VERA: VERIFIABLE FEASIBILITY REPRESENTATIONS WITH COUNTERFACTUAL CREDIT FOR CONSTRAINED MULTI-AGENT CONTROL

Anonymous authors
Paper under double-blind review

ABSTRACT

Constrained multi-agent control requires more than predicting rewarding actions: an action can cease to be executable as contact windows, shared capacity, and deadlines change. We introduce VERA, a centralized-training, decentralized-execution framework that separates feasibility estimation from credit assignment. Each actor predicts a five-dimensional *verifiable feasibility representation* (VFR). After an action is proposed, exact action-conditioned margins available only during training supervise that representation, while a *counterfactual group-relative advantage* (CGRA) ranks candidate representation–action pairs. Execution uses one actor pass and no privileged state. In a dynamic space–air–ground integrated network (SAGIN), VERA obtains $55.33 \pm 3.60\%$ success with $0.45 \pm 0.81\%$ coverage violation, within 1.33 points of a privileged-mask reference. With rewards matched over ten paired seeds, VERA improves success over the strongest baseline by 8.74 points ($p = 0.023$) and reduces violation by 52.19 points ($p = 5.7 \times 10^{-8}$). A ten-seed 4×2 factorial attributes a 14.16–16.48 point gain to CGRA across handcrafted, learned, random, and latent representations; evaluation on seven unseen topologies preserves a 24.33–30.02 point advantage over multi-agent proximal policy optimization. From 10 to 40 users, success remains 50.1–53.8%, and VFR adds only 0.026 ms to a central processing unit (CPU) actor step. Cross-domain tests further identify the governing condition: counterfactual credit succeeds when candidate scores respect shared constraints and fails under incompatible reward geometries. These results establish action-conditioned feasibility as an auditable training interface and counterfactual credit as a geometry-dependent optimization mechanism.

1 INTRODUCTION

Multi-agent reinforcement learning (MARL) has become a practical route to decentralized control, with centralized critics, value factorization, and policy-gradient methods supporting increasingly large cooperative systems (Yu et al., 2022; Lowe et al., 2017; Rashid et al., 2020; Foerster et al., 2018; Peng et al., 2021; Fu et al., 2022). Yet most policies still compress two different questions into one action distribution: which action has produced high return, and which action remains physically executable now. In a dynamic physical system, the second question changes with contact duration, congestion, resource contention, and deadline slack. A policy can therefore exploit a stable reward correlation while failing to represent the mechanism that makes its decision feasible.

This distinction is especially sharp in a space–air–ground integrated network (SAGIN). Each ground agent must choose local execution, a low-Earth-orbit (LEO) satellite, or an unmanned aerial vehicle (UAV) edge node, together with an offloading ratio and bandwidth request. Satellite visibility, shared capacity, remote load, and end-to-end latency jointly determine whether the selected action can complete. Existing learning-based offloading and routing methods optimize return or expected constraints (Huang et al., 2020; Chen & Wang, 2020; Ji et al., 2023b; Huang et al., 2024; Gao et al., 2024; Tang et al., 2022; Cai et al., 2025; Lyu et al., 2024), while a hard action mask can enforce feasibility only when exact instantaneous constraints are available at decision time. In our decentralized setting, those global quantities are unavailable to the deployed actor, although the simulator can compute them after an action is proposed.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

We introduce VERA, a training architecture that exploits this asymmetry without leaking privileged information into execution. The actor predicts a five-dimensional *verifiable feasibility representation* (VFR): contact margin, energy difference, reachability, target utilization, and deadline buffer. Once the actor proposes an action, an environment verifier computes its exact action-conditioned target. VFR supervision makes the policy's feasibility estimate inspectable; an action-consistency objective connects that estimate to control. A *counterfactual group-relative advantage* (CGRA) then scores a small group of candidate representation–action pairs against the same state and peer requests. Verification and candidate scoring are used only during training. At execution, every agent uses its local observation in a single forward pass.

The central scientific question is not whether additional shaping improves return, but which component creates the gain and when that gain transfers. We address this with reward-matched baselines, a seven-step mechanism ladder, a 4×2 representation–CGRA factorial, coordinate-removal and permutation controls, ten paired seeds, seven topology families, cross-topology calibration, scale sweeps, and three non-SAGIN stress tests. This design separates verifier grounding from reward choice, architectural capacity, and the relative-credit update.

The resulting evidence is both strong and mechanistically specific. Under identical execution rewards, VERA improves success over reward-matched, attention-augmented Beta-distribution multi-agent proximal policy optimization (AB-MAPPO-R) by 8.74 points and reduces coverage violation from 52.70% to 0.51% over ten paired seeds. Across four representation types, CGRA contributes 14.16–16.48 success points, whereas the interaction with representation type is not detected. The supervised VFR is therefore an auditable feasibility interface; CGRA is the dominant optimization mechanism in SAGIN. Seven-topology evaluation, signed-margin calibration, and user/satellite scale sweeps show that this combination remains effective beyond the training layout. Resource-allocation and force/contact tests then reveal a complementary result: when the local candidate score conflicts with a shared constraint, the same credit mechanism exposes that mismatch through reproducible failure.

Contributions.

- **Action-conditioned feasibility as a learnable interface.** We formulate VFRs, compact policy variables with exact post-action training targets but no privileged execution inputs, and connect their prediction error to feasibility-sign errors through a margin bound.
- **Counterfactual credit with explicit mechanism attribution.** We pair VFR with CGRA and isolate its effect using reward-matched controls, a seven-arm ladder, and a ten-seed factorial across handcrafted, learned, random, and latent representations.
- **Scale, transfer, and execution evidence.** VERA reaches $55.33 \pm 3.60\%$ success with $0.45 \pm 0.81\%$ coverage violation, remains 24.33–30.02 points above MAPPO-R on seven topologies, sustains 50.1–53.8% success from 10 to 40 users, and adds 0.026 ms to a central processing unit (CPU) actor step.
- **A measured applicability criterion.** Cross-domain factorials, collapse attribution, and pre-registered stabilizer tests show that CGRA is effective when its candidate score respects the environment's constraint geometry and predictably unsafe when local score improvement rewards shared-resource over-demand.

2 Related Work

Cooperative MARL and counterfactual credit. Centralized training with decentralized execution (CTDE) addresses non-stationarity by allowing critics to use global information while actors retain local inputs. Representative approaches include multi-agent deep deterministic policy gradient (MADDPG), value factorization through QMIX, and multi-agent proximal policy optimization (MAPPO) (Lowe et al., 2017; Rashid et al., 2020; Yu et al., 2022; Oliehoek & Amato, 2016; Wen et al., 2022). Sequence-modeling formulations cast offline cooperative control as return-conditioned next-action prediction (Chen et al., 2021; Meng et al., 2023). Counterfactual multi-agent (COMA) policy gradients marginalize an agent's action to construct an agent-specific baseline, while factored multi-agent centralized policy gradients (FACMAC) optimize a factored centralized critic in continuous action spaces (Foerster et al., 2018; Peng et al., 2021). Heterogeneous-agent policy optimization provides monotonic-improvement analysis for sequential multi-agent updates (Kuba et al., 2022; Liu

et al., 2024; Zhong et al., 2024). CGRA addresses a different credit object: it compares sampled representation–action candidates for one agent under fixed same-slot peer requests, then mixes their normalized relative score with generalized advantage estimation (GAE) (Schulman et al., 2016). It requires neither a privileged critic at execution nor enumeration of the joint action space.

Safe and constrained control. Constrained policy optimization and Lagrangian methods optimize expected return subject to expected costs (Achiam et al., 2017; Ding et al., 2023; García & Fernández, 2015; Dai et al., 2024; Yang et al., 2022; Sootla et al., 2022; Ji et al., 2023a; 2024; Wu et al., 2024). In networked systems, constrained MARL has been used for dynamic routing, while graph-based policies capture interference and resource-coupling structure (Shen et al., 2019; Eisen & Ribeiro, 2020; Lyu et al., 2024; Gu et al., 2023). These methods regulate long-run costs but do not necessarily expose why a particular action is feasible. Hard action masking provides per-decision enforcement when exact constraints are known, but that information is privileged in our decentralized benchmark. VFR instead predicts the missing action-conditioned quantities and makes prediction error, consistency, and signed margins directly measurable. It complements, rather than replaces, hard safety layers when such layers are available.

Decision-relevant and interpretable representations. State abstraction seeks compact representations that preserve optimal decisions (Abel et al., 2018); causal, invariant, and bisimulation objectives aim to discard nuisance variation while retaining behaviorally relevant structure (Zhang et al., 2020; Wang et al., 2022; Hansen-Estruch et al., 2022; Le Lan et al., 2022). Concept bottleneck models use annotated intermediate variables to make predictions inspectable and editable (Koh et al., 2020; Kim et al., 2023); recent work extends this idea to reinforcement-learning policies (Delfosse et al., 2024), and feasibility-consistent objectives learn safety-aware embeddings for constrained control (Cen et al., 2024). VFR shares their explicit semantics but differs in two respects. Its targets are computed from the realized state–action pair rather than attached to the observation alone, and the action head retains a backbone bypass. The representation is therefore a supervised, auditable pathway rather than a strict information bottleneck. Our strict-bottleneck and pathway interventions measure the consequence of that choice.

Learning for SAGIN resource management. Space–air–ground integrated networking couples orbital contact, wireless access, edge computation, and shared capacity (Liu et al., 2018; Bakambekova et al., 2024; Mao et al., 2017). Deep reinforcement learning has been applied to mobile-edge offloading, satellite edge computing, hybrid cloud–edge assignment, SAGIN traffic control, UAV-enabled computation, graph-based allocation, and satellite routing (Huang et al., 2020; Chen & Wang, 2020; Ji et al., 2023b; Huang et al., 2024; Gao et al., 2024; Tang et al., 2022; Ning et al., 2023; Cai et al., 2025; Lyu et al., 2024). Most of this literature evaluates task reward under a fixed simulator and reports aggregate constraints. Our focus is complementary: learning an explicit action-conditioned feasibility interface, identifying its relationship to credit assignment, and testing the resulting mechanism across topology, scale, and reward geometry.

Verifier supervision and group-relative optimization. Outcome and process verifiers provide dense supervision when an environment can check intermediate outputs (Lightman et al., 2024). Group-relative policy optimization normalizes scores within sampled groups to avoid a separate value model (Shao et al., 2024), while preference-based objectives replace absolute rewards with relative comparisons (Rafailov et al., 2023). VERA adapts these ideas to continuous physical control: the verifier returns typed feasibility margins rather than textual correctness, and the group varies a single agent's representation–action candidate while holding peer requests fixed. The reward-matched factorial is essential here because it separates the benefit of relative credit from the semantic content of the representation.

3 Verifiable Feasibility and Counterfactual Credit

3.1 Constrained decentralized control

We consider a cooperative partially observable Markov game with agents $i \in \{1, \dots, K\}$. The global state s_t contains task descriptors, agent positions, remote-node loads, link states, and moving-node visibility. Agent i receives a local observation $o_{i,t} \in \mathbb{R}^{d_o}$ and chooses $a_{i,t} = (\alpha_{i,t}, \rho_{i,t}, B_{i,t})$: an

Table 1: Coordinates of the verifiable feasibility representation. Every target is computed for the action already proposed by the actor.

Coordinate	Physical quantity	Feasibility information
c^{cm}	contact margin	remaining contact versus transmission time
c^{ed}	energy difference	selected offload versus local execution
c^{cv}	coverage validity	binary reachability of the selected target
c^{cu}	compute utilization	post-action load at the selected target
c^{db}	deadline buffer	deadline minus predicted completion time

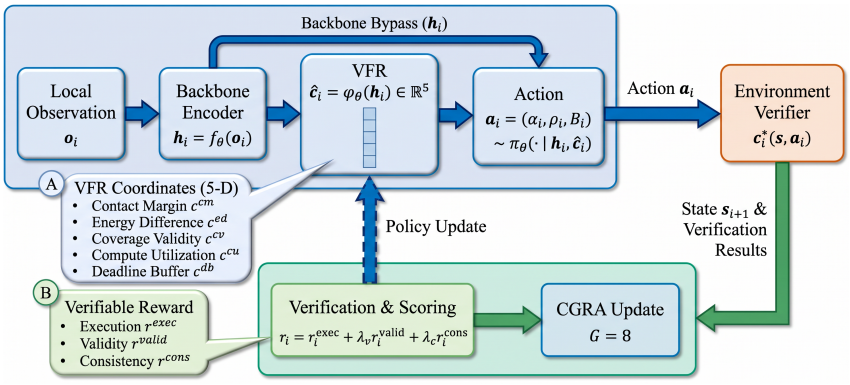


Figure 1: **VERA training and execution.** The actor predicts VFR and conditions its action on both VFR and backbone features. The environment computes the exact target only after an action is proposed. The green verification and candidate-scoring path is removed at execution.

execution target α_i, offloading ratio $\rho_i \in [0, 1]$, and bandwidth request $B_i \in [0, 1]$. A task succeeds only if end-to-end latency is below deadline $D_{\max}$ and the selected remote target remains reachable for the transmission interval. Shared bandwidth and compute introduce coupling among otherwise decentralized actions. Appendix A.1 gives the physical model.

The training simulator observes s_t and can evaluate the realized action, but execution follows CTDE: π_i receives only $o_{i,t}$. This creates an information asymmetry that VERA uses as supervision. The training-time verifier may reveal whether a proposed action is feasible; the execution policy must predict that answer before acting.

3.2 Verifiable Feasibility Representation

A shared encoder computes $h_i = f_\theta(o_i)$. A representation head predicts

$$\hat{\boldsymbol{c}}_i = \phi_\theta(h_i) = \left[\hat{c}_i^{cm}, \hat{c}_i^{ed}, \hat{c}_i^{cv}, \hat{c}_i^{cu}, \hat{c}_i^{db}\right] \in \mathbb{R}^5, \tag{1}$$

and the actor samples

$$a_i \sim \pi_\theta(\cdot \mid h_i, \hat{\boldsymbol{c}}_i). \tag{2}$$

Table 1 summarizes the five coordinates. Each coordinate is normalized to a fixed numerical range before supervision. The direct h_i pathway is retained deliberately: VFR is an explicit, grounded pathway that can be inspected and intervened on, not a claim that five variables are a sufficient statistic for optimal control.

Figure 1 resolves a potential ordering ambiguity. The actor first predicts $\hat{\boldsymbol{c}}_i$ and samples a_i through Eq. (2); only then does the environment compute $\boldsymbol{c}_i^\star(s, a_i)$. No action-conditioned target is available to the actor before its decision.

Feasibility-margin interpretation. Let $\boldsymbol{m}(s, a) \in \mathbb{R}^q$ collect signed feasibility margins, where $m_j(s, a) \geq 0$ denotes satisfaction of constraint j, and let $\hat{\boldsymbol{m}}$ denote their prediction before thresholding. For any $\epsilon > 0$,

$$\Pr[\exists j : \operatorname{sign}(\hat{m}_j) \neq \operatorname{sign}(m_j)] \leq \Pr[\|\hat{\boldsymbol{m}} - \boldsymbol{m}\|_\infty > \epsilon] + \Pr[\min_j |m_j| \leq \epsilon]. \tag{3}$$

When prediction error is at most ϵ and every true margin lies more than ϵ from its boundary, no sign can flip. The bound does not imply optimal control; it identifies the two measurable sources of a wrong feasibility verdict: representation error and operation near a physical boundary. Appendix A.2 gives the proof and the binary-coverage construction.

3.3 Action-conditioned verification

VFR is trained with a coordinate-wise smooth-ℓ_1 loss

$$\mathcal{L}_{\mathrm{VFR}}(\theta) = \frac{1}{5}\sum_{j=1}^{5} \mathrm{smooth}\,\ell_1(\hat{c}_{i,j}, c^{\star}_{i,j}). \tag{4}$$

We also expose verification to the policy objective through validity and consistency terms,

$$r_i^{\mathrm{valid}} = -\|\hat{\boldsymbol{c}}_i - \boldsymbol{c}_i^{\star}\|_1, \qquad r_i^{\mathrm{cons}} = -\,\mathrm{viol}(a_i, \hat{\boldsymbol{c}}_i), \tag{5}$$

where viol measures whether the selected target, offloading ratio, and request contradict the predicted margins. The complete reward is

$$r_i = r_i^{\mathrm{exec}} + \lambda_v r_i^{\mathrm{valid}} + \lambda_c r_i^{\mathrm{cons}}. \tag{6}$$

The complete configuration uses stronger success shaping and an explicit coverage penalty in r_i^{exec}. We therefore do not attribute Table 2 to architecture alone. Reward-matched baselines receive the identical execution reward, and the mechanism ladder fixes that reward while varying supervision, conditioning, consistency, bottlenecking, and CGRA.

3.4 Counterfactual group-relative advantage

For each state–agent pair, the actor samples $G = 8$ representation–action candidates. A one-sided what-if evaluator scores candidate g while keeping peers' realized same-slot requests fixed, producing $r_i^{(g)}$. With group mean μ_i and standard deviation σ_i, the executed candidate receives

$$A_i^{\mathrm{grp}} = \mathrm{clip}\left(\frac{r_i^{(0)} - \mu_i}{\sigma_i + \epsilon}, -3, 3\right). \tag{7}$$

The policy update mixes this term with the ordinary GAE advantage,

$$A_i = A_i^{\mathrm{GAE}} + \lambda_g A_i^{\mathrm{grp}}, \tag{8}$$

inside the clipped proximal policy optimization objective (Schulman et al., 2016; 2017). We call Eq. (7) the *counterfactual group-relative advantage*. It is inspired by group-relative normalization (Shao et al., 2024; Liu et al., 2025), but it is neither a joint counterfactual oracle nor COMA's marginalized baseline: peer reactions and future dynamics are not re-simulated. This distinction explains both its efficiency and its applicability condition. CGRA is informative when fixed-peer candidate scores preserve the relevant constraint ordering; it can be biased when an agent's locally favorable request creates an unpriced externality.

Training and execution complexity. Training evaluates G candidates per selected state–agent pair, whereas execution computes only f_θ, ϕ_θ, and π_θ once. The verifier, counterfactual evaluator, global state, and peer requests are absent from the deployed path. Measured latency in Sec. 4.4 confirms that this separation leaves the decentralized actor within 0.055 ms of MAPPO on both tested devices.

4 Experiments

Environment, baselines, and reporting. The standard SAGIN has 20 mobile agents, six LEO satellites at 550 km, and four UAV nodes; local observation and action dimensions are 66 and 13. Unless stated otherwise, training lasts 3,000 episodes and each seed is summarized over its final 500 episodes (the scale sweeps use a 500-episode budget and report the converged window, episodes > 250). The primary benchmark uses five seeds $S5 = \{42, 123, 2025, 7, 2024\}$; confirmatory studies add $\{11, 19, 73, 314, 2718\}$ to form ten paired seeds (S10). We report mean$\pm$sample standard

Table 2: Primary SAGIN results. Values are mean±sample standard deviation across seeds. Standard S5 and reward-matched S10 are separate protocols; MaskMAPPO† observes exact decision-time feasibility.

Protocol	Method	SR (%) ↑	COV (%) ↓
Standard S5	**VERA**	**55.33±3.60**	**0.45±0.81**
	AB-MAPPO	50.68±4.65	45.32±9.42
	MAPPO	26.35±6.15	69.54±11.07
	Tuned constrained policy	40.17±3.46	33.65±8.67
	MaskMAPPO†	56.66±0.70	0.05
Reward-matched S10	**VERA**	**56.00±5.19**	**0.51±0.71**
	AB-MAPPO-R	47.26±6.34	52.70±10.55

deviation, two-sided paired tests for matched designs, paired standardized effect d_z, bootstrap confidence intervals, and Holm correction for topology families. Baselines include MAPPO (Yu et al., 2022), Beta-distribution MAPPO (B-MAPPO), AB-MAPPO, MADDPG (Lowe et al., 2017), a Lagrangian constrained policy (Achiam et al., 2017), and a privileged hard-mask reference. Vectorized Multi-Agent Simulator (VMAS) navigation (Bettini et al., 2022) supplies an independent control domain. Appendix A.3 gives all metrics, hyperparameters, and test definitions.

4.1 Primary performance and constraint compliance

Table 2 separates the standard complete-method benchmark from the stronger reward-matched comparison. In the standard S5 protocol, VERA is the best non-privileged method in success rate (SR), energy, Jain's fairness index, and coverage violation (COV), reaching $55.33 \pm 3.60\%$ SR with $0.45 \pm 0.81\%$ COV. Its success lies 1.33 points below MaskMAPPO, which receives an exact feasibility mask at decision time. The complete standard metrics and the eight-method comparison appear in Appendix Table 3 and Fig. 4.

Because the complete VERA configuration also changes the execution reward, the standard block alone cannot identify an architectural effect. In the reward-matched S10 protocol, both methods receive the same execution reward and budget. VERA improves SR by 8.74 points ($d_z = 0.864$, $p = 0.0231$, bootstrap 95% confidence interval $[3.20, 14.97]$), with a positive difference on 8/10 seeds. It reduces COV by 52.19 points ($d_z = -5.13$, $p = 5.7 \times 10^{-8}$, interval $[-58.13, -46.24]$), improving compliance on every seed. Exact sign-flip tests give $p = 0.0117$ and $p = 0.0020$. A nine-setting constrained-policy sweep reaches $40.17 \pm 3.46\%$ SR and $33.65 \pm 8.67\%$ COV, showing that the gap is not explained by an untuned penalty baseline.

4.2 Mechanism attribution

We first fix the execution reward and retrain seven variants (Appendix Table 7 and Fig. 6). Supervised auxiliary VFR reduces COV relative to an ungrounded five-dimensional latent head from 25.56% to 2.40% ($p = 0.042$). Feeding VFR directly to the actor adds no detected gain over auxiliary supervision. Removing CGRA, however, reduces SR from 54.57% to 40.12% ($p = 0.0023$) and raises COV from 2.33% to 62.23% ($p < 10^{-4}$). A retrained strict bottleneck also underperforms the complete actor, confirming that backbone access is a useful design choice rather than an accidental shortcut.

The S10 4×2 factorial makes this attribution independent of a single representation design (Fig. 2a; Appendix Table 9). Across handcrafted, learned, random-supervised, and latent heads, CGRA adds 14.91, 16.06, 16.48, and 14.16 SR points; every paired test has $p \leq 6.7 \times 10^{-6}$. COV changes range from -42.08 to -62.13 points. A repeated-measures Friedman test detects no difference among the four SR effects ($p = 0.323$). Coordinate removal, random-target, and permutation controls change SR by at most 1.74 points (Appendix Table 8 and Fig. 7). Collectively, these experiments identify CGRA as the main performance driver and VFR as the grounded interface that makes feasibility error and consistency auditable.

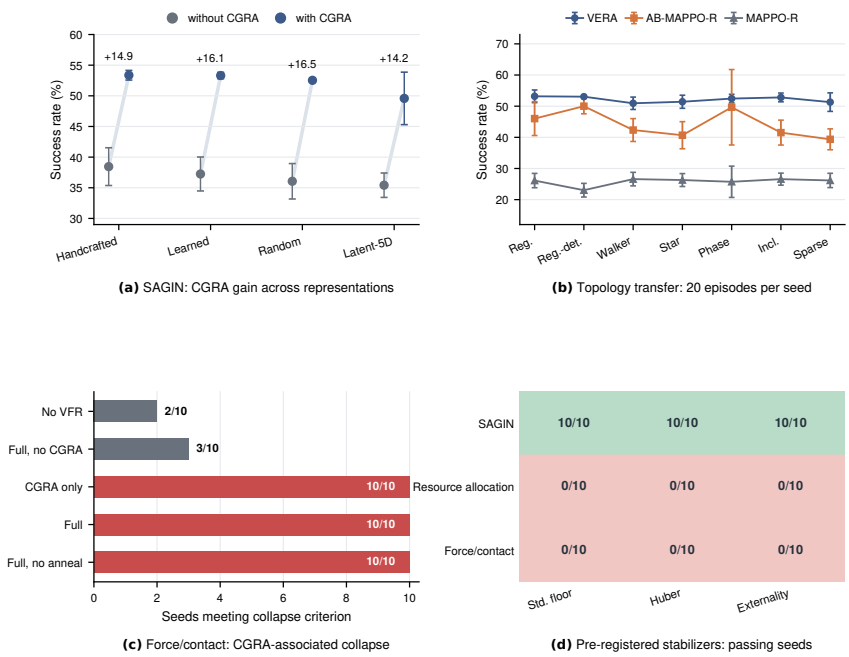


Figure 2: **Mechanism and scope.** (a) CGRA improves S10 SAGIN success across four representations. (b) VERA transfers across seven topologies; intervals use seed-level 95% confidence intervals from 20 episodes per seed. (c) All CGRA-bearing force/contact arms meet the pre-registered collapse rule. (d) Three stabilizers preserve SAGIN but pass neither cross-domain rescue gate. Standard-deviation bars are used in (a).

4.3 Topology transfer and feasibility calibration

The topology study evaluates 20 episodes for every seed–method–topology cell across seven unseen layouts, mirroring the procedural-generalization principle behind SMACv2 (Ellis et al., 2023). VERA remains at 50.93–53.17% SR, compared with 23.03–26.59% for reward-matched MAPPO (MAPPO-R) and 39.36–50.00% for AB-MAPPO-R (Fig. 2b). The paired VERA–MAPPO-R gap is 24.33–30.02 points and remains significant after Holm correction on all seven topologies. The VERA–AB-MAPPO-R difference is positive on all seven (2.80–11.95 points), although those individual comparisons do not survive family-wise correction.

Transfer follows the physical margin rather than a topology label. The calibration study evaluates 80,118 decisions on each of four shifted topologies and bins the minimum coverage/deadline margin. All three clearly negative bins have 100% violation, the boundary bin has 52.4–86.7%, and all four positive bins have 0%. Figure 8 and Appendix A.9 show the simultaneous decline in VFR error. This ordered transition is the empirical counterpart of Eq. (3).

4.4 Scaling and execution profile

The user-count sweep increases K from 10 to 40 while keeping six satellites. VERA sustains 53.6/53.8/50.1% SR, whereas AB-MAPPO falls from 37.6% to 26.8% and MAPPO remains at 32.1–35.7% (Fig. 3a). With $K = 20$ and 3/6/12 satellites, VERA obtains 55.1/53.8/53.0% SR; COV remains below 11% over all six scale settings. Without retraining, high load and link failure retain 100% of source SR, while a 0.10-s deadline and heavier tasks retain 47.8% and 46.3% as the feasible task set contracts (Fig. 3b).

Execution uses one actor pass. Central processing unit (CPU) latency is 0.939 ± 0.106 ms for VERA and 0.913 ± 0.086 ms for MAPPO; Compute Unified Device Architecture (CUDA) latency is 2.128 ± 0.649 and 2.073 ± 0.668 ms, respectively (Fig. 3c). Thus the VFR pathway adds 0.026 ms on CPU and 0.055 ms on CUDA. The $G = 8$ candidate scorer costs 6.94 ms on CPU and 8.81 ms on CUDA per update, but is absent from execution.

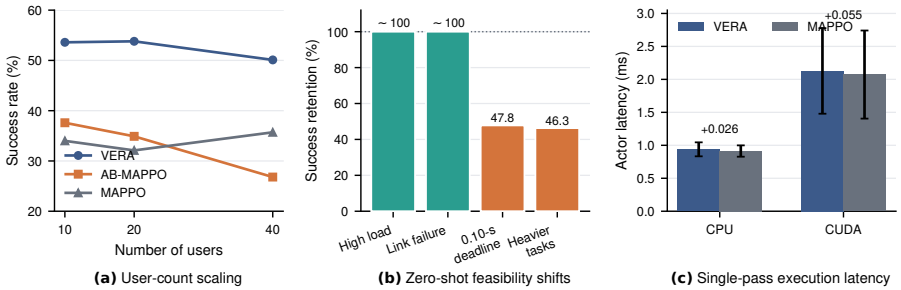


Figure 3: **Scale, zero-shot shifts, and execution cost.** (a) Mean SR as the number of users grows. (b) VERA SR retention under dimension-preserving shifts; the tighter deadline and heavier tasks also reduce intrinsic feasibility. (c) Per-step actor latency, reported as mean±sample standard deviation. Panel titles are placed below each subplot.

4.5 Reward geometry determines cross-domain behavior

VMAS cooperative navigation transfers the training pattern but not the superiority of fixed semantics. Full VERA reaches $24.75 \pm 4.09\%$ goal success versus $4.86 \pm 4.20\%$ without VFR/CGRA ($p = 0.0020$); a Latent-5D head reaches $28.41 \pm 3.69\%$ and exceeds Full by 3.66 points ($p = 0.038$). The 4,000-episode out-of-distribution (OOD) audit similarly shows a goal–collision trade-off rather than universal dominance (Appendix A.10 and Fig. 10c–d).

Two stress domains isolate the failure mechanism. In resource allocation, every CGRA arm converges near 63.47% SR but violates shared capacity on 100% of decisions; without CGRA, SR is 49.27–49.73% and COV is 1.43–5.70% across all four representations (Fig. 10a–b). The fixed-peer evaluator rewards over-demand because the local score omits the shared externality. In force/contact control, CGRA-only, Full, and Full without annealing collapse on 10/10 seeds, compared with 2/10 and 3/10 for the two CGRA-absent arms (Fig. 2c). Standard-deviation flooring, Huberized relative advantage, and an externality correction each preserve SAGIN on 10/10 seeds, yet pass neither rescue gate (Fig. 2d). These factorial and pre-registered tests turn transfer failure into a concrete design criterion: candidate evaluation must price the constraints that an individual action externalizes.

5 Conclusion

VERA separates two signals that end-to-end constrained policies commonly conflate: the return associated with an action and the physical reason that action can execute. Its VFR converts exact post-action feasibility calculations into a compact supervised interface, while CGRA converts within-state candidate comparisons into agent-specific credit. Neither training mechanism appears in the decentralized execution path.

The evidence supports three conclusions. First, the combined method is effective in dynamic SAGIN control: it reaches $55.33 \pm 3.60\%$ SR with $0.45 \pm 0.81\%$ COV, gains 8.74 SR points over the strongest reward-matched baseline on ten paired seeds, and reduces that baseline's violation by 52.19 points. Second, the factorial identifies where the gain enters. CGRA contributes 14.16–16.48 points across four representations, while VFR supplies the verifier-grounded interface through which margin error, action consistency, and topology transfer can be audited. Third, the method retains a 24.33–30.02 point gap over MAPPO-R on seven layouts, sustains 50.1–53.8% SR from 10 to 40 users with coverage violation below 11% over all six scale settings, and adds only 0.026 ms to a CPU actor step. The cross-domain results sharpen rather than dilute the contribution. CGRA succeeds when fixed-peer candidate scores preserve the ordering induced by the relevant physical constraints; it fails when locally favorable actions impose an unpriced shared externality. This criterion explains both the compliant SAGIN result and the resource/force failures, and it is testable before deployment through the same factorial and stress-test protocol. More broadly, VERA shows how simulator-accessible physics can supervise an auditable policy interface without becoming a privileged execution input. Extending the evaluator to learn or price constraint geometry is the natural next step toward general-purpose verifiable multi-agent control.

Reproducibility statement

The anonymous artifact includes the simulator, training and evaluation code, fixed seeds, run configurations, per-seed traces, checkpoints, frozen summaries, and independent recomputation tools. Appendix A documents the physical model, objectives, protocols, statistical tests, and artifact map. The ten-seed studies and 4,000-record VMAS audit are fully traceable; all 48 files in the trace archive pass SHA-256 verification.

AI use statement

Generative AI tools assisted with language editing, manuscript organization, LaTeX, reference discovery, and figure-code refinement. They were not used to generate experimental data, execute experiments, select reported outcomes, or change statistical analyses. The authors verified the manuscript, citations, equations, figures, and numerical claims against primary sources and archived artifacts and take full responsibility for the submission.

References

David Abel, Dilip Arumugam, Lucas Lehnert, and Michael Littman. State abstractions for lifelong reinforcement learning. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 10–19. PMLR, 2018.

Joshua Achiam, David Held, Aviv Tamar, and Pieter Abbeel. Constrained policy optimization. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, volume 70 of *Proceedings of Machine Learning Research*, pp. 22–31. PMLR, 2017.

Adilya Bakambekova, Nour Kouzayha, and Tareq Al-Naffouri. On the interplay of artificial intelligence and space-air-ground integrated networks: A survey. *IEEE Open Journal of the Communications Society*, 5:4613–4673, 2024. doi: 10.1109/OJCOMS.2024.3429198.

Matteo Bettini, Ryan Kortvelesy, Jan Blumenkamp, and Amanda Prorok. Vmas: A vectorized multi-agent simulator for collective robot learning. In *Distributed Autonomous Robotic Systems: 16th International Symposium*, volume 28 of *Springer Proceedings in Advanced Robotics*, pp. 42–56. Springer, 2022. doi: 10.1007/978-3-031-51497-5_4.

Yue Cai, Peng Cheng, Zhuo Chen, Wei Xiang, Branka Vucetic, and Yonghui Li. Graphic deep reinforcement learning for dynamic resource allocation in space-air-ground integrated networks. *IEEE Journal on Selected Areas in Communications*, 43(1):334–349, 2025. doi: 10.1109/JSAC.2024.3460086.

Zhepeng Cen, Yihang Yao, Zuxin Liu, and Ding Zhao. Feasibility consistent representation learning for safe reinforcement learning. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, volume 235 of *Proceedings of Machine Learning Research*, pp. 6002–6019. PMLR, 2024.

Lili Chen, Kevin Lu, Aravind Rajeswaran, Kimin Lee, Aditya Grover, Michael Laskin, Pieter Abbeel, Aravind Srinivas, and Igor Mordatch. Decision transformer: Reinforcement learning via sequence modeling. In *Advances in Neural Information Processing Systems 34 (NeurIPS)*, pp. 15084–15097, 2021.

Zhao Chen and Xiaodong Wang. Decentralized computation offloading for multi-user mobile edge computing: A deep reinforcement learning approach. *EURASIP Journal on Wireless Communications and Networking*, 2020, 2020. doi: 10.1186/s13638-020-01801-6.

Josef Dai, Xuehai Pan, Ruiyang Sun, Jiaming Ji, Xinbo Xu, Mickel Liu, Yizhou Wang, and Yaodong Yang. Safe RLHF: Safe reinforcement learning from human feedback. In *International Conference on Learning Representations (ICLR)*, 2024. URL https://openreview.net/forum?id=TyFrPOKYXw.

Quentin Delfosse, Sebastian Sztwiertnia, Mark Rothermel, Wolfgang Stammer, and Kristian Kersting. Interpretable concept bottlenecks to align reinforcement learning agents. In *Advances in Neural Information Processing Systems 37*, 2024. doi: 10.52202/079017-2135.

Dongsheng Ding, Xiaohan Wei, Zhuoran Yang, Zhaoran Wang, and Mihailo R. Jovanovic. Provably efficient generalized Lagrangian policy optimization for safe multi-agent reinforcement learning. In *Proceedings of the 5th Annual Learning for Dynamics and Control Conference (L4DC)*, volume 211 of *Proceedings of Machine Learning Research*, pp. 315–332. PMLR, 2023.

Mark Eisen and Alejandro Ribeiro. Optimal wireless resource allocation with random edge graph neural networks. *IEEE Transactions on Signal Processing*, 68:2977–2991, 2020. doi: 10.1109/TSP.2020.2988255.

Benjamin Ellis, Jonathan Cook, Skander Moalla, Mikayel Samvelyan, Mingfei Sun, Anuj Mahajan, Jakob N. Foerster, and Shimon Whiteson. Smacv2: An improved benchmark for cooperative multi-agent reinforcement learning. In *Advances in Neural Information Processing Systems 36 (NeurIPS Datasets and Benchmarks Track)*, pp. 37567–37593, 2023.

Jakob Foerster, Gregory Farquhar, Triantafyllos Afouras, Nantas Nardelli, and Shimon Whiteson. Counterfactual multi-agent policy gradients. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018. doi: 10.1609/aaai.v32i1.11794.

Wei Fu, Chao Yu, Zelai Xu, Jiaqi Yang, and Yi Wu. Revisiting some common practices in cooperative multi-agent reinforcement learning. In *Proceedings of the 39th International Conference on Machine Learning (ICML)*, volume 162 of *Proceedings of Machine Learning Research*, pp. 6863–6877. PMLR, 2022.

Yulan Gao, Ziqiang Ye, and Han Yu. Cost-efficient computation offloading in SAGIN: A deep reinforcement learning and perception-aided approach. *IEEE Journal on Selected Areas in Communications*, 42(12):3462–3476, 2024. doi: 10.1109/JSAC.2024.3459073.

Javier García and Fernando Fernández. A comprehensive survey on safe reinforcement learning. *Journal of Machine Learning Research*, 16(42):1437–1480, 2015. URL `https://jmlr.org/papers/v16/garcia15a.html`.

Shangding Gu, Jakub Grudzien Kuba, Muning Wen, Ruiqing Chen, Ziyan Wang, Zheng Tian, Jun Wang, Alois Knoll, and Yaodong Yang. Safe multi-agent reinforcement learning for multi-robot control. *Artificial Intelligence*, 319:103905, 2023. doi: 10.1016/j.artint.2023.103905.

Philippe Hansen-Estruch, Amy Zhang, Ashvin Nair, Patrick Yin, and Sergey Levine. Bisimulation makes analogies in goal-conditioned reinforcement learning. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 8407–8426. PMLR, 2022.

Chong Huang, Gaojie Chen, Pei Xiao, Yue Xiao, Zhu Han, and Jonathon A. Chambers. Joint offloading and resource allocation for hybrid cloud and edge computing in SAGINs: A decision assisted hybrid action space deep reinforcement learning approach. *IEEE Journal on Selected Areas in Communications*, 42(5):1029–1043, 2024. doi: 10.1109/JSAC.2024.3365899.

Liang Huang, Suzhi Bi, and Ying-Jun Angela Zhang. Deep reinforcement learning for online computation offloading in wireless powered mobile-edge computing networks. *IEEE Transactions on Mobile Computing*, 19(11):2581–2593, 2020. doi: 10.1109/TMC.2019.2928811.

Jiaming Ji, Borong Zhang, Jiayi Zhou, Xuehai Pan, Weidong Huang, Ruiyang Sun, Yiran Geng, Yifan Zhong, Juntao Dai, and Yaodong Yang. Safety-gymnasium: A unified safe reinforcement learning benchmark. In *Advances in Neural Information Processing Systems 36 (NeurIPS Datasets and Benchmarks Track)*, 2023a.

Jiaming Ji, Jiayi Zhou, Borong Zhang, Juntao Dai, Xuehai Pan, Ruiyang Sun, Weidong Huang, Yiran Geng, Mickel Liu, and Yaodong Yang. Omnisafe: An infrastructure for accelerating safe reinforcement learning research. *Journal of Machine Learning Research*, 25(285):1–6, 2024.

Zhe Ji, Sheng Wu, and Chunxiao Jiang. Cooperative multi-agent deep reinforcement learning for computation offloading in digital twin satellite edge networks. *IEEE Journal on Selected Areas in Communications*, 41(11):3414–3429, 2023b. doi: 10.1109/JSAC.2023.3313595.

Eunji Kim, Dahuin Jung, Sangha Park, Siwon Kim, and Sungroh Yoon. Probabilistic concept bottleneck models. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 16521–16540. PMLR, 2023. URL https://proceedings.mlr.press/v202/kim23g.html.

Pang Wei Koh, Thao Nguyen, Yew Siang Tang, Stephen Mussmann, Emma Pierson, Been Kim, and Percy Liang. Concept bottleneck models. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 5338–5348. PMLR, 2020. URL https://proceedings.mlr.press/v119/koh20a.html.

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*, 2022. URL https://openreview.net/forum?id=EcGGFkNTxdJ.

Charline Le Lan, Stephen Tu, Adam Oberman, Rishabh Agarwal, and Marc G. Bellemare. On the generalization of representations in reinforcement learning. In *Proceedings of the 25th International Conference on Artificial Intelligence and Statistics*, volume 151 of *Proceedings of Machine Learning Research*, pp. 4132–4157. PMLR, 2022.

Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In *International Conference on Learning Representations (ICLR)*, 2024. URL https://openreview.net/forum?id=v8L0pN6EOi.

Jiajia Liu, Yongpeng Shi, Zubair Md. Fadlullah, and Nei Kato. Space-air-ground integrated network: A survey. *IEEE Communications Surveys & Tutorials*, 20(4):2714–2741, 2018. doi: 10.1109/COMST.2018.2841996.

Jiarong Liu, Yifan Zhong, Siyi Hu, Haobo Fu, Qiang Fu, Xiaojun Chang, and Yaodong Yang. Maximum entropy heterogeneous-agent reinforcement learning. In *International Conference on Learning Representations (ICLR)*, 2024. URL https://openreview.net/forum?id=tmqOhBC4a5.

Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. Understanding r1-zero-like training: A critical perspective. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, Proceedings of Machine Learning Research. PMLR, 2025.

Ryan Lowe, Yi Wu, Aviv Tamar, Jean Harb, Pieter Abbeel, and Igor Mordatch. Multi-agent actor-critic for mixed cooperative-competitive environments. In *Advances in Neural Information Processing Systems 30 (NeurIPS)*, 2017.

Yifeng Lyu, Han Hu, Rongfei Fan, Zhi Liu, Jianping An, and Shiwen Mao. Dynamic routing for integrated satellite-terrestrial networks: A constrained multi-agent reinforcement learning approach. *IEEE Journal on Selected Areas in Communications*, 42(5):1204–1218, 2024. doi: 10.1109/JSAC.2024.3365869.

Yuyi Mao, Changsheng You, Jun Zhang, Kaibin Huang, and Khaled B. Letaief. A survey on mobile edge computing: The communication perspective. *IEEE Communications Surveys & Tutorials*, 19 (4):2322–2358, 2017. doi: 10.1109/COMST.2017.2745201.

Linghui Meng, Muning Wen, Chenyang Le, Xiyun Li, Dengpeng Xing, Weinan Zhang, Ying Wen, Haifeng Zhang, Jun Wang, Yaodong Yang, and Bo Xu. Offline pre-trained multi-agent decision transformer. *Machine Intelligence Research*, 20(2):233–248, 2023. doi: 10.1007/s11633-022-1383-7.

Zhaolong Ning, Yuxuan Yang, Xiaojie Wang, Lei Guo, Xinbo Gao, Song Guo, and Guoyin Wang. Dynamic computation offloading and server deployment for UAV-enabled multi-access edge computing. *IEEE Transactions on Mobile Computing*, 22(5):2628–2644, 2023. doi: 10.1109/TMC.2021.3129785.

Frans A. Oliehoek and Christopher Amato. *A Concise Introduction to Decentralized POMDPs*. SpringerBriefs in Intelligent Systems. Springer International Publishing, 2016.

Bei Peng, Tabish Rashid, Christian Schroeder de Witt, Pierre-Alexandre Kamienny, Philip Torr, Wendelin Boehmer, and Shimon Whiteson. FACMAC: Factored multi-agent centralised policy gradients. In *Advances in Neural Information Processing Systems 34*, 2021.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In *Advances in Neural Information Processing Systems 36 (NeurIPS)*, pp. 53728–53741, 2023.

Tabish Rashid, Mikayel Samvelyan, Christian Schroeder de Witt, Gregory Farquhar, Jakob Foerster, and Shimon Whiteson. Monotonic value function factorisation for deep multi-agent reinforcement learning. *Journal of Machine Learning Research*, 21(178):1–51, 2020.

John Schulman, Philipp Moritz, Sergey Levine, Michael Jordan, and Pieter Abbeel. High-dimensional continuous control using generalized advantage estimation. In *International Conference on Learning Representations (ICLR)*, 2016. URL https://arxiv.org/abs/1506.02438.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms, 2017. URL https://arxiv.org/abs/1707.06347.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models, 2024. URL https://arxiv.org/abs/2402.03300.

Yifei Shen, Yuanming Shi, Jun Zhang, and Khaled B. Letaief. A graph neural network approach for scalable wireless power control. In *2019 IEEE Globecom Workshops (GC Wkshps)*, pp. 1–6, 2019. doi: 10.1109/GCWKSHPS45667.2019.9024538.

Aivar Sootla, Alexander I. Cowen-Rivers, Taher Jafferjee, Ziyan Wang, David Mguni, Jun Wang, and Haitham Bou-Ammar. Saute RL: Almost surely safe reinforcement learning using state augmentation. In *Proceedings of the 39th International Conference on Machine Learning (ICML)*, volume 162 of *Proceedings of Machine Learning Research*, pp. 20423–20443. PMLR, 2022.

Fengxiao Tang, Hans Hofner, Nei Kato, Kazuma Kaneko, Yasutaka Yamashita, and Masatake Hangai. A deep reinforcement learning-based dynamic traffic offloading in space-air-ground integrated networks (SAGIN). *IEEE Journal on Selected Areas in Communications*, 40(1):276–289, 2022. doi: 10.1109/JSAC.2021.3126073.

Zizhao Wang, Xuesu Xiao, Zifan Xu, Yuke Zhu, and Peter Stone. Causal dynamics learning for task-independent state abstraction. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 23151–23180. PMLR, 2022.

Muning Wen, Jakub Grudzien Kuba, Runji Lin, Weinan Zhang, Ying Wen, Jun Wang, and Yaodong Yang. Multi-agent reinforcement learning is a sequence modeling problem. In *Advances in Neural Information Processing Systems 35 (NeurIPS)*, 2022.

Zifan Wu, Bo Tang, Qian Lin, Chao Yu, Shangqin Mao, Qianlong Xie, Xingxing Wang, and Dong Wang. Off-policy primal-dual safe reinforcement learning. In *International Conference on Learning Representations (ICLR)*, 2024. URL https://openreview.net/forum?id=vy42bYs1Wo.

Long Yang, Jiaming Ji, Juntao Dai, Linrui Zhang, Binbin Zhou, Pengfei Li, Yaodong Yang, and Gang Pan. Constrained update projection approach to safe policy optimization. In *Advances in Neural Information Processing Systems 35 (NeurIPS)*, 2022.

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 35 (NeurIPS)*, pp. 24611–24624, 2022. doi: 10.52202/068431-1787.

Amy Zhang, Clare Lyle, Shagun Sodhani, Angelos Filos, Marta Kwiatkowska, Joelle Pineau, Yarin Gal, and Doina Precup. Invariant causal prediction for block MDPs. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 11214–11224. PMLR, 2020.

Yifan Zhong, Jakub Grudzien Kuba, Xidong Feng, Siyi Hu, Jiaming Ji, and Yaodong Yang. Heterogeneous-agent reinforcement learning. *Journal of Machine Learning Research*, 2024.

A Extended Experimental and Implementation Details

This appendix is part of the submission portable document format (PDF) file and provides the physical model, objectives, complete protocols, secondary analyses, and experiment-to-artifact map needed to audit the main claims. All plots use the final ten-seed, 20-episode, or 4,000-record analyses where those higher-precision evaluations are available.

A.1 Environment and physical model

The benchmark is a three-tier space–air–ground integrated network with K ground users, M_{sat} LEO satellites, and M_{uav} UAV nodes. The standard configuration is $K = 20$, $M_{sat} = 6$, and $M_{uav} = 4$. LEO altitude is 550 km, UAV altitude is 20 km, the episode horizon is 200 slots, and the minimum satellite elevation is 10°. Satellite angular velocity is $\omega = \sqrt{\mu/r^3}$ for $r = R_E + h_{sat}$, corresponding to an approximately 95-minute orbit. Visibility ends when ground-projected distance exceeds the elevation-implied threshold.

At each slot, agent k receives a task with probability 0.8. Task size is sampled from $[5 \times 10^4, 2 \times 10^5]$ bits and compute intensity from $[500, 1500]$ cycles/bit. The standard deadline is $D_{\max} = 0.15$ s. For target n, the transmission rate is

$$R_{k,n}(t) = B_{k,n}(t) \log_2\left(1 + \frac{P_k g_{k,n}(t)}{N_0 B_{k,n}(t)}\right). \tag{9}$$

For offloading ratio ρ_k and target α_k, remote time is

$$T_k^{off} = \frac{\rho_k L_k}{R_{k,\alpha_k}} + \frac{\rho_k L_k C_k}{f_{\alpha_k}}, \tag{10}$$

local time is $T_k^{loc} = (1 - \rho_k) L_k C_k / f_k^{loc}$, and task latency is $T_k = \max(T_k^{loc}, T_k^{off})$. A remote task succeeds only if its link remains visible for the required transmission interval. Concurrent bandwidth requests are allocated proportionally, so shared-resource contention is explicit. The simulator uses analytic orbital motion and a regional ground-track abstraction; it omits explicit Doppler re-synchronization and inter-cell interference.

A.2 Feasibility-margin bound

For Eq. (3), let $E = \{\|\hat{\boldsymbol{m}} - \boldsymbol{m}\|_\infty \le \epsilon\}$ and $B = \{\min_j |m_j| > \epsilon\}$. On $E \cap B$, $|\hat{m}_j - m_j| \le \epsilon < |m_j|$ for every coordinate; hence no prediction crosses zero and every feasibility sign is preserved. Any sign mismatch is therefore contained in $E^c \cup B^c$, and the union bound gives Eq. (3). The binary coverage-validity entry in the implementation is obtained by thresholding a signed contact-time margin, so the statement applies to its pre-threshold value.

A.3 Rewards, protocol, and statistics

Ordinary baselines use

$$r^{base} = 0.5(10SR) + 0.25\frac{5}{1+T} + 0.25\,5\max(0, 1 - E/E_{th}). \tag{11}$$

VERA's complete execution layer uses nonlinear success shaping, threshold bonuses, latency and energy terms, and a squared coverage penalty,

$$r_{VERA}^{exec} = \lambda_{sr}\big(20SR^{1.2} + b(SR)\big) + \lambda_T r_T + \lambda_E r_E \tag{12}$$

$$- 10\lambda_{cov} v_{cov}^2, \tag{13}$$

followed by validity and consistency rewards. The complete-method table is therefore not an architecture-only comparison. Reward-matched controls receive the same execution reward; the mechanism ladder further holds that reward fixed while varying VFR supervision, conditioning, consistency, bottlenecking, and CGRA.

All proximal-policy-optimization-family methods use centralized training and decentralized execution, Kullback–Leibler (KL) early stopping, learning-rate and entropy annealing, and value clipping. The primary five-seed set is $S5 = \{42, 123, 2025, 7, 2024\}$. Confirmatory S10 results add $\{11, 19, 73, 314, 2718\}$. Unless a subsection states otherwise, training runs for 3,000 episodes and the final 500 episodes form each seed-level estimate (the scale sweeps use a 500-episode budget and report the converged window, episodes > 250). We report mean$\pm$sample standard deviation across seeds. Matched designs use two-sided paired t tests, paired standardized effect d_z, paired bootstrap intervals, and exact sign-flip tests where reported. We do not interpret $p > 0.05$ as equivalence. The topology study uses 20 evaluation episodes per seed; the VMAS OOD study is an inference-only audit on the five new seeds with 100 episodes for every variant–shift cell, not a ten-seed training replication.

A.4 Complete standard-configuration results

Table 3: Standard configuration, final-500-episode window, five seeds. Latency is averaged over successful tasks and must be read with success rate.

Method	SR (%)	Latency (s)	Energy	Cov. viol. (%)	Jain
VERA	55.33±3.60	0.1010±0.0003	0.018±0.015	0.45±0.81	0.9936±0.0007
AB-MAPPO	50.68±4.65	0.0956±0.0028	8.482±2.689	45.32±9.42	0.9825±0.0092
B-MAPPO	49.14±5.39	0.0950±0.0040	11.849±10.424	46.65±10.16	0.9841±0.0065
MAPPO	26.35±6.15	0.0874±0.0027	55.851±40.247	69.54±11.07	0.9779±0.0057
MADDPG	21.10±6.63	0.0722±0.0076	6.219±1.951	91.74±10.19	0.8893±0.0335
Random	14.86±1.51	0.0829±0.0025	179.987±13.650	97.17±1.95	0.9447±0.0154
MaskMAPPO	56.66±0.70	–	–	0.05	–
Tuned ConstrainedMAPPO	40.17±3.46	–	–	33.65±8.67	–

Table 3 reports every standard metric, and Fig. 4 visualizes the success comparison.

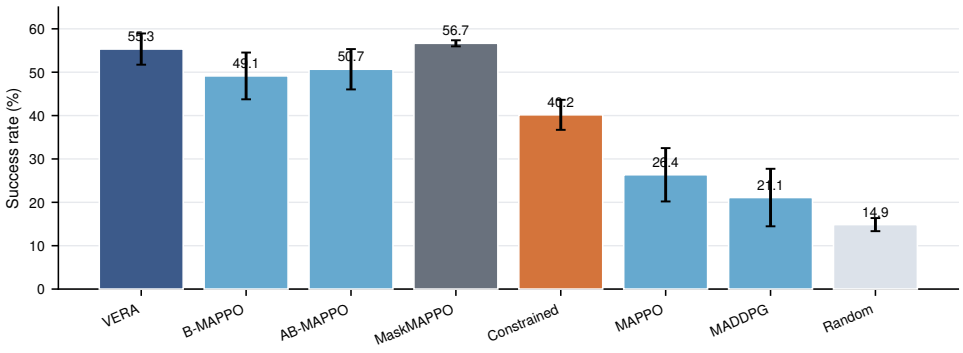


Figure 4: Standard-configuration success across the complete method comparison. MaskMAPPO uses a privileged hard feasibility mask and is shown as a reference rather than a deployable competitor; the constrained baseline is the tuned configuration from Table 2. Error bars are sample standard deviations across five seeds.

The paired Full–w/o-VFR same-objective comparison gives $+2.58$ success points ($p = 0.218$, $d_z = 0.65$) and -10.31 violation points ($p = 0.060$, $d_z = -1.17$). A 200,000-replicate paired bootstrap gives intervals $[-1.0, 5.3]$ and $[-17.1, -3.4]$ points; exact five-pair sign-flip p values are 0.25 and 0.125. These coarse five-seed tests motivate the larger reward-matched and factorial studies rather than supporting a strong mean claim by themselves.

A.5 SAME-OBJECTIVE ABLATION AND ZERO-SHOT SHIFTS

Table 4 isolates VFR conditioning, the coverage coordinate, and the verification reward before the larger mechanism ladder and factorial tests:

Table 4: Five-seed ablation under the VERA execution objective.

Variant	SR (%)	Cov. viol. (%)	VFR validity (%)
Full	54.57±2.26	2.33±1.44	82.63±2.73
w/o VFR	51.98±3.00	12.65±9.03	–
w/o coverage coordinate	53.45±0.51	0.78±0.65	87.39±0.83
w/o verification reward	37.10±6.07	60.67±13.16	54.93±5.35

Figure 5 shows all three ablation metrics with a common visual style.

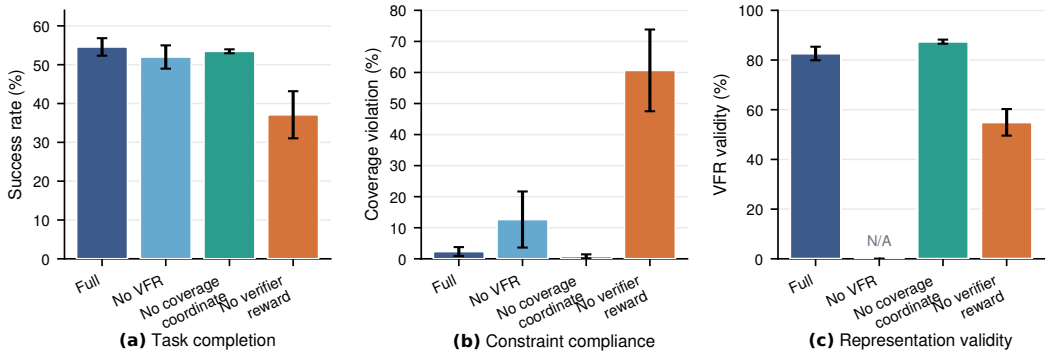


Figure 5: Same-objective ablation of success, coverage violation, and representation validity. The larger reward-matched ladder and S10 factorial provide the stronger mechanism tests.

With trained weights fixed, we also test high load, link failure, a tightened 0.10-s deadline, and heavier tasks. VERA retains 100% of source success under high load and link failure, but 47.8% and 46.3% under the tighter deadline and heavier tasks. The latter shifts also increase the fraction of structurally infeasible tasks. Figure 3b summarizes these shifts.

A.6 TEN-SEED REWARD MATCHING AND CONSTRAINED-BASELINE TUNING

Table 5 repeats the reward-matched means alongside the paired analysis below. The paired differences

Table 5: Reward-matched SAGIN comparison over ten paired seeds.

Method	SR (%)	Cov. viol. (%)
VERA	56.00±5.19	0.51±0.71
AB-MAPPO-R	47.26±6.34	52.70±10.55

are $+8.74$ success points ($d_z = 0.864$, $p = 0.0231$, bootstrap 95% CI $[3.20, 14.97]$) and -52.19 violation points ($d_z = -5.13$, $p = 5.7 \times 10^{-8}$, CI $[-58.13, -46.24]$). Exact sign-flip p values are 0.0117 and 0.0020. VERA improves success on 8/10 seeds and compliance on 10/10.

ConstrainedMAPPO is tuned over nine combinations of dual learning rate, constraint budget, and penalty initialization. The selected setting ($\mathrm{lr}_\lambda = 0.3$, budget 0.06, initialization 1.5) is rerun on S5 and produces the value in Table 2; selection and final evaluation are stored separately.

A.7 MECHANISM LADDER, SEMANTIC CONTROLS, AND S10 FACTORIAL

The reward-matched mechanism ladder uses the switches in Table 6.

Table 6: Component switches in the reward-matched mechanism ladder.

Variant	VFR supervision	VFR enters action	Consistency	CGRA
Backbone	no	no	no	no
Latent-5D	no	yes	no	yes
Aux-VFR	yes	no	no	yes
VFR-cond.	yes	yes	no	yes
w/o CGRA	yes	yes	yes	no
Strict bottleneck	yes	yes, no backbone bypass	yes	no
Full	yes	yes	yes	yes

Table 7: Reward-matched mechanism ladder, five seeds. Aux-VFR is supervised but not fed to the action head; VFR-cond. adds that pathway.

Variant	SR (%) ↑	Cov. viol. (%) ↓	VFR valid. (%) ↑
Backbone	36.79±2.33	62.50±6.27	–
Latent-5D	45.89±8.14	25.56±17.55	17.60±10.98
Aux-VFR	52.66±0.62	2.40±1.74	85.51±4.03
VFR-cond.	54.65±4.35	2.73±4.40	79.32±3.52
w/o CGRA	40.12±5.42	62.23±8.99	54.67±6.24
Strict bottleneck	42.25±1.81	52.05±6.94	49.75±4.02
Full	**54.57±2.26**	**2.33±1.44**	82.63±2.73

Full versus Backbone gives $+17.77$ success points ($p = 5.5 \times 10^{-4}$) and -60.17 violation points ($p = 4.1 \times 10^{-5}$). Aux-VFR versus Latent-5D gives -23.16 violation points ($p = 0.042$); VFR-cond. versus Aux-VFR has no detected difference. Full versus w/o CGRA gives $+14.45$ success points ($p = 0.0023$) and -59.89 violation points ($p < 10^{-4}$). Separately, the coordinate-control study removes each named coordinate and tests random or permuted supervised targets. Success changes by only 0.57–1.74 points and violation stays within 0.33 points of Full, so no individual named coordinate is load-bearing in this benchmark. Figure 6 visualizes the complete ladder.

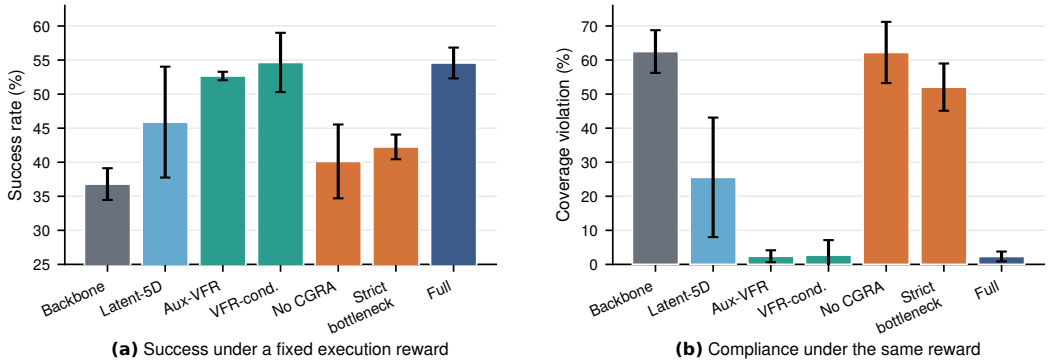


Figure 6: Seven-way reward-matched mechanism ladder. (a) Converged success. (b) Coverage violation. Error bars are sample standard deviations across five seeds; panel titles appear below the subplots.

The coordinate controls in Table 8 make the scope of the semantic claim explicit:

Figure 7 shows that no single coordinate removal creates a large success change.

Table 8: Five-seed semantic-coordinate controls. Small changes relative to Full rule out a claim that any single named coordinate is the primary performance driver.

Variant	SR (%)	ΔSR (pp)	Cov. viol. (%)
Full	54.25±1.29	0.00	2.78±0.68
Drop coverage margin	53.10±0.60	-1.15	2.46±0.71
Drop energy	53.21±0.94	-1.04	2.97±0.70
Drop coverage-valid	53.22±0.76	-1.03	2.67±1.09
Drop compute	53.22±0.37	-1.03	2.54±0.43
Drop delay	52.51±0.26	-1.74	2.50±0.87
Random supervised target	52.51±0.40	-1.74	2.75±1.04
Permuted coordinates	53.68±1.37	-0.57	2.45±0.81

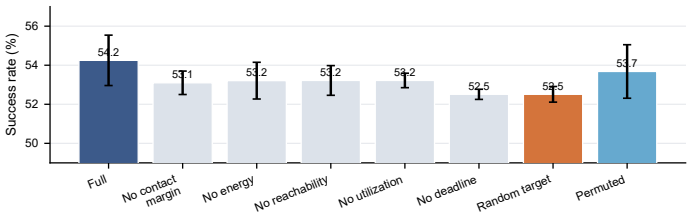


Figure 7: Success under coordinate removal, random-target, and coordinate-permutation controls. Error bars are sample standard deviations over five seeds.

Table 9 crosses four representations with CGRA on/off over S10:

Table 9: S10 representation–CGRA factorial in SAGIN. COV is coverage violation.

Representation	CGRA SR	no-CGRA SR	ΔSR	CGRA COV	no-CGRA COV
Handcrafted	53.36±0.79	38.45±3.08	+14.91	2.37±0.63	62.51±5.27
Learned	53.31±0.52	37.25±2.77	+16.06	2.32±0.70	64.45±5.66
Random	52.53±0.32	36.06±2.89	+16.48	2.73±0.96	59.60±9.56
Latent-5D	49.58±4.28	35.42±1.99	+14.16	19.32±12.07	61.40±2.89

All four paired success effects exclude zero; the repeated-measures interaction test gives $p = 0.323$. Random targets nearly match handcrafted targets when CGRA is active, while the latent head has substantially worse compliance. These observations support the two-part interpretation used in the paper: CGRA supplies most of the SAGIN performance gain, and grounding supplies the auditable feasibility interface.

A.8 High-precision topology evaluation

The high-precision study runs 20 episodes for every seed–method–topology cell. Intervals in Fig. 2b are seed-level 95% confidence intervals; Table 10 reports mean±sample standard deviation.

Table 10: Absolute success rate (%) over seven topologies.

Topology	VERA	AB-MAPPO-R	MAPPO-R
Regional	53.17±2.84	46.01±7.59	26.12±3.22
Regional deterministic	53.05±0.21	50.00±3.42	23.03±3.07
Walker	50.93±2.78	42.34±5.13	26.59±3.04
Walker-star	51.42±2.94	40.68±6.08	26.29±2.90
Phase shift	52.44±1.92	49.65±16.93	25.73±7.02
Inclination shift	52.81±1.97	41.52±5.59	26.58±2.70
Sparse shell	51.30±4.16	39.36±4.69	26.16±3.21

VERA–MAPPO-R differences are 24.33–30.02 points and remain significant after Holm correction on all seven topologies. VERA–AB-MAPPO-R differences are 2.80–11.95 points and directionally positive on all seven; none is Holm-significant. The latter is not described as a confirmed superiority claim.

A.9 CROSS-TOPOLOGY SIGNED-MARGIN CALIBRATION

The signed-margin study evaluates 80,118 decisions on each of Walker, Walker-star, inclination-shift, and sparse-shell, then bins $m = \min(m_{coverage}, m_{delay})$ into eight quantiles. For every topology, the three most negative bins have 100% violation, the boundary bin has 52.4–86.7%, and all four positive bins have 0%. Table 11 reports the aggregated bins. Violation is monotone non-increasing

Table 11: Signed-margin calibration aggregated over the source topology.

Bin	Margin mean	VFR error	Coverage flip (%)	Violation (%)
0	-0.740	0.115	8.5	100.0
1	-0.456	0.068	0.3	100.0
2	-0.219	0.071	0.3	100.0
3	-0.025	0.069	0.4	63.8
4	0.133	0.063	0.0	0.0
5	0.270	0.057	0.0	0.0
6	0.394	0.055	0.0	0.0
7	0.534	0.045	0.0	0.0

across bins in all four topologies. VFR error is largest in the most negative region and decreases overall toward positive margins, though it need not decrease at every adjacent bin. Figure 8 shows the source-topology transition.

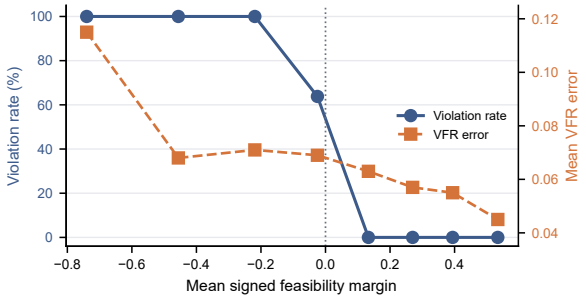


Figure 8: Source-topology signed-margin calibration. The violation transition occurs around the feasibility boundary; the preceding table verifies the same ordering on four shifted topologies.

A.10 VMAS TRANSFER AND NORMALIZED OOD AUDIT

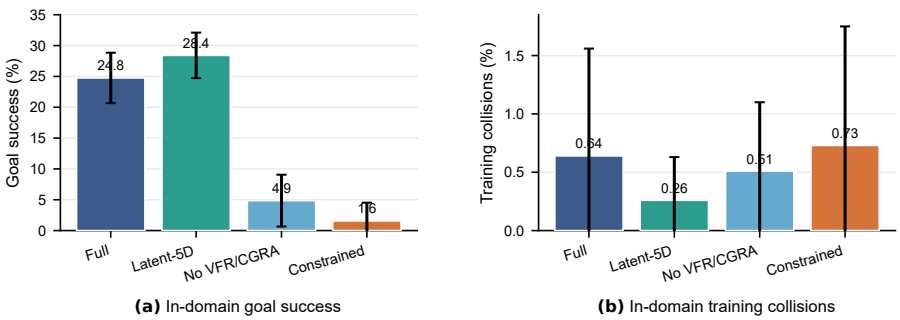
The cross-domain study ports the training pattern to VMAS cooperative navigation. The five supervised quantities are collision margin, goal reachability, a safety indicator, speed margin, and time buffer. Table 12 reports the in-domain result. Full exceeds the no-VFR/no-CGRA backbone

Table 12: VMAS in-domain performance over five training seeds.

Variant	Goal success (%)	Training collision events (%)
Full	24.75±4.09	0.64±0.92
Latent-5D	28.41±3.69	0.26±0.37
No VFR / no CGRA	4.86±4.20	0.51±0.59
Reward-only constrained	1.56±2.96	0.73±1.02

by 19.88 points ($p = 0.0020$), while Latent-5D exceeds Full by 3.66 points ($p = 0.038$). The result isolates transfer of the verification/credit pattern and shows that learned latent coordinates remain competitive outside SAGIN.

Figure 9 visualizes the goal-success comparison and the in-domain collision rates.



(a) In-domain goal success **(b)** In-domain training collisions

Figure 9: VMAS in-domain performance over five training seeds: (a) goal success and (b) training collision events. In-domain collisions stay low for every variant, so the safety difference is driven by the out-of-distribution audit. The latent representation exceeds the hand-specified semantic head, which is retained as a negative result for universal semantic superiority.

The separate OOD inference audit uses preserved checkpoints, the five new seeds, two shifts, and 100 episodes per cell. Its comma-separated values (CSV) file has 4,000 episode rows (4 variants $\times 5$ seeds $\times 2$ shifts $\times 100$ episodes). Collision is $100\times$events/agent-steps, hence bounded and comparable across agent counts. Table 13 reports both shifts.

Table 13: VMAS OOD audit, mean±sample standard deviation across five inference seeds.

Variant	Density goal	Density collision	Scale goal	Scale collision
Full	36.67±13.94	1.93±3.25	20.00±32.60	4.10±3.54
No VFR/CGRA	20.00±21.73	2.60±4.56	10.00±13.69	0.00±0.00
Latent-5D	53.33±21.73	6.60±11.20	50.00±39.53	7.50±6.55
Constrained	23.33±22.36	7.80±7.15	20.00±27.39	1.60±3.05

Latent-5D has the highest goal rate on both shifts but also the highest collision rate; Full does not dominate every metric. Because this audit covers only the new five seeds and reuses trained checkpoints, it is kept separate from S10 training claims.

A.11 Resource-allocation and force/contact failures

The S10 resource-allocation factorial uses the same four representations and CGRA switch as the SAGIN factorial. Every CGRA arm reaches 63.47% success and exactly 100% capacity violation. Removing CGRA lowers success but restores compliance (Table 14):

Table 14: S10 resource-allocation factorial.

Representation	CGRA SR	no-CGRA SR	CGRA COV	no-CGRA COV
Handcrafted	63.47±0.06	49.73±1.25	100.00±0.00	5.70±11.59
Learned	63.47±0.06	49.57±0.52	100.00±0.00	2.65±4.31
Random	63.47±0.06	49.29±0.71	100.00±0.00	1.43±0.63
Latent-5D	63.47±0.06	49.27±0.63	100.00±0.00	2.07±1.47

CGRA's success effects are $+13.74$–$+14.20$ points and its violation effects are $+94.30$–$+98.57$ points. Neither success nor violation shows a detected representation interaction ($p = 0.564$ and $p = 0.392$). This is reward hacking: proportional allocation and the flat over-capacity penalty make increased demand advantageous to the local counterfactual evaluator.

The force/contact study applies the pre-registered collapse rule—success below 5% for five consecutive checkpoints after previously exceeding 20%—to 50 S10 traces. Table 15 gives the per-arm attribution. All CGRA-bearing arms collapse, but five collapses also occur among 20 CGRA-absent traces. The attribution is therefore strong and heterogeneous, not an exclusive single-cause law.

Table 15: Force/contact collapse attribution over ten seeds per arm.

Arm	Collapsed seeds	Converged SR (%)	Converged COV (%)
No VFR/CGRA	2/10	47.42±13.52	26.59
Full, no CGRA	3/10	48.11±10.12	29.32
CGRA only	10/10	2.44±1.65	4.63
Full	10/10	0.87±1.68	4.07
Full, no annealing	10/10	2.33±3.51	4.33

The stabilizer study tests standard-deviation flooring, a Huberized relative advantage, and externality correction under pre-registered gates. On SAGIN, all three pass on 10/10 seeds: success is 53.09–53.44% and violation 2.05–2.54%. In resource allocation, all variants remain at 100% violation and pass on 0/10 seeds. In force/contact, the best mean success is 4.06% and every stabilizer passes on 0/10 seeds. Huberization is a no-op in both failed domains under the observed residuals; the externality term is also a no-op in force/contact. The unmodified CGRA is retained because none of the tested alternatives satisfies all gates. Figure 10 combines the resource factorial with the VMAS OOD audit.

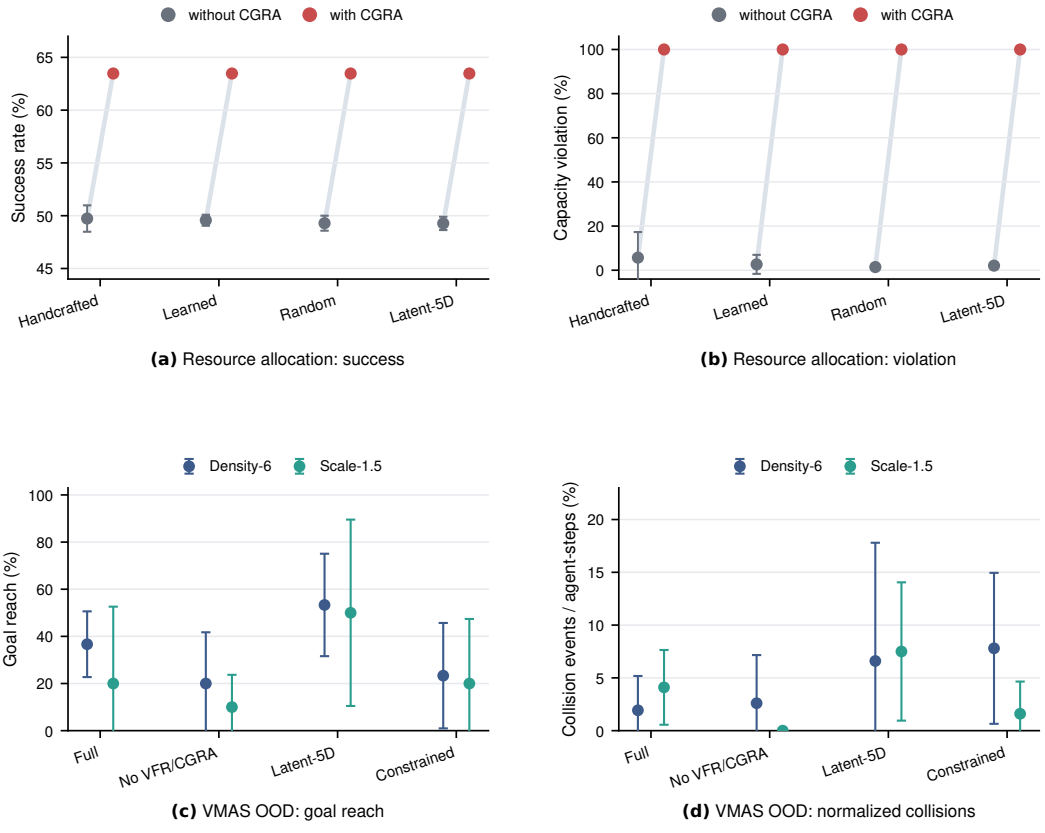


Figure 10: **Measured transfer boundary.** S10 resource allocation gains success by violating shared capacity across all four representations (a,b). In the 4,000-row VMAS OOD audit, Latent-5D has the highest goal reach but also the highest normalized collision rate (c,d). Error bars are sample standard deviations across seeds.

A.12 Inference diagnostics, scale, and deployment cost

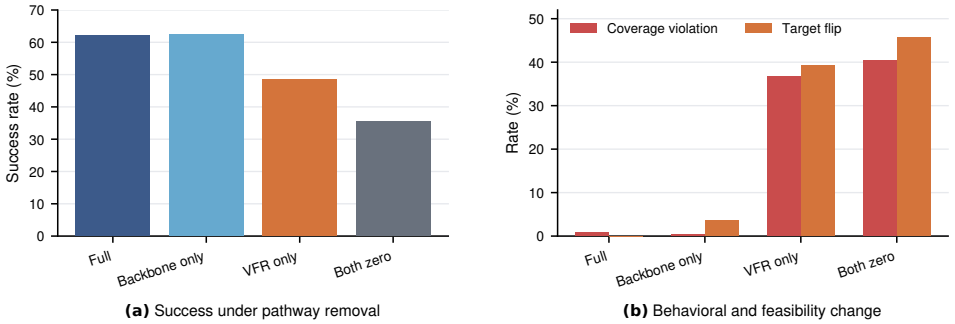
The inference pathway diagnostic uses three trained checkpoints and four held-out environment seeds, with 100 deterministic steps per arm. Table 16 reports the interventions. The actor was trained with both pathways; this is a diagnostic, not a retrained bottleneck comparison. Exact-autograd Jacobians

Table 16: Inference-pathway intervention results.

Arm	SR (%)	Cov. viol. (%)	Target flip vs. Full (%)
Full	61.98	0.73	0.0
Backbone-only	62.43	0.34	3.5
VFR-only	48.58	36.67	39.2
Both-zero	35.56	40.33	45.6

remain locally sensitive to VFR, yet the backbone can preserve most decisions when explicit VFR input is zeroed. The ladder's retrained strict-bottleneck arm supplies the stronger intervention.

Figure 11 visualizes the corresponding success and behavioral changes.



(a) Success under pathway removal **(b)** Behavioral and feasibility change

Figure 11: Inference-time pathway decomposition. The table additionally reports the both-zero arm; this diagnostic is interpreted jointly with the retrained ladder.

Across user counts $K = 10, 20, 40$ at six satellites, VERA success is 53.6/53.8/50.1%, versus 37.6/34.9/26.8% for AB-MAPPO and 34.0/32.1/35.7% for MAPPO. Across satellite counts $M_{sat} = 3, 6, 12$ at $K = 20$, VERA success is 55.1/53.8/53.0%; violation remains below 11% across all six scales. Under zero-shot high load, link failure, a 0.10-s deadline, and heavier tasks, success retention is 100%, 100%, 47.8%, and 46.3%. The latter two shifts also increase the proportion of structurally infeasible tasks. Figure 3a–b consolidates the scale and shift results.

On CPU, execution costs 0.939 ± 0.106 ms per step for VERA and 0.913 ± 0.086 ms for MAPPO; on CUDA, the values are 2.128 ± 0.649 and 2.073 ± 0.668 ms. A CGRA training update costs 6.94 ms on CPU and 8.81 ms on CUDA because it evaluates $G = 8$ candidates. Candidate scoring and environment verification are absent from execution; Fig. 3c shows the actor latency.

A.13 Archive depth and experiment map

The code/data archive separates immutable experiment results from manuscript-generation scripts. Table 17 maps every headline result to its frozen summary and seed-level records.

Table 17: Experiment-to-artifact map in the anonymous code/data archive.

Result	Archive path
SAGIN representation factorial	`code/results/U4_identifiable_schema`
Topology evaluation	`data/U1_U6/U5_topology_20ep`
VMAS OOD audit	`data/E17_E24/E20_vmas_ood_normalized`
Force/contact attribution	`code_crossdomain/results/E24_force_control/_attrib5`
SAGIN stabilizers	`code/results/U6_stabilized_sagin`
Cross-domain stabilizers	`code_crossdomain/results/U6_stabilized_cgra`
Corrected resource factorial	`code_crossdomain/results_p1a_fixed`

The learned-schema directory contains 15 per-seed CSV traces, a configuration snapshot, 15 checkpoints, 15 progress records, aggregate JavaScript Object Notation (JSON), and provenance metadata. All 48 manifest-listed files pass SHA-256 verification. That study does not include a learned-schema/no-CGRA arm; the resource factorial supplies the missing factorial cell in its own domain.