

# デジタルカーリングの最終局面における獲得可能点数の推定のためのショットシミュレーション回数の削減

益子 直<sup>1</sup> 吉仲 亮<sup>1</sup> 篠原 歩<sup>1</sup>

概要：デジタルカーリングの最終ショットとなる 16 投目のショットは、各エンドの獲得点数が決まるため、ゲームの勝敗を左右する極めて重要なショットである。よって、デジタルカーリング AI には、より多くの点数が獲得できる最終ショットを高速に導き出すことが求められる。本稿では、最終ショットにおけるシミュレーション回数の削減を行い、獲得点数が多くなるショットを実用的な時間で算出する方法を提案する。また、提案手法の有用性を実験により検証し、2016 年 GAT 杯に出場した全ての AI よりも良いショットが導けることを示す。

## Reducing the number of shot simulations for estimating the achievable score in the final phase of digital curling

SUNAO MASHIKO<sup>1</sup> RYO YOSHINAKA<sup>1</sup> AYUMI SHINOHARA<sup>1</sup>

**Abstract:** In digital curling, the final shot, i.e. the sixteenth shot of each end is very important because it decides the score of the end. Therefore, a digital curling AI is required to quickly calculate the final shot which achieves a higher score. In this paper, we propose methods that find promising final shots in practical time by reducing the number of shot simulations. Also, we show that final shots by our proposed methods are better than those by all AIs which participated in GAT 2016 by experiments.

### 1. はじめに

カーリングは、氷上のチェスとも呼ばれる戦略性の高いスポーツである。デジタルカーリングは、カーリングの戦術支援のために考案され、カーリングのルールを基に作られた二人対戦ゲームである [1]。デジタルカーリングは二人零和有限不確定完全情報ゲームに分類され、現在では AI 同士の対戦が主流となっている。

デジタルカーリングの AI は、ゲームに勝利するため、自身が有利となるような良いショットを持ち時間の範囲内で考える必要がある。良いショットを発見する手段として、デジタルカーリングにはショットの物理シミュレーションを行う機能がある。物理シミュレーションを用いる事により、盤面とショットの情報を入力すると、ショットを行った後の盤面を出力として得ることができる。よって、様々な

ショットに対し十分な試行回数を行えば、自身が有利になる良いショットを発見することができる。しかし、シミュレーションには時間がかかるため、有望なショット候補のみに絞ってシミュレーションを行うことが重要となる。

デジタルカーリングの代表的な AI としては、モンテカルロ木探索を行う歩 [2] や、ゲーム木探索を行うじりつくん [3] が挙げられる。また、現在では機械学習を用いてショットを決定する AI も検討されている [4]。AI の強化を考える際、シミュレーション回数の削減は、探索 AI や機械学習 AI のどちらに対しても非常に有効である。探索においては、より深い盤面に対してシミュレーションを行うことが可能となり、機械学習においては、学習用データを高速に作成することが可能となる。

本稿では、デジタルカーリングにおける最終ショットとなる 16 投目に関して、探索時間を削減する方法を提案する。まず準備として、高得点となるショットの傾向を把握するため、大量のログデータに対しあらゆるショットのシ

<sup>1</sup> 東北大学大学院情報科学研究科  
Graduate School of Information Sciences, Tohoku University

シミュレーションを行い、スコアマップとして可視化を行った。スコアマップを観察して得られた知見を基に、最終ショットの探索時間を削減する方法を提案し、実験により提案手法の有効性について検討を行う。

## 2. デジタルカーリング

本節では、まずカーリングのルールを説明し、次にデジタルカーリングについて説明する。ただし、本研究において重要ではないルールに関しては説明を省略するものとする。

### 2.1 カーリングのルール

カーリングは氷のリンク上で行われ、2つのチームが交互に目標の円をめがけて石を氷上に滑らせ、点数を取り合うウィンタースポーツである。目標となる同心円をハウスと呼び、滑らせる石をストーンと呼ぶ。また、ストーンを氷上に滑らせることをショットと呼ぶ。

カーリングにおける1ゲームはエンドと呼ばれる単位に分割されており、エンドを特定の回数繰り返すことによりゲームが終了する。各チームには、1ゲームを通しての持ち時間として、73分間の考慮時間が与えられる。ゲーム終了前に持ち時間が切れた場合には、現在の点数状況に関わらず負けとなる。各エンドでは先手、後手ともに交互に8回ずつ、計16回のショットを行う。エンドが終了した時点で、ハウスの内側にあり、かつ相手チームの全てのストーンより内側にある自分のストーンの数、が、チームの得点となる。なお、ハウスの一番外の青い円に一部でも触れているストーンは全てハウスの内側にあるとみなす。エンド終了時にハウス内にストーンが存在せず、両チームともに点数が入らない場合はブランクエンドと呼ばれ、次のエンドにおける先後は直前のエンドと同じとなる。どちらかに点数が入った場合には、点数を入れたチームが次のエンドで先手、点数を入られたチームが後手となる。先後の違いが発生する可能性がある以外は、第2エンド以降も第1エンドと同様にゲームが進行し、ゲームが終了した時点での総得点が多いチームの勝利となる。

### 2.2 デジタルカーリング

デジタルカーリングは、カーリングのルールを基にしたゲームであり、図1に示す計算機上での仮想リンクにおいて、2人のプレイヤーが点数を取り合うゲームである。デジタルカーリングが通常のカーリングと異なる部分について述べる。ただし、本稿で使用されるゲームのパラメータは、2016年GAT杯以降のルールに準拠して設定するものとする。(http://minerva.cs.uec.ac.jp/curling/wiki.cgi?page=GAT\_2016)

まず、デジタルカーリングにおける各プレイヤーの考慮時間は219秒、1ゲームは8エンドであり、従って1回の

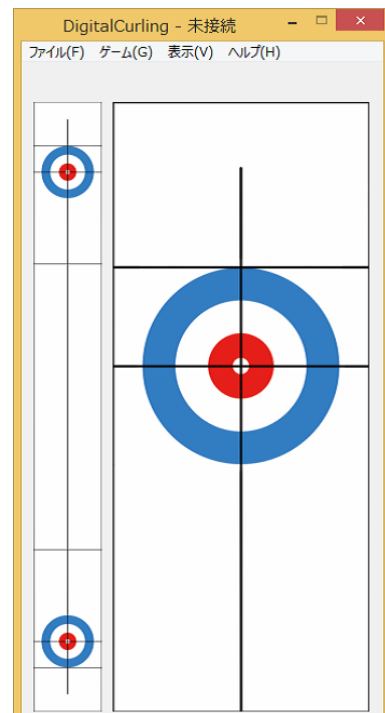


図1 デジタルカーリングリンク

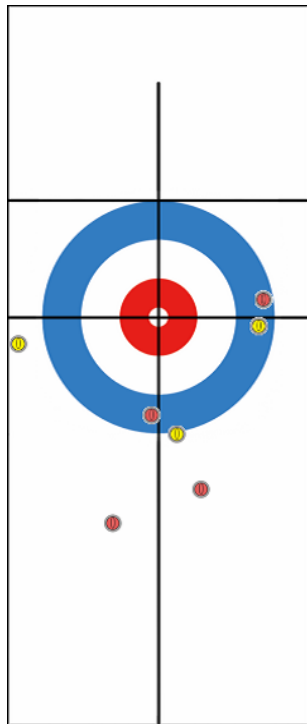
ショットに使用できる平均考慮時間は約3.4秒となっている。また、デジタルカーリングにおけるショットは、 $x$ 軸初速度  $v_x \in \mathbb{R}$ 、 $y$ 軸初速度  $v_y \in \mathbb{R}$ 、回転方向  $angle \in \{0, 1\}$  の3つのパラメータで指定する。回転方向  $angle = 0$  は右回転を表し、 $angle = 1$  は左回転を表す。ショットパラメータのうち  $v_x$  と  $v_y$  には、初速度乱数として  $x$  軸に対し  $\mu = 0, \sigma = 0.0725$ 、 $y$  軸に対し  $\mu = 0, \sigma = 0.2900$  の正規乱数が加わるため、ストーンは必ずしも期待した動きをするとは限らない。デジタルカーリングには、物理シミュレーションを行う機能が存在し、盤面および3つのショットパラメータを入力すると、ショット後の結果盤面を得ることができる。ただし、シミュレーションには時間がかかり、毎秒500回程度しか行うことができない。プレイヤーが持ち時間の範囲で良いショットを探索するには、シミュレーションの候補を絞り込み、有望なショットのみを探索することが必要となる。

## 3. スコアマップによる可視化

本節では、シミュレーション回数の削減の方針を定めるために作成したスコアマップによる最終ショットの獲得点数の可視化について述べる。スコアマップには、乱数を考慮しない固定スコアマップおよび乱数の影響を考慮する乱数考慮スコアマップの2種類があり、それぞれについて説明を行う。

### 3.1 スコアマップ

スコアマップは、盤面と3つのショットパラメータを入力とし、エンド終了時の得点を出力とするヒートマップ



(a) 元盤面 (手番 : 赤)



(b) スコアマップ

図 2 固定スコアマップ



図 3 スコア配色表

である．スコアマップは黒い線を挟み 2 枚の画像で可視化される．赤プレイヤーが最終ショットを行う際のスコアマップの例を図 2 に示す．スコアマップの左側の画像は、回転方向  $angle = 0$  で右回転の場合を表し、右側の画像は  $angle = 1$  で左回転の場合を表す． $x$  軸は初速度  $v_x$  を表し、それぞれの画像で左端が  $v_x = -3.0$ 、右端が  $v_x = 3.0$  と対応し、120 のセルに等分割されている． $y$  軸は初速度  $v_y$  を表し、それぞれの画像で上端が  $v_y = -33.7419$ 、下端が  $v_y = -28.7419$  と対応し、50 のセルに等分割されている． $v_x$  と  $v_y$  の値域は、ハウスに届かないショットや、リンクから出てしまうショットのような、獲得点数に影響する可能性が低いショットを除外するように決定している． $y$  軸に関し、上端の 0 行目は  $v_y$  が最大であるショットを表し、ハウスの中心にストーンを止めるショットは 41, 42 行目付近である．各セルの色は、対応するショットパラメータで最終ショットを行った際の獲得点数を示している．点数による配色を図 3 に示す．

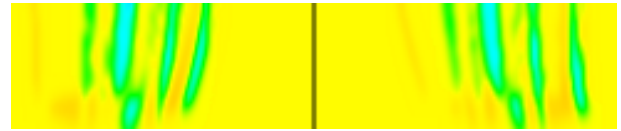


図 4 乱数考慮スコアマップ

### 3.2 固定スコアマップの作成

固定スコアマップは、 $120 \times 50$  のセルに対し、対応したショットパラメータで物理シミュレーションを行うことにより作成する． $v_x, v_y$  の最小・最大値と分割数より、それぞれの  $angle$  における左から  $x$  番目、上から  $y$  番目のセルのショットパラメータは次のように表される．

$$v_x = \frac{3.0x - 3.0(119 - x)}{119} \quad (x \in \mathbb{Z} \mid 0 \leq x \leq 119)$$

$$v_y = \frac{-28.7419y - 33.7419(49 - y)}{49} \quad (y \in \mathbb{Z} \mid 0 \leq y \leq 49)$$

セルに対応するショットパラメータで物理シミュレーションを行うと、エンド終了時の盤面を得ることができ、獲得点数を計算することができる．全てのセルに対して獲得点数を計算し、図 3 に従った色を塗ることによって固定スコアマップが完成する．

### 3.3 乱数考慮スコアマップの作成

乱数考慮スコアマップは、得られた固定スコアマップに対して初速度乱数を考慮し、乱数の確率分布を行列の形で表す事によって、畳み込み演算により得られる点数期待値のヒートマップである．例として、図 2(b) の固定スコアマップから得た乱数考慮スコアマップを図 4 に示す．

乱数の確率分布は、1% 以下の分布については無視するものとして、 $x$  軸に関しては中心及び左右 5 セルずつ、 $y$  軸に関しては中心及び上下 9 セルずつのみを考慮する．乱数考慮スコアマップの各点の点数期待値  $rand-score_{x,y}$  は、固定スコアマップの点数  $score_{x,y}$  と確率分布  $probability_{x',y'}$  の畳み込みで表されるため、単純に書けば次のように表される．

$$rand-score_{x,y} = \sum_{x'=-5}^5 \sum_{y'=-9}^9 (score_{x+x',y+y'} \cdot probability_{x',y'})$$

ただし、確率分布に関して、参照箇所がスコアマップの範囲外で未定義である場合には確率 0 と考えることにする．

$$probability_{x',y'} = \begin{cases} 0 & (score_{x+x',y+y'} \text{ が未定義のとき}), \\ probability_{x'} \cdot probability_{y'} & (\text{それ以外}). \end{cases}$$

すると、スコアマップの端の部分の点数の絶対値が小さくなってしまいうため、補正をかけるために、乱数考慮スコアマップの式を次のように書き換える．

$$rand-score_{x,y} = \frac{\sum_{x'=-5}^5 \sum_{y'=-9}^9 (score_{x+x',y+y'} \cdot probability_{x',y'})}{\sum_{x'=-5}^5 \sum_{y'=-9}^9 probability_{x',y'}}$$

それぞれの軸に対する確率分布は、累積分布関数 CDF を用いて、各軸におけるセル間の距離を  $x$ -interval,  $y$ -interval とおくと、

$$probability_{x'} = CDF\{(x' + \frac{1}{2}) \cdot x\text{-interval}\} - CDF\{(x' - \frac{1}{2}) \cdot x\text{-interval}\}$$

$$probability_{y'} = CDF\{(y' + \frac{1}{2}) \cdot y\text{-interval}\} - CDF\{(y' - \frac{1}{2}) \cdot y\text{-interval}\}$$

となる。  $x$ -interval と  $y$ -interval は、スコアマップの範囲と分割数から算出できるため、次のように表される。

$$x\text{-interval} = \frac{3.0 - (-3.0)}{119}$$

$$y\text{-interval} = \frac{-28.7419 - (-33.7419)}{49}$$

以上より、乱数考慮スコアマップにおける点数期待値が計算できるため、固定スコアマップと同様に色を塗って乱数考慮スコアマップが完成する。

乱数の確率分布を利用する手法は既に存在するが [5]、既存手法が  $x, y$  座標に対して乱数を考慮しているのに対し、本手法では  $v_x, v_y$  座標に対し乱数を考慮している。デジタルカーリングにおける乱数は  $v_x, v_y$  に対してかかるため、本手法の方が計算が正確、または計算が単純と考えられる。

#### 4. 提案手法

本節では、デジタルカーリングの各エンドの最終ショットにおけるシミュレーション回数を削減しつつ、なるべく精度を保つ方法を提案する。

##### 4.1 人間の知見を用いたシミュレーション回数の削減

2016 年 GAT 杯のログデータから作成された 11960 盤面のスコアマップを観察して得られた知見として、獲得点数の分布は  $y$  軸方向に同じ点数が続くような相関性が高く、 $x$  軸方向は相関性が低いことが分かった。よって、図 5 のようにスリット状にシミュレーションを行えば、少ないシミュレーション回数で、精度があまり落ちない獲得点数予測ができると考えた。また、獲得点数が高くなるショットパラメータは図 5 における  $y = 0$  の  $v_y$  が最大となるような行、または  $y = 42$  のハウス中心付近にストーンが停止する行の上に多い事を見つけた。また、獲得点数が高くなる行だけでなく、周辺についても調査すべきであると考え

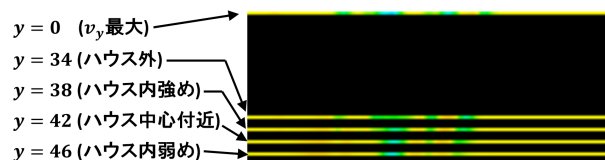


図 5 部分探索の範囲

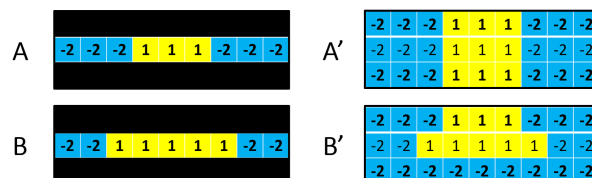


図 6 周辺探索の重要性

た。周辺探索が重要となる例を図 6 に示す。図 6 の A と B について比較を行うと、B の方が A より優れているように考えられる。しかし、A と B に関して周辺を調査した A' と B' について比較を行うと、A' の方が優れていることが分かる。獲得点数の予測ミスを防ぎ、予測精度をより高めるため、周辺の探索も重要である。以上より、点数の高い行および少し離れた行に対してスリット状にシミュレーションを行うことにより、図 5 のような乱数考慮部分スコアマップを得ることができる。乱数考慮部分スコアマップの中で一番点数の高い地点を最終ショットとして採用すれば、シミュレーション回数を削減しつつ、精度を保った点数予測ができると考えた。なお、乱数考慮部分スコアマップの計算は、通常の乱数考慮スコアマップよりも点数未定義領域が増加するが、同様に行うことができる。

##### 4.2 n 行探索によるシミュレーション回数の削減

スコアマップから得た人間の知見ではなく、50 行の固定スコアマップの中から抽出すべき  $n$  行を全探索によって求めることを考える。50 行から  $n$  行を選択する組み合わせを全て考えるため、 $n$  が増えると計算量が爆発してしまうが、 $n$  が小さい場合には全探索を行うことが可能である。また、50 行の中でも重要度が高いと考えられる、一番強いショットの行である  $y = 0$ 、ハウス中心付近を狙う行である  $y = 41$  を必ず選択し、残りの  $n - 2$  行を探索する 2 行固定探索を考えれば、ナイーブに全探索を行うよりも探索可能行数が増加すると考えた。

採用する  $n$  行の決定方法としては、まず、大量の最終ショットの盤面を用意し、それぞれに対して一番高い獲得点数を予測できた行の組み合わせを見つける。次に、見つけた組み合わせについて、組み合わせの要素である各行に対し、選択された回数をカウントする。ただし、一番点数が高くなる組み合わせが  $m$  個存在した場合には、増やす選択回数のカウントは  $\frac{1}{m}$  とする。最後に、全ての最終ショットの盤面に対して選択回数をカウントした後、前後の  $y$  よ

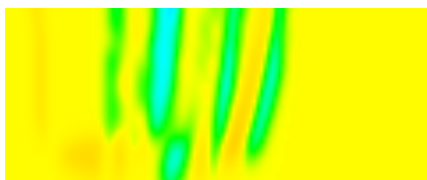


図 7 全域探索

りも選択回数が多い, ピークとなるような  $y$  を選択回数が多い順に  $n$  行採用する. 採用する行を決定した後は, シミュレーションにより乱数考慮部分スコアマップを得て, 一番点数が高い地点を最終ショットとして採用する.

#### 4.3 弱ショットの無視による精度向上

最終ショットを行う局面において, ハウスに届かないショットに関しては獲得点数に関わる可能性は少ない. また, ハウス内に入るショットでも, ハウス中心よりも奥を狙うショットは, 別のストーンをはじき出したり, 別のストーンにぶつけて自分のストーンを止める用途があるが, ハウス中心よりも手前を狙うことによる利点は少ない. 以上より, ハウス中心より手前を狙うショットが最善となる可能性は低いと考えた. しかし, 図 6 に示したような獲得点数の予測ミスによって, ハウス中心より手前を狙うショットが最善と予測してしまう場合がある. よって, 乱数考慮部分スコアマップを作成した際に,  $43 \leq y \leq 49$  の行に一番高い点数の地点があると予測した場合には,  $0 \leq y \leq 42$  の範囲内で一番高い点数の地点を最終ショットとして採用することで, 獲得点数の予測精度が高くなると考えられる.

### 5. 実験

本節では, 提案手法の有用性を調べる実験を各種行う. 本実験に使用する計算機のスペックは以下の通りである.

- OS: Windows 7 Professional
- CPU: Intel Xeon CPU W3565 @ 3.20GHz
- コア数: 8
- メモリ: 12GB
- HDD: 2TB

#### 5.1 人間の知見による探索範囲の変化と点数期待値の損失

4.1 章で述べた探索範囲の絞り込みによる点数期待値の損失を調べる. 乱数考慮スコアマップの全域を調べた場合における点数の最大期待値と, 乱数考慮部分スコアマップの点数の最大期待値を比較し, シミュレーション回数の削減により期待値がどれだけ減少するかを実験により評価する. スコアマップのうち, シミュレーションを行う部分を表したものを図 5, 図 7 に示す.

実験は図 5 における 5 種の  $v_y$  の中から数種類についてシミュレーションを行い, 期待値が最大となる点を求める. ただし, 乱数考慮部分スコアマップによって求められた期

表 1 探索範囲の変化による点数期待値の損失

探索範囲 (図 5 参照)	平均最大期待値	全域探索との差分
全域	1.2054	-
$y = 42$	0.3194	0.8860
$y = 0, 42$	0.8072	0.3982
$y = 0, 38, 42, 46$	0.9303	0.2751
$y = 0, 34, 38, 42, 46$	0.9925	0.2129

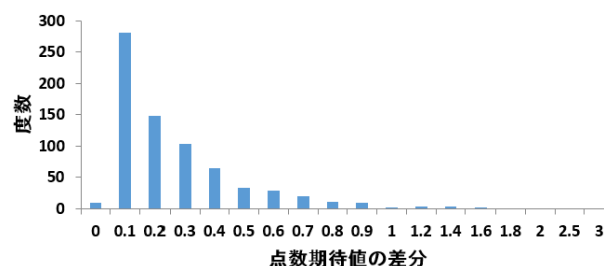


図 8 全域探索と 5 列探索の期待値差分のヒストグラム

待値は不正確であるため, 求めた点における全域探索の期待値を結果とする. 実験は, 2016 年 GAT 杯のログデータより, 16 投目の盤面状態を対戦 AI の全組み合わせにおいて, 先後別に 3 試合ずつ, 90 試合分抽出した 720 盤面に対して行う. 実験の結果を表 1 に示す.

表 1 より,  $y = 0, 42$  だけでなく近辺の  $y = 38, 46$  も調べた方が大幅に精度が向上することが分かる. 5 種の  $v_y$  を探索した場合における, 全域探索との差分のヒストグラムを図 8 に示す.

図 8 より, 5 種の  $v_y$  を全て探索した場合には, 1 点以上の誤差が生まれるケースが少ないことが分かる. 探索時間は元の 10% に抑え, 小さな誤差での探索に成功している.

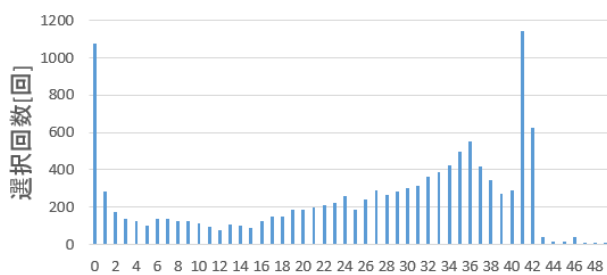
#### 5.2 ナイープ $n$ 行探索

4.2 章で述べたナイープ  $n$  行探索の実験を行う. 全探索により一番点数が高くなる行の組み合わせを発見し, 各行が一番点数が高くなる組み合わせの要素として何回選択されたかを記録する. 一番点数が高くなる組み合わせが  $m$  個存在した場合の選択回数は  $\frac{1}{m}$  回とする. 実験は 2016 年 GAT 杯のログデータより, 16 投目の盤面状態を 1495 試合全て抽出した 11960 盤面に対して行う. 探索をナイープに行った場合における, 探索行数別の結果を図 9 に示す. ただし, 3 行の探索にかかる実験時間が約 4 日半であったため, 4 行以上の探索は行っていない.

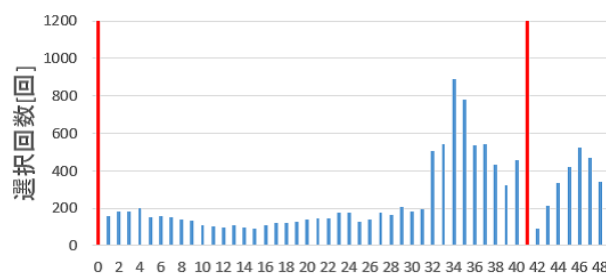
図 9 より, 行数に関わらず, 一番強いショットである  $y = 0$  と, ハウス中心に近い行である  $y = 41$  が多く選択され, 重要であることが分かる.

#### 5.3 2 行固定探索

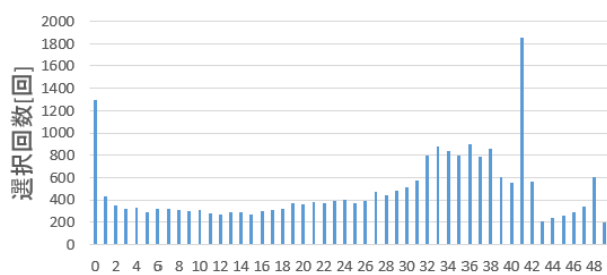
4.2 章で述べた 2 行固定探索について実験を行う.  $y = 0, 41$  は必ず選択するものとし, 残りの 3 行目や 4 行目としてどの行を選択するかを調査する. 実験は 2016



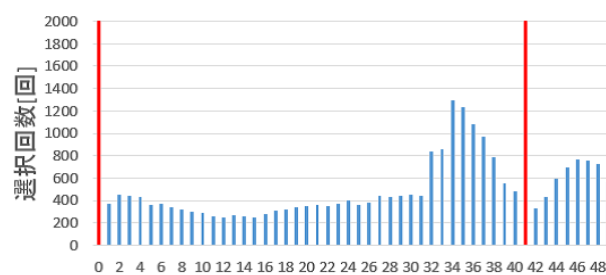
(a) 1行探索



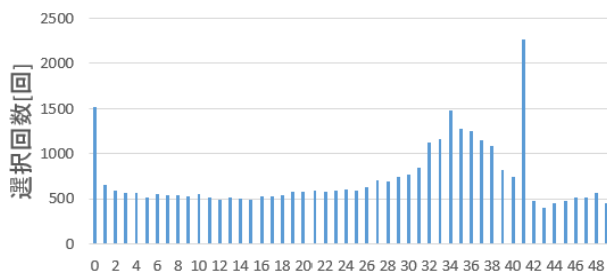
(a) 3行探索 (2行固定)



(b) 2行探索



(b) 4行探索 (2行固定)



(c) 3行探索

図 10 2行固定探索の結果

表 2 弱ショットの無視による点数の変化

$y = 46$ の採用	平均獲得点数の期待値
採用する	1.089
採用しない	1.141

した 11960 盤面に対して行うものとし、全盤面における獲得点数期待値の平均を比較する。結果を表 2 に示す。

表 2 より、 $y = 46$  を最終ショットとして採用しない方が点数期待値が向上することが分かる。

図 9 ナイブ  $n$  行探索の結果

年 GAT 杯のログデータより、16 投目の盤面状態を 1495 試合全て抽出した 11960 盤面に対して行う。3 行、4 行での場合の実験結果を図 10 に示す。図 10 中の赤い線は固定された行であり、グラフとしては見切れているが、全盤面で選択しているため選択回数は 11960 回である。また、5 行の探索にかかる実験時間は約 17 日間と推定されたため、5 行以上の探索は行っていない。

図 10 より、 $y = 0, 41$  の他には、 $y = 34, 46$  を選択すると良い結果が得やすいと考えられる。図 9(c) と図 10(a) を比較すると、2 行固定を行うことによって選択回数が全体的に減り、固定された行の付近の選択回数については大幅に減少、そしてピークが顕著になっていることが分かる。

#### 5.4 弱ショットの無視

4.3 章で述べた弱ショットの無視による効果を検証する。 $y = 0, 34, 41, 46$  の 4 行を選択して点数計算を行う際に、 $y = 46$  を最終ショットとして採用する場合と採用しない場合の比較をする実験を行った。実験は 2016 年 GAT 杯のログデータより、16 投目の盤面状態を 1495 試合全て抽出

#### 5.5 他 AI との終盤力比較

本提案手法による終盤力の強さを検証するため、2016 年 GAT 杯出場ソフトから、ダウンロード可能である全ての AI に対して終盤力の比較を行う。実験は 2016 年 GAT 杯のログデータより、16 投目の盤面状態を 1495 試合全て抽出した 11960 盤面に対して行うものとし、実際に試合で取った点数と提案手法による点数期待値を比較する。実際の点数と期待値の比較であるため、多少の誤差は出ると思われるが、実験に用いる盤面数が多くなれば平均誤差は小さくなると考えられる。

提案手法としては、4.2 章で述べた手法について、 $y = 41$  の 1 行のみ、 $y = 0, 41$  の 2 行、 $y = 0, 34, 41$  の 3 行、 $y = 0, 34, 41, 46$  の 4 行を調べる 4 種類を比較対象とする。また、4.1 章で述べた、 $y = 0, 38, 42, 46$  の 4 行、 $y = 0, 34, 38, 42, 46$  の 5 行を調べる方法についても比較対象とする。ただし、提案手法において、 $43 \leq y < 49$  については最終ショットとして採用しないものとする。また、AI はブランクエンドを狙って、1 点を取れる盤面でも 0 点を狙う場合があるため、実際の獲得点数が 0 点であった口

表 3 他 AI と  $n$  行探索手法の終盤力比較

AI 名	盤面数	点数差の試合平均			
		1 行	2 行	3 行	4 行
Ayumu	1556	-0.859	+0.027	+0.141	+0.196
jiritsu-kun	1525	-0.455	+0.123	+0.229	+0.292
CSACE	1657	-0.647	+0.028	+0.142	+0.210
GCCS	1705	-0.512	+0.214	+0.314	+0.375
SUC-Ks	2157	+0.534	+0.861	+0.973	+1.056
DIS	2853	+1.352	+1.669	+1.816	+1.931

表 4 他 AI と人間の知見による手法の終盤力比較

AI 名	盤面数	点数差の試合平均	
		4 行	5 行
Ayumu	1556	+0.104	+0.206
jiritsu-kun	1525	+0.204	+0.302
CSACE	1657	+0.119	+0.223
GCCS	1705	+0.287	+0.389
SUC-Ks	2157	+0.994	+1.077
DIS	2853	+1.772	+1.940

グに対しては比較の対象外とする．結果を表 3 および表 4 に示す．

表 3 と表 4 より，2 行の探索を行えば他の AI と同等以上の終盤力を持つことが分かる．また，探索行数が多くなるにつれて獲得点数が大きく向上していることが分かる．さらに，人間の知見による手法と 4 行探索手法について比較を行うと，同じ 4 行では探索手法の方が良い結果が出ているが，人間の知見による 5 行の手法と 4 行探索手法では，5 行の手法がわずかに良い結果となることが分かる． $n$  行の探索におけるシミュレーション回数は  $240n$  回，1 秒間に行えるシミュレーション回数は約 500 回であるため，5 行の探索であっても 2.5 秒程度の考慮時間で良いショットを導くことに成功している．

## 6. おわりに

本稿では，デジタルカーリングの最終ショットにおいて，有望なショットの探索時間を短縮する方法を提案し，提案手法は実用的な時間で高い終盤力を実現できることを示した．今後の展望としては，既存手法と本提案手法を組み合わせることにより，探索のさらなる高速化を図ることや，本提案手法を機械学習に応用することが考えられる．また，本提案手法を 15 投目以前に応用していく方法の検討も考えられる．

## 参考文献

- [1] 北清勇磨, 伊藤毅志. カーリングの戦略を支援するシステムの提案と構築. ゲームプログラミングワークショップ 2013 論文集, pp. 154–161, Nov. 2013.
- [2] 大渡勝己, 田中哲朗. カーリング AI に対するモンテカルロ木探索の適用. ゲームプログラミングワークショップ 2016 論文集, pp. 180–187, Oct. 2016.
- [3] 加藤修, 飯塚博幸, 山本雅人. ゲーム木探索に基づくカーリ

ング AI の開発. 精密工学会学術講演会講演論文集 2015 年度精密工学会春季大会, pp. 915–916. 公益社団法人 精密工学会, 2015.

- [4] 松井亮平, 保木邦仁. 畳込みニューラルネットワークを用いたデジタルカーリング AI 作成の試み. Technical Report 12, 電気通信大学, Feb. 2017.
- [5] 加藤修, 飯塚博幸, 山本雅人. デジタルカーリングにおけるゲーム木枝刈りの有効性検証. エンタテインメントコンピューティングシンポジウム 2015 論文集, pp. 412–417, Sep. 2015.