큐 안정성을 고려한 다중 유저 엣지 컴퓨팅 시스템에서 캐싱과 컴퓨팅 기반 지연 시간과 에너지 동시 최소화에 대한 연구

정구선, 권오승, 이인규
고려대학교 전기전자공학과

rntjs300@korea.ac.kr, worat@korea.ac.kr, inkyu@korea.ac.kr

Joint Latency and Energy Minimization Based on Caching and Computing with Queue Stability in Multi-User Mobile Edge Computing Systems

Guseon Jeong, Ohseung Kwon, and Inkyu Lee
School of Electrical Eng., Korea Univ., Korea

요 약

본 논문은 캐싱과 오프로딩 정책이 주어졌을 때, 여러 작업에 대해 각 유저 단말의 지연 시간과 에너지 소모의 가중치 총합을 최소화하는 해에 대해 근사된 closed-form 표현을 제시한다. 실험 결과를 통해 제안하는 근사된 closed-form 으 표현된 해는 최적해와 성능이 유사함을 증명한다. 또한, 지연 시간과 에너지 소모를 줄이는데 있어 캐싱과 오프로딩의 영향 및 동시에 수행해야 함을 실험 결과를 통해 증명한다.

Ⅰ. 서론

가상 현실(virtual reality; VR)과 증강 현실(augmented reality; AR) 같은 지연시간에 민감한 어플리케이션으로 인해 급증하는 트래픽 수요를 만족하기 위해 클라우드 네트워크에서 백홀 전송 지연을 줄이고 고성능 컴퓨팅 능력을 유저 단말(user equipment; UE)과 근접한 네트워크에서 수행하는 엣지 컴퓨팅(mobile edge computing; MEC)이 등장했다. 앞서 언급한 어플리케이션을 유저 단말에서 실행할 때, 성능과 품질(quality-of-service; QoS)는 유저 단말의 제한된 계산 능력에 크게 영향을 받는다. 또한, 유저 단말에서 강도 높은 계산은 에너지를 많이 소비하여 배터리 수명을 심각하게 단축시킨다. 앞서 언급한 지연 시간과 에너지 소모 문제를 해결하기 위해, 유저 단말은 근접한 엣지 컴퓨팅 서버에 무선으로 작업을 오프로딩하고, 엣지 컴퓨팅 서버에 반복적으로 요청하는 작업들은 엣지 컴퓨팅 서버에서 캐싱하는 기술이 제안되었다. [1],[2]

본 논문에서는 다수의 작업에 대해 유저 단말의 지연 시간과 에너지 소모의 가중치 합을 최소화하도록 하는 문제에 대한 해를 근사된 closed-form 표현을 제시한다. 추가로 다양한 오프로딩 및 캐싱 알고리즘과 결합하여 성능을 확인하고, 최적해와 결합된 성능과 비교한다.

Ⅱ. 본론

본 논문에서는 엣지 컴퓨팅 서버를 탑재한 하나의 안테나를 가진 한 개의 base station(BS)와 하나의 안테나를 가진 N 개의 유저 단말이 균일하게 분포하며, time-division duplexing(TDD) 및 유저 단말 간에 간섭이 없는 네트워크를 가정하고, 각 유저 단말과 BS 간의 채널 정보는 알고 있다고 가정한다. N 개의 유저 단말은 같은 K 개의 작업을 시행해야 하며, 각 유저 단말 n 의 작업 k 을 엣지 컴퓨팅 서버에 요청하는 빈도를 $\pi_{n,k}$ 로 정의하며, Zipf 분포를 적용하여 $\pi_{n,k} = k^{-\gamma}/\sum_{m=1}^{K} m^{-\gamma}$ [3]로 정의할 수 있다. BS 는 요청하는 빈도 $\pi_{n,k}$ 을 알고 있다고 가정한다. 작업 k 는 세 개의 파라미터 $\{I_k, D_k, L_k\}$로 정의된다. I_k 는 계산을 위한 데이터 크기, D_k 는 계산을 끝낸 결과에 대한 데이터 크기, L_k는 실행하기 위한 계산 작업량이다.

오프로딩 정책을 $o_{n,k}$로 표시하며, 엣지 컴퓨팅 서버에 유저 단말 n 의 작업 k 을 오프로딩 했으면 $o_{n,k} = 1$, 그렇지 않다면 $o_{n,k} = 0$ 이다. 마찬가지로 캐싱 정책을 c_k 로 표시하며, 엣지 컴퓨팅 서버에서 작업 k 의 데이터를 캐싱했으면 $c_k = 1$, 그렇지 않다면 $c_k = 0$ 이다. 엣지 컴퓨팅 서버는 $\sum_{k=1}^{K} c_k D_k \leq S$의 캐싱 저장 공간 제약조건을 가진다고 가정한다.

유저 단말 n의 작업 k에 대해 총 지연시간을 다음과 같이 세 개로 구성한다. 1) 유저 단말과 엣지 컴퓨팅 서버에서 큐잉 지연$\left(t_{n,k}^{l_q}, t_{n,k}^{s_q}\right)$, 2) 유저 단말과 엣지 컴퓨팅 서버에서 계산 지연$\left(t_{n,k}^{l}, t_{n,k}^{s}\right)$, 3) 오프로딩과 작업에 대한 결과를 전송하기 위한 무선 전송 지연$\left(t_{n,k}^{u}, t_{n,k}^{d}\right)$이다.

$$t_{n,k}^{l} = L_k/f_n,\ t_{n,k}^{s} = L_k/f_s,\ t_{n,k}^{u} = I_k/R_n^u,\ t_{n,k}^{d} = D_k/R_n^d$$

$$t_{n,k}^{l-q} = \left(t_{n,k}^{l}\right)^2 \Big/ \left(t_{n,k}^{d} - t_{n,k}^{l}\right),\ t_{n,k}^{s-q} = \left(t_{n,k}^{s}\right)^2 \Big/ \left(t_{n,k}^{u} - t_{n,k}^{s}\right).$$

여기서 $R_n^u = B\log_2(1 + p_n^u|h_n|^2/N_0)$ 은 오프로딩 전송률을 나타내며, $R_n^d = B\log_2(1 + p_n^d|h_n|^2/N_0)$ 은 결과 다운로딩 전송률을 나타낸다. f_n 은 유저 단말 n 의 CPU 주파수, f_s 는 엣지 컴퓨팅 서버의 CPU 주파수를 나타낸다.[2] 큐잉 지연 수식은 선입선출(first come first serve; FCFS) 기반의 M/M/1 큐를 적용했다.

이어서, 유저 단말 n 의 작업 k 에 대해 총 에너지 소모를 다음과 같이 두 개로 구성한다. 1) 유저 단말과 엣지 컴퓨팅 서버에서 계산에 대한 에너지 소모$\left(e_{n,k}^{l}, e_{n,k}^{s}\right)$와 2) 오프로딩과 작업에 대한 결과를 전송에 대한 에너지 소모$\left(e_{n,k}^{u}, e_{n,k}^{d}\right)$이다.

$$e_{n,k}^{l} = \kappa_0 L_k f_n^2,\ e_{n,k}^{s} = \kappa_0 L_k f_s^2,\ e_{n,k}^{u} = p_n^u t_{n,k}^{u},\ e_{n,k}^{d} = p_n^d t_{n,k}^{d}.$$

여기서 κ_0은 컴퓨팅 에너지 효율 파라미터다. [2]

최종적으로 오프로딩 및 캐싱 정책과 결합된 총 지연시간 T와 총 에너지 소모 E는 다음과 같이 정의된다. [4]

$$T=\sum_{n=1}^{N}\sum_{k=1}^{K}\pi_{n,k}\Big[c_k t_{n,k}^{d}+(1-c_k)\Big((1-o_{n,k})(t_{n,k}^{l}+t_{n,k}^{l-q}) +o_{n,k}(t_{n,k}^{d}+t_{n,k}^{s}+t_{n,k}^{s-q}+t_{n,k}^{u})\Big)\Big]$$

$$E=\sum_{n=1}^{N}\sum_{k=1}^{K}\pi_{n,k}\Big[c_k t_{n,k}^{d}+(1-c_k)\Big((1-o_{n,k})t_{n,k}^{l} +o_{n,k}(t_{n,k}^{d}+t_{n,k}^{s}+t_{n,k}^{u})\Big)\Big].$$

이를 기반으로 다수의 작업에 대해 유저 단말의 지연 시간과 에너지 소모의 가중치 합을 최소화하도록 하는 문제는 다음과 같이 정의된다.

$$\text{P1:}\ \underset{\{o_{n,k}\}_{n=1k=1}^{N\ K},\{c_k\}_{k=1}^{K}}{\text{minimize}}\ \ \beta T+(1-\beta)E$$

$$\underset{\{p_n^u,p_n^d\}_{n=1}^{N},\{f_n\}_{n=1}^{N}}{}$$

$$\text{subject to}\ \ \text{C1: } 0\le p_n^u\le P_{\max}^u \qquad \text{C5: } t_{n,k}^{s}+t_{n,k}^{s-q}\le T_{\max}^{s}$$

$$\text{C2: } 0\le p_n^d\le P_{\max}^d \qquad \text{C6: } o_{n,k}\in\{0,1\}$$

$$\text{C3: } 0\le f_n\le f_{max} \qquad \text{C7: } c_k\in\{0,1\}$$

$$\text{C4: } t_{n,k}^{l}+t_{n,k}^{l-q}\le T_{\max}^{l} \qquad \text{C8: } \sum_{k=1}^{K}c_k D_k\le S.$$

P1 문제는 다수의 작업에 대해 유저 단말의 지연 시간과 에너지 소모의 가중치 합을 최소화하는 유저 단말의 오프로딩 송신 전력(p_n^u), BS 의 결과 다운로딩 송신 전력(p_n^d), 유저 단말의 CPU 주파수(f_n), 그리고 오프로딩$(o_{n,k})$ 및 캐싱(c_k) 정책을 찾는 문제다. β는 지연시간과 에너지 소모 중 어디에 더 중점을 둘지를 나타내는 가중치 계수다.

P1 문제에서 $\{p_n^u,p_n^d,f_n\}$과 $\{t_{n,k}^u,t_{n,k}^d,t_{n,k}^l\}$는 일대일함수(one-to-one function)이므로 C1~C3 의 최적화 변수 $\{p_n^u,p_n^d,f_n\}$를 $\{t_{n,k}^u,t_{n,k}^d,t_{n,k}^l\}$에 대한 제약조건으로 바꾸고 C4~C5 를 $\{t_{n,k}^u,t_{n,k}^d,t_{n,k}^l\}$에 대하여 정리를 한 P2 문제는 다음과 같이 정의된다.

$$\text{P2:}\ \underset{\{o_{n,k}\}_{n=1k=1}^{N\ K},\{c_k\}_{k=1}^{K}}{\text{minimize}}\ \ \beta T+(1-\beta)E$$

$$\underset{\{t_{n,k}^u,t_{n,k}^d,t_{n,k}^l\}_{n=1k=1}^{N\ K}}{}$$

$$\text{subject to}\ \ \text{C1: } t_{n,k}^{u}\ge\frac{I_k}{B\log_2\left(1+\frac{P_{\max}^u|h_n|^2}{N_0}\right)} \qquad \text{C4: } t_{n,k}^{u}\ge\frac{T_{\max}^{s}t_{n,k}^{s}}{T_{\max}^{s}-t_{n,k}^{s}}$$

$$\text{C2: } t_{n,k}^{d}\ge\frac{D_k}{B\log_2\left(1+\frac{P_{\max}^d|h_n|^2}{N_0}\right)} \qquad \text{C5: } t_{n,k}^{d}\ge\frac{T_{\max}^{l}t_{n,k}^{l}}{T_{\max}^{l}-t_{n,k}^{l}}$$

$$\text{C3: } \frac{L_k}{f_{max}}\le t_{n,k}^{l}\le T_{\max}^{l} \qquad \text{C6~C8 in P1}$$

P2 문제의 목적함수와 제약함수는 모두 convex 임을 증명을 통해 확인하였다. 따라서 P2 문제에서 $o_{n,k}$와 c_k가 주어졌을 때, 최적화 변수 $\{t_{n,k}^u,t_{n,k}^d,t_{n,k}^l\}$의 해를 Karush-Kuhn-Tucker(KKT) 조건을 통해 근사된 closed-form 표현으로 정리할 수 있으며, 본 논문에서 해당 수식의 정리는 생략한다.

Ⅲ. 모의실험 및 결론

본 논문에서는 캐싱 저장 공간 S에 따라 유저 단말의 지연 시간과 에너지 소모의 가중치 합의 성능을 모의실험 기반으로 분석한다. 유저 단말 수 N는 3, 작업 수 K는 4 로 설정했다. 작업에 대한 $\{I_k,D_k,L_k\}$는 각각 [5,8] Mb, [10,14] Mb, 그리고 $30I_k$ cycle 로 설정했다. 통신대역폭 B는 10 MHz, P_{max}^u는 1 watt, P_{max}^d는 0.1 watt, 그리고 잡음전력 N_0는 10^{-10} watt 로 설정했다. κ_0는 10^{-26}, f_s는 10 GHz, 그리고 f_{max}는 1 GHz 로 설정했다. 가중치 β는 0.5 로 설정했다. 마지막으로 채널이득 $|h_n|^2=0.2A_d((3\cdot10^8)/4\pi f_c d_n)^{d_e}$ [2]으로 설정했으며, $A_d=4.11$는 안테나 이득, $f_c=10^9$는 중심 주파수, $d_e=2.6$는 경로손실지수, 그리고 $d_n=[50,65,80]m$는 BS 와 유저 단말 간의 거리다.

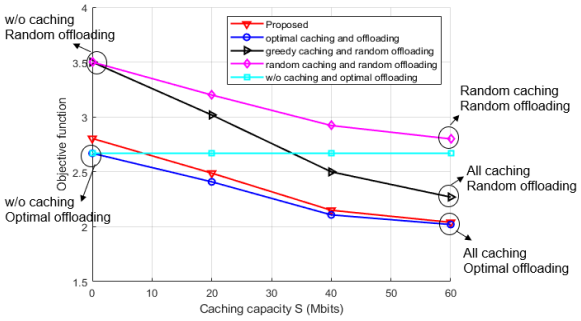

Fig 1. 캐싱 저장 공간에 따른 지연시간과 에너지 소모의 가중치 합의 성능

그림 1 은 P2 문제의 지연시간 해를 여러 오프로딩 및 캐싱 정책과 결합했을 때 지연 시간과 에너지 소모의 가중치 합에 대한 실험 결과이다. 최적의 오프로딩 및 캐싱 정책은 exhaustive search 기법을 적용한 것을 나타낸다. 제안하는 "Proposed" 기법은 최적의 오프로딩 및 캐싱 정책을 적용하고 P2 문제의 지연시간의 해에 대해 근사된 closed-form 을 적용한 기법이다. "Proposed" 기법을 제외한 나머지 기법들은 P2 문제의 지연시간의 해에 대해 CVX 라는 최적화 툴을 통해 도출된 최적해를 적용한 기법이다. 따라서 "optimal caching and offloading" 기법은 성능의 lower bound 이다. 그림 1 에서 greedy caching 은 제한된 캐싱 저장 공간 S 내에서 탐욕적으로 작업을 캐싱하는 기법, random caching 은 무작위로 캐싱하는 기법, 그리고 w/o caching 은 캐싱을 하지 않는 기법이다.

제안하는 기법이 lower bound 와 유사한 성능을 가지는 것을 확인했으며, S 가 작아짐에 따라 성능 열화가 발생하는 이유는 closed-form 을 구하는 과정에서 근사화 과정의 문제로 보이며, 추후 연구에서 분석이 필요하다. 또한, 지연시간과 에너지 소모를 줄이는데 있어서 캐싱과 오프로딩을 동시에 수행해야 하는 이유에 대해 결과를 통해 확인하였다.

ACKNOWLEDGMENT

본 연구는 한국연구재단의 지원을 받아 수행되었음. (No. 2022R1A5A1027646.)

참 고 문 헌

[1] C. F. Liu, M. Bennis, M. Debbah, and H. V. Poor, "Dynamic task offloading and resource allocation for ultra-reliable low-latency edge computing," *IEEE Trans. Commun.*, vol. 67, no. 6, Jun. 2019.

[2] S. Bi, L. Huang, and Y. J. A. Zhang, "Joint optimization of service caching placement and computation offloading in mobile edge computing systems," *IEEE Trans. Wirel. Commun.*, vol.19, no. 7, Jul. 2020.

[3] M. C. Lee and A. F. Molisch, "Optimal delay-outage analysis for noise-limited wireless networks with caching, computing, and communications," *IEEE Trans. Wirel. Commun.*, vol. 22, no. 2, Feb. 2023.

[4] M. Chen, Y. Hao, L. Hu, M. S. Hossain, and A. Ghoneim, "Edge-CoCaCo: toward joint optimization of computation, caching, and communication on edge cloud," *IEEE Wirel. Commun.*, vol. 25, no. 3, Jun. 2018.