

コンピュータブリッジにおけるディフェンス

小田和 友仁 村上 隆志 上原 貴夫
東京工科大学

概要

コンピュータにコントラクトブリッジをプレイさせる場合の難しさは、ディフェンスにおけるパートナーシップにあるといわれる。本論文では、上級者のディフェンスの技法をコンピュータブリッジに実装する方法について検討し、実験により評価する。

Winning Defense at Computer Bridge

Tomohito Otawa Takashi Murakami Takao Uehara
Tokyo University of Technology

Abstract

The partnership by defenders is the most difficult task for computer bridges. This paper describes how to implement the winning defense of top level player. The proposed implementation method is estimated based on experimental results.

1 はじめに

コントラクトブリッジのプレイは、コントラクトを達成しようとするディクレアラー側と、これを阻止しようとするディフェンダー側に分けて考えられる。ディクレアラーのパートナーはダミーとよばれ、最初のカードが出された直後にハンドをテーブル上に開き、以後のプレイはディクレアラーが代行する。従って、ディクレアラーは自分達のハンドを見て、コントラクトを達成するためのプランをたてることができる。ディフェンダー側は、ダミーのハンドは見えるが、ディクレアラーはもちろん、パートナーのハンドも見えないのでプレイはより難しくなる。よいディフェンスをするためには、ディフェンダー間のパートナーシップが重要であるが、コンピュータブリッジでパートナーシップを実現するのは容易ではない。

よいパートナーシップを実現するための基本的な手段として、シグナルが知られている。シグナルにより、自分のハンドに関する情報を伝えることができる。著者等は先に3種類のシグナルの実装と実験を行い、よい結果を得ている。第4章でその概要を紹介する。

上級者のディフェンスの技法は様々あり、その分類も容易ではない。本論文では桜井恒夫著「ディフェンスのシグナルとカードプレイ[1]」の分類と例題に従い、各技法をコンピュータブリッジに実装する

方法について検討し、実験により評価した。

2 コンピュータブリッジ

本研究で実験に使うコンピュータブリッジは、オークション部とプレイ部にわけられる。オークション部では自分のハンドとビッドの経過を観察し、ビディングシステムに基づきビッドする。ビディングシステムは制約論理プログラミングにより知識として記述した[2]。これによりビッドの経過から各プレイヤーのハンドに関する制約条件を得られ、この情報はプレイ部へと引き継がれる。

プレイ部での最善手の決定にはモンテカルロ法の原理を利用する。まずオークションとプレイの経過を観察し、制約条件にまとめる。そしてすべての制約条件を満たす仮想のハンドを多数生成し、それぞれにおいて探索する。最善手は、すべての仮想のハンドの探索結果から総合的に決定する。

2.1 モンテカルロ法によるカード選択

見えていないカードの可能な分布(世界)を多数生成し、各世界でダブルダミーブリッジのミニマックス値を求め、選ばれた候補の中から全体として良さそうな行動を選ぶ方法が何人かの研究者により提案された。

Ginsberg[3]は、行動の候補の集合を M としたとき、つぎのようにしてひとつの行動(つぎに出すカ

ード) を決定している。

- 1) それまでのビッドおよびプレイと矛盾しないようにカードをくばり、ディールの集合 D を作る。
- 2) 各ディール $d \in D$ ごとに、各行動 $m \in M$ を選んだらどのような結果になるかダブルダミーで評価してスコア $s(m, d)$ を計算する。
- 3) $\sum_d s(m, d)$ が最大となるような行動 m を選ぶ。

このモデルでは、つぎに出すべき1枚のカードを除き、各プレイヤーが (ダブルダミーで) ベストプレイをすることになる。このために、いくつかのタイプの誤ったプレイをしてしまう可能性があることを、Frank等が指摘している[4]。

このアルゴリズムでは、ディフェンスにおいて、パートナーに情報をえるためのプレイを選択することは原理的に不可能である。パートナーに情報を伝えるためのプレイも上記アルゴリズムで自律的に選択されることはないが、強制的にシグナルを出すような実装は容易にできる。

2. 2 パートナーのモデル

ディフェンスにおけるパートナーシップの基本のひとつは、パートナーがミスをしたように誘導することである。ところが、前記のカード選択アルゴリズムは、自分のハンドがパートナーに見えており、パートナーはベストプレイをすると仮定しているので、パートナーがミスをしたように配慮したカード選択は不可能である。

完全情報ゲームでは、通常、敵がベストプレイをし、ミスをするのではないと仮定する。しかし、飯田等は、様々な敵のモデルを検討し、敵の戦略に応じたプレイの選択法について研究している[5][6]。不完全情報ゲームであるブリッジにおける同様なモデルは、著者等が提案した[7][8]。このモデルの特徴は敵のモデルと同時にパートナーのモデルを考えていることである。ディフェンダの一人である自分をA、パートナーをB、敵であるディクレアラーをCとすると、自分のハンドがB、Cからどのように見えるか推論して、それに基づいてB、Cがプレイをすると仮定し、ゲーム木探索を行なう。この探索により、パートナーBがミスをする場合も発見することができ、パートナーが間違えないようなディフェンス方法を選択できる。このモデルに基づく (abcアルゴリズムとよぶ) ゲーム木探索法を次節に示す。

2. 3 abcアルゴリズム

プレイヤーAがとりうる行動の候補の集合を M としたとき、つぎのようにして一つの行動 (つぎに出すカード) を決定する。

[ステップ 1] それまでのビッドおよびプレイと矛

盾しないようにカードをくばり、プレイヤーAから見たディールの集合 $D(A)$ を作る。

[ステップ 2] 各ディール $d \in D(A)$ ごとに、つぎの手順で、各行動 $m \in M$ を選んだらどのような結果になるか評価してスコア $s(m, d)$ を計算する。

[ステップ 2-1] 各ディール $d_a \in D(A)$ ごとに、それまでのビッドおよび m を含むプレイと矛盾しないようにカードをくばり、他の2人のプレイヤーB、Cの立場から見たディールの集合 $D(B, m, d)$ 、 $D(C, m, d)$ を作る。

[ステップ 2-2] 3つのディール $d_a \in D(A)$ 、 $d_b \in D(B, m, d)$ 、 $d_c \in D(C, m, d)$ の組ごとに、行動 $m \in M$ を選んだ結果として得られるスコア $s(m, [d_a, d_b, d_c])$ をダブルダミーで計算する。ただし、プレイヤーA、B、Cはそれぞれディール d_a 、 d_b 、 d_c を仮定して行動を選択するものとする。

[ステップ 2-3] 各ディール $d_a \in D(A)$ について $[d_a, d_b, d_c]$ の全て (n 個) に対する平均値 $\sum s(m, [d_a, d_b, d_c]) / n$ を求め、これを $s(m, d_a)$ とする。

[ステップ 3] $\sum_d s(m, d)$ が最大となるような行動 m を選ぶ。

2. 4 ヒューリスティックな選択

ゲーム木探索による (2. 1節あるいは2. 3節の) カード選択アルゴリズムにおいて、評価値が同じ複数のカードが選択された場合、その内のどれを実際に出すかは、その後のプレイに微妙に影響する。ディフェンスにおいては、同じスーツ内では最低のカードを出すことにしている。その方が一般にパートナーに与える情報が多いと考えたからである。これは、ダックしてもしなくても同じ評価値である場合、ダックする方を選ぶ結果にもなる。ただし、このヒューリスティックな選択は、他のプレイヤーのハンドに関する推論が正しくないと、ローカードで取られるリスクがある。

3 シグナル

パートナーとのコミュニケーションの手段である下記のシグナルをコンピュータブリッジに実装する方法について述べる。

[シグナル]

- (1) アティチュードシグナル
- (2) カウントシグナル
- (3) スートプリファランスシグナル

3. 1 シグナルの実装

コンピュータブリッジに制約条件としてシグナルの知識を記述し、シグナルを考慮に入れた推論を可能にした。シグナルを考慮に入れた推論は以下の3つの場合がある。

1. シグナルとして大きいカードを出した場合.
2. シグナルとして小さいカードを出した場合.
3. シグナルを出せなかった場合.

本研究ではこれらの場合をすべて制約条件とし、他の制約条件と論理積をとることで整合性をとる。

またコンピュータブリッジが出すカードを決定する部分において、最善手がシグナルを出せるカードだった場合は、シグナルの約束事にしたがってカードを出すように変更した。

3種のシグナルの使い分けに関しては、桜井氏の著書を教科書として学び、現状ではコンピュータへの実装が難しい部分を以下のように変更して実装した。以下において、AT, CS, SPはそれぞれアティチュードシグナル、カウントシグナル、スートプリファランスシグナルをあらわす。

1. ATとCSの使い分けに関する高度の判断は無理なので、原則として、AあるいはQのリードはATの要求、KのリードはCSの要求とし、機械的に、短いスートからはAまたはQ、長いスートからはKを出すものとした。また、それ以外のリードに対しては、原則としてCS（後にATに変更）を出すものとした。
2. AT, SPにおけるスートの好ましさの判断を、先読みにより決定することは計算時間の関係で難しい。そこで機械的にA, K, Qおよびボイドをチェックして選択するものとした。
3. AT, SPを受信した側は、必ずしも要求されたスートを出さないものとした。要求されたスートにA, K, Qおよびボイドをもっている可能性を制約条件に加え、2.1で述べたアルゴリズムにより出すカードを決定する。

3. 2 シグナルを利用したディフェンスの実験

桜井恒夫著「ディフェンスのシグナルとカードプレイ」からとった約100の問題について実験をおこない、90%以上の正解率を得た[9]。その内容を分析し、コンピュータ間でのシグナルと人間のペア間でのシグナルの違いについて考察した。

また、比較のため、本研究で実験に使用した例題をJackやGIBなどの商用コンピュータブリッジに与えた。JackやGIBではカウントシグナルとアティチュードシグナルが定義されているが、スートプリファランスシグナルについては明記されていない。比較実験の結果、カウントシグナルとアティチュードシグナルに関する例題は、解説書と同じようなプレイができた。しかしスートプリファランスシグナルを利用すべき例題の正解率は4割程度だった。我々のコンピュータブリッジのディフェンスは、桜井氏の教科書から学んだ結果、この点で比較的強化された。

3. 3 コンピュータと人間のコミュニケーション

アティチュードシグナル、スートプリファランスシグナルは形式的な要件のチェックでおこなうため、先読みの結果で判断する人間からみると正しくないカードを出す場合が多かった。しかしコンピュータはパートナーのシグナルを参考にしながらも、自分の判断で出すカードを選択するので、結果は正しいディフェンスになる可能性がある。これはコンピュータ同士がペアならば、結果的に問題がない場合が多いが、人間とペアを組むときには問題がある。人間とペアを組む場合は機械的にパートナーの選択に従う方式の方がよいかもしれない。商用のコンピュータブリッジは、様々なレベルの人間とのコミュニケーションを考慮して、複雑な約束事は組み込めないのが実情かもしれない。

3. 4 シグナルを出すべきか出さざるべきか

前記の実験の対象とした例題は、シグナルが有効である場合のみである。シグナルは敵、味方同時に情報を与えるので、その結果、敵の利益になる場合もある。著者等はabcアルゴリズムを用い、シグナルを出すべきか出さざるべきか判断できることも示した[7]。

4 各種ディフェンス技法

桜井恒夫著「ディフェンスのシグナルとカードプレイ」にある下記のディフェンス技法について、現在のコンピュータブリッジで実行可能か考察する。これらの技法の基本的な前提となるシグナルについては、3章で述べたように実装を済ませた上で実験した。

[ディフェンスの技法]

「ダックする、ダックしない、ダックさせない」

「相手方のエントリーをなくす、味方のエントリーを作る」

「味方のトランプトリックを増やす、相手方のトランプトリックを減らす」

「2, 3, 4, 5枚カードの注意すべきディフェンス」

「リードを取る」

「パートナーの絵札を知る方法、自分の絵札を知らせる方法」

「コントロールを残す」

「スローイン、スクイズを避ける」

「パートナーを迷わせない」

4. 1 実装上の要件

上記の技法をコンピュータブリッジで実行する上で必要とされる機能はつぎのようなものがある。

- (1) 他のプレイヤーのハンドを正確に推論する

- (2) リアルタイムで、深いゲーム木探索をする
- (3) リアルタイムで多くのサンプル (ディール) の検索を行なう
- (4) 他のプレイヤーの立場から推論を行なう
- (5) 情報を獲得するためのプレイの価値を認識する

(1) の推論については、我々は制約論理プログラムによる実装でよい結果を得ている。また、第3章で述べたシグナルの実装により、より正確な推論が可能になった。

(2) と (3) に関しては、独自の並列処理技術を用い、64 台の PC で 40 倍から 50 倍の高速化に成功した[10]。ちなみに、7 トリック (28 枚) の先読みが必要な例題もあり、リアルタイムのプレイには並列処理を必要とする。

(4) に関しては、我々が提案した abc アルゴリズムをもちいれば、問題を解くことはできる。リアルタイムのプレイには並列処理が必要であり、(2) (3) と両立させるためには、さらなる研究が必要である。

(5) に関しては、現在、どのコンピュータブリッジも解決していない。今回の実験では、ヒューリスティックな方法で一部解決したが、より本質的な解決が望まれる。

4. 2 実験に基づく考察

先に述べた (1) から (9) に分類された約140題の例題について実験を行い、実装上の難易度について考察した。

(1) 「ダックする、ダックしない、ダックさせない」

正確な推論と深い探索でほぼ成功した。

[例題 71] H8-HJ となったとき、H7 の AT を出してダックした。

ただし、つぎのタイプは注意を要する。

(1-1) 「パートナーのディスカードによるシグナルを見るためにダックする」

情報を獲得するためのダックは実装できていない。ただし、「同じ評価ならば弱いカードを選択する」というヒューリスティックにより結果的にダックする場面がある。つぎの例ではダックした。

[例題 81] CJ-CQ-C2-C8, HJ-H6-H2-H5(ダック), H10のときにHAでとってパートナーのディスカードを見る。

[例題 71]

♠ Q, 6, 4
 ♥ J, 4, 3
 ♦ A, K, 10, 8
 ♣ Q, J, 5

	N	
W		E
	S	

♠ 10, 9, 8, 7, 5
 ♥ 8, 2
 ♦ 7, 6, 4
 ♣ K, 8, 2

♠ J, 3, 2
 ♥ A, K, 7, 6,
 ♦ Q, 3, 2
 ♣ 7, 3

♠ A, K
 ♥ Q, 10, 9
 ♦ J, 9, 5
 ♣ A, 10, 9, 6, 4

Dealer: North
Bidding

North	East	South	West
1D	1H	3NT	pass
pass	pass	-	-

[例題 81]

♠ Q, J, 8
 ♥ J, 10, 8, 4
 ♦ Q, 9, 7
 ♣ Q, 6, 4

	N	
W		E
	S	

♠ A, 9, 3, 2
 ♥ 5
 ♦ 8, 5
 ♣ J, 10, 9, 7, 5, 3

♠ K, 10, 7, 6
 ♥ A, 9, 6
 ♦ J, 10, 6, 4
 ♣ 2

♠ 5, 4
 ♥ K, Q, 7, 3, 2
 ♦ A, K, 2
 ♣ A, K, 8

Dealer: South
Bidding

North	East	South	West
-	-	1H	pass
2H	pass	4H	pass
pass	pass	-	-

(1-2)「フィネスが利いたふりをするためにダックする」

ディクレアラの立場からの推論を行い、ミスプレイの可能性を発見するabcアルゴリズムを用いれば可能である。

[例題 86] CA-S2-C9-C3, SJ-S3-S7-S6 (ダック), S9-S4-SQ-SKでCを2つとる。

♠ J, 9, 2
 ♥ K, 5, 2
 ♦ A, K, Q, 10, 8, 6, 2
 ♣

♠ K, 6
 ♥ 10, 8, 6, 4, 3
 ♦ J, 4
 ♣ A, K, 8, 6

	N		
W			E
		S	

♠ 5, 4, 3
 ♥ 9, 7
 ♦ 9, 5
 ♣ Q, 10, 9, 7

♠ A, Q, 10, 8, 7
 ♥ A, Q, J
 ♦ 7, 3
 ♣ J, 4, 3

Dealer: South
Bidding

North	East	South	West
-	-	1S	pass
2D	pass	2S	pass
4NT	pass	5S	pass
6S	pass	pass	pass

(1-3)「ゲスをさせるためにダックする」

abcアルゴリズムを用いれば可能である。ただし、前記のヒューリスティックによっても結果的にダックする。

[例題89] (前略) SouthからD4がでたときにD6を出してDKを出すかDJを出すかゲスさせる。

なお、ダックしなければならない時とダックしてはならない時を判断できていることは、つぎの例でダックしなかったことからわかる。

[例題 92] HQ-H3-H2-HA, C3-CA (ダックしない) -C4-C2. Cを1回ダックすると、今度はDにシフトされて3NTが作られてしまう。

(2)「相手方のエントリーをなくす、味方のエントリーを作る」

正確な推論と深い探索ではほぼ実行できる。

[例題 89]

♠ 8, 7, 2
 ♥ K, Q, 10, 5
 ♦ 4
 ♣ A, K, J, 8, 3

♠ K, Q, 10, 6, 4
 ♥ 7, 6
 ♦ Q, 9, 8, 2
 ♣ 5, 2

	N		
W			E
		S	

♠ J, 9, 3
 ♥ 9, 4, 2
 ♦ A, 7, 6, 3
 ♣ 10, 9, 4

♠ A, 5
 ♥ A, J, 8, 3
 ♦ K, J, 10, 5
 ♣ Q, 7, 6

Dealer: South
Bidding

North	East	South	West
-	-	1NT	pass
2C	pass	2H	pass
4C	pass	4S	pass
4NT	pass	5H	pass
6H	pass	pass	pass

[例題 92]

♠ A, 5, 2
 ♥ 7, 3
 ♦ Q, 7, 5
 ♣ K, J, 10, 9, 4

♠ Q, 10, 3
 ♥ Q, J, 10, 9, 5
 ♦ A, 6, 2
 ♣ A, 5

	N		
W			E
		S	

♠ J, 9, 8, 7,
 ♥ 8, 4, 2
 ♦ 8, 3
 ♣ 8, 6, 2

♠ K, 6
 ♥ A, K, 6
 ♦ K, J, 10, 9, 4
 ♣ Q, 7, 3

Dealer: South
Bidding

North	East	South	West
-	-	1NT	pass
3NT	pass	pass	pass

(3) 「味方のトランプトリックを増やす, 相手方のトランプトリックを減らす」

正確な推論と深い探索ではほぼ実行できる. ただし「クー・アンパッサンにする」例における継続要求のシグナルを出すことは abc アルゴリズムでないと困難である.

[例題 120] S9-S10-SJ-S3, SK-S6-S4-S5, SA-S7-C3-S8. スペードを継続してもらいたいことをシグナルで伝える.

♠ Q, 10, 8, 5
 ♥ A, Q
 ♦ A, K, Q, 4
 ♣ 7, 6, 5

	N		
W			E
		S	

♠ 9, 4
 ♥ J, 7, 4
 ♦ J, 9, 7, 2
 ♣ K, J, 9, 3

♠ A, K, J, 2
 ♥ 3, 2
 ♦ 10, 8, 6, 5
 ♣ 10, 8, 2

♠ 7, 6, 3
 ♥ K, 10, 9, 8, 6, 5
 ♦ 3
 ♣ A, Q, 4

Dealer: North
Bidding

North	East	South	West
1NT	pass	4H	pass
pass	pass	-	-

(4) 「2, 3, 4, 5 枚カードの注意すべきディフェンス」

十分なハンドのサンプルを生成し, 注意すべきカードの配置が含まれるようにする必要がある. 結果的に失敗した例題が多い.

(5) 「リードを取る」

パートナーのシグナルを見たい (情報を獲得したい) ために「オーバーテイクして, パートナーに向けてリードする」ことは, abc アルゴリズムでも不可能である.

[例題 161] HK-H3 の時 HA で取って CK を出し, C のカウントシグナルにより情報を得る.

「オーバーテイクして自分のスートをリードする」理由が, パートナーの立場からは自分のスートがわからないことである場合には, abc アルゴリズムでないと困難である.

[例題 165] SQ-S4 の時 SK で取って HK を出すと 1 ダウンする.

[例題 161]

♠ 10, 5, 4, 2
 ♥ 9, 3
 ♦ A, K, 10, 8, 7
 ♣ Q, 5

	N		
W			E
		S	

♠ 9, 3
 ♥ A, 5, 4, 2
 ♦ 6, 3
 ♣ K, 8, 6, 4

♠ A, K, Q, J, 8, 7, 6
 ♥ 6
 ♦ Q, J, 4
 ♣ J, 7

Dealer: North
Bidding

North	East	South	West
-	-	1S	2H
3S	4H	4S	5H
pass	pass	5S	pass
pass	pass	-	-

[例題 165]

♠ 8, 7, 6, 4
 ♥ 10, 7, 6
 ♦ Q, J, 7, 3
 ♣ J, 8

	N		
W			E
		S	

♠ Q, J, 10
 ♥ 8, 5, 3
 ♦ 10, 8, 6, 4
 ♣ K, 9, 5

♠ A, K, 2
 ♥ K, Q, J, 9
 ♦ 9, 5
 ♣ 7, 4, 3, 2

♠ 9, 5, 3
 ♥ A, 4, 2
 ♦ A, K, 2
 ♣ A, Q, 10, 6

Dealer: North
Bidding

North	East	South	West
-	-	1NT	pass
pass	pass	-	-

(6) 「パートナーの絵札を知る方法, 自分の絵札を知らせる方法」

「自分の絵札を知らせる方法」は, 正確は推論と abc アルゴリズムによって実現できる. また, 知らせる意図はなくても, 「同じ評価ならば弱いカードを選択する」というヒューリスティックにより結果的に実行する場合がある.

[例題 166] S3-S2 で, W の SA のアンダーリードは絶対がないから SJ を出す. このプレイで SQ をだれが持っているかの情報が得られる. この情報を無視すれば, SK でも SJ でも同じになるので, 上級者でなければ SK を出す.

♠ 9, 5, 2
 ♥ 10, 9, 3
 ♦ 9, 7, 5
 ♣ K, Q, 10, 8

	N	
W		E
	S	

♠ 10, 8, 7, 3
 ♥ A, Q, 5
 ♦ 3, 2
 ♣ J, 9, 6, 3

♠ K, J, 6
 ♥ J, 8, 6, 4,
 ♦ 6
 ♣ A, 7, 4, 2

♠ A, Q, 4
 ♥ K, 7
 ♦ A, K, Q, J, 10, 8, 4
 ♣ 5

Dealer: North
Bidding

North	East	South	West
Pass	pass	5D	pass
pass	pass	-	-

「パートナーの絵札を知る方法」は, ヒューリスティックにより結果的に実行する場合がある. しかし, 「パートナーのリードしたスートをAまたはKで取った場合に, 他のスートのAまたはKを出して見て, そのスートについてカモンかノンカモンかをみる」方法は, 上記のヒューリスティックは通用しない. しかし「パートナーを迷わせない」意図で abc アルゴリズムが結果的に実行する場合もある.

[例題 170] D2-D4-DA-D7. CAをとってノンカモンなのでDをリターンし1ダウン.

[例題 170]

♠ 7, 5, 3, 2
 ♥ A, 6
 ♦ K, J, 9, 5, 4
 ♣ 7, 4

	N	
W		E
	S	

♠ 9, 8, 2
 ♥ 9, 8, 7, 2
 ♦ 2
 ♣ J, 10, 9, 6, 3

♠ Q, J, 10, 5
 ♥ A, 10, 6, 3
 ♣ A, Q, 8

♠ A, K, Q, J, 10, 6
 ♥ K
 ♦ Q, 8, 7
 ♣ K, 5, 2

Dealer: South
Bidding

North	East	South	West
-	-	1S	pass
2S	3H	4S	5H
5S	pass	pass	pass

(7) 「コントロールを残す」

コントロール残す方法として「ダックをする」「アンダーリードをする」などがある. 後者はダミーの見ていないオープニングリード時は難しいが, 前記のヒューリスティックで結果的に実行される場合もある.

(8) 「スローイン, スクイズを避ける」

前者は, 正確な推論と深い探索ではぼ実行できる. 後者は, 最後で実現するスクイズを防ぐために早い段階で対処するには, きわめて深い探索を必要とするため, リアルタイムのプレイが難しい.

(9) 「パートナーを迷わせない」

abc アルゴリズムの目的とするところである. 通常のアルゴリズムでは結果的に実行することはあるが, 確実性にかける.

[例題 180] DA-D6-D9-D2, DK-D7-D2-D4. ここでつぎの D をパートナーがラフした後で S と C のどちらをリードしたらよいか分からないだろうから先に CA を取ってから D5 をリードする.

[例題 180]

	♠ K, 4											
	♥ K, Q, 9											
	♦ J, 10, 8, 7, 6											
	♣ Q, 3, 2											
♠ J, 9, 8, 7, 5	<table style="width: 100%; height: 100%; border-collapse: collapse;"> <tr><td style="text-align: center;">N</td><td></td><td></td></tr> <tr><td style="text-align: center;">W</td><td></td><td style="text-align: center;">E</td></tr> <tr><td></td><td style="text-align: center;">S</td><td></td></tr> </table>	N			W		E		S		♠ Q, 10, 6, 2	
N												
W		E										
	S											
♥ 6, 5		♥ 3, 2										
♦ A, K, 5		♦ 9, 2										
♣ A, 7, 4		♣ J, 10, 9, 8										
	♠ A, 3											
	♥ A, J, 10, 8, 7, 4											
	♦ Q, 4, 3											
	♣ K, 6											

Dealer: South
Bidding

North	East	South	West
-	-	1H	1S
3H	pass	4H	pass
pass	pass		

5 おわりに

コンピュータブリッジによるディフェンスについて、1冊の教科書をもとに実験を行い、その難しさについて考察した。まだ結果の分析は不十分であるが、広い範囲のプレイテクニックについて検討することができた。今後、個々の問題についてさらに掘り下げていきたい。

本研究での例題の引用を許可して下さった桜井恒夫氏に対して感謝いたします。また、実験に協力してくれた内沼聖君をはじめ研究室の皆さんに感謝します。

参考文献

[1] 桜井 恒夫:”ディフェンスのシグナルとカードプレイ”, エスアイビー・アクセス(1998), 改訂版(2002)

[2] 安藤剛寿, 小林紀之, 上原貴夫:”コンピュータブリッジのビッドにおける協調と競合”, 電子情報通信学会論文誌, Vol.J83-D-I, No.7, pp.759-769, (2000)

[3] Ginsberg, M.L.: “GIB: Imperfect information in a computationally challenging game”, Journal of Artificial Intelligence Research, (2001)

[4] Frank, I.: “Search and planning under incomplete information : A study using Bridge card play”, Springer-Verlag in the Distinguished Dissertations Series,(1998)

[5] Iida, H., Uiewijk, J, W.H.M., van den Herik, H.J. and Herschberg, I.S.:”POTENTIAL APPLICATIONS OF OPPONENT – MODEL SEARCH: Part 1 : The Domain of Applicability”, ICCA Journal, Dec (1993)

[6] Iida, H., Uiewijk, J, W.H.M., van den Herik, H.J. and Herschberg, I.S.:”POTENTIAL APPLICATIONS OF OPPONENT – MODEL SEARCH: Part 2 : Risks and Strategies”, ICCA Journal, Mar (1994)

[7] 小林紀之, 小田和友仁, 上原貴夫:”コンピュータブリッジによるカウントシグナル”, 情報処理学会論文誌, Vol.45, No.6, pp.1704-1714, (2004)

[8] Ottawa, T. and Uehara, T.:”Model of bridge players who make mistakes”, Game Informatics Special Interest Group, IPSJ, (2005)

[9] 村上隆志, 小田和友仁, 上原貴夫:”コンピュータブリッジにおける各種シグナルの実装”, 情報処理学会第 67 回全国大会, 6L-8(2005)

[10] 小田和友仁, 埴敏博, 上原貴夫:” コンピュータブリッジの並列処理”, 情報処理学会論文誌 (投稿中), (2005)