Energy-Efficient 3D Vehicular Crowdsourcing for Disaster Response by Distributed Deep Reinforcement Learning

Professor Chi (Harold) Liu

Fellow of IET, British Computer Society, and Royal Society of Arts

Vice Dean, School of Computer Science and Technology

Beijing Institute of Technology

August 20, 2021

Co-authors: Hao Wang, Zipeng Dai, Jian Tang, Guoren Wang

Winner of ACM SIGKDD'21 Best Paper Runner-up Award

Outline

- Background & Problem Formulation
- Challenges
- Preliminaries
- Our Solution: DRL-DisasterVC(3D)
- Simulator Design
- Experimental Results
- Conclusion

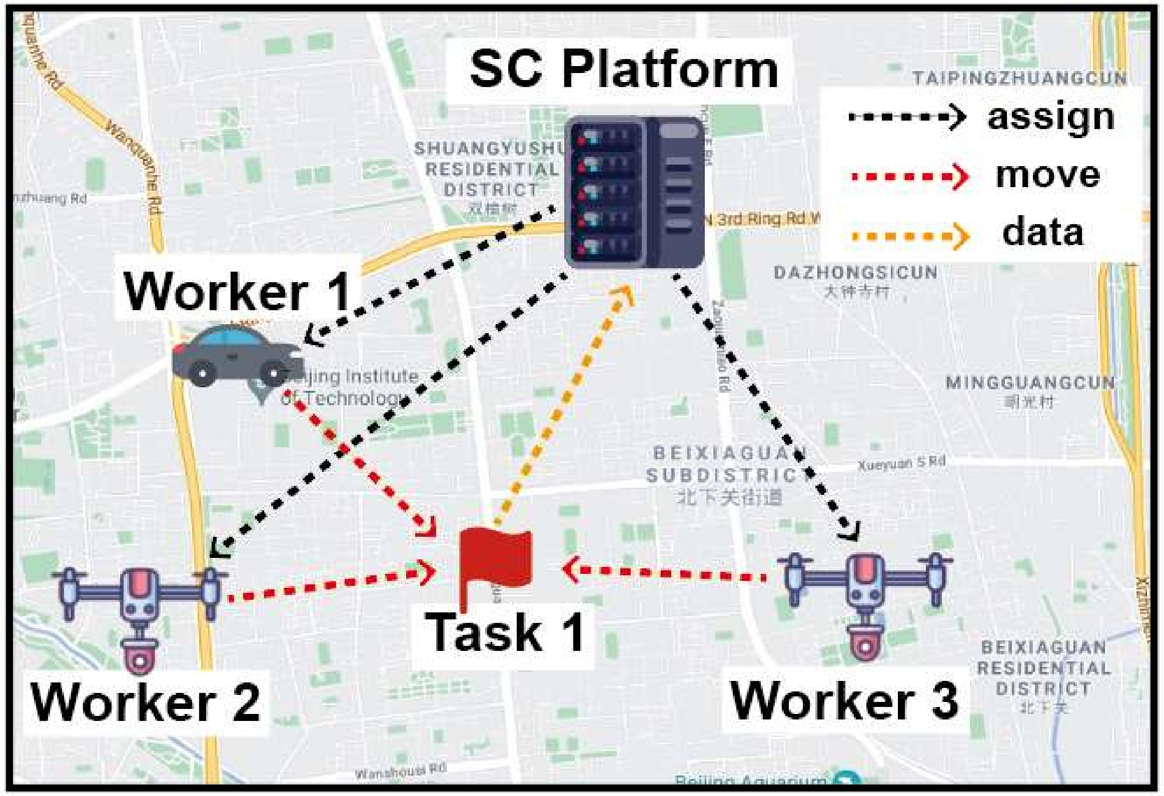
Outline

- Background & Problem Formulation
- Challenges
- Preliminaries
- Our Solution: DRL-DisasterVC(3D)
- Simulator Design
- Experimental Results
- Conclusion

Background & Problem Formulation

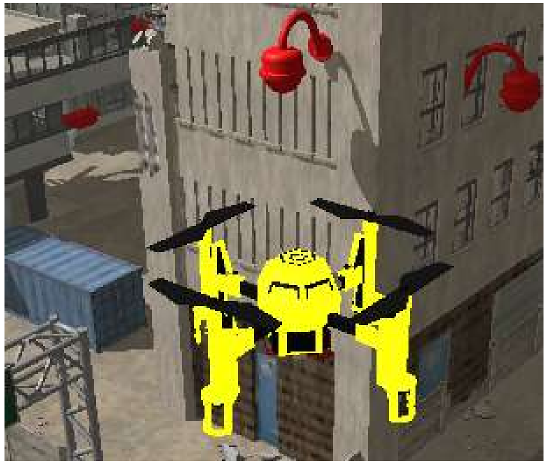
■ Spatial Crowdsourcing (SC)

- Tasks with spatiotemporal constraints (e.g., deadlines) are submitted to the platform, which performs task assignment to suitable workers.
- To complete a task, workers (usually with limited energy supply) physically move to the assigned position and submit their collected spatiotemporal data to the platform.

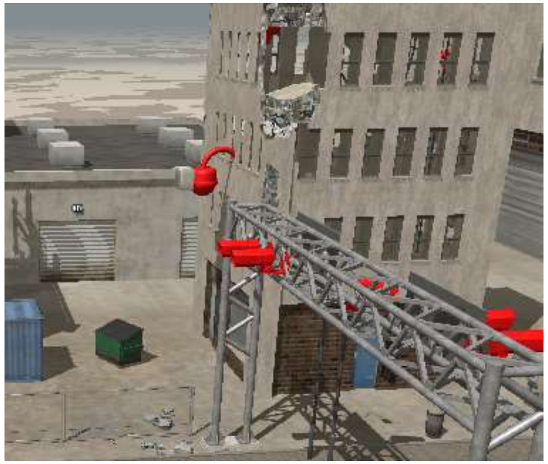


Background & Problem Formulation

- Application: unmanned vehicles-assisted disaster response



Unmanned Vehicles (drone)



PoIs (Surv. Cameras)

Obstacles (collapsed building)

- Unmanned Vehicles (UVs) equipped with multiple sensors and receivers can be quickly deployed over disaster workzone, to provide rapid situational awareness by collecting environmental and life data from point-of-interests (PoIs).
- We explicitly consider the problem of routing multiple UVs for disaster response.

Background & Problem Formulation

■ Formulation as a Markov Decision Process (MDP)

Mathematically, the optimization problem is formulated as:

$$\boldsymbol{P1}: \max_{\{\vartheta_t^u, l_t^u\}} \xi$$

$$s.t. \sum_{t=1}^{T} e_t^u \leq e_0, \forall u \in \mathcal{U}$$

- ➢ Maximize the energy efficiency ξ
- ➢ Limited energy consumption of UVs during movement and data collection

To solve this problem, we then formulate P1 as a MDP $(\mathcal{S}, \mathcal{A}, R, \Omega, \gamma)$. In each timeslot t:

- **State** $(\boldsymbol{s}_t \in \mathcal{S})$ is all task information, including: (1) all UVs' current location and remaining energy (2) all PoIs' remaining data (3) obstacles locations.
- **Action** $(\boldsymbol{a}_t \in \mathcal{A})$ includes movement directions $\boldsymbol{\vartheta}_t^u$ and traveling distance l_t^u of each UV.
- **Reward** is calculated by:

$$r_t = \left(\frac{1}{U} \sum_{u=1}^{U} \frac{d_t^u}{e_t^u} \cdot \left(1 - \frac{d_{t-}^u}{d_t^u + d_{t-}^u} \right) \right) \cdot \kappa_t - \varrho_t$$

where d_t^u and d_{t-}^u is the amount of collected and dropped data by UV u at timeslot t, respectively. κ_t is time-varying geographical fairness index. $\varrho_t$denotes the penalty applied to UVs.

Outline

- Background & Problem Formulation
- Challenges
- Preliminaries
- Our Solution: DRL-DisasterVC(3D)
- Simulator Design
- Experimental Results
- Conclusion

Challenges

- Optimize multiple metric simultaneously in a complex scenario

Without loss of generality, the successfully uploaded data depends on:

- Data collection time $\tau_{t,c}^{u} = \tau - \tau_{t,m}^{u}$
- Do not crash into obstacles

UVs movement

- Transmission rate $v_t^{u,p}$
- Number of serviced PoIs $|\overline{\mathcal{P}_t^u}|$

UVs location

- SNR threshold snr_0

Scene noise

Maximize:

Data collection ratio ζ

Geographical fairness κ

Energy efficiency ξ

Minimize:

Data dropout ratio σ

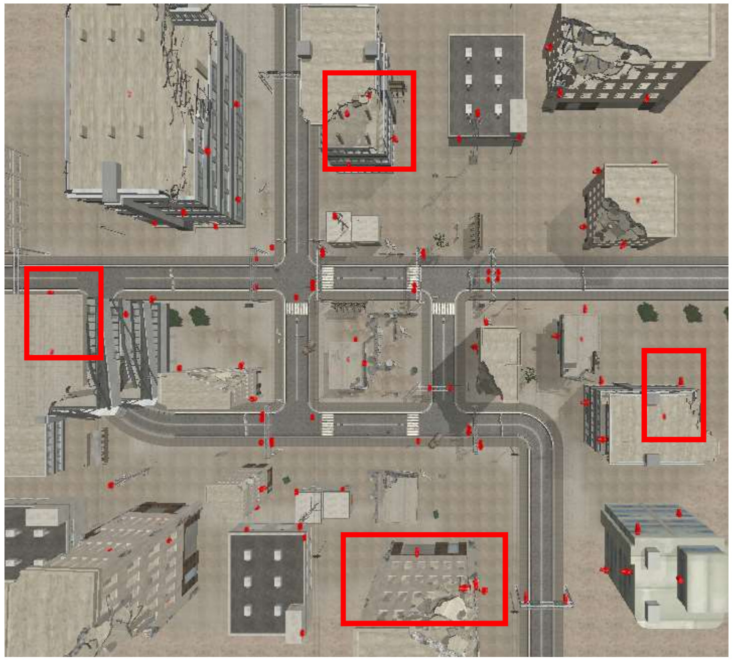
It is quite difficult to derive an optimal long-term policy for UVs scheduling by fully considering spatiotemporal data complexity and correlation.

Challenges

- Trade-off between environment exploration and energy consumption

Geographical fairness $$\kappa = \frac{\left(\sum_{p=1}^{P} \frac{d_0^p - d_t^p}{d_0^p}\right)^2}{P \sum_{p=1}^{P} \left(\frac{d_0^p - d_t^p}{d_0^p}\right)^2}$$

Energy efficiency $$\xi = \frac{\zeta \cdot \sum_{p=1}^{P} d_0^p}{\boxed{e_T}} \cdot (1-\sigma) \cdot \kappa$$



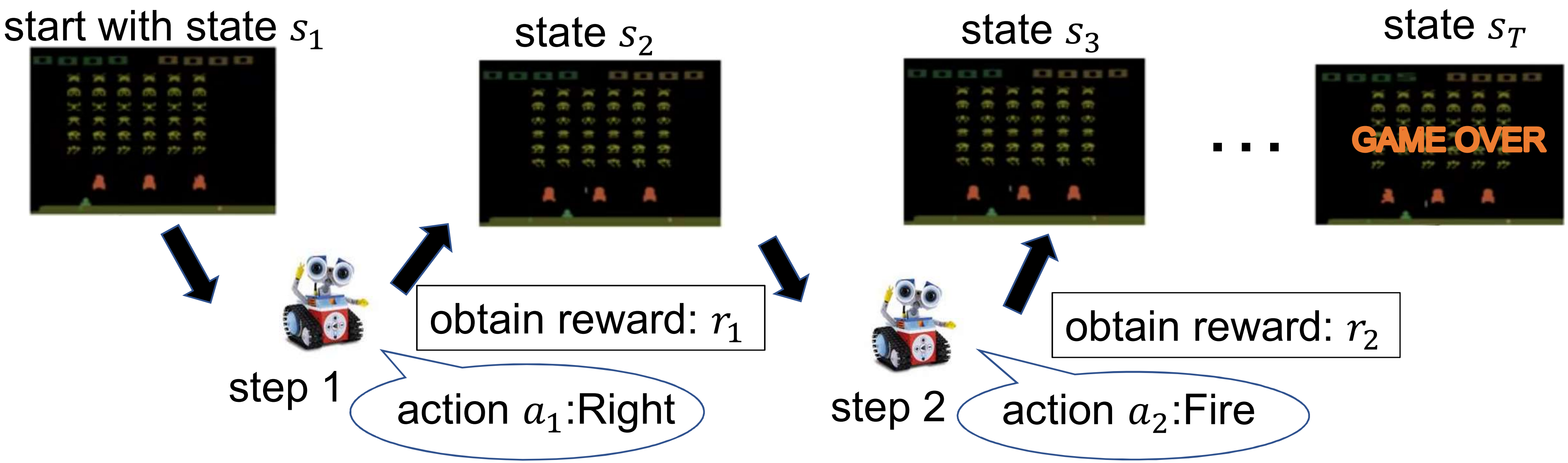
- Lack of exploration results in the failure of collecting enough data.
- Some PoIs are far-off which are hard to visit.
- Finding a trade-off between environmental exploration and energy consumption is non-trivial.

Outline

- Background & Problem Formulation
- Challenges
- Preliminaries
- Our Solution: DRL-DisasterVC(3D)
- Simulator Design
- Experimental Results
- Conclusion

Preliminaries

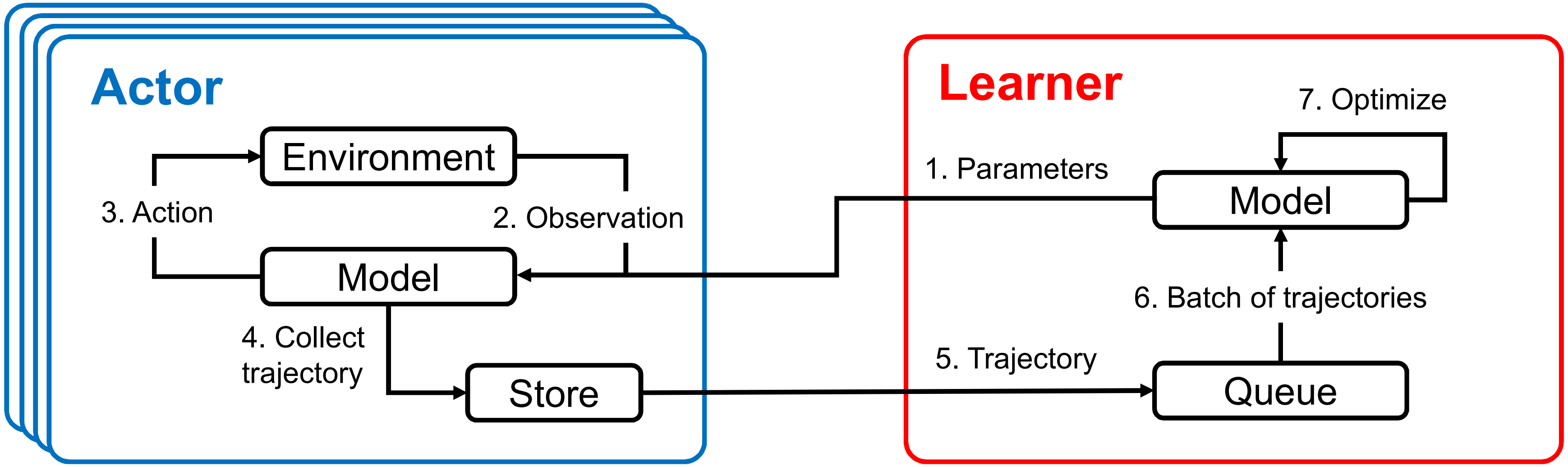
- Deep reinforcement learning (DRL)



Reinforcement learning (RL) is to learn a state-action mapping to maximize the a numerical accumulated reward signal.

Preliminaries

- IMPALA: Scalable Distributed Deep-RL with Importance Weighted Actor-Learner Architectures, ICML 2018 by Google



Multi-actor-one-learner architecture increases sample throughput, but sample efficiency drops significantly.

Lasse Espeholt, Hubert Soyer, Rémi Munos, Karen Simonyan, Volodymyr Mnih,et al. 2018. IMPALA: Scalable Distributed Deep-RL with Importance Weighted Actor-Learner Architectures. In *ICML'18*, Vol. 80. 1406–1415.

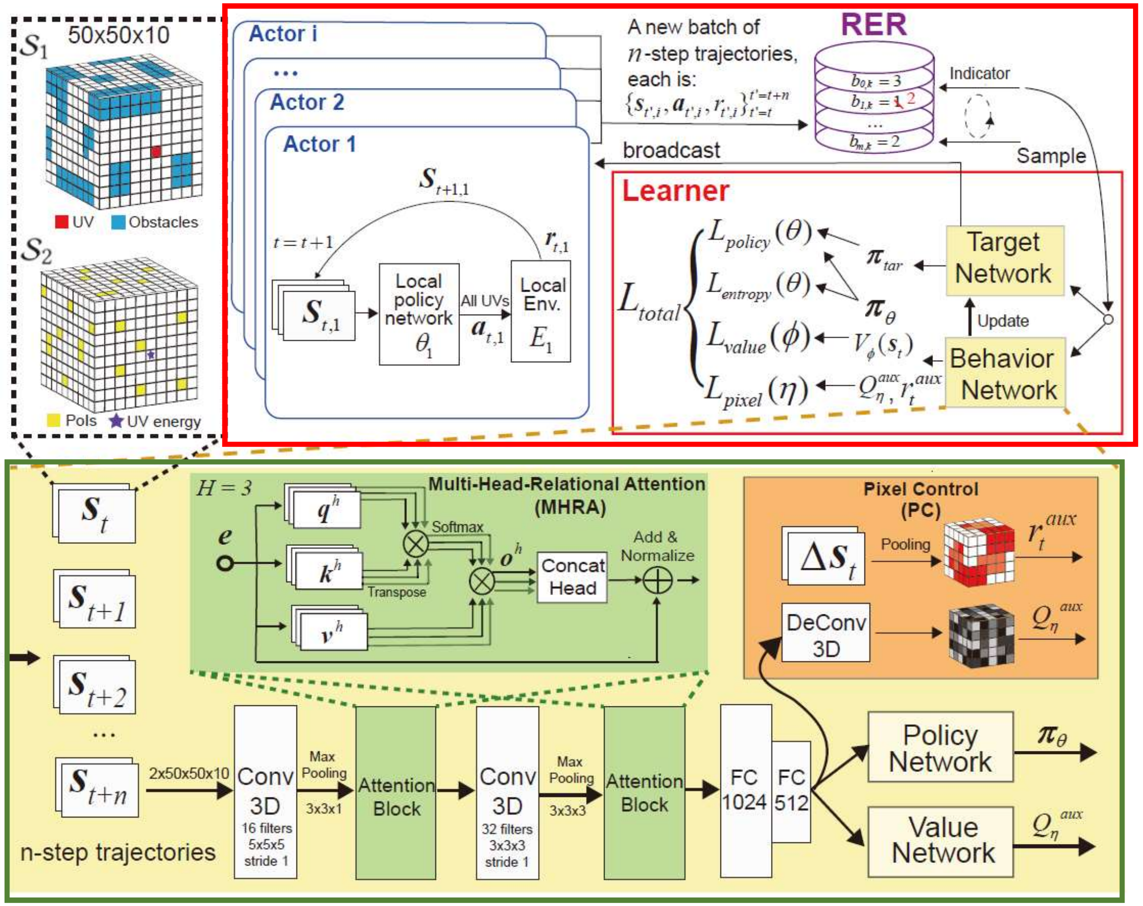
Outline

- Background & Problem Formulation
- Challenges
- Preliminaries
- Our Solution: DRL-DisasterVC(3D)
- Simulator Design
- Experimental Results
- Conclusion

Our Solution: DRL-DisasterVC(3D)

- ➢ Distributed DRL Framework with a repetitive experience replay (RER) for Multi-UV Planning in Disaster Response**DRL-DisasterVC(3D)**

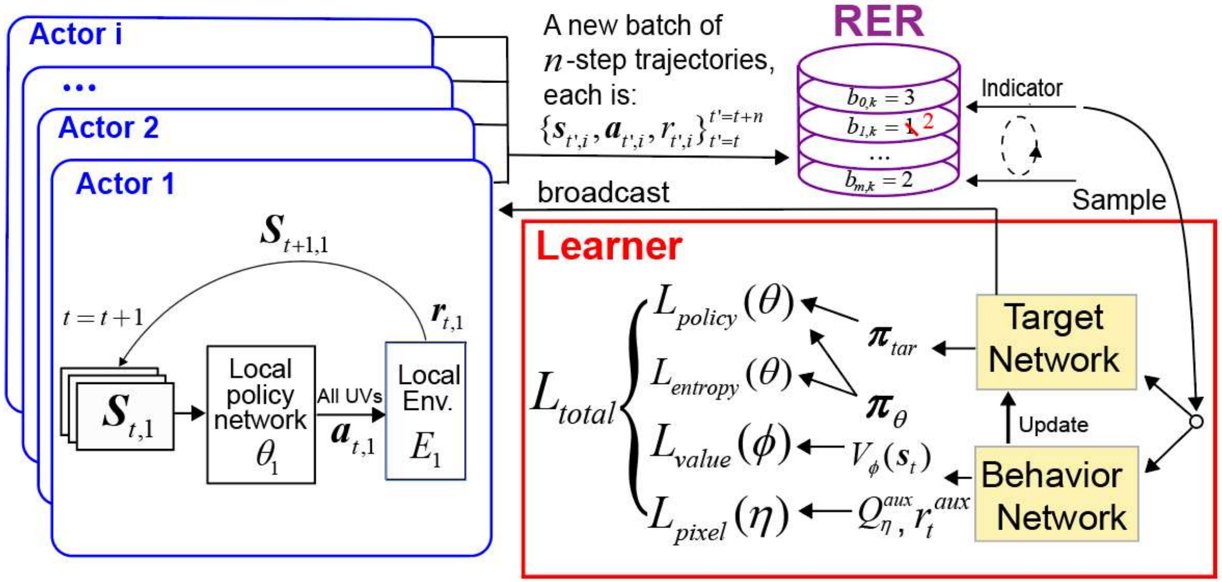
- ➢ Attentive 3D CNN Convolution with Pixel Control for Spatial Exploration



Our Solution: DRL-DisasterVC(3D)

■ Repetitive Experience Replay (RER) and Target network

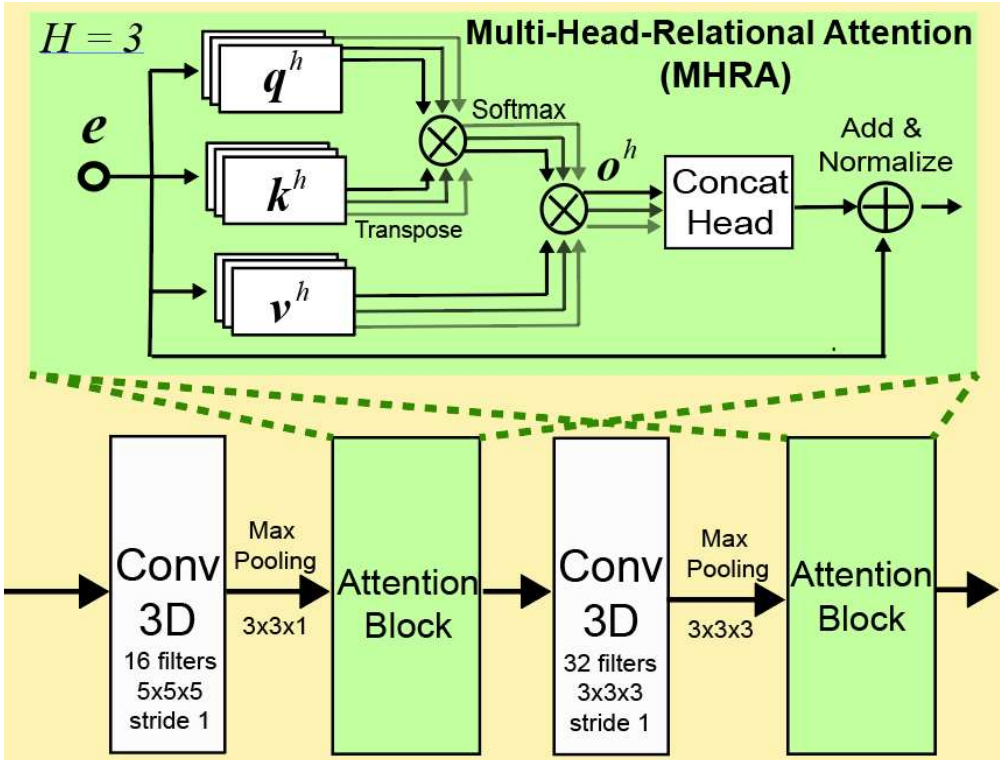
- To better utilize previous experiences for multiple UVs ⟶ Repetitive Experience Replay
- To stabilize the distributed training process ⟶ Clipped Target network



- Action space $\mathcal{A}$ in our multi-UV scenario expands exponentially in dimensions, which enlarges the difference among $\boldsymbol{\pi}_{\mathrm{act_i}}$.
- Limit the policy update speed by using truncated importance sampling $\min(\frac{\pi_{act_i}}{\pi_{tar}}, \rho)\frac{\pi_\theta}{\pi_{act_i}}$
- The agent learns fast when setting ρ a high value at the cost of training instability.

Our Solution: DRL-DisasterVC(3D)

■ Multi-Head-Relational Attention (MHRA) for Spatial Modeling



- Better extracting these relationships helps UVs learn more reasonable trajectories, by adding a MHRA module between every two 3D CNN layers.

$$\text{Query} : \quad \boldsymbol{q}^h = f_q(e)$$
$$\text{Key} : \quad \boldsymbol{k}^h = f_k(e)$$
$$\text{Value} : \quad \boldsymbol{v}^h = f_v(e)$$

- H independent attention heads indicate the different relational semantics.

$$\mathcal{O} = softmax\left(\frac{\mathcal{Q}\mathcal{K}^{\top}}{\sqrt{g}}\right)\mathcal{V}$$

Zambaldi, V., David Raposo, A. Santoro, V. Bapst, Yujia Li, et al. "Deep reinforcement learning with relational inductive biases." In *ICLR'19*.

Our Solution: DRL-DisasterVC(3D)

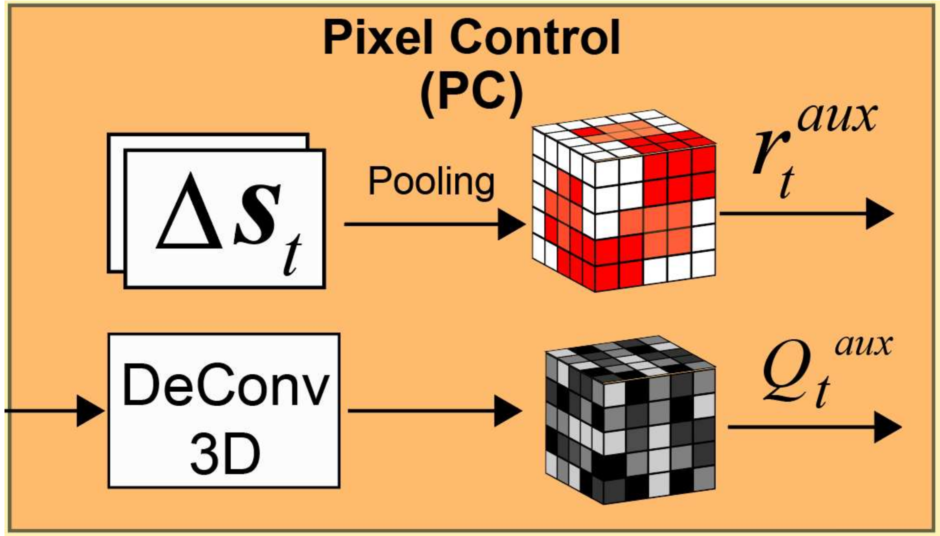
- Auxiliary Pixel Control (PC) for Spatial Exploration

$$L_{pixel}(\eta) = \mathrm{E}[(\boxed{y_t^{aux}} - \boxed{Q_t^{aux}(s_t, a_t, \eta))^2}]$$

Expected pixel change

$$\boxed{y_t^{aux} = \sum_{k=1}^{n} \gamma^k r_{t+k}^{aux} + \gamma^n \max_{a'} Q_t^{aux}(s_{t+n}, a_{t+n}, \eta')}$$

Real pixel change



- Pixel difference as the "intrinsic reward".
- r_t^{aux} is calculated by the average absolute pixel difference of adjacent input state.
- Q_t^{aux} is a 3D spatial grid of action values from 3D deconvolutional network.

Max Jaderberg, Volodymyr Mnih,Wojciech Marian Czarnecki, Tom Schaul, Joel Z. Leibo, et al. Reinforcement Learning with Unsupervised Auxiliary Tasks. In *ICLR'17*.

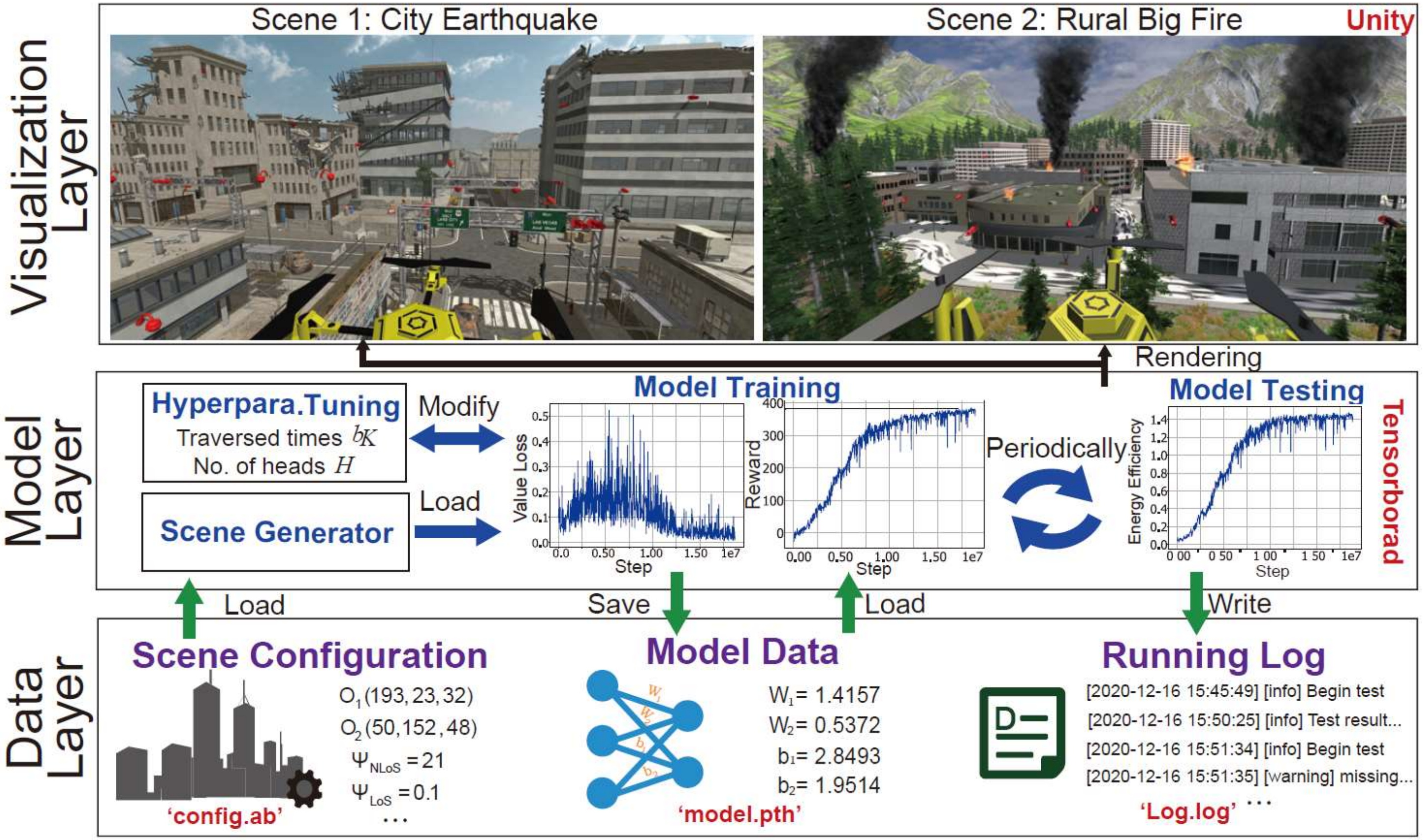
Outline

- Background & Problem Formulation
- Challenges
- Preliminaries
- Our Solution: DRL-DisasterVC(3D)
- Simulator Design
- Experimental Results
- Conclusion

Simulator Design

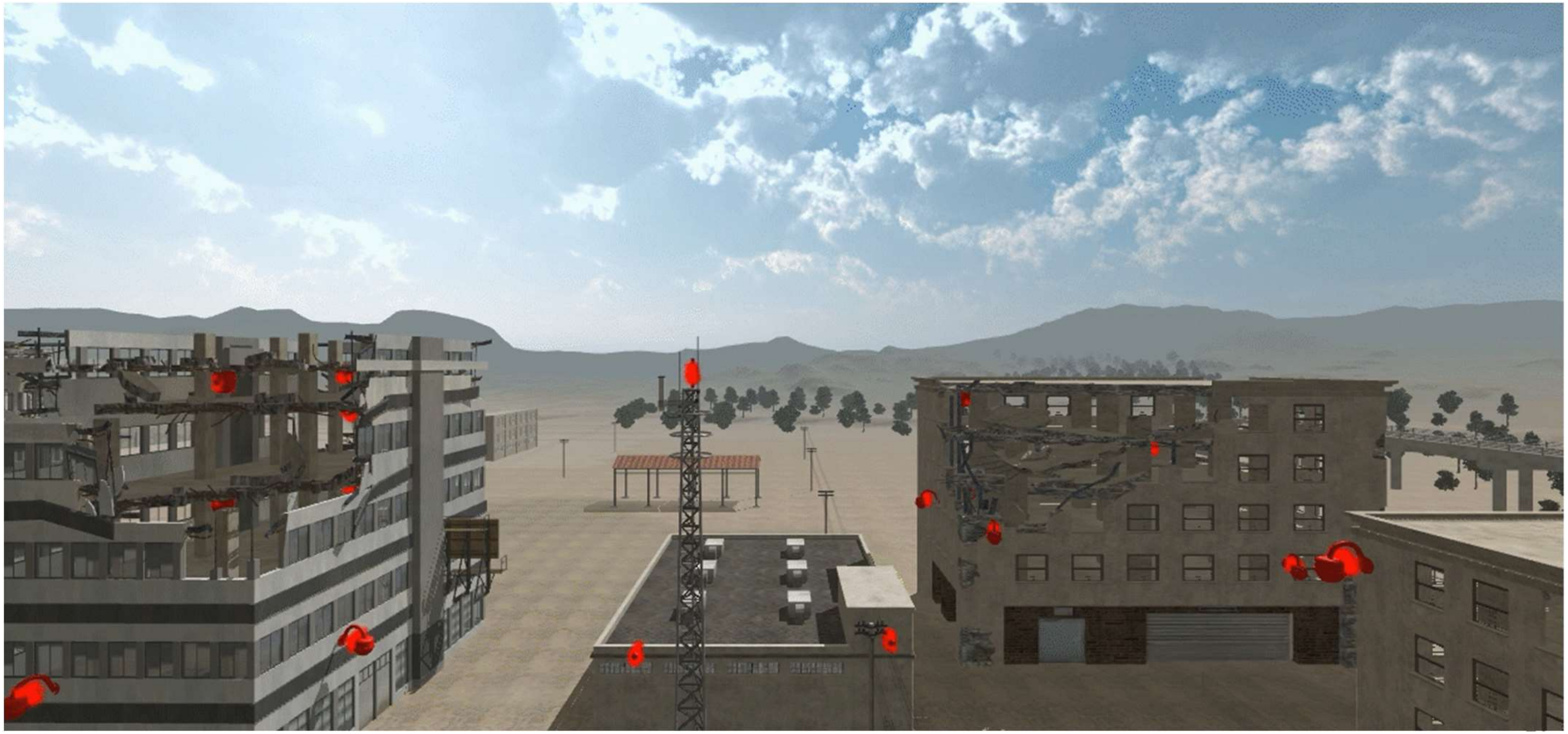
■ DisasterSim

➢ To bridge model training, testing and visualization for multi-UV trajectory planning.



Simulator Design

- **DisasterSim**



Outline

- Background & Problem Formulation
- Challenges
- Preliminaries
- Our Solution: DRL-DisasterVC(3D)
- Simulator Design
- **Experimental Results**
- Conclusion

Experimental Results

- DNN Hyperparameters Tuning

		$H=1$	$H=2$	$H=4$	$H=8$	$H=16$
$b_K=1$	ζ	0.780	0.843	0.840	0.836	0.840
	σ	0.140	0.127	0.131	0.133	0.135
	κ	0.814	0.873	0.877	0.866	0.852
	ξ	1.117	1.275	1.309	1.186	1.201
$b_K=2$	ζ	0.821	0.905	0.913	0.879	0.855
	σ	0.127	0.119	0.117	0.124	0.127
	κ	0.852	0.921	0.934	0.912	0.879
	ξ	1.191	1.402	1.388	1.262	1.193
$b_K=3$	ζ	0.850	0.890	**0.920**	0.874	0.808
	σ	0.122	0.120	**0.109**	0.129	0.142
	κ	0.862	0.927	**0.943**	0.892	0.840
	ξ	1.238	1.358	**1.437**	1.235	1.135
$b_K=4$	ζ	0.843	0.864	0.874	0.830	0.797
	σ	0.129	0.123	0.119	0.134	0.158
	κ	0.860	0.894	0.905	0.871	0.818
	ξ	1.150	1.316	1.317	1.181	1.072

We find that 4 heads in MHRA with 3 traversed times in RER give the best performance in terms of energy efficiency ξ.

Experimental Results

- Ablation Study

	ζ	σ	κ	ξ
DRL-DisasterVC(3D)	**0.921**	**0.108**	**0.945**	**1.440**
- w/o PC	0.876	0.114	0.906	1.355
- w/o MHRA	0.898	0.119	0.919	1.304
- w/o PC, MHRA	0.842	0.133	0.867	1.227

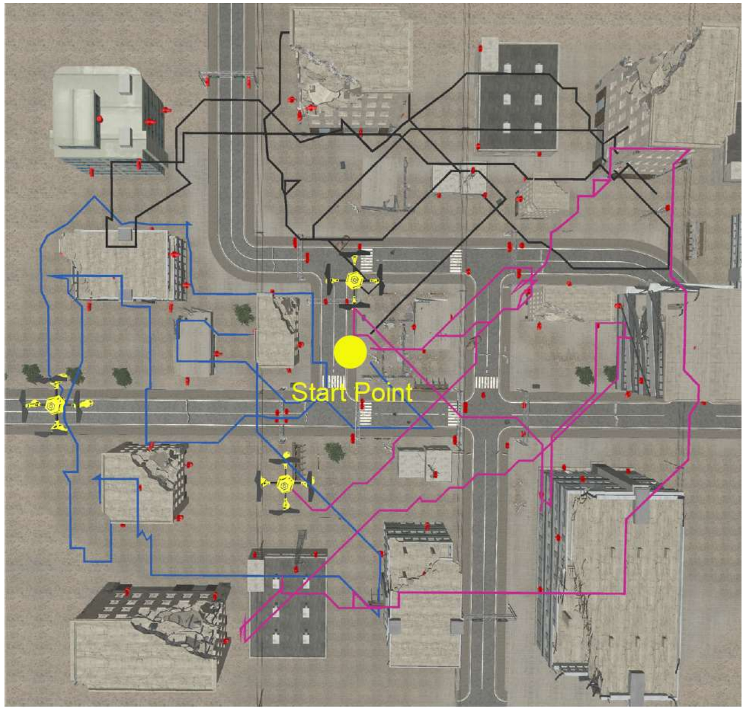
17.3%

- PC helps to achieve a better spatial exploration.
- But PC sacrifices some degree of efficiency to achieve a wider spatial exploration, and MHRA weakens this shortcoming.
- When removing both PC and MHRA, the complete version is 17.3% better which confirms the benefits of putting MHRA and PC together.

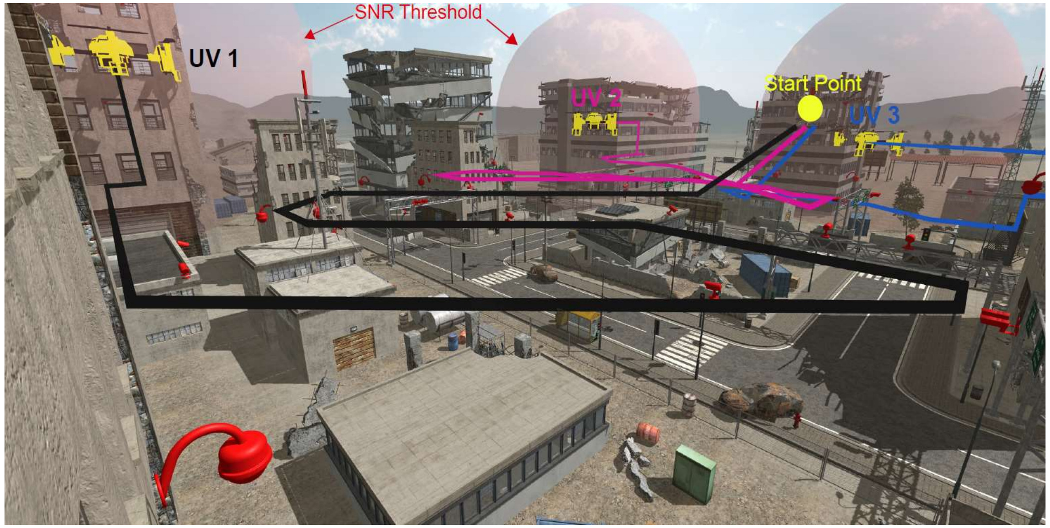
Experimental Results

■ Illustrative Data Collection Trajectories by 3 UVs

➢ UVs learn to collaborate by roughly dividing the workzone into 3 parts, and move around in its responsible one.



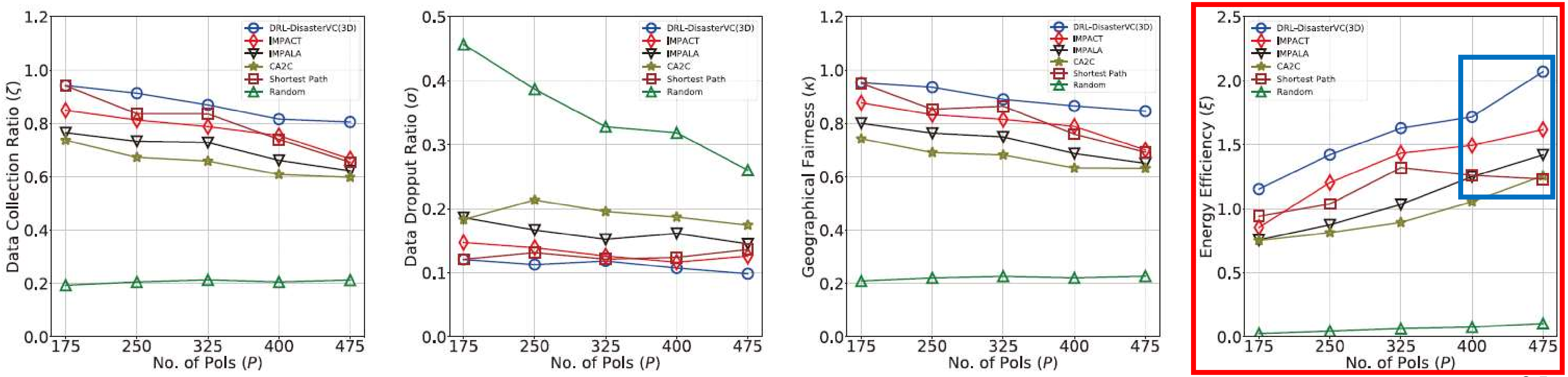
➢ Flying around the exterior of buildings, which helps achieve maximum tx rate.



Experimental Results

■ Impact of No. of PoIs (P)

- ➢ With more deployed PoIs, UVs can collect more data without moving far away and achieve higher energy efficiency.
- ➢ Data collection ratio ζ decrease significantly due to the lack of exploration.
- ➢ The gap of ξ between SP and other algorithms gets wider with the increase of P.



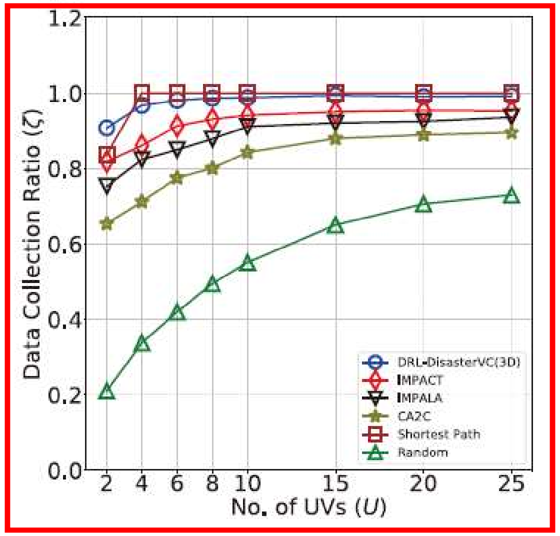
Experimental Results

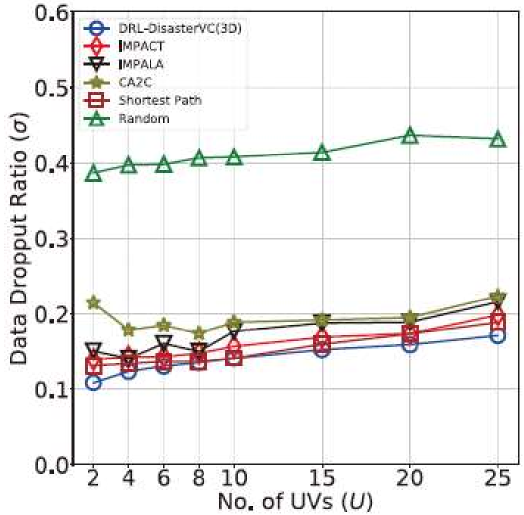
- Impact of No. of UVs (U)

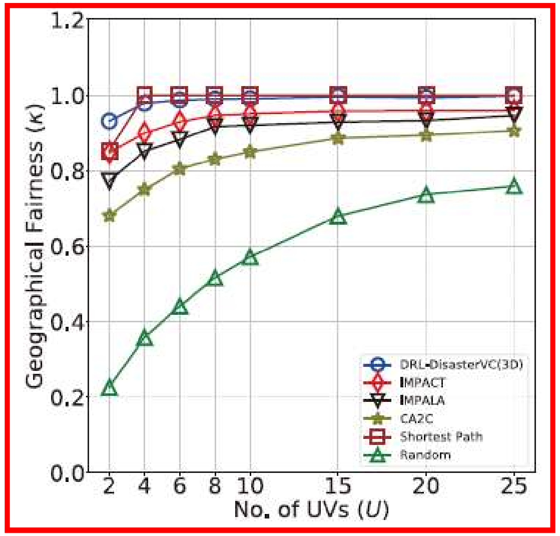
➢ More UVs could result higher data collection ratio and higher fairness.

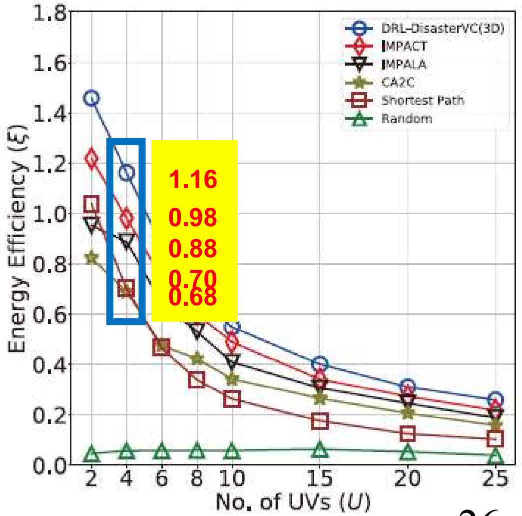
➢ Too many UVs (e.g., U = 25) will not bring further benefit.

➢ SP nearly collect all data when deploying 4 or more UVs but its energy efficiency only reaches 0.70 maximally. The energy consumption of DRL-DisasterVC(3D) and SP are 2455.82kJ and 4740.46kJ.



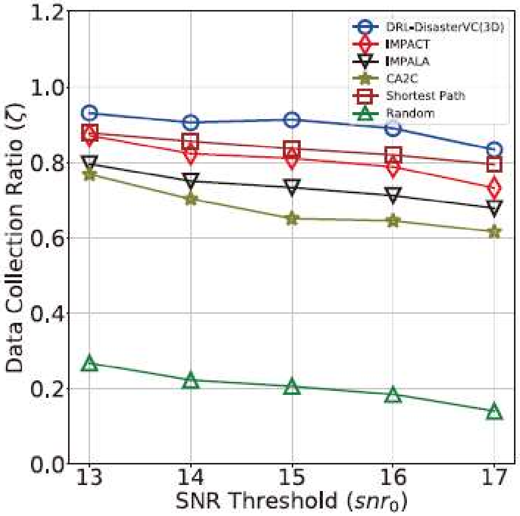


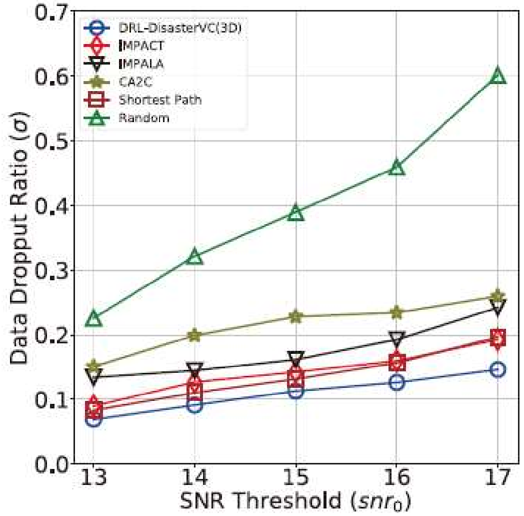


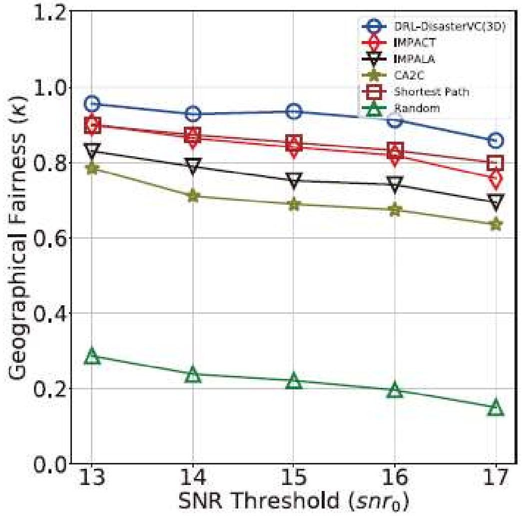


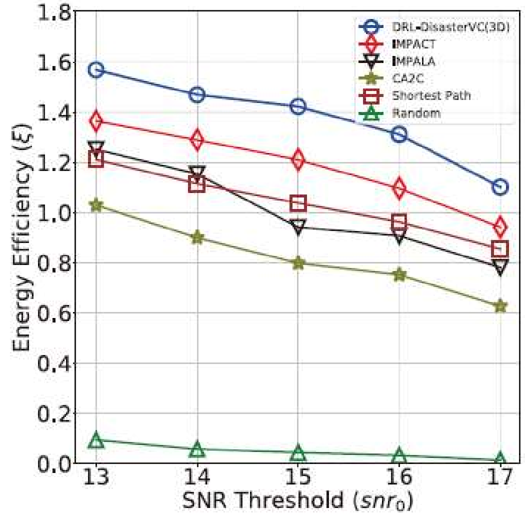
Experimental Results

- Impact of SNR threshold (snr_0)
  - High SNR threshold leads to the smaller amount of PoIs to successfully upload their data to a UV by the tx rate constraint.









Conclusion

- We consider a vehicular crowdsourcing problem of routing multiple UVs for disaster response.
- We propose “DRL-DisasterVC(3D)”, a distributed DRL framework for VC tasks in disaster response.
  - Distributed DRL framework with RER and clipped target network for learning efficiency and stability improvement
  - Attentive 3D CNN with pixel control for spatial exploration.
- We designed a novel disaster response simulator, called “DisasterSim”, to explicitly bridge model training, testing and visualization processes.
- We conduct extensive experiments and results verify the effectiveness of DRL-DisasterVC(3D) when comparing with five baselines.

Thanks a lot！

Any Questions?