

StarCraft AIへの隊列導入

鎌田徹朗^{1,a)} 橋本剛¹ 高野誠也¹

概要: StarCraft はリアルタイムストラテジー (RTS) ゲームの中でも特に人気のシリーズであり、多数のプロプレイヤーがいる。AI 同士で対戦を行う大会が開催され、StarCraft AI の開発は徐々に盛んになってきているが、まだプロに勝てるほど強くない。2012 年と 2014 年に開催された大会で上位入賞 AI 対プロの対戦が行われたが、結果は AI 側の 12 戦全敗であった。この対戦を分析すると、ユニットの移動時の配置に大きな問題があることがわかった。本研究では StarCraft AI に隊列の概念導入を提案し、実装を行い実験により有効性を調べる。

キーワード: ゲーム AI, リアルタイムストラテジー, StarCraft, 隊列, BWAPI

Introducing ranks of troops to StarCraft AI

Abstract: StarCraft is a popular series of Real Time Strategy (RTS) games with which many professional players play. Tournaments for AI are often held and development of StarCraft AI gradually becomes popular, however it can not defeat professional players yet. Higher-ranking prize StarCraft AI programs on 2012 and 2014 tournaments played with a professional player and the result is zero win twelve loses among twelve games from AI. Analysis of these games makes aware that disposition of units while moving has great difficulty. This research introduces a concept of ranks of troops to StarCraft AI, implements it and examines its effectiveness by experiments.

Keywords: Game AI, Real Time Strategy, StarCraft, Ranks of troops, BWAPI

1. はじめに

近年、チェスをはじめとするボードゲーム AI はコンピュータの処理性能の向上とアルゴリズムの発達によりプロに勝てるまで進化した。これら零和有限確定情報ゲームの AI が高いレベルに進化した現在では、不完全情報ゲームの研究が注目されている。その中でも研究が盛んに行われているのは、リアルタイムストラテジー (以下、RTS) ゲームである。RTS は世界で流行しており、特に 1998 年に Blizzard Entertainment より発売された StarCraft シリーズは多くのプロがいる人気のシリーズであり、大会が盛んに開催されている。(図 1)

StarCraft は、敵のユニット (兵士) を全滅させることを勝利条件とする対戦ゲームである。プレイヤーは異なる特性を持つ複数のユニットに命令を与え、リアルタイムに進行する時間の中で、戦場の探索や自軍の強化、敵ユニッ

トへの攻撃といった複数の作業を並行して効率よく行わなければならない。

StarCraft は個人での AI 開発が可能であり、AI 同士で対戦を行う大会が開催されている。StarCraft AI の開発は徐々に盛んになってきているが、まだプロプレイヤーに勝てるほど強くない。StarCraft AI とプロプレイヤーの対戦を見て分析した結果、StarCraft AI には多くの弱点があるが、特にユニット移動時の配置に大きな問題があると思われる。本研究では、この問題を解決するため StarCraft AI に隊列の概念を提案及び実装し、StarCraft AI における隊列の有用性を調べる。

2. StarCraft とは

本章では StarCraft の概要と主な StarCraft AI を説明する。

2.1 StarCraft 概要

StarCraft は、敵のユニットを全滅させることを勝利条

¹ 松江工業高等専門学校

^{a)} j1010@matsue-ct.jp



図 1 StarCraft eSportsLeague (WallPlayed.org より引用)

件としている。ゲームの進行と共に味方ユニットは増え、プレイヤーは最大 200 体ものユニットに命令を出すことになる。また、StarCraft の戦場となるマップは広く、大会で使用された Benzene という 2 人対戦用マップでは、ゲーム中にプレイヤーが見ている画面は全体のおよそ 70 分の 1 程度にしか過ぎない。このように StarCraft はとても規模の大きいゲームである。

ゲームの開始時にプレイヤーは自分が操作するユニットの種族を選ぶ。種族は 3 種類あり、それぞれ Terran, Zerg, Protoss という。各種族の特徴を以下に示す。

- Terran (テラン) Terran は人間に近い見た目をしている。施設建設の際に他の種族と異なり制限が無く、指定した位置に建設出来る。更に、建設後に施設を浮遊させて移動することも可能である。また、Terran のユニットや施設はダメージを受けた際に修理が可能である。3 種族中 Terran のユニットは最も弱いが、核ミサイルを代表とする遠隔攻撃兵器を利用した攻撃が強力。
- Zerg (ザーグ) Zerg は醜悪な化け物の姿をしている。施設は Worker ユニットの Drone が変態をすることで完成する。Zerg の施設は Creep という組織が覆う地面の上のみ建設が可能で、他種族は Creep の上に施設を建設出来ない。各ユニットの体力は低いが攻撃力は高く、ユニット 1 体当たりの生産に必要な資源が少ないため短期決戦を得意とする種族である。
- Protoss (プロトス) Protoss は所謂宇宙人のような姿をしている。Protoss のユニットや施設は体力とは別にシールドを持ち、攻撃を受けてシールドが破られることで初めてダメージを受ける。施設は位置を指定すると他の場所から転送されてくるため、Protoss の Worker ユニットの完成を待たずに他の作業を行うことが出来る。他の種族のユニットと比較すると Protoss のユニットは高性能だが、ユニット 1 体当たりの生産コストが高いという欠点がある。

これらの 3 種族の均衡がとれているため StarCraft は高い人気があるといわれている。プレイヤーは選択した種族の



図 2 探索前のマップ



図 3 探索後のマップ

ユニットに命令を出し、大別して次の 4 つの作業を行う。

2.1.1 探索

StarCraft のゲーム初期段階では、図 2 のようにマップ(戦場)は暗闇に包まれている。後述する資源回収や敵への攻撃をするに当たり、戦場の状況を知ることは極めて重要になる。各ユニットには見える範囲(視界)が存在し、プレイヤーはユニットを探索に送ることでそのユニットの視界の範囲内の戦場の情報を得ることが出来る。(図 3)

また、StarCraft のマップ情報は一度探索した場所であってもユニットの視界の範囲外であれば見ることが出来ない。マップの状況は常に化する。最新のマップ情報を得るために、探索は出来る限り多く出来ると良い。

2.1.2 資源回収

資源は図 4 に示す Mineral (ミネラル) と Gas (ガス) の 2 種類ある。どちらの資源も自軍を強化するために必要である。資源はマップ上に点在しており、各種族の Worker と呼ばれるユニットで回収を行う。ミネラル、ガス共に有限であるため、相手プレイヤーより多く資源を確保することで有利にゲームを進めることが出来る。

2.1.3 研究命令・建設命令

Worker ユニットの建設命令を与えることで、資源を消費して施設の建設が出来る。施設はユニットと同様に多種多



図 4 資源 (左: Gas, 右: Mineral)

様な特徴を持ち、ユニットの移動速度を上昇させる等の自軍が有利になる効果がある。一部の施設には研究命令を与えることでユニットの装備や能力の強化が出来る。研究命令により得る能力には、特定のユニットが一定時間透明になり、敵に見つからなくなる能力など強力なものが多い。研究命令により強化した装備や能力はゲーム終了時まで有効である。相手プレイヤーよりも多く研究命令・建設命令を与え、ユニットの装備や能力の強化をすると有利にゲームを進めることが出来る。

2.1.4 敵への攻撃

2.2 StarCraft AI

BWAPI[4] や BWSAL[5] が開発されたおかげで StarCraft AI は個人での開発が可能になった。

BWAPI は StarCraft Broodwar の AI 開発用 C++ フレームワークとして開発され、2010 年 5 月に BWAPI Beta2.7.2 が公開された。BWAPI は JNI-BWAPI として Java による開発も進められている。また、現在では開発が停止しているが BWAPI は PHP や Python での開発も行われていた。2015 年 1 月には、BWAPI4.1.0Beta が公開されており、42 名が参加するプロジェクトとなっている。BWAPI は図 5 に示すクラス相関図から分かるように、とても規模が大きい。

最近では BWAPI, BWSAL などの StarCraft AI 開発支援の進化、StarCraft AI 同士での対戦大会の開催も相まって StarCraft AI の開発は盛んになりつつある。

BWSAL は BWAPI の機能をより便利にするために開発されたライブラリである。現在 BWSAL は、BWAPI3.6.1 に対応した BWSAL0.9.12 まで公開されている。BWSAL には BWSAL を用いた StarCraft AI の実装例として BasicAIModule が付属している。BasicAIModule は StarCraft の基本的な部分を実装した StarCraft AI である。

2010 年より University of Alberta (カナダ) の David Churchill[1] らが StarCraft AI Competition[2] を毎年開催している。2012 年と 2014 年に開催された StarCraft AI Competition では上位に入賞した StarCraft AI とドイツのプロ StarCraft プレイヤー Bakuryu[3] との対戦が行われた。対戦の結果は AI 側の 0 勝 12 敗と惨敗であった。StarCraft AI はまだプロのレベルには遠い。



図 6 複数ユニットの移動



図 7 ユニット移動の弱点

3. 弱点の分析

2.2 章で紹介した、StarCraft AI とプロプレイヤー Bakuryu との対戦の動画を 12 戦分見て分析を行った結果、StarCraft AI には誘導行動に弱い、ユニットの配分が甘いなど多くの弱点が見付かった。その中でも特にユニットの移動に問題があると考えた。

StarCraft において複数のユニットに対して同時に移動命令を与えた場合、ゲームの仕様上図 6 に示すように 1 列になって移動することが多い。このため、図 7 のように移動先に敵ユニットが待機していた場合、1 番と 2 番のユニットが包囲され敵から集中攻撃を受けている際に、3 番から 8 番のような隊列後方のユニットが戦闘に参加出来ないという状態が生じる。結果として 1 番と 2 番のユニットが倒され、次は 3 番と 4 番のユニットが包囲され集中攻撃を受けるといった繰り返しになってしまい、大きな被害へと繋がる。このような被害を実際のプレイで軽減するにはユニットを 1 列にならないように移動命令を出せば良いのだが、そのためには個々のユニットに命令を出す必要がある。具体的には図 8 のように移動させたいユニットが 11 体いる場合、まずユニット A を選択し、移動命令を出す。次にユニット B を選択し、移動命令を出すという作業を移動させたい 11 体のユニット全てに適用しなければならない。こ

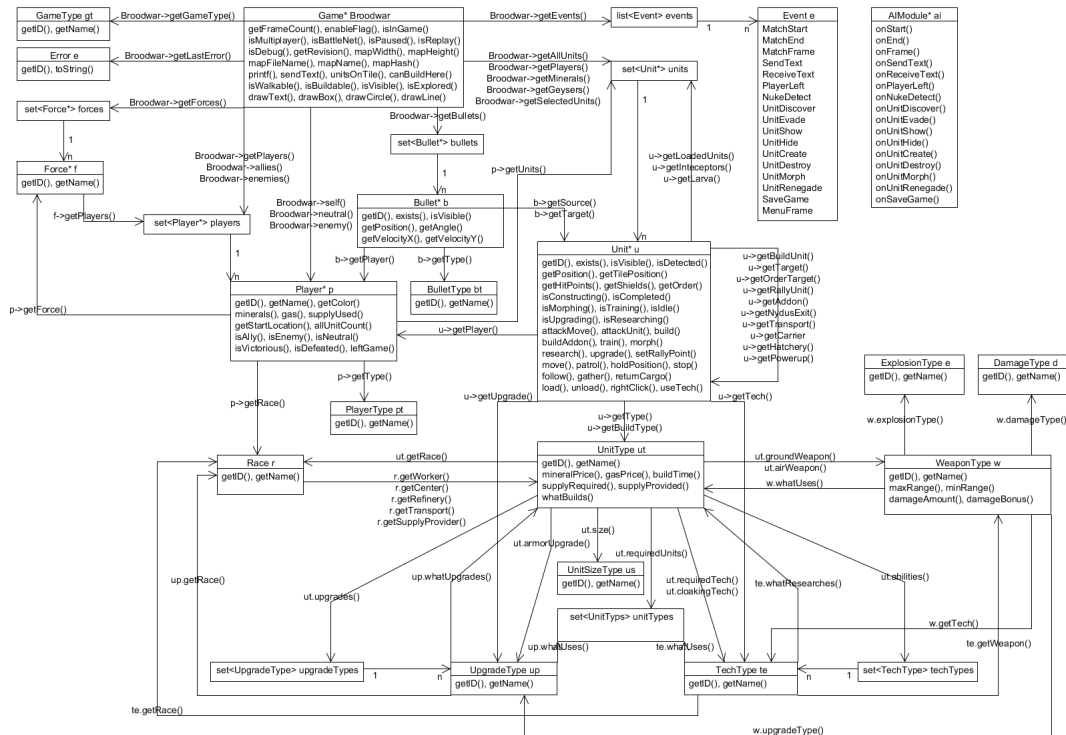


図 5 BWAPI クラス相関図



図 8 ユニット移動の改善

これらの移動命令を短時間でユニットに与えないと、11体のユニットが散らばって移動してしまい1列で移動するよりも多くの被害が出る。この移動命令は短時間に多くの操作が要求され、人間には難しい操作である。AIであればこの移動命令が可能であると考えた。本研究では、ユニットが1列となって移動する弱点の解決手法として StarCraft AI への隊列導入を提案する。

4. 隊列の提案

4.1 隊列と陣形

隊列とは、軍隊などで隊を組んで作った列のことで、

StarCraft AI に隊列を導入することを考える。どのような隊列が有効か考えることは陣形を考えることに近い。合戦の際に軍勢が敷く隊形や方法を陣形または陣立てという。日本の戦争で用いられた陣形の中では戦国時代武田家の八陣形が有名である。本研究では、まず簡単な隊列を StarCraft AI に導入しようと考え、基本陣形の1つである横陣隊列を移動時に用いる。

4.2 横陣隊列

横陣隊列 (以下、横陣) は図 9 に示すように、敵の居る方向に対して兵士を垂直方向に広げて配置する隊列である。横陣は攻撃範囲が広いという利点があるが、同時に隊列が分断されやすいという欠点も有する。本研究では StarCraft AI への横陣の導入を行い、有用であるかどうかを検討する。次章にて横陣の実装について述べる。

5. 実装

本研究では、2.2 章で説明した BWSAL0.9.12 に付属する BasicAIModule に横陣で移動する機能を追加した。

5.1 ユニットのグルーピング

第 2 章で説明した通り、StarCraft は規模の大きいゲームである。マップの各地でユニットの戦闘が起こることがある。そのため、ユニットは全体で制御するよりも小規模なグループを作り制御したほうが良い。本研究では BWSAL



図 9 横陣隊列

に横陣を導入する準備としてユニットのグルーピング機能を実装した。StarCraft では Woker ユニットなどの非戦闘用ユニットと戦闘用ユニットの 2 種類のユニットが存在する。グルーピングによって、これらを差別化し攻撃を制御する。各グループに必ず 1 体のユニットを代表ユニットとして設定し、このユニットを基準に陣形を構成する。

ユニットのグルーピングの方法は、同じ視界内にいるユニットで行うのが妥当だと考えられるが、本研究では簡略化のためユニットの種類ごとにグループ分けを行う。StarCraft でプレイヤーは同時に最大 12 体まで選択出来ること、奇数体のユニットだと横陣が制御しやすいという 2 つの理由より、1 グループ当たりの最大ユニット数を 11 体とした。

5.2 横陣の実装

ユニットのグルーピングによって出来たユニット 11 体のグループ内で横陣を構成する。グループの代表ユニットを中心として左右 5 体ずつユニットを配置する。移動の際は陣形が乱れないように代表ユニットの移動速度を下げる工夫を行った。そのため横陣による移動は通常の移動より遅くなる。また、移動に横陣を用いる AI (以下、横陣 AI) との比較対戦のために隊列の考慮なしに移動をする AI (以下、隊列なし AI) の作成を行った。なお、両 AI は移動戦術以外全く同じ戦術を採るように作成した。

6. 実験

6.1 実験方法

StarCraft Broodwar1.16 を専用のソフト Chaoslauncher でハックし、自作した AI で対戦が出来るようにした。この環境で以下の実験を行った。2 台の PC 間で、横陣 AI と隊列なし AI の LAN (UDP) 対戦を 100 回行った。戦闘は公平を規すため双方 11 体の Protoss Zealot を使用し、勝敗を一方のユニットの全滅をもって判定した。Protoss Zealot は地上で近接攻撃をする Protoss の最も基本的な戦

表 1 実験結果

	横陣 AI	隊列なし AI
勝利数 (100 戦)	85 回	15 回
倒した平均ユニット数 (11 体)	10.52 体	7.36 体

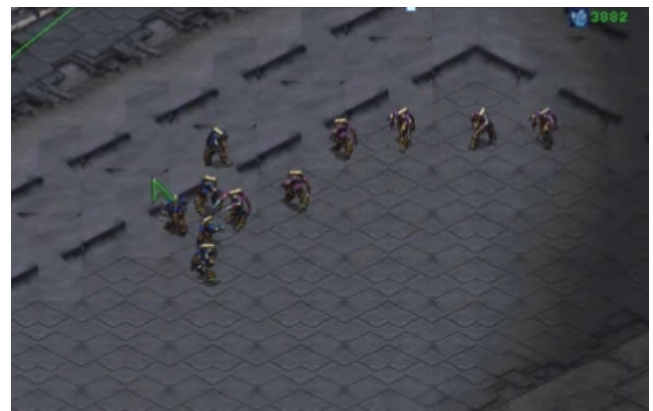


図 10 先頭ユニットへの集中攻撃

闘用ユニットである。

6.2 実験結果

実験の結果は表 1 に示すように横陣 AI の 85 勝 15 敗であった。戦闘 1 回当たり平均 10.52 体の敵ユニットを倒し、逆に、平均 7.36 体の味方ユニットを倒された。以上のことより、ユニットの戦闘において陣形を考慮することは明らかに有用だといえる。

7. 考察

7.1 横陣 AI の勝因

図 10 に横陣 AI と隊列なし AI の対戦実験の様子を示す。図 10 左側の肩が青色のユニットが横陣 AI、右側の肩が紫色のユニットが隊列なし AI である。実験で横陣 AI が大勝した要因として、図 6 で示したゲームの仕様上の弱点である、隊列なし AI が 1 列になって移動していたことが考えられた。図 10 より分かるように、横陣 AI はその列の先頭ユニットに対して集中攻撃をしていたが、第 3 章で弱点分析した結果とは違い、隊列後方のユニットが到着する前に隊列先頭のユニットを倒すことが出来ていなかった。しかし、横陣 AI のユニットが隊列なし AI のユニットを包囲するように戦闘しており、多数対少数の状況を作ることに成功していた。このことが横陣 AI の大勝という実験結果に繋がったと思われる。

7.2 横陣の規模

StarCraft において敵ユニットを攻撃するにはある程度の距離まで接近しなければならない。横陣を大きく広げすぎると、横陣の端を構成するユニットは敵ユニットに攻撃が届かない。本研究で実装した横陣 AI ではグループに参加している 11 体のユニット全てを横並びにしていたため、

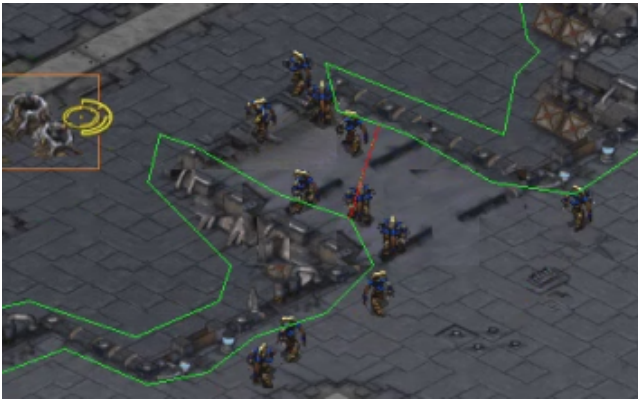


図 11 横陣崩壊

横陣端のユニットが戦闘の際に攻撃可能な位置まで移動することが多く見られた。横陣の横に並ぶユニット数を減らし、複数列で実装した方が効果的だと思われる。

7.3 地形の考慮

本研究で実装した横陣 AI では、地形の影響を考慮に入れていないために、図 11 のように狭い通路を通過する際に横陣が乱れてしまう。横陣 AI を地形に対応した AI に改良する必要がある。実験で横陣 AI が 15 敗した原因は、隊列なし AI がこの欠点を突いたためだと思われる。

8. 今後の課題

本研究では横陣の実装を行ったが、武田家の基本八陣形のように戦争に用いられた陣形も StarCraft AI へ導入を行いたい。また、本研究で挙げた移動に関する弱点以外にも StarCraft AI にはまだ多くの弱点がある。例えば、敵の誘導行動に気付けない点や、ユニットの配分の無駄などが挙げられる。これらの弱点も積極的に改善したい。BWSAL で作成した BasicAIModule はゲームのごく基本的な部分しか実装が出来ていない。今後は実践的な StarCraft AI に隊列や陣形の導入を行い、その AI で 8 月に University of Alberta で開催される大会に参加し、優勝したい。

参考文献

- [1] David Churchill Computing Science, University of Alberta <http://webdocs.cs.ualberta.ca/~cdavid/>
- [2] David Churchill, Michael Buro : 2014 StarCraft AI Competition <http://webdocs.cs.ualberta.ca/~cdavid/StarCraftAIcomp/>
- [3] Liquipedia The StarCraft Encyclopedia <http://wiki.teamliquid.net/starcraft/Bakuryu>
- [4] bwapi blog <http://bwapi.blogspot.jp/>
- [5] bwsal Standard Add-on Library for BWAPI <http://code.google.com/p/bwsal/>