Anti-Swing Control Strategy for Unmanned Crane Based on Embodied Intelligence

Shengyu Lu, Yikuan Yu, Qi Liu, Jiakang Huang, and Xinyi Le

***Abstract*—Overhead cranes play a critical role in manufacturing, shipping, and construction industries. To improve operational efficiency and safety, effective anti-swing control is essential for crane automation. Traditional anti-swing algorithms often struggle with the non-linearity of system and are incompatible with the existing velocity control interface. In this paper, we propose a novel anti-swing control method for overhead cranes based on embodied intelligence. We implement a conventional anti-swing control algorithm based on trajectory planning and PID controllers to generate demonstration data in simulated environment. Using the collected demonstration data, we apply imitation learning to train an embodied agent in performing anti-swing control. Action chunking with transformer (ACT) algorithm is utilized to enhance the ability of agent to model the mapping between observations and action sequences. In simulation experiments, our proposed method outperforms conventional anti-swing control algorithms in suppressing the maximum transient of payload and eliminating residual swing under similar efficiency.**

***Index Terms*—Overhead crane, anti-swing control, embodied intelligence, imitation learning, action chunking**

I. INTRODUCTION

A. Motivation

Overhead cranes are essential large-scale transportation equipment widely used in manufacturing, shipping, and construction, as shown in Fig. 1. Because the payload is not rigidly connected to the crane, swing during operation is inevitable. The pendulum-like movement of suspended loads not only increases the risk of accidents but also reduces operational efficiency and shortens the lifespan of crane [1]. As a result, anti-swing control techniques are crucial for enhancing the automation and intelligent operation of overhead cranes.

Experienced crane operators can reduce swing by reversing handle operations in acceleration and deceleration phases. However, this manual approach requires a high level of skill from operators and is often ineffective in fully eliminating minor oscillations.

Manuscript received: 31 August 2024; revised: 28 October 2024; accepted: 2 November 2024. *(Corresponding author: Xinyi Le.)*

Citation: S. Lu, Y. Yu, Q. Liu, J. Huang, and X. Le, Anti-swing control strategy for unmanned crane based on embodied intelligence, *IJICS*, 2024, 29(4), 177–183.

Shengyu Lu and Xinyi Le are with School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University, Shanghai 200240, China (e-mail: copenhagen@sjtu.edu.cn; lexinyi@sjtu.edu.cn).

Yikuan Yu, Qi Liu, and Jiakang Huang are with Zhenjue Technology (Shanghai) Co., Ltd., Shanghai 200241, China (e-mail: yuyikuan@zhenjuetech.com; liuqi@zhenjuetech.com; huangjiakang@zhenjuetech.com).

Digital Object Identifier 10.62678/IJICS202412.10140

Traditional control methods are successful in reducing oscillations by modeling the crane system and designing anti-swing control laws. However, there are several limitations. First, accounting for the flexibility of the rope and external disturbances in crane-load system models is challenging. As a result, traditional anti-swing control algorithms often struggle with cranes that have high lifting heights, such as quayside cranes. Second, these algorithms typically use the driving force of the crane as the control input. However, most existing overhead cranes are controlled by PLCs, which primarily offer a velocity control interface. This makes it difficult to implement these algorithms in existing systems. Finally, traditional anti-swing algorithms often face difficulties in balancing swing reduction with operational efficiency, as this trade-off cannot be easily managed by simply adjusting controller parameters.

With the booming of artificial intelligence (AI), machine learning methods are used in conventional control tasks [2, 3] and industrial scenario [4]. Among all paradigms for AI, embodied artificial intelligence emphasizes the importance of physical interaction between an agent and its environment [5]. Unlike pure computational models, embodied agents use their physical presence to gather sensory input, adapt to their surroundings, and refine their actions in real-time. This approach is particularly valuable for tasks that require continuous adaptation and dynamic control, such as crane operation, where the environment and payload conditions are constantly changing.

Imitation learning is one of the key machine learning methods employed in embodied AI. The method advocates for agents to learn and acquire the ability to solve specific tasks by observing and imitating expert demonstration data [6]. Imitation learning not only accelerates the learning process, but also helps agents better understand and adapt to their physical environment. Through imitation learning, agents can learn to perform complex tasks by observing expert behavior without the need for explicit programming instructions. The core idea of imitation learning is to extract knowledge from expert behavior and transform it into strategies that the agent can understand and execute. Behavior cloning (BC) is one typical approach to this process [6]. Currently, imitation learning technique is widely applied in tasks, such as autonomous navigation, robotic manipulation, and interactive learning [5, 7].

Embodied intelligence and imitation learning hold great potential in developing the anti-swing techniques for overhead cranes. By leveraging data from expert crane operators, imitation learning can train deep learning models to replicate expert decision-making processes, leading to smoother and

(a) Bridge crane

(b) Quay crane

Figure 1 Overhead crane.

more accurate control of the movement of crane. Unlike traditional anti-swing control algorithms that rely on complex mathematical models, the anti-swing algorithm based on imitation learning can handle non-linearities such as rope flexibility and external disturbances without detailed system modeling. Deep neural networks (NNs) actually model the nonlinear crane-payload system during the training process. Moreover, imitation learning based algorithms can be adapted to work with the velocity control interfaces of existing crane systems, making it easy to deploy the algorithm in real world. The flexibility of imitation learning enables a more effective balance between minimizing payload swing and maintaining high operational efficiency. By providing varied expert data, we can adjust the operating style of agent to suit different performance needs.

To solve the problems of conventional anti-swing algorithms, novel embodied anti-swing control algorithms based on imitation learning need to be proposed.

B. Contribution

In this paper, we implement a novel embodied anti-swing control algorithm for overhead cranes through imitation learning. Our main contributions are listed as follows:

(1) We formulate the anti-swing control problem considering the non-linearity and uncertainty.

(2) We propose embodied AI imitation to learn this non-linearity and uncertainty. To the best of our knowledge, it is the first time that embodied intelligence has been applied in learning the anti-swing behavior.

(3) The proposed algorithm is successfully validated through experiments in simulated environment.

C. Organization

Related works about anti-swing control, imitation learning, and relevant deep learning architectures are introduced in Section II. Then, our proposed methods are presented in Section III, followed by experiments in Section IV. Conclusions are summarized in Section V.

II. Related Work

A. Anti-Swing Control for Overhead Crane

Based on control theory, plenty of anti-swing control techniques for overhead cranes have been developed. They can be divided into two categories: open-loop methods and closed-loop methods. Input shaping [8–10] is a classic open-loop control technique. Input shaping reduces oscillations of the crane-load system by pre-programming a sequence of impulse control commands that counteract oscillation caused by each other. In contrast, the linear quadratic regulator (LQR) is a closed-loop optimal control method for crane control. This technique adjusts the control inputs according to the real-time state feedback and designs the optimal control law based on linearized crane system [11]. The closed-loop controller is also widely used in the anti-swing control of overhead cranes [12]. With the booming of artificial intelligence, neural network based control and other AI-based methods have been introduced to improve anti-swing performance. Neural networks can learn complex relationships in the crane dynamics and adapt over time to reduce swing in uncertain or changing conditions. Adaptive proportional derivative like neural network (APIDLNN) [3] proposed an NN-based controller that is capable of eliminating the payload swing under non-zero initial condition and disturbances.

B. Imitation Learning

Due to the rapid development of deep learning, imitation learning, an essential machine learning method of embodied AI, has been widely applied to tasks such as robot control and autonomous driving in recent years. Autonomous land vehicle in a neural network (ALVINN) [13] is one of the earliest and most famous applications of behavior cloning. ALVINN uses neural networks to map visual inputs to steering actions for autonomous vehicle control. It is a pioneering work that demonstrates the potential of imitation learning by imitating human driving behavior. Generative adversarial imitation learning (GAIL) [14] is built upon the framework of generative adversarial networks (GANs) [15] and applies this adversarial learning approach to imitation learning. It is highly useful in complex environments, where specifying a reward function is difficult. In project ALOHA (ALOHA = a low-cost open-source hardware system for bimanual teleoperation), action chunking with transformer (ACT) algorithm was proposed to address the compounding error problem of imitation learning [16, 17]. The novel imitation learning algorithm manages to train bimanual robots to accomplish complex tasks, such as opening a translucent condiment cup and inserting a battery, with a success rate of 80%–90%.

C. Conditional Variational Auto-Encoder (CVAE)

Conditional variational auto-encoder [18] is an extension of the standard variational auto-encoder (VAE) [19], designed to incorporate additional context into its latent variable generation process. Unlike the VAE, which learns to represent input data as a latent distribution, the CVAE conditions this representation on some auxiliary information. This makes it particularly suitable for structured prediction tasks, where the output is expected to be influenced by specific conditions.

D. Transformer Model

The transformer model is a deep learning architecture first introduced in 2017 by researchers at Google [20]. The core innovation of transformer is its self-attention mechanism, which allows the model to weigh the importance of different tokens in a sequence, regardless of their positions. Compared with earlier models like recurrent neural networks and long short-term memory networks, the transformer model achieves greater performance and efficiency in sequence-to-sequence tasks.

III. APPROACH

In this section, we first introduce the baseline anti-swing algorithm based on trajectory planning and PID controller. Following that, we provide a detailed explanation of the proposed imitation learning based algorithm, including the structure of our model as well as the training processes.

A. Trajectory Planning and PID Based Anti-Swing

To efficiently collect demonstration data, we first implement a conventional anti-swing control algorithm based on trajectory planning and PID.

The trajectory planner serves as the feedforward of the control framework. The trajectories are generated based on the simplified dynamic model of overhead cranes. The real-world overhead crane system is a multi-variable nonlinear system. It can be simplified to a trolley-load system, as shown in Fig. 2. M and m denote the mass of the trolley and the payload. F and f represent the driving force and the friction of the trolley. x, l, and θ are three system variables indicating the position of trolley, rope length, and the swing angle of the payload. Assuming that the rope length is constant and the swing angle is generally less than 0.1 rad, we can linearize the system. Based on the Lagrange equation [21], we can further obtain the relationship between the acceleration of trolley and the swing angle of payload as

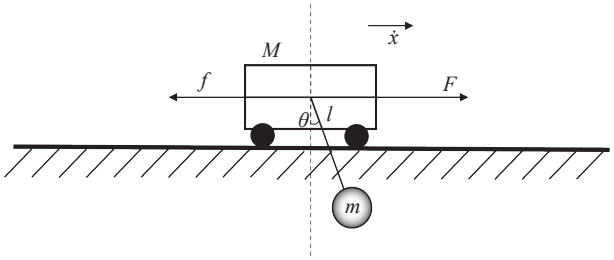


Figure 2 Simplified crane system.

$$\ddot{\theta}(t) + g\theta(t) = -\ddot{x}(t) \tag{1}$$

where g is the acceleration of gravity. If the acceleration of trolley is a step signal, the time response of swing angle is

$$\theta(t) = -\frac{a}{l}(\cos(\omega_n t) - 1) \tag{2}$$

where $\omega_n = \sqrt{g/l}$, and a is the acceleration of trolley. According to the phase plane analysis method, the swing angle can be represented as a circle with center $(-a/l, 0)$ and radius a/l in the phase plane with $\dot{\theta}/\omega_n$ as the y-axis and θ as the x-axis, as shown in Fig. 3. By ensuring that the acceleration time and deceleration time are the same as the period of the swing angle, we can avoid the unnecessary oscillations. Therefore, we can plan a three-phases motion trajectory for the trolley with acceleration/deceleration time as $2\pi\sqrt{l/g}$.

To increase the robustness and the stability of the anti-swing algorithm, we utilize two parallel PID controllers as the frame of the anti-swing controller [22]. The open-loop trajectory planning serves as the feedforward for our controller. One PID controller is used to regulate the swing angle, and the other deals with the error between the position of trolley and the desired position calculated by the trajectory planner. The overall structure for this controller is shown in Fig. 4. x_d is the desired position planned by the trajectory planner, and v_d is the control input for the crane system.

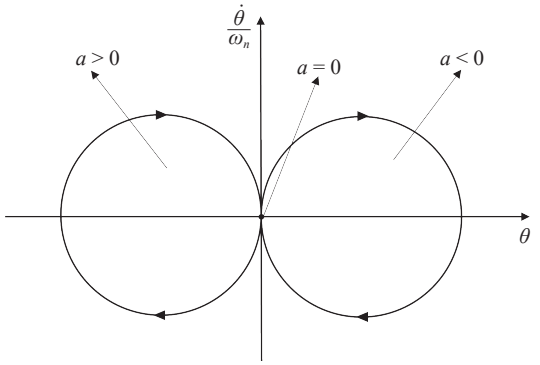


Figure 3 Representation of swing angle in phase plane. For an ideal trajectory, the time for acceleration and deceleration is equal to the cycle of the system, so that the payload will only swing for one cycle.

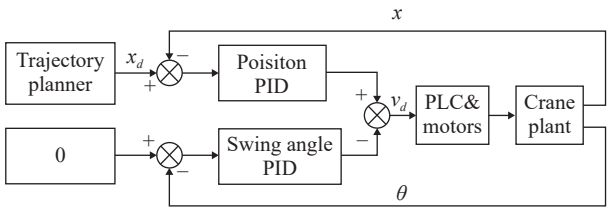


Figure 4 Overall structure for the trajectory planning and PID-based anti-swing controller. The trajectory planner outputs the reference position signal. The outputs of parallel PID controllers are sumed up to calculate the reference trolley velocity.

B. Action Chunking in Anti-Swing Task

The ACT algorithm, originally developed for fine-grained bimanual manipulation tasks, shows great potential for crane anti-swing control. In the original algorithm, action chunking [23] is used to mitigate the compounding errors typically encountered in imitation learning algorithms. Let θ represent the parameter of the deep learning model, o_t represent the observation, and a_t represent the control action. Unlike conventional imitation learning algorithms that learn $\pi_\theta(a_t \mid o_t)$, the ACT algorithm trains the network to model $\pi_\theta(a_{t+k} \mid o_t)$, where k is the chunking time step length. In bimanual manipulation tasks, chunking allows the network to better capture the non-Markovian behaviors from human demonstrations, effectively reducing the accumulation of compounding errors by folding episode length. As shown in Fig. 5, action chunking actually shortens the episode length. The imitation gap [24] between the agent policy and the expert policy is bounded by

$$\frac{|\mathcal{S}|T^2}{N} \tag{3}$$

where $\mathcal{S}$ is the state space for the task, T represents the episode length, and N denotes the number of trajectories in the demonstration datasets. With the episode length T being

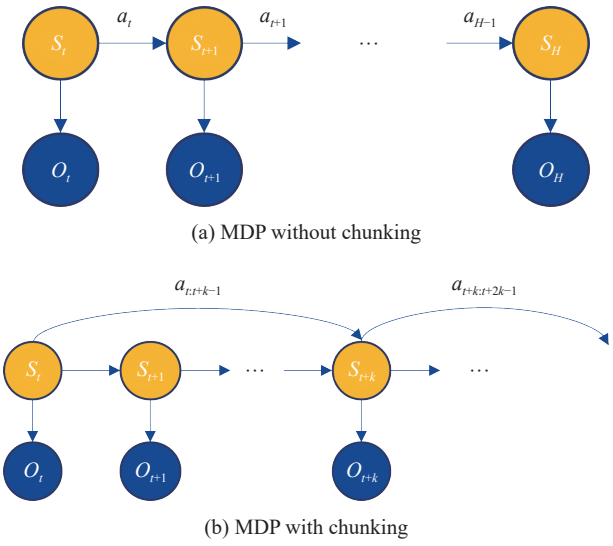


Figure 5 Action chunking in Markov decision process (MDP). Conventional MDP models only take one action for each step, resulting in a long episode. Action chunking takes a sequence of actions for each step, effectively limiting the episode length.

folded, the bound for imitation gap decreases at a quadratic rate correspondingly. This implies a better performance of algorithm with action chunking design.

Furthermore, chunking enables pre-programming control actions for the next k time steps, rather than reacting to errors after they occur. This proactive control approach aligns with the principles of manual anti-swing operation, where crane operators must anticipate and reverse the control handle in advance. By applying the ACT algorithm, the agent can effectively model the complex, nonlinear crane-payload system through deep learning training. This enables the agent to predict potential swings and preemptively counteract them by pre-programming control action sequences. As a result, the algorithm not only anticipates the onset of oscillations but also takes corrective actions in advance, minimizing the need for reactive adjustments during the crane operation.

C. Implementation of ACT in Anti-Swing Task

The ACT algorithm utilizes a conditional variational auto-encoder as the core structure of its model, because this generative approach is well-suited to capture the variability in decision-making from noisy human demonstration data. Human operators often produce different trajectories when face with the same task due to the inherent uncertainty and complexity of their decision processes. By using a generative policy network, the CVAE enables the model to output the most appropriate action sequence under a given observation, effectively imitating human flexibility while ensuring robust and efficient control.

The ACT algorithm incorporates a transformer-based architecture, which is highly effective for modeling sequential data with long-term dependencies. When combined with the CVAE structure, the encoder of transformer can efficiently capture high-dimensional features from the input action sequences, while the decoder can accurately reconstruct and generate control actions based on the conditional input. The overall structure of our implementation of ACT model is shown in Fig. 6. Specially, the ACT encoder consists of a transformer module that processes the embeddings of partial observations and action sequences to output the mean and standard deviation for the latent style variable z. For the crane anti-swing task, we use the position error of trolley $x_d - x$ and velocity $\dot{x}$ as the partial observation. Unlike the original ACT model, sample z only using the first token of the output sequence, implementation averages all output tokens. This adjustment ensures that information from the entire input sequence is retained, addressing the potential issue of losing critical data by relying solely on the first token. The decoder for ACT employs a full transformer model. The encoder processes the observations, and its outputs are passed to the transformer decoder through a cross-attention mechanism. A fixed query embedding sequence is fed to the decoder in order to acquire embeddings of the reconstructed action sequence. The embedding sequence is further projected to the final action sequence through a linear layer. We obtain feedback on the swing of payload via image data, processed into embedding vectors using a convolutional neural network. In our implementation, we utilize ResNet18 [25] to extract feature vectors from the image inputs. During the inference stage, only the decoder is utilized. The latent style variable z

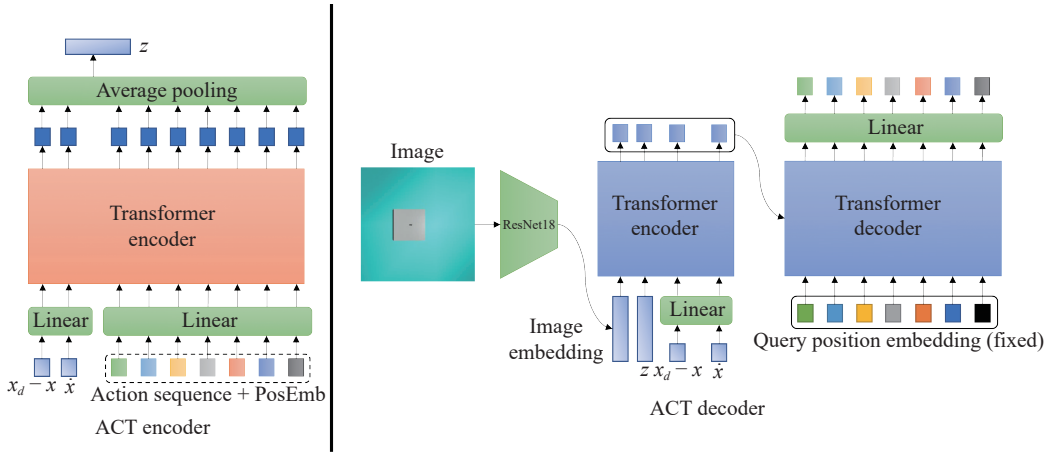


Figure 6 Structure of ACT model. The ACT encoder takes the partial observation as the input and outputs the style variable z. The ACT decoder takes the embedding of observation and the style variable as the input and outputs the predicted action sequence.

is set to its mean value, typically zero, to generate the prediction with the highest likelihood.

D. Training of ACT

The training process of ACT algorithm in the crane anti-swing task follows a supervised learning framework. The dataset consists of pairs of crane states and corresponding control action sequences. The total loss function is composed of two key components: the reconstruction loss and the Kullback-Leibler (KL) divergence.

The primary objective of the ACT algorithm is to minimize the difference between the predicted control action sequence and the expert demonstrations. The reconstruction loss achieves this by penalizing discrepancies between the predicted action sequence $\hat{a}_{t:t+k}$ and the ground truth action sequence $a_{t:t+k}$. For crane anti-swing task, whose action space is continuous, the mean square error

$$\mathcal{L}_{\text{reconst}} = \text{MSE}(\hat{a}_{t:t+k}, a_{t:t+k}) \tag{4}$$

is used to measure this difference.

The KL divergence measures how much the learned latent distribution deviates from a prior distribution, typically a standard normal distribution $\mathcal{N}(0,1)$. The KL divergence ensures that the learned distribution stays close to the prior, preventing the latent space from overfitting to specific data points. More importantly, it regularizes the latent space and enables the decoder to generate action sequence with fixed latent input. In our implementation, the KL divergence is

$$\begin{aligned}\mathcal{L}_{\text{reg}} = D_{\text{KL}}\left(q_\phi(z \mid a_{t:t+k}, \bar{o}_t) \| \mathcal{N}(0,I)\right) = \\ -\log\sigma_\phi + \frac{\sigma_\phi^2 + \mu_\phi^2}{2} - \frac{1}{2}\end{aligned} \tag{5}$$

where q_ϕ is the encoder model with parameter ϕ, σ_ϕ and μ_ϕ are the standard deviation and the mean value of z, and $\bar{o}_t$ represents the partial observation, respectively.

The total loss function is a weighted sum of the reconstruction loss and the KL divergence

$$\mathcal{L} = \mathcal{L}_{\text{reconst}} + \beta\mathcal{L}_{\text{reg}} \tag{6}$$

where β is a hyperparameter that balances the trade-off between accurate action reconstruction and regularization of the latent space. The pesudocode of training procedure is shown in Algorithm 1.

Algorithm 1 ACT training

Input: Training dataset $\mathcal{D}$, chunking length k, and weight β

1: Initialize encoder model q_ϕ

2: Initialize decoder model π_θ

3: **for** iteration $n = 1,2,3,\ldots$ **do**

4: From $\mathcal{D}$ sample $a_{t:t+k}, o_t$

5: Encoder output μ_ϕ, $\sigma_\phi \leftarrow q_\phi(a_{t:t+k}, \bar{o}_t)$

6: Sample $z \sim \mathcal{N}(\mu_\phi, \sigma_\phi^2)$

7: Decoder output $\hat{a}_{t:t+k} \leftarrow \pi_\theta(z, o_t)$

8: $\mathcal{L}_{\text{reconst}} = \text{MSE}(\hat{a}_{t:t+k}, a_{t:t+k})$

9: $\mathcal{L}_{\text{reg}} = D_{\text{KL}}\left(q_\phi(z \mid a_{t:t+k}, \bar{o}_t) \| \mathcal{N}(0,I)\right)$

10: $\mathcal{L} = \mathcal{L}_{\text{reconst}} + \beta\mathcal{L}_{\text{reg}}$

11: Update model weight ϕ and θ to minimize $\mathcal{L}$

12: **end for**

IV. Experiment

In this section, we perform experiments on our proposed crane anti-swing algorithm in simulated environment. MuJoCo [26] is used as the phyiscal engine for our simulation platform. We employ PyTorch as the deep learning framework, with which we build, to train and test our models.

A. Simulation Setting

In our experiments, we use MuJoCo [26] as the phyiscal engine to simulate the overhead crane system. The system parameters are closely aligned with those of a real-world bridge crane. The trolley has a mass of 2000.0 kg, while the payload weighs 373.3 kg. The rope length is set to 1.65 m. Its maximum velocity is 2 m/s. The high-level control algorithms interact with the crane through a velocity control interface, implemented via an inner-loop velocity PID controller. Additionally, a camera is mounted at the bottom of the trolley to provide image feedback of the payload.

B. Dataset and Training Detail

We utilize the trajectory and PID-based anti-swing control algorithm to generate demonstration data in simulation environment. For each episode, the destination point is randomly selected within a range of 10–20 m. Totally, we collect 100 episodes of demonstration data, which are divided into 130,256 observation-action sequence data pairs.

When training models, the learning rate and batch size are set to 0.0001 and 256. Adam optimizer is utilized to optimize the weights for 70 epochs on the demonstration datasets.

C. Experiment in Simulated Environment

The proposed embodied crane anti-swing algorithm is tested in simulated environment. In simulated experiments, the destination x_d is set to 16 m. For comparison, we also implement three conventional anti-swing control methods in simulations. The resulting swing angle and position curves are plotted in Fig. 7. The key performance indicators for evaluating anti-swing control are the maximum transient (MT) and residual swing (RS) [3]. Additionally, the efficiency of each method is assessed by measuring the time spent (TS) to complete the movement. The summary of these performance metrics is provided in Table 1. Judging from the results, it is clear that the imitation learning based anti-swing algorithm outperforms other methods in terms of suppressing maximum transient and eliminating residual swing. The residual swing performances for different anti-swing algorithms are also visualized in Fig. 8. Conventional anti-swing algorithms may lead to residual swing of payload, which may be harmful to the operation effiency and safety of crane. Our algorithm manages to eliminate the residual swing. Notably, the maximum swing angle achieved is even smaller than that of the algorithm used to generate the demonstration data. Thanks to the deep neural network, the agent can better deal with the non-linearity and predict the motion of payload. The action chunking design allows the agent to predict the response of crane system and take preemptive actions, reducing the maximum swing angle.

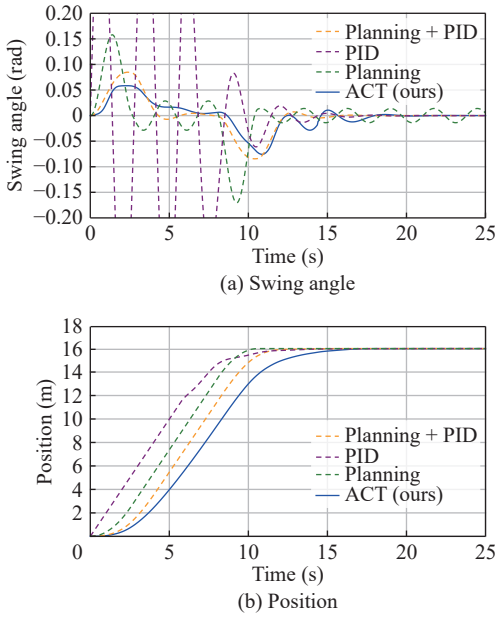


Figure 7 Swing angle and position curves for imitation learning based anti-swing algorithm. The swing angle curve plot is tailored, so that the advantages of our algorithm can be well presented.

Table 1 Performance indicator of the experiment. Bold indicates the optimal result.

Method	MT (rad)	RS (rad)	TS (s)
PID	0.4720	0.0003	17.975
Planning	0.1706	0.0153	**11.438**
Planning + PID	0.0848	0.0001	15.647
ACT (ours)	**0.0757**	**0.0000**	16.587

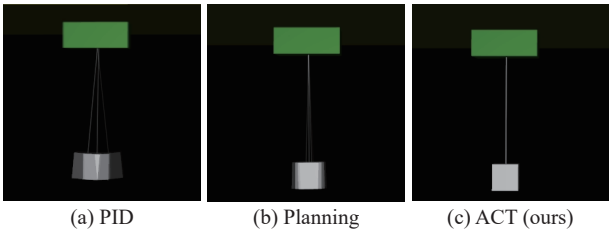


Figure 8 Residual swing for different anti-swing methods. The green boxes represent the trolleys of cranes and the grey boxes are the payloads.

V. Conclusion

In this paper, we propose a novel embodied anti-swing control algorithm for overhead cranes through the imitation learning method. We first combine closed-loop PID control and open-loop trajectory planning to effectively generate demonstration data in simulated environment. Basd on the collected data, we employ imitation learning in crane anti-swing task via ACT algorithm. Action chunking is used to tackle the accumulation of errors. The CVAE and transformer model are utilized to model the mapping from observations and action sequences. Experiments conducted in a simulated environment demonstrate the effectiveness of our proposed algorithm.

However, the imitation learning agent is trained exclusively on data generated in the simulation environment. Future research will require real-world human expert demonstration data. Additionally, anti-swing control is just one aspect of crane automation. In the future work, we aim to extend this research to achieve full automation of crane operations via imitation learning.

Acknowledgment

This work was supported by the National Natural Science Foundation of China (Nos. 62422311 and 62176152) and the Shanghai Committee of Science and Technology, China (No. 24TS1413500).

References

[1] V. Suvorov, M. Bahrami, E. Akchurin, I. Chukalkin, S. Ermakov, and S. Kan, Anti sway tuned control of gantry cranes, *SN Appl. Sci*., 2021, 3(8), 729.

[2] X. Fang, D. Pang, J. Xi, and X. Le, Distributed optimization for the multi-robot system using a neurodynamic approach, *Neurocomputing*, 2019, 367, 103–113.

[3] A. Syazwin, W. A. Faizzah, L. Ramli, Z. Mohamed, and I. M. Lazim, Swing control of 2D overhead crane using proportional integral derivative neural network, in *Proc. IEEE International Conference on Automatic Control and Intelligent Systems*, Shah Alam, Malaysia, 2024, 77–82.

[4] J. Li, D. Pang, Y. Zheng, X. Guan, and X. Le, A flexible manufacturing assembly system with deep reinforcement learning, *Control Eng. Pract*., 2022, 118, 104957.

[5] Y. Du, C. Gan, and P. Isola, Curious representation learning for embodied intelligence, in *Proc. IEEE/CVF International Conference on Computer Vision*, Montreal, Canada, 2021, 10388–10387.

[6] M. Zare, P. Kebria, A. Khosravi, and S. Nahavandi, A survey of imitation learning: Algorithms, recent developments, and challenges, *IEEE Trans. Cybern*., 2024, 54(12), 7173–7186.

[7] X. Liu, Interactive imitation learning in robotics based on simulations, arXiv preprint arXiv: 2209.03900, 2022.

[8] W. E. Singhose, L. J. Porter, and W. P. Seering, Input shaped control of a planar gantry crane with hoisting, in *Proc. American Control Conference*, Albuquerque, NM, USA, 1997, 97–100.

[9] Y. Gu, D. Niu, M. Liu, Q. Li, W. Zhu, and Y. Yang, A composite control method for industrial ship unloaders with time-varying rope length, in *Proc. 36th Chinese Control and Decision Conference*, Xi'an, China, 2024, 155–160.

[10] H. Zhang and J. Huang, Application of model estimation-based input shaping technique in tower crane control design, in *Proc. 10th International Forum on Electrical Engineering and Automation*, Nanjing, China, 2023, 825–830.

[11] B. Yang and B. Xiong, Application of LQR techniques to the anti-sway controller of overhead crane, *Adv. Mater. Res*., 2010, 139–141, 1933–1936.

[12] M. Ali Mohammed, M. Maguire, and K. Kim, Simulated annealing algorithm based tuning of LQR controller for overhead crane, in *Proc. 13th International Conference on Developments in eSystems Engineering*, Liverpool, UK, 2020, 37–42.

[13] D. A. Pomerleau, ALVINN: An autonomous land vehicle in a neural network, in *Proc. 1st International Conference on Neural Information Processing Systems*, Cambridge, MA, USA, 1988, 305–313.

[14] J. Ho and S. Ermon, Generative adversarial imitation learning, in *Proc. 30th International Conference on Neural Information Processing Systems*, Barcelona, Spain, 2016, 4572–4580.

[15] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, Generative adversarial networks, *Communications of the ACM*, 2020, 63(11), 139–144.

[16] T. Zhao, V. Kumar, S. Levine, and C. Finn, Learning fine-grained

bimanual manipulation with low-cost hardware, in *Proc. Robotics: Science and Systems*, Daegu, Republic of Korea, 2023.

[17] Z. Fu, T. Zhao, and C. Finn, Mobile ALOHA: Learning bimanual mobile manipulation with low-cost whole-body teleoperation, arXiv preprint arXiv: 2401.02117, 2024.

[18] K. Sohn, X. Yan, and H. Lee, Learning structured output representation using deep conditional generative models, in *Proc. 28th International Conference on Neural Information Processing Systems*, Montreal, Canada, 2015, 3483–3491.

[19] D. P. Kingma and M. Welling, Auto-encoding variational Bayes, in *Proc. 2nd International Conference on Learning Representations*, Banff, Canada, 2014.

[20] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, Attention is all you need, in *Proc. 31st International Conference on Neural Information Processing Systems*, Long Beach, CA, USA, 2017, 6000–6010.

[21] J. W. Auernig and H. Troger, Time optimal control of overhead cranes with hoisting of the load, *Automatica*, 1987, 23(4), 437–447.

[22] H. Li, Y. Hui, Q. Wang, H. Wang, and L. Wang, Design of anti-swing PID controller for bridge crane based on PSO and SA algorithm, *Electronics*, 2022, 11(19), 3143.

[23] L. Lai, A. Huang, and S. J. Gershman, Action chunking as conditional policy compression, arXiv preprint arXiv: 10.31234, 2022.

[24] N. Rajaraman, L. Yang, J. Jiao, and K. Ramchandran, Toward the fundamental limits of imitation learning, in *Proc. 34th International Conference on Neural Information Processing Systems*, Vancouver, Canada, 2020, 245.

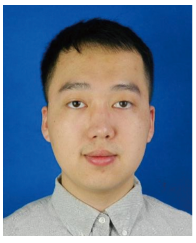
[25] K. He, X. Zhang, S. Ren, and J. Sun, Deep residual learning for image recognition, in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas, NV, USA, 2016, 770–778.

[26] E. Todorov, T. Erez, and Y. Tassa, MuJoCo: A physics engine for model-based control, in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems*, Vilamoura-Algarve, Portugal, 2012, 5026–5033.

Shengyu Lu received the BS degree from Shanghai Jiao Tong University, Shanghai, China, in 2024. He is currently pursuing the MS degree at Shanghai Jiao Tong University, Shanghai, China. His research interests include embodied intelligence and imitation learning.

Yikuan Yu received the BS and MS degrees from Shanghai Jiao Tong University, Shanghai, China, in 2017 and 2020, respectively. He is currently a chief technology officer at Zhenjue Technology (Shanghai) Co., Ltd., Shanghai, China. His research interests include computer vision, machine learning, large language models, and multimodality learning.

Qi Liu received the BS degree from Nanjing University, Nanjing, China, in 2011, and the PhD degree from Shanghai Jiao Tong University, Shanghai, China, in 2017. He is currently a chief executive officer at Zhenjue Technology (Shanghai) Co., Ltd., Shanghai, China. His research interests include machine vision, artificial intelligence, robotic control, and Internet of Things (IoT).



Jiakang Huang received the BS degree from Zhejiang University, Zhejiang, China, in 2021. He is currently a senior algorithm researcher and algorithm engineering architect at Zhenjue Technology (Shanghai) Co., Ltd., Shanghai, China. His research interests include machine vision and intelligent control.

Xinyi Le received the BS degree from Tsinghua University, Beijing, China, in 2012, and the PhD degree from The Chinese University of Hong Kong, Hong Kong, China, in 2016. She is currently an associate professor and doctoral supervisor at School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University, China. She was selected for Forbes 30 Under 30 and was awarded the Pujiang Talent of Shanghai and the Rising Star of Youth Science and Technology in Shanghai. She has led several national-level projects, including the Outstanding Youth Fund of the National Natural Science Foundation and the Youth Scientist Project of the National Key R&D Program. Her research interests include embodied intelligence in industrial robots and industrial perception.