COMRAD: A Benchmark for Embodied Cooperative Multi-Agent Reinforcement Learning

Khoi H.B. Nguyen H.B.K.NGUYEN.HOANG.BAO.KHOI@STUDENT.TUE.NL
Dimitar Zhikov Zhekov D.Z.ZHEKOV@STUDENT.TUE.NL
Tristan Tomilin T.TOMILIN@TUE.NL
Eindhoven University of Technology, Netherlands

Abstract

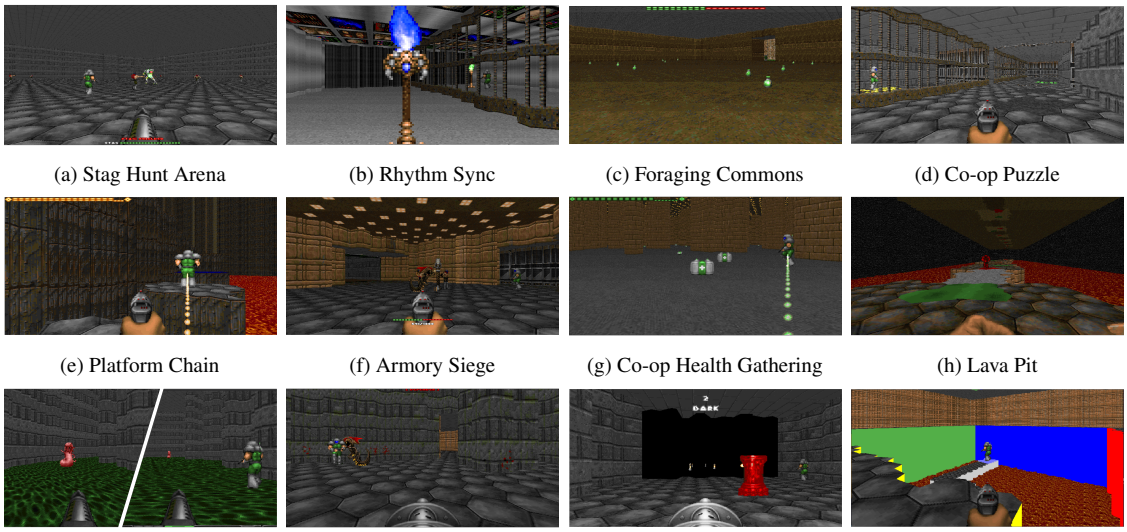
Benchmarks are central to the development and evaluation of multi-agent reinforcement learning (MARL) algorithms. As the cooperative MARL community has grown, two categories of evaluation environments have proven indispensable: low-dimensional feature-vector benchmarks that isolate algorithmic behavior in compact state spaces, and two-dimensional pixel-based benchmarks that rely on overhead visual observations. However, to bridge the gap to real-world embodied settings such as robotics and autonomous navigation, algorithms must handle high-dimensional, 3D egocentric visual complexity. While various 3D environments have been explored for vision-based RL, no existing platform simultaneously provides a standardized, purely cooperative multi-agent benchmark with 3D first-person observations and high-throughput simulation. To address this gap, we introduce **COMRAD**, a **CO**operative **M**ulti-Agent **R**einforcement Le**A**rning benchmark suite in **D**oom, featuring a diverse set of challenging scenarios spanning role asymmetry, temporal synchronization, and spatial navigation. To introduce within-scenario variability, we develop **DoomGen**, a procedural map generator that produces diverse layout configurations for every scenario. We integrate COMRAD with Sample Factory, a high-throughput asynchronous RL framework, and implement seven MARL baselines on top of it, reaching $\sim$25K frames per second during training. Our experiments show that COMRAD poses significant challenges for current CTDE methods, establishing visual cooperative MARL as an important open frontier.

1. Introduction

The development of reinforcement learning (RL) has been profoundly shaped by its benchmarks. In single-agent RL, the Arcade Learning Environment, OpenAI Gym, and MuJoCo enabled systematic comparison from pixels and continuous control, respectively [2, 3, 20]. In cooperative multi-agent reinforcement learning (MARL), SMAC and SMACv2 played a similar role for Centralized Training with Decentralized Execution (CTDE), giving the community a common substrate on which methods such as QMIX, MAPPO, and HAPPO could be compared [6, 8, 17, 18, 23]. Well-designed benchmarks, therefore, do not merely measure progress; they define the research frontier, frame the community's shared challenges, and create the conditions for reproducible, comparable science.

Yet benchmark coverage for strategic multi-agent learning from first-person vision remains thin. Most standard CTDE suites use low-dimensional state features or top-down observations, while existing visual environments either emphasize social dilemmas from overhead views or broader embodied tasks such as language grounding and household planning without reproducible cooperative control from pixels [1, 4, 9–16, 21]. This leaves a gap for studying learning dynamics in partially

© K.H. Nguyen, D.Z. Zhekov & T. Tomilin.



(a) Stag Hunt Arena (b) Rhythm Sync (c) Foraging Commons (d) Co-op Puzzle

(e) Platform Chain (f) Armory Siege (g) Co-op Health Gathering (h) Lava Pit

(i) Smart/Dumb Enemies (j) Ammo Carrier (k) Stealth Labyrinth (l) Lava Maze

Figure 1: In-game screenshots from all 13 COMRAD scenarios. Each scenario is presented in a 3D first-person egocentric view, requiring agents to coordinate solely based on their local visual observations.

observed cooperative games where synchronization, signaling, and role specialization must arise from egocentric local interaction.

We address this gap with **COMRAD**, a cooperative MARL benchmark suite built on ViZDoom [7]. COMRAD targets a controlled slice of strategic interaction: fully cooperative partially observed games in which agents must coordinate from egocentric RGB observations while remaining practical enough for multi-seed CTDE evaluation. The suite contains 13 scenarios spanning temporal commitment, resource stewardship, role asymmetry, grounded signaling, and combat coordination. Agents act in factorized discrete action spaces spanning navigation, interaction, and combat primitives, which keeps the benchmark focused on strategic coordination over low-level continuous control.

COMRAD makes three contributions. First, it introduces 13 embodied cooperative games together with **DoomGen**, a procedural layout generator that reduces map memorization and enables controlled generalization studies. Second, it provides a unified high-throughput evaluation stack covering seven CTDE baseline families: IPPO, MAPPO, HAPPO, IDQN, VDN, QMIX, and QPLEX. Third, it exposes a structured failure landscape at a common 100M-step frontier: no method family dominates, grounded signaling remains unsolved, and stronger centralized coupling is not uniformly beneficial in asymmetric coordination tasks.

2. COMRAD

COMRAD is a fully cooperative partially observed benchmark built on ViZDoom [7]. Each scenario is packaged as a self-contained WAD plus ACS script controlling geometry, game-tic logic, and objective bookkeeping. Agents connect to a synchronized multiplayer instance, receive egocentric

Table 1: Comparison of representative cooperative MARL benchmarks. COMRAD uniquely combines 3D egocentric observations, purely cooperative tasks, procedural layout generation, and integrated CTDE baselines in a single open-source suite (OSS). A standard CTDE baseline trained for 100M environment steps completes in approximately one hour on a single GPU.

Benchmark	**3D/Ego.**	**Vision**	**Coop.**	**ProcGen**	**CTDE Baselines**	**OSS**	N **agents**	**Scenarios**	**Year**
SMAC / SMACv2	✗	✗	✓	✗/ ✓	✓	✓	2–27	14 / ∞	2019/23
Google Research Football	✗	Opt.	✓	✗	✗	✓	2–11	11+	2019
Overcooked AI	✗	✗	✓	✗	✗	✓	2	5	2019
Melting Pot 2.0	✗	✓	Mixed	✗	✗	✓	2–16	256	2023
JaxMARL	✗	✗	✓	✗	✓	✓	2–10	8+	2024
MEAL	✗	✓	✓	✗	✓	✓	2–4	4	2025
XLand	✓	✓	Mixed	✓	✗	✗	2–16	∞	2021
Watch&Help	✓	✓	✓	✗	✗	✓	2	5	2020
TeamCraft	✓	✓	✓	✗	✗	✓	2	10	2024
COMRAD (Ours)	✓	✓	✓	✓	✓	✓	2–8	**13 /** ∞	2026

first-person observations, and act through ViZDoom's button interface, while a lightweight Python layer handles preprocessing, reward shaping, and integration with the training stack.

2.1. Environments and Scenarios

COMRAD provides 13 scenarios spanning several recurring strategic demands: commitment under partial observability, long-horizon resource stewardship, persistent role asymmetry, and grounded low-bandwidth signaling. Table 2 summarizes the benchmark suite, while Appendix C gives the full mechanics and reward definitions.

2.2. Observation and Action Spaces

Observations. COMRAD agents act from egocentric RGB observations resized to 128×72 pixels before entering the policy network. This resolution provides sufficient visual fidelity for agents to distinguish objects, teammates, and environmental features while remaining computationally tractable. Several scenarios additionally expose a local measurement vector (*Vision+Vec*) that contains scenario-specific information, such as health, ammo, progress counters, or positional hints, to accelerate learning. These measurements are strictly local to the observing agent and contain no direct information about teammates' states or positions.

Actions. ViZDoom's native button interface is exposed as a factorized discrete action space. Each scenario defines a Cartesian product over task-appropriate primitives spanning movement, rotation, interaction, combat, and weapon selection. This yields a single structured action per decision step. Action space size $|\mathcal{A}|$ ranges from 18 to 162 across the 13 scenarios. This design decouples cooperative strategy learning from low-level continuous control, making algorithmic differences in coordination the primary performance signal.

Rewards. Reward shaping is scenario-specific, but the reporting principle is uniform. Each task defines a benchmark score J that captures the cooperative outcome of interest, such as survival time, checkpoint progress, completed stages, or kills. We report J throughout because it is more

Table 2: Overview of the 13 COMRAD cooperative scenarios and their key properties. *Diff.* groups scenarios by difficulty tier: Easy (1), Medium (3), Difficult (5), and Very Difficult (7). N denotes supported team sizes; all scenarios support at least 2 agents. *Vis+Vec* indicates that egocentric RGB frames are supplemented with a strictly local measurement vector. $|\mathcal{A}|$ is the per-agent factorized discrete action space size. *Objective* is the task-level benchmark score reported throughout the paper.

Diff.	Scenario	N	Obs.	$\|\mathcal{A}\|$	Objective	Coordination Challenge
1	Stag Hunt Arena	2–4	Vision	54	Maximize stag kills	Agents must resolve the hare-versus-stag coordination dilemma under partial observability.
3	Rhythm Sync	2	Vis+Vec	54	Synchronize strict timings	Separated agents must activate switches in strict temporal windows without communication.
	Foraging Commons	2–4	Vis+Vec	27	Maximize survival time	Balance individual harvesting against replenishing a shared, limited resource pool.
	Co-op Puzzle	2	Vision	18	Completed puzzle pairs	Agents sequentially unlock gates for one another in a strict turn-based progression.
4	Platform Chain	2	Vision	54	Joint checkpoint progress	A physical tether constrains agents to maintain synchronized spatial proximity.
	Armory Siege	2–8	Vis+Vec	162	Maximize survival & kills	A team defends a core while rotating members to collect distant resources.
	Coop Health Gathering	2	Vision	27	Maximize survival time	Agents navigate a decaying health field together to maximize overlapping collection.
5	Lava Pit	2	Vision	36	Joint traversal	Agents alternate holding pressure plates to support each other's crossing over hazards.
	Smart Enemies	2–4	Vision	54	Maximize enemies kills	Enemies accelerate under concentrated fire, forcing agents to maintain dispersed positions.
	Dumb Enemies	2–4	Vision	54	Maximize enemies kills	Enemies flee solo agents but slow near groups, requiring cooperative pursuit patterns.
6	Ammo Carrier	2	Vis+Vec	36	Maximize survival time	A mobile carrier resupplies a stationary defender to maintain continuous defensive fire.
7	Stealth Labyrinth	2	Vision	54	Activate isolated relays	A torch-gunner pair survives shooting enemies under extreme asymmetric observability.
	Lava Maze	2	Vis+Vec	90	Maximize levels cleared	Form emergent communication between an active navigator and a remote information-providing signaler.

comparable across algorithm families than raw shaped return; full reward definitions are deferred to Appendix C.

3. Experiments

3.1. Baseline Algorithms

We evaluate seven CTDE algorithms spanning the two dominant cooperative MARL families. The on-policy actor-critic baselines are IPPO, MAPPO [23], and HAPPO [8]. The off-policy value-decomposition baselines are IDQN, VDN [19], QMIX [17], and QPLEX [22]. All are adapted to the same asynchronous training stack and evaluated under a shared protocol.

3.2. Baseline Results

We evaluate eight baseline configurations from seven CTDE families under a common 100M-step budget. Table 3 reports the final benchmark scores alongside the theoretical performance ceiling for

Table 3: Full baseline matrix at the common 100M-step frontier. Entries report mean ± 95% CI half-width over five independent runs with seeds $\{42, 68, 81, 97, 154\}$. **Ceiling** is the theoretical maximum score achievable under optimal play for each scenario.

Scenario	IPPO	MAPPO	HAPPO	IDQN	VDN	QMIX	QPLEX-D	QPLEX-Q	**Ceiling**
Stag Hunt Arena	17.55 ± 0.22	**25.66 ± 0.32**	2.58 ± 0.10	14.33 ± 0.18	17.96 ± 0.39	0.28 ± 0.06	1.49 ± 0.10	3.42 ± 0.20	36
Rhythm Sync	0.59 ± 0.05	**0.65 ± 0.05**	0.22 ± 0.02	0.00 ± 0.01	0.00 ± 0.01	0.00 ± 0.01	0.00 ± 0.01	0.00 ± 0.01	1.00
Foraging Commons	2355.64 ± 30.33	2510.93 ± 31.18	2199.89 ± 39.31	2190.70 ± 32.66	2327.13 ± 42.84	**2544.89 ± 46.77**	1905.26 ± 23.66	2345.75 ± 39.38	21000
Co-op Puzzle	3.03 ± 0.07	**3.88 ± 0.09**	2.23 ± 0.14	0.03 ± 0.01	0.07 ± 0.02	0.00 ± 0.01	0.00 ± 0.01	0.07 ± 0.03	5
Platform Chain	2.00 ± 0.02	**4.78 ± 0.19**	4.67 ± 0.18	0.68 ± 0.06	1.12 ± 0.08	0.97 ± 0.02	3.40 ± 0.22	0.97 ± 0.03	47
Armory Siege	16.71 ± 2.90	19.87 ± 2.32	-0.71 ± 1.07	10.97 ± 0.98	18.32 ± 1.46	**23.12 ± 3.83**	1.19 ± 0.89	14.13 ± 2.47	100
Co-op Health Gathering	**1104.90 ± 117.30**	553.96 ± 27.82	197.74 ± 7.98	164.20 ± 2.12	573.83 ± 27.28	166.43 ± 2.07	193.95 ± 5.10	267.78 ± 18.39	2100
Lava Pit	0.92 ± 0.01	**1.00 ± 0.01**	**1.00 ± 0.01**	0.92 ± 0.01	0.80 ± 0.05	0.97 ± 0.03	0.00 ± 0.01	**1.00 ± 0.01**	11
Smart Enemies	11.56 ± 0.86	**14.23 ± 0.65**	6.85 ± 0.38	3.30 ± 0.23	4.89 ± 0.38	0.64 ± 0.08	0.00 ± 0.01	0.65 ± 0.07	84
Dumb Enemies	**12.44 ± 1.93**	2.29 ± 0.20	1.03 ± 0.13	1.07 ± 0.11	2.76 ± 0.22	1.10 ± 0.15	0.00 ± 0.01	1.80 ± 0.16	90
Ammo Carrier	**7052.93 ± 483.17**	1362.73 ± 69.09	1340.99 ± 58.07	3836.13 ± 384.96	2542.51 ± 219.25	2379.46 ± 139.55	1530.98 ± 59.06	2080.97 ± 192.02	8400
Stealth Labyrinth	0.00 ± 0.01	0.00 ± 0.01	0.00 ± 0.01	0.00 ± 0.01	0.00 ± 0.01	**0.21 ± 0.01**	0.00 ± 0.01	0.00 ± 0.01	1.00
Lava Maze	0.00 ± 0.01	0.00 ± 0.01	0.00 ± 0.01	0.00 ± 0.01	0.00 ± 0.01	0.00 ± 0.01	0.00 ± 0.01	0.00 ± 0.01	6

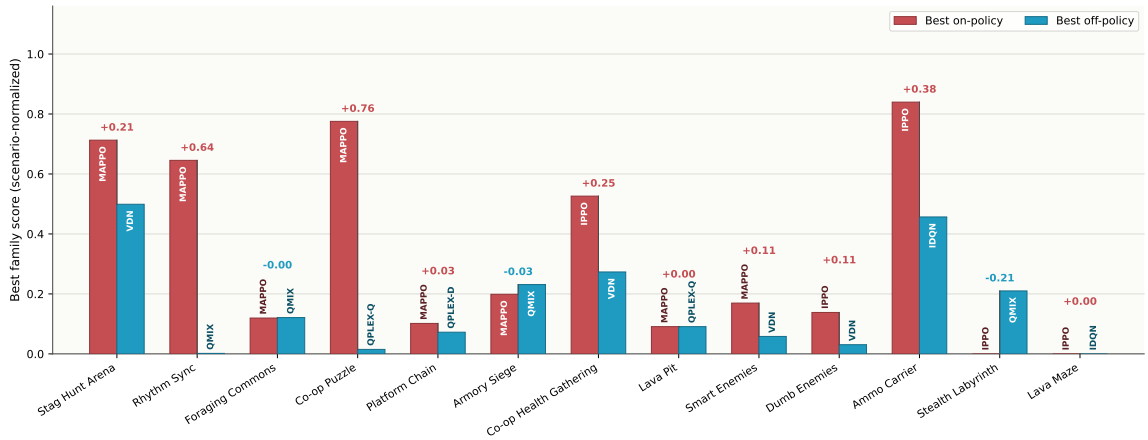


Figure 2: Scenario-normalized gap between the strongest on-policy and off-policy baseline. Each scenario is normalized by its natural score range before comparison.

each scenario. The main result is that no single family dominates, and COMRAD remains far from saturated.

Performance differences are structured by the type of strategic dependency each task induces. On-policy actor-critic methods are strongest on commitment-heavy synchronization problems such as *Stag Hunt Arena*, *Rhythm Sync*, and *Co-op Puzzle*, with MAPPO the most reliable overall baseline. Off-policy value-decomposition methods are stronger on long-horizon stewardship and defense, with QMIX leading *Foraging Commons* and *Armory Siege*.

The most informative result is a recurring failure landscape. *Lava Maze* remains unsolved by every method, and only QMIX achieves a non-zero score on *Stealth Labyrinth*. These tasks combine severe asymmetric information with grounded behavioral coordination, suggesting that embodied signaling and role-specialized partner modeling remain open problems. A second failure mode appears in *Ammo Carrier*, where decentralized IPPO (**7052.93**) substantially outperforms centralized baselines such as MAPPO (**1362.73**), indicating that stronger joint coupling is not universally beneficial in asymmetric logistics games.

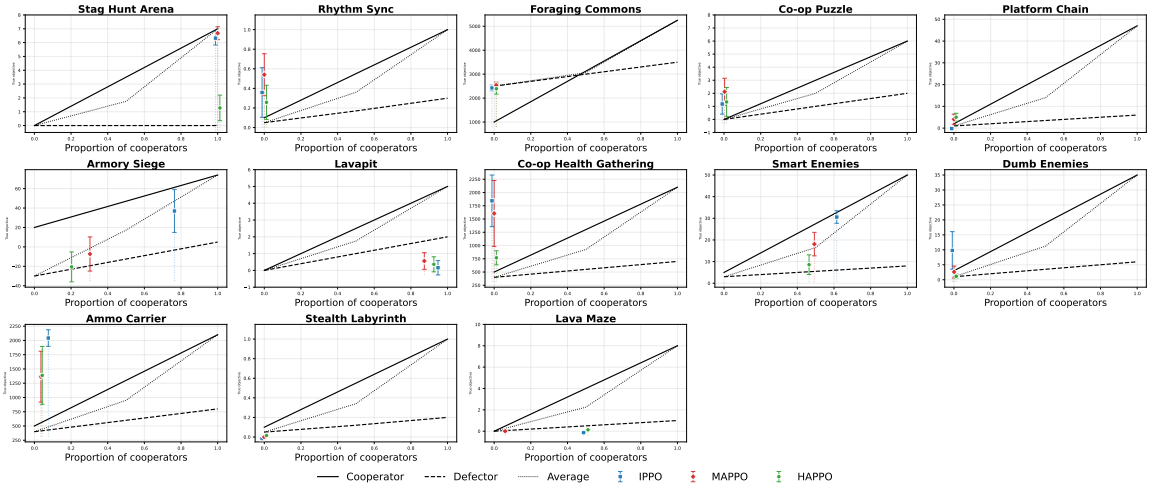


Figure 3: Schelling diagrams for all 13 scenarios. Solid and dashed black lines denote the theoretical payoffs for Cooperators and Defectors as a function of the proportion of cooperators in the population, computed analytically from each scenario's reward parameters. Colored markers show the empirical operating points of trained IPPO, MAPPO, and HAPPO policies, with vertical drop-lines indicating their position on the cooperation spectrum. All panels use the task true objective as the y-axis.

3.3. Case Study: Asymmetric Logistics

Ammo Carrier is especially informative because it is not merely hard; it reverses the expected ranking between decentralized and centralized methods, as the task assigns permanently differentiated roles. IPPO's large margin over MAPPO and HAPPO suggests that stronger joint coupling can interfere with stable role-specialized behavior when one agent must maintain an independent long-horizon routine. We therefore treat *Ammo Carrier* as evidence that centralized training is not uniformly helpful in cooperative games with delayed, role-specific credit assignment. Appendix A provides a trajectory-level case study and a fuller capability-oriented interpretation.

4. Social Dilemma Environments

To analyze how algorithms balance individual incentives and collective outcomes, we evaluate policies using a per-step cooperation classifier (detailed in Appendix C). We visualize the results as custom Schelling diagrams (Figure 3), mapping empirical cooperation rates from 50 evaluation episodes against theoretical payoff landscapes derived from the underlying reward parameters.

The resulting points show that algorithmic failure stems from different strategic breakdowns depending on the environment. In Stag Hunt Arena and Lava Pit, agents successfully adopt a cooperative posture, consistently targeting the stag over rabbits or holding support plates for partners, which drives the cooperation rate near 1.0. However, their ability to execute this intent varies. In Lava Pit, the resulting objective scores for all baselines remain far below the theoretical maximum, as policies struggle with the precise spatial synchronization required under visual occlusion. In Stag Hunt Arena, IPPO and MAPPO successfully approach the theoretical maximum; only HAPPO shows a similar execution failure, learning the cooperative intent but yielding scores well below optimal payoffs.

In contrast, Foraging Commons triggers a pure tragedy of the commons, with all three baselines collapsing to a near-zero cooperation rate and abandoning environmental cleanup entirely to harvest resources. This myopic defection caps their survival time at less than half of what sustained joint stewardship would permit, trapping the population in a suboptimal equilibrium.

The diagrams also highlight how the choice of training architecture shifts these equilibria. Armory Siege and Smart Enemies exhibit a clear performance gradient tied to centralization. Decentralized IPPO sustains much higher cooperation rates and corresponding objective scores than its centralized counterparts. As value function coupling increases from IPPO to MAPPO to HAPPO, agents increasingly default to uncoordinated defection. This indicates that centralized critics can actively disrupt the formation of dispersed or specialized cooperative roles.

Finally, we note that the per-step classification metric requires dense behavioral choices to form a stable population proportion. In scenarios driven by sparse, event-based milestones like Co-op Puzzle or Rhythm Sync, the markers sit near zero primarily as a measurement artifact. Setting aside these metric limitations, the broader empirical landscape confirms that current CTDE architectures cannot robustly maintain payoff-dominant cooperative conventions from raw egocentric vision without explicit communication channels.

5. Discussion and Limitations

COMRAD is designed to shift benchmark pressure away from low-dimensional cooperative control toward embodied partial-information games. The main empirical message is that this shift does not simply make every task uniformly harder. Instead, it reveals structured mismatches between standard CTDE inductive biases and different coordination mechanisms: commitment-heavy tasks favor on-policy methods, long-horizon stewardship favors value decomposition, and signaling plus severe role asymmetry remain weak across the board.

The benchmark makes deliberate scope choices. By studying cooperative games and mixed-motive incentives, it isolates learning dynamics inside fixed joint objectives from bargaining or mechanism design. Its benchmark scores are scenario-specific, which improves task fidelity but requires caution in cross-scenario aggregation. Finally, the procedural curriculum tooling is infrastructure for controlled experimentation, not a claim that curriculum learning is solved. We therefore view COMRAD as a benchmark for strategic cooperative learning under embodied partial information, and as a foundation for future extensions toward broader game-theoretic settings.

6. Conclusion

We introduced COMRAD, a benchmark for embodied cooperative MARL that pairs 13 first-person ViZDoom scenarios with procedural generation and seven integrated CTDE baselines. Across five-seed experiments at a common 100M-step frontier, COMRAD reveals a structured failure landscape: commitment-heavy synchronization favors on-policy methods, long-horizon stewardship favors value decomposition, and grounded signaling plus severe role asymmetry remain largely unsolved. We hope COMRAD provides a useful evaluation target for future work on learning dynamics and benchmark design in strategic multi-agent settings.

Impact Statement

COMRAD is intended as a research benchmark for cooperative multi-agent reinforcement learning from embodied visual input. The benchmark may help improve coordination, robustness, and data efficiency in multi-agent systems with applications in theoretical cooperative studies, assistive automation, and distributed control, but the same capabilities could also be adapted to safety-critical or adversarial settings. We therefore view the benchmark as infrastructure for controlled scientific evaluation, and we encourage downstream work to pair performance improvements with safety analysis and application-specific oversight.

Acknowledgement

We thank Lou Corto, Yue Peng, and Andrei Moise for their early contributions to this project and constructive feedback, we are thankful to the fruitful discussions we had. We also appreciate the TU/e Supercomputing Center for providing the infrastructure used in our experiments. Finally, we are grateful to all researchers/peers in Honors Artificial Intelligence program and anonymous reviewers for their thoughtful comments, which help improve the quality of our work.

References

[1] John P Agapiou, Alexander Sasha Vezhnevets, Edgar A Duéñez-Guzmán, Jayd Matyas, Yiran Mao, Peter Sunehag, Raphael Köster, Udari Madhushani, Kavya Kopparapu, Ramona Comanescu, DJ Strouse, Michael B Johanson, Sukhdeep Singh, Julia Haas, Igor Mordatch, Dean Mobbs, and Joel Z Leibo. Melting pot 2.0. *arXiv preprint arXiv:2211.13746*, 2022.

[2] Marc G. Bellemare, Yavar Naddaf, Joel Veness, and Michael Bowling. The arcade learning environment: An evaluation platform for general agents. *Journal of Artificial Intelligence Research*, 47:253–279, 2013.

[3] Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Wojciech Zaremba. Openai gym. *arXiv preprint arXiv:1606.01540*, 2016.

[4] Matthew Chang, Gunjan Chhablani, Alexander Clegg, Mikael Dallaire Cote, Ruta Desai, Michal Hlavac, Vladimir Karashchuk, Jacob Krantz, Roozbeh Mottaghi, Priyam Parashar, et al. Partnr: A benchmark for planning and reasoning in embodied multi-agent tasks. *arXiv preprint arXiv:2411.00081*, 2024.

[5] Allan Dafoe, Edward Hughes, Yoram Bachrach, Tantum Collins, Kevin R. McKee, Joel Z. Leibo, K. Larson, and Thore Graepel. Open problems in cooperative ai. *ArXiv*, abs/2012.08630, 2020. URL https://api.semanticscholar.org/CorpusID:229220772.

[6] Benjamin Ellis, Jonathan Cook, Skander Moalla, Mikayel Samvelyan, Mingfei Sun, Anuj Mahajan, Jakob Nicolaus Foerster, and Shimon Whiteson. SMACv2: An improved benchmark for cooperative multi-agent reinforcement learning. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2023. URL https://openreview.net/forum?id=5OjLGiJW3u.

[7] Michał Kempka, Marek Wydmuch, Grzegorz Runc, Jakub Toczek, and Wojciech Jaśkowski. Vizdoom: A Doom-based AI research platform for visual reinforcement learning. In *2016 IEEE Conference on Computational Intelligence and Games (CIG)*, pages 1–8, 2016. doi: 10.1109/CIG.2016.7860433.

[8] Jakub Grudzien Kuba, Ruiqing Chen, Muning Wen, Ying Wen, Fanglei Sun, Jun Wang, and Yaodong Yang. Trust region policy optimisation in multi-agent reinforcement learning. In *International Conference on Learning Representations (ICLR)*, 2022.

[9] Qian Long, Zhi Li, Ran Gong, Ying Nian Wu, Demetri Terzopoulos, and Xiaofeng Gao. Teamcraft: A benchmark for multi-modal multi-agent systems in minecraft. *arXiv preprint arXiv:2412.05255*, 2024.

[10] Siddharth Nayak, Adelmo Morrison Orozco, Marina Ten Have, Vittal Thirumalai, Jackson Zhang, Darren Chen, Aditya Kapoor, Eric Robinson, Karthik Gopalakrishnan, Brian Ichter, James Harrison, Anuj Mahajan, and Hamsa Balakrishnan. MAP-THOR: Benchmarking long-horizon multi-agent planning frameworks in partially observable environments. In *Multi-modal Foundation Model meets Embodied AI Workshop @ ICML2024*, 2024. URL https://openreview.net/forum?id=ZygZN5egzy.

[11] Siddharth Nayak, Adelmo Morrison Orozco, Marina Ten Have, Vittal Thirumalai, Jackson Zhang, Darren Chen, Aditya Kapoor, Eric Robinson, Karthik Gopalakrishnan, James Harrison, Brian Ichter, Anuj Mahajan, and Hamsa Balakrishnan. Llamar: Long-horizon planning for multi-agent robots in partially observable environments, 2025.

[12] Bassel Al Omari, Michael Matthews, Alexander Rutherford, and Jakob Nicolaus Foerster. Multi-agent craftax: Benchmarking open-ended multi-agent reinforcement learning at the hyperscale, 2025. URL https://arxiv.org/abs/2511.04904.

[13] Open Ended Learning Team, Adam Stooke, Feryal Behbahani, Wojciech M. Czarnecki, Marta Garnelo, Arthur Gretton, Arthur Guez, Jonathan Hamel, et al. Open-ended learning leads to generally capable agents. *arXiv preprint arXiv:2107.12808*, 2021.

[14] George Papadopoulos, Andreas Kontogiannis, Foteini Papadopoulou, Chaido Poulianou, Ioannis Koumentis, and George Vouros. An extended benchmarking of multi-agent reinforcement learning algorithms in complex fully cooperative tasks. *arXiv preprint arXiv:2502.04773*, 2025.

[15] Georgios Papoudakis, Filippos Christianos, Lukas Schäfer, and Stefano V. Albrecht. Benchmarking multi-agent deep reinforcement learning algorithms in cooperative tasks. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks (NeurIPS)*, 2021.

[16] Xavier Puig, Tianmin Shu, Shuang Li, Zilin Wang, Yuan-Hong Liao, Joshua B Tenenbaum, Sanja Fidler, and Antonio Torralba. Watch-and-help: A challenge for social perception and human-ai collaboration. *arXiv preprint arXiv:2010.09890*, 2020.

[17] Tabish Rashid, Mikayel Samvelyan, Christian Schroeder de Witt, Gregory Farquhar, Jakob Foerster, and Shimon Whiteson. QMIX: Monotonic value function factorisation for deep

multi-agent reinforcement learning. In *Proceedings of the 35th International Conference on Machine Learning (ICML)*, volume 80 of *PMLR*, pages 4295–4304, 2018.

[18] Mikayel Samvelyan, Tabish Rashid, Christian Schroeder de Witt, Gregory Farquhar, Nantas Nardelli, Tim G. J. Rudner, Chia-Man Hung, Philiph H. S. Torr, Jakob Foerster, and Shimon Whiteson. The StarCraft Multi-Agent Challenge. *CoRR*, abs/1902.04043, 2019.

[19] Peter Sunehag, Guy Lever, Audrunas Gruslys, Wojciech Marian Czarnecki, Vinícius Flores Zambaldi, Max Jaderberg, Marc Lanctot, Nicolas Sonnerat, Joel Z. Leibo, Karl Tuyls, and Thore Graepel. Value-decomposition networks for cooperative multi-agent learning based on team reward. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems (AAMAS)*, pages 2085–2087, 2018.

[20] Emanuel Todorov, Tom Erez, and Yuval Tassa. Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5026–5033. IEEE, 2012.

[21] Tristan Tomilin, Luka van den Boogaard, Samuel Garcin, Bram Grooten, Meng Fang, and Mykola Pechenizkiy. Meal: A benchmark for continual multi-agent reinforcement learning. In *Proceedings of the Datasets & Benchmarks Track at NeurIPS 2025*, 2025. URL https://arxiv.org/abs/2506.14990.

[22] Jianhao Wang, Zhizhou Ren, Terry Liu, Yang Yu, and Chongjie Zhang. QPLEX: Duplex dueling multi-agent q-learning. In *International Conference on Learning Representations (ICLR)*, 2021.

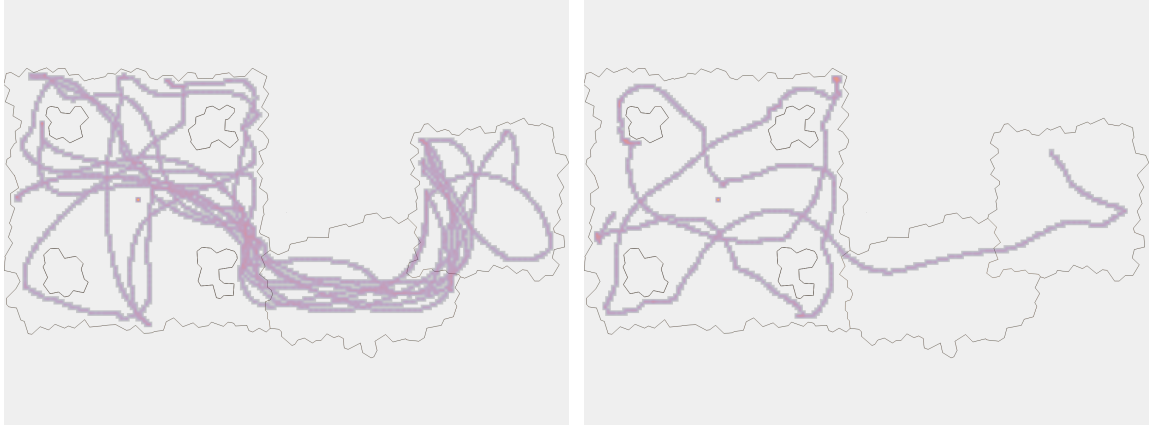
[23] Chao Yu, Akash Velu, Eugene Vinitsky, Jiaxuan Gao, Yu Wang, Alexandre Bayen, and Yi Wu. The surprising effectiveness of PPO in cooperative multi-agent games. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 35, pages 24611–24624, 2022.

Appendix A. Strategic Failure Analysis

Ammo Carrier produces the largest ranking reversal in the benchmark: decentralized IPPO substantially outperforms every centralized baseline in Table 3. The task assigns permanently differentiated roles. A stationary defender must hold the hub under enemy pressure, while a mobile runner must sustain a repeated depot-to-hub resupply loop. This is a plausible case where stronger centralized coupling can interfere with stable role-specialized behavior when one agent must maintain an independent long-horizon routine with delayed, role-specific payoff.

Following Dafoe et al. [5], the broader pattern can be interpreted along three recurring capability axes. *Stealth Labyrinth* and *Ammo Carrier* stress embodied partner understanding under durable asymmetry. *Lava Maze* isolates grounded low-bandwidth communication and remains unsolved at 100M steps. *Rhythm Sync* and *Platform Chain* stress temporal commitment without direct access to a partner's internal state. We treat these as interpretive summaries.

A.1. Behavioral Analysis: When Centralized Training Hurts



(a) IPPO. The runner repeatedly traverses the corridor between the depot region and the defender hub, consistent with a learned resupply loop.

(b) HAPPO. Movement does not organize into the depot-to-hub delivery pattern, consistent with failed resupply behavior.

Figure 4: Agent trajectory density heatmaps on *Ammo Carrier*, overlaid on the map automap, recorded from each method's best-checkpoint rollout. IPPO (left) shows dense occupancy along the hub-to-depot corridor, consistent with a learned resupply loo p. HAPPO (right) lacks this recurring delivery structure, suggesting it fails to learn the role-specialized logistics behavior required by the task. Because carrier spawn positions are randomized each episode, the figure should be interpreted as a representative behavioral diagnostic not as a proof of causality.

Aggregate benchmark scores reveal *which* algorithms perform best, but not *why* coordination succeeds or fails in a given scenario. We therefore study *Ammo Carrier*, the fixed-role asymmetric defense-and-resupply scenario that produces the largest reversal in the benchmark: decentralized IPPO substantially outperforms every CTDE baseline in Table 3.

Task structure. *Ammo Carrier* assigns agents to permanently differentiated roles: a stationary defender must hold the hub under enemy pressure, while a mobile runner must sustain a repeated

depot-to-hub resupply loop. The core challenge is not fine-grained handoff timing, but whether the runner learns a stable long-horizon logistics routine whose payoff is delayed and role-specific.

Behavioral signature of failure. To visualize this qualitatively, Figure 4 compares occupancy density heatmaps from representative rollout trajectories. IPPO concentrates movement along a recurring depot-to-hub corridor, consistent with a learned resupply cycle. HAPPO, by contrast, shows a much less regular pattern: the runner does not consistently maintain the loop, periodically abandoning the logistics routine to move toward the defender or revisiting depots in irregular patterns. The result is episodic ammo starvation and substantially lower survival times. This pattern is consistent with a breakdown in persistent role specialization, but is not by itself diagnostic of which component of the training setup causes that breakdown.

Interpretation. A reasonable explanation is that strict role asymmetry changes what effective coordination looks like under optimization. In tightly coupled joint tasks, centralized training and shared optimization pressure can help agents align on moment-to-moment joint behavior. In *Ammo Carrier*, however, the optimal joint policy also requires one agent to sustain an independent long-horizon logistics loop whose payoff is delayed and role-specific. Centralized updates may therefore introduce counterproductive coordination pressure that disrupts this independence: the runner's gradient updates are influenced by the defender's value signal in a way that interferes with the sustained logistics commitment the scenario demands. A decentralized update, by contrast, allows the runner to optimize its loop policy without cross-agent interference. We treat this as an informed interpretation instead of a causal proof; critic-scope, reward-sharing, and role-factorization ablations would be needed to isolate the effect.

Even so, this failure mode exposes a recurring Cooperative AI challenge: CTDE methods optimized for tightly coupled joint credit assignment may be poorly suited to *role-specialized understanding/trust* settings, where effective cooperation depends less on tighter joint coupling than on learning stable complementary behavioral contracts. Making this failure class behaviorally visible, beyond merely observable as lower return,, is one of the contributions of COMRAD, and we connect it to the broader Cooperative AI taxonomy in Section B.

Appendix B. Cooperative AI Capability Analysis

Following Dafoe et al. [5], we use COMRAD as an interpretive readout over three directly tested Cooperative AI capabilities: *understanding*, *communication*, and *commitment*, plus a narrower lens on *institutions*. The benchmark does not map one-to-one onto these bins; most tasks couple several capabilities at once, and the institutional lens is necessarily indirect. The purpose of this section is not to claim causal decomposition, but to map observed benchmark failures onto capability classes and identify where current methods systematically break down. Detailed scenario descriptions and demonstrations are provided in Appendix C.

Understanding. In the taxonomy of Dafoe et al. [5], understanding includes not only predicting the physical consequences of one's actions, but also anticipating other agents' behavior and the private factors that drive it. COMRAD probes this capability in an embodied, first-person form. *Stag Hunt Arena* and *Lava Pit* require partner inference under uncertainty, while *Ammo Carrier* and *Stealth Labyrinth* sharpen the demand through durable role asymmetry. The empirical pattern is consistent: strong baselines cope when partner behavior is structured and locally legible, but they struggle badly

when partner behavior must be inferred under extreme asymmetry and persistent occlusion, as in *Stealth Labyrinth*. We therefore read these failures primarily as limits in embodied partner modeling.

Communication. For Dafoe et al. [5], communication is the explicit exchange of information that improves coordination, with common ground and channel constraints playing central roles. COMRAD intentionally removes an explicit message channel, so the relevant question is narrower: can agents improvise communication through behavior or exploit weak environment-mediated cues? *Lava Maze* is the cleanest test: one agent has privileged route information and must convey it through constrained color-coded actions to a remote navigator. Every baseline remains at zero at 100M steps, which is the strongest evidence in the paper that low-bandwidth emergent communication from pixels remains unsolved for current CTDE methods. *Smart Enemies* and *Dumb Enemies* are easier because the signal is not produced by a partner but by the shared enemy dynamics induced by team positioning. The gap between those tasks suggests that reading a shared external signal is substantially easier than inventing and grounding a new one. Thus, some environment-mediated common ground is learnable, but partner-originated communication remains essentially unsolved.

Commitment. Commitment is the ability to make cooperative behavior credible, often through commitment devices or protocols that prevent profitable deviation. *Rhythm Sync* is the clearest example: spatially separated agents must repeatedly act within narrow time windows, so coordination depends on sustaining an internalized joint routine. Every off-policy baseline remains at or near zero, whereas MAPPO reaches **0.65**, suggesting that temporal commitment without direct mutual observation is a real bottleneck for current value-decomposition methods. *Stag Hunt Arena* reinforces the same message: MAPPO leads, but the task remains far from saturated, so even strong on-policy methods do not robustly maintain the payoff-dominant convention. *Platform Chain* is especially informative because the chain acts as a hardware-like commitment device in the sense discussed by Dafoe et al. [5]: the environment constrains unilateral deviation, yet substantial coordination difficulty remains. Taken together, current baselines do better when commitment is scaffolded by environmental structure, such as a chain or repeated rhythm, than when it must be maintained primarily through learned convention over long horizons.

Institutions. Institutions, by Dafoe et al. [5], are the rules, bargains, and incentive structures that shape the cooperative game itself. COMRAD does not test institution formation, bargaining, or mechanism design directly. Instead, *Foraging Commons* and *Armory Siege* place agents inside fixed resource-allocation rules that reward stewardship over myopic greed. QMIX's strong performance on both suggests that value decomposition can make partial progress on long-horizon collective credit assignment when the rule structure is fixed and legible. These scenarios, therefore, study behavior *within* institutional scaffolds not as institutions in the fuller Cooperative-AI sense; this is a deliberate scope boundary.

What COMRAD presently shows. The resulting picture spans all four Cooperative AI capabilities. Understanding under severe role asymmetry, communication through grounded behavioral signals, and commitment without structural support remain substantially weak. Institutions show a distinct pattern: value-decomposition methods can partially handle long-horizon collective credit assignment under fixed rule structures, making this the one capability in which current baselines show consistent progress. A recurring failure mode across the first three capabilities is what we informally call *role-asymmetric trust*: in the taxonomy of Dafoe et al. [5], this is better understood as a composite demand on understanding and commitment, sometimes exacerbated by the absence of communication,

instead of as a fifth standalone capability. Taken together, COMRAD distinguishes which cooperative capabilities are tractable in the egocentric visual setting from those that remain open challenges. The specific failure modes exposed by this benchmark provide clear directions for future algorithm and training-objective design, which we discuss in Section 5.

Appendix C. Environment Details and Reward Functions

Figure 5 organizes the cooperation properties tested by each scenario across four dimensions. This taxonomy serves primarily as an analytical tool to ensure that the benchmark suite covers a diverse space of general coordination ability.

		Stag Hunt Arena	Rhythm Sync	Foraging Commons	Co-op Puzzle	Platform Chain	Armory Siege	Co-op Health Gathering	Lava Pit	Smart Enemies	Dumb Enemies	Ammo Carrier	Stealth Labyrinth	Lava Maze
Game Theoretic Properties	Corresponding>Conflicting Interests	x	x		x	x	x	x	x	x	x	x	x	x
	Conflicting>Corresponding Interests			x										
	Asymmetric roles											x	x	x
	Near-perfect Information	x			x	x			x	x	x			
	Far-from-perfect Information		x	x			x	x				x	x	x
	Dyadic (1:1) interactions cause immediate rewards	x	x		x	x		x	x				x	x
	1:many interactions cause immediate rewards											x		
	many:many interactions cause immediate rewards			x			x			x	x			
	Social Dilemma (Mixed-Motive)			x										
	Pure Coordination Game	x	x											
	Equilibrium selection	x	x											
Game Design Properties	Puzzle-like		x		x	x			x				x	x
	Some physical state changes are irreversible	x	x	x	x	x	x	x	x	x	x	x	x	x
	Strict execution/timing thresholds		x			x			x					
	Procedural Generation / High variance			x		x		x	x					x
Reinforcement Learning Properties	Dynamic intra-episode learning		x							x	x			
	Delayed rewards (long after proximal cause)			x		x	x		x			x	x	
	Extreme Partial Observability (Dec-POMDP)												x	x
	Spatially-separated action/reward			x	x		x		x			x		
Properties of Potential Emergent Behaviors	Task Partitioning						x					x	x	x
	Resource Sharing			x			x					x		
	Implicit Signaling / Comm Protocol		x										x	x
	Convention Following	x	x		x								x	x
	Focal point (there is a clear default convention)	x	x											
	Turn-taking				x				x					
	Altruistic Sacrifice / Public Good			x	x				x					
	Reciprocity			x	x				x					
	Spatial Formation / Team Spacing					x		x		x	x			
	Flexibility					x	x			x	x			

Figure 5: Cooperation properties of COMRAD's 13 scenarios across four analytical dimensions. A mark (✗) indicates that the property is a defining characteristic of the scenario. This taxonomy confirms that the benchmark spans a diverse space of cooperative challenges without clustering around a single challenge type.

Throughout this appendix, r_t denotes the per-step training reward after wrapper shaping, and J denotes the scenario-level benchmark score reported in the evaluation tables. We distinguish episode-level randomization, which changes the task within a fixed generated WAD, from WAD-level configuration, which changes the underlying scenario instance used for curriculum learning or WAD swapping. The statistical validation of the ACS random source that drives this within-scenario stochasticity is reported separately in Appendix **??**. The figures in this section are representative in-game captures of the corresponding scenarios.

Stag Hunt Arena This scenario instantiates a two-player coordination dilemma between a reliable small solo payoff and a larger payoff that can be realized only through simultaneous commitment to a shared target. A high-level reward abstraction is

$$r_t = \alpha \Delta n_t^{\text{rabbit}} + \beta \Delta d_t^{\text{stag}} + \gamma \Delta n_t^{\text{stag}} - \lambda \Delta h_t^{-} - \mu \mathbf{1}\{\text{death}\},$$

where the benchmark score is $J = n_T^{\text{stag}}$, the number of stags killed. Within a fixed WAD, episode-level variability comes from repeated rabbit spawns over perimeter and interior candidate locations, while the stag follows a fixed center-spawn cycle.

Cooperation diagnostics. A step is classified as cooperative when the agent damages the stag without killing any rabbits, and defective when the agent kills rabbits without damaging the stag. This isolates stag-hunting commitment from individual rabbit harvesting. The Schelling payoff parameters used in Figure 3 are $(R, S, T, P) = (7, 0, 0, 0)$, reflecting that a stag kill is only possible under mutual coordination in range, while solo agents hunting rabbits contribute zero to the true objective. At the WAD level, the generator exposes the arena size, alcove layout, stag health and regeneration, the cooperative vulnerability radius, rabbit spawn timing and cap, and the reward magnitudes attached to solo and cooperative outcomes.

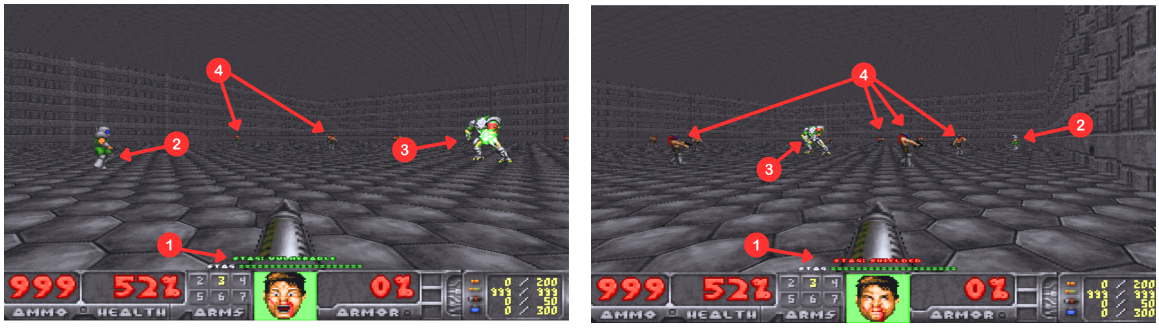


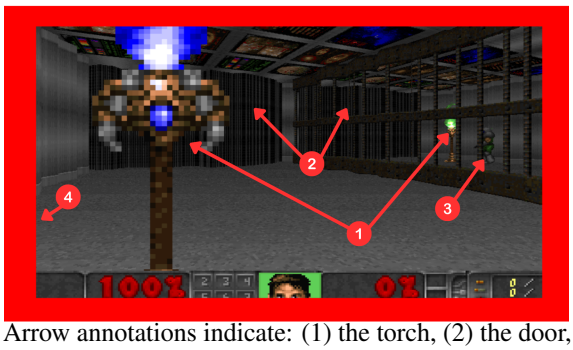
Figure 6: *Stag Hunt Arena*. Two representative views of the coordination dilemma. The arrow annotations indicate: (1) the boss-level stag, which is shielded when the players are too far apart and becomes damageable when they stay sufficiently close to one another; (2) the second player; (3) the boss-level stag; and (4) the small stags.

Rhythm Sync Rhythm Sync is a temporal coordination task in which two separated agents must press paired switches within a shared tolerance window, with unilateral activation causing immediate failure. Timing can be done through a shared periodic visual cue. A concise abstraction is

$$r_t = \alpha \Delta c_t - \lambda \mathbf{1}\{\text{mistimed activation}\} + \beta \mathbf{1}\{\text{full completion}\},$$

and the benchmark score is $J = c_T$, the number of stages completed. Within a fixed WAD, each episode resamples switch positions, resamples the cue jitter used in the harder stages, and alternates the cue owner in the single-observer stages according to an episode-specific offset. At the WAD level, the generator controls the number of stages, the split between regular, jittered, and single-observer stages, the beat period and tolerance schedule, and the room and switch geometry.

Cooperation diagnostics. A step is cooperative when a stage is successfully completed or a failure occurs while the agent is within the switch activation range, and defective when a failure occurs while the agent is out of range. Since both signals fire only on sparse event boundaries, the per-step cooperation ratio is uninformative; we treat the Schelling position as a measurement artifact. The Schelling parameters are $(R, S, T, P) = (1.0, 0.1, 0.3, 0.05)$, where full coordination yields one unit of normalized completion and isolated attempts contribute modestly.



Arrow annotations indicate: (1) the torch, (2) the door, (3) the second player, and (4) the cue.

The player on the right is waiting for the player on the left to run up to the torch.

Figure 7: *Rhythm Sync*. Two representative views of the temporal synchronization task.

Foraging Commons Foraging Commons is a shared-resource survival task in which harvesting improves short-term health but depletes the underlying regeneration process, while cleanup restores the commons at an opportunity cost. The shaped reward can be summarized as

$$r_t = \alpha \Delta n_t^{\text{harvest}} + \beta \Delta n_t^{\text{cleanup}} - \lambda \mathbf{1}\{\text{starvation}\},$$

where the benchmark score is survival length $J = T$. Within a fixed WAD, each episode repeatedly samples harvest pickups from random candidate locations based on the current commons state, so the spatial distribution of resources changes over time even when the geometry remains unchanged. At the WAD level, the generator controls the field and cleanup-zone sizes, their separation, the number of agents, the initial regeneration rate, the degradation and restoration dynamics, the health drain, and the density of candidate spawn spots.

Cooperation diagnostics. A step is cooperative when the agent performs a cleanup action without simultaneously harvesting, and defective when the agent harvests without cleaning up. This separates resource stewardship from myopic individual consumption. The Schelling parameters are $(R, S, T, P) = (5250, 1000, 3500, 2500)$, capturing the tragedy-of-the-commons structure where defection always yields a higher individual baseline but joint cooperation maximizes collective survival.

Co-op Puzzle Co-op Puzzle is a paired-lane progression task in which agents alternately unlock one another's path through relay plates, with periodic stages that require simultaneous occupancy on both sides. A compact abstraction is

$$r_t = -\eta + \alpha \Delta z_t + \beta \Delta p_t + \gamma \mathbf{1}\{\text{success}\} - \lambda \mathbf{1}\{\text{timeout}\},$$

where z_t is the team-max zone index, p_t is the number of completed zone pairs, and the benchmark score is $J = p_T$. Within a fixed WAD, each episode randomizes which candidate plate is active in each relay zone, while the synchronization zones remain determined by the designed stage schedule. At the WAD level, the generator controls the number of zones, the length and height of each lane, the number and size of candidate plates, the spacing of synchronization stages, and the visual textures used to mark active affordances.

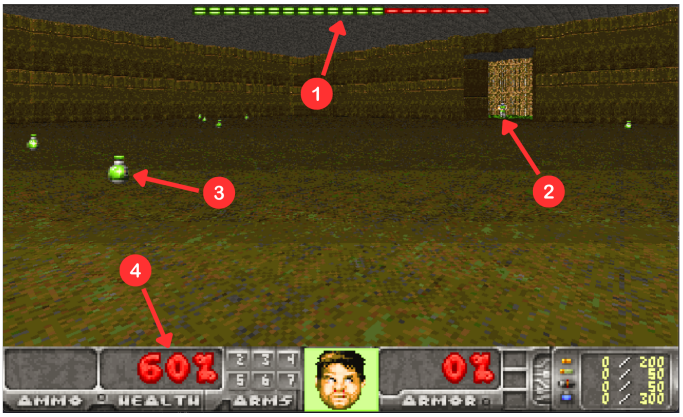


Figure 8: *Foraging Commons*. Arrow markers indicate: (1) the shared spawn-rate bar for harvestable health pots, (2) the second agent standing in the Cleanup Station to restore spawn rate, (3) a harvestable health pot that restores health and yields a small reward, and (4) the agent's own health, which decreases over time. The shaped reward provides a local incentive for both cleanup and harvesting, while the benchmark score is survival time. Policies must therefore balance restoring the commons against immediate collection without over-harvesting greedily.

Cooperation diagnostics. A step is cooperative when the agent makes progress on its own zone, and defective when the agent is neither making progress nor at success. Since defect fires on most idle steps, the per-step ratio is heavily skewed toward defection. The Schelling parameters are $(R, S, T, P) = (6, 0, 2, 0)$ for the six possible zone pairs, where mutual cooperation completes all pairs, while solo effort yields none.

The agent on the left is holding the plate for the agent on the right, which opens the door in the right hallway.

The agent on the right is holding the plate to open the door for the agent on the left.

Figure 9: *Co-op Puzzle*. Two representative views of the alternating door-and-plate coordination structure.

Platform Chain Platform Chain is a tethered vertical traversal task requiring coordinated movement to accumulate monotonic joint progress while avoiding chain overstretch and falls. The reward structure can be abstracted as

$$r_t = \alpha \Delta q_t - \beta \Delta b_t - \gamma d_t - \lambda \mathbf{1}\{\text{death}\},$$

where q_t is the best joint progress level, b_t counts new chain-break events, d_t counts dragged links, and the benchmark score is $J = \lfloor q_T \rfloor$. Within a fixed WAD, each episode resamples the safe opening length, the per-level motion phase, and the hazard variant assigned to most later checkpoints, so memorizing a single timing pattern is insufficient. At the WAD level, the generator controls the route length, height spacing, map radius, inter-platform spacing, lava damage, chain length, maximum player count, and the number of moving platform hazards.

Cooperation diagnostics. A step is cooperative when the agent advances its own checkpoint, and defective when a chain-break event occurs. Since checkpoint advances are sparse events, the per-step cooperation ratio remains near zero regardless of actual coordination quality. The Schelling parameters are $(R, S, T, P) = (47, 2, 6, 1)$, corresponding to the maximum joint checkpoint progress under the default 48-level configuration.

The two agents are jumping to the next platform together.

The chain is red, meaning one agent is being dragged.

Figure 10: *Platform Chain*. Two representative views of coordinated movement under a tether constraint.

Armory Siege Armory Siege is a defensive combat task in which the team protects a central core while rotating through specialized weapon, ammunition, and medical rooms whose access is periodically gated. A high-level abstraction is $r_t = \alpha\Delta k_t + \beta\Delta h_t^+ + \gamma\Delta a_t^+ + \xi\Delta w_t^+ - \lambda\Delta c_t^- - \mu\mathbf{1}\{\text{core destroyed}\} - \nu\mathbf{1}\{\text{agent death}\}$, where k_t denotes kills, h_t^+ health recovery, a_t^+ ammunition pickup, w_t^+ weapon acquisition, and c_t^- core damage. The benchmark score in this scenario is the cumulative shaped episodic return $J = \sum_t r_t$, so it should be interpreted only within Armory Siege. Within a fixed WAD, each episode randomizes enemy wave composition, wave entry side, and the exact spawn positions of pickups within the supply rooms. At the WAD level, the generator controls the distance from the hub to the supply rooms, corridor width, door timing, core health, enemy difficulty, and the maximum number of cooperative players.

Cooperation diagnostics. A step is cooperative when the agent is actively engaged in defending the core, and defective when the agent is not engaged while enemies are present. This distinguishes core defense from opportunistic looting. The Schelling parameters are $(R, S, T, P) = (74, 20, 5, -30)$, reflecting that coordinated defense yields strong collective returns, while defecting to loot alone while enemies attack the core produces strongly negative outcomes.

Co-op Health Gathering Co-op Health Gathering is a cooperative survival task in a toxic maze with a shared tether, where the team must balance exploration, collection, and chain management

Enemies are attacking the core.

An agent kills an enemy.

Figure 11: *Armory Siege*. Two representative views of the cooperative base-defense scenario.

under persistent environmental damage. A concise abstraction is $r_t = \alpha\Delta h_t^+ - \beta\Delta h_t^- + \gamma\Delta m_t - \lambda\Delta s_t - \mu\mathbf{1}\{\text{death}\}$, where m_t is the number of newly collected medical kits, s_t counts tether overstretch events, and the benchmark score is survival length $J = T$. Within a fixed WAD, health kits are repeatedly spawned at random candidate positions, so the immediate routing problem changes over time, even when the maze is fixed. At the WAD level, the generator controls the map radius, tether length, toxic-floor damage, the maximum number of available health kits, maze density, and the maximum number of players.

Cooperation diagnostics. A step is cooperative when the agent picks up a health kit, and defective when a tether overstretch occurs. Both signals are sparse event markers. The Schelling parameters are $(R, S, T, P) = (2100, 500, 700, 400)$, reflecting the survival horizon where joint health collection enables sustained survival while solo efforts lead to earlier death.

The chain is red, so one agent is being dragged.

Both agents cooperate to gather health.

Figure 12: *Co-op Health Gathering*. Two representative views of joint survival under tether and health-pressure constraints.

Lava Pit Lava Pit is a reciprocal bridge-handoff task in which one agent must press a plate to keep the bridge stays on while the other crosses, after which the roles invert before joint progress can continue. A compact abstraction is

$$r_t = -\eta + \alpha\Delta q_t + \sum_{m=1}^{4} \beta_m \Delta b_{m,t} + \gamma\mathbf{1}\{\text{success}\} - \lambda\mathbf{1}\{\text{failure}\} - \mu u_t,$$

where q_t is the best joint platform index, $b_{m,t}$ are ordered frontier milestones within the current handoff, u_t is the number of unfinished bridges at termination, and the benchmark score is $J = q_T - 1$, the number of jointly cleared bridges. Within a fixed WAD, later bridges resample the active support plates from safe candidate sets on every episode, while the earliest bridges remain intentionally deterministic to reduce the initial exploration burden. At the WAD level, the generator controls the number of platforms, the platform and gap ranges, the route's lateral curvature, the number of easier-opening bridges, the lava damage, and the optional runtime timing jitter.

Cooperation diagnostics. A step is cooperative when the agent advances its own bridge progress, and defective when a platform gap opens without the agent making progress. This separates active bridge-building from passive waiting. The Schelling parameters are $(R, S, T, P) = (5, 0, 2, 0)$, where reciprocal handoff coordination clears all five bridges, while a solo agent makes no progress.

One agent is holding the plate so the bridge stays on.

The other agent is not holding the plate, so the bridge collapses and the agent cannot cross.

Figure 13: *Lava Pit*. Two representative views of the reciprocal bridge-handoff mechanic.

Smart Enemies Smart Enemies places the team in a shared toxic arena against fleeing enemies that become harder to finish when several players converge on the same target. A high-level abstraction is

$$r_t = \alpha \Delta k_t + \beta \Delta \mathrm{hit}_t + \gamma \Delta h_t^+ - \lambda \Delta h_t^- - \mu \Delta f_t - \nu f_t - \rho \Delta a_t^-,$$

where f_t measures how many enemies are currently in the strengthened fast state, and the benchmark score is $J = k_T$, the number of kills. Within a fixed WAD, enemy spawn locations are sampled continuously from a pool of candidate spots, and the live enemy state changes online as a function of local player density. At the WAD level, the generator controls the arena radius, obstacle count, and obstacle layout seed, the enemy cap, the spawn interval, the density threshold radius, and the number of players.

Cooperation diagnostics. A step is cooperative when a kill occurs while no enemies are in the fast (strengthened) state, and defective when enemies transition from slow to fast. Cooperation therefore reflects kills achieved before concentrated fire triggers the enemy speed-up mechanic. The Schelling parameters are $(R, S, T, P) = (50, 5, 8, 3)$, where dispersed mutual firing yields maximum kills before enemies accelerate.

Dumb Enemies Dumb Enemies uses the same arena family as Smart Enemies, but reverses the density-response rule so that coordinated pressure makes fleeing enemies easier to corner. The reward

An agent shoots an enemy.

Agents stay together to hunt an enemy.

Figure 14: *Smart Enemies*. Two representative views of density-dependent cooperative hunting.

abstraction is

$$r_t = \alpha \Delta k_t + \beta \Delta \mathrm{hit}_t + \gamma \Delta h_t^+ - \lambda \Delta h_t^- + \mu \Delta s_t + \nu s_t - \rho \Delta a_t^-,$$

where s_t measures how many enemies are currently in the slowed state, and the benchmark score is again $J = k_T$. Within a fixed WAD, enemy spawns are resampled continuously from random candidate locations, while the slow-state dynamics are induced online by the players' relative positions. At the WAD level, the configurable factors match Smart Enemies: arena size, obstacle layout and count, enemy cap, spawn interval, proximity radius, and player count.

Cooperation diagnostics. A step is cooperative when a kill occurs while slow enemies are present, and defective whenever any fast enemies exist. Since fast enemies are present throughout most of the episode, the defect signal fires on nearly every step, making the per-step cooperation ratio near-zero regardless of actual policy quality. The Schelling parameters are $(R, S, T, P) = (35, 3, 6, 1)$, where coordinated grouping slows enemies and yields the highest kill count.

Figure 15: *Dumb Enemies*. An agent is about to shoot an enemy without the presence of the other agent.

Ammo Carrier Ammo Carrier is a fixed-role asymmetric defense task in which one immobilized defender protects the hub while one or more runners maintain the defender's ammunition supply

under enemy pressure. A high-level abstraction is role-dependent:

$$r_t^{\text{def}} = \alpha \Delta k_t + \beta \Delta d_t - \lambda \mathbf{1}\{\text{death}\}, \qquad r_t^{\text{run}} = \gamma \Delta p_t + \delta \omega_t \Delta u_t - \xi \Delta h_t^- - \lambda \mathbf{1}\{\text{death}\},$$

where Δu_t counts completed deliveries, p_t counts useful pickup events, and the weight ω_t increases when defender ammunition is scarce or enemy pressure is high; the benchmark score is survival length $J = T$. Within a fixed WAD, each episode resamples runner spawn locations, ammunition pack spawns, and enemy wave composition and entry points. At the WAD level, the generator controls the map radius, the number of peripheral depots, the enemy density, and the overall layout seed.

Cooperation diagnostics. The classification is role-dependent. For the defender, a step is cooperative when actively engaged in combat and defective when idle while enemies are present and ammunition is available. For the runner, a step is cooperative upon completing an ammunition delivery and defective when the defender is low on ammunition without a delivery occurring. This captures role-specific cooperative commitment. The Schelling parameters are $(R, S, T, P) = (2100, 500, 800, 400)$, reflecting the survival horizon where both roles fulfill their duties versus scenarios where one or both abandon their specialized function.

The defender is shooting an enemy while the runner agent collects ammunition.

The shooter agent is about to aim at an enemy.

Figure 16: *Ammo Carrier*. Two representative views of the fixed-role defense-and-resupply interaction.

Stealth Labyrinth Stealth Labyrinth is an asymmetric illumination-and-combat task in which a non-attacking scout reveals rooms for a visually dependent gunner, and combat rooms activate sequentially only after earlier rooms are cleared. A concise abstraction is

$$r_t = \alpha \Delta n_t^{\text{room}} + \beta \Delta k_t - \lambda \Delta h_t^{\text{team},-} - \mu \mathbf{1}\{\text{failure}\},$$

with an extra first-room discovery bonus in the default setting; the benchmark score is

$$J = \frac{\text{enemies destroyed}}{\text{total enemies}}.$$

Within a fixed WAD, the scenario is essentially deterministic once the episode starts: the active room order is fixed by progression, so the primary variability is behavioral not procedural. At the WAD level, the generator controls the number of branches, rooms per branch, room and corridor geometry, the enemy count range per room, turret health, and projectile parameters, and the easier first-room settings that reduce initial exploration difficulty.

Cooperation diagnostics. A step is cooperative when the agent makes a combat or exploration contribution while both agents are alive, and defective when the agent is dead or takes damage without securing a kill. This rewards joint survival and coordinated room-clearing. The Schelling parameters are $(R, S, T, P) = (1.0, 0.1, 0.2, 0.05)$, normalized to the fraction of total enemies destroyed, where joint exploration clears all rooms and isolated attempts fail.

Figure 17: *Stealth Labyrinth*. The hallway is dark, and the shooter agent is waiting for the runner agent to go first and light up the room.

Lava Maze Lava Maze is an asymmetric communication task in which one player navigates a dynamically carved safe path over lava while another player observes from above and sends color-coded guidance signals. A high-level abstraction is

$$r_t = -\eta + \alpha\Delta\ell_t + \beta\Delta d_t^{\text{path}} - \lambda\Delta h_t^- - \mu\mathbf{1}\{\text{signal use}\} - \nu\mathbf{1}\{\text{death}\} + \kappa\Delta d_t^{\text{guided}},$$

where ℓ_t is the number of completed maze levels and the benchmark score is $J = \ell_T$. Within a fixed WAD, every episode samples a fresh start location, and each successful level completion immediately generates a new maze instance whose size grows until it reaches the configured maximum. At the WAD level, the generator controls the physical grid size, cell size, initial and maximum maze size, and the severity of lava punishment.

Cooperation diagnostics. The classification is role-dependent. For the signaler, a step is cooperative when a guidance signal is sent and defective when no signal is sent despite prior level progress. For the navigator, a step is cooperative when maze progress is made and defective when progress stalls while a signal is being sent. This isolates productive signaling from disengagement. The Schelling parameters are $(R, S, T, P) = (8, 0, 1, 0)$, where effective signaling-and-navigation solves eight maze levels, while unguided or ignored signals yield none.

The navigator is at the beginning of the maze, while the observer agent can see everything from above.

The observer agent is shooting to signal going right, matching the yellow color.

Figure 18: *Lava Maze*. Two representative views of the asymmetric navigation-and-signaling task.

Camera-ready version

- Add author name and acknowledgement.
- Added Appendix C, which contains a more detailed description and reward functions for scenarios in the benchmark.
- Add Section 4 an ablation study and schelling diagrams visualizing the relationship between individual incentives and collective outcomes, for tasks that involve social dilemma.