

MOBA における半自律型チームメイト AI

張輝陽^{†1} 星野准一^{†2}

Multiplayer Online Battle Arena (MOBA) はチーム対戦ゲームとして、プレイヤー間の協力行動が MOBA のゲームプレイの一つ重要な部分である。MOBA においてプレイヤーはマーキングを出すことで自分の意図を他のチームメイトに伝い、協力を求める。そして、他のプレイヤーはマーキングの意味を理解し、適切な行動でそのプレイヤーに協力する。プレイヤーはゲーム AI と一緒にプレイする時、もし AI エージェントも人間プレイヤーのようにプレイヤーが出したマーキングの意図を理解し、適切な戦術を選択してプレイヤーと協力することができる、ゲームの楽しさが向上できると考えられる。本稿ではグラフ理論と時相理論を用いた適切な戦術でプレイヤーと協力できるチームメイト AI を提案し、AI エージェントの行動とゲームの楽しさの関連性について検討する。

Semi-Autonomous Teammate AI in MOBA

HUIYANG ZHANG^{†1} JUNICHI HOSHINO^{†2}

The cooperation between players is an important part of the game play of Multiplayer Online Battle Arena (MOBA) which is a team-based battle game. When playing with human players, players use marking to tell teammates about his intention and seek for assistant. Other players will read the idea of the marking and do something to cooperate with that player. Therefore, when playing with game AI, if AI agents can also read the idea of marking and choose some tactics to cooperate with players, the game would be funnier. In this paper, we introduced a way to use graph theory and temporal logic to make AI agents reason about players' marking and cooperate with them, and then we will discuss about the relationship between AI agents' behavior and the merriment of game.

1. はじめに

Multiplayer Online Battle Arena (MOBA) は Real Time Strategy (RTS) のサブジャンルの一つで、プレイヤーが二つのチームに分けて対戦を行うようなチーム対戦ゲームである。従来の RTS と違うのは MOBA にはベースやユニットを作る必要が無いほか、プレイヤー一人につき一体のキャラクタを操ることである。MOBA は現在海外で高い人気を集まっており、その代表作の一つ League of Legends (LOL) では月間アクティブユーザ数が 2000 万を超えている[1]。

チーム対戦ゲームとして、プレイヤー間の協力が MOBA のゲームプレイにおいて重要な要素の一つである[2]。例えば、共に戦術を組み立てて実行することやキャラクタの役割を果たすことや、リソースを合理的に分配することなどが挙げられる。戦術を立てるときには適度なコミュニケーションが必要であるが、リアルタイムなゲームなのでコミュニケーションをする余裕が無い。そのため、MOBA においてはコミュニケーションの代わりにプレイヤーは特定の操作によってゲームのマップにマーキングを出すことで自分の意図をチームメンバーに伝達することができる。そして、他のプレイヤーはマーキングの意味を理解し、マーキングを出したプレイヤーの意図に合わせて適切な行動を取るようにになっている。MOBA はマルチプレイヤーのゲームであるが、プレイヤーはゲーム AI と一緒にプレイする場合もある。AI エージェントが人間のように行動する方がプレイヤーにとって

面白い[3]ので、AI エージェントも人間プレイヤーのように他のプレイヤーが出したマーキングを理解し、相手の意図に合わせて適切な行動を取ることが必要だと考えられる。

しかし、現在 MOBA の AI エージェントはプレイヤーのマーキングに対して、簡単な行動しか取れない。例えば、プレイヤーが「ある敵を倒す」という意図でマーキングを出したことに對して、AI キャラクタがすぐその敵に向けて適当に攻撃を始めることしかできない。しかし同じマーキングに対して、人間プレイヤーであれば敵を倒す意図を理解し、倒すための戦術を考えてからマーキングを出したプレイヤーと連携して行動することができる。もし AI キャラクタも人間プレイヤーのように行動できると、ゲームの楽しさが向上できると考えられる。

そこで、本稿では MOBA において人間のプレイヤーが出したマーキングに対してその意図を理解し、適切な戦術を選択してプレイヤーと連携できるような半自律型チームメイト AI を提案し、プレイヤーがどのようなゲーム AI と一緒にプレイする方が楽しいかを検討する。半自律というのは AI エージェントが普段勝手に動いているが、プレイヤーがマーキングを出す時、AI エージェントがマーキングに従って行動を変化することである。

2. Multiplayer Online Battle Arena

2.1 MOBA のゲームプレイ

ゲームを始める前にプレイヤーが複数のキャラクタから

^{†1} 筑波大学大学院システム情報工学研究科
Graduate School of Systems and Information Engineering, University of Tsukuba

^{†2} 筑波大学システム情報系
Faculty of Engineering, Information and System, University of Tsukuba

 : ベース
  : 進軍ルート



図 1 MOBA のマップ

一体のキャラクタを選ぶ。MOBA においてキャラクタはゲームごとにレベル 1 から始まり、次のゲームにレベルやアイテムなどを持ち越さないようになっている。

ゲームが開始すると、両チームのプレイヤーが図 1 のようなマップの両端にあるベースに現れ、相手チームのベースにある特定の建物を破壊することを目的としてマップ中の三つのレーンを通して自軍の NPC 兵士と共に敵ベースに向けて進行する。敵のプレイヤーや NPC 兵士を倒すことでお金や経験値を稼ぎ、強い装備アイテムを買うことやキャラクタをレベルアップさせることができるようになる。キャラクタのレベルが高ければ、スキルの効果や攻撃力などは強くなり、敵をより容易に倒せる。キャラクタは敵に倒された場合、一定時間を経過したら復活することができる。

また、一つのキャラクタはいくら強くても、複数の敵プレイヤーと同時に戦って、そして勝つことは難しい。そのため、チームメンバー間の協力が重要となる。チームメンバーの協力的行動として、敵を包囲することや同時に攻撃を仕掛けることや弱い味方を守るなどが挙げられる。本稿ではこのような協力的な行動を戦術という。

2.2 マーキング

プレイヤーは戦術行動を行いたいときに、自分の意図をマーキングを用いてチームメンバーに伝達する。他のプレイヤーはマーキングを出したプレイヤーの意図に合わせて適切な行動を取る。プレイヤーが特定のキーを押しながら、マップにマウスでクリックすることでその位置にマーキングを出すことができる。数秒間経過したら、マーキングが消える。

マーキングが表したプレイヤーの意図はその位置によって簡単に a. プレイヤー自身のキャラクタの近く、b. 味方キャラクタの近く、c. 敵キャラクタの近く、d. 何もない所の四種類に分けられる。a の意図はプレイヤーキャラクタを援護するか助けるかの二つで、b の意図は味方キャラクタ現在の行動をやめるか周辺を注意するかの二つで、c の意図は敵キャラクタを倒すことで、d の意図はその場所を偵察

することである。

3. 関連研究

Magnusson らは RTS ゲームの StarCraft においてプレイヤーとコミュニケーションができて、プレイヤーからもコントロールできるようなチームメイトボットを提案した[4]。そして、実験を通してプレイヤーはどのようなボットと一緒にプレイの方が面白いかを検討した。結果として、コミュニケーションができるボットの方がコミュニケーションできないボットより面白いことが明らかになった。また、経験者のプレイヤーはコントロールできないボットよりできるボットとプレイしたいことに対して、新人プレイヤーはコントロールできないボットの方とプレイしたいことがわかった。

Tan らは FPS ゲームにおいてプレイヤーのステータスに合わせて行動を取るようなゲームエージェントを提案した[5]。プレイヤーが五つのチームメイト NPC と一緒にゾンビを倒すようなゲームにおいて、NPC の全部可能な行動は事前設定しており、いつプレイヤーを支援するかをニューラルネットワークによって行動選択を行う。同じプレイヤーの行動に適応する AI 研究として、Abraham らでは動的にプレイヤーの行動やニーズに応じてプレイヤーを支援するようなチームメイト AI を提案した[6]。

これらの研究と違って、本稿の提案内容は AI エージェントとプレイヤー間の戦術的行動に注目している。プレイヤーがマーキングを出したら、AI エージェントがマーキングの意図に合わせて戦術的行動を取って、プレイヤーと協力する。

4. 提案手法

チームメイト AI エージェントがプレイヤーに協力するには、ゲーム場面を正確に認識し、実行可能な戦術を判別し、最も成功率の高い戦術を選択することが必要である。そのため、本手法ではゲーム場面をドロネー図で表し、部分グラフマッチングで実行可能な戦術を検索する。次に、時相を考慮した戦闘シミュレーションによって最も良い結果が出せる戦術を選択する。

4.1 ドロネー図によるゲーム場面認識

MOBA においてゲーム場面というのは各キャラクタの配置位置、ステータス (HP, MP, 攻撃力など)、スキル情報によって構成されている。ステータスやスキル情報などの単純な数値データと違って、キャラクタの配置位置や周囲の地形環境などの人間プレイヤーが目視で得られる情報を認識するのは AI エージェントにとって容易ではない。そのため、Perkins はボロノイ図を用いた AI エージェントが地形を認識する手法を提案し、実験でこの手法の有効性を実証した[7]。

本手法ではボロノイ図の相対であるドロネー図で各キャラクタの配置位置を記述する。図 2 のように、一つの母点の一つのキャラクタで、その位置はキャラクタの位置を

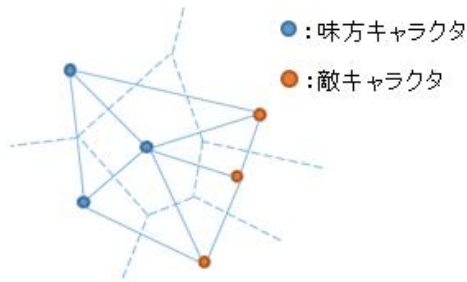


図 2 ドロネー図による配置位置の記述

反映している。母点間のドロネー図ではキャラクター間の距離で、母点の度数はこのキャラクターの隣接キャラクター数である。

キャラクターの配置位置のドロネー図を計算するにはまず各キャラクターを母点としてボロノイ図を計算する。そして、ボロノイ図からドロネー図を生成する。

4.2 部分グラフ同型判定による戦術選択

MOBA において、ゲーム中の戦術行動はキャラクターの配置位置次第で決められる場合が非常に多い。また、キャラクターの位置はずっと変化していて、プレイヤーがマーキングを出したら、AI エージェントが速やかにその場の配置位置によって適切な戦術を見つける必要がある。そこで、本手法では適用可能な戦術を部分グラフ同型判定で探す。そのため、各戦術に対してあらかじめ適用図と配置図を設定する。適用図とはある戦術の参加キャラクターを確定するとき、キャラクターが満たすべき隣接関係を表すグラフであり、配置図とは戦術を実行するときに参加キャラクターの配置位置を表すグラフである。一つの戦術に対して、参加キャラクター数に応じて複数の適用図が存在し、一つの適用図に対して一つの配置図がある。

部分グラフ同型判定とは二つのグラフが与えられた時、一方のグラフがもう一方のグラフに含まれるかどうかを判定する手法である。部分グラフ同型判定問題は NP 完全であり、効率的にこの問題を解くため Ullmann の提案手法[8]、Nauty 法[9]、VF 法[10]などが提案された。しかし、戦術選択の時に扱うキャラクター配置位置のドロネー図においてその頂点数は多くても 10 個なので、手法の効率にこだわる必要は無い。本稿では VF 法を用いた。

戦術選択する時、まずマーキングの種類によって戦術の数を一定程度に絞り、全体の配置位置ドロネー図から各戦術の適用図があるかどうかを探索し、対応している部分グラフを出力する。そして、出力された部分グラフの中からマーキングの目標キャラクターが入っている部分グラフだけを最後の結果として出力する。

つまり、この処理によって現在の配置位置のドロネー図において実行可能な戦術と戦術の参加キャラクターのリストを得ることができる。

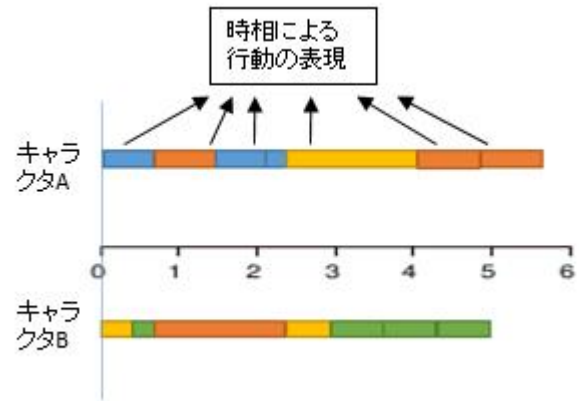


図 3 戦闘シミュレーション

4.3 時相を考慮した戦闘シミュレーション

前節の手法でキャラクターの配置位置の情報から複数の実行可能な戦術が得られた。本節ではキャラクターのステータスやスキル情報を元に各戦術に対して戦闘シミュレーションを行い、最も良い結果の戦術を選択する手法について説明する。

ゲームの戦術の評価について、M. Chung らは capture-the-flag というゲームにおいてパラメータ化したプランをモンテカルロ法によって評価する手法を提案した[11]。また、Sailer らはナッシュ均衡を取り入れた戦略シミュレーションの手法を提案した[12]。しかしリアルタイムのゲームとして、シミュレーションの結果をより正確に計算するには時間の要素を考慮する必要があると考えられる。特に MOBA では一つの行動（攻撃、スキルを使うなど）を開始したとしても、必要な時間が経過するまで行動の結果（ダメージなど）が反映されないという特徴がある。そのため、本手法では個々の行動に対し、時相の概念を追加する。時相とは一つの行動が行動開始から結果が反映されるまで必要な経過時間である。

戦闘シミュレーションの目的はある戦術がマーキングの目的（敵を倒す、味方を守るなど）を達成できるかどうかを評価することである。MOBA のキャラクターの最も基本的な行動は移動、普通攻撃、スキル使用の三つである。しかし、MOBA において、キャラクターの移動をシミュレーションするのは非常に難しいので、普通攻撃とスキルの使用だけをシミュレーションする。

戦闘シミュレーションの方法は図 3 のように、戦闘開始したら各キャラクターは一定の方法で次の行動を選択する。キャラクター数は三人以上の時、行動結果の反映対象は目標選択によって決められる。そして、最初の行動が終わったらずぐ次の行動を選択する。時相の短い行動を選択したキャラクターは先に次の行動を選択することになる。また、相手の行動を阻止するような行動をした場合、相手の実行中の行動が中断され、そして一定時間内に行動選択できなくなる。このように 6 秒間シミュレーションし続ける。6 秒と設定した理由はこの時間内で一つのキャラクターは大体す

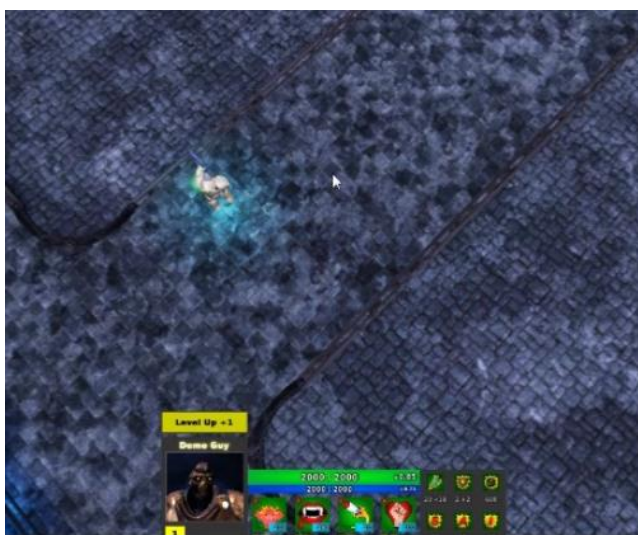


図 4 Simple MOBA

すべてのスキルを使い終わって、残るのは普通行動だけのためである。もし時間内に敵の HP は 0 になったら、敵を倒す目的が達成できると判断する。戦術を選択する時は目的が達成できる戦術のなかから味方キャラクターの残留 HP が最も多い方を選択する。そして、戦術に参加するキャラクターが戦術の配置図のように目的へ移動し、戦術を実行する。

5. 評価実験

5.1 Simple MOBA

市販の MOBA ゲームは設定が複雑なほか、ゲーム AI を変更することはできないので、本稿の提案手法をテストするため図 4 のような Simple MOBA を作った。

Simple MOBA はゲームのキャラクター数を十体から六体に減らした。お金や経験値及びレベル、アイテムなどの本手法とあまり関係のないゲーム要素は省いた。全部のキャラクターはゲームの最初から四つのスキルをもっており、攻撃力や防御力などのステータスがキャラクターによって少し異なっている。プレイヤーは Ctrl キーを押しながら、マウスでマップをクリックするとマーキングを出すことができる。

5.2 実験

実験では提案手法のゲーム AI とプレイヤーのマーキングを無視または簡単に行動するゲーム AI を実装した Simple MOBA を被験者にプレイしてもらい、アンケートを通して違う AI エージェントの行動がゲームの楽しさとの関連性について評価する。

実験には二つの部分に分ける。最初の部分は練習シーンでプレイヤーに Simple MOBA のゲームプレイに慣れてもらう。次はメインの部分で、3 対 3 のシーンにおいてプレイヤーは二回のゲームで違う AI と一緒にプレイする。AI の順番はランダムに決められる。

6. おわりに

本稿では MOBA においてプレイヤーのマーキングに対して積極的に戦術を組み立て、プレイヤーに協力する AI エージェントを提案した。グラフ理論による戦術選択と時相理論による戦闘シミュレーションによって AI エージェントは違うゲーム場面において適切な戦術を選択してプレイヤーに協力する事ができた。そして、AI エージェントの行動がゲームの楽しさとの関連性を調べるための実験プランについて説明した。

今後ではより複雑なゲーム場面における AI エージェントの意思決定と評価実験手法の改善について検討を進めていきたい。

参考文献

- [1] Merrill, M. (2012). "League of Legends Community Infographic." 11 October 2012, <http://na.leagueoflegends.com/news/league-legends-community-infographic>
- [2] Ferrari, Simon. "From Generative to Conventional Play: MOBA and League of Legends." DiGRA, 2013.
- [3] Soni, Bhuvan, and Philip Hingston. "Bots trained to play like a human are more fun." Neural Networks, 2008.
- [4] Magnusson, Matteus M., and Suresh K. Balsasubramaniyan. A Communicating and Controllable Teammate Bot for RTS Games. Diss. Master's thesis, School of Computing, Blekinge Institute of Technology, 2012.
- [5] Tan, Chek Tien, and Ho-lun Cheng. "Personality-based Adaptation for Teamwork in Game Agents." AIIDE. 2007.
- [6] Abraham, Aswin Thomas, and Kevin McGee. "AI for dynamic team-mate adaptation in games." Computational Intelligence and Games (CIG), 2010 IEEE Symposium on. IEEE, 2010.
- [7] Perkins Luke. "Terrain Analysis in Real-Time Strategy Games: An Integrated Approach to Choke Point Detection and Region Decomposition." AIIDE 10 (2010): 168-173.
- [8] J.R. Ullmann. "An algorithm for subgraph isomorphism." Journal of the ACM (JACM) 23.1 (1976): 31-42.
- [9] McKay, Brendan D. Practical graph isomorphism. Department of Computer Science, Vanderbilt University, 1981.
- [10] Cordella, Luigi P., et al. "A (sub) graph isomorphism algorithm for matching large graphs." Pattern Analysis and Machine Intelligence, IEEE Transactions on 26.10 (2004): 1367-1372.
- [11] M. Chung, M. Buro, and J. Schaeffer, "Monte Carlo Planning in RTS Games," in Proceedings of the 2005 IEEE Symposium on Computational Intelligence in Games. New York: IEEE Press, 2005, pp.117-124.
- [12] Sailer, Franisek, Michael Buro, and Marc Lanctot. "Adversarial planning through strategy simulation." Computational Intelligence and Games, 2007. CIG 2007. IEEE Symposium on. IEEE, 2007.