

<http://dx.doi.org/10.7236/IIBC.2015.15.3.137>

IIBC 2015-3-20

모바일 클라우드 응용에서 센싱 데이터 동기화를 고려한 응답 시간 분석

Response Time Analysis Considering Sensing Data Synchronization in Mobile Cloud Applications

민 흥*, 허준영**

Hong Min*, Junyoung Heo**

요약 모바일 클라우드 컴퓨팅은 모바일 기기의 자원 제약적인 문제를 해결하기 위해 클라우드 서비스를 활용한다. 이렇게 모바일 기기에서 수행해야 할 작업을 클라우드로 위임하는 것을 오프로딩이라고 하고 에너지 소모의 관점에서 많은 연구들이 진행되고 있다. 본 논문에서는 응답 시간의 관점에서 오프로딩 기법 적용의 효율성을 측정하기 위해 센싱 데이터 동기화를 고려한 응답 시간 모델을 설계하였다. 제안 모델에서는 클라우드에서 모바일 기기가 요청한 작업을 처리할 때 필요한 센싱 데이터에 대한 동기화 작업을 고려하여 정교한 응답 시간 예측을 가능하도록 했으며 모의실험을 통해 새로운 센서 데이터의 발생 비율과 동기화 주기가 응답 시간에 어떤 영향을 주는지를 확인하였다.

Abstract Mobile cloud computing uses cloud service to solve the resource constraint problem of mobile devices. Offloading means that a task executed on the mobile device commits to cloud and many studies related to the energy consumption have been researched. In this paper, we designed a response time model considering sensing data synchronization to estimate the efficiency of the offloading scheme in terms of the response time. The proposed model considers synchronization of required sensing data to improve the accuracy of response time estimation when cloud processes the task requested from a mobile device. We found that the response time is effected by new sensing data generation rate and synchronization period through simulation results.

Key Words : Response time model, sensing data synchronization, mobile cloud computing, offloading

1. 서론

최근 모바일 클라우드 컴퓨팅은 모바일 기기의 자원 제약적인 문제를 해결하기 위해서 클라우드 컴퓨팅의 연산 파워, 메모리, 저장 장치 등의 컴퓨팅 자원을 활용하는 패러다임으로 다양한 분야에서 연구 발전되고 있다^[1,2]. 모바일 기기는 배터리로 동작하고 에너지 효율적인 프로

세서를 탑재하고 있기 때문에 복잡한 연산을 수행하는 것이 어렵다. 또한 무선 통신의 경우 현재 연결된 통신 모듈의 특성에 따라 대역폭이 제한되는 문제점이 발생한다^[3]. 따라서 상당한 컴퓨팅 파워와 데이터 통신이 필요한 이미지 프로세싱과 같은 응용에서는 모바일 기기 내에서 작업을 수행하는 것 보다는 클라우드에 작업을 위임하고 결과만 전송받는 오프로딩 기법을 활용하는 것이

*정회원, 호서대학교 컴퓨터정보공학부(교신저자)

**정회원, 한성대학교 컴퓨터공학과

접수일자 2015년 4월 13일, 수정완료 2015년 5월 13일

계재확정일자 2015년 6월 12일

Received: 13 April, 2015 / Revised: 13 May, 2015 /

Accepted: 12 June, 2015

*Corresponding Author: hmin@hoseo.edu

School of Computer and Information Engineering, Hoseo University, Korea

효율적이다^[4].

오프로딩 기법을 활용하면 모바일 기기 내에서 작업을 수행했을 때 보다 에너지 소모를 줄이거나 응답 시간을 향상 시킬 수 있다. 또한 모바일 클라우드 응용의 주요 해결 과제중의 하나가 기존에 수집된 데이터를 바탕으로 분석 결과를 도출하고 최근에 수집된 센싱 데이터를 통해 사용자의 상태를 파악하여 사용자가 원하는 서비스를 제공해주는 것이다^[5]. 따라서 모바일 클라우드 응용의 응답 시간은 사용자의 만족도를 평가하는 중요한 요소이다^[6]. 그러나 모든 작업을 클라우드에서 처리할 경우 필요한 센싱 데이터를 동기화하는 작업으로 인해 오히려 응답시간이 떨어지는 문제가 발생할 수 있다. 본 논문에서는 오프로딩 시 센싱 데이터를 동기화 주기에 따라 응답 시간에 어떤 영향을 주는지에 대한 모델을 설계하고 모바일 기기 자체에서 처리할 때의 응답시간과 비교하여 사용자에게 최적의 응답시간을 제공하는 기법을 제안한다.

본 논문의 구성은 다음과 같다. 2장에서는 오프로딩 기법의 개념과 센싱 데이터 동기화를 고려하지 않은 기존의 응답 모델을 소개하고, 3장에서는 제안한 센싱 데이터 동기화를 고려한 응답 시간 모델에 대해 설명한다. 4장에서는 제안한 방법의 모의실험 및 결과에 대해 분석하고, 5장에서는 결론을 맺는다.

II. 오프로딩 개념과 응답 시간

모바일 클라우드 컴퓨팅에서는 모바일 기기의 자원 제약적인 한계를 극복하기 위해 풍부한 컴퓨팅 파워, 메모리, 저장 장치 등을 갖춘 클라우드 서비스를 활용한다. 이를 통해 모바일 기기에서 실행해야 할 연산 집약적인 작업을 클라우드에게 위임하고 그 결과를 반환받는 오프로딩 기법이 사용된다.

오프로딩의 작업 크기에 따라 부분 오프로딩 기법과 전체 오프로딩 기법으로 나눌 수 있다. 부분 오프로딩 기법은 전체 작업 중에서 일부의 작업을 클라우드에 위임하는 방법이고 전체 오프로딩 기법은 모바일 기기에서는 사용자와의 상호작용만을 처리하고 다른 모든 작업은 클라우드에 수행을 위임하는 방법이다. 부분 오프로딩의 경우 특정 작업이 자주 사용되고 병렬적으로 수행될 수 없는 경우에는 통신비용 및 대기 시간의 증가의 문제가 발생하며 전체 오프로딩의 경우에는 사용자와의 빈번한 상호작용이 발생할 경우 응답성 저하 및 동기화 문제가

발생한다. 따라서 오프로딩에 적합한 작업을 선택적으로 클라우드에 요청하는 것이 필요하다.

응답 시간의 관점에서 오프로딩을 수행하지 않고 작업을 수행했을 때에는 L. Li 등이 제안한 것과 같이 CPU의 클럭 속도에 따라 수식 (1)과 같은 선형 함수로 표현할 수 있다^[7]. α_f 와 β_f 은 기존의 실험 데이터를 바탕으로 CPU의 클럭 속도에 따른 피팅 값(Fitting result)을 의미하고 W 는 주어진 작업의 부하를 의미한다. 따라서 응답 시간은 작업 부하와 CPU의 클럭 속도에 의해 결정된다.

$$RT_{local} = \alpha_f * W + \beta_f \quad (1)$$

오프로딩을 수행 했을 때의 응답 시간은 오프로딩에 필요한 데이터를 클라우드에 전달하는 지연시간(d_{tx})과 클라우드에서 작업 처리한 후에 결과 값을 전달 받는 데 필요한 지연시간(d_{rx})을 포함하고 이는 수식 (2)와 같이 표현할 수 있다. 클라우드를 통해 작업을 위임하는 경우 클라우드에서 작업 처리를 위한 충분한 컴퓨팅 파워를 제공하기 때문에 클라우드 상에서 수행되는 연산 처리시간은 무시할 수 있을 정도로 작다고 가정했다.

$$RT_{offloading} = d_{tx} + d_{rx} \quad (2)$$

III. 제안모델

1. 표기법과 가정

표 1은 본 논문에서 사용하는 표기법들에 대한 설명을 보여준다.

표 1. 표기법
Table 1. Notations

항목	설명
λ_u	새로운 센싱 데이터의 발생비율
T	센싱 데이터 동기화 주기 ($T_n + T_{sync}$)
T_n	클라우드가 오프로딩을 처리할 수 있는 시간
T_{sync}	센서 데이터 동기화 작업으로 오프로딩을 처리할 수 없는 시간
d_{tx}	모바일 기기에서 클라우드로의 데이터 전송시간
d_{rx}	클라우드에서 모바일 기기로의 결과값 전송 시간
d_{wait}	클라우드로의 오프로딩 실패 시 모바일 기기의 대기 시간

제안 모델은 그림 1과 같은 상황을 모델링 하였다. 임의 시간에 모바일 기기들 상에서 새로운 센싱 데이터가 생성되면 클라우드는 새로운 데이터를 반영한 결과 값을 생성하기 위해 새로운 센싱 데이터의 전송을 요청한다. 모바일 기기들이 요청을 받아드리고 이를 클라우드로 전송하는 동기화 작업이 수행되면 클라우드는 해당 데이터를 요구하는 오프로딩 작업을 수행할 수 없게 된다.

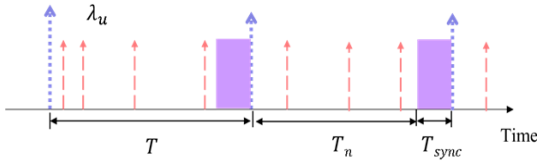


그림 1. 센싱 데이터 생성과 클라우드에서의 동기화
 Fig. 1. Sensing data generation and synchronization in cloud

제안 모델은 단순화를 위해서 아래와 같은 가정 하에 설계되었다.

- 신규 센싱 데이터의 발생비율(λ_u)은 포아송 프로세스(Poisson Process)를 따름
- 새로운 센싱 데이터 처리를 위해 클라우드에서는 주기적(T)으로 동기화 과정을 수행
- 클라우드에서 센싱 데이터 동기화 과정 중에는 모바일 기기의 오프로딩 요청을 처리할 수 없음
- 데이터 동기화 과정은 T_{sync} 의 고정 시간이 소요됨
- 모바일 기기는 오프로딩 요청이 실패하면 d_{wait} 시간 동안 기다린 이후에 재요청

2. 데이터 동기화를 고려한 응답 시간 모델

앞서 가정한 것처럼 새로운 센싱 데이터의 발생 비율은 포아송 프로세스를 따르고 t 시간동안 λ_u 의 비율로 새로운 데이터가 발생한다고 하면 새로운 센싱 데이터가 발생할 확률은 수식 (3)와 같이 정의할 수 있다.

$$E(t) = 1 - e^{-\lambda_u t} \quad (3)$$

새로운 센싱 데이터가 발생하지 않는 동안에는 클라우드의 오프로딩 서비스 가용성이 보장된다고 할 수 있기 때문에 새로운 센싱 데이터가 발생한 경우에는 동기화 작업을 위해서 클라이언트의 요청을 받아 드릴 수 없다. 이러한 클라우드의 오프로딩 서비스 가용성은 수식 (4)와 같이 정의할 수 있다.

$$C(t) = \begin{cases} e^{-\lambda_u t} & \text{if } 0 \leq t < T_n \\ 0 & \text{if } T_n \leq t < T_n + T_{sync} \end{cases} \quad (4)$$

이때 클라우드가 T 주기에 한 번씩 동기화를 수행한다면 평균 클라우드의 오프로딩 서비스 가용성은 수식 (5)와 같이 정의할 수 있다.

$$\begin{aligned} E[C(T)] &= \frac{\int_0^{T_n} C(t) dt}{T} = \frac{\int_0^{T_n} e^{-\lambda_u t} dt}{T} \\ &= \frac{1}{\lambda_u T} \cdot (1 - e^{-\lambda_u T_n}) \end{aligned} \quad (5)$$

수식(5)를 바탕으로 클라우드를 통한 응답시간은 수식 (6)와 같이 계산할 수 있다. 모바일 기기는 클라우드의 오프로딩 서비스를 활용할 수 있을 경우에는 오프로딩 작업에 필요한 데이터를 전송하고 결과 값을 반환 받으면 된다. 그러나 동기화 작업으로 인해 오프로딩 서비스를 활용할 수 없는 경우에는 데이터 전송 이후에 동기화 작업이 종료될 때까지 기다려야 한다. 따라서 센싱 데이터의 발생 빈도와 동기화 주기에 따라 오프로딩 서비스의 응답시간이 변화된다.

$$RT_{offloading} = E[C(T)] \cdot (d_{tx} + d_{rx}) + (1 - E[C(T)]) \cdot (d_{tx} + d_{rx} + \max(T_{sync}, d_{wait})) \quad (6)$$

IV. 모의실험 및 결과분석

본 논문에서 제안한 센싱 데이터 동기화를 고려한 응답 시간 모델의 성능을 평가하기 위해 표 2에서 보이는 바와 같이 실험 파라미터를 설정하고 모의실험을 수행하였다.

표 2. 실험 파라미터
 Table 2. Simulation Parameters

Parameter		설정 값
α_f	CPU 클럭 2.7 GHz	1.83
β_f	일 때 피팅 값	-55.52
W	작업부하	최소: 50 ms 최대: 5000 ms
λ_u	센싱 데이터 생성 비율	0.001 ~ 0.1
T	동기화 주기	10 sec. ~ 1 min.
T_n	오프로딩 서비스 시간	T 의 90%
T_{sync}	동기화 작업 시간	T 의 10%
d_{tx}	데이터 송신 시간	178 ms
d_{rx}	결과 값 수신 시간	178 ms
d_{wait}	대기 시간	T_{sync} 와 동일 값

그림 2는 각 모델별 응답시간에 대한 비교 결과를 보여준다. 모바일 기기 내에서 작업을 수행하는 경우 (Local) 작업의 부하에 따라 응답 시간의 변화가 심하기 때문에 작업 부하가 큰 경우에는 오프로딩 기법을 활용하여 응답시간을 줄일 수 있다. 센서 데이터의 동기화를 고려하지 않는 기존의 모델의 경우(Offloading without sync) 클라우드에서 작업이 수행되기 때문에 작업의 부하에 따라 응답시간이 영향을 받지 않지만 제안 모델 (Our model)의 결과와 비교했을 때 오프로딩 기법의 응답 시간을 정확하게 반영하지 못한다.

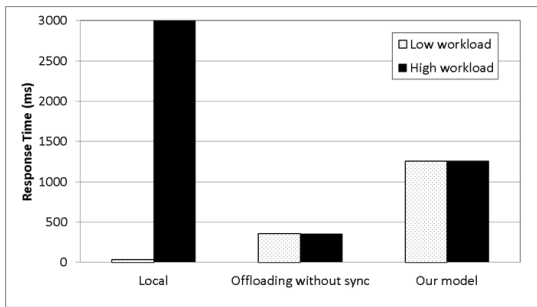


그림 2. 각 모델별 응답 시간
Fig. 2. Response time of each models

그림 3은 제안 모델을 기반으로 센싱 데이터 생성 비율(λ_u)의 변화에 따른 예상 응답 시간의 결과를 보여준다. 센싱 데이터 생성 비율이 0.001에서 0.01 사이의 값을 가지면 응답시간이 급격하게 변화 되는 것을 알 수 있다. 그러나 센싱 데이터 생성 비율이 0.04 보다 크면 응답시간은 크게 변화하지 않는다. 따라서 새로운 데이터가 많이 발생하는 상황에서는 클라우드에게 오프로딩을 요청할 경우 즉시 처리되는 것이 어렵다.

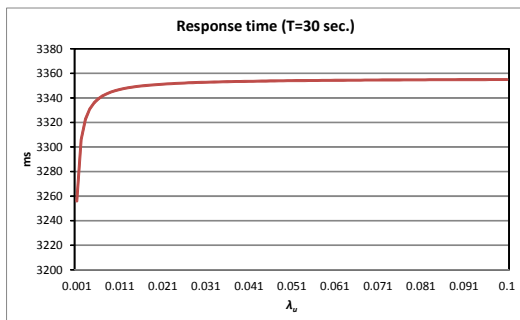


그림 3. 센싱 데이터 생성 비율에 따른 응답시간
Fig. 3. Response time for the sensing data generation rate

그림 4는 제안 모델을 기반으로 클라우드 상에서 동기화 주기(T)에 따른 예상 응답 시간의 결과를 보여준다. 동기화 주기를 10초에서 1분으로 변경하면서 응답 시간의 변화를 살펴보면 센싱 데이터를 동기화하는데 필요한 시간(T_{sync})에 비례하여 증가하는 것을 알 수 있다. 응답 시간의 향상을 위해서는 센싱 데이터의 생성 비율에 따라 최적의 주기를 찾고 동기화에 필요한 작업시간을 줄이는 것이 필요하다.

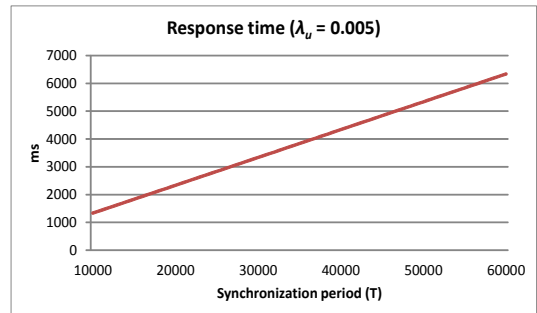


그림 4. 동기화 주기에 따른 응답시간
Fig. 4. Response time for the synchronization period

V. 결론

모바일 클라우드는 모바일 기기의 자원 제약적인 문제를 해결하기 위해 클라우드 서비스를 활용함으로써 상당한 연산 파워와 시간을 필요로 하는 작업을 수행하는데 필요한 에너지를 줄이고 응답 시간을 향상시킬 수 있다. 이러한 오프로딩 기법은 모든 작업에 적용하기 보다는 작업의 특성에 따라 부분적으로 적용하는 것이 더 효율적이다.

본 논문에서는 이러한 모바일 클라우드 환경에서 오프로딩 기법 사용 시 응답 시간을 예측하는 모델을 제안하였다. 제안 모델의 경우 기존 모델에서 모바일 기기와 클라우드 사이의 통신에 필요한 지연 시간만을 응답 시간에 고려한 것에 반해 클라우드에서 요청받은 작업을 수행하는데 필요한 센싱 데이터를 동기화하는 데 필요한 시간을 함께 고려하였다. 모의실험 결과 제안 모델이 기존의 모델에 비해 응답 시간을 정확하게 예측할 수 있고, 센싱 데이터 생성 비율에 따른 최적의 동기화 주기를 설정하는 것이 가능함을 보였다.

References

- [1] N. Fernando, S. Loke, and W. Rahayu, "Mobile cloud computing: A survey," Future Generation Computer Systems, Vol.29, No.1, pp.84~106, Jan. 2013.
- [2] M. Hwang, K. Lee, and S. Yoon, "Software Development Methodology for SaaS Cloud Service," The Journal of The Institute of Internet, Broadcasting and Communication(JIIBC), Vol. 14, No. 1, pp.61~67, Feb. 2014.
- [3] H. T. Dinh, C. Lee, D. Niyato, and P. Wang, "A Survey of Mobile Cloud Computing: Architecture, Applications, and Approaches," Wireless Communications and Mobile Computing, Vol.13, No.18, pp.1587~1611, Dec. 2013.
- [4] K. Kumar, and Y. Lu, "Cloud Computing for Mobile Users: Can Offloading Computation Save Energy?," Computer, Vol.43, No.4, pp.51~56, Apr. 2010.
- [5] W. Lee, and D. Nam, "Volume Rendering Architecture of Mobile Medical Image using Cloud Computing," The Journal of The Institute of Internet, Broadcasting and Communication(JIIBC), Vol. 14, No. 4, pp.101~106, Aug. 2014.
- [6] M. Rahimi et al., "Mobile Cloud Computing: A Survey, State of Art and Future Directions," Mobile Networks and Applications, Vol.19, No.2, pp.133~143, Nov. 2014.
- [7] L. Li, R. Tian, B. Yang, and Z. Gao, "A Model of Web Server's Performance-Power Relationship," Proc. of International Conference on Communication Software and Networks, pp. 260~264, 2009.

저자 소개

민 홍(정회원)



- 2004년 : 한동대학교 전산과학 졸업 (학사).
- 2011년 : 서울대학교 컴퓨터공학부 졸업(박사).
- 2013년 ~ 현재 : 호서대학교 컴퓨터 정보공학부 조교수.

<주관심분야 : 운영체제, 무선 센서 네트워크, 스마트폰 센싱, 임베디드 시스템, 결합허용 시스템>

허 준 영(정회원)



- 1998년 : 서울대학교 컴퓨터공학과 졸업(학사).
- 2009년 : 서울대학교 컴퓨터공학부 졸업(박사).
- 2009년 ~ 현재 : 한성대학교 컴퓨터 공학과 조교수.

<주관심분야 : 운영체제, 무선 센서 네트워크, 임베디드 시스템, 결합허용 시스템, 기계학습>

※ 본 연구는 한성대학교 교내학술연구비 지원 과제임.