OSAKA UNIVERSITY

Effects of Service Function Relocation on Application-level Delay in Multi-access Edge Computing

Junichi Kaneda, Shin'ichi Arakawa, Masayuki Murata

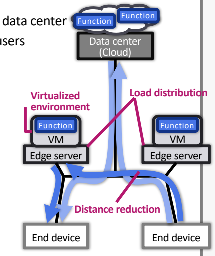
Graduate School of Information Science and Technology,
Osaka University, Japan

5G Technologies IX
IEEE 5G World Forum

July 10, 2018

Expectations for Multi-access Edge Computing (MEC)

- **Improve responsiveness**
  - Data is processed at **edge server** rather than data center
  - Multiple edge servers are deployed close to users
    - Long communication distance can be reduced
    - Load can be distributed
- **Provide services flexibly**
  - Service functions are deployed on virtual machines (VMs) in virtualized environment
  - VM locations are changed flexibly
    - By live migration of VM



Service Application for MEC

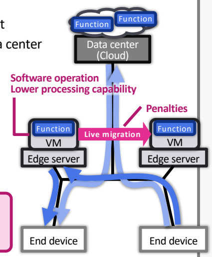
- **Monitoring agent service using augmented reality (AR)**
  - Robots go to a physical place
    - The robots are equipped with sensors and cameras
  - Users can monitor from home, as if they were there
  - Using AR, object information is added to the video taken by the robots
    - AR services requires low latency
      - Because it is needed to analyze, process and display video in real-time
    - Hosting AR function is expected to reduce RTT and ensure high bandwidth

---

Concerns about Responsiveness in MEC Environment

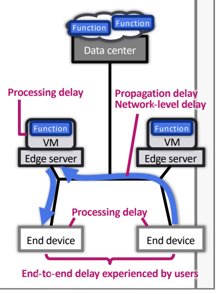
- **Increase of processing delay**
  - Software operation in virtualized environment
  - Lower processing capability compared to data center
- **Penalties of VM live migration**
  - Temporary delays or packet loss may occur
    - during connection re-establishment (service downtime)

For deployment of MEC for future services, it is important to investigate whether responsiveness improves as expected



Application-level delay

- **End-to-end delay experienced by users**
  - Includes
    - Propagation delay at Layer 2
    - Network-level delay in socket at Layer 3 and 4
    - Processing delay at Layer 7
- **Relate to quality of experience (QoE)**
  - Our experiment will contribute to understanding QoE in MEC environments
  - QoE metrics include delay experienced by users



Purpose and Approach

- **Purpose**

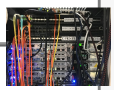
  Investigating the effects of service function relocation on application-level delay in MEC environment

- **Approach**

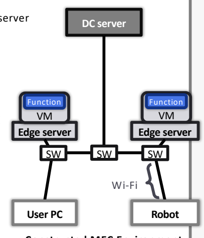
  Constructing MEC environment and experimenting with it

- **Steps**
  1. Construct MEC environment using OpenStack
  2. Implement MEC service
  3. Measure application-level delay and reveal its factors at each node
  4. Investigate effects and penalties of live migration

Construction of the MEC Environment



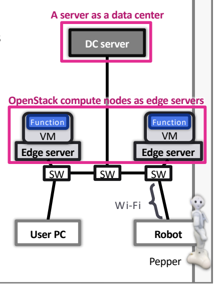
- **Connect servers, a PC and a robot with switches**
  - Constructed in a LAN
    - In actual environment, the network delay to DC server (data center) is about 100 ms
- **Use OpenStack for MEC's virtualization environment**
  - OpenStack is open source software for building virtualization environments
  - Use OpenStack compute nodes as edge servers
    - Service functions are deployed on VM



Constructed MEC Environment

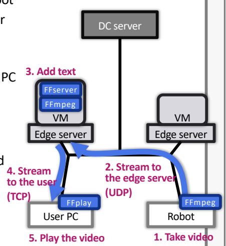
Nodes in the MEC Environment

- **Edge servers**
  - Set up at both of user side and robot side
  - Implemented using OpenStack compute nodes
  - Run functions in virtualized environment
- **DC server**
  - Operates as a data center
  - Runs functions **not** in virtualized environment
- **User PC**
  - General PC
- **Robot**
  - "Pepper" manufactured by SoftBank Robotics
  - Camera is equipped
  - Connected to network via Wi-Fi



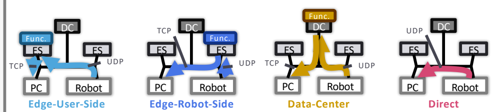
Implemented MEC Service

- **Video live streaming form robot to user**
  1. Take video using camera equipped with robot
  2. Live-stream video form robot to edge server
  3. Add text on video at edge server
     - Simple video processing considering AR
  4. Live-stream video form edge server to user PC
  5. Play video at user PC
- **Using FFmpeg, FFserver and FFplay**
  - FFserver uses UDP and TCP for reception and transmission, respectively, because of its specification



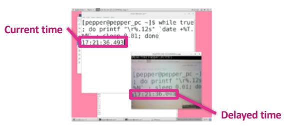
Four Scenarios with Different Service Provision Forms

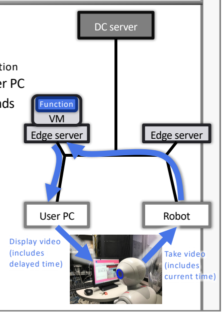
- **MEC scenarios**
  - Edge-User-Side
    - Use the edge server of user side
      - Virtualized environment
    - TCP path length: Short
  - Edge-Robot-Side
    - Use the edge server of robot side
      - Virtualized environment
    - TCP path length: Long
- **Comparisons**
  - Data-Center
    - Use DC server
      - Non-virtualized environment
    - TCP path length: Medium
  - Direct
    - Stream directly from robot to PC
    - Aim to measure processing time at end devices



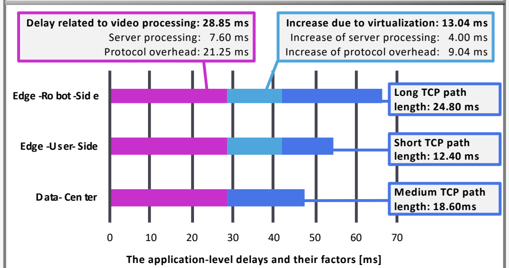
The Way of Application-level Delay Measurement

- **Delay of live streaming video**
  1. Display digital clock in front of robot
     - It is displayed on user PC for time synchronization
  2. Display streamed video next to clock on user PC
  3. Take a screenshot per second for 100 seconds
  4. Calculate the difference between the two displayed times in each screenshot
  5. Calculate the average





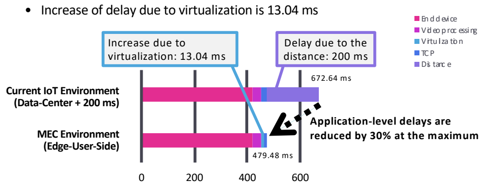
Result of Application-level Delay Measurement



Difference from the result of scenario Direct (425.19 ms) is shown

Comparison between Current IoT Environment and MEC

- **Delay to data center occurs in the order of 100 ms***
  - Assume delay due to the distance is 200 ms at the maximum
  - Increase of delay due to virtualization is 13.04 ms



Responsiveness is improved by providing services using edge servers

*RTTs to AWS data centers
From Osaka to Tokyo: 13 ms (400 km)
From Osaka to Ohio: 174 ms (10800 km)
From Osaka to Paris: 272 ms (9600 km)

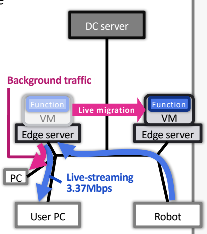
Consideration on Processing Time at End Device

- **Delay of about 400 ms occurs in the robot**
  - The result of scenario Direct is 425.19 ms
    - Most of this delay is caused by compression of the video in the robot
- **In the future, the delay expected to be reduced to about 40 ms**
  - That is when Core i7 or equivalent CPU is applied to robot products
    - FLOPS of CPU on Pepper robot is about one tenth of Intel Core i7
      - Pepper robot is equipped with Intel Atom E3845
  - Lead to increase the proportion of delay occurring in the network

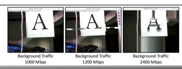
Considering improvement of end device performance, it will be effective to provide services using edge servers

Live Migration Scenario and Setting

- **Generate background traffic**
  - To cause packet loss at edge server of user side
  - Increase monotonically
  - Use 1 Gbps network interface
    - Traffic of live streaming is about 3.37 Mbps
- **Live-migrate VM from user side to robot side based on total amount of traffic**
  - Start when total traffic exceeds 950 Mbps
    - Over 1 Gbps traffic causes packet loss due to exceeding the capacity of the interface
    - Use SNMP to monitor the interface
  - Traffic of migration uses out-of-band network
    - Management network for communication among OpenStack nodes
    - Migration is not interfered by background traffic

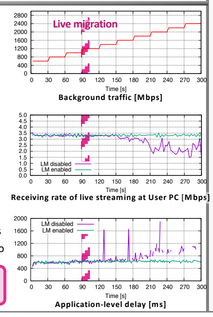


Impact of Live Migration



- **Application-level delay and packet loss**
  - Significant delay or packet loss occurred with increase of background traffic
  - Application-level delay clearly worsen when receiving rate drops to 80%
    - Extremely large application-level delays
    - Damaged video frames
- **Penalty of VM live migration**
  - The penalty of migration is not large
    - Time required to migration: about 13 seconds
      - Without increase of delay and packet loss
    - Communication downtime: about 0.5 seconds
      - With temporary blocking artifacts on video

Service function relocation is useful for maintaining application-level delay



Conclusion and Future Work

- **Conclusion**
  - Purpose
    - Investigating the effects of service function relocation on application-level delay in MEC environment
  - Approach
    - Constructing MEC environment and experimenting with it
  - Result and evaluation
    - Providing services using edge servers can reduce application-level delays by 30% at the maximum
    - Service function relocation is useful for maintaining application-level delay
- **Future work**
  - Perform live migration and evaluate the effects at larger scales
    - Construct MEC environment in metropolitan area network (MAN) or wide area network (WAN)