Vol.2025-GI-54 No.12
2025/3/7

人間の副目的や不満を尊重する協力型ゲームAIの提案

林 辰宜[1,a)]　池田 心[1,b)]　シュエ ジュウシュエン[1,c)]

概要：ゲーム AI は，プレイヤ間の協力を含むゲームにおいても，人間に匹敵，超越する強さを獲得しつつある．Population-based Training(PBT) では，人間のプレイデータを用いない強化学習により，様々な特性を有する訓練相手 AI のプールを作成し，それらとプレイした際にゲームの主目的（スコア等）が高くなるような本番用 AI を訓練する．この本番用 AI は，様々な相手の癖を学習しているため，初対面の相手とプレイした際にも，相手の動きに対応して高いスコアを達成することが可能である．一方，既存手法では，主目的を追求することのみを目的とし，人間に見られるような主目的とは関係の薄い副目的は考慮されていない．本研究ではまず，主目的ではなく相手の副目的を叶えることを目指す「接待 HSP」を提案する．さらに，人間同士の協力では，お互いの目的にすれ違いがあった場合に，ゲームの内外の行動でそれを伝達しようとする場合があることを踏まえ，そのような伝達行動を示すような訓練相手 AI を用いることで相手の効用を識別し，より適切に AI を協調させる「不満忖度 HSP」を提案する．

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，良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）[1] と呼ばれる枠組みがある．PBT 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 段階目では，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 と独立させて学習する．2 段階目では，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 を学習する．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．訓練用 Population が様々な振る舞いを見せるゲーム AI から構成され，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）[2] 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．この観点において，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る．

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 つの手法を提案する．

1 つ目の手法は「接待 HSP」と呼ぶもので，(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ものである．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

[1] 北陸先端科学技術大学院大学
Japan Advanced Institute of Science and Technology(JAIST), Nomi, Ishikawa 923-1211, Japan
[a)] tatsugi@jaist.ac.jp
[b)] kokolo@jaist.ac.jp
[c)] hsuehch@jaist.ac.jp

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る．

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る．

2. 関連研究

2.1 Population-based Training

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 [3] 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を獲得するようになる．

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 [3] 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．

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ない．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）と呼ばれる問題設定が注目されている [4]．ZSC ではテスト時（本番）の協力相手の方策に関する情報を，訓練時に知ることはできず，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．この問題設定では，自己学習しかしない SP や，十分なデータを要する模倣学習は有力な選択肢ではない．

ZSC に対する主要なアプローチは，1 章で紹介した Fictitous Co-Play（FCP）[1] や Maximum Entropy Population-based Training（MEP）[5] がある．初めての相手とも協力できるようにするためには，Population の多様さが重要であることが知られている．

2.2 Hidden Utility Self-Play

Hidden Utility Self-Play（HSP）[2] 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．これにより，R で規定される報酬の期待累積値を最大化する方策を目指す．

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

- 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

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．

$$\begin{aligned} J(\pi_a,\pi_\omega|R_m) &= \sum_t \mathbb{E}_{(s_t,a_t)\sim\pi}[R_m(s_t,a_t)] \\ J(\pi_a,\pi_\omega|R_\omega) &= \sum_t \mathbb{E}_{(s_t,a_t)\sim\pi}[R_\omega(s_t,a_t)] \end{aligned} \quad (1)$$

- 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，$\mathcal{R}$ はあらゆる人間の効用を意味し，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，$\mathcal{R}$ をイベント特徴量に対する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 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 π_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[*1]．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．

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．

$$
\begin{aligned}
J(\pi_A^{'}) &= \mathbb{E}_{R_\omega \sim \hat{P_R}}[J(\pi_A^{'}, \widetilde{\pi}_\omega(R_\omega))] \\
&= \sum_t \mathbb{E}_{(s_t, a_t) \sim \pi_A^{'}, R_\omega \sim \hat{P_R}}[R_m(s_t, a_t) | \widetilde{\pi}_\omega(R_\omega)]
\end{aligned}
\tag{2}
$$

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

このように，Population は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．

実装上は，主目的に対する報酬関数 R_m を報酬関数として与えたもとで，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，

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 1: Overcooked 環境における報酬空間の例

[*1] 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場合は，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

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．

3. 提案手法 1：接待 HSP

3.1 原始 HSP の課題

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利用し，本番用エージェントは主目的報酬のみを最大化する設定となっていた．一方，実際の人間は，主目的に留まらず，複数の副目的を包含した「効用」の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 1** のような「Overcooked 環境」[3] のステージを考える．Overcooked 環境は，アクションゲーム「Overcooked！」をもとに，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．図 1 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．料理は，食材（画面左側の玉ねぎ，トマト）を鍋（画面右側）に入れて火にかけ，表の「調理時間」に対応するステップ数の経過を待った後に鍋の中身を皿（画面中央）に盛ることで得られる．この料理を提出口（画面右側上・下部）に提出することで，料理に応じたスコアが得られる．

ここで，右プレイヤ（青の帽子）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」「トマトを含む料理を提出したら+30」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

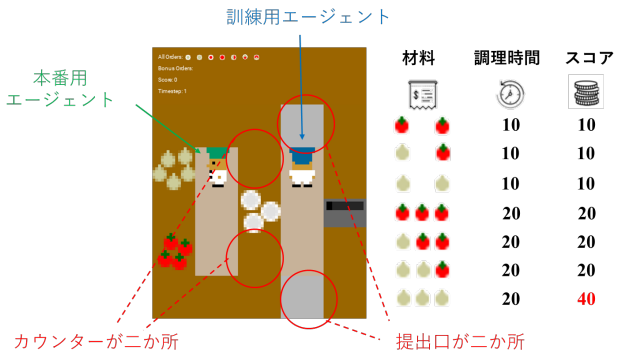

図 1: Overcooked 環境のステージ 1

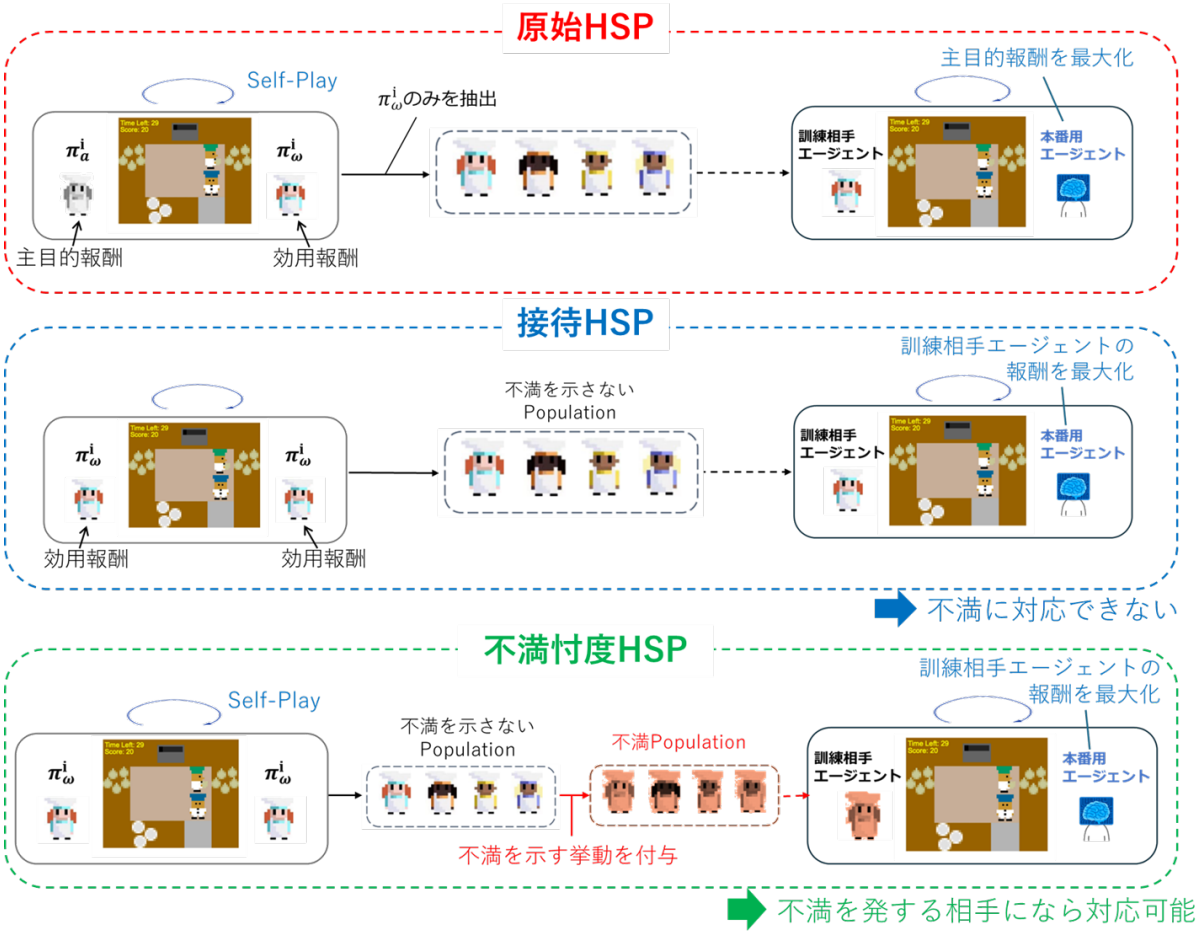

図 2: 提案手法の概念図

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：$40 \times 1 = 40$）する工程を続ければ良い．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 3 個の料理を作成して提出（獲得報酬：$20 \times 1 + 30 = 50$）する工程を続ける方が高い報酬を得られる．

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．

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．一方で，原始 HSP の本番用エージェントは，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[*2]．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．

相手の効用に対する報酬を最大化させることを考えたとき，O に対して上のカウンタに玉ねぎのみを，T に対しては下のカウンタにトマトを置くことが求められるが，原始 HSP の本番用エージェントは原理的に O のみを満足させる方策に収束する．人間を楽しませる観点において，これは原始 HSP の課題の 1 つであると考える．

3.2 接待 HSP

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（**図 2** 上）．接待 HSP におけ

[*2] 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

る本番用エージェントの学習では，原始 HSP 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_ω^j を報酬関数として与える．例えば，3.1 節で述べた例（図 1）でトマト好きの T を相手にした場合，玉ねぎ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．

図 1 の例では，O と T はどちらのカウンタで待つかという外見上の違いがある．そのため，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できる．

3.3 評価実験

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を相手にしたときの本番用エージェント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，ここでは，原始 HSP が対応できないような状況を具体的に想定し，それに接待 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．

本実験では，先述の図 1 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」に加え，次に説明する**図 3** のステージにおける「皿が置かれている状態が好きな F」の 2 つの例について，実験を行った．以降，それぞれを実験 1，実験 2 と呼称する．

実験 1 の例は，主目的を含むような効用の相手を想定しているが，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 [6]，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 3 のようなステージで，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に置く必要がある．

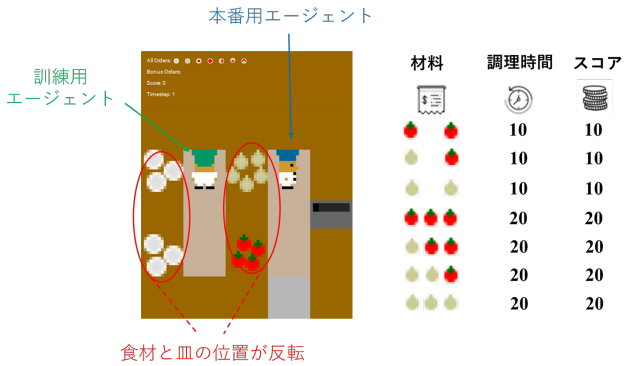

図 3: Overcooked 環境のステージ 2

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，左プレイヤに「皿が置かれている状態が好きな F」，右プレイヤに原始 HSP の本番用エージェントが割り当てられたとき，本番用エージェントは F 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．

実験 1 で使用した訓練相手エージェントの報酬関数の一覧を**表 2** に，実験 2 で使用したものを**表 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，「スコア獲得」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．

また，これらの報酬だけでは，目的の振る舞いを学習できないことが実験した結果明らかになったため，先行研究 [5],[2] に習い，Reward Shaping も加えて学習を行った．Reward Shaping は，学習開始時に最大値を取り，学習の最終ステップで 0 になるよう線形的に減衰していくような報酬関数である．本実験では，F の学習に「相手にカウンタを介して皿を渡したら+3」，その他の訓練相手・本番用エージェントの学習に「皿を山から取ったら+3」「食材を鍋に入れたら+3」「料理を鍋から皿に盛ったら+5」という最大値を取る Reward Shaping を設定した．

訓練相手エージェントは各種 5 シードづつ訓練し，それら全ての効用エージェントから Population を構成した．本番用エージェントの訓練時には，その Population からランダムにサンプリングしたエージェントを訓練相手とした．

表 2: 実験 1 で訓練相手 O，T の学習に用いた報酬関数

	トマトを山から取る	トマトを鍋に入れる	スコア獲得
O	0	0	× 1
T	+5	+5	× 1

表 3: 実験 2 で訓練相手 F の学習に用いた報酬関数

	皿をカウンタに置く	皿をカウンタから取る	皿が上限まで置かれている
F	+1	-1	毎ステップ+20

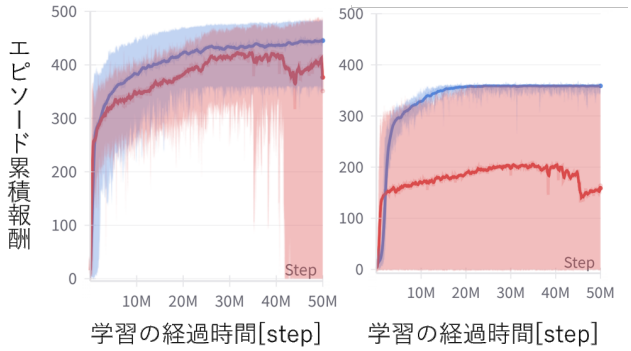

図 4: 原始 HSP（赤）と接待 HSP（青）の比較.
相手が O の場合（左）と T の場合（右）

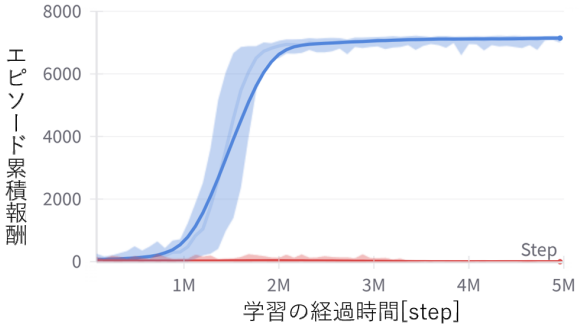

図 5: 原始 HSP（赤）と接待 HSP（青）の比較（相手が F）

強化学習のアルゴリズムは，Overcooked 環境を対象にした先行研究同様，PPO を用いた．ニューラルネットワークには，3 層の CNN の後ろに，訓練用エージェントの学習では 2 層の全結合層，本番用エージェントの学習では 1 層の GRU とその更に後ろに 2 層の全結合層が置かれたものを使用した．

3.3.1 実験 1 の結果

本実験では，原始 HSP と接待 HSP の本番用エージェントが，学習中にどれだけ相手（トマト好きの T と玉ねぎ好きの O）を満足させられるようになったかを調べる．本番用 AI は，相手が誰なのか分からず，挙動の違いから見分け，対応を変えなければならない．満足度合は，T や O が受け取ったエピソードあたりの累積報酬で評価する．

本番用エージェント「A：原始 HSP」「B：接待 HSP」の学習経過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（5 試行の平均）を**図 4** に示す．

相手エージェントが T の場合，学習が収束した本番用エージェントについて，接待 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．これは，4.2 で予想した通り，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に置く」場合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したことを示している．この結果から，この例のように外見上の振る舞いや満足させる接待戦略が異なる相手には，接待 HSP が有効であると考える．

3.3.2 実験 2 の結果

本番用エージェント「C：原始 HSP」「D：接待 HSP」の学習経過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（5 試行の平均）を図 5 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．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ていると考察できる．

この 2 つの実験結果から，「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「主目的と関係の薄い副目的を有する AI への接待」を試みる場合について，接待 HSP の本番用エージェントは原始 HSP のそれを上回っていると考える．

4. 提案手法 2：不満忖度 HSP

4.1 接待 HSP の課題

接待 HSP では，様々な効用を持つ訓練相手を作り，それぞれの効用の合計を高めるように本番用エージェントの方策を学習していく．結果として，本番用エージェントは，相手の振る舞いに区別がつくならば，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期待できる．

しかし，必ずしも訓練相手エージェントの外見上の挙動のみで，その効用を識別できるとは限らない．例えば，図 3 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「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」を配置することを考える．左プレイヤが訓練相手エージェント，右プレイヤが本番用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．そのため，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．このように，外見上の方策が一致する効用の相手を，相手を行動系列のみをもとに見分けて協調するのは原理的に不可能である．

4.2 不満忖度 HSP

4.2.1 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 [6]．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，しゃがみ動作でお礼を表したり，ジャンプ動作で呼びかけを行うなど，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，ゲーム内の行動系列を用いた伝達行動に注目する．

人間の伝達行動は，4.1 節で挙げたような状況で，相手が何に不満を持っているのか見分けるのに役立つ．仮に左プレイヤが「なるべく多く皿が置かれている状態」が好きで，相手に不満をアピールするようなプレイヤの場合，相手が置かれている皿を取って料理に使用してしまったときに，困惑して立ち尽くしたり，あえて不用意に食材をカウンタに置くなどの妨害行為を行い抗議の意を示したりするかもしれない．これは，「自分で皿を置くこと」が好きなプレイヤとは外見上異なる方策であり，本番用エージェントは原理的には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（図 2 下）．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．

4.2.2 不満を示す挙動

人間はさまざまな場合に不満を感じ，さまざまな方法で不満を表す．本論文では，自分の満足度が低くなりそうなとき，つまり「効用に対する期待累積報酬が低くなる」ときに，特定の挙動を行うものとする．例えば，4.1 節の例で相手が料理をせずに皿を並べようとしているとき，自分が皿を置くのが好きな F は，その試合（エピソード）で置ける皿の数の上限が低くなったことに，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．

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，4.1 の例で F が本番用エージェントに皿を消費されたときに不満を示すような事例では，相手が「料理を皿に盛る」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．

4.3 評価実験

接待 HSP が対応できないような状況を採り上げ，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，評価を行う．本実験では，4.1 で挙げたような図 3 のステージで「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．

F の学習時に与える報酬関数は表 3 と同様のものを使用し，P の学習には**表 4** のものを使用した．ここで，報酬の値は，本番用エージェントが片方のエージェントに偏った方策に収束しないよう，F と P を最大限満足させた場合の累積報酬値が同程度になるように設定した．訓練相手エージェントは各種 5 シードづつ訓練し，それら全ての効用エージェントから構成する Population を本番用エージェントの学習に使用した．

不満を示す条件は，F は「本番用エージェントが料理を皿に盛った」，P は「本番用エージェントが皿をカウンタに置いた」が過去 10 ステップ以内に 1 回以上発生している

表 4: 訓練相手 P の学習に用いた報酬関数（右側 2 つは，訓練時に右側プレイヤを担当した場合のもの）

	皿をカウンタに置く	皿をカウンタから取る	鍋を火にかける	料理を提出
P	+400	-400	+1	+400

場合とした．この条件を満たしている場合に，F と P のいずれも「上下交互に移動する」行動のパターンを示す．

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（5 試行の平均）を**図 6** に示す．

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（右側も皿で埋める）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．相手が P のとき，不満忖度 HSP の本番用エージェントは，収束後に比較的安定的に高い報酬を得ている（右側で料理をして皿を消費し，置けるようにする）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 6 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 30M ステップ以降を境に，報酬の高い方策に収束していったことがわかる．**図 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，30M ステップ以降で（学習はしていない）F が不満を示すようになっており，つまり本番用 AI が料理を作れるようになってきたことが分かる．そして徐々に不満の数は減っていくので，本番用 AI は，F と P を数回の不満から素早く見分けて，それぞれに適切な行動を取れるようになっていったということである．

外見上の方策が類似する一方で相反する効用を有するプレイヤ群に対し，接待 HSP の枠組みでは適切な対応が取れなかった．不満忖度 HSP では，プレイヤ群に不満を示す

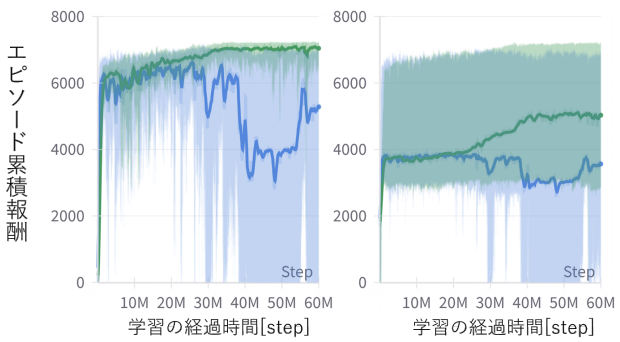

図 6: 接待 HSP（青）と不満忖度 HSP（緑）の比較．相手が F の場合（左）と P の場合（右）

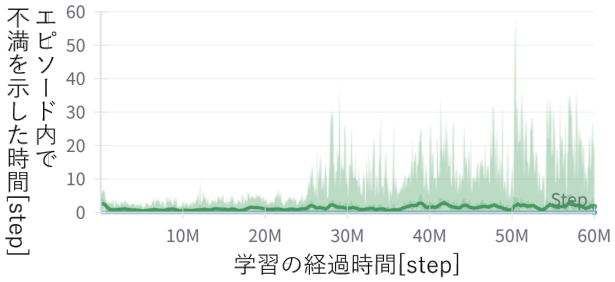

図 7: F が不満を示した時間の推移

ような行動パターンを付与することで，これに対応できるようにした．実際の人間との協力を想定したとき，さまざまな不満のパターンを付与したエージェントを Population に含めることで，積極的に不満を示すような人間に対してはより満足させることが可能になると考える．

5. おわりに

協力型ゲームで人間を楽しませる 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することが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 の不満提示部分を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．実応用を考えると，期待累積報酬の低下を条件とするなどの一般化や，模倣学習を用いた行動パターン生成も有望であろう．

参考文献

[1] STROUSE, D. J., et al. Collaborating with humans without human data. NeurIPS, 2021.
[2] YU, Chao, et al. Learning zero-shot cooperation with humans, assuming humans are biased. arXiv preprint arXiv:2302.01605, 2023.
[3] CARROLL, Micah, et al. On the utility of learning about humans for human-ai coordination. NeurIPS, 2019.
[4] HU, Hengyuan, et al. “other-play” for zero-shot coordination. ICML. PMLR, 2020.
[5] ZHAO, Rui, et al. Maximum entropy population-based training for zero-shot human-ai coordination. AAAI, 2023.
[6] 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.

正誤表

- P.1　1. はじめに　3 段落　「～と呼ばれる枠組みがある.」の次に挿入

  その定義には様々なものが考えられるが，本研究で は「Population を用い，2 段階の工程で初対面の人間との 協調可能な AI の作成を目標としている強化学習手法」を PBT と呼称するものとする.

- P.4　3.1 節　5 段落

  X　4.1 節で述べたように　　○　一方で，原始 HSP の本番用エージェントは

- P.4　3.2 節　1 段落　3 行目　　X　図 2 上　　○　図 2 中段

- P.5　表 2　列「トマトを鍋に入れる」2 行目　　X　+5　　○　+10

- P.6　3.3.1 項　3 段落　4 行目　　X　4.2　　○　前節

- P.6　4.1 節　2 段落　4 行目-6 行目

  X　左プレイヤに本番用エージェント，右側にスコアを「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」を配置することを考える.

  ○　左プレイヤに「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」，右プレイヤに本番用エージェントを配置することを考える.

- P.6　4.1 節　下から 2 行目　　X　相手を，相手を　　○　相手を，

- P.7　4.2.2 項　1 段落 6 行目　　X　F　　○　P

- P.8　3 段落　下から 1 行目　　X　上下交互に移動する　○　下方向に移動する

- P.8　4.3 節 6 段落　下から 1 行目-5 行目

  X　つまり本番用 AI が料理を作れるようになってきたことが分かる．そして徐々に不満の数は減っていくので，本番用 AI は，F と P を数回の不満から素早く見分けて，それぞれに適切な行動を取れるようになっていったということである.

  ○　本番用エージェントが料理 を作れるようになってきたことが分かる．つまり，本番用エージェントは，F と P の不満を示す行動から両者を見分け，それぞれに適切な行動を取れるようになっていったと考える.