ターン制ストラテジーのための
効率的な探索アルゴリズムの構築

北陸先端科学技術大学院大学
情報科学研究科

藤木 翼

平成 28 年 3 月

修 士 論 文

ターン制ストラテジーのための効率的な探索アルゴリズムの構築

1310062　藤木 翼

主指導教員　池田　心
審査委員主査　池田　心
審査委員　飯田　弘之
長谷川　忍

北陸先端科学技術大学院大学

情報科学研究科

平成 28 年 2 月

概 要

これまで，チェスや将棋，囲碁などの古典的なゲームのコンピュータプレイヤに関する研究は幅広く行われてきた． これらの成果として，チェスではDEEPBLUEが人間のチャンピオンに勝利するといった結果を残している．将棋では，ボナンザ法や実現確率探索などの登場により，現在ではプロ棋士に迫る性能を発揮している．囲碁においても2016年にGoogleのALPHAGoがディープラーニングやモンテカルロ木探索を用いることで，人間のプロ棋士に勝利している．これらのゲームのコンピュータプレイヤの“強さ”は一般的な人間プレイヤにとってほぼ十分である．

しかし，一回の手番に複数の駒を動かせる要素「複数着手性」を持つゲームは広く遊ばれているにも関わらず，学術的研究は少数が行われているのみである． 複数着手性を持つゲームジャンルとしてはターン制ストラテジー，シミュレーションロールプレイング，リアルタイムストラテジーなどが挙げられる．本稿では，それらの中から，比較的付加要素が少なくシンプルなターン制ストラテジーを対象とする．既存のターン制ストラテジーとしてはCivilization，大戦略，ファミコンウォーズなどが挙げられるが，これらにおいてコンピュータプレイヤは人間の中級者とハンデなしで戦える強さにはほど遠い．ゲーム内ではハンデでその弱さを補っているが，対等に戦いたいという要求には応えられていない． そこで本稿では，こういったゲームで自然さや楽しさといったものの前にまず“強い”コンピュータプレイヤを作成することを目的とする．

本稿ではTUBSTAPと呼ばれる学術研究を目的としたターン制ストラテジーゲームを対象ゲームとした．既存ターン制ストラテジーの要素は内政要素やキャラクタの成長要素など数多い．TUBSTAPはその中でも重要な要素である複数着手性に注目し，不完全情報性や占領・生産などの要素を排除した二人零和有限確定完全情報のターン制ストラテジーゲームである．将棋などと比べると複数着手性を持つ点で複雑ではあるが，既存のターン制ストラテジーに比べると非常にシンプルなゲームとなっている．複数着手性を有する事と最低限の要素のみを持つことから，純粋に複数着手性を持つゲームにおける探索アルゴリズムの構築に専念することができる．また，コンピュータプレイヤが作りやすい環境が用意されているされているなど利点が多い事からTUBSTAPを対象ゲームとした．

将棋などと異なり，ターン制ストラテジーでは全ての駒を任意の順序で動かすことができ，合法手数が膨大になることが探索の障害となる．探索時間が限られていることを考えれば，浅い探索にするか疎らな探索にするかどちらかを受け入れざるを得ない．安易に全探索が行えない一方で，局面に応じて最善手を見つけやすい深さが大きく異なる．一つは攻撃的な手が必要とされる局面であり，この局面では自身の手までを浅くとも綿密に探索しなければ最善手が見つかりにくい．もう一つは防御的な手が必要とされる局面であり，この局面では相手のターンまでを含め疎らであっても深く探索しなければ最善手が見つからない．この二種類の局面が存在するため，アルゴリズムによっては得意不得意の局面が大きく異なる．

ターン制ストラテジーにおいて有効であると考えられている既存手法は，UCT や攻撃行動探索などが挙げられる．UCT はモンテカルロ木探索の一種であり，探索の指標に UCB1 値を用いる．囲碁や将棋など様々なゲームでしばしば用いられる手法であり，ターン制ストラテジーであっても有望な手法である可能性が高い．また，攻撃行動探索は将棋などにおける駒を取る動作（攻撃行動）のみを探索するやや特殊な探索手法であり，攻撃行動のみであっても合法手が多いターン制ストラテジーに対して考えられた探索手法である．これらの既存手法は攻撃的な手を探索することが得意な一方で，防御的な手を有効に取れてはいない．

そこで本稿では防御的な手を重視した探索手法である深さ限定モンテカルロ探索：Depth-Limited Monte-Carlo（DLMC）を提案する．DLMC は合法手を全て探索せず，ランダムなサンプリングを用いる事で探索する広さを制限している．また状態評価関数とシミュレーションの打ち切りを用いることで相手のターンという特定の深さまでを探索させている．これらによって DLMC は防御的な手の探索にのみ特化している．本稿では，防御的な手を有効に取ることが出来るかを定量的に評価する実験として，特定の局面の初手のみを対象のアルゴリズムに動作させ．残り全てを対戦相手のアルゴリズムに動作させるという実験を行っている．この評価において UCT は勝率 52.3%ほどであったが，DLMC は 59.5%と性能が向上した．しかし，攻撃的な手を有効にとることができず，単純な性能評価では UCT よりも弱く，単独では十分な性能を発揮できなかった．

そこで，攻撃的な手を取ることができる UCT と防御的な手をとることができる DLMC を組み合わせる事で，それぞれの問題点を解消した手法を提案する．具体的にはそれぞれのアルゴリズムによる最善手をシミュレーションで評価し，より良い評価を得た手を採用するという非常に単純な手法である．この手法を用いる事で UCT_PW との対戦実験では勝率 59.5%と勝率が向上した．

目 次

第1章　はじめに

これまで，チェスや将棋，囲碁などの古典的なゲームのコンピュータプレイヤに関する研究は幅広く行われてきた．これらの成果として，1997年にチェスプログラムDEEPBLUE[1]が人間のチャンピオンに勝利するといった結果を残している．将棋では，ボナンザ法や実現確率探索などの登場により，現在ではプロ棋士に迫る性能を発揮しており[2]，囲碁においても2016年にGoogleのALPHAGo[3]がディープラーニングやモンテカルロ木探索を用いることで，人間のプロ棋士に勝つ目覚しい成果を挙げている．これらのゲームのコンピュータプレイヤの“強さ”は一般的な人間プレイヤにとってほぼ十分である．

しかし，一回の手番に複数の駒を動かせる要素「複数着手性」を持つゲームは広く遊ばれているにも関わらず，新しいゲームであることや，探索空間の大きさ，ルールの煩雑さなどもあって，学術的研究は少数が行われているのみである． 複数着手性を持つ既存ゲームとしては大戦略[4]，ファミコンウォーズ[5],Civilization[6]，などのターン制ストラテジーが挙げられるが，コンピュータプレイヤは人間の中級者とハンデなしで戦える強さにはほど遠い．ゲーム内ではハンデでその弱さを補っているが，対等に戦いたいという要求には応えられていない． そこで本稿では，こういったゲームで自然さや楽しさといったものの前にまず“強い”コンピュータプレイヤを作成することを目的とする．

本稿ではターン制ストラテジーの一つである「TUBSTAP」[7][8][9]を対象ゲームとした．ターン制ストラテジーでは局面に応じて，防御的な手，攻撃的な手それぞれが必要とされる場合があり，それぞれの手を見つける為に必要な探索範囲が異なる．UCT，攻撃行動探索などの既存手法の得意な局面は攻撃的手が必要とされる局面であり，防御的な手は十分に取れていない．そこで本稿では，ランダムなサンプリングや探索の打ち切りを組み合わせることで防御的な手を重視した探索手法であるDLMCを提案する．その結果として，既存手法とDLMCを組み合わせる事で防御的な手を取れないといった問題点を改善し，UCT_PWとの対戦実験では勝率59.5%と勝率が向上したことを示す．

本論分の構成は以下の通りである．まず2章では複数着手性のゲームの紹介，対象とするターン制ストラテジーについての解説，またターン制ストラテジーの持つ特性やルールについて説明をおこなう．3章では本稿で用いるアルゴリズムの評価方法の説明．4章はターン制ストラテジーに有用であると考えられる既存のアルゴリズムをそれぞれ紹介し，その上で性能を評価する．5章はそれらを踏まえた提案手法の説明と性能評価．6章では既存手法と提案手法を組み合わせた手法の提案と性能評価をおこなう．7章はまとめである．

第2章　複数着手性を持つゲーム

本章では，複数着手性を持つゲームの代表例や，対象とするターン制ストラテジーについての解説，またターン制ストラテジーの持つ特性やルールについて説明する．尚，本章において挙げるゲームの詳細やルールなどは我々が提案したターン制ストラテジーTUBSTAPを初めて紹介した論文「学術研究用プラットフォームとしての大戦略系ゲームのルール提案」[8] を引用または改編したものである．

2.1　既存の複数着手性を持つゲーム

本節では，古典的戦略ゲームと言える将棋やチェスから見た場合に，複数着手性を持つ他のゲームがどのような要素を持つのか，また研究対象として適しているかを簡単に考察する．

ターン制ストラテジー

ターン制ストラテジーはチェスや将棋のように，盤面（マップ）上に配置された駒（戦車や戦闘機などの兵器．ユニット）を手番に動かし，それを勝敗条件が満たされるまで繰り返すゲームであるが，複数着手性などさまざまなルールが追加されている．単純に駒同士を戦わせるだけではなく，駒を生産する資源の確保，生産拠点の設置などといった内政要素を含む場合もあり，多くの要素を合わせ持ったゲームとなる場合が多い．内政要素を持たない代表的なゲームとしては大戦略 [4]（図 2.1），ファミコンウォーズ [5]，内政要素を持つ代表的なゲームとしては Civilization[6]（図 2.2）があげられる．

これらのゲームは複数着手性のあるゲームの学術的ベンチマークとして利用可能だが，ルールが複雑な点・複数着手性を持つことによって合法手が膨大となる点などがコンピュータプレイヤを作成する上で問題となる．既存研究としては内政要素に着目したものが多く [11][12]，純粋に複数着手を持つボードゲームといった要素に着目した研究はあまり多くない．

図 2.1: 大戦略のプレイ画面
（大戦略 PERFECT3.0 より引用 [4]）

図 2.2: Civilization のプレイ画面
（CIVILIZATION V より引用 [6]）

Arimaa

Arimaa[13]（図 2.3）は，チェスの盤，駒を使用して遊べる二人対戦用のボードゲームで，チェスと同様の二人零和確定完全情報ゲームであるが,「同一ターン内で 4 手まで自分の駒を動かすことが可能」といった複数着手性の要素を持つ．このため，ターン制ストラテジーと同様に合法手数が膨大になるため，ナイーブな木探索アルゴリズムの適用は困難である．本ゲームは複数着手性のあるゲームの学術的ベンチマークとして利用可能だが [14]，プレイヤ数が少なく棋譜収集や評価が困難という問題もある．

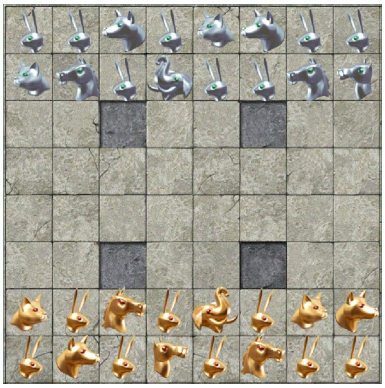


図 2.3: Arimaa のプレイ画面
（ArimaaHP より引用 [13]）

Simulation Role Playing （SRPG）

シミュレーションロールプレイングは戦略ゲームの要素と，RPG の要素（物語性・成長など）を併せ持ったゲームジャンルである．代表的なものとして Final Fantasy Tactics[15] があり，国内 170 万本を売り上げており，他にもスーパーロボット対戦 [16]，ファイアーエムブレム [17] など人気シリーズは多い．しかし，思考ゲームの学術的ベンチマークとしては駒の成長要素や，戦闘における回避などの乱数要素，命中率の概念，複数の駒への同時攻撃など付加要素が多すぎるきらいがある．

Real Time Strategy （RTS）

StarCraft[18] や Age of Empire[19] を代表とする RTS は，PC の性能向上に伴い比較的新しく出来たジャンルである．RTS では，手番という概念は存在せず，プレイヤ達は駒にそれぞれ任意のタイミングで指示を出す．盤面にも離散化されたマスという概念はない．欧米では RTS は非常に人気であり，また学術研究も盛んに行われ，国際会議 IEEE-CIG では論文の数割が RTS に対するものである．プログラム開発のためのプラットフォーム [20] も公開されており，プログラミングコンテスト [21] も頻繁に開催されている．しかし，ルールが複雑・煩雑でまた思考時間がリアルタイム制のために限定されるなど，チェスや将棋からは一足とびになっている感が否めない．

対象ゲーム：TUBSTAP

既存ターン制ストラテジーの要素は内政要素やキャラクタの成長要素など数多い．TUBSTAP（図 2.4）[7][8] はその中でも重要な要素である複数着手性に注目し，不完全情報性や占領・生産などの要素を排除した二人零和有限確定完全情報のターン制ストラテジーゲームである．将棋などと比べると複数着手性を持つ点で複雑ではあるが，既存のターン制ストラテジーに比べると非常にシンプルなゲームとなっている．複数着手性を有する事と最低限の要素のみを持つことから，純粋に複数着手性を持つゲームにおける探索アルゴリズムの構築に専念することができる．また，コンピュータプレイヤが作りやすい環境が用意されているされている点から本稿では TUBSTAP を対象ゲームとした．

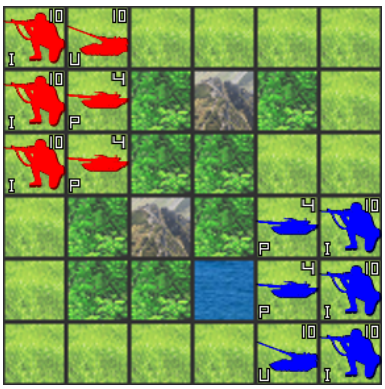


図 2.4: TUBSTAP のプレイ画面（赤先手）

2.2　TUBSTAP のルール

本節では，TUBSTAP における重要なルールを抜粋する．尚，紹介するこれらの要素は既存のターン制ストラテジーであれば殆どが持っている共通の要素である．

詳細なルール

R1. 盤面

将棋盤などと同じ，四角形マスからなる二次元の盤面を用いる．縦サイズ，横サイズは特に指定しない．

R2. プレイヤ数

2 人ゲームとする．それぞれのプレイヤの駒は赤（先手）と青（後手）で表す．

R3. 駒

戦闘機 (F)，攻撃機 (A)，戦車 (P)，対空戦車 (R)，歩兵 (I)，自走砲 (U) の 6 種類の駒（図 2.5）を用いる．これらの駒間には相性が存在する．相性は後述する HP に与えるダメージ量に影響する．

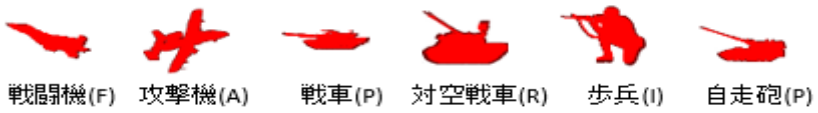


図 2.5: 駒の種類

R.4 マップ

将棋等では常に同じ盤面サイズ・駒の初期配置を用いるのに比べ，戦略ゲームでは通常，特徴の異なる複数の設定を用い，プレイ時にそれを選ぶ．一つの盤面サイズ，それぞれのマスの地形，用いる駒の種類と数と初期配置の組合せを「マップ」と呼ぶ．これらには非対称性が導入されていることも多い．図 2.4 は実際のゲームのプレイ画面で，駒の背景にある模様がそれぞれの地形を表している．

R5. 着手順

各手番では，プレイヤは全てのユニットを 1 回ずつ自由な順番で選んで行動させることができる．全てのユニットを行動させると，相手の手番となる．両者がそれぞれ手番を終えるまでを 1 ターンと呼ぶ（図 2.6）．ユニットが 1 回にできる行動は，「移動のみ行う」「移動して隣接攻撃する」「（隣接・遠距離）攻撃のみ行う」「何もしない」の 4 つである．

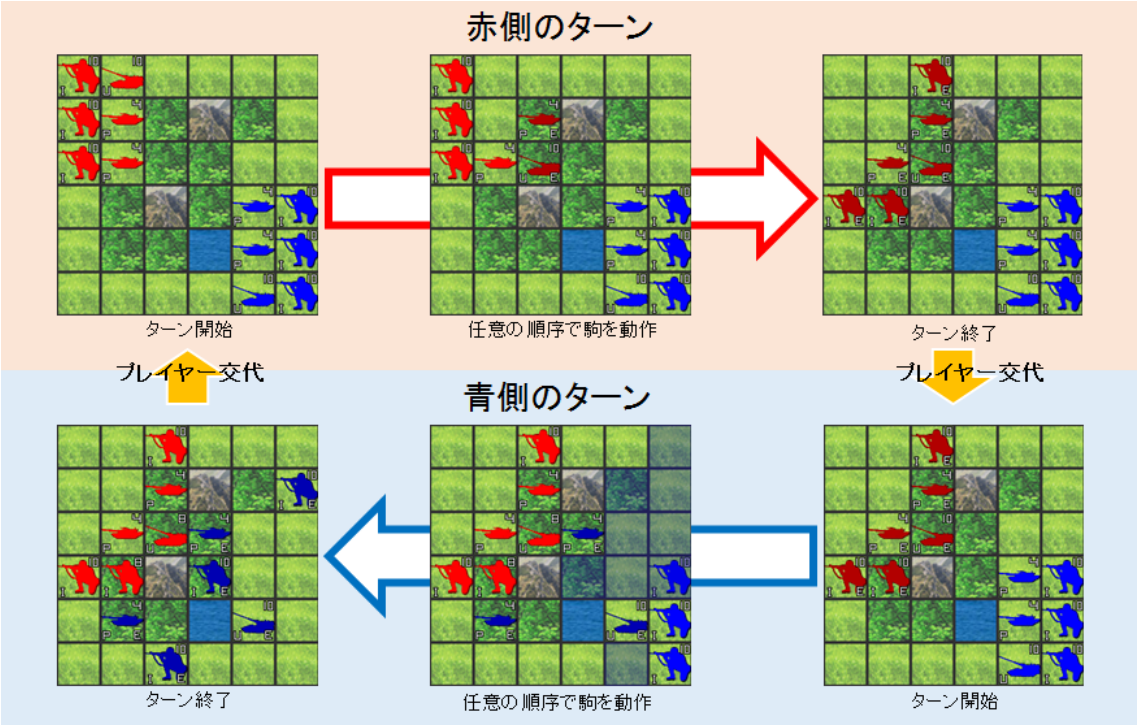


図 2.6: 実際のプレイの流れ

R6. 勝利条件

いずれかのチームのユニットが全滅した場合，ユニットが盤上に存在しているチームの勝利となる．ターン数に上限を設け，その上限以内に全滅条件を満たさない場合の判定条件は別途定める．

R7.HP

各ユニットは 1 から 10 の HP を持つ．攻撃を受けることで HP は減少し，0 になるとその駒は盤上から取り除かれる．

R8. 攻撃

U 以外の各ユニットは，移動前あるいは移動後に，敵ユニットと隣接した状態でのみそのユニットに攻撃ができる．攻撃を受けた側は，HP が 1 以上残っていれば，反撃ができる．U は，移動前のみ，マンハッタン距離で 2 以上 3 以内の敵ユニットに対して一方的に（反撃されることなしに）攻撃ができる．

R9. 地形

山，海，森，草原，道路，要塞，進入不可の 7 種類の地形（図 2.7）を用いる．地形は，防御効果と移動コストに影響を与える．

図 2.7: 地形の種類

R10. 攻撃の効果

攻撃の結果どれだけ相手の HP を減らせるか（ダメージ）は，攻撃側防御側の現在の HP，相性，防御側地形によって定まる．ランダム性は存在しない．

R11. 移動

各ユニットは固有の移動力を持つ．ユニットは上下左右に 1 マスずつ，地形に対する移動コストを消費しながら移動できる．移動力を使い切る必要はない．移動時に自ユニットのいるマスを一時的に通過することはできるが，敵ユニットがいるマスを通過することはできない．

2.3　ターン制ストラテジーの特徴

本節ではターン制ストラテジー全般において発生する，将棋やチェスなどと異なる特徴を説明する．

2.3.1　複数着手性

本稿で使用する TUBSTAP は，大戦略に比べれば多くの要素が削られたかなり単純なものであるが，それでも将棋などと比べれば複雑であり，将棋などに使われる手法（α β 探索など）をそのまま適用することは難しい．特に 1 ターン内における複数着手性により，ターン制ストラテジーではプレイヤがとれる行動の順列（合法手）が非常に多い．例えば 1 駒あたりの平均合法手が 10，駒数が 6 とすると，1 ターン内に取れる行動の数は 7 億 2000 万通りにも及ぶ（同一局面に遷移するものも含む）．実際にはより駒数や 1 駒あたりの合法手数が多い場合も珍しくない．

2.3.2　駒同士の相性

2.2 節 R3 において説明したとおり，ターン制ストラテジーには相性と呼ばれるルールが存在する．この相性は図 2.8 のように駒 R で駒 P に攻撃するよりも，駒 A で駒 P に攻撃する方がダメージが大きくなるといった効果を持つ．相性がもたらすターン制ストラテジーにおける特徴的な状況として図 2.9 のような状況が挙げられる．この盤面は一見青が有利なように見える．しかし，赤の A に対してダメージを与えられる駒は青に存在せず，このような場合，青に勝ち目は無い．このように駒の数を大きく覆す要素に成りえるのが相性である．このような特徴から，駒の価値はマップに応じて変更する必要があり，棋譜の機械学習よりも探索が重視される原因にもなっている．

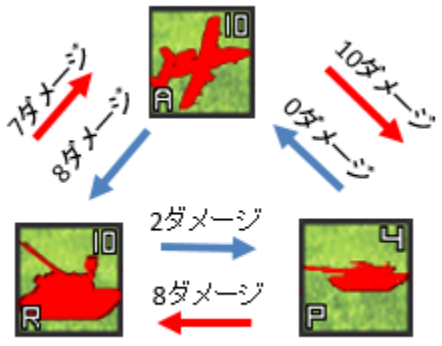


図 2.8: 駒間の相性

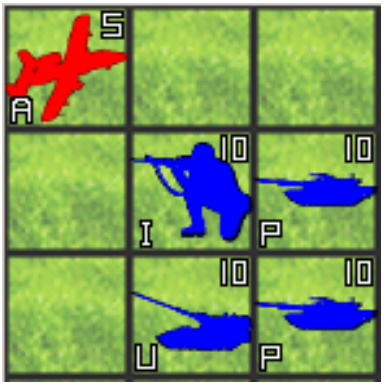


図 2.9: 相性による錯覚

2.3.3　盤面の状態とその性質

TUBSTAP を代表とするターン制ストラテジーには概ね 2 種類の盤面の状態が存在している．1 つは図 2.10 のような防御的な行動が必要とされる局面．この局面において赤側が安易に攻撃を行うと次のターンに大きな反撃を受ける．このような局面の場合，相手の反撃を考慮し陣形を組まなければならない．そのため探索は反撃のある相手のターンまでを含めて深く行う必要がある．また陣形を組む為に行動順序を考える必要は無く，このような局面における最善手は合法手全体から見ると比較的多いものとなる．

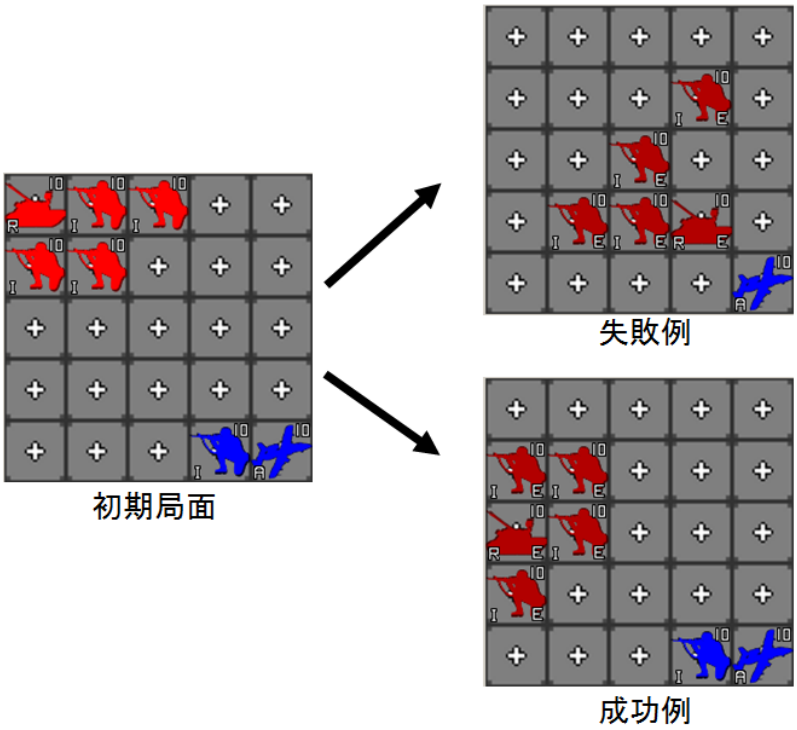


図 2.10: 防御的な行動が必要な曲面（赤先手）

もう 1 つは図 2.11 のような攻撃的な行動が必要とされる局面．このような局面では，自軍（赤）の攻撃で相手の駒を如何に減らす事ができるかが重要であり，その為に駒の攻撃順序や相性などを考慮する必要がある．1 ターン内で全てが行われる為，相手のターンまでを読む必要はないが自軍の駒分は必ず読み切らなければ最善手は打てない．防御的な行動が必要な局面に比べ，攻撃順序を考える必要があるため，このような局面における最善手は合法手全体からすると非常に少なくなる．図 2.11 を例にすると，合法手は約 5400 と非常に多く存在するのに対して，最善手は 6 手しか存在しない．

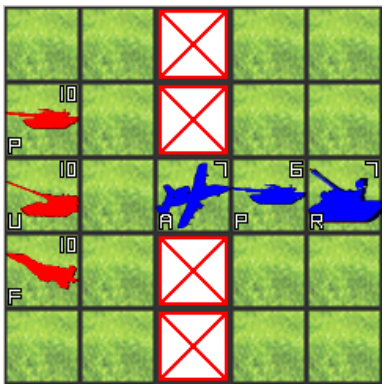


図 2.11: 攻撃的な行動が必要な曲面（赤先手）

合法手と最善手の関係は図 2.12 のようになっている．攻撃的な手であれば，順序を含め自軍のみを重点的に探索せねば見つからない．一方で防御的な手は手順を考える必要は無いが相手の手も考える必要がある．それぞれの局面において最善手を求める為に要求される探索の性質が異なり，そのため探索アルゴリズムによって得手不得手な局面が存在する．しかし，ターン制ストラテジーは多くの場合，防御的な局面において上手く相手の攻撃を受けた後に攻撃的な局面に移りやすい．その為，攻撃・防御どちらか一方が強くても意味はなく，両局面に上手く対応できるかが重要である．これらの考えはあくまでも自身の経験に基づいたものであり，定義化にされたものではない．しかしながらある程度ターン制ストラテジーを遊んだ事のある人間であれば理解でき，重要な考え方であると考える．なお，5 章，6 章における提案手法はこの考え方が根底に存在している．

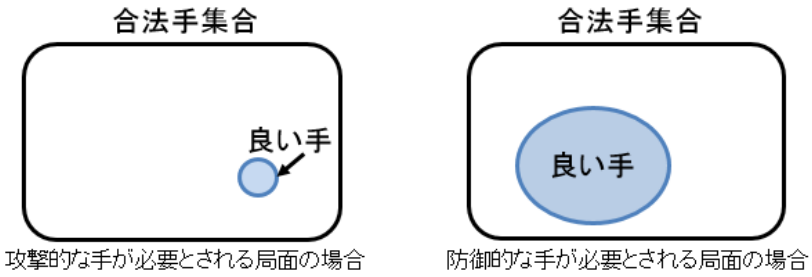


図 2.12: 合法手と最善手の関係

第3章　アルゴリズムの評価方法

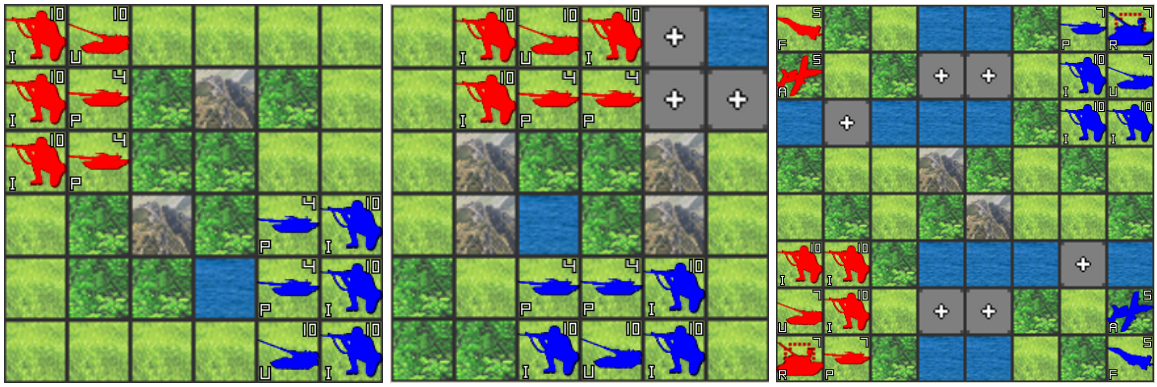
基本的にはAIの性能は1 vs 1の対戦性能で評価すべきではあるが，2.3.3節において説明した通り，攻撃・防御等，状況に応じて異なる能力が求められることが多い．様々な能力が求められることを考えると，各能力のみを評価することにも価値がある．その為，本稿ではそれぞれの性能を定量的に評価するために通常の対戦実験に加えて序盤戦評価実験，詰め状況評価実験の計3種類の実験を用いる．それぞれ，総合評価，防御性能の評価，攻撃性能の評価といった役割である．尚，それぞれの実験における1ターンの実行時間はCore i5 3.3GHz，メモリ8GBのPCを用いて約3秒程度とした．本稿で用いる各アルゴリズムのパラメータは表A.1の通りである．本稿において，各アルゴリズムの動作時間を一定にして性能の比較を行うのは，各アルゴリズムの手法が大きく異なり探索数やシミュレーションなどで公平にすることはできなかった為である．実験内容の詳細は以下に説明する．

3.1 序盤戦限定評価

TUBSTAP などに代表されるターン制ストラテジーにおける序盤戦は適切な受けの形（防御的な陣形）を取れるかがそれ以後の勝敗に大きく関わる．そこで，序盤戦において適切に陣形を組めるかという点を定量的に評価するため以下のような実験を行った．

- 1 ターン目先攻のみを測定対象の AI にプレイさせる．
- 2 ターン目以降は全て対戦相手と同一の AI にプレイさせる．
- 上記のような形式の対戦を 500 回行い勝率を測定する．
- 図 3.1 の計 3 種類のマップを用いる．

使用するマップを上記した 3 種類としたのは，それぞれのマップにおける初手がより重要なマップであると考えている為である．なお，対戦相手とする AI は UCT+PW で固定した．これは UCT+PW が終盤戦に強いと感じたためである．これらの結果によって得られる勝率は，UCT+PW の初手に比べどの程度勝率上昇しているかが重要であり，50%より上になっていれば良いものではない．



(a) マップ：map α　(b) マップ：map β　(c) マップ：map γ

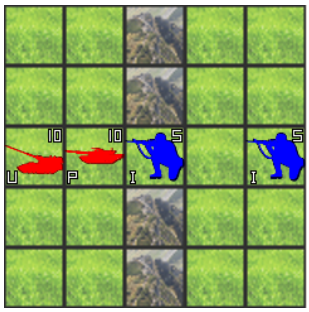
図 3.1: 序盤戦評価マップ（赤先手）

3.2　詰め状況の評価

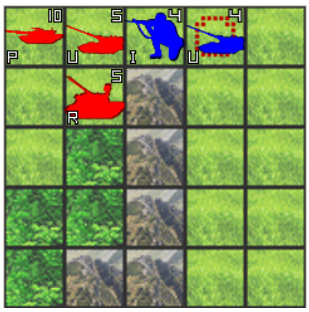
終盤の詰め局面において，十分に読みきれるかどうかは勝率に大きく影響を与える．そこで，1 ターンの間に敵の駒を全て倒せる局面を用意し詰めきれるかを評価する．詳細は以下の通りである．

- 1 ターンの間に全ての敵を倒しきれる 4 種類のマップ（図 3.2）を用いる．
- 1 ターンの間に全ての敵を倒しきれれば勝利，倒しきれなければ敗北とする．
- 上記のような形式の対戦を 100 回行い勝率を測定する．

それぞれのマップは数字が大きくなるほど難しいものとしている．正答は図 3.3 の通りとなっている．map01 は単純に自軍の攻撃順序に気をつければ解けるようになっており，駒 U，駒 P の順序で動かす事が正解となる．map02 は正答する為に駒 U の移動，駒 R の攻撃，駒 P の攻撃といった順序で動かす必要がある．一番最初に移動行動が必要となっており，攻撃順序のみを注意すれば良い問題ではない．しかし，探索するべきなのはたった 3 手分であり，これも簡単な問題となっている．map03 は 4 駒を相性の良い相手（対になっているアルファベットの駒）に向かって攻撃する必要があり，やや難しい問題となっている．map04 は 6 駒全てを相性の良い順序で攻撃する必要があり，探索空間が非常に広い問題となっている．全問題中で最も難しい問題となる．

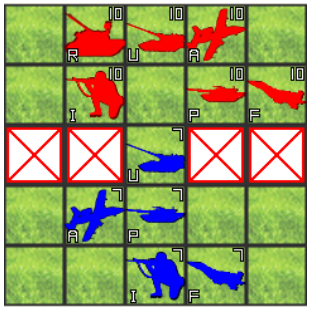


(a) マップ：map01



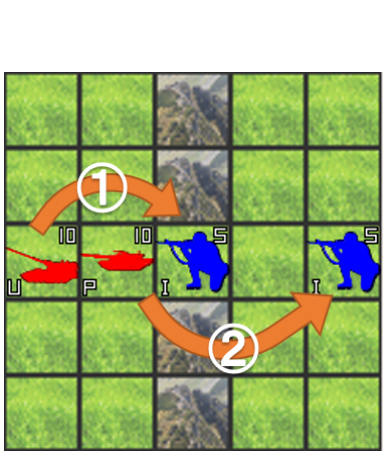
(b) マップ：map02

(c) マップ：map03

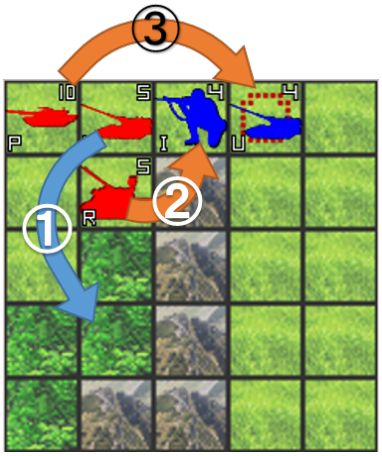


(d) マップ：map04

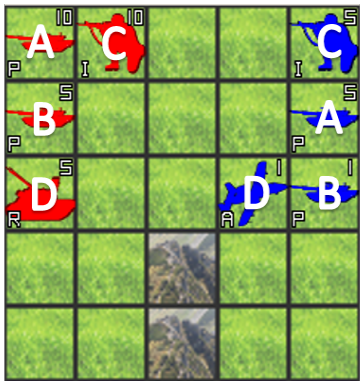
図 3.2: 詰め状況マップ（赤先手）



(a) マップ：map01



(b) マップ：map02



(c) マップ：map03

(d) マップ：map04

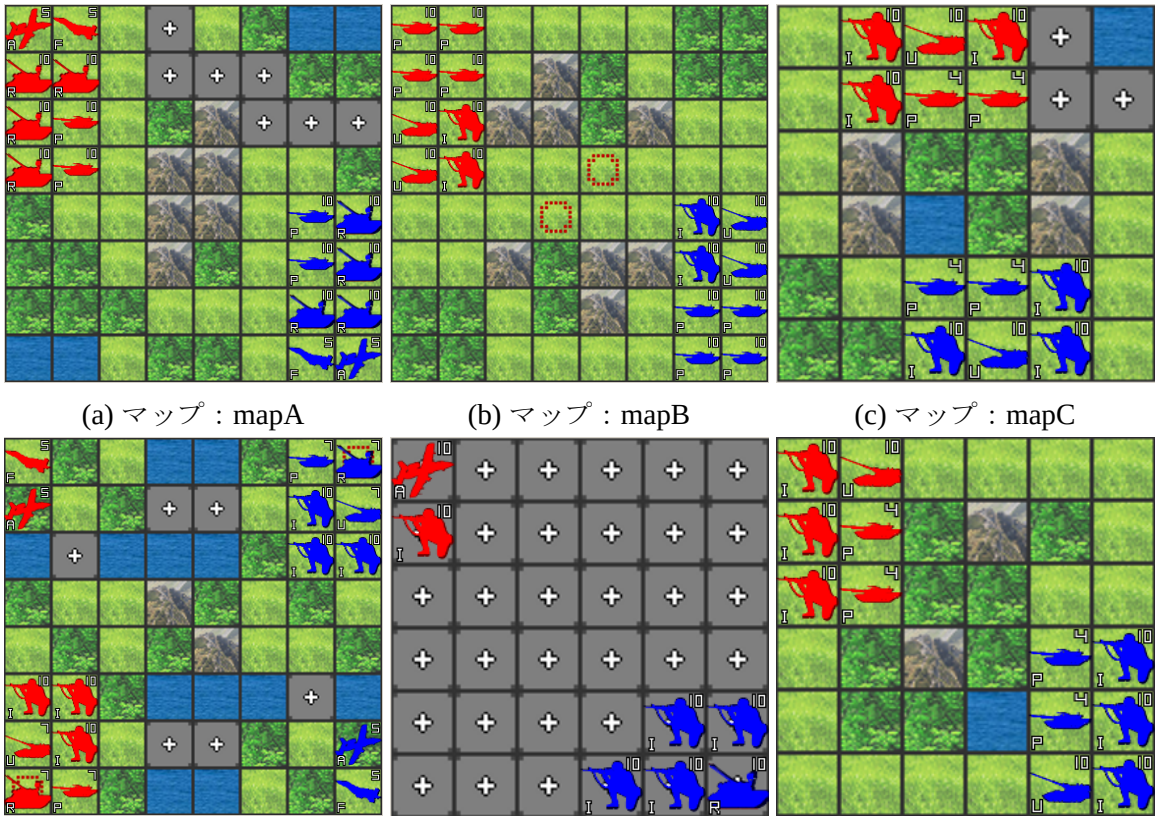
図 3.3: 詰め状況マップ正答手順

3.3 対戦実験

対戦実験は AI 自体の本質的な強さを測る上で欠かせないものである．本稿では以下のような条件で行った．

- マップは GPW2015 対戦会で使用されたマップ 5 種類（図 3.4 の mapA〜mapE）+TUBSTAP のデフォルトマップ 1 種類（図 3.4 の mapF）の計 6 種を用いる．
- 先手後手 300 戦，計 600 戦
- 引き分けは 1/2 勝として勝率を算出する．

使用したマップの形状は図 3.4 の通りである．GPW2015 対戦会 [22] で使用されたマップ（図 3.4 の mapA〜mapE）を用いるのはこれらがそれぞれ異なる参加者が製作したものであるためであり，マップの傾向に偏りが無い為である．また，TUBSTAP に標準で用意されているデフォルトのマップ（図 3.4 の mapF）は過去に多く使われており [9][10]，傾向が分かりやすい為である．



(a) マップ：mapA　(b) マップ：mapB　(c) マップ：mapC

(d) マップ：mapD　(e) マップ：mapE　(f) マップ：mapF

図 3.4: GPW2015 対戦会にて使われたマップ（赤先手）

第4章　既存のコンピュータプレイヤ

この章では上記した複数着手性が持つ探索空間の広さに対して有用だと思われる既存の探索手法を紹介する．以下に本稿で用いる記号を示す.

s : 盤面の状態.

S : 全ての盤面の状態 s の集合.

a : 1 つの駒の行動.

$A(s)$: 状態 s の合法手となる全ての行動 a の集合.

$s'(s,p)$: 状態 s で行動 a を行った次状態．次状態にランダム性は存在せず，一意に決まる．行った場合，行動可能な駒が残っている.

$p = \{a_1, a_2, ...\}$:1 ターン内全ての駒による行動の順列.

$P(s)$: 状態 s の合法手となる行動の順列 p の集合.

$s'(s,p)$: 状態 s で行動の順列 p を行った次状態．行った場合，相手の手番となる.

$g(s)$: 状態評価関数による状態 s の評価値.

4.1　ゲーム木の生成方針

複数着手性を持つゲームには木探索の実装方法が大きく分けて 2 通り存在する．1 つは図 4.1a のように個々の駒の行動を枝として 1 ターン内の行動を数段に分けた探索木を作る方法，もう 1 つは図 4.1b のように 1 ターン内に行動可能な全ての駒の行動をまとめて 1 つの枝として探索木を作成する方法である．これらは根からの枝の数こそ異なるものの，1 ターンを終えた後に辿りつくノードは同じ数となり， 2.3 節で説明した特徴からこれら全てのノードを調べきることは難しい．本稿では以後探索アルゴリズムを紹介するが，それらにおいてもどちらかの手法を用いている．

(a) 1 駒の行動を枝としたゲーム木　　(b) 1 ターンの行動全てを枝としたゲーム木

図 4.1: ゲーム木の生成方針

4.2　UCT

UCT はモンテカルロ木探索の一種であり，探索の指標に UCB1 値（式 4.1）を用いる．囲碁や将棋など様々なゲームでしばしば用いられる手法であり，ターン制ストラテジーであっても有望な手法である可能性が高い．尚，1 ターンの行動全てを 1 つの枝とするゲーム木を用いる場合，UCT としての利点が生かせないため，1 駒の行動を 1 枝として実装する．

$$UCB1 = \overline{x_j} + \frac{\sqrt{2\log n}}{n_j} \tag{4.1}$$

$\overline{x_j}$:あるノード j の勝率
n_j:あるノード j のシミュレーション回数
n:シミュレーション回数の合計

シミュレーションの詳細

TUBSTAP における駒の行動は大きく分けて単純に移動するだけの行動と攻撃を含む行動の 2 通りがある．1 つの駒が持つ行動のうち攻撃する行動は移動するだけの行動に比べると平均的には非常に少ない．そのため，完全にランダムなシミュレーションを行うと指定ターン内に勝敗が決せず，引き分けで終わることが多々発生する．そこで本稿では攻撃可能な駒が存在する状況ならば必ず攻撃を行う方策を用いている．尚，シミュレーション手法には様々な改善が考えられるが，探索手法の比較という観点から本論文の実験ではすべての手法で同じシミュレーション手続きを用いている．

4.2.1 Progressive Widening

ターン制ストラテジーは 1 ターンの候補手が膨大であるという理由から，単純に UCT による探索を行ったとしても，強い AI を作ることは難しい．一方で，過去に UCT の枝刈りを行う事で AI の性能が向上する事が確認されている [23]．そこで枝刈りの代わりとして Progressive Widening[24] を用いて探索空間を狭める．Progressive Widening（PW と略す）[24] とは，UCT においてある局面の訪問回数 n に応じて，有望な着手だけを探索する手法である．本稿では，攻撃行動を優先して探索させる為 UCT に対して PW を導入する．攻撃行動を優先させるのは，ターン制ストラテジーにおいて相手の駒を減らす唯一の行動であり，勝利するために最も重要な行動であると考える為である．具体的には，式 4.2,4.3 に従って探索する候補手を増やしていく．追加する候補手は，攻撃行動を重複無くランダムに選ぶものとする．攻撃行動を全て追加した場合，移動行動をランダムに追加する．なお，式 4.2,4.3 は PW の導入する際に参考にした文献 [24][25] に基づくものである．

$$Candidate = \frac{\log \frac{n}{40}}{1.4} + 2 \qquad (n < 3000) \tag{4.2}$$

$$Candidate = \frac{\log \frac{n+2000}{45}}{1.2} - 11 \qquad (n > 3000) \tag{4.3}$$

n:シミュレーション数

4.3 攻撃行動探索:Attack Action Search（AAS）

村山らは攻撃する行動に注目した手法を提案した [8]．以下に詳細を示す．

1. 状態 s における長さ l 以下の攻撃行動のみで構成される順列 p を全て列挙する．攻撃行動が存在しない場合と長さ l 以降の行動は特定のルーチンに従った行動を順列に追加し 1 ターン分の順列とする．

$$\mathrm{P}'(\mathrm{s}) = \{p_1,\ p_2,\ \ldots,\ p_m\} \subset \mathrm{P}(\mathrm{s})$$

2. $P'(s)$ に含まれる順列 p から遷移する次状態 $s'(s,p)$ を状態評価関数 g により評価し，最も評価の高い順列 p を選択する．

$$p^* = \arg\max_{p \in \mathrm{P}'(\mathrm{s})} g(\mathrm{s}'(\mathrm{s},\ p))$$

1 において長さ l 以下の攻撃行動にのみで構成される順列しか探索を行わないのは計算量爆発を防ぐ為である．たとえ攻撃行動のみに限定したとしても全駒が偶然攻撃行動をとれる場合，計算量の爆発が起こりえる．そこで事前に探索する順列の長さを制限しておくことで，計算量の爆発を防いでいる．この手法は図 4.2 のような駒の行動順が重要となる局面で上手く動作する．ここで大文字小文字はチームを表しており，数字は残り HP，× 印は移動できないマスを表す．駒 U（自走砲）は移動しなければ距離 2-3 までの位置に攻撃することができ，駒 P（戦車）は移動後も攻撃が可能だが隣接する駒にしか攻撃できない．そのため，P を先に動かしてしまうと u を攻撃できずターンが終了してしまう．最善となるのは U で先に p を攻撃し P で u を攻撃する場合である．攻撃行動探索は攻撃行動のみ 2 手目以降を読み切るためこういった局面でも上手く探索が行える．ただし，上手く動作するのは攻撃が行える場合のみである．

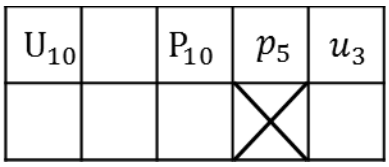


図 4.2: 攻撃行動探索がうまく働く局面の例（先手：大文字）

図 4.3 のような状況は攻撃行動探索で上手く探索できない例である．U は隣接する p に対して攻撃できず，u に対しては攻撃できるが戦果が低く反撃を受けてしまう．また，U の位置取りが邪魔なため P も p に対して攻撃できない．このような状況であれば，先に U を退かした上で，P で p を攻撃するのが最善となる．

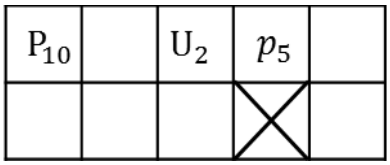


図 4.3: 攻撃行動探索が失敗する局面の例（先手：大文字）

また，図 2.4 のような初期局面でも上手く動作しない，このような初期局面は移動ルーチンしか動作しない為である．

4.4 性能評価

本節では，先ほど紹介したUCT，UCT_PWのそれぞれの性能を上記した方法により評価する．また，攻撃行動探索は攻撃時のみの探索法であり対戦実験などで評価する対象としては使用する状況が限定された手法である．その為，今回は攻撃性能評価である詰め状況の評価でのみ使用する．尚，攻撃行動探索に単純な移動ルーチンを組み合わせUCTと対戦した場合，9割以上の確率でUCTが勝利する．

4.4.1 序盤戦限定評価

UCTとUCT_PWの序盤戦限定評価の結果を表4.1に示す．この実験におけるUCT_PWの結果は単純に自己対戦しているだけであり，他のアルゴリズムと比べる場合のベースラインとなる．UCTであれば，map α，map γにおいてPWに比べ勝率が上昇していることから，それぞれのマップにおいてPWに比べ有効な手が取れており，総合的にもUCTの方が優位な結果を得ている．UCTとPWは似通っているがPWの方がやや攻撃的である．特に防御的な手が求められるような盤面では攻撃的な手を優先して探索しているPWでは有効な手を探索しきれない場合が多い．その為，序盤戦限定であれば何もしていないUCTの方が優位な結果となった．

表4.1: UCT,UCT_PWの序盤戦評価（先手のみそれぞれのAI vs UCT_PW）

AI名	map α勝率	map β勝率	map γ勝率	総合勝率
UCT_PW	43.9	57.7	45.5	49.0
UCT	56.9	44.2	55.8	52.3

4.4.2 詰め状況の評価

UCT,UCT_PW, 攻撃行動探索による詰め状況評価を表 4.2 に示す．map01 は非常に単純な問題であり，全ての手法が 100%正答した．一方で map02 では攻撃行動探索のみが 0%の正答率であった．map02 は正答する為に駒 U の移動，駒 R の攻撃，駒 P の攻撃といった順序で動かす必要がある．その為，最初から攻撃行動のみを探索する攻撃行動探索では正答手順を導けない．一方で map04 のように攻撃行動のみが多く続く状況では攻撃行動探索は上手く動作している．これらの結果より，詰め状況においても探索手法毎に得意な状況が存在しているのがわかる．

表 4.2: UCT,UCT_PW, 攻撃行動探索の詰め状況評価

AI 名	map01 正答率	map02 正答率	map03 正答率	map04 正答率
UCT	100	100	94	0
UCT_PW	100	100	100	13
攻撃行動探索	100	0	100	100

4.4.3 対戦実験

UCT と UCT_PW の対戦結果を表 4.3 に示す．表 4.3 より，mapF を除いたマップにおいて PW を使用した UCT が通常の UCT に勝ち越す結果が得られた．今回使用した攻撃行動を優先して探索するといった簡易的な PW でも十分に性能は向上しており，攻撃行動が優先して探索する価値のある行動であることも示している．一方で，mapF だけにおいて通常の UCT に負け越す結果が得られたが，これは図 3.4e をみて分かるとおり，赤と青両軍に平等な数のユニットを配置しておらず，通常のマップとは毛色が異なる．実際にプレイする場合，先手・後手どちらにしろ先制して攻撃する事がその後の負けに繋がりやすい構成となっており，その為攻撃行動を優先して探索している PW が負け越しているのだと考えられる．

表 4.3: UCT_PW vs UCT（UCT_PW の勝率）

マップ名	mapA	mapB	mapC	mapD	mapE	mapF	総合勝率
先手勝率	68.3	72.6	62.5	76.1	51.8	65.8	65.0
後手勝率	61.3	63.8	53.8	74.0	13.1	59.1	55.8
総合勝率	64.8	68.2	58.1	75.0	34.0	62.5	60.4

第5章　提案手法 深さ限定モンテカルロ法:DLMC

4.4.2 節における結果から，既存の手法は殆ど防御的な手が求められる局面において有効な手が取れていないことがわかっている．そこで本稿では，防御的な手を優先して探索する手法，Depth-Limited Monte-Carlo[10]（DLMC）を提案する．DLMC はゲーム木を図 4.1b のような 1 ターンの行動全てを枝とするような形で展開している．これは，全駒の動作後を評価することで防御的な陣形を上手く取れるようにといった狙いがある．このように DLMC はここまでに紹介した全ての手法とは異なり，防御陣形を組ませる事を主眼として作られている．その為，2.3.3 節における防御的な行動が必要な局面において上手く動作しやすい．尚，以下は「ターン制ストラテジーのための状態評価型深さ限定モンテカルロ法における消極的行動の抑制」[10] を再編したものである．

5.1　手法説明

1. 順列 p をランダムに m 個サンプルする．（図 5.1(a)）

$$\mathrm{P}'(\mathrm{s}) = \{p_1,\ p_2,\ \ldots,\ p_m\} \subset \mathrm{P}(\mathrm{s})$$

2. $P'(s)$ に含まれる p による次局面 $s'(s,p)$ を d ターン先までシミュレーションし（図 5.1(b)），状態評価関数 $g(s)$ により評価を行う．（図 5.1(c)）これを n 回繰り返し，評価値の合計を測定する．　（図 5.1(d)）

$$g_n(p) = \sum_{1}^{n} g(h_i(s'(s,\ p)))$$

   $\mathrm{h}_i(s) : S \to S$i 回目のシミュレーション．d ターン進める．

3. $P'(s)$ の中で評価最大の行動 p^* を選択する．

$$p^* = \arg\max_{p \in \mathrm{P}'(\mathrm{s})} g_n(p)$$

上気している通り，DLMC ではゲーム木の展開を図 4.1b のような 1 ターンの行動全てを枝とするような形で展開している．これにより，全駒の動作が全て終わった状態のみを評価するため，駒同士の陣形を評価しやすい．しかし，全駒を動作と順番を一つの枝とする場合，遷移する局面は膨大な数となる．そこで，探索空間を減らすため，図 5.1 のように状態評価関数を使ったシミュレーションの打ち切りとランダムな前向き枝刈りを用いている．シミュレーションの打ち切りはボードゲーム Amazons において優れた結果を示している手法である [26][27]．一方でランダムな前向き枝刈りは，高速化に大きな影響を与えるが，良い手の見落としのリスクが大きく存在する．ランダムな着手を用いる DLMC は移動するだけの行動を取りやすく，防御的な行動を取りやすい．これは合法手全体で見ると攻撃する行動より移動する行動の方が多いためである．このような特性から図 2.4 のような初期局面において上手く動作するが，図 4.2 のような局面には弱い．また，図 4.3 のような状況であれば，DLMC は一旦引く手を取りやすく，その後の相手の行動次第で打開できる状況に持ち込みやすい．

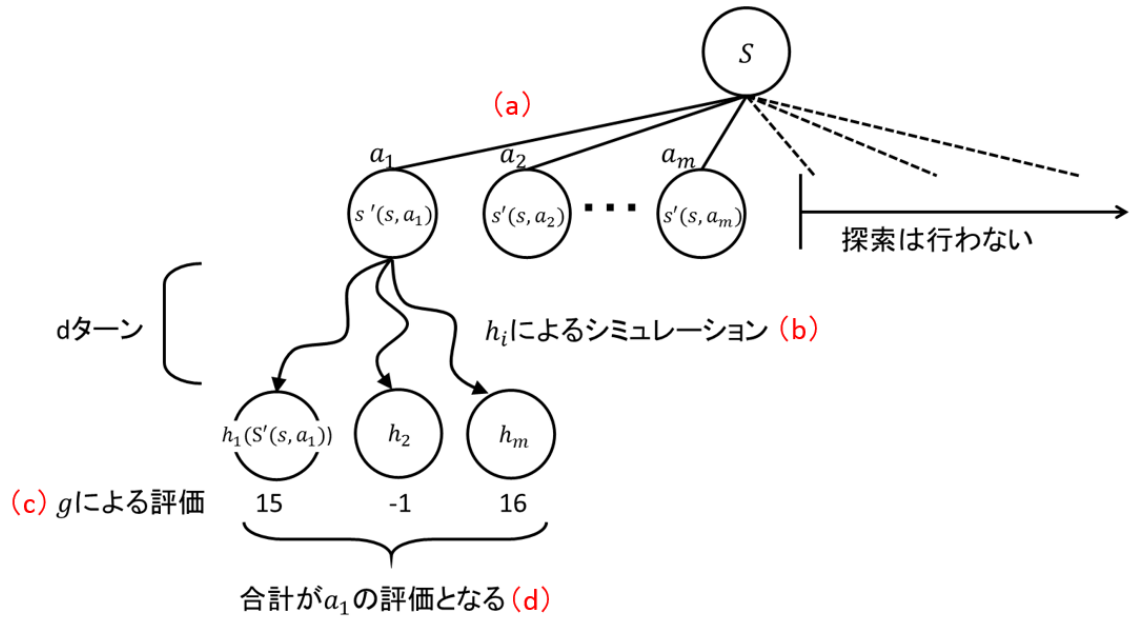


図 5.1: DLMC の概要図

状態評価関数

打ち切った後に使用する状態評価関数 g(s) は重要な要素ではあるが，1 回のシミュレーション毎に使用するため， DLMC ではシンプルな状態評価関数を用いている．

$$
\begin{aligned}
g(s) &= (M-E)*B \\
B &= \begin{cases} 2, & if \quad M=0 \quad or \quad E=0 \\ 1, & else \end{cases}
\end{aligned}
$$

M：自チームの駒の HP の総和（駒 F,A,P,U,R,I はそれぞれ 2，2，1.5，1.0，0.7，0.2 倍）
E：敵チームの駒の HP の総和（駒 F,A,P,U,R,I はそれぞれ 2，2，1.5，1.0，0.7，0.2 倍）
B：ボーナス係数

ここで B は勝敗が決した際に，対戦中の場合と区別し，評価値を倍にするための係数である．勝利であれば自チームの方が敵チームよりも HP の総和が大きいためより良い評価となり，敗北であれば敵チームの方が自チームよりも HP の総和が大きいためより悪い評価となる．また，総和を計算する際には，駒の価値に応じた係数を掛け合わせた上で合計を計算する．この係数は手動で調整したものであり，最適なものではない．

5.2 性能評価

5.2.1 序盤戦限定評価

DLMC の序盤戦限定評価の結果を表 5.1 に示す. 結果より，総合勝率が UCT の 52.3%に比べ 59.5%と大きく上昇していることが確認できる．特に map γ における 68.9%の勝率は非常に大きく，同マップにおける初手の重要性がわかる．以上のような結果から，DLMC は防御的な行動が有効に取れていると考えられる．

表 5.1: 序盤戦評価（先手のみそれぞれの AI vs UCT_PW）

AI 名	map α 勝率	map β 勝率	map γ 勝率	総合勝率
UCT_PW	43.9	57.7	45.5	49.0
UCT	56.9	44.2	55.8	52.3
DLMC	53.9	55.8	**68.9**	**59.5**

5.2.2 詰め状況の評価

DLMC による詰め状況評価を表 5.2 に示す．攻撃行動探索のように攻撃的な手を優先して探索するわけでもなく，UCT のように探索順序を操作しているわけでもない DLMC ではランダムに引いた手の中から正答が出る事を期待する事しかできない．その為，表 5.2 のように全体としての正答率は低い．非常に簡単な map01 でさえも 24%の確率で誤答している．DLMC は防御的局面では非常に高い性能を発揮する一方で詰め局面では極端に弱く，その為通常の対戦では勝率が低いのだと考えられる．

表 5.2: 詰め状況評価

AI 名	map01 正答率	map02 正答率	map03 正答率	map04 正答率
UCT	100	100	94	0
UCT_PW	100	100	100	13
攻撃行動探索	100	0	100	100
DLMC	74	21	49	0

5.2.3 対戦実験

4.4 節の対戦実験において良好な成績を収めた UCT_PW と DLMC での対戦実験を行った．表 5.3 にその結果を示す．結果より，殆どの場合において DLMC は UCT_PW に対して優位な結果を残せていない事がわかる．これは，DLMC が基本的には相手からの攻撃を待つだけの手法となってしまっているためである．5.2.1 節，5.2.3 節の結果から，何らかの組み合わせが有望と考えられる．

表 5.3: DLMC vs UCT_PW（DLMC の勝率）

マップ名	mapA	mapB	mapC	mapD	mapE	mapF	総合勝率
先手勝率	2.6	16.3	35.3	51.6	0	22.3	21.3
後手勝率	6.3	8.3	33.0	48.1	74.0	24.6	32.4
総合勝率	4.5	12.3	34.1	49.9	37.0	23.5	26.9

第6章　既存手法と提案手法の組み合わせ

本章では，防御的な行動において有効な手を取ることができるが，攻撃的な手が上手くとれない DLMC と，攻撃的な手が上手く取る事ができるが防御的な手が上手くとれない既存の手法を組み合わせる事で性能の向上を図る．

6.1　DLMC と攻撃行動探索

攻撃行動探索は攻撃的な手をとることに優れている一方で，防御的な手が取れない．特に図 2.4 のような状況で陣形をとることが出来ない．DLMC であればこのような状況であっても陣形をとることが可能である．しかし，DLMC は図 4.2 のような詰め行動は非常に弱い．これら 2 つはお互いに明確な欠点を持っており，補うことが出来る．そこで，DLMC に攻撃行動探索を組み合わせる事で性能の向上を図る．

1. DLMC を用い最大評価の行動の順列 p^*_{DLMC} を決定する．
2. 攻撃行動探索を用い最大評価の行動の順列 p^*_{AAS} を決定する．
3. p^*_{DLMC} と p^*_{AAS} によるそれぞれの次局面 $s'(s,p)$ を終局までシミュレーションし，勝敗により評価を行う．これを n 回繰り返し，勝利数の合計を測定する．この際，勝利は 1，引き分けは 0.5，敗北は 0 として扱う．
4. 評価の高かった行動の順列 p^* を選択する．

この手法は特定の局面における良い手の見逃しを防ぐ事を目的としている．動作は非常に単純で，DLMC による最善手と攻撃行動探索による最善手をそれぞれシミュレーションにより終局まで評価し，より評価値の高い方を採用する．攻撃行動探索と DLMC 両者の特性を利用できるため，図 2.4 や図 4.2 どちらの状況にも対応することが出来る．

6.2 DLMC と UCT

DLMC と攻撃行動探索の組み合わせと同様に DLMC と UCT の組み合わせも試すのは自然な発想である．特に UCT_PW は詰め局面において攻撃行動探索よりも正答率が高く，その点でより期待できる手法と言える．以下に動作手順を記述する．

1. DLMC を用い最大評価の行動の順列 p^*_{DLMC} を決定する．
2. UCTPW を用い最大評価の行動の順列 p^*_{UCT} を決定する．
3. p^*_{DLMC} と p^*_{UCT} による次局面 $s'(s,p)$ を終局までシミュレーションし，勝敗により評価を行う．これを n 回繰り返し，勝利数の合計を測定する．この際，勝利は 1，引き分けは 0.5，敗北は 0 として扱う．
4. 評価の高かった行動の順列 p^* を選択する．

攻撃行動探索との組み合わせと同じように，DLMC と UCT それぞれの最善手をシミュレーションを用いて評価し，最終的により良い評価を得た物を採用する．

6.3 性能評価

本節ではDLMC+攻撃行動探索，DLMC+UCT_PWの性能評価を行なう．尚，パラメータは表A.1の通りとしている．DLMC+攻撃行動探索において攻撃行動探索は深さ4の探索すると実行時間が3秒となってしまい，DLMCの実行時間が残らない為，深さ3で実行している．動作時間が0.1秒ほどと非常に早いため，実行時間の殆どはDLMCが動作していることになる．また，DLMC+UCTはどちらも実行時間を細かく設定できるので，1.5秒づつ動作させている．

6.3.1 序盤戦限定評価

それぞれの提案手法による序盤戦限定評価の結果を表6.1に示す．結果より，DLMC+UCTはUCT_PWのみの場合（表4.1）に比べ，総合勝率が49%から55.2%と上昇していることが確認できる．DLMCの実行時間は表5.1の半分となっているが，DLMCが実行時間半分でも十分に動作することが確認できる．また，DLMC+攻撃行動探索はUCTとの組み合わせに比べ総合勝率が58.4%と高いが，これはDLMCに割り振った時間がUCTとの組み合わせに比べて長いためである．

表 6.1: 序盤戦評価（先手のみそれぞれのAI vs UCT_PW）

AI名	map α勝率	map β勝率	map γ勝率	総合勝率
UCT_PW	43.9	57.7	45.5	49.0
UCT	56.9	44.2	55.8	52.3
DLMC	53.9	55.8	68.9	59.5
DLMC+攻撃行動探索	51.4	57.2	66.7	58.4
DLMC+UCT_PW	50.5	55.1	60.1	55.2

6.3.2 詰め状況の評価

それぞれの提案手法による詰め状況評価を表 6.2 に示す．結果より，それぞれ DLMC 単独より性能が向上したものの，組み合わせた既存手法より下がる結果となった．特に攻撃行動探索の結果は大きく下がっており，map04 を探索しきれていないことが確認できる．有望な手に優先してシミュレーションを割り振っていく UCT に比べ，攻撃行動探索は決まった深さまでを調べるだけといった手法な為，割り振られる時間に応じて大きく性能が下がったのだと考えられる．

表 6.2: 詰め状況評価

AI 名	map01 正答率	map02 正答率	map03 正答率	map04 正答率
UCT_PW	100	100	100	13
UCT	100	100	94	0
攻撃行動探索	100	0	100	100
DLMC	74	21	49	0
DLMC+攻撃行動探索	100	27	100	3
DLMC+UCT_PW	98	96	92	13

6.3.3 対戦実験

4.4 節の対戦実験において良好な成績を収めた UCT_PW とそれぞれの提案手法で対戦実験を行った．DLMC と攻撃行動探索による結果を表 6.3 に，DLMC と UCT_PW による結果を表 6.4 に示す．結果より，DLMC+UCT_PW の組み合わせは全てのマップにおいて UCT_PW に勝ち越しており，総合勝率も 59.5%と高い性能を発揮している．6.3.1 節,6.3.2 節において示したとおり，UCT_PW との組み合わせは序盤戦評価，詰め状況評価どちらも高い性能を示していた．攻撃，防御どちらもの性能がある程度の性能でまとまっているため，高い勝率となったと考える．一方で，攻撃行動探索との組み合わせは総合勝率で 51.6%とあまり性能が向上していない．特に低い勝率を記録している mapA や mapB などにおける対戦ログを精査してみると，図 2.9 のように相性の悪い駒を相手側に残し負けている事が確認できた．攻撃行動探索，DLMC 共に状態評価関数を用いて盤面を評価している．また，この状態評価関数は駒同士の相性は考慮せず，残る駒数が高ければ評価も高くなるといった仕様になっている．その為，一時的には駒数が多く保有でき有利な状態に立つものの，その後は相性の悪い駒に一方的な攻撃を受けるといった状況を作りやすい．このような特性が原因となり負け越しているのではないかと考える．

表 6.3: DLMC+攻撃行動探索 vs UCT_PW（DLMC+攻撃行動探索の勝率）

マップ名	mapA	mapB	mapC	mapD	mapE	mapF	総合勝率
先手勝率	38	37.1	49.6	74.3	75.0	52.1	54.3
後手勝率	53.8	33.8	59.6	56.7	49.7	39.8	48.9
総合勝率	45.9	35.5	54.6	65.5	62.3	46.0	51.6

表 6.4: DLMC+UCT_PW vs UCT_PW（DLMC+UCT_PW の勝率）

マップ名	mapA	mapB	mapC	mapD	mapE	mapF	総合勝率
先手勝率	47.5	65.0	61.8	62.0	75.8	62.5	62.4
後手勝率	55.8	59.7	70.3	46.8	52.7	54.5	56.6
総合勝率	51.7	62.3	66.0	54.4	64.2	58.5	**59.5**

第7章　まとめ

本稿では，ターン制ストラテジーにおける特性，コンピュータプレイヤ作成における注目すべき点，UCT や攻撃行動探索などの既存手法を紹介した．ターン制ストラテジーでは局面に応じて，防御的な手，攻撃的な手それぞれが必要とされる場合があり，それぞれの手を見つける為に必要な探索範囲が異なることを踏まえた上で，新たに防御的な局面を重視した探索手法 DLMC を提案した．序盤戦評価実験おいて UCT は勝率 52.3%ほどであったが，DLMC は 59.5%と性能が向上しており，DLMC が防御的な局面において有効に働くことを示した．また，UCT と DLMC を組み合わせる事で防御的な手を取れないといった問題点を改善し，UCT_PW との対戦実験では勝率 59.5%と勝率が向上したことを報告した．

付 録A　パラメータ表

AI 名	探索する深さ	シミュレーション数	サンプル数	備考
攻撃行動探索	4	-	-	-
UCT	-	6000	-	-
UCT_PW	-	6000	-	-
DLMC	3	100	200	-
攻撃行動探索	3	-	-	DLMC+攻撃行動探索に使用
DLMC	3	90	200	DLMC+攻撃行動探索に使用
UCT	-	3000	-	DLMC+UCT に使用
DLMC	3	50	200	DLMC+UCT に使用

表 A.1: パラメータ表

謝辞

まず初めに，本研究を始めるにあたりご指導いただきました主指導教員の池田心准教授，副指導教員の飯田弘之教授に深く感謝致します．TUBSTAP を共に開発してきた，池田研究室の村山公志朗さんにも感謝しております．また，TUBSTAP を利用して下さった，池田研究室の佐藤直之さん, 飯田研究室の石飛太一さんを初め，他大学の方々から多くの意見を頂き，大変感謝しております．他にも，池田研究室・飯田研究室の皆様にも様々なご協力を頂き，感謝いたします．

参考文献

[1] DeepBlue,*http://sjeng.org/ftp/deepblue.pdf*. （アクセス日時：2016 年 2 月 1 日）

[2] 小谷善行, **第 *3* 回将棋電王戦を振り返って : *3*. コンピュータ将棋の棋力の客観的分析 ～人間のトップに到達したか？～**. 情報処理, Vol.55, No.8, pp.851-852, 2014-08

[3] AlphaGo,*https://storage.googleapis.com/deepmind-data/assets/papers/deepmind-mastering-go.pdf*. （アクセス日時：2016 年 2 月 1 日）

[4] 大戦略 PERFECT3.0,*https://www.ss-alpha.co.jp/products/dsperfect3.html*. （アクセス日時：2016 年 2 月 1 日）

[5] ファミコンウォーズ DS,*https://www.nintendo.co.jp/ds/awrj/*. （アクセス日時 : 2016 年 2 月 1 日）

[6] CIVILIZATION V,*http://www.civilization5.com/*. （アクセス日時：2016 年 2 月 1 日）

[7] TUBSTAP,*http://www.jaist.ac.jp/is/labs/ikeda-lab/tbs*. （アクセス日時：2016 年 2 月 1 日）

[8] 村山, 藤木, 池田, **学術研究用プラットフォームとしての大戦略系ゲームのルール提案**. IPSJ-GPW 2013-11-01, pp.146-153, 2013-11

[9] Tsubasa Fujiki, Kokolo Ikeda and Simon Viennot, *A Platform for Turn-Based Strategy Games, with a Comparison of Monte-Carlo Algorithms*. IEEE Conference on Computational Intelligence and Games (CIG2015), pp.407-414, 2015-08

[10] 藤木, 村山, 池田, **ターン制ストラテジーのための状態評価型深さ限定モンテカルロ法における消極的行動の抑制** . IPSJ-GPWS 2014

[11] Thomas R. Hinrichs, Kenneth D. Forbus, *Analogical Learning in a Turn-Based Strategy Game*. 21th International Joint Conference on AI, pp. 853-858

[12] Wender S, Watson I, *Using reinforcement learning for city site selection in the turn-based strategy game civilization IV.*. 21th International Joint Conference on AI, pp. 853-858

[13] Arimaa Homepage, *http://arimaa.com/arimaa/*, （アクセス日時：2016 年 2 月 1 日）

[14] David Fotland, *Building a World-Champion Arimaa Program*, Volume 3846 of the series Lecture Notes in Computer Science, pp. 175-186

[15] FinalFantasyTactics,*http://dlgames.square-enix.com/fft/*. （アクセス日時 : 2016 年 2 月 1 日）

[16] スーパーロボット大戦, *http://www.suparobo.jp/*, （アクセス日時 : 2016 年 2 月 1 日）

[17] ファイアーエムブレム, *https://www.nintendo.co.jp/fe/*, （アクセス日時 : 2016 年 2 月 1 日）

[18] StarCraft, *http://us.blizzard.com/en-us/games/sc/*, （アクセス日時 : 2016 年 2 月 1 日）

[19] Age of Empire, *http://www.ageofempires.com/*, （アクセス日時 : 2016 年 2 月 1 日）

[20] BWAPI, *https://code.google.com/p/bwapi/*, （アクセス日時 : 2016 年 2 月 1 日）

[21] IEEE-CIG Competitions, *http://geneura.ugr.es/cig2012/competitions.html*, （アクセス日時 : 2016 年 2 月 1 日）

[22] TUBSTAP GPW2015 対戦会, *http://www.jaist.ac.jp/is/labs/ikeda-lab/tbs/gpw_2015.htm*, （アクセス日時 : 2016 年 2 月 1 日）

[23] Tihiro Kato,Makoto Miwa,Yoshimasa Tsuruoka,Chikayama Takashi, *UCT and Its Enhancement for Tactical Decisions in Turn-based Strategy Games*, IPSJ-GPW 2013-11-01, pp.138-145, 2013

[24] Remi Coulom, *Computing Elo Ratings of Move Patterns in the Game of Go*. ICGA Journal Vol.30, pp.198-208, 2007

[25] 松原, 美添, 山下, **コンピュータ囲碁モンテカルロ法の理論と実践**.

[26] Julien Kloetzer, Hiroyuki Iida, Bruno Bouzy, *The Monte-Carlo Approach in Amazons*, Computer Games Workshop, 2007

[27] Julien Kloetzer, *Monte-Carlo Techniques:Applications to the Game of the Amazons*. Japan Advanced Institute of Science and Technology, Doctor 240 Includes bibliographical references pp. 87-92, 2010