

# 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<sup>a)</sup> 松崎 公紀<sup>b)</sup>

**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ーが作られている。本研究では、ミニ 2048 において構築可能な 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形状を用いて、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。特に最先端プレイヤーの開発で用いられた手法の観点から、その効果と特性について考察を行う。

**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の一人ゲームである [1]。これまでに、2048 のさまざまなコンピュータプレイヤーが作られてきた。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 [2] と Expectimax 探索 [3] 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 [3], [4], [5], [6]。Guei によって作られた最先端プレイヤーは [6] では Expectimax 探索の深さが 6 の場合に平均得点 625 377 を達成した。著者らの研究の大きな目標は、それらの AI 技術のそれぞれについて、どのように働いているかをより詳細に分析し説明することである。本研究では、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、プレイヤーの性能および探索によるスコアに与える影響について実験的に評価した。

本研究の主な貢献は以下の 4 点である：

- N タプルネットワークのタプルサイズおよび数 (= パラメータ数) が、スコアに与える影響を定量的に評価した。
- Optimistic Initialization (OI) の初期値が、学習の安定性とスコアに与える影響を評価した。

- 探索 (Expectimax) による強化が、N タプルの構成および OI 設定とどう相互作用するかを明らかにした。
  - タプル数や OI による性能変化を可視化・解析し、設計指針の定量的な知見を提供した。
- 実験の結果、以下の知見が得られた。
- パラメータ数とスコアの関係は方物線的であり、NT5～NT6 付近が性能のピークであった。
  - OI の初期値が小さすぎると学習が不安定になり、大きすぎると学習が進まない傾向があった。
  - 適切な OI の設定により、有効な学習が進み、パラメータ数が増えてもスコアが向上する構成が見られた。
  - Expectimax 探索はスコア全体を底上げするが、OI の影響は限定的であった。

以上より、N タプルネットワークの構成や初期化戦略が 2048 型ゲームにおける AI 性能に大きく影響することを示し、今後の高性能プレイヤー設計に向けた基礎的な知見を提供する。

## 2. ミニ 2048

本研究はミニ 2048 [7] を研究対象として使用する。ミニ 2048 は、 $3 \times 3$  の盤面でプレイされることを除いて、確率の一人ゲーム 2048 と同じゲームである。

### 2.1 ルール

ミニ 2048 は、 $3 \times 3$  の盤面でプレイされる。初期局面は 9 マスの内どこか 2 マスに 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、プレイヤーは上下左右いずれかの方向を

<sup>†1</sup> 現在、高知工科大学  
Presently with Kochi University of Technology  
Presently with

<sup>a)</sup> 295141a@gs.kochi-tech.ac.jp

<sup>b)</sup> matsuzaki.kiminori@kochi-tech.ac.jp

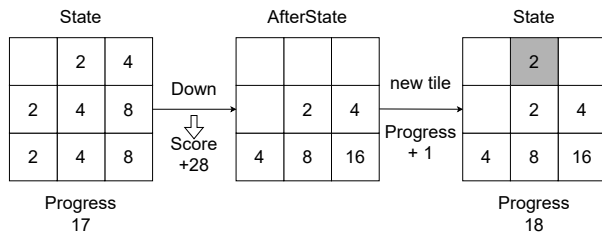


図 1: state, afterstate, progress の例

選択する．すると各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  $\_2\_$ ,  $22\_$ ,  $422$  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，空白のマスの中のランダムな 1 マスに 2 (確率 0.9) か 4 (確率 0.1) のタイルが置かれる．

プレイヤー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．このゲームの目標はゲームが終了する前に出来るだけ高得点を獲得することである．

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

**afterstate** プレイヤー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標を用いる．

**progress** タイルの値の合計値の半分を *progress* [8] 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

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%

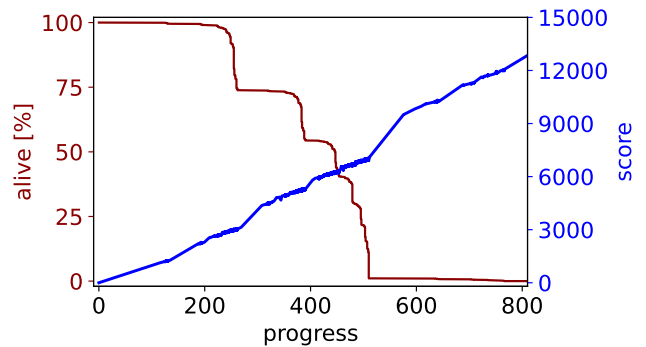


図 2: パーフェクトプレイヤーの生存率とスコア [8]

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 [7] 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 [8] 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ーを実現できる．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ーであっても，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ーが 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ーが 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ーでも，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ー

#### 3.1 N タプルネットワーク評価関数

2048 における最も成功したプレイヤーの多くは、N タプルネットワークに基づく評価関数を強化学習によってチューニングするアプローチを採用している [2]. Guei らの最新のプレイヤー [6] も、Matsuzaki [9] が提案したタプルの組合せをベースに、Expectimax 探索や Multistaging[3], Optimistic Initialization, Tile Downgrading[6] などの改良を加えることで高い性能を達成している。

本研究では、こうした知見を踏まえ、ミニ 2048 において 1 タプルから 9 タプルまでの N タプルネットワークを構築可能な全てのタプル列挙を行い、

- M (Manual) : 人間の知見に基づいて、効果的と考えられるタイル位置のみを選択して構成したタプル集合。NT3M などの”M”はこれを表す
- F (Full) : 指定したサイズのタプルについて、盤面上で連続するタイルの全ての組み合わせを使用したタプル集合。NT1F などの”F”はこれを表す

この 2 つの設計方針に基づき、タプル数 (N) を 1 から 9 まで変化させてネットワークを構築した。ここで、プレイヤー名の数字はタプル数を、末尾の M と F はそれぞれ Manual と Full の設計方針を表している。

この結果、最大で 18 種類の N タプルネットワークが得られたが、1 タプル、2 タプル、9 タプルにおいては両方の設計が一致したため、プレイヤーの種類としては 15 通りとなった。これらのプレイヤーを用いて、N タプルネットワークの大きさ (タプル数) と学習性能の関係を実験的に評価した。

表 2 に、本研究で使したタプルの構造を示す。また、ミニ 2048 の盤面の持つ対称性 (回転・反転) \*\*を活用し、1 つのタプルに対して対称な 8 通りの位置からのサンプリングを行うことで、学習に必要なタプル数の削減を図っている。

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 学習は学習率自動調整機能を備えた TD 学習で、Jaśkowski [5] が始めて 2048 に導入した。

**Optimistic initialization (OI)** 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

表 2: タプルサイズと形状の一覧

タプル名	タプルの形状
NT1F	
NT2F	
NT3M	
NT3F	
NT4M	
NT4F	
NT5M	
NT5F	
NT6M	
NT6F	
NT7M	
NT7F	
NT8M	
NT8F	
NT9F	

ての afterstate の初期値が 1200 になるように重みを初期化してある。

それぞれの N タプルニューラルネットワークに対して、 $5 \times 10^8$  局面分のデータで学習を行った。いずれのニューラルネットワークも、十分に学習が収束していることを確

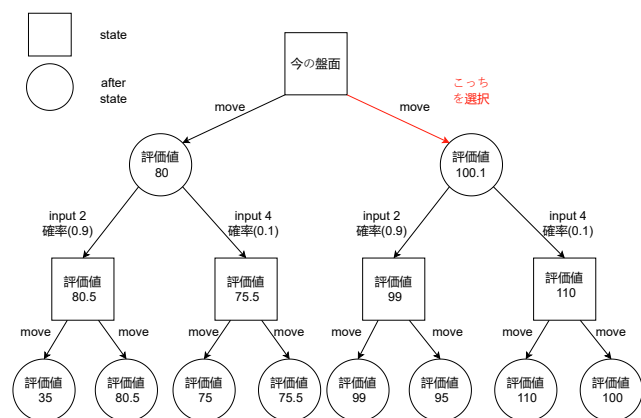


図 3: 深さ 2 の Expectimax 探索の例

認してある [8].

## 4. Expectimax 探索

Expectimax 探索は，確率的一人ゲームにおける標準的な探索手法である．ミニ 2048 のゲームの進行は，state におけるプレイヤーの選択と，afterstate における新規タイルの出現が交互に起こる．したがって，ミニ 2048 のゲーム木は，根が現在の state に対応し，根から葉への各パス上に，afterstate に対応するノード（chance ノード）と state に対応するノード（max ノード）が交互に現れる．本研究では，ミニ 2048 のゲーム木の高さ（探索の深さ）を，各パス上の afterstate に対応するノードの数と定める．例えば，高さ 2 のミニ 2048 のゲーム木は，根，afterstate に対応するノードの層，state に対応するノードの層，afterstate に対応するノードの層，の合計 4 層からなる（図 3）．

Expectimax 探索では，ゲーム木の各ノードに対して次のように再帰的に計算を行う．

- max ノードでは、子要素の値のうちの最大値を計算する。
- chance ノードでは、子要素の値を、その出現確率を用いた重み付き平均を計算する。

Expectimax 探索プレイヤは、Expectimax 探索によって得られた子ノードのうち、評価値の最も大きなものを選択する。

図 3 に深さ 2 の Expectimax 探索の例を示す.

ミニ 2048 のゲーム木では、特に、同じ afterstate が複数出現する。そのような同じ afterstate をまとめる（合流）工夫を実装した。ただし、合流を考慮しない Expectimax と結果が一致するよう、同じ afterstate であってもゲーム木中の深さが異なる場合には別のものとして扱った。この工夫により、特に深い探索において大幅な高速化が実現された。本研究では [10] で実装した Expectimax 探索を用いた。

## 5. 実験

本研究では、ミニ 2048 における N タプルネットワークの学習性能を多角的に分析するため、構造の異なる複数のプレイヤを構築し、Greedy および Expectimax 探索による評価を行った。

2で示した15種類のプレイヤーを用いて評価を行った。

さらに、Optimistic Initialization (OI) の影響を調べるため、各プレイヤについて OI の初期値を 0, 1200, 5400 に設定し、それぞれ学習を実施した。すべてのプレイヤは  $5 \times 10^8$  手分の行動に基づいて学習を行い、乱数シードを変えて 10 体ずつ学習させた。結果として、15 タプル構成  $\times 3$  回 (OI の初期値)  $\times 10$  回 (シード) で、計 450 体のプレイヤが作成された。

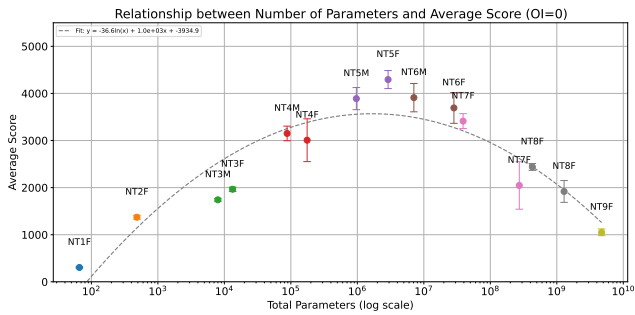
これらのプレイヤに対して，Greedy プレイおよび Expectimax 探索（深さ 2～6）による 1000 ゲームプレイを行い，seed 違いの結果をまとめた 10000 ゲームのログを用いて解析を行なった。

### 5.1 スコアとパラメータ数の関係

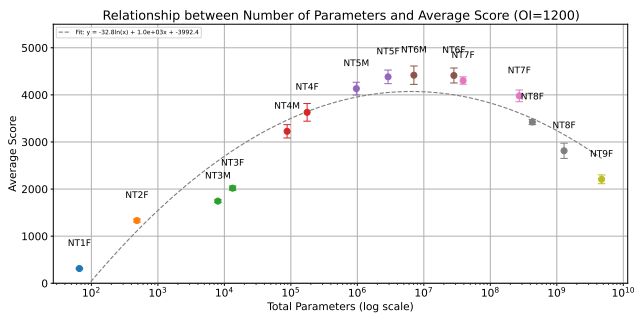
まず初めに図 4a から図 4c は、OI の初期値を変えた場合の Greedy プレイでのパラメータ数の対数とスコアの関係を示している。グラフの点が seed ごとの平均スコアの平均、点から伸びる線が seed ごとの平均スコアの標準偏差を示している。

いずれのグラフも放物線を描いており、NT5 か NT6 を頂点とすることが確認できた。また、図 4aOI の初期値を 0 に設定した場合、3 種類の OI の初期値の中で平均の標準偏差が大きいものはいくつか目立ち学習の安定性に影響を与えることが確認できた。OI の初期値を 1200,5400(図 4b, 図 4c) に設定した場合、スコアのばらつきは小さくなり、スコアの向上がパラメータ数を増加させても続き、放物線の頂点が右に移動する傾向が見られた。平均の平均自体はあまり大きく変化しなかったが、図 4bOI の初期値を 1200 に設定したものが良い傾向にあった。

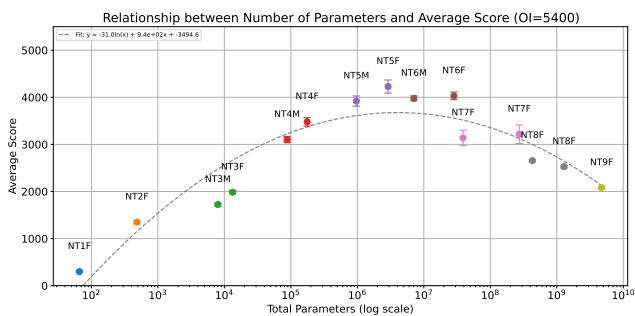
図 5a から図 5c は、OI の初期値を変えた場合の各プレイヤと Expectimax の深さ 6 を組み合わせたプレイヤでの、パラメータ数の対数とスコアの関係を示している。いずれのグラフも放物線を描いていること、いずれの平均の平均のスコアも Greedy プレイのスコアよりも高いことが確認できた。また、OI の初期値を 0 に設定した場合、探索を加えてもスコアのバラつきが多いプレイヤが見られた、特に Full タブルのスコアのバラつきが大きいことが確認できた。これはパラメータ数が多い方が評価値の修正が起こり難しく、局所最適解から抜け出しにくいのではないかと考えられる。また OI の初期値を 1200 に設定したばあい NT5M から NT7F までスコアにほとんど差がなく、放物線の上昇



(a) タプル数の変化によるスコアの変化 (OI=0)



(b) タプル数の変化によるスコアの変化 (OI=1200)



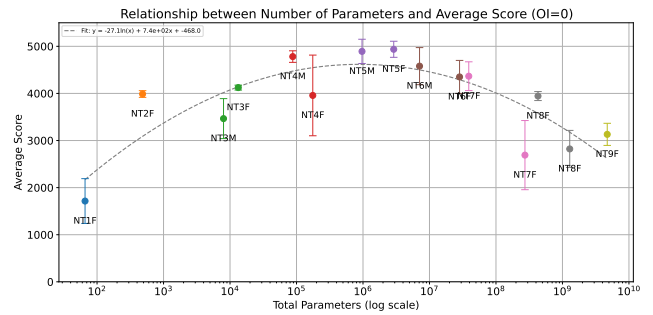
(c) タプル数の変化によるスコアの変化 (OI=5400)

図 4: OI の初期値ごとのタプル数の変化によるスコアの変化

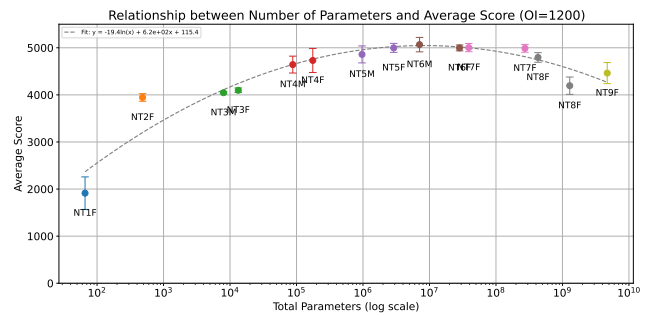
と下降の傾きが小さくなっていることが確認できた。これは強いプレイヤーは N タプルプレイヤーの現実的なスコア上限に近づいていて、ほとんど上昇しないことと、弱いプレイヤーは探索によってスコアが上昇することを示している。また OI の初期値を 5400 に設定した場合、平均の標準偏差はとても小さくなったが、放物線の形やスコアの上限は OI の初期値 0 の時とあまり変わらなかった。

図 6 は、NT5M と NT7F の学習の際のスコアの推移を示している。このグラフでは y 軸にスコア、x 軸に学習回数を取り、各学習回数ごとのスコアの平均を求め、それを学習回数 10000 ごとに移動平均を取り、グラフにプロットしている。

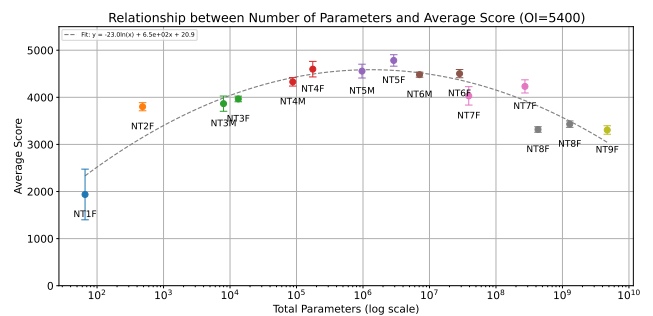
OI の初期値を 0 に設定した場合 (図??, 図 6b), NT5M は学習が進むにつれてスコアが上昇しているが、NT7F はスコアが上昇しないことが確認できた。これは NT7F が局



(a) タプル数の変化によるスコアの変化 深さ 6 (OI=0)



(b) タプル数の変化によるスコアの変化 深さ 6 (OI=1200)



(c) タプル数の変化によるスコアの変化 深さ 6 (OI=5400)

図 5: Expectimax 深さ 6 での OI の初期値ごとのタプル数の変化によるスコアの変化

所最適解にハマっていることを示している。OI の初期値を 1200 に設定した場合 (図 6c, 図 6d), NT5M と NT7F の両方のスコアの上昇が緩やかになり、NT5M の場合学習途中スコアとしては約 2800 点ほどで一度停滞が入り、その後再び上昇に転じている。このスコアの停滞しているのは表??を見ると 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。この停滞期の盤面は盤面に比較的空きタイルがあり、色んな手を選択することで効率の悪い手を選んでしまい停滞しているのではないかと考えている。また、NT7F は OI の初期値を 1200 に設定した場合、序盤のスコアの上昇が緩やかになっているのも同じ原因ではないかと考える。OI の初期値を 5400 に設定した場合 (図 6e, 図 6f), NT5M と NT7F の両方のスコアの上昇が緩やかになり、両方とも途中で停滞が発生している。NT5M は停滞を乗り越えて大きくスコアを上昇させている



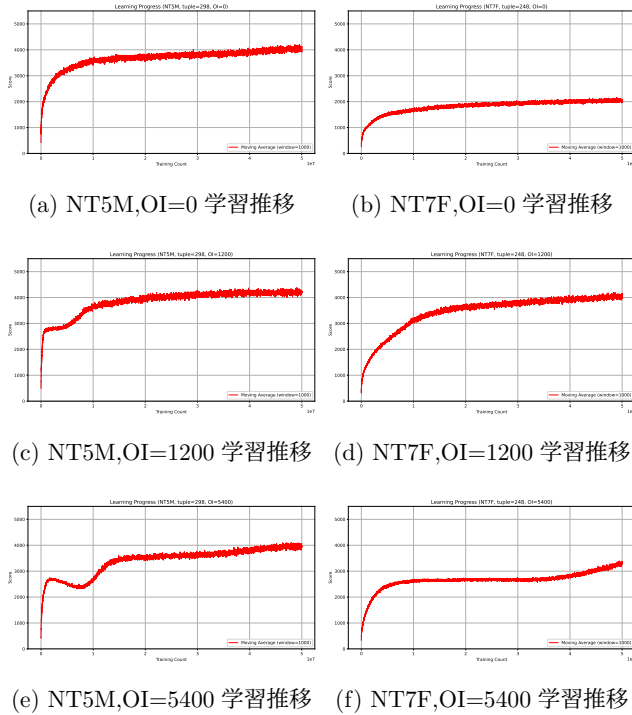


図 6: NT5M と NT7F の学習推移の比較

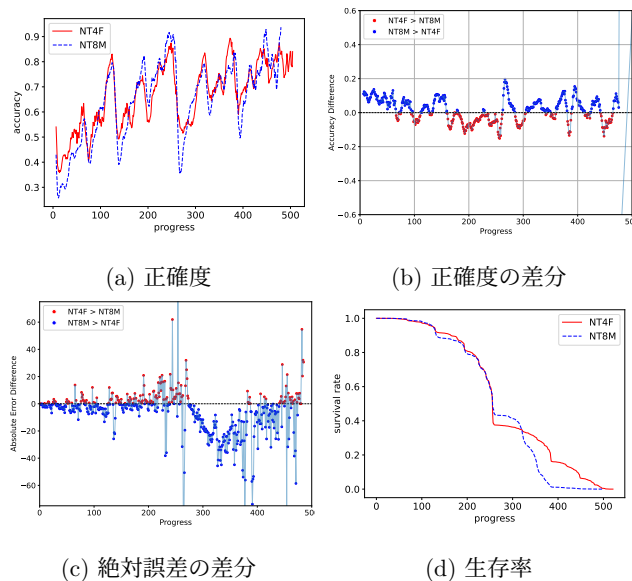


図 7: NT4F と NT8M の比較結果

が、NT7F は停滞が長引き大きなスコア上昇にまで至っていないので、学習不足であると考えている。

## 5.2 パラメータ数の増加によるプレイヤーの挙動の変化

ここから OI の初期値 1200 のスコアが同等でパラメータ数が違う NT4F と NT8M, NT5M と NT7F のプレイヤーを比較して、パラメータ数の増加がプレイヤー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。

まず初めに図 7 は、NT4F と NT8M のプレイヤーの Greedy プレイの比較を示している。7a と 7b は、NT4F と NT8M

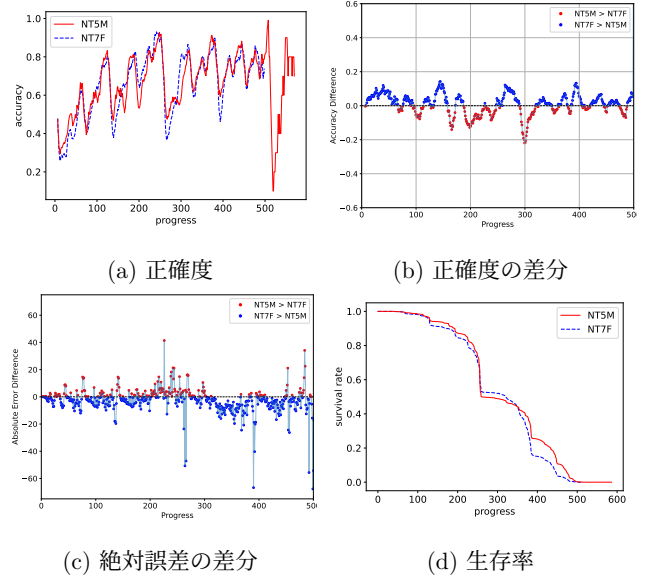


図 8: NT5M と NT7F の比較結果

のプレイヤーの正確度を示している。これを見ると序盤は NT4F の方が正確度が高いが、中盤以降はどちらも抜きつ抜かれつ状態であることが分かる。7c は、NT4F と NT8M のプレイヤーの絶対誤差の差分を示している。これを見ると、NT4F の方が NT8M 多くの progress で絶対誤差が小さいことが分かる。正確度のグラフと絶対誤差のグラフを合わせてみると、NT4F の方が後半かなり良いことがわかる。これは 7d の生存率のグラフにも表れている。

次に図 8 は、NT5M と NT7F のプレイヤーの Greedy プレイの比較を示している。8a と 8b を見ると NT5M の方が NT7F よりも正確度が高いことが分かる。8c は、NT5M と NT7F のプレイヤーの絶対誤差の差分を示している。これを見ると、NT5M の方が NT7F よりも絶対誤差が小さい progress が多いことが分かる。8d は、NT5M と NT7F のプレイヤーの生存率を示している。これを見るとどちらも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 7 と図 8 は、Greedy プレイの結果を示しているが、どちらの結果からも特徴的な形は見られなかった。

図 9 と図 10 は、図 7 と図 8 の Expectimax 探索深さ 6 を組み合わせたプレイヤーである。図 9a と図 9b を見ると正確性は NT4F の方が良い図 10a と図 10b を見ると正確性は NT5M の方が良い図 9c と図 10c は絶対誤差の差分を示していて、ここは探索を加えたことで上下の振れ幅がとても小さくなっていることが分かるが結局どちらの結果からも特徴的な形は見られなかった。

## 6. 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形的であることが確

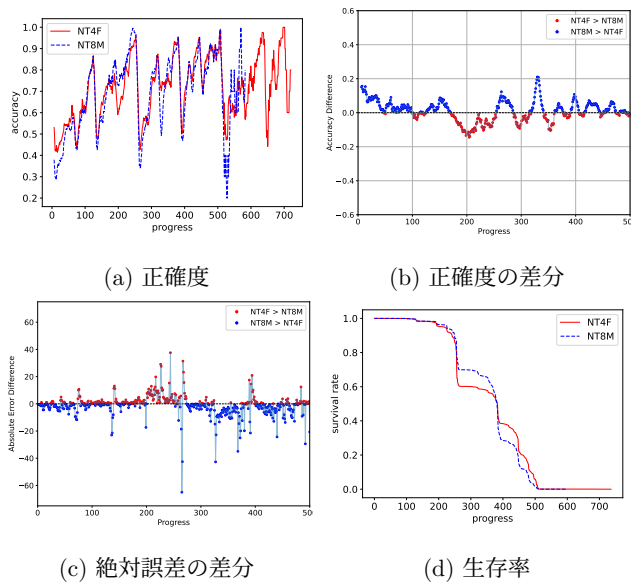


図 9: NT4F と NT8M の比較結果 (深さ 6)

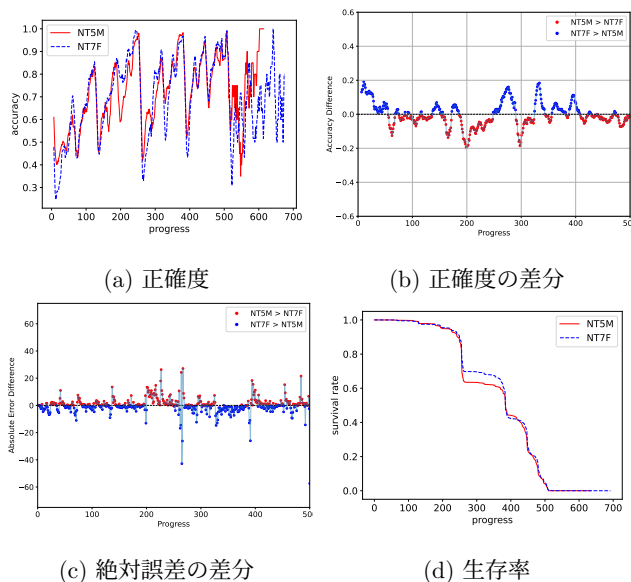


図 10: NT5M と NT7F の比較結果 (深さ 6)

認された。2048 においても同様の傾向だとすると NT1 から NT16 までのタプルのなかでミニ 2048 の NT5 か NT6 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 7 と図 8, 図 9 と図 10 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。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。

## 7. まとめ

本研究では, 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。実験の結果, 以下の重要な知見が得られた:

- パラメータ数の増加とスコアの関係は方物線的であり, ミニ 2048 では NT5 か NT6 を頂点とすることが確認された。
- OI の初期値は 0 ではスコアのばらつきが大きく, 局所最適解にハマるのか, 学習の安定性に影響を与えることが確認された。しかし大きな値を設定しすぎると学習が終わらない可能性も示された。
- OI の初期値を適切に設定すると, スコアの向上が続くパラメータ数が増加する傾向が確認できた。
- 探索を組み合わせることでもどのパラメータ数でもスコアが向上することが確認された。
- 探索を実装した場合でも OI の値と平均スコアのばらつきは変わらなかった。

以上の結果からミニ 2048 においてパラメータ数と N タプルネットワークのサイズがスコアに与える影響と Optimistic Initialization (OI) の初期値がスコアに与える影響を明らかにした。今後の課題としては, これらの結果を踏まえて, 2048 で NT1 から NT9 までのタプルを用いた場合の学習性能を評価することと, 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。それによって得られたタプルを Expectimax 探索に組み込むことで, より高いスコアを達成することが挙げられる。

**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] G. Cirulli. 2048. <http://gabrielecirulli.github.io/2048/>, 2014.
- [2] M. Szubert and W. Jaśkowski. Temporal difference learning of N-tuple networks for the game 2048. In *2014 IEEE Conference on Computational Intelligence and Games*, pages 1–8, 2014.
- [3] K.-H. Yeh, I.-C. Wu, C.-H. Hsueh, C.-C. Chang, C.-C. Liang, and H. Chiang. Multi-stage temporal difference learning for 2048-like games. *IEEE Transactions on Computational Intelligence and AI in Games*, 9(4):369–380, 2016.
- [4] K. Matsuzaki. Developing 2048 player with backward temporal coherence learning and restart. In *Proceedings of Fifteenth International Conference on Advances in Computer Games (ACG2017)*, pages 176–187, 2017.
- [5] W. Jaśkowski. Mastering 2048 with delayed temporal coherence learning, multi-stage weight promotion, redundant encoding and carousel shaping. *IEEE Transac-*

- tions on Computational Intelligence and AI in Games, 10(1):3–14, 2018.
- [6] Hung Guei, Lung-Pin Chen, and I-Chen Wu. Optimistic temporal difference learning for 2048. *IEEE Transactions on Games*, 14(3):478–487, 2022.
  - [7] 山下 修平, 金子 知適, and 中屋敷 太一.  $3 \times 3$  盤面の 2048 の完全解析と強化学習の研究. In **第 27 回ゲームプログラミングワークショップ (GPW-22)**, pages 1–8, 2022.
  - [8] Shunsuke Terauchi, Takaharu Kubota, and Kiminori Matsuzaki. Using strongly solved Mini2048 to analyze players with N-tuple networks. In *2023 International Conference on Technologies and Applications of Artificial Intelligence (TAAI 2023)*, 2023.
  - [9] K. Matsuzaki. Systematic selection of n-tuple networks with consideration of interinfluence for game 2048. In *Proceedings of the 2016 Conference on Technologies and Applications of Artificial Intelligence (TAAI 2016)*, 2016.
  - [10] 寺内 俊輔 and 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 探索プレイヤーの分析. In **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)**, pages 83–90, 2024.