Joint Allocation of Computing and Wireless Resources to Autonomous Devices in Mobile Edge Computing

Slađana Jošilo, György Dán

Department of Network and Systems Engineering
School of Electrical Engineering and Computer Science
KTH Royal Institute of Technology

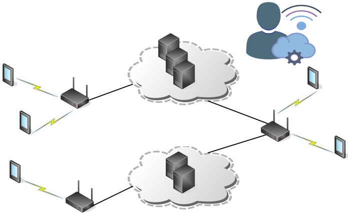
Budapest, August 20, 2018

Mobile Edge Computing (MEC)

Mobile Edge Computing (MEC)

Enabler of 5G

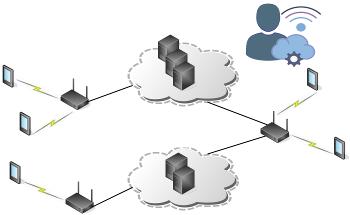
- High bandwidth and computing resources close to the end users
- Interaction between end users decisions and MEC infrastructure decisions



Mobile Edge Computing (MEC)

Enabler of 5G

- High bandwidth and computing resources close to the end users
- Interaction between end users decisions and MEC infrastructure decisions

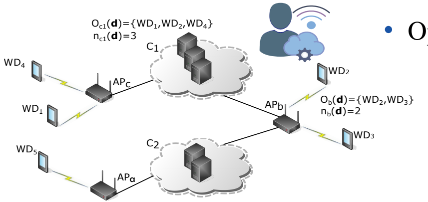


Important Question

- How users and the infrastructure interact with each other?

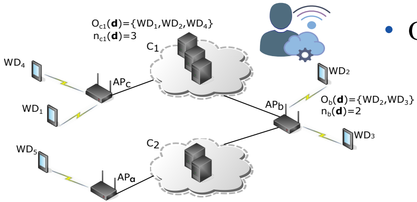
MEC System

MEC System



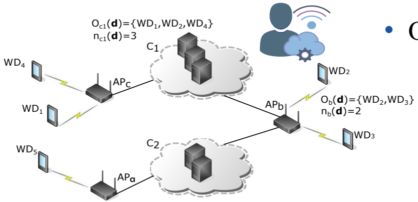
- Operator that manages
  - multiple edge clouds $\mathcal{C} = \{1, 2, ..., C\}$
  - multiple APs $\mathcal{A} = \{1, 2, ..., A\}$

MEC System



- Operator that manages
  - multiple edge clouds $\mathcal{C} = \{1, 2, ..., C\}$
  - multiple APs $\mathcal{A} = \{1, 2, ..., A\}$
  - multiple wireless devices (WDs) $\mathcal{N} = \{1, 2, ..., N\}$

MEC System

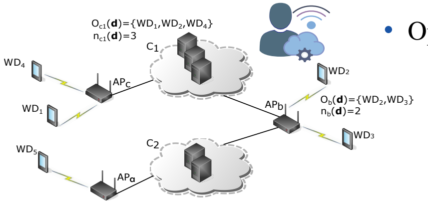


- Operator that manages
  - multiple edge clouds $\mathcal{C} = \{1, 2, ..., C\}$
  - multiple APs $\mathcal{A} = \{1, 2, ..., A\}$
  - multiple wireless devices (WDs) $\mathcal{N} = \{1, 2, ..., N\}$

Computation Offloading

- Task of WD i, $< D_i, L_i >$
  - size of the input data D_i
  - computational complexity L_i

MEC System

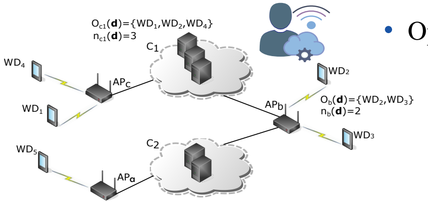


- Operator that manages
  - multiple edge clouds $\mathcal{C} = \{1, 2, ..., C\}$
  - multiple APs $\mathcal{A} = \{1, 2, ..., A\}$
  - multiple wireless devices (WDs) $\mathcal{N} = \{1, 2, ..., N\}$

Computation Offloading

- Task of WD i, $< D_i, L_i >$
  - size of the input data D_i
  - computational complexity L_i
- Decision of WD $i \in \mathcal{N}$ is $d_i = (c, a), c \in \mathcal{C}, a \in \mathcal{A}_c$
- WDs can communicate with cloud c through APs $\mathcal{A}_c \subseteq \mathcal{A}$

MEC System

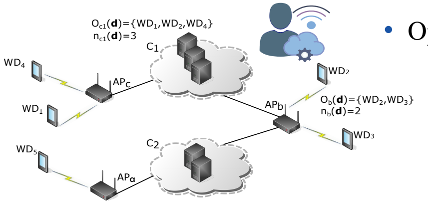


- Operator that manages
  - multiple edge clouds $\mathcal{C} = \{1, 2, ..., C\}$
  - multiple APs $\mathcal{A} = \{1, 2, ..., A\}$
  - multiple wireless devices (WDs) $\mathcal{N} = \{1, 2, ..., N\}$

Computation Offloading

- Task of WD i, $< D_i, L_i >$
  - size of the input data D_i
  - computational complexity L_i
- Decision of WD $i \in \mathcal{N}$ is $d_i = (c, a), c \in \mathcal{C}, a \in \mathcal{A}_c$
- WDs can communicate with cloud c through APs $\mathcal{A}_c \subseteq \mathcal{A}$
- Set of decisions for all WDs is a *strategy profile* $\mathbf{d}$

MEC System

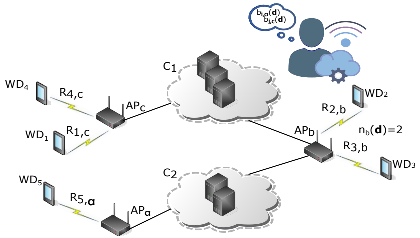


- Operator that manages
  - multiple edge clouds $\mathcal{C} = \{1, 2, ..., C\}$
  - multiple APs $\mathcal{A} = \{1, 2, ..., A\}$
  - multiple wireless devices (WDs) $\mathcal{N} = \{1, 2, ..., N\}$

Computation Offloading

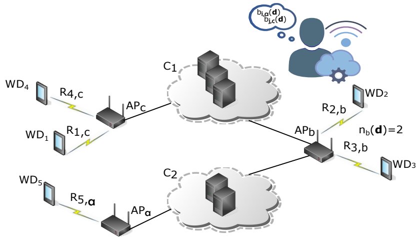
- Task of WD i, $< D_i, L_i >$
  - size of the input data D_i
  - computational complexity L_i
- Decision of WD $i \in \mathcal{N}$ is $d_i = (c, a), c \in \mathcal{C}, a \in \mathcal{A}_c$
- WDs can communicate with cloud c through APs $\mathcal{A}_c \subseteq \mathcal{A}$
- Set of decisions for all WDs is a *strategy profile* $\mathbf{d}$
- $O_a(\mathbf{d})$: set of offloaders via AP a in strategy profile $\mathbf{d}$, $n_a(\mathbf{d}) = |O_a(\mathbf{d})|$
- $O_c(\mathbf{d})$: set of offloaders to cloud c in strategy profile $\mathbf{d}$, $n_c(\mathbf{d}) = |O_c(\mathbf{d})|$

Wireless Resource Management



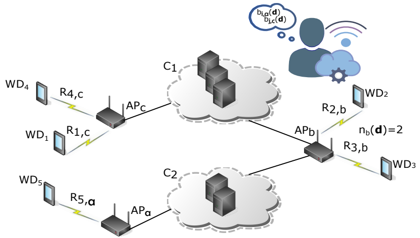
- $R_{i,a}$: PHY rate of WD i on AP a

Wireless Resource Management



- $R_{i,a}$: PHY rate of WD i on AP a
- $n_a(\mathbf{d})$: number of offloaders via AP a
- $b_{i,a}(\mathbf{d})$: uplink access provisioning coefficient

Wireless Resource Management



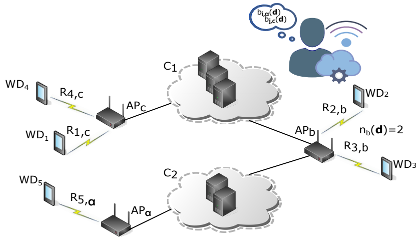
- $R_{i,a}$: PHY rate of WD i on AP a
- $n_a(\mathbf{d})$: number of offloaders via AP a
- $b_{i,a}(\mathbf{d})$: uplink access provisioning coefficient

Computation Offloading through AP a

- Uplink rate of WD i via AP a

$$\omega_{i,a}(\mathbf{d}, \mathbf{b}_a) = \frac{R_{i,a}}{n_a(\mathbf{d}) b_{i,a}(\mathbf{d})}$$

Wireless Resource Management



- $R_{i,a}$: PHY rate of WD i on AP a
- $n_a(\mathbf{d})$: number of offloaders via AP a
- $b_{i,a}(\mathbf{d})$: uplink access provisioning coefficient

Computation Offloading through AP a

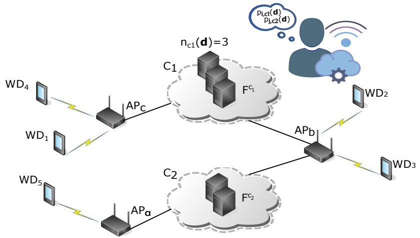
- Uplink rate of WD i via AP a

$$\omega_{i,a}(\mathbf{d}, \mathbf{b}_a) = \frac{R_{i,a}}{n_a(\mathbf{d}) b_{i,a}(\mathbf{d})}$$

- Transmission time of WD i for offloading via AP a

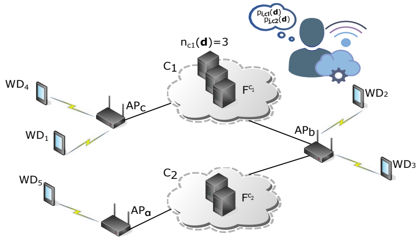
$$T_{i,a}^{off}(\mathbf{d}, \mathbf{b}_a) = \frac{D_i}{\omega_{i,a}(\mathbf{d}, \mathbf{b}_a)}$$

Computing Resource Management



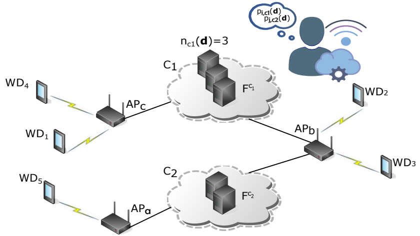
- F^c : computing capability of cloud c

Computing Resource Management



- F^c : computing capability of cloud c
- $n_c(\mathbf{d})$: number of offloaders to cloud c
- $p_{i,c}(\mathbf{d})$: computing power provisioning coefficient

Computing Resource Management



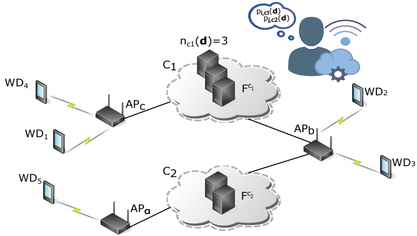
- F^c : computing capability of cloud c
- $n_c(\mathbf{d})$: number of offloaders to cloud c
- $p_{i,c}(\mathbf{d})$: computing power provisioning coefficient

Computation Offloading to Cloud c

- Computing capability allocated to WD i by cloud c

$$F_i^c(\mathbf{d}, \mathbf{p}_c) = \frac{F^c}{n_c(\mathbf{d}) p_{i,c}(\mathbf{d})}$$

Computing Resource Management



- F^c : computing capability of cloud c
- $n_c(\mathbf{d})$: number of offloaders to cloud c
- $p_{i,c}(\mathbf{d})$: computing power provisioning coefficient

Computation Offloading to Cloud c

- Computing capability allocated to WD i by cloud c

$$F_i^c(\mathbf{d}, \mathbf{p}_c) = \frac{F^c}{n_c(\mathbf{d})p_{i,c}(\mathbf{d})}$$

- Execution time of WD i's task in cloud c

$$T_{i,c}^{exe}(\mathbf{d}, \mathbf{p}_c) = \frac{L_i}{F_i^c(\mathbf{d},\mathbf{p}_c)}$$

Cost Model

- Task completion time minimization

Cost of WD i

$$C_{i,a}^{c}(\mathbf{d},\mathbf{b}_a,\mathbf{p}_c) = T_{i,a}^{off}(\mathbf{d},\mathbf{b}_a) + T_{i,c}^{exe}(\mathbf{d},\mathbf{p}_c)$$

System Cost

$$C(\mathbf{d},\mathbf{b},\mathbf{p}) = \sum_{(c,a)\in\mathcal{C}\times\mathcal{A}_c}\ \sum_{i\in O_c(\mathbf{d})\cap O_a(\mathbf{d})} C_{i,a}^{c}(\mathbf{d},\mathbf{b}_a,\mathbf{p}_c)$$

Fundamental Questions

Fundamental Questions

1. How does an operator allocate resources to selfish devices?

Fundamental Questions

1. How does an operator allocate resources to selfish devices?
2. Is there an allocation of tasks in which all selfish devices are satisfied?

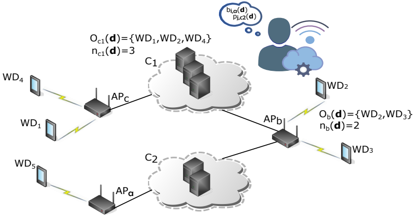
Fundamental Questions

1. How does an operator allocate resources to selfish devices?
2. Is there an allocation of tasks in which all selfish devices are satisfied?
3. Can it be computed using a decentralized algorithm?

Fundamental Questions

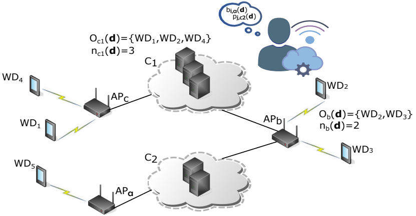
1. How does an operator allocate resources to selfish devices?
2. Is there an allocation of tasks in which all selfish devices are satisfied?
3. Can it be computed using a decentralized algorithm?
4. How good is the system performance?
5. What is the complexity of the algorithm?

Mobile Edge Computation Offloading Game (MEC-OG)



- Multi-leader common-follower Stackelberg game

Mobile Edge Computation Offloading Game (MEC-OG)



- Multi-leader common-follower Stackelberg game

Objective of the Operator

- Joint optimization of wireless and computing resources

$$\min_{\mathbf{b},\mathbf{p}\succeq 0} C(\mathbf{d},\mathbf{b},\mathbf{p})$$

$$\text{s.t.} \sum_{i\in O_a(\mathbf{d})} \frac{1}{b_{i,a}(\mathbf{d})} = n_a(\mathbf{d}), \forall a \in \mathcal{A}$$

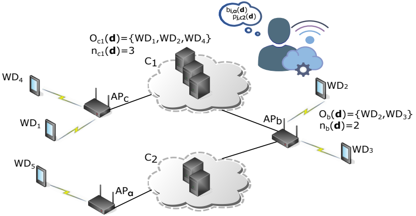
$$\sum_{i\in O_c(\mathbf{d})} \frac{1}{p_{i,c}(\mathbf{d})} = n_c(\mathbf{d}), \forall c \in \mathcal{C}$$

Objective of WDs

- Minimization of their own cost

$$\min_{d_i\in\mathfrak{D}_i} C_i(d_i, d_{-i}, \mathbf{b}_a^*, \mathbf{p}_c^*)$$

Mobile Edge Computation Offloading Game (MEC-OG)



- Multi-leader common-follower Stackelberg game

Objective of the Operator

- Joint optimization of wireless and computing resources

$$\min_{\mathbf{b},\mathbf{p}\succeq 0} C(\mathbf{d},\mathbf{b},\mathbf{p})$$

$$\text{s.t.} \sum_{i\in O_a(\mathbf{d})} \frac{1}{b_{i,a}(\mathbf{d})} = n_a(\mathbf{d}), \forall a \in \mathcal{A}$$

$$\sum_{i\in O_c(\mathbf{d})} \frac{1}{p_{i,c}(\mathbf{d})} = n_c(\mathbf{d}), \forall c \in \mathcal{C}$$

Objective of WDs

- Minimization of their own cost

$$\min_{d_i\in\mathfrak{D}_i} C_i(d_i, d_{-i}, \mathbf{b}_a^*, \mathbf{p}_c^*)$$

Important Question

- Does the MEC-OG have a subgame perfect equilibrium (SPE)?

Optimal Resource Allocation Policy of the Operator

- Best response of the operator to strategy profile **d** chosen by WDs

$$b^*_{i,a}(\mathbf{d}) = \frac{\sum_{j \in O_a(\mathbf{d})} \sqrt{D_j/R_{j,a}}}{n_a(\mathbf{d})\sqrt{D_i/R_{i,a}}}, \forall i \in O_a(\mathbf{d}), \forall a \in \mathcal{A}$$

$$p^*_{i,c}(\mathbf{d}) = \frac{\sum_{j \in O_c(\mathbf{d})} \sqrt{L_j/F^c}}{n_c(\mathbf{d})\sqrt{L_i/F^c}}, \forall i \in O_c(\mathbf{d}), \forall c \in \mathcal{C}$$

Optimal Resource Allocation Policy of the Operator

- Best response of the operator to strategy profile **d** chosen by WDs

$$b^*_{i,a}(\mathbf{d}) = \frac{\sum_{j\in O_a(\mathbf{d})}\sqrt{D_j/R_{j,a}}}{n_a(\mathbf{d})\sqrt{D_i/R_{i,a}}}, \forall i \in O_a(\mathbf{d}), \forall a \in \mathcal{A}$$

$$p^*_{i,c}(\mathbf{d}) = \frac{\sum_{j\in O_c(\mathbf{d})}\sqrt{L_j/F^c}}{n_c(\mathbf{d})\sqrt{L_i/F^c}}, \forall i \in O_c(\mathbf{d}), \forall c \in \mathcal{C}$$

Interaction Between WDs under Optimal Policy of the Operator

- Original player-specific weighted congestion game can be transformed into a congestion game $\Gamma = <\mathcal{N}, (\mathfrak{D}_i)_i, (C_i)_i>$ with resource dependent weights

weights: $$\omega_{i,a} = \sqrt{\frac{D_i}{R_{i,a}}}, \omega_{i,c} = \sqrt{\frac{L_i}{F^c}}$$

Optimal Resource Allocation Policy of the Operator

- Best response of the operator to strategy profile **d** chosen by WDs

$$b^*_{i,a}(\mathbf{d}) = \frac{\sum_{j\in O_a(\mathbf{d})}\sqrt{D_j/R_{j,a}}}{n_a(\mathbf{d})\sqrt{D_i/R_{i,a}}}, \forall i \in O_a(\mathbf{d}), \forall a \in \mathcal{A}$$

$$p^*_{i,c}(\mathbf{d}) = \frac{\sum_{j\in O_c(\mathbf{d})}\sqrt{L_j/F^c}}{n_c(\mathbf{d})\sqrt{L_i/F^c}}, \forall i \in O_c(\mathbf{d}), \forall c \in \mathcal{C}$$

Interaction Between WDs under Optimal Policy of the Operator

- Original player-specific weighted congestion game can be transformed into a congestion game $\Gamma =< \mathcal{N}, (\mathfrak{D}_i)_i, (C_i)_i >$ with resource dependent weights

weights: $\omega_{i,a} = \sqrt{\frac{D_i}{R_{i,a}}}, \omega_{i,c} = \sqrt{\frac{L_i}{F^c}}$

cost of WD i: $C^c_{i,a}(\mathbf{d}) = \omega_{i,a}\sum_{j\in O_a(\mathbf{d})}\omega_{j,a} + \omega_{i,c}\sum_{j\in O_c(\mathbf{d})}\omega_{j,c}$

Optimal Resource Allocation Policy of the Operator

- Best response of the operator to strategy profile **d** chosen by WDs

$$b^*_{i,a}(\mathbf{d}) = \frac{\sum_{j\in O_a(\mathbf{d})}\sqrt{D_j/R_{j,a}}}{n_a(\mathbf{d})\sqrt{D_i/R_{i,a}}}, \forall i \in O_a(\mathbf{d}), \forall a \in \mathcal{A}$$

$$p^*_{i,c}(\mathbf{d}) = \frac{\sum_{j\in O_c(\mathbf{d})}\sqrt{L_j/F^c}}{n_c(\mathbf{d})\sqrt{L_i/F^c}}, \forall i \in O_c(\mathbf{d}), \forall c \in \mathcal{C}$$

Interaction Between WDs under Optimal Policy of the Operator

- Original player-specific weighted congestion game can be transformed into a congestion game $\Gamma =< \mathcal{N}, (\mathfrak{D}_i)_i, (C_i)_i >$ with resource dependent weights

weights: $\omega_{i,a} = \sqrt{\frac{D_i}{R_{i,a}}}, \omega_{i,c} = \sqrt{\frac{L_i}{F^c}}$

cost of WD i: $C^c_{i,a}(\mathbf{d}) = \omega_{i,a}\sum_{j\in O_a(\mathbf{d})}\omega_{j,a} + \omega_{i,c}\sum_{j\in O_c(\mathbf{d})}\omega_{j,c}$

Important Question

- Does the resulting strategic game have a Nash equilibrium (NE)?

Main Results

NE Existence

- Weighted congestion game Γ played between WDs has a NE $\mathbf{d}^*$
  - proof based on showing that the game has an exact potential function

Main Results

NE Existence

- Weighted congestion game Γ played between WDs has a NE $\mathbf{d}^*$
  - proof based on showing that the game has an exact potential function

Improve Offloading (IO) Algorithm

- adds WDs one at a time
- lets WDs to play their best improvement steps given the other WDs' strategies

Main Results

NE Existence

- Weighted congestion game Γ played between WDs has a NE $\mathbf{d}^*$
  - proof based on showing that the game has an exact potential function

Improve Offloading (IO) Algorithm

- adds WDs one at a time
- lets WDs to play their best improvement steps given the other WDs' strategies

SPE Existence

- Stackelberg game played between WDs and the operator has a SPE $(\mathbf{d}^*, \mathbf{b}^*, \mathbf{p}^*)$
  - optimal provisioning coefficients $\mathbf{b}^*$ and $\mathbf{p}^*$ have closed form expressions
  - WDs can compute an equilibrium allocation $\mathbf{d}^*$ of offloading decisions in a decentralized manner using the IO algorithm

User Focused Performance Analysis

Evaluation Scenario

- $A = 5$ APs, homogeneous clouds with $F^c = 64$ *Gcycles*
- Tasks: $D_i \sim \mathcal{U}(0.2, 4)$ *Mb* , $L_i = D_i X$ *Gcycles* , $X \sim \Gamma(0.5, 1.6)$ *Gcycles/b*

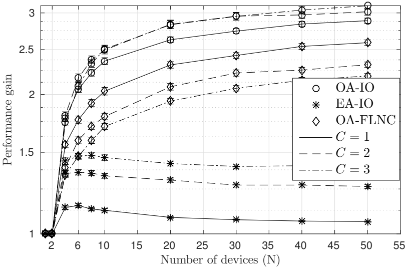
User Focused Performance Analysis

Evaluation Scenario

- $A = 5$ APs, homogeneous clouds with $F^c = 64$ *Gcycles*
- Tasks: $D_i \sim \mathcal{U}(0.2, 4)$ *Mb* , $L_i = D_i X$ *Gcycles* , $X \sim \Gamma(0.5, 1.6)$ *Gcycles/b*

Performance Gain

Defined w.r.t. *equal alocation* (EA) policy and the *fastest-link nearest-cloud* (FLNC) algorithm



- *Performance gain* increases with decreasing marginal gain in N

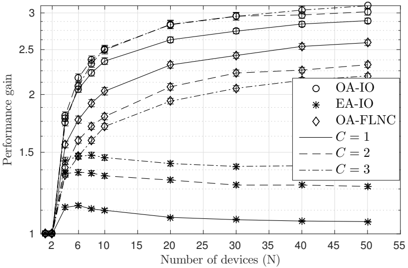
User Focused Performance Analysis

Evaluation Scenario

- $A = 5$ APs, homogeneous clouds with $F^c = 64$ *Gcycles*
- Tasks: $D_i \sim \mathcal{U}(0.2, 4)$ *Mb* , $L_i = D_i X$ *Gcycles* , $X \sim \Gamma(0.5, 1.6)$ *Gcycles/b*

Performance Gain

Defined w.r.t. *equal alocation* (EA) policy and the *fastest-link nearest-cloud* (FLNC) algorithm



- *Performance gain* increases with decreasing marginal gain in N
- *Performance gain* increases with the number of clouds

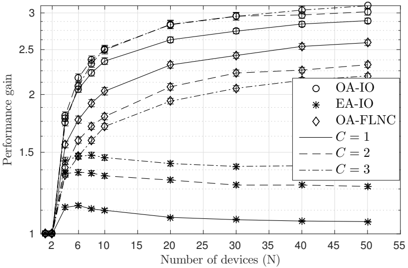
User Focused Performance Analysis

Evaluation Scenario

- $A = 5$ APs, homogeneous clouds with $F^c = 64$ *Gcycles*
- Tasks: $D_i \sim \mathcal{U}(0.2, 4)$ *Mb* , $L_i = D_i X$ *Gcycles* , $X \sim \Gamma(0.5, 1.6)$ *Gcycles/b*

Performance Gain

Defined w.r.t. *equal alocation* (EA) policy and the *fastest-link nearest-cloud* (FLNC) algorithm



- *Performance gain* increases with decreasing marginal gain in N
- *Performance gain* increases with the number of clouds
- Largest *performance gain*
  - operator implements OA policy
  - WDs compute offloading decisions using the IO algorithm

Cloud Focused Performance Analysis

Evaluation Scenario

- $A = 5$ APs, $C = 3$ heterogeneous clouds
- Tasks: $D_i \sim \mathcal{U}(0.2, 4)$ *Mb* , $L_i = D_i X$ *Gcycles* , $X \sim \Gamma(0.5, 1.6)$ *Gcycles/b*

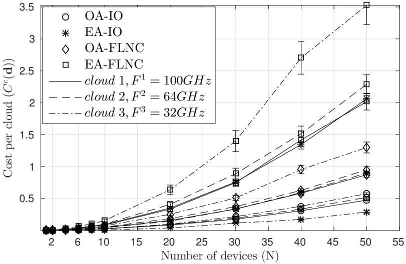
Cloud Focused Performance Analysis

Evaluation Scenario

- $A = 5$ APs, $C = 3$ heterogeneous clouds
- Tasks: $D_i \sim \mathcal{U}(0.2, 4)$ *Mb* , $L_i = D_i X$ *Gcycles* , $X \sim \Gamma(0.5, 1.6)$ *Gcycles/b*

Cost per Cloud

Defined as $C^c(\mathbf{d}) = \sum_{i \in O_c(\mathbf{d})} C_i(\mathbf{d})$



- *Cost per cloud* increases with the number N of WDs

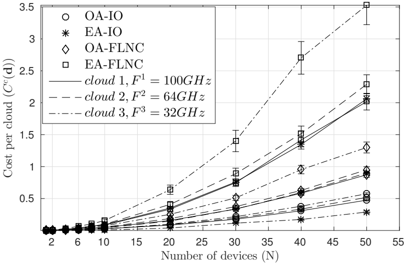
Cloud Focused Performance Analysis

Evaluation Scenario

- $A = 5$ APs, $C = 3$ heterogeneous clouds
- Tasks: $D_i \sim \mathcal{U}(0.2, 4)$ *Mb* , $L_i = D_i X$ *Gcycles* , $X \sim \Gamma(0.5, 1.6)$ *Gcycles/b*

Cost per Cloud

Defined as $C^c(\mathbf{d}) = \sum_{i \in O_c(\mathbf{d})} C_i(\mathbf{d})$



- *Cost per cloud* increases with the number N of WDs
- *Cost per cloud* is approximately the same for all clouds in the case of the OA policy and the IO algorithm

Computational Complexity

Evaluation Scenario

- $A = 5$ APs, homogeneous clouds with $F^c = 64$ *Gcycles*
- Tasks: $D_i \sim \mathcal{U}(0.2, 4)$ *Mb* , $L_i = D_i X$ *Gcycles* , $X \sim \Gamma(0.5, 1.6)$ *Gcycles/b*

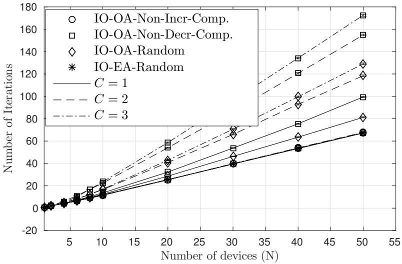
Computational Complexity

Evaluation Scenario

- $A = 5$ APs, homogeneous clouds with $F^c = 64$ *Gcycles*
- Tasks: $D_i \sim \mathcal{U}(0.2, 4)$ *Mb* , $L_i = D_i X$ *Gcycles* , $X \sim \Gamma(0.5, 1.6)$ *Gcycles/b*

IO Algorithm - Three Orders of Adding WDs

- Non-increasing order of tasks' complexities
- Non-decreasing order of tasks' complexities
- Random order



- Number of iterations scales approximately linearly with the number N of WDs

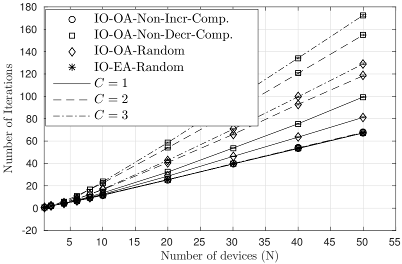
Computational Complexity

Evaluation Scenario

- $A = 5$ APs, homogeneous clouds with $F^c = 64$ *Gcycles*
- Tasks: $D_i \sim \mathcal{U}(0.2, 4)$ *Mb* , $L_i = D_i X$ *Gcycles* , $X \sim \Gamma(0.5, 1.6)$ *Gcycles/b*

IO Algorithm - Three Orders of Adding WDs

- Non-increasing order of tasks' complexities
- Non-decreasing order of tasks' complexities
- Random order



- Number of iterations scales approximately linearly with the number N of WDs
- Number of iterations is sensitive to the order of adding WDs
  - it is smallest when WDs are added in non-increasing order of their task complexities

Summary and Future Work

- Provided game theoretical analysis of the interaction between
  - operator that manages wireless and computing resources in a MEC system and
  - autonomous WDs that aim at minimizing their own task completion time

Summary and Future Work

- Provided game theoretical analysis of the interaction between
  - operator that manages wireless and computing resources in a MEC system and
  - autonomous WDs that aim at minimizing their own task completion time

- Interesting extensions
  - model in which WDs can perform computation locally
  - energy consumption minimization problem
  - stochastic model of task arrivals

Joint Allocation of Computing and Wireless Resources to Autonomous Devices in Mobile Edge Computing

Slađana Jošilo, György Dán

Department of Network and Systems Engineering
School of Electrical Engineering and Computer Science
KTH Royal Institute of Technology

Budapest, August 20, 2018