

カツ丼将棋アピール文書

――開発者紹介――

代表者：松本浩志

所属：NPO法人AI電竜戦プロジェクト 理事長(代表)

――今回の特徴――

- 今回は完全フロムスクラッチではなく、公開ソフトをカツ丼将棋exeで包んで出場します。
- カツ丼将棋と公開ソフトでコラボ、リレーしながら指します。
- カツ丼将棋exeにmultiponder(複数候補先読み)を実装し、**予選から決勝まですべてノータイムで指します。**
※自分の手番で考えてしまうと、単に公開ソフトを代理で出しているだけになってしまうので。。。
- 相手の強さを鑑みて、**穴角を目指す場合があります。**
- 不成を積極的に活用**し、相手の読み筋を外して時間を使わせます。
- 今回使う公開ソフトはやねうら王v6.0.0+水匠3



※育児多忙につき、まだ何もできていないのでひとまず去年と同じです。

※弱くてもフロムスクラッチ、完全自作のカツ丼将棋で出ます。奨励会入門レベルです！

――ポエム――

私は将棋倶楽部24でAIを出して、普通の指す将にも将棋AIに触れてもらおうということをやっている。それをやるためにカツ丼坊という将棋倶楽部24を自動で操縦するシステムを作った。これはUWSCを使ったもので画像認識を駆使したものである。画像認識を使うと人間のやる作業を直観的に再現できるのがよいところだが、先方のアップデートで画像作り直しリスクがあるのと、相手の指し手を認識するのに時間がかかる。遅い時は体感0.5秒か。そんな折に24がHTML5化された。いずれjava版は廃止されるので現在のカツ丼坊は作り直せばならなくなった。HTML5で動作しているので、python+seleniumで自動化ができるのと、chromの通信ログを見ることにより高速に指手を認識できる。中断対局の処理も簡単になった。せっかく指手が高速にできるなら極限まで光速で指すノータイム指しソフトを作ってみようと思った。それをやるためにはmultiponderを実装し、相手の手番中に複数手を考え、相手の指し手に対する応手を即指しする。候補になれば仕方がない0.2秒考えたものを指す。同様のことは第5回電王トーナメント準優勝のshotgunがやっているが、相手の時間を使わせるため微妙に最善手ではない、というのは今はしていない。いわばショットgun方式である。この手法で現在24で光の速さで指す非人間的なソフトとして放流している。Multiponder、簡単そうで将棋所の標準機能では実現できないので通信も自分でつくってないと難しいのでそれなりに価値があるかもしれない。

公開ソフトを使うことについては、AMD5950x+RTX3090のマシンを購入し、これはディープラーニングの勉強のためにそちら方面に移行するつもりで、今のカツ丼将棋を改良する予定はない。なので前述した通り、24でノータイムで強いという非人間的な友達のなくなるソフトとシステムを作ったので、今回の選手権でもそのネタでいく。公開ソフトをカツ丼将棋exeが包むことで、中身を知らないという馬鹿を輩出すると同時に、中身をまるっと入れ替えられるメリットもある。

電竜戦のきふわるべは、駒得のみの評価関数でユニークな動きをしていた。もしカツ丼将棋を駒得だけにしたらどうなるかとためしてみたところ、取る手がないので探索順序が9九から調べ始める関係で、9八香、9九角、その後6九の金が7九と6九を反復横跳びする穴角戦法が偶然にもできてしまった。せっかくできたのなら使えるところで使いたい。



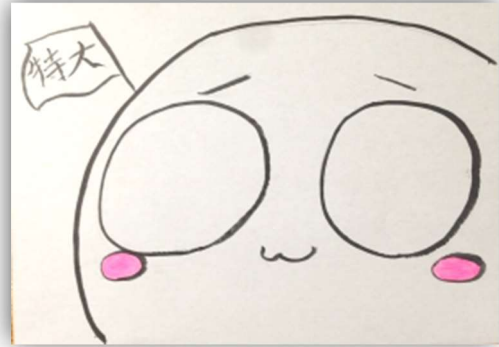
TMOQ アピール文書

2022 年 03 月 13 日 作成

2022 年 05 月 01 日 改訂

【ソフト名】TMOQ (特大もっきゅ)

“TMOQ”と書いて「特大もっきゅ」と呼びます、
愛娘が命名しデザインしたものです



【コンピュータ将棋大会実績】

2016 年の WCSC26 以降、ほぼ全ての大会に出場、
中位の成績をキープ

【TMOQ の特徴】

1. dlshogi ベース (WCSC29 より継続)
2. ネットワークに ResNet 9b を使用 (当初予定の GhostNet 15b は取下げ)
3. 学習データに「GCT の学習に使用したデータセット」+過去の TMOQ 棋譜を利用
4. 2000 万手を超える定跡 (Floodgate、電竜戦等の棋譜を寄せ集めたもの)
5. 『定跡チェッカー』別の USI エンジンが定跡手を評価・ダメ出し
6. 『入玉および駒得モード』MCTS の報酬に玉の縦位置および自分の駒数を考慮
7. 『入玉後ボタンタッチ』自玉が敵陣に入ったら、別の USI エンジンが宣言勝ちを目指す
8. Note PC を使用、莫大な計算資源がなくてもコンピュータ将棋は楽しめる！

【使用ライブラリ】

- ベースに「DeepLearningShogi」(Commit 790e2f4 on 3 Feb 2022) (GPL ライセンス)
- 『定跡チェッカー』および『入玉後ボタンタッチ』に「やねうら王 V7.00」(GPL ライセンス) を改悪して使用させていただきます

【御礼】

今回も山岡氏、加納氏&やねうらお氏を中心に、多くの方の公開情報により参加できました。
この場を借りて御礼申し上げます

第32回世界コンピュータ将棋選手権
なのは アピール文書 V0
2022年3月31日 川端一之

去年から なんの進歩も ありません

https://www.apply.computer-shogi.org/wcsc31/appeal/Nanoha/Nanoha_appeal2021v0.pdf

「芝浦将棋 Softmax」のチーム紹介

2022 年 3 月 25 日

芝浦工業大学情報工学科

糸川叶, 岩本裕大, 五十嵐治一, 湯浅翔太, 鳥居昭吾

1. はじめに

本稿は, 第 32 回世界コンピュータ将棋選手権 (2022 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 6 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大きく異なります. 伝統的な Min-max 探索 ($\alpha \beta$ 探索) ではなく, 本チームが独自考案した「MC Softmax 探索」を使用しています. 棋力的には従来の $\alpha \beta$ 探索やモンテカルロ木探索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

五十嵐は本年 3 月まで芝浦工業大学工学部情報工学科に勤務していた教員です. その他のメンバーは五十嵐研の OB です. 今年のチームリーダーは糸川で, 芝浦工大の修士課程に在籍しています.

3. 「芝浦将棋 Softmax」の特徴

本チームの特徴を, 以下の 1) ~ 4) のようにまとめました. このうちの 1) ~ 3) が本チーム独自の探索方式である「MC Softmax 探索」[1]に関する説明です. この探索方式は, 文献[2][3]の研究が基になっています.

1) モンテカルロ・ソフトマックス (MC 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」が伝統的な探索方式です. この探索方式では探索木のすべてのノードを探索する必要があります (全幅探索) が, $\alpha \beta$ 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 ($\alpha \beta$ 探索). この全幅探索に対して, 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. コンピュータ囲碁で有名な AlphaGo はこの探索方式を採用しています.

本チームの探索方式は後者の選択探索に分類されます．特に，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」(Monte Carlo Tree Search)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，我々は「モンテカルロ・ソフトマックス探索方式」(Monte Carlo Softmax Search)と呼んでいます．

2) ノードの局面評価関数と探索深さ指標を用いた確率的なノード選択方策

前項で述べた MC Softmax 探索によりルートノードから探索木を降りていく際には，本チームでは指し手の良さを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．

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

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 s' 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．（今回のバージョンでは，leaf でさらに静止探索を行っています）

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

$$E(s) = \sum_{x \in A(s)} \pi(x|s) E(v(x; s)) \quad (2)$$

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．

昨年のチームからは，「探索深さ指標」 $d(s)$ を Softmax 探索のための深さの指標として新たに定義し，これを用いて探索を行います．探索深さ指標は以下の式で定義しています．

$$d(n) \equiv \sum_{l \in \text{leaf}(n)} \text{depth}(n, l) \prod_{i=1}^{\text{depth}(n, l)} P(n_{i+1}|n_i) \quad (3)$$

ここで, $\text{leaf}(n)$ はノード n を根とする探索木の末端ノードの集合, $\text{depth}(n, l)$ は根 n から末端 l への経路の長さ, n_i は根 $n = n_0$ から末端 $l = n_{\text{depth}(n, l)}$ までの経路上のノード, $P(s'|s)$ は s から s' への選択確率 ($s \xrightarrow{a} s' \Rightarrow P(s'|s) = \pi(s'|s)$) です.

この式(3)を探索の中で効率的に計算するため式変形します. 末端ノードの探索深さ指標を0として, 内部ノードの探索深さ指標を以下の式で再帰的に求めます.

$$d(s) = \sum_{s' \in C(s)} \{P(s'|s)d(s') + 1\} \quad (4)$$

$C(s)$ は s の子ノードの集合です. この式変形による誤差は 10^{-10} 程度とごく僅かです(予備実験で確認済み).

この探索深さ指標 $d(s)$ を用いて, ノード選択時の評価値に次のような補正を掛けることで, 深さを考慮した探索を行います.

$$E_s(s) = E(s) \left\{ 1 - \frac{1}{2d(s)} \right\} + E_0(s) \frac{1}{2d(s)} \quad (5)$$

$E_0(s)$ は局面 s の局面評価値です. 静止探索を行わないなどの精度の低い評価関数を用いた場合は特に効果が大きく, 用いない場合よりもレートにして300程度棋力が向上しました.

3) バックアップ操作

MC Softmax 探索の全体の流れを図1に示します. ルートノードから, 2)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)の計算を繰り返す, 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. また, (1),(2)で定義される期待値操作の方法を「バックアップ方策」と呼び, 2)で述べた「ノード選択方策」と区別しています.

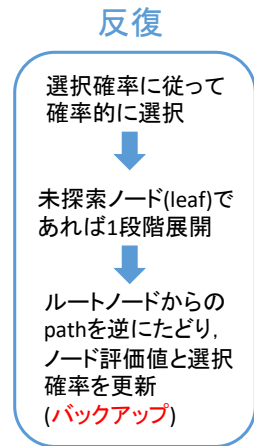


図1

また, MC Softmax 探索の名前の由来は, 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2)の過程は, ルート局面における各指し手の良さを求めるためのモンテカルロ・サンプリング (一種のシミュレーション) 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。

4) 評価関数について

昨年までは評価関数は既存の公開された評価関数をそのまま使用していましたが、実装が間に合えば MC Softmax 探索を用いた学習法によって作成した評価関数を使用することを検討しています。この学習法ではサンプリングを用いて勾配ベクトルを近似的に計算し、同時に複数の学習則を適用することが可能です[9]。詳細は 5. の後半に示しています。

上記の評価関数が間に合わなかった場合は、Apery の KPPT 型評価関数を利用します。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値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 $\alpha \beta$ 探索を使用しています。

4. 今後の課題

今年も、64 コア (128 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。

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て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、MC Softmax 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 というチームが採用してくれました[5]。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[5]。我々の研究グループもこのメリットを実験により確認しています[6]。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。

評価関数については、独自の学習法を考案しており、5 五将棋にて棋力の向上を確認できたため、今年度から使用していく予定です。将棋では「Bonanza メソッド」と呼ばれる教師付学習方式が有名ですが、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が、本学習法では、正解手以外の手（例えば有力な次善手）の評価値も学習データとして利用することが可能です。

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[1]。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[8][9]。

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 α β 探索) からソースコードのレベルまで、Stockfish 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[8]。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ります。また、探索と複数局面の評価に関する並列化の効果も高く、特に局面評価では GPU を用いたニューラルネットワークモデルの並列計算に向いています。さらに、プログラム作成が容易で、他のゲームプログラムへの適用も容易あることから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] 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) .
- [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] 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

クシヨツプ 2016 予稿集, pp.108-111(2016).

- [4] 岩本 裕大, 五十嵐 治一, “コンピュータ将棋における MC Softmax 探索のための探索深さの指標”, 第 25 回ゲーム・プログラミング・ワークショップ 2020 予稿集, pp.46-52 (2020.11.14-15) ,
- [5] コンピュータ将棋ソフト mEssiah の内部構造,
<https://qiita.com/sakuramaru7777/items/ebb397eef94fc02be2d8>
- [6] 吉野 拓真, 五十嵐 治一, 川島 馨, “MC Softmax 探索における局面の並列評価: GPU とニューラルネットワークモデルの利用”, 第 25 回ゲーム・プログラミング・ワークショップ 2020 予稿集, pp.16-21 (2020.11.14-15)
- [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] 五十嵐治一, 森岡祐一, 山本一将, “MC Softmax 探索における局面評価関数の学習”, 第 23 回ゲーム・プログラミング・ワークショップ 2018 予稿集, pp.212-219 (2018) ,
- [9] 岩本 裕大, 糸川叶, 五十嵐 治一, “MC Softmax 探索における局面評価関数の強化学習—5 五将棋への適用—”, 第 47 回ゲーム情報学研究発表会(2022.3.18-19, 東京), 情報処理学会研究報告, Vol. 2022-GI-47, No. 13, pp.1-8 (2022).

Argo(アルゴ) WCSC32 アピール文 2022/3/30

ソフト名/soft-name: Argo(アルゴ)

開発者/developer: 市村豊(いちむらゆたか/Yutaka Ichimura)

Twitter: @argonworks

Blog: <http://blog.livedoor.jp/argon1/>

「水匠3改」を元にして以前やねさんが配布していた150GBの教師データを3つ使って学習させたものが、水匠5に対して50回くらい対局させて勝率が40%くらい(20勝30敗くらい)になりました。現状はそれでやろうかと思っています。

使用ライブラリとそれを使用した理由

やねうら王: 使いやすくて改造をしやすいから。

水匠3改: 色々と評価関数を作っていてこれを元にしたのが現状で一番良さそうだったからです。

*****以上です。以下はおまけです。

この文章を読んでいただきましてありがとうございます。あなた様の大変貴重な時間を使ってこの文章を読んでもいただけること、感謝しております。

以下の文章は無理に読む必要はありません。個人的にはコンピュータ将棋選手権のアピール文を読む人にとって興味深く読めそうな作文を以下に書いたつもりでおります。それぞれの作文は独立していますので、興味がある作文だけを読んでいただいて大丈夫です。順番も好きな順番で読んでもらって大丈夫だと思います。

目次

作文1: コミュニケーション能力がないみたいな理由で会社をクビになった私のコミュニケーション論

作文2: ウェブ世界の素晴らしさを再認識した話

作文3: 「電竜戦」を作ったカツ井さんの偉大さについて

作文4: ビジネスと競技の違いについて

作文5: せっかく無職になったので一人で出来る仕事はないかを考えている最近です

作文6: 20回やって1回でもちょっとうまく行ったら勝利っていう勝利条件にすれば良い話

作文7: アニメの女の子はなぜ可愛いのか、「りゅうおうのおしごと!」を例に考えてみる

作文8: 株価の値動きを予想する話・「IDECO」と「つみたてNISA」が株式投資において理論的に最高の投資手法だと個人的に思う理由について書いてみる

作文9: マルクスの「資本論」を勉強して、金持ちになる方法を考えてみた

作文10: 仏教が何を主張しているかをなるべく頑張ってわかりやすく説明してみる

作文11: Argo 開発物語

作文12: 市村さんのおしごと

作文 1: コミュニケーション能力がないみたいな理由で会社をクビになった私のコミュニケーション論

他人と不必要に関わらないことを心がける感じです。

経済学にはいくつかバージョンがあって、バージョンごとに前提となる考え方が違っていたりします。

近代経済学というのか新古典派経済学というのか、というバージョンの経済学を私は基本的に勉強したのですが、その前提として「人間は自分が得をすることをする(人間は自分が損をすることはしない)」というものがあります。実際からは多少ずれている仮定ではありますが。

その仮定に則ると、「私(市村)が得をするならば他人と関わる」わけですが、このときに他人も同じように自分が得をすることだけをするので「私(市村)に関わられて得をする状態じゃないと他人は私(市村)と関わらない」わけですよ。ちょっと話が込み入ってきましたね。将棋ソフトの開発で言ったらミニマックス探索みたいな感じで、相手のことも考慮して自分のやることを考えるって話ですよ。他人と関わることにしては自分の都合だけを考えてはいけません。他人が他人にとっての最善の振る舞いをするのを考えた上で、自分が取るべき振る舞いを考える。

それなので、私と他人とがコミュニケーションをとることで双方が得をする状態じゃないとコミュニケーションをとるという状態は実現しないわけですよ。

それで、最近の私は、どういうときに私と他人が両方とも得をするという状態が実現するのか？ ということを考えているわけなんです。それというのも、率直に思うこととして、私と他人がコミュニケーションを取ることで両方とも得をするという状態はまずそんなにしょっちゅう発生するものではない、ように感じるんですよ。だからそれがどういうときに発生するのかということを考えているわけです。

42 歳になって考えて、私に関わられて他人は得をしないんじゃないか？ ということを私は思うんですよ。私が他人とコミュニケーションを取ることで、私に関わられた他人はなにか得をすることがあるのだろうか？ を考えてみて、他人が得をすることがないんじゃないかと思う状況においては関わらないようにする、ということを私は最近はあるべくするようにしているんですよ。

インターネットの良さを再認識している最近です。ツイッターとか FACEBOOK がすごく良い。だいたい非同期コミュニケーションだし。非同期コミュニケーションが素晴らしいもので可能な限り非同期コミュニケーションにしようとは私は思っています。ツイッターも、私がツイートをして、他人のツイートを見て、いいねとか RT をすれば、それで十分にコミュニケーションを取っているじゃないかと個人的には思います。というか、現実世界で直接話すよりもツイッターでなんとなく興味のある人のツイートを見ている方が遥かに良いと思っているのでそういう他人との関わり方をするようにします。みんなそうすれば良いんだ。

個人的には、「ルパン三世」のルパン一味とか「HUNTER×HUNTER」のハンターの人間関係が理想的だと思っています。

ルパン一味って、次元とか五右衛門とか、ルパンと四六時中いつも一緒にいるわけではないし、別にルパンの部下って言うわけでもないんだけど、ルパン一味って「仲間」じゃないですか。あの関係性が素晴らしいというか理想的な有り様だと個人的に思っているんですよ。

「HUNTER×HUNTER」のハンターの皆さんもルパン一味と同じような感じで、四六時中いつも一緒にいるわけじゃないけど、何かあると皆ワツと集まってきて一つのプロジェクトをみんなで達成するじゃないですか(グリードアイランドを攻略するとかアリを倒すとか)。私がコンピュータ将棋の大会とか開発者のみなさんのことを好きなのは、この「HUNTER×HUNTER」のハンターの人たちみたいだからです。世界コンピュータ将棋選手権や電竜戦があるとワツと集まってきて皆で一つのプロジェクトを達成するっていうのが、「HUNTER×HUNTER」で、アリを倒すぞっていうときにハンターの皆さんが集まってくる感じに似ていると思うのでそういうのが個人的に好きです。

作文 2: ウェブ世界の素晴らしさを再認識した話

42 歳になりました。で、42 歳まで現実世界をそれなりに見て回ってどういうものか多少は分かった感じがしました。そうして「インターネットってすごいものだ」ということを再認識するようになりました。まず、なんで移動をしなくちゃいけないのかに腹が立ちます。私はとにかく移動をしたくない。私は家から大学に通うのに片道 1 時間半くらいかけて通学をしていましたが、それを平日に 5 日間やっていると往復で 3 時間の移動するのにものすごく疲れるんですよ。その時はそういうものだと思っていましたが、エンジニアだというのは科学技術の力で移動をしなくても良い世界を作らないといけないと今の私は思います。そして私は外に出るのが嫌です。だいたい交通事故で死ぬかもしれないし、通り魔みたいな人に刃物で刺されるかもしれない。家にいたって地震で家が倒壊して死ぬかもしれないということは思いますが、確率的に考えると、さすがに外に出ることよりは家の中にいることの

ほうが安全度は高いと思う。だから、家から出ないでインターネットだけで生活が成り立つように科学技術の力で実現したいと思います。「ひきこもり」といいますが、別に家から出ないことそれ自体は悪いことではないと思う。だいたい、宇宙ステーションとか、これから作る月面基地とかだったらそこにいる人は基本的にずっと家から外に出ない「ひきこもり」じゃないですか。現在、国際宇宙ステーションにいる宇宙飛行士は言ってみれば全員ひきこもりじゃないですか(船外活動する人もいますが)。学校にしても会社にしても、なぜ物理的に移動をして複数人で大部屋で作業を長時間するのかということが疑問です。これまではそういうものかとなんとも思ってきましたが、冷静に考えると科学技術の力でなくさないといけなく私は力強く思います。それなので、各人がそれぞれの部屋から非同期コミュニケーションで遣り取りをするということを私は理想としている人なので、その状態の実現を目指します。

「メタバース」と呼ばれている、三次元の仮想空間のことだと思いますが、この「メタバース」という言葉についてはいわゆるパスワードですぐに使われなくなるかもしれないですが、VRとARがこれからますます重要になるということは私もそうだと思うんですよ。

国立科学博物館「うちでかほく」という、博物館をVRで再現したというコンテンツがあったので見てみましたが、これは良いなと思いました。というか、世界中の博物館とか美術館とか、世界遺産みたいな観光名所とか、あるいは日本で言ったら秋葉原みたいな場所とか、そういうのをVRで見れたらすごく良いんじゃないかと思いました(個人的に大英博物館とルーブル美術館とスミソニアン航空宇宙博物館は見てみたい)。通信の「5G」についても、サッカーの試合みたいなのを特等席からのVR映像みたいなのをリアルタイムで配信してそれをVRゴーグルとかでみんなで見たら良いんじゃないかと言う話は私もそうだと思います。アイドルとかのコンサートについて、最前列にカメラを置いてそのVR映像をみんなでVRゴーグルで見たら、全員が最前列からの眺めでアイドルのコンサートとか見れるじゃないですか。それって良いんじゃないですかと思うのですが。

私は2005年からコミックマーケットという同人誌即売会に参加している人で20回くらいはサークル参加をしている人ですが、とにかくコミケは人が多すぎる。おかげで行列という無意味な苦行をしないといけなくのですが(コミケは夏は暑くて熱中症で倒れる人が出るし冬は年末にやるので寒くてしょうがない)、メタバースで同人誌即売会をしたら良いんじゃないかと思うんですよ。3Dのアバターは仮想空間で重なって良いので行列をしなくて良いじゃないですか。それでコミケっていうか、同人誌即売会は提供物が基本的には「本」が中心なので、それは情報に価値があるものだから電子書籍で良いんですよ。だから3Dの仮想空間で同人誌即売会をして電子書籍を売るということで良いと思うのですが。

図書館で日刊工業新聞を読んでいたら宇宙商社のスペースBDのCEOが「人類にとって残されたフロンティアは宇宙空間と仮想空間の2つだと言われている」みたいなことを言っていたのが印象に残っているのですが、個人的には私は「仮想空間」の可能性にワクワクしている人です。VRとARが発展してすぐ実用化したらきっと素晴らしいものになるはずだと思ってワクワクしているし早くそうなってほしいと思っているのです。

2019年12月に発生して、2020年3月くらいから日本では深刻化した新型コロナウイルスについて、「起こらなかったほうが良い厄災」には違いがないですが、結果的にZOOMのようなオンライン会議が一般化して、テレワーク(在宅勤務)が一般的なものになったことは、悪いことばかりではなかったと思っています。どうやらやれば出来ることだったみたい。この事によって「あれ、物理的に移動をして学校とか会社とかに複数人が一箇所に集まらなくて良いんだ」ということを再認識したわけですよ私は。京都アニメーションの火災で多くのアニメーターが亡くなられた事件についても、そもそも多くの人が在宅勤務で作業をしていて物理的に違う場所にいたらこういうことは起こらなかったのではないかなと思うこともあります。物理的に職場に行くと家が空室になってその空室分に家賃を払わないといけなく、それが在宅勤務になったらオフィスの家賃を払わなくて良いんですよ。その分効率化してモノやサービスの価格が下る。

作文3:「電竜戦」を作ったカツ井さんの偉大さについて

「電竜戦」を作ったカツ井さんはすごいクリエイターだということを思います。そしてカツ井さんを目標にするべきだということを思います。

将棋ソフトを作る、ということが価値であるように「将棋ソフトの大会を作る」ということもまた価値である。というか、大会を作るほうが上位のことをしているんじゃないかということです。

「電竜戦」を作ったカツ井さんがどのくらいすごいかというと、AtCoderを作った高橋直大さんとか、QuizKnock【クイズノック】の伊沢拓司さんと同じくらいにすごいことをしているというふうに思います。

それで、高橋直大さんがAtCoderを売るか上場させると、高橋直大さんは資産が10億円以上になるといいます。伊沢さんも同じです(伊沢さんは現時点でも資産が10億円以上あるかもしれないですが)。

それなので、カツ井さんが「電竜戦」を株式会社にして、それを売るか上場させると、カツ井さんの資産が10億円以上になると思います。それをするかどうかはともかくとしてやろうと思えばそれが出来ると思います。だからカツ井さんと同じことが出来るようになると、その人は資産を10億円以上にすることが出来るわけです。だからカツ井さんを目標にしてカツ井さんと同じことが出来るようになるべく頑張った方が良いと思うんですよ。

カツ井さんの「電竜戦」がどうして継続的に集客できているのか？ ということについては、「顧客に対して価値を提供して顧客を満足させているから」だと個人的に思います。

それでは、カツ井さんの「電竜戦」が顧客に対して提供している価値は何か？

それを考えると「友達」の提供じゃないかと思います。

前にも書いた気がしますが、私は価値があるものというのは「時間、仲が良い友達、仲が良い家族」この3つがすごく重要なものだと思っています。この3つは「いくらお金を出してもほしいが、いくらお金を出しても買うことができないもの」なんですよ。

人生は限り・寿命があり、時間は希少資源なんですよ。だから時間売る(時間を節約する)ものに対しては人はお金を払うんですよ。電車に乗るときの電車賃、炊飯器、洗濯機、食器洗い乾燥機はあれは時間を買っているんですよ。あとはマクドナルドが売っているものは時間、とか(マクドナルドの提供価値は「食べ物」ではない)。この辺は白坂慎太郎さんの動画の受け売りです。

仲が良い友達について。ツイッターで見た話ですが「キャバクラというのは『お金を払ってくれたら友だちになってあげる』というビジネス」というものです。私は42歳ですね。冷静に考えて、42歳の無職のブサイクなおじさんである私はもう、「お金を払ってくれたら友だちになってあげる」という人にお金を払って友だちになってもらうという以外に新しく友達を作る方法はないんですよ。そのことを自覚して「なるほどー」と感慨深い気分になりましたが、そんなわけで、「友達」を提供するということをするとそのことに対しても人はお金を払うんですよ。カツ井さんの「電竜戦」、いわゆる「コミュニティ」というのは、この「友達」を提供するということをしているサービスだと個人的に思うんですよ。ツイッターとかFACEBOOKとか、あるいは「5ちゃんねる(旧・2ちゃんねる)」がやっていることと同じ。

それなので、カツ井さんの「電竜戦」というのはすごいことをしていると個人的に思うわけです。そういうふうにしてインターネット上を見てみるとコミュニティを作ろうとしている人はちらほらと見つかるわけですよ。星さんがDiscordを衆院選のときに作ってコミュニティを作ろうとされていたのを見てみると、星さんはこの「コミュニティに価値がある」ということを分かっている人だと思うんですよ。顧客と言うかフォロワーというのは価値があるものだからです。だからそれを取りに行った。

私が読んだ本で農林中央金庫の投資家・奥野さんの本(奥野一成「ビジネスエリートになるための教養としての投資」ダイヤモンド社)で、「ビジネスというのは参入障壁を作るゲーム」というのが印象に残っています(「参入障壁」というのは他の人が簡単に真似をすることができない要素のこと)。あとは何かの本で読んだ(今井孝「起業1年目の教科書」かんき出版、だったような気がする)「ビジネスモデルをみるときは集客・提供価値・マネタイズ(収益化)の3つに注目して見る」というのが印象に残っているので、個人的に私がビジネスモデルについて見るときは「集客・提供価値・マネタイズ(収益化)、参入障壁があるかどうか」という4点について注目して見るのが最近が多いように思います。

カツ井さんの「電竜戦」について上記の視点で見えてみると、集客ができていて価値を提供している。多分簡単に真似できないので参入障壁もある感じがする。だからマネタイズ(収益化)をしていないだけです(AtCoderのマネタイズ(収益化)については求人広告を載せているからその掲載料もしくはAtCoder経由で採用した場合にお金をもらおうとかっていうことでマネタイズをしているのかと思うのですが他に何をしているのかがよく分かってないです。あとはグーグルが初期は検索連動型広告でマネタイズをする前は「すごい検索エンジンだけどどうやってお金を稼ぐんだろう？」と言われていたり)。HEROZ株式会社の「将棋ウォーズ」について、参入障壁が何かということを考えてみると、顧客がついているということだと思います。私が「将棋ウォーズ」を丸パクリしたサービスを作ろうとしたときに何がパクリにくい？ というのは「将棋ウォーズ」はサービスをプレイしている顧客が常時5000人いるんだっただかな。その状態が一番パクリにくいと思います。なので、顧客というかフォロワーと言うか、はすごく大事で価値があるものだと思います。ちなみに、「特許を取得しているから他の人がパクれない」ということももちろん参入障壁です。ただ、特許を取得しているということはそこまで強力な参入障壁ではない。特許とかなくてもどうやっても他が真似できないというような状態が強いです。その具体例がグーグルの検索エンジンで、今からグーグルの検索エンジンと同じものを作ろうとしても、仮に特許とかなくて好きに作って良いと言われても作るのが大変じゃないですか多分(グーグルの検索エンジンはインターネット上にあるすべてのデータのコピーを自分の手元に持っていないと作れないと個人的には認識している)。その奥野さんの本だと「GAFAの中で一番参入障壁がないのがFACEBOOK」だと書いてありました。FACEBOOKのサービス自体は割とパクリやすい。実際、FACEBOOKの前に日本だと「mixi」というSNSが一番だったけど割とあっさりFACEBOOKに取って代わ

られた。だから FACEBOOK のサービスには参入障壁があまりない、という話を書いてありました。あとは UBER は参入障壁があまりないとか。UBER がやっているライドシェアと料理を宅配するサービスは基本的にただのマッチングアプリだから割と簡単に誰でもパクれる。実際アメリカでライドシェアは競合のリフトがあったりしてそこまで圧倒的ではないし、日本のウーバーイーツについてもウォルトとか出前館とかがあってそこまで圧倒的ではないとか。

作文 4: ビジネスと競技の違いについて

他人と違うことをして他人と補完関係になるのがビジネスの目標じゃないかと個人的に思っています。

アダム・スミスの「国富論」という本があってですね。経済学という学問の一番のもとになった本がこれだと思うんですけど(250年前のこの本が一番の元なので経済学という学問は250年くらいの歴史しかない結構歴史が浅い学問です。ちなみに物理学が「自然現象を数学的に取り扱う行為」であることに対して経済学は「社会現象を数学的に取り扱う行為」というのが私の物理学と経済学の定義です)。仕事をするときは「複数人で分業と協業をすることによって効率が向上する」という話をしているみたいです。ピンを作る話が例として出てきて、一人でピンを作ると一日で10本くらいしか作れないけど、複数人で分業と協業をすると4人で1日に7000本のピンを作ることが出来る、みたいな話を書いてあるみたいです(この喩え話がどこまで本当かというのはともかくとして)。「国富論」においてはこのことを国家間の関係にも拡張します。アメリカの前の大統領のトランプさんが「アメリカは貿易赤字で、貿易赤字だとアメリカは損をしているんだ」みたいなことを言っていた記憶があるのですが、アダム・スミスの頃はそういう考えが標準的だった。それに対して「国富論」において、工業国は工業製品を作ることに集中をして、農業国は農作物を作ることに集中をして、その作った工業製品と農作物を貿易で交換するということをする双方が得をするんだ、ということを主張しているみたいです。

将棋とかスポーツのような競技になると、みんな同じことをして誰が一番だとか順位がはっきり出たりするのですが、それは実はビジネスとは違うというか、実社会的には割と珍しいあり方じゃないかと思うんですよ。

ビジネスというのは、他人と違うことをして他人と補完関係になることを目指すムーブメントじゃないかと思うんですけれど。

無人島に3人いるぞ、とかなったら、一人は専門の漁師になって魚を捕る、一人は専門の米作り農家になって、一人は専門の野菜農家になって、それぞれが作った魚と米と野菜を交換するということをするほうが良いわけですよ。三人全員が魚をとって米を作って野菜を作るということをするよりも、分業と協業をするほうが全体の効率が向上すると思うんですよ。それでその場合に魚をとっている人と米を作っている人と野菜を作っている人の誰が一番かとかは順位付けできないんですよ(やっていることが違うので)。そういうのがビジネスの考え方で、だからビジネスをする上では将棋やスポーツのような競技をすることは悪影響があるというか、根本的な考え方が違うので、ビジネスはライバルに勝つとか考えなくても良いですよ多分。朝日新聞のオピニオン面でビジネスとスポーツが違うって話をしているのを読んだ記憶があるのでその話を書いてみました。

他人と違うことをするという例として台湾の受託製造メーカー(TSMCとかシャープの親会社の鴻海とか)の話をしします。かつて1990年頃のバブル崩壊まで日本の電機メーカー(パナソニック・日立・東芝・三菱電機・シャープ・NEC・富士通とか)が強かったんですよ。で、台湾のメーカーは日本の電機メーカーと同じことをしていたら勝てないと思ったので、「受託製造」という製造に特化するというところをした(日本の電機メーカーは自分で考えて自分で製造もしていた)。その結果、製造については台湾のメーカーのほうが日本メーカーよりも優れている状態になった。そして考える能力はアメリカの会社の方が日本よりも優れているんですよ。だからアメリカの会社が考えたものを台湾の受託製造メーカーが作る、という製品に日本メーカーは勝てない(考える能力はアメリカに劣り、製造については台湾のメーカーに劣る)という状態になっているのが現在なんだと個人的に思っています。台湾に限らず、1990年頃のバブル崩壊まで日本メーカーが安く大量生産するということで強かったときに、ヨーロッパのメーカーは安く大量生産するということをしなくなったりとか、やることを変える・ライバルと同じことをしないんですよ。それがビジネスの基本じゃないかと個人的には思うんですけれど。

作文 5: せっかく無職になったので一人で出来る仕事はないかを考えている最近です

経済学的には仕事みたいなことをするときには複数人で分業と協業をするということをしたほうが全体の効率が上がるというのがほとんど常識的な考えだと思います。それなので大体は「会社」みたいに複数人で仕事をしているところが多いわけなんですけど。それなのでまず基本的には「複数人で分業と協業をする」ことをしても効率が上がらないことは何か？ ということを考えてみる。「小説を書く」ことは複数人でやることで効率があまり上がらないと思います。だから「小説を書く」という仕事は相対的に一人でやることに向いている仕事になります。これが「マンガを描く」ということになると、「複数人で分業と協業をする」ことで効率が上がると思います(漫画家本人がメインキャラの作画をして、アシスタントが背景を描く人がいて、モブキャラを描く人がいて、スクリーントーンを張る人がいて、みたいな感じ。そもそも話を考える人と作画をする人が違ってたりする)。

それで結構考えてみたのですが、「小説を書く」以外に「複数人で分業と協業をすることで効率が上がらない」仕事というと、小説に限らずに作家業・判断を下す仕事系(経営者・投資家・政治家・学者)とかがあるのかなと思いました。

それなので、「複数人で分業と協業をする」ということと、「一人で仕事をする」ということを両立させることを考えます。言っていませんでしたが、複数人で仕事をするにはデメリットもあります(調整にコストがかかるとか労働の疎外とか)。そんなわけで、なんとか一人で仕事することはできないのかということを最近の私は考えているのです。テクノロジーが進めば機械を使うことで一人で仕事ができるようになるはずですが。現時点においてもそれは実現できないのだろうか？ ということを考えています。

リナックスの OS を作ったというプロジェクトがすごいことだという話を色んな所で読みます。「伽藍とバザール」という本にもその話を書いてあったと記憶していますが、リナックスの OS を作るプロジェクトは、インターネットを通じてボランティアのプログラマーが寄ってたかって皆ですごい OS を作り上げた。OS はマイクロソフトみたいなところがすごい人数で管理をしながらじゃないと作れないようなものだと思っていたのに、こんなやり方でリナックス OS というすごい成果物ができてしまったという話です。

<https://www.aozora.gr.jp/cards/000029/card227.html>

↑「伽藍とバザール」の全文は以上で無料公開されています。

それで、リナックス OS を作るみたいなことをして、その貢献度に応じて報酬を得ることができるというようなシステムになっていたならそれは良いんじゃないかと思うわけです。それで、そういうのが実現されるんじゃないかと私は思っていてそういうふうな形で報酬を得るということをして生きていけないだろうかということを考えているのです。

それで注目しているのが DeFi のプロジェクトです。

仮想通貨というのか暗号資産(クリプトアセット)からクリプトと通称するのか、そんな感じのものについてですけど。

仮想通貨について、「なんだかよくわからないお金みたいなモノ」という認識の人も多いような気がしますが、で、一番最初に作られた「ビットコイン」というものはそういう認識でまああっていると思いますが、実は現時点の仮想通貨というのか暗号資産(クリプトアセット)については「お金」じゃないものもたくさんあります。その話を書きたいんですけど。

お金じゃないならば何って言ったら株式に近いものなんですけど。なんかのサービスみたいなものがあってそのサービスの権利みたいなものです。

もうちょっと具体的に書きますが、まずビットコインが作られたあとに「イーサリアム」というシステムと言うかプラットフォームが作られたんですよ。それはビットコインがそれしかやり取りすることができないシステムと言うかネットワークであることに対して価値があるものは何でもやり取りできるシステムと言うかネットワークと言うかプラットフォームを作った。それが「イーサリアム」です。それは言ってみればスマートフォンの iOS とか AndroidOS みたいな感じで、その上にアプリを作ることができる。

成功したアプリが出てきたんです。それが「ユニスワップ」というものと「コンパウンド」という2つのアプリと言うか、サービスなんですけど。

「ユニスワップ」は仮想通貨の交換所です(現実世界で言ったら通貨の両替所みたいな感じ)。ビットコインとイーサリアムを交換するみたいなことができるサイトです。それで、「コンパウンド」は仮想通貨の銀行です。仮想通貨を預けて利息を得ることができるし、預けた仮想通貨を担保にして自分が仮想通貨を借りることができる。その2つのサービスが成功して一気に「これはすごい」という感じの評価になったんだと思うんですけど。

それで、そもそもプラットフォームである「イーサリアム」についても、後発の互換品みたいなのがたくさん出てきています(iPhone に対して後発の互換品である AndroidOS が出てきたみたいな感じ)。それが「バイナンス・ス

マート・チェーン(BSC)」というバイナンスという会社が運営しているシステムだったり、あとは「ポリゴン」というシステムは私は使っています。全部追いきれていませんが結構たくさんあります。それで BSC 上で、ユニスワップの互換サービスとしてパンケーキスワップという仮想通貨の交換所があって、コンパウンドの互換品としてヴィーナス・プロトコルという仮想通貨の銀行みたいなサービスがあったりします。

それで仮想通貨に投資をするとと言うと、かつてはビットコインを買って持っていたら値上がりしました、みたいなのが中心でしたが、現在はパンケーキスワップという仮想通貨の交換所に仮想通貨を預けて利息を得る、みたいなことが仮想通貨の投資の中心的なことになっていると思います(「流動性マイニング」といいます)。そのサービスの維持に協力すると協力した分だけ報酬がもらえる、みたいな感じです。

それで、いろんな実験的なことを現在進行系でバンバンやっているのが仮想通貨と言うかクリプトの世界だと思うんですけど。

個人的に興味深いプロジェクトとしていくつか紹介すると、

AWS や GCP をみんなで作ろうという感じのプロジェクト

<https://bitjournal.bitcastle.io/post-8287/>

コンピュータのリソースをインターネットを通して売買するサービス。

ギャンブルと見せて群衆の叡智による未来予測・保険を中央管理者無しで実現する

「Augur」

<https://news.mynavi.jp/cryptocurrency/augur/>

広告を見ると広告料がもらえるブラウザ。

ベーシックアテンショントークン

<https://fisco.jp/media/bat-future/>

保険(オプション)の売買をする事ができるサイトというサービス

ヘルメット

<https://www.owl-coin.com/coins/helmet-insure>

↑こんな感じの興味深いサービスみたいなのが立ち上がっています。

仮想通貨についてはマナブさんのメルマガが情報源としては良いのではないかと個人的には思っています。一読するとためになるんじゃないかと思います。

<https://manablog.substack.com/>

作文 6:20 回やって 1 回でもちょっとうまく行ったら勝利っていう勝利条件にすれば良い話

自動車レースの F1 の話をします。F1 は大体 10 チームあって 20 台でレースをすることが多いのですが、最下位のチームが何をしているかという話です。

F1 は今はどうか知りませんが大体 20 台でレースをして 10 位に入ると 1 ポイント取れるんですけど(9 位は 2 ポイントで 8 位が 4 ポイントだったか)、最下位のチームは基本的には 10 位以内に入ることが難しいです。それで、一年間で 20 戦するうちに一回でも 10 位以内に入って 1 ポイント取ると、年間チームランキングが 9 位になって最下位じゃなくなることが多いです(最下位は 0 ポイントのことが多いので)。それなので、F1 で最下位のチームというのは「20 回レースをして一回でも 10 位以内に入ったら勝ち」という勝利条件でバトルをしているわけなんです(年間チームランキングがビリの 10 位かビリのひとつ上の 9 位かで賞金の額が変わる)。

この勝利条件というのが個人的に面白いと思っていて、私もこんな感じの勝利条件でやっています。こう思うと割と気軽にチャレンジが出来ると思うので。

野球で言ったら「三割打てたらプロ、四割打ったら超一流」という話は時々聞きますがそれに近いです。全部勝たなくて良いんですよ。「20 回やって 1 回でも 10 位以内に入ったら勝利」というのが勝利条件です。そう思ったら 10 連敗しても問題ないんですよ。むしろ最初の 10 戦なんて練習というかデータ取りじゃないですか。

何かをするときはそういうふうにしてやってみると良いと思います。もちろんコンピュータ将棋の大会みたいなものそんな感じでやるものです。

作文7:アニメの女の子はなぜ可愛いのか、「りゅうおうのおしごと！」を例に考えてみる

アニメの女の子になることはできないのか、ということを考えてみた。アニメの女の子が可愛くて魅力的である。そうであるならば、私はアニメの女の子になりたいと思うのは必然であると言える。

アニメの女の子が可愛いのは「挑戦をしているから」だと思っています。

ジブリがアニメ化したことで知っている人も多いかもしれませんが「ゲド戦記」という児童文学作品があり、元は英語の原作を日本語に翻訳した翻訳者のエッセイを読んでいたら、児童文学が普通の文学よりも下に見られていることに反論する文章があり、その理由の一つとして「児童文学は幸福を描くハッピーエンドの話が多く、普通の文学は不幸を描くバッドエンドの話が多くて『だから児童文学は嘘っぱちだ』という批判があるが、厳しい現実を否定するよりもあえて肯定するほうがよほど難しいことなのだ」という意味のことを言っていた。「『めでたしめでたし』で終わる話を語っていた昔の農民は自分が決してめでたい境遇でないことは分かって話をしていたはず」だと。

日本のアニメやマンガというのは児童文学と同じ系譜に属するもので、幸福を描くハッピーエンドの話が多いという印象を個人的には持っています。

アニメとかマンガとか、そういう作品でよく出てくるのは、「困難な目標に挑戦する・自分よりも強い相手に挑戦する」という話です。で、アニメとかマンガってというのは、序盤から中盤にかけては苦戦するけど最終的には勝利するというストーリーになっていることが多い。ちょっと考えてみると、これはフィクションと言うか、嘘なんですよ。なぜならば、実際の人生においては「最後に勝利する」ということは起きないことが多い。大体の人は、人生は最初から最後までずっと負け続ける。なので、「最後に勝利する」というストーリーになっているアニメとかマンガは嘘であるわけなのですが、アニメとかマンガを読んでいるとそんなことは大して重要なことじゃないということがわかるわけですよ。

白鳥士郎さんの小説「りゅうおうのおしごと！」を例にしてみますが(私は現状は9巻までしか読んでいませんが)、この作品も「困難な目標に挑戦する・自分よりも強い相手に挑戦する」ということが繰り返し描かれています。で、読んでいるときにはあまり気にしなかったのですが、気をつけてみると、実は形式的には負けていることが結構あるんですよ。個人的に印象に残っているエピソードとしては、

清滝桂香が女流棋士資格を獲得しようとするけど女流棋士になれない話(4巻)。

清滝鋼介がB級2組残留を目的に順位戦を闘うが、C級1組へ降級する話(7巻)。

夜叉神天衣が女流タイトル戦で空銀子に挑戦して敗北する話(9巻)。

・あなたの竜王戦を闘うという話。

そもそも話が重層的なんですよ。

清滝桂香が女流棋士資格の取得を目的に死ぬ思いで闘う。女流棋士になった夜叉神天衣は女流タイトル戦を死ぬ思いで闘う。女流タイトルを取った空銀子は四段昇段を目的に三段リーグを死ぬ思いで闘う。四段に昇段してプロ棋士になったおじさん棋士の清滝鋼介はC級1組降級阻止を目標にして死ぬ思いで闘う。そして竜王位を獲得した九頭竜八一は竜王位を獲得したあともなんか死ぬ思いで闘っていることが多い。

そもそも「りゅうおう」って誰？ってことで、「りゅうおうのおしごと」って何？っていう話なんですけど。これが重要なポイントなんですよ。

「りゅうおう」というのは、「困難な目標に挑戦する人」のことです。そして「りゅうおうのおしごと」というのは「困難な目標に挑戦する事」です。

「りゅうおう」は、「強い人」の象徴です。「強い人」というのは、困難に挑戦するということをしている・それが出来る人のことです。今は全然ダメな状態だとしても勝ちを諦めないで挑戦を続ける事ができる人のことです。そういう人のことをこの作品は「りゅうおう」と言っているんですよ(外形的に勝っている人のことではありません)。アニメ版のOP主題歌の「コレカラ」という曲の歌詞にそう書いてあります。この「コレカラ」という曲は、「りゅうおう」ならびに「りゅうおうのおしごと」の定義を述べているんです。だから九頭竜八一に限っていないんです。

<https://www.nicovideo.jp/watch/so32559385>

↑りゅうおうのおしごと！ 第一局「押しかけ弟子」 22分10秒から

<https://www.uta-net.com/song/243297/>

↑「コレカラ」歌詞。

そういう点で、この小説の登場人物は大体全員「りゅうおう」なんですよ(立場や境遇は違っていてもこの物語の

登場人物はだいたいみんな「困難な目標に挑戦する」ことをしている)。そして、「困難な目標に挑戦する」ことをするのならば私もあなたも「りゅうおう」なんですよ。

困難な目標に挑戦していたら、他人がバカにしてくるかもしれません(「そんなこと出来るわけがない」とか言って)。自分で自分を信じられなくなる事があるかもしれません。それでもその目標はあなたにとっての竜王位です。そのために挑戦を続けることはあなたにとっての竜王戦七番勝負を闘うことなんです。色々と辛くて苦しいこともあると思います。なぜならばそれは竜王戦七番勝負だからです。楽なはずがないでしょう。「コレカラ」という曲の歌詞に書いてありますが、「そんなこと出来るわけがない」ということを他人が言うてくるのはともかく、自分自身がそう思ってしまったら負けなんです。それとの闘いなんです。将棋の用語で言ったら、「詰まされるまでは自分からは投了しない」ということが出来るかです。だって「りゅうおうのおしごと!」の作中で、清滝桂香は女流棋士になることを諦めていないし、夜叉神天衣は女流タイトルを取ることを諦めていないし、清滝鋼介は名人になることを諦めていないんですよ。実際、本人が諦めない限り可能性はあるんですよ。だけど本人が諦めたらその瞬間に可能性がなくなるんですよ。だから自分で自分を信じる事が出来るかという勝負なんですよこれは。

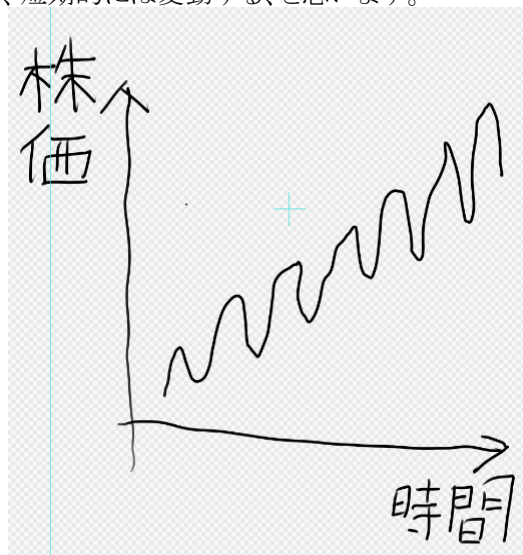
作中で主人公の九頭竜八一は多くの女性から好意を寄せられます。それは九頭竜八一が「困難な目標に挑戦する人」だからです。「困難な目標に挑戦する人」だから九頭竜八一は魅力的なんです。だから多くの女性が好意を寄せるんです(竜王のタイトルを持っているからではありません)。

それなので、「困難な目標に挑戦する人」はそれをしている時点でみんな「りゅうおう」なのだ、と主張している「りゅうおうのおしごと!」という小説は、外形的に勝っている人が偉いみたいな扱いをされることが多い現実の将棋界に対するアンチテーゼになっているんだと思います。

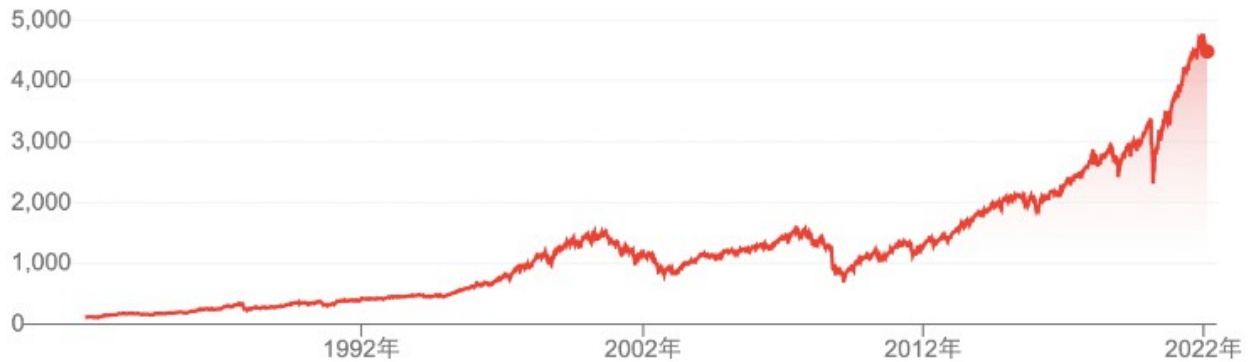
それで、アニメの女の子が可愛いという話についても同じで、アニメの女の子が可愛いのは「困難な目標に挑戦している」からだと個人的に思います。だからアニメの女の子は魅力的で可愛いんです。そして自分自身が「困難な目標に挑戦している」のならばそれは自分自身がアニメの可愛い女の子になっているんじゃないかと思っています。

作文 8: 株価の値動きを予想する話・「IDECO」と「つみたて NISA」が株式投資において理論的に最高の投資手法だと個人的に思う理由について書いてみる

株価は長期的には上昇するが、短期的には変動する、と思います。



↑ グラフで書けばこんな感じ。



↑実際の株価指標、S&P500の1980年から2022年までの推移。2000年頃に下がっているのはITバブル崩壊、2009年に下がっているのがリーマンショック、2020年3月にコロナショックによる株価の下落があります。

株価の長期的な上昇は以下の2つが原因だと思います。

- 1 経済成長した分が株価の上昇として反映される
- 2 法定通貨の貨幣価値が下落した分が見かけ上、株価の上昇として反映される

世界の経済成長率は一年間で3.5%-4%弱くらいだと予想されているんじゃないですかね(コロナの前まではIMFの予想はそのくらいだった気がする)。なので、そのくらいは株価は上昇すると思います。世界の株式に投資する投資信託みたいなやつを保有している場合についてですが。

法定通貨の貨幣価値が下落することで見かけ上株価が上がる話について。

世の中にある「お金(貨幣)」と「財(モノやサービス)」の分量について。

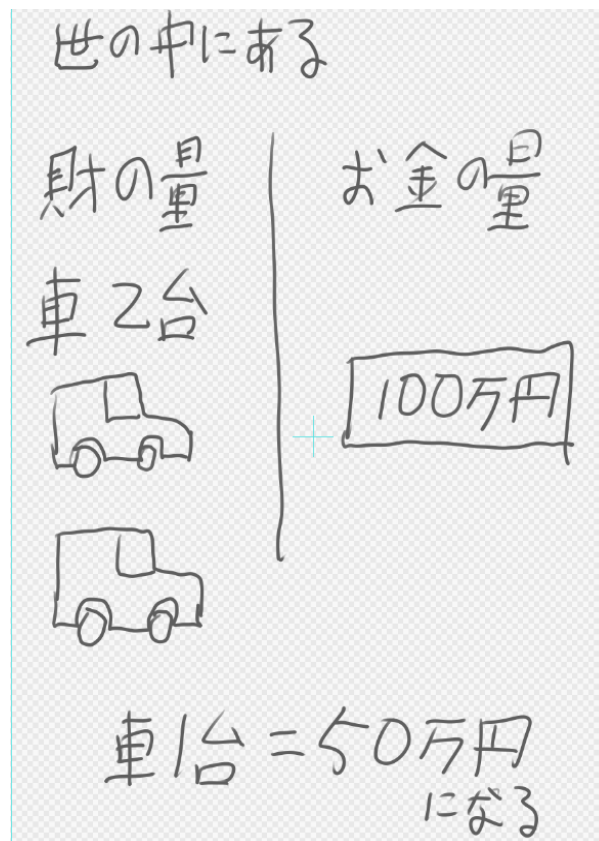
世の中にある「お金(貨幣)」と「財(モノやサービス)」の分量

100万円:クルマ1台



↑ だったらクルマ 1 台の値段は 100 万円です。

経済活動をしてクルマをもう一台生産して



世の中にある「お金(貨幣)」と「財(モノやサービス)」の分量 100 万円:クルマ 2 台
↑ だったらクルマ一台の値段は 50 万円になってしまいます。

だから、お金を追加供給して
世の中にある「お金(貨幣)」と「財(モノやサービス)」の分量
200 万円:クルマ 2 台

世の中にある	
財の量	お金の量
車2台	200万円
	+ 100万円
	100万円
	↑ 追加供給
車1台 = 100万円	になる

↑にして、これでクルマ1台の値段が100万円になります。

そんなふうに、経済活動をして世の中にある「財(モノやサービス)」の分量が増えたらそれと同じ分だけ「お金(貨幣)」を追加供給することで物の値段を安定させるというか、お金と財との交換のバランスを取っているのだと思います。

これがもし、お金の供給量が多すぎて、

世の中にある「お金(貨幣)」と「財(モノやサービス)」の分量

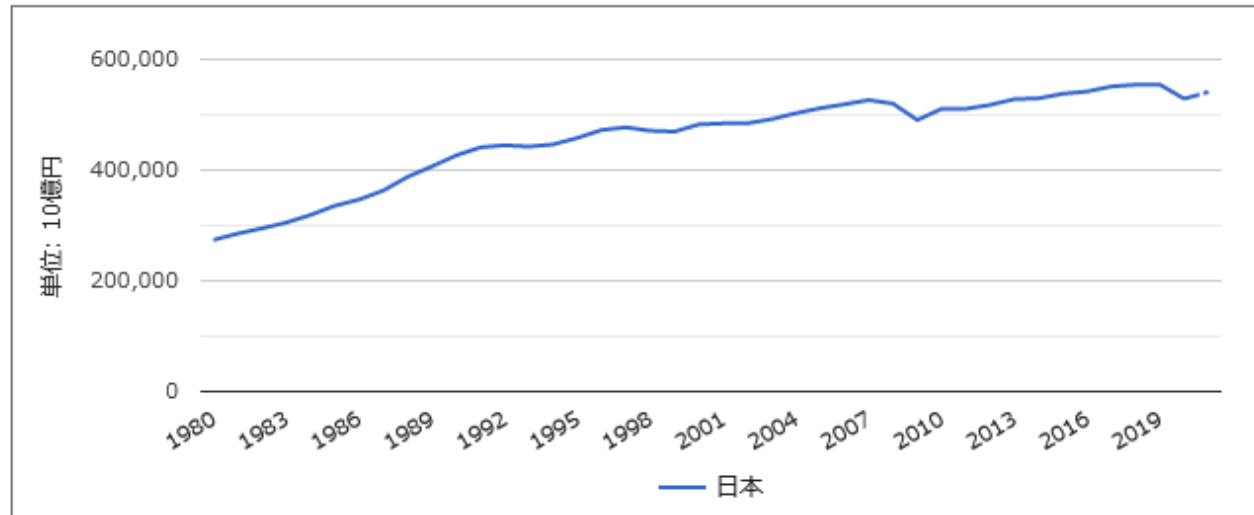
1000万円:クルマ2台

↑だったらクルマ1台が500万円になってしまいます。これはクルマの価値が上がったのではなくて、お金(貨幣)の価値が下がったんです。

それでは、日本円の通貨供給量(マネーサプライ)と、財の量として日本国の実質 GDP の推移を見てみると、

日本の実質GDP(自国通貨)の推移

※ 実質GDPは、当年の物価変動の影響を除いたものである。



※ [日本と比較](#) ※ [5年先の見通し](#) ※ [画像出力](#)

年	1980	1981	1982	1983	1984	1985	1986	1987	1988	1989
	273,862.70	285,390.60	294,844.00	305,231.20	318,973.00	335,666.00	346,832.10	363,239.50	387,885.40	406,729.00
年	1990	1991	1992	1993	1994	1995	1996	1997	1998	1999
	426,629.20	441,209.10	444,950.70	442,646.40	446,522.30	458,270.20	472,631.90	477,269.50	471,206.60	469,633.00
年	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009
	482,616.80	484,480.30	484,683.40	492,124.10	502,882.40	511,953.90	518,979.70	526,681.20	520,233.10	490,615.00
年	2010	2011	2012	2013	2014	2015	2016	2017	2018	2019
	510,720.00	510,841.60	517,864.30	528,248.10	529,812.70	538,081.30	542,137.50	551,220.00	554,300.40	554,381.70
年	2020	2021								
	528,960.50	541,433.13								

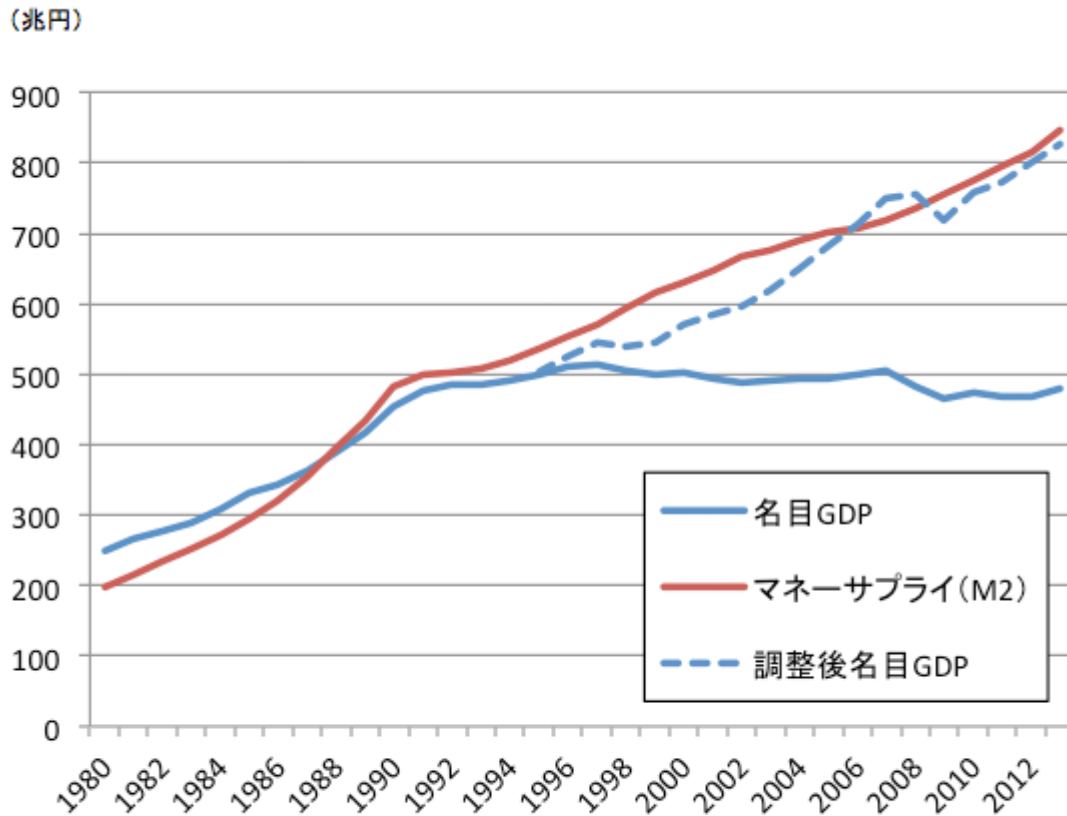
単位: 10億円

※ 数値 はIMFによる2021年10月時点の推計

※ SNA(国民経済計算マニュアル)に基づいたデータ

https://ecodb.net/country/JP/imf_gdp.html

図表1：日本：名目GDPとマネーサプライの推移



(注) 調整後名目GDPはアメリカのGDPデフレーターを使用して日本の名目GDPを修正したもの

(出所) 日本銀行 内閣府 総務省 BEA

https://www.rieti.go.jp/jp/columns/s15_0010.html

実質 GDP の上昇よりも、日本円の通貨供給量(マネーサプライ)の増え方のほうが大きいので、日本円の貨幣価値は下落していると思います。

「投資とかよくわからないで投資とかしていない」という日本人の大半は『日本円』という金融商品に資産の100%を投資している」という投資を行っているように見えますが、そもそも日本円という金融商品はそんなに良いものなのか？ ということがありまして。個人的に私は、「日本円」という金融商品は時間がたつと価値が下がることが予想される、長期保有には向かない金融商品だと思っています(日本国は日本円の通貨供給量・マネーサプライを増やすことで日本円の貨幣価値を恣意的に下げることが出来る)。そういう日本円という金融商品の特性をある程度考えた上で意識的に日本円に投資をしているというのであれば別に構いませんが、日本円という金融商品についてその性質が良くわからないままに資産の大半を日本円という金融商品に投資しているというのはそれは良い状態に思えないので、なんだかなーと個人的には思っているところです。

・株価の短期的な変動を予想する話、「ブラック・ショールズ方程式」の私なりの簡単な説明

<https://ja.wikipedia.org/wiki/ブラックショールズ方程式>

↑ウィキペディア(ブラック・ショールズ方程式)

株価の短期的な変動は、物理学における「ブラウン運動」と同じ現象だと考えるんだと思います。

<https://ja.wikipedia.org/wiki/ブラウン運動>

↑ウィキペディア(ブラウン運動)

水に花粉を浮かべると花粉が動くという話。水分子が動いていてそれが花粉にぶつかることで花粉が動くとい

う話。その花粉がどれくらい動くかを予想するという話。

数学的には熱伝導方程式(拡散方程式)で表されます。

$$\frac{\partial \rho}{\partial t} = D \frac{\partial^2 \rho}{\partial x^2}$$

↑ 熱伝導方程式(拡散方程式)の一般的な形。

熱伝導方程式(拡散方程式)というのは「時間の一回微分と位置の二回微分が等しい」というものだと思います。

その解の初期条件と境界条件みたいなを入力することで、株価の短期変動を予想することができるんじゃないかと思います。

取引されている S&P500 のオプション価格から逆算した、「VIX」というのがあってですね。これはボラティリティ・インデックスの略で、変動幅の大きさのことなんですが通称「恐怖指数」と呼ばれているもののなのですが、これがブラウン運動で言ったら「水の温度」に相当するものだと思います。

参考: 世界の株価と日経平均先物

<https://nikkei225jp.com/>

ちなみに日経平均株価のような株価というものはですね。お金が儲かるとか損をするとか言うことの他に「その時の社会の状態を数値で表現する」ということをしているのだと思います。コンピュータ将棋ソフトで言ったら「評価値」みたいな感じのモノだと思ってくれたら良いです。だから日経平均株価みたいな指標のモニタリングはその時点での社会の状態を端的に知ることができる(株価指標は社会についての評価値である)ので株式を保有していなくても有意義だと思います。日経平均株価(2022年3月16日時点で2万5千円くらいです)みたいなのが社会の状態をどういうふうに反映しているかということについては、コンピュータ将棋の評価値で言ったら100くらいの差ならば大きな差じゃないとか、1000も差がついたら大分差がついたとか、そういう感じで株価指数も見方が分かると大分役立つ指標だと思います(クラウド・オブ・ウィズダム、群衆の叡智によって未来予測をしているので)。もうちょっと具体的に言うと「テレビのニュースは新型コロナウイルスのオミクロン株が大流行して大変、みたいなことを言っているけど、日経平均株価的にはどうやら新型コロナウイルスのオミクロン株は大したことはない」と判断しているっぽいな」ということが分かったりします(株価があんまり下がっていないということは世界は平和だと判断されている)。

・ギャンブル必勝法の話

株式投資というのは言ってみれば「期待値がプラスのギャンブル」です。「期待値がプラスのギャンブル」のことを「投資」というのだと思います。逆の言い方をすると「ギャンブル」というのは期待値がゼロかマイナスの投資のことです。繰り返しになりますが株式投資の期待値は、

1 経済成長した分が株価の上昇として反映される

2 法定通貨の貨幣価値が下落した分が見かけ上、株価の上昇として反映される

以上の2つだと思います。

で、期待値分だけ取ることはそこまで難しくないと思います。

「IDECO」とか「つみたてNISA」で低コストのインデックス投信を積立投資(ドルコスト平均法)で買う。

以上のことをすると、期待値がプラスならば期待値分だけ利益を得ることができると思います。金融庁の回し者っぽくなってきましたが、考えてみるとこの方法が最高の方法だと現時点の私にはそう思えてしょうがないです。おそらくは株式投資とかについて詳しい人がよく考えて理論的に最高の方法を制度化したものが「IDECO」や「つみたてNISA」なのだと思います。

あと、一応書いておきますが投資信託は銀行で買ったらダメです(意味もなく手数料が高い)。個人的には楽天証券かSBI証券で買うのが良いと思うのですが、2022年2月時点で言ったら「つみたてNISA」については楽天のクレジットカードで楽天証券で投資信託を買うということをするすると購入金額の1%分の楽天ポイントがもらえるのでこれは良いと思うのですが。あと、個人的には、「S&P500」とか「先進国株式」とか「全世界株式」という感じの投資信託を買ったら良いと思います。「日経平均」とか「TOPIX」というのは積極的には買わないほうが良いと思います。あとよくわからないというのでしたら「イーマクス・スリム」のシリーズの投資信託を買ったら良いと思います。

世界の株価全体を買う投資信託を持っていたら良ければ年利7%くらい行くかもしれません。それは世界の経

済成長率が年 3.5%-4%弱くらいで、法定通貨の貨幣価値の下落分が年 3%くらいで、その合計分が株価の上昇として反映されると思うからです。

作文 9: マルクスの「資本論」を勉強して、金持ちになる方法を考えてみた

経済制度について考えるということは、「コンピュータ将棋ソフトの開発」とか「競技プログラミング」に近い部分があると私は思っています。経済制度というのを、プログラムによって作られているソフトウェアだと考えて、そのソフトウェアのより良さを競う、みたいな感じなので。だから「共産主義」というか経済制度について考えてみるということが面白いと思ってハマる人がどういうふうに面白がっているかと言うと、それは「コンピュータ将棋ソフトの開発」とか「競技プログラミング」と同じような面白さだと個人的には思います。だから私もより良い経済制度について考えるのはだいぶ面白いです。かなりパズル的な感じがします。

マルクスの「資本論」ですね。マルクスについては、作家の佐藤優さんが言っている「資本主義の強さを知るために『資本論』を読むべき」という話なのですけど。共産主義の提唱については余り良い出来ではなかったのかもしれませんが(コンピュータ将棋ソフトの開発でも良いと思って実装したアイデアが全然ダメだったということはよくあるのでそれ自体はそこまで変ではないと思うけど)、資本主義の批判的な分析についてその分析は現在でも妥当な部分があるんじゃないか、だから資本主義の強さを知るためにマルクスの分析を読む価値がある、というようなことを佐藤優さんが書いていて、私もそういう考えで「資本論」を勉強しています。

金持ちになる方法は、他人の「剰余価値(じょうよかち)」を取るという話です。

誰かに雇われて働いているというときの賃金は、「一時間に 1500 円分の価値を生み出して、時給 1000 円の賃金を受け取る」というときに 500 円を雇っている人に取り残されている、みたいな感じですよ多分。この取り残されている 500 円を「剰余価値」というのだと思うのですが(説明がだいぶ雑ですが)。

それで、金持ちになりたいというのだったら、「他人の剰余価値を自分が取る(搾取する側になる)」ということをしなないといけないのだと思います。

それは、

- ・起業する(他人を雇って働かせる)
- ・投資する(株式を保有する)

ということになると思うのですが。

「他人を雇って働かせる」ことをすると、「一時間に 1500 円分の価値を生み出す人に、時給 1000 円の賃金を払う」ということをすると 1 時間に 500 円自分の収入になるわけですよ。それを 100 人雇うと一時間に 5 万円収入が入ってくるわけですよ。そんな感じです。

投資(株式を保有する)について、株式を保有することによる利益を得る方法は、買った株が値上がりする(キャピタル・ゲイン)と、配当・分配金を受け取る(インカム・ゲイン)の 2 つがありますが、両方とも「剰余価値が形を変えたもの」なんだと思います。

逆に、「誰かに雇われて働いている」という状態だと金持ちになれないのだと思います。それは、「一時間に 1500 円分の価値を生み出して、時給 1000 円の賃金を受け取る(雇っている人に 500 円取られている)」という状態だからです。金持ちになりたいというのならば、「一時間に 1500 円分の価値を生み出す人に、時給 1000 円の賃金を払う(自分が 500 円取る)」ということになるべく多くの人を雇って大々的にやる必要があるんですよ(株式を保有することは、それを間接的にやっている)。

ハッキリ言うと、お金持ちになりたいのならば自分で働いたらダメですよ。「他人が働いてお金を稼いだそのお金がなぜかよくわからないけど私のものになる」という状態じゃないとお金持ちにはなれないんですよ。資本主義という経済制度の国で金持ちになった人は控えめに言ってほとんど全ての人がそういう方法で金持ちになっていると思います。そのことを心してください。

そんな感じの話です。

ちなみに、「資本論」について勉強をしていると、格差は広がるという話をしています。現在、GAFAM の 5 社の時価総額が日本の全部の会社の時価総額を上回っている件についてですが。「資本論」的には、資本主義という経済制度を続けていくと、最終的には一つの会社だけが生き残る(その一つの会社以外は全部競争に負けて淘汰される)、そして生き残った一つの会社が世界のすべての富を独占する、ということになるのだと思います。多分、機械化が進むと一人の人が機械を使うことで会社が運営されるので、最終的には一人の人が世界のすべ

ての富を独占して、その一人以外は全員すごい貧乏人、みたいな状態になるんじゃないかと思います。

だからマルクスの「資本論」的には「資本主義という経済制度はソフトウェアだと考えると、致命的なバグを抱えている。短期的にはうまく動いているように見えるかもしれないが、致命的なバグを抱えているので長期的にはこのシステムは必ずクラッシュする」ということを言っているんじゃないかと思います。別の言い方をすると「資本主義という経済制度のアルゴリズムを分析して、システムとして成立していないことを数学的に証明する」みたいなことをしているのが「資本論」じゃないかなと思っているんですけど。だからプログラミングの問題とすごく似ているのでプログラムが分かる人はだいぶ面白いと思います。

資本主義という経済制度のアルゴリズムを理解すると、そのシステムをハックすると言うか、バグを突いて合法的に自分が利益を得る方法を考えることが出来るようになると思います。だから「資本論」について勉強すると言うか、経済制度のアルゴリズムについて考えることは大事だと思っています。

マルクスや共産主義者の方々が「資本主義という経済制度は欠陥品だからやめたほうが良い」と言っていることについては気持ちはわかります。それは資本主義という経済制度は、社会の構成メンバー全員がお金持ちになることが原理的に不可能なシステムだからです。上に書いたように最終的には一人の人間が社会のすべての富を独占するようになると思います。マルクスは「あなたがその一人になれるとは思うな」と言っていたと思いますし、それは私もそうだと思いますが、幸いなことにそこまで行くにはまだしばらく時間がかかると思います。だから、あなたを含めて少数の人だけがお金持ちという状態はまだ実現できると思います。頑張ってください。

作文 10: 仏教が何を主張しているかをなるべく頑張ってわかりやすく説明してみる

仏教はもともとは宗教をする気がなかったんじゃないかと思います。釈迦(しゃか)が「生きていることが精神的に苦しい」と思ったので、その苦しみを無くす方法を研究した感じなので。だから仏教については「幸せになる方法を科学的に研究する学問」という定義があるくらいです。なので、仏教はジャンルの的には「心理学」です、多分。「宗教の定義は宗教学者の数だけある」と言ったときに、「仏教は宗教ではないのではないか？(あれは心理学)」ということが特に問題になるんですよ。仏教が宗教か宗教じゃないかという点は実は統一的な見解はない、感じなんですよ。

それで、私が仏教について勉強をしていて面白い・勉強になったと思った点として「プラスのものを積み重ねることでは幸せになれない」ということが挙げられます。私達そこへんの人間はよく「私が現在幸せでないのはプラスのものが少ないからで、プラスのものがたくさんあったら私は幸せになれるはずだ」みたいなことを思いがちです。具体的に言えば、「宝くじで6億円あたって、大金持ちになったら私は幸せになれるはずだ(私が現在幸せでないのはお金がないからだ)」とか、男性の場合だったら「年齢が若くて美しい女性と結婚したら幸せになれるはずだ(私が現在幸せでないのは年齢が若くて美しい女性のパートナーがいないからだ)」とか、コンピュータ将棋ソフトの開発者だったら「世界コンピュータ将棋選手権で優勝したら私は幸せになれるはずだ」とか、そんな感じのことを思いがちです。しかし、仏教は「そうではない」ということを言っています。そもそも釈迦は王子として生まれ育ったので、お金持ちで物質的な欲望を全部満たすことが出来る状態だったんですよ。にもかかわらず、釈迦は自分が幸せだと思えることができて生きていくことに対して精神的な苦しみを感じていた、みたいです。

有名なエピソードが「四門出遊(しもんしゅつゆう)」です。

> 四門出遊は、釈迦がまだ太子の時、王城の東西南北の四つの門から郊外に出掛け、それぞれの門の外で老人、病人、死者、修行者に会い、人生の苦しみを目のあたりに見て、苦諦に対する目を開き、出家を決意したという伝説。(Wikipedia より)

いくらお金持ちであったとしても、歳を取ることで病気になることで死んでしまうことから逃れられない、ということ。だからお金があるみたいなことでは幸せになれないんですよ。釈迦は自分が王子として生まれ育ったのでそのことに早い段階で気づくことができた。

そんなわけで、釈迦は王子としての立場を捨てて当時のバラモン教(現在のヒンドゥー教の元になった宗教)の修行者になった。それでしばらく修行・苦行をして、苦行は無意味だと思って菩提樹の下で瞑想をしているときに「悟り(さとり)」を得たんですよ。

釈迦の「悟り」の内容とはなにか？

端的に言うと「色即是空(しきそくぜくう:この世界の全ては空(くう)である)」ということだと思います、多分。このあたりから宗教的と言うか、形而上学的な話になってくるんですけど。

空(くう)って何？ ということを説明するとですね。端的に言うと、「空(くう)」は『存在している』と『存在していない』の中間形態だと思います、多分。これだけ言うとよく分からないと思うので、もうちょっと説明しますが、「鏡に映る人」とか、「テレビに写っている人」とか、「パソコンの液晶ディスプレイに写っている像」みたいな感じ。多分この文章を読んでいる人は液晶ディスプレイに表示されているテキストを読んでいると思うのですが、それは「存在

している」というほどはっきり存在しているわけではないじゃないですか(ディスプレイをオフにしたらぱっと消えてしまう)。だけど「何もない」訳でもないんですよ。そういう状態の有様を『存在している』と『存在していない』の中間形態・「空(くう)」というんですよ(多分)。言葉で説明するのがちょっと難しい概念なんですけど。空(くう)の例として「水面に写った月」とか言ったりする。はっきり存在しているわけではないけど何もないわけでもない、みたいな状態のこと。

わかりやすい例としては「寝ているときに見る『夢』みたいなもの」と言えば比較的わかりやすいかと個人的には思います。

「私とかあなたの本体はどこかで寝ていて、どこかで寝ている本体が見ている『夢』が、今のこの世界の私達」
↑って言ったら言いたいことを分かってはもらえませんか？ この言い方が私的には一番わかりやすい
仏教の説明だと自負しているのですが(この表現は厳密には正確ではないが大筋は外していないと思う)。私達は今現在「夢の世界にいる」ということ。私達が今いるこの世界は「本物の世界」ではなくて「仮の世界」であるということ。

私達は今現在そういう状態にある、ということ。仏教はそう主張しています。その主張を信じるか信じないかはともかく言っていることは理解してほしいです。そのことを理解すると苦しみが無くなる、みたいなことを仏教は言っているんですよ(多分)。

それで、私達が今現在「夢の中にいる」として、その夢が楽しい夢だというのならまだ相当にマシですが、残念ながら私達が今現在見ている夢は「悪夢」です。私達は今現在「見たくもない『悪夢』を無理やり見せられている」という状態にある。だから私達は这个世界でこんなに苦しいんだ、ということが仏教の主張じゃないかと個人的には思っているんですけど。

そのことを理解すると「ああ、这个世界は全部『夢』なんだー」と思うとちょっと这个世界で苦しくてもメタな視点で苦しんでいる自分を見れるんですよ。なんかホラー映画とか見たら怖いけど、所詮はニセモノじゃないですか。この世界というのはそれと同じ感じの怖さと言うか苦しみなんですよ。

もうちょっと続けます。

「私とかあなたの本体はどこかで寝ていて、どこかで寝ている本体が見ている『夢』が、今のこの世界の私達」
↑こう言ったときに「本体が寝ている場所はどこ？」っていうことが当然気になります。釈迦はそこまでは分からなかったのですが、後世の人達が色々と考察して、本体が寝ている場所のことを、仏教だと「極楽浄土」(キリスト教で言ったら「天国」)みたいな上位世界がどうやらあるっぽいということを主張しています。それで、私達は今現在「夢の世界にいる」「見たくもない『悪夢』を無理やり見せられている」という状態にあるわけですが、そういう状態でどういうふうに振る舞うかということによって、夢から覚めたときに「天国・極楽浄土」に行けるか「地獄」に落ちるかがテストされている、みたいなことを仏教は主張しています(この話はキリスト教と同じことを言っているように私には見えます)。

別の言い方をしてみます。

「ソードアート・オンライン(以下 SAO と略します)」という作品があります。もとは電撃文庫の小説でアニメ化されてアニメも結構人気の作品なんですけど。SAO という作品は、フルダイブ型の VR ゲームが登場して、主人公はそのフルダイブ型の VR ゲームの中に入って冒険をする話だと個人的には認識しているのですが。SAO 的に言うと、「現実世界だと思っている世界もフルダイブ型の VR ゲームの中」だと仏教は主張している、といえば良いですかね。私達が今いるこの世界は、フルダイブ型の VR ゲームです。私達は今現在 VR ゲームの中にいる。ただ、这个世界という VR ゲームのプレイヤーは全員記憶を消された状態でゲームをスタートしているから、这个世界が VR ゲームであることに気づいていない人が多い、という話。この話をしているのが、「マトリックス」という 1999 年の映画です。結構有名な映画です。「マトリックス」という作品も、私達が今いるこの世界は VR ゲームだという話をしています(「マトリックス」という映画は仏教の映像化と言われていたりする)。それで「プリンセスコネクト」というスマホのゲームがあって、アニメも作られたんですけど。「プリンセスコネクト」はおそらくは「マトリックス」をオマージュしていると思うので世界観は「マトリックス」的で、私達が今いるこの世界は VR ゲームだという、そういうストーリーと言うか世界観になっています。

「プリンセスコネクト」のアニメ版の 2 期が 2022 年の 1 月から 3 月まで放送・配信されていたのですが、その 3 話で「なかよし部」というグループの話をします。ユニ先輩というキャラがいて、彼女は「自分たちがいる世界がコンピュータゲームの中である」ということを証明しようとして研究をしている学者なのですが、彼女が最初に目が覚めたときに、「おはよう世界、今日も欺(あざむ)いているね」とつぶやいて話が始まるんですよ。3 話のラストにユニ先輩のモノローグが入るんですけど、それが

「やあ世界、今日も欺(あざむ)いているね
ぼくは世界の真理を曝かない
勇者でも英雄でもないぼくは
ただ知ってひとり
胸の中で愉悦(ゆえつ)に微笑むだけだ
しかしだ世界、お前の秘密にぼくは一瞬でも手をかけたんだぞ
その事実を
いずれ忘れてしまうかもしれないけれど
お前の年表には、ちっぽけな汚点として永劫残り続ける
ふっ…、ざまあみろだっ！」

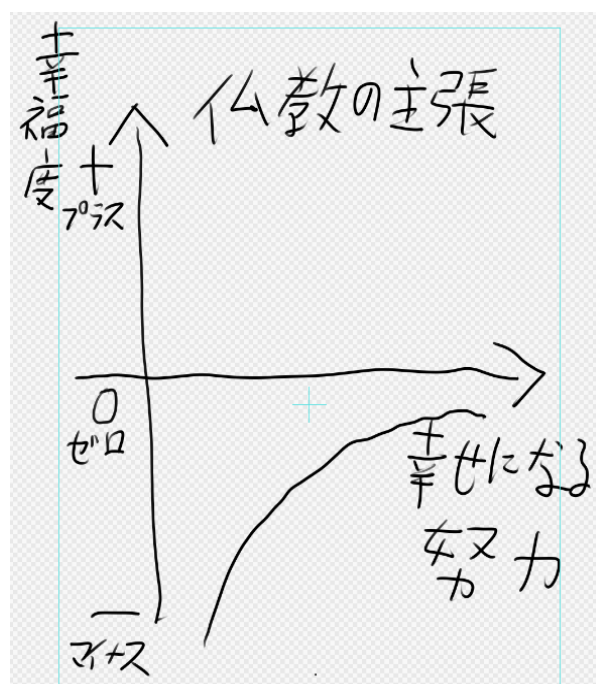
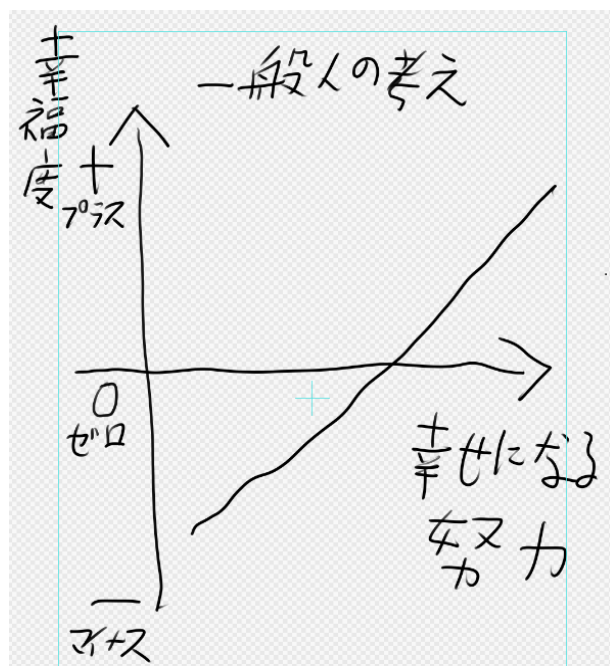
↑ こういうもので、個人的にすごい好きなんですよ。

参考 URL:「プリンセスコネクト！ Re:Dive」アニメ 2 期 第 3 話『不思議な森のティーパーティー～アフターヌーン
は危険な香り～』感想
<https://ponpokonwes.blog.jp/archives/87657686.html>
(<https://gno.blog.jp/izumi/45817823.html>) サイトが消えました。

宗教について勉強をしていて私が現在思っていることとして、救いを得る(夢から覚めたときに「天国・極楽浄土」に行く)条件として、以下のことが言えるんじゃないかと思っています。

1. 誰か・どこかにお金を払う必要はない(仏教で言ったら「布施(ふせ)」、寄付は推奨はされているけど義務ではない。ちなみに仏教における「布施(ふせ)」はお金を寄付することに限らず、笑顔を向ける・挨拶をする・褒める・良い情報を教える・席とか順番を譲る、みたいな相手にとって良いものを提供する行為全般を指すみたいです)。
2. 宗教団体に加入する必要はない(できれば加入したほうが良い・洗礼を受けたほうが良いみたいなことをキリスト教とか言っているけど加入は義務ではない)。
3. 外形的なことは何もしなくても良い(浄土真宗の念仏を唱えるとか、日蓮の教えの題目を唱えるとか、あと特別な服装や髪型にしないといけないとかっていうことはやりたくなければやらなくて良い)。
4. 死ぬまでに信じたら良くて、死ぬ 10 秒前に信じたとかでも十分に間に合う。
5. 条件を満たしたら全員救われる。救済に人数制限はない(自動車の教習所で運転免許証を取るみたいな感じです)。

幸せについて。仏教においてはこの世界は全部「苦」である、と言っているのかな。ここで言う「苦」というのは「思い通りにならない」、ということなのかな。このあたり私も十分に理解できているか疑わしいのですが、個人的な感想ですが、仏教的には「この世界には幸せはない、苦しいばかり。」で、最高で苦しさゼロ・幸福度ゼロの状態に持っていけるだけだと思うので、そういう苦しさや楽しさがゼロの状態に持っていこうということを仏教は主張しているんじゃないかと思うんですけど。



とにかく、「楽しいことや快樂」を仏教は評価しません。「楽しいことや快樂」はそれがなくなったり壊れたりするときに苦しみが発生するので、仏教的には「楽しいことや快樂」は「苦しみが発生源」であり、「可能ならば最初からないほうが良い」と考えるんじゃないかと個人的には思うんですけど（こころへんは我ながら理解が甘いポイントですが）。とにかく、苦しいことを避けたほうが良いのと同程度には「過度な快樂」は避けたほうが良いというのは心がけたほうが幸福度が増すんじゃないかと思います。

あと、「布施（ふせ）」の考え方に通じると思うんですけど、他人を幸せにすることをすると自分が幸せになると思います。自分が得をすることを考えるのではなくて、困っている人や苦しんでいる人を助けることを目的にして生きていると自分が幸福を感じやすいんじゃないですかね（そんなような話をするから「仏教は宗教じゃなくて心理学」とか言われるわけなんですけど）。

個人的な意見としては、コンピュータが影も形もなかったときに仏教のテキスト（経典）がコンピュータ・サイエンスの話をしているとしか思えないのを読んでみると、「私達が今いるこの世界は本当にコンピュータ・シミュレーションである」と思うほうが合理的だとは思っていますが、宗教については信じたくなければ信じなくても良いと思っています。それはこの世界は「苦しい状況下においてどれだけ他人に親切にすることができるのかを問われているゲーム」だと思うので、他人に親切にすることをしているのならばその人が宗教を信じていなかったとしても神はきっとその人のことを評価するに違いがないと個人的には思っているからです。

作文 11: Argo 開発物語

2019 年末のコミックマーケットにサークル参加して同人誌を配布しているときのことだった。

私のサークルにメイド服を着た美しい女性がやってきた。

高田悠奈さん。当時 25 歳。話を聞くと、都内の大学の大学院で理論物理学を専攻している博士後期課程の学生で、その傍ら、秋葉原のメイド喫茶でアルバイトをしているということだった。「随分変わった人だな」

私はそう思った。ただ、もともとコミックマーケットに来るような人には変わった人が多いので、そんなに驚くことはなかった。コミケに参加し始めた当初こそ、そこに来る人が変わった人ばかりであることに驚いていたが、最近は流石に慣れてきたので変わった人が現れても驚かないのである。

ツイッターのアカウントを交換したこともありたまに連絡を取り合っていた。私も基本的には理系なので考え方が似ていて気があったのです。

私が世界コンピュータ将棋選手権に参加していてその話を彼女にしたら、理系だけあって興味を持ったらしい。

「市村さん、もう少し詳しく教えてくれませんか？」

彼女がそう言うので、時々オンラインのビデオ通話をして将棋 AI についてレクチャーをしていた。

物理学専攻の大学院生であり、彼女はプログラムもある程度書ける。

アピール文には書かなかったですが、実は 2021 年 5 月の世界コンピュータ将棋選手権・WCSC31 のときも、彼女に少し「Argo」の開発を手伝ってもらっていました。

「やっぱり勝つのは難しいですね」

大会の後、彼女は残念そうに言った。WCSC31 において「Argo」の成績が二次予選・全敗で二次予選・最下位だったからである(一次予選はシードで免除)。

「まあ、参加者の皆様は情報系にすごく詳しくて努力されていますから」

「市村さんだって努力していますよ」

「私はそんなでもないですよ」

そんなふうに彼女は私のことを励ましてくれるのだった。

大会後の打ち上げも兼ねて、彼女と二人で秋葉原を散歩することになった。

私が驚いたこととして彼女は普段からメイド服を着ている。

あれはコミケのコスプレで普段は普通の服を着ているのだろうと思っていたところ、彼女はコスプレとかではなく普段着としてメイド服を着用している(正確にはメイド服ではないのかもしれないのだが、そうとしか見えない感じの服を着ている)。

「可愛い服が好きなので」

ということだった。

秋葉原ではアニメショップとかパソコンのパーツショップをはしごして見て回って歩いた。

彼女は理系の大学院生なのでパソコンのパーツにも詳しくて(使っているパソコンも自分で自作している)、メイド喫茶でアルバイトをしているけどそもそもアニメを見ることが好きな人なのです。そういう女性と秋葉原を散歩するのはなかなか楽しいものであった。

しばらく歩いていてだいぶ疲れてきた。

「市村さん、少し先にあるメイド喫茶に行きませんか？ これからサプライズ・イベントがあるので」

そう言って彼女はこれから始まるサプライズ・イベントがどういうものを私に教えてくれた。

そのメイド喫茶に行くとコーヒーを飲んでいたら、サプライズ・イベントが始まった。そのイベント内容は事前に彼女が教えてくれたものだった。

「ああ、例のやつですか」

「はい、例のやつです」

そう言って彼女はいたずらっぽく笑った。

「未来を見る」ことができる能力。

それが彼女の持つ特殊能力だった。

自分が望む未来が見えるわけではない。寝ているときに夢を見るように、ランダムに将来に起こる未来が見えてしまうことがあるという。

最初にその話を聞いたときに、私は何を言っているのかよくわからなかった。だが、彼女と関わっていてたしかに彼女が未来を予め知っているとは思えないことが何度か起きて本当にそうなのかもしれないと思うようになった。

「市村さん、10 年後の将棋ソフトに使われているコードを見ることができました」

ある日、彼女がそんな事を言ってきた。

「10 年後の将棋ソフトに使われているコード？」

「はい。せっかくですからそれを試してみるというのはどうでしょう？ 意外と強いかもしれないですよ」

彼女がそう言うので、私も試してみようかと思った。

それから、彼女が見たという「10 年後の将棋ソフトに使われているコード」を私の将棋ソフト「Argo」に実装する作業を行った。

彼女はある程度プログラムが書けるが、それでもまだ私のほうがコードを書く能力が高い。それなので、二人で同じ場所にいて作業をしたほうが良いだろうということで時々都内のコワーキング・スペースに出かけて一緒に作業をしていた。

普段着だというメイド服を着た美しい女性と一緒にソフトの開発するのはなかなか楽しいものだった。

しばらく作業をしていて「10年後の将棋ソフトに使われているコード」を実装した将棋ソフト「改良型 Argo」が完成した。

テスト対局をしてみると、びっくりすることにこれがすごく強い。「水匠5」と1000回ほど対局をさせると、「改良型 Argo」が997勝1敗2分、くらいの結果で圧倒的に勝つ。「dlshogi」と対局をさせてみてもやはり「改良型 Argo」は同じように99%以上の勝率で圧倒的に勝つのであった。「市村さん、やっぱり、未来のコードはすごいですね。このソフトを使ったら次の大会では優勝しちゃうんじゃないですか？」

彼女は楽しそうに言った。「改良型 Argo」を使えば、次の世界コンピュータ将棋選手権で「Argo」が優勝する事はほとんど間違いない。それは疑いようがない事実である。

だが、私はそれをする気にはならなかった。彼女が払う代償の問題である。

「歴史を歪めることには代償が伴う」

彼女はそう言っていた。

彼女が以前、未来が見えたというその情報を元にして自分が得をするように振る舞ったところ得たモノと同じくらいの不幸な出来事とその直後に起こる。それが何度か繰り返されたという。

それで彼女は、どうやら見えた未来の情報を自分が得をするために使ってはならないということを思うようになった。

もし私が、彼女が見た「10年後の将棋ソフトに使われているコード」を実装した「改良型 Argo」を使って大会で優勝をしてしまったら、恐らくはそれと同等の不幸が彼女に降りかかることだろう。私はそこまでして大会で優勝したいとは思わない。

それなので、「改良型 Argo」を使わないで、いつも通り従来型の「Argo」を使って大会に参加することにしようと思う。

そのことを彼女に伝えたところ、「それは残念ですね。市村さんが優勝するところを私は見てみたかったですけど」

彼女は寂しそうにそう言った。

補足： 大会では「Argo」のコンピュータの操作は高田悠奈さんに担当して頂く予定です。おそらく彼女はいつものように普段着だというメイド服を着てそれをするでしょう。初めてなのでうまくできない部分もあるかもしれませんが、よろしくお願いします。

作文 12: 市村さんのおしごと

会社をクビになってしまった。そうやって家でゴロゴロとしていたらツイッターからメッセージが届いた。

「君をスカウトしたい」

怪しい。そう思いましたが、私はいい感じに無職だったので次の仕事も探さないとはいけません。とりあえず話を聞いてみることにしました。

「我々は英国の女王からの密命で動いている」

麻生と名乗る女性はそんな事を言っていました。

「なんですか、それ？」

「英国は、日本国の動向に関心を寄せている。表立っては活動をしていないが、我々の組織を通じて英国は日本政府に対して働きかけをしている。ついては、人手が足りないので、リクルートをしたいと思っている」

そんなようなことでした。私はいい感じに無職だったのでとにかく職にありつければそれでよいだろうと思ってその話を受けてみることにしました。

「職場は御茶ノ水にある」

そういうことだったので JR 御茶ノ水駅から歩いて 15 分くらいのところにある雑居ビルの 3 階に行ってみました。

「英国コンサルティング株式会社」

入り口にはそう書いてありました。

「なんですか、これ？」

「表向きは、コンサルティング会社ということになっている」

「ふーん、そうなんですか。色々大変なんですね」

入ると年の若い女性を紹介されました。

「澄川美香、君の上司に当たる」
「市村です。よろしくお願いします」
「あんた、何？」

よくわからないが澄川さんは私のことをお気にめさなかったらしい。
「ちょっと、私は使えそうなのをよこせて言ったはずよ。なんでこんな見るからに使えなさそうなのが私の部下になるわけ？」

「澄川くん、残念だが、他に適当なのが見つからなかったんだ。今回はコレで我慢してほしい」

ひどい言われようだ。しかし、残念ながら言い返す言葉を私は持たない。

「しょうがないわ、今はコレで我慢してあげる。アンタ、名前は？」

さっき名乗ったんだけど。そう思ったが、しょうがないのでまた名乗ることにした。

「市村です。よろしくおねがいします」

「市村、アンタ、これまで何をやっていたの？」

「発電所を作る会社の品質保証部で画像処理のプログラムとか書いていました」

「使えなさそうね。仕事はできるの？」

「自分で言うのもなんですが、そもそも実用的な仕事は私には向いていないんです」

「私もそう思うわ。アンタに実用的な仕事ができる気がしないもの」

「はい。就職して最初に配属されたのは研究所だったんですが、配属されて半年は研究をしていたのですが(石炭と酸化鉄を混ぜて加熱して二酸化炭素が発生する速度を測定していた)、あまりに仕事ができなくて半年後に研究職から外されて『倉庫にある粗大ごみを捨てる仕事』に回されました。あのときは我ながらビビりました」

「お似合いの仕事ね」

「はい。見張りがいるわけでもないので倉庫においてあった古雑誌を読んで時間を潰していたら結構面白くて、『コレは良い感じの仕事だ』と自分でも思いました」

「良い感じね」

「そんなことをしていたら、そもそも研究所では使い物にならないって判断されて、それで品質保証部に移動になったんですよ」

「聞けば聞くほど使えなさそうね。市村、結局アンタは何ができるの？」

「『アニメを見てアニメの感想文を書く仕事』だったらできるんじゃないかと自分では思っているんですけど」

「残念なことにそれが仕事として成立するのはまだしばらく先のことになりそうよ」

「私もそう思います。本当に残念です」

「人手が足りないのは事実だから、しょうがないからアンタで我慢してあげるわ。私はアンタの上司なんだから私のことは『美香様』と呼びなさい」

そんな感じで、しばらくは試用期間ということで雇ってもらえることになった。

「国際的なスポーツイベントをなんとかして実施せよと英国女王から指令が来た」

麻生さんが言ってきた。

「あれ、やらないといけないんですか？」

夏に国際的なスポーツイベントが予定されていた。前年度からの感染症の世界的な流行を受けて一年間延期になっていたイベントである。

「日本政府は感染症の流行を受けてスポーツイベントの中止をしたいという動向らしい。だが、英国はなんと少しでも開催するようにと日本政府に対して要求をしている。ついては我々も開催するように日本政府に対して働きかけるようにとのことだ」

「面倒なのでやりたくないんですけど」

「市村、アンタはそうかも知れないけど、そもそも私達は英国女王の密命で動いている組織だということを忘れてはダメよ」

「美香様がそういうのでしたら、しょうがないですね」

「ついては君たち二人で首相官邸に赴き、総理に開催を依頼してほしい」

そういうわけで私は美香様とともに首相官邸に向かった。

「お越しいただき、ありがとうございます」

総理は美香様を見て言った。

「英国女王はスポーツイベントの開催を要求しています。日本政府としても毅然とした意思で開催に当たって欲しいと考えています」

「そうは言われましても、やはり、感染症の流行があり、現在の日本でスポーツイベントを開催することは難しいのではないかと」

「英国女王の意思です。そのためには日本国民に多少の犠牲が出ることはやむを得ません」

「この件については私だけで決めることはできず、やはり、都知事の同意を得ないことには開催することができません」

「都知事が合意すればよいのですか？」

「はい。それでしたらなんとか」

そういうわけで、今度は都庁にでかけて都知事に対して開催を依頼することにした。

その結果として近日行われる都議選において都知事の支持基盤を固めることが開催の必須条件であることが分かった。

都議会においては、都知事が主導する地域政党が第一党をしめている。それ故に都知事は影響力を発揮できていると言える。近日行われる都議選においてその地域政党が議席を大きく減らすことがあれば都知事の影響力の低下は必須であり、国際的なスポーツイベントの開催を押し切ることが難しくなることが予想された。

都議選において地域政党を勝利に導くこと。それがどうやら重要らしい。

「美香様、どうしましょうか？」

「悩ましいわね。4年前の選挙では大勝したけれども、今回は苦戦が報じられている。この状況では勝てない」

「やはり無理なんじゃないですか」

「仕事である以上諦めるわけにはいかないわよ、市村、アンタも考えなさい」

「ちなみに、どうすれば勝ちなんですかね？」

「やはり、鍵は一人区。ここをどう抑えるか」

都議選は中選挙区制で行われる。

その中で7つある一人区で勝利することが重要であることは論をまたない。

「とにかく一人区を抑える。市議を買収するわよ」

「美香様、そんなお金があるんですか？」

「心配しないで。英国から機密費が出るわ」

そういうわけで、我々は都議選の一人区の市の市議会議員全員に対して一人あたり10万円を献金し、市議に地域政党の候補を都議選で支持するようにと依頼することにした。

「美香様、数百人いる市議全員にお金を配って依頼するのはけっこう大変だと思うのですが、その作業は誰に依頼するんですか？」

「市村、アンタがやるに決まっているでしょ」

「え、私がやるんですか？」

「アンタ以外に誰がやるのよ」

そういうわけで、私は市議会議員に対して一人あたり10万円を献金という名目で振り込んで、そして電話・封書・FAX・電子メールの4つの方法で市議に都知事の主導する地域政党の候補を都議選で応援するようにという依頼をひたすら続けた。

そして都議選投票・開票日。

目的としていた一人区では7つあるうちの4つで支援する地域政党の候補が勝利することができた。その結果もあり、その地域政党は選挙前の苦戦を覆す善戦を見せ、都議会の第一党ではなくなったものの、都知事の影響力を維持するのに十分な議席を確保することができた。

「我々が行った献金と依頼は、果たしてどの程度選挙結果に影響を与えたんでしょうか？」

「それは良くはわからないわね。すごく影響を与えたかもしれないし、逆に全く関係がなくて我々が何もなかったとしても結果は変わらなかったかもしれない」

美香様は遠い目をして言った。

「ただ、英国女王の依頼に対してベストをつくすこと。それが我々の仕事よ」

「これでスポーツイベントも開催できるでしょうか」

「都議会を抑えた以上、おそらく大丈夫だと思う。やるべきことはやったと言って良いんじゃないかしら」

「市議に依頼をひたすらするのめけっこう大変でしたよ」

「市村、アンタも結構やるじゃない。見直したわ。少しは仕事ができるようね」

そうして、美香様は言葉を続けた。

「市村、アンタのことを気に入ったわ。正社員として雇うように、私から言っておくことにしてあげる。感謝しなさい」

こうして私の長かった試用期間は終わり、英国コンサルティング株式会社の正社員になることができた。

AobaZero のアピール文書

山下 宏

yss@bd.mbn.or.jp

1 AlphaZero の追試が最初の目的

AobaZero は Bonanza、LeelaZero のコードをベースに AlphaZero の追試をするべく MCTS + ディープラーニング で実装されてます。ネットワークは 3x3 のフィルタが 256 個の 20 block の ResNet でパラメータの個数は 2340 万個。棋譜生成をユーザの皆様と協力して行う分散強化学習です。オープンソースです*1。

2 AlphaZero の追試は 2021 年 4 月に終了

AlphaZero の将棋の追試は、2019 年 3 月から開始し、2021 年 4 月に 3900 万棋譜を作成して終了しました*2。その後、表 1 の変更を行っています。2022 年 3 月 30 日現在、5360 万棋譜を作成しています。

表 1 追試終了後の改良

日付	
2021 年 4 月	40 block に移行
2021 年 9 月	20 block に戻し温度を 1.0 から 1.3 に変えて序盤の変化を増やす
2021 年 12 月	Value の学習を「実際の勝敗」から「実際の勝敗」+「探索勝率」の平均に
2022 年 1 月	1 手 800payout 固定から 100~3200 と可変に
2022 年 2 月	ネットワークの構造を dlshogi 風に変更 利きの情報あり、ReLU を Swish に 30 手目までのランダム性を変更

3 40 block に移行

まずネットワークのサイズを 20 block から倍の 40 block に変更しました。囲碁の KataGo では +150 Elo ほどの効果があり、この程度の上昇を期待していたのですが実際は +40 Elo 程度でした。原因はいくつか考えられます。

- 学習が収束した状態、の棋譜から再学習したもので開始したせい。最初から 40 block だと違う？

- 学習が収束した状態で、30 手後のユニークな局面は 40% 程度 (100 万棋譜で)。同じような (相掛かりの) 棋譜ばかりで多様性がない。30 手後の細かい定跡の変化で強くなっている？
- そもそも将棋では 20 block 以上にしても効果が少ない？ 20 block で作った棋譜を学習させた 10 block は -120 ELO 弱い。
40 block +40 ELO 強い
20 block
10 block -120 ELO 弱い
- 40 block の再学習で学習率の下げ方が早すぎた (Cosine Annealing を 1 回だけ、の方が良かった？)。
- バグ

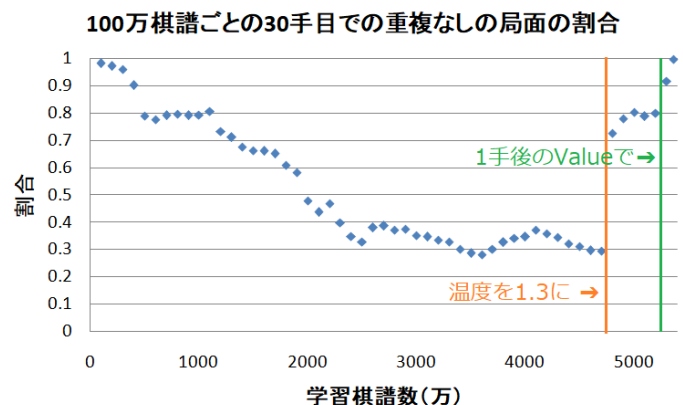


図 1 重複なしの (ユニークな) 局面の割合

4 重複局面を減らすために温度を 1.3 に

図 1 は学習棋譜の 30 手目で重複していない (ユニークな) 局面の割合です (100 万棋譜ごと)。学習開始時はほぼすべての棋譜がバラバラですが、徐々に同じ棋譜を生成するようになり、AlphaZero が学習を終了した 2400 万棋譜では 35% 程度まで下がります。具体的には初手から▲ 26 歩△ 84 歩の相掛かりの将棋ばかり指すようになります。重複を減らそうと 4700 万棋譜の時点で温度*3を 1.0 から 1.3 に変更して、初手から 30 手までは訪問回数が少ない手も選ばれやすいよ

*1 <https://github.com/kobanium/aobazero>

*2 <https://github.com/kobanium/aobazero/issues/54>

*3 温度 → 0 で訪問回数最大の手を、温度 1 で訪問回数の割合で、温度 → ∞ ですべての手を均等に選ぶ

*10 最善の Value との差を diff とすると、 $1/\exp(\text{diff} \cdot 70)$

30 手までの手順が決まれば、少なくともその手を 1 回は探索するようにして、800playout 後に強制的にその手を選んでます。

これらによって棋譜の重複なしの割合は 99.1% とほぼ完全にバラバラになってます。他にすべてのノードで 3 手詰を調べるようにしました。これは +20 ELO 程度の向上でした。そもそも 1 手詰の形や 3 手詰の形は簡単なようで、NN が「覚えて」しまっています。これらの変更でも現状、棋力に変化はなく、正しい方向なのかは不明です。

9 人間の知識は使っていない、をおそらく継続

利きの情報の追加や 3 手詰などで AlphaZero からは離れてきましたが、まだ全体としては「人間の知識は使っていない」を継続していると考えています。

10 1 手 1playout で将棋クエストで 6 段に

AobaZero の w3880^{*11}が 1playout で将棋クエストの :FuriJirouBot というアカウントで長考 (持ち時間 10 分) で 6 段 (2250 点) になっています。名前から推測できるように振飛車しか指さない設定です^{*12}。序盤はユーザ棋譜から作った定跡で振飛車を選ぶようになってるそうです。現在の AobaZero は振飛車を指さないのですが (初期の 400 万棋譜時点では後手のみ四間飛車を指していました) それでも学習棋譜では時々出てくるのでそれなりに指せるようです。

floodgate で 1 手 1playout は 2150 点 (w3392) でした。将棋クエストの長考、のレートはほぼ一致するようです。floodgate は対戦相手が偏っているのと勝率が低い (9 勝 83 敗、勝率 0.10)、アンカー (3300 点) から離れてる、でやや不確かですが。

余談ですが囲碁の KataGo も 1playout で囲碁クエストで :Katago1pBot で動いており下の段位になっています。

- 9 路で 2470 点 (7 段)
- 13 路で 2820 点 (9 段)
- 19 路で 2750 点 (9 段)

11 NN は内部で探索してる？

まったく探索なしで局面を NN に与えて返ってくる最善手を指すだけでこれだけの強さがあるのは驚きです。ただ探索してないとはいえ、NN の内部ではおそらく複雑な if 文の

組み合わせで疑似的な探索をしている (例えば、この形は角を切って同玉なら頭金で詰む形なので角を切る手の確率が高い、みたいな) ので、まったく読んでいない、というのはやや語弊があるのかもしれませんが。

12 詰を読むと水平線効果の無駄な王手をするように

また棚瀬さんから指摘されたのですが、3 手詰を読むようにして学習させた NN の重みは 1 手 1playout でも負けの局面で無意味な王手をするようになりました。これは探索部が負けの局面では王手以外の手を指さなくなり、NN もそれを学習したためです。きれいな形作りには投了直前は人間の棋譜からのみ学習させる、などが必要なのかもしれません。

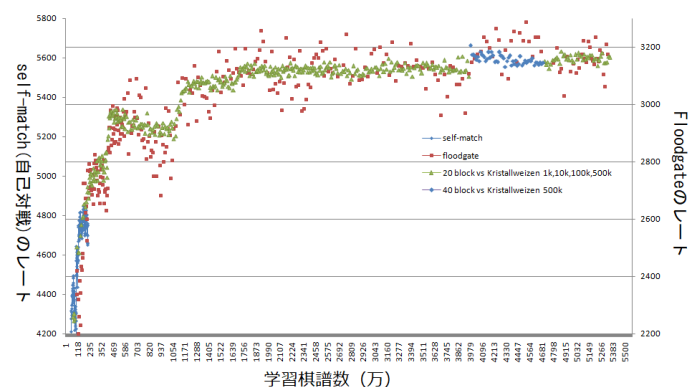


図3 AobaZero の棋力の推移。右軸が floodgate レート

13 で、強くなったの？

去年からほとんど強くなっていません。強くするのは大変ですね……。図3がELOの推移です。

14 3 年で 5300 万棋譜

5300 万棋譜、という膨大な棋譜を 3 年間で生成してきました。棋譜生成に協力していただいてる皆様に感謝いたします。

^{*11} w3880 で 3880 番目に作成された重み、を意味します。w3880 は利き情報を使ってない最後の重みでもあります。

^{*12} 将棋クエストはトライルール、AobaZero は宣言なので入玉になりにくいように振飛車に、とのことでした。

手抜きについて

手抜きチーム*

2022 年 3 月 31 日

手抜きは CSA プロトコルで対局を行うコンピュータ将棋プログラムです。開発者らが将棋プログラムの仕組みを理解するために作っています。

リポジトリ：<https://github.com/hikaen2/tenuki-d>

- 初段から二段くらいの棋力
- NNUE
- α β 探索
- D 言語

使用ライブラリ

- 『どうたぬき』(tanuki- 第 1 回世界将棋 AI 電竜戦バージョン) の評価関数ファイル nn.bin^{*1}

TODO

- 差分評価
- 千日手チェック
- 詰将棋
- ponder

* 鈴木太朗 (Twitter: @hikaen2), 玉川直樹 (Twitter: @Neakih_kick)

^{*1} <https://github.com/nodchip/tanuki-/releases/tag/tanuki-denryu1>

CGP アピール文章

2022/3/27
2022/5/3 修正
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
- ・ 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
- ・ AVX-512 命令をはじめとした拡張命令を活用
(ただし今回、AVX-512 命令非対応 GPU のマシンを使用するため、
AVX-512 命令部分は代替コードになっています)
AVX-512 対応 CPU を使用することにしたので有効化しています。
- ・ 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 を微修正したものです。

・ 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
Core i9 7940X で Ryzen9 3950X を上回るほどではありません。(1 スレッドあたり
では AVX-512 有効の Core i9 7940X のほうが上)

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。ただし持ち駒周りの評価を一般的な 3 駒関係より拡張
しております。

▪ 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、指し手の生成、オーダリング、構造体のコピーが主な使用箇所です。ただし構造体のコピーに関しては x64 のストリング命令が高速化されてきているので変更するかもしれません。

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 α β 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）

→バグをとったら効いたので入ってます(2018大会時点)

→なんか評価関数をいじったら明確に弱くなったどうしよう(20190301)

→なんか静止探索をいじったらまれに動かなくなったどうしよう(20190301)

- ・いわゆるボナンザメソッドを使ってます、もうちょっと改造できないかな

- ・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）

→怪しかったので試すのをやめました(20190301)

- ・なんと打ち歩詰めのバグが未だにあったので修正しました(20200330)

- ・なんと王手されている時に受ける手に未だにバグがあったので修正しました(20200330)

→めったに起きないので強さにあまり影響がない、というかむしろ探索が遅くなって弱くなりました。。。なぜ。。。けどさすがに外せない

- ・探索がもうちょっといい感じになる予定です(20200330)

- ・今年は正直何も手を入れられていないです・・・健康はいいぞ(20210331)

- ・2020年バージョンから探索を丸っと作り直し、評価関数も学習しなおしたら、ちょっとだけ強くなりました。

もっと強くなると思ったのになぜ同じくらいに収束するんだ・・・(20220329)

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。。。

毎年なんだかんだ起きるので、参考URLは増える一方です。

■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でして、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すが、
世界で初めて？パズルで博士を取ることになる先輩にそそのかされ（なぜか今同僚）、
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。
初日は全く寝ることができず、すごくつらい思いをしたことを覚えています。
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://jiquoo.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、今のとても強そうな名前になりました。
最近の事例を見る限り、読み方をつけておけばリジェクトにはならなかったほいで、ちょっと後悔をしています。

デビュー戦はなかなか熱かったです、白砂将棋さんと対局させていただきました。

あの負け方（王手千日手負け）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、早いうちに取りっ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/>>

■会場に行かずに近くで遊ぼう

有名なマクドナルド定跡(関東人の意地としてマクド定跡とはよばない)について、毎年試している割には全然勝ち星が増えないので、もしかすると効果がないのではないかと最近思い始めました。

まだ試行回数が足りないと思うので、今年も行きたいと思います。

また、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になっていたのも、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。

行った場所は下のほうにまとめておきました。

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」とご指導いただきました、ありがとうございました。

この話を最近（2018年夏ころ？）小谷研の後輩さんにお話ししたところ、「マジっすか、ご褒美じゃないですか！？」と言っていたので、今後ぼくの人生におけるご褒美イベントの一つとして大切にすることにします。

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。

2018年は永瀬先生のお父様のお店である川崎家に行き、ネギチャーシューラーメンを食べました、辛みがアクセントになって非常においしかったです。

家系ラーメンは大好きですし、きっと二日目は暇になるので（え、今年もぜひ行きたいと思います、もしよかったら皆さん行きましょう！

(20200330追記)

去年はラーメン食べている間に対局が始まってしまい会場入りが間に合わなかったのですが、きちんと対局がスタートしていました。

自動化されているって素晴らしいなと思っていました、もう会場いなくてずっと川崎家でラーメン食べててもいいですね。

あと今年は二次予選に進出しないと二日目に川崎家いけないので、是が非でも初日に川崎家に行っておきたいと思います。

(20220329追記)

地元のおいしいレストラン（とある弁護士事務所のそばだったりする）でランチを食べることを楽しみに対局してました。

だけど5/3は月曜日で確か空いてなかったのよな……。5/5に食べに行った記憶がある。

今年は川崎家だ……。行きたい……

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>>

マクドナルド 川崎ソリッドスクエア店

<<https://map.mcdonalds.co.jp/map/14707>>

川崎家 榎町店

<<https://tabelog.com/kanagawa/A1405/A140502/14013754/>>

停車場

<<https://tabelog.com/chiba/A1202/A120202/12000477/>>

■まさか自宅から大会に参加することになるとは・・・

2020年はコロナの影響もあり、まさかのリモートでの大会となりました。

割と暇な時間が長かったので、大会中にずっとゲームをやっていたのは内緒です。

下記のゲーム、渋滞を起こして街が死ぬってパターンが多い気がする。

シティーズスカイライン

<https://www.spike-chunsoft.co.jp/cities_skylines/>

■送りバントはいいぞ

自分のプログラム名について一応拾っておきますか。

2019年ですが下記の試合を観に行っていました。送りバントからのサヨナラ。送りバントはいいぞ。
ただ今年(2021年)、日テレ系プロ野球中継で、「イニング得点確率」を予測する人工知能があるのですが、バントしたら確率下がってたんですよ・・・送りバントはいいぞ。

【ハイライト】7/3 劇的なサヨナラ勝ちの巨人が今シーズン3度目の4連勝！【巨人対中日】
<<https://www.youtube.com/watch?v=eb9etHJK-2o>>

野球のイニング得点確率や作戦成功確率を予測するAIを開発日本テレビ系プロ野球中継で、「AIキャッチャー」に次ぐ進化系AI施策としてサービス提供が決定
<<https://www.datastadium.co.jp/news/6655>>

■目標

まったりゆうちゃんを倒して師匠超えしたいです。
最近では直対すると勝てるケースが多いのですが、順位で勝てないケースがたまにあるので、ぜひ勝ちたいですね。
あとできれば勝ち越ししてみたいですし、久しぶりに2次予選にも進出してみたいです・・・！
去年(2021年)は二日目暇になることを見越して、あらかじめ桃鉄やる約束を取ってしまっていました。
その時他のメンバーにボコられて悔しかったので感想戦をやったのですが、「1年目の5月のムーブが悪い」と言われました、なんて厳しいゲームだ。

桃太郎電鉄 ～昭和 平成 令和も定番！
<<https://www.konami.com/games/momotetsu/teiban/>>

一昨年は惜しかったんですよ・・・王手ラッシュして逃げられるっていう負け方しました。。。あの対局勝てれば上に上がれたらしい・・・。
同じ方に去年当たったのですが、レベルめっちゃ上がってて、手も足も出ませんでした・・・。

今年も去年同様に、CSA運営の皆さんも頭を抱えられたことと思います。

まだ予断を許さない状況でもあります。川崎での開催に向けて進めてくださっている運営の皆様には心より御礼申し上げます。

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。

まったりゆうちゃんのアピール文書 2022

前回からの新マシンはうまくいっていて、それですすめている（4 倍近くはやい）。少しの改良を加えようとしているが間に合わないかもしれない。

複数のシステムを動作させて、その間でデータをやり取りして並列動作をする方法を検討している。

以下は前回とほぼ同様の説明である。

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。しかしそうでは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。

実現できるかどうかかわからないが、全面的に書き換えることを考えている。今日からみれば、適切でないコーディングもある。それを直すことで、新しい並列方式が実現できると考えていて、少しとりかかっている。

開発者の年齢がもっとも高いのではないか。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。

きのあ将棋の最近の研究 2022 年 WCSC 向け v20220502

作成 2022/03/31 山田元氣

更新 2022/05/02 山田元氣

人間向けの将棋思考エンジンを用意する際に、「不安定な強さ」「曖昧な思考」というものをこれまで意識して作成していました。ですが「不安定な強さ」「曖昧な思考」は直感的なチューニングにとどまり、さらに一步踏み込んだ研究には着手してきませんでした。

それを反省し、現在はこのテーマを中心として研究を進めています。

おもなアプローチは 3 つです。

- ・「不安定な強さ」「確定的な強さ」の構築と検証。
- ・「不安定な強さ」「確定的な強さ」は、どちらの方が人に好まれるのか。
- ・「曖昧な思考」と「生物進化モデル&集団交配」を組み合わせることで思考の強化。

■「不安定な強さ」「確定的な強さ」の構築と検証。

まず「確定的な強さ」な強さのプログラム A、それに対しより探索量を増やしたプログラム B を用意します。このプログラム B を確率的に読みをせずに着手するようにしました。

確率が低いほど B が強くなり、確率が高いほどプログラム A が強くなります。確率を調整し勝率を同じにすることで「確定的な強さ」「不安定な強さ」の棋力が同じプログラムが出来上がるものと仮定します。

この手法で「不安定な強さ」「確定的な強さ」の思考エンジンを構築できるか検証のために、floodgate で同程度のレーティングとなるかを計測中です。**現在(2022/04/06 11:43 ごろ)は、レーティング差が 26 であり同程度の強さが実現でき仮説を肯定するものと考えます。このことは「レーティングのあり方」や「検証方法」に一石を投じる結果でもありと考えています。**

▼「不安定な強さ」の強さの将棋 AI R1462

http://wdoor.c.u-tokyo.ac.jp/shogi/view/show-player.cgi?event=LATEST&filter=floodgate&show_sel_f_play=1&user=qinoa-v20211120u20220329--yowaex

▼「確定的な強さ」の強さの将棋 AI R1488

http://wdoor.c.u-tokyo.ac.jp/shogi/view/show-player.cgi?event=LATEST&filter=floodgate&show_sel_f_play=1&user=qinoa-v20211120--yowa

前項でレーティングから同程度の棋力としてのシステムを構築に成功はしましたが、まだ「不安定な強さ」とは言えません。

そのため計算量をかえたプログラムとの対局によってそれを検証しました。

	不安定な棋力勝率	安定した棋力勝率
vs 評価ノード数 1000000の思考	1.5	0.0
vs 評価ノード数 100000の思考	13.8	2.7
vs 評価ノード数 10000の思考	48.3	34.0
vs 評価ノード数 1000の思考	75.4	87.8
vs 評価ノード数 100の思考	99.0	99.9
	不安定な棋力手数中央値	安定した棋力手数中央値
vs 評価ノード数 1000000の思考	102.0	96.0
vs 評価ノード数 100000の思考	113.0	105.0
vs 評価ノード数 10000の思考	109.0	120.0
vs 評価ノード数 1000の思考	96.0	106.0
vs 評価ノード数 100の思考	62.0	65.0
※対局数はいずれも1000局。		
※「安定した棋力勝率」vs「不安定な棋力勝率」は51.9% (1000局)		

「不安定な棋力」のプログラムは強いプログラムに勝つ確率が「安定した棋力」のプログラムよりも高いことが確認でき、同時に格下のプログラムにも「不安定な棋力」のほうが勝率が低いことが確認されます。

つまり勝てる相手にもそこそこ負け、勝てない相手にもそこそこ勝ちやすい状況を再現できました。

私は不安定な棋力＝不安定な強さのほうが、これまで人に好まれるのではないかと考えていました。それは幅広い棋力の人間にもは勝てるか負けるかわからずハラハラ感を演出でき、勝ちや負けが確定しづらいからです。また対局の中でシーソーゲームにもなりやすいとも考えられるからです。ですがこちらも今までは検証を実施してきませんでした。

■「不安定な強さ」「確定的な強さ」は、どちらの方が人間に好まれるのか

先に記した手法で、同程度の棋力の「不安定な強さ」「確定的な強さ」のプログラムを用意しました。これを自動生成された問題で、弱めの「不安定な強さ」「確定的な強さ」のセット、そしてやや人間には強めの「不安定な強さ」「確定的な強さ」のセットを用意しました。

2つのセットの強いほうか弱いほうかを人間は選べることができるものの「不安定な強さ」「確定的な強さ」かは確率的に決定されるシステムを作成しました。

このシステムの対局において、ユーザさんに対し good/bad を投稿していただく仕組みを同時に取り入れました。この人間に好まれるかを検証は 2022/03/31 にスタートしたばかりであり、データ蓄積の段階。ただし残念ながら不十分なデータなものの現在(2022/05/02 16:05 ごろ)、「確定的 good 率 84% (287 件)」「不安定な強さ 78% (297 件)」であり仮説を肯定する内容ではなさそうです。

■不安定な強さが人に好まれるかの追加実験

先の手法での「不安定な強さ」は極端なものであり、より穏やかな手法で「不安定さ」を実現しそれがきのあ将棋サイトユーザさんの good 率に影響を与えるかを 2022/04/12 より実験開始しました。

- ・ C：通常モード
- ・ D：ノイズを通樹モードに対し 10 倍加える。
- ・ E：うっかりモード 10%で発動。うっかりモードは計算量(ノード数)を $1/4$ にする。
- ・ F：ノイズ 10 倍。うっかりモードの両方を入れる。

今回はノイズやうっかりモードでの棋力の低下を調整するための作業は省略しました。これは棋力の低下が若干であり調整が困難だったためです。

※こちらを強さ強い/弱いそれぞれ作り計 8 パターンを準備し分析しました。

※「うっかりモード」については芝先生の意見を参考にした。芝先生の意見では計算量を $1/2$ 。

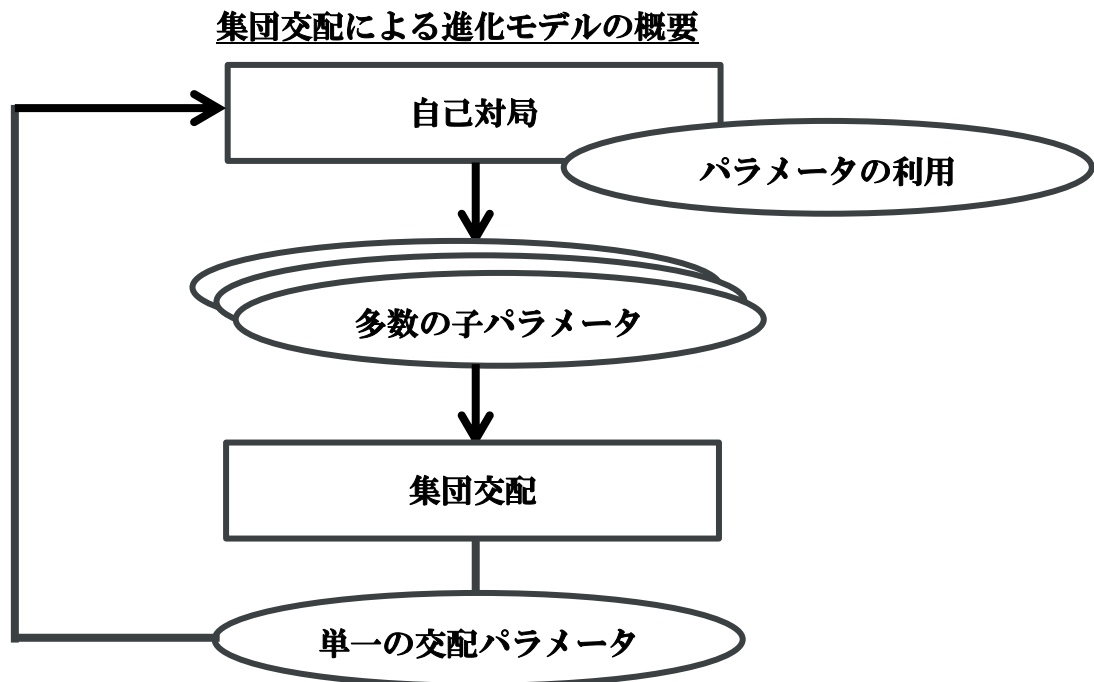
こちらの実験を集計しました表は下記のとおりです。

	good	bad	件数	good率			
うっかりなし、ノイズ通常	139	51	190	73%			
うっかりなし、ノイズ10倍	128	23	151	85%			
うっかり10%、ノイズ通常	142	50	192	74%			
うっかり10%、ノイズ10倍	147	37	184	80%			
※2022/05/02 16:05ごろ集計							
※うっかりが発動すると計算量1/4							
※ノイズ値は、-10000～ +10000が通常スケールの評価値に対して、ノイズ10倍の時に +100～ -100を 2回加味。							
※候補手評価もノイズを加えていて、ノイズ10倍の時には -20～+20を 2回加味。							
※ノイズは正規分布にするため、2重で加味。							

こちらはまだデータ不足なためお強さについてはあわせて集計しました。とはいえ**正規分布するノイズを大きく加味する方法が効果的な可能性が示唆されました**。なお正規分布しないノイズのほうがよかったのかどうかについては実験をしていないため不明です。

■ 「曖昧な思考」と「生物進化モデル&集団交配」を組み合わせることで思考の強化。

生物進化モデルはき残った(勝った)パラメータを優秀とみなし次世代を残す仕組みであり、集団交配は多数の子世代パラメータを、集団交配して次世代を算出する仕組みです。



この時の方法ではこれまで多くのパラメータを集計できなかった。また世代を重ねることにより思考が強化されるのは最初期のみで、世代を重ねるごとに弱くなる現象に苦しんでいました。

この現状を集団交配方法を見直すことで打破できそうな兆候があります。

そのため確率的 if 文というものを定義することで、適応範囲を広げたいと考えます。

// c 言語風に表現すると下記のイメージ

```

if(何らかの条件 && 乱数(0~1) < [パラメータ]) {
    処理
}
  
```

これによりほとんどの箇所のプログラムを曖昧に処理することが実現でき、合議的な手法やハッシュ保存と反復深化を組み合わせることで思考の強化することが出来るものと考えています。

これらの研究成果は、下記サービス運営に活用するものです。

きのあ将棋サイト <https://syougi.qinoa.com/ja/>

きのあ囲碁サイト <https://igo.qinoa.com/ja/>

以上

山田将棋について（2022 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

$\alpha\beta$ 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

こまあそび アピール文書

2022/03/31

探索部

- 基本アルゴリズムは $\alpha \beta$ 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数点を高くしている。
- 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数点を高くしている。
- 中盤、終盤は角と金の価値がほぼ同じにしている。
- 竜王、飛車の価値を高めを設定している。

[探索]

$\alpha \beta$ で9手程度読みます。

[評価関数]

数手で勝ち/負けという局面を静的に評価することを目指しています。

この「数手」を選手権までに可能な限り伸ばしていきたいと考えています。

[GUI]

強さとは全く関係ありませんが、例年のように画面はOpenGLによる3D画面です。

ponkotsu アピール文書

プログラム名

pokotsu

初参加

第 31 回コンピュータ将棋選手権

ソースコード行数

796行

採用している手法

- Value Network
- Policy Network
- マルチタスク学習
- MCTS

昨年12月に発売された「強い将棋ソフトの創り方」を読み進めながら学習部の改良を行いました.
今回はこれに加え, 振り飛車の棋譜で学習させて振り飛車を指させるようにする予定です

臥龍 WCSC32アピール文書

高田淳一

2022/4/23



プログラム概要

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

82勝127敗4分

開発者

高田淳一

おまけ情報

臥龍とは…摩訶大大将棋の駒の1つ



今回の特徴

従来の思考部を捨てて、ディープラーニングベースの評価関数を採用

いわゆるValue Networkで、1手読みを行う

開発コンセプト

評価関数のみでどこまで行けるかを追求する

盤面情報だけではなく、ドメイン知識を入力層に入れ、浅いネットワークで性能を出す



Value Networkの構成

入力層 38チャンネル

CNN 7層 + 全結合 2層

パラメータ数 約260万

学習用データ

floodgate 2021年棋譜のうち、レーティング3500以上のプレイヤーの評価値・局面

学習環境

AWS Sagemaker



プログラム構成

開発言語 Java, Python

ソースコード行数

Java 思考部(詰み探索) 7000行

Java UI部 10000行

Python 学習部 200行

Python 推論部 200行

Deep Learning フレームワーク

Tensorflow + Keras

Java UI部とPython推論部はソケット通信で接続

Java部はOS変更(macOS→Linux)したものの問題なく動作



参加マシン

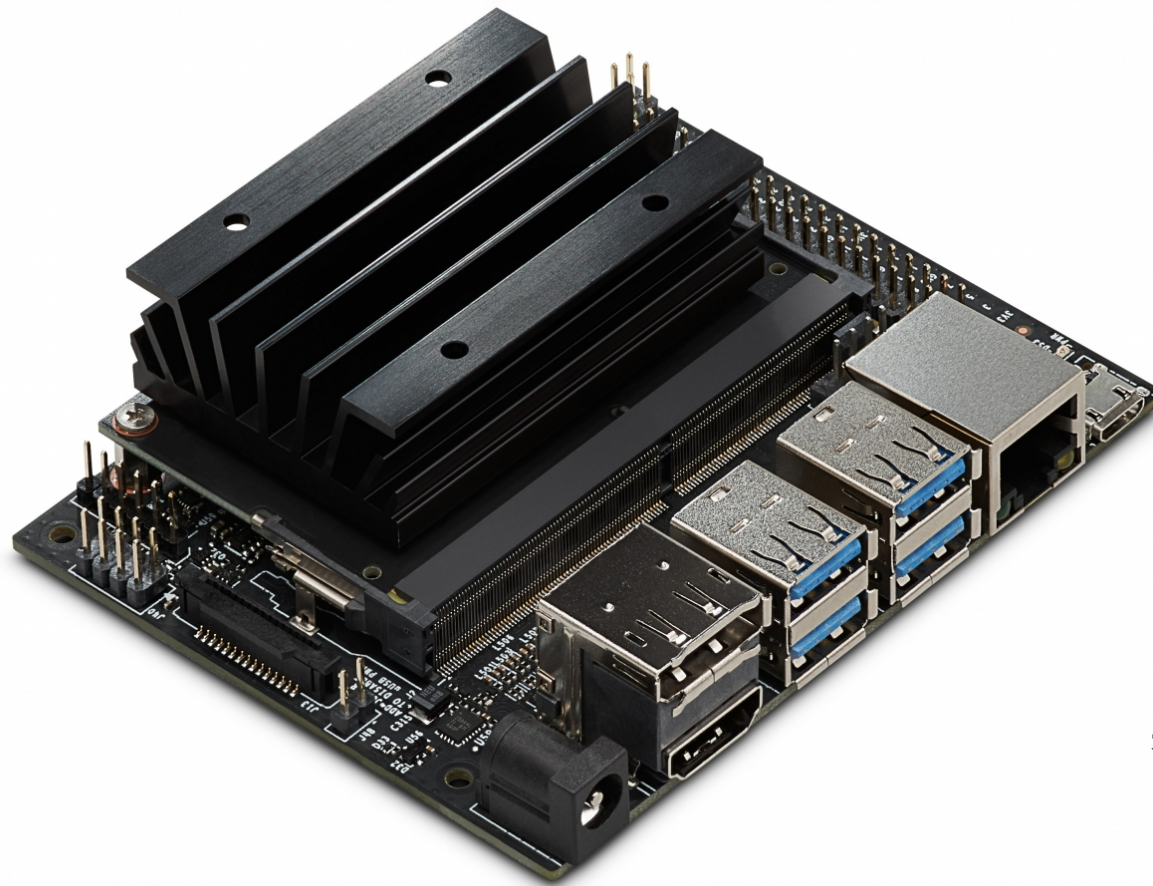
Jetson Nano

CPU Arm Cortex-A57 1.43GHz 4コア

GPU Maxwell 128CUDAコア

メモリ 4GB

OS Linux (Ubuntu 18.04)



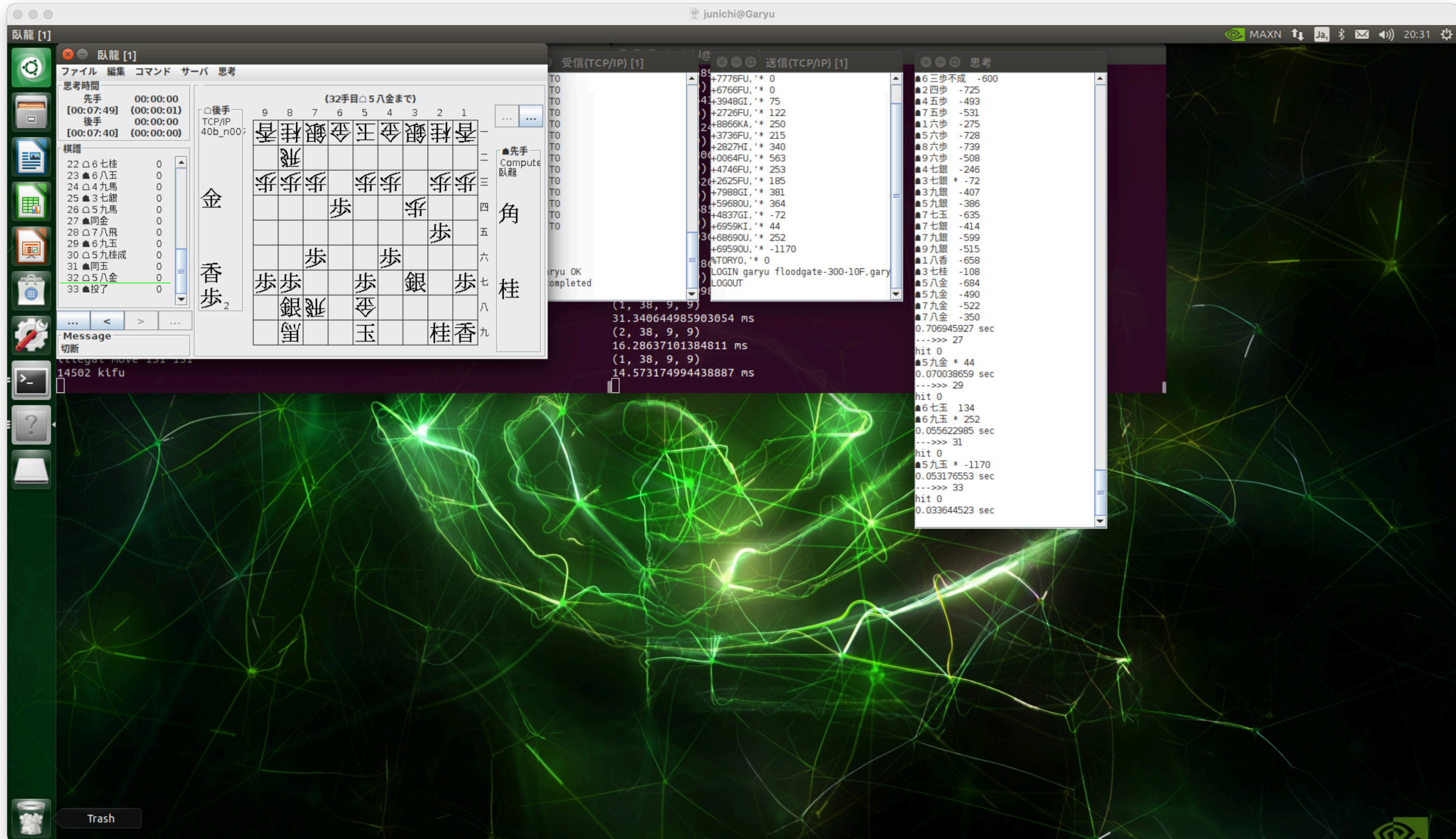
エッジデバイスとして注目の製品

WCSC30用に購入したものを今回も使用

参加マシンの中で最安値?(最近は入手難らしい)
約15,000円!!!

メモリ2GBの廉価版(約7,000円)もあるが、動作確認していない

画面イメージ



学習状況

入力チャンネル数を増やすことはできなかった

なので、残念ながら今回も非常に弱い! (泣)



参考文献

山岡忠夫著, 将棋AIで学ぶディープラーニング, マイナビ出版

山岡忠夫、加納邦彦著, 強い将棋ソフトの創りかた, マイナビ出版



使用言語:Python

使用ライブラリ:tensorflow(ディープラーニングフレームワーク), cshogi (学習データ作成時・対局時に利用 ※参考1)

プロ棋士の谷合廣紀氏のbert-mctsに触発され(※参考2)、深層学習における自然言語処理の枠組みの活用

に挑戦します。

2つのNN設計により、自然言語処理のテキスト分類を将棋に適用します。
NNへの入力に応じ重みを付けるself-attentionの仕組みに興味を持ち、transformerモデルを利用しました。 ※参考3,4

・ policy network

プロ棋士が対局中に棋譜用紙を見る行為から着想を得て、ニューラルネットワークへの入力は初手から現局面までの棋譜符号を用います。

プロ棋士は盤面だけではなく、棋譜からも対局の流れを読み取り、戦略を立てているのではという仮説・ロマンを立てました。

棋譜の指し手一つを自然言語処理での単語として扱い、棋譜を単語が時系列に並ぶ文章として扱います。

棋譜符号を数値化し、ベクトルの前方をゼロ埋めし、256で固定長としました。

分類問題の推論として、次の単語を示します。

・ value network

盤面の升目に入る駒を単語として扱い、1つの局面を95個の単語からなる文章として扱います。 ※参考2
升目の端から持ち駒まで、架空の時間軸に沿った文章といえるでしょうか。

回帰問題の推論として、出力は1つのスカラー値を示します。

・ 探索(?)

policy networkの出力の最大値が、設定値以上なら、即採用しvalue networkを用いず指し手を決定します。

設定値以下の場合、2手先の局面をvalue networkで評価し、指し手を決定します。

2つのネットワークを用いたmctsを用いる可能性もあります。

・ 最後に

まだまだ弱いですが楽しみます！

よろしくお願いします！

参考

1.<https://github.com/TadaoYamaoka/cshogi>

2.<https://github.com/nyoki-mtl/bert-mcts-youtube>

3.https://www.tensorflow.org/text/tutorials/nmt_with_attention

4.https://keras.io/examples/nlp/text_classification_with_transformer/

第32回世界コンピュータ将棋選手権

きふわらベ アピール文書

2022年04月3日 高橋智史

SETSUZOKU

リモート 接続 の用意をしましょう



「手前のノートPCから
後ろのメタルラックに
隠れているスレッドリッパを
遠隔操作している
様子だぜ」



「お父さんが 川崎の会場から
ノートPC越しに
新座の自宅のPCを触って
インターネット接続させれば
オンライン参加してるのと
同じというわけかだぜ」

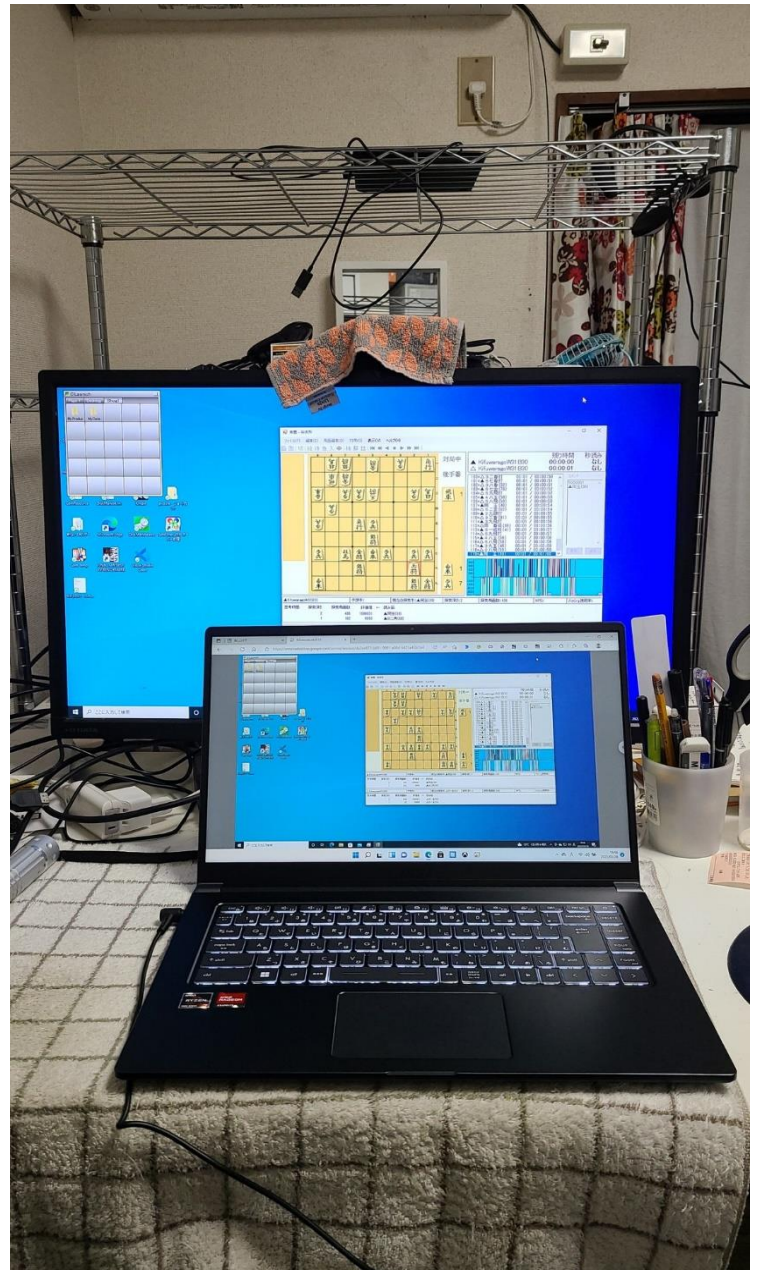
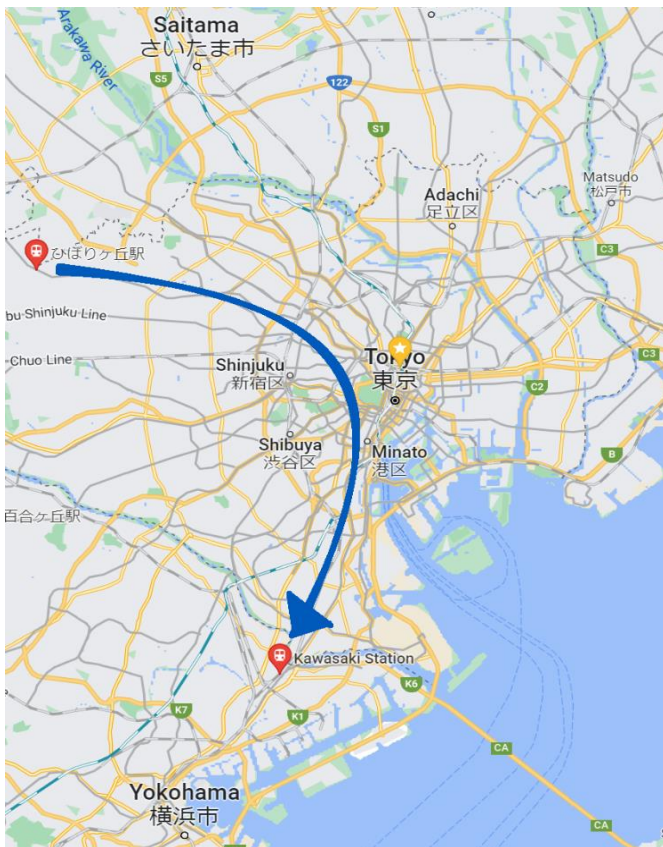


Photo : (C) Takahashi Satoshi / 2022

Map data : (C) Google / 2022



「家のスレッドリッパーの電源が
落ちたらどうなんの？」



「会場の天井でも
眺めてようぜ？」

MA

間に合わせ

DARE

誰 なの？



開発者
高橋 智史 （著作権上、問題ありません）

「以前に出た きふわらべ のうち フロムスクラッチのやつ
適当に選んで出ようぜ？」



コンピュータ将棋エンジン
きふわらべ （著作権上、問題ありません）

「どれを選んでも 非合法手を指す」



ひよ子 （著作権上、問題ありません）

「じゃあ PR文書の 残り23ページを らくがきページにしましょう」

フロム・スクラッチ宣言

思考部に大きな影響を与える、他者の作成したプログラム・
データ等を利用していません。

フリーフォント「たぬき油性マジック」 作者：たぬき侍 （利用ライセンス上、問題ありません）

<https://tanukifont.com/tanuki-permanent-marker/>

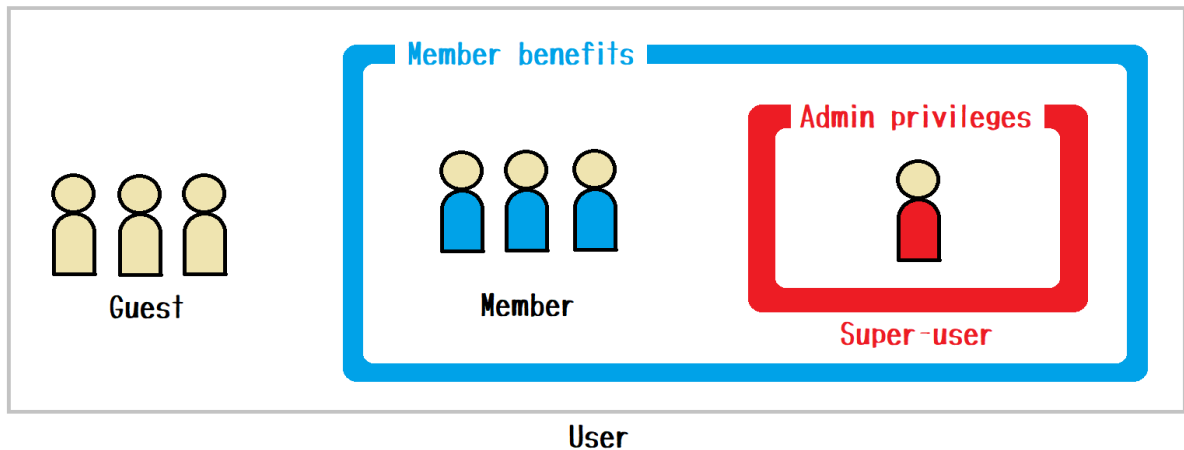
将棋サイトを 知 りましょう

TAIKYOKU

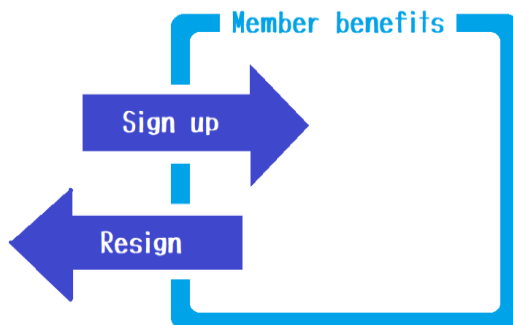
将棋 対局 サーバー って何なんだぜ？



「その前に 平凡なつくりの 会員制サーバーの説明をしよう」



「すべての User（ユーザー）は Guest（ゲスト）だぜ。
そのうち 会員特典 をもらえるのが Member（メンバー）、
サーバーを止める力もあるのが Super-user（スーパーユーザー）だぜ」



E-mail address

hoge@example.com

Password

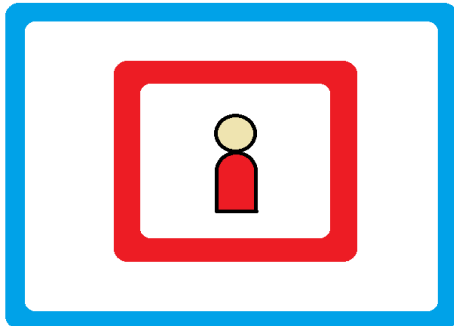
サインイン



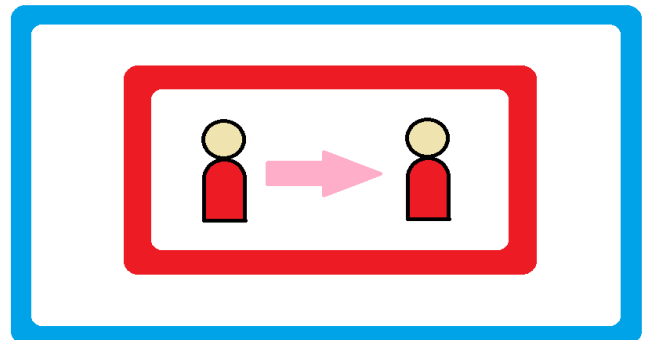
「Guest から Member に『成る』のが Sign up（サインアップ）だぜ。
最低限必要なのが その人しか知らない Password（パスワード）で、
E-mail address（イーメール アドレス）は Password を忘れた人に
『パスワードを変えるページ』に案内するために使う」



「『パスワードを他人に教える』という **行為が存在しなくなる** わけだな」



The **first one**
has existed since
the birth of the system



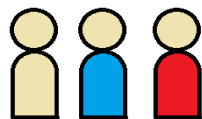
After the second person



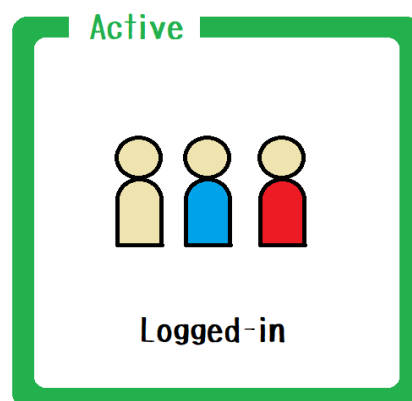
「1人目の Super-user に『成る』ことはできなくて、
最初から居て、すぐにパスワードを掛けるのがあるわよね。
『外部からSuper-userを作る』という **行為が存在しなくなる** わけね」



「2人目の Super-user に『成る』には
1人目の Super-user が管理画面に入って
2人目を『成らせる』だけだな」



Logged-out



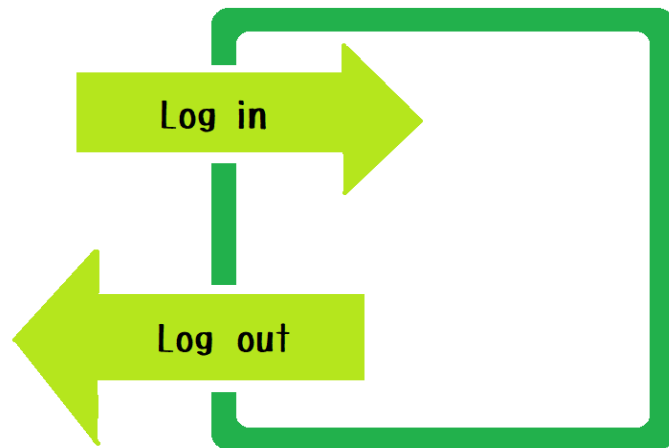
「また別の見方では Active (アクティブ) か どうか
というのもあるな」



「本人が触ってないのに、その人がゲームしてるように見えたら乗っ取られているんだぜ。 アクティブ だぜ」



「犯罪に巻き込まれているかも知れないから
アクティブな間ずっと 操作ログ を残しておくものなのよ」



「アクティブに『成る』のが Log in (ログイン)、
アクティブを止めるのが Log out (ログアウト) だぜ」



「事後にがんばってログ見るより、パスワードが命綱だぜ」



「サインインと ログインは似ているが、
ログイン時には
『パスワードを忘れた』リンクがないと不便だな」

E-mail address

hoge@example.com

Password

パスワードを忘れた

ログイン

パスワード変更の案内

3 分前

次のリンクをクリックして
パスワードを変更してください。

<http://hoge.example.com/wasureta/aabbccdd>

Password

パスワードを変更する



「floodgate (shogi-server) には サインアップの仕組みはなく
ログインだけ。全員 Guest。
NNGS や lishogi にはサインアップの仕組みがある」



「サインアップって どんな必要性があって するの？」



「Account (アカウント；口座) の開設だぜ。
例えば 会費 を取ってるとき
ログアウトしても 会費の支払いを中止しないが、
リザインしたら 会費の支払いは中止だぜ。
『EC-Shop 決済』でググれば技術的な話も出てくるだろう」



「floodgate は無料だし、CSA も WCSC も郵便振込だし、
Account は要らないのよ」



「免責事項や マナーなど 会が求めるものを 意識 させるのも
サインアップのタイミングだぜ。
ログインというのは ユーザー名とパスワードでするって入ってくる。
ほんとに サインアップは 要らないのかだぜ？」



「要らないなあ」



「Account を作るにしろ Guest でログインするにしろ、
次にやることがある。 権限の付与 だぜ」

フォルダー より グループ の方が 知 られてるぜ

KENGEN

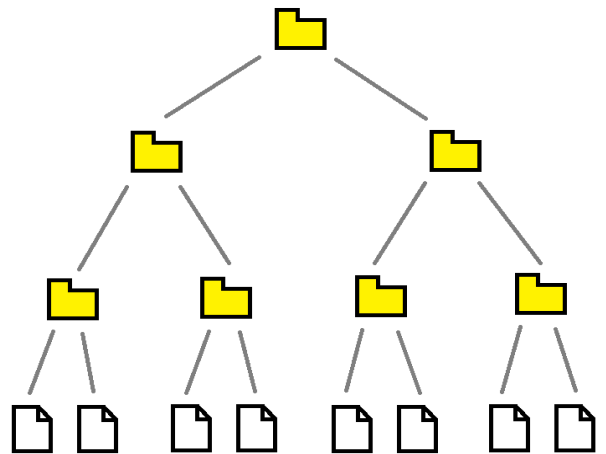
権限 って何なのよ？



「まず 2つの概念を
説明しよう」



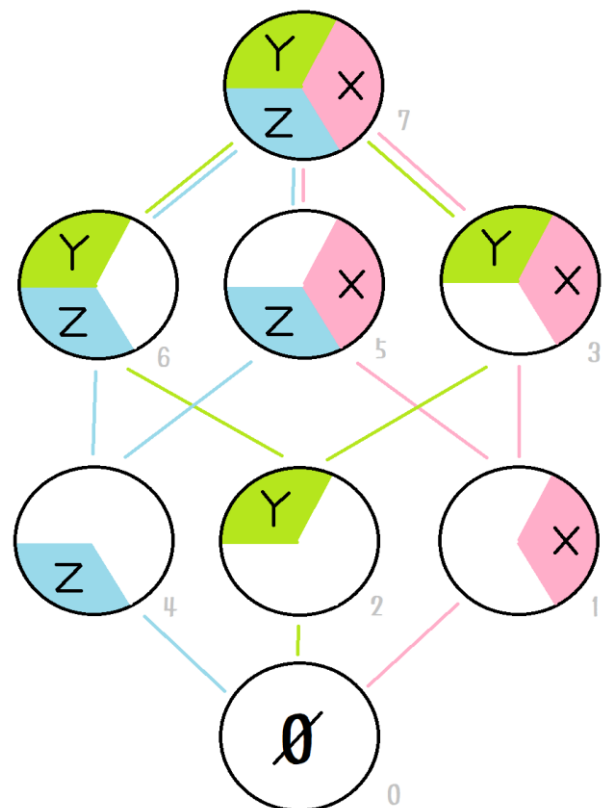
「☞ フォルダ－というのは
ツリー構造だぜ」



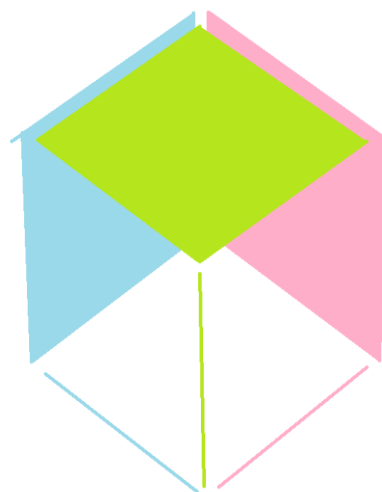
「☞ グループというのは
べき集合の各元だぜ」



「何よこのセフィロツリー
みたいな図は。
どうやって見んのよ？」



「☞ こうだぜ」



「XYZが何かって
聞いてんのよ」



「☞ 『無し』から『すべて』
まで一通りの表し方を
できるのが グループ
だぜ」



「当たり前のことを
もったいぶって名前が
付いてるのが数学よね」



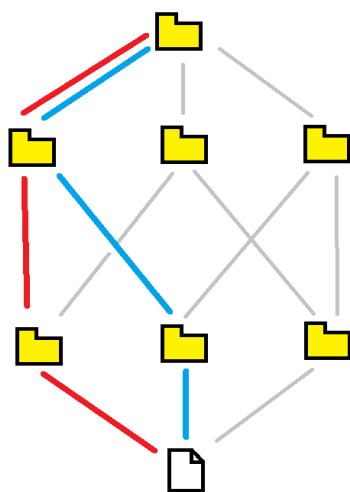
「将棋向きの 数学を創ろうと
して失敗したのがお父ん
だからな」



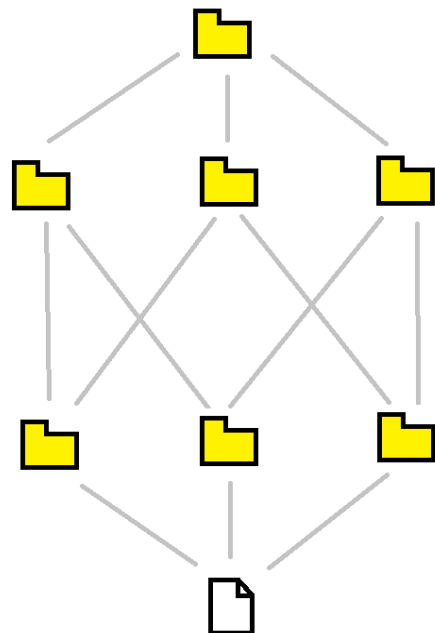
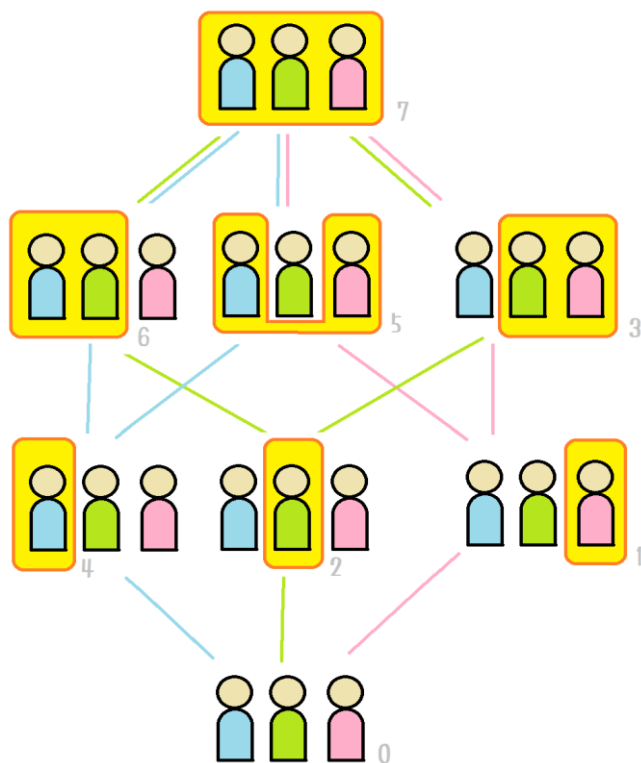
「構造を覚えたら いろんなものに 当てはめてほしい」



「☞ もし パソコン のフォルダー構造が
ツリーではなく ベキ集合だったら
どうなるか 想像してほしい」

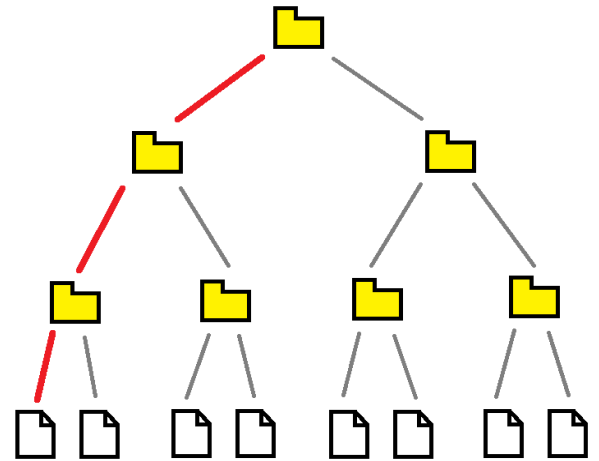


「☞ 経路（パス：Path）が
2通りできてしまったな。
住所が一意に決まらなくて
同じファイル何回も
数えてしまうかも」





「ツリー構造は Path が一意に決まるというのが利点だぜ」



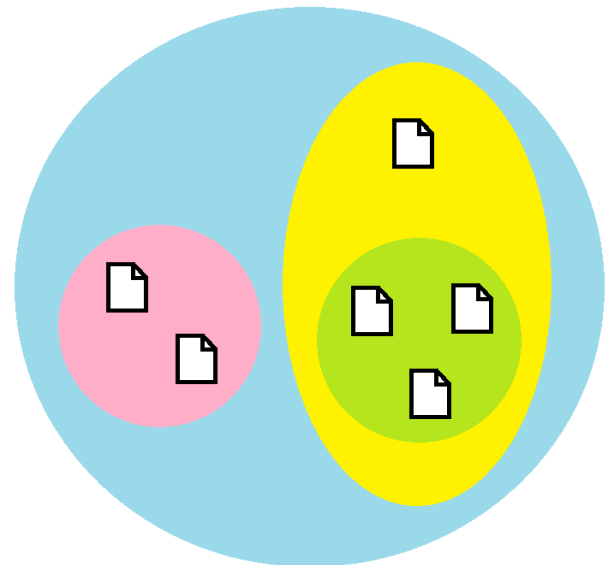
「重複して数えるという行為が存在しなくなるから全検索するのに早そうだな」



「当たり前の概念に名前が付いてるから 疑問も 言語化できるし、ネットで検索して 調べることができるんだぜ！」



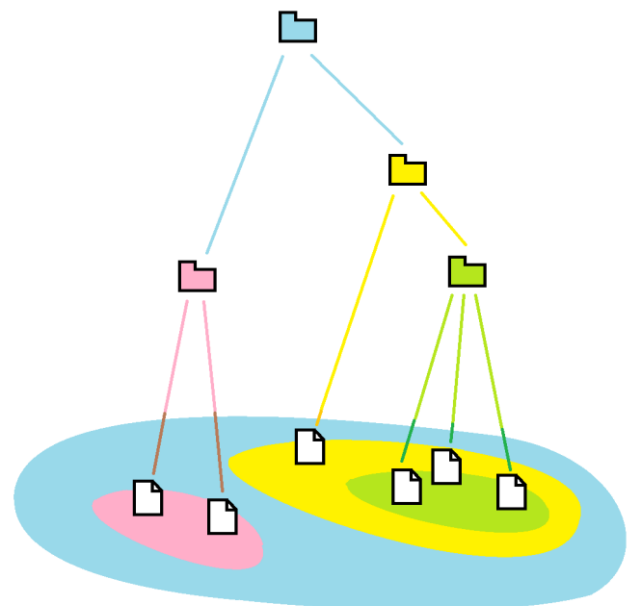
「例えば ツリー構造は上から見ればグループでもある。それが右図」



「ツリー構造を上から見るって何なんだぜ？」

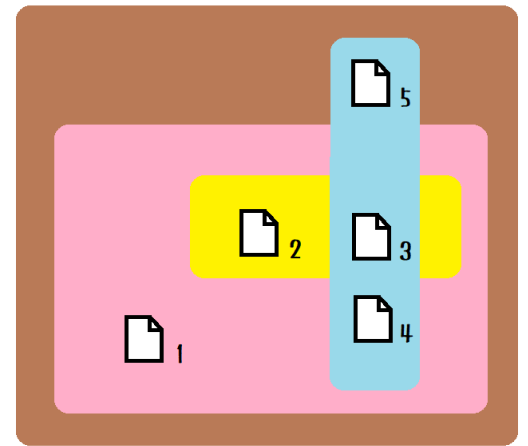


「グループ構造を地図とするなら、ツリー構造は住所よね」





「☞ じゃあ どんなグループは
どんなツリー構造になるのか
気になるだろ。
積み重なって はみ出ているグループを
ツリー構造にしてみてください」



「グループなんだか ツリーなんだか
まったく」



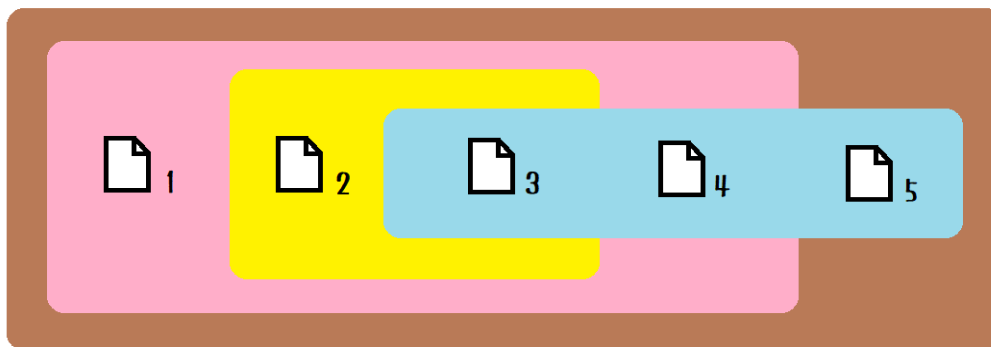
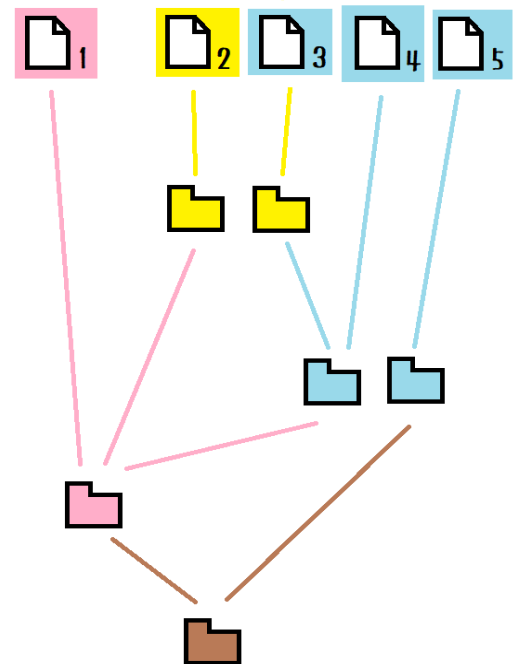
「☞ これは 上下逆さ の木構造に
した方が 感覚に合いますかねえ」



「面白いだろ。
そして 1, 2, 3, 4, 5 の並びを見て
気づくことがあるだろ」



「☞ この図は 1列に 並ぶんだぜ」



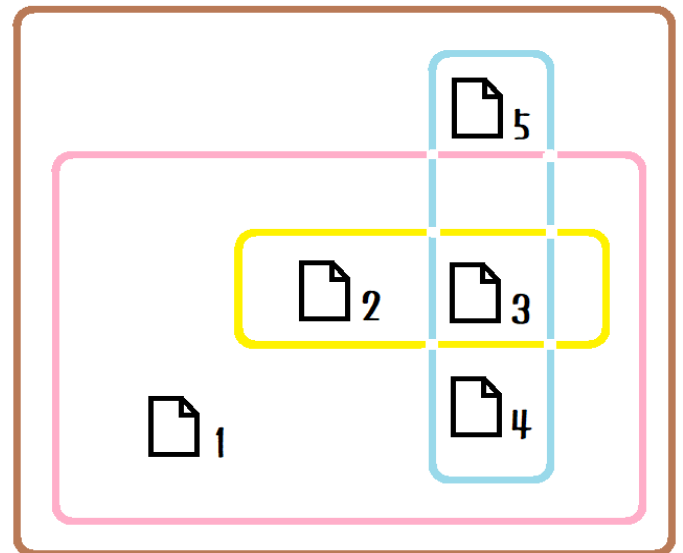
「そんなにヒマしてんなら 将棋エンジン 作ればいいのに……」



「☞ グループの図から
塗りつぶし を止めたぜ。」

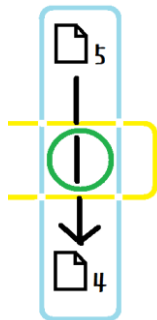
あれっ！ 前景、後景の区別が
付かなくなったな！

□3 は
青の中に黄色があるのか、
黄色の中に青があるのか、
どちらとも言えるぜ！



横から見るのが難しくなったな！」

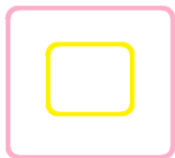
「全順序じゃなくした んだな」



「☞ 青グループの □5 と □4
は 黄色グループによって
分断されてるの？ それとも
つながってんの？ それとも
場合によるの？」



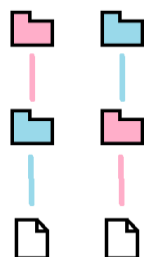
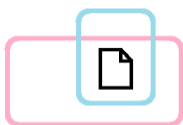
「道をさえぎるとか
同じグループを
2度カウントする
のは無いぜ」



「☞ 赤色グループに
黄色グループは
すっぽり含まれているから
赤、黄という順は
逆にはできないの？」



「そうだぜ」

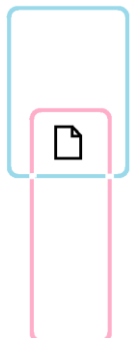
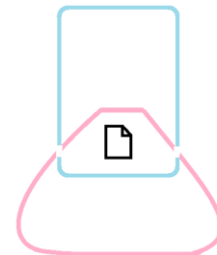
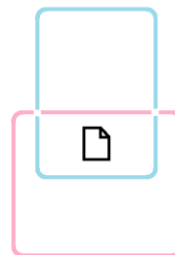
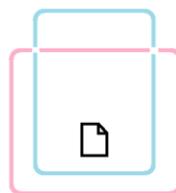


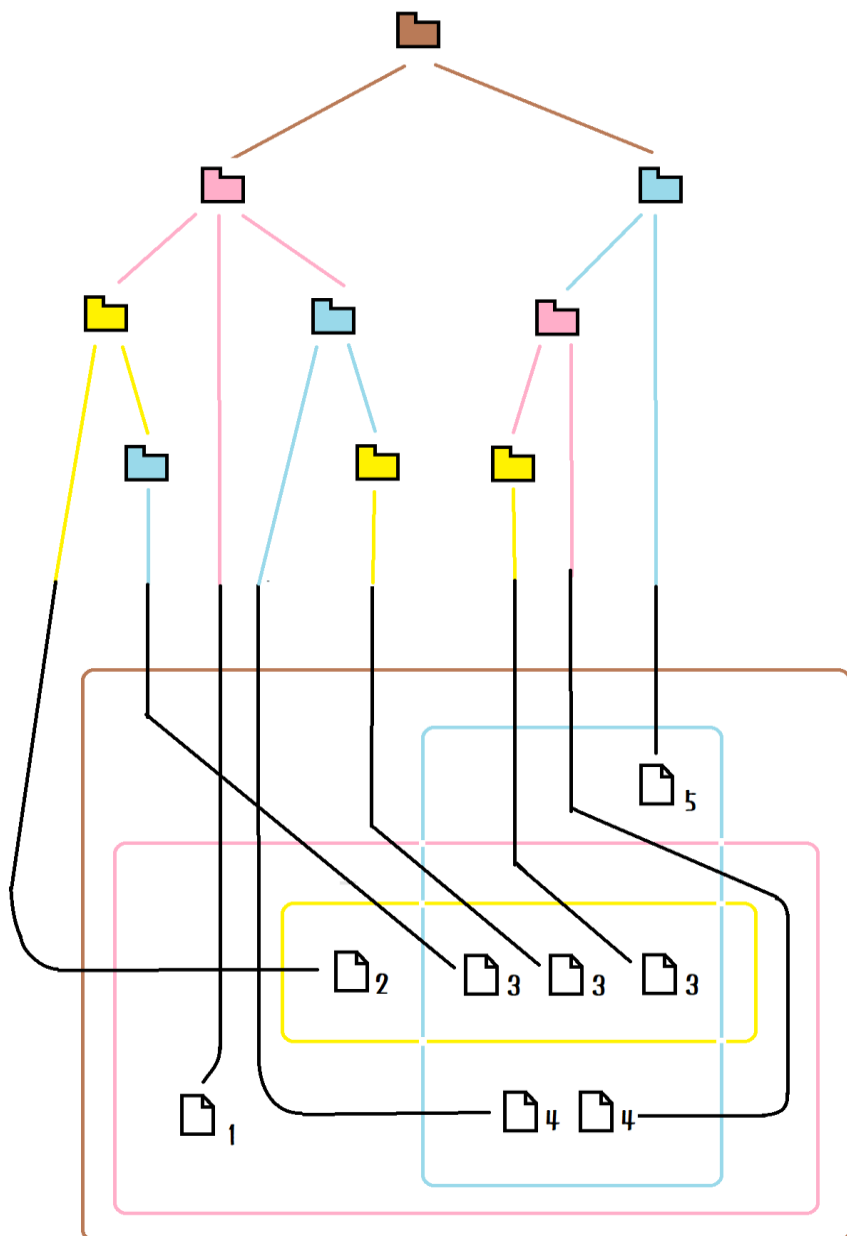
「☞ 青グループは はみでて
いるけど、 □4 は
置いている場所だけ見て
赤グループにすっぽり
含まれていると考えるの？」



「☞ 赤グループ
の方が
はみ出ている
とも言えるだろ！

順列だぜ！」





「👉 こうかなあ？」

でも フォルダが ばらばら
になって、
グループの利便性が無いような
気がするのよね」



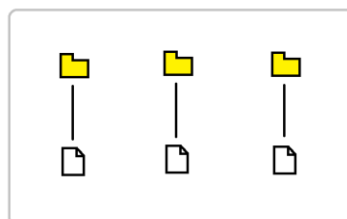
「外側から 歩いてきて、
踏んづけた色の順だな」



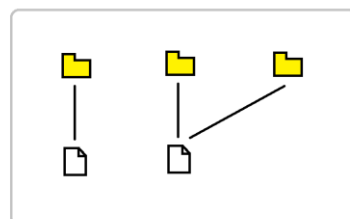
「全射であり 単射じゃないのも
フォルダっぽくないなあ」



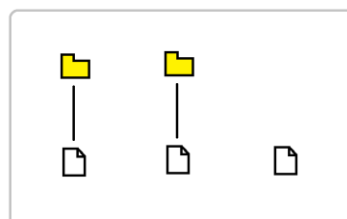
「ゼンシャってどんな
電車だぜ？」



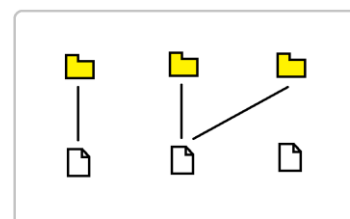
全単射



全射だが単射でない



単射だが全射でない



全射でも単射でもない

👉 フォルダからは1本線が伸びるものとする



「👁 わたしが今 関心を
持っているのが 右図だけ」



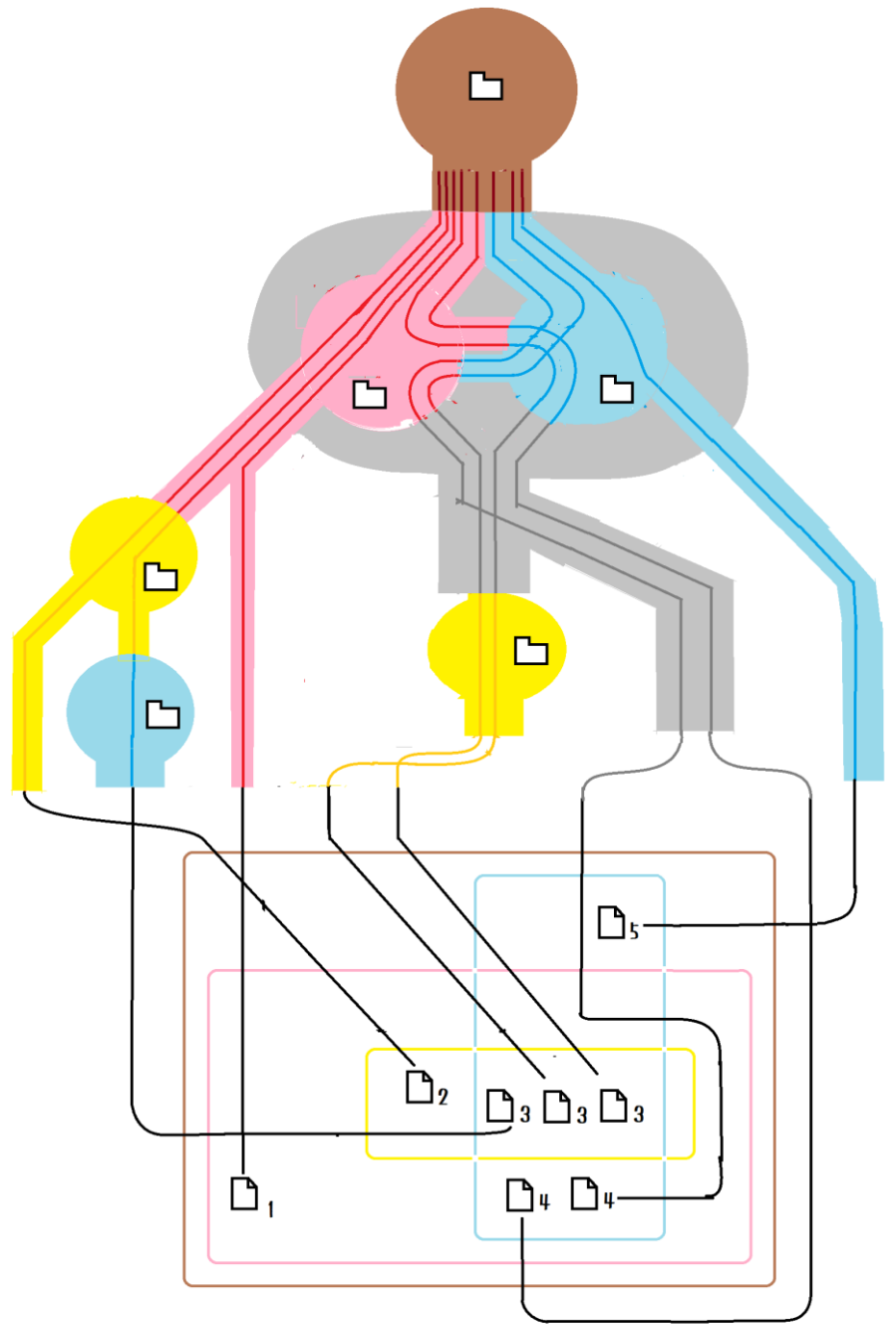
「ケーブル カバー
みたいだな」



「順列 (Permutation) になる
2つのフォルダーを
まとめたのね」



「順列というのは
手順前後というやつだな」

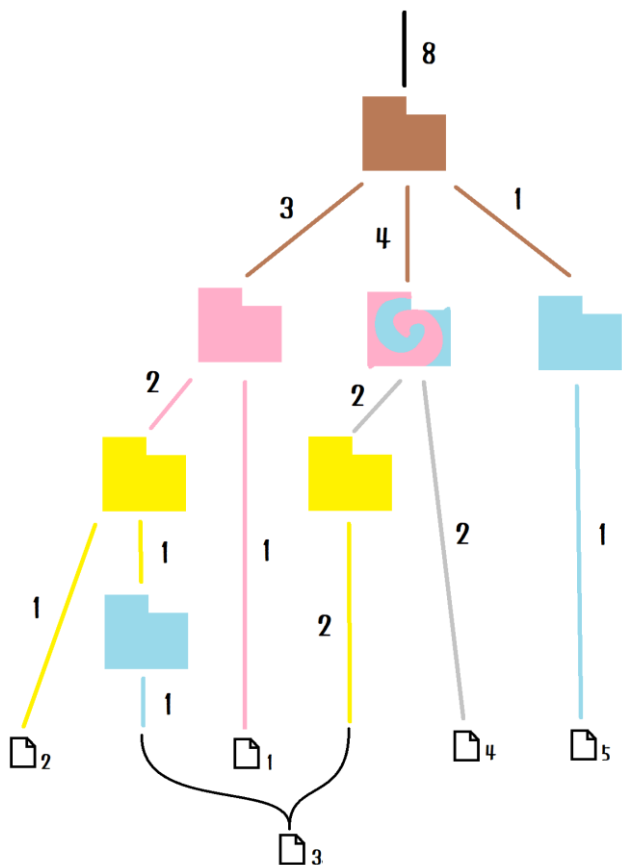
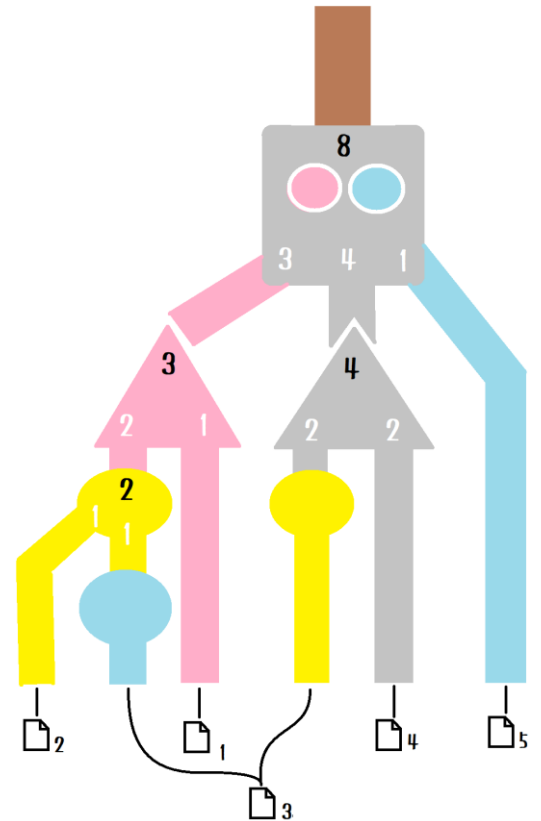




「それぞれの分岐での、
入ってきた数を黒で、
出ていく数を白で
示したぜ」



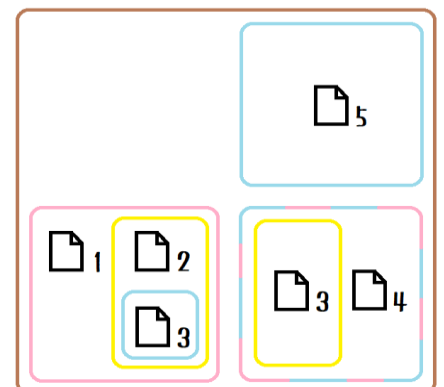
「パチンコの釘みたいだな」



「混色フォルダーを導入すれば
ひよこのツリーより すっきりしたな！」



「人類には100年早そうなフォルダーねえ」



「右図のような
はみだしの無い グループを作れば
ツリーの形を しているな」



「☞ そのツリーの経路を全て通れば
確率 分かるな」



「□3 が3つ、□4 が2つ で
他より多いな」



$$\frac{3}{8} \times \frac{2}{3} \times \frac{1}{2} \times \frac{1}{1} \times \text{document}_2 = \frac{1}{8}$$

$$\frac{3}{8} \times \frac{2}{3} \times \frac{1}{2} \times \text{document}_3 = \frac{1}{8}$$

$$\frac{3}{8} \times \frac{1}{3} \times \text{document}_1 = \frac{1}{8}$$

$$\frac{4}{8} \times \frac{2}{4} \times \frac{1}{2} \times \text{document}_3 = \frac{1}{8}$$

$$\frac{4}{8} \times \frac{2}{4} \times \frac{1}{2} \times \text{document}_4 = \frac{1}{8}$$

$$\frac{4}{8} \times \frac{1}{4} \times \text{document}_3 = \frac{1}{8}$$

$$\frac{4}{8} \times \frac{1}{4} \times \text{document}_4 = \frac{1}{8}$$

$$\frac{1}{8} \times \frac{1}{1} \times \text{document}_5 = \frac{1}{8}$$

$$\begin{aligned} & \text{sum} \quad \text{document}_3 = \frac{3}{8} \\ & \text{sum} \quad \text{document}_4 = \frac{2}{8} \end{aligned}$$



「☞ 出口の数÷入り口の数
が基本的な考え方だぜ。」

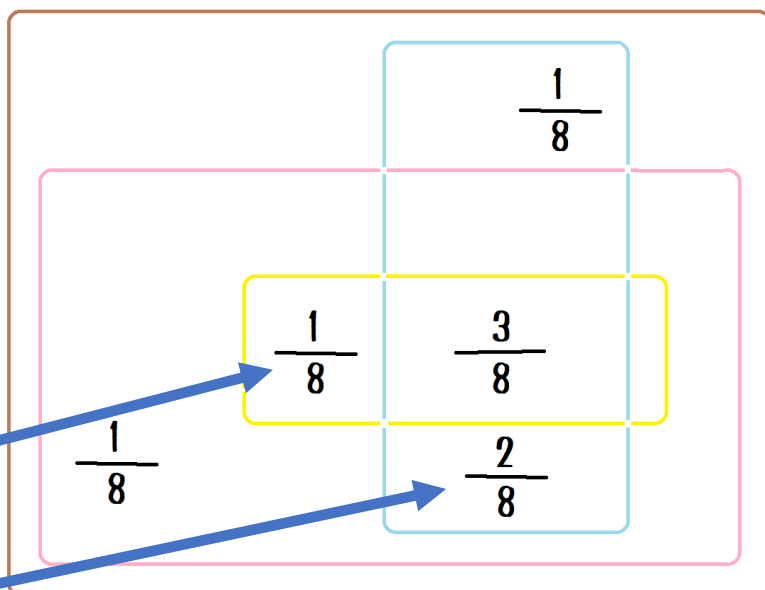
経路が違うが同じものなら
最後に 足し合わせるのを
忘れるなだぜ」



「☞ そしたら
分布 分かるな」



「黄色グループは 赤グループに
すっぽり入ってるけど、青色グループ
は赤グループからはみ出ているから
順列の分 確率が高いところがあるのね」





「ツリー構造と グループは 横から見たか、上から見たか
ぐらいの違いだということが 分かったかだぜ？」



「横とは何か」

「上とは何か」



「これらのことは セミラティス構造 の 例 の画像を見て 気づいたんだぜ。
都市設計の アレグザンダーという人は あれやこれや
考えて ツリー構造ではないものに 関心があったらしいぜ」

A-11 なぜ人工的な都市は失敗するのか?-----クリストファー・アレグザンダー
http://www.kenji-world.net/fieldlabo/d_d.html

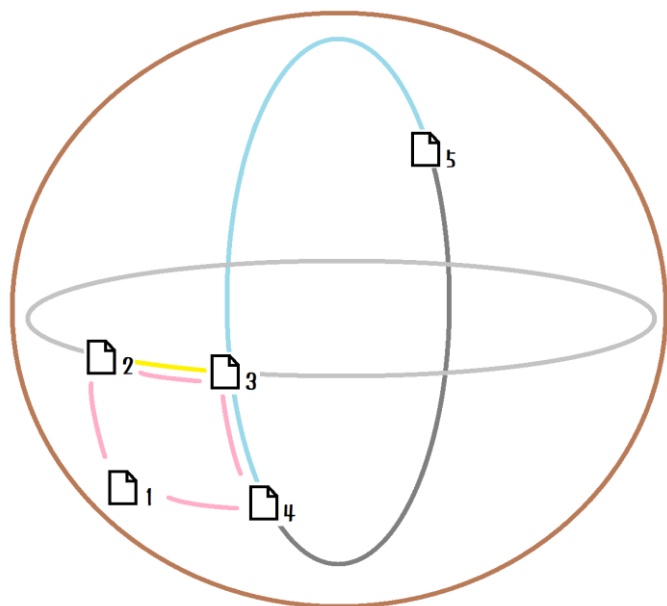
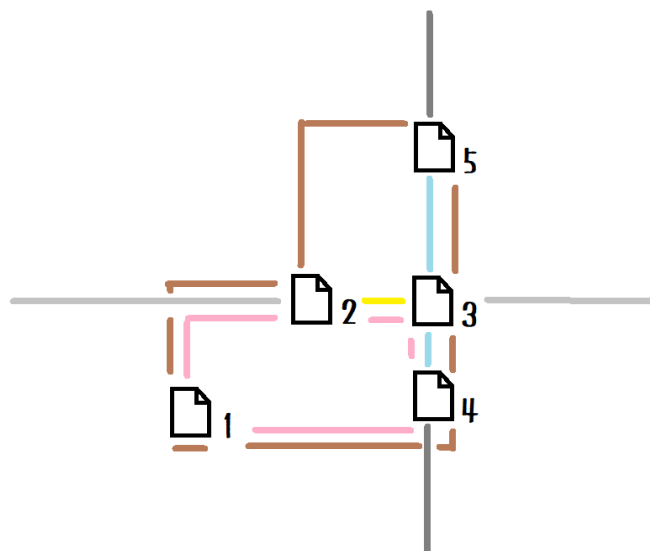


「そこらへんの いろんなところに グループ構造はあるよな。」

☞ 路線図にしたって グループ
足りえるはずだぜ」



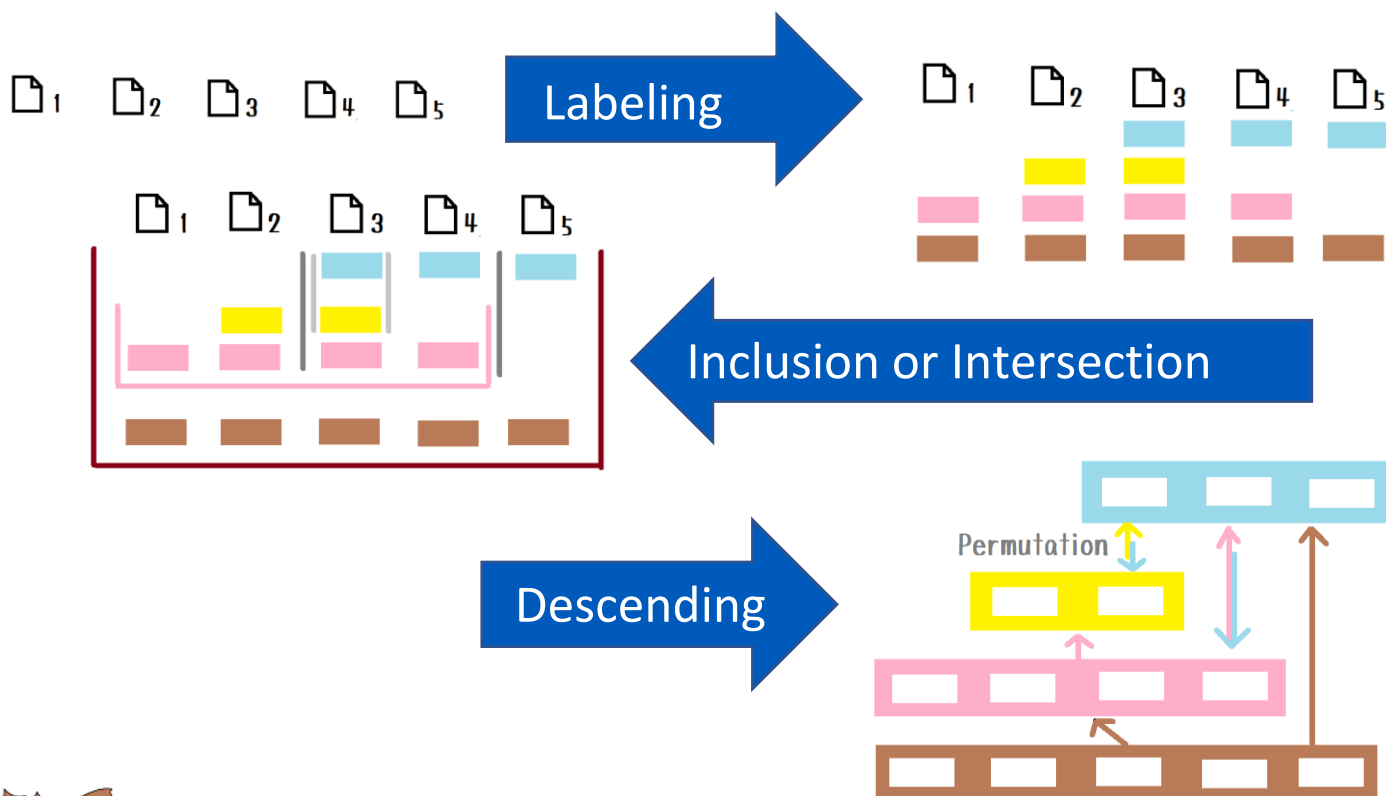
「通勤は 経路 だしな」



☞ 地球儀も興味深い」



「グループに はまってるのねえ」



「👉 ラベルを付けたら ソートするだけで よくね？」



「うさんくさい 本屋の新刊のようだ」

権限の 付与 って 何なのよ？

グループ DAZE だぜ



「で、権限って何なの？」



「☞ グループ構造だぜ」



「話が雑なの わらう」



「構造じゃなくて
物自体の説明は
無いの？」



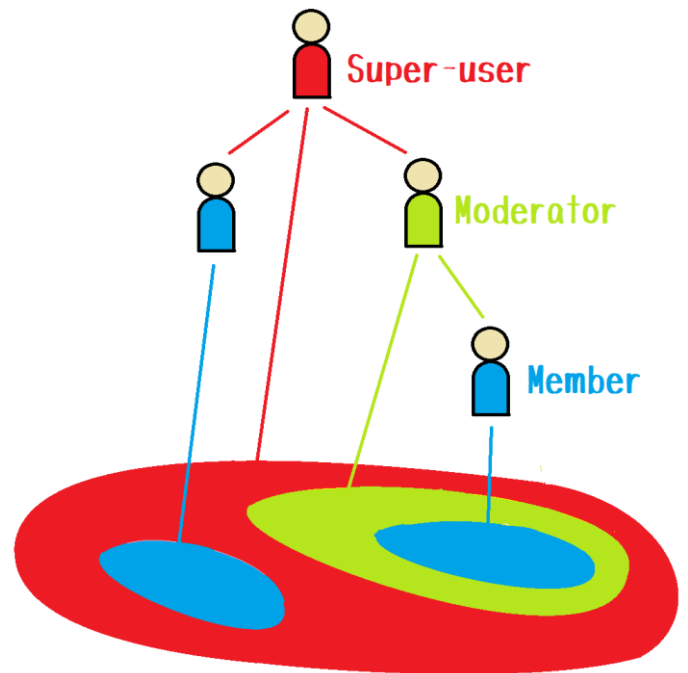
「構造が分かれば
構造の中に何が入ってようが
どうでもいいじゃないか」



「確かにお父んはずっと 構造の説明をしているな」



「Django のドキュメントを読んでいると
権限の **プロキシーモデル** ってのが出てくるのよ。
プロキシー って何なのか 説明がどこにも無いんだけど
これって 何なのよ？」

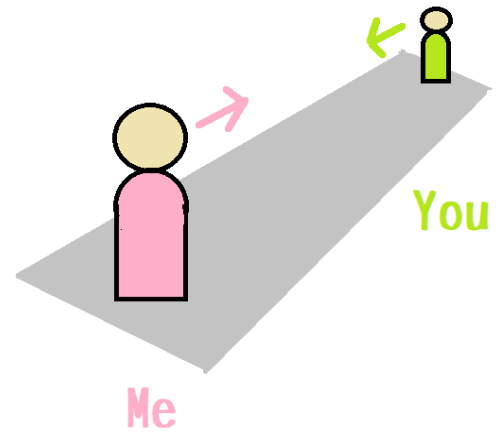




「お父んが知ってるわけないぜ」



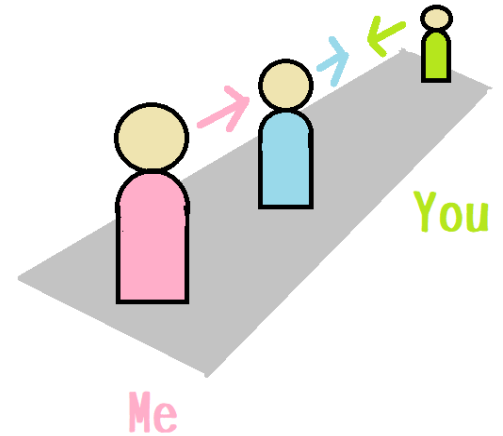
「☞ 西洋の言葉は わたしが外を見た、
みたいな 構造をしていて
自分を起点に 方向があるが……」



「有向グラフ なんだな」



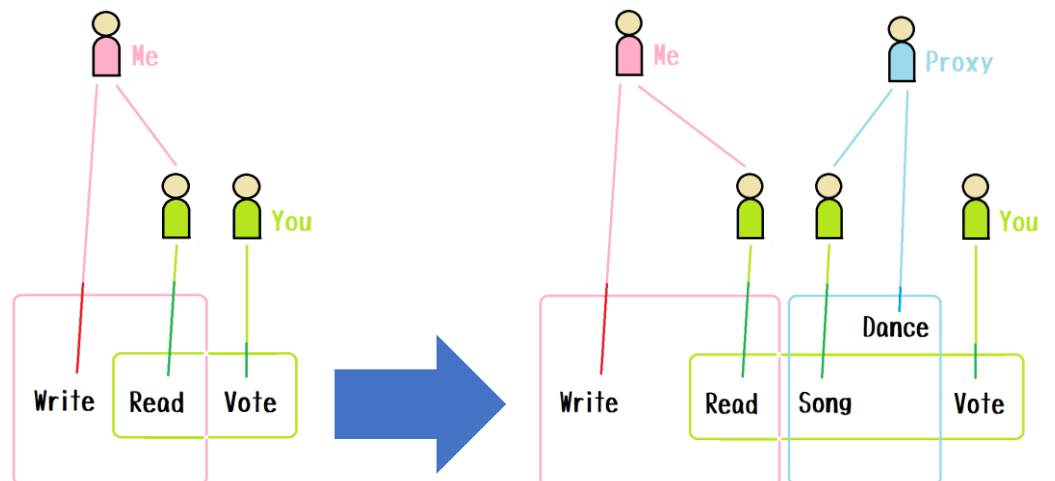
「プロキシーは うしろの わたしを
隠すために 2者の間に置かれる、
わたしと同じ方を向いているものだけ」



「それが 権限の付与の話題の中で
出てきたら、何なのよ？」



「☞ こうだろ」



「思っているのと 違うところに Proxy さんが増えた！」

グループで将棋エンジンを作れないのかだけ？

NEURAL NETWORK

なんで ニューラルネット にしないのよ？



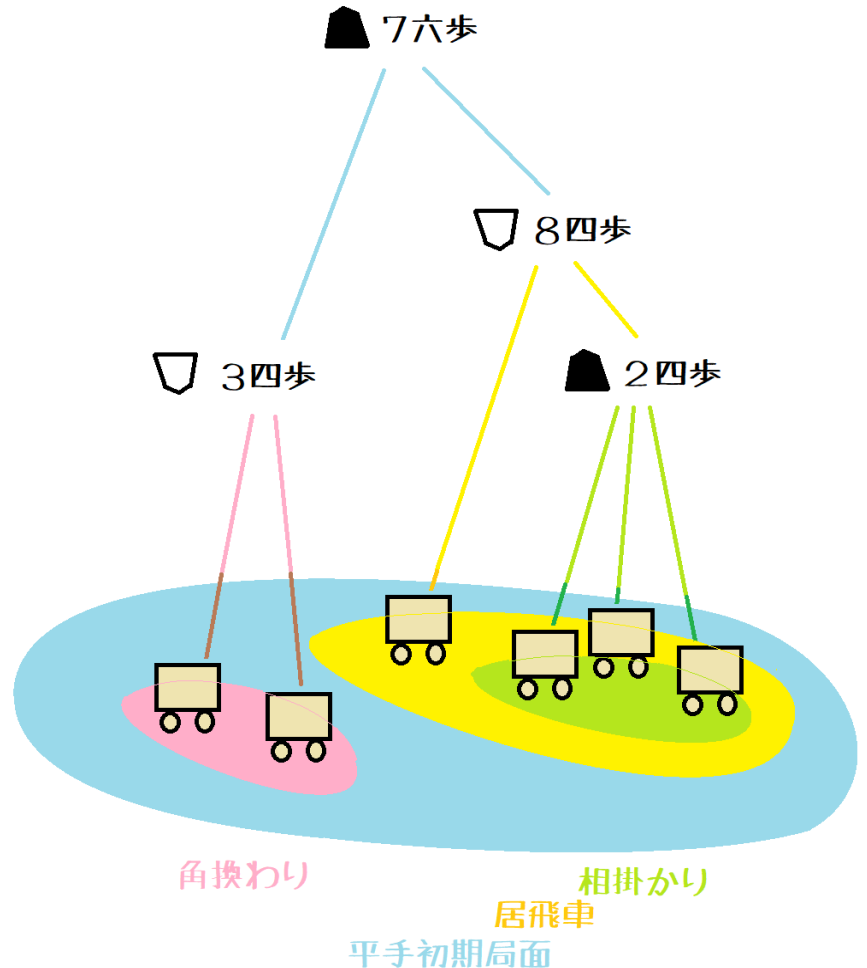
「お父んの大好きな
グループで 将棋エンジン
作れないのかだけ？」



「将棋の状態数を
グループにすると
約10の68～69乗あるそうだけ。」

将棋の経路の数は
下限値でも 10の220乗は
あるそうで、

宇宙に観測可能な陽子の数
エディントン数
約10の79乗より多くなるな」



コンピュータ将棋基礎情報研究所 2016年03月21日『将棋の局面数 1：局面数は無量大数』

<http://lfics81.techblog.jp/archives/2249793.html>

コンピュータ将棋基礎情報研究所 2016年03月25日『将棋の棋譜数：“10の220乗”説の真相！？』

<http://lfics81.techblog.jp/archives/2319578.html>

巨大数研究 Wiki 『エディントン数』

<https://googology.fandom.com/ja/wiki/エディントン数>



「情報技術の最先端の
やねうら王 や elmo に勝たなくていいので
れさかい に勝ってくれだけ」



「それも難しいぜ。
将棋のルール通りに指すのが難しい」



「思考を組み立てるには
論理 が必要でしょ。
グループだけを使って
NAND回路を獲得できないの？」



「面白い課題だね。」

異なる始点を持つ
2つの全順序の構造 が
共通する
1つのグループを作るような、

全射であり単射ではない
構造なんだろう？」



「思考フレームワーク を
作るところからだね」



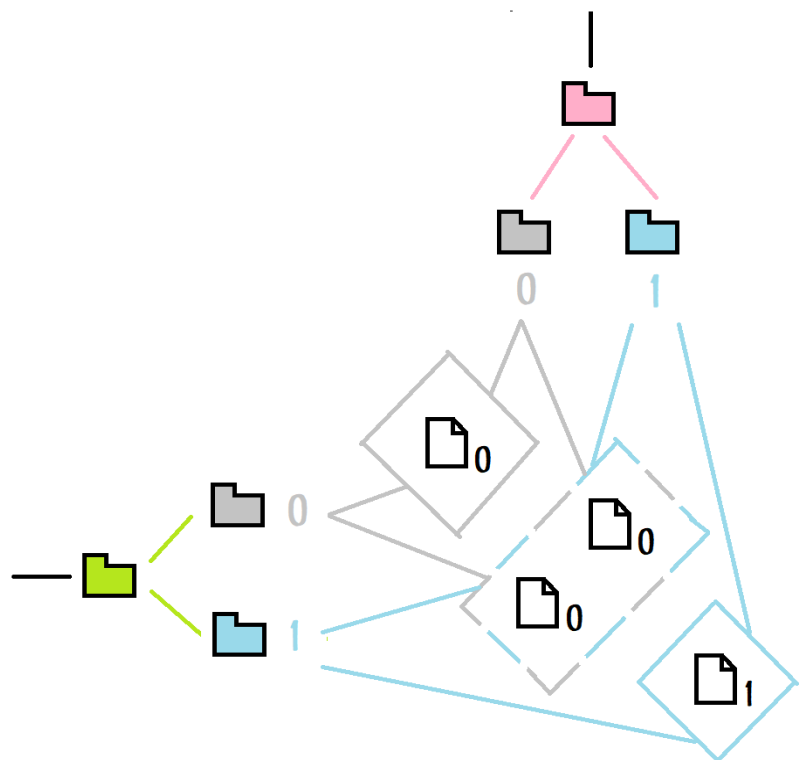
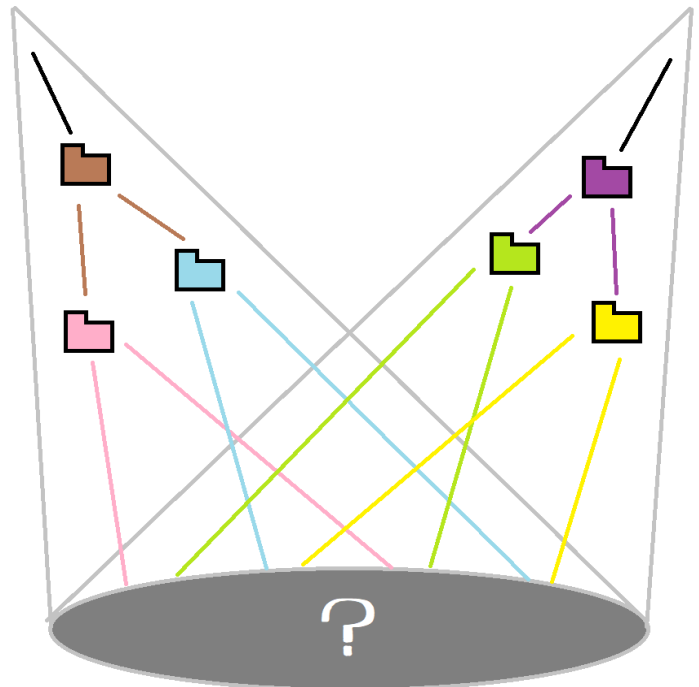
「☞ あっ、
AND回路は簡単だったぜ」



「☞ この イラスト
の表現が見慣れない
だけで、やってることは
既存の 情報処理
じゃないの？」



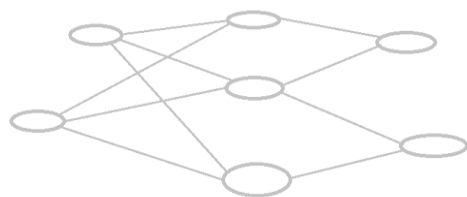
「えっ？ 何？
情報処理を
車輪の再発明
しているのかだね??」



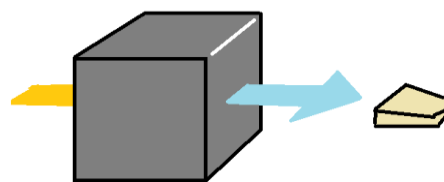
☞ AND回路



「要らん話しをしていたら 日付が変わった。 寝る時間だぜ」



Neural network



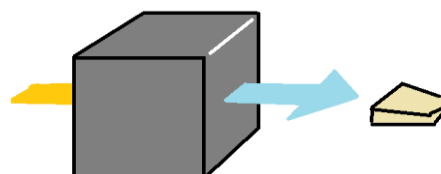
Black box



「👉 できれば ニンゲン と同じ 思考フレームワーク を
獲得してほしいぜ。
ニューラルネットは ブラックボックス だからな」



Human



Black box



「👉 ニンゲンの思考フレームワークとは何か を定義しなさいよ」



「もう 寝ろ」



「👂 言語は 自分を他人に
するツールだぜ。

感情は自分のものしかない。
しかし言語は 他人が使っ
ているのを覚えて 使ってる
んだぜ。

言語で考える人は
他人になっている人」



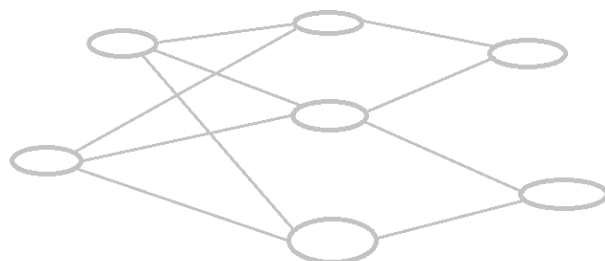
「意識が欲しいわけか」



「人工知能は作りたいけど、
生命は創りたくないのよねえ」

Language

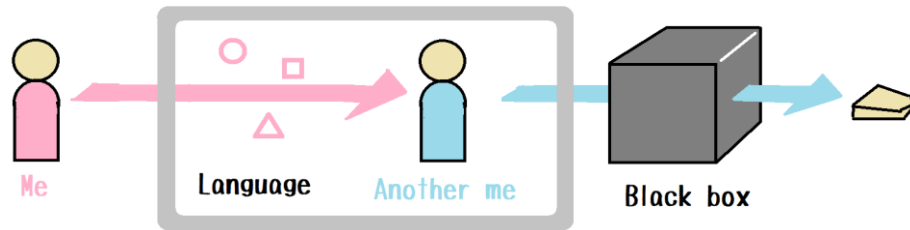
玉の固さ



Neural network



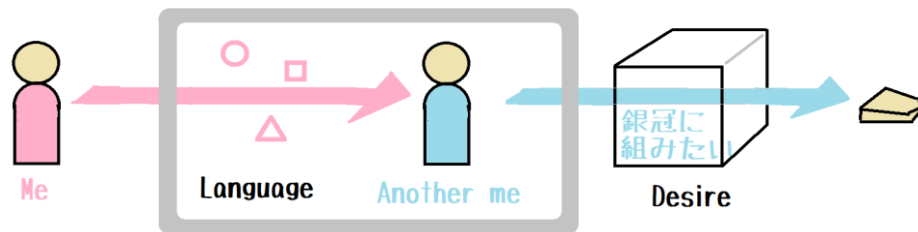
「まあ ニンゲンの思考のフレームワークを真似たら
そこには 生命 ならではのものが ありと考える方が自然かも知らんけどな」



「👉 お父んの説明を絵にしても ブラックボックスが消えないが」

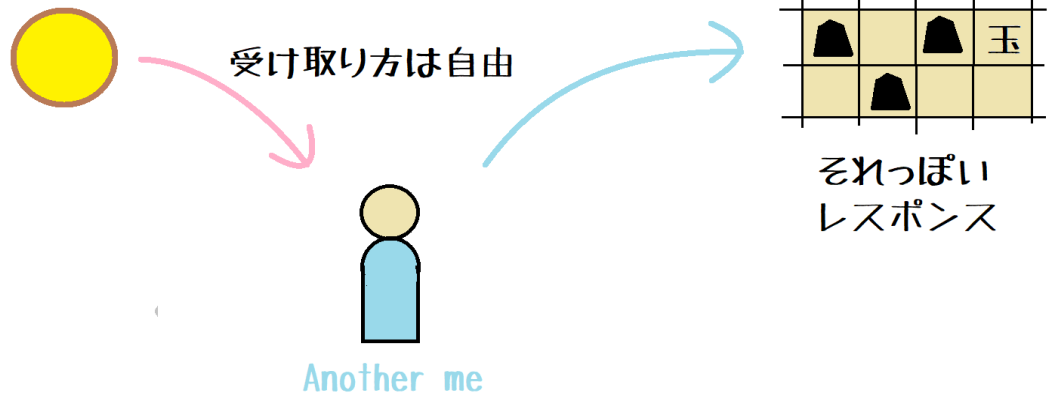


「アマチュア5級の Black box の中身は簡単だろ。
相手を見ず 銀冠 一直線に組みたいとか、
ガンガン 金、銀の駒を上がりたいとか、欲望が入ってるんだぜ」



「👉 ふつつ 奥底に欲望があって、その欲望を押さえるように理性があると
思うけど この絵では 前面 に欲望があって、うしろに 理性 があるのね」

言語というボール



「👉 だいたい、入力した情報より 精密な出力が出てくるの
おかしいよな。 もとから精密な出力ができるのが 自然だぜ」



「潜在的に 自発的に動けるが、まだそれができていない ニンゲン に対して
きっかけを与える のが 言葉 の限界じゃないのか？
できないやつに 言葉は届かない」

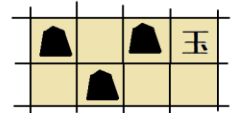
言葉という最後のピース



ほとんどできている
パズル



Another me



それっぽい
レスポンス



「👉 じゃあ こうよ」



「間違ったピース渡して ダメにするやつ いるよな」



「もし 言語化できないところが **強さの大部分を占める** のだとしたら、

コンピューターに 分かりやすく 説明してもらって

楽しんで 将棋が強くなる、

と 考えている わたしたちが そもそも 間違いなんじゃない？」



「気づいたか。
長い旅を終えたな」



「指導棋士なら 急所を突く 指針みたいなものを 教えてくれるだろ。

言葉の最後のピース のようなものは 実在するだろ。

大部分を占める弱さ の部分が オートで動かしている欲望によっているなら、
少ない言葉で いい感じで コントロールできることもあるんじゃないか？」



「弱さを Not できる言語、そんなもんがあれば 強さか。
その前提として、ニンゲン全員の欲望は 似たり寄ったりか？

宇宙人の子どもの オートで動いている欲望を

地球人の子ども向けの 言語セット で指導できるか分からないぜ？」



「わたしは思う。むしろ、
ハードコーディングされている **欲望こそロジックで説明できる** のでは。
理性の方こそ統計でしか説明できない のでは」



「腹が減ったら 食べますもんね。
人の理性は 人生の数 だけあるとしたら、
ロジックで説明する楽しさが 減るわよね」



「典型的な 欲望 を グループで記述して、
その欲望を うしろからコントロールするような 言語 を
予め 決めておいたらどうだろうか？」



「それで コンピューターは 人間と似たような 思考のフレームワークを
獲得できるのかだけ？」



「きふわらべ に 言語で指示出しても 言うこと理解してくれないしな。
指し手が ごそっと 修正されるような 何か仕組みを考えないといけないな」



「間に合わないわねえ」

>>> Dad, it's time to
sleep!!



十六式いろは煌(きらめき)



自己紹介

末吉 竜介「学生の皆さんと、いろいろと学び試し楽しみながら作っていければと思います。」

岩田 大夢「AIを学んで早一年...まだまだ分からないことだらけですが、この大会を通じて成長していきたいと思っています！」

紺野 誠「将棋の知識を活かしていきたい」

若月 翔威「この大会を通じて成長できるようにがんばりたいです」

小林 優輝「手取り足取りみんなで頑張っていきたいと思います！」

上野 勇樹「精一杯頑張りたいと思います！」

村越 小友梨「AIと将棋どちらも上達できるように頑張ります！」

長谷部 太一「大会を通して何か得られるものが有ればと思っています！」

畑中 慎吾「わからないことばかりですが、大会を通して経験を積み成長に繋がればと思います」

平沢 蒼「将棋もAIも未熟者ですが、精一杯頑張ります。」

李 愚昶「将棋AIは初めてでまだまだですが、上達できるようがんばります！」

若松 萌生「ひとつでも多くのことを学び、成長出来たらと思います」

山内 伊織「...ノーコメント」

風間 俊介「...ノーコメント」



「十六式いろは 煌」の由来

様々な名前の候補が上がり最終的に決まったのが考え始めてからなんと **1か月**かかりました！。

皇(すめらぎ)、煌(きらめき)、日本工学院、かまトウ(学校のマスコットキャラ) ...などなど。

「日本工学院の名前があった方がよいのではないか」や「ローマ字で書いた方がカッコいい！」

などかなりの意見などがありました但最终的には末吉先生の「十六式いろは」と生徒達で考えた「煌(きらめき)」を組み合わせで決定しました！



ソフトの概要

採用予定

- dlshogi
- やねうら王系
- 詰将棋(※)

(※)以下のいずれか

- 脊尾詰
- なのは詰め
- KomorningHeights

ソフトの説明

- 2～3台のマシンのクラスタ化(予定)。
- 合議制を採用。
- 持ち時間の管理を工夫(予定)。(詳細:序盤や深読み無しで指せる手はすぐに指して、熟慮が必要な場面に時間を多く割けるようにする。)
- floodgate、AobaZeroの棋譜に加えて、2022年3月現在で公開されている将棋ソフト同士の対戦の棋譜を用いて、dlshogiの機械学習をさせる。
- Optimizerの再選定。

A. I. Ari shogi アピール文章WCSC32

ver2

兵頭優空

1. 概要

AI Ari shogi (以後Ari)は、私が2021年7月ごろから開発している、ディープラーニングを使った(くそ弱い)将棋AIです。

ディープラーニングは、魔法の(ような)技術なわけですが、それを使っているのにも関わらず、くそ弱いです。

私が別で作っている、AI AN shogiとは別物ですが、ソースコードの一部を共有しています。(一部を流用することで、低コスト化を図っています。ようするに手抜きです※1)

~~ネットワークは、ResNet13b + α となっており、Value出力があります。(都合上、下記のスモールバージョンで出場する予定です)~~

ResNet3b + α のスモールバージョンがあり、2022年3月10時点ではスモール版でしか学習できていません。

最下位にならないように頑張りたいです。

2. 使用ライブラリ

将棋関連のものは、Python-shogiのみです。

ディープラーニング系はTensorflow. Kerasなどです。

(定跡生成に関して:外部のUSIエンジンを使用する可能性が高い

です。詳しいことは下に書いてあります)

3. アピールポイント

一般的なディープラーニング系と違い、モンテカルロ系の探索部ではなく、MinMax探索系の探索部を持っています。

(Policy出力のラベル関連の実装が面倒くさくなり、Policyヘッドをなくした。失敗だったと思う。)

GPUにデータを渡すとき、一気に沢山送ったほうが、ちまちま一つずつ送るよりも効率よく処理できると聞いたので、一気にデータを送るように工夫をしています。

僕が作った、ネタAIの「単細胞くん」と「シンプルニューラルネット」の探索部(現在の僕の最高傑作の探索部※2)を流用する方法を思いついたので、探索部がだいぶ変わりました。

流用前の探索部ですが、 $\alpha\beta$ 探索 + 静止探索 + move ordering といった感じです。(電竜戦さくらリーグ出場時から改良した)

図1

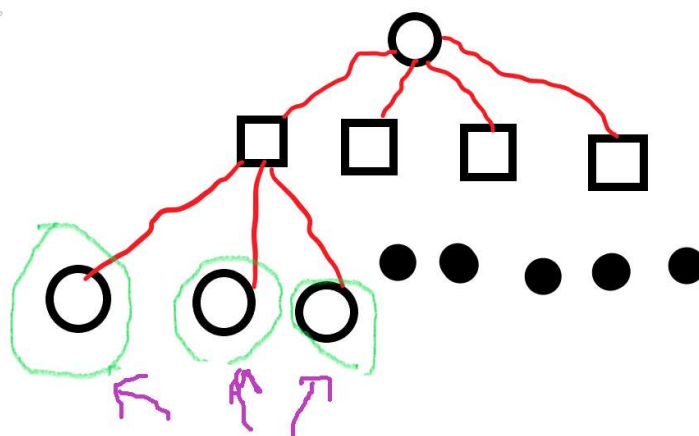
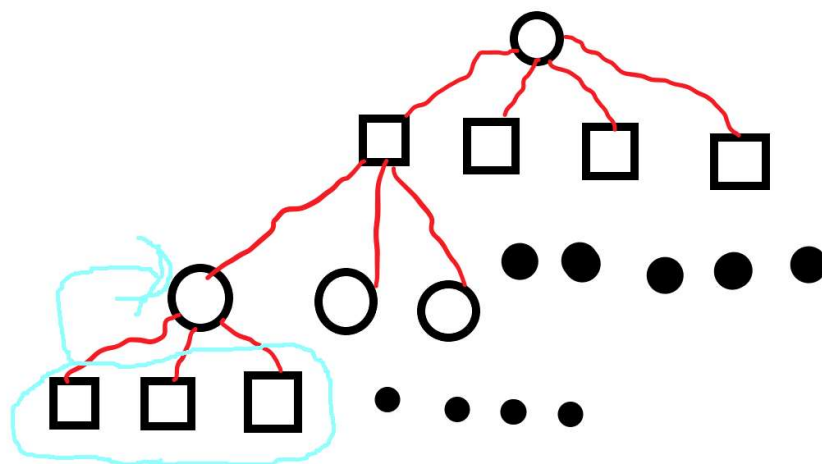


図2



※とても汚い図ですが、気にしないでください。

図1では、普通に1局面ずつ評価関数で評価してます。

単細胞くんとかは、こんな感じです。

図2では、○の局面の評価値を求めるのに、その子供たちである□の局面をまとめて評価しています。

こうすることで、末端の評価部分をちょっと変えるだけで最高傑作の探索部を流用でき、かつGPUをいい感じに使えるんです。

(枝狩りの部分が流用できるのが良い)

それに、新型詰みサーチと詰み回避機能を別でつけています。

あと、評価関数の精度を上げるように色々いじっています。

(が、まったく上手くいかないし、試行回数も全然足りてない)

具体的には、EarlyStoppingとかを試しています。

4. ニューラルネット

ResNet3ブロック + α したやつです。

出力は(1.0 ~ -1.0)のvalueです。

floodgateの棋譜とか、AobaZeroの棋譜とか、自己対局の棋譜とか使ってます。(他に使ったら追記します)

=====こっから追記=====

1: Ari shogiは将棋所などに対応してなかったのが、これまで外部AIとの対戦が困難だったのが、先日開発したUSIエンジン呼び出せる自作ツールのお陰で簡単に対戦できるようになったので、それらとの(ランダム要素配合)対戦データも使用しています。(対戦相手は主にれさ改です。ぼこぼこにやられています)

2: Ari shogiは、評価関数の精度が悪いのに加え、探索もノロいので、非常に弱いです。

毎回毎回、瞬殺されるので、少しでもマシになるように(そして根本的課題の存在を誤魔化すために)定跡を入れようと前々から思っていたのですが、公開されている定跡に対応するのが面倒くさかったため、保留にしていました。

その後、手動定跡ツール(ゴミ)を作ったのですが、定跡作成作業による精神的苦痛に耐えられなくなり(苦痛の主な原因は作者が将棋超初心者で定跡とかよくわからんのと、UIがゴミ未満だったこと)、秒で投げ出して無かったことにしました。

電竜戦さくらリーグを前に、いよいよやばいと思った作者は、簡易の自動定跡ツールを先日製作しました。

このプログラムは、

- 1: 局面を移動する。
- 2: 思考関数(後述)に局面を渡して、bestmoveを受け取る。
- 3: Pythonのdictで管理している定跡に、その局面のsfenをKeyに、bestmoveをValueにして追加する。
- 4: 上記の作業を繰り返してできた定跡を、jsonファイルに保存する。

という作業をします。

思考関数ですが、局面を渡されてbestmove返せばなんでもいいです。

ですが、弱かったら意味ないので、現在はとりあえず上記のUSIエンジン呼び出しツールを利用して作っています

4月時点の予定では、れさ改を使用して作った定跡で出る予定です。

5. 色々

- ・今回の大会、何故かAN shogiの後継扱いされていますが、AN shogiのほうの開発は並行で続行しています。

(つまり、本当は後継ではない)

- ・作者は中高一貫校に通っていて、高校受験等々がなかったので開発につかえる時間が沢山あったので、大会までに少しはましにしたいと思います。

- ・作者は、少し前からGANとかでも遊んでいるので、学習が余計に停滞しています。(電竜戦のときは、将棋の盤面を生成するShogiGANというのを作っていましたが、あれは大失敗しました。今、新たにバージョン2を製作中です)

・れさ改って、作者が生まれる前からあるそうで、最初は「ディープラーニング使っているし、少し学習すれば勝てるだろう」と思っていたましたが、まったく勝てません。

・作者は他にも、自作AI用のシンプルなストラテジーゲームを作っていたり、TetrisのAIを作ったり、株価予想システムを作ったりしているので、リソース(時間・計算資源・気力)が足りていません。

=====

※1: 大抵の問題は、100しかないリソースをどうするかが問われている。

つまり、如何に手抜きをするのかは、とても大事ということだ。

※2: 僕の技術力は上がっている = 最新のやつが最高傑作

申し込み者：北川博隆（HN:雨宮一也）

グループ名：重力団

プログラム名：重力場計算法

第 32 回大会提出アピール文

盤面評価を重力場計算法を用いて計算します。

重力場計算法 PV

将棋の数学的解明に必要となる哲学と理論物理学、そして開発者に課せられる心構えについて謳った動画。

ニコニコ動画

<https://www.nicovideo.jp/watch/sm40237301>

YouTube

<https://youtu.be/m3nkswy9X3g>

第 31 回大会提出アピール文

■「参加プログラムは、主要な開発者が思考部に技術的に何らかの明示的な工夫を施したプログラムであること。」
を満たすことをアピールしていただくための文書

このプログラムの思考部は「盤面評価を最高精度で算出する事で最善の一手が特定できる」との趣旨で構成されています。

その為以下の様な特徴があります。

- ・通常の「深い読み」を行わない。
- ・「定跡・棋譜」を一切利用しない。解析データも利用しない。
- ・将棋のルールと評価設定のみを入力する。
- ・盤面判断は重力場理論（簡易版）を使用して計算を行う。

将棋はお互いに一手ずつ指すことで競技が進行しますが、手番、非手番に与えられる条件は都度同じです。従って将棋の本質とは与えられた盤面に対する最小手数範囲を計算できれば良いのであり、それ以上の処理は重複作業になるため不要と言えます。結果、根拠のない「深い読み」を意識的に行わないプログラムとなります。

又、将棋のルール内に過去の「棋譜・定跡」の使用を義務付ける項目はないし、それらによって競技ルールに何かしらの影響が発生する事ありません。従って「棋譜・定跡」をプログラム内に入力することは無駄であると言えます。

盤面判断を行う上で最も困難とされていたのは「個々の設定の解析」と「異なる単位を正確に比較する手法」です。その為今までの盤面評価プログラムは精度が悪く、進歩が止まりました。しかし、この二つの課題は重力場理論によって克服することが出来ます。

私たちの制作したこのプログラムは異なる単位の設定であっても、重力場理論（簡易版）によって計測処理を行い、上昇値の合計を比較する事で「最高精度の盤面評価」が理論上達成されます。

重力場理論に基づく究極のアルゴリズムは「神の一手」を特定できるのです。つまり「将棋の数学的解明」はこの理論によって完成されるでしょう。

■「このプログラムの思考における工夫や独自性について」

従来のプログラム思考技術や人間同士の対局で培われた知識や成果を基にした「工夫」は行われていません。なぜならこのプログラムはそれらを全否定する所から構成が行われているからです。

真逆の方向性を目指して作られている上に、将棋のルールと評価設定をプログラムに入力しているだけの代物であり、面白味は何もありません。只、淡々と盤面向上を追求する一手を計算しています。それこそが究極の将棋なので、人為的努力の「工夫」は不要です。

しいて言えば、「人間の知恵と技術と歴史と情熱を皆無にする事」が既存のプログラムにない「独自性」なのかもしれません。

対象物を”鏡に映しただけ”の作業は工夫もなければ独自性も発生しません。しかし、世界で初めてその行為を行った場合は「イノベーション」として評価されます。そして現実社会に多大な影響と貢献をもたらせば「究極 AI 産業革命」として歴史に記録されます。”将棋を鏡に映しただけ”なのに……ね。

第3 2回世界コンピュータ将棋選手権「水匠」アピール文書

令和4年3月31日

杉 村 達 也

第1 使う他者作成プログラム

やねうら王→探索部、学習部を使用させていただきます。

dlshogi→探索部、学習部を使用させていただきます。

第2 評価関数の学習

1 NNUE評価関数の学習

NNUEにはFV_SCALEという値があり、この値を変更しながら学習させると、評価関数の評価値スケールや勝敗項に関するlossの減り方、勝率項に関するlossの減り方が変わります。水匠評価関数の追加学習は、NNUE学習パラメータのlambdaを0、すなわち勝率項を見ずに勝敗項だけで学習させているので、FV_SCALEをいい感じに変更して学習させることが大切です（なぜ学習が上手くいっているのかはよくわかりません。）。

教師局面は水匠/やねうら王で1手約200万ノード探索させて対局させた棋譜を使用しています（現在約3億局面）。

2 dlshogi評価関数の学習

dlshogi評価関数は15ブロック224フィルタのモデルを学習させています。先ほどの水匠で作成した教師局面と、dlshogiで作成した教師局面をepoch毎に交互に学習させると徐々に強くなるので、半年前くらいから学習を継続しています（なぜ学習が上手くいっているのかはよくわかりません。）。

第3 その他

1 探索部の改善

最新のStockfishのコードで強くなる部分をマージします。

2 定跡の作成

- ① 任意の局面を与え、その局面から連続対局
- ② 特定の局面からの対局数が一定数(例:>100)を超えたら枝局面を開始局面として以下同様に繰り返す。
- ③ 最終的には勝率を用いてmin-max的な手法で、リーフ局面からルート局面までを伝播させる。

という手法で定跡を作成します。

3 楽観合議

学習させたNNUE評価関数とdlshogi型評価関数の両者とも使いたいので、評価値のスケールをいい感じに調整した上で、楽観合議をさせる予定です。

以上

探索アルゴリズム — Best First MinMax Search（最良優先ミニマックス探索）をアレンジ

普通の実装では「末端でノード展開したら終わり」ですが、これでは探索結果が不安定になったので最低深度まで展開します。以前の探索がそこまで達していた場合、そこから更に探索します。これを繰り返します。

時として、評価値がぶれるので最終評価値は平均値を使用。

AWS/EC2使用で並列処理予定。

2022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て

一般的な手法である $\alpha \beta$ 法、ハッシュテーブル、null move等を使っています。

3 駒間評価値をプロ棋譜約2600とコンピュータ将棋選手権上位の棋譜、

強い将棋プログラムと旧なり金との対戦棋譜を使い教師あり学習させています。

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1013万ステップくらいあります。また別に詰将棋部が1万ステップくらい、画像部が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 ラベルは Windows です

ボリューム シリアル番号は 6A33-AF78 です

C:\¥VC2019¥OkiSikou のディレクトリ

2021/12/25 06:53 <DIR> .

2021/12/25 06:53 <DIR> ..

2021/10/08 15:44 407,424 aa0.h

2021/10/16 17:53 980,438 aa1.h

2021/11/25 12:54 983,467 aa2.h

2021/11/09 11:55 949,696 aa3.h

2021/10/17 05:57 459,606 aa4.h

2021/10/16 17:13 903,681 aa5.h

2021/10/24 16:13 914,913 aa6.h

2021/10/03 09:51 889,805 aa7.h

2021/10/03 09:51	121,598 aa8.h
2021/10/03 09:51	928,566 aa9.h
2021/10/12 12:29	927,880 aaa.h
2021/10/03 10:07	904,803 aab.h
2021/10/08 15:56	894,965 aac.h
2021/11/25 13:43	1,313,184 aad.h
2021/07/11 10:20	896,482 ab1.h
2021/07/10 09:06	584,175 ab2.h
2021/11/26 09:17	893,480 ag1.h
2021/11/25 11:05	906,960 ag2.h
2021/11/26 09:51	645,966 ag3.h
2021/11/29 06:29	481,567 Ai.h
2007/02/26 16:50	332 Ai_g.cpp
2007/06/30 07:42	333 Ai_s.cpp
2021/10/11 16:43	927,036 aj1.h
2021/11/04 21:58	881,770 aj2.h
2021/12/02 06:13	894,685 aj3.h
2021/10/05 15:33	180,730 aj4.h
2021/10/05 15:33	898,595 aj5.h
2021/10/05 15:33	868,320 aj6.h
2021/10/05 15:33	255,407 aj7.h
2021/12/02 06:05	898,324 aj8.h
2021/10/05 15:33	895,006 aj9.h
2021/12/02 06:05	385,038 aja.h
2021/11/23 10:10	917,758 ak1.h
2021/11/23 10:24	911,249 ak2.h
2021/11/23 10:10	903,281 ak3.h
2021/11/23 10:10	902,967 ak4.h
2021/11/23 13:03	894,958 ak5.h

2021/11/23 13:03	450,967 ak6.h
2009/09/13 16:05	444 amari.cpp
2020/05/13 17:25	7,683 ana.h
2007/02/26 16:50	765 Ana_g.cpp
2007/02/26 16:50	747 Ana_s.cpp
2021/12/23 11:45	970,897 ao1.h
2021/12/23 11:45	608,942 ao2.h
2021/10/23 05:55	938,375 as1.h
2021/10/23 11:14	918,778 as2.h
2021/12/22 14:59	898,642 as3.h
2021/10/23 11:20	887,973 as4.h
2021/12/23 06:09	107,171 as5.h
2021/05/09 16:47	965,171 at1.h
2021/05/09 17:08	888,162 at2.h
2021/12/19 17:22	861,556 az1.h
2021/11/04 12:42	874,659 az2.h
2021/12/19 17:22	421,372 az3.h
2021/09/18 06:25	348,659 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
2021/11/10 13:49	258,762 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 gfttotal.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
2021/03/18 06:15	32,368 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
2021/12/03 12:47	955,875 gx1.h
2021/12/03 11:56	900,818 gx2.h
2021/12/03 11:56	882,508 gx3.h
2021/12/03 15:12	70,700 gx4.h
2011/10/08 17:50	23,935 gxctotal.cpp
2010/04/20 06:06	22,298 gzttotal.cpp
2009/12/10 08:19	3,952 g_kiki.cpp
2021/06/01 06:15	16,564 g_total.cpp

2021/04/30 11:09	386 hasshu.cpp
2004/02/26 22:26	15,987 hikyo_s.h
2000/04/08 05:06	2,115 hikyo_u.h
2019/02/24 17:11	54,145 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
2021/10/25 16:30	18,878 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
2021/12/17 11:01	1,024 INFO.dat
2016/05/04 08:57	2,038 Jissen.lib
2014/05/19 10:25	13,199 Joseki.cpp
2021/06/21 10:46	167,453 Joshiki.h
2019/10/17 10:44	37,575 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
2021/07/23 08:57	82,379 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
2021/12/06 06:22	192,692 Mukai.h
2018/05/26 06:29	378 Mukai_g.cpp
2018/05/26 06:29	381 Mukai_s.cpp
2021/05/28 06:12	199,784 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

2018/03/04 10:22	910 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
2021/10/28 12:16	16,912 OkiSikou.cpp
2006/02/15 21:41	146 OkiSikou.def
2006/02/15 21:57	527 OkiSikou.h
2021/01/11 09:10	3,145 OkiSikou.rc
2021/04/25 16:34	1,232 OkiSikou.sln
2021/12/03 17:43	29,942 OkiSikou.vcxproj
2021/12/03 17:43	50,803 OkiSikou.vcxproj.filters
2021/12/25 06:34	863 OkiSikou.vcxproj.user
2009/02/12 13:09	3,522 ouchifu.cpp
2021/03/22 12:09	37,424 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
2021/11/12 13:01	164,689 p1sort.cpp
2021/01/12 11:56	65,621 p2sort.cpp
2021/10/08 06:14	170,726 p3sort.cpp
2021/11/08 16:21	223,947 p4sort.cpp
2021/09/04 17:50	226,479 p5sort.cpp
2021/08/08 16:27	212,865 p6sort.cpp
2020/11/15 08:18	78,730 p7sort.cpp
2021/07/29 14:02	5,886 pmsort.cpp
2021/08/09 14:44	245,961 psort.cpp
2021/04/30 11:09	29,778 ransu.cpp
2006/02/15 21:41	2,402 ReadMe.txt

2021/11/01 18:43	<DIR>	res
2006/02/15 21:41	375	Resource.h
2021/11/10 05:35	10,352	rhasshu.cpp
2016/10/13 06:32	73,728	r_ems.dat
2010/09/11 11:20	4,201	s11_map.cpp
2021/04/30 11:09	843	sahasshu.cpp
2010/09/15 17:03	583	sakujo_y.cpp
2021/10/26 09:29	184,297	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
2021/08/10 06:02	41,521	semeru.h
2010/09/07 09:44	2,315	seme_h.cpp
2021/10/28 10:14	2,597	seme_r.cpp
2020/06/06 11:29	11,279	Senpo_g.cpp
2020/06/06 11:27	11,532	Senpo_s.cpp
2005/01/29 06:20	4,847	Sente.h
2011/01/04 09:08	13,598	Sfttotal.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	Sicttotal.cpp
2021/11/12 05:58	349,883	Siken.h
2007/06/04 08:24	386	Siken_g.cpp
2007/06/04 08:24	385	Siken_s.cpp
2021/12/25 06:52	13,375	sikou.txt

2021/12/24 17:28	161,972 sikou11.cpp
2010/10/07 13:03	1,390 sikou11a.cpp
2021/10/24 17:29	912,444 sikou11aa.cpp
2021/07/10 08:54	633,247 sikou11ab.cpp
2021/11/25 11:06	673,596 sikou11ag.cpp
2021/11/12 12:28	685,379 sikou11aj.cpp
2021/11/23 11:41	644,871 sikou11ak.cpp
2021/12/23 12:10	476,486 sikou11ao.cpp
2021/12/23 06:09	826,771 sikou11as.cpp
2021/05/09 17:08	375,828 sikou11at.cpp
2021/10/01 17:34	191,215 sikou11az.cpp
2021/12/03 15:12	676,878 sikou11gin.cpp
2010/10/17 05:56	4,861 sikou11naru.cpp
2014/07/04 16:26	2,392 sikou11oute.cpp
2021/06/11 11:06	7,889 sikou11wz.cpp
2015/03/09 19:10	1,459 sikou11y.cpp
2021/09/29 14:54	685,628 sikou11yb.cpp
2021/12/25 06:53	721,758 sikou11yc.cpp
2021/12/07 10:58	575,684 sikou11yj.cpp
2021/11/30 05:47	843,885 sikou11yk.cpp
2021/11/28 09:07	754,585 sikou11yo.cpp
2021/12/08 09:03	934,932 sikou11ys.cpp
2021/11/12 15:19	456,828 sikou11yt.cpp
2021/12/13 12:32	984,204 sikou11yy.cpp
2021/10/01 13:45	188,152 sikou11yz.cpp
2021/07/12 09:57	805,772 sikou11zb.cpp
2021/12/05 09:38	775,478 sikou11zg.cpp
2021/12/22 09:41	235,832 sikou11zi.cpp

2021/12/04 17:38	860,460 sikou11zj.cpp
2021/11/29 11:40	932,840 sikou11zk.cpp
2021/07/01 06:11	790,238 sikou11zl.cpp
2021/10/17 09:25	720,826 sikou11zo.cpp
2021/12/24 14:53	907,813 sikou11zs.cpp
2021/11/12 16:50	626,069 sikou11zt.cpp
2021/12/04 16:43	930,853 sikou11zz.cpp
2021/10/29 09:48	15,945 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
2021/05/16 14:24	108,869 Sode.h
2021/01/30 06:31	387 Sode_g.cpp
2021/01/30 06:31	395 Sode_s.cpp
2010/09/04 16:31	43,605 softtotal.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
2021/10/28 17:39	28,846 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
2021/06/01 06:15	16,598 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
2021/10/28 10:14	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
2020/07/21 05:52	4,998	uke_r.cpp
2015/05/09 08:34	13,719	ura.cpp
2010/09/15 17:09	628	wrong_y.cpp
2021/11/01 18:43	<DIR>	x64
2020/09/12 05:57	23,660	xana.h
2021/08/06 12:40	86,148	Xhineri.h
2014/03/27 06:31	406	Xhineri_g.cpp
2014/03/27 06:31	407	Xhineri_s.cpp
2020/10/12 08:45	12,728	xinaniwa.h
2013/11/06 05:13	190	xinaniwa_g.cpp
2013/11/06 05:13	189	xinaniwa_s.cpp
2021/08/14 15:33	179,554	Xmukai.h
2018/05/26 06:29	392	Xmukai_g.cpp
2018/05/26 06:29	393	Xmukai_s.cpp
2021/11/21 14:42	262,757	Xnaka.h
2012/11/28 14:00	344	Xnaka_g.cpp
2012/11/28 14:00	345	Xnaka_s.cpp
2021/08/11 15:58	212,590	Xsanken.h
2013/11/18 11:18	398	Xsanken_g.cpp
2013/11/18 11:18	401	Xsanken_s.cpp
2021/11/21 06:20	520,417	Xsiken.h

2007/08/16 06:13	405 Xsiken_g.cpp
2007/08/16 06:13	408 Xsiken_s.cpp
2021/12/18 09:35	508,501 Yagura.h
2021/12/02 09:49	413,897 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
2021/09/29 14:54	936,533 yb1.h
2021/09/29 14:54	854,530 yb2.h
2021/10/12 05:40	559,162 yb3.h
2021/12/24 17:40	952,552 yc1.h
2021/12/24 17:40	891,729 yc2.h
2021/12/24 17:40	866,641 yc3.h
2021/12/24 17:40	890,061 yc4.h
2021/12/24 17:40	275,380 yc5.h
2021/12/07 10:58	944,854 yj1.h
2021/12/18 10:19	912,903 yj2.h
2021/12/07 10:58	881,064 yj3.h
2021/12/07 10:58	129,164 yj4.h
2021/12/07 10:58	927,747 yj5.h
2021/12/07 10:58	900,707 yj6.h
2021/12/07 10:58	855,073 yj7.h
2021/12/07 10:58	894,045 yj8.h
2021/12/07 11:51	675,131 yj9.h
2021/11/22 10:23	937,944 yk1.h
2021/12/24 17:25	922,550 yk2.h
2021/11/30 06:12	908,665 yk3.h

2021/11/22 10:01	892,936 yk4.h
2021/11/30 06:21	889,699 yk5.h
2021/11/22 10:01	96,276 yk6.h
2021/11/22 10:01	897,099 yk7.h
2021/12/24 17:25	974,166 yk8.h
2021/11/15 15:51	961,037 yo1.h
2021/09/17 13:35	936,697 yo2.h
2021/10/27 07:15	925,403 yo3.h
2021/10/09 17:30	921,779 yo4.h
2021/11/28 09:04	719,958 yo5.h
2021/12/08 09:03	95,319 ys0.h
2021/12/08 09:22	957,889 ys1.h
2021/12/08 09:03	894,631 ys2.h
2021/12/08 09:03	875,799 ys3.h
2021/12/08 09:03	131,666 ys4.h
2021/12/08 09:03	886,030 ys5.h
2021/12/08 09:03	882,666 ys6.h
2021/12/08 09:03	887,376 ys7.h
2021/12/08 10:45	298,645 ys8.h
2021/12/12 16:18	320,625 Ysenpo.h
2010/09/15 17:26	621 ysenpo_g.cpp
2021/12/23 09:49	3,126 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
2021/11/12 15:19	982,735 yt1.h
2021/11/12 15:10	928,487 yt2.h
2021/11/12 15:10	430,167 yt3.h
2021/10/27 20:17	57,169 YThreadProc.cpp

2021/06/03 16:33	3,120 yusen.cpp
2010/09/15 16:57	620 yusen_y.cpp
2021/12/23 12:34	965,550 yy0.h
2021/12/10 06:06	944,004 yy1.h
2021/12/10 06:06	986,715 yy2.h
2021/12/10 06:14	955,464 yy3.h
2021/12/10 06:06	568,945 yy4.h
2021/12/10 06:06	956,677 yy5.h
2021/12/10 06:06	959,607 yy6.h
2021/12/10 06:06	959,975 yy7.h
2021/12/10 06:06	575,816 yy8.h
2021/12/10 06:06	956,867 yy9.h
2021/12/14 13:20	931,351 yya.h
2021/12/10 06:06	936,694 yyb.h
2021/12/10 06:06	915,984 yyc.h
2021/12/10 06:06	27,528 yyd.h
2021/12/10 06:06	925,316 yye.h
2021/12/10 10:52	881,276 yyf.h
2021/12/10 06:06	902,867 yyg.h
2021/12/10 06:06	924,693 yyh.h
2021/12/10 06:14	920,739 yyi.h
2021/12/10 06:06	907,326 yyj.h
2021/12/10 06:06	96,723 yyk.h
2021/12/10 06:06	874,053 yyl.h
2021/12/10 06:06	891,513 yym.h
2021/12/14 13:34	1,064,912 yyn.h
2021/11/10 17:10	896,011 yz1.h
2021/11/10 17:52	591,814 yz2.h
2009/03/13 07:28	274 y_zahyo.cpp

2021/10/06 05:44	935,530 zb1.h
2021/07/12 09:51	861,589 zb2.h
2021/10/06 05:55	888,947 zb3.h
2021/12/05 09:24	947,903 zg1.h
2021/12/05 09:38	910,912 zg2.h
2021/12/05 09:24	196,650 zg3.h
2021/12/05 09:24	899,524 zg4.h
2021/12/05 09:38	91,333 zg5.h
2021/12/22 09:03	890,885 zi1.h
2021/12/22 09:03	882,191 zi2.h
2021/12/22 09:03	362,098 zi3.h
2021/12/03 11:45	925,306 zj1.h
2021/11/27 06:33	918,944 zj2.h
2021/11/22 16:23	934,322 zj3.h
2021/11/22 13:59	142,067 zj4.h
2021/11/22 13:59	906,746 zj5.h
2021/11/22 13:59	923,807 zj6.h
2021/11/22 13:59	325,487 zj7.h
2021/11/22 13:59	859,946 zj8.h
2021/11/22 13:59	870,666 zj9.h
2021/12/02 10:03	906,033 zja.h
2021/11/22 13:59	894,925 zjb.h
2021/11/22 13:59	891,163 zjc.h
2021/12/04 17:38	639,142 zjd.h
2021/12/01 06:31	964,512 zk1.h
2021/11/07 09:08	913,562 zk2.h
2021/12/01 11:49	898,975 zk3.h
2021/10/07 05:48	894,243 zk4.h
2021/12/18 16:56	845,702 zk5.h

2021/10/11 08:44	891,563 zk6.h
2021/10/14 11:52	891,797 zk7.h
2021/12/01 11:49	666,923 zk8.h
2021/06/30 17:45	799,520 zl1.h
2021/06/30 17:45	722,686 zl2.h
2021/06/30 17:45	659,739 zl3.h
2021/10/02 12:30	928,540 zo1.h
2021/10/02 12:30	911,655 zo2.h
2021/10/27 11:25	925,830 zo3.h
2021/10/27 11:25	319,707 zo4.h
2021/12/23 17:36	210,010 zs0.h
2021/12/22 15:57	949,501 zs1.h
2021/12/23 17:45	895,253 zs2.h
2021/12/23 06:25	844,076 zs3.h
2021/11/03 13:57	187,660 zs4.h
2021/12/08 16:27	897,322 zs5.h
2021/12/23 17:50	903,340 zs6.h
2021/11/03 13:57	194,682 zs7.h
2021/11/03 13:57	892,363 zs8.h
2021/11/03 13:57	889,982 zs9.h
2021/12/24 15:10	714,980 zsa.h
2010/07/06 10:43	2,495 zsort.cpp
2021/11/12 16:29	945,387 zt1.h
2021/11/12 16:29	923,057 zt2.h
2021/11/12 16:50	273,200 zt3.h
2021/12/04 16:43	936,147 zz0.h
2021/12/05 08:58	994,097 zz1.h
2021/12/04 16:43	991,373 zz2.h

2021/12/04 16:43	976,288 zz3.h
2021/12/07 17:00	430,172 zz4.h
2021/12/04 16:43	988,922 zz5.h
2021/12/04 16:43	922,571 zz6.h
2021/12/07 17:40	905,618 zz7.h
2021/12/04 16:43	915,050 zz8.h
2021/12/04 16:43	964,399 zz9.h
2021/12/04 16:43	954,516 zza.h
2021/12/04 16:43	986,290 zzb.h
2021/12/14 06:26	954,092 zzc.h
2021/12/04 16:43	958,365 zzd.h
2021/12/04 16:43	935,729 zze.h
2021/12/04 16:43	969,293 zzf.h
2021/12/04 16:43	894,380 zzg.h
2021/12/04 16:43	923,352 zzh.h
2021/12/04 16:43	932,245 zzi.h
2021/12/04 16:43	952,584 zzj.h
2021/12/04 16:43	900,193 zzk.h
2021/12/04 16:43	958,319 zzl.h
2021/12/04 16:43	928,045 zzm.h
2021/12/04 16:43	933,085 zzn.h
2021/12/04 16:43	936,538 zzo.h
2021/12/04 16:43	918,090 zzp.h
2021/12/04 16:43	893,978 zzq.h
2021/12/04 16:43	902,500 z zr.h
2021/12/04 16:43	871,353 zzs.h
2021/12/04 16:43	907,648 zzt.h
2021/12/14 12:14	335,964 zzu.h

494 個のファイル 194,583,875 バイト

4 個のディレクトリ 168,728,715,264 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、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。

また、ステップ数が大きいとデバッガーとエディターが異常な動きをします。

例えば、1 ファイルのステップ数が 65536 ステップ数を超すとデバッガーがソースの位置を正しく表示しなくなります。

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くように思えます。

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32回世界コンピュータ将棋選手権】

コンピュータ将棋ソフト
『元気もいもいニンニクパワー』
アピール文



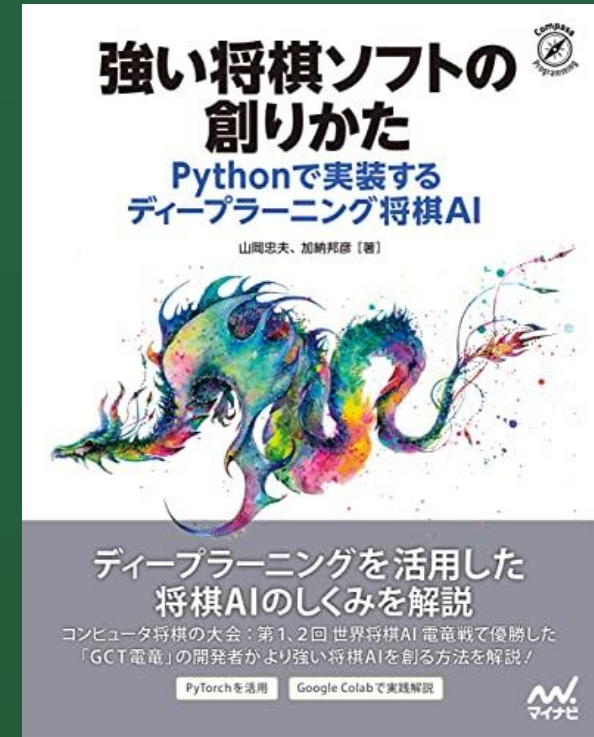
【第32回世界コンピュータ将棋選手権】 『元気もいもいニンニクパワー』アピール文(1)

- ・Vtuber 都賀町えいだが作成したソフト
- ・名前は公募＋アンケートによって命名
名付け親は牛車さん(@gyuusha3)
- ・第2回電竜戦バージョンとはまったく別物

【第32回世界コンピュータ将棋選手権】 『元気もいもいニンニクパワー』アピール文(2)

- ・「強い将棋ソフトの創いかた」
著：山岡忠夫、加納邦彦

これをもとに作成
最終目標は同等の棋力になること



【第32回世界コンピュータ将棋選手権】

『元気もいもいニンニクパワー』アピール文(3)

- ・問題はpython、C++があまり分からない
→ 知っている言語(C#)で作成する
- ・C#なら分かる人向けに作成してみたかった
(私自身がそれで困っていた)
すべてC#でなく、pythonも多少残る

【第32回世界コンピュータ将棋選手権】

『元気もいもいニンニクパワー』アピール文(4)

- ・ライフライ「cshogi」は使用しない
合法手生成部分をちゃんと自作したかった
- ・フロムスクラッチ申請はしません
書籍をC#化しているだけなので、
フロムスクラッチとは言い難い

【第32回世界コンピュータ将棋選手権】

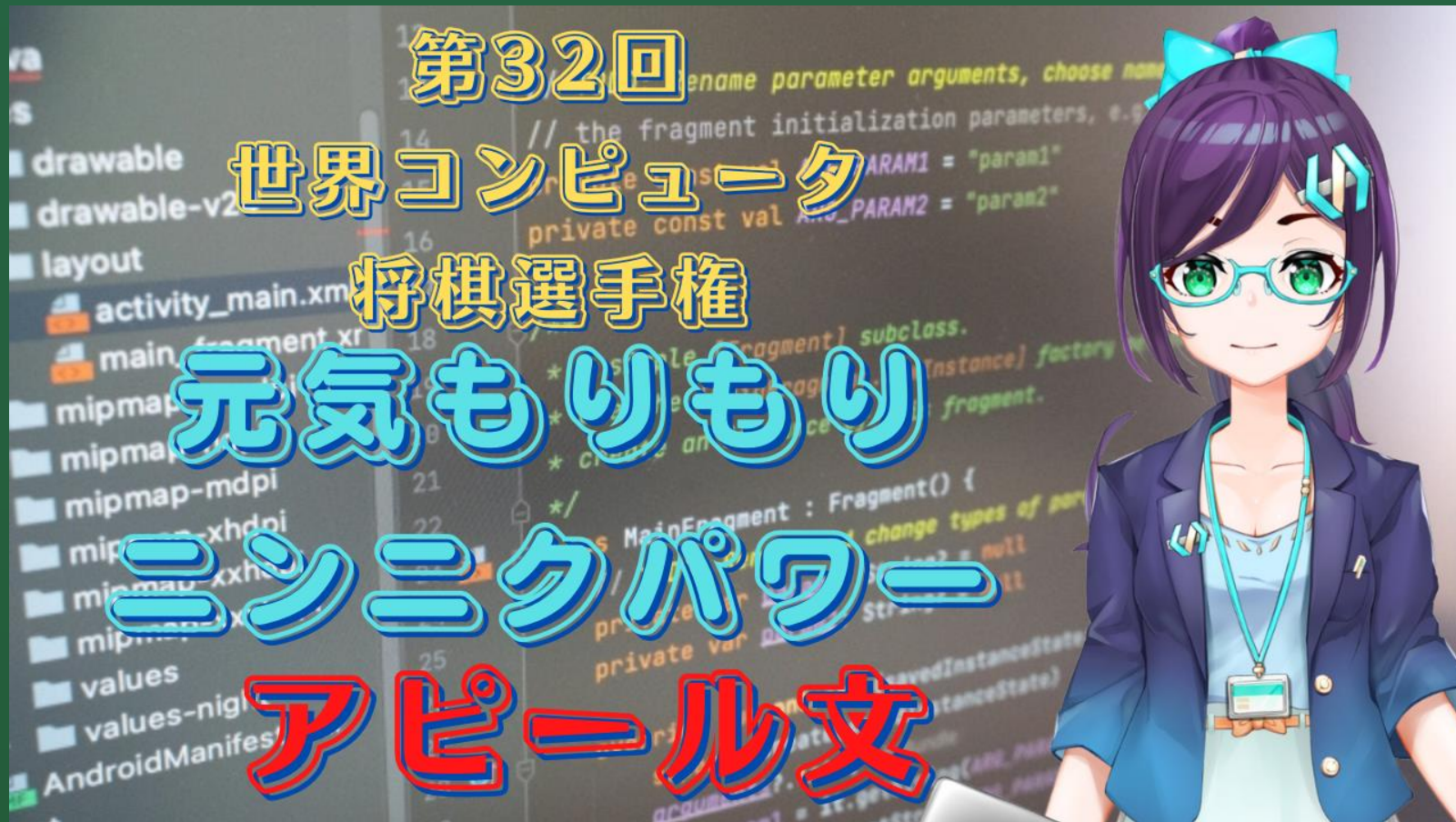
『元気もいもいニンニクパワー』アピール文(5)

・2022/03/31時点

合法手生成部が完成していないので
全然動きません

当日までには動くようにがんばります





動画版はコチラ

<https://youtu.be/dyB-uvXWY3M>

prelude アピール文書

谷合廣紀, 中村朋生

2022 年 3 月 28 日

1 独自の工夫

基本は dlshogi/ふかうら王などのいわゆる DL 系で採用されている、policy+value Network + MCST のアプローチを取っています。dlshogi/ふかうら王と大きく異なるのは、モデル構造とその入出力です。

1.1 モデル入力のエンコード

モデルへの入力を画像的エンコード (9x9xfeatures) ではなく、文字列的エンコード (95xfeatures) としています。具体的には 1 一、1 二... 9 九の駒と先後の持ち駒を並べた 95 字を、自然言語処理と同様の枠組みで処理しています。基本的なアイデア・モチベーションは拙作の bert-mcts と同じで、広域的な相関を浅い層のうちから取ることがひとつの狙いです。

1.2 モデル構造

扱うデータが画像的な 2 次元ではなく 1 次元のため、MLP のみで構成した独自のモデルを構成しています。

1.3 モデル出力

dlshogi/ふかうら王で採用されている policy の出力と比べて、構造的かつ無駄のないラベルエンコーディングを導入して、学習が進みやすいように工夫しています。value 側では、損失関数に互角局面に近いほど重みを大きくするなどの工夫を施しています。

2 使用ライブラリ・使用データ

- ふかうら王
- 「強い将棋ソフトの創り方」公開データ

2022.3.31

神田 剛志

■開発コンセプト

名前の通り軽く速く。（末尾のEFはDNNのモデル(EfficientNet)が由来です）
ローカルPC環境でもCPU系/GPU系に限らず、つよつよインスタンス勢や
高級スリッパと戦うことがLightweightシリーズの目的です。

■アピールポイント

・ DNNモデルの改良

本家のResNetをEfficientNetで再構築し、1 から学習しなおしました。
第2回電竜戦時のモデルから6層追加した改良型に当たります。

・ USIエンジンのパラメータ設定変更によるNPS向上

GPUに局面を渡す際のバッチサイズを1024に上げています。
これと軽量なモデルと組み合わせにより、ローカルPC環境での平均NPSを向上
させています。

・ GCT学習データによる教師あり学習とLightweight-EF自身による強化学習

dlshogiチームが公開してくださっている学習データとLightweight-EFの
自己対局データを用いた強化学習を実施しています。
またこの際、学習時のバッチサイズをあげる事で学習の安定化を図っています。
（学習データを無償で公開されている山岡さん、加納さんには感謝申し上げます）

・ UCB選択アルゴリズムの一部変更（NPS向上のための簡易枝刈りの実施）

PUCTアルゴリズムに従って探索木を降りていく際、各子ノードの着手確率を
利用した簡易的な枝刈りを実施することで、最大UCB値の子ノード選択処理に
かかる時間を短縮しています。

・ 定跡の使用

初期局面の事前探索結果を利用することで、持ち時間の消費を抑えます。

■使用ライブラリ

dlshogi：自己対局データ生成・探索部・モデル学習・定跡作成に利用
Gikou2 ：検証時のテスト対局に使用
Suisho3：検証時のテスト対局に使用
Suisho4：検証時のテスト対局に使用
elmo for learn：学習データ作成に利用

第 32 回 世界コンピュータ将棋選手権

TeamShiny アピール文書

湯川和雄 近藤俊彦 田之畑諒

2022 年 3 月 31 日

【概要】

「将棋 AI で学ぶディープラーニング」「強い将棋ソフトの創り方」の著書を参考にさせていただきました。

探索部と思考部は「強い将棋ソフトの創り方」第 5 章 6 章に記載されている手法を使用させていただいております。

【独自の工夫】

振り飛車ソフトにしたかったので、初手に「先手なら 7 六歩」「後手なら 3 四歩」を着手するようにプログラムで決めています。合わせて、定跡を登録して出来るだけ振り飛車の進行で着手するようにしました。

序盤を有利に進められるように定跡の登録を増やしていこうと試みています。

floodgate 内の棋譜から振り飛車勝利の棋譜を多めに集めて学習しています。

【学習環境】

ディープラーニングのモデルの学習は、GoogleColabPro+で学習しています。

【データ】

主に floodgate 内の棋譜を使用させていただきました。

約 100 万件の棋譜を学習に使用予定です。（振り飛車勝利の棋譜約 25 万件）

時間があれば強化学習による棋譜生成や追加学習などを考えています。

【開発言語】 Python

【開発環境】 GoogleColabPro/Pro+

【実行環境】 将棋所、Anaconda

【使用・参考にしたライブラリ】pytorch、python-shogi、cshogi[1]、dlshogi[2]、
dlshogi2[3]

【特徴】 ディープラーニング系ソフト、振り飛車ソフト

[1]cshogi : <https://github.com/TadaoYamaoka/cshogi>

[2]dlshogi : <https://github.com/TadaoYamaoka/python-dlshogi>

[3]dlshogi2 : <https://github.com/TadaoYamaoka/python-dlshogi2>

【自己紹介、意気込み】 2022/03/30 湯川和雄

将棋は、アマチュア三段です。

現在、AI やディープラーニングの勉強をしています。AI の勉強を始める前は、プログラミングの勉強をしながら個人で将棋 GUI 開発^[4]や将棋アプリ開発^{[5][6]}をしていました。

約 3 年前にプログラミングの勉強を始めました。その時の最初の夢が、「将棋アプリ開発」「世界コンピュータ将棋選手権に参加」をすることでした。

約 15 年前にネット将棋の対局を通じて、本大会第 1 回参加者の森田将棋の森田さんに 10 局ほど将棋を教えていただきました。また、チャットでコンピュータ将棋や AI についてお話させていただいたのを覚えています。その時は、この大会に参加することができるとは思いもしませんでした。

夢が叶う場所までこれて、とても嬉しく思います。

初参加ですが、大会の雰囲気を感じて楽しめればと思います。

よろしくお願いいたします。

[4]JavaScript 将棋 GUI : <https://github.com/baron2020/js-shogi>

[5]将棋研究アプリ : <https://play.google.com/store/apps/details?id=com.baron.syougiarrangementgamerecord>

[6]将棋 AI アプリ : <https://play.google.com/store/apps/details?id=com.teamshiny.shogiai>

【自己紹介、意気込み】 2022/03/30 近藤俊彦

チームシャイニーという施設のスタッフをしています近藤俊彦です。

チームシャイニー^[7]は先端 IT 特化型の就労移行支援ということで、主に AI について勉強をする人が集まっている施設です。様々な才能を持った人がおり日々切磋琢磨しています。そんなわけで将棋に興味を持つ人も少なからずおり、今回参加する運びとなりました。

個人的には将棋の腕前は2級で全然たいしたことはないですが、対局を見るのは好きな見る将です。そんなわけで将棋大会は雲の上の存在かと思っていましたが、このたび参加できる機会をいただきとても感謝しています。参加する際はまずは楽しみたいと思います。

[7]チームシャイニー：<https://team-shiny.org/>

【自己紹介、意気込み】2022/03/31 田之畑諒

同じくチームシャイニーのスタッフをしています、田之畑諒です。

私自身、昔から AI に興味があり、また、ご縁もありチームシャイニーで働かせて頂いております。

将棋は全く打ちませんが、近年 AI ブームでの将棋 AI にも非常に強い関心を持っています。今回、サポートとして参加させて頂ける事に非常に感謝しています。

私自身、見る将でして、YouTube で将棋を見ている内に一手一手の意図が少しですが分かるようになって来たかな？という程度です。

32 年と長い歴史のある大会に参加できる事を嬉しく、またワクワクしております。

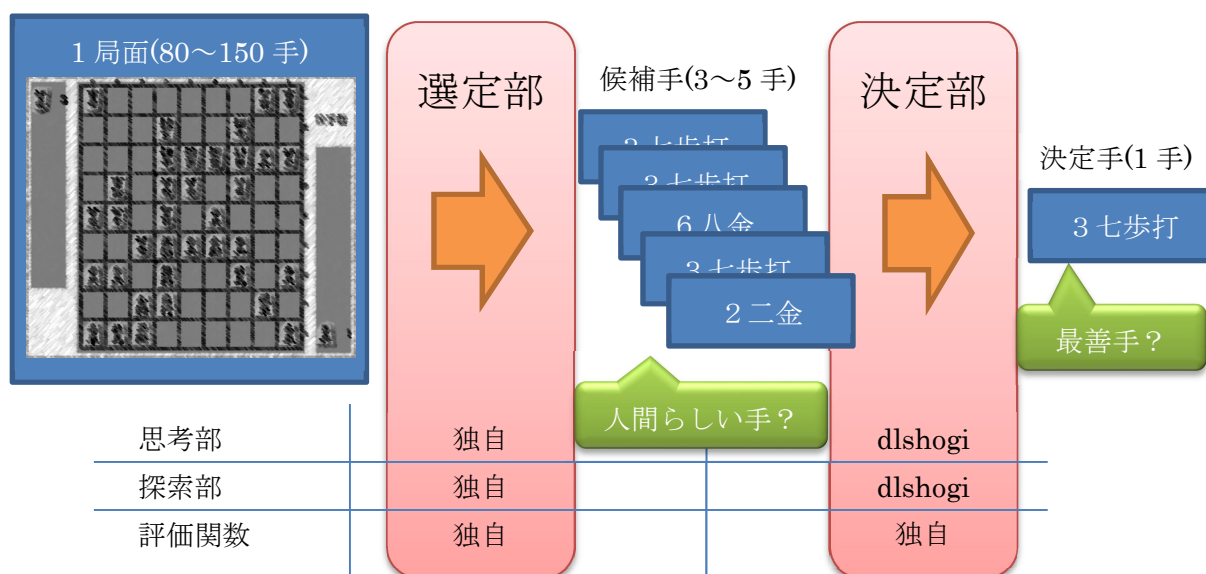
よろしくお願い致します。

コンセプト

人間が指しそうな手を複数候補に挙げ、その中から DL 評価関数を元に最有力な手を選ぶことにより、『人間っぽいけど強い』ソフトが出来るかを検証します。

1 思考部

1 局面から有力な複数候補を選定する部(以下、選定部)、および、選定部で選定した候補手の中から最善手を決定する部(以下、決定部)の 2 段階構成とします。今後決定部の入れ替え等も柔軟に行えるようにするため、選定部から決定部へのインタフェースは USI(Universal Shogi Interface)プロトコルを使用しています。



1.1 選定部(独自)

選定部では、人間のアマチュア棋士が指しそうな手を 3~5 手ほど候補に挙げられるような仕様を目指しました。評価関数は人間の感覚で指しやすい手をスコア化(※)し、手作業で作成しました。

可変 α β 法を採用 局面・残り時間・候補手の数から思考時間を見積り、深さ上限を 4~8 の値に設定して探索します。

また 2 段階の IDDFS を採用し、2 手目以降が同じ手は一つにまとめることにより、探索量を簡略化します。

スコアが高かった候補手は決定部に渡します。候補手が 1 手のみとなった場合は決定部には渡さず、そのまま最善手に確定します。

(※) 飛車や角道を開ける、垂れ歩を作る、両取りを仕掛ける、歩を突き捨てる等の人間がよく思考する手を独断と偏見でスコア化

1.2 決定部(dlshogi ライブラリ使用)

選定部で選定された候補手 3~5 手を dlshogi を使用して評価を行い、一番評価が高い手を最善手に確定します。評価関数は floodgate の 2018 年~2021 年のデータをディープラーニング したうえでその後は強化学習を行う予定です。

2 定跡部(独自)

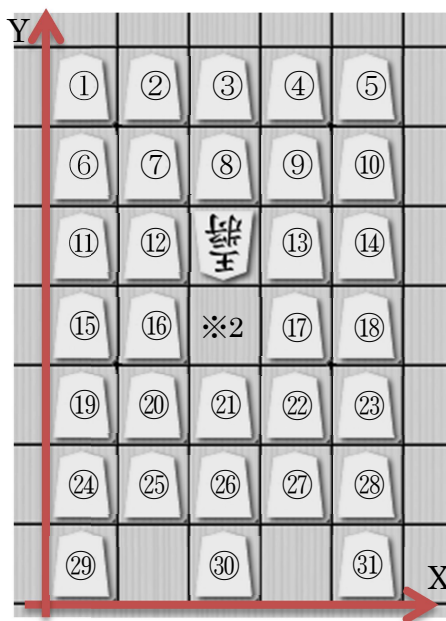
独自実装しました。過去の棋譜(プロ定跡 約 1.2 万)の 50 手までの局面をハッシュ化しました。また、本将棋 AI が得意な戦法が候補として挙がりやすくなるようチューニング済みです。

3 詰将棋部(独自)

独自実装しました。ソースコードは 1800 行程度となりました。

3.1 候補手選定方法の改良

従来は全ての駒を動かし、動かした手が王手かをチェックするプロセスのため、無駄が多かったです。今回は逆に相手王を中心に下図のようなマップ情報を用意し、マップに合致する座標に味方の駒がある場合、あらかじめ決まった手を選定する形式としました。



【各座標(①～⑩)に特定の駒がある場合の王手候補手の例】

①銀の場合 $\rightarrow X+1, Y-1$, 不成

②銀の場合→ $X+1, Y-1$, 成

③銀の場合→ $X+1, Y-1$, 不成 および $X-1, Y-1$, 不成

金(※1)の場合→ $X \pm 0, Y - 1$, 不成

: (中略)

②②歩の場合 $\rightarrow X \pm 0, Y + 1$, 成

銀の場合 $\rightarrow X-1, Y+1$, 不成 および $X-1, Y+1$, 成

および $X \pm 0, Y+1$, 不成 および $X \pm 0, Y+1$, 成

金(※1)の場合 $\rightarrow X-1, Y+1$, 不成 および $X \pm 0, Y+1$, 不成

: (中略)

③⑩桂馬の場合 $\rightarrow X-1, Y+2$, 不成 および $X+1, Y+2$, 不成

③①桂馬の場合 $\rightarrow X-1, Y+2$, 不成

※1 金と同等の動きをする各種成り駒も含む

※2 敵将前の位置に駒があることはありえない想定(すでに王手状態となっているため)

また、飛車角香車や駒打ちについてもそれぞれ固有のマップテーブル情報を用意することで、高速化を目指しました。独自計測ですが従来の全ての手から王手となっている手を洗い出す方式より 20%ほど速度向上が見込まれます。

王手逃れ側も同様にマップ情報を持っています。

3.2 候補手の優先度付け

人間が指しやすいと思われる手(頭金を打つ、王手の駒を取る等)を優先度高として

制限時間を設けることで、優先度の低い王手をスキップする、いわゆる“読み抜け”を再現します。

4 今後の目標

昨年後半から始めたプロジェクトのため、粗削りが多い部分も多く、改善の余地が多数あります。将来的には対人間向け練習ソフトとしての地位が確立できればと思います。

4.1 選定部評価関数の機械学習化

今回は時間の都合上、手動で評価部を作成しましたが限界があるため、より人間らしい手が選定できるように機械学習を取り入れたいです。

4.2 決定部の独自実装

今回は時間の都合上、`dlshogi` を使用させて頂いておりますが、機械学習をより深く習得し独自の評価関数作成アルゴリズムを提唱出来ればと思います。

4.3 プログラミング言語の選定

今回は時間の都合上、一番知識のある `C#` を使用しましたが、やはり処理速度では `C++` に数段劣るため、次回開発までには `C++` により速度を向上したいです。

以上

爆裂駒捨太郎R アピール文書

みなさん

詰将棋
やってますか？

大事なことなので
もう一度

詰将棋
やってますか？

詰将棋はすべてを解決します

詰将棋意味あります

爆裂駒捨太郎Rが
証明しましょう

目次

1. 作者紹介
2. 爆裂駒捨太郎Rについて
 1. 主な使用ライブラリ
 2. 定跡部
 3. 探索部
 4. 評価関数部

作者紹介

- ◇ Twitter : ブロコロン(@burokoron_)
- ◇ GitHub : Yolts(<https://github.com/burokoron/Yolts>)
 - ◇ 開発中の将棋エンジンリポジトリ
 - ◇ Yolts : You only look tume-shogi
- ◇ 棋力 : 将棋ウォーズ3段
 - ◇ 得意囲い : 船囲い
 - ◇ 得意戦法 : 右玉
 - ◇ NHK杯鑑賞のみで2級到達後、将棋ソフトとの局后感想戦のみで3段到達
 - ◇ 詰将棋は全くやっていない

目次

1. 作者紹介
2. 爆裂駒捨太郎Rについて
 1. 主な使用ライブラリ
 2. 定跡部
 3. 探索部
 4. 評価関数部

主な使用ライブラリ

◇ Python

- ◇ 評価関数の作成に使用

◇ Rust

- ◇ 対局エンジンの作成に使用

◇ cshogi[1], shogi-rs[2]

- ◇ Pythonで利用できる将棋ライブラリ
- ◇ 指し手生成、局面管理に使用

[1] cshogiのURL : <https://github.com/TadaoYamaoka/cshogi>

[2] shogi-rsのURL : <https://github.com/nozaq/shogi-rs>

目次

1. 作者紹介
2. 爆裂駒捨太郎Rについて
 1. 主な使用ライブラリ
 2. 定跡部
 3. 探索部
 4. 評価関数部

定跡部

◇ 作る予定

◇ 2022/03/29時点では全く作っていません

目次

1. 作者紹介
2. 爆裂駒捨太郎Rについて
 1. 主な使用ライブラリ
 2. 定跡部
 3. 探索部
 4. 評価関数部

探索部

◇ 探索アルゴリズム

- ◇ $\alpha\beta$ 探索+王手1手延長+1手詰め探索
- ◇ 反復深化探索

◇ ムーブオーダリング

- ◇ 各局面で評価値の良かった手から探索する
- ◇ 各探索深さで評価値の良かった着手位置の手から探索する

◇ 枝狩り

- ◇ Null Move Pruning
- ◇ Mate Distance Pruning

目次

1. 作者紹介
2. 爆裂駒捨太郎Rについて
 1. 主な使用ライブラリ
 2. 定跡部
 3. 探索部
 4. 評価関数部

評価関数部(1/2)

◇ 方針

- ◇ やねうらお氏公開の詰将棋問題集500万問[3]を使用
- ◇ 機械学習を使用しない
- ◇ 出現率の低い特徴量は価値が高いと仮定して評価関数を作成する

◇ 特徴量

- ◇ ある駒がどのマスにいるか(1駒関係)
- ◇ ある駒が持ち駒として何枚あるか(1歩×2 = 2歩ではない)

[3] 詰将棋問題集500万問のURL : <https://yaneuraou.yaneu.com/2020/12/25/christmas-present/>

評価関数部(2/2)

◇ パラメータの調整

1. 盤面、持ち駒それぞれで各特微量の出現率を計算する
2. 盤面、持ち駒それぞれで上記出現率を最小0、最大1に正規化する
3. 各特微量値 =
$$\begin{cases} \log_2(\text{上記出現率}) & \text{上記出現率} > 0 \\ 0 & \text{上記出現率} = 0 \end{cases}$$
4. 盤面は81マス、持ち駒は14マス(先後駒種)より盤面特微量値を $\frac{14}{81}$ する
5. [3]において勝ち(負け)が多い上記特微量の値は正(負)とする

NNUE評価関数と定跡の極北を目指しています。

- ・従来の強化学習手法に加え独自のNNUE評価関数の強化
- ・昨年優勝のelmoとほぼ同じ手作業による定跡の生成。特に後手番の定跡の強化に主眼を置いています
- ・探索部はやねうら王、いわゆるやねうら王チルドレンです。やねうら王+最新のStockfishのキャッチアップを予定

よろしくお願いいたします。

・役割論理とは

コンピュータ将棋においてついさっき確立された論理ですなwww

定跡とNNUE型評価関数と高級スリッパの圧倒的火力によって評価値ダメージレースを制する必勝の戦術ですぞwww

・定跡について

現状はほぼ先手必勝ですなwwwもちろんS振りではありませんぞwww

千日手模様になりやすい角換わりより青野流の方が深堀りされているようですなwww

独自の千日手回避手段を用いる場合もあるようですなwww

まずは後手番でどこまでダメージを最小限にするかが重要ですぞwww

互角の分かれならあとはNNUE型評価関数の終盤の強さと高級スリッパの超火力で踏み潰すだけですなwww

現状は後手番ではs-book_blackに少し分が悪いですぞwww

もちろん定跡を整備しないとDL系や水〇、や〇うら王などの強豪には手も足も出ませんなwww

相手が強豪で後手番ならもう千日手で逃げてしまう手もありますぞwww千日手専用定跡wwwありえないwww

そもそもプロ棋士でもないのに何でこんなことをする必要があるんですかなwww

・評価関数について

最低限水匠5より強くないと話になりませんなwwwただし評価関数の相性もあるので難しいですぞwww

もちろん振り飛車評価関数は総合的にロジックするまでもなくありえないwww

・必然力とは

論者を圧倒的勝利に押し上げる力ですなwww

絶対 に 先手を引く、スイス式の当たりが良い、裏街道最高wwwなどは必然力とされていますなwww

ヤーティ神への信仰によって得られる加護とされていますぞwww

現状これが最も重要なファクターとなっていますなwww

・ムックとは

ロジカル語法を使うもののヤーティを使わず、役割論理という戦術を理解・実践しない者を指しますなwww

つまりこのアピール文そのものですなwww完全に異教徒ですなwww

・さいごに

結論としては「評価関数の強化」と「定跡の強化」というごく当たり前の結論に至る訳ですなwww

最低限s-book_black対策くらいはしておきたいところですが難しいですなwww