KIKS Extended Team Description for RoboCup 2023

Daichi Miyajima, Kosei Naito, Hayato Mitsuda, Kazuaki Harada, Mizuki Nonoyama, Ryo Shirai, Futa Sato, Ryuto Tanaka, Yota Dori, and Toko Sugiura[1]

National Institute of Technology, Toyota College, 2-1 Eisei-cho, Toyota, Aichi 471-8525, Japan
sugiura.toko@toyota.kosen-ac.jp,
URL: www.ee.toyota-ct.ac.jp/~sugi/RoboCup.html

Abstract. This paper presents the robot and software system of SSL team KIKS, which is planned to participate in the RoboCup 2023 Bordeaux. In this ETDP, we mainly present our study on the improvement of the dribbling bar, dribbling mechanism and kicker circuit, and prediction of the opponent robot's behavior using machine learning. We describe the weaknesses of the previous generation robot and the improvements to overcome them, including software upgrades. In addition, we overview the study of improving robot position control with on-board cameras and encoders to correspond to Vision-Blackout.

Keywords: RoboCup, small size league, autonomous robot, global vision, engineering education

1 Introduction

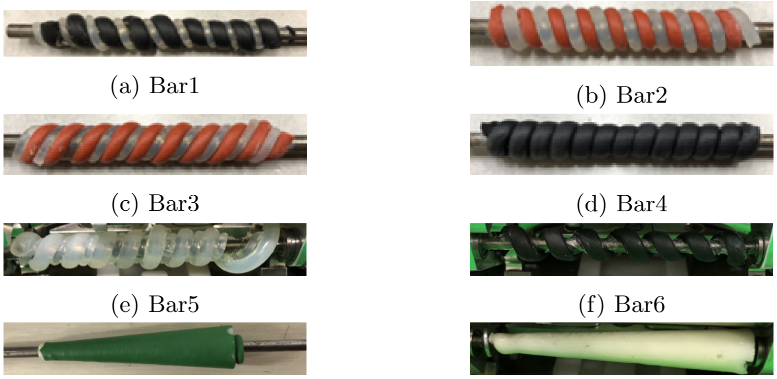
Team KIKS has continued to work toward developing higher performance hardware and smarter AI systems. This year, we studied the improvement and extension of ball control performance on hardware, especially on the improvement of the dribbling system. We also experimented with motion control based on the robot's own judgment, using a local vision system and encoders corresponding to Vision Blackout, based on the Jetson Nano introduced last year. In the software, we tried to predict the opponent robot's behavior using machine learning. The results of the experiments are described below.

2 Mechanical system

In RoboCup Small Size League, the dribbling system is very important to realize the strategy of AI, recently. This is because enhanced ball possession and the realization of curved shots lead to an expansion of the strategy. In 2022, we improved the traditional dribbling bar to increase ball keeping ability by simply splitting in the center of the bar[1]. The improvement was very simple and could be done in a very short time. As a result, the ball keeping ability of the robot was improved, but its performance is still not sufficient and several problems remain to be solved. In this section, we describe the trial experiments we performed to solve those problems and the results of the verification.

2.1 Dribbling bar

The dribble bar [1] proposed in the ETDP 2022 is made by cutting a rubber pipe of uniform thickness. Due to the relatively hard material, when the ball was caught at the end of the bar, it bounced and could not hold the ball. In addition, the ball could not be moved to the center of the bar, and sufficient rotational and holding force could not be given to the ball. If the ball could be moved to the center of the dribbling bar in a short time by improving the shape and material of the dribbling mechanism and bar, it would be effective in enhancing the ball holding force and expanding the variety of strategies. A typical dribbler that moves the ball to the center is a spiral structure [2], [3]. We introduced it in 2020, but its effectiveness was not sufficient due to the individual robot differences and spiral accuracy of the robots, and it was not applied to all robots because of its high fabrication difficulty. Therefore, we aim to develop a dribble bar that is easier to fabricate than the spiral structure and more effective in moving the dribble to the center. In this section, we focus on the material and shape of the dribble bar, and make prototypes of different types of dribble bars. The purpose of this section is to experimentally evaluate the lateral movement performance and ball possession of the ball in contact with the dribble bar, and to provide guidelines for the development of more effective dribble bars.


(a) Bar1

(b) Bar2

(c) Bar3

(d) Bar4

(e) Bar5

(f) Bar6

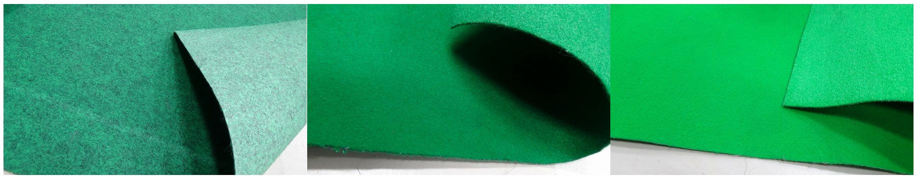
(g) Bar7

(h) Bar8 (Super soft gel)

Fig. 1. Dribbling bar used in the experiment

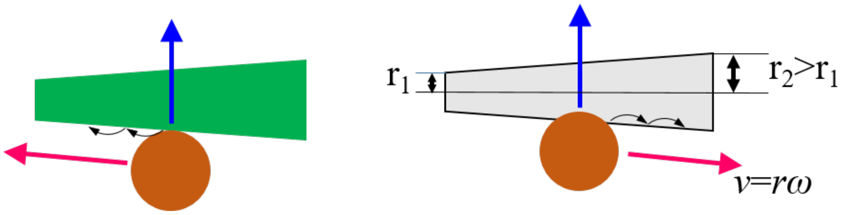
Figure.1(a)-(h) show the newly designed and fabricated dribble bars. (a)-(f) show a metal bar as a core around which a string of rubber or plastic is wrapped,

(g) shows a PLA cone formed by a 3D printer and covered with a balloon rubber to enhance friction, and (h) shows a conical mold created by a 3D printer and then molded with a flexible material called ”human skin gel”. (h) is a conical mold made by a 3D printer and molded with a flexible material called HITOHADA gel. HITOHADA gel is the super soft urethane resins for molding manufactured by EXSEAL co. ltd[5]. Its hardness is ASKER C/15 just like human skin. As a result of the actual experiment using a ball, only three types of horizontal movement on the dribble bar could be confirmed. Therefore, to evaluate the effectiveness of the three types of dribble bars (f)-(h), we measured the velocity of the ball moving horizontally along the dribble bar. Three types of carpets were used in the experiment. These pictures are shown in Fig. 2. The material of all carpets is 100% polypropylene. Carpet1 shown in Fig.2(a) is used in our playing-field. The surface structures of (a) Carpet1 and (b) Carpet2 shown in Fig.2 are similar and relatively soft. (c) Carpet3, on the other hand, has rougher and harder. There is no significant difference in terms of friction. The results are summarized in Table 1. The results show that there is no significant difference in the speed of ball movement by the dribble bars. However, the durability of the balloon rubber of Bar7 is poor, indicating that it is not suitable for practical use. In addition, the direction of horizontal movement of the ball was opposite for both Bar7 and Bar8, even though they have the same cone shape. That is, in Fig.3, where the dribbling bar and the ball are viewed from above the robot, (A) Case of Bar7, the ball moved with a slight bounce toward the top of the cone, and (B) Case of Bar8, the ball moved smoothly in the opposite direction. The reason for this can be attributed to the difference in the contact area between the ball and the dribble bar due to the difference in the surface hardness (viscosity) of the materials. That is, the surface of Bar 7 is relatively hard, and because the ball makes contact at a point, it bounces and moves gradually to the left. On the other hand, Bar 8 is made of a softer material, which is thought to increase the size of the contact area as the ball dips into the dribble bar, pulling the ball toward the right with a larger cone radius (higher rotation speed). It was reconfirmed that the material (hardness) of the dribble bar is one of the most important factors in choosing an appropriate dribble bar.



(a) Carpet1 (b) Carpet2 (c) Carpet3

Fig. 2. Carpets used in the experiment for a ball-holding performance



(a) in case of Bar7

(b) in case of Bar8

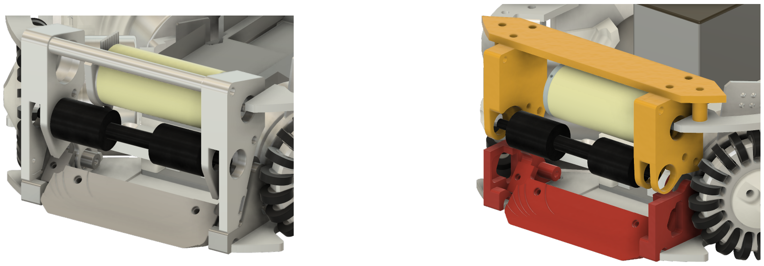
Fig. 3. Direction of ball movement for different bar materials

Table 1. Ball speed moving horizontally along dribble bar[m/s]

Dribble bar\Materials	Carpet1	Carpet2	Carpet3
Bar6	0.1	0.09	0.11
Bar7	0.09	0.11	0.12
Bar8	0.1	0.1	0.11

2.2 Tentative dribbling device

One of the problems with our dribblers is their low maintenance. Current dribblers consist of a ball sensor, dribbling mechanism, and tip kick bar in a single component, and improving any of them would require redesigning all parts. In this section, we evaluate the performance of a tentative dribbling device with improved maintainability by dividing the dribbling peripheral mechanism into a dribbling mechanism and a ball sensor & tip kick mechanism.



(a)Present

(b)Tentative without damper

Fig. 4. Dribbling device

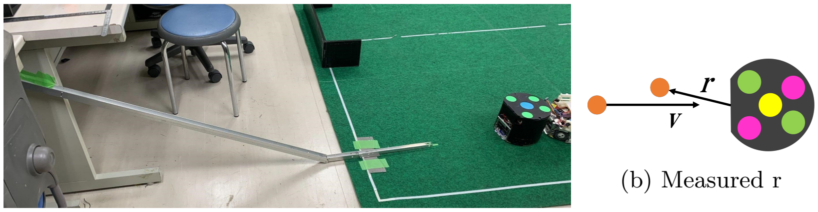
Figure 4(a) shows the current dribbling device, and Figure4(b) shows the dribbling device fabricated tentatively, respectively. The dribbling motor is a maxon

#283860 with a gear ratio of 1.4. The height from the carpet to the metal shaft with 4mm diameter of the dribbler is about 36 mm. While the current dribbler has all elements fixed on the base plate, in trial dribbler, it is divided into the components as units. Yellow part in Fig. 4(b) shows the dribbling unit, red part shows the ball sensor & tip kick unit, respectively. On the other hand, instead of improving maintainability, this dribbler has removed the damper unit from the current dribbler. If the dribbling bar made of super soft gel shown in Fig. 1h can sufficiently play the role of a damper unit, the dribbling mechanism can be simplified. We verified the shock absorption and ball retention performance of the dribbler with and without the bumper in the experiments.

Verification of Shock Absorption Performance of Tentative dribbler Conventionally, it is quite common to mount a damper on a dribbler. It is interesting to note that the ETDP2022 from Tigers Mannheim introduces a dribbler equipped with a 2 degree of freedom damper[6]. On the other hand, our recent damper unit has a problem that the dribbling bar bounces the ball during dribbling, so we removed the damper unit for trial and compared its behavior with two types of dribbling bars. One is made of conventional polyurethane rubber and the other is made of newly fabricated super soft gel. By comparing these two types, the possibility of simplifying the dribbling unit is investigated.

The following experiment was carried out to compare the shock-absorbing performance of two dribblers. A ball moving at a constant speed (about 2.3 ms^{-1}) was impacted head-on with a dribbler that was not rotating. The distance r between the robot and the bouncing ball was measured 50 times, the average was calculated, and compared for each dribbler. The ball was ejected using the slope shown in Fig.5(a), and the robot was fixed on the field. The measured distances are shown in Table 2.

Table 2 shows that the bounce distance of SUPER SOFT GEL is longer than that of the current dribble bar, with or without damper. This result indicates that the shock-absorbing ability of the super soft gel only is insufficient. As a result, it was shown that the tentative dribbling device has a lower shock absorption ability than the present device due to its structure without a damper. Namely, it was again confirmed that dampers are important to improve the performance of shock absorption.



(a) Environment

(b) Measured r

Fig. 5. Experimental condition

Table 2. Evaluation of shock absorption performance for dribbler

Dribbling bar	Distance between robot and stopped ball [mm]
Current rubber with Damper	379
Current rubber without Damper	600
Super soft gel with Damper	463
Super soft gel without Damper	819

Verification of ball keeping performance In real games, performance in possession of the ball (e.g., ball placement, preliminary movements before shooting, etc.) is critical. If fast and stable movement could be achieved while keeping possession of the ball, AI's tactics would be greatly expanded. The following experiment was carried out to compare the ball keeping performance of the dribbler described in the previous section. While the robot was keeping the ball, the robot was allowed to rotate until it released the ball at a constant angular velocity. The ball keeping time against the angular velocity of rotation was measured and compared between the present dribbler and the proposed bar used with super soft gel shown in Fig.6. The speed of the dribbling motor was about 100^{-1}(6300rpm).

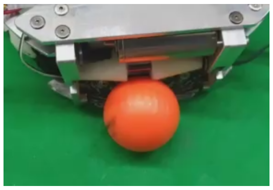


Fig. 6. Dribbling bar made from super soft gel

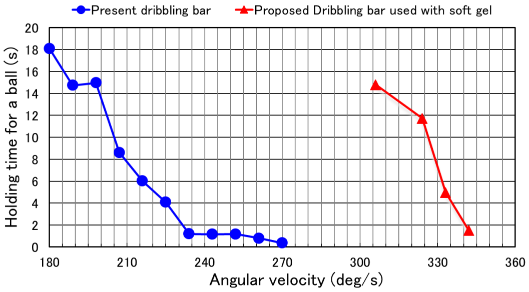


Fig. 7. Ball keeping performance

The results are shown in Fig.7. The horizontal axis represents the angular velocity of the robot's rotation, and the vertical axis represents the average keeping time. The results of Fig.7 show that the current dribbler cannot keep the ball stably when the robot's angular velocity is 200 deg/s or higher. On the other hand, the dribbler with super soft gel could keep the ball stably for more than 10 seconds if the angular velocity was less than 300 deg/s. However, contrary to our prediction, the super soft gel dribblers kept the ball at the end of the bar. This indicates that the keeping force of the super soft gel is due to the softness of the material. That is, the contact area with the ball is larger, which may be the result of sufficient frictional force transferring the dribbler's rotational force to the ball. In this section, we describe the shock absorption and ball keeping performance of a prototype dribbler. The results showed the importance of the damper and the usefulness of the new material (super soft gel). In the future, the durability of the super soft gel should be investigated in detail.

3 Electrical system

The main circuit was modified significantly last year. This year, we are continuing to use that circuit. In this section, especially, we describe a new circuit board for kicker device.

3.1 Introducing a New Kicker Circuit Using the Two-Step Booster Method

KIKS has conventionally used a one-stage voltage booster circuit as shown in Fig.8 as a kicker circuit to operate the solenoid. In this circuit, a reverse voltage equal to the voltage of the capacitor for the solenoid operation is instantially applied to the rectifier diode during the operation.
Recently, in SSL, the power of the kick and continuous operation have been required, and it has been necessary to increase the capacitor voltage when charging the device. Since our previous charging method applied a large load to the device, there were concerns about malfunctions during a game and a decrease in the circuit life. Therefore, we designed and introduced a new kicker circuit using a two-step boost method. It can be applied to the new main board introduced in ETDP[1] in 2022. The effectiveness of the circuit is described below.

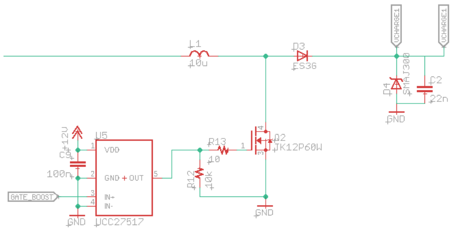


Fig. 8. previous kicker circuit for voltage booster

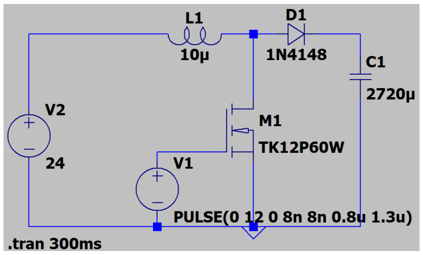


Fig. 9. LTSpice circuit for previous voltage booster

Figure 9 and 10 show the simulator schematics of the previous kicker circuit and the voltage booster part of the new kicker circuit by LTspice XVII, respectively. The results of the transient analysis are shown in Fig. 11 and 12, respectively. In the previous kicker circuit in Fig. 11, a maximum reverse voltage of about 70 V is applied to rectifier diode D1 300 ms after the start of charging. On the other hand, in the new kicker circuit in Fig. 12, the reverse voltage applied to rectifier diode D4 is suppressed to about 52 V (Max 70 V - Min 18 V). This means that the voltage applied to the device is reduced by 3/4 of that in the previous kicker circuit. Therefore, the improvement to the 2-stage voltage booster circuit is considered to contribute to the improvement of the durability of the circuit.

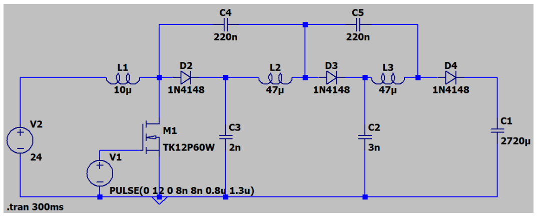
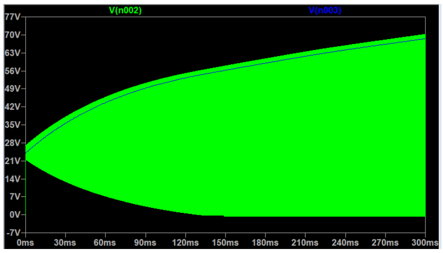


Fig. 10. LTSpice circuit for new voltage booster



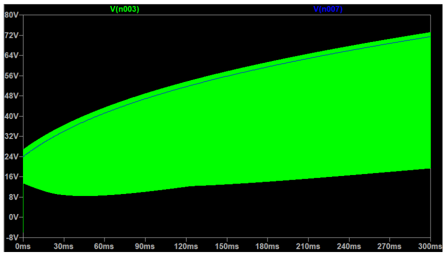


Fig. 11. Waveforms of C1 terminal voltage (green) and D1 anode voltage (blue) in previous kicker circuit

Fig. 12. Waveforms of C1 terminal voltage (green) and D4 anode voltage (blue) in new kicker circuit

4 Software system

4.1 Prediction of actions using machine learning

Predicting the actions of the opposing robot is one of the major factors that dominate soccer tactics, as it is useful in getting the ball and determining effective positioning. Our team, in particular, has the problem of low ball control during the game. To solve this problem, we tried to predict which robot the opponent robot will pass to next based on information from the field. In this section, we consider this problem as a classification problem of the opponent's behavior and analyze it using machine learning. We defined a total of 12 classes:

10 candidate classes for passing, and 2 classes for not passing (dribbling) and shooting.

Models and Data Structures In order to effectively learn information on a field, image classification methods are considered suitable. Therefore, we used one of them, ResNet (Residual Neural Networks)[7].
The input to ResNet requires a matrix representation of the field information. Since in the field of image classification, convolution is performed by superimposing two-dimensional images of each RGB and convolving them as three-dimensional, we divided the field information into multiple dimensions and performed convolution. Here, the ball and each team's robot are represented as another 120×120 matrix, each corresponding to a position on the field. For the team being trained, the value of the robot's angle and speed were similarly placed in the corresponding elements of the matrix, resulting in five matrices of field information.
ResNet is one of the CNNs (Convolutional Neural Networks), but it is generally considered unable to learn the positional relationships in an image. Therefore, to solve this problem, we added matrices representing *x-axis* and *y-axis* as field information in addition to the five matrices mentioned above, as proposed by Rosanne Liu et al.[8]. That is, a 7×120×120 matrix was used as input information to the model. The IDs used as output were assigned in *x-axis* in ascending order to ensure uniqueness of IDs across data. The above dataset was used as the input to the model when the robot of the team being trained kept the ball. Then, the ID of the robot with the ball at the end of one play was used as the output of the model to create the teacher data. We played our AIs against each other on the GRSim simulator and generated data for 149 pass plays. Using those data as training data, training and inference were performed on the training and evaluation data, respectively.
The results are shown in Table 3. The results show that inference was not sufficient for the evaluation data. On the other hand, for the training data, it shows an accuracy of 80%, suggesting that the task is classifiable. Table 4 also shows the results of inference on the same training data under different conditions. From the table, it can be seen that the amplitude of velocity has a significant

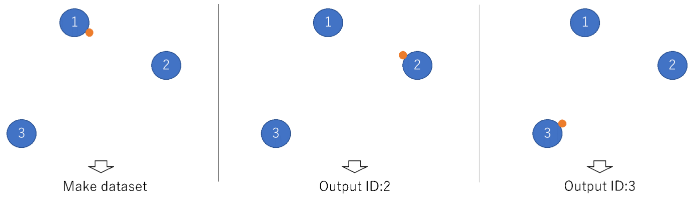


Fig. 13. Method for making dataset

Table 3. Probability predicted by ResNet(149 samples)

Data \	Accuracy	Avg.Precision	Avg.Recall	Avg.F-measure
Learned data	0.8055	0.7404	0.6251	0.6527
Untrained data	0.2564	0.0751	0.1138	0.0895

effect on the accuracy of inference, while the angle parameter contributes very little to inference. This means that it is almost meaningless in the data due to the fact that most robots are facing the ball direction. On the other hand, it can be seen that it works effectively with regard to the location matrix. However, when we experimented with increasing the number of data, we observed a decrease in accuracy. As a proposal for improvement to increase accuracy, we will consider using images or time-series data as input data. Md Amirul Islam et al.[9] demonstrate that the parameters of the Convolution layer can also be used to store location information, and we would like to try this out. In addition, in this section, we used inputs that represent coordinates with reference to CoordConv, but *xy-axis* should originally be incorporated into the Convolution layer, and we will consider its implementation. After the completion of this model, we will apply it to actual matches and consider applications such as countermeasures for each opponent, or diversion of the opponent's passing algorithm to our team like imitation learning.

Table 4. Probability predicted by ResNet under the other condition

Data \	default	no speed, norm	no angle	no speed, norm, angle	no position matrix
Accuracy	0.8055	0.2592	0.8148	0.3611	0.6944

4.2 Verification of the effectiveness of position control performed by the robot itself

In this section, we try to improve the motion performance of the robot by controlling the position of the robot itself.

Problems with Global Vision Only Control At present, we are using global vision for position control based only on information from the server side. However, this method has the following problems.

- If the communication delay from the server to the robot is large, the robot oscillates back and forth due to the difference in the control cycles between the server and the robot.

- The robot cannot be controlled if there is a vision problem (e.g., the ball is in the robot's shadow and precise position information is not obtained) or if the communication is interrupted.

We tried to mount local vision on a robot in 2022 and reported on the utility of tracking the ball with local vision(Fig. 14) only and the control to support global vision[1]. In this section, we investigate the effectiveness of using an encoder attached to the motor in addition to local vision. In the following experiments, we try to control the robot's position using only an encoder.

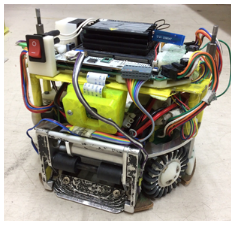
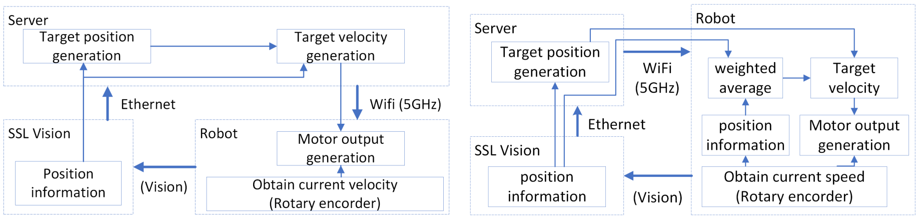


Fig. 14. Local camera mounted on the robot

System overview Present system and trial system built on tentatively are shown in Fig.15. The current system uses only information from the server side to generate velocity to the target position, but in this experiment, control is performed only with the robot's encoder. If the precision of the position information from the local vision and encoder improves, it is expected that the robot's position control can be improved by adding it to the information from SSL vision.



(a) Present system (b) Trial system

Fig. 15. Motion contol system

Experimental As shown in Fig.16, the robot moved 1000 mm in a straight line and rotated 90° at the same time, as commanded by the server, and measured the time from the start to the stop of the action. The time was compared between the case with an Ethernet connection (corresponding to an environment with good communication conditions) and the case with a Wii connection (corresponding to an environment with poor communication conditions). Table 5 summarizes the experimental results. In the case of the Ethernet connection to the server, there was almost no difference in the process time between the current sytem and the trial system. On the other hand, when connected to the server via Wifi, the current system did not finish its operation for more than 10 seconds, whereas the trial system completed its operation in about 2 seconds, as in the case of the Ethernet connection. This result suggests that there is a problem with the stability of wireless communication in the current system and confirms the effectiveness of the trial system.

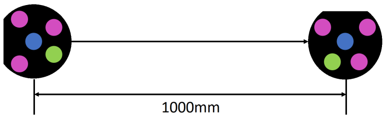


Fig. 16. Experimental method for motion control performance

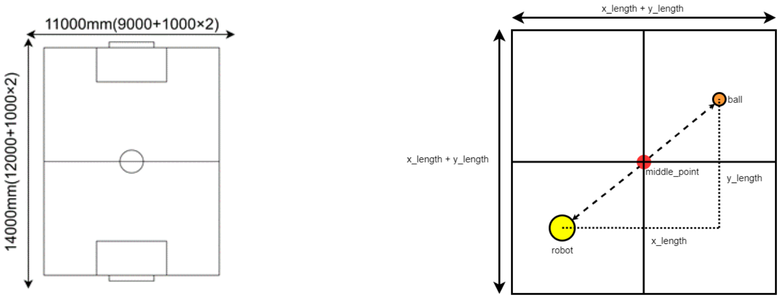
Table 5. Verification results (10 times average)

Process time for \ Connection	Ethernet [ms]	Wifi [ms]
Ping between server and robot	6.7	11.5
Ping between server and vision	2.2	6.1
Present system	2159	>10000
Trial system	2060	1987

4.3 Trial experimetal in path planning based on Informed RRT*

At present, KIKS uses the Human-Like algorithm as the robot's path planning method. While the Human-Like algorithm is simple and fast, it sometimes fails to generate an optimal path when the path is blocked by obstacles or under some conditions.。 Therefore, in this section, we aim to introduce path planning based on Informed RRT*[10] and verify its effectiveness.

Experimental method In this experiment, the random place to place a point to generate the first path is somewhere in the range 14000 mm×11000 mm with a margin of 1000 mm around a field of 12000 mm×9000 mm, as shown in Fig. 17a.



(a) Area of points placed randomly at the beginning

(b) Limited area of points placed randomly

Fig. 17. Area where random points are placed

After the first path is generated, as shown in Fig. 17b, a square is created centered at the midpoint between the distances of the robot and the ball, and random points are placed within this range. After the first path is generated, as shown in Fig. b, a square is created centered at the midpoint between the distances of the robot and the ball, and random points are placed within this range. Note that the length of one side of this square in the first stage should be double the length of the x- and y-components of the distance between the robot and the ball added together. If the path can be generated again by placing a number of random points, the coefficient applied to the length of the square is slightly reduced to narrow the range, and then a random point is placed in the same way. The coefficient in this case is written as the $expand\ coefficient$, and the change in this value is given by eq. (1).

$$expand\ coefficient = e^{-0.05x} + 1 \tag{1}$$

Note that x is the number of times the path is generated. The length of one side of the square in the range where the final random point is placed converges to the sum of the lengths of the x and y components of the distance between the robot and the ball, as in the notation in Fig. 17b. In this case, after 100 random points are placed, the path with the shortest distance, generated by the range reduction described above, is taken as the optimal path. The reason why we did not specify the range with an ellipse, which is typically used in Informed RRT*, is that we are concerned about a decrease in process speed when operating path planning in real-time at high speed. In this experiment, the minimum length of one side of the square for range limitation is 2000 mm. Thus, the distribution range of random points is reduced to a minimum of about 2.60%(=$(2000)^2/(11000\times14000)$) of the starting point for the search.

Results and discusson Place the ally robot (yellow), the enemy robot (blue), and the ball (orange) as shown in Fig. 18 on the simulator. One of the three enemy robots is placed in the corner of the penalty area, which is a prohibited area, and the other enemy robots are placed at a distance of 100 mm from the other robots. This is to verify that the robot does not try to force its way through a space smaller than the size of the robot and collide with an enemy robot. The ball and the allly robot should be placed 900 mm from the row of enemy robots, and the ball and the allly robot should be placed so that the straight line connecting them perpendicularly crosses the row of enemy robots.

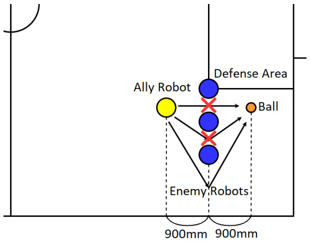
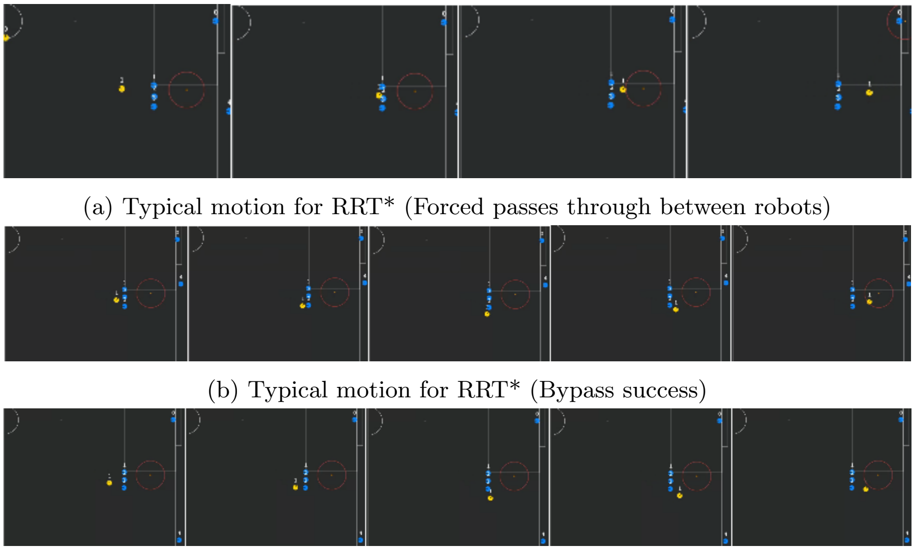


Fig. 18. Experimental to evaluate performance of path planning



(a) Typical motion for RRT* (Forced passes through between robots)

(b) Typical motion for RRT* (Bypass success)

(c) Typical motion for Informed RRT* (Bypass success)

Fig. 19. Typical motion of RRT* and Informed RRT* on path planning

The behavior of each of the three methods, Human-Like, RRT* (unrestricted range), and Informed RRT*, was confirmed. Considering the possibility that slight errors in the actual ball and robot placement could affect the results, more than 10 experiments were performed. In particular, since the path generation of the RRT* and Informed RRT* methods is characterized by randomness, 50 experiments were carried out. An example of the results is shown in Fig. 19. Table 6 also tabulates the success rates. In all experiments in Human-Like, the allly robot interrupted its action by touching the enemy robot. The success rate of bypassing RRT* was about 40%, but it sometimes tried to forcefully pass between enemy robots, and the rate of stopping when it touched an enemy robot was as high as 50%. On the other hand, Informed RRT* did not pass between enemy robots, and its success rate exceeded 90%. These results indicate the usefulness of Informed RRT*, although only under the conditions of this experiment. The average process time under the present experimental conditions, however, was the fastest for Human-like at about 18.0 ms and the slowest for Informed RRT* at about 36.0 ms (RRT* was about 29.5 ms). Therefore, it is necessary to consider faster process time for future introduction.

Table 6. Experimental results

Path generation method \	Success rate for bypass(%)	Rate of forced passes through between robots(%)	Rate of Stoppage(%)
Human-like (10times)	0	0	100
RRT* (50times)	38	12	50
Informed RRT* (50times)	92	0	8

Acknowledgments

This work was supported in part by JSPS KAKENHI Grant Number 20K03263, Grant-in-Aid from the Chuden Foundation for Education and The Nitto Foundation in Japan, respectively.

References

1. Ryoma Mitsuoka, Yusei Naito, Yasutaka Tsuruta, Daichi Miyajima, Kosei Naito, Hironobu Suzuki, Ryuto Tanaka, Hayato Mitsuda, Yota Dori, and Toko Sugiura, KIKS Extended Team Description for RoboCup 2022; https://ssl.robocup.org/wp-content/uploads/2022/04/2022_ETDP_KIKS.pdf
2. Zheyuan Huang, Lingyun Chen, Jiacheng Li, Yunkai Wang, Zexi Chen, Licheng Wen, Jianyang Gu, Peng Hu, and Rong Xiong: ZJUNlict Extended Team Description Paper for RoboCup 2019, https://ssl.robocup.org/wp-content/uploads/2019/03/2019ETDPZJUNlict.pdf(2019).

3. Nicolai Ommer Andre Ryll and Mark Geiger: TIGERs Mannheim Extended Team Description for RoboCup 2019, https://ssl.robocup.org/wp-content/uploads/2019/03/2019ETDPTIGERsMannheim.pdf
4. Yusei Naito, Shin Ohno, Yuta Imaeda, Akihito Odanaka, Yasutaka Tsuruta, Ryoma Mitsuoka, Taisuke Tane, Masato Watanabe, and Toko Sugiura , KIKS Extended Team Description for RoboCup 2020; https://ssl.robocup.org/wp-content/uploads/2020/03/2020_ETDP_KIKS.pdf
5. HITOHADA Gel, https://www.exseal.co.jp/en/products/
6. Nicolai Ommer, Andre Ryll, Mark Geiger, TIGERs Mannheim Extended Team Description for RoboCup 2022, https://ssl.robocup.org/wp-content/uploads/2022/04/2022_ETDP_TIGERs-Mannheim.pdf
7. Zifeng Wu, Chunhua Shen, and Anton van den Hengel, Wider or Deeper: Revisiting the ResNet Model for Visual Recognition, https://arxiv.org/pdf/1611.10080.pdf
8. Rosanne Liu, Joel Lehman, Piero Molino, Felipe Petroski Such, Eric Frank, Alex Sergeev, Jason Yosinski, An intriguing failing of convolutional neural networks and the CoordConv solution, 32nd Conference on Neural Information Processing Systems (NeurIPS 2018), Montréal, Canada, https://proceedings.neurips.cc/paper/2018/file/60106888f8977b71e1f15db7bc9a88d1-Paper.pdf.
9. Md Amirul Islam, Sen Jia, Neil D. B. Bruce, How Much Position Information Do Convolutional Neural Networks Encode? ,https://doi.org/10.48550/arXiv.2001.08248
10. Jonathan D. Gammell, Siddhartha S. Srinivasa, and Timothy D. Barfoot, Informed RRT*: Optimal Sampling-based Path Planning Focused via Direct Sampling of an Admissible Ellipsoidal Heuristic, https://arxiv.org/pdf/1404.2334.pdf