2020 年度　卒　業　論　文

機械学習によるビヘイビアツリーの強化に関する研究

指導教員：渡辺 大地 教授

メディア学部 ゲームサイエンスプロジェクト
学籍番号　M0117319
陳　方建

2021 年 2 月

2020 年度　　卒　業　論　文　概　要

論文題目

機械学習によるビヘイビアツリーの強化に関する研究

メディア学部 **学籍番号：M0117319**	**氏名**	陳　方建	**指導教員**	渡辺 大地 教授

キーワード	強化学習、ビヘイビアーツリー、AI、Q － Learning

テレビゲームなどにおいて AI 技術を活かして、多様性また複雑なキャラクター行動を実現した作品が出現した。現在では、コンピューター性能の進化と AI 理論の開発、機械学習が今の主流になっている。機械学習とは機械に大量のデータからパターンやルールを学習することでさまざまな物事に利用することで判別や予測をする技術のことである。本研究では、その背景の下で、今多くのゲームに使っている行動決定手法ビヘイビアーツリーと機械学習分野の一つ強化学習、また Q － Leading という言葉を使用する場合が多いが、それを利用することで、より正しい選択する研究である。ビヘイビアーツリーは今キャラクターの行動決定手法の主流になっている。しかしビヘイビアツリーはキャラクター AI を処理する時、簡単な AI を処理するのでいいが、複雑な AI を処理する時、ノード行動選択の外れかルート選択時間をかかるというデメリットがある。そのため本研究では、ビヘイビアツリーをメインにして、更に機械学習を用いて、より正しい動きを選択する研究である。まず、Q － Learning アルゴリズムを利用して Reward 表でビヘイビアツリー各自の報酬を決定する。そして Q アルゴリズムを利用することで、各自の最終報酬を Q 値を算出することで、最終の Q 値を完成する。そしてビヘイビアーツリーは行動を選択する時に、Q 値の報酬に設定した値の価値が高いルートを発見ことは本研究の目的である。本提案手法を用い、ビヘイビアーツリーは行動を選択する時に、Q 値の報酬に設定した値の価値が高いルートを発見ことができた。

目 次

図 目 次

第 1 章

はじめに

1.1　背景と目的

近年、AI を使って自動運転 [1] や自動翻訳 [2] など人に便利になる技術が大量に出現した。AI は人工知能計算という概念とコンピュータという道具を用いてコンピューターに人間のように思考する、計算する計算機科学の一分野を指す言葉である。言語の理解や推論、問題解決などの知的行動を人間に代わってコンピューターに行わせる技術である。計算機による知的な情報処理システムの設計や実現に関する研究分野の言葉も使用している。テレビゲームなどにおいて AI 技術を活かして、多様性また複雑なキャラクター行動を実現した作品が登場した。ゲーム AI ではゲームを開発する上で使っている言葉で、もともとの目的としてゲーム中でのキャラクターを人の知性を持っているかのように振る舞う対象に対して使っている。しかし、現在ではゲームのシナリオなどもゲーム AI が管理してユーザーをプレイする時、より多様性のプレイ方法になっている。例として 1980 年代の有名なゲーム作品、パックマン [3] がある。このゲームでは、パックマンの敵キャラクターに世界で初めてキャラクター AI が導入することで、ゲームする時それぞれの敵キャラクターは独立した存在で異なる個性持っており、ステージ内をそれぞれ独自のパターンで動いている。

そして現在では、コンピューター性能の進化と AI 理論の開発で機械学習が今の主流になっている。機械学習とは機械に大量のデータからパターンやルールを発見して、それをさまざまな物事に利用することで判別や予測をする技術のことである。近年もっとも代表的なのは AlphaGo[4][5] である。AlphaGo は、グーグル傘下の DeepMind 社によって開発された人工知能コンピューターである。AlphaGo の最大の特徴は、機械学習分野の一つニューラルネットワークを応用していることである。人間が設定した評価経験則に従うのではなく、人間の棋譜のデータを元に、コンピューターが自分自身との対戦を数千万回にわたり繰り返すことで強化していく。この際モンテカルロ木探索と呼ばれる探索アルゴリズムを組み合わせたことも AlphaGo の特徴の一つである。そして 2015 年 10 月に、当時ヨーロッパチャンピオンだった樊麾に勝利し、大きな注目を集めた。その後、2016 年には、世界大会で 18 回も優勝した経験を持ち、名実ともに世界トップレベルの棋士であった韓国のイ・セドルに五番勝負で 4 勝 1 敗という戦績で勝利し、その実力が人類トップレベルのプレイヤーを凌駕するものであると証明した。

本研究では、その背景の下で、今多くのゲームに使っている行動決定手法ビヘイビアツリーと機械学習分野の一つ強化学習を利用することで、より正しい選択することを目的とする。ビヘイビアツリーは今キャラクターの行動決定手法の主流になっているが、しかしビヘイビアツリーはキャラクター AI を処理する時、簡単な AI を処理する時には扱いやすい手法だが。複雑な AI を処理する時、行動選択の外れ、選択時間がかかるというデメリットがある。そのため本研究では、ビヘイビアツリーを利用して、ビヘイビアツリーの中で機械学習を用いて、より正しい動きを選択する研究である。Q 値の取得方法で行う Q 学習研究がいくつか提案している。例えば、馬野元秀ら [6] の研究、浅沼駿哉ら [7] の研究と Liu ら [8] の研究などがある。本研究は最初 Q − Learning を利用して Reward 表でビヘイビアツリー各自の報酬を決定する。そして Q アルゴリズムを利用することで、各自の最終報酬を Q 値を算出することで、最終の Q 値つまり最終の Reward 表を完成する。そしてビヘイビアーツリーは行動を選択する時に、Q 値の報酬に設定した値の価値が高

いルートを発見ことは本研究の目的である。

評価は、Q アルゴリズムを利用して完成したゲームを対戦する、Q 学習アルゴリズムの学習回数を設置して、学習で得られた Q 値をビヘイビアツリーに反映して、ビヘイビアツリーをルート選択する時、Q 値の高いルートを選択できることを評価する。その結果本提案手法を用い、ビヘイビアーツリーは行動を選択する時に、Q 値の報酬に設定した値の価値が高いルートを見つけ出すことができた。

1.2 論文の構成

本論文の構成は、2 章では現状調査について述べる。3 章では提案手法について述べる。4 章では検証とその結果について述べる。5 章では本研究のまとめについて述べる。

第 2 章

先行手法

本章では研究手法に元にした、強化学習とビヘイビアーツリーについて記述する。

2.1 ビヘイビアツリー

ビヘイビアツリー [9] は当時多くの AI に使用していた行動決定手法有限状態マシン [10](Finite State Machine,FSM) と階層有向限状機マシン [11](Hierarchical Finite StateMachines, HFSM) を改良した技術である。一般的にはゲームの規模に合わせて大きくなると、FSM のメンテナンスすることが困難となる。FSM では状態ごとに現在に対応しているエージェントの 1 つの状態なので、エージェントが同時に 2 つ状態を取ることは不可能である。階層 FSM も同様の欠点を持つが、状態を割り付けるツリーノードでは非常にはっきりとした表示方法である。

ビヘイビアーツリーと HFSM の根本的な違いは、HFSM における各ノードにある 1 つの状態を表し、ビヘイビアーツリーのノードは 1 つのタスクを表す。まだはゲーム論理はますます複雑になり、ビヘイビアーツリーのいくつかの位置に単一のノートを追加することができる。一つのビヘイビアーツリーのモジュール化と再利用可能性はビヘイビアーツリーを非常に良くなる強力な AI (Artificial Intelligence 人工知能) 決策ツールとなる。特定のビヘイビアーツリーはエージェントの任意の状態に直接依存しないことによって、同一エージェントは複数のビヘイビアー

ツリーを同時に実行することができ、複数開発することが可能になる。

有限状態マシンはキャラクターのエラーを許容しないが、しかし現実の時 AI が間違えることはないとは言えない。そのために FSM は知能的ではなく、非知能的だと認められる。ビヘイビアーツリーに重み付きの選択ノードと順番ノードがあるため、重みを合理的に配置できる値は合理的なランダム性をうまく実現できる。ビヘイビアツリーはゲームの AI だけでなく他にもロボットの行動決定 [12] に用いられることなど もあるが、本論文ではデジタルゲームのキャラクター AI に用いられるビヘイビアツリーを対象としている。キャラクター AI に用いられるビヘイビアツリーの研究がほかいくつか提案している。例えば、義澤 [13] の研究や Wensheng ら [9] の研究がある。

以下の図 2.1 は、簡単なビヘイビアツリーの例である。

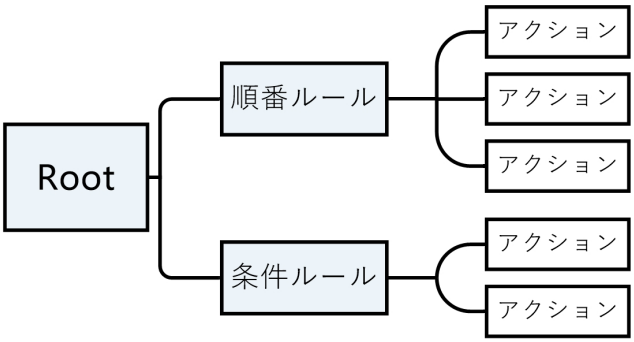

図 2.1　簡単なビヘイビアツリーの例

通常のビヘイビアツリー、一つのビヘイビアツリーは一つの AI のロジックである。このロジックを実行するために、最初の Root ノートからビヘイビアツリーをトラバースする。トラバースする時、親ノートは自分自身の形に応じて、子ノートを実行するかどうかを決める。子ノートは実行完了後、実行の結果は親ノートにフィードバックをする。実行の結果は通常には 3 種に分けている。

1. Fail　ノートの実行結果は失敗の場合（例えば、ノート条件判定する時、結果は False の場合、ノート実行失敗など)。
2. Success　ノートの実行結果は成功の場合（例えば、ノート条件判定する時、結果は True の場合、ノート実行成功など)。
3. Running　ノートは今実行中の場合（例えば、今動画プレイしている、今目標に向かって走っているなど)。

実際ゲーム AI をする時、上記の三つ以外にも他の実行結果が存在する場合がある。また、内容に多少差がある場合もあるが、本研究では上記の 3 つのみを対象として使用する。

本論文ではビヘイビアツリーを評価するために下記の 2 つの選択ルールを対象として使用する。

1. 条件ルール

   条件ルールの実行方式は、左から右に順に全ての子ノードを実行する。子ノードが 失敗の結果をフィードバックする時次の子ノードを続けて実行する、一方子ノードが 成功または実行中の結果をフィードバックする時次の子ノードの実行を停止する。成功また実行中を結果としてフィードバックする時、親ノードに 成功また実行中という結果をフィードバックする。そうでなければ、 失敗の結果を親ノードにフィードバックする。図 2.2 は条件ルールの例である。条件ルールに 2 つの子ノートが持っている、条件ルールの条件による、どっちの子ノートを実行する。

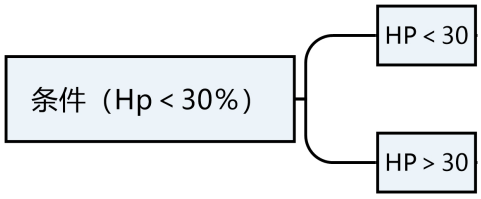

図 2.2　条件ルールの例

2. 順番ルール

順番ルールの実行方式は、左から右に順に全ての子ノードを実行する。子ノードが 成功の結果をフィードバックする時次の子ノードを続けて実行する、一方のノードが 失敗と実行中の結果にフィードバックする時次の子ノードの実行を停止する。すべてのノードが 成功の結果をフィードバックする場合のみ、 成功の実行結果を親ノードにフィードバックする。図 2.3 は順番ルールの例である。順番ルールに 3 つの子ノートが持っている、順番で子ノートを実行する。

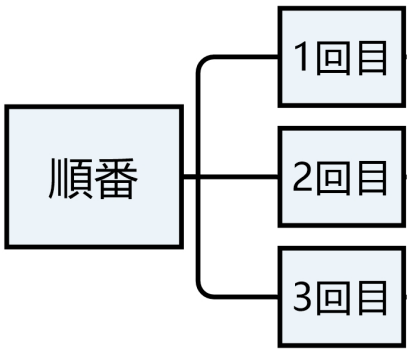

図 2.3　順番ルールの例

ビヘイビアーツリのルールには上記以外に他の選択ルールが存在する場合がある。例えば飾るルール、並列ルール、ランダムルールなど存在する場合がある。本研究では上記の 2 つのみを対象として使用する。

2.2 Q 学習

Q 学習は強化学習の代表的な手法であり、最初は行動心理学の研究から始まった。1911 年 Thorndike[14] が効用の法則 law of effect を提唱して以来、1954 年に Minsky が初めて強化と強化学習という概念と用語が提唱した。1965 年にウォルツと傅京孫は制御理論の中でこの概念を提唱し、賞罰による学習の基本的な考え方を発表した。1957 年にベルマン [15] はマルコフ決定過程 mdp の確率的離散バージョンである最適制御問題を解く動的計画法を提案したこの方法の解法は強化学習のトライとエラー反復解法のような仕組みを採用している。今の強化学習では一連の行動の結果として報酬が得るような状況での学習に用いている。Q 学習では行動と状態の組に対して与えている Q 値を報酬に応じて更新する。行動と状態が離散の場合ではルックアップテーブル形式で Q 値を記憶しており、Q 値の大きさに応じた行動選択が行う。テーブル形式での離散的な Q 値の記憶は行動や状態の数に比例した記憶容量の増大し、汎化能力の欠如に伴う学習時間の増大し、行動や状態が連続変数である場合への拡張の困難さなどの問題点がある。このような問題点に対処するため、Q 値の記憶をルックアップテーブル以外の方法で行う Q 学習がいくつか提案している。例えば、階層型ニューラルネットワーク [16]、基底関数 [17]、ファジールール [18][19] などに基づく Q 学習がある。

Q 学習アルゴリズムはモデルとは無関係な強化学習である。このアルゴリズムの基本モデルはエージェントが周囲の環境を感知して、学習に通じて最適化な選択を達成するための机械学習方法の一種である。Q 学習は遅延報告、探索、部分状態観察、終生学習のメリットがある。Q 学習の一般的なモデルにしたものは図 2.4 に示す。一つの Agent が環境 S の中にある、アクションを実行することで環境 S に作用し、環境 S は Agent の動作を受信してその状態が変化すると同時に Q 学習システム報酬値にフィードバックする。つまり Q 学習は Agent の報酬値を最大化することが原則です。それで Q 学習による unsupervised learning [20] の自己適応能力とその行動動作

の自律性を実現できる。

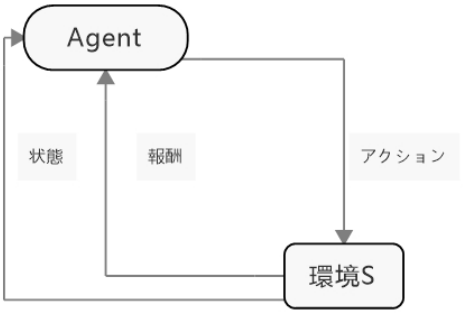

図 2.4　Q 学習基本モデル

本研究で Q 値を所得するために基本ルールを以下のルールで示す。

1. Q 値を求める為に計算式を設定する。基本の形は

   Q(状態、アクション)=R(状態、アクション)　γ × max(Q(次の状態、すべてのアクション))。それに公式にしたものは

$$Q(s,a) = R(s,a) + \gamma \times max(Q(s',A)) \tag{2.1}$$

2. Q(s,a) は評価関数であり、状態 s からの為に動作実行時の最大換算累積報酬として a を用いる、 s は当前の状態であり、a は今のアクションである。
3. R(s,a) は今の報酬表であり、 s は当前の状態の報酬であり、a は今のアクションの報酬である。
4. γ（ガンマ）パラメータと報酬を決める。 γ は割引率と呼ぶこともので範囲は 0〜1 である。0 に近いほど目先の報酬を重視する傾向となる。今回 γ の設定は 0.8 とする。
5. max(Q(s′,A)) は評価関数であり、Q(s′,A) の s′ は次の状態であり、A は次の状態の全て実行可能なアクションである。その値の最大値 max を取得する。

6. 学習を反復して、学習終了ごとに Q 値表を記録する。

7. 1 つの学習は、ランダムに指定された状態から始まり、最大の報酬に到着した時点で終了する。

8. Q 学習計画流れを図にしたものは図 2.5 で示す。

（a）ランダムに全部の状態どこかを探索開始位置とする。

（b）現在の状態から移動できるアクションの中から 1 つを選択する。

（c）Q の値の式で計算して、Q 値表に更新する。

（d）移動先が目的地なら学習終了。そうでない場合は移動先の状態を現在の状態にする。

（e）上記 b から繰り返す。

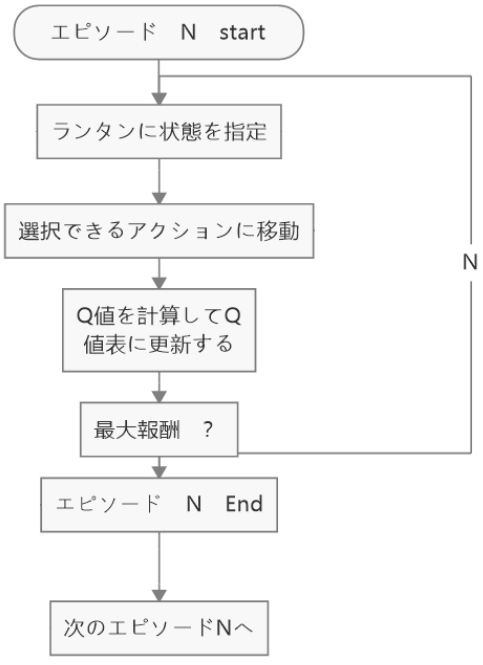

図 2.5　Q 学習の処理流れ

2.3 問題点

Q 学習にはいくつかの問題点がある。例えば Q 学習による理論的保証は値の収束性のみであり、収束途中の値には具体的な合理性が認められないため、価値反復法の Q 学習は方策勾配法と比べると学習途中の結果を近似解として用いにくい為。また、パラメータの変化に敏感でありその調整に多くの手間が必要である。

ビヘイビアツリーは近年使用されることが増えてきている。その理由はビヘイビアツリーを用いることで比較的簡単に AI を作れることが挙げられる。ビヘイビアツリーを用いることでプログラマーだけではなく、ゲームデザイナーやプランナーがキャラクター AI を調節するまだは制作することが可能になる。

しかし、簡単に AI を作ることができるビヘイビアツリーにも問題点がいくつかある。その一つとして、各アクションノードで手作業する時キャラクター AI の規模が大きくなるとビヘイビアツリーのノードも複雑になる問題がある。ノードを調節しなければならない時、ビヘイビアツリーのアクションノートが複雑になるとノードの調節とデバッグが難しくなる。ビヘイビアツリーの学習枠組みが不足し、ヘイビアーツリーの設計は効率的ではないことになる。特にビヘイビアツリーの AI を調節することは容易ではない。その最たる理由として挙げるのは、ビヘイビアツリーのどの部分を調節する必要があるのか分からない点である。図 2.6 はビヘイビアツリーのアクションノードの複雑な例を図にしたものである。

ビヘイビアツリーの調節方法として考えられるのは大きく分けて 2 つある。1 つはビヘイビアツリーの構造自体を変更することである。もう 1 つは中間ノードの選択ルールや条件を変更することである。しかし、この 2 つの方法はビヘイビアーツリー自体からの調節しかできないという点がある。このようにこのように調節する箇所を発見するだけでも困難なビヘイビアツリーのノード選択を行いやすくするために、本論文では近年流行っている機械学習の一種手法 Q −

Leading を使用して、ビヘイビアーツリーの状況で Q 学習で得られた Q 値による高い値を発見する手法を提案する。

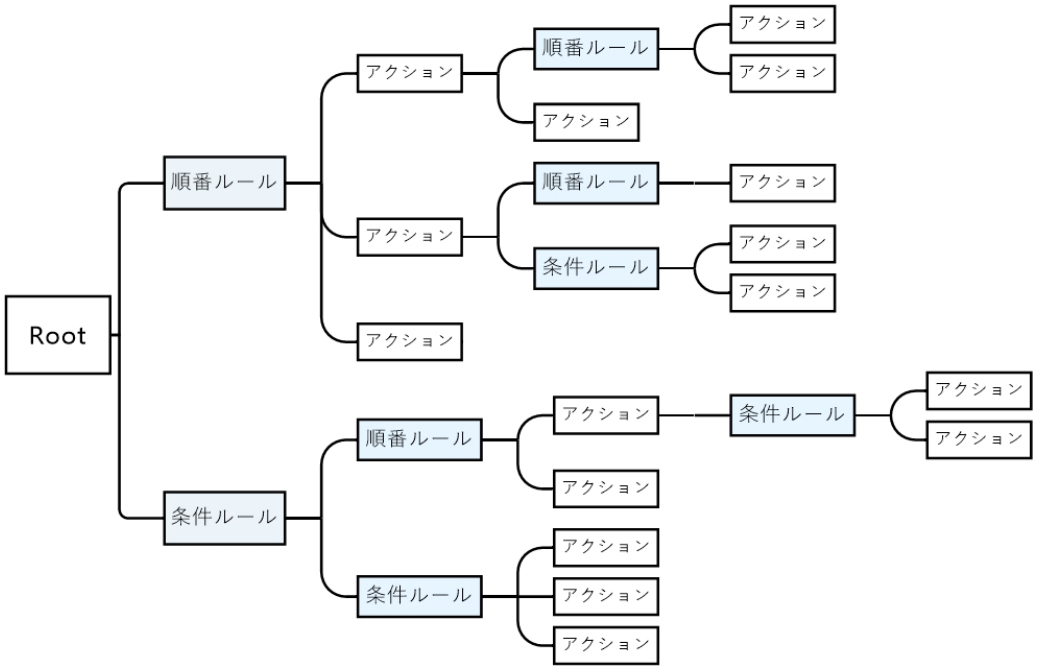

図 2.6　ビヘイビアツリー調節複雑な例

第 3 章

提案手法

本章では、本研究で提案する手法機械学習におけるビヘイビアツリーの強化ついて述べる。

本手法ではビヘイビアツリーの中で学習完成した最終の Q 値表を反映して、学習で得られた Q 値による高い値を発見することの考え方を提案する。

3.1 提案手法の Q 値

最終の提案手法の Q 値表の完成表現はビヘイビアツリーの親ノードは Q 値表の状態になる。子ノートが Q 値表のアクションになる。例として Q 学習で得られた Q 値表は表 1 のように状態 1 はアクション 1、アクション 2 を持っている。各自学習で得られた Q 値は 3 と 2 である。

これを表にしたものは表 3.1 で示す。

表 3.1 最終の完成した Q 値表

Q 値表	アクション	
状態 1	アクション 1	アクション 2
学習で得られた Q 値	3	2

3.2 ビヘイビアツリーへの変換

図 3.1 は Q 値表をビヘイビアツリーへの変換にするものである。

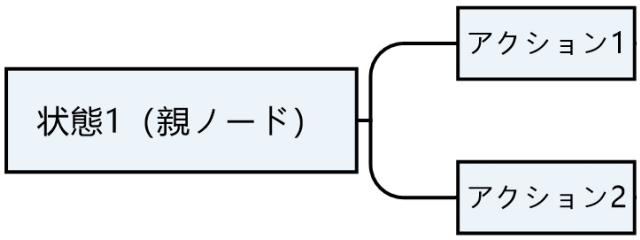

図 3.1　Q 値表をビヘイビアツリーへの変換

状態 1 はビヘイビアツリーの親ノードになる。アクション 1 、アクション 2 はビヘイビアツリーの子ノードになる。

ビヘイビアツリーをノードを選択する時に Q 値の高いノードを選択するものとする図 3.2 にその要素を示す。

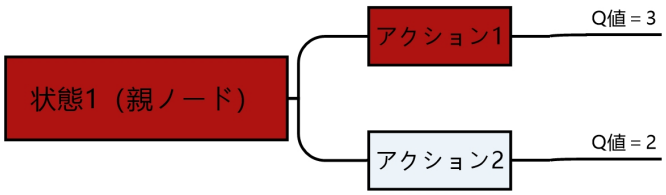

図 3.2　赤色についている所は今 Q 値による、報酬が一番高いルート

3.3　最終のビヘイビアツリー

最終 Q 学習で得られた Q 値表をビヘイビアツリーに反映して、Q 値の報酬に設定した値の価値が高いルートを選択するものとする図 3.3 にその要素を示す。そして提案手法を完成する。

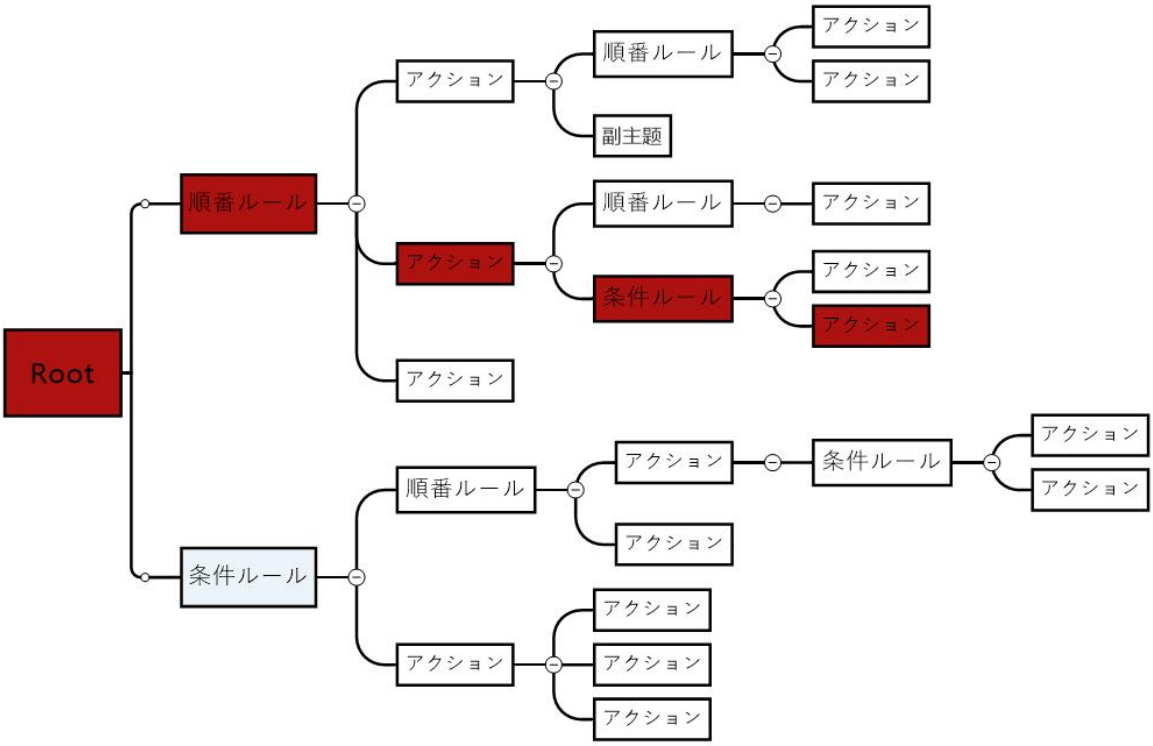

図 3.3　Q 値を反映したビヘイビアツリー

図 3.3 赤色についている所は今 Q 値による、報酬が一番高いルートである。

第 4 章

検証と結果

4.1 概要

本章では、提案手法を用いてビヘイビアーツリーは行動を選択する時に、Q 値の報酬に設定した値の高いノードを発見すること出来るかを検証した。提案手法を用いて、実際に Q アルゴリズムを利用して価値が高いルートを発見することできた。

4.2 制作したゲーム

検証で用いる作成したゲームについて簡単に記述する。作成したゲームは Unity を用いて作成して FPS ゲームにおいて敵 AI とプレイヤーを戦闘する。敵の AI 行動と Q 学習報酬の設置は以下で記述した設定で実装した。それによって提案手法の検証を行う。

4.3 ビヘイビアーツリーの設置

本研究で提案した手法を評価するために、シーディングゲームのキャラクター AI を元に、ビヘイビアツリーの各種状態の設置する。ゲームには敵対両方が設置した。敵対両方には今回同じ行動ツリーを共有する。しかし、行動ツリーには各自パラメータの設置が違う為に、キャラクター各自の個性を持って、単純な動きを避けて、多様性を確保することになる。キャラクターの行動

ツリーを下記の 5 つで設置する。

1. パトロール行為

   パトロール行為は最も基本的な行動であり、idle の状態でキャラクターがデフォルトで行う行動である。ゲームがオンになればキャラクターはパトロール状態になる。それは具体的な設定によって、まず自分の近くの地図にパトロールを行い、敵がいなければ、ランダムにいくつかの目印に向かう経路探索を実行する。それでも敵が見つからなければ、次のランダム表示地点へと進んでいく。すべての地点を捜索した後、敵が見つからなかったとすれば、今回の捜索はなかったことになる。その原因が不用意に敵とすれ違っていた可能性があれば、新たな捜索を開始する。その為、知能体は数回の捜索をした後、必ず敵に出会い戦闘となり、デッドループに入ることを避けることができる。

2. 攻撃行為

   攻撃行為では、まず敵に移動して攻撃範囲を確保する。移動している時、移動に関するノードが Running にフィードバックする敵が射程範囲に入れば、自分の現在の動作を完了した後 Success にフィードバックして、次の行動ノードに進むことになる。

3. 退避行為

   退避行為では、自分自身の HP が 30 %以下になると、近くのシェルターを探して。退避行為を行う。

4. 追いかけ行為

   追いかけ行為では、敵が逃げる場合が、まず自分自身の HP を確認する、HP が 30 %以上の場合は敵を追いかける。

5. 回復行為

   回復行為では、シェルターに到着すれば、HP が一秒 1 %を回復する。

以下の図 4.1 はキャラクタービヘイビアツリーの設置である。

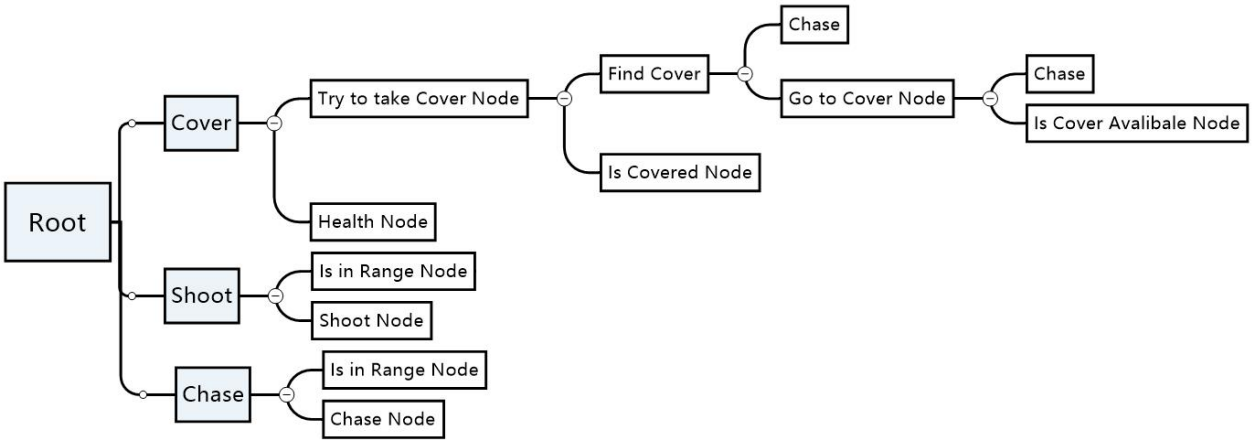

図 4.1　ビヘイビアツリーによるキャラクター各自ノードの設置

4.4　Q 学習報酬の設置

今回 Q 学習報酬の設置はキャラクターの HP、今距離一番近いカバーへの距離 Is Covered、目標への距離 Is Range、今ダメージを受けているか Is Attacked、四つの状態で設置する。表 4.1 は Q 学習報酬の設置表である。

表 4.1　報酬の設置　* を付けているところに対して奨励は 0 として設置する

Q 値表	Health	Is Range	Is Covered	Is Attacked	cation	reward
statel	None	*	*	*	*	0
state2	L	*	*	True	Find Cover	80
state3	L	*	N/M	False	Find Cover	90
state4	H/M	N/M/F	*	*	Chase	90
state5	H/M	N/M/F	*	False	Shoot	100
state6	H/M	N/M/F	*	True	Find Cover	95

上記の表 4.1 各自パラメータの説明下記通りとする。

1. HP(Node,low,Medium,high)
2. 今ダメージを受けている:Is Attacked (yes,no)

3. 距離最近のカバー:Is Covered(None,near,Medium,Far)

4. 目標との距離: Is Range(None,near,Medium,Far)

5. 表 4.1 HP 中での None は HP ＝ 0 の意味である、 距離は None の時距離が遠いまだは見つからない状態の意味のである。図中の L ＝ low、M ＝ Medium、H ＝ high、N ＝ near、F=Far の設定をしている。

4.5 評価手法

今回提案手法を評価するために最初本手法がビヘイビアツリーのノードを評価する際に使用する行動選択は有効と無効があるとした。

4.6 評価手法ー有効

ビヘイビアツリーを行動選択する時、Q 値に高いルートを選択する時有効選択とする。その時、提案手法の評価が有効である。以下の図 4.2 はビヘイビアツリーを行動選択する時、 Q 値の高い値を選択する図である。

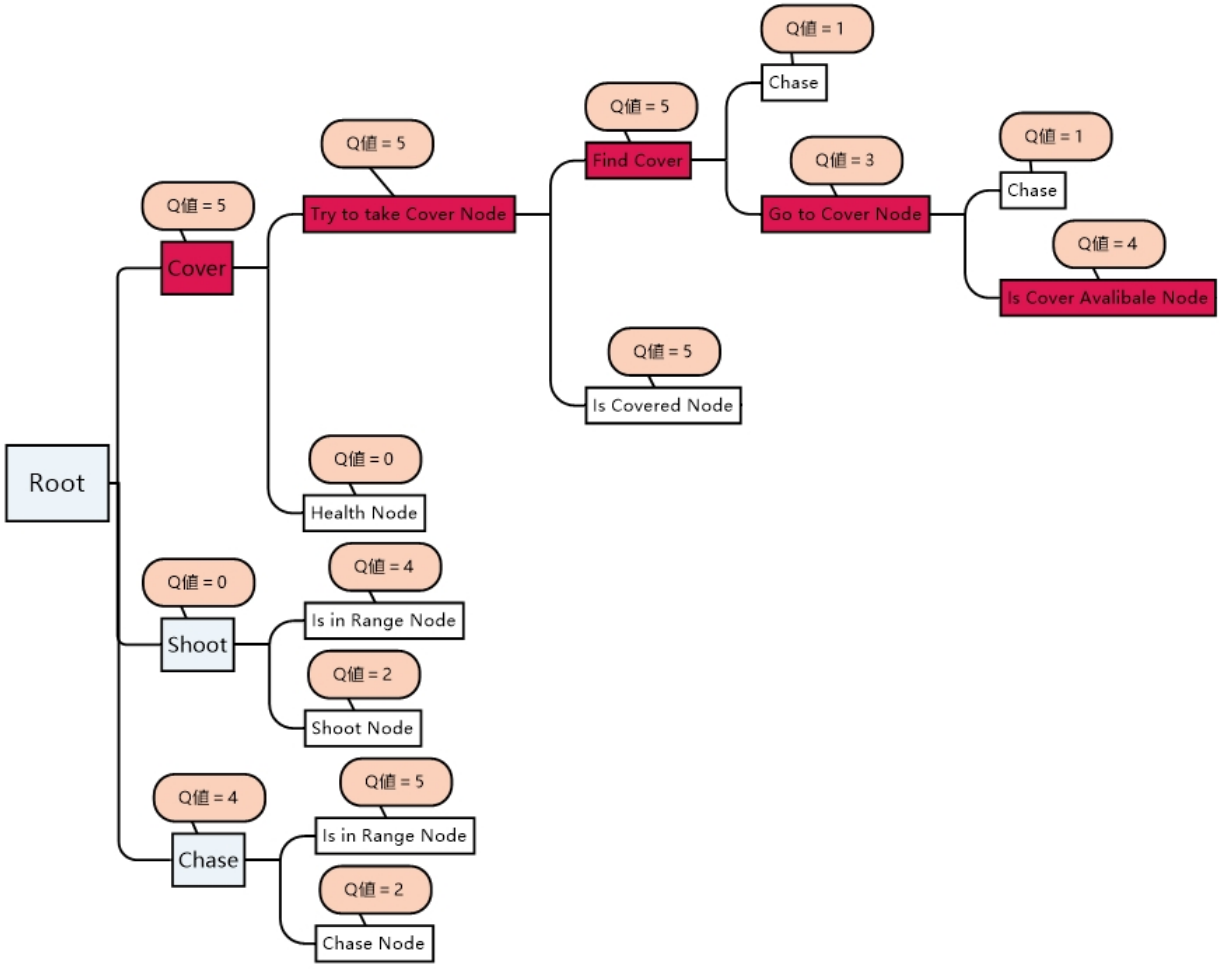

図 4.2　赤色についている所は今 Q 値による、報酬が一番高いルートである

4.7　評価手法ー無効

ビヘイビアツリーを行動選択する時、 Q 値の高い値を選択してない時。その時、提案手法の評価が無効である。以下の図 4.3 はビヘイビアツリーを行動選択する時、 無効選択する図である。

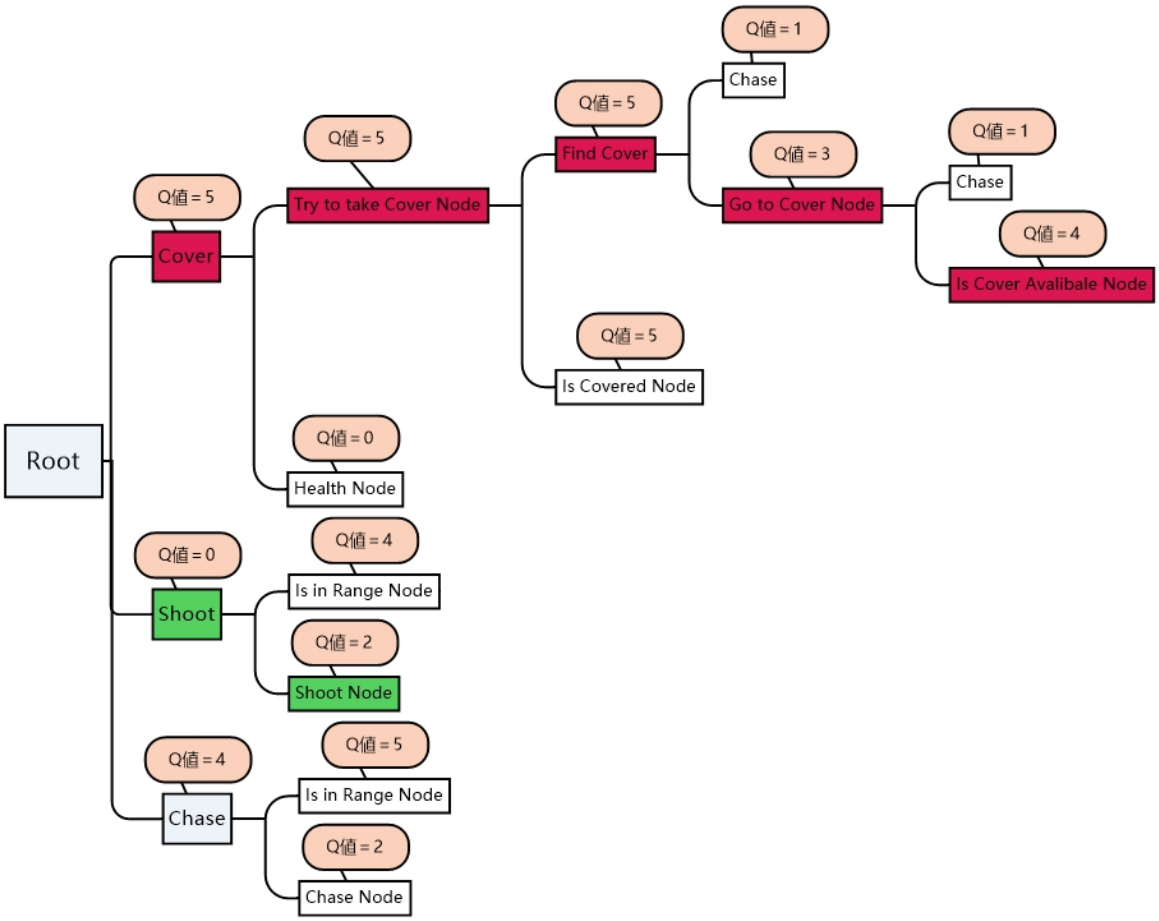

図 4.3　赤色についている所は今 Q 値による、報酬が一番高いルートである。緑色についている所は今 Q 値による、無効選択である

今回提案手法を評価するために複雑なビヘイビアツリーは設置してない為、Q 学習の学習回数は大きく設置していない。今回の Q 学習回数回数 100 を設置した。キャラクター最初の HP は 100 %とする。

4.8　実験結果 1

実験結果 1 はキャラクター最初の HP は 100 %の時、Q 値一番高いルートはゲームスタートした時、最初敵はキャラクターを探することは一番いいの選択である。つまりエージェントは防衛より攻撃のほうが優先的である。図 4.4 赤色についている所は今 Q 値によるビヘイビアツリー一

番選択いいのルートである。図 4.5 エージェントを提案手法としての有効性の検証の実行画像である。

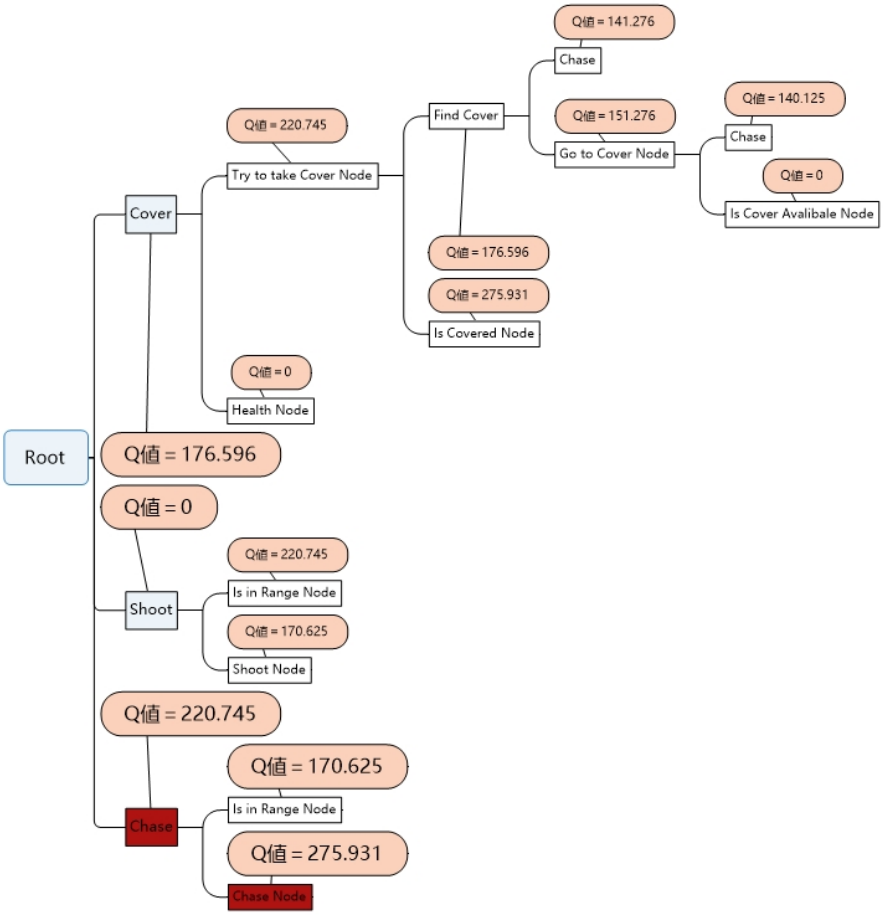

図 4.4　赤色についている所は今 Q 値による、報酬が一番高いルートである

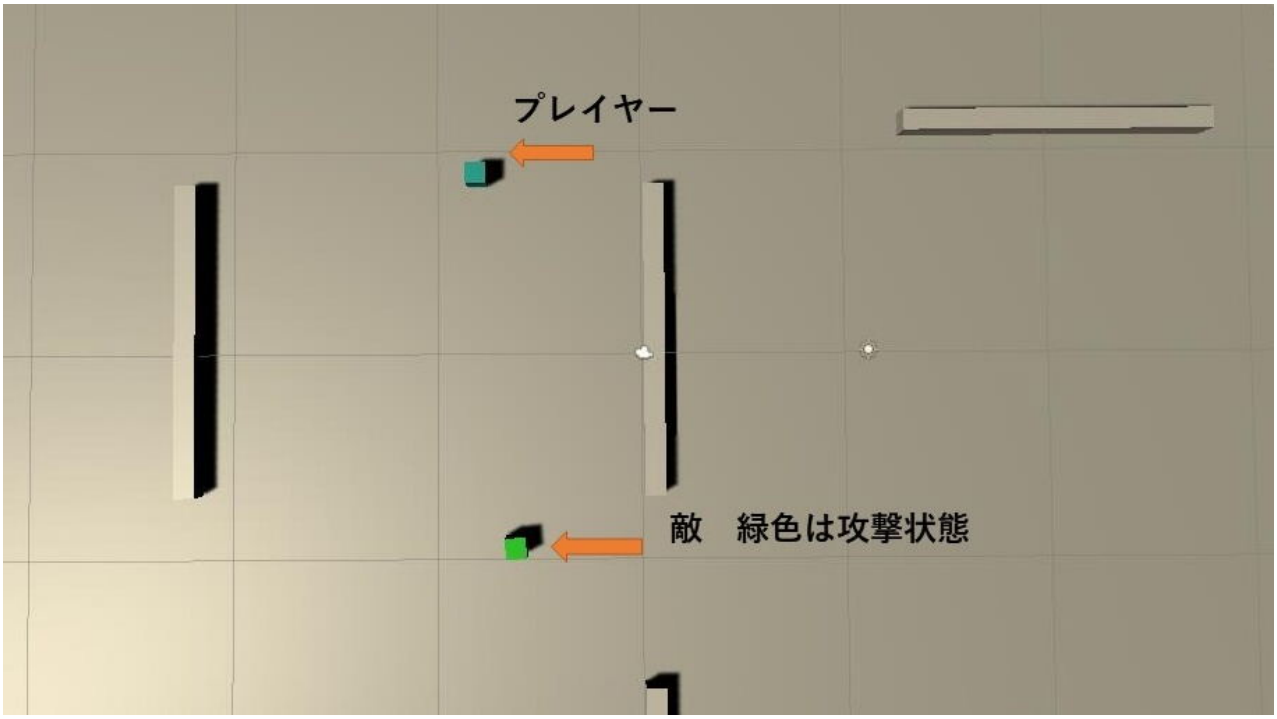

図 4.5　実験結果 1 の実行画像

4.9　実験結果 2

実験結果 2 はキャラクターの HP は 30 %以下の時。Q 値一番高いルートはゲームスタートした時、最初敵はカバーを探すこと一番いいの選択である。つまりエージェントは攻撃より防衛のほうが優先的である。図 4.6 赤色についている所は今 Q 値によるビヘイビアツリー一番選択いいのルートである。図 4.7 エージェントを提案手法としての有効性の検証の実行画像である。

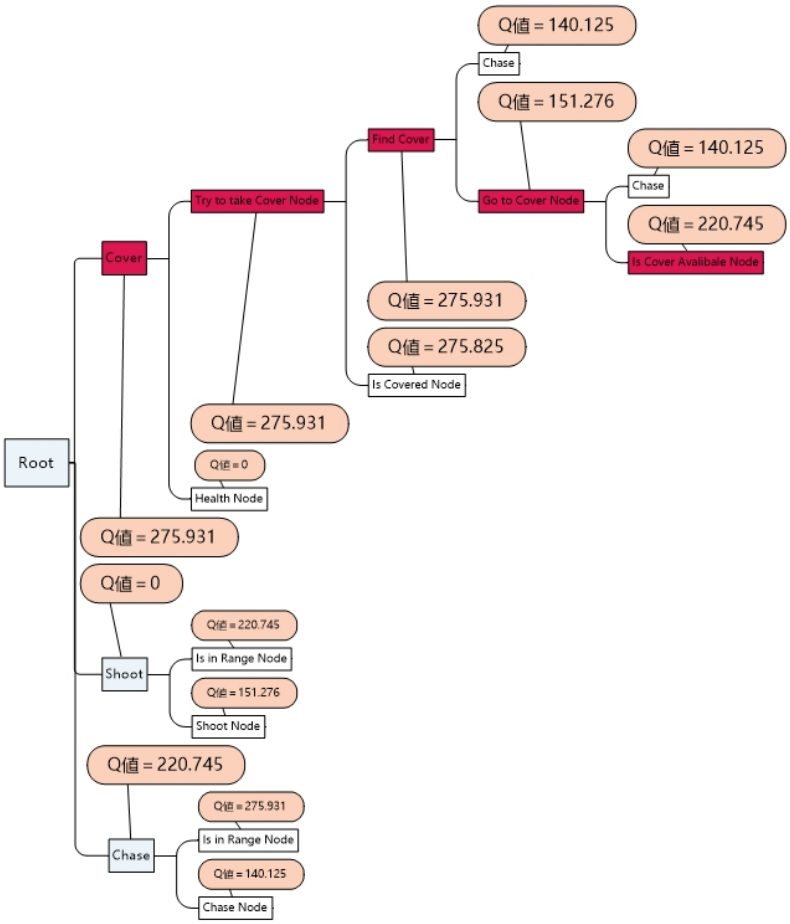

図 4.6　赤色についている所は今 Q 値による、報酬が一番高いルートである

図 4.7　実験結果 2 の実行画像

4.10　提案手法の最終結果

今回の提案手法を使用して、ビヘイビアツリーに評価した時。ゲームをする時、HP の各自の設置で最初の評価が有効になることを証明した。しかし、ゲームを進行時、提案手法による Q 値のリアルタイムの更新が出来てない為、提案手法による対戦評価の有効性が評価ができてない。これは提案手法の不足である。

4.11　考察

本手法によって、ビヘイビアーツリーは行動を選択する時に、Q 値の報酬に設定した値の価値が高いルートを発見すること出来てるが。実際ゲームにした時、ビヘイビアツリー実行した時のプライオリティを衝突したことがある。そのためビヘイビアツリーをルート選択した時 Q 値の報酬に設定した値の価値が高いルートを選択した時、エラーが発生したことがある。それにより本

手法不足しているところがあるということが分かった。その為さらなる機能を追加する必要があると考察する。

第 5 章

まとめ

デジタルゲームは昔と比べ複雑化しそれに合わせて AI も複雑になった。そこで本研究は今のキャラクター AI 行動決定ツールの一つビヘイビアツリーと機械学習の一種 Q 学習を使用して、本研究を行った。本手法は機械学習手法の一つ Q 学習を用い、ビヘイビアーツリーは行動を選択する時に、ビヘイビアツリーに Q 学習で得られた Q 値による高い値を発見することを提案した。本手法によって、ビヘイビアツリーは行動を選択する時、提案手法によるビヘイビアツリーのルート選択は発見できた。実際ゲームにした時、ビヘイビアツリー実行した時のプライオリティを衝突したことがある。そのためビヘイビアツリーをルート選択した時 Q 値の報酬に設定した値の価値が高いルートを選択した時、無効選択が発生したことがある。それにより本手法不足しているところがあるということが分かった。これは今後検討するところである。

謝辞

本研究を進めるにあたってご指導いただいた先生方、先輩方やプログラミングを協力にしてくれた友人たちに感謝いたします！

参考文献

[1] 津川定之. 自動運転システムの展望. *IATSS review*, Vol. 37, No. 3, pp. 199–207, 2013.

[2] 中村哲, 隅田英一郎, 清水徹ほか. 多言語自動通訳技術の実現に向けて: 2. ここまできた音声翻訳技術. 情報処理, Vol. 49, No. 6, pp. 606–610, 2008.

[3] 岩谷徹, 聞き手, 三宅陽一郎, 構成, 高橋ミレイほか. アーティクル ゲーム ai の原点『パックマン』はいかにして生み出されたのか?: 岩谷 徹インタビュー. 人工知能, Vol. 34, , 2019.

[4] Fei Yue Wang, Jun Jason Zhang, Xinhu Zheng, Wang Xiao, and Liuqing Yang. Where does alphago go: From church-turing thesis to alphago thesis and beyond. Vol. 3, No. 2, pp. 113–120, 2016.

[5] 囲碁の最強人工知能 AlphaGo（アルファ碁）の仕組み. `https://tech-camp.in/note/technology/32855/`.

[6] 馬野元秀, 立野宏樹, 伊瀬顕史. カーレースゲームへのファジィ q 学習の適用:—次の目標の通過しやすさを優先した学習—. 日本知能情報ファジィ学会 ファジィ システム シンポジウム 講演論文集, Vol. 29, pp. 231–231, 2013.

[7] 浅沼駿哉, 長名優子ほか. 負の報酬を獲得する状況を重視した deep q-network. 第 82 回全国大会講演論文集, Vol. 2020, No. 1, pp. 561–562, 2020.

[8] Yu Tao Hu Xi-bing Liu, et al. Multi-objective optimal power flow calculation based on

multi-step q (λ) learning algorithm. *Journal of South China University of Technology (Natural Science)*, Vol. 38, No. 10, p. 139, 2010.

[9] Xu Wensheng, Wu Bo, and Jiang Jianhong. Design and realization of behavior tree in weapon equipment virtual maintenance training system. *Journal of System Simulation*, Vol. 30, No. 7, p. 2722, 2018.

[10] Wu Huayao and Deng Wenjun. Research progress on the development of microservices. *Journal of Computer Research and Development*, Vol. 57, No. 3, p. 525, 2020.

[11] 惠良和隆, 三宅陽一郎. Ai 技術のゲームコンテンツへの適応. 映像情報メディア学会誌, Vol. 63, No. 9, pp. 1218–1223, 2009.

[12] Lin Yi-Lun, DAI Xing-Yuan, LI Li, WANG Xiao, and WANG Fei-Yue. The new frontier of ai research: generative adversarial networks. *Acta Automatica Sinica*, Vol. 44, No. 5, pp. 775–792, 2018.

[13] 義澤勇輝, 阿部雅樹, 渡辺大地ほか. ベイズ理論を用いたビヘイビアツリーの 中間ノードの評価に関する研究. ゲームプログラミングワークショップ 2019 論文集, Vol. 2019, pp. 195–197, 2019.

[14] Edward L Thorndike. The law of effect. *The American journal of psychology*, Vol. 39, No. 1/4, pp. 212–222, 1927.

[15] Liu Fa-gui Mai Wei-peng and Huang Kai-yao. Design and implementation of stochastic model algorithm for dynamic power management. *Journal of South China University of Technology (Natural Science)*, Vol. 35, No. 9, p. 60, 2007.

[16] L. J. Lin. Reinforcement learning with hidden states. In *Proc.2nd International Conference on Simulation of Adaptive Behavior*, 1993.

[17] 橋本博幸. 基底関数を用いた q-learning による強化学習の考察. 第 39 回システム制御情報

学会研究発表講演会講演論文集, 京都, pp. 69–70, 1995.

[18] Il Hong Suh, Jae-Hyun Kim, and FC-H Rhee. Fuzzy-q learning for autonomous robot systems. In *Proceedings of International Conference on Neural Networks (ICNN'97)*, Vol. 3, pp. 1738–1743. IEEE, 1997.

[19] L Jouffe. Comparison between connectionist and fuzzy q-learning. In *Proc. of the 4th International Conference on Soft Computing*, pp. 557–560, 1996.

[20] Horace B Barlow. Unsupervised learning. *Neural computation*, Vol. 1, No. 3, pp. 295–311, 1989.