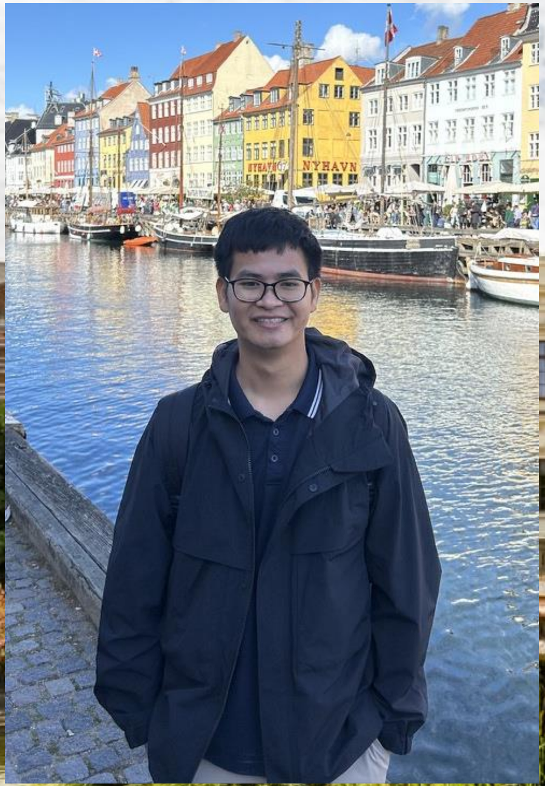
A Metaheuristic Approach for Mission Assignment and Task Offloading in Open RAN-Enabled Intelligent Transport Systems

IEEE GLOBECOM 2025, 8-12 December 2025, Taipei, Taiwan

SUSTAINABLE COMMUNICATIONS FOR UBIQUITOUS INTELLIGENCE



Author: Ngoc Hung, Nguyen
Department of Electrical and Information Technology, LTH, Lund University | ngoc_hung.nguyen@eit.lth.se

Co-author:
Mr. Nguyen Van Thieu, Phenikaa University, Vietnam
Dr. Quang-Trung Luu, Paris-Saclay University, France
Mr. Vo-Phi Son, VinUniversity, Vietnam
Dr. Van-Dinh Nguyen, Trinity College Dublin, Ireland

Outline

1. Introduction, Problem statement, System model
2. Optimization problem formulation
3. Proposed meta-heuristic algorithm
4. Experimental results
5. Conclusion and future work

Outline

1. ***Introduction, Problem statement, System model***
2. Optimization problem formulation
3. Proposed meta-heuristic algorithm
4. Experimental results
5. Conclusion and future work

Introduction, Problem statement, System model

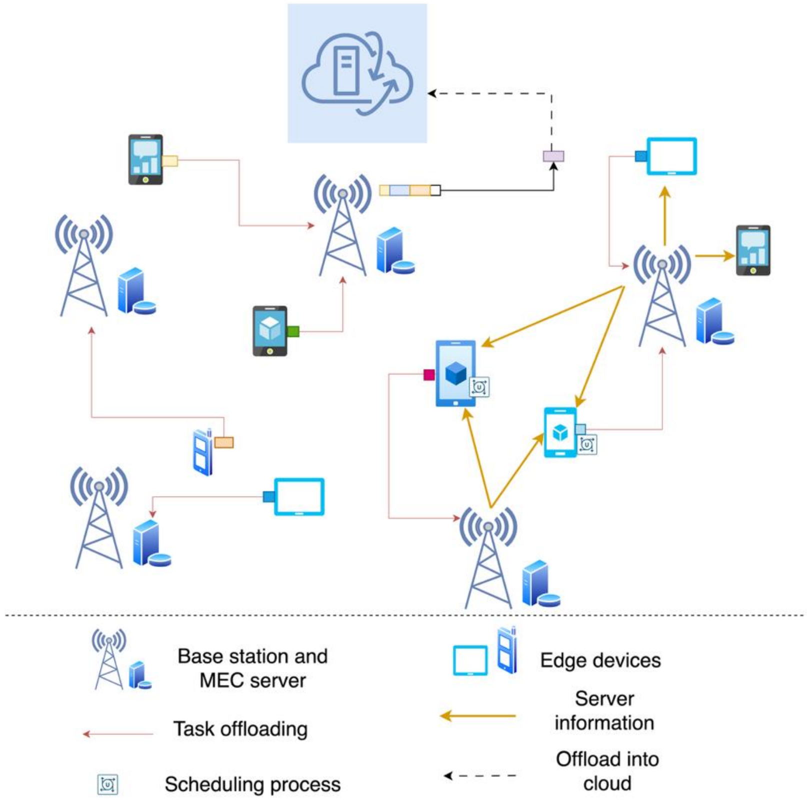
Mobile edge computing systems

<u>*Advantages:*</u>

- Low latency & reduced backhaul traffic
- Efficient task offloading for autonomous/ITS services
- Proximity to users improves reliability & QoS

<u>*Drawbacks:*</u>

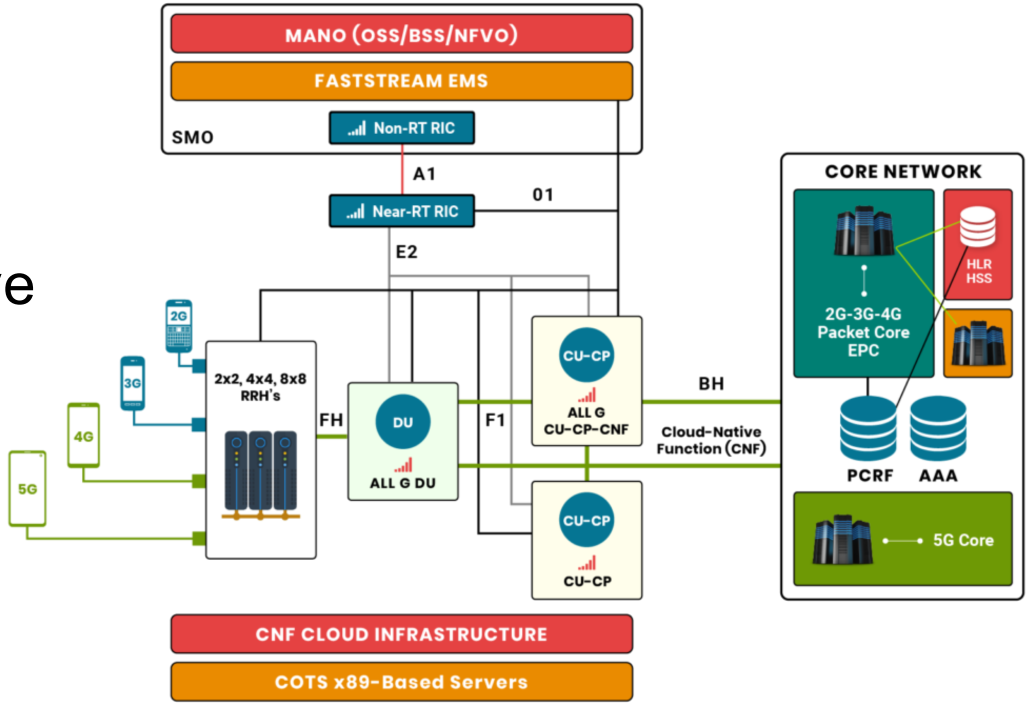
- *High deployment & maintenance cost*
- *Limited compute/storage resources vs Cloud*
- *Complexity in mobility, orchestration & security*



Picture source: *https://arxiv.org/pdf/2507.18864*

Introduction, Problem statement, System model

- Open, interoperable multi-vendor ecosystem
- Flexible, software-defined, and cloud-native architecture
- Supports intelligent control via RIC (rApps/xApps)



Picture source: https://www.faststreamtech.com/products/5g-open-ran/

LUND UNIVERSITY

Introduction, Problem statement, System model

Trade-off between latency and offloading cost

Dependence between missions in system

Offloading based on traffic conditions

RQ:
What is the best way to assign missions to smart vehicles?
How do the MEC, ORAN support delivery systems?
How to design a good search-based solution for the future work?

Picture source: *https://www.mdpi.com/2075-1702/10/11/1059c*

LUND UNIVERSITY

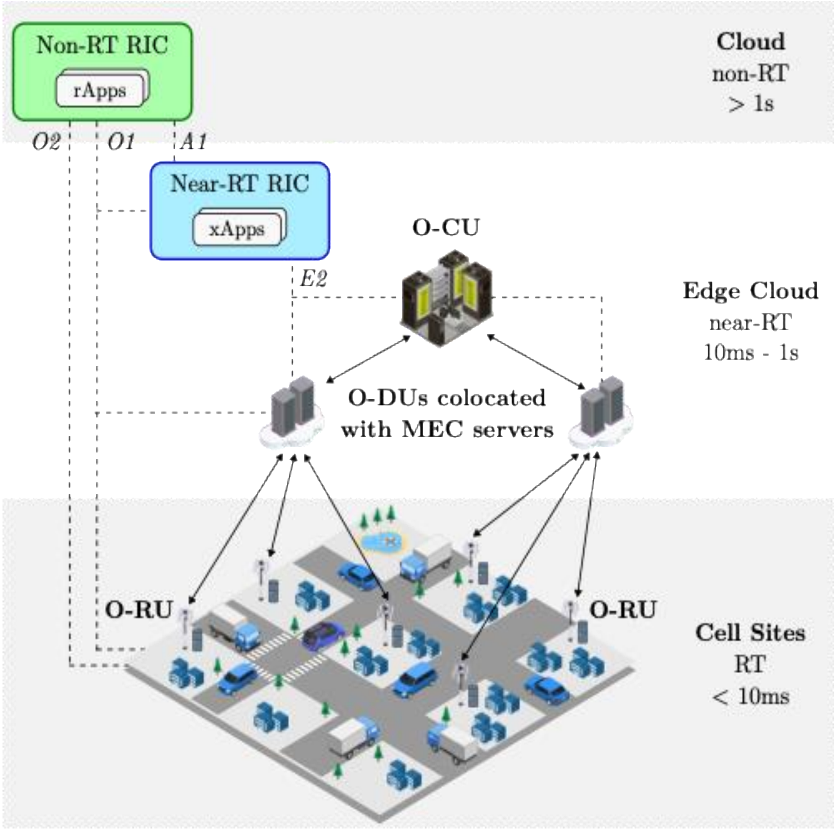
Introduction, Problem statement, System model

Optimize mission assignment & task offloading for autonomous vehicles in Open RAN–ITS

Goal: *Maximize missions completed before deadlines*

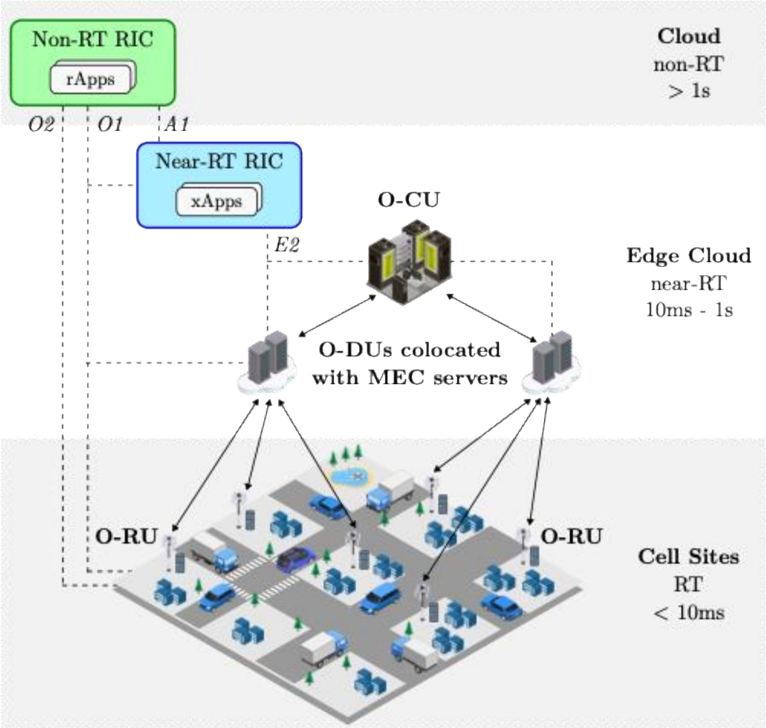
Considering:

- *Mission dependencies*
- *Offloading cost/budget*
- *Communication, computation & travel delays*



Problem statement, System model

- There are MEC servers (edge) and a cloud server
- O-RU, O-DU, O-CU components map to real Open RAN layers
- RIC: Near-RT RIC (xApps) and Non-RT RIC (rApps) influence scheduling and control
- Vehicles communicate with RU → DU (MEC) → CU (Cloud)



Problem statement, System model

Def. ***Mission***

$$M_i(\tau) \triangleq \langle r_i(\tau), T_i(\tau), B_i(\tau), \mathcal{M}_i^-, \mathcal{M}_i^+ \rangle$$

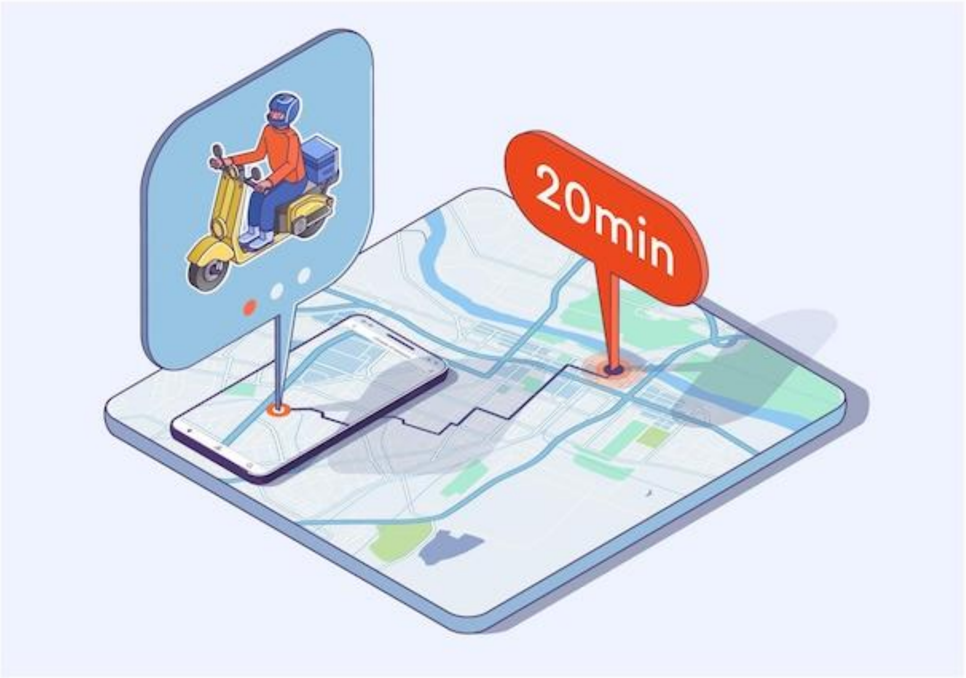
Road (start, end)

Deadline

Allocated offloading budget

Predecessor missions

Predecessor missions



Picture source: *https://www.freepik.com/free-vector/food-delivery-isometric-vector-illustration-bike-courier-city-street-map-online-service_25515984.htm*

Problem statement, System model

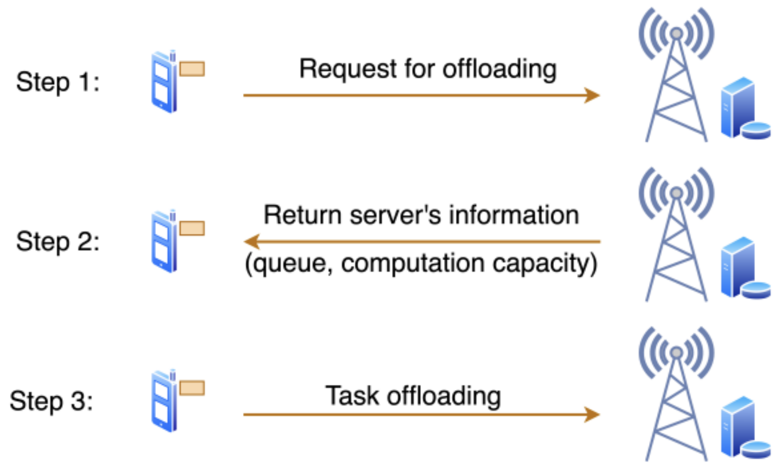
<u>Def</u>. ***Offloading Tasks of a mission i***

$$j \in \mathcal{J}_i = \langle \alpha_{i,j}(\tau), \beta_{i,j}(\tau) \rangle$$

All offloading tasks to completing a mission

Transmission data size

Computing requirements



Picture source: *https://arxiv.org/pdf/2507.18864*

Problem statement, System model

Def. ***Time spend to complete a mission***

$$d_i(\tau) = d_i^{\text{move}}(\tau) + d_i^{\text{comm}}(\tau) + d^{\text{comp}}$$

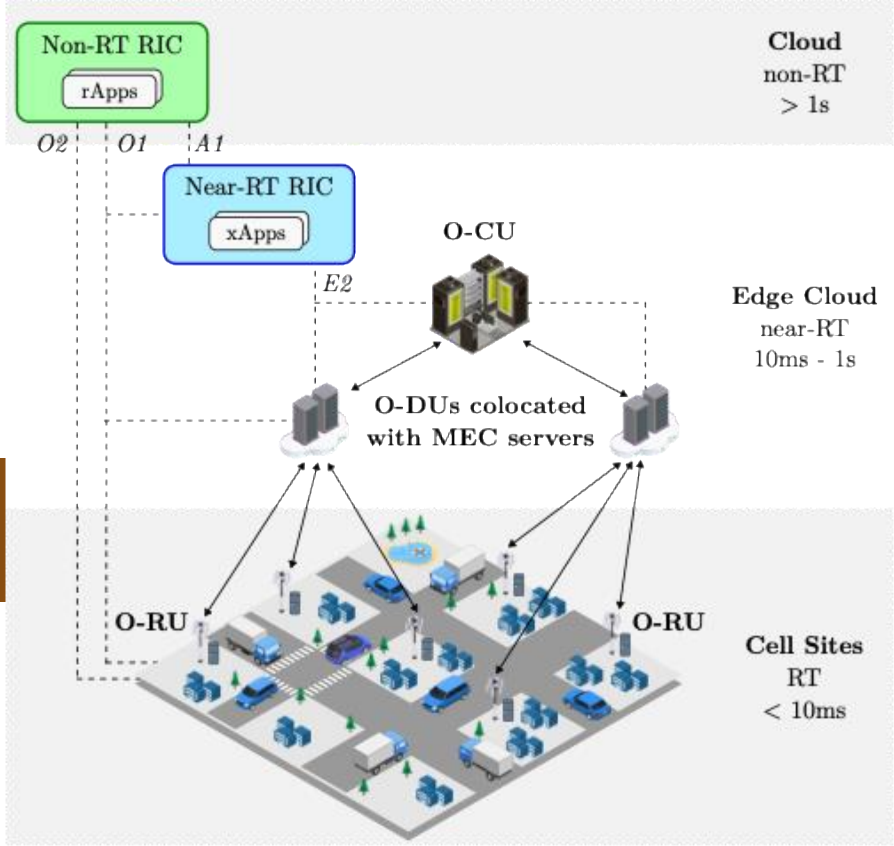
Def. ***Offloading cost for completing mission***

$$C_i(\tau) = \sum_{j=1}^{|\mathcal{J}_i|} C_{i,j}(\tau) = \sum_{j=1}^{|\mathcal{J}_i|} c_o(d_{i,j}^{\text{comm}} + d_{i,j}^{\text{comp}})$$

Per-unit cost charge

Def. ***Remaining benefits***

$$B_i^{\text{rema}}(\tau) = B_i(\tau) - C_i(\tau) \geq 0, \quad \forall M_i(\tau) \in \mathbf{M}(\tau)$$



LUND UNIVERSITY

Outline

1. Introduction, Problem statement, System model
2. ***Optimization problem formulation***
3. Proposed meta-heuristic algorithm
4. Experimental results
5. Conclusion and future work

Optimization problem formulation

$$\mathscr{P}_1 : \max_{\mathbf{D}(\tau)} \sum_{M_i(\tau) \in \mathbf{M}(\tau)} \mathbb{1}_{\{\delta_i(\tau) \leq T_i(\tau)\}}$$

$$\text{s.t.} \quad (1), (3), (4), (5), (6), (10), (15).$$

(1) All mission is assigned to K vehicles.*

(3) One mission is handled by a vehicle

(4) Each mission has a specified order

(5) All assigned missions has order in one vehicles

(6) Ordering mission has to based on its successors and predecessors

(10) All missions is completed validly if it is in considered time frame

(15) Remaining benefits should not negative

LUND UNIVERSITY

Outline

1. Introduction, Problem statement, System model
2. Optimization problem formulation
3. ***Proposed meta-heuristic algorithm***
4. Experimental results
5. Conclusion and future work

Proposed meta-heuristic algorithm

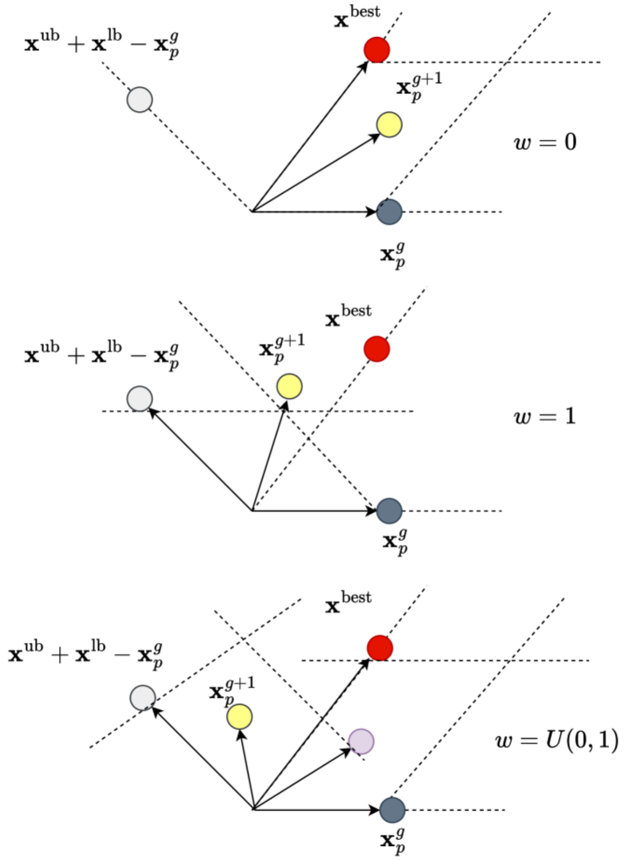
<u>*Limitation of original ARO:*</u>

- *Low initialization diversity → risk of premature convergence*
- *Detour foraging mixes exploration/exploitation inefficiently*
- *Random hiding lacks guidance from good solutions*
- *Less effective in dynamic, constraint-rich ITS optimization*

Picture source: *https://www.mathworks.com/matlabcentral/fileexchange/110250-artificial-rabbits-optimization-aro*

Proposed meta-heuristic algorithm

Improvement Category	**Enhancement in CGG-ARO**	**Main Benefits**
Population Initialization	Uses **Piecewise Chaotic Map (PCM)** instead of uniform random initialization	Increases population diversity, avoids premature convergence
Exploration Mechanism	Introduces **Gaussian-based detour foraging** to generate controlled random perturbations	Enhances global search capability and coverage of the search space
Exploitation Mechanism	Integrates **Opposition-Based Learning (OBL)** when updating solutions	Accelerates convergence toward high-quality regions and improves refinement
Random Hiding Stage	Updated using **global best solution + random agent guidance** instead of pure randomness	Stronger local exploitation, reduces aimless search behavior
Search Balance Strategy	Clearer separation between exploration and exploitation phases based on energy factor (A)	More stable optimization, reduced risk of local optima trapping
Position Updating Dynamics	Combination of **chaotic behavior + Gaussian noise + OBL influence**	Achieves better exploration–exploitation trade-off in dynamic ITS environments



LUND UNIVERSITY

Proposed meta-heuristic algorithm

1. Improving initial population set

$$x_p^{g+1}(l) = \begin{cases} \frac{x_p^g(l)}{\rho}, & 0 \le x_p^g(l) < \rho \\ \frac{x_p^g(l)-\rho}{0.5-\rho}, & \rho \le x_p^g(l) < 0.5 \\ \frac{1-\rho-x_p^g(l)}{0.5-\rho}, & 0.5 \le x_p^g(l) < 1-\rho \\ \frac{1-x_p^g(l)}{\rho}, & 1-\rho \le x_p^g(l) < 1. \end{cases} \quad (17)$$

2. Balance between exploration/exploitation stage moving direction

$$\mathbf{x}_p^{g+1} = \mathbf{x}_p^g + \mathbf{r}_1 \mathcal{N}(0, \boldsymbol{\sigma}) \quad (18)$$

$$\mathbf{x}_p^{g+1} = \mathbf{x}_p^g + \mathbf{r}_2 \left[w(\mathbf{x}^{\mathrm{ub}} + \mathbf{x}^{\mathrm{lb}} - \mathbf{x}_p^g) + (1-w)(\mathbf{x}^{\mathrm{best}} - \mathbf{x}_p^g) \right] \quad (19)$$

3. Random hiding stage during exploitation

$$\mathbf{x}_p^{g+1} = \mathbf{x}_{p'}^g + (2U(0,1)-1)\mathbf{r}(\mathbf{x}^{\mathrm{best}} - \mathbf{x}_{p''}^g) \quad (20)$$

Algorithm 1 Chaotic Gaussian-based Global AROs (CGG-ARO)

Initialization: Initialize population set: $[\mathbf{x}_1^g, \mathbf{x}_2^g, \cdots, \mathbf{x}_P^p]$ follow Section III-A. // *P is the size of the population*;
Set $g = 0$ and maximum number of iterations $g_{\max}$;
Calculate fitness for population and find the best individual $\mathbf{x}^{\mathrm{best}}$
while $g < g_{\max}$ **do**
  for $p \in [1 : P]$ **do**
    Calculate the energy factor A [15];
    if $A > 1$ **then**
      if $U(0,1) > 0.5$ **then**
        Calculate the solution $\mathbf{x}_p^{g+1}$ using Eq. (18);
      else
        Calculate the solution $\mathbf{x}_p^{g+1}$ using Eq. (19);
      end if
    else
      if $U(0,1) > 0.5$ **then**
        Calculate the solution $\mathbf{x}_p^{g+1}$ using Eq. (20);
      else
        Apply ARO's original updating rule;
      end if
    end if
    Retain the better solution based on fitness value;
  end for
  Update the best solution $\mathbf{x}^{\mathrm{best}}$;
end while
Return: $\mathbf{x}^{\mathrm{best}}$;

LUND UNIVERSITY

Outline

1. Introduction, Problem statement, System model
2. Optimization problem formulation
3. Proposed meta-heuristic algorithm
4. ***Experimental results***
5. Conclusion and future work

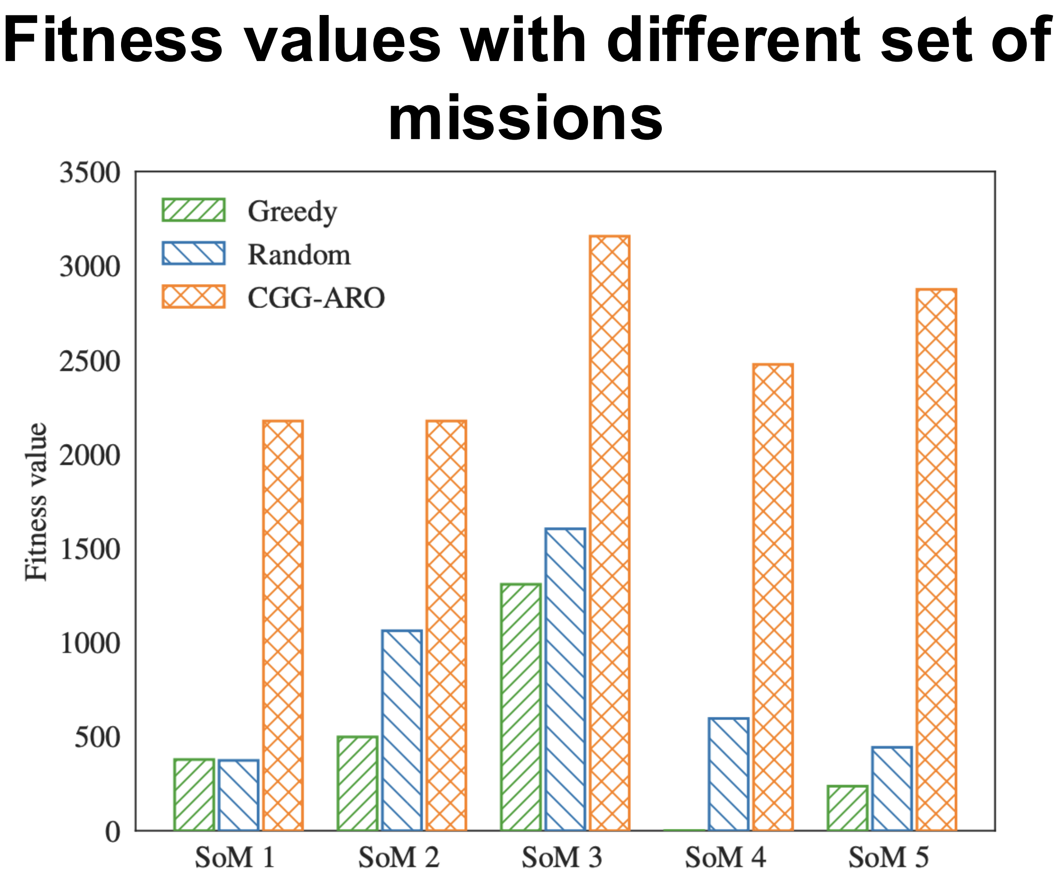
Experimental results

Mean and Standard Deviation of Algorithms Results over 15 runs

Model	Fitness	Completed missions	Total benefits
APO	2834.6±51.5	22.8±0.6	1138.2±30.9
SHADE	2876.8±68.4	23.2±0.8	1158.2±39.4
L-SHADE	2885.6±68.8	23.4±0.9	1168.2±43.4
EO	2868.3±48.2	23.3±0.7	1164.8±35.6
ARO	2909.0±76.1	23.7±0.9	1181.5±45.4
CGG-ARO	**2941.2±84.6**	**24.0±1.0**	**1198.2±49.4**

- *CGG-ARO has outstanding results in all measurement metrices*

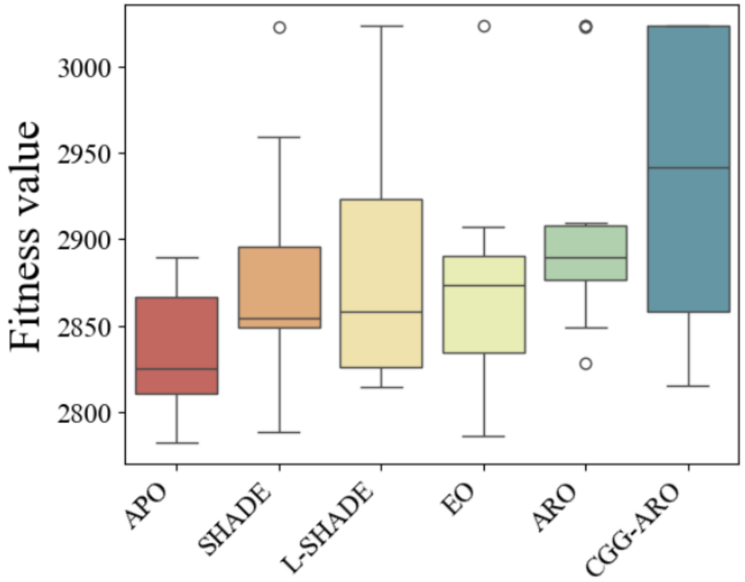
Experimental results



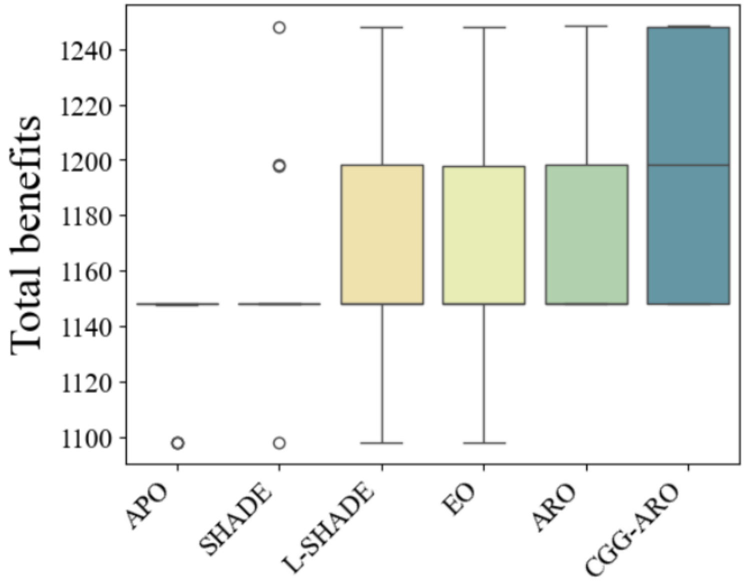
- *CGG-ARO archives excellent performance compared to others*

Experimental results

Boxplot comparison of (a) the fitness value and (b) the total benefits



(a) Fitness value.



(b) Total benefits.

- CGG-ARO consistently outperforms all baselines across all measured metric

Outline

1. Introduction, Problem statement, System model
2. Optimization problem formulation
3. Proposed meta-heuristic algorithm
4. Experimental results
5. ***Conclusion and future work***

LUND UNIVERSITY

Conclusion and future work

Conclusion

- CGG-ARO optimizes mission assignment & task offloading in Open RAN–ITS.
- Considers dependencies, deadlines, cost, delays.
- Outperforms baselines: more missions completed, higher benefits.
- Shows strong potential for real-time ITS optimization.

Future Work

- Larger-scale / multi-region ITS.
- Energy-aware vehicle scheduling.
- Hybrid ML + optimization approaches.
- Real Open RAN/MEC testbed evaluation.

LUND UNIVERSITY