方策勾配法による局面評価関数とシミュレーション方策の学習

五十嵐治一（芝浦工業大学）
森岡祐一
山本一将（株式会社コスモ・ウェブ）

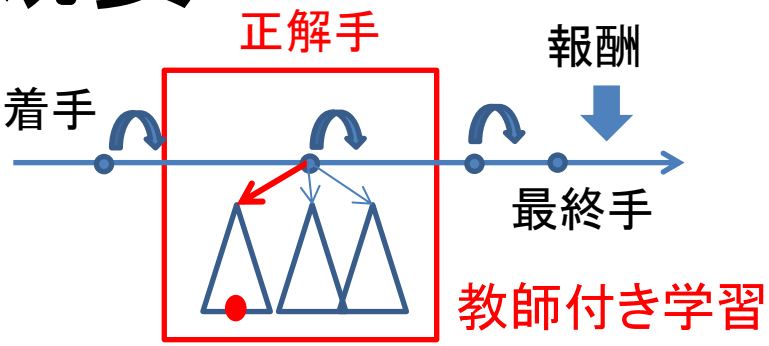
発表の流れ

0. 本報告の概要
1. 背景と目的
2. 方策勾配法
3. 指し手評価の期待値と着手決定方策
4. 探索と学習則①：再帰
5. 探索と学習則②：シミュレーション
6. 教師付き学習への適用

本報告の概要

強化学習

- TD(λ)法 or TDLeaf(λ)法
- 方策勾配法



本研究では

Boltzmann分布による確率的な着手決定方策 → T→0 通常の Min-max探索

目的関数を「指し手評価の期待値」とし，探索（読み）により求める

探索
- ① 再帰：着手決定方策 + leaf局面評価
- ② シミュレーション：シミュレーション方策 + leaf局面評価

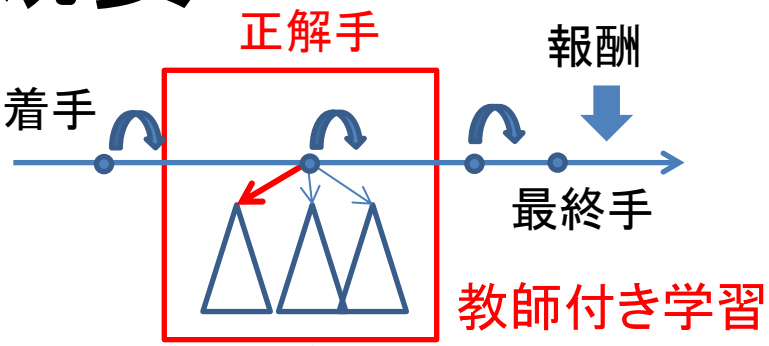
2種類のパラメータを学習

教師付き学習にも適用可

本報告の概要

強化学習

- TD(λ)法 or TDLeaf(λ)法
- 方策勾配法



① 再帰についてはGPW2012で発表済.
② シミュレーションによる探索の場合と教師付き学習が本報告の対象範囲

探索
- ① 再帰：　着手決定方策　+ leaf局面評価
- ② シミュレーション：シミュレーション方策 + leaf局面評価

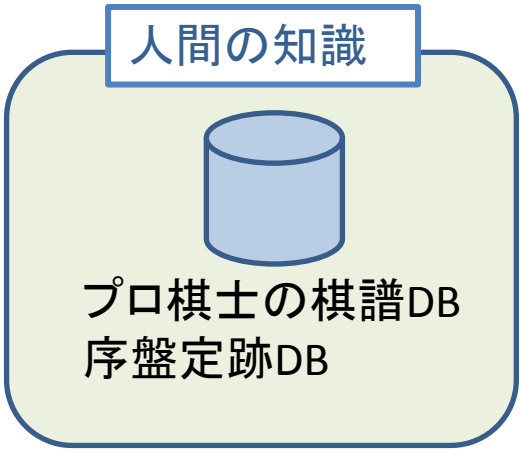
教師付き学習にも適用可

2種類のパラメータを学習

1. 背景と目的

- コンピュータ将棋が強くなった要因
  - ハードウエアの進歩・・・CPUの高速化，マルチコア
  - 並列化アルゴリズムの考案，導入
  - 機械学習による評価関数の構築

Bonanzaメソッドに代表される教師付き学習



第2回電王戦 （コンピュータ将棋Top5 vs プロ棋士5名)
コンピュータ将棋側の３勝１敗１分け

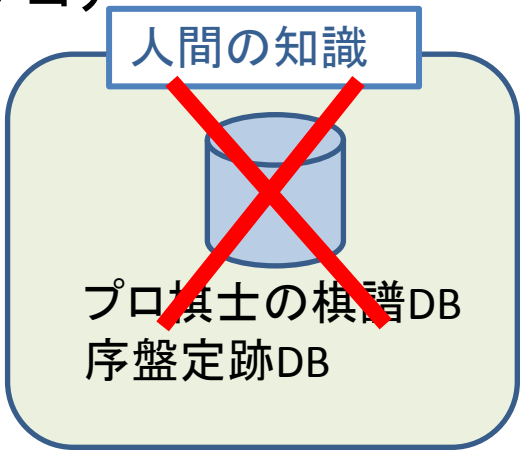
コンピュータ vs 人間 → コンピュータが人間に勝った

と言えるだろうか？

1. 背景と目的

- コンピュータ将棋が強くなった要因
  - ハードウエアの進歩・・・CPUの高速化，マルチコア
  - 並列化アルゴリズムの考案，導入
  - 機械学習による評価関数の構築

利用しない



Bonanzaメソッドに代表される教師付き学習

しかし，

将棋のルールを教え，対局させるだけでコンピュータはプロ棋士に勝てるようになるだろうか？

そこで，

強化学習の適用

コンピュータ自らが試行錯誤し，将棋を学ぶことは可能か？　新戦法，新定跡が生まれる？

2. 方策勾配法

[Williams92, 石原&五十嵐04]

目的関数： $E(a_t, s_t; \omega)$ [離散時刻t，状態s，行動a]

Boltzmann分布による確率的方策：“Boltzmann選択”

$$\pi(a_t | s_t; \omega) = \exp\left(E(a_t, s_t; \omega)/T\right)/Z$$

エピソードあたりの報酬期待値E[r]を極大化する学習則：

$$\Delta\omega = \varepsilon \cdot r \sum_{t=1}^{L} e_\omega(t) \quad \Leftarrow \quad \partial E[r]/\partial\omega = E\left[r \sum_{t=1}^{L} e_\omega(t)\right]$$

確率的勾配法

[L：エピソード長，r：報酬，ε：学習係数]

特徴的適正度： $e_\omega(t) \equiv \partial \ln \pi(a_t | s_t; \omega)/\partial\omega$

$$= (1/T)\left[\partial E(a_t, s_t; \omega)/\partial\omega - \sum_{x \in A(s_t)} \pi(x | s_t; \omega)\partial E(x, s_t; \omega)/\partial\omega\right]$$

3. 指し手評価の期待値と着手方策

- 指し手の評価は，読み（探索木の展開）を伴う方がより正確

- 「指し手評価の期待値」を目的関数とする

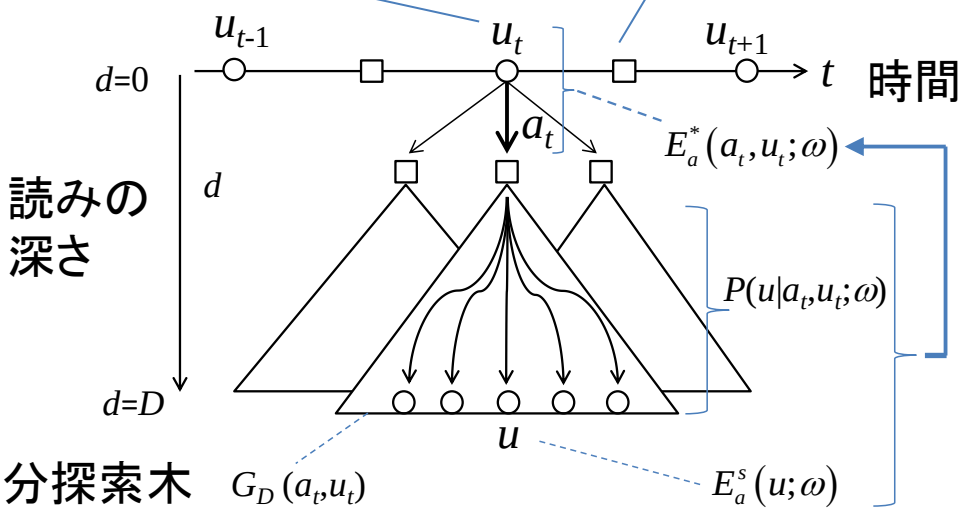
$$E_a^*(a_t,u_t;\omega) \equiv \sum_{u \in U_D(a_t,u_t)} P(u|a_t,u_t;\omega) E_a^s(u;\omega)$$

leafの局面評価関数

leafへの遷移確率

学習エージェント
Aの手番局面

相手の手番



Boltzmann選択による探索へ

通常は

- min-max探索
- 選択的探索
- モンテカルロ木探索

3. 指し手評価の期待値と着手方策

- 指し手の評価は，読み（探索木の展開）を伴う方がより正確

- 「指し手評価の期待値」を目的関数とする

$$E_a^*(a_t, u_t; \omega) \equiv \sum_{u \in U_D(a_t, u_t)} P(u | a_t, u_t; \omega) E_a^s(u; \omega)$$

leafの局面評価関数

学習エージェント
Aの手番局面

leafへの遷移確率

最善応手手順以外の有力な変化手順も考慮して指し手を評価した方が，評価値誤差に起因する探索揺らぎに対して頑健ではないか？

4. 探索と学習則①：再帰

着手決定方策：

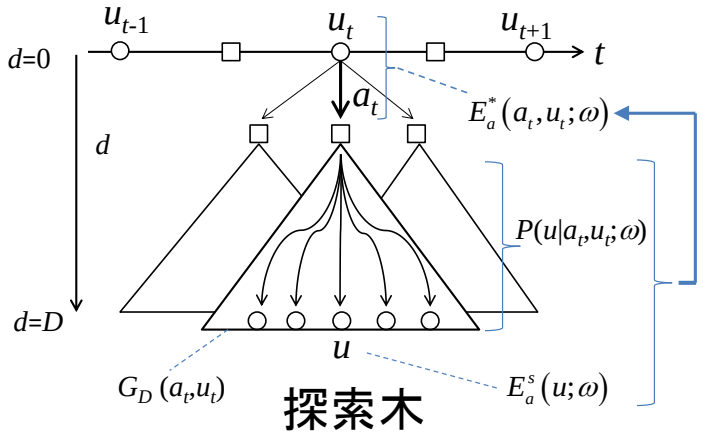
$$\pi_a\left(a_t \middle| u_t;\omega\right)=\exp\left(E_a\left(a_t,u_t;\omega\right)/T_a\right)/Z_a$$

「指し手評価の期待値」・・・①再帰，②シミュレーション

$$\pi_a\left(a_t \middle| u_t;\omega\right)=\exp\left(E_a^*\left(a_t,u_t;\omega\right)/T_a\right)/Z_a$$

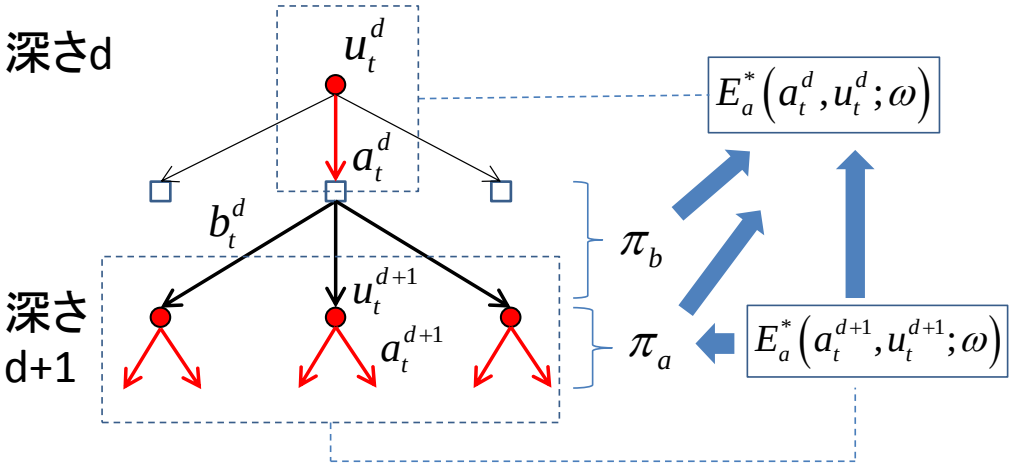
PG行動期待値法

再帰により計算可能



学習則：

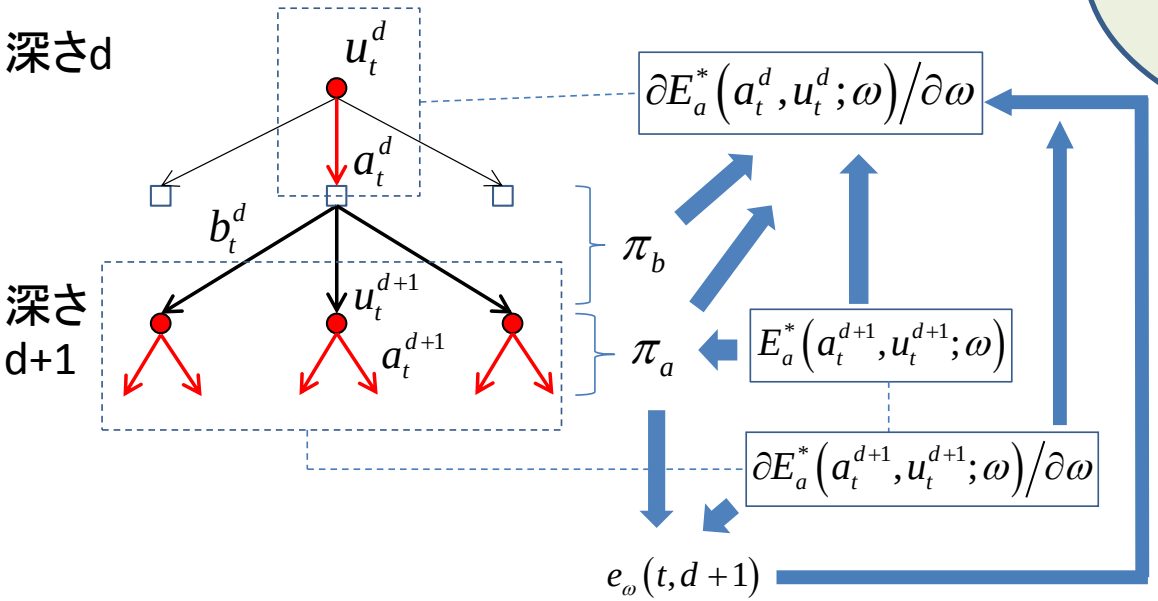
$$\begin{cases}\Delta\omega=\varepsilon\cdot r\sum_{t=1}^{L_a}e_\omega(t)\\ e_\omega(t)=(1/T_a)\left[\partial E_a^*\left(a_t,u_t;\omega\right)/\partial\omega-\sum_{x\in A(u_t)}\pi_a\left(x|u_t;\omega\right)\partial E_a^*\left(x,u_t;\omega\right)/\partial\omega\right]\end{cases}$$



$E_a^*(a_t^d, u_t^d; \omega)$

(a) 指し手評価の期待値

しかし，全leaf局面での計算が必要！



$\partial E_a^*(a_t^d, u_t^d; \omega)/\partial\omega$

(b) 1階微係数

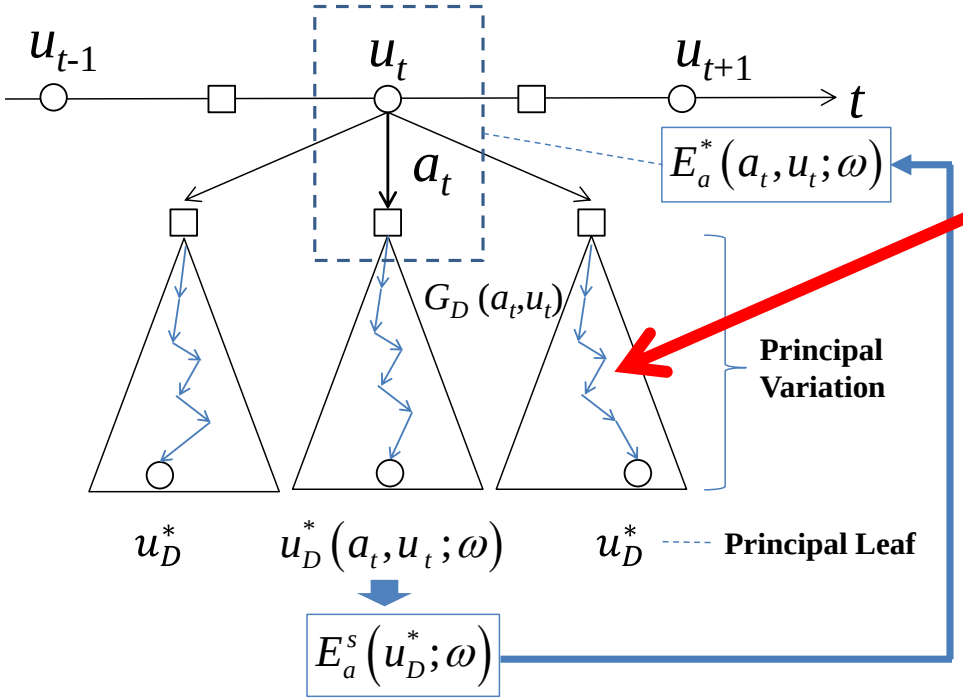
計算量削減のための近似手法のアイデア

(1) min-max探索または$\alpha\beta$ 探索の適用 : PGLeaf法

[森岡et al. GPW2012]

(2) 反復深化法の適用

(3) 異なる評価関数のprincipal leafによる期待値の計算法



$$P(u|a_t,u_t;\omega)=\begin{cases}1 & \text{if} \quad u=u_D^*(a_t,u_t;\omega)\\ 0 & \text{otherwise}\end{cases}$$

$$E_a^*(a_t,u_t;\omega)\equiv\sum_{u\in U_D(a_t,u_t)}P(u|a_t,u_t;\omega)E_a^s(u;\omega)$$

leaf局面による平均操作を
PV leafの評価値で代用

＊(2),(3)の説明は割愛

5. 探索と学習則②:シミュレーション

- 探索に再帰を用いると全leaf局面の情報が必要
- シミュレーション(またはサンプリング)による指し手評価を行う
  - 「激指」の“実現確率”
  - 専門知識による選択的探索
  - モンテカルロ木探索のplayout

前向き枝刈

期待値計算

- シミュレーション方策をどう設定するか？
  - 従来は，人間が知識を与える or 棋譜DBを用いた教師付き学習
- シミュレーション方策も方策勾配法で学習できないか？

$$E_a^*(a_t,u_t;\omega,\theta) \equiv \sum_{u\in U_D(a_t,u_t)} P(u|a_t,u_t;\theta)E_a^s(u;\omega)$$

指し手評価の期待値

シミュレーション方策

相手の方策（既知と仮定）

$$P(u|a,u_t;\theta) = \prod_{d=0}^{D-1} \pi_a'(a^d|u_t^d;\theta)\pi_b(b_t^d|v_t^d)$$

手番局面u_tから指し手a_tを経由したleaf局面uへの遷移確率

学習則

局面評価関数 $E_a^s(u;\omega)$

- 局面の形勢判断を行う

$$\Delta\omega = \varepsilon \cdot r \sum_{t=1}^{L_a} e_\omega(t)$$

$$\Delta\theta = \varepsilon \cdot r \sum_{t=1}^{L_a} e_\theta(t)$$

方策勾配法

特徴的適正度

$$e_\omega(t) \equiv \partial \ln \pi_a\left(a_t \middle| u_t;\omega,\theta\right)/\partial\omega$$

$$e_\theta(t) \equiv \partial \ln \pi_a\left(a_t \middle| u_t;\omega,\theta\right)/\partial\theta$$

シミュレーション方策 $\pi_a'\left(a^d \middle| u_t^d;\theta\right)$

- 深い読みを伴わない経験的で断片的なミニ知識による指し手選択

「手筋」「型」「直観」「第一感」

例，「王手かどうか」「ひもをつける手」（激指）

指し手の特徴量

- Boltzmann選択であれば，兄弟手の良し悪しにより探索深さを制御できる

例，他に有力な手がなければ端歩を突く手を深く読む （ツツカナ）

着手決定方策：局面評価関数とシミュレーション方策を含む

学習則

局面評価関数 $E_a^s(u;\omega)$

$$\Delta\omega = \varepsilon \cdot r \sum_{t=1}^{L_a} e_\omega(t)$$

シミュレーション方策 $\pi'_a(a^d|u_t^d;\theta)$

$$\Delta\theta = \varepsilon \cdot r \sum_{t=1}^{L_a} e_\theta(t)$$

$$e_\omega(t) \equiv \partial \ln \pi_a(a_t|u_t;\omega,\theta)/\partial\omega$$

$$= (1/T_a)\left\{E_{\pi'_a}\left[\partial E_a^s(u;\omega)/\partial\omega \middle| a_t, u_t\right] - E_{\pi_a}\cdot E_{\pi'_a}\left[\partial E_a^s(u;\omega)/\partial\omega \middle| x, u_t\right]\right\}$$

実現手

実現手＋兄弟手

$$e_\theta(t) \equiv \partial \ln \pi_a(a_t|u_t;\omega,\theta)/\partial\theta$$

$$= (1/T_a)\left\{E_{\pi'_a}\left[E_a^s(u;\omega)\sum_{d=0}^{D-1} e'_\theta(d) \middle| a_t, u_t\right] - E_{\pi_a}\cdot E_{\pi'_a}\left[E_a^s(u;\omega)\sum_{d=0}^{D-1} e'_\theta(d) \middle| x, u_t\right]\right\}$$

$$E_{\pi_a}\cdot E_{\pi'_a}\left[z|x,u_t\right] \equiv E_{\pi_a}\left[E_{\pi'_a}\left[z|x,u_t\right]|u_t\right] = \sum_{x\in A(u_t)} \pi_a(x|u_t;\omega,\theta)\left[\sum_{u\in U_D(x,u_t)} z\cdot P(u|x,u_t;\theta)\right]$$

局面

指し手　局面

$$e'_\theta(d) \equiv \partial \ln \pi'_a(a^d|u_t^d;\theta)/\partial\theta$$

シミュレーション方策の特徴的適正度

着手決定方策による期待値操作

シミュレーション方策による期待値操作

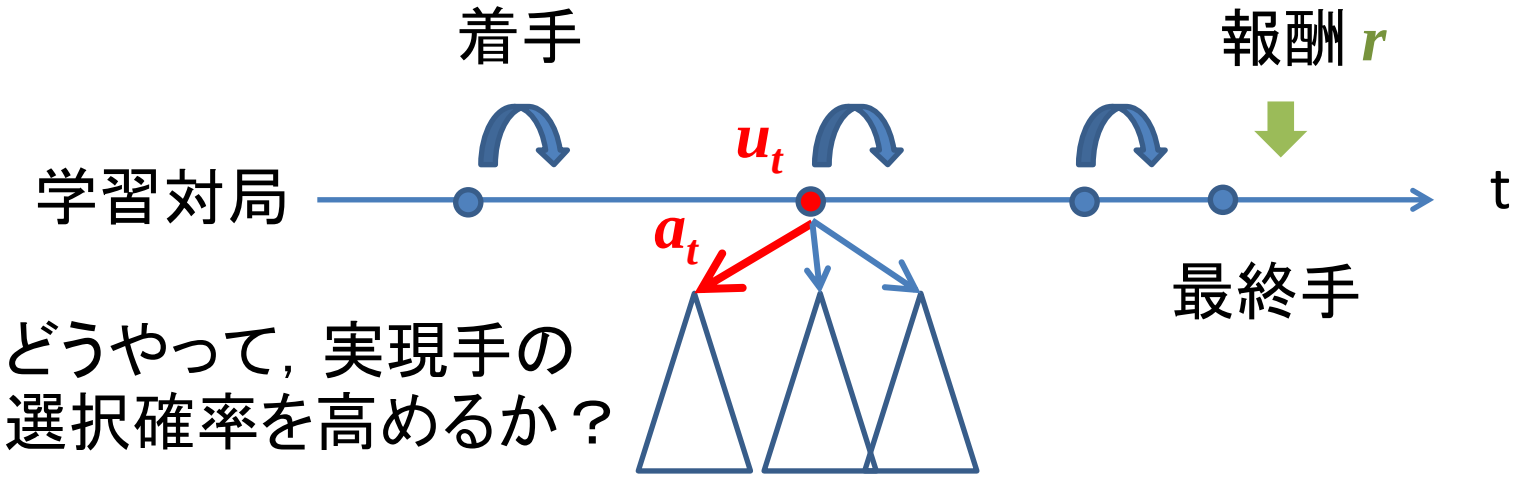
シミュレーション方策による期待値を，実現手とその兄弟手について計算し，偏差を計算

学習則

局面評価関数 $E_a^s(u;\omega)$	シミュレーション方策 $\pi'_a(a^d \mid u_t^d;\theta)$
$\Delta\omega = \varepsilon \cdot r \sum_{t=1}^{L_a} e_\omega(t) \Leftarrow \frac{\partial E[r]}{\partial \omega}$	$\Delta\theta = \varepsilon \cdot r \sum_{t=1}^{L_a} e_\theta(t) \Leftarrow \frac{\partial E[r]}{\partial \theta}$

$$e_\omega(t) \equiv \partial \ln \pi_a(a_t \mid u_t;\omega,\theta)/\partial\omega \quad \Rightarrow \quad e_\theta(t) \equiv \partial \ln \pi_a(a_t \mid u_t;\omega,\theta)/\partial\theta$$

高報酬を得た対局から，各出現局面u_tにおける実現手a_tの選択確率を高めるようにパラメータを強化



どうやって，実現手の選択確率を高めるか？

着手決定方策 $\pi_a(a_t|u_t)$

u_{t-1} 出現局面 u_t u_{t+1} 学習対局

t

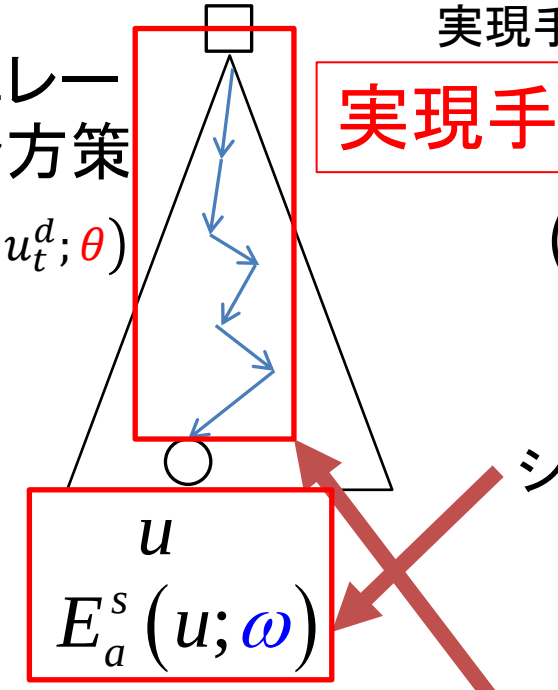
実現手 a_t

$$E_a^*(a_t,u_t;\omega,\theta) \equiv \sum_{u\in U_D(a_t,u_t)} P(u|a_t,u_t;\theta)E_a^s(u;\omega)$$

実現手の評価値

シミュレーション方策

$\pi'_a(a^d|u_t^d;\theta)$



実現手a_tの選択確率を高めるには，．．．

(i) θを固定し，ωを勾配方向へ更新

$$e_\omega(t) \equiv \partial \ln \pi_a(a_t|u_t;\omega,\theta)/\partial\omega$$

シミュレーションで得られるleafの評価値を高める

(ii) ωを固定し，θを勾配方向へ更新

$$e_\theta(t) \equiv \partial \ln \pi_a(a_t|u_t;\omega,\theta)/\partial\theta$$

高評価のleafへの遷移確率を高める

学習のアルゴリズム（全体の流れ）

Step1:実現局面の各合法手について評価値計算のシミュレーションを行う

Step2: シミュレーション内で得られたleaf局面の勾配を観測して期待値を計算する: $E_{\pi'_a}\left[\partial E_a^s(u;\omega)/\partial\omega \middle| a_t, u_t\right]$

Step3: シミュレーション内での実現手の特徴的適正度e'_θを計算し，次の期待値を求める: $E_{\pi'_a}\left[E_a^s(u;\omega)\sum_{d=0}^{D-1} e'_\theta(d) \middle| a_t, u_t\right]$

Step4: 実現手と兄弟手の計算結果から$e_\omega(t)$, $e_\theta(t)$を計算する

Step5:「指し手評価の期待値」を計算し，着手決定方策により着手を選択する

Step6: Step1～5を対局終了まで繰り返す

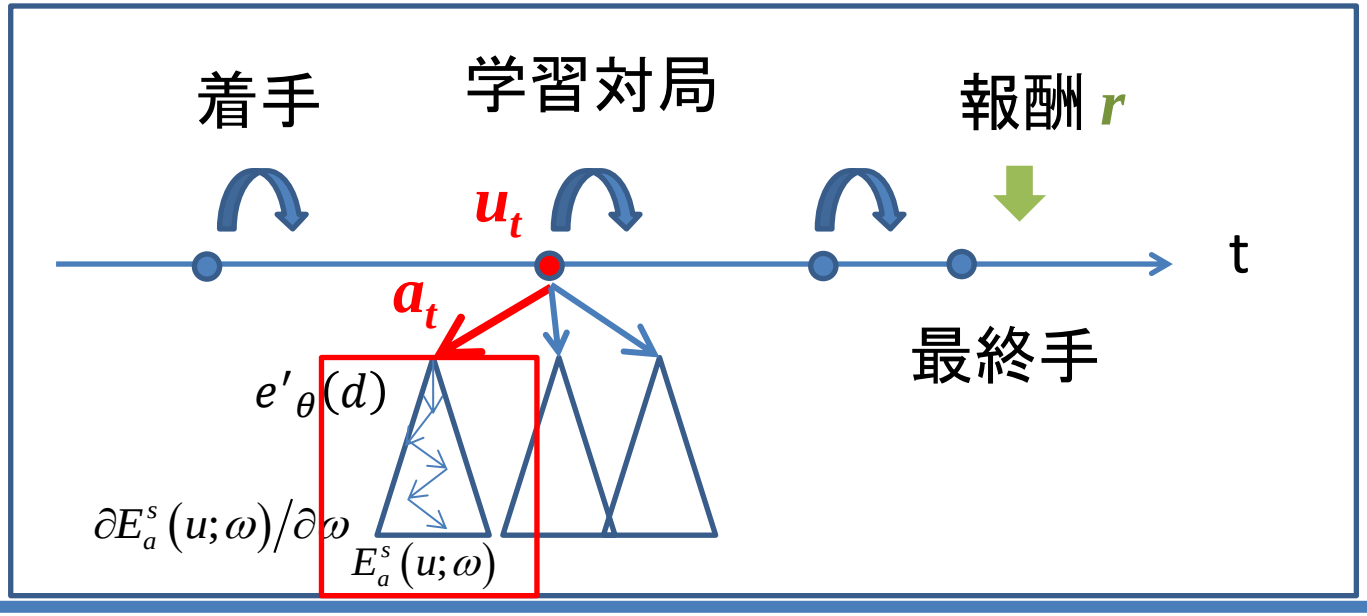
Step7: 対局結果に報酬rを与えて$\Delta\omega$, $\Delta\theta$を計算し，局面評価関数とシミュレーション方策（の目的関数）を修正する

学習のアルゴリズム（全体の流れ）

Step1:実現局面の各合法手について評価値計算のシミュレーションを行う

Step2: シミュレーション内で得られたleaf局面の勾配を観測して期待値を計算する: $E_{\pi'_a}\left[\partial E_a^s(u;\omega)/\partial\omega \middle| a_t, u_t\right]$

Step3: シミュレーション内での実現手の特徴的適正度e'_θを計算し，次の期待値を求める: $E_{\pi'_a}\left[E_a^s(u;\omega)\sum_{d=0}^{D-1} e'_\theta(d) \middle| a_t, u_t\right]$



学習のアルゴリズム（全体の流れ）

Step1: 実現局面の各合法手について評価値計算のシミュレーションを行う

Step2: シミュレーション内で得られたleaf局面の勾配を観測して期待値を計算する: $E_{\pi'_a}\left[\partial E_a^s(u;\omega)/\partial\omega \middle| a_t, u_t\right]$

Step3: シミュレーション内での実現手の特徴的適正度e'_θを計算し，次の期待値を求める: $E_{\pi'_a}\left[E_a^s(u;\omega)\sum_{d=0}^{D-1} e'_\theta(d) \middle| a_t, u_t\right]$

Step4: 実現手と兄弟手の計算結果から$e_\omega(t)$, $e_\theta(t)$を計算する

$$e_\omega(t) \equiv (1/T_a)\left\{E_{\pi'_a}\left[\partial E_a^s(u;\omega)/\partial\omega \middle| a_t, u_t\right] - E_{\pi_a}\cdot E_{\pi'_a}\left[\partial E_a^s(u;\omega)/\partial\omega \middle| x, u_t\right]\right\}$$

$$e_\theta(t) = (1/T_a)\left\{E_{\pi'_a}\left[E_a^s(u;\omega)\sum_{d=0}^{D-1} e'_\theta(d) \middle| a_t, u_t\right] - E_{\pi_a}\cdot E_{\pi'_a}\left[E_a^s(u;\omega)\sum_{d=0}^{D-1} e'_\theta(d) \middle| x, u_t\right]\right\}$$

実現手

実現局面

実現手＋兄弟手

学習のアルゴリズム（全体の流れ）

Step1:実現局面の各合法手について評価値計算のシミュレーションを行う

Step2: シミュレーション内で得られたleaf局面の勾配を観測して期待値を計算する: $E_{\pi'_a}\left[\partial E_a^s(u;\omega)/\partial\omega \middle| a_t, u_t\right]$

Step3: シミュレーション内での実現手の特徴的適正度e'_θを計算し，次の期待値を求める: $E_{\pi'_a}\left[E_a^s(u;\omega)\sum_{d=0}^{D-1} e'_\theta(d) \middle| a_t, u_t\right]$

Step4: 実現手と兄弟手の計算結果から$e_\omega(t)$, $e_\theta(t)$を計算する

Step5:「指し手評価の期待値」を計算し，着手決定方策により着手を選択する

⬇ シミュレーション（サンプリング）

$$E_a^*(a_t,u_t;\omega,\theta) \equiv \sum_{u\in U_D(a_t,u_t)} P(u|a_t,u_t;\theta)E_a^s(u;\omega) = E_{\pi'_a}\left[E_a^s(u;\omega)\middle|a_t,u_t\right]$$

⬇

$$\pi_a(a_t|u_t;\omega,\theta) = \exp\left(E_a^*(a_t,u_t;\omega,\theta)/T_a\right)/Z_a$$

学習のアルゴリズム（全体の流れ）

学習対局中の1手ごとの処理

Step1:実現局面の各合法手について評価値計算のシミュレーションを行う

Step2: シミュレーション内で得られたleaf局面の勾配を観測して期待値を計算する: $E_{\pi'_a}\left[\partial E_a^s(u;\omega)/\partial\omega \middle| a_t, u_t\right]$

Step3: シミュレーション内での実現手の特徴的適正度e'_θを計算し，次の期待値を求める: $E_{\pi'_a}\left[E_a^s(u;\omega)\sum_{d=0}^{D-1}e'_\theta(d)\middle| a_t, u_t\right]$

Step4: 実現手と兄弟手の計算結果から$e_\omega(t)$, $e_\theta(t)$を計算する

Step5:「指し手評価の期待値」を計算し，着手決定方策により着手を選択する

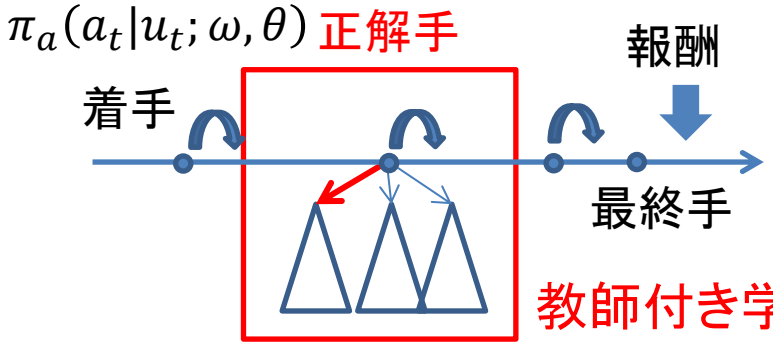
Step6: Step1～5を対局終了まで繰り返す

Step7: 対局結果に報酬rを与えて$\Delta\omega$, $\Delta\theta$を計算し，局面評価関数とシミュレーション方策（の目的関数）を修正する

$$\Delta\omega = \varepsilon\cdot r\sum_{t=1}^{L_a} e_\omega(t) \qquad \Delta\theta = \varepsilon\cdot r\sum_{t=1}^{L_a} e_\theta(t)$$

・・・一局ごとにパラメータ更新

6. 教師付き学習への適用



局面uにおける各指し手aの
選択確率が計算できるので
教師付き学習も可能

Bonanzaメソッド
よりも一般的に

正解手を1つに限定せずに，正解と思われる複数の指し手に対してそれらを選択する確率分布を学習させる

誤差関数はカルバック・ライブラー情報量

$$U_{err}(\pi^*,\pi_a) \equiv \sum_{a\in A(s)} \pi^*(a|s)\ln\left[\pi^*(a|s)/\pi_a(a|s;\omega,\theta)\right]$$

正解の着手決定方策

学習則

$$\Delta\omega = -\varepsilon\cdot\partial U_{err}/\partial\omega = +\varepsilon\sum_{a\in A(s)}\pi^*(a|s)\cdot\partial\ln\pi_a(a|s;\omega,\theta)/\partial\omega$$

$$\Delta\theta = -\varepsilon\cdot\partial U_{err}/\partial\theta = +\varepsilon\sum_{a\in A(s)}\pi^*(a|s)\cdot\partial\ln\pi_a(a|s;\omega,\theta)/\partial\theta$$

「指し手評価の期待値」の勾配
①再帰，②シミュレーション

すでに導出済

まとめ

探索（読み）を必要とするゲームにおいて，方策勾配法（強化学習）の適用方法を検討

⬇ そのために，

「指し手評価の期待値」による確率的な着手決定方策を提案

⬇ 計算手段として

①再帰，②シミュレーションの2つの場合の学習則を導出

⬇

局面評価関数，シミュレーション方策を同時に学習できる

⬇ 利用　　共通の学習目標（「報酬の最大化」）

・探索木中のノード局面への遷移確率値の計算 ➡ 前向き枝刈

・モンテカルロ木探索(playout)における期待値計算 ➡ 試行回数の削減

＊上記の勾配計算は，教師付き学習へもそのまま適用可能

・・・複数の正解手がある場合など，着手の確率分布を学習できる