

노드 중심성을 이용한 효율적 네트워크 토폴로지 시각화 연구*

장 범 환*, 류 제 민**, 권 구 형***

요 약

그래프 시각화 이론에 근간을 둔 네트워크 토폴로지 시각화는 복잡한 네트워크의 전체 구조와 노드간의 상호작용을 보다 이해하기 쉽게 만든다. 네트워크 토폴로지를 시각화하는 도구는 과거부터 많이 개발되었지만, 일정 수준의 기능을 갖춘 도구들은 도구마다 고유한 네트워크 구성 정보(노드의 식별자, 종류, 속성, 연결된 노드 등)를 입력으로 요구하기 때문에 범용적으로 사용하기 어렵다. 반면에 최소한의 네트워크 구성 정보인 노드간의 연결만을 사용하는 도구들은 네트워크의 실제 연결 형태를 표시하는 기능이 부족하다. 본 논문에서는 네트워크 노드간의 연결 정보만을 이용하여 토폴로지를 시각화하는 효율적인 방법을 제안한다. 이 방법은 네트워크에서 노드의 영향력을 나타내는 중심성 지수를 활용하여 중심노드를 찾고, 자식노드의 가중치를 이용하여 전체 노드들의 표시 영역을 동적 분할한 후 3D 공간 상에 노드들을 배치함으로써 토폴로지를 시각화한다. 매우 간단한 방법이지만 노드간의 연결 정보만으로 실제 네트워크 연결 형태를 시각화할 수 있다.

A Study on Efficient Network Topology Visualization using Node Centrality

Beom-Hwan Chang*, Jemin Ryu**, Koohyung Kwon***

ABSTRACT

Network topology visualization has been studied a lot since the past and developed with many tools. The network topology has strength in understanding the overall structure of a network physically and is useful for understanding data flow between nodes logically. Although there are existing tools, not many can be utilized efficiently while using the general network node data structure and express the topology similar to the actual network structure. In this paper, we propose an efficient method to visualize topology using only connection information of network nodes. The method finds the central node by using the centrality, the influence of nodes in the network, and visualizes the topology by dynamically segmenting all nodes and placing network nodes in 3D space using the weight of the child node. It is a straightforward method, yet it effectively visualizes in the form of an actual network structure.

Key words : Network Topology, Topology Visualization, Node Centrality Visualization

접수일(2021년 06월 14일), 수정일(2021년 06월 18일),
게재확정일(2021년 06월 25일)

★ 본 논문은 국방과학연구소(계약번호 UD200024ED)의 연구
비 지원에 의하여 연구되었음.

* 호원대학교/IT소프트웨어보안학과 (주저자)

** (주)에이알씨엔에스 (공동저자)

*** 국방과학연구소 (교신저자)

1. 서 론

네트워크 토폴로지(topology, 망구성 방식)는 컴퓨터 네트워크의 요소들(노드, 엣지 등)을 물리적으로 연결해 놓은 것, 또는 그 연결 방식을 말한다. 네트워크를 이해하기 위한 대규모 연결망 구조의 시각화는 수년 동안 많은 프로젝트에서 진행되어 왔다. 연결망 시각화는 인간의 지각 능력을 활용하여 네트워크 형상과 연결 관계로부터 네트워크의 특징을 찾고 구조를 이해하는데 유용한 방법이다[1][2]. 네트워크 형상을 실제와 같은 모습으로 표시하는 물리적 토폴로지 시각화는 네트워크의 전체 구조를 이해하는데 유리하고, 네트워크 구조 보다는 네트워크 요소들의 의미(중요도, 종류, 속성 등)를 부각하여 표시하는 논리적 토폴로지는 노드의 행위, 노드간의 데이터 흐름 등 상호작용을 이해하는데 유용하다[3][4].

토폴로지 시각화는 네트워크 구조를 점과 선으로 연결하여 그래프 형태로 표시하는 방법이다. 구조적인 측면에서 연결망 그래프는 그 자체로 분석적 의미를 표현할 뿐만 아니라, 그래프로 시각화된 패턴들은 연구자들로 하여금 더 많은 해석 가능성도 열어준다. 네트워크 구조를 알지 못하며 망구성 방식이 복잡하더라도 토폴로지를 시각화함으로써 네트워크 내의 어떤 노드가 중요한 허브(hub) 노드이고, 노드 간 연결성이 떨어지는 지점은 어디인지, 병목 예상 구간은 어디인지를 파악할 수 있다[5]. 망구성 방식의 시각화는 복잡한 네트워크의 전체 구조를 파악하고 네트워크 체계를 구성하는 노드들의 행위, 연결 관계, 상호작용 등 네트워크 특성을 쉽게 이해할 수 있는 도구이다.

본 논문에서는 네트워크를 구성하는 노드들의 연결 정보만을 이용하여 토폴로지를 시각화하는 효율적인 방법을 제안한다. 이 방법은 자식노드의 가중치를 이용하여 전체 노드들을 표시 영역을 동적 분할하고 3D 공간 상에 노드들을 배치하여 토폴로지를 시각화한다. 논문의 구성은 2장에서 토폴로지 시각화 도구 및 관련된 연구들을 살펴보고, 3장에서는 시각화를 위한 구체적인 구현 알고리즘과 방법에 대해 기술한다. 4장은 시각화 인터페이스를 설계하고 설계된 시각화 인터페이스를 통해 실제 네트워크 데이터를 적용하여 구현한다. 5장에서는 실제 활용 측면에서 유효성이 있는 지

를 검증하고 6장에서는 결론 및 향후 계획에 대해 설명한다.

2. 관련 연구

네트워크 토폴로지 시각화 기술에 대한 포괄적인 조사와 연구는 CAIDA에서 과거부터 꾸준히 발전해 오고 있다. CAIDA는 1998년부터 인터넷의 거시적 토폴로지를 측정하기 위해 다양한 시각화 도구들을 개발했으며, 최초의 Skitter부터 현재 Scamper까지 일련의 장비에서 수 백만개의 프로브 패킷을 전송하여 글로벌 IP 레벨의 연결을 추적하고 있다. 수집된 데이터는 토폴로지와 트래픽 측정을 위해 다양하게 이용된다[6].

전통적인 토폴로지 시각화 방법은 그래프를 이용하는 것이다. 그래프 시각화는 네트워크 구조를 자연스럽게 모델링할 수 있으며, 최근 사회연결망(Social Network) 분석에도 많이 활용되고 있다. 사회연결망에서는 주로 노드의 영향력(상대적 중요성)을 측정할 수 있는 중심성(centrality)들을 활용하여 노드 배치를 결정한다[7]. 그래프 시각화 기능을 제공하는 도구로는 Tulip, Graphviz, Gephi, Pajek 등이 있다[8].

2.1 CAIDA의 네트워크 토폴로지 시각화 도구

Skitter[9]는 인터넷의 토폴로지와 성능을 분석하기 위해 능동적으로 토폴로지를 검색하는 도구이다. Skitter는 거시적 토폴로지 측정 프로젝트의 일환으로 인터넷의 토폴로지 맵 정보를 생성하는데 활용되었으며, 이후 스키어 인프라(Ark)로 대체되었다. 인터넷의 토폴로지 맵은 중요한 인프라의 특성, 동적 동작 및 진화를 이해하는데 중요한 도구로 활용되고 있으며 인터넷 인프라 및 기타 대규모의 복잡한 네트워크에 대한 현실적인 모델링, 시뮬레이션 및 분석에 이용된다.

Plankton[10]은 글로벌 루트 캐시 관점에서 NLNR(National Laboratory for Applied Network Research)의 웹 캐시 토폴로지를 시각화한다. 미국의 국립과학재단(NSF: National Science Foundation)은 전 세계 웹 캐싱 계층 구조의 프로토타입을 개발 및 배포하고 있으며 인터넷 트래픽 증가로 인한 일부 문

제를 완화하기 위해 NLANR을 후원하고 있다.

Otter[11]는 CAIDA의 대표적인 토폴로지 시각화 도구로써 노드-엣지 또는 경로 집합으로 표현되는 임의의 네트워크 데이터를 시각화한다. Otter는 토폴로지, 작업량, 성능 및 라우팅에 대한 데이터 세트를 포함하여 다양한 인터넷 데이터에 대한 시각화 작업을 처리할 수 있는 도구이지만, 현재는 CAIDA에 의해 유지·관리만 되고 있는 상태이다. Otter의 강점은 데이터의 독립성이다. 엣지와 노드로 구성된 모든 형식의 데이터 세트를 처리할 수 있다.

Walrus(H3Viewer)[12]는 3차원 공간 상에서 스페닝 트리를 기반으로 노드들의 레이아웃을 계산한다. 어안(fisheye)과 같은 왜곡으로 하이퍼볼릭 지오메트리(hyperbolic geometry)를 사용하여 구(sphere) 상에 토폴로지를 표시한다. 노드들은 구 중심으로부터 방사상 거리에 따라 확대되는데, 중앙 부근의 노드들은 확대되고 외곽의 노드들은 축소된다. 배율은 노드들의 크기가 0으로 줄어들 때(중심에서 경계)까지 지속적으로 감소한다. 사용자는 토폴로지의 다른 부분을 중앙 영역으로 가져오면 확대되므로 토폴로지의 모든 부분을 자세히 검사할 수 있다.

2.2 그래프 시각화 도구

Tulip[13]은 그래프의 계층 구조 또는 클러스터 트리(중첩된 하위 그래프)를 효율적으로 정의하고 탐색할 수 있는 도구이다. 이 기술은 시각적 복잡성을 줄이기 위해 큰 그래프를 계층적 구조를 갖는 하위 그래프들로 클러스터링하고 데이터를 탐색한다. Tulip의 데이터 구조는 처음부터 중첩 하위 그래프를 생성할 수 있도록 설계되었고, 일관성과 최적의 공간 사용을 모두 고려하여 제작되었다.

Graphviz[14]는 다양한 그래프 레이아웃을 제공하는 오픈 소스 그래프 시각화 소프트웨어이다. 가장 널리 사용되는 그래프 시각화 방법 중 하나인 Kamada와 Kawai의 에너지 최적화 기법을 기반으로 다차원 척도(MDS: multidimensional scaling) 커뮤니티에서 스트레스 함수 주요화를 통해 그래프를 시각화한다. 이 방법을 통해 러닝 시간과 안정성면에서 성능을 개선했다. 웹 및 대화형 그래픽 인터페이스와 보조 도구,

라이브러리 및 다양한 프로그래밍 언어를 지원한다.

Gephi[4]는 그래프 및 네트워크 분석을 위한 오픈 소스 소프트웨어로써 그래프를 탐색하고 이해하려는 데이터 분석가에 적합한 도구이다. 3D 렌더링 엔진을 사용하여 대규모 네트워크를 실시간으로 시각화하고 인터랙티브 탐색과 조작 및 네트워크 데이터에 대한 쉽고 광범위한 액세스를 제공한다. 또한 네트워크 탐색 도구에 요구되는 고품질 레이아웃, 데이터 필터링, 클러스터링, 통계 및 주식 기능을 제공한다. 실제로 이러한 기능들이 유연하고 확장 가능하도록 설계되었기 때문에 전문가와 초보자 모두에게 사용자 친화적인 소프트웨어로 각광받고 있다.

Pajek[15]은 대규모 네트워크의 구성 요소들을 6가지(네트워크, 분할, 벡터, 클러스터, 순열, 계층 구조) 데이터 유형으로 분석하고 시각화하는 도구이다. Pajek에서 사용 가능한 네트워크 레이아웃에는 Kamada-Kawai 최적화, Fruchterman Reingold 최적화, VOS 매핑, Pivot MDS, FishEye 변환 등이 있다. 또한, 결과 화면을 다양한 2D 또는 3D 출력 형식(예: SVG, EPS, X3D, VOSViewer, Mage 등)으로 내보낼 수 있어서 외부의 다른 뷰어 및 편집기를 사용하여 레이아웃을 추가로 편집하거나 세부적으로 분석할 수도 있다.

2.3 중심성 지수

토폴로지 시각화는 단적으로 표현 목적에 맞는 중요한 노드를 찾고 2D/3D 공간 상에 노드들을 배치하는 것이다. 노드 중심성을 계산하면 네트워크에서 중요한 노드를 쉽게 찾을 수 있다. 중심성(centrality)은 네트워크에서 노드의 영향력인 상대적 중요성을 나타내는 척도(지수)로써 수치값으로 계산된다. 중심성 지수는 그 계산 방식에 따라 다양한 종류가 있으며, 일반적으로 연결중심성(C_d : degree centrality), 근접중심성(C_c : closeness centrality), 매개중심성(C_b : betweenness centrality), 고유벡터중심성(C_e : eigenvector centrality) 등이 주로 사용된다[7].

연결중심성(C_d)은 인접한 노드들의 연결 정도에 대한 척도로써 하나의 노드가 직접적으로 연결된 노드들의 개수 또는 연결값을 모두 합한 값으로 측정된다.

즉, 많은 연결을 가진 노드일수록 더 중요한 노드라고 측정된다. 고유벡터 중심성(C_e)는 무조건 연결된 노드가 많다고 중요한 노드가 되는 것이 아니라 중요한 노드와 많이 연결된 노드가 더 중요하다는 아이디어에서 착안했다. 예를들면, 페이스북의 친구가 많다고 해서 유명인사인 것이 아니라 유명인사인 친구가 많을수록 더 유명인사라고 할 수 있다. 단순히 연결된 노드의 숫자만 살핀다는 점에서 C_d 의 약점이 있는데 C_e 는 중심성을 계산할때 다른 노드의 중심성을 반영해서 계산하는 방법이다[7][16].

근접중심성(C_c)은 하나의 노드가 다른 모든 노드들에 도달하는 최단거리를 합한 값이다. 여기서, 최단거리는 가능한 모든 연결에 대한 거리 값의 합들 중에서 최소값을 의미한다. 사람들의 네트워크에서 근접중심성이 높은 사람이 감염되거나 소문을 퍼트릴 경우 전체 네트워크에 가장 빠르게 확산된다. 매개중심성(C_b)은 네트워크 상의 모든 두 노드 사이의 최단거리가 해당 노드를 통과하는 정도를 측정·환산한 값이다. 네트워크에서 매개중심성이 높은 노드들의 주변 연결을 차단하면 전체 네트워크로의 확산을 효과적으로 막을 수 있다. 따라서, 근접중심성이 높은 노드는 네트워크를 공격할 때 공격의 시작점으로서 효과적이며, 매개중심성이 높은 노드는 진원지를 알 수 없는 소문이나 전염병을 차단하고자 할 때 방어 지점으로서 효과적이다[7][16].

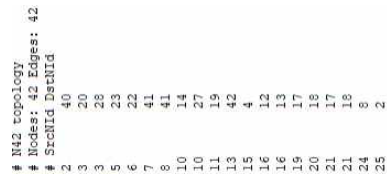
네트워크 토폴로지의 시각화가 단적으로 일정한 공간 상에 노드들을 배치시키고 연결하는 것이라고 할 때, 네트워크 또는 일정한 공간의 기준이 되거나 중심이 되는 노드를 찾고 선택하는 것은 어려운 문제이다. 물론, 관점에 따라 차이가 있겠지만 매개중심성 또는 근접중심성 값이 높은 노드를 선택하면 쉽게 이 문제는 해결할 수 있다.

3. 연구 방법

3.1 토폴로지 데이터 구조 및 중심성 지수 계산

네트워크란 노드(Node)와 엣지(Edge)로 이루어진 추상적인 개념으로써 노드는 하나의 개념이나 실체가

되고 엣지는 노드 사이의 연결 또는 관계를 의미한다. 즉, 네트워크 구조는 노드와 노드들을 연결하는 엣지들로 표현할 수 있는데 엣지가 방향성을 가지는가에 따라서 Directed 엣지와 Undirected 엣지로 구분한다 [7][17]. 본 연구에서는 토폴로지 데이터 구조가 범용성을 갖도록 노드들의 연결정보만을 사용한다. 토폴로지 데이터 구조는 (그림 1)와 같이 노드 식별자(ID) 2개가 쌍으로 이루어진 Undirected 엣지들의 집합이다. (그림 1)은 토폴로지 데이터 구조를 나타낸 것이다.



(그림 1) 토폴로지 데이터 구조

```
#Network: ./node42_edges.txt
#Nodes: 42 Edges: 42
#NodeId Degree Closeness Betweenness EigenVector PageRank HubScore
2 0.250000 0.627240 0.079051 0.001804 0.325997 0.000000
40 0.750000 0.726141 0.691700 0.009251 0.453322 0.002978
3 1.000000 0.770925 0.545455 0.976727 1.000000 0.000000
20 0.250000 0.675676 0.090909 0.763814 0.601251 0.000000
28 0.125000 0.655431 0.000000 0.660650 0.601251 0.000000
5 0.250000 0.627240 0.079051 0.001804 0.253280 0.000000
23 0.125000 0.548589 0.000000 0.001208 0.391504 0.000000
6 0.250000 0.627240 0.079051 0.001804 0.253280 0.000000
22 0.125000 0.548589 0.000000 0.001208 0.391504 0.000000
7 0.250000 0.791855 0.652174 0.004767 0.253280 0.000000
41 0.375000 0.862069 0.841897 0.018571 0.795933 0.000000
8 0.375000 0.732218 0.156126 0.004124 0.475775 0.000000
10 0.375000 0.829384 0.156126 0.129606 0.286620 0.000000
14 0.125000 0.697211 0.000000 0.087663 0.298030 0.000000
27 0.125000 0.697211 0.000000 0.087663 0.298030 0.000000
11 0.250000 0.845411 0.193676 0.117996 0.286620 0.000000
19 0.250000 0.732218 0.136364 0.107244 0.419841 0.000000
13 0.250000 0.977654 0.818182 0.115133 0.251108 0.000000
42 0.500000 1.000000 1.000000 0.584697 0.389659 0.757534
15 0.125000 0.548589 0.000000 0.001208 0.176218 0.000669
```

(그림 2) SNAP을 이용한 중심성 지수 계산

네트워크에서 노드의 상대적 영향력인 중심성은 스탠포드 대학에서 개발한 네트워크 분석 라이브러리 SNAP(Stanford Network Analysis Project)[18]을 이용하여 계산한다. SNAP을 활용하여 계산한 노드들의 중심성 지수는 (그림 2)와 같다.

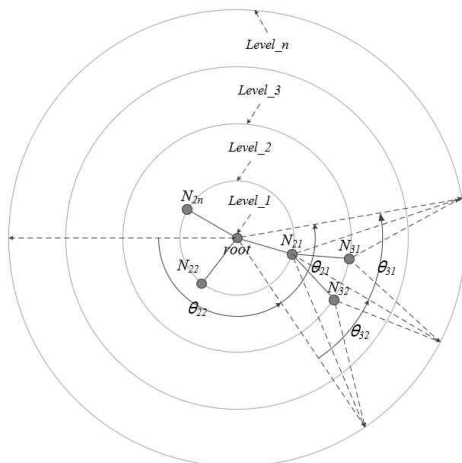
3.2 연결-우선 동심원 확장 배치 구조

토폴로지 시각화를 위한 노드 배치 방법은 (그림 3)과 같이 Level별 반지름 길이가 다른 다수의 동심원들을 이용하여 최심부에서부터 연결정보를 탐색하며 외곽으로 확장하는 구조로 설계한다. 중심성 지수가 가

장 높은 노드를 네트워크의 중심인 root노드로 설정하고 연결된 자식 노드들을 탐색하며 동심원의 Level을 지정한다. (그림 3)에서 root노드에 연결된 자식노드들이 $N_{21}, N_{22}, \dots, N_{2n}$ 이라고 할 때, N_{21} 이 가지는 고유각도의 범위는 식(1)에 따라 θ_{21} 이 된다. 여기서, ω_{21} 은 N_{21} 와 연결된 자식노드들의 개수(가중치)이다.

$$\theta_{21} = \frac{\sum_{n=1}^k N_{2n}}{2\pi} \times \omega_{21}, \theta_{21} = \theta_{31} + \theta_{32} \quad (1)$$

부모노드는 자식노드들의 배치를 위해 고유각의 범위를 자식 노드들에게 전파 및 분할시키고, 반복적인 방법으로 자식노드들을 확장 배치한다.



(그림 3) 연결-우선 동심원 확장 배치 구조

3.3 가중치 기반 동적 분할 배치 알고리즘

전체 노드의 중심성 지수가 계산되었다면, 연결된 자식노드들의 개수를 이용하여 상위노드들의 고유값에 가중치를 부여하고 3D 화면 상에 표시될 노드 위치를 계산하는 가중치 기반 동적 분할 배치 알고리즘을 사용한다. 알고리즘은 [1단계: 각 노드들이 배치될 동심원의 Level을 계산], [2단계: 노드들의 Level이 계산된 후 동심원 내에서 각 노드들이 위치될 각도를 계산], [3단계: 계산된 Level과 각도에 따라 3D 공간상에 노드들을 배치] 과정으로 이루어진다.

【1단계: Level계산】

- ① 중심성 지수가 가장 높은 노드를 Level 1의 root 노드로 지정한다.
- ② root노드를 시작으로 연결정보에 따라 탐색하여 자식노드를 지정, 자식노드는 부모 노드보다 1단계 높은 Level을 갖도록 한다.
- ③(종단 노드 배치 적용 옵션): ①과 ②에서 계산된 Level 정보를 참조하여 가장 높은 Level을 기준으로 각 노드의 Level을 하위의 최대 Level로 정규화하고 각 종단노드들을 최외곽으로 배치한다. 종단 노드 배치 적용을 위한 정규화는 식(2)과 같다.

$$Level = \frac{\max Level \times Level}{\max Level_{child}} \quad (2)$$

【2단계: 각도계산】

- ④ 각 노드는 자신을 포함하여 하위 노드들의 개수의 합을 가중치 점수로 부여하고 고유각도 범위를 지정한다.
- ⑤ 부모노드에 할당된 고유각 범위를 각각의 자식노드들이 가중치 점수 비율대로 분할받고, 자신과 자식 노드들이 배치될 고유각 범위를 할당받는다. 이후 각 노드는 자신에게 할당받은 고유각 범위의 중심각도에 배치된다.

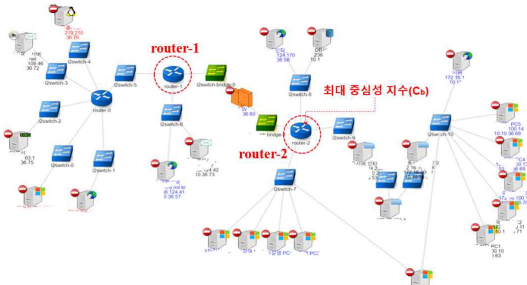
【3단계: 3D배치】

- ⑥ 각 노드별 Level에 따라 다른 반지름과 높이를 가지는 동심원 위에 ⑤에서 계산된 고유각도에 따라 배치되도록 3차원 좌표를 계산한다.
- ⑦ 계산된 좌표에 따라 3D 공간상에 노드를 표시하고, 연결정보에 따라 엣지를 표시한다.

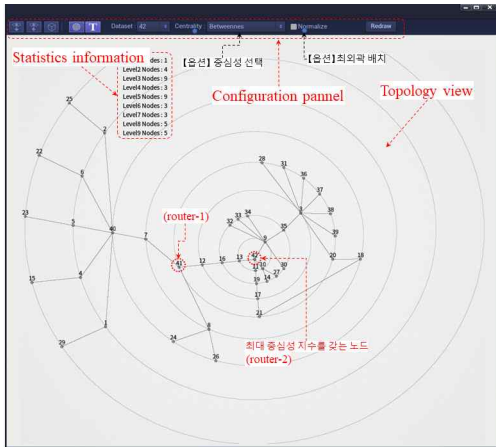
3.4 시각화 인터페이스 구현

토폴로지 시각화 인터페이스 구현 시에 사용한 네트워크 개념 구조는 (그림 4)와 같다. 실제 현업에서 네트워크 관리를 위해 노드를 수동으로 배치하여 운영하고 있는 인터페이스이다. 전체 42개의 노드와 42개의 엣지로 구성되며, 토폴로지 형상은 router-1과 router-2를 기준으로 노드들이 좌우측으로 확장 배치된 구조이다. 논문에서 제안한 방법으로 구현한 토폴로지 시각화 인터페이스는 (그림 5)와 같다. 최소 네트워크 엣지 정보만을 이용했지만, 구현된 시각화 인터페이스는 (그림 4)의 네트워크 개념 구조와 동일하게

router-1과 router-2를 기준으로 노드들이 좌우측으로 확장된 모습이다. (그림 5)에서 적용한 중심성 종류는 매개중심성이다.



(그림 4) 네트워크 개념 구조

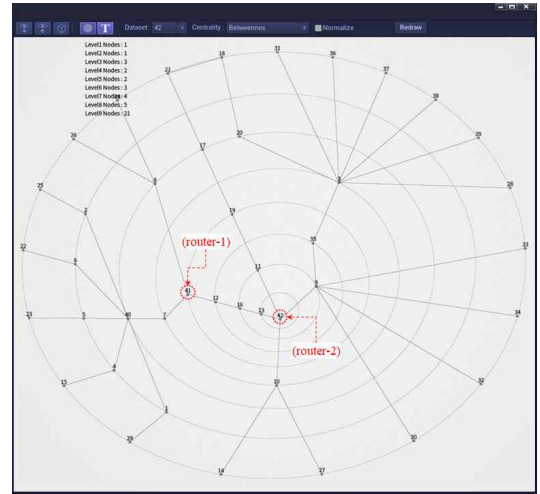


(그림 5) 토폴로지 시각화 인터페이스

(그림 5)의 토폴로지 시각화 인터페이스는 토폴로지-뷰, 설정패널, 통계정보로 구성된다. 토폴로지-뷰는 3D 공간에 네트워크 토폴로지를 시각화하는 화면으로 각 레벨에 따라 반지름과 높이 값이 다른 동심원 상에 네트워크 노드들이 배치된다. 토폴로지는 마우스 조작을 통해 다양한 스케일과 각도에서 살펴볼 수 있다.

설정패널은 시각화할 토폴로지 노드의 데이터-세트, 중심성 종류, 외곽배치 사용 여부, 동심원 표시, 노드명 표시와 같은 시각화 옵션과 토폴로지-뷰의 시점 초기화 기능을 제공한다. 통계정보는 토폴로지-뷰 위에 오버레이되며, 각 레벨별 몇 개의 노드가 존재하는지 깊이 정보를 제공하여 토폴로지 구조 분석을 돕는다. (그림 6)는 중단 노드의 최외곽 배치 옵션을 적용

한 화면이다. 중단 노드들의 고유 레벨을 최대 레벨로 변환 및 적용하여 화면을 효율적으로 사용할 수 있고, 비대칭적인 네트워크를 구조를 대칭적 구조로 살펴볼 수 있다.



(그림 6) 중단 노드 최외곽 배치 적용

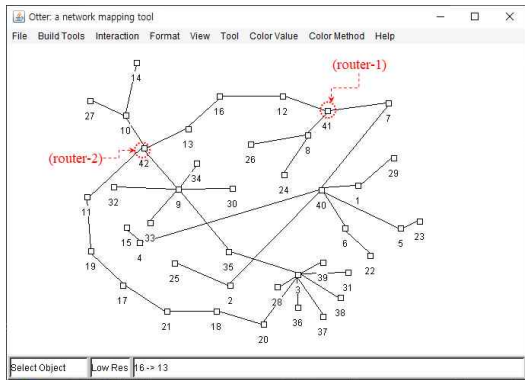
4. 검증 및 결과

본 논문에서 제안한 시각화 방법이 실제 활용 측면에서 유효성이 있는지를 검증하기 위해, 실제 네트워크의 토폴로지를 가시화하고 기존 시각화 도구와의 인터페이스를 비교한다. 검증에 사용된 토폴로지 데이터는 국방과학연구소의 테스트베드 데이터와 퍼듀대학교에서 제공하는 네트워크 데이터 저장소(Network Repository)[19] 데이터를 사용했다. <표 1>은 논문에서 시험 토폴로지 데이터이다. 노드 개수가 42개인 소규모 T1 데이터 세트는 (그림 5)와 (그림6)와 같이 토폴로지 구조화 알고리즘 및 시각화 인터페이스 구현 시에 활용했으며, CAIDA의 Otter와 Plankton 도구에 적용하여 제안한 방법과 비교했다.

<표 1> 시험 토폴로지 데이터

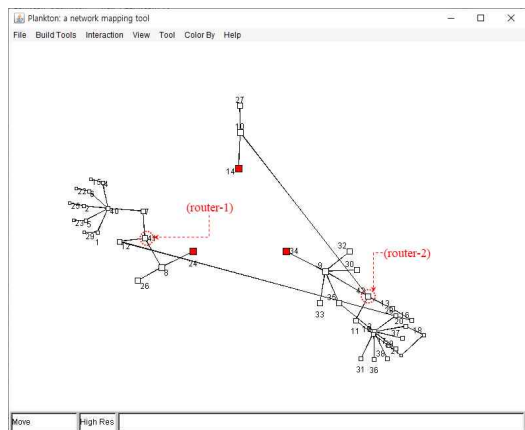
구분	Nodes	Edges	데이터 출처
T1	42	42	국방과학연구소
T2	203	207	
T3	2,113	6,632	퍼듀, 데이터 저장소

(그림 7), (그림 8)는 각각 Otter와 Plankton을 이용하여 T1 토폴로지 데이터를 시각화한 화면이다. Otter은 기본적으로 토폴로지 데이터 내에 노드의 좌표 정보가 포함되어야 하기 때문에 옛지 정보만으로는 실제와 같은 네트워크 형상으로 가시화하는 것은 불가능하고 사용자가 수동적으로 노드들을 위치시켜야 한다.



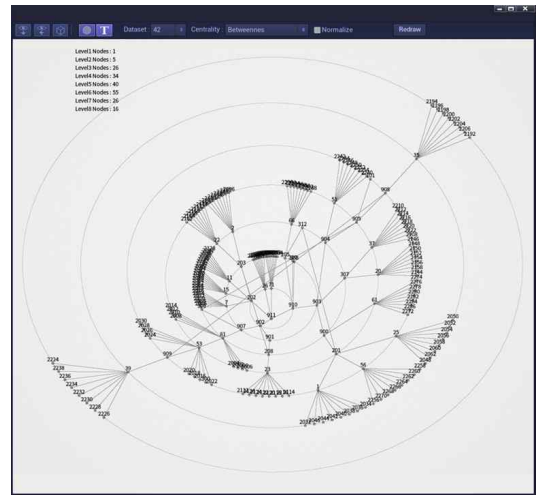
(그림 7) Otter를 이용한 T1 토폴로지 시각화

Plankton은 일정 정도 실제 네트워크 형상으로 가시화되긴 했지만, 14번, 24번, 34번 노드를 화면 중심부에 기준을 잡고 노드들을 배치함으로써 노드들끼리 겹치는 영역이 많이 생겨난다. 따라서, 논문에서 제안한 방법으로 토폴로지를 시각화한 (그림 5)와 (그림 6)이 실제 네트워크 운용 인터페이스인 (그림 4)와 가장 유사하다.

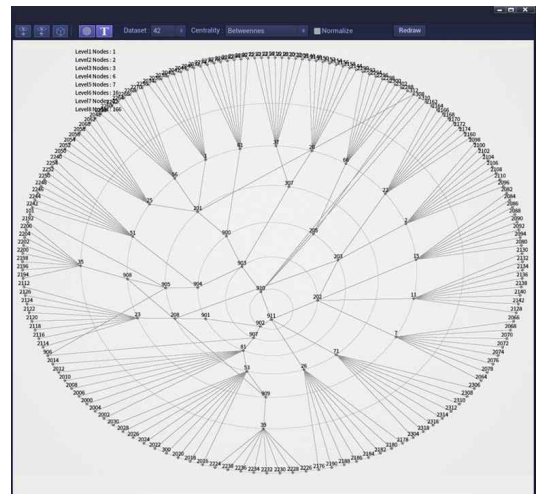


(그림 8) Plankton를 이용한 T1 토폴로지 시각화

(그림 9)는 매개중심성(C_i)을 이용하여 T2 토폴로지 데이터를 시각화했고, (그림 10)은 최외곽 배치 옵션을 적용하여 토폴로지를 가시화했다. (그림 9)를 통해 옛지 라우터들의 배치와 네트워크 깊이 및 비대칭 구조를 탐색할 수 있으며, (그림 10)을 통해서는 중단 노드들의 분포, 중요 라우터들의 연결 구조를 파악할 수 있다.

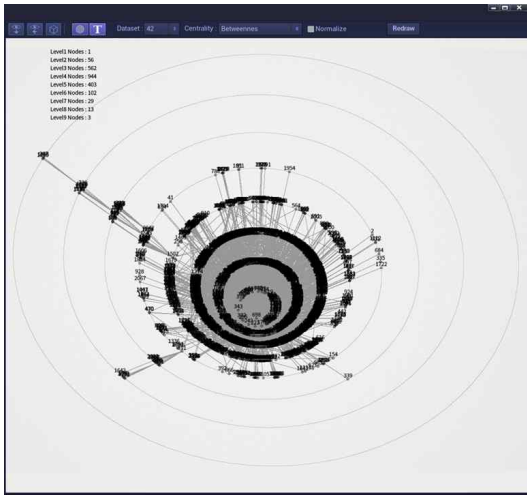


(그림 9) T2 토폴로지 시각화



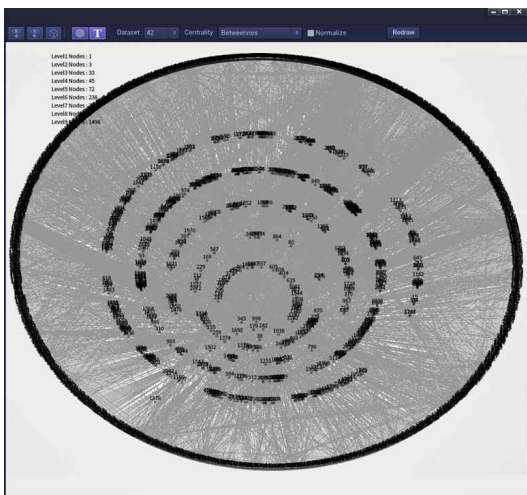
(그림 10) T2 시각화(중단 노드 최외곽 배치)

(그림 11)은 매개중심성(C_i)을 이용하여 T3 데이터의 토폴로지를 시각화한 것이고, (그림 12)는 최외곽 배치 옵션을 적용하여 토폴로지를 가시화했다.

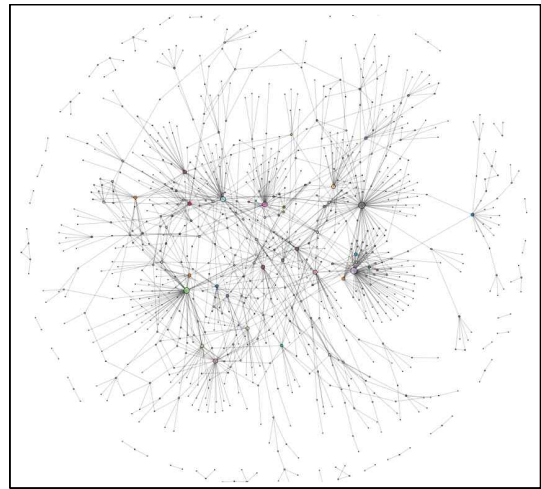


(그림 11) T3 토폴로지 시각화

(그림 13)은 GraphViz를 이용하여 T3 데이터의 토폴로지를 시각화했다. 심미적 관점에서 GraphViz는 토폴로지의 외형을 잘 표출했지만 네트워크의 구조, 대칭 특성, 노드의 중요도 등은 파악하기가 힘들다. 반면에, (그림 11)과 (그림 12)는 동심원 가이드라인을 활용하여 중심노드와 중단노드를 구분 및 직관적으로 표현했으며, 매개중심성(C_p) 이외에 다른 중심성 지수를 선택하고 가시화하면 다른 관점에서 네트워크 구조를 파악할 수 있다.



(그림 12) T3 시각화(중단 노드 최외곽 배치)



(그림 13) GraphViz를 이용한 T3 시각화

5. 결 론

본 논문에서는 네트워크 노드들의 연결 정보만을 이용하여 네트워크 토폴로지를 구조화하는 간단한 알고리즘과 시각화 인터페이스를 설계 및 구현하였다. 네트워크 노드들은 시각화 인터페이스 상에 중심성이 가장 높은 노드를 중심으로 구분된 동심원들을 따라 3차원 공간좌표 상에 배치 및 연결된다. 관리자는 시각화된 토폴로지를 통해 네트워크의 구조와 대칭성을 살펴볼 수 있으며 중단노드의 최외곽 옵션 적용을 통해 중단노드의 위치를 직관적으로 확인할 수 있다.

제안하는 방법은 네트워크의 장애와 성능을 관리하는 분야, 네트워크 트래픽을 분석하고 측정하는 분야, 각종 네트워크 이상 상황을 탐지하고 대응하는 보안관리/관제 분야 등에 활용도가 높을 것으로 예상된다. 향후에는 제안한 시각화 방법을 관리도메인이 다른 분리된 네트워크(멀티 도메인)에 적용하여 토폴로지를 통합 관리하는 방법, 중심성 지수를 시각화 인터페이스의 노드간 거리와 노드 크기에 적용하는 방법, 토폴로지를 하이퍼볼릭 공간으로 왜곡·변형하여 대규모 토폴로지에 적용하는 방법으로 연구를 진행할 계획이다.

참고문헌

- [1] T. Landesberger, A. Kuijper, T. Schreck, et al., “Visual Analysis of Large Graphs: State-of-the-Art and Future Research Challenges”, *Computer Graphics Forum*, Wiley, Vol. 30, No. 6, pp. 1719-1749, 2011.
- [2] I. Herman, G. Melancon, M.S. Marshall, “Graph visualization and navigation in information visualization: A survey”, *IEEE Transactions on Visualization and Computer Graphics*, Vol. 6, No. 1, pp. 24-43, 2000.
- [3] E. Adar, “Guess: A Language and Interface for Graph Exploration”, *Proc. of the SIGCHI conference on Human Factors in computing systems*, pp. 791 - 800, 2006.
- [4] M. Bastian and S. Heymann, M. Jacomy, “Gephi: An Open Source Software for Exploring and Manipulating Networks”, *Proc. of the Third International ICWSM Conference*, pp. 361-362, 2009.
- [5] 김용학, 김영진, ‘사회 연결망 분석(4판)’, 박영사, 2016.
- [6] CAIDA(Center for Applied Internet Data Analysis), <https://www.caida.org/>, (Accessed Feb. 1, 2021.)
- [7] R. Zafarani, M. A. Abbasi, H. Liu, ‘Social Media Mining: An Introduction’, Cambridge University Press, 2014.
- [8] M. J. McGuffin, “Simple Algorithms for Network Visualization: A tutorial”, *Tsinghua Science and Technology*, Vol. 17, No. 4, pp. 383-398, 2012.
- [9] B. Huffaker, D. Plummer, D. Moore, k. claffy, “Topology discovery by active probing”, *Symposium on Applications and the Internet (SAINT)*, pp. 90-96, 2002.
- [10] B. Huffaker, J. Jung, E. Nemeth, D. Wessels, K. Claffy, “Visualization of the Growth and Topology of the NLANR Caching Hierarchy”, *Comput. Networks and ISDN Systems*, Vol. 30, No. 22-23, pp. 2131-2139, 1998.
- [11] B. Huffaker, E. Nemeth, K. Claffy, “Otter: A General - Purpose Network Visualization Tool”, *Proc. of the International Networking Conference (INET)*, 1999.
- [12] T. Munzner, “Drawing large graphs with H3 Viewer and Site Manager”, *Lecture Notes in Computer Science, LNCS 1547*, pp. 384-393, 1998.
- [13] D. Auber, D. Archambault, R. Bourqui, et al., “The Tulip 3 Framework: A Scalable Software Library for Information Visualization Applications Based on Relational Data”, *Technical Report RR-7860, INRIA Bordeaux Sud-Ouest*, 2012.
- [14] J. Ellson, E. R. Gansner, E. Koutsofios, S. C. North, G. Woodhull, “Graphviz and Dynagraph - Static and Dynamic Graph Drawing Tools”, *Graph Drawing Software, Springer-Verlag*, pp. 127-148, 2003.
- [15] V. Batagelj, A. Mrvar, “Pajek - Program for Large Network Analysis”, *Connections*, Vol. 21, No. 2, pp. 47-57, 1998.
- [16] 네트워크 이론 - 다양한 중심성(Centrality) 척도들, <https://bab2min.tistory.com/554>, Accessed Feb. 1, 2021.
- [17] B. Lee, C. Plaisant, C. S. Parr, et al., “Task taxonomy for graph visualization”, *Proc. of the AVI Workshop BEyond Time and Errors: Novel Evaluation Methods for Information Visualization (BELIV)*, pp. 1-5, 2006.
- [18] J. Leskovec, R. Sosič., “SNAP: A General Purpose Network Analysis and Graph Mining Library”, *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 8, no. 1, pp. 1-20, 2016.
- [19] NETWORK REPOSITORY, <http://networkrepository.com>, Accessed Feb. 1, 2021.

— [저 자 소 개] —



장 범 환 (Beom-Hwan Chang)

1997년 2월 성균관대학교 전자공학과
공학사
1999년 2월 성균관대학교 컴퓨터공학과
공학석사
2003년 2월 성균관대학교 컴퓨터공학과
공학박사
2012년 3월 ~ 현재 호원대학교
IT소프트웨어보안학과 부교수
email : bchang@howon.ac.kr



류 제 민 (Jemin Ryu)

2010년 2월 경원대학교 인터넷미디어학
공학사
2012년 8월 가천대학교 모바일소프트웨
어학과 공학석사
2015년 8월 ~ 현재 (주)에이알씨엔에
스 제직
email : jmryu@arcns.kr



권 구 형 (Koohyung Kwon)

2001년 2월 고려대학교 전기전자전파
공학 학사
2003년 2월 고려대학교 전파공학과 석
사
2006년 7월 ~ 현재 국방과학연구소 제
직
email : koohyung@add.re.kr