

Discontiguous Network에서 라우팅 축약 알고리즘의 효율화에 대한 방법론

황성규*

Methodology for the efficiency of routing summary algorithms in discontiguous networks

Seong-kyu Hwang*

*Assistant Professor, Department of Information & Commnincation Engg., Chosun College University of Science & Technology, Gwangju 61453, Korea

요 약

본 논문에서는 불연속 네트워크(discontiguous network)의 라우팅 축약기능(summary) 알고리즘에 대한 방식의 효율화에 대해 고찰을 한다. 서로 다른 전체의 서브넷 정보를 업데이트하여 전송하는 것보다 네트워크 정보를 축약하여 축약된 업데이트 정보만 전송하면 라우팅 테이블의 축약으로 라우터의 자원의 효율화가 이루어 지며 네트워크 안정과 성능을 향상시킬 수 있다. 그러나 네트워크 설계과정에서 불연속적인 네트워크가 구성될 경우 네트워크 축약 기능으로 문제가 발생되며 근본적인 라우터의 효율화의 결과를 가져오지 못한다.

본 논문에서 제안하는 알고리즘은 자동 축약 알고리즘의 단점을 보완한 알고리즘으로, 이로 인해 라우팅 테이블의 안정성을 높이고 그 결과 네트워크 장비의 CPU Utilization을 기존 16.5%에서 6.5%까지 낮추어 효율화가 구현됨을 확인하였다.

ABSTRACT

In this paper, we consider the efficiency of the scheme for for routing summary algorithms in discontiguous networks. Router than updating and transmitting the entire subnet information in the routing protocol, only the shortened update information is sent and the routing table is shortened to make the router resources more efficient and improve network stability and performance. However, if a discontiguous network is formed in the network design process, a problem arises due to the network contraction function and does not bring about the result of fundamental router efficiency. Using different major networks subnets one major network, causing problems in communication and routing information exchange if the configuration is incorrect.

The algorithm proposed in this paper removes only the auto-summary algorithm from the existing algorithm, which increases the complexity and stability of the routing table and reduces the CPU utilization of network equipment from 16.5% to 6.5% Confirmed.

키워드 : 동적 라우팅 프로토콜, 라우팅 축약 알고리즘, 불연속적인 네트워크, 자동 축약 알고리즘

Keywords : dynamic routing protocol, routing summary algorithm, discontiguous network, auto-summary algorithm

Received 27 November 2019, Revised 27 November 2019, Accepted 7 December 2019

* Corresponding Author Seong-kyu Hwang (E-mail:okhsk@cst.ac.kr Tel:+82-62-230-8840)

Assistant Professor, Dept. of Information & Commnincation Engg., Chosun College University of Science & Technology, Gwangju 61453, Korea

Open Access <http://doi.org/10.6109/jkiice.2019.23.12.1720>

print ISSN: 2234-4772 online ISSN: 2288-4165

© This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License(<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.
Copyright © The Korea Institute of Information and Communication Engineering.

I. 서론

라우트 축약기능은 주소들의 연속적인(contiguous) 집합을 보다 더 구체적이고 더 짧은 서브넷 마스크를 사용해서 하나의 주소로 어드버타이즈(advertise)하는 프로세스이다. 클래스풀 경계들이 제한을 무시하고 디폴트 클래스풀 마스크보다 더 적은 마스크들을 사용해서 요약한다. 이런 타입의 축약은 라우팅 업데이트와 로컬 라우팅 테이블에 있는 엔트리들이 수를 감소시킨다. 또한 라우팅 업데이트들에 대한 대역폭 이용률을 감소시켜 주고 더 빠른 라우팅 테이블 lookup을 가능하게 한다[1].

1.0.0.0/24 is subnetted, 3 subnets 1.1.1.0 [90/20640000] via 172.16.1.1, 00:00:08, Serial0/0 1.1.2.0 [90/20640000] via 172.16.1.1, 00:00:08, Serial0/0 1.1.3.0 [90/20640000] via 172.16.1.1, 00:00:08, Serial0/0 172.16.0.0/30 is subnetted, 1 subnets 172.16.1.0 is directly connected, Serial0/0

Fig. 1 Routing Table

그림 1 라우팅 테이블의 정보는 상세히 기록되고 있으며 상세한 라우팅 정보가 많을수록 패킷 스위칭 속도가 느려지기 때문에 적절한 크기로 축약 시켜주는 것이 네트워크 축약기능이다.

그런데 서로 다른 메이저 네트워크를 사용하면 통신 및 라우팅 정보 교환에 문제는 없지만 하나의 메이저 네트워크를 서브넷화하여 구성이 잘못된 경우에는 서브넷 정보를 전달하지 않으므로 통신 및 라우팅 정보 교환에 문제가 발생하게 된다[2]. 본 논문에서는 이러한 문제를 해결할 수 있는 기존의 방법으로 static route를 사용하는 방법과 RIPv1 같은 경우는 VLSM을 지원하는 RIP version2를 적용하는 방법 등이 있는데 이러한 기존 방식보다 제안한 방식이 효율적이고 성능 향상이 됨을 확인하였다. 그리고 최단 경로 우선 프로토콜(OSPF)는 RIP와 관련된 많은 문제를 해결했으며 네트워크 상태에 따라 경로 선택을 동적으로 선택할 수 있으나 복잡성 때문에 정말로 필요할 때 이외에는 잘 사용하지 않아 본 논문에서는 다루지 않는다.

본 논문의 구성은 2장에서는 네트워크 축약과 경로 계산 알고리즘의 사례에 대해 살펴보고 3장에서는 고찰한 내용을 기반으로 효율적인 방식의 알고리즘을 제안하고 4장에서는 제안한 알고리즘을 시뮬레이션을 통하

여 성능을 분석하고, 5장에서 본 논문의 결론에 관하여 기술한다.

II. 관련연구

2.1. 자동 축약(auto-summary)

디스턴스 벡터(distance vector) 라우팅 프로토콜의 특징 중 하나인 자동 축약은 서브넷팅 하지 않았을 때의 주 네트워크(major network)를 의미 하며 주 네트워크 경계(boundary)에서는 주 네트워크(boundary)만 어드버타이즈(advertise) 한다.

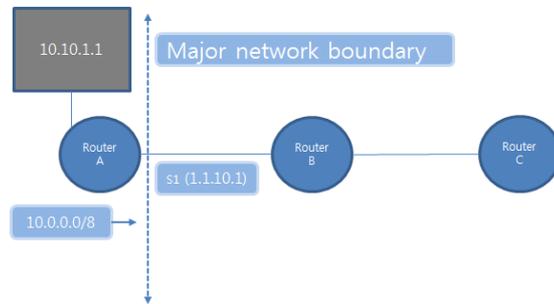


Fig. 2 Major network boundary

그림 2의 Router A에 포함된 네트워크 중 하나의 IP 10.10.1.1을 디스턴스 벡터 라우팅 프로토콜로 어드버타이즈 하면 주 네트워크는 10.0.0.0/8 이다. Router A의 라우팅 정보에 포함된 네트워크 10.10.1.0과 시리얼 인터페이스(serial interface) 네트워크와 1.1.10.0가 서로 다르며 이처럼 주 네트워크가 다른 S1 시리얼 인터페이스를 주 네트워크 경계라 한다. 그림 1과 같이 라우팅 정보에 포함된 네트워크가 주 네트워크 영역에 이르면 주 네트워크 주소만 전송하는 것은 자동 축약이다.

2.1.1. 연속적(contiguous)으로 구성된 네트워크

동적 라우팅 프로토콜 중에서 RIP과 EIGRP는 주 네트워크 경계(major network boundary)에서 자동 축약(auto summary)이 이루어진다[3]. 그림 3의 Router A의 내부망 네트워크가 50.50.16.0/24에서 50.50.23.0/24까지의 8개의 네트워크가 라우팅 프로토콜에 의해 어드버타이즈 되어야 한다. Router A의 시리얼 구간에서는 네트워크 주소가 10.10.123.1이므로 어드버타이즈 되어야

할 네트워크 주소와 시리얼 지점의 네트워크 주소가 다르기 때문에 그 지점이 주 네트워크 경계가 된다. Router A에서는 연속된 내부 네트워크를 전송할 때에는 자동축약 알고리즘에 의해 주 네트워크인 50.0.0.0 정보만 Router B에게 전송되고 다른 네트워크에게도 주 네트워크만 전송한다. 그림 2와 같이 연속된 네트워크일 경우 효과적인 자동축약 알고리즘에 의해 네트워크의 안정성과 성능을 향상시킬 수 있다[4].

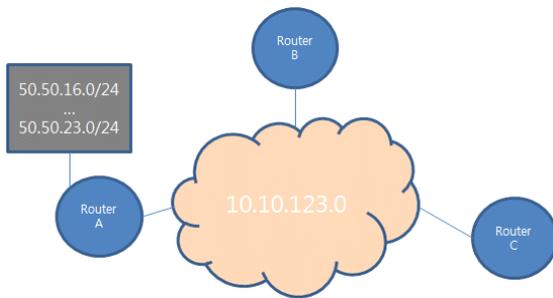


Fig. 3 Network diagram requiring automatic reduction

2.1.2. 불연속적(discontiguous)으로 구성된 네트워크 불연속적 네트워크는 그림 3과 같이 네트워크 50.50.0.0 가 같은 서브넷이 다른 주 네트워크에 의해서 분리되어 있는 구성일 경우는 클래스풀 라우팅 프로토콜을 사용하면 같은 주네트워크가 2개의 네트워크로 상호 단절된다. 네트워크 설계가 잘못된 경우는 원인을 찾아 다시 설계하여 해결하거나 클래스리스 라우팅 프로토콜을 사용하면 해결점을 찾을 수 있으나 원인을 분석하고 해결하기 위해서는 많은 시간이 필요할 것이고 사용자에게 필요한 서비스를 제공하기 힘들 것이다[5].

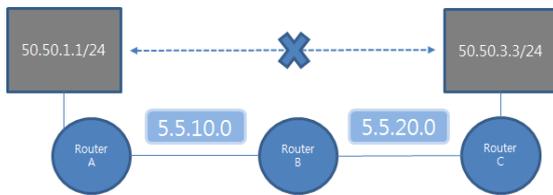


Fig. 4 Discontinuous Network Diagram

그림 4의 Router A와 Router B는 각각의 서브넷들을 요약했기 때문에 각각의 라우팅 테이블은 50.0.0.0의 메이저 클래스풀 네트워크 주소만 포함하게 되고 /8의 클래스 서브넷 마스크를 첨가하게 되어 네트워크가 단절되게 된다. 그 이유는 각 라우터의 인터페이스들에서는

50.0.0.0 네트워크를 학습하였기 때문에 동일한 인터페이스로 전송되는 업데이트 안에 그 네트워크를 포함시키지 않기 때문이다.

2.2. 경로 계산 알고리즘

2.2.1. 다익스트라 알고리즘(Dijkstra algorithm)

다익스트라 알고리즘의 특징은 네트워크 라우팅에서 최소비용의 경로를 빠르고 정확하게 찾아 라우팅 테이블을 갱신하여 패킷을 전달 할 수 있다[6]. 그런데 다익스트라 알고리즘을 적용한 네트워크에서 변화가 발생되면 최소 비용의 경로를 다시 갱신하기 위해서 다시 다익스트라 알고리즘을 수행한다. 링크 비용이 전혀 수정되지 않는 노드에 대해서는 반복 수행하여 계산한다. 그러므로 계산의 절차가 많아지고 라우팅 테이블의 수정이 발생되며 라우팅 테이블이 불안정하게 될 수 있다[7].

다익스트라 알고리즘의 주요 내용은 루트(root) 노드에서 모든 목적지까지 최단거리를 찾는 것이며 네트워크 토폴로지에 변화가 있을 때는 라우팅 테이블에 있는 최단 경로 정보를 참조하지 않고 다시 최단경로를 계산한다. 다익스트라 알고리즘은 라우팅테이블에 갱신할 최단경로의 계산은 라우터의 자원인 CPU의 부하를 가중시키는 큰 문제가 발생한다.

2.2.2. 벨만-포드 알고리즘(Bellman-Ford algorithm)

벨만-포드 알고리즘은 모든 네트워크를 통과하는 비용은 같다는 방향 그래프 이론의 정의에서 출발하며 음의 가중치를 갖는 그래프에서 SSP(Single Source Shortest Path)를 찾는 것이 목적이다[8]. 반면에 다익스트라 알고리즘은 음의 가중치를 갖지 않는 방향 그래프에서 최단 경로를 계산하는 알고리즘으로 차이가 있다. 벨만-포드 알고리즘은 정확성을 보장하기 위해 비효율적인 알고리즘을 수행한다. 모든 연결된 네트워크의 노드만큼 알고리즘 계산 작업을 수행한다. 그러므로 시간 복잡도에서 많은 시간이 소요된다.

벨만-포드 알고리즘은 노드수 만큼 반복문을 수행하면서 라우팅 테이블을 작성한다. 음의 가중치를 갖는 순환 경로가 존재하는지 확인하며 존재하면 false를 반환하고 존재하지 않으면 알고리즘이 정상 수행했다는 true를 반환한다. 벨만-포드 알고리즘은 모든 노드의 수만큼 수행하기 때문에 많은 연산량이 필요하다[9].

또한 네트워크 토폴로지가 불연속적인 경우는 다시

계산하지 않고 불연속 네트워크를 반영하지 못하기 때문에 문제가 발생하고 패킷 손실이 된다. 이런 불연속 네트워크는 네트워크 주소가 연속적이지 못해서 발생하는 경우로 언제든지 발생할 수 있으며 매우 치명적인 경우라 할 수 있다[10].

그래서 이런 경우를 발생하지 않게 설계하여야 패킷 손실을 방지 할 수 있으며 이런 경우는 계산시간이 많이 걸리더라도 네트워크 축약기능이 없는 경우가 더 효과적이다. 다시 계산하는 것은 비효율적이지만 안정적인 네트워크 구성을 위해서는 네트워크 축약기능은 없는 것이 안정적인 측면에서는 더 좋다.

III. 제안하는 알고리즘

라우팅 테이블에 적은 수의 라우트를 갖는다는 것은 라우팅 테이블 프로세스가 패킷을 전송하는데 필요한 라우트를 더 빠르게 찾을 수 있는 장점이 있다. 즉 각각 다른 서브넷마다 업데이트를 보내는 대신에 전체 클래스플 네트워크에 대하여 오로지 하나의 업데이트만을 보내기 때문이다. 그래서 동적 라우팅 프로토콜에서 네트워크 축약기능의 역할은 중요하다. 효과적인 라우팅 기능은 축약을 통해 네트워크의 안정성과 성능을 향상시킬 수 있기 때문이다. 축약이 제대로 구현되지 않으면 속도와 장비의 성능이 좋더라도 네트워크가 불안정해진다.

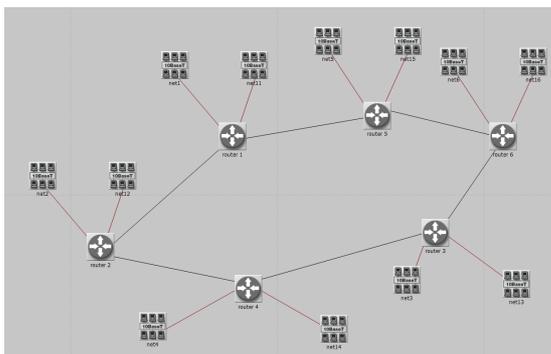


Fig. 5 Network diagram

불연속적인 네트워크(discontiguous network)에서는 네트워크 축약기능에 문제가 발생하게 된다. 클래스플 라우팅 프로토콜은 라우팅 업데이트에 서브넷 마스크

를 포함하지 않고 메이저 네트워크 경계를 넘을 때 자동 요약되기 때문이다. 불연속적인 네트워크는 요약 기능을 해제하고 서브넷을 네트워크 경계 이후의 네트워크에 어드버타이즈하는 기능을 수행 명령어로 제어할 수 있지만 근본적인 라우터의 효율화 결과를 가져오지 못한다.

그림 5의 네트워크 구성도와 같이 라우터 6개로 구성된 네트워크에서 자동 축약 알고리즘을 사용함으로써 개선된 효과도 있지만 불연속 네트워크등과 같은 설계나 구성으로 이로 인한 네트워크와의 단절등 라우팅 테이블 갱신에 지대한 문제를 야기시킬 수 있다. 제안하는 알고리즘은 기존의 알고리즘에서 자동 축약 알고리즘을 개선한 알고리즘으로, 이로 인해 라우팅 테이블의 복잡성과 안정성을 높이고 그 결과 네트워크 장비의 CPU Utilization 까지 효율적으로 사용할 수 있다.

IV. 성능 평가

동적 라우팅 프로토콜에서 네트워크 자동 축약으로 인한 장점들의 역할이 중요했다. 그러나 불연속적인 네트워크에서는 문제가 발생된다. 시뮬레이션 환경은 그림 5의 네트워크로 구성하여 표 1의 실험 parameter로 설정하여 시뮬레이션은 OPNET을 사용하여 12개의 네트워크를 6개의 라우터로 랜덤하게 각 노드에서 전송하는 환경으로 시뮬레이션을 수행하였다.

Table. 1 Performance Evaluation Experiment Environment Parameter

Address Family	IPv4 -Any
Process Parameters (Start Time)	uniform(5,10)
Update Interval(seconds)	30
Route Invalid(seconds)	180
Flush(seconds)	240
Holddown(seconds)	180
Failure Impact	Tetain Route Table
Admiistrative Weight	120
Process Tag	1

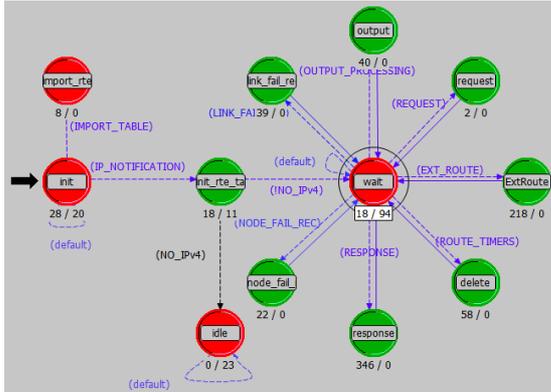


Fig. 6 Routing Behavior Mechanism

그림 6은 본 논문에서 라우팅 테이블 작성하는 라우팅 동작 메커니즘이며 절차는 입력된 IP정보를 import 하여 IPv4로 구성 되어 있는지 check하고 IPv4의 정보만 받아들인다. 그리고 wait 상태로 들어간다. node fail 때는 request 메시지를 전달하여 테이블 갱신과 response 하고 route_timers 상태일 때는 라우팅 테이블에서 delete가 진행되며 이러한 절차에 의해 생성된 정보를 라우팅 테이블에 갱신하는 동작 메커니즘이다.

이러한 구성에서 시뮬레이션한 OPNET 결과로 그림 7의 y축은 라우터의 cpu utilization을 나타내며 x축은 시간의 경과를 나타내고 있다. 파란색 그래프는 자동 축약 기능이 있는 기존 방식 알고리즘을 나타내고 있으며 빨간색 그래프는 기존 방식의 자동 축약 알고리즘을 개선한 제안하는 알고리즘을 적용한 방식의 알고리즘이다. 각각의 라우터 중 하나의 라우터 시간의 경과함에 따라 라우터의 cpu utilization을 나타내고 있으며 제안하는 방식이 6.5% 이고 기존 방식이 16.5%로 라우터의 cpu utilization에서 더 효율적임을 알 수 있다.

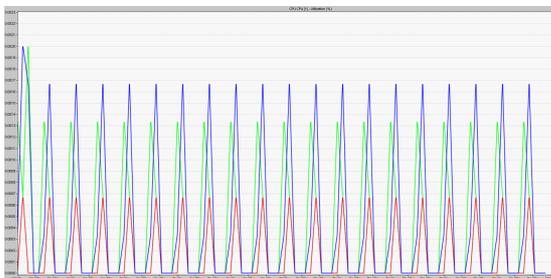


Fig. 7 cpu utilization (%)

Table. 2 Traffic Received(bit/sec)

time (sec)	0.1	6	12	30	36	60	66	90
Existing Algorithm	277.3	2282	938.7	864	2592	864	2592	864
Proposal Algorithm	0	0	0	0	0	0	0	0

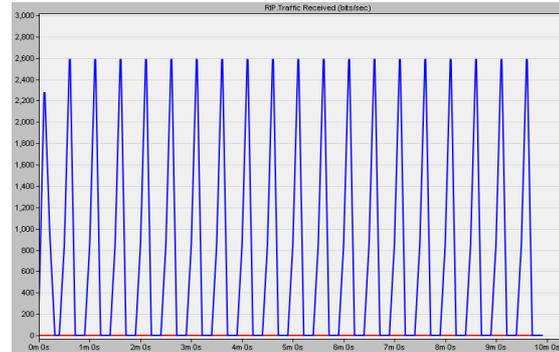


Fig. 8 Traffic Received(bit/sec)

표 2와 그림 8은 네트워크 축약 Traffic Received를 초당 비트수로 표시하여 기존 알고리즘과 제안 알고리즘을 비교 하였으며 기존 알고리즘은 시간당 Traffic 발생되고 제안 알고리즘은 네트워크 축약 Received Traffic이 발생하지 않고 있다. 네트워크 축약 Traffic이 cpu utilization에 영향이 있음을 알 수 있었다.

V. 결론

본 논문에서는 동적 라우팅 프로토콜에서의 네트워크 축약기능의 알고리즘에 대한 방식의 효율에 대해 고찰하였다. 동적 라우팅 프로토콜에서 네트워크 축약기능의 역할은 중요하며 효과적인 라우팅기능은 축약을 통해 네트워크의 안정성과 성능을 향상시킬 수 있다. 축약이 제대로 구현되지 않으면 속도와 장비의 성능이 좋더라도 네트워크가 불안정해진다. 연속적이지 못한 불연속적인 네트워크에서는 네트워크 축약기능에 문제가 발생하게 된다. 본 논문에서는 동적 라우팅 프로토콜에서의 불연속 네트워크에서의 자동 축약 기능으로 라우팅 테이블 갱신 문제 등으로 네트워크 단절하는 지대한 문제를 야기 시켜 기존 자동 축약 알고리즘을 개선한 알고리즘을 적용하여 기존 문제를 해결하고 시뮬레이션

을 통해 라우터 자원의 cpu utilization이 16.5%에서 6.5%로 변화하여 효율화 성능을 검증하였다.

REFERENCES

- [1] W.Stalling, "High-Speed Networks: TCP/IP and ATM Design Principles,"Prentice Hall, Inc., 1998.
- [2] G. Malkin, "RIP Version 2," *Request for Comments 2453*, Now. 1998.
- [3] B. Xiao, J. Cao, Z. Shao, and E.H.m. sha, "An Efficient Algorithm for Dynamic Shortest Path Tree Update in Network Routing," *Journal of Communication and Networks*, vol.9, no.4, pp.499-510, Dec.2017.
- [4] E.P.F. Chan, and Y.Yang, "Shortest Path Tree Computation in Dynamic Graphs," *IEEE Trans. on computers*, vol.58, no.4, Apr. 2015.
- [5] T.W Kwon,"Improvement of IS-IS Protocol for TICN Tactical Backbone," *Korea Institute OF Communication Sciences*, The Kor. In. Comm. Sci., vol.36 no.8, pp.996-1002, Aug. 2016.
- [6] E. Dijkstra, "A noet two problems in connection with graphs," *Numerical Math*, vol.1, pp.269-271, 1959.
- [7] J. Moy, "OSPF Version 2," *Request for Comments 2328*, Apr. 1998
- [8] R. Bellman, "On a routing problem," *Quarterly Appl. Math*, vol.16, pp.87-90,1958.
- [9] S. H. Lee, and S.H. Rhee, "Efficient Flooding for Reliability in Link-state Routing Protocols," *ICTC 2015*.
- [10] V. Eramo, M. Listanti, and A. Cianfrani, "Design and Evaluation of a New Multi-Paht Incremental Routing Algorithm on Software Routers," *IEEE Transactions on Network. and service management*, vol. 5, no.4, Dec.2016.



황성규 (Seong-kyu Hwang)

정보통신공학과 공학박사

現 조선이공대학교 정보통신과 조교수

※관심분야 : 통신 프로토콜, 네트워크 설계, 통신 네트워크