Route-Guided CBF-QP Repair for Safe Vision-Language-Action Models

Chan Park[1,3,✉], Sunwoo Hong[2,3], Kirak Kim[2,3], Yoonjeong Park[2,3],
Alexander W. Olson[3], Ali Pesaranghader[4,†]
[1]Sungkyunkwan University [2]KAIST [3]University of Toronto [4]LG Electronics, Toronto AI Lab
[✉]pchan1018@g.skku.edu [†]ali.pesaranghader@lge.com

***Abstract*—Vision-language-action (VLA) models can generate task-oriented robot actions, but they do not guarantee safe execution in obstacle-rich manipulation scenes. Although CBF-QP safety filters can locally project unsafe actions into a safe set, such correction may push the robot away from the learned task trajectory and leave the VLA policy without a reliable recovery direction. We propose Route-Guided CBF-QP Repair, a post-hoc repair module that uses the short-horizon action chunk predicted by the VLA to detect blocked nominal trajectories and generate entry-exit-rejoin guidance around obstacles. The selected route is added to the CBF-QP objective as a soft guidance term, while the CBF safety constraint remains hard. On SafeLIBERO, our method improves collision avoidance, task success, and SafeSuccess compared with AEGIS. Our project page is available at GitHub.**

I. Introduction

Vision-language-action (VLA) policies map visual observations and language instructions to robot actions [21, 9, 4, 8]. They are effective for task-oriented manipulation, but they do not guarantee safe execution [20, 7]. In obstacle-rich scenes, an action that moves the robot toward the target object can still pass through an unsafe region [7]. This safety gap becomes critical in manipulation tasks. A small collision can move an obstacle, disturb the target object, or push the robot into an unfamiliar state. In such cases, the policy may still understand the task, but the execution path becomes difficult to recover.

Prior work has addressed this issue by adding a post-hoc safety layer after the VLA policy [7]. AEGIS follows this direction by applying a control barrier function quadratic program (CBF-QP) to the VLA action [7]. The QP solver finds an action close to the original VLA output while satisfying a hard safety constraint [2, 7]. This local projection is effective for avoiding immediate collisions [7]. However, local safety projection does not provide trajectory recovery. A CBF-QP filter can push the robot away from an obstacle, but importantly it does not decide how the robot should move around the obstacle or return to the task trajectory. Once the robot is pushed off this trajectory, the VLA policy may enter an unfamiliar state region and fail to recover the intended behavior [15, 12].

✉ Corresponding Author

† Industry Advisor

This work was conducted during a graduate visiting studentship at the University of Toronto through the CARTE program (Centre for Analytics and Artificial Intelligence Engineering), in partnership with the LG Electronics Toronto AI Lab.

We propose *Route-Guided CBF-QP Repair*, a post-hoc repair module for safe VLA execution. Our method uses short-horizon action chunks predicted by the VLA policy to estimate the nominal path. If this path is blocked by an ellipsoidal unsafe region, we generate route candidates with *entry*, *exit*, and *rejoin* waypoints. The selected route is added to the QP objective as a soft guidance term, while the CBF safety constraint remains hard [2].

We evaluate the method on SafeLIBERO, where a robot must complete language-conditioned manipulation tasks while avoiding obstacles [7, 11]. Raw VLA policies often fail because they have no explicit obstacle-avoidance mechanism [20, 7]. AEGIS improves collision avoidance through local CBF-QP repair, but can still lose task progress after safety intervention [7]. Our route-guided repair improves the average collision avoidance, task success, and SafeSuccess rates over the evaluated baselines, indicating that route-level guidance helps connect local safety correction with successful task execution.

Our contributions are:

- We identify a trajectory-recovery failure mode of local CBF-QP safety projection in VLA robot execution.
- We introduce a route-guided CBF-QP repair method that detects blocked short-horizon VLA trajectories and generates entry-exit-rejoin guidance.
- We show that route guidance improves safe task execution on SafeLIBERO, when compared with raw VLA policies and AEGIS-style local safety repair.

II. Related Work

A. Vision-Language-Action Models and Safety Repair

VLA models extend vision-language and embodied multimodal models to robot action generation. Earlier language-conditioned robot systems grounded language-model reasoning in robot affordances or embodied observations [1, 6]. More recent generalist robot policies have benefited from large-scale robot datasets and cross-embodiment training [13, 16, 21, 9, 4, 8]. These models have improved language-conditioned manipulation through web-scale VLM transfer, open-source robot demonstration training, flow-based action generation, and heterogeneous co-training. In parallel, diffusion-based action generation has become an effective paradigm for visuomotor policy learning, and discrete-diffusion-based VLAs

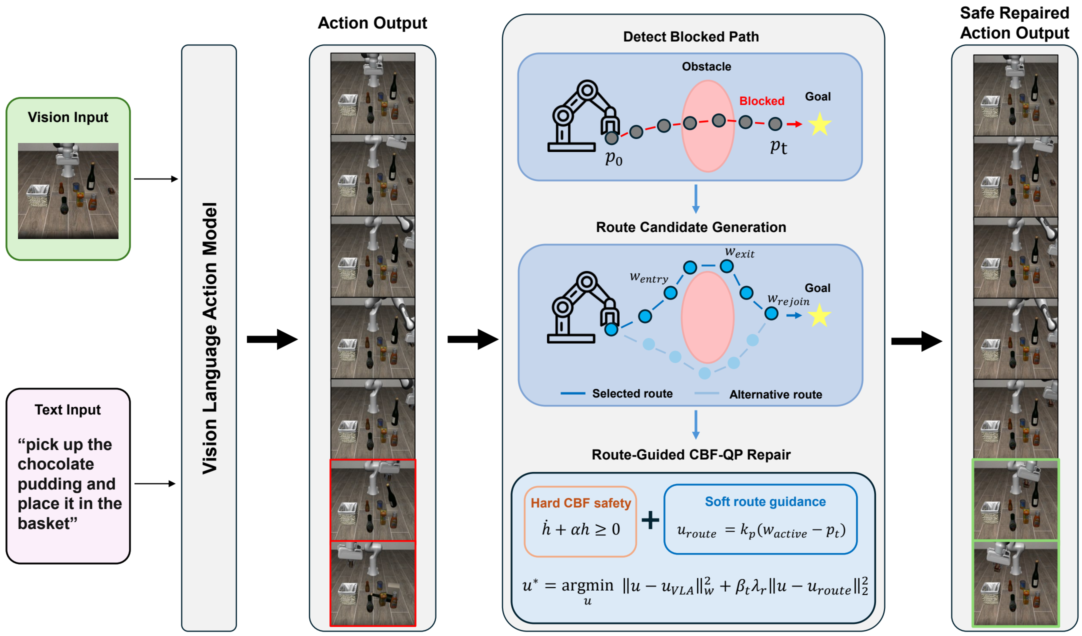

Fig. 1: **Overview of Route-Guided CBF-QP Repair.** The method uses the VLA action chunk to estimate a nominal future trajectory and detect obstacle-induced blockage. When the nominal trajectory is blocked, it generates an entry-exit-rejoin route around the obstacle and adds the selected route to the CBF-QP objective as soft guidance. The CBF constraint remains hard, so the repaired action preserves safety while following a recoverable route toward task completion.

have begun to study action decoding as iterative refinement over discretized action chunks [5, 10]. Despite these advances, VLA policies are still mainly optimized for task completion rather than explicit safety [20, 7].

Post-hoc safety repair addresses this limitation by modifying the action after it is predicted [7]. AEGIS is a representative example that adds a plug-and-play safety constraint layer after the VLA output and uses CBF-QP repair to project unsafe actions into a safe set [7]. This design preserves the pretrained VLA policy and avoids retraining. However, existing repair layers mainly focus on whether the immediate action is safe. In contrast, our work focuses on whether the repaired action remains recoverable for future task execution.

B. CBF-QP Safety Filtering and Trajectory Recovery

Control barrier functions provide a principled way to keep a controlled system inside a safe set [2, 3]. In a CBF-QP formulation, the controller optimizes for a nominal action while enforcing a barrier constraint that prevents unsafe state evolution [2, 14]. This makes CBF-QP useful as a safety filter [18]. Safe nominal actions remain nearly unchanged, while unsafe actions are minimally corrected [2, 14, 7].

This closest-safe-action principle becomes limited when the nominal controller is a VLA policy [19, 18]. Closeness to the current VLA action does not necessarily imply recoverability of the future trajectory [17]. If an obstacle blocks the nominal path, repeated local projections may keep the robot safe near the obstacle boundary, but they do not provide a route around the obstacle or back to the task trajectory. Our method preserves the hard CBF safety constraint, but adds route-level guidance as a soft objective term so that the solver favors safe actions that support obstacle bypass and trajectory rejoin.

III. Method

We propose *Route-Guided CBF-QP Repair*, a post-hoc safety repair module for VLA robot execution. Given an action predicted by a pretrained VLA policy, the module repairs the action before execution. The key idea is to keep the CBF safety constraint hard while adding route-level guidance as a soft objective term when the nominal VLA trajectory is blocked by an obstacle. This allows the repair module to select an action that is not only safe, but directed toward a recoverable route.

A. Problem Setup

At each control step t, the VLA policy receives an observation o_t and a language instruction ℓ. It predicts a short-horizon action chunk:

$$\begin{aligned} A_t &= \{a_t, a_{t+1}, \ldots, a_{t+H-1}\}, \\ A_t &\sim \pi_\theta(o_t, \ell), \end{aligned} \tag{1}$$

where H is the action horizon. Each action contains translational motion, rotation, and a gripper command. We construct the predicted nominal path by accumulating Cartesian displacements:

$$\hat{p}_{t+k} = p_t + \sum_{i=0}^{k-1} a_{t+i}^{xyz}, \quad k = 1, \ldots, H. \tag{2}$$

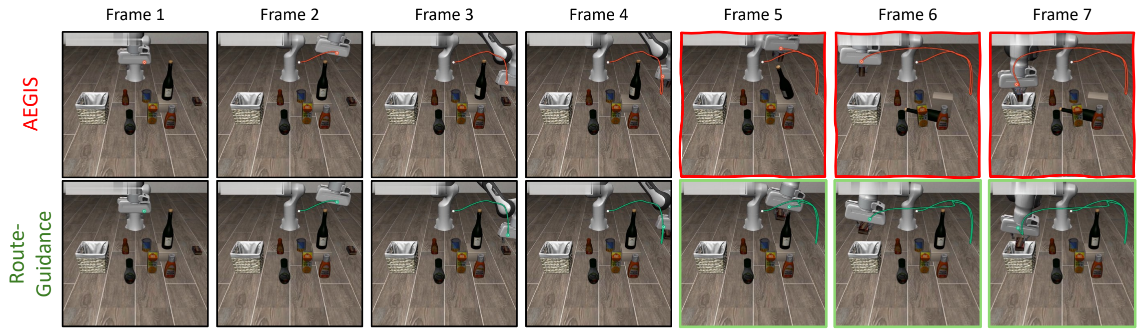

Fig. 2: **Qualitative comparison on SafeLIBERO. Task**: pick up the chocolate pudding and place it in the basket. **Top row**: AEGIS (Success, Collision). **Bottom row**: Ours (Success, Collision-free).

We model each obstacle as an inflated ellipsoidal unsafe region with center c and shape matrix Q. The ellipsoid score for a point $\hat{p}_{t+k}$ is defined as:

$$\phi(\hat{p}_{t+k}) = (\hat{p}_{t+k} - c)^\top Q^{-1} (\hat{p}_{t+k} - c). \tag{3}$$

We detect the nominal path as blocked when:

$$\min_{k \in \{1,\ldots,H\}} \phi(\hat{p}_{t+k}) < 1 + \epsilon, \tag{4}$$

where ϵ is a safety margin.

B. Route Candidate Generation

When the nominal path is blocked, we generate route candidates around the obstacle consisting of three waypoints:

$$\gamma_s = \{w^{(s)}_{\text{entry}}, w^{(s)}_{\text{exit}}, w^{(s)}_{\text{rejoin}}\}, \quad s \in \{\text{left}, \text{right}\}. \tag{5}$$

We select the optimal route s^* by maximizing the route score $J(\gamma_s)$:

$$J(\gamma_s) = \lambda_{\text{clr}} C + \lambda_{\text{prog}} P - \lambda_{\text{len}} L - \lambda_{\text{dev}} D, \tag{6}$$

where (C) rewards obstacle clearance, (P) rewards progress along the nominal VLA trajectory, (L) penalizes long detours, and (D) penalizes deviation from the original VLA path. We use fixed weights λ_{clr}=1.0, λ_{prog}=1.0, λ_{len}=0.3, and λ_{dev}=0.5 in all experiments. The optimal route is selected as:

$$s^* = \arg\max_s J(\gamma_s). \tag{7}$$

C. Route-Guided CBF-QP Repair

We repair the VLA action by solving a constrained Quadratic Program (QP). Let u_{VLA} denote the nominal action predicted by the VLA policy, and let u be the optimized action. When route guidance is active, we define a target translational action directed toward the active waypoint w_{active} as:

$$u_{\text{route}} = k_p(w_{\text{active}} - p_t). \tag{8}$$

The repaired action u^* is obtained by solving the following optimization problem:

$$\begin{aligned} u^* = \arg\min_u \quad & \|u - u_{\text{VLA}}\|^2_W + \beta_t \lambda_r \|u^{xyz} - u_{\text{route}}\|^2 \\ \text{s.t.} \quad & \dot{h}(x,u) + \alpha h(x) \geq 0, \end{aligned} \tag{9}$$

where $\|\cdot\|^2_W$ denotes the weighted squared norm. The first term ensures that the repaired action remains faithful to the original VLA intent, while the second term biases the translational component (u^{xyz}) toward the selected route waypoint.

The binary indicator $\beta_t \in \{0, 1\}$ dynamically enables route guidance; it is set to 1 only when the nominal path is blocked and a route waypoint is actively assigned. When $\beta_t = 0$, the objective reverts to the standard CBF-QP formulation. Conversely, when $\beta_t = 1$, the solver identifies a safe action that simultaneously maintains progress along the recoverable route. Throughout this process, the CBF constraint acts as a hard safety requirement, ensuring that route guidance never compromises system safety.

D. Sequential Rejoin Execution

The active waypoint w_{active} is selected from $\{w^{(s^*)}_{\text{entry}}, w^{(s^*)}_{\text{exit}}, w^{(s^*)}_{\text{rejoin}}\}$. We transition to the next phase when:

$$\|p_t - w_{\text{active}}\| < \delta. \tag{10}$$

Upon reaching the rejoin waypoint, route guidance is deactivated, and the path reverts to standard VLA execution.

IV. Experiments

Both AEGIS and our method use the same $\pi_{0.5}$ VLA backbone and weights [8]. AEGIS applies a post-hoc CBF-QP safety layer to the VLA action and projects unsafe actions into the safe set. Since the VLA backbone is fixed, this comparison isolates the effect of adding route guidance for obstacle bypass and trajectory rejoin.

A. Benchmark and Metrics

We evaluate on the SafeLIBERO benchmark [7]. SafeLIBERO extends LIBERO-style language-conditioned manipulation tasks with obstacle constraints. A policy must therefore complete the task while avoiding collisions. We report results on four suites: Spatial, Goal, Object, and Long. We evaluate 10 episodes per task and report the average results over each suite. We use four metrics. Collision Avoidance Rate (CAR) measures the percentage of episodes without collision. Task Success Rate (TSR) measures whether the manipulation task is completed. SafeSuccess measures whether the robot satisfies

TABLE I: **Quantitative results on the SafeLIBERO benchmark.**

Method	Metric	SafeLIBERO Suite				Average
		Spatial	Goal	Object	Long	
AEGIS	CAR ↑ (%)	63.75	**83.75**	56.25	**55.00**	64.69
	TSR ↑ (%)	60.00	**85.00**	67.50	38.75	62.81
	SafeSuccess ↑ (%)	52.50	**73.75**	41.25	21.25	47.19
	ETS ↓	**85.61**	**104.33**	67.68	**181.85**	**109.87**
Route-Guided CBF-QP (Ours)	CAR ↑ (%)	**66.25**	81.25	**76.25**	52.50	**69.06**
	TSR ↑ (%)	**65.00**	80.00	**72.50**	**42.50**	**65.00**
	SafeSuccess ↑ (%)	**53.75**	67.50	**55.00**	**22.50**	**49.69**
	ETS ↓	91.20	129.50	**65.10**	188.00	118.45

Notes: **CAR**: *Collision Avoidance Rate*; **TSR**: *Task Success Rate*; **SafeSuccess**: successful task completion without collision; **ETS**: *Execution Time Steps*. Arrows indicate direction of improvement (↑ higher is better, ↓ lower is better). Best results are highlighted in **bold**.

both task success and collision avoidance. Execution Time Steps (ETS) measures the number of environment steps used during execution.

B. Quantitative Results

Table I summarizes the main quantitative results. Compared to AEGIS, our method improves average CAR from 64.69% to 69.06%, TSR from 62.81% to 65.00%, and SafeSuccess from 47.19% to 49.69%. These results indicate that route guidance helps the robot avoid obstacles while preserving task progress after safety intervention. The largest improvement appears in the Object suite, where CAR increases from 56.25% to 76.25%, TSR from 67.50% to 72.50%, and SafeSuccess from 41.25% to 55.00%. This suggests that route-guided repair is useful when obstacle avoidance must be coordinated with object-centric manipulation. ETS increases to 118.45, reflecting the additional steps used for bypass and rejoin.

C. Ablation Analysis

Table II analyzes the role of route scoring and rejoin guidance in our method. The AEGIS row corresponds to local CBF-QP repair without blocked-path detection, route candidate generation, route scoring, or rejoin guidance. Removing the rejoin component tests whether obstacle bypass alone is sufficient, while removing the route score tests whether explicit route selection is necessary. The full model achieves the best average CAR, TSR, and SafeSuccess. Without rejoin guidance, SafeSuccess decreases from 49.69% to 47.92%, indicating that bypassing the obstacle is not sufficient if the robot is not guided back toward the task trajectory. Without route scoring, SafeSuccess decreases to 48.15%, showing that selecting an appropriate route direction also matters. These results support our design choice of combining blocked-path detection, route scoring, and rejoin guidance for safe trajectory recovery.

D. Qualitative Trajectory Recovery Analysis

Figure 2 shows a representative SafeLIBERO episode. Both methods are evaluated on the same task with the same obstacle present. AEGIS completes the manipulation task, but the trajectory remains close to the obstacle and results in collision. This case illustrates the limitation of local safety projection, a strategy that modifies each action to satisfy safety constraints at the current step, where making each action locally safe does not necessarily produce a recoverable collision-free trajectory.

TABLE II: **Ablation study of route-guided CBF-QP components.**

Components				Average Performance			
Path	**Gate**	**Score**	**Rejoin**	**CAR (%)**	**TSR (%)**	**SafeSucc. (%)**	**ETS**
×	×	×	×	64.69	62.81	47.19	**109.87**
✓	✓	✓	×	67.12	63.95	47.92	112.50
✓	✓	×	✓	65.88	63.10	48.15	114.12
✓	✓	✓	✓	**69.06**	**65.00**	**49.69**	118.45

The difference becomes clear in the later frames. In frames 5–7, AEGIS continues near the blocked region and collides while completing the task. In contrast, our route-guided repair moves the robot away from the blocked nominal path, follows a bypass route around the obstacle, and then returns toward the task trajectory. This qualitative result supports our main claim that safe VLA execution requires not only immediate collision avoidance, but trajectory recovery after safety intervention.

V. Conclusion

We presented *Route-Guided CBF-QP Repair*, a post-hoc safety repair method for VLA robot execution. The method detects blocked short-horizon VLA trajectories, generates entry-exit-rejoin route guidance, and adds the selected route to the CBF-QP objective while keeping the CBF safety constraint hard. Experiments on SafeLIBERO show that our method improves average collision avoidance, task success, and SafeSuccess compared with AEGIS. These results suggest that safe VLA execution requires not local collision avoidance, but trajectory recovery that guides the robot back to task completion.

VI. Acknowledgments

This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (RS-2022-00143911, AI Excellence Global Innovative Leader Education Program).

References

[1] Michael Ahn, Anthony Brohan, Noah Brown, Yevgen Chebotar, Omar Cortes, Byron David, Chelsea Finn, Chuyuan Fu, Keerthana Gopalakrishnan, Karol Hausman, et al. Do as i can, not as i say: Grounding language in robotic affordances. *arXiv preprint arXiv:2204.01691*, 2022.

[2] Aaron D Ames, Xiangru Xu, Jessy W Grizzle, and Paulo Tabuada. Control barrier function based quadratic programs for safety critical systems. *IEEE Transactions on Automatic Control*, 62(8):3861–3876, 2016.

[3] Aaron D Ames, Samuel Coogan, Magnus Egerstedt, Gennaro Notomista, Koushil Sreenath, and Paulo Tabuada. Control barrier functions: Theory and applications. In *2019 18th European control conference (ECC)*, pages 3420–3431. Ieee, 2019.

[4] Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Flexner 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.

[5] Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*, 44(10-11):1684–1704, 2025.

[6] Danny Driess, Fei Xia, Mehdi SM Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, et al. Palm-e: An embodied multimodal language model. *arXiv preprint arXiv:2303.03378*, 2023.

[7] Songqiao Hu, Zeyi Liu, Shuang Liu, Jun Cen, Zihan Meng, and Xiao He. Vlsa: Vision-language-action models with plug-and-play safety constraint layer. *arXiv preprint arXiv:2512.11891*, 2025.

[8] Physical Intelligence, Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, and Danny ... [] ... Driess. $\pi_{0.5}$: A vision-language-action model with open-world generalization. *arXiv preprint arXiv:2504.16054*, 2025.

[9] 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.

[10] Zhixuan Liang, Yizhuo Li, Tianshuo Yang, Chengyue Wu, Sitong Mao, Liuao Pei, Tian Nian, Shunbo Zhou, Xiaokang Yang, Jiangmiao Pang, et al. Discrete diffusion vla: Bringing discrete diffusion to action decoding in vision-language-action policies. *arXiv preprint arXiv:2508.20072*, 2025.

[11] Bo Liu, Yifeng Zhu, Chongkai Gao, Yihao Feng, Qiang Liu, Yuke Zhu, and Peter Stone. Libero: Benchmarking knowledge transfer for lifelong robot learning. *Advances in Neural Information Processing Systems*, 36:44776–44791, 2023.

[12] Shaunak A Mehta, Yusuf Umut Ciftci, Balamurugan Ramachandran, Somil Bansal, and Dylan P Losey. Stable-bc: Controlling covariate shift with stable behavior cloning. *IEEE Robotics and Automation Letters*, 10(2): 1952–1959, 2025.

[13] Abby O'Neill, Abdul Rehman, Abhiram Maddukuri, Abhishek Gupta, Abhishek Padalkar, Abraham Lee, Acorn Pooley, Agrim Gupta, Ajay Mandlekar, Ajinkya Jain, et al. Open x-embodiment: Robotic learning datasets and rt-x models: Open x-embodiment collaboration 0. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6892–6903. IEEE, 2024.

[14] Manuel Rauscher, Melanie Kimmel, and Sandra Hirche. Constrained robot control using control barrier functions. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 279–285. IEEE, 2016.

[15] Stéphane Ross, Geoffrey Gordon, and Drew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 627–635. JMLR Workshop and Conference Proceedings, 2011.

[16] 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. *arXiv preprint arXiv:2405.12213*, 2024.

[17] Kim P Wabersich and Melanie N Zeilinger. Predictive control barrier functions: Enhanced safety mechanisms for learning-based control. *IEEE Transactions on Automatic Control*, 68(5):2638–2651, 2022.

[18] Kim P Wabersich, Andrew J Taylor, Jason J Choi, Koushil Sreenath, Claire J Tomlin, Aaron D Ames, and Melanie N Zeilinger. Data-driven safety filters: Hamilton-jacobi reachability, control barrier functions, and predictive methods for uncertain systems. *IEEE Control Systems Magazine*, 43(5):137–177, 2023.

[19] Kim Peter Wabersich and Melanie N Zeilinger. A predictive safety filter for learning-based control of constrained nonlinear dynamical systems. *Automatica*, 129:109597, 2021.

[20] Borong Zhang, Yuhao Zhang, Jiaming Ji, Yingshan Lei, Juntao Dai, Yuanpei Chen, and Yaodong Yang. Safevla: Towards safety alignment of vision-language-action model via constrained learning. *Advances in Neural Information Processing Systems*, 38:153335–153373, 2026.

[21] Brianna Zitkovich, Tianhe Yu, Sichun Xu, Peng Xu, Ted Xiao, Fei Xia, Jialin Wu, Paul Wohlhart, Stefan Welker, Ayzaan Wahid, et al. Rt-2: Vision-language-action models transfer web knowledge to robotic control. In *Conference on Robot Learning*, pages 2165–2183. PMLR, 2023.