Task offloading in Edge and Cloud Computing

A survey on mathematical, artificial intelligence and control theory solutions

Firdose Saeik, Marios Avgeris, Dimitrios Spatharakis, Nina Santi, Dimitrios Dechouniotis, John Violos, Aris Leivadeas, Nikolaos Athanasopoulos , Nathalie Mitton , Symeon Papavassiliou.

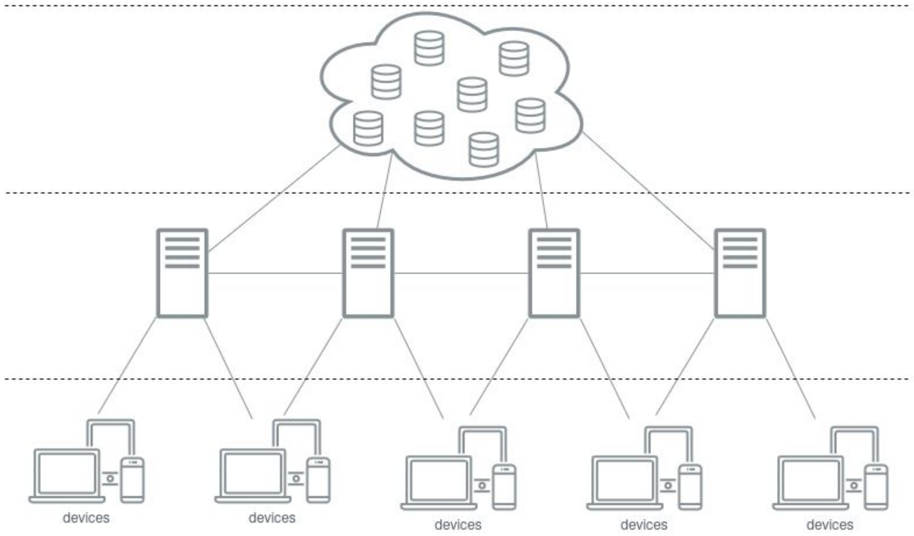
Marco Savarese - Mat. 195506

Why Task Offloading?

- Explosion of resource-intensive application (VR, autonomous vehicles, IoT, etc.).
- Devices have limited compute power.
- Task offloading = sending tasks to powerful remote systems.

Computing Paradigms

- Cloud Computing.
- Fog Computing.
- Edge Computing.

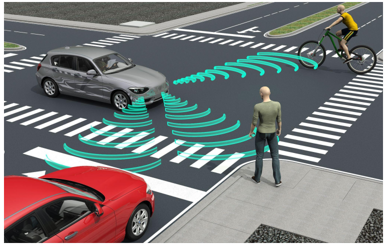
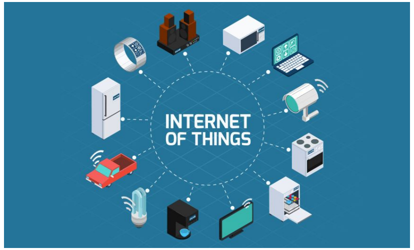


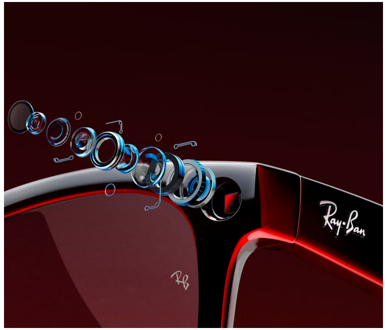
Paradigms Compared

Paradigm	Latency	Resources	Proximity to users
Cloud	High	∞	Far
Fog	Medium	Medium	Intermediate
Edge	Low	Limited	Close

Where Task Offloading Makes a Difference

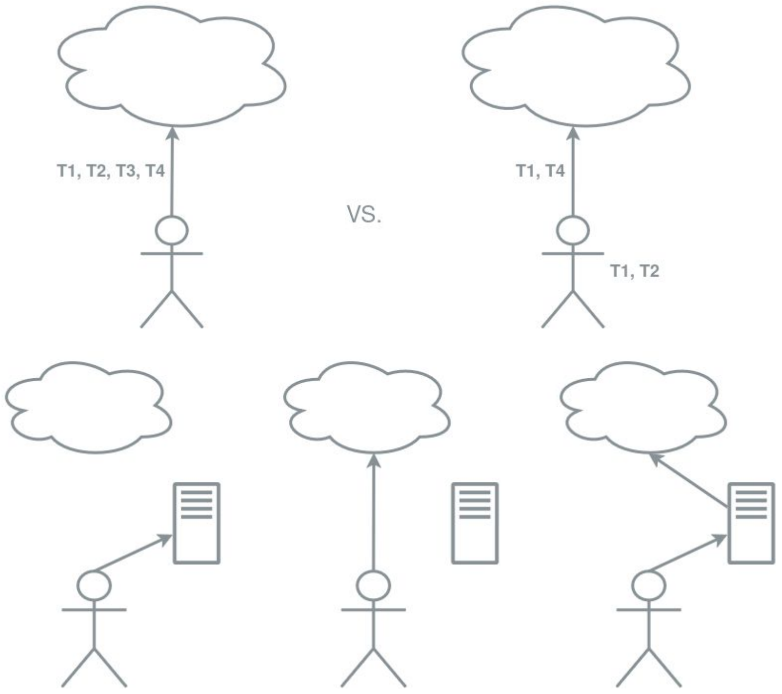
- **Autonomous Vehicles**: real-time decisions.
- **VR/AR**: low latency required.
- **IoT**: battery-constrained devices.
- **Video streaming**: fast local caching.
- **Disaster Response**: UAVs with limited computational power.



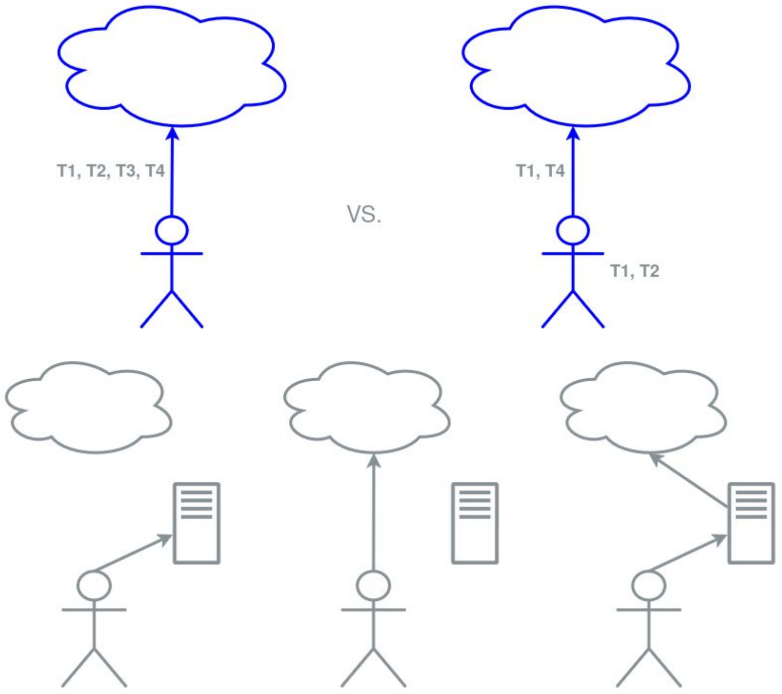
Offloading Strategies

- Granularity of offloaded tasks.
- Full vs. Partial Offloading.
- Offloading to Edge, Cloud or both.



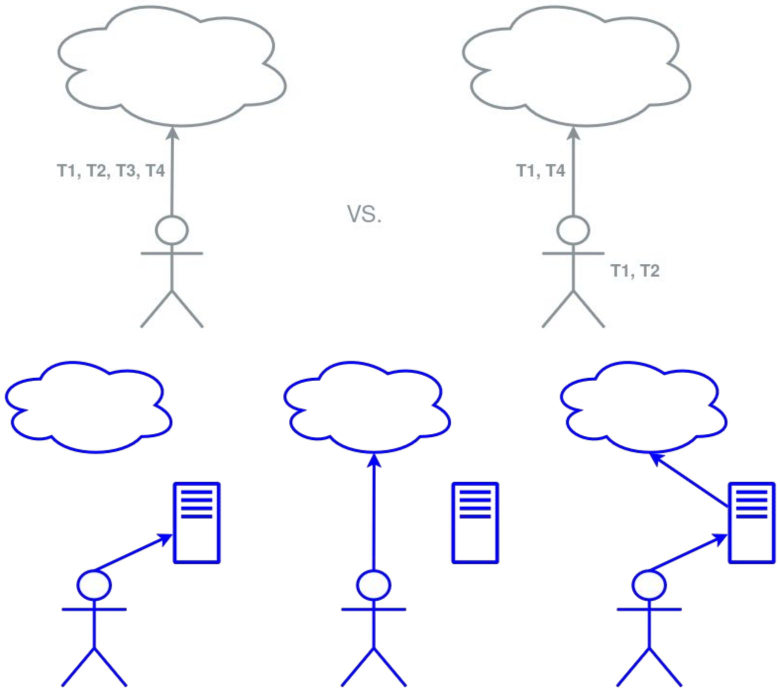
Offloading Strategies

- Granularity of offloaded tasks.
- **Full vs. Partial Offloading.**
- Offloading to Edge, Cloud or both.



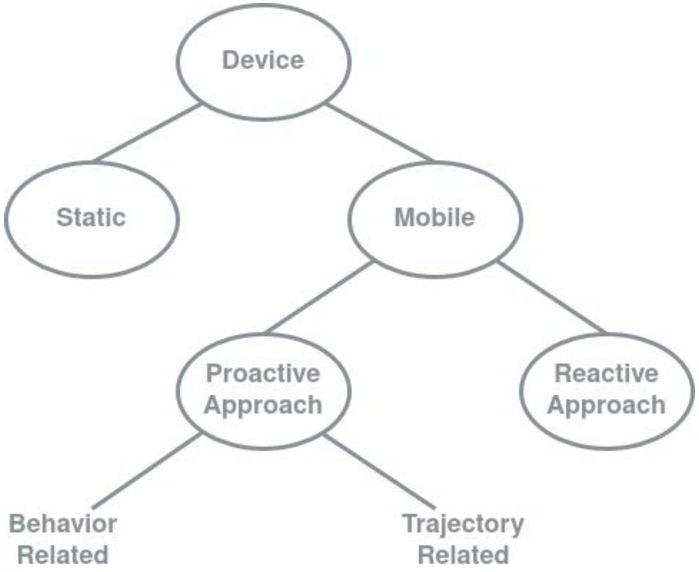
Offloading Strategies

- Granularity of offloaded tasks.
- Full vs. Partial Offloading.
- **Offloading to Edge, Cloud or both.**



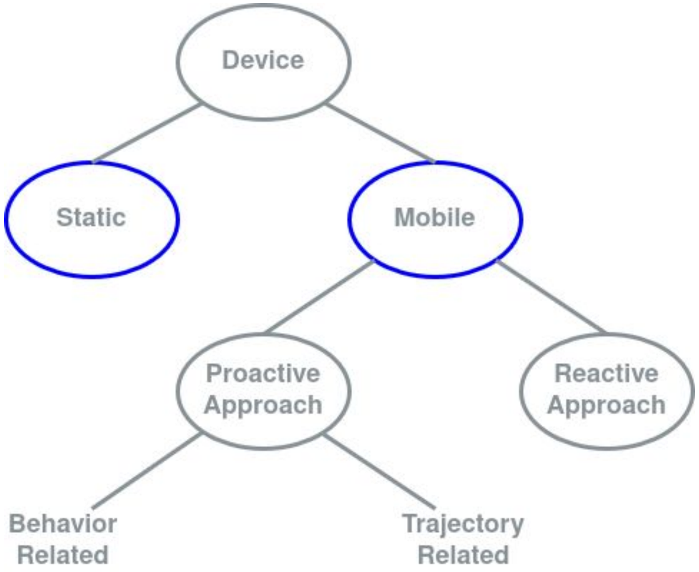
Mobility of End Devices

- Service migration and load balancing.
- Static vs. Mobile Devices.
- Proactive vs. Reactive approaches.



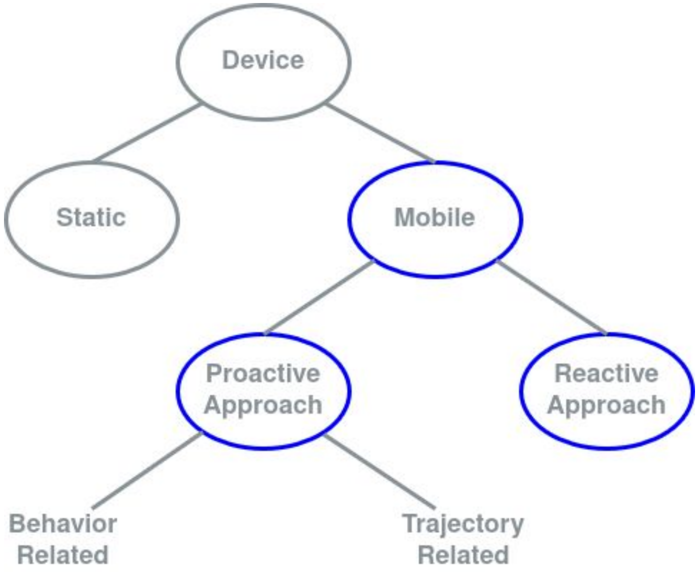
Mobility of End Devices

- Service migration and load balancing.
- **Static vs. Mobile Devices.**
- Proactive vs. Reactive approaches.



Mobility of End Devices

- Service migration and load balancing.
- Static vs. Mobile Devices.
- **Proactive vs. Reactive approaches.**



Task Offloading Objectives: What Has to Be Optimized?

- Delay.
- Energy.
- Bandwidth.
- Load Balancing.
- Deployment Cost.
- Model Accuracy (for AI-based approaches).

Task Offloading Challenges

- Network dynamics:
  - Dynamic Network Conditions.
  - Dynamic User Behavior.
  - Edge/Cloud Dynamics.

- Resource allocation:
  - Partitioning Decision.
  - Resource Availability.
  - Performance Modeling.
  - Task Management.

Task Offloading Approaches: Overview

- Mathematical Optimization (MO).
- Artificial Intelligence (AI).
- Control Theory (CT).

Task Offloading Approaches: Mathematical Optimization

- Solves offloading as constrained optimization problem.
- Examples: MIP, heuristics, game theory, contract theory, local search.
- **Pros**: precise, formal guarantees.
- **Cons**: slow, static.

$$
\begin{array}{ll}
\max\limits_{w,\mu} & -\frac{1}{2}w^T w + \mu^T c \\
\text{sub. to} & w - \mu^T A = 0 \\
 & \mu^T y = 0 \\
 & \mu \geq 0
\end{array}
$$

Mathematical Optimization: Game Theory

- Models task offloading as a game between users or infrastructures.
- Multi-slot game: the system tries to evolve to a stable state (Nash Equilibrium).
- Edge-Cloud interplay (Stackelberg): maximize provider revenue while satisfying user delay constraints.

Task Offloading Approaches: Artificial Intelligence

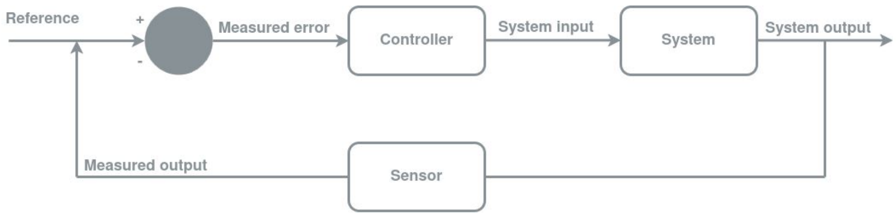
- Learns from data to make decisions
- Examples: ML, Population-based methods, Constraint satisfaction methods.
- **Pros**: flexible, supports online learning.
- **Cons**: needs training data, risk of overfitting.

Artificial Intelligence: Machine Learning

Approach	Techniques	Use Cases
Supervised Learning	Decision Trees, SVM, Regression	Task classification, load prediction
Unsupervised Learning	K-Means, Fuzzy Clustering, Policy-based Clustering	Resource grouping, node/task clustering
Deep Learning	Deep Neural Networks, CNNs	Task scheduling, VM migration, energy/QoS
Deep Reinforcement Learning	Deep Q-Networks, LSTM, GRU	Adaptive offloading decisions, sequential

Task Offloading Approaches: Control Theory

- System modeling and real-time control.
- Good for dynamic networks.
- Guarantees stability.
- **Pros**: fast response, real-time control.
- **Cons**: requires accurate system modeling.



Control Theory: State feedback control

- Adjusts system based on real-time measurements.
- Enables dynamic resource scaling.
- Used to maintain performance during workload changes.
- Applied in UAV wildfire detection (SMOKE system).

Approaches Compared

Feature	MO	AI	CT
Stability			✓
Low Complexity	✓		
Optimality	✓		✓
Online Training		✓	
Real-time Decision	✓	✓	✓

Open challenges

- Heterogeneity (networks, nodes, resources, applications)
- Mobility and handover prediction.
- How to move edge resources closer to the end devices.
- Resource scarcity at Edge.
- Security and privacy.
- Fault tolerance.

Key Takeaways

- Task offloading enables advanced applications.
- Hybrid Edge-Cloud collaboration is key.
- Many optimization approaches available.
- Tradeoffs matter.

Thank you for your attention!

Marco Savarese - Mat. 195506

271055@studenti.unimore.it - inbox@marcosavarese.dev