Cooperative Fine-Tuning of Pretrained Vision-Language-Action Policies: Centralization, Communication, and Inference Recipes on TwoArmTransport

Johnathan Tucker
Department of Aeronautics and Astronautics
Stanford University

Kyler Shu
Department of Aeronautics and Astronautics
Stanford University

Purushotham Mani
Department of Aeronautics and Astronautics
Stanford University

Extended Abstract

Pretrained vision-language-action (VLA) models promise generalist robot control, but their fine-tuning recipe—LoRA adapters over a frozen backbone trained with a flow-matching or diffusion loss—have been studied primarily for single-arm and bimanual manipulation. Cooperative multi-agent manipulation introduces new challenges: a larger joint action space, long-horizon contact-rich coordination under sparse rewards, and partial observability for each agent. We ask whether, and how, a single-arm-pretrained VLA transfers to this regime. Using π_0 on Robomimic's TwoArmTransport, we run a controlled comparison of six architectures—spanning per-arm imitation, centralized and decentralized residual reinforcement learning (RL), and direct partner-action sharing—each evaluated identically at n=100 closed-loop episodes over four random seeds.

Four results define the design space. **Architecture:** decentralized residual RL reaches 38.2% success rate with *no* inter-agent communication, beating per-arm imitation (30%), the centralized residual policy (27.5%), and even the joint imitation ceiling (32%); when the backbone is single-arm-native, decentralization with implicit coordination wins. **Communication:** a 16-dimensional supervised message matches a 70-dimensional oracle that forwards the partner's complete action chunk (41.2% vs. 41.0%), so the cooperatively useful information on this task compresses to low bandwidth—and a linear probe confirms the learned channel genuinely encodes partner phase, gripper state, readiness, and pose. **Deployment:** three inference-time interventions (chunk-length tuning, mean-of-N sampling, $\pm 2\sigma$ action clipping) add 12.25 points ($25.25\% \rightarrow 37.5\%$) with no retraining—more than the 8.2-point gain from the residual-RL training itself. **Limit:** a single-trajectory overfit isolates LoRA adapter capacity—not data scale, normalization, or generalization—as the ceiling: the action head reproduces demonstrations at $r > 0.99$ yet emits rare $\sim$2.0-magnitude gripper-transition outliers that cascade in open-loop execution, identically at 1 and 200 demonstrations.

Two methodological observations arise from this study: (1) closed-loop success is *non-monotonic* in training step—it peaks near 30k steps while training loss keeps falling—so checkpoints must be selected by simulated success rather than loss. And (2) single-trajectory overfitting is a cheap, general diagnostic for locating VLA fine-tuning limits. Together these findings map cooperative VLA fine-tuning along three axes—centralization, communication, and inference—identify where each contributes measurable benefit, and yield a deployment recipe that transfers to any cooperative system built on the same backbone family.

1 Introduction

Vision-language-action (VLA) models trained on broad robot demonstration corpora have become a promising foundation for generalist robot control. Systems such as π_0 [Black et al., 2024], OpenVLA [Kim et al., 2024], and RT-2 [Brohan et al., 2023a] show that a single pretrained policy can be adapted to many manipulation tasks through lightweight fine-tuning, transferring in ways task-specialized policies cannot. The dominant recipe—LoRA adapters on a frozen backbone, trained with a flow-matching or diffusion loss on expert demonstrations—has been characterized for *single-arm* manipulation on benchmarks like LIBERO [Liu et al., 2023].

Cooperative bimanual manipulation is a different problem. Two agents must coordinate contact-rich actions over long horizons under sparse reward, often from partial observations of the joint workspace. Whether pretrained VLAs transfer here is poorly understood, and the standard recipe rests on assumptions that may not hold: that LoRA capacity tuned for one arm suffices when the action space doubles; that open-loop chunk execution is compatible with multi-agent timing; and that the cooperative communication structure emphasized by the multi-agent RL literature can live inside a frozen-backbone architecture.

We probe these assumptions empirically on the TwoArmTransport benchmark, organizing the study around three questions. **(i) Architecture:** does a centralized policy controlling both arms beat a decentralized pair, and by how much? **(ii) Communication:** for decentralized agents, what does an explicit inter-agent channel buy, and what is the minimum information sufficient for coordination? **(iii) Deployment:** which inference-time interventions raise closed-loop success without changing training? To answer them we compare six architectures that share one pretraining backbone and span the space from per-arm imitation, through centralized and decentralized residual RL, to direct partner-action sharing, all evaluated under a uniform inference recipe at $n{=}100$ closed-loop episodes and four seeds per trained method.

Our contributions are: (1) a controlled architectural comparison showing that, with a single-arm-native backbone, *decentralized* residual RL with implicit coordination matches or beats centralized policies; (2) a matched-pair communication analysis establishing that the cooperative information content of TwoArmTransport compresses to a 16-dimensional channel without loss against a 70-dimensional oracle; (3) an inference recipe that contributes more closed-loop performance than the residual-RL training itself; and (4) a single-trajectory overfit diagnostic that isolates LoRA adapter capacity as the recipe-level performance ceiling.

2 Related Work

Pretrained vision-language-action models. VLAs pair large vision-language backbones with action heads to map instructions, images, and proprioception to continuous commands. RT-1 [Brohan et al., 2023b] and RT-2 [Brohan et al., 2023a] established that scaling data and model size yields generalist manipulation policies; OpenVLA [Kim et al., 2024], Octo [Octo Model Team et al., 2024], and π_0 [Black et al., 2024] explore the design space. Two axes matter for us: the action head—discretized tokens (RT-2, OpenVLA) versus continuous diffusion/flow-matching heads (Octo, π_0) that better capture multimodal behavior—and the pretraining corpus, which is dominated by single-arm data such as Open X-Embodiment [Open X-Embodiment Collaboration, 2024]. The now-standard LoRA [Hu et al., 2022] fine-tuning recipe has been characterized on single-arm LIBERO but not on the cooperative bimanual setting, where neither the doubled action space nor inter-agent coordination was in scope. We extend that characterization directly.

Cooperative and bimanual manipulation. Beyond classical force-position and optimization-based controllers, low-cost bimanual hardware (ALOHA [Zhao et al., 2023], Mobile ALOHA [Fu et al., 2024]) has enabled large-scale teleoperation data and diffusion-based imitation [Chi et al., 2023]. The Robomimic suite [Mandlekar et al., 2021], source of our TwoArmTransport task, provides standardized cooperative environments and reproducible baselines. Most learning-based bimanual work uses *centralized* policies, which simplify coordination but double the action space and must learn cross-arm coordination from scratch. Decentralized alternatives, especially atop a pretrained generalist VLA, are far less studied; we contribute one—two per-arm policies trained cooperatively via residual RL on a single-arm-pretrained backbone.

Multi-agent RL with communication. The CTDE paradigm—MAPPO [Yu et al., 2022], QMIX [Rashid et al., 2018], COMA [Foerster et al., 2018], value decomposition [Sunehag et al., 2018]—and learned inter-agent communication channels [Foerster et al., 2016, Sukhbaatar et al., 2016] are well studied, but largely in grid-world, particle, or symbolic settings where the channel is load-bearing *by construction*. Empirical characterization of learned channels in continuous-control cooperative manipulation is sparse, and rigorous intervention probes (testing whether a trained channel is actually useful versus merely expected by the actor) are rarely reported. We supply such probes—zero, random, shuffled, oracle, and matched-compression replacements—and identify when explicit communication confers measurable benefit.

Residual RL with strong priors. Residual RL refines a strong prior with a small additive policy. We compose three modern tools on top of a frozen VLA prior: implicit Q-learning (IQL) [Kostrikov et al., 2022], which trains the value via expectile regression and avoids actor-bootstrapped Q inflation; advantage-weighted regression (AWR) [Peng et al., 2019], a stable off-policy actor update; and RLPD [Ball et al., 2023], which mixes offline demos with online rollouts at a fixed ratio for sparse-reward stability. Relative to the demonstration-only priors of prior residual-RL work, our prior (a fine-tuned VLA) is substantially stronger, and we show the inference recipe matters at least as much as the training recipe for deployment.

3 Method

3.1 Task and environment

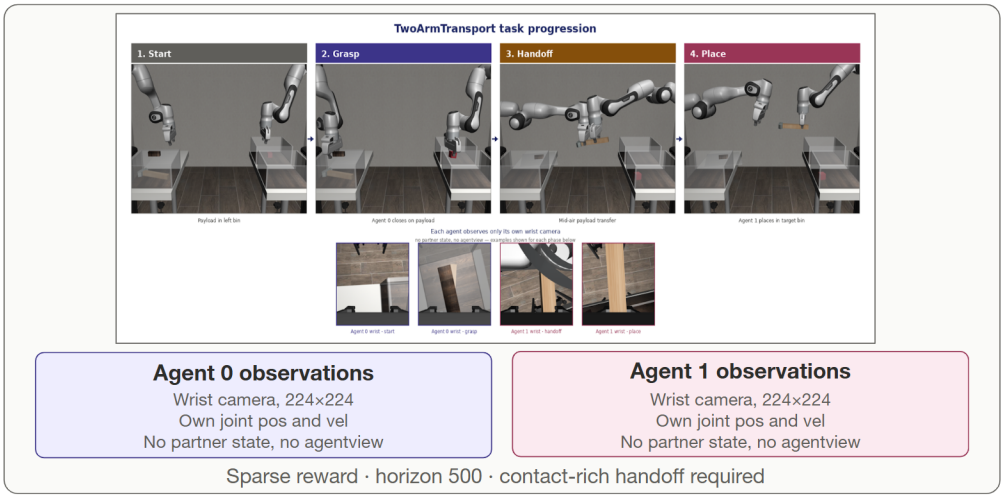

Figure 1: Robomimc's cooperative bimanual manipulation task setup, with phases of the task (top) and information available to each manipulator (bottom).

TwoArmTransport (Robomimic [Mandlekar et al., 2021], in Robosuite/MuJoCo [Zhu et al., 2020, Todorov et al., 2012]) places two 7-DoF Franka Panda arms at opposite ends of a workspace. The left arm must grasp a payload, hand it off mid-air to the right arm, which then places it in a target bin. Each arm controls a 6-DoF end-effector pose plus a binary gripper via operational-space control. Episodes run 500 steps; reward is sparse (success only when the payload reaches the bin with both grippers released), though we add a shaped potential during RL training to ease exploration. Critically, at evaluation each agent sees *only* its own wrist camera (224×224 RGB) and proprioception—no partner observation, agentview camera, or privileged state. This defines the decentralized regime for all methods. Training uses the same 200 expert demonstrations (Transport-PH split) throughout, so cross-method differences reflect architecture and procedure, not data quantity.

3.2 Phase 1: VLA imitation pretraining

We fine-tune π_0 [Black et al., 2024]—PaliGemma 3B [Beyer et al., 2024] with a flow-matching action head—in two variants. The **per-arm** variant predicts 7-DoF chunks for a single arm from that arm's wrist camera, the agentview camera, its proprioception, and the prompt; queried independently per arm, it turns the 200 cooperative demos into 400 single-arm trajectories and anchors our decentralized experiments. The **joint** variant predicts 14-DoF chunks for both arms from the full bimanual observation, serving as the centralized-imitation baseline. Both use the openpi π_0-LIBERO LoRA recipe (rank 16 on PaliGemma, rank 32 on the action expert) with default optimizer and augmentation, trained 40k steps at batch size 64.

A key methodological finding emerged here: *closed-loop success is non-monotonic in training step*. The joint VLA scored $2\%, 17\%, 32\%, 23\%$ at 10k/20k/30k/40k steps—peaking at 30k and then degrading—while training loss fell throughout. We therefore select the 30k checkpoint for both variants by simulated success, not loss.

3.3 Phase 2: cooperative residual RL

Each frozen VLA is wrapped with a learned residual policy that emits small additive corrections to the predicted chunk, trained by off-policy RL against the sparse reward with the VLA as prior. Agents share architecture but not weights. At each chunk boundary, agent i runs (a) *frozen π_0 inference* producing a length-C chunk (C=10 in training) and prefix embeddings; (b) an *RL token encoder*, a light Transformer mapping prefix embeddings and proprioception to a 2048-D latent z_{rl}; and (c) a *communication head*, an MLP producing a 16-D stochastic Gaussian message m_i transmitted to the partner. The residual actor consumes z_{rl}, proprioception, the VLA reference, and the partner message m_j (via FiLM modulation [Perez et al., 2018]), and outputs a chunk-shaped delta. The executed action is

$$a_i = \text{clip}(\text{ref}_i + s \cdot \tanh(\Delta a_i),\ -1,\ +1)\,, \tag{1}$$

where s is a learned residual scaling factor whose effective magnitude is bounded by a maximum residual radius of 0.10. It is initialized at approximately half of this maximum, yielding an initial effective radius of ≈ 0.05 and keeping the day-zero policy close to the pretrained VLA.

Objective. We use an IQL value/critic with an AWR actor, which we found avoids the Q-inflation of SAC/TD3-style targets [Haarnoja et al., 2018, Fujimoto et al., 2018] when residual actions stray into extrapolated regions. The value is trained by expectile regression against an actor-free target; the twin critic against a SmoothL1 TD target using $V(s')$; and the actor by advantage-weighted regression:

$$\mathcal{L}_V = \mathbb{E}[\text{expectile}_\tau(Q_{\text{target}}(s,a) - V(s))]\,, \quad \tau = 0.6, \tag{2}$$

$$\mathcal{L}_Q = \text{SmoothL1}(Q(s,a) - [r + \gamma V(s')])\,, \tag{3}$$

$$\mathcal{L}_\pi = \mathbb{E}\big[\min(\exp(\beta A), 100) \cdot \|\pi(s) - a\|^2\big]\,, \quad A = Q_{\text{target}} - V,\ \beta = 3.0. \tag{4}$$

Targets use Polyak averaging (τ=0.005); gradients are clipped at 1.0 (critic/value) and 5.0 (actor); LayerNorm on the Q/V trunks bounds Q magnitude and prevents divergence. Following RLPD [Ball et al., 2023], each batch of 256 is $50/50$ demo/online (demo buffer ≈10k transitions; FIFO online buffer 10k); this mix was essential—pure-online drifts, pure-demo ignores feedback. We train 3,000 chunks per seed (≈2 GPU-hours), four seeds per method.

3.4 The six architectures

- **B1 – Per-arm VLA:** frozen per-arm π_0 run independently per arm, open-loop, no residual or communication. The floor.
- **B2 – Joint VLA:** frozen joint π_0 run centrally with full bimanual observation, no RL. The centralized-imitation baseline.
- **V3 – Centralized residual RL:** one 14-DoF residual actor over the joint reference, conditioned on both agents' state. Centralized by construction, no message. (± 0.05 and ± 0.20 residual radii were indistinguishable, so scale is not the binding constraint.)

- **V4 – Per-arm residual RL, no message:** two independent 7-DoF residual actors; the communication path is replaced by identity. Isolates cooperative residual RL with implicit coordination only.
- **V5 – Per-arm residual RL, supervised 16-D message:** V4 plus a learned 16-D channel, supervised through a decoder to the sender's phase (5-way), gripper (binary), readiness (binary), and 6-DoF pose (λ_{sup}=0.1; a low-weight VIB term $\mathrm{KL}(m_i \| \mathcal{N}(0, I))$ [Alemi et al., 2017] at 10^{-4} prevents collapse).
- **V6 – Per-arm residual RL, oracle 70-D action passing:** V4 plus the partner's raw per-arm reference chunk (7 dims $\times$ 10 steps) fed directly via FiLM, with no compression. The upper bound on direct partner-action information.

V5 and V6 are a matched pair isolating compression: both expose partner behavior, through a learned 16-D bottleneck versus 70-D raw access.

3.5 Inference-time recipe

We identify three deployment-time interventions that improve closed-loop success without modifying training.

Chunk-length tuning. Although π_0 is trained with action chunks of length 50, shorter chunks permit more frequent replanning at deployment. Evaluating chunk lengths $\{10, 25, 50\}$, we find that the joint VLA performs best at 25: chunk 10 under-utilizes the model's planning horizon, whereas chunk 50 effectively abandons feedback. In contrast, the per-arm VLA prefers chunk lengths near 10, likely due to improved timing during object handoff.

Mean-of-N sampling. The flow-matching action head is stochastic, exhibiting per-dimension variability of approximately $\sigma(dx) \approx 0.029$ across $N = 20$ samples. Averaging $N = 4$ independently sampled action sequences reduces prediction noise and improves the per-arm VLA by approximately 5 percentage points. For the joint VLA, however, averaging slightly degrades performance: averaging coordinated 14-dimensional trajectories tends to blur distinct action modes, whereas averaging independent 7-dimensional trajectories primarily acts as denoising.

Per-dimension clipping. Teacher-forced analysis reveals that both VLAs closely match demonstration actions ($r \approx 0.98$) yet occasionally emit large-magnitude outliers (1.8–$2.1\times$ the nominal range), particularly in shoulder and gripper dimensions. These rare events can compound during open-loop execution. Clipping actions to $\pm 2\sigma$ of the demonstration distribution activates on roughly 10–17% of outlier-prone timesteps and suppresses these failure modes.

The final inference recipe—consisting of the per-arm 30k checkpoint, chunk length 10, mean-of-N sampling with $N = 4$, and $\pm 2\sigma$ clipping—is applied uniformly across all methods.

4 Experiments and Results

Setup. All methods train on the same 200 demos and share the evaluation harness. Each trained method runs four seeds; deterministic baselines run once. Every (method, seed) pair is scored on n=100 closed-loop episodes at its best checkpoint, from fixed reset seeds 1–100. We report success rate with the Wilson 95% CI and the 4-seed mean $\pm$ standard error, all under the recipe of Section 3.5.

Decentralized residual RL beats both imitation baselines and the centralized policy. V4 reaches 38.2% ($\pm 1.3\%$, per-seed range 36–42%), $+8.2$ pp over the per-arm VLA and $+6.2$ pp over the joint VLA—all without any communication, with agents coordinating implicitly through the shared payload state both wrist cameras observe. Centralized residual RL (V3, 27.5%) is *worse* than decentralized V4 and even worse than the un-tuned joint VLA (B2, 32%): a 14-DoF residual head has more parameters and a larger action space to coordinate than two 7-DoF heads that inherit per-arm inductive bias from the prior. Against the MARL expectation that centralization dominates, we observe no centralization advantage—only a penalty—at this data scale.

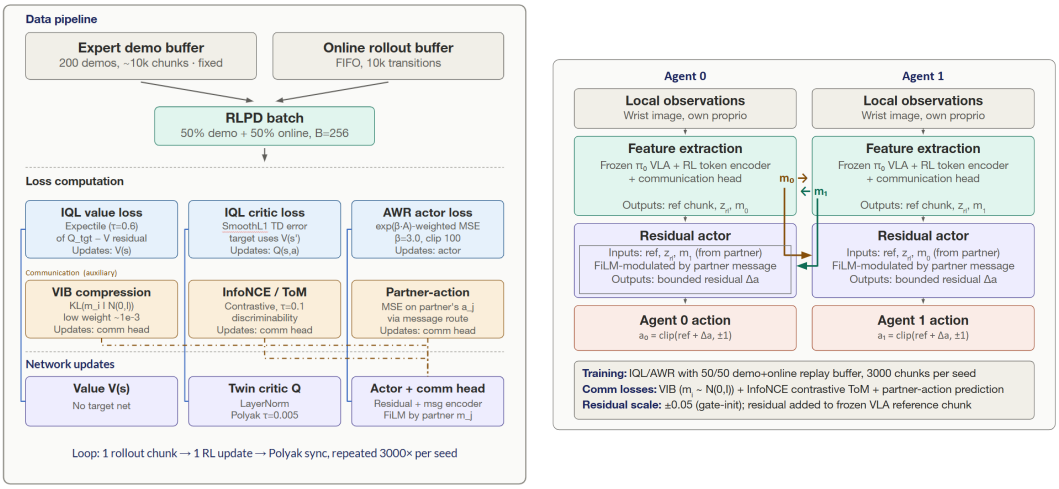

Figure 2: Training procedure (left) and per-agent inference pipeline (right).

Table 1: Closed-loop success on TwoArmTransport, n=100 per seed under the full inference recipe. Bold marks the decentralized residual-RL variants. The BC reference is the Robomimic-published number on the same 200 demos.

Method	Success	Std err / 95% CI	Seeds
B1 Per-arm VLA	30.0%	[21.9, 39.6]	—
B2 Joint VLA	32.0%	[23.7, 41.7]	—
V3 Centralized residual RL	27.5%	±0.9%	4
V4 Per-arm residual RL, no message	**38.2%**	±1.3%	4
V5 Per-arm RL, supervised 16-D msg	**41.2%**	±1.5%	4
V6 Per-arm RL, oracle 70-D passing	**41.0%**	±1.7%	4
Simple BC (task-specialized reference)	62.0%	(reported)	—

A 16-D supervised channel equals a 70-D oracle. V5 ($41.2\% \pm 1.5\%$) and V6 ($41.0\% \pm 1.7\%$) are statistically indistinguishable, so the 54 extra dimensions of raw action detail in V6 add no behavioral coordination: the cooperative information content of TwoArmTransport, at the per-step level, is bounded by what 16 supervised dimensions carry. A linear probe confirms the channel is real, recovering phase at 67% (chance 20%), gripper at 100% (50%), readiness at 71% (50%), and pose at R^2=0.27. Both channels beat no-message V4 by ≈3 pp—consistent in direction across seeds ($p < 0.1$ paired) though with overlapping per-seed CIs—reflecting a modest residual value of partner state for handoff coordination beyond the implicit shared-payload signal.

Inference recipe delivers more than residual-RL training. Ablating the recipe on V4 (Table 2), mean-of-N sampling is the single largest contributor; the cumulative recipe lifts success $25.25\% \rightarrow 37.5\%$ (+12.25 pp). This exceeds the 8.2 pp that residual-RL training itself contributes at matched inference. Because the recipe acts only on the per-arm π_0 reference, we expect it to transfer to other cooperative setups on the same backbone family—and practitioners should tune inference before scaling training machinery.

The ceiling is LoRA capacity, not data. To explain the gap to the BC reference (62%), we overfit the joint VLA to a *single* demonstration for 5k steps. Teacher-forced, it reproduced that trajectory at r=0.993 with mean absolute error 0.017—but a maximum absolute error of 2.03 in one dimension (the a_1 gripper), and closed-loop execution from the exact initial condition succeeded only 1/10 times. The same dimension's error in the 200-demo fine-tune is 2.06, essentially identical. Because the catastrophic outlier signature is invariant to data scale, the failure is not data-, generalization-, or normalization-limited: it is a capacity limit of the LoRA recipe (rank-32 action expert, default flow-matching loss) for sparse, high-magnitude, discrete-like transitions under open-loop chunk

Table 2: Inference-recipe ablation on V4 (per-arm residual RL, no message). Marginal lifts are not additive—the interventions interact.

Configuration	Success	Δ cumulative	Δ marginal
Baseline (chunk 10, N=1, no clip)	25.25%	—	—
+ mean-of-N=4 sampling	32.0%	+6.75 pp	+6.75 pp
+ per-dim $\pm 2\sigma$ clipping	35.5%	+10.25 pp	+3.5 pp
+ chunk 25 (where applicable)	37.5%	+12.25 pp	+2.0 pp

execution. Larger LoRA ranks, classifier heads on discrete dimensions, and single-step closed-loop execution would each relax this constraint—natural next steps we did not explore here.

5 Discussion

The communication results characterize a property of the *task*: TwoArmTransport admits a strong implicit-coordination strategy through the shared payload visible to both wrist cameras, leaving only a low-bandwidth, $\sim$3-pp residual for explicit messages—which a 16-D supervised channel captures fully. We would expect the balance to shift toward higher communication value under private information asymmetries or occluded partner state, a clean direction for follow-up.

The centralization results invert the usual MARL intuition, and we attribute this to the backbone: a single-arm-native prior transfers cleanly to two per-arm heads but forces a joint policy to learn bimanual coordination from scratch within 200 demos, where it also incurs correlated cross-arm outliers from the shared LoRA ceiling. The practical heuristic is that with a single-arm-native VLA, decentralized residual RL with implicit or low-bandwidth coordination is competitive with—and here superior to—centralized policies at small data scale.

Finally, two transferable lessons stand somewhat apart from the headline numbers. Training loss is a poor selector for closed-loop deployment (success peaked at 30k and degraded while loss fell), so simulated-success checkpointing is worth its compute. And single-trajectory overfitting—5k steps on one demo—is a cheap, general probe that cleanly separates a recipe ceiling from data and generalization effects, and we recommend it as a standard diagnostic for VLA fine-tuning.

6 Conclusion

We mapped cooperative VLA fine-tuning along three axes on TwoArmTransport. With a single-arm-native π_0 backbone, decentralized residual RL with implicit coordination (38.2%) outperforms both centralized imitation and centralized residual RL; a 16-D supervised message matches a 70-D oracle, bounding the task's cooperative information content; an inference recipe contributes more than the residual-RL training itself; and a single-trajectory overfit pins LoRA adapter capacity as the binding limit rather than data, normalization, or generalization. Extending the study to tasks with genuine information asymmetry, to other backbone families, and to higher-capacity adapters would directly test the scope conditions of these findings.

Team Contributions

- Multi-agent message channel — John and Kyler
- Multi-agent RL token — John and Kyler
- Actor–critic training — John and Purush
- Baseline and perturbed-setup evaluation — Purush
- Poster and paper — all

References

Alexander A Alemi, Ian Fischer, Joshua V Dillon, and Kevin Murphy. Deep variational information bottleneck. In *International Conference on Learning Representations (ICLR)*, 2017.

Philip J Ball, Laura Smith, Ilya Kostrikov, and Sergey Levine. Efficient online reinforcement learning with offline data. In *International Conference on Machine Learning (ICML)*, 2023.

Lucas Beyer, Andreas Steiner, André Susano Pinto, Alexander Kolesnikov, Xiao Wang, Daniel Salz, Maxim Neumann, Ibrahim Alabdulmohsin, Michael Tschannen, Emanuele Bugliarello, et al. PaliGemma: A versatile 3b vlm for transfer. *arXiv preprint arXiv:2407.07726*, 2024.

Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, et al. π_0: A vision-language-action flow model for general robot control. *arXiv preprint arXiv:2410.24164*, 2024.

Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Xi Chen, Krzysztof Choromanski, Tianli Ding, Danny Driess, Avinava Dubey, Chelsea Finn, et al. RT-2: Vision-language-action models transfer web knowledge to robotic control. In *Conference on Robot Learning (CoRL)*, 2023a.

Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Joseph Dabis, Chelsea Finn, Keerthana Gopalakrishnan, Karol Hausman, Alex Herzog, Jasmine Hsu, et al. RT-1: Robotics transformer for real-world control at scale. In *Robotics: Science and Systems (RSS)*, 2023b.

Cheng Chi, Siyuan Feng, Yilun Du, Zhenjia Xu, Eric Cousineau, Benjamin Burchfiel, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. In *Robotics: Science and Systems (RSS)*, 2023.

Jakob Foerster, Ioannis Alexandros Assael, Nando de Freitas, and Shimon Whiteson. Learning to communicate with deep multi-agent reinforcement learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2016.

Jakob Foerster, Gregory Farquhar, Triantafyllos Afouras, Nantas Nardelli, and Shimon Whiteson. Counterfactual multi-agent policy gradients. In *AAAI Conference on Artificial Intelligence*, 2018.

Zipeng Fu, Tony Z Zhao, and Chelsea Finn. Mobile ALOHA: Learning bimanual mobile manipulation with low-cost whole-body teleoperation. In *Conference on Robot Learning (CoRL)*, 2024.

Scott Fujimoto, Herke van Hoof, and David Meger. Addressing function approximation error in actor-critic methods. In *International Conference on Machine Learning (ICML)*, 2018.

Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *International Conference on Machine Learning (ICML)*, 2018.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations (ICLR)*, 2022.

Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Foster, Grace Lam, Pannag Sanketi, et al. OpenVLA: An open-source vision-language-action model. *arXiv preprint arXiv:2406.09246*, 2024.

Ilya Kostrikov, Ashvin Nair, and Sergey Levine. Offline reinforcement learning with implicit q-learning. In *International Conference on Learning Representations (ICLR)*, 2022.

Bo Liu, Yifeng Zhu, Chongkai Gao, Yihao Feng, Qiang Liu, Yuke Zhu, and Peter Stone. LIBERO: Benchmarking knowledge transfer for lifelong robot learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Ajay Mandlekar, Danfei Xu, Josiah Wong, Soroush Nasiriany, Chen Wang, Rohun Kulkarni, Li Fei-Fei, Silvio Savarese, Yuke Zhu, and Roberto Martín-Martín. What matters in learning from offline human demonstrations for robot manipulation. In *Conference on Robot Learning (CoRL)*, 2021.

Octo Model Team, Dibya Ghosh, Homer Walke, Karl Pertsch, Kevin Black, Oier Mees, Sudeep Dasari, Joey Hejna, Tobias Kreiman, Charles Xu, et al. Octo: An open-source generalist robot policy. In *Robotics: Science and Systems (RSS)*, 2024.

Open X-Embodiment Collaboration. Open X-Embodiment: Robotic learning datasets and RT-X models. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2024.

Xue Bin Peng, Aviral Kumar, Grace Zhang, and Sergey Levine. Advantage-weighted regression: Simple and scalable off-policy reinforcement learning. *arXiv preprint arXiv:1910.00177*, 2019.

Ethan Perez, Florian Strub, Harm de Vries, Vincent Dumoulin, and Aaron Courville. FiLM: Visual reasoning with a general conditioning layer. In *AAAI Conference on Artificial Intelligence*, 2018.

Tabish Rashid, Mikayel Samvelyan, Christian Schroeder de Witt, Gregory Farquhar, Jakob Foerster, and Shimon Whiteson. QMIX: Monotonic value function factorisation for deep multi-agent reinforcement learning. In *International Conference on Machine Learning (ICML)*, 2018.

Sainbayar Sukhbaatar, Arthur Szlam, and Rob Fergus. Learning multiagent communication with backpropagation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2016.

Peter Sunehag, Guy Lever, Audrunas Gruslys, Wojciech Marian Czarnecki, Vinicius Zambaldi, Max Jaderberg, Marc Lanctot, Nicolas Sonnerat, Joel Z Leibo, Karl Tuyls, and Thore Graepel. Value-decomposition networks for cooperative multi-agent learning based on team reward. In *International Conference on Autonomous Agents and MultiAgent Systems (AAMAS)*, 2018.

Emanuel Todorov, Tom Erez, and Yuval Tassa. MuJoCo: A physics engine for model-based control. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2012.

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

Tony Z Zhao, Vikash Kumar, Sergey Levine, and Chelsea Finn. Learning fine-grained bimanual manipulation with low-cost hardware. In *Robotics: Science and Systems (RSS)*, 2023.

Yuke Zhu, Josiah Wong, Ajay Mandlekar, Roberto Martín-Martín, Abhishek Joshi, Soroush Nasiriany, and Yifeng Zhu. robosuite: A modular simulation framework and benchmark for robot learning. In *arXiv preprint arXiv:2009.12293*, 2020.