Motion Planning and Simulation of a Mobile Manipulator Robot for Curling

Guangzhou Xiao
School of Astronautics
Harbin Institute of Technology
Harbin, China
guangzhouxiao@hit.edu.cn

Tong Wu
School of Astronautics
Harbin Institute of Technology
Harbin, China
wu_tong@hit.edu.cn

Ruixian Zhang
School of Astronautics
Harbin Institute of Technology
Harbin, China
ruixianzhang@stu.hit.edu.cn

Yunpeng Li
School of Astronautics
Harbin Institute of Technology
Harbin, China
yunpengli@hit.edu.cn

2023 8th International Conference on Robotics and Automation Engineering (ICRAE) | 979-8-3503-2765-6/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICRAE59816.2023.10458658

***Abstract*—This paper is concerned with the problem of building a physics-based digital curling simulator to train the AI-driven battle generator for mobile manipulator robots. Aiming at overcoming the deficiencies of existing simulators in complex dynamic interaction between the robot and the ice sheet, a physics-based digital simulator is built to train the AI-driven battle generator. Moreover, according to the actual configuration and parameters, the robot dynamics simulation scene is built in the V-REP simulation engine to achieve high-fidelity simulation of physical processes such as friction, gravity, and collision in curling confrontation, thereby narrowing the gap between digital simulators and the real world.**

Keywords—mobile manipulator robot, curling, battle generator, digital simulator, simulation engine

I. INTRODUCTION

Mobile manipulator robot is composed of a ground mobile platform, a high-DOF manipulator, and a decision-making hardware system, which has both ground mobile ability, flexible operation ability, and intelligent perception ability. Its market demand has seen explosive growth in the past few years, e.g., boxes pushing [1][2], casualty rescuing [3][4], environment inspecting [5][6], components assembling [7][8], materials transporting [9][10], logistics service [11], etc.

Curling has been described as a combination of bowling and chess, requiring a high level of strategic thinking and throwing performance [12]. As shown in Fig. 1, using mobile manipulator robot systems for curling in real-world environments is interesting and attractive, which stimulates greater participation of the general public in winter sports [13]. It is also a window and stage to demonstrate scientific innovations and technological capabilities, because it requires precise throwing and strategic planning to win.

The curling robot senses the posture of curling stones through a binocular camera, and transmits the information to the AI-driven battle generator. The battle generator automatically analyzes the battle scenario based on its database and determines the next action [14][15]. Finally, the robot plans and tracks the motion trajectory, to throw the curling stone at a desired velocity toward a desired direction [16][17]. In the whole process, the battle generator is the most challenging and the key to winning the curling competition [18].

Adaptive deep reinforcement learning battle generator enables robots with human-like performance for curling games through a large number of confrontation test training in real-world conditions [19][20]. However, indoor curling ice sheets are few and expensive, so it is difficult to carry out experimental tests such as RL model training for a long time and high frequency. Furthermore, the ice surface has the characteristics of low temperature and light reflection, which makes it difficult to carry out visual perception, data acquisition and algorithm deployment in the actual field. Thus, it is necessary to design a physics-based digital simulator to train the battle generator or verify the feasibility of the decision algorithm [21][22].

Digital simulators can reduce the space investment and time cost as well as the difficulty of collecting and labeling data in the real world, thereby greatly improving the efficiency of algorithm development. In addition, rendering realistic images of curling competitions can enable the participating players to learn curling-throwing strategies at any time. The existing digital simulators provide functions such as algorithm training, AI battle, and visual playback of curling trajectory [23][24]. However, they did not consider the complex dynamic interaction between the robot and the ice sheet, which has an important influence on the motion state of curling, resulting in the decision-making results of the trained battle generator cannot accurately reflect the final motion state of curling.

Fig. 1. Mobile manipulator robot used in this paper for curling. KINOVA lightweight 6-DOF bionic manipulator is mainly used to grab the handle of curling. SCOUT-mini four-wheel differential mobile is used to provide thrust to throw curling stone to the predetermined position.

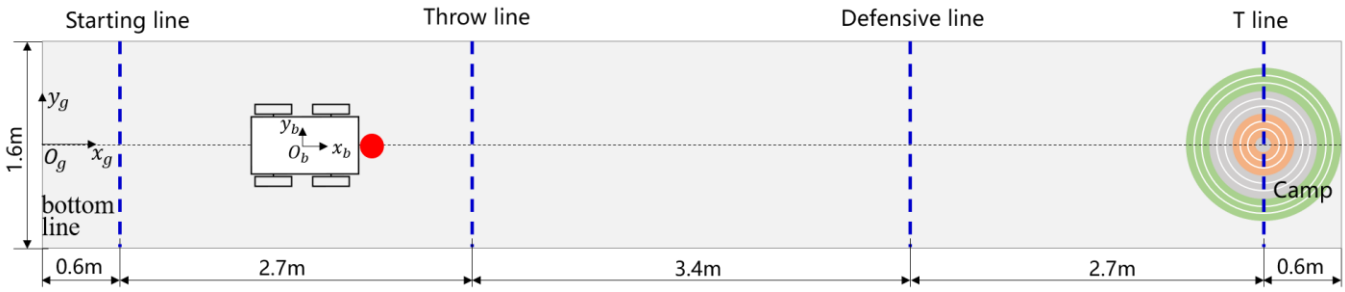

Fig. 2. Curling sheet designed in the digital simulator based on the parameter data of the real-world competition venue and real robot.

Motivated by the above considerations, in this paper, a physics-based digital simulator is built up to train the AI-driven battle generator. Moreover, according to the actual configuration and parameters, the robot dynamics simulation scene is built in the V-REP to achieve high-fidelity simulation of physical processes such as friction, gravity and collision in curling confrontation, thereby narrowing the gap between digital simulators and the real-world.

The rest of the paper is organized as follows: In Section II, the kinematics and inverse kinematics models of curling are established in the case of collision and non-collision, respectively. Section III plans the robot's movement according to the actual competition field. Section IV builds the physics-based digital simulator. Section V concludes this paper with some suggestions for further research.

II. Curling Kinematics Analysis

A. Parameters Definition of Digital Simulator

To narrow the gap between digital simulators and the real world, the physics-based digital simulator built in this paper is based on the parameter data of the real-world competition venue and real robot, as shown in Table I. The ground inertial frame O_g-$x_g y_g$ is established to describe the position and posture of the robot and curling, as shown in Fig. 2.

TABLE I. Parameters of the Digital Simulator

Parameters	Value
Total length of the curling ice sheet l_{total}	10 m
Total width of the curling ice sheet w_{total}	1.6 m
Distance between starting line and bottom line l_{initial}	0.6 m
Distance between throw line and bottom line l_{throw}	3.3 m
Distance between camp center and bottom line l_{center}	9.4 m
Radius of the camp R_{score}	0.6 m
Sliding friction force between curling and ice sheet F_{s_h}	0.3 N
Radius of the curling stone $R_{curling}$	0.1 m
Longitudinal wheel track of mobile platform l_{mobile}	0.45 m
Transverse wheel track of mobile platform w_{mobile}	0.43 m
Maximum speed of mobile platform $v_{mobile\_\max}$	2.7 m/s
Rolling friction between mobile platform and ice F_{m_h}	50 N
Distance between end-effector and mobile platform l_{b_h}	0.4 m

B. Forward Kinematics Model of Curling

Before establishing the kinematics model of curling, it is necessary to judge whether curling collides during the movement and then establish the kinematics models of curling with and without collision, respectively.

1) Judge Whether Curling Collides During Movement

As shown in Fig. 3, given the initial position (x_0, y_0) and velocity (ϕ_0, U_0) of curling, the expression of the curling trajectory line $\overline{O_0O_1}$ can be given by:

$$x\tan(\phi_0) - y - x_0\tan(\phi_0) + y_0 = 0 \tag{1}$$

Then the distance between the curling (x_2, y_2) and the curling trajectory line $\overline{O_0O_1}$ can be expressed as:

$$d_{2\_01} = \frac{\left|x_2\tan(\phi_0) - y_2 - x_0\tan(\phi_0) + y_0\right|}{\sqrt{1+\left(\tan(\phi_0)\right)^2}} \tag{2}$$

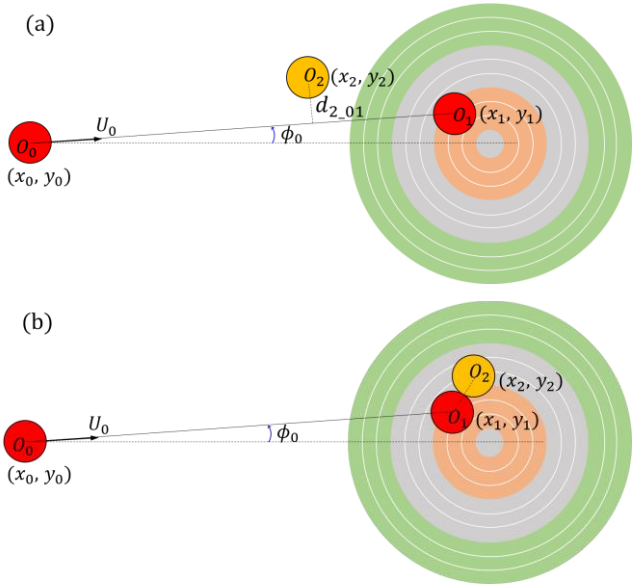

Fig. 3. Motion state parameters and trajectory line used to judge whether curling collides during movement. **(a)** $x_1 \geq x_2$. **(b)** $x_1 < x_2$.

Given the sliding friction force F_{s_h} between curling and the court, according to Newton's second law, we can know that:

$$\begin{cases} (U_0)^2 = \dfrac{2F_{s\_h}}{m_h}\sqrt{(x_1-x_0)^2+(y_1-y_0)^2} \\ \tan(\phi_0) = \dfrac{y_1-y_0}{x_1-x_0} \end{cases} \tag{3}$$

Then the position coordinates expression of the curling trajectory line's end point is obtained as:

$$\begin{cases} x_1 = x_0 + \dfrac{m_h(U_0)^2}{2F_{s\_h}\sqrt{1+(\tan(\phi_0))^2}} \\ y_1 = y_0 + (x_1-x_0)\tan(\phi_0) \end{cases} \tag{4}$$

Based on the above analysis, the criterion for judging the collision of curling during movement can be expressed as:

$$\begin{cases} d_{2\_01} \le 2r_h \ , \quad if\ x_1 \ge x_2 \\ \sqrt{(x_2-x_1)^2+(y_2-y_1)^2} \le 2r_h \ , \quad if\ x_1 < x_2 \end{cases} \tag{5}$$

2) Forward Kinematics of Curling without Collision

When the curling does not collide with other curling, its final position coordinates (x_{1_goal}, y_{1_goal}) can be expressed as:

$$\begin{cases} x_{1\_goal} = x_0 + \dfrac{m_h(U_0)^2}{2F_{s\_h}\sqrt{1+(\tan(\phi_0))^2}} \\ y_{1\_goal} = y_0 + (x_{1\_goal}-x_0)\tan(\phi_0) \end{cases} \tag{6}$$

3) Forward Kinematics of Curling with Collision

As shown in Fig. 4, before the collision, the red curling O_1 approached the static yellow curling O_2 at a speed V_0. The angle between the velocity V_0 and the x-axis of ground inertial frame O_g-$x_g y_g$ is ϕ_0. The included angle between the center line $\overline{O_1O_2}$ and the speed V_0 is ϕ_{0_2}. After the collision, the speed of the red curling becomes V_1 and the speed of the yellow curling becomes V_2. The purpose of forward kinematics is to solve the expressions of x_{1_goal}, y_{1_goal}, x_{2_goal} and y_{2_goal}.

Given the initial position (x_2, y_2) of yellow curling, the initial position (x_0, y_0) and velocity (ϕ_0, U_0) of red curling, we can know that:

$$\begin{cases} 2r_h = \sqrt{(x_1-x_2)^2+(y_1-y_2)^2} \\ \tan(\phi_0) = \dfrac{y_1-y_0}{x_1-x_0} \end{cases} \tag{7}$$

which can be calculated as:

$$\begin{aligned} 4(r_h)^2 = {} & (x_1)^2\left(1+(\tan(\phi_0))^2\right) \\ & -2x_1\left(x_2+(x_0\tan(\phi_0)+y_2-y_0)\tan(\phi_0)\right) \\ & +(x_2)^2+(x_0\tan(\phi_0)+y_2-y_0)^2 \end{aligned} \tag{8}$$

Then according to the root formula of the standard quadratic equation, x_1 and y_1 can be obtained as:

$$\begin{cases} x_1 = \dfrac{-b-\sqrt{\Delta}}{2a} \\ y_1 = y_0 + (x_1-x_0)\tan(\phi_0) \end{cases} \tag{9}$$

where

$$\begin{cases} a = 1+(\tan(\phi_0))^2 \\ b = -2\left(x_2+(x_0\tan(\phi_0)+y_2-y_0)\tan(\phi_0)\right) \\ c = (x_2)^2+(x_0\tan(\phi_0)+y_2-y_0)^2-4(r_h)^2 \\ \Delta = b^2-4ac \end{cases} \tag{10}$$

According to Newton's second law, V_0 is expressed as:

$$V_0 = \sqrt{(U_0)^2 - \frac{2F_{s\_h}}{m_h}\sqrt{(x_1-x_0)^2+(y_1-y_0)^2}} \tag{11}$$

Then V_1 and V_2 can be obtained as:

$$\begin{cases} \phi_{0\_2} = \arctan\left((y_2-y_1)/(x_2-x_1)\right)-\phi_0 \\ V_1 = V_0\sin(\phi_{0\_2}) \\ V_2 = V_0\cos(\phi_{0\_2}) \end{cases} \tag{12}$$

According to Newton's second law, we can know that:

$$\begin{cases} (V_2)^2 = \dfrac{2F_{s\_h}}{m_h}\sqrt{(x_{2\_goal}-x_2)^2+(y_{2\_goal}-y_2)^2} \\ \tan(\phi_0+\phi_{0\_2}) = \dfrac{y_{2\_goal}-y_1}{x_{2\_goal}-x_1} \end{cases} \tag{13}$$

which can be calculated as:

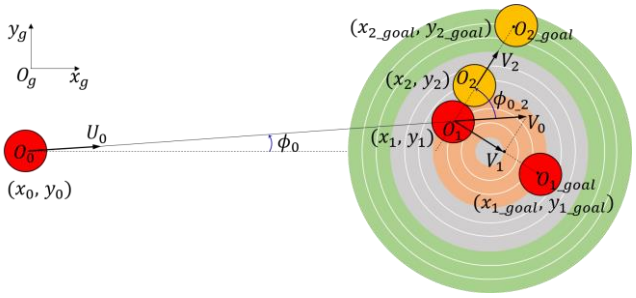

Fig. 4. Changes of motion state parameters of curling before and after collision with other curling.

$$\left(\frac{m_h(V_2)^2}{2F_{s\_h}}\right)^2 = (x_{2\_goal})^2\left(1+\left(\tan(\phi_0+\phi_{0\_2})\right)^2\right) - 2x_{2\_goal}\left(\begin{matrix}x_2+\left(x_1\tan(\phi_0+\phi_{0\_2})+y_2-y_1\right)\\ \tan(\phi_0+\phi_{0\_2})\end{matrix}\right) + (x_2)^2+\left(x_1\tan(\phi_0+\phi_{0\_2})+y_2-y_1\right)^2 \tag{14}$$

According to the root formula of the standard quadratic equation, x_{2_goal} and y_{2_goal} can be expressed as:

$$\begin{cases} x_{2\_goal} = \dfrac{-b+\sqrt{\Delta}}{2a} \\ y_{2\_goal} = y_1 + (x_{2\_goal}-x_1)\tan(\phi_0+\phi_{0\_2}) \end{cases} \tag{15}$$

where

$$\begin{cases} a = 1+\left(\tan(\phi_0+\phi_{0\_2})\right)^2 \\ b = -2\left(x_2+\left(x_1\tan(\phi_0+\phi_{0\_2})+y_2-y_1\right)\tan(\phi_0+\phi_{0\_2})\right) \\ c = (x_2)^2+\left(x_1\tan(\phi_0+\phi_{0\_2})+y_2-y_1\right)^2-\left(m_h(V_2)^2/2F_{s\_h}\right)^2 \\ \Delta = b^2-4ac \end{cases} \tag{16}$$

According to Newton's second law, we can know that:

$$\begin{cases} (V_1)^2 = \dfrac{2F_{s\_h}}{m_h}\sqrt{\left(x_{1\_goal}-x_1\right)^2+\left(y_{1\_goal}-y_1\right)^2} \\ -\tan\left(\dfrac{\pi}{2}-\phi_0-\phi_{0\_2}\right) = \dfrac{y_{1\_goal}-y_1}{x_{1\_goal}-x_1} \end{cases} \tag{17}$$

which can be arranged as:

$$\left(\frac{m_h(V_1)^2}{2F_{s\_h}}\right)^2 = \left(x_{1\_goal}-x_1\right)^2\left(1+\left(\cot\left(\phi_0+\phi_{0\_2}\right)\right)^2\right) \tag{18}$$

Then according to the root formula of the quadratic equation, x_{1_goal} and y_{1_goal} can be obtained as:

$$\begin{cases} x_{1\_goal} = x_1 + \dfrac{m_h(V_1)^2}{2F_{s\_h}\sqrt{1+\left(\cot\left(\phi_0+\phi_{0\_2}\right)\right)^2}} \\ y_{1\_goal} = y_1-(x_{1\_goal}-x_1)\cot(\phi_0+\phi_{0\_2}) \end{cases} \tag{19}$$

C. Inverse Kinematics Model of Curling

The inverse kinematics model calculates the curling throwing speed (ϕ_0, U_0) based on the curling target position given by the AI-driven battle generator.

1) Inverse Kinematics of Curling without Collision

When the curling does not collide with other curling during movement, as shown in Fig. 3(a), according to the analytic geometric relationship, given the initial position (x_0, y_0) and the final position (x_{1_goal}, y_{1_goal}) of curling, its velocity (ϕ_0, U_0) can be expressed as:

$$\begin{cases} \phi_0 = \arctan\left(\dfrac{y_{1\_goal}-y_0}{x_{1\_goal}-x_0}\right) \\ U_0 = \sqrt{\dfrac{2F_{s\_h}}{m_h}\sqrt{\left(x_{1\_goal}-x_0\right)^2+\left(y_{1\_goal}-y_0\right)^2}} \end{cases} \tag{20}$$

2) Inverse Kinematics of Curling with Collision

As shown in Fig. 4, given (x_2, y_2) and (x_{1_goal}, y_{1_goal}), according to the analytic geometric, we can know that:

$$\begin{cases} \left|y_2-y_1\right| = \sqrt{4(r_h)^2-\left(x_2-x_1\right)^2} \\ \dfrac{y_2-y_1}{x_2-x_1} = -\dfrac{x_{1\_goal}-x_1}{\left(y_{1\_goal}-y_2\right)+\left(y_2-y_1\right)} \end{cases} \tag{21}$$

which can be calculated as:

$$\begin{matrix}\left(\left(y_2-y_{1\_goal}\right)^2+\left(x_{1\_goal}-x_2\right)^2\right)K^2 \\ -8(r_h)^2\left(y_2-y_{1\_goal}\right)K \\ +16(r_h)^4-4(r_h)^2\left(x_{1\_goal}-x_2\right)^2=0\end{matrix} \tag{22}$$

According to the root formula of the standard quadratic equation, (x_1, y_1) can be expressed as:

$$\begin{cases} x_1 = x_2-\sqrt{4(r_h)^2-K^2} \\ y_1 = y_2-K \end{cases} \tag{23}$$

where

$$K = \frac{-b+\sqrt{\Delta}}{2a} \tag{24}$$

$$\begin{cases} a = \left(y_2-y_{1\_goal}\right)^2+\left(x_{1\_goal}-x_2\right)^2 \\ b = -8(r_h)^2\left(y_2-y_{1\_goal}\right) \\ c = 16(r_h)^4-4(r_h)^2\left(x_{1\_goal}-x_2\right)^2 \\ \Delta = b^2-4ac \ge 0 \end{cases} \tag{25}$$

To make curling O_1 and O_2 reach position O_{1_goal} and O_{2_goal} respectively after the collision, according to Newton's second law, we can know that:

$$\begin{cases} V_1 = \sqrt{\dfrac{2F_{s\_h}}{m_h}\sqrt{\left(x_{1\_goal}-x_1\right)^2+\left(y_{1\_goal}-y_1\right)^2}} \\ V_2 = \sqrt{\dfrac{2F_{s\_h}}{m_h}\sqrt{\left(x_{2\_goal}-x_2\right)^2+\left(y_{2\_goal}-y_2\right)^2}} \end{cases} \tag{26}$$

which can be arranged as:

$$U_0 = \sqrt{\left(V_1\right)^2+\left(V_2\right)^2+\frac{2F_{s\_h}}{m_h}\sqrt{\left(x_1-x_0\right)^2+\left(y_1-y_0\right)^2}} \tag{27}$$

According to the analytic geometric, ϕ_0 can be obtained as:

$$\phi_0 = \phi_2 - \arctan\left(\frac{V_1}{V_2}\right) \tag{28}$$

where

$$\phi_2 = \arctan\left(\frac{y_2-y_1}{x_2-x_1}\right) \tag{29}$$

III. Robot Motion Planning

After knowing the throwing position (x_0, y_0) and velocity (ϕ_0, U_0) of curling, we also need to plan the motion trajectory of the robot from the initial state to the throwing state, so that the robot can push curling out at a predetermined position (x_0, y_0) and velocity (ϕ_0, U_0).

A. Calculate Robot Position and Velocity at Throw Line

As shown in Fig. 5(a), given the curling's position (x_0, y_0) and velocity (ϕ_0, U_0) at the throwing line, according to the analytic geometric, the state of the robot O_{b2} can be expressed as:

$$\begin{cases} x_{b2} = x_0 - l_{b\_h}\cos(\phi_0) \\ y_{b2} = y_0 - l_{b\_h}\sin(\phi_0) \\ \phi_{b2} = \phi_0 \\ V_{b2} = U_0 \end{cases} \tag{30}$$

B. Calculate Robot Position and Velocity at Starting Line

As shown in Fig. 5(b), given the robot's position (x_{b2}, y_{b2}) and velocity (ϕ_{b2}, V_{b2}), according to the analytic geometric, the state of the robot O_{b1} when curling at the starting line can be expressed as:

$$\begin{cases} x_{b1} = x_0 - (l_{throw} - l_{start}) - l_{b\_h}\cos(\phi_0) \\ y_{b1} = y_0 - (l_{throw} - l_{start})\tan(\phi_0) - l_{b\_h}\sin(\phi_0) \\ \phi_{b1} = \phi_0 \\ V_{b1} = 0 \end{cases} \tag{31}$$

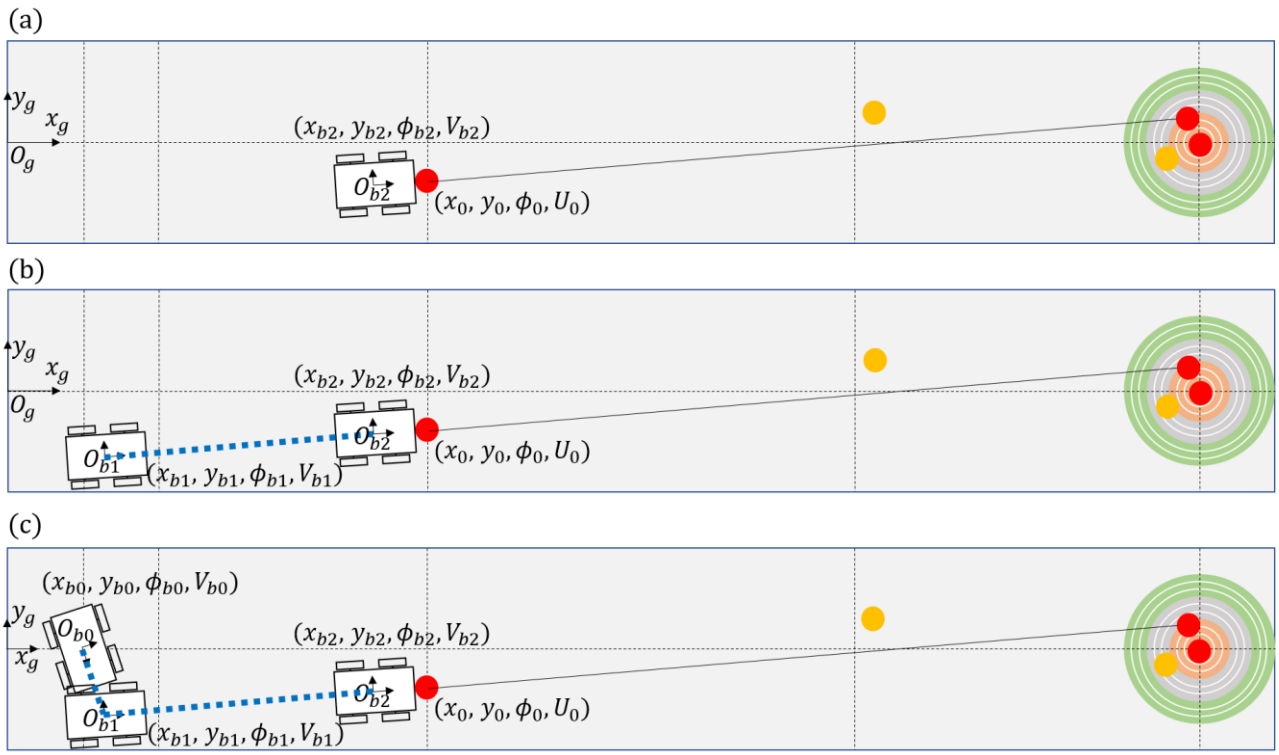

Fig. 5. Motion process of the robot to throw the curling stone at a desired velocity toward a desired direction.

IV. Physics-based Digital Simulator

The physics-based digital simulator is built based on Python and V-REP in this Section. First of all, as shown in Fig. 6, based on the curling kinematics model established in Section II and the robot motion planning model established in Section III, the kinematics simulation calculation is carried out by Python, and the human-computer interaction interface is established for data input and visual presentation of curling motion effect. Secondly, as shown in Fig. 7, the mobile manipulator robot and curling are built based on the dynamic simulation engine V-REP, and the concentric circles of the ice sheet and camp are designed strictly according to the actual competition field. The rolling friction between the four wheels of the mobile manipulator and the ice sheet, and the sliding friction between the curling and the ice sheet are all set as given values (obtained through actual field tests). In addition, curling position, camera pose, illumination direction, and intensity can be adjusted to obtain a large number of simulation training data to train the AI-driven battle generator, as shown in Fig. 8. Finally, the communication link between Python and V-REP platform is established through programming to realize the interaction of motion state data.

The simulation process is as follows: ① Run the Python main program and start the V-REP scene project. ② Input the values of parameters such as y_0, U_0, ϕ_0 by dragging three sliders on the human-computer interaction interface. ③ The digital simulator updates the trajectory and final state of the robot and curling under the corresponding real-time input. ④ Iteratively adjust the values of input parameters, and finally select a set of best parameters, so that curling can score the most points after exercise. ⑤ The motion planning path of the robot under the current parameters is sent to the V-REP engine. ⑥ The robot in V-REP follows the given motion planning trajectory to push the curling motion (the robot is also affected by ice sheet friction). ⑦ Curling stone in V-REP glides on the ice sheet under the push of the robot, and finally hits another curling stone in the camp to score. The movement process of robot and curling stone on the interactive interface in Python is shown in Fig. 9, and the dynamic interaction process of the robot and curling stone in V-REP is shown in Fig. 10.

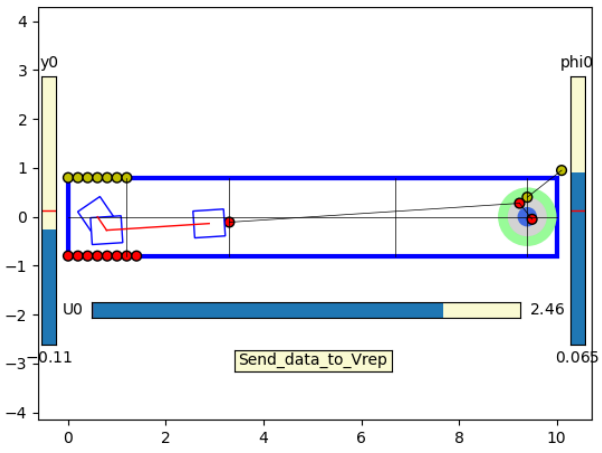

Fig. 6. Human-computer interaction interface based on Python.

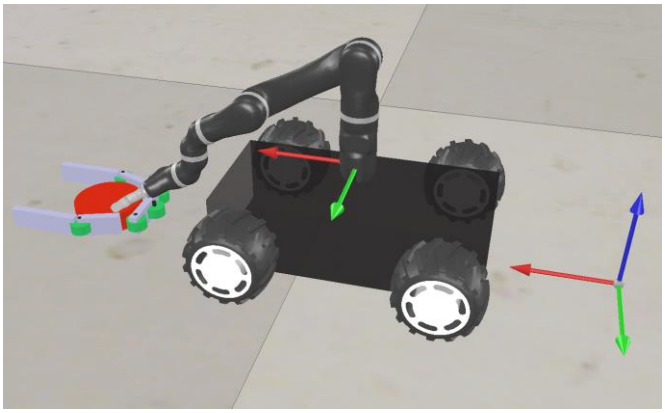

Fig. 7. Mobile manipulator robot and curling in digital simulator.

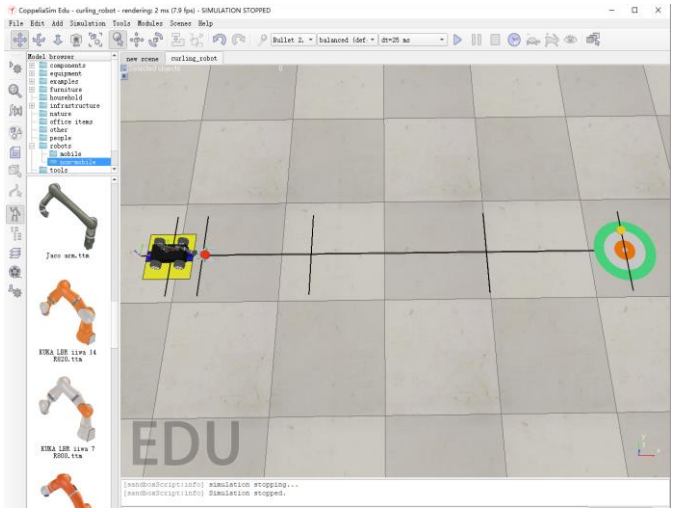

Fig. 8. Curling ice sheet and dynamic interaction module are built based on the dynamic simulation engine V-REP.

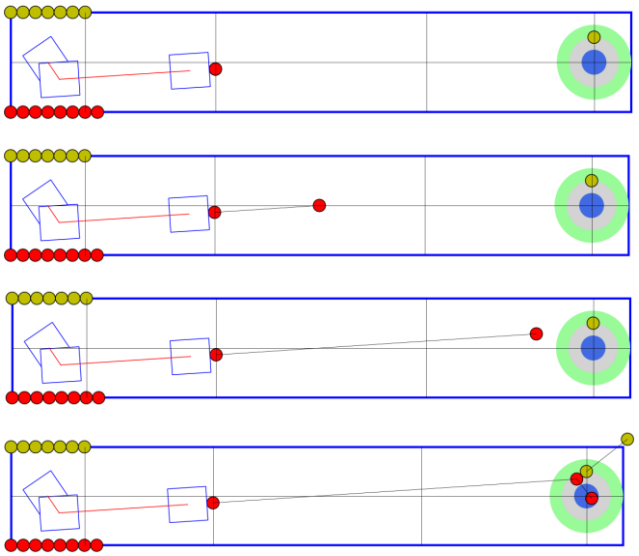

Fig. 9. Movement process of curling stone on the interactive interface.

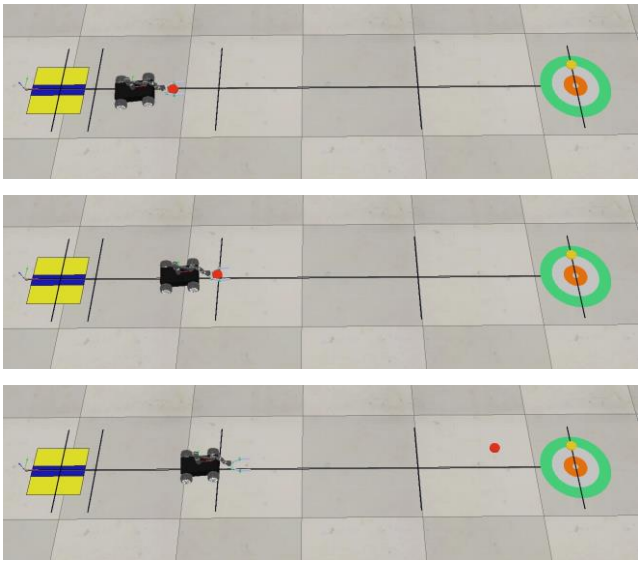

Fig. 10. Dynamic interaction process of robot and curling stone in V-REP.

V. Conclusion and Future Work

In this paper, a physics-based digital simulator is built to train the AI-driven battle generator. Moreover, according to the actual configuration and parameters, the robot dynamics simulation scene is built in the V-REP to achieve high-fidelity simulation of physical processes such as friction, gravity, and collision in curling confrontation, thereby narrowing the gap between digital simulators and the real world.

To simulate the movement process of the real robot on the ice sheet more truly, the robot kinematics model with relative sliding and rotation between tire and ice can be further considered in the future. In addition, to get more complicated and rich battle strategies, a rotating mechanism can be attached to the manipulator end-effector in the future, so that curling can have the desired angular velocity in the process of moving forward, thereby achieving more a flexible motion trajectory.

References

[1] X. Ren, Y. Liu, Y. Hu, and Z. Li, "Integrated task sensing and whole body control for mobile manipulation with series elastic actuators," *IEEE Trans. Autom. Sci. Eng.*, vol. 20, no. 1, pp. 413-424, Jan. 2023.

[2] H. Xing, A. Torabi, L. Ding, H. Gao, Z. Deng, and M. Tavakoli, "Enhancement of force exertion capability of a mobile manipulator by kinematic reconfiguration," *IEEE Robot. Autom. Lett.*, vol. 5, no. 4, pp. 5842-5849, Oct. 2020.

[3] Y. Liu, Z. Li, H. Su, and C. Y. Su, "Whole-body control of an autonomous mobile manipulator using series elastic actuators," *IEEE/ASME Trans. Mechatron.*, vol. 26, no. 2, pp. 657-667, Apr. 2021.

[4] Z. Sun, H. Yang, Y. Ma, X. Wang, Y. Mo, H. Li, and Z. Jiang, "BIT-DMR: A humanoid dual-arm mobile robot for complex rescue operations," *IEEE Robot. Autom. Lett.*, vol. 7, no. 2, pp. 802-809, Apr. 2022.

[5] M. Naazare, F. G. Rosas, and D. Schulz, "Online next-best-view planner for 3d-exploration and inspection with a mobile manipulator robot," *IEEE Robot. Autom. Lett.*, vol. 7, no. 2, pp. 3779-3786, Apr. 2022.

[6] R. J. Yan, E. Kayacan, I. M. Chen, L. K. Tiong, and J. Wu, "QuicaBot: Quality inspection and assessment robot," *IEEE Trans. Autom. Sci. Eng.*, vol. 16, no. 2, pp. 506-517, Apr. 2019.

[7] T. N. Truong, A. T. Vo, and H. Kang, "A novel ANSMC algorithm for tracking control of 3-DOF planar parallel manipulators," *Int. J. Mech. Eng. Robot. Res.*, vol. 12, no. 1, pp. 32-39, Jan. 2023.

[8] Y. Qin, A. Escande, F. Kanehiro, and E. Yoshida, "Dual-arm mobile manipulation planning of a long deformable object in industrial installation," *IEEE Robot. Autom. Lett.*, vol. 8, no. 5, pp. 3039-3046, May 2023.

[9] H. Zhang, Q. Sheng, J. Hu, X. Sheng, Z. Xiong, and X. Zhu, "Cooperative transportation with mobile manipulator: A capability map-based framework for physical human–robot collaboration," *IEEE/ASME Trans. Mechatron.*, vol. 27, no. 6, pp. 4396-4405, Dec. 2022.

[10] Z. Cao, N. Gu, J. Jiao, S. Nahavandi, C. Zhou, and M. Tan, "A novel geometric transportation approach for multiple mobile manipulators in unknown environments," *IEEE Syst. J.*, vol. 12, no. 2, pp. 1447-1455, Jun. 2018.

[11] D. Wahrmann, A. C. Hildebrandt, C. Schuetz, R. Wittmann, and D. Rixen, "An autonomous and flexible robotic framework for logistics applications," *J. Intell. Robot. Syst.*, vol. 93, no. 3-4, pp. 419-431, 2019.

[12] F. Gao, S. Li, Y. Gao, C. Qi, Q. Tian, and G. Yang, "Robots at the Beijing 2022 Winter Olympics," *Sci. Robot.*, vol. 7, no. 65, eabq0785, Apr. 2022.

[13] J. A. Stork, "Preparing to adapt is key for Olympic curling robots," *Sci. Robot.*, vol. 5, no. 46, eabe2547, Sep. 2020.

[14] J. H. Choi, K. Nam, and S. Oh, "High-accuracy driving control of a stone-throwing mobile robot for curling," *IEEE Trans. Autom. Sci. Eng.*, vol. 19, no. 4, pp. 3210-3221, Oct. 2022.

[15] V. Honkanen, M. Ovaska, M. J. Alava, L. Laurson, and A. J. Tuononen, "A surface topography analysis of the curling stone curl mechanism," *Sci. Rep.*, vol. 8, no. 8123, May 2018.

[16] G. Xiao, T. Wu, R. Weng, R. Zhang, Y. Han, Y. Dong, and Y. Liang, "NA-OR: A path optimization method for manipulators via node attraction and obstacle repulsion," *Sci. China-Technol. Sci.*, vol. 66, no. 5, pp. 1205-1213, Apr. 2023.

[17] W. Zhao, A. Pashkevich, D. Chablat, and A. Klimchik, "Mechanics of compliant serial manipulator composed of dual-triangle segments," *Int. J. Mech. Eng. Robot. Res.*, vol. 10, no. 4, pp. 169-176, Apr. 2021.

[18] Y. Sun, K. Zhang, and C. Sun, "Model-based transfer reinforcement learning based on graphical model representations," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 34, no. 2, pp. 1035-1048, Feb. 2023.

[19] D. O. Won, K. R. Müller, and S. W. Lee, "An adaptive deep reinforcement learning framework enables curling robots with human-like performance in real-world conditions," *Sci. Robot.* vol. 5, no. 46, eabb9764, Sep.2020.

[20] K. Lee, S.-A. Kim, J. Choi, and S.-W. Lee, "Deep reinforcement learning in continuous action spaces: A case study in the game of simulated curling," *Int. Conf. Mach. Learn.*, pp. 2943–2952, 2018.

[21] T. Ito, Y. Kitasei, "Proposal and implementation of digital curling," *IEEE Conf. Comput. Intell. Games*, pp. 469–473, 2015.

[22] K. Ohto and T. Tanaka, "A curling agent based on the Monte-Carlo tree search considering the similarity of the best action among similar states," *Int. Conf. Advan. Comput. Games*, pp. 151–164, 2017.

[23] M. H. Heo and D. Kim, "The development of a curling simulation for performance improvement based on a physics engine," *Procedia Eng.*, vol. 60, pp. 385–390, Aug. 2013.

[24] M. Yamamoto, S. Kato, and H. Iizuka, "Digital curling strategy based on game tree search," *IEEE Conf. Comput. Intell. Games*, pp. 474–480, 2015.