

卒業研究報告書

題目

モンテカルロ法を用いたミニ囲碁

指導教員

石水 隆 講師

報告者

09-1-037-0193

増井 拓視

近畿大学工学部情報学科

平成 24 年 1 月 31 日提出

概要

本研究では囲碁のプログラムを作成した。囲碁とは本来 19×19 の盤面を使い、対戦をすすめていくが、その盤面での計算量は膨大なものになってしまうため、 5×5 と規模を縮小してのプログラム作成を進めた。このミニ囲碁は完全解析がなされており、黒の 25 目勝ちになることが示されている。その後、囲碁の強い AI などを作成されるときにモンテカルロ法と呼ばれる方式があるのを知り、この方式ならどの程度の強さになるのだろうと思い、本研究のテーマを、モンテカルロ法を用いたミニ囲碁、とすることを決定した。

目次

1	序論	1
1.1	本研究の背景	1
1.2	二人零和有限確定完全情報ゲームの完全解析に関する既知の結果	1
1.3	完全解析されていない二人零和有限確定完全情報ゲームに対する手法	1
2	囲碁について	2
2.1	囲碁のルール	2
2.2	コウについて	4
3	研究内容	4
	参考文献	8

1 序論

1.1 本研究の背景

囲碁や将棋やリバーシ等に代表されるボードゲームは、二人零和有限確定完全情報ゲームに分類される。零和とは、ゲーム上プレイしているプレイヤーの利得の合計が常に0になることである。有限とは、そのゲームにおける各プレイヤーの可能な手の組み合わせの総数が有限であることである。確定とは、プレイヤーの着手以外に偶然がゲームに影響を与えるといった事が無いという意味である。完全情報ゲームとは、各プレイヤーが自分の手番において、各プレイヤーがこれまでに行った選択、または意思決定についての全ての情報を完全にすることができるゲームである。二人零和有限完全情報ゲームは、その性質上解析を行いやすいため、ゲーム理論においてさまざまな研究がなされてきた。

1.2 二人零和有限確定完全情報ゲームの完全解析に関する既知の結果

二人零和有限確定完全情報ゲームは双方最善手を指した場合、先手勝ち、後手勝ち、引き分けのどれになるかはゲーム開始時点で決定しており、理論上、すべての可能な局面を解析することができれば最善の手を打つことができる。しかし多くのボードゲームでは、可能な局面数がきわめて大きいため、完全解析を行うことは不可能である。例を挙げれば、可能な局面数はリバーシが 10^{28} 通り、チェスが 10^{50} 通り、将棋が 10^{69} 通り、囲碁が 10^{170} 通り程度であるとされており、現在の計算機の性能を超えている。一方、可能な局面数が少ないゲームでは完全解析されているものもある。連珠は双方最善手を打った場合、47 手で先手が勝つ[3]。チェッカーは双方最善手を指すと引き分けとなる[4]。その他にもバンダイが発売したシンペイと呼ばれるゲームは価値に要する最長手数が 49 手で、後攻が勝つ[5]とわかっているなど多数のゲームの完全解析が行われている。

局面数が大きいゲームについては、ゲーム盤をより小さいサイズに限定した場合の解析も行われている。サイズ 6×6 のリバーシでは、双方最善手を打つと 16 対 20 で後手勝ちとなる[6]。将棋では、盤サイズ 3×4 に減らし、駒の種類を 4 つに減らしたどうぶつしょうぎ[7]が完全解析されており、双方最善手を指すと 78 手で後手が勝つことが判明している[8]。

囲碁の場合は、サイズ 4×4 では双方最善手を打つと引き分けになり[9]、 5×5 の囲碁は黒の 25 目勝ちとなる[10]。しかしサイズ 6×6 以上については、現在のところまだ解析されていない。図 1, 図 2 にそれぞれサイズ 4×4 , 5×5 のミニ囲碁で双方の最善手を示す。

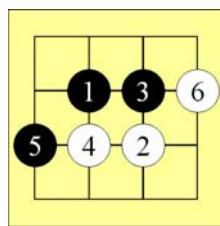


図1 4×4 路盤の最善手[9]

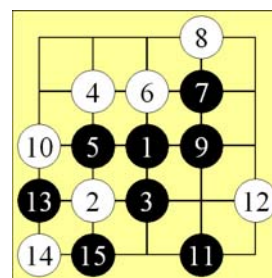


図5 5×5 路盤の最善手[10]

1.3 完全解析されていない二人零和有限確定完全情報ゲームに対する手法

可能な局面数が多いゲームに対して完全解析を行うことは困難である。そのようなゲームに対しては確実な最適手を得ることはできないが、局面の評価値計算、定石データベース、一定手数の先読み、終盤での必勝読みと完全読み、モンテカルロ法[2]などを用いてより有利だと思われる手を選択することができる。囲碁は盤面が 19×19 と大きく局面数が多いため解析は難しい。また、例えば将棋では各駒が持つ能力に合わせた価値を付加し、それを元に局面の評価値を求めることができるが、囲碁では石同士に差は無く、そのような手法を採れない。囲碁では、プロ棋譜から配置パターンの辞書を作り、スコア付けし、学習を行い、着手する等の戦術がある。現在ではモンテカルロ法を用いた AI が

勢いを増している[17][18][19][21][26]。モンテカルロ法とは、終局までシミュレーションし勝率の最も高い手を選択する方法である。

以上の手法を用いることにより、完全解析を行わなくてもある程度の強さのプログラムを作ることが可能である。

現在強いとされているソフトには「天頂の囲碁」というものがある。このソフトに用いられている Zen という思考エンジンが 2012 年 3 月に九段との対局で 4 子、5 子ともに圧勝という結果を出した。その強さはアマチュア高段者レベルといわれている。

そしてその Zen が、今年の 2 月 11 日と 16 日に開催される第一回囲碁電王戦にて対局することとなっている。1 日目には張豊猷八段と平田智也三段と対局を行う。9 路盤の勝負ならコンピューターが勝っても不思議ではない、というレベルにはなっているとされているコンピュータ囲碁が、この対局でどのような結果を残すのかとても興味深いものである。

1.4 本研究の目的

本研究のテーマである 5×5 の囲碁は 1.2 節で述べた通り既に完全解析が行われており、黒の 25 目勝ちとなることがわかっている。本研究ではデータベース参照等を行わず、モンテカルロ法をのみを用いてどの程度の強さの AI を制作できるのか、という目的のもと行う。

1.5 本報告書の構成

本報告書の構成は以下の通りである。

まず第二節章では、まず囲碁について説明し、作成したクラスについての説明をする。そして第三章は本研究の結果・結論を述べる。

2 囲碁について

本章では囲碁について説明する。

2.1 囲碁のルール

囲碁の基本的なルールは下記の 7 つである。

1. 石は交点に置く。
2. 一度置いた石は動かさない。
3. 白と黒交互に打っていく。
4. 地を多く取った方が勝ち。
5. 相手の石は囲めば取れる。
6. 呼吸点が無くなる所へは打てない。
7. 一手前と全く同じ状況になるところへは打てない。

2.1.1 地とは

地とは、図 1 のように石で囲まれている場所である。囲碁は最終的にこの地の多少で勝敗を判定する



図1:地について

2.1.2 呼吸点について

呼吸点とは、石が連続して繋がることのできる場所の事を言う。図1の場合 a に打つことにより黒は連続して繋がることのできる。そしてこのように呼吸点が一か所しかない場合をアタリと呼ぶ。次の手で a に白を置かれた場合、図2のように取られてしまう。そして図2のような場合 a に黒を打てばその時点で呼吸点が0となるため打つことができない。

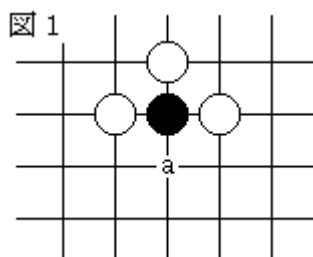


図1:黒の呼吸点

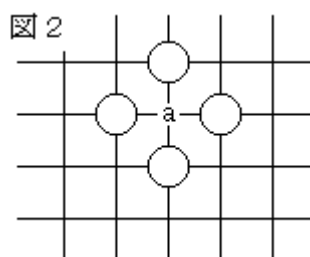


図2:呼吸点が無くなる場合

2.2. 石の生死

囲碁には生きている石と死んでいる石と呼ばれるものがある。図1の場合 A と B に黒が置ければ白を取れるという状況だが A も B も黒を置けば、呼吸点が無くなり自殺手となるのでルール上置くことができないので、白は絶対取られることが無くなる。つまり図1の白石は生きている石ということである。逆に図2のような場面の場合、黒が A に置いたとして次に白が B に置くとする。その場合 A にある黒石は取れるが次の手で A に黒を置けば白石を全て取ることができ。つまり図2の白石は死んだ石ということである。

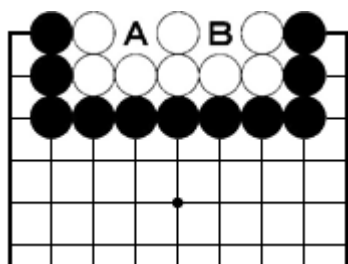


図1:生きている石

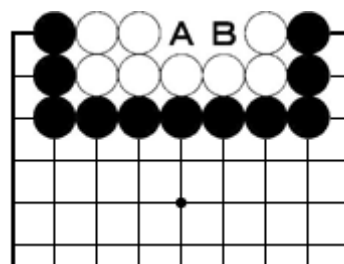


図2:死んでいる石

2.2 コウについて

上記の7番で述べたように、打つと一手前と全く同じような状況、これを「コウ」と呼ぶ。図1に示す。図1において黒1の後に、白2と打って黒1を取り、その後黒1と打って白2を取れば最初の形と同形となる。このように、黒と白の石の取り合いが無限にできてしまう状況、これが「コウ」である。この場合取られた側は一度違う場所に打たなければならない、これを「コウダテ」と呼ぶ。一度違う場所に打てば次はそこに打つことが可能であるが、もし黒が取った場合次の白の手までに黒がツイてしまうことでコウを終了させることができる。

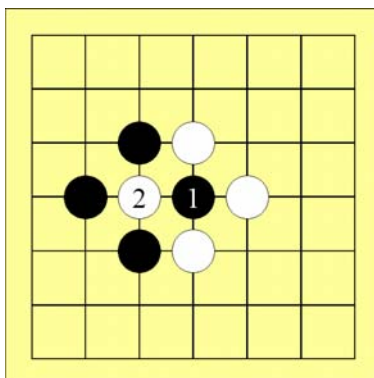


図1: コウの配置

2.3 セキについて

図1のように、aに黒か白、どちらを打ってもアタリにはなるが次の手で取られてしまう、という状況をセキと言う。この場合、図の×の場所はお互いに判定されない。

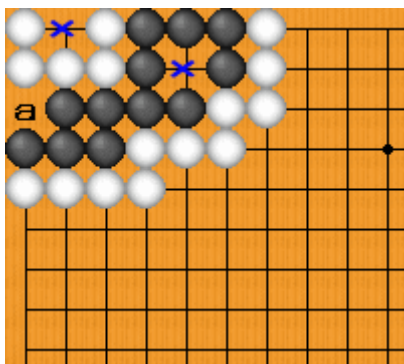


図1: セキの配置

3 研究内容

本章では研究で作成した5×5囲碁のプログラムについて説明する。

3.1 囲碁プログラムで用いた手法

1章で述べたように、次に打つべき手をどのように選択するかは様々な手法がある。本節では作成した5×5囲碁プログラムについて説明する。AIとしては現在強いAI制作に有効といわれているモンテカルロ法[2]を用いる。

3.1.1 着手の判定

5×5囲碁プログラムではモンテカルロ法を用い、一手毎に終局まで対局ということを何度も繰り返し

その中から一番勝率の高かった手を選択する。呼吸点の無い場所には着手できないようにし、一手前と全く同じ配置にならないかどうかの判定をし、石を置く。着手可能な手を打った場合、相手側の石において呼吸点が無くなっていた場合その石を取り除く判定を出す。

3.1.2 勝敗の判定

囲碁の勝敗の判定は投了か、白と黒の地の多さで勝敗を決定する。本研究で制作したプログラムには投了の判定を出す機能を搭載していないので勝敗がつくまで対局という形になる。黒と白のいずれでもない地が無くなった際に、地の多さによって勝ちを決める。

3.1.3 コウの判定

コウの判定については、新しくコウの判定用の配列を作り、前の盤面と同じにならないよう記憶し判定する。

3.2 ミニ囲碁プログラム

本節では作成したプログラムについて述べる。本研究では、Go クラスと Delete クラスを作成し、Go クラスでは囲碁の基本動作、Delete クラスでは石を取り除く動きをするよう作成した。

3.2.1 Go クラス

このクラスでは囲碁の基本動作の構築を行っている。モンテカルロ法では、石を打つ場所はランダムに決定されるので、乱数を用いて動作を決める。

- `static int intGo[][]`
7×7 の int 型の配列。5×5 だと石を消す判定を出すときに困るので外枠に-1 を入れる、中に 0~24 の数字を入れ、乱数にて参照するときを使う。
- `static int igo[][]`
intGo[][] 同様の理由により 7×7 の配列。中にはすべて 0 を入れ、黒石が置いてある場所に 1、白石が置いてある場所に 2、黒石が置けない場所に 3、白石が置けない場所に 4 と入れていく。
- `void sansyo() {}`
盤面を読み込み igo[][] に対応する数字を入れていく。
- `boolean put() {}`
石を置くメソッド。乱数により intGo[][] に対応した場所に石を置けるかどうかの判定を出す。置けるなら true、無理ならば false を返す。
- `void rand() {}`
乱数を生成するメソッド。do while を使いループさせる。乱数を生成し、put() にて true が返ってきた場合ループを脱出する。
- `Boolean breakLoop() {}`
igo[][] を参照し、0 が存在しなくなったら動作を止めるようする。

3.2 Delete クラス

このクラスでは石を取り除く動作を行う。盤面を読み込み、石が別色に囲まれているようならば石を取り除く動作を行う。石の四方を参照し、連なりを見る。途中で空白が存在する場合動作を停止。その過程でその石が別色に囲まれていた場合に石を取り除く。ここでは自殺手となるような打てない場所の判定も出す。

4 結果

本研究で制作したプログラムがどの程度の強さかを検証するために、ランダム配置をする CPU との対

戦を 100 回行った。その結果 83 勝 17 敗という結果になった。ランダム配置との対戦とはいえ高確率で勝てる結果となった。

もちろんランダム配置なので、この結果は偶然の物かもしれないが、劇的に変化することは無いと思われる。

5 結論および今後の課題

本研究では、 5×5 の囲碁プログラムを作成した。そしてランダム配置の CPU との対戦でそこその成績をあげることができた。しかし、囲碁に対して必要最低限の知識しか持たないまま研究に着手したため至らない部分も多々あったと思われる。当時は、単純な動作を繰り返すだけという認識だったが、研究を進め、囲碁を知るにつれて、非常に奥が深く、興味深いものであると思った。今後の課題としては、単純なランダム配置からの手の選択だけでなく、それに加え評価値やデータベース参照といった物を取り入れる等が挙げられる。今回は 5×5 の盤面で行ったが、本来の 19×19 の盤面において、モンテカルロ法を用い、評価関数やデータベース参照と組み合わせることにより強い AI を作り出すことが今後の課題と言えるだろう。

謝辞

2年もの間、時に厳しく、時にやさしく根気よく指導していただき感謝の気持ちでいっぱいです。いろいろと興味深い物も教えていただき、研究ながらも楽しんでできたのはひとえに石水隆先生のおかげだと思います。本当にありがとうございました。

参考文献

- [1] 清慎一, 佐々木宣介, 山下宏, コンピュータ囲碁の入門, 共立出版(2005)
- [2] 美添一樹, 山下宏, 松原仁, コンピュータ囲碁—モンテカルロ法の理論と実践—, 共立出版, (2012).
- [3] Janos Wagner and Istvan Virag, Solving renju, ICGA Journal, Vol.24, No.1, pp.30-35 (2001), http://www.sze.hu/~gtakacs/download/wagnervirag_2001.pdf
- [4] Jonathan Schaeffer, Neil Burch, Yngvi Bjorsson, Akihiro Kishimoto, Martin Muller, Robert Lake, Paul Lu, and Steve Suphen, Checkers is solved, Science Vol.317, No.5844, pp.1518-1522 (2007). <http://www.sciencemag.org/content/317/5844/1518.full.pdf>
- [5] 田中哲朗, ボードゲーム「シンペイ」の完全解析, 情報処理学会論文誌, 48(11), pp.3470-3476, 情報処理学会, (2007), <http://id.nii.ac.jp/1001/00009789/>
- [6] Joel Feinstein, Amenor Wins World 6x6 Championships!, Forty billion noted under the tree (July 1993), pp.6-8, British Othello Federation's newsletter., (1993), <http://www.britishothello.org.uk/fbnall.pdf>
- [7] 北尾まどか, 藤田麻衣子, どうぶつしょうぎねっと, (2010), <http://dobutsushogi.net/>
- [8] 田中哲郎, 「どうぶつしょうぎ」の完全解析, 研究報告ゲーム情報学(GI), Vol.2009-GI-22 No.3, pp.1-8, 情報処理学会, (2009), <http://id.nii.ac.jp/1001/00062415/>
- [9] 清慎一, 川嶋俊, 探索プログラムによる四路盤囲碁の解, 研究報告ゲーム情報学(GI), Vol. 2000-GI-004, pp.69-76, 情報処理学会, (2000), <http://id.nii.ac.jp/1001/00058633/>
- [10] Eric C.D. van der Welf, H.Jaap van den Herik, and Jos W.H.M.Uiterwijk, Solving Go on Small Boards, ICGA Journal, Vol.26, No.2, pp.92-107 (2003).
- [11] 高橋克吉, 猪爪歩, 伊藤毅志, 村松正和, 松原仁, 次の一手課題に基づく囲碁と将棋の特徴比較ゲームプログラミングワークショップ 2012 論文集, Vol.2012, No.6, pp.1-8, 情報処理学会, (2012), <http://id.nii.ac.jp/1001/00091320/>
- [12] 伊藤毅志, 高橋克吉, 猪爪歩, 加藤英樹, 村松正和, 松原仁, 人間とコンピュータの思考の違い～囲碁の次の一手問題による考察～, ゲームプログラミングワークショップ 2012 論文集, Vol.2012, No.6, pp.9-16, 情報処理学会, (2012), <http://id.nii.ac.jp/1001/00091321/>
- [13] 本田拓朗, Viennot Simon, 池田心, 囲碁における大局観を実現する広域パターンマッチング, ゲームプログラミングワークショップ 2012 論文集, Vol.2012, No.6, pp.17-21, 情報処理学会, (2012), <http://id.nii.ac.jp/1001/00091322/>
- [14] 大島真, 山田孝治, 遠藤聡志, 囲碁におけるポテンシャル領域探索の適用, ゲームプログラミングワークショップ 2011 論文集, Vol.2011, No.6, pp.84-87, 情報処理学会, (2011), <http://id.nii.ac.jp/1001/00078258/>
- [15] 佐藤真史, 穴田浩一, 堤正義, 囲碁の数理モデル化とその応用, ゲームプログラミングワークショップ 2011 論文集, Vol.2011, No.6, pp.100-103, 情報処理学会, (2011), <http://id.nii.ac.jp/1001/00078262/>
- [16] 中西惇, 中村貞吾, 局面評価とパターンによる着手予測を用いた囲碁の好手の判別, ゲームプログラミングワークショップ 2011 論文集, Vol.2011, No.6, pp.108-111, 情報処理学会, (2011), <http://id.nii.ac.jp/1001/00078264/>
- [17] GPGPU によるモンテカルロ碁のシミュレーションの並列処理岩川夏季, 成見哲, 村松正和, 研究報告ゲーム情報学(GI), Vol.2011-GI-26, No.10, pp.1-6, 情報処理学会, (2011), <http://id.nii.ac.jp/1001/00074629/>
- [18] 荒井光, 福永修一, ガウス過程を用いたモンテカルロ碁における戦略獲得研究報告ゲーム情報学(GI), Vol.2011-GI-25, No.2, pp.1-5, 情報処理学会, (2011), <http://id.nii.ac.jp/1001/00073006/>
- [19] 豊田琢磨, 松本祐輔, 佐々木健太, 小谷善行, 囲碁におけるシミュレーション結果の継承を用いたモンテカルロ法の改良研究報告ゲーム情報学(GI), Vol.2010-GI-23, No.2, pp.1-7, 情報処理学会, (2010), <http://id.nii.ac.jp/1001/00068062/>
- [20] 松本祐輔, 小谷善行, 溢れ碁ルールの提案とそれを用いた囲碁の小路探索, 研究報告ゲーム情報学(GI), Vol. 2010-GI-23, No.1, pp.1-8, 情報処理学会, (2010), <http://id.nii.ac.jp/1001/00068061/>

- [21] 門脇聡広, 村松正和, モンテカルロ基における統計的手法による特徴の学習, ゲームプログラミングワークショップ 2009 論文集, Vol.2009, No,12, pp.71-74, 情報処理学会, (2009), <http://id.nii.ac.jp/1001/00097702/>
- [22] 瀧澤武信, 数理ゲーム理論の囲碁の最終盤への応用ゲームプログラミングワークショップ 2009 論文集, Vol.2009, No,12, pp.119-126, 情報処理学会, (2009), <http://id.nii.ac.jp/1001/00097713/>
- [23] 矢島敏雄, 石の働きと盤の効果, 研究報告ゲーム情報学(GI), Vol.2009-GI-21, pp.41-46, 情報処理学会, (2009), <http://id.nii.ac.jp/1001/00061658/>
- [24] 松井利樹, 野口陽来, 土井佑紀, 橋本剛, 囲碁における勾配法を用いた確率関数の学習, 研究報告ゲーム情報学(GI), Vol.2009-GI-21, pp.33-40, 情報処理学会, (2009), <http://id.nii.ac.jp/1001/00061657/>
- [25] 鎌田真人, プロプロ9路盤囲碁の序盤の変化(その一), 研究報告ゲーム情報学(GI), Vol. 2009-GI-21, pp.25-32, 情報処理学会, (2009), <http://id.nii.ac.jp/1001/00061656/>
- [26] 野口陽来, 松井利樹, 橋本隼一, 橋本剛, モンテカルロ基で用いるパターンの大きさに関する考察, 研究報告ゲーム情報学(GI), Vol.2008-GI-020, pp.31-35, 情報処理学会, (2008), <http://id.nii.ac.jp/1001/00058475/>
- [27] 平本直之, 成瀬正, 囲碁の中盤における盤面の評価関数に関する検討, 研究報告ゲーム情報学(GI), Vol.2007-GI-018, pp.23-30, 情報処理学会, (2007), <http://id.nii.ac.jp/1001/00058491/>
- [28] 鎌田真人, 松坂昇子, 松原仁, 基力認定問題によるコンピュータ囲碁の評価(その 3), 研究報告ゲーム情報学(GI), Vol. 2006-GI-015, pp.1-8, 情報処理学会, (2006), <http://id.nii.ac.jp/1001/00058516/>
- [29] 鎌田真人, 松坂昇子, 松原仁, 基力認定問題によるコンピュータ囲碁の評価(その 2), 研究報告ゲーム情報学(GI), Vol. 2004-GI-013, pp.35-42, 情報処理学会, (2005), <http://id.nii.ac.jp/1001/00058540/>
- [30] 高濱昌孝, 大沢英一, 囲碁における注意決定と分節化について, 研究報告ゲーム情報学(GI), Vol. 2004-GI-013, pp.43-48, 情報処理学会, (2005), <http://id.nii.ac.jp/1001/00058541/>
- [31] 美添一樹, 今井浩, 囲碁の部分問題における両利きの探索研究報告ゲーム情報学(GI), Vol.2005-GI-014, pp.63-70, 情報処理学会, (2005), <http://id.nii.ac.jp/1001/00058533/>
- [32] 福井真人, 竹内義則, 松本哲也, 工藤博章, 山村毅, 大西昇, 囲碁盤面の評価方法研究報告ゲーム情報学(GI), Vol.2003-GI-011, pp.21-26, 情報処理学会, (2004), <http://id.nii.ac.jp/1001/00058555/>
- [33] 二宮勘輔, 囲碁ヨセ計算における近似法, 研究報告ゲーム情報学(GI), Vol. 2004-GI-012, pp.55-62, 情報処理学会, (2004), <http://id.nii.ac.jp/1001/00058549/>
- [34] 『第1回囲碁電王戦』開催決定, 棋士、アマチュアとコンピュータ「Zen」が対戦, 公益財団法人日本棋院, 2014年1月31日, http://www.nihonkiin.or.jp/news/2014/01/1_21116.html
- [35] 山田ユウキ, 「囲碁電王戦」2月開催決定、プロ棋士と小沢一郎氏が最強ソフト「Zen」に挑戦, マイナビニュース 2014年1月31日, 株式会社マイナビ, <http://news.mynavi.jp/articles/2014/01/31/denou/>
- [36] 最強はどちらだ・・・店長の囲碁 VS 銀星囲碁 VS 最強の囲碁、
<http://www.computer-igo.com/category2/entry12.html>
- [37] ハウコレ、写真付きで解説！「地」ってなに？ 囲碁の対局の流れ（前半の打ち方）、
<http://howcollect.jp/article/881>
- [38] とりあえず遊ぶための囲碁ルール（基礎）、
<http://www.geocities.co.jp/Playtown-King/9806/igo/igo01.html>
- [39] 石の生死を見分けよう - 棋聖戦：囲碁将棋：YOMIURI ONLINE（読売新聞）、
http://kisei.yomiuri.co.jp/column/shinan_igoamigo/02.htm