Trends in Multi-Agent Deep Reinforcement Learning for Distributed Computing

Prof. Joongheon Kim
Korea University, School of Electrical Engineering
Artificial Intelligence and Mobility Laboratory
https://joongheon.github.io
joongheon@korea.ac.kr

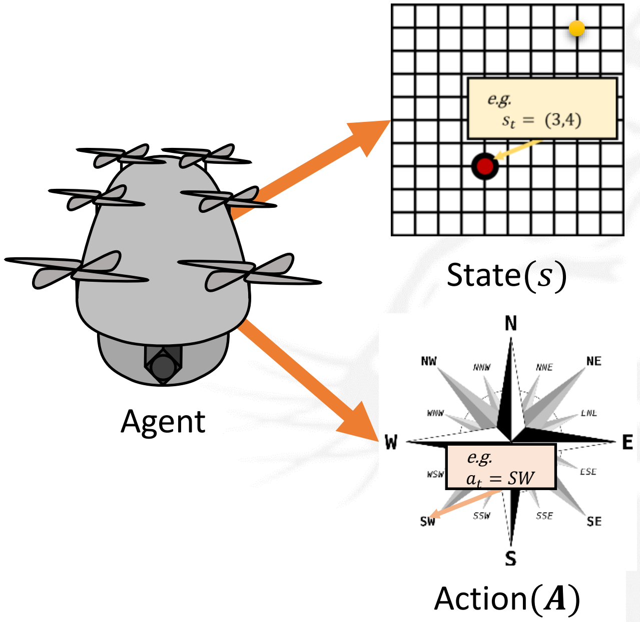
Research Collaborators

Ph.D. Students/Candidates	Faculty Collaborators	Related Projects
Won Joon Yun (Ph.D. Student at EE@KU) **MyungJae Shin** (Reseacher at SNUH) **Soyi Jung** (Ph.D. Candidate at ECE@Ajou)	**Prof. Jae-Hyun Kim** (ECE@Ajou) **Prof. Marco Levorato** (CS@UC-Irvine) **Prof. David Mohaisen** (CS@UCF)	**Hanyang-ITRC (5G/Unmanned Vehicle Research Center)** • [PI] Hanyang University • [WP2] Ajou University

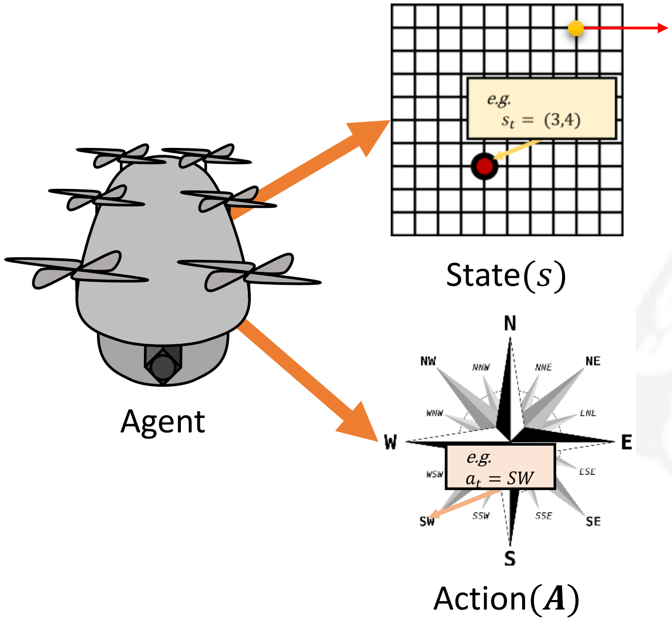
Contents

Contents

Review: Reinforcement Learning Mechanism



Review: Reinforcement Learning Mechanism

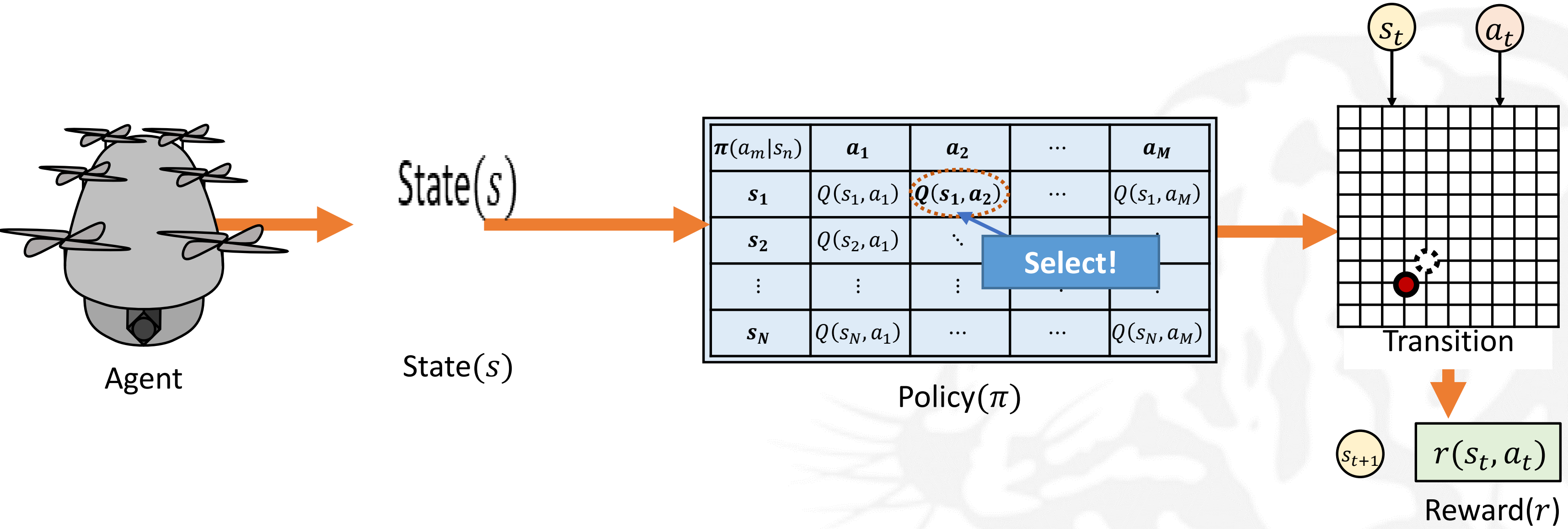


Objective: Arrive at the destination

What agent can do…

Move to 16 direction!

Review: Conventional Reinforcement Learning Mechanism



$\pi(a_m \mid s_n)$	a_1	a_2	$\cdots$	a_M
s_1	$Q(s_1, a_1)$	$Q(s_1, a_2)$	$\cdots$	$Q(s_1, a_M)$
s_2	$Q(s_2, a_1)$	$\ddots$		
$\vdots$	$\vdots$	$\vdots$		
s_N	$Q(s_N, a_1)$	$\cdots$	$\cdots$	$Q(s_N, a_M)$

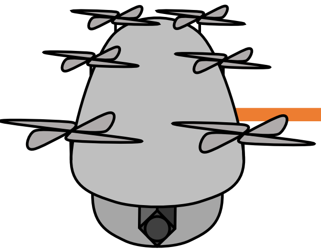
Iterate until the scenario terminated!

Trajectory(Dataset): $\tau = \{s_0, a_0, r_0, s_1, a_1, \ldots., s_T\}$

Objective Function: $J(\theta) = E_\tau[\sum_{t=0}^{T} \gamma^t \cdot r(s_t, a_t)]$ ← **Maximize!**

[1] Watkins., Q-Learning, *1989*

Review: Conventional Reinforcement Learning Mechanism



Agent

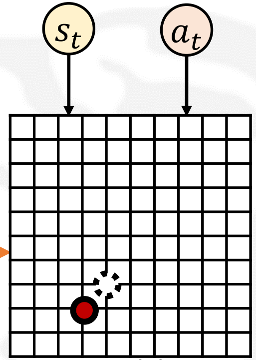
State(s)

State(s)

$\boldsymbol{\pi}(a_m\|s_n)$	$\boldsymbol{a_1}$	$\boldsymbol{a_2}$	$\cdots$	$\boldsymbol{a_M}$
$\boldsymbol{s_1}$	$Q(s_1, a_1)$	$\boldsymbol{Q(s_1, a_2)}$	$\cdots$	$Q(s_1, a_M)$
$\boldsymbol{s_2}$	$Q(s_2, a_1)$	$\ddots$		
$\vdots$	$\vdots$	$\vdots$		
$\boldsymbol{s_N}$	$Q(s_N, a_1)$	$\cdots$	$\cdots$	$Q(s_N, a_M)$

Select!

Policy(π)



Transition

s_{t+1} $r(s_t, a_t)$

Reward(r)

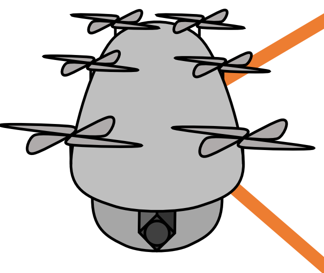
Iterate until the scenario terminated!

Trajectory(Dataset): $\tau = \{s_0, a_0, r_0, s_1, a_1, \ldots., s_T\}$

Objective Function: $J(\theta) = E_\tau[\sum_{t=0}^{T} \gamma^t \cdot r(s_t, a_t)]$ ← **Maximize!**

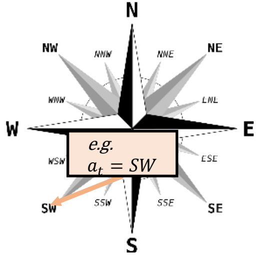
Review: Reinforcement Learning Mechanism

Artificial Intelligence and Mobility Lab



State(s)

State(s)

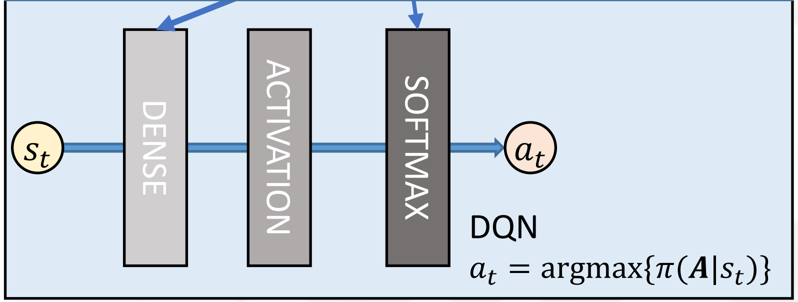


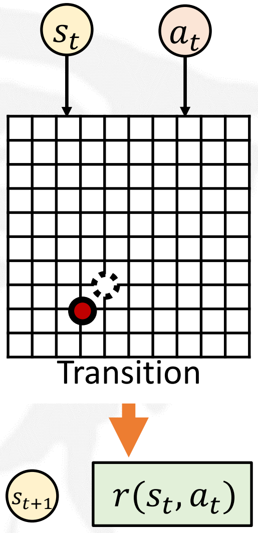
Action$(\boldsymbol{A})$

$\boldsymbol{\pi}(a_m \mid s_n)$	$\boldsymbol{a_1}$	$\boldsymbol{a_2}$	$\cdots$	$\boldsymbol{a_M}$
$\boldsymbol{s_1}$	$Q(s_1, a_1)$	$\boldsymbol{Q(s_1, a_2)}$	$\cdots$	$Q(s_1, a_M)$
$\boldsymbol{s_2}$	$Q(s_2, a_1)$	$\ddots$		
$\vdots$	$\vdots$	$\vdots$		
$\boldsymbol{s_N}$	$Q(s_N, a_1)$	$\cdots$	$\cdots$	$Q(s_N, a_M)$

Select!

Conventional Q-Table





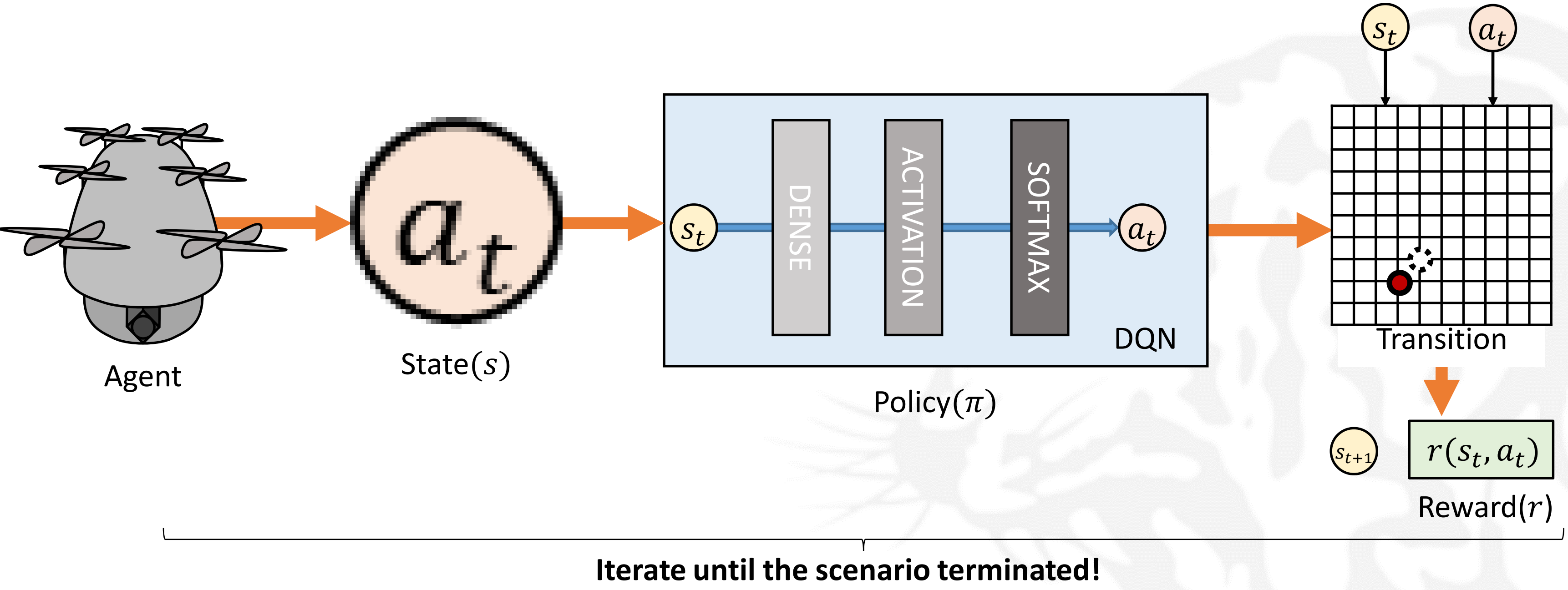
Policy(π)

Iterate until the scenario terminated!

Trajectory(Dataset): $\tau = \{s_0, a_0, r_0, s_1, a_1, \dots, s_T\}$

Objective Function: $J(\theta) = E_\tau[\sum_{t=0}^{T} \gamma^t \cdot r(s_t, a_t)]$ ← **Maximize!**

Review: Deep Reinforcement Learning Mechanism



Trajectory(Dataset): $\tau = \{s_0, a_0, r_0, s_1, a_1, \ldots, s_T\}$

Objective Function: $J(\theta) = E_\tau[\sum_{t=0}^{T} \gamma^t \cdot r(s_t, a_t)]$ ← **Maximize!**

[2] V. Mnih., Playing Atari with Deep Reinforcement Learning, *NIPS 2013*

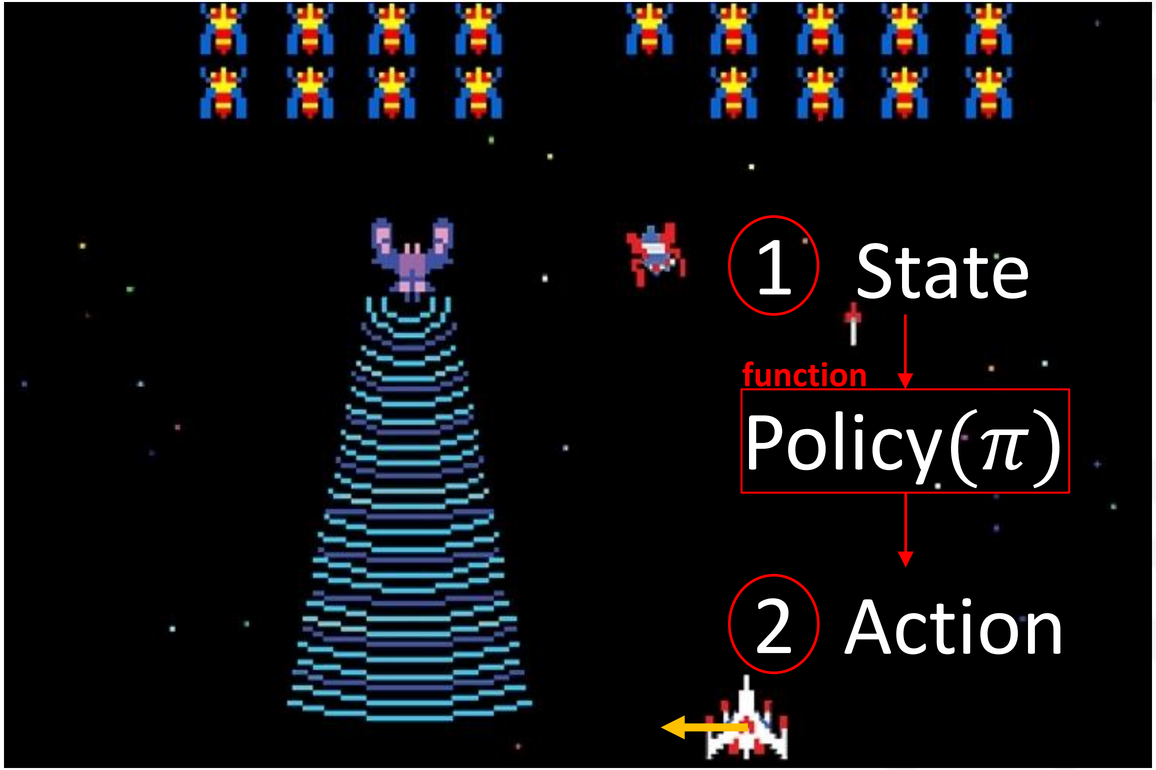
Contents

Basic Concept of DRL

Policy-based MADRL

Applications of MADRL

Review: Single Agent Reinforcement Learning

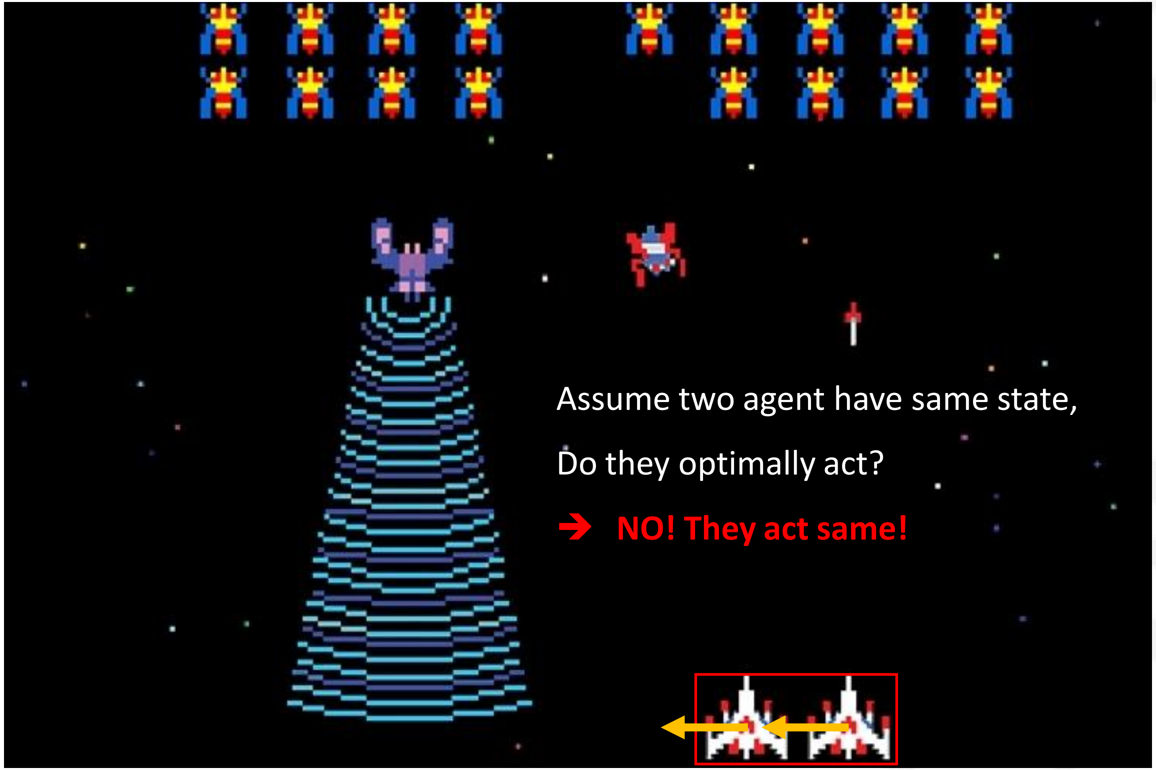


What if there exists more than one agent?

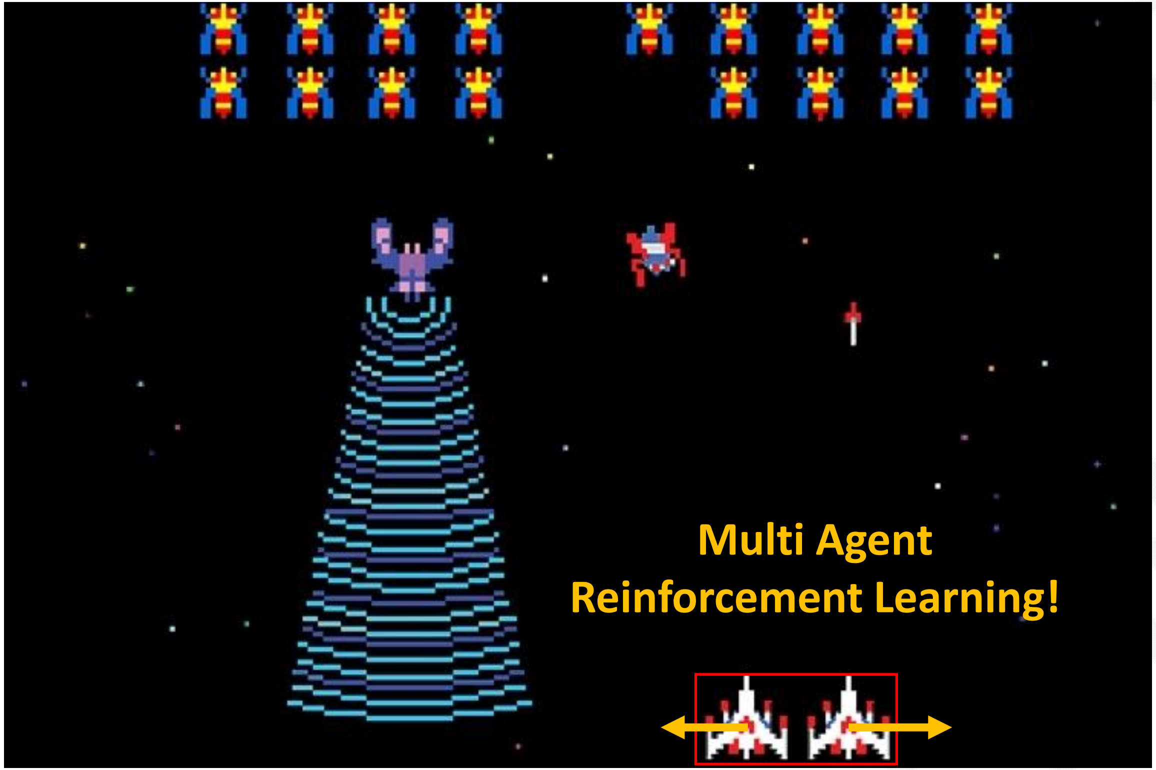
Assume two agents have same state,

Do they optimally act?

What if there exists more than one agent?

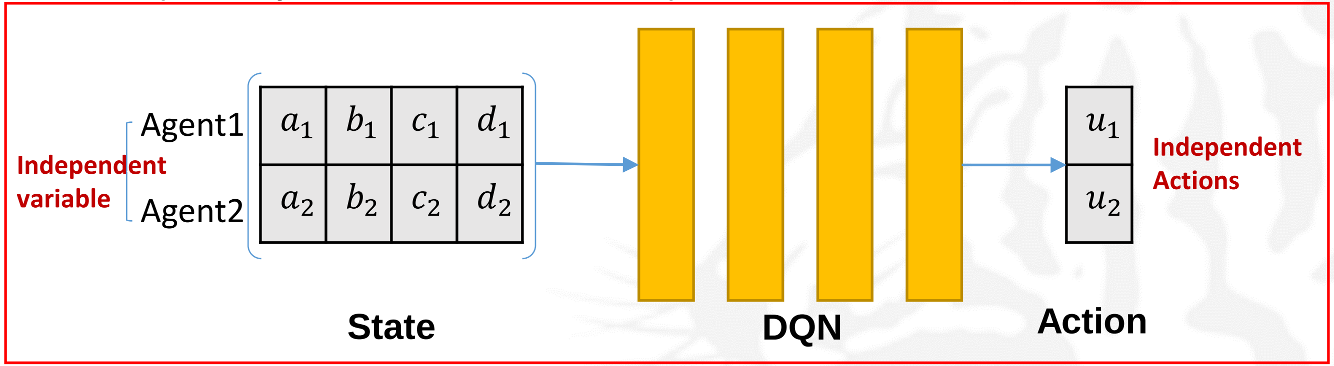


What we are going to learn is …

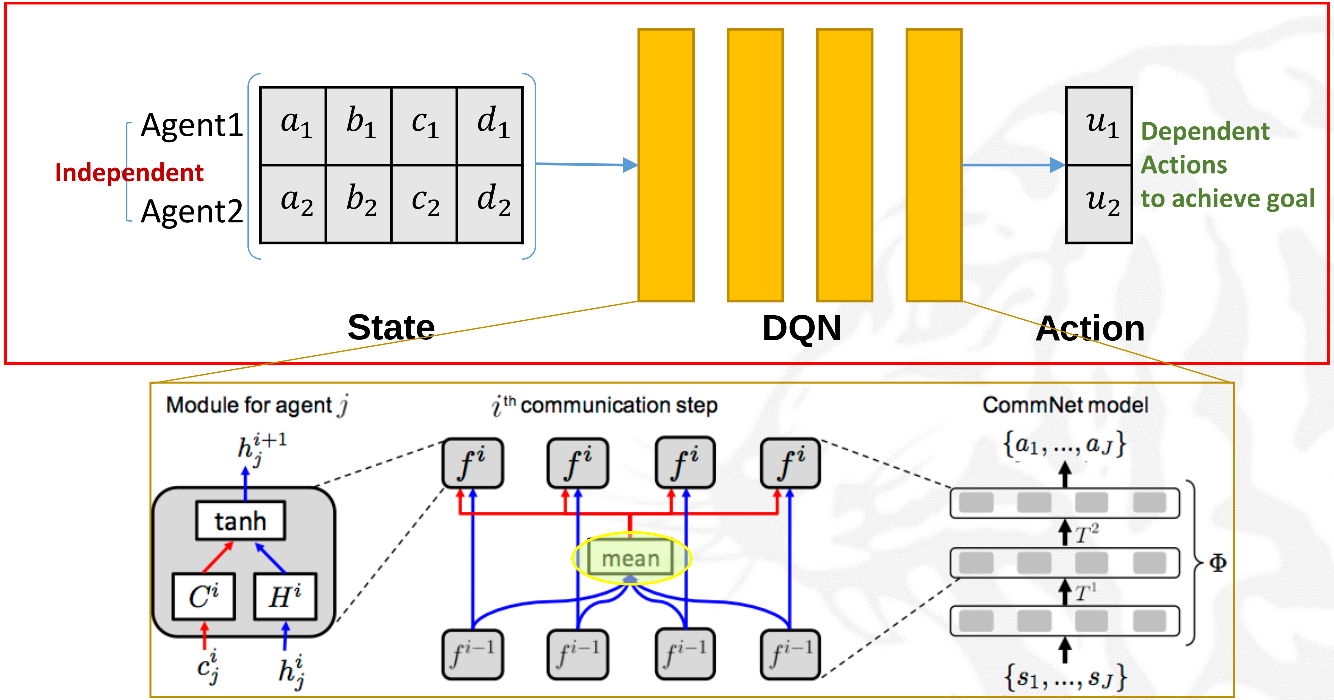


With Previous method.

With DQN(Partially Observable Environment)



DQN-based CommNet

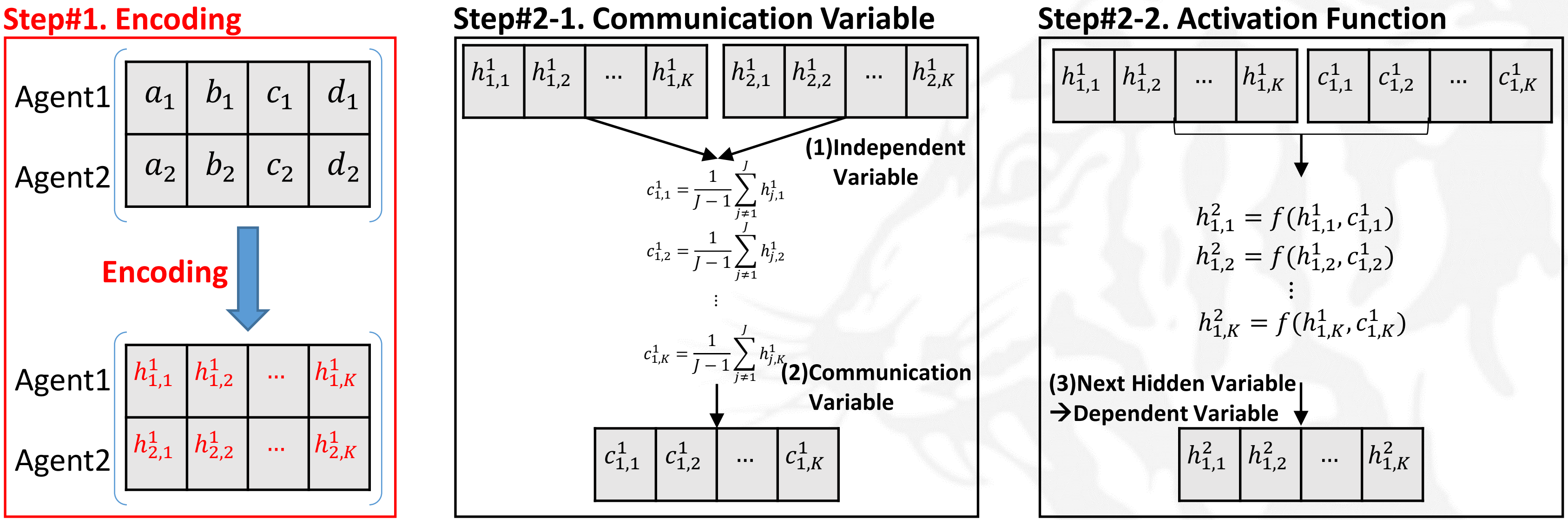


h_j^i : j-th agent's hidden state variable in i-th layer

c_j^i : j-th agent's communitive state variable in i-th layer

$$h_j^{i+1} = f^i(h_j^i, c_j^i)$$

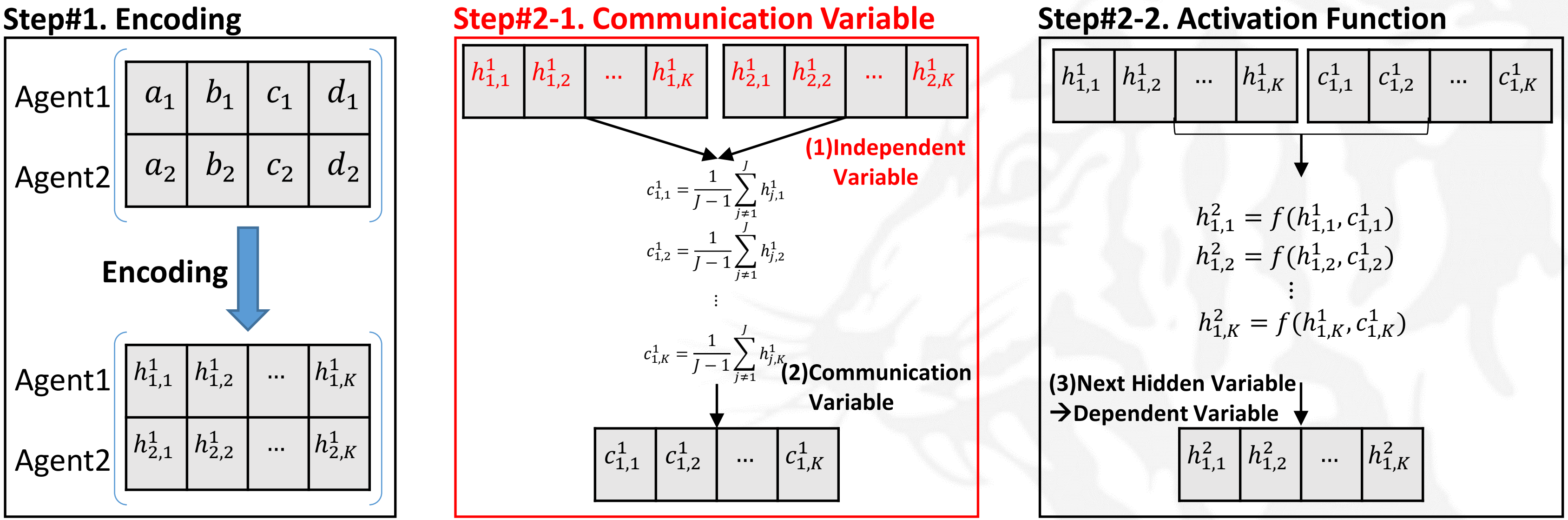
CommNet Mechanism



[3] S. Sukhbaatar et al., Learning Multiagent Communication with Backpropagation, *NIPS 2016*

CommNet Mechanism

Artificial Intelligence and Mobility Lab



CommNet Mechanism

Step#1. Encoding

Agent1	a_1	b_1	c_1	d_1
Agent2	a_2	b_2	c_2	d_2

Encoding

Agent1	$h^1_{1,1}$	$h^1_{1,2}$	…	$h^1_{1,K}$
Agent2	$h^1_{2,1}$	$h^1_{2,2}$	…	$h^1_{2,K}$

Step#2-1. Communication Variable

$h^1_{1,1}$	$h^1_{1,2}$	…	$h^1_{1,K}$	$h^1_{2,1}$	$h^1_{2,2}$	…	$h^1_{2,K}$

(1)Independent Variable

$$c^1_{1,1} = \frac{1}{J-1}\sum_{j\neq 1}^{J} h^1_{j,1}$$

$$c^1_{1,2} = \frac{1}{J-1}\sum_{j\neq 1}^{J} h^1_{j,2}$$

$$\vdots$$

$$c^1_{1,K} = \frac{1}{J-1}\sum_{j\neq 1}^{J} h^1_{j,K}$$

(2)Communication Variable

$c^1_{1,1}$	$c^1_{1,2}$	…	$c^1_{1,K}$

Step#2-2. Activation Function

$h^1_{1,1}$	$h^1_{1,2}$	…	$h^1_{1,K}$	$c^1_{1,1}$	$c^1_{1,2}$	…	$c^1_{1,K}$

$$h^2_{1,1} = f(h^1_{1,1}, c^1_{1,1})$$

$$h^2_{1,2} = f(h^1_{1,2}, c^1_{1,2})$$

$$\vdots$$

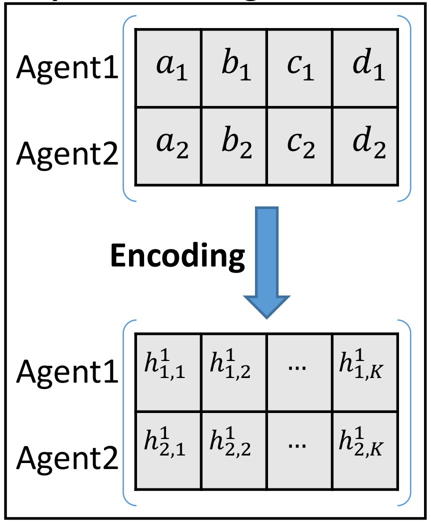
$$h^2_{1,K} = f(h^1_{1,K}, c^1_{1,K})$$

(3)Next Hidden Variable
→Dependent Variable

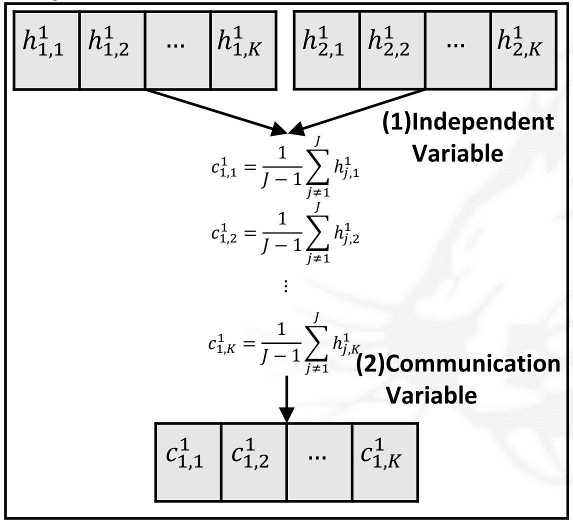
$h^2_{1,1}$	$h^2_{1,2}$	…	$h^2_{1,K}$

CommNet Mechanism

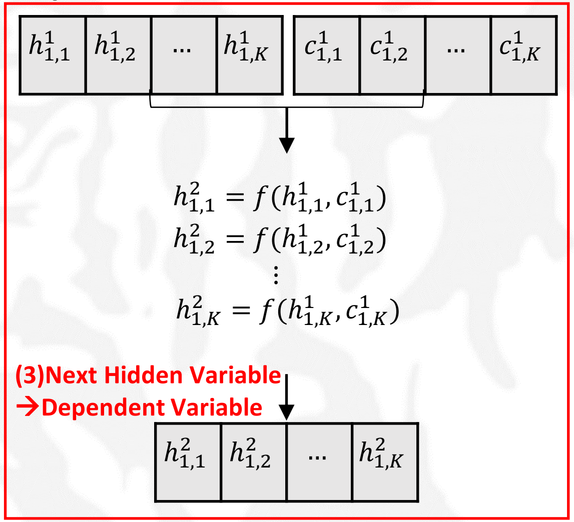
Step#1. Encoding



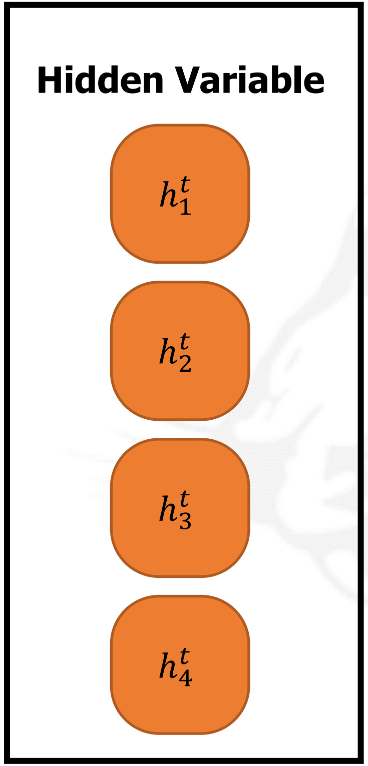
Step#2-1. Communication Variable



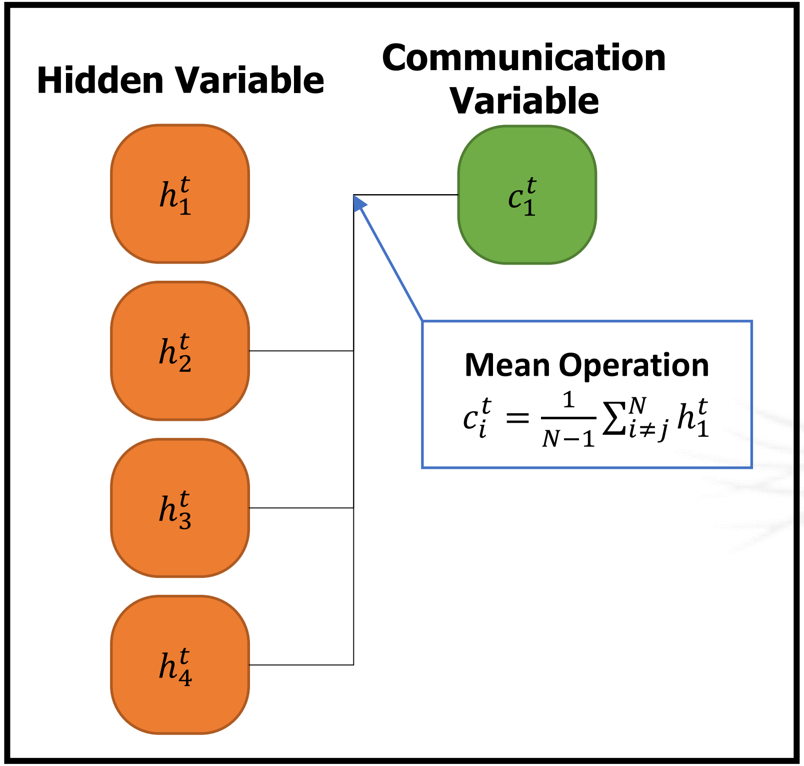
Step#2-2. Activation Function



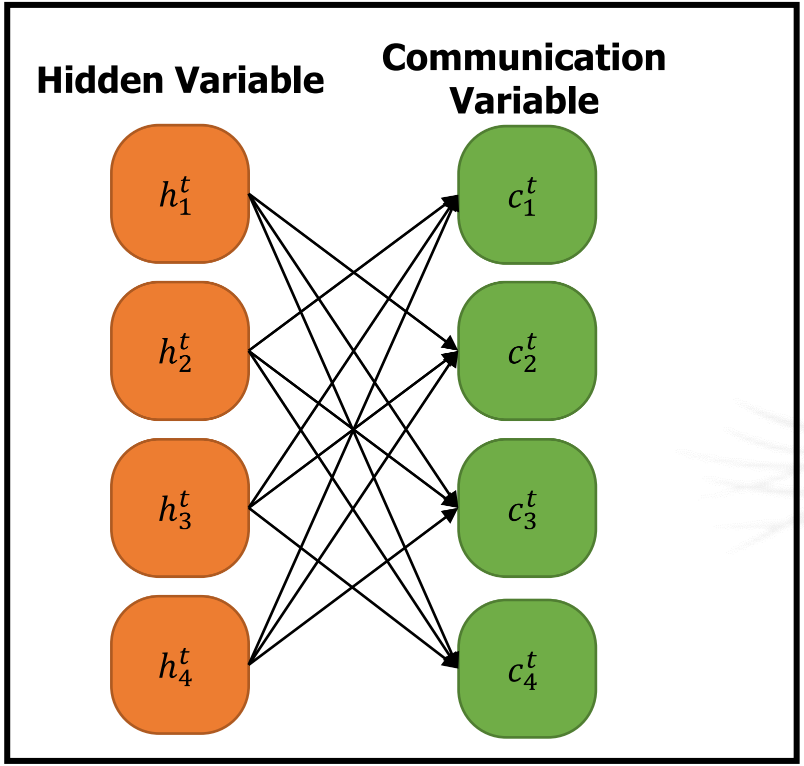
Summary of CommNet



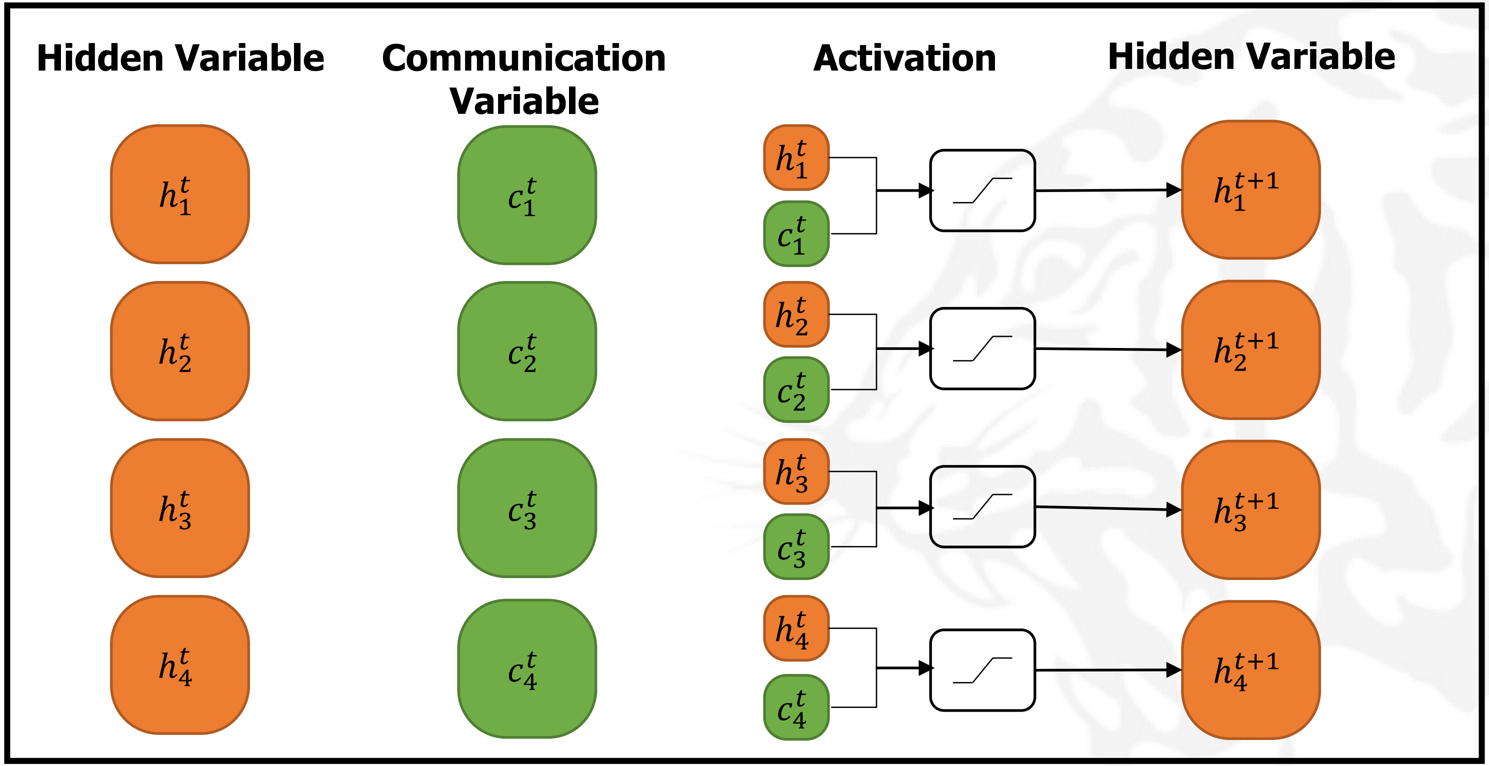
Summary of CommNet

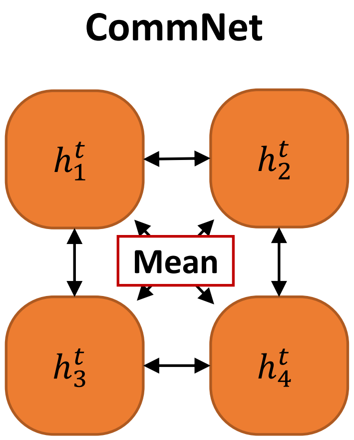


Summary of CommNet



Summary of CommNet



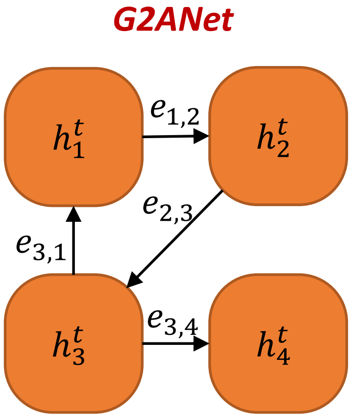


In Graph Approach.

1. Should the agent **communicate with all agent**?

2. Can we **transfer only essential** information between agents?

→ ***G2ANet will be the solution to the above problem.***



In Graph Approach.

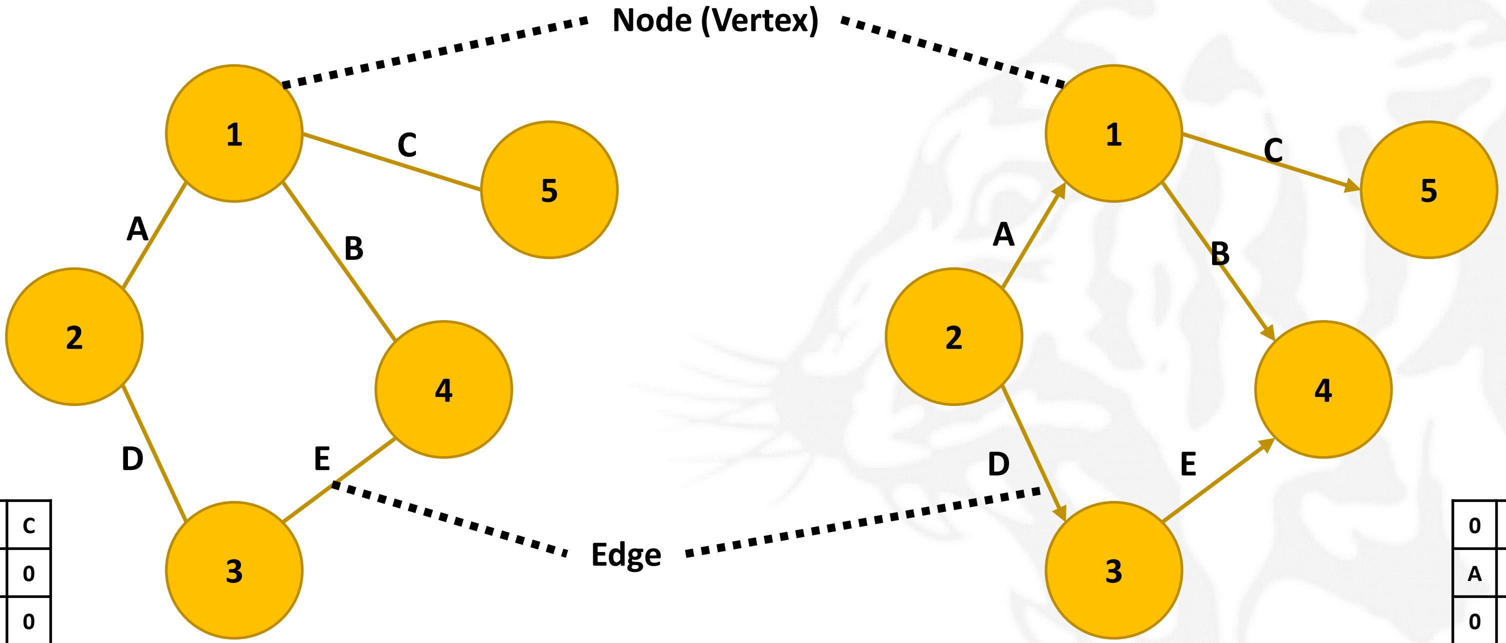
1. Should the agent communicate with all agent?

2. Can we transfer only essential information between agents?

***G2ANet* will be the solution to the above problem.**

[4] Y. Liu et al., Multi-Agent Game Abstraction via Graph Attention Neural Network, *Proc. AAAI 2020*

- Graph



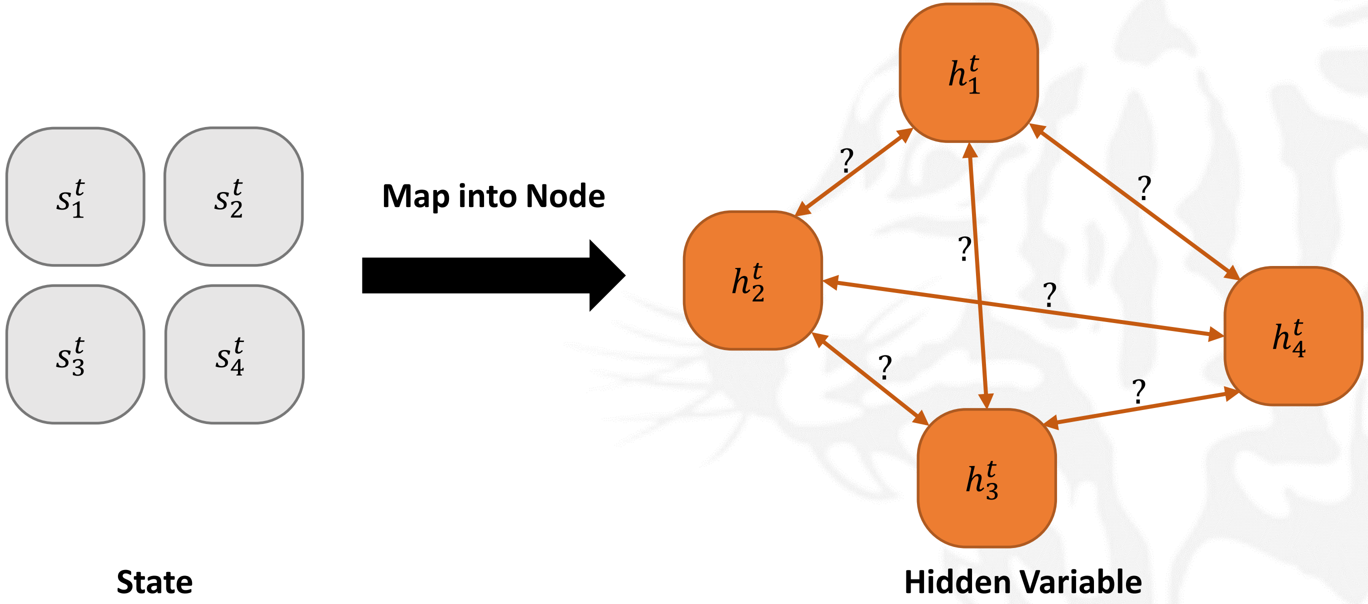
0	A	0	B	C
A	0	D	0	0
0	D	0	E	0
B	0	E	0	0
C	0	0	0	0

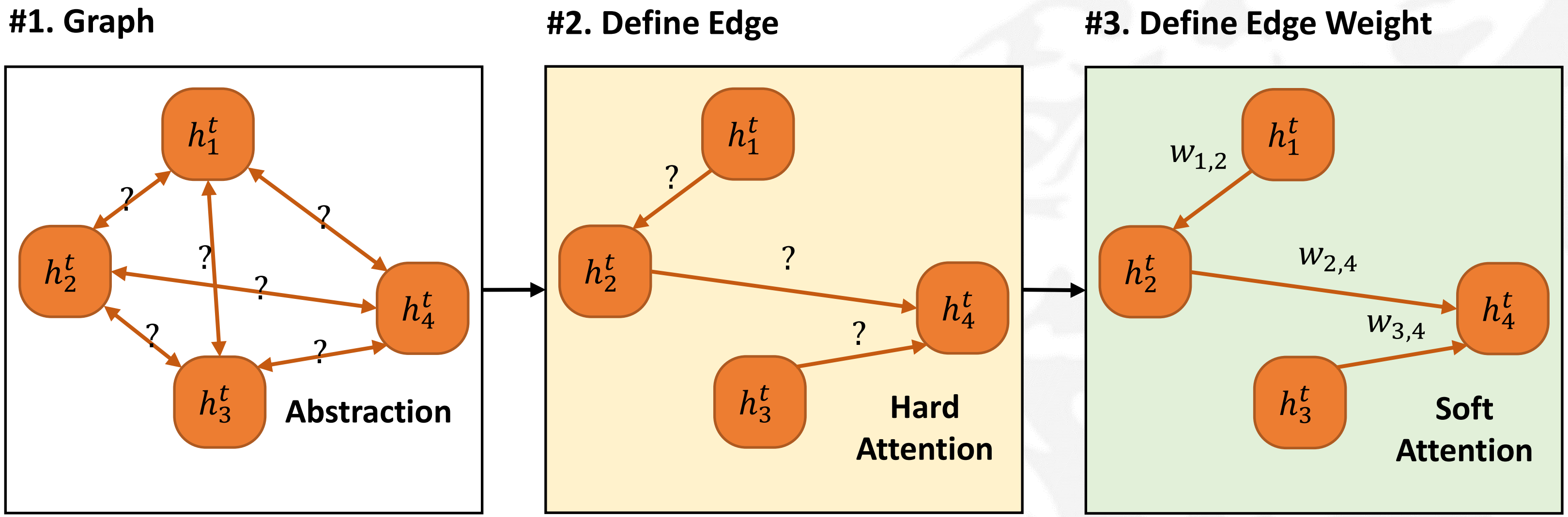
Undirected graph

Directed graph

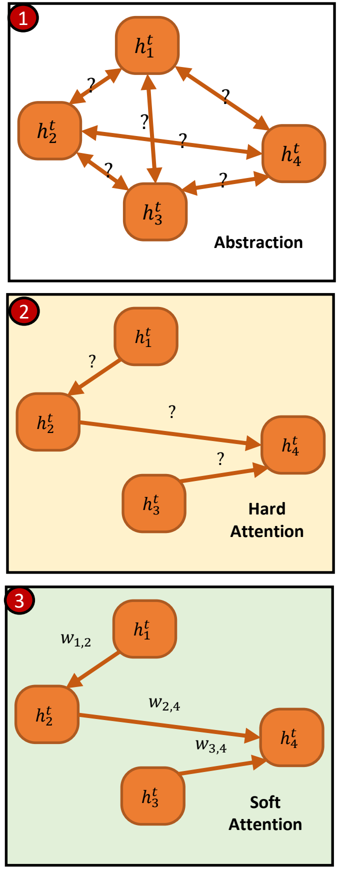
0	0	0	B	C
A	0	D	0	0
0	0	0	E	0
0	0	0	0	0
0	0	0	0	0

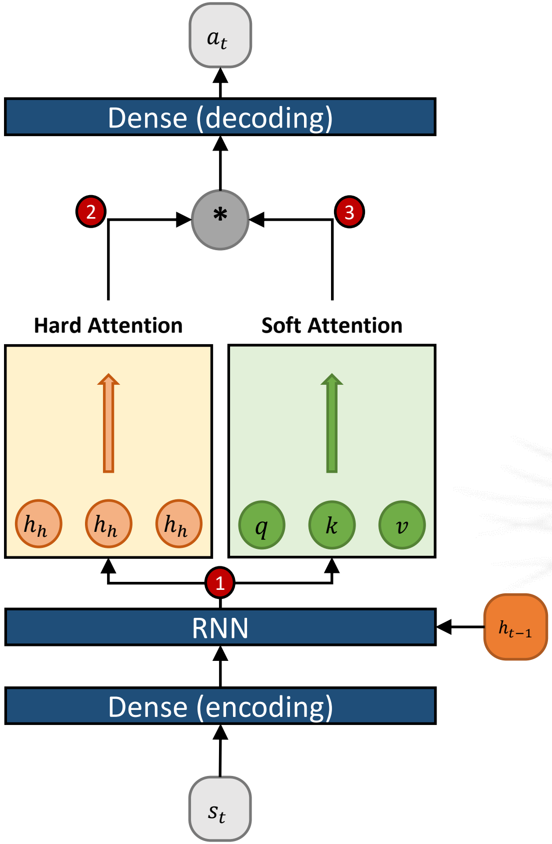
- States of agent are mapped into nodes (vertices).



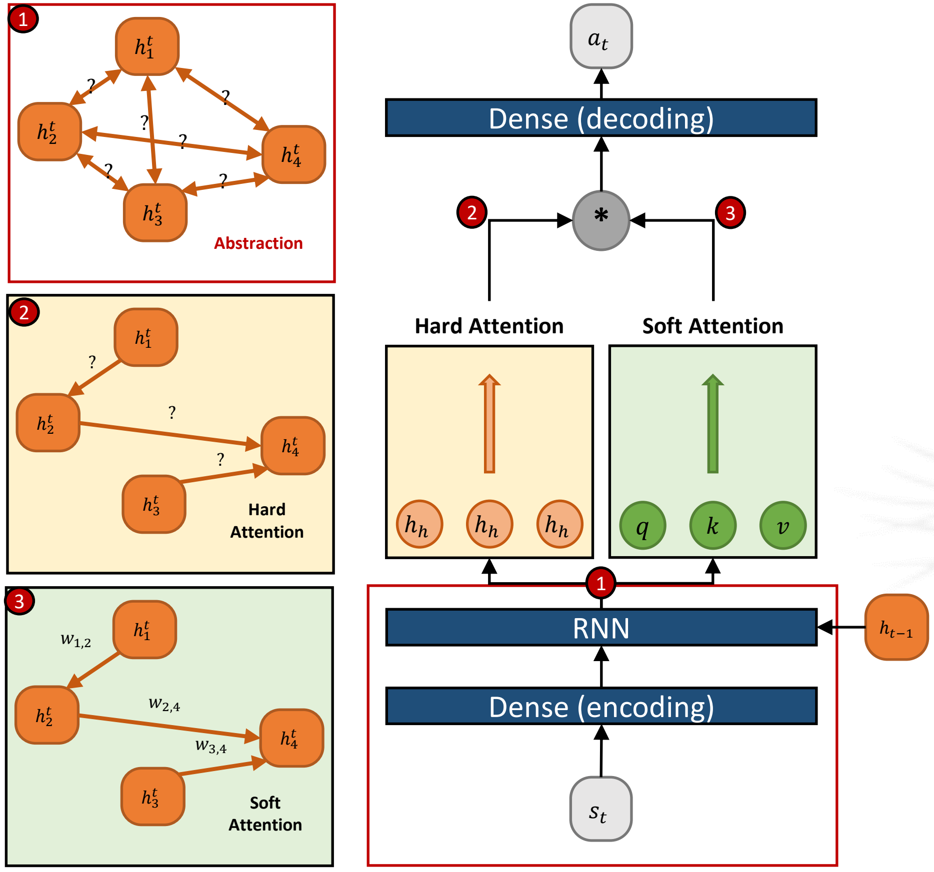

#1. Graph
h_1^t
h_2^t
h_3^t
h_4^t
?
Abstraction
#2. Define Edge
h_1^t
h_2^t
h_3^t
h_4^t
?
Hard Attention
#3. Define Edge Weight
$w_{1,2}$
$w_{2,4}$
$w_{3,4}$
h_1^t
h_2^t
h_3^t
h_4^t
Soft Attention

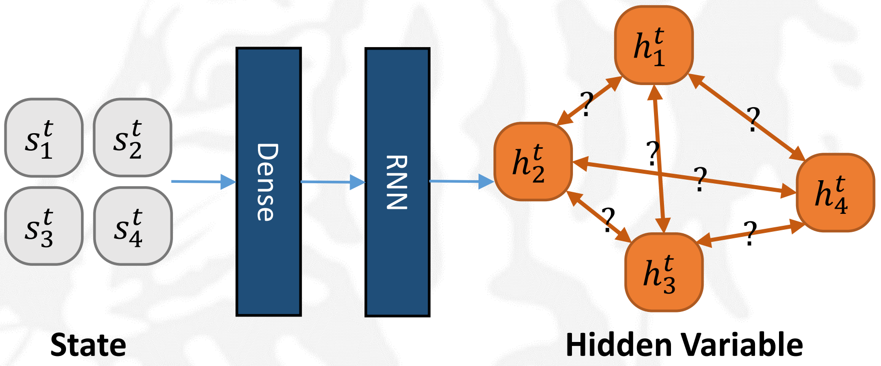
G2ANet Architecture



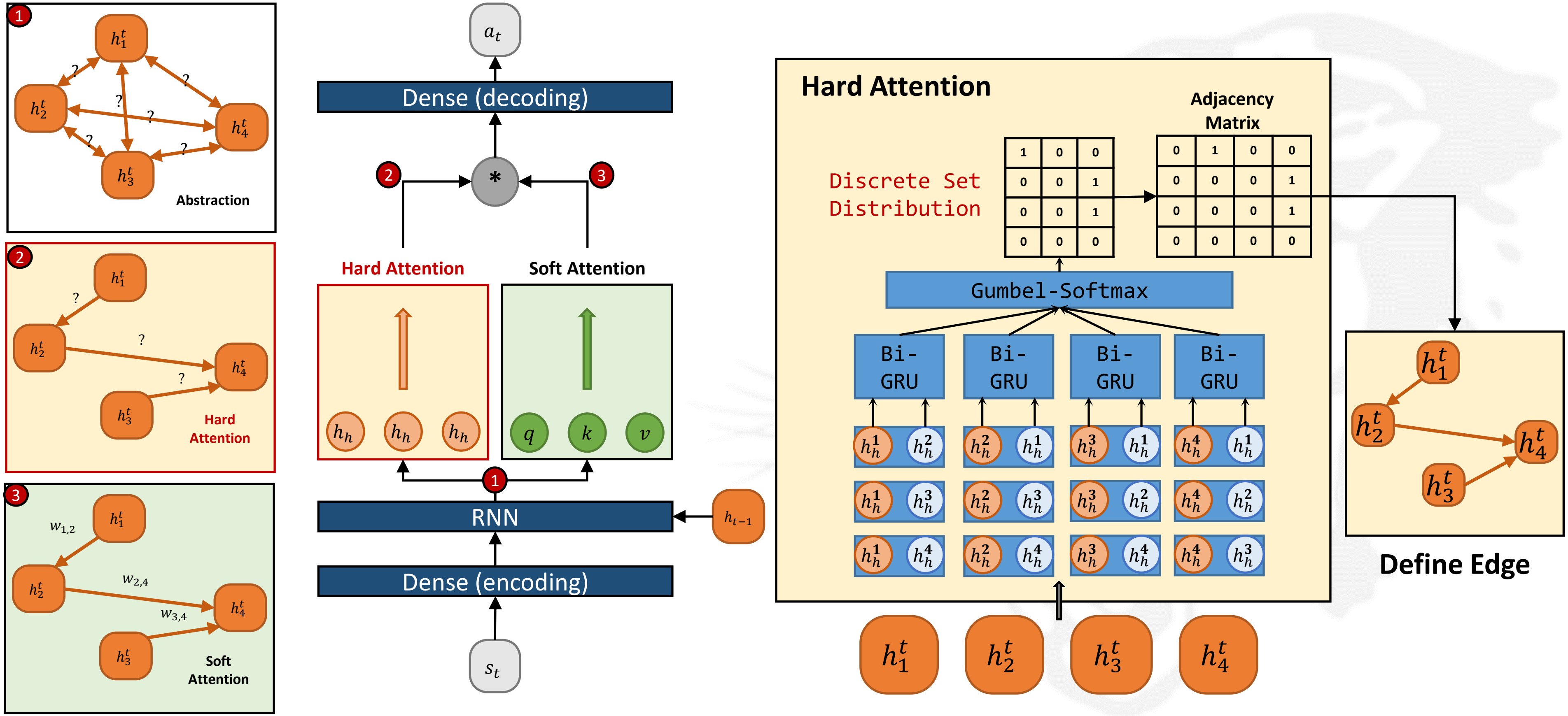


G2ANet Architecture: Abstraction



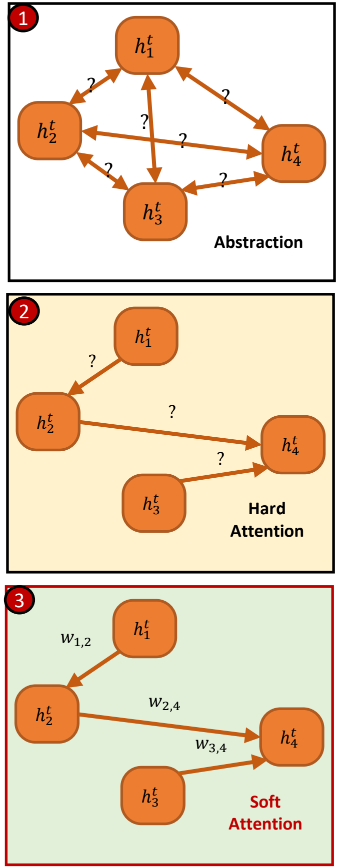


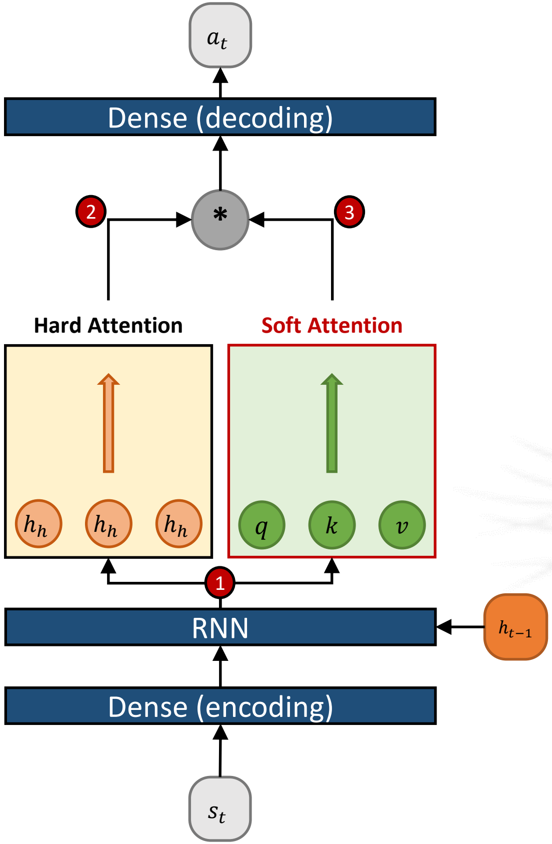
G2ANet Architecture: Hard Attention



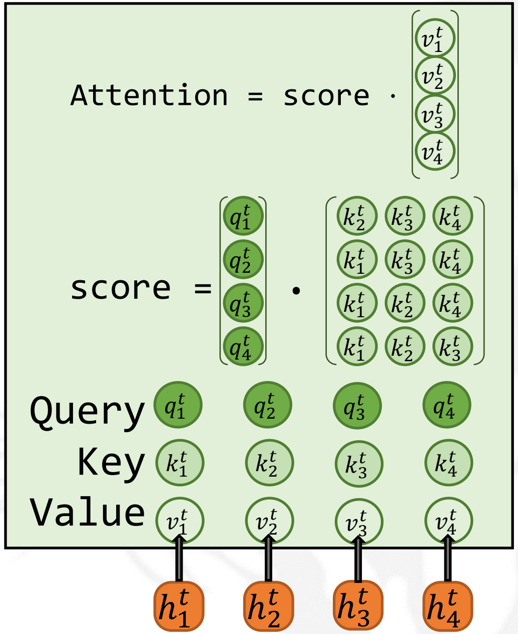
G2ANet Architecture: Soft Attention

Artificial Intelligence and Mobility Lab





Soft Attention



$$Score = \boldsymbol{q} \cdot \boldsymbol{k} = \boldsymbol{q}^{\boldsymbol{T}} * \boldsymbol{k}$$

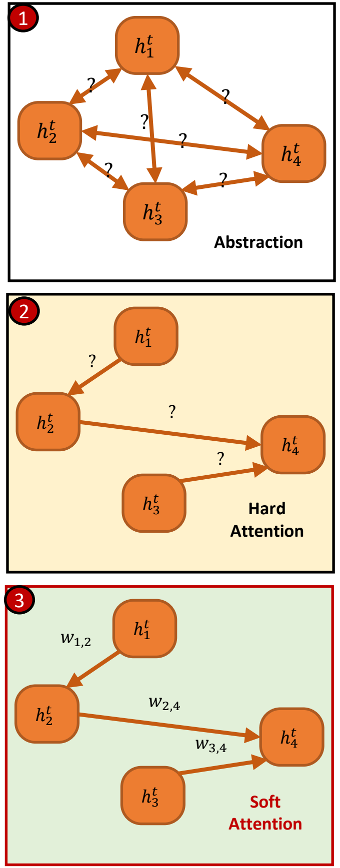
$$Score_{scaled} = \frac{Score}{\sqrt{n}}$$

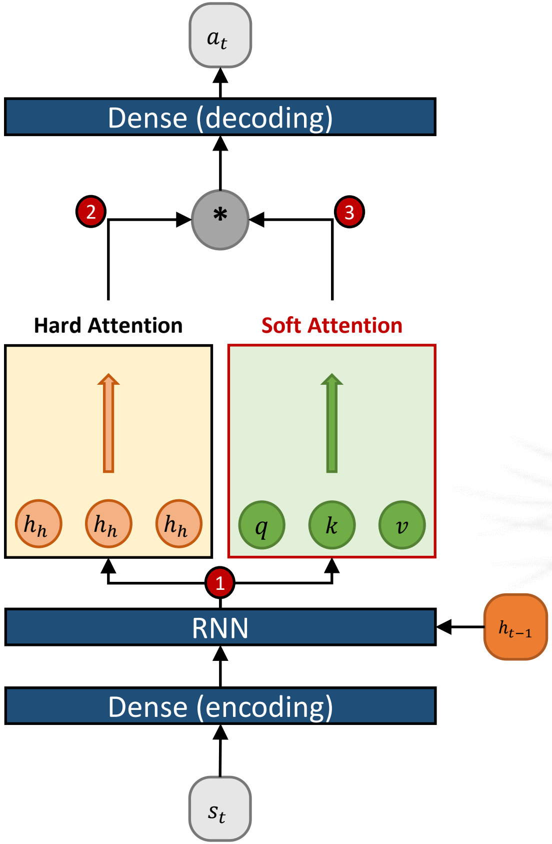
$$Attention(\boldsymbol{q}, \boldsymbol{k}, \boldsymbol{v}) = Score_{scaled} * \boldsymbol{v}$$

Soft Attention Output (message)

0.11	0.84	0.4
0.1	0.18	0.72
0.34	0.38	0.28
0.16	0.14	0.70

G2ANet Architecture: Soft Attention & GNN





Hard Attention Output (connection)

1	0	0
0	0	1
0	0	1
0	0	0

*

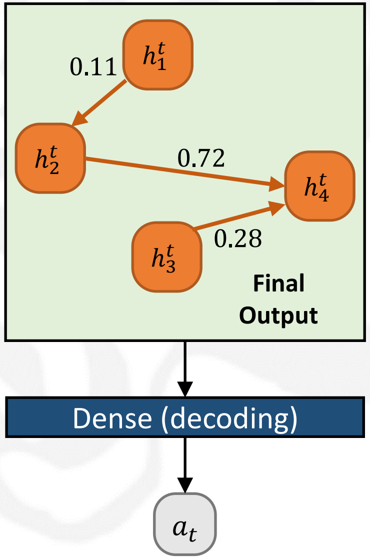
Soft Attention Output (message)

0.11	0.84	0.4
0.1	0.18	0.72
0.34	0.38	0.28
0.16	0.14	0.70

=

Final Output (connection & message)

0.11	0	0
0	0	0.72
0	0	0.28
0	0	0

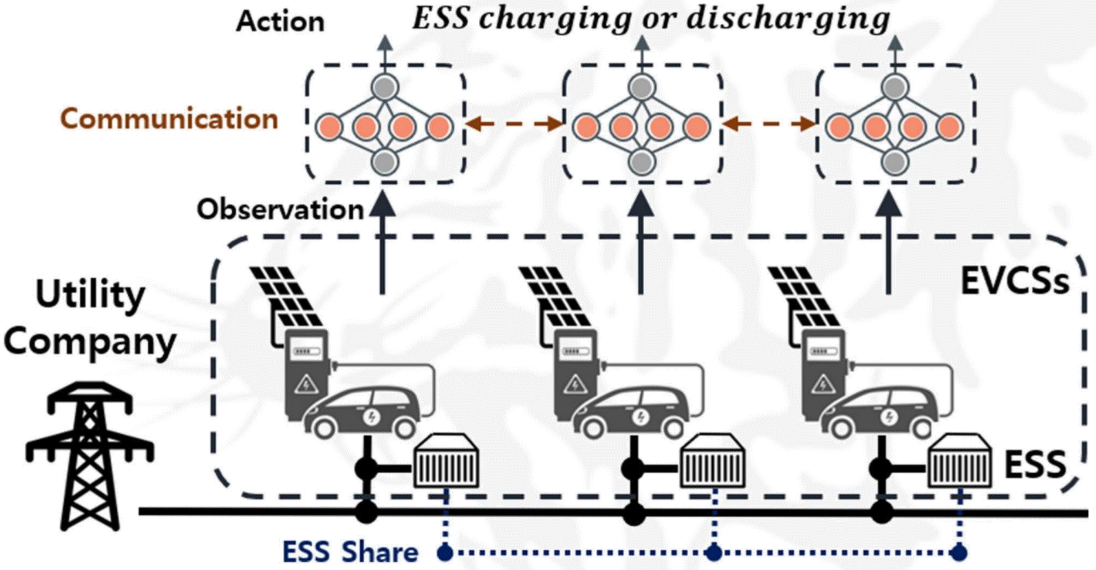


Basic Concept of DRL

Policy-based MADRL

Applications of MADRL

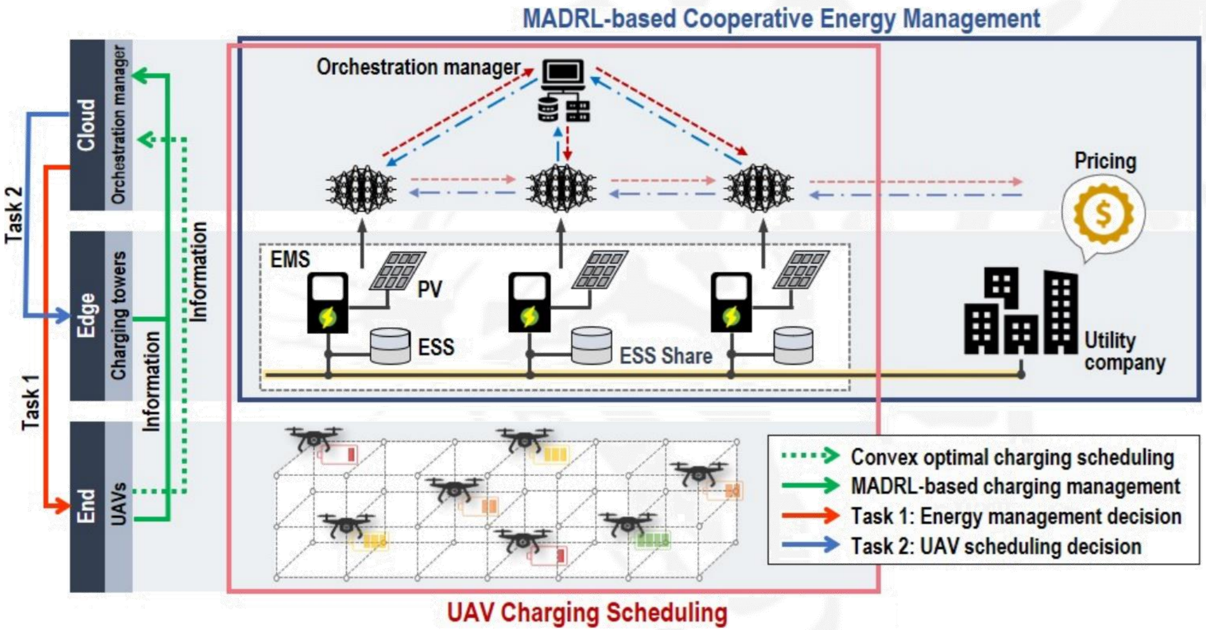
Authors: MyungJae Shin (Korea Univ.), Prof. Dae-Hyun Choi (Chung-Ang Univ.), and Prof. Joongheon Kim (Korea Univ.)



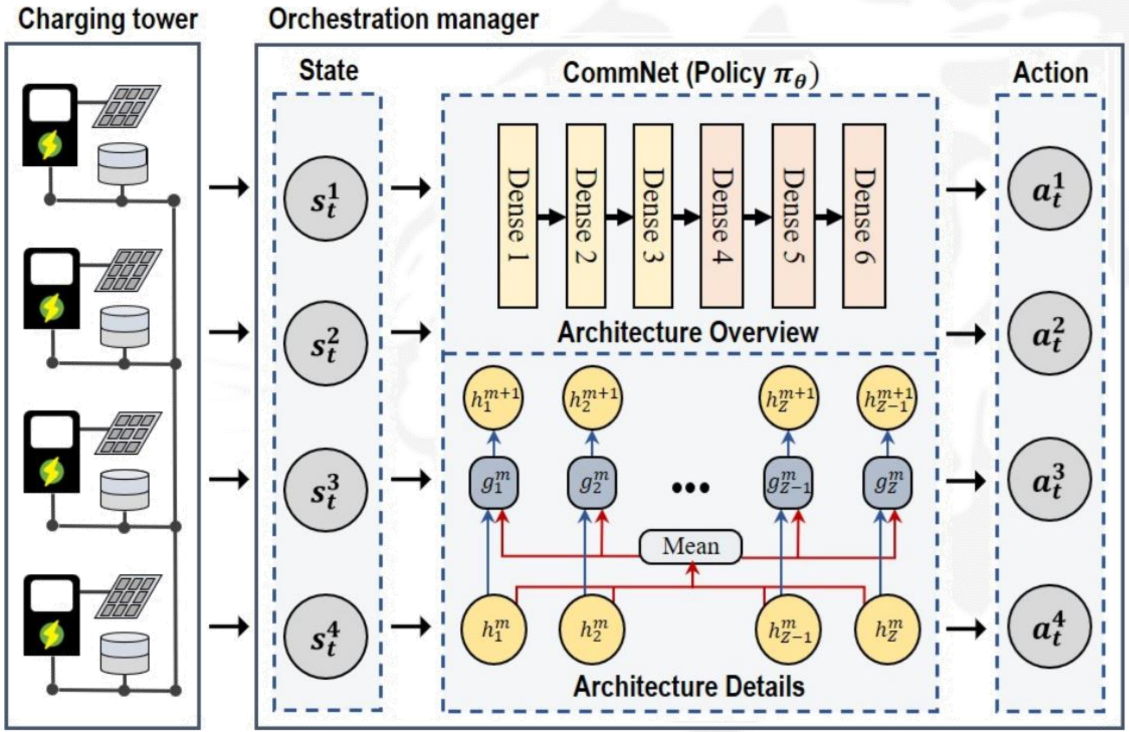
[7] M. Shin, D.-H. Choi, and J. Kim, "Cooperative Management for PV/ESS-Enabled Electric-Vehicle Charging Stations: A Multiagent Deep Reinforcement Learning Approach," *IEEE Transactions on Industrial Informatics*, 16(5):3493-3503, May 2020.

CommNet: Multi-UAV Charging Systems

Authors: Soyi Jung (Ajou Univ.), Won Joon Yun (Korea Univ.), MyungJae Shin (Korea Univ.), Prof. Joongheon Kim (Koera Univ.), and Prof. Jae-Hyun Kim (Ajou Univ.)



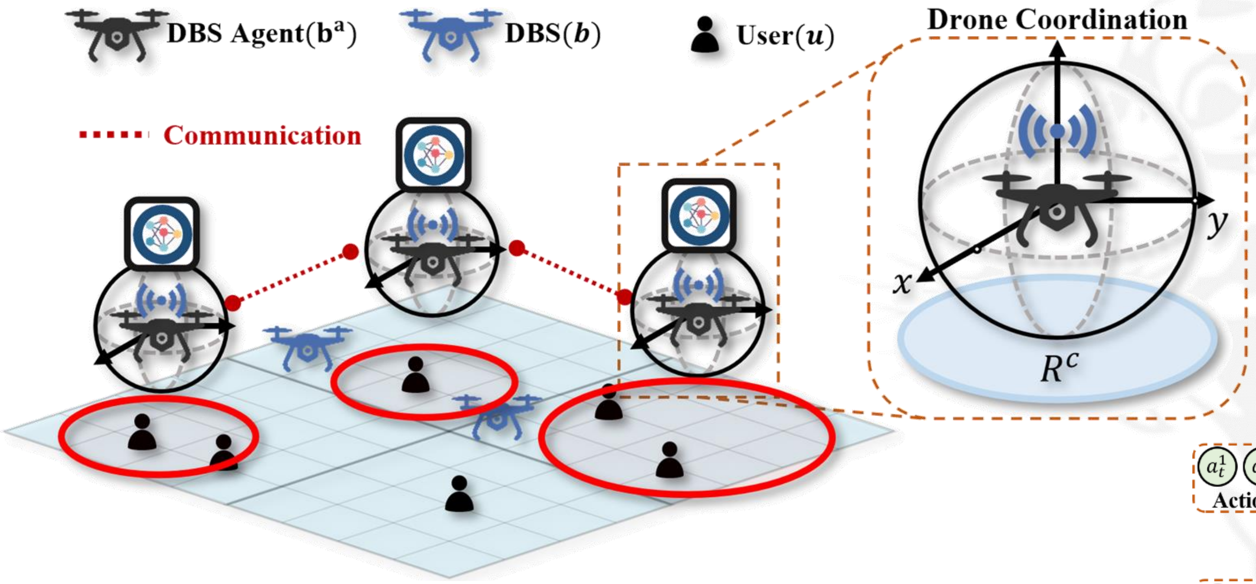
Authors: Soyi Jung (Ajou Univ.), Won Joon Yun (Korea Univ.), MyungJae Shin (Korea Univ.), Prof. Joongheon Kim (Koera Univ.), and Prof. Jae-Hyun Kim (Ajou Univ.)

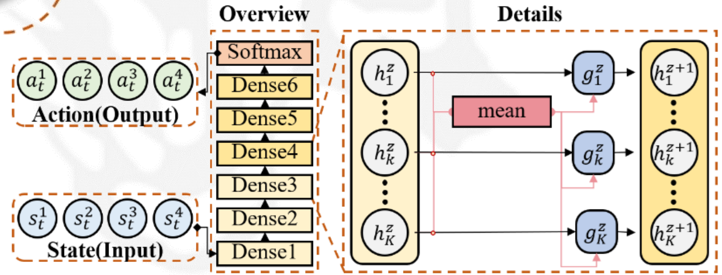


CommNet: Autonomous Surveillance Drones

Artificial Intelligence and Mobility Lab

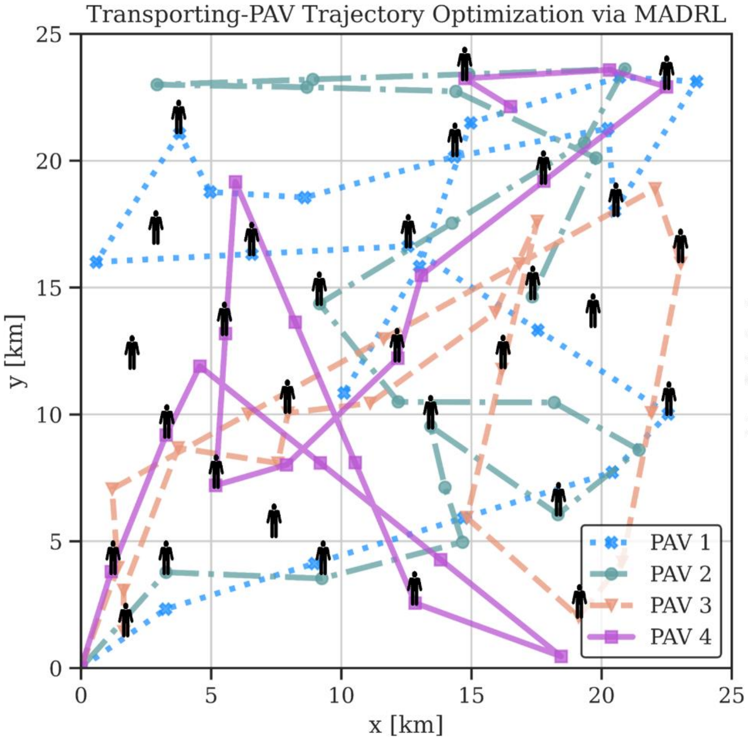
Authors: MyungJae Shin (Korea Univ.), Won Joon Yun (Korea Univ.), Soyi Jung (Ajou Univ.), Soohyun Park (Korea Univ.), Prof. David Mohaisen (UCF), Prof. Joongheon Kim (Koera Univ.), and Prof. Jae-Hyun Kim (Ajou Univ.)

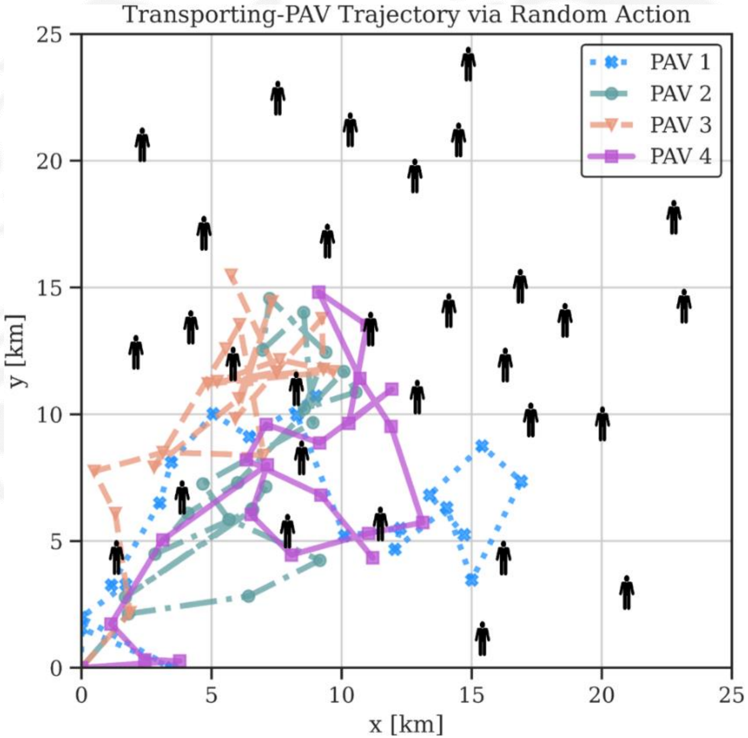




G2ANet: Autonomous Drone Trajectory Learning

Authors: Won Joon Yun (Korea Univ.), Yoo Jeong Ha (Korea Univ.), Soyi Jung (Ajou Univ.), Prof. Jae-Hyun Kim (Ajou Univ.), Prof. David Mohaisen (UCF), and Prof. Joongheon Kim (Koera Univ.)





Thank you for your attention!

- Special thanks to **Won Joon Yun (Ph.D. Student at EE, Korea Univ.)**
- More questions?
  - joongheon@korea.ac.kr