

멀티 도메인 네트워크 토폴로지 시각화 연구★

장 범 환*

요 약

도메인이 여러 개로 구성된 멀티 네트워크를 운영하는 조직에서는 단일 화면 상에 다수의 네트워크 토폴로지를 시각화하여 전체 노드들과 링크들을 감시할 수 있는 기능이 필요한데, 도메인간의 연결점이 없는 토폴로지를 단일 화면 상에 표시하는 것은 간단하지가 않다. 게다가, 다수의 네트워크 형상들을 단일 화면 상에 실제와 같은 물리적 모습으로 표시하는 것은 어렵기 때문에, 멀티 도메인 네트워크를 하나의 화면을 통해 관리하는데는 어려움이 있다.

본 연구에서는 멀티 도메인 환경에서 노드간의 최소 연결 정보만을 이용하여 토폴로지를 시각화하는 효율적인 방법을 제안한다. 이 방법은 네트워크를 구성하는 노드들의 중심성 지수를 활용하여 도메인별 중심 노드를 찾고, 자식 노드들과의 연결 정보를 재귀적으로 탐색하여 토폴로지를 생성한다. 화면 상의 도메인별 토폴로지 표시 영역은 자식 노드들의 가중치의 합으로 계산 및 동적 분할되고, 가상의 Root 노드를 활용하여 3D 공간 상에 전체 노드들을 배치함으로써 멀티 도메인 토폴로지를 시각화한다. 이 방법은 노드간의 최소 연결 정보만을 이용하는 단순한 방법이지만, 멀티 도메인 네트워크의 물리적 형상을 유지하면서 하나의 화면에 표시할 수 있는 효율적인 토폴로지 시각화 방법이다.

Study on Visualization of Multi-domain Network Topology

Beom-Hwan Chang*

ABSTRACT

In general, organizations operating multi-domain networks find it difficult to represent and manage multiple domain networks on a single screen space. Instead, most of them are managed with multiple screens visualizing network topology by domain or partitioning one screen area into multiple domains. We propose an efficient method to visualize the topology using only minimal connection information between domain-agnostic nodes in this work. This method visualizes the topology by utilizing centrality indices representing the influence of nodes in the network. Furthermore, the method dynamically segments the entire node's display area using virtual Root nodes to auto-separate domains and weights of child nodes and placing nodes in 3D space. Thus, although it is a straightforward method, the multi-domain network topology can be visualized with only minimal connection information between nodes.

Key words : Network Visualization, Multi-domain Network Topology Visualization, Node Centrality Visualization

접수일(2022년 10월 05일), 게재확정일(2022년 10월 31일)

* 호원대학교/IT소프트웨어보안학과 (주저자, 교신저자)

★ 본 논문은 2022년 호원대학교 학술연구비의 지원을 받았음.

1. 서 론

네트워크 토폴로지는 컴퓨터 네트워크의 요소(노드, 엣지 등)들을 물리적으로 연결해 놓은 모습, 또는 그 연결방식을 말한다. 토폴로지 시각화는 네트워크 구조를 점과 선으로 연결하여 물리적 또는 논리적인 그래프 형태로 표시하는 방법이다[1][2]. 구조적인 측면에서 연결망 그래프는 그 자체로 분석적 의미를 표현할 뿐만 아니라, 그래프로 시각화된 패턴들은 연구자들로 하여금 더 많은 해석 가능성도 열어준다[3].

네트워크 형상을 실제와 같은 모습으로 표시하는 물리적 토폴로지 시각화는 네트워크의 전체 구조를 이해하는데 유리하다[4][5]. 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[3]. 망구성 방식을 보여주는 토폴로지 시각화는 복잡한 네트워크의 전체 구조를 파악하고 네트워크 체계를 구성하는 노드들의 행위, 연결 관계, 상호작용 등 네트워크 특성을 쉽게 이해할 수 있는 도구이다[6].

관리 도메인이 여러 개로 구성된 멀티 네트워크를 운영하는 조직에서는 단일 화면 상에 다수의 네트워크 토폴로지를 시각화하여 전체 노드들과 링크들을 감시할 수 있는 기능이 필요하다. 기존의 멀티 토폴로지 시각화 방식들은 별도의 화면들을 이용하여 도메인별 네트워크 토폴로지를 시각화하거나, 또는 하나의 화면을 다수로 분할하고 도메인별 네트워크 토폴로지를 각각 시각화하는 형태가 대부분이다. 전자는 관리 측면에서 네트워크 전체 구조를 이해하고 감시하는데 불리하기 때문에 일반적으로 후자를 선호하지만 화면 분할 방법은 도메인의 특성에 따라 동적으로 시각화하기 어려운 문제점이 있다. 예를 들면, 화면 분할을 수행하는데 있어서 대상 토폴로지 개수가 2 또는 4의 배수가 아닌 경우, 사각으로 구성된 화면을 단순 배분하게 되면 표시 영역이 비효율적으로 나뉠 수 있고, 토폴로지의 물리적 형상은 좁은 영역에 축소·표시되어 네트워크 구조를 파악 못할 수도 있다. 또한, 토폴로지를 구성하는 노드 수에 비례(토폴로지 크기)하여 화면을 분할하는 상황과 토폴로지 개수가 동적으로 변경되는 상황에서

는 화면 분할을 더욱 어렵게 만든다.

반면에, 일종의 트릭이지만 단일 화면에 다수의 토폴로지를 시각화하는 방법으로 단일 토폴로지를 가상의 연결점(노드)을 이용하여 다수의 토폴로지를 시각화할 수 있다. 대부분의 기존 시각화 알고리즘들은 노드들의 연결을 탐색하며 노드들을 배치하는데, 노드들의 연결이 단힌 그래프가 아닌 경우, 즉 끊긴 점이 발생하게 되면 끊긴 점 이후의 노드들은 메인 토폴로지와는 떨어진 섬(붕괴된 노드들) 형태로 화면에 가시화된다. 하지만, 가상의 연결점을 이용하여 토폴로지 데이터를 구성하면 하나의 화면 상에 다수의 토폴로지를 표시할 수는 있으나 메인 토폴로지 외의 다른 토폴로지는 메인 토폴로지 내의 좁은 영역으로 한정되어 표시되거나 본래 토폴로지 물리적 형상을 유지하기가 어렵다.

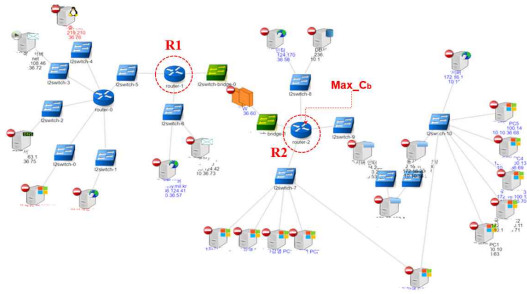
본 논문에서는 네트워크를 구성하는 노드들의 최소 연결 정보만을 이용하여 단일 화면 상에 멀티 도메인의 네트워크 토폴로지를 표시하는 방법을 제안한다. 이 방법은 네트워크를 구성하는 노드들의 중심성 지수를 활용하여 도메인별 중심 노드를 찾고, 자식 노드들과의 연결 정보를 재귀적으로 탐색하여 토폴로지를 생성한다. 화면 상의 도메인별 토폴로지 표시 영역은 자식 노드들의 가중치의 합으로 계산 및 동적 분할되고, 가상의 Root 노드를 활용하여 3D 공간 상에 전체 노드들을 배치함으로써 멀티 도메인 토폴로지를 시각화한다. 이 방법은 노드간의 최소 연결 정보만을 이용하는 단순한 방법이지만, 멀티 도메인 네트워크의 물리적 형상을 유지하면서 하나의 화면에 표시할 수 있는 효율적인 토폴로지 시각화 방법이다.

2. 관련 연구

네트워크란 노드(Node)와 엣지(Edge)로 이루어진 추상적인 개념으로써 노드는 하나의 개념이나 실체가 되고 엣지는 노드 사이의 연결 또는 관계를 의미한다 [7]. 멀티 도메인 네트워크는 개별적으로 관리·운영되는 단일 도메인 네트워크들의 집합으로 정의할 수 있으며, 물리적/논리적으로 연결 또는 분리될 수도 있다 [6]. 따라서, 단일 도메인 토폴로지 시각화를 기반으로

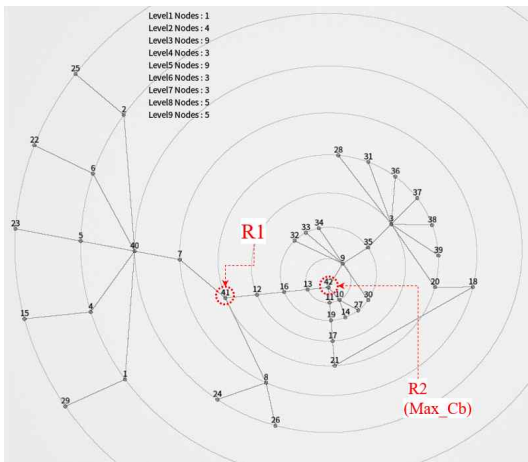
멀티 도메인 토폴로지 시각화를 완성할 수 있다.

이전 연구에서 단일 도메인 토폴로지 시각화 연구를 위해 (그림 1)와 같은 42개의 노드와 42개의 엣지로 구성된 네트워크 개념 구조인 N1 데이터셋과 시각화 인터페이스를 참조했다[6]. (그림 1)은 실제 현업에서 네트워크 관리를 위해 노드를 수동으로 배치하여 운영하고 있는 인터페이스로써, 토폴로지 형상은 router R1과 router R2를 기준으로 노드들이 좌·우측으로 확장 배치된 구조이다[6].



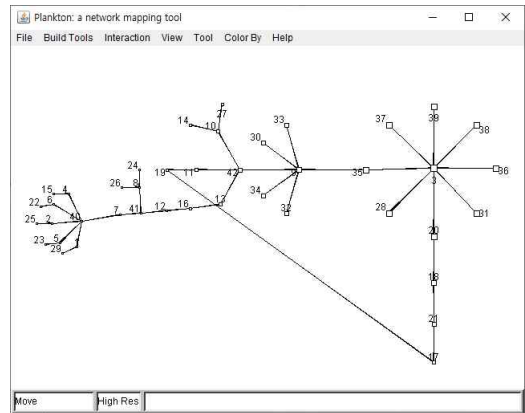
(그림 1) 네트워크 개념 구조

이전 연구 방법에 따라 구현한 토폴로지 시각화 인터페이스는 (그림 2)와 같다[6]. (그림 1)의 네트워크 개념 구조와 유사하게 router R1과 R2를 기준으로 노드들이 좌·우측으로 확장된 모습이다. 최소 네트워크 엣지 데이터만을 이용하였고, 네트워크를 구성하는 노드들의 중심성 지수값을 기반으로 토폴로지를 시각화했다. 여기서, 적합한 중심성의 종류는 매개중심성(C_b)이다[8].

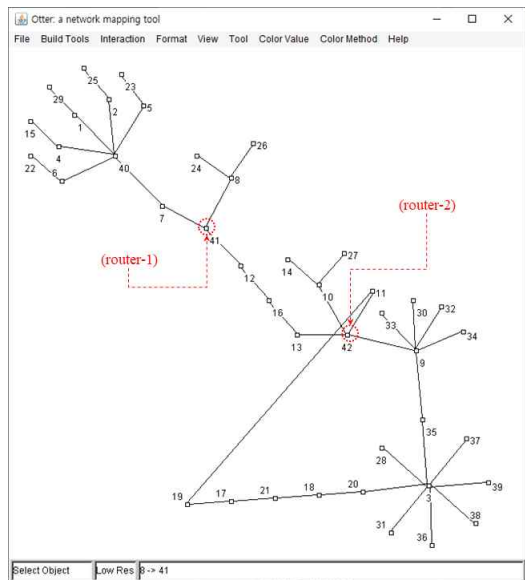


(그림 2) 노드 중심성을 이용한 N1 시각화

CAIDA는 과거부터 네트워크 토폴로지 시각화 기술에 대한 포괄적인 조사와 연구를 꾸준히 발전시켜오고 있다[9]. 대표적인 도구로는 Plankton[10]과 Otter[11]이 있으며, 특히 Otter는 범용적으로 사용되는 엣지와 노드로 구성된 대부분의 연결 데이터를 처리할 수 있다. (그림 3)과 (그림 4)는 Plankton과 Otter를 이용하여 N1 토폴로지를 시각화한 인터페이스이다. 토폴로지 형상은 참조한 네트워크 개념 구조인 (그림 1)과는 약간의 차이가 있지만 대체적으로 유사하며, Otter가 참조한 네트워크 개념 구조와 근사하게 시각화를 수행했다.



(그림 3) Plankton를 이용한 N1 토폴로지 시각화

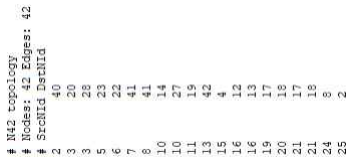


(그림 4) Otter를 이용한 N1 토폴로지 시각화

3. 멀티 도메인 토폴로지 시각화 방법

3.1 토폴로지 데이터 구조 및 중심성 지수 계산

본 연구에서 사용하는 토폴로지 데이터는 범용성을 갖는 일반적인 네트워크 연결 데이터로써, (그림 5)와 같이 도메인 구분이 없는 최소 연결(엣지) 정보만을 갖는, 즉 SrcNid(근원지 노드 식별자)와 DstNid(목적지 노드 식별자)만을 사용한다.



(그림 5) 토폴로지 데이터 구조

첫째, 노드의 쌍으로 이루어진 연결 정보들을 탐색하며 노드의 중복 여부를 검사하고 네트워크의 전체 노드 목록을 만든다. 노드 탐색 과정 중에 네트워크 분석 라이브러리인 SNAP(Stanford Network Analysis Project)[12]를 사용하여 모든 노드들의 연결중심성(C_d : degree centrality), 근접중심성(C_c : closeness centrality), 매개중심성(C_b : betweenness centrality), 고유벡터중심성(C_e : eigenvector centrality) 등의 값을 계산하여 저장한다. SNAP은 스탠포드 대학에서 개발한 네트워크 분석 라이브러리로서, 네트워크를 구성하는 노드들의 중심성 값을 계산할 수 있다. SNAP을 활용하여 계산한 노드들의 중심성 값은 (그림 6)와 같다.

```

#Network: ./node42_edges.txt
#Nodes: 42 Edges: 42
#NodeId Degree Closeness Betweeness 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 0.419341 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 6) SNAP을 이용한 중심성 지수 계산

둘째, 사용자가 설정한 배치 옵션인 토폴로지의 배치 기준(중심성 종류에 따른 중심성 지수 값)에 따라 전체 노드를 내림차순으로 정렬한 노드 목록을 생성한

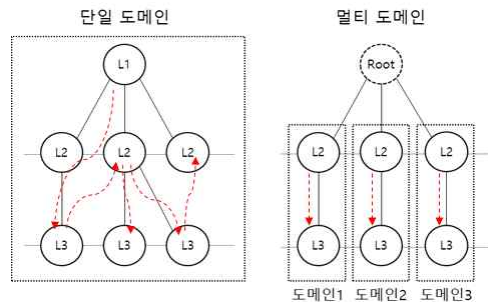
다. 일반적으로 배치 기준 기본값은 매개중심성(C_b)인데, 이는 매개중심성이 네트워크 상의 모든 노드를 통과하는 정도를 측정·환산한 값으로써 네트워크의 중심부를 표현하는데 효과적이기 때문이다[8][13].

3.2 도메인 자동 분류 및 하위 노드 탐색

전체 노드들의 중심성 지수가 계산 및 정렬되었으면, 가상의 Root 노드를 생성하여 중심(Level 1) 노드로 지정한다. 전체 노드 목록에서 첫 번째 노드 즉, 중심성 수치가 가장 높은 노드인 최중심 노드를 Root 노드의 첫 번째 자식(Level 2) 노드로 지정하는데 이 노드는 해당 도메인의 최상위(Root_domain) 노드가 된다. 이후 최중심 노드를 시작점으로 연결 정보를 따라가며 자식 노드를 탐색하고 자식 노드는 부모 노드보다 1단계 높은 Level을 갖도록 부여한 후 노드 목록에서 제거한다. 더 이상 연결된 자식 노드가 없을 경우, 노드 탐색은 다음 두 가지 형태로 진행되며 (그림 7)과 같은 경로로 하위 노드 탐색을 수행한다.

최중심 노드로부터 하위노드 탐색 후, ① 전체 노드 목록에 노드가 남아 있지 않다면 모든 노드를 탐색했으므로, 전체 네트워크가 단일 도메인 네트워크로 이루어졌기 때문에 가상의 Root 노드를 제거하고 도메인의 Root_domain 노드를 최상위 부모 노드로 삼는다.

② 전체 노드 목록에 노드가 남아 있다면, 남은 노드 중에서 중심성 수치가 가장 높은 노드를 다시 가상 Root 노드의 자식노드(새로운 도메인의 Root_domain 노드)로 지정하고, 연결 정보를 따라가며 하위 노드들 탐색하며 노드의 Level을 부여한다. 모든 노드의 Level이 부여될 때까지 재귀적으로 위의 작업을 반복한다. 여기서, Level은 네트워크의 깊이 정보이다.



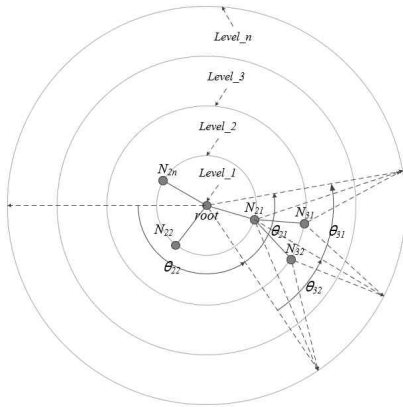
(그림 7) 도메인 분류 및 자식 노드 탐색

모든 노드의 탐색이 완료된 후, 도메인별 각 노드의 Level 값을 전체 노드의 최대 Level 값을 이용하여 정규화하여 저장하고, 3.4절의 시각화 단계에서 중단 노드의 최외곽 배치 옵션이 설정되었다면 중단 노드들을 최외곽으로 배치한다. 중단 노드 배치 적용을 위한 정규화는 식(1)과 같다.

$$Level_{new} = \frac{\max Level_{domain} \times Level}{\max Level_{all}} \dots\dots \text{식(1)}$$

3.3 가중치 기반 동적 분할 배치

가중치 기반 동적 분할 배치는 실제 노드들을 화면 상에 배치하기 위한 노드들의 점 좌표 계산 단계이다. 각 노드에는 자기 자신을 포함하여 모든 하위 자식 노드 개수의 총합을 가중치로 부여하고, 최상위 부모는 가중치 점수에 따라 고유각의 범위를 부여받는다. 이후 부모 노드에 할당된 고유각의 범위를 각각의 자식 노드들이 가중치 점수의 비율에 따라 분할받고, 각 노드는 자신에게 할당받은 고유각 범위와 중심각도에 의해 (그림 8)와 같이 배치된다.

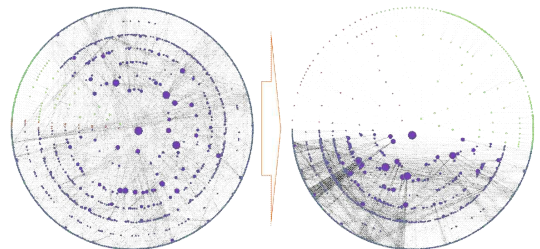


(그림 8) 가중치 기반 동적 분할 배치

(그림 8)에서 Root 노드에 연결된 자식 노드들이 $N_{21}, N_{22}, \dots, N_{2n}$ 이라고 할 때, N_{21} 이 가지는 고유각도의 범위는 식(2)에 따라 θ_{21} 이 된다. 여기서, ω_{21} 은 N_{21} 와 연결된 자식 노드들의 개수의 합(가중치)이다. 이때 도메인별 네트워크 규모의 차이가 큰 경우에는 소규모 네트워크 토폴로지의 가시성을 위해서 최상위 노드 (Root_domain)의 고유각 범위는 가중치의 로그-스케일

일(Logarithmic Scale, 로그배율)을 적용한 비율에 따라 할당할 수도 있다. 이 때 가중치 점수가 1인 경우는 log 값이 0이 되므로 모든 노드의 가중치 점수에 1을 더하여 log를 적용하도록 한다. 로그스케일이 적용된 결과는 (그림 9)과 같다.

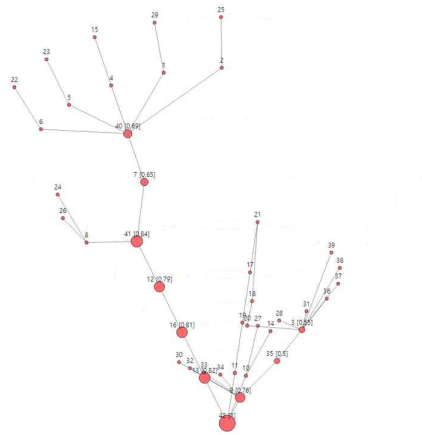
$$\theta_{21} = \frac{\sum_{n=1}^k N_{2n}}{2\pi} \times \omega_{21}, \theta_{21} = \theta_{31} + \theta_{32} \dots\dots \text{식(2)}$$



(그림 9) 로그스케일 적용 토폴로지 시각화

3.4 네트워크 토폴로지 3D 변형

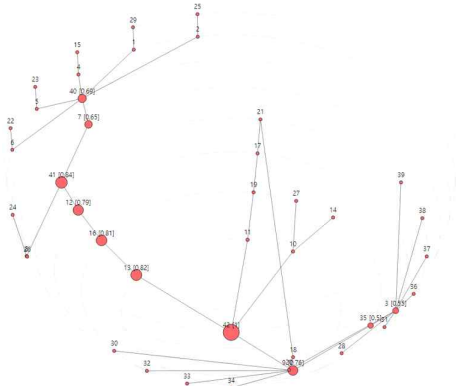
3.2절에서 노드의 Level이 결정되고, 3.3절에서 노드의 배치각이 결정되면, 노드들이 배치될 동심원(Level)의 반지름(r)과 노드의 높이(h)를 조정하여 3D 공간 상에서 토폴로지를 다양하게 변형할 수 있다. 첫째, level에 따라 r 과 h 의 값을 선형적(Linear)으로 증가시키면 토폴로지는 (그림 10)와 같이 원뿔형으로 가시화된다.



(그림 10) 원뿔형 노드 배치

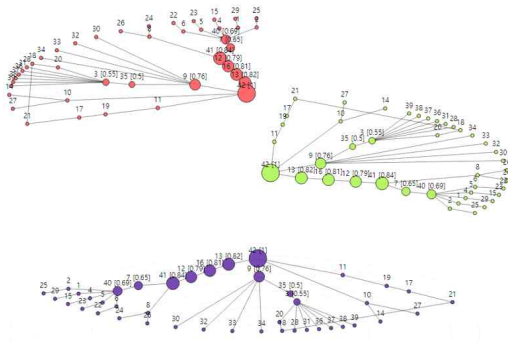
둘째, 동심원의 반지름(r)과 높이(h)를 피타고라스 정리를 이용하여, 식(3)에 의해 반지름(r)을 변화시킬 경우에는 (그림 11)과 같은 반구형 배치가 가능하다.

$$r = \sqrt{(MaxH - MinH)^2 - (MaxH - h)^2} \dots\dots \text{식(3)}$$



(그림 11) 반구형 노드 배치

셋째, 동심원의 반지름(r)을 선형적(Linear)으로 증가시키고 도메인 개수에 따라 높이(h)를 다르게 하면, (그림 12)과 같이 도메인별 레이어 배치가 가능하다.



(그림 12) 도메인별 계층식 노드 배치

지금까지 3.2절 도메인 자동 분류 및 하위 노드 탐색, 3.3절 가중치 기반 동적 분할 배치, 3.4절 네트워크 토폴로지 3D 변형 알고리즘을 요약하면, 아래와 같이 [1단계: 노드들의 r (배치될 동심원의 Level) 계산], [2단계: 노드들의 θ (동심원 상의 노드들의 각도) 계산], [3단계: 3D 공간 상에 노드 배치] 과정으로 이루어진다.

【1단계: 노드의 r (Level) 계산】 - 3.2절

- 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(중단 노드 최외곽 배치 적용 옵션): ①과 ②에서 계산된 Level 정보를 참조하여 가장 높은 Level을 기준으로 각 노드의 Level을 하위의 최대 Level로 정규화하고 각 중단노드들을 최외곽으로 배치한다. 중단 노드 배치 적용을 위한 정규화는 식(1)과 같다.

【2단계: 노드의 θ (각도) 계산】 - 3.3절

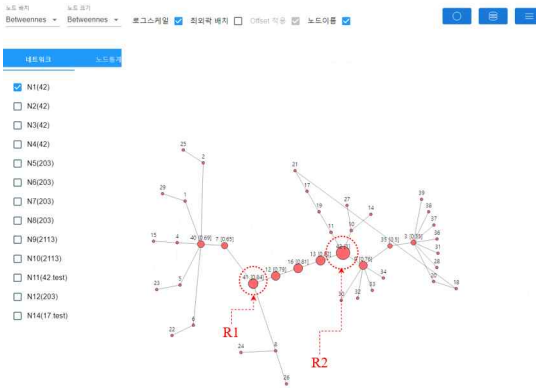
- ④ 각 노드는 자신을 포함하여 하위 노드들의 개수의 합을 가중치 점수로 부여하고 고유각도 범위를 지정한다. 고유각도 범위는 식(2)에 의해 계산된다.
- ⑤ 부모노드에 할당된 고유각 범위를 각각의 자식노드들이 가중치 점수 비율대로 분할받고, 자신과 자식 노드들이 배치될 고유각 범위를 할당받는다. 이후 각 노드는 자신에게 할당받은 고유각 범위의 중심에 배치된다.

【3단계: 노드의 3D 배치 및 변형】 - 3.4절

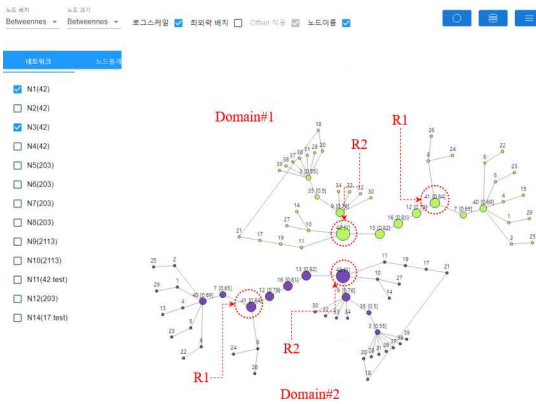
- ⑥ 각 노드별 Level에 따라 다른 반지름과 높이를 가지는 동심원 위에 ⑤에서 계산된 고유각도에 따라 배치되도록 3차원 좌표를 계산한다. 3차원 좌표의 높이(h) 값은 사용자 옵션에 따라 원뿔형, 반구형, 계층형으로 변형 및 적용된다.
- ⑦ 계산된 좌표에 따라 3D 공간상에 노드를 표시하고, 연결정보에 따라 엣지를 표시한다.

3.5 멀티 도메인 토폴로지 시각화

전체 노드들의 위치 정보($r, \theta, h=0$)가 계산되었으면, 가상의 Root 노드는 실제 노드가 아니므로 생략한 후, Root_domain 노드부터 하위 모든 자식 노드들을 (그림 13), (그림 14)와 같이 가시화한다. 이때, 사용자가 설정한 노드 배치 기준과 노드 크기 기준에 따라 중심성 지수 값을 노드 크기에 반영하여 노드들을 화면에 표시한다.



(그림 13) 단일 도메인 토폴로지 시각화



(그림 14) 멀티 도메인 토폴로지 시각화

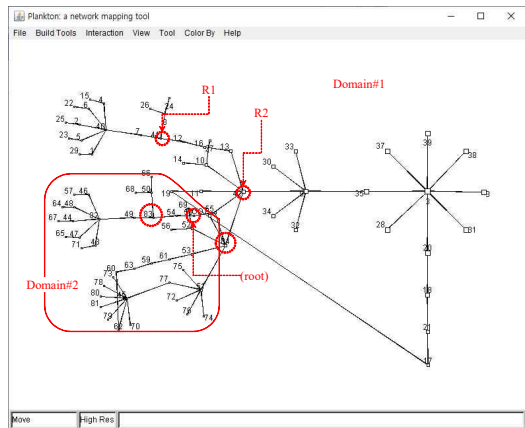
4. 시각화 검증 및 분석

본 논문에서 제안한 시각화 방법이 실제 활용 측면에서 유효성이 있는지를 검증하기 위해, 다수 도메인의 네트워크 데이터를 <표 1>와 같이 준비하고 CAIDA의 Plankton과 Otter를 이용하여 토폴로지를 시각화한 후 제안하는 방법과 비교 및 분석한다. 검증에 사용된 토폴로지 데이터는 멀티 도메인의 시각화 비교 및 분석을 위해 국방○○연구소의 테스트베드 데이터와 퍼듀대학교에서 제공하는 네트워크 저장소(Network Repository)[14] 데이터를 복수 개로 복사한 후 사용했다.

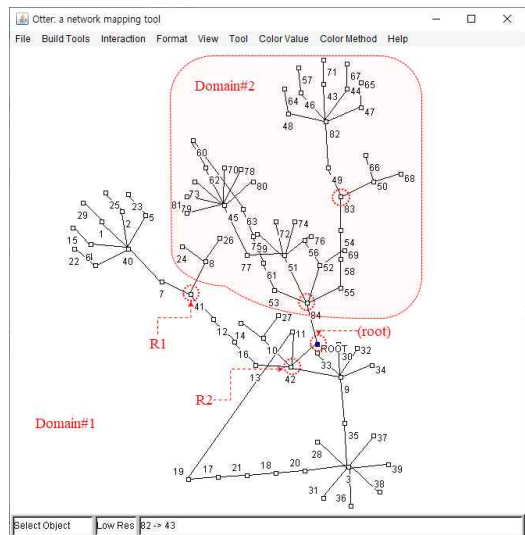
<표 1> 시험 네트워크 데이터

구분	Nodes 수	Edges 수	데이터 출처
N1~4	42	42	국방○○연구소
N5~8	203	207	
N9~10	2,113	6,632	퍼듀, 데이터 저장소

Plankton과 Otter의 경우, 네트워크 내의 노드 간 연결 정보에서 끊긴 지점이 발생하면 끊긴 지점 이후의 노드들은 화면 좌상단 특정 영역에 떨어진 섬(뿔친 노드들) 형태로 표시된다. 따라서 두 개 이상의 도메인을 가시화하기 위해서는 가상의 Root 노드를 활용하여 도메인 간을 연결(노드 42번과 노드84번)하도록 데이터를 구성했다.



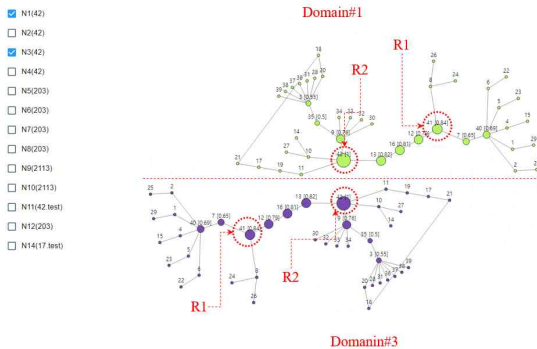
(그림 15) Plankton을 이용한 N1 및 N2 가시화



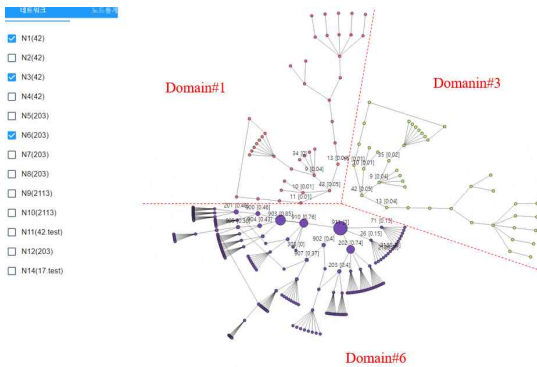
(그림 16) Otter를 이용한 N1 및 N2 가시화

Plankton과 Otter를 이용한 도메인 N1과 도메인 N2 가시화 결과는 (그림 15), (그림 16)과 같다. 메인 토폴로지 N1 이외의 다른 토폴로지 N2는 메인 토폴로지에 연결된 좁은 영역으로 한정되어 시각화되었고, 본래 토폴로지의 물리적 형상을 유지하지 못하며 도메인 구분(식별) 역시 어려운 상태이다. 따라서, 시각화할 대상 도메인 개수가 증가할수록 토폴로지 시각화를 통한 네트워크 관리는 불가능하게 된다.

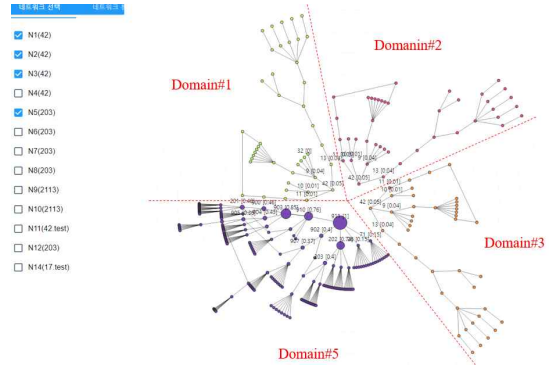
본 논문에서 제안한 방법으로 멀티 도메인 토폴로지를 시각화한 결과는 (그림 17) ~ (그림 21)와 같으며, 기본값으로 도메인별 로그-스케일 옵션을 적용했다. (그림 17)은 2개 도메인 및 84개의 노드를, (그림 18)은 3개 도메인 및 287개의 노드를, (그림 19)와 (그림 20)은 4개 도메인 및 329개의 노드를, (그림 21)은 4개 도메인 및 2,400개의 노드를 시각화 결과이다.



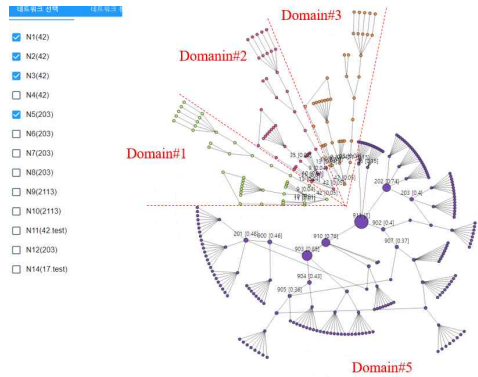
(그림 17) N1(42), N3(42) 가시화



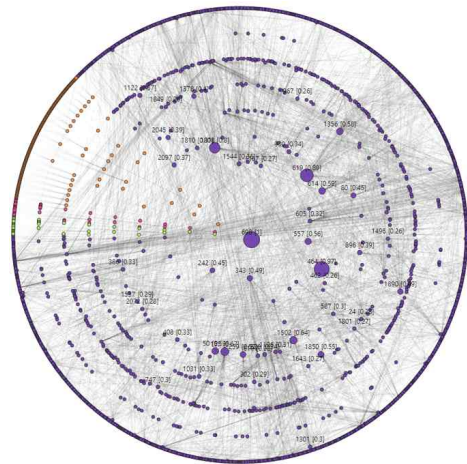
(그림 18) N1(42), N3(42), N6(203) 가시화



(그림 19) N1(42), N2(42), N3(42), N5(203) 가시화



(그림 20) N1(42), N2(42), N3(42), N5(203) 가시화 (로그-스케일 옵션 미적용)



(그림 21) N1(42), N3(42), N6(203), N9(2113) 가시화

제안하는 시각화 방법은 각 도메인별 네트워크의 물리적 형상을 유지하면서 다수 도메인의 토폴로지를 실시간 시각화한다. 또한, 토폴로지를 통해 네트워크의 구조, 네트워크의 대칭 특성, 노드의 중요도를 관찰할 수 있고, 노드의 배치 옵션과 노드 크기 옵션, 그리고 도메인의 로그-스케일 적용 옵션을 통해 사용자는 각 도메인의 노드들을 감시 및 관찰할 수 있다. 다만, (그림 21)과 같은 대규모 네트워크 시각화에 있어서는 가시성이 아쉬운 측면이 있지만, 사용자는 중단 노드의 최외곽 옵션 기능, 도메인 선택 기능, 줌-인/아웃 기능 등을 통해 일정 정도 문제점을 해결할 수 있다고 생각된다.

5. 결 론

본 논문에서는 도메인 구분없는 네트워크 노드들의 최소 연결 정보만을 이용하여 단일 또는 멀티 도메인의 네트워크 토폴로지를 시각화하는 방법을 제안했다. 이 방법은 노드 간의 연결 정보를 이용하여 노드의 중심성 값을 계산하고 도메인을 자동 구분하며, 가상의 Root 노드를 이용하여 기준점(Root 노드)에서 일정 반지름을 갖는 동심원들을 따라 3차원 공간좌표 상에 노드들을 배치 및 연결한다. 토폴로지는 동심원의 반지름(r)과 노드들의 높이(h) 값을 변경함으로써 원뿔형, 반구형 등 다양한 모습으로 변형하여 가시성을 극대화했다. 관리자는 시각화된 멀티 도메인 토폴로지를 통해 다수의 관리 네트워크 구조와 대칭성을 살펴볼 수 있으며, 중단 노드(예, 호스트)의 최외곽 옵션 적용을 통해 중단 노드들을 직관적으로 확인할 수 있다.

제안하는 방법은 네트워크 토폴로지를 기반으로 노드 연관정보 데이터가 존재할 경우, 즉 노드의 속성(종류, 이름, IP주소, 크기, 색상 등)과 상태(온라인/오프라인, 정상/비정상 등) 등을 노드에 적용하여 표시할 수 있다. 따라서, 네트워크의 장애와 성능을 관리하는 분야, 네트워크 트래픽을 분석하고 측정하는 분야, 각종 네트워크 이상 상황을 탐지하고 대응하는 네트워크 보안 관리 및 관제 분야 등에 활용도가 높을 것으로 예상된다.

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] 김용학, 김영진, '사회 연결망 분석(4판)', 박영사, 2016.
- [4] 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.
- [5] 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.
- [6] 장범환, 류제민, 권구형, "노드 중심성을 이용한 효율적 네트워크 토폴로지 시각화 연구", 융합보안논문지, Vol. 21, No. 2, pp. 47-56, 2021.
- [7] 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.
- [8] R. Zafarani, M. A. Abbasi, H. Liu, 'Social Media Mining: An Introduction', Cambridge University Press, 2014.
- [9] CAIDA(Center for Applied Internet Data Analysis), <https://www.caida.org/>, (Accessed Feb. 1, 2021.).
- [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] 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.
- [13] 네트워크 이론 - 다양한 중심성(Centrality) 척도들, <https://bab2min.tistory.com/554>, (Accessed Feb. 1, 2021.).
- [14] 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