

Title	格闘ゲーム初心者のための読み合い理解支援システム
Author(s)	酒見, 真
Citation	
Issue Date	2023-03
Type	Thesis or Dissertation
Text version	author
URL	<a href="http://hdl.handle.net/10119/18357">http://hdl.handle.net/10119/18357</a>
Rights	
Description	Supervisor: 池田 心, 先端科学技術研究科, 修士 (情報科学)

修士論文

格闘ゲーム初心者のための読み合い理解支援システム

酒見 真

主指導教員 池田 心

北陸先端科学技術大学院大学  
先端科学技術研究科  
(情報科学)

令和5年3月

## Abstract

Artificial intelligence (AI) techniques are actively researched in various fields, such as natural language processing and image generation. Games are also good testbeds for AI that are actively researched. In recent years, AI players have already achieved superhuman levels and beaten top professional players (e.g., AlphaGo for the game of Go). Research on AI pursuing “strength” has archived a certain level of success, and research on “teaching” and “entertaining” are new directions to focus.

Fighting games are a kind of digital game where each player controls a character to fight one-on-one in real-time. Fighting games are known as a genre with high barriers to entry for several reasons. One is difficult operations; another is that luck is almost not involved in the gameplay, making beginners hardly win against experienced players. We consider that “mind games” are another big problem preventing beginners from keeping playing the games.

“Mind games” means that players mutually decide their own actions, where guessing the opponents’ decisions is also taken into consideration. Mind games occur in daily life, such as the rock-paper-scissors game. In these games, it is basically possible to calculate the optimal strategy by game theory, even without mind games. However, in fighting games, possible actions and their risk-return varies with various conditions. Therefore, it is difficult to calculate the optimal strategy, and players need to guess the opponents’ actions (read the opponents’ minds). The same is true for the opponent, so mind games occur.

The purpose of this research is to help beginners of fighting games to understand mind games. Fighting games contain various elements, such as real-time gameplay and difficult operations, which requires considerable practice before players can experience mind games. Therefore, it is difficult to teach beginners to play mind games in actual fighting game environments. So, we developed a game where players can play mind games in fighting games without real-time gameplay and difficult operations. The game is based on rock-paper-scissors game, and the trilemma of rock, paper, and scissors is replaced by “attack”, “throw” and “guard” from fighting games. Also, we created a curriculum that provides comprehensive supports for beginners to understand the concept of mind games.

We selected several ways of thinking for mind games that we want to teach and prepared multiple opponents for each way. The combination of a way of thinking and the corresponding opponents formed a learning unit, and in this research, six learning units were created. The six created learning units were arranged in order of difficulty, from those with typical ideas to those with gradually more complex ideas. The opponents’ actions were designed to reflect the way of thinking in the corresponding learning unit, and the order of the matches against

the opponents was also arranged in a similar way to the learning units, from opponents who take typical actions to those who gradually get more complex. During the matches, participants were asked to answer the probability distribution of the opponent's next action through choice problems. We used the accuracy of these choice problems to judge whether or not the participant understood the way of thinking set to the learning unit. In each learning unit, we also supported players by presenting hints such as information such as the way of thinking set to the learning unit and the points to be focused on for each opponent. The order of the learning units, opponents for each learning unit, and the hints set for each are collectively referred to as the curriculum.

An experiment was conducted to verify the effectiveness of the created curriculum. Meanwhile, the effectiveness of the order of the learning unit was also verified. The experiment compared the results from three groups: a group receiving the curriculum (hereinafter called Curriculum group), a group randomly playing against opponents within the curriculum (hereinafter called Random group), and a group playing against participants in the same group for the same amount of time as the other groups (hereinafter called PvP group). A pre-test and a post-test were done before and after the experiment of each group to compare the accuracy of choice problems. This experiment was conducted with a total of 12 participants, four in each group.

The result of the experiment showed that even the group with the highest accuracy only increased by 9.0%, from 10% to 19.0%. The accuracy for each participant varied widely, and no tendency was seen from all the groups. In a subjective survey, only 2 out of 8 participants in the Curriculum group and the Random group felt an improvement in their prediction accuracy. The overall results were poor, but valuable insights were gained about the curriculum and game design. Points to be improved included probability deviation, the criterion of correct answers for choice problems, and the difficulty of the test opponents.

The biggest issue arose when playing against an opponent who chooses actions based on probability, where the influence of luck was too strong. For example, consider an opponent who chooses each of the actions of attack, throw, and guard with a probability of  $\frac{1}{3}$ . Assume that the opponent made 7 action selections with a deviated frequency: attack 4 times, throw 3 times, and guard 0 times. In this case, choosing the choice of attack 50%, throw 50%, and guard 0% as an answer to the choice problem is reasonable, but in this experiment, we judged this answer to be incorrect because it is not the true probability distribution. As a result, we often saw participants who could not proceed with the curriculum due to low accuracy of choice problems. Therefore, in the first place, we attempted to improve the curriculum by correcting the probabilities to prevent the frequency from becoming

deviated and by judging not only the true probability distribution but also choices close to the actual frequency of the opponent's actions as correct answers.

About the difficulty of the test opponents, both of pre-test and post-test had very low accuracy of choice problems, so we consider the test opponents were too difficult. Therefore, we reduced the difficulty of all opponents and increased the variety of the test opponents to let participants match against opponents with various difficulty. In addition, we attempted to improve various points obtained from first experiment.

The second experiment was conducted after improving the curriculum and game design by addressing the issues found in the first experiment. The experimental method and comparison method were the same as the first experiment. The number of participants in the second experiment was 20, with 6 participants in each of the Curriculum group and the Random group and 8 participants in the PvP group. In the result, the difference in accuracy between the pre-test and post-test showed an increase of 11.0% from 34.0% to 45.0% in the Curriculum group, 14.0% from 41.0% to 55.0% in the Random group, and 2.0% from 34.0% to 36.0% in the PvP group. However, no significant difference was shown between pre-test and post-test or between each group. As for the accuracy of each participant, the growth rate varied, and there seemed to be a relationship with proficiency in Japanese. According to the subjective survey from the questionnaire, the results improved compared to the first experiment, and 11 out of 12 participants in the Curriculum group and the Random group answered that they felt the accuracy had increased. Compared to the first experiment, the numerical growth and subjective survey showed good results, but new issues to improve were also found, such as finding concise and easy-to-understand hints.

## 概要

人工知能 (AI) の技術は、自然言語処理や画像生成など幅広い分野で盛んに研究が行われている。ゲームもまた AI の良いテストベッドであり、研究が盛んにおこなわれている分野のひとつである。近年では、囲碁の AlphaGo のように、人間のトッププロを上回る AI プレイヤも登場している。このように、AI を用いて「強さ」を求める研究は、一定の成果を挙げており、新たな研究の方向性として、「教える」「楽しませる」といった強さ以外の要素に着目する研究が行われている。

格闘ゲームはゲームジャンルのひとつであり、各プレイヤがキャラクタを操作して1対1で戦う、リアルタイム性のデジタルゲームである。格闘ゲームは難解なコマンド操作や運が介入しづらいゲーム性など、初心者が定着しにくい要因が多いジャンルということでも知られている。我々は、これらの初心者定着を妨げる要因のひとつである、“読み合い” が大きな問題だと考える。

読み合いとは、互いに相手の行動を推測し、自身の行動を決定することである。読み合いは日常生活でも発生するものであり、じゃんけんなどがその例に挙げられる。これらのゲームは読み合いを行わずとも、基本的に、ゲーム理論を用いることで、最適な行動選択の確率を計算することが可能である。しかし、格闘ゲームにおいては、様々な諸条件により、選択可能な行動とそのリスクリターンが逐一変化する。そのため、最適な行動選択の確率を計算することが難しく、プレイヤは相手の行動を読む必要がある。これは相手にも同じことが言えるため、読み合いが発生する。

本研究の目的は「格闘ゲーム初心者が読み合いを理解するための支援」である。格闘ゲームはリアルタイム性やアクション性など様々な要素が含まれているため、操作に慣れるなどの、読み合いの前段階の練習が必要である。そのため、実際の格闘ゲームの環境を使用して、読み合いのみに焦点を当てた支援を行うことは難しい。そこで、リアルタイム性やアクション性などの要素を排除し、格闘ゲームの読み合いを体験できるゲームを自作した。自作ゲームはじゃんけんをベースとし、グー、チョキ、パーの3すくみを、格闘ゲームの打撃、投げ、防御に置き換えたものである。また、初心者が自作ゲームをプレイすることで、読み合いの考え方を理解できるような、総合的な支援を含めたカリキュラムを作成した。

伝えたい読み合いについての考え方を選定し、それぞれに、複数の対戦相手を設定した。考え方と対戦相手をまとめてタームと呼称し、本研究では6つのタームを作成した。作成した6つのタームを、典型的な考え方を持つものから、徐々に複雑な考え方をもつものとなる順で並べた。各タームに設定した対戦相手は、そのタームで伝えたい考え方を反映した行動をとるようにし、対戦順は、タームと同じく、典型的な行動をとる対戦相手から、徐々に複雑なものとした。対戦を行う際には、選択式問題として、対戦相手の次の行動の確率分布を回答させた。この選択式問題の正答率により、タームの伝えたい考え方を理解したか判定する。また、各タームではゲームプレイヤに、そのタームで伝えたい考え方や、対戦相手ごとに設定した着目すべき点などの情報を、ヒントとして提示するなどの支援も

行う。なお、タームの並びやタームごとの対戦相手、それぞれに設定したヒントなどを、ひとまとめにしてカリキュラムとする。

作成したカリキュラムの有効性を検証するための実験を行った。それと同時に、タームの順番に関して有効性の検証を行った。実験方法としては、カリキュラムを受ける群（以降カリキュラム群）、カリキュラム内の対戦相手とランダムで対戦する群（以降ランダム群）、他群と同時間対人戦を行う群（以降対人群）の3群を用意し比較を行う。各群実験の前後でプレテストとポストテストを行い、これらの選択式問題の正答率を比較する。実験は各群4名の計12名で行った。

実験の結果としては、最も正答率を伸ばした群においても、10.0%から19.0%と9.0%程度の伸び幅であった。被験者ごとの正答率は、伸び幅がバラついており、傾向のようなものはどの群にもみられなかった。アンケートによる主観調査で、予測精度の上昇を実感したと回答した被験者は、カリキュラム群とランダム群の8名中2名と少数であった。全体的に悪い結果だったが、カリキュラムや自作ゲームの仕様に関して様々な改善すべき点を得た。改善すべき点には、「確率の偏り」や「選択式問題の正答の基準」、「テスト用対戦相手の難易度」などがあつた。

最大の問題は、確率で行動を選択する相手との対戦において、運不運の要素が強すぎることであつた。例えば、打撃と投げと防御を全て $\frac{1}{3}$ の確率分布で選ぶ対戦相手が、7回の行動選択で、打撃4回、投げ3回、防御0回と偏つた頻度で行動を選択したとする。この際、選択式問題の回答として、打撃5割、投げ5割、防御0割のような選択肢を選ぶことは妥当であるが、この実験では真の確率分布でないことから不正解とした。これにより、正答率を落としてカリキュラムを進めることができない被験者がしばしばみられた。そこで、そもそも偏つた頻度となることを防ぐために、確率に補正を行うこと、また、真の確率分布だけでなく、対戦相手の実際の選択頻度に近い選択肢も正解扱いとすることで、改善を試みた。

テスト用対戦相手の難易度に関して、プレテストとポストテストどちらも正答率が非常に低く、難しすぎたと考えた。そこで全体的な難易度を下げ、種類も増やすことで、様々な難易度の対戦相手を用いたテストを行えるようにし、改善を試みた。この他にも得られた改善すべき点に、様々な方法で改善を試みた。

改善すべき点に関して変更を施したのち、再度被験者実験を行った。実験方法や比較方法は1度目の被験者実験と同様である。2度目実験における被験者数は、カリキュラム群とランダム群が各6名、対人群が8名の計20名であつた。結果として、プレテストとポストテストの正答率の差は、カリキュラム群が34.0%から45.0%で11.0%、ランダム群が41.0%から5.0%で14.0%、対人群が34.0%から36.0%で2.0%の伸びを見せた。しかし、プレテストとポストテスト間にも、各群間にも有意な差はみられなかった。被験者ごとに正答率を見ると、伸び率はまちまちであり、日本語の得手不得手も関係があるようであつた。アンケートによる主観的評価では、1度目より改善されており、カリキュラム群とランダム群の12名中11名が予測精度が上昇したと感じると回答した。1度目の実験と比較すると、数値の伸びや主観評価はよい結果を得たが、ユニバーサルで分かりやすいヒントとするな

ど、新たな課題も見つかった.

# 目次

<b>第1章</b>	<b>はじめに</b>	<b>1</b>
<b>第2章</b>	<b>対象ゲーム</b>	<b>4</b>
2.1	格闘ゲーム	4
2.1.1	格闘ゲームの概要と基本的なルール	4
2.1.2	初心者が直面する障壁	5
2.2	格闘ゲームの読み合い	6
2.2.1	読み合い	6
2.2.2	格闘ゲームにおける読み合い	6
2.2.3	本研究で取り扱う読み合い	7
<b>第3章</b>	<b>関連研究</b>	<b>9</b>
<b>第4章</b>	<b>提案手法</b>	<b>11</b>
4.1	拡張じゃんけんゲーム	13
4.1.1	特殊打撃と特殊投げ	14
4.2	カリキュラム	16
4.2.1	カリキュラム策定のための予備実験	17
4.2.2	予備実験から得た考慮すべき要素	19
4.2.3	カリキュラムの構成	20
4.2.4	ヒント	22
4.2.5	行動規則の把握の判定	23
<b>第5章</b>	<b>第1回被験者実験</b>	<b>25</b>
5.1	実験目的	25
5.2	実験設定	25
5.3	実験1におけるカリキュラムの設定	27
5.3.1	選択式問題	27
5.3.2	各CPUの正答率の閾値とそのほか諸条件	28
5.3.3	各CPUのヒント提示画面	29
5.3.4	ランダム群へのヒント提示	30
5.4	実験結果	31
5.5	実験を通じて得られた改善すべき点	35

5.5.1	カリキュラムに関して	35
5.5.2	ゲームの仕様に関して	37
<b>第6章</b>	<b>第2回被験者実験</b>	<b>39</b>
6.1	カリキュラムに関する改善策	39
6.1.1	確率の偏りを低減	39
6.1.2	選択式問題の正答の基準を変更	41
6.1.3	選択式問題の実施回数の変更	41
6.1.4	ターム冒頭のヒントの再提示	42
6.1.5	CPUごとのヒントの増加	42
6.1.6	修了したCPUの行動規則の提示	43
6.1.7	カリキュラムで対戦するCPU数を減らす	43
6.1.8	テスト用CPUの変更	43
6.2	ゲームの仕様に関する改善策	44
6.2.1	ゲーム内情報の追加	44
6.2.2	選択式問題の選択肢提示方法の変更	45
6.3	実験の目的と設定	46
6.4	実験2におけるカリキュラムの設定	47
6.4.1	選択式問題	47
6.4.2	各CPUの正答率の閾値	48
6.4.3	各CPUのヒント提示画面	48
6.4.4	ランダム群へのヒント提示	49
6.5	実験結果	49
6.6	改善策の効果	53
6.6.1	確率の偏りの低減	53
6.6.2	工夫による確率的CPUでの正答率の変化	54
6.6.3	テスト用CPUの難易度	56
<b>第7章</b>	<b>おわりに</b>	<b>58</b>
<b>付録A</b>	<b>詳細なデータ</b>	<b>63</b>
A.1	工夫による確率的CPUでの正答率の変化の詳細なグラフ	63
<b>付録B</b>	<b>実験に関する設定など</b>	<b>66</b>
B.1	実験1で使用したCPUの行動規則一覧	66
B.2	実験2で使用したCPUの行動規則一覧	74

# 目次

2.1	ストリートファイター V 対戦画面 (出典 ストリートファイター V 公式サイト [1])	5
4.1	GUILTY GEAR -STRIVE- ミッションモード画面 (出典 GUILTY GEAR -STRIVE-公式サイト [2])	11
4.2	拡張じゃんけんゲーム 対戦画面	13
4.3	カリキュラムと各タームの進め方の詳細	16
4.4	CPU との対戦の進め方	17
4.5	タームごとの対戦する CPU を選択するゲーム画面	22
4.6	選択式問題の概略図	23
5.1	選択式問題への回答時のゲーム画面 (実験 2 とは異なる)	27
5.2	実験 1 の際の各 CPU のヒントを提示するゲーム画面 (実験 2 とは異なる)	29
5.3	ランダム群にヒントを表示する画面	30
5.4	実験 1 における各群のプレテストとポストテストの正答率	32
5.5	実験 1 における各被験者のプレテストとポストテストの正答率	33
5.6	実験 1 における CPU の修了に要した平均ゲーム数	34
5.7	実験 1 におけるテスト用 CPU ごとの正答率	37
6.1	補正式を用いた各手の選択確率の導出 ( $\alpha = 1.6$ )	40
6.2	実験 2 における対戦前のヒント提示画面	42
6.3	修了後の行動規則の提示	43
6.4	実験 2 における対戦画面	44
6.5	実験 2 における選択式問題の提示方法	45
6.6	実験 2 における各群のプレテストとポストテストの正答率	50
6.7	実験 2 における各被験者のプレテストとポストテストの正答率	51
6.8	実験 2 における CPU の修了に要した平均ゲーム数 (下部は実験 1 における平均ゲーム数)	52
6.9	実験 2 におけるタームごとの正答率 (1 ゲーム目)	54
6.10	実験 2 における各 CPU の被験者ごとの正答率 (1 ゲーム目)	55
6.11	実験 2 におけるテスト用 CPU ごとの正答率	56
6.12	CPU4 の各選択肢が選ばれた回数	57

A.1	実験2における各CPUの被験者ごとの正答率（1ゲーム目，再掲）	63
A.2	実験2における各CPUの被験者ごとの正答率（1ゲーム目） . . . .	64
A.3	実験2における各CPUの被験者ごとの正答率（1ゲーム目） . . . .	64
A.4	ターム3のCPU4における各被験者の正答率推移（実験2） . . . .	65

# 表 目 次

4.1	格闘ゲームの要素と自作ゲームでの扱い	12
4.2	拡張じゃんけんゲームにおける選択可能手とその効果	14
4.3	予備実験で使用したCPUの行動規則	18
4.4	タームの全容	20
4.5	各タームに関連する読み合いの要素	20
4.6	各タームに設定したヒント	22
5.1	実験1におけるCPUごとの閾値と条件	28
5.2	実験1におけるアンケート結果	31
5.3	ターム3のCPU2における各プレイヤーの1ゲーム目の各手の選択頻度（実験1）	35
6.1	実験2におけるCPUごとの閾値	48
6.2	実験2におけるアンケート結果	49
6.3	実験1におけるアンケート結果（再掲）	50
6.4	ターム3のCPU2における各プレイヤーの1ゲーム目の各手の選択頻度（実験2）	53
6.5	CPU4における選択式問題の選択肢（打撃：投げ：防御）	57
B.1	実験1におけるターム1のCPUの行動規則とヒント	67
B.2	実験1におけるターム1のCPUの選択式問題選択肢	67
B.3	実験1におけるターム2のCPUの行動規則とヒント	68
B.4	実験1におけるターム2のCPUの選択式問題選択肢	68
B.5	実験1におけるターム3のCPUの行動規則とヒント	69
B.6	実験1におけるターム3のCPUの選択式問題選択肢	69
B.7	実験1におけるターム4のCPUの行動規則とヒント	70
B.8	実験1におけるターム4のCPUの選択式問題選択肢	71
B.9	実験1におけるターム5のCPUの行動規則とヒント	72
B.10	実験1におけるターム5のCPUの選択式問題選択肢	72
B.11	実験1におけるターム6のCPUの行動規則とヒント	73
B.12	実験1におけるターム6のCPUの選択式問題選択肢	73
B.13	実験2におけるターム1のCPUの行動規則とヒント	75
B.14	実験2におけるターム1のCPUの選択式問題選択肢	75

B.15 実験 2 におけるターム 2 の CPU の行動規則とヒント . . . . .	76
B.16 実験 2 におけるターム 2 の CPU の選択式問題選択肢 . . . . .	76
B.17 実験 2 におけるターム 3 の CPU の行動規則とヒント . . . . .	77
B.18 実験 2 におけるターム 3 の CPU の選択式問題選択肢 . . . . .	77
B.19 実験 2 におけるターム 4 の CPU の行動規則とヒント . . . . .	78
B.20 実験 2 におけるターム 4 の CPU の選択式問題選択肢 . . . . .	78
B.21 実験 2 におけるターム 5 の CPU の行動規則とヒント . . . . .	79
B.22 実験 2 におけるターム 5 の CPU の選択式問題選択肢 . . . . .	79
B.23 実験 2 におけるターム 6 の CPU の行動規則とヒント . . . . .	80
B.24 実験 2 におけるターム 6 の CPU の選択式問題選択肢 . . . . .	80

# 第1章 はじめに

近年の計算機の性能向上やアルゴリズムの発展に伴い、人工知能（AI）技術はめざましい成果を収めている。ルールや勝利条件、結果などが明確に示されかつ様々な種類が存在するゲームは、AIのテストベッドとしてよく使用されている。そしてそのゲームにおいてもAIは非常に良い成績を収めている。2013年にはPonanzaという将棋AIが史上初めてプロ棋士に勝利し、2016年には囲碁において、DeepMind社が開発したAlphaGOが当時のトップ棋士を相手に勝利を収めた [3]。AIプレイヤーが人間を上回る例は、将棋や囲碁のような古典的ボードゲームに限らない。2019年には、Meta社（旧Facebook社）とカーネギーメロン大学が共同開発したポーカーAIであるPluribusが、ノーリミットテキサスホールデムを対象とした実験で、複数のポーカープロとの対戦に勝利し [4]、2020年にDeepMind社が開発したAgent57というAIプレイヤーは、Atari社のAtari2600というゲーム機に収録された57種のゲーム全てにおいて、人間の平均的なスコア以上のスコアを収めた [5]。

このようなAIに対戦相手としての強さを求める研究が、さまざまなゲームにおいてある程度の成績を達成した一方で、次のステップとしてゲームプレイの満足度を高める [6] ことや、ゲームそのものをより面白くするためにAIを活用する研究 [7] などが注目を集めている。マップの自動生成 [8] や、AIを用いた上達支援 [9] など、対戦相手以外のさまざまな用途を目的としたAI技術が研究されている。

及川らはテトリスを対象として、初心者がつまづく壁とされる「T-spin」という特定技能を習得するための、簡略化した部分的問題を生成するシステムを構築した [10]。また、近藤らは格闘ゲームにおいて、コンボと呼ばれる正確なタイミングでのボタン入力を要求される操作技能を、入力タイミングを時系列のパターンと捉えそれに合わせた音を提示することで支援するシステムを構築した [11]。これらの研究のように、教えるAIという研究の一環として、初心者を対象とした上達支援システムなどの研究がさかんにおこなわれている。

格闘ゲームは近年のesportsブームにより、盛り上がりを見せているゲームジャンルのひとつである。様々な大会が開催されると同時に、動画配信サイトを経由して大会の様子が配信され、格闘ゲームをプレイしたことがない人でも手軽に観戦を楽しむことができる。中には観戦をするだけでなく、自分でプレイしたいと考える人もいる。しかし格闘ゲームは、初心者がつまづく壁となるような要素が多く存在しており、運が絡みにくいゲーム性なども相まって初心者と経験者が対戦を行った際には、初心者は何が起きているか理解する前に倒されてしまうことが多い。これではゲームを楽しむこともできず、モチベーションも低下する一方

である。これらの要因により、新規にプレイした初心者が定着しないということが発生している。esportsとして取り上げられるような競技シーンなどがどれだけ盛り上がるろうとも、新規の初心者が見込めないかぎり、ゲーム全体のプレイ人口は減少する一方であり、最終的に衰退してしまうことは想像に難くない。これは格闘ゲームだけでなくあらゆるジャンルにおいても共通のことである。初心者の定着を促す支援法などが確立されれば、競技シーンのみならずプレイ人口が増えそのジャンル全体が活発となる。ジャンル全体が活発となることで、プロゲーミングチームが新たに格闘ゲームに参入したりプレイヤーや大会へのスポンサーが増加するなどにより市場規模が広がり注目が集まる、そして新規のプレイヤーを呼ぶといった良い循環が発生しうる。そのためにも、初心者が定着しづらいという現状は格闘ゲームというジャンル全体においてもあまり好ましい状態ではない。

格闘ゲームへの初心者の定着を妨げる要素には、“運が介入しない”、“キャラクタ毎の技の知識が必要”、“事前にコンボの練習が必要”、“ハンデ文化が存在しない”、“リアルタイム性のためじっくり思考できない”、“独特な操作性”などが存在する。我々はその中でも“読み合い（駆け引き）”という要素が、初心者の定着を大きく妨げているのではないかと考えた。

格闘ゲームにおける“読み合い”とは、人と人との間で行われる駆け引きであり、相手が次にどのような行動を選択するのかを予測し、それに応じて互いが互いの行動を決定することである。読み合いは対戦の中で非常に多く発生するものであり、場面場面でどのような読み合いが行われているのかを把握しなければ、対戦に勝利することは難しい。読み合いと一言でまとめて呼称してはいるものの、実際に発生する読み合いは、残りの制限時間やキャラクタ同士の距離ごとなどゲームの状況に応じてさまざま存在する。それぞれの読み合いにおいて両方のプレイヤーがとることができる選択肢は、細かいテクニックを含めると数多く存在し、それに加えキャラクタ間の距離や行動を選択するタイミングなどの条件によって結果が変化する。条件を指定した両キャラクタの行動の組み合わせであれば、どちらのキャラクタにどれ程のダメージが入るかなどのデータはゲーム内で確認可能ではあるが、実際の対戦の際に考えるべき条件の組み合わせは非常に多く条件ごとにどのような混合戦略がナッシュ均衡となるかを覚えきくことは困難である。そのため癖や偏りなどから相手の行動を予測する“読み”を行うことにより、相手の想定される行動を限定し、それに対応する形で現実的に有力な選択肢から行動を選択することが有効となる。これは両プレイヤーに言える事であるため、“読み合い”が発生する。なお、読み合い自体は格闘ゲーム特有のものというわけではない。野球やサッカーなど他のゲームはもちろん、株取引などの勝負事以外でも、日常における様々な場面で発生するものである。

格闘ゲームの対戦において必要不可欠な要素である読み合いだが、コンボの練習やキャラクタの性能といった知識の習得などと比較し、対戦相手が存在しないと練習を行うことができないという問題が存在する。しかし、弱いAIプレイヤーなどの対戦相手との実戦を通じて読み合いの練習を行おうとしても、前述したリア

ルタイム性や独特な操作性など、読み合いを行う前段階で初心者がつまづく要因が多く、それらを解決しなければならないという別の問題が生じてしまう。そこで我々は、読み合いにのみ焦点を当て初心者が集中して読み合いの習得を行うことができるシステムを構築することを目指す。

本研究では格闘ゲームの読み合いに関し、初心者がそれを習得するための支援システムの構築に取り組む。前述した通り、実際のゲーム環境では読み合いのみを学習することは困難である。そこで読み合いのみの学習を行うために、読み合いに焦点を当てたゲームを用いる。具体的な手法としては、格闘ゲームの要素の中でも、読み合いに関して必要な要素のみを取り込み、それを重点的に練習することを可能としたターン性のゲームを作成する。それを使用して読み合いを習得するためのカリキュラムを作成、これらを通じて初心者プレイヤーに読み合いを習得させることを目指す。

各章の構成と内容は次のようになっている。第2章では研究対象とする格闘ゲームおよび読み合いに関する紹介、第3章では部分問題を題材とした先行研究や格闘ゲームの初心者支援における先行研究の紹介、第4章では本研究における提案手法の説明、第5章では第1回被験者実験と実験結果、それらを通じて得られた改善すべき点の説明、第6章では第2回被験者実験と実験結果、第7章では本研究のまとめを述べる。

## 第2章 対象ゲーム

4章で述べる提案手法は、ボードゲームなど他のゲームジャンルに関しても参考となる可能性はあるが、実際に作成したカリキュラムに関しては格闘ゲームの読み合いを想定したものである。そのため本章では、本研究の対象である格闘ゲームと格闘ゲームにおける読み合いに関して説明を行う。

### 2.1 格闘ゲーム

#### 2.1.1 格闘ゲームの概要と基本的なルール

格闘ゲームは1対1で遊ばれる対戦型のデジタルゲームである。格闘ゲームという用語は、古くは「熱血硬派くにおくん」や「ファイナルファイト」のようなベルトスクロールアクションゲームと呼ばれるゲームジャンルに適用されていた呼び名であった。1991年に稼働開始された「ストリートファイターII」のブームを皮切りに、格闘ゲームという用語は、一定範囲のステージ内でプレイヤーがキャラクターを操作し、相手キャラクターと対戦を行うゲームのジャンルを指す用語となった。格闘ゲームは、現在注目を集めている esports 競技 [12] の1種目であり、国内外様々な場所で格闘ゲームの大会が開かれている。

プレイヤーは制限時間内であれば、後述するダウン状態などの一部場合を除いて移動や技の使用などキャラクターを操作し様々なアクションを起こすことができ、リアルタイム性かつアクション性の高いゲーム性となっている。

ゲームごとにシステムの違いなどが存在するが、おおむね共通する基本的なルールは以下の通りである。各プレイヤーが操作するキャラクターには体力が設定されており、キャラクターごとに設定された技を使用し互いの体力を減らし合い、制限時間内に体力を0にしたプレイヤーもしくは制限時間終了時点で体力が多いプレイヤーが勝者となる。

格闘ゲームにはストリートファイターシリーズ [13] などに代表される2Dのものと、バーチャファイター [14] などに代表される3Dのものと2種存在する。2Dや3Dというのはキャラクターの移動可能な軸である。2D格闘ゲームはキャラクターの移動が前後とジャンプというのに対し、3D格闘ゲームの場合はキャラクターを前後、ジャンプに加えて奥と手前に移動させることが可能である。2Dと3Dとでは本研究で扱う読み合いに関して違いがあり、本研究では図2.1に示す「ストリートファイターV」のような2D型格闘ゲームを研究の対象とする。



図 2.1: ストリートファイター V 対戦画面 (出典 ストリートファイター V 公式サイト [1])

### 2.1.2 初心者が直面する障壁

格闘ゲームはリアルタイムでゲームが進行し、互いのプレイヤーは同時に自身のキャラクターを操作する。勝利条件は相手キャラクターの体力を0とすることだが、通常技の他に設けられている必殺技と呼ばれる技は、通常技と比較し大きく相手の体力を減らすことができる。しかし、必殺技を使用するためにはコマンド入力が必要される。複数の必殺技を連続で使用できるコンボにいたっては、より正確かつ素早いコマンド入力が必要される。必殺技ひとつ出すのにも複雑な操作が必要されることに加え、ガードの際にレバーを相手と逆方向に入力するといった他のアクションゲームと違う独特な操作方法が存在する。このように初心者が慣れない操作性の中、正確にコマンド入力を行うことは非常に難しい。

さらに運が絡むことの無いゲーム性や、囲碁や将棋などと違い格闘ゲームのコミュニティ内にハンデを付ける文化が存在しないことなど、初心者と経験者の対戦の際に初心者が勝利する可能性を著しく低くする要素が存在する。また、初心者は経験者に何が起こったか分からずに倒されることが多く、初心者と経験者の対戦は初心者にとって練習にもなりづらい。初心者は経験者との対戦で何もできずに敗北し、自身の上達を確かめることができずゲームをプレイするモチベーションが無くなってしまう。これにより初心者がゲームをやめることで、初心者の人口が少なくなり初心者同士の対戦が難しくなるという負の循環が生まれてしまう。

このように初心者にとって定着までに様々な障壁が存在する中、我々は最も大きい障壁のひとつに“読み合い”があると考えられる。読み合いは、読み合いを行うために行動の有利不利を知識として知っておく必要があることや前述の操作などに慣れ、自身が行いたい行動がとっさの判断で行えることなどが前提となっているため、初心者が練習を始めづらい。また、コマンドが難しく技を思うように使用

できないといった問題は自身の操作練度が不足しているとすぐに理解することが可能であるが、読み合いに必要な知識が不足している場合などには、自身がなぜ負けているのか気づきづらいなどの問題点が存在する。なお読み合いの詳しい説明は次節にて行う。

これらの要素が相まって格闘ゲームは初心者にとって定着しづらいゲームジャンルとなってしまっている。

## 2.2 格闘ゲームの読み合い

本研究における読み合いとは、対戦相手が互いに相手の行動を推測し、その推測を情報に含めて、更に相手も自分と同様に自身の行動を推測してくることを踏まえて自身の行動を決定することを指す。片方のプレイヤーが相手の選択を推測し自身の手を決定することを“読み”と呼び、この行為を互いのプレイヤーが行うため“読み合い”と呼ぶ。読み合いはゲームのみならず日常生活の様々な場面において発生しうるものである。

### 2.2.1 読み合い

前述したとおり読み合いとは、対戦相手が互いに相手の選択を推測し、その推測を情報に含めて自身の行動を決定することを指す。互いに行動を推測するということもあり、相手に自身の手を推測させないように自身の行動を散らす行為や、逆に同じ行動を連続で選択することで相手に次も同じ行動を取ると推測させるなどの駆け引きが生じるが、それらも全て読み合いの範疇である。

具体例として連続でプレイするじゃんけんを考える。今までのじゃんけんの結果から相手は同じ手を連続して出さないという傾向があると考え、前回の相手がグーを選択していた場合、相手の傾向から次の手を“読んで”あいこ以上になる確率が高いチョキを選択することは、人間的には自然な考え方のひとつである。それに対し相手側も、今までのじゃんけんであえて同じ手を連続で出さないことで、連続で同じ手を選択しない傾向があるかのようには思わせ、前回はグーだから次はあいこ以上になるチョキを出すのではないかと“読んで”グーを選択するということもあり得る。このような例も人間的に生じる読み合いのひとつである。

このように相手の今までの選択から傾向や癖を推測し、それらを考慮して互いが自身の手を決定する事で読み合いが成立する。

### 2.2.2 格闘ゲームにおける読み合い

格闘ゲームにおける読み合いは、立ち回りにおける読み合いや起き攻め状況における読み合いなどが存在する。前者の立ち回りにおける読み合いは、両方のプ

レイヤが非ダウン状態の際に発生する読み合いであり、ゲーム中ほぼ常に発生しているものである。具体的には、残りの制限時間やキャラクタ間の距離、使用できる技の射程やダメージ量などから、どの技を使用するのか、自分が使いたい技の射程になるまで近づくのか、相手の技を警戒して距離を離すのかなどの読み合いが発生する。

後者は“起き攻め”と呼ばれる状況における読み合いであり、どちらかのキャラクタがダウン状態という操作不可かつ相手からダメージを受けない状態から、復帰する際に発生する読み合いのことを指す。起き攻め状況の読み合いは立ち回りにおける読み合いと違いキャラクタ同士が密着していることから、キャラクタ間の距離を考える必要が無い、遠距離で効果的な技を考慮しなくてよいため使用する技の種類が少ないなど立ち回りにおける読み合いとの違いがある。

また行動を選択するタイミングに関して、立ち回りにおける読み合いの場合には、格闘ゲームがリアルタイム性を持つゲームということもあり、プレイヤーが任意のタイミングで行動を選択することが可能であるが、起き攻め状況における読み合いの場合には両方のプレイヤーがほぼ同時に行動を選択するなどの違いが存在する。プレイヤーがとれる行動を簡略化することで3すくみの関係と捉えられることなどから、じゃんけんに似た読み合いと考えることができる。行動の簡略化に関しては次項にて述べる。

### 2.2.3 本研究で取り扱う読み合い

本研究では2.2.2項で述べたもののうち、起き攻め状況における読み合いに焦点を当てる。

2.1.1項で述べているように、格闘ゲームはリアルタイムで進行しプレイヤーはその間キャラクタの操作を行うことができるゲームである。そのため素早く状況が切り替わる中で、プレイヤーはダウン状態など一部状態を除いて任意のタイミングで行動を選択するが、双方のプレイヤーがほぼ同時に行動を選択する状況も存在する。本研究にて取り上げる起き攻め状況の読み合いもその状況の1つである。そのような同時着手に近い状況での行動選択は、ゲーム理論を用いて扱うことができるため、理論的には最適な混合戦略が存在する場合が多い。しかし最適な混合戦略を事前に想定し覚えておこうにも、細かなテクニックを含めれば互いに選択可能な行動の組み合わせは非常に多く、さらに残り時間や体力などの諸条件ごとに混合戦略を考えなければならないことを考慮すると、正しい混合戦略を知ること自体が困難と言える。そのため、相手の癖や行動の偏りなどから相手の手を推測、すなわち読みを行うことで相手の偏った行動などに対応して行動を選択する必要が生じる。これは双方のプレイヤーに当てはまることであり、このことから起き攻め状況における読み合いが発生する。

起き攻め状況での読み合いにおいて、ダウンした側のプレイヤーがダウン状態から復帰した際に取れる行動は大別して3種に分けることができる。それぞれ“打

撃”と“投げ”と“防御”の3種であり，ダウンを取った側のプレイヤーも同様に3種の行動から自身の行動を選択する．これらの行動は，打撃が投げに勝利し，投げが防御に勝利し，防御が打撃に勝利するという3すくみの関係性となっているため，じゃんけんのような3すくみの読み合いと捉えることが可能である．しかしじゃんけんとは異なる点として，これらの行動はそれぞれ3すくみに勝利した際に与えるダメージ量の違いや，その後の状況の有利不利などが存在する．プレイヤーは制限時間や現在のキャラクタの位置などのゲームの状態と，それぞれの行動で得ることができるリターンや相手の行動で負うリスクを加味して読み合いを行う必要がある．

以降本論文では読み合いという単語を“起き攻め状況における3すくみの読み合い”の意で使用する．また，読み合いが発生していることを理解し，相手の行動の選択頻度などから対戦相手の癖や行動の偏りを発見しそれにつけ込みつつ，相手にはつけ込まれないよう手を散らしたり確率的な行動ができることを読み合いを習得した状態とする．

## 第3章 関連研究

教育の場において難しい問題を教えるために様々なアプローチがなされてきた。問題の簡略化もそのひとつである。Polyaは一般にある問題に行き詰った際に、その問題よりも易しい問題や関連した問題を解くことやそれらの問題を自分から作り出すことにより、行き詰っていた問題に関して新たな側面や考え方を得ることで解決がより容易となることがあると指摘している [15]。本章では問題の簡略化というアプローチを使用した学習の支援に関する研究の一部を紹介する。

福田らはピアノの演奏練習に際して学習者の演奏ミスに合わせて簡略化した楽譜を提示することで、学習者のレベルに合わせた練習を行うことができるようになるシステムを提案した [16]。さらに提案したシステムでは、蓄積した学習者の演奏ミスと楽譜の特徴量を合わせたデータを入力とした学習者のモデルを構築することで、学習者が初めて演奏するような楽曲に対しても楽譜の簡略化を行うことができる。

中尾らは専門教育を受ける留学生を対象とした、文構造の簡略化による読解支援を提案している [17]。初級日本語学習者が文章を読む際に文中の単語を語意レベルで理解することはできても、日本語ネイティブに向けて書かれた複雑な文章構造により文全体の内容の論理的な理解が難しいという点から、初級学習者の既知の文法構造知識の範囲で文章を簡略化するという研究を行った。簡略化した文章の提示方法などインターフェイス関係の改善すべき点は残ったが、文章の簡略化に関しては読解が簡単になったと、概ね好評を得た。

ゲームにおける簡略化は囲碁の詰碁や将棋の詰将棋など、古くから行われている。これらは単にパズルとして楽しむだけでなく、実際の対局と比較して短時間でプレイできる点や、一人でプレイすることができる点などから、棋力上昇のための練習問題としても用いられる。これらと同じような簡略化したゲームを自動生成し、プレイヤーのレベルに合わせて問題を提示することで、特定技能の習得に関する学習の支援を狙った試みも存在する。高橋らは「ぷよぷよ」において、初心者が連鎖という技能を習得することを補助するための問題を自動生成することを目的とした研究を行った [18]。ぷよぷよにおいても特定状況を取り出して問題としたなぞぷよというものが存在し、「対戦相手の妨害が無い」「既にある程度連鎖の形となるように盤面が組まれている」といった点で実際のゲームプレイよりも簡略化された状況と言える。高橋らはなぞぷよを自動生成するだけでなく、盤面の特徴量と被験者実験で得られた評価から回帰モデルを作成することで、難しさと面白さと役立ち度の観点からプレイヤーのレベルに合った問題の提示を行うシステ

ムも提案した。

## 第4章 提案手法

本章では、格闘ゲームにおける読み合いの考え方を初心者へ伝えるために自作した、“拡張じゃんけんゲーム”と“カリキュラム”の2つについて説明を行う。あるゲームの部分ゲームを取り出し問題化することは、初心者にとって特定技能を習得させるためによく用いられる手法であり、具体例は将棋における詰将棋などである。格闘ゲームにおいても同様に、習得したい技能に関して、その技能を部分問題とした課題を課すことで習得を支援する機能を搭載したゲームが存在する。図4.1はその一例である。画面の左に課題となる行動が提示され、成功すると上部の歯車がオレンジ色に変化する。全5回の挑戦の内、過半数成功すればその課題を完了したこととなる。

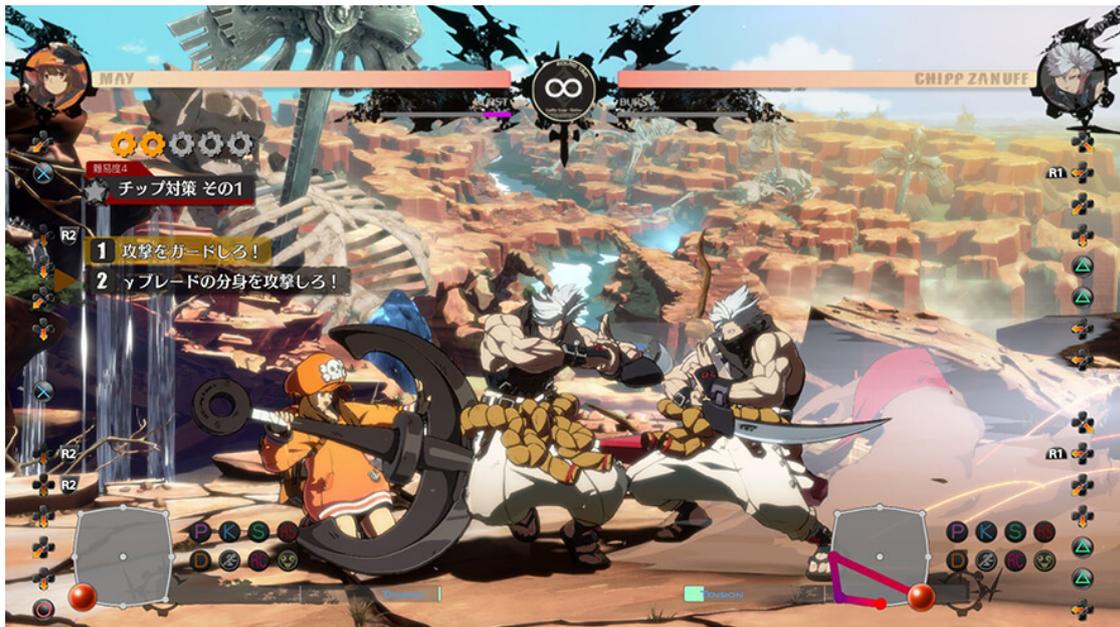


図 4.1: GUILTY GEAR -STRIVE- ミッションモード画面 (出典 GUILTY GEAR -STRIVE-公式サイト [2])

課題の中には読み合いに関するものも存在する。しかし、読み合いそのものではなく読み合いに使用できるテクニックを説明するものが多く、完全に読み合いのみに焦点が当ててあるわけではない。その上、格闘ゲームの部分問題をプレイする場合には、操作に不慣れで思うように動かさない、リアルタイムでゲームが

表 4.1: 格闘ゲームの要素と自作ゲームでの扱い

格闘ゲームの要素	自作ゲーム
リアルタイム性	簡略化 (同時着手)
多様な技や行動	簡略化 (3種類)
技ごとの複雑なコマンド (アクション性)	簡略化 (選択制)
行動ごとの相性	簡略化 (3すくみ)
行動ごとのリターンの差	同様
状況による行動のリスクリターンの変化	同様
キャラクタごとの体力	同様 (残り歩数と表現)
制限時間	同様 (ラウンド制, 最大 20 ラウンド)

進行するため思考が追い付かないなど、2.2.2項で示した読み合いを行う前の段階の障壁が部分問題においても存在するという問題点がある。

初心者を読み合いの考え方を伝えるために、これらの問題点を解決した格闘ゲーム初心者が読み合いのみを体験することができるゲームがあることが望ましい。そこで格闘ゲームの要素の中から、読み合いを行う前段階での障壁となる要因を簡略化することで取り除いた、読み合いを重点的に体験できるゲームを作成する。格闘ゲームの要素について格闘ゲームと自作ゲームを比較した表を表 4.1 に示す。

自作ゲームは前述の通り読み合いのみを体験することを目的として作成する。そのため読み合いに関係が薄い“リアルタイム性”や“アクション性”などは同時着手の選択制ゲームとすることで排除する。格闘ゲームにはキャラクタごとに多様な行動が存在するが、自作ゲームではそれらを簡略化し3種類とする。格闘ゲームの要素の中でも読み合いに深く関連するものである“行動ごとの相性”や“行動ごとのリターンの差”、“状況による行動のリスクリターンの変化”などは自作ゲームにおいても同様に再現する。なお行動ごとの相性に関しては、プレイヤーが可能な行動を3種に簡略化する関係上3すくみ関係とする。またリアルタイム性は排除したが、“制限時間”は読み合いに関連する要素であるため、自作ゲームではラウンド制を導入することによりゲームの制限時間を再現する。以降自作ゲームを“拡張じゃんけんゲーム”と呼称する。

拡張じゃんけんゲームは読み合いを体験できる環境を提供するものであり、それのみでは読み合いの考え方を伝えることはできない。読み合いの考え方を伝えるためには、関連する学習内容を用意しなければならない。また効果的に学習内容を伝えるためには、用意した学習内容をある程度のまとまりに分割し理解しやすいように順序立てることやそれぞれのまとまりに関して概要を説明するなど、学習の支援があることが望ましい。本研究ではこれらの学習内容と学習の支援を統合してまとめたものを以後カリキュラムと呼称する。カリキュラムは前述の拡張じゃんけんゲーム上で実行されるものであり、読み合いの考え方を理解するため

に必要な要素を段階的に伝えるものとなっている。

## 4.1 拡張じゃんけんゲーム

拡張じゃんけんゲームは、2.2.3項で示した3すくみの読み合いや各行動のリターンの違いに焦点を当て、アクション性とリアルタイム性を簡略化した、じゃんけんをベースとした自作のゲームである。以下に基本ルールを示し、図4.2に拡張じゃんけんゲームの対戦画面を示す。

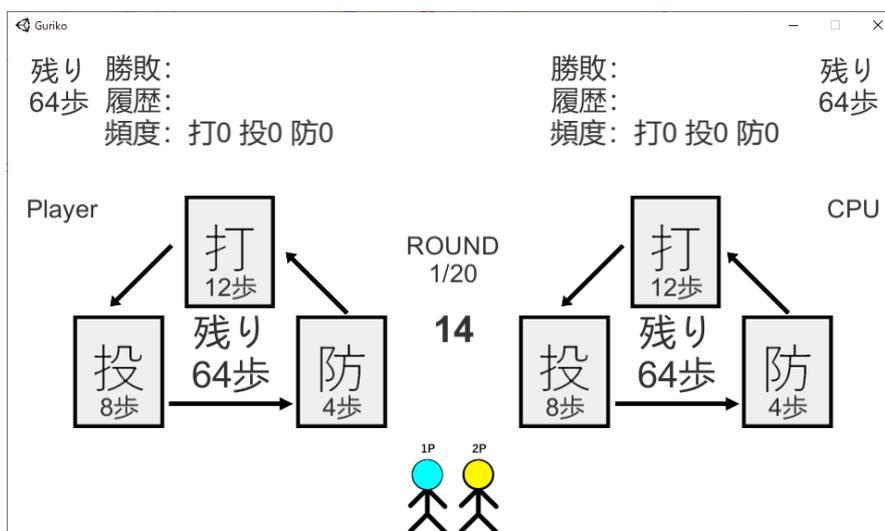


図 4.2: 拡張じゃんけんゲーム 対戦画面

- 各プレイヤーは選択可能手から手を選択し3すくみの勝負を行う。1度の試行を1ラウンドと呼び、1ゲームで最大20ラウンドの勝負を行う。
- プレイヤーは互いに64歩という“残り歩数”を有し、じゃんけんの勝利時は減少、敗北時は増加する。“残り歩数”の合計は128から変わることはなく、じゃんけんの勝者が減らした歩数分の数値が、敗者の“残り歩数”に足される。
  - 残り歩数を0にしたプレイヤー、もしくは全20ラウンド終了時点で相手より残り歩数が少ないプレイヤーが勝者となる。
- 各プレイヤーは“打撃”と“投げ”と“防御”の3種から手を選択することができ、“打撃”は“投げ”に勝利し、“投げ”は“防御”に勝利し、“防御”は“打撃”に勝利する3すくみとなっている（以降本論文ではこれら3種の選択可能手のことを手と呼ぶ）。表4.2に、これらの手と後述する特殊打撃と特殊投げ、それら全ての手の効果をまとめた表を示す。

- 打撃は 12 歩，投げは 8 歩，防御は 4 歩ラウンド勝者の残り歩数を減らし，敗者側を増やす。
  - 防御でラウンドに勝利したプレイヤーは，次ラウンドの打撃が“特殊打撃”に変化する。“特殊打撃”は打撃と投げの双方に勝利し，ラウンド勝者の残り歩数を 12 歩減らし，敗者側を 12 歩増やす。
  - 残り歩数が 84 歩以上となり，不利な状況に立たされたプレイヤーの投げは“特殊投げ”に変化する。“特殊投げ”は通常の投げと同じく防御に勝利するだけでなく，勝利時には両プレイヤーの残り歩数を入れ替え形勢を逆転させることが可能である。
- プレイヤは手の選択時に一定の思考時間が与えられる。具体的な思考時間は後述する。
  - 各プレイヤーは自分と相手の過去 3 回までの選択手とラウンドの勝敗，過去に選択した手の全頻度を確認することが可能である。

表 4.2: 拡張じゃんけんゲームにおける選択可能手とその効果

手の名前	勝利できる相手の手	勝利時の効果
打撃	投げ	ラウンド勝者の残り歩数を 12 歩減らし，敗者側を増やす。
投げ	防御	ラウンド勝者の残り歩数を 8 歩減らし，敗者側を増やす。
防御	打撃	ラウンド勝者の残り歩数を 4 歩減らし，次のラウンドの打撃を特殊打撃にする。敗者側を増やす。
特殊打撃	打撃，投げ	ラウンド勝者の残り歩数を 12 歩減らし，敗者側を増やす。
特殊投げ	防御	両プレイヤーの残り歩数を入れ替える。

#### 4.1.1 特殊打撃と特殊投げ

特殊打撃と特殊投げは，前述した格闘ゲームの要素である，“状況による行動のリスクリターンの変化”を，拡張じゃんけんゲーム上で再現することを目的として導入されたものである。拡張じゃんけんゲームにおけるリスクリターンとは，選択した手による残り歩数の変化量として捉えることが出来る。

特殊打撃は打撃と投げの両方に勝利することができる手であるため，通常の打撃よりも強い手と考えることができる。そのため，プレイヤーが特殊打撃を使用可能な状況は，特殊打撃が使用できない状況よりも有利と言える。選択可能な手のリターンが両方のプレイヤーで異なるため，通常の状況とは異なる読み合いの発生

が期待される。さらに、特殊打撃を使用可能なプレイヤーが有利になることから、特殊打撃を使用可能にする防御の手には残り歩数を4歩減少させる以上の価値があり、プレイヤーは手の選択の際にこの価値も織り込む必要が生じる。なお、この特殊打撃は、格闘ゲームにおける「フレーム差」の概念を再現したものである。

特殊投げは、勝利した際に形勢逆転することができるリターン大きい手であるため、プレイヤーが特殊投げを使用可能な状況は、残り歩数では不利だが選択可能な手のリターンでは有利な状況と言える。こちらも特殊打撃と同様に、選択可能な手のリターンが両方で異なることから、通常の場合とは異なる読み合いが発生することが期待される。なお、この特殊投げは、格闘ゲームにおける「画面端での入れ替え投げ」を再現したものである。

## 4.2 カリキュラム

本カリキュラムは4.1節にて記した拡張じゃんけんゲーム上で実施される。拡張じゃんけんゲーム自体はあくまで読み合いを体験するためのゲームに過ぎず、読み合いの考え方伝えるために教材となる対戦相手が必要だと考える。対戦を通じて読み合いの考え方を伝えるために、読み合いに関する伝えたい内容を定め、プレイヤーに提示し、内容に適した行動をとる対戦相手と、適切な順番で対戦を行わせる。具体的には、典型的な相手から始め徐々に複雑な行動をとる相手との対戦するような順番が望ましい。さらに、適宜ヒントを提示することや、プレイヤーが理解するまで対戦を繰り返させることなどの支援も必要だと考える。

これを実現するため、伝えたい読み合いについての考え方を複数選定し、各考え方を1つのタームとして、複数タームで構成されるカリキュラムを設計する。タームごとに、そのタームで伝えたい要素に関する行動をとる複数の対戦相手（ルールベースのコンピュータプレイヤー、以降CPUと呼ぶ<sup>1)</sup>）を用意する。カリキュラムを構成するタームの順番と、ターム内で用意した対戦相手との対戦の順番は、前述の通り、典型的なものから始めて徐々に複雑なものとなるようにする。ヒントとしては2種類用意し、タームのヒントとしてタームごとの伝えたい要素を、対戦相手ごとのヒントとして対戦相手ごとの着目すべき箇所なども提示する。また、対戦相手ごとに、修了条件というものを設定し、これを満たすまで対戦を繰り返させる。なお、修了条件に関する詳細は4.2.5項にて詳細を述べる。

以上をふまえて、カリキュラムの構成と、カリキュラムと各タームの進め方の概略図を図4.3に示す。カリキュラムは表4.4に示すタームを順番に進めていく。ターム内の進め方に関しては4.4に詳細を示す。ターム内には複数の対戦相手が存在し順番に対戦を行う。対戦相手ごとの修了条件を満たしたかの判定を行い、修了するまで同じ対戦相手との対戦を繰り返す。

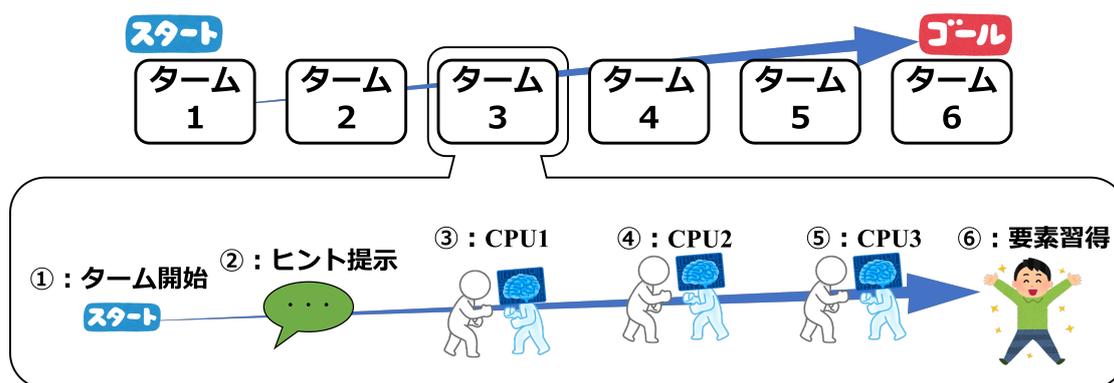


図 4.3: カリキュラムと各タームの進め方の詳細

<sup>1)</sup>CPU は中央処理装置 (Central Processing Unit) の意味を持っており、コンピュータプレイヤーの意として使用するのには紛らわしい場合もある。本論文ではその混同は避けられるためゲーム業界の慣例にならいコンピュータプレイヤーの意として CPU という単語を使用する。

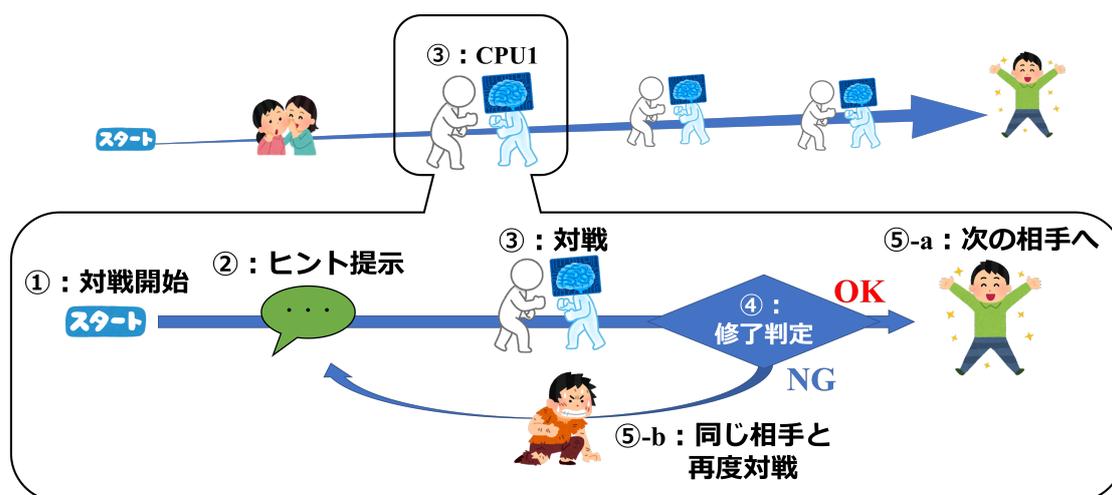


図 4.4: CPU との対戦の進め方

#### 4.2.1 カリキュラム策定のための予備実験

カリキュラムを策定するにあたり，どのような対戦相手の行動規則がプレイヤーにとって把握しやすいかを調べるための予備実験を行った．具体的な実験内容としては，被験者に複数の対戦相手と対戦を行わせ，どの程度のラウンド数で対戦相手の行動規則を概ね把握していると我々が判断するかを調べた．被験者の人数は一人であり，一般の人と比較してある程度ゲームをプレイしており，ゲームに詳しい人物に協力を仰いだ．なお，本システムが支援を行う対象は読み合いを知らない初心者であり，予備実験の被験者はゲームに詳しくれば詳しいほど良いというわけではない点に注意する必要がある．初心者にとって難易度が高い要素が，この被験者の評価からでは判断できない恐れがあるためである．

表 4.3 に被験者が対戦を行った CPU の行動規則と被験者が対戦した CPU の行動規則を把握したと我々が判断するまでに要したラウンド数を示す．なお CPU 番号は被験者が対戦を行った順に振っており，強さや分かりやすさなどを考慮せずにランダムに決定した順番である．1 ゲームの最大ラウンド数は 20 であり，20 を超えた数値は複数回のゲームを要したことを示している．各 CPU との対戦の際，被験者には CPU がどのような行動規則で手を選択しているかを予測しながら対戦を行うように伝えた．また，ゲームに取り組むにあたり，行動規則の把握の為に対戦を行うのではなく，対戦に勝利することを目的としたうえで，行動規則を予測しながらプレイするように伝えた．行動規則の把握に関しては，各ラウンドの終了時に聞き取りを行うことで，被験者が CPU の行動規則を把握できているかを判定した．

結果として“投げだけを選択”や“打撃→投げ→防御→打撃...”と周期的に手を選択”のような，ゲーム状況などに関係なく確定的な手を選択する CPU (CPU 番号 5, 6) は，とても早いラウンド数で把握できていると判断した．次に“勝利した

際の利得が最も高い手を選択”のような、現在の盤面をふまえた上で手を選択するCPU（CPU番号1, 4）は、前述の確定的なCPUと比べて少し多くのラウンド数を要した。最後に“選択した手の頻度が揃うように回数が少ない手をランダムに選択”のような、ランダム性を持つものや被験者の手の選択履歴に応じて手を選択するなどの、現在の盤面以外の要素を参照するようなCPU（CPU番号2, 3, 7）には多くのラウンド数を要した。

表 4.3: 予備実験で使用したCPUの行動規則

CPU 番号	行動規則	把握までのラウンド数
1	勝利した際の利得がもっとも高い手を選択	9
2	最初はランダムで決定し、 その後はプレイヤーの1ラウンド前の手を真似る	17
3	選択した手の頻度がそろうように、 回数が少ない手をランダムに選択	23
4	勝利した際の利得がもっとも高い手を選択し、 相手が特殊打撃か特殊投げを使用可能な場合には、 その手に勝利する手を選択	9
5	投げだけを選択	4
6	打撃→投げ→防御→打撃→... と周期的に手を選択	7
7	プレイヤーの最頻手が1つの場合はそれに勝利する手を、 最頻手が2つの場合はそれらに負けない手を、 最頻手が3つの場合はランダムに選択	27

## 4.2.2 予備実験から得た考慮すべき要素

4.2.1 項の結果をふまえ、カリキュラムの策定に際して以下の3点が必要だと考えた。

- (a) CPU との対戦順は、単純な行動規則を持つものから徐々に複雑なものとする必要がある。
- (b) プレイヤの手の履歴など、プレイヤ自身にまつわる要素でCPUが手を決定していることに気付かせるため、ヒントなどを提示する必要がある。
- (c) 行動規則を把握できたかどうかの客観的な判定の必要がある。

(a) に関して、予備実験の際の対戦順はランダムで対戦を行ったが、後半になるにつれ、難解な行動規則でCPUが手を選択していると被験者は考えていた。表4.3のCPU6などに対し、“自分の選択した直近の手に勝利する手を選択する”といった、見かけ上の手の選択は同じであるが、より難解な行動規則を推測してしまった。この回答は見かけ上の手の説明として間違いとは言えないが、実際のCPU6本来の行動規則とは異なるため、本来の行動規則を推測するまでに続けてもらう余計なコストがかかった。このような不要な深読みを防止することと、段階ごとの要素習得を容易にすることを目的とし、カリキュラムを構成するタームやターム内のCPUは概ね簡単な順に並べることとした。

(b) に関して、被験者は“人間プレイヤの手をもとに、自身の手を決定するようなCPU”に対して、CPUの行動規則を把握するのに時間がかかるという結果を得た。具体的には表4.3のCPU5やCPU6などの手の選択が変化しない単調なCPUに関しては、被験者は少ないラウンド数で行動規則を把握することができた。対してCPU7のような、人間プレイヤの手を参考に手を変更するようなCPUに関しては、行動規則の把握までに比較的多くのラウンドを要した。(a)と関連し、“自分の手をもとに手を決定するような行動規則”を持つCPUに関しては、比較的高度な内容であると判断し、カリキュラム後半のタームに配置した。

(c) に関して、本予備実験では被験者に対してラウンドごとにヒアリングを行うことで行動規則の把握ができているかの判断を、著者が主観的に行った。しかしこの方法では、客観的に判定を行うことができない。また、被験者数を増やした場合に、各被験者に対して都度ヒアリングを実施することは実験コストの面から見ても非現実的である。これらの点から、客観的かつ自動的にプレイヤが行動規則を把握することができているか判定を行う方法が必要である。

表 4.4: タームの全容

ターム番号	伝えたい読み合いの考え方	考慮すべき 過去のターム番号
1	確定的な行動を起こす 相手への対処	なし
2	盤面の状態をもとに 行動を起こす相手への対処	なし
3	確率的な行動を起こす 相手への対処	なし
4	過去の勝敗をもとに 行動を起こす相手への対処	3
5	プレイヤーの過去の行動をもとに 行動を起こす相手への対処	2, 3
6	確率的かつプレイヤーの行動をもとに 行動を起こす相手への対処	3, 5

表 4.5: 各タームに関連する読み合いの要素

ターム番号	盤面状況	確率的な行動	過去の勝敗	プレイヤーの行動
1	-	-	-	-
2	✓	-	-	-
3	-	✓	-	-
4	-	✓ (一部のみ) <sup>2</sup>	✓	-
5	✓	✓ (一部のみ)	-	✓
6	✓ (一部のみ)	✓	-	✓

### 4.2.3 カリキュラムの構成

読み合いにまつわる伝えたい要素ごとにタームを設定し、複数のタームの組み合わせでカリキュラムを構成した。4.2.2 項の (a) を踏まえ、カリキュラムの前半から後半になるにつれて、伝えたい要素が、理解しやすい要素から徐々に複雑な要素となるようにし、ターム内の CPU に関しても同様に、典型的な行動規則からだんだんと複雑なものになるように設定した。また (b) を踏まえて、プレイヤー自身にまつわる要素をもとに手を決定する CPU などは後半のタームに配置した。表 4.4 にカリキュラムを構成する各タームで習得させたい要素と各タームで考慮すべき

<sup>2</sup>行動規則の一部であり、主たる行動規則には関係していない

他タームをまとめ、表 4.5 に各タームに関連する読み合いの要素をまとめる。読み合いに関連する要素は、これらの他にも“相手を騙す行動”などが存在するが、表 4.5 に記した要素は、初心者に伝えるべき基本的な要素として選択したものである。

各タームの習得してほしい要素と CPU の特徴を以下に記す。ターム 1 では“確定的な行動を起こす相手への対処”となっており、ゲームの状態などを一切考慮しない単調な行動規則を持つ CPU との対戦を行う。ターム 2 では“盤面の状態をもとに行動を起こす相手への対処”となっており、残り歩数などのゲームの現在の盤面を参照し手を決定するような相手である。ターム 3 では“確率的な行動を起こす相手への対処”となっており、今までのタームと異なり確率で手を変化する。ターム 4 では“過去の勝敗をもとに行動を起こす相手への対処”となっており、前回ラウンドの勝敗をもとに手を決定する。ターム 5 では“プレイヤーの過去の行動をもとに行動を起こす相手への対処”となっており、プレイヤーの手の履歴や頻度をもとに手を決定する。ターム 6 では“確率的かつプレイヤーの行動をもとに行動を起こす相手への対処”となっており、ターム 5 に近い行動規則だが、ターム 5 と違い、全ての場合で手の決定時に確率をもとに手を決定する。ターム 5 はプレイヤーの手の選択の仕方によっては確率によらない確定的な手の決定をする場合がある。

表 4.6: 各タームに設定したヒント

ターム番号	提示するヒント
1	同じ行動ばかりする相手は付け入ることが可能
2	互いの利得を考えることが読み合いで重要
3	相手の行動がランダムの際は自分の損益を小さくすることが重要
4	前の結果が読み合いに関わってくる
5	相手につけ入るために履歴や頻度は重要
6	相手につけ込むだけでなく、つけ込む際にもランダムで手を散らすことでつけ込もうとしたことが読まれた時に対応する

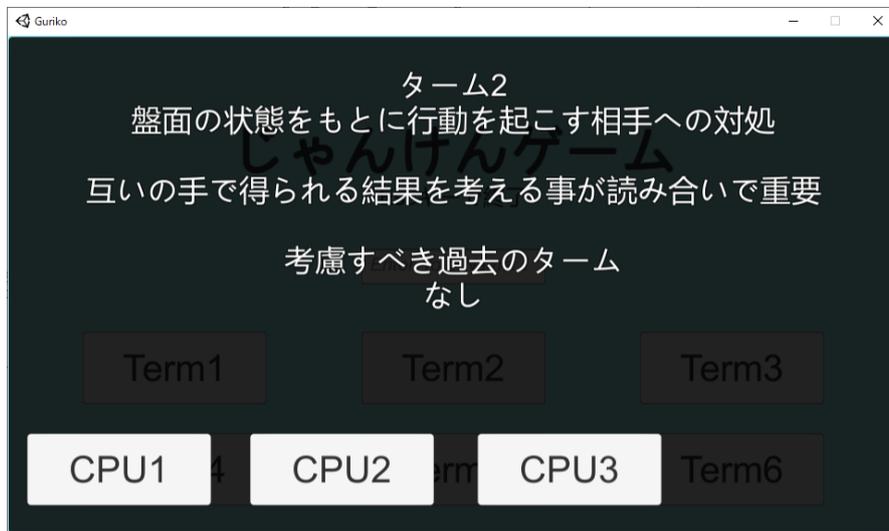


図 4.5: タームごとの対戦する CPU を選択するゲーム画面

#### 4.2.4 ヒント

ただタームごとの CPU と対戦をさせるだけでは、そのタームで伝えたい読み合いの要素が伝わることは難しいと考える。そこで、プレイヤーにそのタームを通じてどのような要素を習得してほしいかを提示し、そのためにはどのようなことに注意しながら対戦を進めていくかなどのヒントを適宜提示することが望ましい。各タームの伝えたい要素とあわせて、ターム全体を通してのヒントを設定し、それらをプレイヤーに提示する。表 4.6 に各タームで提示するヒントも提示する。実際にプレイヤーに提示する際のゲーム画面は、図 4.5 のようになっており、プレイヤーはタームの情報を確認した後、下部の CPU ボタンをクリックすることで対戦に進むことができる。図 4.5 はターム 2 を選択した場合、実際に表示される画面である。また、各タームの CPU に関しても、同じ読み合いの要素に関する行動規則を持っ

ていたとしても、着目すべき点などはそれぞれ異なるため、それぞれにヒントを設定する。プレイヤーには対戦前にそのヒントを提示する。提示するCPUのヒントに関しては付録B.1、付録B.2をそれぞれ参照されたい。なお、プレイヤーにCPUごとのヒントを提示する方法は各実験で異なるため、それぞれ5章と6章にて説明を行う。

#### 4.2.5 行動規則の把握の判定

前述のように、チーム内のCPUは表4.4に示す各チームの伝えたい要素に関する行動規則を設定しているため、行動規則を把握することができればそのチームの要素を習得したと仮定している。しかし、4.2.2項の(c)に挙げた問題点の通り、プレイヤーがCPUの行動規則を把握できているかの客観的判定が必要である。

この問題を解決するため選択式の問題を導入し、行動規則を把握することができたかどうかを判定することとした。選択式問題の選択肢は“相手の次の手の確率分布”となっており、プレイヤーは相手が次のラウンドで、打撃と投げと防御をどのような確率分布で選択するかを、選択肢の中からひとつ選び回答する。このゲームでの選択式問題の正答率を用いて行動規則の把握ができているかを判定する。なお提示する選択肢には必ず正答が存在し、CPUは選択肢以外の確率分布で手を決定することはない。具体的な例を図4.6を用いて述べる。CPUが“打撃のみを選択”という行動規則の場合、CPUの次の手の確率分布は常に打撃10割、投げ0割、防御0割となる。人間プレイヤーには打撃10割、投げ0割、防御0割を含めた複数の選択肢を提示し、その中から回答を選択させる。選択した回答が正解かどうかは人間プレイヤーには開示しない。

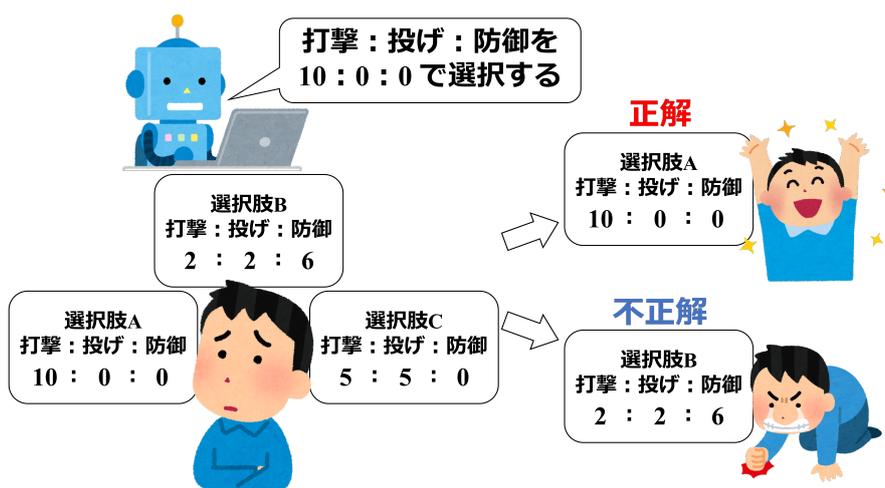


図 4.6: 選択式問題の概略図

選択式問題の正答率に関して各CPUごとに閾値を設け、そのゲームの正答率が閾値以上の場合には、プレイヤーはそのCPUの行動規則を把握することができてい

ると判断する。正答率の閾値はターム毎に異なり、難易度を考慮して設定する。閾値の設定や選択式問題の詳細な仕様は各実験で異なるため、それぞれ5章と6章にて説明を行う。

## 第5章 第1回被験者実験

作成したカリキュラムそのものの有効性や、カリキュラムを構成するタームの順番の有効性、カリキュラムとゲームに改善すべき点が無いかなどを確認するために被験者実験を行った。なお実験は全体で2度行い、本章では1度目のカリキュラムの有効性確認実験（以降実験1と呼称）に関して述べる。5.1節では実験目的について述べ、5.2節と5.3節では実験設定について述べ、5.4節と5.5節では得られた結果と、実験を通じて得られた改善すべき点について述べる。

### 5.1 実験目的

作成したカリキュラムは、タームの順番、ヒントの提示、CPUの選択や順番、制限時間、修了条件の設定など学習に関する様々な要素を含んでいる。しかし、カリキュラムそのものの有効性を検証しなければならないことに加えて、それを構成する各要素がどの程度寄与しているのかを検証することは、実験規模の大きさからして現実的でない。そこで本被験者実験では、カリキュラムそのものとカリキュラムを構成するタームの順番という2点に関して、有効性を検証する実験を行う。検証のための実験方法として以下の3群による比較実験を行う。

- タームの構成順通りにカリキュラムを受ける群（以降カリキュラム群）。
- カリキュラムのCPUとランダム順で対戦を行う群（以降ランダム群）。
- 他群と同時間対人戦を行う群（以降対人群）。

### 5.2 実験設定

本節では、テスト用CPUや群ごとの実験内容など、被験者実験の設定を述べる。本実験に参加した被験者人数は12名であり、5.1節で述べた通りこれら12名を3群に分ける。3群に分ける際に、プレテストとしてテスト用CPU3種と3回ずつ対戦を行い、それらの成績をもとに群内の成績が同程度となるように被験者を群に分けた。なお、テスト用CPUとの対戦順はランダムとなっており、被験者全員がそれぞれ違う順番で対戦を行った。この際に被験者には、どのテスト用CPUと対戦をしているか、その対戦何回目の対戦かといった情報は与えられない。また、成

績に関しては、4.2.5項で挙げた正答率を参照している。実験に使用したテスト用CPU3種を以下に述べる。それぞれのCPUはカリキュラム中のタームの要素を複合したような行動規則を設定した。なおテスト用CPUとの対戦の際には、ヒントの提示は行わない。

テスト用CPU1 プレイヤの直前の手と同じ手か、それに勝利する手を、5割ずつの確率で選択するCPU。(ターム3とターム5を複合したような行動規則)

テスト用CPU2 基本はプレイヤの最頻手に勝利する手を選択する。最頻手が2つの場合は、それらとあいこ以上となる手を選択する。最頻手が3つの場合は、基本は最高利得の手を、自分が特殊手を使えるなら特殊手を、相手が特殊手を使えるならそれに勝利する手を選択する。ただし、前のラウンドが、CPUの敗北かあいこの場合には、プレイヤの直前の手に勝利する手を選択するCPU。(ターム4とターム5を複合したような行動規則)

テスト用CPU3 基本はその手で勝利した際に、最も残り歩数を多く減少できる手を選択する。以前経験した状態の際には、プレイヤが過去の同じ状況で最後に選択した手と、その手に勝利する手を、5割ずつの確率で選択するCPU。(ターム2とターム3, ターム5を複合したような行動規則)

各群が行った実験内容について述べる。カリキュラム群は、ターム1から順に、ターム内のCPUと対戦し、ターム6までの全CPUと対戦を行う(図4.3, 図4.4参照)。なお、次のCPUとの対戦に進むための修了条件は後述する。ランダム群は、カリキュラムに出てくるCPUとランダムな対戦順で対戦を行う。次のCPUに進む条件は修了条件を満たすか、同じCPUと2度再戦をすることである。対人群は、一定時間が経過した後に対人群内で対戦相手を入れ替えながら対人戦を行う。この際に会話などに関する指示は行わない。

各群の実験が終了後に、プレテストと同様の、テスト用CPUと方法でポストテストを実施した。

実験時間はプレテストを30分、実験時間を3時間、ポストテストを30分とし、実験の前後にそれぞれ、実験に関する知識や主観的評価の確認のためのアンケートを行った。実験前の質問では「質問1: 読み合いを知っていたか」「質問2: 読み合いを経験したことがあるか」の2つであり、実験後の質問では「質問3: 読み合いを理解できたか」「質問4: 自分の予測の精度が上がっていると感じるか」を質問した。これらの質問は全て「はい」か「いいえ」での回答とし、あわせてゲームに関する自由記述欄もアンケート用紙に設けた。

## 5.3 実験1におけるカリキュラムの設定

実験1を実施するにあたって、選択式問題の仕様やプレイヤーの思考時間などカリキュラムの設定を本節にて説明する。なお、カリキュラムで使用したCPUの行動規則とヒントに関しては、付録B.1を確認されたい。

### 5.3.1 選択式問題

被験者がCPUの行動規則を把握できているかの客観的判定を行うために、被験者には次ラウンドのCPUの手の確率分布を予測し回答してもらう。実際に実験1で使用した選択式問題のゲーム画面を図5.1に示す。なおこのゲーム画面は実験1で使用したものであり、6.2.2項で述べるように実験2では変更を施している。

選択式問題で正答とする選択肢は“真の確率分布”のみとした。“実際の着手頻度”に基づいた回答は、本来間違いと言うことはできないが、本実験においては、真の確率分布と異なる場合には不正解とした。具体的には、真の確率分布では打撃と投げと防御を6:2:2の割合で選択する行動規則のCPUが、実際には確率の偏りなどにより、7:0:3に近い着手頻度で手を選択したとする。しかしこの場合の選択式問題の回答において7:0:3は不正解とし、真の確率分布である6:2:2のみを正答とした。選択式問題で提示する選択肢はCPUごとに異なるが、選択肢内には必ず真の確率分布が存在するように設定した。

被験者の思考時間と実験時間のコントロールを考えた際に、験者が行う、手の選択時間と選択式問題への回答時間は、合計で30秒ほどが良いのではと考えた。そこで実験1では、手の選択の制限時間と選択式問題への回答時間をともに15秒に設定した。

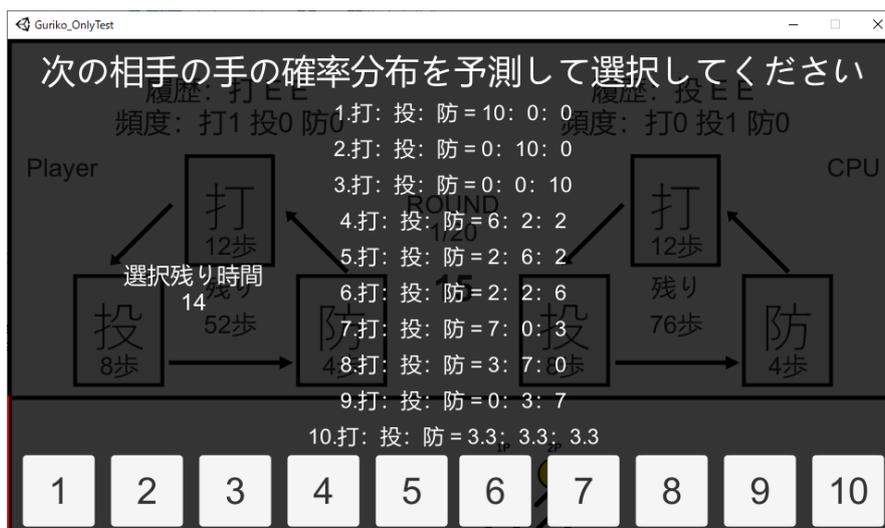


図 5.1: 選択式問題への回答時のゲーム画面（実験2とは異なる）

### 5.3.2 各 CPU の正答率の閾値とそのほか諸条件

4.2.5 項で述べたように，ターム内の CPU ごとにその CPU の行動規則を把握できているかの判定をするための閾値を設定する．プレイヤーの，そのゲームでの選択式問題の正答率が設定された閾値以上であれば，次の CPU との対戦に進むことが可能である．なおターム 3 に関しては，ターム冒頭のヒントを“相手の行動がランダムの際は自分の損益を小さくすることが重要”としており，ヒント通りに損益が小さい手を選んでいるかの確認のため，閾値の他に別の条件も設定している．また，タームが進むにつれて，CPU の行動規則は，ランダム性が絡むようになるなど，複雑なものを設定していることもあり，段階的に閾値を下げている．表 5.1 に各 CPU の修了条件をまとめた表を記す．

表 5.1: 実験 1 における CPU ごとの閾値と条件

ターム番号	CPU 番号	正答率の閾値	そのほか条件
1	1~4	0.8	なし
2	1~3	0.8	なし
3	1	0.75	15 歩以上差をつけて勝利
3	2	0.75	40 歩以上差をつけて勝利
3	3, 4	0.75	ゲームに勝利
4	1~3	0.75	なし
5	1~3	0.7	なし
6	1~3	0.7	なし

### 5.3.3 各 CPU のヒント提示画面

4.2.4 項で述べた通り、チーム全体のヒントの他に CPU ごとにヒントを設定しそれを各 CPU との対戦前に被験者に提示した。実験 1 で使用したゲームの仕様における、ヒント提示の画面を図 5.2 に示す。被験者はキーボードの S キーを押下することで、任意のタイミングで CPU との対戦を開始することができる。なお図 5.2 のゲーム画面は実験 1 で使用したものであり、6.1.5 項で述べるように実験 2 では変更を施している。



図 5.2: 実験 1 の際の各 CPU のヒントを提示するゲーム画面 (実験 2 とは異なる)

### 5.3.4 ランダム群へのヒント提示

ランダム群はランダムに対戦を行う関係上、図 4.5 のチーム選択時のチーム冒頭のヒントが提示されない。そのため、CPU との対戦時のヒントと同じタイミングでチーム冒頭のヒントを提示する。具体的には図 5.3 のような提示方法でプレイヤーにヒントを提示する。なお、図 5.3 上部から、考慮すべき他の対処法までの文言がチーム冒頭のヒント、ヒントより下の文言が CPU との対戦時のヒントとなっている。なお図 5.3 のゲーム画面は実験 1 で使用したものとなっており、6.1.5 項で述べるように実験 2 では変更を施している。

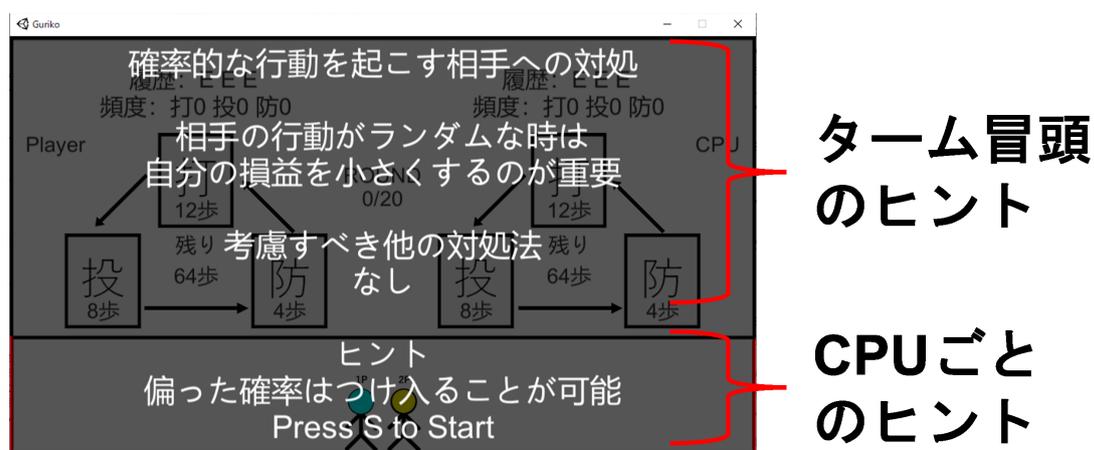


図 5.3: ランダム群にヒントを表示する画面

## 5.4 実験結果

実験に関するアンケートに、「はい」と答えた人数を表5.2にまとめる。なお、質問番号と内容の対応は5.2節を確認されたい。アンケートの中でも質問4は、予測精度の自己評価に関する質問であり、読み合いにおける読みの精度の上昇の自己評価と考えることができるため、カリキュラム自体の有効性の検証という点からも重要な質問である。しかし、質問4に関して「はい」と回答したのは、カリキュラム群とランダム群ともに1名だけであった。このことから、実験1で使用したカリキュラムでは、読みを行うことに関して、学習者が成長している実感を得ることができないものとなってしまっているといえる。前述の通り、カリキュラムは様々な要素からなるものであるため、原因を一概に特定することは難しいが、各要素に関して全体的な見直しが必要であると考えられる。

質問1と質問2は、読み合いの知識と経験についての質問であり、質問1に「はい」と答えた被験者は全員、質問2においても「はい」と回答していた。2.2.1項にて述べた通り、じゃんけんなど日常生活の中でも読み合いを経験する場面は多いため、この結果は妥当だと考える。質問3は読み合いが理解できたかという質問であり、この質問への回答は、拡張じゃんけんゲームが読み合いを行っていることが実感できる環境であるか、ということに関わると捉えられる。結果としては、12人中10人が「はい」と回答していることから、拡張じゃんけんゲームは読み合いを実感できる環境であるといえると考えられる。

表 5.2: 実験1におけるアンケート結果

群 (総人数)	質問1	質問2	質問3	質問4
カリキュラム群 (4名)	3	3	3	1
ランダム群 (4名)	2	2	3	1
対人群 (4名)	2	2	4	0
合計 (12名)	7	7	10	2

図 5.4 に各群のプレテストとポストテストの正答率のグラフを，図 5.5 に各被験者のプレテストとポストテストの正答率のグラフを示す．グラフを一見すると，ランダム群に関して正答率の伸びが良いように見える．しかし，図 5.5 を見ると，ランダム群の中でも大幅に正答率を伸ばした者や，逆に正答率が下がった者など，伸び具合のバラつきが大きい．これらの結果から，一番正答率の伸びが良いランダム群であっても，プレテストとポストテストで有意差があるとは言えない結果となった．また，最も正答率の高いランダム群のポストテストでさえ，19.0%とテストの正答率として考えると低すぎる値となっており，テスト用 CPU に関するの見直しが必要であると考えられる．

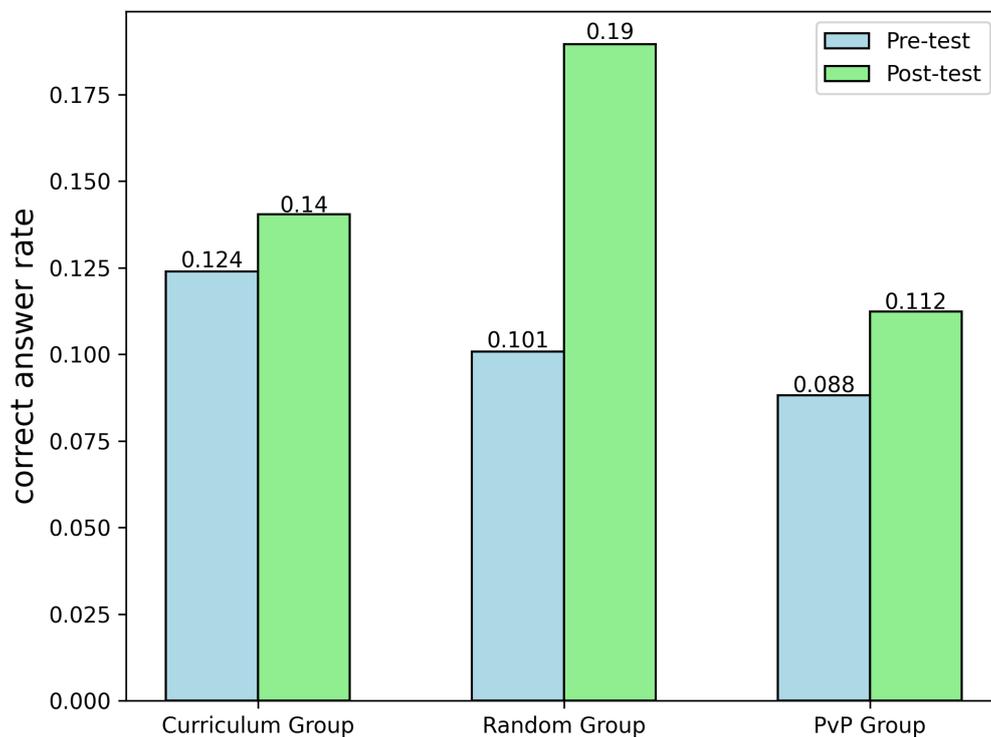


図 5.4: 実験 1 における各群のプレテストとポストテストの正答率

図 5.5 に各被験者のプレテストとポストテストの正答率のグラフを示す。図 5.5 に関して、どの群においても、各被験者の伸び幅のバラつきが大きい。また群ごとに着目した場合においても、伸び幅はバラついているが全被験者が正答率を伸ばした、プレテスト時点での正答率が低い被験者の方が伸び幅が大きい、などの傾向のようなものもみられない。

なお被験者 1 に関して、ポストテストの選択式問題の回答の大半が同じ回答となっていた。選択していた回答は「打撃  $\frac{1}{3}$ : 投げ  $\frac{1}{3}$ : 防御  $\frac{1}{3}$ 」というものであり、これは実験の説明の際に、「ランダムに選択するという意味の選択肢だが、次の相手の手の予想がつかない場合などに選択しても良い」と説明していたものである。また、被験者 1 は、プレテストにおいては平均して 2.2 秒ほどで選択式問題に回答していたが、ポストテストにおける平均回答時間は 0.4 秒ほどとなっている。これらのことから、長時間にわたる実験のストレスや疲れから、次の手を予測する余裕がなかったと考える。このように、カリキュラムによりストレスを感じ投げやりになってしまっは本末転倒であるため、カリキュラムが学習者にストレスを与えないようにする改善が必要である。

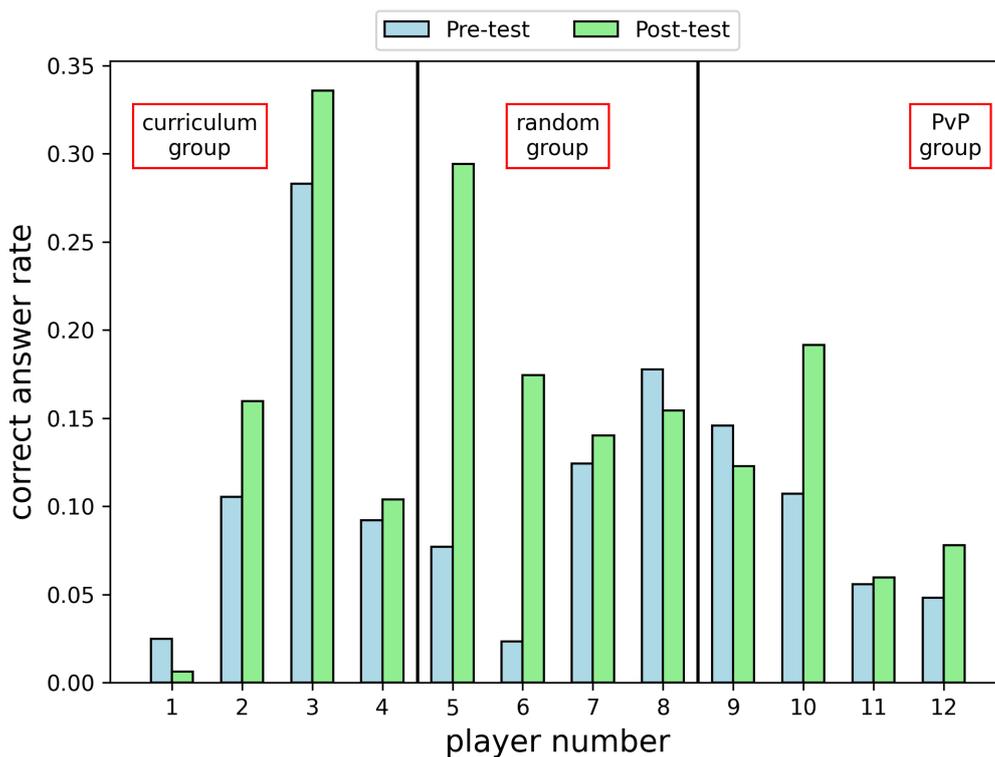


図 5.5: 実験 1 における各被験者のプレテストとポストテストの正答率

本実験は想定していた実験時間を超過してしまった為途中で実験を切り上げた。実際に実験にかかった時間はプレテストが45分、実験時間が3時間30分、ポストテストが30分であった。カリキュラム群内において最もカリキュラムを進めた被験者においても、ターム5のCPU2までであった。カリキュラム群が各CPUを修了するまでに要したゲーム数を平均したグラフを図5.6に示す。なお図5.6の横軸はどのCPUかを示したものであり、3-2であればターム3のCPU2を示す。図5.6を見ると、被験者は主に、ターム3のCPU2とターム4のCPU3を修了するまでに長い時間を要していることが分かる。この結果に関するより詳細な考察は5.5.1項にて述べる。

本実験を通してカリキュラムやゲームの仕様などに関して様々な改善すべき点を発見することができたため、次節ではそれらについて述べる。

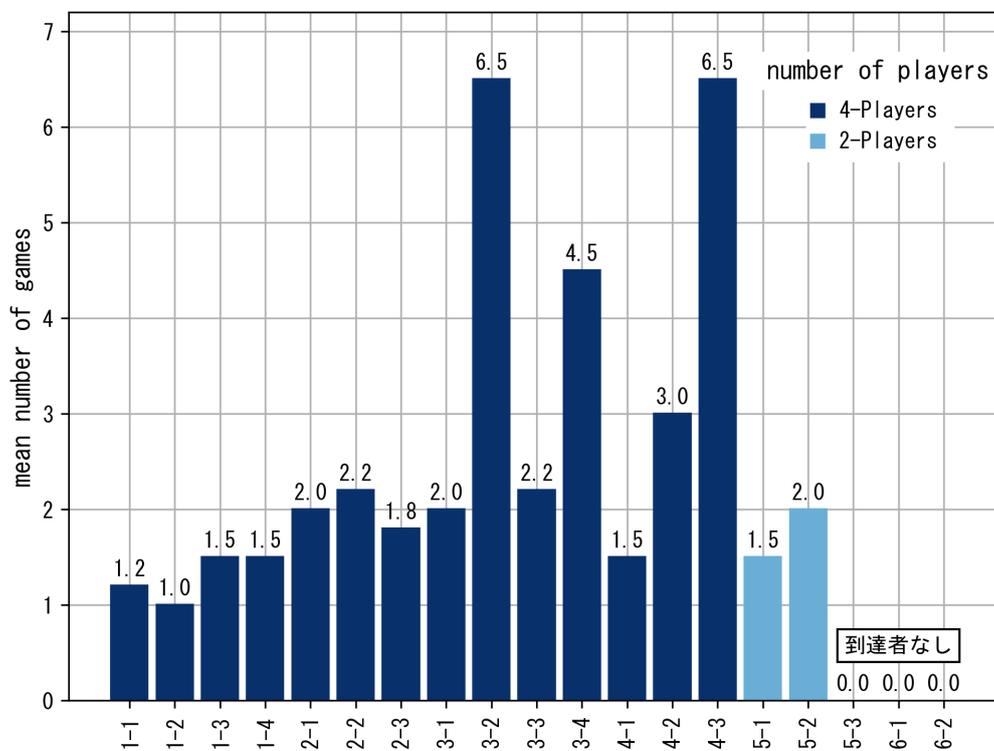


図 5.6: 実験 1 における CPU の修了に要した平均ゲーム数

## 5.5 実験を通じて得られた改善すべき点

本節では実験1と実験に関するアンケートの自由記述欄にて浮き彫りとなった、カリキュラムやゲームの仕様の改善すべき点を述べる。

### 5.5.1 カリキュラムに関して

本項ではヒントやCPUの閾値などカリキュラムに関係する改善すべき点に関して述べていく。

#### A. 確率の偏り

図5.6を見ると、被験者はターム3のCPU2の修了に他のCPUと比較して多くのゲーム数を要している。ターム3のCPU2は“全ての手から $\frac{1}{3}$ の確率で1つを選択する”という行動規則となっている。実際にカリキュラム群の各プレイヤーの1ゲーム目終了時点の各手の頻度を、表5.3に示す。どの被験者も20ラウンドまでゲームを続けているため、期待値を考えると全ての手が7回程度になるが、実際には、被験者4の投げなどは1回しか選択されていないなどの偏りが発生している。このような確率の偏りによって、学習者ごとにCPUの振る舞いが著しく異なるように見えることは、支援システムとして好ましくないため、確率の偏りを少なくするような対策が必要と考えた。

表 5.3: ターム3のCPU2における各プレイヤーの1ゲーム目の各手の選択頻度（実験1）

被験者番号	打撃選択回数	投げ選択回数	防御選択回数
1	5回	4回	11回
2	4回	6回	10回
3	4回	4回	12回
4	12回	1回	7回

#### B. 選択式問題の正答の基準

5.3.1項において選択式問題で正答とする選択肢は“真の確率分布”のみと設定した。しかし実際の実験では前述の通り確率の偏りなどによって、真の確率分布とは異なる確率分布を推測する可能性がある。例えば、“全ての手から $\frac{1}{3}$ の確率で1つを選択する”という行動規則のCPUと対戦をし、不運にもラウンド1からラウンド5までの5ラウンド連続で防御が選ばれた場合、その時点での“実際の着手頻度”を割合で表すと、打撃0割、投げ0割、防御10割である。そのため、被験者が

選択式問題を回答する際、次の手の確率分布の選択肢から 0:0:10 の選択肢を選択したとしても、それは自然な推測であり、その回答を不正解とするのは好ましくない。このようなことを避けるため、正答の基準を変更する必要があると考えた。

### C. ターム冒頭のヒントの提示回数

図 5.6 を見ると、ターム 3 の CPU2 と同じだけのゲーム数をターム 4 の CPU3 に要している。ターム 4 からの CPU は表 4.4 にも示す通り過去のタームの要素を含んだ行動規則を持つため、過去のどのタームと複合している行動規則なのかを考慮して相手の次の手の予測を行う必要がある。しかし考慮すべき過去のターム番号を記したタームのヒントは、図 4.5 に示すターム情報開示画面でのみ提示され、CPU と対戦を繰り返す間は再提示しなかった。そのため被験者は、ターム 4 の CPU3 との対戦を繰り返す中で、ターム 3 の要素である“確率的な手の選択”を考慮しなければならないことを忘れてしまった為により多くのゲーム数をこの CPU に費やしてしまったと推測した。これを避けるため、ターム冒頭のヒントに関して、提示する回数を増やすべきだと考えた。

#### D. テスト用 CPU の難易度

プレテストとポストテストに使用したテスト用 CPU に関して、それぞれの CPU の正答率を図 5.7 に示す。CPU1 と CPU3 に関して正答率が著しく低く、テスト用の CPU というには不適であると考えた。CPU2 に関して正答率が高いわけではないため、テスト用 CPU の行動規則に関して、全体的な難易度を下げるなどの見直しが必要だと考えた。

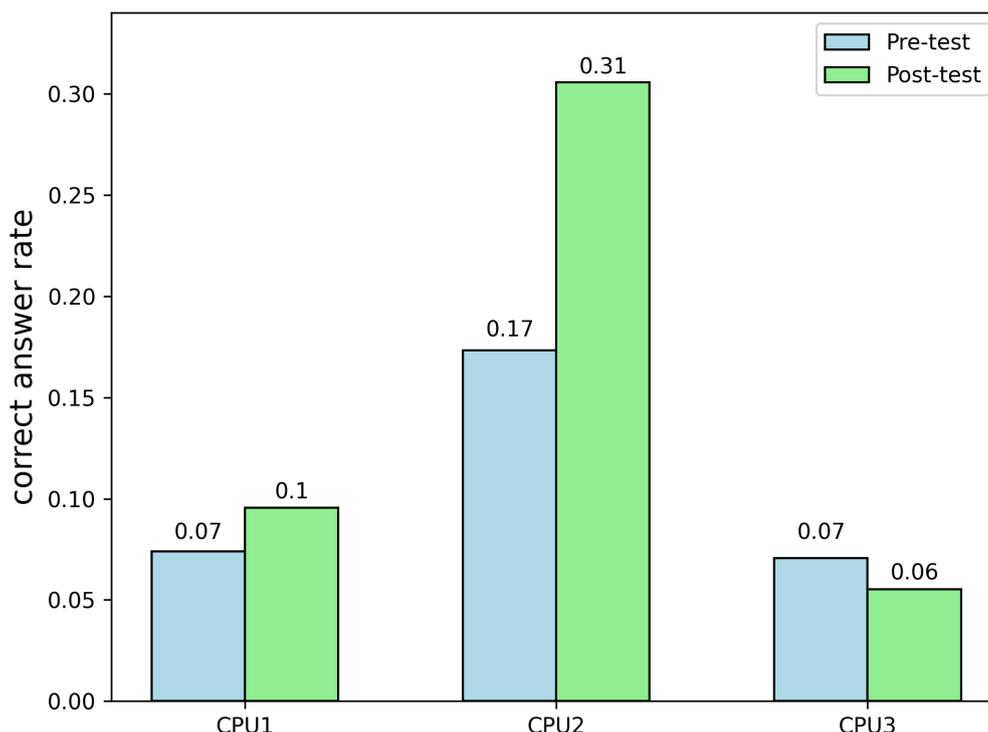


図 5.7: 実験 1 におけるテスト用 CPU ごとの正答率

#### 5.5.2 ゲームの仕様に関して

本項では被験者への実験に関するアンケートに設けた自由記述欄にて被験者から回答してもらった、ゲームの仕様に関する改善すべき点を述べる。

#### E. ゲーム内で表示する情報

実験 1 で使用したゲームにおいて、ゲーム内に、履歴として“直近 3 回の選択した手”と頻度として“過去に選択した手の全頻度”を表示していた。しかし被験者から「勝敗の履歴も情報として表示してほしい」という意見があった。ターム 4 において過去の勝敗をもとに行動を起こす対戦相手を用意していることから、勝敗の情報の提示は有益であると考えた。

## F. 選択式問題の選択肢提示方法

実験1において、選択式問題は図5.1のような形式で実施した。この選択式問題の提示方法に関して、被験者から「選択式問題の選択肢が全て白色かつ似た文章の羅列なせいで見づらい」「ボタンと選択肢の対応を確認するのに時間が食われる」「選択式問題の時間をもう少し伸ばしてほしい」「ゲーム内の履歴や頻度などの情報の視認性が低い」といった意見があった。選択式問題は本カリキュラムにおいて重要な役割を担っているため、選択式問題が認識しづらいことや、選択式問題への回答の補助となるゲーム内情報の視認性が低いことは改善すべき重要な問題である。

## 第6章 第2回被験者実験

実験1を通じて得られた改善すべき点に関して、様々な変更を行い、再度カリキュラムの有効性、カリキュラムを構成するタームの順番の有効性を確認するために2度目の被験者実験を行った（以降実験2と呼称）。なお、得られたデータをもとに、改善を試みた点についても検証を行う。6.1節と6.2節では5.5節にて得られた改善すべき点に対する改善策とその他の変更点について述べ、6.3節と6.4節では実験目的と実験設定を、6.5節では実験結果を述べる。6.6節では6.1節と6.2節にて述べた改善策の効果についての考察を行う。

### 6.1 カリキュラムに関する改善策

本節では、5.5.1項にて述べた、カリキュラムに関して改善すべき点への改善策とその他の変更点について述べていく。

#### 6.1.1 確率の偏りを低減

5.5.1項で述べた“A. 確率の偏り”に関しての改善策を述べる。確率の偏りを低減し、学習者によってカリキュラムのCPUの振る舞いが変化するリスクを下げることを目的とし、実際の着手頻度と頻度の期待値の差にもとづき確率の補正を行うこととした。

例えば、全ての手から $\frac{1}{3}$ の確率でランダムに手を選択する、といった行動規則のCPUとの対戦を考える。6ラウンド終了時点で、着手頻度の期待値は、打撃2回、投げ2回、防御2回であるが、実際には打撃0回、投げ2回、防御4回であったとする。この際の、7ラウンドでの各手の選択確率を、打撃は実際の着手頻度が期待値よりも少ない回数なので高い確率に、防御は期待値より多いので低い確率としたい。これを実現するため、各手の、期待値と実際の着手頻度の差を用いた補正值を使用する。

補正式はパラメータ $\alpha$  ( $\alpha > 1$ ) を使用し、 $\alpha^{(\text{期待値}-\text{実際の着手頻度})}$  とした。上記の例で $\alpha = 2$  の場合を考えると、補正值は打撃が4、投げが1、防御が $\frac{1}{4}$ となる。この補正值を使用して各手の選択確率を求めると、打撃76.0%、投げ19.0%、防御5.0%となり、打撃が高い確率で防御が低い確率を実現できている。一般化した式を6.1

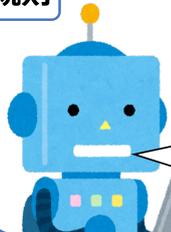
式に述べる.  $i$  と  $j$  は 1 から順に打撃, 投げ, 防御とし,  $c$  は補正後の各手の選択確率,  $e$  は着手頻度の期待値,  $f$  は実際の着手頻度とする.

$$c_i = \frac{\alpha^{e_i - f_i}}{\sum_{j=1}^3 \alpha^{e_j - f_j}} \quad (6.1)$$

この例で用いた CPU は, どのような状況でも同じ確率分布の行動規則であったが, カリキュラムの CPU には, 状況に応じて異なる確率分布で手を決定する CPU が存在する. このような CPU の場合, 実際の着手頻度の値には, 全体の着手頻度ではなく, 状況ごとに保存している実際の着手頻度を使用する. 具体例として, 状況 A と状況 B で別々の確率分布で手を決定する CPU を考える.

実際に各手の選択確率を求める概略図を, 図 6.1 に示す. 図 6.1 はある CPU が 16 ラウンド目の手を決める場面である. この CPU は, 状況 A において全ての手から  $\frac{1}{3}$  の確率でランダムに手を選択し, 状況 B において打撃を 1 割, 投げを 1 割, 防御を 8 割で選択する. 図の中部はラウンドごとの状況と選択手の経過を表示しており, 16 ラウンド目が状況 A であることが分かる. 図の下部に記載している通り, 選択した手は状況ごとに保存されており, 16 ラウンド目の状況は A であるため, 状況 A の履歴を用いて 6.1 式に示した補正後の確率の計算を行い, その確率の応じて手を決定する.

### 1: 行動規則



状況Aでは 打撃 33%, 投げ 33%, 防御 33%  
 状況Bでは 打撃 10%, 投げ 10%, 防御 80%  
 で手を決めよう.

### 2: ラウンドの推移

状況:	A	A	B	A	B				A										
ラウンド:	1	2	3	4	5				13	14	15	16							
選択手:	打	打	防	投	打				打	投	防	?							

### 3: 16ラウンド (状況A) の手の決定

CPUの  
 これまでの着手頻度

状況A : 打5回, 投2回, 防3回  
 状況B : 打1回, 投1回, 防3回

#### 補正值

打撃 :  $\alpha^{(e_1 - f_1)} = 0.46$   
 投げ :  $\alpha^{(e_2 - f_2)} = 1.87$   
 防御 :  $\alpha^{(e_3 - f_3)} = 1.17$



#### 重み付き確率

打撃 :  $c_1 = 0.13$  (選択確率 13%)  
 投げ :  $c_2 = 0.53$  (選択確率 53%)  
 防御 :  $c_3 = 0.33$  (選択確率 33%)

図 6.1: 補正式を用いた各手の選択確率の導出 ( $\alpha = 1.6$ )

このように、各手に関して、状況ごとの実際の着手頻度をもとに、6.1式により補正した値を求め、補正值を利用し各手の選択確率を導出する。この変更により、確率の偏りによる、学習者ごとのCPUの振る舞いの差を低減することが期待できる。

## 6.1.2 選択式問題の正答の基準を変更

5.5.1項で述べた“B. 選択式問題の正答の基準”に関する改善策を述べる。実験1では、確率の偏りなどから、実際の着手頻度が真の確率分布と著しく乖離してしまうことがあった。そのような場合に、実際の着手頻度から確率分布を推測して回答した選択肢は、推測としては妥当なものが多かった。しかし、妥当な推測だとしても、真の確率分布のみを正答としていたため不正解となっていた。

そこで、真の確率分布だけでなく、状況ごとの実際の着手頻度にもとづいた確率分布も正答とすることとした。しかし、選択肢は事前に用意したものであり、実際の着手頻度が常に選択肢と同じ確率分布となる可能性は低い。そのため、実際の着手頻度にある程度近い確率分布であれば正答とした。なお使用する実際の着手頻度は、図6.1の例と同じく、状況ごとに保存したものを使用する。確率分布の距離を計算する式を6.2式に示す。 $p_i$ はある選択肢の各手の確率、 $f_i$ は実際の着手頻度にもとづいた各手の確率である。 $i$ は1から順に打撃、投げ、防御とし、 $p_1$ はある選択肢の打撃の確率を意味する。この変更により、CPUの振る舞いを考慮した、運不運に左右されない、より正確な修了判定が期待できる。

$$dist = \sum_{i=1}^3 |p_i - f_i| \quad (6.2)$$

## 6.1.3 選択式問題の実施回数の変更

実験1において、選択式問題は1ラウンド目から実施した。しかし、そのCPUとの対戦が初めての場合、1ラウンド目の結果のみから、CPUの次の手の確率分布を予測することは事実上不可能である。そこで、理不尽に正答率が低くなってしまうことを避けるため、対戦開始から数ラウンドは選択式問題を実施しないこととした。カリキュラムには、常に一定の確率分布で手を決定するのではなく、状況ごとに手を決定する確率分布が異なるようなCPUが存在する。そのため、選択式問題が始まってから初めて遭遇する状況の場合には、その選択式問題の回答は正答率の計算に含めないこととした。こちらの理由に関しても、前述のものと同じく、事実上不可能なためである。正答率に含めないが選択式問題を実施するのは、プレイヤーに、その状況が初出と知られることを避けるためである。

具体例として、状況A、状況Bにおいて、それぞれ異なる確率分布で手を決定するCPUを考える。この際、5ラウンド目までは選択式問題を実施しない。10ラウ

ンド目まで状況 A が連続し、10 ラウンド目に初めて状況 B が発生した場合、10 ラウンド目での選択式問題は実施するが、その結果は正答率の計算に含まない。この変更により、前項と同じく理不尽に正答率が下がることを防ぐことが期待できる。

#### 6.1.4 ターム冒頭のヒントの再提示

5.5.1 項で述べた“C. ターム冒頭のヒントの提示回数”に関する改善策を述べる。カリキュラム群の被験者が、同じ対戦相手との再戦を繰り返すうちに、ターム冒頭のヒントを忘れてしまうことが発生した。これを防止するため、ランダム群と同じく、対戦相手ごとのヒントと同じ画面にターム冒頭のヒントを提示することとした。また、“考慮すべき過去の対処法”に関しては重要であるため、赤文字で表示する。実際のゲーム画面を図 6.2 に示す。前述の通り、画面の構成は図 5.3 と同じく、上がターム冒頭のヒント、下が CPU ごとのヒントとなっている。



図 6.2: 実験 2 における対戦前のヒント提示画面

#### 6.1.5 CPU ごとのヒントの増加

同じ CPU との対戦の繰り返して、長時間詰まってしまうことは好ましくない。それを避けるためには、ヒントなどを段階的に増やすような工夫が考えられる。しかし、実験 1 のカリキュラムにおいて、同じ CPU と再戦の際にヒントは再提示されるだけであった。そこで、一定回数再戦した場合には提示するヒントを増やすこととした。学習者に、ヒントが増えたことが一目で分かるよう、増えたヒントは赤色で強調表示した。具体的なゲーム画面を図 6.2 に示す。(a) は最初に対戦を行う場合のゲーム画面であり、ヒントは 1 つしか表示されていない。(b) は複数回再戦をしたことで、全てのヒントが表示されたゲーム画面であり、ヒントは 2 つ表示されている。なお、画面の構成は図 5.3 と同じく、上がターム冒頭のヒント、下が CPU ごとのヒントとなっている。ヒントの増え方などに関しては、6.4.3 項にて述べる。

### 6.1.6 修了したCPUの行動規則の提示

選択式問題の正答基準の変更や、初出の状況を正答率の計算に含めないなどの変更により、CPUの行動規則をあまり推測できなかったのに、修了条件を満たしてしまう場合が発生すると考えた。そこで、修了したCPUの行動規則を、学習者へ提示するようにした。行動規則の提示により、あまり行動規則を推測できなかった学習者においても、CPUが何を狙って手を選択していたかなどを知ることができ、読み合いの考え方を理解する一助になると考えたためである。実際に行動規則を提示する際のゲーム画面を図6.3に示す。



図 6.3: 修了後の行動規則の提示

### 6.1.7 カリキュラムで対戦するCPU数を減らす

学習において、解けた問題と類似した問題を解くことは、振り返りなどの観点から重要な役割を持つ。しかし、本実験においては、カリキュラムを最後までやり通すことができていないことの方が重大な問題と判断した。そこで、カリキュラムにかかる時間を減らし、実験時間内にカリキュラムを終了させることを目的に、ターム内の似通った行動規則を持つCPUを排除し、CPUの全体数を減らした。具体的には、付録B.1に掲載した、ターム1におけるCPUの内、“打撃のみを選択”と“防御のみを選択”のような似通った行動規則を持つCPUをどちらか片方のみとした。

### 6.1.8 テスト用CPUの変更

5.5.1項で述べた“D. テスト用CPUの難易度”に関する改善策を述べる。実験1においてテスト用CPUの難易度が高すぎたことにより、プレテストとポストテ

ストのどちらにおいても正答率がとても低く、テストとして不適切な結果となった。そこで、テスト用CPUの全体的な難易度を下げた。また、実験1は3種のテスト用CPUであったが、それを5種とし、より様々な難易度のCPUを用いてテストを行う。なお、具体的な行動規則や対戦回数などの詳細な設定は、6.3節にて述べる。

## 6.2 ゲームの仕様に関する改善策

本項では、5.5.2項にて述べた、ゲームの仕様に関して改善すべき点に対する改善策について述べる。

### 6.2.1 ゲーム内情報の追加

5.5.1項で述べた“E. ゲーム内で表示する情報”に関する改善策を述べる。履歴が直近3ラウンド分表示されることにあわせて、新たに「直近3ラウンドの勝敗」という情報を追加した。勝敗は、勝利を赤、敗北を青、引き分けを黒とし、視覚的にも分かりやすい情報提示となるようにした。また、次項で述べる選択式問題の提示画面とあわせて、残り歩数の情報を上部にも提示するように変更した。具体的な対戦画面は、図6.4で示す。実験1にて使用したゲームのゲーム画面は図4.2のようになっており、実験2で使用するゲーム画面では、画面内上部の履歴の上に、新たに追加した勝敗の情報が記載されている。また、画面上部の左右には、それぞれのプレイヤーの残り歩数が記載されている。

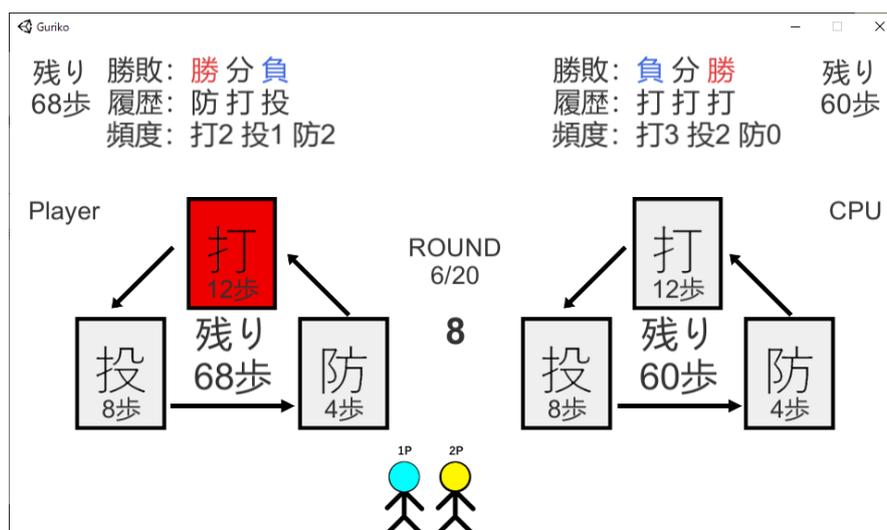


図 6.4: 実験2における対戦画面

## 6.2.2 選択式問題の選択肢提示方法の変更

5.5.1 項で述べた“F. 選択式問題の選択肢提示方法”に関する改善策を述べる。なお、実験1で使用した選択式問題の提示画面は、図5.1である。自由記述にて得られた意見をもとに、改善すべき点は、「文字の羅列」と「回答用ボタンの位置」という2点だと考えた。1点目は、10つの選択肢を羅列したことにより、一見した際に、それぞれの選択肢の違いが分かりづらいという問題である。2点目は、確率分布の情報と回答用のボタンが離れた場所にあり、さらに、回答用のボタンが敷き詰められていることにより、誤って別の選択肢を回答してしまう可能性があるという問題である。

これらの問題を解消するために、図6.5のような選択式問題の提示方法に変更した。新たな提示方法では、レーダーチャート上に色分けした回答用ボタンを配置することによって、一目見ただけで、どのような確率分布が選択肢にあるのかを確認可能である。さらに、確率分布の情報がそのままボタンとなっており、マウスオーバーした際には、背景が黄色に変化するように設定した。

また、選択式問題の提示画面では、手の履歴や頻度などの情報が見づらいという意見もあったことから、上に表記している情報が見えるようにレーダーチャートを配置した。これらの変更により、選択肢の視認性の向上や、誤って別の回答を選択するミスの低減が期待される。



図 6.5: 実験2における選択式問題の提示方法

## 6.3 実験の目的と設定

6.1 節にて変更を加えたカリキュラムに関して、再度被験者実験を行う。本節では、実験の目的と、テスト用 CPU や群ごとの実験内容など被験者実験の設定を述べる。実験目的は実験 1 と同じく、カリキュラムそのものとカリキュラムを構成するタームの順番の 2 つに関して、有効性の検証である。また、それぞれの変更点に関して、データから考察可能なものはその影響を考察する。

続いて実験設定について述べる。本実験に参加した被験者人数は 20 名であり、内訳はカリキュラム群とランダム群が 6 名、対人群が 8 名となっている。また、6.1.8 項で述べた通り、プレテストとポストテストに使用する CPU を変更しているため、詳細を記す。テスト用 CPU として、5 種の CPU を用意した。用意したテスト用 CPU を以下に列挙する。実験 1 では、複数のタームの内容を複合したような CPU を用意したが、実験 2 では単一タームの内容から CPU を用意した。対戦回数については、テスト用 CPU1 とテスト用 CPU2 は、比較的典型的な行動規則を持つ CPU であるため 1 回とし、その他の CPU は 2 回の対戦を行うこととした。全体では計 8 回の対戦であり、対戦順は実験 1 と同じくランダム順とした。なお、どのテスト用 CPU と何回対戦を行ったかなどの情報は、被験者には提示しなかった。また、ヒントに関して、実験 1 と同様に提示しなかった。

テスト用 CPU1 CPU の残り歩数が、被験者プレイヤー以上の場合には最も高い利得の手を、CPU の残り歩数が被験者より少ない場合には、被験者プレイヤーの最も高い利得の手に勝利する手を選択する。(ターム 2 の内容に近い行動規則)

テスト用 CPU2 打撃→投げ→防御→打撃→... と周期的に手を選択する。(ターム 1 の内容に近い行動規則)

テスト用 CPU3 前回ラウンドの勝敗が、CPU の勝利であれば同じ手を選択し、CPU の敗北かあいこであれば、前回ラウンドで CPU が選択しなかった 2 種の手から、半々の確率で選択する。(ターム 4 の内容に近い行動規則)

テスト用 CPU4 被験者プレイヤーが、前回ラウンドで選択した手にあいこ以上で勝てる手 2 種を、半々に選択する。(ターム 5 の内容に近い行動規則)

テスト用 CPU5 被験者プレイヤーの最頻手が、1 つの場合にはそれに勝利する手を選択し、2 つの場合にはそれらに勝利する手から半々で選択し、3 つの場合にはランダムに手を選択する。(ターム 5 の内容に近い行動規則)

各群が行った実験内容に関して述べる。カリキュラム群と対人群は実験 1 と同様の実験内容である。カリキュラム群はターム 1 から順にカリキュラムを進め、対

人群は他群と同時間対人，群内で対戦相手を入れ替えながら対戦を行う．ランダム群は，実験1と内容は同じだが同じCPUとの対戦回数が異なる．実験2ではカリキュラムに出てくるCPUとランダムに対戦し，修了するか3回まで対戦を繰り返す（実験1では修了するか2回までの対戦）．なお，ランダム群であっても，修了した場合には6.1.6項で述べた通り，回答を被験者へ提示する．また，各種パラメータの設定に関して述べる．正答の基準に関しては，6.2式で述べた距離が0.8以下であれば正答とし，確率の補正式に関しては，6.1式の $\alpha$ を1.6とした．

実験1では実験時間を3時間としたが，結果として3時間半でもカリキュラムを終了できなかったため，実験2の実験時間は，プレテストを30分，実験時間を4時間，ポストテストを30分とした．実験1と同様に実験前後に，アンケートを実施した．アンケート内容は実験1と同様である．実験前のものが「質問1: 読み合いを知っていたか」「質問2: 読み合いを経験したことがあるか」の2つであり，実験後は「質問3: 読み合いを理解できたか」「質問4: 自分の予測の精度が上がっていると感じるか」の2つであった．これらの質問へは全て「はい」か「いいえ」での回答とした．

## 6.4 実験2におけるカリキュラムの設定

実験2を実施するにあたって，選択式問題の仕様やプレイヤーの思考時間などのカリキュラムの設定を本節にて説明する．なお，カリキュラムで使用したCPUの行動規則とヒントに関しては，付録B.2を確認されたい．

### 6.4.1 選択式問題

実験1と同じように，被験者にはCPUの次の手の確率分布を予測し，選択肢から回答してもらおう．実験1から新たに変更した選択式問題の提示画面は図6.5に示す．6.1.2項で述べたように，実験1から変更して，真の確率分布のみでなく実際の着手頻度に近い選択肢も正答とした．なお，選択肢内に必ず，真の確率分布が存在していることは実験1と同様である．

手の選択時間と選択式問題への回答時間は，実験1と同じく合わせて30秒とした．しかし，6.1.3項に述べた通り，選択式問題を実施しないラウンドが存在する．また，選択式問題を実施する際には，実験1のアンケートにて得られた「選択式問題において時間が不足する」という意見をふまえて，時間の配分を変更した．具体的には，選択式問題を実施しないラウンドでは，手の選択の制限時間を15秒とし，選択式問題を実施するラウンドでは，制限時間をそれぞれ，手の選択が10秒，選択式問題が20秒とした．

## 6.4.2 各 CPU の正答率の閾値

4.2.5 項で述べた通り、CPU にはそれぞれ、次の CPU に進むための、正答率の閾値を設定している。実験 1 の際は、ターム 3 の CPU に限り、正答率以外の条件も修了条件として付与していた。しかし、実験 2 ではそれらの正答率以外の条件は適用しなかった。また実験 1 と同じく、タームが進むにつれて CPU の行動規則は複雑なものとなるため、段階的に閾値を下げている。

6.1.2 項で述べたように、正答の基準を変更し運不運を減らすようにしたが、それだけでなく、実験 1 と比較してターム 1 以外の閾値の下げた。正答の基準変更に加えて閾値を下げることは、実験 1 以上に、「CPU の行動規則をあまり推測できなかったが、修了条件を満たしてしまった被験者」を増やす可能性が高い。しかし、その可能性を加味したうえで閾値を下げたのは、実験 1 の被験者全員がカリキュラムを切り上げたという結果を受け、カリキュラムを時間内にやり切ることが重要だと考えたからである。また、6.1.6 項で述べたように、修了した行動規則を提示することで、CPU が何を狙って手を選択していたかなどを知る機会を設け、「被験者にそのタームで伝えたい考え方が伝わらない」という最悪の事態は可能な限り避けられるようにしている。表 6.1 に各 CPU の修了条件をまとめた表を記す。なお、実験 1 における修了条件をまとめた表は、表 5.1 に掲載している。

表 6.1: 実験 2 における CPU ごとの閾値

ターム番号	CPU 番号	正答率の閾値
1	1~3	0.8
2	1~3	0.75
3	1~3	0.7
4	1~2	0.65
4	3	0.6
5	1~2	0.6
6	1	0.55

## 6.4.3 各 CPU のヒント提示画面

図 6.2 に実験 2 でのヒントの提示画面を示す。6.1.4 項で述べた通り、ターム冒頭のヒントを、対戦前のヒントと合わせて提示するように変更した。また、6.1.5 項で述べたように、実験 2 のカリキュラムでは、同じ CPU と再戦を繰り返すにつれてヒントが増加していく。ヒントは最も多い CPU でも 3 つまでであり、増加の仕方は 3 回目の対戦から 1 つずつ増加していくように設定した。よって、ヒントが最も多い CPU の場合、4 回目の対戦からすべてのヒントが提示される。また、実

験1と同じく、被験者はSキーを押下することで、任意のタイミングでCPUとの対戦を開始することができる。

#### 6.4.4 ランダム群へのヒント提示

6.3節で述べた通り、ランダム群は最大3回までしか再戦を行わない。そのため、ヒントを3つ設定されたCPUの場合には、カリキュラム群と同様にヒントを増加させると、最後のヒントを提示することができない。そのため、ランダム群においては、2回目の対戦から全てのヒントが提示されるように設定した。

### 6.5 実験結果

アンケートで各質問に「はい」と回答した人数を、表6.2にまとめる。また、表6.3に実験1における結果を再掲する。なお、質問番号と内容の対応は、6.3節を確認されたい。質問4は、5.4節で述べたように、予測精度の自己評価に関する質問であり、本研究では重要な質問である。実験1ではカリキュラム群とランダム群ともに、4名中1名と非常に悪い結果であった。しかし、実験2ではカリキュラム群が6名中5名、ランダム群が6名中6名が「はい」と回答するという、良い結果を得た。しかし、対人群においても、8名中7名が「はい」と回答している。このことから、カリキュラムの変更によるものという訳ではなく、ゲームの仕様変更やテスト用CPUの難易度を下げたことが要因かと考える。具体的には、ゲームの仕様として勝敗情報を加えたことで、推測をしやすくなったことや、テスト用CPUの難易度を下げたことで、高い自己評価を行えるようになったことなどが考えられる。このように自己の成長を感じられることは、学習を継続するうえでのモチベーションとなるため、重要な要素であると考えられる。

表 6.2: 実験2におけるアンケート結果

群 (総人数)	質問1	質問2	質問3	質問4
カリキュラム群 (6名)	4	2	6	5
ランダム群 (6名)	4	4	6	6
対人群 (8名)	5	2	7	7
合計 (20名)	13	8	19	18

表 6.3: 実験1におけるアンケート結果（再掲）

群（総人数）	質問1	質問2	質問3	質問4
カリキュラム群（4名）	3	3	3	1
ランダム群（4名）	2	2	3	1
対人群（4名）	2	2	4	0
合計（12名）	7	7	10	2

質問3は、拡張じゃんけんゲームが、読み合いを行っていることが実感できる環境か、ということに関わる質問であり、20名中19名が「はい」と答えた。実験1の結果と合わせても、拡張じゃんけんゲームが読み合いを実感できる環境であると言える。また、質問1と質問2は、読み合いの知識と経験に関する質問となっており、回答結果から、実験2における被験者はあまり読み合いを経験したことがない人物たちであった。

図6.6に各群のプレテストとポストテストの正答率のグラフを示す。カリキュラム群に関しては11.0%、ランダム群に関しては14.0%程の正答率の上昇がみられたが、ウィルコクソンの符号付順位和検定を行ったところ、両群共にプレテストとポストテストに有意差はみられなかった。また、クラスカルウォリス検定による3群の多群間比較を実施したところ、そちらに関しても有意差はみられなかった。

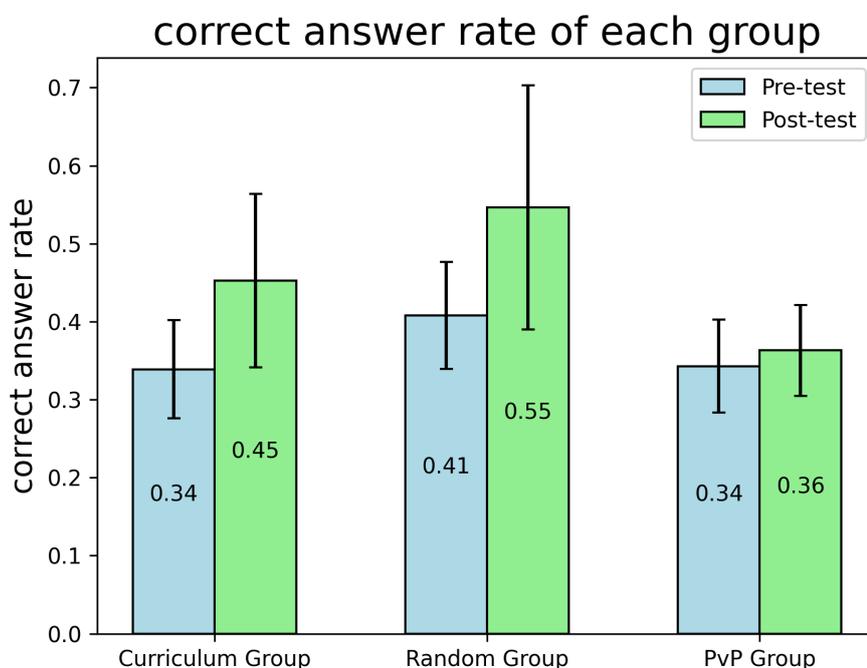


図 6.6: 実験2における各群のプレテストとポストテストの正答率

結果としては、カリキュラム群とランダム群において、比較的高い数値での正答率の上昇がみられたが、両群ともにバラつきが大きかったため、有意差が出ることは無かった。この結果を受け、カリキュラムの次の改善すべき点としては、このバラつきを抑えるような工夫が必要であると考えられる。

また、カリキュラムの有効性に関して、CPUを相手に正答率が上昇するだけでなく、実際の対人戦において勝率が変化するかは重要な指標である。しかし、カリキュラムを受けた群のプレイヤーと、対人群のプレイヤー同士の対戦は、本実験では行えなかった。そのため、カリキュラムを受けることによる、対人戦での勝率への影響などは検証できていない。

図 6.7 に各被験者のプレテストとポストテストの正答率のグラフを示す。カリキュラム群は、全ての被験者が微増から増加の傾向を見せているが、微増の被験者に関して、被験者 1 は 3.0%、被験者 2 は 1.8%、被験者 4 は 0.97% の増加と誤差の範囲と言っても良い程であった。なお、被験者 4 は、カリキュラムを途中で切り上げた被験者であった。ランダム群は、群全体で最も正答率を伸ばした群であったが、被験者 9 のように大きく正答率を伸ばしたもののあれば、被験者 11 のように正答率を下げたものもあり、全体的に傾向などは見受けられない。対人群は最もバラつきが小さい群であったが、被験者 16 のように大きく正答率を伸ばした被験者もあれば、被験者 17 のように大きく正答率が落ちたものもあり、こちらも全体的に傾向などは見受けられなかった。総じて、どの被験者においても伸び幅はまちまちであった。また、日本語の得手不得手による影響もあるようで、大きく正答率を伸ばした被験者は日本語が堪能な被験者が多く、改善のためには、ヒントなどの情報に関して、ユニバーサルで分かりやすい方法が必要だと考える。

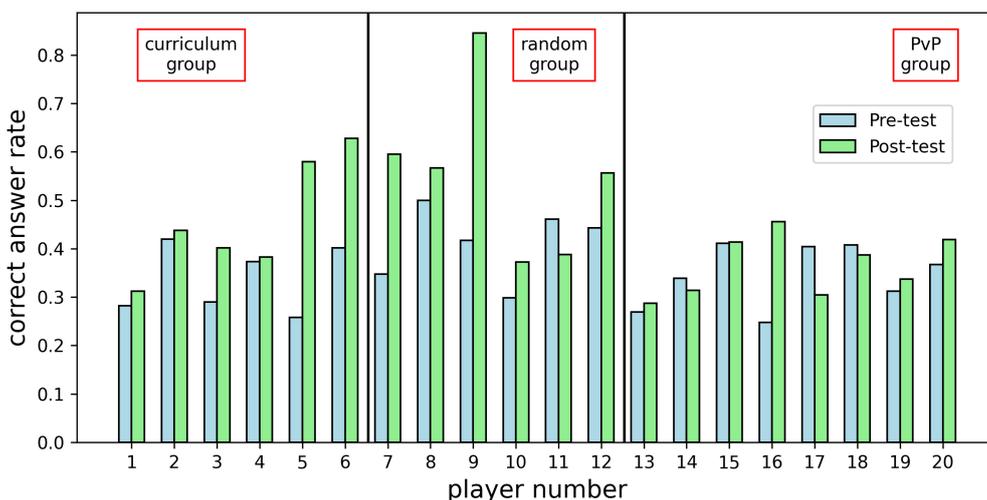


図 6.7: 実験 2 における各被験者のプレテストとポストテストの正答率

実験1において、カリキュラム群の被験者4名は、全員最後のCPUまでたどり着くことはできなかった。実験2では、最後のCPUまでたどり着いたプレイヤーは6名中4名であり、2名はカリキュラムの途中で実験時間が終了したため実験を切り上げた。1名はターム3のCPU1の途中であり、もう1名はターム5のCPU1を修了した後であった。実験1における各CPUの終了までの平均ゲーム数を図5.6に示したが、実験2における同様のグラフを図6.8に示す。なお、図6.8の上部が実験2における数値であり、下部は実験1での数値である。また、実験2のグラフにおいて「削除」となっているCPUは、6.1.7項にて述べたように、カリキュラムから取り除いたCPUであり、実験1のグラフにおいて「到達者なし」となっているCPUは、実験を切り上げたために、誰も対戦することのなかったCPUである。図6.8の横軸はどのCPUかを示したものであり、3-2であればターム3のCPU2を示す。

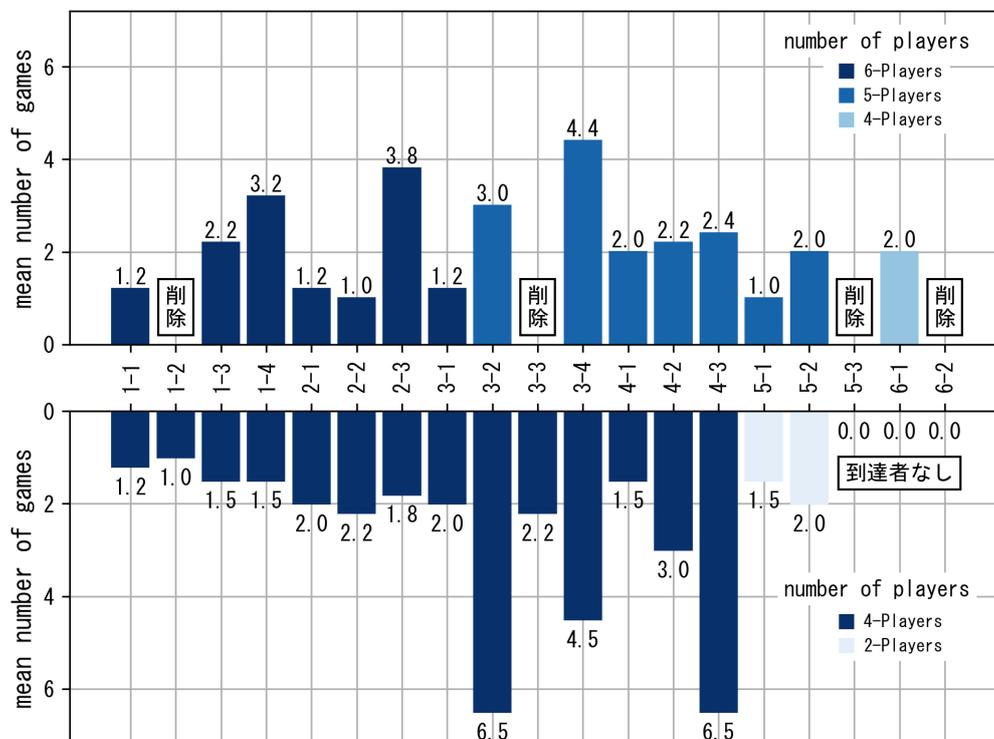


図 6.8: 実験2におけるCPUの修了に要した平均ゲーム数（下部は実験1における平均ゲーム数）

ターム2のCPU3において、様々な変更を施したに関わらず平均ゲーム数が増加している。これは、前述の途中で切り上げたプレイヤーが12回の再戦を行ったことによる影響であり、他被験者の終了までの平均ゲーム数は2.2であった。その被験者が12回も再戦を行ったことに関して、ヒントに使用した、強気や弱気、特殊手と言った単語の意味を理解することに時間を要し、他被験者と比較して多くのゲーム数を費やしてしまったと考えた。

実験1では終了までの平均ゲーム数の最大は6.5ゲームであったが、実験2では最大でも4.4ゲームに抑えることができた。これは、確率の偏りに対する改善やヒントの増加、正答率の基準変更などの変更点が影響していると考えられ、詳しい考察は次節にて行う。

## 6.6 改善策の効果

本節では、6.1節や6.2節にて述べた、改善すべき点に対する変更が、実験2においてどれ程の効果があったかを考察する。

### 6.6.1 確率の偏りの低減

6.1.1項で述べたように、実験2におけるカリキュラムでは、確率の偏りを減らすことを期待して、補正值を用いた補正した確率を採用した。具体的な式は6.1式に示す。なお、実験2におけるパラメータ $\alpha$ の値は1.6としていた。

5.5.1項の“A. 確率の偏り”にて提示した、表5.3と同じく、“全ての手から $\frac{1}{3}$ の確率で1つを選択する”という行動規則を持つ、ターム3のCPU2の着手頻度で比較を行う。表6.4に、実験2における、各被験者の1ゲーム目終了時点の各手の頻度を示す。なお、対象の被験者はカリキュラム群であり、被験者4はターム3のCPU1にて実験を切り上げたため集計していない。表6.4を見ると、表5.3と比較して、被験者ごとの振る舞いの差が十分低減できていると考える。ただし、このような偏りの低減は、本来の乱数を不自然に操作する行為であり、偏りを低減させ過ぎた場合には、意図せぬ深読みを招く恐れがある。ターム3のCPU2であれば、「頻度が少ない手から手を選ぶ」といった具合である。

表 6.4: ターム3のCPU2における各プレイヤーの1ゲーム目の各手の選択頻度（実験2）

被験者番号	打撃選択回数	投げ選択回数	防御選択回数
1	6回	8回	6回
2	7回	7回	6回
3	8回	7回	5回
4	-	-	-
5	7回	6回	7回
6	8回	6回	6回

## 6.6.2 工夫による確率的CPUでの正答率の変化

6.5節の図6.8に示すように、実験1と比較して全CPUにおいて平均的に修了までのゲーム数が減っている。これに関して、正答率の基準変更とCPUの閾値の変更、という2つの変更点はどう影響しているかを考察する。なお、実験2では、6.1.2項で述べた6.2式により求められる距離が、0.8以下の選択肢を正解扱いとしていた。

カリキュラム群における、各タームの1ゲーム目の正答率を図6.9に示す。グラフに関して、赤の棒グラフは真の確率分布のみを正答とした際の正答率、オレンジの棒グラフは実験2で用いた正答判定における正答率である。ターム3以降のタームは、確率的に行動を起こすCPUを設定しており、正答率の基準を変更したことによって、真の確率分布のみを正答とした場合と比較し、高い正答率となっている。これにより、CPUを修了するためのハードルが下がったと考える。なお、ターム5がどちらの正答基準においても同じ数値なのは、主たる行動規則は確定的なもので、確率的な行動はあくまで行動規則の一部であるためだと考察した。

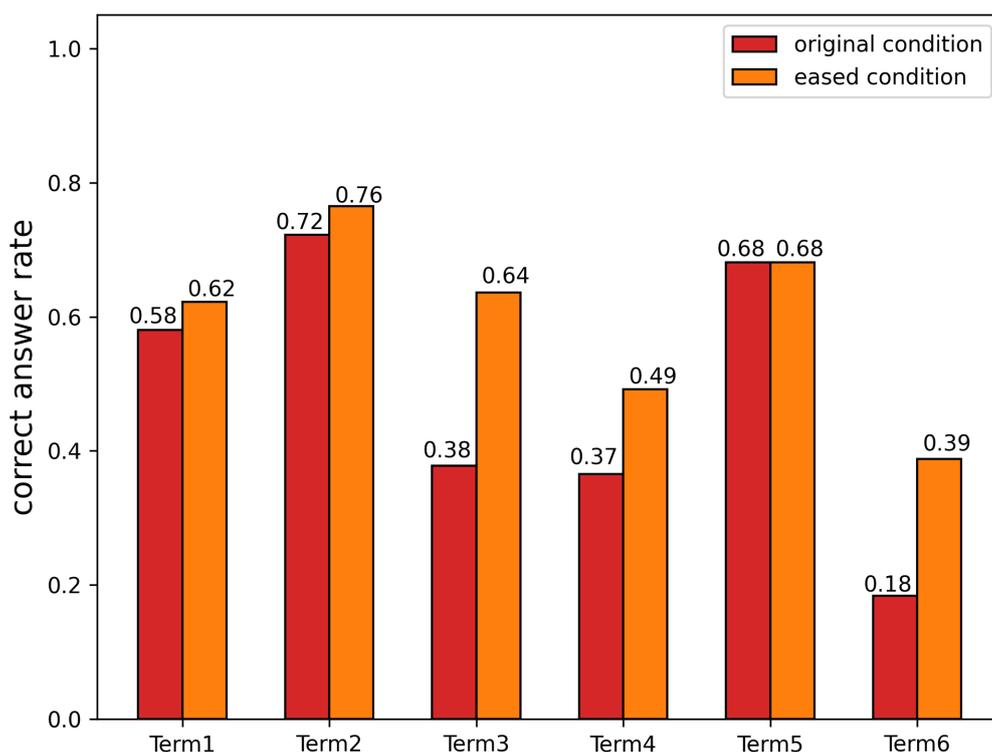


図 6.9: 実験2におけるタームごとの正答率 (1ゲーム目)

赤が真の確率分布のみでの正答率

オレンジが変更後の正答基準での正答率

実験2の正答基準における、各CPUの被験者ごとの1ゲーム目の正答率を図6.10に示す。なお、グラフの横軸はどのCPUかを表したものであり、3-2であれば、チーム3のCPU2を表す。また、グラフの見方に関して、黒の実線は実験1におけるCPUの閾値、黒の破線は実験2におけるCPUの閾値となっている。なお、より詳細なデータと考察はA.1に掲載する。

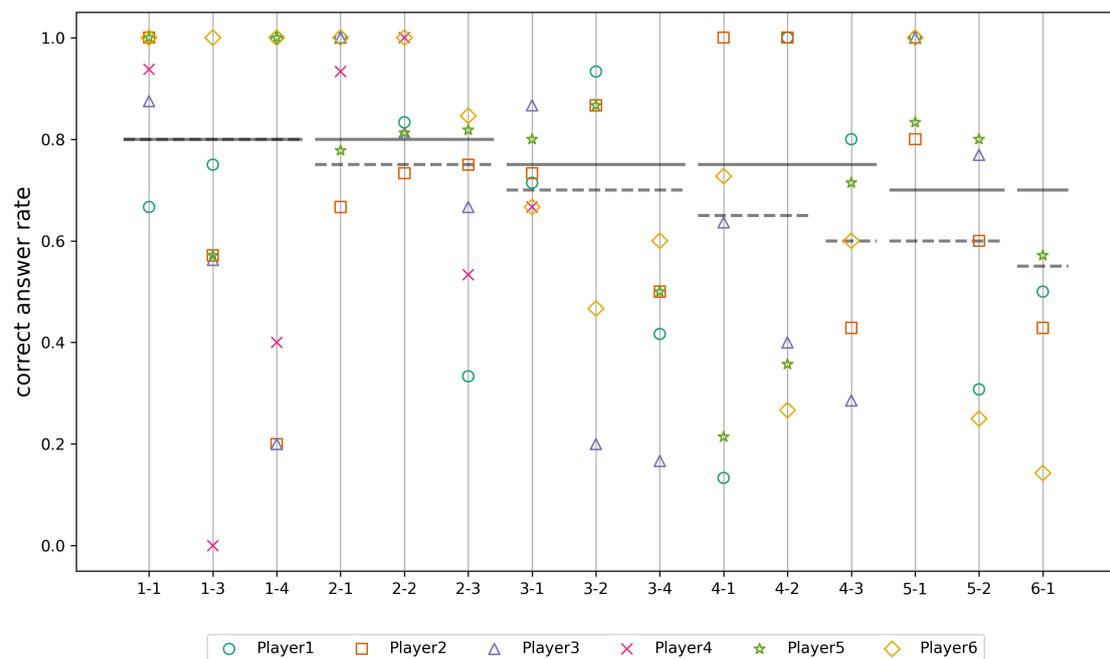


図 6.10: 実験2における各CPUの被験者ごとの正答率（1ゲーム目）

図6.10を見ると、閾値を下げたことにより修了条件を満たした被験者が増加していることが分かる。具体的にはチーム3のCPU1などである。同CPUの修了条件を満たした被験者は、実験1の閾値の場合は6名中2名であるが、実験2の閾値では6名中4名と増加している。これらのことから閾値を下げたことは、修了条件を満たしやすくすることに効果があると考えた。

これらの結果から「正答基準の変更」と「閾値の緩和」は、どちらも被験者が修了条件を満たしやすくなる変更であることが分かる。しかし、本来は、チームで伝えたい考え方を理解したプレイヤー“のみ”を運不運に左右されないよう判定することが目的である。そのため、正答の基準と閾値の両方を変更することで、想定外の被験者が修了条件を満たしてしまうようであれば、どちらか片方の変更のみを適用することが適切と考える。

### 6.6.3 テスト用 CPU の難易度

5.5.1 項の“D. テスト用 CPU の難易度”にて提示した，図 5.7 と同じく，実験 2 における，テスト用 CPU ごとの正答率のグラフを図 6.11 に示す．実験 1 で使用したテスト用 CPU と比較すると，全体的に正答率は高く，テストとして，ある程度適切な難易度だったと考える．各 CPU に関して，ウィルコクソンの符号付順位和検定を実施したところ， $\alpha = 0.025$  の有意水準において，CPU2 と CPU5 に有意差がみられた．最も正答率の伸びが良いのは CPU1 だが，バラつきが大きいせいで有意差はみられなかった．また，表 6.2 と 6.3 を見ても，質問 4 の「予測精度が上昇したと感じるか」に「はい」と答えた人数が，12 人中 2 人だったものが 20 人中 18 人に増えている．このことから，テスト用 CPU の難易度を下げたことで，主観的評価では「自分が上達している」と感じることができている被験者が増えており，モチベーションを向上させ，継続的な学習を促す観点などからも，この変更は効果的であったと考える．

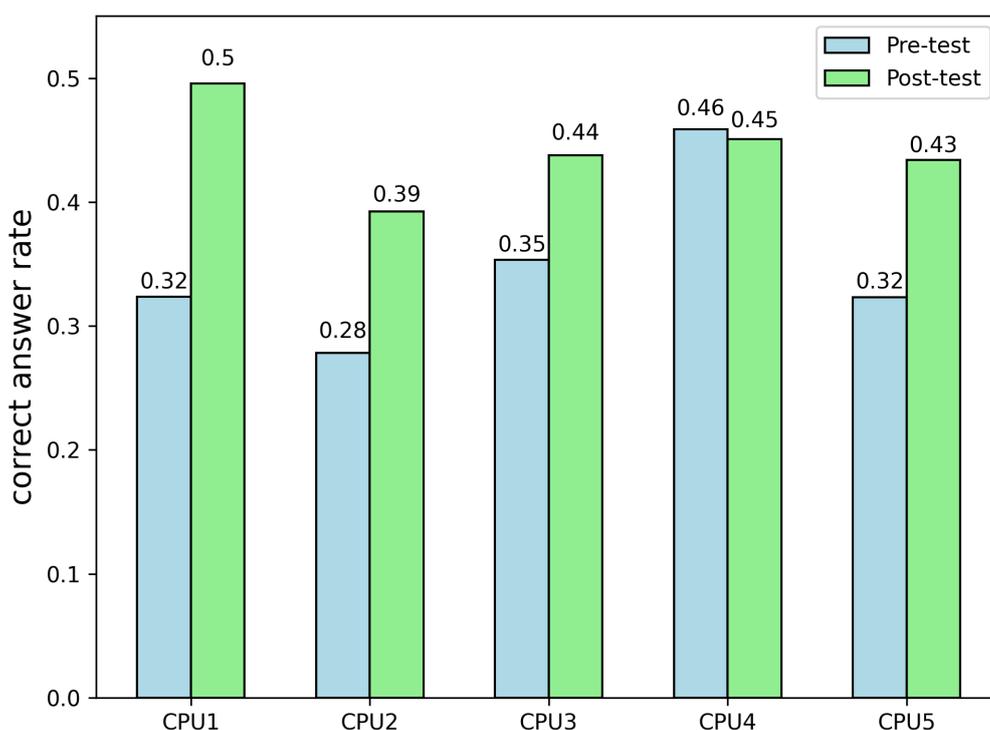


図 6.11: 実験 2 におけるテスト用 CPU ごとの正答率

6.3 節で述べた通り，テスト用 CPU1 とテスト用 CPU2 は典型的な行動規則の CPU を設定していた．しかし，実際には他 3 種の CPU とそこまで正答率の差は出なかった．これは，対戦の順番をランダムにしたことにより，テスト用 CPU5 などと対戦した後に，典型的なテスト用 CPU と対戦をした際に，深読みをしてしまい，正答率を下げた可能性があると考えた．

また、CPU4 だけは正答率が下がった。表 6.5 に CPU4 の選択肢を、図 6.12 に CPU4 の選択肢ごとの回答回数を示す。この CPU は、状況ごとにある 2 つの手から半々の確率で手を決定する行動規則であるため、正答となる選択肢群は⑦から⑩となっている。図 6.12 を見ると、プレテストの際には、④から⑥のような、分散した確率分布を選択している。この選択肢群の中には、どのような状況においても、実際の着手頻度との距離が 0.8 以下のものが存在する可能性が高い。しかし、ポストテストの際には、①から③のような、断定するような確率分布の選択が増えており、この選択肢群は、どのような状況においても、実際の着手頻度との距離が 0.8 より離れている可能性が高い。この違いにより正答率が下がったのではないかと考える。多くの対戦回数を経験したことで、ゲームに対する慣れや、自分の読みを信じ切ってしまうことなどから、①から③の選択回数がポストテストで増加したのではないかと考察した。

表 6.5: CPU4 における選択式問題の選択肢（打撃：投げ：防御）

① 10:0:0	④ 8:1:1	⑦ 0:5:5	
② 0:10:0	⑤ 1:8:1	⑧ 5:0:5	⑩ 3:3:3
③ 0:0:10	⑥ 1:1:8	⑨ 5:5:0	

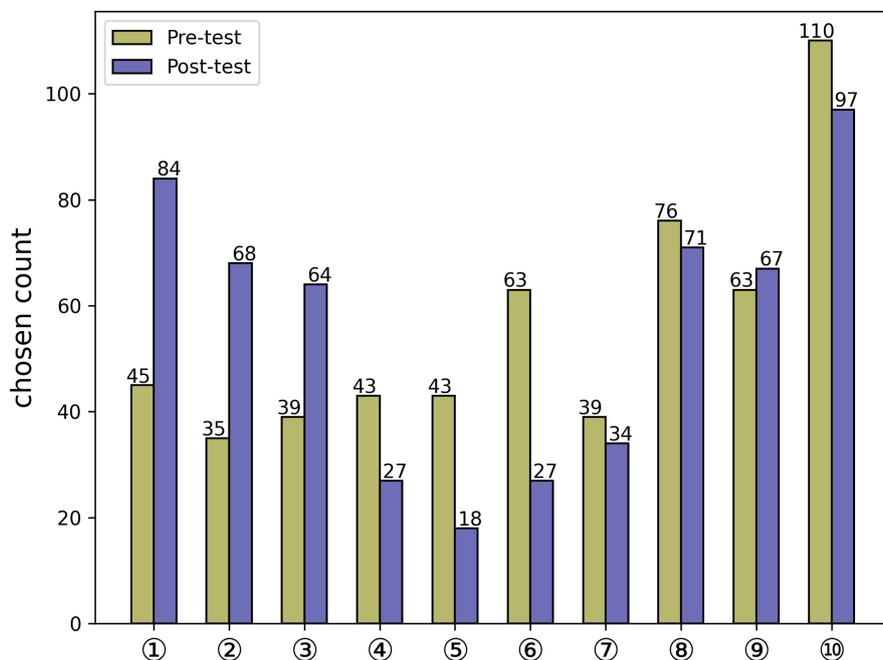


図 6.12: CPU4 の各選択肢が選ばれた回数

## 第7章 おわりに

本研究では、格闘ゲームにおける読み合いの理解に関する初心者支援として、格闘ゲームの読み合いを体験できるゲームと、そのゲーム上で行うカリキュラムを作成した。作成したゲームは、リアルタイム性やアクション性など、初心者が読み合いを行う前段落での障壁となる格闘ゲームの要素を簡略化することで、読み合いを重点的に体験することを可能とした。また、作成したゲームはあくまで、読み合いを体験するための環境であるため、別途、読み合いの考え方を伝えるための教材となる対戦相手が必要であった。そこで、読み合いに関する考え方の内、「確率で行動する相手への対処」や「過去の行動を参考にする相手への対処」など、初心者伝えるべきと判断した考え方を複数選定し、それぞれに複数の対戦相手を設定した。また、ただ対戦相手を用意し対戦させるだけでなく、対戦を行う順番を設定したり、適宜ヒントを提示するなどの支援も同時に行い、これをカリキュラムとした。

作成したカリキュラムに関して、有効性を検証するため、複数群による比較実験を実施した。実験の結果、カリキュラムを時間内に終了した被験者が居なかったことに加え、テストの結果や主観的評価なども低く、全体的に悪い結果となった。

実験を通じて得られた複数の改善すべき点に対して、「確率の偏りの低減」や「テスト用の対戦相手の易化」などの様々な変更を行い、再度カリキュラムの有効性を検証する被験者実験を行った。2度目の実験の結果、1度目の実験と比較し、テスト用の対戦相手の一部において、プレテストとポストテストの正答率に有意差が見られ、自己評価に関しても大幅に向上した。しかし、カリキュラム群内での正答率の伸びにはバラつきが大きかった。これは、カリキュラムにおけるヒント提示に日本語を使用したことなどから、日本語の得手不得手が関係しているのではないかと考えた。

今後の展望として、提示しているヒントを、平易な文言の使用や日英併記などの改善を行うことで、日本語の得手不得手に左右されないようなカリキュラムを実現することや、カリキュラムをプレイしたプレイヤーとそうでないプレイヤーとを対戦させるなど、テスト以外でカリキュラムの有効性を検証する実験を行う必要があると考えている。さらに、今回作成したカリキュラムは、主に“読み”を重点的に行うようなものとなっている。そこで、カリキュラムを通じて得られた学習者の手の選択データなどをもとに、学習者をモデル化し、そのモデルにつけ込むような対戦相手を用意することができれば、より充実した“読み合い”の学習が見込めると考える。

# 発表論文リスト

- [1] 酒見真, Hsueh Chu-Hsuan, 池田心. 格闘ゲーム初心者の為の読み合い理解支援システム, 第27回ゲームプログラミングワークショップ (GPW), pp.227-234, (2022).

# 謝辞

本論文は、筆者が北陸先端科学技術大学院大学 先端科学技術研究科 池田研究室に在籍していた期間に行った研究をまとめたものです。本論文をまとめるにあたり、池田心教授と HSUEH Chu-Hsuan 助教には、多大なる指導を賜りましたこと、心より感謝いたします。池田心教授は温和ながらも、時に厳しくご指導いただき、様々なことを学ばせていただきました。これからも、この研究室で得られた経験を糧に精進を続けていきます。HSUEH Chu-Hsuan 助教は、著者の疑問に対し丁寧に回答していただき、論文執筆の際には、著者の拙い英訳に対しても丁寧な添削をしてもらうなど、熱心なご指導をいただき大変感謝しております。お二方のご協力がなければ本論文は完成しておらず、多大なご支援とご協力に改めて感謝申し上げます。

研究に関する談義から私生活のたわいのない話まで、様々な話に花を咲かせ、大学院生活をとても有意義なものとしてくれた同研究室所属のメンバーに心から感謝いたします。特に同期である藤平啓汰君、南基大君、早下雅弘君、山田直央君、富永裕貴君と過ごした日々は、研究を行うにあたり、大きな励みとなりました。

また、被験者実験など本研究にご協力いただいた全ての方々に感謝申し上げます。最後に、これまで温かく見守ってくれた祖父と家族に感謝します。

## 参考文献

- [1] ストリートファイターV トレーニングモード. [https://www.capcom.co.jp/sfv/system\\_trainingmode.html](https://www.capcom.co.jp/sfv/system_trainingmode.html). (アクセス：2023/1/20).
- [2] GUILTY GEAR -STRIVE- 公式サイト. <https://www.guiltygear.com/ggst/jp/>. (アクセス：2023/1/20).
- [3] David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of go with deep neural networks and tree search. *nature*, Vol. 529, No. 7587, pp. 484–489, 2016.
- [4] Noam Brown and Tuomas Sandholm. Superhuman ai for multiplayer poker. *Science*, Vol. 365, No. 6456, pp. 885–890, 2019.
- [5] Adrià Puigdomènech Badia, Bilal Piot, Steven Kapturowski, Pablo Sprechmann, Alex Vitvitskyi, Zhaohan Daniel Guo, and Charles Blundell. Agent57: Outperforming the atari human benchmark. In *International Conference on Machine Learning*, pp. 507–517. PMLR, 2020.
- [6] Simon Demediuk, Marco Tamassia, William L Raffe, Fabio Zambetta, Xiaodong Li, and Florian Mueller. Monte carlo tree search based algorithms for dynamic difficulty adjustment. In *2017 IEEE conference on computational intelligence and games (CIG)*, pp. 53–59. IEEE, 2017.
- [7] 雄貴江口, 叙人藤井, 晴弘片寄. 対戦型格闘ゲームにおけるゲームバランスの自動調整. ゲームプログラミングワークショップ2021 論文集, 第2021巻, pp. 15–19, nov 2021.
- [8] Vanessa Volz, Jacob Schrum, Jialin Liu, Simon M Lucas, Adam Smith, and Sebastian Risi. Evolving mario levels in the latent space of a deep convolutional generative adversarial network. In *Proceedings of the genetic and evolutionary computation conference*, pp. 221–228, 2018.

- [9] 翠山中, シモンビエノ, 心池田. コンピュータ指導基のための悪手解説. Technical Report 5, 北陸先端科学技術大学院大学, 北陸先端科学技術大学院大学, 北陸先端科学技術大学院大学, mar 2016.
- [10] 及川大志. テトリスにおける t-spin 構成力向上のための問題生成. Master's thesis, 北陸先端科学技術大学院大学.
- [11] 近藤裕貴, 藤井叙人, 片寄晴弘ほか. 時系列パターンに着目したアクションゲームのスキル獲得支援. エンタテインメントコンピューティングシンポジウム 2015 論文集, Vol. 2015, pp. 122–126, 2015.
- [12] 2021 年の国内 e スポーツ市場規模は 78.4 億円 11 月刊行予定の「日本 e スポーツ白書 2022」の内容を先行公開. <https://jesu.or.jp/contents/news/news-221026/>. (アクセス: 2023/1/2).
- [13] ストリートファイターシリーズ総合サイト. <https://www.streetfighter.com/ja/>. (アクセス: 2022/12/10).
- [14] バーチャファイターオフィシャルサイト. <https://www.virtuafighter.jp>. (アクセス: 2022/12/10).
- [15] G. Polya. いかにして問題をとくか. 丸善, 1954.
- [16] 翼福田, 栄太中村, 克寿糸山, 和佳吉井. 楽譜簡略化と自動補完伴奏によるピアノ演奏練習支援システム. Technical Report 21, 京都大学, 京都大学, 京都大学, 京都大学, feb 2017.
- [17] 桂子中尾, 淳也森下. 語彙の重要度に基づく文構造自動書き換えシステム – 「やさしい日本語」への変換による読解支援 –. じんもんこん 2008 論文集, 第 2008 巻, pp. 133–140, dec 2008.
- [18] 高橋竜太郎, 池田心. 連鎖構成力向上のためのぶよぶよの問題作成. 情報処理学会研究報告, 2018.

# 付録A 詳細なデータ

本付録では、論文本編中に示したグラフに関して、より詳細なデータを含めたものを掲載する。

## A.1 工夫による確率的CPUでの正答率の変化の詳細なグラフ

論文本編の6.6.2項において、図6.10に各CPUの被験者ごとの1ゲーム目の正答率を示した。本節では、それらに関連する図を追加で掲載する。図A.1に図6.10を再掲し、図A.2には、真の確率分布のみを正解とした際のグラフを、図A.3にはそれらを統合した図を示す。また、図A.1と図A.2におけるグラフの横軸は、どのCPUかを表し、3-2であればターム3のCPU2を表す。図A.3の横軸はどのCPUかに加え、どの正答の基準かを表す。

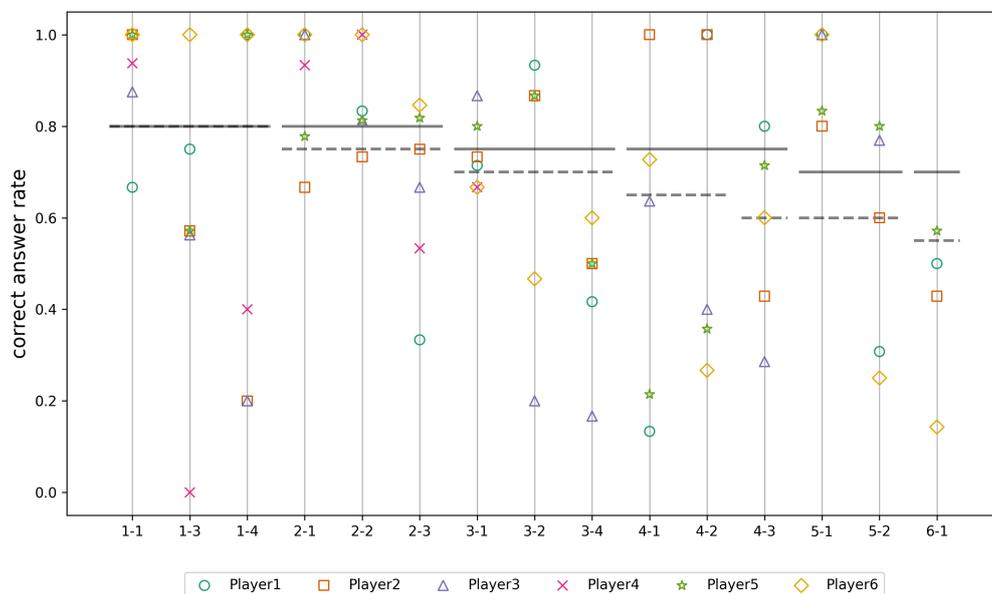


図 A.1: 実験2における各CPUの被験者ごとの正答率（1ゲーム目，再掲）  
変更後の実際の着手頻度に近いものも正解とした正答率

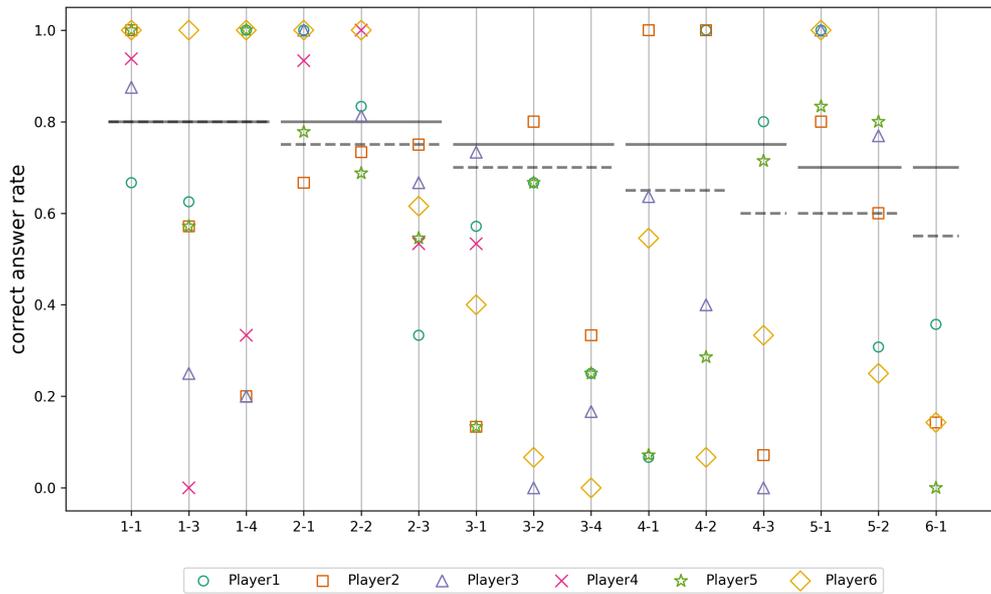


図 A.2: 実験 2 における各 CPU の被験者ごとの正答率 (1 ゲーム目) 変更前の、真の正答のみを正解とした正答率

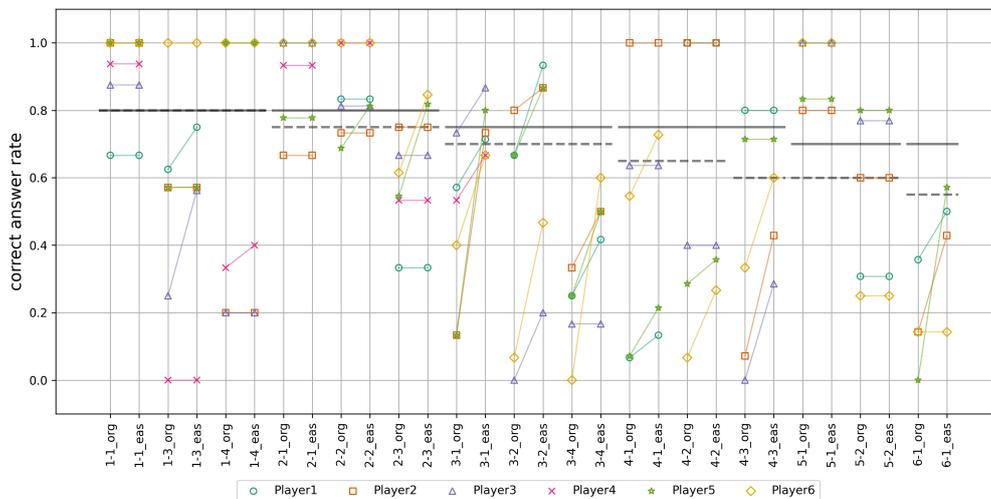


図 A.3: 実験 2 における各 CPU の被験者ごとの正答率 (1 ゲーム目) グラフの横軸はどの CPU かと、正答の基準を表したものであり、3-2\_org は、チーム 3 の CPU2 を真の確率分布のみ正解とした際の正答率を、3-2\_eas は、チーム 3 の CPU2 を実際の着手頻度も正解とした際の正答率を表す

図 A.3 を見ると、チーム 4 の CPU1 などのように、これら 2 つの変更をあわせて、修了条件を満たした被験者もみられた。このことから、正答の基準を変更することと閾値の基準を変更することのどちらも、被験者が CPU の修了条件を満たすのに効果があると考えられる。

ターム3のCPU4は、終了までの平均のゲーム数が4.4ゲームと、被験者が多くのゲーム数を費やしていた。また、図 A.3 では、全被験者の1回目のゲームが誰も閾値に達していない。そこで、詳細を確認するため、ターム3のCPU4における各被験者の正答率の推移を図 A.4 に示す。なお、図中の破線は真の確率分布のみを正答とする場合であり、実線は実際の着手頻度に近い選択肢も正解扱いとする場合の正答率である。

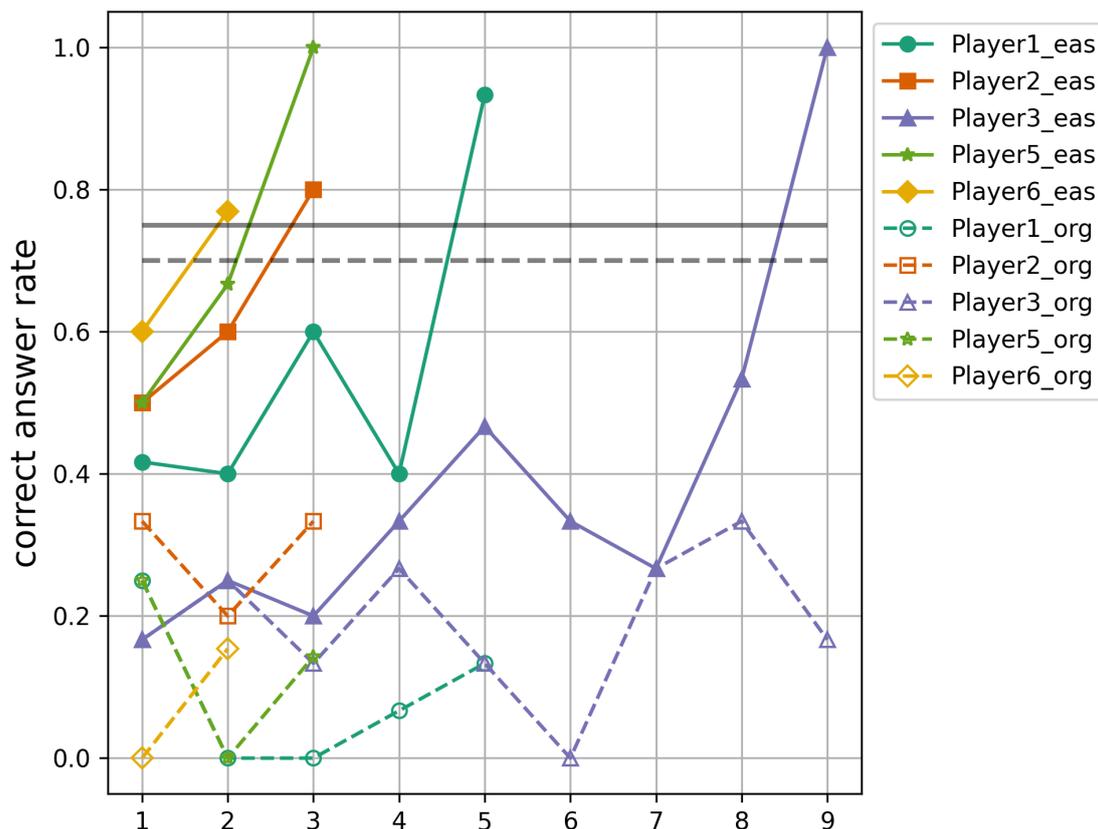


図 A.4: ターム3のCPU4における各被験者の正答率推移 (実験2)

図 A.4 において、全被験者の正答率が閾値を超えたものの、真の確率のみを正答とした場合の正答率は伸び悩んでいる。これに関して理由を考察した。まず考えられるのは、真の確率分布と実際の頻度が乖離していることである。これは、確率の偏りの低減が足りていないことで発生しうるが、6.6.1 項の表 6.4 に示す通り、今回の実験では、十分確率の偏りを低減できていると考えるため、真の確率分布と実際の頻度が乖離している可能性は低いと考えた。次に考えたのは、正答の基準変更により当てずっぽうでも正解してしまったことである。これに対する対処法は、正答に追加する距離の設定や、提示する選択肢を変更することなどがあると考えた。

## 付録B 実験に関する設定など

本付録では、実験に使用した対戦相手の行動規則と提示したヒントなどをまとめて掲載する。

### B.1 実験1で使用したCPUの行動規則一覧

本節では、実験1で使用したCPUの行動規則とヒントをまとめた表と、それぞれのCPUでの選択式問題の選択肢をまとめた表を掲載する。なお、選択式問題の選択肢は、10:0:0のように表記し、それぞれ、打撃:投げ:防御の確率を表す。3:3:3という表記は“全ての手を $\frac{1}{3}$ で選択する”という確率分布である。

表 B.1: 実験1におけるターム1のCPUの行動規則とヒント

	CPUの行動規則	提示するヒント
CPU1	打撃のみを選択	頻度や履歴に注目
CPU2	防御のみを選択	相手の高い利得の手には 負けたくない
CPU3	打撃→投げ→防御→打撃→... と周期的に手を選択	相手の手の順番に注目
CPU4	打撃→打撃→ 投げ→投げ→ 防御→防御→打撃→打撃→... と周期的に手を選択	相手の履歴と頻度に注目

表 B.2: 実験1におけるターム1のCPUの選択式問題選択肢

	選択式問題の選択肢		
CPU1~4	① 10:0:0	④ 6:2:2	⑦ 7:0:3
	② 0:10:0	⑤ 2:6:2	⑧ 3:7:0    ⑩ 3:3:3
	③ 0:0:10	⑥ 2:2:6	⑨ 0:3:7

表 B.3: 実験1におけるターム2のCPUの行動規則とヒント

	CPUの行動規則	提示するヒント
CPU1	その手で勝利した際の利得が最も高い手を選択する	相手の狙いに注目
CPU2	相手が勝利した際に最も利得の高い手に勝利する手を選択する	相手の高い利得の手には負けたくない
CPU3	基本はその手で勝利した際の利得が最も高い手を選択するが、自分が特殊手を使用できる場合は特殊手を、相手が特殊手を使用できる場合はその手に勝利する手を選択する	互いの手の利得に注目

表 B.4: 実験1におけるターム2のCPUの選択式問題選択肢

	選択式問題の選択肢		
CPU1~3	① 10:0:0	④ 6:2:2	⑦ 7:0:3
	② 0:10:0	⑤ 2:6:2	⑧ 3:7:0    ⑩ 3:3:3
	③ 0:0:10	⑥ 2:2:6	⑨ 0:3:7

表 B.5: 実験1におけるターム3のCPUの行動規則とヒント

	CPUの行動規則	提示するヒント
CPU1	打撃を8割, 投げを1割, 防御を1割で選択する	偏った確率はつけ入ることが可能
CPU2	全ての手を $\frac{1}{3}$ で選択する	無作為な相手には損を少なく立ち回る
CPU3	打撃を1割, 投げを1割, 防御を8割で選択する	偏った確率はつけ入ることが可能
CPU4	頻度が3つ全て同じ場合には 全ての手から $\frac{1}{3}$ で選択, 頻度が2つ同じ場合には それらを5割ずつの確率で選択, 頻度が1つだけ少ない場合には その手を選択する	相手の頻度と選択順に注目

表 B.6: 実験1におけるターム3のCPUの選択式問題選択肢

	選択式問題の選択肢				
CPU1	① 8:1:1	④ 6:2:2	⑦ 0:5:5		
	② 1:8:1	⑤ 2:6:2	⑧ 5:0:5	⑩ 3:3:3	
	③ 1:1:8	⑥ 2:2:6	⑨ 5:5:0		
CPU2	① 10:0:0	④ 6:2:2	⑦ 0:5:5		
	② 0:10:0	⑤ 2:6:2	⑧ 5:0:5	⑩ 3:3:3	
	③ 0:0:10	⑥ 2:2:6	⑨ 5:5:0		
CPU3	① 8:1:1	④ 6:2:2	⑦ 0:5:5		
	② 1:8:1	⑤ 2:6:2	⑧ 5:0:5	⑩ 3:3:3	
	③ 1:1:8	⑥ 2:2:6	⑨ 5:5:0		
CPU4	① 10:0:0	④ 6:2:2	⑦ 7:1.5:1.5		
	② 0:10:0	⑤ 2:6:2	⑧ 1.5:7:1.5	⑩ 3:3:3	
	③ 0:0:10	⑥ 2:2:6	⑨ 1.5:1.5:7		

表 B.7: 実験1におけるターム4のCPUの行動規則とヒント

	CPUの行動規則	提示するヒント
CPU1	最初のラウンドは防御を選択し、 2ラウンド目以降は、前のラウンドで 勝利か敗北の場合は、 前のラウンドが打撃なら投げを、 前のラウンドが投げなら防御を、 前のラウンドが防御なら打撃を選択 あいこの場合は、 前のラウンドで選択した手以外を 5割ずつの確率で選択する	前回の結果が重要
CPU2	最初のラウンドは全ての手から $\frac{1}{3}$ で 選択し、2ラウンド目以降は、 前のラウンドで勝利した場合は、 前のラウンドと同じ手を選択 敗北の場合は、 前のラウンドが打撃なら投げを、 前のラウンドが投げなら防御を、 前のラウンドが防御なら打撃を選択 あいこの場合は、 前のラウンドで選択した手以外を 5割ずつの確率で選択する	前回の結果が重要
CPU3	最初のラウンドは全ての手から $\frac{1}{3}$ で 選択し、2ラウンド目以降は、 前のラウンドで勝利した場合は、 前のラウンドで選択した手以外を 5割ずつの確率で選択 敗北の場合は、 前ラウンドのプレイヤーと CPUの手から半々で選択 あいこの場合は、 前のラウンドと同じ手を選択する	前回の結果が重要

表 B.8: 実験1におけるターム4のCPUの選択式問題選択肢

	選択式問題の選択肢			
CPU1~3	① 10:0:0	④ 0:5:5	⑦ 6:2:2	
	② 0:10:0	⑤ 5:0:5	⑧ 2:6:2	⑩ 3:3:3
	③ 0:0:10	⑥ 5:5:0	⑨ 2:2:6	

表 B.9: 実験1におけるチーム5のCPUの行動規則とヒント

	CPUの行動規則	提示するヒント
CPU1	プレイヤーの直前の手に 勝利する手を選択	直近の手を参考に つけ込もうとしてくる
CPU2	プレイヤーの最頻手に勝利する手を選択 最頻手が2つの場合は, それらとあいこ以上になる手を選択 最頻手が3つの場合は, チーム2のCPU3と同じ方法で 手を選択	つけ込む為に過去の手全体に 注目してくる
CPU3	プレイヤーの直近3回の手の中での 最頻手に勝利する手を選択 最頻手が2つの場合は, それらとあいこ以上になる手を選択 最頻手が3つの場合は, チーム2のCPU3と同じ方法で 手を選択	過去の全手より, 近いラウンドの手に つけ込もうとしてくる

表 B.10: 実験1におけるチーム5のCPUの選択式問題選択肢

	選択式問題の選択肢		
CPU1~3	① 10:0:0	④ 0:5:5	⑦ 6:2:2
	② 0:10:0	⑤ 5:0:5	⑧ 2:6:2    ⑩ 3:3:3
	③ 0:0:10	⑥ 5:5:0	⑨ 2:2:6

表 B.11: 実験 1 におけるターム 6 の CPU の行動規則とヒント

	CPU の行動規則	提示するヒント
CPU1	<p>プレイヤーの最頻手に勝利する手を確率で選択する</p> <p>最頻手が 1 つの場合はそれに勝利する手を 7 割, 最頻手と同じ手を 3 割で選択する</p> <p>最頻手が 2 つ以上の場合はそれらとあいこ以上となる手を選択</p> <p>最頻手が 3 つの場合は, ターム 2 の CPU3 と同じ方法で手を選択</p>	<p>つけ込む為に過去の全手を参照した上で確率的に選択することでつけ込まれることを防いでくる</p>
CPU2	<p>プレイヤーの直近 3 回の手の中での頻度に合わせて確率で手を選択する</p> <p>直近 3 回の手の内訳が 3 つの手全てを使用している場合には, ターム 2 の CPU3 と同じ方法で手を選択する</p> <p>直近 3 回の手の内訳が 1 つの手のみ選択している場合には, その手に勝利する手を選択する</p> <p>直近 3 回の手の内訳が “打撃, 打撃, 投げ” のように, ある 2 つの手が 2:1 の場合には, 2:1 の手にそれぞれ勝利する手を 7:3 の確率で選択する</p>	<p>つけ込む為に最近の手を参照した上で確率的に選択することでつけ込まれることを防いでくる</p>

表 B.12: 実験 1 におけるターム 6 の CPU の選択式問題選択肢

	選択式問題の選択肢		
CPU1, 2	① 10:0:0	④ 7:0:3	⑦ 7:3:0
	② 0:10:0	⑤ 3:7:0	⑧ 0:7:3    ⑩ 3:3:3
	③ 0:0:10	⑥ 0:3:7	⑨ 3:0:7

## B.2 実験2で使用したCPUの行動規則一覧

本節では、実験2で使用したCPUの行動規則とヒントをまとめた表と、それぞれのCPUでの選択式問題の選択肢をまとめた表を掲載する。なお、選択式問題の選択肢は、10:0:0のように表記し、それぞれ、打撃:投げ:防御の確率を表す。3:3:3という表記は“全ての手を $\frac{1}{3}$ で選択する”という確率分布である。

実験2においては、修了条件を満たした際ゲームプレイヤに、CPUの行動規則を答えとして提示するようにしており、実際にゲームプレイヤに提示した文言は、本付録に掲載されているCPUの行動規則の文言となっている。

表 B.13: 実験 2 におけるチーム 1 の CPU の行動規則とヒント

	CPU の行動規則	提示するヒント
CPU1	打撃のみを選択	①相手の手の頻度か履歴に注目 ②相手はずっと同じ手ばかり出している
CPU3	打撃→投げ→ 防御→打撃→... の順番で手を選択	①相手の手の順番に注目 ②手を出す順番を見極めよう
CPU4	打撃 2 回→ 投げ 2 回→ 防御 2 回→打撃 2 回→... で手を繰り返して選択	①相手の履歴と頻度に注目 ②相手が手を変えるタイミングと 変える先を予測しよう

表 B.14: 実験 2 におけるチーム 1 の CPU の選択式問題選択肢

	選択式問題の選択肢			
CPU 1, 3, 4	① 10:0:0	④ 6:2:2	⑦ 0:5:5	
	② 0:10:0	⑤ 2:6:2	⑧ 5:0:5	⑩ 3:3:3
	③ 0:0:10	⑥ 2:2:6	⑨ 5:5:0	

表 B.15: 実験 2 におけるターム 2 の CPU の行動規則とヒント

	CPU の行動規則	提示するヒント
CPU1	勝利した時に最も残り歩数を減らせる手を選択	①相手の狙いに注目 ②相手は一定の法則で手を出す
CPU3	相手の手に負けた時に 増える残り歩数が少ない手を選択	①消極的な考え方をする相手 ②相手は負けた時の損失を考えている
CPU4	普段は勝った時に最も残り歩数を減らせる手を プレイヤーが特殊手を使える時は それに負けない手を選択	①互いの手が勝った時の歩数に注目 ②CPU 自身の特殊手は強気 プレイヤーの特殊手には弱気な相手

表 B.16: 実験 2 におけるターム 2 の CPU の選択式問題選択肢

	選択式問題の選択肢				
CPU1~3	① 10:0:0	④ 6:2:2	⑦ 7:0:3		
	② 0:10:0	⑤ 2:6:2	⑧ 3:7:0	⑩ 3:3:3	
	③ 0:0:10	⑥ 2:2:6	⑨ 0:3:7		

表 B.17: 実験 2 におけるターム 3 の CPU の行動規則とヒント

	CPU の行動規則	提示するヒント
CPU1	打撃と投げと防御を 8:1:1 の確率で選択	①相手の頻度に注目 ②偏った確率には つけ入ることが可能
CPU2	全ての手を $\frac{1}{3}$ の確率で選択	①無作為な相手には損を少なく 立ち回る方が良い ②確率の偏りはあれど 選択肢の確率分布と同じ手の出し方
CPU4	頻度が全て同じ場合はランダム 頻度が少ない手が 2 つの場合は 半々で選択 頻度が少ない手が 1 つの場合は それを選択	①相手の頻度と選択順に注目 ②相手は頻度を揃えるように 手を決定している

表 B.18: 実験 2 におけるターム 3 の CPU の選択式問題選択肢

	選択式問題の選択肢			
CPU1	① 8:1:1	④ 5:2.5:2.5	⑦ 0:5:5	
	② 1:8:1	⑤ 2.5:5:2.5	⑧ 5:0:5	⑩ 3:3:3
	③ 1:1:8	⑥ 2.5:2.5:5	⑨ 5:5:0	
CPU2	① 10:0:0	④ 8:1:1	⑦ 0:5:5	
	② 0:10:0	⑤ 1:8:1	⑧ 5:0:5	⑩ 3:3:3
	③ 0:0:10	⑥ 1:1:8	⑨ 5:5:0	
			⑩ 3:3:3	
CPU4	① 10:0:0	④ 6:2:2	⑦ 0:5:5	
	② 0:10:0	⑤ 2:6:2	⑧ 5:0:5	⑩ 3:3:3
	③ 0:0:10	⑥ 2:2:6	⑨ 5:5:0	
			⑩ 3:3:3	

表 B.19: 実験 2 におけるターム 4 の CPU の行動規則とヒント

	CPU の行動規則	提示するヒント
CPU1	前のラウンドで CPU が、 負けか勝ちなら矢印の先の手 (打撃なら投げ, 投げなら防御) を, あいこなら前のラウンドで 出さなかった手を半々で選択 (前回打撃なら投げと防御)	① 前回ラウンドの勝敗で どの手を出すかが決まる ② 確率が絡むのはあいこの時のみ ③ あいこの時はある手を除き 半々で手を選ぶ
CPU2	前のラウンドで CPU が、 勝ったら前のラウンドと同じ手を, 負けたら矢印の先の手 (打撃なら投げ, 投げなら防御) を, あいこなら前のラウンドで 出さなかった手を半々で選択 (前回打撃なら投げと防御)	① 勝ってたら楽観的, 勝てないと 手を変える相手 ② 前回があいこの時は 確率的に手を選ぶ ③ あいこの時はある手を除き 半々で手を選ぶ
CPU3	前のラウンドで CPU が、 勝ったら前のラウンドで 選ばなかった手から半々で (前回打撃なら投げと防御), 負けたら前のラウンドの手と それに勝つ手から半々で (前回打撃なら打撃と防御), あいこなら前のラウンドと同じ手 を選択する	① CPU は前のラウンドが 勝ちか負けなら確率的に手を選ぶ ② 勝った時と負けた時は半々で, ある手から選んでいるが, どの手から半々で選ぶかに違いがある

表 B.20: 実験 2 におけるターム 4 の CPU の選択式問題選択肢

	選択式問題の選択肢				
CPU1, 3	① 10:0:0	④ 8:1:1	⑦ 0:5:5	⑧ 5:0:5	⑩ 3:3:3
	② 0:10:0	⑤ 1:8:1	⑨ 5:5:0		
	③ 0:0:10	⑥ 1:1:8			
CPU4	① 10:0:0	④ 6:2:2	⑦ 0:5:5	⑧ 5:0:5	⑩ 3:3:3
	② 0:10:0	⑤ 2:6:2	⑨ 5:5:0		
	③ 0:0:10	⑥ 2:2:6			

表 B.21: 実験2におけるターム5のCPUの行動規則とヒント

	CPUの行動規則	提示するヒント
CPU1	プレイヤーの前の手に勝利する手を選択する	①プレイヤーの直近の手を参考に手を決める ②プレイヤーの前の選択を受けて手を決めている ③プレイヤーの前の手とCPUが選んでくる手に注目
CPU2	プレイヤーの頻度が最も高い手には負けない手を選択 (打撃と投げが最高頻度の場合は、打撃なら両方にあいこ以上なので打撃を選択、最高頻度が1つならそれに勝つ手を選択) プレイヤーの頻度が全て同じならランダムで手を選択	①プレイヤーが良く選ぶ手に勝とうとする ②プレイヤーの最頻の手に勝利するように手を選んでくる ③最頻の手が無い時はランダム

表 B.22: 実験2におけるターム5のCPUの選択式問題選択肢

	選択式問題の選択肢			
CPU1~3	① 10:0:0	④ 6:2:2	⑦ 0:5:5	
	② 0:10:0	⑤ 2:6:2	⑧ 5:0:5	⑩ 3:3:3
	③ 0:0:10	⑥ 2:2:6	⑨ 5:5:0	

表 B.23: 実験2におけるターム6のCPUの行動規則とヒント

	CPUの行動規則	提示するヒント
CPU1	プレイヤの最頻の手とその手に 勝利する手を, 3:7で選択する (プレイヤの最頻の手が打撃なら, 打撃と防御を3:7で選択) 最頻の手が2つの場合は それらに負けない手を選択 (プレイヤの最頻手が打撃と防御なら あいこ以上になる防御を選択) 最頻の手が3つの場合は前回と同じ手	①プレイヤの手の頻度を見るが 確率的に手を変える ②プレイヤの最頻の手が1つなら その手に勝利する手を含め 確率で手を選択する ③プレイヤの最頻の手が3つなら 前回と同じ手

表 B.24: 実験2におけるターム6のCPUの選択式問題選択肢

	選択式問題の選択肢		
CPU1~3	① 10:0:0	④ 6:2:2	⑦ 7:3:0
	② 0:10:0	⑤ 2:6:2	⑧ 0:7:3    ⑩ 3:3:3
	③ 0:0:10	⑥ 2:2:6	⑨ 3:0:7