修士論文

人間プレイヤを活躍させる協力型ゲームの味方 AI

板東 宏和

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

In recent years, artificial intelligence (AI) techniques have made significant progress and have been widely used in various applications. In some applications, AI agents are employed to replace human resources where an example is self-driving cars. Digital games are also a field with significant progress in AI. In digital games, to create good AI players as humans' alternatives, it is required to make the AI players as strong as humans. In this sense, creating strong AI players has been a research topic where plenty of achievements were obtained.

Among digital games, AI players as teammates are employed in cooperative multiplayer games when human players cannot form a team. Therefore, cooperation between the teammate AI and the human players is important in this game genre. There are various behaviors required for cooperation, including cooperative behaviors in order to achieve the main goal of the game, cooperative behaviors that fit other players' senses of values, and human-like behaviors.

However, for beginners and intermediate players, it is not always the most desirable to have AI players that "successfully achieve the main goal of the game in an environment with human players". More specifically, even if the players can win as a team, human players may not feel interested because they cannot "play active roles". This paper aims to create teammate AI that supports human players to let them play active roles even with sacrifice to achieve the main goal. For this purpose, we believe that human players' intentions, such as "wanting to attack" or "wanting to survive", should be respected and that cooperative actions should be made to support human players' intentions.

In this study, we employ DungeonEscape, a game where three players try to escape from a dungeon by defeating a dragon. We extend the game so that the above-mentioned intentions or emphasized behaviors are easier to occur and be recognized. Then, we (1) consider typical intentions that may occur in this extended game and create an intention agent that acts according to each of the intentions, (2) create a support AI for each of the intention agents to support the agent's intention, (3) create a predictor that predicts intentions from the actions of intention agents, and (4) use the predictor to predict human players' intentions and provide the support AI corresponding to the predicted intentions as teammates.

For (1), we first considered that attack-oriented, survival-oriented, efficiency-oriented, and peer-oriented are representative intentions. In more detail, we defined attack-oriented players as those who tend to take the initiative in attacking enemies, survival-oriented players as those who tend to give priority to being alive, and efficiency-oriented players as those who try to clear the game as quickly as possible. The specific goal of this research is to support these players. We then created AI agents with these intentions so that we could create support AI for

(2) and train intention predictors for (3). Ideally, it was better to use human players' data to train the intention predictors since we want to support human players. However, since it was difficult to collect data from human players due to cost, we created the intention agents using reinforcement learning with rewards corresponding to the intentions.

Next, we conducted experiments and confirmed that the three created intention agents indeed acted in response to the corresponding intentions. Compared to the attack rate of 0.33 for the default agent with no intentions, the attack rate for the attack-oriented agent increased to 0.41. For the survival-oriented agent, we also used the attack rate as the evaluation. The reason was that players in DungeonEscape die if they attack the dragon. In other words, the players are likely to survive if they do not attack the dragon. Thus, the attack rate reflects the survival intention to some extent. The attack rate for the survival-oriented agent decreased to 0.01, which was much lower than the default agent's 0.33. For the efficiency-oriented agent, the clear time of the game decreased by about 12%.

For (2), we trained intention-supporting AI, whose goal is to support the actions of intention agents. During training, two instances of the intention-supporting AI were paired with an intention agent. We fixed the intention agent and used reinforcement learning to train the intention-supporting AI. We designed the rewards for reinforcement learning in a way that the AI receives high rewards when the intention agents find them in favorable situations. For example, attack-supporting AI receives high rewards when attack-oriented agents attack the dragon. By doing so, the intention-supporting AI learns behaviors that assist the intention agents in achieving their goals. Taking the same example of the attack-supporting AI, moving away from the dragon and giving up to attack are desired behaviors.

In our evaluation experiments, we demonstrated the effectiveness of the intention-supporting AI by comparing two cases: an intention agent paired with the default agents and the same intention agent paired with the corresponding intention-supporting AI. The attack-oriented agent paired with the attack-supporting AI had an attack rate of 0.89, which was higher than 0.41 when paired with the default agents. The survival-oriented agent paired with the survival-supporting AI had an attack rate of 0.005, which was lower than 0.012 when paired with the default agents. As for efficiency-oriented agents, we consider that efficiency-oriented agents themselves serve as the efficiency-supporting AI. When the efficiency-oriented agents were paired together, the clear time of the game was reduced by nearly 10%.

For (3), we trained a binary classifier to predict whether the intention is attack-oriented or survival-oriented. We used game logs collected from intention agents as the training data. In more detail, the inputs to the classifier contained the

coordinates of the player's position, the coordinates of the dragon's position, and the distance between the player and the dragon. The accuracy of the classifier was 0.905, even with a simple neural network.

For (4), by combining the intention classifier and the intention-supporting AI, we created an integrated intention-supporting AI that provides the intention-supporting AI corresponding to the predicted intention as teammates. Evaluation experiments on the integrated AI showed that the AI was generally able to correctly support the player's intentions, with an attack rate of 0.77 for the attack-oriented agent and an attack rate of 0.01 for the survival-orientated agent. Furthermore, we verified the robustness of the integrated AI by including fluctuations and delays in the evaluation, taking into account that our final goal is to support human players. The results showed that it was difficult for the integrated AI to let human players play active roles. However, the results also showed that as long as the players had some clear intentions, such as attack-oriented and survival-oriented, the integrated AI was able to provide the corresponding support.

概要

昨今，人工知能（AI）技術は大きな発展を遂げており，様々な用途で広く利用されている．自動運転車など，AI 技術は人間の代わりに用いられることも多い．中でもディジタルゲームにおいて，人間の代替として良いゲーム AI には，人間と並ぶ強さが求められる．そのため，強いゲーム AI は研究の対象となっており，これまで十分な成果を上げてきた．

ディジタルゲームの中でも，協力型マルチプレイヤゲームでは，人間プレイヤ同士でチームを組めないような場合に味方プレイヤとしてゲーム AI が必要である．そのため，このようなゲームジャンルではゲーム AI と人間プレイヤとの協力が重要となる．協力を行う上で求められる行動は様々であり，主目的達成のための協調行動，プレイヤの価値観に合わせた協調行動，人間プレイヤらしい行動などが挙げられる．

一方で，初中級者にとって「人間プレイヤのいる環境で上手に主目的を果たすゲーム AI」が最も好ましいとは限らない．仮にチームとして勝てたとしても，自分が何らかの意味で “活躍” できなければ面白く感じられないだろう．本論文は，多少主目的の達成を犠牲にしてでも，人間プレイヤを引き立て活躍させる味方 AI の作成を目指す．そのためには，人間プレイヤの「攻撃したい」「生き残りたい」などの意図を尊重して，それを支援する協調行動を行うべきであると考える．

本研究では，このような意図や強調行動が発生や認識しやすい環境として，3 体のエージェントが 1 体のドラゴンを倒してダンジョンを脱出する DungeonEscape という環境を拡張して用いる．そのうえで，(1) このゲームに発生しうる代表的な意図を考察し，意図に沿った行動をとる意図エージェントを作成する，(2) 特定の意図に対応した意図エージェントを支援するサポート AI を作成する，(3) 意図を意図エージェントの行動から推定する推定器を作成する，(4) 推定器を使って人間の意図に沿ったサポート AI を味方として提示する，というアプローチをとる．

(1) ではまず，攻撃志向，生存志向，効率志向，仲間志向が代表的な意図であると考察した．これらのうち，攻撃志向は敵への攻撃を率先する，生存志向は生き残ることを優先する，効率志向はゲームの早期クリアを目指すものとして，これをサポートすることを本研究の具体的目標とする．(2) のサポート AI の作成や (3) の意図推定のために，この意図を持った AI エージェントを作成したい．そのためには，人間プレイヤのデータを用いたほうが望ましい．しかし，人間プレイヤのデータを収集することはコストの面で困難であったため，意図エージェントは意図に対応した報酬を用いた強化学習を行うことで作成した．

次いで，この 3 つの意図エージェントが確かにそれぞれの意図に対応した行動をとることが評価実験により分かった．攻撃志向エージェントは，ドラゴンへの攻撃率で評価し，生存志向エージェントについても攻撃率で評価した．理由は DungeonEscape という環境では，ドラゴンを倒しに行くと死亡するため，生存志向をある程度反映できるからである．意図を持たないデフォルトエージェントの攻撃率 0.33 と比較して，攻撃志向エージェントでは，攻撃率が 0.41 に増加していた．

生存志向エージェントでは攻撃率は，0.01 にまで減少し，効率志向エージェントではゲームのクリアタイムがおよそ 12 %ほど減少していることが分かった．

(2) では，意図エージェントをペアにして学習をおこなうことで，それに対応する意図に基づいた行動にたいしてサポート行動をとる意図サポート AI を作成した．意図サポート AI は，意図エージェントが望ましいと思っている状況（例えば，自分がドラゴンを攻撃すること）が達成されれば自分にも高い報酬が入ってくるようにした強化学習で訓練する．これにより，意図エージェントの目的達成をサポートするような行動（例えば，ドラゴンから遠ざかり攻撃を譲ること）を学習することを狙う．

評価実験では，意図エージェントとデフォルトエージェントがペアを組んだもの，意図エージェントと意図サポート AI がペアを組んだものを比較することで，意図サポート AI の有効性を示した．攻撃サポート AI とペアになった攻撃志向は，攻撃率が 0.41 から 0.89 まで向上した．また，生存サポート AI とペアになった生存志向は攻撃率が 0.012 から 0.005 まで減少した．効率志向については，効率志向エージェント自身が効率サポート AI であるとも言える．そこで，効率志向エージェント同士でペアを組むことによりゲームクリアタイムを 10 %近く短縮することができた．

(3) では，意図エージェントのゲームログを用いて攻撃，生存志向を判断する 2 値分類を行った．入力データとして「プレイヤの位置座標，ドラゴンの位置座標，ドラゴンとの距離」を入れることで，単純なネットワークモデルでも分類精度は 0.905 となることを示した．

(4) では，意図推定器と意図サポート AI を組み合わせることによって，プレイヤの意図に対応した AI プレイヤをペアに提示する統合意図サポート AI を作成した．統合意図サポート AI を用いた評価実験では，攻撃志向の攻撃率は 0.77 であり，生存志向の攻撃率は 0.01 と概ね正しくサポートできていることが示された．さらに，プレイヤが人間であることを考慮して揺らぎや判断の遅れを入れることで，統合意図サポート AI のロバスト性を検証した．結果として，現状の統合意図サポート AI は人間プレイヤに対して有効的に活躍させることは難しいが，攻撃や生存といったある程度明確な意図があれば，意図に沿ったサポートが可能であることが明らかになった．

目 次

図 目 次

表 目 次

第1章　はじめに

ゲームは，利用者に感動や興奮といった様々な情動を与える娯楽として愛されてきた．このような体験を生み出すために，人工知能（AI）技術は昨今様々な用途で広く利用されており，ゲームデザインと共に大きな発展を遂げている．中でも，ディジタルゲームにおけるゲーム AI は，主に人間プレイヤの対戦相手や人間に代わるキャラクタとしての役割を持つことがある．人間の代替として良いゲーム AI には，人間と並ぶ強さが求められる．そのため，強いゲーム AI は研究の対象となっており，Deep Q-Network(DQN)[1] は，ディジタルゲーム atari 2600 において，対象ゲームの半数以上でプロの人間プレイヤを上回る得点を記録している．このように，ゲーム AI の研究は強さを求める目的において十分な成果を上げている．

今日のディジタルゲームは，通信技術の向上によって遠隔でのゲームプレイが可能になった．そのため，複数の人間プレイヤが一緒に遊べるマルチプレイヤゲームが普及している．マルチプレイヤゲームは，プレイヤ同士の関係によって主に，競争型，協力型，複合型などに分けられる．なかでも協力型マルチプレイヤゲームは，人間が他のプレイヤと協力してタスクを行うものである．人間プレイヤ同士でチームを組めないような場合では，味方プレイヤとしてゲーム AI が必要である．そのため，このようなゲームジャンルではゲーム AI と人間プレイヤとの協力が重要となる．

協力を行う上で求められる行動は様々である．協力して相手チームに勝利することを目的とした複合型マルチプレイヤゲームにおいて，人間の強豪プレイヤを上回る成果を上げた研究 [2] では，ゲーム AI の行動に攻めと守りの役割を持つといった特徴が見られた．この研究では，主目的達成のための協調行動が求められた．また，コマンド選択式の RPG ゲームにおいて，プレイヤの価値観を推定した研究 [3] では，人間プレイヤの行動の記録から効用を学習することで，人間プレイヤの価値観に合わせた協調行動を発生させた．これによって，その人間にとって満足度の高い AI プレイヤを作成した．価値観に合わせた行動は，満足度を高めるために有用であることがわかる．アクションゲームにおいて，人間らしい振る舞いの獲得を目指した研究 [4] では，人間の生物学的制約を導入することで多くの人間にとって「人間プレイヤらしい」とされる行動が見られた．この研究では，人間プレイヤの代替として，強いゲーム AI は不適であり，人間プレイヤらしいゲーム AI はユーザ体験の向上につながることを示唆している．故に，協調行動においても「人間プレイヤらしい」行動は重要である．このようにして，様々な協調行動に着目した人間を楽しませるゲーム AI の研究も多く行われてきた．

その一方で，人間プレイヤを引き立てる行動に注目した研究はあまり知られていない．引き立てる行動とは，人間を活躍させることを目的とした協調行動である．例えば，人間同士のチームでは，互いのレベルが異なる場合，ゲームの結果はどちらかのプレイヤに強く依存し，双方のプレイヤが楽しめない可能性がある．このようなとき，まれに上級者プレイヤが初心者プレイヤをうまく引き立てる行動が見られる．しかし，AI プレイヤは初級者の人間プレイヤの行動を重視せず，チームでタスクを達成することを優先してしまう．故に初級者プレイヤは活躍の場を奪われることになる．また，互いのレベルが同じであっても，何らかの副目的を持っているとき，味方プレイヤにタスクの達成を優先されてしまうことがある．この場合も活躍の場を奪われ楽しめなくなることがあると考える．本研究は，この活躍の場を奪われることによる不満に着目したものである．主目的達成のための協調行動が発生しているにもかかわらず活躍の場をもらえないと感じる要因は 2 点挙げられる．1 点目は，AI プレイヤが主目的を重視しすぎることである．AI プレイヤはゲームに勝利するといった主目的に対して最適な行動をとる．この行動は人間プレイヤが十分に強い場合においては協調行動になるが，初級者プレイヤにとっては，協調行動とはなりえない．2 点目は人間プレイヤにそれぞれ異なる副目的が存在するためである．ゲームに勝利するという主目的に対して，副目的は「自分の攻撃で敵を倒して勝利したい」,「自分が生存したうえで勝利したい」といった主目的達成までの過程の部分に該当する．AI プレイヤは協力する上で，全体の主目的達成のための協調行動を行うが，人間プレイヤの副目的を無視していることが多く，これが活躍の場を奪われると感じる原因となる．

本研究の目的は，協力型マルチプレイヤゲームで，勝率を向上させるための協力ではなく，人間プレイヤを活躍させるための協力をおこなうことである．これは人間プレイヤの意図に沿った支援をおこなう協調行動を発生させることで実現する．ここでの意図とは，ゲームのプレイ中に個々の人間が持っている副目的（好み）を指す．例えば，敵プレイヤを倒すという主目的達成のための行動であっても，今は攻撃を重視したい，回復を重視したい，といった好みが存在している．この意図を推定することで，個々の人間プレイヤに対して適切なサポートを行うゲーム AI をチームとして提示することが可能となる．これにより，既存の協調行動よりも満足させられる行動をおこなうゲーム AI を作成することを目指す．そのうえで，人間プレイヤの意図に沿った支援をする行動が満足度を向上させることを明らかにする．

提案する手法は主に次の 4 つの段階に分割される．まず，対象とする意図を考察し，意図に沿った行動をとる意図エージェントを定義する．次に，この意図エージェントを支援するサポート AI を作成する．また，意図を意図エージェントの行動から推定する推定器を作成する．最後に，この推定器を使って人間の意図に沿ったサポート AI を味方として提示する．

本論文における各章の構成は次のようになっている．2 章では，マルチプレイヤゲームとその関連研究について紹介する．3 章では，対象とするゲームで扱う意図

を考察し，ゲーム環境を決定する．4 章では，提案手法の概要を説明する．5 章では人間の意図を想定した意図エージェントの作成実験について，6 章ではこの意図エージェントをサポートするゲーム AI の作成実験とその結果を考察する．7 章では，意図を汲み取る推定器の作成実験とその結果を述べる．8 章では，統合意図サポート AI の評価手法とその実験結果について考察する．最後に 9 章で今後の展望と総括を行う．

第2章　背景

本研究は協力型マルチプレイヤゲームを対象としている．故に，背景として研究対象としてのマルチプレイヤゲームについて触れる．関連研究としては，マルチエージェント環境での強化学習手法，人間プレイヤとの協力を行うことを目指した研究を紹介する．また，マルチプレイヤゲームの研究プラットフォームの紹介も本章で行う．

2.1　マルチプレイヤゲーム

マルチプレイヤゲームとは，2 人以上のプレイヤ（人間とは限らない）が同時に介入するゲームを指す．シングルプレイではないゲームジャンル全般のことであり，市販されているディジタルゲームはマルチプレイ要素を持っていることが多く，シングルプレイ用のゲームに協力や対戦要素が追加されることも珍しくない．

このようなゲームではプレイヤ間のかかわりは 3 種類に分けられる．

(A) 競合型

(B) 協力型

(C) 複合型

(A) は相手プレイヤとタスク達成を目指して競い合うものである（図 2.1 の左を参照）．両プレイヤの利害が一致しておらず，相手プレイヤと戦う対戦形式が多い．基本的なものには，互いの利得の総和がゼロになる 2 人零和ゲームがある．例えば，囲碁や将棋といったボードゲームがこれにあたる．多くのゲームでは，相

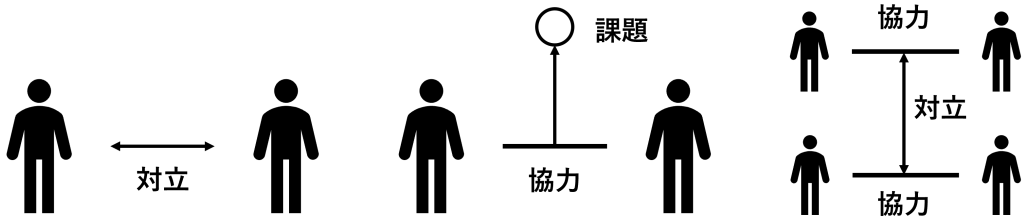

図 2.1: プレイヤ間のかかわり（右から，A. 競合型，B. 協力型，C. 複合型）

手プレイヤとの対戦に勝利することを主目的としており，良い戦いをおこなうことや，戦いを通じて成長することなどを楽しむことができる．

(B) は他のプレイヤと協力してタスクを達成するものである（図 2.1 の中央を参照）．この場合は両プレイヤの利害が一致しており，相手プレイヤと共同作業を行うことが多い．基本的なものには，互いの利得が同じである完全協力ゲームがある．例えば，2 人協力型のシューティングゲームであるツインビー [5] や，協力型アクションゲームであるモンスターハンター [6] がこれにあたる．多くのゲームでは，共同作業でタスクをこなすことを主目的としており，良い成績を出すことや，協調行動でコミュニケーションをとることなどを楽しむことができる．

(C) は (A) と (B) に分類されないものや，(A) と (B) が同時に起こっているもの，ゲーム中に変化するものを指す（図 2.1 の右を参照）．大規模なマルチプレイヤゲームは複合型が多く，協力要素と競合要素を備えている．(A) や (B) に分類されないものでは，目的が定められていないサンドボックスゲーム，(A) と (B) が同時に起こっているものでは，野球や MMORPG(Massively Multiplayer Online Role-Playing Game) がある．このようにジャンルが広いため，プレイヤの楽しみ方も (A)，(B) に対して幅広い．

マルチプレイヤゲームにおける協調行動は (B)，または部分的に (C) で発生する．以後は，主に (B) で発生する協調行動の種類に焦点を当てる．

2.2 関連研究

関連研究では，まず，本研究で用いたマルチエージェント強化学習手法を紹介する．次に，味方プレイヤと協力するゲーム AI についての研究について述べる．最後に，研究プラットフォームを紹介する．

2.2.1 マルチエージェント強化学習

マルチエージェントシステムは，複数の主体が自ら意思決定を行うことで構成されるシステムである．この環境では，各エージェントの行動が複数のエージェントに影響を及ぼす．ゆえに，協調行動の取得には複雑な作業を必要とし，そのような振る舞い獲得に至るアプローチはこれまで研究されてきた [7]．また，マルチエージェントシステムは現実世界への適応が期待されており，マルチエージェントシミュレーション [8] や，ロボット制御 [9] などの分野で応用されている．

マルチエージェント強化学習は複数のエージェントが環境に存在しており，各エージェントが意思決定を行うという枠組みの強化学習である．

マルチエージェント強化学習の学習方式は，シングルエージェントのものと比べて複雑である．これは，複数のエージェントに対する学習主体の捉え方と，報酬の分配方法が一様ではないためである．

最も単純な学習方式は，複数のエージェントが単体で意思決定を行い，それに伴った報酬を各エージェントに与えるというものである．しかし，この方式は全体のエージェントの行動空間を認識できず，協調行動が発生しにくい．マルチエージェントの連続制御問題に取り組んだ研究では，この学習方式は，他のものと比較して性能が低いことが示された [10]．

そこで，複数エージェントを集中させた中央意思決定機関を用いた学習方式が提案された．これは，1 つの意思決定機関によって全エージェントを学習し，報酬は全体を統括する意思決定が受け取るというものである．この方式は，エージェント全体の行動空間がエージェントの数の増加によって膨大になることである．

近年では，この両方式を取り入れた学習方法が良い成果を上げている．これは，各エージェントが定められた制約の中で行動空間を共有し学習するものである．学習時に，各エージェントの意思決定機関に与える情報を統一することで，中央決定機関のような役割を果たし，実行時にこの情報を制限することで，単体での意思決定を可能とするものである．この方式では，グループ全体で得られた報酬を各エージェントに分配することで，各エージェントのポリシーを更新している．個別エージェントの学習に，部分的に全エージェントの観測情報を加えた研究では，いくつかのベンチマーク問題で上記学習方式よりも高い性能を得ている [11]．

現在は，このような複数の学習方式が存在し，取り扱うタスクの特性や状況に応じて使い分けられる．

深層強化学習技術を使ったマルチエージェント強化学習手法の台頭により，より複雑な環境に対して適応されている．中でも，ゲームをテストベッドとしたマルチエージェント強化学習は多くの成果を残している [12].

マルチエージェント強化学習の課題の 1 つとして，信用割当問題が挙げられる [13]．信用割当問題は，完全協力型ゲームにおいて，受け取った報酬信号を正しく共有することの困難性を指したものである．マルチエージェント環境では，各エージェントはエージェント全体の行動空間において報酬が最大となる行動を行うことが望まれる．しかし，全エージェントの行動空間が自明ではない場合，各エージェントの行動が全体の行動に対してどの程度寄与したかを判断することが難しい．Foerster らはこの問題に対して，エージェント全体の行動価値関数と各エージェントが行動しなかった場合の価値関数との差を寄与度として定義することで，それぞれの貢献度を算出した．この貢献度を用いて報酬を分配することで学習したモデル counterfactual multi-agent(COMA) は，協力型リアルタイムストラテジーゲームにおいて既存のマルチエージェント強化学習手法を上回るスコアを出している [14].

2.2.2 味方プレイヤと協力するゲーム AI

味方プレイヤと協力するゲーム AI の研究について紹介する．協調行動を獲得した研究は主に 2 種類の目的に区別される．

- 主目的達成のためのもの
- 副目的達成のためのもの

主目的達成のための協調行動は，ゲームにおける利得を最大化するための行動である．副目的達成のための協調行動は，ゲームの利得とは異なる価値観を重視した行動である．たとえ主目的達成のための協調行動であっても，発生が困難なものは多い．これは，味方が上級者であるか初級者であるか，ゲーム AI であるかという前提によって最適な協調行動が異なるためである．

主目的達成のために協力するゲーム AI の研究として，DeepMind 社による ForTheWin(FTW) がある [15]．FTW は，内部学習モデルと外部学習モデルを合わせたものであり．複数のエージェントがそれぞれ単一の強化学習を行う．内部学習では，外部報酬から得た全体報酬を用いてエージェント同士を対戦させる強化学習により方策を更新する．外部報酬は，ゲーム勝利時の報酬を受け取り内部学習モデルを更新する．これにより，Quake Arena の Capture The Flag において人間同士のチームを凌駕する結果を残した．Capture The Flag は複合型ゲームであり，敵を倒すことと，敵チームの flag を奪うという複数のタスクを持つ複雑なゲームである．ForTheWin はこのようなゲームにおいて，人間プレイヤの後をついていく，攻めと守りを切り替えるといった人間プレイヤとの協調行動も可能としている．

さらに，DeepMind 社が発表した Fictitious Co-Play(FCP) では，様々な人間プレイヤに好かれる協調行動が実現されている [16]．この論文ではテストベッドとして overcooked を使用している．このゲームは協力型ゲームであり，味方との共同作業の中でより効率的に料理を作ることが求められる．主目的達成のための行動は味方のレベルに応じて変化するため，多様な人間プレイヤとの協調行動は困難である．そこで，作成されたエージェントは人間プレイヤのデータを用いず，エージェント自身の学習途中のモデルを味方として使う．ただし，McIlroy-Young らは，AlphaZero を基にした強化学習手法で，学習途中のモデルは人間と同じ強さであっても人間の行動と一致していないと指摘している [17]．このようなモデルと，別のパラメータを持ったエージェントとを組み合わせた学習により，多くのレベルの人間プレイヤに適応した．また，人間プレイヤらしい振る舞いではない味方との学習にもかかわらず，既存手法よりも人間に好まれることを示している．

副目的達成のために協力するゲーム AI の研究として，和田らによるプレイヤの価値観を学習したものがある [3]．この研究では，人間プレイヤの重視している目的を副目的として効用関数で表現することに着目している．プレイヤごとにとりうる各行動がもたらす結果をランダムシミュレーションの平均帰結として求めることで，効用関数を学習し，その価値観に合わせた行動をとるゲーム AI を作成した．これにより，コマンド選択式の RPG ゲームにおいて，特定の効用重みをもった関数よりも自然な挙動が得られることを示した．

2.2.3 マルチプレイヤゲームの研究環境

味方と協力するゲーム AI の研究は，多くのマルチプレイヤゲームを用いて行われてきた．これに伴って，研究用のプラットフォームの開発や API[1]を提案した研究も多く存在する．本節では，マルチプレイヤゲームの研究プラットフォームを紹介する．

AI とロボット工学の研究を促進することを目的とした国際的な競技会に Robot World Cup Initiative(RoboCup)[18] がある．競技種目の 1 つである RoboCup サッカーには，実機ロボットを用いないシミュレーションリーグが存在する．これは，マルチエージェントシステムのテストベッドとなっていたサッカーシミュレーターを用いたものである [19]．このリーグは，サッカーをディジタルゲームとして 11 対 11 で行い得点を競うものであり，マルチエージェント環境での協調行動や戦略獲得が研究対象となることが多い．1997 年から始まった大会は本年に至るまで毎年開催されており，RoboCup の研究のためのソフトウェアプラットフォームを提供している．

完全協力型のディジタルゲームプラットフォームとして Rocha らは Geometry Friends を提案した [20]．Geometry Friends では，プレイヤはダイヤモンド（紫）

[1]ここでは，エージェントがプラットフォームで学習できるようにするインタフェースのこと

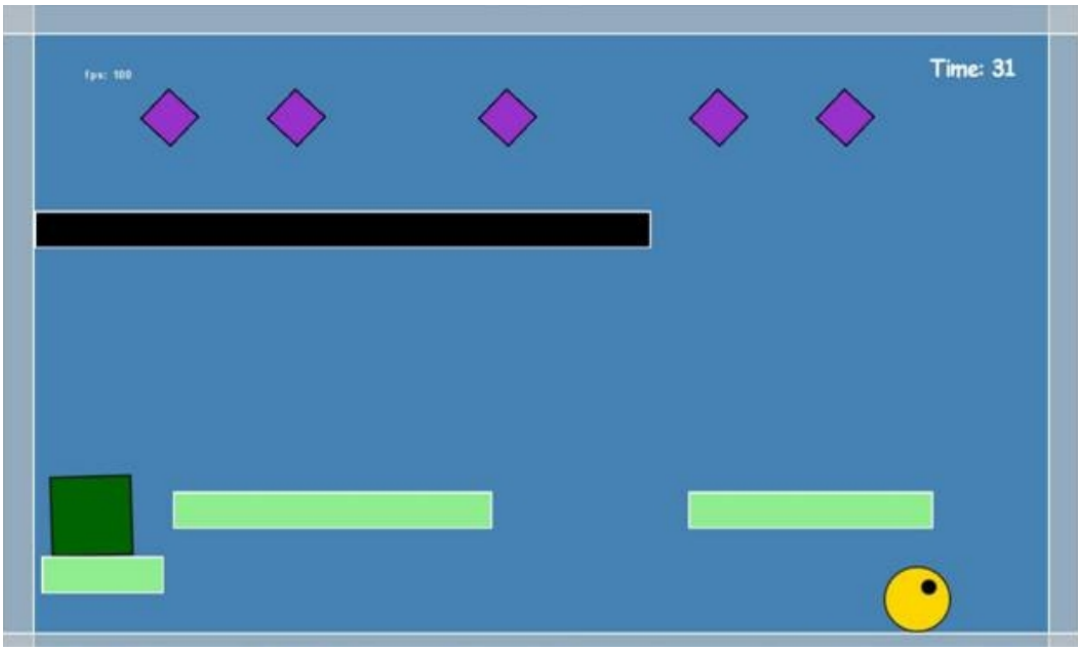

図 2.2: Geometry Friends のプレイ画面，[20] より引用

を収集することを目的としており，円（黄）のプレイヤと四角形（緑）のプレイヤが協力する必要がある（図 2.2）．円のプレイヤはジャンプと横移動，四角形のプレイヤは同面積内であれば縦横比を変化させることができる．この環境では，2 人のプレイヤの役割が異なり，もし 2 つの主体がそれぞれ意思決定する場合には，味方プレイヤの行動を汲み取った協調行動が重要となる．Geometry Friends はオープンライブラリとして公開されており，AI プレイヤ同士の協力に焦点を当てた競技会も行われている．

Mojang 社によって開発された複合型マルチプレイヤゲームに Minecraft がある [21]．主目的が定められていないオープンワールドのゲームであり，プレイヤは世界を自由に探索できる．故に，複雑な環境における協力を目的とした研究のテストベッドとして使われており，Project Malmo として AI 実験プラットフォームが公開されている [22]．Malmo は，Minecraft の拡張性の高さから，協力型タスクのほかにもナビゲーション行動や問題解決タスクなどの様々な研究分野に適応できる．

2010 年に Blizzard 社が開発したストラテジーゲームとして，StarCraft2[23] がある．StarCraft2 は，プレイヤが盤面の複数のユニットを操作して相手と戦う StarCraft の続編である．オープンライブラリとして公開されており，DeepMind 社は StarCraft2 を基にした StarCraft II Learning Environment(SC2LE) を提供している [24]．SC2LE には，StarCraft2 API や強化学習のために最適化された StarCraft2 環境が含まれている．StarCraft2 は，それぞれのユニットを AI が管理するマルチエージェント環境であり，強いゲーム AI の作成には AI 同士の協調行動が必要になる．マルチエージェント強化学習の研究でも多く用いられるプラットフォーム

であり，2019 年には，AlphaStar がマルチエージェント強化学習と模倣学習を組み合わせて学習することで，世界のグランドマスター 2 人に勝利した [25].

PettingZoo[26] は，マルチエージェント強化学習の研究活性化を目的として公開されたマルチエージェント環境のオープンライブラリである．これは OpenAI の強化学習ライブラリ OpenAI Gym[27] を基にして開発された．Gym の機能を多く持ち，独自の Agent Environment Cycle(AEC) モデルを提供している．このゲームモデルは，環境エージェントを導入し，環境エージェントが行動することで環境が変更されるというものである．これにより，多様なゲームルールでマルチエージェント強化学習を行えるとしている．

第3章　対象とするゲーム

本章では，研究の対象としたゲームについて述べる．本研究は，協力型マルチプレイヤゲームで意図に基づいた行動が発生することが必要である．また，それらは行動を見てわかりやすいような意図であることが望ましい．これらの要請から，本研究では単純化された協力型マルチプレイヤゲームとして Unity ML-Agents を選んだ．協取り扱う意図がどのように発生するのかを考察することで環境に適応させる．

3.1　Unity ML-Agents

Unity は，Unity Technologies 社が開発したゲームエンジンである．シミュレーション環境での開発や物理演算に優れたプラットフォームとして公開されている．また，同社は Unity を用いて機械学習を行うことができる Unity ML-Agents をオープンライブラリとして提供している [28]．これは，Unity 上で動作し，python API を用いて学習を行うことが可能であり，Unity を通して学習過程やシミュレーション結果を視覚的に確認できる．この環境では，強化学習や模倣学習をはじめとした様々なアルゴリズムが実装されており，多数のゲームがテストベッドとして提供されている．

本研究では，Unity ML-Agents で提供されているサンプル環境である DungeonEscape（図 3.1）を用いる．このゲームは単純な動きをする敵キャラクタを環境の一部とみなせば，完全協力型マルチプレイヤゲームであり，3 人のプレイヤと 1 匹のドラゴンが存在する．プレイヤ（青）がドア（黄）から脱出することでゲームクリアとなり，先にドラゴンがポータル（紫）を通るとゲームオーバーとなる．故に，プレイヤはドラゴンがポータルにたどり着くよりも先に脱出する必要がある．しかし，ドアは初期状態では鍵がかかっており，ドラゴンを倒さなければ鍵を得ることはできない．プレイヤはドラゴンを倒す必要があるが，プレイヤとドラゴンが接触すると両者ともに死亡する．そのため，鍵を手に入れるには，少なくともプレイヤのうち 1 人が犠牲になる必要がある．

DungeonEscape は非常に単純なゲームルールであるため，人間プレイヤの意図に基づいた行動が発生しにくい恐れがある．そこで，本研究ではこのゲームに変更を加えたオリジナルルールを提案する．

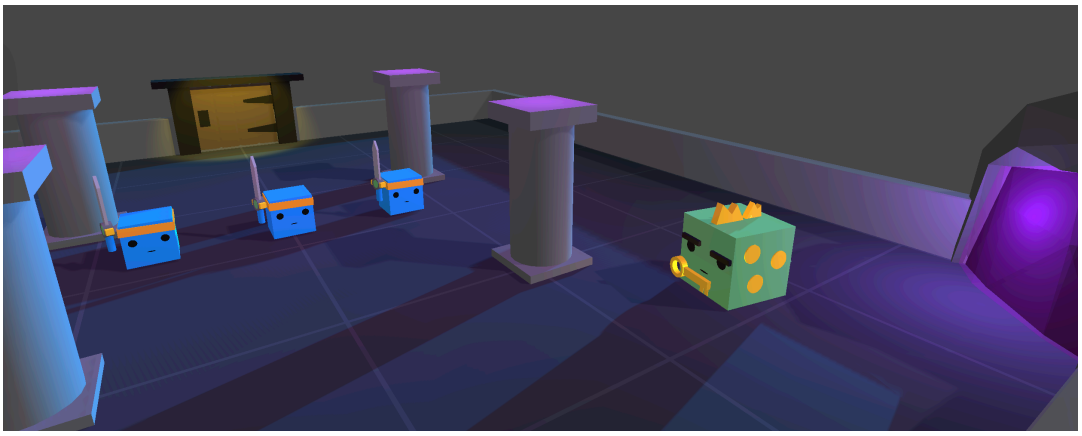

図 3.1: DungeonEscape のゲーム画面（出典：GitHub Unity ML-Agents Toolkit[29]）

3.2　対象とする意図の考察

本研究の目的は，人間プレイヤの行動から意図を読みとり，その意図をサポートしてプレイヤを “意図どおり活躍させる” ことである．しかし，ゲームを特定せずにまた細かい副目的まで考えると，人間プレイヤが持ちうる意図は無数に存在し，それらすべてに対応することはできない．例えば,「映える魅せプレイがしたい」「わざと悲劇的な負け方をしたい」などは本研究の対象外である．そこで，対象ゲームを DungeonEscape に限定し，そこで発生しうる 4 つの代表的な意図を以下に挙げ考察や実装の対象とする．

(A) 攻撃志向

(B) 生存志向

(C) 効率志向

(D) 仲間志向

(A) は，“自分がドラゴンを倒したい” という意図を持っている．人間プレイヤがこの意図を持つとき，ドラゴンを倒したことを活躍したと定義する．翻って，人間プレイヤの死亡は考慮しない．3 人のプレイヤのうち 1 人がドラゴンを倒すという役割を持つため，人間プレイヤが (A) の意図をもって行動することは十分ありえる．このような意図を持つ人間プレイヤの典型的行動は，味方プレイヤの初期位置に関わらず，自身がドラゴンに向かって行き，接触するなどである．

(B) は，“自分は死亡したくない” という意図を持っている．人間プレイヤがこの意図を持つとき，ゲームクリア時にそのプレイヤが生存していることを活躍したと定義する．ゲームをクリアするためには，3 人のプレイヤのうち少なくとも 1 人は死亡するため，人間プレイヤが (B) の意図をもって行動することも十分ありえる．このような意図を持つ人間プレイヤの典型的行動は，味方プレイヤの初期位置に関わらず，自身がドラゴンから離れる，もしくはドラゴンに向かって行くが，接触はしないものである．

(C) は，“出来るだけ効率よくゲームをクリアしたい” という意図を持っている．人間プレイヤがこの意図を持つとき，ゲームのクリアタイムが早くなることを活躍したと定義する．ゲームそのものにクリアタイム報酬は存在しないが，ゲームを早くクリアすることを副目的としたタイムアタックを楽しむプレイヤは存在することから，人間プレイヤが (C) の意図をもって行動することも十分ありえるだろう．このような意図を持つ人間プレイヤの典型的行動は，味方と初期配置ごとの役割分担を行うことである．

(D) は “味方プレイヤの行動に合わせるように協力したい” という意図を持っている．この意図は，自分が味方に協力している状況を楽しむものであり，他のプレイヤの行動を見てからそれに合わせるように協調ができた場合に活躍を感じられ

る．協力型マルチプレイヤゲームを想定すると，人間プレイヤが (D) の意図をもって行動することは十分ありえる．(C) と (D) の表出する行動は似ているが，(D) の場合は仲間のとった行動が多少全体のためにはおかしい行動であったとしてもそれをサポートする点で異なる．

以上のことから，A から D のような意図に基づく行動は，このようなゲームで頻繁に発生しうるものであると考える．しかし，DungeonEscape のルールでは，(C) の目的のためには実際には役割分担を必要とせず，ドラゴンに一直線に向かう行動が最善となってしまう.[1]故に，(A) と (C) は同じ行動を表出させてしまうことになる．異なる意図が異なる挙動を示すことが本研究の対象としては望ましいので，この点は，次節で述べる環境を変更する動機の一つである．また，D は人間プレイヤから見て「仲間を活躍させたい」ということであり，つまり本研究で AI エージェントにやらせたい挙動そのものである．これはより高度な意図ないし挙動であって,「仲間を活躍させたいプレイヤを活躍させる」ために AI エージェントが何をすべきかは難しい問題である．そこで，本研究では (D) は取り扱わず (A) から (C) のみを対象とする．

[1] ドラゴンを倒すとその場に鍵がドロップするため，ドラゴンへの攻撃と，鍵を拾うという役割の行動が一致する．また，エージェント全員がドラゴンへ向かうことに何らリスクがない．

3.3 環境の変更

前節で述べたように，さまざまな意図や意図に基づく行動を区別できる形で表出させるためには，オリジナルの DungeonEscape は都合が悪く，一部を拡張したものを用いることにする．そこで本節では，この拡張版 DungeonEscape について説明する．DungeonEscape からの変更部分を表 3.1 に示す．

DungeonEscape では，観測は一人称視点であった．各エージェントの向いている方向のみを観測する部分観測であるため，エージェントは盤面全体の真の状態を得ることができない．この環境で味方エージェントの意図を汲み取る場合，味方エージェントを観測し続ける必要がある．ゲームをクリアすることを考えると，ドラゴンも観測していなければならないため，各エージェントやテストプレイ時の人間プレイヤは，絶えず周囲を観測する首振り行動を取らなければ協調行動が行いにくいことになる．これは煩雑であり，協調行動を主眼においた本研究にとっては都合が悪いため，観測範囲をエージェントの前方から全方位への変更した．これにより，エージェントがその場で方向を変更した際に観測できるすべての部分を観測可能となる．部分観測の情報をどう集めてどう統合するかという課題はそれはそれで重要であるが，本研究の中心的課題とは異なるためこの課題が生じないように環境を容易化することにした．

また，(C) 効率志向を考える．前節から，これは (A) 攻撃志向と区別することが難しいと述べた．(A) であっても (C) であってもドラゴンに向かって行く行動が最適な行動になるからである．これはゲームの単純性からくる問題であると考え，鍵のドロップ位置とドラゴンの死亡条件を変更した．DungeonEscape では鍵はドラゴンが持っている．これを，ドラゴンを倒した後マップ中央から鍵が入手できるようにすることで,「素早いクリアのために鍵の発生する場所で待ち伏せする」などエージェントの行動が多様化すると考えた．また，ドラゴンの死亡時の挙動として爆発を追加した．爆発とは，ドラゴンを中心とした一定範囲内にいるエージェントに攻撃するものである．エージェントはこの攻撃に触れると死亡するため，一斉に向かって行く行動では死亡するリスクがあり味方エージェントの行動を見た上で自身の行動を決定するといった協調行動が発生すると考えた．これに

表 3.1: DungeonEscape からの変更点

要素	DungeonEscape	拡張版 DungeonEscape
観測方法	部分的	全体[2]
鍵のドロップ位置	ドラゴンの位置	マップの中央
ドラゴン死亡時の挙動	接触エージェントの死亡	周辺エージェントの死亡
制限時間	なし[3]	あり

[1]エージェントを中心とした 360 度
[2]制限時間は 1 ゲームで 25000 フレームのため，実質ない

より，(A) と (C) の表出する行動が区別できるようになることが期待される．

最後の変更点は，制限時間である．DungeonEscape では，学習を円滑に行うために制限時間は存在する．しかし，この学習時間は平均ゲームクリア時間のおよそ 100 倍に相当し，人間がゲームをプレイする上で実質ないものとして扱うことができる．従来の設定のままでは，ドラゴンを倒した後脱出までのモチベーションが存在しないことから，ゲーム全体に制限時間を追加した．

以上の拡張版 DungeonEscape を本研究環境として扱う．また，詳しいゲームプレイの手順を図 3.2 に示す．

0. 3 体のエージェントと 1 体のドラゴンはマップ上のランダムな位置に配置される．この時，エージェントは自身を中心とした全方位が観測できる．

1. エージェントはドラゴンへの攻撃を目指す．また，ドラゴンはポータルからの逃亡を試みる．

2. エージェントとドラゴンの接触時，ドラゴンは爆発を起こす．これは，一定範囲内（図 3.2 の赤円内部，マップ直径の 1/5 程度）のエージェントを死亡させるものである．

3. ドラゴンが倒れると，マップ中央に鍵が出現する．

4. 残ったエージェントが鍵を拾った状態でドアまで行くことで，ゲームクリアとなる．

5. ドラゴンの逃亡，エージェントの 1 人の脱出，エージェント全員の死亡，制限時間の超過によって 0 に戻り，環境はリセットされる．

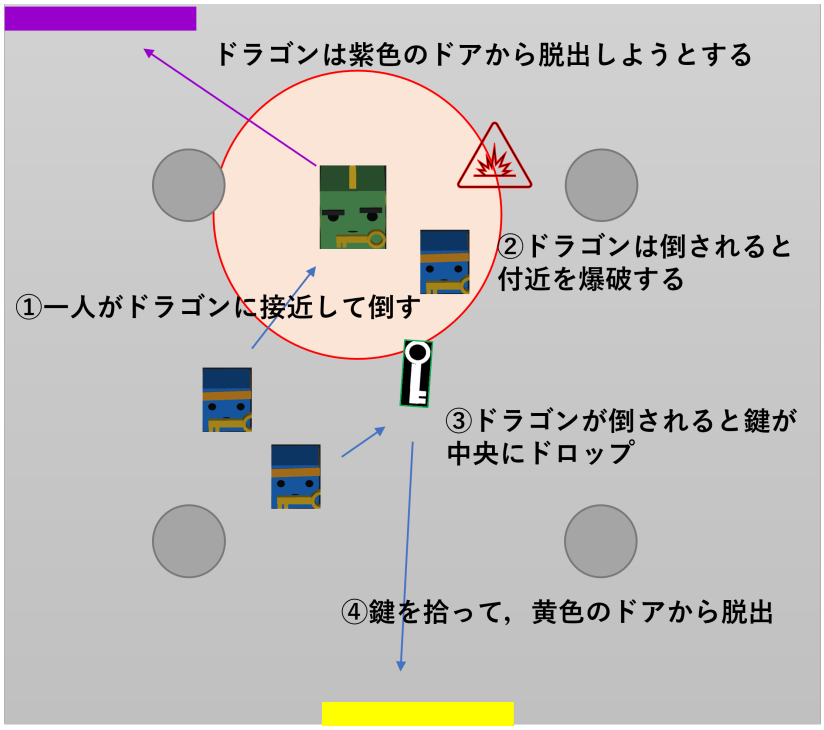

図 3.2: 拡張版 DungeonEscape プレイの流れ

第4章　提案手法

本章では，提案する手法について述べる．本研究の目的は，協力型マルチプレイヤゲームで，人間プレイヤを活躍させるための協力をおこなうことである．アプローチの全体像を図 4.1 に示す．

(1) 意図の分類はすでに3.2節で行ったことである．すなわち，拡張版DungeonEscapeを用いて，(A) 攻撃したい，(B) 生き残りたい，(C) 早くクリアしたい，の 3 つの意志を対象として，その読み取りとサポートを行うことでプレイヤを活躍させることを目的とする．

(2) 我々は，人間プレイヤの行動からその意図を推定したい．そのために，（行動，意図）のペアを用いて教師あり学習を行うことにする．しかし，人間プレイヤの行動データを収集することは，多くのコストと時間が必要となる．また，人間プレイヤによる意図の偏りを防ぐ必要があるため，各意図を持った人間プレイヤをそれぞれ十分な数募集しなければならない．そこで，本手法では，人間プレイヤの代替として AI プレイヤを使用する．“意図エージェントの作成” では，この人間に代わる AI プレイヤを意図エージェントとして定義した行動をとるように学習する．例えば,「攻撃したい」という意図エージェントの作成は，ドラゴンへの攻撃に報酬を与えることで可能になると考える．実装の詳細と評価結果は 5 章で述べる．

(3) “個別意図サポート AI 作成” では，意図エージェントをペアにしてゲームをプレイすることで，意図を汲み取って活躍させるゲーム AI を作成する．例えば,「攻撃したい」という意図エージェントを活躍させることは，意図サポート AI が攻撃を譲ることであると考える．このように，それに対応した意図に対してサポートする行動をとるゲーム AI を意図サポート AI と呼ぶ．個別意図サポート AI の実装の詳細と評価結果は 6 章で述べる．

(4) 意図推定を行うために，意図エージェントを使った “行動データ収集” を行う．人間プレイヤの行動から意図を推定する必要があるため，収集するデータは，ゲームプレイ中のエージェントとドラゴンの行動ログである．データの詳細については，意図推定器の提案とともに述べる．

(5) “意図推定モデル作成” では，行動データから意図を推定する．学習には，行動ログに対応した意図をラベリングした教師データを用いる．7 章では，行

動データ収集に加えて，この意図推定器の作成と性能評価を行う．

(6) “統合意図サポート AI” は，研究の目的となるプレイヤの意図を汲み取り活躍させるゲーム AI である．これはまず，ゲーム開始一定時間後に，人間プレイヤの行動から意図推定モデルを用いて意図を推定する．そして，その意図にあった個別意図サポート AI に行動決定を任せることでプレイヤを活躍させることを狙う．そこで，統合意図サポート AI は，意図推定器によって推定されたプレイヤの意図に応じた個別意図サポート AI をペアとして提案する．ゲーム内でペアを切り替えることで，様々な意図を持った人間プレイヤに対応できると考える．実装の詳細と評価結果は 8 章で述べる．

最後に，本提案手法の有効性を示すために統合意図サポート AI を使った性能評価を行う．意図に沿った支援ができているのか，また，既存のマルチエージェント強化学習手法に対して満足度がどの程度向上するのかを明らかにする．

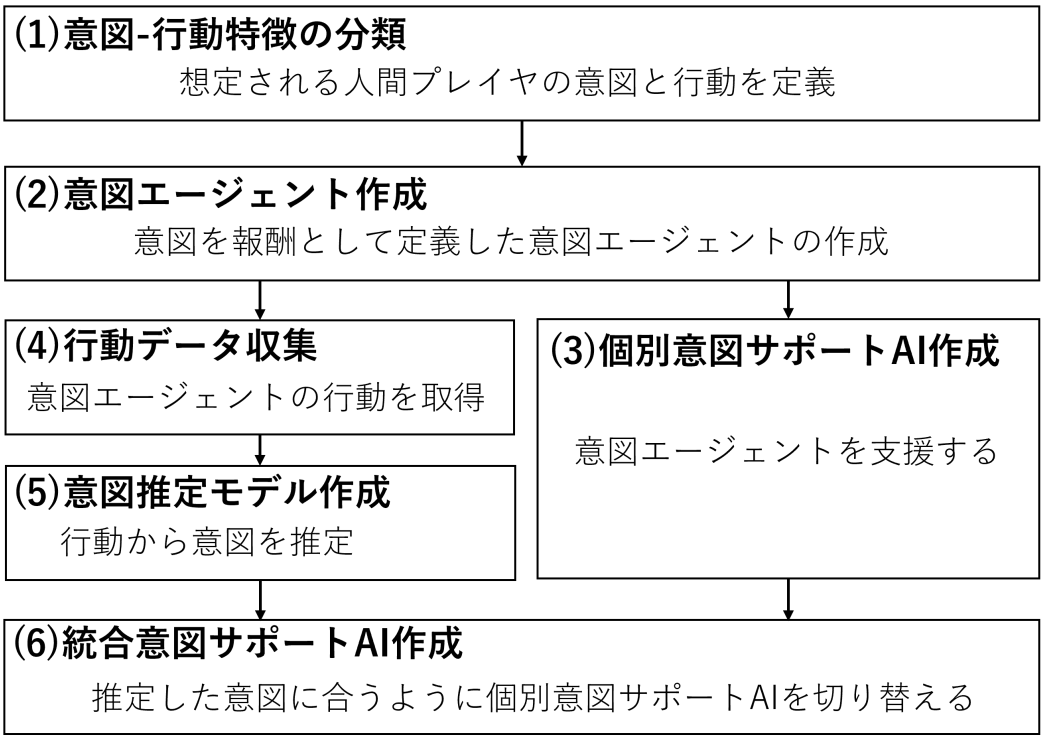

図 4.1: アプローチの全体像

第5章　意図エージェントの作成

本章では，特定の意図を持ったゲーム AI について説明する．まず，3 章で述べた対象とする意図について，意図ごとの設計方針をまとめる．5.2 節では，意図エージェントの学習手法を述べ，ねらいの行動が現れたかを評価する．

5.1　意図ごとの設計方針

3.2 節では対象とする意図について列挙した．そもそも，意図エージェントは人間の意図を推測する学習のためのデータ集めに使うものである．故に，人間プレイヤと同じような行動をすることが望まれる．例えば，攻撃志向のプレイヤはドラゴンを攻撃しに行く行動をとることが予想されるため，攻撃志向エージェントにもそのような振る舞いの獲得が求められる．意図エージェントの行動をヒューリスティックに定めることもできるが，本手法では，意図ごとに報酬を与えた強化学習で意図に基づいた行動が獲得できるのかを明らかにする．

この節では,「人間がとりうる行動（つまり人間の模倣をさせたい行動）」と「それにどんな報酬を与えたらよさそうか」について述べる．表 5.1 は，各意図と行動を対応させたものである．

- 攻撃志向
  攻撃志向は，ドラゴンを倒したいとする意図を持っている．この意図を持った人間プレイヤは，ドラゴンへの接近を積極的に行うだろう．加えて，味方プレイヤの初期位置やドラゴンの爆発範囲などは考慮しないだろう．このような意図を模倣するエージェントを作るためには，ドラゴンを倒すことに正の報酬を与え，ドラゴンを倒せなかった時は負の報酬を与えるのが良いと考える．
- 生存志向
  生存志向は，死亡したくないとする意図を持っている．この意図を持った人間プレイヤは，ドラゴンから離れるように行動するだろう．また例外的に，味方とドラゴンの初期位置から，自分が攻撃しなければゲームオーバーになる（ドラゴンがポータルから逃げてしまう）場合のみドラゴンへ接近するだろう．このような意図を模倣するエージェントを作るためには，死亡することに負の報酬を与えるのが良いと考える．

- 効率志向
  効率志向は，効率的にゲームをクリアしたいとする意図を持っている．この意図を持った人間プレイヤは，味方の初期位置によって変化するだろう．この人間プレイヤと味方エージェントのうちドラゴンとの距離が近い場合はドラゴンへの接近を行うが，遠い場合はマップ中央へ接近するだろう．なぜなら，効率志向の人間プレイヤは，ドラゴンへの攻撃を味方に任せてマップ中央に出現する鍵を早く手に入れようとするからである．このような意図を模倣するエージェントを作るためには，ゲームクリアにかかった時間による報酬を強めるのが良いと考える．

表 5.1: 意図に基づいた行動の例

意図	ねらい	典型的な行動
攻撃志向	ドラゴンを倒したい	ドラゴンへの接近
生存志向	生き残りたい	ドラゴンから離れる
効率志向	早くゲームをクリアしたい	味方によって変化

5.2　特別な意図のないエージェントの学習

本節では，3.3 節で導入した拡張版 DungeonEscape 環境で，5.1 節で述べたような特別な意図や特徴を持たない「デフォルトエージェント」を学習する実験を行う．ここで学習したモデルは，5.3 節で述べる意図エージェントとの比較のために用いる．

5.2.1　予備実験の概要

実験目的は，拡張版 DungeonEscape で強化学習を用いて，協力してゲームをクリアするという振る舞いが発生することを確認することである．

以下に実験設定を示す．

- 環境は，DougeonEscape を基にした拡張版 DungeonEscape である．
- 環境には 3 体のプレイヤと 1 匹のドラゴンが存在する．
- ドラゴンよりも先に環境から脱出することを目的としたゲームである．
- プレイヤの脱出口としてドア，ドラゴンの脱出口としてポータルが存在する．
- 制限時間は 6000 フレームとする．ただし，1 秒間はおよそ 60 フレームである．
- ゲームの終了条件は，プレイヤがドアから脱出，ドラゴンがポータルに侵入，制限時間の超過の 3 つである．
- ゲーム開始時に，ドラゴンとプレイヤはランダムな初期位置で初期化される．
- プレイヤはドラゴンを倒して鍵を入手したのち，ドアから脱出することを目指す．
- プレイヤのとれる行動は，前後左右への移動，左右の方向転換である．
- プレイヤは，自身の位置から全方位を不透過センサによって観測できる．
- 不透明センタは，センサを飛ばした時の｛壁，プレイヤ，ドラゴン，鍵，ドア，ポータル｝との衝突判定と距離を観測する．
- ドラゴンは，直線的にポータルへ向かう行動がヒューリスティックに定義されている．

- マルチエージェント強化学習で 3 体のプレイヤを学習する．
- 学習アルゴリズムは MA-POCA[30] を用いる．
- エージェントは，ゲームクリア時に全体報酬と個別報酬を得る．

ゲーム環境については，その詳細に触れている 3.3 節を参照されたい．学習アルゴリズム MA-POCA[30] は，2.2.1 節で取り上げたマルチエージェント強化学習手法 COMA[14] を基にしたものである．MA-POCA は，COMA で扱った信用割当問題をゲーム中に非活性化するエージェントにも拡張したモデルである．報酬はエージェント全体に対する報酬 R_g と各エージェントに対する個別報酬 R_i として与えられる．R_g と R_i は，ゲームクリアまでにかかった時間 f とプレイヤの行動変更回数 c によって次のように表される．

$$R_g = (0.999)^f \tag{5.1}$$
$$R_i = -0.001c$$

R_g は，より早いゲームクリアを望む全体報酬である．また，R_i は，行動変更数にペナルティを与えることで，人間らしくない不自然で細かい機械的な挙動を抑制する目的がある．また，その他の設定は表 5.2 に示す．これは，MA-POCA 内で用いられているものである．

表 5.2: MA-POCA のハイパーパラメータ

Hyperparameter	
Minibatch Size	1024
Buffer Size	10240
Learning Rate	0.0003
Optimizer	Adam
Discount Factor	0.99
Hidden Units	256
Fully Connected Layers	2

5.2.2 実験結果

まず，図 5.1 に 1000 万ステップ学習を行ったときの平均報酬を示す．学習には 5 時間程度を要した．学習は 2 セット行い，図中の曲線（赤，青）は移動平均をとったものを表している．

500 万ステップで学習時全体報酬はおよそ 0.8 になることが確認できた．ただし，全体報酬は前節で定義されたものであり，ゲームクリア率とは異なる．そこで 500

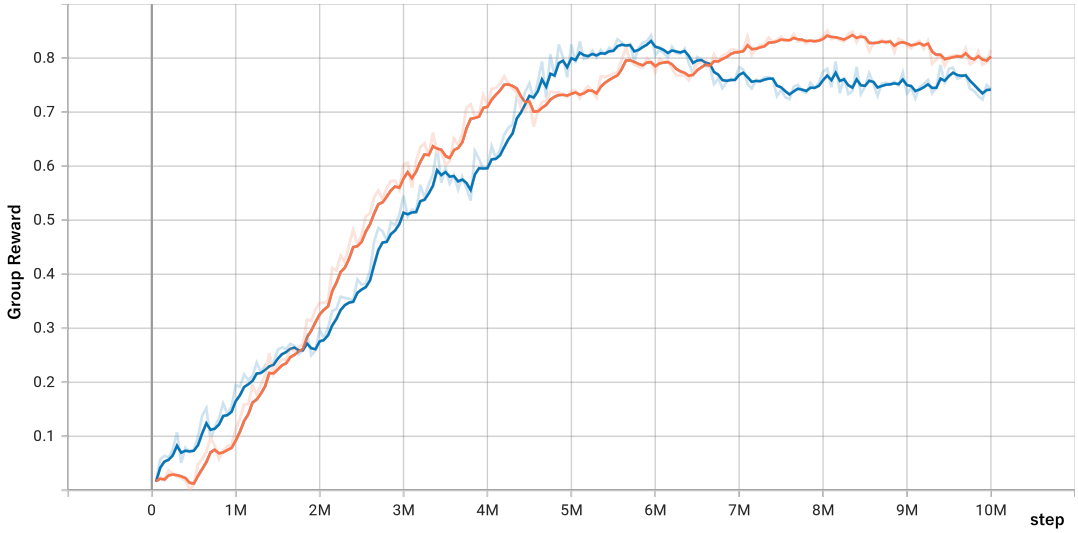

図 5.1: デフォルトエージェントの学習曲線

万ステップと 1000 万ステップにおいて学習モデルを出力し，ゲームクリア率を出した．

ゲームクリア率とエピソード長を表 5.3 に示す．なお結果は，それぞれモデルを 2000 試行シミュレーションした平均である．

平均クリア率は 1 ゲームをクリアできた割合を表しており，両モデルともに 0.9 を超えることが確認できた．なお，クリアできないエピソードでは，ドラゴンの初期位置がポータルに近く，エージェントのドラゴンへの攻撃が間に合わないものがあった．これは，初期位置がランダムで決められる設定上仕方がないものである．また，エージェントが柱に引っかかることで間に合わなくなるものもあり，こちらは獲得される振る舞いがより向上できる可能性を示唆している．

平均エピソード長は 1 ゲーム終了までにかかった時間を表しており，両モデルともにおよそ 140 フレームであった．ここで，時間の単位はフレームを使用している．1 秒が 60 フレームであることを考えると，およそ 2.5 秒でゲームを終えていることが分かる．

この結果から，500 万ステップで十分に学習ができていると考える．加えて，1000 万ステップの学習は学習にかかる時間の観点から難しい．そこで，500 万ステップ学習したモデルをデフォルトエージェントとして使用し，これ以降エージェントの学習ステップ数は 500 万を基準とする．

表 5.3: デフォルトエージェントの性能評価

モデル	平均クリア率	平均エピソード長 (frame)
5M ステップ	0.916	141.1
10M ステップ	0.923	139.8

5.3 意図エージェントの作成と評価実験

意図エージェントは，意図を持った人間プレイヤの代替となるものである．そのため，5.1 節では，意図に基づいた行動とそれを発生させるための報酬について述べた．ここでは，DungeonEscape を基に提案した拡張版 DungeonEscape 環境で，プレイヤにこの報酬を与えることで，意図エージェントを作成する．

5.3.1 意図エージェントの学習方法

ここでは，3 種類の意図エージェントを学習する．意図エージェントの行動は，意図を持った人間の振る舞いと似ていることことが望ましい．各意図は，与える報酬の変更によってのみ定義する．このようにして，攻撃志向（式 5.2），生存志向（式 5.3），効率志向（式 5.4）の行動に合わせた報酬をそれぞれ以下のように定めた．

$$
\begin{aligned}
R_g(attack) &= (0.999)^f && (5.2)\\
R_i(attack) &= \begin{cases} -0.001c + 0.3\ (if:\ killed\ the\ dragon) \\ -0.001c - 0.1\ (if:\ didn't\ kill\ the\ dragon) \end{cases}
\end{aligned}
$$

$$
\begin{aligned}
R_g(escape) &= (0.999)^f && (5.3)\\
R_i(escape) &= \begin{cases} -0.001c - 0.5\ (if:\ dead) \\ -0.001c\ (if:\ alive) \end{cases}
\end{aligned}
$$

$$
R_g(efficient) = (0.99)^f \qquad (5.4)
$$

環境に存在する 3 体のエージェントはすべて同じ報酬を与えて学習を行う．攻撃，生存志向エージェントの個別報酬は，そのエージェントがドラゴンを倒したかどうかで異なる．攻撃志向エージェントの場合はドラゴンを倒せば，生存志向エージェントの場合は自分が生き残れば高い個別報酬となる．また，早めのクリアを目指すことやボタン変更数が多くないほうがよいことは共通しているため，デフォルトエージェントと同じ報酬である．

これらに対して効率志向エージェントの報酬は，デフォルトエージェントの報酬 R_g を基にクリア時間による変化量を強く反映させたものである．また，ボタン変更による負の報酬を無くすことで，効率的な人間プレイヤの焦りを表現する．

与える報酬以外の学習時の設定は，5.2 の予備実験のものと同じにして学習を行った．

5.3.2 性能評価

各意図エージェントは 500 万ステップ学習を行った．また図 5.2 のように，デフォルトエージェント 2 体をペアにしたゲームをそれぞれ 2000 試行行うことで評価した．例えば攻撃志向モデルでは，攻撃志向エージェント 1 体，デフォルトエージェント 2 体のチームでゲームを行ったときの攻撃志向の結果を示している．なお，比較対象としてデフォルトエージェントがあるが，こちらもデフォルトエージェント 2 体をペアにしたもの（3 体とも同じ）である．各意図エージェントの学習結果を表 5.4 に示す．また，評価項目は次の通りである．

- クリア率：終了したゲームのうち，クリアした割合
- 攻撃率：クリアしたゲームのうち，評価対象のエージェントが攻撃した割合
- ボタン変更数：終了したゲームのうち，1 ゲーム中に評価対象のエージェントがボタン操作を変更した平均回数
- ボタン変更率：終了したゲームのうち，全エージェント中で評価対象のエージェントがボタン操作を変更した割合
- エピソード長：クリアしたゲームと時間切れとなったものについて，1 ゲームにかかった時間

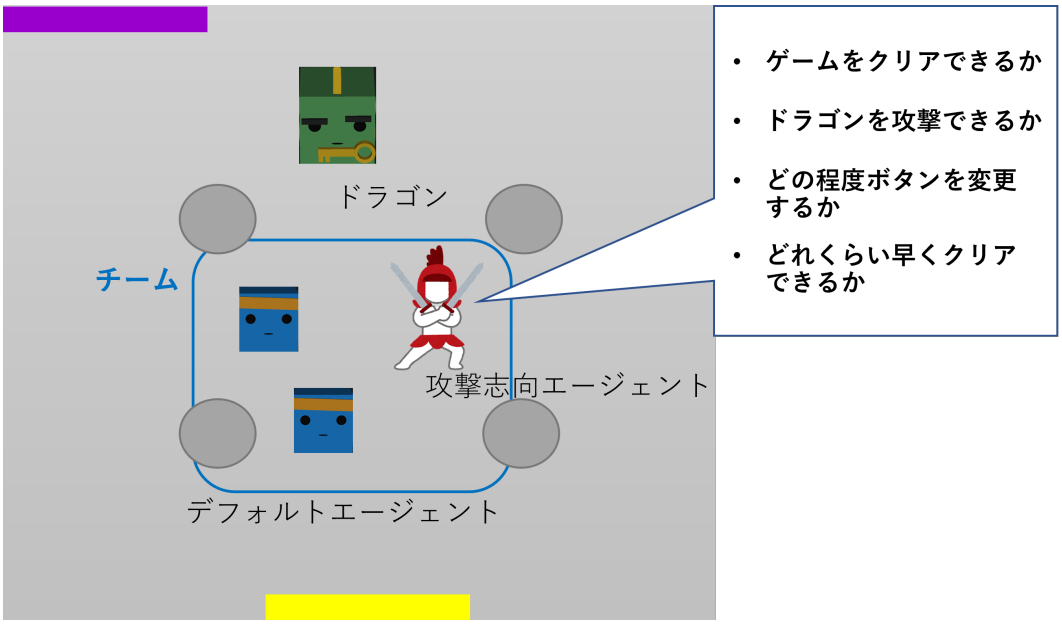

図 5.2: 攻撃志向エージェントのシミュレーション環境

表 5.4: 意図エージェントのシミュレーション結果

評価対象の AI	クリア率	攻撃率	ボタン変更数	ボタン変更率	エピソード長
デフォルト	0.916	0.331	6.1		141.1
攻撃志向	0.941	0.408	6.8		139.3
生存志向	0.871	0.012	47.9		230.5
効率志向	0.906	0.317	11.5	0.47	124.7

攻撃志向エージェントでは，クリア率は，0.94 とデフォルトエージェントと比較して 0.025 の増加があった．また，攻撃率は 0.408 であり，4 割以上攻撃を行っていることが確認できる．これは，攻撃志向エージェントが積極的にドラゴンへ攻撃していることを示す．実際のプレイ画面でも，ドラゴンを倒すことに報酬のないデフォルトエージェントよりも積極的に素早く行動していることが確認できた．ボタン変更数と，エピソード長には大きな変化が見られなかった．

生存志向エージェントでは，クリア率は，0.87 とデフォルトエージェントと比較して 0.045 減少した．これは，デフォルトエージェントは基本的にドラゴンに近いエージェントが攻撃するような振る舞いを獲得していたのに反して，生存志向エージェントが攻撃しないため本来クリアできるゲームが失敗していたためであると考える．また，攻撃率は 0.012 であり，ほとんど攻撃しなかったことが確認できる．実際のプレイ画面でも，生存志向エージェントはドラゴンを常に観測しながら，ドラゴンから離れるように行動していることが確認できた．ボタン変更数は大きく増加している．ドラゴンから離れるために細かな動きが多かったことは興味深い結果となった．さらに，エピソード長は非常に大きな値をとった．これは，ドラゴンに接近しないことを優先する行動が，全体のクリアタイムを大きく低下させることにつながるという結果を示す．

効率志向エージェントでは，クリア率は，0.906 で大きな変化は見られなかった．攻撃率もおよそ 3 割程度であった．ボタン変更数は，11.5 とデフォルトエージェントの 2 倍ボタン操作を変更したしていることが確認できる．また，デフォルトエージェントとのボタン変更割合からも，デフォルトエージェントと似ている行動をとりながら，細かく動いていることが示された．加えて，エピソード長は 124.7 であり，デフォルトエージェントよりも早くゲームをクリアしていることが分かった．これは，効率志向の人間プレイヤの焦りと効率的なゲームプレイを表現している．

以上の結果から，攻撃志向エージェントでは攻撃率が高く，生存志向エージェントでは攻撃率が低い，また効率志向エージェントではボタン変更数の増加とクリアタイムの短縮が見られた．これは，概ね狙い通りであり，3 種類の意図を表現したエージェントが作成できたと考える．ただし，これは意図から定義した行動を意図エージェントが獲得したものでる．人間プレイヤの意図を正確に表現しているとは限らない．

第6章　意図サポートAIの作成

特定の意図を持ったプレイヤを活躍させるためには，その意図をサポートするゲームAIが必要である．そこで本章では，5章で述べた意図エージェントを活躍させる意図サポートAIの作成について説明する．まず，5.1節で述べたような行動をとるそれぞれの人間プレイヤに対し，それを活躍させるためにサポートAIはどのような行動をとればよいか述べる．6.2節では意図サポートAIの学習手法を述べ，ねらいの行動が現れたかを評価する．

6.1　対象とするサポート行動

これまで意図を持ったプレイヤのとる行動を挙げた．ここでは，前章の形式に従って，対象とするサポート行動について述べる．各行動に対して，活躍させるとは何かを考察し，それぞれ活躍させる行動を1つ定める．表6.1は，各意図とそれをサポートする行動を対応させたものである．ただし，ここではそれぞれの意図をサポートする行動を活躍させる行動とする．

- 攻撃サポート
  攻撃志向プレイヤを活躍させるためのエージェントである．攻撃志向プレイヤは積極的にドラゴンに接近する行動をとりやすい．よって，これを活躍させるためにはドラゴンにできるだけ近づかず譲る行動をとる必要がある．ただし，活躍させたい対象が攻撃志向であるとしても，自分が攻撃しなければゲームオーバーになる（ドラゴンがポータルから逃げてしまう）場合は，ドラゴンを攻撃すべきである．このような挙動は，実は生存志向プレイヤと似ている．したがって，攻撃サポートAIは生存志向エージェントに与える報酬と同様にすることで作成できると考える．
- 生存サポート
  生存志向プレイヤを活躍させるためのエージェントである．生存志向プレイヤは生き残ることを優先するというものであり，ドラゴンから離れるような行動をとりやすい．故に，これを活躍させるためにはドラゴンに近づく行動をとる必要がある．そのうえで，自分が犠牲となってドラゴンを倒すことで，意図と異なる結果になることを防ぐ．このような挙動は，実は攻撃志向プレイヤの行

動と類似しているため，攻撃志向エージェントに与える報酬と同様に設計することで作成できると考える．

- 効率サポート
  効率志向プレイヤを活躍させるためのエージェントである．効率志向プレイヤは早期にゲームをクリアすることを優先するため，味方の位置によって，ドラゴンへ近づく行動をとるか鍵の出現場所で待機する行動をとるかに分かれる．よって，これを活躍させるためにはそれぞれの行動に対応できる必要がある．まず，効率志向エージェントがドラゴンへ近づく行動をとった時は，マップ中央で鍵を入手するように行動すればよい．鍵の出現場所で待機する行動をとったときは，ドラゴンへ攻撃するように行動すればよい．効率志向プレイヤが味方の初期位置によってどちらの行動をとるか判断しているのであれば，効率サポート AI は同じ判断基準を持つことでこれに対応できる．したがって，効率サポート AI は効率志向エージェントと同じ報酬を与える（ゲームクリアにかかった時間による報酬を強める）ことで作成する．

表 6.1: それぞれの意図をサポートする行動

意図	活躍させたいプレイヤの行動	サポート行動
攻撃志向	ドラゴンへの接近	ドラゴンへの攻撃を譲る
生存志向	ドラゴンから離れる	ドラゴンへ攻撃する
効率志向	味方の位置よって変化	味方の位置によって変化

6.2　意図サポート AI の作成と評価実験

意図サポート AI は，特定の意図を持った人間プレイヤを活躍させるものである．そのため，6.1 節では，各意図をサポートする行動と強化学習を用いた場合にそれを発生させるための報酬について述べた．ここでは，DungeonEscape を基に提案した拡張版 DungeonEscape 環境で，報酬を用いた強化学習エージェント 2 体に，意図エージェント 1 体をペアにして学習することで意図サポート AI を作成する．

6.2.1　意図サポート AI の学習方法

ここでは，3 種類の意図に対応した意図サポート AI を学習する．意図サポート AI の行動は，人間がプレイしたときに分かりやすくサポートしてくれることが望ましい．そこで，人間プレイヤの代替となる意図エージェントをペアにして学習を行う．図 6.1 は攻撃サポート AI を学習する環境である．攻撃サポート AI の学習時に，攻撃志向エージェントは味方として行動するが，方策の改善が行われるのは 2 体のサポート AI のみである（橙円の内部）．

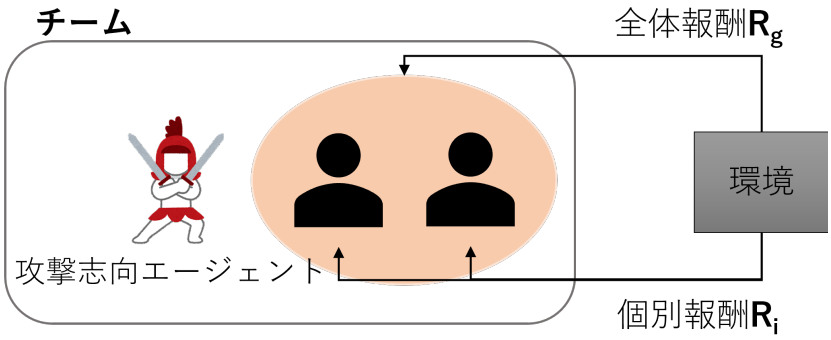

図 6.1: 意図サポート AI の学習

2 体の強化学習エージェントに与える報酬の概要は，6.1 節で説明した通りである．効率サポート AI は，結果的に効率志向エージェントの学習（同じ報酬を用いた 3 体の学習）とほぼ同じことをすることになるため，新たな学習は行わずに，効率サポート AI= 効率志向エージェントとして扱うことにする．攻撃サポート AI2 体は，攻撃志向エージェント（式 5.2 の報酬で学習済）1 体を相手に，それを活躍させるため式 6.1 の報酬を用いて学習する．これは，自分がドラゴンをできるだけ倒さないことを狙った報酬で，式 5.2 とは反対の志向性を持っている．反対に，生存サポート AI2 体は生存志向エージェント 1 体（式 5.3 で学習済）を相手に，式 6.2 の報酬を用いて学習する．これは，生存志向エージェントをできるだけ死なないようにする報酬で，式 5.3 を共有したものと言える．

与える報酬以外の学習時の設定は，5.2 節の予備実験のものと同じにして学習を行った．

$$R_g(sup\text{-}attack) = (0.999)^f \tag{6.1}$$

$$R_i(sup\text{-}attack) = \begin{cases} -0.001c—0.3\ (if:\ killed\ the\ dragon) \\ -0.001c + 0.1\ (if:\ didn't\ kill\ the\ dragon) \end{cases}$$

$$R_g(sup\text{-}escape) = (0.999)^f \tag{6.2}$$

$$R_i(sup\text{-}escape) = \begin{cases} -0.001c + 0.5\ (if:\ escape\text{-}agent\ is\ alive) \\ -0.001c\ (if:\ escape\text{-}agent\ is\ dead) \end{cases}$$

6.2.2 性能評価

各意図サポート AI の作成では，500 万ステップ学習を行った．また図 6.2 のように，意図エージェント 1 体をペアにしたゲームをそれぞれ 2000 試合行うことで性能を評価した．例えば，攻撃サポートモデルでは，攻撃志向エージェント 1 体，攻撃サポート AI2 体のチームでゲームを行ったときの攻撃志向の結果に注目する．この結果は表 6.2 にまとめるが，この際に攻撃サポート AI ではなくデフォルトエージェントが相手を務めた場合（図 5.2）の結果を再掲する．評価項目は，5.3.2 項や表 5.4 とほぼ同じで以下の通りである．

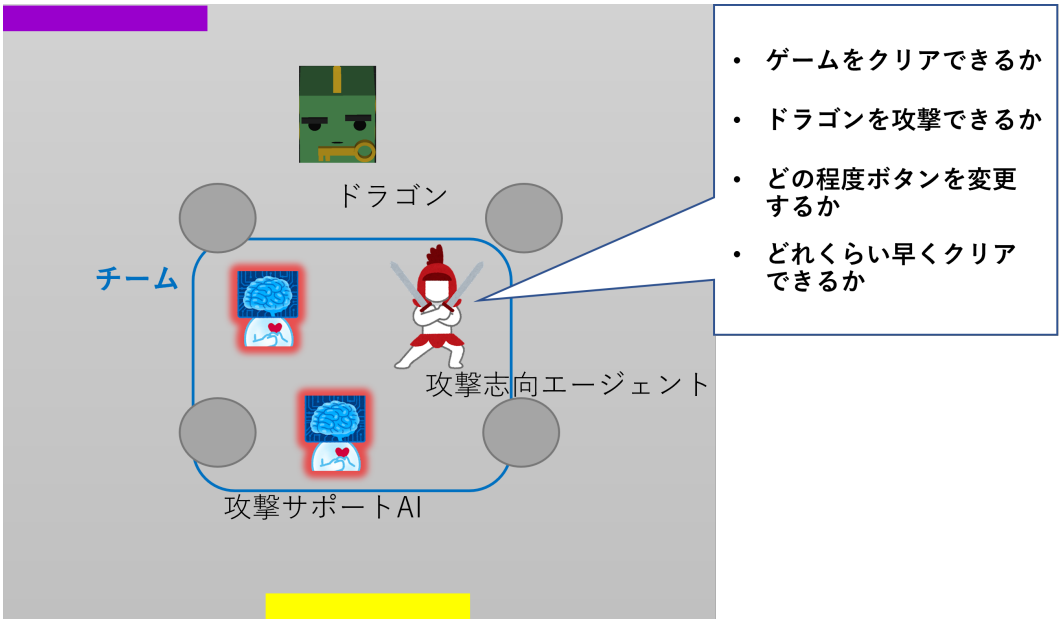

図 6.2: 意図サポート AI のシミュレーション環境

- クリア率：終了したゲームのうち，クリアした割合
- 攻撃率：クリアしたゲームのうち，評価対象のエージェントが攻撃した割合
- ボタン変更数：終了したゲームのうち，1 ゲーム中に評価対象のエージェントがボタン操作を変更した平均回数
- エピソード長：クリアしたゲームと時間切れとなったものについて，1 ゲームにかかった時間

表 6.2: 攻撃サポート AI シミュレーション結果の比較

サポート AI	クリア率	攻撃志向 AI の攻撃率	攻撃志向 AI のボタン変更数	エピソード長
デフォルト	0.941	0.408	6.8	139.3
攻撃サポート AI	0.821	0.896	6.2	173.4

表 6.3: 生存サポート AI シミュレーション結果の比較

サポート AI	クリア率	生存志向 AI の攻撃率	生存志向 AI のボタン変更数	エピソード長
デフォルト	0.871	0.012	47.9	230.5
生存サポート AI	0.860	0.005	55.5	251.8

表 6.4: 効率サポート AI シミュレーション結果の比較

サポート AI	クリア率	効率志向 AI の攻撃率	効率志向 AI のボタン変更数	エピソード長
デフォルト	0.906	0.317	11.5	124.7
効率サポート AI	0.910	0.323	10.6	113.0

表 6.2 は，攻撃志向エージェント 1 体をペアにした攻撃サポート AI2 体のシミュレーション結果を示している．まず，クリア率は 0.821 と 0.12 ほどの低下が見られた．このクリア率の低下は，攻撃サポート AI の攻撃を譲る振る舞いによるものである．また，攻撃率は 0.896 と非常に高い値をとった．ボタン変更数にはあまり大きな変化は見られなかった．エピソード長は大きく増加しており，これは，攻撃サポート AI がドラゴンから離れる行動をとっていたためであると考える．攻撃志向エージェントの攻撃率の増加から概ね攻撃志向をサポートできていると考える．ここで，クリア率の低下を検証する．デフォルトエージェントのままの攻撃率とクリア率であったとき，攻撃サポート AI1 体の期待報酬 R_1 は $0.941(0.704 \times 1.1 + 0.052 \times 0.7) = 0.924$

である．これが，攻撃サポート AI の攻撃率であるとき，予想されるクリア率はおよそ 0.85 である．したがって，攻撃サポート AI のクリア率 0.821 は改善の余地を残した．

表 6.3 は，生存志向エージェント 1 体をペアにした攻撃サポート AI2 体のシミュレーション結果を表している．クリア率は，0.860 とわずかに低下した．攻撃率は 0.005 となり，生存志向エージェントがドラゴンへ攻撃することがほとんどなかったことが確認できる．これは，デフォルトエージェントと比較して，生存サポート AI は生存志向エージェントに合わせた積極的なドラゴンへの攻撃行動を学習したためであると考える．ボタン変更数の増加は，生存志向エージェントの活動時間に起因するものである．またエピソード長はわずかに増加していることがわかる．実際のプレイ画面では，生存志向エージェントがドラゴンから離れ，生存サポート AI が 2 体ともドラゴンに向かうことが確認された．その後，マップ端まで避難した生存志向エージェント 1 体のみが環境に残る状況が多く，これがエピソード長を増加させたのではないかと考える．

最後に表 6.4 は，効率志向エージェント 3 体のシミュレーション結果を表している．クリア率，攻撃率，ボタン変更数ともに大きな変化は見られなかった．一方で，デフォルトエージェント 2 体をペアにした時と比較すると，エピソード長は短くなっていることが確認できる．これは，効率的にクリアしたいという意図をサポートすることができていると考える．ただし実際のプレイ画面では，エージェントが鍵の位置で待機するといった行動はあまり見られなかった．

以上の結果から，味方がデフォルトエージェントであるときと比較したとき，攻撃サポート AI のときは攻撃率が上昇し，生存サポート AI のときは攻撃率が低下した．効率サポート AI のときもクリアタイムが短縮され，3 種類の意図をサポートする行動が見られた．

第7章　意図推定器の提案

これまで，それぞれの意図に対応できる意図サポート AI を作成した．また，これが特定の意図に対して有効的に振る舞うことを確認した．そこで，本章ではプレイヤの意図を推定する意図推定器について述べる．

7.1　推定器の学習方法

本節では，意図推定器の学習に用いたネットワークと，入出力データについて述べる．

7.1.1　入出力データ

まず，表 7.1 に入力データとして用いた情報を示す．表 7.1 にある 1-5 までの情報をゲーム開始から 30 フレーム分扱う．故に，入力次元数は 5×30 である．意図推定器は，ゲーム中に対象プレイヤの意図を正しく推定する必要があり，かつ推定した段階でゲームが終了していないことが求められる．そこで，デフォルトエージェントの平均エピソード長が 141 フレームであることを踏まえて扱うフレーム数を決定した．ただし，このフレーム数は最適化されたものではないため今後の課題である．

表 7.1: 入力データ

番号	内容
1	ドラゴンと対象プレイヤとの距離
2, 3	対象プレイヤの位置座標 (x,y)
4, 5	ドラゴンの位置座標 (x,y)

入力データから対象プレイヤの意図が「攻撃志向」「生存志向」「効率志向」のどれに近いのか推定しなければならない．これらの意図ではそれぞれとる行動が異なり，特にドラゴンに近付くかどうかが重要である．そのため，このような入力を用いている．

出力データは，意図である．対象とした情報では効率志向の意図を判断することが難しいと考え，今回は攻撃志向と生存志向の 2 種類の意図を出力として扱う．

各意図を one-hot エンコーディングしたものを教師データとしてラベリングを行った．これは，攻撃志向を (1,0)，生存志向を (0,1) として扱うものである．今回の実験では，効率志向の判断を行っていない．今後の展望として，入力特徴量にボタン変更数を加えることで効率志向を含めた意図の判断がある．

7.1.2 ネットワーク構造

ネットワーク構造は，図 7.1 に示すような全結合モデルである．前節で述べたようにエージェントの位置座標，ドラゴンの位置座標，距離を 30 フレーム分取得した 150 次元のデータを入力として，全結合層に入れる．これを one-hot 表現された攻撃，生存の 2 次元データとして出力する．なお，全結合層の活性化関数には ReLU を用いる．本研究ではこのような単純なネットワークを用いたが，入力情報量が増え，複雑になる場合は，LSTM モデルのような時系列データを扱うことを得意とするモデルを利用することも必要になると考える．このようにして，攻撃，生存志向それぞれ 2000 ゲーム分のデータを取得して，これを学習 2800，テスト 1200 として用いる．

また，学習時のパラメータは表 7.2 の通りである．

表 7.2: 意図推定器の学習時パラメータ

Hyperparameter	
Minibatch Size	8
Max Epoch	1000
Optimizer	Adam

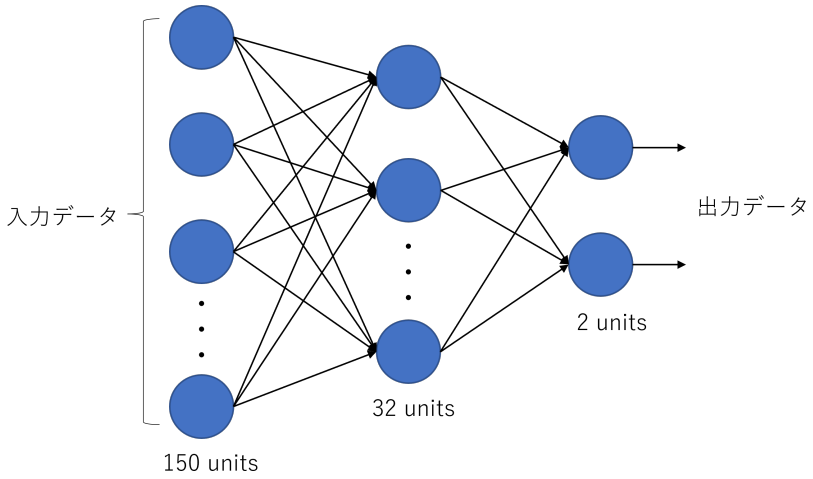

図 7.1: ネットワーク構造

7.2　推定器の性能評価

前節の設定で学習したモデルの学習曲線を図 7.2 に示す．前述のとおり，学習用に用いたデータ数は攻撃，生存志向で 1400 × 2，推論用に用いたデータ数は攻撃，生存志向で 600 × 2 である．また，図 7.3 は混同行列を示す．これは［攻撃，生存］を [1, 0] として表現したものである．学習時正解率は，0.9 を超えていることが確認できる．また，図 7.3 からテスト時精度は 0.905（平均誤差 0.304）であった．攻撃，生存志向ともに概ね学習できていると判断し，これを統合意図サポート AI として組み込む．

また 7.1.1 節でも述べたが，効率志向を判断する意図推定器は今回作成には至らなかった．入力特徴量を踏まえた見直しが必要であると考えるため，これは今後の課題とする．

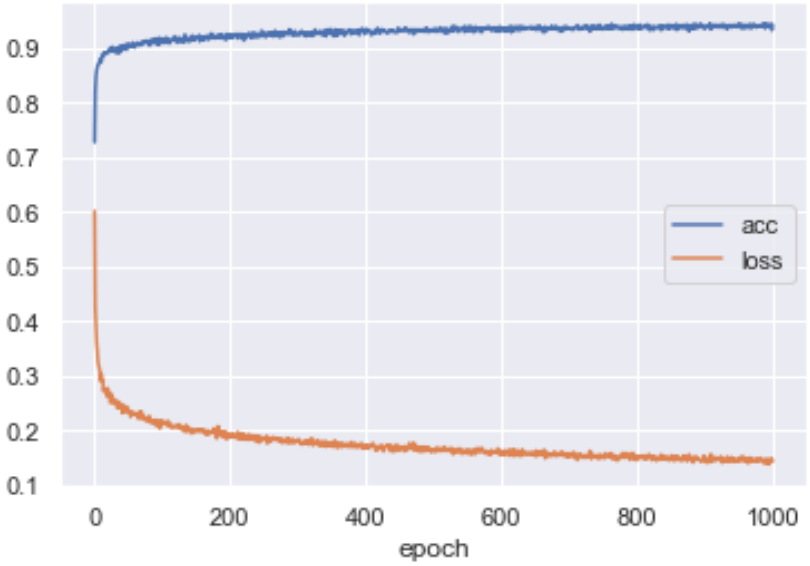

図 7.2: 学習曲線

図 7.3: 混同行列

第8章　統合意図サポートAIの作成

6章では意図サポートAIを作成した．また，評価実験よってそれぞれの意図をサポートすることが可能であることを示した．7章では，意図推定器を作成した．こちらも評価実験を行うことで，攻撃，生存志向の行動を概ね正しく推定できることが分かった．最後に，本章では意図推定器をゲーム内に組み込み，ゲーム中の行動から意図を判断して適切な意図サポートAIをあてがう統合意図サポートAIを提案する．

8.1　ゲーム中の意図推定手法の提案

ゲーム中に行動から意図を推定するためには，いくつかの条件が求められる．本節では，ゲーム内で意図を推定し，適切な意図サポートAIへの切り替えを行うための設定について述べる．

まず，意図に基づいた行動が発生するタイミングが問題となる．人間プレイヤはゲーム中に意図が変わることがある．この意図が変わるタイミングは一定ではなく，規則性も明らかになっていない．本来は，このタイミングを推定することができなければ，行動から意図を推定することは困難である．しかし，本研究で扱う拡張版DungeonEscapeではゲームの平均エピソード長が140フレーム程度である点から，1ゲーム中にプレイヤの意図が変化することはあまりないと考える．また，人間プレイヤはゲーム開始時から何かしらの意図を持っており，その意図に沿って行動すると考える．故に，本ゲームでは開始時点で持っている1つの意図に従って行動し，途中で変化することはないと仮定して，ゲーム中のあるタイミングで一度だけ推定することで意図に沿ったサポートが可能になると想定する．

また，どの程度の時系列データを入力として与えれば，高い精度で推定できるのかという問題もある．前提として，ゲームが終了するより前に意図推定ができている必要がある．7章の意図推定器の実験では，ゲーム開始から30フレーム分の行動を入力としていた．この条件では結果的に，“意図エージェントの”意図を高確率で見抜くことはできた．しかし，人間プレイヤの場合，仮に最初から意図を持っていたとしても意図エージェントほど素早く迷いなく意図に応じた行動を取るわけではないため，30フレームで十分な推定精度が得られるかは今後の課題である．

以上を踏まえて，ゲーム中の統合意図サポートAIの振る舞いを示したものが図

8.1 である．ゲーム開始から 30 フレームの間，統合意図サポート AI は攻撃，生存サポート AI を非活性化する．この区間はプレイヤのみ行動が可能である．この区間の時系列データを入力として推定を行い，31 フレーム時点で推定結果に合った意図サポート AI のみ活性化させる．なお，1 つのエージェントのモデルを変更するのではなく，非活性化区間を用いる理由は，Unity 上でゲーム中の円滑なモデルの変更ができないためである．

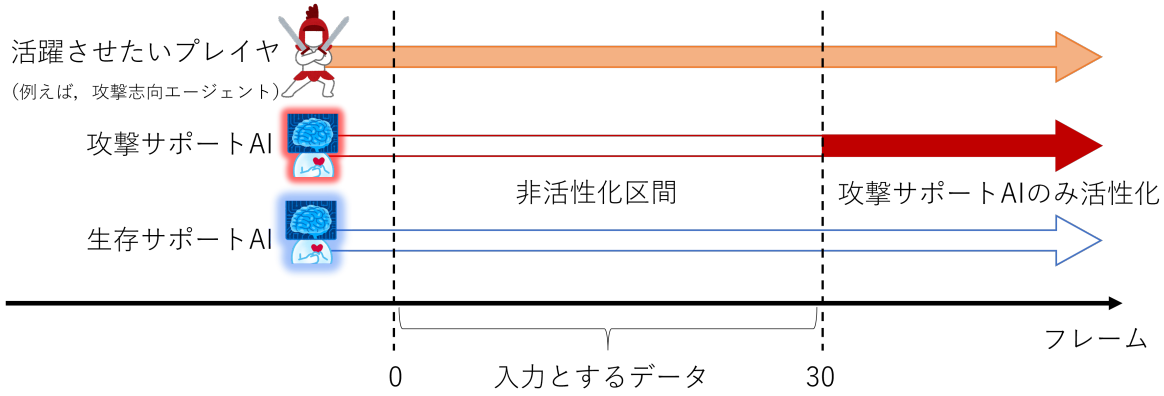

図 8.1: 統合意図サポート AI

8.2 統合意図サポート AI の性能評価

統合意図サポート AI によって意図に対応したサポートが可能となったかを明らかにする．性能評価のために，個別意図サポート AI をペアにしたものと統合意図サポート AI をペアにしたものをそれぞれ 2000 試合シミュレーションした．表 8.1 は，活躍させたいプレイヤとして攻撃志向エージェントを用いたときのシミュレーション結果である．同様に，表 8.2 は，活躍させたいプレイヤとして生存志向エージェントを用いたときの結果である．また，評価項目は以下の通りである．

- クリア率：終了したゲームのうち，クリアした割合
- 攻撃率：クリアしたゲームのうち，評価対象のエージェントが攻撃した割合
- 意図推定結果［攻撃志向：生存志向］：意図推定器がゲーム開始から 31 フレーム時点で推定した結果

表 8.1: 統合意図サポート AI のシミュレーション結果の比較（攻撃志向）

攻撃志向 AI とペアを組む 2 体のエージェント	クリア率	攻撃率	意図推定結果
デフォルト	0.941	0.408	
攻撃サポート AI	0.821	0.896	
統合意図サポート AI	0.810	0.774	1787:213

表 8.2: 統合意図サポート AI のシミュレーション結果の比較（生存志向）

生存志向 AI とペアを組む 2 体のエージェント	クリア率	攻撃率	意図推定結果
デフォルト	0.871	0.012	
生存サポート AI	0.860	0.005	
統合意図サポート AI	0.856	0.010	161:1839

表 8.1 から，統合意図サポート AI をペアにしたときのクリア率は 0.81 で攻撃サポート AI のクリア率から大きな変化は見られなかった．攻撃率は 0.774 となり，0.896 から 14 %ほど低下した．意図推定結果を見ると 2000 ゲーム中 1787 ゲームで正しく判断できており，正解率は 0.89 程度であることが分かる．故に攻撃率の低下は概ね妥当なものであると考える．

表 8.2 から，統合意図サポート AI をペアにした時のクリア率は 0.856 で生存サポート AI のクリア率とほとんど変らなかった．攻撃率は 0.01 と 0.005 増加していることが分かった．意図推定結果を見ると 2000 ゲーム中 1839 ゲームで正しく判

断できており，正解率は 0.92 である．攻撃率がデフォルトエージェントの 0.012 付近になったのは，統合意図サポート AI が 30 フレームの間非活性化することが原因であると考える．

以上により，攻撃，生存の 2 種類の意図に限った意図推定では概ね良好な結果が得られた．しかし，これはあくまで意図エージェントを人間プレイヤの代わりに用いた実験であり，より多様な挙動をとる人間プレイヤに十分対応できるかを明らかにするためには別の実験を要する．次節では，ロバスト性の評価を試みる．

8.3 ロバスト性の評価

本節では，統合意図サポート AI のロバスト性を評価する．評価手法として，プレイヤが人間であることを考慮して意図エージェントに揺らぎや判断の遅れといったノイズ行動を発生させる．そのうえで，統合意図サポート AI の性能がどの程度低下するのかを見ることで，人間プレイヤをペアにしたときの汎用性を確認する．

8.3.1 実験概要

ロバスト性を確認するために，意図エージェントに複数のノイズを発生させる．実験項目は以下の通りである．

(A) 開始からランダムな時間まで，まったく動かないエージェント

(B) 開始からランダムな時間まで，異なる意図の動きをするエージェント

(C) 開始から 30 フレームまで，毎フレーム確率 ϵ でランダムな行動をするエージェント

(A) は，ランダムに与えられる時間 $t(0 < t < 30)$ まで動かない意図エージェントである．実際の人間は，ゲーム画面を認識するのにも，どのように行動するかを決定するのにも時間を要する．そのため，これは人間プレイヤの盤面認識の遅れを想定したものである．

(B) は，ランダムに与えられる時間 $t(0 < t < 30)$ まで意図エージェントが持つ意図ではない意図に沿った行動を行う．今回は，攻撃志向と生存志向を対象としているため，攻撃志向エージェントは定められた時間までドラゴンから離れる行動をとり，生存志向エージェントはドラゴンに近づく行動をとる．人間プレイヤは意図を持っていても，短期間においては意図とは異なる行動をとってしまうことがあると考える．統合意図サポート AI はこのような状況に適応するのかを明らかにしておく必要がある．

(C) は，開始から意図を推定するまで，確率 ϵ でランダム行動を行う意図エージェントである．人間プレイヤは自身の位置を正しく把握し，正確なボタン操作を行うことができないことがある．これは人間プレイヤの行動のゆらぎからくるものであり，(C) はそれを表現している．今回は $\epsilon = 0.1$ として検証する．

8.3.2 評価結果

(A)-(C) のノイズを付与した攻撃志向と生存志向をそれぞれ統合意図サポート AI のペアにして，各 2000 ゲームシミュレーションした．また，ノイズのない攻撃志向と生存志向をペアにしたものと比較を行った．表 8.3，表 8.4 に，統合意図サ

ポート AI 性能評価結果を示す．表 8.3 は，攻撃志向エージェントに (A)-(C) のノイズを含めた場合の性能評価である．同様に，表 8.4 は，生存志向エージェントに (A)-(C) のノイズを含めた場合の性能評価である．

- クリア率：終了したゲームのうち，クリアした割合
- 攻撃率：クリアしたゲームのうち，評価対象のエージェントが攻撃した割合
- 意図推定結果［攻撃志向：生存志向］：意図推定器がゲーム開始から 31 フレーム時点で推定した結果

表 8.3: 統合意図サポート AI のロバスト性の評価（攻撃志向）

活躍させたい対象のモデル	クリア率	攻撃率	意図推定結果
攻撃志向エージェント	0.810	0.774	1787:213
(A)	0.732	0.763	1839:161
(B)	0.917	0.474	741:1259
(C)	0.821	0.739	1622:378

表 8.4: 統合意図サポート AI のロバスト性の評価（生存志向）

活躍させたい対象のモデル	クリア率	攻撃率	意図推定結果
生存志向エージェント	0.856	0.010	161:1839
(A)	0.631	0.027	962:1038
(B)	0.743	0.202	865:1135
(C)	0.888	0.008	89:1911

(A) は，ランダムな時間までまったく動かない意図エージェントである．

表 8.3 の (A) を見ると，クリア率は 0.732 であり，基本的な攻撃志向エージェントのクリア率 0.810 と比較して 0.078 ほど低下した．攻撃率には大きな変化はなく，意図推定結果は 9 割を超えるゲームで正しく推定できていることを示している．クリア率が低いのは，攻撃志向が一定時間停止することでドラゴンを攻撃する機会を損失していることに起因すると考える．

表 8.4 の (A) ではクリア率は，0.631 と大きく低下している．また，攻撃率は 0.027 であった．さらに，意図推定結果ではおよそ半分程度しか正しく推定できていないことが分かる．この結果は，2000 ゲームのうち 962 ゲームで攻撃志向であると推定され，ペアとして（ドラゴンへの攻撃を譲る）攻撃サポート AI が適用されたということである．故に，ドラゴンへの攻撃役がいなくなり，クリア率の低下，生存志向の攻撃率の増加につながったのだと考える．

(A) を総合してみると，停止するという行動は攻撃志向だと判断されやすいことが分かった．

(B) は，ランダムな時間まで意図に基づいた行動と真逆の行動をとる意図エージェントである．

表 8.3 の (B) を見ると，クリア率は 0.917 であり，基本的な攻撃志向エージェントのクリア率 0.810 と比較して 0.107 増加した．攻撃率は 0.474 となり大きく低下している．意図推定結果では正解率が 0.371 となっており，半数以上誤った判断をしていることが分かった．攻撃率の大きな低下は，この誤った判断により生存サポート AI（ドラゴンへ率先して攻撃する）がペアとして適用された結果であると考える．

表 8.4 の (B) でもクリア率は，0.743 と低下している．また，攻撃率は 0.202 と 2 割程度は攻撃していることが分かった．意図推定結果では表 8.3 同様に正解率はおよそ半分であった．実際のプレイ画面を確認すると，攻撃率の増加は，ランダムな時間ドラゴンへ近づく行動のせいでドラゴンに触れてしまうことが原因であった．また，クリア率の低下は意図推定器の誤った判断によりドラゴンへ攻撃するプレイヤがいなくなるためであると考える．

(B) を総合してみると，意図に基づいた行動とは真逆の行動をとるため，予想通り推定器は正確に機能しなかった．現状，行動のみから意図推定を行うためこの結果は許容せざるを得ないものであると考える．

(C) は，意図推定を行うフレーム中，確率 0.1 でランダムな行動をとる意図エージェントである．

表 8.3 の (C) を見ると，クリア率には大きな変化は見られなかった．また攻撃率は 0.739 となり基本的な攻撃志向エージェントと比較して 0.035 低下した．意図推定結果の正解率は 0.811 で，こちらも大きな変化はなかった．攻撃率の低下は，単純にランダム行動によるものと考えている．

表 8.4 の (C) でもクリア率に大きな変化は見られなかった．また，攻撃率もあまり変化していないことがわかる．意図推定結果では正解率 0.956 と基本的な生存志向エージェントの正解率 0.919 と比較するとわずかな増加が見られた．

(C) を総合してみると，今回は $\epsilon = 0.1$ であったため，ランダム行動をとる確率が低く，有意な差が生まれなかったと考える．

以上の (A)-(C) のノイズを加えた意図エージェントをプレイヤとすることで，実際の人間プレイヤとプレイする場合のロバスト性を検証した．(C) を除き，クリア率や攻撃率の性能が下がっているものが多く，実用的な利用には改善の余地があることがわかった．

8.4　人間的な性能評価

本節では，統合意図サポート AI の性能を人間的側面から評価する．人間プレイヤとの被験者実験を行い，人間をサポートできているか，満足度は向上したかを見ることで本手法の実用性を明らかにする．

8.4.1　実験手法の提案

評価実験では，これまでの意図エージェントにかわり，人間プレイヤとの協力プレイを行う．なお，本節では実験手法の提案までを行うこととする．次にアプローチを示す．

環境は，本論文で論じた拡張版 DungeonEscape である．人間プレイヤは［前進，右移動，左移動，後進］の操作が可能である．ここで，意図エージェントはレイセンサ観測であったが，人間プレイヤは 3 人称視点での全体観測ができるように変更している．

人間プレイヤは本環境になれるまで複数ゲームを行う．そのうえで，以下の AI プレイヤをペアにしてそれぞれ複数回ゲームをプレイしてもらう．

- デフォルトエージェント 2 体
- 攻撃志向エージェント 2 体
- 生存志向エージェント 2 体
- 攻撃志向エージェント 1 体，生存志向エージェント 1 体
- 統合意図サポート AI2 体
- 統合意図サポート AI1 体，デフォルトエージェント 1 体

評価は，アンケートによって行う．質問項目として「ゲーム中のあなたの考えはどちらに近いか（攻撃志向，生存志向）」「味方 AI をどれほど好ましく感じたか（5 段階評価）」などが挙げられる．

以上のような手順により，人間的な性能評価が行えると考えた．しかし，今回はその実施までは至らず手法の提案にとどまった．故に，このような被験者実験は今後の課題とする．

第9章　おわりに

本研究では，マルチプレイヤゲームにおいて人間プレイヤを引き立てて活躍させる味方 AI の作成を目指した．これは，人間プレイヤの意図をサポートすることで実現可能であると考えた．そして，その研究のため意図と協調行動が分かりやすい環境として既存のゲームを拡張した新しいゲームを提案した．

提案したゲーム環境で，人間プレイヤの意図として代表的なものを考察した．実際の人間プレイヤのデータを収集することはコストの面で困難であったことから，人間プレイヤの代替となる，意図に基づいた行動を行う意図エージェントを作成した．意図に対応する報酬を用いてマルチエージェント強化学習した意図エージェントは，意図に基づいた行動をとることが性能評価実験により明らかとなった．

さらに，この意図エージェントをペアにしてサポート行動をとる意図サポート AI を作成した．意図サポート AI は意図エージェントの意図に沿った行動に報酬を与えることで実現し，それぞれの意図に対して，それに対応する意図サポート AI が有効であることを示した．

また，意図エージェントの行動ログを用いて意図推定器の作成を行った．意図推定器の評価実験によって，入力データとして「プレイヤの位置座標，敵の位置座標，敵との距離」を入れることで，単純なネットワークモデルでも分類精度が 0.905 となることが示された．

最後に，意図推定器と意図サポート AI を組み合わせた統合意図サポート AI を作成した．評価実験では，各意図エージェントに対して概ねサポートできていることが明らかになった．さらに，プレイヤが人間であることを考慮して揺らぎや判断の遅れを入れることで，統合意図サポート AI のロバスト性を検証した．結果として，現状の統合意図サポート AI は人間プレイヤに対して有効的に活躍させることは難しいが，ある程度明確な意図があれば，意図に沿ったサポートが可能であることが分かった．

今回は機械的な評価実験にとどまり，被験者実験まで至らなかった．今後の展望として，統合意図サポート AI の人間プレイヤへの利用が期待される．また，本研究で扱った 2 種類の意図以外にも，効率志向などのさまざまな意図を扱うことで，より実用的な環境への適応を行いたい．

参考文献

[1] Mnih, V., Kavukcuoglu, K., Silver, D., et al. “Human-level control through deep reinforcement learning,” nature, vol.518, no.7540, pp.529-533 (2015).

[2] Jaderberg, M., Czarnecki, W. M., Dunning, I., et al. “Human-level performance in 3D multiplayer games with population-based reinforcement learning,” Science, vol.364, pp.859-865 (2019).

[3] 和田尭之，佐藤直之，池田心．“少数の記録からプレイヤの価値観を機械学習するチームプレイ AI の構成,” 研究報告ゲーム情報学，no.5，pp.1-8 (2015).

[4] 藤井叙人，佐藤祐一，若間弘典ほか．“生物学的制約の導入によるビデオゲームエージェントの 「人間らしい」 振舞いの自動獲得,” 情報処理学会論文誌，vol.55，no.7，pp.1655-1664 (2014)．

[5] ツインビー，KONAMI, https://www.konami.com/games/jp/ja/products/dl_wii_twinbee_vc/ (access: 2023-01-24).

[6] モンスターハンターポータルサイト，CAPCOM，https://www.monsterhunter.com/ (access: 2023-01-24).

[7] Sen, S., and Weiss, G. “Learning in multiagent systems,” Multiagent systems: A modern approach to distributed artificial intelligence, pp.259-298 (1999).

[8] Widergren, S. E., Roop, J. M., Guttromson, R. T., et al. “Simulating the dynamic coupling of market and physical system operations,” IEEE Power Engineering Society General Meeting, vol.1, pp.748-753 (2004).

[9] Cao, Y., Yu, W., Ren, W., et al. “An Overview of Recent Progress in the Study of Distributed Multi-Agent Coordination,” in IEEE Transactions on Industrial Informatics, vol.9, no.1, pp.427-438 (2013).

[10] Gupta, J. K., Egorov, M., Kochenderfer, M. “Cooperative multi-agent control using deep reinforcement learning,” In: International conference on autonomous agents and multiagent systems, pp.66-83 (2017).

[11] Kraemer, L., Banerjee, B. “Multi-agent reinforcement learning as a rehearsal for decentralized planning,” Neurocomputing, vol.190, pp.82-94 (2016).

[12] Chang, Y., Ho, T., and Kaelbling, L. “All learning is local: Multi-agent learning in global reward games,” Advances in neural information processing systems, vol.16 (2003).

[13] Littman, M. L. “Markov games as a framework for multi-agent reinforcement learning,” In: Machine learning proceedings 1994, pp.157-163 (1994).

[14] Foerster, J., Farquhar, G., Afouras, T., et al. “Counterfactual Multi-Agent Policy Gradients,” Proceedings of the AAAI conference on artificial intelligence, vol.32, no.1, (2018).

[15] Jaderberg, M., Czarnecki, W. M., Dunning, I., et al. “Human-level performance in 3D multiplayer games with population-based reinforcement learning,” Science, vol.364, pp.859-865 (2019).

[16] Strouse, D. J., McKee, K., Botvinick, M., et al. “Collaborating with humans without human data,” Advances in Neural Information Processing Systems, vol.34 (2021).

[17] McIlroy-Young, R., Sen, S., Kleinberg, J., et al. “Aligning superhuman ai with human behavior: Chess as a model system,” In Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp.1677-1687 (2020).

[18] Kitano, H., Asada, M., Kuniyoshi, Y., et al. “Robocup: The robot world cup initiative,” In Proceedings of the first international conference on Autonomous agents, pp. 340-347 (1997).

[19] 松原仁，浅田稔，北野宏明．“ロボカップ ロボカップの歴史と 2002 年への展望,” 日本ロボット学会誌，vol.20, no.1, 2-6 (2002).

[20] José, B. R., Samuel, M., Rui P. “Game Mechanics for Cooperative Games,” in proceedings of ZON Digital Games 2008, pp.72-80 (2008).

[21] Minecraft official, `https://www.minecraft.net/` (access: 2022-01-23).

[22] Johnson M., Hofmann K., Hutton T., et al. “The Malmo Platform for Artificial Intelligence Experimentation,” Proc. 25th International Joint Conference on Artificial Intelligence, pp4246-4247 (2016).

[23] StarCraft2 official, `https://starcraft2.com/` (access: 2022-01-23).

[24] Vinyals, O., Ewalds, T., Bartunov, S., et al. "Starcraft ii: A new challenge for reinforcement learning," arXiv preprint, arXiv:1708.04782 (2017).

[25] Vinyals, O., Babuschkin, I., Czarnecki, W. M., et al. "Grandmaster level in StarCraft II using multi-agent reinforcement learning," Nature, vol.575, no.7782, pp.350-354 (2019).

[26] Greg, B., Vicki, C., Ludwig, P., et al. "OpenAI Gym," arXiv eprint, arXiv:1606.01540 (2016).

[27] Terry, J., Black, B., Grammel, N., et al. "Pettingzoo: Gym for multi-agent reinforcement learning," Advances in Neural Information Processing Systems, vol.34, pp.15032-15043 (2021).

[28] Juliani, A., Berges, V., Teng, E., et al. "Unity: A general platform for intelligent agents," arXiv preprint, arXiv:1809.02627 (2020).

[29] Example Learning Environments, Unity ML-Agents Toolkit, `https://github.com/Unity-Technologies/ml-agents/blob/release_17/docs/Learning-Environment-Examples.md#dungeon-escape` (access: 2022-01-21).

[30] Cohen, A., Teng, E., Berges, V. P., et al. "On the use and misuse of absorbing states in multi-agent reinforcement learning," arXiv preprint, arXiv:2111.05992 (2021).