WCSC28 elmoアピール文書

1. elmoについて
   elmoは主に評価関数に改良を加えたソフトです。
   Apery/やねうら王を主に利用しています。
   ＜以前の内容＞
      WCSC26: 自己対局の勝率に基づいて評価値生成
              →とても弱かった
      WCSC27: 自己対局時の勝敗と深く読んだ時の評価値を用いて評価値を更新、
              大規模(50億局面)学習
              →優勝：やねうら王/Apery等で採用！

2. 評価関数
   昨今のコンピュータ将棋は500点程度差がつくと逆転するのは難しいので、
   序盤を含めた評価値の近い「競っている局面」を手厚く学習する方向で進めています。
   学習部分の変更も色々試しているのですが、良くも悪くもあまり変わらず悩ましいです。

3. 定跡生成
   定跡抜けたら負け確定でしたとかツラいじゃないですか。
   勝敗への影響が大きい割に今までなおざりにしていたので、
   評価関数と同じような方法で作るようにしました。
   (自己対局→勝敗＋評価値を利用して評価)

4. 利用ライブラリ
   やねうら王:
      対局時と定跡生成で利用しています。定跡に互換性が無いので少し手を入れています
      採用理由：探索部分が優秀なため。既存拡張機能の流用
   Apery:
      評価関数生成と定跡生成で利用しています。
      また、評価関数もAperyのものをベースに更新しています。
      採用理由：評価関数(および学習部)が優秀ため。既存拡張機能の流用
   Qhapaq/tanuki-:
      評価関数を混ぜて使おうかと思っています(多分使います)。
      採用理由：評価関数が強いため

5. その他
   ビッグウェーブに乗れませんでした

コンピュータ将棋ソフト
「大合神クジラちゃん」

第 28 回世界コンピュータ将棋選手権　アピール文書

2018/04/11　鈴木雅博、大賀一貴

1. 大合神クジラちゃんの紹介

マスタ／スレーブの構成を取り、マスタがスレーブを管理しながらクラスタリング探索を行います。スレーブを動かすのは主にニコニコ生放送や YouTube 視聴者のパソコンを考えています。前回の選手権ではパソコン４００台以上のクラスタで動かしました。

クラスタの基本的な手法は GPS 将棋を参考にしています。マスタが各指し手をスレーブに展開することでより読みを深くしています。また、冗長性を持たせるため同じ手を 2 つのスレーブに探索させています。

2. 独自の工夫

独自の工夫として、探索を探索ツリーの末端ノードだけでなくすべてのノードで行うことで、スレーブの性能差を気にすることなく安定した棋力向上を可能にしました。　過去のシステムでは末端でないノードでは探索させず、末端ノードのみに指し手を展開し、展開できなかった手はそれ以外の手を探索する「その他ノード」に探索させるようにしていました。しかしながらその手法では、ある指し手とそれ以外の指し手の探索量が大幅に違った場合、なにが最善手であるか判断するのが極めて難しいという問題がありました。具体的には、評価値が 1000 点・探索深さ 14 という手と、評価値が 900 点・探索深さ 20 という手があった場合、どちらが良い手なのか判断できないという問題がありました。

現在のシステムでは、末端以外の親ノードでも探索を行うようにしこの不安定性を少し減らすことに成功しました。例えば上記の例であれば、末端ノードの探索結果（指し手 A：評価値が 1000 点・探索深さ 14、指し手 B：評価値が 900 点・探索深さ 20）に加え、親ノードの探索結果が加わります。例えばその結果が評価値 950・探索深さ 18 で指し手が A だったとすると、探索深さが深くなった場合でも指し手 A が有力ということがわかります。

またこのシステムはビンゴマスターさんのアドバイスをもとに作られたものなので、ビンゴマスターさんに深く感謝申し上げます。

ディープラーニングによる指し手探索もやろうと考えていますが、現状（4/11）全く手つかずです。もし間に合いそうであればクラスタの指し手の初期探索部分を DL で行いたいと考えています。

3. その他

- 探索部分に YaneuraOu、評価関数は既存のものを追加学習したバージョンを使用しております。
- 非公開協力者として、まふさんに定跡ファイルを作って頂く予定です。
- ディープラーニングによる指し手探索もやろうと考えていますが、現状（4/11）全く手つかずです。もし間に合いそうであれば指し手の初期探索部分を DL で行いたいと考えています。

4. ライブラリの選定理由について

- YaneuraOu
  - クラスタのスレーブ部分の基礎として使用。非常に強く探索部分も優秀なため使わせて頂いています。
- Apery
  - クラスタのマスター部分の将棋ライブラリとして使用。探索は行わず合法手の確認や指し手生成に使っています。YaneuraOu に変えてもいいのですが、昔から使っているので今も使用しています。
- 人造棋士 18 号
  - 関連ツールを使用させて頂く可能性があるため登録しています。
- dlshogi
  - マスターの初期探索部分をディープラーニングで行おうかと考えているので、念のため登録しております。ただし現在は全く手がついておりません。

5. 過去の戦績

- 第 23 回コンピュータ将棋選手権 27 位（独創賞を獲得）
- 第 24 回コンピュータ将棋選手権 16 位
- 第 25 回コンピュータ将棋選手権 19 位
- 第 27 回コンピュータ将棋選手権 4 位

6. 完全オリジナルのＧＵＩもあります。

- コンピュータ将棋を動かすための GUI も自作しています。
- クジラちゃんを擬人化し、GUI 内に取り込みました。
- URL: http://garnet-alice.net/programs/whalewatcher/
- ニコニコミュニティ： http://com.nicovideo.jp/community/co516151

コンピュータ将棋の概念を打ち砕き
進化させるべく三駒関係を封印する

高速な差分計算を特徴とし
三駒関係と同等の NPS を実現する
ディープラーニング評価関数を搭載

the end of genesis
T.N.K.evolution turbo type D

ザイオソフト
コンピュータ将棋サークル

野田久順　岡部淳　鈴木崇啓
那須悠　河野明男

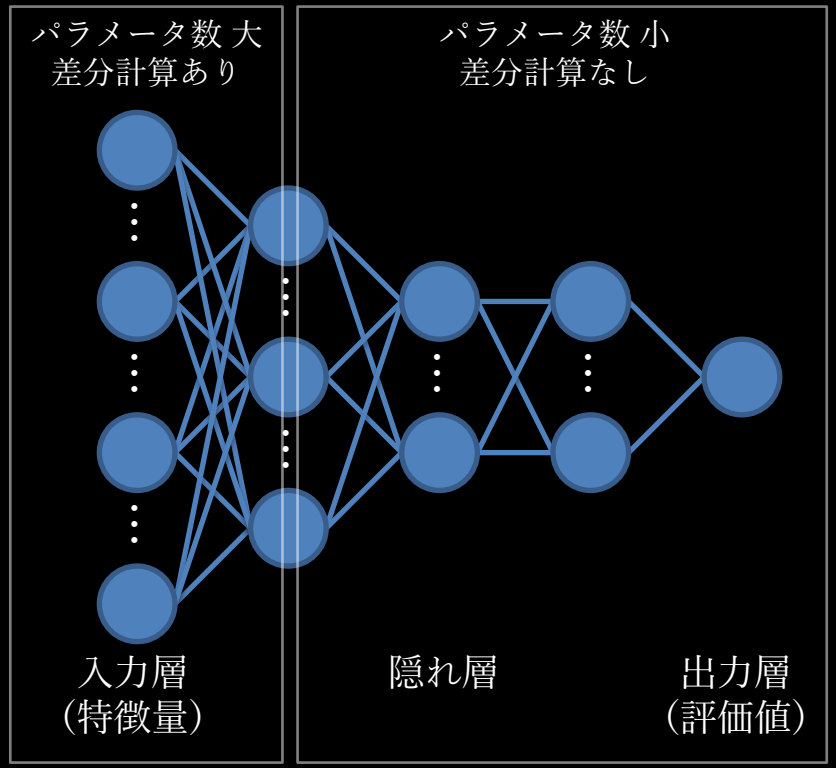
ベンチマーク

- 測定環境
  - やねうら王ベンチマーク
  - CPU: Core i7 6700K
  - 置換表サイズ: 16GB
  - スレッド数: 8
  - トーナメント版
  - NPSはやねうら王ベンチマークに収録されている3局面のNPSの平均値

CPUによる演算

- T.N.K. のディープラーニング評価関数はGPUを使わず、CPUで１局面ずつ評価します
  - αβ探索ベースの探索ルーチンにそのまま組み込むことができます
- パラメータを整数化し、SIMD演算で高速化しています

差分計算



- KPに相当する特徴量を入力とする全結合ニューラルネットワークです
- 入力のアフィン変換を差分計算で高速化しています
  - 二駒関係の差分計算をベクトルに拡張して適用しています

使用ライブラリ

- やねうら王
  - 用途
    - エンジンの
      基礎部分として使用
  - 選定理由
    - レーティングの高さ
    - 改造のしやすさ
- Apery
  - 用途
    - 学習データ生成時の
      評価関数として使用
  - 選定理由
    - レーティングの高さ

乞うご期待…

高速に差分計算可能なニューラルネットワーク型将棋評価関数

那須 悠 †

† ザイオソフト コンピュータ将棋サークル

2018 年 4 月 28 日

概要

現在のコンピュータ将棋ソフトの多くで使用されている評価関数である三駒関係は，高速な計算が可能であるが，非線形な評価を行うことができない．本稿では，CPU で高速に動作するニューラルネットワーク評価関数，「NNUE 評価関数」(Efficiently Updatable Neural-Network-based evaluation functions) を提案する．NNUE 評価関数は，CPU での演算に最適化した設計と，差分計算を始めとする高速化技術により，三駒関係と同等の実行速度で動作する非線形評価関数である．NNUE 評価関数を使用する初めての将棋ソフト『the end of genesis T.N.K.evolution turbo type D』は，第 28 回世界コンピュータ将棋選手権に参加を予定している．

NNUE: Efficiently Updatable Neural-Network-based Evaluation Functions for Computer Shogi

Yu Nasu†

†*Ziosoft Computer Shogi Club*

April 28, 2018

Abstract

Most of the strongest shogi programs nowadays employ a linear evaluation function, which is computationally efficient but lacks nonlinear modeling capability. This report presents a new class of neural-network-based nonlinear evaluation functions for computer shogi, called NNUE (Efficiently Updatable Neural-Network-based evaluation functions). NNUE evaluation functions are designed to run efficiently on CPU using various acceleration techniques, including incremental computation. The first shogi program with a NNUE evaluation function, *the end of genesis T.N.K.evolution turbo type D*, will be unveiled at the 28th World Computer Shogi Championship.

0. まえがき

本稿は，第 28 回世界コンピュータ将棋選手権 [1] に参加を予定しているソフト『the end of genesis T.N.K.evolution turbo type D』のアピール文書の一部である．同ソフトの評価関数に関する技術を論文風の体裁で説明するが，実験は掲載しない．同ソフトの具体的な棋力にも本稿では言及しないので，選手権の結果を参照されたい．

1. はじめに

コンピュータ将棋ソフトの指し手の決定は，主として形勢判断に相当する「評価関数」と，読みに相当する「探索」によって行われる．高い棋力を実現するためには，評価値の精度が高く，高速に計算できる評価関数を用いることが望ましい．評価値の精度が高ければ正確な形勢判断ができ，計算が高速であればより深く読むことができるためである．

2018 年 3 月現在，コンピュータ将棋ソフトで使用される最も代表的な評価関数の方式は，「三駒関係」と呼

ばれる線形モデルである．機械学習によって駒 3 つの位置関係に重みを付けておき，局面上に出現する位置関係に付けられた重みの総和を評価値とする．2003 年に初めてアイデアが示され [2]，2009 年に玉を含む組み合わせに限定して実装した『Bonanza』 [3] のソースコードが公開されて以降，広く用いられるようになった．2014 年に『NineDayFever』 [4] で導入された手番評価など，いくつかの拡張を経て，トップクラスの棋力を持つ将棋ソフトのほとんどで現在も採用されている．

三駒関係の優位性は，膨大な数のパラメータによって表現される評価関数でありながら，高速に評価値を算出できる点にある．ある局面から 1 手指した先の局面の評価値を求めるとき，移動した駒に関する重みを元の局面の評価値に足し引きする差分計算によって，全体を計算した場合と同じ結果を，遥かに効率的に求めることができる．差分計算の容易さは，評価関数に線形モデルを採用することによってもたらされる大きな利点である．

しかし，線形モデルであることは，評価関数の表現能力の足枷にもなっている．特定の駒の位置関係が，ある状況においては有利に働くが，別の状況では無駄になる，といった非線形な評価を，三駒関係では直接表現することができない．線形モデルのまま表現能力を高めるためには，評価項目を変える必要があるが， 2018 年 3 月現在，三駒関係より優れた線形モデルは知られていない．

より高い表現能力を持つ評価関数の実現を狙ったアプローチとして，非線形モデル，特にニューラルネットワークの利用も試みられている．先駆けとなったのは『習甦』 [5] で，自玉および相手玉の安全度によって駒の価値を重み付けした，非線形な評価関数を特徴とする将棋ソフトである．近年では，コンピュータ囲碁において，大規模な CNN (Convolutional Neural Networks) による評価関数を用いたソフトが成功を収めた [6] ことから，コンピュータ将棋でも CNN を利用する試みが行われている．しかし， CNN による評価関数を活用するためには，線形モデルよりも大きな計算リソースが必要となることが課題である．CNN を利用する将棋ソフトのほとんどは，並列演算を得意とする GPU を使用し，複数の局面をまとめて評価するバッチ処理を行って高速化している．それにもかかわらず， 2018 年 3 月現在，ニューラルネットワークに最適化された特殊なハードウェアを使用する『AlphaZero』 [7] を除いて，トップクラスの将棋ソフトに比肩する棋力を実現した例は報告されていない．

本稿では， CPU で高速に動作するニューラルネットワーク評価関数を提案する．CPU での演算に最適化した設計と，差分計算を始めとする高速化技術により，複数の隠れ層を持つニューラルネットワーク評価関数を，三駒関係と同等の実行速度で動作させることができる．以降，この方式の評価関数を「NNUE 評価関数」 (Efficiently Updatable Neural-Network-based evaluation functions) と呼ぶことにする．

2. NNUE 評価関数

NNUE 評価関数は， GPU もバッチ処理も用いず，CPU で 1 局面ずつ評価するニューラルネットワーク評価関数である．三駒関係と同様， 1 局面を評価するまでのレイテンシが小さく，枝刈りを伴うミニマックス系の探索手法と組み合わせやすい．

『the end of genesis T.N.K.evolution turbo type D』に搭載する NNUE 評価関数の模式図を図 1 に示す．CPU で高速に計算するために設計した構造であり，『AlphaZero』などで採用されているニューラルネットワークの構造とは大きく異なる．

以下では， NNUE 評価関数の設計と高速化技術について述べる．

2.1 全結合ニューラルネットワーク

将棋盤の二次元構造を利用する CNN ではなく，全結合ニューラルネットワークを使用する．主な利点は，メモリアクセスパターンが単純で高速化しやすいことと，後述する差分計算が容易に実現できることの 2 点である．

メモリアクセスパターンは，三駒関係より処理効率が高い評価関数を設計するために重要な要素である．三駒関係では，評価値を算出するためにかかる時間の大半を，メモリへのランダムアクセスが占めている．効率的なメモリアクセスパターンを用いれば，同じ時間でより複雑な計算が可能になる．

全結合ニューラルネットワークによる評価値の計算を次式に示す．入力となる特徴ベクトルを $\boldsymbol{x}$，隠れ層の数を L，層 l の重み行列を $\boldsymbol{W}_l$，層 l のバイアスベクトルを $\boldsymbol{b}_l$，活性化関数を σ とすると，評価値 y は

$$[y]_{1\times 1} = \boldsymbol{z}_{L+1} \tag{1}$$

$$\boldsymbol{z}_l = \boldsymbol{b}_l + \boldsymbol{W}_l \boldsymbol{a}_{l-1} \tag{2}$$

$$\boldsymbol{a}_l = \begin{cases} \sigma\left(\boldsymbol{z}_l\right) & (\text{if } l > 0) \\ \boldsymbol{x} & (\text{if } l = 0) \end{cases} \tag{3}$$

と求められる．

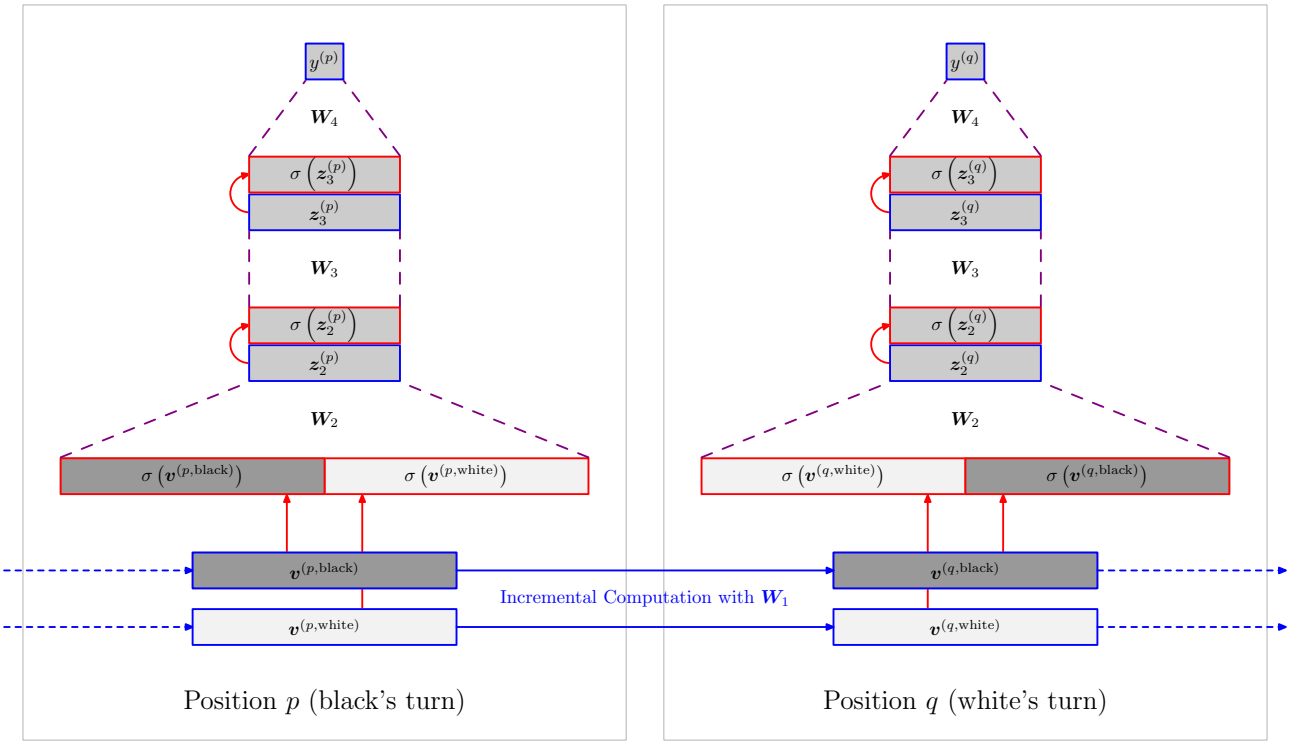


図 1　『the end of genesis T.N.K.evolution turbo type D』の評価関数．局面 p から 1 手指した先の局面 q について評価値を求めるとき，処理の一部で差分計算を利用する．評価には手番が考慮されている．

2.2　活性化関数

活性化関数には，層 l のユニット数を N_l として，次式で表される clipped ReLU を用いる．

$$\begin{aligned}\sigma\left(\boldsymbol{z}_l\right) &= \sigma\left(\begin{bmatrix} z_{l,1} \\ z_{l,2} \\ \vdots \\ z_{l,N_l} \end{bmatrix}_{N_l \times 1}\right) \\ &= \begin{bmatrix} \sigma(z_{l,1}) \\ \sigma(z_{l,2}) \\ \vdots \\ \sigma(z_{l,N_l}) \end{bmatrix}_{N_l \times 1}\end{aligned} \tag{4}$$

$$\sigma\left(z_{l,i}\right) = \begin{cases} 0 & (\text{if } z_{l,i} \leq 0) \\ z_{l,i} & (\text{if } 0 < z_{l,i} < 1) \\ 1 & (\text{if } z_{l,i} \geq 1) \end{cases} \tag{5}$$

値域が有限であるため整数化に適しており，後述する SIMD 演算によって効率的に計算できることが利点である．

2.3　特徴量

特徴ベクトルには，原則として，各要素が 0 または 1 の値を取る二値ベクトルを使用する．この制約も差分計算を容易にするためである．高速に差分計算を行うためには，1 手指す前後の局面で値が変化する要素の数が少ないことが望ましい．

このような性質を満たす特徴量の例としては，自玉または敵玉と，玉以外の駒 1 つとの位置関係を表す KP (King-Piece) が挙げられる．『the end of genesis T.N.K.evolution turbo type D』で実際に採用している特徴量は，KP をベースとして改変したものである．詳細は後述する．

2.4　アフィン変換とメモリレイアウト

式 (2) のようなアフィン変換を行うためには，通常，行列の行とベクトルの内積を計算するのが効率的である．$\boldsymbol{W}_l$ の i 行目を $\boldsymbol{W}_l(i,:)$ として，

$$\boldsymbol{z}_l = \begin{bmatrix} z_{l,1} \\ z_{l,2} \\ \vdots \\ z_{l,N_l} \end{bmatrix}_{N_l \times 1} \tag{6}$$

$$z_{l,i} = b_{l,i} + \boldsymbol{W}_l(i,:)\boldsymbol{a}_{l-1} \tag{7}$$

と計算する．行列の行の要素を連続して参照するメモリアクセスパターンであるから，メモリレイアウトは row-major とする．

しかし，ベクトル $\boldsymbol{a}_{l-1}$ がスパースである（要素の大

部分が 0 である）場合は，行列の一部の列だけを用いて，次式のように計算する方が高速になる．$\boldsymbol{W}_l$ の j 列目を $\boldsymbol{W}_l(:,j)$ として，

$$\boldsymbol{z}_l = \boldsymbol{b}_l + \sum_{j \in \{j | a_{l-1,j} \neq 0\}} a_{l-1,j} \boldsymbol{W}_l(:,j) \tag{8}$$

と計算する．この場合は，行列の列の要素を連続して参照するメモリアクセスパターンになり，メモリレイアウトを column-major とする方が効率が良い．

NNUE 評価関数では，特徴ベクトル $\boldsymbol{x}$ のアフィン変換

$$\boldsymbol{z}_1 = \boldsymbol{v} \tag{9}$$

$$\boldsymbol{v} = \boldsymbol{b}_1 + \boldsymbol{W}_1 \boldsymbol{x} \tag{10}$$

を，式 (8) の方法によって計算する．前述の通り，$\boldsymbol{x}$ は二値ベクトルである．$\boldsymbol{x}$ がスパースであれば，式 (8) によってアフィン変換を高速に計算できる[*1]．さらに，二値ベクトルであることを利用すると，式 (8) において $a_{l-1,j} = x_j \in \{0,1\}$ であるので，

$$\boldsymbol{v} = \boldsymbol{b}_1 + \sum_{j \in \{j | x_j = 1\}} \boldsymbol{W}_1(:,j) \tag{11}$$

と単純化できる．

2.5 差分計算

差分計算は，ある局面から 1 手指した先の局面の評価値を計算する際に，元の局面との差異が少ないことを利用して高速化する方法である．NNUE 評価関数では，局面 q の特徴ベクトル $\boldsymbol{x}^{(q)}$ をアフィン変換した結果のベクトル

$$\boldsymbol{v}^{(q)} = \boldsymbol{b}_1 + \boldsymbol{W}_1 \boldsymbol{x}^{(q)} \tag{12}$$

を，1 手前の局面 p について計算済みの $\boldsymbol{v}^{(p)}$ を利用して差分計算する．具体的には，次式によって $\boldsymbol{v}^{(q)}$ を求める．

$$\begin{aligned} \boldsymbol{v}^{(q)} = \boldsymbol{v}^{(p)} &- \sum_{j \in \left\{ k \middle| x_k^{(p)}=1 \wedge x_k^{(q)}=0 \right\}} \boldsymbol{W}_1(:,j) \\ &+ \sum_{j \in \left\{ k \middle| x_k^{(p)}=0 \wedge x_k^{(q)}=1 \right\}} \boldsymbol{W}_1(:,j) \end{aligned} \tag{13}$$

これは，線形モデルの差分計算をベクトルに拡張した式であると解釈することができる．

[*1] 実際には差分計算を行うので，1 手指す前後の特徴ベクトルの変化（ハミング距離）が小さいことが重要であり，スパースである必要はない．

差分計算を利用するのは特徴ベクトルのアフィン変換だけで，それ以外の部分では利用しない．

2.6 手番評価

将棋においては，駒の配置が同じであっても，手番がどちらにあるかによって形勢は異なるので，評価関数も手番を考慮する方が高精度になる．三駒関係では，線形性を利用して評価値を 2 つの項に分解し，手番に依存しない項と手番に依存する項の和で表現している [4]．

NNUE 評価関数では，常に手番側の視点から見た局面を評価することによって手番評価を実現する．まず，単純な方法について説明する．局面 p の手番側プレイヤーを active(p)，プレイヤー c から見た局面 p の特徴ベクトルを $\boldsymbol{x}^{(p,c)}$ として，

$$\boldsymbol{z}_1^{(p)} = \boldsymbol{b}_1 + \boldsymbol{W}_1 \boldsymbol{x}^{(p,\mathrm{active}(p))} \tag{14}$$

とすると，手番側の視点から見た局面の評価が実現できる．

手番がどちらにあるかによって特徴ベクトルが変わるため，差分計算は，先手（black）および後手（white）の両方の視点について行う必要がある．1 手指すごとに，$c \in \{\mathrm{black}, \mathrm{white}\}$ に対して，それぞれ

$$\boldsymbol{v}^{(p,c)} = \boldsymbol{b}_1 + \boldsymbol{W}_1 \boldsymbol{x}^{(p,c)} \tag{15}$$

を差分計算しておく．評価値の計算には手番側を使用し，

$$\boldsymbol{z}_1^{(p)} = \boldsymbol{v}^{(p,\mathrm{active}(p))} \tag{16}$$

とする．

次に，この方法を拡張して，非手番側のベクトルも評価値の計算に活用する方法を説明する．局面 p の非手番側のプレイヤーを opponent(p) として，式 (16) の代わりに次式を用いる．

$$\begin{aligned} \boldsymbol{z}_1^{(p)} &= \begin{bmatrix} \boldsymbol{v}^{(p,\mathrm{active}(p))} \\ \boldsymbol{v}^{(p,\mathrm{opponent}(p))} \end{bmatrix}_{2N_1 \times 1} \\ &= \begin{cases} \begin{bmatrix} \boldsymbol{v}^{(p,\mathrm{black})} \\ \boldsymbol{v}^{(p,\mathrm{white})} \end{bmatrix}_{2N_1 \times 1} & (\text{if active}(p) = \text{black}) \\ \begin{bmatrix} \boldsymbol{v}^{(p,\mathrm{white})} \\ \boldsymbol{v}^{(p,\mathrm{black})} \end{bmatrix}_{2N_1 \times 1} & (\text{if active}(p) = \text{white}) \end{cases} \end{aligned} \tag{17}$$

すなわち，手番側と非手番側の特徴ベクトルをそれぞれアフィン変換し，結合したベクトルを $\boldsymbol{z}_1^{(p)}$ とする．アフィン変換のパラメータ $\boldsymbol{b}_1, \boldsymbol{W}_1$ は共有されるので，こ

の計算は手番方向の畳み込みであると解釈できる．

図 1 に示した『the end of genesis T.N.K.evolution turbo type D』の評価関数は，式 (17) の方式を採用している．

2.7 HalfKP 特徴量

前述の通り，『the end of genesis T.N.K.evolution turbo type D』では，二駒関係の一種である KP を改変した特徴量を採用している．KP が「自玉または敵玉と，玉以外の駒 1 つとの位置関係」を表す特徴量であるのに対して，敵玉を含む位置関係を使用せず，自玉を含む位置関係だけを使用する．以下では，この特徴量を「HalfKP」と呼ぶことにする．

HalfKP 特徴量は，前項で説明した，手番側と非手番側の特徴ベクトルを両方とも使用する手法と組み合わせるための特徴量である．HalfKP 特徴量を使用すると，式 (17) において， $\boldsymbol{v}^{(p,\mathrm{active}(p))}$ は「手番側から見た，手番側の玉に関する KP」に相当する情報を持ち，$\boldsymbol{v}^{(p,\mathrm{opponent}(p))}$ は「非手番側から見た，非手番側の玉に関する KP」に相当する情報を持つことになる．従って，特徴量そのものには敵玉の位置に関する情報を含まないが，手番側から見た特徴量と，非手番側から見た特徴量の両方を使用することによって，局面全体の情報を表現することができる．通常の KP を特徴量として使用する場合と比較して，同等の情報を表現するために必要な計算コストが小さくなることが利点である．

2.8 整数 SIMD 演算

SIMD (Single Instruction Multiple Data) 演算は，同一の処理を複数のデータに同時に適用する演算である．NNUE 評価関数は，各部分において整数の SIMD 演算を活用して高速に計算できるように設計されている．

特徴ベクトルのアフィン変換は，差分計算を行う場合も行わない場合も，ベクトルの加減算で求められる．重み行列 $\boldsymbol{W}_1$ を 16-bit の整数に変換して保持し， 16-bit の符号付き整数ベクトルの加減算を行う命令（AVX2 では `VPADDW`, `VPSUBW`）で計算する．

特徴ベクトル以外のアフィン変換では，直前の層のアクティベーション $\boldsymbol{a}_{l-1}$ と重み行列 $\boldsymbol{W}_l$ を 8-bit で表現し， 8-bit 整数ベクトル 2 つの積和を求める命令（`VPMADDUBSW`）などを用いて内積を計算する．

活性化関数は，整数のベクトルのビット幅を変換する命令（`VPACKSSDW`, `VPACKSSWB`）と， 8-bit 整数ベクトル 2 つの要素ごとの最大値を取る命令（`VPMAXSB`）などによって計算する．

表 1 『the end of genesis T.N.K.evolution turbo type D』の評価関数で使用する重み行列のサイズ．

	列数	行数
$\boldsymbol{W}_1$	125388	256
$\boldsymbol{W}_2$	512	32
$\boldsymbol{W}_3$	32	32
$\boldsymbol{W}_4$	32	1

2.9 モデルサイズ

表 1 に，『the end of genesis T.N.K.evolution turbo type D』の評価関数で使用する重み行列のサイズを示す．列数と行数が，それぞれアフィン変換の入力と出力の次元数に対応する．このモデルサイズは，単位時間あたりに読める局面数が，評価関数に三駒関係を使用した場合と同等になるように設定したものである．

全パラメータ数の大半を，特徴ベクトルのアフィン変換に使う重み行列 $\boldsymbol{W}_1$ が占めている．この部分は差分計算を行うため，実際の計算コストは大きくない．$\boldsymbol{W}_2, \boldsymbol{W}_3, \boldsymbol{W}_4$ を使うアフィン変換は差分計算ができないが，パラメータ数を比較的少なく設定し， 8-bit 整数の SIMD 演算によって高速な計算を可能にしている．

3. おわりに

本稿では，ニューラルネットワークを用いた将棋の評価関数を提案し， CPU で高速に動作させるための設計と高速化技術について述べた．本技術による評価関数を搭載する初めての将棋ソフト『the end of genesis T.N.K.evolution turbo type D』は，第 28 回世界コンピュータ将棋選手権 [1] に参加を予定している．

また，本技術を実装したソースコードを後日公開する予定である．オープンソースの将棋ソフト『やねうら王』[8] をベースとして C++ で実装したもので，本稿で述べた内容に加えて，以下のような特徴がある．

- クラステンプレートを利用した，特徴量やネットワーク構造のカスタマイズ性
- コンパイル時計算やコンパイラによる最適化を活用した高速な実装
- 三駒関係の学習で用いられる手法と同様の，
  - 静止探索と組み合わせた学習
  - 特徴量の分解による学習時のパラメータ共有

特徴量やネットワーク構造は，『the end of genesis T.N.K.evolution turbo type D』で採用したものが最適

であるとは限らない．より優れた構造の検討は今後の課題であり，ソースコードを拡張して利用される他の開発者にも期待したい．

本技術がコンピュータ将棋の更なる発展に繋がれば幸いである．

参考文献

[1] 第 28 回世界コンピュータ将棋選手権. http://www2.computer-shogi.org/wcsc28/.

[2] 金子知適, 田中哲朗, 山口和紀, 川合慧. 駒の関係を利用した将棋の評価関数. ゲームプログラミングワークショップ 2003 論文集, pp. 14–21, 2003.

[3] Kunihito Hoki and Tomoyuki Kaneko. Large-scale optimization for evaluation functions with minimax search. *Journal of Artificial Intelligence Research*, Vol. 49, No. 1, pp. 527–568, 2014.

[4] 金澤裕治. NineDayFever アピール文書. http://www.computer-shogi.org/wcsc24/appeal/NineDayFever/NDF.txt, 2014.

[5] 竹内章. コンピュータ将棋における大局観の実現を目指して. 人工知能学会誌, Vol. 27, No. 4, pp. 443–448, 2012.

[6] David Silver, Aja Huang, Chris J. Maddison, Arthur Guez, Laurent Sifre, George van den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, Sander Dieleman, Dominik Grewe, John Nham, Nal Kalchbrenner, Ilya Sutskever, Timothy Lillicrap, Madeleine Leach, Koray Kavukcuoglu, Thore Graepel, and Demis Hassabis. Mastering the game of Go with deep neural networks and tree search. *Nature*, Vol. 529, No. 7587, pp. 484–489, 2016.

[7] David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dharshan Kumaran, Thore Graepel, Timothy Lillicrap, Karen Simonyan, and Demis Hassabis. Mastering chess and shogi by self-play with a general reinforcement learning algorithm. arXiv:1712.01815v1 [cs.AI], 2017.

[8] 磯崎元洋. やねうら王オープンソースプロジェクト. http://yaneuraou.yaneu.com/yaneuraou_mini/.

読み太　アピール文書

2018/3/30 塚本隆三

自己紹介

・世界コンピュータ将棋選手権は第26回から参加させて頂いております。

・開発自体は4年前から行っています。プログラミングを覚えたのもそれくらいからです。

・毎回ノートパソコン1台で出場していました。

・ノートパソコンでの出場を期待されている方は、ごめんなさい。今回は違います。

評価関数

・3駒関係+手番評価（KPPT型）です。

・第5回電王トーナメントで使った評価関数に追加学習を行いました。

・depth8で200億局面の教師局面を作りました。

・現時点でWCSC27のelmoに勝率75%程度の強さです。

疎結合並列探索

・疎結合並列探索（クラスタ並列探索）を行います。

・Amazon EC2 を利用します。c4.8xlargeを16台借りる予定です。

・並列化アルゴリズムに関しては「技巧」を非常に参考にさせて頂きましたので、ライブラリとして申請させて頂きました。

・「技巧」は並列化部分のソースを公開している唯一のプログラムであり、並列化に限らず、将棋プログラムとしての設計が非常にきれいで私の好みであることが選定理由です。

Byteboard

・Byteboardというデータ構造を使っていますが、現時点ではBitboardよりも高速化できていません。どうやらByteboard本家である「たこっと」とは少し違った実装をしていたようです。

・指し手生成にByteboardを使ってはいけないそうです。びっくりしました。

・大会までに余裕があれば使えるようにします。（Byteboardで高速化できる部分はボトルネックになっている部分ではないので、どうしても後回しになってしまっています）

その他

・Linuxでの開発に戸惑っています。

・AWSの使い方が難しいです。

・学校を卒業したので、もうパソコンが借りることができなくなりました。今まで快く貸してくださって、近畿職業能力開発大学校の先生方には本当に感謝しています。おかげでとても強くなりました。

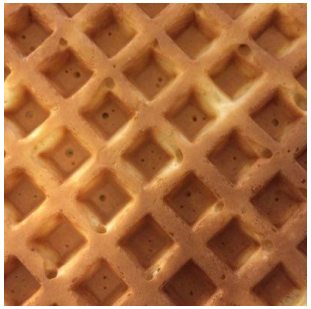
最後に

・読み太はStockfish、やねうら王、Apery、技巧を参考にさせて頂いております。感謝申し上げます。

・ハードウェアが強くなった分、もう言い訳はできません。全力の読み太にご期待ください。


HoneyWaffle
第28回世界コンピュータ将棋選手権　アピール文書
開発者　渡辺 光彦

開発者



氏名: 渡辺 光彦

職業: プログラマー

棋力: 将棋ウォーズで3級、ぴよ将棋でR700-800程度の振り飛車党

Twitter: @shiroi_gohanP (https://twitter.com/shiroi_gohanP)

ニコ生の電王戦をきっかけにコンピュータ将棋を始める。

将棋連盟Liveやニコニコ放送、AbemaTVの将棋中継が好き。

HoneyWaffle (ハニーワッフル) 名前の由来

・四角いワッフルは将棋盤と似ている

・ゆるふわスイーツ的なスナック感覚の軽さを表現

　元々タブレット向けに開発していたので物理的に軽いこと、振り飛車の軽い捌きができるようになるといいなという想いから命名しました。

※表紙やTwitterアイコンのワッフルはうちで焼いたものを使用しています。

以下のリンク先で出せるものは公開しています。使い方がおかしいのはいつものこと。

https://github.com/32hiko

戦績

2016年、Go言語でオリジナル開発版

・2016年5月 第26回世界コンピュータ将棋選手権 出場　一次予選2勝5敗

・2016年10月 第4回将棋電王トーナメント 出場　予選リーグ3勝5敗

2017年、やねうら王ライブラリ使用

・2017年5月 第27回世界コンピュータ将棋選手権 出場　決勝リーグ7位

・2017年11月 第5回将棋電王トーナメント 出場　決勝トーナメント初戦敗退

本題に入る前にポエム①

今回のコンセプトの前に、将棋とは何かをもういちど考える。将棋とは？

> 二人が「盤」の上で交互に「駒」を動かします。そして相手の「玉将（ぎょくしょう）」という駒を先に捕獲したほうが勝ちとなります。

（日本将棋連盟サイトから引用　https://www.shogi.or.jp/knowledge/about/ ）

→先に捕獲するとは、先に攻めるということ？

→先に攻めるのが有利ということ？

→そう考えるのが自然か。

本題に入る前にポエム②

ここ最近の将棋（コンピュータ将棋や、それに影響を受けたプロの将棋）は、確かに先に攻めようとする傾向がさらに強くなったかも。

・玉は1手2手動かすだけで、金銀もあまり玉に近づけずバランス重視（角換わりとか横歩取りとか）

・じっくりした矢倉が減った

・早く攻めるために桂馬も早く跳ねる

今現在の3駒関係の学習も、要は「早く勝ちやすい形を高く評価する」もの （現局面の評価値を、何手か先の局面の評価値に近づくよう学習する。勝った局面はより高く評価するよう学習する。）

本題に入る前にポエム③

「早く攻める」のを重視する前提なら、なるほど振り飛車は不利だと言わざるを得ない。少なくとも飛車を振る手で1手かかってるのは事実。工夫しないで学習回すと、飛車振らなくなっていくのも納得。

ただ、逆に、将棋の勝敗そのもので不利なわけではないよねと。純粋に手数がのびたとしても、最後に相手よりも1手早いだけでいいから。

先に攻めるのは先に詰ます可能性もあるけど、先にカウンターを食らう、先に攻めが切れるという恐れもあるよねと。

相掛かりで5手目に▲２四歩から無理やり攻めようとすると、先に歩を手にした後手が有利になる変化。他の戦型でも極限まで攻めを早くしていったら、それと似たような局面ばかりになるかもしれない。

本題に入る前にポエム④

現在の潮流（居飛車が多い）：最速の攻めを目指す。攻撃が最大の防御。

→相居飛車でお互いに最速の攻めを目指すのは、戦型にこだわらない陣営が自然と当たり前にやっている。

→ならば、それに対抗できる振り飛車のソフトはかくあるべし！

・最速の攻めが来てもギリギリ受けられる受けの力（構想、囲い）

・隙を見て豪快に攻め、鮮やかに1手違いで勝つ

定跡や変な評価関数でただ飛車を振るだけでは真の「振り飛車」ではない。1手遅れたあげくに、中途半端に囲って、それから最速の攻めを目指してもね…。この辺は割と当たり前というか基本のはずなのにどうして…。

コンセプト

・「公開されている技術を最大限に活用し、最強の振り飛車党ソフトを目指す」

実は第5回将棋電王トーナメントの時と同じですが、意味合いはポエムで述べた通りまったく違います。去年一年、飛車を振らせるのに苦労したのは確かなのですが、それが精一杯で、考えが全然足りませんでした。

一般的に形勢判断の基準は①駒の損得②駒の働き③玉の堅さ④手番と言われていますが、現状の3駒関係では、①②④の評価がとても洗練された半面、前述のとおり攻めを重視した結果③の影が薄いように思います。意識して「堅く囲ってから鋭く攻める」ことができるよう、以前までの取り組みに加えて③を重視します。

ハードはAWS EC2インスタンス1台（極力高性能なものを選択）の予定です。

構成

■開発部

ライブラリ申請したやねうら王を強引に改造（既存の3駒での評価に加え、別に堅さ評価する等）★採用理由：最高の探索、フレームワーク。去年から使用している。

AWS接続兼時間攻めモジュール（ローカルにて使用）は自作

■評価関数

ライブラリ申請したAperyのものがベース　★採用理由：最高の評価関数

教師局面生成にて評価値を意図的に改変し追加学習する等で調整　https://github.com/32hiko/HoneyWaffleSDT5

■定跡

第5回将棋電王トーナメントで使用したものをベースに、より持久戦を目指す方針で調整

最後まで読んでいただき、ありがとうございます。

●NineDayFever

局面評価の極北を目指しています。

2018年の相違点ですが、強化学習に使用する自己対戦棋譜の初期状態生成のため、以前からやっていた自己対戦の勝率に基づいた定跡生成手法を流用しています。定跡生成のため抽出したbookの拡張候補各局面から最低数十局程度の棋譜を生成し(並列処理による結果変動によって結果は異なる)、生成された5千万局程度の棋譜に基づいてelmo方式(勝率からのロジスティック回帰＋クロスエントロピーによる数手先の探索結果の学習)による学習を行うと同時に、勝率に基づいて book 内の各手の採用確率を決定します。

・ライブラリ選択理由
2012年ごろにKPP/KKP テーブルで機械学習手法を試そうとした時点では bonanza 以外に選択肢がなかったので、そのまま使い続けています。惰性です。

以下は2017年の内容です。

・対戦中に現れた局面を調べ、機械学習結果の欠陥を探して修正していています。
・三駒関係の各変数を分解して共通する要素を抽出したうえで機械学習することで未知の局面への対応能力を高めています。
・手番を考慮した評価値を使用しています。
・定跡では自己対戦結果から各局面の勝率の分布を推定して各手の採用確率を決めています。
・プログラムはbonanza 6.0にstockfish の手法を取り入れています。

第 28 回 世界コンピュータ将棋選手権

大将軍（たいしょうぐん） アピール文書

開発者 横内 健一　横内 靖尚

○大将軍の概要

評価関数に主眼をおいた将棋ソフトです。

過去には 4 駒の位置関係（KPPP:N4(2013) / KKPP:N4S(2013-2015)）を評価関数に使用していました。現在は、学習作業の効率を考慮し現在は 3 駒関係の評価関数を用いています。

○大将軍の特徴

評価関数の作成に関しては、以下の点を工夫しています。

（評価関数は、以下の 2 ステップの手順にて学習しています）

1.プロの棋譜からの学習

いわゆる Bonanza メソッドをベースに、手番の評価やミニバッチの手法を取り入れています。

ミニバッチを用いることで、学習が安定し、短時間で学習の成果を確認することができます。

手番に関する評価は、3 駒に手番加えた KKPT 型を採用しています。KPPT 型よりも計算コストが小さいため、10%程度探索速度が向上します。

2.自己の探索結果からの強化学習

プロの棋譜からの学習において勝率が飽和したところで、浅い探索結果と深い探索結果の評価値を用いて、評価値の不整合を修正していきます。

学習させる局面により勝率に影響するようですが、とのように設定するとよいかは今後の課題です。

○ライブラリの使用と実装方針

やねうら王ライブラリを使用します。評価関数の開発に注力するために、探索部分はライブラリを使用して他のソフトと同等レベルをキープしたいと考えました。他のライブラリと比較して、ソースコードが理解しやすいというのも選定理由となりました。

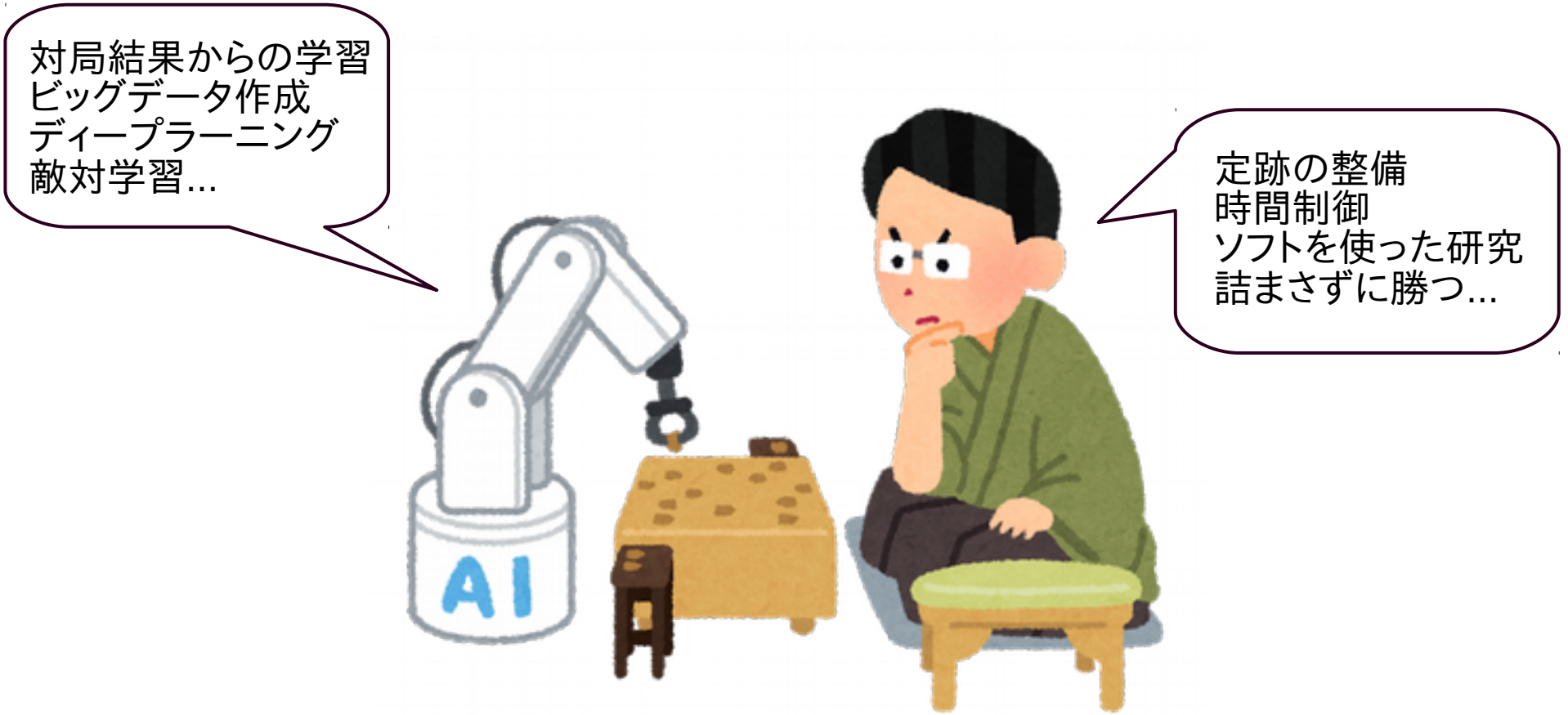
今日では、プロの棋譜を使わなくても評価関数を作成することが可能ですが、従来手法の学習結果（ウナギ屋のタレ）とミックスして活用していきます。

elmo ライブラリは現在使用していませんが、評価関数が優れており、選手権までに定跡部や追加学習等で使用するかもしれませんので、使用申請をしておきます。

妖怪惑星Qhapaqのアピール文

Ryoto Sawada, Yuki Ito and Toshihiro Shirakawa

ITに強い将棋部 (updated 2018/04/08)



Who is Qhapaq ?

カパック（ケチュア語で「偉大なもの」を指す形容詞）と読みます。本作が数々の巨人の肩に立った作品であることを示しています。

数理解析を駆使したご家庭レベルのPCで行える強化学習や、高速化を売りにしています

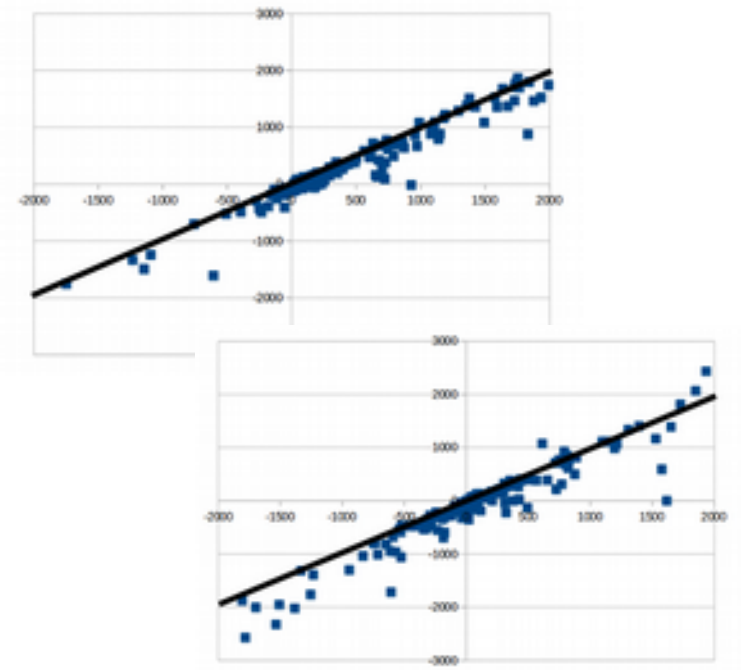
主な実績：
・第27回世界コンピュータ将棋選手権10位
・第五回将棋電王トーナメント：5位入賞
・現在最強の公開関数Apery-Qhapaq関数
・電王トーナメント累計23戦中18回後手

チーム名：ITに強い将棋部

Ryoto Sawada : 某企業 and 大学の研究者。専門は量子物理
Yuki Ito : Ubuntu帽を被った野生の物理学徒。高速化のプロ
Toshihiro Shirakawa : パズル世界の怪人。趣味は電子ゴルフ

将棋フィーバー(ただしAI)

AIを使って将棋の可能性を広げたい！



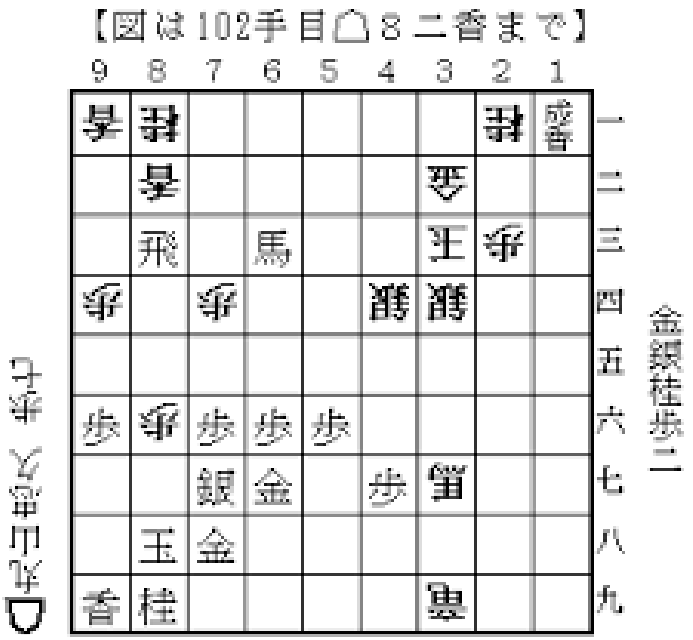
将棋界：
- 藤井フィーバー
- 羽生永世七冠誕生
- ひふみん超おもしろい

AI将棋界：
- Ponanza引退
- オープンソース勢の大躍進
- AIブームに便乗した手法の進歩

評価値推移から見る棋風解析
藤井 vs 羽生は藤井勝率65%ぐらいらしい(Qhapaq曰く)

1年前のチャンピオンとのレート差が300-400。角落ちのレート差が600ぐらいと言われているので、2年で角落ちを埋める程度の成長率

Qhapaqの挑戦：まず強くする



今の流行は強化学習

- 人間やソフトの棋譜を使った学習
- 勝った局面は良い、負けた局面は悪い
- ソフトの評価値と勝敗を合議(elmo絞り)
- やねうら王などのライブラリで個人でも再現可能

今後の課題：

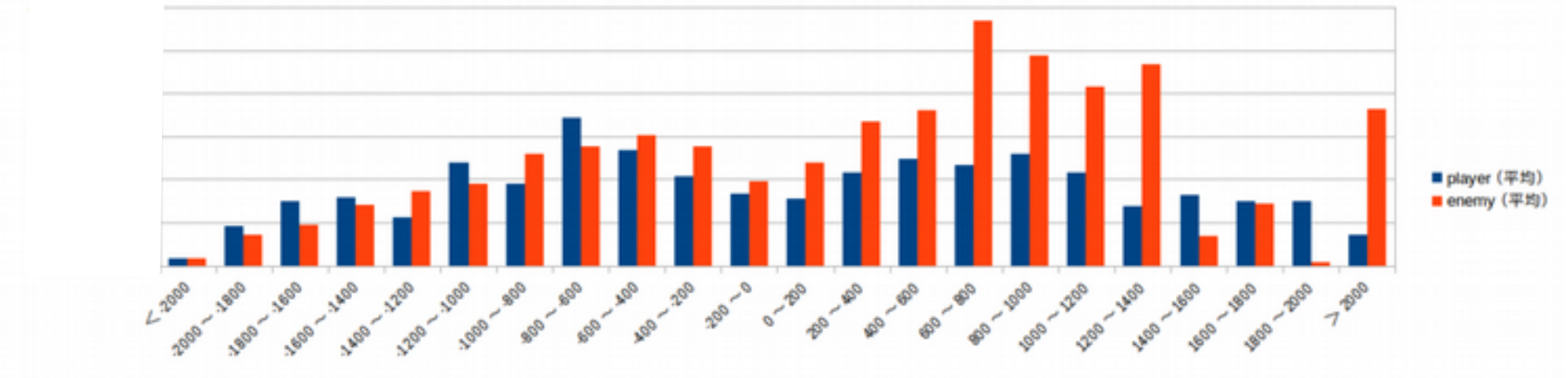
- 普通に棋譜を作ると滅茶苦茶計算資源がかかる
- 教師の数、読みの深さ双方が必要
- 人間ならすぐ駄目と解ることでもコストが高い
- 評価関数は線型のままで良いのか

Qhapaqの工夫：

- floodgateなどで得られる深い読みの局面を利用（教師精度はNo1）
- 教師データが少ないのでその局面から浅い読みで作られた局面も教師にする
- その教師の勝敗や評価値は他データから類推する
- 今で言うVirtual Adversarial Trainingに近い
- KPPTを強くする以外にも新規評価関数開拓をやってる
- 高速化もやってる（チームメンバーのItoさんが）

Qhapaqの挑戦2：将棋を面白くする

棋力解析ソフトELQ



上の図は藤井六段と対局相手のソフト視点での悪手率。
横軸は評価値（左が不利な局面、右が有利な局面）

- 藤井六段、対局相手の双方とも、-600～600ぐらいのイーブンな局面が一番ミスが出やすい
- 藤井六段は有利な局面でのミスが少ない。詰将棋万歳

このデータを用いて対局者の強さを予想することも可能。藤井六段は
今後羽生永世七冠を越えて行くとQhapaqは予想していますが果たして...

利用ライブラリとその選定理由

・Apery
評価関数の初期値に使っています。また、学習部には
Aperyの学習部を改造したものを用いています

・やねうら王
探索部、学習部として使っています

・技巧
進行度などの取り扱いや、ssh通信部は技巧のコードを
参考にしています。ゼロから書き直しているとはいえ、コピペ
部分も多いのでライブラリ申請しておきます

最後にひとこと

Qhapaq強いですよ!

将棋ブームをエンジョイしてるそこの人、コンピュータ将棋も見てくれると嬉しいです!

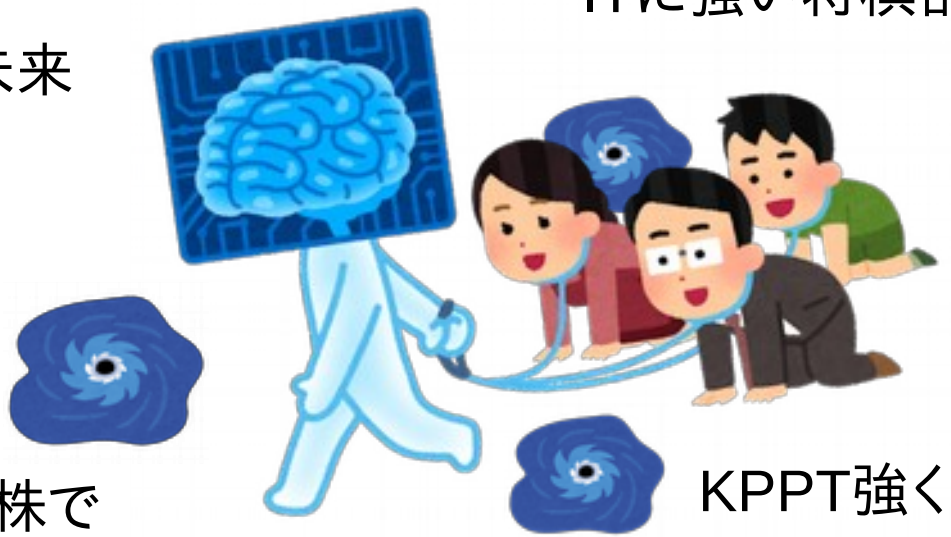
再放送

妖怪惑星Qhapaqのアピール文

Ryoto Sawada, Yuki Ito and Toshihiro Shirakawa

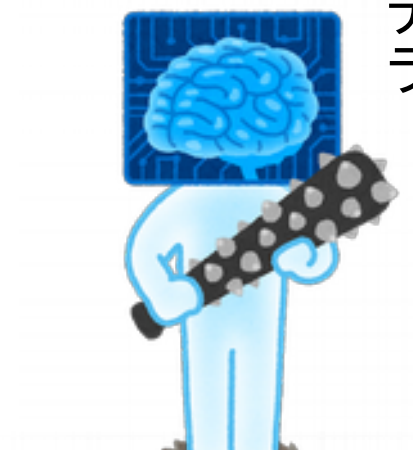
AIが導く
明るい未来

ITに強い将棋部



KPPT強くするぞ

ディープ
ラーニング



ハイテク株で
X万円を溶かした
Qhapaq

GPU

小市民

Who is Qhapaq ?

カパック（ケチュア語で「偉大なもの」を指す形容詞）と読みます。本作が数々の巨人の肩に立った作品であることを示していますが後述するやらかしは巨人たちの責任ではありません。

主なやらかし:
・被り物を被ってニコニコ動画出演
・振り駒の結果に中指を立てる
・対戦カードを正しく組んでもらえない
・通信器具を会場に持ち込み忘れる
・電王トーナメントでのサイリウム配布テロ
・なんとかchに経歴を晒される

チーム名:ITに強い将棋部

Ryoto Sawada : wikipediaに名前がある。1日の食費は300円ちょい
Yuki Ito : 異議あり民の一人。ラーメン屋で外人に絡まれて困惑してた
Toshihiro Shirakawa : 物書き時々ﾕｰﾁｭｰﾊﾞｰ。服装がﾌｧﾝﾀｽﾃｨｯｸ

将棋フィーバー(ただし身内)

AIを使って開発者のキャリアを広げたい！

- bing+カタカナだと出てこない
- googleだと将棋をサジェストされる
- Qhapaqで検索すると出てくる
- カタカナ検索の結果は諸事情で出せない

将棋界：
- 三浦九段の冤罪事件
- 加藤 One hundred twenty three
- 某電王による大量献金

妖怪惑星：
- アカデミアから(強制)引退
- AIブームに便乗(ただしAI株で失敗)
- 履歴書に書けるネタを稼いでｷｬﾘｱｱｯﾌﾟ

最近AIができる物理屋ではなく、物理ができるAI屋と思われているフシが有る。

Qhapaqの挑戦：人の失敗を吊るし上げる

elmo絞りの此処が駄目：

- 短い持ち時間だと強くならないことが多い
- 一度しか出てないKPPTの値がelmo絞りだと大きくなりすぎる問題
- 教師局面110億とか狂気の沙汰
- ニューラルネット系の学習だと超強力

評価関数合成の此処が駄目：

- 合議制では強くなるような関数でも強くならない(ことがある)
- 強くなる関数では合議より強くなる
- どの成分の変化がレート変更に結びつくかを分析する必要あり

Qhapaqの此処が駄目：

- こうご期待の一言でアピール文を片付けられず、大会前に役立つ情報を漏らしてしまう

Qhapaqの挑戦2
ソフト開発者 ＝ 秀才の構図を破壊する

■参加プログラム情報

チーム名：

振り飛車!飛車!飛車!飛車ぅぅううぅわぁあああああああああああああああああああああああああん!!! あぁあああああ…ああ…あっあっー!あぁあああああああ!!!飛車飛車飛車ぅううぁわぁあああああ!!! あぁクンカクンカ!クンカクンカ!スーハースーハー!スーハースーハー!いい匂いだなぁ…くんくん んはぁっ!48に居る飛車たんの裏面の桃色ブロンドの文字をクンカクンカしたいお!クンカクンカ!あぁあ!! 間違えた!モフモフしたいお!モフモフ!モフモフ!龍の文字モフモフ!カリカリモフモフ…きゅんきゅんきゅい!!りゅうおうのおしごと7巻絶賛発売中だよぅ!!あぁぁああ…ああああ…あっあぁあああああ!!ふぁぁあああんんっ!! sdt未だ終わってなくて良かったね飛車たん!あぁあああああ!かわいい!飛車たん!かわいい!あっあぁぁああ! アニメも評価されてて嬉し…いやぁあああああああ!!!にゃあああああああああん!!ぎゃあああああああ!! ぐあああああああああああああ!!!将棋なんて現実じゃない!!!!あ…小説もアニメもよく考えたら… 振り飛車が勝ち越す未来は現実じゃない?にゃあああああああああああああああん!!うぁあああああああああああ!! そんなぁあああああああ!!いやぁぁぁああああああああああ!!はぁあああああああん!!48飛車ぁあああああ!! この!ちきしょー!やめてやる!!振り飛車なんかやめ…て…え!?見…てる?角交換四間飛車を指しこなす本の飛車ちゃんが僕を見てる? 四筋の飛車ちゃんが僕を見てるぞ!飛車ちゃんが僕を見てるぞ!駒台の角ちゃんも僕を見てるぞ!! アニメのあいちゃんが僕に話しかけてるぞ!!!よかった…世の中まだまだ捨てたモンじゃないんだねっ! いやっほぉおおおおおおお!!!僕には将棋ちゃんがいる!!やったよケティ!!ひとりでできるもん!!! あ、コミックのひなたちゃあああああああああああああああん!!いやぁあああああああああああああああ!!!! あっあんああっあんあアン様ぁあ!!セ、セイバー!!シャナぁあああああああ!!!ヴィルヘルミナぁあああ!! ううっうぅうう!!俺の想いよ将棋盤へ届け!!readyok! と言った感じでITに強い将棋部は常に最強を求め将棋ソフト開発に余念がない。しかし、開発の疲れからか不幸にも疲れからか、不幸にも黒塗りの高級車に追突してしまう。後輩をかばいすべての責任を負った三浦に対し、車の主、暴力団員谷岡に言い渡された示談の条件とは…。

WCSCのサイトが前時代的なcgiであり
チーム名の文字数制限が抜けていたので
チーム名をルイズコピペにしてみた
（360RTぐらいされた）

たぬき開発者と共謀して電王トーナメントの
ライブに自腹でサイリウムを持ち込んだ

肩を借りた巨人とその選定理由

・既存定跡を組み込んだ量子アニーリング
定跡作りの際に使っています。複数の既存定跡にノイズ（具体的には一定確率で定跡を無視する）を加えたものと自己対局を行い、勝った棋譜を収集することを繰り返すことで、未知の定跡に対してロバストな定跡を作っています

・妖怪惑星クラリス
今回のソフト名の元ネタです。このゲームが末永く発展することを願って付けました

・Lostorage conflated WIXOSS
sdt5でのソフト名の元ネタです。四期の放送（当時未確定）を願って付けました。願いが叶ったわけですが、これはもしや...

最後にひとこと

妖怪惑星クラリスが唐突にサービス終了しました

ｳｷﾞｬｱｱｱｱｱｱ

nozomi アピール文書 2018

2018/3/31 Yuhei Ohmori

はじめに

- コンピューター将棋業界にもDeepLearningとMCTSのビッグウェーブがやってきました
- 「乗るしかないこのビッグウェーブに！」

評価関数

- 今まで通り、KPP + KKPTの3駒関係で、自己対局の勝敗と評価値を教師として学習しています
- なるべく深いdepthで自己対局を行うようにしています
- 時間とお金を節約するため、少ない棋譜からでも学習できるようにしました
  - 一回のパラメーターの更新に2000万局面使用しています
  - 損失関数もelmo形式ではなくちょっと変更しています
- ちょっとだけ評価値の計算を改善しました

探索

- 今まで通り、Stockfishベースの探索です
- 指し手の選択確率を計算し、オーダリングに使用しています
  - 単純に指し手と駒の位置関係でロジスティック回帰しました
    - 評価関数の学習とほぼ一緒なのでお手軽
  - nozomiの指し手を一手も読まずに35%の確率で予言可能
    - 残念すぎますが、まあこんなもんかなあ
- 指し手の選択確率をLMRに対しても使用する予定です

その他

- チェスソフトのStockfishをベースに作成しています
  - https://github.com/official-stockfish/Stockfish
- 一部Aperyを参考にさせていただいています
  - https://github.com/HiraokaTakuya/apery
- いつものことながらこれらのソフトがなければnozomiはここにいなかったので、大変感謝しております
- C#はAWSとのSSH通信で使用しています

おわりに

- 誠に遺憾ではありますが、ビッグウェーブに乗れなかったものと認識しております
- それでは今年もよろしくお願いいたします

第28回世界コンピュータ将棋選手権
Aperyアピール文書
2018年3月31日 平岡 拓也、杉田 歩

・Aperyとは
- 読み方は「えいぷりー」です。
- GPL v3ライセンスでオープンソースで開発しています。
- ソースコードや実行ファイルは以下で公開しています。
- http://hiraokatakuya.github.io/apery/
- CSAライブラリとして登録しているので、誰でもAperyを改造してご自身のソフトとして出場出来ます。

・作者紹介
- 平岡 拓也 (ひらおか たくや)
  趣味で作っています。
  Mail　: hiraoka64@gmail.com
  Twitter: @HiraokaTakuya
- 杉田 歩 (すぎた あゆむ)

・実績
- 第22回世界コンピュータ将棋選手権 22位 (2次予選敗退)
- 第23回世界コンピュータ将棋選手権　9位 (2次予選敗退)
  決勝進出に一歩足りず。
  新人賞に一歩足りず。
- 第1回将棋電王トーナメント 6位 (5位決定戦敗退)
  第3回電王戦出場に一歩足りず。
- 第24回世界コンピュータ将棋選手権 優勝
- 第2回将棋電王トーナメント 5位
- 将棋電王戦FINAL 斎藤慎太郎五段(段位は当時)に完敗
- 第25回世界コンピュータ将棋選手権 4位
- 第3回将棋電王トーナメント 3位
  出場ソフト名は「大樹の枝」。
  名前はヤフオク！の「みんなのチャリティー」で命名権をオークション(全額チャリティー)で販売して落札者の方に決めて頂きました。
- 第26回世界コンピュータ将棋選手権 4位
- 第4回将棋電王トーナメント 2位
  出場ソフト名は「浮かむ瀬」。
  名前はヤフオク！の「みんなのチャリティー」で命名権をオークション(全額チャリティー)で販売して落札者の方に決めて頂きました。
- 第26回世界コンピュータ将棋選手権 12位
- 第5回将棋電王トーナメント 2位

・技術的特徴
- 全体的にチェスソフトのStockfishの設計を取り入れており、評価関数は3駒関係です。利きのデータなどは持っておらず、シンプルな構成になっています。
- 評価関数で手番も評価しています。
- 評価関数の教師データ生成を皆さんにお願いするシステムを今年も使っています。
  今回は Linux 限定になっています。
  ご協力ありがとうございます！
  https://github.com/HiraokaTakuya/apery-machine-learning

・昨年との違い
- 昨年の選手権で elmo が採用していた、評価関数の学習データ生成時の対局で、
  勝敗を記録し、学習にその局面と後の勝敗を考慮する方式を取り入れました。
- 学習部を書き直した過程でバグが取れたのか、WCSC27 の elmo の評価関数より精度が上がり、
  半年前の第5回将棋電王トーナメント時には、オープンソースの中では最も評価関数の性能が高かったようです。
- 半年前からの変更点は、学習データ生成時の探索深さを8から10に変更した点です。
  とても計算量が多いので、半年前は出来ませんでしたが、やることにしました。
  これは DeepMind の AlphaZero が AlphaGo と同様の方法で将棋でも既存のソフトより強くなったとい

う発表を受けて、
　単に学習に使った計算量が桁違いに大きいからではないかと考えたからです。
　既存手法でももう少し計算量を増やして学習すれば、更に強くなるのではないかと考えました。
　具体的には、探索深さを 10 で学習し、その後 12 でもう一度学習する計画でした。
　学習データ生成を誰でも出来るシステムなども作りましたが、時間的に探索深さ 12 は実験出来そうにないです。
　まだ実験中で、探索深さ 10 でのデータ量が 8 の時よりも少し少なかったりで、結論を出すには早いですが、
　探索深さ 10 の時点で伸びがほとんど無いかも知れません。
　本番までにもう少し実験しようかと思います。

第28回
世界コンピュータ将棋選手権
なのはアピール文書

2018年4月1日 川端一之

■**なのは**ってなんだよ

- 熱血魔法バトルアクションアニメ「魔法少女リリカル**なのは**」シリーズの主人公**高町なのは**を由来にし、さまざまな称号を冠する彼女のような強さを盤上で実現したいという願いを込めています。

- よく「名前の割に強い」という声をいただきますが、その認識は逆で、「名前負けしている」や「名前の割に弱すぎる」というほうが妥当な評価です。

■作者はどんな人？

- 静岡県出身 愛知県在住
- とあるメーカーに勤務
- 好きな食べ物は焼肉、しゃぶしゃぶ、寿司
- 好きなアニメは魔法少女リリカルなのは、りゅうおうのおしごと！、とある科学の超電磁砲、ラブライブ！
- 将棋ウォーズ 1級
- 囲碁は日本棋院 初段(アマチュア)
- スマホゲーム非課金勢…スクフェス、デレステ、Fate/GO、ぷにぷに等
- ！職業プログラマ∩！研究者　　∩！東大∩！学生

■開発環境

- こんなPCで開発していますｗ

- 出場PC(予定)

  CPU：AMD R7 1700

  メモリ：32GB

  OS：Windows10

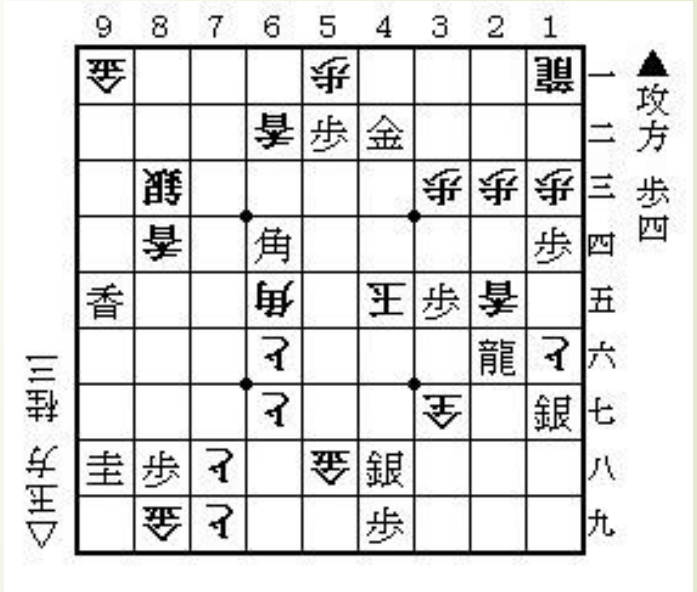
■なのはの構成

- Visual Studio Community 2017(C++)にて開発
- 手生成では歩、角、飛の不成も生成
- 探索はStockfish使用
  - 評価ベクトル縮小、詰めルーチン(df-pn)削除したものを、**なのは**miniとして公開
- Bitboard未使用(盤情報は配列)
  - AMD RYZENに適したデータ構造
- 定跡部は実戦での出現数および勝率を考慮して手を選択
- 評価ベクトルはライブラリ使用予定
- 詰めルーチン(df-pn)搭載

…と、どこにでもあるような平凡な構成

■**なのは**の特徴は？

- 強力な詰めルーチン搭載(**なのは**詰めとして公開)
- **なのは**詰めは江戸時代の名作 611手詰めの「寿」を解く
- 常に詰みを狙い、**一発逆転するポテンシャル**を秘めています！



■なのはの新たな特徴(予定)は？

- 詰めルーチンの更なる強化
- 終盤の強化
  - コア数が増加したため、詰め探索用のスレッドを設定

ユ「詰め探索変更ってどんな風に変えたの？」

な「うん…あれって探索が遅くて高速戦では使えないから」

ユ「やっぱり高速化？」

な「ううん　逆！チャージタイムを増やして威力を大幅アップ！」

な「最大詰め手数の強化を最優先してみたの」

ユ「そ…　そう……」

※以下を参考に改変:「魔法少女リリカルなのはA's THE COMICS」pp.20-21, 原作 都築真紀, 作画 長谷川光司, 学習研究社, 2006.

■なのはの強さ

- 第24回世界コンピュータ将棋選手権で**1位！**(*1) (*2)
- 第25回世界コンピュータ将棋選手権で**1位！**(*1) (*3)
- 第26回世界コンピュータ将棋選手権で**2位！**(*1) (*4)
- 第27回世界コンピュータ将棋選手権で**1位！**(*1) (*5)

(*1) AMD製CPUをメインに使ったシステム構成での参加ソフトの中で(当者調べ)

(*2) トータルでは17位　　(*3) トータルでは11位

(*4) トータルでは12位　　(*5) トータルでは13位

■意気込み

- できれば**シード権獲得！**
- あわよくば**入賞！**(あと、AMD勢で1位奪取したい)
- **全力全開、手加減なしで！！**

(今後の課題)

- 検討に使われるように振り飛車が不利飛車にならないようにする
- 詰めろ絡みの局面が続いても正着を続ける終盤力(即詰みがない時)
- 最長詰め手数の詰将棋の「ミクロコスモス」 (1525手詰め)を解く
- カットインやエフェクトなど**派手な演出**！！

■使用ライブラリについて

- なのはmini　　自作のため、いろいろなところで使っています。
- やねうら王　　学習部や評価ベクトルを使用(予定)。公開からこなれて信頼性が高いため。
- Bonanza　　ちょっとしたテストをするときなど、慣れもあって使いやすいため。

■最後に

- **なのは**アピール文書は以上です
- 最後まで読んで頂きありがとうございます

絵： COCOさん

■参考文献

- 小谷善行、他:「コンピュータ将棋」, サイエンス社, 1990.
- 松原仁 編著:「コンピュータ将棋の進歩」, 共立出版, 1996.
- 松原仁 編著:「コンピュータ将棋の進歩2」, 共立出版, 1998.
- 松原仁 編著:「コンピュータ将棋の進歩3」, 共立出版, 2000.
- 松原仁 編著:「アマ四段を超えるコンピュータ将棋の進歩4」, 共立出版, 2003.
- 松原仁 編著:「アマトップクラスに迫るコンピュータ将棋の進歩5」, 共立出版, 2005.
- 池泰弘:「コンピュータ将棋のアルゴリズム」, 工学社, 2005.
- 金子知適, 田中哲朗, 山口和紀, 川合慧:「新規節点で固定深さの探索を併用するdf-pnアルゴリズム」, 第10回ゲーム・プログラミングワークショップ, pp.1-8, 2005.
- 脊尾昌宏:「詰将棋を解くアルゴリズムにおける優越関係の効率的な利用について」, 第5回ゲーム・プログラミングワークショップ, pp. 129-136, 1999.
- 保木邦仁:「局面評価の学習を目指した探索結果の最適制御」 http://www.geocities.jp/bonanza_shogi/gpw2006.pdf
- 岸本章宏:「IS 将棋の詰将棋解答プログラムについて」, http://www.is.titech.ac.jp/~kishi/pdf_file/csa.pdf, 2004.
- 橋本剛, 上田徹, 橋本隼一:「オセロ求解へ向けた取り組み」, http://www.lab2.kuis.kyoto-u.ac.jp/~itohiro/Games/Game080307.html

■参考Web

- やねうら王 公式サイト: http://yaneuraou.yaneu.com/
- 千里の道も一歩から: http://woodyring.blog.so-net.ne.jp/
- 小宮日記: http://d.hatena.ne.jp/mkomiya/
- State of the Digital Shogics [最先端計数将棋学]: http://ameblo.jp/professionalhearts/
- ながとダイアリー: http://d.hatena.ne.jp/mclh46/
- 毎日がEveryday: http://d.hatena.ne.jp/issei_y/
- Bonanzaソース完全解析ブログ: http://d.hatena.ne.jp/LS3600/
- aki.の日記: http://d.hatena.ne.jp/ak11/
- FPGA で将棋プログラムを作ってみるブログ: http://blog.livedoor.jp/yss_fpga/

※読めなくなったサイト含む

昨年大会のアピール文書をご覧ください。
http://www2.computer-shogi.org/wcsc27/appeal/OkaraManju/OkaraWCSC27.pdf

第 27 回世界コンピュータ将棋選手権
おから饅頭　アピール文書

渡辺　敬介

2017 年 4 月 9 日

プログラムの特徴

実現確率探索や強いプレイヤの棋譜を用いた評価関数の最適化などのオーソドックスな手法を採用しているほか、様々な高速化の工夫を施しています。

昨年からの変更点

昨年の世界コンピュータ将棋選手権からの主な変更点は以下の通りです。

- 並列探索アルゴリズムの改善

  昨年同様 YBWC をベースとした並列化を行っていますが、ルートにおいても split するように変更しました。また、helpful master concept を実装しました。

- 局面評価関数の改善

  指し手だけでなく、勝敗も教師データとして利用するようにしました。

- 探索アルゴリズムの改善

  残り深さの少ない局面での late move reduction の実装や、指し手の遷移確率計算に用いる特徴の変更などを行っています。

2018 年第 28 回世界コンピュータ将棋選手権アピール文書

GPS 将棋は，東京大学大学院総合文化研究科の教員・学生が開催しているゲームプログラミングセミナー（Game Programming Seminar = GPS）のメンバーが中心となって開発が行われているソフトウェアです．フリーソフトウェアとしてソースコードやデータを公開しています．

2003 年から GPS 将棋として世界コンピュータ将棋選手権に参加し，2009 年の第 19 回及び，2012 年の第 22 回世界コンピュータ将棋選手権で優勝した他，2010 年の第 20 回及び，2013 年の第 23 回世界コンピュータ将棋選手権では 3 位の成績を得ています．また，清水女流王将（当時）とコンピュータ将棋（あから 2010）との対局では，激指，Bonanza, YSS とともに GPS 将棋も参加しました．2013 年に行われた第 2 回電王戦において三浦弘行八段（当時）と対局しました．

2018 年の第 28 回世界コンピュータ将棋選手権は GPS 将棋として 16 回目の参加となります．

技術的な特徴としては，コンピュータチェスやコンピュータ将棋の最新の研究を取り入れていることが挙げられます．例えば，利きを管理する高速な将棋盤，実現確率を用いた探索，評価関数の自動学習などがあります．また評価関数は現在，序盤，中盤 1，中盤 2，終盤の 4 種類を用いています．他にも，疎結合並列探索や df-pn（並列協調）を用いた詰探索にも対応しています．技術的な詳細は参考文献をご覧下さい．

Team GPS

参考文献・ウェブサイト

- WWW サイト: `http://gps.tanaka.ecc.u-tokyo.ac.jp/gpsshogi`
- 多数の計算機を活用したゲーム木探索技術の進歩 -三浦弘行八段と GPS 将棋との対局を振り返って-, 金子知適, 田中哲朗, 情報処理 54(9), 914—922, 2013. : `http://id.nii.ac.jp/1001/00094757/`
- コンピュータ将棋の新しい波: 3. 最近のコンピュータ将棋の技術背景と GPS 将棋金子知適, 情報処理 50(9), 878—886, 2009.: `http://id.nii.ac.jp/1001/00067191/`
- S. Yokoyama, T. Kaneko, and T. Tetsuro: Parameter-Free Tree Style Pipeline in Asynchronous Parallel Game-Tree Search, The 14th International Conference on Advances in Computers and Games (ACG2015) : `http://www.graco.c.u-tokyo.ac.jp/~kaneko/papers/acg2015-yokoyama.pdf`
- Twitter: `http://twitter.com/gpsshogi`
- Floodgate: `http://wdoor.c.u-tokyo.ac.jp/shogi/`

第28回世界コンピュータ将棋選手権

スーパーうさぴょん２ アピール文書

==========

2018.3.31 うさぴょんの育ての親

●目標

・AMD製CPU1位の座を取り戻す！

●使用ライブラリ

・なのはmini

・Apery

●プログラム全体

スクラッチから「うさぴょん３」として開発していたものが行き詰って（というか仕事と並列だと十分な時間が確保できなかった…）昨年の「うさぴょん２′TURBO」に追加開発を行うこととしました。その為、名前は見る人が見れば昨年からの連番になっています（苦笑）。

その為、コードのベースには以前と変わらず「なのはmini」を利用しています。

探索部はStockFishを参考にしつつ、自己対局で効果がマイナスだったり、効果が認められなかった枝刈をいくつか外したり、自前で考えてみた枝刈を追加してみたりしています。（自前の枝刈は効果がプラスになったものがまだありませんが…。）

●定跡

今回は「なのはmini」の定跡部は使いません。新たに定跡部だけコードを含めて書き起こしています。なお、定跡は０ベースで作成中です。定跡に少し力点を置いて開発しているのですが、公開されている評価関数の中では「Apery」の最新のものが

強い＆準備中の定跡との相性が良いようです。

その為、今回の大会では「Apery」の評価関数を用います。

定跡の話は細かく書くと他の開発者に対策を練られそうなので思いっ切り割愛させていただきます…が、驚かれるような事はやっていません、とだけ書いておきます。

「習甦」

特徴：多層構造を持つ評価関数

・駒の価値：盤面全体の駒の利きと持駒および手番から算出する玉の安全度に対応した非線形関数

・駒の働き：入玉していない玉と玉以外の２つの駒の位置関係

→入玉して玉の安全度が高くなると、持駒と敵陣にいる駒が小駒：大駒＝１：５の駒割に近づく。

評価パラメータの機械学習方法

・前世代の評価関数を用いてフィッシャーランダムチェスに準じたユニークな初期局面2494800からの自己対戦棋譜を数セット作成する。

・自己対戦棋譜における評価値の推移をフィードバックして推定された勝率を割引報酬として強化学習する。

第 28 回世界コンピュータ将棋選手権アピール文書

概要

「たこっと」は，電王戦を見てコンピュータ将棋に興味を持った筆者らが， フルスクラッチで実装した(している)将棋プログラムです。 Web 上の解説記事や論文，Stockfish, Apery, やねうら王, Bonanza 6.0 のソースコードを参考にしています。

特徴

- 各種処理が AVX2 命令で実装されていて高速
- AVX2 命令を適用しやすいデータ構造 (非ビットボード)
- Stockfish 風の探索アルゴリズム
- オンライン学習ルーチン

以前のたこっとから引き継いでいる特徴については過去のアピール文書を参照してください。

- 第 26 回世界コンピュータ将棋選手権アピール文書
- 第 27 回世界コンピュータ将棋選手権アピール文書
- 第 4 回将棋電王トーナメント PR 文書
- 第 5 回将棋電王トーナメント PR 文書

ライブラリの使用について

以下のライブラリの最新バージョンを使用申請しています。

- **Apery**
  教師データの生成や強化学習時の初期値として評価関数バイナリのみを使用する予定
  強いということが選定理由
- やねうら王 コンピューター将棋フレームワーク
  現時点では使用するかどうか未定
  必要なモジュールがそろっており，ソースコードを読み込んでいて慣れ親しんでいるのが選定理由
- **elmo**
  現時点では使用するかどうか未定
  Ponanza を倒したという実績が選定理由

ライブラリを提供していただき，Apery の平岡様，やねうら王の磯崎様，elmoの瀧澤様には感謝いたします。

学習ルーチン

以下に挙げるような近代的な機械学習の機能を一通り実装してあります。

- 確率的勾配降下法
- オンライン学習
- ミニバッチ
- 複数の損失関数
- 複数の最適化手法
- 正則化

第 5 回将棋電王トーナメントで elmo 方式の学習アルゴリズムを実装し，コンピュータ将棋界のトレンドにも追従しています。

評価関数に KPPT 型の三駒関係のモデルを採用していますが，近年は強化学習しても棋力の向上が微々

たるものになってきました。 大量の学習データの作成と学習を何度も繰り返す必要がありますが，計算資源の確保に苦慮している貧乏プログラマーには厳しい限りです。 そこで学習で棋力向上が望めるような新たなモデルの開発に取り組むことにしました。

が．．．
今のところ結果がついてきません。

あまりに強くならない場合はライブラリの評価関数バイナリを強化学習し，大会に参加するかもしれません。

CGP　アピール文章

2018/3/30
大熊　三晴

主な特徴

・無駄に一から作成
・非ビットボード型
・無駄に高 **NPS** を目指してるけど最近この部分はさぼり気味
・強くするならならライブラリ不使用前提でも評価関数から手を入れていくべき状態だけど
　無駄に探索をいじっていました。
・局面構造体に各マスへの利きの状態を保持
・局面構造体に評価関数の演算途中結果のうち変化の頻度が少ないものを中心に保持
・評価関数も自力で学習(結果は駄目駄目です)
・**AVX-512** 命令をはじめとした拡張命令を活用
・コンピュータ将棋では一般的にはあまり使われていない機能を使用
・今までの大会は **WCSC**、電王トーナメント計 **5** 回すべてで毎回勝率 **5** 割
・今のところ結果的に昨秋からはあまり変更できていません。
・大会までにはアピール文書を書き直すくらいに開発が進んでて欲しいです。
・一般に流布している定跡データや、一般に流布している「局面と評価値のセット」、
　読み筋等は使用しておりません。無駄なこだわりだとは思いますが。

・1から作成

強さをあまり考えずに高 **NPS** を目指して自作したプログラムをベースとしております。
並列化手法は現在は **LazySMP** です。

・非ビットボード型，利き等を保持

非ビットボードだとビットボードに比べ遅くなる処理もありますが、複雑な情報を持てることにより速く処理できる可能性もあります。ビットボードに比べ遅い処理をうまく避けるために利きを保持したり、局面構造体の配置をビット位置を含めて工夫しております。**AVX-512** でかなりの並列化が出来そうですがまだ **AVX2** を使っていた部分からの置き換えくらいしか出来ておりません。

また利きの保持以外にも、演算途中のデータを保持することによりメモリアクセス待ち時に演算を回す事により高速化を狙っております。

・評価関数

現在は手番付き **KPP** です。評価関数テーブルは駒割＋入玉時の位置評価のみの初期値から自力で機械学習したものですが、学習の仕方が古いので精度はかなり悪いです。評価関数の拡張は考えてますが無駄に探索部を頑張ったせいで手が回っておりません。

・SIMD 等の活用

高速化のため SIMD を活用しております。SIMD は現在評価値の算出、オーダリング、構造体のコピーが主な使用箇所です。

オーダリングの一部は VPMAX 命令で次に試す手を抜き出す方式を取っております。この方式は条件分岐なく複数の手を比較できるため、挿入ソートを通常の x86 命令で行うより高速化できております。SIMD を用いたソートの使用は試していないのでこちらのほうがより高速になる可能性もあります。

また置換表や評価値テーブルのページテーブルにラージページ(Windows での言い方 Linux 用語だと Huge　Page)を使用し、高速化を図っております。（1GB サイズのページテーブルも確かめてみたいけど扱える OS があるのかすら未調査）

==========

2018年

柿木将棋のアピール文章

今年は昨年と殆ど変わりません。そのため、昨年のアピール文章も添付します。

1985年頃に開発を始めました。
前向き枝刈を行う選択探索で、アルファβ法を使っていました。
評価関数は、手作業で作成・調整していました。

2007年、ボナンザの影響を受け、全幅探索を行い、ボナンザ法で学習した評価関数を使うプログラムを新しく作成しました。

2008年から2012年までは、従来の選択探索のプログラムと新しい全幅探索のプログラムを組み合わせていました。

2013年、新しい全幅探索のプログラムだけとしました。
次のような手法を使っています。

・全幅探索
・局面の評価関数は、ボナンザ法で学習
・利きデータを使用し、bit-boardは使っていない。
・局面の評価項目は、独自のもの
・山下さんの 0.5手延長方式を採用
・並列化 PVS
・NULL move 枝刈
・王手延長
・静止探索

参加する意味はあまりありませんが、ほぼ同じソフトを同じハードで参加しているので、
そういう意味のベンチマークはなるかと思います。

2018/03/25 柿木

==========

2017年
柿木将棋のアピール文章

今年は昨年と殆ど変わりません。そのため、昨年のアピール文章も添付します。

1985年頃に開発を始めました。
前向き枝刈を行う選択探索で、アルファβ法を使っていました。
評価関数は、手作業で作成・調整していました。

2007年、ボナンザの影響を受け、全幅探索を行い、ボナンザ法で学習した評価関数を使うプログラムを新しく作成しました。

2008年から2012年までは、従来の選択探索のプログラムと新しい全幅探索のプログラムを組み合わせていました。

2013年、新しい全幅探索のプログラムだけとしました。
次のような手法を使っています。

・全幅探索
・局面の評価関数は、ボナンザ法で学習
・利きデータを使用し、bit-boardは使っていない。
・局面の評価項目は、独自のもの

・山下さんの 0.5手延長方式を採用

・並列化 PVS

・NULL move 枝刈

・王手延長

・静止探索

2017/03/28 柿木

追記

選手権前に floodgate で、しばらく対戦させたところ、レーティングは 約2300 です。

2014/8/30 には 2234 でしたが、その後、殆ど改良はしてなくて、ハードも同じです。

2017/04/26 柿木

==========

2016年

柿木将棋のアピール文章

今年は昨年と殆ど変わりません。そのため、昨年のアピール文章も添付します。

変わる予定なのは、次の点だけです。

・フィッシャールールに対応（予定）

2016/03/26 柿木

==========

2015年

柿木将棋のアピール文章

今年は昨年と殆ど変わりません。そのため、昨年のアピール文章も添付します。

変わったのは、次の点です。

・秒読みルールに対応

・バグの修正

・評価関数を少し改良したつもり

昨年の選手権直後の 2014/5/7、floodgateでのレーティングは 2177 でした。

上記改良を行い、2014/8/30 には 2234 になったので、57 上がりました。

その後は、改良できなかったので、この 1 年の改良点は、これだけです。

2015/04/02 柿木

==========

2014年

柿木将棋のアピール文章

今年は昨年と殆ど変わりません。そのため、昨年のアピール文章も添付します。

変わったのは、次の点です。

・バグの修正

・評価関数を少し改良したつもり

2014/03/26 の時点のfloodgateでのレーティングは、2178 と昨年とほぼ同じです。

2014/03/26 柿木

==========

2013年

柿木将棋のアピール文章

今年は、去年と大きく変わりました。

昨年まで、全幅探索のプログラムと選択探索のプログラムの２種を組み合わせていました。
今年は、全幅探索のプログラムだけにしました。ようやく、その方が強くなったからです。

全幅探索のプログラムは、2007年に新しく開発を始めたものです。
ボナンザの影響を受け、全幅探索を行い、評価関数はボナンザ法の学習で作成しているものです。
ただし、ボナンザとは色々違っています。
例えば、bit-boardは使わず、利きデータを使っています。
評価関数の評価項目は独自のものです。

floodgate では、今年のプログラムは昨年のプログラムに対して、７割程度の勝率があります。
ただし、floodgateでのレーティングの点数は、後述するように、少し不可解な点があります。

昨年のプログラムは、昨年のfloodgateでの点数は 2145点、
ほぼ同じプログラムが今年は、2089点と少し下がりました。

全幅探索のプログラムは、昨年に対して、次の改良を行いました。

１．並列化
２．山下さんの 0.5手延長方式を採用
３．評価関数の調整

全幅探索のプログラムのfloodgateでのレーティングは、改良前 2094点、
改良後 2184 点と 90点上がった程度です。

2013/04/18 柿木

==========

『ひまわり』

ひまわり 開発者
山本 一将，永塚 拓，高木 厚成

ひまわりについて

- 2012年頃から作っている将棋ソフト
- メイン開発者は2012年まで『芝浦将棋』の開発者として大会に出場
- バグが多かったのですが、第２回電王トーナメントから成績が向上
- 38時間かけて△２八角を回避した事で有名?
  - https://twitter.com/kyamamoto9120/status/589357152663248896?ref_src=twsrc%5Etfw

WCSCでの成績

- 第27回世界コンピュータ将棋選手権 22位
- 第26回世界コンピュータ将棋選手権 19位
- 第25回世界コンピュータ将棋選手権 9位
- 第24回世界コンピュータ将棋選手権 31位
- 第23回世界コンピュータ将棋選手権 23位

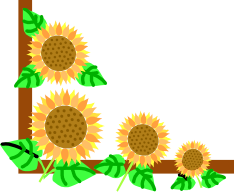
その他の成績

- 第3回電王トーナメント ベスト8
- 第2回電王トーナメント 12位
- 第1回電王トーナメント 18位

- 魅力的な将棋AIコンテスト プレマッチ 優勝

特徴

- Bonanzaメソッドを使用していない
  - 方策勾配を用いた教師有り学習
- 教師付学習後に強化学習している
  - 自己対局の結果（勝ち負けなど）から学習



前大会からの変更点

- AlphaZeroのアルゴリズムで新規に作成する予定でしたが、時間が取れずに間に合いませんでした。
- ほぼ前回大会のプログラムで参加します

「芝浦将棋 Softmax」のチーム紹介

2018 年 2 月 19 日
芝浦工業大学情報工学科
五十嵐治一，村松昌，原悠一，古根村光，横田直之，吉谷和人

1. はじめに

本稿は，第 28 回世界コンピュータ将棋選手権（2018 年 5 月 3 日～5 日開催）に出場予定の「芝浦将棋 Softmax」（シバウラショウギ　ソフトマックス）のアピール文書です．本チームは昨年に引き続いて 2 回目の出場です．本チームの原型は 2016 年まで出場していた「芝浦将棋 Jr.」チームですが，探索手法が従来の Min-max 探索（α β探索）とは異なる Softmax 探索である点が大きく異なります. ただし, 合法手生成までは芝浦将棋 Jr.と共通で, 選手権公式ライブラリとして登録されている「芝浦将棋 Jr.合法手生成プログラム」[1]を使用しています. 棋力的には従来の探索手法のチームにはまだまだ及びませんが, アルゴリズムが単純でコーディングの容易さや並列性に優れています. 以下, 簡単に本チームの特徴を紹介していきます.

2．開発メンバー

五十嵐は芝浦工業大学工学部情報工学科に勤務する教員です．村松，原，古根村は五十嵐研究室の卒業生で，横田と吉谷は学部 4 年生です.

3.「芝浦将棋 Softmax」の特徴

本チームの特徴を，以下の１）～５）のようにまとめました．このうちの２)～４）が本チーム独自の探索方式である「MCSoftmax 探索」[6]に関する説明です．この探索方式は, 文献[2]の研究が基になっています.

１）芝浦将棋 Jr. の合法手生成ルーチンを使用

芝浦将棋 Jr. では盤面表現のデータ構造を独自の Magic bitboard を用いて駒の利き場所での駒の配置状況などを計算しています[3]．この計算を含む合法手生成のプログラムは「芝浦将棋 Jr.合法手生成プログラム」の名称で選手権公式ライブラリとして登録されています. 芝浦将棋 Softmax はこの合法手生成プログラムをそのまま使用しています.

２）Softmax 探索を使用

現在のチェスや将棋のプログラムは「Min-max 探索」という探索方式をほぼ 100%採用しています．これには探索木のすべてのノードを探索する必要がありますが，α β カットなどの枝刈りの処理により探索にかかる計算時間を短縮しています．これに対して，そのゲーム特有の知識（ヒューリスティクス）を用いて探索するノードを限定したり，優先順位をつけて選択的に探索する「選択探索」という探索方式があります．本チームはノードの選択方式としてノード評価値の min-max 演算ではなく，確率分布に基づく選択（Softmax 探索）を使用しています．したがって，探索木をルートノード（実際の盤面の局面）から選択して降りていく（読んで行く）際には，実際にサイコロをふりながら確率的に選んで末端局面まで降りていきます．この確率的選択方式は，AlphaGo のようなコンピュータ囲碁ソフトで用いられているモンテカルロ木探索における決定論的な木の選択方法（UCT など）とは一線を画しており，本チームの大きな特徴と言えます．

３）ノードの評価関数を用いたボルツマン分布による確率的なノード選択

前項で述べた Softmax 探索には，本チームでは指し手の良さを用いたボルツマン分布を利用します．すなわち，各ノードでの指し手の選択確率を次の式で計算し，その確率に従ってノードを選択していきます．

$$\pi(a|s) = exp(E_a(a;s)/T) / \sum_{x \in A(s)} exp(E_a(x;s)/T) \qquad (1)$$

ただし，s は局面（ノード），a は指し手，$E_a(a;s)$は局面 s における指し手 a の良さですが，指した後の局面ノードの評価値 $E_s(s)$で置き換えることにします．$A(s)$は s における合法手の集合，T は温度と呼ばれているパラメータです．温度が低ければ最良優先探索に，温度が高ければランダム探索に近づきます．ノードの評価値は，探索木の末端ノード（leaf）であればそのノードの局面評価関数により計算します（実際にはそこで静止探索も行っています）．

一方，内部ノードであれば子ノード $v(x;s)$の評価値 $E_s(v)$をその子ノードの選択確率 $\pi(x|s)$で重みづけた期待値

$$E_s(s) = \sum_{x \in A(s)} \pi(x|s) E_s\big(v(x;s)\big) \qquad (2)$$

で定義します．したがって，読んだ先（子孫ノード）に評価の高い手があるような手は高く評価されます．また，十分探索が進んで探索木をすべて展開した後では，(1)で T をゼロに近づける（低温化）と，Softmax 探索による探索結果は Min-max 探索の探索結果に近づいて行きます．

４）深さ制御とバックアップ操作

探索の全体の流れを図１に示します．ルートノードから，３）の選択法に従ってノードを選択し，末端ノードまで到達すると，そのノードの子ノードを一段階だけすべて展開します．展開後は新たな末端ノードの評価値を局面評価関数で計算し，その値をルートノードへ向けて(2)の計算を繰り返し，ルートノードまでの経路上のノード評価値を更新していきます．我々はこの更新操作を「バックアップ操作」と呼んでいます．

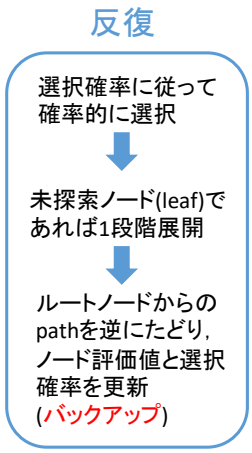


図１

我々は，上記２）～４）に述べた探索方式を“Randomized Softmax Search”または“Monte Carlo Softmax Serach”（MCSoftmax 探索）と呼んでいます[6]．この名前の由来は，ルートノードから末端ノードへ到達するまで，(1)の選択確率に従って確率的にノード選択を行って経路が生成される３）の過程は，指し手の良さ（＝その手の子ノードの評価値の期待値）を求めるためのモンテカルロ・サンプリングに相当するからです．

上記のモンテカルロ・サンプリングを一定回数あるいは一定時間行った後，確率値の最も高い子ノードだけを次々に選択して得られた手順を最善応手手順であると決定します．

今回のバージョンでは深さ制御のために特別な処理を何もやっておりません．しかし，(1)の選択確率の値を用いて，決定論的な最良優先探索を行う探索法も考えられます．これは選択確率の積を実現確率と定義し，実現確率の閾値を深さとする反復深化法と結びつけることができます[4]．この方式も実装して対局実験を少しだけ行ってみたのですが，上で述べた選択確率に基づく確率的なサンプリング方式の棋力が上で，並列処理の効果も高かったので，本チームでは 2017 年の初参加のときから採用しています．

５）評価関数について

現在のところ，評価関数の特徴量は，選手権公式ライブラリである Bonanza (Ver. 6.0.0) [5] のものをそのまま使用しています．Bonanza は「Bonanza メソッド」と呼ばれる教師付学習方式が有名です．我々の研究グループは，この方式をより一般化した「方策勾配を用いた教師付学習法」を提案しています[7]．通常の教師付学習では，棋譜の着手を正解手として，この正解手の情報だけを用います．[7]の学習法では，正解手以外の手の評価値も学習データとして利用することが可能です．大会までに間に合えば，Bonanza6.0 の評価関数パラメータではなく，自分たちで学習した評価関数パラメータを準備する予定です．

さらに，３）で述べたモンテカルロ・サンプリングで生成された探索木において，全 leaf に出現する特徴量の重みを探索時と同様なモンテカルロ・サンプリングとバックアップ操作だけで学習することが可能です[6]．将来的にはこの学習法も実装していく予定です．また，上記のような教師付学習だけではなく，報酬の最大化を目的とする強化学習（TD 法や

方策勾配法），勝敗の予想確率を学習する回帰法，深い探索結果を利用する Bootstrap 法（RootStrap 法や TreeStrap 法）も，Softmax 探索とモンテカルロ・サンプリングの組合せで実行することが可能です．これにより，最善応手手順だけでなく，有力変化手順の近傍局面に出現する特徴量パラメータも，その重要度に応じて積極的に学習できるので，学習の精度や速度の向上に繋がると期待しています．

なお，末端ノードでの局面評価には静止探索（駒の取り合いだけを考慮する探索）を行って，その結果を局面評価として返す処理を行っています．現バージョンのプログラムでは，この静止探索においては高速化のために従来のαβ探索を使用しています．

4. 今後の課題

今年は 36 コア(72 スレッド)のワークステーションを使用する予定です．各スレッドが探索木を共有し，図 1 に示した処理を独立に行っています．しかし，有力手順を重点的に探索するためにはスレッドの割当方法を工夫する必要があります．また，各スレッドが同じ温度パラメータを持って枝を選択して降りて行く必要性は必ずしもありません．選択時に温度の高いスレッドや低いスレッドがあって，広く浅く探索するスレッドや狭く深く読むスレッドなどのバリエーションを持たせることも可能と考えています（ただし，バックアップ時の温度は一定としておく）．

さらに，親ノードとそれ以下の探索木とをスレッドに分散して割当て，完全な並列分散化処理を行うことも可能です．上記のスレッド割当と探索木の分割処理とをうまく動的に行うことが今後の課題の一つです．今のところ，最善応手手順や有力手順の近傍を中心に，スレッドと探索木を探索途中で動的に割り当てることを考えています．

また，MCSoftmax 探索方式は，ニューラルネットワークモデルによる評価関数表現と非常に相性が良いとされています．実際，2017 年 11 月開催の第 5 回将棋電王トーナメントでも mEssiah というチームが早速採用してくれました．今後，ディープラーニングを用いた学習方式がコンピュータ囲碁だけではなく将棋へも波及して来ると予想されます．mEssiah の開発者の話によれば，ニューラルネットワークモデルによる評価値計算にはかなりの時間的コストがかかるが，GPU などを用いると多くの局面の評価値計算を一度に並列化して計算することができ，例えば，図 1 における子ノードの一斉展開と評価値計算には適しているとのことです[9]．このように，局面評価関数としてニューラルネットワークモデルを用いることも本チームの今後の課題の一つです．

5. おわりに

現在のコンピュータ将棋プログラムの多くは，探索方式（Minimax 探索の高速版であるαβ探索）からソースコードのレベルまで，Stockfish[8]などのチェスプログラムから大きな影響を受けています．それに対して，本チームは Softmax 探索とモンテカルロ・サンプリングをベースにしています．本探索方式は囲碁プログラムで用いられているモンテカルロ木

探索の一種と思われますが，プレイアウトを行わない点や，確率的選択を行っている点が異なっています．また，本探索方式は，プログラム作成が容易で，並列化の効果も高い上に，他のゲームプログラムへの適用も容易であるという点で汎用性にも優れていると考えています．

まだまだ問題点も多いのですが，新しい探索方式と学習方式を研究する上では面白さが多く，開発者自身，今後の展開を楽しみにしております．最終的には，プロ棋士の棋譜を用いることなく，コンピュータ自身が自己対局を（あるいは他者との他流試合も）通して，探索法や局面評価関数を学習し，人類の棋力を超えて，新しい定跡や戦法を創出し，棋士や将棋ファンを大いに楽しませてくれることを目標としております．

参考文献

[1] 「芝浦将棋 Jr.合法手生成プログラム」の機能説明書とプログラムは次のページからダウンロードできます：http://www2.computer-shogi.org/library/

[2] 五十嵐治一，森岡祐一，山本一将，“方策勾配法による静的局面評価関数の強化学習についての一考察”，第 17 回ゲームプログラミングワークショップ 2012 予稿集, pp.118-121 (2012).

[3] 例えば，http://www2.computer-shogi.org/wcsc26/appeal/Shibaura_Shougi_Jr./appeal.pdf に記載されています．

[4] 原悠一，五十嵐治一，森岡祐一，山本一将，“ソフトマックス戦略と実現確率による深さ制御を用いたシンプルなゲーム木探索方式”，第 21 回ゲーム・プログラミング・ワークショップ 2016 予稿集，pp.108-111(2016).

[5] Bonanza のホームページ，http://www.geocities.jp/bonanza_shogi/

[6] 桐井杏樹，原悠一，五十嵐治一，森岡祐一，山本一将，“確率的選択探索の将棋への適用”，第 22 回ゲーム・プログラミング・ワークショップ 2017 予稿集，pp.26-33（2017）.

[7] 古根村光，山本一将，森岡祐一，五十嵐治一，“方策勾配を用いた将棋の局面評価関数の教師付学習：静止探索の導入と AdaGrad の適用”，第 22 回ゲーム・プログラミング・ワークショップ 2017 予稿集，pp.1-7（2017）.

[8] Stockfish のホームページ，https://stockfishchess.org/

[9] コンピュータ将棋ソフト mEssiah の内部構造，https://qiita.com/sakuramaru7777/items/ebb397eef94fc02be2d8

まったりゆうちゃんのアピール文書 2018

1990 年過ぎから開発を始めた。4 半世紀にわたって開発しているシステムである。当初のコードもたくさん含んでいる。

完全独自開発であり、他のシステムを参考にしていない(考え方は参考にしている)。アイデア的にも独自工夫をしている。

今日、AI というとディープラーニングをはじめとしてパラメータ学習に基づくものが多い。コンピュータ将棋での駒価値学習もそうである。しかしそうでない進化論的計算などの方式を試そうとしている。またディープラーニングと多量パラメータ学習の中間的なメカニズムを考えたいと思っている。

いまのところ、去年からみてあまり進んでいないが、ある程度改良を進める予定である。

開発者の年齢が高いということで、希少価値があると思う。可能な限りやめないで続けたいと思っている。コーディング能力がいつまで続くか試したい。

去年は久しぶりに一次予選を通過した。それが定常的になるようにしたいと思っている。

名人コブラアピール文書

生粋のライブラリ勢

過去の経験から、ポンコツプログラマな自分がプログラムを改造すると大幅な低速化を招くことがわかりました。ですから、今回も探索部や評価部には手を付けないことにしました。低速化をしても直接的には読みのスピードに影響の無い、学習部だけに手を加えたいと思いいます。

使用ライブラリ

- Apery
  - 評価関数だけならば、トップを走っているといわれているので、学習部を改造して使用させていただきたいと思います。キメラの素として、評価関数も使用させていただきます。
- やねうら王
  - 現在探索部は最強だといわれているので、探索部を使用させていただきたいと思います。また学習部も改造して使用させていただきたいと思います。キメラの素として、評価関数も使用させていただきます。
- elmo、人造棋士 18 号
  - Apery-やねうら王系統の評価関数を使用しているため、キメラの素として、評価関数を利用させていただきます。

評価関数のキメラ化にこだわります

評価関数をブレンドすると強くなるらしいので、少なくとも 20 個の評価関数を作って、それを混ぜたいと思います。お手軽だし、「アンサンブル学習」という言葉やコンセプトがなぜか大好きなのです。

キメラ素材の調達方法

1. ライブラリの評価関数

   ある程度実績のある評価関数を使用させていただきたいと思います。

2. ゼロから新たに学習させた評価関数

   同じものを混ぜても、キメラ化の効果はありません。バリエーションが必要なので、次の方法でバリエーションを持たせたいと思います。

   - 教師局面の復元抽出
   - 特徴の部分的使用（ランダム部分空間）

教師局面と評価値

キメラ素材となる評価関数の学習には以下の教師局面と評価値を使用します。

1. Apery を使用して生成したもの
2. やねうら王作者の磯崎氏が配布しているもの

最後に

ライブラリ勢なのに今まであまりライブラリを使いこなせていなかった感じがします。今回はあまり余計な改造はせず、ライブラリを使いこなして、良い評価関数を作ることに重点を置きたいと思います。

日本将棋連盟TOP ＞ 電棋士データベース ＞ カツ丼将棋

電棋士データベース

五段 (将棋倶楽部24基準、R2400)

桜井昇八段門下 カツ丼将棋

Katsudon Shogi

電棋士番号	-9999
生年月日	2013年4月20日(4歳)
出身地	東京都千代田区
開発者	松本 浩志
師匠	桜井昇八段
評価関数	簡易4コマ(予定)
探索	カツ丼フィッシュ

前回からの進化した点

カツ丼将棋がその功績を師匠の桜井昇八段に認められて、桜井昇八段門下の一員となりました。惜しむらくは棋力は 1 ミリも進化してません。。

評価関数

簡易 4 コマ(予定)です。簡便であってもとりあえず 4 コマにしておけばいいかなと思ったのですが、肝心の棋力が弱くなっているので、本番までに調整します。間に合わなければ 2 駒で。

簡易 4 コマとは、まず将棋ソフトは 2 駒でもそこそこ強くて軽くて扱いやすい。3 駒はポテンシャルが高いが玉の差分とか面倒である。そこで、ルートにおける玉玉の位置で呼び出す 2 駒を決めてしまうイメージ。玉玉の位置関係は 81x81≒1600 通り。これを真面目に考えると容量が莫大になるため、将棋盤を 3x3 で 1 ブロックにしてやることで 81 通り。これで容量・計算は現実的なものとなり、一応 KKPP ということで俺は 4 駒だーと言える。

+手調整を加えています。手調整何てはやらない時代ですが、将棋というゲームで勝利に近づくには以下の要件が考えられます・・・(①相手より駒得をする。②敵玉に迫る)。教師局面が少ないと、②がうまく学習されません。しかし我々は②の敵玉に迫れば勝利に近づくと知っているので、手調整で強制的に加点してやります。あとは探索先生におまかせしておけば自己対戦では強くなった気がしました。

探索

StockFish 風の探索に実現確率的なのを取り入れようとしたり、いろいろ努力をしています。が、一つもうまくいったものがないので、アピールするに至っておりません。。

その他

将棋倶楽部 24 における自動対局システム「カツ丼坊」を開発しました。これは画像認識・マウス操縦を駆使して汎用将棋 USI を将棋倶楽部 24 に 24 時間延々と自動対局ができるようにするシステムです。このシステムをフルに活用して、将棋倶楽部 24 のレートとソフトレートの直接比較を行いました。その詳細な仕組みや結果は、2018 年 3 月発行の CSA 会誌に掲載しておりますのでぜひご覧ください！。

レートを計るということ以外にも、将棋ソフトと人間がたくさん対局をして交流を深めるという形で将棋界に貢献ができたと思っています。

将棋倶楽部 24 の今年に入ってからの戦績(自由対局ですが。。。)

2850 勝 478 敗

勝ち負けはどうでもよくて、こんだけ対戦させてもらったことが何よりの財産です。

最後に

永遠のネタ勢として楽しんできます。

年表

2013 年 4 月　第 2 回電王戦　Ponanza VS 佐藤慎一四段戦でえりりんから客いじりで「カツ丼さん」と呼ばれて将棋ソフトを「カツ丼将棋」に決める。

2013 年 10 月 第 1 回電王トーナメントデビュー
初陣は Ponanza。ぼこぼこにされる。
カツ丼定跡で古豪スケルツォを破り念願の初勝利
永遠のライバル Labyrinthus に恥ずしめ詰めされる。

2014 年 5 月　選手権で全敗。Libshogi に全駒ステイルメイトされて Wikipedia にも

掲載される。

2016 年 5 月　選手権でブービー。さすがに恥ずかしかったので少し強くすることを決意

2016 年 10 月　少々強くしたが、電王 T で竜の価値を 0 点にしてしまううっかりで惨敗。
メカ女子将棋に一手詰めのバグのうっかりで敗北。

2017 年 1 月　将棋倶楽部 24 で R2500 弱の成績を収め、五段に昇段。

2017 年 5 月　世界選手権一次予選で 12 位で次点。

いつものおまけ。

●毎度のことで恐縮なのですが、将棋の歌が少ないと思って将棋の演歌を作曲しました。

曲名　千駄ヶ谷エレジー

歌手　長沢千和子女流四段

作詞　袋小路宇治夫

作曲　カツ井将棋

日本将棋連盟推奨、桜井昇八段推薦

もしよかったらどうぞ。最近は長沢先生が老人会に遠征したりしているそうです。

Itune store、レコチョクから DL できます。itune store から購入可能）

Novice PR文書

Noviceとは

2015年より開発を行う

第三回電王トーナメントから5大会連続最年少出場
（当社調べ）

前回からの変更点

過去のコードを全て捨てました

0からのスタート

Stockfish 9に沿って作成

横型のRotated Bitboard を採用

実装

(結局、高速1手詰入れました)

王手周りは0.5手延長や枝刈りを甘くしている

PolyglotBook というチェスと同様の定跡形式を採用

StockFishの探索から将棋に向いてなさそうなものを除去

指し手のオーダリングも将棋向けに調整

ライブラリ

AperyのKPP.bin/KKP.binをお借りします

自作のConverterを通して横型にしたものを使います

評価バイナリ以外は自作です

選定理由として最新の探索部と自己の探索部との比較が

行いやすいことが挙げられます

ハードウェア

とりあえず初日は友人から借りたi9 7900Xを使います。

意気込み

そろそろ最年少じゃなくなりそうなので結果を出したいです

雑感

大学も折り返しが見えてきて

特になりたい職もなく

人生いろいろ悩みます

（ここに書くな）

■きのあ将棋について

○作成 :2018/03/31 ...過去のものを元に作成。

現在(2018/03/31)、

2018年の将棋思考エンジンの研究は環境を整えただけでまだ進めてないです。。

ですので、現状の特徴や昨年までの模索、ならびに模索予定の内容を記します。

●きのあ将棋の特徴

1手ごと思考エンジンを実行し、次の着手を思考する方法を採用しています。(2011年採用)

・この方法のメリット

→１台のマシンで沢山の相手と対局する際にて、コンピュータ資源の節約。

→思考プログラムの実行を管理しやすくなる。など。。

・この方法のデメリット

→前回の思考時のハッシュの利用ができなくなる。

→相手の思考時間を有効利用できない。など。。

●使用ライブラリなど

現在のところ過去においても将棋ライブラリの利用ならびに利用予定なし。

他プログラム(手作成も含め)の評価値や読み筋、定跡データなどの利用もなし。

プロ棋士の棋譜データの利用のみあり。(椿原チーム様から。椿原チーム様ありがとうございます)

とはいえ研究開発を汎用化/効率化の目的から、きのあ将棋/囲碁などで共通化する仕組みを模索。

(過去のPR文章にも書いた記憶)

ですので昨年2017年から、

きのあ将棋と囲碁でコピペで使いまわしていたところをライブラリ化をすすめています。
2018年wcscでは実使用を間に合わせたいです。

●最近の模索：評価について
ここ数年、加算連結する評価処理、乗算連結する評価処理をしたものを、
抽象化しくりかえし処理をさせる評価関数を作成。これを機械学習する模索をしていました。

↓抽象例
加算連結の評価 = (A[i]+B[i]+C[i]+D[i]) + (E[i]+F[i]+G[i]+H[i]) + (I[i]+J[i]+K[i]+L[i]) …
乗算連結の評価 = (A[i]+B[i]+C[i]+D[i]) * (E[i]+F[i]+G[i]+H[i]) * (I[i]+J[i]+K[i]+L[i]) …

ながらく乗算連結の方に期待(表現力が高いと考えていた)していたのですが、
結局は加算連結より機能しませんでした。
原因は機械学習時のパラメータ値が発散するためだと思われ、乗算連結は断念する方向。
(ですので今年は昨年までと同様に加算連結を採用予定。)

今年は加算連結のみですすめる代わりに評価関数の機械的構築を実現したいと考えています。

●最近の模索：機械学習について
評価パラメータにノイズを加味する際において、
「2重の乱数方式」で行っていたのを「2の倍数固定値方式」に変更しました。

```
// 2重の乱数方式
for( g_noise = 0; g_noise == 0; ) { g_noise = ((QiRand_rand()%17) -8) +((QiRand_rand()%17) -8); }

// 2の倍数固定値方式
```

```
for( g_noise = 0; g_noise == 0; ) { g_noise = ((QiRand_rand()%3) -1) *16; }

// メモ
// 上記は実際のプログラムから抜粋
// QiRand_rand関数 :rnd関数と使い方が同じな精度の高い乱数生成関数
// for文は乱数加算値が0になるのを回避する目的
```

昨年(2017年)に発見した「2の倍数固定値方式」は、
学習結果のパラメータ精度は収束までにきわめて高速(少ない計算量)に機能します。

他にも、計算資源をより消費することで精度向上が担保される性質を持ちます。
具体的にはパラメータ精度の粗さは、必要に応じて2の倍数の固定値を利用し分解能の精度とします。
これにより細かい精度は、粗い精度がとりえない値を取ることを保証することが可能になります。

例として ±128固定でパラメータ調整した時：
prm1 = 128
prm2 = -128
prm3 = 512
prm4 = 384

例として ±16固定でパラメータ調整した時：
prm1 = 144
prm2 = -112
prm3 = 544
prm4 = 368

※当然「2重の乱数方式」などの方式でも乱数の取りうる値を、

学習が進むにつれ小さくするやりかたは一般的であることは把握しています。

ですが今回の方式は、パラメータに加えるノイズを2の倍数の固定とし分解能を表すことが特徴。

「2の倍数固定値方式」は、評価関数の機械的構築に貢献できるものと考えています。

End

第28回世界コンピュータ将棋選手権
SilverBullet　アピール文

2018/03/31

手塚規雄
山内浩之

SilverBulletの由来

- 一般的には狼男や悪魔を撃退できるといわれている便利な武器という意味合い
- ソフトウエア工学分野では「No Silver Bullet（銀の弾丸はない）」の論文というものがある。その内容はすべての問題に通用する万能な解決策はないという内容。（Wikipedia参照）
- その論文に反抗して「銀の弾丸」はあるよ！という無謀な挑戦をしてみた。

SilverBullet開発方針

- 使用するライブラリ
  やねうら王：強い、使い慣れている
  python-shogi：Pythonで開発するため
  dlshogi：DeepLiearningを使用しているため

- ライブラリの使用目的
  異なるライブラリを1つのソフトの中で
  目的どおりに動作した例がなかった（？）
  のでその実現を目指す

なぜ複数のライブラリを同居させるのか？その意味は？

- 第5回電王トーナメントではやねうら王に対する対策がいくつか行われていた。昔から言われていたオープン化したためだが、露骨に対策されるとやはり強化しても勝つのが難しくなってくる
- 有名ライブラリと狙い撃ちされにくいライブラリを同居させることでアンチ対策ができるメリットがある。その分棋力は下がってしまうデメリットもある

現在、研究中のもの

- ライブラリごとに序盤、中盤、終盤が得意なライブラリが存在する。そこでまずは序中盤と中終盤とで使用するライブラリを変更する
- ライブラリ変更タイミングは手数でなく局面から判断する。その判断基準は持ち駒、コマの配置、成り駒の有無、打ち込み先の多さなど研究中ではある。本番までに間に合うかは不明。

こまあそび
アピール文書

2018/03/29

探索部

- 基本アルゴリズムはαβ法。
- 王手、王手回避手、駒をとる手などを延長している。
- 延長深さの制限を先手番、後手番別々に持っている。
  たとえば先手番Max4手、後手番Max4手の場合、
  先手4手延長＋後手4手延長=計8手延長はOKだが、
  先手5手延長＋後手3手延長=計8手延長はNGなど。
- 手を読む広さは探索深さによって変えている。

評価部

- 評価関数は学習は使わず手でチューニングしている。
- 駒組みは落とし穴方式で行っている。
- 銀桂は敵陣に近くの手の点数を高くしている。
- 金は上部に出る点を低くしている。
- 金は角と筋違いの位置の点数を高くしている。
- 中盤、終盤は角と金の価値がほぼ同じにしている。
- 竜王、飛車の価値を高めに設定している。

臥龍 アピール文書 (第28回世界コンピュータ将棋選手権)

プログラム情報

- プログラム名 :
  - 臥龍
- 初参加 :
  - 第3回コンピュータ将棋選手権
- 通算成績 :
  - 73勝112敗4分
- 開発言語 :
  - Java
- ソースコード行数 :
  - 思考部 20100行、UI部 10971行
- 採用している手法 :
  - αβ探索、反復深化、トランスポジションテーブル、null move pruning
  - 探索の延長 : 王手
  - 詰み探索 : 深さ優先探索
- 採用していない手法 :
  - 並列探索、進行度、評価関数の学習、df-pn
- 探索速度 :
  - 約30万nodes/s (シングルスレッド Core i7 4960HQ 2.6GHz)
- 評価関数パラメータ :
  - 駒損得、成駒、当たり駒、駒の絶対位置、玉の自由度、各枡の利き、駒の働き、囲い、ピン、手番
  - 合計 約750

開発者情報

- 開発者 :
  - 高田淳一
    - Twitter
    - Facebook
- 関連Web Site :
  - コンピュータ将棋プログラム「臥龍」
  - 「臥龍」開発メモ

前回からの改良

- 全く変更なし。

2018/3/11記

dainomaruDNNcは昨年出場したライブラリーと異なり、dlshogiを使用した。

dlshogiではディープラーニングによる強化学習を行っているものとなっている。

elmo_for_leamにて訓練データ(17.6GB)、テストデータ(2057件)により

強化学習を行った。MomentumSGDをRMSpropGravesに変更、ミニバッチ256、エポック2にて実施した。

batchsize=256、

RMSpropGraves(lr=0.001)

WeightDecay(rate=0)

val_lambda=0.5

train position num = 500000000

test position num = 6234

dlshogiのライブラリからの変更点は強化学習における最適化手法をMomentumSGDから

RMSpropGravesに変更

した 1 点のみである。

山田将棋について（2018 年）

全体像

クラシカルな構造・アルゴリズムとなっています。

データ構造

配列による盤駒表現、駒背番号制、利き数、
飛び利き方向ビットの OR 値、
利いている駒背番号ビットの OR 値、
囲いへ誘導するための落とし穴表、
玉位置からの距離に応じた評価値を納めた表、
pinned と cover の概念、置換表、
8 近傍利き位置を納めた表、8 近傍合法移動先を納めた表、
など

アルゴリズム

αβ探索、反復深化、局面が静かでない場合の探索延長、
手調整した仮評価による手のオーダリングと前向き枝狩り、
null move pruning、late move reduction、
killer move heuristic、pass move、
YSS 式指し手の反復生成、Crafty 方式の並列探索、
反復深化による詰め将棋ルーチン
root ノードでの簡易必死検出、
leaf ノード付近での簡易一手詰み検出、
予測読み、フィッシャークロック対応、

など

評価関数

駒割、玉の安全度、囲い、盤上の利き、駒への当たり、大駒の働き、
そっぽ、金駒へのひもなどの、手調整した評価値の線形和

未採用

多重反復深化、影の利き、SEE、持ち駒の優劣表現、
bitboard、実現確率探索、評価関数評価値の学習、など

『海底』アピール文書

迫田真太郎

平成 30 年 3 月 30 日

1　概要

『海底』は 2017 年 1 月から開発が始められた将棋ソフトです. ライブラリを使わずフルスクラッチで作られており, 将棋ソフト開発経験を通じて作者の技術力を向上させ, 採取的により強い将棋ソフトを作ることを目標としています.

去年は第 27 回世界コンピュータ将棋選手権, 第 5 回電王トーナメントに参加し, 結果はそれぞれ 3 勝 4 敗の 24 位で一次予選敗退,4 勝 4 敗の 26 位で予選敗退でした. 今年は勝ち越しを目指して日々開発を行っています.

将棋ソフトのレートを測定しているサイトで　は WCSC27 版が R:707, SDT5 版が R:970 となっており, 現時点で最新のバージョンは SDT5 版に対して 9 割程度勝つことのできる棋力となっています.

2　探索部

$\alpha\beta$ 法を基本とした探索を行っています.StockFish や強豪ソフトを, 開発者自身の技術力が許す範囲で部分的に模倣した形となっており, 採用している枝刈りとしては

- Razoring
- Futility pruning
- Null Move Pruning
- 多重反復深化
- PVS(Principal Variation Search)
- LMR(Late Move Reduction)

が挙げられます.

また残り探索深さが 1 未満になった段階で静止探索を行っています. 静止探索では最初の一度だけその局面で駒を取る手を全て生成し, 以降は直前に動いた駒を取り返す手だけを生成しています.

3　評価関数

2 駒関係 (KP,PP) を特徴量とする線形評価関数を用いています.SDT5 版は floodgate の棋譜から Bonanza メソッドを用いて学習したものでしたが, そこから elmo 式の RootStrap を行うことで棋力が向上しました.

棋譜を書き出すのではなく数百局の自己対局を行い, 棋力変化の検証と学習用データ生成をまとめて行ったのちその場で学習することを繰り返しているのですが, かなり早い段階で頭打ちとなってしまっています.

4　指し手生成など

基本的には Bitboard を用いて指し手生成を行っています. さらにピンや自殺手などの細かい合法性の取り扱いのため, 各マスについて利きの数と, 長い利きがある方向を管理しています.

指し手のオーダリングに関しては駒を取る手を先に生成することを基本とし, 駒を取らない手については history を用いて浅い探索での結果が考慮されるようにしています.

5　その他

現在は Deep Learning を用いた将棋ソフト開発も並行して進めており, 何らかの形で組み合わせることができないかと方法を模索しているところです.

実戦ではフレームワークとして tensorflow を用いて C++で動かすことを想定していますが、学習においては python を用いて実装をしているため python-shogi をライブラリ登録しています. また学習部の細かい実装やおおまかなアルゴリズムは dlshogi を参考にしたためこちらもライブラリ登録しています.

質問のアピールという言葉の意味がわからない。
アピールは、野球で３塁ランナーが外野フライを取る前にスタートしたときに３塁にボールを送って審判に言ってアウトにする手法だと思うが将棋でなぜこの言葉がでてくるのかわからない。
著作権に関する質問だとすると、隠岐は自分でコーディングしたものであります。
思考部だけで約６７３万ステップくらいあります。また別に詰将棋部がん万ステップくらい、画像部がインターネットで公開してますが、１万ステップくらいあると思います。
だから、ステップ数だけだと他のどのソフトにも勝つと思います。
ただ、これを自分だけでコーディングしたかとなると、偽で、実はコンピュータにコーディングさせております。
つまり、将棋の思考部はワンパターンになりやすく、それを利用して将棋ソースを作るプログラムを開発してそのプログラムにやらせてます。
下記が隠岐の思考部のソースの一覧です。

```
 ドライブ C のボリューム ラベルがありません。
 ボリューム シリアル番号は 8411-68B6 です
 ドライブ C のボリューム ラベルは Windows です
 ボリューム シリアル番号は FE50-A6AE です

 C:¥VC2012¥OkiSikou のディレクトリ

2017/12/15  17:23    <DIR>          .
2017/12/15  17:23    <DIR>          ..
2017/12/21  16:13           233,591 aa0.h
2017/12/21  16:13           971,847 aa1.h
2017/12/21  16:13           951,790 aa2.h
2017/12/21  16:22           948,607 aa3.h
2017/12/21  16:13             1,915 aa4.h
2017/12/21  16:13           958,522 aa5.h
```

2017/12/21 16:13 926,355 aa6.h
2017/12/21 16:13 880,167 aa7.h
2017/12/21 16:22 904,298 aa8.h
2017/10/14 10:57 854,125 ab1.h
2017/09/20 11:46 922,819 ag1.h
2017/11/02 05:46 490,627 ag2.h
2017/11/08 16:27 390,748 Ai.h
2007/02/26 16:50 332 Ai_g.cpp
2007/06/30 07:42 333 Ai_s.cpp
2017/11/03 09:55 933,730 aj1.h
2017/11/03 09:55 917,566 aj2.h
2017/11/03 09:39 915,992 aj3.h
2017/12/17 05:25 886,966 aj4.h
2017/12/17 06:33 155,339 aj5.h
2017/12/22 10:17 910,259 ak1.h
2017/12/22 08:59 913,186 ak2.h
2017/12/22 08:59 698,803 ak3.h
2009/09/13 16:05 444 amari.cpp
2017/02/24 05:23 7,457 ana.h
2007/02/26 16:50 765 Ana_g.cpp
2007/02/26 16:50 747 Ana_s.cpp
2017/12/17 17:01 955,920 ao1.h
2017/12/03 06:32 911,282 as1.h
2017/10/13 05:56 862,004 as2.h
2017/11/12 12:27 1,005,819 at1.h
2017/11/12 15:35 419,606 at2.h
2017/12/15 06:32 912,379 az1.h
2017/12/15 06:32 541,833 az2.h
2017/12/24 15:42 305,109 Bougin.h

2007/02/26 16:50 427 Bougin_g.cpp
2007/02/26 16:50 428 Bougin_s.cpp
2009/09/05 15:20 684 check_2fu.cpp
2015/06/28 06:00 2,820 core0a.cpp
2015/06/28 05:53 3,828 core0y.cpp
2017/11/17 09:43 223,544 Eishun.h
2007/06/22 12:12 340 Eishun_g.cpp
2007/06/22 12:12 341 Eishun_s.cpp
2010/09/11 11:21 4,209 g11_map.cpp
2015/10/21 15:51 38,562 ga_total.cpp
2015/07/13 10:26 1,030 get_sasite.cpp
2011/01/04 09:08 13,565 gftotal.cpp
2015/02/26 09:53 20,985 gfttotal.cpp
2010/06/30 09:22 5,835 gictotal.cpp
2010/09/14 09:21 2,054 gobtotal.cpp
2005/11/08 06:12 4,768 Gote.h
2010/04/20 13:23 523 gou1total.cpp
2016/11/25 06:12 326 gou2total.cpp
2012/01/08 11:32 877 gou3total.cpp
2011/05/29 09:19 1,205 gou4total.cpp
2010/09/14 08:54 1,279 gou5total.cpp
2010/09/14 09:05 1,051 gou6total.cpp
2010/09/14 09:08 988 gou7total.cpp
2010/09/15 07:15 6,579 Goutotal.cpp
2011/01/15 10:28 29,165 Gqctotal.cpp
2010/10/04 14:44 18,649 gqctotal2.cpp
2010/09/11 10:12 1,754 gq_total.cpp
2010/10/04 14:49 1,684 gq_total2.cpp
2014/06/29 07:33 32,303 grctotal.cpp

2010/09/09 08:50 16,105 gtntotal.cpp
2010/12/29 13:53 606 gu1total.cpp
2010/09/14 09:32 1,786 guctotal.cpp
2017/11/04 06:18 924,679 gx1.h
2017/11/04 06:18 737,884 gx2.h
2011/10/08 17:50 23,935 gxctotal.cpp
2010/04/20 06:06 22,298 gztotal.cpp
2009/12/10 08:19 3,952 g_kiki.cpp
2012/04/02 17:14 16,380 g_total.cpp
2010/10/04 16:35 377 hasshu.cpp
2004/02/26 22:26 15,987 hikyo_s.h
2000/04/08 05:06 2,115 hikyo_u.h
2017/08/23 05:29 53,337 Hineri.h
2014/03/27 06:31 392 Hineri_g.cpp
2014/03/27 06:31 393 Hineri_s.cpp
2010/12/20 15:11 82,574 hisshi.cpp
2011/01/03 10:26 7,203 hisshi5.cpp
2011/01/09 06:22 14,674 hi_s.h
2007/02/26 16:50 5,628 hi_sort.cpp
2011/01/09 06:27 4,550 hi_u.h
2016/04/23 10:08 6,273 h_open.cpp
2017/08/17 05:29 16,814 ibiana.h
2012/04/29 10:09 943 inaniwa.h
2011/05/21 13:58 211 Inaniwa_g.cpp
2011/05/21 13:58 212 Inaniwa_s.cpp
2016/10/13 06:30 1,048,576 index_s.dat
2017/12/19 05:35 1,024 INFO.dat
2016/05/04 08:57 2,038 Jissen.lib
2014/05/19 10:25 13,199 Joseki.cpp

2017/12/12 06:28 162,270 Joshiki.h
2017/08/01 12:39 36,037 Joshiki2.h
2007/02/26 16:50 1,532 Josiki_g.cpp
2007/02/26 16:50 1,538 Josiki_s.cpp
2004/03/11 05:45 5,449 kaku_s.h
2004/03/11 22:40 2,666 kaku_u.h
2010/07/13 15:25 483 kiki.cpp
2012/07/15 09:07 34,975 Kousoku.cpp
2009/08/19 09:20 23,950 ksort.cpp
2004/02/12 22:45 2,171 kyo_s.h
2004/02/14 07:23 3,710 kyo_u.h
2010/09/16 08:34 3,640 mawari.cpp
2011/09/07 06:16 4,156 mawari_ten2.cpp
2017/11/03 17:19 74,197 Migi4ken.h
2007/02/26 16:50 394 Migi_g.cpp
2007/02/26 16:50 395 Migi_s.cpp
2010/09/05 14:52 326 modori.cpp
2010/09/04 09:13 563 move_flag.cpp
2017/10/14 17:05 170,487 Mukai.h
2007/06/25 07:51 581 Mukai_g.cpp
2007/06/25 07:51 584 Mukai_s.cpp
2017/12/21 16:49 187,606 Naka.h
2012/11/28 13:58 380 Naka_g.cpp
2013/10/18 06:56 381 Naka_s.cpp
2010/09/15 13:50 1,909 narabikae.cpp
2010/09/15 13:50 1,671 narabikae_n.cpp
2010/09/15 16:42 1,475 nigeru.cpp
2010/09/15 16:42 975 nigeru2.cpp

2000/10/08 08:49 333 ochi10_s.h

2000/04/08 08:34 303 ochi10_u.h

2009/02/25 15:43 20,098 Ochi2_s.h

2005/01/05 22:22 10,295 Ochi2_u.h

2004/04/01 23:01 18,085 Ochi4_s.h

2004/04/02 22:39 4,097 Ochi4_u.h

2006/04/19 06:18 13,240 Ochi6_s.h

2000/04/08 08:09 1,221 Ochi6_u.h

2010/12/26 05:26 2,501 Ochi8_s.h

2000/04/08 08:20 827 Ochi8_u.h

2007/02/26 16:50 1,896 Ochi_s.cpp

2007/02/26 16:50 689 Ochi_u.cpp

2015/01/03 06:54 37,412 OkiSikou.aps

2017/06/28 16:32 16,896 OkiSikou.cpp

2006/02/15 21:41 146 OkiSikou.def

2006/02/15 21:57 527 OkiSikou.h

2013/08/11 10:20 47,762,432 OkiSikou.ncb

2016/01/01 08:15 3,145 OkiSikou.rc

2017/07/12 16:15 1,232 OkiSikou.sln

2012/12/17 05:58 29,733 OkiSikou.vcproj

2013/08/11 10:20 2,762 OkiSikou.vcproj.TSUMA-VAIO.TSUMA.user

2017/12/15 17:21 27,035 OkiSikou.vcxproj

2017/12/15 17:21 42,810 OkiSikou.vcxproj.filters

2017/12/24 12:49 863 OkiSikou.vcxproj.user

2009/02/12 13:09 3,522 ouchifu.cpp

2015/07/24 05:41 36,639 oute.h

2007/02/26 16:50 335 Oute_g.cpp

2007/02/26 16:50 338 Oute_s.cpp

2009/01/06 11:29 2,333 outori.cpp

```
2016/09/30  10:35         2,560 out_moji2.cpp
2017/04/19  17:08        93,170 p1sort.cpp
2015/06/26  17:19         5,687 p2sort.cpp
2017/09/26  06:26       100,002 p3sort.cpp
2017/11/02  05:53        73,809 p4sort.cpp
2017/10/13  15:14       116,784 p5sort.cpp
2017/07/11  17:01       112,628 p6sort.cpp
2017/11/12  15:14        38,601 p7sort.cpp
2017/09/28  06:31       147,479 psort.cpp
2009/05/12  14:16        29,769 ransu.cpp
2006/02/15  21:41         2,402 ReadMe.txt
2017/07/03  05:23    <DIR>      res
2006/02/15  21:41           375 Resource.h
2016/10/12  15:04        10,373 rhasshu.cpp
2016/10/13  06:32        73,728 r_ems.dat
2010/09/11  11:20         4,201 s11_map.cpp
2011/04/21  10:36           834 sahasshu.cpp
2010/09/15  17:03           583 sakujo_y.cpp
2017/10/12  10:56       170,182 Sanken.h
2013/11/18  11:18           384 Sanken_g.cpp
2013/11/18  11:18           387 Sanken_s.cpp
2009/03/12  05:49           894 sasu.cpp
2015/10/21  15:32        38,586 sa_total.cpp
2010/09/07  06:16         1,757 seme.cpp
2017/08/22  11:23        35,479 semeru.h
2010/09/07  09:44         2,315 seme_h.cpp
2010/09/07  15:42         2,597 seme_r.cpp
2013/11/06  05:13        11,233 Senpo_g.cpp
2013/11/06  05:13        11,486 Senpo_s.cpp
```

2005/01/29 06:20 4,847 Sente.h
2011/01/04 09:08 13,598 Sftotal.cpp
2011/06/07 06:26 21,483 sfttotal.cpp
2009/04/04 14:38 1,666,568 sfu_jun.cpp
2009/04/05 07:38 21,702 sgin_jun.cpp
2012/03/14 16:58 21,863 shi_jun.cpp
2010/06/30 09:22 5,815 Sictotal.cpp
2017/11/20 13:01 316,061 Siken.h
2007/06/04 08:24 386 Siken_g.cpp
2007/06/04 08:24 385 Siken_s.cpp
2017/12/24 17:42 14,209 sikou.txt
2017/12/24 16:02 161,313 sikou11.cpp
2010/10/07 13:03 1,390 sikou11a.cpp
2017/12/21 16:22 899,238 sikou11aa.cpp
2017/10/08 09:52 595,209 sikou11ab.cpp
2017/11/02 05:46 504,356 sikou11ag.cpp
2017/12/17 06:33 543,685 sikou11aj.cpp
2017/12/22 09:02 567,082 sikou11ak.cpp
2017/12/17 10:00 322,883 sikou11ao.cpp
2017/12/18 15:05 703,218 sikou11as.cpp
2017/12/08 05:39 329,514 sikou11at.cpp
2017/12/15 05:52 162,447 sikou11az.cpp
2017/11/04 06:03 599,506 sikou11gin.cpp
2010/10/17 05:56 4,861 sikou11naru.cpp
2014/07/04 16:26 2,392 sikou11oute.cpp
2015/04/23 05:20 7,927 sikou11wz.cpp
2015/03/09 19:10 1,459 sikou11y.cpp
2017/12/06 05:28 650,111 sikou11yb.cpp
2017/11/08 17:29 591,099 sikou11yc.cpp

2017/12/15 08:19 485,531 sikou11yj.cpp
2017/12/23 06:26 748,003 sikou11yk.cpp
2017/11/21 06:53 651,305 sikou11yo.cpp
2017/10/28 09:13 844,247 sikou11ys.cpp
2017/11/14 13:03 303,277 sikou11yt.cpp
2017/12/17 09:17 949,471 sikou11yy.cpp
2017/07/10 10:01 128,633 sikou11yz.cpp
2017/12/01 17:15 738,017 sikou11zb.cpp
2017/12/04 17:44 644,096 sikou11zg.cpp
2017/12/18 12:24 186,206 sikou11zi.cpp
2017/12/15 17:21 749,318 sikou11zj.cpp
2017/12/23 16:37 765,930 sikou11zk.cpp
2017/10/26 12:21 711,106 sikou11zl.cpp
2017/11/30 17:36 572,390 sikou11zo.cpp
2017/11/17 14:02 933,878 sikou11zs.cpp
2017/12/18 15:39 468,017 sikou11zt.cpp
2017/12/20 16:01 931,631 sikou11zz.cpp
2017/09/05 08:37 22,500 sikou2.txt
2009/05/07 10:11 21,856 sjun.cpp
2012/03/12 12:49 21,865 skaku_jun.cpp
2009/04/20 07:28 17,324 skei_jun.cpp
2015/10/21 08:16 1,804,350 skin_jun.cpp
2012/04/01 17:00 19,920 skyo_jun.cpp
2010/09/14 09:17 2,054 sobtotal.cpp
2010/10/04 16:02 40,020 Soctotal.cpp
2017/09/23 15:27 74,097 Sode.h
2007/02/26 16:50 334 Sode_g.cpp
2007/02/26 16:50 335 Sode_s.cpp

2010/09/04 16:31 43,605 softotal.cpp
2010/04/20 13:23 524 sou1total.cpp
2016/11/25 06:11 328 sou2total.cpp
2012/01/08 11:28 874 sou3total.cpp
2011/05/29 09:19 1,203 sou4total.cpp
2011/03/15 17:09 1,265 sou5total.cpp
2010/09/14 09:01 1,054 sou6total.cpp
2010/09/14 09:13 988 sou7total.cpp
2010/09/15 07:15 6,803 Soutotal.cpp
2012/07/15 09:13 190,757 speed_sort.cpp
2011/01/15 10:28 24,711 Sqctotal.cpp
2010/10/04 17:24 18,676 sqctotal2.cpp
2011/01/15 10:28 26,364 sq_total.cpp
2010/10/04 17:19 1,688 sq_total2.cpp
2013/08/12 08:23 28,696 Srctotal.cpp
2010/04/05 08:51 696 ssort.cpp
2007/02/26 16:50 205 stdafx.cpp
2013/08/12 05:59 1,336 stdafx.h
2010/09/09 08:29 15,841 stntotal.cpp
2010/09/14 08:36 609 su1total.cpp
2010/09/14 09:29 1,787 suctotal.cpp
2011/10/08 17:50 23,284 sxctotal.cpp
2010/04/20 06:06 24,753 sztotal.cpp
2009/12/10 08:19 3,940 s_kiki.cpp
2012/04/02 17:14 16,410 S_total.cpp
2013/08/12 05:59 371 targetver.h
2010/09/20 07:21 1,741 te_valuen.cpp
2015/06/10 05:56 1,527 torui.cpp
2010/09/08 16:55 1,571 toruj.cpp

```
2016/07/15  13:11           2,664 toruj10.cpp
2016/10/27  11:35           2,590 toruj10f.cpp
2010/09/09  07:23           2,469 toruj10u.cpp
2010/09/09  07:33           1,593 toruj2.cpp
2010/09/20  07:24           1,855 toruj2n.cpp
2010/09/09  07:38           1,928 toruj3.cpp
2010/09/09  07:25           2,137 toruj4.cpp
2010/09/08  17:00           1,996 toruj4f.cpp
2010/09/08  17:08           1,948 toruki.cpp
2017/12/09  15:41           6,165 Total.cpp
2010/09/07  09:46           6,563 tumi_h.cpp
2010/09/07  09:50           2,616 tumi_h5.cpp
2011/02/20  08:04           1,125 tumi_k.cpp
2010/09/07  14:14           3,480 tumi_r.cpp
2009/06/02  07:02           4,764 uchifu.cpp
2010/09/09  09:07           2,524 uchifu_g.cpp
2010/09/09  13:25           2,523 uchifu_s.cpp
2011/12/15  07:41           5,031 uke.cpp
2010/09/07  09:38           4,718 uke_h.cpp
2013/11/06  05:13           4,931 uke_r.cpp
2015/05/09  08:34          13,719 ura.cpp
2010/09/15  17:09             628 wrong_y.cpp
2017/07/03  05:23    <DIR>          x64
2017/09/19  14:37          20,184 xana.h
2016/03/21  05:34          84,918 Xhineri.h
2014/03/27  06:31             406 Xhineri_g.cpp
2014/03/27  06:31             407 Xhineri_s.cpp
2017/01/26  08:53          12,485 xinaniwa.h
2013/11/06  05:13             190 xinaniwa_g.cpp
```

2013/11/06 05:13 189 xinaniwa_s.cpp
2017/10/14 17:52 160,736 Xmukai.h
2017/12/21 16:56 225,870 Xnaka.h
2012/11/28 14:00 344 Xnaka_g.cpp
2012/11/28 14:00 345 Xnaka_s.cpp
2017/10/12 15:30 199,802 Xsanken.h
2013/11/18 11:18 398 Xsanken_g.cpp
2013/11/18 11:18 401 Xsanken_s.cpp
2017/11/20 05:54 481,142 Xsiken.h
2007/08/16 06:13 405 Xsiken_g.cpp
2007/08/16 06:13 408 Xsiken_s.cpp
2017/11/11 17:17 430,715 Yagura.h
2017/10/10 15:51 362,775 yagura1.h
2007/02/26 16:50 365 yagura1_g.cpp
2007/02/26 16:50 368 yagura1_s.cpp
2007/02/26 16:50 526 Yagura_g.cpp
2007/02/26 16:50 525 Yagura_s.cpp
2017/12/05 17:33 911,091 yb1.h
2017/12/05 17:24 694,835 yb2.h
2017/11/08 17:29 922,638 yc1.h
2017/11/08 17:29 912,557 yc2.h
2017/11/08 17:29 227,890 yc3.h
2017/10/31 16:43 934,334 yj1.h
2017/10/10 12:46 925,816 yj2.h
2017/12/15 08:16 914,872 yj3.h
2017/10/22 16:45 754,739 yj4.h
2017/12/15 08:20 462,869 yj5.h
2017/12/23 06:04 929,167 yk1.h
2017/12/23 06:20 902,980 yk2.h

2017/12/23 06:04 916,692 yk3.h
2017/12/23 06:04 896,052 yk4.h
2017/12/24 05:27 331,256 yk5.h
2017/11/29 06:25 980,149 yo1.h
2017/11/21 06:53 953,651 yo2.h
2017/11/21 06:53 929,612 yo3.h
2017/11/21 06:53 94,967 yo4.h
2017/10/28 09:13 932,110 ys1.h
2017/10/28 08:57 932,602 ys2.h
2017/10/28 08:57 931,213 ys3.h
2017/10/28 09:13 297,450 ys4.h
2017/12/04 17:32 289,587 Ysenpo.h
2010/09/15 17:26 621 ysenpo_g.cpp
2015/11/14 11:40 2,086 ysenpo_o.h
2010/09/16 08:40 405 ysenpo_og.cpp
2010/09/16 08:37 406 ysenpo_os.cpp
2010/09/15 17:10 622 ysenpo_s.cpp
2017/11/14 13:03 1,011,496 yt1.h
2017/11/14 13:01 556,239 yt2.h
2017/07/08 11:25 56,292 YThreadProc.cpp
2010/09/02 07:31 3,114 yusen.cpp
2010/09/15 16:57 620 yusen_y.cpp
2017/12/02 06:35 966,970 yy0.h
2017/12/17 17:26 945,804 yy1.h
2017/12/08 06:08 974,692 yy2.h
2017/12/08 06:19 952,449 yy3.h
2017/12/02 06:30 18,545 yy4.h
2017/12/02 17:16 933,979 yy5.h

2017/12/02 06:30 942,414 yy6.h

2017/12/02 06:30 933,504 yy7.h

2017/12/06 05:51 376,063 yy8.h

2017/12/02 06:30 961,784 yy9.h

2017/12/02 06:30 955,704 yya.h

2017/12/02 06:30 951,908 yyb.h

2017/12/02 06:30 956,221 yyc.h

2017/12/02 06:30 945,218 yyd.h

2017/12/02 06:30 952,224 yye.h

2017/12/17 17:27 517,896 yyf.h

2017/08/07 12:29 904,688 yz1.h

2009/03/13 07:28 274 y_zahyo.cpp

2017/12/01 17:06 949,162 zb1.h

2017/12/01 17:06 628,375 zb2.h

2017/12/05 05:36 932,336 zg1.h

2017/12/04 17:42 918,441 zg2.h

2017/12/05 05:37 111,541 zg3.h

2017/12/18 12:06 896,526 zi1.h

2017/12/18 12:24 342,339 zi2.h

2017/12/15 17:19 904,325 zj1.h

2017/12/15 17:19 909,624 zj2.h

2017/12/15 17:19 914,725 zj3.h

2017/12/15 17:19 914,528 zj4.h

2017/12/15 17:26 919,504 zj5.h

2017/12/15 17:26 72,947 zj6.h

2017/12/15 16:58 845,223 zj7.h

2017/12/15 17:07 724,389 zj8.h

2017/12/24 15:59 946,363 zk1.h

2017/12/24 15:51 909,718 zk2.h

2017/12/23 16:37 911,438 zk3.h
2017/12/23 16:46 890,280 zk4.h
2017/12/24 13:28 632,328 zk5.h
2017/10/26 12:21 818,275 zl1.h
2017/10/26 12:24 816,278 zl2.h
2017/10/26 12:21 326,150 zl3.h
2017/11/30 17:36 954,126 zo1.h
2017/11/25 12:58 945,150 zo2.h
2017/11/30 17:36 375,846 zo3.h
2017/10/04 06:09 81,709 zs0.h
2017/10/04 06:33 952,861 zs1.h
2017/10/04 10:44 920,635 zs2.h
2017/10/04 06:09 928,773 zs3.h
2017/10/04 06:09 934,280 zs4.h
2017/11/17 14:02 517,401 zs5.h
2010/07/06 10:43 2,495 zsort.cpp
2017/12/18 15:35 980,861 zt1.h
2017/11/13 05:46 253,015 zt2.h
2017/12/20 11:24 934,514 zz0.h
2017/12/20 11:24 1,003,377 zz1.h
2017/12/20 11:24 997,078 zz2.h
2017/12/20 16:09 957,309 zz3.h
2017/12/20 11:24 952,002 zz4.h
2017/12/20 11:24 941,396 zz5.h
2017/12/20 11:24 935,372 zz6.h
2017/12/20 11:24 512,932 zz7.h
2017/12/20 11:24 932,479 zz8.h
2017/12/20 11:24 956,779 zz9.h
2017/12/20 11:29 976,123 zza.h

2017/12/20 11:24   958,849 zzb.h
2017/12/20 11:24   939,383 zzc.h
2017/12/20 11:24   959,865 zzd.h
2017/12/20 11:24   954,547 zze.h
2017/12/20 11:24   937,656 zzf.h
2017/12/20 11:24   959,125 zzg.h
2017/12/20 11:24   933,298 zzh.h
2017/12/20 11:24   948,183 zzi.h
2017/12/20 11:24   950,873 zzj.h
2017/12/20 11:24   960,291 zzk.h
2017/12/20 16:09   140,301 zzl.h
             418 個のファイル   177,688,623 バイト
               4 個のディレクトリ 49,014,206,464 バイトの空き領域

一部が同じタイムスタンプになってますが、これがコンピュータにコーディングさせた証拠で人間がやるとこうなりません。
通常、１ステップ４０バイトくらいでしょうから、ステップ数がどれだけ大きいかわかられると思います。
ただ、六百万ステップだとまだ足りなくて、将棋は複雑ですので八百万ステップくらい欲しいと見てます。
嘘八百ともいいますが。
ただ、これだけステップ数が大きいとコンパイルからリンクまで２０分くらいかかるのでおすすめできないと言うか課題です。

思考部がどうなっているかの質問ですと、序盤は定跡データを使った同一局面検索と類似局面検索で、同一局面検索は乱数で棋譜ファイルを検索して同一局面ではその手を指して類似局面ではそれが見つからなかったら、if文のら列でyagura.hとかがその部分になります。
従って、これらのソースがどうなっているかがわかってしまうと隠岐の指し手にどこが弱

点かばれてしまいますので公開等はとてもじゃないですが、できないんです。
中盤以降は、２手読みなってます。
２手読みとは、森田将棋の故・森田和郎さんが教えてくれた手法で人間の感覚に近いと判断しましたので、この手法を採用してます。
２手読みでは読みの緩い部分が出るので、それを補正したらこれだけ大きいステップになったとも言えます。
隠岐の思考は、１手１秒を目標としてますが、終盤になって持ち駒が増えますと遅くなります。

最近、Bonanzaのソースを見て思ったのですが、彼のソースは極端に少ないです。
将棋というものをほとんど教えてなくて、単に統計データを使って、指し手を補正してると判断しました。
初期は、CSA将棋を使って作られたソースなのに、座標系は違うわ、bitboardは使うわ、知能指数の高い連中というか、毛の３本多い連中は、考え方が違うんだなーと思います。
まっ、彼の手法によって、将棋という古代人が残したパズルを解明する方程式が見つかりつつあるように思えます。
従って、最近自信喪失というか、隠岐の修正を止めて、将棋の思考を若い人にまかせておいた方が良いように思えます。
ただ、それによって将棋という文化が失われつつあるのを危惧してる。

第28回世界コンピュータ将棋選手権　アピール文章

オッズの魔法使い

作者　David Wada

〜〜〜〜〜〜〜〜〜〜〜〜〜〜〜〜〜〜〜〜〜〜〜〜〜〜

今回は趣向を変えて流行りのモンテカルロ探索に手を出してみます

ですが、絶賛社畜労働中なので、出来合いのパーツを組み合わせてみましょう

１．評価関数は、今や古典（？）の　ボナンザ６　から拝借

⇒　もっと新しいライブラリーから移植するべきでしょうが、単に我が怠け者なだけです。

２．詰将棋探索使用

３．モンテカルロ探索ですが、少々アレンジして…

⇒　プレイアウトは終局まで指さない　－　？？手程度で打ち切り、優勢な方を　勝ち　とします
⇒　少々深さ優先にバイアスを掛けます

と、ここまで書いてみて今回の実装は

モンテカルロ付きの並列最良優先探索…みたいに変化しています。（笑）

⇒　静的探索無しです

４．クラウドのマシン使用を考慮中　（多分無理）

⇒　無理です、時間無いです

〜〜〜〜〜〜〜〜〜〜〜〜〜〜〜〜〜〜〜〜〜〜〜〜〜〜

２０１８－０４－１０　評価関数について加筆

2018年アピール文書

プログラムの基本は初回参加時から使いまわしているものを使っています。

今回も、教師無し学習にて３駒間＋２駒間の評価値を学習させています。

学習は強化学習と遺伝的学習を組み合わせています。

前回に比較して大きな変更はありません。バグ修正レベルの変更です。

****

2017年アピール文書

プログラムの基本は初回参加時から使いまわしているものを使い、

今回(前回も)は、教師無し学習を試してみたくなったので

強化学習と遺伝的学習を組み合わせて作成している。

去年は、時間の都合で学習がほとんど進まなかったので

既存の評価値データでの参加だったが、

今回は学習に時間が取れそうなので学習した評価値での参加を考えている。

序盤については、以前から持っているの序盤データをそのまま使用する予定。

たとえ、以前のものよりも弱くなっていたとしても、

学習したデータで望むつもりである。

手抜きについて

手抜きチーム

2018 年 2 月 7 日

手抜きは CSA プロトコルで通信する簡単な将棋プログラムです。手を抜いて作っています。開発の目的は私が将棋のプログラムの仕組みを理解することです。今年の目標はルール通りに指すことと，投了すること（去年は投了しませんでした）と，KP で評価することです。KP の評価ベクターになのは mini の fv_nano.bin を使います。選定の理由は fv_nano.bin が KP だからです。

手抜きは前大会では C++ で実装していましたが，C++ がスタックトレースを吐かないのと.h を書かなければならないのと STL と Boost が嫌になって今大会では C#で実装しました。これにより 3 倍くらい遅くなりました。開発が落ち着いたらまた C++ にします。

（2018 年 3 月 19 日追記）ところが D 言語にしました。D は.h を書かなくていいところや前方宣言がいらないところや C++ と同じ速度で動くところがいいです。リポジトリ：https://github.com/hikaen2/tenuki-d

特長

- コンパクトなコードを目指しています
- 盤面のデータ構造に升の 1 次元配列を採用しています
- $\alpha\beta$ 法で全幅探索します
- 利きを持っていません
- 詰めルーチンを搭載していません
- 定跡を組み込んでいません
- 持ち時間を管理していません

開発者の紹介

鈴木太朗

鈴木はソフトウェア会社の会社員です。仕事のかたわら趣味の自転車と合唱に打ち込んでいます。趣味のかたわら手抜きの開発に打ち込んでいます。将棋は駒の動かし方が分かる程度です。Twitter: @hikaen2

玉川直樹

玉川は P 兼召喚士です。KPP とレーティングシステムを疑っています。将棋ウォーズ 2 段。プログラミングはまったくできません。Twitter: @Neakih_kick

去年からの変更点

1. ルートノードの探索を $\alpha\ \beta$ にした

去年はルートノードは minimax で探索していました（ルートでないノードは $\alpha\ \beta$ で探索していました)。今年はルートノードも $\alpha\ \beta$ 探索するようにしました。これによって探索時間はこうなりました（指し手生成祭りの局面, Core2 SL9400, 深さ 6 までの反復深化）:

変更前： 探索時間 784 秒, 探索ノード数 1,803,710,077

変更後： 探索時間 85 秒, 探索ノード数 138,771,448

2. 反復深化のときに前回の最善手から探索するようにした

たとえば深さ 5 の探索のときに，深さ 4 の探索で得た最善手から探索します。

変更前： 探索時間 85 秒, 探索ノード数 138,771,448

変更後： 探索時間 81 秒, 探索ノード数 130,748,443

3. 合法手を生成するとき，駒を取る手を先に生成するようにした

変更前： 探索時間 81 秒, 探索ノード数 130,748,443

変更後： 探索時間 68 秒, 探索ノード数 59,871,223

4. 静止探索を入れた

深さ 4 の静止探索 (Quiescence Search) を入れました。

変更前： 探索時間 68 秒, 探索ノード数 59,871,223

変更後： 探索時間 92 秒, 探索ノード数 107,209,671

5. KP で評価するようにした

KP で評価するようにしました。評価ベクターになのは mini の fv_nano.bin を使っています。

変更前： 探索時間 92 秒, 探索ノード数 107,209,671

変更後： 探索時間 287 秒, 探索ノード数 197,226,589

6. 置換表を実装した

move のみの置換表 (Transposition Table) を実装しました。

変更前： 探索時間 287 秒, 探索ノード数 197,226,589

変更後： 探索時間 86 秒, 探索ノード数 60,318,785

第２８回世界コンピュータ将棋選手権

きふわらべ
アピール文書

２０１８年０１月３０日　高橋智史

ENUKOMA　KANKEI

Ｎ駒関係　以外のものを　やってみようぜ☆！

DARE

誰なのよ☆？

開発者
高橋　智史

「きふわらべ　を　１から書き直す予定だぜ☆
間に合わなければ、　去年の　飛車を左右に　パタパタする
きふわらべ　を持っていくからな☆」

北白河ちゆり
／TOHO PROJECT
FANMADE　※

コンピュータ将棋エンジン
きふわらべ

「今年の　わたし　に　独自性が無いのなら
アピール文書を読む必要が無いだろう☆　解散だな☆」

「今年から　アピール文書は　Ａ４サイズ　２５ページ　が
上限なのよ。
残りの２４ページ分で、　来年の分を書きましょうよ」

岡崎夢美
／TOHO PROJECT
FANMADE　※

※北白河ちゆり、岡崎夢美は　東方夢時空　の登場キャラクター／（Ｃ）上海アリス幻樂団　様の著作物です。

＼ＣＨＥＣＫ／

「今年のＰＲ文章では、

何が　思考部　で、　何が　思考部　じゃないのか、
意義あり☆！　といった　声を踏まえ、
コンピューターの　思考　について　詳細に書くぜ☆」

NANI

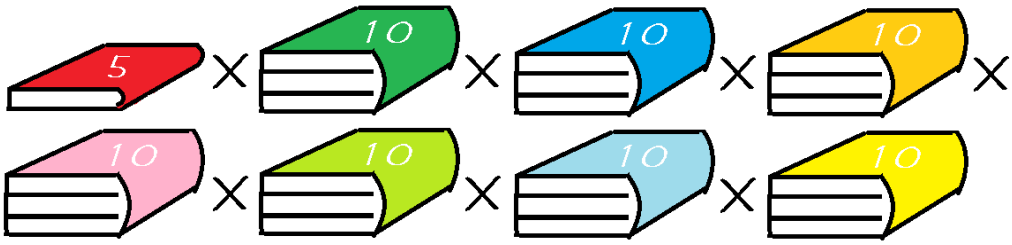
既存ライブラリーは　何　がすごいの？

KYOKUMENSU

局面数　を小さくまとめたのが　すごいんだぜ☆

「例えば　コーヒーの味の違いを　言葉を組み合わせて
ごせんまん
５０００００００種類　利き分ける　ことが　できるとしよう☆」

「味　５０種　×　香り　１０００種　×　コク　１０００種
と　３つの特徴　に分けて　香りとコクについて　深めるか…☆
ボナンザ
というのが、　Ｂｏｎａｎｚａ６．０　とすると……☆」

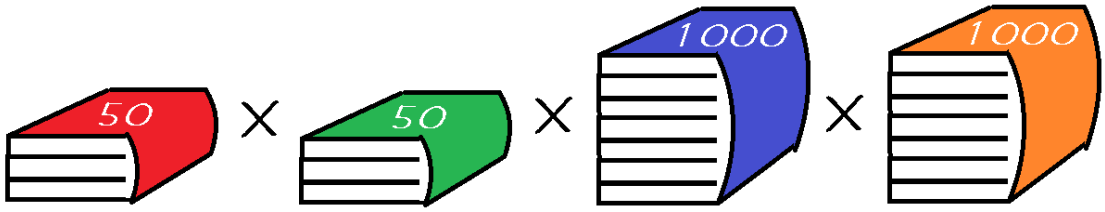


「味　５種　×　香り１０種　×　コク１０種　×　色１０種　×
量１０種　×　ビン１０種　×　フタ１０種　×　ラベル１０種

と　種類を減らして　特徴　を多くするのもあるかと思ったんだぜ☆
いんすうぶんかい
５０００万を　因数分解　しろだぜ☆」

「特徴を増やしても　学習サンプル　をまんべんなく用意できないと
種類が浅くなって　計算増えただけの　持ち腐れよね～。
かくちょうたかい
全ての特徴が使われる　格調高い　設計がいいのよ」

※格調高い　…　全部の駒を使うこと



「４駒関係というのは　２５億だ☆！　ということで
リソースを配分して　やりくりしているのではなく、増強だからな☆」

TIGAUKOTO

局面を見分けていては　違うこと　にならないんだぜ☆

TANSAKU SINAI

局面を 探索しない エキスパート・システムにするのね

「５０００万とか　２５億とか、　０が７桁、９桁　の数で
将棋の局面を　広く　カバーできるのは　すごい☆
将棋の局面数は　０　が　６９桁　あるはずだぜ☆」

※６９桁　…
「将棋の局面数　１：局面数は無量大数」コンピュータ将棋基礎情報研究所
http://ifics81.techblog.jp/archives/2249793.html

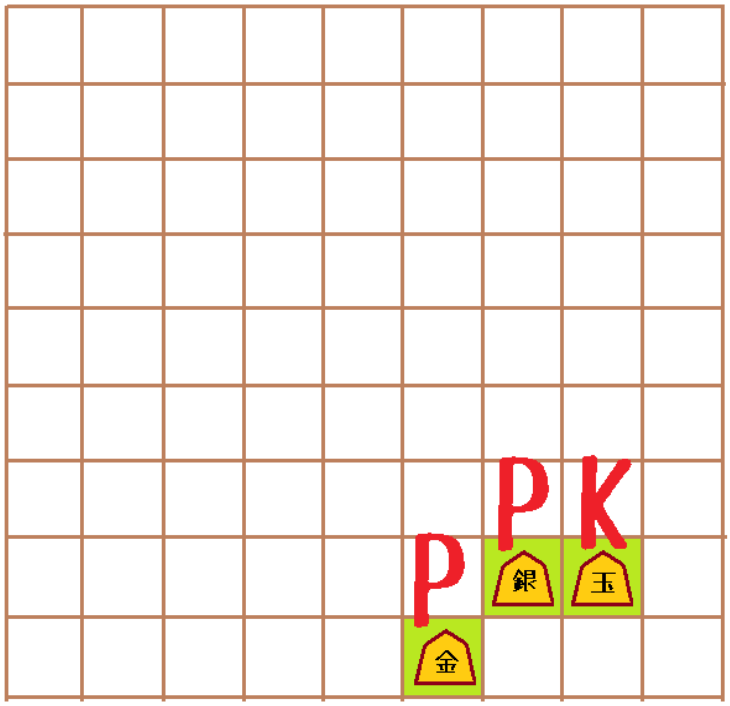
「Ｂｏｎａｎｚａは
局面　を　３駒の部分
が集まったもの、

と捉えていて、
全ての組み合わせは
５０００万☆

言葉では　どっちとも
言えない、という
違いを　表せないような
２つの盤面　にも

どちらが小さい、
どちらが大きい、と
比較できるのが

特筆すべき　長所　だぜ☆」



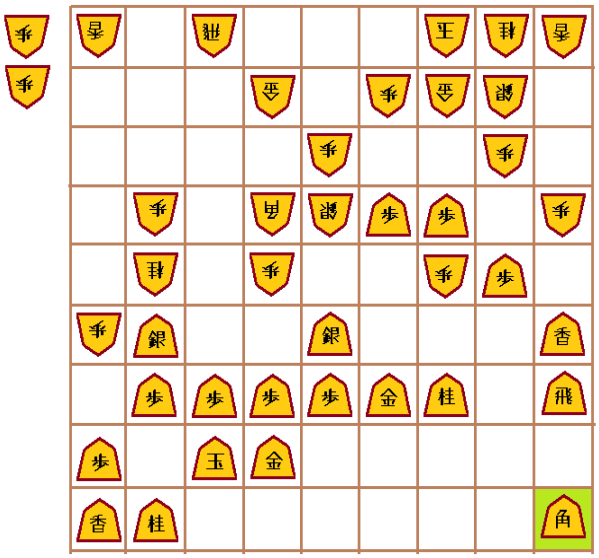
「それも　探索部がたくさんの局面に到達してくれることが　あってのものなのよ。
Ｎ駒関係に勝て、というのは　Ｎ駒関係と　探索部　の相性の良さにも勝て、
ということでもあるのよ」

「局面を　徐々に良くしていく、　という

オセロの盤面にはなくて　将棋にはあるもの、　推移（すいい）　は
あなたに味方しないわよ。　それ以外の土俵を見つけなさい」

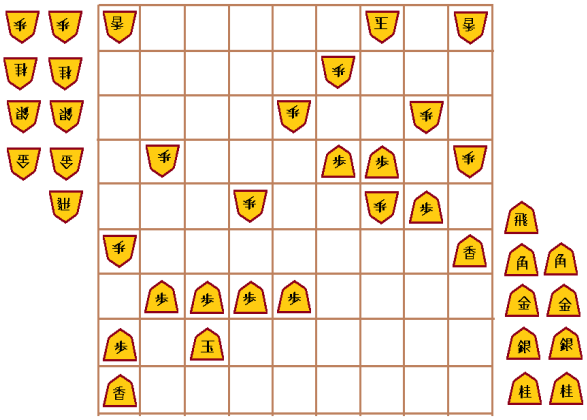
「例えば　何が見えているのか
ｅｌｍｏ　が
１九　に角を打った場面☆

Ｎ駒関係　は　こういう
駒の入り組んだ局面を
評価するのが得意☆」



第２７回世界コンピュータ将棋選手権　決勝リーグ
ｅｌｍｏ　ＶＳ　Ｐｏｎａｎｚａ　Ｃｈａｉｎｅｒ　７１手目
http://live4.computer-shogi.org/wcsc27/kifu/WCSC27_F7_ELM_PON.html

「ならば　将棋を
オセロにするまで☆
これが　Ｎ駒関係を倒す
方針だぜ☆」



「駒が　盤から　無くなってる
じゃないのよ！」

「Ｎ駒関係　は　局面　さえ渡せば　評価してくれる☆
だったら　持ち駒ドッジボール　の　探索勝負　で
局面　を　読み漏れ　させて　Ｎ駒関係にヒマ　させようぜ☆？」

「ストックフィッシュ系探索は　駒を取る指し手は　優先的に調べるし

駒の取り合いを読んでいるときは　盤上が落ち着くまで
読みを延長する　ＳＥＥ　も　よくある技法だし、

飛車が　９一　に回るのをけん制する　１九　に　角を打つような
遠い局面も　よく見てきている　探索に　勝つ気か☆？」

※ＳＥＥ　…　読みを延長することで、水平線効果（すいへいせんこうか）に対策するもの

「 魚は　本質的な水平線効果　で釣ろうぜ☆
釣る方の自分も水平線にならないためにゲームツリーを探索しない
エキスパート・システム　を提案する☆」

※水平線効果　…　悪くしてでも　あがいて、
もっと悪い局面を読みの遠くに追いやること

※本質的な水平線効果　…　将来の局面は　現局面だけの情報から決まるのではないとしたとき、どのような過去局面をたどったかは考えているわけではないコンピューターの水平線効果

「水平線効果は何が問題なのか？」コンピュータ将棋基礎情報研究所
http://lfics81.techblog.jp/archives/2869479.html

※エキスパート・システム　…　アキネーターみたいなやつ

「 エキスパート・システムが
幅広く局面をカバーする　N駒関係と
前向き枝刈り自慢の　ストックフィッシュ系探索　を　上回れるの？」

「 恐らく　全ての　コンピューター将棋ソフトは
現局面から　合法手で進める局面を　読んでいる☆

そして　詰め　は　局面評価　とは全く別物だぜ☆
よくなった局面で　探索部　が
たまたま　詰め　を見つけるんだぜ☆

それに比べて　わたしは　いわば
常に詰めルーチン　をやろう　という思想に属する　方向だぜ☆」

「 急所を確認し　駒を剥がして詰ますのに必要な駒の枚数を把握　し

盤上の駒を　さばいて　読み漏れが詰む　局面に持ち込み、

局面を少しずつ良くするような評価値を見ずに　勝負を狙って寄せ
に行く　将棋ソフト　が作れれば
N駒関係　＋　ストックフィッシュ系探索　とは
原理的に　違い　を出せるだろう☆」

「 局面を　見ずにか☆」

「 N駒関係より　精度の悪い　局面の見方　をするぐらいなら、
局面なんか　見ない　というのも　手　だぜ☆」

SOBOKU SYUGORON

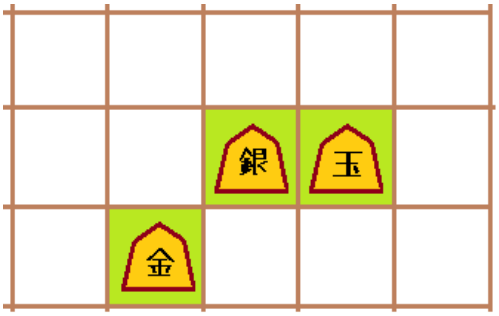
素朴集合論　の表現力を　何に使うの？

MONDAI NO CASE

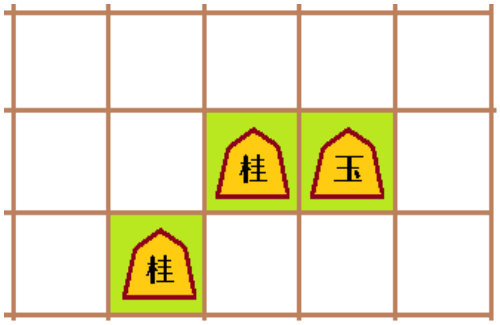
どの 問題のケース に当てはまるか照らし合わそうぜ☆

「将棋の局面数が　なんで　０　が　６９桁　で、
３駒関係も何で　５〇〇〇〇〇〇〇（ごせんまん）パラメーター　も使ってるのか
まず説明する☆」

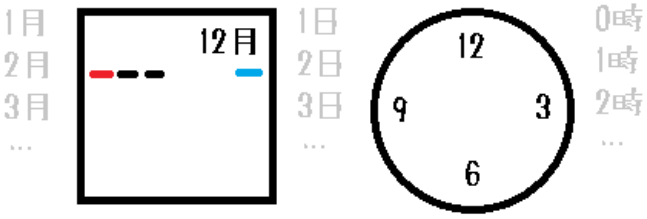
「玉を含めた　３駒なんで
玉・銀・金　の片美濃も
もちろん　表現できるが☆」



「玉・桂・桂　みたいな
変な形も
しっかり　表現する☆」



「人間が　良し悪しを
予断　しないのが
機械学習の強みなのよ！」



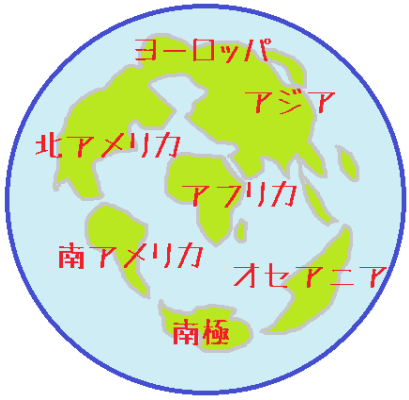
「不揃いな　合同算法（ごうどうさんぽう）　の仲間と　わたしは見ているんだが、

桂の形は　あまり使わないから　省く、とか
２月は　２８日だ、　とか　うるう年だ、　とか

人が　微調整を入れようとする案は
計算コスト　が増えて　使い物から遠くなる☆」

「カレンダーも　年・月・日　の３駒関係よね。
時・分・秒　も入れたら　６駒よ。
夜は寝てるから、と省いていい　夜の２時とか　無いのよ」

「なんだぜ　この　世界地図は☆？」

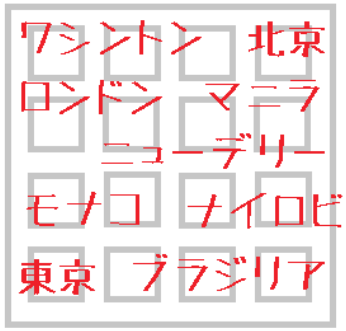


「ものとの関わり合いには　まず　大きく捉えて、

掘り下げたい　ところは掘り下げ、
見どころのない　ところ　は　まとめて扱う
といった　関心の粒度（かんしんりゅうど）　があるんじゃ
ないかだぜ☆？」

「３駒関係は　その　関心の粒度　を
一様にしたことが　強味なのよ。

オールラウンダーのように強いのが
Ｎ駒関係　なのだから」



「局面　を覚えるのなら　そうだろうな☆」

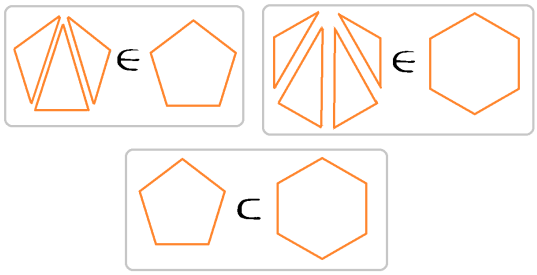
「５０００万サイズのパイを　どう割り振るの、
という分配を　やっているに過ぎない　という意味で、
Ｎ駒関係を　ちょっと変えてみただけ、
から抜け出ない　話しだが……☆、」

「局面を覚えるのは　止めるんで、

合同算法の仲間が　カバーしていたものとは　異なったもの、
関心の粒度　に応えてくれる　表現力を手に入れることが　必要だぜ☆」

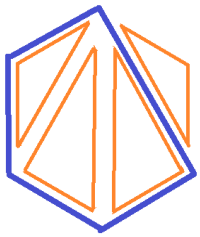
NERAI

狙い　をもって　利きを使いこなすのよ！



※図の解説　…　５角形の中には
３角形が３枚あるし
６角形の中には
３角形が４枚あるから
３角形の枚数の話しをしているのなら
５角形は
６角形の中にあるだろ、
ぐらいの意味☆

素朴集合論



「　素朴集合論（そぼくしゅうごうろん）　というのは、言ってしまえば
あいまい　なものを　無くすために　あるんだぜ☆

どれが中身で　どれが中身じゃないか
そうである、　そうでない

を　はっきり　分けれるもので、
その表現力を　ちょっぴり　借りようぜ☆」

「関心の粒度　に応えることが　できるの？」

「できるんじゃないかと　思っている☆
まず、シンプルな使い方　を説明する☆」

$$A \times B = \{ (a, b) \mid a \in A \ \land \ b \in B \}$$

「数学は　集合論の上に組み立てられている、
と聞くほどの　優れもので、
例えば　デカルト積（掛け算）は　こう☆」

横書き

$A \times B = \{ (a, b) \mid a \in A \ \land \ b \in B \}$

言いたかったこと

$A \times B = (a, b)$

$a \in A \ \land \ b \in B$

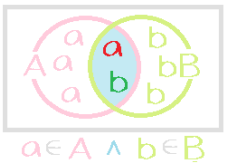
「 数式は　別に　縦に並べるのでなければ　一列に収めなくていいと思うんだが
アンダーラインを引いて　吹き出し　を付けた方が　見やすいだろ☆
｛　｝　が中身で、　｜　の右は　フィルター条件　のようなものだぜ☆」

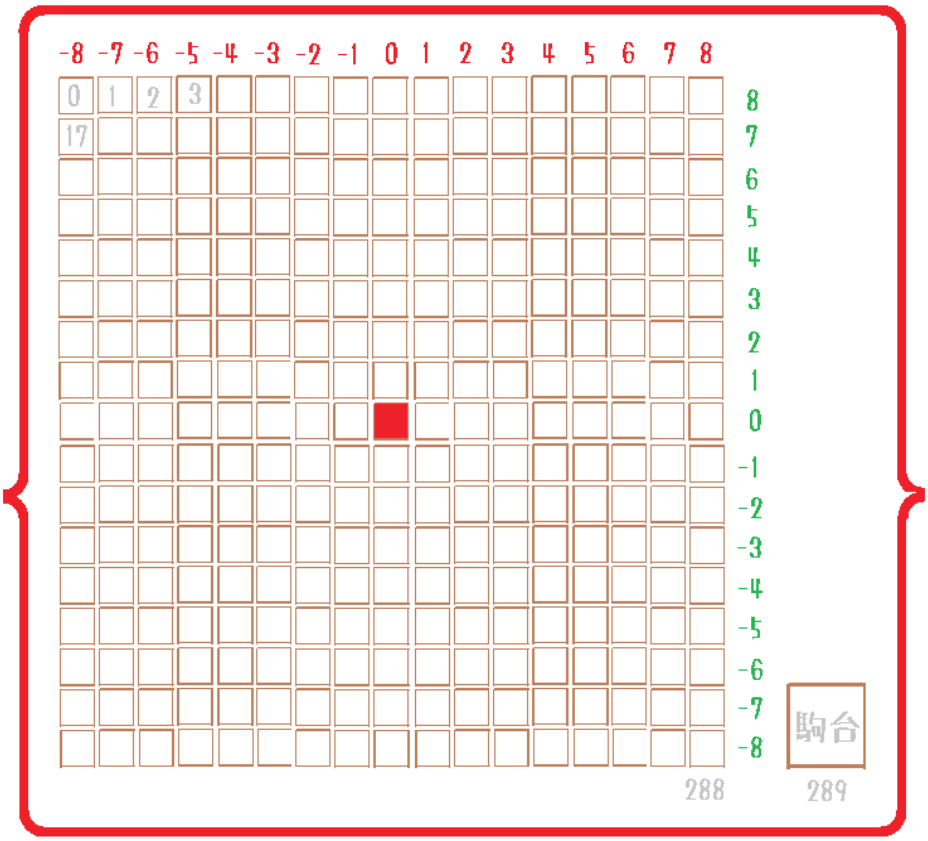
※図の解説　…　５角形の中には
３角形が３個、
６角形の中には　３角形が４個
あるので、

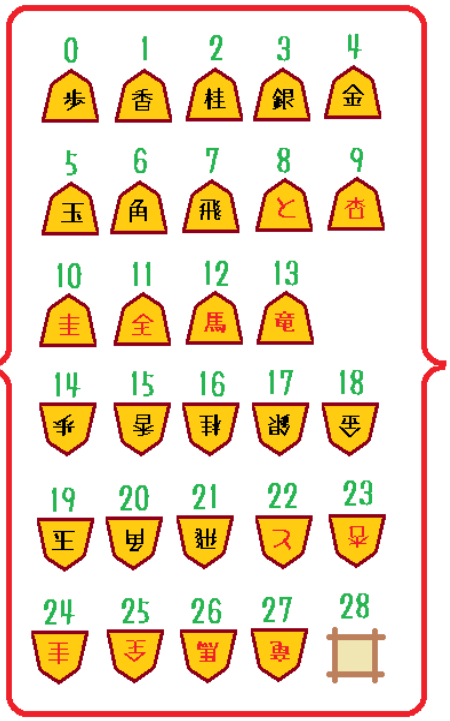
集合同士の掛け算　５角形×６角形
の計算結果……というより
新しく作った　大きな集合の中身は
どのようなメンバーが入っているのか
書いてみると

５角形の要素１つ　　a∈A
６角形の要素１つ　　b∈B
が交わるところ　∧（かつ）
が新しいメンバー１つ　（a，b）
と言っているので
横に３つ、縦に４つ並べてみた☆

「 なんだ、メンバーの　どのペアも　ありえる
という話しかだぜ☆」

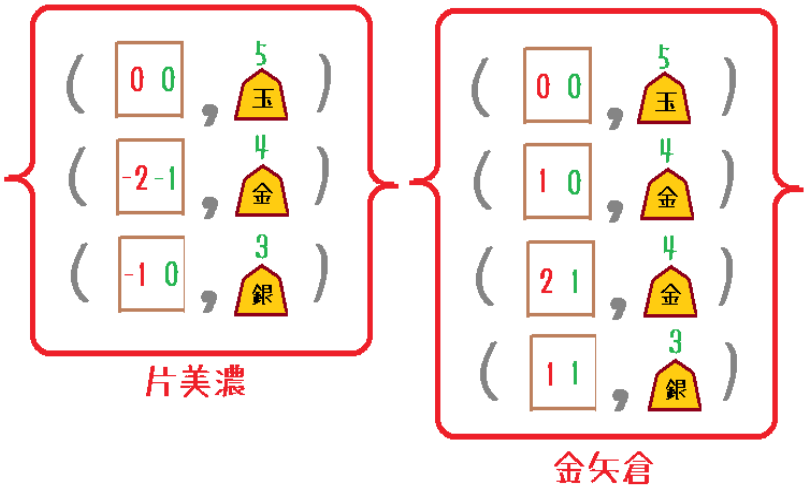






「 相対位置を力バーした　２９０マスを　メンバーとする　盤チーム　と
そこに乗っかる　２９もの　駒チーム　を　はっきりさせておこう☆」

「デカルト積を取って
囲い　を表現だぜ☆」



「ただのリストと
何が違うの？」

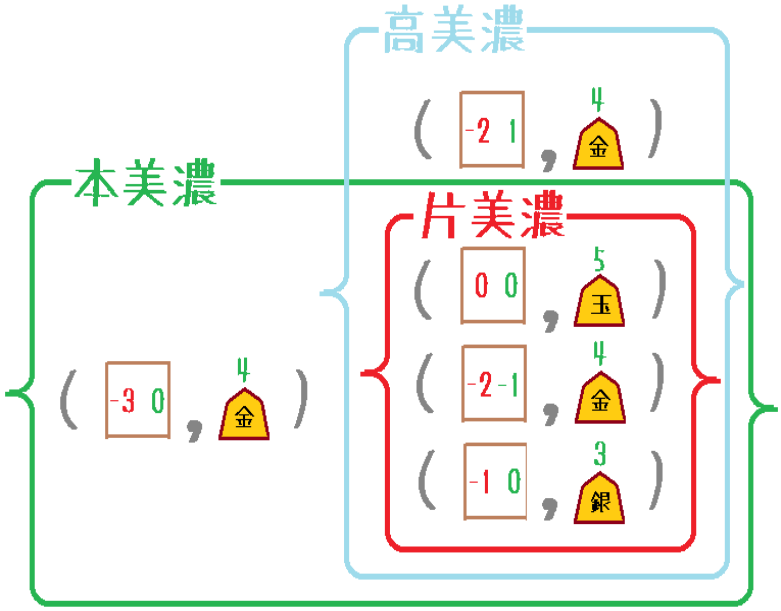
「リスト　も　書けるのが
集合論だぜ☆

片美濃は
本美濃や
高美濃の
真部分集合

と気づいて

これだ☆！

と思ったので
１番最初の
バージョンの
きふわらべ　で
実は　取り組んでいた☆」



「第２回電王トーナメント　出場時は
将棋の合法手生成を　お父んの考えた集合論　で　やったよな☆」

「なんで　止めちゃったの？
というか、　なんで　そこに戻るの？　と言ったらいいの？」

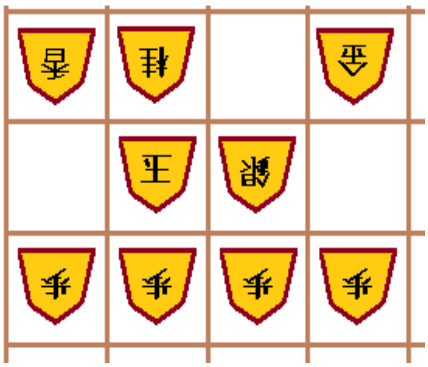
「実行速度が遅くて　２手しか読めなかったからだぜ☆

そのあと　Ｂｏｎａｎｚａ　の　本当の機械学習　とは何なのか☆
Ａｐｅｒｙ　が１００万局面とか　すいすい　読むのは
なぜなのか
実行速度が欲しくて　ソースコード　を読んでいた☆」

ENUKOMA TEIKEI

N駒定型　が探し求めた　答え　だぜ☆

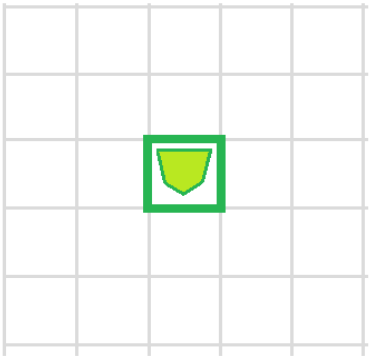
「集合　なんか使っても、　やれ　片美濃だ、
玉・金・銀と　駒３つの位置を　調べていたら
N駒関係と　やること
変わらないけどな☆」



「駒が　この位置にあるから
どうこう　したい……というのは
N駒が　得意とするところであり、

集合　を　使うなら
その表現力を　自由配置　を覚えることに
使っては　いけなかったんだぜ☆

配置は定型、
言ってしまえば　N駒定型　で
その　数え上げられる程度の数の　定型　１つ１つ
に対して　抽象的な表現力を与える、
という答えの探し方をしなければいけなかった☆」

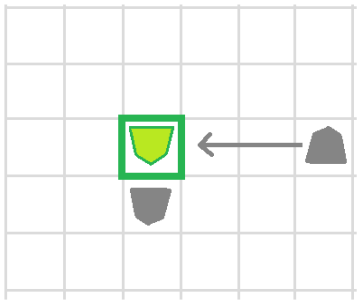


「ちから　の使い方　に気づいたのね」

座標の原点

何らかの変化を
与えたい相手駒

「ちから　を、
ものを好きなように動かす能力　としよう☆」



配置は、いくつかの定型

「自分の駒は　自分で動かせるんで……、

わざわざ　意図通りに　動かしたい
なんていう駒は　相手の駒　しかないんだが☆」

「狙い　を持ち、　相手の駒を

- 取る　とか
- 位置をずらす　とか
- 置いてもらう　とか
- 釘付けにする

といった　意図　を実現するための手段として
N駒定型　は問題のテンプレートとして使い、長手数の　手筋の組み立て
を自動立案できるように　集合　を　用いれないか　調べていこうぜ☆」

DATA KOUZOU

データ構造 と 操作 は 黄金コンビなんだぜ☆

CHOSHO TO CHOSHO

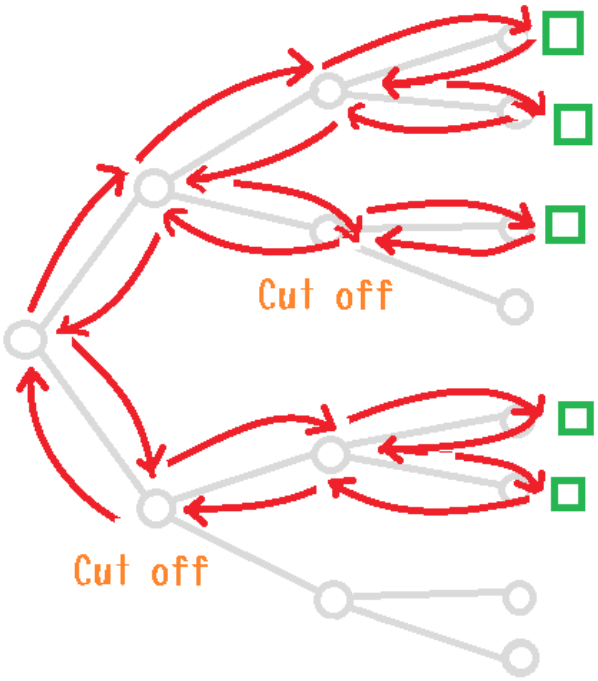
強いとこは 当然、長所と長所 を活かしてくるのよ？

「主流の 将棋ソフトは

探索に ストックフィッシュ、
末端局面評価に Ｎ駒関係、

ということで、枝を カット して
局面評価 の回数を少なくするか、
局面評価 を高速化することで

２年前の 大樹の枝（たいじゅのえだ） の場合でも
家の オンボロ i3 ＰＣ（アイスリー ピーシー） で
３秒で １００万局面 ２０手読み
してくる☆」



「２０手という 深い読みを
たったの １００万局面 に
絞り込み、

瞬速の ３秒 で
強い指し手 を返してくるぜ☆」

合法手　局面
探索　局面評価

「その評価値も もっと深い将来の数手先の 評価 を盛り込んであるから、
探索した末端局面の、さらに先 を見ているのよ」

「そして Ｎ駒関係は どうやって
高速な局面評価を実現しているかというと☆」

「将棋の 親子局面、兄弟局面 は
駒４つ ぐらいしか違わないので☆、」

「隣の局面を　評価するには

N駒関係の　動いた駒が絡む
部分点だけを　引いたり　足したり
するだけでいい☆」

「部分点にしてるのが
メリット　なのよね～」

「兄弟局面に評価値を付ける
N駒関係に比べて、

集合　は、
何が仲間で、何が仲間でないか

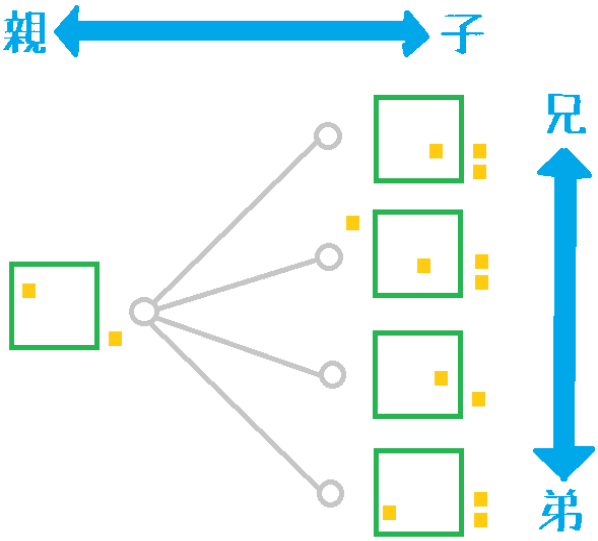
を表現することが
得意なんだぜ☆」

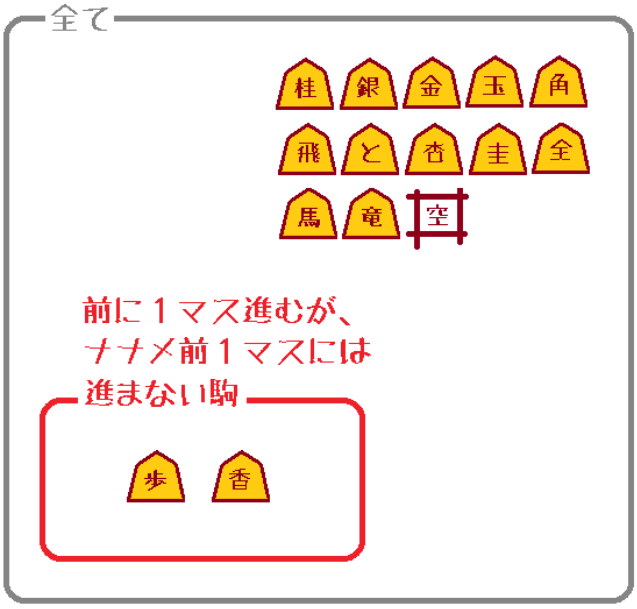
「じゃあ　集合　は
ストックフィッシュ探索
と組んでも、
ストックフィッシュの探索性能を
活かせないってこと？」

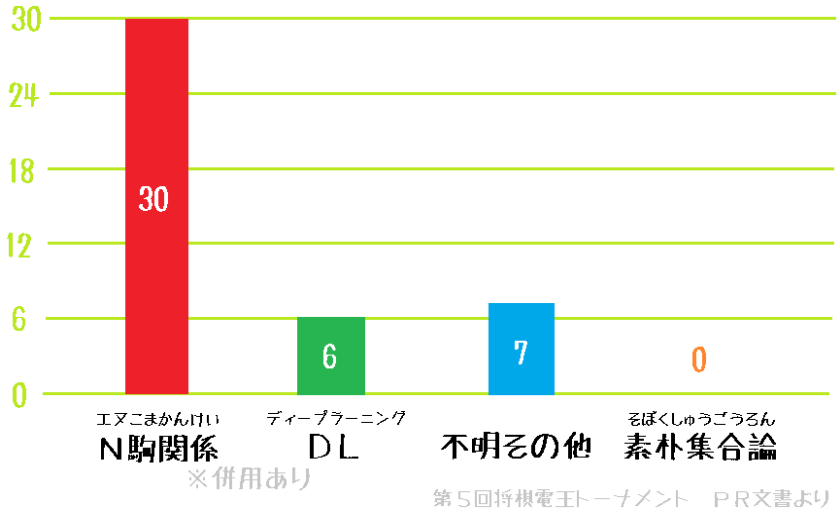
「別の　相方が
必要だな☆」

「データ構造を
全く変えれば
探索も変えるのは
当然だな☆」

「ディープラーニング
とかも　そうよね～」







第５回将棋電王トーナメント　ＰＲ文書より

SAITEIGI

将棋のルールを　再定義　するのね

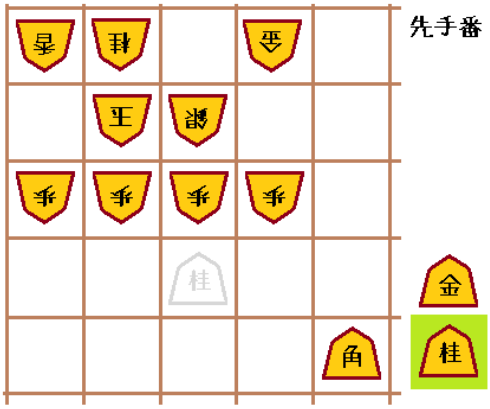
KURIKAESU　DAKE

１つのことを　繰り返すだけ　で勝てだぜ☆

「 将棋というゲームを、

お互いにとって要らない駒を渡して
お互いにとって欲しい駒をもらうゲーム

と　再定義　するぜ☆」



「 玉を囲うとか、駒の働きとか、
入玉するとか
金銀の厚みで　押し上がるとか
飛車を成り込むとか　考えなくていいの？」

「 ここが差異点なんだが、
将棋の元の定義から　何を　止めよう　というと、

駒を動かすゲーム　であることを　止めよう

ということだぜ☆」

「 トレーディング・カードゲームに　してしまう　わけかだぜ☆」

「 複数の異なる　価値の系統　があって
２つの系統の間で　比較が　できない場合、
その調停に　またルールが必要になってしまう☆

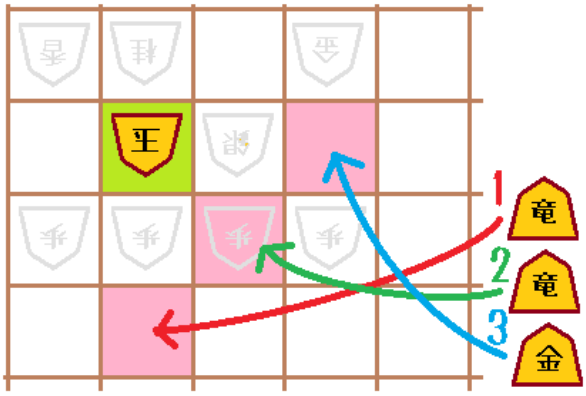
玉を詰ます　全体としての　目標も、
形勢を良くする　途中での　目標も、
単純なルールの繰り返しで　どちらも通過点に
過ぎない、という　１系統　に持っていく　構想だぜ☆」

「 何が欲しい駒で、何が要らない駒なのか
どうやって決めるんだぜ☆？」

「 最初に　狙う駒　は　玉　で、　以降　狙う駒　は変えていくが

その　狙う駒　を取るのに　十分（じゅうぶん）　な手駒が、
わたしの　欲しい駒　で、
そうでない駒は　玉は除くが、要らない駒　だぜ☆」

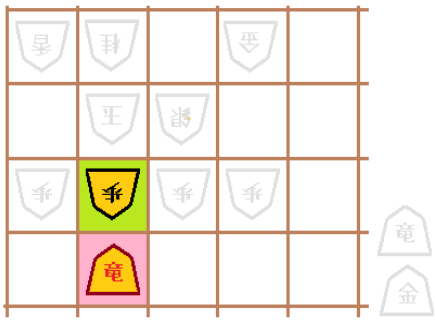
「竜・竜・金　…　と強い駒から
置いて　いけだぜ☆
これを　楽観筋（らっかんすじ）　と定義するぜ☆」



「狙う駒　以外の相手の駒は
無いものとして　考えていいの？」

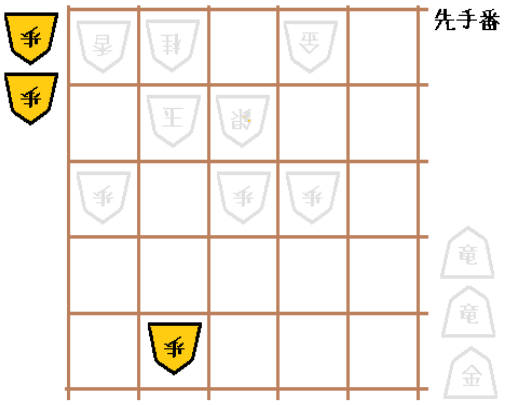
「楽観筋　が最初に決まることで
邪魔駒（じゃまごま）　が定義できる☆

こうやって　駒配置全体は相手にせず
つねに　問題１ケース　と　候補の手筋
に　分解　していくんだぜ☆」

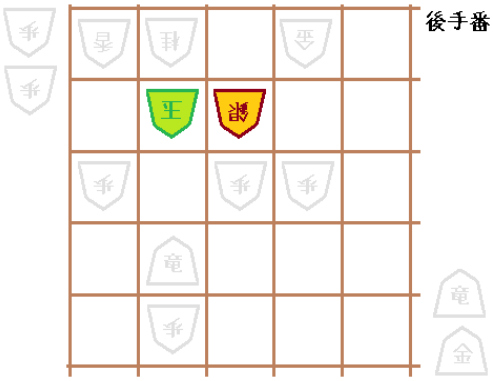


「“歩２枚で叩きの歩の手筋カード”が
登録されていたら　それを使い、

無かったら　駒を置きたいところから
邪魔駒の利きが　消えるまで、
力まかせ探索して
“新しい動きのカード”を作成し、
新規登録してから　使えだぜ☆」



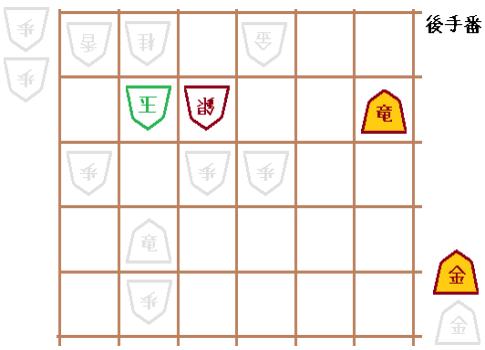
「次は　玉　が避ける他に、
銀　が　休み明けで仕事に復帰して
竜の利きを　塞いでくる一手が
邪魔だな☆」



「じゃあ　やりたいことは
“銀の　釘付けカード”　か☆」

「あとで使おうと思っていた
竜　で　銀をピンするぜ☆

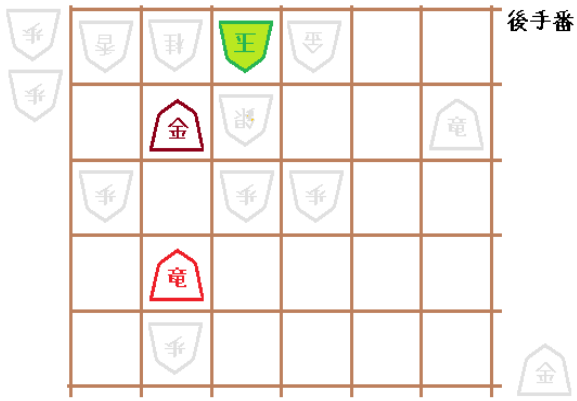
代わりに持ち駒には
金を追加だぜ☆」



「 合いごま　バンバン　打ってこられたり、
手抜き　されて　手筋が　調子狂ったり、
邪魔駒が増えたことで　早逃げ　ができるように　なっていたら
どうするの？」

「 N駒定型　で　どれぐらい
集合論を用いた抽象的な表現を
扱えて　“カード化”できるのか、

合いごま　に使える駒の案内、
早逃げした先の　N駒定型への案内、
とかは　次回構想で　やろうぜ☆」



「 現局面で　玉を取る
楽観筋の１つについての

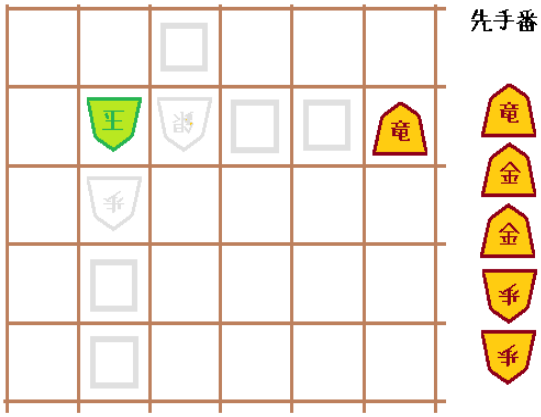
らっかん　みつもり
楽観見積もり　は

設備投資　が
竜・竜・金・金　で、

返品　が
歩・歩、

売上げ　が
玉

と定義しよう☆」



「 売上げが　玉　なら見合っているな☆」

「 じゃあ　玉　の　引用数　は　１　、
竜・金・歩　の　引用数　は　それぞれ　２　として、

まだ　楽観見積もり　を取っていない盤上の駒を
楽観見積もり　しろだぜ☆」

「 二枚竜　の楽観筋で　ほとんど　取れるのよ！」

「 二枚竜　を設備投資して　歩　を取ってたら　損　だろ☆
引用数の少ない駒で　引用数の多い駒を取ることを
利益　としろだぜ☆」

PATH

楽観 パス を評価しろだぜ☆

JUBUN SOZAI

十分素材　を考えるプログラムなのね

「 さっきの章で　将棋を再定義し、

楽観見積もり　で　引用数　を算出し、
引用数の低い駒で　引用数の多い駒を集め、
その繰り返しで　玉　を取る

という　構想　を示したぜ☆

この研究対象を　ざっくり　　楽観素材錬成ツリー（らっかんそざいれんせい）　とでも
名づけよう☆」

「 黙って　N駒関係　にしておけば　いいのに……☆」

「 引用数の少ない設備投資よね」

「 勝負ごとなんで、一番　勝ちやすい　経路（けいろ）　を知りたいだろう☆

玉を取るために　十分な駒を　すべて揃えるために、
トータルで　一番利益の高い　取引の集まりを
楽観パス　、そのときの利益を　楽観利益（らっかんりえき）　と呼ぼうぜ☆」

※パス　…　経路。なぜ英語にしたのか

「 集合型　将棋コンピューターは　局面評価値　ではなく、
楽観利益、つまり　駒を交換していけるか、
を　評価するものとするぜ☆」

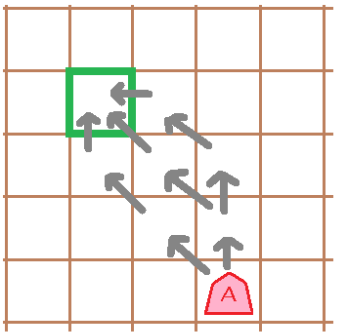
「 駒が　交換できるかどうか　教えてくれるだけなら、
盤上の位置的に　むずかしい　のもあるんじゃない？」

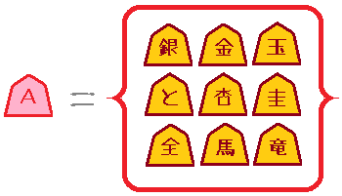
「 すべての駒を　持ち駒と考えて　バンバン　撃ち合う
ドッジボール構想　だぜ☆
盤上に置いてしまった駒は　すべて　リスク、
１マスずつ　ずらしていくしかない☆」

現局面の相手の楽観利益　－
現局面の自分の楽観利益　＝　撃ち合いの形勢

「駒を　撃ち合い現場まで
運んでいく方法を教えてくれないのか☆
ゲームツリーの探索や、
駒の働き　をよくすることが
目的の動きは　ないのか☆？」

「駒を動かすのは無くなって、カード・ゲームの
移動のカード　になったと思えだぜ☆
移動のカード　は、
狙いのカード　に　ぶら下がっている☆」





「自分の　狙い　ばかり見ていて、任務中の駒を
ぽろぽろ　取られたりしない？」

「手筋に　はまりやすい　マスと駒の　逆引き検索（ぎゃくびけんさく）　は
用意したいよな☆
危険マスを避けて　駒を　１マスずつ　ずらして　いけだぜ☆」

「予算の無くなったプロジェクトみたいだな☆」

「現局面　固定で　すべての駒の取り方を　楽観的に調べているけど、
１つの駒を取ったあとは　現局面の　駒の位置　がだいぶ
変わっちゃうんじゃないの？」

「それも　合いごま　や　逆引き検索　と一緒に、
次回構想　に持ち越しで☆」

「次回構想　って何だぜ☆」

「無駄な手であがいてしまう　水平線効果　は無いかもしれないが、
目の前の実現性が見えていない　足元効果（あしもとこうか）　は有りそうだぜ☆」

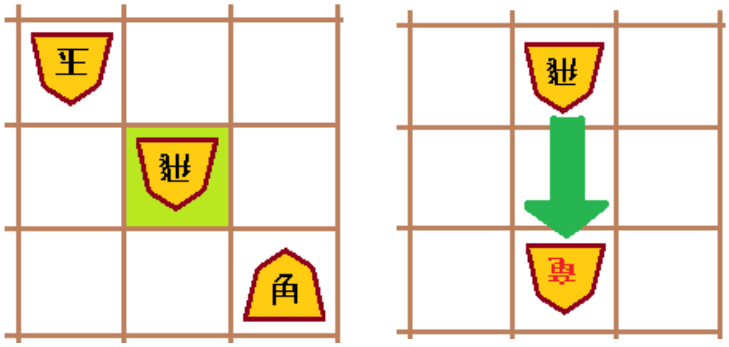
RAKKAN PATH

どうやって　楽観パス　を算出するの？

晩飯の　KAIMONO 買い物　と同じだぜ☆

「カレー玉ライス　を食べようと
思ったら　竜ニンジン　と
金イモ、歩タマネギ　が必要だぜ☆」

「竜なんか取ってこれないんだけど」



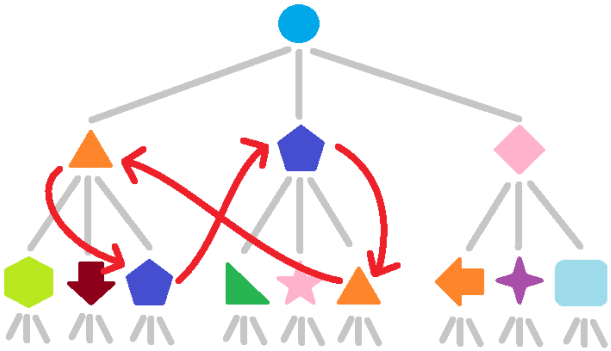
「“ピン・カード”とか
“両取りカード”を使って
飛車を持ち駒にしろだぜ☆　そのあと
“成りカード”と　“移動カード”で
竜を運んでこいだぜ☆」

「緊急時の資材調達マニュアルが必要だぜ☆」

「ショッピング　じゃなくて、
狩り　なんじゃないの？」

「循環していて永遠に取れない
ニンジンとか　あるだろ☆」

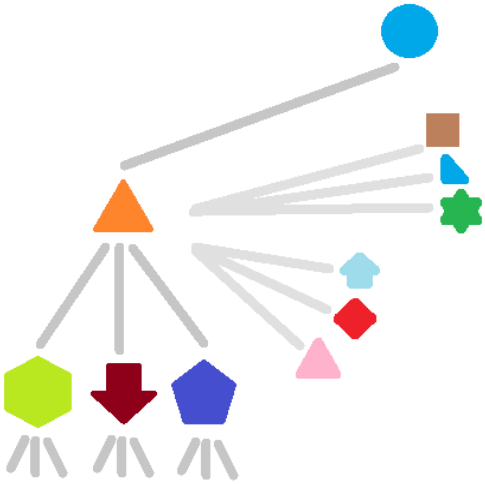
「駒なんか
取れないじゃないか☆」



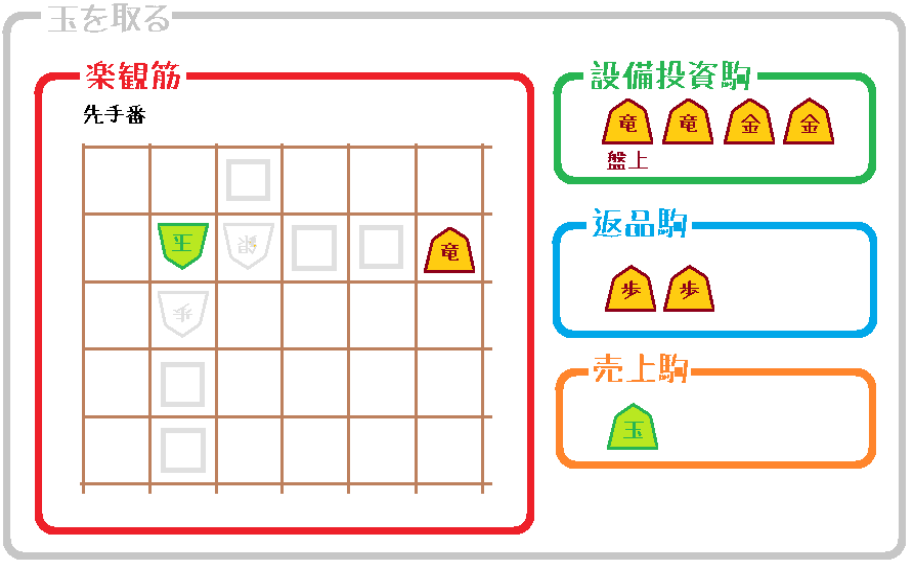
「自分と相手が　駒を動かしている内に
どちらかが　ヘタして
盤上が変わって　駒の　引用数　が
変わっていくはずだぜ☆」

「じゃあ　駒なんか　ヘタに動かしたく
ないじゃないか☆」

「１つの“駒を取る狙いスレッド”にも
いくつもの方法の“カード”が
ぶら下がっていて、いったい
どれぐらいの分量になるの？」



「組み合わせ爆発してるだけで、
分解していけば　数えるほどの分量しか
ないのではないか、
ということに　期待している☆」

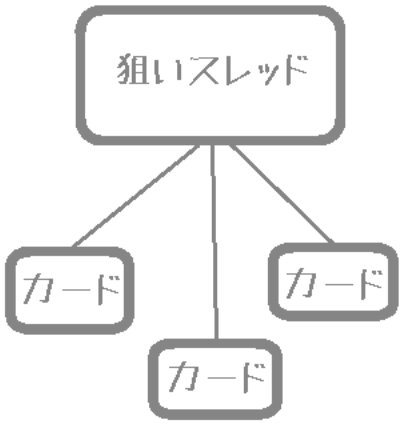


「“玉を取る狙いスレッド”の　邪魔駒　が　図の通りだけなら
竜　竜　金　金　歩　歩　で
片美濃を　上から潰せるのかだぜ☆？」

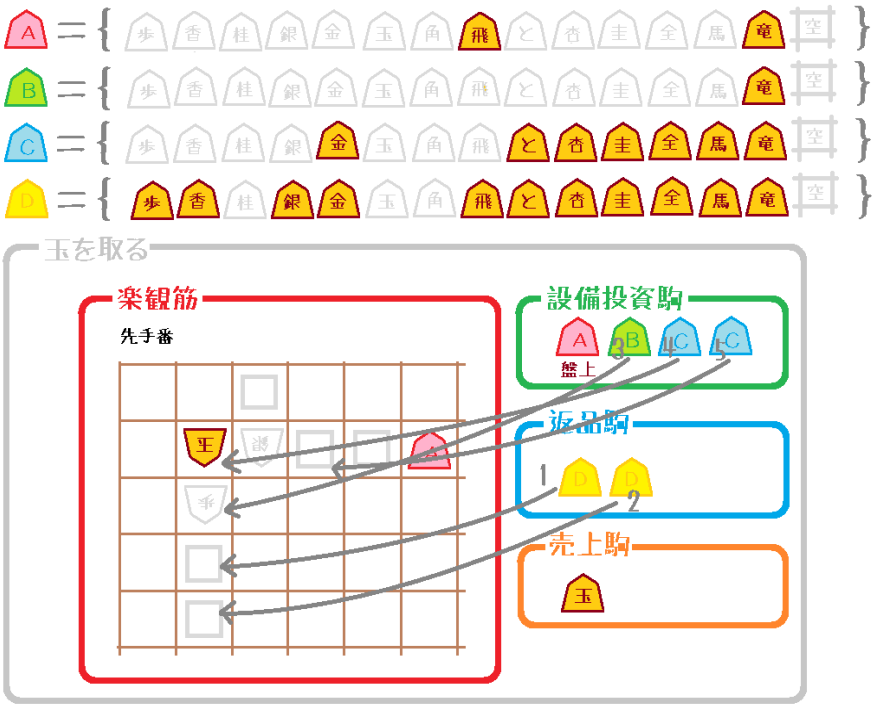
「楽観的すぎると　言いたいんだろ☆
合いごま、手抜き、早逃げ、王手の仕返し　などを
考慮してないからだぜ☆」

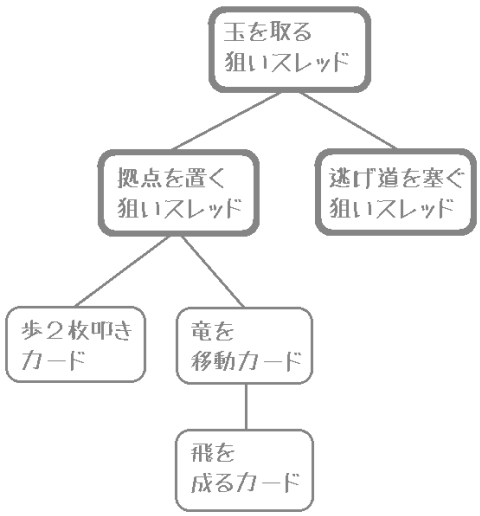
「今更だが　楽観筋　が　同型（どうけい）　のもの
すべて　を　1つに　おまとめ　して、
“カード”　と呼ぶとしよう☆

そして　全ての“カード”は
どこかの“狙いスレッド”に　ぶら下がっていて
使った“カード”は消していき
“狙いスレッド”を　満足　させていこうぜ☆」



「まとまんの？」





※グラフの解説　…　スレッドは集合、
カードはメンバーに相当。
スレッドが　フォルダーで、
カードが　具体的な行動。
葉（下の方）から満足させていく。

「集合　は　“狙いスレッド”と　“カード”の関係でもあり、
Aとか　Bとか、該当する　駒パターンを代表する
表現としても　使うぜ☆」

「“玉を取る狙いスレッド”を　木構造の根とし、
他の全ての“狙いスレッド”を　ぶら下げろだぜ☆
“カード”は　“狙いスレッド”に
ぶら下がる葉だぜ☆」

「設備投資駒、返品駒　を　費用　と定義しよう☆
この　費用の　A、B、C　といった　変数　が
引用数　になるんで、最小構成のパターンだけ　覚えろだぜ☆」

「お父んは　あまりに局地的な　パーツ作りをしているが、
駒を１つずつ　動かしてゲームツリーを探索するのではなく、
将棋の対局　まるごと１つ分　“狙いスレッド”と“カード”だけで
つなげるつもりかだぜ☆」

「多分　そうなのよね。
駒を　もっと細かな　利き　の単位へ　分解　して
盤上のストーリーを　利きと狙いの集合論　で
書き直せると思ってるのよ。　困ったものだわ。
盤上は　さながら　マルチ・シナリオも　いいところなのに……」

「引用数　を出すというが　マルチ・シナリオ全体なのか
１本のシナリオごとに出すのかも　分からないぜ☆」

「見積もりとは　ざっくりしたもの　だぜ☆
読みのような　不確定なものに　正確性を求めても　意味はない☆
自分の能力も含めた　期待値というか、

計算して　指すのではなく、
きふわらべ　には　物の考え方　を与えるので、

答えのないものに対して、自分の得意を知り、
自分は　こう思った、だから　こうした、というような
意見のようなものを持てだぜ☆」

「ただの道具に向かって　無茶を言う☆」

「”思考”するチャンク式コンピューターをやりたいのよ、この人。
映画の中の　全ての演技に意味はある、みたいな。
それも答えのあることに　”正しい”計算結果を　返すんじゃなくて

何か　１つの物の見方を獲得して
それに従って　アレンジし、
１つの体系として統一されたものを創って　投げ返してくるような」

「簡単に言う☆
楽観パスは　計算で　出すんじゃないのか☆ｗｗｗ」

「勝ち方のアテも付けて
きふわらべ　も
カレー玉ライスを　作れるように　ならないとな☆」

RUST

ラスト言語　で書こうぜ☆

MEMORY NO KANRI

メモリーの管理　が　関心の対象　なのよね

「Ｌａｓｔ（ラスト）　じゃなくて　Ｒｕｓｔ（ラァースゥー）　な☆
最後じゃなくて、　錆（さ）びついている方だぜ☆」

「なぜ　Ｃ＋＋　という　最速　を手にしないのか☆？
コンピューター将棋勢の半数以上は　Ｃ＋＋　だと言うのに
真似しぃのお父んも　そこは　真似しないんだな☆

メモリーの　安全な操作性　は　Ｃ＋＋　より後発の
Ｒｕｓｔ　の方が　進んでいるという話しも見るが☆
そうか　お父んの四次元コーディングでは　メモリリークが……☆」

「べつに……☆
平岡っちが　ストックフィッシュ　をいじれば　わたしもいじるし
平岡っちが　ヨーグルト　を食べれば　わたしも食べるし
平岡っちが　Ｒｕｓｔ　にしよかなと呟（つぶや）けばわたしもするんだぜ☆」

「今までに　キャッチアップ　できたのが
ヨーグルト　だけなのよね」

「わたしのような、一見　論理的に動かない人の行動を予測するのに
良い方法が　統計　だぜ☆
偶然というには　ちょっと回数が多いんじゃないかというところは
何かの原因があるかもしれない☆
わたしは　わたしが簡単だと思ったこと　ばかりやるぜ☆」

「ビデオゲームでも　プレイが　上手い人ほど
比例して　より簡単そう　に見えるんだぜ☆

お父んが考える　より簡単なことから始めよう　という戦略では
優先して　むずかしいところ　に飛び込むことになるんだぜ☆」

「そんな話し　聞いてないよな☆」

「気づいたのね」

MAWARIKOMU

流行りには　過ぎ去る前に　回り込むのよ

JYUNEN

向こう１０年　遊ぶ　計画なんだぜ☆

「－ Ｂｏｎａｎｚａ６．０ の　ｌｅａｒｎ（ラーッン）　や
Ｒｏｔａｔｅ　Ｂｉｔｂｏａｒｄ（ローゥテイドッ　ベッボーッィド）

－ Ａｐｅｒｙ（大樹の枝）　の　ｇｅｎｅｒａｔｅＭｏｖｅｓ（ジェネレイッ　ムーブスゥー）

－ Ｓｔｏｃｋｆｉｓｈ７　の　ｓｅａｒｃｈ（サァーッチィ）

などが　コンピューター将棋県の　地元の名産物かと思って
オープンソース名所めぐり　をしてきたわけだが……☆」

※発音　Ｇｏｏｇｌｅ翻訳

「お父んの中で　有名だと思っているものが　有名なのだろう☆
お父んの中ではな☆」

「有名かどうかより、　Ｒ（アーゥ）　（レーティング）が
１００　上がるかどうかを　重要視しなければいけないのよ」

「電王戦ＩＩ　に　和服で来た　アマチュアの　おっちゃん（※）が
ノーパソで　木造の盤駒を持ったプロと　戦っている写真を見て

今後１０年は　あれで遊べる　と思ったら
まだ５年目だぜ☆
コンピューター囲碁も　神　が降りてくるし、
Ｐｏｎａｎｚａ　も引退してしまった☆」

※簡単そうにノーパソをいじっている

「遊べると思ったから、で　本当に　遊んでしまうことを　計画とは
呼ばないんだぜ☆　まず　気持ちを整理し　本当にやり」

「 Alpha Zero が elmo を倒したとか　言ってるらしいぜ☆」

「 次の名所が　決まったわね」

「 挑戦状を受け取る準備をしておこうぜ、
分かったか、　Ｋｉｆｕｗａｒａｂｅ（キフゥ　ウォラ　ベェー）☆」

「 たいことは何かを考え……☆？」

＞＞＞ Ｄｏｓｅ　ｔｈｅ　ｏｐｔｉｍｉｓｔ　ｓｅｅ　ｔｈｅ　ｅｎｄ　ｏｆ　ｔｈｅ　ｈｏｌｉｚｏｎｔａｌ　ｌｉｎｅ　ｆｒｏｍ　ｔｈｅ　ｐｏｓｉｔｉｏｎ　ａｐｐｒｏｘｉｍａｔｉｏｎ☆！？

762alpha（旧President_X）アピール文書

2018年1月21日 V0.1

2018年2月11日 V0.2

天野史斎

■ まえがき

762alphaは前回までPresident_Xと名乗っていたソフトの後継ソフトです。

というより、やろうとしていることと実現方針は全く同じで変わっていません。

過去2回はプログラムの完成度の問題でたいした成績を残せていませんが、今年はどうなりますやら

なお過去のアピール文書はわざと実現方針をわかりにくくわかりにくくぼかして書いてみました

ｽﾐﾏｾﾝ；；

当ソフトの目的： 学習を自動化し、コンピュータ将棋プログラム開発への人間の介在を不用にする

ひた隠しにしてきたキーワード：局面の場合分け

■ 問題提起

[既存のコンピュータ将棋]

（特徴）

1. 優れたプレイヤー同士の棋譜を学習サンプルとする。

2. 学習サンプルで評価関数を調整する。

3. そのまま対戦時の局面評価にも使う。

（問題点）

1.「優れたプレイヤー同士の棋譜」が必須である割に、「優れたプレイヤー同士の棋譜」とは何である

かがシステムの中で十分には規定されていない。

　このため学習サンプル集めに人間（システムの外の存在）の関与が必要

2. ボナンザメソッドの結果を見てパラメータ数をスケールさせるのも人間（PPT、KKP＋KPP、KPPP、KPPT、KKPT...人間がこれらを進化させている。

3. 異なる棋風（最善手（必勝手）が複数ある局面での選択の好み）のn駒間の関係ベースでのグローバルな合算が現実の調整手段で常に最良の結果をもたらすのか、確定的な結論は出ていなさげ

■ 提案する方法

[762alpha（および旧President_X）]

（特徴）

1. 全力で戦って負けたケースの棋譜を学習サンプルとする。

2. 学習サンプルから局面の場合分けを学習し、評価関数の調整は場合毎に行う。

3. 対戦時の局面評価には軽量な評価関数を使う（1つの場合の中の形勢だけ説明できれば良い

（既存手法の問題点を解決する手段）

1.「全力で戦って負けたことを学習サンプルの条件とする」という明確な基準がシステム内にあることから、

　学習サンプル集めが真に全自動化される。（「優れたプレイヤー同士の棋譜」が何であるかについて悩む必要が無い

2. 学習サンプルの数が増えるにつれ局面の場合分けが勝手に細分化することで、パラメータ数のスケール相当の作業も全自動で進行

3. 対戦時の評価関数は1つの場合の中の形勢だけ説明すれば良いので、個々の場合の中で異なる棋風の交絡は起き難い。

なお、学習サンプル集めは特徴1さえ満たせばオンライン学習でもバッチでも可能

■ 局面の自然な場合分け

終盤の局面から場合分けを学ぶ。次の通り、終盤の局面は考えやすい。

1) 終盤の局面は、玉に睨みを利かせている駒（必要な駒）と、そうでない駒（不要な駒）からなる。
　必要な駒の集合をキーにして分類すればまず間違い無い分類になる
　- 必要な駒は取られては困るので高い得点とし、不要な駒は捕られてもかまわないので低い得点とすれば、
　　勝利という目的に局面評価を自然に整合させられる。

2) 終盤より前の局面においては駒の要／不要がはっきりせず、うまい分類キーを見出せそうにない。
　これは当方の終盤力が継続的に上昇し、「終盤の局面」の範囲が拡大し続ける ・・・（仮定1）
　という仮定の下で、考えないこととする。

■ 終盤局面内の「必要な駒」の抽出
局面をbag of KPPsとみなしてベイジアンフィルタを適用する。
このときの勝利/敗北の教授信号（局面に対する勝利/敗北のラベル付け）は、
「局面別の勝敗推定」で述べる手段で推定して与える。

勝利にとって必要なn駒間の関係は、勝利時に頻出するため、対戦を重ねるにつれ勝利条件として認識されるようになる。
そうでないn駒間の関係は、勝利時と敗北時の両方に現れるため打ち消しあって無視されるようになる。
（言うまでも無く前者が「玉に睨みを利かせている」n駒間の関係である。

勝利と敗北の出現頻度が違っても、両方が1回以上現れたなら、ベイジアンフィルタは打ち消すべきものを打ち消してくれる。

あとは、n駒間の関係を駒の要不用判断に落とし込む。
すなわち、ベイジアンフィルタによる得点（log(P(当方の勝利¦n駒間の関係(i,j,k))/P(当方の敗北¦n駒間の関係(i,j,k)))を

(駒の種類, 手番)毎に合計すれば、その値の大きさで(駒の種類, 手番)が「玉に睨みを利かせている」駒か否かﾜｶﾙ

※ 個人的にはベイジアンフィルタの上記性質は終盤局面の良し悪しを盤面の正規表現で表すという捨てたアイデアがリバイバルした感じでたいへん喜ばしい

なお、特徴1の通り、当方法では当方の負けた対局からしか学習しないので、勝利の学習サンプルを得るために次の仮定をおく。
　当方が全力で戦って負けたのであれば、相手の指し回しから学習すれば（同じ負け方が回避されることにより）強くなれる ・・・（仮定2）
この仮定の下では、相手の指し回しは勝利の学習サンプルとみなせる。

■ 局面の場合分け
局面ごとの勝利/敗北の教授信号が与えられる前提で（その手段は「局面別の勝敗推定」で述べる
　局面（盤面）を適当な固定長ベクトルに変換
すれば、いかような手段でも教授信号に従い局面を分類できる。

分類器はここでは線形分離×決定木を使う。

盤面の適当な固定長ベクトルへの変換については…
前回は
 玉に睨みを利かせている駒を知りたいのだから駒の座標に注目すべきである
 → 任意の座標変換を表現し得るベクトル表現にしよう（座標の差や平均が勝手に特徴になるﾊｽﾞだ
と考えて変に複雑化してた；
（置駒・持駒・上手/下手毎の駒の個数が変動するから、
 盤全体の駒の座標は形勢判断にとって意味のある形ではたいへん固定長ベクトルにし難い

■■■■{■■■■}■{■■}■■■■■xx■■■■■■■■■

[redacted]([redacted], [redacted])

[redacted]

■ 局面別の勝敗推定

当方が全力で戦って負けたとして、投了付近では

1. +の評価値の期間が連続
2. -の評価値の期間が連続
3. 投了

という経過を辿るはずである。（頓死等、2の期間の長さが0手のケースもある。）

1の期間の当方手番局面は、当方が形勢判断を誤った局面と考えられる。

ただし、相手が優勢だったとする確証は乏しい。

（∵当方の形勢判断能力不足で相手が助かった可能性がある。

2の期間は、当方が全力で戦って評価値を覆せなかったのだから、相手が優勢だったのだとかなり確信を持って言える。

少なくとも仮定2の下では、相手手番を勝利の学習サンプルとみなして問題無い。

よって、2に属する局面の当方手番を「敗北」、相手手番を「勝利」とラベル付けして学習サンプルとする。

■ 評価関数の調整

上記1の期間に属する当方手番局面を分類器で場合分けし、その場合に対応する評価関数の評価値を下げる調整を行う。

（ボナンザメソッド風にパラメータを振って探索し、評価値が下がるポイントを探す

■ その他

Futility purningやnull move purningのしきい値を自動調整するしくみを備えています。

（一定回数ごとにpurnせずに探索し、結果を蓄積→ヒストグラムの上側x %点をしきい値にする。

■ F.A.Q.

Q1. 最適化の行き届いた探索部を有するソフトに勝てますか（・∀・）？

A1. 機械が全自動で行う局面の場合分けのたゆまぬ細分化とpurningしきい値の自動調整によりそのうち勝てる日も来るのではないでしょうか。

（対戦相手が開発に飽きるまで続ければ。）

Q2. 評価関数はどういうものですか。

A2. 駒の移動の可能性は探索でもって確かめるとして、評価関数では

駒がそこに居る価値

駒の攻撃可能性

だけを評価します。

具体的には次の通り。

```
///  - 相手玉への利き
///  - 相手玉の退路への利き （退路＝相手玉の利きの中の空白マス）
///  - 相手玉の守りへの利き （守り＝相手玉の利きの中にいる相手の駒）
///  - 相手玉の突破口への利き（突破口＝相手玉の利きの中にいる当方の駒）
///  - 相手玉に近い段への利き
///  - 相手駒への利き
///  - 自駒への利き
///  - 相手からの利きに対する反撃可能性
///  - 相手からの利きに対する当方利きの優越
```

Q3. 最適化の行き届いた探索部を有するソフトにいつ勝てますか（・∀・）？？

A3. わかりません。

Q4. Bag of KPPsとおっしゃいますがどのKPPを使うんですか（・∀・）？？？

A4. 普通のKPPです。すなわち、局面を

- 当方の王K に対する当方の王以外の2駒PPの位置関係
- 相手の王K'に対する相手の王以外の2駒PPの位置関係

のbagとみなします。（駒台も位置に含める。また、PPはP=Pのケースも含む。）

Q5. 普通のKPPでKPPTやKPP_KKPTに勝てるんですか（・∀・）？？？？

A5. KPPベースのボナンザメソッドとBag of KPPsのベイジアンフィルタを比較すると、
パラメータの自由度が（前者）＜＜（後者）であるため、
ボナンザメソッドのKPPが、よりパラメータ数の多いKPP亜種に変更されたとしても、
後者で対抗できる余地があると考えています。

ボナンザメソッドは

- 学習結果が探索結果として妥当となること

という制約条件を含むことから、同じKPPが、

- ある親局面の子に現れるときは兄弟局面に対して評価値を最大化することに寄与
- 別の親局面の子に現れるときは兄弟局面に対して評価値を最大化「しない」ことに寄与

という2役を兼ねる必要があります。つまり、ボナンザメソッドにおいては、
任意のKPPは、兄弟局面として現れ得る大量の他のKPPとの関係において、
とれる値の範囲が（おそらくかなり厳しく）制限されます。
ということは、ボナンザメソッドのKPPおよびその亜種は、
独立パラメータ換算にすると見かけほど複雑なモデル表現ではない
（特徴空間の次元＜＜1局面に現れるKPPの数）である可能性がある。

一方、ベイジアンフィルタは「勝った」「負けた」の事実の積み重ねを記録するだけで

機能するので、そのような制約はありません。

つまり、特徴空間の次元≒1局面に現れるKPPの数、と看做し得ます。

（去年のアピール文書で力説したことの言い換え。

※ 実際には扱いやすく、意味のある固定長ベクトル化をする目的で制限した次元数にします

Q6. いいことづくめに聞こえるBag of KPPsのベイジアンフィルタですが

流行っていないのはなぜか考えたことはありますか（・∀・）？？？？？

A6. ありません。

モデルに対して学習データがスパースなのはボナンザメソッドのKPPも

ベイジアンフィルタも同じなので、挫折した先駆者が居たとしたら

平滑化に難儀されたんじゃないですかね（適当

Q7. Bag of KPPsのベイジアンフィルタと評価関数の関係が相変わらずよくわかりませんが（・∀・）？？？？？？

A7. Bag of KPPsのベイジアンフィルタの結果でもって

f: {学習データ内の局面} → { 特徴ベクトル }

という写像を改訂して特徴空間内での局面の場合分け能力をグレードアップしていき、

グレードアップされた場合分け能力でもって分けられた場合毎に

局面の評価と評価値の調整をやるということです。

Q8. ありがとうございます。たいへんよくわかりました。お薬を増しておきますね。

A8. いいえ。

以上

AlphaZeroの論文を参考に将棋のルールだけを教えて自己対戦だけ

ディープラーニングをしています。

学習にはCaffeを使って50万棋譜の中から

64局面(Minibatch 64)をランダムに取り出しています。

学習に使ってるのはGTX 1080です。

特徴はAlphaZeroと同一ですが、手数の情報だけは使っていません。

プロの棋譜から学習させたときに、手数なしの方が収束が速かったためです。

特徴は

局面(現在＋過去7手前まで8種類)

駒の配置

持ち駒

繰り返しの回数(3回まで)

手番

で45*8 + 1 = 361種類。

Bonanzaは盤、駒のデータ構造のみを利用してます。評価関数は使ってません。

Alpha Zeroの論文

https://arxiv.org/abs/1712.01815

- - - - - - - - - - - - - - - - - - - -

入力データのサンプル

後手の持駒：

９ ８ ７ ６ ５ ４ ３ ２ １

+--------------------------+

¦・v金 ・v飛v銀v玉v角 とv香¦一

¦v香 ・ ・ ・ ・v銀 ・ ・v香¦二

¦v桂 ・v歩 ・ ・ と ・ ・ ・¦三

¦ ・ ・ ・ ・ ・v金v銀 ・ 歩¦四

¦ と ・ ・ 歩 ・ ・ ・v桂 ・¦五

¦ 歩 歩 銀 ・v歩 ・ ・ ・ ・¦六

¦ ・ ・ 歩 ・v圭 ・ 桂 金 香¦七

¦v馬 ・ ・v金 ・ ・ ・ ・ ・¦八

¦ ・ ・vと ・ 歩 ・ 飛 ・ 玉¦九

+--------------------------+

先手の持駒：歩六

- - - - - - - - - - - - - - - - - - - -

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000010 000000010 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000001 000000001 000000001 000000001 000000001 000000001 000000001 000000001

000100000 000100000 000100000 000100000 000100000 000100000 000100000 000100000

110000000 110000000 110000000 110000000 110000000 110000000 110000000 110000000

001000000 001000000 001000000 001000000 001000000 001000000 001000000 001000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000010000 000010000 000010000 000010000 000010000 000010000 000010000 000010000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000001 000000001 000000001 000000001 000000001 000000001 000000001 000000001

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000100 000000100 000000100 000000100 000000100 000000100 000000100 000000100

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

001000000 001000000 001000000 001000000 001000000 001000000 001000000 001000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000010 000000010 000000010 000000010 000000010 000000010 000000010 000000010
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000100 000000100
000000100 000000100 000000100 000000100 000000100 000000100 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000001 000000001 000000001 000000001 000000001 000000001 000000001 000000001

000000010 000000010 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000001000 000001000 000001000 000001000 000001000 000001000 000001000 000001000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
100000000 100000000 100000000 100000000 100000000 100000000 100000000 100000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
001000000 001000000 001000000 001000000 001000000 001000000 001000000 001000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000010000 000010000 000010000 000010000 000010000 000010000 000010000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000001 000000001 000000001 000000001 000000001 000000001 000000001 000000001
100000001 100000001 100000001 100000001 100000001 100000001 100000001 100000001
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
100000000 100000000 100000000 100000000 100000000 100000000 100000000 100000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000010 000000010 000000010 000000010 000000010 000001010 000001010 000001010
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000010000 000010000 000010000 000010000 000010000 000010000 000010000 000010000
000001000 000001000 000001000 000001000 000001000 000001000 000001000 000001000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000100 000000100 000000100 000000100 000000100 000000100 000000100 000000100
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

010000000 010000000 010000000 010000000 010000000 010000000 010000000 010000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000001000 000001000 000001000 000001000 000001000 000001000 000001000 000001000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000100000 000100000 000100000 000100000 000100000 000100000 000100000 000100000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000100 000000100 000000100 000000100 000000100 000000100 000000100 000000100
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000100000 000100000 000100000 000100000 000100000 000100000 000100000 000100000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000001000 000001000 000001000 000001000 000001000 000001000 000001000 000001000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
001000000 001000000 001000000 001000000 001000000 001000000 001000000 001000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000010000 000010000 000010000 000010000 000010000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 100000000 100000000 000000000 000000000 000000000 000000000 000000000
100000000 000000000 000000000 010000000 010000000 010000000 010000000 010000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

666666666 666666666 666666666 666666666 777777777 777777777 777777777 777777777
666666666 666666666 666666666 666666666 777777777 777777777 777777777 777777777
666666666 666666666 666666666 666666666 777777777 777777777 777777777 777777777
666666666 666666666 666666666 666666666 777777777 777777777 777777777 777777777
666666666 666666666 666666666 666666666 777777777 777777777 777777777 777777777
666666666 666666666 666666666 666666666 777777777 777777777 777777777 777777777
666666666 666666666 666666666 666666666 777777777 777777777 777777777 777777777
666666666 666666666 666666666 666666666 777777777 777777777 777777777 777777777
666666666 666666666 666666666 666666666 777777777 777777777 777777777 777777777

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 111111111
000000000 000000000 000000000 000000000 000000000 000000000 000000000 111111111
000000000 000000000 000000000 000000000 000000000 000000000 000000000 111111111
000000000 000000000 000000000 000000000 000000000 000000000 000000000 111111111
000000000 000000000 000000000 000000000 000000000 000000000 000000000 111111111
000000000 000000000 000000000 000000000 000000000 000000000 000000000 111111111
000000000 000000000 000000000 000000000 000000000 000000000 000000000 111111111
000000000 000000000 000000000 000000000 000000000 000000000 000000000 111111111
000000000 000000000 000000000 000000000 000000000 000000000 000000000 111111111

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000
000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

000000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000

111111111

111111111

111111111

111111111

111111111

111111111

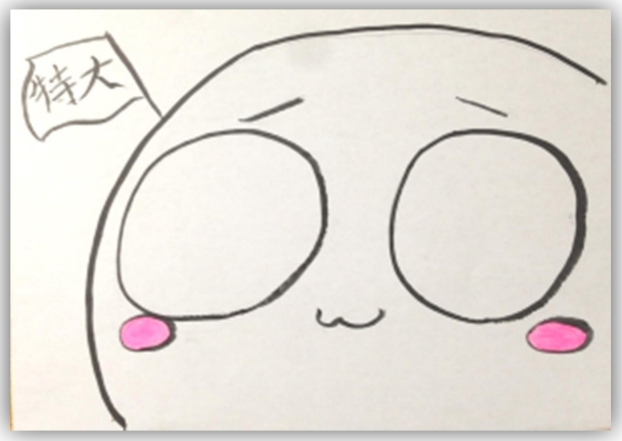
111111111

111111111

111111111

TMOQ アピール文書

2017 年 3 月 31 日　作成
2018 年 3 月 22 日　改訂



【ソフト名】TMOQ
“TMOQ”と書いて「特大もっきゅ」と呼びます。愛娘が命名しキャラクターデザインしてくれました。

【コンピュータ将棋大会実績】
WCSC26：　36 チーム中 15 位
WCSC27：　参加辞退
SDT5：　　42 チーム中 23 位

【使用ライブラリ】
今回は定跡の加工しやすさを主眼に、磯崎元洋氏の「やねうら王」ライブラリを使わせていただいています。

【特徴】
今回は定跡/戦型にこだわりを持っての参加です。
前回大会までの TMOQ の棋譜を見直して、自分の分身たる TMOQ が、自分の指すはずの無い「横歩取り」等の戦型を選ぶのは納得いきませでした。そこで「まふ互角局面 ver4（やねうら王形式）」をベースに、①指し手の採用率の変更および②指し手の追加を行い、自分らしい特定の定跡/戦型になるべく教育を行っております。
また、Deep Learning を学習するために購入した GPU 搭載の NOTE PC を活かすべく、TMOQ は CUDA というコンピュータ言語で書かれております。
昨年の SDT5 で TMOQ は CUDA を使ったチーム内 1 位を達成しました。今回の WCSC28 でも CUDA 採用ソフト内 1 位を目指しております。(参加チーム一覧を見る限り、1 位確定！)
定跡を加工して楽しみ、並列化プログラミングにチャレンジして楽しみ、テスト対局を見て楽しみ、5 月に向けて楽しく準備を進めていきます！

【作者】
宮崎の生まれ、幼少の頃より将棋を嗜むも、指し将棋は 5 級程度（初段の免状は保有）。IT 企業勤務ですが、コンサルや管理系の仕事で海外を飛び回ることが多く、ここ 10 年は仕事で開発を行っておりません。大好きな開発を行うため、将棋のソフトに触り始めました。
毎回大会の前後は海外を飛び回っていることが多く、今回も大会の翌日からインドへ飛びます。この季節はマンゴーが美味しいんですよ。

悲劇的 with Zero アピール文書正式版

1. ソフトの紹介

【ソフト名】

「ひげきてき　うぃず　ぜろ」と読みます。名前はマーラーの交響曲第 6 番イ短調『悲劇的』にちなんでいます。作者の人生観が投影されています。

【2 つのバージョン】

現在、①利きを中心の特徴として、母体は Bonanza 6.0 で学習部に最新のライブラリ（技巧・やねうら王・Apery）という構成のソフトと、②2 年前の選手権バージョンに強化学習風の学習で追加学習させたソフトの 2 つのバージョンを作成しており、現状ではどちらのバージョンで出場するかを決めていません。なので、①②について併記します。

【使用ライブラリ】

①②共に母体として Bonanza 6.0 を使用しています。上述したとおり、①のバージョンでは学習部に技巧・やねうら王・Apery から移植したコードを用います。なお、現時点で技巧の移植を試しましたが、本番までに間に合わない感じです。

【評価関数】

①は KKP + PP + 玉の位置と利きです。利きを使っているのがポイントで、利きは飛・角・香とそれ以外を分けて保持しています。②は 3 駒そのままです。

【探索】

①②両方とも Bonanza 6.0 の探索に Stockfish 風の Razoring と ProbCut を加えています。若干ですが棋力がアップしているようです。ただし、スレッド分割方式は従来のままですので、当世風の探索からするとかなり弱いです。そろそろ最新のライブラリを使っても良い気がしますが、フルスクラッチで開発されている方々のことを考えると、そこまでマキャベリストにはなれないです。

【盤面構造】

①②両方とも Bonanza 6.0 の Rotated Bitboards を踏襲しています。1 方向版の Pext Bitboards は既に実験が終了していて、組み込める状態にはなっていますが、再テストの工数を割くことができないので、そのままになっています。

【その他前々回からの変更点】

特にありません。時間はかけているのですが、作者のスキル不足が大きく響いています。

悲劇的 with Zero アピール文書正式版

2. 今後の開発について

今回の選手権とは直接的には関係ありませんが、簡単に書いてみます。

【評価関数】

前々回出場時から利きの有効利用を考えています。それでいて、なるべく主流となっている学習方法を使わないで開発したいのですが、そうはなっていません。今回も本当は利きを Bonanza Method で学習させたかったのですが（作者の理解が進んでいるので)、本番までに時間が足らず、オンライン学習を試そうとしています（結局妥協…)。

【探索】

結局オリジナリティのある探索は何も思い付いていません。「そろそろ完全 Stockfish 化するしかないか…」とも思っていますが、「既に有効だと分かっていることをやってもなぁ…」という気持ちが勝ってしまい、今のところやっていません。じゃあ、何か思い付け(笑)！

【詰めエンジン→ゼロシステム】

詰めエンジンは、このところ情熱を失っていて、何もやっていません。評価関数や探索に比べると棋力への影響が少ないせいで、やる気スイッチが入らないのだと思っています。

【定跡】

プロの先生方の棋書を購入し、現在 354 パターンほど定跡ファイルを作りました。しかしながら、時間が足りない！　最低 1,000 ファイルくらいはできないと実戦で使えないので、どこかで時間を捻出してラッシュするしかありません。学習用の棋譜ファイルから作る方法だと、私の棋力が投影されないので、地道に今の方法でがんばりたいですが…。

3. 今後どういうソフトにしたいか？

出場する度に決意を新たにしていますが、リック・フレアーのように箒が相手でも、あるいは相手がいなかったとしても将棋が指せるソフトの開発を将来的には目指しています。

日本人ではヒロ斎藤さんを目標にしています。

4. 謝辞

以下の方々にあつく御礼申し上げます。

悲劇的 with Zero アピール文書正式版

・使用ライブラリ開発者の皆様

・私の母

【追伸】
　確約はできませんが、本番の直前と直後にこの文書をアップデートする予定です。

【2018/04/08 追記】
　学習部に最新のライブラリを適用するのは難易度が高くて無理でした。なので、プラン②で行きます（Bonanza 6.0 のみ使用）。Bonanza 6.0 を採用した理由についても追記します。作者がその全体像の 80 パーセントは理解しているからです。

-以上-

岩﨑　高宗

悲劇的 with Zero アピール文書正式版

バージョン履歴

バージョン	日付	改版内容
1.0	2018/03/14	新規作成
1.1	2018/04/08	一部を追記

動いていることが奇跡で、特段アピールすることなんてないのですが、
アピール箇所がないとアピール文書リジェクトとなることを（確か）2015年に実証しておりまして、
運営の方に迷惑をかけるので、とりあえず前回のアピール文書から技術的なところをコピペしておきます。

・探索はαβ中心で、LMRとかFutilityとかの至って普通の技術を使ってます
→なんか評価関数をいじったらLMR効かなくなったのでどうするか（20180331）
・オンライン学習を勉強してみたかったので、平均化パーセプトロンを試しています
→今更ながらこれ本当に平均化パーセプトロンなのか・・・？って気もしてきた（20180331）
以上です。

アピールについては以上なので、思い出話と参考URLを貼っておきますね。
といいつつ、過去のアピール文書をコピーしたところも多いですが。。。

■始めての参加
始めての参加は第17回の時で、選手としてではなく、アルバイトとして小谷研から徴集されました。
場所はかずさアークだったのですが、近くのホテルに泊まるとバイト代から足が出るという訳の分からない状態だったので、
君津のネカフェに3泊4日し、毎朝となりのマックでご飯を食べてました。
もう二度とネカフェで3泊4日はしたくありません。

第17回大会
<http://www2.computer-shogi.org/wcsc17/>
かずさアーク
<http://www.kap.co.jp/>
自遊空間君津店
<https://jiqoo.jp/shop/9931865/>

マクドナルド君津店

<https://map.mcdonalds.co.jp/map/12031>

■始めての出場と、（恐らく大会史上初の）プログラム名リジェクト

実は当初プログラム名は、「(´・ω・`)」にしていたのですが、

運営の方から「読めません」と言われリジェクトされ、今のかわいいかわいい名前になりました。

デビュー戦はなかなかに熱かったです、白砂将棋さんと対局させていただきました。

あの負け方は二度と忘れることは無いでしょう、悔しかったw

実は二次予選に行って20位でしたすげえ。

第21回大会

<http://www2.computer-shogi.org/wcsc21/>

■始めての賞罰

・独創賞受賞（解説記事の生成）

・（恐らく大会史上初の）アピール文書リジェクト

2012年に独創賞いただきました、ﾔｯﾀｰ

2017年にどう考えてもポナがDLぶん回していて独創賞取れる状況にも関わらず、

「優勝しなかったから独創賞対象なし」という恐ろしい裁定が下されていたので、早いうちに取っておいて本当に良かったと思いました。

第22回大会

<http://www2.computer-shogi.org/wcsc22/>

■大会を見に行かないでどこか遊びに行こう

再びかずさアークでの開催となった時期から、なぜかいつも二日目が暇だったので、

将棋を見るのではなくて、どうせだからどこかに遊びに行くとかそんなことやってました。
2016年には罰ゲームでうまるちゃんやりました。
その際は、（確か菅井先生だったかな）プロ棋士の方が「なんか変な人いるんですけど大丈夫なんですか？」と運営の方に相談されていたそうです、すみません、ぼくは大丈夫な人です。
また、千田先生からは「ちょっと身長が高すぎるかもしれませんね」と直接アドバイスをしていただきました、ありがとうございました。
ところでコスプレは罰ゲームだと思っていたのに、たぬきさんとか自発的にコスプレされているので、コスプレは罰ゲームじゃなかったんだなあと思いました。
もうやりたくありませんが、とりあえず一式は取ってありますので、うまるちゃんになりたい人がいればお貸しすることは可能です。

喜楽飯店(担々麺)
<https://tabelog.com/chiba/A1206/A120603/12012396/>
東京ドイツ村
<http://t-doitsumura.co.jp/>
マザー牧場
<http://www.motherfarm.co.jp/>
東京湾観音
<http://www.t-kannon.jp>
食事処やまよ
<http://www.yamayo7240.com/>
うまるが家でかぶってるアレ [干物妹！うまるちゃん]
<http://cospa.co.jp/detail/id/00000065274>

■目標
まったりゆうちゃんを倒して師匠超えしたいです。
本当はメカ女も倒して先輩超えもしたかったのですが・・・。
そろそろゆうちゃんと直対したいのですが、全然当たりません・・・。

それでは今年も、参加者の皆さん、CSA運営の皆さん、よろしくお願いします。

がんばるぞー。

broaden アピール文章

中屋敷 太一

目次

- broadenについて
- broadenの強さ
- なぜDeep Learningを使うのか
- 学習に成功したら嬉しいこと
- 付録 ネットワーク図

broadenについて

- Alpha Zero風の実装
  - Policy NetworkとValue Networkを学習し、MCTSで探索
  - 学習にはCaffe[1]を使用
- 目標：なるべく小さいネットワークで強くする
  - 現在(3/27) Convolution 3層、全結合 1 + 2 + 2層
  - このアピール文章の末尾に、ネットワーク図を添付
- 学習にはfloodgateの棋譜3年分(2015, 2016, 2017)を使用
  - 指し手と勝ち負けの結果のみ使用
  - 評価値、読み筋は不使用

[1] Caffe (https://github.com/BVLC/caffe)

broadenの強さ

- floodgateに投入しているbroaden_nnはCPUのみ
  - Ryzen 1700
  - レーティング1248（3/27）
- グラフィックボードの使用の有無により大きく異なる
- グラフィックボードは現在注文中

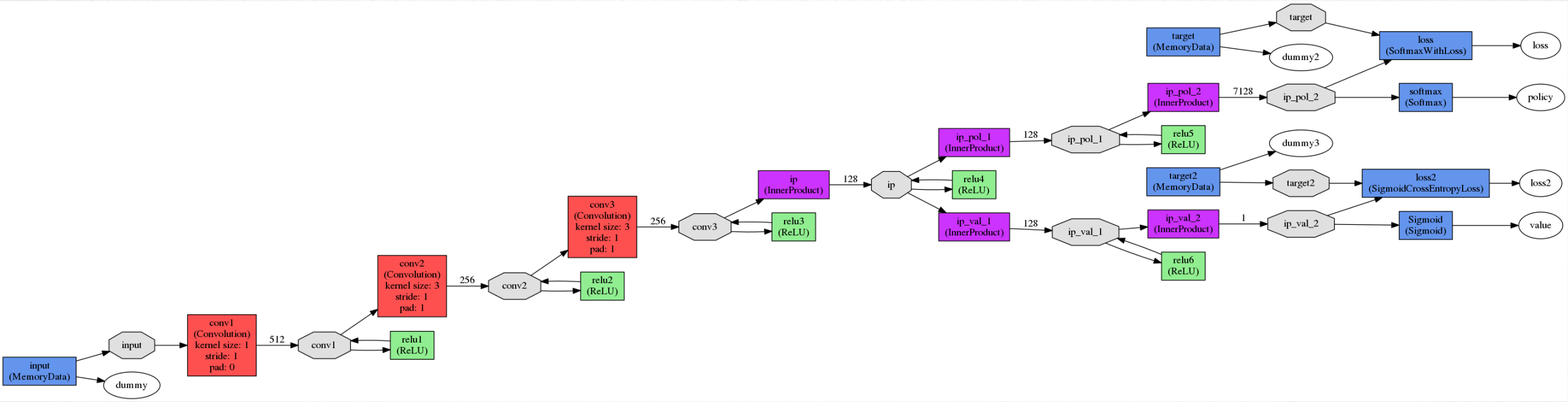
なぜDeep Learningを使うのか

- bonanzaなどの学習では評価項目を教える必要がある
  - 駒の価値
  - 手番の価値
  - 三駒関係 など
- 人間が考えた評価項目では、それが適切か判断が難しい
- Deep Learningを用いると局面だけでも学習可能
  - 評価項目を教える必要なし

学習に成功したら嬉しいこと

- もし、ネットワークが評価している項目が分かれば…
  - 人間が将棋を指す際にも、何に注目すれば良いかわかる
- では、ネットワークが評価している項目は分かるのか
  - わかるのか、わからないのか、がまだ不明
  - でも、小さいネットワークのほうが分かりやすいはず
  - そして、そもそも学習に成功しないと意味がない
    - これが、小さいネットワークで強くしたい動機

付録 ネットワーク図


input
(MemoryData)
dummy
input
conv1
(Convolution)
kernel size: 1
stride: 1
pad: 0
512
conv1
relu1
(ReLU)
conv2
(Convolution)
kernel size: 3
stride: 1
pad: 1
256
conv2
relu2
(ReLU)
conv3
(Convolution)
kernel size: 3
stride: 1
pad: 1
256
conv3
relu3
(ReLU)
ip
(InnerProduct)
128
ip
relu4
(ReLU)
ip_pol_1
(InnerProduct)
128
ip_pol_1
relu5
(ReLU)
ip_pol_2
(InnerProduct)
7128
ip_pol_2
softmax
(Softmax)
policy
target
(MemoryData)
target
dummy2
loss
(SoftmaxWithLoss)
loss
ip_val_1
(InnerProduct)
128
ip_val_1
relu6
(ReLU)
ip_val_2
(InnerProduct)
1
ip_val_2
Sigmoid
(Sigmoid)
value
target2
(MemoryData)
dummy3
target2
loss2
(SigmoidCrossEntropyLoss)
loss2

GA将!!!!!!!!!
アピール文書

2018年03月26日 森岡 祐一

はじめに

- GA将（がしょう）は、私が趣味で開発しているコンピュータ将棋プログラムです。
- コンセプトは「エキスパート（≒プロ棋士、強豪プログラム）の棋譜を用いずに、どこまで強くなれるかを追求する」です。
- 現在は、GA将同士の自己対局の経験から、強化学習を用いて評価関数パラメータの調整を行っています。
- “GA将!!!!!!!!!”の‘!’の数はメジャーバージョンです。現在はVer.9ですので‘!’は9個です。

経歴

- 公式戦デビューは2006年の世界コンピュータ将棋選手権です。
- 分かりやすく言うと、Bonanzaと同期です。実力は比べるまでもありませんが。
- 世界コンピュータ将棋選手権では、一次予選敗退と二次予選進出を行ったり来たりしている状況です。
- 本将棋と同一の対局・学習ルーチンで5五将棋にも対応しているので、そちらの大会にも出ています。GPW杯5五将棋大会での優勝経験もあります。（ちょっと自慢。）
- 最近流行りの強化学習ですが、GA将では2007年から既に取り入れていています。以前の大会はアピール文書が無いので正確には分かりませんが、強化学習を用いるプログラムの中では、GA将が最古参だと思います。

対局ルーチンの構成

- αβ探索＋線形の評価関数（駒割＋三駒関係＋α）の、比較的オーソドックスな構成です。と言うか、正直ここには目新しい部分はありません。
- マルチスレッド化ですが、並列αβ探索ではなく多数決合議を用いています。
  - シングルスレッド探索のクライアントを、1台のPCで8クライアント並列に動作させています。
  - 合議クライアントの作り方ですが、文殊やBonanza Felizとは異なり、Dropoutを用いて一部の評価関数パラメータをランダムに0クリアする方式です。
- 後は、ABC探索ルーチンベースの詰将棋ルーチンがあります。

学習ルーチンの概要

- GA（遺伝的アルゴリズム）→TDLeaf(λ)→PGLeafと、色々な学習ルーチンを試してきました。
- ちなみに、“GA将”の名称の由来は、初期バージョンでGAを用いていた事です。
- 2017年のAlphaGo Zero登場に影響を受け、複数の損失項を組み合わせてみようと思い、現在は“PGLeaf Zwei”と呼んでいる学習ルーチンを使用しています。
- ざっくり概要を書くと、「損失関数＝PGLeaf項＋TDLeaf(λ)項＋交差エントロピー項＋L2正則化項」です。
  - 各項は、それぞれ「勝率の最大化」「評価関数の精度向上」「浅い探索と深い探索の結果の一致率の向上」「オーバーフィッティングの抑制」を目的としたものです。
- 学習は、評価関数パラメータをランダムに初期化し、その後は自己対局とパラメータ修正（損失関数の最小化）を繰り返す流れになります。

目標

- 正直な所、ライブラリ勢全盛のこの状況では一次予選突破は難しいと思います。
- という訳で、「独創的で面白い将棋を観てもらう事」を目標にして開発中です。
- ただ、出来れば一次予選で勝ち越しはしたいなぁ、という欲もあります。

最後に

- ブログ・Twitter・Webサイトも有りますので、興味を持って頂ければご覧下さい。
  - ブログ：http://d.hatena.ne.jp/Gasyou/
  - Twitter：https://twitter.com/MoriokaYuichi（鍵付きアカウントです）
  - Webサイト：http://gasyou.is-mine.net/

第28回世界コンピュータ将棋選手権

ツツカナ アピール文書

ツツカナの一番の特徴は指し手を読む深さを機械学習によって決定していることです。

詳細につきましては第21回選手権のアピール文書をご参照下さい。

*アピール文書へのリンク

http://www.computer-shogi.org/wcsc21/appeal/tsutsukana/WCSC21_tsutsukana_20110327.pdf

また細かな点としては以下の通りです。

・自己対戦の結果を利用した評価関数の学習

・二駒の位置関係や玉の安全度、8つの進行度を駆使したコンパクトで表現力の高い評価関数

・Kindergarten Bitboardsを使用したシンプルで移植性に優れた利きデータ

・機械学習+ハンドチューニング(探索中に得られる動的な情報)によって構成されたreduction

2018/03/31

アピール文

（参加の動機）

Linuxとはなんぞえという興味、そしてJavaの勉強のために参加させていただきます。元来、目標が明確でないとやる気が出ない性格なもので、、、。

（開発の状況）

昨年の12月頃から作り始めました。現時点（３月始め）では、連続して次の一手問題を解ける程度までになっています。今は、各種パラメータの調整、ネットワーク通信部分の調整に精を出しています。

（今後の予定）

並列、Ponder、評価関数の機械学習は、残念ながら間に合いません。したがって、評価関数は第２１回大会参加時にボナンザメソッドで作成したものを流用します。近々にfloodgateに利用させていただき、バグチェック、ストレステストを行いたいと思います。

（ソフトの特徴）

すべてオリジナルです。プログラムは、基本に忠実で、とてもシンプルです。

（最後に）

一次予選敗退は確実ですが、何とか、複数回の勝利をあげたいと思います。

W@ndre アピール文

2018/03/27

【由来】

W@ndreはwanderでwonderな将棋を指してほしいという意図で命名したプログラムです。

【目標】

長期的：角頭歩戦法を指しこなしてもらう。

今回：ディープラーニングのプログラムと既存プログラムを組み合わせて戦う。

【使用ライブラリ】

dlshogi

やねうら王

【定跡部】

角頭歩を指します。角頭歩ができない（▲２６歩△３４歩▲２５歩のオープニング等）ときは阪田流向飛車を指します。

昨年の第５回将棋電王トーナメント（以下SDT5）ではプロの棋譜やFloodgateの棋譜を利用して定跡を作成していましたが、

特定の変化に対して角頭歩側にマイナスとなる変化も多々含まれていました。

今回は自己対戦や手調整で特定局面での探索結果を反映させることで、簡単に不利な局面に持っていかれないよう工夫を試みています。

【評価関数】

やねうら王で利用するKPPT型評価関数について３種類の評価関数を用意したので、

それぞれを単体かブレンド(*a)して使い分ける予定です。

１．

昨年の選手権以降、教師局面の作成時に定跡を用いると、定跡を指しこなせるように見える評価関数が出来ることが確認されています。[1], [2], (*b)

本選手権では、上記【定跡部】で作成した定跡を用い、ゼロベクトルから強化学習を行い評価関数を作成しました。

作成された評価関数を元に定跡を改良し、その定跡で評価関数の追加学習を行うというサイクルを回しました。

２．

SDT5にてHoneyWaffleが行っていた学習法[3]を参考に、特定の形に強制的に減点・加点することで角頭歩局面を評価する評価関数を作成しました。

３．

やねうら王ライブラリより配布されていた教師局面データ(*c)で、評価関数バイナリ[181_0020G]を再学習させました。

１，２は角頭歩戦法を理解できるかの試みとして作成した評価関数、

３はやねうら王ライブラリで配布されている教師局面と評価バイナリを学習の練習も兼ねて作成した評価関数です。

また、dlshogiに関して、３の教師局面を学習させたモデルを利用する予定です（執筆時点）

【探索部】

序盤は２駒や３駒より表現力の高いディープラーニング系の評価関数と探索手法が優れている気がしたので(*d, *e, *f, *g, *h)、序盤に関してはdlshogiを利用し、

中終盤以降はやねうら王探索部に頑張ってもらいます。

現時点ではライブラリの探索部に変更を加える予定はありませんが、

やねうら王ライブラリ利用者の参加状況次第では、探索延長部のパラメタに手を加えるかもしれません。[4][5]

【謝辞】

本プログラムを作成するにあたり、やねうら王ライブラリとdlshogiライブラリの作者様に深く感謝いたします。

また、金沢将棋、白砂将棋、Labyrinthus、HoneyWaffleの存在なくしてはW@ndreはありませんでした、アイデアをくださった皆様に御礼申し上げます。

【参考文献】

[1] 棋風を覚える将棋ソフトが完成してた件

http://yaneuraou.yaneu.com/2017/06/26/%E6%A3%8B%E9%A2%A8%E3%82%92%E8%A6%9A%E3%81%88%E3%82%8B%E5%B0%86%E6%A3%8B%E3%82%BD%E3%83%95%E3%83%88%E3%81%8C%E5%AE%8C%E6%88%90%E3%81%97%E3%81%A6%E3%81%9F%E4%BB%B6/

[2] 素人による評価関数自作記---ゼロから自分好みの評価関数を育成

http://www.uuunuuun.com/single-post/2017/06/22/%E7%B4%A0%E4%BA%BA%E3%81%AB%E3%82%88%E3%82%8B%E8%A9%95%E4%BE%A1%E9%96%A2%E6%95%B0%E8%87%AA%E4%BD%9C%E8%A8%98---%E3%82%BC%E3%83%AD%E3%81%8B%E3%82%89%E8%87%AA%E5%88%86%E5%A5%BD%E3%81%BF%E3%81%AE%E8%A9%95%E4%BE%A1%E9%96%A2%E6%95%B0%E3%82%92%E8%82%B2%E6%88%90

[3] HoneyWaffle 第5回将棋電王トーナメント版

https://github.com/32hiko/HoneyWaffleSDT5

[4]魔女より強くすると本当は強くならない？！

http://yaneuraou.yaneu.com/2016/06/14/%E9%AD%94%E5%A5%B3%E3%82%88%E3%82%8A%E5%BC%B7%E3%81%8F%E3%81%99%E3%82%8B%E3%81%A8%E6%9C%AC%E5%BD%93%E3%81%AF%E5%BC%B7%E3%81%8F%E3%81%AA%E3%82%89%E3%81%AA%E3%81%84%EF%BC%9F%EF%BC%81/

[5]Stockfish DD - search 探索部

http://yaneuraou.yaneu.com/2015/02/24/stockfish-dd-search-%E6%8E%A2%E7%B4%A2%E9%83%A8/

(*a) 評価関数のキメラ化コマンド公開しました
http://yaneuraou.yaneu.com/2017/06/28/%E8%A9%95%E4%BE%A1%E9%96%A2%E6%95%B0%E3%81%AE%E3%82%AD%E3%83%A1%E3%83%A9%E5%8C%96%E3%82%B3%E3%83%9E%E3%83%B3%E3%83%89%E5%85%AC%E9%96%8B%E3%81%97%E3%81%BE%E3%81%97%E3%81%9F/

(*b) 定跡の生成に使った評価関数を用いるべき？
http://yaneuraou.yaneu.com/2014/12/27/%E5%AE%9A%E8%B7%A1%E3%81%AE%E7%94%9F%E6%88%90%E3%81%AB%E4%BD%BF%E3%81%A3%E3%81%9F%E8%A9%95%E4%BE%A1%E9%96%A2%E6%95%B0%E3%82%92%E7%94%A8%E3%81%84%E3%82%8B%E3%81%B9%E3%81%8D%EF%BC%9F/

(*c)-1 将棋ソフトの機械学習の成否を判定するための資料
http://yaneuraou.yaneu.com/2017/05/26/%E5%B0%86%E6%A3%8B%E3%82%BD%E3%83%95%E3%83%88%E3%81%AE%E6%A9%9F%E6%A2%B0%E5%AD%A6%E7%BF%92%E3%81%AE%E6%88%90%E5%90%A6%E3%82%92%E5%88%A4%E5%AE%9A%E3%81%99%E3%82%8B%E3%81%9F%E3%82%81%E3%81%AE%E8%B3%87/

(*c)-2 depth10で作った110億局面の教師データ、期間限定で公開します
http://yaneuraou.yaneu.com/2017/11/21/depth10%E3%81%A7%E4%BD%9C%E3%81%A3%E3%81%9F110%E5%84%84%E5%B1%80%E9%9D%A2%E3%81%AE%E6%95%99%E5%B8%AB%E3%83%87%E3%83%BC%E3%82%BF%E3%80%81%E6%9C%9F%E9%96%93%E9%99%90%E5%AE%9A%E3%81%A7%E5%85%AC%E9%96%8B/

(*c)-3 クリスマスプレゼントの配布はじめました
http://yaneuraou.yaneu.com/2017/12/25/%E3%82%AF%E3%83%AA%E3%82%B9%E3%83%9E%E3%82%B9%E3%83%97%E3%83%AC%E3%82%BC%E3%83%B3%E3%83%88%E3%81%AE%E9%85%8D%E5%B8%83%E3%81%AF%E3%81%98%E3%82%81%E3%81%BE%E3%81%97%E3%81%9F/

(*c)-4 月刊教師局面 2018年1月号
http://yaneuraou.yaneu.com/2018/01/23/%E6%9C%88%E5%88%8A%E6%95%99%E5%B8%AB%E5%B1%80%E9%9D%A2-2018%E5%B9%B41%E6%9C%88%E5%8F%B7/

(*d) https://twitter.com/messiah_ai/status/920236374527053824

(*e) https://twitter.com/ymg_aq/status/940899696389767168

(*f) https://twitter.com/cute_na_piglets/status/944583388874153984

(*g) https://twitter.com/hillbig/status/938910529396936704

(*h) アピール文から読み解くWCSC27の見どころ（技巧編）
http://qhapaq.hatenablog.com/entry/2017/04/18/210624

(*i) 開発メモ : 角交換に特化した高速化
http://d.hatena.ne.jp/LS3600/20100117

*前回(SDT5)のアピール文書
http://denou.jp/tournament2017/img/pr/wandre.pdf

Crazy Shogi is an implementation of the AlphaZero algorithm, as published by DeepMind. It is entirely original code developed from scratch, in C++. The neural network code uses the CUDNN library to run on Nvidia GPUs.

第 28 回世界コンピュータ将棋選手権
「S.S.E.」アピール文書

2018 年 3 月 30 日

開発者：和田悠介、斉藤優輝、吉野拓真

S.S.E.の特徴

- フルスクラッチで開発

  Stockfish 8 をベースに 0 から開発を行いました．盤面を表現するビットボードは縦型ビットボードを使用しています．合法手生成部はオリジナルのものを実装しました． この間まで，開き王手の生成にバグがあったので，他にもないか少し不安です．

- 探索

  基本的には Stockfish 8 ベースになっています．定跡は使用しないつもりです．今後、Stockfish 9 の変更を一部取り入れ，探索パラメータの調整や枝刈り手法の変更を行う予定です．

- 評価関数

  評価関数は，コンピュータ将棋選手権使用可能ライブラリである elmo の評価関数をそのまま使用する予定です．選定理由は，ライブラリの中で最も強い評価関数であるためです．ディープラーニングを利用した評価関数の開発も進めており，うまくいけば今後変更する可能性があります．

謝辞

縦型ビットボードの実装，千日手の判定を行う際に，一部 Apery の実装を参考にしました．Apery 開発者の平岡拓也さんに感謝を申し上げます．また，コンピュータ将棋選手権使用可能ライブラリである elmo の評価関数を使用しました．elmo 開発者の瀧澤誠さんに感謝を申し上げます．

第 28 回世界コンピュータ将棋選手権参加ソフト

「ねね将棋」アピール文書

ねね将棋(NEural NEtwork Shogi)は、深層学習(Deep Learning)を用いた評価関数により思考する将棋ソフトです。従来の 3 駒関係+α β 探索に代わるアーキテクチャで強くすることを目指しています。

使用ライブラリ

やねうら王 [1] (ソースコードおよび教師局面) : ユーザ定義エンジンの追加がしやすいため

~~python-shogi [2] (ソースコード) : 通信に用いる python 言語と親和性が良いため~~ （クラウドとの通信切断時に指し手生成を行わせることを予定していましたが、状態管理のミス等で逆に信頼性が下がりそうだったので実装しませんでした）

探索部

AlphaZero [3]等で採用されている MCTS (Monte-Carlo Tree Search)を実装します。USI 通信・合法手の列挙部分まではやねうら王ライブラリを利用しています。

2017 年 11 月の第 5 回将棋電王トーナメント(SDT5)では python 言語でゲーム木を実装していたのですが、速度上のボトルネックとなっていたため今回は C++言語で実装します。探索はシングルスレッドで行います。

定跡として、floodgate の 2017 年の棋譜にて 100 回以上登場した局面それぞれを 100 万回ずつ読ませた状態の置換表を初期値としてロードすることで代用します。探索の初期段階は GPU の並列性を活かしにくいのですが、この方式であれば定跡から外れた局面でも浅い読みの結果が残っているため、ちょっとだけ有利なのではないかと期待しています。

通常探索とは別のスレッドでルート局面からの詰将棋探索を行います。この結果が詰みの場合、通常探索の結果を上書きして詰ませに行きます。実装はやねうら王に搭載されているものを改造して用いています。通常探索の末端からも浅い詰み探索を行うことを試みましたが、NPS が下がってしまうためできませんでした。探索のマルチスレッド化が必要だと思われます。

評価関数

局面の勝率および各指し手の事前確率を出力する Deep Neural Network を深層学習により学習します。

やねうら王プロジェクトで提供されている教師データによる教師あり学習を行います。

速度上の問題で自己対戦による強化学習には至っていません。

評価関数の実行には GPU を用います。SDT5 では 5,000NPS 程度(NVIDIA GTX 1080Ti)で、CPU からのデータ供給側のボトルネックにより GPU の性能を使いきれませんでした。今回はクラウド上のマルチ GPU マシンを有効に活用できるよう、実装を刷新します。

モデル構造

14 層の Convolutional Neural Network です。Residual 構造があり、中間層は 192ch あります。

盤面入力は Ponanza チームの資料[4]を参考にした 86 チャンネル、指し手出力は AlphaZero の資料を参

考にした 139 チャンネルとなっています。
やねうら王の棋譜[5]を用いて教師あり学習しています。局面ごとの指し手および勝敗を回帰します。評価値は使っていません。最適化手法は Momentum SGD です。
学習した結果、探索なしに棋譜の指し手を正解できる確率が 40%程度となりました。この状態で floodgate レート 1700 程度でした。
モデル構造をこれ以上大きくしても正解率が上がりませんでした。ただ、不正解の 60%に悪手が含まれるため、現状が最善かはわかりません。

探索との組み合わせ

探索中に新たに到達した局面すべてについて、GPU 上で評価を行います。1 局面ずつ評価すると極端に性能が悪いので、512 局面のミニバッチにして非同期的に評価します。
Deep Learning フレームワークとして Python で書かれた Chainer を用いており、マルチスレッドが原則できません。マルチ GPU を活用するため、Chainer を用いた評価プロセスを複数立てて、C++で書かれたメインのエンジンプロセスとプロセス間キュー（自作）で接続しました。
GPU は AWS 上の NVIDIA Tesla V100 を用います。モデルを通常の float32（単精度浮動小数点数）から float16（半精度浮動小数点数）にし、Tensor Core を有効にすることで、ランダム入力に対する評価のみのベンチマーク上は 9,000NPS 程度得られました。
GPU を 8 台並列動作させ、探索部と組み合わせた場合標準的に 30,000NPS 程度となります。探索結果の重複が少なくなる、探索時間が長い場合で、最大 60,000NPS 得られるようです。
MCTS は前向き枝刈りと同様の性質があり、NPS が向上しても見えない変化があるようです。対局相手に意外な手を指されて置換表にない局面に飛び込み、一気に評価値が悪くなるという事象がみられます。現状、うまい対策ができていません。

おわりに

SDT5 からは開発言語の変更が主となりますが、大幅な探索速度向上が狙えます。
モデルを大きくしたこともあり、SDT5 から floodgate レートが 1000 以上上がっています。
クラウドに課金して一次予選突破が目標です。

[1] 磯崎元洋 https://github.com/yaneurao/YaneuraOu
[2] 末永匡 https://github.com/gunyarakun/python-shogi
[3] D. Silver et al., Mastering Chess and Shogi by Self-Play with a General Reinforcement Learning Algorithm. http://arxiv.org/abs/1712.01815 (2017)
[4] HEROZ-JAPAN 「Ponanza における強化学習とディープラーニングの応用」 https://www.slideshare.net/HEROZ-JAPAN/ponanza-83900718 (p.19)
[5] やねうら王公式サイト「depth10 で作った 110 億局面の教師データ、期間限定で公開します」 http://yaneuraou.yaneu.com/2017/11/21/depth10%E3%81%A7%E4%BD%9C%E3%81%A3%E3%81%9F110%E5%84%84%E5%B1%80%E9%9D%A2%E3%81%AE%E6%95%99%E5%B8%AB%E3%83%87%E3%83%BC%E3%82%BF%E3%80%81%E6%9C%9F%E9%96%93%E9%99%90%E5%AE%9A%E3%81%A7%E5%85%AC%E9%96%8B/

2018 年 3 月 7 日（4 月 30 日修正） 日高雅俊 https://github.com/select766

WCSC28 アピール文

2018/3/26

ソフト名「ArgoCorse_IcSyo（あるごこるせいっしょ）」

市村豊（いちむらゆたか）

Twitter：@argonworks

ブログ：http://argon1.blog55.fc2.com/

使用ライブラリは「やねうら王」です。選定理由はそもそもコンピュータ将棋を始めるにあたってやねうら王nano/miniのコードを読んで改造するところから始めてそのまま使い続けている感じです。また今回は評価関数のキメラ化というのをやってみたくなったので、混ぜる元としてelmo,tanuki-,Apery,Qhapaqを選定させていただきました。また、1月のときになんか技巧もやねうら王系だと思いこんでいたため書いてしまいましたが違っていたので取り消します。すみません。

独自性として、SDT5のときの文と重複しますが、もともとコンピュータ囲碁から入ったことからモンテカルロ法の将棋への応用の文献をインターネットで読んでいて、それが「現時点から10手先の局面を評価関数で評価して元局面よりも評価値が高い場合を勝ちとする」という方法でモンテカルロ将棋をやっていたものがあったので、メインの探索には使えないとしても定跡の選択手法としてはやって見る価値があるのではないかと思って、定跡に候補が複数ある場合に二手先読みをしてその時の評価値が一番高い手の一手目を採用する、という変更を行いました。それでSDT5のときは「やねうら王」に付属していた定跡を使ったのですが、今回はfloodgateの2009-2017までの棋譜からとりあえず31手目までをそのまま定跡ファイルに変換したものを定跡として作ってみたのでそれでやりたいと思っています（棋譜を全部変換すると定跡のファイルサイズが大きくなって動くかどうか不安なので31手までにした、そのくらいだと650MBくらいですむので問題なく動作するかなと）。それで、「やねうら王」に付属の定跡では元よりも悪くなる手は入っていないとあったのでそのまま使っていたのですが、floodgateの棋譜から単純に定跡に変換したものだとそのままだと元より悪くなる手が含まれているため、「元局面を評価関数で評価して、それよりも評価値が低くなる手は採用しない」というコードを追加しました。

ちなみに、評価関数のキメラ化を少し試してみて、現状では以下のようになりました。定跡にyaneura_book3を使って16手目まで確率で動いているので全く同じ手順にはなってはないと期待してい

ます。時間は0秒+一手あたりそれぞれ10秒ずつ加算で、名前の変え方が分からないのでsse41とsse42でコンパイルして「41」「42」の部分で判別していたのでその違いが出ているかもしれないです。ちなみに左側が41で右側が42でコンパイルしたものです。「＆」が1：1でキメラ化したことを表しています。256手で勝敗がつかない場合を引き分けに設定しています。51回ずつ自己対戦させてみました。メインのPCのi5-2500Kを1コアでことこと動かしていました。左側から見て「勝ち-引き分け-負け」です。

110億生成 VS elmo(wcsc27) 39-1-11
110億生成 VS Apery(sdt5) 19-3-29
110億生成＆Apery(sdt5) VS Apery(sdt5) 22-3-26
tanuki-(sdt5) VS Apery(sdt5) 21-4-26
elmo(sdt5) VS Apery(sdt5) 16-2-33
tanuki-(sdt5)＆Apery(sdt5) VS Apery(sdt5) 27-6-18
aperypaq VS Apery(sdt5) 23-8-20
aperypaq VS tanuki-(sdt5)＆Apery(sdt5) 18-10-23
tanuki-(sdt5)＆Apery(sdt5) VS aperypaq 16-10-25
aperypaq VS tanuki-(sdt5)＆aperypaq 26-6-19
aperypaq VS (tanuki-(sdt5)＆Apery(sdt5))＆aperypaq 25-9-17

とりあえず、「やねうら王」の最新版が公開されたのでそれにSDT5のときの旧版から移植をしたいと思っているのですが、適当にコードをコピペしてコンパイルしようとしたらエラーがたくさん出て、そのエラーの消し方がいまいちわからないので「C++は本当に難しいなー」と思って面倒くさくなって放置している現状です。なんとかやる気を出して最新版に移植したいとは思っています。評価関数にtanuki-(sdt5)＆Apery(sdt5)を双方使っての自己対戦結果が、
> やねうら王旧版（手作り定跡） VS やねうら王新版（yaneura_book3） 13-4-33
だったので最新版じゃないと勝負権が無いような気がするのでなんとかやる気を出して大会までには移植したいと思ってはいます（希望的観測）。

あとはAWSの高そうなのを借りないと勝負権が無いような気がするので、そうしたいとは思うのですが、現状AWSの使い方がよくわかりません。AI竜星戦の時に会場でDeepEsperの中の人にレクチャーし

て頂いてアカウントは作りましたが、操作は「簡単ですよ」というほど言うほど簡単じゃないと思うのですが。あと、使い方はともかく、プランがたくさんあってどれを借りたらいいのかがよく分からないのですが。なんでみんな「サーバーを借りる」なんてマニアックな事をごく普通にできるのですかね。それは義務教育ではやってないと思うのですが。

それはともかく、まあ、テンションを上げてAWSが使えるようになって大会に臨めたら嬉しいなと希望しています。それなので「よくわからないけどAWSの高そうなのを使う」みたいな申請をしたいと思いますが、テンションが上がらなかったら「やねうら王」旧版で、パソコンはレノボのX220、OSはWindows10になりますので、そうなったらすみません。できるだけテンションを上げてやりたいとは今は思ってはいます。

・SDT5参加とかの感想

2017/11にSDT5に参加出来ました。コンピュータ将棋の大会に参加してみたいと思っていたので参加できて嬉しかったです。その一ヶ月後の2017/12にコンピュータ囲碁のAI竜星戦に参加することが出来ました。参加できて良かったです。ただ、年末のコミケの原稿を書かないといけなかった時期でだいぶ大変でした。

両大会ともに参加していて「前から思っていたが、つくづくすごい人しかいない感じだなー」ということを改めて思いました。なんか、エース・オブ・エースがごろごろいる。私なんかが混じっていて本当にすみませんと思いました。ただ、この場所にいると本当に面白いということも改めて思いました。

それにしても、念願かなってコンピュータ囲碁とコンピュータ将棋の大会に参加することが出来て考えさせられたのは、「そういえばこれって何を目的にしているゲームだったんだっけ？」ということ。私は「これ、参加して対局したい」と思っていて「対局すること」を目的にしていたし今もしているのだが、そういえば囲碁とか将棋って「勝つこと」が暫定の目的だったのだなということを周りの参加者を見ていると思い出してきた。

ゲームの目的が何であるかということを思うのだが、私はゲームの目的は「楽しむこと」だと思っている。だから「勝ち負けはあくまでも暫定的な仮の目的であり、本来の目的ではない」ということを思っている。だって、勝つことが目的だとしたら優勝者以外のすべての人が目的不達成になる。それはかまわないけど、それで参加者が翌年から優勝者一名になったらもはや大会が成立しないんじゃないかとい

うことを思う。だから「負けたとしても楽しかった」と負けた人に思わせるようなゲームじゃないとゲームとして成立しないんじゃないかと言うことを思う。囲碁・将棋に限らず、スマホのゲームとかプレステのゲームみたいなのを新しく作るとしたら「プレイすることそれ自体に楽しさ・快感を感じさせる」ようなゲームを設計しないとダメなんじゃないかと思ってしまってしょうがない。2017/12月の囲碁のAI竜星戦で私はDeepEsperの中の人とチームを組んで出場したけど、DeepEsperの中の人が2017/3の囲碁のUEC杯で最下位になって心が折れたのでやめようと思う、という意味のことを言っていて私は本当にびっくりした。「どうしてこんなにすごいソフトを開発できるあなたがたかだか大会で最下位になった程度のことでやめようと思うのか」と。

SDT5のアピール文に書きましたが個人的に自動車のレースが好きなので、思い出すのが2006年にF1のレースにスーパーアグリF1が参戦したときのことです。「準備期間半年でのF1参戦は無謀」という状態で開幕戦バーレーンGPの予選でタイムを出した時に、テレビの解説者が「決勝レースのグリッドを獲得しました！」と興奮気味に言っていた気がする。後にも先にも、決勝レースの最後尾のグリッドを獲得することを「快挙」として言っていたのはこの時しか聞いたことがない（翌日の決勝レースで彼らはトップから4周遅れの最下位でチェッカーを受けて、「F1の決勝で完走したのが快挙」と日本のメディアは概ね肯定的だったのもつくづく印象的だった）。多分、私の勝負っていうのはこういうような勝負をしているんだろうなーということを今でも時々思う。SDT5の会場で「ここにいる人達は基本的に頭のいい人しかいない」というコメントが流れてきて、「そうなんだろうなー」と思った。だから、大会に参戦する権利を獲得することが一番の難題なんだと私は思っていた。インターネットのスラングなのかもしれないですが「0回戦負け」というやつです。基本的に「0回戦突破」が一番の難題であって、参戦権さえ得られたのならばあとはもう最下位でもそれは「快挙」だと私は思っている。

それにしてもSDT5のときに「オリジナリティ」がどうすればあるといえるのかという話になって、そういうことを考えると私もだんだんと自信がなくなってきたものである。「やねうら王」を改造したと言っても、元との違いがどの程度あるかということを考え出すと、戦闘機で言ったら「ミラージュ3とネシェル」、爆撃機で言ったら「B29とツポレフ4」、宇宙機で言ったら「スペースシャトルとブラン」くらいの違いしか無いんじゃないかというようなことを思って、せめてVTOLで言ったら「ハリアーとYak38」くらいには違っていないとまずいかなーと言うようなことを思わないでもない。というか、改造元が強かったせいでそれから大きく変更ができなかったせいで結果的に強い、と言うのは当人である私も余り気分が良くない。対局中に対局相手に話を聞いていると全部自分で作っているという人がたくさ

んいて、そのほうが明らかにすごいのに改造元のソフトが強かったせいで勝ってしまうというのは何か違う気がする。いっそデチューンした方がいいだろうかということも思うがそれもなんか違う気がする（前回の選手権で最下位だった「Mirage」の荒木さんは、全部自分で作って挑戦したから最下位だったわけで、それはむしろ評価すべき事のはずだ）。

SDT5に参加してその時のネタは「定跡のより良い選択手法の考案」ということだったのですが、それをするのが個人的にはだいぶ一杯一杯だったのですがやってみて「そういえば、ネタとして瑣末過ぎないか？」ということを参加してからなんか思ってきたわけです。

なのですが、「定跡を選択するときのよりよい方法を考える」というのは研究テーマとして十分に成立しうるのではないかということをSDT5に参加していて思った。それというのも、タヌキさんやカパックさんと話をしていてSDTみたいな時間が限られている勝負のときは序盤は定跡でノータイムで指して時間をセーブする、ということが有効であるということを教えていただいた。お二方に限らず結構色んな人から「時間攻め」という事を聞いて、そんな用語があるのかと勉強になりました。それなので定跡の使用がわりと重要である、ならば定跡のより良い選択方法を考える、ということも十分重要な研究テーマではないかということを思ったわけです。SDT5の文に書きましたが、2006年頃のモンテカルロ将棋の試行錯誤で、「現時点から10手先の局面を評価関数で評価して」という方法は、メインの探索のアルファ-ベータ法よりも優れてはいないようですが、それでも定跡の選択手法としては一考の価値があるのではないか、ということをSDT5に参加して改めて思いました。

それにしても、タヌキさんがSDT5の賞金300万円を全部寄付したとツイッターで見たときはつくづく衝撃的だった。そしてカパックさんがやねさんのほしいものリストのものを全部買ってプレゼントしたというのを読んだときも衝撃的でした。SDT5で彼らはタヌキのキグルミを着たり狼の被り物をかぶったりしていた。そんなような人がどうして誰よりも立派なことができるのだろうかということが改めて衝撃的でしょうがありません。私はおそらくはこういうふうな振る舞いは出来ないだろうと思わずにはいられません。そしてそんなんだから私は彼らに勝てないのだろうなということも改めて思いました。

あとは改めて思うのが「きふわらべ」の高橋さんの素晴らしさです。彼の著作物を読んでいて「数学というのはこんなにも面白いものだったのか」というのをほとんど初めて知った（学校で数学をちょっとやったというかやらされたけどあんまり面白いものだとそれまで思えなかった）。高橋さんこそまさに現代のラマヌジャンであるというようなことを個人的に思ってしまって「なんでこんなにすごい人があ

んまり評価されていない感じなのだろう」というか、「きふわらべ」の高橋さんはものすごい能力者だと思うのだけど、正当に評価されていない感じがして何なのだろうかということを思う。

これからのコンピュータ囲碁・将棋の開発について、示唆的だと思ったのがSDT5でHoneyWaffleさんと対局して話を聞いていたときのことです。HoneyWaffleさんが「現状では居飛車が優勢で振り飛車は不利とされているが、それは単に多数派が居飛車だからそうなっているだけでよく研究したら振り飛車はそんなに不利じゃない可能性がある」という事をおっしゃっていて、たしかに単純に強いソフトを作るとかよりも「振り飛車は居飛車に比べて本当に不利なのか？」を研究するということのためにコンピュータ将棋のソフトを作る、というようなことがこれからの囲碁・将棋のソフト開発では重要なテーマとしてなってくるんじゃないかという事を思います。有名なテーマかもですが、そもそも「先手・後手必勝なのか？」とか、そういうことがコンピュータ将棋のソフトを作るということからは導けるのではないかと思って、単純に強さを競うよりもそういうことをすることのほうがよほど大切なことなんじゃないかということをHoneyWaffleさんの開発コンセプトから考え込んでしまってしょうがないです。

・ソフト名「ArgoCorse_IcSyo（あるごこるせいっしょ）」について

SDT5の時にこのソフト名を誰も読めなかったのでなんか悪いような気分になる。ただ、個人的にレースが好きなのでこんな名前にしたけど、レーシングチームはスポンサー名がチーム名に入るので「オートバックス・レーシング・チーム・アグリ」とか「伊藤忠エネクス・チーム・インパル」とか、正式名称が妙に長かったりする。それなので読むほうが適当に略して読むのが慣習的なのでそういうふうなことでいいかと思うことにしたい。ちなみに、富士スピードウェイとかは、コーナーごとにスポンサーが付いていて「コカコーラ・コーナー」「アドヴァン・コーナー」「ダンロップ・コーナー」「プリウス・コーナー」「パナソニック・コーナー」とか、割と読みやすい名前で良いけど、鈴鹿サーキットのシケインは「日立オートモティブシステムズ」がスポンサーになったので、解説者はシケインに言及するたびにこのスポンサー名を言わないといけないらしく「ただいま先頭車両が日立オートモティブシステムズシケインを通過しました」というようなことを毎回言うのがいかにも言いにくそうだ。だから「鈴鹿サーキットのシケイン名よりは読みやすからいいか」と思うことにする。

・羽生竜王と井山七冠の国民栄誉賞受賞について

優れた棋士が評価されるのは大変に良いことだと思います。お二方とも偉大な棋士であることは明らかであると思うので順当ではないかと思います。

井山七冠は世界戦を重視されているという話を読んだので、先日のLG杯決勝も注目していたのですが、第二局を劇的に逆転勝ちして三局目に期待していたのですが、惜しくも敗れてしまって、今回は残念でしたがまた近いうちに世界を制してくれるものと個人的に期待していたら、その後のワールド碁チャンピオンシップでも決勝に進んで、さすがだと思っていたら今回も残念だったですが、また近いうちに世界を制してくれるものと個人的に期待しています。というか単に時間の問題だと思いますが。

また、それにしても藤井六段が強すぎますね。Twitterでよく書かれているとおりですが「アニメや漫画の主人公の設定だったらリアリティがないと言って編集に蹴られる」ようなことを現実にやってのけている様子はまさに驚愕です。

・アニメ「りゅうおうのおしごと！」の感想

1月から将棋をテーマにしたライトノベル「りゅうおうのおしごと！」のアニメが放送されました。面白い作品だと思ったので以下に感想を書きます。

> りゅうおうのおしごと！ 第一局「押しかけ弟子」

原作既読組。アニメ版はコメディ色が原作よりも強いと思った。原作はもうちょっと重っ苦しい感じ。ただ個人的には原作の重っ苦しい部分が好きなのでアニメ版がコメディによりすぎて重い部分の良さが薄くならないかなとちょっと心配している。

最初の八一が竜王戦をフルセットで竜王位を奪取するシーン。八一が勝ちまで読み切るが疲労と緊張で体が動かずに駒を持つことが出来ない、それどころか「発声しても着手は認められるが発声すら出来ない」というほどに追い詰められているところに、あいから水をもらうシーン。原作を読んでいるときはここまでヘロヘロになって死闘を戦っているということに衝撃を受けたものであるが、アニメ版では妙にあっさりとスルーしたなって感じなところがちょっと惜しい。そもそも「発声しても着手が認められる」というのが豆知識として妙に面白かった。「月下の棋士」を読んでいるときに、最終回で主人公が

名人戦の対局で「早く指せ、俺が投了できないだろう」という台詞があり、私「そうか、投了って自分の番じゃないとできないのか」というのが豆知識として妙に面白かったのだが（作中では名人が自分の手番で対局を放棄して時間切れで主人公の勝ちになる）、「りゅうおうのおしごと」の原作の最初のシーンでも発声で着手が認められるということとその発声すら出来ない、という状態がなんだか妙に印象的だった。

あとは、あいが読むシーンも一話から出てきてすごく良かったと思った。原作を読んでいてあいが読むシーンがものすごく印象的だったのでこれをアニメ版ではどういうふうに描いてくれるのか、ちゃんと描けるのかというのが気になっていたのだが、素晴らしいと思う演出で、「日高里菜グッジョブ！」と見ながら心で思った。「ハチワンダイバー」においては「読む」という行為を「深い水の中に潜る」というメタファーで表現していたが、本作は漫画ではなくて小説である。その文章での表現をこういうふうにアニメでもちゃんと表現できることが本当に素晴らしいと思った。

個人的には原作では銀子が一番好きなキャラなのだが、アニメ化発表のときに銀子の声を金元寿子がやるとあったので、そのときにやっていた「妹さえいればいい。」の可児那由多の声の印象が強すぎて、「これが銀子になるのか、え、どうやって？」と思っていたのだが、アニメ版の一話を見ると「なるほどなるほど」となかなか納得する声で良かった。

原作のプロローグは大阪城公園の満開の桜の下で八一とあいが将棋を指すシーンで、それがすごく印象的で良かったのでアニメ版ではどうするのだろうかと思っていたらOPで使っていて「ああ、なるほど」と思った。

「将棋+萌え」の作品では「駒ひびき」が個人的に好きだったのだが、あんま受けなかったらしく3巻で終わったのが残念だなと改めて思う。20巻くらい読みたかった。

星4つ。

> りゅうおうのおしごと！ 第二局「弟子のいる日常」

JS研の設立とか、あいの両親が来ることとか。1話を受けて話を展開してきた。歩夢との対局はほぼ原作通り。本作に限らず主人公が女の子とばっかり仲が良い、というなかで同性の友人がいるということが貴重であると思った。原作でももっと男性棋士を出してほしいということは思う。あとは、あいのアホ毛がなにげにいい味を出している。あのアホ毛の動きはアニメ版で初めて実現できたことで実に良い。研修会試験で全勝しろというところでの引きがうまい。「勝敗は関係ない」といっているのに勝敗にこ

だわりまくるところがなんか将棋指しっぽくて妙なリアリティが有る。
星3.5。

> りゅうおうのおしごと！　第三局「研修会試験」

　研修会の入会試験。対局シーンが凄く良かった。2話の八一と歩夢の対局は演出で描いていたが、今回のあいの対局は、「将棋を指す」という行為をガチで描いていてそれでいて見応え充分だった。原作でも将棋を指すシーンを文章で表現するということが出来ていることがすごいと思ったが、アニメでも将棋を指すシーンをここまで見応えがある映像として描けていることがすごいと思った。
　それにしても、原作を読んでいると銀子がすごく萌えるのに、アニメ版では銀子の萌えエピソードを結構まるまるカットしているっぽいのがファンとしては実に残念である。
星4つ。

> りゅうおうのおしごと！　第四局「もう一人のあい」

　天衣登場。原作ではなんのためにいるのだかよく分からないキャラだと思っていたが、アニメでは非常に映えるなと思った。声が佐倉綾音というのも非常に良い。予想以上にロリ声だったのでもうちょっとボーイッシュな感じで演じてくれても良かったんじゃないかと思わないでもないが「ニセコイ」の春ちゃんみたいな感じでこれはこれで良いと思う。
　新世界の将棋屋に行くエピソードは、原作では真剣をしていたのでアニメ版でどういうふうに描いてくれるのかと楽しみにしていたのだが、アニメ版では真剣をしないことになっていたのでちょっと残念な気はする。「タバコに千円札を巻きつけるんだ」とか、妙にディティールが細かくて新世界ならばほんとうに真剣をやっていそうなところがなんか面白かったのだが。個人的には天衣がタバコの箱を見せて真剣を誘うシーンが文章だとよくわからなかったのでビジュアルがついたらどうなるのだろうというのが見たかったのでそれが見れなかったのが残念なのだが。
　ラストの引きが非常に良い。
星4.5。

> りゅうおうのおしごと！　第五局「天衣無縫」

　あいVS天衣の対局シーンが見応えがあった。文字通りに目の色が変わるというアニメ版での描写が非常に面白い。そしてあいの瞳の色がもとに戻ることで負けを表現するところとかすごく良かった。今回の対局は演出と解説で見せていたが、これまでの将棋の対局シーンの見せ方が違っているところがすごいなと思う。天衣に負けて泣いているあいのシーンが印象的すぎて、日高里菜はうまいなーとまた思った。というかよく分からないが将棋指しと言うのはこういうふうに考えるものなのだろうかというのもよくわからなかったことなので負けることがここまで悔しいというそれが印象深かった。
星4つ。

> りゅうおうのおしごと！　第六局「オールラウンダー」

　終盤の八一が将棋を指すシーンが凄く良かった。これまではあいが将棋を指すシーンに注力していた感じがしたが、今回の八一の対局シーンが実に見ごたえがあったと思う。原作だと将棋についての豆知識が結構はいっているのだが、アニメ版ではあまり深く将棋の話をすることが出来ないのがちょっと残念な気がする。今回のように振り飛車と居飛車の違いが、ってやるのが限界なのかなと。個人的には一手損角換わりについての解説がこれを読んで初めて意味を知ったのですごく面白かったのだが。それでもエッセンスは出ていると思う。
星3.5。

> りゅうおうのおしごと！　第七局「十才のわたしへ」

　原作既読組は「はしょりすぎ」として評価が低いというのを見る前にTwitterで見ていたのだが、たしかに原作の重さからしたら「はしょりすぎ」には違いがないということは思うが、それでもワンクールのアニメでやるにはこれ以上は出来ないだろうということはわかるし、エッセンスは伝わる内容だったと思うからこれはこれでいいんじゃないかなと思った。
　序盤の八一が読むシーンが格好良すぎた。あいが読むシーンとの対比で八一が読むシーンも原作では非常に面白いのだが、今回の八一の読むシーンを見ていて「こんなにこいつは格好良く読んだっけ？」とは思った。

星4つ。

> りゅうおうのおしごと！　第八局「はじめての大会」

　マイナビオープン編。原作を読んでいると「これまでの総力戦」という感じがしていたが、アニメで見ているとどうしてもダイジェストな感じがして重さと言うか凄さがいまいち伝わりにくい感じがする。やっぱり天衣とか桂香さんのことをもっと掘り下げて描いてほしかったという気がする。そこの掘り下げが足りていないので見ていて軽く感じられてしまうのだ。
星3.5。

> りゅうおうのおしごと！　第九局「八月一日」

　祭神雷のキャラが強くてすごく良かった。戸松遥が本気を出して演じたら視聴者がドン引きするような狂気を演じられると思ったが、視聴者が引かないようにマイルドに演じてくれたのかなと思った。
　あいVS祭神の対局シーンがまた見応え充分ですごく良かった。王道ではあるが、あいの「あなたに私の竜王が取れますか！」のダブルミーニングのセリフが面白かった。
星4.5。
****

2018/3/31追記
　日付がドラフト時点の3/20だったので提出時点の3/26に変更。

　ライブラリの選定理由を追記します。
「やねうら王」コードがわかりやすかったから。
「elmo,Qhapaq,tanuki-,Apery」評価関数が強いから。
以上です。

SMS将棋のアピール文書 2018年

ライブラリの中かられさぴょんを選ばせて頂いた理由は、

ソースコードが理解しやすかったからです。

工夫した点は、探索を序盤は深く、終盤は広くしています。

また、評価は手作業で行い、細かく調整しています。

第28回コンピューター将棋選手権
アピール文書

ソフト名

girigiri

由来
ぎりぎりの斬り合いを制していきたいと思います。
完成がぎりぎりにならないように頑張ります。

開発者

阿部健信、徐子健

出場歴

- 第五回電王トーナメント（アピール文書）

前回からの変更点

- 探索部の実装がC++からRustに
- 駒の損得のみだった評価関数を、機械学習による3駒関係を用いたものに
- 静止探索の追加

目標

- 反則をしない
- 1勝する

第 28 回世界コンピュータ将棋選手権
dlshogi アピール文章

山岡忠夫
2018 年 5 月 1 日 更新

※下線部分は、第 5 回将棋電王トーナメントからの差分を示す。

1　特徴

- ディープラーニングを使用
- 指し手を予測する Policy Network
- 局面の勝率を予測する Value Network
- 入力特徴にドメイン知識を活用
- モンテカルロ木探索
- 並列化
- 自己対局による強化学習
- 既存将棋プログラムの自己対局データを使った事前学習
- REINFORCE アルゴリズムによる Policy Network の学習
- ブートストラップ法による Value Network の学習
- マルチタスク学習
- 詰み探索
- 序盤局面の事前探索（定跡化）
- マルチ GPU 対応
- CUDA、cuDNN を直接使用
- GPU ごとに異なるモデルの読み込み

2　使用ライブラリ

- elmo[1] (Commits on May 29, 2017)
- Apery(elmo の派生元) (14 Apr, 2017 commit:e3eb33ffa6aa840765d2e2efdacf1c618528a3be)

※学習データ生成、局面管理、合法手生成のために部分的に使用

2.1　ライブラリの選定理由

本プログラムは、将棋におけるディープラーニングの適用を検証することを目的としており、学習局面生成、局面管理、合法手生成については、使用可能なオープンソースがあれば使用する方針である。そのため、学習局面を圧縮形式(hcpe)で生成する機能と読み込む機能

[1] https://github.com/mk-takizawa/elmo_for_learn

を備えており、合法手生成を高速に行える elmo(派生元 Apery)を選定した。

3　各特長の具体的な詳細（独自性のアピール）

3.1　ディープラーニングを使用

DNN(Deep Neural Network)を使用して指し手を生成する。

従来の探索アルゴリズム(α β法)、評価関数(3 駒関係)は使用していない。

3.2　Policy Network

局面の遷移確率を Policy Network を使用して計算する。

Policy Network の構成には、Wide Residual Network[2]を使用した。

入力の畳み込み 1 層と、ResNet 10 ブロック(畳み込み 2 層で構成)と出力層の合計 22 の畳み込み層で構成した。フィルターサイズは 3（入力層の持ち駒の面のみ 1)、フィルター枚数は 192 とした。

3.3　Value Network

局面の勝率を Value Network を使用して計算する。

Value Network は、Policy Network と出力層以外同じ構成で、出力層に全結合層をつなげ、シグモイド関数で勝率を出力する。

3.4　入力特徴にドメイン知識を活用

Alpha Zero では、入力特徴に呼吸点のような囲碁の知識を用いずに盤面の石の配置と履歴局面のみを入力特徴とすることで、ドメイン知識なしでも人間を上回ることが示された。しかし、その代償として、入力特徴にドメイン知識を活用した AlphaGo Lee/Master に比べて倍のネットワークの層数が必要になっている。AlphaGo Zero の論文の Figure 3 によると、ネットワーク層数が同一のバージョンでは Master を上回る前にレーティングが飽和している。

強い将棋ソフトを作るという目的であれば、積極的にドメイン知識を活用した方が計算リソースを省力化できると考えられる。

そのため、本ソフトでは、入力特徴に盤面の駒の配置の他に、利き数と王手がかかっているかという情報を加えている。それらの特徴量が学習時間を短縮する上で、有効であることは実験によって確かめている。

3.5　モンテカルロ木探索

対局時の指し手生成には、Policy Network と Value Network を活用したモンテカルロ木探索を使用する。

[2] https://arxiv.org/abs/1605.07146

ノードを選択する方策に、Policy Network による遷移確率をボーナス項に使用した PUCT アルゴリズムを使用する。PUCT アルゴリズムは、AlphaGo の論文[3]に掲載された式を使用した。

また、末端ノードでの価値の評価に、Value Network で計算した勝率を使用する。

通常のモンテカルロ木探索では、末端ノードからプレイアウトを行った結果（勝敗）を報酬とするが、プレイアウトを行わず Value Network の値を使用する。

3.6 並列化

複数スレッドで並列化を行う。並列化の方式には、スレッド間でゲーム木のノード情報を共有するツリー並列化を採用する。

モンテカルロ木探索は、並列化が容易だが、Policy Network と Value Network を計算するための GPU の数以上の並列化を行うと GPU の使用で競合が発生する。

GPU は複数の計算要求をバッチで処理することが可能であるため、各スレッドからの要求をキューイングして、専用スレッドでバッチ処理することで競合を回避する。

3.7 自己対局による強化学習

Alpha Zero[4]と同様の方式で強化学習を行う。自己対局により教師局面を生成し、その教師局面を学習したモデルで、再び教師局面を生成するというサイクルを繰り返すことでモデルを成長させる。

※教師ありより強くなっていないため、大会では使用しない

3.8 既存将棋プログラムの自己対局データを使った事前学習

本プログラムを使用して、Alpha Zero と同様に、ランダムに初期化されたモデルから強化学習を行うことも可能だが、使用可能なマシンリソースが足りないため、スクラッチからの学習は行わず、既存将棋プログラムの自己対局データを教師データとして、教師あり学習でモデルの事前学習を行う。

教師データには、elmo で生成した自己対局データを使用する。

3.9 REINFORCE アルゴリズムによる Policy Network の学習

単純に自己対局の指し手を学習するのではなく、学習局面の価値と勝敗データと関連付けて学習を行う。良い局面から負けになった手は、悪手として負の報酬を与え、悪い局面から勝ちになった手は善手として正の報酬を与える。学習アルゴリズムには、AlphaGo の論文に掲載されている REINFORCE アルゴリズムを使用した。

---

[3] https://storage.googleapis.com/deepmind-media/alphago/AlphaGoNaturePaper.pdf
[4] https://arxiv.org/abs/1712.01815

3.10 ブートストラップ法による Value Network の学習

Value Network の学習の損失関数は、勝敗を教師データとした交差エントロピーと、探索結果の評価値を教師データとした交差エントロピーの和とした。

このように、本来の報酬（勝敗）とは別の推定量（探索結果の評価値）を用いてパラメータを更新する手法をブートストラップという。

経験的にブートストラップ手法は、非ブートストラップ手法より性能が良いことが知られている。

3.11 マルチタスク学習

Policy Network と Value Network のネットワーク構成が同じ層を共通化し、出力層を分けることで、同時に学習を行う。

関連する複数のタスクを同時に学習することをマルチタスク学習という。タスク間に関連がある場合、単独で学習するよりも精度が向上する。

また、対局時に Policy Network と Value Network を同時に計算できるため、高速化の効果もある。

3.12 詰み探索

モンテカルロ木探索は最善手よりも安全な手を選ぶ傾向があるため詰みのある局面で手を抜くことがある。

対策として、詰み専用の探索を行い、詰みの場合はその手を指す。

また、モンテカルロ木探索の末端ノードでも、数手の詰み探索を行い、詰みの局面を評価できるようする。Policy Network と Value Network の計算中に、CPU が待ち状態の間に詰み探索を行うため、探索速度が落ちることはない。

3.13 序盤局面の事前探索（定跡化）

出現頻度の高い序盤局面は、対局時に探索しなくても、事前に探索を行い定跡化しておくことができる。また、事前に探索することで、対局時よりも探索に時間をかけることができる。

定跡データには、局面に対して複数の手を記録し、それぞれの手に探索時の訪問回数を記録しておく。対局時に定跡を使用する際は、訪問回数に応じた確率で手を選択することで、ゲームの進行が固定されないようにする。

3.14 マルチ GPU 対応

複数枚の GPU を使いニューラルネットワークの推論を分散処理する。

GPU ごとにキューを処理する専用スレッドを割り当てる。キューを処理するスレッドに対

して、複数のモンテカルロ木探索を行うスレッドを割り当てる。ゲーム木のノード情報は、異なる GPU に割り当てた探索スレッド間で共有する。

3.15 CUDA、cuDNN を直接使用

モデルの学習にはディープラーニングフレームワークとして Chainer を使用しているが、対局プログラムには、ディープラーニングフレームワークを用いず CUDA、cuDNN を直接使用する。Chainer で学習したモデルを読み込み、推論を行う処理をスクラッチで開発した。

高速化と対局の実行環境にディープラーニングフレームワークの環境構築を不要とすることを目的とする。

3.16 GPU ごとに異なるモデルの読み込み

モデルごとに誤る確率が独立である場合、複数モデルが同時に誤る確率は、単一のモデルを使用する場合より低くなる。GPU ごとに異なるモデルを読み込むことで、探索の精度を向上が期待できる。

4 学習について

4.1 自己対局による強化学習

4.1.1 学習データ、パラメータ

- 事前学習データ：elmo(wcsc27)で深さ 8 で生成した 4.9 億局面
- ミニバッチサイズ：64
- 学習アルゴリズム：Momentum SGD（学習率 0.01、慣性係数 0.9）
- 強化学習 1 サイクルで生成する局面：500 万局面
- 強化学習のイテレーション数：18（4 月末時点）

4.1.2 学習結果

【事前学習を行ったモデル】

- Policy Network の一致率：45.6%
- Value Network の一致率：78.1%

※ 既存将棋プログラムの自己対局データを増やすことでさらに一致率を上げられるが、Alpha Zero 方式の強化学習によって成果を上げたいため、収束する前のモデルから開始した。

4 月末時点で、18 サイクルの強化学習を実施した結果、学習開始モデルに対して、1 手 3 秒 50 回対局で勝率 81%となり、有意に強くすることができた。しかし、教師ありで収束するまで学習したモデルよりまだ弱いため、大会では強化学習したモデルは使用しないことにする。

4.2　教師ありによる学習

4.2.1 学習データ、パラメータ

- elmo(wcsc27)で深さ 8 で生成した 11 億局面
- ミニバッチサイズ：64
- 学習アルゴリズム：Momentum SGD（学習率 0.01～0.0001、慣性係数 0.9）

4.2.2 学習結果

- Policy Network の一致率：46.1%
- Value Network の一致率：78.1%

以上

第28回世界コンピュータ将棋選手権

PAL
アピール文書

山口 祐

PALの概要

- pal /ﾊﾟﾙ/【名】<話> 仲間、友だち、仲よし
  (例) a pen ー
- **ショーギはともだち** こわくないよ!

開発者

- 山口 祐 @ymg_aq
- 囲碁もやってます
  ➡️ https://github.com/ymgaq/AQ

PALの特徴

- Deep Learningを**使わない**
- 0ベースの評価関数の学習
- 強化学習と **敵対的学習** のハイブリッド
- クラウドサーバにおける**高回転並列学習**（N連ガチャ）
- 並列探索の最適化

使用ライブラリ

- やねうら王 v4.80　（改変しやすそうだったので）

Windfall

第28回世界コンピュータ将棋選手権アピール文章

作成：井本 康宏　作成日：*2018/3/*吉日

忙しい人のためのキーワード一覧

これさえあれば、1分で全てわかる

- 今回のWindfallは以下の要素で成り立っています
- 評価関数：deep neural networkに変更
- ゲーム木探索：なくても何とかなる
- 仕様：脱ビットボード, 合法手の計算をニューラルネットで実行
- プログラミング：Python, Tensorflow, sonnet, 今回はC++のコードはなし
- その他：なぜか後退した進捗

わからなかった方は、次項以降をどうぞ

開発者自己紹介

- 開発者
  - 井本 康宏
- 職業
  - エンジニア
- 棋力
  - 測ったことがないのでわかりませんが、すごく弱いです
  - 学生時代に囲碁・将棋部だったなんて言えない
- 開発のきっかけ
  - 趣味、好奇心、研究スキル、プログラミングスキルの向上を目的として、当時、話題だったこともあり開発を始めました

開発コンセプト

- 「せっかく作るのだったら、独創性にあふれるソフトにしたい」
  - (少なくとも開発を開始した当時の)既存のソフトの抱える多くの理論的な疑問点・問題点の解決
    - 例えば、どの局面を探索するか
      - 探索の深さ、消費時間
      - 枝刈り
      - 水平線効果
      - bias-varianceを考慮しない評価値
  - 理論的に有効性が確かめられた手法を取り入れれば、簡単に強いソフトが作れるはず
- 直近の半年は評価関数を重点的に作成

評価関数の開発

- 2駒関係を行列演算で表現するものからDNNに変更
- しかし、学習は思う様に進まず
  - 内部表現が全く学習できていなかった
- 仕方がないので、end-to-endでの学習をあきらめて特徴量を作り込むことに計画を変更

そして

なぜか、Tensorflowで差し手を生成するライブラリの開発が始まるのであった

差し手生成ライブラリ

- Tensorflow(ラッパーにsonnet)を利用して将棋の差し手を生成
  - たぶん世界初
- つい、カッとなって作った。公開はしていない。
  - そのうち公開する予定
- C++との連携を考えたくなかったので作りました
- 大量の局面を同時にGPUで計算すれば早いと信じています

しかし、大会に持参するPCにGPUはないのだよ

その他

- 半年前の電王トーナメントで利用したC++のコードは利用しない予定です
  - PythonからC++を呼び出しに謎のバグがあることがわかったので、今回はPythonのコードだけで参加します
  - スレッドの主な制御がC++側にあるので複雑怪奇で手に負えなくなりました
    - バグの原因はおそらくスレッド関係
  - その煽りを受けて、探索ルーチンもなくなりました
- 使用するライブラリもPythonから利用できるということで、python-shogiを利用します
- やねうら王は多分使いません

チーム Barrel house のアピール文書@第 28 回世界コンピュータ将棋選手権

Barrel house とは岡山の駅前にあるビアバーです．職場や社会人勉強会などで仲間を求めさすらって行きついたところで一人目の仲間を見つけたのと，マスターに許可頂いたので名前をお借りしました．その後，メンバーが増えて当初の方向性とは全く違って来ましたが，まぁ一度出した名前を変更するのもアレなのでそのまま行きます．

プログラム名：Hefeweizen
ドイツ南部の酵母入りビール．濁った白ビールってのが日本で通る表現かと．命名経緯は上記の通りです．

チームの特徴．
メンバーが全員初参加で冷たい白ビールのようなフレッシュなチームです．
メンバー間の擦り合わせも適当ですが昨年秋の電王トーナメント経験者が 2 名おり，多少は実績がありますので乞うご期待という感じです．電子工作が得意なメンバーや家電販売が得意なメンバーもおります．結成が遅れ現在手分けしている段階ですので確定事項は少ないのですが，やねうら王の探索部とオリジナルの局面評価関数と分岐先読みクラスタを使用する予定です．深層学習で作った評価関数や魔改造した技巧 2 などがあるのですが，本番で使われるかどうかはある程度ツールが揃って強さを計測してから決定する予定です．
疎結合なクラスタ構成なのでパーツ交換で対戦ごとに構成を変える可能性もあります．

CSA 使用可能ライブラリ使用表明．
メンバーの意思統一が図れておりませんので多めに入れておきます．また，試行錯誤ツールで色々使わせて頂いております．
Apery，やねうら王，tanuki-，Qhapaq，elmo，技巧，python-shogi，人造棋士 18 号
（作者本人は入れなくていい？）

使用マシン．
普通のノートパソコンに加えて，クラウドの力をお借りする予定です．ベンチマーク等も未だなので具体的なことは全く決まっていません．

4 月 13 日追記
ライブラリの選定理由を加筆修正しろとのことですので，追加します。

ライブラリ選定理由
python-shogi：python を用いた棋譜および局面の管理，sfen 文字列の展開など
Apery：評価関数作成のための教師データ作成に利用。
やねうら王：探索部が高速なため主に探索部の利用。定跡部作成時にも利用。
tanuki-：互換性があり強いため評価関数の作成及び仮想敵として勝率計算に利用。
Qhapaq：互換性があり強いため評価関数の作成及び仮想敵として勝率計算に利用。
Elmo：互換性があり強いため評価関数の作成及び仮想敵として勝率計算に利用。
技巧：評価関数の作成及び仮想敵として勝率計算に利用。探索部のオーダリングにも利用。
人造棋士 18 号：独自のルーチンを用いた評価関数の作成に利用。

クラスタ形式でのテストをまだ行っていないため本戦での運用で若干変更があるかもしれません。

第28回
世界コンピュータ将棋選手権
Claire アピール文書

2018年3月31日

上原 大輔

自己紹介

- ✓ プログラミング歴は4年
- ✓ 2015年10月頃から開発開始
- ✓ 第26回世界コンピュータ将棋選手権
  やねうら王ライブラリを使用して出場
  1次予選27位（2勝4敗1分）
- ✓ 第27回世界コンピュータ将棋選手権
  オリジナル勢に転向して出場
  1次予選16位（4勝3敗）

評価関数

- ✓ 駒割、3駒関係、ディープラーニング評価関数を使わない。
- ✓ 評価関数は駒の利きのみを使用。

その他

- ✓ 定跡はfloodgateの棋譜約8万局より抽出。
  局面の出現数・勝率を考慮し指し手を選択します。
- ✓ 探索も完全オリジナルで行きたかったのですが、
  時間がないのでStockfish9をベースに。
- ✓ 飛・角・歩の不成も探索します。

現在の状況

- ✓ 懲りずに前回大会後ゼロから作り直した結果、
  開発が間に合っていない。
- ✓ 最近なかなか開発時間が取れない。
- ✓ 知識・技術力が足りず、実装が遅い。
- ✓ 開発言語をRustに変えたが、
  まだまだ理解が浅く、実装速度が半分。

本大会での目標

- ✓ 時間切れ負け、反則負けをしない。（第26回世界コンピュータ将棋選手権では時間切れ・反則でそれぞれ1敗ずつしています）
- ✓ 1勝する。
- ✓ できれば前回の順位を超える。

最後に

- ✓ 最後まで読んで頂き、ありがとうございました。
- ✓ 去年よりも弱いかもしれませんが、
  お手柔らかにお願いします。

Anicca アピール文書

更新履歴
2018/03/30 初版作成

超個人的な事情により、今日(3/30)までほとんど開発できておりません。
さらに、言語をRustからGoに変更いたしました。
本番までに終局までさせ、評価関数に基づく指し手選択ができるようにするのが目標です。
(昨年、評価関数が全く機能していなかったため)

◯自己紹介
・都内の某IT企業に勤めております
・棋力はウォーズ2段程度です

◯名前の由来
・名前を決めようと思い立った直前に買ったアルバムの中で、一番好きな曲の名前を勝手に拝借しました
・当初はRustを予定していたので、「Rustanica」という名前にしようと思ったのですが、個人的な事情により
Go言語で開発することにしたので以前と同じ名前にしました

◯特徴
・ルール通りに指せます
・ライブラリを利用しない
・初手から定跡を利用しない（手抜き）

◯評価関数
・駒割と利きを予定しております
・手動調整

◯探索
・αβ法で枝刈りをします

第 28 回世界コンピュータ将棋選手権 PR 文書

【ソフト名】
十六式いろは改（よみ）じゅうろくしきいろはかい

【ソフト名の由来】
十六式の部分は、西暦 2016 年（6 月）からつくりはじめたことからです。
いろはの部分は、これからはじめるという意味をこめて。柔らかい感じを出したく、ひらがなで。
改の部分は、改装……開発言語を変更して新たに一から作り直したので。
そして「・」を抜きました。まぁ、由来はやっぱり昨年とほとんど変わらずです。

【開発者】
末吉 竜介（よみ）すえよし りょうすけ
棋力はどうも5級くらいらしいです。
（「どうぶつしょうぎウォーズ」では4級です……ずっと同じ……あれ？？？Σ(ﾟдﾟlll)w）

【ソフトについての大きな特徴】
「いろはは待たせませんっ！ Σ(ノ∀｀*)ﾍﾟﾁｯw」

【ソフトについて】
・ライブラリ不使用です。
・昨年出場したときから変わらず Lua です。（ホントは再度作り直そうと思ったりしたのですが）
・ Lua の特長は、軽量なスクリプト言語。さらに luvi（発音は、らび？）を利用して、
スクリプト言語の弱点である実行速度の遅さを、ある程度克服しています。
（参考） GitHub - luvit/luvi: https://github.com/luvit/luvi
・以前なんか**すごく目立った**気がしますが**気のせいです** Σ(ノ∀｀*)ﾍﾟﾁｯw
・**以前よりもまして弱くなったりした**ことがあるといういう不思議さん。
（でも、今回はちゃんと強くなっている！……はずw）

【ツイッターアカウントやブログ】
https://twitter.com/16shiki168
http://16siki168.seesaa.net/

【ソフトの独自性、主に今までとの相違】
・学習部
なし（……えっ！？）……でも、マジです orz

・指し手生成部
~~序盤は居飛車党で「箱入り娘」を一心不乱に目指し、その後は合法手をすべて探し、後述の評価方法により、指す手を選びます。だから、十六式いろは改は「娘（女の子）」だそうです。~~
~~あと、王手されるとそれを逃れる手を選ぼうとするはずです…… ※というはずだったんですが、~~
~~なんか箱入り娘を目指さなくてもいいかなという思いがありますが~~
娘なのは間違いないはずです、**天然な方向で**w

・評価方法や探索方法
駒の損得、~~駒の効率、~~玉の硬さ、~~利き、ひも付き、手の強さ、厚み・位取り~~を評価します。（何かひとつくらいは評価を追加させたいなぁ）
それぞれの評価値は、手作業で入力し調整します。機械学習はしません、というか今回はもそこまで手が回りません。
※そのときの**最低限の駒の損得は判断できるようになりました**（**評価値を出せるようになっています！**）が、次の手の探索にまったく活かしません……
駒を得しているなぁとか、損しているなぁとか思うだけです……えっ！？[2018/3 現在]
（選手権当日までには、活かせるようになっているはず！？）

~~探索は、反復深化法とミニマックス（アルファベータ）法ですが、基本的に「せまい」範囲を探索します……深くいけばいいのですが、たぶん浅い（3手先とか）です。……つまり、せまく浅く~~

~~(！？)探すので、静的評価の精度をよくすることを目指しています。~~(※と思っていたときもありました)
以下のモードを搭載しています。もっと機能を載せたかったのですがたぶん、搭載する時間がなく……[2018/3 現在]

・搭載モード
~~局地深読みモード：相手の打った位置を中心に 9～16 マスのみを深く探索し、できるだけ王手する。~~
局地暴走モード：相手の打った位置を中心に9マスに、ランダムに指す。
(↑できるようになりました！(*ﾟ▽ﾟ)ﾉ)
広域暴走モード(悪あがきモード)：合法手の中から、ランダムに指す。たまに奇跡を起こす！？(昨年からの**安定の平常運転**です(*￣m￣)ﾌﾟｯw)

【独自性・希少性】
開発言語は「Lua」です。今回は変えようと思っていましたが、時間がなくてそのままです。その Lua 特長は以下の通り。
「Lua は、C 言語のホストプログラムに組み込まれることを目的に設計されており、高速な動作と、高い移植性、組み込みの容易さが特徴である。」(Wikipedia より https://ja.wikipedia.org/wiki/Lua)
でも C 言語は使いません。さらに LuaJIT を含む luvi(発音は、らび？)を利用して、スクリプト言語の弱点である実行速度の遅さを、ある程度克服しています。
(参考) GitHub - luvit/luvi: https://github.com/luvit/luvi Lua(luvi)を採用した最大の理由は、簡単に実行形式のファイルを出力でき、ソースの量も C 言語よりも 20～30%くらい減りました。
現在公開されている将棋所対応のエンジンの中で、もっとも初心者にとって優しい弱さです。
(開発者の知り合いの**「初めて将棋を指した人」にもバッチリ負けました**！w)
１勝を実力でもぎ取るくらいの強さはあるかもしれませんw

私の知る限りでは「世界で一番強い将棋エンジン(開発言語「Lua」で作られたもので)」です。
……これ以外に開発言語「Lua」でつくられた将棋エンジンを知らないだけなんですけどw

【意気込み】
やっぱりずっと**完全にエンジョイ勢**です、すみません Σ(ﾉ∀`*)ﾍﾟﾁｯw
(今までエンジョイできたので満足中です)

……ですが、**記念参加勢**でもあり**イチから勢**(ライブラリ不使用勢のこと？)でもあります。
えっと、ネタ勢なんですか？？？どうなんですかね？w

とりあえずいつもどおり**予選当日の朝のギリギリまで**ガンバる予定です。

今回の目標も、実力で1勝すること(大事なことなので2回言いました。1勝は大事です！)と、私自身が楽しむこと。記録よりも記憶に残りたい！どれもいつものことです Σ(ﾉ∀`*)ﾍﾟﾁｯw
みなさまの応援うえるかむです。**どんどんどんとこい**なのです。
それに応えられるようガンバります！想像もつかない方向へ……えっ！？
(ヤバい、変な方向にハードル上げすぎた。普通な指し方になったらどうしよう(;´Д`))

2018.4.1

SBOYアピール文書

自己対戦を繰り返す事によって得られる局面毎のデータから勝つ確率が最も高い手を選択する評価関数の作成をめざしています。

Yorkies PR 文書

- アプリ名
  - 「ヨーキーズ」
  - ヨークシャテリア(犬種)の略称の「ヨーキー」の複数形
- 評価関数
  - KPPT形式の手番ありの3駒関係
  - elmoメソッドで学習
- 探索部
  - 置換表のキーを256bit化する。
- 定跡部
  - elmo(WCSC27版)をベースに追加する。
- 使用CPU
  - Mac BookPro(2015Mid)を本体にし、そこにGoogle Compute Engineのプリエンプティブ料金の複数台のVirtual Machine(以下、VM)を接続する。
  - VMのOSはUbuntu
  - プリエンプティブ料金のVMは、その時の遊休VMから割り当てられ、Cloud側の都合で割当が解除される場合があるため不安定だが、その分、安い。
- 本体とVM間のやりとり
  - 置換表の内容をやりとりする。
- 使用ライブラリ
  - やねうら王
    - 探索部と学習に使う。
    - わかりやすく書かれており、また、昨年11月の電王トーナメントで使用したため内容に慣れている。
    - SIMD化も含めて十分に最適化されていて、高速に動作する。
  - python-shogi
    - 本体とwcscサーバー側とのやりとりに使う。
    - わかりやすく書かれており、拡張しやすい。
    - 昨年11月の電王トーナメントで使用したため内容に慣れている。
- 学習用の教師データ
  - やねうらお氏が昨年11月に公開なさった110億局面(depth10)+自作50億局面(depth8)を使う。
- その他
  - C++17とPython3を使い、実装で楽できる部分は楽をする。
  - C++記述箇所は、以下を行い、コンパイラによる高速化に期待する。
    - Header Only化
    - なるべくconstexpr

こい将棋 アピール文章 （2018年3月31日） 児島彰

現在のところ，標準的な反復深化αβ探索＋手番付き３駒関係の評価関数の独自コードを書いています.

これから評価関数，高速化を工夫していきます.

現時点ではCSA使用可能ライブラリを使っていませんが，今後，使う可能性のあるものをライブラリ使用申請しておきます.

ライブラリの選定理由

Apery，やねうら王，tanuki-, Qhapaq，elmo，技巧，人造棋士18号

これまでに強い探索部や評価関数が作られているため

dlshogi，python-shogi

Deep Learningの導入を検討しているため