# GoPro: A Low Complexity Task Allocation Algorithm for a Mobile Edge Computing System

Arghyadip Roy

Mehta Family School of Data Science & Artificial Intelligence
IIT Guwahati

Joint work with Dr. Nilanjan Biswas, BIT Mesra

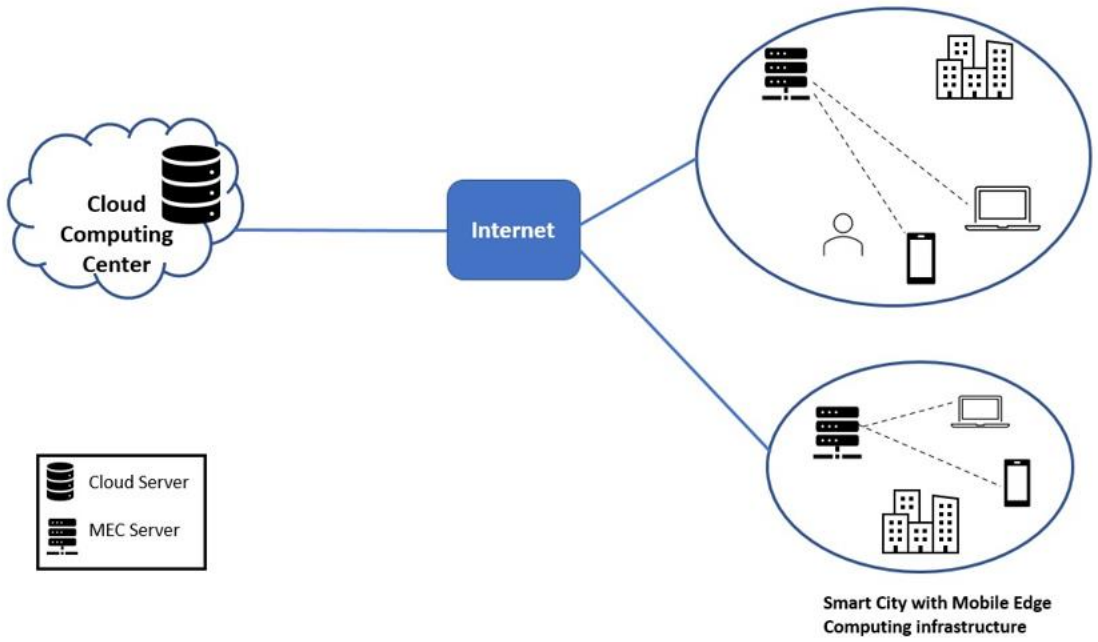
# Outline

- Introduction & Motivation
- System Model & Problem Formulation
- Proposed Heuristic
- Complexity Analysis
- Numerical Results
- Conclusions

# Introduction

- Enormous number of wireless devices with data-hungry applications
- Computationally-intensive applications require a lot of processing power
  - Example: Internet of Things (IoT) applications, virtual reality, augmented reality
- Use cases in Fifth Generation (5G)/Sixth Generation (6G) networks
  - Industry 4.0, Smart City, Smart Agriculture
- Operation of IoT devices in cellular bands : Narrow-Band (NB) IoT in Third Generation Partnership Project (3GPP) Release 13
- Incorporation of Mobile Edge Computing (MEC) in 5G : 3GPP Release 17

# Introduction: A smart-city Scenario

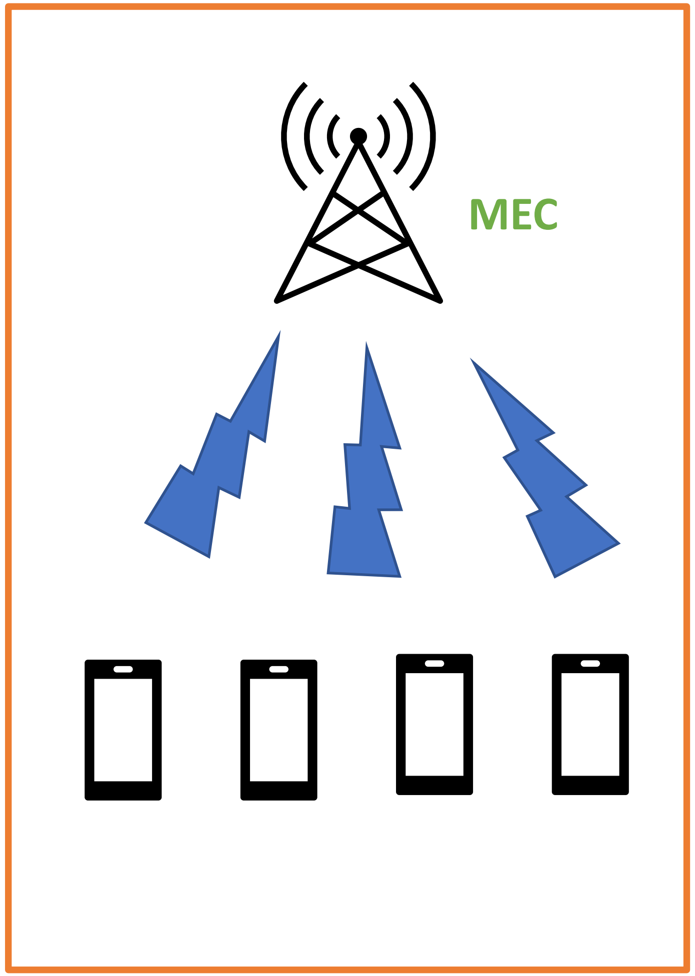



**MEC server**

- An alternate to cloud computing server
- Located closer to user
- Performance similar to cloud computing server

Where to compute/ execute tasks?
Local vs MEC

# Motivation




MEC Computation: High Power, Less deadline violation

Local Computation: Less Power, High deadline violation

Random arrival of tasks: Soft deadline

Channel condition of nodes w.r.t MEC server: Random

Average power consumption minimization subject to deadline violation probability constraint

# Related Work

## Single Wireless Device

- Drift-plus-penalty method: Suboptimal $O(\frac{1}{V}, V)$trade-off
- Consideration of infinite CPU computation ability

## Multiple Devices

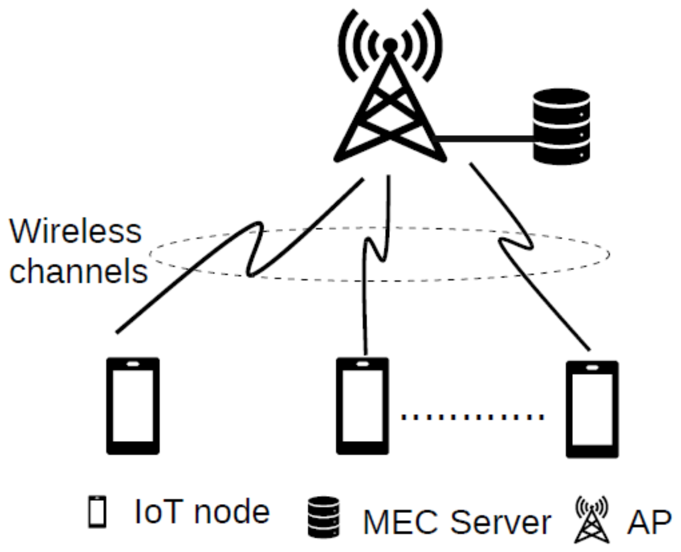
- Power-delay trade-off for full offloading

# Contributions

- Minimization of average power consumption subject to deadline violation probability constraint in an IoT network: CMDP Formulation
- Optimal policy calculation may be computationally infeasible
- Low-complexity Heuristic: Can be implemented online

- Dynamic arrival and departure of tasks at multiple IoT nodes
- Deadline violation probability instead of delay: Unnecessary offloading to MEC server
- Finite CPU computational resources at MEC server

# System Model




**Assumptions**

- Task arrival at IoT nodes: Poisson ($\lambda$)
- Service time of tasks: Exponential ($\mu$)
- Licensed Frequency band (NB-IoT): Centralized distribution of resources by MEC server

# State and Action Space

State Space $S$

$s = (i, j_G, j_B)$

$i$: Number of local tasks

$j_G$: Number of MEC tasks with good channel

$j_B$: Number of MEC tasks with bad channel

Constraint:

$i \leq N_1,$

$(j_G + j_B) \leq N_2$

Action Space $A$

$A = \{A_1, A_2, A_0\}$

$A_1$: Local Computation

$A_2$: MEC offloading

$A_0$: Stay idle

# Power Consumption Cost

- Moves to different states with different probabilities depending on the event and action
- Power Consumption Cost
  - Due to computation $P_{c,y} = w_y f_y^3, y \in \{I, M\}$
  - Due to offloading $P_{o,b} = P_{o,g} d$

$w_y$: Co-efficient (chip architecture)
$f_y$: CPU frequency
$d$: Channel degradation factor

- Power Consumption cost function example

$$c_p(s, A_1, E_1) = P_{c,I}(i+1) + P_{c,M}(j_G + j_B) + P_{o,g}(j_G + d j_B)$$

# Deadline Violation Cost

Equal allocation of CPU resources

- Local computation time: $T_{c,I} = \frac{BC}{f_I}$
- Remote computation time: $T_{c,M}(s) = \frac{B}{R_m} + \frac{BC(j_G+j_B)}{f_M}$

Task Offloading Time

$B$: Number of bits for computation
$C$: CPU cycles to compute a bit
$d$: Channel degradation factor
$R_m$: Channel capacity
$f_y$: CPU frequency

Deadline violation cost function example

$$c_d(s, A_2, E_1) = 1_{\{T_{c,M}(i, j_G+1, j_B) > L\}}$$

# Problem Formulation

- Policy: Sequence of rules regarding actions at different states and epochs

**Optimal task scheduling policy:**
Power consumption minimization subject to a constraint on the deadline violation probability

**Continuous time CMDP Problem:**
Stationary randomized policy
Arrival and departure can happen at any time

**Minimize** $C^P = \lim_{t\to\infty} \frac{1}{t} E_P[C_{1(t)}]$

**Subject to** $D^P = \lim_{t\to\infty} \frac{1}{t} E_P[C_{2(t)}] \leq C_{max}$

Solution Methodology: Value Iteration Algorithm+ Lagrangian Approach[1]

[1]E. Altman, Constrained Markov decision processes. CRC Press, 1999, vol. 7.

# Curse of Dimensionality

- Computational complexity is exponential in state space
- Storage complexity high if number of states are more
- Requires offline computation
- Low complexity heuristic?
- Feasible for online implementation?

| Algorithm | Storage Complexity | Computational Complexity |
|---|---|---|
| Optimal Policy | $O(\|S\|) = O(N_1 N_2^2)$ | $O(\|A\|^{\|S\|})$ |

# Heuristic Sketch

Good Task: Always offload to MEC server

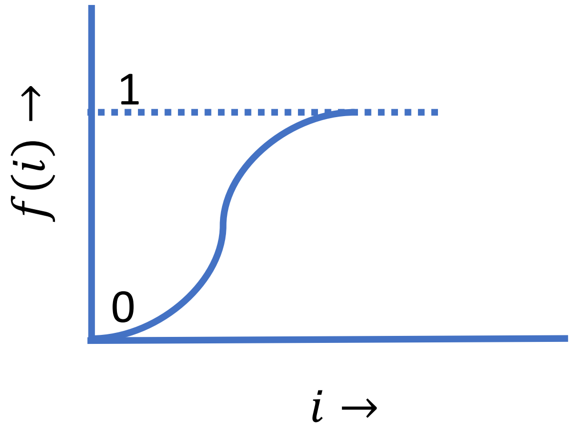
**GoPro: Good? Offload, else probabilistic offload**

Good channel task consumes less power for offloading

Bad Task: Probabilistically offload to MEC server

Excessive MEC offloading: High power consumption

Bad Task: Probability is increased as $i$ increases

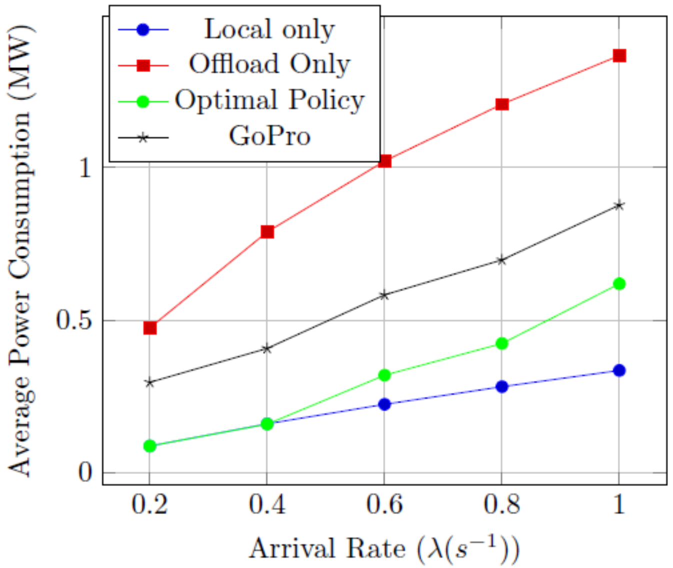


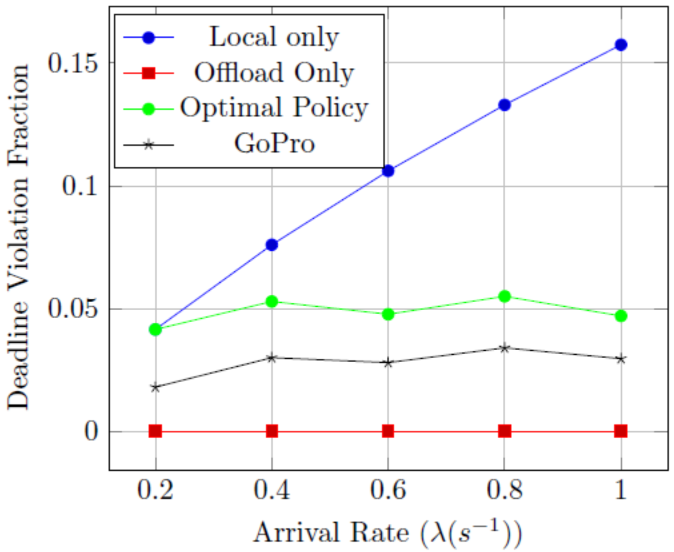
# Addressing the curse of dimensionality

- Computational complexity: Flip a coin, decide offload/local
- Storage complexity: Need to store $i$ and $f(i)$ as $i \leq N_1$
- Does not require knowledge of system parameters: Online implementable

| Algorithm | Storage Complexity | Computational Complexity |
|---|---|---|
| Optimal Policy | $O(\|S\|) = O(N_1 N_2^2)$ | $O(\|A\|^{\|S\|})$ |
| GoPro | $O(N_1)$ | $O(1)$ |

# Numerical Results




- Better than local only and offload only policies
- Similar to optimal policy
- More conservative than optimal policy in deadline violation

# Conclusions

- Trade-off between power consumption and deadline violation probability in an MEC based IoT network
- Optimal Policy has high storage and computational complexities, requires offline computations
- GoPro:
  - Low-complexity heuristic
  - Can be implemented online
  - Balances between power consumption and deadline violation

Arghyadip Roy and Nilanjan Biswas," GoPro: A Low Complexity Task Allocation Algorithm for a Mobile Edge Computing System," IEEE National Conference on Communications (NCC) (Invited Paper), 2022.

# Other Research Activities

- Risk-sensitive/ Robust Reinforcement Learning
- Multi-armed Bandit Algorithms for Markovian Bandits
- mmwave beamforming using machine learning
- SDN load balancing using machine learning

Moharrami, M., Murthy, Y., Roy, A., & Srikant, R. (2022). A Policy Gradient Algorithm for the Risk-Sensitive Exponential Cost MDP. *arXiv preprint arXiv:2202.04157*.

Roy, A., Shakkottai, S., & Srikant, R. (2020). Adaptive KL-UCB based Bandit Algorithms for Markovian and iid Settings. *arXiv preprint arXiv:2009.06606*.

# References


- A. Hoglund, X. Lin, O. Liberg, A. Behravan, E. A. Yavuz, M. Van Der Zee, Y. Sui, T. Tirronen, A. Ratilainen, and D. Eriksson, “Overview of 3GPP release 14 enhanced NB-IoT,” IEEE network, vol. 31, no. 6, pp. 16–22, 2017.
- Y. C. Hu, M. Patel, D. Sabella, N. Sprecher, and V. Young, “Mobile edge computing—a key technology towards 5G,” ETSI white paper, vol. 11, no. 11, pp. 1–16, 2015.
- G. Zhang, W. Zhang, Y. Cao, D. Li, and L. Wang, “Energy-delay tradeoff for dynamic offloading in mobile-edge computing system with energy harvesting devices,” IEEE Transactions on Industrial Informatics, vol. 14, no. 10, pp. 4642–4655, 2018.
- N. Biswas, H. Mirghasemi, and L. Vandendorpe, “Joint optimization of relaying rate and energy consumption for cooperative mobile edge computing,” in IEEE WiOPT, 2020, pp. 1–8.
- D. Han, W. Chen, and Y. Fang, “Joint channel and queue aware scheduling for latency sensitive mobile edge computing with power constraints,” IEEE Transactions on Wireless Communications, vol. 19, no. 6, pp. 3938–3951, 2020.
- M. Merluzzi, P. Di Lorenzo, S. Barbarossa, and V. Frascolla, “Dynamic computation offloading in multi-access edge computing via ultra-reliable and low-latency communications,” IEEE Transactions on Signal and Information Processing over Networks, vol. 6, pp. 342–356, 2020.

Thank you