Strategic Audit of 14-Dimensional Mission Embeddings for Goal-Conditioned Tactical Reinforcement Learning

1. Executive Summary

This report constitutes a rigorous technical audit of a proposed 14-dimensional (14-d) mission embedding designed for a goal-conditioned Reinforcement Learning (RL) system. The analysis evaluates the embedding's efficacy in facilitating robust policy convergence, particularly within the high-fidelity demands of tactical spatial navigation, multi-agent coordination, and complex objective fulfillment. The audit is grounded in the "Geometric Simplicity Principle" and advanced representation learning theories, asserting that the performance of an RL agent is bounded not merely by the learning algorithm (e.g., PPO, SAC) but fundamentally by the geometry and semantic structure of its observation space.[1]

The review identifies critical vulnerabilities in the baseline 14-d design. The reliance on raw Cartesian coordinates, scalar angular representations, and likely integer-based task identifiers introduces significant non-stationarity, manifold discontinuities, and optimization instability. Furthermore, the embedding suffers from a "semantic vacuum" regarding tactical affordances; it lacks the requisite relational features—such as relative orientation, line-of-fire availability, and team formation integrity—necessary for emergent behaviors like flanking or suppressive fire.[3] The fixed-dimensional nature of the vector also imposes severe scalability limits, failing to account for the permutation invariance required when dealing with variable numbers of teammates and adversaries.[5]

Drawing upon cross-disciplinary research from cognitive neuroscience—specifically the integration of egocentric and allocentric reference frames [7]—and state-of-the-art developments in Vision-Language-Action (VLA) models [9], this report proposes "Embedding V2." This redesigned architecture transitions from a flat, heterogeneous vector to a structured **Tactical Tensor** system. Key recommendations include:

1. **Geometric Normalization:** Adopting spherical and polar coordinate systems to enforce rotational invariance and eliminate singularities.
2. **Permutation-Invariant Processing:** Utilizing attention-based encoders (Transformers) to process entity lists, enabling scalable coordination.
3. **Task Semantic Embedding:** Replacing discrete IDs with continuous task embeddings derived from language or goal-conditioned encoders.
4. **Explicit Tactical Feature Engineering:** Integrating calculated metrics for "Enfilade Index" and "Suppression Density" directly into the observation stream.

An accompanying ablation study is outlined to empirically isolate the contributions of these architectural shifts, providing a roadmap for validating the V2 design's superior generalization and coordination capabilities.

---

2. Theoretical Framework: The Geometry of Representation

To rigorously audit the 14-d embedding, we must first establish the theoretical imperatives for state representation in deep reinforcement learning. A state representation $\phi(s)$ maps the environmental state s to a vector space $\mathbb{R}^d$. The "Manifold Hypothesis" suggests that high-dimensional data (like tactical scenarios) lie on lower-dimensional manifolds embedded within the input space.[11] An effective embedding must reflect the intrinsic geometry of this manifold to allow the policy network to learn smooth, continuous control laws.

2.1 The Egocentric-Allocentric Dichotomy in Spatial Cognition

The primary failure mode of naive RL embeddings is the conflation of reference frames. Cognitive science establishes that biological navigation relies on two distinct parallel systems:

- **Egocentric (Body-Centered):** This framework encodes spatial information relative to the organism's current position and orientation (e.g., "The target is 30 degrees to my left"). It is processed in the parietal cortex and is essential for immediate sensorimotor transformations and continuous control.[7]
- **Allocentric (World-Centered):** This framework encodes spatial relationships independent of the observer, relying on stable environmental landmarks (e.g., "The objective is in the North sector"). It is associated with the hippocampal formation and place cells, supporting long-term planning and cognitive mapping.[7]

In the context of the 14-d embedding, mixing absolute coordinates (Allocentric) with local body velocities (Egocentric) without explicit structural separation forces the neural network to internally compute the necessary coordinate transformations. While theoretically possible for universal function approximators, this adds significant computational burden and sample complexity.[13] Research indicates that agents using explicit egocentric vector representations perform better in "aiming" and immediate interaction tasks, while allocentric representations are superior for "guidance" and global navigation.[12]

Audit Implication: The 14-d design likely provides absolute positions (x, y) without relative vectors. This forces the policy to relearn the concept of "distance" and "direction" for every new location in the map, violating translational invariance. A robust embedding must provide pre-computed egocentric vectors (Range, Bearing) alongside allocentric context.[13]

2.2 Invariance and Symmetry Groups

A critical property of robust RL policies is invariance to symmetry groups, specifically $SE(2)$

(translation and rotation in 2D) or $SE(3)$ (in 3D). If a tactical scenario is rotated by 90 degrees, the optimal relative action (e.g., "turn left to engage") should remain unchanged. Standard Cartesian inputs are not rotationally invariant. A policy trained on scenarios oriented North will struggle to generalize to East-facing scenarios unless it sees exhaustive data augmentation.[15] Recent work in geometric deep learning demonstrates that utilizing equivariant network architectures or invariant input features (like relative distances and angles) drastically reduces the volume of the state space the agent must explore.[16] By embedding the physics of the environment into the observation space—ensuring that the representation of a "flanking maneuver" looks identical regardless of map position—we accelerate convergence and improve zero-shot transfer.[17]

2.3 The Curse of Discontinuity in Angular Representation

A specific, pervasive error in simulation embeddings is the representation of cyclic variables, such as heading or turret yaw, as scalar values (e.g., $$ degrees or $[-\pi, \pi]$ radians). This creates a numerical discontinuity: the values 359° and 1° are numerically distant despite being physically adjacent. This "zig-zag" problem in the gradient landscape forces the neural network to learn a highly non-linear mapping to handle the boundary condition.[18] Furthermore, using a single scalar prevents the network from easily performing vector arithmetic. The standard and mathematically robust solution is to project the angle θ onto the unit circle (or sphere in 3D), representing it as the pair $(\sin \theta, \cos \theta)$. This ensures that the representation is continuous and differentiable everywhere, stabilizing the learning dynamics of the policy gradient.[17]

---

3. Comprehensive Audit of the 14-Dimension Embedding

Based on standard patterns in tactical simulation design and the constraints of a "14-d" limit, we reconstruct the likely composition of the baseline embedding to perform a specific critique. A typical 14-d vector for a ground or aerial unit usually comprises:

- **Dims 1-3:** Agent Absolute Position (x_a, y_a, z_a)
- **Dims 4-6:** Agent Velocity (v_x, v_y, v_z)
- **Dims 7-9:** Goal/Target Absolute Position (x_g, y_g, z_g)
- **Dim 10:** Agent Heading/Yaw (θ)
- **Dim 11:** Mission/Goal ID (Integer)
- **Dim 12:** Health Status
- **Dim 13:** Ammo/Energy
- **Dim 14:** Time Remaining

3.1 Critique 1: Spatial Non-Stationarity and Leakage

The inclusion of absolute coordinates (Dims 1-3, 7-9) is the most severe flaw. By feeding raw global positions, the agent learns a policy dependent on specific map locations. This is known

as **overfitting to the coordinate frame**. If the mission area shifts by 1km, the inputs change drastically, rendering the learned weights useless.[17]
Moreover, absolute coordinates often inadvertently leak oracle information if the coordinate origin $(0,0,0)$ has semantic meaning (e.g., center of the map) that the agent shouldn't inherently know. In a proper goal-conditioned system, the agent should only know the *vector to the goal*, not the goal's universal GPS coordinate, unless simulating a GPS-guided munition.[13]
Assessment: High Risk. The agent will likely fail to generalize to new maps or shifted starting positions.

3.2 Critique 2: Scaling and Normalization Pathology

Deep RL algorithms (PPO, SAC, DDPG) rely on gradient descent, which is highly sensitive to feature scaling.

- **Position:** Values may range from 0 to 5000 (meters).
- **Velocity:** Values range from -50 to 50 (m/s).
- **Heading:** Values range from 0 to 6.28 (radians).
- **Mission ID:** Integers $1, 2, 3$.
- **Health:** 0 to 100 or 0 to 1.

Without rigorous normalization, the gradients from the "Position" dimensions (magnitude ~5000) will dwarf the gradients from "Heading" (magnitude ~6), effectively blinding the agent to its own orientation. While Batch Normalization layers can mitigate this, relying on them for such extreme disparities is bad practice.
The "Zero-Centering" Necessity: Standard practice often defaults to Min-Max scaling to $$. However, for continuous control, symmetric scaling to $[-1, 1]$ (zero-centered) is superior. Activation functions like tanh saturate at ± 1 and are linear near 0. Inputs in $$ introduce a systematic mean shift (bias) in the first hidden layer, which can slow down training.[18]
Assessment: Critical Risk. The agent will struggle to converge, likely ignoring orientation and mission type in favor of position matching.

3.3 Critique 3: The "Integer Trap" in Goal Representation

Representing the Mission ID (Dim 11) as a scalar integer is mathematically unsound for neural networks. If "Patrol" = 1, "Attack" = 2, and "Defend" = 3, the network infers an ordinal relationship: $Defend > Attack > Patrol$. It also assumes $Attack$ is the average of $Patrol$ and $Defend$. This imposes a false topology on the task space.[22]
While one-hot encoding is the traditional fix, it increases dimensionality. A 14-d limit suggests a bottleneck. A single scalar cannot robustly encode categorical intent without confusing the value function.[24]

3.4 Critique 4: Information Starvation & The "Lone Wolf" Bias

The most profound limitation is the absence of **coordination features**. The 14-d vector contains zero information about:

- Teammate positions or status.
- Enemy positions or threats.
- Terrain obstacles or cover.

In this "sensory vacuum," the agent acts as a solipsistic entity. It cannot coordinate fire because it doesn't know where allies are.[25] It cannot flank enemies because it has no representation of the enemy's facing angle.[4] It cannot use cover because it doesn't sense the environment.[26]

Assessment: Fatal Flaw for Tactical Simulation. This embedding supports simple navigation (pathfinding) but renders tactical decision-making impossible. The agent will never learn to suppress or flank because the *state space does not contain the variables that define those concepts*.

3.5 Critique 5: The Permutation Problem

Even if we attempted to hack in enemy data (e.g., adding "Nearest Enemy X, Y"), we encounter the permutation problem. If "Enemy A" is closest at time t, and "Enemy B" is closest at time $t+1$, the inputs jump discontinuously. Fixed-vector observations cannot handle the variable cardinality of multi-agent environments (e.g., 3 enemies vs. 5 enemies). Naive concatenation ($[e_1, e_2, \dots, e_N]$) imposes an artificial ordering that confuses the critic network.[5]

---

4. Best Practices for Tactical State Representation

To engineer a "V2" embedding, we must synthesize best practices from MARL literature, specifically focusing on how to represent spatial intent, relative geometry, and complex mission objectives.

4.1 Goal Embeddings: Beyond One-Hot to Semantics

Modern goal-conditioned RL moves beyond discrete IDs. Research into task embeddings suggests using a **Learned Task Encoder**. This can be a Variational Autoencoder (VAE) or a contrastive embedding that maps high-dimensional task descriptions (e.g., "Defend sector A while minimizing casualties") into a dense, low-dimensional latent vector z.[27]

Alternatively, **Vision-Language-Action (VLA)** models demonstrate the power of language conditioning. By encoding the mission instruction using a frozen language model (like BERT or CLIP), the agent receives a semantically rich vector where "Patrol" and "Scout" might naturally be close in vector space, facilitating transfer learning.[9] For the V2 embedding, replacing the integer ID with a small dense embedding (e.g., 4-8 dimensions) allows for interpolation between mission types and generalization to unseen commands.[30]

4.2 Entity-Centric Observations via Attention

The standard solution to the "Permutation Problem" (variable number of units) is the **Entity-Centric** approach. Instead of a fixed vector, the observation is treated as a *set* of feature vectors $\{\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_N\}$. These are processed by

a permutation-invariant architecture, typically a **Transformer Encoder** or a **DeepSet** network.[31]
This mechanism creates a "Tactical Context" vector. The agent learns to attend to the entities that matter (e.g., the enemy aiming at me) and ignore those that don't, regardless of their order in the list. This is crucial for scalability; the same policy can control an agent in a 3v3 skirmish or a 50v50 battle without retraining.[5]

4.3 Explicit Geometric Feature Engineering

While deep learning can theoretically learn geometry, providing explicit **inductive biases** accelerates training.

- **Polar Coordinates:** Representing the goal as $(\rho, \sin \theta, \cos \theta)$ allows the agent to decouple "steering" (dependent on θ) from "throttle" (dependent on ρ).[12]
- **Aspect & Bearing Angles:** To enable dogfighting and flanking, the state must include the *Aspect Angle* (AA) and *Antenna Train Angle* / Bearing Angle (BA). AA measures the angle between the enemy's tail and the agent's line of sight. Minimizing AA places the agent in the enemy's blind spot (the "six").[4]
- **Line of Fire (LOF):** A boolean or continuous value indicating if a raycast to the target is obstructed. This is essential for learning when to shoot vs. when to move. An agent provided with LOF=0 learns "don't shoot into the wall" much faster than one who must infer it from coordinate geometry.[34]

4.4 Hierarchical Decomposition

For complex missions ("Go to A, then Suppress B"), a single flat policy often fails due to the long time horizon. **Hierarchical RL (HRL)** separates the problem into a "High-Level Controller" (which selects a sub-goal or tactic) and a "Low-Level Controller" (which executes movement and shooting). The embedding for the Low-Level controller can be simpler (local geometry), while the High-Level controller consumes the global mission embedding.[36] This aligns with military command structures, decomposing "Mission" into "Tactical Directives."

---

5. Recommended Embedding V2: The "Tactical Tensor" Architecture

We propose abandoning the 14-d flat constraint for the internal representation, instead using a composite observation space that is projected down to a dense embedding *before* the policy network. The V2 design uses a "Feature Interaction" approach.

5.1 Architecture Overview

The observation space is defined as a tuple $\mathcal{O} = (\mathbf{o}_{proprio}, \mathbf{o}_{task}, \mathbf{o}_{tactical}, \mathbf{o}_{env})$.

5.1.1 Stream A: Proprioception (The "Self" Vector)

Replaces absolute coordinates with normalized, relative kinematic data. All angular data is projected to manifolds.

Feature	Type	Dimensionality	Description & Rationale
Body Velocity	Continuous	3	Local frame velocity (v_x, v_y, v_z) normalized to $[-1, 1]$. Crucial for motor control.
Body Acceleration	Continuous	3	Local frame acceleration. Provides the agent with a sense of mass and inertia.[38]
Orientation (Manifold)	Continuous	4	$(\sin P, \cos P, \sin Y, \cos Y)$ for Pitch/Yaw. Avoids singularity at $0/360$.[17]
Integrity State	Continuous	2	Health/Max, Shield/Max. Normalized $$.
Resource State	Continuous	2	Ammo/Max, Energy/Max. Normalized $$.
Action History	Continuous	N_{act}	The previous action vector a_{t-1}. Essential for temporal smoothness and overcoming system latency.[39]

Note: Absolute Heading is removed. Only Relative Heading to targets matters.

5.1.2 Stream B: Task Semantic Embedding (The "Mission" Vector)

Replaces the Integer ID.

Feature	Type	Dimensionality	Description & Rationale
Task Embedding	Dense	8	A learned latent vector derived from a Task Encoder (e.g.,

			"Suppress", "Flank"). Allows semantic interpolation.[27]
Goal Vector (Polar)	Continuous	3	$(\rho_{norm}, \sin \theta_g, \cos \theta_g)$. Relative vector to current waypoint. Log-normalized distance handles multi-scale environments.
Phase Indicator	Continuous	1	Normalized time or progress within the current sub-task $$.

5.1.3 Stream C: The Tactical Matrix (Entity Attention)

This is the core innovation for coordination. It is a set of vectors $\{\mathbf{e}_1, \dots, \mathbf{e}_k\}$ processed by a **Transformer Block**.

Feature	Norm.	Description & Rationale
Relative Dist	Log	Distance to entity. Log scale emphasizes immediate threats over distant ones.
Relative Azimuth	$[-1, 1]$	$(\sin, \cos)$ direction to entity.
Relative Orientation	$[-1, 1]$	**Critical for Flanking:** The entity's facing relative to the agent. $(\sin, \cos)$ of their forward vector. Allows agent to detect "Back" vs "Front".[4]
Line of Fire (LOF)	Binary	Raycast result: 1 = Clear Shot, 0 = Blocked. Enables suppression logic.[34]
Is_Target	Binary	Is this entity the specific target of my current mission?
Alliance ID	One-hot	Friend/Enemy/Neutral.
Threat Index	$$	Heuristic: Is this enemy aiming at me? (Dot product of their forward vector and vector to me).[40]

The Transformer output is pooled (e.g., Global Max Pooling) into a fixed-size "Tactical

Context" vector.

5.1.4 Stream D: Environmental Affordance (The "Map")

To navigate complex terrain without absolute coordinates.

Feature	Type	Description
Lidar/Raycast Array	Array	A 1D array of normalized distances to obstacles (e.g., 16 rays). Provides local obstacle avoidance and cover detection.[41]

5.2 The "Enfilade" and "Suppression" Features

To satisfy the specific user request regarding coordination:

- **Suppression Feature:** We introduce a **"Local Bullet Density"** feature in Stream D. This quantifies the number of projectiles passing near the agent in the last k frames. This gives the agent the sensory input "I am being suppressed," triggering cover-seeking behavior.[26]
- **Enfilade/Flanking Feature:** The **Relative Orientation** in Stream C is the mathematical key. By explicitly providing the cosine similarity between the enemy's look vector and the agent's position vector, the policy can easily learn that $Values \approx 1$ (behind enemy) correlate with high rewards (survival, damage bonus).

---

6. Identifying Risks: Leakage, Scaling, and Invariance

6.1 Information Leakage & The Oracle Problem

A subtle but critical risk in tactical RL is "Oracle Leakage"—providing the agent with data it cannot physically know.

- **Through-Wall Awareness:** If the Tactical Matrix (Stream C) includes enemies that are fully occluded by terrain, the agent effectively has "wall-hacks." This creates unrealistic behaviors where agents track targets through solid objects.[43]
- **Mitigation:** The environment wrapper must implement a **Field of View (FOV)** and **Occlusion Culling** filter. Entities not visible to the sensor suite should be masked out or replaced with a "Last Known Position" ghost entity (a "Belief State" mechanism).[44] This transitions the problem from an MDP to a POMDP (Partially Observable MDP), requiring the use of Recurrent Neural Networks (LSTM/GRU) in the policy to maintain memory of hidden foes.[45]

6.2 Normalization Pitfalls: The $[-1, 1]$ Standard

As highlighted in the snippets, the choice of normalization range is not arbitrary.

- **Zero-Centering:** Normalizing to $$ pushes the mean of the input distribution to 0.5. In deep networks initialized with zero-mean weights (e.g., Xavier/Glorot), this creates a bias shift in the early layers that the optimizer must laboriously correct.[18]
- **Recommendation:** Use **Z-Score Standardization** ($x' = \frac{x - \mu}{\sigma}$) or symmetric Min-Max scaling to $[-1, 1]$. This keeps the input distribution centered around zero, aligning with the linear region of tanh and relu activations, ensuring faster convergence.[46]

6.3 Scaling: Dealing with Density

In scenarios like aerial firefighting or swarm combat, the number of entities can be large. Concatenating vectors fails. Using a **Transformer Encoder** in the embedding layer makes the architecture invariant to the number of agents (N). The attention mechanism learns to assign high weights to the K most relevant entities (e.g., the closest enemy, the VIP) and low weights to distant ones, effectively performing dynamic feature selection.[31]

---

7. Ablation Plan: Validating the V2 Design

To scientifically prove the superiority of Embedding V2, we propose a subtractive ablation study. We start with the full V2 architecture and remove components to measure their impact on specific tactical metrics.
Test Environment: A 5v5 "Capture the Flag" scenario with complex cover, requiring both navigation and combat.

7.1 Experiment 1: The "Compass Test" (Geometric Invariance)

- **Objective:** Validate the shift from Absolute (x,y) to Relative/Polar (ρ, θ) coordinates.
- **Setup:** Train agents on maps oriented North.
- **Evaluation:** Test the trained agents on the *same* maps rotated by $90^\circ, 180^\circ, 270^\circ$.
- **Hypothesis:** The Baseline (14-d) agent's performance will collapse (near random) on rotated maps because it overfitted to global coordinates. The V2 agent will maintain near-identical performance due to rotational invariance.[15]

7.2 Experiment 2: The "Flanking Index" (Tactical Features)

- **Objective:** Determine if "Relative Orientation" inputs enable emergent flanking.
- **Metric:** Define **Flanking Index (F_{idx})** as the proportion of successful engagements where the firing angle is $>90^\circ$ relative to the target's forward vector (attacking from the rear arc).[4]
- **Conditions:**
    - *Baseline:* V2 Embedding *without* Relative Orientation features.
    - *Full V2:* V2 Embedding *with* Relative Orientation features.

- **Hypothesis:** The Full V2 agent will show a statistically significant increase in F_{idx}, proving that explicit geometric cues are necessary for the policy to discover enfilade tactics.

7.3 Experiment 3: The "Swarm Scalability" Test (Permutation Invariance)

- **Objective:** Validate the Transformer/DeepSet encoder against fixed vector concatenation.
- **Setup:** Train on 3v3 scenarios.
- **Evaluation:** Zero-shot transfer to 5v5 and 10v10 scenarios.
- **Hypothesis:** The Baseline (concatenated vector) cannot even run 10v10 without retraining/architecture changes. The V2 (Transformer) agent will adapt, maintaining formation cohesion even with increased entity counts.[5]

7.4 Experiment 4: The "Suppression" Response (Coordination)

- **Objective:** Test the "Local Bullet Density" feature.
- **Setup:** Script an enemy to fire "suppressive fire" (high volume, low accuracy) at a chokepoint.
- **Metric:** Time spent in "Cover" when under fire.
- **Hypothesis:** Agents with the "Local Bullet Density" feature will learn to retreat/cover significantly faster than agents who only sense direct damage (Health drops).

---

8. Conclusion

The audit of the provided 14-dimensional embedding reveals a design that is fundamentally fragile for high-fidelity tactical RL. By relying on absolute coordinates, discontinuous scalars, and implicit entity ordering, the baseline design invites the "Curse of Dimensionality" while simultaneously failing to provide the specific geometric cues required for tactical coordination.

The recommended **Embedding V2** represents a paradigm shift from **"State Logging"** to **"Semantic Perception."** By enforcing rotational invariance through polar manifolds, enabling scalability through attention-based entity encoders, and explicitly engineering tactical affordances (Line of Fire, Aspect Angle, Suppression Density), the new embedding effectively lowers the "cognitive load" on the RL policy. This allows the agent to bypass the struggle of basic spatial comprehension and focus its capacity on discovering higher-order strategies like envelopment, suppression, and dynamic formation control.

Implementing this V2 design, specifically utilizing the ablation plan to verify the necessity of the "Relative Orientation" and "Permutation Invariant" modules, is the direct path to achieving emergent coordinated behaviors that rival human tactical intuition.

---

9. Detailed Appendices on Implementation Mechanisms

9.1 Implementing the Vision-Language-Action (VLA) Paradigm

Recent advancements in VLA models, such as RT-2 and OpenVLA [9], demonstrate the utility of "tokenizing" actions and observations. While the V2 embedding uses continuous vectors for precision control, the *Task Semantic Embedding* (Stream B) draws directly from this paradigm.

- **Mechanism:** A pre-trained language encoder (e.g., a distilled BERT or CLIP text encoder) processes the textual mission string (e.g., "Provide covering fire for Alpha Squad").
- **Latent Projection:** The resulting high-dimensional vector (e.g., 512-d) is projected down to the 8-d Task Embedding via a learnable linear layer.
- **Benefit:** This allows "Instruction Following." If the agent is trained on "Attack Alpha" and "Defend Bravo," it can zero-shot understand "Attack Bravo" because the language embedding space captures the compositional semantics of the verb "Attack" and the noun "Bravo".[29]

9.2 The Mathematics of the Enfilade Metric

To quantitatively measure the "Flanking" capability requested, we define the Enfilade Index (E) derived from the V2 features.
For Agent A targeting Enemy E:

$$E = \text{LOF}_{A \to E} \times \frac{1}{2} (1 + \mathbf{v}_{A \to E} \cdot \mathbf{f}_E)$$

Where:

- $\text{LOF}_{A \to E}$ is binary Line of Fire.
- $\mathbf{v}_{A \to E}$ is the normalized vector from Agent to Enemy.
- $\mathbf{f}_E$ is the Enemy's forward facing vector.
- The dot product $\mathbf{v} \cdot \mathbf{f}$ ranges from -1 (Head-on) to 1 (Rear/Tail chase).
- The result is a metric $$ where 1 represents a perfect rear-attack (maximum enfilade).

This metric should be used both as an evaluation KPI and potentially as a *Reward Shaping* term during training to encourage the agent to seek advantageous geometry.[4]

9.3 Coordination via Intention Sharing

A key missing feature in the 14-d baseline is explicit coordination. In V2, we recommend exploring **Intention Sharing** if bandwidth allows. This involves appending a "Future Intent" vector (e.g., the agent's predicted position at $t+5s$) to the Entity Stream of its teammates.

- **Research Basis:** Studies in "Spatial Intention Maps" [52] and cooperative MARL [53] show

that when agents broadcast their intended path, collision rates drop and formation cohesion improves.

- **Implementation:** The Policy network outputs an auxiliary head $\hat{p}_{t+k}$ (predicted future state). This is broadcast to teammates and ingested into their "Stream C" (Tactical Matrix). This allows Agent B to "know" that Agent A is moving left, allowing Agent B to cover the right flank automatically.

9.4 Normalization of Fire Suppression Dynamics

Simulating "Suppression" requires specific environmental dynamics often modeled in systems like Cell2Fire or military combat simulators.[54]

- **The Suppression Variable:** We define a variable $S_t \in $ for each agent.
  - S_t increases when projectiles pass within a radius $r_{suppress}$ of the agent ($S_{t} = S_{t-1} + \alpha$).
  - S_t decays over time ($S_{t} = S_{t-1} \times \gamma$).
- **Observation:** This S_t value is fed into **Stream A (Proprioception)**.
- **Effect:** When $S_t > Threshold$, the agent's accuracy (action noise) should arguably increase (simulating stress), or the agent should be rewarded for breaking LOF (seeking cover). Without this explicit feedback loop in the observation space, "suppression" is just invisible noise.

By integrating these advanced mechanisms, the RL system moves beyond simple "bot movement" to genuine tactical cognition.

Works cited

1. Beyond Distributions: Geometric Action Control for Continuous Reinforcement Learning, accessed December 25, 2025, https://arxiv.org/html/2511.08234v1
2. A Geometric Perspective on Optimal Representations for Reinforcement Learning, accessed December 25, 2025, http://papers.neurips.cc/paper/8687-a-geometric-perspective-on-optimal-representations-for-reinforcement-learning.pdf
3. Expandable Decision-Making States for Multi-Agent Deep Reinforcement Learning in Soccer Tactical Analysis - ResearchGate, accessed December 25, 2025, https://www.researchgate.net/publication/396095020_Expandable_Decision-Making_States_for_Multi-Agent_Deep_Reinforcement_Learning_in_Soccer_Tactical_Analysis
4. Aerial Combat Relative Geometry | Download Scientific Diagram - ResearchGate, accessed December 25, 2025, https://www.researchgate.net/figure/Aerial-Combat-Relative-Geometry_fig1_339347161
5. PIC: Permutation Invariant Critic for Multi-Agent Deep Reinforcement Learning, accessed December 25, 2025, http://proceedings.mlr.press/v100/liu20a/liu20a.pdf
6. PIC: Permutation Invariant Critic for Multi-Agent Deep Reinforcement Learning -

arXiv, accessed December 25, 2025, https://arxiv.org/pdf/1911.00025
7. Egocentric and allocentric spatial memory in typically developed children: Is spatial memory associated with visuospatial skills, behavior, and cortisol? - PubMed Central, accessed December 25, 2025, https://pmc.ncbi.nlm.nih.gov/articles/PMC7218242/
8. The Role of Temporal Order in Egocentric and Allocentric Spatial Representations - MDPI, accessed December 25, 2025, https://www.mdpi.com/2077-0383/12/3/1132
9. Vision-language-action model - Wikipedia, accessed December 25, 2025, https://en.wikipedia.org/wiki/Vision-language-action_model
10. Vision-Language-Action Models for Robotics: A Review Towards Real-World Applications - IEEE Xplore, accessed December 25, 2025, https://ieeexplore.ieee.org/iel8/6287639/10820123/11164279.pdf
11. GEOMETRY OF NEURAL REINFORCEMENT LEARNING IN CONTINUOUS STATE AND ACTION SPACES - Brown Computer Science, accessed December 25, 2025, https://cs.brown.edu/people/gdk/pubs/geometry_neural_rl.pdf
12. Navigation task and action space drive the emergence of egocentric and allocentric spatial representations | PLOS Computational Biology - Research journals, accessed December 25, 2025, https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1010320
13. Navigation task and action space drive the emergence of egocentric and allocentric spatial representations - PMC - NIH, accessed December 25, 2025, https://pmc.ncbi.nlm.nih.gov/articles/PMC9648855/
14. Continuous integration of heading and goal directions guides steering - PMC - PubMed Central, accessed December 25, 2025, https://pmc.ncbi.nlm.nih.gov/articles/PMC11527344/
15. GEOMETRY-AWARE RL FOR MANIPULATION OF VARY- ING SHAPES AND DEFORMABLE OBJECTS - ICLR Proceedings, accessed December 25, 2025, https://proceedings.iclr.cc/paper_files/paper/2025/file/7571c9d44179c7988178593c5b62a9b6-Paper-Conference.pdf
16. Geometry-aware RL for Manipulation of Varying Shapes and Deformable Objects - arXiv, accessed December 25, 2025, https://arxiv.org/abs/2502.07005
17. Designing Effective State Representations in Reinforcement Learning - CodeSignal, accessed December 25, 2025, https://codesignal.com/learn/courses/advanced-rl-techniques-optimization-and-beyond/lessons/designing-effective-state-representations-in-reinforcement-learning
18. Normalizing to [0,1] vs [-1,1] - Stack Overflow, accessed December 25, 2025, https://stackoverflow.com/questions/46597877/normalizing-to-0-1-vs-1-1
19. RL in coordinate system : r/reinforcementlearning - Reddit, accessed December 25, 2025, https://www.reddit.com/r/reinforcementlearning/comments/n4tjf9/rl_in_coordinate_system/
20. (PDF) A 3D Spatial Information Compression Based Deep Reinforcement Learning Technique for UAV Path Planning in Cluttered Environments - ResearchGate,

accessed December 25, 2025, https://www.researchgate.net/publication/388868564_A_3D_Spatial_Information_Compression_Based_Deep_Reinforcement_Learning_Technique_for_UAV_Path_Planning_in_Cluttered_Environments
21. Input normalization: (-1 to 1) or (0 to 1)? : r/MLQuestions - Reddit, accessed December 25, 2025, https://www.reddit.com/r/MLQuestions/comments/cwdtrf/input_normalization_1_to_1_or_0_to_1/
22. One hot encoding of integer features in RL state representations : r/reinforcementlearning, accessed December 25, 2025, https://www.reddit.com/r/reinforcementlearning/comments/lljmdq/one_hot_encoding_of_integer_features_in_rl_state/
23. Why is one-hot encoding used in RL instead of binary encoding? - Stats StackExchange, accessed December 25, 2025, https://stats.stackexchange.com/questions/670027/why-is-one-hot-encoding-used-in-rl-instead-of-binary-encoding
24. What is the difference between embeddings and one-hot encoding? - Milvus, accessed December 25, 2025, https://milvus.io/ai-quick-reference/what-is-the-difference-between-embeddings-and-onehot-encoding
25. Multi-Agent Reinforcement Learning for Coordinated Aerial Wildfire Suppression, accessed December 25, 2025, https://openreview.net/forum?id=IdJvKeDQ5A
26. Terrain Reasoning for 3D Action Games - Game Developer, accessed December 25, 2025, https://www.gamedeveloper.com/programming/terrain-reasoning-for-3d-action-games
27. Learning Embeddings for Sequential Tasks Using Population of Agents - IJCAI, accessed December 25, 2025, https://www.ijcai.org/proceedings/2024/0523.pdf
28. Meta Reinforcement Learning with Task Embedding and Shared Policy - IJCAI, accessed December 25, 2025, https://www.ijcai.org/proceedings/2019/0387.pdf
29. Complex Instruction Following with Diverse Style Policies in Football Games - arXiv, accessed December 25, 2025, https://arxiv.org/html/2511.19885v1
30. DISENTANGLED CODE EMBEDDING FOR MULTI-TASK REINFORCEMENT LEARNING: A DUAL-ENCODER APPROACH WITH DYNAMIC GAT - OpenReview, accessed December 25, 2025, https://openreview.net/pdf?id=dcqnFZAczW
31. SPECTra: Scalable Multi-Agent Reinforcement Learning with Permutation-Free Networks, accessed December 25, 2025, https://arxiv.org/html/2503.11726v1
32. Centralized Permutation Equivariant Policy for Cooperative Multi-Agent Reinforcement Learning - arXiv, accessed December 25, 2025, https://arxiv.org/html/2508.11706
33. Entity-based Reinforcement Learning for Autonomous Cyber Defence - arXiv, accessed December 25, 2025, https://arxiv.org/html/2410.17647v3
34. how do i make line of sight obstructions in gamemaker? - Reddit, accessed December 25, 2025, https://www.reddit.com/r/gamemaker/comments/lelz08/how_do_i_make_line_of_s

ight_obstructions_in/
35. We are developing enemy line of sight in our game Dros. It's cool being able to hide from enemies in a little nook or behind a wall as they patrol. Check out some of our dev shots. : r/Unity3D - Reddit, accessed December 25, 2025, https://www.reddit.com/r/Unity3D/comments/oc143r/we_are_developing_enemy_line_of_sight_in_our_game/
36. Hierarchical Multi-Agent Reinforcement Learning - Emergent Mind, accessed December 25, 2025, https://www.emergentmind.com/topics/hierarchical-multi-agent-reinforcement-learning-hmarl
37. The Promise of Hierarchical Reinforcement Learning - The Gradient, accessed December 25, 2025, https://thegradient.pub/the-promise-of-hierarchical-reinforcement-learning/
38. Evaluating the Coordination of Agents in Multi-agent Reinforcement Learning, accessed December 25, 2025, https://www.researchgate.net/publication/330173457_Evaluating_the_Coordination_of_Agents_in_Multi-agent_Reinforcement_Learning
39. Investigating the Impact of Observation Space Design Choices On Training Reinforcement Learning Solutions for Spacecraft Problems - arXiv, accessed December 25, 2025, https://arxiv.org/html/2501.06016v1
40. Project TH: Korean Tactical Shooter AI Echoes F.E.A.R.'s Intelligent Enemies | Technetbook, accessed December 25, 2025, https://www.technetbooks.com/2025/03/project-th-korean-tactical-shooter-ai.html
41. Reinforcement Learning for Adaptive Planner Parameter Tuning: A Perspective on Hierarchical Architecture - arXiv, accessed December 25, 2025, https://arxiv.org/html/2503.18366v1
42. (PDF) Optimization of Fire Suppression Mechanisms Using Multi-Agent Robotic Systems, accessed December 25, 2025, https://www.researchgate.net/publication/397560719_Optimization_of_Fire_Suppression_Mechanisms_Using_Multi-Agent_Robotic_Systems
43. This upcoming Korean tactical shooter features 'situational awareness AI' that reminds me of the uber-smart clones from Monolith's FEAR | PC Gamer, accessed December 25, 2025, https://www.pcgamer.com/games/action/this-upcoming-korean-tactical-shooter-features-situational-awareness-ai-that-reminds-me-of-the-uber-smart-clones-from-monoliths-fear/
44. The Use of Egocentric and Allocentric Reference Frames in Static and Dynamic Conditions in Humans - PMC - PubMed Central, accessed December 25, 2025, https://pmc.ncbi.nlm.nih.gov/articles/PMC8549915/
45. Dota 2 with Large Scale Deep Reinforcement Learning - OpenAI, accessed December 25, 2025, https://cdn.openai.com/dota-2.pdf
46. Observation Normalization - Maze documentation - Read the Docs, accessed December 25, 2025, https://maze-rl.readthedocs.io/en/latest/environment_customization/observation_

normalization.html
47. Normalization: Min-Max and Z-Score – AI Robotics - Reinforcement Learning Path, accessed December 25, 2025, https://www.reinforcementlearningpath.com/normalization/
48. MetaSpatial: Reinforcing 3D Spatial Reasoning in VLMs for the Metaverse - arXiv, accessed December 25, 2025, https://arxiv.org/pdf/2503.18470?
49. Vision-Language-Action Models: The Architecture Powering the Robot Revolution - Medium, accessed December 25, 2025, https://medium.com/@nraman.n6/vision-language-action-models-the-architecture-powering-the-robot-revolution-76f2ce9f400a
50. Language-conditioned Multi-Style Policies with Reinforcement Learning - OpenReview, accessed December 25, 2025, https://openreview.net/forum?id=KohdorhwHt
51. Lanchester's laws - Wikipedia, accessed December 25, 2025, https://en.wikipedia.org/wiki/Lanchester%27s_laws
52. [2103.12710] Spatial Intention Maps for Multi-Agent Mobile Manipulation - arXiv, accessed December 25, 2025, https://arxiv.org/abs/2103.12710
53. Complementary Attention for Multi-Agent Reinforcement Learning, accessed December 25, 2025, https://proceedings.mlr.press/v202/shao23b/shao23b.pdf
54. Firehose : Wildfire Prevention and Management using Deep Reinforcement Learning, accessed December 25, 2025, https://williamshen-nz.github.io/firehose/firehose-paper.pdf
55. CFD Modeling of Fire Suppression and - National Institute of Standards and Technology, accessed December 25, 2025, https://www.nist.gov/system/files/documents/el/fire_research/R0301584.pdf