

デジタルカーリングサーバーの提案と紹介

北清勇磨^{†1} 岡田雷太^{†1} 伊藤毅志^{†1}

カーリングは氷上のチェスと言われるほど戦術性の高い競技であるが、実際のカーリングでは、気温やそれまでのプレーによる氷の状態の時間変化や選手の疲労など考慮しなくてはならないことが多く、戦術のみを切り出して議論することが少ない。そのためカーリングの戦術のデジタル的な研究が遅れている。

ここでは、物理シミュレーションを用いた理想的なデジタルカーリングコンディションを提案し、戦略のみを切り出して議論できる場を提供しようと考えている。

Introduction of the proposed Digital curling server

YUMA KITASEI^{†1} RAITA OKADA^{†1}
TAKESHI ITO^{†1}

Curling is a sport highly strategic properties even called chess on ice, but more that the curling practice, must be considered such as the fatigue of the player and the time change in the state of ice by the play of the previous temperature and it will be discussed by cutting only strategy is small. Research of digital tactics of curling is delayed for that.

In this case, we believe that the proposed digital curling ideal conditions for the use of physical simulation, attempts to provide a forum that can be discussed with cut out only strategy.

1. はじめに

カーリングは氷上のチェスと言われるほど戦術性の高い競技であるが、実際のカーリングでは、気温やそれまでのプレーによる氷の状態の時間変化や選手の疲労など考慮しなくてはならないことが多く、戦術のみを切り出して議論することが少なく、戦術のデジタル的な研究が遅れているという現状がある。

ここでは、物理シミュレーションを用いた理想的なデジタルカーリングコンディションを提案し、戦略のみを切り出して議論できる場を提供しようと考えている。

2. カーリングとは

2.1 カーリングのルール

カーリングは氷上で行われるウィンタースポーツである。4人ずつ2チームで行われ、約40メートル先に描かれたハウスと呼ばれる円を目標として各チームが交互に8回ずつストーンを氷上に滑らせる。すべてのストーンを投げ終わると得点をカウントして終了する。ストーンをティーと呼ばれる円の中心により近づけたチームが得点を得る。この一区切りをエンドと呼び、試合では10エンドを行い総得点で勝敗を競う。先攻後攻は1エンド目ではコイントスなどで決め、2エンド目以降は前エンドで得点したチームが不利な先攻となる。どちらも得点しなかった場合には先攻後攻を入れ替えずに次のエンドを行う。

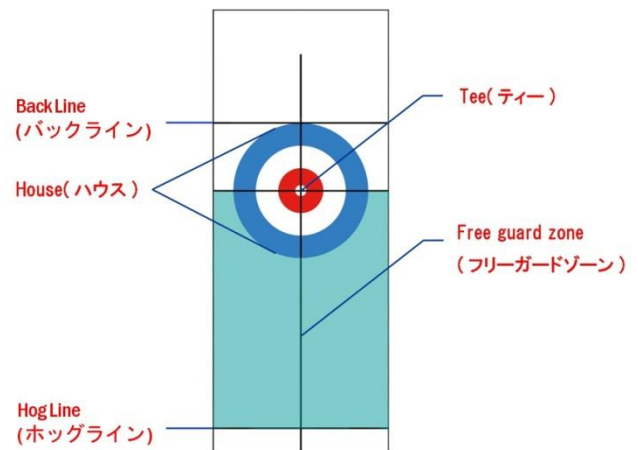


図1 カーリングのリンク

投げたストーンは、ホッグラインを越えないとアウトになり、バックラインを越えてもアウトになる。アウトになったストーンはプレーから外される。また、アウトにならないホッグラインからバックラインの間をプレーエリアと呼ぶ。

2.2 得点の計算

カーリングでは、両チームがすべてのストーン（1チーム8個、計16個）を投げ終わった時点で、ハウスの中心に1番近いチームだけが得点できる。つまり負けたチームは必ず0点となる。

また、このとき得られる得点は相手チームのストーンより中心に近いストーンすべてが得点になる。ただし、ハウ

^{†1} 電気通信大学
University of Electro-Communications.

スの外にあるストーンは得点にはならない。

2.3 カーリングのゲームの特徴と基本戦略

先攻と後攻ではどちらが有利だろうか。先攻は、後攻の最後の一投でハウスの中央へショットを決めるコースを完全に塞ぐことは難しい。一般に、ラストショットで後攻はハウスの中央を狙うショットを投げる事ができるので、一般に後攻が圧倒的に有利なゲームである。そのため、勝つためにはハウスの中心にストーンを置きに行くだけではなく、相手を邪魔するような位置にストーンを置く戦略が必要となってくる。また、先攻のセオリーとして、無理をせず「後攻チームに1点を取らせる」というものがある。カーリングでは、得点をあげると次のエンドで有利な後攻が回ってくるので、最少失点の1点を相手に取らせる代わりに次のエンドで後攻を取るという10エンドを通した戦略の先読みも必要となる。

このようにカーリングは一投ごとに深い戦略が必要とされるゲームのため、「氷上のチェス」と呼ばれている。

3. シミュレーター

局面を表す構造体とショットを表す2つのパラメータ(v , ω)を与えると次の局面を生成するシミュレーターを作成した。

3.1 シミュレーターの座標系

このシミュレーターではカーリングのリンクの左上を基準(0, 0)としており、横がX軸で縦がY軸になっている。

リンクやハウスのサイズは実際のカーリングのリンクと同じ大きさになっており、シミュレーターで扱う数字の単位はm(メートル)とした。

3.2 物理シミュレーション

シミュレーターにはBox2Dという物理エンジンを用いており、物理演算によって次の局面を生成する[a]。本システムでは厳密な物理シミュレーションは行わずに、戦略解析のための必要最低限の範囲で演算を行っている。

そのため、氷の状態はリンク内において全て一定とした。実際のカーリングの試合では、ショットを投げるたびにリンクの氷の状態は変化していく。そのため試合の前半と後半ではリンクの摩擦などの状態は大きく変わってくる。しかし、シミュレーションを単純化するためにこのシステムでは氷の摩擦力を一定として扱った。

また、スウィーピングについても考慮していない[b]。スウィーピングは、投げたストーンを狙った位置に誘導するためのものなので、ショットの不確実性に吸収できるため扱っていない。

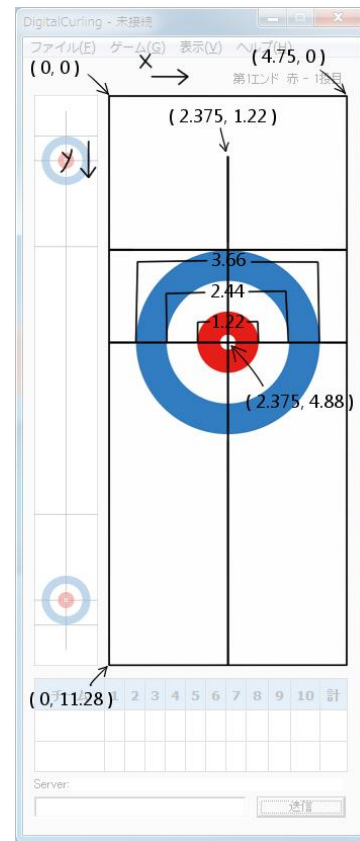


図2 シミュレーターの座標

3.3 ショットの表現

ストーンを投げることをショットと呼ぶ。ショットは初速度ベクトル v と回転方向 ω の2つのパラメータ(v , ω)で表す。初速度ベクトルは、ハックからのベクトルである[c]。

カーリングでは、まっすぐなショットというものは基本的には存在せず、必ず左右どちらかにカーブするものである。そしてそのカーブの軌跡はストーンの回転角速度によらないとされているため、システムで扱う回転角速度の大きさを一定とし、 ω が正の値だと右カーブ、負の値だと左カーブとした。

3.4 ショットの不確実性と乱数

不確実性については、受け取ったショットのパラメーター(v , ω)に乱数を加えシミュレーションを行っている。乱数には正規乱数を利用しており、ショットの強さベクトル v のx方向とy方向それぞれに対し乱数の値を加えている。

これはスウィーピングによる影響やショットミスなどを考慮したものである。

3.5 カーブについて

ベストなコンディションのリンクでは、8フット円とテールラインの交差する点を狙ってショットを行うとストー

a) Box2D : <http://box2d.org/>

b) スウィーピングとは、投げたストーンの進路前をブラシで掃く行為。

c) ハックとは、ストーンを投げる時に使うけり台のこと。

ンがティーへ行くことが知られている[d][e]. 本システムではそれを基準にカーブの曲がり具合を定めている.

ストーンの回転によるショットの影響については1フレームごとに演算を行っており, ストーンの進行方向に対し常に垂直方向に力が加わるようにしてある.

ストーンの世界速度ベクトルを求める具体的な計算手順は以下の通りである.

手順1. n フレーム時のストーンの世界速度ベクトルを v とし, ストーンとリンクとの間の動摩擦係数を a , ストーンの回転により発生する進行方向に対し垂直な力を b とする.

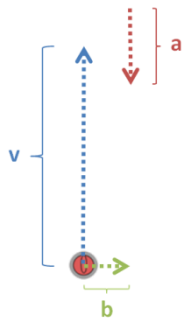


図3 速度ベクトルの計算手順1.

手順2. 速度の方程式 $v = v_0 + at$ より, $n+1$ フレーム時のストーンの世界速度ベクトルの大きさは $|v - a|$ である.

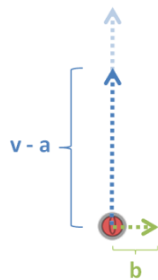


図4 速度ベクトルの計算手順2.

手順3. そこに回転による垂直方向の力 b を加える.

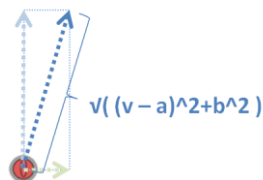


図5 速度ベクトルの計算手順3.

手順4. このままでは $n+1$ フレーム目のストーンの世界速度

ベクトルの大きさが $|\sqrt{(v-a)^2 + b^2}|$ になってしまうので, 速度ベクトルの大きさが $|v-a|$ になるように調整する.

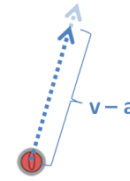


図6 速度ベクトルの計算手順4.

以上の手順によって次のフレームにおけるストーンの世界速度ベクトルを求めている.

この計算方法では, ストーンの回転による垂直方向の力 b は定数としているので, ストーンの世界速度が落ちてきた場合に相対的に b の影響が大きくなっていくことがわかれると思う. そのため実際のカーリングのショットと同様, 前半に比べ後半に行くほど曲がるようになっている.

3.6 シミュレーターが提供している関数

シミュレーターを公開すると同時に, このシステムを使い思考エンジンを作る際に必要となるいくつかの関数を提供している.

(1) Simulation

現在のゲームの局面とショットの強さベクトルを与えるのと次の局面を返してくれる関数.

(2) CreateShot

この関数はドローショットを生成したい場合に用いる関数で, ドローショットを止めたい位置を与えると, ショットの強さベクトルに変換して返してくれる関数[f].

(3) CreateHitShot

この関数はヒットショットを生成したい場合に用いる関数で, ヒットショットを通過させたい座標と, ショットの強さを与えると, ショットの強さベクトルに変換して返してくれる関数[g].

開発者は思考エンジンを作成する際には, これら3つの関数を利用しショットを生成することになる.

3.7 ショットの強さについて

カーリングで使われるショットの強さを参考にして, 本システムでは以下の図のようにショットの強さを定義した.

ショットの強さは0~16までの範囲で扱っており, 図では強さ11までしか表記していないが, 11以降は10~11の間隔幅と同じ間隔で16まで設定されている.

またショットの強さはfloat型として数値を扱っているため小数によるシステムへの指定も有効である.

d) 8フット円とは, 直径が8フット(約2.44m)の円のこと. ハウスの白い円を指す.

e) ティーラインとは, ハウスの中心をセンターラインと垂直に横切るライン.

f) ドローショットとは, ハウス内にストーンを止めるショット. ここでは狙った位置に止めるショットの意味で用いている.

g) ヒットショットとは, 投げたストーンを既にあるストーンにぶつけるショット.

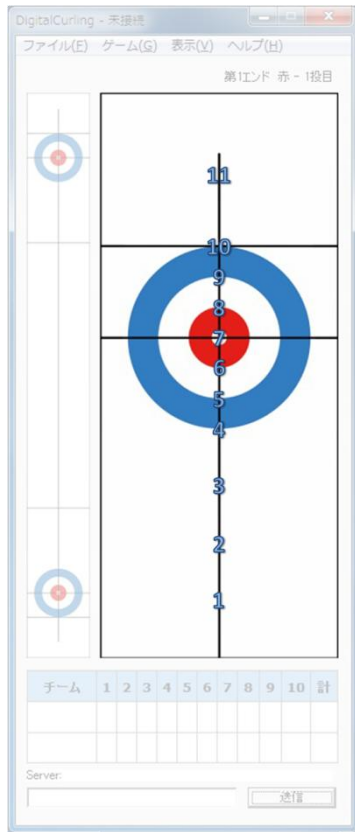


図 7 ショットの強さ

4. サーバー

上記で説明したシミュレーターを持つデジタルカーリングサーバーをネットワーク上に置く。サーバーではデジタルカーリングの対戦を管理し、プレイヤーから送られてきたショットのデータから次の局面を生成し、各プレイヤーに新しく生成された次のゲームの状態を返す。

このサーバーを利用することによりデジタルカーリングの思考エンジン同士がいつでも対戦できる環境を整えておく。また、このデジタルカーリングサーバーに接続することによりコンピュータ同士の対戦だけではなく、コンピュータと人間、人間同士のデジタルカーリングの対戦もできる環境を整えている。

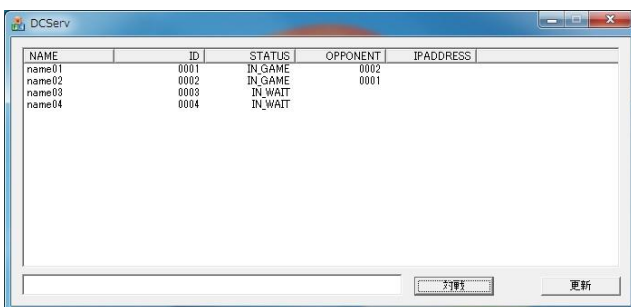


図 8 サーバー

5. クライアント

クライアントはこちらで用意している通信プロトコルに準拠していれば誰でも作成することができる。

また、こちらでもクライアントを提供している。そのクライアントを利用することで、思考エンジンのみを作成すればよく、簡単にサーバーと接続し対戦のできる AI を作ることができるようになっている。

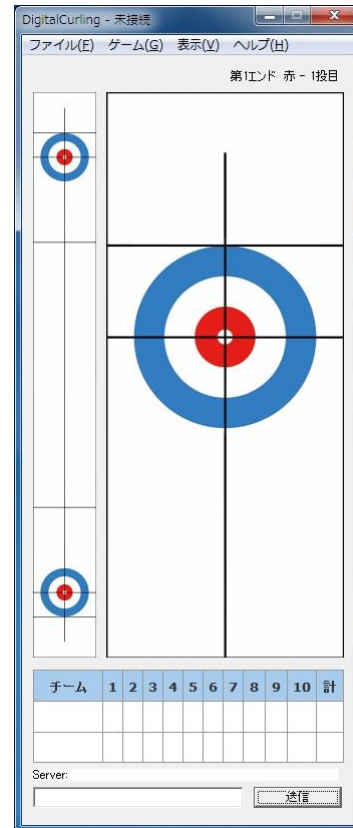


図 9 クライアント

またこのクライアントでは、サーバーへの接続をサポートしているだけではなく描画処理も行っており、プログラミング初心者の方でも思考エンジンの開発ができるようになっている。

6. 思考エンジン

思考エンジンは、ストーン的位置情報や現在のストーンの投数や得点といった情報から次にどこに投げるのかというショットの情報を返すものを作るだけで作成することができる。また、こちらで提供しているクライアントを用いる場合には、標準入出力を用いてクライアントにメッセージを送ることによってサーバーと簡単に通信が行えるようになっています[h].

h) デジタルカーリング : <http://minerva.cs.ucc.ac.jp/curling/wiki.cgi>

7. 通信プロトコル

今回このシステムを作成するにあたって、新たに通信プロトコルを作成した。詳しいプロトコルの内容については本研究室で管理しているホームページ上で公開しているが、基本的には POSITION コマンドと SETSTATE コマンドで局面的情報が送られ、その局面に対し最善と考えられるショットを BESTSHOT コマンドで返すだけの簡単な仕様となっている。

サーバーとクライアント間での通信ログの一例を以下に示す。

7.1 通信ログの例

Server : CONNECTED

↑ 正しく接続できたことの確認メッセージ

Client : LOGIN ID0001 1234 Player01

↑ 左から順に ユーザーID パスワード 表示名

Server : LOGIN OK

↑ 正しくログインできたことの確認メッセージ

:

Server : ISREADY

↑ 対戦開始前の準備ができたかの確認

Client : READYOK

↑ 対戦の準備ができた合図

Server : NEWGAME Player01 Player02

↑ ゲーム開始の合図

(先手後手両方のクライアントから READYOK が返ってきたらゲームを開始する)

Server : POSITION 0.000000 0.000000 0.000000 0.000000
0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
0.000000 0.000000 0.000000 0.000000 ←1 投目から 16 投分の
ストーンの位置座標 ($x_1 y_1 x_2 y_2 \dots x_{16} y_{16}$)

Server : SETSTATE 0 1 10 0

↑ ゲームの情報(左から順に 現在何投まで投げ終わったか 現在何エンド目か 最終エンドは何エンドか 手番はどちらか (0: 初回先手だった方の手番 1: 初回後手だった方の手番))

Server : GO

↑ クライアントに思考を開始してもらう合図 (手番のクライアントのみに送るメッセージ)

Client : BESTSHOT -0. 105485 -2. 846352 0

↑ 左から順に ショットの強さの x ベクトル ショットの強さの y ベクトル 回転方向 (0 なら右, 1 なら左カーブ)

Server : RUNSHOT -0. 106634 -2. 848441 0

↑ サーバーにより乱数に加えられ、実際に局面を進める

上で使われたショットのデータ

Server : POSITION 2. 237748 7. 471060 0. 000000 0. 000000
0. 000000 0. 000000 0. 000000 0. 000000 0. 000000 0. 000000
0. 000000 0. 000000 0. 000000 0. 000000 0. 000000 0. 000000
0. 000000 0. 000000 0. 000000 0. 000000 0. 000000 0. 000000
0. 000000 0. 000000 0. 000000 0. 000000 0. 000000 0. 000000
0. 000000 0. 000000 0. 000000 0. 000000 0. 000000 0. 000000
0. 000000 0. 000000 0. 000000 0. 000000 0. 000000

Server : SETSTATE 1 1 10 1

:

以降ゲーム終了まで繰り返す

:

Server : SCORE -2

↑ エンドごとの得点

Server : GAMEOVER LOSE

↑ ゲーム終了の通知と結果

思考を開始する GO コマンドの直前にプレイヤーの残り時間を伝えるコマンドは追加予定です。

8. おわりに

この機会にカーリングというゲームに興味を持っていた、この試みに共感していただいた方にはぜひ思考エンジンのプログラムの作成にチャレンジしていただきたい。またデジタルカーリングの大会を開催したいと考えているので、大会開催時にはぜひ参加していただきたい。

このサーバーや通信プロトコルを利用していただき、デジタル上でカーリングを議論し、最善戦略を求める研究が進むことを期待している。また、ここで得られた結果が実際のカーリングに活かされ、日本のカーリングの戦術レベルの向上につながれば幸甚である。

謝辞

アソボウズ

謝辞 本システムを開発するにあたり、カーリングに関する様々なご助言をいただいた株式会社絆の片山宗臣様、石山龍彦様に御礼申し上げます。ご協力ありがとうございました。

参考文献

1) デジタルカーリング

<http://minerva.cs.uec.ac.jp/curling/wiki.cgi>