修 士 論 文

証明数・反証明数を用いた
罰回避政策形成アルゴリズムの高速化に関する研究

北陸先端科学技術大学院大学
情報科学研究科情報処理学専攻

登 崇志

2005 年 3 月

修 士 論 文

証明数・反証明数を用いた
罰回避政策形成アルゴリズムの高速化に関する研究

指導教官　東条敏 教授

審査委員主査　東条敏 教授
審査委員　鳥澤健太郎 助教授
審査委員　島津 教授

北陸先端科学技術大学院大学
情報科学研究科情報処理学専攻

310083 登 崇志

提出年月: 2005 年 2 月

概 要

強化学習とは，報酬という特別の入力を手がかりに環境に適応する機械学習システムである．明示的に正解を与える教師なしに学習できるので魅力的な枠組みと言える．つまり，設計者が何をすべきか (what) をエージェントに報酬という形で指示しておけば，どのように実現するか (how to) をエージェントが学習によって自動的に獲得する枠組となっている．

ゲーム問題において，方策 (現状態に対して行動を決定する関数) を考案するには，そのゲームに対する十分な知識が必要である．しかし，強化学習を用いることにより，勝敗を報酬とすることで自動的に方策を獲得することができる．強化学習の適用例として，オセロゲームの盤を表現したニューラルネットワークを，TD(λ) で学習するもの，チェスプログラム Knight Cap の線形評価関数の各重みを，TD(λ) で調整するものなどがある．これらは最適政策すなわち，大差で勝利する方策を獲得することを目的としている．

一般的な 2 人対戦のゲーム問題は負けないこと，すなわち罰の回避が大前提とされ，必ずしも圧勝等の最適性は要求されない．このような特性を利用したゲーム問題の強化学習の適用例として，宮崎による罰回避政策形成アルゴリズムを用いたオセロゲームへの応用があげられる．罰回避政策形成アルゴリズムとは，それまでに経験した罰 (負け, 負の報酬) を得る状態と行動を記憶して，それを回避する政策を形成するアルゴリズムである．ある状態で選択可能な全ての行動が罰行動 (負ける行動) となったときに，その状態を罰状態とする．また，相手が最善の行動を行ったときに罰状態へ至ると確定した行動も罰行動として，罰行動を伝播していく．しかし，膨大な状態空間を持つオセロゲームでこれを応用する場合，各状態の罰の伝播にかかる計算コストが膨大となる．また，勝敗が確定するまで罰を得ることが無い問題では，序盤から中盤にかけて罰が伝播するためには，膨大な試行回数を要する問題点がある．そのため，これらの問題を解消するためにオセロゲームの応用では以下の改良がなされた．現在経験している状態と，現在の状態から数手先を読んで展開した状態だけから罰を伝播し，罰の伝播にかかる計算コストを軽減した．また，既存知識を与えることによりヒューリスティックな罰 (準罰) を与え，序盤と中盤において罰の伝播を可能にした．しかし，準罰は必ずしも正確な罰の規範ではない．そのため，罰の伝播を準罰に頼っている序盤から中盤において，準罰が少ないときは勝ち筋を獲得するまでに必要な罰の伝播に遅延が起こる．逆に多いときは誤った伝播が頻繁におこり，獲得すべき勝ち筋を罰とみなしてしまう問題点があった．つまり，罰回避政策形成ア

ルゴリズムによる勝ち筋の獲得には，ヒューリスティックによる罰の精度に大きく依存してしまうのである．

本稿では宮崎による罰回避政策形成アルゴリズムにおいて問題となる，ヒューリスティックの評価にへの依存と学習に必要な試行回数の削減を目的とし，罰の伝播に証明数と反証数を用いた手法を提案する．(反) 証明数とは，現在の局面を勝ち (負け) を証明するために，勝ち (負け) を証明しなければいけない現在展開されている末端の局面の数である．証明数・反証数をオセロゲームに適用すると，(反) 証明数はその盤面が勝ちであるために勝ち (負け) を示さないといけない展開される盤面の個数の最小値となる．先端の (反) 証明数の個数が最小の手は現在の状態を (負け) 勝ちと示すために，必要な状態展開数が最も少ない手となる．つまり，展開する状態が少ないため，深く探索することができ終了盤面により近い状態を先読みすることが可能となる．

提案手法においては，伝播すべき罰を最終局面とみなし，証明数と反証数を罰の伝播に用いることにより，現在の状態が罰 (必負) であるか高速に判断しする．それにより，学習速度が向上し，少ない試行回数で罰回避政策の獲得が可能となる．その結果，ヒューリスティックな評価による罰が少なために試行回数が多くかかる問題であっても，罰回避政策を求めるのに十分可能な試行回数で求めることができる．つまり，ヒューリスティックによる罰の依存度が減りことに繋がる．

本研究では，証明数と反証数を用いた罰回避政策アルゴリズムを実際にオセロゲームに実装した．その結果，提案手法が従来手法に比べ少ない試行回数で罰回避政策 (勝ち筋) を獲得することができた．従来の手法では計算コストの問題で，罰回避政策 (勝ち筋) を見つけることができなかったヒューリスティックな評価による罰を少なくした場合において，提案手法では獲得することに成功した．

本研究ではさらに，学習により獲得した政策の汎化を行った．罰回避政策形成アルゴリズムは特定の相手プレイヤーに対して勝ち筋を発見する方法である．よって，学習結果を類似盤面といえど汎化させることはできない．そこで，異なる対戦相手に対して，過去の類似盤面での学習成果を汎用するために，ニューラルネットワークの評価によって罰を判定する手法を提案する．盤面の罰判定をニューラルネットワークで表現し，これに強化学習によって得られる罰を信号として与えることによって学習を行う．それにより，類似盤面に対して汎化を行うことができる．この手法を実装することにより，異なる対戦相手に対し勝率をあげることに成功したことを実験結果により示す．

目 次

第1章　はじめに

1.1　研究の背景と目的

強化学習とは，報酬という特別の入力を手がかりに環境に適応する機械学習システムである．明示的に正解を与える教師なしに学習できるので魅力的な枠組みと言える．強化学習研究は環境同定型と経験強化型のふたつに類別できる [?]．環境同定型とは，離散マルコフ決定過程 (MDPs) で記述される環境を同定し，そこでの最適性を保証する接近である．現在，環境の状態遷移確率を明らかに同定するものとして k-確実探索法 [?]，暗に同定するものとして Q-learning[?] が知られている．

環境同定型は MDPs 環境下で最適性が保証される反面，膨大な試行錯誤を要する欠点をもつ．そこで，最適性を重視せず，経験した範囲内での合理性を保証する接近が経験強化型である．現在，経験を系列すなわちエピソードの形で記憶して学習するものとして，Profit Sharing[?] やモンテカルロ法 [?]，状態遷移の有無のみを記憶し学習するものとして，合理的政策形成アルゴリズム [?] や罰回避政策形成アルゴリズム (PARP)[?] が知られている．

ゲーム問題に対する強化学習の適用例には，バックギャモン盤を表現したニューラルネットワークを，TD(λ)[?] で学習するもの [?]，オセロゲームの盤を表現したニューラルネットワークを，TD(λ) で学習するもの [?]，チェスプログラム Knight Cap の線形評価関数の各重みを，TD(λ) で調整するもの [?] などがある．これらの手法は，適用した問題領域では一定の成果を得ているが，結果の成否は，使用したニューラルネットワークや評価関数に大きく依存する．

一般にゲーム問題では負けないこと，すなわち罰の回避が大前提とされ，必ずしも最適性は要求されない．現在このような領域に適した強化学習の手法として，先行研究である PARP がある．これは，それまでに経験した罰を得る状態と行動を記憶して，罰を回避する手法である．この手法を膨大な状態空間 (オセロ) に対応させるために先読みによる罰のスキャンや，既存知識を与えることによりより罰を緩和する方法を行っている．先読みによる罰のスキャンは，実際に行った手以外の範囲を学習することを可能にした．既存知

識による罰の緩和は，罰の無い序盤から中盤かけての既存知識による罰 (準罰) の伝播を可能とした．先ほども述べたが，ゲーム問題の強化学習において，ゲームに対する深い知識を必要としない利点がある．しかし，この既存知識に大きく依存する手法ではこの魅力が失われることになる．強化学習の魅力を保持するためにも，既存知識による罰に頼らない手法が必要となる．そこで本研究では，罰回避政策形成アルゴリズムの既存知識による罰の必要性を削減し，膨大な状態空間に応用させる手法を提案する．PARP の試行回数の削減と，既存知識による罰の必要性の削減が重要となる．

1.2 論文の構成

本論文の構成 (図??) として，第 2 章では本論文で使用する用語の定義，対象問題クラスならびに提案手法のベースとなる PARP のオセロへの適用方法，そして，従来の PARP の問題点について記述する．第 3 章では，提案手法に用いる証明数・反証数について説明し，先読みによる罰のスキャンを効率化することにより，試行回数を削減することを考える．また，この手法により伝播の効率化を示すと共に，既存知識の依存を軽減することを考える．第 4 章では従来の手法と提案手法をオセロゲームで実装し対戦させる．対戦相手には，村上 9 段により考案された開放度理論を用いた評価関数を用いる．従来の手法と試行回数を比較し削減されていることを示す．第 5 章では，ある相手から勝ち筋を獲得したとき，異なる対戦相手に適用する手法を提案する．第 6 章は結論であり，本研究の成果と今後の課題について述べる．

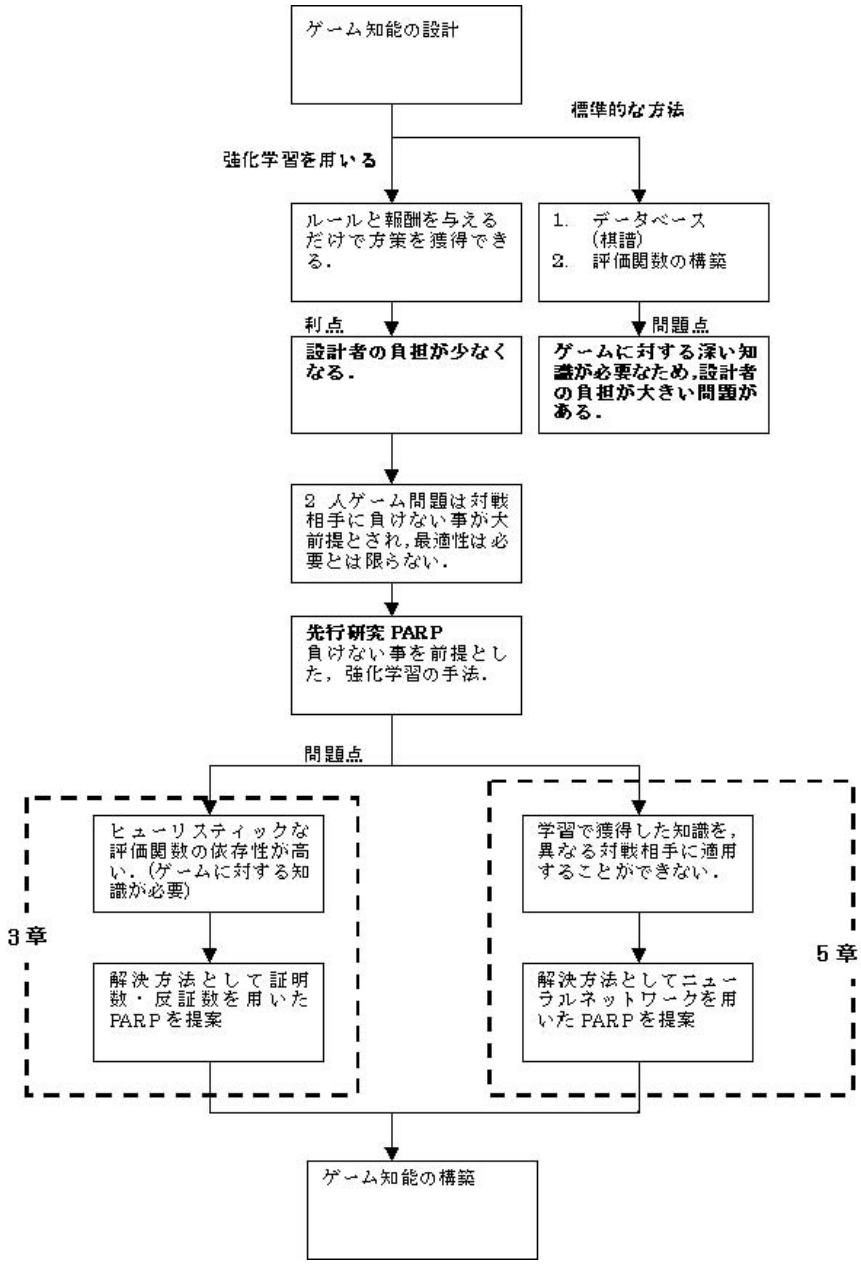
ゲーム知能の設計
標準的な方法
強化学習を用いる
ルールと報酬を与えるだけで方策を獲得できる．
1. データベース（棋譜）
2. 評価関数の構築
利点
問題点
設計者の負担が少なくなる．
ゲームに対する深い知識が必要なため，設計者の負担が大きい問題がある．
2 人ゲーム問題は対戦相手に負けない事が大前提とされ，最適性は必要とは限らない．
先行研究 PARP
負けない事を前提とした，強化学習の手法．
問題点
3 章
ヒューリスティックな評価関数の依存性が高い．（ゲームに対する知識が必要）
解決方法として証明数・反証数を用いたPARPを提案
5 章
学習で獲得した知識を，異なる対戦相手に適用することができない．
解決方法としてニューラルネットワークを用いたPARPを提案
ゲーム知能の構築

図 1.1: 論文の構成

第2章　先行研究であるPARPと問題点

2.1　対象問題クラス

本研究では，離散マルコフ決定過程 (MDPs) で記述される環境を対象とする．学習器は環境からの感覚入力に対し，行動を選択して実行に移す．学習器は行った一連の行動により環境から報酬，または罰を受け取る．時間は感覚入力と行動のサイクルを一つの単位として離散化される．感覚入力の状態は離散的な属性と値のベクトルとして与えられ，行動は離散的に分けられた種類の中から選ばれる．以下では，感覚入力の状態を単に状態と呼び，ある時点でそれまでに経験したことのある状態を既知状態と呼ぶ．

ある状態で実行可能な行動はルールとし，状態 s で行動 a を選択するというルールを sa と記述する．状態またはルールを入力として，報酬または罰を返す関数を報酬関数と呼ぶ．時刻 t で状態 s_t に存在し行動 a を選択したときに，時刻 $t+1$ で訪問する状態 s_{t+1} をルール s_{ta} による遷移先状態と呼ぶ．任意のルールによる遷移先状態とその遷移に関する報酬関数が予測可能な場合を先読みできると呼ぶ．この先読みは全てを網羅する必要は無いが，実際に遷移するはずのない状態へ遷移したり，受けとるはずの無い報酬 (罰) を受け取ることがあってはならない．

初期状態あるいは報酬 (罰) を得た直後から次の報酬 (罰) を受け取るまでのルール系列をエピソードという．さらに各状態に対し，選択すべき行動を与える関数を政策と呼び，各状態に対して行動がただ一つだけ与えられる政策を決定的政策と呼ぶ．本論文では，決定的政策のみを扱うので，以下では決定的政策を単に政策と呼ぶ．単位行動当たりの期待獲得報酬量が正である政策を合理的政策，罰を得ることのない政策を罰回避政策と呼ぶ．本論文では，罰回避政策の中に合理的政策が存在する環境を対象とする．

2.2　罰回避政策形成アルゴリズム (PARP)

ある合理的政策の構成要素となっているルールを合理的ルール，それ以外のルールを非合理的ルールと呼ぶ．直接罰を得たルールを罰ルールと呼ぶ．また，罰ルールと非合理的

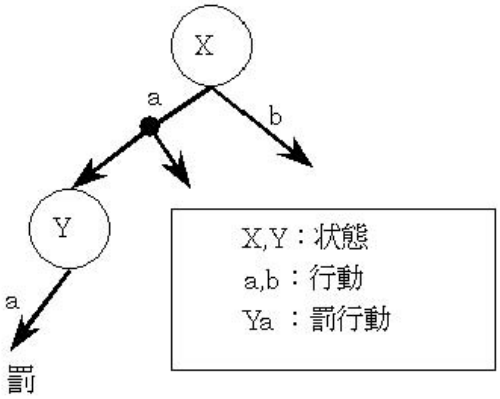

図 2.1: 罰ルール (xa,xy) および罰状態 (y) の例

ルールしか持たない状態へ遷移するルールも罰ルールとする．加えて，選択可能ルールが罰ルールと非合理的ルールで構成されている状態を罰状態と呼ぶ.

例えば, 図??ではルール Ya は罰ルールである．全てのルールが罰ルールとなったので状態 Y は罰状態である．そして，罰状態 Y へ遷移する可能性のある行動 Xa が罰ルールとなり，罰が伝播される．

罰状態へ遷移すると未知のルールがない限り，罰をうける可能性が残る．そのため，既知状態の中から罰ルールおよび罰状態を正しく判定することが必要となる．そのための手法として罰ルール判定手続きが (図??) がある．この手法により，全ての既知状態について罰ルールおよび罰状態を判定した後，罰状態をのぞいた既知状態に対して合理的政策を形成するものが罰回避政策形成アルゴリズム (PARP) である．

PARP は罰回避政策の中に合理的政策が存在する場合，それらの中のひとつを確実に獲得できる．しかし，状態数を N，行動の種類を M とした場合，領域計算量は $O(MN^2)$ となり，対象問題の状態数が増加するにつれ，計算量が膨大になってくる．実問題では膨大な状態空間をもつものが多いため，この制約を緩和する必要がでてくる．

2.3 PARP のオセロゲームへの応用

PARP を膨大な状態空間を持つ問題であるオセロゲームへ対応させるために，3 つの改良がなされている [?].

```
procedure 罰ルール判定手続き
begin
  これまでに経験したエピソードの中で,
  直接罰を得たことのあるルールにマークする
  do
    以下の条件が成立する状態にマークする.
      その状態で選択可能なルールが非合理的ルールまたは
      マークが付与されたルールのみである.
    以下の条件が成立するルールにマークする.
      そのルールで選択可能な状態の中の少なくとも
      一つがマークされている.
  while 新たに少なくともひとつの状態がマークされる.
end.
```

図 2.2: 罰ルール判定手続き

- **改良 1：既知状態全てをスキャンしていた罰の伝播をエピソードのみに限定する**
  通常の罰の伝播をエピソードのみに限定することより，領域計算量が $O(MN^2)$ から $O(MN)$ へと削減される．これは，既知状態全体をスキャンする場合に比べ，通常，罰の伝播が遅延する．罰の伝播の遅延を改善するために以下の改良 2，改良 3 を用いる．

- **改良 2：先読みによる罰の伝播**
  2.1 節で述べた先読みができる問題クラスにおいて，先読みを用いるために必要な環境を図??に示し，それを用いた罰ルール判定手続きを図??に示す．行動選択前には，現在の状態から遷移可能な状態を展開し，罰ルールを判定する．ここでいう展開とは，罰ルールの存在を停止条件にした横型探索を意味する．さらに，行動選択後，新たな罰を得た場合，再度，罰ルールを判定 (図??) する．先読みによる罰の伝播は一回の行動選択ごとに，罰ルール判定の手続きを起動しなければならず，時間計算量としては，オリジナルの罰ルール判定手続きを上回る可能性がある．しかし，オリジナルの報酬または罰を得るごとに，全ての既知状態を調べなければならないのに対し，先読みによって展開された状態に対してのみ調べればよい．しかし，先読みにかかるコストが大きい環境では注意を要する．

- **改良 3：ヒューリスティックな評価による罰の緩和**
  罰回避政策形成アルゴリズムを膨大な状態空間を有する問題に適用した場合，学習

１状態が持つフラグの種類
S_PENARTY：罰状態
SLEAF：無調査状態
S_LOOK：減状態から遷移可能である状態
S_EPISODE：実際に遷移してきた状態

１行動（ルール）が持つフラグの種類
A_PENARTY：罰ルール
A_NEWPENARTY：新しく見つかった罰ルール

アルゴリズムの動作に必要なもの
- 判定した罰ルールを記憶する領域：**長期記憶部**
- １エピソードと先読みにより展開された状態を記憶する領域：**短期記憶部**
- 短期記憶部に記憶可能な状態数：**n**

図 2.3: アルゴリズムに必要な環境

途中でメモリ不足となり十分に罰ルールが伝播されない可能性がある．この問題を改善するために，既存知識を与え，ヒューリスティックな評価により，環境が与える罰の他に制約条件を設ける．以下ではこれを準罰と呼ぶ．準罰とはつまり,「この行動を選ぶとおそらく罰を得る」「このような状態にきてしまうとほぼ罰を得てしまうだろう」というような，既存知識による緩和された罰である．

上記の改良による実際に行われる罰の伝播は次のようになる．

1. 現在の状態から行動を展開する (図??)
2. 行動が罰行動であるかを判定 (図??)
3. 罰行動以外の行動を展開し遷移可能な状態を展開 (図??)
4. 2 で遷移可能な状態のうち 1 つの行動を展開 (図??)
5. 2〜4 を規定の状態展開数に達成するまで繰り返す (図??，図??)
6. 展開された状態と行動から罰を伝播する (図??)
7. 罰状態以外の行動をとる (図??)
8. 全ての行動が罰になれば (図??)，次の試行で 1 つ上の状態で別の行動をとる (図??)

Initialize　　開始状態の状態－行動対をりストに加える
　　(S_LEAF，S_EPISODE をマーク，その他はクリア)
while(リストの状態数が規定値 n を超えない)
{
　S_LEAF がマークされたある 1 つの状態 s に対して
　　長期記憶部に罰ルール照会
　　該当の行動 a∈As に A_PENARTY をマーク
　　行動全てが罰ルールだった場合は状態 s に S_PENARTY をマーク
　　if(該当ルールなし)
　　状態 s を持つ全行動を報酬関数でチェック，
　　罰を得る行動に A_PENARTY，A_NEWPENARTY をマーク
　　状態 s の S_LEAF を外す
　　状態 s の行動のうち，A_PENARTY がマークされていない行動の s'を計算，
　　S_LEAF をマークしてリストの末尾に追加.
　　(s'がすでに存在する場合は S_LEAF は変更しない)
}
リストの状態－行動対に対して罰ルール判定.
新たに罰ルールになった行動に A_PENARTY，A_NEWPENARTY をマーク.
A_PENARUTY がマークされた行動すべてを長期記憶に登録.
A_PENARTY をクリア.

図 2.4: 罰ルール判定手続き (行動選択前)

状態 s で行動 a を選び状態 s'に遷移し，罰 r(0 or 1)を受け取ったとする.

リスト中の状態 s'に S_EPISODE をマーク.
if(行動 a は罰ルールではなかったのに罰 r=1 を受け取った)
　行動 a に A_PENARTY，A_NEWPENARTY をマーク.
　罰ルール判定．新規罰ルールに A_PENARTY，A_NEWPENARTY をマーク
　A_NEWPENARTY がマークされた罰ルールを長期記憶部に登録.
状態 s'から展開されている状態 S_LOOK をマーク.
S_LOOK，S_EPISODE どちらもマークされていない状態をリストから削除.

図 2.5: 罰ルール判定手続き (行動選択後)

図 2.6: 現状態から行動を展開

図 2.7: 行動の罰判定

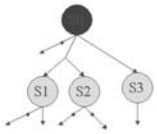

図 2.8: 罰行動以外の行動から遷移可能な状態を展開

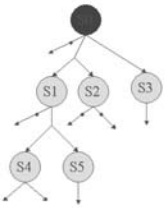

図 2.9: 遷移可能な状態のうち 1 つの行動を展開

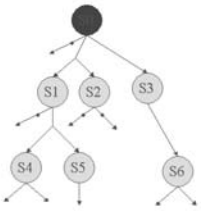

図 2.10: 状態の展開 1

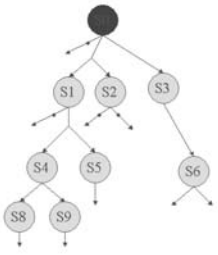

図 2.11: 状態の展開 2

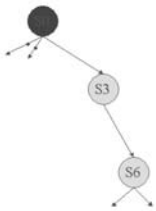

図 2.12: 罰の伝播

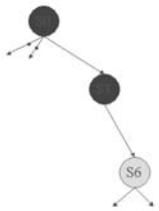

図 2.13: 行動選択

図 2.14: 行動全てが罰の状態

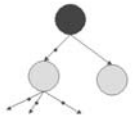

図 2.15: 罰回避行動

2.4 PARP の問題点

膨大な状態空間に対応させるために改良された PARP であるが，実際にはいくつかの問題点がある．もっとも大きな問題が中盤での罰の伝播である．オセロゲームの特性として学習器が報酬 (罰) を受け取る状態は，ゲームが終了した盤面の状態のみである．よって，中盤の状態に罰を伝播するには膨大な試行回数が必要となってくる．この膨大な試行回数を削減するために用いられる準罰であるが，準罰の設定には人間の主観による試行錯誤を要する．実際に先行研究の実験結果では，準罰の設定を少し変化させるだけで，数千の試行回数では勝ち筋である罰回避政策を習得できないことが起きている．準罰状態を少なくすると伝播すべき罰が減り，勝ち筋を得るのに時間がかかり，逆に多くすると，準罰はヒューリスティックな評価により与えられる罰であるため，不確かなことがあり開始状態までが準罰状態となることが起きている．つまり，試行回数削減のために用いた準罰の精度が，学習速度に大きく影響を及ぼす結果となっている．そこで，ヒューリスティックな評価による準罰を用いた試行回数の削減方法以外の手法が必要となってくる．

第3章　PARPの改善方法

3.1　改良の方法

オセロのゲームの特性として終了盤面になって初めて報酬 (罰) を受け取ると 2.4 節で記述した．従来手法の先読みでは幅優先探索をもちいて罰の伝播を行っている．これは，中盤の一つの状態で平均 10 通りのルールがあるオセロゲームにおいて，膨大な状態展開数がないと先読みによる罰の伝播が行われないことを意味する．この特性はチェス，囲碁，将棋等のボードゲームにおいても同様に存在する．中盤状態での罰の伝播が効率化されれば，罰回避政策を獲得するのに必要な試行回数を削することができる．また，罰の伝播の効率化と同様に準罰の伝播も効率化され，準罰の数を少なくすると罰回避政策できなかった問題において，罰回避政策を獲得できることが期待される．そこで本研究では従来の PARP では，罰の伝播における先読みに幅優先探索を用いていたのに対し，提案手法では深さ優先探索を用いる．次節では先読みの用いる深さ優先探索の手法の一つである証明数・反証数を用いた探索について述べる．

3.2　証明数・反証数

証明数・反証数の元となったのは McAllester によって提案された，共謀数 [?] という概念である．本論文では共謀数の概念は直接関係ないので，興味のある方には文献 [?] を参照してもらいたい．共謀数は minimax 木という，有名な α-β 法が探索の対象とする木に対して提案された概念で，これを AND/OR 木に適用すると，証明数・反証数の概念が出てくる．AND/OR 木とは 2 人ゲームでの探索木で，自分の手番を OR ノード，相手の手番を AND ノードとする．AND ノードは相手の行動であるため，最悪の場合 (自分が負ける，自分に不利になる行動をとられる) を想定する．つまり，1 つでも自分が負ける行動があれば負けと判断できる．OR ノードでは逆に自分が行動を選択できるため，良いと判断した行動を選択できる．つまり，1 つでも自分が勝つ行動があれば勝ちと判断できる．AND/OR 木探索における最終的な 2 つの評価値 (「勝ち」「負け」) を true と false で一般

化すると，証明数・反証数の定義は次のようになる．

定義 [証明数 (反証数)]

各ノードに対して定義される値で，そのノードの最終的な価値を true(false) にするために，true(false) となることを示さないといけない先端ノードの個数の最小値．

ノード n の証明数を $\mathrm{pn}(n)$，反証数を $\mathrm{dn}(n)$ で表すと，具体的な証明数・反証数の計算法は図?? のように再帰的に行われる．この計算法を用いた AND/OR 木における証明数と反証数の値は，図??のようになる．AND ノードのとき証明数が子ノードの証明数の和となるのは，相手に行動選択権があるので，相手が取れる行動全てに対して勝ちを示さなければ必勝とならないためである．また，OR ノードのとき証明数が子ノードの証明数の最小値となるのは，自分に行動選択権があるので，現在とれる行動の中でただ 1 つ勝ちを証明すれば必勝となるためである．反証数は，自分の負けを証明することは相手の勝ちを証明することと等しいため，証明数とは AND ノードと OR ノードの計算法が逆となる．この証明数と反証数は，そのノードが勝ちか負けかを証明するために必要な労力を示す良い指針となる．未展開のノードは証明数と反証数は未知数であるため，これらの全ノードの展開にかかる労力 (計算コスト) は等しいとして両方を 1 として計算される．現在展開されている木が図??の状態の場合，ルートノードが勝ちであることを示すためには，その状態の証明数である 2，つまり，2 つの盤面を勝ちであると証明しなくてはならない．これは図??中で証明数最小の行動である右下のノードへ遷移する行動を選んだとき，勝ちを証明するためには相手が選択可能な 2 つの盤面の勝ちを証明しなくてはいけないことを示す．また，この行動選択が最も勝ちを証明する労力が少ない可能性がある行動であることも示している．証明数と反証数のを用いた先行研究として，詰め将棋問題を解いた脊尾のアルゴリズム [?]，df-pn アルゴリズム [?] がある．

証明数・反証数をオセロゲームに適用すると，証明数はその盤面が勝ちであるために勝ちを示さないといけない先端盤面の個数の最小値となる．証明数の個数が最小の手は現在の状態を勝ちと示すために，必要な状態展開数が最も少ない手となる．つまり，展開する状態が少ないため，深く探索することができ終了盤面に近い状態を先読みすることが可能となる．逆に反証数は，その盤面が負けであるために負けを示さないといけない先端盤面の個数の最小値となる．反証数の個数が最小の手は現在の状態を負けと示すために，必要な状態展開数が最も少ない手となる．これらはその盤面が勝ちか負けかを示そうとするときの難易度の非常に良い尺度となる．

ノード n が末端ノード

a. 最終的な評価が `true`

`pn`(n)=0

`dn`(n)=∞

b. 最終的な評価が `false`

`pn`(n)=∞

`dn`(n)=0

c. 最終的な評価が不明

`pn`(n)=1

`dn`(n)=1

ノード n が内部ノード

a. 最終的な評価が `true`

`pn`(n)=子ノードの `pn` の最小

`dn`(n)=子ノードの `dn` の和

b. 最終的な評価が `false`

`pn`(n)=子ノードの `pn` の和

`dn`(n)=子ノードの `dn` の最小

図 3.1: 証明数・反証数の計算法

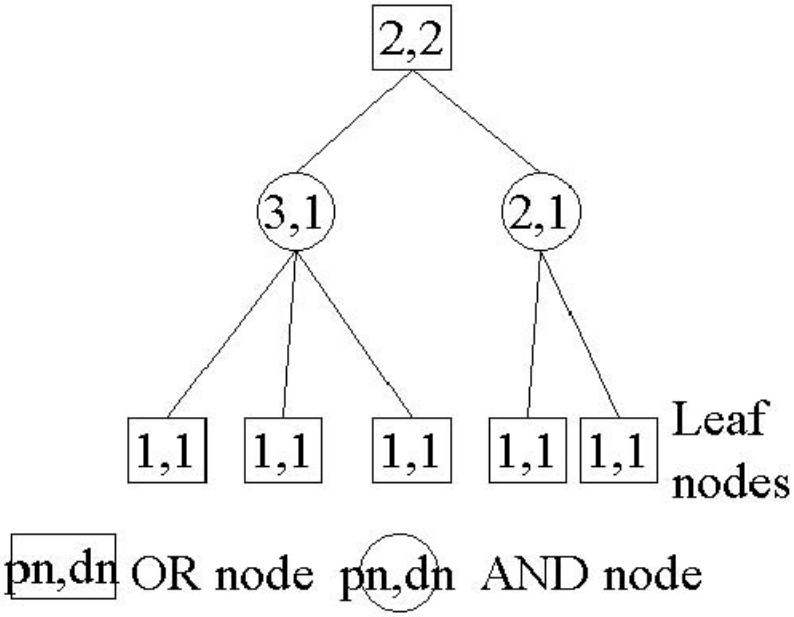

図 3.2: AND/OR 木における証明数・反証数の値

3.3 証明数・反証数を用いたPARP

証明数・反証数を PARP に適用することを考える．負けの状態は反証数が 0 となる．反証数が 0 である状態を罰状態とすると，反証数の小さいノードを優先的に探索することで罰の伝播の効率化が期待できる．証明数はその状態が勝ちであるかを示すのに必要な労力の尺度とすることができるので，先読みにより現在の状態が罰状態ではないことを判定することになり，罰回避政策を獲得する上で重要となってくる．証明数・反証数は展開されるノード数であるので，試行回数が増す毎に未知のノードが展開されてより正確な値となる．つまり，罰の判定に多くの試行回数が必要となる状態では，より正確な証明数・反証明数が求まり，先読みによる罰の伝播において，先読みするノードを選択する良い尺度となる．

証明数・反証数を用いた PARP に必要な環境を図??，行動選択前の罰ルール判定手続きを図??????，行動選択後の罰ルール判定手続きを図??に示す．実際の AND/OR 木の証明数・反証数の伝播例を図??に示す．また，そのときの証明数・反証数の伝播の順序は以下の通りとなる．

1. 制限された状態展開数の数だけ状態 (盤面) を展開 (先読み) し，そのときの証明数と

反証数の値を記憶する (図??，1).

2. 罰行動以外の行動を任意に選択し，行動選択後に更新されたエピソードの証明数と反証数をデータベースに上書きする (図??，2).
3. 次の試行で同一盤面に訪れたとき，証明数と反証数から罰の判定が容易なノードを展開 (先読み) し，更新された証明数と反証数をデータベースに上書きする (図??，3).
4. 先読中に罰が発見された場合，その証明数を ∞，反証数を 0 として計算し，更新された証明数と反証数をデータベースに上書きする (図??，4).

証明数と反証数による先読みは，証明数と反証数それぞれに閾値を用いる．図??の先読みでは，現在の状態から証明数と反証数にそれぞれ 1 を足した値を閾値として，OR ノード (自分の手番) では証明数の少ない行動をとる．AND ノード (相手の手番) では反証数の少ない行動をとる．証明数か反証明数が閾値を超えると現在の状態に戻り，再度現在の状態から証明数と反証数に 1 を足して先読みを行う．この操作を状態展開数が規定値を超えるまで行う．この手法により，勝ちと負けを判断するのに労力の少ない行動から先読みすることができる．この図??の例で発見された罰は，提案手法では 2 度目の訪問で罰を見つけることができた．同じ状態展開数を用いた従来手法での先読み (幅優先探索) では，2 度目の訪問では発見できない罰である．このように，試行回数を繰り返す毎に証明数と反証数が更新されていき，少ない労力で罰が発見できると予想される状態を深く探索できる．それにより罰の伝播が効率化される．

１状態が持つデータの種類

S_PN：証明数

S_DN：反証数

１行動（ルール）が持つデータの種類

A_PN：遷移先の証明数

A_DN：遷移先の反証数

アルゴリズムの動作に必要なもの

- 状態を記憶する領域：**長期記憶部**
- １エピソードと先読みにより展開された状態を記憶する領域：**短期記憶部**
- 短期記憶部に記憶可能な状態数：**n**

図 3.3: 提案手法のアルゴリズムに必要な環境

行動選択

先読みにより罰の伝播を行う.

`MID(S, INT_MAX, INT_MAX)`

状態 S の A_DN が 0(罰行動) 以外の行動 P_AN で, 評価の最も高い行動を選択する.

現在の状態 s と行動選択後に遷移した状態 s ’をエピソードとして記憶する

図 3.4: 提案手法罰ルール判定手続き (行動選択前)

```
MID(S,PN,DN) {
  状態 S に対して長期記憶部を参照
  if(長期記憶部に状態 S が存在する)
    状態 S に S_PN,S_DN,A_PN,A_DN を代入
  else
  A_PN と A_DN に 1 を代入
  S_PN には A_PN の合計,S は 1 を代入
  長期記憶部に状態 s を保存
  if(S_PN>PN || S_DN>DN)
    return
  if(記憶状態数が 1000 を超える)
    return
  else
    記憶状態数+=1
  while(A_PN<PN の行動が存在する || 記憶状態数が 1000 を超えない) {
    A_PN が最小の行動 a に対して,
    NEXT_PN=PN-(A_PN の合計)+A_PN
    NEXT_DN=A_DN で 2 番目に小さい値
    NEXT_S=状態 S から行動 a によって遷移する状態
  MID2(NEXT_S,NEXT_PN,NEXT_DN,&A_PN,&A_DN)
  }
  S_PN には A_PN の合計,S_DN には A_DN の最小値を代入
  長期記憶部に状態 s を保存
}
```

図 3.5: 提案手法罰ルール判定手続き (行動選択前) 内部関数 1

```
MID2(S,DN,PN,&A_PN,&A_DN) {
  while(S_DN<DN の行動が存在する || 記憶状態数が 1000 を超えない) {
    S_DN が最小の行動 a に対して,
    NEXT_DN=DN-(A_DN の合計)+A_DN
    NEXT_PN=A_PN で 2 番目に小さい値
    NEXT_S=状態 S から行動 a によって遷移する状態
    MID(NEXT_S,NEXT_PN,NEXT_DN)
  }
  S_PN には A_PN の合計,S_DN には A_DN の最小値を代入
  A_PN に S_PN,A_DN に S_DN を代入
}
```

図 3.6: 提案手法罰ルール判定手続き (行動選択前) 内部関数 1

1 試行終了後の罰の伝播

```
for(最終盤面からエピソード全て) {
  更新された A_PN と A_DN に TMP_PN と TMP_DN を代入
  if(OR ノード) {
    S_PN=A_PN の合計
    S_DN=A_DN の最小値
    長期記憶部に状態 s を保存
  }
  if(AND ノード) {
    S_PN=A_PN の最小値
    S_DN=S_DN の合計
  }
  TMP_PN=S_PN
  TMP_DN=S_DN
}
```

図 3.7: 提案手法罰ルール判定手続き (行動選択後)

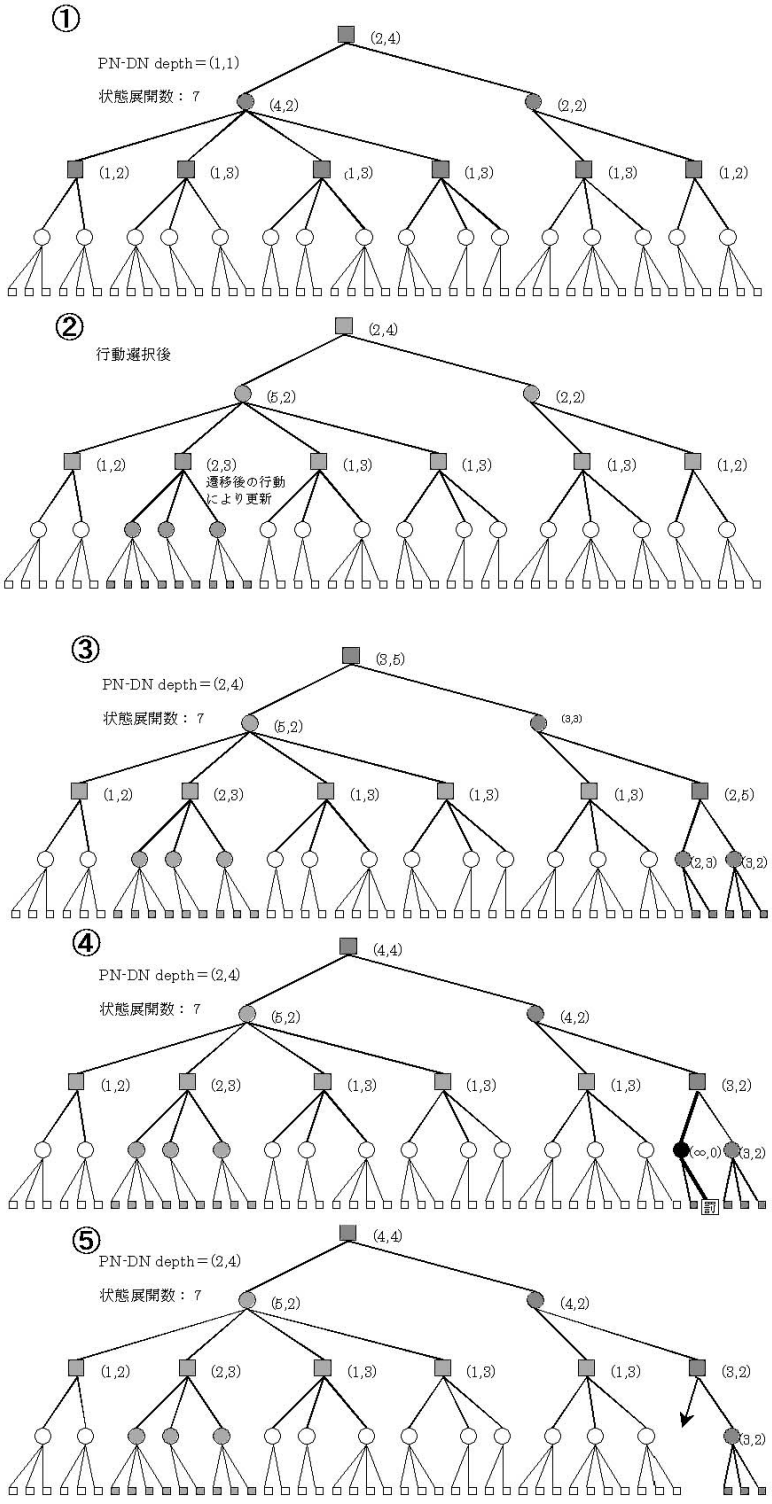

図 3.8: 証明数・反証数の伝播

第4章　PARPの従来手法と提案手法との比較実験

4.1　実験設定

本章では，従来手法と提案手法をオセロゲームの学習プログラムとして実装する．以下この2つをそれぞれ，従来手法テストプレイヤー，提案手法テストプレイヤーと呼ぶ．対戦相手にはα-β法を用いたものを使い，その評価関数にはオセロの世界チャンピオンである村上八段と宮崎四段が考案した，開放度理論 [?] を用いたものと, 石の位置により評価をつけたものを用いた．開放度理論とは裏返る石の周りの空白のマスの数であり，この数が少ない方が有利であるという考えである．図??の例でいうと，自分の手番が白として，A に置くと 3f が裏返り周りの空白のマスが 3 つであるので開放度は 3 となる．また，J に置くと 5e と 6e が裏返り其々の周りの空白のマスは 1 と 4 になり，開放度は 5 となる．したがって，A と J の手を比べると開放度が少ない A の手の方が開放度が低いので有利であると見れる．これが開放度理論の考え方である．石の位置による評価は減点法を採用する．これは序盤に石を取りすぎると不利ということを元に考えられた評価の仕方で，不利になる位置には減点される評価を大きくする．図??が実際に用いた石の位置による評価の点数である．この 2 種類の評価関数それぞれ持つ 2 つの対戦相手を設ける．また，これらの評価を基本として盤面で以下の形が存在した場合での評価も付け加えた．

- 隅 (図??)
  絶対に裏返ることの無い確定石で，一般的にとると有利になる．

- X 打ち (図??)
  相手に隅をとられやすくなるため，中盤までは悪手となることが多い．

- 山 (図??)
  図??の白の形がそれである．一見隅を取られやすそうな形であるが良形で，隅をとられる可能性は少なくほぼ確定石となる．

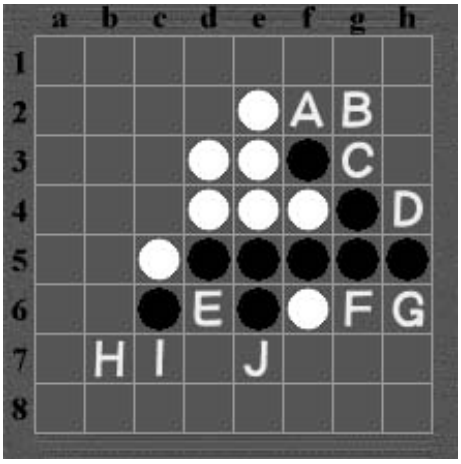

図 4.1: 開放度理論

- 翼 (図??)
  図??の白の形がそれである．相手が X 打ちをした場合，隅を 1 つ取ることができるが相手に大量の確定石をとられる場合があり悪形とされる．

- 確定石 (図??)
  勝負中にもう二度と裏返らない石のことで，隅から続く石であり，これが多いと有利となる．

これらの形を絶対に有利・不利であるとすることはできないが，高い確率 (6 割〜 9 割) で判別することができる．本研究において上記の手法を用いて構築された対戦相手は最も評価の高い手を一意にとり，フリーで入手できるプログラムの中では最強と言われている WZebra[?] の 4 手読みを相手に，7 手読みの α-β を用いて勝利することができる．

4.2 テストプレイヤーの構築

オセロゲームは 2.1 節で定義したような先読みが可能な問題である．つまり，テストプレイヤーは，現在の盤面の状態から遷移可能な状態の集合と報酬関数を計算コストと記憶容量の資源が許す限りもとめることができる．しかし，全ての最終盤面まで先読みすると，計算コストと記憶容量は膨大な量が必要となってくる．そこで，テストプレイヤーの先読みに用いる記憶容量に制限を設けて先読みをする．従来手法テストプレイヤーと提案手法テストプレイヤー共に，状態展開数が 1000 となる量の記憶容量を与えるものとす

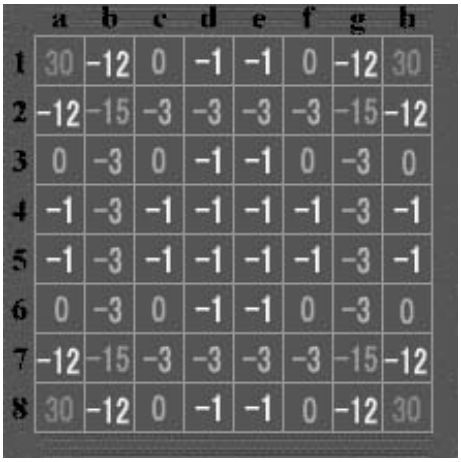

図 4.2: 石の位置による評価

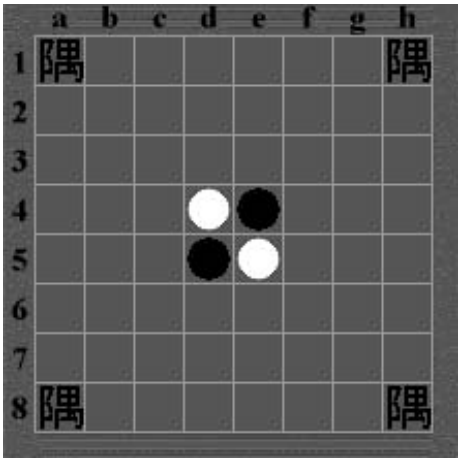

図 4.3: 隅

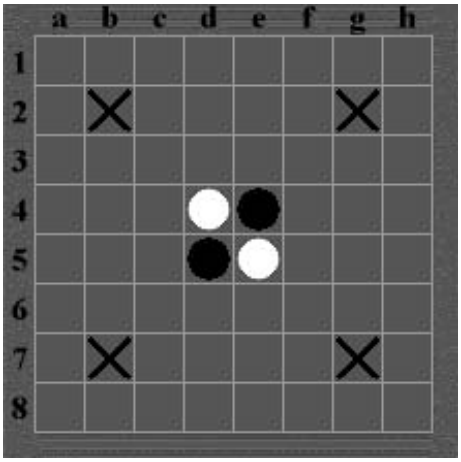

図 4.4: X 打ち

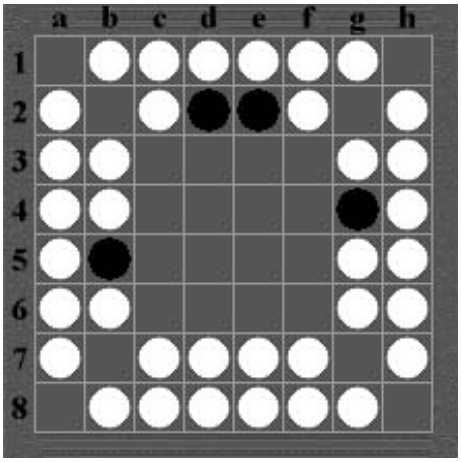

図 4.5: 山

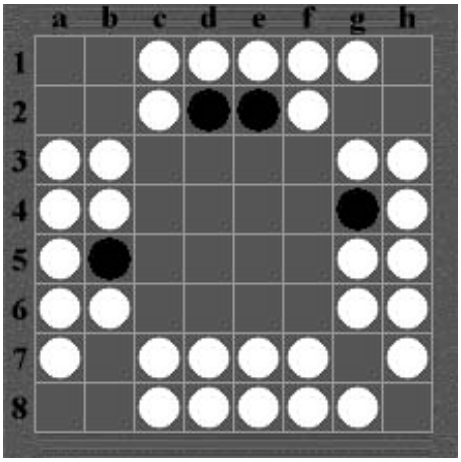

図 4.6: 翼

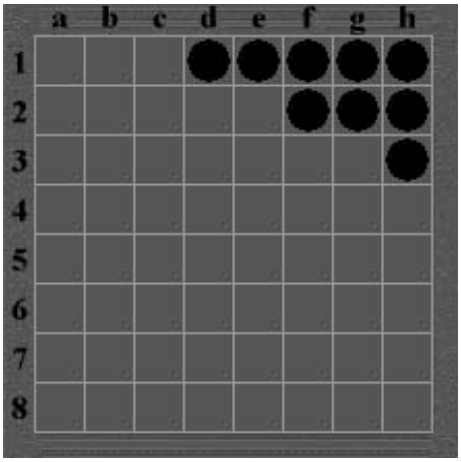

図 4.7: 確定石

る．この 1000 という数字は，行動選択の幅が最も多い中盤においても，2 手先 (次の自分の手番) まで展開するのに十分な状態数である．展開にかかる計算速度がほぼ同一であるため，これにより，従来手法と提案手法のテストプレイヤーの各状態における先読みにかかる計算時間をほぼ同一にすることができる．オセロゲームの性質上，1 エピソードは最悪を見積もっても状態遷移数 60 以内で終了する．また，オセロゲームでは同じ状態を 2 度経験することはないため，無限ループ等の永久に正の報酬 (勝ち) か負の報酬 (負け) を得られないような非合理的ルールは存在しない．したがって，負けを回避することで勝ち筋を獲得することができる．

4.3 既存知識の利用

先行研究で行われた既存知識の設定には，以下の 2 つの方法が行われている．

- **序盤から中盤にかけては棋譜を用いて定石とした**
  過去に他で行われた試合 (10 万試合) の棋譜の中で，学習者の手番が勝利した盤面へ遷移する可能性が最も高い手をとるようにした．これは罰が伝播されるまで多くの試行回数が必要である序盤の行動選択に使われた．

- **対戦相手の評価関数の利用**
  対戦相手の評価関数を用いて対戦相手の評価値が，ある一定の閾値以上有利 (学習者が不利) と判断される手を準罰とし，ヒューリスティックな評価による罰 (準罰) として用いた．先行研究ではこの閾値の設定によって，罰回避政策形成アルゴリズムの勝ち筋獲得までに必要な試行回数が大きく左右された．

本研究で行った実験では，上記の対戦相手の評価関数による準罰のみを採用した．これにより，罰と判断される状態を先行研究と比べて減らすことができる．従来手法の問題点であった，ヒューリスティックな評価の依存性の軽減 (準罰が少ない設定で罰回避政策を獲得する) という目的からこの設定を用いた．この設定で従来手法との必要な試行回数を比較する．先行研究と同様，対戦相手の評価関数を用いて一定値以下なら準罰と設定した．この一定値を，以下では準罰の閾値と呼ぶ．準罰の閾値を変更させることにより，準罰と判定される状態数を変化させることができる．閾値を高くすると，準罰と判定される状態数が減り，閾値を低くすると，準罰と判定される状態数が増える．つまり，罰回避政策形成アルゴリズムにおいて，準罰の閾値が高いと準罰への依存が高くなり，準罰の閾値が低いと準罰への依存が低くなる．

表 4.1: 開放度を評価関数としてもつ対戦相手に対して，勝ち筋を獲得するまでに必要な試行回数

	従来手法			提案手法		
	5 手読み	6 手読み	7 手読み	5 手読み	6 手読み	7 手読み
閾値-10	(50)	309	(69)	(23)	(23)	(23)
閾値-15	139	817	1053	49*	216*	895*
閾値-20	337	319	1456	430	356	1236*
閾値-25	597	648	1689	1591	204*	1431*
閾値-30	1622	1166	3567	3219	482*	3983
閾値-35	3022	1764	-	5785	397*	-
閾値-40	3774	2145	-	310*	751*	4311*

- -:1 万試行しても勝ち筋を獲得できなかった
- *:従来手法に比べ提案手法の試行回数が削減された
- ():開始状態までが準罰状態となり勝ち筋を獲得できなかった

4.4 提案手法と従来手法との比較実験

4.4.1 実験 1:証明数と反証数による先読み

従来手法のテストプレイヤーと提案手法のテストプレイヤーを開放度と石の位置による評価関数をもつ対戦相手とそれぞれ戦わせた．その際，テストプレイヤーは対戦相手と同じ評価関数を持つとする．そのときの準罰の閾値は-10 から-40 まで 5 刻みで行う．開放度を評価関数としてもつ対戦相手に，従来手法と提案手法のテストプレイヤーの勝ち筋を得るために必要な試行回数は，それぞれ表??のようになる．石の位置による評価をもつ対戦相手の結果は表??のようになる．表中で開始状態までが準罰状態，つまり，獲得すべき勝ち筋が誤った罰の伝播により罰判定されたものは，行動を罰行動から選ばなくてはいけなくなる．そうなると，行動選択は現状態でとれる全行動から何の指針も無しに選ばなくてはならず，全探索に近い形となる．このような状態になると勝ち筋を獲得するのに必要な試行回数 (計算コスト) が膨大となるので，本研究で行った実験ではこのような状態になると罰回避政策を獲得できなかったとする．

図??〜図??は，各試行毎に，罰状態へ至るまでの手数を示したグラフである．この数

表 4.2: 石の位置を評価関数としてもつ対戦相手に対して，勝ち筋を獲得するまでに必要な試行回数

	従来手法		提案手法	
	5 手読み	6 手読み	5 手読み	6 手読み
閾値-10	33	47	21*	59
閾値-15	234	321	398	221*
閾値-20	435	239	137*	178*
閾値-25	249	209	213*	79*
閾値-30	221	482	198*	32*
閾値-35	298	544	272*	98*
閾値-40	191	887	99*	321*

値が 60 に達すると終了時まで罰状態に陥らないということを示す．つまり，勝ち筋を獲得したこととなる．図??，図??，図??は，実験で試行回数が削減された例で，図??と図??は，逆に試行回数が増えた例を示したものである．罰状態へ至るまでの手数が一度上がってから下がるのは，探索木の中で現在辿っている筋道が罰の伝播により，罰状態となり，次の試行ではその前の状態から別の行動を選択していることを示す．つまり，グラフの上下の振れ幅が大きい程，実際に行った行動 (手) が罰の伝播により罰行動 (負ける手) であることが判り，違う手を選ぶようになったことを表している．

また，表??は，対戦相手とは異なるヒューリスティックな準罰を用いた結果である．テストプレイヤーに用いた評価関数は石の位置による評価で，開放度による評価関数を持つ相手と対戦させた．

4.4.2 実験 2:先読みの深さに閾値設定した比較実験

実験結果である表??，表??，表??において，従来手法と提案手法を比較する．少ない準罰を設定する (準罰の閾値を小さくする) と，殆どの場合ででも勝ち筋を発見するまでの試行回数が削減されていることがわかる．試行回数が削減されるパターンとして図??の約 130 試行目 (点 B) から 330 試行目 (点 C) のように，多い手数 (この例では約 20 手) の行動後に発見される罰の伝播によって罰状態となる盤があることがあげられる．今回の例では，点 A と点 B と点 C の盤面が同一であった (グラフの点は罰状態であるので，正確に

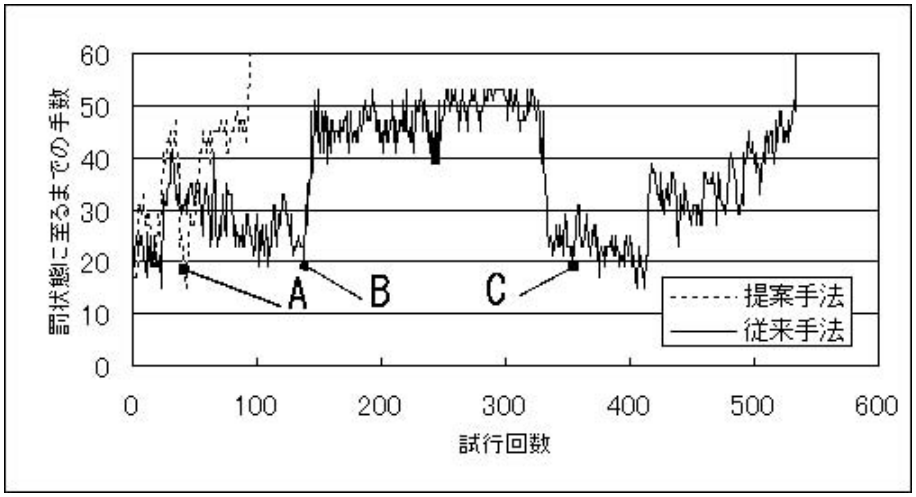

図 4.8: 罰状態に陥るまでの手数 (表??の閾値-35)

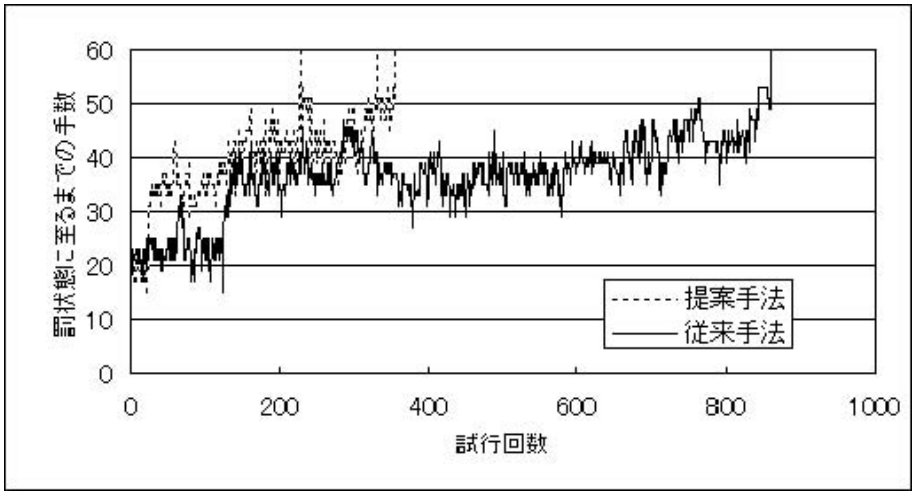

図 4.9: 罰状態に陥るまでの手数 (表??の閾値-40)

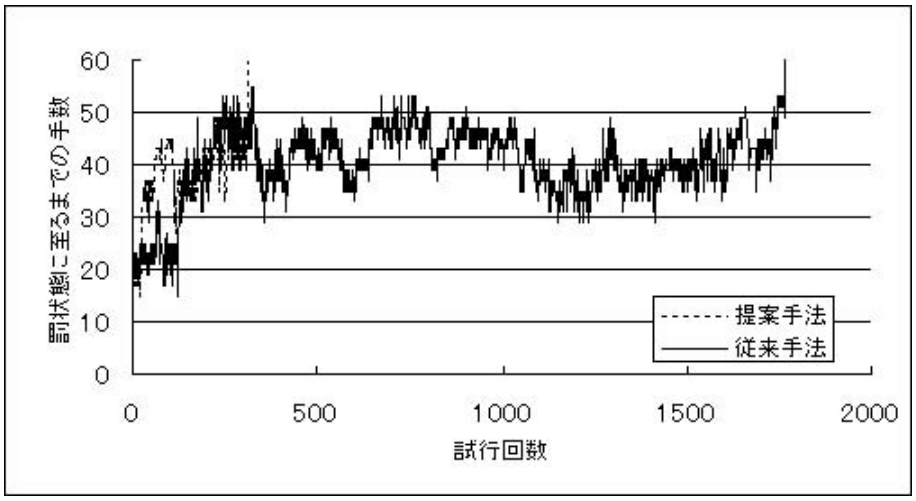

図 4.10: 罰状態に陥るまでの手数 (表??の閾値-35)

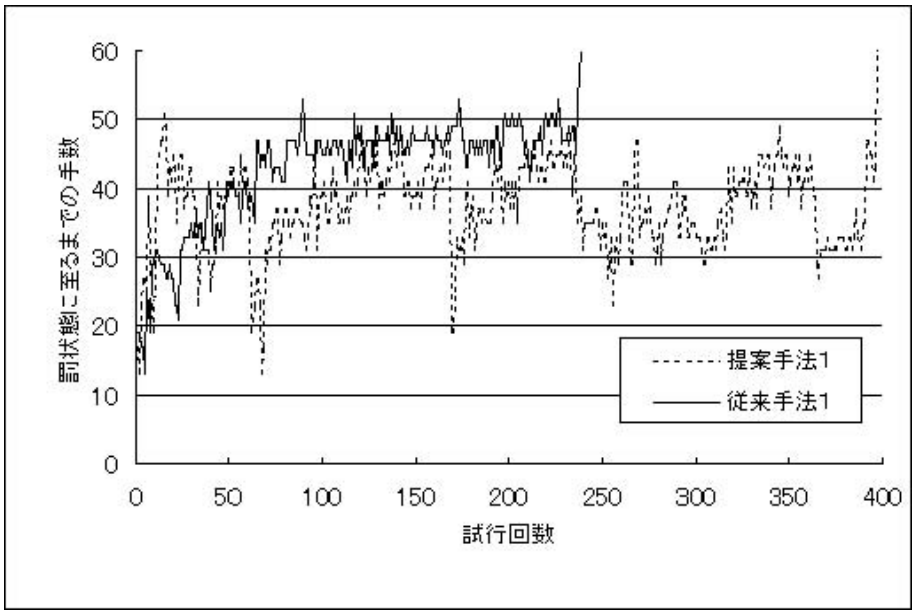

図 4.11: 罰状態に陥るまでの手数 (表??の閾値-15)

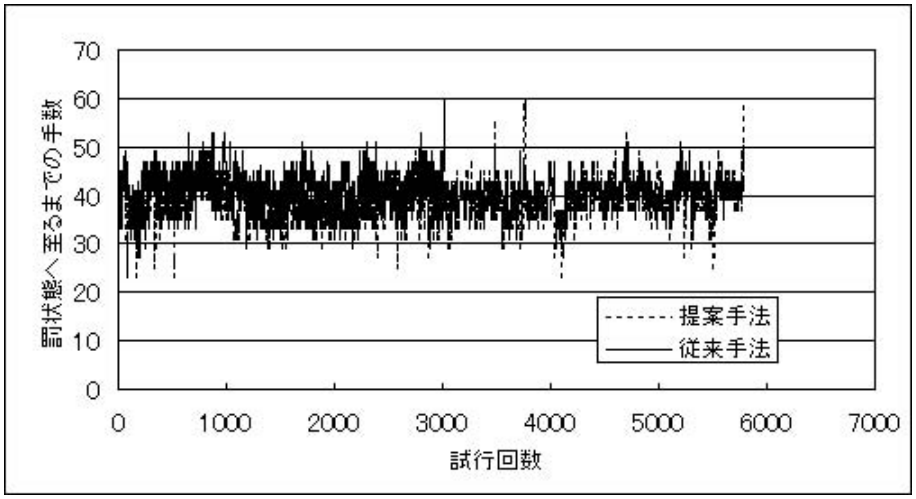

図 4.12: 罰状態に陥るまでの手数 (表??の閾値-35)

表 4.3: 対戦相手とは異なる評価関数による準罰での，勝ち筋を獲得するまでに必要な試行回数

	従来手法		提案手法	
	5 手読み	6 手読み	5 手読み	6 手読み
閾値-10	(40)	(76)	(21)	(25)
閾値-15	1493	238	78*	398
閾値-20	803	534	133*	95*
閾値-25	763	610	636*	197*
閾値-30	1602	668	1052*	177*
閾値-35	265	861	248*	356*
閾値-40	265	1761	322	314*

表 4.4: 深さに閾値を用いた試行回数

	従来手法			提案手法		
	5 手読み	6 手読み	7 手読み	5 手読み	6 手読み	7 手読み
閾値-10	(50)	309	(69)	(23)	(23)	(23)
閾値-15	139	817	1053	49*	216*	895*
閾値-20	337	319	1456	430	356	1236*
閾値-25	597	648	1689	1268	204*	1031*
閾値-30	1622	1166	3567	1489*	482*	2653*
閾値-35	3022	1704	-	1839*	703*	3230*
閾値-40	3774	2145	-	310*	751*	5431*

は 1 手前の状態)．提案手法では点 A の時点でこの盤面が罰である事を判定でき，従来手法だと，点 B から点 C の約 200 試行を必要とした．このように，多い手数の行動後の罰を伝播する必要がある状態が存在するとき，罰の伝播が効率化された提案手法では判定までに必要な試行回数は激減される．しかし，いくつかの場合において従来手法より試行回数が増加する傾向のものが出ている．これは，伝播を効率化することにより，相手が読み切れていない有効な手を，提案手法テストプレイヤーが罰として伝播してしまい，別の行動を取るようになってしまったことが理由として考えられる．すなわち，提案手法が罰であると判定した状態が，相手がその状態を自分にとって有利と読みきれないために，実際には勝てる状態であったということが考えられる．その結果，簡単に勝てる筋道を見落としてしまい，試行回数が増加したと考える．このような問題は対戦相手と学習者の盤面の価値の相違により生じる．そこで先読みに深さの閾値を設定して行ったのが表??である．深さに閾値を設定することにより，対戦相手より深い読み (罰と判定しなくとも勝てる行動を罰としてしまう) をしてしまうことを防ぐことになると考える．

4.5 考察

表??，表??，表??を見て，多くの場合で試行回数が削減されていることがわかる．つまり，先読みによる罰の伝播が効率的に行われたと推測できる．また，試行回数が増加する場合においても，表??により従来手法に比べ試行回数が増加するパターンが減少していることから，先読みの判定では相手が読み切れていない罰を伝播してしまっていること

に原因があるといえる．対戦相手が学習者の先読みにより発見した罰 (対戦相手が勝利する手) をとらなかったとき，その罰は正しいものであってもその罰の発見に時間 (試行回数) を必要とすれば，罰回避政策の獲得に時間がかかってしまうこととなる．つまり，伝播しない方が少ない時間 (試行回数) で勝ち筋を獲得できる罰が存在するのである．この問題を回避するために行われた探索の深さの閾値の設定であるが，本研究の環境では評価関数と読み手数が明白な対戦相手であり，それよりすこし大きめに深さの閾値を設定すればよかった．しかし，実際には相手の読み手数は判らないことが多く，最初から深さの閾値を適切に設定することは困難である．しかし，このような状況は対戦相手が弱いために生じる問題であり，十分に強い対戦相手とする場合は，このような問題が生じることは少ないと考えられる．

第5章　PARPの類似問題への対応

5.1　ニューラルネットワークを用いたPARP

PARP は特定の相手に対して学習により罰回避政策のうちの 1 つを獲得する手法である．よって，学習結果を汎化して類似盤面に対して有効な手をとることはできない．また，いままでの実験で用いた対戦相手は最も評価の高い手を一意にとるものである．一意の手をとる相手の場合，獲得すべき勝ち筋は 1 つの筋道となった．しかし，複数の手の中から確率的に手をとる相手のとき，勝ち筋を獲得するためには相手によって分岐する行動全てに対して罰回避政策を獲得しなくてはならない．したがって，必要な試行回数が膨大となる．複数の手をとる 4 手読み相手に実験をしたが，1 万試程度では罰回避政策を獲得することはできなかった．そこで本章では異なる対戦相手に対して，過去の類似問題での学習成果を汎用するために，ニューラルネットワーク (図??) の評価による準罰を用いた手法を提案する．現在までに伝播により確定した罰盤面を教師信号とし，ニューラルネットワークの中間層と出力層の重みを更新していく．ニューラルネットワークによる評価によってもたされた準罰は，ヒューリスティックによる評価によるものに比べ信頼性が低い．つまり，通常の伝播を行えば誤った伝播が頻繁に行われ，獲得すべき勝ち筋までが準罰となることが多くなる．そこでニューラルネットワークによってもたされる準罰に信頼度を設定する．この信頼度とはニューラルネットワークの学習の進行から，罰が正しいである確率を推測してヒューリスティックによって設定する．誤った伝播が頻繁に行われるようであれば，低く設定を行う必要がある．準罰の信頼度の計算方法は図??に示す．現在展開している準罰の信頼度は任意の値で，ニューラルネットワークの学習の進行によって決められる．準罰の信頼度に閾値をもちいて，準罰の伝播を行うかを判定する．

5.2　実験結果

本研究で用いたニューラルネットワークは入力層を 192(1 つのマスに白・黒・空の bool 値が入る) をとり，中間層の数は 50，出力層は罰であるなら 1 を返すものとした．ニュー

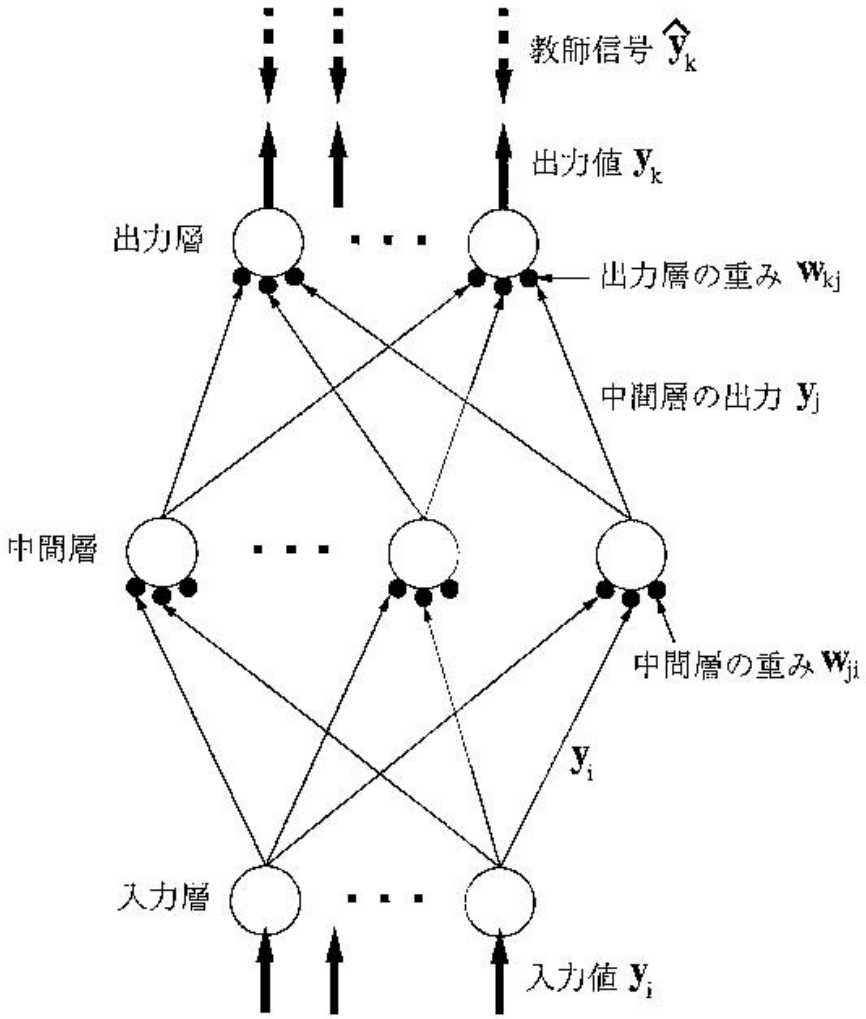

図 5.1: ニューラルネットワーク

- S:現在の状態
- S_i:S の子ノード
- Po:S の子ノード
- $Po(S)$:`or` ノードにおいて，状態 S が罰である確率
- $Pa(S)$:`and` ノードにおいて，状態 S が罰である確率
- S が先端ノード
  - 罰であるとき
    $Po(S) = Pa(S) = 1$
  - 準罰であるとき
    $Po(S) = Pa(S) = e$　　(e は $0 < e < 1$ の任意の値)
  - 上記以外のとき
    $Po(S) = Pa(S) = 0$
- S が末端ノード
  - $Po(S) = \prod_c Pa(S_c)$
  - $Pa(S) = 1 - \prod_c \{1 - Po(S_c)\}$

表 5.1: 準罰の信頼度の計算

ラルネットワークを用いた準罰を用いて，4.1 節で用いた 7 手読みの対戦相手に，ニューラルネットワークの重みの初期値をランダムで 5 種類 (player1〜5) とり，ヒューリスティックによる準罰の閾値を-40(準罰が少ない) の環境を元に実験を行った．ニューラルネットワークを用いない場合と，ニューラルネットワークの準罰に信頼度を設定しない場合と，信頼度を設けた場合を表??に示す．信頼度は試行回数を n とすると，$n \times 0.0005$ とし，閾値は 0.5 とした．これは 1000 試行でニューラルネットワークの準罰が有効になることを示す．また，信頼度の上限を 0.8 とした．この値はニューラルネットワークによる準罰をうける行動の個数が，その盤面でとれる行動のうち 3 つ以内でのみ伝播を行うことを示す．それ以上の行動がニューラルネットワークにより準罰と判定されても，伝播は行われない．表??により信頼度を設けないと誤った準罰の伝播が行われ，獲得すべき勝ち筋までが準罰行動となってしまうことがわかる．また，7 手読みを相手に行った学習結果を用いて，信頼度の初期値を 0.5 とし，6 手読みを相手に行った実験結果が表??である．この 7 手読みと 6 手読みは，先端盤面の手番が違うので異なる対戦相手とみなすことができる．学習者が (対戦相手の評価基準で) 最善手を取ったとき，7 手読みは対戦相手の最善手が先端あり，6 手読みは対戦相手の有利な盤面がある．

また，ある評価関数のもと行動の選択が複数ある相手に対しての実験を行った．対戦相手は 3 手読みの α-β で，開放度の評価を用いて評価値の高い上位の手からランダムに手をとる．この対戦相手では，行動を複数とる相手には勝ち筋のうちの 1 つを獲得する，通常の罰回避政策形成アルゴリズムでは膨大な試行回数が必要となる．そこで，行動を複数とる相手と戦うには盤面の汎化能力必要となり，実装したニューラルネットワークの実際の汎化能力を示すものである．ニューラルネットワークで学習前と学習後の 3 手読みの複数の手をとる相手と学習機能を停止させ，100 回対戦させたときの戦績を表??に示す．これは，異なる対戦相手に過去の学習結果のみで勝ち筋を得れるかを試した実験である．この実験での設定では学習者は 2 手から 3 手読みの先読みが使われているので，学習前で 100 回中 23 回勝ち筋を獲得することができる．

5.3 考察

罰回避政策形成アルゴリズムでヒューリスティックな評価による準罰であるが，ヒューリスティックな評価による準罰と違い，試行回数が増えることにより学習が進み「準罰が正しい」とする確率が変化する．表??の信頼度を設定しない場合，誤った罰の伝播が行われて勝ち筋がまでが準罰となるケースが多くなることから，信頼度の設定は必要不可欠で

表 5.2: 7 手読み相手にニューラルネットワークの準罰を用いた試行回数の比較

	NN の準罰無し	NN の準罰有り (信頼度無)	NN の準罰有り (信頼度有)
player1	4311	(201)	-
player2	4311	(483)	4977
player3	4311	(23)	3203*
player4	4311	(68)	5439
player5	4311	(540)	2191*

表 5.3: 6 手読み相手にニューラルネットワークの準罰を用いた試行回数の比較

	NN の準罰無し	NN の準罰有り
player1	751	89*
player2	751	732*
player3	751	865
player4	751	(45)
player5	751	569*

表 5.4: 複数の中から行動を選択する対戦相手との勝率

	%勝率
学習前	23
player1	74
player2	56
player3	44
player4	67
player5	71

あることがいえる．また，信頼度を設定した結果であるが，前節で示した上限を 0.8，信頼度の閾値を 0.5 とした実験結果では試行回数が削減されケースがあるが，上限をこれより下げる, 又は信頼度の閾値を上げる (ニューラルネットワークによる準罰行動が 3 つより大きい数でも伝播される) と，政策獲得までの試行回数は殆ど変化しなくなる (ニューラルネットワークを用いないときと同じになる場合が増える)．逆に設定を変えると上限を 0.9 とすると，開始状態までが準罰となるケースが多発した．ニューラルネットワークによる準罰は，生成の際にゲームのルール以外の知識を必要としないため汎用性が高い．しかし，準罰をニューラルネットワークだけに頼ることは，本研究で行った程度の試行回数ではできなかった．異なる相手に対して行った 2 つの実験であるが，6 手読み (勝ち筋の獲得に試行回数が必要なケース) では，表??の罰回避政策を獲得するまでに必要な試行回数は削減はされるケースの方が多いが，やはり誤った伝播が行われて罰回避政策が獲得できないケースが出てくる．誤った伝播が頻繁に行われるのであれば，信頼度の見直しが必要となってくる．また，表??ではニューラルネットワークで判定される準罰を回避する政策をとることにより，勝率を上げることができた．これは行動選択を決める上でニューラルネットワークによる準罰が有効に働いていることがわかる．ニューラルネットワークによる準罰は伝播に用いるときは，信頼度を低く設定して行動選択の指針として活用することが重要となってくるといえる．

第6章　おわりに

6.1　まとめ

本研究では，証明数と反証数を用いた先読みと，ニューラルネットワークの評価による罰を用いた罰回避政策形成アルゴリズムを提案した．実装例としてオセロゲームを用いて従来手法との比較実験の結果により，証明数と反証数を用いて罰回避政策の獲得に必要な試行回数が多くのケースで削減されることを示した．また，試行回数が増加するケースについても，対戦相手が気が付いて居ない罰 (負ける行動) を学習器が先読みで発見してしまい，伝播しなくとも勝てる罰を伝播してしまっていることが原因であると，深さに閾値を設定した先読みの実験結果により示した．ニューラルネットワークについても，信頼度を設定することにより罰回避政策形成アルゴリズムに用いる事ができることと，初見での対戦で行動選択の指針として用いる事に有効なことを，行動選択の幅が複数ある相手と対戦する実験により示した．

6.2　今後の課題

今後の課題としては，本研究で新たにでてきた問題である以下の 2 点である．

- 伝播しない方が少ない試行回数で勝ち筋を獲得できる罰が存在する．
- ニューラルネットワークの評価は十分に学習しなければ不正確であり，罰の伝播に用いると支障をきたす可能性が高い．

1 の問題は，対戦相手より深く読みすぎない事により回避できたことを示したが，実際には対戦相手の読みの深さは解らないものである．また，読みの深さに閾値を設けたとしても，盤面の評価基準によっても相手に発見できにくそうな良手の存在すればこの問題はでてくる．完全にこの問題を解決しようとすれば，対戦中に相手の能力を解析する手法が必要となる．2 の問題については，ニューラルネットワークの評価による罰は行動選択の指針として用いる事は有効であることを示したが，罰の伝播に用いる場合，誤った伝

播が行われる可能性が高くなる．本研究で用いた設定ではあくまでヒューリスティックな罰の補助的なものであった．ニューラルネットワークの設定はそのゲームの深い知識や棋譜データベースは不要であるので,「準罰が正しい」確率が変動する準罰を上手く用いる手法があれば，完全にヒューリスティックによる罰を排除することができる．ニューラルネットワークを用いた罰回避政策形成アルゴリズムは，ゲームのルールと勝ち負けの報酬を設定するだけでより強力なオセロプレイヤーの作成が期待される．

謝辞

本研究を進めるにあたって，熱心な御指導，有益な御助言を賜った東条敏教授には厚く御礼申し上げます．さらに終始熱心に御指導を賜りました永田祐一助手には大変感謝しております．本研究に際し，有益な御助言を頂きました鳥澤健太郎助教授には感謝致します．最後になりましたが，本学知識工学講座の皆様に厚く御礼申し上げます．

参考文献

[1] 宮崎和光, 山村雅幸, 小林重信 k-確実探査法: 強化学習における環境同定のための行動選択戦略 人工知能学会誌, Vol. 10, No. 3, pp454–463, 1995.

[2] C.J.H.Watkins and P.Dayan Technical note: Q-learning Machine Learning Vol. 8, pp. 55–68, 1992.

[3] 宮崎和光, 山村雅幸, 小林重信強化学習における報酬割当ての理論的考察 人工知能学会誌, Vol. 9, No. 4, pp580–587, 1994.

[4] R. S. Sutton and A. Barto Reinforcement Learning: An Introduction A Bradford Book, The MIT Press, 1998.

[5] 宮崎和光, 荒井幸代, 小林重信 POMDPs 環境下での決定的政策の学習 人工知能学会誌, Vol. 14, No. 1, pp148–156, 1999.

[6] 宮崎和光, 坪井創吾, 小林重信 罰を回避する合理的政策の学習 人工知能学会誌, Vol. 16, No. 5, pp148–156, 2001.

[7] R. S. Sutton Learning to Predict by the Method of Temporal Differences Machine Learning Vol. 3, pp. 9–44, 1998.

[8] G. Tessauro TD–Gammon, A self-teaching backgammon program, achieves master–level play Neural Computation 6, pp. 215–219, 1994.

[9] A. V. Leouski What a Neural Network Can Learn about Othello, Technical Report 96–10 Department of Computer Science University of Massachusetts Amhert, 1996.

[10] J. Baxtor, A. Tridgell, L. Weaver KnightCap A chess program that learns by combining TD(*lambda*) with game–tree search, Proc. of the 15th International Conference on Machine Learning, pp. 28–36, 1998.

[11] 宮崎和光, 坪井創吾, 小林重信 罰回避政策形成アルゴリズムの改良とオセロゲームへの応用 人工知能学会誌, Vol. 17, No. 5, pp548–556, 2002.

[12] McAllester, D. A. Conspiracy Numbers for Min–Max Search Artificial Intelligence, Vol. 35, pp287–310, 1988.

[13] Seo, M. The C* Algorithm for AND/OR Tree Search and Its Application to a Tsume–Shogi Program Master's Thesis, Department of Information Science, University of Tokyo, Japan, 1995.

[14] 長井歩, 今井浩df–pn アルゴリズムの詰将棋を解くプログラムへの応用 情報処理学会論文誌, Vol. 43, No. 6, pp1769–1777, 2002.

[15] 谷田邦彦図解早わかりオセロ 日東書院, 2002.

[16] WZebra http://www.radagast.se/othello/download.html