ReVer: Reasoning-Guided Verification for Embodied Agents

Akila Ayanthi[1,2], Darshana Priyasad[2], Tharindu Fernando[2], Sridha Sridharan[2], Clinton Fookes[2] and Peyman Moghadam[1,2]

***Abstract*— Recent Vision–Language–Action (VLA) models for embodied manipulation typically evaluate action quality only through execution, relying on environment feedback or reinforcement signals for refinement. This trial-and-error paradigm introduces substantial computational and operational overhead, particularly in long-horizon tasks where early errors propagate. We propose Reasoning-Guided Verification (ReVer), a framework that leverages intermediate Chain-of-Thought (CoT) reasoning to assess action reliability prior to execution. Instead of committing to a single reasoning trajectory, ReVer samples diverse reasoning–action candidates and introduces a learned verifier that evaluates their validity. The verifier is trained on a curated dataset of both successful and failed CoT trajectories, enabling it to detect flawed reasoning patterns and anticipate downstream failures. By selecting actions conditioned on verified reasoning, ReVer reduces reliance on costly environment interaction. Experiments on the SIMPLER benchmark show that ReVer improves task success by 13.37% over OpenVLA and 11.47% over ECoT, demonstrating enhanced robustness and efficiency in embodied decision-making.**

I. INTRODUCTION

Developing robotic systems that can perceive complex environments, interpret natural-language instructions, and execute long-horizon tasks remains a fundamental challenge in embodied AI, requiring tight integration between perception, reasoning, and control. Vision–Language–Action (VLA) models [1], [2], [3], [4], [5] address this by fine-tuning Vision–Language Models (VLMs) [6], [7] on large-scale robotic datasets [8], [9], enabling policies that map multimodal observations, such as visual inputs and language instructions, directly to robot actions.

To improve long-horizon decision making, recent work augments these policies with Chain-of-Thought (CoT) reasoning [10], [11], enabling explicit intermediate reasoning for task decomposition and planning in embodied agents [12], [13], [14]. However, most approaches rely on a single reasoning trajectory, allowing reasoning errors to propagate directly to executed actions and leading to cascading failures. While generalist policies such as RT-1 [15], RT-2 [1], Octo [16], and OpenVLA [2] demonstrate strong generalization, they typically perform single-shot action prediction without assessing decision reliability, raising the risks of task failure and unsafe interactions.

In contrast, verification-based approaches in language reasoning [17], [18], [19], [20] improve robustness by evaluating multiple candidate solutions, while providing feedback and supervision to the generation process. While they contribute to substantial successes in domains such as algorithmic and mathematical reasoning, they remain underexplored in embodied settings due additional challenges, including the need for visual grounding, temporal and logical consistency and action level evaluation. To address this gap, we introduce ReVer, a reasoning-guided verification framework that evaluates multiple reasoning–action candidates generated by a VLA model before execution. By selecting actions conditioned on the predicted validity of reasoning, ReVer improves performance while remaining fully compatible with existing VLA architectures.

Corresponding author: akila.mudiyanselage@data61.csiro.au
[1]CSIRO Robotics, Data61, CSIRO, Brisbane, Australia
[2]SAIVT, Queensland University of Technology, Brisbane, Australia

II. METHOD

We propose ReVer, a reasoning-guided verification framework for VLA policies (Fig. 1). We consider a vision–language–action (VLA) policy trained from expert demonstrations consisting of sequences of image observations, natural language instructions, and corresponding robot actions. At each timestep, the policy receives the current observation o, and an instruction l and predicts a 7-DoF action a, utilizing the autoregressive next-token prediction ability of the underlying VLM. Unlike standard VLA policies that output only actions, we assume a reasoning-enabled backbone, π that additionally generates an explicit intermediate reasoning trace r interleaved with the action sequence a. This can be further expressed as, $\pi : (l, o_t) \rightarrow (a_t)$.

Given an expert demonstration dataset $D = \{\tau_i\}$, where each trajectory $\tau_i = \{(o_t, l_t, a_t, r_t)\}_{t=1}^{T}$ consists of step-wise vision–language–action–reasoning tuples, the policy is trained using a standard imitation learning objective that maximizes the likelihood of expert behavior as in Eq. 1 where ω denotes the parameters of the policy, $\tau_{(i,t)}$ denotes the discrete timestep t of the i^{th} trajectory.

$$\mathcal{L}(\omega, \tau_{(i,t)}) = -\mathbb{E}_{(o_t, l_t, a_t, r_t)\sim D}[\log \pi_\omega(a_t, r_t | o_t, l_t)]. \quad (1)$$

In this work, we adopt the architecture of OpenVLA, although our approach is compatible with any VLA that produces reasoning outputs, as the verifier operates independently of the policy architecture. Building on this backbone, our method, ReVer, treats embodied reasoning as a signal for both planning and decision validation. We keep the underlying VLA generator frozen and call it the reasoning generator. This explicitly supervises the reasoning generation based on the autoregressive language modeling objective,

$$p(Y \mid X) = \prod_{t=1}^{n} p(y_t \mid y_{<t}, X). \quad (2)$$

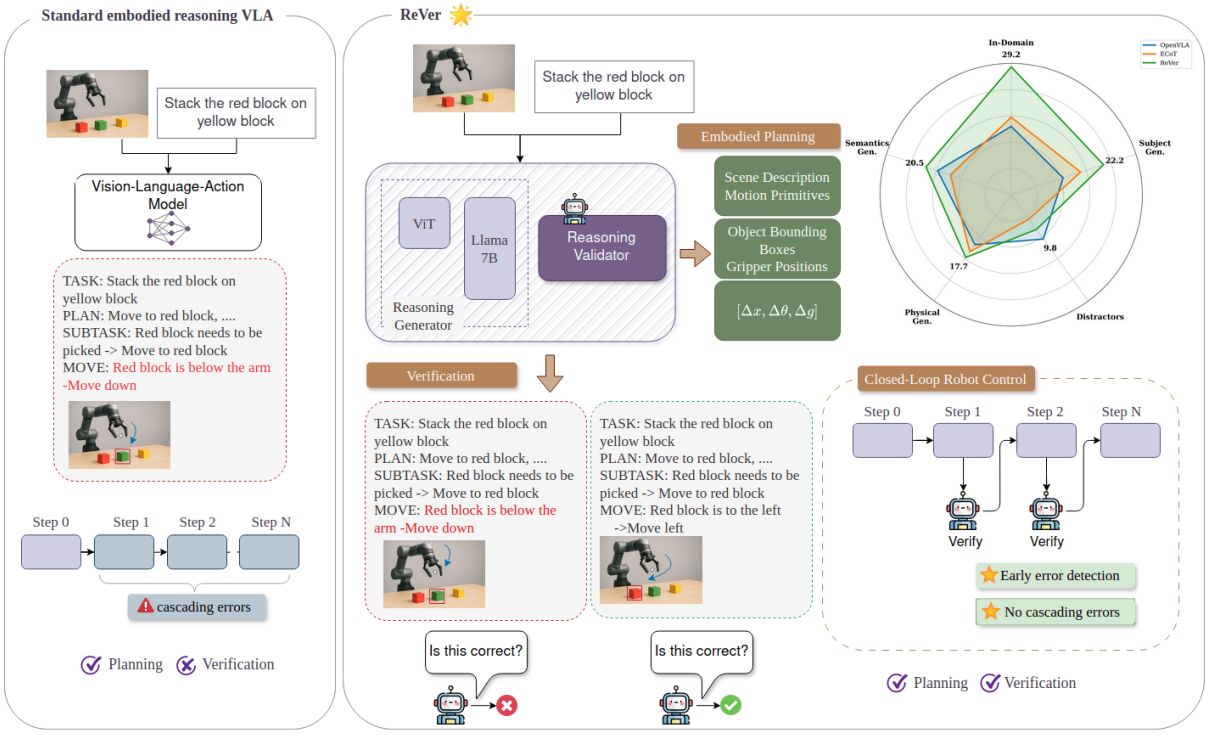

Fig. 1. Positioning of ReVer. Unlike standard VLA planning that commits to a single reason–action pair, ReVer evaluates multiple candidate reasoning trajectories using a validator and selects the most reliable action. This verification mechanism improves decision reliability and task success.

TABLE I

IN-DOMAIN TASK PERFORMANCE. VALUES SHOW GRASP AND TASK SUCCESS RATES (%). * INDICATES RESULTS FROM SIMPLER [21]. BOLD INDICATES THE BEST PERFORMANCE AND UNDERLINED VALUES INDICATE THE SECOND-BEST PERFORMANCE FOR EACH COLUMN.

Policy	**Put Spoon on Towel**		**Put Carrot on Plate**		**Stack Green on Yellow**		**Put Eggplant in Bucket**		**Avg.**
	Grasp	**Success**	**Grasp**	**Success**	**Grasp**	**Success**	**Grasp**	**Success**	
RT-1-X* [8]	16.7	0.0	20.8	4.2	8.3	0.0	0.0	0.0	1.00 ± 1.04
Octo-Base* [16]	34.7	12.5	52.8	8.3	31.9	0.0	66.7	<u>43.1</u>	15.97 ± 3.74
OpenVLA (Bridge) [2]	62.5	25.0	79.2	12.5	58.3	0.0	50.0	25.0	15.97 ± 3.74
RoboVLM [22]	37.5	20.8	33.3	25.0	8.3	8.3	0.0	0.0	13.53 ± 3.49
π_0 [3]	45.8	<u>29.1</u>	25.0	0.0	50.0	**16.6**	91.6	**62.5**	<u>27.05 ± 4.53</u>
ECoT [14]	45.8	25.0	41.7	<u>29.2</u>	58.3	0.0	50.0	16.7	17.70 ± 3.90
ReVer	54.2	**45.8**	41.7	**33.3**	75.0	<u>12.5</u>	45.8	25.0	$\mathbf{29.17 \pm 4.64}$

TABLE II

COMPARISON OF REASONING VERIFICATION STRATEGIES IN SIMPLER. VALUES ARE GRASP AND TASK SUCCESS RATES (%). BOLD INDICATES THE BEST PERFORMANCE AND UNDERLINED VALUES INDICATE THE SECOND-BEST PERFORMANCE FOR EACH COLUMN.

Verification Strategy	**Put Spoon on Towel**		**Put Carrot on Plate**		**Stack Green on Yellow**		**Put Eggplant in Bucket**		**Avg.**
	Grasp	**Success**	**Grasp**	**Success**	**Grasp**	**Success**	**Grasp**	**Success**	
Majority Vote (Self-Consistency)	45.83	<u>29.17</u>	33.33	<u>29.17</u>	66.67	<u>8.30</u>	50.00	**33.30**	<u>24.99 ± 4.42</u>
GPT-as-a-Judge	45.83	20.83	25.00	16.67	50.00	0.00	54.17	25.00	15.63 ± 3.71
LLaVA-Critic-as-a-Judge	58.33	20.83	33.33	<u>29.17</u>	62.50	0.00	50.00	16.67	16.67 ± 3.83
ReVer (Ours)	54.17	**45.83**	41.67	**33.33**	75.00	**12.50**	45.83	<u>25.00</u>	$\mathbf{29.17 \pm 4.64}$

where the visual, instruction, and reasoning tokens are concatenated and passed to the language model as an input sequence $X = (x_1, x_2, ...x_{m-1}, x_m)$ and the model autoregressively predicts an output sequence $Y = (y_1, y_2, ..., y_{n-1}, y_n)$. The model is trained to generate structured intermediate reasoning steps, including task rephrasings, high-level plans, subtask decompositions, salient visual cues, and low-level movements, prior to producing the final action commands.

At inference time, candidate solutions can be generated either deterministically via greedy decoding or stochastically via temperature sampling. Greedy decoding produces

TABLE III

PERFORMANCE COMPARISON ACROSS OUT-OF-DOMAIN (OOD) GENERALIZATION SCENARIOS. ALL VALUES ARE GRASP AND FINAL TASK SUCCESS RATES (%). MEAN ± SE IS REPORTED FOR BOTH GRASP AND SUCCESS RATE, WITH THE BEST FINAL SUCCESS IN BOLD.

Generalization	**Task**	**OpenVLA (Bridge)**		**ECoT**		**ReVer (Ours)**	
		Grasp	**Success**	**Grasp**	**Success**	**Grasp**	**Success**
Subject Generalization (unseen objects)	Put coke can on the towel	45.83	12.50	37.50	25.00	41.67	12.50
	Put pepsi can on the towel	41.67	16.67	33.33	12.50	37.50	20.83
	Put sprite can on the towel	33.33	8.33	37.50	12.50	45.83	33.33
	Average	40.28	12.50	36.11	16.67	41.67	**22.22**
Physical Generalization (unseen sizes/shapes)	Put carrot on plate (size 0.5)	12.50	4.17	12.50	12.50	20.83	16.67
	Put carrot on plate (size 1.1)	66.67	12.50	66.67	25.00	50.00	16.67
	Put carrot (wider collision box)	79.17	12.50	41.67	20.83	45.83	20.83
	Put carrot (longer collision box)	79.17	12.50	41.67	20.83	54.17	25.00
	Put spoon on towel (size 0.5)	29.17	4.17	37.50	0.00	37.50	8.33
	Put spoon on towel (size 1.1)	41.67	29.17	58.33	25.00	45.83	16.67
	Put spoon (wider collision box)	62.50	25.00	41.67	12.50	54.17	20.83
	Put spoon (longer collision box)	62.50	12.50	41.67	12.50	41.67	16.67
	Average	54.17	14.06	42.71	16.15	43.75	**17.71**
Semantics Generalization (unseen instructions)	Put the vegetable on the plate	54.17	8.30	25.00	20.83	37.50	29.16
	Move eggplant into basket	58.33	37.50	41.67	12.50	33.33	20.83
	Put green cube onto yellow	54.17	8.30	75.00	12.50	62.50	4.17
	Place spoon onto towel	50.00	16.67	25.00	12.50	44.00	28.00
	Average	54.17	17.69	41.67	14.58	44.33	**20.54**
Presence of Distractors	Put carrot on plate	37.50	12.50	29.17	8.30	34.48	13.79
	Stack green block on yellow	45.83	0.00	66.67	4.17	66.67	8.30
	Put spoon on towel	50.00	25.00	37.50	8.33	51.85	7.40
	Average	44.44	12.50	44.45	6.93	51.00	**9.83**
Overall Average		50.23 ± 2.40	14.34 ± 1.67	41.67 ± 2.37	14.34 ± 1.69	44.74 ± 2.39	$\mathbf{17.77 \pm 1.84}$

a single reasoning–action trajectory, whereas temperature sampling reshapes the output distribution and yields multiple diverse but still plausible reasoning–action candidates. Because many robotic tasks allow for several valid execution strategies, this diversity is crucial and exposes the limitations of purely greedy decoding. At the same time, temperature sampling can also produce incorrect outputs, which motivates the need for an explicit reasoning verifier mechanism to distinguish valid from invalid solutions.

The proposed reasoning verifier utilizes the interpretable internal states in the form of reasoning-action pairs to identify suitable candidate solutions for execution. Such training requires exposure to both successful and unsuccessful reasoning patterns. For this we prepare a dataset consisting of successful and failed executions of the model. During data collection, we apply temperature-controlled sampling to the pretrained generator, producing multiple diverse reasoning–action trajectories for each task instance. Given a vocabulary V of reasoning–action tokens with logits $\{l_n\}_{n=1}^{V}$, the probability of sampling token y_n under a temperature parameter $T > 0$ can be defined as in Eq. 3. The logits l_n are conditioned on the previous tokens $y_{<t}$ and input context X, ensuring consistency with the autoregressive model.

$$p(y_n|y_{<t}, X) = \frac{\exp(l_n/T)}{\sum_{n'} \exp(l_{n'}/T)} \tag{3}$$

With this sampling strategy, the model can generate multiple distinct reasoning–action pairs for the same input, each representing a strategy toward task completion.

To construct training labels for the validator, we adopt outcome-based supervision at the trajectory level. For each sampled trajectory, we execute or simulate the full sequence and assign a binary success label depending on whether the final task objective is achieved. All step-level reasoning–action pairs within a successful trajectory are treated as positive examples, whereas all steps in a failed trajectory are treated as negative. This strategy avoids expensive, ambiguous per-step annotation while still aligning supervision with task-level success, following similar outcome-based approaches used in other reasoning tasks. The resulting dataset consists of stepwise reasoning–action pairs paired with binary correctness labels that capture how reasoning quality correlates with action correctness.

The reasoning validator is implemented as a lightweight classifier head operating on the hidden states of the generator, estimating the probability that a predicted reasoning–action sequence is correct. Given an input consisting of the current observation, instruction, and a candidate solution (reasoning tokens followed by action tokens), the backbone produces a sequence of hidden states. We extract the final token embedding as a compact representation of the full reasoning–action proposal and pass it through a multi-layer perceptron with two hidden layers to obtain a scalar score.

Given a verification dataset D^V containing both positive (D^+) and negative data (D^-), $D^V = D^+ \cup D^-$, we train our

verifier as follows:

$$L(\theta, D^V) = -w^+ \cdot \mathbb{E}_{(x,y^+)\sim D^+}\left[\log r_\theta(x, y^+)\right] - w^- \cdot \mathbb{E}_{(x,y^-)\sim D^-}\left[\log\left(1 - r_\theta(x, y^-)\right)\right], \quad (4)$$

where r_θ represents the reasoning validator, $r_\theta(x, y) = sigmoid(z_{\text{cls}})$ and $z_{\text{cls}} = logit_\theta(cls\|y, x)$. y^+ are positive and y^- are negative solutions, and cls corresponds to the last nonpadding token. w^+ and w^- represent the class weights for positive and negative samples, respectively. To account for the class imbalance, we compute these weights inversely proportional to the class frequencies:

$$w^+ = \frac{|D^V|}{2\,|D^+|}, \quad w^- = \frac{|D^V|}{2\,|D^-|}$$

where $|D^V| = |D^+| + |D^-|$. This normalization ensures that positive and negative samples contribute equally to the total loss.

At inference time, our method uses the pretrained generator to propose multiple candidate reasoning–action sequences per decision step via high-temperature sampling. For each candidate, we compute the corresponding hidden representation and score it with the trained validator. The final action is chosen according to a best-of-N selection rule, selecting the candidate with the highest predicted correctness probability. This procedure combines the exploration benefits of stochastic decoding with the safety of learned verification, enabling the policy to exploit diverse strategies while screening out implausible or unsafe reasoning–action sequences before execution.

III. EXPERIMENTS AND RESULTS

We evaluate ReVer on the SIMPLER benchmark [21] under the Bridge-V2 setup across four in-domain manipulation tasks: *put spoon on towel*, *put carrot on plate*, *stack green block on yellow block*, and *put eggplant in yellow basket* as well as generalization across four dimensions. For Subject Generalization (unseen objects), Semantic Generalization (unseen instructions), and Physical Generalization (unseen object sizes/shapes), we follow the task suites in [23]. In addition, we introduce a new Distractor setting with unrelated objects present on the tabletop. These tasks require grounding visual observations and language instructions into sequential actions, providing a controlled setting for evaluating perception-driven decision making.

We compare ReVer against representative generalist policies, including RT-1-X, Octo-Base, RoboVLM, π_0, OpenVLA, and ECoT as shown in Table I. Our method samples $N{=}5$ reasoning–action candidates from a frozen generator and ranks them using a learned validator conditioned on visual observations and task instructions. As shown in Table I, ReVer achieves the highest average success rate of **29.17%**, outperforming OpenVLA (15.97%) and ECoT (17.70%) by 13.37 and 11.47 points, respectively, and surpassing π_0 (27.05%). Notably, ReVer improves success on challenging tasks such as *stack green on yellow* (12.5% vs. 0% for most baselines) and *put spoon on towel* (45.8% vs. 25.0% for OpenVLA), indicating more reliable long-horizon execution. Despite accurate object localization and high grasp success across several baselines, most methods fail to complete the tasks due to the precise spatial alignment and geometric reasoning required for task completion. However, ReVer demonstrates improved spatial reasoning and execution robustness beyond object localization due to its reasoning validator.

We further compare against alternative verification strategies, including majority voting, GPT-as-a-judge, and LLaVA-Critic-as-a-judge. Table II shows that ReVer outperforms all alternatives, improving average success from 24.99% (self-consistency) to 29.17%. These results demonstrate that transferring verifier concepts from pure language reasoning to embodied decision making requires fundamentally different paradigms and that simple consensus-based selection or language based judging alone is insufficient for embodied decision making. Overall, ReVer demonstrates the effectiveness of an embodied verifier that leverages chain-of-thought reasoning as a mechanism for action grounded verification.

Table III reports the generalization performance of OpenVLA (Bridge), ECoT, and our method across 4 generalization settings. In Subject Generalization, where the model is evaluated on unseen objects, ReVer achieves the highest average success rate of 22.22%, outperforming both OpenVLA (Bridge) 12.5% and ECoT 16.67%, indicating stronger robustness to changes in object identity. Under Physical Generalization, where variations in size and shape of objects are introduced ReVer achieves the best success rate of 17.71%, suggesting the adaptability under physical variations. For Semantic Generalization, which involves unseen instructions, ReVer continues to lead by achieving 20.54% average success rate, demonstrating better language grounding. In the Distractor setting, where irrelevant objects are introduced to the scene, OpenVLA (Bridge) achieves the highest average success rate. ReVer still outperforms ECoT, highlighting the ability to filter out distractor induced-errors. As demonstrated by the results across all generalization settings, ReVer maintains more stable performance, whereas baseline methods exhibit variance.

IV. CONCLUSION

We presented ReVer, a reasoning-guided verification framework that improves action selection in VLA policies by ranking multiple candidate reasoning–action trajectories prior to execution. By combining stochastic reasoning generation with a learned, perception-conditioned validator, ReVer enables uncertainty-aware decision making without additional environment interaction. Experiments on the SIMPLER benchmark show that ReVer achieves consistent gains over baselines, highlighting that reasoning validation is effective for enhancing robustness and reliability in embodied agents, providing a scalable alternative to execution-driven evaluation in long-horizon manipulation tasks.

REFERENCES

[1] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, X. Chen, K. Choromanski, T. Ding, D. Driess, A. Dubey, C. Finn *et al.*, "Rt-2: Vision-

language-action models transfer web knowledge to robotic control," in *Proceedings of The 7th Conference on Robot Learning*, 2023, pp. 2165–2183.

[2] M. J. Kim, K. Pertsch, S. Karamcheti, T. Xiao, A. Balakrishna, S. Nair, R. Rafailov, E. P. Foster, P. R. Sanketi *et al.*, "Openvla: An open-source vision-language-action model," in *Proceedings of The 8th Conference on Robot Learning*, 2024, pp. 2679–2713.

[3] K. Black, N. Brown, D. Driess, A. Esmail, M. Equi, C. Finn, N. Fusai, L. Groom, K. Hausman, B. Ichter *et al.*, "π0: A vision-language-action flow model for general robot control," *arXiv:2410.24164*, 2024.

[4] P. Intelligence, K. Black, N. Brown, J. Darpinian, K. Dhabalia, D. Driess, A. Esmail, M. Equi, C. Finn, N. Fusai *et al.*, "$\pi_{0.5}$: a vision-language-action model with open-world generalization," *arXiv:2504.16054*, 2025.

[5] J. Wen, Y. Zhu, J. Li, M. Zhu, Z. Tang, K. Wu, Z. Xu, N. Liu, R. Cheng, C. Shen *et al.*, "Tinyvla: Towards fast, data-efficient vision-language-action models for robotic manipulation," *IEEE Robotics and Automation Letters*, 2025.

[6] H. Liu, C. Li, Q. Wu, and Y. J. Lee, "Visual instruction tuning," in *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

[7] Z. Peng, W. Wang, L. Dong, Y. Hao, S. Huang, S. Ma, Q. Ye, and F. Wei, "Grounding multimodal large language models to the world," in *The Twelfth International Conference on Learning Representations*, 2024.

[8] Q. Vuong, S. Levine, H. R. Walke, K. Pertsch, A. Singh, R. Doshi, C. Xu *et al.*, "Open x-embodiment: Robotic learning datasets and RT-x models," in *Towards Generalist Robots: Learning Paradigms for Scalable Skill Acquisition @ CoRL2023*, 2023.

[9] A. Khazatsky, K. Pertsch, S. Nair, A. Balakrishna, S. Dasari, S. Karamcheti, S. Nasiriany, M. K. Srirama, L. Y. Chen, K. Ellis *et al.*, "DROID: A large-scale in-the-wild robot manipulation dataset," in *RSS 2024 Workshop: Data Generation for Robotics*, 2024.

[10] J. Wei, X. Wang, D. Schuurmans, M. Bosma, brian ichter, F. Xia, E. H. Chi, Q. V. Le, and D. Zhou, "Chain of thought prompting elicits reasoning in large language models," in *Thirty-sixth Conference on Neural Information Processing Systems*, A. H. Oh, A. Agarwal, D. Belgrave, and K. Cho, Eds., 2022.

[11] D. Zhou, N. Schärli, L. Hou, J. Wei, N. Scales, X. Wang, D. Schuurmans, C. Cui, O. Bousquet, Q. V. Le, and E. H. Chi, "Least-to-most prompting enables complex reasoning in large language models," in *The Eleventh International Conference on Learning Representations*, 2023.

[12] W. Huang, F. Xia, T. Xiao, H. Chan, J. Liang, P. Florence, A. Zeng, J. Tompson, I. Mordatch, Y. Chebotar, P. Sermanet, T. Jackson, N. Brown, L. Luu, S. Levine, K. Hausman, and brian ichter, "Inner monologue: Embodied reasoning through planning with language models," in *6th Annual Conference on Robot Learning*, 2022.

[13] Y. Mu, Q. Zhang, M. Hu, W. Wang, M. Ding, J. Jin, B. Wang, J. Dai, Y. Qiao, and P. Luo, "EmbodiedGPT: Vision-language pre-training via embodied chain of thought," in *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

[14] M. Zawalski, W. Chen, K. Pertsch, O. Mees, C. Finn, and S. Levine, "Robotic control via embodied chain-of-thought reasoning," in *8th Annual Conference on Robot Learning*, 2024.

[15] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, J. Dabis, C. Finn, K. Gopalakrishnan *et al.*, "Rt-1: Robotics transformer for real-world control at scale," in *Proceedings of Robotics: Science and Systems*, Daegu, Republic of Korea, July 2023.

[16] D. Ghosh, H. R. Walke, K. Pertsch, K. Black, O. Mees, S. Dasari, J. Hejna, T. Kreiman, C. Xu, J. Luo *et al.*, "Octo: An open-source generalist robot policy," in *Robotics: Science and Systems XX, Delft, The Netherlands, July 15-19, 2024*, D. Kulic, G. Venture, K. E. Bekris, and E. Coronado, Eds., 2024.

[17] H. Lightman, V. Kosaraju, Y. Burda, H. Edwards, B. Baker, T. Lee, J. Leike, J. Schulman, I. Sutskever, and K. Cobbe, "Let's verify step by step," in *The Twelfth International Conference on Learning Representations*, 2023.

[18] P. Wang, L. Li, Z. Shao, R. Xu, D. Dai, Y. Li, D. Chen, Y. Wu, and Z. Sui, "Math-shepherd: Verify and reinforce llms step-by-step without human annotations," in *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, L. Ku, A. Martins, and V. Srikumar, Eds. Association for Computational Linguistics, 2024, pp. 9426–9439.

[19] L. Zhang, A. Hosseini, H. Bansal, M. Kazemi, A. Kumar, and R. Agarwal, "Generative verifiers: Reward modeling as next-token prediction," in *The Thirteenth International Conference on Learning Representations*, 2025.

[20] L. Zheng, W.-L. Chiang, Y. Sheng, S. Zhuang, Z. Wu, Y. Zhuang, Z. Lin, Z. Li, D. Li, E. Xing, H. Zhang, J. E. Gonzalez, and I. Stoica, "Judging LLM-as-a-judge with MT-bench and chatbot arena," in *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2023.

[21] X. Li, K. Hsu, J. Gu, O. Mees, K. Pertsch, H. R. Walke, C. Fu, I. Lunawat, I. Sieh, S. Kirmani, S. Levine, J. Wu, C. Finn, H. Su, Q. Vuong, and T. Xiao, "Evaluating real-world robot manipulation policies in simulation." in *CoRL*, ser. Proceedings of Machine Learning Research, P. Agrawal, O. Kroemer, and W. Burgard, Eds., vol. 270. PMLR, 2024, pp. 3705–3728.

[22] X. Li, P. Li, M. Liu, D. Wang, J. Liu, B. Kang, X. Ma, T. Kong, H. Zhang, and H. Liu, "Towards generalist robot policies: What matters in building vision-language-action models," *CoRR*, vol. abs/2412.14058, 2024.

[23] Z. Zhang, K. Zheng, Z. Chen, J. Jang, Y. Li, S. Han, C. Wang, M. Ding, D. Fox, and H. Yao, "GRAPE: Generalizing robot policy via preference alignment," in *ICRA 2025 Workshop on Foundation Models and Neuro-Symbolic AI for Robotics*, 2025.