

モンテカルロ碁におけるポテンシャルモデルを利用した枝刈りの可能性 A Probability of a Potential Model Pruning in Monte Carlo Go

大島 真十
Makoto Oshima

山田 孝治 †
Koji Yamada

遠藤 聡志 ‡
Satoshi Endo

モンテカルロ碁は知識表現を用いずに棋力を成立させるコンピュータ囲碁である。精度を保つには膨大な計算量を必要とするが、ゲーム木に対して適切な枝刈りを行うことで効率化が可能である。本実験では既存のゲーム知識、特に置石が周囲に与える影響を表したポテンシャルモデルを枝刈りに利用することで計算量の削減を行った。枝刈り方法は設定の異なる4種用意し、其々の削減の効果・傾向を測った。最も効果の高い枝刈り方法では18%、また観測された特性を考慮し、2種の枝刈り方法を組み合わせることで23%まで計算量が削減可能となった。但し限定された環境下での結果の為、実戦の観点に立ち、更に調整を加え試行を重ねる必要がある。

Monte Carlo go is the computer go which satisfy the strength without the knowledge expressions of *igo*. Monte Carlo go needs an enormous computational complexity to keep the precision. Though, reductions of the computational complexity are possible by proper pruning for the *igo* game tree. In this study, we tackled the reduction of the computational complexity by the pruning for the *igo* game tree using the potential model which was the knowledge expression of *igo*. In this experiment, 4 kind of pruning were tried and measured. The best one pruning reached an 18% reduction of the computational complexity and the proper combination of two pruning reached a 23% reduction of the computational complexity. We only showed a probability of the potential model pruning this time. Thus we need to tackle many trials in different environments after this.

1. 序言

モンテカルロ碁(Brüggmann 1993)において、ポテンシャルモデルを利用した枝刈りの効果について実験を行った。

最近、モンテカルロ碁はコンピュータ囲碁分野で非常に重要になってきている。モンテカルロ碁とはモンテカルロ法を囲碁のゲーム木に適用させたコンピュータ囲碁のことである。ゲーム知識を用いずに高い棋力を実現できる方法として注目を浴びている。しかしモンテカルロ碁(モンテカルロ探索)の精度を確保するには膨大な計算を必要とする為、囲碁の広大なゲーム木における枝刈り方法の開発が期待されている。

以前のコンピュータ囲碁研究は、囲碁の知識を巧妙に計算機へ実装し、より適切な評価関数を作成することに重点が置かれていた。しかし、モンテカルロ碁の優位性が確認されて以来、従来の研究内容に替わり、モンテカルロ碁に基づいたコンピュータ囲碁の研究が主流となってきている。囲碁は二人零和有限確定完全情報ゲームの中でも特に評価関数の作成が難しく、それがコンピュータ囲碁の棋力向上を阻む一因となっている。それに対してモンテカルロ碁は判断の際に評価関数を必要としない。それ故、囲碁においてはその特徴が有効に機能し、モンテカルロ碁の高い棋力を成立させている。近年のコンピュータ囲碁の大会では、モンテカルロ碁から発展したコンピュータ囲碁(モンテカルロ探索木を用いたコンピュータ囲碁)(Coulom 2007)(Gelly, et al. 2006)が上位を占め、その高い実力を示した。そして、現在のコンピュータ囲碁研究はモンテカルロ碁の効率化・精度向上に集約されつつある。

モンテカルロ碁の棋力は計算量に大きく依存する。その

為モンテカルロ碁のゲーム木における探索効率化は非常に重要である。現在では、より高い精度が得られるゲーム木の構築方法(Kocsis and Szepesvári 2006)や、有限である計算量の効率的な投入方法(Kitagawa, Kurita and Chikayama 2009)など、確率に基づいた研究が多く為されている。それらに対してここでは、従来のゲーム知識、特にポテンシャルモデルを用いたゲーム木の枝刈り方法について実験をした。

モンテカルロ碁においてゲーム知識を用いた枝刈り方法には、未だ決定的なものではなく、様々な知識において検討の余地が多く残されている。ポテンシャルモデルもその一つであり、モンテカルロ碁においてポテンシャルモデルを用いた場合の振る舞いや棋力の変化は未だ明らかでない。

本研究の目的は、ポテンシャルモデルを用いた枝刈りによる、モンテカルロ碁の計算量の削減である。

本実験では、ポテンシャルモデルによる枝刈り機能を備えたモンテカルロ碁を用意し、設定条件下においてノーマルなモンテカルロ碁との対戦を軸に、棋力を落とさずに削減できる計算量を求めた。

ポテンシャルモデルとは基盤上の石をポテンシャルと見なすことであり、それによって盤上の各合法手にポテンシャル値およびポテンシャル勾配値が分布する。本実験での枝刈りとは、それらの値に対して一定の基準を定め、それに従い合法手の選別を行うことである。

ポテンシャルモデルによる枝刈り機能はまず4種用意した。後にその枝刈り機能の中から特徴的な傾向を示した2種の方法を意図して組み合わせた枝刈り機能も用意した。更にランダムな枝刈りを行う方法も用意し、提案手法との比較要素とした。また、ポテンシャルモデルによって有効とされた合法手群と、モンテカルロ探索によって有効とされた合法手群との一致率、及びポテンシャルモデルによって有効とされた合法手群における、モンテカルロ探索によ

† 琉球大学大学院理工学研究科 総合知能工学専攻

‡ 琉球大学工学部情報工学科

る最善手の含有率を測り、モンテカルロ探索と各々の枝刈りとの適合性を測った。

実験の結果、4種の枝刈り方法は、各々異なる特性を見せた。その中でもポテンシャル勾配値の高い合法手のみ次手の候補とする枝刈り方法が最も計算量を削減した。またこの枝刈り機能と、ポテンシャル値の絶対値が低い合法手のみ次手の候補とする枝刈り方法を組み合わせることで、枝刈りの更なる改善が行えた。

結論として、ポテンシャルモデルは計算量の削減に関して一定の効果を持ち、その特性は様々に変化することを確認した。また異なる枝刈り方法を組み合わせることで更に効果を伸ばす事も確認した。未だ実戦には及ばないものと判断するが、モンテカルロ基におけるポテンシャルモデルを用いた枝刈りの可能性を示した。今後、ポテンシャルモデルの設定や枝刈りのタイミングなどの調整を行うことで更なる計算量の削減効果を調査する。

2. 方法

囲碁では、置石の配置が次手の評価に影響を与える。モンテカルロ探索において、その現象を捉える事が出来る。

Fig. 1は、置石により変化したモンテカルロ探索による最善手（黒石）の分布（盤上に石がないときの最善手の分布と、盤上に石を置いたときの最善手の分布との差分、サンプル数1万）を表す。ポテンシャルによる影響を強調する為に複数の石を固めた。(a, b)では、黒石群周辺の合法手において数値が上昇し、離れた場所では減少している様に見て取れる。白石群とした(c, d)でも、(a, b)と近い現象が見て取れる。(e, f)では黒石群と白石群の狭間にある合法手の値が上昇している様に見て取れる。これらは単純な数パターンを試行だが、その変化量は置石位置との距離に相関性があることを示唆する。この変化をポテンシャルモデルによって定式化できれば、膨大な計算量によって成立するモンテカルロ探索を、計算量の少ないポテンシャルモデルによって代替し、一局に必要な計算量を削減できる。

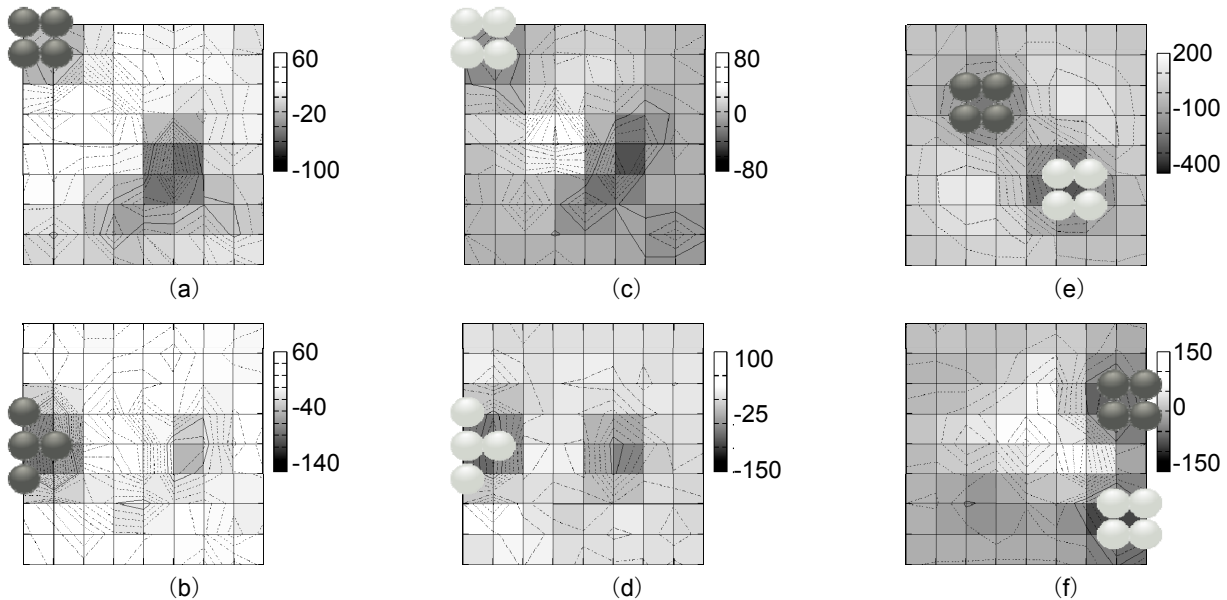


Fig. 1: Distributions of the Best Move by Monte Carlo Search

2.1. モンテカルロ基

モンテカルロ基とは、モンテカルロ法に基づいたゲームシミュレーションによって現局面の評価を行う（モンテカルロ探索）コンピュータ囲碁である。

2.1.1. モンテカルロ探索

現局面から終局まで両者がランダムに合法手を打ち合う。このシミュレーション（プレイアウト）を繰り返すことで、現局面から其々の合法手を選択した場合の勝率（評価）が得られる。合法手*i*のプレイアウト回数を s_i 、プレイアウトによって得られた報酬の和を X_i とする。報酬とはプレイアウトの結果で、勝てば1、負ければ0である。このとき勝率 \bar{X}_i を式(1)に示す。

$$\bar{X}_i = \frac{X_i}{s_i} \quad (1)$$

この手法は評価関数を用いていない為、知識表現の難しいゲームにおいて有効とされている。

モンテカルロ探索ではランダムシミュレーションの結果により、それぞれの合法手の評価値が決まる。その為、結果の信頼性を高めるには多くのシミュレーション回数を必要とする。しかし実際に投入できる計算量は有限である為、探索効率を改善する事によって精度を高める方法が求められている。

2.2. ポテンシャルモデル

置石は周囲の交点に対して地になる確率を上昇させる。ポテンシャルモデルとはその確率の変化を定式化したものである。これまでもポテンシャルモデルは現盤面の評価方法として幾つか研究が成されている(Nakamura and Kitoma 2002),(Yajima 2009),(Tajima and Sanechika n.d.)。モンテカルロ探索と比べ人間の盤面認識に近く、また囲碁ゲームの知識表現の一つといえる。置石を中心とし

で最も高いポテンシャルを示し、中心からの距離に伴ってポテンシャルが減衰していくという概要は共通しているが、極値や減衰比率など、研究毎にその設定は様々である。

2.2.1. ポテンシャル分布

本実験におけるポテンシャル値 P の設定を式(2, 3, 4)に示す。黒石を正、白石を負のポテンシャルとすることで、盤上にはその石の配列独自のポテンシャル分布が発生する。

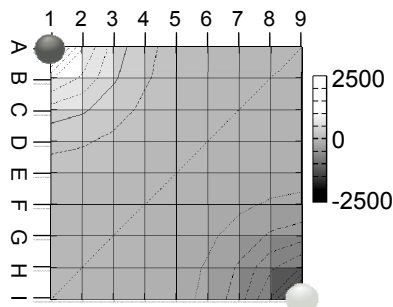
盤上に現れるポテンシャル分布の例を Fig. 2 に示す。

$$P(X, Y) = \sum_{k=1}^n P_k(X, Y) \quad (2)$$

$$P_i(X, Y) = \frac{Stone_i}{2^d} \quad (3)$$

$$d = \sqrt{(X - x_i)^2 + (Y - y_i)^2} \quad (4)$$

- n : 盤上に置かれた石の総数
- $P(X, Y)$: 交点 (X, Y) のポテンシャル値
- $P_k(X, Y)$: 石 k から交点 (X, Y) へ掛るポテンシャル値
- $Stone_i$: 石 i のポテンシャル値 (極値)
- d : ユークリッド距離
- x_i, y_i : 石 i の交点位置



Board Size : 9
Stone(Potential Extremum) : ±2048
Fig. 2: A Potential Distribution

2.2.2. ポテンシャル勾配の設定

ポテンシャル勾配値 PG の設定を式(5)及び Fig. 3 に示す。また、ポテンシャル勾配値の分布の例を Fig. 4 に示す。

$$PG(i, j) = |P(i-1, j) - P(i+1, j)| + |P(i, j-1) - P(i, j+1)| \quad (5)$$

$PG(i, j)$: 交点 (i, j) におけるポテンシャル勾配値

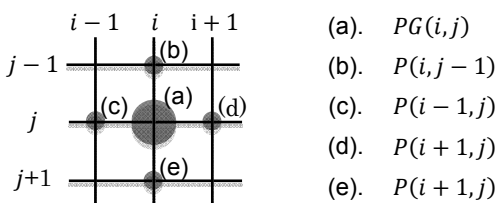
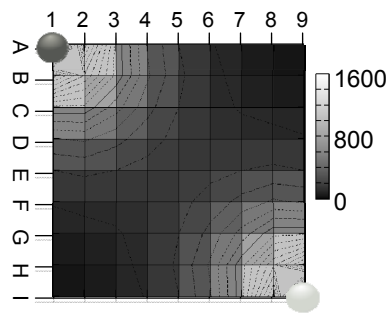


Fig. 3: A Potential Gradient



Board Size : 9
Stone(Potential Extremum) : ±2048
Fig. 4: A Potential Gradient Distribution

2.2.3. ポテンシャルフィルタの機能

ポテンシャルフィルタとは、本実験におけるゲーム木の枝刈り器である。このポテンシャルフィルタが各局面において合法手を選別する。この選別が、本実験の提案する枝刈りである。

ポテンシャルフィルタは合法手が持つポテンシャル値又はポテンシャル勾配値に対して一定の基準を持ち、それに従い合法手の選別を行う。ポテンシャルフィルタは種類毎に独自の選別条件を持つ。

ポテンシャル値又はポテンシャル勾配値が条件を満たさない場合、その合法手は次手の候補から除外される。そして条件を満たした合法手群のうちでモンテカルロ探索を行い、次手を算出する。

ポテンシャルフィルタによって除外した合法手の数に従い、モンテカルロ探索に要する計算量が削減できる。

2.2.4. ポテンシャルフィルタの設定

本実験で作成したポテンシャルフィルタは Table. 1 に示した 4 種類である。ポテンシャル値又はポテンシャル勾配値の降順に従い合法手をランキングし、その順位をもとに選別を行う。Fig. 5 は、各ポテンシャルフィルタのイメージである。ランキングの対象とした値の小・大を濃・淡で表す。数字はその交点の順位を表す。編み掛け部分は、フィルタによって次手の候補から除外された交点を示す。

また無作為に選別を行うフィルタも用意し、ポテンシャルモデルを用いたフィルタとの比較要素とした。

いずれの枝刈りも、次手の候補の数を常に半分にする様設定した。つまりフィルタを使用した場合、その局面において次手の決定に要する計算量は半分になる。

Table. 1: Types of Pruning Filters

	対象値	除外対象
Filter1	ポテンシャル	下位半分
Filter2	ポテンシャル	上位半分
Filter3	ポテンシャル	1/4 以上と 3/4 以下
Filter4	ポテンシャル勾配	下位半分
Random	—	1/2 の確率で除外

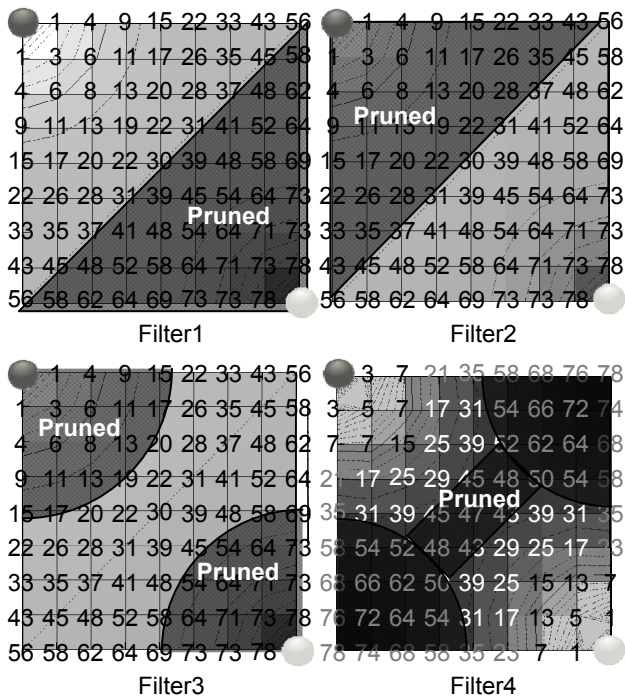


Fig. 5: Example of Pruning

2.3. 枝刈り機能を備えたモンテカルロ碁の棋力

枝刈り機能を備えたモンテカルロ碁の棋力は、枝刈り機能を備えたモンテカルロ碁（先手）とノーマルなモンテカルロ碁（後手）との対戦勝率によって表す。

先手後手共にノーマルなモンテカルロ碁であった場合、先手の勝率は57%であった（棋力は同じだが、囲碁は先手有利の為50%を上回る）。枝刈り機能を備えたモンテカルロ碁の勝率がこの57%を上回るか下回るかが、枝刈りの効果測定の一つの判断となる。2.2.3で示したポテンシャルフィルタ及びランダムフィルタについて試行した。

合法手に設けた閾値（2.4節）毎に1000局の対戦を行いその勝率を求めた。

2.4. 閾値によるフィルタ有効・無効の切替

各フィルタの効果・傾向を測る際、局中、合法手の数に対して閾値を決め、残りの合法手の数が閾値以上の場合はフィルタを有効にし、逆に残りの合法手の数が閾値未満の場合はフィルタを無効にした。本実験では9路盤を扱ったので、合法手は81から1までである。それら一つ一つの数を閾値として、各フィルタに対して81通りの試行を行った（但し合法手が81の場合は、碁盤上に碁石が無い為、ポテンシャルフィルタの効果は無い）。

モンテカルロ碁は探索法であり、ゲーム木が小さい程その精度が高まる事は明らかである。枝刈りが有益になるのは合法手が多い（探索木が大きい）序盤であり、逆にモンテカルロ探索の精度が高まる終盤では枝刈りは有害（不要な偏り）になると考えた。初手からポテンシャルフィルタを使用し、使用を止めるタイミングを一手ずつ変えて棋力を測ることで、フィルタが有益から有害となる（勝率57%を下回る）境界を調査した。

2.5. ポテンシャルフィルタにおけるモンテカルロ探索との一致率及び最善手の含有率

合法手の数（局面の進行）によって変化する、ポテンシャルフィルタの結果とモンテカルロ探索の結果との適合性を測った（各合法手のサンプル数：1000）。

ノーマルなモンテカルロ碁同士の対局中、各局面でポテンシャルフィルタを掛けたとして（実際には使用しない）、その時の合法手の数（81～1）と以下の2点の関連性を測定した。この試行によって各フィルタの傾向を捉えた。

1. ポテンシャルフィルタにより有効とされた合法手群と、モンテカルロ探索によって有効とされた合法手群との一致率（式(5)）

$$(\text{一致率}) = \frac{\text{Num}_m \cap \text{Num}_p}{\text{Num}_p} \quad (5)$$

Num_m : モンテカルロ探索が有効とした合法手の数

Num_p : ポテンシャルフィルタが有効とした合法手の数

但しモンテカルロ探索によって有効とされた合法手とは、モンテカルロ探索によって高い値を示した合法手の中の上位半分とする。

2. ポテンシャルフィルタにより有効とされた合法手群の中に、モンテカルロ探索による最善手が含まれる確率

2.6. ポテンシャルフィルタの組み合わせ

3.1 枝刈り機能を備えたモンテカルロ碁の棋力の結果より序盤に高い棋力を示した Filter4 と、中盤で棋力の下降率が最も低かった Filter3 を組み合わせ（Combination）、他のポテンシャルフィルタと同様に棋力の推移を測った。序盤を Filter4 として高い棋力を獲得し、棋力が下降してくる辺りの合法手 68 から Filter3 に切り替え、棋力の推移を測った。

2.7. 実験環境

2.7.1. 計算機

OS	:	Mac OS X 10.6.4
CPU	:	Intel Core 2 Duo 2.66GHz
メモリ	:	8GB
実装言語	:	Java

2.7.2. 囲碁

盤のサイズ	:	9路盤
ルール	:	中国式

2.7.3. モンテカルロ碁

プレイアウト数	:	100
---------	---	-----

（モンテカルロ探索における、探索の精度を決める要素）

3. 結果

3.1. 枝刈り機能を備えたモンテカルロ碁の棋力

2.3 枝刈り機能を備えたモンテカルロ碁の棋力の試行結果を Fig. 10-Graph1 (左軸) に示す。ノーマルなモンテカルロ碁同士が対戦した場合の先手の勝率 57%と、ランダムフィルタ (Random) の結果との比較が各枝刈り方法の効果・傾向の判断要素となる。

3.1.1. Randomによる棋力の変化

序盤だけに Random を使用したときは 57%付近を推移していたが、合法手 68 辺りから次第に下降し始めた。中盤が最も下降率が大きくなるが、後半では下降率が小さくなった。終盤まで Random 使用した場合、勝率は 15%程度になった。

3.1.2. Filter1

序盤は 57%及び Random を上回った。合法手 76 で 57%及び Random を下回り、合法手 50 まで Filter1 を使用すると、勝率 0 に収束した。

3.1.3. Filter2

序盤は 57%及び Random を上回った。合法手 72 まで Filter2 を使用すると 57%及び Random を下回り、合法手が 30 まで Filter2 を使用すると勝率 0%に収束した。

3.1.4. Filter3

4 種の中で唯一、最初の合法手 80 から 57%及び Random を下回った。序盤の勝率の下降率は他の方法と比較し最も大きい、逆に合法手 70 辺りからは下降率が最も小さくなった。後半は合法手 45 辺りから Filter2、Filter4 と重なり、合法手 30 辺りで勝率 0%に収束した。

3.1.5. Filter4

Filter4 は合法手 64 まで 57%を上回った。全てのフィルタの中で、57%を上回る期間が最も長く、上回る値も大きい。しかし中盤は最も大きく下降し、合法手 60 以降でランダムな枝刈りを下回った。後半は合法手 45 辺りから Filter2、Filter3 と重なり、合法手 30 辺りで勝率 0%に収束した。

3.2. ポテンシャルフィルタにおけるモンテカルロ探索との一致率及び最善手の含有率

2.5 ポテンシャルフィルタにおけるモンテカルロ探索との一致率及び最善手の含有率の試行の結果、一致率を Fig. 10-Graph2、また一致率の標準偏差を Fig. 10-Graph4 に示す。そして含有率を Fig. 10-Graph3 に示す。Random はその設定から一致率 50%、一致率の標準偏差 0、含有率 50%となる (理論値)。各方法の傾向を捉える為にこれらの値が重要な判断基準となる。

3.2.1. Filter1

一致率・含有率共に終始 50%辺りを推移した。Filter1 と Filter2 はその条件より 50%を軸にして線対称となった。

3.2.2. Filter2

一致率・含有率共に終始 50%辺りを推移した。Filter1 とは 50%を軸にして線対称であった。

3.2.3. Filter3

一致率・含有率共に序盤は 50%を下回った。しかし合法手が減るに連れて上昇し、中盤はごくわずかだが 50%を上回った。4 種の中で唯一値の上昇を見せた。後半では 50%辺りを推移した。

3.2.4. Filter4

一致率・含有率共に序盤は 50%を上回った。4 種の方法の中で最も大きい率を示す。しかし合法手が減るにつれて下降し、終盤は 50%辺りを推移した。

3.3. ポテンシャルフィルタの組み合わせ

2.6 ポテンシャルフィルタの組み合わせ (Combination) の試行結果を Fig. 10-Graph1 に示す。

Filter4 単体であれば合法手 68 辺りから棋力が下降するが、そのタイミングで中盤の下降率が最も低い Filter3 に切り替えると勝率の下降率が小さくなった。結果、57%を下回るのは合法手 58 から、そして Random を下回るのは合法手 50 辺りからと、枝刈りの効果が改善された。

3.4. 計算量の変化

本実験では、一局に要したプレイアウトの総数を、一局に要した計算量として示す。本実験の設定では、一局に要するプレイアウトの総数は、(モンテカルロ探索を行った合法手の数の累計) × 100 である。例えば合法手の数 81~2 まで全ての局面でモンテカルロ探索を行った場合、そのプレイアウト数は $(81+80+\dots+2) \times 100=332000$ となる。各局面で、フィルタによって除外した合法手の数に比例して全体のプレイアウト数が削減される。

枝刈りの際に必要なとなるポテンシャルモデルの計算量は、プレイアウトに必要な計算量と比較して極めて小さいので考慮に入れない (一局面において、次手を決める際に行われる数千回のプレイアウトにはおおよそ数十万回の乱数の発生が必要となる。それに対し、ポテンシャルフィルタが必要とするのは数百回の加減算である)。各合法手までフィルタを使用したときの計算量 (一局に要するプレイアウト数) を Fig. 10-Graph1 (右軸) に示す。また勝率 57%を下回る直前の合法手の数を境界として、各フィルタの削減結果を Table. 2 に示す。

Table. 2: 計算量の削減

	57%境界	プレイアウト数	削減率
Nomal	—	332000	—
Random	69	287300	13.4
Filter1	77	316300	4.7
Filter2	73	301400	9.2
Filter3	81	332000	0
Filter4	64	270800	18.4
Combination	59	255500	23.0

4. 考察

本研究の目的はポテンシャルモデルを用いた枝刈りによって、モンテカルロ碁に要する計算量を削減する事である。

実験の結果、用意した4種のポテンシャルフィルタがそれぞれ特性を見せた中、ポテンシャル勾配値を基にしたポテンシャルフィルタによる枝刈りが最も有効であった。また、このポテンシャルフィルタと、異なるポテンシャルフィルタを組み合わせた手法は更に高い有効性を示した。

モンテカルロ碁は、ゲーム後半の詰め（合法手が少なく、正確な手筋を必要とする段階）に入ると、判断の精度が高くなる。逆にゲーム序盤の布石（合法手が多く、領地獲得を見据えてバランス良く石を配置する段階）では、有効な手を選びきれない。そこで我々は、ゲーム序盤においてポテンシャルモデルを利用したフィルタを用い、次手の候補を絞り込み、計算量の削減を図った。

ノーマルなモンテカルロ碁との対戦勝率、ポテンシャルフィルタにより有効とされた合法手群とモンテカルロ探索により有効とされた合法手群の一致率、ポテンシャルフィルタにより有効とされた合法手群における、モンテカルロ探索による最善手の含有率から、各ポテンシャルフィルタの効果・傾向を捉えた。

4.1. Random

Randomはその設定から、1/2の確率でモンテカルロ探索における最善手を打つ。確率的に悪くとも数番手を打つ。序盤のモンテカルロ碁は判断の精度がもともと低く、最善手も数番手も評価として大きな差は生じず、勝率は変化しなかったと考えられる。しかし、合法手の数が減少していくに連れて、モンテカルロ碁の精度が高まり、差が大きくなっていったと考えられる。しかし最終的な勝率が15%に下げ止まるのは、フィルタを掛け続けても、良くて最善手、悪くとも数番手（悪手ではない）を打ち続ける為、ポテンシャルフィルタと異なり、ある程度の棋力の低下で収まった為と考える。

4.2. Filter1

Filter1は、その設定から黒石を固めて打つバイアスとなる。序盤の着手においては、陣地を強固に固める有効手となるが、そのバイアスを掛け続けると陣地が広がらず、結果的に白石に大きな地を獲得される。

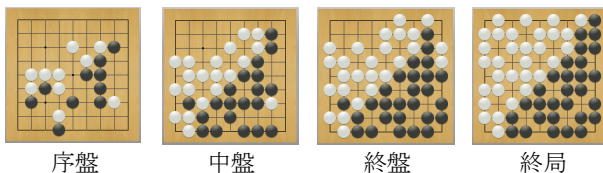


Fig. 6: A Game Using Filter1

57%以上を保つならば、合法手76までは有効な枝刈りに成り得ることが分かる。序盤の57%を上回っている部分は、モンテカルロ探索の精度が不十分で無駄と判断しきれなかった枝をFilter1が刈った結果と考えられる。それ以降は逆に有効な枝を刈ってしまい棋力が減少していったと考えられる。

一致率・含有率においては、双方が50%を推移していることから、Filter1はモンテカルロ探索との適合性が高くない。モンテカルロ探索では置石周囲の合法手の値が高くなるといった大まかな傾向があるものの、それが疎らであったり、また隣接する合法手の値はそれ程高くならない場合があるなど、そういった複雑な振る舞いに対応しきれなかった為と考えられる。

4.3. Filter2

Filter2は、その設定から白石の周辺に打つバイアスとなる。序盤の着手においては、そのバイアスが相手を抑える有効手となったが、それ以降、ある程度地を固めた白石の周辺に置く黒石は搾取され易く、結果として白石に大きな地を獲得される。

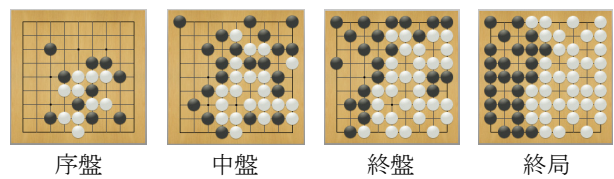


Fig. 7: A Game Using Filter2

合法手71までは有効な枝刈りに成り得ることが分かる。それ以降は有害な枝刈りとなって棋力が減少したと考えられる。

一致率・含有率においては、双方が50%を推移しており、合法手の数と、Filter2はモンテカルロ探索との適合性が高くない。白石の場合も、黒石の場合と同様に、モンテカルロ探索の複雑な振る舞いに対応しきれなかった為と考える。

4.4. Filter3

Filter3は、その設定から黒石と白石からより離れた交点打つバイアスとなる。黒石が固まらず、より広い範囲に黒石を置く事になるが、黒石を固めるべき場合や白石を攻めるべき場合に対応できず、結果として白石に大きな地を獲得される。

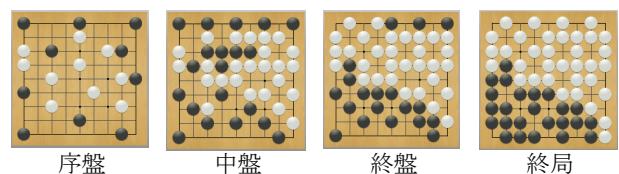


Fig. 8: A Game Using Filter3

最初の合法手80から有効な枝を刈ってしまっており、棋力が下降したと考えられる。しかし中盤は4種の中で最も下降が緩くなる。他よりは良い枝刈りがなされていると判断できる。

一致率・含有率においては、序盤から50%を下回っており、適合性が低い。しかし、中盤では値の上昇を見せる。序盤から中盤にかけて囲碁の傾向が変化し、且つFilter3がそれに対応したと考えられる。

4.5. Filter4

Filter4 は、その設定からポテンシャル勾配値の高くなる黒石の周辺、白石の周辺、黒石と白石がある程度の接近した場所に打つバイアスとなる。

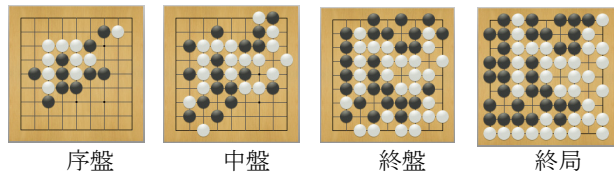


Fig. 9: A Game Using Filter4

それらは囲碁において重要な交点であり、その為4種の中で最も効果のある枝刈りとなったと考えられる。しかしそれだけでは序盤以降は対応できず、棋力が低下した。一致率、含有率においても、序盤は共に高いが、終盤にかけて常に減少していった。

4.6. Combination

Filter4 単体よりも Filter3 と組み合わせた場合の方が良い結果となったのは、序盤以降で変化する囲碁の傾向に対して、Filter4 に替わり Filter3 が対応した為と考えられる。

4.7. 枝刈りの終盤の振る舞い

Random を終始使用した場合でも勝率は 15% に止まるが、ポテンシャルフィルタを使用した場合は全て 0% に収束する。Random はその設定から、一致率・含有率は 50%、一致率の標準偏差は 0 である (理論値)。その為、正確な手筋を必要とする詰めの段階でも、良くてモンテカルロ探索による最善手、悪くとも数番手を打つので、ある程度の勝率に下げ止まったと考えられる。それに対して、全てのポテンシャルフィルタに関しては、合法手の数が少ない場合の一致率・含有率は Random と同じほぼ 50% となるが、一致率の標準偏差は 0 ではなく、合法手の数が少なくなるに連れて値が上昇した。これは、一致率が高い場合と低い場合が混在しているということである。つまり、モンテカルロ探索による最善手を打つ確率は Random と同じ 50% だが、後の 50% では Random の打つ数番手よりも悪い手を打つ可能性があるということである。中盤から終盤にかけては、詰めの要素が強くなる為、より正確な手筋を間違えずに打つことが求められるが、ポテンシャルフィルタによる枝刈りが正しい手の障害となり、勝率を 0% までに下げたのだと考える。

5. 結論

モンテカルロ碁において、ポテンシャルモデルを用いた枝刈りを試行し、計算量削減の効果を調査した。

ノーマルなモンテカルロ碁同士を対戦させた場合の先手の勝率 57% を基準とし、それを下回らずに削減できるプレイアウト数を求めた。その削減量を各枝刈り方法 (ポテンシャルフィルタ) の評価とした。

本実験で用意したポテンシャルフィルタの中で、ポテンシャル勾配値に基づいた枝刈り方法 (Filter4) が最も有効であり、計算量の削減率は 18% であった。これはポテン

シャル勾配値が、囲碁において重要な点、黒石と白石の周辺、黒石と白石が接近した箇所を選定する様に機能した為と考える。

更に、Filter4 で棋力が落ち始めるタイミング (合法手数 68) から、中盤で最も良い枝刈りを行ったと見られるポテンシャルフィルタ (Filter3) と切り替えることで計算量の削減を 23% まで改善することができた。囲碁は合法手の数 (局面の進行) に従い、判断の傾向が変化する事を伺わせる。本実験では、その傾向の変化に近づけることで、ある程度の計算量の削減に成功したと考える。

しかし、ポテンシャルフィルタが有効であったのは合法手の数の多い序盤であり、中盤から終盤にかけては勝率を 0% まで落としていった。これは、合法手の数が減るに連れて高まるモンテカルロ探索の精度にポテンシャルフィルタが対応しきれなかった為と考えられる。精度が高まったモンテカルロ探索にとっては、ポテンシャルフィルタの機能は有害なものとなった。

本実験では、4 種のポテンシャルフィルタを用意し、その設定によってランダムな枝刈りとは異なる個々の効果・傾向が生じる事を確認した。これは囲碁の判断の傾向と、ポテンシャルモデルとの間に関連性があることを示唆する。

今回は 9 路盤や、モンテカルロ探索の設定要素であるプレイアウト数が 100 といったことで、限定された環境下であった為、実戦的な段階までの調査とはならないが、ポテンシャルモデルによる枝刈りの可能性を示す事が出来た。今後は更に大きな盤やプレイアウト数を増やすなど、様々な環境下で試行する必要がある。

7. 参考文献

- Brügmann, Bernd. "Monte Carlo Go." Technical report, Physics Department, Syracuse University, 1993.
- Coulom, Rémi. "Computing elo ratings of Go." 2007.
- Gelly, Sylvain, Yizao Wang, R'emi Munos, and Olivier Teytaud. "Modification of UCT with Patterns in MonteCarlo Go. ." 2006, 1-19.
- Kitagawa, Ryouhei, Teppei Kurita, and Takashi Chikayama. *Pruning in UCT search based on limitedness of computational resource.* magisterial thesis, The University of Tokyo, 2009.
- Kocsis, Levente, and Csaba Szepesvári. *Bandit based Monte-Carlo Planning.* Enropean Conference on Machine Learning, 2006, 282 - 293.
- Nakamura, Katsuhiko, and Syuhei Kitoma. "Analyzing Go Board Patterns Based on Numerical Features." *IPSJ Journal* (The Information Processing Society of Japan) 43, no. 10 (10 2002): 3021 - 3029.
- Tajima, Morihiko, and Noriaki Sanechika. "Determination of Inner Areas of Moyo by Potential Values." <http://staff.aist.go.jp/tajima.m/gpw04.files/frame.htm> (accessed 23, 2011).
- Yajima, Tosio. *Effect of Stone and Effect of Table.* SIG Technical Reports, Surugadai, Kanda, Chiyoda: The Information Processing Society of Japan, 2009, 41-46.

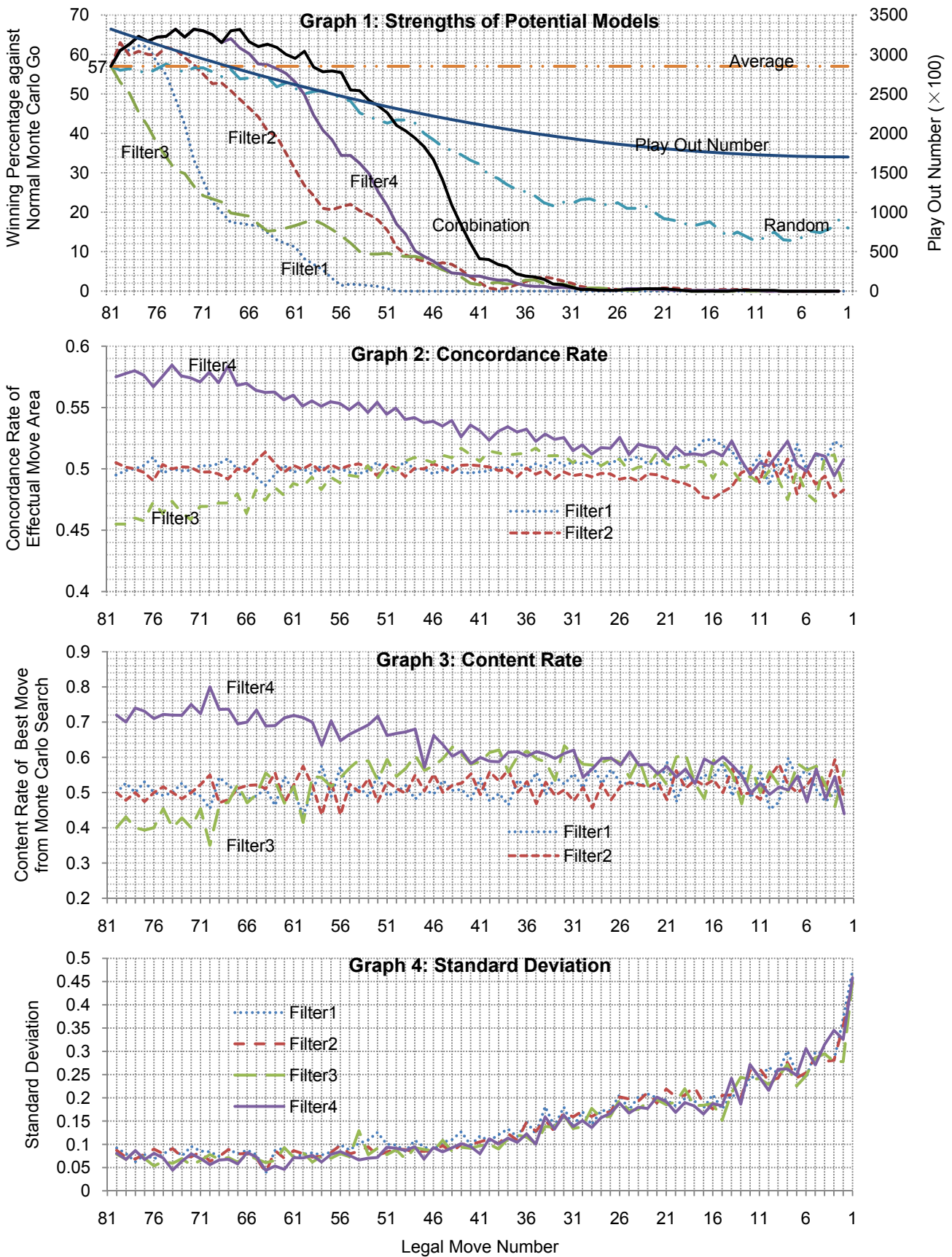


Fig. 10: Graphs