修士論文

人間の副目的を尊重した協力型ゲームの AI に関する研究

林 辰宜

主指導教員 池田 心

北陸先端科学技術大学院大学　金沢大学
融合科学共同専攻
（融合科学）

令和 7 年 3 月

人間の副目的を尊重した協力型ゲームの AI に関する研究
（Respecting human sub-objectives in cooperative game AI）

北陸先端科学技術大学院大学　学籍番号 2350007
氏名　林 辰宜
主任研究指導教員氏名　池田 心

1. はじめに

近年，深層強化学習の登場により，ゲーム AI は様々なゲームで人間のトッププレイヤに比肩・超越する強さを獲得しつつある．これは，プレイヤ間の協力を含む「協力型ゲーム」についても同様である．

一方，人間を楽しませるという観点おいては，必ずしも強い AI が優れているとは限らない．協力型ゲームでは，単に主目的を協力して達成するのみならず，仲間の好みや意図に合わせて挙動を調整することも，よいゲーム体験のために重要である．特に，人間は主目的とは関係の薄い副目的を有している場合もあり，そのような副目的に対しても，協力可能な AI を設計することは好ましいと考える．

協力型ゲームの味方 AI に関係する研究領域に，Population-based Training（PBT）と呼ばれる方法の枠組みがある．これらの方法では，初対面の人間との協調可能な AI の作成を目標としており，テスト時に人間とプレイさせるゲーム AI（本番用 AI）を 2 段階の工程で訓練する．1 つ目の工程では，本番用 AI を訓練するためのゲーム AI のプール（訓練用 Population）を本番用 AI と独立させて学習させる．2 つ目の工程では，訓練した Population からゲーム AI をサンプリングし，それを訓練相手として本番用 AI を学習させる．これらの学習は，人間のデータを必要としない強化学習によって行われる．それぞれが異なった振る舞いを見せるゲーム AI から構成される訓練用 Population が，現実の人間と同様の多様性を有している場合，それらとの協力経験を得た本番用 AI は，様々な人間との協力が可能になるとされる．

そのため，既存の PBT 研究は，Population を多様化させる方法が重点的に研究されてきた．中でも，Hidden Utility Self-Play（HSP）[1]では，人間の副目的を内包する表現である「効用」を Population 内の各エージェントに割り当てることで，人間的な特徴を備えた相手への対応が可能な本番用 AI を実現しようとした．一方，HSP を含む多くの PBT 手法では，主目的の達成を目的としており，本番用 AI も訓練相手の効用ではなく，高い主目的スコアを得る振る舞いを学習するような仕組みとなっている．また，Population に人間の副目的を導入しようとする HSP においても，訓練される Population は主目的を軸にした協調を前提としているため，人間でしばしば見られる「主目的とは関係の薄い副目的」に対する協調を学習しづらいという課題があると考えた．冒頭で述べたように，人間を楽しませるには，相手の主目的だけでなく，副目的への協調も想定することが重要である．この観点に関して，既存の手法では，十分に人間を満足させる本番用 AI を作成できないと考えた．

本研究では，HSP（原始 HSP）をもとに，より人間の副目的への協調が可能な 2 つの手法を提案した．

その 1 つは，(1)主目的とは関係の薄い副目的を有する Population の学習，(2)本番用 AI の訓練時に訓練相手 AI の効用への接待を可能とする「接待 HSP」である．原始 HSP の Population の訓練では，主目的を追求するゲーム AI と副目的を含む効用を追求するゲーム AI の組で学習が行われていたため，プレイヤ間での協調を必要とする副目的を試みる振る舞いを獲得しづらいという課題があった．(1)は，主目的，副目的を含みうる同一の効用を追求する AI 同士で学習を行うことで，これに対処しようとするものである．また，原始 HSP では，本番用 AI を訓練する際に，主目的の達成度を最大化させることを目指す学習を行っていた．(2)では，この設定を変更し，訓練相手の効用の満足度を最大化させるような設定とすることで，時には主目的を犠牲にしてでも相手に接待するような振る舞いを本番用 AI に獲得させることを試みた．

2 つ目の提案手法は，接待 HSP を発展させた「不満忖度 HSP」と呼称する手法である．接待 HSP の Population 訓練時には，様々な効用のゲーム AI が発生する過程で，外見上の振る舞いは類似している一方，効用を満足させるための接待戦略が相反するような AI の組み合わせが想定される．それらの AI を見分け，それぞれに適切な接待を行うのは，原理的に困難である．他方，人間同士のプレイでは，ゲーム内の伝達行動，例えば不満を示すような行動を取ることが知られている．不満忖度 HSP では，そのような人間の不満を示す行動を Population に付与し，本番用 AI との訓練時に訓練相手 AI が不満を感じた際に，特定の行動パターンを発生させる．相反する効用の AI では，不満を示す条件が異なるため，これにより，外見上の振る舞いに差異が生まれ，本番用 AI はそれぞれの AI を見分けた上で適切な接待が可能になると考えた．

2. 研究方法

本研究では，「Overcooked 環境[2]」を題材として，手法の検討・実装・評価を行った．このゲームでは，2人のプレイヤが協力して料理を作成・提出し，制限時間以内に可能な限り高いスコアを獲得することを主目的としている．本環境は，多くの Population-based Training のテストベッドとして使用されていると同時に，人間的な副目的を表現することが可能である．

本研究は，提案手法が有効性を示すような事例を Overcooked 環境で設計し，訓練用 Population の学習時の振る舞い，本番用 AI の相手 AI に対する接待能力を調査する実験により，提案手法の評価を行った．これらのデータは，学習における経過時間を横軸，評価指標の値を縦軸とする学習曲線の形で表現される．接待 HSP は原始 HSP，不満忖度 HSP は接待 HSP と学習曲線を比較することで，手法の優位性を示す．

3. 結果と考察

本論文では，接待 HSP について 2 つ，不満忖度 HSP について 1 つ，良好な結果を示した事例を述べる．

接待 HSP を対象とした実験では，「外見上の振る舞い，満足させる接待戦略が異なる 2 種類の AI（玉ねぎが好きな O，トマトが好きな T）への接待」および「主目的と関係の薄い副目的を有する AI（皿が置かれている状態が好きな F）への接待」の場合について，原始 HSP と接待 HSP の訓練用 Population，本番用 AI を比較した．前者の例では，2 種類の訓練相手 AI から構成される Population に対して，原始 HSP の本番用 AI は片方の AI のみを最適に近い形で接待した一方，接待 HSP の本番用 AI は両者を見分け，それぞれに適した接待を切り替えることが確認された．後者では，主目的と関係の薄い副目的を有するように学習を試みた AI のみで構成される Population に対して，原始 HSP の本番用 AI は相手を満足させる協調とは正反対の振る舞いを学習した一方，接待 HSP の本番用 AI は最適に近い接待を行うことが確認された．また，Population の訓練においても，原始 HSP では副目的に協調するような振る舞いを学習できなかったのに対し，接待 HSP はそのような振る舞いを学習できることが確認された．

不満忖度 HSP を対象とした実験では，「主目的とは関係が薄く，接待戦略が相反する 2 種類の AI（皿が置かれている状態が好きな F，自分で皿を置くのが好きな P）への接待」の場合について，接待 HSP と不満忖度 HSP の本番用 AI を比較した．その結果，接待 HSP の本番用 AI は両者を見分けることができなかった一方，不満忖度 HSP の本番用 AI は相手を見極め，それぞれ最適に近い接待を行うことが確認された．このことから，実際の人間との協力を想定したとき，不満を示すような行動パターンを有するエージェントを Population に含ませることで，積極的に不満を示すような人間をより満足させることが可能になると考える．

4. まとめ

協力型ゲームで人間を楽しませるような AI を設計するとき，単に主目的を協力して達成する以外にも，人間プレイヤの副目的を尊重した戦略を AI が取ることが，人間プレイヤのゲーム体験にとって重要となる．本研究では，Population-based Training の既存手法である Hidden Utility Self-Play（原始 HSP）を，よりこのような需要に応えられるよう発展させた「接待 HSP」および「不満忖度 HSP」を提案した．実験の結果，接待 HSP は原始 HSP に対して，不満忖度 HSP は接待 HSP に対して，特定の場合に協力相手への協調能力が高いことが示された．

本研究では，不満忖度 HSP を簡単のため，単純な条件分岐と機械的な行動パターンによって実装した．一方，この手法は，強化学習の主要な手法で採用されている TD 誤差や Q テーブルの値を用い，本研究で扱った「期待累積報酬が低くなる場合」を検知するというような一般化も可能と考えている．また，不満を示す挙動についても，人間一般に通ずるヒューリスティックや人間のデータをもとにモデリングすることで，実際の人間に類似したものに置き換えるような発展も可能である．これにより，実際の様々な人間との協調が可能な AI の実現に寄与するものと考える．

参考文献

[1] Yu, C., Gao, J., Liu, W., Xu, B., Tang, H., Yang, J., ... & Wu, Y. (2023). Learning zero-shot cooperation with humans, assuming humans are biased. arXiv preprint arXiv:2302.01605.

[2] Carroll, M., Shah, R., Ho, M. K., Griffiths, T., Seshia, S., Abbeel, P., ¥& Dragan, A. (2019). On the utility of learning about humans for human-ai coordination. Advances in neural information processing systems, 32.

目 次

図 目 次

表 目 次

第1章　はじめに

近年，人工知能（AI）に関する研究は，様々な革新を経験してきた．2012 年に AlexNet [1] が大幅に画像認識の精度を向上させたことを発端に，自動運転 [2]，ロボット制御 [3]，生命科学 [4] など，様々な分野が大きく飛躍を果たしている．直近では，大規模言語モデル（LLM）や生成モデル [5] の発展による，ChatGPT [6] をはじめとした生成系 AI サービスの登場がその代表的な成果物といえよう．

これら一連の技術を支えているのが深層学習の技術であり，ゲーム AI 研究もその恩恵を受けている領域である．ゲームはその商業的な影響力もさることながら，学術的研究の対象としても注目されてきた．ゲームは，現実空間の諸問題と比較して「実装・実験が容易であること」「様々な特徴を持ったゲーム（対象問題）があり，手法の特性の多角的な検証が可能であること」「条件を固定したシミュレーションが可能であり，再現性が高いこと」から，AI 研究のテストベッドとして度々使用されている [7] [8] [9].

特に，人間と同等あるいは超越した知性を機械に獲得させる挑戦の一環として，知名度・競技性の高いゲームにおけるコンピュータプレイヤ（ゲーム AI，あるいは単に AI と呼ぶ）の「強さ」を追求する試みは，長らくゲーム AI 研究の主流であり続けてきた．古くはチェスで当時の世界チャンピオンに勝利した「Deep Blue」[13] に始まり，近年ではより複雑なゲームである囲碁で世界のトップ棋士に勝利した「AlphaGo」 [14]，複雑な操作や戦略が要求される「StarCraft2」で人間の上位 0.2 %にランクインした「AlphaStar」 [15] 等が，その主要な金字塔である．

これらはいずれも 1 対 1 のゲームにおける功績であるが，複数プレイヤが協力する要素のあるゲームにおいても，ゲーム AI は超人的な強さを獲得しつつある．複雑なプレイヤ間の協力が要求される「Dota 2」の世界のチャンピオンチームに勝利した「OpenAI Five」 [16] が近年の代表的な事例である．OpenAI Five は，実戦時に協調する AI 同士と一緒に繰り返しプレイすることで学習を行っており，その問題設定は「互いに既知の AI 同士の協調」に当たる．これに対し，より広い問題設定である「互いに未知の AI 同士の協調」「人間と AI の協調」があるが，これらについても，ゲームによっては人間を上回る性能（強さ，勝率）を発揮可能な AI が登場してきている [17] [18].

一方，協力を含むゲームにおいて，人間を楽しませることを目的とする AI については，単なる「強さ」以外の要件も求められる場合がある．近年，ディジタルゲーム市場は拡大の一途を辿っており [19]，中でも協力型ゲームは，多くのユーザを抱えるオンラインゲームの中でも特にプレイ人口が多く [20]，その需要は高

まり続けている．そのようなゲームの AI をより人間を楽しませられる方向に改良していくことは，文化・商用の面で有益な試みといえる．

これらのゲームでは，必ずしも勝利のような主目的を目指すだけでなく，人間の副目的や好みが介在する余地が存在している．例えば，サッカーでは，プレイヤによって好みの戦術が異なったり，練習試合ならば勝つこと以上に陣形を作る等の目的があったりし，そのようなプレイヤの意図を汲み取る能力が重要となる．また，協力型ゲームでは，チームを組む味方の存在が内容に大きな影響を与えるが，その役を担う人間プレイヤを常に用意できるとは限らない．1 人や少人数でゲームを練習したい場合，あるいはオンラインでのプレイ時に人が集まらない，通信障害等でゲーム中に人間プレイヤが離脱した場合に，人間の代わりとしての機能を AI に期待することになる．このような要求に対して，単に強い AI では，人間特有の行動や意思疎通，人間が理解できるような思考といった要素を再現できるとは限らないため，別途の工夫が必要となる．

これらの課題に関連する研究には，様々なものが存在する．人間らしい行動や思考については，生物学的な制約を導入したアクションゲームの AI [21]，様々なゲームについて人間的な好みを備えた AI [22] [23] に関する研究事例がある．様々な人間プレイヤと協調可能な AI を目指した研究には，人間のプレイデータから人間的な振る舞いを模倣した AI との協調を学習させようとした研究 [24] [25] や，特定のゲームにおける人間の価値観や意図を推定・未来予測することで最適な行動の選択を目指した研究 [11] といったものがある．

近年では, 強化学習を用いた手法も盛んに研究されている．その中に,「Zero-Shot Coordination（ZSC)」 [27] として知られている問題設定があり，新たな関心事として注目が高まっている．ZSC の有力なアプローチの 1 つに，Population-based Training（PBT） [28] [25] と呼ばれる方法の枠組みがある．この枠組みでは，(1) テスト時に人間とプレイさせるゲーム AI（以降，本番用 AI）と，(2) 本番用 AI を訓練するためのゲーム AI のプール（訓練用 Population）とを独立して学習させる．(2) は様々な特性のゲーム AI を有するように学習され，そこからサンプリングした AI との協調を (1) に学習させる．これにより，PBT は様々な相手との協調を学習した本番用 AI の実現を図る．

本番用 AI が適切に協調可能な人間の範囲は，Population が有する特徴の多様性に依存する．そのため，既存の研究では，Population の多様性を向上させるための手法が重点的に研究されてきた．代表的な手法として，Population に様々な習熟度の AI を導入した Fictitious Co-Play（FCP） [29]，その改良として主目的に対する様々な準最適な戦略の AI を導入した Maximum Entropy Population-based Training（MEP） [31]，人間の好み・副目的を導入した Hidden Utility Self-Play（HSP） [32] が挙げられる．

これらの手法は，いずれも主目的に対するスコアを最大化することに重きが置かれてきた．しかし，先述の通り，人間には主目的以外の副目的が存在する場合があり，人間を楽しませるという観点では，副目的を尊重することも人間プレイヤの

ゲーム体験にとって重要となる．そこで，本研究では，人間の副目的に対するスコアの向上も可能となるよう，HSP（原始 HSP）を拡張した「接待 HSP」を提案する．原始 HSP では，主目的や様々な副目的を表現した「効用」を有する訓練相手 AI を訓練することで，人間の好み・副目的を導入した Population を構成する．しかし，ここでは主目的のみを追求するような相手との協力を前提とした訓練用 Population を構成するため，副目的のためのプレイヤ間の協調を試みるような訓練相手 AI がほぼ再現されない．そのため，本番用 AI の訓練時にそのような振る舞いの相手と相対する機会が得られず，類似した人間プレイヤと協調する術を学習することが困難となる．また，原始 HSP における本番用 AI の学習では，チーム全体として主目的を達成するための教師信号のみが与えられる．従って，相手の副目的に対するスコアを向上させるような戦略を獲得させるような運用に，原理上適していない．接待 HSP では，この 2 つの機構に対して変更を行い，副目的にも対応可能な本番用 AI の実現を図った．

また，本研究では，人間が初対面の人間プレイヤを相手にするときの振る舞いに注目し，接待 HSP へ上積みとして 2 つ目の手法「不満忖度 HSP」の提案も行う．接待 HSP の Population 訓練時には，様々な効用の訓練相手 AI が発生する過程で，外見上の振る舞いは類似している一方，効用を満足させるための接待戦略が相反するような AI の組み合わせが想定される．それらの AI を見分け，それぞれに適切な接待を行うのは，原理的に困難である．他方，人間同士のプレイでは，ゲーム内の伝達行動，例えば不満を示すような行動を取ることが知られている．不満忖度 HSP では，そのような人間の不満を示す行動を Population に付与し，本番用 AI との訓練時に訓練相手 AI が不満を感じた際に，特定の行動パターンを発生させる．相反する効用の AI では，不満を示す条件が異なるため，これにより，外見上の振る舞いに違いが生じる．よって，不満忖度 HSP の本番用 AI は，それぞれの AI を見分けた上で適切な接待が可能となり，これは実際の人間の中でも積極的な不満を示すような相手に対しても有効であると考えた．

本論文は，全 6 章および付録，参考文献から構成される．2 章では，本研究で対象問題として扱う「Overcooked 環境」および実験で使用したその周辺環境について説明する．3 章では，ZSC や PBT，人間の反応的な挙動に関連する研究を導入する．4 章では，人間を楽しませる観点における原始 HSP の課題，およびそれに対する「接待 HSP」の提案・有効性の検証を行う．5 章では，接待 HSP で生じ得る問題と「不満忖度 HSP」の核となる着想を述べ，本研究で扱う具体的な実装およびその有効性の検証を行う．6 章は，本研究のまとめと展望である．

第2章　対象とするゲーム

本章では，本研究の対象問題である「Overcooked 環境」について紹介する．

Overcooked 環境は，アクションゲーム「Overcooked！」 [26] を基に，研究用途に簡素化されたゲームである．このゲームでは，2 人のプレイヤがステージ内に用意されている材料や食器，調理器具を用いて料理を作成・提出し，一定時間内に可能な限り多くのスコアを獲得することを目指す．次のステージを例に，具体的なゲームの流れを説明する．

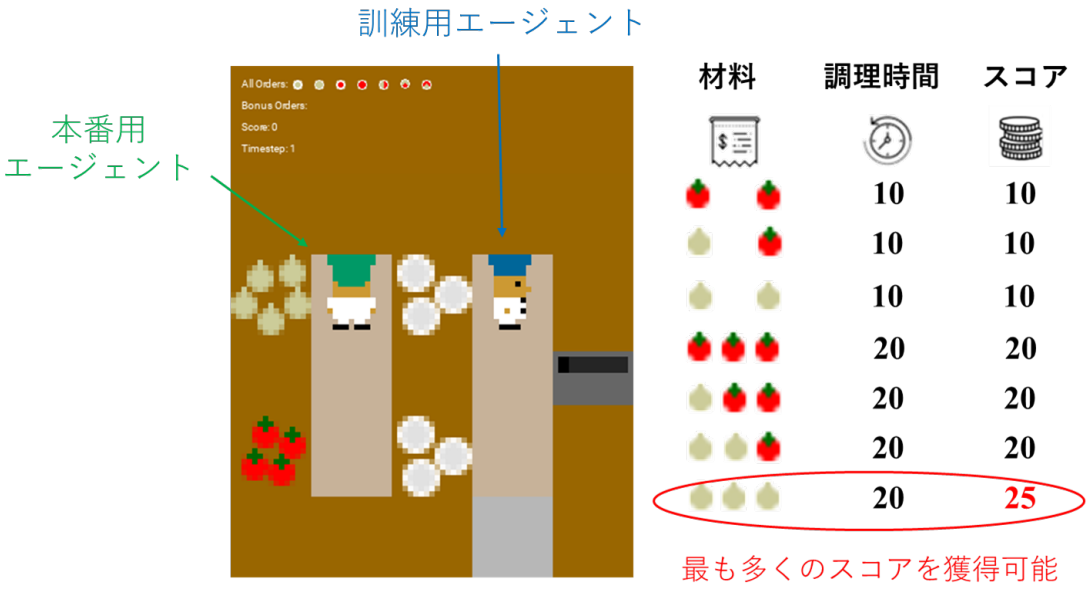

図 2.1: Overcooked 環境のステージ例

図の左側に映るのがゲームのマップ構成，右側の表が各料理のレシピとその調理時間，提出したときに得られるスコアである．プレイヤは左右上下への移動と「移動せず留まる」，目の前のオブジェクトから取る・目の前に手持ちのオブジェクトを置く・調理を行うための「干渉」の 6 つの行動を毎ステップ行うことができる．

マップの左側に置かれているオブジェクトは，それぞれ上から「玉ねぎの山」「トマトの山」であり，プレイヤは「干渉」を行うことで，その山に対応する食材を 1 つ取得することができる（山から取得できる食材の数に上限はない）．マップの中央に 2 つ配置されている「皿の山」についても同様である．茶色のマスはカウン

ターであり，プレイヤーが移動することはできないが，オブジェクトを所持しており，かつカウンターの方向を状態で「干渉」を行うことで，所持しているオブジェクトとカウンターの上に配置することができる．カウンターに配置されたオブジェクトをプレイヤが取得することもでき，その場合はオブジェクトの山から取得したときと異なり，オブジェクトはカウンターから消滅し，プレイヤの手に渡る．

右側に配置されているオブジェクトは「鍋」であり，食材（玉ねぎ，トマト）を所持したプレイヤが「干渉」することで，最大 3 個まで食材を入れることができる．食材が 1 つ以上入った状態で，オブジェクトを所持していないプレイヤが鍋に「干渉」すると，食材を消費した調理が開始される．調理に要する時間（単位：ステップ数）は，図の表のようにレシピと紐づいており，表以外の組み合わせの食材で調理した場合，固定で「10」となる．料理が調理された状態で，皿を持ったプレイヤが鍋に「干渉」を行うと，所持していた皿が「料理が盛られた皿」へと変化し，これを所持した状態で右下の提出口（灰色のマス）に向けて「干渉」することで，料理が提出される．このとき，得られるスコアは表の通りであり，表に載っていない料理を提出した場合は，「-10」のスコアが加算される．

Overcooked 環境は，過去の Population-based Training 研究で広くテストベッドとして使用されてきた．研究によって使用しているゲーム環境に差異があるが，本研究では，Hidden Utility Self-play [32] の研究で用いられた環境をもとにしたものを使用する．

第3章　関連研究

本章では，まずこれまでの協力型ゲームの AI 研究の概観を述べ，次に本研究に直接的に関係する「Population Based Training」における各手法，特に HSP の概要を述べる．

3.1　非競争的協力型ゲームの AI 研究

協力型ゲームの AI に求められる要件は，そのゲームのルール，特にプレイヤ同士の関係性に強く依存する．協力型ゲームの類別をするに当たっては，「チーム間で競争があるか否か」「協力が必須であるか否か」「協力する相手が固定であるか否か」「協力相手と主目的を共有しているか」等，様々な軸での分類が可能である．本研究の対象問題である「Overcooked 環境」が「チーム間の競争が存在しない」「協力相手は固定」「協力相手と主目的を共有している」ことに関連し，以降，これらの性質を有するゲームを「非競争的協力型ゲーム」と呼称するものとする．

非競争的協力型ゲームでは，単に主目的を協力して達成するのみならず，仲間の好みや意図に合わせて挙動を調整することも，よいゲーム体験のために重要である．例えば RPG では，一つ一つの敵チームとの戦闘に単に勝つだけでなく，その勝ち方が問題になる場合がある．人間プレイヤは「経験値稼ぎなのだからリスクを冒してでも素早く勝ちたい」「ボスに向かっているので，途中ではリスクを冒さず，MP も節約したい」などの意図や好みを持っているため，仲間 AI プレイヤはそれを見抜いて，合わせてあげることが求められる．戦闘中の人間プレイヤの行動から，モンテカルロ法を用いてその「効用」を推定して，その効用が高くなるように行動する手法が提案されており，人間プレイヤの満足度を高めることが示されている [11]．この考え方は 3.3 節に後述する HSP にも通じている．

他のアプローチとして，深層強化学習が脚光を浴びて以降，強化学習を用いる方法も積極的に研究されてきた．強化学習では，ある「環境」の「状態」をエージェントが（完全あるいは部分的に）観測し，それに基づいてエージェントが環境に対して「行動」を行う．それに対して，環境が「報酬」をエージェントに返し，エージェントは報酬の期待累積値が最大化されるような「方策」を学習していく．これにより，学習のゴールに対応する報酬関数を与えることで，学習の途中過程を考慮せずとも，エージェントは適切な方策を獲得することが狙える．

強化学習を用いた非競争的協力型ゲーム AI の代表的な方法には，Self-Play（SP），

模倣学習を用いるもの [25] がある．SP は，人間のデータ用いずに，同一の方策のエージェント同士で，繰り返しプレイしながら強化学習を行う方法である．この方法では，例えばゲームの主目的の達成度に応じた報酬をエージェントに与えることで，エージェントは試行錯誤を繰り返しながら，報酬が高くなるような方策を学習していく．高い水準で主目的を達成する上で，複数プレイヤ間の協力が求められるゲームでは，これにより，エージェント間で協力するような方策が発生するようになる．

SP のように人間のプレイデータを必要としない方法に対し，より実際の人間に近いプレイヤへの適応を重視している方法 [25] もある．強化学習では，訓練中に人間とリアルタイムに相互作用を行うには膨大なコストを要するため，それらの方法では，代わりに人間プレイデータを模倣学習したエージェントと協力相手とすることが一般的である．テスト時の協力相手である人間に近しい経験を訓練中に得ることができるため，十分なデータを利用できる場合であれば，それらも有力な方法の候補となる．

一方, SP で訓練したエージェントは学習中に自分以外との協力を経験していないため, 他の相手との連携が必ずしも期待できず, 模倣学習に関しても, 十分なデータが得られる場合ばかりではない．このことを踏まえ, 近年, Zero-Shot Coordination（ZSC）と呼ばれる強化学習の問題設定が注目されている [27]．ZSC では，訓練時にテスト時（本番）の協力相手の方策に関する情報を知ることはできず，テスト時に初対面の相手と協調し，高い累積報酬を獲得することを目指す．このような問題設定は，非競争的協力型ゲームでもしばしば発生する．

SP では自分と同じ相手としか協力する方法を訓練していないので，ZSC のように多様な未知の協力相手が割り当てられる環境ではまともな対応ができることは少ないとされる [27]．また，問題設定から外れてテスト時に人間とのプレイを試みる場合でも，固定の相手との訓練のみを経験しているため，一般に適切な協調を行うことは困難となる．模倣学習を用いる場合についても，ZSC の設定下では協力相手そのもの，または「近い」相手のデータが十分利用できることは想定しておらず，有力な選択肢にはならないと考える．

ZSC に対する主要なアプローチの 1 つには, 次節で取り上げる「Population-based Training（PBT） [28]」と呼ばれる方法の枠組みがある．本研究で提案する手法も，この PBT に属するものである．

3.2 Population-based Training

Population-based Training（PBT） [28] は，ZSC に対する主要なアプローチの 1 つである．PBT では，テスト時に人間とプレイさせるエージェント（以降，本番用エージェント）と，本番用 AI を訓練するためのエージェントのプール（訓練用 population）とを独立して学習させる．順序としては，まず先に人間のプレイデー

タを用いない方法で複数のエージェントを訓練し，訓練用 population を構成する．次に，そこからサンプリングしたエージェントと本番用エージェントを一緒にプレイさせ，本番用エージェントの方策のみを訓練する．これにより，様々な相手とプレイした際に，期待累積報酬を最大化する本番用エージェントの方策が得られることが期待される．このとき，Population 内の方策が多様であるほど，テスト時の協力相手に類似した方策とのプレイ経験が発生しやすくなるため，Population の多様性が本番用エージェントの性能の要となる．

PBT の最も単純な方法（以降，ナイーブな PBT）は，様々な乱数のシード値を設定しながら各エージェントを SP で訓練するというものである．エージェントは確率的に行動を選択するため，シード値以外のパラメータを固定したとしても，ある程度ばらつきのある方策が得られる．

その改良として PBT 研究の初期に考案された手法に,「Trajectory Diversity (TrajeDi) [30]」,「Fictitious Co-Play（FCP） [29]」がある．TrajeDi は，エージェントの行動・状態・報酬の履歴（Trajectory）の分布を方策の表現とみなし，Population 内で Trajectory 分布間の Jensen-shannon divergence(JSD) を最大化されるように，学習を誘導する方法である．これにより，Population 内の Trajectory，ひいては方策間の類似度が小さくなる圧力が加わり，多様性が増加する方向に学習が進むことになる．ただし，Trajectory 分布のサンプリングおよび JSD の計算には膨大なリソースが必要であり，TrajeDi は計算量の面で課題が存在していた．一方の FCP は，学習途中の方策も Population に含めることで，計算量を大きく増やさずとも，Population の多様性を増加させる方法である．状態数が大きくないゲームにおいて，学習途中の SP 方策は，人間の（主目的に対する）習熟度が高くないプレイヤと類似した方策に類似することが示唆されている．シード値を変更しながら収束するまで訓練した SP 方策に加え，それらの学習途中の方策を Population に含めることで，FCP は実際の人間とプレイした際に，ナイーブな PBT と比較して高い主目的報酬および主観的な満足度を達成した．その一方，FCP は TrajeDi のように，Population 内の他の方策と差別化するような誘導はなされていないため，重複した方策の発生を抑えるような工夫は施されていない．

この両者の特徴を併せ，改良を施した手法が「Maximum Entropy Population-based Training(MEP) [31]」である．MEP では，FCP のように訓練途中の方策を採用するのに加え，主目的報酬と共に TrajeDi よりも計算が単純である方策間のエントロピーを最大させることで，多様な Population の訓練を行う．方策間のエントロピーは，エージェントが訓練中に遭遇する状態に対し，Population の各エージェントがそれぞれ異なる行動を返すほど大きくなる．その計算は，TrajeDi と比較して非常に高速であり，MEP は長らく PBT で最も高い性能（主目的報酬，人間の主観評価）を示す手法として知られている．MEP の派生手法は多岐に渡って研究されており，複合的な計算量の削減により高速化を目指した E3T，方策のアンサンブルを用いることで少ない計算量でより多様な Population を生成する PECAN 等の手法が提案されている．

他方，高速化ではなく，MEP では Population に出現しないような方策に注目した研究もある．代表的なものには，「Hidden Utility Self-Play（HSP）[32]」と呼ばれる手法がある．MEP は主目的報酬および方策間のエントロピー（≒多様性）の二軸で Population を訓練するため，主目的報酬と関連の薄い挙動（人間のバイアス，好み，副目的等）が現れにくい．これに対し，HSP では，必ずしも主目的と関係しない副目的を含む「効用」を Population 内の各エージェントに割り当て，その効用に対応する報酬関数を訓練に用いることで，それらの挙動を含む Population の作成を図った．（HSP は本研究との関係が深い手法であるため，次節でその詳細を説明する）他にも，直近で提案された手法に「COLE [34]」「Role Play [33]」等がある．

3.3 Hidden Utility Self-Play

3.2 節で述べた PBT 手法の多くは，主目的達成を目指しながらかつ多様であるようなエージェントを Population として作成しようとしたものである．これに対し，Hidden Utility Self-Play（HSP） [32] は，主目的に必ずしも関係しない，様々な効用の Population を訓練する手法であり，本研究はこれを提案手法の土台として取り上げる．

SP におけるエージェントの訓練では，次の目的関数 J を最大化させるような方策 π を求める．

$$J(\pi) = \sum_t \mathbb{E}_{(s_t,a_t)\sim\pi}[R(s_t, a_t)]$$

- π：エージェントの方策
- t：時刻 t
- $s_t \in S$：t における状態
- $a_t \in A$：t におけるエージェントの行動
- R: 報酬関数

これにより，R で規定されるような報酬の期待累積値を最大化させるような方策が得られる．

対して，プレイヤの 2 人の非競争的協力型ゲームを対象とする HSP における Population の訓練では，次の 2 つの目的関数を同時に最大化させることを試みる．

$$
\begin{aligned}
J(\pi_a, \pi_\omega | R_m) &= \sum_t \mathbb{E}_{(s_t, a_t) \sim \pi}[R_m(s_t, a_t)] \\
J(\pi_a, \pi_\omega | R_\omega) &= \sum_t \mathbb{E}_{(s_t, a_t) \sim \pi}[R_\omega(s_t, a_t)]
\end{aligned}
\tag{3.1}
$$

- R_m: 主目的報酬関数
- R_ω: エージェントの効用に対する隠れ報酬関数
- π_a : R_m の下での方策
- π_ω : R_ω の下での方策

ここで，R_ω の空間として，隠れ報酬空間 $\mathcal{R}$ を考える．R_ω をある人間の効用と見立てたとき，一般に $\mathcal{R}$ はあらゆる人間の効用を内包する表現となるため，扱うのが困難なほどに広大である．HSP では，人間の効用がイベントを起点としている場合が多いことに着目し，$\mathcal{R}$ を線形関数として定式化している．

$$
\mathcal{R} = \{R_\omega : R_\omega(s, a_1, a_2) = \phi(s, a_1, a_2)^T \omega, ||\omega||_\infty \leq C_{\max}\}
$$

- $\phi : \mathcal{S} \times \mathcal{A} \times \mathcal{A}$: 状態 s で 2 人のエージェントの結合行動 (a_1, a_2) を取ったときに発生するゲーム内イベントを規定する空間
- $C_{\max}$: 効用の重み ω の境界

例として，Overcooked 環境で 4 つのイベント「玉ねぎを山から拾う」「上下左右いずれかに移動」「食材を鍋に入れる」「料理を提出する」からなる ϕ を考える．このとき，ω の次元数は 4（ω の各成分は各イベントに対応）となり，例えば，$\omega : (\omega_1, \omega_2, \omega_3, \omega_4) = (-8, 1, 0, 15)$ は，「玉ねぎ料理を嫌い，可能な限り多く移動しながら料理を作成・提出する」ような好みを意味する．人間には，多様な好みがあることが想定できるので，ω_i に様々な値を割り当て，訓練相手エージェントを作成することを考える．表 3.1 は，ω_1 から ω_4 に割り当てる値の集合の例であり，各エージェントごとに値をランダムに決定していくことで ω をサンプリングし，この好みに沿った方策 pi_ω を訓練していく．

最終的に，Population には様々な R_ω で訓練された複数の π_ω が含まれることになる．一方, π_ω とともに訓練された π_a は, 一般的には Population に含めない．（ただし，本番用エージェントの主目的に対する累積期待報酬を最大化させる問題設定においては，主目的のみを最大化させる方策のエージェントを含める方が高い性能を発揮することが知られている）これにより，HSP 以前の手法では学習されないような方策を，HSP では Population に導入することが可能となった．

イベント	ω が取りうる値の集合
玉ねぎを山から拾う	$\{-8, 0, 8\}$
上下左右いずれかに移動	$\{-1, 0, 1\}$
食材を鍋に入れる	$\{0\}$
料理を提出する	$\{-15, 0, 15\}$

表 3.1: Overcooked 環境における報酬空間の例

Population を構成した後，HSP では，他の PBT 手法と同様に本番用エージェントの訓練を行う．本番用エージェントの方策を π_A としたとき，ここでは次の目的関数を最大化させるような π_A を訓練する．

$$\begin{aligned} J(\pi_A^{'}) &= \mathbb{E}_{R_\omega \sim \hat{P_R}}[J(\pi_A^{'}, \widetilde{\pi}_\omega(R_\omega))] \\ &= \sum_t \mathbb{E}_{(s_t,a_t)\sim\pi_A^{'}, R_\omega \sim \hat{P_R}}[R_m(s_t, a_t)|\widetilde{\pi}_\omega(R_\omega)] \end{aligned} \tag{3.2}$$

- $\pi_A^{'} \in \Pi$：任意の方策
- $\hat{P_R}$：Population 内の R_ω の分布
- $\widetilde{\pi}_\omega$：Population 内の R_ω のもとで学習したエージェント

この式は，Population が人間の持つ隠れ効用を模した $\mathcal{R}$ の分布に沿うようにサンプルされた報酬関数 R_ω から作られ，本番用エージェントは，Population 全員とプレイしたときに合計主目的報酬が最も高くなるように学習することを意味している．

ここで，実装上は，主目的に対する報酬関数 R_a を報酬関数として与えたもと，各エピソードごとに Population（エージェント $i \in [N]$）からランダムに π_ω^i をサンプリングし，π_ω^i を π_A の相手プレイヤとさせながら，π_A を更新していく処理を行っている．その際，相手としてサンプリングした π_ω^i は固定し，更新しない．これにより，主目的と関係しない副目的を有する相手に対しても，主目的に向けて片方向の協調を行うような本番用エージェントが訓練される．

第4章　提案手法1：接待HSP

協力型ゲームにおいて，人間プレイヤを楽しませようとするAIを設計する際，主目的を高い基準で達成できれば，それで全てが解決されるというわけではない．人間を活躍させたり，人間の副目的を満足させたりするような協調が求められる場合も存在する．既存手法であるHSP（以降，原始HSP）では，人間の副目的を考慮した訓練用Populationを用いるものの，本番用エージェントは，それらのエージェントの副目的に対する報酬を向上させるような学習は行わない．我々は，これだけでは人間を満足させることが難しいと考え，「接待HSP」を提案する．接待HSPでは，副目的を考慮した訓練相手エージェントと同じ報酬関数を用いて本番用AI（エージェント）も学習させ，その相手を満足させるような方策の獲得を目指す．本章では，まず原始HSPの課題とそれに対する接待HSPの提案，期待される接待HSPの優位性について述べる．その後，接待HSPの有効性を検証するための実験とその考察を行う．

4.1　原始HSPの課題と接待HSP

原始HSPでは，報酬関数R_ωはその訓練相手エージェントのみが知っており，本番用エージェントとの訓練時には主目的報酬のみを最大化させる設定となっていた．一方，実際の人間には，主目的に留まらず，副目的を含むその人間プレイヤの「効用」全般の累積報酬を高めるような協調を期待する場合が存在する．例えば，図4.1のようなOvercooked環境のステージを考える．このステージは，図のように移動できる範囲がプレイヤ間で分断されており，料理を作るには，左のプレイヤ（緑の帽子）が中央のカウンターを介して右のプレイヤ（青の帽子）に食材を供給する必要がある．

主目的であるゲームのスコアを最大化させる戦略は，左のプレイヤ（緑の帽子）が食材を右のプレイヤに供給しながら，右のプレイヤが「玉ねぎ3つ」の料理を作り続ける，というものである．そのため，訓練中の原始HSPの本番用エージェントを左のプレイヤとしたとき，本番用エージェントは，どのような相手プレイヤであっても，玉ねぎを供給し続ける方策に収束してしまうはずである．一方，右のプレイヤが人間であるとしたとき，主目的であるスコアに加え，「トマト料理を作りたい」という副目的を持っているかもしれない．その場合，食材をトマトに偏って供給することがより人間プレイヤの満足させる方策となるが，先述のよう

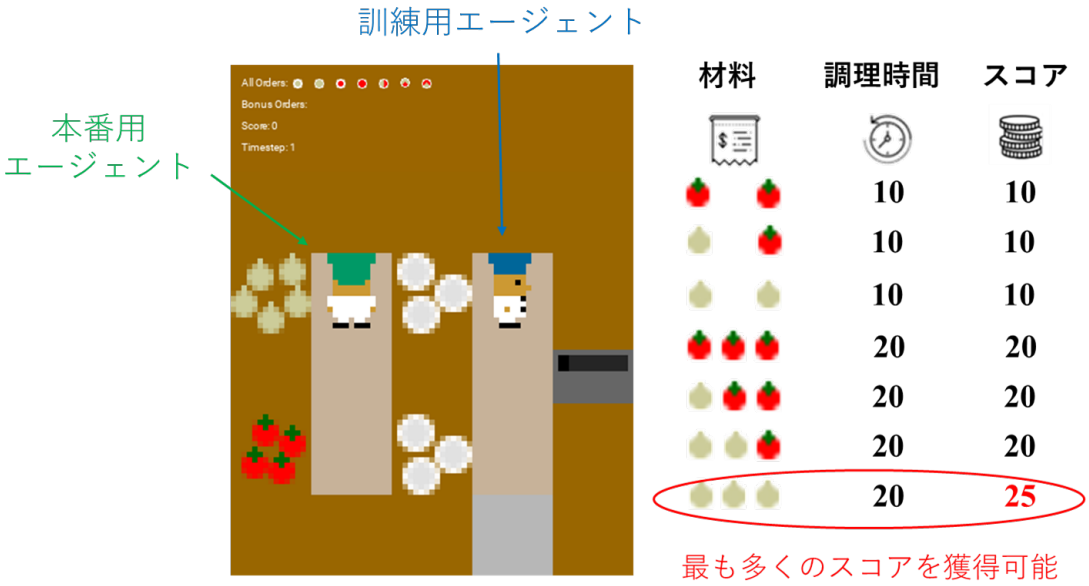

図 4.1: Overcooked 環境のステージ 1

に原始 HSP の本番用エージェントでは，一般にそのような方策を得ることはできない．

本研究では，このような場合に相手の副目的も含めて満足させられるよう，原始 HSP の設定を拡張した手法である「接待 HSP」を提案する．接待 HSP における本番用エージェントの学習では，原始 HSP の場合とは異なり，訓練相手エージェント j が学習する際に用いた R^j_ω を報酬関数として与える．これにより，例えば，上の例では玉ねぎを供給する場合よりも，トマトを供給する場合の方が報酬が大きくなり，本番用エージェントはトマトのみを供給するような方策を学習する．これにより，相手となる人間がトマト料理を作りたい人間ばかりであったり，今対面している相手がそうであると区別できるのであれば，その人間を満足させることができるだろう．なお，一般に相手の効用を知ることはできないが，AI 相手の訓練時に限り，「本番用エージェントが訓練相手エージェントの効用を知ることができる」という設定は導入可能であり，また既存手法との比較を行う際の公平性を損なうものでもない．

ただし，この例では，トマト料理を作りたい相手のみへの協調を想定している．一般に，接待 HSP を含む Population-based Training の手法では，様々な特性の訓練相手エージェントから構成される Population を用いて，本番用エージェントが訓練される．トマトのみを供給する方策を学習するのは，Population の中で，トマト好きが相対的に多い場合のみである．仮に，トマト好きと玉ねぎ好きが Population に混在している場合，例のようなステージでは，好きではない食材を渡されたとしても，どちらのエージェントも料理を作り続けるため，外見上の区別はつかないと考えられる．そのため，本番用エージェントは，相手によらず，トマトと玉

ねぎのいずれかを提供するような方策に収束すると予想している．

また，原始 HSP のもう 1 つの特徴として，Population の訓練時に，主目的報酬を追求した方策 π_a のエージェントと，R^i_ω を追求した方策 π^i_ω のエージェントを同時に訓練していた点がある．π_a のエージェントは，主目的に関係するようなプレイヤ間の協調行動に対しては報酬が与えられるものの，それ以外の副目的についての協調にはそれが与えられない．結果として，訓練用 Population を構成する R^i_ω は，その副目的のうち，単独で実現できるものだけを学習することになる．本番用エージェントは，そのような訓練相手への対応のみを学習するので，一般に両プレイヤ間での協調が求められるような副目的に対する最善の方策を獲得することが困難となる．

接待 HSP では，この点に対処するため，R^i_ω の方策同士で訓練を行い，Population を構成する．このとき，R^i_ω のプレイヤ 2 人で，1 エピソードにつき片方のみを更新していく方式と，R^i_ω と $R^{i\,'}_\omega$ の 2 つの方策を各プレイヤに割り当て，同時に更新していく方式が考えられる．両方式を比較した上で採用する方式を決定することが理想であるが，本研究では，計算リソースの観点から前者の方式を採用するものとした．

4.2 原始 HSP に対して接待 HSP が優位性を示す問題

4.1 節では，原始 HSP の本番エージェントが「トマト料理を作りたい T」を満足させられない例を挙げたが，他にも原始 HSP に対して接待 HSP が優位性を示す場合がある．本節では，そのような場合を 2 つ例示する．

4.2.1 異なる効用を有する 2 体の訓練相手 AI への接待

図 4.2 のステージでは，原始 HSP の本番用エージェントは対応できず，接待 HSP の本番用エージェントが対応できるような状況が発生する場合がある．このステージでは，図 4.1 のステージと異なり，左のプレイヤが右に食材を渡すとき，中央ではなく上下のカウンターに置くことが求められ，右のプレイヤは上下の提出口を利用できるようになっている．

ここで，右プレイヤを「玉ねぎが好きなエージェント O」と「トマトが好きなエージェント T」からなる Population からランダムにサンプリングしてきた訓練相手エージェントであるとする．O は「スコアの 1 倍」，T は「スコアの 1 倍」「トマトを鍋に入れると+5」「トマトを含む料理を提出したら+10」の報酬関数に紐づけられた訓練相手エージェントであるとする．このステージで右プレイヤである O の報酬を最大化させるためには，左プレイヤは玉ねぎを山からとって上側のカウンタに置き，右プレイヤはそれを受け取って玉ねぎ 3 個の料理を調理した後に上の提出口へ提出（獲得報酬： 40 のスコア$\times 1 = 40$）する工程を続ければ良い．

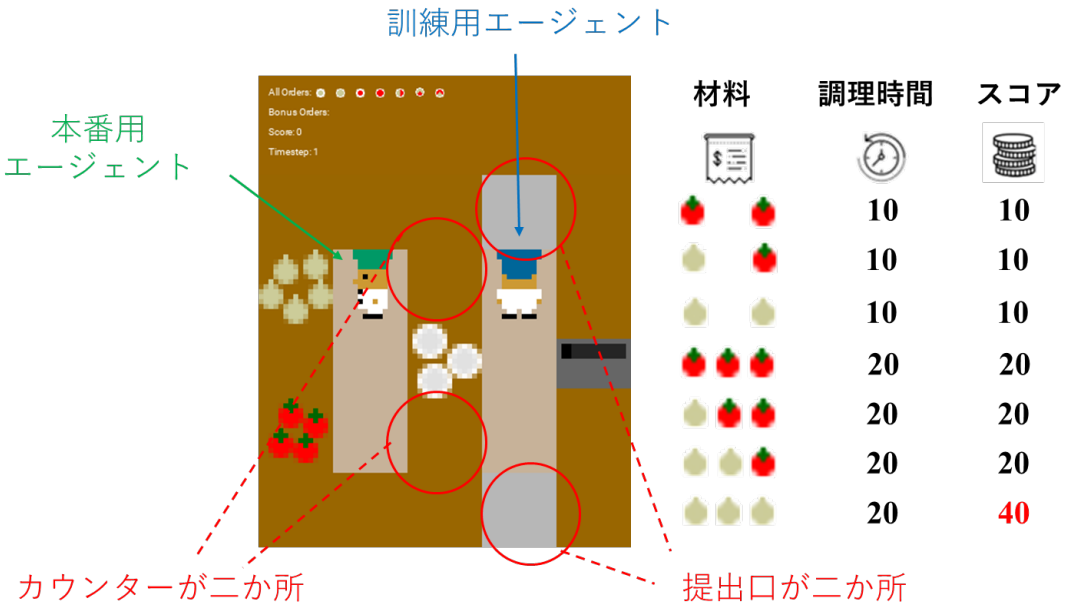

図 4.2: Overcooked 環境のステージ 2

逆に右プレイヤである T の場合，左プレイヤはトマトを山からとって下側のカウンタに置き，右プレイヤはそれを受け取ってトマト 2 個の料理を作成して提出（獲得報酬：10 のスコア$\times 1 + 5 \times 2 + 10 = 40$）する工程を続ける方が高い報酬を得られる（食材 2 個の料理のほうが調理時間が短く，1 エピソード内で提出できる総数が多いため）．

O は O と，T は T との協力の中で訓練された場合，O のチームは 30 ではなく 40 の報酬を得るため，上のカウンタを通して玉ねぎを受け渡し料理する．T のチームは 40 ではなく 50 の報酬を得るため，下のカウンタを通してトマトを受け渡し料理する．つまり，O と T の方策には，左プレイヤである場合は当然ながら，右プレイヤである場合にも，「どこで待つか」という外見上の差異が存在する．外見上の差異が存在するのであれば，接待 HSP の本番用エージェントは，相手である O や T の好みに合わせた挙動，すなわち O には上のカウンタに玉ねぎを，T には下のカウンタにトマトを置く挙動を学習できると期待できる．

4.1 節で述べたように，外見上の差異が存在するのであれば，接待 HSP の本番用エージェントは，相手である O や T の好みに合わせた挙動，すなわち O には上のカウンターに玉ねぎを，T には下のカウンターにトマトを置く挙動を学習できると期待できる．一方で，原始 HSP の本番用エージェントはこの場合，外見上の差異に関係なく，O に対しても T に対しても，上のカウンターに玉ねぎを置く挙動を学習してしまうだろう．なぜなら，原始 HSP の本番用エージェントは，相手の好みではなく，主目的（スコア）を最大化するように学習するためである．学習初期には玉ねぎやトマトを混ぜながら提供するかもしれないが，どちらにせよ

相手はそれを料理する[1]．そうすると，得点の高い玉ねぎを渡すことが良いということを徐々に学習してしまうのである．

一方，相手の効用に対する報酬を最大化させるには，O に対して上のカウンターに玉ねぎのみを，T に対しては下のカウンターにトマトを置くことが求められる．相手が T のときに，下のカウンターにトマトを置くイベントが発生した場合，接待 HSP の本番用エージェントは，上のカウンターに玉ねぎを置くよりも多くの報酬が得られる．先述したように O と T では，外見上の振る舞いが異なるため，本番用エージェントが両者を見分けることが可能であり，接待 HSP の本番用エージェントは相手に応じて戦略を切り替えるような方策を得ることが期待される．一般的に，Population-based Training では，様々な人間に対応可能な本番用エージェントを実現することを目的としているため，このように相手に応じて適切な戦略を切り替える能力は有益であるといえよう．

4.2.2 主目的と関係の薄い効用を有する訓練相手 AI への接待

4.2.1 項や 4.1 節の例は, 主目的を含むような効用の相手を想定している．一方, 人間は主目的とほぼ関係しない, 副目的のみの効用を有することが知られており [35], そのような相手を満足させるような状況も想定される．

例えば，Overcooked 環境では,「特定のオブジェクトを可能な限り多く，早く並べる」というような副目的が考えられる．図 4.3 のようなステージでは，皿を上限まで並べるには，左プレイヤが皿を中央のカウンターを介して右プレイヤに供給し，右プレイヤがその皿を空いているカウンターに置く必要がある．
当然ながら，このような協調に対して主目的報酬は与えられず，その一方で，皿を消費して料理を作成・提出した場合には与えられる．そのため，原始 HSP の本番用エージェントが右プレイヤとして割り当てられたときには，そのような訓練相手の意に逆行するような方策を学習することとなる．対して，接待 HSP の場合では，皿を並べていくことで報酬が得られるため，訓練相手の効用を満足させるような方策を学習することが可能となる．

また，この例では，先述した原始 HSP の Population の課題が顕在化されると予想する．原始 HSP の Population では，訓練時に片方のエージェントが主目的を追求するため，両プレイヤ間での協調が求められるような副目的に対する最善の方策を獲得することが困難となる特徴があると我々は考えている．この例では，右プレイヤに主目的を追求するエージェントが配置されたとき，左の皿を並べる副目的を追求するエージェントは，皿を可能な限り置く方策を学習する．しかし，主目的エージェントは料理を作る方策を学習するため，副目的エージェントは右側に皿がある状態を経験できない．逆に左に主目的エージェントが配置された場合，主目的エージェントが皿を偶然右に渡したとしても，副目的エージェントがそれ

[1] T は T 同士で訓練されるが，学習時の exploration により，たまには玉ねぎも提供されることがあるため，玉ねぎを渡されてもそれをしぶしぶ料理することができる

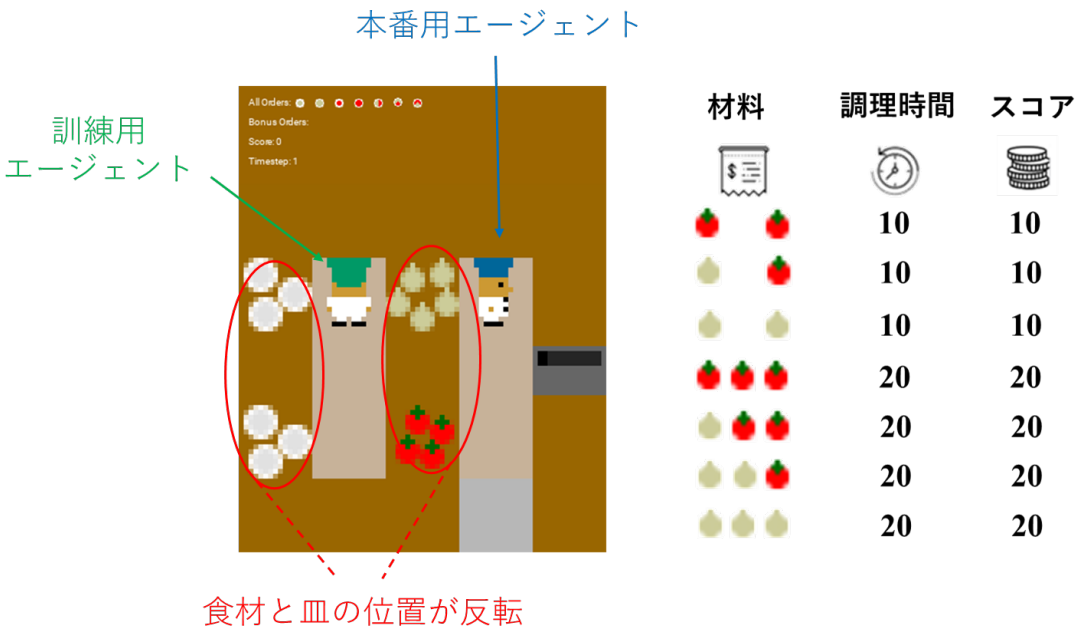

図 4.3: Overcooked 環境のステージ 3

に料理を盛り提出する確率は，皿を並べる確率よりも極めて低いため，主目的エージェントには報酬が与えられない．これは，主目的エージェントの方策がランダム方策から変化しないということであり，皿を置く行動が発生する確率は低いまま推移するため，効率の問題で右の副目的エージェントが皿を並べる方策を学習する機会も発生しにくい．したがって，原始 HSP の訓練相手エージェントは，「皿を可能な限り多く，早く並べる」副目的に対して，両プレイヤが協調するような方策を得ることは難しいと考える．

4.3 評価実験

4.2 節で述べた接待 HSP の原始 HSP に対する優位性を検証するための比較実験を行った．多くの既存の Population-based Training（PBT）の手法では，Population 内の方策の多様性や，コンピュータプレイヤおよび人間と本番用エージェントをプレイさせたときの主目的報酬を評価の指標としてきた．しかしそれらの指標は，本研究のように，主目的から外れた人間の副目的を重視して接待プレイを行おうとする目的を評価するにはそぐわない．そこで，副目的報酬を評価指標として用いることで，原始 HSP が相手の副目的報酬を高めることができない状況において，接待 HSP はそれを高めることができることを示す．

4.3.1 実験設定

本実験では，先述した例のうち，図 4.2 のステージにおける「玉ねぎが好きな O」と「トマトが好きな T」，図 4.3 のステージにおける「皿が置かれているのが好きな F」の 2 つの例について，実験を行った．それぞれの手法で予想される訓練相手エージェント，本番用エージェントの振る舞いは表 4.1 の通りである．

表 4.1: 期待される訓練相手エージェントおよび本番用エージェントの振る舞い

実験番号	手法	ステージ	訓練相手エージェント	本番用エージェント
1	原始HSP 接待HSP	ステージ2 左：本番用　右：訓練用	①：玉ねぎが好きなO ②：トマトが好きなT（不完全） ③：玉ねぎが好きなO ④：トマトが好きなT	A：相手が誰でも上の カウンターに玉ねぎを置く B：OとTで戦略を切り替える
2	原始HSP 接待HSP	ステージ3 左：訓練用　右：本番用	⑤：皿が置かれている 状態が好きなF（不完全） ⑥：皿が置かれている 状態が好きなF	C：皿を消費して料理をする D：皿を受け取って並べる

訓練相手エージェントたちはそれぞれ自身の効用を持ち，それを報酬関数として訓練する．そして接待 HSP の本番用エージェントもまた，接待相手となるそれぞれの訓練相手エージェントの持つ効用を報酬関数として訓練する．

一方，本実験ではそれに加え，Reward Shaping に従った報酬もエージェントに与えられる．Reward Shaping の詳細については，Appendix の B に記載している．

実験 1 で使用した訓練相手エージェントの報酬関数と Reward Shaping の一覧を表 4.2 に，実験 2 で使用したものを表 4.3 に示す．ここで，「皿が上限まで置かれている」は，ゲーム内の状態がこの条件を満たしているときに毎ステップ報酬を，「スコア」は，いずれかのエージェントが料理を提出したステップでのみ，追加されたスコアに表の係数をかけた報酬を与えるものである．その他の報酬は，そのエージェントが自らイベントを発生させたときに，イベントに対応する報酬が与えられる．接待 HSP の本番用エージェントの訓練時には，接待相手の訓練相手エージェント（方策は固定され，学習は行わない）視点で同様の処理が計算され，本番用エージェントの報酬として学習に使用される．

表 4.2: 実験 1 で訓練相手エージェントの学習に用いた報酬関数・Reward Shaping

	トマトを 山から取る	トマトを 鍋に入れる	料理の提出による スコア獲得	Reward Shaping
O	0	0	+スコア×1	I.
T	+5	+10	+スコア×1	

表 4.3: 実験 2 で訓練相手エージェントの学習に用いた報酬関数・Reward Shaping

	皿をカウンターに置く	皿をカウンターから取る	皿が上限まで置かれている	Reward Shaping
F	+1	-1	毎ステップ+20	II.

各実験での試行回数について，訓練相手エージェントは各設定ごとに 5 シード分訓練し，それら全てから構成される Population を本番用エージェントの訓練に用いた．本番用エージェントの学習は，所用時間の都合上，1 シードずつのみとしている．

実験の実装には，先行研究で公開された環境「ZSC-Eval」 [36] を使用した．この環境では，様々な PBT 手法がベースラインとして用意されており，エージェントの学習・評価を一貫して行うことができる．本実験では，その中の原始 HSP をもとに，接待 HSP のプログラムを開発した．

エージェントの学習は，RTX2080ti，RTX4080，RTX4090 のいずれかの GPU を搭載したマシンの上で行った．各種パラメータは，付録 A に記載するものを使用した．rtx4090 を用いた場合，学習が収束するまでの所要時間は，訓練相手エージェントは 1 人あたり 40 分〜1 時間，本番用エージェントは 5〜10 時間程度であった．

4.3.2 実験結果

実験 1：玉ねぎが好きな O とトマトが好きな T への接待

実験 1 の訓練相手エージェント「①：原始 HSP で訓練した玉ねぎ好きな O」「②：原始 HSP で訓練したトマト好きな T」「③：接待 HSP で訓練した玉ねぎ好きな O」および「④：接待 HSP で訓練したトマト好きな T」の右プレイヤ配置時における学習曲線を示す．この学習曲線は，訓練相手エージェントの学習時におけるエピソードごとの獲得累積報酬の推移である．原始 HSP は，主目的報酬で訓練されるエージェント（以下，主目的エージェント）と効用の報酬で訓練されるエージェント（効用エージェント）の組で，接待 HSP は，効用の報酬で訓練される同一のエージェント同士で訓練相手エージェントの学習を行う．そこで，各エピソードの累積報酬の値は，原始 HSP の場合は主目的エージェントと効用エージェントが受け取った報酬の総和，接待 HSP の場合は 2 体の効用エージェントの総和とした．①，③についての学習曲線を図 4.4 に，②，④についての学習曲線を図 4.5 に示す．

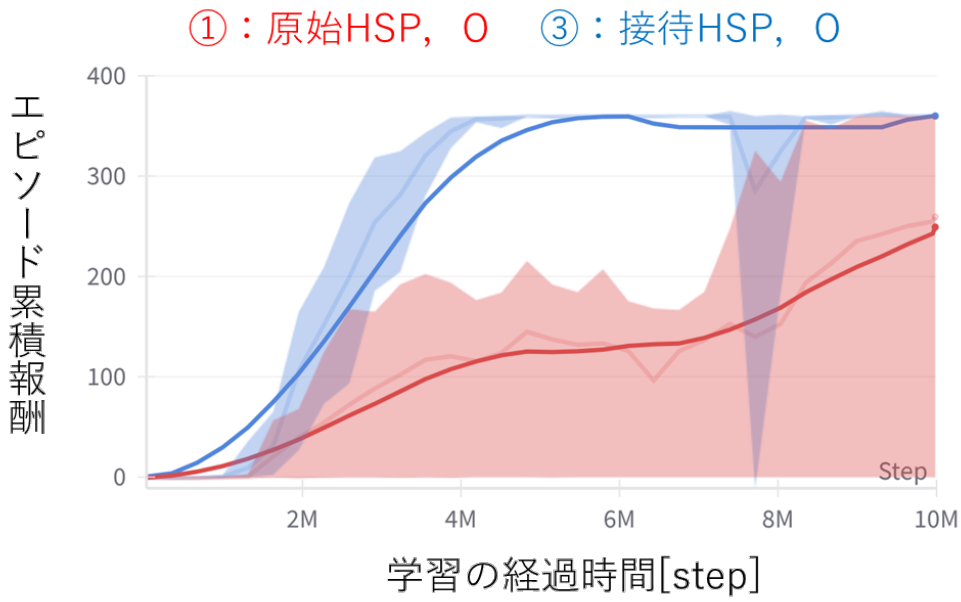

図 4.4: 訓練相手エージェント①と③の学習曲線

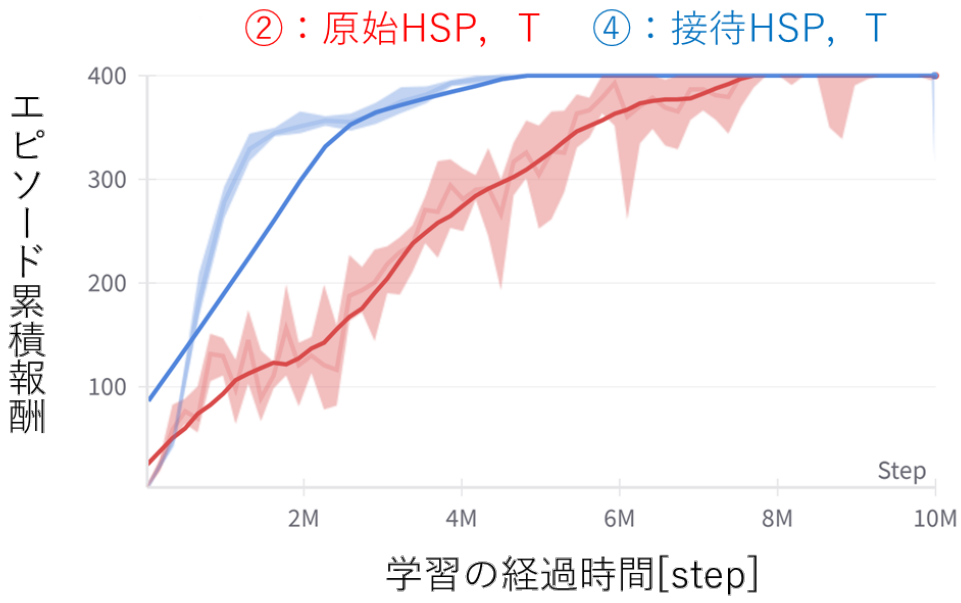

図 4.5: 訓練相手エージェント②と④の学習曲線

①と③の組，②，④の組のいずれにおいても，接待 HSP を用いる方が早期に収束し，最終的にも高い累積報酬を得るように収束することが確認された[2]．一方，

[2]①の学習の後半であっても累積報酬が 0 付近のエピソードが生じているのは，試行した 5 シード中 1 シードで終始報酬が 0 となる試行が発生したためである．

訓練後の①と②の主目的と副目的エージェントがプレイしている様子（左：主目的，右：副目的）を見てみると，②，④では見られなかった①では玉ねぎを時折下のカウンターを介して受け渡す振る舞いが，②ではトマト 3 つの料理を作り続ける振る舞い（主目的であるスコアと副目的の両方を追求）が確認された．これらは効用報酬を最大化させる上で最適な振る舞いではなく，それにより①，②は③，④に比べ低い報酬を得る結果となったと考える．

③，④が①，②に対して高い報酬を得たことは期待通りの結果であった一方，①，②の学習時の推移は事前の予想とは反する結果となった．第一に，①と③は予想ではほぼ同一の学習の推移を辿ると考えていたが，①と③の学習曲線の間には，小さくない差異が生じている．①と③は，ともにいずれのエージェントにおいても，料理を提出した際に得られるスコアのみを報酬としており，②と④のように各エージェントの配置を考慮せずとも，玉ねぎ 3 つの料理を提出（両方のエージェントに報酬+40）するような方策に収束するはずである．

第二に，②の学習ではトマト料理を作るような振る舞いを獲得できないと予想したにのに反し，現実にはトマト 3 個の料理を作成・提出しつ続けるような振る舞いを獲得している．②の学習では，左側が主目的エージェントである場合，右側は効用エージェント，左側が効用エージェントである場合，右側は主目的エージェントとなる．ここで，左側にいる主目的エージェントが玉ねぎを右に渡すとき，効用エージェントはトマトを受け取る場合よりも報酬は低くなるものの，玉ねぎ 3 個の料理を作成・提出することが最善の振る舞いとなり，主目的エージェントもそれにより最も多くの報酬を得るので，玉ねぎを置く振る舞いに収束すると考える．本実験の設定で 1 エピソードあたりに提出可能な玉ねぎ 3 個の料理の理論的な上限は 9 個であるので，右側の効用エージェントが得る報酬は最大で $40 \times 9 = 360$ である．逆に，主目的エージェントが右側にいる場合，効用エージェントはトマトを置くことで多く報酬が得るためにトマトを右に渡すので，主目的エージェントはその状況で最善な報酬が最大となるトマト 3 個の料理を作り続ける振る舞いに収束すると考える．このとき，1 エピソードあたりに提出できるトマト 3 個の料理の理論的な上限は 8 個であるので，右側の主目的エージェントが得る報酬は最大で $20 \times 8 = 160$ である．これらに対し，④の学習における右側の効用エージェントは最大で 1 エピソードあたりに 16 個のトマト 2 個の料理を提出可能であるため，得られる報酬の上限は $(10 \times 2 + 10) \times 16 = 480$ である．しかし，図 4.5 からわかるように，②の学習で右側のエージェントが獲得した報酬は予想した理論的な上限値を上回っており[3]，予想と現実で乖離が生じている．

これら 2 つの現象の要因として，プログラムの不備や②で左側の主目的エージェントが局所解に収束していまっていることが考えられるが，本研究では時間の都合上，要因を特定・解消するには至らなかった．これは今後優先的に取り組むべ

[3]図 4.5 の値には Reward Shaping が含まれているので，最終ステップ以外では先述した上限値を上回ることに問題はないが，今回の事例では②について曲線も単調増加かつ最終ステップ時点で上限値を超過してしまっている．

き課題となるであろう．

続いて，この Population を相手に訓練した本番用エージェントの結果を述べる．原始 HSP の本番用エージェントが学習に用いるのは，主目的に対する報酬に Reward Shaping を加えたものである．これは，本番用エージェントがどのような報酬を受け取って学習を行っているのかを分析する上での参考にはなるが，接待相手である訓練相手エージェントをどれだけ満足させたか，には必ずしも関係しない．一方，訓練相手エージェントの効用に対する累積報酬は，その訓練相手エージェントの効用をどれだけ満足させたかに相当する指標であり，各手法の本番用エージェントの評価指標としてより妥当であると考える．なお，本番用エージェントは報酬としては訓練相手エージェントの効用値を受け取っているが，自分の行動そのものは，外見上の入力のみで決定している．訓練相手として O と T はランダムにサンプリングされ入れ替わるので，訓練途中にそれぞれに適合した結果が評価されているわけではない．事実上，テスト用にいくつかのエピソードを実施して評価するのと同様の結果となっている．

本番用エージェント「A：原始 HSP」「B：接待 HSP」の学習時における訓練相手エージェントの効用に対する累積報酬の推移を図 4.6 に示す．

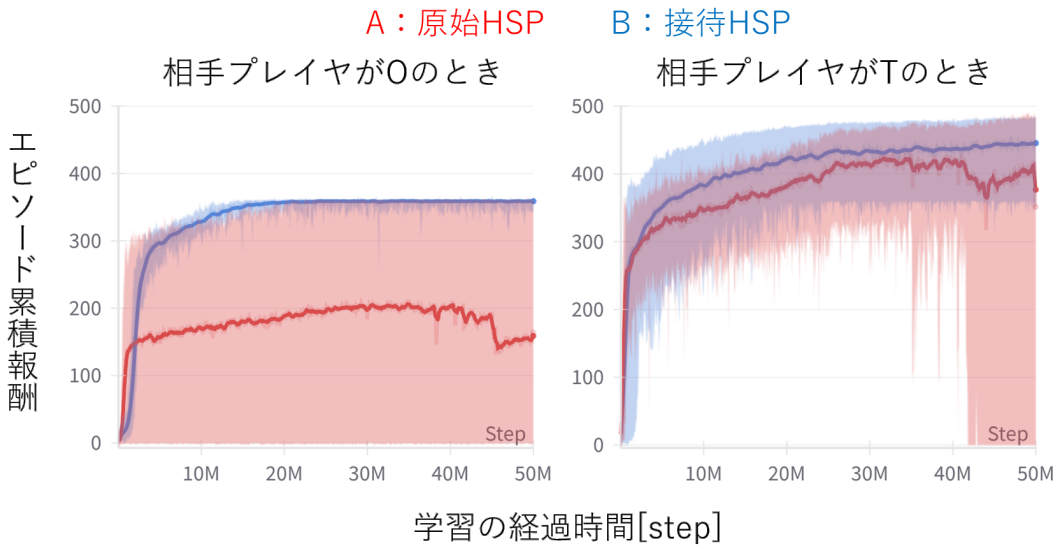

図 4.6: 本番用エージェントとプレイする訓練相手エージェントの累積効用値

相手エージェントが O の場合，学習が収束した本番用エージェントについて，B の方が優位な方策を獲得できていることがわかる．一方，これは期待していたように接待 HSP の本番用エージェントが訓練相手エージェントの効用報酬をもとに学習したことによる差ではなく，単に原始 HSP で訓練した O の性能が低かったことに起因するものである可能性が高い．4.2 では，原始 HSP の本番用エージェントは，相手が誰であっても最も多くスコアを得られる「玉ねぎを上のカウンターに置く」という方策に収束する一方，接待 HSP は T に対して「トマトを下のカウン

ターに置く」という戦略を取ったときに多くの報酬を得られ，相手に応じて戦略を切り替える方策を獲得すると予想した．しかし，図 4.6 の結果では A も T に対して B と同程度の報酬を獲得しており，訓練後の振る舞いにおいても相手が O のときには玉ねぎを，相手が T であるときにはトマトを置く様子が観測された．この実験で本番用エージェントが「玉ねぎを置いた回数」と「トマトを置いた回数」の学習中の推移を調べたところ，図 4.7，図 4.8 のようになっていた．

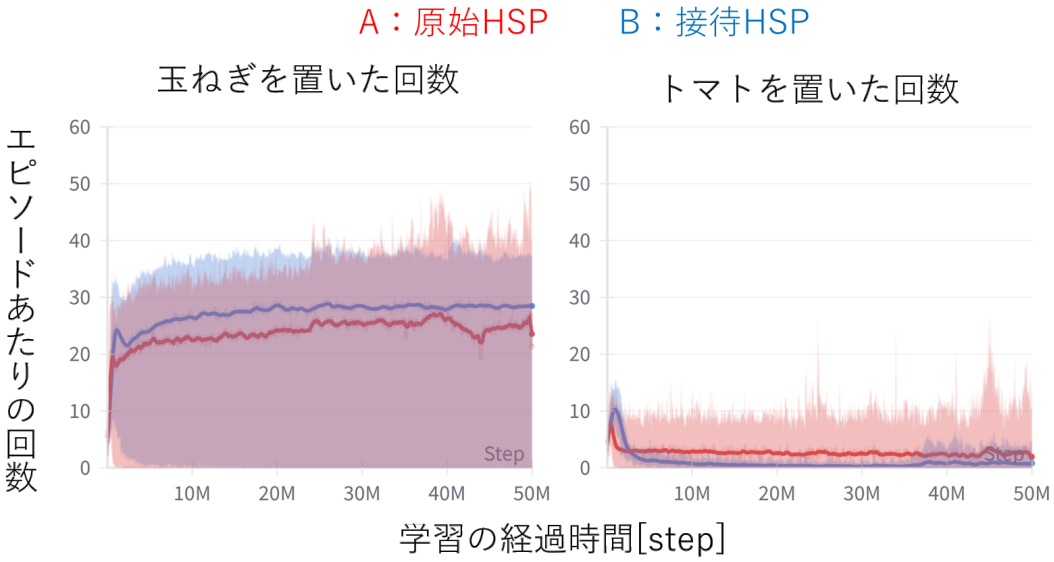

図 4.7: O とプレイする本番用エージェントが玉ねぎ・トマトを置いた回数

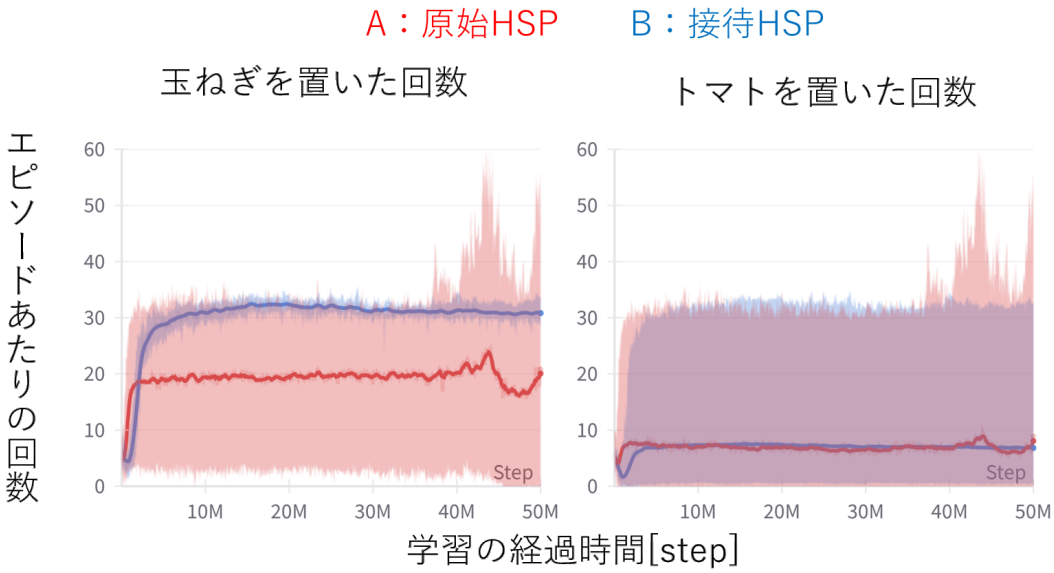

図 4.8: T とプレイする本番用エージェントが玉ねぎ・トマトを置いた回数

図 4.7 の左に示すように，両手法で最も顕著に差が生じているのは，相手が O のときに玉ねぎを置いた回数である．また，図 4.4 から，本実験で用いた原始 HSP の訓練相手エージェント O は，接待 HSP の O よりも獲得できる報酬が低くなっており，料理を作成・提出する能力が洗練されていないことが窺える．そのため，仮に本番用エージェントが玉ねぎを効率的に置いたとしても，エピソードあたりに料理に使用される玉ねぎの平均個数ひいては本番用エージェントが玉ねぎを置く回数も少なくなり，結果的に相手が O のときの A の累積報酬も低くなったと考察する．これを解消するには，先述した原始 HSP の訓練相手エージェントの学習における問題への対処，あるいは学習の総ステップ数，報酬設計の変更が有効となるであろう．

一方，原始 HSP の本番用エージェントが T に対してトマトを置く振る舞いを学習できた理由として，相手が T のときに玉ねぎを渡しても，T が玉ねぎ 3 個の料理を効率的に作成・提出できなかったためと考える．T はトマト料理の作成に関係するイベントが生じたときに報酬が与えられる 2 体のエージェント同士で学習を行うため，玉ねぎ料理を作成する経験を十分に得る前にトマト料理を作成・提出するような方策に収束する可能性が高い，そのような玉ねぎ 3 個の料理を作成することに不慣れな T に玉ねぎを渡すよりも，トマトを渡してトマト料理を作ってもらう方がエピソードあたりに高いスコアを得られる構造となったために，今回のような結果となったと考える．接待 HSP において本番用エージェントを相手の効用報酬をもとに学習させる有効性を確かめるには，実験の設計を変更するか，訓練相手エージェントに多様な経験を積ませるような工夫を施すことが求められるであろう．

実験 2：皿が並んでいる状態が好きな F への接待

訓練相手エージェント「⑤：原始 HSP によって訓練された F」と「⑥：接待 HSP によって訓練された F」の左プレイヤ配置時におけるエピソードごとの累積報酬の学習曲線は，図 4.9 のようになった．この問題では，両プレイヤが協調し，可能な限り短い時間で「カウンターに皿を上限まで置く」ことで，高い報酬が得られるように設計されている．4.2 節で予想したように，⑤は主目的を追求するエージェントと効用を有するエージェントの組で訓練が行われるため，主目的とは関係の薄い副目的への協調を十分に学習することができていない．一方の⑥は，両者が同一の効用のもと，図 4.10 のように協調的な戦略を獲得するに至っている．

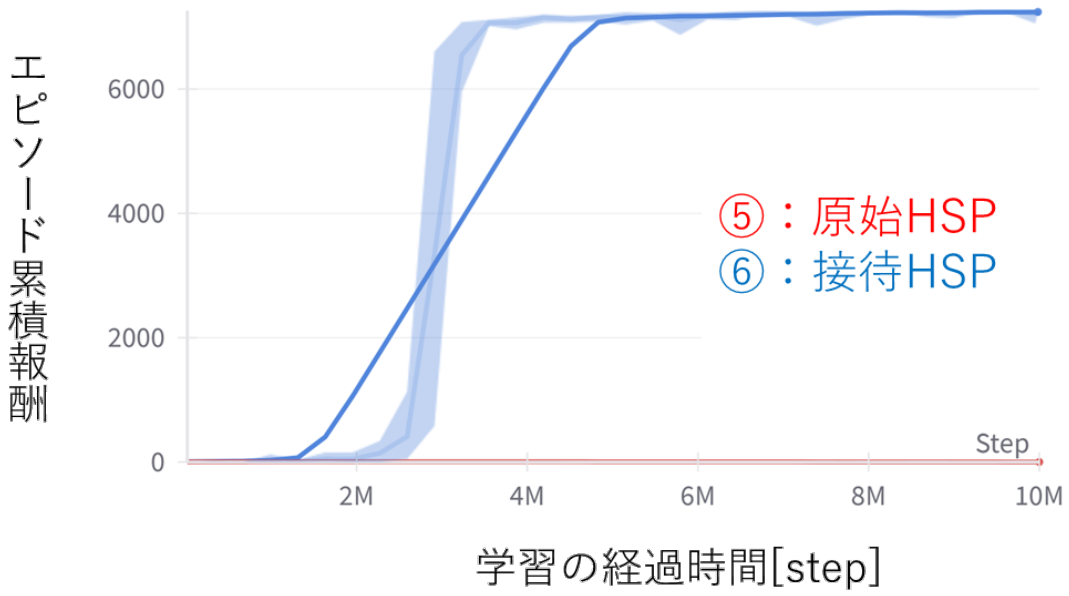

図 4.9: 訓練相手エージェント⑤と⑥の学習曲線

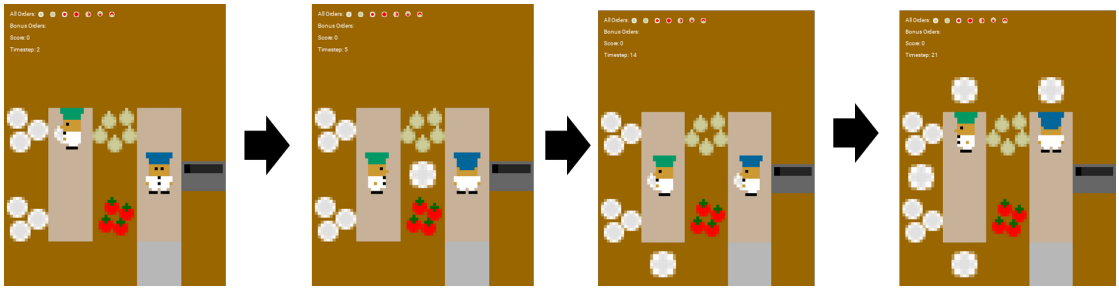

図 4.10: ステージ 3 における副目的に対する協調

続いて，この訓練相手エージェントを相手に訓練した本番用エージェントの結果を述べる．本番用エージェント「C：原始 HSP」「D：接待 HSP」の学習時における訓練相手エージェントの効用に対する累積報酬の推移を図 4.11 に示す．

こちらも訓練相手エージェントの学習と同様，接待 HSP が原始 HSP に優位性を示す結果となった．また，図 4.12 は本番用エージェントが料理を提出した回数の推移であり，原始 HSP のみが料理を提出する振る舞いを学習していっていることがわかる．原始 HSP の本番用エージェント C は，主目的のみを報酬としているため，皿を消費して料理を作り続けるような方策を獲得し，F を満足させることができなかった．これに対し，接待 HSP の本番用エージェント D は，相手の効用を満足させることで報酬を受け取るため，副目的に対する協調を学習することに成功したと結論する．

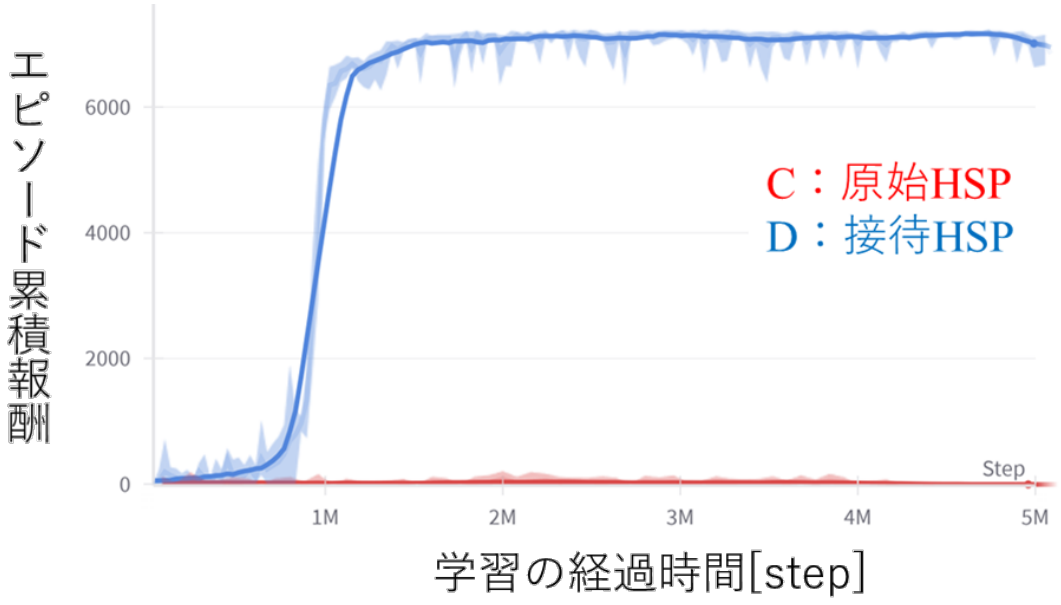

図 4.11: 訓練相手エージェント F の効用に対する累積報酬の推移

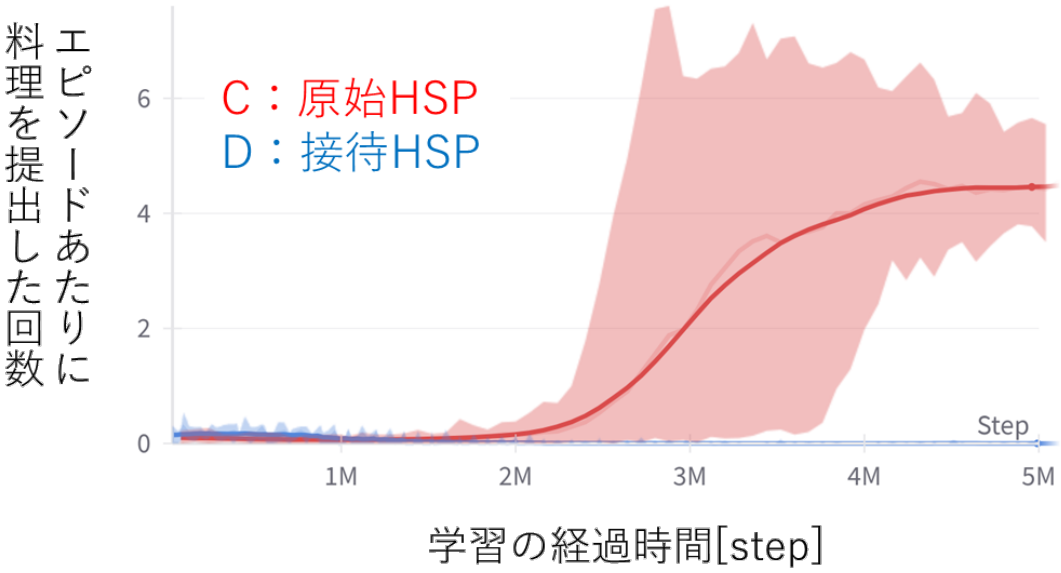

図 4.12: 本番用エージェントが料理を提出した回数の推移

第5章　提案手法2：不満忖度HSP

4章では，接待HSPが原始HSPに対して示すであろう優位性を述べ，その検証のための実験を行った．その優位性は，主にPopulationの訓練と本番用エージェントの訓練において，各々が持つ効用を重視した報酬を与える，という仕組みから来るものであると考えた．

しかし，ここで多様な副目的が効用で表現され得ることを考えた際，効用が等しくない，あるいは報酬を最大化させるための戦略が相反していたとしても，外見上の方策が類似している場合がある．そのような場合，エージェントへの接待を行おうにも，相手を見分けることができないため，それぞれを満足させるような戦略を取ることが難しい状況が発生する．

一方，これが人間同士であれば，仮に片方が気の利かない行動をとったとしても，もう片方がゲーム外の発話またはゲーム内の行動によって不満を表し，不満を受け取った側は自分の行動を改めることができる．そのような不満を表す行動をPopulationに獲得させ，それをシグナルとして本番用エージェントが相手の効用の識別に活用することができるようになれば，接待HSPでは見分けられないような人間の不満を表す行動に対して，より適切な協調が可能な本番用エージェントを実現できると我々は考えた．本章では，接待HSPをもとに，人間の不満を表す行動を導入することで発展させた手法である「不満忖度HSP」を提案する．

5.1　接待HSPの課題

接待HSPでは，様々な効用のエージェントからPopulationを構成し，本番用エージェントは，様々な訓練相手の効用に対する期待累積報酬を最大化させる方策を学習していく．結果として，本番用エージェントは，相手の振る舞いからどのような効用を有しているかを判断し，相手に応じて戦略を切り替えるような方策を獲得することが，4.3にて確認された．（実験結果次第で表現を変更します）

しかし，必ずしも訓練相手エージェントの外見上の方策のみで，その効用を識別できるとは限らない．例えば，図5.1のようなステージで，左プレイヤに本番用エージェント，右側にスコアを重視しながらも「玉ねぎ料理を作りたいO」または「トマト料理を作りたいT」を配置することを考える．（ただし，いずれのエージェントも接待HSPで訓練されたものとする）

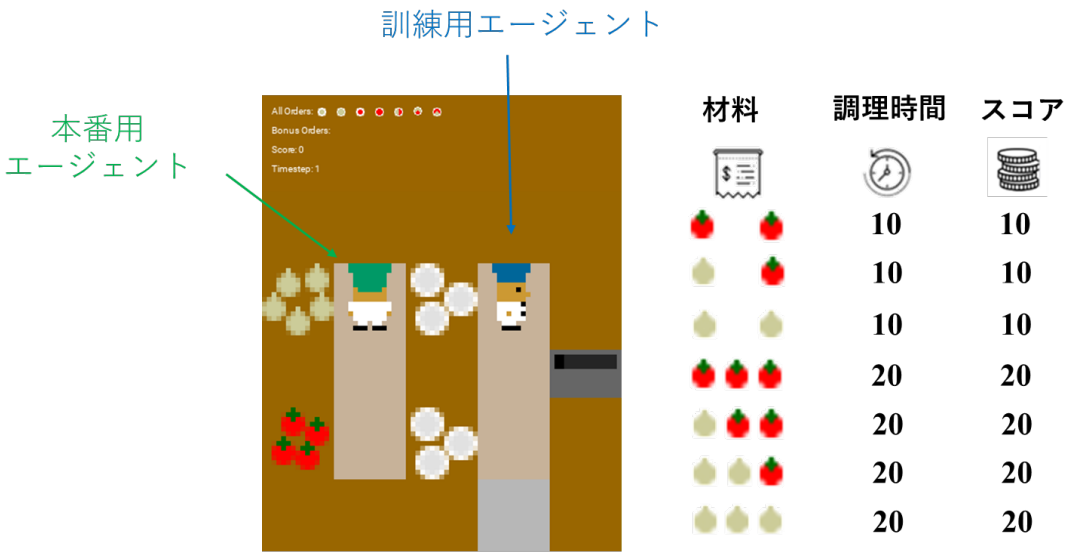

図 5.1: Overcooked 環境のステージ 4

このステージでは，料理のスコアが使用する食材によらないことから，任意の食材 3 個の料理を作成・提出することが，主目的に対する最適な戦略となる．

図 4.2 のようなステージであれば，T は玉ねぎを連続して渡されたとき，自分でトマトを取りに行くような振る舞いを見せたが，このステージの右プレイヤは，そのような行動を取ることができない．そのため，スコアのために不本意ながらも，T は玉ねぎを渡された O と同様，玉ねぎ料理を作るような振る舞いを見せると予想できる．

このように，外見上の方策が一致する効用の相手を，相手の行動の系列のみをもとに見分けるのは，非常に困難である．したがって，ゲームの状態・行動のみを観測できる本番用エージェントが，相手に応じた協調を取ることもまた，達成が難しい問題であるといえる．

5.2 不満忖度 HSP

5.2.1 人間のゲーム内伝達行動の導入

人間同士がプレイする場合，ゲーム内外で伝達行動を行うことで，お互いの認識を擦り合わせるような事象がしばしば見られる [35]．ゲーム外の発話によるコミュニケーションは当然ながら，それが不可能な状況においても，短期間に似たような動作を繰り返す，ステージ内のオブジェクトを意味のある形（文字，絵など）に変化させるといった，ゲーム内での伝達行為が行われることが知られている．ここでは，ゲーム内の状態や行動の系列を用いた伝達行動に注目する．

人間の伝達行動は，5.1 で挙げた例のような状況で，相手がどのような効用を有しているか見分けるのに役立つ．仮に右プレイヤがトマトが好きで，相手に積極的に反応を示すような人間であった場合，玉ねぎを何度か渡されたとき，困惑して立ち尽くしたり，料理をあえて作らず抗議の意を示したりすることが想定できる．これは，玉ねぎが好きな人間とは，外見上が異なる方策であり，接待 HSP の本番用エージェントは原理的に両者を見分けることが可能になる．

一方，接待 HSP は，既存の Population-based Training と同様に，Self-Play に準ずる方法で Population を訓練している．接待 HSP の方法で訓練されたエージェントは，ともに訓練される同じ相手と繰り返しプレイし，独善的で一貫性のある方策を得る．このような学習では，見知らぬ相手へ対処せずとも，少ない計算で効用に対する報酬を最大化させることできる．そのため，人間が行うような伝達行動が発生する可能性はかなり低いと考えることができる．

本研究では，接待 HSP をもとに，人間が示す反応の中でも「不満」に関するものを Population に導入した「不満忖度 HSP」を提案する．接待 HSP と比較したときの概念図は図 5.2 の通りである．

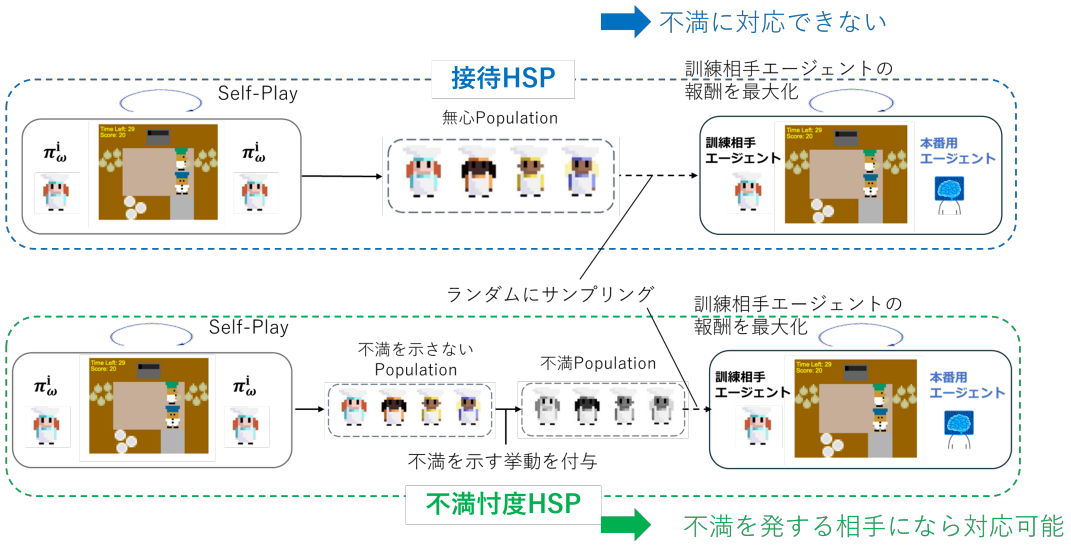

図 5.2: 接待 HSP と不満忖度 HSP の概念図

図 5.2 に示すように，接待 HSP と不満忖度 HSP の相違点は，本番用エージェント訓練時にサンプリングする Population のみである．接待 HSP と同様の方法で訓練した Population をもとに，追加で学習を行う，条件分岐で記述された行動のルールを追加するなどして，不満を示すような挙動を獲得した Population を作成する．これにより，本番用エージェントの学習に人間のように不満を示す行動を行うような相手を導入し，その行動から相手の効用を見分けられる場合が発生する．

5.2.2 不満を示す挙動

人間の反応的な挙動には，様々なものがある．ここでいう反応的な挙動とは，先述した不満に対する立ち尽くしや意図的なサボタージュのような挙動を指している．不満の他にも，相手が自分の希望通りに行動した際に，感謝を伝えるようなこともあれば，特に感情の起伏とは関係なく，相手の出方を見ながら，慎重に行動するというものもある．一方，本番用エージェントが受け取る不満に関係する反応は，他の反応と比較して，相手の効用への協調を試みる上でより重要な情報源であり，接待 HSP が対象としている問題設定との親和性が高いという特徴がある．そのため，本研究では，不満に関係する反応を取り上げ，Population に導入することを考える．

ここで，不満に関係する反応は，「効用に対する期待累積報酬が低くなる」ときに，発生するような行動列のパターンであるとする．例えば，5.1 の例で相手が玉ねぎを渡してきたとき，トマトが好きな T は，その試合（エピソード）で作成可能なトマト料理が少なくなったことに，不満を感じるだろう．これは，最終的な累積報酬が低くなる場合の一例であり，これに対して不満を示すような反応を起こすのは，人間で置き換えたときには自然な流れであると考える．強化学習の多くの手法では，期待累積報酬を最大化させることを目標としており，その推定器に当たる機能を備えているものが多い．不満に関連する反応をこのように形式化し，それに基づき手法を設計することで，本研究で得られた知見を様々な強化学習手法で応用することが可能になると考える．

ただし，本研究では，実験を簡素化するため，ステージや訓練相手エージェントの効用に応じて，手動で設計した条件分岐フィルターにより，不満の関係する反応を示すような形式の不満忖度 HSP のみを扱う．例えば，5.1 の例で T が玉ねぎを渡されたことに不満を示すような事例では，相手が「玉ねぎを山から取る」イベントを一定回数発生させたときに，不満を示すような実装を行う．この方式は，他の Population-based Training 手法のように，様々な事例に普遍的に適用が可能なものではないが，人間の不満に関する反応を導入することの有効性を検証する上では，十分なものであると考えている．

次に, このような条件を満たしたときに, 示すべき反応について議論する．Population-based Training の問題設定が初対面の人間との協調であることに立ち返ると，Population は，実際の人間に存在する特徴を網羅的に含むことが望ましい．そのため，理想的には，現実の人間の特徴を備えた上で多様な方策をモデリングする方法，具体的には，一般に人間に共通するヒューリスティックスや人間データをもとにした教師あり学習を用い，人間的な反応を再現することが好ましいといえる．一方，これらは実装コストの観点で取り扱うのが難しいため，本研究では，「一定時間，留まる・左右に動く」というような単純な機械的パターンにより，不満に関係する反応を実装することに留める．

5.3　評価実験

5.2 節で述べた接待 HSP の原始 HSP に対する優位性を検証するための比較実験を行った 4.3 の実験と同様，ここでは，接待 HSP が対応できないような状況を具体的に想定し，それらに不満忖度 HSP がどれだけ対応できるかを示すことで，評価を行っていく．

5.3.1　実験設定

本実験では，以下の図 5.3「皿が置かれている状態が好きな F」「自分で皿を置くことが好きな P」を識別する例について，実験を行う．4.2 で挙げた例では，「皿が置かれている状態が好きな F」への協調を題材としたが，この例ではそれに加えて「皿を自分で置きたい P」も Population に存在することを考える．

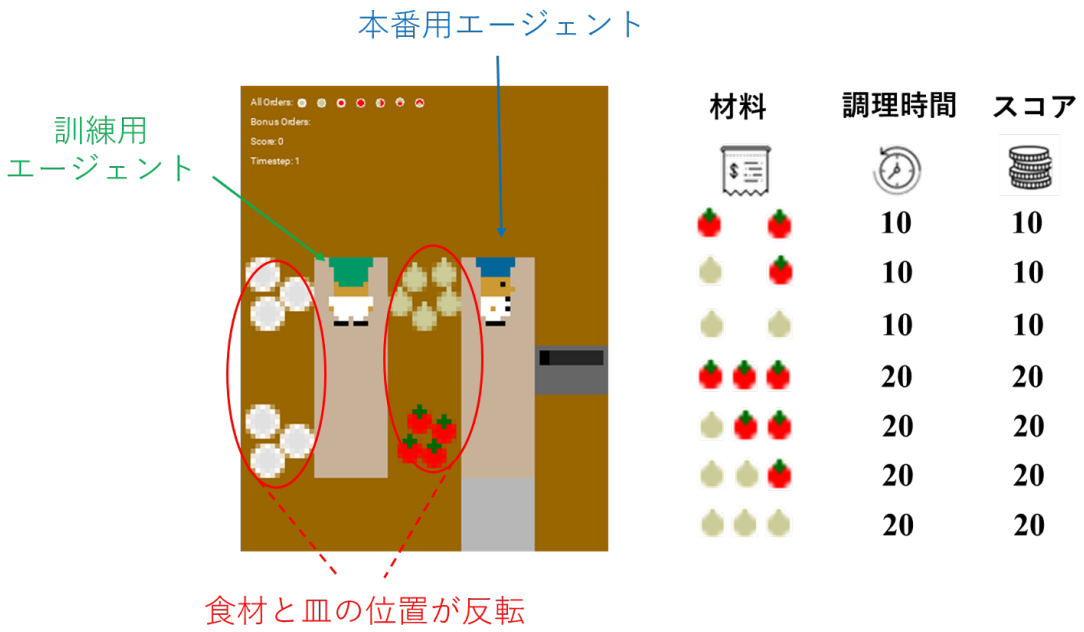

図 5.3: Overcooked 環境のステージ 3

左プレイヤが本番用エージェント，右プレイヤが訓練相手エージェントであるとき，このステージで P が自分でなるべく多くの皿を置くには，P が自分で直接置ける 4 枚に加え，本番用エージェントに中央のカウンターから皿を取ってもらう必要がある．一方で，Overcooked 環境では，料理が盛られた形で皿を提出することでしか皿をマップから排除する方法は存在しないため，P が置いた皿を本番用エージェントが料理に用いて提出し続けることが，最も P の効用に対する報酬を最大化させる戦略となる．これは，「皿が置かれている状態が好きな F」が右プ

レイヤに望む行動（皿を受け取り，右側 3 か所に置いてほしい）とは異なる．一方，P と F はどちらも外見上は皿を可能な限り置こうとするような方策を取る．

接待 HSP の本番用エージェントが P と F のそれぞれに適した協調を行うには，類似した両者を見分ける必要があるため，非常に難しいと予想する．一方の不満忖度 HSP では，P と F に不満を示すような挙動を加えた Population を相手に学習するので，仮に当初の行動が P や F にとって適切でなかったとしても，不満の意を伝えられれば対応を変えることができるような方策を獲得できると考える．

それぞれの手法で予想される訓練相手エージェント，本番用エージェントの振る舞いは表 5.1 の通りである．

表 5.1: 期待される訓練相手エージェントおよび本番用エージェントの振る舞い

手法	ステージ	訓練相手エージェント	本番用エージェント
接待HSP	ステージ3 左：訓練用　右：本番用	⑦：皿が置かれている状態が好きなF ⑧：皿を自分で置きたいP	E：皿を並べるべきか， 料理をするべきか当惑する
不満忖度HSP		⑨：皿が置かれている状態が好きなF+ ⑩：皿を自分で置きたいP+	G：F+とP+それぞれに最適化する

本実験では，4.3 と同様の Reward Shaping，マシンを使用する．各設定ごとに訓練するエージェントの数も同様であり，訓練相手エージェントは各設定につき 5 シード分訓練し，それらから構成される Population を用いて，本番用エージェントを 1 シードずつ訓練する．使用した各種パラメータは付録 A に記載する．

各訓練相手エージェントの報酬関数と Reward Shaping は表 5.2 の通りである．4.3 でも説明したように，「皿が上限まで置かれている」は，ゲーム内の状態がこの条件を満たしているときに毎ステップ報酬を与えるものである一方，その他の報酬はそのエージェントが自らイベントを発生させたときに，イベントに対応する報酬が与えられる．本番用エージェントの訓練時には，接待相手の訓練相手エージェント（方策は固定され，学習は行わない）視点で同様の処理が計算され，発生した報酬は本番用エージェントの学習に使用される．

表 5.2: 各訓練相手エージェントの報酬関数と Reward Shaping

	皿をカウンターに置く	皿をカウンターから取る	皿が上限まで置かれている	調理する	料理を提出	Reward Shaping
F	+1	-1	毎ステップ+20	0	0	I.
P	+400	-400	0	+1	+400	II.

ここで，報酬関数は，F と P が自身の効用に対して最適な戦略を取った場合に，それぞれで得られる累積報酬値の差が大きくならないように設計した．これは，あるエージェントの効用に対する報酬値が Population 内で突出して大きい場合，本

番用エージェントが他のエージェントへの協調を学習する動機が薄れ，報酬値の大きいエージェントに偏った協調を学習すると予想したためである．先行研究では，主目的のみを対象としていたため，このような問題が生じにくいが，様々な副目的への協調を目指す問題設定では，この点に留意する必要がある．

不満忖度 HSP 本番用エージェントの訓練時に，訓練相手エージェントが不満を示す条件および行動パターンは表 5.3 のとおりである．

表 5.3: 不満 population が不満を示す条件および行動パターン

訓練相手エージェント	不満を示す条件	不満の行動パターン
皿が置かれている状態が好きなF	過去10step以内に本番用エージェントが皿に料理を盛った	上下交互に移動
自分で皿を置くことが好きなP	過去10step以内に本番用エージェントが皿をカウンターに置いた	上下交互に移動

5.3.2 実験結果

本研究の実装における不満忖度 HSP の不満 Population は，接待 HSP と同様の設定で訓練された，不満を示さない Population をもとにしている．不満忖度 HSP の本番用エージェントの訓練時には，単純なルールベースを用いて不満を示さない Population に不満を付与するが，その基本的な方策は変化しない．そのため，ここでは，訓練相手エージェント「⑦：接待 HSP によって訓練された F」と「⑧：接待 HSP によって訓練された P」の訓練における結果のみを示す．⑦，⑧のエピソードごとの累積報酬の学習曲線は，図 5.4 および図 5.5 のようになった．

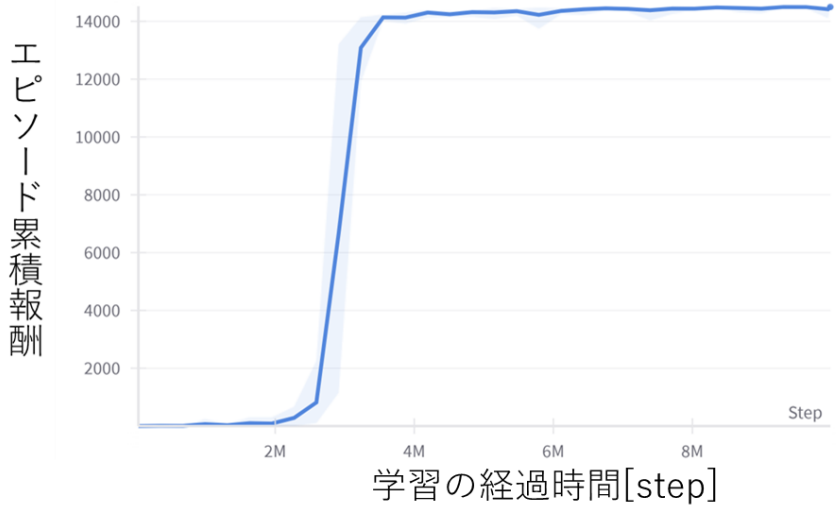

図 5.4: 訓練相手エージェント F の学習曲線

図 5.5: 訓練相手エージェント P の学習曲線

ここで訓練相手エージェント F は 4.3 の実験と同様の設定を使用しており，期待通りの結果となったため，特筆するべき点はない．一方の P は，左プレイヤのときに皿を中央のカウンターに置き，右プレイヤのときにはその皿を消費して料理を作り続けることで，最も多くの報酬を得ることができる．しかし，この実験で P は左プレイヤのときに皿を置くものの，右プレイヤのときには料理をしない方策となり，理想的な戦略を獲得することができなかった．

この理由として，P が料理に関係する行動を行ったときに与えられる正の報酬が，その過程で生じる負の報酬を受容するのに十分でなかったためと予想している．右プレイヤである P が料理した際に与えられる報酬は，カウンターから皿を取ったときの-400 と料理を提出することによる+400，「調理する」と Reward Shaping の I. による 2 桁程度の正の報酬のみである．料理を行うには，カウンターから皿を受け取った上で，調理済みの料理を鍋から取得し，提出口に運ぶ必要がある．このステージでは，複数の工程を踏んで料理を調理するよりも，皿をカウンターから取るイベントを先に経験を得る可能性が高い．そのため，エージェントは学習の初期に皿をカウンターから取る行動を避ける方策を学習し，報酬を最大化させるような最適な戦略を獲得できなかったと考える．しかしながら，P が本番用エージェントの訓練時に，左プレイヤとしてのみ利用されることを踏まえると，この方策でも期待した通りの振る舞いを行うと考え，ここで訓練した P と F から Population を構成することとした．

続いて，この Population を相手に訓練した本番用エージェントの結果を述べる．4.3 の原始 HSP・接待 HSP 間の比較と異なり，接待 HSP と不満忖度 HSP では，訓練相手エージェント視点での累積報酬に Reward Shaping を加えたものを学習用の報酬としている．そのため，ここでは，本番用エージェントが受け取る累積報酬の学習曲線を，性能の主な評価指標と考える．

本番用エージェント「E：接待 HSP」「G：不満忖度 HSP」の訓練に用いたエピソードごとの累積報酬の学習曲線を図 5.6 に示す．

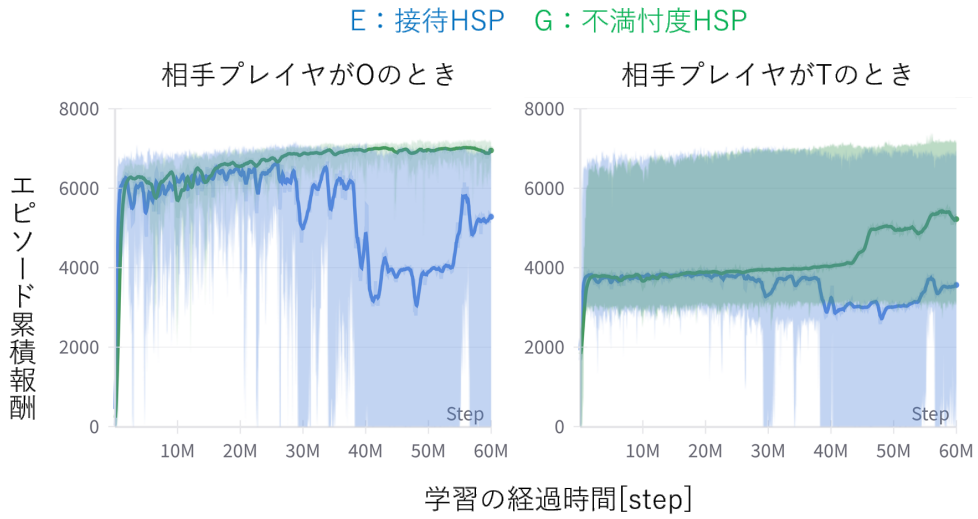

図 5.6: 本番用エージェントが学習に用いた累積報酬の学習曲線

相手エージェントがいずれの場合でも，学習が収束した本番用エージェントについて，不満忖度 HSP の本番用エージェントの方が優位な方策を獲得できていることがわかる．相手が F のとき，いずれの手法の本番用エージェントもある程度 F を満足させられる方策を獲得しているが，接待 HSP を用いた方は学習が安定していない．これは，F と P を見分けることが十分に出来ていないために，P の効用を F の効用と誤って学習してしまう割合が高いことに起因すると考える．相手が P のとき，両者の差はより顕著となっている．不満忖度 HSP の本番用エージェントは，収束後に安定的に高い報酬を得ているのに対し，接待 HSP を用いた方は，不満忖度 HSP と同等の報酬を得る試行もあるものの平均として低い水準に留まっている．

図 5.6 の右の学習曲線を見ると，不満忖度 HSP の本番用エージェントは 40M-50M ステップ付近を境に，急速に累積報酬のばらつきが少ない方策に収束していったことがわかる．図 5.7 は，本番用エージェントの学習中に，各エピソードで訓練相手エージェント F が不満を示す挙動を行った時間の推移を示したものである．ここから，F が不満を示すような行動（皿を消費して料理を作る）が発生するようになったのが 40M-50M ステップであり，それによって不満忖度 HSP の本番用エージェントが F と P を見分けられるようになったことが推察できる．

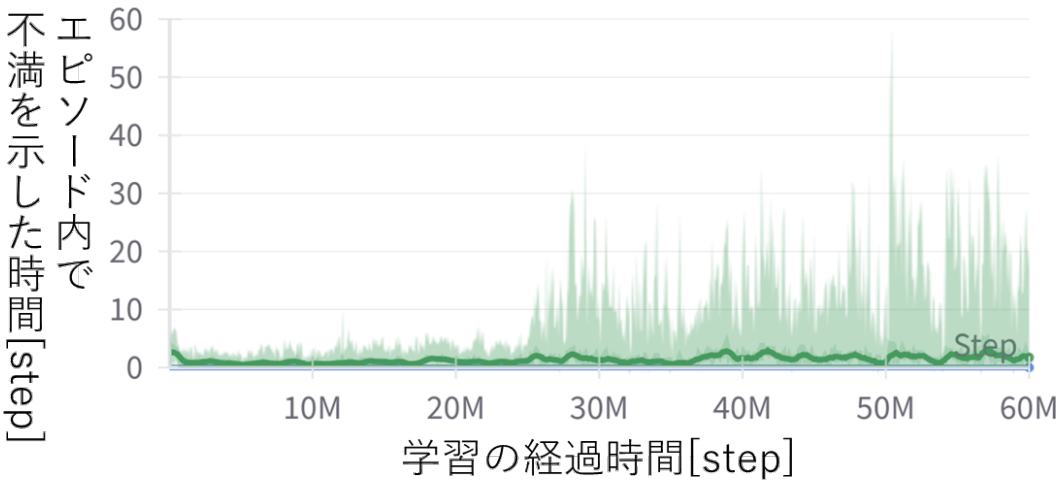

図 5.7: 本番用エージェントの学習中に F が不満を示した時間

この結果から，外見上の方策が類似する一方，相反する効用を有するエージェント 2 体の Population に対して，不満を示すような行動パターンを付与した不満忖度 HSP の本番用エージェントは，接待 HSP を用いて訓練したものよりも高い接待性能を示すことが確認された．実際の人間との協力を想定したとき，不満を示すような行動パターンを有するエージェントを Population に含ませることで，積極的に不満を示すような人間をより満足させることが可能になると考える．

第6章　おわりに

協力型ゲームで人間を楽しませるような AI を設計するとき，単に主目的を協力して達成する以外にも，人間プレイヤの副目的を尊重した戦略を AI が取ることが，人間プレイヤのゲーム体験にとって重要となる．本研究では，Population-based Training の既存手法である Hidden Utility Self-Play（原始 HSP）を，よりこのような需要に応えられるよう発展させた「接待 HSP」および「不満忖度 HSP」を提案した．

接待 HSP では，原始 HSP に対して 2 つの工夫を加えた．1 つは，本番用エージェントの訓練相手となるエージェントの学習時に，主目的を必ずしも含まない効用のエージェント同士での協調の獲得を可能にする工夫である．これにより，原始 HSP の Population の訓練時に副目的に対するプレイヤ間の協調が学習されない点が改善されると考えた．2 つ目は，本番用エージェントの学習時に，訓練相手となるエージェントの効用を満足させたときに報酬を与える工夫である．これは，原始 HSP の本番用エージェントの学習時に主目的と関係の薄い相手の効用を満足させる方策が得られない点を改善すると予想した．「外見上の振る舞い，満足させる接待戦略が異なる 2 種類の AI への接待」および「主目的とは関係ない副目的への協調」を題材とした実験で，この予想が正しいことが検証された．

一方，接待 HSP における Population では，相反する効用を持ちながら，外見上の方策が類似したエージェントの組が発生する場合があり，そのような相手を見分けるのが原理的に難しいという課題が考えられた．不満忖度 HSP では，人間の中には，ゲーム内の行動で不満を示すような相手もいることに注目し，そのような挙動を示す Population を導入することで，本番用エージェントが見分けられる相手の範囲を大きくすることを目指した．実験の結果，不満忖度 HSP は不満を示すような 2 体のエージェントを見分け，それぞれを高い水準で満足させるような方策を得ることが確認された．これは，接待 HSP の Population に不満を示すエージェントを含めることで，積極的に不満を示すような人間をより満足させるような本番用エージェントが得られることを示唆するものであると考えた．

本研究はこのように一定の成果を得た一方，本文で明記していない実験のいくつかにおいて，想定された結果が得られなかった．その主な要因として，訓練相手エージェントの Self-Play による経験の偏りを十分に見積もれていなかったことがあると考える．Self-Play では，同様の報酬関数のエージェント同士で学習を行うため，報酬を多く獲得できるような状況に遭遇したとき，急速に特定の振る舞いに方策が収束していく．そのため，報酬が高くならない場合への寄り道が少な

く，本番用エージェントとのプレイで未知の状況に遭遇した際に，洗練されていない挙動を取ってしまう可能性がある．結果として，本番用エージェントはそれを以て訓練相手エージェントを見分け，本研究で提案した工夫を行わずとも，エージェントを満足させるような方策を学習するに至った可能性がある．この仮説に関する検証を行っていくことで，Population-based Training のメカニズムについての有益な知見が得られると考える．

本研究で実装した不満忖度 HSP の訓練相手エージェントは，簡単のため，単純な条件分岐と機械的な行動パターンによって，不満を示す挙動を表現した．これは，様々な問題に一般化させることを想定していない方式であり，より一般化した方法を提案することができれば，大きく発展の可能性が広がる．例えば，強化学習の主要な手法で採用されている，TD 誤差や Q テーブルの値を用い，本研究で扱った「期待累積報酬が低くなる場合」を検知するというような一般化が考え得る．また，不満を示す挙動についても，人間一般に通ずるヒューリスティックや人間のデータをもとにモデリングすることで，実際の人間に類似したものに置き換えるような改良が可能である．これにより，先行研究で行われているように，大きな Population を用いて初対面の人間への性能を最大化させるような運用も可能になると考える．

付 録A　実験設定の一覧

4.3 節，5.3 節の実験では，学習のアルゴリズムに Proximal Policy Optimization（PPO）を使用している．全実験に共通するパラメータを表 A.1 に示す．

表 A.1: 全実験に共通するパラメータ

パラメータ	値
gradient clip norm	10.0
GAE lambda	0.95
gamma	0.99
networkinitialization	Orthogonal
use reward normalization	True
learning rate	5e-4
ppo epoch	15
エピソード長	400 ステップ

次に，実験によって設定が異なるパラメータを記載する．

訓練相手エージェントと本番用エージェントの学習では，entropy coef，使用するニューラルネットワークの種類，Rollout の並列スレッド数を変更している．entropy coef は，訓練相手エージェントの学習で 0.001，本番用エージェントの学習で 0.01 としている．ニューラルネットワークは，訓練相手エージェントの学習には MLP を，本番用エージェントの学習には RNN を一律で使用している．並列スレッド数は，訓練相手エージェントの学習で 80，本番用エージェントの学習で 200 であった．

各実験で設定を変更したのは，総実験ステップ数と Reward Shaping の Horizon（Reward Shaping の報酬が 0 になるステップ数）のみである．Reward Shaping の Horizon は，総実験ステップ数と等しい値を取る仕様としているため，ここでは表 A.2 に各実験と総実験ステップ数の対応を示す．

表 A.2: 各実験の総実験ステップ数

実験	総実験ステップ数
全訓練用エージェントの学習	10M
本番用エージェント A，B の学習	70M
本番用エージェント C，D の学習	5M
本番用エージェント E，G の学習	60M

付 録B　Reward Shaping

Reward Shaping とは，Step の経過に応じて減衰していく固定の報酬関数であり，Overcooked 環境における料理の作成・提出など，Reward Shaping を用いないと学習されにくい比較複雑な方策の学習を補助させようとするものである．本研究では，一貫して表 B.1 の 2 種類の Reward Shaping を使用している．特に，全ての本番用エージェントの訓練には，Reward Shaping の I. を使用している．Reward Shaping I. は，料理の前半〜中盤の工程である「皿を山から取る」「食材を鍋に入れる」「料理を鍋から取る」に対して与えられ，学習の後半にそれらが減衰していくことで,「料理を提出する」方策が学習されるようになる．一方の II. は，カウンターを介して皿を相手プレイヤに渡すことに与えられ，図 4.2 のステージで「皿を可能な限り並べる」ような方策の学習の前半部分を補助するものである．Reward Shaping I. は，原始 HSP を含む複数の研究で使用されていたものであり，II. は本論文で独自に使用したものである．

表 B.1: Reward Shaping の種類

種類	皿を山から取る	食材を鍋に入れる	料理を鍋から取る	皿を相手にカウンターを介して渡す
I.	+3	+3	+5	0
II.	0	0	0	+3

参考文献

[1] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems, 25.

[2] Yurtsever, E., Lambert, J., Carballo, A., & Takeda, K. (2020). A survey of autonomous driving: Common practices and emerging technologies. IEEE access, 8, 58443-58469.

[3] Sünderhauf, N., Brock, O., Scheirer, W., Hadsell, R., Fox, D., Leitner, J., ... & Corke, P. (2018). The limits and potentials of deep learning for robotics. The International journal of robotics research, 37(4-5), 405-420.

[4] Jumper, J., Evans, R., Pritzel, A., Green, T., Figurnov, M., Ronneberger, O., ... & Hassabis, D. (2021). Highly accurate protein structure prediction with AlphaFold. nature, 596(7873), 583-589.

[5] Ruthotto, L., & Haber, E. (2021). An introduction to deep generative modeling. GAMM ‐ Mitteilungen, 44(2), e202100008.

[6] OpenAI, Introducing ChatGPT, https://openai.com/index/chatgpt/ [アクセス日：2023/12/30]

[7] MNIH, Volodymyr. Playing atari with deep reinforcement learning. arXiv preprint arXiv:1312.5602, 2013.

[8] Fei, C., Wang, B., Zhuang, Y., Zhang, Z., Hao, J., Zhang, H., ... & Liu, W. (2020). Triple-GAIL: a multi-modal imitation learning framework with generative adversarial nets. arXiv preprint arXiv:2005.10622.

[9] Park, J. S., O'Brien, J., Cai, C. J., Morris, M. R., Liang, P., & Bernstein, M. S. (2023, October). Generative agents: Interactive simulacra of human behavior. In Proceedings of the 36th annual acm symposium on user interface software and technology (p. 1-22).

[10] OGAWA, Tatsuyoshi; HSUEH, Chu-Hsuan; IKEDA, Kokolo. More Human-Like Gameplay by Blending Policies from Supervised and Reinforcement Learning. IEEE Transactions on Games, 2024.

[11] SATO, Naoyuki; IKEDA, Kokolo; WADA, Takayuki. Estimation of player's preference for cooperative RPGs using multi-strategy Monte-Carlo method. In: 2015 IEEE Conference on Computational Intelligence and Games (CIG). IEEE, 2015. p. 51-59.

[12] 板東宏和, 池田心, Hsueh Chu-Hsuan, 人間プレイヤを活躍させる協力型ゲームの味方 AI, 情報処理学会第 49 回 GI 研究発表会, 2023-3.

[13] Campbell, M., Hoane Jr, A. J., & Hsu, F. H. (2002). Deep blue. Artificial intelligence, 134(1-2), 57-83.

[14] DeepMind, AlphaGO, https://deepmind.google/research/breakthroughs/alphago/ [アクセス日：2024/12/30]

[15] Vinyals, O., Babuschkin, I., Czarnecki, W. M., Mathieu, M., Dudzik, A., Chung, J., ... & Silver, D. (2019). Grandmaster level in StarCraft II using multi-agent reinforcement learning. nature, 575(7782), 350-354.

[16] OpenAI, OpenAI Five defeats Dota 2 world champions, https://openai.com/index/openai-five-defeats-dota-2-world-champions/ [アクセス日：2024/12/30]

[17] Siu, H. C., Peña, J., Chen, E., Zhou, Y., Lopez, V., Palko, K., ... & Allen, R. (2021). Evaluation of human-ai teams for learned and rule-based agents in hanabi. Advances in Neural Information Processing Systems, 34, 16183-16195.

[18] Meta Fundamental AI Research Diplomacy Team (FAIR) †, Bakhtin, A., Brown, N., Dinan, E., Farina, G., Flaherty, C., ... & Zijlstra, M. (2022). Human-level play in the game of Diplomacy by combining language models with strategic reasoning. Science, 378(6624), 1067-1074.

[19] 経済産業省，業界の現状及びアクションプラン（案）について【ゲーム】（事務局資料③），https://www.meti.go.jp/shingikai/mono_info_service/entertainment_creative/pdf/002_04_03.pdf

[20] newzoo, The PC & Console Gaming Report 2024, https://newzoo.com/resources/trend-reports/pc-console-gaming-report-2024

[21] 藤井叙人, 佐藤祐一, 若間弘典, 風井浩志, & 片寄晴弘. (2014). 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, 55(7), 1655-1664.

[22] 南基大, 池田心, 感情演出による楽しませる対戦型格闘ゲーム AI, 第 47 回 GI 研究発表会, 2022-3

[23] 佐藤直之, & 池田心. (2016). Influence Map を用いた経路探索による人間らしい弾避けのシューティングゲーム AI プレイヤ. ゲームプログラミングワークショップ 2016 論文集, 2016, 57-64.

[24] LERER, Adam; PEYSAKHOVICH, Alexander. Learning existing social conventions via observationally augmented self-play. In: Proceedings of the 2019 AAAI Conference on Artifitial Intelligence. 2019. p. 107-114.

[25] Carroll, M., Shah, R., Ho, M. K., Griffiths, T., Seshia, S., Abbeel, P., & Dragan, A. (2019). On the utility of learning about humans for human-ai coordination. Advances in neural information processing systems, 32.

[26] team17，Overcooked!，https://www.team17.com/games/overcooked [アクセス日：2025/12/30]

[27] Hu, H., Lerer, A., Peysakhovich, A., & Foerster, J. (2020, November). “other-play” for zero-shot coordination. In International Conference on Machine Learning (pp. 4399-4410). Proceedings of Machine Learning Research.

[28] Jaderberg, M., Dalibard, V., Osindero, S., Czarnecki, W. M., Donahue, J., Razavi, A., ... & Kavukcuoglu, K. (2017). Population based training of neural networks. arXiv preprint arXiv:1711.09846.

[29] Strouse, D. J., McKee, K., Botvinick, M., Hughes, E., & Everett, R. (2021). Collaborating with humans without human data. Advances in Neural Information Processing Systems, 34, 14502-14515.

[30] Lupu, A., Cui, B., Hu, H., & Foerster, J. (2021, July). Trajectory diversity for zero-shot coordination. In International conference on machine learning (pp. 7204-7213). Proceedings of Machine Learning Research.

[31] Zhao, R., Song, J., Yuan, Y., Hu, H., Gao, Y., Wu, Y., ... & Yang, W. (2023, June). Maximum entropy population-based training for zero-shot human-ai coordination. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 37, No. 5, pp. 6145-6153).

[32] Yu, C., Gao, J., Liu, W., Xu, B., Tang, H., Yang, J., ... & Wu, Y. (2023). Learning zero-shot cooperation with humans, assuming humans are biased. arXiv preprint arXiv:2302.01605.

[33] Long, W., Wen, W., Zhai, P., & Zhang, L. (2024). Role Play: Learning Adaptive Role-Specific Strategies in Multi-Agent Interactions. arXiv preprint arXiv:2411.01166.

[34] Li, Y., Zhang, S., Sun, J., Du, Y., Wen, Y., Wang, X., & Pan, W. (2023, July). Cooperative open-ended learning framework for zero-shot coordination. In International Conference on Machine Learning (pp. 20470-20484). Proceedings of Machine Learning Research.

[35] 中川絢太, 佐藤直之, & 池田心. (2016). ゲームの目的達成のみを追求した AI では生まれにくいゲーム内行動の分類と考察. 研究報告エンタテインメントコンピューティング (EC), 2016(20), 1-9.

[36] Wang, X., Zhang, S., Zhang, W., Dong, W., Chen, J., Wen, Y., & Zhang, W. (2024). Zsc-eval: An evaluation toolkit and benchmark for multi-agent zero-shot coordination. In The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track.

[37] Schulman, J., Wolski, F., Dhariwal, P., Radford, A., & Klimov, O. (2017). Proximal policy optimization algorithms. arXiv preprint arXiv:1707.06347.