Learning of Expected Scores Distribution for Positions of Digital Curling

Masahito Yamamoto[1,a)] Shu Kato[2] Hiroyuki Iizuka[1]

Abstract: Curling is a team sport played by two teams of four players on the ice. Recently, not only the skill but also the strategy becomes very important for winning the curling games. In order to discuss about the optimal strategy for curling, the curling playing simulator called by digital curling has been developed by Ito and his colleagues. In digital curling, rules are equivalent to actual curling, and the outcome of the delivery is simulated by internal physical computation. On the digital curling, we have developed curling AI program jiritsu-kun which can search for the best play based on the game tree search. In this paper, we propose the learning method of the expected scores distribution at the end of the "end" as a static evaluation function of the game tree search. It is based on a deep neural network model. In order to evaluate our proposed method, we compare the learned evaluation function with hand-crafted evaluation function in the previous version of jiritsu-kun.

1. Introduction

Curling is a team sport played by two teams of four players on the ice, and it has been an official sport in the Winter Olympic Games since 1998. Recently, not only the skill but also the strategy becomes very important for winning the curling games. Ito et al. have developed a simulated environment for playing Curling in order to discuss and study on the strategy of Curling [1]. In Digital Curling, the player (AI program or human) should decide the target point and the turn direction (in-turn or out-turn) in a given position. The outcome of this delivery is simulated with some random noise and therefore it is not guaranteed that the same delivery leads to the same result.

To overcome such an uncertainty, we adopt an expectimax method based on the game tree search and developed AI program called jiritsu-kun. An expectimax method is a specialized variation of a minimax game tree that play two-player zero-sum games such as backgammon, in which the outcome depends on a combination of the player's skill and chance elements such as dice rolls or other noise. In order to apply the game tree search to find a good curling strategy, the goodness of the current state must be evaluated by using the static evaluation function. Thus it is very important to design an evaluation function appropriately which maps the current state (position) to a real value. In the previous version of jiritsu-kun called jiritsu-Ev, a hand-crafted evaluation function have been proposed [2]. In this paper, we propose a learning method of a static evaluation function which can be used for game tree search and developed the Curling AI program called jiritsu-NN. In oder to evaluate the learned static evaluation function, we conducted the tournament experiments of three AI programs.

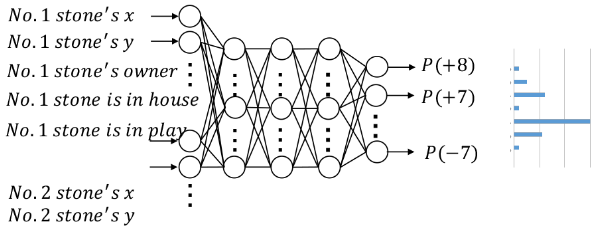

Fig. 1 Neural network structure.

2. Learning of Evaluation Function

2.1 Expected Scores Distribution

In this paper, a deep neural network is used for learning the evaluation function of Curling. Figure 1 shows the structure of the deep neural network with five layers. The inputs of the neural network are the x and y coordinations of the stone, stone's owner, in house or not and in play or not for each stone. The closest stone to the tee (the center of the house) is called No. 1 stone and the second closest stone is called No. 2 stone and so on. The input order of stones is the above five values of No. 1 stone, and that of No. 2 stones, and so on as shown in Fig. 1.

The outputs of the neural network are the expected probability of getting 8 points, 7 points,...., -8 points (losing 8 points), respectively at the end of the "end" [*1]. If the output of neural network can predict the end score appropriately,

[1] Graduate School of Information Science and Technology, Hokkaido University, Sapporo 060-0814, Japan
[2] Panasonic Corporation, Osaka 571-8501, Japan
[a)] masahito@complex.ist.hokudai.ac.jp

[*1] An end means a portion of a curling game that is completed when each team has thrown eight stones and/or the score has been decided.

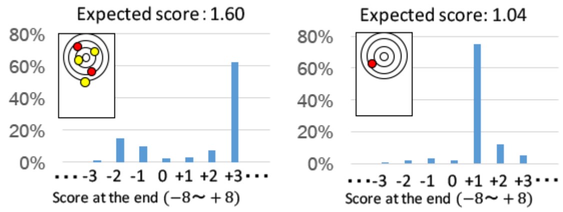

Fig. 2 An example of expected scores distribution for two positions.

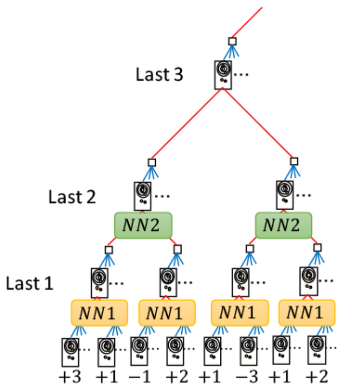

Fig. 3 Learing of other neural networks NN-x.

the decision making process of the player in each turn will be improved. For example, a position with the expected scores distribution as shown in Fig. 2 (left) is superior in terms of the expected score than the right position in Fig. 2. However, the right position is more favorable than the left position in the case that the team is leading or tied at the last end, because at least one point is enough for the team.

To implement this kind of decision making through the game, we designed the following evaluation function E_{nn},

$$E_{nn} = \sum_{i=-8}^{+8} y_i \times w(r, d, i), \quad (1)$$

where y_i is the $i-$th output of the neural network and $w(r, d, i)$ is the winning percentage for the team in case that the team gets i points in score differences d with r ends remaining. The values of $w(r, d, i)$ is called "Game Equity Table" inspired by the backgammon software, and it is calculated based on a large number of self-plays in advance.

2.2 Learning of the neural networks

First we try to learn the expected scores distribution of the last shot position that 15 stones have already delivered and only the last shot is remaining. The aim of the learning is to adjust the wights of the neural network so as to predict the true scores distribution for arbitrary positions well.

A large number of (125 thousands) positions for learning are extracted from the self-play game records between jiritsu-Ev and jiritsu-Ev. Another 125 thousands positions are generated by searching result from the above positions. Other 250 thousands position are randomly generated. The symmetric positions of the above positions are added to the data set and 90% positions of all data set are used for training the neural network. The teaching signals are calculated by the search results to the end of the "end". The obtained learned neural network is called NN1.

Table 1 The result against the GAT champion program in 2016

Player	Opponent	Winning % (First/Second)
jiritsu-NN	AyumuGAT2016	62.5 ± 3.4 (59.0/66.0)
jiritsu-NN	jiritsu-Ev	58.0 ± 3.5 (49.0/67.0)
jiritsu-Ev	AyumuGAT2016	38.5 ± 3.4 (25.0/52.0)

Next the evaluation function of the last 2 shot positions are learned by using the neural network NN1 learned in the previous procedure. For a certain last 2 shot position, the best shot searched by using jiritsu-Ev is selected, and the resulting position is evaluated by using NN1. This evaluation values become the teaching signal of the original position.

In this manner, the neural network NNx in the last x shot position is learned by using the neural network NN$x-1$. In all learning process, the number of epochs is 20 and the batch size is 10 and Mean Square Error (MSE) is used as the loss function.

3. Results

In order to evaluate the learned static evaluation function, we prepared the Curling AI program jiritsu-NN with the leaned static evaluation function and compared it with jiritsu-Ev. Furthermore the champion program Ayumu-GAT2016 of GAT cup 2016 is also played against jiritsu-NN and jiristu-Ev.

Table 1 shows the tournament results for three AI programs. The number of ends for all games is 8 and the total 400 games are played (200 games for the first player and 200 games for the second player). The search depth of both jiritsu-NN and jiritsu-Ev is 2 and the learned neural networks NN1∼NN14 are used in jiritsu-NN.

The result shows that our proposed AI program jiritsu-NN is the strongest among three programs.

4. Concluding Remarks

We claimed that the expected scores distribution is more effective for finding the best shot in terms of the winning percentage considering the score differences and the number of ends remaining. The five layered deep neural network was used for learning the expected scores distribution of arbitrary positions. It is shown that the neural network NNx for the last x shot position can be learned by using NN$x-1$. The experimental results show that our proposed AI program jiritsu-NN is superior than other two programs. In the near future, the prediction precision of the expected scores distribution will be improved by using more large number of position data.

References

[1] Digital curling website (Ito laboratory at the University of Electro-Communication (UEC), in Japanese), http://minerva.cs.uec.ac.jp/ curling/wiki.cgi?page=FrontPage

[2] Yamamoto M., Kato S., Iizuka H.: Digital Curling Strategy based on Game Tree Search, Proceedings of 2015 IEEE Conference on Computational Intelligence and Games, pp. 474-480 (2015)