

# 동적 환경에서의 엣지 컴퓨팅을 위한 효율적 태스크 오프로딩 기법

정 소 향\*, 장 민 희\*, 김 우 영\*\*, 이 재 호\*\*, 이 재 민\*, 김 동 성<sup>o</sup>

## Efficient Hybrid Task Offloading Scheme for Edge Computing in Dynamic Environments

So-hyang Jung\*, Min-hui Jang\*, Woo-young Kim\*\*, Jae-ho Lee\*\*,  
Jae-min Lee\*, Dong-seong Kim<sup>o</sup>

### 요 약

본 논문은 두 가지 유형의 오프로딩 대상을 이용한 하이브리드 태스크 오프로딩 기법을 제안한다. 기지국 기반 네트워크 구조에 UAV 기반 엣지 서버를 추가로 도입하여 데이터를 병렬처리 하여 태스크 오프로딩 성능을 향상하고자 한다. 이는 모바일 디바이스의 데드라인을 만족하기 위해 태스크를 가중치에 따라 분할하여 로컬, UAV 서버, 기지국 서버에서 처리하는 구조를 가진다. 본 논문에서는 공중/지상 경로에 따라 발생하는 총 실행 시간을 계산하고 이를 문제 수식으로 정리하였다. 또한 문제 수식을 최소화하기 위해 디스커버리 프로세스와 가중치를 적용한 Greedy 알고리즘을 제안한다. 시뮬레이션 결과 제안한 기법을 사용했을 때 총 실행 시간이 29% 정도 향상된 것을 확인하였다.

**Key Words** : UAV, Edge Computing, Hybrid Network, Task-Offloading, Greedy Algorithm

### ABSTRACT

This paper proposes a hybrid task offloading scheme using two types of offloading path. This intended to improve task offloading performance by parallel processing data by additionally introducing UAV-based edge servers to the base station-based network structure. And this has a structure in which tasks are divided according to weights and processed by local, UAV, and base station servers to satisfy the deadline of the mobile device. In this paper, the total running execute time is calculated by the problem formula according to the air/ground path. This paper also proposed a Greedy algorithm with a discovery process and weights to minimize problem formulas. As a result of the simulation, it was confirmed that the total execution time was improved by about 29% when the proposed scheme was used.

\* 본 연구는 대학중점연구소 지원사업(2018R1A6A1A03024003)과 2022년도 한화시스템(주)의 재원을 지원받아 수행된 연구임.

• First Author : Samsung Electronics, so\_hyang@kumoh.ac.kr, 학생회원

o Corresponding Author : Kumoh National Institute of Technology, Dept. of IT Convergence Eng., dskim@kumoh.ac.kr, 종신회원

\* Kumoh National Institute of Technology, Dept. of IT Convergence Eng., jmhee@kumoh.ac.kr, 학생회원; ljmpaul@kumoh.ac.kr, 종신회원

\*\* Hanwha Systems, wooyoung2.kim@hanwha.com; jh8028@hanwha.com

논문번호 : 202209-203-B-RU, Received September 5, 2022; Revised October 25, 2022; Accepted November 2, 2022

## I. 서 론

최근 사물인터넷(Internet of Things, IoT) 기술이 발달하고 VR, AR, 머신러닝, 인공지능 등 새로운 형태의 서비스를 이용한 어플리케이션이 개발되고 있다<sup>[1,2]</sup>. 이와 같은 서비스는 높은 연산 능력을 요구하며, 모바일 디바이스가 처리해야 할 데이터양을 증가시켰다. 모바일 디바이스는 포터블한 특성으로 인해 저장 공간, 연산 능력 등의 자원이 제한되어 있다<sup>[3]</sup>. 그 중에서 IoT 소형 디바이스와 웨어러블 디바이스의 경우 심각한 자원 제약 문제를 가지고 있다<sup>[4,5]</sup>. 이러한 자원 제약은 모바일 디바이스의 작업 부하를 초래하였고, 이를 해결하기 위해 태스크 오프로딩(Task Offloading) 기술이 등장했다<sup>[6-8]</sup>. 태스크 오프로딩은 연산 성능이 우수한 서버를 도입하여 태스크를 대신 처리해주는 기법을 말하며, 서버와 디바이스 간 물리적인 구조에 따라서 엣지(Edge), 포그(Fog), 클라우드(cloud) 서버로 분류된다<sup>[9]</sup>. 그 중 엣지 서버는 물리적인 거리로 발생하는 네트워크 지연시간을 줄이기 위해 서버를 노드 가까이 배치한 구조를 나타낸다<sup>[10]</sup>.

모바일 엣지 컴퓨팅(Mobile Edge Computing, MEC)은 분산 클라우드 컴퓨팅 기술을 적용하여 모바일 네트워크 에지에서 컴퓨팅 능력 및 서비스 환경을 제공한다<sup>[11]</sup>. 이는 모바일 디바이스의 한정된 컴퓨팅을 엣지 서버를 통해 태스크를 대신 처리해주는 것을 포함한다. 엣지 서버가 기지국과 같이 노드 근처에 배치되어 있어 비교적 네트워크 지연이 작고 데이터를 분산적으로 처리할 수 있다. 또한 안정적인 전력 공급이 가능하여 연산 능력이 우수한 서버를 배치할 수 있고 대규모 연산을 처리할 수 있다<sup>[12]</sup>. 그러나 서버의 위치가 고정되어 있어 모바일 디바이스가 기지국의 통신 범위를 벗어날 경우 서비스를 지원받을 수 없다. 특히 기지국 인프라의 성능이 부족한 곳이나 재난 상황으로 인해 네트워크 인프라가 손상된 지역의 경우 태스크 오프로딩을 진행할 수 없다는 문제점이 있다<sup>[13,14]</sup>.

최근 모바일 디바이스의 이동성과 제한적인 네트워크 인프라 구조를 고려하기 위해 UAV(Unmanned Aerial Vehicle)에 엣지 서버를 탑재한 연구가 진행되고 있다<sup>[15,16]</sup>. 최근 안정적인 운용이 가능하고 효율적인 비용을 가지고 있는 UAV를 이용한 무선 네트워크 시스템에 대한 연구가 활발히 진행되고 있다<sup>[17]</sup>. 특히 기지국 인프라가 손상되거나 신속하게 추가 인프라 구축이 필요한 지역에 투입되어 통신 중계기의 역할을 담당한다<sup>[18]</sup>. UAV기반의 엣지 서버를 이용한 태스

크 오프로딩은 기지국 기반의 네트워크 송수신이 원활하지 모바일 디바이스를 목표로 한다. 그러나 UAV 또한 배터리를 통해 전력을 공급받기 때문에 자원 제약을 가지고 있다. UAV기반 태스크 오프로딩에 대한 기존 연구는 주로 UAV의 에너지 소모량을 최소화하는 데에 초점이 맞추어져 있다<sup>[19-21]</sup>. UAV로만 이루어진 태스크 오프로딩 기법은 장시간 운용되기 어렵다는 문제점을 가진다.

따라서 본 논문에서는 단일 유형의 엣지 서버만 고려한 기존 연구와 달리 서로 다른 특성을 가진 기지국/UAV 기반의 엣지 서버를 모두 고려한 하이브리드 태스크 오프로딩 기법을 제안한다. 기지국 기반의 엣지 서버는 안정적으로 전력을 공급받지만, 높은 설치 비용을 요구하며 서비스 가능 범위가 고정적이라는 단점을 가지고 있다. UAV 기반의 엣지 서버는 이동성으로 인해 서비스 가능 범위가 가변적이지만, 에너지 제약으로 인해 장시간 서비스를 제공하기가 어렵다는 단점을 가지고 있다. 본 논문에서는 두 가지 유형의 엣지 서버를 도입함으로써 각 엣지 서버의 단점을 보완하고 장점을 활용하여 전반적인 태스크 오프로딩 성능 향상을 목표로 한다. 본 논문에서는 각 과정에서 요구되는 총 실행 시간을 계산하고, 이를 최소화 하기 위해 경로에 가중치를 배정하고 탐욕(Greedy) 알고리즘을 적용한다. 그리고 MATLAB을 이용하여 엣지 서버가 운용되는 환경을 구성하고 제안하는 기법의 성능 평가를 진행한다.

## II. 관련 연구 및 기존 연구 문제점 분석

모바일 엣지 컴퓨팅은 IoT의 급속한 발전에 따라 요구되는 높은 대역폭과 낮은 대기 시간을 만족하는 환경을 위해 등장하였다. 이러한 환경은 엣지 디바이스의 컴퓨팅 자원과 계산 속도 그리고 하드웨어 기능의 한계를 극복하기 위해 작업의 전체 혹은 일부를 오프로드 하는 방안을 요구한다. [22]에서는 IoT 환경에서 클라우드 컴퓨팅과 MEC의 협력을 위한 오프로딩 방안을 제안하였다. 기존 오프로딩 방안은 단일 사용자 환경에서 작업을 할당하는 방안에 중점을 둔다. 그러나 IoT 환경은 다수의 노드 및 사용자로 구성되어 있다. 따라서 다중 사용자 상황을 전체로 작업 자원을 고려해야 한다. 또한 해당 기법에 사용되는 최적 오프로드 결정 방안식은 계산의 복잡성으로 인해 평균 실행 지연시간이 발생할 수 있다. 따라서 [22]에서는 MEC 서버 대비 모바일 디바이스 수 및 사용자 수에 따라 시나리오를 분배하였다. 이는 풍부한 컴퓨팅 리

소스를 가진 클라우드와 낮은 전송 지연을 가진 엣지의 특성을 종합적으로 고려하여 좀 더 나은 서비스를 제공할 수 있는 환경에서 작업을 처리하여 클라우드와 엣지 간의 협력 환경과 네트워크 자원을 효율적으로 활용할 수 있는 환경을 제공함을 뜻한다. 그러나 이러한 방안은 모바일 엣지 디바이스가 이동 중 기지국의 범위를 벗어나 네트워크 끊어짐으로 인한 작업 처리가 불가능한 상황의 대처방안이 부족하다. 기지국을 적절하게 배치하여 모든 디바이스가 안정적으로 서비스를 제공 받으면 문제 상황을 해결할 수 있지만, 이는 높은 비용을 요구하게 된다. 그러나 이러한 구조의 MEC의 경우 지형적 특성으로 인해 네트워크 인프라가 부실하거나 재난 상황으로 손상되는 경우 태스크 오프로딩 성능이 현저하게 저하된다<sup>[16-18]</sup>.

UAV 기반의 엣지 서버를 이용한 태스크 오프로딩은 기지국 네트워크 인프라에 영향을 받지 않는다. UAV가 직접 모바일 디바이스로 접근하여 태스크 오프로딩을 진행한다. 그러나 UAV는 안정적으로 전력을 공급받지 않고 배터리에 의존하기 때문에 이로 인해 에너지 제약이 발생한다. 참고 문헌 [19]의 경우 태스크별 QoS를 만족하면서 에너지 소모량 최소화하는 것을 우선적으로 고려하고 있다. 참고 문헌 [20-21]에서는 UAV의 에너지 잔여량을 계산하고 이를 기준으로 UAV 경로를 최적화하고, 태스크량을 할당하였다. 즉 UAV를 이용한 MEC는 에너지 제약 문제를 중요한 이슈로 고려하며 이와 관련된 다양한 논문이 연구되고 있다<sup>[23-24]</sup>. 참고문헌 [25]에서는 다중 UAV를 사용하여 태스크 오프로딩을 진행했으며 UAV의 에너지 절약에 대한 연구를 진행하였다. 그러나 UAV만으로 구성된 태스크 오프로딩 시스템은 안정적으로 장시간 운용되기 어렵다는 문제점을 가진다.

### III. 하이브리드 태스크 오프로딩 설계

#### 3.1 시스템 모델

본 논문에서는 단일 UAV와 다수의 기지국을 이용한 하이브리드 태스크 오프로딩 기법의 시스템 모델과 시나리오를 고려하였다. 그림 1은 제안하는 시스템 모델의 아키텍처를 나타낸다. 모바일 디바이스, 기지국, UAV로 구성되어 있으며, 태스크를 기지국으로 할당하는 Ground Offloading(지상 오프로딩)과 UAV로 할당하는 Air Offloading(공중 오프로딩)을 고려한다. 각 오프로딩 과정에서 요구되는 통신 링크는 LoS(Line of Sight) 채널에 지배된다고 가정하며, 제안하는 모델은 3차원 직교 좌표계 위에 배치된다.

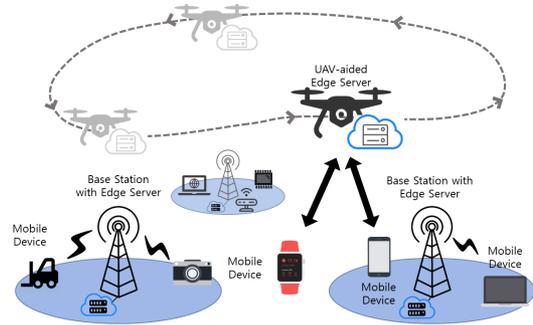


그림 1. 하이브리드 태스크 오프로딩을 위한 아키텍처  
Fig. 1. Architecture for Hybrid task-offloading

모바일 디바이스와 기지국은 지상 위에 존재하며, 고도가 0이라고 가정한다. UAV는 일정한 고도를 가지고 비행하며 공중 오프로딩을 요구하는 모바일 디바이스의 최소 거리에 진입할 수 있도록 순차적으로 비행한다.

본 논문에서는 두 가지 유형의 태스크 오프로딩 방법을 고려한 하이브리드 태스크 오프로딩 기법을 제안한다. 제안하는 하이브리드 태스크 오프로딩의 시나리오는 다음과 같이 진행된다.

#### 3.2 하이브리드 태스크 오프로딩 시나리오

디바이스에서 처리해야 할 태스크가 발생할 경우 자체적으로 데이터를 처리하는 경우와 UAV/기지국 기반의 엣지 서버에서 데이터를 처리하는 경우의 소요 시간을 모두 계산한다. 이때 총 실행 시간이 낮은 경로에 더 높은 가중치를 부여하여 태스크를 분할하고자 하며 가중치는 총 실행 시간의 역수를 퍼센트로 환산한 값이다. UAV/기지국 기반의 태스크 오프로딩 결정 여부는 디스커버리 프로세스를 통해 결정된다. 기지국 범위 밖에 디바이스가 존재할 경우 기지국에 연결된 엣지 서버와 통신할 수 없으므로 지상 오프로딩은 고려하지 않는다. 기지국 범위 내에 디바이스가 존재할 경우 기지국/UAV 기반 엣지 서버를 모두 고려하여 태스크 오프로딩을 진행한다. UAV는 공중 오프로딩을 요구하는 디바이스 근처로 이동하며, 이때 발생하는 이동시간이 일정 수준을 넘을 경우 공중 오프로딩을 고려하지 않는다. UAV는 에너지 제약이 존재하기 때문에 UAV 기반의 엣지 서버는 중량을 우선으로 고려해야 한다. 기지국 기반의 엣지 서버는 컴퓨팅 성능을 우선으로 고려한다. 이로 인해 각 엣지 서버 간의 연산 성능 차이가 존재하며, 이러한 특성을 고려하여 각 유형의 엣지 서버에 가중치를 부여한다.

### 3.3 문제 수식 정의

표 1은 제안하는 기법의 시스템 모델을 정의하기 위해 필요한 파라미터를 정의한 것이며, 본 모델은 다음과 같은 가정이 존재한다.

- 하나의 엣지 서버는 동 시간대에 여러 태스크를 처리할 수 없다.
- 태스크 오프로딩 진행 중에는 모든 모바일 디바이스의 위치가 고정되어 있다.
- 태스크 오프로딩 후 결과값을 전송하는데 소요되는 다운 링크 시간은 무시한다.
- UAV의 속도와 시간당 소모하는 에너지량은 고정되어 있다.
- 태스크 오프로딩 과정에서 발생하는 대기열 및 스케줄링은 고려하지 않는다.

표 1. 파라미터 설정  
Table 1. Parameter setting

$D_k$	입력 데이터 크기	$C_k$	1 bit를 처리하는데 요구되는 CPU 사이클 [CPU cycles/bit]
$f_k^{nd}$	k번째 모바일 디바이스의 컴퓨팅 능력 [CPU cycles/s]	$f^{uc}$	UAV 엣지 서버의 컴퓨팅 능력 [CPU cycles/s]
$f^{bc}$	기지국 엣지 서버의 컴퓨팅 능력 [CPU cycles/s]	$f^{bc}$	기지국 엣지 서버의 컴퓨팅 능력 [CPU cycles/s]
$\sigma^2$	노이즈	$B$	대역폭
$\alpha_0$	채널 이득 기준 값	$g_k^{trans}$	송수신 디바이스 간 채널 이득
$p_k$	k번째 디바이스의 송신전력	$d_{x,y}$	x와 y 사이의 거리
$t_k^{md}$	모바일 디바이스가 태스크를 처리하는 데 소요되는 처리 시간	$t_k^{uc}$	UAV 엣지 서버에서 태스크를 처리하는 데 소요되는 처리 시간
$t_k^{bc}$	기지국 기반 엣지 서버에서 태스크를 처리하는 데 소요되는 처리 시간	$R_k^{uav}$	디바이스와 UAV 간 데이터 전송 속도
$g_k^{uav}$	디바이스와 UAV 간 채널 이득	$v^{uav}$	UAV의 이동 속도
$t_k^{mov}$	UAV가 k번째 모바일 디바이스로 이동하는 데 소요되는 시간	$R_k^{bs}$	디바이스와 기지국 간 데이터 전송 속도
$g_k^{bs}$	디바이스와 기지국 간 채널 이득	$t_k^{uav}$	디바이스와 UAV 간 데이터 전송 시간
$t_k^{bs}$	디바이스와 기지국 간 데이터 전송 시간	$t_k^{tc}$	k번째 모바일 디바이스를 처리하기 위한 대기 시간

$P_j^{bc}$	기지국 기반 엣지 서버에서 기존 데이터를 처리하는데 소요되는 시간	$t_k^{trans}$	태스크를 오프로딩할 때 발생하는 전송 시간
$E^{uav}$	UAV가 가지는 총 에너지	$E_h$	UAV가 초당 소모하는 에너지
$L_k^m$	k번째 모바일 디바이스에서 발생한 태스크의 메모리	$L_k^s$	k번째 모바일 디바이스에서 발생한 태스크의 저장용량
$B_j^m$	j번째 기지국 엣지 서버의 메모리	$U^s$	UAV 기반 엣지 서버의 저장용량
$U^m$	UAV 기반 엣지 서버의 메모리	$B_j^s$	j번째 기지국 엣지 서버의 저장용량
$T_k^{loc}$	로컬에서 태스크를 처리할 때 소요되는 총 실행 시간	$T_k^{pr}$	지상 오프로딩시 소요되는 총 실행 시간
$T_k^{air}$	공중 오프로딩시 소요되는 총 실행 시간		

#### 3.3.1 데이터 처리 시간 모델링

본 절에서는 모바일 디바이스와 각 엣지 서버에서 데이터를 처리할 때 요구되는 데이터 처리 시간을 서술한다. 제안하는 시스템의 모바일 디바이스는 k개로 구성되어 있고,  $k = \{1, 2, \dots, K\}$ 로 정의된다. 태스크를 처리하기 위해 소요되는 처리 시간은 수식 (1)과 같이 표현한다.  $t_k^{proc}$ 는 k번째 모바일 디바이스가 태스크를 처리할 때 소요되는 데이터 처리 시간을 의미한다.  $C_k$ 는 주어진 태스크의 컴퓨팅 강도,  $D_k$ 는 입력된 데이터의 크기,  $f$ 는 태스크를 처리하는 컴퓨터의 연산 성능을 나타낸다.  $f$ 값이 클수록 연산 능력이 우수한 것을 의미하며, 같은 데이터를 더 빨리 처리할 수 있다.

$$t_k^{proc} = \frac{C_k D_k}{f} \quad (1)$$

#### 3.3.2 전송 시간 모델링

본 절에서는 태스크 오프로딩을 진행할 때 데이터를 각 엣지 서버로 전송하는 데 소요되는 시간에 대해 서술한다. 전송 시간은 전송 속도와 전송할 데이터의 크기에 영향을 받는다. 이를  $t_k^{trans}$ 라 표현하고 수식 (2)와 같이 표현한다.

$$t_k^{trans} = \frac{D_k}{R_k^{trans}} \quad (2)$$

$R_k^{trans}$ 는 송수신 디바이스 간의 전송 속도를 나타내며 수식 (3)으로 표현할 수 있다. 수식 (4)는 채널이득을 나타내며 송수신 대상 간의 거리에 영향을 받아 태스크를 오프로딩 진행할 때마다 가변적인 값을 가진다.

$$R_k^{trans} = B \log_2 \left( 1 + \frac{P_k g_k^{trans}}{\sigma^2} \right) \quad (3)$$

$$g_k^{trans} = \alpha_0 d_{x,y}^{-2} \quad (4)$$

### 3.3.3 로컬 프로세싱 및 공중/지상 오프로딩에 요구되는 총 실행 시간

본 절에서는 태스크를 로컬에서 처리할 때 혹은 공중/지상 오프로딩을 통해 처리했을 때 요구되는 총 실행 시간을 계산한다. 그림 2는 로컬 및 공중/지상 오프로딩에서 데이터를 처리할 때 발생하는 총 실행 시간을 나타낸다.

#### 3.3.3.1 디바이스에서 처리

모바일 디바이스에서 자체적으로 데이터를 처리할 때 요구되는 총 실행 시간  $T_k^{loc}$ 을 수식 (5)로 나타낼 수 있다. 모바일 디바이스는 단일 태스크를 가지고 있어 대기 시간, 송신 시간을 고려할 필요가 없다. 모바일 디바이스가 가진 연산 능력을 이용한 데이터 처리 시간  $t_k^{md}$ 만 고려한다.

$$T_k^{loc} = t_k^{md} = \frac{D_k C_{ok}}{f_k^{loc}} \quad (5)$$

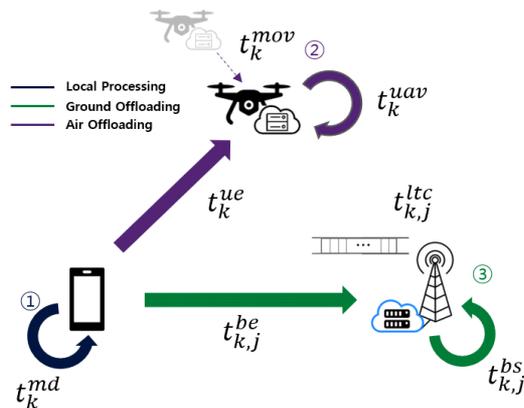


그림 2. 각 과정에서 데이터를 처리할 때 발생하는 총 실행 시간  
Fig. 2. Total run time during each task processing

#### 3.3.3.2 UAV 기반 엣지 서버에서 처리

공중 오프로딩을 진행할 때 요구되는 총 실행 시간  $T_k^{air}$ 를 수식 (6)으로 나타낼 수 있다. UAV 기반의 엣지 서버를 사용하므로 데이터 처리 시간  $t_k^{uc}$ , 데이터 전송 시간  $t_k^{uav}$  그리고 UAV가 디바이스 근처로 이동하는 시간  $t_k^{mov}$ 를 고려한다.  $t_k^{mov}$ 는 UAV의 속도와 UAV의 수평적 위치와 k번째 모바일 디바이스의 수평적 위치의 차이에 의해 결정된다. UAV는 k번째 모바일 디바이스의 태스크를 처리하면서 동시에 k+n번째 모바일 디바이스로 이동하기 때문에 대기 시간이 없다고 가정한다.

$$T_k^{air} = t_k^{uc} + t_k^{uav} + t_k^{mov} = \frac{C_k D_{\beta k}}{f^{uc}} + \frac{D_{\beta k}}{B \log_2 \left( 1 + \frac{P_k g_k^{uav}}{\sigma^2} \right)} + \frac{d_{k,uav}}{v^{uav}} \quad (6)$$

#### 3.3.3.3 기지국 기반 엣지 서버에서 처리

지상 오프로딩을 진행할 때 요구되는 총 실행 시간  $T_{k,j}^{gr}$ 은 수식 (7)로 나타낼 수 있다. k번째 모바일 디바이스가 j번째 기지국 기반의 엣지 서버로 지상 오프로딩을 진행하는 경우 데이터 처리 시간  $t_{k,j}^{bc}$ , 데이터 전송 시간  $t_{k,j}^{bs}$  그리고 j번째 기지국 기반의 엣지 서버가 가진 대기 시간  $t_{k,j}^{ltc}$ 을 고려한다.

$$T_{k,j}^{gr} = t_{k,j}^{bc} + t_{k,j}^{bs} + t_{k,j}^{ltc} = \frac{C_k D_{\gamma k}}{f_j^{bc}} + \frac{D_{\gamma k}}{B \log_2 \left( 1 + \frac{P_k g_{k,j}^{bc}}{\sigma^2} \right)} + P_j^{bc} \quad (7)$$

#### 3.3.3.4 문제 수식 모델링

본 절에서는 하이브리드 태스크 오프로딩 과정에서 발생하는 총 실행 시간을 수식으로 나타낸다. 앞서 설명한 실행 시간  $T_k^{loc}$ ,  $T_k^{air}$ ,  $T_{k,j}^{gr}$ 을 기반으로 수식 (8)과 같이 문제 수식 P1을 모델링 할 수 있다.

$\alpha_k$ ,  $\beta_k$ ,  $\gamma_k$ 는 데이터를 처리하는 경로를 결정하는 변수이다.  $\alpha_k$ ,  $\beta_k$ ,  $\gamma_k$ 의 값이 1일 경우 로컬, 공중/지상 오프로딩이 발생한다. C2는 모바일 디바이스의 로컬 컴퓨팅 성능과 각 엣지 서버가 가진 컴퓨팅 성능 차이를 나타낸다. C3은 지상 태스크 오프로딩 과정에서 j번째 기지국에 연결된 엣지 서버에 오프로딩한 태스크 메모리 합과 저장 용량 합이 해당 엣지 서버의

$$\begin{aligned}
 P1 : & \sum_{k \in K} \alpha_k T_k^{loc} + \beta_k T_k^{air} + \gamma_k T_{k,j}^{gr}, \\
 C1 : & \alpha_k = \{0,1\}, \beta_k = \{0,1\}, \gamma_k = \{0,1\}, \\
 C2 : & f_k^{loc} < f_k^{ue} < f_k^{fe}, \\
 C3 : & \sum_{k \in K} \beta_k L_{k,j}^s < B_j^s, \sum_{k \in K} \beta_k L_{k,j}^m < B_j^m, \\
 C4 : & \sum_{k \in K} \beta_k L_k^s < U^s, \sum_{k \in K} \beta_k L_k^m < U^m, \\
 C5 : & \sum_{k \in K} E_h t_k^{mov} \leq E^{uav},
 \end{aligned} \tag{8}$$

메모리와 저장 용량을 초과할 수 없다는 것을 의미한다. C4는 공중 오프로딩 과정에서 태스크 저장 용량 합과 메모리 합이 UAV에 연결된 엣지 서버의 저장 용량 합과 메모리 합을 초과할 수 없다는 것을 의미한다. 즉 C3와 C4는 각 엣지 서버의 에너지 제약을 나타낸다. 마지막으로 C5는 UAV의 에너지 제약을 나타내며, UAV를 운용할 수 있는 시간  $t_k^{mov}$ 는 UAV의 총 에너지 제약  $E^{uav}$ 를 만족해야 한다.

#### IV. 하이브리드 태스크 오프로딩 알고리즘

##### 4.1 하이브리드 태스크 오프로딩 알고리즘

본 논문에서는 문제 수식 P1을 해결하기 위해 그림 3과 같은 태스크 오프로딩 알고리즘을 제안한다. 주어진 태스크의 데드라인을 만족하고 총 실행 시간을 줄이기 위해 디스커버리 과정, 가중치를 적용한 탐욕 알고리즘 단계를 거친다.

##### 4.1.1 디스커버리 과정

모바일 디바이스가 로컬, 공중/지상 오프로딩을 결정했다고 가정하고 이때 소요되는 총 실행 시간을 모두 계산한다. 공중 오프로딩을 고려하기 위해 UAV와 모바일 디바이스 간의 통신할 수 있는 최소 거리인 수평적 거리를 구한다. UAV가 최소 거리까지 이동하는데 소요 시간을 계산하고 그 값이 로컬에서 처리하는 시간보다 클 경우 공중 오프로딩을 고려하지 않는다. 지상 오프로딩은 모바일 디바이스가 기지국 범위 내에 있을 경우만 고려되기 때문에 모바일 디바이스와 j 번째 기지국 간의 거리  $d_{k,j}$ 가 기지국 범위  $d^{cov}$ 를 만족하는지 조사한다.  $d_{k,j}$ 가  $d^{cov}$ 보다 클 경우 모바일 디바이스가 기지국 범위를 벗어나므로 지상 오프로딩은 고려하지 않는다. 접근 가능한 기지국이 다수일 경우 총 실행 시간이 가장 낮은 기지국을 선택한다. 전체 실행 시간을 최소화하기 위해 가중치를 적용한 탐

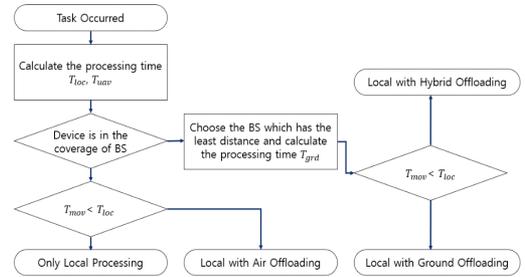


그림 3. 제안하는 태스크 오프로딩 기법의 순서도  
Fig. 3. Flowchart of proposed task offloading techniques

욕 알고리즘을 통해 값을 결정한다.

##### 4.1.2 가중치를 적용한 탐욕 알고리즘

표 2는 총 실행 시간을 줄이기 위해 디스커버리 과정에서 계산한 값을 기준으로 각 경로의 가중치를 설정하는 방법을 보여준다. 각 경로에서 발생하는 총 실행 시간의 역수를 퍼센트로 환산하고 정수로 근사화시킨 값을 가중치로 선정하고, 문제 수식을 최소화하기 위해 탐욕 알고리즘을 기반으로 값을 결정한다.

표 2. 공중/지상 오프로딩 결정을 위한 탐색 과정

Table 2. Exploration process for aerial/ground offloading decisions

Algorithm 1 discovery process for air/grd offloading

---

Input:  $D_k, C_k, j, k, K$  and  $d^{cov}$   
 Output: T  
 Initialize k  
 repeat  
   Initialize j,  $d^{cov}$   
   repeat  
     calculate  $d_{k,j}$   
     if  $d_{k,j} > d^{cov}$  then  
        $\gamma_k = 0$   
     else  
        $\gamma_k = 1$   
       select minimum  $d_{k,j}$   
     end if  
      $j \leftarrow j - 1$   
   until  $j > 0$   
   if  $\gamma_k \neq 0$  then  
     calculate  $P_j^{be}$   
     calculate the  $T_{k,j}^{grd}$   
   end if  
   calculate  $d_{k,uav}$   
   calculate  $T_k^{loc}$   
   if  $d_{k,uav} > T_k^{loc}$  then  
      $\beta_k = 0$   
   else  
      $\beta_k = 1$   
   calculate the  $T_k^{air}$

---

```

end if
calculate the weight
if  $\beta_k = 1$  then

$$T_k^{air} = \frac{C_k D_{\beta k}}{f^{aw}} + \frac{D_{\beta k}}{(1 + \frac{p_k g_k^{uav}}{\sigma^2})} + \frac{d_{k,uav}}{v^{uav}}$$

else

$$T_k^{air} = \frac{C_k D_{\beta k}}{f^{aw}} + \frac{D_{\beta k}}{B \log_2(1 + \frac{p_k g_k^{uav}}{\sigma^2})} + \frac{d_{k,uav}}{v^{uav}}$$

end if
if  $\gamma_k = 1$  then

$$T_{k,j}^{gr} = \frac{C_k D_{\gamma k}}{f_j^{be}} + \frac{D_{\gamma k}}{B \log_2(1 + \frac{p_k g_k^{be}}{\sigma^2})} + P_j^{be}$$

else

$$T_{k,j}^{gr} = 0$$

end if

$$T_k^{loc} = \frac{C_k D_{\alpha k}}{f_k^{loc}}$$

choose minimum T
k ← k + 1
until k < K
    
```

## V. 성능평가 및 분석

### 5.1 시뮬레이션 환경 설정

제안하는 하이브리드 태스크 오프로딩 기법의 성능을 평가하기 위해 MATLAB을 사용하였으며, 시뮬레이션 환경은 다음과 같이 구성되었다. 성능평가를 위해 4개의 기지국과 단일 UAV를 이용하였으며, 태스크를 처리하는 데 요구되는 총 실행 시간을 비교하였다. 모바일 디바이스는  $1,000 \times 1,000 m^2$  필드 위에 임의로 배치되었다. 각 모바일 디바이스가 가진 태스크의 크기와 컴퓨팅 능력은 표 3에 제시된 것과 같이 설정되어 있다. 각각의 태스크는 동시에 발생한다고 가정하였으며 태스크는 첫번째 모바일 디바이스부터 순차적으로 발생한다. UAV는 일정한 고도와 속도를 가지며, 그 값은 20m, 20m/s를 유지한다고 가정한다. UAV의 처음 위치는 유클리드 위치  $(x, y, z) = (500, 500, 20)$ 이다. 기지국은  $(250, 250, 0)$ ,  $(750, 250, 0)$ ,  $(250, 750, 0)$ ,  $(750, 750, 0)$ 에 위치한다. 모바일 디바이스, UAV, 기지국은 각각  $\{1, 6, 10\} GHz$ 의 컴퓨팅 능력을 가진다. 또한 표 3에는 데이터를 송수신할 때 발생하는 파라미터가 정의되어 있으며, 각 모바일 디바이스는 동일한 환경에 놓여있다고 가정한다.

### 5.2 시뮬레이션 결과 및 분석

그림 4는 임의의 태스크를 부여한 모바일 디바이스

표 3. 시뮬레이션 파라미터 설정  
Table 3. Setting Simulation Parameter

파라미터	값
Field Size	$1k m^2$
Speed of UAV	$20 m/s$
Altitude of UAV	$20 m$
Location of mobile devices	Random
Location of base station	BS1 (250,250,0) BS2 (750,250,0) BS3 (250,750,0) BS4 (750,750,0)
Initial location of UAV	(500,500,20)
Coverage of BS	$250 m$
Computing ability of mobile device	$1 GHz$
Computing ability of UAV	$6 GHz$
Computing ability of base station	$10 GHz^2$
Task size	$D_k = \{1 \sim 30 random\} \times 10^6$
Task intensity	$C_k = 1$
Bandwidth, $B$	$20 MHz$
Noise, $\sigma^2$	$-100 dB$
송신전력, $p_k$	$10 dBm$
Reference channel gain, $g_0$	$-50 dB$

를 임의로 배치한 후 소요되는 총 실행 시간의 합산을 비교한 그래프이다. 파란색, 빨간색, 주황색 막대 그래프는 각각 하이브리드 태스크 오프로딩 기법, random partial 태스크 오프로딩 기법, 단일 유형 태스크 오프로딩 기법을 나타낸다. 모두 4개의 기지국과 1개의 UAV로 이루어진 환경에서 시뮬레이션을 진행했으며, 동일한 모바일 디바이스 환경에서 놓여있다. random partial 태스크 오프로딩 기법은 태스크를 분할하여 태스크 오프로딩을 진행하지만 각 경로에 대한 가중치는 설정되어 있지 않다. 단일 유형 태스크 오프로딩 기법은 로컬/공중 오프로딩/지상 오프로딩 중 가장 최적의 한가지 경로를 선택해서 태스크 오프로딩을 진행하는 기법이다. 하이브리드 기법과 random partial 기법을 이용한 그래프에서는 단일 유형의 태스크 오프로딩에 비해 성능이 월등히 향상된 것을 확인할 수 있었다. 또한 태스크를 분할 할 때 각 경로에서의 소요 시간을 고려하여 태스크 분할에 대한 가중치를 적용함으로써 제안하는 하이브리드 태스크 오프로딩 기법에서 성능이 향상된 것을 확인하였다.

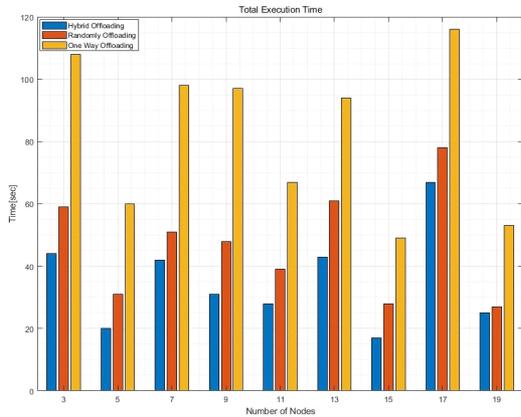


그림 4. 시뮬레이션 결과 발생하는 총 실행 시간 그래프  
Fig. 4. Graph of total run time resulting from the simulation

## VI. 결 론

본 논문에서는 두 가지 유형의 엣지 서버를 이용한 하이브리드 태스크 오프로딩 기법을 제안했다. UAV 기반의 엣지 서버를 추가로 도입하여 기지국 네트워크 인프라가 유실되거나 부족한 환경에서도 태스크 오프로딩을 수행할 수 있다. 오프로딩 과정에서 발생하는 총 실행 시간을 수식으로 표현하고, 이러한 문제 수식을 최소화하기 위해 디스커버리 과정과 가중치를 적용한 탐욕 알고리즘을 제안했다. MATLAB을 이용하여 시뮬레이션을 진행한 결과 제안하는 기법을 사용했을 때 총 실행 시간이 29% 정도 향상된 것을 확인하였다. 향후에는 각 엣지 서버의 대기열 관리 및 스케줄링을 고려하고, 다중 UAV를 도입하는 연구를 진행할 예정이다.

## References

[1] M. A. P. Putra, D.-S. Kim, and J. M. Lee, "Adaptive LRFU replacement policy for named data network in industrial IoT," *ICT Express*, vol. 8, pp. 258-263, Jun. 2022. (<https://doi.org/10.1016/j.icte.2021.10.004>)

[2] Z. Ahmad, L. A. C. Ahakonye, R. Akter, D.-S. Kim, and J.-M. Lee, "An efficient Hybrid-DNN for DDoS detection and classification in software-defined IIoT networks," *IEEE Internet of Things J.*, pp. 1-15, Aug. 2022. (<https://doi.org/10.1109/JIOT.2022.3196942>)

[3] S. H. Im, "Asymmetric multi-core based real-time task scheduling to increase energy conservation in smart mobile devices," *J. Korean Soc. for Digital Contents*, vol. 19, pp. 1219-1224, 2018. (<https://doi.org/10.9728/dcs.2018.19.6.1219>)

[4] D. Oletic and V. Bilas, "Energy-efficient respiratory sounds sensing for personal mobile asthma monitoring," *IEEE Sensors*, vol. 16, no. 23, pp. 8295-8303, Dec. 2016. (<https://doi.org/10.1109/jsen.2016.2585039>)

[5] T. J. Voss, V. Subbian, and F. R. Beyette, "Feasibility of energy harvesting techniques for wearable medical devices," *Annu. Int. Conf. IEEE Eng. in Med. and Biol. Soc.*, pp. 626-629, Chicago, USA, Aug. 2014. (<https://doi.org/10.1109/embc.2014.6943669>)

[6] H. A. Alameddine, S. Sharafeddine, S. Sebbah, S. Ayoubi, and C. Assi, "Dynamic task offloading and scheduling for low-latency IoT services in multi-access," *IEEE J. Sel. Areas in Commun.*, vol. 37, pp. 668-682, 2019. (<https://doi.org/10.1109/jsac.2019.2894306>)

[7] H. Tran-Dang and D.-S. Kim, "FRATO: Fog resource based adaptive task offloading for delay-minimizing IoT service provisioning," *IEEE Trans. Parall. and Distrib. Syst.*, vol. 32, pp. 2491-2508, Oct. 2021. (<https://doi.org/10.1109/TPDS.2021.3067654>)

[8] K. Kumar, J. Liu, Y.-H. Lu, and B. Bhargava, "A survey of computation offloading for mobile systems," *Mobile Netw. and Appl.*, vol. 18, pp. 129-140, Feb. 2013. (<https://doi.org/10.1007/s11036-012-0368-0>)

[9] G. Ortiz, M. Zouai, O. Kazar, A. Garcia-de-Pradoa, and J. Boubeta-Puig, "Atmosphere: Context and situational-aware collaborative IoT architecture for edge-fog-cloud computing," *Comput. Standards & Interfaces*, vol. 79, pp. 1-21, Jan. 2022. (<https://doi.org/10.1016/j.csi.2021.103550>)

[10] X. Sun and N. Ansari, "Latency aware workload offloading in the cloudlet network," *IEEE Commun. Lett.*, vol. 21, pp. 1481-1484,

- Apr. 2017.  
(<https://doi.org/10.1109/lcomm.2017.2690678>)
- [11] H. Guo, J. Liu, J. Zhang, W. Sun, and N. Kato, "Mobile-edge computation offloading for ultradense IoT networks," *IEEE Internet of Things J.*, vol. 5, pp. 4977-4988, May 2018.  
(<https://doi.org/10.1109/jiot.2018.2838584>)
- [12] J. Liu and Q. Zhang, "Offloading schemes in mobile edge computing for ultra-reliable low latency communications," *IEEE Access*, vol. 6, pp. 12825-12837, Feb. 2018.  
(<https://doi.org/10.1109/access.2018.2800032>)
- [13] F. Zhou, Y. Wu, R. Q. Hu, and Y. Qian, "Computation rate maximization in UAV-Enabled wireless-powered mobile-edge computing systems," *IEEE J. Sel. Areas in Commun.*, vol. 36, pp. 1927-1941, Aug. 2018.  
(<https://doi.org/10.1109/jsac.2018.2864426>)
- [14] J. Liu, L. Li, F. Yang, X. Liu, X. Li, X. Tang, and Z. Han, "Minimization of offloading delay for two-tier UAV with mobile edge computing," *IWCMC*, pp. 1534-1538, Tangier, Morocco, Jun. 2019.  
(<https://doi.org/10.1109/iwcmc.2019.8766778>)
- [15] X. Diao, J. Zheng, Y. Cai, Y. Wu, and A. Anpalagan, "Fair data allocation and trajectory optimization for UAV-Assisted mobile edge computing," *IEEE Commun. Lett.*, vol. 23, pp. 2357-2361, Sep. 2019.  
(<https://doi.org/10.1109/lcomm.2019.2943461>)
- [16] Y. Zeng, R. Zhang, and T. J. Lim, "Wireless communications with unmanned aerial vehicles opportunities and challenges," *IEEE Commun. Mag.*, vol. 54, pp. 36-42, May 2016.  
(<https://doi.org/10.1109/mcom.2016.7470933>)
- [17] R. Akter, M. Golam, V.-S. Doan, J.-M. Lee, and D.-S. Kim, "IoMT-Net: Blockchain integrated unauthorized UAV localization using lightweight convolution neural network for internet of military things," *IEEE Internet of Things J.*, pp. 1-19, May 2022.  
(<https://doi.org/10.1109/jiot.2022.3176310>)
- [18] E. A. Tuli, M. Golam, D.-S. Kim, and J. M. Lee, "Performance enhancement of optimized link state routing protocol by parameter configuration for UANET," *Drones* 2022, vol. 6, pp. 1-15 Jan. 2022.  
(<https://doi.org/10.3390/drones6010022>)
- [19] A. A. A. Ateya, A. Muthanna, R. Kirichek, M. Hammoudeh, and A. Koucheryavy, "Energy and latency-aware hybrid offloading algorithm for UAVs," *IEEE Access*, vol. 7, pp. 37587-37600, Mar. 2019.  
(<https://doi.org/10.1109/access.2019.2905249>)
- [20] Y. Qian, F. Wang, J. Li, L. Shi, K. Cai, and F. Shu, "User association and path planning for UAV-Aided mobile edge computing with energy restriction," *IEEE Wireless Commun. Lett.*, vol. 8, pp. 1312-1315, Apr. 2019.  
(<https://doi.org/10.1109/lwc.2019.2913843>)
- [21] L. Fan, W. Yan, X. Chen, Z. Chen, and Q. Shi, "An energy efficient design for UAV communication with mobile edge computing," *China Commun.*, vol. 16, pp. 26-36, Feb. 2019.  
(<https://doi.org/10.12676/j.cc.2019.01.003>)
- [22] Z. Ning, P. Dong, X. Kong, and F. Xia, "A cooperative partial computation offloading scheme for mobile edge computing enabled internet of things," *IEEE Internet of Things J.*, vol. 6, pp. 4804-4814, Jun. 2019.  
(<https://doi.org/10.1109/jiot.2018.2868616>)
- [23] C. Yang, Y. Liu, X. Chen, W. Zhong, and S. Xie, "Efficient mobility-aware task offloading for vehicular edge computing networks," *IEEE Access*, vol. 7, pp. 26652-26664, 2019.  
(<https://doi.org/10.1109/access.2019.2900530>)
- [24] F. Cui, Y. Cai, Z. Qin, M. Zhao, and G. Y. Li, "Multiple access for mobile-UAV enabled networks: Joint trajectory design and resource allocation," *IEEE Trans. Commun.*, vol. 67, pp. 4980-4994, Apr. 2019.  
(<https://doi.org/10.1109/tcomm.2019.2910263>)
- [25] F. Xiaowei and G. Xiaoguang, "Multi-UAVs cooperative control in communication relay," *2016 IEEE ICSPCC*, pp.1-5, Hong Kong, Aug. 2016.  
(<https://doi.org/10.1109/icspcc.2016.7753600>)

**정 소 향 (So-hyang Jung)**



2018년 2월: 금오공과대학교 전자공학부 학사졸업  
2020년 2월: 금오공과대학교 IT 융복합공학과 석사졸업  
2020년~2022년: SK M&S  
2022년~현재: 삼성전자  
<관심분야> UAV 네트워크, 스마트 IoT, 엣지 컴퓨팅

[ORCID:0000-0002-4337-186X]

**장 민 희 (Min-hui Jang)**



2019년 8월: 금오공과대학교 전자공학과 졸업  
2021년 8월: 금오공과대학교 IT 융복합공학과 석사졸업  
2021년 9월~현재: 금오공과대학교 IT 융복합공학과 박사과정  
<관심분야> 산업용 IoT, 엣지 컴퓨팅, 미들웨어, 항공기 네트워크, 함정전투체계

[ORCID:0000-0002-0349-5091]

**김 우 영 (Woo-young Kim)**



2003년 2월: 인제대학교 전자공학 졸업  
2003년~2010년: 해군 OCS  
2010년 8월~현재: 한화시스템 해양연구소 시스템 엔지니어  
<관심분야> 함정전투체계, 아키텍처 설계, UX/UI, 시스템 보안

[ORCID:0000-0001-5755-3480]

**이 재 호 (Jae-ho Lee)**



2007년 2월: 금오공과대학교 전자공학부 졸업  
2009년 2월: 경북대학교 센서 및 디스플레이공학과 석사졸업  
2009년~2011년: LG 디스플레이 OLED 사업부 연구원  
2011년~현재: 한화시스템 해양연구소 시스템 엔지니어

<관심분야> 영상 신호처리, 임베디드 시스템

[ORCID:0000-0003-0861-128X]

**이 재 민 (Jae-min Lee)**



1997년 2월: 경북대학교 전자공학과 학사졸업  
1999년 2월: 경북대학교 전자공학과 석사졸업  
2005년 3월: 서울대학교 전기 및 컴퓨터공학부 박사졸업  
2004년~2016년: 삼성전자 수석연구원

2016년~2017년: 금오공과대학교 ICT융합특성화 연구센터 산학협력중점교수

2017년~현재: 금오공과대학교 전자공학부 부교수  
<관심분야> 산업용 통신망, 네트워크 기반 임베디드 시스템 설계 및 성능분석

[ORCID:0000-0001-6885-5185]

**김 동 성 (Dong-seong Kim)**



1992년 2월: 한양대학교 전자공학과 졸업  
2003년 3월: 서울대학교 전기 및 컴퓨터공학부 박사졸업  
2004년: Cornell 대학교 ECE 박사 후 연구원  
2004년~현재: 금오공대 전자공학부 정교수

2015년~2018년: 금오공과대학교 융합기술원 원장

2014년~현재: ICT 융합특성화연구센터 센터장(과기정통부 ITRC 및 연구재단 중점연구소)

2014년~현재: IEEE/ACM Senior 회원

2016년~현재: 국방부 CIO 자문위원

2019년~2022년: 금오공과대학교 산학협력단장

<관심분야> 실시간 S/W, 실시간 통신망 및 IoT 시스템, 네트워크 기반 분산 제어 시스템

[ORCID:0000-0002-2977-5964]