2025-2 RL project

팀명 : Explorer (7조)

팀원 : 120250673 조장현
120250677 박성진
220250012 고건영

https://github.com/Sungjin-Park-dev/RL_Project_25-2

Index

프로젝트 주제 및 목표

MARVEL (baseline method)

- Multi-Agent Reinforcement Learning for constrained field-of-View multi-robot Exploration in Large-scale environments
- 목표: 여러 대의 로봇이 미지의 실내 환경을 가장 짧은 궤적으로 탐사해서 효율적으로 100%의 coverage를 달성하는 것
  - map정보를 graph로 축약해 node로 변환하여 사용, 학습은 SAC(Soft Actor-Critic)로 진행

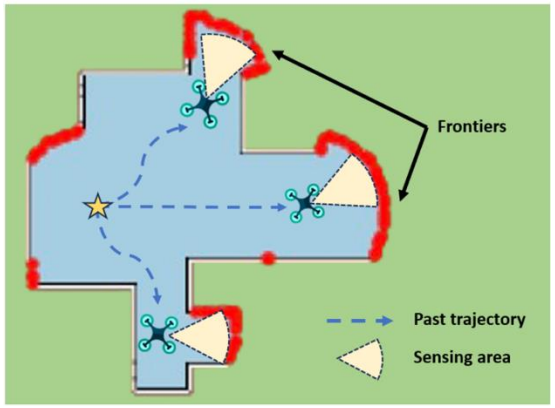
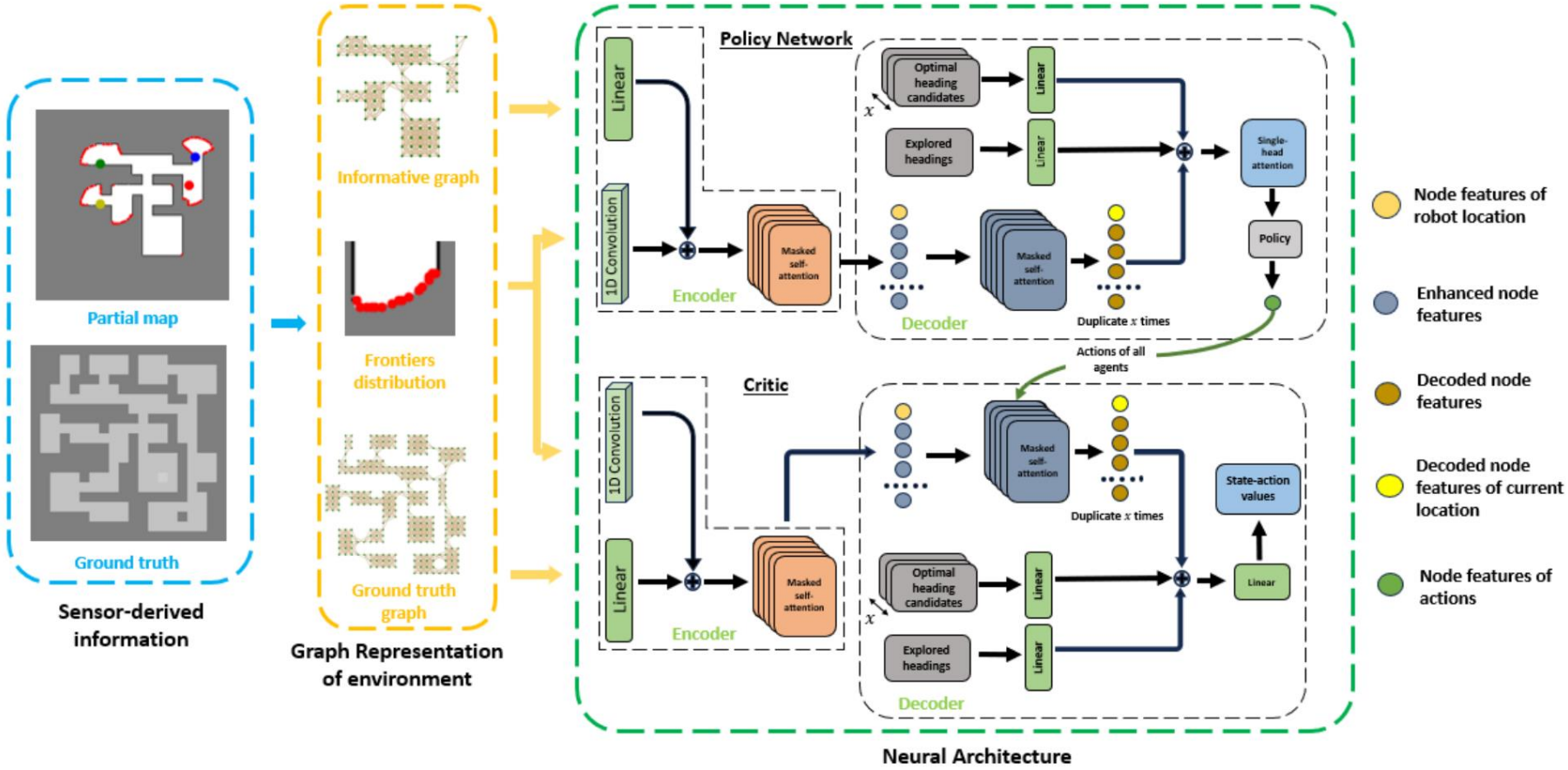


Illustration of multi-robot exploration



MARVEL's policy and critic network architecture

프로젝트 주제 및 목표

MARVEL (baseline method)

- Multi-Agent Reinforcement Learning for constrained field-of-View multi-robot Exploration in Large-scale environments
- 목표: 여러 대의 로봇이 미지의 실내 환경을 가장 짧은 궤적으로 탐사해서 효율적으로 100%의 coverage를 달성하는 것

  - 현실적인 제약:
    - 기존 방법들은 LiDAR같은 360도 전방위 센서를 가정
    - 하지만 드론, 소형 로봇은 카메라처럼 Field of View(FoV)가 제한된 센서만 사용하는 경우가 많음
    - 어디로 이동할지(waypoint) 뿐만 아니라 어느 방향 (heading)을 바라볼 지까지 함께 계획해야 해서 state, action space가 훨씬 커짐

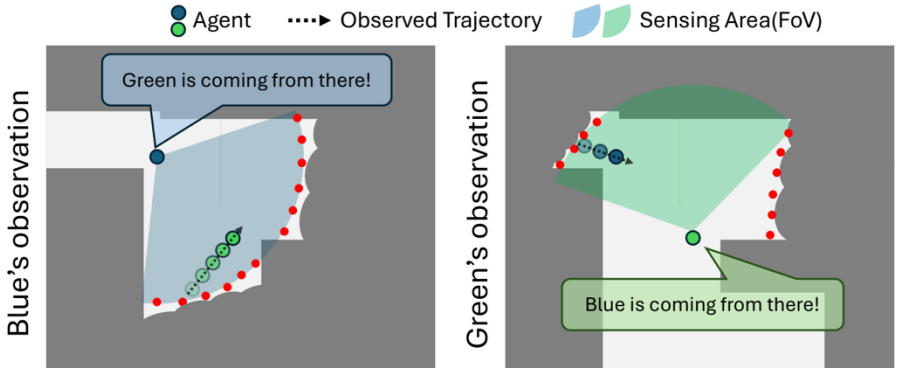
  - 기존방법 한계:
    - 완전한 통신(perfect communication)을 전제하지만, 실제 환경에서는 통신 인프라 부족, 무선 환경 불안정하여 현실적이지 않음
    - 전역 지도(global map)을 공유하고 명시적인 메시지 교환이 있어야만 policy 학습이 이루어짐
      - 통신이 없는 상태에서는 서로 정보공유가 불가하므로 탐사가 어려움

→ 그렇다면 통신이 끊겨 Map & Message를 주고받을 수 없을 때, **다른 로봇의 이동경로만 보고도 효율적인 탐사가 가능하지 않을까?**

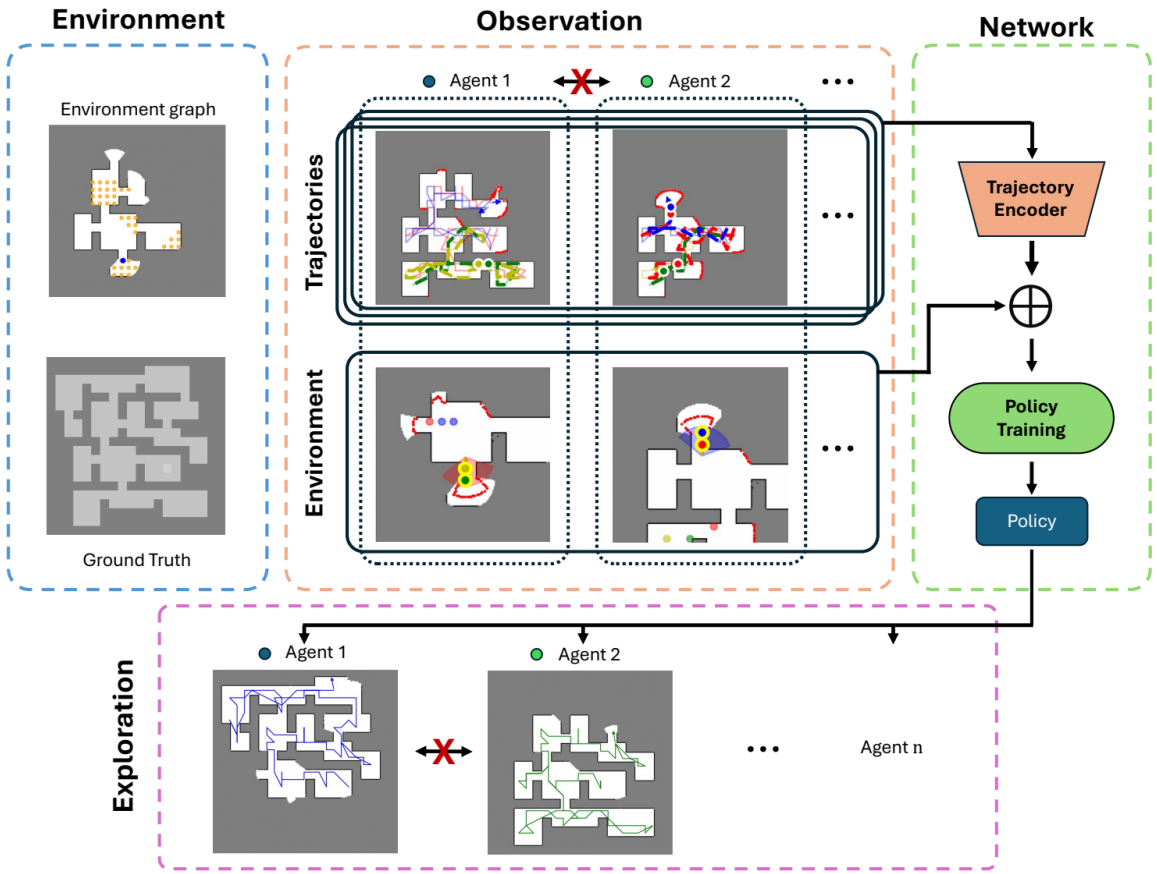
프로젝트 주제 및 목표

Our method (proposed algorithm)

- Multi-Agent Reinforcement Learning for Communication-Free Exploration via Inter-Agent Trajectory Observation
  - 어떠한 통신도 하지 않고, 시야 안으로 들어온 다른 로봇들의 최근 이동 궤적(위치,속도,방향)을 계속 관찰하고 저장
  - 이 궤적들을 trajectory sequence로 보고, transformer 기반 encoder로 인코딩, 환경 정보와 융합하여 policy 학습
    → 통신 없이도 다른 로봇이 어디서 왔는지 & 어디로 향하는지를 보고, 중복 탐사를 피하고 암묵적으로 효율적인 action을 스스로 학습



Communication-free multi-robot exploration



환경 및 데이터셋

환경 구성

- 미지의 실내환경 *E*에서 여러 대의 로봇이 동시에 탐사를 수행하는 시나리오를 가정
- 각 환경은 로봇이 이동하며 known region과 unknown region이 시간에 따라 갱신되는 형태이며, frontier-based exploration 문제로 모델링
  - frontier란? 로봇이 이동할 수 Open-space와 아직 센서 정보가 없는 Unknown region의 경계

로봇 구성

- 6대의 로봇으로 고정, 모든 실험에서 동일하게 6대가 동시에 탐사

센서 제약 조건(partial observation)

- 각 로봇은 제한된 시야(Field of View, FoV)를 가짐
- 센서 사거리 : 10m, 시야각 : 120도
  - 따라서 각 로봇은 자기 주변의 일부 환경과 시야각 안에 들어온 다른 로봇만 관측할 수 있으며, 전역(global) 지도를 직접 보거나 통신으로 공유받지 않음

맵/환경 구성

- 실험은 90m x 90m 크기의 지형과 구조가 서로 다른 실내 환경 10개에 대하여 수행

환경 및 데이터셋

Training set

- 5663개의 무작위로 생성된 대규모 실내 map
- 모두 90m x 90m, 각 환경 당 4대의 로봇을 기준으로 학습

Test set

- 100개의 unseen map으로 평가 (10개의 서로 다른 indoor 환경 x 10개의 상이한 start-goal 세트)
- test시에도 train과 마찬가지로 90m x 90m, 동일 센서 설정(FoV 10도, range 10m)

환경 및 데이터셋

Data processing pipeline

- Grid map → Graph 변환

  매 시점 t마다 로봇들은 환경을 collision-free graph $G_t = (V_t, E_t)$로 표현한다.
  - V_t: 후보 viewpoint (위치)들의 집합
  - E_t: 해당 viewpoint 사이의 collision-free edge (reachable connection)

- Frontier 분포 F_t계산
  - 각 노드별로 frontier distribution F_t 를 추출 후 36개 방향(10도 단위)로 FoV 샘플링
  - 각 방향 bin마다 해당 heading으로 센서를 둘 때 frontier가 얼마나 보이는지 측정 → 정규화된 scalar로 저장
  - 36차원 벡터가 해당 노드의 frontier feature가 되고,1D convolution등을 통해 벡터를 임베딩하여 node-level의 frontier feature로 사용

- 향상된 그래프 G_t' 생성
  - 원 그래프 G_t에서 각 노드는 다음과 같은 feature를 가짐 $(\Delta x_{ik}, \Delta y_{ik}, u_k, o_k, g_k, h_k)$

    $\Delta x_{ik}, \Delta y_{ik}$: 현재 로봇 *i*기준 상대 좌표, u_k: 센서 범위 내에 보이는 frontier수, o_k: agent occupancy, g_k: guidepost signal, h_k : informative heading
  - 위 scalar feature들과 frontier distribution F_t임베딩을 concat해 다시 d차원 feature로 projection → 그래프 encoder 입력 G_t' 로 사용

- 시야 내의 다른 로봇의 경로(Trajectory) observation data처리 $T_t^j = \{(x^{t-k}, y^{t-k}, \phi^{t-k}, v^{t-k})\}_{k=0}^{9}$
  - 각 로봇 *i*는 FoV안에 보이는 다른 로봇 *j*에 대해, 최근 10step의 상태*(x, y, heading, vel)*를 버퍼에 저장
  - Trajectory encoder로 각 step의 4d 벡터를 임베딩 후 positional encoding을 더한 다음, temporal encoder로 long-term motion pattern추출
  - 여러 로봇 궤적을 multi-head attention기반 aggregation으로 통합 → 단일 trajectory embedding Z_t를 생성

State, Action, Reward 설계

State 설계

- 기존 논문인 MARVEL에서 각 시점의 observation은 다음과 같이 정의:

$$o_t = (G'_t, \Psi_t, F_t)$$

G'_t : 위에서 설명한 node feature가 포함된 environment graph.

Ψ_t : 모든 에이전트의 현재 viewpoint(노드 위치) 집합.

F_t : 각 노드에 대한 36-bin frontier distribution.

- Trajectory observation 추가를 고려한 확장

$$o_t^{new} = (G'_t, \Psi_t, F_t, T_t)$$

여기서 T_t는 설명한 시야 내 타 로봇들의 최근 10step trajectory 집합이며, trajectory와 fusion layer를 통해 하나의 embedding Z_t 로 요약됨

즉, 기본 baseline(그래프+frontier)과 동일한 그래프 상태에, 서로 간의 궤적 관측이라는 새로운 parameter를 추가

State, Action, Reward 설계

Action 설계

- Action space
  - 로봇의 행동은 환경 그래프 상의 이웃 node + heading 선택으로 정의됨
  - 기본적으로는 로봇의 현재 node에서 인접 node로 이동하는 action후보
  - 하지만 FoV 방향까지 포함해야 하므로, 각 node에 대해 정보량이 높은 heading후보를 몇 개만 남기는 action pruning방법을 사용

- 정보기반 Action pruning
  - 각 인접 node에 대해, frontier 관측량이 높은 상위 3개 heading을 골라서, joint action을 discrete action으로 사용
  - frontiers가 보이지 않는 경우에는, guidepost g_k로부터 A* 경로를 따라가는 heading후보 샘플링

- Pointer network 기반 policy 출력
  - policy decoder는 로봇의 현재 node feature h_c를 query로, 주변 neighbor node들의 feature + heading feature를 key/value로 사용해 attention 수행
  - 이 때, 각 neighbor-heading 조합에 대한 attention weight가 곧 policy 확률로 사용 $\pi_\theta(a_{i,t}|o_{i,t})$

State, Action, Reward 설계

Reward 설계

- 기존 논문인 MARVEL에서 개별 로봇 *i*에 대한 reward는 다음과 같이 정의됨
  - $r_i = r_o + ar_h + r_t + r_f$
    - r_o: 새 viewpoint에서 관측 가능한 frontier 수 기반의 단일 로봇 reward
      - 이 노드에 있으면 얼마나 frontier를 많이 볼 수 있는가에 대한 reward
    - r_h: heading alignment reward
      - A* 경로방향과 현재 선택한 heading사이의 각도 차이 cosine값 (계수 a=0.3으로 scaling)
      - 즉, A*로 향할수록 보상이 커지므로 목표 frontier를 향해 부드럽게 회전하도록 유도하는 reward
    - r_t: 로봇 전체가 관측한 frontier수에 대한 team-shared reward
      - 모든 로봇의 frontier 관측량을 합산해 normalization
    - r_f: 탐색 완료에 대한 보상
      - 모든 노드의 frontier utility가 0이 되어 탐사가 완료되고 coverage가 99%에 도달했을 때 +10 부여
- Trajectory observation에 대한 보상 추가
  - 이미 다른 로봇 궤적이 많이 지나간 영역을 피하면 $r_t^{trajectory}$가 커지도록 설계
    - $r_t = r_t^{utility} + r_t^{trajectory} + r_t^{team}$

RL algorithm & Hyperparameter

사용 알고리즘: Soft Actor-Critic (SAC, discrete)

- Soft Actor-Critic(SAC)기반의 off-policy RL을 사용
- CTDE(Centralized Training, Decentralized Execution) 구조:
  - Actor: 실제 실행에서 사용할 partial observation 상태 o_t(그래프+trajectory)만 입력
  - Critic(Q-network): 학습 시에만 full ground-truth map을 더 많이 보며 value를 학습
- Discrete action SAC (Christodoulou, 2019)를 사용
  - Multi-head attention / pointer network가 뱉는 discrete action probability를 바로 policy로 사용

- Network Architecture
  - 각 로봇 궤적에 대해 transformer 기반 temporal encoder적용, multi-head attention으로 여러 로봇 궤적을 aggregate하여 단일 embedding Z_t 형성
  - Policy decoder
    - 로봇의 현재 node feature h_c를 query로, 주변 neighbor node들의 feature + heading feature를 key/value로 사용 → pointer network 형태
  - Critic decoder
    - Critic은 actor와 동일한 encoder를 쓰되, decoder에서 다른 로봇들의 state/action까지 함께 attention에 넣음
    - 이를 통해 credit assignment를 개선하는 구조(Actor-Attention-Critic)

RL algorithm & Hyperparameter

Hyperparameter

- Episode 최대 길이: 128step
- Discount factor: 1 (long-horizon coverage를 위해 감가부여하지 않음)
- Batch size: 256
- Episode replay buffer size: 10000 steps
- Target entropy: 0.01*log(k) (k=action수)
- Optimizer: Adam, learning rate: 1e-5 (policy & critic 동일)
- Target critic 업데이트 주기: 매 256step마다

Experiment setup & Evaluation metric

실험환경

- 환경: 90m x 90m 실내 환경, 100개의 unseen map(서로 다른 환경 10개 x 각 환경마다 random start-goal set 10개)
- 로봇 구성 및 센서: 로봇 6대, 사거리 : 10m, 시야각 : 120도
- 비교 설정:
  - **Perfect Communication (MARVEL)**
    - 항상 완전한 shared 맵을 가진 이상적인 방법
  - **Zero Communication (MARVEL)**
    - 로봇 간 shared 맵/정보 없이, 각자 local state만 보고 결정
  - **Zero Communication + Trajectory observation (Our method)**
    - 아무런 통신 없음
    - FoV안에서 관측되는 다른 로봇의 trajectory(궤적)만 보고 policy 결정

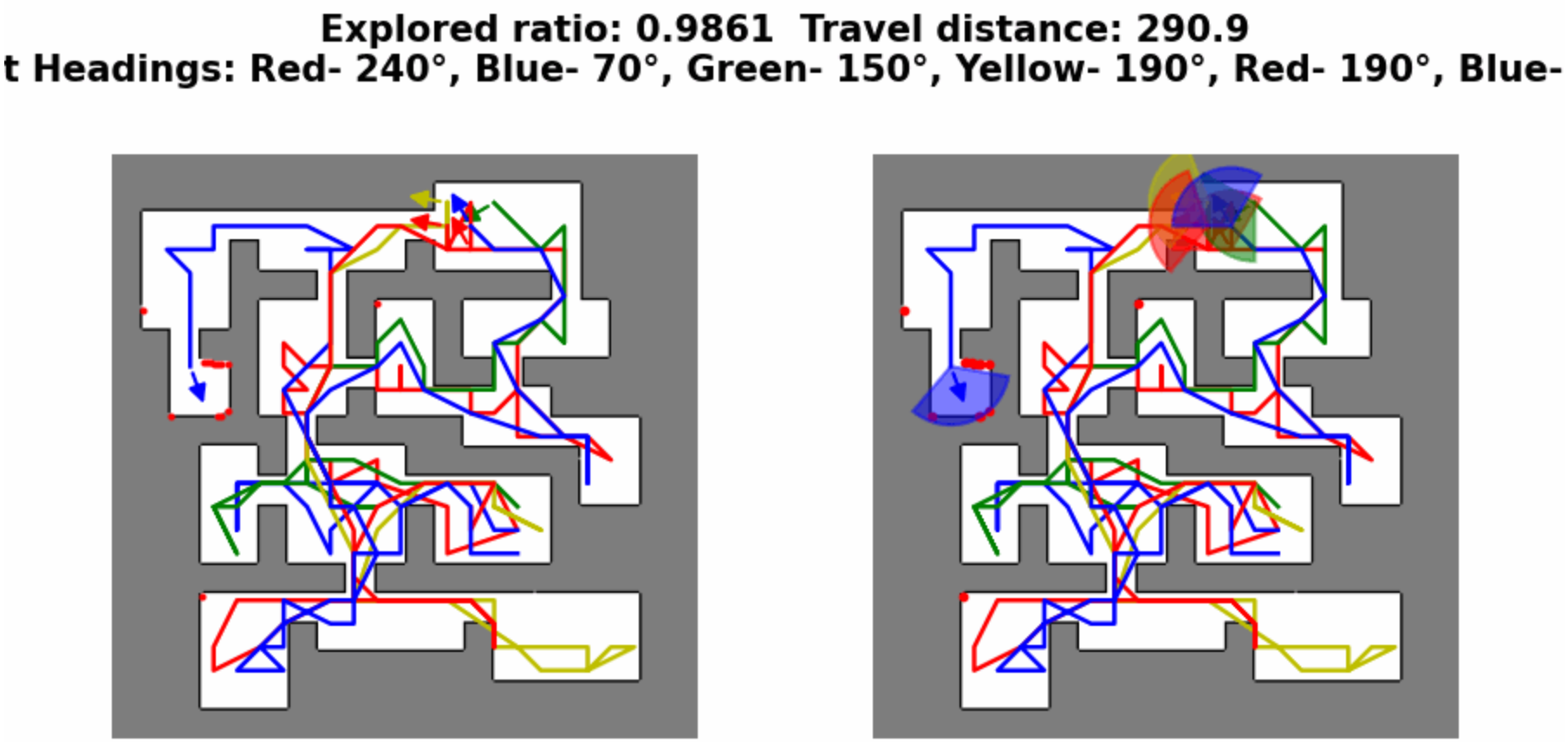
Evaluation metric

- The average and standard deviation of the trajectory length required to achieve full exploration 99% coverage and 90% coverage

실험 결과

Perfect Communication (MARVEL)

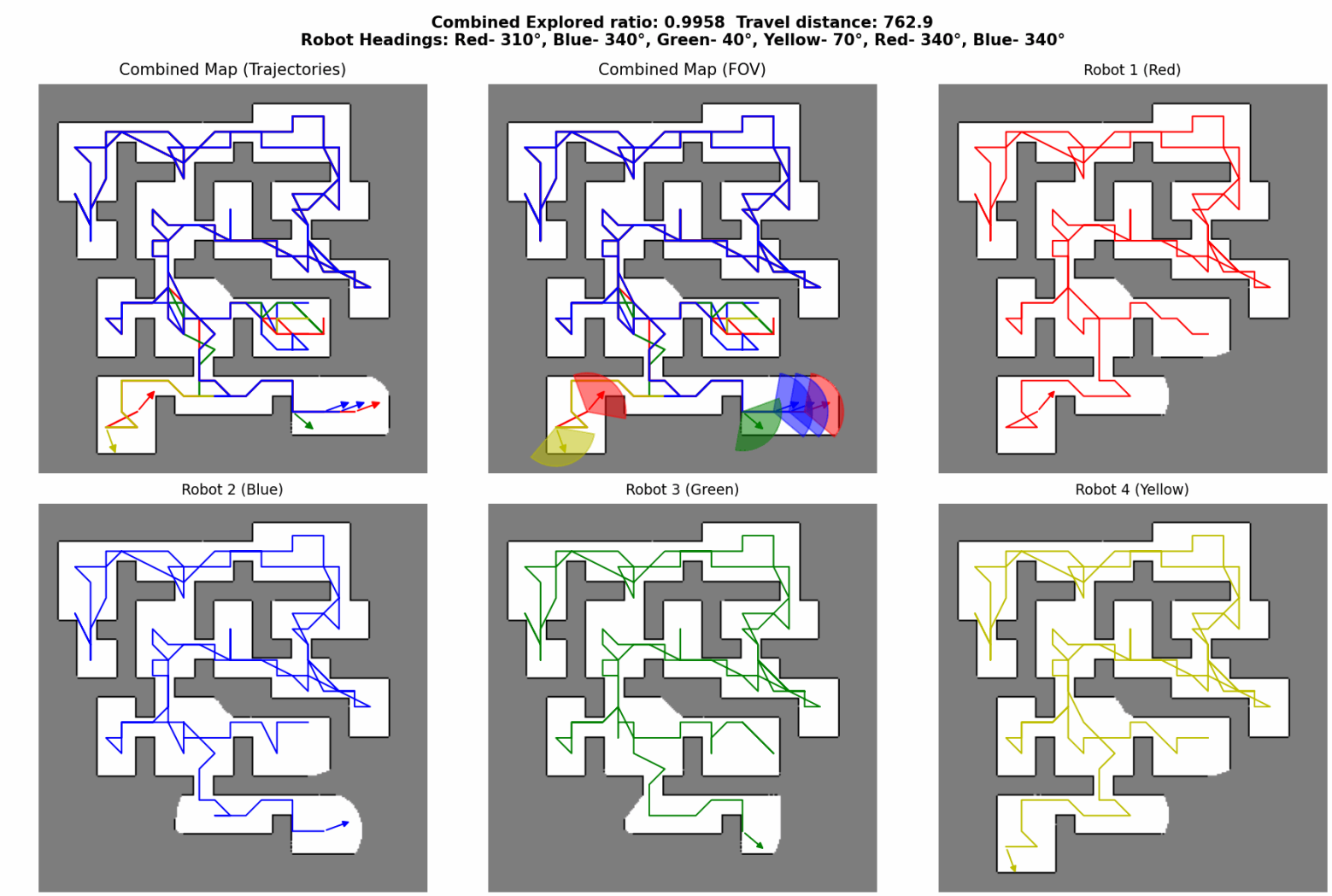
- Travel distance : 290.9



실험 결과

Zero Communication (MARVEL)

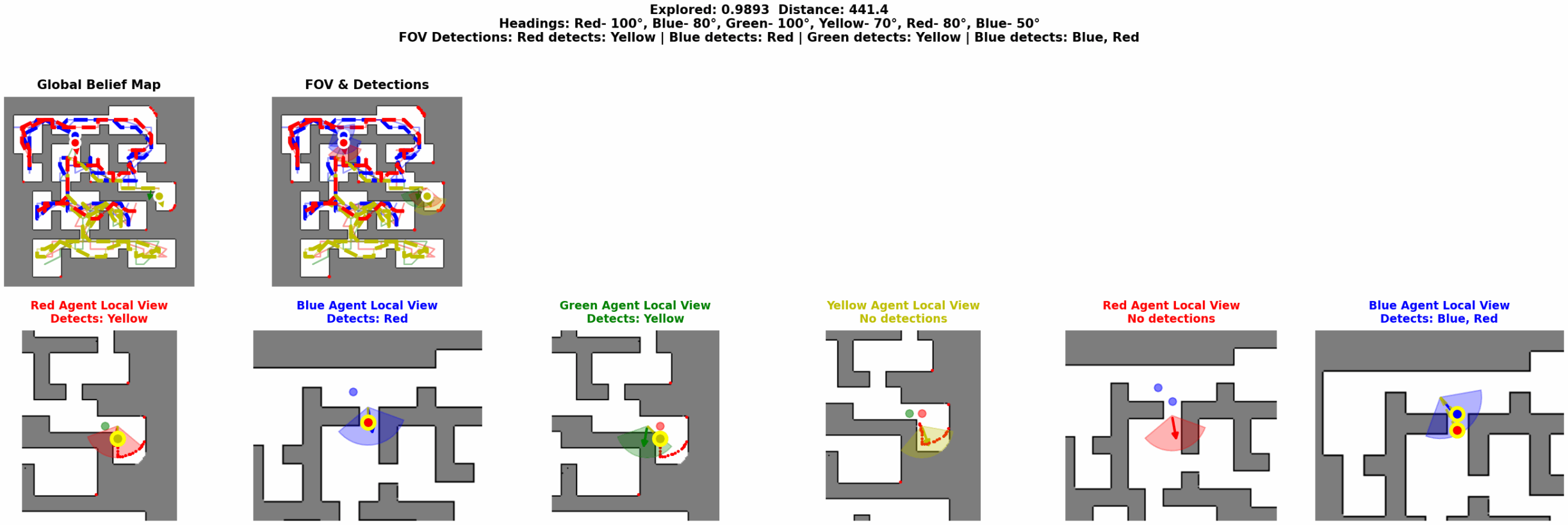
- Travel distance : 762.9



실험 결과

Zero Communication + Trajectory observation (Our method)

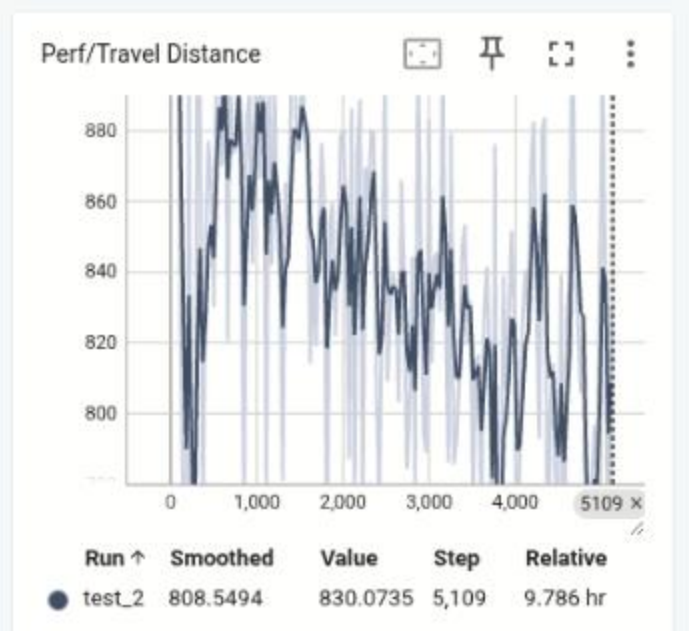
- Travel distance : 441.4



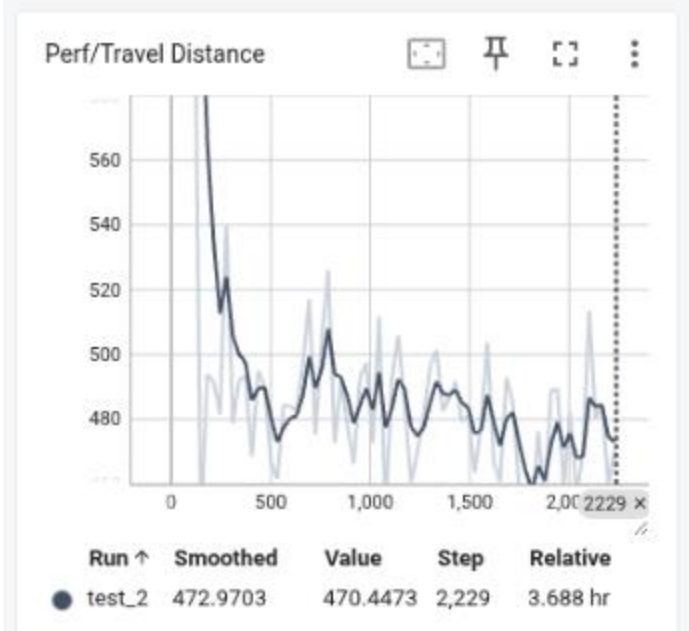
실험 결과

Zero communication / Zero communication + Trajectory

- Zero communication에서 학습했을 때 vs Zero communication + Trajectory를 encoding해 fusion했을 때의 distance 그래프



Zero comm. training



Zero comm. + Trajectory encoding → training

실험 결과

Perfect communication / Zero communication / Zero communication + Trajectory

- 99% coverage 와 90% coverage 두가지로 측정

Method	Zero Comm. (MARVEL)	Perfect Comm. (MARVEL)	Zero Comm. (Proposed)
Traj. Length	850.7(±351.2)	296.5(±49.2)	409.4(±67.4)
90% Coverage	605.5(±209)	240.8(±34.2)	340.9(±54.9)

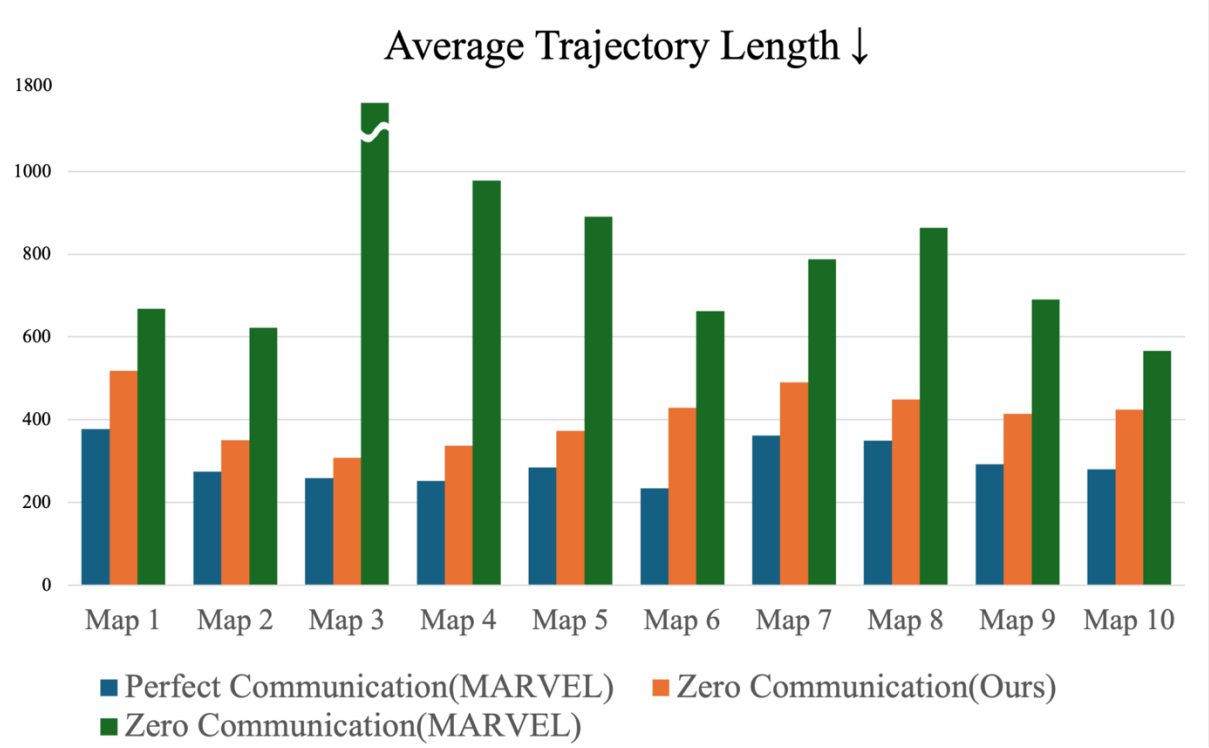
The comparison of the performance metrics

Metric	
Average max length	377.20
Maximum max length	1054.95
Minimum max length	286.38
Std max length	195.61
Average explored rate	0.99
Average success rate	0.92
Average distance to 0.9 explored	263.57
Std distance to 0.9 explored	12.74
Average overlap ratio	0.35
Std overlap ratio	0.16

실험 결과

Perfect communication / Zero communication / Zero communication + Trajectory

- 각 환경에 대한 average trajectory length를 10가지 다른 환경에서 실험하여 측정



Comparison of trajectory length in 10 environments

실험 결과

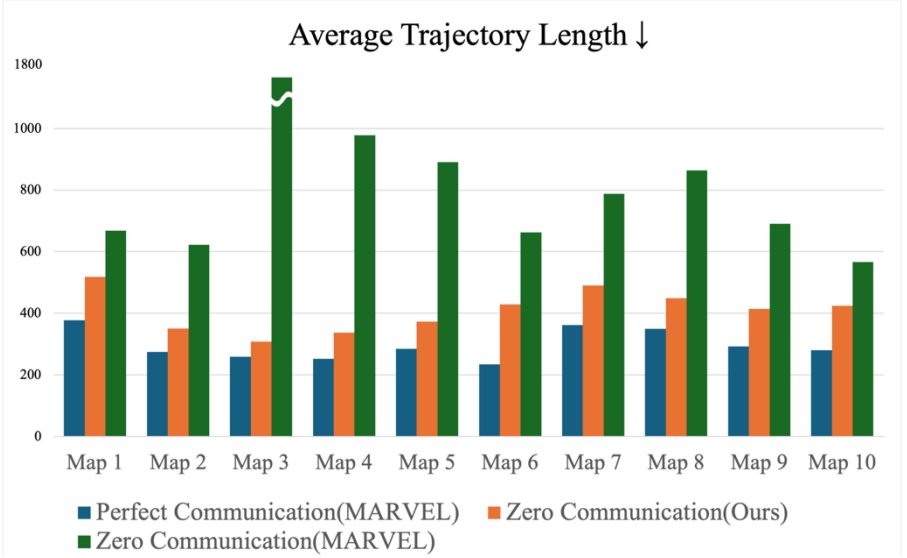
결과 해석

- Zero communication 대비 proposed method는 99% coverage에서 약 51.9% trajectory length 감소
- 기존 baseline인 MARVEL (Perfect communication)과 비교하면 여전히 길지만, zero comm ↔ perfect comm 사이의 성능 격차 중 약 79.7% 회복
- 즉, 어떠한 통신도 없이, 단지 FoV안에서 관측한 다른 로봇의 trajectory만으로도 통신 기반 collaborative exploration의 대부분을 확보 할 수 있음을 보여줌

Method	Zero Comm. (MARVEL)	Perfect Comm. (MARVEL)	Zero Comm. (Proposed)
Traj. Length	850.7(±351.2)	296.5(±49.2)	409.4(±67.4)
90% Coverage	605.5(±209)	240.8(±34.2)	340.9(±54.9)

The comparison of the performance metrics

- 10개 테스트 환경 각각에 대해 trajectory length를 plot했을 때
  - 모든 환경에서 our method가 zero comm보다 항상 짧고
  - 대부분의 환경에서 perfect comm(MARVEL)과 근접한 수준까지 성능 끌어올림



Comparison of trajectory length in 10 environments

Conclusion & Future work

Conclusion

- 기존 baseline인 MARVEL은 graph attention + frontier/orientation fusion + SAC 기반으로
- 이전의 classical한 방법들보다 안정적으로 짧은 궤적, 높은 success rate, 좋은 generalization을 보임

- **Trajectory Observation 확장**
  - MARVEL의 상태에 타 로봇 궤적이라는 새로운 observation에 대한 modality 추가
  - 통신 없이도 어디가 이미 탐사 완료되었는지, 다른 로봇이 어디로 향하는지 trajectory를 통해 간접적으로 파악
  - 이를 통해 중복탐사 회피, 영역 분할, frontier 방향 추론 등 emergent coordination을 보여줌
  - 결과적으로, zero communication 상황에서 perfect communication환경에 근접하는 수준까지 탐사 성능을 끌어올림

Future works

- 실제 환경으로의 deployment
- 3차원으로의 확장을 고려한 altitude, 로봇 간의 collision avoidance등 요소 추가설계

References

[1] J. Chiun, S. Zhang, Y. Wang, Y. Cao, and G. Sartoretti, “MARVEL: Multi-agent reinforcement learning for constrained field-of-view multi- robot exploration in large-scale environments,” in Proc. of IEEE Int. Conf. on Robotics and Automation, 2025.

[2] B. Zhou, H. Xu, and S. Shen, “Racer: Rapid collaborative exploration with a decentralized multi-uav system,” IEEE Trans. on Robotics, vol. 39, no. 3, pp. 1816–1835, 2023.

[3] J. Yu et al., “SMMR-Explore: Submap-based multi-robot exploration system with multi-robot multi-target potential field exploration method,” in Proc. of IEEE Int. Conf. on Robotics and Automation. IEEE, 2021.

[4] A. Bautin et al., “Towards a communication free coordination for multi- robot exploration,” in Proc. of National Conf. on Control Architectures of Robots, 2011.

[5] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, “Attention is all you need,” Advances in Neural Information Processing Systems, vol. 30, 2017.

[6] T. Haarnoja et al., “Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor,” in Proc. of Int. Conf. on Machine Learning, 2018, pp. 1861–1870.

감사합니다.