On the Complexity of Trick-Taking Card Games

Édouard Bonnet, Florian Jamain, and Abdallah Saffidine
LAMSADE, Université Paris-Dauphine, Paris, France
{edouard.bonnet,abdallah.saffidine}@dauphine.fr
florian.jamain@lamsade.dauphine.fr

Abstract

Determining the complexity of perfect information trick-taking card games is a long standing open problem. This question is worth addressing not only because of the popularity of these games among human players, e.g., DOUBLE DUMMY BRIDGE, but also because of its practical importance as a building block in state-of-the-art playing engines for CONTRACT BRIDGE, SKAT, HEARTS, and SPADES.

We define a general class of perfect information two-player trick-taking card games dealing with arbitrary numbers of hands, suits, and suit lengths. We investigate the complexity of determining the winner in various fragments of this game class.

Our main result is a proof of PSPACE-completeness for a fragment with bounded number of hands, through a reduction from Generalized Geography. Combining our results with Wästlund's tractability results gives further insight in the complexity landscape of trick-taking card games.

1 Introduction

Determining the complexity class of games is a popular research topic [Hearn, 2006], even more so when the problem has been open for some time and the game is actually of interest to players and researchers. For instance, the game of AMAZONS was proved PSPACE-complete by three different research groups almost simultaneously [Furtak *et al.*, 2005; Hearn, 2006]. In this paper, we investigate the complexity of *trick-taking card games*. The class of trick-taking card games encompasses numerous popular games such as CONTRACT BRIDGE, HEARTS, SKAT, SPADES, TAROT, and WHIST.[1]

The rules of the quintessential trick-taking card game are fairly simple. A set of players is partitioned into teams and arranged around a table. Each player is dealt a given number of cards t called *hand*, each card being identified by a *suit* and a *rank*. The game consists in t tricks in which every player plays a card. The first player to play in a given trick is called lead, and the other players proceed in the order defined by the seating. The single constraint is that players should follow the lead suit if possible. At the end of a trick, whoever put the highest ranked card in the lead suit wins the trick and leads the next trick. When there are no cards remaining, after t tricks, we count the number of tricks each team won to determine the winner.[2]

Assuming that all hands are visible to everybody, is there a strategy for the team of the starting player to ensure winning at least k tricks?

Despite the demonstrated interest of the general population in trick-taking card games and the significant body of artificial intelligence research on various trick-taking card games [Buro *et al.*, 2009; Ginsberg, 2001; Frank and Basin, 1998; Kupferschmid and Helmert, 2006; Luštrek *et al.*, 2003], most of the corresponding complexity problems remain open. This stands in stark contrast with other popular games such as CHESS or GO, the complexity of which was established early [Fraenkel and Lichtenstein, 1981; Lichtenstein and Sipser, 1980; Robson, 1983].

There are indeed very few published hardness results for card games. We only know of a recent paper addressing UNO [Demaine *et al.*, 2010], a card game not belonging to the category of trick-taking card games, and Frank and Basin's result on the best defense model [2001]. They show that given an imperfect information game tree and an integer w, and assuming the opponent has perfect information, determining whether one has a pure strategy winning in at least w worlds is NP-complete.

As for tractability, after a few heuristics were proposed [Kahn *et al.*, 1987], Wästlund's performed an in-depth combinatorial study on fragments of perfect information two-hands WHIST proving that some important fragments of trick-taking card games are polynomial [Wästlund, 2005a; 2005b].

Note that contrary to the hypotheses needed for Frank and Basin's NP-completeness result [2001], this paper assumes perfect information and a compact input, namely the hands and an integer k. There are several reasons for focusing on perfect information. First, it provides a lower bound to the imperfect information case when compact input is assumed. More importantly, perfect information trick-taking card games

[1] A detailed description of these games and many other can be found on `http://www.pagat.com/class/trick.html`.

[2] There are more elaborate point-based variants where tricks might have different values, possibly negative, based on cards comprising them. We focus on the special case where each card has the same positive value.

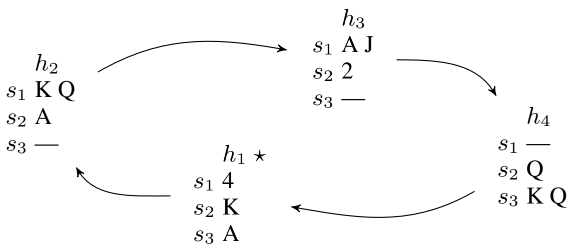

Figure 1: Example of a trick-taking game position with 4 hands, 3 suits, and 1 as lead turn. If team A controls h_1 and h_3 and team B controls h_2 and h_4, then team A can make all three remaining tricks by starting with (s_3, A)

actually do appear in practice, both among the general population in the form of DOUBLE DUMMY BRIDGE problems, but also in research as perfect information Monte Carlo sampling is used as a base component of virtually every state-of-the-art trick-taking game engine [Levy, 1989; Ginsberg, 2001; Sturtevant and White, 2006; Long *et al.*, 2010].

It is rather natural to define fragments of this class of decision problems, for instance, by limiting the number of different suits, the number of hands, or even limiting the number of cards within each suit. We define the lattice of such fragments in Section 2. In Section 3, we show that the general problem is PSPACE-complete and it remains so even if we bound the number of cards per suit. The proof is a rather straightforward reduction from Generalized Geography (GG). Our main result is a more involved reduction from GG to address the fragment with bounded number of hands; it is presented in Section 4. In Section 5, we provide a new tractability result for two hands and four cards per suit. We conclude by providing a graphical summary of the complexity landscape in trick-taking card games and putting forward a few open problems.

2 Definitions and notation

2.1 Trick-taking game

Definition 1. A *card* c is a pair of two integers representing a *suit* (or color) $s(c)$ and a *rank* $r(c)$. A *position* p is defined by a tuple of hands $h = (h_1, \ldots, h_\eta)$, where a *hand* is a set of cards, and a *lead turn* $\tau \in [1, \eta]$. We further assume that all hands in a given position have the same size $\forall i, j \in [1, \eta], |h_i| = |h_j|$ and do not overlap: $i \neq j \Rightarrow h_i \cap h_j = \emptyset$.

An example position with 4 hands and 12 total cards is given in Figure 1. The position is written as a diagram, so for instance, hand h_3 contains 3 cards $\{(s_1,\text{A}), (s_1,\text{J}), (s_2, 2)\}$.

Definition 2. Playing a *trick* consists in selecting one card from each hand starting from the lead: $c_\tau \in h_\tau, c_{\tau+1} \in h_{\tau+1}, \ldots, c_n \in h_\eta, c_1 \in h_1, \ldots, c_{\tau-1} \in h_{\tau-1}$. We also require that *suits are followed*, i.e., each played card has the same suit as the first card played by hand τ or the corresponding hand h_i does not have any card in this suit: $s(c_i) = s(c_\tau) \vee \forall c \in h_i, s(c) \neq s(c_\tau)$.

Definition 3. The *winner of a trick* is the index corresponding to the card with highest rank among those having the required suit. The position resulting from a trick with cards $C = \{c_\tau, \ldots, c_{\tau-1}\}$ played in a position p can be obtained by removing the selected cards from the hands and setting the new lead to the winner of the trick.

In the example in Figure 1, the lead is to 1. A possible trick would be $(s_3,\text{A}), (s_1,\text{Q}), (s_2, 2), (s_3,\text{Q})$; note that only hand h_4 can follow suit, and that 1 is the winner so remains lead.

Definition 4. A *team mapping* σ is a map from $[1 \ldots \eta]$ to $\{A, B\}$ where η is the number of hands. A (perfect information, plain) *trick-taking game* is pair consisting of a position and a team mapping σ.

For simplicity of notation, team mappings will be written as words over the alphabet $\{A, B\}$. For instance, $1 \mapsto A, 2 \mapsto B, 3 \mapsto A, 4 \mapsto B$ is written $ABAB$.

Definition 5. A trick is won by team A if its winner is mapped to A with σ. The *value* of a game is the maximum number of tricks that team A can win against team B.

The value of the game presented in Figure 1 is 3 as team A can ensure making all remaining tricks with the following strategy known as *squeeze*. Start with (s_3, A) from h_1 and play $(s_2, 2)$ from h_3, then start the second trick in the suit where h_2 elected to play.

2.2 Decision problem and fragments

The most natural decision problem associated to trick-taking games is to compute whether the value of a game is larger or equal to a given value ν. Put another way, is it possible for some team to ensure capturing more than ν tricks? We will see in Section 3 that the general problem is PSPACE-hard, but there are several dimensions along which one can constrain the problem. This should allow to better understand where the complexity comes from.

Team mappings only allow team mappings belonging to a language $\mathcal{L} \subseteq \{A, B\}^*$, typically $\mathcal{L} = \mathcal{L}_i = \{(AB)^i\}$ or $\mathcal{L} = \_ = \{A, B\}^*$.

Number of suits the total number of distinct suits s is bounded by a number $s = S$, or unbounded $s = \_$.

Length of suits the maximal number of ranks over all suits l is bounded by a number $l = L$, or unbounded $l = \_$.

Symmetry for each suit, each hand needs to have the same number of cards pertaining to that suit.

The fragments of problems respecting such constraints are denoted by $\mathcal{B}(\mathcal{L}, s, l)$ when symmetry is not assumed. If symmetry is assumed, then we denote the class by $\mathcal{B}^{\mathcal{M}}(\mathcal{L}, s, l)$. The largest class, that is, the set of all problems without any restriction is $\mathcal{B}(\_, \_, \_)$.

Example 1. The class of double-dummy Bridge problems is exactly $\mathcal{B}(\mathcal{L}_2, 4, 13)$.

Proposition 1. *$\mathcal{B}(\_, \_, \_)$ is in* PSPACE.

Proof. The game ends after a polynomial number of moves. It is possible to perform a minimax search of all possible move sequences using polynomial space to determine the maximal number of tricks team A can achieve. $\square$

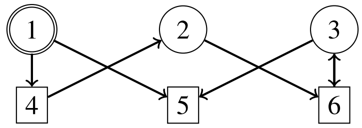

Figure 2: An instance of GG with 1 as the starting vertex.

2.3 Generalized Geography

Generalized Geography (GG) is a zero-sum two-player game over a directed graph with one vertex token. Players take turn moving the token to an adjacent vertex and thereby removing the origin vertex. The player who cannot play anymore loses. Figure 2 presents an instance of GG on a bipartite graph.

Deciding the winner of a GG instance is PSPACE-complete [Schaefer, 1978], and GG was used to prove PSPACE-hardness for numerous games including GO [Lichtenstein and Sipser, 1980], OTHELLO [Iwata and Kasai, 1994], AMAZONS [Furtak *et al.*, 2005], UNO [Demaine *et al.*, 2010]. Lichtenstein and Sipser have shown that GG remains PSPACE-hard even if the graph is assumed to be bipartite [1980].

3 Unbounded number of hands

We present a polynomial reduction from bipartite GG to $\mathcal{B}(\_,\_,\_)$. An instance of GG on a bipartite graph is given by $(G = (V_A \cup V_B, E_{AB} \cup E_{BA}), v_1)$ where $v_1 \in V_A$ denotes the initial location of the token. Let $m = m_{AB} + m_{BA}$ the number of edges and n the number of vertices. We construct an instance of $\mathcal{B}(\_,\_,\_)$ using $m + 2$ suits, and $n + 2$ hands as follows. Each vertex $v \in V_A$ (resp. $\in V_B$) is encoded by a hand h_v owned by team A (resp. B). We add two additional hands, hand h_A for team A and h_B for team B.

Each edge $(s,t) \in E_{AB}$ (resp. E_{BA}) is encoded by a suit $s_{s,t}$ of length 5, for instance {AKQJT}. The cards in suit $s_{s,t}$ are dealt such that hand h_s receives JT, hand h_t receives A, and hand h_A (resp. h_B) receives KQ. We add two additional suits s_A and s_B. s_A (resp. s_B) is dealt such that h_A (resp. h_B) receives $2m_{BA} + 1$ (resp. $2m_{AB} + 1$) cards of highest rank. Each hand $h_v, v \in V_B$ (resp. $v \in V_A$) is filled with low-rank cards in suit s_A (resp. s_B) until a total of $t = 2m + 1$ cards. Low-rank cards are written x when the exact rank matters not.

The game starts in the hand h_{v_1} and team A wins if it makes $2m_{BA} + 1$ tricks, or equivalently, team B wins if it makes $t - 2m_{BA} = 2m_{AB} + 1$ tricks. We denote that instance of $\mathcal{B}(\_,\_,\_)$ by $\phi(G, v_1)$. Figure 3 shows the reduction of an instance of GG to a trick-taking card game in $\mathcal{B}(\_,\_,\_)$.

Lemma 1. *Team A can ensure that as long h_B does not win a trick, the sum of the number of tricks won by team A and the number of cards in suit s_A in hand h_A is greater or equal to $2m_{BA} + 1$. The converse holds for team B.*

Proof. Hand h_A only discards when a suit $s_{v',v}$ is lead with $v \in V_A$, in that case h_v plays an ace and wins the trick. □

Thus, as soon as hand h_A gets the lead, team A achieves its goal. As a result, playing in suit s_A is a losing move for team B. Also, if hand $h_v, v \in V_A$ ever gets the lead again after having played in suit $s_{v,v'}$, then it can give the lead to hand h_A through playing $s_{v,v'}$ once more. Therefore, after hand h_v is in the lead, playing in any suit $s_{v',v}$ effectively becomes a losing move for any $v' \in V_B$.

h_1		h_2		h_3	
s_B	x...	s_B	x...	s_B	x...
$s_{1,4}$	JT	$s_{2,6}$	JT	$s_{3,5}$	JT
$s_{1,5}$	JT	$s_{4,2}$	A	$s_{3,6}$	JT
				$s_{6,3}$	A

h_4		h_5		h_6	
s_A	x...	s_A	x...	s_A	x...
$s_{1,4}$	A	$s_{1,5}$	A	$s_{2,6}$	A
$s_{4,2}$	JT	$s_{3,5}$	A	$s_{3,6}$	A
				$s_{6,3}$	JT

h_A	
s_A	AKQJT9...
$s_{1,4}$	KQ
$s_{1,5}$	KQ
$s_{2,6}$	KQ
$s_{3,5}$	KQ
$s_{3,6}$	KQ

h_B	
s_B	AKQJT9...
$s_{4,2}$	KQ
$s_{6,3}$	KQ

Figure 3: Reduction from Figure 2. h_1 leads the first trick.

Lemma 2. *If the first player has a winning strategy in G, then team A can make $2m_{BA} + 1$ tricks in $\phi(G)$.*

The same result also applies to team B. Therefore team A has a winning strategy in $\phi(G, v_1)$ if and only if the first player has a winning strategy in the instance (G, v_1) of GG. The reduction is thus complete, leading to PSPACE-hardness.

Theorem 1. *$\mathcal{B}(\_,\_,\_)$ is PSPACE-complete.*

Note that the previous reduction can be slightly adapted so that the number of cards in each suit is at most 5. Suits s_A and s_B can indeed be split into multiple equivalent suits of length at most 5 without essentially changing the reduction.

Theorem 2. *$\mathcal{B}(\_,\_,5)$ is PSPACE-complete.*

4 Bounded number of hands

The *trick balance* in an intermediate position is the number of tricks made by team A so far minus the number of tricks made by team B so far.

The basic idea of this reduction is that we have a termination gadget that allows both team to end the game by splitting the remaining tricks evenly. However, using the termination gadget comes with a small cost. So each team tries to achieve a sufficiently high trick balance before terminating the game. The termination gadget involves two suits s_A and s_B and four hands h_3 through h_6 that do not otherwise influence the game.

Besides the termination gadget, we have two hands, one per team, and one suit s_v for each vertex v of the GG instance. A team, say A, can threaten to increase the trick balance in their favour by playing in the attacking gadget of a suit s_v. This can only be defended by having the opponent team, say B, counter-attacking in a suit $s_{v'}$. B can choose $s_{v'}$, but for the defense to be successful, v' needs to be a neighbor of v in GG. After this exchange is performed, team B has priority to attack but can only attack in suit $s_{v'}$'s attacking gadget. The same process goes on until the defending team cannot find an appropriate counter-attacking suit. At that moment, the attacking team manages to increase the trick balance enough to safely terminate the game in their favour. We see that picking the counter-attacking suits emulates a game of GG on G.

Let $G = ((V = V_A \cup V_B, E), v_1)$ be a directed bipartite graph and one of its vertices, $v_1 \in V_A$, be an instance of GG. Let $n = n_A + n_B$ be the number of vertices, and $N(v)$ denote

Hand	Suit	Ranks
h_1	s_A	1
h_2	s_B	1
h_3	s_A	$2t+1 \overset{2}{\text{——}} 3$
h_4	s_A	$2t \overset{2}{\text{——}} 2$
h_5	s_B	$2t+1 \text{——} 2t - {}^{5\omega}/_{2} + 2 \overset{2}{\text{——}} {}^{5\omega}/_{2} + 2$
h_6	s_B	$2t - {}^{5\omega}/_{2} + 1 \overset{2}{\text{——}} {}^{5\omega}/_{2} + 1 \text{——} 2$

Figure 4: The termination gadget. $x \overset{2}{\text{——}} x'$ is a shorthand for $x, x-2, \ldots, x'+2, x'$, and $x \text{——} x'$ is a shorthand for $x, x-1, \ldots, x'+1, x'$. Only the suits involved are displayed.

the neighbors of a vertex v, $\{v', (v, v') \in E\}$. We construct in polynomial time an equivalent instance of $\mathcal{B}(\mathcal{L}_3, \_, \_)$ using 6 hands and $s = n + 2$ suits. In the instance we create, the seating order does not have any influence so we will represent gadgets and positions simply by listing the cards in each hand in a table. In the following, we will set $\omega = 24n$ and use t to represent the total number of tricks to be made.[3] Team A wins if they make strictly more than $k = {}^{t}/_{2} + {}^{3\omega}/_{8}$ tricks.

4.1 Presentation of the gadgets

Unless the gadgets are not symmetric, we only describe one team's version of the gadgets. Assume an arbitrary ordering on the vertices in V_A and in V_B, that is $V_A = \{v_1, v_2, \ldots v_{n_A}\}$ and $V_B = \{v'_1, v'_2, \ldots v'_{n_B}\}$.

The termination gadget. The suit s_A is possessed only by hands h_1 , h_3 and h_4. Hands h_3 and h_4 have $2t$ interleaved cards, and hand h_1 has just enough cards of lowest ranks so that h_1 has a total of t cards after all other gadgets have been taken into account. Thus, hands h_3 and h_4 have only cards in the suit s_A. The suit s_B is owned only by hands h_2, h_5 and h_6. and is displayed similarly for hands h_2, h_5, and h_6, except that h_5 has ${}^{5\omega}/_{2}$ top cards to cash before $2t - {}^{5\omega}/_{2}$ cards are interleaved (see Figure 4).

The following two lemmas allow us to focus on hands h_1 and h_2 in the remaining of this reduction.

Lemma 3. *If h_1 leads and the trick balance is at least ${}^{3\omega}/_{4}$, then team A can ensure winning the game.*

Proof. Team A can play in the suit s_A then the rest of the game will hold between hands h_3 and h_4. Hence team A wins half of the remaining tricks for a total greater than k. $\square$

Lemma 4. *If h_2 leads and the trick balance is at most $-{}^{7\omega}/_{4}$, then team B can ensure winning the game.*

The $e|w$-block. An $e|w$-block in a suit s for team A is a gadget allowing that team to cash (*win*) w tricks by first sacrificing (*establishing*) e tricks in the suit. In other words, hand h_2 has the e top cards, h_1 has the $e + w$ following top cards, and h_2 has w cards of lowest ranks x to avoid any discarding. Figure 5 provides an example of a 3|4-block.

[3] t's exact value can be computed and is polynomial in the input.

Hand	Suit	Ranks	Concise notation
h_1	s	7 — 1	3|4
h_2	s	10 — 8 xxxx	

Figure 5: A 3|4-block in a suit s for team A. Only the hands and the suits involved in this gadget are displayed. h_2 does have cards in suit s, but they are not displayed in the compact notation, as they can be deduced from the cards in h_1.

Hand	Suit	Ranks	Concise notation
h_1	s	27 — 20 15 — 9 5 — 1	3|5 4|3 3|2
h_2	s	30 — 28 19 — 16 8 — 6 xx…	

Figure 6: Concatenation of 3|5-, 4|3- and 3|2-blocks in s for team A. There are $5 + 3 + 2 = 10$ x in suit s in hand h_2.

We can concatenate several $e|w$-blocks for the same team in the same suit. For instance, Figure 6 shows how blocks are concatenated and provides a more concise notation.

Given a vertex $v \in V$, a concatenation of $e|w$ blocks with various values for e allows to encode the index of v in an *attacking* gadget. It also allows to encode which vertex v', v is adjacent to in a *counter-attacking* gadget via their indices. If $v \in V_B$ (resp. V_A), the (counter-)attacking gadgets will be for team A (resp. B) and we say that the corresponding suit is a defensive suit for team B (resp. A). The $e|w$-blocks we need in the following have e an integer in $[1, \ldots, 6]$, and w a fraction of ω.

The counter-attacking and attacking gadgets. Consider two words over $\{0, 1\}$ for each suit s_v with $v \in V_A$. The attacking word for suit s_{v_i} is a such that $a(0) = 1$, $a(n_A + 1) = 0$, and for each $j \neq i, j \in [1, n_A]$, $a(j) = 0$ and $a(i) = 1$. The counter-attacking word for suit s_{v_i} is c such that $c(0) = 1$, $c(n_B + 1) = 1$, and for each $j \in [1, n_B]$, if $v_i \in N(v_j)$ then $c(j) = 1$ else $c(j) = 0$.

The gadgets can be built by looking at adjacent letters in these words. If these letters are 11 or 00, put a $2|\omega$-block. If they are 10, put $3|\omega$-block, and if they are 01, put $1|\omega$. We thus define for each suit s_v, a *counter-attacking* gadget $\mathcal{C}(v)$ and an *attacking* gadget $\mathcal{A}(v)$. The words and gadgets for the GG instance in Figure 2 are given in Figure 7.

Let $v \in V_A$ and $v' \in V_B$. Observe that the sum of the e

Suit	Counter-attacking		Attacking	
	Word c	Gadget $\mathcal{C}$	Word a	Gadget $\mathcal{A}$
s_{v_1}	1 000 1	$3\|\omega$ $2\|\omega$ $2\|\omega$ $1\|\omega$	1 100 0	$2\|\omega$ $3\|\omega$ $2\|\omega$ $2\|\omega$
s_{v_2}	1 100 1	$2\|\omega$ $3\|\omega$ $2\|\omega$ $1\|\omega$	1 010 0	$3\|\omega$ $1\|\omega$ $3\|\omega$ $2\|\omega$
s_{v_3}	1 001 1	$3\|\omega$ $2\|\omega$ $1\|\omega$ $2\|\omega$	1 001 0	$3\|\omega$ $2\|\omega$ $1\|\omega$ $3\|\omega$
s_{v_4}	1 100 1	$2\|\omega$ $3\|\omega$ $2\|\omega$ $1\|\omega$	1 100 0	$2\|\omega$ $3\|\omega$ $2\|\omega$ $2\|\omega$
s_{v_5}	1 101 1	$2\|\omega$ $3\|\omega$ $1\|\omega$ $2\|\omega$	1 010 0	$3\|\omega$ $1\|\omega$ $3\|\omega$ $2\|\omega$
s_{v_6}	1 011 1	$3\|\omega$ $1\|\omega$ $2\|\omega$ $2\|\omega$	1 001 0	$3\|\omega$ $2\|\omega$ $1\|\omega$ $3\|\omega$

Figure 7: Counter-attacking and attacking gadgets in the instance corresponding to Figure 2.

Hand	Suit	Ranks	
h_2	s_{v_1}		$6\|\omega$ $1\|{}^{3\omega}/_{2}$ $\mathcal{A}(v_1)$ $1\|{}^{3\omega}/_{2}$
	s_{v_2}	$4\|{}^{\omega}/_{2}$ $3\|2\omega$ $\mathcal{C}(v_2)$ $2\|2\omega$	$6\|\omega$ $1\|{}^{3\omega}/_{2}$ $\mathcal{A}(v_2)$ $1\|{}^{3\omega}/_{2}$
	s_{v_3}	$4\|{}^{\omega}/_{2}$ $3\|2\omega$ $\mathcal{C}(v_3)$ $2\|2\omega$	$6\|\omega$ $1\|{}^{3\omega}/_{2}$ $\mathcal{A}(v_3)$ $1\|{}^{3\omega}/_{2}$
h_1	s_{v_4}	$5\|\omega$ $1\|{}^{3\omega}/_{2}$ $\mathcal{C}(v_4)$ $2\|\omega$	$4\|\omega$ $3\|{}^{3\omega}/_{2}$ $1\|{}^{3\omega}/_{2}$ $\mathcal{A}(v_4)$ $1\|\omega$
	s_{v_5}	$5\|\omega$ $1\|{}^{3\omega}/_{2}$ $\mathcal{C}(v_5)$ $2\|\omega$	$4\|\omega$ $3\|{}^{3\omega}/_{2}$ $1\|{}^{3\omega}/_{2}$ $\mathcal{A}(v_5)$ $1\|\omega$
	s_{v_6}	$5\|\omega$ $1\|{}^{3\omega}/_{2}$ $\mathcal{C}(v_6)$ $2\|\omega$	$4\|\omega$ $3\|{}^{3\omega}/_{2}$ $1\|{}^{3\omega}/_{2}$ $\mathcal{A}(v_6)$ $1\|\omega$

Figure 8: Combination of attacking and counter-attacking gadgets for the instance corresponding to Figure 2, with v_1 as starting vertex.

parts of the $\mathcal{A}(v)$ gadgets is equal to $2(n_A+1)+1$ and that of the $\mathcal{C}(v')$ gadgets is $2(n_A+1)$. Similarly, the sum of the w parts is $\omega(n_A+1)$. The same holds for $\mathcal{A}(v')$ and $\mathcal{C}(v)$, replacing n_B with n_A.

In the next two lemmas, we assume optimal play from both teams subject to leading from a single suit.

Lemma 5. *Assume the initial trick balance is b, team A starts, team A only leads cards from $\mathcal{A}(v)$, and team B only leads cards from $\mathcal{C}(v')$. The trick balance remains $\geq b-\omega$. If $v' \in N(v)$, it remains $\leq b+1$, else it reaches $b+\omega$.*

Proof. It is optimal to play blocks from the highest to the lowest ranked when in lead, and to take any trick offered when not in lead. Let i be the index of v. Observe that team A needs to $2j+1$ tempi to establish the jth block if $j \neq i$, and $2i$ tempi for the ith block.[4] Team B needs $2j$ tempi to establish the jth block if $v' \in N(v_j)$, and $2j+1$ tempi otherwise. ☐

Lemma 6. *Assume the initial trick balance is b, team A starts, team A only leads cards from $\mathcal{C}(v)$, and team B only leads cards from $\mathcal{A}(v')$. The trick balance remains $\leq b+\omega$. If $v \in N(v')$, it remains $\geq b-1$, else it reaches $b-\omega$.*

When team A attacks in suit s_v and team B does not play in an admissible counter-attacking suit, team A wins ω tricks (Lemma 5) and wins the game by termination (Lemma 3). Conversely, if team A does not play in a neighboring suit when team B attacks, team A loses (Lemma 4, 6). Thus, Lemmas 5 and 6 give the graph structure to the suits.

Combining the attacking and counter-attacking gadgets. We now complete the picture so that the assumptions of Lemmas 5 and 6 are met.

In each suit s_v of hand h_1 (resp. h_2) but the one corresponding to the starting vertex v_1, we start by the counter-attacking gadget $\mathcal{C}(v)$ surrounded by the fixed sequences $5|\omega$ $1|{}^{3\omega}/_{2}$ and $2|\omega$ (resp. $4|{}^{\omega}/_{2}$ $3|2\omega$ and $2|2\omega$) and call it *first part* of the suit. We then add to each suit, including s_{v_1}, the attacking gadget $\mathcal{A}(v)$ surrounded by the fixed sequences $4|\omega$ $3|{}^{3\omega}/_{2}$ $1|{}^{3\omega}/_{2}$ and $1|\omega$ (resp. $6|\omega$ $1|{}^{3\omega}/_{2}$ and $1|{}^{3\omega}/_{2}$) and call it *second part* of the suit. Figure 8 displays the combination resulting from the GG instance in Figure 2; h_1 also has cards in suits s_{v_1}, s_{v_2}, and s_{v_3} but they are not displayed as per the notation of Figure 5.

These fixed starting sequences ensure that once a team leads in suit s_v, they will continue leading only in s_v until the suit is emptied. They also ensure, that while one team chooses the attacking suit first, the opponent actually starts leading in the counter-attacking gadget.

The ending sequences, on the other hand, ensure that after the attacking suit s_v and the first part of the counter-attacking suit $s_{v'}$ have been emptied, the situation corresponds to the reduction from the GG instance with the edges adjacent to v removed and v' as a starting vertex.

Lemma 7. *Leading in defensive suits can never help to win.*

Proof. A team cannot cash more than $6n$ tricks in defensive suits. As the fractions of ω are all discretised by $^1/_4$, and $6n = {}^{\omega}/_{4}$, this cannot participate in winning. ☐

Ensuring the teams simulate GG. Let P be a position resulting from one constructed from a GG instance. Assume there exists a suit s_v (and v the corresponding vertex in the original GG instance), such that for any suit s different from s_A, s_B, and s_v, s is dealt among hands h_1 and h_2 so as to form a first part and a second part in h_1 or in h_2. If s_v forms only a second part in h_2 (resp. h_1), h_2 (resp. h_1) is on the lead, and the trick balance is ${}^{\omega}/_{2}$ (resp. 0), then we say that P is *A-clean* (resp. *B-clean*) and s_v is the current *starting suit*.

Lemma 8. *Let P be a B-clean position with starting suit s_v, and a suit $s_{v'}$ such that $v' \in N(v)$. Assume team A can ensure winning with optimal play from P. If team B only leads from suit $s_{v'}$ and team A only leads from s_v until s_v is empty, then we reach a A-clean position P' with $s_{v'}$ as the starting suit. Moreover, team A can ensure winning from P'.*

Proof. As P is a B-clean position, the trick balance is 0 and the lead is on h_1. Suppose team A plays in a suit s_u with $u \neq v$ and $u \in V_A$, before having established and cashed the ω tricks of the $6|\omega$-block of the color s_v (first block of the second part of that color). After 6 tempi, team B has had time to cash the ${}^{5\omega}/_{2}$ tricks of the two first block $5|\omega$ and $1|{}^{3\omega}/_{2}$ of the color $s_{v'}$, while team A has at most cash the ${}^{\omega}/_{2}$ winners of the first block $4|{}^{\omega}/_{2}$ of s_u. Consequently, the balance trick takes a value smaller than -2ω and team B wins accordingly to Lemma 4. Thus, team A has to use the 6 first tempi to play in s_v. At this point, team B threatens to enter in $\mathcal{C}(v')$ while team A cannot yet enter $\mathcal{A}(v)$. That forces team A to play again in v. Then, team B enters $\mathcal{C}(v')$ immediately followed by team A cashing ${}^{3\omega}/_{2}$ tricks (the trick balance is now 0) and entering $\mathcal{A}(v)$. Note that if team A plays in another suit than v, team B will win by persisting in $\mathcal{C}(v')$ since the trick balance will go be beneath $-{}^{7\omega}/_{4}$. According to Lemma 5, if team A plays solely in $\mathcal{A}(v)$, the trick balance is never losing for any team. In the end, the balance is $-\omega$, the lead is in h_2 and the only remaining cards in h_1 in the color s_v are ${}^{3\omega}/_{2}$ winners. So the trick balance is virtually ${}^{\omega}/_{2}$ and we reach a A-clean position winning for team A. ☐

Lemma 9. *Let P be a A-clean position with starting suit $s_{v'}$. Assume team A can ensure winning with optimal play from P. Then there exists a suit s_v such that $v \in N(v')$ and such that if team B only leads from suit $s_{v'}$ and team A only leads from s_v until $s_{v'}$ is empty, then we reach a B-clean position*

[4]We use the expression *use a tempo* as short for *lose a trick lead.*

P' with s_v as the starting suit. Moreover, team A can ensure winning with optimal play from P'.

Lemma 10. *Let G be a* GG *instance, and consider the corresponding $\mathcal{B}(\mathcal{L}_3, \_, \_)$ instance B. If team A can win in B, then the second player in G does not have a winning strategy.*

Proof. Let σ bet a strategy for the second player in G and let us show that σ is not winning. Assume team B plays according to σ in the B instance. Team A can answer by keeping simulating GG and still ensure winning (Lemma 8 and 9), thereby generating a strategy in GG. Since there are only finitely many suits to be emptied, we will reach a B-clean position with starting suit s_v and without any suit $s_{v'}$ such that $v' \in N(v)$. This shows that the corresponding GG situation is lost and that σ is not a winning strategy. □

Lemma 11. *Let G be a* GG *instance, and consider the corresponding $\mathcal{B}(\mathcal{L}_3, \_, \_)$ instance B. If team B can win in B, then the first player in G does not have a winning strategy.*

Proof. Similar proof with the dual to Lemmas 8 and 9. □

These two propositions lead us to our main result.

Theorem 3. *The $\mathcal{B}(\mathcal{L}_3, \_, \_)$ fragment is* PSPACE-*hard.*

5 Tractability results

In this part, we present a positive result when there is only one hand per team and when the number of cards in any suit is bounded by 4. Only a few positive results on trick taking card games are known. The two following are the main ones.

Theorem 4 (Wästlund [2005a])**.** $\mathcal{B}(\mathcal{L}_1, 1, \_)$ *is in* P.

Theorem 5 (Wästlund [2005b])**.** $\mathcal{B}^{\mathcal{M}}(\mathcal{L}_1, \_, \_)$ *is in* P.

We now focus on the $\mathcal{B}(\mathcal{L}_1, \_, 4)$ fragment. Assume for the sake of simplicity and wlog, that after each trick a normalization process takes place so that suits with 4 and 3 cards have ranks in $\{A, K, Q, J\}$ and $\{A, K, Q\}$ respectively.

A suit s is *controlled* by a player, if they can play in the suit s and make all tricks in s.

We define a strategy σ as follows. When a player is not leading and can follow suit, they play the lowest card ensuring the trick is made if any, or the lowest card overall otherwise. Otherwise suits are categorized and priority are associated to each category for leading as well as for discarding. The categories, their priority and the associated card to lead/discard is displayed in Table 1.

Lemma 12. *The strategy σ is optimal.*

The strategy σ can be computed in polynomial time. We can compute the total number of tricks a team can make by having both players apply σ. This leads to the desired result.

Proposition 2. $\mathcal{B}(\mathcal{L}_1, \_, 4)$ *is in* P.

Table 1: Strategy σ for player P against O in $\mathcal{B}(\mathcal{L}_1, \_, 4)$. P should not lead nor discard in a suit not mentionned.

Suit category	Lead		Discard	
	Card	Priority	Card	Priority
Controlled by P	highest	1	lowest	5[1]
KQJ vs A	any	2	any	3
KQ vs A or KQ vs AJ	any	3	any	4
Controlled by O	any	4	never	
AJ vs KQ	A	5	J	1
AQ vs KJ	A	6	Q	2[2]
KJ vs AQ	J	7	never	

[1] If P has strictly more cards than O in the suit, else never.
[2] If O does not controle all their suits, else never.

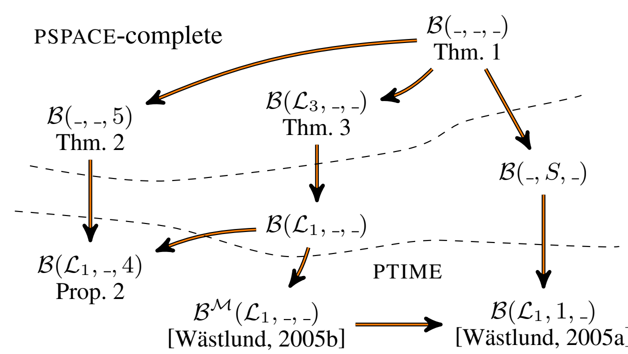

Figure 9: Summary of the hardness and tractability results known for the fragments of $\mathcal{B}(\mathcal{L}, s, l)$.

6 Conclusions and perspectives

In his thesis, Hearn proposed the following explanation to the standing lack of hardness result for BRIDGE [2006, p122].

> There is no natural geometric structure to exploit in BRIDGE as there is in a typical board game.

Theorem 3 achieves a significant milestone in that respect. The gadgets in the reduction indeed show that it is possible to find a graphical structure within the suits. From all, the attacking and counter-attacking gadgets stand as the central idea, giving an adjacency list structure to suits, by means of a precise race to establishment. Termination gadgets make those races decisive.

Finding a PSPACE-hardness proof necessitating only 2 hands is very appealing. Another interesting problem is to find a hardness proof with a bounded number of suits. These two new open problems along with Wästlund's tractability results, and the results derived in this paper are put in perspective in Figure 9 which displays the complexity landscape for noteworthy fragments of $\mathcal{B}(\mathcal{L}, s, l)$.

Many actual trick-taking card games also feature a *trump suit* and potentially different values for tricks based on which cards were involved. Such a setting can be seen as a direct generalization of ours, but remains bounded. Therefore our PSPACE-completeness results carry over to point-based trick-taking card games involving trumps.

References

[Buro *et al.*, 2009] Michael Buro, Jeffrey R. Long, Timothy Furtak, and Nathan Sturtevant. Improving state evaluation, inference, and search in trick-based card games. In *21st International Joint Conference on Artificial Intelligence (IJCAI)*, 2009.

[Demaine *et al.*, 2010] Erik Demaine, Martin Demaine, Ryuhei Uehara, Takeaki Uno, and Yushi Uno. Uno is hard, even for a single player. In *Fun with Algorithms*, pages 133–144. Springer, 2010.

[Fraenkel and Lichtenstein, 1981] Aviezri S. Fraenkel and David Lichtenstein. Computing a perfect strategy for $n \times n$ Chess requires time exponential in n. *Journal of Combinatorial Theory, Series A*, 31(2):199–214, 1981.

[Frank and Basin, 1998] Ian Frank and David Basin. Search in games with incomplete information: A case study using Bridge card play. *Artificial Intelligence*, 100(1):87–123, 1998.

[Frank and Basin, 2001] Ian Frank and David Basin. A theoretical and empirical investigation of search in imperfect information games. *Theoretical Computer Science*, 252(1):217–256, 2001.

[Furtak *et al.*, 2005] Timothy Furtak, Masashi Kiyomi, Takeaki Uno, and Michael Buro. Generalized Amazons is PSPACE-complete. In *19th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 132–137, 2005.

[Ginsberg, 2001] Matthew L. Ginsberg. GIB: Imperfect information in a computationally challenging game. *Journal of Artificial Intelligence Research (JAIR)*, 14:303–358, 2001.

[Hearn, 2006] Robert A. Hearn. *Games, Puzzles, and Computation*. PhD thesis, Massachusetts Institute of Technology, 2006.

[Iwata and Kasai, 1994] Shigeki Iwata and Takumi Kasai. The Othello game on an $n \times n$ board is PSPACE-complete. *Theoretical Computer Science*, 123(2):329–340, 1994.

[Kahn *et al.*, 1987] J. Kahn, JC Lagarias, and HS Witsenhausen. Single-suit two-person card play. *International Journal of Game Theory*, 16(4):291–320, 1987.

[Kupferschmid and Helmert, 2006] Sebastian Kupferschmid and Malte Helmert. A Skat player based on Monte-Carlo simulation. In H.Jaap van den Herik, Paolo Ciancarini, and H.H.L.M.(Jeroen) Donkers, editors, *Computers and Games*, volume 4630 of *Lecture Notes in Computer Science*, pages 135–147. Springer-Verlag, Berlin Heidelberg, 2006.

[Levy, 1989] David N.L. Levy. The million pound Bridge program. In *Heuristic Programming in Artificial Intelligence: The First Computer Olympiad*, pages 95–103. Ellis Horwood, 1989.

[Lichtenstein and Sipser, 1980] David Lichtenstein and Michael Sipser. Go is polynomial-space hard. *Journal of the ACM (JACM)*, 27(2):393–401, 1980.

[Long *et al.*, 2010] Jeffrey Long, Nathan R. Sturtevant, Michael Buro, and Timothy Furtak. Understanding the success of perfect information Monte Carlo sampling in game tree search. In *24th AAAI Conference on Artificial Intelligence (AAAI)*, pages 134–140, 2010.

[Luštrek *et al.*, 2003] Mitja Luštrek, Matjaž Gams, and Ivan Bratko. A program for playing Tarok. *ICGA Journal*, 23(3):190–197, 2003.

[Robson, 1983] John Michael Robson. The complexity of Go. In *IFIP*, pages 413–417, 1983.

[Schaefer, 1978] Thomas J. Schaefer. On the complexity of some two-person perfect-information games. *Journal of Computer and System Sciences*, 16(2):185–225, 1978.

[Sturtevant and White, 2006] Nathan R. Sturtevant and Adam M. White. Feature construction for reinforcement learning in Hearts. In H.Jaap Herik, Paolo Ciancarini, and H.H.L.M.(Jeroen) Donkers, editors, *Computers and Games*, volume 4630 of *Lecture Notes in Computer Science*, pages 122–134. Springer, Berlin Heidelberg, 2006.

[Wästlund, 2005a] Johan Wästlund. A solution of two-person single-suit Whist. *The Electronic Journal of Combinatorics*, 12(1):R43, 2005.

[Wästlund, 2005b] Johan Wästlund. Two-person symmetric Whist. *The Electronic Journal of Combinatorics*, 12(1):R44, 2005.