AIMY: An Open-source Table Tennis Ball Launcher for Versatile and High-fidelity Trajectory Generation

Alexander Dittrich, Jan Schneider, Simon Guist, Nico Gürtler, Heiko Ott, Thomas Steinbrenner, Bernhard Schölkopf and Dieter Büchler

Abstract— To approach the level of advanced human players in table tennis with robots, generating varied ball trajectories in a reproducible and controlled manner is essential. Current ball launchers used in robot table tennis either do not provide an interface for automatic control or are limited in their capabilities to adapt speed, direction, and spin of the ball. For these reasons, we present AIMY, a three-wheeled open-hardware and open-source table tennis ball launcher, which can generate ball speeds and spins of up to $15.4\,\mathrm{m\,s^{-1}}$ and $192.0\,\mathrm{s^{-1}}$, respectively, which are comparable to advanced human players. The wheel speeds, launch orientation and time can be fully controlled via an open Ethernet or Wi-Fi interface. We provide a detailed overview of the core design features, and open-source the software to encourage distribution and duplication within and beyond the robot table tennis research community. We also extensively evaluate the ball launcher's accuracy for different system settings and learn to launch a ball to desired locations. With this ball launcher, we enable long-duration training of robot table tennis approaches where the complexity of the ball trajectory can be automatically adjusted, enabling large-scale real-world online reinforcement learning for table tennis robots.
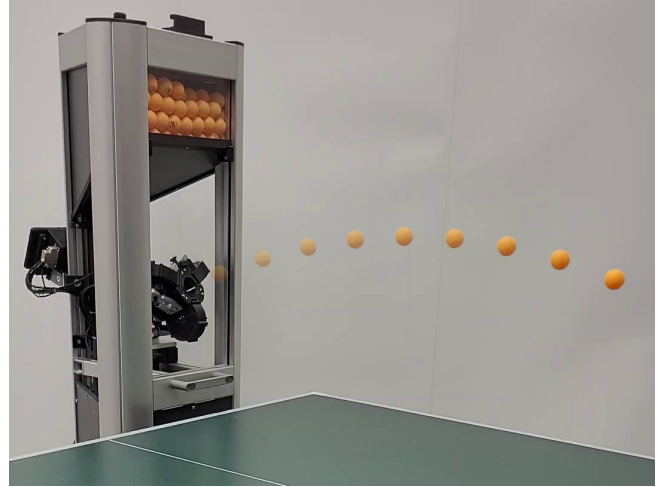
Fig. 1: AIMY, an open-source table tennis ball launcher capable of generating speeds and spins comparable to advanced human players. The five independent degrees of freedom allow launching balls with arbitrary target positions, orientations, and spins.

I. INTRODUCTION

Table tennis is a fast-changing and uncertain challenge for robots, where current systems do not reach professional human performance. In modern competitive table tennis, effective command of ball speed and spin is crucial both for offensive and defensive play. While current state-of-the-art table tennis robots can cope with increasing reliability with medium to high-paced shots, ball trajectories with considerable spin remain a significant obstacle on the way to reach the "AlphaGo moment" of robot table tennis.

Recent successes in robot table tennis [1], [2], [3], [4] show that Reinforcement Learning (RL) has the potential to significantly increase the performance of table tennis robots. However, training with large-scale real-world interactions of robot and ball – as required with RL – is hard to carry out practically and hence these approaches rely on simulations in some form. Simulating contacts between racket and ball and in the bounce with the table require an increasingly complex rebound model, which is more difficult to identify with increasing speed and spin of the ball. This limitation restricts progress in achieving performance comparable to advanced human players.

Consequently, rich real-world data created during training under competitive conditions holds great potential for acquiring even more sophisticated table tennis skills. For meaningful use of this data in sample-intensive learning, replications of a wide range of challenging but reproducible game situations are of great importance. Therefore, a reliable method is needed to generate versatile, high-fidelity ball trajectories.

For this purpose, we developed the high-precision table tennis ball launcher that we name AIMY and depict in Figure 1. In experiments, we show that AIMY reliably generates arbitrary trajectories with translational speeds up to $15.4\,\mathrm{m\,s^{-1}}$ and spins of approximately $192.0\,\mathrm{s^{-1}}$ with high accuracy. For comparison, Lee et al. [5] measured average ball speeds of $16.8\,\mathrm{m\,s^{-1}}$ and ball spins with $113.9\,\mathrm{s^{-1}}$ among players on national table tennis team level, which used maximum strength to hit a defined target.

To make AIMY as adjustable and easy to use as possible, the user can adjust the wheel velocities and the orientation of the ball launch unit via a remote low-latency Ethernet interface. We provide a Python API to modify these parameters and enable controlled and precisely timed launching.

The contribution of this paper is four-fold: We (i) open-source the hardware and the software interface[1], (ii) quantify the accuracy of our new ball launcher under varying

[1]AIMY project web page: https://webdav.tuebingen.mpg.de/aimy/

hardware considerations, (iii) learn to return a ball to a desired landing position on the table with a supervised learning approach and (iv) provide a rich and big data set of 3761 ball trajectories of various ball speeds and spins for benchmarking model learning approaches. In this manner, we hope to facilitate research on more complex scenarios in robot table tennis.

In the following, we provide an overview of the core design features and components, along with the corresponding low-level control software, in Section III. Subsequently, Section IV reports the evaluations on repetition accuracy for different system parameters. Furthermore, AIMY is validated for different target distances compared to commercially available products. Section V describes a supervised learning approach to target shooting, as well as the data set of various ball trajectories for model learning.

II. RELATED WORK

The majority of past works in robot table tennis focussed on data-efficient approaches to return table tennis balls of lower speeds and less spin compared to professional play, and therefore did not require an automated ball launcher. For example, multiple studies use real-world data on a small scale such as in Matsushima et al. [6], Huang et al. [7], Ma et al. [8], Koç et al. [9], Gomez-Gonzalez et al. [10] and Mülling et al. [11] that either use some form of iterative learning control or imitation learning approaches. All these approaches required only small sample sizes. More recent approaches that use RL, avoid long-term training with real balls by using prerecorded data and train in a hybrid sim and real setting (Büchler et al. [3]), use data efficient approaches and reduce the task to its minimum complexity (Tebbe et al. [1]), or leverage sophisticated simulation environments (Yang et al. [12] and Abeyruwan et al. [13]).

A large number of commercially available table tennis launchers exist and are tailored for training with humans ranging from beginners to advanced players but not for automated long-term training. The TTmatic 303A [14], for example, accelerates the ball with a single wheel and a manually exchangeable slide that alters the altitude of the launching angle. In an azimuthal motion, the launching frequency as well as the motor speed are adapted via a physically wired control panel. Usage of this ball launcher for long-term training is not feasible since launching parameters and launching frequency have to be changed manually. Additionally, it cannot adapt the spin of the ball. Other models like the Donic Newgy Robo-Pong 2050 [15] feature the manual rotation of the single wheel launch unit to enable restricted spin generation. More sophisticated products like the PowerPong Table Tennis Robot series [16] and the Butterfly AMICUS series [17] rely on a three-wheel design concept to generate noteworthy spin. However, neither of these products provide an API that allows researchers to control launching parameters and launching.

The number of reports on custom-designed ball launchers in the scientific literature is fairly limited. For example, [18], [19] facilitate automated devices for table tennis ball launching, but neither of them satisfies the need for remote controlled, high-fidelity ball trajectory generation.

III. DESIGN OVERVIEW

Multiple objectives guided the development of the ball launcher. This section summarizes them, followed by the presentation of the core mechanical design features and the software framework.

A. Design Objectives

The main intention for creating the ball launcher is to generate versatile ball trajectories in a controlled and repeatable fashion to facilitate research in robot table tennis. The following list contains detailed and specific derivations of this main goal that steered the development of AIMY:

- **Arbitrary Trajectory Generation (R1):** Sufficient coverage of target area on an International Table Tennis Federation (ITTF) compliant table tennis table; capability to generate ball speed (approx. $16\,\mathrm{m\,s^{-1}}$) and spin (approx. $114\,\mathrm{s^{-1}}$) comparable to professional human level.
- **High-fidelity Launching (R2):** Sufficiently repetitive, low-deviation target point hitting; accuracy should be equal or better than commercially available products; low number of trajectories with large deviation (outliers) for a fixed parameter setting.
- **Suitability for Long-duration Usage (R3):** Characteristics of the ball trajectory should remain the same for the same parameters also under long-duration measurement conditions; reliable supply of the launch unit without clogging of ball supply; reliable launching without interrupts by the launch unit.
- **Remote Controllability (R4):** Remote online parameter control in soft real-time (approx. $500\,\mathrm{ms}$); remote on demand ball launching in soft real-time (approx. $500\,\mathrm{ms}$); launching also in irregular intervals, e.g., due to computation time of policy optimization after epochs.
- **Advanced Trajectory Specification (R5):** Automated launch parameter selection for specification of landing position, angle of ball impact, ball speed, and ball spin.
- **Costs (R6):** Material and component costs in total should not significantly exceed purchase prices of commercial products and should provide a good compromise between the necessary accuracy, longevity, and price.
- **General Handling Aspects (R7):** Manufacturability with regular workshop equipment; assemblability; maintainability; extensibility; user-friendly installation and mechanical parameter adjustment.

B. Hardware Overview

With these design objectives in mind, AIMY is divided into distinct functional core units, which are depicted in Figure 2a. The most important of these core units are the launch unit, the ball feeding system consisting of the ball feeding unit and the ball supply channel, and the reservoir stirrer.
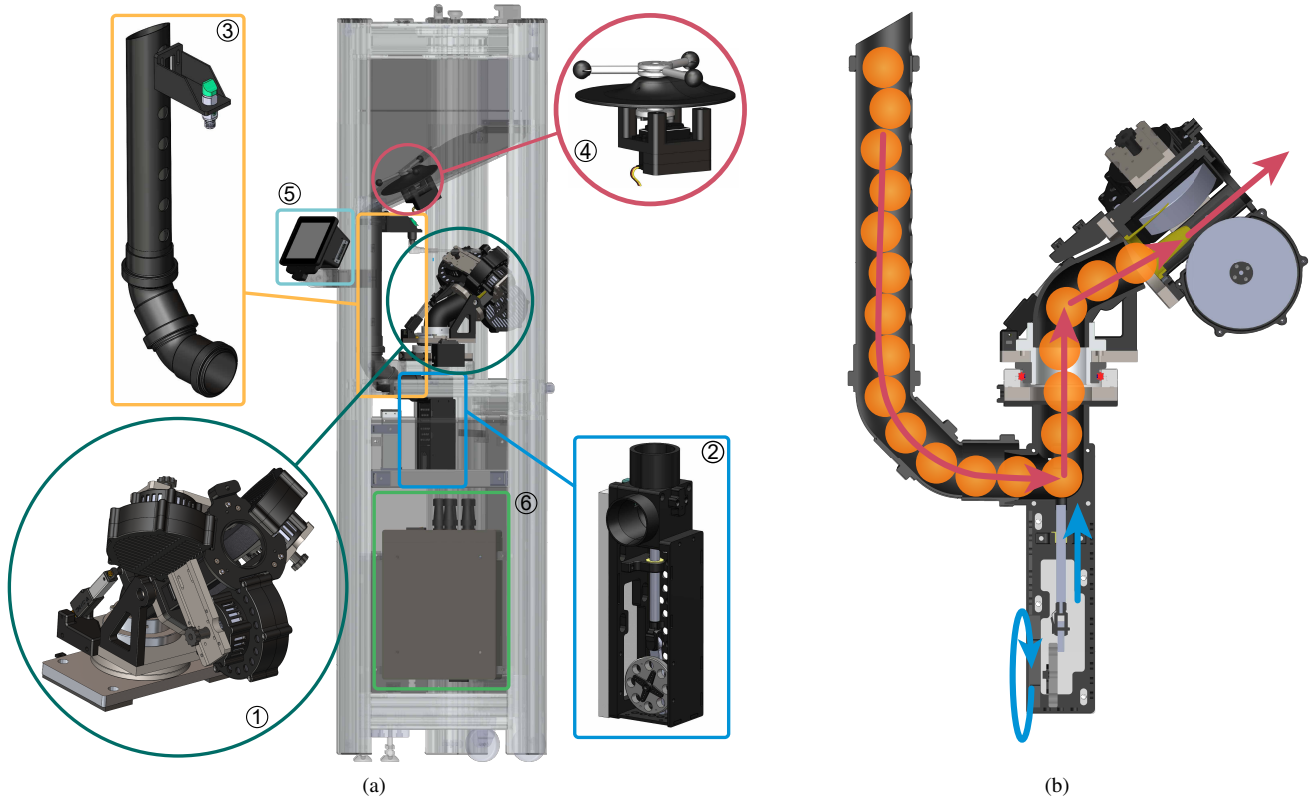
Fig. 2: **(a)** Overview of AIMY with core functional units. ① Launch unit in three-wheel configuration. The launch unit is rotatable due to a ball bearing in combination with a servo motor and adjustable in altitude angle via a linear motor. ② Ball feeding unit with crank mechanism for supplying the launch unit with balls from the ball supply channel. ③ Ball supply channel connecting ball supply unit with ball reservoir. The ball supply channel is equipped with an optical sensor for detecting the state of filling. ④ Reservoir stirrer generates movement in the ball reservoir for resolving ball clogging. ⑤ The control unit enables hardware control, hosts the ball launcher server for remote controllability, and provides a graphical user interface via a 7-inch touchscreen. ⑥ Electronics box with a 5 V and 24 V power supply and ESCs for motor control. **(b)** Sectional view of the ball feeding flow from reservoir to launch unit. Ball supply mechanics from ball supply channel through ball feeding unit towards launch unit and launch, shown in a cross-section view of the respective components. Red arrows indicate ball flow, blue arrows indicate flow of forces via the crank mechanism.

1) Ball Feeding System: The ball feeding system ensures that the launch unit is supplied with table tennis balls from the ball reservoir. The ball feeding unit uses a crank mechanism to transform the rotary motion of the servomotor (Hitec HSB-9381TH) into a linear motion, pushing the balls in queue into the engagement point of the throwing wheels of the launch unit. Figure 2b illustrates this process. Furthermore, an optical sensor (Pepperl+Fuchs GLV18-8-400-S) mounted at the ball supply channel detects the state of filling and triggers the reservoir stirrer if required.

2) Launch Unit: The launch unit receives a ball from the ball feeding system and accelerates it via three throwing wheels that are arranged at 120° to each other. Inspired by commercial products like the Butterfly AMICUS Prime or the PowerPong Omega, this concept can generate significant amounts of spin around the launching tube (yaw) and the horizontal ball axis (pitch). Roll spin can be produced to a limited extent via suitable selection of orientation and wheel speed parameters. For actuating the wheels, the high-performance brushless motor Antigravity MN5008 KV170 of the manufacturer T-MOTOR has been selected. Although developed for drone propulsion, this series of motors proved to be reliable in other robotics applications, such as in [20] and [21]. Hard foam wheels by the manufacturer Butterfly – also used in the AMICUS table tennis launcher series – transfer the motor torque to the ball. The compliant body and the roughened surface reduces slip between the ball and wheel. The T-MOTOR AIR 40A electronic speed controller supplies the motors of the launch unit. A regular 5 V servomotor (Hitec D954SW 32-bit) adapts the azimuthal orientation, allowing the launch unit to rotate by 17° in both positive and negative direction. The altitude orientation of the launch unit can be modified via a linear actuator (Actuoinx L12-30-210-6-R) enabling angles from 6.5° to 37.0°. The selected linear actuator provides higher holding torque than comparable rotary servo motors and is therefore better suited

for the intended use.

3) Reservoir Stirrer: One characteristic of table tennis balls is a rough surface, which players leverage to their advantage by creating spin with rubberized table tennis rackets. However, this surface also promotes clogging of the balls in the ball reservoir. This issue results in the launch unit no longer being supplied with balls, which potentially disrupts the operation of the ball launcher and ends long-term tests. Therefore, in consideration of objective R3, a reservoir stirrer is added. The stirrer moves the balls in the reservoir to resolve potential clogging. The motion is either started automatically after each launch or optionally by the signal of the sensor of the ball supply channel. A servo motor (Pololu SpringRC SM-S4303R) puts the stirrer into motion and can be equipped with different stirring attachments.

4) Control Unit: A network-compatible terminal device adjusts the hardware remotely. For this purpose, a Raspberry Pi 4 Model B with Raspberry Pi OS is used, since it offers a wide range of network interfaces but also enough GPIOs to directly address sensors and actuators. Xinabox OC05 servo drivers directly connected to the Raspberry Pi via the GPIOs control the servomotors for the orientation of the launch unit, the reservoir stirrer and the ball feeding unit.

C. Software Framework

The software is structured in a client-server scheme, where the control unit with direct access to the hardware hosts a server. Any device connected to the network can start a client for remote controlling the hardware. The server provides on-request functions for changing the current system state, such as orientation of the launch unit or wheel speeds, and can initiate launching at a desired time. While the server waits for control requests, it observes the state of the ball queue within the supply channel and activates the reservoir stirrer if necessary. The client software provides functions to submit requests to the server to initiate changes of the ball launcher's parameters and to initiate launching. For the client we provide a Python and C++ API, to enable embedding, e.g., in an RL environment. Server and client can be connected via regular Ethernet (wired) or Wi-Fi (wireless) networks. For communication, the open-source and widely supported messaging library ZeroMQ is used in a TCP configuration. Additionally, we provide a simple to use Tkinter graphical user interface for manual operation and testing of AIMY on a Raspberry Pi 7-inch touchscreen.

IV. DESIGN AND ACCURACY EVALUATION

This section provides a detailed evaluation of the defined objectives of Section III. We first introduce the experimental setup, followed by an investigation of the ball launcher's accuracy for different hardware settings and distances. We close this section by comparing AIMY's capability to generate ball speed and spin in comparison to professional human table tennis players.

A. Experimental Setup

For the evaluation, four optical high-speed cameras (Prosilica Gigabit GE640C) record the ball trajectories at 200
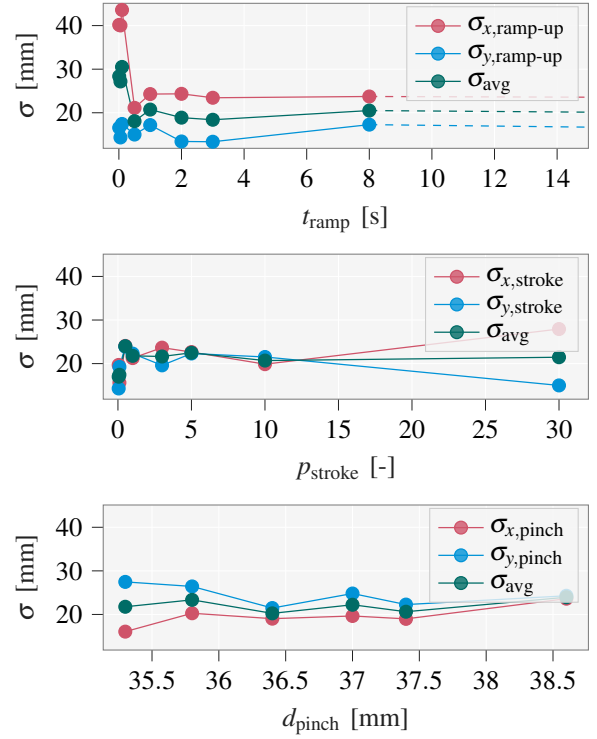

Fig. 3: Standard deviation with respect to ramp-up times, stroke gain, and pinching diameter. The deviation increases significantly for ramp-up times lower than 0.5 s. The standard deviations stay consistent for ramp-up times larger than 0.5 s up until continuous operation (illustrated as dashed line). Small stroke gains show small deviations, while high stroke gain result in large differences between the deviation in x (along long table edge) and y (along short table edge) direction. The differences in deviations for ball pinch are small. High pinching diameters show slightly increased deviations.

frames per second. The tracking system is presented in [22]. For assessing the ball launcher's accuracy, we evaluate the standard deviation of the landing points of the respective ball trajectories. Due to the scatter of measurements and the discretization of the true trajectory, two polynomials are fitted before and after the sample with the lowest height. The intersection of the polynomials provides a better approximation of the true landing point than the data point with the lowest height. First-order polynomials are sufficient for this purpose and show the most reliable results. Before evaluation, a filter removes erroneous measurement samples of the recorded ball trajectories, and we inspected the ball trajectories manually for plausibility.

B. Evaluation of System Parameters

Besides the control parameters, we identified three system parameters (the ramp-up time of the motors, the stroke gain of the ball feeding unit and ball pinching in the launch unit). We evaluate the influence of these parameters on the reproducibility of launched ball trajectories in the following. Figure 3 depicts the accuracy evaluation of all three
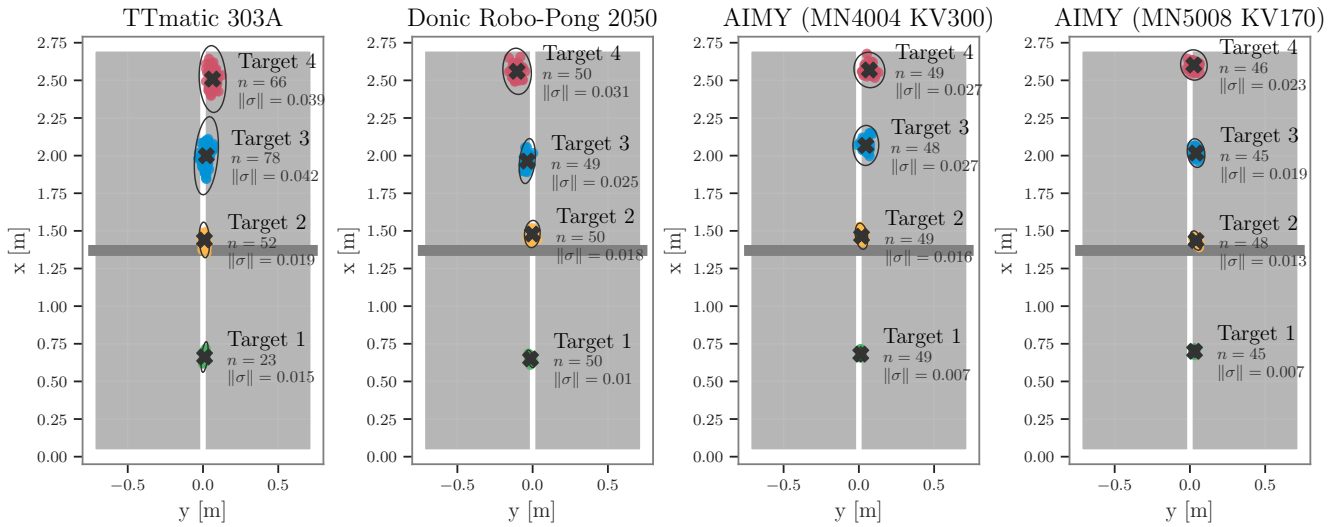
Fig. 4: Distance accuracy experiment for each ball launcher with medium altitude angle. The standard deviation for different target positions for each ball launcher is evaluated. Each measurement point represents an approximated landing point of a ball trajectory on the table and is colored according to the affiliation to each distance measuring series. Additional information is given for each measuring series, like the number of measurement samples, the mean and confidence ellipses with 4 times the standard deviation, and the norm of the respective standard deviation $\|\sigma\|$ are indicated for each measuring series.

parameters. The ramp-up time is the time between launch initiation and actuation of the ball supply unit. During this time, the motors accelerate from standstill to the required speed. Figure 3 shows that ramp-up times below 0.5 s have a negative influence on the accuracy of the ball launcher. Measurements in continuous operation show comparable deviations of ramp-up times above 2 s and simultaneously provide a smoother operation of the motors. The stroke gain influences the speed at which the balls are fed to the launch unit via the crank mechanism. With each control pass, the stroke gain increases the angle of the servo motor, causing steady insertion of the balls to the throwing wheels in the launch unit. The evaluation in Figure 3 shows that small stroke gains result in smaller deviations, while increasing the total launch time. A stroke gain of 0.05, e.g., requires 4.21 s from actuation until launch, while a stroke gain of 1.0 only takes 1.39 s. Ball pinching refers to the compression of the ball by the wheels during the launch procedure. The individual wheel units can be installed on the launch unit with different radial distances to the center. The smaller the distance, the higher the pressure generated on the ball during the launching process. The evaluation shows that increased pinching results in lower deviation of the landing points.

C. High-fidelity Launching

To assess the fidelity of AIMY, the accuracy is evaluated in comparison with the commercially available ball launchers TTmatic 303A and Donic Robo-Pong 2050 for four different target distances. Since the characteristics of motors powering the throwing wheels have a major influence on accuracy, we selected and evaluated two different motors, the T-MOTOR Antigravity MN4004 KV300 and the T-MOTOR Antigravity MN5008 KV170 motors. Both motors have been installed

with the same system parameters, control parameters, and experimental conditions. Launcher-specific differences of the custom-developed launcher to the commercial launchers due to different operation principles are reduced by the selection of comparable launch altitude angles and mechanical guides. Figure 4 shows that AIMY with both motors has a lower deviation of the landing points of Target 1, 2 and 4 than both commercial products. The TTmatic 303A showed an increased deviation on Target 3, which was caused by a parameter jump of the ball feed mechanism for low launch rates. The Robo-Pong 2050 shows a slighly smaller deviation on Target 3 than AIMY with MN4004 motors. The MN5008 motors have lower deviations on Target 2 to 4 than the MN4004 version, while MN4004 has higher accuracy on Target 1. However, the differences of both motors on Target 1 are not significant. Similar results can be observed for the lowest and highest altitude settings in other measurement series. In conclusion, based on the experiments, it can be stated that AIMY has higher fidelity than the commercial products. In the presented measurement series, no outlier was detected or manually removed. Therefore, AIMY satisfies design objective R2 regarding the accuracy, reliable operation and a low number of outliers.

D. Suitability for Reinforcement Learning

The sensor for the state of filling of the ball supply channel and the reservoir stirrer ensure a continuous and reliable supply of the launch unit. In the measurement series presented here, balls could be reliably shot remotely and at arbitrary times with delay times lower than 500 ms via the Python API. Based on the ramp-up time evaluation, we recommend setting a delay of three seconds in case of noncontinuous operation of the motors. In summary, AIMY

sufficiently satisfies suitability for long-duration operation (R3) and fully satisfies remote controllability (R4).

E. Comparison to Human Performance

A maximum ball speed of $15.4\,\mathrm{m\,s^{-1}}$ generated by AIMY is measured, which is slightly lower than the average maximum force serves with $16.8\,\mathrm{m\,s^{-1}}$ of national team level players measured by Lee et al. [5]. AIMY generates top spins up to $192.0\,\mathrm{s^{-1}}$, measured by observing the change of orientation of marked table tennis balls with 960 frames per second recordings. This value significantly exceeds the average of $113.9\,\mathrm{s^{-1}}$ measured for human professional players. Ball speeds and ball spins have been recorded independently of each other.

V. TARGET SHOOTING

AIMY enables precise adjustment of training conditions for learning algorithms. To be able to set the ball trajectories to meet the demands of controllable training scenarios, we developed an algorithm which selects suitable control parameters based on the desired state parameters, e.g., the landing point, initial rotational yaw, roll of the ball, angle of impact of the landing point, or rebound direction. The use of an analytical model introduces approximation errors of non-linear effects such as inelastic and rolling collision dynamics at the rebound and friction and slip in the launch unit. Therefore, we decided to use a feed-forward neural network for the target shooting algorithm.

A. Training Data

A balanced and rich data set has a decisive influence on whether suitable control parameters can be found for the desired state parameters. A grid selects a balanced set of control parameters for the training data set of the target shooting algorithm. Table I describes the composition of the data used for target shooting. The data is biased towards hitting the opponent's half of the table. Therefore, higher velocities and higher angles occur more often. The data set also includes trajectories that do not bounce on the table. Of the 3761 trajectories provided, 3250 have a landing point on the table. Our project web page provides the used data set,

TABLE I: Characteristics of the training data set

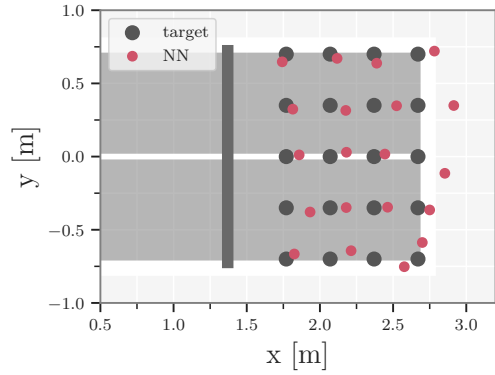| #	Number samples	Spin magnitude	Wheel speeds	Altitude angle
1	415	none to low	all wheels same	various
2	64	high	high speeds	various
3	364	low to medium	low speeds	various
4	1103	low to high	various	high
5	1385	low to medium	various	medium
6	430	low	various	low
Total	**3761**			

Fig. 5: Target shooting experiment. Grey markers represent the targets. Red markers indicate landing positions of ball trajectories generated with the neural network approach. While the landing points roughly match the target locations, deviations are visible.

the trained model for target shooting along with evaluation scripts.

B. Evaluation

The model consists of a feedforward, fully connected neural network with three hidden layers. Input to the model is the desired landing position in Cartesian coordinates. The model outputs the set of control parameters. For evaluation, we selected a grid of 20 points on the opposite half of the table. Figure 5 illustrates the result. The obtained landing points indicate that the model can roughly hit desired landing points.

VI. CONCLUSION

In this work, we provide an open-source table tennis ball launcher capable of generating versatile ball trajectories in a reproducible and controlled manner. The design satisfies the requirements to enable long-duration training of robot table tennis. AIMY has higher accuracy than commercially available products and additionally features remote APIs, crucial for embedding in an RL setting. In a first learning attempt, we showed that AIMY shoots balls close to a desired landing position on the table. The limitations of this work include a missing comparison of AIMY with latest commercial products like the Butterfly AMICUS Prime and the precision of the proposed target shooting approach. More sophisticated approaches using RL or incorporate analytical knowledge of the ball launcher and the ball dynamics could fully utilize AIMY's accuracy. In future work, we will investigate how to return and generate different kinds of spin with a real robot. We deeply hope that AIMY will contribute to increased performance of table tennis robots in the future. Videos of the experiments, as well as construction details, can be found on the project webpage.

REFERENCES

[1] J. Tebbe, L. Krauch, Y. Gao, and A. Zell, "Sample-efficient reinforcement learning in robotic table tennis," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 4171–4178.

[2] L. Yang, H. Zhang, X. Zhu, and X. Sheng, "Ball motion control in the table tennis robot system using time-series deep reinforcement learning," *IEEE Access*, vol. 9, pp. 99 816–99 827, 2021.

[3] D. Büchler, S. Guist, R. Calandra, V. Berenz, B. Schölkopf, and J. Peters, "Learning to play table tennis from scratch using muscular robots," *IEEE Transactions on Robotics*, pp. 1–11, 2022.

[4] S. Abeyruwan, L. Graesser, D. B. D'Ambrosio, A. Singh, A. Shankar, A. Bewley, and P. R. Sanketi, "i-sim2real: Reinforcement learning of robotic policies in tight human-robot interaction loops," 2022. [Online]. Available: https://arxiv.org/abs/2207.06572

[5] M. J. C. Lee, H. Ozaki, and W. X. Goh, "Speed and spin differences between the old celluloid versus new plastic table tennis balls and the effect on the kinematic responses of elite versus sub-elite players," *International Journal of Racket Sports Science*, vol. 1, 6 2019. [Online]. Available: http://hdl.handle.net/10481/57324

[6] M. Matsushima, T. Hashimoto, M. Takeuchi, and F. Miyazaki, "A learning approach to robotic table tennis," *IEEE Transactions on Robotics*, vol. 21, no. 4, pp. 767–771, 2005.

[7] Y. Huang, D. Büchler, O. Koç, B. Schölkopf, and J. Peters, "Jointly learning trajectory generation and hitting point prediction in robot table tennis," in *2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids)*. IEEE, 2016, pp. 650–655.

[8] H. Ma, D. Büchler, B. Schölkopf, and M. Muehlebach, "A learning-based iterative control framework for controlling a robot arm with pneumatic artificial muscles," in *Robotics: Science and Systems*, 2022.

[9] O. Koç, G. Maeda, and J. Peters, "Online optimal trajectory generation for robot table tennis," *Robotics and Autonomous Systems*, vol. 105, pp. 121–137, 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0921889017306164

[10] S. Gomez-Gonzalez, G. Neumann, B. Schölkopf, and J. Peters, "Adaptation and robust learning of probabilistic movement primitives," *IEEE Transactions on Robotics*, vol. 36, no. 2, pp. 366–379, 2020.

[11] K. Mülling, J. Kober, O. Kroemer, and J. Peters, "Learning to select and generalize striking movements in robot table tennis," *The International Journal of Robotics Research*, vol. 32, no. 3, pp. 263–279, 2013.

[12] L. Yang, H. Zhang, X. Zhu, and X. Sheng, "Ball motion control in the table tennis robot system using time-series deep reinforcement learning," *IEEE Access*, vol. 9, pp. 99 816–99 827, 2021.

[13] S. Abeyruwan, L. Graesser, D. B. D'Ambrosio, A. Singh, A. Shankar, A. Bewley, and P. R. Sanketi, "i-sim2real: Reinforcement learning of robotic policies in tight human-robot interaction loops," *arXiv preprint arXiv:2207.06572*, 2022.

[14] "TTmatic 303 A," http://www.megaspin.net/download/ttmatic/2007_catalog.pdf, accessed: 2022-08-21.

[15] "Donic Newgby Robo-Pong 2050," https://www.donic.com/donic/roboter/3455/donic-newgy-robo-pong-2055-inkl.-72-donic-coach-p40-baelle?c=29, accessed: 2022-08-21.

[16] "Power Pong Omega," https://www.powerpong.org/collections/professional-robots/products/power-pong-omega-table-tennis-robot, accessed: 2022-08-21.

[17] "AMICUS PRIME Roboter," https://de.butterfly.tt/amicus-expert.html, accessed: 2022-08-21.

[18] K. Nemire, M. Goettsche, and B. Bridgeman, "Automated system for ball launching, visual occlusion, and data acquisition in a ball-hitting task," *Behavior Research Methods, Instruments, & Computers*, vol. 23, no. 1, pp. 36–44, Mar 1991. [Online]. Available: https://doi.org/10.3758/BF03203333

[19] B. Ponnusamy, W. F. Yong, and Z. Ahmad, "A low cost automated table tennis launcher," *ARPN Journal of Engineering and Applied Sciences*, vol. 10, no. 1, pp. 291–296, Jan 2006.

[20] M. Wüthrich, F. Widmaier, F. Grimminger, J. Akpo, S. Joshi, V. Agrawal, B. Hammoud, M. Khadiv, M. Bogdanovic, V. Berenz, *et al.*, "Trifinger: An open-source robot for learning dexterity," *arXiv preprint arXiv:2008.03596*, 2020.

[21] F. Grimminger, A. Meduri, M. Khadiv, J. Viereck, M. Wüthrich, M. Naveau, V. Berenz, S. Heim, F. Widmaier, T. Flayols, *et al.*, "An open torque-controlled modular robot architecture for legged locomotion research," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 3650–3657, 2020.

[22] S. Gomez-Gonzalez, Y. Nemmour, B. Schölkopf, and J. Peters, "Reliable real-time ball tracking for robot table tennis," *Robotics*, vol. 8, no. 4, 2019. [Online]. Available: https://www.mdpi.com/2218-6581/8/4/90

APPENDIX

A. *Technical Specifications*

Table II provides a brief overview of the specifications of AIMY.

TABLE II: Technical Specifications

Specification	Value	Remarks
Maximum Speed	$15.4\,\mathrm{m\,s^{-1}}$	Average speed in first 0.5 s
Maximum Spin	$192.0\,\mathrm{s^{-1}}$	Initial spin
Altitude Angle	$6.4° - 37.1°$	
Azimuthal Angle	$-15.8° - 15.6°$	
Supply Voltage	$230\,\mathrm{V}$ (50 Hz)	
Python API	✓	
C++ API	(✓)	Not extensively tested
Ethernet	✓	
Wi-Fi	✓	

B. *Third Party Hardware*

Table III lists third party hardware selected for AIMY. In Section IV-C, the evaluation assessed two different motors, both being listed. We recommend the T-MOTOR Antigravity MN5008 KV170, as it provides higher torque in the lower turning speed region. The ball launcher mainly operates in lower turning speed regions, which results in higher accuracy in comparison to the T-MOTOR Antigravity MN4004 KV300.

TABLE III

Component	Manufacturer	Model
Motor Wheels	T-MOTOR	Antigravity MN4004 KV300
Motor Wheels	T-MOTOR	Antigravity MN5008 KV170
ESC	T-MOTOR	AIR 40A
Servomotor Azimuthal Orientation	Hitec	D954SW
Servomotor Altitude Orientation	Actuoinx	L12-30-210-6-R
Servomotor Reservoir Stirrer	Pololu	SpringRC SM-S4303R
Servomotor Ball Feeding Unit	Hitec	HSB-9381TH
Optical Sensor Ball Feeding Unit	Pepperl+Fuchs	GLV18-8-400-S
Control Unit	Raspberry Pi Foundation	Raspberry Pi 4 Model B
Motor Driver Wheels	Xinabox	OC05
24 V Power Supply Unit	Mean Well	HDR-150-24
5 V Power Supply Unit	Mean Well	MDR-40-5
Azimuthal Bearing	Franke	LEL1_5_0070
Interface	Raspberry Pi Foundation	Raspberry Pi Touch Display (7-inch)
Throw wheel	Butterfly	AMICUS Ball Throw Wheel
Housing	item	various

C. *Experimental Setup*

In addition to the description of the experimental setup in Section IV-A, we provide further details on the setup, and the approach of processing and collecting data for evaluation. After recording and before evaluation, we transform the data points to a common coordinate system and preprocess the trajectories according to the following procedure.

- **Measurement System:** The accuracy of the ball tracking system has been evaluated beforehand. We measured a deviation of $0.024\,\mathrm{m}$ in x-direction, $0.011\,\mathrm{m}$ in y-direction and $0.001\,\mathrm{m}$ in z-direction per meter within the relevant recording region.
- **Table Tennis Launcher:** For high-fidelity evaluation, we assess our custom-designed ball launcher in two different configurations. The first configuration is equipped with the T-MOTOR MN4004 KV300 and the second configuration equipped with the T-MOTOR MN4004 KV300 for accelerating balls in the desired direction. In addition to the motors, the structure of the launch unit was made mechanically stiffer in the second configuration, which may have an impact on accuracy in addition to motor differences. For system parameter evaluation, the second configuration is used. In the current version of AIMY, there are no sensors installed for measuring turning speeds of the motors and providing closed-loop control of turning speeds. Individual motor differences in motor speeds for the same actuation are compensated with measured characteristic motor curves for each motor. The characteristic motor curve maps actuation and resulting motor speeds and interpolates between each measurement point. Figure 6 shows the characteristic motor curves. Figure 7 shows the measured actuation to angle change relationship of the orientation actuators.
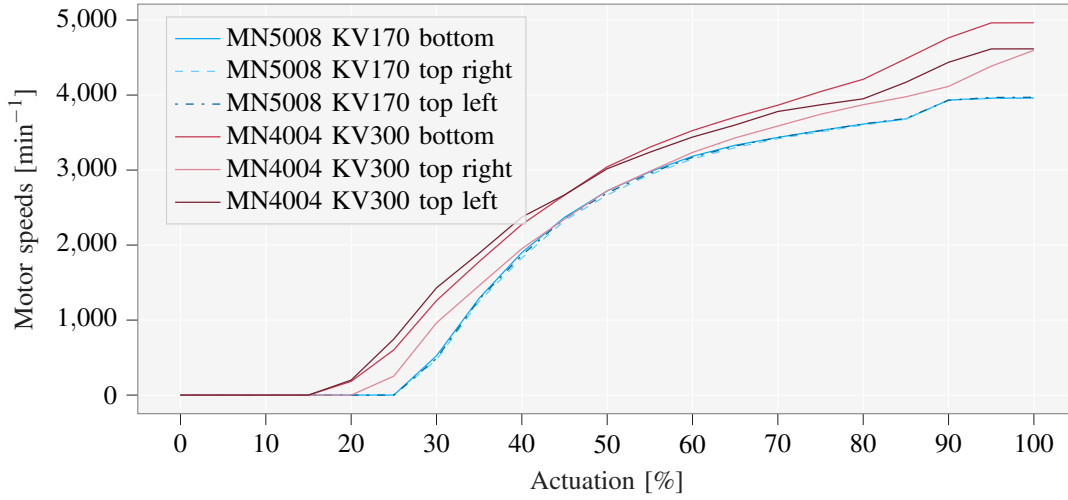
Fig. 6: Motor characteristic curves of each evaluated motors with the available actuation as input and the turning speed as output. The MN5008 show similar curves for each position, while the curves of the MN4004 have significant differences in behavior.
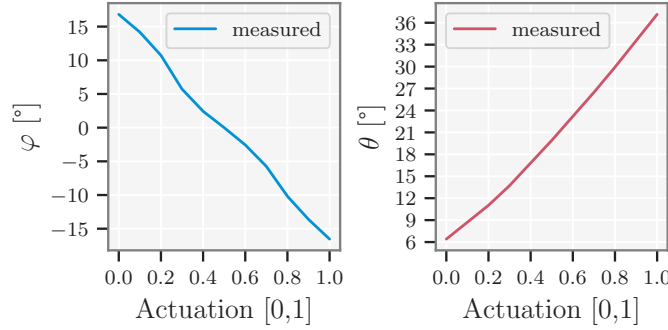

Fig. 7: Characteristics of the orientation actuators for different actuation signals.

- **Setup:** For reducing discrepancies between the different structures of the two configurations of the custom designed ball launcher and the TTmatic 303A, similar altitude angles are chosen for evaluation and specified with the terms "low", "medium" and "high". The ball launchers are positioned directly at the front-end of the table. The TTmatic 303A is not capable of launching the small distance to Target 1 with automatic ball feed. Therefore, the distance of the TTmatic 303A to the front-end of the table had to be increased by approximately 30 cm. For target shooting, the data set has been recorded with AIMY in the second configuration with a distance of 80 cm to the front-end of the table. With increased distance higher speeds and spins can be utilized, however, with a compromise on accuracy.

The provided data sets have been processed before the – in Section IV described – bounce detection is applied. The processing includes several filter processes addressing either faulty samples or faulty trajectories:

- A time jump filter removes a trajectory from the data set, if there exists a time gap between two consecutive samples exceeding 0.5 s.
- A rebound filter removes a trajectory from the data set, if there is no rebound on the table. However, trajectories closely missing the table edges provide important information for target shooting learning. Therefore, the data sets have been exported with rebound filter applied and without.
- The position jump filter removes samples with position deviations of more than 5 cm between one sample and the neighborhood of 15 closest samples. This process proved to remove most outliers caused by the ball tracking system.
- The region filter removes all samples outside the table area, with some relaxation in x- and y-direction.

Additionally, all samples have been transformed to fit the defined initial coordinate system, with x increasing along the long edge of the table and y along the short edge of the table. We set the origin at the frontend of the table in the middle of the table. The time stamps are converted from nanoseconds to seconds. In addition to jump detection, we inspected all trajectories visually for rough outliers to ensure plausible training data and evaluation results.

D. Evaluation of Orientation Jumps

Additionally to the system parameters assessed in Section IV, we evaluated the influence of orientation jumps on the ball trajectories. Orientation jumps are significant changes in altitude and azimuthal angles of the launching tube . For evaluation, we recorded ball trajectories in which the orientation was not changed beforehand and launches in which an orientation jump was made beforehand. An altitude angle of 19.9° and an azimuthal angle of 0° define the default launching position of the orientation actuators. Before each shot, the table tennis launcher moves into the neutral position from a displaced position with an altitude angle of 6.4° and an azimuthal angle of −15.8°. Table IV and Figure 8 provide the results. The figure shows no significant deviations of the landing points for trajectories recorded with orientation jumps in comparison to trajectories without orientation jumps.

TABLE IV: Investigation on the influence of orientation jumps on the landing points accuarcy

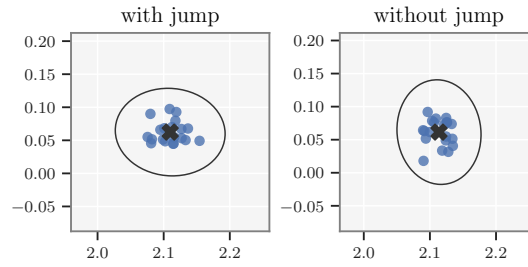| Orientation jump	Number samples [-]	σ_x [mm]	σ_y [mm]
no	20	**0.0155**	0.0193
yes	20	0.0195	**0.0176**

Fig. 8: Orientation jump experiment. Each measurement point represents a landing point of a ball trajectory. The dimension of the confidence ellipse is roughly equal between the measurements with and without jump. However, the orientation of the larger deviation switched between the measurement series.

E. Evaluation of Stroke Gain

Table V lists the standard deviation values in each direction of the table plane of Figure 3 for different stroke gain values. Stroke gain is briefly defined in Section IV-B. Figure 9 further illustrates the deviation of the landing points.

TABLE V: Investigation on the influence of stroke gain on the landing points accuarcy

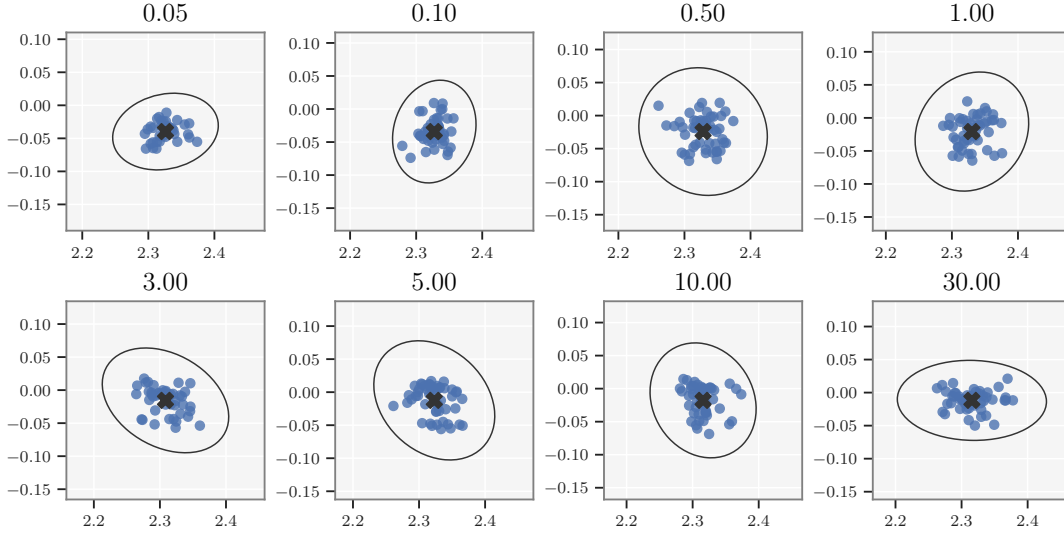| Stroke gain [-]	Number samples	σ_x [mm]	σ_y [mm]	A_σ [mm^2]	Launch time [s]
0.05	35	0.0197	**0.0143**	**0.00088**	8.83
0.10	50	**0.0156**	0.0192	0.00095	4.21
0.50	50	0.0241	0.0239	0.00181	1.60
1.00	49	0.0213	0.0223	0.00149	1.39
3.00	46	0.0237	0.0196	0.00146	0.81
5.00	50	0.0227	0.0223	0.00159	0.61
10.00	50	0.0199	0.0215	0.00134	0.67
30.00	50	0.0279	0.0150	0.00132	0.62

Fig. 9: Stroke gain experiment. Landing points are illustrated for different stroke gain values along the confidence ellipses of three times the standard deviation, and the mean value given as a cross. One can observe only minor differences of the deviations between each stroke gain values.

F. Evaluation of Ball Pinching

Table VI lists the standard deviation values in each direction of the table plane of Figure 3 for different ball pinching values. Section IV-B gives a brief definition of ball pinching. Figure 10 further illustrates the deviations of the landing points.

TABLE VI: Investigation on the influence of ball pinching on the landing points accuarcy

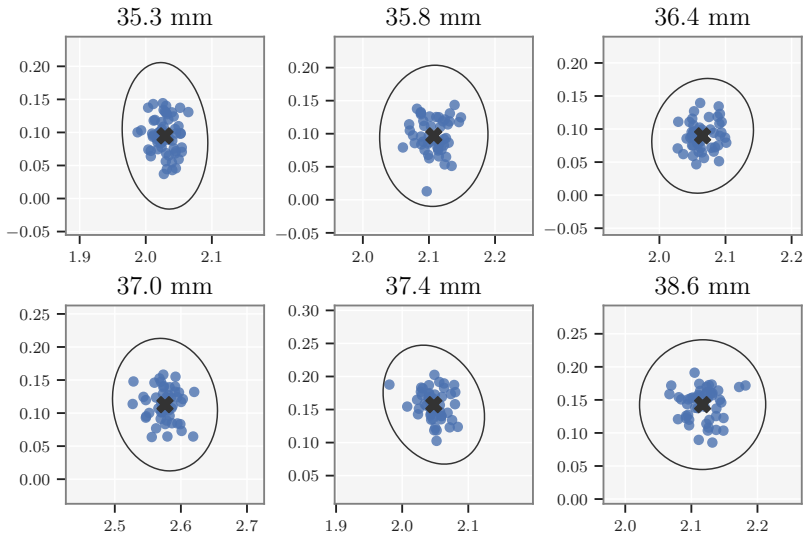| Pinching diameter [mm]	Number samples [-]	σ_x [mm]	σ_y [mm]	A_σ [mm^2]
35.3	51	0.03194	0.03062	0.00307
35.8	48	0.01932	0.02658	0.00161
36.4	49	0.01906	**0.02171**	0.00130
37.0	49	0.01965	0.02480	0.00153
37.4	49	**0.01866**	0.02208	**0.00129**
38.6	49	0.02361	0.02428	0.00180

Fig. 10: Ball pinching experiments. Landing points are illustrated for different ball pinching values along confidence ellipses of three times the standard deviation, and the mean value given as a cross. It can be seen that small and high pinching results in slightly increased deviations.

G. Evaluation of Motor Ramp-up

Table VII lists the standard deviation values in each direction of the table plane of Figure 3 for different ramp-up times. Section IV-B gives a brief definition of the Ramp-up time. Figure 11 further illustrates the deviations. The motors need a certain time to reach the requested speed. However, under 0.5 seconds of ramp-up time, the motor speeds are not yet converged at the time of launching. This noisy wheel velocities result in increased deviations of landing points.

TABLE VII: Ramp-up times accuracy experiments

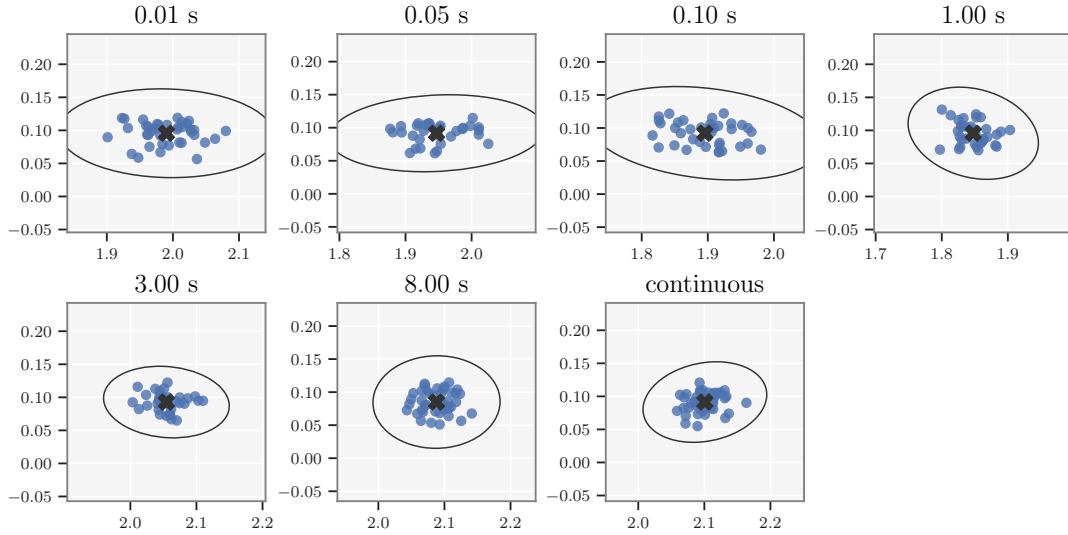| Ramp-up time [s]	Number samples [-]	σ_x [mm]	σ_y [mm]	A_σ [mm^2]
0.01	39	0.0402	0.0166	0.002096
0.05	32	0.0400	0.0143	0.001797
0.10	38	0.0436	0.0174	0.002383
0.50	34	**0.0211**	0.0150	0.000994
1.00	34	0.0243	0.0172	0.001313
2.00	33	0.0243	**0.0134**	0.001023
3.00	36	0.0234	**0.0134**	**0.000985**
8.00	40	0.0247	0.0173	0.001342
continuous	40	0.0231	0.0150	0.001089

Fig. 11: Ramp-up time experiments. Landing points are illustrated for different ramp-up times between actuation and launch of the ball along confidence ellipses of three times the standard deviation, and the mean value given as a cross. It can be seen that low ramp-up times result in significantly increased deviations in x-direction.

H. Evaluation of Versatility

Table VIII lists the experimentally determined maximum ball speed and topspin of the evaluated ball launchers. AIMY with the T–Motor N5008 KV170 motors is capable of generating the highest ball speeds and spins.

TABLE VIII: Comparison of maximum ball speeds and topspins of the evaluated ball launchers

	Max Speed	Max Topspin
Professional Human (average)	16.8 m s^{-1}	113.9 s^{-1}
TTmatic 303A	9.8 m s^{-1}	64.0 s^{-1}
Donic Robo-Pong 2050	10.7 m s^{-1}	106.7 s^{-1}
AIMY (MN4004 KV300)	10.8 m s^{-1}	− s^{-1}
AIMY (MN5008 KV170)	15.4 m s^{-1}	192.0 s^{-1}

I. Evaluation of High-fidelty

In Figure 4, the accuracy of AIMY was compared to the TTmatic 303A and Donic Robo-Pong 2050 for four distances with the medium altitude setting. Table IX lists the angle values for each altitude setting. Figure 12 extends Figure 4 with low and high altitude angles.

TABLE IX: Altitude angles of evaluated ball launchers

	Low Angle	Medium Angle	High Angle
TTmatic 303A	Slide with smallest curvature	Slide with medium curvature	Slide with largest curvature
Donic Robo-Pong 2050	8°	26°	30°
AIMY MN4004 KV300	14°	26°	28°
AIMY MN5008 KV170	6.4°	19.9°	37.1°

J. Data Set

For illustration purposes, Figure 13 shows 415 of the 3761 recorded trajectories for training after preprocessing. The landing points of the ball trajectories of the training data set are shown in Figure 14. The ball tracking system described in Section IV records with 200 Hz. Figure 15 depicts a histogram of the time stamp differences of consecutive samples to verify the equidistance of the recorded samples. We assume, that computation in the ball detection algorithm leads to slightly reduced recording frequency.
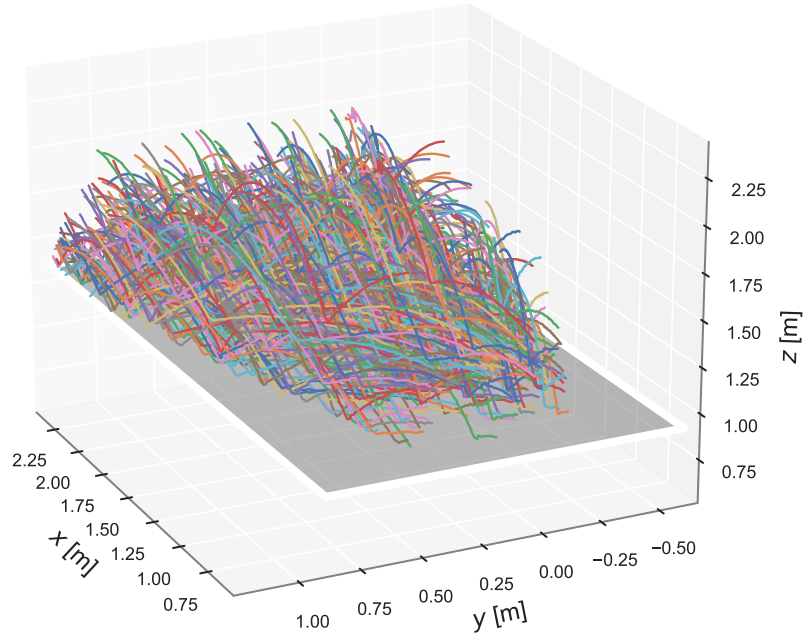

Fig. 13: Ball trajectories from recorded training data set in three-dimensions. Grey plane illustrates the table tennis table. Ball trajectories start from the frontend of the table and cover the other side of the table, cone-shaped.
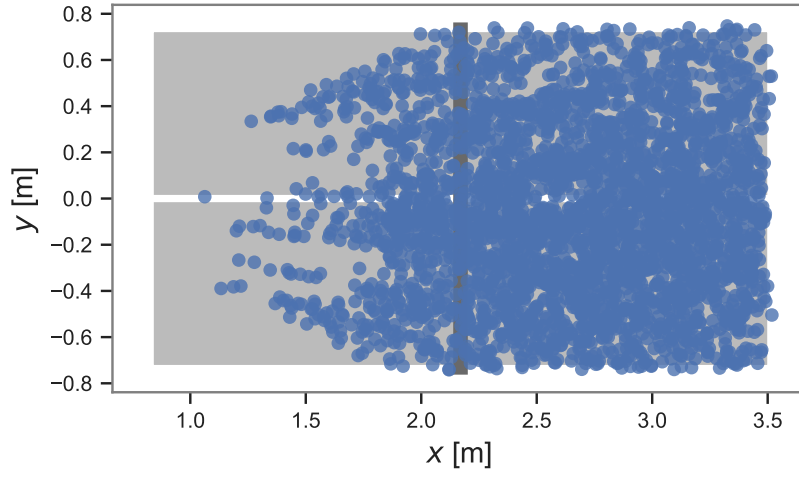
Fig. 14: Landing points of the ball trajectories of the training data set. Grey plane illustrates the table tennis table. The opposite side of the table has a high coverage of landing points.
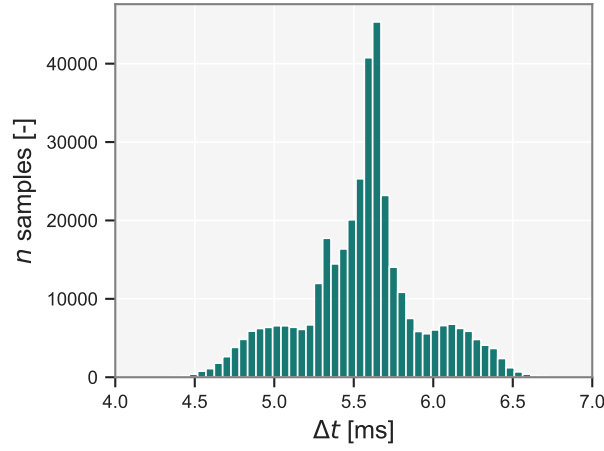

Fig. 15: Histogram of the time delta between each recorded sample. The time differences are located between 4.5 and 6.5 ms. The majority of the time differences are between approximately 5.25 and 5.75 ms.

K. Target Shooting

For learning to hit landing position, we trained a feed-forward neural network, as described in Section V-B. For training, we consider each position measurement of the presented data set and the corresponding control parameters as a training sample. All training samples after the rebound are not considered. For target shooting, we simplified the problem by only considering trajectories shot with equal throwing wheel speeds. This simplification reduced the control parameters to the two orientation angles and one throwing wheel speed applied to all wheels, and herewith the dimensionality of the action space. Therefore, the target shooting network learns the position to control parameter mapping with a total of 66581 training samples from 415 recorded trajectories. Table X lists the specifications of the architecture of the neural network.

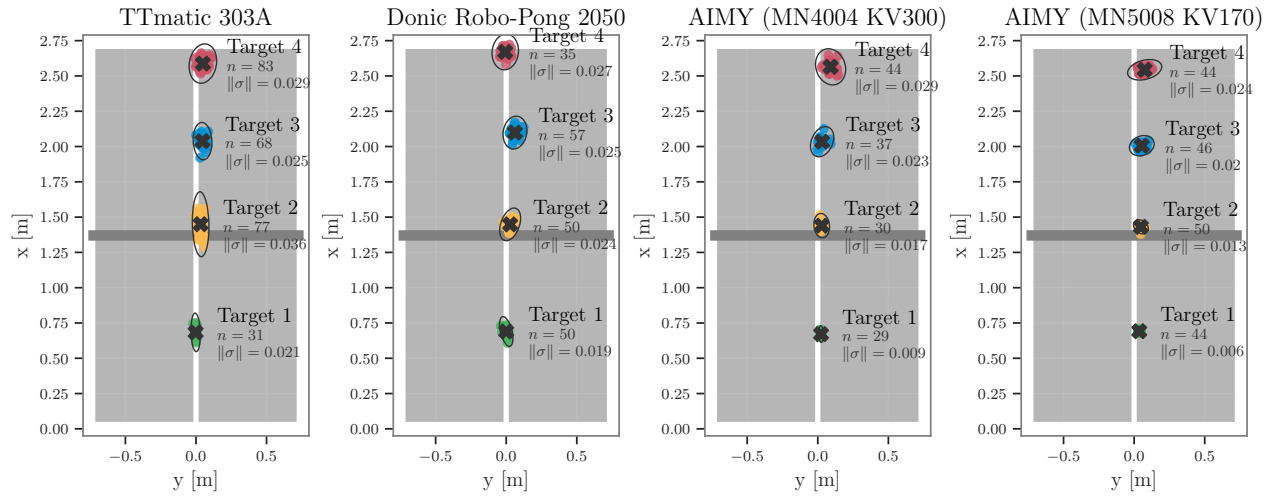TABLE X: Specification of the target shooting network

Specification	Value
Number of Layers	3
Neurons per Layer	2048, 512, 128
Loss	MSE
Activation	sigmoid
Dropout	0.1
Optimizer	ADAM
Epochs	1400
Mini batch size	4096
Minimum learning rate	1e-4
Normalization	standard
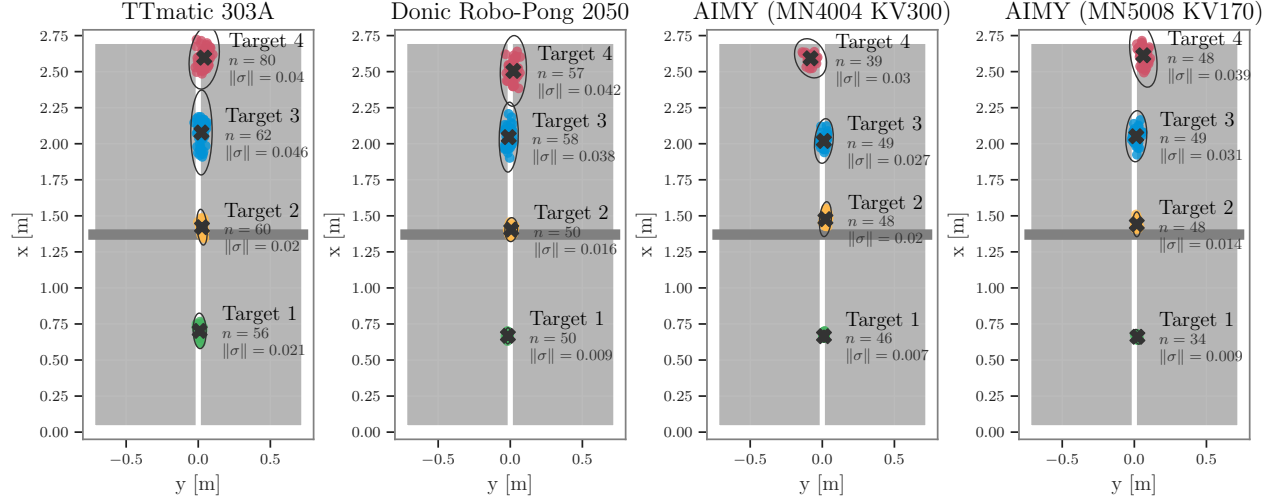
(a) Distance accuracy with low altitude angle
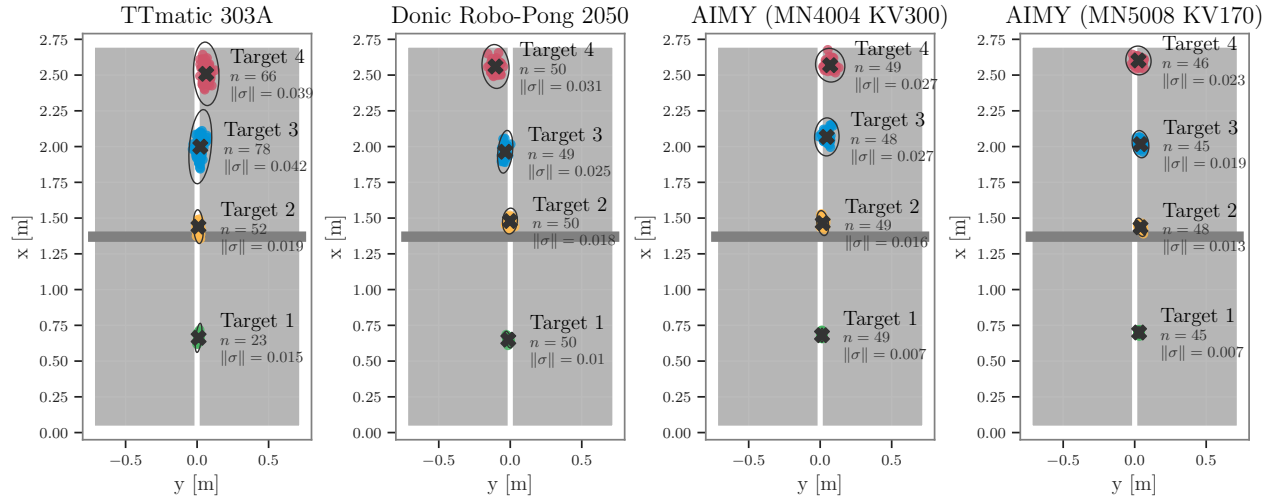
(b) Distance accuracy with medium altitude angle

(c) Distance accuracy with high altitude angle

Fig. 12: Distance accuracy experiment for low, medium, and high altitude angles.