Camellia Dragons

Team Description Paper for RoboCup 2015

Toshiyuki Tanaka, Tatsuya Tsubakimoto, Masanori Kawamura, Kenya Kumagai, Hiroaki Matsubara, Kenta Hidaka, Takuma Murashima, Takuo Suzuki, and Kunikazu Kobayashi[†]

Aichi Prefectural University,
School of Information Science and Technology,
1522-3 Ibaragabasama, Nagakute, Aichi 480-1198, Japan.
[†]kobayashi@ist.aichi-pu.ac.jp

1 Introduction

Camellia Dragons was newly organized in October, 2013 at Aichi Prefectural University (APU), Japan and also a newcomer to RoboCup Standard Platform League (SPL). The first big challenge is to participate in the SPL competition for RoboCup Japan Open 2014. The challenge finally brought the first place in the main competition. In RoboCup Japan Open 2015, the team has won our second championship in a row in the main competition and also won the first place in the technical challenge. The team is really motivated to challenge the SPL competition for RoboCup 2015 to be held in Hefei, China.

2 Team Description

Camellia Dragons is a SPL team set up at APU. The team consists of two masters students, five undergraduate students, and two faculty members; Toshiyuki Tanaka (team leader), Tatsuya Tsubakimoto (ex-team leader), Masanori Kawamura, Kenya Kumagai, Hiroaki Matsubara, Kenta Hidaka, Takuma Murashima, Assist. Prof. Dr. Takuo Suzuki, and Assoc. Prof. Dr. Kunikazu Kobayashi. All of them are affiliated with Intelligent Machine Learning laboratory (IML lab) at APU. Currently, we have 10 NAO robots and all of them are H25 Next Generation (Version 4). The team made a debut at the SPL competition for RoboCup Japan Open 2014 and won the first place in the main competition. In 2015, we participated in the SPL competition for RoboCup Japan Open 2015. Finally, we won the first place in the main competition in a row and also went to the top in the technical challenge.

The team used 2013 B-Human code release at RoboCup Japan Open 2014 and changed 2014 B-Human code release at RoboCup Japan Open 2015. We deeply appreciate B-Human for the great contribution to SPL. For 2015 SPL competition, the team modified 2013 B-Human code and originally added new modules. Specifically, we prepared roles in BehaviorControl module and defined a balance movement in Motion module. The team realized a new function for realizing cooperative action, i.e. action priority. Until the competition, we will implement white goal perception, realistic ball perception, collision avoidance, auto balancer, rhythmic walking and degree of malfunction.

3 SPL Participation

The team prepares to participate all three SPL competitions. We believe that the team has positive impact on development of SPL if participating in RoboCup. Actually, in current SPL, it is hardly seen advanced cooperative play involved two or more robots such as one two pass. Our IML lab has published a lot of papers regarding cooperative behavior in multi-agent system [1–7] in which we use various machine learning techniques [8–11]. We therefore contribute SPL to realize human-like cooperative soccer play involving multi robots. After participating in RoboCup Japan Open 2014, the student members get a chance to learn various fields such as image processing and communication, and then gain broad knowledge from robotics to artificial intelligence. The team make a SPL demo at some robot events in our community to focus spotlight on RoboCup and also APU.

4 Research Contributions

4.1 Role

We assign robots with four kinds of roles, i.e. striker, midfielder, defender, and goalie.

– Striker The basic role is to score a goal. Its home position is near center circle in the opponent field. It can only move in the opponent field. It should wait until a ball enters in own area. When it cannot find a ball, it receives the positional information on the ball from other robots by team communication.
– MidFielder The basic role is to play in all field area and its home position is near the center circle. Mainly, it supports striker by kicking a ball to the opponent field and chasing a ball in the near center circle. It may score a goal.
– Defender The basic role is to kick a ball against the opponent field. It can only act in own field. It always receives the positional information of the ball from other robots by team communication. So it turns to the direction of the ball. When a distance which between current own position and the ball becomes near, it approaches a ball to kick it. We prepare two defenders in right side and left side of own field.
– Goalie The basic role is to defend a goal. Its home position is within own penalty area. When it cannot find a ball, it keeps looking for a ball as shaking own head. After it can find the ball, when the distance between the current position and the ball is far, it moves sideways to the ball. When the distance is near, it goes out of the own penalty area and approach the ball to kick it.

For each role, we define a fixed area for each robot. This represents a home position. Figure 1 shows five centers of home positions corresponding to five robots.

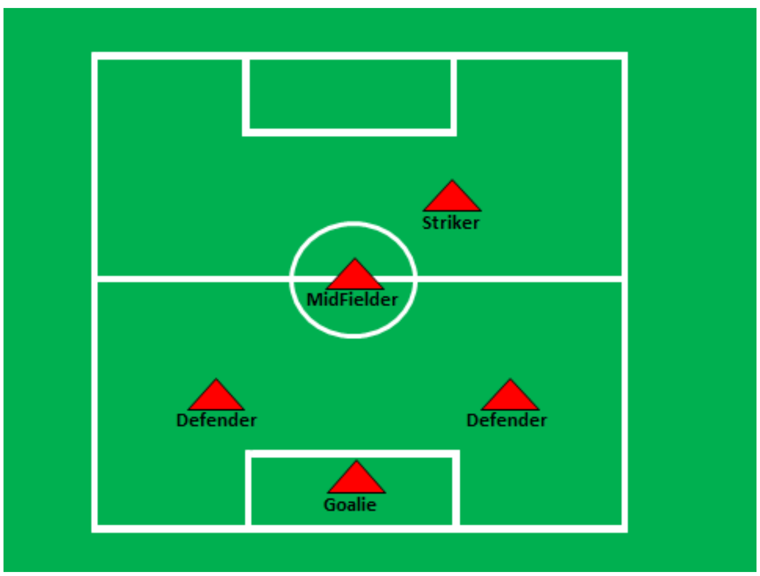

Fig. 1. Home positions for five robots

4.2 Action Priority

In the current SPL competition, individual play is common but cooperative play is hardly seen. That is, only a single robot kicks or dribbles a ball and score a goal. In addition, it is common that a robot handling a ball is interrupted by teammate robots. Of course, cooperative play is important for human soccer and also robot soccer. A method to acquire cooperative action using a reinforcement learning system is proposed by Tsubakimoto et al. [6, 7]. Based on this method, we propose a new method to play soccer cooperatively. The proposed method is not for determining cooperative play but for assist it.

All the robots calculate priority of all the teammates including itself. We need three variable to calculate priority P_k for robot k. That is, d is a distance between a ball and the robot, θ is an angle between an opponent goal and the robot as shown in Fig.2, and v_k is a validity of self-localization. The validity is calculated by unscented Kalman filter in

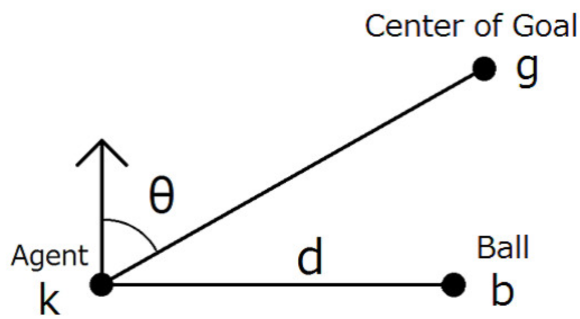

Fig. 2. Positional relation between an agent, a ball and opponent goal.

B-Human's self-localization system. It takes a real value within $[0, 1]$ and the best is 1. Priority is calculated by Eq.(1).

$$P_k = v_k(\alpha\, Dir_{g,k} + \beta\, Dis_{b,k})/(\alpha + \beta)), \tag{1}$$
$$\begin{cases} Dir_{g,k} = (\cos\theta + 1)/2, \\ Dis_{b,k} = d^{-1}, \end{cases}$$

where α and β are both weighting parameters and generally set to 1.0. P_k takes a normalized value within $[0, 1]$. The distance might be more important than the direction in SPL. Both $Dir_{g,k}$ and $Dis_{g,k}$ take a normalized value within $[0, 1]$. Every robots calculate priority for all the teammates and itself at all times. Robots therefore can play soccer cooperatively, e.g. a robot with the best priority will walks to a ball and a robot with the second best priority will receive a ball passed by the robot with the best priority. Furthermore, priority is usable to predict opponent strategy by calculating opponent robot's priority.

4.3 Collision Avoidance

If teammate robots goes toward a ball, they could collide. If collision happens, robots do not move at all or fall down. In such a case, we propose a new collision avoidance method based on a validity measure introduced by B-Human. Originally, the validity is proposed for estimating accuracy of self-localization. We use the validity as defining a personal space which represents a circle centered at a robot. As seen in Fig.3, the personal space for robot A is defined as a circle with radius cv_a centered at robot A. Here, v_a refers to a validity for robot A and c is a constant. The personal space is used for judging how near two robots. If two circles for robot A and B are overlapped, i.e. Eq.(2) is satisfied, one and only one robot which distance between the robot and a ball is near will approach the ball.

$$d \le c(v_a + v_b). \tag{2}$$

In such a way, two robots could avoid a collision and also the other robot could take another role such as a pass receiver or a defender.

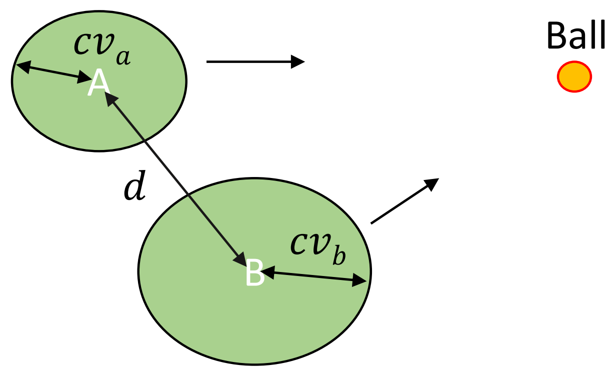

Fig. 3. Collision Avoidance

4.4 AutoBalancer

As mentioned before, the team uses a walking engine by B-Human [12, 13]. We however found that it tends to fall down on a slippery field and causes malfunction of robots. To overcome this, the team introduce a new idea of autobalancer, which is an automatic balance control system when a robot lose balance and is going to fall down.

Although we think several possible methods to control balance, we pick up a simple case that a robot is falling forward and backward. We inspired by human behavior and then come up with the following idea. If a robot is falling forward and backward, it swing the arms forward and backward to prevent falling down, respectively. To implement this idea, we firstly have to detect forward or backward lean of torso using a gyro sensor embedded in the NAO robot. Secondly, a robot swing the arms forward or backward based on the measured value of gyro.

4.5 Rhythmic Walking

When we use the walking engine by B-Human [12, 13], we need to empirically adjust various waking parameters according to field surface conditions. If an non-expert sets bad parameters, a robot easy to fall down. We therefore introduce a rhythmic walking engine with automatic parameter setting. Rhythmic walking is realized by a model of central pattern generator (CPG). CPG has inspired theoretical studies of motor pattern generation in isolated neural networks in the absence of sensory feedback [14].

As introducing a CPG model to the walking engine, we try to realize rhythmic walking. As a result, it could improve the stability of walking. In addition, we introduce genetic algorithm (GA) to automatically adjust CPG's parameters. GA is one of evolutionary computation methods [15]. As optimizing the parameters by GA, we do not require an expert who adjusts optimal parameters in any conditions. Through computer simulation, rhythmic walking is verified.

4.6 Degree of Malfunction

When a human is playing soccer in a game, he/her will go on the blink according to some physical accidents or injuries. Similarly, a soccer robot tends to drop the performance during a game and finally likely to break down. It mainly comes from heat of joint motors and a CPU, physical contacts with another robots, and falling down. NAO robot has an automatic shutdown function when the heat exceeds a threshold determined in advance. This however cannot predict malfunction of a robot.

We therefore propose a degree of malfunction to detect a robot with bad condition. It is calculated by a three-layered neural network as shown in Fig.4 [8]. The inputs to the network are factors which cause a malfunction such as heat of joint motors and a CPU and remaining battery level. The merits to use the neural network are easily to get any input-output function if giving training data set and having high generalization ability. The neural network has to train using a game data set by error backpropagation algorithm in advance [16]. After that, the neural network realizes relationship between factors of malfunction (inputs) and a degree of malfunction (an output). During a game, a robot can classify unknown data as an appropriate class and detect malfunction of a robot. The proposed degree is available for calling request for pick-up and exchanging robots.

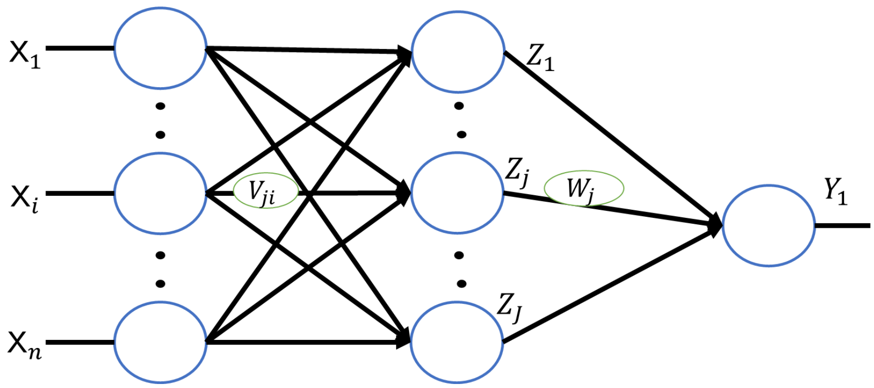

Fig. 4. A three-layered neural network

5 Conclusion

We have prepared the four new roles for striker, midfielder, defender, and goalie based on 2014 B-Human code release for RoboCup 2015 in the SPL competition. The team have proposed the new function for realizing cooperative action, i.e. action priority. Through the main competition for RoboCup Japan Open 2015, it is verified that the proposed methods provide good performance.

Acknowledgments

Camellia Dragons is financially supported by President's Special Research Fund from Aichi Prefectural University, Japan from April 2014.

References

1. Kobayashi K, Kanehira R, Kuremoto T, and Obayashi M (2011), An Action Selection Method Based on Estimation of Other's Intention in Time-Varying Multi-Agent Environments. Lecture Notes in Computer Science, Vol.7064, pp.76-85, Springer-Verlag.
2. Kobayashi K, Kurano T, Kuremoto T, and Obayashi M (2012), Cooperative Behavior Acquisition in Multi-agent Reinforcement Learning System Using Attention Degree. Lecture Notes in Computer Science, Vol.7665, pp.537-544, Springer-Verlag.
3. Kuremoto T, Tsurusaki T, Kobayashi K, Mabu S, and Obayashi M (2013), An Improved Reinforcement Learning System Using Affective Factors. Robotics, Vol.2, No.3, pp.149-164.
4. Watada S, Obayashi M, Kuremoto T, Mabu S, and Kobayashi K (2014), A Decision Making System of Robots Introducing a Re-construction of Emotions Based on Their Own Experiences. Journal of Robotics, Networking and Artificial Life, Vol.1, No.1, pp.27-32.
5. Kawamura M and Kobayashi K (2014), An Action Selection Method Using Degree of Cooperation in a Multi-agent Reinforcement Learning System. Journal of Robotics, Networking and Artificial Life, Vol.1, No.3, pp.231-236.
6. Tsubakimoto T and Kobayashi K (2014), Cooperative Action Acquisition Based on Intention Estimation Method in a Multi-agent Reinforcement Learning System. Proceedings of the International Conference on Artificial Life and Robotics (ICAROB2014), pp.122-125.
7. Tsubakimoto T and Kobayashi K (2015), Cooperative Action Acquisition Based on Intention Estimation in a Multi-agent Reinforcement Learning System. IEEJ Transactions on Electronics, Information and Systems, Vol.135 No.1 pp.117–122 (in Japanese).

8. Alpaydin E (2009), Introduction to Machine Learning. The MIT Press.
9. Marsland S (2009), Machine Learning: An Algorithmic Perspective. Chapman and Hall/CRC.
10. Murphy KP (2012), Machine Learning: A Probabilistic Perspective. The MIT Press.
11. Bishop CM (2010), Pattern Recognition and Machine Learning. Springer.
12. Röfer T, Laue T, Müller J, Bartsch M, Batram MJ, Böckmann A, Böschen M, Kroker M, Maaß F, Münder T, Steinbeck M, Stolpmann A, Taddiken S, Tsogias A, and Wenk F (2013), B-Human Team Report and Code Release 2013. Only available online: http://www.b-human.de/downloads/publications/2013/CodeRelease2013.pdf.
13. Röfer T, Laue T, Müller J, Schüthe D, Böckmann A, Janett D, Koralewski S, Maaß F, Maier E, Siemer C, Tsogias A, Vosteen J (2014), B-Human Team Report and Code Release 2014. Only available online: https://www.b-human.de/downloads/publications/2014/CodeRelease2014.pdf.
14. Taga G, Yamaguchi Y, and Shimizu H (1991), Self-organized control of bipedal locomotion by neural oscillators in unpredictable environment. Biological Cybernetics, Vol.65 pp.147–159.
15. Goldberg DE (1989), Genetic Algorithms in Search, Optimization and Machine Learning. Addison-Wesley Longman Publishing.
16. Rumelhart DE, Hinton GE, and Williams RJ (1986), Learning Representations by Back-Propagating Errors. Nature, Vol.323 pp.533–536.