

建築作業ゲームにおけるプレイヤーと協調動作を行うエージェントの提案 Collaborative Agent with Players in Construction Games

沼野 陸¹⁾ 山崎 憲一¹⁾
Riku Numano Kenichi Yamazaki

1 はじめに

『Minecraft』等の建築作業ゲームでは、プレイヤーはグリッドワールド上を移動しながらブロックを設置していくことで、自由な建築作業を行える。それに加え、複数のプレイヤーによる大規模な建築作業も可能である。

また近年では、デジタルゲーム等におけるプレイヤーのゲーム体験の向上や没入感を高めるために、コンピュータが制御するキャラクターであるゲームエージェントの重要性が増している。本研究では、建築作業ゲームにおける協調型エージェントに焦点を当てる。

従来の RPG ゲーム等における協調型エージェントは、『敵を倒す』といったプレイヤーと同じ具体的な目的を持つため、協調的な行動が行いやすい。一方、建築作業ゲームにおける協調型エージェントには、プレイヤーが行う建築作業の進行状況や、プレイヤーの行動に合わせて適切な行動を行うことが求められる。そのため、『建物を造る』という大きな目的を設定しただけでは、プレイヤーが期待するような協調的な行動を行うことは難しいと考えられる。本研究では建築作業ゲームにおいて、プレイヤーと協調動作を行うエージェントを提案する。

2 既存研究

グリッドワールド上において、エージェントが移動しながらブロックを設置することで、指定された建築物を完成させるタスク（以降、「移動建築タスク」と呼ぶ）を考案した研究 [1] がある。DQN を用いてエージェントを学習させることで、エージェントは指定された建築物を完成できるようになった。しかし、複数人で協調しながら移動建築タスクを行うことは考慮されていない。

ストラテジーゲームである『StarCraft』において、プレイヤーや他のエージェントと協調しながら行動を行うゲームエージェントを提案した研究 [2] がある。プレイヤーは『資源の収集』、『敵を攻撃』などの目標を持っている。エージェントはそれらの目標から得られる報酬を最大化するような行動を選択することで協調を図る。報酬は、簡単に数値化できるため、エージェントが適切な行動を選択することが容易である。しかし、建築作業ゲームにおいては、プレイヤーが行う建築作業の進行状況や、プレイヤーの行動に合わせて適切な行動を行うことが求められるため、このような手法をそのまま適用することは難しい。そこで、エージェントはプレイヤーの行動を意識しながら行動選択できるような報酬設計が必要となる。

3 提案と実装

本研究では、移動建築タスクをプレイヤーと協調しながら行うエージェントを提案する。本提案が目指す協調とは、プレイヤーが行う建築作業を、プレイヤーの行動を考慮しつつ、作業完了までの時間を短縮させることである。

1) Graduate School of Science and Engineering, Shibaura Institute of Technology

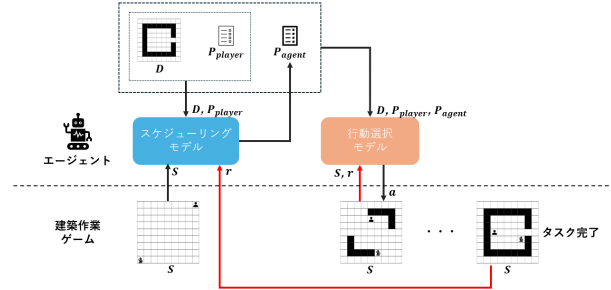


図 1 提案するエージェントの全体像

本研究ではプレイヤーの行動の配慮として、プレイヤーの行動を妨害しないことを主に考える。本章では、まず移動建築タスクを拡張した協調移動建築タスクについて説明し、その後に協調移動建築タスクをプレイヤーと協調しながら行うエージェントの提案について述べる。

3.1 タスクの概要

2次元グリッドワールド上において、2人のキャラクターが与えられた設計図に従ってブロックを設置することで建築物を完成させるタスクを考える。各キャラクターの位置やブロックの設置状況を指すグリッドワールドの状態を S 、各キャラクターが実行できる移動やブロックの設置等の行動を A とする。キャラクターには、建築物を構成する n 個のブロックの座標の集合である設計図 $D = \{d_1, d_2, \dots, d_n\}$ が与えられる。また、実際の建築作業ゲームには様々な種類のブロックが存在するが本研究では簡易化のためにブロックの種類は1つとする。

各キャラクターは1ステップ行動をするごとにターンを交代する。自身のターンでは、現在の状態 S を確認し、行動 $a \in A$ を選択する。行動 a は、移動、ブロックの設置のいずれかである。設計図 D の通りにブロックが設置されることでタスクが完了する。これ以降の章では、タスクを行う2人のキャラクターはプレイヤーとエージェントとする。

3.2 提案エージェント

提案するエージェントは、2つのモデルで構成される。1つ目は、設計図や環境から、エージェントが設置すべきブロックの順番をスケジュールとして決定するスケジューリングモデルである。2つ目は、決定されたスケジュールと環境から、移動やブロックの設置等の、エージェントが次に取るべき行動を選択する行動選択モデルである。タスク開始前にスケジューリングモデルがスケジュールを出力し、それをもとに行動選択モデルを繰り返し実行することでタスクを行う。提案するエージェントの全体像を図1に示す。

3.2.1 スケジューリングモデル

スケジューリングモデルは、プレイヤーの行動を妨害せず作業時間を短縮できるようなエージェントのブロック

の設置順序を決定する。スケジューリングモデルは、設計図 D とプレイヤーのスケジュール P_{player} とグリッドワールドの初期状態 S を入力とし、エージェントのスケジュール $P_{agent} = (p_1, p_2, \dots, p_n)$ を出力する。 P_{player} , P_{agent} は、ブロックの設置すべき順番を表す。また本研究では、 P_{player} はあらかじめ決定されていることとする。

スケジューリングには、入力シーケンスの要素の並び順を出力シーケンスとして出力するエンコーダ・デコーダモデルである PointerNetworks[3] を利用する。スケジューリングモデルの学習には方策勾配法を用いる。式 (1) を用いて、出力されたスケジュールの評価を行う。

$$R_{schedule} = H_a \times \frac{1}{s} - H_b \times d \quad (1)$$

式 (1) により、タスクの完了までのステップ数が少なく、かつプレイヤーを妨害する回数が少ないほど高い報酬が与えられる。 s はタスクの完了までに要したステップ数、 d はプレイヤーを妨害した回数、 H_a , H_b はハイパーパラメータである。プレイヤーを妨害した回数 d は、プレイヤーが移動やブロックを設置したい座標にエージェントが既に存在した場合に 1 ずつ加算される。

3.2.2 行動選択モデル

行動選択モデルは、 P_{agent} の順番通りにブロックを設置していくことを目的として、エージェントに行動させる。行動選択モデルは、現在の状態 S と設計図 D と各スケジュール P_{agent} , P_{player} を入力とし、次の行動 a を出力する。エージェントまたはプレイヤーによって、 D に従ったブロックが設置されると、 P_{agent} , P_{player} から当該ブロックを削除する。 P_{agent} が空になるまで、エージェントは次の行動を選択し続ける。

行動選択には階層型強化学習を用いる。「目標座標への移動」、「目標座標へブロックの設置」の 2 つのサブゴールに分割し、それぞれを DQN で学習させる。行動選択モデルの報酬は以下の通りである。

- P_{agent} の最初のブロックである p_1 の座標方向に移動した距離へ応じて報酬を与える。
- D に従った座標にブロックを設置した場合、そのブロック p_i の P_{agent} 中の順番 i の逆数 $1/i$ で報酬を与える。
- D の通りにすべてのブロックが設置されたときに、エージェントの貢献度に応じて報酬を与える。また、 D とは無関係な座標にブロックが設置されている場合、そのブロックの数に応じて報酬を差し引く。タスク完了時の報酬は、式 (2) で表される。

$$R_{complete} = H_c \times \frac{b_{correct}}{n} - H_d \times b_{incorrect} \quad (2)$$

$b_{correct}$ は D の通りに設置されたブロックの数、 $b_{incorrect}$ はそれ以外の設置ブロック数、 H_c , H_d はハイパーパラメータである。

- エージェントの設置したブロックにより、キャラクターが行動不能になった場合は大きな負の報酬を与える。

表 1 各難易度による評価

難易度	Easy	Normal	Hard
短縮率平均	1.66	1.62	1.32
妨害回数平均	0.250	0.250	1.62

4 評価

提案手法の有効性を評価する。ブロックの密度や形状の複雑さに基づいて Easy, Normal, Hard の 3 つの難易度の設計図と、それらに従ったプレイヤーのスケジュール P_{player} を 8 個ずつ用意した。プレイヤーとエージェントが協調することによる、タスク完了までのステップ数の短縮率を、式 (3) のようにプレイヤーが単独でタスクを完了させた場合のステップ数と比較することで求める。

$$eval = \frac{s_{solo} \times 2}{s_{coop}} \quad (3)$$

2 つのモデルにそれぞれ 15,000 エピソードの学習を行い、学習後、各設計図 D に対し協調移動建築タスクを実施した。評価結果を表 1 に示す。

3 つの難易度において、プレイヤーとエージェントが協調することによって、総ステップ数が減少した。ただし、難易度 Hard ではプレイヤーを妨害する回数が上昇し、他の難易度ほど短縮率が高くない。式 (1) による妨害回数のペナルティだけでは不十分と考えられる。それに加え、エージェントがプレイヤーを妨害するとプレイヤーのゲーム体験が悪化する可能性も考えられる。今後は妨害回数を減らすようなエージェントの行動選択方法の改善が必要になる。例えば、プレイヤーの行動予想の範囲をスケジュール P_{agent} から具体的な行動 A_{agent} まで拡張し、エージェントにプレイヤーの細かい行動を考慮させるような報酬とすることが考えられる。

また現状では、プレイヤーの行動手順が固定的であること、立体構造の建築物に対応できないことが課題である。さらに移動建築タスクにおける協調を短縮率と妨害回数だけで評価することも検討が必要である。

5 結論

本稿では、建築作業ゲームにおける協調型エージェントの提案を行った。提案したエージェントによって移動建築タスクの完了時間を短縮できることを示した。しかし、複雑な構造に対してはプレイヤーの行動を妨害してしまうことがあり、想定するような協調が実現できないことがわかった。今後は、協調の評価指標を見直し、プレイヤーとより円滑な協調を行えるように提案手法の改善を行う。

参考文献

- [1] W. Han, H. Wu, E. Hirota, A. Gao, L. Pinto, L. Righetti, and C. Feng, "Learning simultaneous navigation and construction in grid worlds," in *The Eleventh International Conference on Learning Representations*, 2023.
- [2] D. Xie and X. Zhong, "Semicentralized deep deterministic policy gradient in cooperative starcraft games," *IEEE transactions on neural networks and learning systems*, vol. 33, no. 4, pp. 1584–1593, 2020.
- [3] O. Vinyals, M. Fortunato, and N. Jaitly, "Pointer networks," *Advances in neural information processing systems*, vol. 28, 2015.