

縮小盤オセロ 6×6 の完全解析

— 先行研究の調査と、本研究の位置づけ —

児玉

2026 年 8 月

概要

6×6 盤オセロは後手（白）必勝であり、最善手同士では「黒 16 対 白 20」で終わることが 1993 年に知られている。本稿は、この事実の出典と、そこで報告されている計算量にかかわる数値を文献にあたって整理し、さらに自作の探索プログラムによる実測と対比する。目的は「 6×6 の完全解析を Raspberry Pi のクラスタで 1 時間以内に再現するには何台必要か」を見積もるための、信頼できる基準値を得ることである。調査の結果、広く引用される「約 400 億局面」は Feinsein の報告値であり、文献 [3] の表題そのもの (*Forty Billion Nodes Under The Tree*) に由来すること、一方で竹下ら [4] の追試は約 8844 億局面を要していること、両者は探索手法（置換表の有無）が異なるため単純に比較できないことが分かった。自作ソルバによる実測では、ビットボード・置換表・PVS の組合せにより、 6×6 盤の完全解析（石差 -4 の確定）が **41,004,075,510 局面・1990.81 秒 (33 分 11 秒)**、**1 コア**で完了した。この 410 億という値は Feinsein の報告値（約 400 億）とほぼ一致し、竹下らの 8844 億に対しては約 $1/22$ である。後手必勝の判定のみであれば 18,435,230,807 局面・925.09 秒（15 分 25 秒）で足りる。これらの実測から、 6×6 の完全解析は Raspberry Pi 5 を**数台**用いれば 1 時間以内に完了すること、そして台数よりも置換表の有無が結果を支配することを示す。

1 背景と目的

オセロは二人零和有限確定完全情報ゲームであり、理論上は先手必勝・後手必勝・引き分けのいずれかに定まる。標準の 8×8 盤は現在も未解決だが、縮小盤については完全解析が行われている。

本稿で扱う 6×6 盤は、そのなかでも最も有名な例である。 6×6 は空きマス 32 個から始まる（36 マスのうち中央 4 マスが初期配置で埋まる）。これは 8×8 盤オセロの終盤 32 手読みに相当し、現代の終盤ソルバが日常的に扱う規模を、序盤から通しで解くという性格をもつ。

本稿の目的は次の 2 点である。

1. 6×6 の解に関する数値（探索局面数・計算時間）の出典を確定し、各文献が「何を数えているのか」を区別して整理する。
2. 自作の探索プログラムで同じ問題を解き、現代のハードウェアと現代的な枝刈りを用いた場合の実測値を得る。これを Raspberry Pi クラスタの台数見積りの基礎とする。

2 調査した文献

2.1 Feinstein (1993) : 6×6 盤の解決

6×6 盤オセロが後手必勝であることは、1993 年に数学者 Joel Feinstein が示した [3]。最善手同士の終局は「黒 16 対 白 20」、すなわち後手 4 石差の勝ちである。

重要な点として、この結果は査読論文ではなく英国オセロ協会 (British Othello Federation) のニューズレター上で報告された。記事の表題は “*Forty Billion Nodes Under The Tree*” であり、しばしば引用される「約 400 億局面」という数値は、この表題そのものに由来する。Feinstein は、ムーブオーダリングを用いたアルファ・ベータ法により約 4×10^{10} の局面を、およそ 2 週間で探索したと記述している [4, 1 章]。

2.2 白濱 (2007) : 4×4 , 4×6 盤と一般化予想

宮崎大学の白濱拓也は卒業論文において、 4×4 盤が後手必勝、 4×6 盤が先手必勝であることを示し、さらに盤の大きさを $2m \times 2n$ と一般化したとき、 $m + n$ が奇数ならば先手必勝、偶数ならば後手必勝と予想した [5]。この予想は後述のとおり、竹下らによって否定される。

2.3 竹下ら (2015) : 縮小盤オセロにおける完全解析

竹下祐輝・池田諭・坂本真人 (宮崎大学) と伊藤隆夫 (広島大学) による「縮小盤オセロにおける完全解析」[4] は、本調査で得られた最も詳細な日本語文献である。 4×4 , 4×6 , 4×8 , 4×10 , 6×6 の各盤について完全解析を実行し、Feinstein の 6×6 の結果を追試している。

その結果を表 1 に示す。表の「Position」は論文中で**最終局面数** (終局まで到達した葉の数) と説明されている点に注意が必要である。

表 1 竹下ら [4] による完全解析結果 (表 4-1 より)

盤	最終局面数	実行時間	結果	終局
4×4	218	0.001 s	後手勝ち (−8)	黒 3, 白 11
4×6	139,803	0.1 s	先手勝ち (+16)	黒 20, 白 4
4×8	294,430,331	2 m 15 s	先手勝ち (+32)	黒 28, 白 0
4×10	1,195,804,922,641	6 d 6 h 22 m	先手勝ち (+40)	黒 39, 白 0
6×6	884,392,099,420	5 d 12 h 16 m	後手勝ち (−4)	黒 16, 白 20

4×8 盤が先手必勝となったことにより、白濱の予想 ($m + n$ の偶奇による判定) は否定された。竹下らは代わりに、 $m \neq n$ または $m = n \wedge m, n \geq 5$ の盤は先手必勝である、という予想を提示している。

■手法 竹下らの実装は、ネガマックス形式のアルファ・ベータ法に、盤上の位置に基づく静的なムーブオーダリング (角優先, 角の隣を後回し) を組み合わせたものである。本稿の観点から重要なのは、以下の 3 点である。

- 置換表を用いていない。論文の「今後の課題」において「局面情報を保存するハッシュテーブルを実装していく」と述べられており、さらに「ハッシュテーブルの実装により、局面の重複及び対称性が考慮できる」と書かれている。すなわち、同一局面の重複探索も、盤面の対称性による重複も除去されていない。
- 対称性の利用は初手のみ。初手の黒は 4 箇所には置けるが、対称性により 1 箇所のみ探索している。
- 必勝読みの効果を「数倍程度」と見積もっている (3.7 節および 7 章)。終局評価を石差ではなく勝敗の 3 値で行う必勝読みは、アルファ・ベータの窓が狭くなるため速いが、その効果は数倍にとどまる、としている。本稿ではこの見積りを実測で検証する (4.3 節)。

■評価値の定義 竹下らは終局の評価を「黒石数 - 白石数」とするが、一方が全滅した場合には勝者に空きマス数を加算する (3.8 節)。このため 4×8 盤の結果は、終局盤面が「黒 28, 白 0」(空き 4) でありながら +32 と表記されている。後述するように、この評価関数の差は最善手順そのものを変えうるため、他の実装と数値を突き合わせる際には注意を要する。

■探索速度 論文は、空きマスの双方向リスト化 (約 2 倍) と石数の差分更新 (約 1.5 倍) により、毎秒 150 万~200 万局面の探索速度を得たと報告している。

2.4 その他の資料

個人による技術記事 [6] では 5×5 , 5×6 盤を対象に、標準的なアルファ・ベータ法に対して 3 倍~12 倍の高速化を得たことが報告されている。ただし絶対的な局面数や計算時間は記載されていない。

2.5 「約 816 億」という数値について

本調査の発端は、Web 検索で「約 816 億」という数値が見られる、という指摘であった。しかし、本調査で参照しえた資料のなかに、この数値と一致するものは見つからなかった。関連する大きな数値は次のとおりである。

表 2 6×6 オセロに関連して現れる大きな数値

数値	意味	出典
約 400 億 (4×10^{10})	Feinstein が探索した局面数	[3] (表題 <i>Forty Billion Nodes</i>), [4]
約 1000 億 (1×10^{11})	検証再実行時の局面数 (5 週間)	Feinstein の報告として流布
8,844 億	6×6 の最終局面数 (置換表なし)	[4] 表 4-1
8,422 億	4×10 リバーシ盤の最終局面数	[4] 表 4-2
1 兆 6287 億	6×6 リバーシ盤の最終局面数	[4] 表 4-2

「816 億」に最も近いのは 8422 億 (4×10 リバーシ盤) および 8844 億 (6×6) であり、桁を読み違えた可能性が考えられる。また、Feinstein の 400 億を別の単位で言い換えた可能性もある。現時点では出典が確認できないため、本稿ではこの数値を採用しない。出典となる URL が特定でき

ば、改めて確認する。

3 数値の読み方：何を数えているのか

文献間の数値を比較する際に最も混乱を招くのは、「局面数」の定義が一致していないことである。少なくとも次の 3 つが区別されなければならない。

最終局面数（葉の数） 終局まで到達したノードの数。竹下ら [4] の「Position」はこれである。

探索節点数（内部ノードを含む） 探索木のノード総数。Feinstein の「40 billion nodes」はこちらと解される。分岐数が平均 b のとき、葉の数のおよそ $b/(b-1)$ 倍になるため、オーダーとしては葉の数に近い。

到達可能な相異なる局面数 重複を除いた状態数。置換表や対称性による削減が効くのはこの差である。

したがって、「Feinstein の 400 億」と「竹下らの 8844 億」が 20 倍以上違うのは、ハードウェアの世代差ではなく**探索アルゴリズムの差**である。Feinstein はムーブオーダリングの精度で稼ぎ、竹下らは置換表を持たない。一方で竹下らの実装は毎秒 150~200 万局面と高速であるため、「重複を除かずに速く回す」戦略で 5 日半に収めている。

4 本研究における実装と実測

4.1 実装方針

先行研究と比較可能な実測値を得るため、 6×6 オセロの完全解析器を新たに実装した (`solve6.c`, 約 220 行の C 言語)。主な設計は以下のとおりである。

- **ビットボード**：盤面を 64 ビット整数 2 本（自分の石・相手の石）で保持する。 6×6 盤を 8×8 ビットボードの内側 ($r, c \in [1, 6]$) に配置することで、外周が常に空となり、反転処理が盤外へ漏れない。着手生成・反転はいずれも 8 方向のシフトとマスク演算で分岐なく求める。
- **置換表**：局面を 64 ビットにハッシュし、評価値の**上限と下限**の組として保存する。これにより、同一局面への合流だけでなく、異なる窓での再訪も再利用できる。置き換え方針は「空きマスが多い（＝部分木が大きい）エントリを優先して残す」とした。
- **PVS (principal variation search)**：第 1 手のみ全窓で探索し、以降は幅 1 の窓 (null window) で試し、窓を超えた場合のみ再探索する。
- **ムーブオーダリング**：静的な位置評価（角優先、角の隣を後回し）に加え、残り 13 マス以上では**着手後の相手の着手可能数が少ない手**を優先する (mobility ordering)。浅い局面では順序付けの計算コストが上回るため適用しない。
- **終盤の切り替え**：残り 10 マス以下では置換表を参照せず、静的順序のみで一気に読み切る。置換表のアクセス遅延のほうが高くつくためである。

測定環境は Apple M4 (10 コア, 16 GB), `clang -O3 -march=native`, 1 コアのみ使用 である。探索速度は 約 3×10^7 局面/秒であり、竹下ら [4] の 150 万~200 万局面/秒に対して約 25 倍にあたる。これはハードウェアの世代差とビットボード化の双方によるものである。

4.2 妥当性の検証

実装の正しさを、公表値のある小さい盤で確認した（表 3）。

表 3 自作ソルバの検証（石差は黒から見た値）

盤	竹下ら [4]	本実装	判定
4 × 4	−8	−8	一致
4 × 6	+16	+16	一致
4 × 8	+32	+29	評価関数の差（下記）

4 × 8 盤の差は誤りではなく、評価関数の定義の違いによる。竹下らは一方が全滅した場合に勝者へ空きマスを加算する（3.8 節）ため、終局盤面「黒 28, 白 0」（空き 4）を +32 と評価する。本実装は素の石差を最大化するため、最善手順そのものが異なり、+29 に到達する。いずれも先手必勝という結論は変わらない。

なお 6 × 6 盤では最善手順の終局が「黒 16, 白 20」と盤面が完全に埋まるため、両者の評価関数は一致する。したがって 6 × 6 の比較は定義の差の影響を受けない。

4.3 必勝読みの効果

竹下ら [4] は、必勝読み（終局評価を勝敗のみに落とす）の枝刈り効果を「数倍程度」と見積もり、したがって現実的な時間での大盤の完全解析は不可能であると結論している。

本実装で 4 × 8 盤について両者を測定した結果を表 4 に示す。

表 4 完全読みと必勝読みの比較（4 × 8 盤、本実装）

	探索局面数	時間
完全読み（窓 [−64, 64]）	216,537,497	4.65 s
必勝読み（窓 [−1, 0]）	3,259,082	0.09 s
比	66.4 倍	

4 × 8 盤では、必勝読みの効果は 66.4 倍に達した。

■ただし、この比は盤に強く依存する。6 × 6 盤について同じ測定を行うと、比は 2.2 倍にとどまった（4.5 節、表 6）。理由は最終石差の値そのものにある。4 × 8 盤の解は +29 と 0 から遠いため、窓 [−1, 0] での探索は「黒が 0 以上を取れる」ことが早々に確定し、直ちに打ち切られる。これに対し 6 × 6 盤の解は −4 と 0 に近く、「黒は 0 以上を取れない」ことの証明は、石差を確定させる作業とほとんど同じだけの手間を要する。

したがって、竹下ら [4] が必勝読みの効果を「数倍程度」と見積もったことは、6 × 6 盤に関する限り妥当である。4 × 8 盤で観測される 66.4 倍を一般則として扱うことはできない。必勝読みが劇的に効くのは、解が引き分けから離れている盤に限られる。

4.4 切り替え点の調整

実装には 2 つの閾値がある。空きマスが S 以下のとき置換表を使わずに読み切る閾値と、空きマスが D 以上のとき動的なムーブオーダーリング（着手後の相手の着手可能数による並べ替え）を行う閾値である。当初はそれぞれ $S = 10$, $D = 13$ と置いていたが、これは「浅い局面では並べ替えの計算コストが枝刈りの利得を上回る」という直観に基づく設定であった。

4 × 8 盤でこの 2 つを掃引した結果を表 5 に示す。直観は誤りであり、**並べ替えは深いところまで効かせたほうが一貫して良い**。

表 5 閾値の掃引 (4 × 8 盤, 完全読み, 探索局面数/時間 [s])

S	D (この空きマス数以上で並べ替え)				
	5	7	9	11	13
4	37.5 M / 1.35	44.6 M / 1.46	57.5 M / 1.76	72.0 M / 2.16	—
6	—	57.2 M / 1.53	75.2 M / 1.90	98.6 M / 2.43	—
8	—	—	92.1 M / 2.13	122.9 M / 2.78	168.2 M / 3.75
10	—	—	—	142.8 M / 3.11	200.0 M / 4.31

最良の組 $(S, D) = (4, 5)$ では 37,475,489 局面・1.35 秒であり、当初設定の 216,537,497 局面・4.65 秒に対して **5.8 倍**の削減となった。なお $S = 0$ (常に置換表を使う) では局面数は 20.6 M までさらに減るが、表アクセスの遅延により所要時間は 1.78 秒と悪化する。局面数と時間の最小点は一致しない。

この結果は、先行研究 [4] が「最も優先すべき課題は探索空間の削減」としつつ、その手段としてパターンに基づく評価関数を挙げていた点に対し、**評価関数を導入せずとも、既存の順序付けを適用する範囲を広げるだけで数倍の削減が得られる**ことを示している。

4.5 6 × 6 盤の実測

6 × 6 盤についての測定結果を表 6 に示す (4.4 節で得た $(S, D) = (4, 5)$, 置換表 2^{26} バケツ = 2 GB)。

表 6 6 × 6 盤の実測 (本実装, 1 コア)

	探索局面数	時間
完全読み (石差を求める)	41,004,075,510	1990.81 秒 (33 分 11 秒)
必勝読み (勝敗のみ)	18,435,230,807	925.09 秒 (15 分 25 秒)

■**完全読み** 石差を確定させる探索の結果は -4 , すなわち**黒 16 対 白 20**で後手 (白) の勝ちであり, Feinstein [3] および竹下ら [4] の結果を厳密に再現した。所要は 41,004,075,510 局面・1990.81 秒 (33 分 11 秒) (1 コア) である。

ここで注目すべきは、この **410 億局面**という値が、Feinstein の報告した約 400 億節点と**ほぼ一**

致していることである。実装も年代もまったく異なるにもかかわらず同程度の値になったことは、アルファ・ベータ法に妥当なムーブオーダリングを与えたときの 6×6 オセロの探索木の大きさが、 4×10^{10} 前後という固有の規模をもつことを示唆する。言い換えれば、Feinstein の 400 億は運の良い一例ではなく、この問題の標準的な難度の目安と考えてよい。

■必勝読み 窓を $[-1, 0]$ に置いて「黒の石差は 0 以上か」だけを判定する探索の結果は「いいえ」、すなわち白（後手）必勝であった。所要は 18,435,230,807 局面・925.09 秒（15 分 25 秒）であり、完全読みの 2.2 分の 1 である。

■先行研究との比較 竹下ら [4] の 8844 億最終局面・5 日 12 時間に対し、本実装の完全読みは局面数で約 1/22、時間で約 240 倍の短縮となった。このうち探索速度の寄与は $20.6/1.75 \approx 12$ 倍であり、残る約 22 倍がアルゴリズム（置換表）による削減である。すなわち、竹下ら自身が「今後の課題」に挙げたハッシュテーブルの導入だけで、5 日半の計算が半日以下になる計算になる。

なお探索速度が 4×8 盤の 29.2 M 局面/秒から 20.6 M 局面/秒へ落ちているのは、置換表が 2 GB とキャッシュに載らない大きさになり、1 局面あたりのメモリ待ちが増えるためである。ただし 4×8 盤で置換表を 32 MB から 2 GB まで変化させても速度低下は 9% にとどまり（31.0 → 28.5 M 局面/秒）、本実装はメモリ帯域に強く律速されていない。この点は後述の Raspberry Pi への移植に有利に働く。

5 先行研究との関係

本研究の位置づけを表 7 にまとめる。

本研究の貢献は次の 3 点である。

1. 置換表の有無が支配的であることを、実測で示した。先行研究の 400 億と 8844 億という 20 倍以上の開きは、計算機の性能差ではなく重複除去の有無に起因する。本実装は置換表を持つことで 410 億に収まり、Feinstein の値とほぼ一致した。竹下ら自身が「最も優先すべき課題は探索空間の削減」と述べており、本研究はその方向に定量的な裏づけを与える。
2. Feinstein の 400 億が偶然の一例ではないことを示した。四半世紀を隔てた独立実装が同程度の値に到達したことは、 6×6 オセロの探索木の大きさが 4×10^{10} 前後という固有の規模をもつことを示唆する。
3. 必勝読みの効きは解の値に依存することを示した。 4×8 盤（解 +29）では 66.4 倍という大きな削減が得られるのに対し、 6×6 盤（解 -4）では 2.2 倍にとどまる。竹下らの「数倍程度」という見積りは 6×6 に関しては妥当であり、 4×8 の値を一般則として外挿してはならない。

6 Raspberry Pi クラスタによる再現の見積り

本調査の出発点となった問いは、「 6×6 の完全解析を 1 時間以内に終えるには Raspberry Pi が何台必要か」であった。以上の実測から、これに答える。

表 7 6 × 6 オセロ完全解析の各実装

	Feinstein (1993) [3]	竹下ら (2015) [4]	本研究
盤面表現	不明	配列 + 空きマスの双方向リスト	ビットボード
枝刈り	アルファ・ベータ + ムーブオーダリング	同左 (静的な位置評価)	アルファ・ベータ + PVS + 動的 mobility ordering
置換表	不明	なし (今後の課題)	あり (上下限を保存)
対称性	不明	初手のみ	初手のみ (今後の課題)
必勝読み	不明	併用 (効果は数倍と報告)	併用 (6 × 6 で 2.2 倍を実測)
探索速度	不明	150~200 万局面/秒	約 3×10^7 局面/秒
6 × 6 局面数	約 400 億 (節点)	8844 億 (葉)	410 億 (完全読み) 184 億 (必勝読み)
6 × 6 時間	約 2 週間	5 日 12 時間 16 分	33 分 11 秒 (1 コア) 15 分 25 秒 (必勝読み)

6.1 Raspberry Pi 1 コアあたりの性能

Raspberry Pi 5 の CPU は Cortex-A76 (2.4 GHz, 4 コア) である。実機を用いずにこの性能を推定するため、測定機 Apple M4 の**高効率コア** (約 2.6 GHz, 命令発行幅が狭く A76 に近い構成) で同一プログラムを実行し、高性能コアとの比を測った (表 8)。

表 8 コア種別による探索速度 (4 × 8 盤, 同一バイナリ, `taskpolicy -b` で高効率コアへ割当)

	探索速度 [M 局面/秒]	比
M4 高性能コア	29.2	1.00
M4 高効率コア	5.40	1/5.4

高効率コアは高性能コアの 1/5.4 であった。6 × 6 盤 (置換表 2 GB) における高性能コアの実測 20.6 M 局面/秒にこの比を適用すると, Raspberry Pi 5 の 1 コアあたり 約 3.8 M 局面/秒と見積もられる。4 コアで置換表を共有する場合, メモリ競合により線形には伸びないためスケーリングを 3.2 倍と見積もると, **1 台あたり 約 12 M 局面/秒**となる。

なお本実装は, 置換表を 32 MB から 2 GB へ 64 倍にしても速度低下が 9% にとどまり (4.5 節), メモリ帯域への依存が小さい。Raspberry Pi 5 のメモリ帯域は測定機より一桁狭いが, この性質により帯域よりコア性能が支配的であると考えられる。

6.2 必要台数

表 9 に、探索方式ごとの必要台数を示す。複数台に分散する場合はルートで部分木を分割し、台間では置換表を共有しない構成を想定する。この場合、重複探索により総局面数が増える。ここでは安全側に 2 倍の増加を見込んだ。

表 9 6×6 の完全解析を 1 時間以内に終えるための Raspberry Pi 5 台数

方式	総局面数	分散時 ($\times 2$)	必要台数
必勝読み (後手必勝の判定のみ)	184 億	368 億	1 台 (26 分)
完全読み (石差 -4 の確定)	410 億	820 億	2 台 (57 分)
完全読み (余裕をみる)	410 億	820 億	4 台 (29 分)
置換表なし (竹下ら方式)	8844 億	—	21 台

結論は次のとおりである。

- **台数はほとんど問題ではない。**置換表を持つ実装であれば、Raspberry Pi 5 (8 GB) が **2 台** あれば完全読みが 1 時間以内に終わる。余裕をみても 4 台で足りる。後手必勝の判定だけでよければ 1 台で 26 分である。
- **支配的なのはアルゴリズムである。**置換表を持たない実装では 8844 億局面を要するため、同じ 1 時間の制約を満たすには 21 台が必要になる。1 台あたり 8 GB のメモリを 2 GB の置換表に充てるだけで、台数が 1 桁変わる。
- **メモリは足りる。**置換表 1 バケツ 32 バイトとして 2^{26} バケツで 2 GB、 2^{27} バケツで 4 GB である。8 GB モデルであれば余裕がある。

6.3 さらに縮める余地

本実装にはまだ未実装の改良があり、1 台での完了も射程に入る。

- **対称性の正準化：** 6×6 盤には 8 通りの対称性がある。ハッシュ前に 8 通りの最小値へ正準化すれば置換表の実効容量が最大 8 倍になる。正準化のコストとの釣り合いを実測する必要がある。
- **偶奇 (parity) を用いた終盤の順序付け：**残り 16 手程度から、空き領域の偶奇に基づく手の並べ替えが強く効くことが知られている。本実装は静的な位置評価しか使っておらず、ここに削減の余地が大きい。
- **MTD(f):** 石差 -4 という解が既知であれば、その近傍の窓から始めることで完全読みを必勝読みに近い手間に落とせる。ただし「再現」ではなく「検証」になる点に注意を要する。
- **並列探索の効率：**本見積りは分散時の重複を 2 倍と仮定したが、YBWC (young brothers wait concept) や置換表の分散共有により、この係数は 1.2~1.5 程度まで下げられる可能性がある。

参考文献

- [1] J. von Neumann and O. Morgenstern: *Theory of Games and Economic Behavior*, Princeton University Press (1944).
- [2] J. Schaeffer, N. Burch, Y. Björnsson, A. Kishimoto, M. Müller, R. Lake, P. Lu and S. Sutphen: “Checkers Is Solved,” *Science*, Vol. 317, pp. 1518–1522 (2007).
- [3] British Othello Federation: *Forty Billion Nodes Under The Tree* — The Newsletter of the British Othello Federation, pp. 6–8 (1993). <http://www.britishothello.org.uk/fbna11.pdf>
- [4] 竹下祐輝, 池田諭, 坂本真人, 伊藤隆夫: 「縮小盤オセロにおける完全解析」, 情報処理学会研究報告 (2015). (英題: *Perfect Analysis in miniature Othello*)
- [5] 白濱拓也: 「縮小版オセロにおける必勝木の作成」, 宮崎大学工学部情報システム工学科 平成 18 年度卒業論文 (2007).
- [6] tosik: 「昔書いたオセロの完全プレイ解析研究のメモ」, note (オンライン記事).