Robust Road-Side Unit Location Problem

Chenyi FU[a], Minglong ZHOU[b], Ning ZHU[c]

[a] School of Management, Northwestern Polytechnical University, Xi'an 710129, China
[b] Department of Management Science, School of Management, Fudan University, Shanghai, China
[c] College of Management and Economics, Tianjin University, Tianjin 300072, China

January 7, 2025

西北工業大學·管理学院
SCHOOL OF MANAGEMENT
NORTHWESTERN POLYTECHNICAL UNIVERSITY

Outline

Introduction

Background

- Autonomous vehicle (AV) technology is expected to bring a fundamental transformation of current transportation systems.
- The adaptation of new AV technology can make our daily travel activity safer, more economical, and more environmentally friendly.
- Many giant companies such as Google, Toyota, Audi, DiDi, and research institutions are investing in the development of AVs.
- However, focusing on AV technology on the vehicle side is not easy.
  1. Many expensive sensors make the vehicle costly.
  2. Hard to overcome complicated road situations worldwide.

Introduction

Background

- Cloud services have seen great growth and application in various fields.
- RSU (Road-Side Unit) is an infrastructure-enabled device of cloud computing in the field of transportation.
- RSU can reduce the data processing needs of AVs and provide them with services such as
  1. Road network data interaction,
  2. Driving data analysis,
  3. Real-time route scheduling,
  4. Personalized travel design,
  5. ...
- RSU network design is a potential direction for development in the AV field.

Introduction

Background

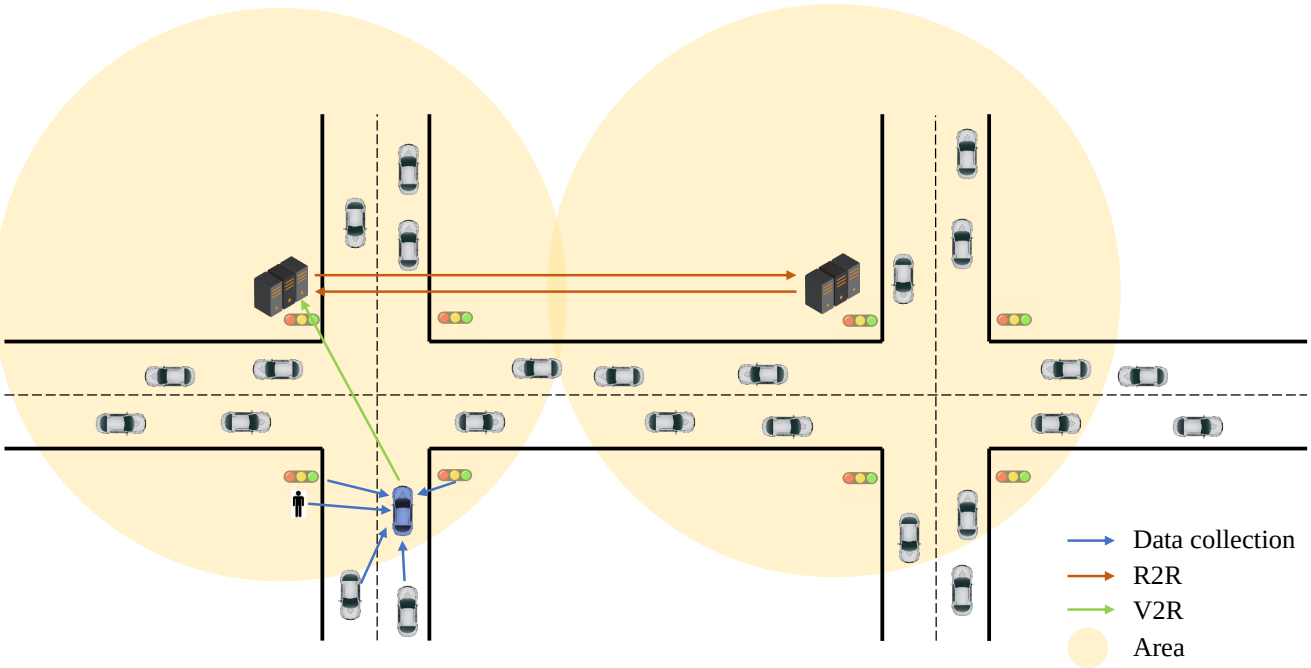


Figure: Illustration of the RSU system.

Literature review

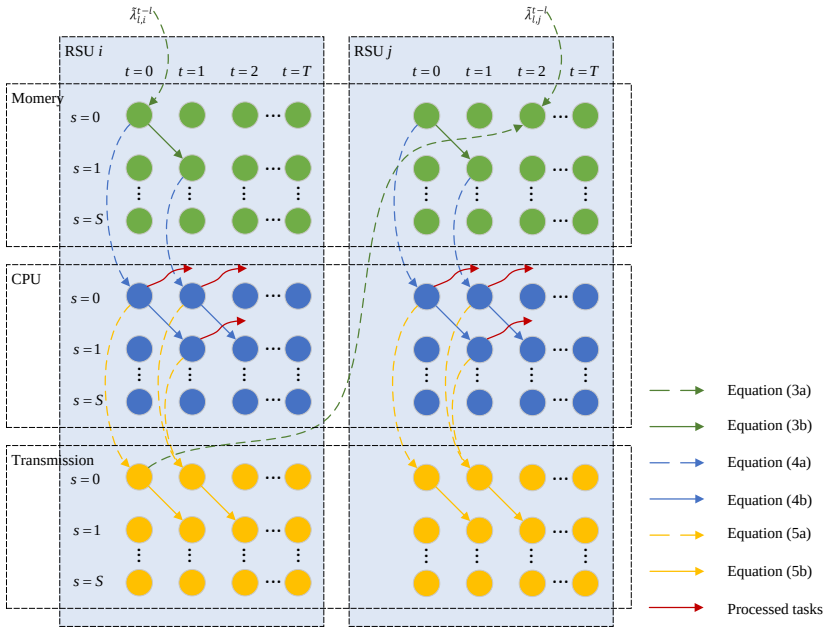
- Vehicle edge computing
  1. Game-theoretic model: Passacantando et al. (2016); Wang et al. (2022a);
  2. Non-convex optimization: Liang et al. (2021c); Qi et al. (2020); Yuan et al. (2020);
  3. Chance-constraint optimization: Cohen et al. (2019); Liu et al. (2019);
  4. Markov decision process: Zhang et al. (2020);
  5. Online algorithm: Perez-Salazar et al. (2022);
- RSU location
  1. Dai et al. (2018); Li et al. (2020); Salari et al. (2022); Liang et al. (2020); Liang et al. (2021b); Cao et al. (2021); Silva et al. (2015); Nikookaran et al. (2017);

These works do not integrate various uncertainties, including processing efficiency, and transmission delay.

Literature review

- Facility location problems with congestion
  1. Elhedhli (2006); Zhang et al. (2009); Berman and Krass (2019); Zhang et al. (2010); Aboolian et al. (2016); Aboolian et al. (2022); Liang et al. (2021a)
- Robust satisficing and riskiness
  1. Elhedhli (2006); Zhang et al. (2009); Berman and Krass (2019); Zhang et al. (2010); Aboolian et al. (2016); Aboolian et al. (2022); Liang et al. (2021a)
- P-Queue
  1. Bandi and Loke (2018); Jaillet et al. (2022); Zhou et al. (2022a); Zhou et al. (2022b); Fu et al. (2022); Tang et al. (2020)

Problem description



$$\tilde{w}_i^{t,0}(l) := \tilde{w}_{i,1}^{t}(l) + \tilde{w}_{i,2}^{t}(l) \tag{1a}$$

$$\tilde{w}_i^{t,s}(l) = \tilde{w}_i^{t-1,s-1}(l) - q_{l,i}^{t,s} \tag{1b}$$

$$y_i^{t,0}(l) = \sum_{s\in[0;\bar{s}]} q_{l,i}^{t+1,s+1}, \tag{1c}$$

$$\tilde{y}_i^{t,s}(l) \sim \text{Bin}\left(\tilde{y}_i^{t-1,s-1}(l)(1-\beta_{l,i}^{t,s}), 1-\gamma_i^{t,s}\right) \tag{1d}$$

$$\tilde{z}_i^{t,0}(l) := \sum_{s\in[0;\bar{s}]} \tilde{y}_i^{t,s}(l)\beta_{l,i}^{t+1,s+1} \tag{1e}$$

$$\tilde{z}_i^{t,s}(l) := \tilde{z}_i^{t-1,s-1}(l) - \sum_{j\in[J]/\{i\}} m_{l,i,j}^{t,s} \tag{1f}$$

where

$$\tilde{w}_{i,1}^{t,0}(l) := \sum_{j\in[J]} \tilde{\lambda}_{l,j}^{t-l} d_{j,i}$$

$$w_{i,2}^{t}(l) = \sum_{j\in[J]/\{i\}} \sum_{s\in[0;\bar{s}-r_{j,i}]} m_{l,j,i}^{t+1-r_{j,i},s+1}.$$

Model formulation

First-stage decision

$$\sum_{i\in[J]} x_i \leq B, \tag{2a}$$

$$\sum_{i\in[J]} \left(\sum_{n\in\{q,y,z\}} \hat{c}_n L_{n,i} + \sum_{n\in\{w,y,z\}} \bar{c}_n C_{n,i} \right) \leq \bar{B}, \tag{2b}$$

$$\sum_{n\in\{q,y,z\}} L_{n,i} + \sum_{n\in\{w,y,z\}} C_{n,i} \leq M x_i, \tag{2c}$$

$$\boldsymbol{C}, \boldsymbol{L} \geq 0, \boldsymbol{x}, \tag{2d}$$

Constraint (2a) restricts the number of operating RSUs. Constraint (2b) restricts the budgets of the configurations of RSUs. Constraint (2c) implies that the customized configurations are considered only when an RSU is deployed in area i, where M is a large number. Constraint (2d) emphasizes the variable ranges in RSU configurations.

Model formulation

Second-stage decision

$$\sum_{i\in[J]} d_{i,j} \leq M x_j, \qquad \forall\ i \in [J], \tag{3a}$$

$$\sum_{j\in[J]} d_{i,j} = 1, \qquad \forall\ i \in [J], \tag{3b}$$

$$\sum_{l\in[\underline{T}_L;t]} \sum_{s\in[0;\bar{s}]} q_{l,i}^{t+1,s+1} \leq L_{q,i}, \qquad \forall\ i \in [J],\ \forall\ t \in [0;T] \tag{3c}$$

$$\sum_{j\in[J]/\{i\}} \sum_{l\in[\underline{T}_L;t]} \sum_{s\in[0;\bar{s}]} m_{l,i,j}^{t+1,s+1} \leq L_{z,i} \qquad \forall\ i \in [J],\ \forall\ t \in [0;T], \tag{3d}$$

Constraints (3a) and (3b) imply that the task collected in area i needs to be offloaded to exactly one area with an operating RSU, where $\boldsymbol{d}$ is a binary variable.
Constraints (3c)-(3d) restrict the number of tasks transmitted from local memory to CPU and between RSU pairs.

Model formulation

Second-stage decision

$$\mathbb{P}\left[\sum_{l\in[\underline{T}_L;t]}\sum_{s\in[0;\bar{s}]} a_{n,l,i}^{t,s}\tilde{n}_i^{t,s}(l) \le C_{n,i}\right] \ge 1-\epsilon \quad \forall\ i\in[J],\ \forall\ t\in[0;T], \tag{4}$$

Constraint (4) requires that the number of tasks in the three modules do not exceed their capacity limits with a probability $1-\epsilon$. For all $i \in [J]$, $t \in [0;T]$, $l \in [0;t]$, we have

$$\mathbb{P}\left[q_{l,i}^{t+1,s+1} \le \tilde{w}_i^{t,s}(l)\right] \ge 1-\epsilon \qquad \forall\ s\in[0;\bar{s}], \tag{5a}$$

$$\mathbb{P}\left[\sum_{j\in[J]/\{i\}} m_{l,i,j}^{t+1,s+1} \le \tilde{z}_i^{t,s}(l)\right] \ge 1-\epsilon \qquad \forall\ s\in[0;\bar{s}], \tag{5b}$$

$$0 \le \beta_{l,i}^{t,s} \le 1 \qquad \forall\ s\in[\bar{s}+1]. \tag{5c}$$

Constraints (5a) and (5b) are special cases of Constraint (4). Constraint (5c) defines the feasibility of variable $\beta_{l,i}^{t,s}$.

Model formulation

Second-stage decision

To avoid having too many tasks uncompleted within a window of L periods, we need to introduce the following latency constraint

$$\mathbb{P}\left[\Gamma \sum_{i,j\in[J]} \sum_{t\in[I;I+L]} \tilde{\lambda}_{I,i}^{t-I} d_{i,j} \leq F_{y,I}(\tilde{\boldsymbol{y}}, \boldsymbol{\beta})\right] \geq 1-\epsilon \quad \forall\ I \in [0;\ T-L], \tag{6}$$

where we define the total processed task by RSUs within a window of L periods:

$$F_{y,I}(\tilde{y}, \boldsymbol{\beta}) := \sum_{i\in[J]} \sum_{t\in[I;I+L]} \sum_{s\in[0;t-I]} \text{Bin}\left(\tilde{y}_i^{t,s}(I)(1-\beta_{I,i}^{t+1,s+1}), \gamma_i^{t+1,s+1}\right).$$

Similarly to constraint (6), we can formulate the constraint for processing capacity:

$$\mathbb{P}\left[G_{y,i,t}(\tilde{\boldsymbol{y}}, \boldsymbol{\beta}) \leq L_{y,i}\right] \geq 1-\epsilon \quad \forall\ i \in [J],\ \forall\ t \in [0;\ T], \tag{7}$$

where we define the total size of tasks processed by the RSU in area i at time t:

$$G_{y,i,t}(\tilde{\boldsymbol{y}}, \boldsymbol{\beta}) := \sum_{I\in[\underline{T}_L;t]} \sum_{s\in[0;t-I]} \text{Bin}\left(\tilde{y}_i^{t,s}(I)(1-\beta_{I,i}^{t+1,s+1}), \gamma_i^{t+1,s+1}\right).$$

Robust Satisficing RSU Location Model

We utilize the entropic risk measure and tractably incorporate operational constraints involving random variables as risk-based constraints (e.g., Zhou et al. 2022a). For completeness, we define the entropic risk measure below.

Definition 1 (Entropic risk measure)

For a random variable $\tilde{\zeta} \sim \mathbb{P}$, the entropic risk measure with a risk parameter θ, $\mu_\theta(\tilde{\zeta})$, is defined as:

$$\mu_\theta(\tilde{\zeta}) := \theta \log \mathbb{E}_{\mathbb{P}}\left[\exp\left(\tilde{\zeta}/\theta\right)\right]. \tag{8}$$

When dealing with an operational constraint, $\tilde{\zeta} \leq 0$, involving a random variable, we focus on its risk-based counterpart, $\mu_\theta(\tilde{\zeta}) \leq 0$. For instance, the risk-based capacity constraint (4) is

$$\mu_\theta\left(\sum_{l\in[\underline{T}_L;t]}\sum_{s\in[0;\bar{s}]} a_{n,l,i}^{t,s}\tilde{n}_i^{t,s}(l) - C_{n,i}\right) \leq 0 \quad \forall\, i \in [J],\ \forall\, t \in [0;T],\ \forall\, n \in \{w,y,z\}.$$

Robust Satisficing model

$$\text{lexmin } \boldsymbol{\theta}$$

$$\text{s.t. } \mu_{\theta_{n,i,t}} \left(\sum_{l \in [\underline{T}_L; t]} \sum_{s \in [0; \bar{s}]} a_{n,l,i}^{t,s} \tilde{n}_i^{t,s}(l) - C_{n,i} \right) \leq 0 \quad \forall\, i \in [J],\ \forall\, t \in [0; T],\ \forall\, n \in \{w, y, z\} \tag{9a}$$

$$\mu_{\theta_{1,i}} \left(q_{l,i}^{t+1,s+1} - \tilde{w}_i^{t,s}(l) \right) \leq 0 \qquad \forall\, i \in [J],\ \forall\, t \in [0; T-1],\ \forall\, l \in [0; t],\ \forall\, s \in [0; \bar{s}] \tag{9b}$$

$$\mu_{\theta_{2,i}} \left(\sum_{j \in [J]/\{i\}} m_{l,i,j}^{t+1,s+1} - \tilde{z}_i^{t,s}(l) \right) \leq 0 \qquad \forall\, i \in [J],\ \forall\, t \in [0; T-1],\ \forall\, l \in [0; t],\ \forall\, s \in [0; \bar{s}] \tag{9c}$$

$$\mu_{\theta_0} \left(\sum_{i,j \in [J]} \sum_{t \in [l; l+L]} \Gamma \tilde{\lambda}_{l,i}^{t-l} d_{i,j} - F_{y,l}(\tilde{\boldsymbol{y}}, \boldsymbol{\beta}) \right) \leq 0,\ \forall\, l \in [0; T-L] \tag{9d}$$

$$\mu_{\theta_{3,i}} \left(G_{y,i,t}(\tilde{\boldsymbol{y}}, \boldsymbol{\beta}) - L_{y,i} \right) \leq 0 \qquad \forall\, i \in [J],\ \forall\, t \in [0; T] \tag{9e}$$

$$\text{Dynamics (1), Constraints (2), (3), (5c).}$$

Reformulation

We assume that inflows $\tilde{\boldsymbol{\lambda}} \sim \Lambda$ are space- and time-independent, and their moment-generating functions exist.

Theorem 2

$$\mu_{\theta_{y,i,t}} \left(\sum_{l \in [\underline{T}_L; t]} \sum_{s \in [0; \bar{s}]} a^{t,s}_{y,l,i} \tilde{y}^{t,s}_i(l) - C_{y,i} \right) \leq 0,$$

can be reformulated as the following set of affine or convex constraints for a fixed $\boldsymbol{\theta}$*:*

$$\sum_{l \in [0;t]} \left(a^{t,0}_{y,l,i} y^{t,0}_i(l) + \theta_{y,i,t} \sum_{\tau=0}^{\bar{s}-1} \xi^{t-\tau,1}_{y,l,i} \right) + \sum_{l \in [\underline{T}_L; -1]} \left(\sum_{s=t}^{\bar{s}} \theta_{y,i,t} \xi^{1,s-t+1}_{y,l,i} a^{t,0}_{y,l,i} y^{t,0}_i(l) + \theta_{y,i,t} \sum_{\tau=0}^{t-2} \xi^{t-\tau,1}_{y,l,i} \right) \leq C_{y,i}$$

$$\xi^{t,s}_{y,l,i} \geq \eta^{t,s}_{y,l,i} \rho \left(\frac{a^{t,s}_{y,l,i}}{\theta_{y,i,t}}, 1 - \gamma^{t,s}_i \right) \qquad \forall\, l \in [\underline{T}_L; t],\ \forall\, s \in [\bar{s}]$$

$$\xi^{t-\tau,s-\tau}_{y,l,i} \geq \eta^{t-\tau,s-\tau}_{y,l,i} \rho \left(\frac{\xi^{t+1-\tau,s+1-\tau}_{y,l,i}}{\eta^{t-\tau,s-\tau}_{y,l,i}}, 1 - \gamma^{t-\tau,s-\tau}_i \right) \qquad \forall\, l \in [\underline{T}_L; t],\ \forall\, \tau \in [\underline{s} - 1],\ \forall\, s \in [\tau + 1; \bar{s}],$$

where we define $\rho(y, p) = \log(1 - p + p \exp(y))$.

Solution approach

Bisection search

1. Initialize with $\mathcal{C}_0 = \emptyset$ and $\mathcal{C}_1 = [C]$.
2. Set $\underline{\theta} = 0$ and $\bar{\theta} = M$, where M is a large number.
3. If $\bar{\theta} - \underline{\theta} \leq \epsilon$, then set $\theta^* = (\underline{\theta} + \bar{\theta})/2$ and find the index set $\mathcal{C}_b \subseteq \mathcal{C}_1$ such that any constraint indexed by $c \in \mathcal{C}_b$ is binding. We record θ^* and $\mathcal{C}_b$, and go to step 6. Otherwise, set $\theta = (\underline{\theta} + \bar{\theta})/2$ and go to the next step.
4. Set $\theta_c = \theta_c^*$ for $c \in \mathcal{C}_0$, and $\theta_c = \theta$ for $c \in \mathcal{C}_1$. Solve the following feasibility problem by cutting plane:

$$\begin{array}{ll} \min & 0 \\ \text{s.t.} & \text{Deterministic convex counterpart of constraints (9a)-(9e)} \\ & \text{Constraints (3), 5(c).} \end{array} \tag{10}$$

5. If Problem (10) is feasible, set $\bar{\theta} = \theta$. Otherwise, set $\underline{\theta} = \theta$ and go back to step 3.
6. Let $\theta_c^* = \theta^*$ for all $c \in \mathcal{C}_b$. Then, include all elements of $\mathcal{C}_b$ in $\mathcal{C}_0$ and exclude them from $\mathcal{C}_1$.
7. If $\mathcal{C}_1 = \emptyset$, then we terminate the algorithm. Otherwise, repeat the process from step 2.

Solution approach

Acceleration methods

Theorem 3

The set of constraints in Theorem 2 can be reformulated as the following collection of constraints:

$$\sum_{l\in[0;t]}\left(a^{t,0}_{y,l,i}y^{t,0}_i(l)+\sum_{\tau=0}^{\bar{s}-1}\xi^{t-\tau,1}_{y,l,i}\right)+\sum_{l\in[\underline{T}_L;-1]}\left(\sum_{s=t}^{\bar{s}}\xi^{1,s-t+1}_{y,l,i}+a^{t,0}_{y,l,i}y^{t,0}_i(l)+\sum_{\tau=0}^{t-2}\xi^{t-\tau,1}_{y,l,i}\right)\leq\bar{C}_{y,i,t}$$

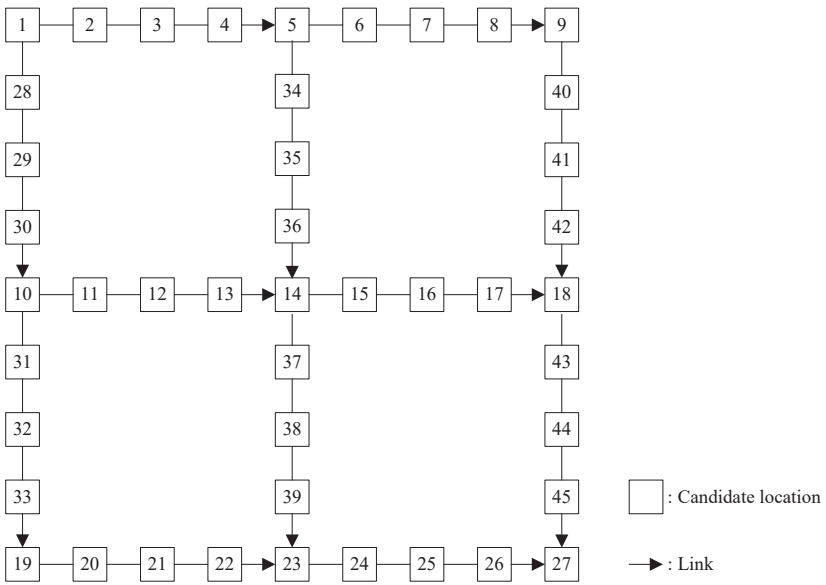
$$\xi^{t,s}_{y,l,i}\geq\theta_{y,i,t}\eta^{t,s}_{y,l,i}\rho\left(\frac{a^{t,s}_{y,l,i}}{\theta_{y,i,t}},1-\gamma^{t,s}_i\right)\qquad\forall\, l\in[\underline{T}_L;t],\ \forall\, s\in[\bar{s}]$$

$$\xi^{t-\tau,s-\tau}_{y,l,i}\geq\theta_{y,i,t}\eta^{t-\tau,s-\tau}_{y,l,i}\rho\left(\frac{\xi^{t+1-\tau,s+1-\tau}_{y,l,i}}{\theta_{y,i,t}\eta^{t-\tau,s-\tau}_{y,l,i}},1-\gamma^{t-\tau,s-\tau}_i\right)\qquad\forall\, l\in[\underline{T}_L;t],\ \forall\,\tau\in[\underline{s}-1],\ \forall\, s\in[\tau+1;\bar{s}]$$

1. Cut reservation: Constraints in Theorem 3 are convex in θ, allowing us to reserve cuts generated in previous iterations
2. Warm start: We can add the set of all risk-based constraints fixing $\theta_c=+\infty$ for all $c\in[C]$.

Numerical Experiments

Experiment setting



We consider two benchmarks.

1. The first benchmark model is a deterministic model that takes the random variables as their expected values, *e.g.*, it assumes a fixed number of tasks offloaded from the vehicles in area i, $\hat{\psi}^s_{i,j} = \lambda_i \bar{\omega}^s_{i,j}$. The deterministic model minimizes the total computation latency.
2. The second benchmark is similar to Liang et al. (2020), which formulates each RSU as an M/G/1 queuing system, and the queuing delay in RSU is represented by the Pollaczek-Khintchine formula.

Numerical Experiments

Results

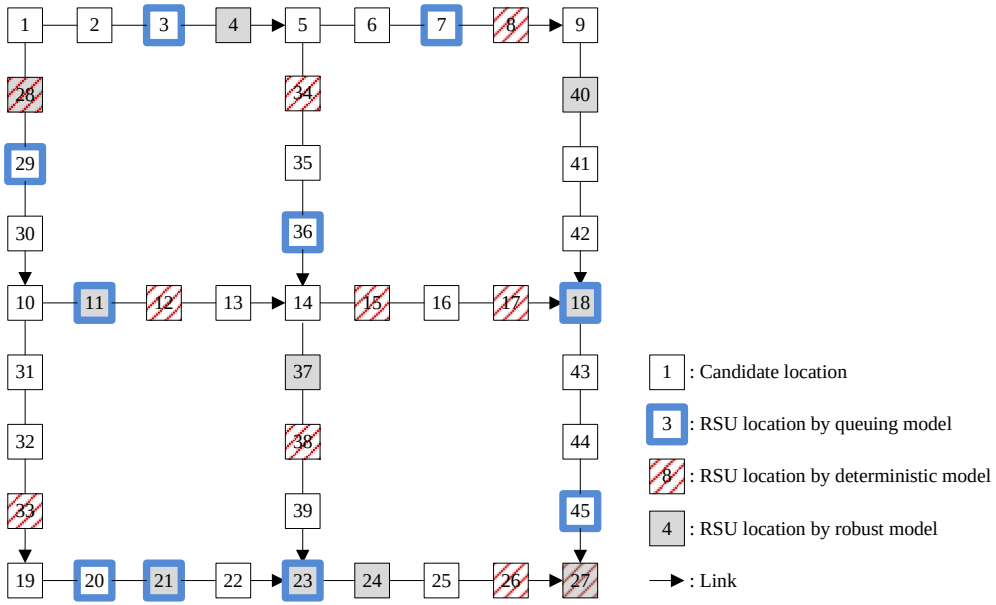


Figure: RSU distribution with fixed $\bar{L}_y$ under $B = 10$.

The deterministic model leads to a less even RSU distribution in the network;
The robust RSU location model and queuing model spread out the facilities more evenly.

Numerical Experiments

Results

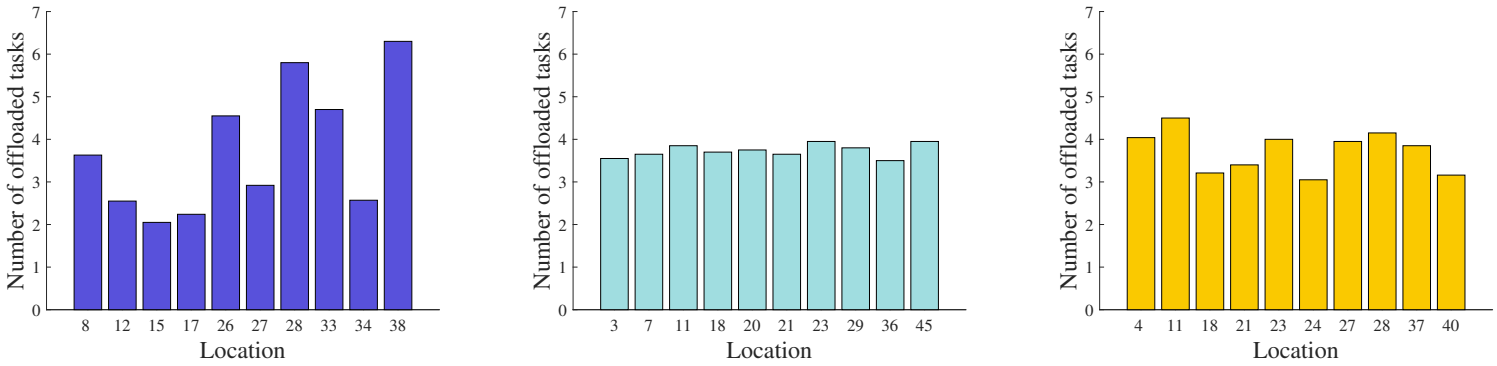
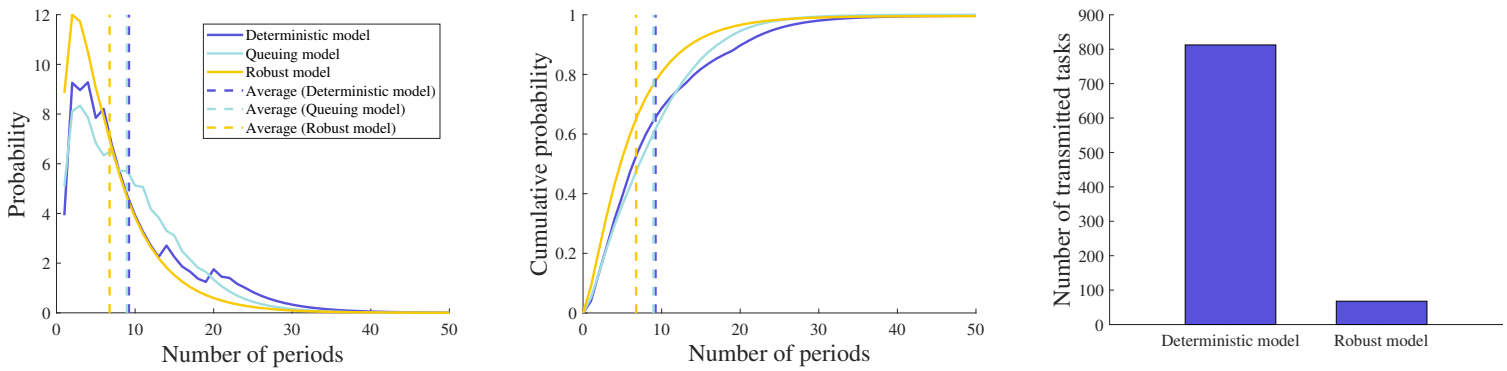


Figure: Average number of the offloaded tasks of the deterministic model (left), queue model (middle), and robust model (right) with fixed processing capacities when $B = 10$.

The queuing and robust models tend to allocate the tasks into each RSU uniformly, while the deterministic model results in an uneven task offloading pattern and violations of the processing capacity, because of the poor RSU locations with uneven coverage.

Numerical Experiments

Results



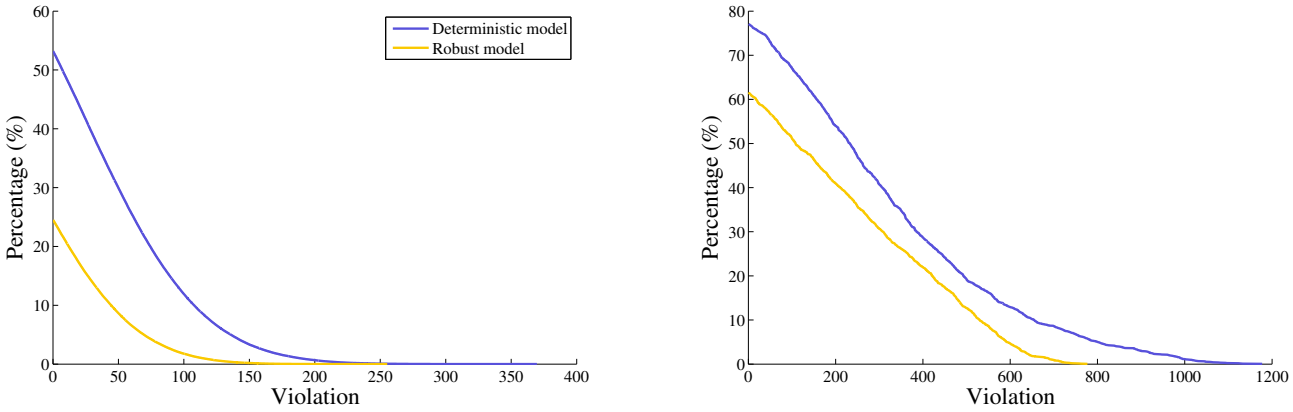
(a) (b) (c)

Figure: Out-of-sample task processing metrics: distribution of completion time (left) and associated cumulative probability distribution (middle) of the processed tasks corresponding to three models, and number of transmitted tasks (right) overall RSUs when $B = 10$.

1. The average completion time of the robust model (6.76 periods) is lower than that of the queuing model (8.96 periods) and the deterministic model (9.22 periods).
2. There is a tiny proportion (0.97%, 0.64%) of tasks whose completion times exceed the latency threshold $L = 35$ under the deterministic and robust models. There are 11.78% of all tasks that cannot be completed by the queue model within L periods.

Numerical Experiments

Results



(a) Without misspecification (b) With misspecification

Figure: Model robustness to safeguard the processing capacity.

Let the random number of task collection in zone i at each period $\tilde{\lambda}_i = \alpha\lambda_i$, where $\alpha \sim U(1, 1.05)$. The robust model outperforms the deterministic model.

Numerical Experiments

Results

B	$\bar{L}_y$	α	Version II		Version III				DM	QM	
			Time (s)	Cuts	Time (s)	Cuts	Time reduction (%)	Cut reduction (%)	Time (s)	Time (s)	Gap (%)
10	4	1	22783	5914	10826	2769	52.48	53.18	43	8647	\
11	4	1	4357	646	2902	347	33.39	46.28	43	10800	15.24
11	4	1.1	1635	190	1438	190	12.06	0.00	40	10800	18.67
12	4	1	5642	1009	4134	722	26.73	28.44	40	10800	9.59
12	4	1.2	3202	792	2275	455	28.95	42.55	40	10800	11.75
13	4	1	5662	1558	3372	680	40.45	56.35	40	10800	5.77
13	4	1.3	3533	1127	2125	434	39.87	61.49	43	10800	5.27
13	4	1.4	29223	11016	10701	4704	63.38	57.30	43	10800	5.27
14	4	1	10728	1962	7480	1313	30.27	33.08	40	10800	1.43
14	4	1.5	20448	6163	9699	2594	52.91	57.91	40	10800	2.66
15	3	1	5664	1690	4241	1143	32.37	32.37	28	10800	20.67

Table: Computing efficiency of cutting plane approaches, the deterministic model, and the queuing model with the fixed capacity.

Version I without warm start (*i.e.,* initial constraints) and cut reservation; Version II with initial constraints; and Version III with a warm start and cut reservation.

Conclusions

1. **Robust satisfising model for RSU location problem.** To the best of our knowledge, this study provides the first robust satisficing model for the RSU location problem combined with dynamic edge computing scheduling with random computing tasks, transmission delay between RSUs, and processing rate in the CPU.
2. **Extending the existing methodology.** We enrich the existing P-Queue techniques.
   - We demonstrate we can jointly optimize facility locations and task allocations in a computationally tractable manner.
   - We utilize an additional state variable to record the timestamp of task input.
3. **Enhancing solution approach.** We present two acceleration strategies. With the warm start, the cut reservation policy can further reduce 12.06%-63.38% computing time and 0%-61.49% generated cuts.
4. **Satisfactory performance in the numerical study.** Compared to the deterministic model, the robust model achieves improvements of 91.65% in task transmission requirements, 26.74% in average task completion time, 30.93% in the magnitude of violation of processing capacity, and 53.92% in the probability of violating processing capacity. Compared with the queuing model, our model leads to a low violation of computational latency.

Thanks for listening