Deep Reinforcement Learning for Delay-Optimized Task Offloading in Vehicular Fog Computing

Mohammad Parsa Toopchinezhad - Mahmood Ahmadi

29th International Computer Conference - Computer Society of Iran - February 2025

Presentation Outline

I Introduction

- The Internet of Vehicles
- Task Offloading
- Reinforcement Learning

II Problem Formulation

- Vehicle Model
- Task Model
- Communication Model
- MDP Specification

III Results

- Simulation Scenario
- Compared Algorithms
- Task Delay
- Queue Length

Part I: Introduction

The Internet of Vehicles

Why future cars will be connected

- 30 years ago, the main edge-devices were normal computers
- Now, almost everything is being connected, including **cars**
- Cars will communicate with each other, transmission towers, and the cloud, forming the **Internet of Vehicles (IoV)**

IoV Applications

Safety

- Collision avoidance
- Accident report
- Early warning

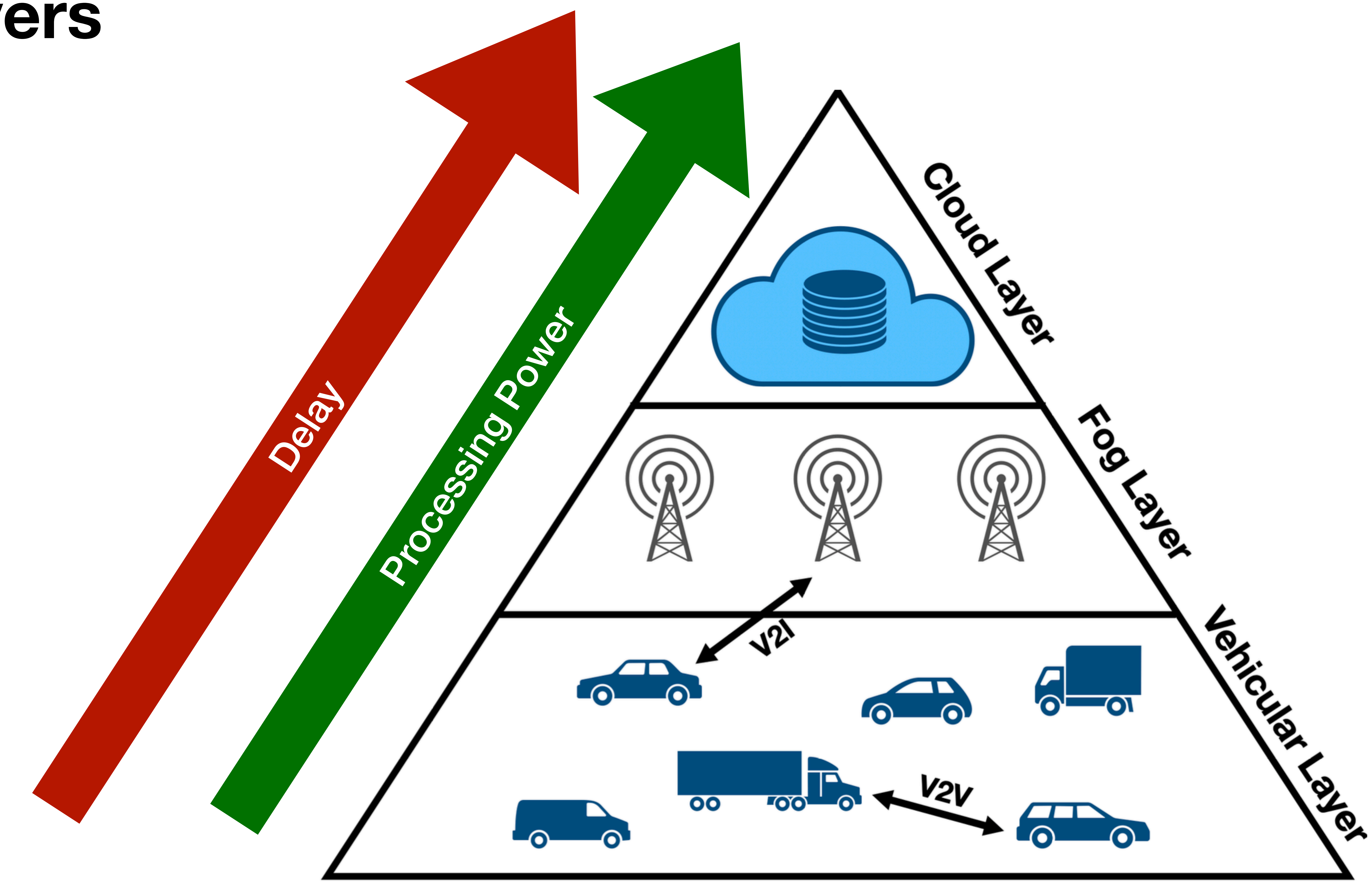
Non-safety

- Gaming
- Content streaming
- Smart parking
- Routing
- Traffic prediction

The Problem of Task Offloading

Offloading at Different Layers

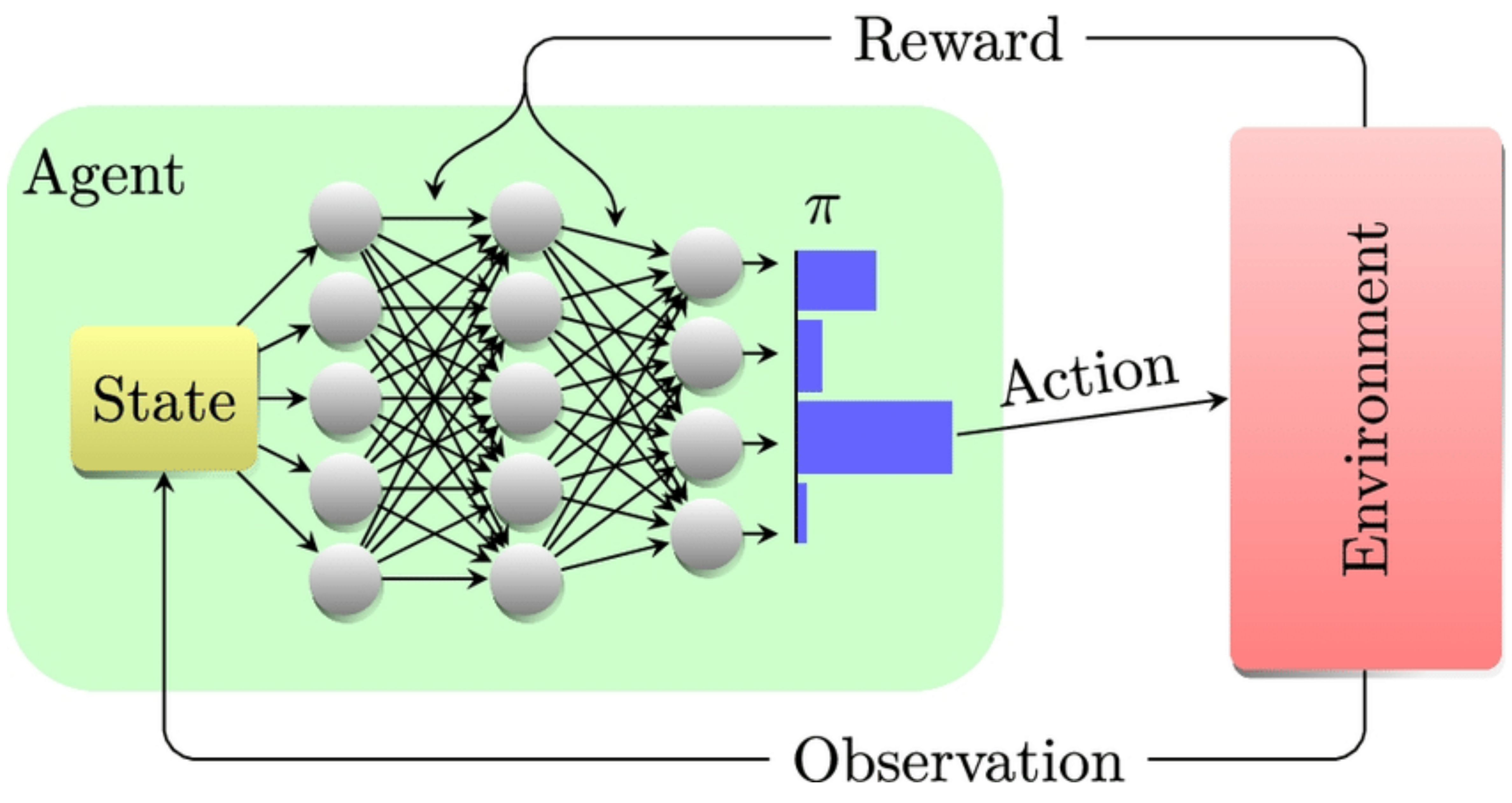
- So, future cars will need to do a lot of processing, but they can benefit from offloading
- **Task offloading**: sending a task to another layer for processing
    - **Edge**: the individual vehicles
    - **Fog**: the communication infrastructure
    - **Cloud**: the powerful servers at the network core
- **Optimization problem:** where to offload task?



Reinforcement Learning

The new paradigm of algorithm design

- **Old method:** humans explicitly think, implement and redesign algorithms
- **New method:** just specify the problem and let a computer discover near-optimal algorithms
- **Reinforcement Learning (RL):** subset of ML instructing agents on problem-solving through repeated trial and error
- **Markov Decision Process (MDP):** mathematical framework of RL



Part II: Problem Formulation

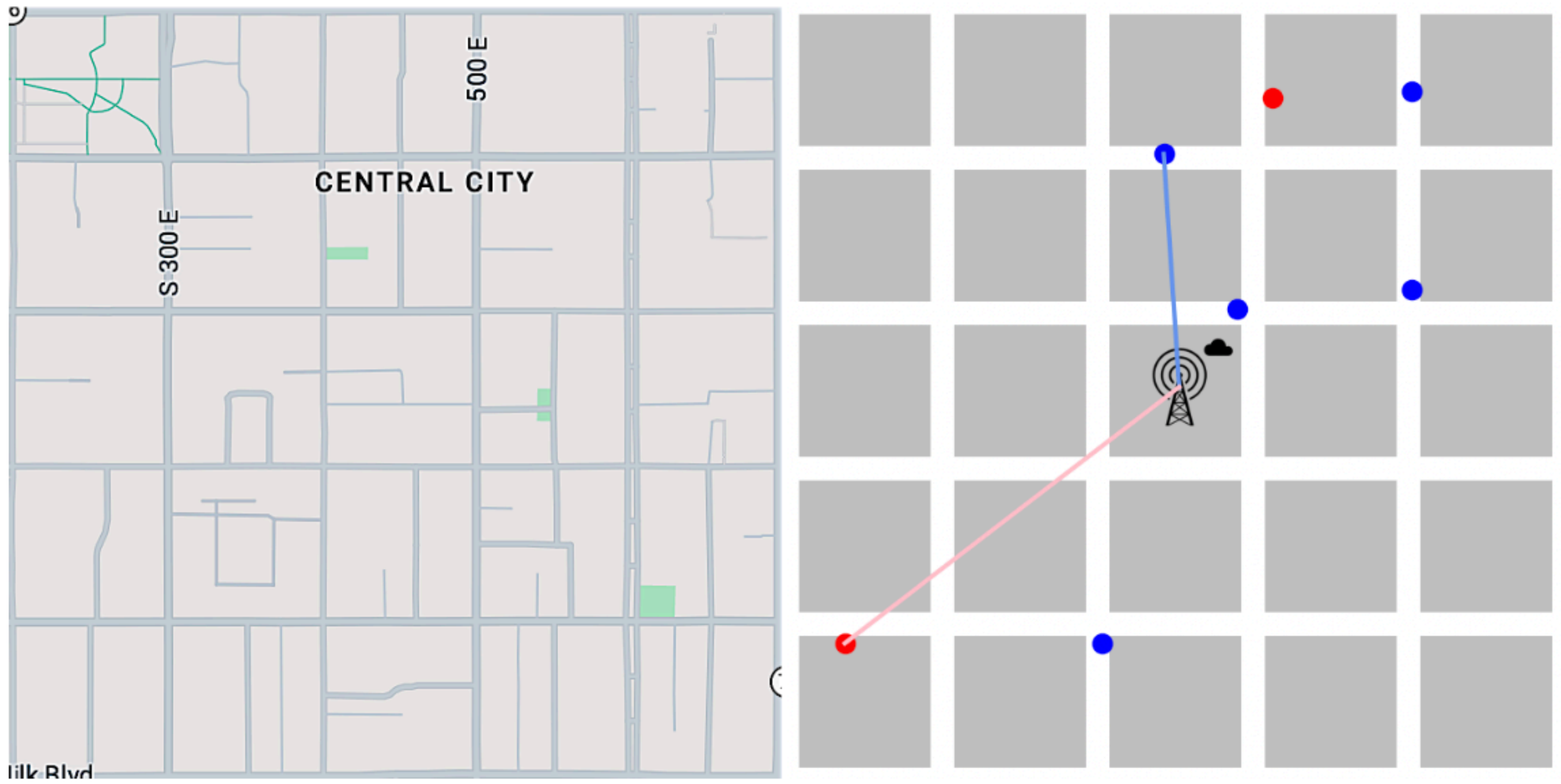
Problem Formulation

Vehicle model

- Vehicles are either moving or parked
- Vehicles are characterized by:
    - (X, Y) coordinates
    - Velocity
    - CPU processing power (based on real hardware)
- Vehicles move randomly in a grid-planned city (common in the USA)

TABLE III: Processor Specifications

Processor Name	Processor Speed (MIPS)
ARM Cortex A72	18,375
AMD Phenom II X4 940	42,820
ARM Cortex A73	71,120



(a) (b)

Fig. 3: Downtown of Salt Lake City, U.S., showcasing its grid-like structure of 200 by 200 meter city blocks (a) (Map data @2024 Google). Simulated environment based on Salt Lake City blocks (b).

Problem Formulation (Cont.)

Task model

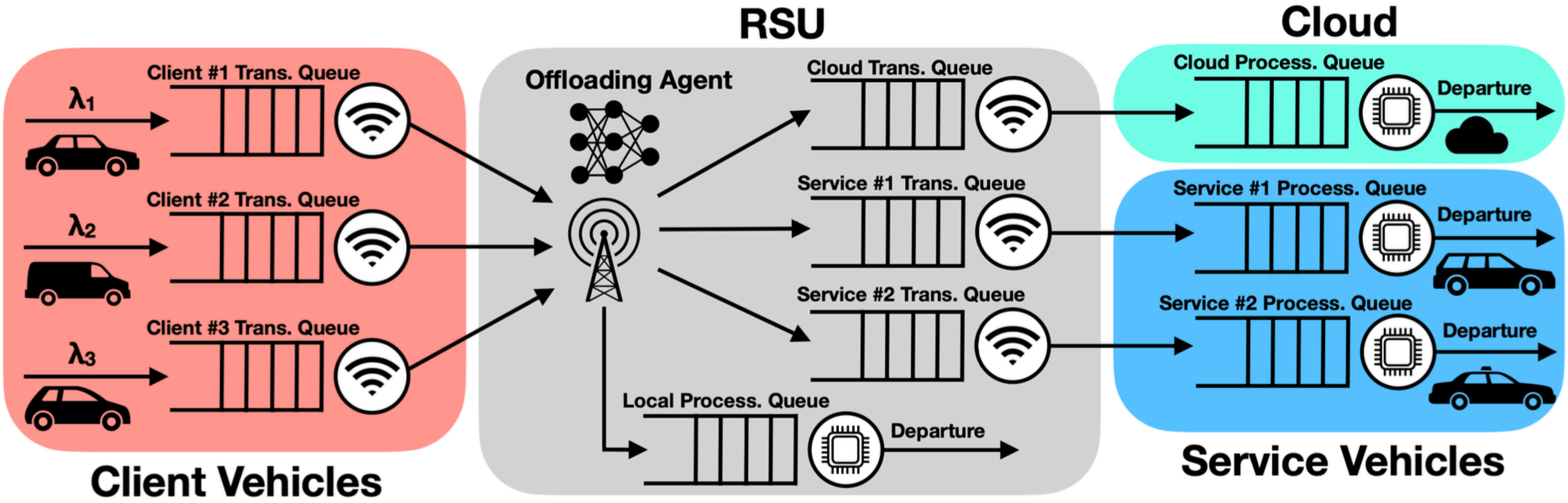
- **Tasks**: a well-defined set of operations which a vehicle creates
- Tasks are characterized by:
    - Instruction count (MI)
    - Size (Mb)
- Tasks are based on **SPEC CPU Dataset**, consisting of 18 tasks with varying computational complexities

SPEC CPU95			
Program	**Input**	**INT/ FP**	**Dynamic Instruction Count**
li	*.lsp	INT	75.6 billion
m88ksim	ctl.in	INT	520.4 billion
compress	bigtest.in	INT	69.3 billion
ijpeg	penguin.ppm	INT	41.4 billion
gcc	expr.i	INT	1.1 billion
perl	perl.in	INT	16.8 billion
vortex	*	INT	*
wave5	wave5.in	FP	30 billion
hydro2d	Hydro2d.in	FP	44 billion
swim	swim.in	FP	30.1 billion
applu	Applu.in	FP	43.7 billion
mgrid	Mgrid.in	FP	56.4 billion
turb3d	turb3d.in	FP	91.9
su2cor	su2cor.in	FP	33 billion
fpppp	natmos.in	FP	116 billion
apsi	apsi.in	FP	28.9 billion
tomcatv	tomcatv.in	FP	26.3 billion

Problem Formulation (Cont.)

Communication model

- The total delay experienced by a task is the sum of the following: $d_{\text{total}} = d_{\text{client}} + d_{\text{rsu}} + d_{\text{service}}$



MDP Specification

- State space:

Task size

Task instruction count

CPU power of vehicles

Trans. & proc. queue lengths

$$s = \{SZ_x, CU_x, PP, DT, Q_t, Q_p\}$$

Relative distance to vehicles

- Action space:

$$A = V \cup \{a_{\text{cloud}}\} \cup \{a_{\text{rsu}}\}$$

- Reward function

Negative of the sum of all task delays

$$R(s,a) = \sum_{x \in U} -d^x_{\text{total}}$$

Part III: Results

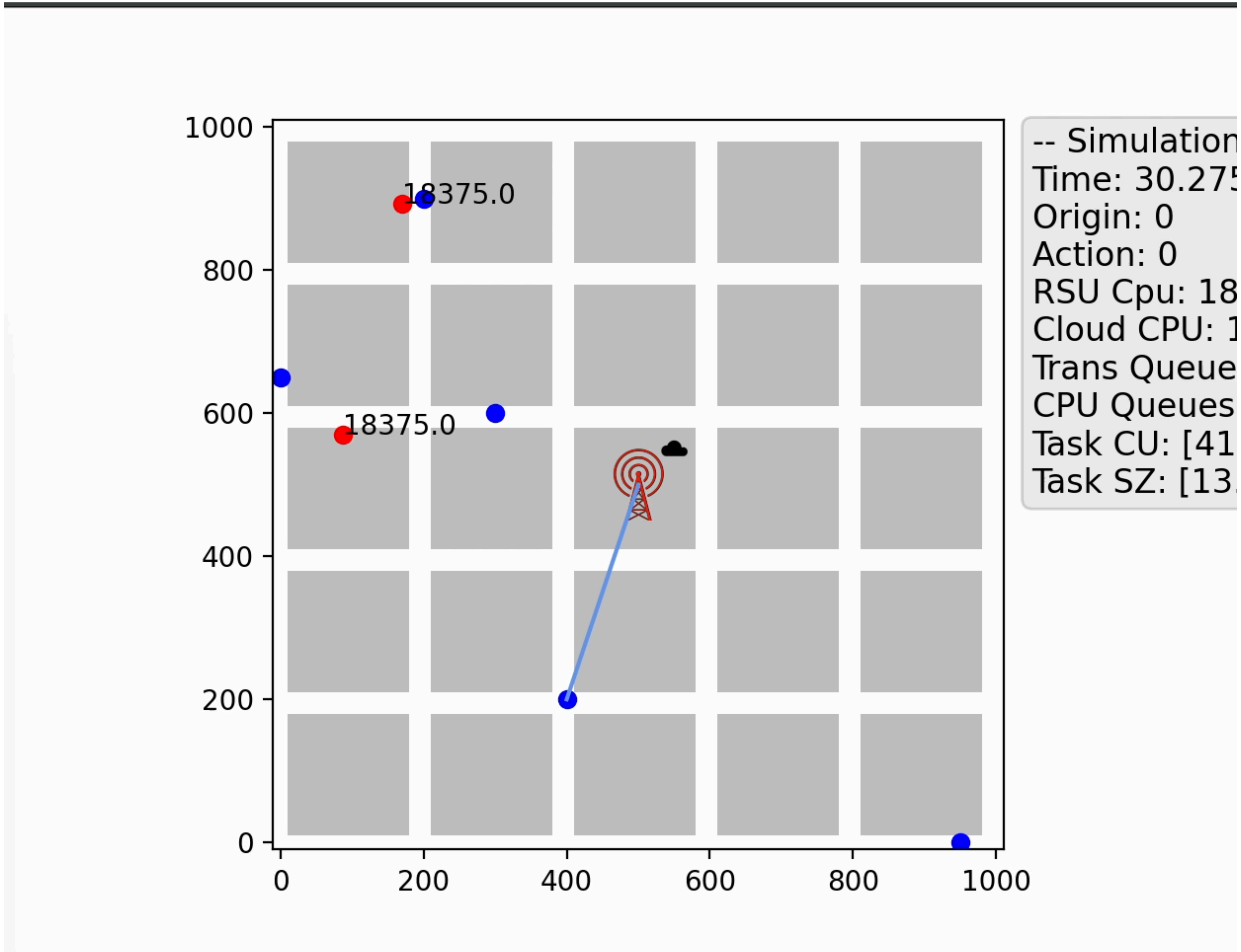
Simulation Scenarios

- Three different scenarios considered:
  - Scenario 1: 5 moving vehicles & 2 parked vehicles
  - Scenario 2: 10 moving vehicles & 4 parked vehicles
  - Scenario 3: 20 moving vehicles & 8 parked vehicles
- Four algorithms compared:
  - Deep RL (PPO)
  - Random
  - Greedy
  - Cloud-only

TABLE II: Environment Settings

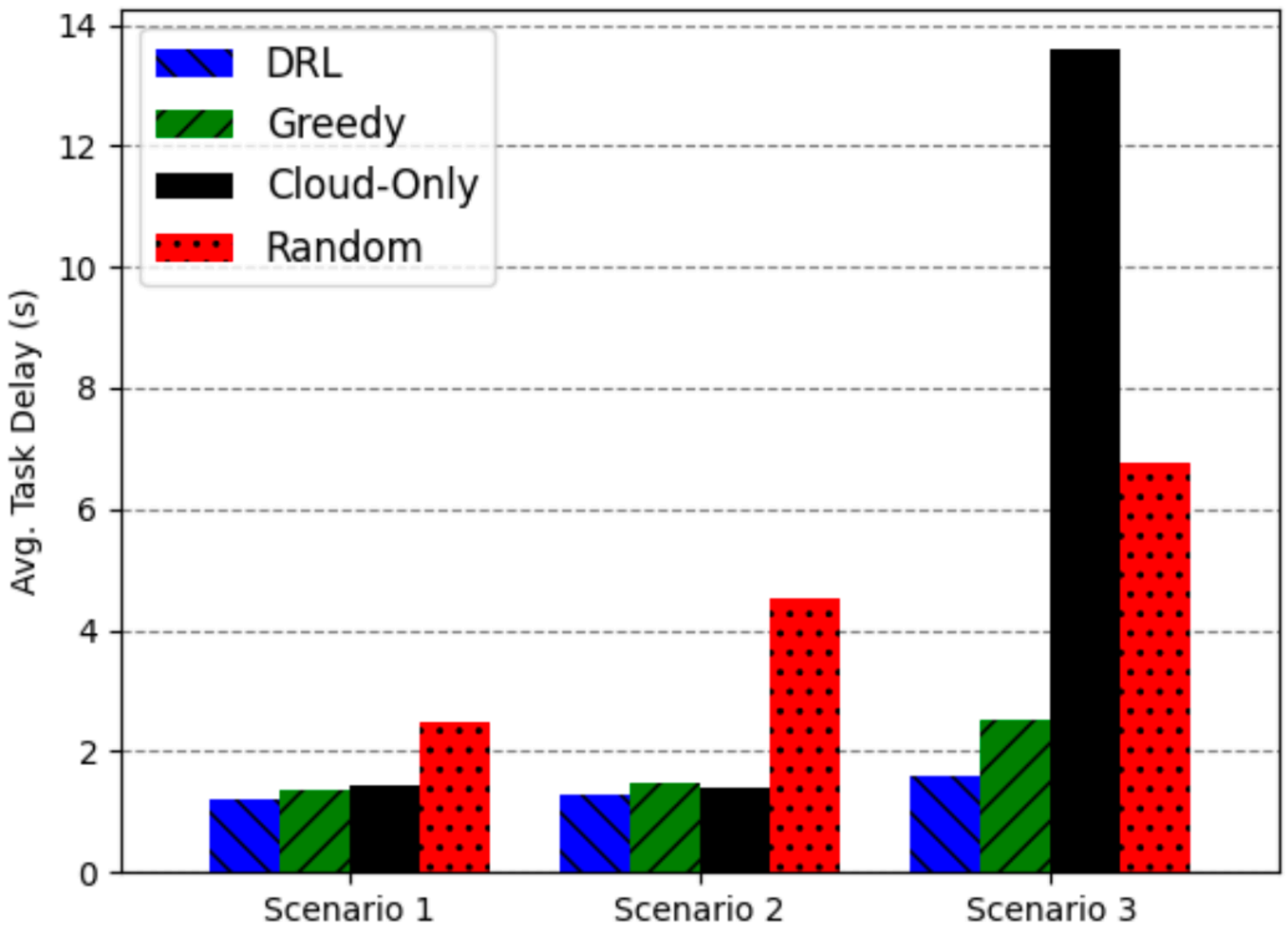
Parameter	Value
$l \times w$	1000 (m) × 1000 (m)
n	5, 10, 20
m	2, 4, 8
λ_i	3 (tasks per sec)
vel_i	$\sim U\{10, 20, 25, 40\}$ (km/h)
SZ_x	$\sim U\{20, 40\}$ (Mb)
B	40 (MHz)
P_T	1 (W)
G_T, G_R	5 (dBi)
N_0	174 (dBm/Hz)
CPU_{rsu}	1×18,375 (MIPS)
CPU_{cloud}	∞×100,000 (MIPS)

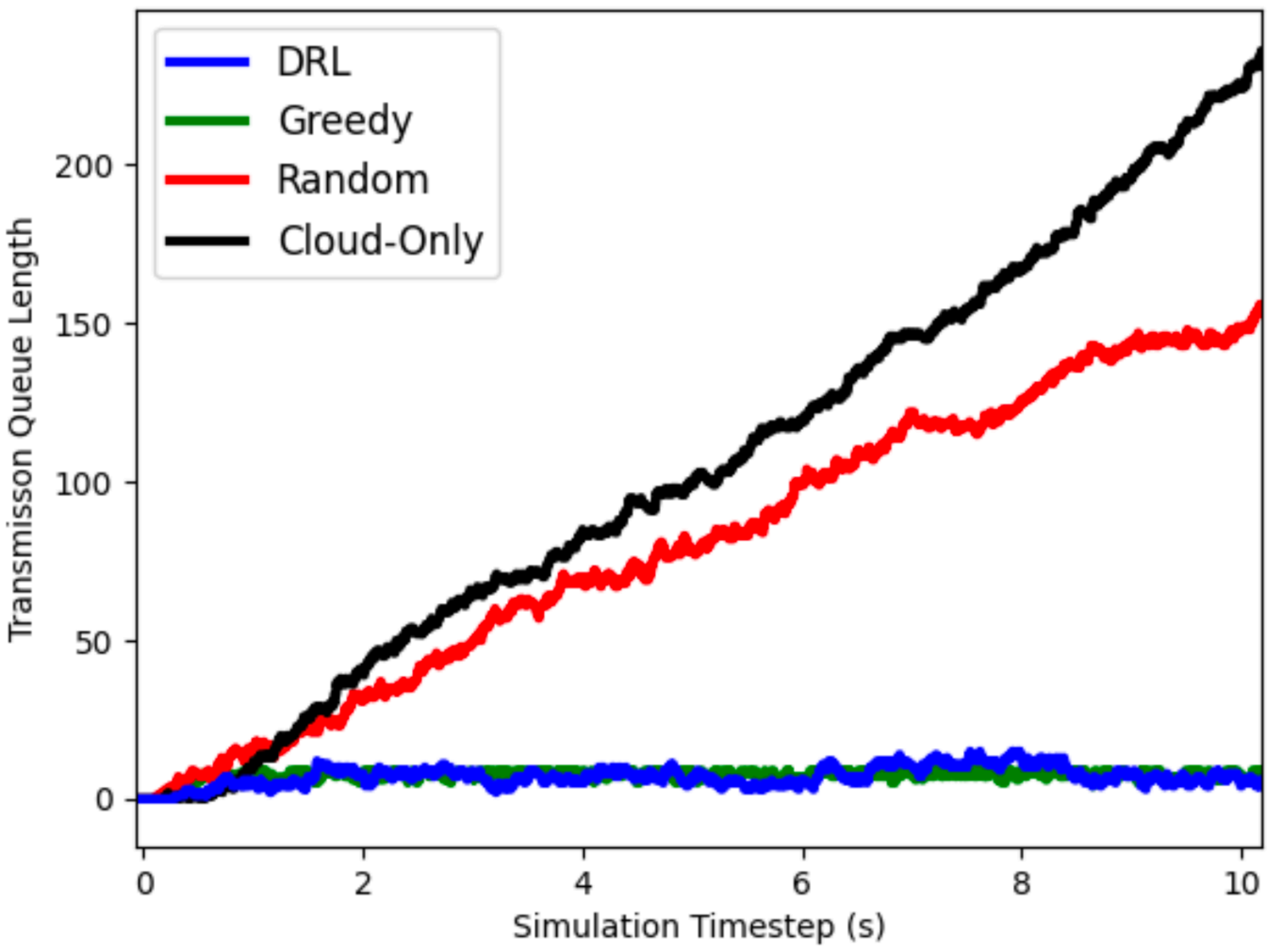
Simulation Animation



Results

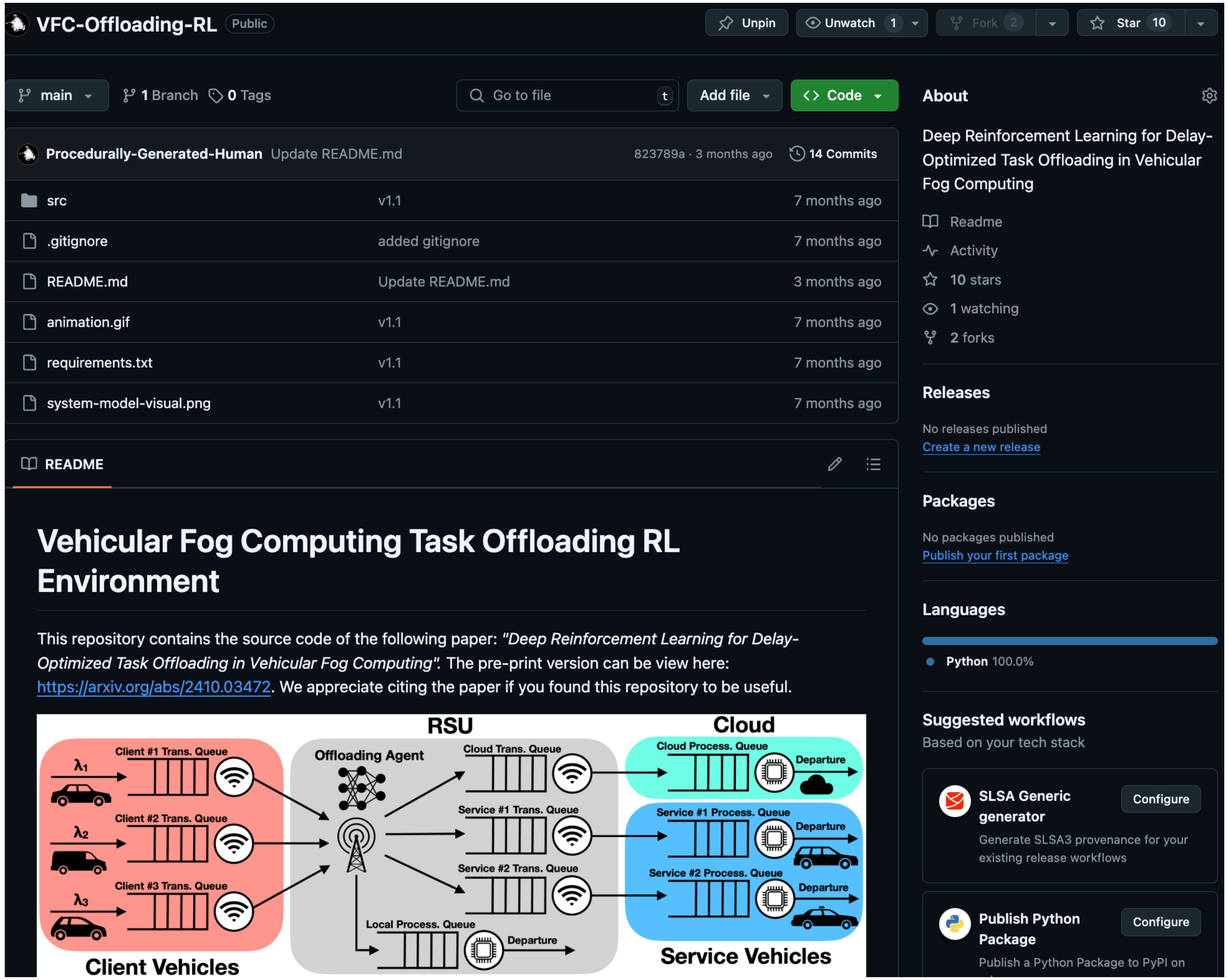
Average task delay & queue length





Code Repository

Open-source, reproducible research



Github repository link:

github.com/Procedurally-Generated-Human/VFC-Offloading-RL

Current stats:

11 stars

2 forks

The End

Thank You

Extra Slide #1

Delay model

$$d_{\text{total}} = d_{\text{client}} + d_{\text{rsu}} + d_{\text{service}}$$

$$d_{\text{client}} = d_{\text{client-queue}} + d_{\text{client-trans}}$$

$$d_{\text{rsu}} = d_{\text{rsu-queue}} + d_{\text{rsu-trans}}$$

$$d_{\text{service}} = d_{\text{service-queue}} + d_{\text{service-process}}$$

$$d_{\text{service-process}} = \frac{CU_x}{CPU_{\text{service}}}$$

$$d_{\text{w-trans}} = \frac{CU_x}{TR_w}$$

TABLE I: Notation used for system model

Symbol	Meaning
l, w	Simulation area length & width (m)
n	Number of client vehicles
m	Number of service vehicles
CPU_j	Processing power of vehicle j (MIPS)
λ_i	Average task emission rate by client i
vel_i	Velocity of client i
CU_x	Number of million instructions of task x
SZ_x	Size of task x (Mb)
TR_w	Transmission rate of medium w (Mbps)
B	Wireless channel bandwidth (MHz)
N_0	Noise Power (dBm/Hz)
P_T	Transmitter's transmission Power (W)
G_T/G_R	Transmitting/Receiving directional gain (dBi)

Extra Slide #2

Transmission queue length

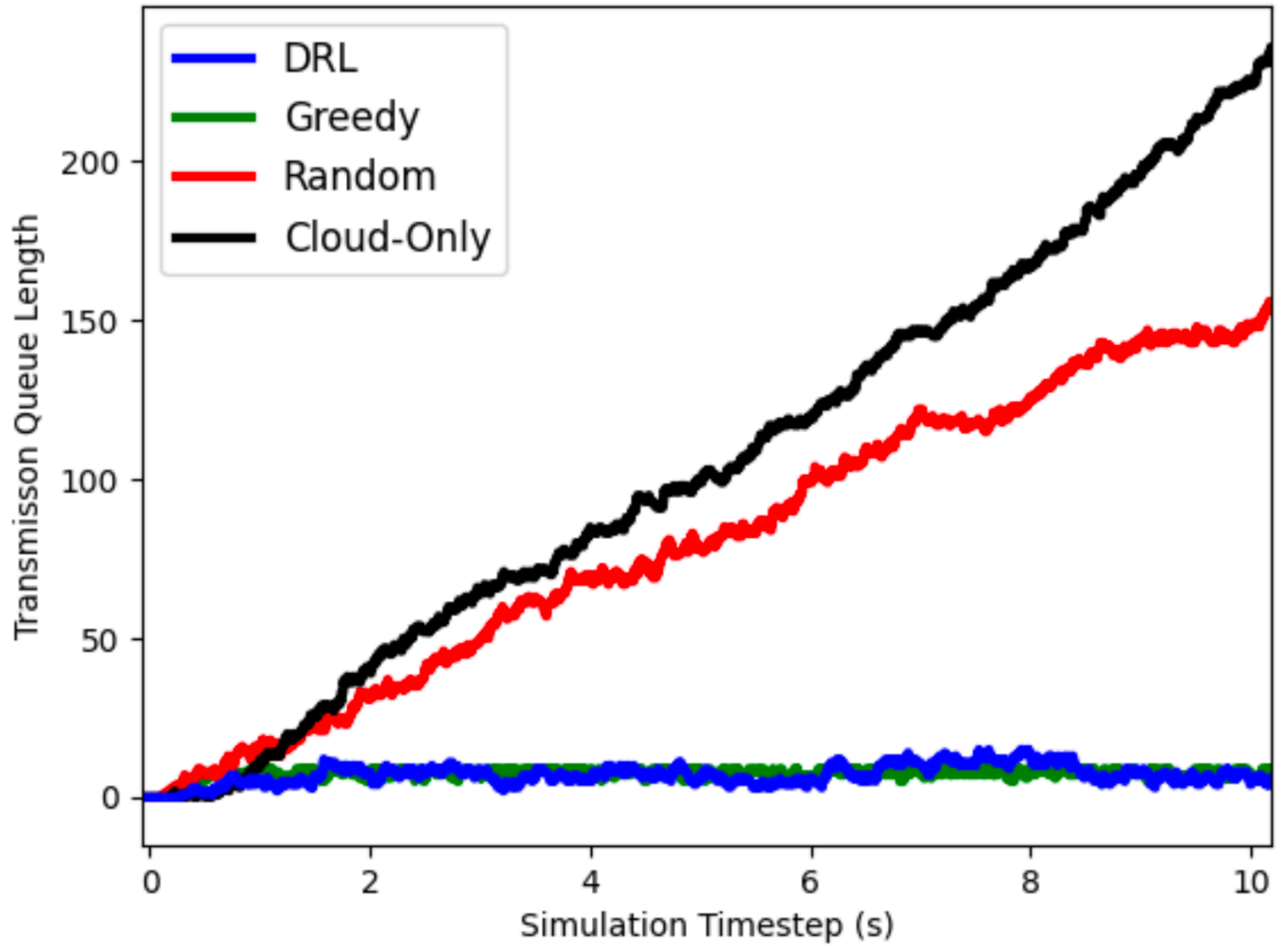


Fig. 6: Total length of all transmission queues throughout the first 10 seconds of simulation (Scenario 3).

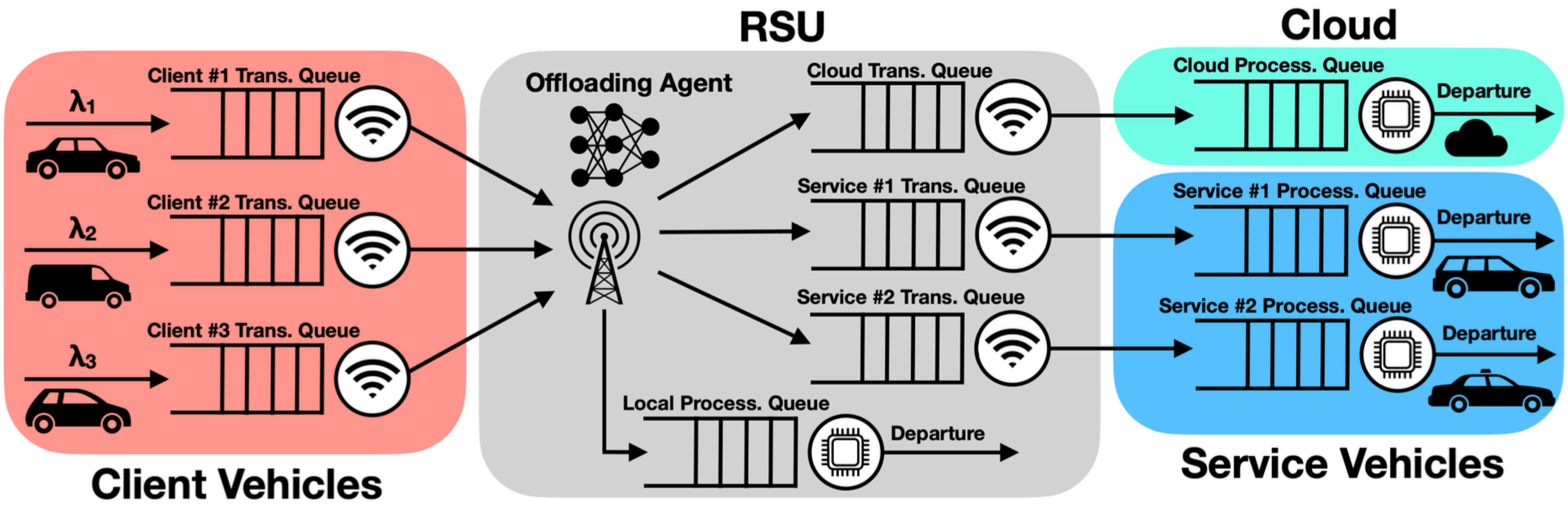
Traffic Intensity = (aL)/R

For cloud only: (3×20)×30/1000 = 1.8 > 1

Extra Slide #3

Delay & Communication model

- The total delay experienced by a task is the sum of the following: $d_{\text{total}} = d_{\text{client}} + d_{\text{rsu}} + d_{\text{service}}$



$$d_{\text{w-trans}} = \frac{CU_x}{TR_w}$$

$$TR = B.\log_2(1 + SNR)$$

$$SNR = \frac{P_R}{N_0.W}$$

$$P_R = P_T.G_R.G_T.(\frac{\lambda}{4\pi d})^2$$

Extra Slide #4
VFC Motivation

- Estimated 1.5 billion cars globally
- This number is projected to reach 2 billion by 2040
- Assume an average of 10 GFlops per vehicle
- 1.5 billion cars * 10GFlops = 1500 Peta Flops
- The most powerful supercomputer system of 2024 has 1195 Peta Flops