Nタプルネットワークの大きさと学習性能の関係: ミニ2048を用いた実験・評価

寺内 俊輔[1,a)]　松崎 公紀[2,b)]

概要：2048では，Nタプルネットワークと TD 学習を拡張した手法及び Expectimax 探索の組み合わせにより優れたプレイヤが作られている．とくに，これまで N タプルネットワークのパラメータ数を増やすことで性能向上が図られてきた．しかしながら，その傾向がどこまで続くかについては，議論がなされていないのが現状である．本研究では，ミニ 2048 において利用可能な 1 タプルから 9 タプルまでの各タプル全てからなる組み合わせと，それぞれにおける妥当な組合せを用いて，N タプルネットワークの大きさと学習性能の関係を実験的に評価する．具体的には，N タプルネットワークのパラメータ数とスコアの関係，Optimistic Initialization 手法の初期値と学習への影響，Expectimax 探索を組み合わせたときの性能向上について詳しく評価した．

キーワード：2048, N タプルネットワーク, Expectimax 探索, ミニ 2048, 強化学習

1. はじめに

「2048」は G. Cirulli によって作られた確率的一人ゲームであり [1]．これまでにさまざまなコンピュータプレイヤが作られてきた．現在最も成功しているアプローチは，強化学習によってチューニングした N タプルネットワーク評価関数 [9] と Expectimax 探索 [12] を組み合わせるものである．その後，N タプルネットワークと Expectimax 探索の組合せを基礎として，一般的もしくはゲームに特化した改良手法が多数提案されてきた [2], [3], [6], [12]．Guei によって作られた最先端プレイヤ [2] では，大きさ 6 のタプル 8 個からなるネットワーク 2 つを切り替えて用いる評価関数，より幅広く学習するための Optimistic Initialization，ゲーム特化型の改良手法である Tile downgrading，および Expectimax 探索を組み合わせることで，平均得点 625 377 を達成した．

N タプルネットワークやニューラルネットワークでは，一般に，ネットワーク内のパラメータ数が増えるほど性能が向上すると言われている [4]．一方で，パラメータ数が増えすぎると，学習に必要となるデータが莫大になるという問題に加えて，過学習（過適合）の問題も発生する．

2048 のコンピュータプレイヤにおいては，これまでネットワークのパラメータ数を増やすことで性能向上が図られてきた．2048 に N タプルネットワークを用いる最初の研究である Szubert と Jaśkowski による研究 [9] では，まず大きさ 4 のタプル 17 個からなるネットワーク（パラメータ数 1.11×10^6）が用いられ，次に 6 タプル 2 個と 4 タプル 2 個からなるネットワーク（パラメータ数 3.37×10^7）が用いられていた．続く Wu らによる研究 [11] では，ネットワークが大きさ 6 のタプル 4 つからなるもの（パラメータ数 6.71×10^7）へ拡張された．2018 年時点での最先端プレイヤ [3] では，6 タプル 5 個に redundant encoding を組み合わせたネットワーク（パラメータ数 8.42×10^7）16 個をゲーム進行に合わせて切り替える手法がとられている．論文投稿時点の Guei による最先端プレイヤ [2] では，Matsuzak [5] が実験的に求めた 6 タプル 8 つの組合せ（パラメータ数 1.34×10^8）2 つを切り替えて用いている．この Guei によるプレイヤの学習では，Optimistic Initialization を用いてより幅広く学習する工夫が取り入れられている．一方，ニューラルネットワークによる評価関数を用いるプレイヤでは，Matsuzaki [7] が，畳み込みネットワークのパラメータ数を 1.85×10^5 から 2.90×10^6 まで変化させたときに性能が向上することを報告している．

ここで生じる疑問は，2048 の N タプル評価関数において，性能向上が見られる範囲でどこまでパラメータ数を増やしていけるのか，という点である．Oka と Matsuzaki [8]，および，Matsuzaki [5] は，大きさ 6 のタプルだけでなく，

[1] 高知工科大学大学院工学研究科
[2] 高知工科大学情報学群
[a)] 295141a@gs.kochi-tech.ac.jp
[b)] matsuzaki.kiminori@kochi-tech.ac.jp

大きさ 7 のタプル複数個を系統的に組み合わせる手法を示し，同一条件での比較においては大きさ 6 のネットワークよりも大きさ 7 のネットワークのほうが性能が高くなりうることを示した．しかしながら，大きさ 8 のタプルからなるネットワークでは，単純に実装するとパラメータ数が 4.29×10^{12} と巨大になり，メモリサイズおよび学習コストの観点から，現時点では実現は困難である．

そこで本研究では，大きさが小さくパラメータ数を減らすことができ，また，すでに完全解析による真の正解が分かってるミニ 2048 を用いて，N タプルネットワークのタプルサイズと Optimistic Initialization（OI）の初期値がプレイヤの性能に与える影響について実験的に評価する．本研究の実施にあたって設定したリサーチクエスチョンは次の 3 つである．

- **RQ1** N タプルネットワークにおけるタプルの大きさ，数，パラメータ数に対して，スコアにどのような影響があるか．
- **RQ2** 幅広く強化学習を行う Optimistic Initialization の初期値を変えたときに，N タプルネットワークの学習にどのような影響があるか．
- **RQ3** N タプルネットワーク評価関数と Expectimax 探索とを組み合わせたとき，探索による性能向上は N タプルネットワークとその学習方法に依存するか．

本研究では，まず，タプルの大きさを 1 から 9 とし，そのそれぞれについて 2 種類のタプルの組合せによる N タプルネットワークを設計した．設計した N タプルネットワークに対し，Optimistic Initialization の初期値を変えた学習を行い，Expectimax 探索の深さを変えて各プレイヤの平均得点を調べた．これらの網羅的な実験の結果，以下に示す知見が得られた．

- RQ1 に関して，パラメータ数の対数とスコアの間にはおよそ二次関数（放物線）の関係が見られる．とくに，タプルの大きさが 5 から 6 のときに性能のピークがある．
- RQ2 に関して，Optimistic Initialization の初期値が小さすぎる（$OI = 0$）場合には，低いスコアで学習が止まってしまう場合がある．逆に，初期値が大きすぎる（$OI = 5400$）場合には，パラメータ数が多くなると学習が停滞することが見られた．初期値が適切である（$OI = 1200$）場合，学習が安定して進むだけでなく，より多くのパラメータ数からなるネットワークでスコアが最大化する効果が見られた．
- RQ3 に関して，Expectimax 探索によるスコアの向上は，Optimistic Initialization の初期値やパラメータ数には大きく影響されない結果となった．具体的には，探索深さ 1（Greedy）から 6 に増やすと，理想的な q 最高平均得点 5469 点までの距離がおよそ半分に減っていた．

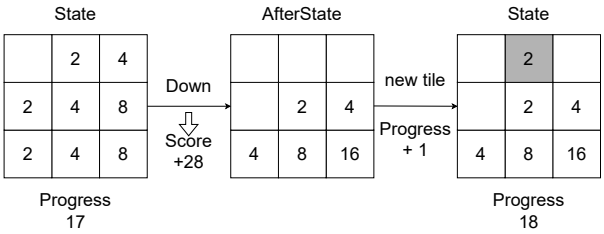


図 1: state，afterstate，progress の例 [15]

2. ミニ 2048

本研究はミニ 2048 [13] を研究対象として使用する．

2.1 ルール

ミニ 2048 は，3×3 の盤面でプレイされる，確率的一人ゲーム 2048 の盤面縮小版である．初期局面は盤面上に 2 つのタイル 2（確率 0.9）か 4（確率 0.1）の数字タイルがランダムに置かれた盤面からなる．

各局面において，プレイヤは上下左右いずれかの方向を選択する．すると全ての数字タイルはその方向にできるだけ移動する．移動した結果，2 つの同じ数字のタイルが移動方向に衝突するとこれらは合体してその合計値のタイルとなり，その合計値がスコアに加算される．合体してできたタイルは，同じターンでは別のタイルと合体することはない．例えば，盤面の行が `2__`，`_22`，`224` であるとき，右を選択するとそれぞれ `__2`，`__4`，`_44` へと変化する．その後，空白のマスのうちのランダムな 1 マスに 2（確率 0.9）か 4（確率 0.1）のタイルが置かれる．

プレイヤはいずれかのタイルが移動または衝突するような方向しか選択することができない．いずれの方向も選択できなくなるとゲームは終了する．このゲームではできるだけ高い得点を獲得することが目標となる．

2.2 用語の導入

通常の 2048 と同様にミニ 2048 における 1 ターンは，「移動・合体ステップ」と「新規タイルステップ」の 2 ステップからなる．これらステップの前後の状態を区別するため，以下の用語を導入する．

state プレイヤが手を選択する盤面状態（とスコア）を *state* と呼ぶ．

afterstate プレイヤが手を選択してタイルが移動・合体した直後の盤面状態（とスコア）を *afterstate* と呼ぶ．すなわち，afterstate は新規タイルが出現する前の盤面状態である．

（通常の 2048 同様に）ミニ 2048 では，新しく出現するタイルはランダムに 2 か 4 の値をとる．そのため，単純にターン数をゲームの流さや進行度の指標に用いるには不都合がある．この問題を解決するため，本研究では以下の指

表 1: Progress, score, and alive ratio of perfect player

Condition	Progress	Score	Alive
256-tile	136	1,750	99.53%
512-tile	263	4,000	73.84%
512-tile & 256-tile	391	5,750	54.40%
512-tile & 256-tile & 128-tile	456	6,500	40.49%
1024-tile	511	9,000	1.07%

標を用いる．

progress　タイルの値の合計値の半分を *progress* [10] と呼ぶ．progress は，1 ターンで 1（新規タイルが 2 の場合）または 2（新規タイルが 4 の場合）だけ増加する．

図 1 は，初期局面から始まるゲームの流れにおいて，state，afterstate，progress について図示したものである．

2.3　完全解析とその結果

ミニ 2048 は確率的一人ゲームであり，その完全解析とは各状態に対して期待スコア求めることである．ミニ 2048 は，到達可能な状態数が 10^9 以下と小さいため，現実的な時間で完全解析ができる．山下ら [13] は，ミニ 2048 の完全解析に最初に取り組み，そこでは幅優先探索による状態列挙と，列挙した状態を用いる後退解析を行った．また，著者ら [10] も完全解析の追試を行い，深さ優先探索による後退解析で，結果の正しさを確認した．

完全解析の結果について，重要なものを以下に示す．初期状態のいずれかから到達可能な state の数は 48,713,519，afterstate の数は 31,431,374 である．初期状態の期待スコアは，5,468.49 である．各 afterstate に対する期待スコアを格納したものを valueDB と呼ぶ．

完全解析で得られる valueDB を用いると，各局面において最適な手を選択するパーフェクトプレイヤを実現できる．ただし，ミニ 2048 は確率的一人ゲームのため，決定的に最善手を選択するパーフェクトプレイヤであっても，ゲームごとにプレイの結果は異なることに注意が必要である．図 2 は，パーフェクトプレイヤが 1 万ゲームを行った際の，progress ごとの生存率とゲーム終了時のスコアを示している．表 1 は，パーフェクトプレイヤが 256，512，1024 タイルに到達したときの進捗状況，スコア，生存率を示している．パーフェクトプレイヤでも，512 タイルに到達した後，1024 タイルに到達する前は，生存率が急激に低下することが分かる．図 2 より，生存率が急激に下がるタイミングがいくつかある．本研究では，そのような生存率が下がる部分を**難易度の高い領域**と呼ぶ．

3.　本研究で用いるプレイヤ

3.1　N タプルネットワーク

本研究では，表 2 に示すとおり，1 タプルから 9 タプルまでの N タプルネットワークを合計で 15 種類設計して用

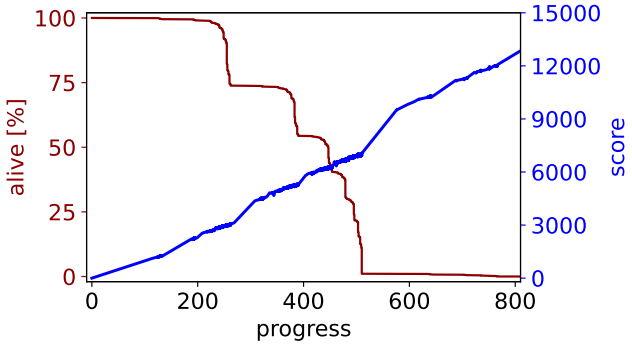


図 2: パーフェクトプレイヤの生存率とスコア [10]

いた．

まず，特徴抽出する N マス（図では丸で示される）がすべて上下左右に連結しているようなものを，有効な N タプルとした．その結果，$N = 1$ から $N = 9$ まで順に有効な N タプルが，3 通り，2 通り，5 通り，6 通り，9 通り，8 通り，7 通り，3 通り，1 通り得られた．指定した大きさ N について，有効な N タプルをすべて含む N タプルネットワークを N-Full と名付けた（本文および図表においては，より短く 1F などと表記する）．

2048 における N タプルネットワークの設計では，有望そうな形をいくつか人手で作成し，それを平行移動させて得られるタプルを組合せる手法がよく用いられている [3], [9], [12]．そこで，この考え方に基づくタプルの組合せを N-Manual として設計した（本文および図表においては，より短く 3M などと表記する）．ただし，1 タプル，2 タプル，9 タプルでは，有効な N タプルがすべて有望そうな形をしていることから，1M，2M，9M はそれぞれ 1F，2F，9F と同一である．

なお，N タプルネットワークで評価値を計算する際には，ミニ 2048 の盤面の持つ対称性（回転・反転）を活用し，各タプルに対して 8 通りの位置からのサンプリングを行う．また，後述する Multistaging により各プレイヤは N タプルネットワークを 2 つ持つことから，表 2 に示すパラメータ数は，タプルサイズを N として

$$\text{パラメータ数} = 11^N \times 2$$

により計算される値である．

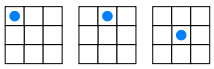
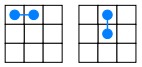
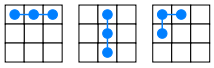
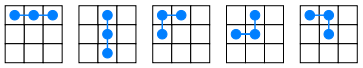
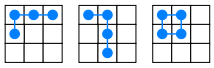
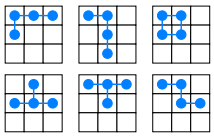
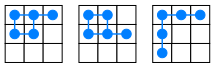
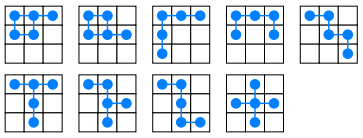
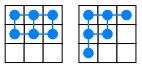
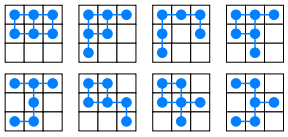
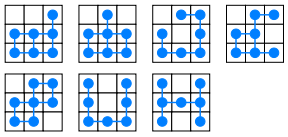
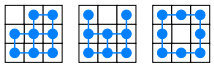
3.2　N タプルネットワークの学習方法

N タプルネットワークの重みは，afterstate 間の評価値の差に基づく TD 学習法の改良手法によって調整した．本研究で用いる N タプルネットワークの学習では，以下の技術を用いた．

Multistaging　ゲームの進行に応じて重みを参照するテーブルを切り替える．本研究では，2 ステージとし，512 のタイルができる前後でステージを分けた．

Temporal coherence 学習（TC 学習）　TC 学習は学

表 2: タプルサイズと組合せの一覧

名称	タプルの組合せ	パラメータ数
1F		66
2F		484
3M		7,986
3F		13,310
4M		87,846
4F		175,692
5M		966,306
5F		2,898,918
6M		7,086,244
6F		28,344,976
7M		38,974,342
7F		272,820,394
8M		428,717,762
8F		1,286,153,286
9F		4,715,895,382

習率自動調整機能を備えた TD 学習で，Jaśkowski [3] が始めて 2048 に導入した．TC 学習では，更新しようとする重みごとに，それまでの学習ステップの更新量の総和を，更新量の絶対値の総和で割った値を学習率とする．本研究の実装では，次の Optimistic initialization の効果がある程度残るように，学習率の最大を 0.5 でクリッピングする変更を加えた．

Optimistic initialization　学習段階での探索を広く行うために，重みを（ゼロではなく）大きな値で初期化する．本研究で用いた N タプルの学習では，すべての afterstate の初期値を $OI = 0$, $OI = 1200$, $OI = 5400$ の 3 通りとした．

それぞれの N タプルニューラルネットワークに対して，5×10^8 局面分のデータで学習を行った．著者らの先行研究 [10] において，この学習量は学習が収束するのに十分であった．

3.3　N タプルネットワークを評価関数とするプレイヤ

本研究で用いるプレイヤは，前節の方法で重みを調整した N タプルネットワークを評価関数とし，Expectimax 探索により手を選択する．Expectimax 探索の実装については，著者らの先行研究 [14] で用いたものをそのまま利用した．

以上より本研究で用いる各プレイヤは，N タプルネットワークの組合せ（1F, 2F, . . . , 9F, 3M, . . . , 8M），学習における Optimistic initialization の初期値（$OI \in \{0, 1200, 5400\}$），探索の深さ（Greedy，$d \in \{2, 3, \ldots, 6\}$）の 3 つにより決定される．（先読みが 1 手の場合は，慣習に従い Greedy と表記する．）

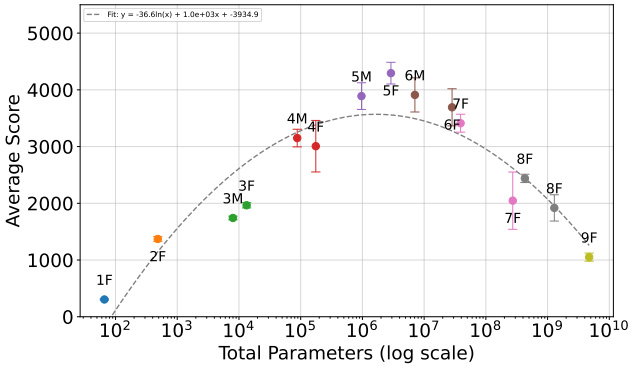
4.　実験

前節で説明した方法により，15 種類の N タプルネットワークのそれぞれについて，Optimistic initialization の初期値 OI を 3 通り変えて学習を行った．ランダム性の影響を抑えるため，各条件について乱数のシードを変えて 10 回の学習を行い，10 個の N タプルネットワークを得た．次に，各 N タプルネットワークに対し，Greedy プレイおよび Expectimax 探索（深さ 2〜6）により 1000 ゲームのプレイを行い，それらの平均スコアを求めた．本節のグラフにおいて，10 個の N タプルネットワークの平均を点や線で示し，それらの標準偏差をエラーバー等で示す．

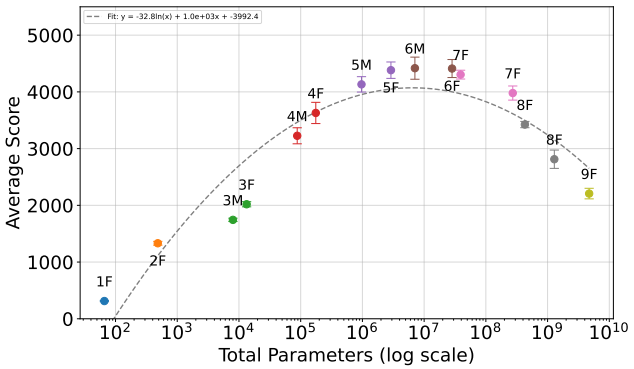
4.1　スコアとパラメータ数の関係

第 1 節で示した RQ1 について考察するため，Greedy プレイのスコアを，Optimistic initialization の初期値（OI）ごとにプロットしたものが図 3a から図 3c である．これらのグラフは，横軸にパラメータ数の対数をとり，縦軸にスコアの平均値と標準偏差をプロットしている．また，それぞれのグラフの点に対し，パラメータ数の対数とスコアの関係を二次関数でフィッティングして得られる近似曲線も描いている．

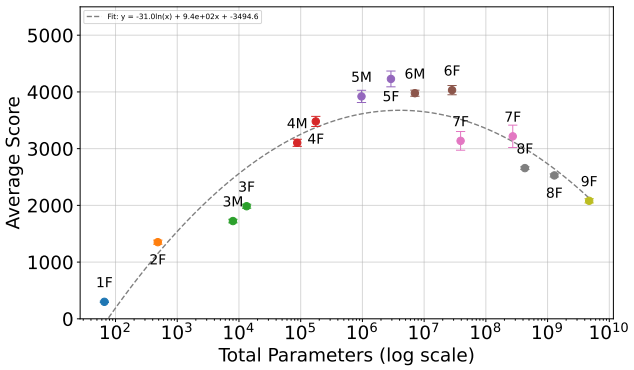
これらのグラフから，いずれのグラフもおよそ放物線を描いていることが分かる．とくに，5M から 6F の区間に放



(a) OI=0

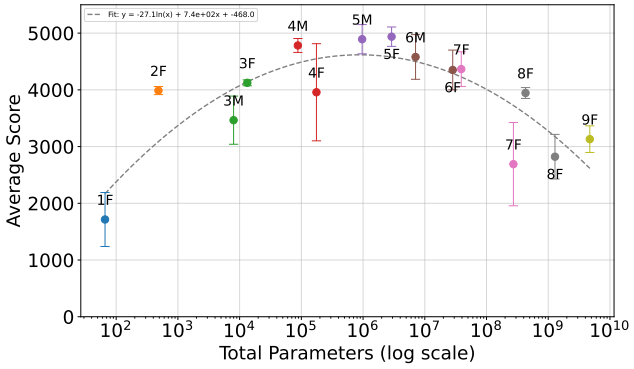


(b) OI=1200

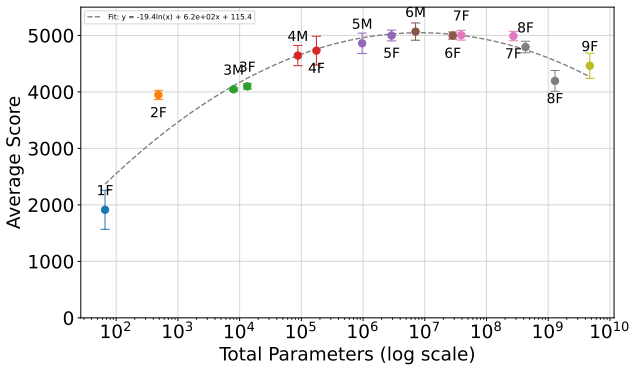


(c) OI=5400

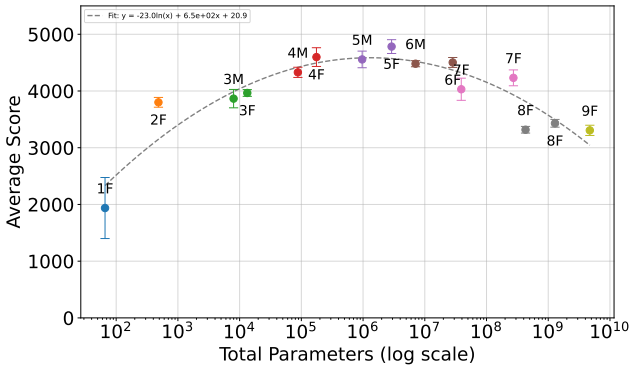
図 3: Greedy プレイにおいて，パラメータ数と平均スコアの関係



(a) OI=0



(b) OI=1200



(c) OI=5400

図 4: Expectimax 深さ 6 において，パラメータ数と平均スコアの関係

物線の頂点が位置することが確認できた．また，$OI = 0$ の場合（図 3a），3 種類の初期値の中で標準偏差が大きいものが目立っている．このことは，Optimistic initialization を行わない学習では，学習の幅広さが足りず，安定的に良い結果が得られないことを意味する．$OI = 1200$ と $OI = 5400$ の場合（図 3b，図 3c），スコアのばらつきは小さい．また，より多くのパラメータ数のところまでスコアの向上が見られる（すなわち，放物線の頂点が右に移動する）．5F よりもパラメータ数の少ない N タプルネットワークでは，平均スコアは初期値にそれほど依存していない．一方，それよりも多くのパラメータを持つ N タプルネットワークでは，初期値が $OI = 1200$ の場合に最も良い結果が得られた．

次に，RQ1 と RQ3 について考察するため，深さ 6 の Expectimax 探索を行った場合の，N タプルネットワークのパラメータ数と平均スコアを関係を図 4a から図 4c に示す．各グラフの近似曲線から，いずれのグラフもおよそ放物線を描いていること，いずれの平均スコアも Greedy プレイのスコアよりも高いことが確認できた．

$OI = 0$ の場合，探索を行ってもスコアのばらつきはそれほど小さくならず，特に Full のものについてばらつきが大きい傾向が見られる．これはパラメータ数が多い方が評価値の修正が起こり難く，局所最適解から抜け出しにく

いのではないかと考えられる．

$OI = 1200$ の場合，5M から 7F まで同程度の平均スコアを達成しており，放物線の上昇と下降の傾きが小さい．これは，強いプレイヤが達成しうるスコアの上限に近づいていて向上の余地が小さいことと，弱いプレイヤが探索によってスコアを上昇させられることを示唆する．$OI = 5400$ の場合には，スコアのばらつきは小さいものの，放物線の形やスコアの最大は $OI = 0$ の場合のそれらとあまり変わらなかった．

4.2 学習の進み方の比較

RQ2 について考察するため，5M と 7M を例にとり，学習過程のスコアの推移を確認した．図 5 は，横軸に学習ステップ数を，縦軸にスコアをとり，学習ステップ 10000 ごとに学習エピソードの平均をプロットしたものである．

Optimistic initialization の初期値を $OI = 0$ と設定した場合，5M では学習が進むにつれてスコアが上昇しているが，7F ではスコアが上昇していない．これは 7F が局所最適解にハマっていることを示唆する．

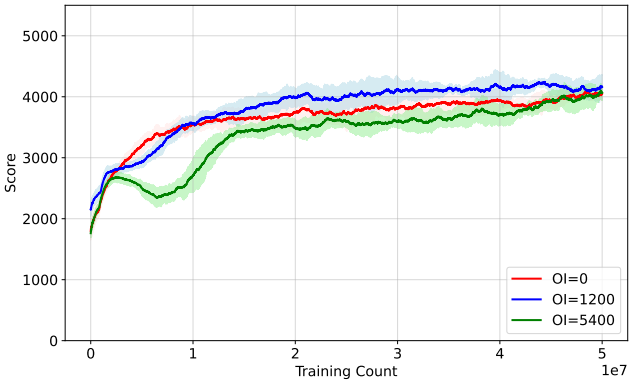
初期値を $OI = 1200$ と設定した場合，5M と 7F の両方でスコアの上昇が $OI = 0$ の場合よりも緩やかになった．詳しく見ると，5M では，約 2800 点を達成したあたりで途中一度停滞しており，その後再びスコアの上昇に転じている．表 1 より，約 2800 点というのは 256 タイルが完成してから 512 タイルが完成するまでの間であることが分かる．この間の盤面では空きタイルが多くあるため，学習に出現する盤面の種類が多いことが停滞の原因ではないかと考えている．また，7F において初期値を $OI = 1200$ と設定した場合，序盤のスコアの上昇が緩やかになっているが，これも同じ原因ではないかと考える．

初期値を $OI = 5400$ と設定した場合，5M と 7F の両方でスコアの上昇が緩やかになり，途中で停滞が発生している．5M では停滞を乗り越えて大きくスコアを上昇させることに成功しているが，7F では停滞が長引いてしまい結果として学習不足であることが判明した．

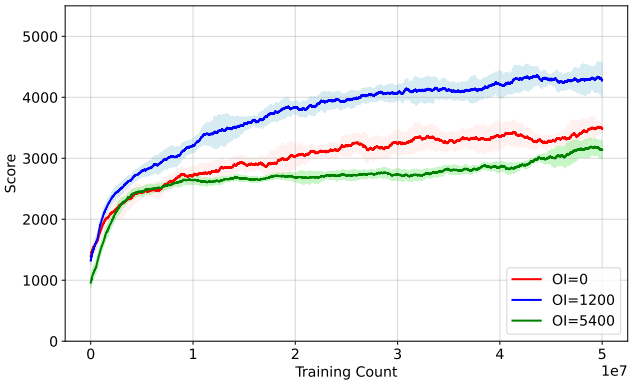
4.3 パラメータ数の増加によるプレイヤの挙動の変化

OI の初期値 1200 のスコアが同等でパラメータ数が違う 4F と 8M，5M と 7F のプレイヤをパーフェクトプレイヤを用いて詳細な比較を行い，パラメータ数の増加がプレイヤの挙動にどのような影響を与えるのかについて詳しく調べて行く．指標としては，正確度，絶対誤差，生存率を用いる．

- 正確度：パーフェクトプレイヤの選択した手とプレイヤの選択した手の一致率
- 絶対誤差：パーフェクトプレイヤの選択した手とプレイヤの選択した手のパーフェクトプレイヤ評価値の差
- 生存率：ある progress において，プレイヤが生存している確率



(a) 5M の学習過程のスコアの変化

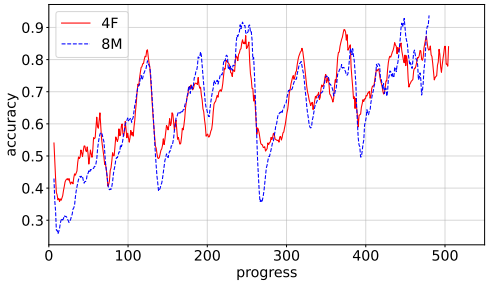


(b) 7F の学習過程のスコアの変化

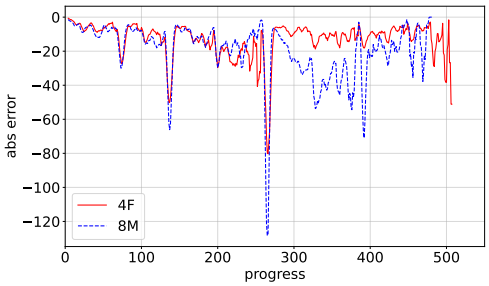
図 5: 5M と 7F の学習推移の比較

まず初めに図 6 は，4F と 8M のプレイヤの Greedy プレイの比較を示している．図 6a は，を見ると正確度のグラフはどちらも同じような形をしているが，8M の方が上下に大きく変動していることが分かる．図 6b を見ると 4F の方が 8M 多くの progress で絶対誤差が小さいことが分かる，これは progress260 を超えた辺りから顕著に現れている．progress260 辺りは 512 のタイルが完成し以前と似ている盤面になるのだが，8M タプルサイズが大きくなることで汎化性能が落ちて，序盤の盤面と全く別物として学習してしまい，512 タイルが完成した後の盤面に対して正確度が下がり，絶対誤差も大きくなってるのではないかと考えられる．これは図 6c の生存率にも表れていて生存率ではミスをした後の progress300 を超えた辺りから顕著に生存率が 8M の生存率が下がっている．

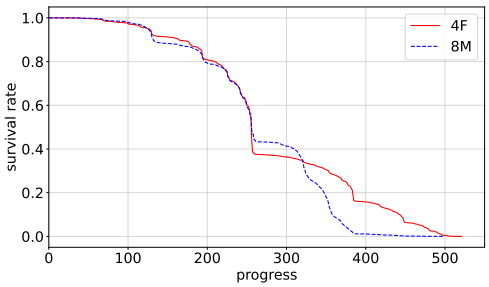
次に図 7 は，5M と 7F のプレイヤの Greedy プレイの比較を示している．図 7a を見ると形は似ているが，両方の正確度が下がる場面で 7F の方が正確度が下がっているのが分かる．図 7b を見ると，どちらもほぼ同じ形になっていて，正確度ほど差のないグラフになっているこれは 7F は間違えても問題ない手を選んでいるだけで学習自体は十分に成功していることがわかる．図 7c を見るとどちらも



(a) 正確度

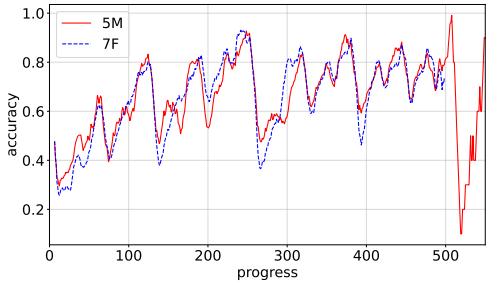


(b) 絶対誤差

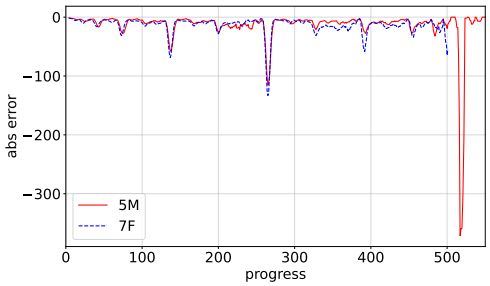


(c) 生存率

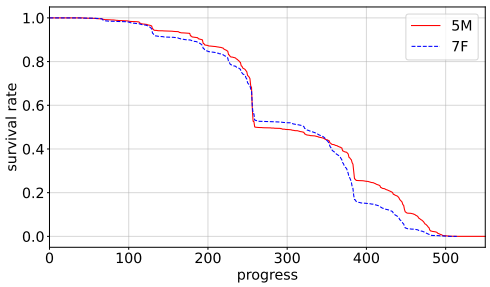
図 6: 4F と 8M の比較結果



(a) 正確度



(b) 絶対誤差



(c) 生存率

図 7: 5M と 7F の比較結果

上下を入れ替わりながら似たような形になっていることが分かる．

図 8 と図 9 は，図 6 と図 7 の Expectimax 探索深さ 6 を組み合わせたプレイヤである．それぞれのスコアとしては向上していてそれは図 8b と図 9b の絶対誤差と図 8c と図 9c の生存率に現れている．図 8a と図 9a の正確度はどちらも形は Greedy と似たような形になってる．

5. 考察

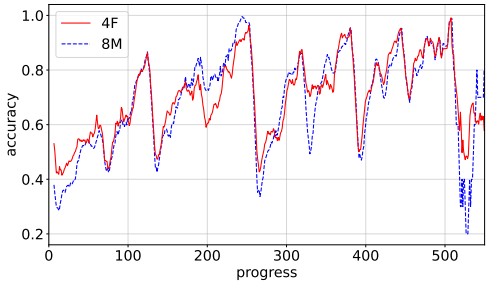
ミニ 2048 において N タプルネットワークのパラメータ数とスコアの関係について詳細な分析を行った結果パラメータ数の log とスコアの関係は方物線的であることが確認された．2048 においても同様の傾向だとすると 1 から 16 までのタプルのなかでミニ 2048 の 5 か 6 に匹敵するのは 8 タプル,9 タプル,10 タプルあたりであると考えられる．本研究では図 6 と図 7, 図 8 と図 9 の比較でスコアが同程度の場合のパラメータ数の差がある場合の挙動について分析をしたが，グラフの形こそ同じようになるもののある部分が致命的に弱いというようなことはなかったので，2048 で 8 タプル,9 タプル,10 タプルを用いた場合の学習性能は既存研究の 6 タプルを上回るようなスコアを期待できる．

6. まとめ

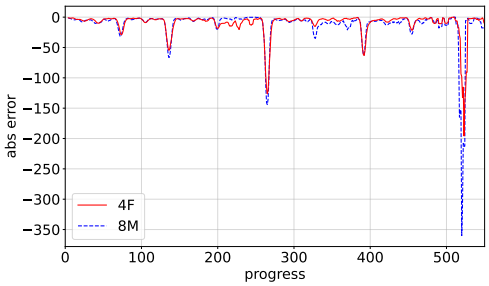
本研究では，ミニ 2048 を用いて，ミニ 2048 における N タプルネットワークのタプルサイズおよび Optimistic Initialization（OI）の初期値が，プレイヤの学習性能および探索によるスコアに与える影響について詳細に分析した研究の実施にあたって 3 つのリサーチクエスチョン（RQ）を設定し，実験を行った．

- **RQ1** N タプルネットワークにおけるタプルの大きさ，数，パラメータ数のスコアへの影響
- **RQ2** Optimistic Initialization の初期値による N タプルネットワークの学習への影響
- **RQ3** N タプルネットワーク評価関数と Expectimax 探索の組合せにおける性能向上の依存関係

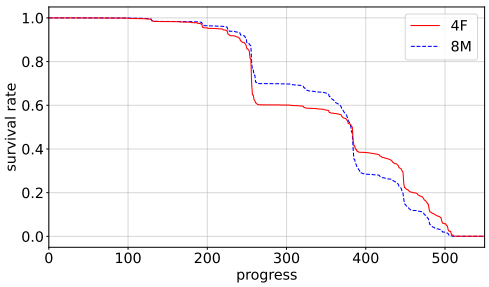
実験の結果，以下の重要な知見が得られた



(a) 正確度

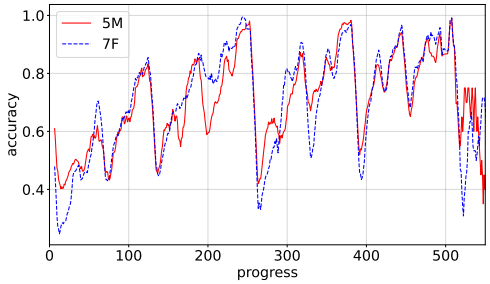


(b) 絶対誤差

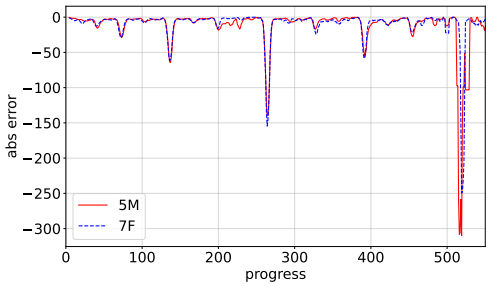


(c) 生存率

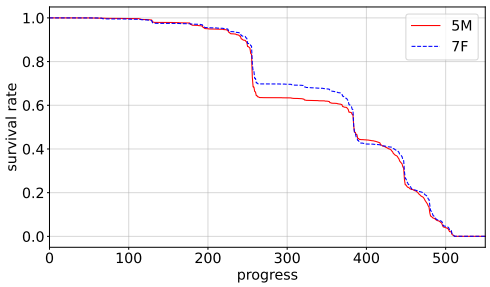
図 8: 4F と 8M の比較結果 (深さ 6)



(a) 正確度



(b) 絶対誤差



(c) 生存率

図 9: 5M と 7F の比較結果 (深さ 6)

- **RQ1 について：**パラメータ数の対数とスコアの関係は放物線的であり，タプルサイズ 5 から 6 付近で性能が最大となることが確認された．これは，2048 で広く用いられている 6 タプルによる評価関数の妥当性を裏付ける結果となった．
- **RQ2 について：**OI の初期値が 0 の場合は局所最適解への収束やスコアのばらつきが大きく，学習の安定性に影響を与えることが確認された．一方，初期値が大きすぎる場合は学習が停滞する可能性も示された．適切な初期値（1200 程度）を設定することで，より多くのパラメータを効果的に活用できることが示された．
- **RQ3 について：**Expectimax 探索による性能向上は，評価関数のパラメータ数や OI の初期値に依らず一貫して効果的であることが確認された．探索深さ 6 では，どの評価関数でもパーフェクトプレイとの差が約半分に縮まることが示された．

今後の課題としては，これらの知見を 2048 に適用し，より大きなタプルサイズでの性能評価や，Multistaging によってパラメータ数を変化させた場合の学習性能を評価することが挙げられる．また本研究で得られた知見に基づき，2048 プレイヤを実装することでより，高性能なプレイヤの実現が期待できる．

謝辞　本研究は JSPS 科研費 JP23K11383 の助成を受けたものである．

参考文献

[1] Cirulli, G.: 2048, `http://gabrielecirulli.github.io/2048/` (2014).

[2] Guei, H., Chen, L.-P. and Wu, I.-C.: Optimistic Temporal Difference Learning for 2048, *IEEE Transactions on Games*, Vol. 14, No. 3, pp. 478–487 (2022).

[3] Jaśkowski, W.: Mastering 2048 with Delayed Temporal Coherence Learning, Multi-Stage Weight Promotion, Redundant Encoding and Carousel Shaping, *IEEE Transactions on Computational Intelligence and AI in Games*, Vol. 10, No. 1, pp. 3–14 (2018).

[4] Kaplan, J., McCandlish, S., Henighan, T., Brown, T. B., Chess, B., Child, R., Gray, S., Radford, A., Wu, J. and Amodei, D.: Scaling Laws for Neural Language Mod-

els, *arXiv preprint arXiv:2001.08361*, (online), available from ⟨https://arxiv.org/abs/2001.08361⟩ (2020).

[5] Matsuzaki, K.: Systematic Selection of N-tuple Networks with Consideration of Interinfluence for Game 2048, *Proceedings of the 2016 Conference on Technologies and Applications of Artificial Intelligence (TAAI 2016)* (2016).

[6] Matsuzaki, K.: Developing 2048 Player with Backward Temporal Coherence Learning and Restart, *Proceedings of Fifteenth International Conference on Advances in Computer Games (ACG2017)*, pp. 176–187 (2017).

[7] Matsuzaki, K.: A Further Investigation of Neural Network Players for Game 2048, *Proceedings of Sixteenth International Conference on Advances in Computer Games (ACG2019)* (2019). Submitted to final publication.

[8] Oka, K. and Matsuzaki, K.: Systematic Selection of N-tuple Networks for 2048, *Proceedings of 9th International Conference on Computers and Games (CG2016)*, Vol. LNCS 10068, Springer, pp. 81–92 (2016).

[9] Szubert, M. and Jaśkowski, W.: Temporal Difference Learning of N-Tuple Networks for the Game 2048, *2014 IEEE Conference on Computational Intelligence and Games*, pp. 1–8 (2014).

[10] Terauchi, S., Kubota, T. and Matsuzaki, K.: Using Strongly Solved Mini2048 to Analyze Players with N-tuple Networks, *2023 International Conference on Technologies and Applications of Artificial Intelligence (TAAI 2023)* (2023).

[11] Wu, I.-C., Yeh, K.-H., Liang, C.-C., Chang, C.-C. and Chiang, H.: Multi-Stage Temporal Difference Learning for 2048, *Technologies and Applications of Artificial Intelligence*, Lecture Notes in Computer Science, Vol. 8916, pp. 366–378 (2014).

[12] Yeh, K.-H., Wu, I.-C., Hsueh, C.-H., Chang, C.-C., Liang, C.-C. and Chiang, H.: Multi-stage temporal difference learning for 2048-like games, *IEEE Transactions on Computational Intelligence and AI in Games*, Vol. 9, No. 4, pp. 369–380 (2016).

[13] 山下修平，金子知適，中屋敷太一：3 × 3 盤面の 2048 の完全解析と強化学習の研究，第 27 回ゲームプログラミングワークショップ (GPW-22)，pp. 1–8 (2022).

[14] 寺内俊輔，松崎公紀：ミニ 2048 の完全解析を用いた N タプルネットワーク+Expectimax 探索プレイヤの分析，情報処理学会プログラミング・シンポジウム予稿集 (プログラミング・シンポジウム予稿集)，pp. 83–90 (2024).

[15] 寺内俊輔，松崎公紀：ミニ 2048 の完全解析を用いた N タプルネットワーク+モンテカルロ木探索プレイヤの分析，情報処理学会プログラミング・シンポジウム予稿集 (プログラミング・シンポジウム予稿集) (2025).