteDest ipRouteMask ipRouteNextHop ipRouteProto
Routing Table	Destination Netmask Next hop ip address IP route protocol	interface.ifTable ifIndex ifPhysAddress
MAC Address Table	Interface.if Table	dot1dBridge.BridgeAddress
BridgeAddress	Device의 대표 bridge Address	dot1dSp.Priority
STP Priority	STP의 Priority	dot1dSp.DesignatedRoot
DesignatedRoot	Device의 Designated Root	

그림 4. 오브젝트 데이터베이스의 필드  
Fig. 4 Fields of Object Database

### 2.4 Node data

그림 4는 각 노드들에게 조사할 데이터들이다. Node collect process에서 찾은 IP 주소로 Ping을 보내고, 이에 대답하는 IP주소에 SNMP Query를 보낸다. SNMP에 답하는 Node에서 IP Address Table, Routing Table, Physical Address Table, Bridge Address, STP Priority, Designated Root Bridge를 수집한다.

### 2.5 계층적 트리 구성

#### 1) Leaf 라우터간 연결 추론

MIB에서 얻을 수 있는 각 장비의 IP 주소만으로 각 네트워크의 연결상태를 추론할 수 있다. 각 라우터가 같은 세그먼트를 공유한다면, 서로 연결되어 있는 것이다.

#### 2) 트리 생성 알고리즘

계층적 네트워크에서 라우터의 IP 주소만으로 연결 상태의 트리를 생성하는 알고리즘이다. 이때 외부망과의 연결 세그먼트를 제거하고 만약 제거된 외부망과의 연결 세그먼트가 가지고 있는 세그먼트가 단 1개 존재할 때 연결 라우터도 제거한 후에 트리를 생성한다. 또한 알고리즘이 반복될 때는 사용한 노드는 제외하고 반복한다. 알고리즘은 네트워크에서 Redundant 경로의 유무에 따라 다른 루틴을 제공한다.

#### ■ 계층적 네트워크 분석을 위한 트리 생성 알고리즘

$n, m : \{1, 2\}$

$N$  : Leaf 라우터의 개수

$R$  : 존재하는 모든 라우터의 집합 =  $\{R_1, R_2..\}$

$RL$  : 존재하는 모든 Leaf 라우터의 집합

$S$  : 존재하는 모든 Segment의 집합 =  $\{S_1, S_2..\}$

$SL$  : 존재하는 모든 Leaf Segment의 집합

$C$  :  $N$ 개의 Leaf 라우터에서 2개로 구성된 조합의 집합 =  $\{C_1, C_2..\}$

$P(C_n)$  :  $C_n$ 간 Path List (Leaf 라우터를 포함)

$E_m(C_n)$  :  $C_n$ 간 Path의  $m$ 번째 라우터 또는 Segment

$CN(R_n)$  : 모든  $P(C_n)$ 의 List에 포함된  $R_n$ 의 개수

$CN(S_n)$  : 모든  $P(C_n)$ 의 List에 포함된  $S_n$ 의 개수

- 모든  $C_n$ 과  $P(C_n)$ 를 계산
- 모든  $P(C_n)$ 의 개수를 계산

▣ Redundant 경로가 존재하지 않을 때

- $NC2$ 만큼의 개수를 가치는  $CN(R_n)$ ,  $CN(S_n)$ 의 존재유무 판단

▷ 존재할 때  
: 1개의 백본이 존재함.  
검출된  $R_n$ 또는  $S_n$ 을 Root Node 로 Tree 생성 후 끝낸다.

▷ 존재하지 않을 때  
: 2개 이상의 백본이 존재함.  
 $P(C_n)$ 중에서 길이가 3인 경로를 추출하여  $P(C_k)$ 이라 두고 모든  $E2(C_k)$ 를 검색

•  $(E1(C_k) \cup E3(C_k)) \supset RL$ 이 True인지 확인  
▷ True일 경우  
:  $E2(C_k)$ 를 Leaf로 간주하여 알고리즘 반복.  
▷ False일 경우  
:  $E2(C_k)$ 를 포함하지 않는  $P(C_n)$ 중에서 길이가 5인 Path를 추출하여  $P(C_q)$ 라 두고 모든  $E3(C_q)$ 를 검색

•  $(E1(C_k) \cup E3(C_k) \cup E1(C_q) \cup E5(C_q)) \supset RL$  가 True임을 확인  
▷ True일 때,  
:  $E2(C_k)$ ,  $E3(C_q)$  Leaf로 간주하여 알고리즘 반복.  
▷ False일 때,  
: 계층적 네트워크가 아님을 알수있다.

▣ Redundant 경로가 존재하는 네트워크 일 때

- $P(C_n)$ 중에서 길이가 3인 경로를 추출하여  $P(C_k)$ 이라 두고 모든  $E2(C_k)$ 를 검색
- $E2(C_k)$ 를 포함하지 않는  $P(C_n)$ 중에서 길이

가 5인 경로를 추출하여  $P(C_q)$ 이라 두고 모든  $E3(C_q)$ 를 검색

- $(E1(C_k) \cup E3(C_k) \cup E1(C_q) \cup E5(C_q)) \supset RL$  가 True임을 확인
- ▷ False일 때  
: 계층적 네트워크가 아님을 알수 있다.
- ▷ True일 때  
 $K(C_k) = \{ E1(C_k), E3(C_k) \}$   
 $Q(C_q) = \{ E1(C_q), E5(C_q) \}$ 라 두었을 때 모든 집합  $K(C_k)$ ,  $Q(C_q)$ 들간에 교집합이 공집합 아닐 때 통합한다.  
 $E2(C_k)$ ,  $E3(C_q)$  Leaf 노드로 간주하여 계층적 네트워크 분석을 위한 Tree 생성 알고리즘 반복

2.6 layer 3 & layer 2 우회 검출

계층적 네트워크는 각 계층의 기능을 분리하여 설계한다. 상위 계층은 전적으로 하위 계층의 패킷의 전송을 목적으로 설계된다. 그러므로 계층적인 네트워크에서 패킷은 단 한번만 상위 계층으로 전송되었다가 다시 하위 계층으로 전송되게 디자인 되어있다.

따라서, 생성한 트리를 기반으로 각 leaf 라우터의 라우팅 테이블을 조사 한다. 우선 Leaf 라우터의 라우팅 테이블에서 임의의 목적지 주소에 대한 Next Hop 라우터로 이동하고, 다음 Next Hop 라우터를 찾아 이동하여, 목적지까지 도착하는 경로와 트리 상에서 어떠한 계층으로 이동 되는지를 조사함으로써 패킷의 우회를 알 수 있다. 이 또한 leaf 라우터에서 두개로 이루어진 모든 조합들 간의 패킷의 전송로를 따라 라우팅 테이블을 조사한다.

Layer 2의 우회 검출은 계층적 네트워크의 구성상 Spanning Tree의 Designated Root는 Leaf Segment의 상위 라우터가 Designated Root가 되어야 한다. 만약 세그먼트의 내부에 있는 스위치가 Designated Root가 되어 있다면, 반드시 우회를 하는 것은 아닌, 우회의 가능성이 존재한다.

### III. 알고리즘 검증

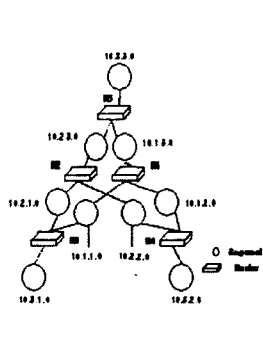
#### 3.1 시뮬레이션 방법

실제적인 네트워크 구성에서 우회로를 만들어, 각 장비의 데이터를 수집하는 것은 현실적으로 실현하기 힘들다. 따라서 우회로를 가지는 임의의 계층적 네트워크 구조에 해당하는 IP 어드레스 데이터와 IP 라우팅 데이터를 만들고 이 데이터로서 계층적 트리를 생성 layer 3의 우회로를 검출하였다. 정보를 수집하는 과정만 생략 되었을 뿐 나머지 상황은 동일하게 시뮬레이션 된다. layer 2의 데이터는 방대한 양 때문에 인위적인 생성이 불가능하여 시뮬레이션에 제외하였다.

#### 3.2 입력 데이터 형태

그림 5는 시뮬레이션에서 사용된 라우터의 IP 주소 리스트이다. R1에서 R3의 Leaf Segment인 10.3.1.0을 목적으로 하는 라우팅 테이블에서 Next Hop이 R3가 아닌 R5로 우회로를 생성하였다.

그림 6은 시뮬레이션에서 사용된 IP 라우팅 테이블이다. 우회를 검출하기 위해 사용된다. 그림 밑줄친 부분의 R1의 라우팅 테이블에서 우회 경로를 임의로 설정하였다.



R1에서 10.3.1.0에 목적지를 갖는 NextHop의 주소로 R5를 명시서 우회로를 만들

RouterID	IPAddrList	Netmask
R1	10.1.1.1	255.255.255.0
R1	10.1.2.1	255.255.255.0
R1	10.1.3.1	255.255.255.0
R2	10.2.1.1	255.255.255.0
R2	10.2.2.1	255.255.255.0
R2	10.2.3.1	255.255.255.0
R3	10.1.1.2	255.255.255.0
R3	10.2.1.2	255.255.255.0
R3	10.3.1.1	255.255.255.0
R4	10.1.2.2	255.255.255.0
R4	10.2.2.2	255.255.255.0
R4	10.3.2.1	255.255.255.0
R5	10.1.3.2	255.255.255.0
R5	10.2.3.2	255.255.255.0
R5	10.3.3.1	255.255.255.0

그림 5. 네트워크 구조 및 IP 주소 리스트  
Fig. 5 Network structure and IP Address List

RouterID	Destination	Next-hop	RouterID	Destination	NextHop	RouterID	Destination	NextHop
R1	10.1.1.0	10.1.1.1	R3	10.1.1.0	10.1.1.2	R5	10.1.1.0	10.1.3.1
R1	10.1.2.0	10.1.2.1	R3	10.1.2.0	10.1.1.1	R5	10.1.2.0	10.1.3.1
R1	10.1.3.0	10.1.3.1	R3	10.1.3.0	10.1.1.1	R5	10.1.3.0	10.1.3.2
R1	10.2.1.0	10.1.1.2	R3	10.2.1.0	10.2.1.2	R5	10.2.1.0	10.2.3.1
R1	10.2.2.0	10.1.2.2	R3	10.2.2.0	10.2.1.1	R5	10.2.2.0	10.2.3.1
R1	10.2.3.0	10.1.3.2	R3	10.2.3.0	10.2.1.1	R5	10.2.3.0	10.2.3.2
R1	10.3.1.0	10.1.3.2	R3	10.3.1.0	10.3.1.1	R5	10.3.1.0	10.2.3.1
R1	10.3.2.0	10.1.2.2	R3	10.3.2.0	10.1.1.1	R5	10.3.2.0	10.1.2.1
R1	10.3.3.0	10.1.3.2	R3	10.3.3.0	10.2.1.1	R5	10.3.3.0	10.3.3.1
R2	10.1.1.0	10.2.1.2	R4	10.1.1.0	10.1.2.1			
R2	10.1.2.0	10.2.2.2	R4	10.1.2.0	10.1.2.2			
R2	10.1.3.0	10.2.3.2	R4	10.1.3.0	10.1.2.1			
R2	10.2.1.0	10.2.1.1	R4	10.2.1.0	10.2.2.1			
R2	10.2.2.0	10.2.2.1	R4	10.2.2.0	10.2.2.2			
R2	10.2.3.0	10.2.3.1	R4	10.2.3.0	10.2.2.1			
R2	10.3.1.0	10.2.1.2	R4	10.3.1.0	10.2.2.1			
R2	10.3.2.0	10.2.2.2	R4	10.3.2.0	10.3.2.1			
R2	10.3.3.0	10.2.3.2	R4	10.3.3.0	10.1.2.1			

그림 6. IP 라우팅 테이블  
Fig. 6 IP Routing Table

#### 3.3 우회 경로 검출 시뮬레이션

그림 7은 사용될 라우터의 IP 주소 리스트와 네트워크에 존재하는 세그먼트를 나타낸다.

그림 8은 각 노드간의 연결을 추론하기 위해 트리를 생성하여 가능한 모든 경로를 찾아내는 트리 생성 결과이다. 즉 R1과 R5의 연결가능한 모든 경로는 R1-S1-R3-S2-R2-S8-R5, R5-S5-R5, R1-S4-R4-S6-R2-S8-R5이다.

Router Data			Segment Data	
IPAddrList	Netmask	RouterID	Segment	IPAddrList
10.1.1.1	255.255.255.0	R1	S1	10.1.1.0
10.1.2.1	255.255.255.0	R1	S2	10.2.1.0
10.1.3.1	255.255.255.0	R1	S3	10.3.1.0
10.2.1.1	255.255.255.0	R2	S4	10.1.2.0
10.2.2.1	255.255.255.0	R2	S5	10.1.3.0
10.2.3.1	255.255.255.0	R2	S6	10.2.2.0
10.1.1.2	255.255.255.0	R3	S7	10.3.2.0
10.2.1.2	255.255.255.0	R3	S8	10.2.3.0
10.1.2.2	255.255.255.0	R4	S9	10.3.3.0
10.2.2.2	255.255.255.0	R4		
10.3.2.1	255.255.255.0	R4		
10.1.3.2	255.255.255.0	R5		
10.2.3.2	255.255.255.0	R5		
10.3.3.1	255.255.255.0	R5		

그림 7. 라우터 데이터와 추론된 세그먼트 데이터  
Fig. 7 Router data and Segment Data

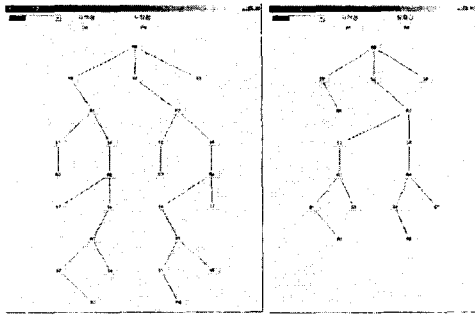


그림 8. 각 노드간의 연결 경로 발견 트리  
Fig. 8 Tree for finding connection routes

그림 10은 조사될 라우팅 테이블과 우회 경로 검출 결과이다. 2개의 우회 경로가 있다. R1에서의 라우팅 우회를 정확하게 검출해 내는 모습을 보여준다.

그림 11은 생성된 트리 기반으로 계층적인 네트워크의 구조를 정확하게 인식하는 모습을 보여준다.

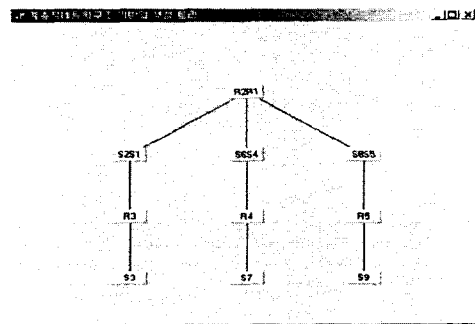


그림 9. 계층적 네트워크에 기반한 트리  
Fig. 9 Tree based on hierarchical network

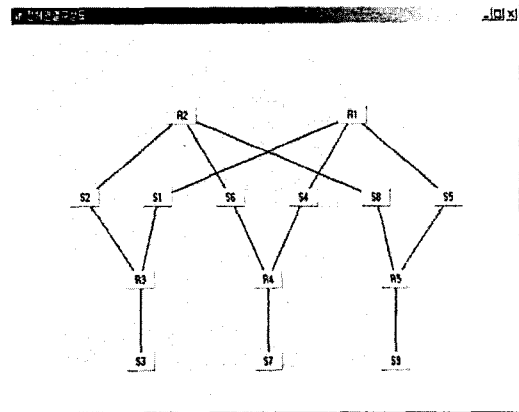


그림 11. 생성된 트리 기반 전체 네트워크 구조  
Fig. 11 Structure of network based on Tree

그림 9는 계층적 네트워크에 기반한 트리 생성 결과이다. 이 값으로서 각 계층을 인지하여 패킷의 우회를 검출한다.

#### IV. 결론

본 논문에서는 계층적 네트워크에서 MIB-II로부터 필요한 정보를 수집하여 장비의 종류를 분석한 후 라우터로 판명된 정보를 기반으로 네트워크 계층에 맞게 트리를 생성, 패킷이 움직일 수 있는 경로를 예측함으로써 우회 경로로 패킷을 라우팅하고 있는 라우터를 검출하는 알고리즘을 제안하였다.

가상의 데이터로 실행한 시뮬레이션을 통하여 알고리즘이 정확한 우회 경로를 발견한다는 것을 증명하였다.

향후 연구 과제로는 계층적 구조의 네트워크 뿐만 아니라 모든 네트워크 구조에 사용될 수 있는 알고리즘을 개발 하는 것이다.

목적 IP 주소	출발 라우터	목적 IP 주소	출발 라우터
192.168.1.1	R1	192.168.1.1	R1
192.168.1.2	R1	192.168.1.2	R1
192.168.1.3	R1	192.168.1.3	R1
192.168.1.4	R1	192.168.1.4	R1
192.168.1.5	R1	192.168.1.5	R1
192.168.1.6	R1	192.168.1.6	R1
192.168.1.7	R1	192.168.1.7	R1
192.168.1.8	R1	192.168.1.8	R1
192.168.1.9	R1	192.168.1.9	R1
192.168.1.10	R1	192.168.1.10	R1
192.168.1.11	R1	192.168.1.11	R1
192.168.1.12	R1	192.168.1.12	R1
192.168.1.13	R1	192.168.1.13	R1
192.168.1.14	R1	192.168.1.14	R1
192.168.1.15	R1	192.168.1.15	R1
192.168.1.16	R1	192.168.1.16	R1
192.168.1.17	R1	192.168.1.17	R1
192.168.1.18	R1	192.168.1.18	R1
192.168.1.19	R1	192.168.1.19	R1
192.168.1.20	R1	192.168.1.20	R1
192.168.1.21	R1	192.168.1.21	R1
192.168.1.22	R1	192.168.1.22	R1
192.168.1.23	R1	192.168.1.23	R1
192.168.1.24	R1	192.168.1.24	R1
192.168.1.25	R1	192.168.1.25	R1
192.168.1.26	R1	192.168.1.26	R1
192.168.1.27	R1	192.168.1.27	R1
192.168.1.28	R1	192.168.1.28	R1
192.168.1.29	R1	192.168.1.29	R1
192.168.1.30	R1	192.168.1.30	R1
192.168.1.31	R1	192.168.1.31	R1
192.168.1.32	R1	192.168.1.32	R1
192.168.1.33	R1	192.168.1.33	R1
192.168.1.34	R1	192.168.1.34	R1
192.168.1.35	R1	192.168.1.35	R1
192.168.1.36	R1	192.168.1.36	R1
192.168.1.37	R1	192.168.1.37	R1
192.168.1.38	R1	192.168.1.38	R1
192.168.1.39	R1	192.168.1.39	R1
192.168.1.40	R1	192.168.1.40	R1
192.168.1.41	R1	192.168.1.41	R1
192.168.1.42	R1	192.168.1.42	R1
192.168.1.43	R1	192.168.1.43	R1
192.168.1.44	R1	192.168.1.44	R1
192.168.1.45	R1	192.168.1.45	R1
192.168.1.46	R1	192.168.1.46	R1
192.168.1.47	R1	192.168.1.47	R1
192.168.1.48	R1	192.168.1.48	R1
192.168.1.49	R1	192.168.1.49	R1
192.168.1.50	R1	192.168.1.50	R1
192.168.1.51	R1	192.168.1.51	R1
192.168.1.52	R1	192.168.1.52	R1
192.168.1.53	R1	192.168.1.53	R1
192.168.1.54	R1	192.168.1.54	R1
192.168.1.55	R1	192.168.1.55	R1
192.168.1.56	R1	192.168.1.56	R1
192.168.1.57	R1	192.168.1.57	R1
192.168.1.58	R1	192.168.1.58	R1
192.168.1.59	R1	192.168.1.59	R1
192.168.1.60	R1	192.168.1.60	R1
192.168.1.61	R1	192.168.1.61	R1
192.168.1.62	R1	192.168.1.62	R1
192.168.1.63	R1	192.168.1.63	R1
192.168.1.64	R1	192.168.1.64	R1
192.168.1.65	R1	192.168.1.65	R1
192.168.1.66	R1	192.168.1.66	R1
192.168.1.67	R1	192.168.1.67	R1
192.168.1.68	R1	192.168.1.68	R1
192.168.1.69	R1	192.168.1.69	R1
192.168.1.70	R1	192.168.1.70	R1
192.168.1.71	R1	192.168.1.71	R1
192.168.1.72	R1	192.168.1.72	R1
192.168.1.73	R1	192.168.1.73	R1
192.168.1.74	R1	192.168.1.74	R1
192.168.1.75	R1	192.168.1.75	R1
192.168.1.76	R1	192.168.1.76	R1
192.168.1.77	R1	192.168.1.77	R1
192.168.1.78	R1	192.168.1.78	R1
192.168.1.79	R1	192.168.1.79	R1
192.168.1.80	R1	192.168.1.80	R1
192.168.1.81	R1	192.168.1.81	R1
192.168.1.82	R1	192.168.1.82	R1
192.168.1.83	R1	192.168.1.83	R1
192.168.1.84	R1	192.168.1.84	R1
192.168.1.85	R1	192.168.1.85	R1
192.168.1.86	R1	192.168.1.86	R1
192.168.1.87	R1	192.168.1.87	R1
192.168.1.88	R1	192.168.1.88	R1
192.168.1.89	R1	192.168.1.89	R1
192.168.1.90	R1	192.168.1.90	R1
192.168.1.91	R1	192.168.1.91	R1
192.168.1.92	R1	192.168.1.92	R1
192.168.1.93	R1	192.168.1.93	R1
192.168.1.94	R1	192.168.1.94	R1
192.168.1.95	R1	192.168.1.95	R1
192.168.1.96	R1	192.168.1.96	R1
192.168.1.97	R1	192.168.1.97	R1
192.168.1.98	R1	192.168.1.98	R1
192.168.1.99	R1	192.168.1.99	R1
192.168.1.100	R1	192.168.1.100	R1

그림 10. 라우팅 테이블과 우회 검출 결과  
Fig. 10 Routing table and detected detour routes

### 참고 문헌

- [1] John Blommers, "Practical Planning for Network Growth", Prentice Hall PTR, 1996
- [2] Priscilla Oppenheimer. "Top-Down Network Design" Macmillan Technical Publishing, 1999
- [3] Jeff Doyle "Routing TCP/IP, Volume I" Cisco press 1998 - routing
- [4] M. Rose and K. McCloghrie. "Management Information Base for Network Management of TCP/IP-based internets: MIB-II" RFC 1213, March 1991

### 저자 소개



#### 김진천(Jin-Chun Kim)

1983년 2월 한양대학교 전기공학과(공학사)

1985년 8월 미국 미시간주립대학교 전자 및 시스템공학과(공학석사)

1996년 2월 한국과학기술원 전산학과(공학박사)

1988년 4월~1996년 2월 삼성종합기술원 선임연구원

1996년 3월~현재 경성대학교 전기전자·컴퓨터공학부 부교수

※ 관심분야 : 멀티미디어 통신, ATM 스위치 구조, 컴퓨터 구조, 초고속 네트워크