



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2017-0113807
(43) 공개일자 2017년10월13일

(51) 국제특허분류(Int. Cl.)
H04L 9/06 (2006.01) H04L 29/06 (2006.01)
H04L 5/00 (2006.01)
(52) CPC특허분류
H04L 9/0637 (2013.01)
H04L 5/0062 (2013.01)
(21) 출원번호 10-2016-0036247
(22) 출원일자 2016년03월25일
심사청구일자 2016년03월25일

(71) 출원인
고려대학교 산학협력단
서울특별시 성북구 안암로 145, 고려대학교 (안암동5가)
(72) 발명자
백상현
서울특별시 서초구 고무래로 35, 109동 1004호(반포동, 반포리채아파트)
고한얼
경기도 고양시 덕양구 화신로 170 2111동 1504호 (행신동, 햇빛마을21단지아파트)
서동은
서울특별시 마포구 도화2길 30 501호 (도화동, 대원빌리지)
(74) 대리인
김동용, 김홍석

전체 청구항 수 : 총 9 항

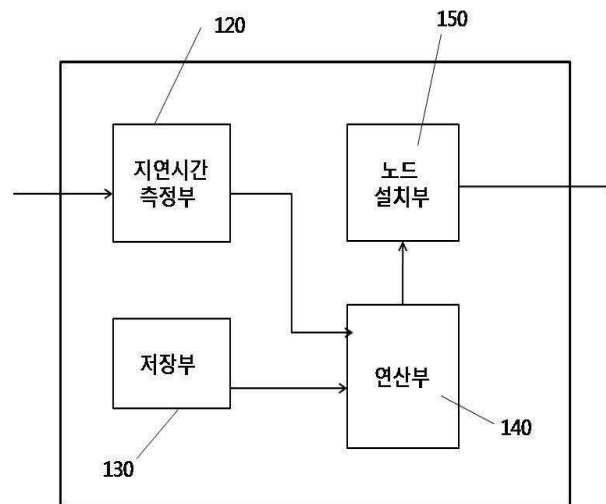
(54) 발명의 명칭 서비스 기능 체이닝 시스템 및 그 방법

(57) 요약

본 발명의 서비스 기능 체이닝 시스템은, 제어 평면 및 데이터 평면을 포함하는 서비스 기능 체이닝 시스템에 있어서, 상기 데이터 평면에 설치된 잉그레스(Ingress node) 노드, SN 노드, SSF(Service Function Forwarder) 노드 및 이그래스(Egress node) 노드 사이의 링크 지연 시간을 측정하는 지연시간 측정부; 상기 각 노드의

(뒷면에 계속)

대표도 - 도2



SFC(Service Function Chaining) 테이블 정보, 각 SN 노드의 자원 용량 정보를 저장하는 저장부; 상기 테이블 정보의 각 체인 길이 및 지연시간 요구량과 상기 측정된 링크 지연시간의 비의 합을 계산하여 최소 값을 갖는 각 SN 노드를 결정하는 연산부; 상기 결정된 각 SN 노드에 상기 해당 SF 인스턴스를 설치하는 노드 설치부를 포함하는 점에 그 특징이 있다.

본 발명은 서비스 기능 체이닝(Service Funtion Chaining) 네트워크에서 각 SF 인스턴스의 배치 장소를 결정하는데 각 체인의 지연시간 요구량과 실제 지연시간의 비를 모든 체인에 대해 합한 값을 최소화하는 SN 노드의 결정하여 트래픽이 해당 체인을 통과하는데 걸리는 지연시간을 줄일 수 있다.

(52) CPC특허분류

H04L 65/40 (2013.01)

H04L 2209/38 (2013.01)

이 발명을 지원한 국가연구개발사업

과제고유번호 1711028339

부처명 미래창조과학부

연구관리전문기관 정보통신산업진흥원

연구사업명 방송통신산업기술개발

연구과제명 글로벌 SDN/NFV 공개소프트웨어 핵심 모듈/기능개발

기 여 율 1/1

주관기관 고려대학교 산학협력단

연구기간 2015.06.01 ~ 2016.05.31

명세서

청구범위

청구항 1

제어 평면 및 데이터 평면을 포함하는 서비스 기능 체이닝 네트워크 시스템에 있어서,

상기 데이터 평면에 설치된 인그레스(Ingress node) 노드, SN 노드, SSF(Service Function Forwarder) 노드 및 이그레스(Egress node) 노드 사이의 링크 지연 시간을 측정하는 지연시간 측정부;

상기 각 노드의 SFC(Service Function Chaining) 테이블 정보, 각 SN 노드의 자원 용량 정보를 저장하는 저장부;

상기 테이블 정보의 각 체인 길이 및 지연시간 요구량과 상기 측정된 링크 지연시간의 비의 합을 계산하여 최소 값을 갖는 각 SN 노드를 결정하는 연산부;

상기 결정된 각 SN 노드에 상기 해당 SF 인스턴스를 설치하는 노드 설치부를 포함하는 서비스 기능 체이닝 시스템.

청구항 2

제1항에 있어서,

상기 테이블 정보는,

트래픽 종류, SF 들의 순서 및 각 체인의 지연시간 요구량을 포함하는 것을 특징으로 하는 서비스 기능 체이닝 시스템.

청구항 3

제1항에 있어서,

상기 연산부는 상기 각 SN 노드의 자원 용량 정보와 설치될 각 SF 인스턴스의 자원 용량 정보를 비교하여 상기 SN 자원 용량보다 SF 인스턴스 자원 용량이 작은지 여부를 판단하여 조건을 충족하는 SN 노드를 결정하는 것을 특징으로 하는 최적 경로 설정을 위한 서비스 기능 체이닝 시스템.

청구항 4

제3항에 있어서,

상기 SN 및 상기 SF 인스턴스 자원 용량 정보는 메모리, CPU 및 저장공간을 포함하는 것을 특징으로 하는 서비스 기능 체이닝 시스템.

청구항 5

제1항에 있어서,

상기 데이터 평면은

상기 제어 평면에서 전달되는 트래픽 플로우 및 경로 결과에 대응하여 플로우를 각 노드에 할당하는 인그레스 노드(Ingress node);

복수 개의 서버가 배치되어 내부에 SF(Service Function) 인스턴스를 가상 머신 형태로 구현시키는 SN 노드

(Service Node);

상기 인그레스 노드에서 할당된 플로우를 상기 SN 노드의 SF 인스턴스에 포워딩하는 SSF(Service Function Forwarder) 노드;

상기 인그레스 노드, 상기 SSF 노드 및 SN 노드를 통과하는 각 경로에 할당된 플로우가 출력되는 이그레스 노드(Egress node)를 포함하는 것을 특징으로 하는 최적 경로 설정을 위한 서비스 기능 체이닝 시스템.

청구항 6

데이터 평면에 설치된 인그레스(Ingress node) 노드, 복수 개의 SN 노드들, SSF(Service Function Forwarder) 노드 및 이그레스(Egress node) 노드 사이의 링크 지연 시간을 측정하는 단계;

상기 SN 노드들의 자원 용량 및 SF 인스턴스들의 자원 필요량을 입력하는 단계;

상기 각 노드에 대한 테이블 정보에서 체인 길이 및 지연시간 요구량을 입력하는 단계;

상기 테이블 정보의 각 체인 길이 및 지연시간 요구량과 상기 측정된 링크 지연시간의 비의 합을 계산하여 최소 값을 갖는 SN 노드를 결정하는 단계; 및

상기 결정된 SN 노드에 해당하는 SF 인스턴스를 설치하는 단계를 포함하는 서비스 기능 체이닝 방법.

청구항 7

제6항에 있어서,

상기 테이블 정보는,

트래픽 종류, SF 들의 순서 및 각 체인의 지연시간 요구량을 포함하는 것을 특징으로 하는 서비스 기능 체이닝 방법.

청구항 8

제6항에 있어서,

상기 SN 노드를 결정하는 단계에서 상기 각 SN 노드의 자원 용량 정보와 설치될 각 SF 인스턴스의 자원 용량 정보를 비교하여 상기 SN 자원 용량보다 SF 인스턴스 자원 필요량이 작은 값을 갖는 조건을 충족시키는 SN 노드를 결정하는 것을 특징으로 하는 서비스 기능 체이닝 방법.

청구항 9

제8항에 있어서,

상기 SN 및 상기 SF 인스턴스 자원 용량 정보는 메모리, CPU 및 저장공간을 포함하는 것을 특징으로 최적 경로 설정을 위한 서비스 기능 체이닝 방법.

발명의 설명

기술 분야

[0001]

본 발명은 서비스 기능 체이닝에서 각 SF 인스턴스의 배치 장소를 결정하는데 각 체인의 지연시간 요구량과 실제 지연시간의 비를 모든 체인에 대해 합한 값을 최소화하는 SN 노드의 결정하여 트래픽이 해당 체인을 통과하는데 걸리는 지연시간을 줄일 수 있는 체이닝 네트워크 시스템 및 그 방법에 관한 것이다.

배경 기술

[0002] 최근 네트워크가 다양한 서비스들을 제공해줌에 따라 점점 보안 유지, 성능 향상 등을 위해 방화벽, deep packet inspection(DPI), network address translation(NAT) 등과 같은 서비스 기능 (SF: service function)에 대한 의존성이 높아지고 있다. 특히, 미래 인터넷 환경에서 다양한 네트워크 서비스를 융합하여 제공하기 위해 기존 하드웨어 기반의 네트워크 서비스 기능들을 소프트웨어적으로 가상화하여 제공하는 Network function virtualization(NFV) 기술을 통해 보다 유연하게 SF를 관리하는 연구가 많이 진행되고 있다.

[0003] 또한, 여러 SF의 사용을 위해 해당 서비스의 트래픽이 논리적인 순서에 따라 사용자에게 전달할 수 있는 합리적인 방법이 필요하다. 즉, 특정 서비스 트래픽의 요구사항에 따라 SF를 하나의 논리적인 연결로 순서화하는 서비스 기능 체이닝 기술(SFC: Service Function Chaining)이 필요하다. SFC는 최근 국제 인터넷 표준화 기구인 IETF WG (Internet engineering task force working group)과 ETSI ISG NFV (European telecommunications standards institute industry specification groups network function virtualization)에서 활발히 표준화가 진행되고 있다.

[0004] 또한, 통신 사업자들 역시, 해당 기술 표준 및 규격을 바탕으로 SFC 프레임워크를 설계하고자 하는 움직임이 활발하게 이루어지고 있다. 하지만, 아직은 요구사항에 대한 효과적인 동적 알고리즘, 아키텍처/프로토콜 구체화 등의 연구는 미비한 상황이다. 특히, 이러한 SFC 프레임워크를 구성하는데 있어 SF 인스턴스나 가상 링크 등의 네트워크 자원 관리는 중요한 이슈 중 하나이다. 이러한 서비스 기능 체인에 포함된 서비스 기능들의 설치 장소에 따라 실제 트래픽이 해당 체인을 통과하는 데에 걸리는 지연시간이 영향을 받게 된다. 따라서 지연시간을 줄이는 서비스 기능 설치 장소의 결정 방법이 필요하다.

발명의 내용

해결하려는 과제

[0005] 본 발명은 서비스 기능 체이닝(Service Funtion Chaining) 네트워크에서 각 SF 인스턴스의 배치 장소를 결정하는데 각 체인의 지연시간 요구량과 실제 지연시간의 비를 모든 체인에 대해 합한 값을 최소화하는 SN 노드의 결정하여 트래픽이 해당 체인을 통과하는데 걸리는 지연시간을 줄일 수 있는 서비스 기능 체이닝 네트워크 시스템 및 그 방법을 제공하는 것을 목적으로 한다.

[0006] 또한, 본 발명은 서비스 기능 체이닝(Service Funtion Chaining) 네트워크에서 서비스 기능들의 CPU, 메모리, 저장 공간 필요량을 고려하여 서비스 노드에 설치된 서비스 기능들의 CPU, 메모리, 저장 공간 필요량의 합이 각 서비스 노드의 CPU 용량, 메모리 용량, 저장 공간 용량을 넘지 않도록 서비스 기능들의 설치 장소를 결정할 수 있는 서비스 기능 체이닝 네트워크 시스템 및 그 방법을 제공하는 것을 목적으로 한다.

[0007]

과제의 해결 수단

[0008] 상기의 과제를 달성하기 위한 본 발명의 일 실시 예에 따른 서비스 기능 체이닝 시스템은, 제어 평면 및 데이터 평면을 포함하는 서비스 기능 체이닝 시스템에 있어서, 상기 데이터 평면에 설치된 잉그레스(Ingress node) 노드, SN 노드, SSF(Service Function Forwarder) 노드 및 이그레스(Egress node) 노드 사이의 링크 지연 시간을 측정하는 지연시간 측정부; 상기 각 노드의 SFC(Service Function Chaining) 테이블 정보, 각 SN 노드의 자원 용량 정보를 저장하는 저장부; 상기 테이블 정보의 각 체인 길이 및 지연시간 요구량과 상기 측정된 링크 지연 시간의 비의 합을 계산하여 최소 값을 갖는 각 SN 노드를 결정하는 연산부; 상기 결정된 각 SN 노드에 상기 해당 SF 인스턴스를 설치하는 노드 설치부를 포함하는 점에 그 특징이 있다.

[0009] 여기서, 특히 상기 테이블 정보는, 트래픽 종류, SF 들의 순서 및 각 체인의 지연시간 요구량을 포함하는 점에 그 특징이 있다.

[0010] 여기서, 특히 상기 연산부는 상기 각 SN 노드의 자원 용량 정보와 설치될 각 SF 인스턴스의 자원 용량 정보를 비교하여 상기 SN 자원 용량보다 SF 인스턴스 자원 용량이 작은지 여부를 판단하여 조건을 충족하는 SN 노드를 결정하는 점에 그 특징이 있다.

[0011] 여기서, 특히 상기 SN 및 상기 SF 인스턴스 자원 용량 정보는 메모리, CPU 및 저장공간을 포함하는 점에 그 특징이 있다.

[0012] 여기서, 특히 상기 데이터 평면은, 상기 제어 평면에서 전달되는 트래픽 플로우 및 경로 결과에 대응하여 플로

우를 각 노드에 할당하는 인그레스 노드(Ingress node); 복수 개의 서버가 배치되어 내부에 SF(Service Function) 인스턴스를 가상 머신 형태로 구현시키는 SN 노드(Service Node); 상기 인그레스 노드에서 할당된 플로우를 상기 SN 노드의 SF 인스턴스에 포워딩하는 SSF(Service Function Forwarder) 노드; 상기 인그레스 노드, 상기 SSF 노드 및 SN 노드를 통과하는 각 경로에 할당된 플로우가 출력되는 이그레스 노드(Egress node)를 포함하는 점에 그 특징이 있다.

[0013] 또한, 상기의 과제를 달성하기 위한 본 발명의 서비스 기능 체이닝 방법은, 데이터 평면에 설치된 인그레스(Ingress node) 노드, 복수 개의 SN 노드들, SSF(Service Function Forwarder) 노드 및 이그레스(Egress node) 노드 사이의 링크 지연 시간을 측정하는 단계; 상기 SN 노드들의 자원 용량 및 SF 인스턴스들의 자원 필요량을 입력하는 단계; 상기 각 노드에 대한 테이블 정보에서 체인 길이 및 지연시간 요구량을 입력하는 단계; 상기 테이블 정보의 각 체인 길이 및 지연시간 요구량과 상기 측정된 링크 지연시간의 비의 합을 계산하여 최소 값을 갖는 SN 노드를 결정하는 단계; 및 상기 결정된 SN 노드에 해당하는 SF 인스턴스를 설치하는 단계를 포함하는 점에 그 특징이 있다.

[0014] 여기서, 특히 상기 테이블 정보는, 트래픽 종류, SF 들의 순서 및 각 체인의 지연시간 요구량을 포함하는 점에 그 특징이 있다.

[0015] 여기서, 특히 상기 SN 노드를 결정하는 단계에서 상기 각 SN 노드의 자원 용량 정보와 설치될 각 SF 인스턴스의 자원 용량 정보를 비교하여 상기 SN 자원 용량보다 SF 인스턴스 자원 필요량이 작은 값을 갖는 조건을 충족시키는 SN 노드를 결정하는 점에 그 특징이 있다.

[0016] 여기서, 특히 상기 SN 및 상기 SF 인스턴스 자원 용량 정보는 메모리, CPU 및 저장공간을 포함하는 점에 그 특징이 있다.

발명의 효과

[0017] 본 발명의 효과는 다음과 같다.

[0018] 첫째, 본 발명은 서비스 기능 체이닝(Service Funtion Chaining) 네트워크에서 각 SF 인스턴스의 배치 장소를 결정하는데 각 체인의 지연시간 요구량과 실제 지연시간의 비를 모든 체인에 대해 합한 값을 최소화하는 SN 노드의 결정하여 트래픽이 해당 체인을 통과하는데 걸리는 지연시간을 줄일 수 있다.

[0019] 둘째, 본 발명은 서비스 기능 체이닝(Service Funtion Chaining) 네트워크에서 서비스 기능들의 CPU, 메모리, 저장 공간 필요량을 고려하여 서비스 노드에 설치된 서비스 기능들의 CPU, 메모리, 저장 공간 필요량의 합이 각각 서비스 노드의 CPU 용량, 메모리 용량, 저장 공간 용량을 넘지 않도록 서비스 기능들의 설치 장소를 결정할 수 있다.

도면의 간단한 설명

[0020] 도 1은 본 발명의 일 실시 예에 따른 SFC 네트워크 토폴로지의 서비스 기능 체이닝 시스템의 구성을 개략적으로 도시한 도면.

도 2는 상기 도 1의 제어 평면의 구성을 개략적으로 도시한 도면

도 3은 본 발명의 일 실시 예에 따른 서비스 기능 체이닝 방법에 대한 순서도.

도 4는 본 발명의 SN 간의 지연시간의 범위를 변화시키면서 목적함수 F의 변화를 살펴본 그래프.

도 5 및 도 6은 본 발명에 따른 지연시간 요구량에 따른 각 서비스 체인의 지연시간과 불만족도를 나타내는 도면.

발명을 실시하기 위한 구체적인 내용

[0021] 본 발명은 다양한 변경을 가할 수 있고 여러 가지 실시 예를 가질 수 있는 바, 특정 실시 예들을 도면에 예시하고 이를 상세한 설명을 통해 상세히 설명하고자 한다. 그러나 이는 본 발명을 특정한 실시 형태에 대해 한정하려는 것이 아니며, 본 발명의 사상 및 기술 범위에 포함되는 모든 변경, 균등물 내지 대체물을 포함하는 것으로 이해되어야 한다.

[0022] 본 발명을 설명함에 있어서, 관련된 공지 기술에 대한 구체적인 설명이 본 발명의 요지를 불필요하게 흐릴 수 있다고 판단되는 경우 그 상세한 설명을 생략한다. 또한, 본 명세서의 설명 과정에서 이용되는 숫자(예를 들어,

제1, 제2 등)는 하나의 구성요소를 다른 구성요소와 구분하기 위한 식별기호에 불과하다.

- [0023] 또한, 본 명세서에서, 일 구성요소가 다른 구성요소와 "연결된다" 거나 "접속된다" 등으로 언급된 때에는, 상기 일 구성요소가 상기 다른 구성요소와 직접 연결되거나 또는 직접 접속될 수도 있지만, 특별히 반대되는 기재가 존재하지 않는 이상, 중간에 또 다른 구성요소를 매개하여 연결되거나 또는 접속될 수도 있다고 이해되어야 할 것이다.
- [0024] 이하, 첨부된 도면들을 참조하여 본 발명의 가상환경의 스토리지 성능 향상을 위한 클라우드 스토리지 시스템 및 그 관리 방법에 관하여 상세히 설명한다.
- [0025] 도 1은 본 발명의 일 실시 예에 따른 SFC 네트워크 토폴로지의 서비스 기능 체이닝 시스템의 구성을 개략적으로 도시한 도면이고, 도 2는 상기 도 1의 제어 평면의 구성을 개략적으로 도시한 도면이다.
- [0026] 도 1에 도시된 바와 같이, 본 발명의 일 실시 예에 따른 서비스 기능 체이닝 시스템은 크게 제어 평면(100) 및 데이터 평면(200)으로 나뉜다. 먼저, 데이터 평면(200)에 대해 설명하기로 한다.
- [0027] 상기 데이터 평면(200)은 상기 제어 평면(100)에서 전달되는 트래픽 플로우 및 경로 결과에 대응하여 플로우를 각 노드에 할당하는 인그레스 노드(Ingress node)(210); 복수 개의 서버가 배치되어 내부에 SF(Service Function) 인스턴스를 가상 머신(230) 형태로 구현시키는 SN 노드(Service Node)(220); 상기 인그레스 노드에서 할당된 플로우를 상기 SN 노드(220)의 SF 인스턴스에 포워딩하는 SSF(Service Function Forwarder) 노드(240); 상기 인그레스 노드(210), SN 노드(220) 및 상기 SSF 노드(240)를 통과하는 각 경로에 할당된 플로우가 출력되는 이그레스 노드(Egress node)(250)를 포함하여 구성된다.
- [0028] 상기 인그레스 노드(Ingress node)(210) 및 이그레스 노드(Egress node)(220)는 각각 SFC 도메인으로 유입되는 패킷들을 미리 정의된 네트워크 정책에 따라 분류하는 역할, 그리고 SFC 도메인 내에서 처리가 완료된 패킷들을 SFC 도메인 외부로 전달하는 역할을 수행한다.
- [0029] 상기 SN(Service Node) 노드(220)는 CPU, 메모리, 저장 공간을 제공하는 물리적인 혹은 가상의 요소로, NFV(Network Function Virtualization) 기술을 통해 VM(Virtual Machine) 형태로 하나 혹은 그 이상의 SF(Service Function) instance들을 호스팅할 수 있는 기능을 제공한다.
- [0030] 상기 SFF(Service Function Forwarder) 노드(240)는 전달받은 패킷들을 정의된 체인에 따라 SN 노드(220) 혹은 SFF 노드(240)에게 전달하는 역할을 수행한다.
- [0031] 도 2에 도시된 바와 같이, 상기 제어 평면(100)은 지연시간 측정부(120), 저장부(130), 연산부(140) 및 노드 설치부(150)를 포함하여 구성된다.
- [0032] 상기 지연시간 측정부(120)는 상기 데이터 평면(200)에 설치된 인그레스(Ingress node) 노드(210), SN 노드(220), SSF(Service Function Forwarder) 노드(240) 및 이그레스(Egress node) 노드(250) 사이의 링크 지연 시간을 측정하게 된다.
- [0033] 보다 구체적으로, 각 노드에서 자신의 인접 노드들에게 주기적인 탐사 패킷을 전송하고, 탐사 패킷을 수신한 인접 노드는 이에 대한 응답 패킷을 전송하게 된다. 각 노드는 탐사 패킷 전송 시점부터 해당 탐사 패킷에 대한 응답 시점까지의 시간을 모든 인접 노드들에 대해 측정하고, 해당 정보를 상기 지연시간 측정부(120)에게 주기적으로 보고하게 된다. 그리고 상기 지연시간 측정부(120)는 해당 정보를 기반으로 모든 링크에 대한 지연시간 정보를 유지하게 된다.
- [0034] 상기 저장부(130)는 상기 각 노드의 SFC(Service Function Chaining) 테이블 정보(110), 각 SN 노드(220)의 자원 용량 정보를 저장한다. 상기 테이블 정보(110)는, 트래픽 종류, SF 들의 순서 및 각 체인의 지연시간 요구량을 포함한다.
- [0035] 상기 저장부(130)는 SFC 테이블을 관리한다. 테이블 엔트리는 트래픽 종류, SF들의 순서, 체인의 지연시간 요구량으로 구성된다. 예를 들어, chain 1의 경우, 트래픽 종류가 HTTP로 명시되어 있고, SF 1 - SF 2 - SF 3의 순서가 정의되어 있다. 이는 SFC 도메인 내로 유입되는 HTTP 트래픽은 chain 1에 정의된 순서대로 처리가 되어야 함을 의미한다. 한편, chain 1에 대한 지연시간 요구량은 19 단위 시간으로, 이는 chain 1에 따라 처리되는 패킷들이 19 단위 시간 내에 처리되어야 함을 나타낸다.
- [0036] 상기 연산부(140)는 상기 테이블 정보(110)의 각 체인 길이 및 지연시간 요구량과 상기 측정된 링크 지연시간의 비의 합을 계산하여 최소 값을 갖는 각 SN 노드(220)를 결정하게 된다.

[0037] 보다 구체적으로, 상기 연산부(140)는 상기 SFC의 지연 시간 요구량을 고려하여 SF 인스턴스(instance)들이 설치될 장소, 즉, SN 들을 결정하는 문제를 수학적 도구인 정수 비선형 프로그램인 INLP(Integer Non-Linear Programming)를 활용하여 모델링을 하게 되고, 최적화를 하고자 하는 목적 함수, 그리고 제한 조건들을 정의한다.

[0038] 먼저, 서비스 성격에 따라 각 SFC는 각기 다른 지연시간 요구량을 가진다. 예를 들어, 지연시간에 민감한 서비스 (예, 멀티미디어, VoIP 서비스)들은 빠른 지연시간 요구량을 가지고, 이와 반대로 파일 다운로드 서비스와 같은 경우 느슨한 지연시간 요구량을 가진다. 이를 반영하여 사용할 목적 함수를 정의하게 된다.

[0039] 우선, 목적 함수를 정의하기 위해 다음과 같은 변수들을 정의한다. 체인 c의 k번째 SF 인스턴스(instance)가 SN i에 설치됐는지에 대한 여부를 나타내는 이진 변수를 $x_i^{c,k}$ 라 한다. 한편, l_{si} 와 l_{ie} 는 잉레스 노드(ingress node)와 SN i사이의 가상 링크 지연시간과 SN i와 egress node사이의 지연시간을 각각 나타낸다. 또한, l_{ij} 는 SN i와 SN j사이의 지연시간을 나타낸다. 이러한 변수들을 통해 RSP의 전체 지연시간을 나타내면 하기 수식 1과 같이 나타낼 수 있다.

[0040] 수식 1

$$\sum_i l_{si} x_i^{c,1} + \sum_k \sum_i \sum_j l_{ij} x_i^{c,k} x_j^{c,k+1} + \sum_i l_{ie} x_i^{c,K_c}$$

[0041]

[0042] 여기서 K_c 는 chain c에서 필요한 SF의 수를 나타낸다.

[0043] 그리고 각 체인의 지연시간 요구량과 실제 지연시간의 비의 합을 최소화하는 것이기 때문에 목적함수를 다음 수식 2와 같이 정의할 수 있다.

[0044] 수식 2

$$F = \sum_c \frac{\sum_i l_{si} x_i^{c,1} + \sum_k \sum_i \sum_j l_{ij} x_i^{c,k} x_j^{c,k+1} + \sum_i l_{ie} x_i^{c,K_c}}{L_c}$$

[0045]

[0046] 위 식에서 L_c 는 chain c의 지연시간 요구량을 나타낸다.

[0047] 또한, 상기 연산부(140)는 상기 각 SN 노드(220)의 자원 용량 정보와 설치될 각 SF 인스턴스의 자원 용량 정보를 비교하여 상기 SN 자원 용량보다 SF 인스턴스 자원 용량이 작은지 여부를 판단하여 조건을 충족하는 SN 노드를 결정하게 된다. 여기서, 상기 SN 및 상기 SF 인스턴스 자원 용량 정보는 메모리, CPU 및 저장공간을 포함한다.

[0048] 즉, 상기 연산부(140)는 정수 비선형 프로그램인 INLP(Integer Non-Linear Programming)에 의해 제약조건을 정의한다. SF를 SN에 설치하는데 있어서 설치된 SF들이 사용하는 총 자원의 양이 SN이 가지고 있는 자원보다 작아야 한다.

[0049] 다시 말해, 체인 c의 k번째 SF의 메모리 필요량을 $m_{c,k}$ SN i의 전체 메모리량을 M_i 로 정의할 때, 다음과 같은 수식 3의 제약조건을 만족해야 한다.

[0050] 수식 3

$$\sum_c \sum_k m_{c,k} x_i^{c,k} \leq M_i, \text{ for } \forall i.$$

[0051]

[0052] 또한, 이와 유사하게 CPU 용량과 저장 공간에 대한 제약조건은 수식 4 및 수식 5와 같이 나타낼 수 있다.

[0053] 수식 4

$$\sum_c \sum_k c_{c,k} x_i^{c,k} \leq C_i, \text{ for } \forall i,$$

[0054]

[0055] 수식 5

$$\sum_c \sum_k s_{c,k} x_i^{c,k} \leq S_i, \text{ for } \forall i,$$

[0056]

[0057] 여기서 $c_{c,k}$ 와 $s_{c,k}$ 는 체인 c의 k번째 SF의 CPU 필요량과 저장 공간 필요량을 나타내고, C_i 와 S_i 는 SN i의 전체 CPU 용량과 저장 공간을 의미한다.

[0058] 상기 노드 설치부(150)는 상기 결정된 각 SN 노드(220)에 상기 해당 SF 인스턴스를 설치하게 된다.

[0059] 여기서, 상기 노드 설치부(150)는 체인의 지연 시간 요구량 및 네트워크 환경(링크 지연시간, SN의 가용 자원)을 고려하여 SF 인스턴스(instance)들을 적합한 SN들에게 위치시킨다. SF 인스턴스들이 SN들에게 설치되고 나면 패킷들은 체인에 정의된 실제 방문 순서인 RSP(Rendered Service Path)에 따라 전달된다.

[0060] 상기 노드 설치부(150)는 각 SF 인스턴스(instance)는 하나의 SN에 설치되어야 하기 때문에 다음 수식 6과 같은 제약조건을 가지고 설치될 노드를 결정하게 된다.

[0061] 수식 6

$$\sum_i x_i^{c,k} = 1, \text{ for } \forall c, \forall k.$$

[0062]

[0063] 또한, $x_i^{c,k}$ 는 이진 변수이기 때문에 수식 7과 같은 또 다른 제약조건을 만족해야 한다.

[0064] 수식 7

$$x_i^{c,k} \in \{0, 1\}.$$

[0065]

[0066] 또한, 도 3은 본 발명의 일 실시 예에 따른 서비스 기능 체이닝 방법에 대한 순서도이다.

[0067] 도 3에 도시된 바와 같이, 본 발명의 서비스 기능 체이닝 방법은, 먼저, 데이터 평면에 설치된 인그레스(Ingress node) 노드, 복수 개의 SN 노드들, SSF(Service Function Forwarder) 노드 및 이그래스(Egress node)

노드 사이의 링크 지연 시간을 측정하는 단계가 수행된다(S301).

- [0068] 보다 구체적으로, 각 노드에서 자신의 인접 노드들에게 주기적인 탐사 패킷을 전송하고, 탐사 패킷을 수신한 인접 노드는 이에 대한 응답 패킷을 전송하게 된다. 각 노드는 탐사 패킷 전송 시점부터 해당 탐사 패킷에 대한 응답 시점까지의 시간을 모든 인접 노드들에 대해 측정하고, 해당 정보를 상기 지연시간 측정부(120)에게 주기적으로 보고하게 된다. 그리고 상기 지연시간 측정부(120)는 해당 정보를 기반으로 모든 링크에 대한 지연시간 정보를 유지하게 된다.
- [0069] 이어서, 상기 저장부(130)에 저장된 상기 SN 노드들의 자원 용량 및 SF 인스턴스들의 자원 필요량을 입력하는 단계가 수행된다(S302). 여기서, 각 SN 노드 및 상기 SF 인스턴스 자원 용량 정보는 메모리, CPU 및 저장공간을 포함한다.
- [0070] 이어서, 상기 각 노드에 대한 테이블 정보에서 체인 길이 및 지연시간 요구량을 입력하는 단계가 수행된다(S303). 상기 저장부에 각 노드의 SFC(Service Function Chaining) 테이블 정보(110) 및 SN 노드(220)의 자원 용량 정보가 저장된 값을 상기 연산부에 입력받게 된다. 상기 테이블 정보(110)는, 트래픽 종류, SF 들의 순서 및 각 체인의 지연시간 요구량을 포함한다. 상기 저장부(130)는 SFC 테이블을 관리한다. 테이블 엔트리는 트래픽 종류, SF들의 순서, 체인의 지연시간 요구량으로 구성된다. 예를 들어, chain 1의 경우, 트래픽 종류가 HTTP로 명시되어 있고, SF 1 - SF 2 - SF 3의 순서가 정의되어 있다. 이는 SFC 도메인 내로 유입되는 HTTP 트래픽은 chain 1에 정의된 순서대로 처리가 되어야 함을 의미한다. 한편, chain 1에 대한 지연시간 요구량은 19 단위 시간으로, 이는 chain 1에 따라 처리되는 패킷들이 19 단위 시간 내에 처리되어야 함을 나타낸다.
- [0071] 그 다음, 상기 테이블 정보의 각 체인 길이 및 지연시간 요구량과 상기 측정된 링크 지연시간의 비의 합을 계산하여 최소 값을 갖는 SN 노드를 결정하는 단계가 수행된다(S304).
- [0072] 보다 구체적으로, 상기 연산부(140)에서 상기 SFC의 지연 시간 요구량을 고려하여 SF 인스턴스(instance)들이 설치될 장소, 즉, SN 들을 결정하는 문제를 수학적 도구인 정수 비선형 프로그램인 INLP(Integer Non-Linear Programming)를 활용하여 모델링을 하게 되고, 최적화를 하고자 하는 목적 함수, 그리고 제한 조건들을 정의한다.
- [0073] 먼저, 서비스 성격에 따라 각 SFC는 각기 다른 지연시간 요구량을 가진다. 예를 들어, 지연시간에 민감한 서비스 (예, 멀티미디어, VoIP 서비스)들은 빠른 지연시간 요구량을 가지고, 이와 반대로 파일 다운로드 서비스와 같은 경우 느슨한 지연시간 요구량을 가진다. 이를 반영하여 사용할 목적 함수를 정의하게 된다. 여기서, 상기 수식 1 내지 수식 7을 참조하여 자세한 설명은 생략하기로 한다.
- [0074] 또한, 상기 연산부(140)는 상기 각 SN 노드(220)의 자원 용량 정보와 설치될 각 SF 인스턴스의 자원 용량 정보를 비교하여 상기 SN 자원 용량보다 SF 인스턴스 자원 용량이 작은지 여부를 판단하여 조건을 충족하는 SN 노드를 결정하게 되는데 SF들이 사용하는 총 자원의 양이 SN이 가지고 있는 자원보다 작아야 한다. 여기서, 상기 SN 및 상기 SF 인스턴스 자원 용량 정보는 메모리, CPU 및 저장공간을 포함한다.
- [0075] 마지막으로, 상기 결정된 SN 노드에 해당하는 SF 인스턴스를 설치하는 단계가 수행된다(S305). 상기 노드 설치부(150)에서 체인의 지연 시간 요구량 및 네트워크 환경(링크 지연시간, SN의 가용 자원)을 고려하여 SF 인스턴스(instance)들을 적합한 SN들에게 위치시킨다. SF 인스턴스들이 SN들에게 설치되고 나면 패킷들은 체인에 정의된 실제 방문 순서인 RSP(Rendered Service Path)에 따라 전달된다.
- [0076] 한편, 도 4는 본 발명의 SN 간의 지연시간의 범위를 변화시키면서 목적함수 F의 변화를 살펴본 그래프이고, 도 5 및 도 6은 본 발명에 따른 지연시간 요구량에 따른 각 서비스 체인의 지연시간과 불만족도를 나타내는 도면이다.
- [0077] 도 4 내지 도 6에서 S_{opt} , S_{rand} , 그리고 S_{minSN} 은 각각 본 발명의 배치 결과에 따른 목적함수 값, SF 인스턴스들을 SN들에게 임의로 배치하는 비교 기법의 배치 결과에 따른 목적함수 값, 그리고 SF 인스턴스들의 배치에 사용되는 SN의 수를 최소로 하는 것을 목적으로 하는 비교 기법의 배치 결과에 따른 목적함수 값을 나타낸다. 한편, 동일한 조건의 비교를 위해 모든 배치 기법들의 배치 결과는 수식 3, 수식 4, 수식 5, 수식 6, 수식 7의 제약 조건들을 만족하도록 한다.
- [0078] 도 4에 도시된 바와 같이, 본 발명의 지연시간이 가장 낮은 것을 확인할 수 있다. 이는 본 발명이 각 서비스 체인의 지연시간 요구를 고려하여 SF이 설치될 SN을 선택하기 때문이다.
- [0079] 또한, 지연시간의 범위가 증가함에 따라 다른 방법들의 목적함수 값과 본 발명의 목적함수의 값의 차이가 점차

증가하는 것을 확인할 수 있다. 즉, 네트워크 토폴로지가 커짐에 따라 본 발명의 효율이 좋아지는 것을 의미한다. 이에 대한 이유는 다음과 같다. 네트워크 토폴로지가 커지게 되면 구성된 서비스체인이 지연시간 요구를 만족하지 못할 가능성이 커지게 된다. 하지만, 본 발명에서는 지연시간 요구가 가장 높은 체인부터 낮은 지연시간을 보장할 수 있는 SN에 설치를 하기 때문에 지연시간 요구가 높다 하더라도 이를 만족할 가능성을 높힐 수 있다.

[0080] 도 5를 참조하면, 본 발명은 지연시간 요구가 가장 뽁뽁한 체인 3 ($L_3=10$)부터 설치하기 때문에 해당 체인의 지연시간이 가장 작다. 한편, 다른 방법들은 이를 고려하지 않고 SF를 설치하기 때문에 지연시간 요구가 가장 뽁뽁한 체인 3의 지연시간이 다른 체인의 지연시간보다 커지는 경우가 발생한다.

[0081] 또한, 도 6에 도시된 바와 같이, 지연시간 요구가 뽁뽁해짐에 따라 모든 방법에서 불만족도가 올라가는 것을 확인할 수 있다. 이를 해당 요구사항을 만족시키는 것이 점차 어려워지기 때문이다. 특히, 종래의 방법에서 체인 3의 불만족도를 살펴보면 1보다 큰 것을 확인할 수 있다. 이는 실제 지연시간이 요구되는 지연시간보다 큰 것을 의미한다. 이 경우, 해당 체인을 사용하는 유저는 낮은 QoE를 얻게 된다.

[0082] 따라서, 본 발명에 의하면, 서비스 기능 체이닝(Service Funtion Chaining) 네트워크에서 각 SF 인스턴스의 배치 장소를 결정하는데 각 체인의 지연시간 요구량과 실제 지연시간의 비를 모든 체인에 대해 합한 값을 최소화하는 SN 노드의 결정하여 트래픽이 해당 체인을 통과하는데 걸리는 지연시간을 줄일 수 있다.

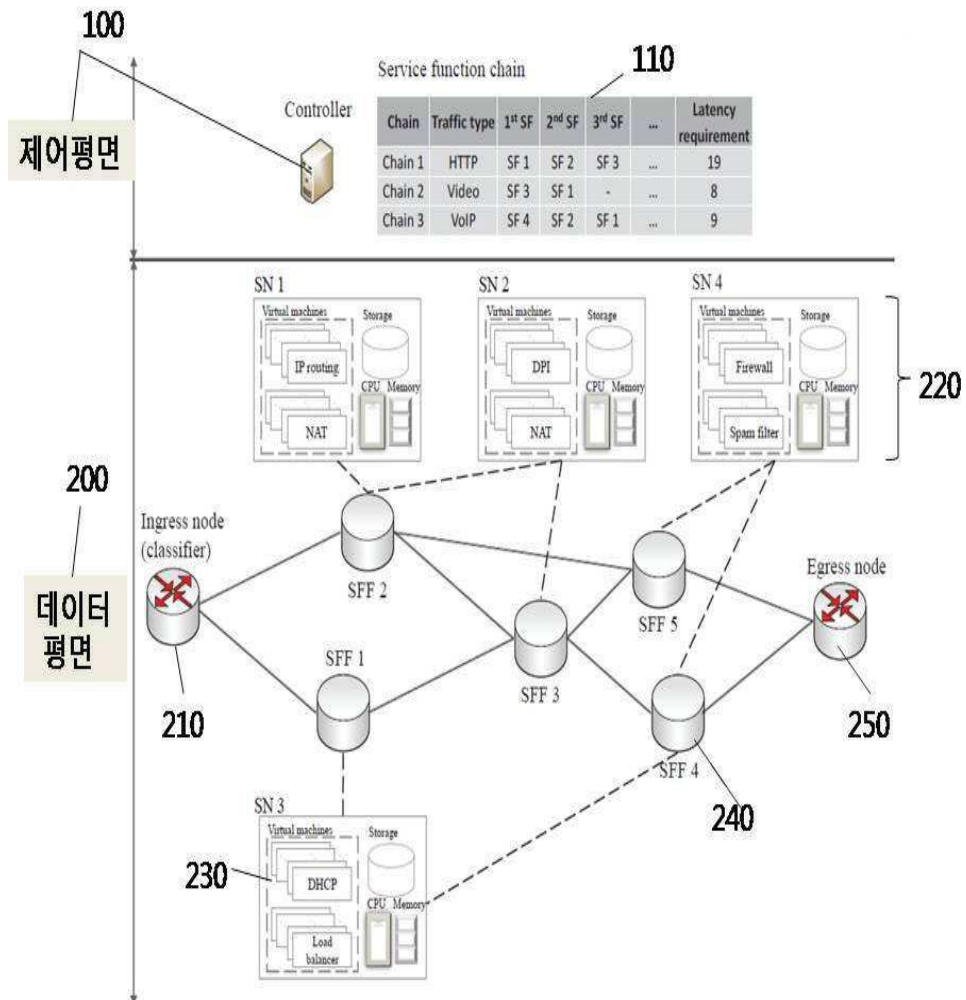
[0083] 본 발명의 권리범위는 상술한 실시예에 한정되는 것이 아니라 첨부된 특허청구범위내에서 다양한 형태의 실시예로 구현될 수 있다. 특허청구범위에서 청구하는 본 발명의 요지를 벗어남이 없이 당해 발명이 속하는 기술분야에서 통상의 지식을 가진자라면 누구든지 변형 가능한 다양한 범위까지 본 발명의 청구범위 기재의 범위 내에 있는 것으로 본다.

부호의 설명

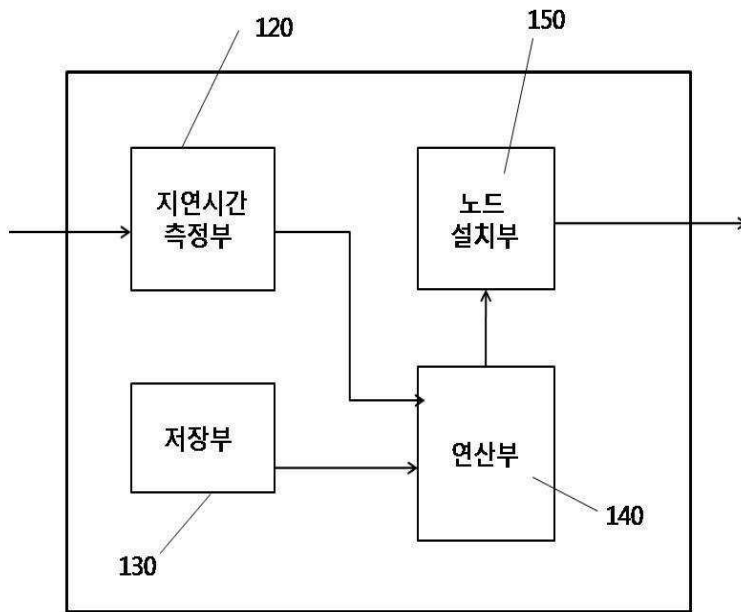
[0084]	100 --- 제어 평면	110 --- 테이블
	120 --- 경로 계산부	130 --- 조건 판단부
	140 --- 저장부	200 --- 데이터 평면
	210 --- 이그레스 노드	220 --- SN 노드
	230 --- SFF 노드	240 --- SF 노드
	250 --- 이그레스 노드	

도면

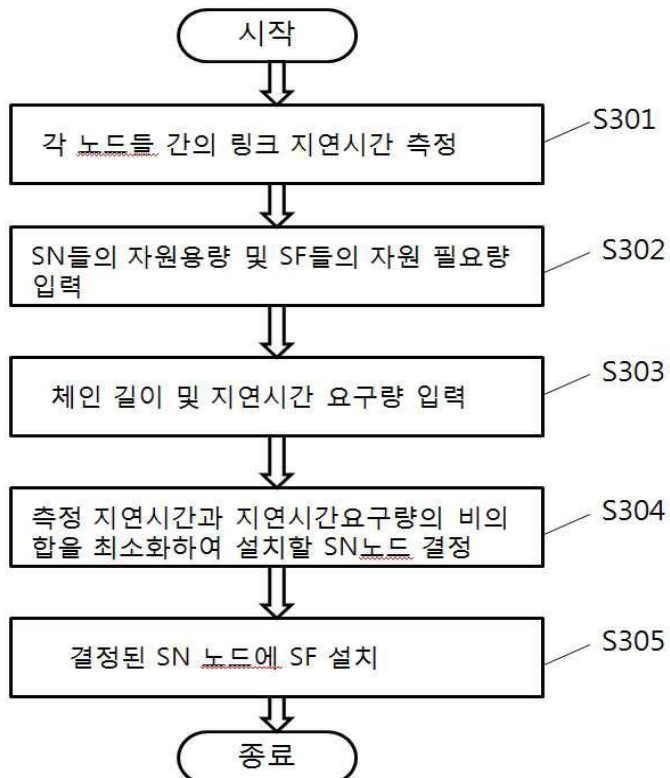
도면1



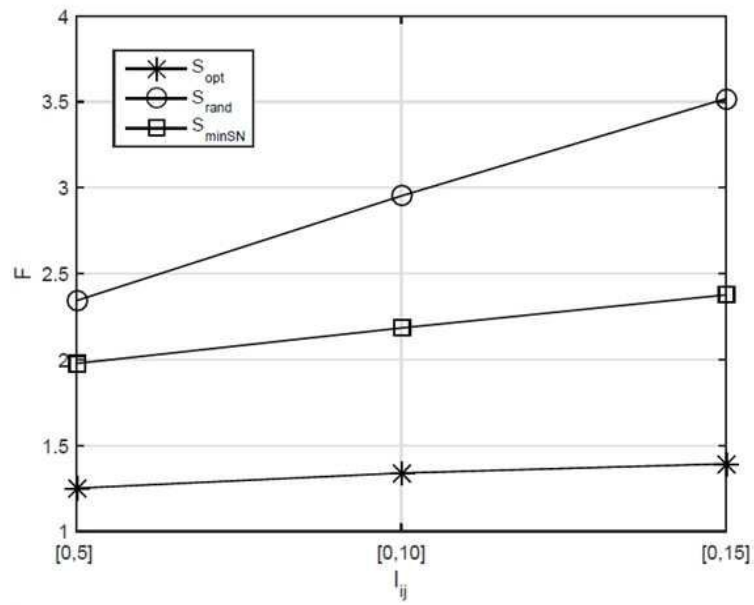
도면2



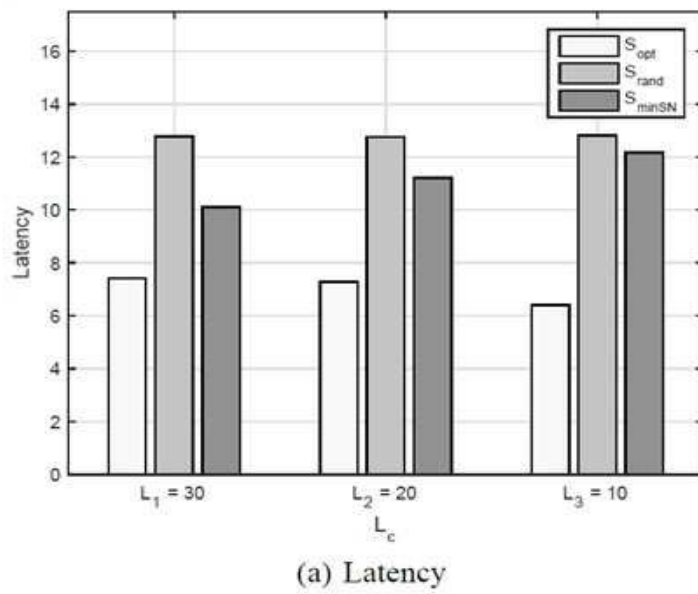
도면3



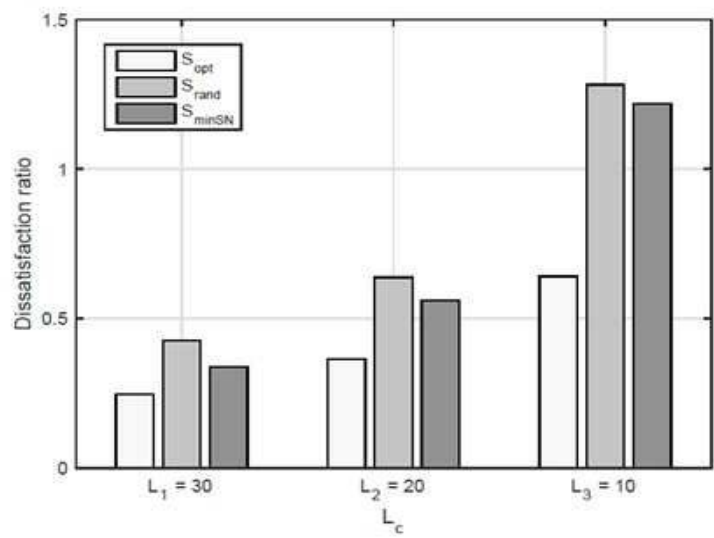
도면4



도면5



도면6



(b) Dissatisfaction ratio