

# 名人を超えるコンピュータ将棋

2013年8月  
伊藤 英紀

# 目次

---

- **コンピュータ将棋概観**
- **コンピュータ将棋の基礎技術**
- **機械学習**
- **並列処理**
- **ボンクラーズ/Puella  $\alpha$ の概要**
- **将棋の後の人工知能**

# 自己紹介

---

- 1988 富士通（株）入社。以来、CPU設計、  
半導体製造のサポート・マーケティングに従事
- 1998 趣味でコンピュータ将棋の開発を始める
- 2011 世界コンピュータ将棋選手権優勝  
（株）富士通研究所に異動。人工知能の研究に従事

# コンピュータ将棋とは

---

- コンピュータが「思考」して指し手を決定する、人工知能の一分野
- 必要な情報は、決められたプロトコルにしたがって入出力される
  - 盤面の画像認識や、駒を動かすロボット制御等は不要

# コンピュータ将棋の歴史

---

- 1970年代 初めてプログラムが作成される 初心者レベル
- 1980年代 商用プログラムが出はじめる ( 森田将棋、柿木将棋等 ) 級位者下位レベル
- 2007年 ボナンザが渡辺竜王と対戦、敗北 アマチュア強豪レベル
- 2010年 あからが清水女流王将に勝つ プロ下位レベル
- 2012年 ボンクラーズが米長永世棋聖に勝つ 名人レベル
- 2013年 コンピュータ対プロ棋士の団体戦で3勝1敗1分。名人を超えた？

# 世界コンピュータ将棋選手権

---

- 1990年から毎年開催（大体GW）
- マシンは制限なし。リモート参加も可
- 11年 優勝 ボンクラーズ 準優勝 Bonanza
- 12年 優勝 GPS将棋 準優勝 Puella  $\alpha$  (ボンクラーズから改名)

# 人間の棋士との対局



○ボンクラーズ — ●米長永世棋聖  
第一回将棋電王戦(2012.1.14)

## 近年の棋力向上の主な要因

---

- コンピュータチェス由来の探索技術(含スレッド並列)
- 機械学習による評価関数の精緻化
- クラスタ並列



# コンピュータ将棋の基礎技術

---

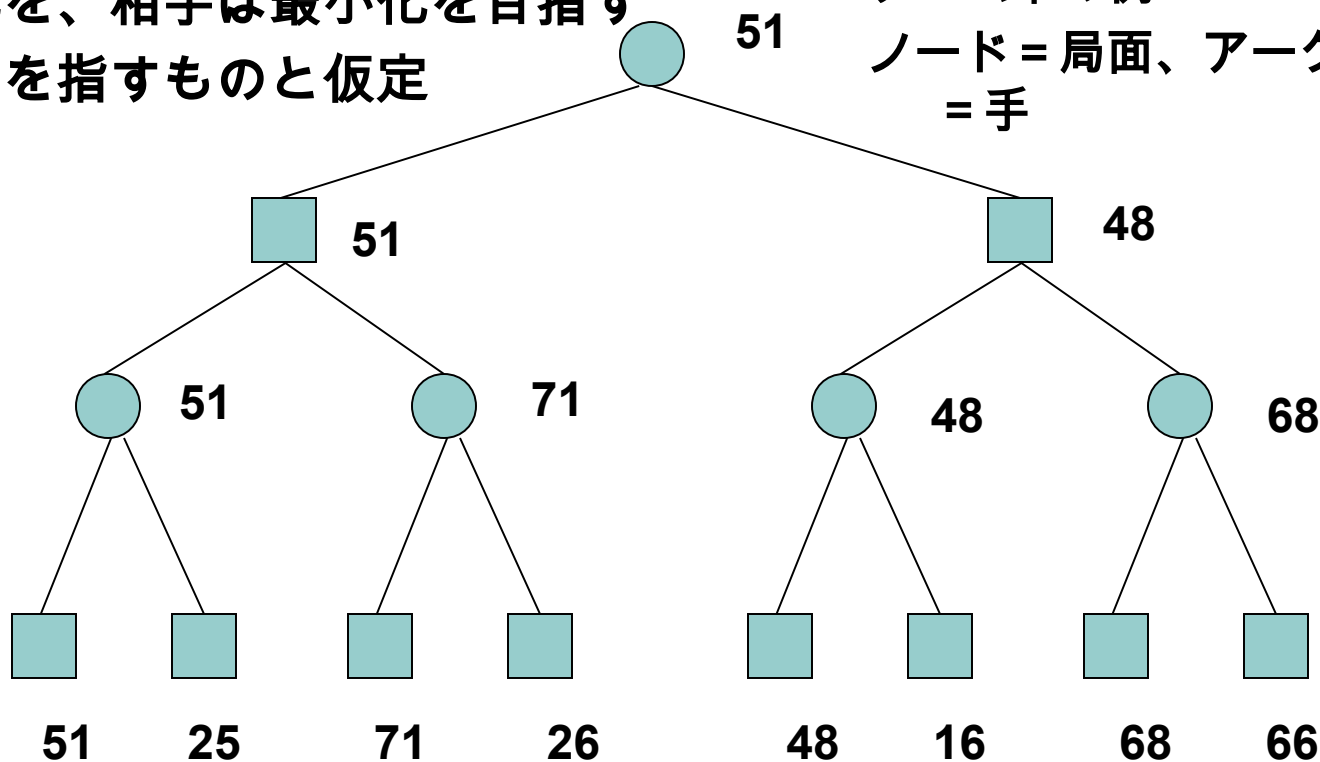
- **Minimax探索**
- **$\alpha\beta$ 探索**
- **評価関数と機械学習**
- **反復深化**
- **遷移表 ( ハッシュ )**
- **Scout探索**
- **Null Move Pruning**
- **Late Move Reduction**
- **探索延長**

# 基礎技術(1) Minimax探索

自分は最大化を、相手は最小化を目指す  
相手は最善手を指すものと仮定

ゲーム木の例  
ノード = 局面、アーク = 手

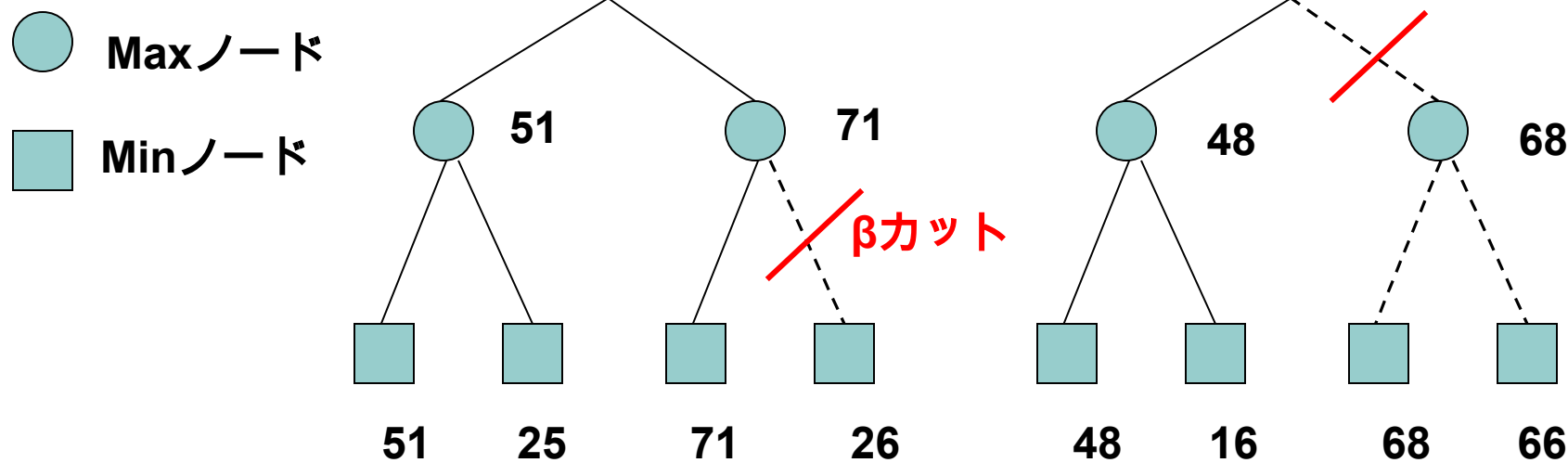
● Maxノード  
■ Minノード



点が高い → Max側が形勢有利

## 基礎技術(2) $\alpha\beta$ 探索

自分は $\alpha$ 以上を、相手は $\beta$ 以下を確保した時、その外側になるなら無視できる



( $\alpha$ 、 $\beta$ ) 幅が狭いほど高効率 → 「良い手」から先に探索すべし

## 基礎技術(2) $\alpha\beta$ 探索 ( つづき )

---

- 深さの設定
  - 典型的には、あらかじめ決められた深さでストップ。そこで何らかの「評価関数」( 形勢判断 ) で局面に得点をつける
  - 状況に応じて深さを変えることも
    - よさそうな手は深く、悪そうな手は浅く  
手ごとに読む量が変わる

## 基礎技術(3) 評価関数と機械学習

- 古典的な評価関数

直観的にわかりやすい数百個程度のパラメタを手調整

古典的な評価の例：駒割

	歩	香	桂	銀	金	角	飛
基本価値	100	430	450	640	690	890	1040
駒が成る価値	320	200	190	30		260	260
持駒の付加価値	15	50	60	80	90	220	220

YSS 1997

$\Sigma$ 先手各駒の和 -  $\Sigma$ 後手各駒の和

## 基礎技術(3) 評価関数と機械学習 ( つづき )

古典的な評価の例：玉との相対位置による得点 ( 後手金 )

	0	1	2	3	4	5	6	7	8
8	50	50	50	50	50	50	50	50	50
7	50	50	50	50	50	50	50	50	50
6	62	60	58	52	50	50	50	50	50
5	80	78	72	67	55	51	50	50	50
4	100	99	95	87	78	69	50	50	50
3	140	130	110	100	95	75	54	50	50
2	170	160	142	114	98	80	62	55	50
1	170	165	150	121	94	78	58	52	50
0	*	145	137	115	91	75	57	50	50
-1	132	132	129	102	84	71	51	50	50
-2	100	97	95	85	70	62	50	50	50
-3	90	85	80	68	60	53	50	50	50
-4	70	66	62	55	52	50	50	50	50
-5	54	53	51	50	50	50	50	50	50
-6	50	50	50	50	50	50	50	50	50
-7	50	50	50	50	50	50	50	50	50
-8	50	50	50	50	50	50	50	50	50

X軸については対称

YSS 1997

## 基礎技術(3) 評価関数と機械学習 ( 続き )

---

- **2006年 Bonanza登場**
  - 約2万個のパラメタを機械学習で自動調整
  - 形勢判断の精度が飛躍的に向上
  - 大会に初出場でいきなり優勝



その後多くのソフトが機械学習を採用  
現在は主流に

## 機械学習の説明（簡単化バージョン）

- 評価関数が次の形だとする

$$E = ax + by + cz + \dots - 1$$

- $x$  :  なら 1 else 0

$a$  : その価値

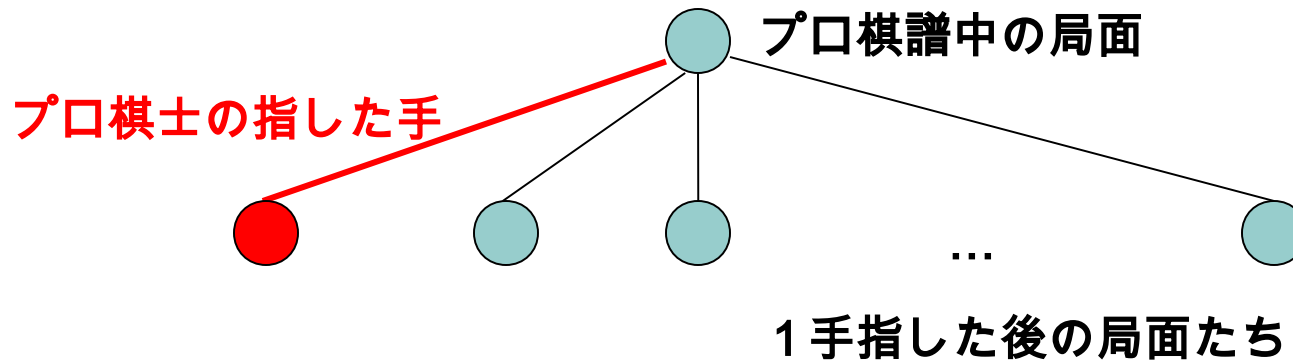
- $y$  :  なら 1 else 0

$b$  : その価値

$x/y/z/..$  は局面ごとに異なる（変数） ※ 「特徴」という  
 $a/b/c/..$  は共通（パラメタ） → これを決めたい



# プロの棋譜を参考にする



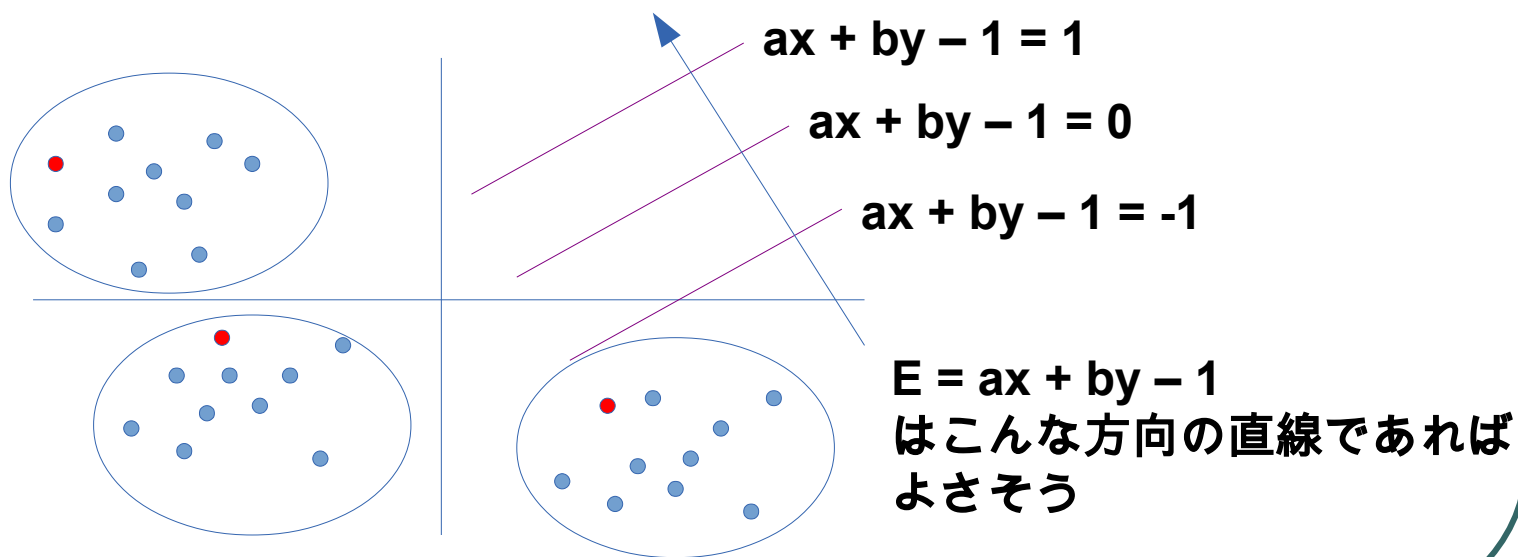
特徴

x	1	0	1	0
y	0	1	0	0
z	0	0	1	1
:				

このときに評価が高くなるようにa/b/c/.. を決めたい

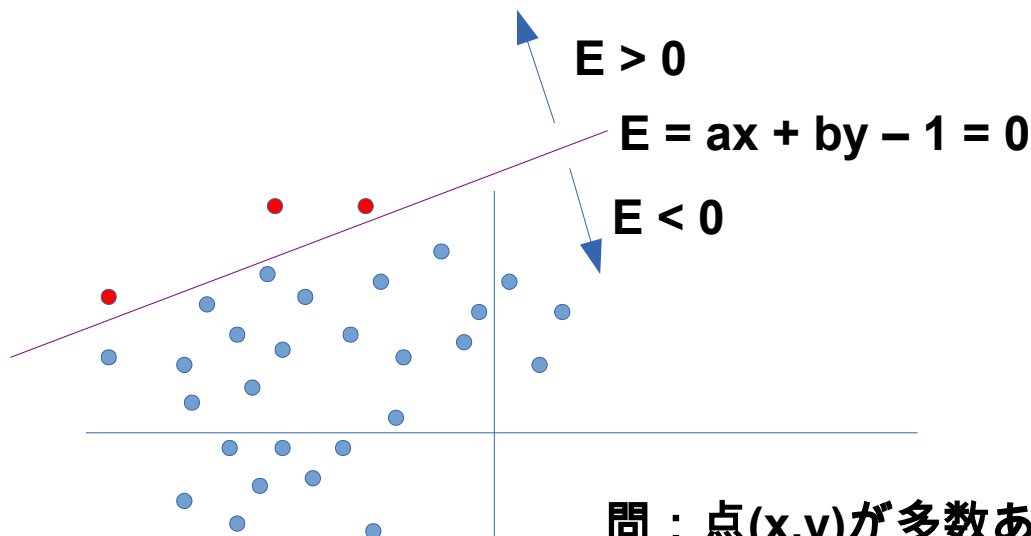
# 少し違う問題を考えてみる

	元の問題	考える問題
次元数	数万～数千万	2
変数定義域	0 or 1	実数( $-\infty \sim +\infty$ )



# もう一段、問題を簡単化

元の問題	考える問題
局面ごとに正解一つ	全部いっしょくた



問：点(x,y)が多数あり、その各々がOKかNGかわかっているとす。これらの点の情報から、「OK  $\leftrightarrow$   $E > 0$ 」となるa,bを決定せよ。

## 解法 ( の一つ )

---

- 最初に、適当な初期値を決める  
 $a = -6, b = 9$  等 ( ランダムでよい )
- 点(x,y)を一つずつ見ていく。OKをNGと誤答する ( $E < 0$ だがOK)点があれば、Eが大きくなるようにa,bを少し変える

例:(6,3)がOKだが  $E = -6 \cdot 6 + 9 \cdot 3 - 1 < 0$

→ a: -6 → -5.8, b: 9 → 9.1 などとする

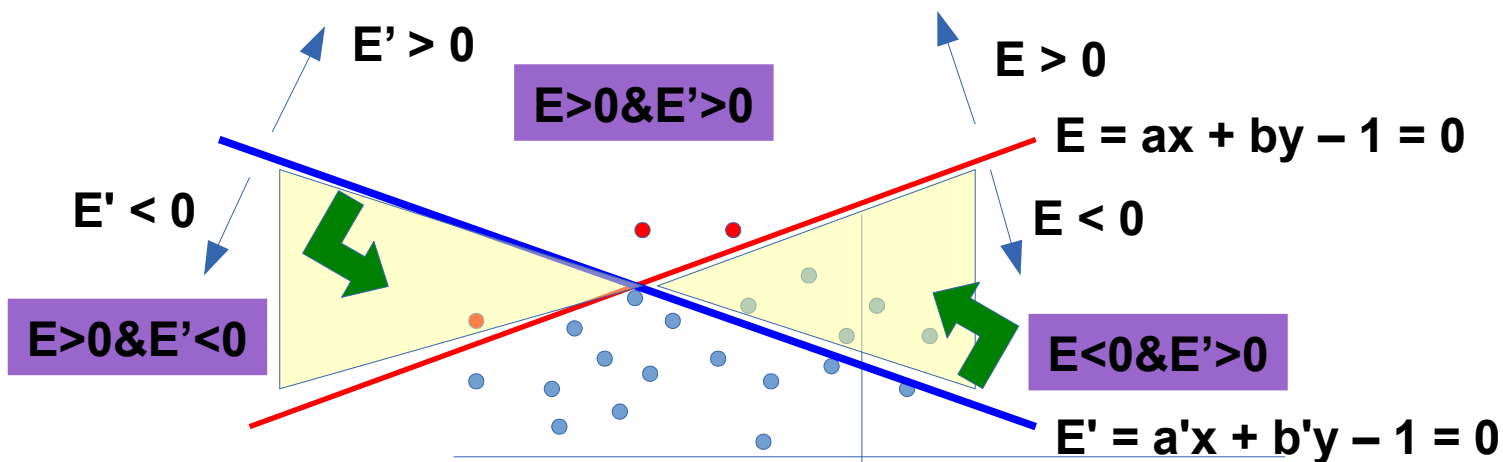
(a,b)を動かす方向は「単位長あたり、Eの増加が最大」に→ (x,y)と同方向

- 逆の場合( $E > 0$ だがNG)はEを小さくする
- 正答( $E > 0$ &OK,  $E < 0$ &NG)ならそのまま

# 要するに何をやってるの？

E : 正しい評価関数

E' : 現在の ( 仮の ) 評価関数



誤答 ( 薄黄 ) を見ると、仮の直線 ( 青 ) を少しずつ緑矢印の方に動かし、正しい直線 ( 赤 ) に近づける

# プログラム実行結果

---

- right answer: a=-2.000 b=1.000
- i= 0: a=-6.000 b=9.000
- i= 20000: a=-8.240 b=4.269
- i= 40000: a=-7.422 b=3.570
- i= 60000: a=-6.426 b=3.253
- i= 80000: a=-5.727 b=2.598
- i=100000: a=-4.834 b=2.353
- i=120000: a=-4.084 b=2.081
- i=140000: a=-3.498 b=1.778
- i=160000: a=-2.953 b=1.398
- i=180000: a=-2.514 b=1.329
- i=200000: a=-2.348 b=1.090
- i=220000: a=-2.125 b=1.069
- i=240000: a=-2.064 b=1.053

多数回繰り返すと正解に  
近づいていく

元の問題も本質的に類似の  
手法で解ける

# 並列処理

---

- スレッド並列とクラスタ並列
- 並列化の困難さ

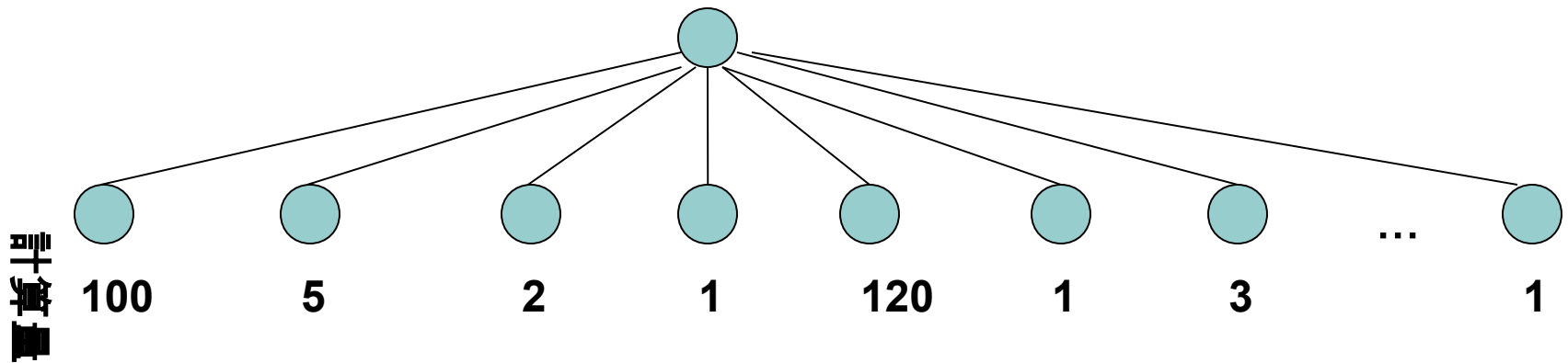
# スレッド並列とクラスタ並列

---

- スレッド（密結合）並列
  - メモリを共有するプロセッサ間の協調（一筐体内）  
（おおむね「CPU内の、コア間の並列」と思えばよい）
  - スレッド使用可能
  - 規模に制限あり。CPU8(?)台程度まで
  - コンピュータチェスで古くから採用されている
- クラスタ（疎結合）並列
  - メモリを共有せず、ネットワークでつながったプロセッサ間の協調
  - 規模はほぼ制限なし
  - スレッド使用不可。MPI等使用 → 難易度高
  - 将棋はつい最近までほとんど実績なし、チェスでも数えるほど



# 並列化の困難さ



重いノードを担当するCPUは延々と計算  
軽いノードを担当するCPUはすぐ終わり、待ちが生じる

( 並列効果が出ない )

単純に仕事を割り振ったのではダメ

→ 様々な工夫が必要

「台数増やせば強くなる」わけではない

# ボンクラーズ/Puella αの実装

---

- ハード環境
- ソフト環境
- ボンクラーズ/Puella αの新技術

# 使用ハードウェア

---

## 第1回電王戦

富士通ブレードサーバ  
(BX400)  
6コア×2ソケット×6ブ  
レード



# 使用ハードウェア

---

**12年選手権**

**普通のPC 4台 &  
100Mb ルータ  
計 ~40万円**



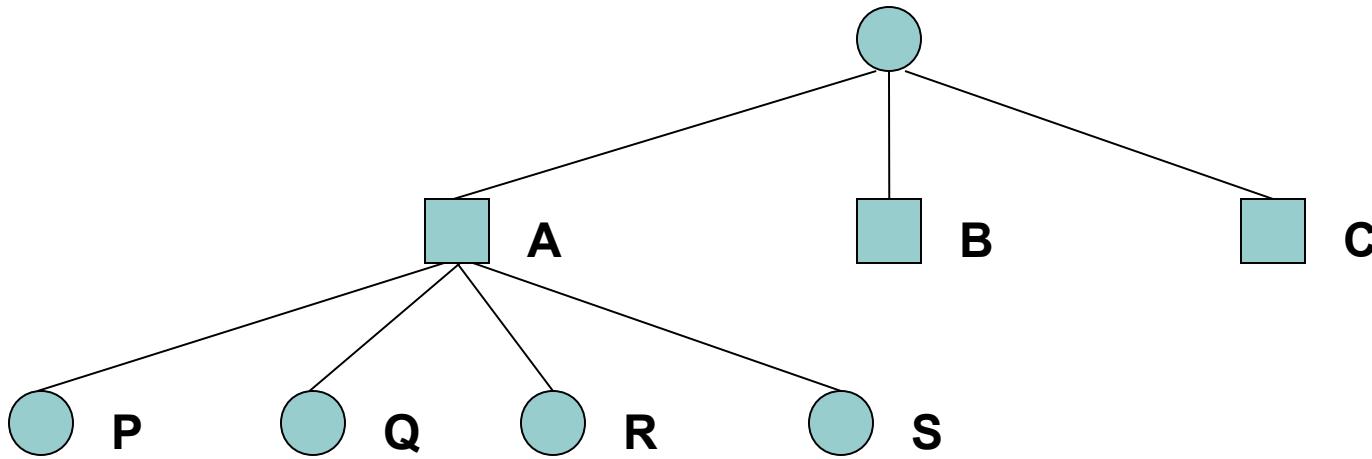
## ソフト環境

---

- **Ubuntu Linux**
- **Intel cc (時々 gcc)**
- **OpenMPI**
- **gprof, gcov, vtune**
- **vi, make, printf(/gdb)**

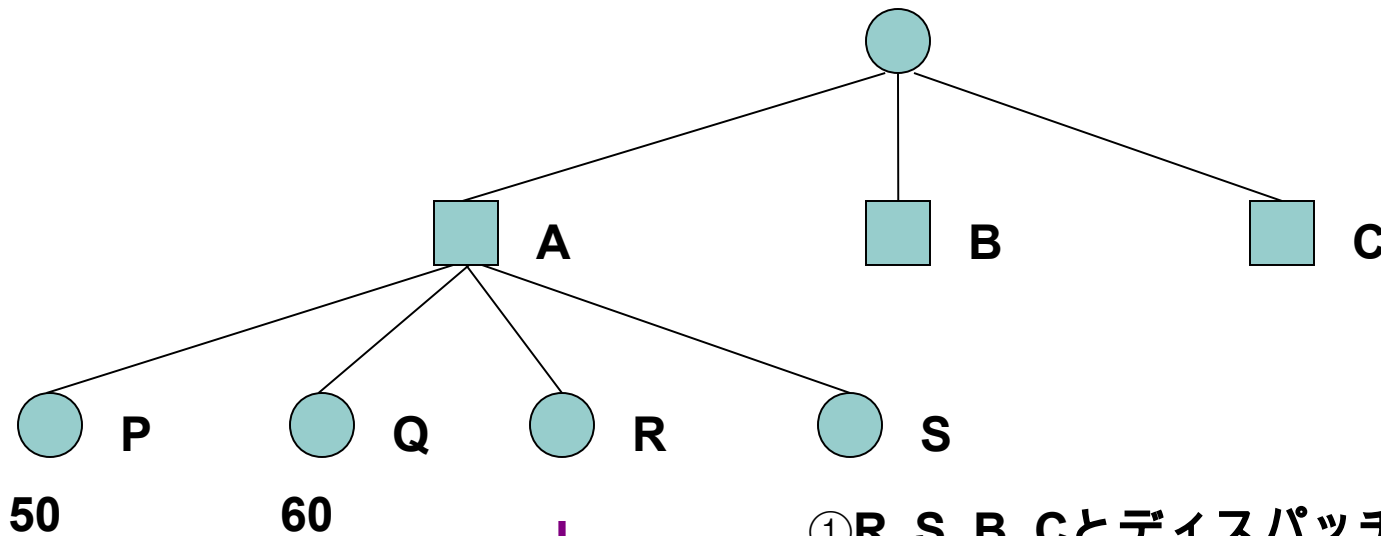
**ぜんぶ無料**

# 並列処理の基礎技術 PVSplit



- あらかじめ最善手列(PV)を予想しておく
- ①まずPを探索 (この時は並列化なし)
  - ②Pが完了後、Q,R,Sを並列探索
  - ③Aが完了 ( i.e. P-S全て完了 ) したらB,Cを開始

# ボンクラーズの技術： 投機的実行とキャンセル / リトライ



① R, S, B, Cとディスパッチする。B, Cでは、Aの現在値(=50)で探索を始める

②その後Rが値40で返ると、Aの値が40に下がるので、B,Cも値40で探索しなおす

## 参考文献

---

伊藤英紀, コンピュータ将棋におけるクラスタ並列探索. pp.117-123, 電子情報通信学会誌 2013年2月号



# 電王戦後の反響

---

- 機械が人間を超えるの？
- 人間は機械に支配されるの？

人工知能の最先端の現状は、今どうなっているのか？

## 他のゲーム

---

### 【完全情報ゲーム】

- ・ チェス、オセロ、チェッカー：既にコンピュータが強い
- ・ 囲碁：2012年3月、Zenが武宮九段に四子で勝利。人間を抜くまであと5～10年？

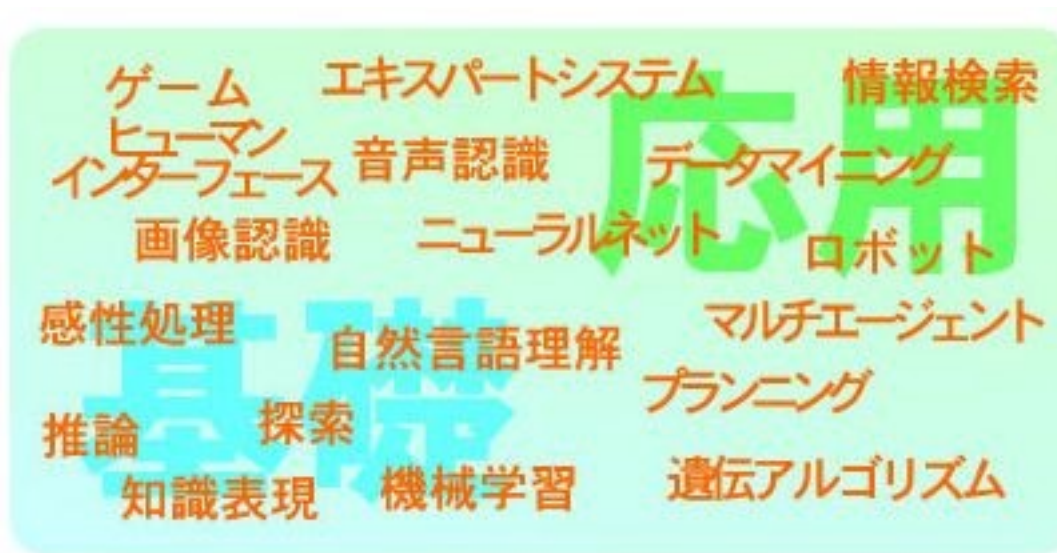
### 【不完全情報ゲーム】

- ・ バックギャモン：既にコンピュータが強い
- ・ 麻雀：コンピュータと人間が戦う文化がない？

おそらく、ゲームは囲碁が最後  
今後は実用が求められる

# 「人工知能」の扱う領域

---



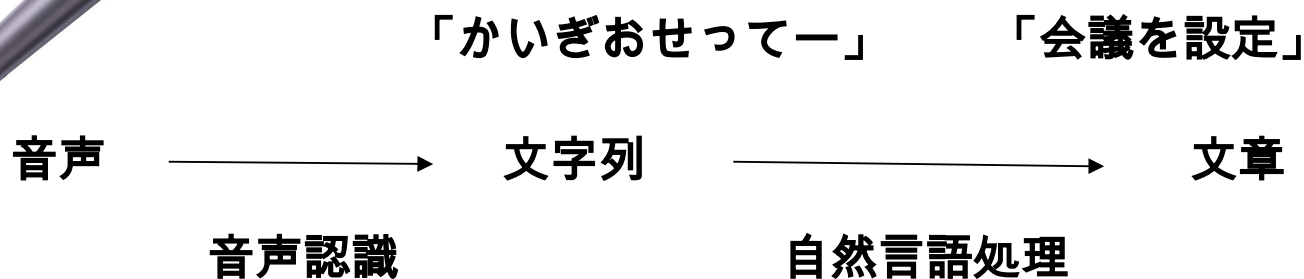
# 「知能」の要素

入力		出力		
視覚	画像認識	状況理解	(表情・身振り?)	計画・行動
聴覚	音声認識	自然言語処理	発話	音声合成
触覚・体性感覚	(各種センサー)		運動	

# 人工知能の現状 ( 1 ) Watson

IBMが開発。アメリカのクイズ番組「ジヨパ  
デイ！」で人間のトッププレイヤーに勝つ

\* 音声認識はなし。入力は文字情報



## 人工知能の現状 ( 2 ) Google Car

---

**2004年からのDARPAの無人運転車の研究をGoogleが引き継いだ**

**\* 日本では2012年CEATECで日産が無人運転デモ  
(スーパー駐車場を想定)**

# 人工知能の現状 ( 3 ) Siri



音声で話すと適切な処理をして  
くれる ( こともある )

\* 日本では「しゃべってコン  
シエル」あり(NTT)

# 人工知能の現状 ( 3 ) Siri ( 続き )



特定のパターンの文に反応する  
だけ。内容を理解しているわけ  
ではない



# 人工知能の現状 ( 4 ) ルンバ

---

部屋の状況、障害物を ( ある程度 ) 認識



経路を自分で考える

## 人工知能の現状（ 5 ） 機械翻訳

---

- 多くの場合かなりまともな訳を出すようにはなってきた
- ときどきひどい間違いをする（ e.g. “president”を「ブッシュ」）
- そのまま実務には使えない。下訳に使うケースはある
- 文を「理解」はしていない（統計的機械翻訳）

# 人工知能の現状 ( 6 ) 音声合成

---

ボークロイド ( 初音ミク等 )

- \* 音域・速度は自由
- \* やや不自然さあり

CeVio/MMDAgent

- \* 感情をある程度出せる
- \* 発話はプログラムされたもののみ

※発話に関する研究でめぼしい物はほぼ皆無

## 人工知能の現状（ 7 ） 運動制御

---

特定動作の繰り返しはかなりできる

- \* 産業用ロボット
- \* 自転車（村田製作所 ムラタセイサクくん）
- \* ポットからお茶を注ぐ（ホンダ アシモ）
- \* ジャグリング - チューリヒ工科大学の例

想定外の状況への対応はまだまだ

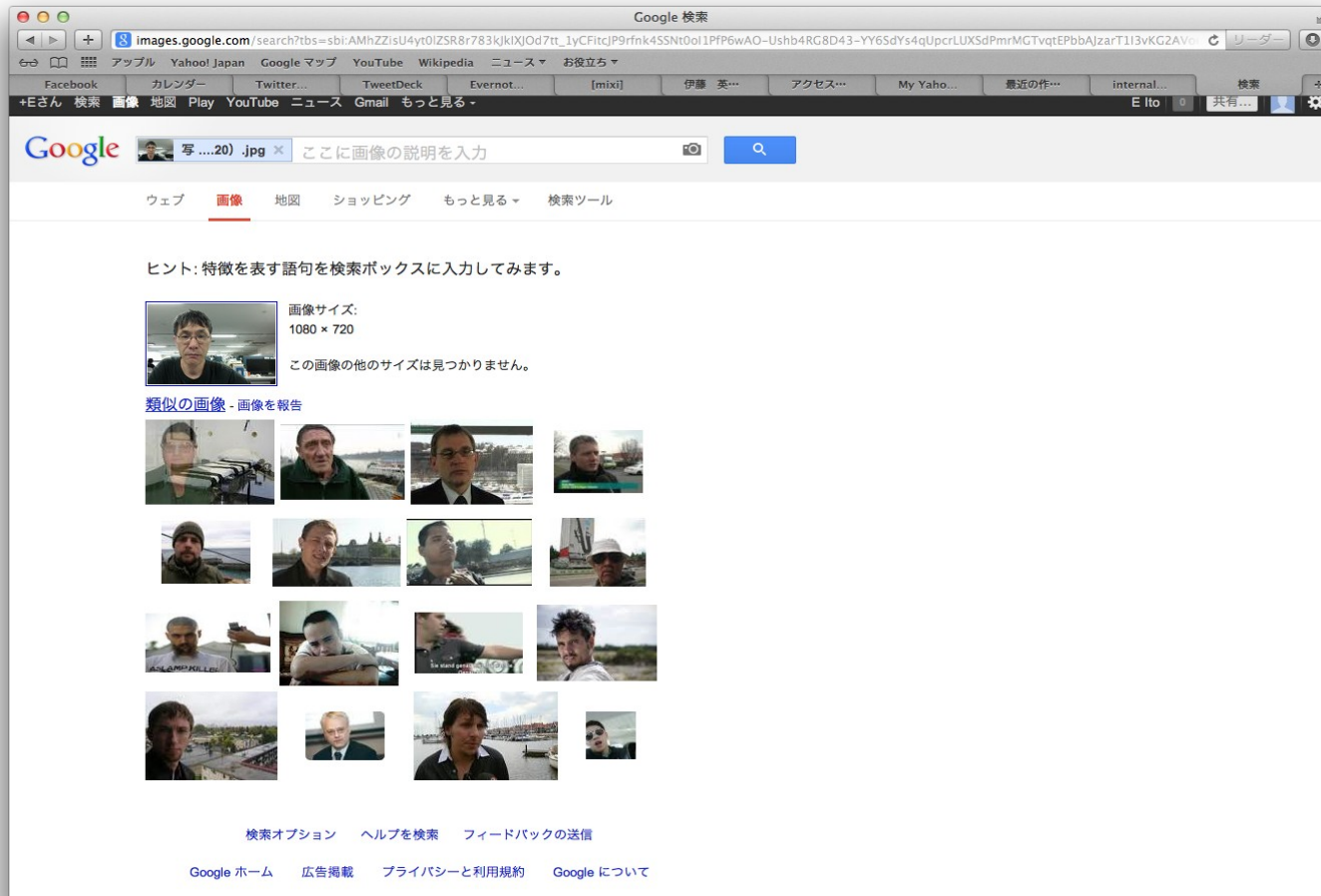
# 人工知能の現状 ( 8 ) 画像認識

---

**Google 画像検索**

**\* 画像で検索できる？試してみた**

# 人工知能の現状 ( 8 ) 画像検索結果



## 人工知能の現状 ( 8 ) 画像認識(続き)

---

Google 画像検索では顔認識はあまり動かない模様

セキュリティ用に導入されている事例はある

正面・無表情・明暗一定・隠れてない(髪やサングラス)等が条件

## 人工知能の現状 ( 9 ) その他の話題

---

### ローブナー賞

- \* チャットしている相手がコンピュータか人間かを当てる ( チューリングテスト )。人間の審判がソフトと人間とそれぞれ話す。

### 東大ロボプロジェクト

- \* 国立情報学研究所が2011年開始。  
5年後にセンター試験、10年後に東大入試合格を目指す



# 「知能」の要素

	入力		出力	
視覚	画像認識 Google画像検索		状況理解 ルンバ 、Google Car	( 表情・身振り? ) 計画・行動 ルンバ、 Google Car
聴覚	音声認識 Siri	自然言語処理 Watson、 機械翻訳	発話 ...	音声合成 CeVio
触覚・体性感覚	( 各種センサー )		運動 Quodrocopter	

## まとめ

---

- ・ 将棋はほぼ人間を超えた
- ・ ゲームはあと囲碁くらい。それももうすぐ
- ・ それでも、コンピュータは人間の知能にはまだまだ及ばない...

**多くのブレークスルーが必要。それだけ  
チャレンジしがいもある**

**ご静聴ありがとうございました**

---