

LLM 기반 초기 해집단 생성을 활용한 서비스 배치 및 트래픽 라우팅 최적화

이 도 영[◦]

Service Placement and Traffic Routing Optimization Using LLM-Generated Initial Population

Doyoung Lee[◦]

요 약

6G 네트워크 시대를 준비하기 위해서는 5G에서 정의한 성능 기준치의 상향 뿐만 아니라, 새롭게 등장할 서비스나 시나리오 요구사항에 맞춘 네트워크 측면의 발전이 요구되고 있다. 특히, 차세대 네트워크에서는 엄격한 서비스 요구사항과 더불어 서비스를 제공함에 있어서 고려되어야 하는 추가적인 제약 사항들로 인해 기존 네트워크에서 운용되는 서비스 배치 및 트래픽 라우팅 기법의 개선을 추진하고 있다. 따라서, 동적인 6G 네트워크 최적화를 위한 연구들이 진행되고 있으며, 그 중 메타휴리스틱 기반 기법은 최적 근사해를 도출하는데 효과적인 방법으로 주목받고 있다. 하지만, 메타휴리스틱 기반 최적화는 초기 해의 구성에 따라 최종 해의 적합도 또는 수렴 시간에 큰 영향을 준다는 한계가 존재한다. 본 논문에서는 이를 극복하기 위하여, 기존 초기 해 임의 생성 방식 대신 LLM을 활용하여 네트워크 상황 및 서비스 요구사항에 맞는 초기 해집단을 생성하는 방식을 제안한다.

Key Words : 6G, Routing optimization, Large Language Models, Metaheuristic algorithm

ABSTRACT

To prepare for the upcoming 6G era beyond 5G, network technologies must evolve not only to achieve enhanced performance targets but also to support emerging services and communication scenarios. In particular, next-generation networks demand improvements in existing traffic routing and service placement techniques under increasingly stringent service requirements and additional operational constraints. Accordingly, metaheuristic-based optimization techniques have attracted attention as effective approaches for deriving near-optimal solutions in dynamic 6G environments. However, their performance and convergence speed are highly dependent on the quality of the initial solution population. To address this limitation, this paper proposes an LLM-based approach that generates high-quality initial solutions according to network conditions and service requirements, replacing conventional random initialization methods.

[◦] First Author and Corresponding Author : Kongju National University, dylee@kongju.ac.kr

I. 서 론

정보통신기술의 발전은 5G 네트워크 시대를 성공적으로 실현할 수 있었던 원동력이 되어왔으며, 이제는 다가오는 6G 네트워크 시대의 기술 구현을 목표로 하고 있다. 이와 함께, 6G 네트워크에서는 5G에서 정의된 용례들(Use-cases)보다 엄격한 성능 요구사항을 가지는 다양한 서비스 시나리오를 제안하고 있다 [1]. 따라서, 6G의 기술적 요구사항을 충족하는 네트워크 구현을 위해서는 지능적이고 유연한 네트워크 관리 및 운용 기술이 필수적이다. 또한, 5G를 넘어서 6G로 향하는 시대적 흐름에서는 인공지능(AI, Artificial Intelligence) 기술을 부가적으로 활용하는 것이 아니라, 네트워크 전 영역에 탑재된 형태로 적극적으로 활용하는 AI 내재형 네트워크(AI-native networks)의 실현이 필요하다 [2].

AI 내재형 네트워크에서는 6G에서 언급되는 초대규모 IoT, 초저지연 서비스, 위성-지상 통합 네트워크 등을 지원하기 위한 최적화 기법의 필요성이 강조되고 있다 [3, 4]. 특히, AI 모델의 학습 및 추론 성능 개선, 지속적인 네트워크 상태 관측, 데이터 수집 및 전처리 등의 작업이 네트워크 운용과 함께 병행되면서 네트워크 운용 및 관리를 위한 비용이 급증할 가능성이 존재한다. 이와 함께, 최근 환경 문제에 대한 관심은 5G와 6G에서는 네트워크 운용 및 서비스 제공 과정에서 발생하는 탄소 배출량을 감축하는 것도 주요 목표로 다루어지도록 하고 있다 [5, 6].

상기 서술한 기술적인 필요성과 서비스 요구사항들은 기존 네트워크 운용 및 관리 기술의 개선을 촉진시키는 원인이 되었다. 예시로, 엄격한 서비스 성능 요구사항, 즉, QoS (Quality-of-Service)를 충족함과 동시에 운용 비용과 탄소 배출량을 동시에 최소화하는 것은 서비스 배치 문제, 트래픽 라우팅 문제 등을 NP-Hard 또는 조합 최적화 문제로 진화시켰다. 한편, AI 기술은 네트워크 내 존재하는 여러 최적화 문제를 위한 해결 방안으로 활용될 수 있지만 [7], 다목적 최적화 문제들을 해결하는 과정에서 최적해를 보장할 수가 없다는 한계가 존재한다 [8]. 따라서 이러한 문제를 효과적으로 해결하기 위해 네트워크 최적화를 위한 연구가 수행되고 있으며, 그 중에서 메타휴리스틱(Metaheuristic) 기반 방식은 허용 가능한 시간 내 최적 근사해를 도출할 수 있는 특성으로 인해 주목받고 있다.

일반적으로 메타휴리스틱 기반 최적화 기법에서는 우선 초기 해를 생성한 후, 각 해에 대한 평가와 새로운 해 탐색, 해 개선 등을 종료 조건에 도달할 때까지 반복하며 최적 근사해를 도출한다 [9]. 즉, 초기 해집단을 구성하는 단계부터 시작하기 때문에 초기 해들의 분포에 따라 최종적으로 산출되는 해의 적합도(Fitness)나 수렴 시간에 영향을 받을 수 있다. 따라서, 해 공간을 효율적으로 탐색함과 동시에 한정된 시간 내 가장 높은 적합도를 가진 해를 도출하기 위해서는 다양성과 품질을 고려한 초기 해들을 생성하는 것이 중요하다. 하지만, 대부분의 메타휴리스틱 기반 최적화 기법에서는 임의로 초기 해집단을 구성하는 방법을 사용하고 있다는 한계가 존재한다.

본 논문에서는 이와 같은 문제를 해결하기 위해 대형 언어 모델(LLM, Large Language Model)-을 초기 해 생성에 사용하는 메타휴리스틱 기반 최적화 기법을 제안한다. 메타휴리스틱 알고리즘으로는 유전 알고리즘을 활용하며, 이는 동적인 네트워크 환경에서 운용 비용을 최소화하고 서비스 요구사항을 만족하는 다목적 최적화 문제인 네트워크 서비스 배치 및 트래픽 라우팅에 적용된다. 또한, 일반적으로 특정 용도를 위한 LLM을 확보하기 위해 수행되는 미세조정(Fine-tuning)은 큰 연산 비용이 요구되기 때문에, 본 논문에서 제안하는 방법은 초기 해 생성을 위해 맥락 내 학습(ICL, In-Context Learning) 기법을 활용한다.

본 논문의 나머지 내용은 다음과 같이 구성된다. 2장에서는 네트워크 서비스 배치와 트래픽 라우팅 관련된 연구 및 기술 동향에 대해 서술한다. 3장에서는 서비스 배치 및 트래픽 라우팅을 다루기 위한 시스템 모델을 정의하고, 이어지는 4장에서는 제안하는 LLM 기반 초기 해 생성 방법을 적용한 유전 알고리즘 기반 서비스 배치 및 트래픽 라우팅 방법에 대해 서술한다. 5장에서는 구현된 최적화 기능에 대한 성능 평가를 수행하며, 마지막 6장에서는 논문 내용을 정리하고 향후 연구 방향에 대해 논의한다.

II. 연구 배경 및 관련 연구

네트워크 운용 및 관리 기술 중, 서비스 배치는 사용자에게 제공되어야 하는 네트워크 서비스를 클라우드, 엣지(Edge) 등 물리적으로 어떤 위치에 배포되어야 하는지 결정하는 최적화 기술이다. 서비스 배치에서는 사용자에게 충족되어야 하는 서비스별

QoS 및 배치, 운용 비용 등을 고려하여 효율적인 배치가 될 수 있도록 하는 것이 중요하다.

네트워크 서비스 배치 문제는 대표적인 NP-Hard 문제이며, 이를 해결하기 위한 다양한 연구가 수행되어왔다. 선행연구 [10]에서는 서비스 지연시간을 최소화하기 위해 사용자와 서비스 노드 간 홉(Hop) 수를 최소화하는 정수 선형 계획법(ILP, Integer Linear Programming) 모델을 정의하고, 이를 해결하기 위한 네트워크 상태 예측 기반 서비스 배치 방법을 제안하였다. 또한, 선행연구 [11]에서는 서비스 배치와 사용자의 서비스 요청이 어떤 엣지에서 처리되어야 하는지 스케줄링하는 문제를 탐욕적 알고리즘 기반으로 해결하였다. 이 외에도 기계학습이나 강화학습 기법을 활용하여 동적인 네트워크 환경에서의 서비스 배치 문제를 최적화하려는 연구가 수행되었다 [12, 13].

한편, 서비스 트래픽 라우팅은 사용자가 특정 서비스를 제공받는 과정에서 발생하는 서비스 트래픽을 네트워크의 어떤 경로를 통해 전달되어야 하는지를 결정하는 최적화 기술이다. 서비스 배치와 유사하게, 트래픽 라우팅에서도 네트워크 운용 비용과 각 서비스별 QoS 보장을 위한 최단 경로 결정이 중요한 조건으로 작용한다.

일반적으로 트래픽 라우팅은 서비스 종류에 따라 트래픽을 최적 경로를 통해 목적지로 전달하는 기술로 인지된다 [14]. 즉, 운용 비용이나 홉 수를 최소화하는 최단 경로를 결정하는 알고리즘들이 주로 연구되었으나, 5G와 6G에서는 위성-지상 통합 네트워크로 범위가 확대되어 동적으로 변화하는 네트워크 토폴로지와 불균형한 트래픽 부하를 처리해야 하는 복잡한 문제로 진화하였다. 예시로, 선행연구 [15]에서는 강화학습을 통해 각 위성 노드를 독립적인 에이전트로 취급하고, 기계학습을 활용하여 트래픽 부하를 예측하는 방안을 제안하였다. 이 외에도 트래픽 라우팅에서 운용 비용과 함께 탄소 배출량을 절감하는 것이 중요시되며 탄소 배출량 절감을 목적으로 하는 라우팅 기법이 제안되었다 [16].

상기 서술한 연구들 외에도, 최근에는 생성형 인공지능 모델을 활용한 네트워크 최적화 연구가 주목을 받고 있다. 특히, LLM은 네트워크 운용 정책을 자연어로 표현된 의도(Intent)로 입력받아 추론 결과를 도출할 수 있는 특성으로 인해 다양한 분야에 적용되고 있다 [17]. 그 예시로, 선행연구 [18]에서는 LLM을 활용하여 VM (Virtual Machine) 생성 및 VNF (Virtual Network Function) 설정을 포함

하는 VNF 배치 자동화 방안을 제안하였고, 선행연구 [19]에서는 사용자 의도를 소프트웨어 정의 네트워크(SDN, Software-Defined Network)의 제어평면(CP, Control Plane)에서 인식할 수 있는 라우팅 결정으로 변환하는 방안을 제시하였다. 이 외에도 LLM의 범용적인 추론 능력을 활용하여, LLM을 최적화 과정의 다양한 단계를 자동화할 수 있는 최적화 도구로써 사용되는 연구가 수행되고 있다 [20, 21, 22].

이처럼 5G와 6G 네트워크에서의 서비스 배치, 트래픽 라우팅과 같은 네트워크 최적화 문제는 엄격한 QoS 조건 달성과 함께 운용 비용, 탄소 배출량 절감 등을 최소화하는 다목적 최적화 문제로 진화하고 있다. 따라서, LLM 적용 등을 통해 동적으로 변화하는 네트워크 환경에서 효과적으로 서비스 배치 및 트래픽 라우팅 결정을 내리는 기술에 대한 수요가 증가하고 있으며, 서비스 배치 및 트래픽 라우팅을 동시에 고려하는 최적 의사 결정 방안이 필요할 것으로 예상된다.

III. 시스템 디자인

3.1. 시스템 모델

네트워크 소프트웨어화(Network softwarization)로 인해 네트워크 운용 및 관리 기능과 서비스를 동적이고 유연하게 배포하는 것이 가능하게 되었다. 특히, SDN은 이를 실현한 대표적인 예로, 중앙 집중화된 제어 평면을 통해 네트워크 상태를 관측하고 서비스 배치 또는 트래픽 라우팅 결정을 내린 후 이를 데이터 평면에 적용하는 구조를 가진다. 5G와 6G에서도 요구사항에 맞는 최적 의사 결정을 내리기 위해서는 해당 구조가 적용될 것으로 예상된다. 그림 1은 제어 평면을 통해 서비스 배치 및 트래픽 라우팅을 수행하는 시스템 모델의 개념을 보인다.

본 논문에서 다루는 서비스 배치 및 트래픽 라우팅 문제는 선행연구 [23]을 기반으로 정의된다. 구체적으로는 서비스 트래픽이 전달되는 네트워크 도메인(Domain)과 서비스가 배치되어 트래픽을 처리하는 연산 도메인이 포함된다. 따라서, 네트워크를 구성하는 노드(Node)들은 역할에 따라서 네트워크 노드 N^d 와 연산 노드 N^c 로 구분할 수 있다. 각 노드는 서로 다른 노드와 링크(Link)로 연결될 수 있다. 즉, 서비스 배치와 트래픽 라우팅 환경인 네

트위크는 노드들의 집합 N 과 링크 집합 L 로 표현되는 그래프 $G = (N, L)$ 로 정의된다.

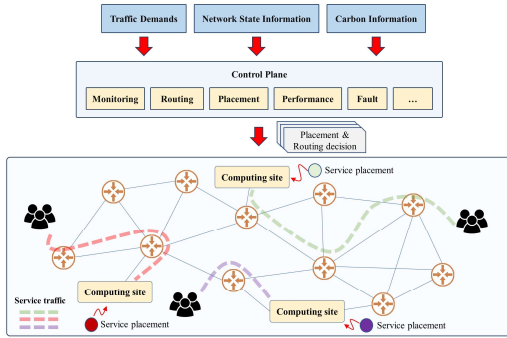


그림 1. 서비스 배치 및 트래픽 라우팅 개념

Fig. 1. Concept of service placement and traffic routing

주어진 네트워크에서 서비스 배치와 트래픽 라우팅은 각 서비스 트래픽의 요구사항 $r \in R$ 이 존재할 때, 사용자로부터 요청받은 서비스가 배치되어야 하는 연산 노드를 N^C 중에서 선택한 후 해당 연산 노드까지의 경로를 구성하는 네트워크 노드들을 N^d 에서 선택하는 것이다. 최종적으로는 선택된 네트워크 노드들을 연결하는 링크들이 경로를 구성하는 요소들이 된다. 즉, 서비스 트래픽 요구에 명시되는 목적지는 고정된 단일 목적지 형태로 표현되는 것이 아니라, 해당 서비스가 배치될 수 있는 위치까지 고려한 여러 선택지가 존재한다. 따라서, 서비스 배치 및 트래픽 라우팅 문제는 서비스 배치 위치와 최적 경로를 결정하는 다목적 최적화 문제이다.

3.2 서비스 배치 및 트래픽 라우팅 모델

서비스 배치 및 트래픽 라우팅 문제는 서비스에 대한 트래픽 요구사항 $r \in R$ 이 생성되었을 때, 사용자에게 제공되어야 하는 서비스가 배치된 연산 노드와 해당 노드로 트래픽이 전달되는 경로에 포함되는 네트워크 노드들에서 발생하는 운용 비용을 최소화하는 문제로 정의할 수 있다. 본 논문에서는 서비스 배치 및 트래픽 라우팅 문제를 다루기 위해 선행 연구 [23]에서 정의된 수식 (1)의 최적화 문제를 활용한다. 해당 수식에서는 트래픽 요구사항 $r \in R$ 을 충족하는 서비스 배치 및 트래픽 라우팅에서 발생하는 운용 비용을 최소화하는 목적을 가진다. 이 때, 운용 비용은 서비스 배치 결정으로 인해 선택된 연산 노드 $n \in N^C$ 에서 발생하는 운용 비용

m_n 과 해당 연산 노드까지 경로에 포함되는 링크들 $l \in L$ 에서의 운용 비용 e_l 을 합한 값으로 표현된다.

$$\min \sum_{r \in R} \left(\sum_{l \in L} x_{rl} e_l + \sum_{n \in N^C} y_{rn} m_n \right) \quad (1)$$

$$\text{s.t.}, \sum_{r \in R} x_{rl} t_r \leq b_l, \forall l \in L \quad (2)$$

$$\sum_{l \in L} x_{rl} d_l \leq g_r, \forall r \in R \quad (3)$$

$$\sum_{l \in L} x_{rl} \log a_l \geq \log k_r, \forall r \in R \quad (4)$$

$$\sum_{l \in L} x_{rl} \leq h_r, \forall r \in R \quad (5)$$

즉, 네트워크 내에서 해당 트래픽 요구사항을 처리하기 위해 선택된 연산 노드와 링크들을 식별할 방법이 필요하며, 이를 위해 수식 (6)과 같이 각각 이진 변수 x_{rl} 와 y_{rn} 을 사용한다.

$$x_{rl}, y_{rn} \in \{0, 1\}, \forall r \in R, l \in L, n \in N^C \quad (6)$$

해당 수식에서 x_{rl} 은 트래픽 요구사항 r 을 목적지로 전달할 때 링크 l 이 경로에 포함될 경우 1의 값을 가지며, y_{rn} 은 연산 노드 n 이 트래픽 요구사항 r 을 처리할 목적지로 선택되는 경우 1의 값을 갖는다. 한편, 해당 트래픽 요구사항에 관련된 트래픽 경로에 포함되지 않은 링크이거나, 서비스 배치에 선택된 연산 노드가 아닌 경우는 0의 값을 갖는다.

그 외 수식 (2)~(5)는 목적 함수인 수식 (1)과 관련된 제약 조건을 나타낸 것이다. 수식 (2)는 네트워크 전송 용량의 한도를 나타내는 제약 사항으로서, 각 트래픽 요구사항 r 이 필요로 하는 전송 용량 t_r 의 합이 링크 l 에서 보장하는 최대 전송 용량인 b_l 이내여야 하는 것을 명시한다. 또한, 수식 (3)은 각 r 이 허용하는 최대 지연 시간인 g_r 가 r 을 위해 결정된 라우팅 경로에 포함되는 각 링크 l 의 지연 시간 d_l 의 합보다 같거나 작아야 하는 조건을 의미한다. 다음으로 수식 (4)와 (5)는 각각 경로에 포함된 링크 l 의 가용성(Availability) a_l 이 트래픽이 요구하는 최소 가용성 k_r 이상이어야 하고, 각 경로는 최대 흡수 h_r 를 넘을 수 없는 것을 제약하는 조건들이다.

한편, 각 트래픽 요구사항 r 을 처리하기 위한 서비스가 배치되는 연산 노드 n 과 관련된 수식 (7)~(9)와 같은 제약 조건들도 존재한다. 수식 (7)은 특정 트래픽 요구사항 r 은 단일 연산 노드에서 처리되어야 한다는 것을 나타내는 조건이다. 다음으로

수식 (8)은 두 종류의 노드 중, 네트워크 노드에서는 트래픽이 처리될 수 없음을 나타내며, 수식 (9)는 선택된 연산 노드 n 에서 트래픽을 처리하는 과정에서 요구되는 자원 $j \in J$ 에 대한 수요량인 f_{rj} 보다 해당 노드의 가용 자원량 c_{nj} 가 크다는 조건을 명시한다.

$$\sum_{n \in N^C} y_{rn} = 1, \forall r \in R \quad (7)$$

$$\sum_{n \in (N - N^C)} y_{rn} = 0, \forall r \in R \quad (8)$$

$$\sum_{r \in R} y_{rn} f_{rj} \leq c_{nj}, \forall n \in N^C, j \in J \quad (9)$$

각 연산 노드에서 트래픽 처리에 사용할 수 있는 자원의 종류로는 CPU, GPU, 메모리가 존재하며 $J = \{\text{CPU, GPU, RAM}\}$ 으로 정의할 수 있다.

IV. 제안 방법

일반적으로 동적 네트워크 환경에서의 서비스 배치는 NP-Hard 문제이며, 트래픽 라우팅 또한 제약 조건이 존재하는 최단 경로 문제(CSP, Constrained Shortest Path)인 경우 NP-Hard로 간주된다 [24]. 따라서, 본 논문의 서비스 배치 및 트래픽 라우팅 문제는 NP-Hard이기 때문에 다항 시간 내 최적해 도출을 보장하기 어렵다. 본 논문에서는 이러한 문제를 해결하기 위해 메타휴리스틱 기법 중 유전 알고리즘(GA, Genetic Algorithm)을 활용한다. 그림 2는 본 논문에서 제안하는 유전 알고리즘 기반 최적화 방식의 절차를 보인다.

4.1 GA 기반 서비스 배치 및 라우팅 방법

유전 알고리즘을 활용한 서비스 배치 및 트래픽 라우팅에서 각 염색체(Chromosome)는 해(Solution)를 표현한다. 여기서 단일 염색체는 목적지가 되는 연산 노드를 포함하는 트래픽 경로를 나타낸다. 즉, 염색체를 구성하는 유전자(Gene)를 살펴보면, 첫 번째 유전자는 항상 트래픽의 출발 지점이 되는 네트워크 노드이고, 서비스 배치로 인해 결정되는 연산 노드는 트래픽의 목적 지점으로서 마지막 유전자에 위치한다. 한편, 제안하는 방법에서는 수식 (5)와 같이 각 트래픽 경로가 허용하는 최대 홉 수가 제한되어 있다. 따라서, 트래픽 경로를 표현하는 염색체의 길이 또한 한계가 존재한다. 이는 곧 유전 알고리즘에서 초기 해를 생성할 때, 출발 노드로부터 목

적지가 될 수 있는 연산 노드를 식별하는 조건이 된다.

제안하는 유전 알고리즘 기반 서비스 배치 및 트래픽 라우팅 방식에서는 트래픽 요구사항이 있을 때, 기본적인 초기 해집단을 생성하기 위해 상기 언급된 제약 조건을 활용하여 너비 우선 탐색(BFS, Breadth-First Search)을 수행한다. 이를 통해 한도 홉 수 내에 도달할 수 있는 연산 노드를 목적으로 고려할 수 있다. 이후, 초기 해집단은 너비 우선 탐색을 통해 식별된 경로들을 초기 해집단 크기에 맞는 개수를 임의로 선택하여 생성한다.

이 외에도 제안하는 방법에서는 LLM을 활용하여 초기 해집단을 생성하는 기능을 제공한다. 초기 해집단을 생성하는 단계에서 LLM을 사용하는 경우, LLM은 네트워크 토폴로지 정보, 트래픽 요구사항, 링크 정보, 운용 비용 등이 포함된 입력값을 프롬프트(Prompt) 형태로 받는다. 이에 대한 결과로 LLM은 초기 해집단에 포함되어야 하는 초기 해들을 생성하며, 이들 중 초기 해집단 크기에 맞는 개수 또는 초기 해집단의 일정 비율을 만족하는 범위에서 선택되어 초기 해집단에 포함된다. LLM을 활용하는 과정에서 불완전한 추론 결과가 도출되는 것을 방지하기 위해 본 논문에서 제안하는 방법에서는 맥락 내 학습 방식을 활용한다. 즉, 입력으로 받는 프롬프트에서는 특정 네트워크 상황에서 생성되었던 초기 해집단을 나타내는 예시가 포함된다.

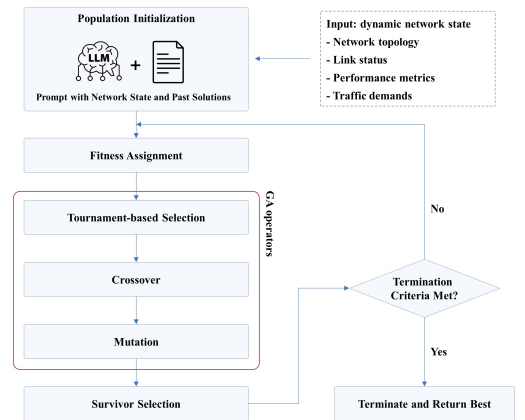


그림 2. 유전 알고리즘 기반 서비스 배치 및 라우팅 순서도
Fig. 2. Procedure of GA-based Service placement & routing

초기 해집단을 생성한 후에는 토너먼트 선택 방식(Tournament-based selection)을 활용하여 해집단 내

에서 부모 염색체 쌍을 선정한 후, 교차(Crossover)와 돌연변이(Mutation)와 같은 유전 연산을 확률적으로 발생시킨다. 이 과정에서 유전 연산으로 인해 단절된 경로나 순환 경로 등 불완전한 경로가 새로운 해로 포함되는 것을 피하기 위해 교차 연산에서는 부모 염색체 간 동일한 유전자, 즉, 두 경로 상 공통으로 존재하는 네트워크 노드 중 하나를 임의로 선택한 후 교차가 발생하는 지점으로 활용한다. 마찬가지로, 돌연변이가 연산에서는 돌연변이가 발생하는 지점의 노드는 해당 노드의 전후에 위치한 노드의 공통 이웃 노드들 중 하나로 변경된다.

상기 과정은 유전 알고리즘이 종결 조건을 달성할 때까지 반복 수행함과 동시에 수식 (10)의 적합도 함수(Fitness function)를 통해 해의 최적화 정도를 측정하여 최적해를 산출한다. 제안하는 방법에서는 일정 세대 수가 반복되는 동안 최적해가 변경되지 않거나 미리 정의된 종결 세대수에 도달하면 알고리즘을 종료하는 것으로 종결 조건을 설정한다.

$$F = C_{total} + P_{total} \quad (10)$$

$$C_{total} = C_{links} + C_{nodes} \quad (11)$$

$$P_{total} = \sum_{z \in Z} P_z^{nw} + \sum_{j \in J} P_j^{cp} \quad (12)$$

$$P_j = \max(0, f_{rj} - c_{nj}) \times w_j \quad (13)$$

앞서 언급된 수식 (10)의 적합도 함수는 각 해의 최적화 정도를 나타낸 값으로서, 트래픽 요구사항을 충족하는 것과 동시에 최소 운용 비용을 가질 수 있도록 설계되었다. 예시로, 해당 적합도 함수를 통해 작은 값이 도출될수록 그 해는 적합한 서비스 배치 및 트래픽 라우팅 결과라는 것을 나타낸다.

구체적으로, 각 해의 적합도는 서비스 배치가 되는 연산 노드에서 트래픽 처리 과정 중 발생하는 운용 비용인 C_{nodes} 와 트래픽 라우팅 경로를 구성하는 각 링크에서의 운용 비용 C_{links} 을 합한 값인 C_{total} 와 함께 트래픽 요구사항의 충족 여부에 따라 부여되는 벌점 P_{total} 을 기반으로 계산된다. 해당 벌점은 수식 (12)의 정의에 따라 네트워크 도메인과 연산 도메인에서의 벌점을 합한 값으로 산출된다. 이 때, 네트워크의 벌점은 성능 지표 $z \in Z$ 에 관련된 충족 여부에 따라 부여되는 P_z^{nw} 로 표현되며, 연산 노드의 벌점은 자원 종류 $j \in J$ 에 해당하는 가용 자원량이 트래픽 처리에 요구되는 정도보다 작을 경우 부여되는 벌점 P_j^{cp} 로 표현된다. 또한, 각 벌점은 요구사항과 대비하여 위반된 수준을 나타내기 위해 가중치 w 를 곱하여 산출한다. 수식 (13)은 자원 종류 j 에

대한 벌점을 계산하는 예시이다. 각 자원 종류별 벌점에 활용되는 가중치는 CPU, GPU, 메모리 순으로 1500, 2000, 1000을 할당하여 활용한다.

4.2 LLM 기반 초기 해집단 생성

LLM 기반 초기 해집단 생성 방법에서는 네트워크 토폴로지, 노드 및 링크 정보, 서비스 요구사항 등이 주어졌을 때, 유전 알고리즘 기반 서비스 배치 및 트래픽 라우팅 수행에 필요한 초기 해집단을 생성하여 제공한다. 일반적인 유전 알고리즘에서 임의로 해집단을 생성하는 것과는 달리, LLM을 통해 다양성을 지닌 높은 품질의 초기 해집단 생성 역할을 수행하기 위해서는 미세조정을 통한 모델의 갱신이 요구된다. 하지만, 미세조정은 큰 연산 비용과 방대한 학습 데이터를 요구할 수 있다는 단점이 존재한다.

본 논문에서 제안하는 방법은 상기 문제에 대응하기 위하여 LLM의 미세조정 대신 프롬프트 내부에 몇 가지 예시를 포함하는 맥락 내 학습 방식을 적용한다. 이러한 방법에서는 프롬프트 내부에 포함되어야 하는 예시를 미리 준비해야 하는 과정이 요구되기 때문에 제안하는 방법에서는 선행연구 [25]의 그래프 신경망(GNN, Graph Neural Network) 기반 해 품질 예측 방법을 통한 초기 해 구성을 활용하였다. 구체적으로는 사전에 특정 네트워크 환경에서 GNN 기반 방식을 통해 고품질 해들을 산출한 후, 각 네트워크 환경과 이에 맞춰 산출된 고품질 해들을 연관지어 저장하여 이들 중 일부를 맥락 내 학습의 예시로 활용하는 것이며, GNN 모델 자체를 초기 해 생성 과정의 구성 요소로 포함하지 않는다. 따라서, 제안하는 방식은 초기 해 생성 단계에서 GNN 모델의 학습 또는 추론은 수행되지 않으며, GNN을 통해 학습 기반으로 수행하던 초기 해 생성을 LLM의 맥락 내 학습 기반 추론으로 대체하는 것을 목적으로 한다.

이와 같이 준비되는 예시는 주어진 네트워크 환경에서 특정 서비스 요구사항을 만족하는 트래픽 라우팅 경로를 의미하는 초기 해집단들 중, GNN을 통해 높은 품질로 판별되는 해집단이다. 즉, 예시는 네트워크 환경, 서비스 요구사항, 그에 대한 고품질 초기 해집단으로 구성되며, 이를 통해 LLM이 임의의 생성 방법 대비 고품질의 초기 해를 구성할 수 있도록 유도한다.

맥락 내 학습 방식을 이용한 LLM 기반 서비스

```
# Task
Generate an initial population for a genetic algorithm, optimizing jointly a service placement and traffic routing.
Each example contains adjacency, computing node resources, edge performance metrics, demand, and population.

# Examples
Adjacency: {"1": [2, 3], "2": [1, 4], "3": [1, 4, 5], "4": [2, 3, 6], "5": [3, 6], "6": [4, 5]}
ComputingNodes: { "2": { "mem": 1000000000, "cpu": 100000000, "gpu": 1000000000, "rmem": 1000000000, "rcpu": 100000000, "rgpu": 1000000000, "cost": 12.34}, ...}
Edges: [{"s": "1", "t": "2", "cap": 100, "avail": 1.0, "delay": 3.49, "rcap": 100, "cost": 5.11}, ...]
Demand: [{"source": "1", "destination": "6", "transmission_capacity": 8.39, "maximum_delay": 61.46, "minimum_availability": 0.91, "maximum_hops": 8, "memory": 14, "cpu": 6, "gpu": 181}
Population: [[1, 2, 4, 6], [1, 3, 5, 6]]

# Output Format
Return only JSON:
{ "population": [[1, "...", 6]] }

# New Task
Source: 1
Destination: N
Adjacency: {"1": [2, 5, 8, ...], "2": [1, 3, ...], "N": [7, 10, ...], ...}
ComputingNodes: {"5": { "mem": 1000000000, "cpu": 100000000, "gpu": 1000000000, "cost": 12.34}, "8": { "mem": 100000, "cpu": 100000, "gpu": 1000000, "cost": 5.64}, ...}
Edges: [{"s": "1", "t": "2", "cap": 200, "avail": 1.0, "delay": 0.24, "rcap": 200, "cost": 8.93}, {"s": "7", "t": "3", "cap": 50, "avail": 1.0, "delay": 2.0, "rcap": 50, "cost": 5.33}, ...]
Demand: {"source": "1", "destination": "N", "transmission_capacity": 4.52, "maximum_delay": 45.3, "minimum_availability": 0.95, "maximum_hops": 8, "memory": 42, "cpu": 5, "gpu": 512}
```

그림 3. 유전 알고리즘 기반 서비스 배치 및 트래픽 라우팅을 위한 LLM 프롬프트 예시

Fig. 3. Example of LLM prompt for GA-based service placement & traffic routing

배치 및 트래픽 라우팅을 위한 초기 해집단 생성에서는 LLM에게 이를 요청하는 프롬프트의 설계가 필수적이다. 제안하는 방법에서 사용하는 프롬프트의 구조는 LLM이 수행해야 하는 서비스 배치 및 트래픽을 라우팅을 위한 초기 해집단을 정의하는 역할의 작업(Task), 맥락 내 학습 방식을 위한 예시(Examples), 결과로 도출되어야 하는 초기 해집단 출력 형식(Output Format), 그리고 초기 해집단이 생성되어야 하는 네트워크 정보와 서비스 요구사항을 명시한 새로운 작업(New Task)으로 구성된다. 그림 3은 일부 내용은 축약된 형태로 표현된 프롬프트의 예시를 보인다.

한편, 제안하는 방법에서는 추론 비용 절감과 초기 다양성 확보를 위해 한 번의 LLM 호출 당 미리 정의된 개수의 초기 해를 생성하는 배치(Batch) 방식을 채택하였다. 즉, 각 LLM 호출이 초기 해집단의 크기에 해당하는 개수의 해를 단일 호출로 생성하지 않고, 복수의 호출을 통해 생성한다. 예를 들어, 10개의 해가 초기 해집단을 구성할 경우, 제안하는 배치 방식에서는 5번의 LLM 호출을 통해 각각 2개의 해를 생성하여 초기 해집단에 포함시킬 수 있다. 이러한 배치 방식은 프롬프트와 추론 결과의 길이 제한으로 인해 정상적으로 해가 반환되지 않는 경우를 방지하기 위해 적용되었다.

마지막으로, LLM을 통해 생성된 초기 해는 해집단에 포함하기 전에 유효성 검사를 수행한다. 정상적인 출력 형식을 만족하지 않거나 네트워크 토폴로지상 유효하지 않은 경로가 생성된 경우, 해당 초기 해는 별도의 수정 없이 폐기(Discard)한다. 이로 인해 유효한 초기 해의 수가 설정된 초기 해집단의 크기에 미달하는 경우에는 기존 GA에서 사용하는

임의의 초기화 방식을 통해 부족한 수의 초기 해를 추가로 생성한다. 이를 통해 LLM의 출력 오류가 발생하는 경우에도 사전에 설정된 초기 해집단의 크기를 유지한 상태에서 이후의 유전 알고리즘을 수행할 수 있도록 구성하였다.

V. 구현 및 성능 평가

5.1. 실험 환경 구성 및 성능 평가 시나리오

LLM을 통해 초기 해집단을 구성하는 유전 알고리즘 기반 서비스 배치 및 트래픽 라우팅 기능은 파이프라인으로 구현되었다.

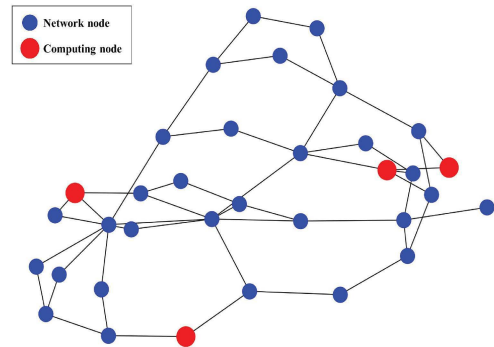


그림 4. Xspedius 네트워크 토폴로지 예시

Fig. 4. Example of Network topology for Xspedius

해당 기능 내 활용된 LLM은 Ollama [26]에서 제공되는 모델 중, 8B 매개변수를 가진 gemma4:e4b를 활용하였으며, 배치 방식의 LLM 호출에서 각

호출 당 반환되는 해의 개수는 2개로 설정하였다. 또한, LLM의 프롬프트와 응답 메시지의 한도를 고려하여 맥락 내 학습으로 인해 포함되는 예시는 1개로 제한하였다.

서비스 배치 및 트래픽 라우팅이 수행되어야 하는 네트워크는 Internet Topology Zoo [27]에서 제공되는 통신 네트워크 토폴로지 중 Xspedius를 사용하였다. 해당 네트워크 토폴로지는 총 34개의 노드가 98개의 유방향 링크를 통해 연결된 형태를 가진다. 이 때, 제안하는 방법의 성능 평가를 위하여 전체 노드 개수의 1/10보다 작은 임의의 개수의 노드를 선택하여 연산 노드로 분류하며, 나머지는 네트워크 노드의 역할을 수행한다. 그림 4는 성능 평가에 사용된 Xspedius 토폴로지의 예시를 보인다.

표 1. 시뮬레이션 및 환경 설정
Table 1. Simulation and environment configuration

Category	Type	Value
Environment	Link capacity (Gbps)	{50, 100, 200}
	Link availability	[0.9999, 0.999999]
	Link operating cost	[1, 10]
	Computing operating cost	[1, 20]
Service req.	Transmission capacity (Gbps)	[0.1, 10]
	Maximum delay (ms)	[10, 100]
	Minimum availability	[0.9, 0.999]
	Required resource - CPU	[1, 10]
	Required resource - RAM	[1, 100]
	Required resource - GPU	[100, 1000]
GA	Population size	10
	Generations	100
	Mutation rate	0.1
	Crossover rate	0.8
LLM	Model	Gemma4:e4b
	Parameters, Quantization	8B, Q4_K_M
	Temperature	1
	Top_k, Top_p	64, 0.95

주어진 네트워크에서 서비스 배치 및 트래픽 라우팅은 필요 연산 자원(CPU, GPU, 메모리), 최소 가용성, 허용 가능한 최대 지연 시간, 최소 전송 용량 등을 포함하는 서비스 요청(Service request)이 발생할 때 수행된다. 이 때, 서비스 요청은 임의의 개수가 생성되며, 각 요청은 성능 요구사항 항목별로 임의의 범위 내 값이 할당된다. 또한, 제안하는 방법은 유전 알고리즘을 기반으로 해당 최적화를 수행하기 때문에 세대 수(Generation), 해집단의 크기(Population size), 유전 연산 확률(Mutation and

crossover rate) 등이 매개변수로 설정되어야 한다. 특히, 이들 중 해집단의 크기는 초기해의 다양성을 위해 세밀한 설정이 필요한 매개변수이며, 네트워크 토폴로지에서 구성될 수 있는 라우팅 경로의 개수를 고려하여 10으로 설정되었다. 이와 관련된 상세 실험 환경 구성 정보는 표 1에 정리되어 있다.

5.2. 성능 분석

성능 평가는 서비스 배치와 트래픽 라우팅이 요구되는 서비스 요청을 25, 50, 75, 100개 생성한 후, 이를 대상으로 유전 알고리즘 기반 서비스 배치 및 트래픽 라우팅을 수행하여 산출한 평균값을 비교 분석하였다. 이 때, 평균값은 각 서비스 요청 개수의 경우마다 20회를 반복 수행하여 계산하였다. 대조군으로는 유전 알고리즘 기반 최적화 방법들을 구현하여 활용하였으며, 구체적으로는 임의로 해를 생성하는 기존 유전 알고리즘 방식(즉, BFS 경로 중 임의의 선택)과 선행연구 [25]에서 초기 해집단의 품질을 GNN을 통해 예측한 후 고품질의 초기 해집단을 선택하여 활용하는 유전 알고리즘 방식을 사용하였다. GNN 기반 GA 방식에서는 Xspedius 토폴로지에서 10,000개의 데이터셋을 통해 학습을 수행한 GNN을 활용하였다. 모델 학습 시에는 은닉층의 개수와 미니배치(Mini-batch) 크기는 각각 64와 32로 설정하였으며, 학습률 0.001을 적용하여 총 100 에포크(Epoch) 동안 학습을 수행하였다. 모델의 파라미터 최적화에는 Adam 최적화를 사용하였다. 각 대조군들은 GA (Base)와 GA (GNN-based)로 표기되며, 제안하는 방식은 GA (LLM-based)로 명시된다.

첫 번째로, 그림 5는 서비스 요청 수의 증가에 따른 총 운용 비용(Total operational cost)을 비교한 결과이다. 전체적으로 서비스 요청 수가 증가할수록 모든 유전 알고리즘 기반 최적화 방법에서 총 운용 비용이 증가하는 것으로 관측되었다. 이는 서비스 요청 개수가 증가할수록 서비스 배치와 트래픽 라우팅에 더 많은 네트워크 자원이 할당되고 링크가 활성화되기 때문인 것으로 해석할 수 있다. 한편, GNN 기반 방식과 LLM 기반 방식에서는 서비스 요청 개수가 100개 미만인 경우 유사한 총 운용 비용이 발생하였지만 100개의 서비스 요청에서는 LLM 기반 방식이 가장 낮은 운용 비용을 달성하였다. 이는 LLM 기반 방법이 대규모 서비스 환경에서 다른 방식 대비 효율적인 초기 해집단을 생성할 수 있다는 가능성을 보인다.

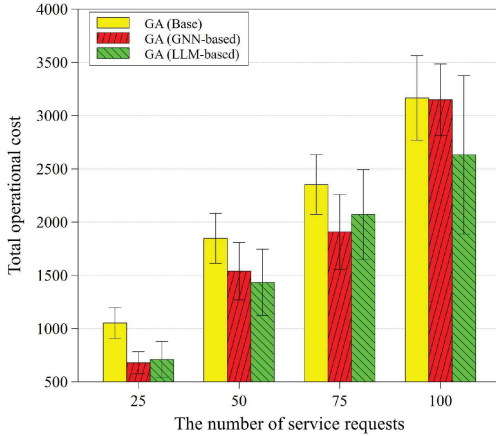


그림 5. 유전 알고리즘 기반 최적화의 평균 총 운용 비용
Fig. 5. Avg. total operational cost of GA-based optimization

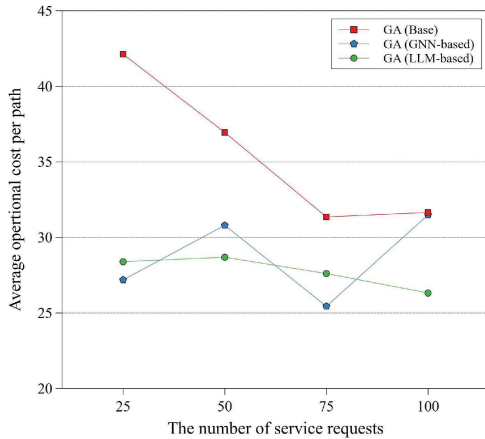


그림 6. 유전 알고리즘 기반 최적화의 평균 운용 비용
Fig. 6. Avg. operational cost of GA-based optimization

다음으로, 그림 6은 각 서비스 요청에 의해 생성되는 경로 당 평균 운용 비용(Average operational cost per path)을 비교한 결과를 나타낸다. 기존 유전 알고리즘 방식에서는 모든 서비스 요청 수에서 가장 높은 평균 운용 비용을 보였으며, 특히 서비스 요청 수가 적은 경우에는 다른 방식과의 차이가 크게 나타났다. 반면, GNN 기반과 LLM 기반 방식에서는 기존 방식보다 낮은 평균 비용을 달성하였다. 그 중에서도 제안하는 방식은 서비스 요청 수가 증가하여도 평균 비용의 변동이 비교적 작게 유지되었으며, 전체적으로 가장 안정적인 비용을 산출하는 것으로 확인되었다. 이는 LLM 기반 초기 해 생성

이 특정 네트워크 환경에 과도하게 의존하지 않고 다양한 초기 해를 생성함으로써 보다 안정적인 탐색을 수행할 수 있기 때문에 분석된다.

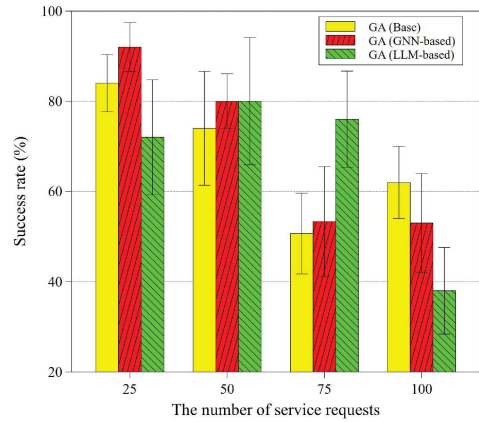


그림 7. 유전 알고리즘 기반 최적화의 서비스 성공률
Fig. 7. Service success rate of GA-based optimization

이 외에도, 그림 7은 서비스 요청 수에 따른 서비스 성공률(Success rate)을 비교한 결과를 나타낸다. 성능 평가 결과, 서비스 요청 수가 25개와 50개인 경우에는 GNN 기반 방식이 가장 높은 서비스 성공률을 보였다. 한편, 제안하는 LLM 기반 방식은 75개 이하의 서비스 요청에서 약 72~80%의 비교적 안정적인 성공률을 보였으며, 특히 75개 요청에서는 약 76%로 다른 방식보다 높은 성능을 나타냈다. 그러나 100개 요청에서는 성공률이 약 38%로 감소하여 75개 요청 대비 약 38% 감소하였다. 이는 서비스 요청 증가에 따른 자원 경합의 심화로 인해 LLM 기반 초기 해의 이점이 제한된 것으로 해석할 수 있다. 따라서 LLM 기반 초기화는 중간 수준의 네트워크 부하에서는 서비스 성공률 향상에 효과적이지만, 높은 부하 환경에서는 일부 한계가 존재함을 보인다.

마지막으로, 그림 8은 서비스 요청 수에 따른 평균 전송 용량(Average transmission capacity)과 평균 지연 시간(Average latency)을 비교한 결과를 나타낸다. 구체적으로는 각 서비스 요청에 따라 생성되는 트래픽 경로별 평균 전송 용량과 지연 시간을 의미한다. 성능 평가 결과, 평균 전송 용량 측면에서는 서비스 성공률 사례와 유사하게 LLM 기반 방식이 중간 수준의 네트워크 부하에서 가장 높은 전

송 용량을 보장하는 경로를 선정하였다. 또한, 평균 지연 시간의 경우, LLM 기반 방식이 변동폭이 가장 안정적인 지연 시간을 달성하였다. 반면, 기존 방식과 GNN 기반 방식에서는 서비스 요청 수에 따라 지연 시간의 변동폭이 크게 관측되었다. 이는 LLM 기반 초기화가 지연 시간과 네트워크 자원 활용을 동시에 고려하는 초기 해를 생성함으로써 최적화 과정에서 보다 균형 잡힌 경로를 선택하도록 유도한 것으로 분석된다. 특히, 서비스 요청 수가 증가하여도 지연 시간 측면에서는 크게 증가하지 않았다는 점은 제안하는 방법이 동적인 네트워크 환경에서도 안정적인 QoS를 제공할 수 있다는 가능성을 보여준다.

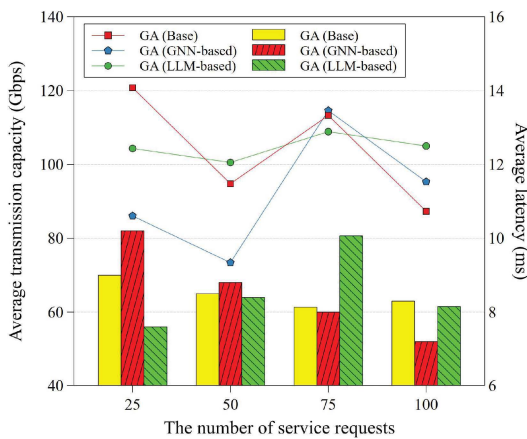


그림 8. 평균 전송량 및 지연 시간

Fig. 8. Avg. transmission capacity and latency

5.3. 논의 및 한계점

종합적인 성능 평가 결과, 제안하는 LLM 기반 초기 해 생성 방식은 모든 성능 지표에서 항상 최적의 결과를 보이지는 않았지만, 평균 운용 비용과 평균 지연 시간 측면에서 비교적 안정적인 성능을 나타냈다. 또한, 서비스 요청이 증가하는 환경에서도 운용 비용 증가를 효과적으로 억제하여, LLM을 활용한 초기 해집단 구성이 유전 알고리즘의 탐색 성능 향상에 기여할 수 있음을 확인하였다.

본 연구에서 LLM은 유전 알고리즘의 반복적인 탐색 과정에 직접 관여하지 않으며, 유전 알고리즘 수행 이전의 초기 해집단 생성 단계에서만 활용된다. 따라서, LLM 기반 초기화는 유전 알고리즘의 탐색

시작점에 영향을 미치는 방식으로 동작하며, 이후의 선택, 교차 및 돌연변이 과정은 기존 유전 알고리즘과 동일하게 수행된다. 이러한 결과는 LLM 기반 초기 해 생성 방식이 유전 알고리즘의 초기 탐색을 지원하기 위한 효과적인 방법으로 활용될 가능성을 보여준다.

다만, 초기 해집단의 품질이 유전 알고리즘의 수렴 과정과 최종 해의 품질에 미치는 영향을 보다 명확하게 규명하기 위해서는 세대별 최적 적합도(Best fitness) 변화와 수렴 세대에 대한 추가적인 분석이 필요하다. 또한, LLM 기반 초기 해 생성에는 별도의 추론 시간이 요구되므로, 실제 적용 가능성을 평가하기 위해서는 초기 해 생성에 소요되는 시간과 이후 유전 알고리즘의 최적화 수행시간을 구분하여 분석할 필요가 있다. 따라서 향후 연구에서는 다양한 네트워크 및 실험 환경을 대상으로 세대별 수렴 특성을 분석하고, LLM 추론 시간을 포함한 전체 수행시간을 정량적으로 평가함으로써 제안 방식의 일반화 가능성과 실제 적용 가능성을 보다 구체적으로 검증할 필요가 있을 것으로 사료된다.

이외에도, LLM을 통해 생성된 유효하지 않은 초기 해는 폐기하고 기존 임의 초기화 방식을 통해 부족한 해를 보완하지만, 이러한 출력 오류가 초기 해집단 구성에 미치는 영향에 대해서는 추가적인 분석이 요구된다. 향후 연구에서는 오류 유형별 발생 빈도를 정량화하고, 다양한 네트워크 및 추론 조건에 따른 LLM 출력의 신뢰성과 초기 해 생성의 안정성에 대한 분석이 필요하다.

VI. 결 론

5G와 다가오는 6G 네트워크에서는 동적 네트워크 환경에서 엄격한 성능 요구사항을 갖는 네트워크 서비스 제공이 주된 과제가 될 것으로 예상된다. 특히, 운용 비용을 최소화하기 위한 네트워크 서비스 배치 및 트래픽 라우팅 기법이 필수적이다. 이를 위해, 본 논문에서는 유전 알고리즘 기반의 서비스 배치 및 트래픽 라우팅 문제 해결 방안을 제안하였다.

제안하는 방법에서는 유전 알고리즘이 초기 해 구성에 따라 성능이 가변적이라는 한계를 극복하기 위해 대형 언어 모델과 맥락 내 학습을 활용하였다. 성능 평가 결과, 본 논문에서 제안하는 방법은 동적으로 변화하는 네트워크 환경에서 임의의 초기 해를 생성하는 기존 유전 알고리즘 방식 대비 운용 비용 측면에서 개선된 결과를 도출하는 것으로 검

증되었다. 향후 연구로는 유전 알고리즘 외 메타휴리스틱 기법에 기반한 성능 평가를 통한 범용성 검증과 메타휴리스틱 알고리즘의 최적 초매개변수 제어를 통한 성능 개선을 수행할 예정이다. 또한, 6G 시나리오를 기반으로 한 최적화 모델을 설계한 후, 제안하는 방법의 효용성을 다양한 네트워크 토폴로지에서 추가적으로 검증할 것이며, 논의 및 한계점에서 다룬 항목들에 대한 개선 및 성능 분석을 수행할 계획이다.

References

- [1] Wang, Cheng-Xiang, et al. "On the road to 6G: Visions, requirements, key technologies, and testbeds." *IEEE Communications Surveys & Tutorials* 25.2 (2023): 905-974.
- [2] Li, Chen, et al. "AI-native user-centric network for 6G." 2022 IEEE/CIC International Conference on Communications in China (ICCC Workshops). IEEE, 2022.
- [3] Jeong-hwan Kim, et al. (2024). Dynamic Offloading and CPU Clock Scaling for Edge Computing in LEO Satellite Networks. *The Journal of Korean Institute of Communications and Information Sciences*, 49(4), 547-555.
- [4] Sanghoon Lee, et al. (2025). QoS-aware Routing Algorithm in Satellite and Ground Integrated Networks. *The Journal of Korean Institute of Communications and Information Sciences*, 50(5), 781-789.
- [5] Yang, Chien-Sheng, et al. "Toward 6G sustainable mobile communications." *IEEE Wireless Communications* 32.1 (2025): 44-50.
- [6] Karunaharan, Vithusan, et al. "A Comprehensive Survey on 6G and Beyond for Climate Action: Toward Sustainable Connectivity." *IEEE Access* (2026).
- [7] Benzaid, Chafika, and Tarik Taleb. "AI-driven zero touch network and service management in 5G and beyond: Challenges and research directions." *Ieee Network* 34.2 (2020): 186-194.
- [8] Yao, Xin. "Finding approximate solutions to NP-hard problems by neural networks is hard." *Information Processing Letters* 41.2 (1992): 93-98.
- [9] Peres, Fernando, and Mauro Castelli. "Combinatorial optimization problems and metaheuristics: Review, challenges, design, and development." *Applied sciences* 11.14 (2021): 6449.
- [10] Velasquez, Karima, et al. "Service placement for latency reduction in the internet of things." *Annals of Telecommunications* 72.1 (2017): 105-115.
- [11] Farhadi, Vajiheh, et al. "Service placement and request scheduling for data-intensive applications in edge clouds." *IEEE/ACM Transactions on Networking* 29.2 (2021): 779-792.
- [12] Rawat, Pradeep Singh, et al. "Efficient virtual machine placement in cloud computing environment using BSO-ANN based hybrid technique." *Alexandria Engineering Journal* 110 (2025): 145-152.
- [13] Liu, Tong, et al. "Deep reinforcement learning based approach for online service placement and computation resource allocation in edge computing." *IEEE Transactions on Mobile Computing* 22.7 (2022): 3870-3881.
- [14] Singh, Sandeep Kumar, Tamal Das, and Admela Jukan. "A survey on internet multipath routing and provisioning." *IEEE Communications Surveys & Tutorials* 17.4 (2015): 2157-2175.
- [15] Ju, Ying, et al. "Dynamic load-balancing routing strategy for leo satellite networks based on spatio-temporal traffic prediction." *IEEE Transactions on Aerospace and Electronic Systems* (2025).
- [16] Tabaeiaghdaei, Seyedali, et al. "Carbon-aware global routing in path-aware networks." *Proceedings of the 14th ACM International Conference on Future Energy Systems*. 2023.
- [17] Seok Bin Son, et al. (2025). Trends in Network Optimization Using Large Language Models. *The Journal of Korean Institute of Communications and Information Sciences*, 50(7), 1073-1084.
- [18] Nam, Sukhyun, Nguyen Van Tu, and James Won-Ki Hong. "LLM-Based AI Agent for VNF Deployment in OpenStack Environment." *NOMS 2025-2025 IEEE Network Operations and*

Management Symposium. IEEE, 2025.

- [19] Xia, Handa, et al. "An LLM-Based Intent-Driven Framework for Automated Routing Configuration." 2025 IEEE 25th International Conference on Communication Technology (ICCT). IEEE, 2025.
- [20] Lange, R.; Tian, Y.; Tang, Y. Large language models as evolution strategies. In Proceedings of the Proceedings of the Genetic and Evolutionary Computation Conference Companion, 2024, pp. 579–582.
- [21] Liu, S.; Chen, C.; Qu, X.; Tang, K.; Ong, Y.S. Large language models as evolutionary optimizers. In Proceedings of the 2024 IEEE Congress on Evolutionary Computation (CEC). IEEE, 2024, pp. 1–8.
- [22] Yang, C.; Wang, X.; Lu, Y.; Liu, H.; Le, Q.V.; Zhou, D.; Chen, X. Large language models as optimizers. In Proceedings of the International Conference on Learning Representations, 2024, Vol. 2024, pp. 12028–12068.
- [23] Ding, Hongqing, et al. "Optimal routing and heterogeneous resource allocation for computing-aware networks." ICT Express 10.3 (2024): 614-619.
- [24] Garey, Michael R., and David S. Johnson. Computers and intractability. Vol. 29. New York: wh freeman, 2002
- [25] Lee, Doyoung, and Taeyeon Kim. "Embedding-Assisted Genetic Algorithm for Routing Optimization in 6G Networks." Mathematics 13.21 (2025): 3536.
- [26] Marcondes, Francisco S., et al. "Using ollama." Natural Language Analytics with Generative Large-Language Models: A Practical Approach with Ollama and Open-Source LLMs. Cham: Springer Nature Switzerland, 2025. 23-35.
- [27] Knight, Simon, et al. "The internet topology zoo." IEEE Journal on Selected Areas in Communications 29.9 (2011): 1765-1775.

이 도 영 (Do-Young Lee)



2015 건국대학교 컴퓨터공학과
학사

2021 POSTECH 컴퓨터공학과
박사

2021~2023 삼성전자 네트워크
사업부 Staff Engineer

2023~2026 한국전자통신연구원
선임연구원

2026~ 현재 국립공주대학교 정보통신공학과 조교수
<관심분야> 6G, 네트워크 최적화, 지능형 네트워크