Time-Optimized Task Offloading Decision Making in Mobile Edge Computing

Ibrahim Alghamdi
i.alghamdi.1@research.gla.ac.uk

Christos Anagnostopoulos
Christos.Anagnostopoulos@glasgow.ac.uk

Dimitrios P. Pezaros
dimitrios.pezaros@glasgow.ac.uk

School of Computing Science
University of Glasgow, UK

Wireless Days 2019, Manchester UK, April 24th - 26th

University of Glasgow

NETLAB
NETWORKED SYSTEMS RESEARCH LABORATORY

Outline

Background[1]

- The recent advances in mobile devices
  - Example
  - Limitation
- Mobile Cloud Computing (MCC)
  - What is MCC?
  - Limitation
- Mobile Edge Computing (MEC)
  - What is MEC?
  - Names
    - iCloud
    - Fog Computing
    - **Mobile Edge Computing**
  - Use cases

[1] Pavel Mach and Zdenek Becvar. "Mobile edge computing: A survey on architecture and computation offloading". In: *IEEE Communications Surveys & Tutorials* 19.3 (2017), pp. 1628–1656.

Computation Offloading

- Dispatching intensive tasks to an external server, i.e., Cloud or an Edge server.[2]
  - Face/speech recognition;
  - Augmented, assisted or virtual reality;
  - Low latency applications, such as online gaming or remote desktop.
- The authors[3] demonstrated on a real MEC testbed that the reduction of **latency** up to 88% and **energy consumption** of the mobile device up to 93% can be accomplished by the computation/task offloading in MEC.

---

[2] Pavel Mach and Zdenek Becvar. "Mobile edge computing: A survey on architecture and computation offloading". In: *IEEE Communications Surveys & Tutorials* 19.3 (2017), pp. 1628–1656.

[3] Jakub Dolezal, Zdenek Becvar, and Tomas Zeman. "Performance evaluation of computation offloading from mobile device to the edge of mobile network". In: *2016 IEEE Conference on Standards for Communications and Networking (CSCN)*. IEEE. 2016, pp. 1–7.

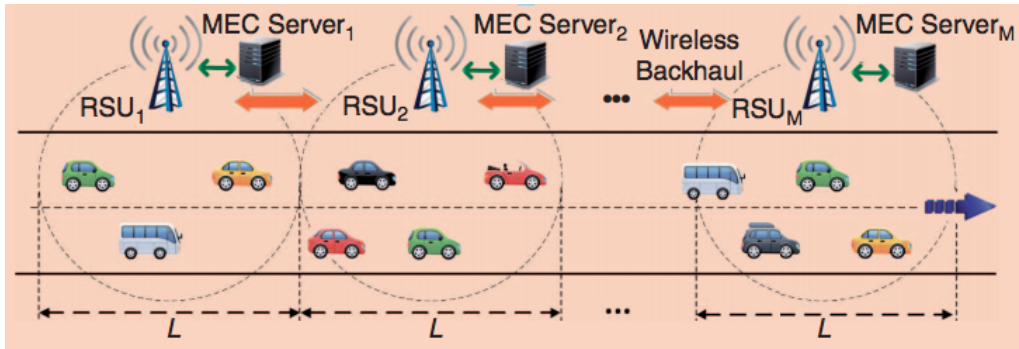
Challenges[4]

- Offloading Sequential Decision Making
  - Doing the tasks **either** locally **or** offloading them
  - Locally, Cloud, or at the Edge
  - **Which Edge server to offload?**
- Mobility Management

[4] Pavel Mach and Zdenek Becvar. "Mobile edge computing: A survey on architecture and computation offloading". In: *IEEE Communications Surveys & Tutorials* 19.3 (2017), pp. 1628–1656.

Previous Work

- Different from previous work, we focus on the **decision of when** to offload to an edge server, i.e., the selection of MEC servers/time.
- ST-CODA[5]: A spatial and temporal computation offloading decision algorithm.
- A predictive off-loading framework in vehicular networks.[6]

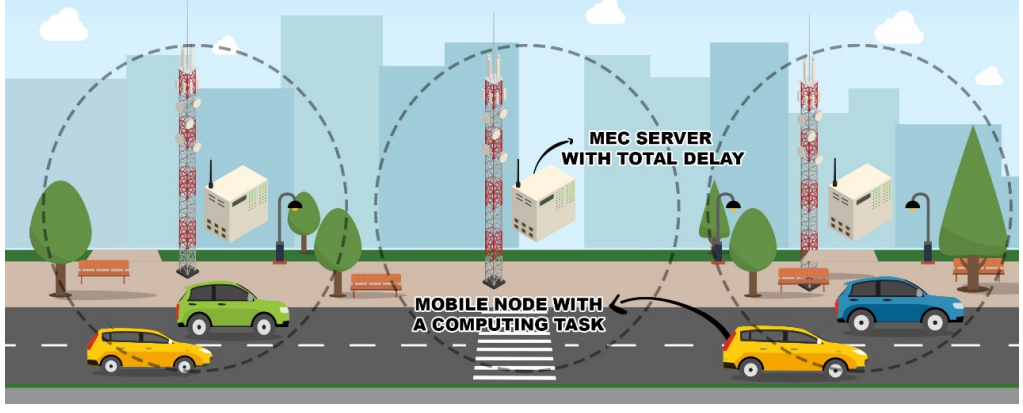


[5] Haneul Ko, Jaewook Lee, and Sangheon Pack. "Spatial and Temporal Computation Offloading Decision Algorithm in Edge Cloud-Enabled Heterogeneous Networks". In: *IEEE Access* 6 (2018), pp. 18920–18932.

[6] Ke Zhang et al. "Mobile-edge computing for vehicular networks: A promising network paradigm with predictive off-loading". In: *IEEE Vehicular Technology Magazine* 12.2 (2017), pp. 36–44.

System Model

- Mobile device
- MEC server
- Computing task with total delay D such that:
  - $D_{offload} < D_{local}$
- $D_{offload}$ delay includes:
  1. Transmission Time;
  2. Processing Time;
  3. Time spent to receive the processed data from MEC server to mobile device.



Problem Statement

A mobile user desires to offload data to an Edge server, and there are many deployed Edge servers in the user path with different total delays. Which is the best offloading strategy at time t^* that:

1. Minimizes the **expected** total delay:

$$\inf_{t>0} \mathbb{E}[D_t] \tag{1}$$

This problem is a sequential decision making solved based on the principles of the Optimal Stopping Theory (OST).

Optimal Stopping Theory (1)

- Concerned with the problem of choosing the **best time instance** to take a given action based on sequentially observed random variables in order to minimize an expected cost.
- We cast our offloading problem as a finite horizon OST problem, in which we know the upper bound n, i.e., the number of stages at which one may stop[7].

[7] Ke Zhang et al. "Mobile-edge computing for vehicular networks: A promising network paradigm with predictive off-loading". In: *IEEE Vehicular Technology Magazine* 12.2 (2017), pp. 36–44.

Optimal Stopping Theory (2)

- The system equation has the form:

$$x_{k+1} = \begin{cases} x_T, & \text{if } x_k = x_T \text{ (stop).} \\ D_k^*, & \text{otherwise (continue).} \end{cases} \tag{2}$$

- Let $J_k(x_k)$ be the cost to offload data/task to the k-th MEC server. By Bellman's equation:

$$J_n(x_n) = x_n \tag{3}$$

for $k = n$, and

$$J_k(x_k) = \min \left[(1+r)^{n-k} x_k, \mathbb{E}[J_{k+1}(D_k^*)] \right] \tag{4}$$

for $k = 1, ..., n-1$, with factor $r \in (0, 1)$.

Optimal Stopping Theory (3)

$$J_k(x_k) = \min\left[(1+r)^{n-k}x_k, \mathbb{E}[J_{k+1}(D_k^*)]\right] \tag{4}$$

- Factor $r \in (0,1)$ is a *delay* parameter, which prompts us to delay/speed our optimal decision.
- The term $(1+r)^{n-k}$ denotes the risk if the offloading happens at k and $\mathbb{E}[J_{k+1}(s_k^*)]$ denotes the expected risk if we continues the observation process.
- **Rule:** It is optimal to stop at stage k iff

$$x_k \leq a_k = \frac{\mathbb{E}[J_{k+1}(D_k^*)]}{(1+r)^{n-k}}, \tag{5}$$

else, it is optimal to continue.

Optimal Stopping Theory (4)

The optimal stopping rule is determined by the scalar values $a_1, a_2, \ldots, a_n$ through which the mobile node decides either to offload or not:

Optimal Task Offloading Rule

Offload the data at the k-th MEC server if $D_k \leq a_k$; otherwise continue the observation if $D_k > a_k$.

Optimal Stopping Theory (5)

The scalar a_k values are calculated once through *backward induction* using (6) and (7).

$$a_k = \frac{1}{1+r}\left(a_{k+1}(1 - F_D(a_{k+1})) + \int_0^{a_{k+1}} u dF_D(u)\right) \quad (6)$$

$$a_n = \frac{1}{1+r}\int_0^1 u dF_D(u) = \frac{1}{1+r}\mathbb{E}[D], \quad (7)$$

where $F_D(u) = P(D \leq u)$ is the cumulative distribution function of the total delay D.

Summary of the Model

Input: Decision scalar values $a_1, a_2, ..., a_n$
Output: Decision of which MEC server to offload

Offload $\leftarrow$ FALSE
for $k = 1 : n$ **do**
  if current total delay $D_k \leq a_k$ **then**
    MEC-Server $\leftarrow k$;
    Offload $\leftarrow$ TRUE; break;
  end if
end for
if Offload == FALSE **then**
  MEC-Server $\leftarrow n$;
end if
Offload tasks/data to the MEC-Server;

Data Sets

- Real mobility trace for many users in a campus[8]

Timestamp	associated AP
1026840585	AdmBldg16AP1

Table 1: Dataset Format

- Mobile Netwrok Dataset[9]
  - Cellular traffic volume observed every hour (phone calls, SMS and Internet communication) for each cell.
- Cell ids from the Mobile Netwrok Dataset are mapped to the APs in the mobility trace.
- For example, we assume that the cell1 to be the access point AdmBldg16AP1.

[8] David Kotz et al. *CRAWDAD dataset dartmouth/campus (v. 2009-09-09)*. Downloaded from https://crawdad.org/dartmouth/campus/20090909/movement. traceset: movement. Sept. 2009. DOI: 10.15783/C7F59T.

[9] Gianni Barlacchi et al. "A multi-source dataset of urban life in the city of Milan and the Province of Trentino". In: *Scientific data* 2 (2015), p. 150055.

Data Sets, Cont'd

- Mobility trace & Cell tower dataset with Internet traffic.
- Map an access point to a cell tower.
- We considered one day interval, i.e., for each user, we take the movements for each day and run our model.
- **Goal:** Select the minimum expected total delay (load).

Time	Access Point
1043522712	AcadBldg18AP2
1043523266	AcadBldg18AP3
1043523287	AcadBldg10AP15
1043523792	AcadBldg10AP12

Time	Cell ID	Internet Traffic
06/11/2013 01.00	1	10.466
06/11/2013 01.00	2	20.1029
06/11/2013 01.00	3	13.2935
06/11/2013 01.00	4	42.45

Time1	Time2	Access Point	Cell ID	Cell Internet Load
1043522712	04/12/2002 23.38	AcadBldg18AP2	1	10.466
1043523266	05/12/2002 22.52	AcadBldg18AP3	2	20.1029
1043523287	20/12/2002 22:52	AcadBldg10AP15	3	13.2935
1043523792	20/12/2002 22:55	AcadBldg10AP12	4	42.45

Result(1)

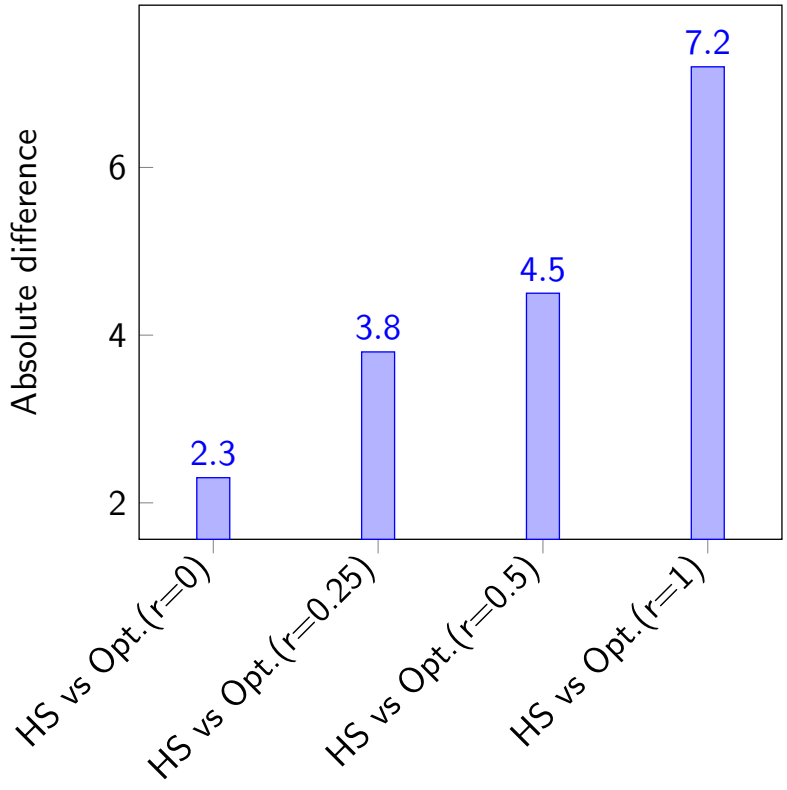


Figure 1: Absolute difference between Optimal and OST-model; σ and μ are taken from the same interval.

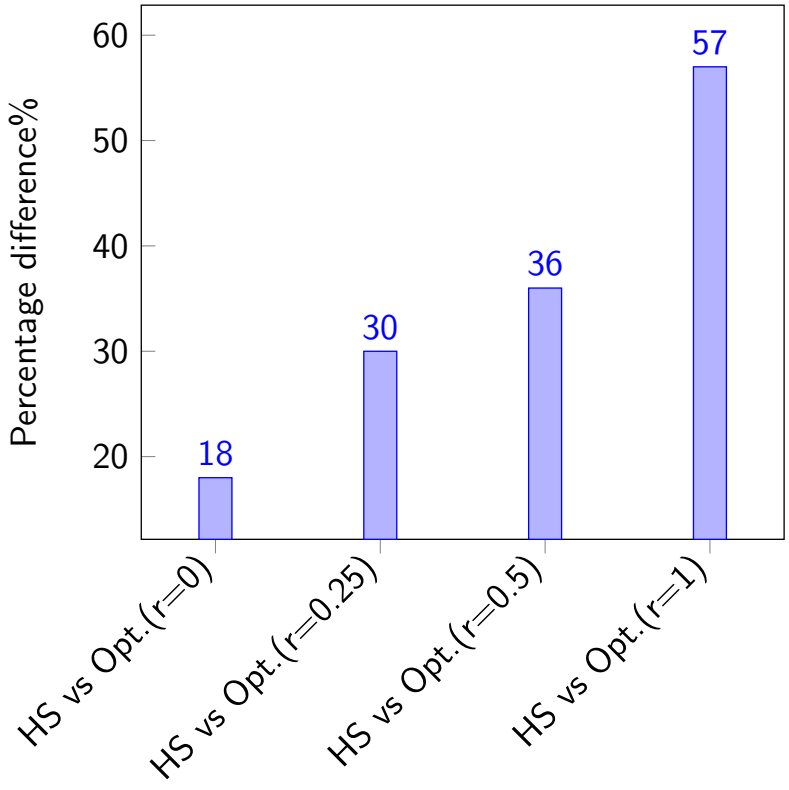


Figure 2: Percentage difference between the Optimal and OST-model; σ and μ are taken from the same interval.

Result(2)

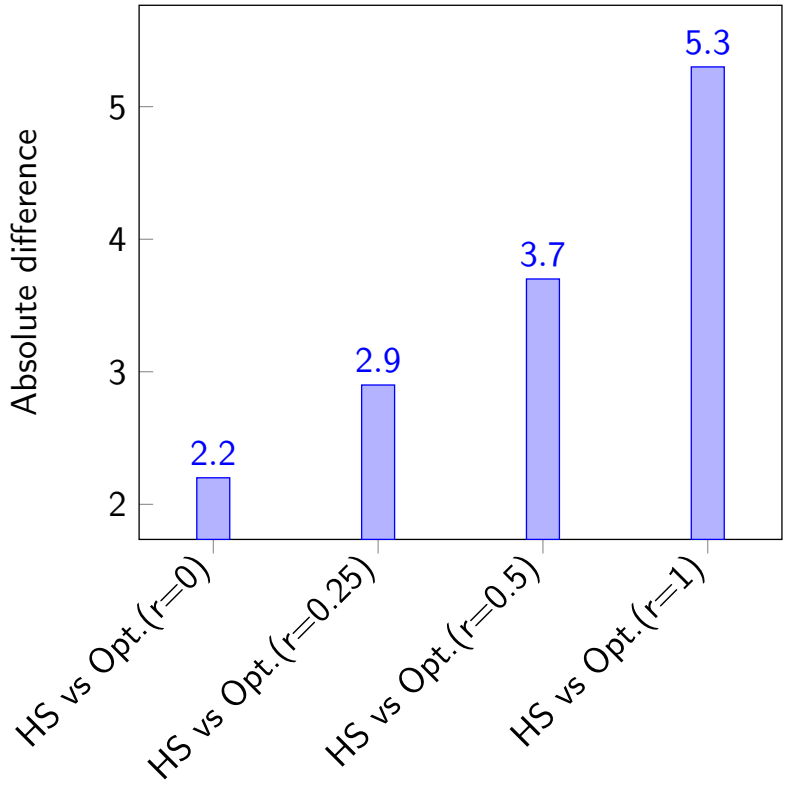


Figure 3: Absolute difference between Optimal and OST-model; σ and μ are taken from the all traces.

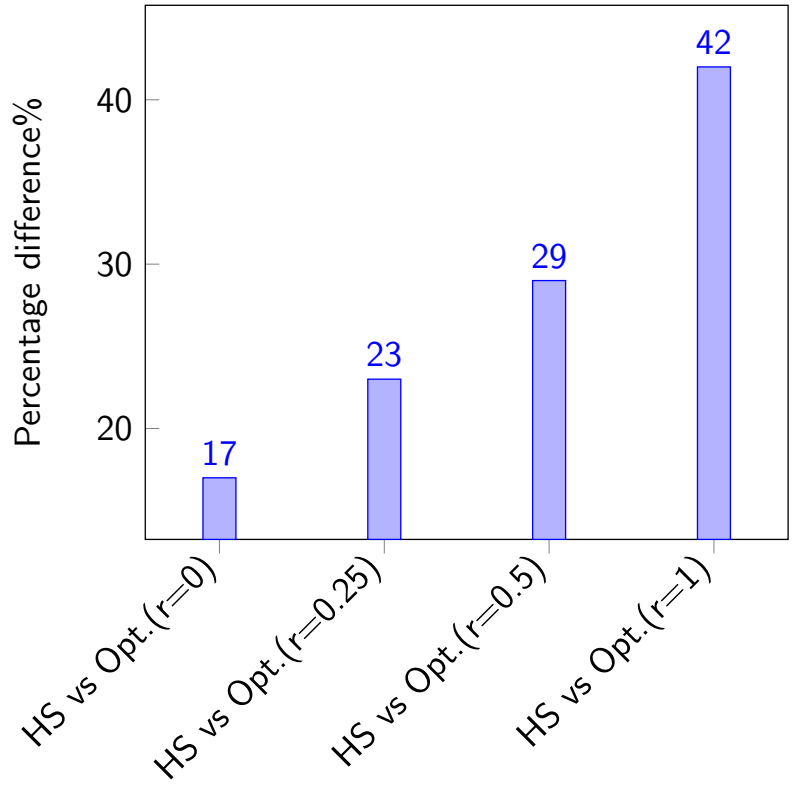


Figure 4: Percentage difference between Optimal and OST-model; σ and μ are taken from the all traces.

Results (3)

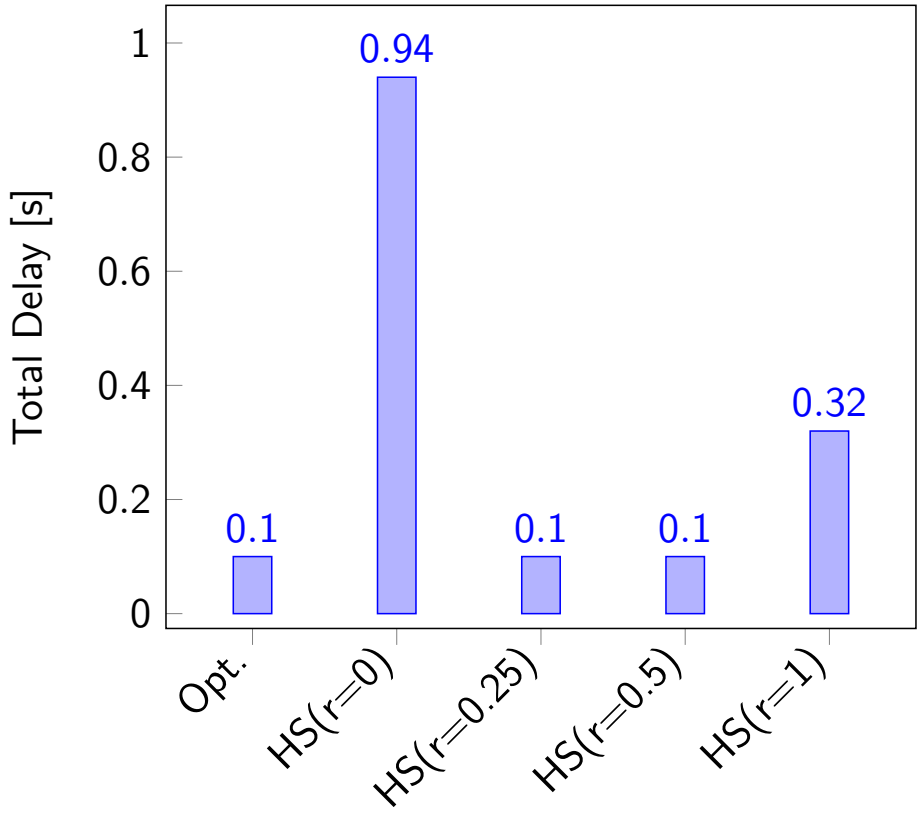


Figure 5: The Optimal and the OST-model for one interval for different delay factors r.

Future Work and Conclusions

- We aim to focus on the case where there is a deadline by investigating the value of the delay factor r and how it can be adapted for different uses cases.
- Consider the case where the number of the server is **unknown and not provided** to the mobile nodes.

The End

Thank you!

Contact: Ibrahim Alghamdi
Email address: i.alghamdi.1@research.gla.ac.uk