ゲーム情報学 6－5
(2001. 6. 7)

将棋局面進行度の定量化法の比較評価

川田 敦史　　福田 賢司　　中野 良平
名古屋工業大学

将棋には，ゲームの進行状況を表す尺度として序盤，中盤，終盤という概念がある．本研究では，学習による局面進行状況の定量化を試みた．学習による定量化に関する従来法では局面の様子が多岐にわたる終盤において誤差が大きくなることが指摘されている．そこで，学習のモデルと学習アルゴリズムを工夫して上記の問題点を克服することを試みた．さらに従来法では，局面から将棋の進行状況を特徴づける要素群を抽出していた．本研究では，自律的な学習を目指して，盤面の駒の位置と持ち駒の原情報を入力としたとき，どの程度の定量化が可能か確かめた．

Conparison of learning of stage evaluation in shogi

Atsushi Kawata,　Kenji Fukuda　and Ryohei Nakano
Nagoya Institute of Technology

A stage is an important concept in shogi, which states how far a game has proceeded from the beginning. We tried to quantify such a stage in shogi. One existing method has shortcomings in evaluation precision. Thus, we tried to overcome the problem by employing a nonlinear neural-network model and a fast learning algorithm. Moreover, we tried learning without using any characteristic elements in shogi.

1 はじめに

将棋には，ゲームの進行状況を表す尺度として序盤，中盤，終盤という概念がある．指し始めから駒がぶつかるまでを序盤，そこから寄せに入るまでを中盤，寄せから投了までを終盤といい，それぞれにおいて戦略や方針が異なる．序盤，中盤では，駒損をしないことが大事であるが，終盤は駒損より相手の王様を詰めることを最優先とする．

コンピュータに将棋を指させる際も，局面の進行状況を何らかの方法で定量化し，現局面がどのような状況であるのか把握させる必要がある．現在のコンピュータ将棋では，人間が何らかの基準を決め，プログラムに作り込むことで将棋の進行状況を把握させている [2][3]．本研究では，これを学習によって実現することは出来ないかと考え，いくつかの方法で進行状況の定量化を試み，その性能を評価する．

2 学習による局面進行度の定量化

学習による局面進行度の定量化は，加藤らによって，次のような方法で試みられた [4]．将棋の進行状況を特徴づける要素群 (表 1) を局面から抽出し，それを入力とする．用いるニューラルネットトモデルは，単層パーセプトロンで，学習には BP(Back Propagation) 法を用いる．加藤らはこの方法を用いることで，ある程度の定量化に成功している．

表 1: 特徴要素

要素	説明
$X[1]$	8，9 段目の先手の歩の数
$X[2]$	7 段目の先手の歩の数
$X[3]$	6 段目の先手の歩の数
$X[4]$	5 段目の先手の歩の数
$X[5]$	4 段目の先手の歩の数
$X[6]$	先手に守りの香がいる
$X[7]$	先手に攻撃の香がいる
$X[8]$	先手に守りの桂がいる
$X[9]$	先手に攻撃の桂がいる
$X[10]$	先手陣の先手の銀の数
$X[11]$	中央の先手の銀の数
$X[12]$	後手陣の先手の銀の数
$X[13]$	後手陣の先手の成り銀の数
$X[14]$	先手陣の先手の金の数
$X[15]$	中央の先手の金の数
$X[16]$	後手陣の先手の金の数
$X[17]$	先手陣左に先手王がいる
$X[18]$	先手陣中央に先手王がいる
$X[19]$	先手陣右に先手王がいる
$X[20]$	中央に先手王がいる
$X[21]$	後手陣に先手王がいる
$X[22]$	先手陣の先手の駒の数
$X[23]$	先手陣の後手の駒の数
$X[24]$	先手の大駒が成っている
$X[25]$	先手の持ち駒の数
$X[26]$	先手の持ち駒の金と銀の数
$X[27]$	当たっている駒の数
$X[28]$	ぶつかっている駒の数
$X[29]$	先手王近傍 24 マスの先手の金の数
$X[30]$	先手王近傍 24 マスの先手の銀の数
$X[31]$	先手王近傍 80 マスの後手の駒の数
$X[32]$	定数

2.1 モデル及び学習アルゴリズムの変更

単層パーセプトロン学習では，局面の様子が多岐に渡る終盤において誤差が大きくなることが指摘されている [4]．そこで特徴要素に関して上記の方法をベースにして，学習モデルを 3 層パーセプトロンに変更し，より誤差の少ない定量化を試みる．学習アルゴリズムとしては，一般に知られている BP 法は遅いので，誤差局面の曲率も考慮に入れた 2 次の学習法である BPQ(Back Propagation based on Partial Quasi-Newton) 法 [5] を用いて学習を行う．

表 2: 追加する特徴要素

要素	説明
$X[32]$	先手王近傍 8 マスに利きがある後手の駒の数
$X[33]$	先手王近傍 24 マスの後手の駒の数
$X[34]$	先手王近傍 48 マスの後手の駒の数
$X[35]$	定数

2.2 入力要素の追加

表 1 をみると，玉近傍における敵駒の情報が少ないように思われる．終盤の特徴の一つとして，自玉周辺に相手の駒が多いことが挙げれると思うので，当面，玉の近傍に関する情報 3 つを新たな入力要素として追加してみた (表 2)．

2.3 ニューラルネットの入力要素について

これまで述べた方法では，人間が将棋の進行状況を特徴づけると思われる要素を決め，それをニューラルネットの入力要素としていた．このようにすることで，学習モデルの縮小が図れ，効率よく学習が進むと思われる．

しかしながら，できるだけ自律的に計算機に将棋を学習させることを考えると，そのような特徴要素群がトップダウンに与えられることは期待できない．そこで，将棋に関する専門的な知識を用いないで，盤面上の駒の位置や持ち駒の情報だけを直接符号化する方法を用いて，学習がどこまで可能か確かめた．

符号化の具体的な方法は次のようなものであ

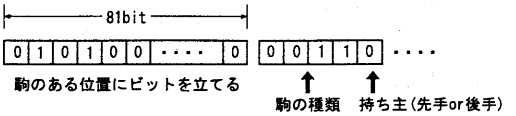

図 1: 符号化の方法

表 3: 駒の種類の表現方法

駒の種類	表現方法
歩	0001
香	0010
桂	0011
銀	0100
金	0101
銀	0110
玉	0111
と	1000
成香	1001
成桂	1010
成銀	1011
馬	1101
竜	1110

る [1]．9×9 のマス目を 81 ビットで表し，駒のあるマスに 1，そうでないマスに 0 を入れる．次に駒の種類 (14 種類) として 4 ビット，先手の駒か後手の駒か示す 1 ビット，合わせて 5 ビットを各駒それぞれ (40 枚) に割り当てる．そうして符号化された 281 ビット の各ビットをニューラルネットの入力とする．ニューラルネットには 3 層パーセプトロンを用い，学習アルゴリズムとして BPQ 法を用いる．

3　実験方法

将棋局面進行度の定量化を行う際，教師信号は開始局面を 0.1，終了局面を 0.9 とする線形の出力とした．実験にはプロの指した棋譜を用いる．用いた棋譜の数は 200 で，戦形は偏りがないようにして，手数は比較的短手数のもの (平均手数 : 約 100 手) を選んだ．評価の方法は以下とする．

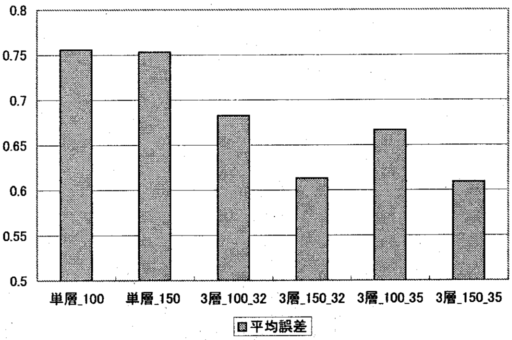

図 2: 一棋譜当たりの平均誤差

1. 棋譜を学習データと評価データに分ける．
2. 学習データを用いてニューラルネットを学習する．
3. 評価データを用いて一棋譜当たりの誤差を求める．

上記の方法を学習データ，評価データの分け方を変えて繰り返し，結果を比較する．

なお，BPQ 法はバッチ型の学習アルゴリズムであり，単層モデルの学習には，オンライン型ではなくバッチ型の SD 法 (Steepest Descent-Method) を使用した．

4　実験結果

4.1　単層と 3 層の違いによる性能比較

学習モデルが単層の場合と 3 層の場合について性能の比較を行う．図 2 は学習データが 100 棋譜と 150 棋譜のものでそれぞれ学習を行った後，別に用意した評価データ 50 棋譜の一棋譜当たりの平均誤差を計算したものである．グラフをみると単層モデルよりも 3 層モデルの方が誤差が少ないことがわかる．また，単層ではサンプル数を増やしても誤差の減少があまりみられないが，これはモデルが単純なためと思われる．これに対して 3 層では非線形の写像が可能なため，サンプル数を 100 から 150 へ増やすと誤差が大きく減少している．

さらに 3 層モデルでは特徴要素を 3 個追加

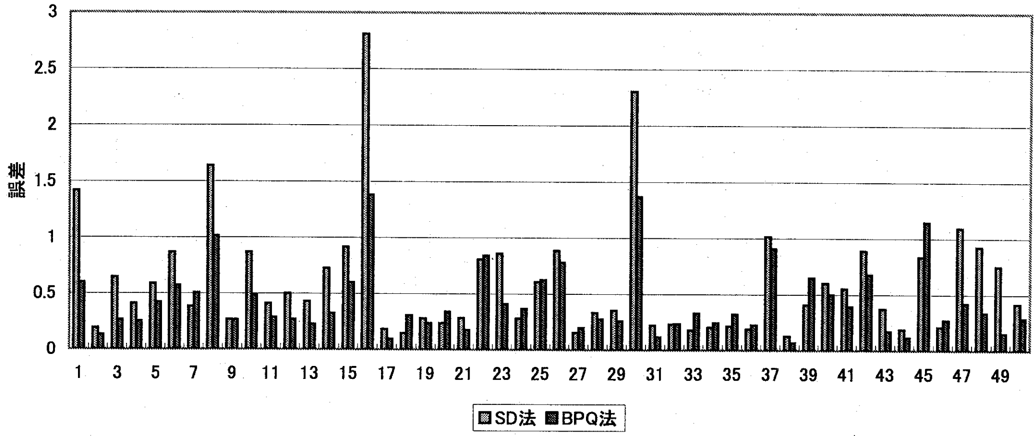

図 3: 各棋譜の誤差

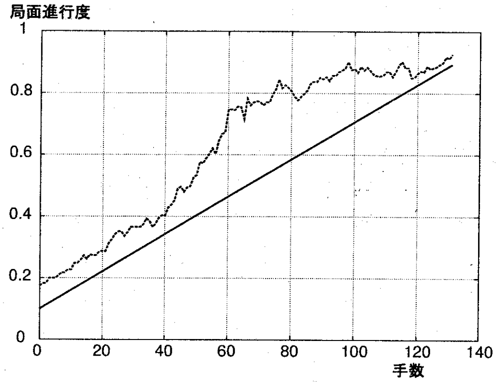

図 4: 誤差の大きな棋譜の例 (単層 SD 法)

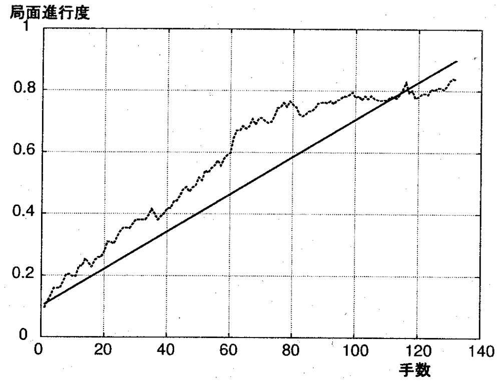

図 5: 誤差の大きな棋譜の例 (3 層 BPQ 法)

して実験してみた．若干の改善は見られるものの，期待していた程の効果が上がっていないことがグラフからわかる．

図 3 は，評価データとして用意した 50 棋譜，おのおのの誤差である．図をみると，棋譜ごとにかなり誤差にばらつきがあることがわかる．将棋では，いつも決まった進行をするわけではなく，例えば，中盤がほとんどなく，序盤からいきなり終盤に入る場合もあるので，ある程度誤差が大きくなる棋譜が存在するのは仕方がないことといえる．また，ほとんどの棋譜において，3 層 BPQ 法による学習の方が良い性能を示しているが，中には単層を用いた方が少ない場合もある．

図 4，図 5 は，図 3 の 16 番目の棋譜を用いて，教師信号と実際の出力を比較したグラフである．共に誤差の大きな棋譜であり，グラフからわかるように，中終盤の誤差が大きい．しかしながら，二つの図を見比べてみると 3 層モデルの方が良い性能を示している．

次に 3 層モデルよりも単層モデルの方が誤差が少ない例として，図 3 の 45 番目の棋譜を用いて，教師信号と実際の出力を比較してみた (図 6，図 7)．グラフを見ると 3 層モデルの方は中終盤において定量化に失敗しているが，単層モデルの方は中終盤で誤差が大きくなっているも

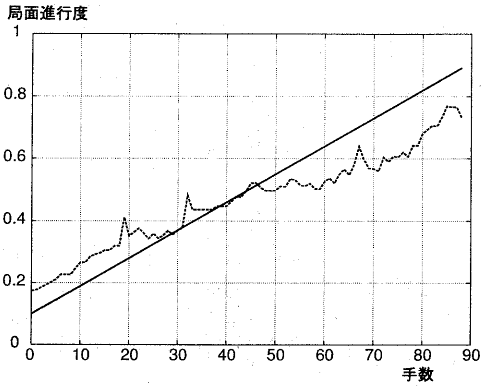

図 6: 単層の方が誤差が少ない棋譜の例 (単層 SD 法)

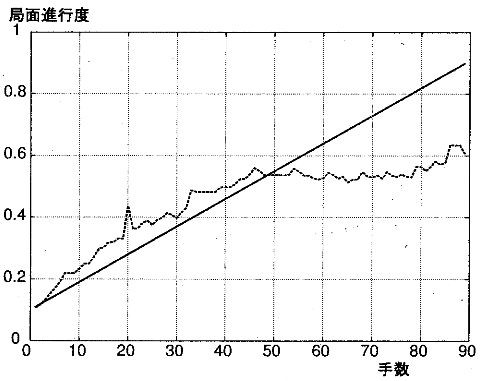

図 7: 単層の方が誤差が少ない棋譜の例 (3 層 BPQ 法)

ののの，3 層モデルに比べれば，教師信号に近い出力が得られている．図 8 は，上記の棋譜で進行度を出力した棋譜の投了図である．この棋譜は，後手の指し切りとなっており，玉を詰めるにはまだ手数がかかりそうである．このように指し切りの棋譜においては，終盤の定量化が難しいと思われる．

図 9 は，学習と評価に要した計算時間を比較したものである．グラフをみると，単層 SD 法に比べ，3 層 BPQ 法の方が短時間で終了していることがわかる．これは学習が収束するまでの繰り返し回数に原因がある．図 11，図 12 は SD 法と BPQ 法の誤差曲線である．2 次法であ

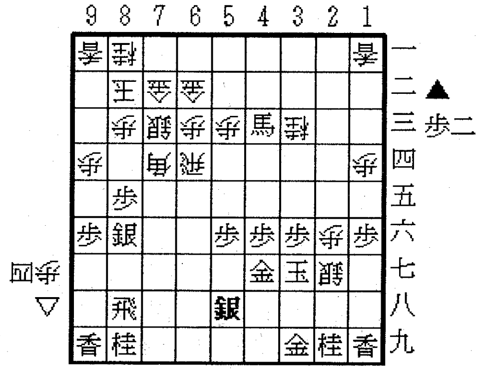

図 8: 単層の方が誤差が少ない局面例 (投了図)

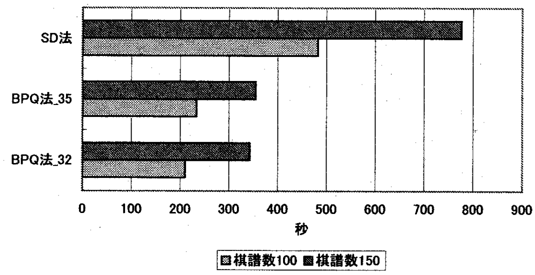

図 9: 計算時間 (Pentium Ⅲ 1 GHz)

る BPQ 法は 1 次法の SD 法に比べて 200 倍以上速く収束している．SD 法が遅い大きな理由のひとつはバッチ型にあると思われる．

4.2 入力空間の違いによる性能比較

図 13 は，281 ビットで原情報を符号化したものと，特徴要素群を用いる方法において一棋譜当たりの平均誤差を比較したグラフである．学習データと評価データをそれぞれ 5 種類用意し，1 つの学習データに対しては初期値を変えて 5 回ランして評価が最良のものを載せた．結果を見ると，原情報方式は，特徴量方式に比べて，誤差が大きくなっている．これは局面を的確に表現する特徴量が利用できないためで，ある程度予想された結果であるが，その差は大きくなく意外に良い性能であった．

図 10 は，評価データ 50 棋譜おのおのの誤差

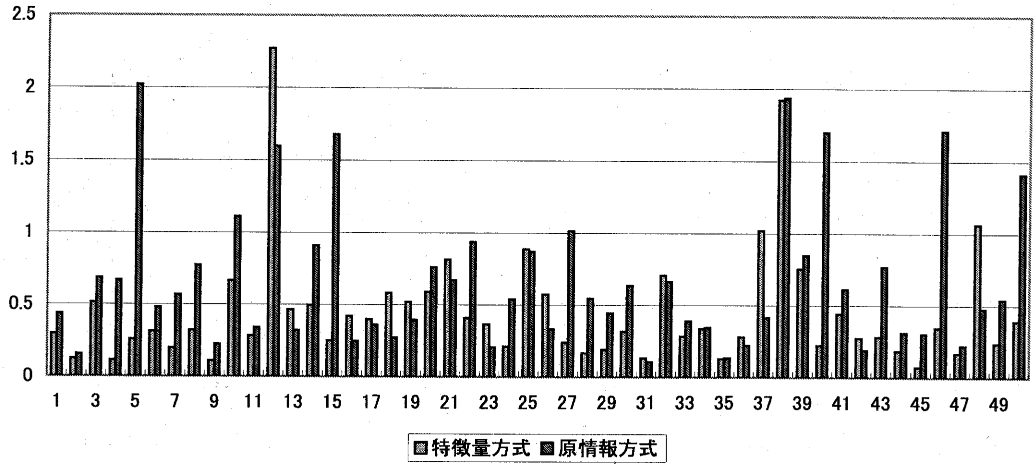

図 10: 各棋譜の誤差

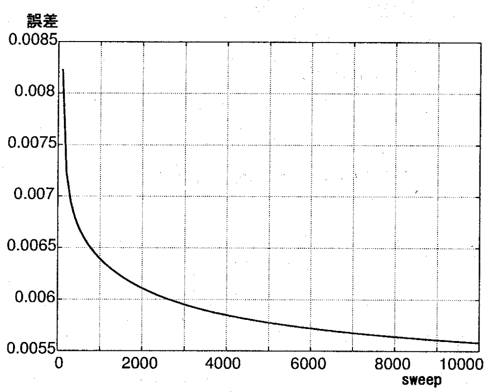

図 11: SD 法を用いた場合の誤差曲線

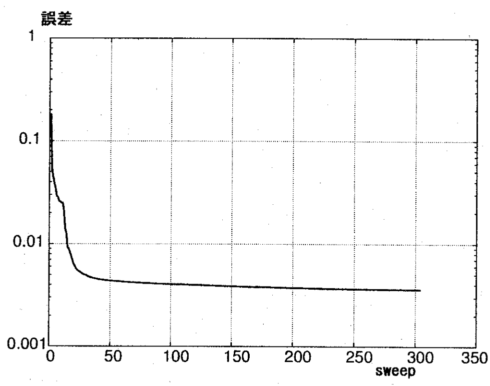

図 12: BPQ 法を用いた場合の誤差曲線

である．特徴量方式に比べ，原情報方式の方が著しく悪い棋譜がいくつか見られ，この影響で図 13 の平均誤差が大きくなっていることがわかる．しかしながら，同程度の性能を示している棋譜や原情報方式の方が勝っている棋譜もある．

図 14，図 15 は，図 10 の 5 番目の棋譜において教師信号と実際の出力を比較してみたグラフである．特長量方式ではかなりの精度で局面進行度の定量化に成功しているが，原情報方式では終盤で大きく乱れている．図 16 は上記で進行度を出力した棋譜の終局間近の局面図である．局面を見ると，お互い相手陣に駒が全くない状態である．この場合，盤上の駒の位置だけで進行度を判定することは難しいと考えられ，原情報方式で終盤の誤差が大きくなるのはある程度仕方がないことかもしれない．

図 17 は原情報方式においてうまく定量化できた例である．グラフをみると非常に精度がよいことがわかる．このように棋譜の中には，原情報方式でも十分な定量化が可能なものがあることがわかった．

しかし，原情報方式の最大の問題点はその計算時間にある．特徴量方式の場合は，先に述べたように約 4 分〜 5 分で収束するのに対し，原情報方式の場合は，約 6 時間もかかった．

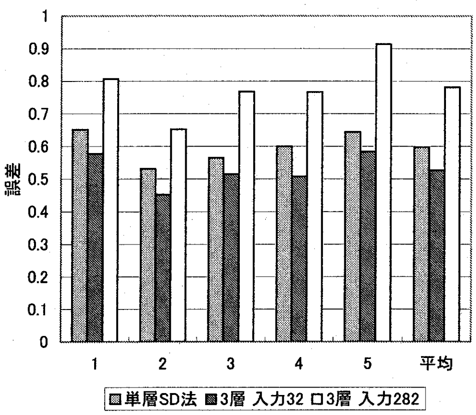

図 13: 一棋譜当たりの平均誤差

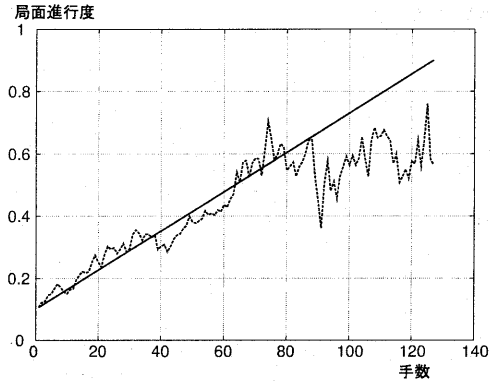

図 15: 原情報方式で定量化が難しい棋譜の例 (原情報方式)

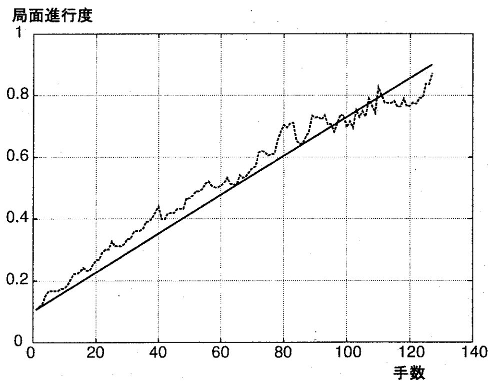

図 14: 原情報方式で定量化が難しい棋譜の例 (特徴量方式)

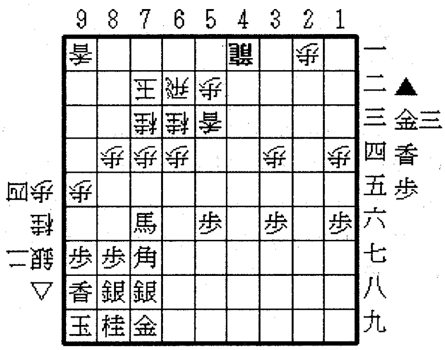

図 16: 原情報方式において終盤と判定するのが難しい局面例 (終局 11 手前)

5 まとめ

学習による局面進行度の定量化について述べた．実験の結果，従来の単層パーセプトロンによる学習よりも 3 層パーセプトロンによる学習の方が正確な定量化が可能であることがわかった．また，局面から入力要素を抽出するときに，人間の専門知識を用いず，駒の位置や持ち駒の情報だけを用いた定量化を試みた．予想以上に誤差の少ない結果となったが，駒の位置だけでは局面状況を判定するのが難しい場合も存在するので，従来法に比べると平均して 25% ほど誤差が増大した．

今後の課題としては，人間の専門知識に基づく特徴量の自動抽出が挙げられる．

参考文献

[1] 小谷 善行ら 著，“コンピュータ将棋”，サイエンス社，1990.

[2] 松原 仁 編著，“コンピュータ将棋の進歩 2”，共立出版，1998.

[3] 松原 仁 編著，“コンピュータ将棋の進歩 3”，共立出版，2000.

[4] 加藤 俊明，鈴木 豪，小谷 善行，堤 正義，“ニューラルネットワークによる将棋の局面進

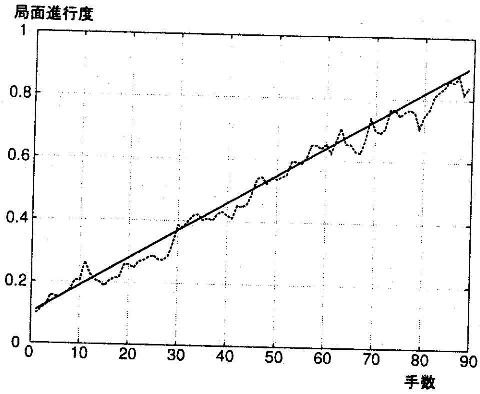

図 17: 原情報方式において定量化がうまくいった例 (原情報方式)

行状況の判断", ゲームプログラミングワークショップ'99.

[5] Kazumi Saito, Ryohei Nakano,"Partial BFGS update and efficient step-length calculation for three-layer neural networks", Neural Computation,vol9,No.1,pp.123-141,1997.