Yale University
Department of Computer Science

Multiparty Secret Key Exchange Using a Random Deal of Cards[1]

Michael J. Fischer Rebecca N. Wright

YALEU/DCS/TR-855
June 1991

[1]This research was supported in part by National Science Foundation grant IRI–9015570.

Multiparty Secret Key Exchange Using a Random Deal of Cards*

Michael J. Fischer Rebecca N. Wright

Computer Science Department
Yale University
New Haven, CT 06520–2158

Abstract

We consider the problem of *secret key exchange*. A "team" of players P_1 through P_k wishes to determine an n-bit secret key in the presence of a computationally unlimited eavesdropper, Eve. Following the example of Fischer, Paterson and Rackoff [5], the team players are dealt hands of cards of prespecified sizes from a deck of d distinct cards; any remaining cards are dealt to Eve. We explore how the team can use the information contained in their hands of cards to determine an n-bit key that is secret from Eve, that is, an n bit string which each team player knows exactly but for which Eve's probability of guessing even one bit correctly is the same before and after she hears the communication between the team players. We describe randomized protocols for secret key exchange that work for certain classes of deals, and we present some conditions on the deal for such a protocol to exist.

1 Introduction

An important problem of cryptography is the problem of *multiparty secret key exchange*. This can be viewed as a multiparty protocol between a group of players. At some point, a subset of $k \geq 2$ players P_1 through P_k form a "team". The rest of the players are considered eavesdroppers. The team players carry out randomized algorithms. Each player's random choices are private to that player. All communication is by public broadcast and is overheard by the eavesdroppers. The following scenario demonstrates a situation in which the need for secret key exchange might arise.

A certain government agency handles security of information on a "community of interest" basis. For each project within the agency, a group of people are chosen to work on the project. We call this group a team. Teams form and dissolve as various projects are started and completed. All communication regarding the project is intended to be shared with those on the team, and to be kept secret from those outside the team. However, the security of the various communication channels—the telephone, interoffice mail, electronic mail, and face-to-face communication—is not guaranteed. Hence, each time a team forms, they would like to exchange a secret key, which they can then use as a part of some cryptographic protocol to securely send all further communication regarding the project.

*This research was supported in part by National Science Foundation grant IRI–9015570.

Another place where this problem may arise is in a distributed system, for example a computer network linking a corporation's headquarters and branch offices.

Formally, the team wishes to determine a random n-bit sequence S satisfying agreement, secrecy, and uniformity. *Agreement* is met if each team player knows S. *Secrecy* is met if the eavesdroppers' probability of guessing S correctly is the same before and after hearing the communication between the team players. *Uniformity* requires that S has equal probability of being any one of the 2^n possible n-bit sequences. Such a secret key is said to be *shared* by the team. Each team player has an output tape that is physically protected from the other players. An *n-bit secret key exchange protocol* is one in which each team player outputs the same n-bit sequence satisfying the secrecy and uniformity conditions. The output can then be used for a variety of cryptographic purposes, for example, as the key in private key cryptosystems. (Cf. [4].)

We allow the eavesdroppers to be computationally unlimited, so standard cryptographic techniques based on computational difficulty cannot be used. In fact, a secret key exchange protocol is not possible without any further assumptions, for an eavesdropper can simulate any team player and thereby learn S. Hence, we give the players secret initial information in the form of correlated random variables. While the value of each player's random variable is unknown to the other players, the distribution from which the random variables are chosen is publically known. For any team that forms, the remaining players are assumed to collaborate against the team, possibly communicating among themselves via private channels. Thus we treat them as a single eavesdropper, Eve, who is given the initial information of all of the non-team players. Note that because the initial information is given before the team forms, it is not possible to deny Eve all initial information. We would like to distribute the initial information in such a way that any team that forms can obtain a secret key.

Our framework is very general and admits the trivial solution in which each player is given *a priori* a secret key for each team to which the player might eventually belong. Any team that forms can use the corresponding preassigned secret key, but since there is an exponential number of possible teams, the amount of initial information is quite high. Also, the structure of the initial random information is rather complicated.

We desire instead correlated random variables that have a simple structure and a small amount of initial information. A familiar example of such correlated random variables is provided by ordinary card games in which players are dealt hands from a randomly shuffled deck of cards. By looking at her own cards, a player gains some information about the other players' hands. Namely, she learns a set of cards that appear in no other player's hand. Peter Winkler developed bidding conventions for the game of bridge whereby one player could send her partner secret information about her hand that was totally unrelated to the actual bid and completely undecipherable to the opponents, even though the protocol was known to them [7, 9, 10, 11]. Fischer, Paterson and Rackoff [5] carried this idea further, using deals of cards for secret bit transmission between two players. We consider secret key exchange protocols based on such card games in the remainder of this paper.

In Section 2, we describe a simple 1-bit secret key exchange protocol that succeeds for all deals in which the team players' hands are sufficiently large relative to the size of the team and the size of Eve's hand. In Section 3, we present a protocol that improves on the first protocol in two ways. First, it establishes an n-bit secret key for arbitrary n. Second, it requires only that each team player hold an arbitrarily small fraction of the cards (assuming that the deck is sufficiently large). We present our formal model in Section 4. In Sections 5

and 6, we present some necessary conditions on the deal for a secret key exchange protocol to exist. In Section 7, we show that the protocol presented in Section 2 is optimal for a natural class of related protocols.

We introduce some terminology used in the remainder of the paper. A *deck* D is a finite set, whose elements we call cards; a *hand* is subset of D. Let d be the size of the deck. The cards in the deck are known to all the players, as is the size of each player's hand, but the cards in each player's hand are private to that player. In an $(h_1, h_2, \ldots, h_k; e)$-*deal*, each team player P_i is given a hand H_i such that $H_i \subseteq D$ and $|H_i| = h_i$. Eve is dealt a hand E such that $E \subseteq D$ and $e = |E| = d - \sum_{i=1}^{k} h_i$. The deal $\delta = (H_1, H_2, \ldots, H_k; E)$ is *legal* if $H_1, H_2, \ldots, H_k, E$ partition D. We call the description of the sizes of the hands, $\xi = (h_1, h_2, \ldots, h_k; e)$, the *signature*[1] of the deal, and call a deal having signature ξ a ξ-*deal*. If all k team players have the same hand size h in a signature, we write $(h^k; e)$.

An n-bit secret key exchange protocol that always succeeds in obtaining an n-bit secret key for all legal ξ-deals is said to *work for* ξ. We also say such a protocol *performs n-bit secret key exchange for* ξ.

2 A One-Bit Secret Key Exchange Protocol

We first consider a simple 1-bit secret key exchange protocol. We use the notion of a *key set* defined in [5]. A key set K consists of two cards, one held by a team player P, the other held by a *different* team player Q. A key set $K = \{x, y\}$ is *opaque* if, given the information available to Eve, it is equally likely that P holds x and Q holds y or that P holds y and Q holds x.

Once P and Q determine a opaque key set K that they hold, they can use it to obtain a bit r that is secret to Eve. Namely, they agree that $r = 0$ if P holds x and $r = 1$ if P holds y, or vice versa. Thus K acts as a 1-*bit secret channel*; that is, it allows P and Q to communicate a single bit secretly.

The structure of our protocol is as follows. We think of the team players as nodes of a graph. We connect two team players by an edge if the team players have a 1-bit secret channel between them. The goal of the protocol is to connect the team players. We obtain 1-bit secret channels by finding opaque key sets between pairs of team players until the team is connected. Then a designated player, say P_1, chooses a bit s randomly. Using flooding on the 1-bit secret channels, s is propagated to all the team players. Hence, s is a 1-bit secret key. Clearly s satisfies agreement and uniformity. Secrecy is satisfied because each 1-bit channel preserves secrecy.

We define the notion of a feasible player. Let each team player P_i hold h_i cards and let Eve hold e cards. Then P_i is *feasible* if $h_i > 1$, or if $h_i = 1$, $e = 0$, and $h_j > 1$ for all $j \neq i$. In the protocol that follows, we say a card x is *discarded* from the deck if all team players agree to play as if x is no longer part of the deck. Similarly, we say a team player P *drops out* of the protocol if the team players agree to play as if P were no longer part of the team. The protocol follows. All ties are broken by choosing the lower numbered player.

1. Let P be the feasible player holding the smallest hand. (Ties are broken in favor of the lower-numbered player.) If no player is feasible, then P is the lowest-numbered player holding a non-empty hand, if any.

[1] This term is borrowed from algebra, and is not intended to have any connection to digital signatures.

2. P chooses a random card x contained in her hand and a random card y not in her hand and proposes $K = \{x, y\}$ as a key set by asking, "Does any team player hold a card in K?"[2]

3. If another team player Q holds y, she knows that K is a key set, so she *accepts* K by announcing that she holds a card in K. The cards x and y are discarded. Whichever player of P and Q holds fewer cards announces the remaining cards in her hand, which are discarded, and drops out of the protocol. The remaining team players go back to step 1 with the "new" deal.

4. If none of the team players holds y, then K is *rejected*. In this case, x and y are discarded, and the players go back to step 1.

The execution of the protocol continues in this manner until either there are not enough cards left to complete step 1 or until only one team player is left. In the first case, the protocol fails. In the second case, all the team players are connected by opaque key sets. To see this, note that every key set $K = \{x, y\}$ accepted in step 3 is opaque because it is equally likely to be proposed by P in the symmetric deal where everything is the same except that P holds y and Q holds x. Hence the team can obtain a 1-bit secret key by flooding as previously described.

We call this protocol the SFP key set protocol (for smallest feasible player).

Theorem 1 *Let $\xi = (h_1, ..., h_k; e)$. Let $h_i \geq 1$ for $1 \leq i \leq k$, and $\max h_i + \min h_i \geq k + e$. Then the SFP key set protocol performs 1-bit secret key exchange for ξ.*

Proof: Define the function $\text{size}((h_1, \ldots, h_k; e)) = k + e$. We show by induction on $\text{size}(\xi)$ that the SFP key set protocol succeeds in connecting the team if ξ satisfies the conditions of the theorem. Recall that by assumption, $k \geq 2$.

If $\text{size}(\xi) = 2$ then, since $k \geq 2$, we have $k = 2$ and $e = 0$, and thus the signature is $(h_1, h_2; 0)$, for some $h_1, h_2 \geq 1$. A proposed key set is guaranteed to be accepted, and connects the team.

Inductively assume that the theorem holds for all ξ such that $\text{size}(\xi) = t$, and consider $\xi = (h_1, \ldots, h_k; e)$ satisfying the conditions of the theorem such that $\text{size}(\xi) = t + 1 > 2$.

Let $M = \max h_i$, and $m = \min h_i$. To show that the protocol works for ξ, we first note that since $\text{size}(\xi) > 2$, we have $M + m > 2$, and hence $M \geq 2$. After step 2 of the protocol, execution goes to either step 3 or step 4.

If step 4 is taken, then the proposed key set cards are discarded and execution returns to step 1 of the protocol with a "new" deal of some signature ξ'. The signature ξ' has the same number of team players as ξ, and Eve has one fewer card. Hence $\text{size}(\xi') = \text{size}(\xi) - 1$. Let M' and m' denote the maximum and the minimum of the team players' hand sizes in ξ, respectively. In going from ξ to ξ', only one team player's hand, which is of size at least 2, changes size, and that hand loses only one card. Hence at most one of the maximum or the minimum decreases, and if either decreases, it decreases by at most one. Hence $M' + m' \geq M + m - 1 \geq \text{size}(\xi) - 1 = \text{size}(\xi') = t$. By induction, we are done.

Suppose instead step 3 is taken. If $k = 2$, then this connects the team and the protocol works. Otherwise, $k > 2$, and execution returns to step 1 with a new deal of some signature

[2] In an abstract setting, $\{x, y\}$ is clearly the same as $\{y, x\}$. In an actual implementation, care must be taken that the communication of $\{x, y\}$ does not reveal which card came from P's hand.

ξ'. In ξ', there are $k-1$ team players, and Eve has the same number of cards, so size(ξ') = size(ξ) $-$ 1. As before let M' and m' be the size of the largest and smallest hand in ξ'. In this case, in going from ξ to ξ', one team player is entirely removed and one team player loses a card. Furthermore, the team player who loses a card has a larger hand than the player who is removed. Hence removing the player does not decrease the maximum, and can only increase the minimum.

If all the h_i are equal, then removing one card decreases the minimum by one and does not change the maximum. Hence $M = m \geq 2$ and so $M' = M \geq 2$, and $m' = M - 1 \geq 1$, so $M' + m' = M + m - 1 \geq \text{size}(\xi) - 1 = \text{size}(\xi')$, and we are done. If the h_i are not all equal, then the maximum decreases by at most one, and the minimum does not decrease. Hence $M' \geq M - 1 \geq 1$,and $m' \geq m \geq 1$. Thus again $M' + m' \geq \text{size}(\xi')$, and we are done. ■

In Section 7 we consider protocols with different rules for choosing P in step 1. We show there that the SFP key set protocol is optimal among all such protocols.

3 An n-Bit Secret Key Exchange Protocol

The SFP key set protocol has two limitations: it seems to require that the team hold more than half the cards in the deck, and it only provides a 1-bit secret key. Moreover, it is not obvious how to modify the protocol to overcome these limitations. For example, the protocol cannot be repeated to obtain additional key bits since players drop out and expose all of their remaining cards during execution.

The first limitation is overcome in [5] for a team of two players. A 1-bit secret key exchange protocol is presented there that works when each team player holds any fixed fraction of the cards and the deck is sufficiently large. An analysis of that protocol establishes the following:

Theorem 2 (Fischer, Paterson, Rackoff) *Let*

$$f(\beta) = \left(\frac{2}{\beta^2}\right) 2^{1/\beta}.$$

There exists a 1-bit secret key exchange protocol $\mathcal{P}$ such that for all $0 < \beta \leq 1/2$ and all $d \geq f(\beta)$, $\mathcal{P}$ works for $(\lfloor \beta d \rfloor, \lfloor \beta d \rfloor; d - 2\lfloor \beta d \rfloor)$.

We show how to use such a protocol to perform n-bit secret key exchange for teams of size k and sufficiently large decks. Our construction is a general reduction of the n-bit, k-player problem for signature $\xi^* = (h^k; d - kh)$ to the 1-bit, 2-player problem for signature $\xi = (\lfloor h/(2n) \rfloor, \lfloor h/(2n) \rfloor; d - 2\lfloor h/(2n) \rfloor)$. Thus, given a protocol $\mathcal{P}$ that performs 1-bit secret key exchange for ξ, we construct a new protocol $\mathcal{P}^*$ that performs n-bit secret key exchange for ξ^*.

Lemma 3 *Let $n \geq 1$, $k \geq 2$ and $d \geq kh$. Let $\mathcal{P}$ be a 1-bit secret key exchange protocol that works for*

$$\xi = \left(\left\lfloor \frac{h}{2n} \right\rfloor, \left\lfloor \frac{h}{2n} \right\rfloor; d - 2\left\lfloor \frac{h}{2n} \right\rfloor\right).$$

Then there is a protocol $\mathcal{P}^*$ that perfoms n-bit secret key exchange for

$$\xi^* = (h^k; d - kh).$$

Proof: Suppose n, k, d, h, $\mathcal{P}$, ξ, and ξ^* satisfy the conditions of the lemma. We construct an n-bit secret key exchange protocol $\mathcal{P}^*$ that works for ξ^*.

Assume the players are linearly ordered, say, by their indices. Two team players are said to be *neighbors* if they are adjacent in the ordering. P_1 is the leader and randomly chooses an n-bit string S to be the secret key. Next, each pair of neighbors P_i and P_{i+1} establishes an n-bit secret key B_i that they share. They will use B_i later as a one-time pad for private communication between themselves. Finally, S is propagated secretly down the chain of players as follows: P_1 sends S, encrypted by B_1, to P_2, then P_2 sends S, encrypted by B_2, to P_3, and so forth until all team players know S.

More specifically, P_1 chooses an n-bit string S at random. Each pair of neighbors P_i and P_{i+1} uses $\mathcal{P}$ a total of n times, as described in detail below, to get an n-bit secret string B_i which they share. Later, when P_i learns S, she sends $E_i = S \oplus B_i$ to P_{i+1} publicly. P_{i+1} recovers S by computing $E_i \oplus B_i$.

We now describe in detail how the one-time pads are established. Given a team player P_i, we say P_{i+1} is the *right neighbor* of P_i and P_{i-1} is the *left neighbor* of P_i. Each player P_i divides her hand into $2n$ parts, H_i^1 through H_i^{2n}, of size $\lfloor h/(2n) \rfloor$ and a (possibly empty) part containing her remaining cards. P_i uses parts H_i^1 though H_i^n to establish B_i with her right neighbor, and she uses parts H_i^{n+1} through H_i^{2n} to establish B_{i-1} with her left neighbor.

The j^{th} bit of the one-time pad B_i is gotten as follows. P_i plays the role of player 1 in $\mathcal{P}$, pretending that the only cards she holds are those in H_i^j. P_{i+1} plays the role of player 2 in $\mathcal{P}$, pretending that the only cards she holds are those in H_{i+1}^{n+j}. The other team players do not participate. We call the cards in $H_i^j \cup H_{i+1}^{n+j}$ the *current cards*. Both players pretend that Eve holds all but the current cards. Thus P_i and P_{i+1} execute $\mathcal{P}$ as if the deal were a ξ-deal. Since $\mathcal{P}$ is assumed to work for ξ, P_i and P_{i+1} obtain a shared secret bit, which they use for the j^{th} bit of B_i.

Note that whenever a card x not in the current cards is referenced, all players behave as if Eve holds x. If Eve does not hold x, she learns that x does not lie in the current cards, but she learns nothing further about the location of x. Thus this process can be repeated, using each part of each team player's hand exactly once, to get all the one-time pads. ■

We now apply Lemma 3 to families of 1-bit protocols.

Theorem 4 *Let $n \geq 1$, $k \geq 2$, and let f be a function on the reals. Suppose for every $0 < \beta \leq 1/4$ and every $d \geq f(\beta)$ that there is a 1-bit secret key exchange protocol $\mathcal{P}$ that works for $(\lfloor \beta d \rfloor, \lfloor \beta d \rfloor; d - 2\lfloor \beta d \rfloor)$. Let $0 < \alpha \leq 1/k$, and let $d \geq f(\alpha/(2n))$. Let $\mathcal{P}^*$ be the protocol constructed as in the proof of Lemma 3. Then $\mathcal{P}^*$ performs n-bit secret key exchange for $(\lfloor \alpha d \rfloor^k; d - k\lfloor \alpha d \rfloor)$.*

Proof: Assume the hypotheses of the protocol, and assume we are given a deal of signature $\xi = (\lfloor \alpha d \rfloor^k; d - k\lfloor \alpha d \rfloor)$. Let $h = \lfloor \alpha d \rfloor$ and let $\beta = \alpha/(2n)$. Since $\alpha \leq 1/k$, it follows that $d \geq k\lfloor \alpha d \rfloor = kh$ and $\beta \leq 1/4$. Also, since n is an integer, $\lfloor \beta d \rfloor = \lfloor \alpha d/(2n) \rfloor = \lfloor \lfloor \alpha d \rfloor /(2n) \rfloor = \lfloor h/(2n) \rfloor$. Hence, $\mathcal{P}$ satisfies the conditions for Lemma 3. It follows from Lemma 3 that $\mathcal{P}^*$ performs n-bit secret key exchange for $(h^k; d - kh) = \xi$ as desired. ■

The following corollary to Theorem 4 is immediate using Theorem 2.

Corollary 5 *Let* $0 < \alpha \leq 1/k$. *Suppose*

$$d \geq 8\left(\frac{n}{\alpha}\right)^2 2^{2n/\alpha}.$$

Then $\mathcal{P}^*$ *performs* n*-bit secret key exchange for* $(\lfloor \alpha d \rfloor^k; d - k\lfloor \alpha d \rfloor)$.

Unfortunately, the required deck size here grows exponentially in n/α. Richard Beigel [2] has suggested an improved 1-bit two-player protocol in which the deck size appears to grow only polynomially in $1/\alpha$. Using such a protocol protocol, our construction yields an n-bit team protocol for which the deck grows only polynomially in n/α.

4 The Formal Model

Before proceeding, we define our model more precisely. We look at a synchronous distributed model of computation in which there is a *team* of k players P_1 through P_k and a passive eavesdropper, Eve. Let $\mathcal{P}$ be an n-bit secret key exchange protocol for P_1 through P_k. In each round of $\mathcal{P}$, each of the team players simultaneously broadcasts a message to all of the other players. All messages are overheard by Eve. Let Z be the set of possible messages, and let $z_i \in Z$ be the message that each P_i sends in the round. The k-tuple $(z_1, z_2, \ldots, z_k) \in Z^k$ is called a *statement* of $\mathcal{P}$. A sequence τ of statements is called a *conversation* of $\mathcal{P}$, denoted by $\tau_{\mathcal{P}}$. If the conversation σ is a prefix of the conversation τ, we write $\sigma \preceq \tau$. We assume each protocol $\mathcal{P}$ always terminates after some fixed number $t_{\mathcal{P}}$ of rounds. A conversation $\tau_{\mathcal{P}}$ is *complete* if $|\tau_{\mathcal{P}}| = t_{\mathcal{P}}$. Formally, $\mathsf{c}_{\mathcal{P}} = \bigcup_{\ell=0}^{t_{\mathcal{P}}} (Z^k)^\ell$ is the set of conversations, $\mathsf{cc}_{\mathcal{P}} = (Z^k)^{t_{\mathcal{P}}}$ is the set of complete conversations, and $\mathsf{pc}_{\mathcal{P}} = \mathsf{c}_{\mathcal{P}} - \mathsf{cc}_{\mathcal{P}}$ is the set of partial conversations. As it will be clear from context which protocol is being discussed, we will omit the protocol subscripts from $t_{\mathcal{P}}, \tau_{\mathcal{P}}, \mathsf{pc}_{\mathcal{P}}, \mathsf{cc}_{\mathcal{P}}$, and $\mathsf{c}_{\mathcal{P}}$.

The protocol run by each player P_i is a randomized algorithm that determines the message for P_i to send at each round based on her hand and the conversation so far. Specifically, let $\mathcal{H}_i$ be the set of possible hands for P_i. Let $H_i \in \mathcal{H}_i$, $\tau \in \mathsf{cc}$ and $\sigma \in \mathsf{pc}$. A protocol for P_i is a pair $(\mu_i, \mathcal{O}_i)$, where $\mu_i(H_i, \sigma)$ is a random variable over the message space Z and $\mathcal{O}_i(H_i, \tau) \in \{0,1\}^n$ specifies an output value. Thus, $\mathrm{Prob}[\mu_i(H_i, \sigma) = z]$ is the probability that P_i sends message z at round $r+1$ given that P_i holds hand H_i and the conversation through round r is σ. While in general the output function is also be a random variable, agreement can never be assured unless each player's output value is uniquely determined by her hand and the conversation. Hence without loss of generality, we restrict the output function $\mathcal{O}_i : \mathcal{H}_i \times \mathsf{cc} \to \{0,1\}^n$ to be a deterministic function.

A *joint protocol* for players P_1 through P_k consists of a set of protocols $(\mu_i, \mathcal{O}_i)$, where each $(\mu_i, \mathcal{O}_i)$ is a protocol for P_i. All the protocols $(\mu_i, \mathcal{O}_i)$ are known to each team player, as well as to Eve. Thus an n-bit secret key exchange protocol that works for ξ is a joint protocol $\{(\mu_1, \mathcal{O}_1), \ldots, (\mu_k, \mathcal{O}_k)\}$ for the team players such that for all possible runs on each legal ξ-deal, if every team player P_i plays according to $(\mu_i, \mathcal{O}_i)$, the team succeeds in obtaining an n-bit secret key. It is a straightforward exercise to modify the protocols we describe in English in this paper to fit this model.

5 A Necessary Condition for Secret Key Exchange

Having seen some conditions under which a deal for a secret key exchange protocol exists, we may wonder when such a protocol does not exist. We generalize a theorem of [5] that shows secret key exchange is not possible if the deal does not provide sufficient shared information.

We begin by exploring the correlation among hands in a legal deal. Throughout this section, we fix a deck D and a signature $\xi = (h_1, h_2, \ldots, h_k; e)$ such that $\sum_{i=1}^{k} h_i + e = |D|$.

Recall that a ξ-deal of a deck D is a collection of $k+1$ hands $(H_1, \ldots, H_k; E)$ such that $|H_i| = h_i$ for $i \in \{1, \ldots, k\}$ and $|E| = e$, and recall that a deal is legal if the hands partition D. We sometimes use the term "general deal" to refer to a deal that is not necessarily legal. Let Δ' be the set of all (general) ξ-deals of D, and let Δ be the set of legal ξ-deals of D. Note that $\Delta \subseteq \Delta'$ and that a general deal δ is legal iff the hands in δ are pairwise disjoint. When speaking of deals, we will sometimes use the notation H_i^δ to refer to the hand of player P_i and E^δ to refer to Eve's hand in deal δ; thus, $\delta = (H_1^\delta, H_2^\delta, \ldots, H_k^\delta; E^\delta)$.

A random legal deal is a uniformly distributed random variable over Δ. A random general deal is a uniformly distributed random variable over Δ'. Note that in both a random legal deal and in a random general deal, each hand H_i is uniformly distributed over $\mathcal{H}_i$. The difference is that in a random general deal, the hands $H_1, \ldots, H_k$ are independent random variables, whereas in a random legal deal, they are correlated. Hence, only in a random legal deal does player P_i get any information about the cards in other player's hands.

Let $\overline{\gamma}$ be the probability that a random general deal is also a legal deal. Intuitively, the smaller $\overline{\gamma}$ is, the more shared information the deal contains. The following theorem provides an upper bound on $\overline{\gamma}$ in order for n-bit secret exchange to be possible.

Theorem 6 *Let ξ and $\overline{\gamma}$ be as defined above, and let $n \geq 1$. If $\overline{\gamma} > 1/2^{k-1}$, then no protocol performs n-bit secret key exchange for ξ.*

For the proof of this theorem, we will need a lemma about real numbers.

Lemma 7 *Let $x_1, x_2, \ldots, x_n, y_1, y_2, \ldots, y_n$ be nonnegative. Then*

$$\min\left(\prod_{i=1}^{n} x_i, \prod_{i=1}^{n} y_i\right) \leq \frac{1}{2^n}\left(\prod_{i=1}^{n}(x_i + y_i)\right)$$

The proof of Lemma 7 uses the theorem of the arithmetic and geometric means (AGM), which says that if a_1 and a_2 are nonnegative, then $\sqrt{a_1 a_2} \leq (a_1 + a_2)/2$.

Proof: (of Lemma 7) Assuming $x_1, x_2, \ldots, x_n, y_1, y_2, \ldots, y_n$ are nonnegative, we have

$$\min\left(\prod_{i=1}^{n} x_i, \prod_{i=1}^{n} y_i\right) \leq \sqrt{\left(\prod_{i=1}^{n} x_i\right)\left(\prod_{i=1}^{n} y_i\right)} \tag{1}$$

$$= \prod_{i=1}^{n} \sqrt{x_i y_i} \tag{2}$$

$$\leq \prod_{i=1}^{n} \left(\frac{x_i + y_i}{2}\right) \tag{3}$$

$$= \frac{1}{2^n}\prod_{i=1}^{n}(x_i + y_i) \tag{4}$$

Here, line 1 holds because the square root of the product of two numbers is always at least as big as the smaller of the two numbers. Line 3 is by the AGM. Lines 2 and 4 are direct algebraic manipulation. ■

Proof: (of Theorem 6) Suppose $\mathcal{P}$ is an arbitrary protocol that performs 1-bit secret key exchange for ξ. We show that $\overline{\gamma} \leq 1/2^{k-1}$. This implies that no n-bit secret key exchange protocol exists for any $n \geq 1$ when $\overline{\gamma} > 1/2^{k-1}$, because in an n-bit protocol, each bit in the sequence is a 1-bit secret key.

We begin by analyzing how a random conversation $\sigma \in \mathsf{c}$ is developed from the random message functions μ_i of the players P_i. Let $H_i \in \mathcal{H}_i$ and let $\sigma = \sigma_1\sigma_2\cdots\sigma_r \in \mathsf{c}$ be a (partial or complete) conversation of $\mathcal{P}$. For each $1 \leq \ell \leq r$, let $\sigma^\ell = (z_1^\ell, z_2^\ell, \ldots, z_k^\ell)$ be the collective statement made by the players at round ℓ of σ.

Let $\nu_i^\ell(H_i, \sigma, z_i^\ell)$ be the event that $\mu_i(H_i, \sigma_1\cdots\sigma_{\ell-1}) = z_i^\ell$, that is, the event that the message which player P_i sends at round ℓ is the one specified for it by σ in a run in which player P_i holds hand H_i and the partial conversation up through round $\ell - 1$ is $\sigma_1\cdots\sigma_{\ell-1}$. These events take the place of random private coins in our model and are assumed to be independent.

Let $\zeta_i(H_i, \sigma)$ be the joint event $\nu_i^1(H_i, \lambda, z_i^1)\&\ldots\&\nu_i^r(H_i, \sigma_1\cdots\sigma_{r-1}, z_i^r)$. Then $\zeta_i(H_i, \sigma)$ is the event that the behavior of player P_i is consistent with σ at every round when P_i holds hand H_i and receives messages at each round as specified by σ. Since $\mathrm{Prob}[\nu_i^\ell(H_i, \sigma, z_i^\ell)] = \mathrm{Prob}[\mu_i(H_i, \sigma_1\cdots\sigma_{\ell-1}) = z_i^\ell]$, it follows that

$$\mathrm{Prob}[\zeta_i(H_i, \sigma)] = \prod_{\ell=1}^{r}\mathrm{Prob}[\mu_i(H_i, \sigma_1\cdots\sigma_{\ell-1}) = z_i^\ell]$$

Now let $\sigma \in \mathsf{cc}$ be a complete conversation and $H_i \in \mathcal{H}_i$. Because $\mathcal{P}$ is assumed to be a correct 1-bit secret key exchange protocol, each P_i produces an output value $\mathcal{O}_i(H_i, \sigma) \in \{0, 1\}$. For $j \in \{0, 1\}$ we define

$$\begin{aligned} C_i^j(H_i, \sigma) &= \mathrm{Prob}[\zeta_i(H_i, \sigma)\ \&\ P_i \text{ outputs } j] \\ &= \begin{cases} \mathrm{Prob}[\zeta_i(H_i, \sigma)] & \text{if } \mathcal{O}_i(H_i, \sigma) = j \\ 0 & \text{otherwise} \end{cases} \end{aligned}$$

Thus, $C_i(H_i, \sigma) = C_i^0(H_i, \sigma) + C_i^1(H_i, \sigma) = \mathrm{Prob}[\zeta_i(H_i, \sigma)]$.

A run of the joint protocol $\mathcal{P}$ given deal δ results in conversation σ iff each player P_i plays according to σ when given hand H_i^δ. Because the events $\zeta_1(H_1^\delta, \sigma), \ldots, \zeta_k(H_k^\delta, \sigma)$ are independent, the probability $C(\delta, \sigma)$ of this occurring is just the product of the probabilities of each player playing according to σ. Thus, we have

$$C(\delta, \sigma) = \prod_{i=1}^{k} C_i(H_i^\delta, \sigma). \tag{5}$$

Since always *some* complete conversation results, we have

$$\sum_{\sigma \in \mathsf{cc}} C(\delta, \sigma) = 1. \tag{6}$$

Define

$$c_i^j(\sigma) = \sum_{H_i \in \mathcal{H}_i} C_i^j(H_i, \sigma).$$

Then $c_i^j(\sigma)$ is the probability that P_i plays according to σ and outputs value j given a random hand in $\mathcal{H}_i$. We also define

$$f^j(\sigma) = \sum_{\delta \in \Delta} \prod_{i=1}^{k} C_i^j(H_i^\delta, \sigma). \tag{7}$$

Then $f^j(\sigma)/|\Delta|$ is the probability that all team players play according to σ and output j when given hands from a random legal deal in Δ.

We complete the proof by establishing

$$|\Delta| = \sum_{\sigma \in \mathsf{CC}} \left(f^0(\sigma) + f^1(\sigma)\right) \leq \frac{1}{2^{k-1}} \cdot \sum_{\sigma \in \mathsf{CC}} \prod_{i=1}^{k} \left(c_i^0(\sigma) + c_i^1(\sigma)\right) = \frac{1}{2^{k-1}} \cdot |\Delta'|. \tag{8}$$

It will follow immediately that

$$\overline{\gamma} = \frac{|\Delta|}{|\Delta'|} \leq \frac{1}{2^{k-1}},$$

as desired.

Let δ be any legal deal and let σ be any conversation which is possible for δ, that is, for which $C(\delta, \sigma) > 0$. Hence, for each i, at least one of $C_i^0(H_i^\delta, \sigma)$ and $C_i^1(H_i^\delta, \sigma)$ must be non-zero. By the agreement condition, either $C_i^0(H_i^\delta, \sigma) > 0$ for all i, or $C_i^1(H_i^\delta, \sigma) > 0$ for all i. Moreover, also by agreement, $C_i^0(H_i^\delta, \sigma)$ and $C_i^1(H_i^\delta, \sigma)$ cannot both be non-zero for any player P_i. It follows from these facts that

$$\prod_{i=1}^{k} C_i^0(H_i^\delta, \sigma) + \prod_{i=1}^{k} C_i^1(H_i^\delta, \sigma) = C(\delta, \sigma) \tag{9}$$

Using lines 6, 7, and 9, we get

$$\begin{aligned} \sum_{\sigma \in \mathsf{CC}} (f^0(\sigma) + f^1(\sigma)) &= \sum_{\sigma \in \mathsf{CC}} \sum_{\delta \in \Delta} \left(\prod_{i=1}^{k} C_i^0(H_i^\delta, \sigma) + \prod_{i=1}^{k} C_i^1(H_i^\delta, \sigma) \right) \\ &= \sum_{\delta \in \Delta} \sum_{\sigma \in \mathsf{CC}} C(\delta, \sigma) \\ &= \sum_{\delta \in \Delta} 1 \\ &= |\Delta| \end{aligned} \tag{10}$$

The following is immediate from the definitions and the fact that $\Delta \subseteq \Delta'$:

$$f^j(\sigma) \leq \sum_{\delta \in \Delta'} \prod_{i=1}^{k} C_i^j(H_i^\delta, \sigma) = \prod_{i=1}^{k} \sum_{H_i \in \mathcal{H}_i} C_i^j(H_i, \sigma) = \prod_{i=1}^{k} c_i^j(\sigma). \tag{11}$$

The secrecy condition implies that $f^0(\sigma) = f^1(\sigma)$, since otherwise Eve could determine from a complete conversation σ which output was more likely to have occurred. Thus, by line 11 and Lemma 7, we have

$$\begin{aligned} f^0(\sigma) + f^1(\sigma) &\leq 2 \cdot \min\left(\prod_{i=1}^{k} c_i^0(\sigma), \prod_{i=1}^{k} c_i^1(\sigma)\right) \\ &\leq 2 \cdot \frac{1}{2^k}\left(\prod_{i=1}^{k}\left(c_i^0(\sigma) + c_i^1(\sigma)\right)\right) \\ &= \frac{1}{2^{k-1}}\left(\prod_{i=1}^{k}\left(c_i^0(\sigma) + c_i^1(\sigma)\right)\right) \end{aligned}$$

Summing over all complete conversations, we get

$$\sum_{\sigma \in CC}\left(f^0(\sigma) + f^1(\sigma)\right) \leq \frac{1}{2^{k-1}} \cdot \sum_{\sigma \in CC} \prod_{i=1}^{k}\left(c_i^0(\sigma) + c_i^1(\sigma)\right) \tag{12}$$

The following is also immediate from the definitions:

$$\begin{aligned} \sum_{\sigma \in CC} \prod_{i=1}^{k}\left(c_i^0(\sigma) + c_i^1(\sigma)\right) &= \sum_{\sigma \in CC} \prod_{i=1}^{k} \sum_{H_i \in \mathcal{H}_i}\left(C_i^0(H_i,\sigma) + C_i^1(H_i,\sigma)\right) \\ &= \sum_{\delta \in \Delta'} \sum_{\sigma \in CC} \prod_{i=1}^{k} C(\delta,\sigma) \end{aligned} \tag{13}$$

Applying line 6 gives

$$\sum_{\sigma \in CC} \prod_{i=1}^{k}\left(c_i^0(\sigma) + c_i^1(\sigma)\right) = \sum_{\delta \in \Delta'} 1 = |\Delta'| \tag{14}$$

Combining lines 10, 12, and 14 yields line 8 as desired, completing the proof. ■

We remark that the above proof goes through even for protocols in which Eve is not allowed to look at her hand. Thus, our proof applies to a larger class of protocols than necessary. We do not know how to use Eve's ability to see her cards to improve this result.

Corollary 8 *Let $n \geq 1$ and $2 \leq k \leq 8$. Then no protocol performs n-bit secret key exchange for $(1^k;1)$.*

Proof: In these cases, $\overline{\gamma} = (k+1)!/(k+1)^k > 1/2^{k-1}$. ■

For $k > 8, \overline{\gamma} = (k+1)!/(k+1)^k < 1/2^{k-1}$, so nothing can be concluded.

6 The impossibility of $(1,1,1;0)$

Theorem 6 says nothing about the $(1^k;0)$ case. However, it is possible to show the following.

Theorem 9 *Let $n \geq 1$. Then no protocol performs n-bit secret key exchange for $(1,1,1;0)$.*

In order to prove Theorem 9, we need only show that no protocol performs 1-bit secret key exchange for $(1,1,1;0)$. To prove this, we look at properties of the possible conversations of a 1-bit secret key exchange protocol on $(1,1,1;0)$-deals. Suppose $\mathcal{P} = \{(\mu_1, \mathcal{O}_1), (\mu_2, \mathcal{O}_2), (\mu_3, \mathcal{O}_3)\}$ performs 1-bit secret key exchange for $(1,1,1;0)$. Let the deck $D = \{0,1,2\}$, let Δ be the set of legal $(1,1,1;0)$-deals of deck D, and let cc be the set of complete conversations of $\mathcal{P}$ on deals in Δ. We denote each of the three possible single card hands by the card comprising the hand, and we denote each deal by the permutation of $(0,1,2)$ describing the deal.

Let $\tau \in$ cc. We say that τ is *realizable* if there is some $\delta \in \Delta$ such that τ is a possible conversation of the protocol when the deal is δ, and in this case we say δ is *consistent* with τ. An output $v \in \{0,1\}$ is *possible* given τ if there is some $\delta = (H_1, H_2, H_3) \in \Delta$ consistent with τ such that $v = \mathcal{O}_i(H_i, \tau)$ for each i.

For a realizable conversation τ, we define the function T_τ with range $\{0,1,-\}$.

$$T_\tau(P_i, H) = \begin{cases} - & \text{if } C_i(H,\tau) = 0 \\ \mathcal{O}(H,\tau) & \text{otherwise} \end{cases}$$

For a given τ, we denote the matrix

$$\begin{pmatrix} T_\tau(P_1,0) & T_\tau(P_2,0) & T_\tau(P_3,0) \\ T_\tau(P_1,1) & T_\tau(P_2,1) & T_\tau(P_3,1) \\ T_\tau(P_1,2) & T_\tau(P_2,2) & T_\tau(P_3,2) \end{pmatrix}$$

by $\langle T_\tau(P_i,j)\rangle$. Lemma 10 says that for a realizable conversation τ, each row and column of $\langle T_\tau(P_i,j)\rangle$ has exactly one 0, one 1, and one $-$.

Lemma 10 *If $\tau \in$ cc, then for each $i \in \{1,2,3\}$ and for each $j \in \{0,1,2\}$,*

$$\{T_\tau(P_i,0), T_\tau(P_i,1), T_\tau(P_i,2)\} = \{0,1,-\}$$

$$\{T_\tau(P_1,j), T_\tau(P_2,j), T_\tau(P_3,j)\} = \{0,1,-\}.$$

Proof: Suppose τ is a realizable conversation. Then outputs 0 and 1 must both be possible given τ. Otherwise, whenever Eve hears τ, she will know that the output value must be the one possible value. Thus there are two disjoint deals $\delta^0 = (H_1^0, H_2^0, H_3^0)$ and $\delta^1 = (H_1^1, H_2^1, H_3^1)$ such that $T_\tau(P_i, H_i^0) = 0$ and $T_\tau(P_i, H_i^1) = 1$. In order to satisfy agreement, the remaining values of $\langle T_\tau(P_i,j)\rangle$ must all be $-$. ■

Thus exactly two deals are consistent with each realizable conversation and there are exactly 12 distinct possible matrices $\langle T_\tau(P_i,j)\rangle$ for realizable conversations τ. We define the *parity* of a deal to be the parity of the permutation describing the deal. Inspection shows that both of the deals consistent with a realizable conversation have the same parity. We say that the parity of a realizable conversation τ is the parity of the two deals consistent with τ. We are now ready to derive a contradiction.

Proof: (of Theorem 9) Suppose $\mathcal{P}$ performs 1-bit secret key exchange for $(1,1,1;0)$. We construct a tree of conversations as follows. The nodes of the tree are conversations, and the edges out of a node are labeled by possible next statements. Thus the interior nodes are partial conversations; leaf nodes are complete conversations. A conversation τ *passes*

through a node σ if $\sigma \preceq \tau$. We say a node is *single valued* if all conversations passing through it have the same parity. It is *multivalued* otherwise.

By the correctness of $\mathcal{P}$, all $(1,1,1;0)$-deals must be possible initially. Thus the root of the tree is multivalued. Because only one conversation passes through any leaf node, all leaves are single valued. Hence there must be a multivalued node σ having only single valued children. Thus there exist complete conversations τ_0 and τ_1 passing through σ such that τ_0 has parity 0 and τ_1 has parity 1.

Then there exist i and H such that $T_{\tau_0}(P_i, H) = T_{\tau_1}(P_i, H) = -$. Without loss of generality, the two deals consistent with τ_0 are $(0,1,2)$ and $(1,2,0)$, and the two deals consistent with τ_1 are $(0,2,1)$ and $(2,1,0)$.

Let Z^0, Z^1 be statements such that $\sigma Z^i \preceq \tau_i$, for $i \in \{0,1\}$, where $Z^i = (z_1^i, z_2^i, z_3^i)$. Then $\hat{Z} = (z_1^0, z_2^1, z_3^1)$ is a possible next statement for deals $\delta_0 = (0,2,1)$, and $\delta_1 = (1,2,0)$.

The two deals δ_0 and δ_1 have opposite parity, so $\sigma\hat{Z}$ is in fact a multivalued child of σ, contradicting the nonexistence of a multivalued child. ■

This proof is highly dependent on the specific properties of the set of possible $\langle T_\tau(P_i, j)\rangle$ matrices of $(1,1,1;0)$-deals, and does not generalize easily to larger teams. However, using an extension to the graph theoretical framework developed by Beaver, Haber and Winkler [1] to represent shared knowledge between two players, it is possible [6] to prove that for any $n \geq 1$ and $k \geq 3$, no protocol performs n-bit secret key exchange for $(1^k; 0)$.

7 Key Set Protocols Revisited

Even for the simple case of $n = 1$, there is a large gap between signatures for which we have a secret key exchange protocol and signatures for which we have shown that no protocol exists. For example, $(2,2,2;2)$ falls into this gap.

One approach to closing the gap is to modify the SFP key set protocol presented in Section 2. In step 1 of this protocol, a team player P, *the proposer*, is chosen. By considering different rules for choosing the proposer, we get a class of protocols. We call such a rule a *proposing rule*. We require a proposing rule to be a deterministic function of the current signature. We call the protocol that results from proposing rule $\mathcal{R}$ the *$\mathcal{R}$ key set protocol.* We call the class of all such protocols the class of key set protocols. By this definition, the SFP key set protocol results from the *smallest feasible player* proposing rule (SFP): If any team player is feasible, the feasible player with the smallest hand is chosen. (Ties are broken in favor of the lower-numbered player.) If no team player is feasible, the lowest-numbered team player holding a non-empty hand is chosen, if any.

Theorem 1 holds for any $\mathcal{R}$ key set protocol where $\mathcal{R}$ always chooses a feasible player if some team player is feasible. However, the converse does not in general hold. For example, the signature $\xi = (3,3,2,1;1)$ does not satisfy the conditions of the theorem, but the SFP key set protocol works for ξ. We have been unable to find an exact characterization of the signatures for which the SFP key set protocol works. Nevertheless, it is possible to show that the SFP key set protocol is optimal for the class of key set protocols. By this we mean that for a signature ξ, if the $\mathcal{R}$ key set protocol works for ξ for some $\mathcal{R}$, then the SFP key set protocol also works for ξ. To prove this we look at a simple combinatorial stick game between a team and an adversary. The stick game abstracts the important aspects of the key set protocol.

7.1 The Stick Game

A configuration of the stick game consists of k team piles P_1 through P_k and a pile E. Pile P_i contains $|P_i|$ sticks, where $|P_i| \geq 0$, and pile E contains e sticks. On the team's turn, the team designates a team pile, say P_i, such that $|P_i| \geq 1$. On the adversary's turn, the adversary either removes one stick from P_i and one from E (this move is only allowed when $|E| \geq 1$), or chooses another team pile P_j such that $|P_j| \geq 1$, removes the smaller of P_i and P_j entirely, and removes one stick from the larger pile. Note that removing a pile is not the same as removing all the sticks in the pile. The team always moves first.

A configuration of the stick game is described by the tuple $(h_1, \ldots, h_k; e; I)$. I is called the *index component.* If it is the team's turn, I has the value T. If it is the adversary's turn, then I is the index of the pile designated by the team on its previous turn. We generally talk about a team move as a choice of P_i, and an adversary move as a match, either with some P_j or with E.

Play ends when there are one or zero team piles, in which case the team wins, or when there is no move available (either to the team or to the adversary), in which case the team loses. We call the stick game starting from configuration C the C *stick game.*

A *strategy* for the team, or team strategy, is a function that, given a configuration with index component T, specifies the next team move. That is, it gives a P_i ($i \in \{1, \ldots, k\}$) such that $|P_i| \geq 1$. Similarly, an adversary strategy is a function that specifies the next adversary move. That is, given a configuration with index component P_i it returns either E, or a P_j such that $j \neq i$ and $|P_j| \geq 1$.

We say a configuration C is *winning* if there is some team strategy $\mathcal{S}$ such that if the team plays the C stick game by strategy $\mathcal{S}$, then the team wins regardless of the moves chosen by the adversary. We say $\mathcal{S}$ is a *successsful team strategy for* C. We call $\mathcal{S}$ an *optimal team strategy* if it is a successful team strategy for every winning configuration C. Similarly, we say that C is *losing* if there is some adversary strategy $\mathcal{A}$ such that if the adversary plays the C stick game by strategy $\mathcal{A}$, then the team loses regardless of the team moves chosen. The notions of $\mathcal{A}$ being a *successful adversary strategy for* C and an *optimal adversary strategy* are defined similarly.

The stick game is a finite game, since every adversary turn decreases the total number of sticks by at least two. Furthermore, it is a game of complete information, since the team and the adversary take turns and all information about the state is known to both the team and the adversary. Hence game theory tells us that every configuration is either winning or losing, and an optimal team strategy $\mathcal{S}$ and an optimal adversary strategy $\mathcal{A}$ both exist [3].

We define a feasible pile in a stick game configuration exactly as we defined a feasible player in a signature, and we similarly define the SFP strategy for the team in the stick game. A team pile P_i is *feasible* if $|P_i| \geq 2$ or if $|P_i| = 1, |E| = 0$ and for all $j \neq i, |P_j| > 1$. The SFP strategy for the team in the stick game chooses i such that P_i is the feasible pile with the smallest size. In the case of ties, the smallest such i is chosen. If no team pile is feasible, then the lowest-numbered non-empty pile P_i is chosen, if any. (Here, SFP stands for smallest feasible *pile*). If all piles are empty, the game ends.

7.2 Correspondence between Key Set Protocols and the Stick Game

There is a close connection between the stick game and key set protocols. Namely, a configuration whose index component is T corresponds to a signature. A team's strategy

$\mathcal{S}$ in the stick game corresponds to a proposing rule $\mathcal{R}$ for the key set protocol. Namely, the pile P_i is chosen in the stick game iff player P_i proposes in the $\mathcal{R}$ key set protocol. The adversary's response in the stick game corresponds to naming the holder of the second card of the proposed key set. Finally, a winning end configuration in the stick game corresponds to a signature in which the team has trivially achieved a shared secret key (since the team is of size at most one), and a losing end configuration in the stick game corresponds a configuration in which no key set is possible (since the team has size two or greater and all but possibly one player has run out of cards).

We claim that a configuration in the stick game is winning for a given team strategy if and only if the key set protocol works for the corresponding signature when the team plays according to the corresponding proposing rule.

In the one direction, the claim is obvious: namely, if the team wins in the stick game, then it wins in the corresponding key set protocol. In the other direction, there is a complication in that the adversary in the stick game has complete control over which pile to match, whereas in the key set protocol, the hand containing the matching card is determined by the *a priori* deal. However, the matching card could be in any of the other hands since the proposed key set consists of two cards not previously mentioned. This allows us to show the following.

Lemma 11 (Stick Game Lemma) *Let stick game strategy $\mathcal{S}$ correspond to proposing rule $\mathcal{R}$. Then $\mathcal{S}$ is a successful team strategy from configuration $(h_1, \ldots, h_k; e; T)$ if and only if the $\mathcal{R}$ key set protocol works for $(h_1, \ldots, h_k; e)$.*

By Lemma 11, the optimality of the SFP stick game strategy implies the optimality of the corresponding SFP key set protocol.

7.3 Optimality of the SFP Stick Game Strategy

For convenience, we define a size function $\text{size}((h_1, \ldots, h_k; e; I)) = k + e$. We prove several lemmas by induction on the size of a configuration. Looking at the key exchange problem, we noted that a team player holding no cards could never learn any secret information. Lemma 12 makes this precise in regards to the stick game.

Lemma 12 *Let $C = (h_1, \ldots, h_k; e; T)$. If $k \geq 2$ and $h_i = 0$ for some i, then C is a losing configuration.*

Proof: We prove this by induction on $\text{size}(C)$. Without loss of generality, we assume $i = 1$. If $\text{size}(C) = 2$, then since $k \geq 2$, we have $e = 0$ and $k = 2$. Hence the configuration is $(0, h_2; 0; T)$. If $h_2 = 0$, then the team can not move, and so loses. If $h_2 > 0$, the team must choose P_2. At this point there is no adversary move, and so the team loses.

Inductively assume that the lemma holds when $\text{size}(C) = m \geq 2$. Consider a configuration $C = (h_1, \ldots, h_k; e; T)$ in which $\text{size}(C) = m + 1$, and suppose C satisfies the conditions of the lemma. Because $k \geq 2$, the team must choose a team pile, say P_j for which $|P_j| \geq 1$. Hence, $j \neq i$. The adversary must choose a nonempty pile to match with and also does not choose P_i. Hence, the resulting configuration C' still has a pile of size 0. Because any adversary match lowers either the number of piles or the size of E by one, C' has $\text{size}(C) = m$

and is therefore losing by the induction hypothesis. ■

Let $C = (h_1, \ldots, h_k; e; T)$ and $C' = (h_1', \ldots, h_k'; e'; T)$ be two configurations. We say C' *dominates* C if $e' \leq e$ and there is a permutation π such that $h_i' \geq h_{\pi(i)}$ for each i. We write $C' \geq C$.

Lemma 13 and Lemmas 15–18 demonstrate various pairs of configurations C and C' such that if C is winning then C' is also winning. The proofs of these lemmas are examples of what is known as a strategy stealing argument. These proofs are by induction on size(C). For the inductive step, we assume the lemma holds when size(C) = m and consider configurations C_0 and C_0' that satisfy the conditions of the lemma and have size(C_0) = $m+1$. We construct configurations $C_1, \ldots, C_i$ and $C_1', \ldots, C_j'$ as shown in Figure 1. Generally i and j will be 1 or 2.

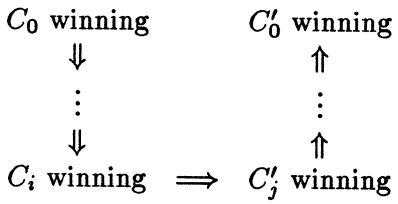

Figure 1: The strategy stealing argument.

The configurations $C_1, \ldots$ are constructed by playing the C_0 stick game. We assume the team never makes a move that would take a winning configuration to a losing one, and we specify the adversary moves. Since an adversary move cannot take a winning configuration to a losing one, it follows that if C_0 is winning, then every C_ℓ is winning. Similarly, the configurations $C_1', \ldots$ are constructed by playing the C_0' stick game. We assume the adversary never makes a move on a losing configuration that results in a winning configuration, and we specify the team moves. It follows that if C_0' is losing, then every C_ℓ' is losing, or conversely, if any C_ℓ' is winning, then C_0' is winning.

To avoid confusion, we call the C_0 stick game G, and we refer to the G team and the G adversary. Similarly, we have the G' game, team and adversary. We will choose the G adversary moves and the G' team moves in such a way that we end up with a C_i and C_j' that either satisfy the induction hypothesis or such that some previous lemma says that if C_i is winning then C_j' is. The main way we do this is *mirroring*. The G adversary mirrors the G' adversary by matching with the same pile the G' adversary previously matched with. The G' team mirrors the G team by choosing the same pile previously chosen by the G team.

Lemma 13 (Domination Lemma) *If C is winning and $C' \geq C$, then C' is winning.*

Proof: By the definition of domination, C and C' have the same number of piles, say k. If size(C) ≤ 1, then there is at most one team pile in C and hence also in C'. Thus the team has won in C', so C' is winning and we are done.

Inductively assume that the lemma holds when size(C) = m. We consider configurations $C_0 = (h_1, \ldots, h_k; e; T)$ and $C_0' = (h_1', \ldots, h_k'; e'; T)$ such that C_0 is winning, size(C_0) = $m+1$ and $C_0' \geq C_0$.

If $k = 1$, then C'_0 is winning and we are done. Otherwise, $k \geq 2$, and so the G team has not won in C_0, and must choose a pile, say P_i. We let C_1 be the resulting configuration $(h_1, \ldots, h_k; e; i)$. In game G', we consider the corresponding team choice $P_{\pi^{-1}(i)}$, which must be possible because $h'_{\pi^{-1}(i)} \geq h_i$, by the definition of domination. Hence, we choose $C'_1 = (h'_1, \ldots, h'_k; e'; \pi^{-1}(i))$. The G' adversary now makes a move, resulting in configuration C'_2. We construct configuration C_2 by having the G adversary mirror the G' adversary, if possible. If the G' adversary matches with E, then the G adversary can match with E, since $e \geq e'$, by the definition of domination. Then $C'_2 \geq C_2$, and so by the induction hypothesis, C'_2 is winning, and hence C'_0 is winning.

Otherwise the G' adversary matches with some P_j. Because C_0 is winning and $k \geq 2$, Lemma 12 implies that $|P_\ell| > 0$ for every ℓ. Hence the G adversary can match with $P_{\pi(j)}$, and again $C'_2 \geq C_2$, so by the induction hypothesis, C'_2 and hence C'_0 are winning. ■

A special case of the Domination Lemma is that if two configurations differ only by a permutation of the pile sizes, they are either both winning or both losing. We sometimes say two such configurations are *equal up to permutation.*

We call a team pile of size 1 a *singleton.* Lemma 14 says that whenever a singleton is feasible in a given configuration, the configuration is winning.

Lemma 14 *Let $C = (h_1, \ldots, h_k; e; T)$. If for some i, P_i is feasible and $h_i = 1$, then C is a winning configuration.*

Proof: We prove this by induction on k. Without loss of generality, assume $i = 1$. Then P_1 is feasible and $h_1 = 1$. This implies that for every $j \geq 2$, $h_j > 1$ and that $e = 0$. If P_1 is the only team pile then C is winning.

Inductively assume that the lemma holds for all configurations C with k team piles. Consider $C_0 = (1, h_2, \ldots, h_{k+1}; 0; T)$. To show that C_0 is winning, we show that there is a team move that results in a winning configuration. If the team chooses P_1, the resulting configuration is $C_1 = (1, h_2, \ldots, h_{k+1}; 0; 1)$. Because $e = 0$, the adversary must match with some P_j. Without loss of generality, let $j = 2$. The new configuration is $C_2 = (h_2 - 1, h_3, \ldots, h_{k+1}; 0; T)$. C_2 dominates $(1, h_3, \ldots, h_{k+1}; 0; T)$, which is winning by the induction hypothesis. Hence by domination, C_2 is winning, and so C_0 is winning. ■

Lemma 15 *Let $C = (h_1, \ldots, h_k, x; e; T)$ and $C' = (h_1, \ldots, h_k, x - 1; e - 1; T)$. If C is winning, $e \geq 1$, and each h_i satisfies $h_i \geq x$ or $h_i = 1$, then C' is winning.*

Proof: If size$(C) \leq 2$, then C' is a win, since size(C') = size$(C) - 1 \leq 1$.

Inductively assume that the lemma holds when size$(C) = m$. Consider C_0 and C'_0 that satisfy the conditions of the lemma such that size$(C_0) = m + 1 > 2$. As shown in Figure 1, we construct configurations C_i and C'_i by playing the C_0 stick game and the C'_0 stick game.

If $k = 0$, then C' is winning. Otherwise $k \geq 1$, and so the G team has not won in C_0, and must choose a pile, say P_i. The G team won't choose a pile with $|P_i| = 1$, because by assumption, $e \geq 1$, so the G adversary can match with E. In this case C_2 is losing by Lemma 12. Hence G chooses P_i such that $|P_i| \geq x$.

If $|P_i| = x$, we have the G' adversary match with Eve. Then up to permutation, the resulting configuration $C_2 = (h_1, \ldots, h_k, x - 1; e - 1; T) = C'_0$, so C'_0 is winning. Otherwise

$|P_i| > x$, and without loss of generality, $i = 1$. Then C_1 is $(h_1, \ldots, h_k, x; e; 1)$. We mirror the G team move to G', so $C_1' = (h_1, \ldots, h_k, x-1; e-1; 1)$.

We then mirror the G' adversary move to G. We consider three types of adversary moves separately. In each case we show that C_2' is winning. Hence, C_0' is winning, as desired.

- G' adversary matches with E. Then

$$\begin{aligned} C_2' &= (h_1 - 1, h_2, \ldots, h_k, x-1; e-2; T) \\ C_2 &= (h_1 - 1, h_2, \ldots, h_k, x; e-1; T) \end{aligned}$$

  Furthermore, $h_1 > x$ implies $h_1 - 1 \geq x$, so by the induction hypothesis, C_2' is winning.

- G' adversary matches with P_j such that $h_j \neq x - 1$. Without loss of generality $j = 2$, so

$$\begin{aligned} C_2' &= (\max(h_1, h_2) - 1, h_3, \ldots, h_k, x-1; e-1; T) \\ C_2 &= (\max(h_1, h_2) - 1, h_3, \ldots, h_k, x; e; T) \end{aligned}$$

  Since $h_1 > x$, we have that $\max(h_1, h_2) - 1 \geq h_1 - 1 \geq x$. By the induction hypothesis, C_2' is winning.

- G' adversary matches with P_j such that $|P_j| = x - 1$. Without loss of generality, $j = k + 1$, and we have the G adversary match with P_{k+1}. We have $h_1 > x$, so

$$\begin{aligned} C_2' &= (h_1 - 1, h_2, \ldots, h_k; e-1; T) \\ C_2 &= (h_1 - 1, h_2, \ldots, h_k; e; T) \end{aligned}$$

  Then $C_2' \geq C_2$. Hence by domination, C_2' is winning. ■

Lemma 16 *Let $C = (h_1, \ldots, h_k; e; T)$ and $C' = (h_1, \ldots, h_k, x; e-1; T)$. If C is winning and $x \geq e \geq 1$, then C' is winning.*

Proof: If size$(C) = 1$, then $C' = (h_1, x; e-1; T)$. Because C is winning and has at least two piles, Lemma 12 implies that $h_1 \geq 1$. Hence by Theorem 1 and Lemma 11 (Stick Game Lemma), C' is a winning configuration.

Inductively assume that the lemma holds if size$(C) = m$. Consider C_0 and C_0' that satisfy the conditions of the lemma such that size$(C_0) = m + 1$. If $k < 2$, then by the above argument C_0' is a win and we are done. Otherwise there are at least 2 team piles in C_0, and so the G team has not won, and must choose a pile, say P_1. We mirror the G team move to G'. In each case we show C_2' is winning.

- G' adversary matches with P_j such that $|P_j| \neq x$. Then we mirror the G' adversary move to G. Without loss of generality, $j = 2$, so

$$\begin{aligned} C_2' &= (\max(h_1, h_2) - 1, h_3, \ldots, h_k, x; e-1; T) \\ C_2 &= (\max(h_1, h_2) - 1, h_3, \ldots, h_k; e; T) \end{aligned}$$

  By the induction hypothesis, C_2' is winning.

- G' adversary matches with P_j such that $|P_j| = x$. Without loss of generality, $j = k+1$. In this case, we have the G adversary match with E. Then

$$\begin{aligned} C_2' &= (\max(x, h_1) - 1, h_2, \ldots, h_k; e-1; T) \\ C_2 &= (h_1 - 1, h_2, \ldots, h_k; e-1; T) \end{aligned}$$

  Hence $C_2' \geq C_2$, so C_2' is winning.

Note that it is not possible for the G' adversary to match with E, since $|E| = 0$. ■

Lemma 17 *Let $C = (h_1, \ldots, h_k, x, x; e; T)$ and $C' = (h_1, \ldots, h_k, x-1, x-1; e-1; T)$. If C is winning, $e \geq 1$, $x \geq \max(e, (e+3)/2)$ and each h_i satisfies $h \geq x$ or $h_i = 1$, then C' is winning.*

Proof: If size$(C) = 2$, then $C' = (x-1, x-1; 0; T)$, which by Theorem 1 and Lemma 11 (Stick Game Lemma), is winning.

Inductively assume that the lemma holds if size$(C_0) = m$. Consider C_0 and C_0' that satisfy the conditions of the lemma such that size$(C_0) = m+1$. There are at least 2 team piles in C_0, so the G team has not won, and must choose a pile, say P_i. It is not possible that the G team chooses P_i such that $|P_i| = 1$, because then the G adversary could match with E, and by Lemma 12, the resulting C_2 is losing.

1. G team chooses P_i such that $|P_i| = x$: In this case the G adversary matches with another pile P_j with $h_j = x$. Without loss of generality, $i = k+1$ and $j = k+2$. Then $C_2 = (h_1, \ldots, h_k, x-1; e; T)$, so C_2 and C_0' satisfy the conditions of Lemma 16, and hence C_0' is winning.

2. G team chooses P_i such that $|P_i| > x$: We mirror the G team move to G', and we mirror the G' adversary match to G. Without loss of generality, let $i = 1$.

   - G' adversary matches with P_j such that $|P_j| \neq x-1$. Without loss of generality, $j = 2$, so

$$\begin{aligned} C_2' &= (\max(h_1, h_2) - 1, h_3, \ldots, h_k, x-1, x-1; e-1; T) \\ C_2 &= (\max(h_1, h_2) - 1, h_3, \ldots, h_k, x, x; e; T) \end{aligned}$$

     By the induction hypothesis, C_2' is winning.

   - G' adversary matches with P_j such that $|P_j| = x-1$. Then without loss of generality, $j = k+1$, and we have the G adversary match with P_{k+1}. Since $|P_i| > x$,

$$\begin{aligned} C_2' &= (h_1 - 1, h_2, \ldots, h_k, x-1; e-1; T) \\ C_2 &= (h_1 - 1, h_2, \ldots, h_k, x; e; T) \end{aligned}$$

     Then C_2 and C_2' satisfy the conditions of Lemma 15, so C_2' is winning. ■

Lemma 18 *Let $C = (h_1, \ldots, h_k, x, y; e; T)$ and $C' = (h_1, \ldots, h_k, y'; e; T)$. If C is winning, $y' + 1 \geq y \geq x$, $e \geq 0$, each h_i satisfies $h_i \geq x$ or $h_i = 1$, and either*

- $(x, y, y') \neq (2, 2, 1)$
- *there is no feasible singleton in* C,

then C' *is winning.*

Proof: If size$(C) \leq 2$, then C' is winning.

Inductively assume that the lemma holds if size$(C_0) = m$. Consider C_0 and C_0' that satisfy the conditions of the lemma such that size$(C_0) = m + 1 > 2$. Because there are at least 2 team piles in C_0, Lemma 12 implies each pile must be of size at least 1. The G team has not won, and must choose a pile P_i.

1. G team chooses P_i such that $|P_i| = x$ (or y): In this case the G adversary matches with a pile P_j with $h_j = y$ (or x). Without loss of generality, $i = k + 1$ and $j = k + 2$. Then $C_2 = (h_1, \ldots, h_k, y - 1; e; T)$. Hence $C_2 \geq C_0'$, so C_0' is winning.

2. G team chooses P_i such that $|P_i| > x$: Without loss of generality, let $i = 1$. We mirror the G team move to G'.

   - G' adversary matches with E. If $(x, y, y') \neq (2, 2, 1)$, or if $(x, y, y') = (2, 2, 1)$ and $e \neq 1$, or if $(x, y, y') = (2, 2, 1)$, $e = 1$ and there is not exactly one singleton in C_0, we mirror the match with E back to G. Then

     $$\begin{aligned} C_2' &= (h_1 - 1, h_2, \ldots, h_k, y'; e - 1; T) \\ C_2 &= (h_1 - 1, h_2, \ldots, h_k, x, y; e - 1; T) \end{aligned}$$

     By induction, C_2' is winning. Otherwise, $(x, y, y') = (2, 2, 1)$, $e = 1$ and there is exactly one singleton, say P_k, in C_0. In this case, instead of mirroring the match with E, we have the G adversary match with P_k. Then

     $$\begin{aligned} C_2' &= (h_1 - 1, h_2, \ldots, h_{k-1}, 1, 1; 0; T) \\ C_2 &= (h_1 - 1, h_2, \ldots, h_{k-1}, 2, 2; 1; T) \end{aligned}$$

     Then $h_1 - 1 \geq x$, so C_2 and C_2' satisfy Lemma 17 and hence C_2' is winning.
   - G' adversary matches with P_j, where $|P_j| \neq y'$. In this case, we mirror the G' adversary move to G. Without loss of generality $j = 2$, so

     $$\begin{aligned} C_2' &= (\max(h_1, h_2) - 1, h_3, \ldots, h_k, y'; e; T) \\ C_2 &= (\max(h_1, h_2) - 1, h_3, \ldots, h_k, x, y; e; T) \end{aligned}$$

     If $(x, y, y') = (2, 2, 1)$ then we haven't changed e or the number of singletons from C_0 to C_2. Furthermore, $h_1 > x$ implies $\max(h_1, h_2) - 1 \geq h_1 - 1 \geq x$. Hence by induction, C_2' is winning.
   - G' adversary matches with P_j such that $|P_j| = y'$. Without loss of generality, $j = k + 1$. Then we have the G adversary match with P_{k+2}, then

     $$\begin{aligned} C_2' &= (h_2, \ldots, h_k, \max(h_1, y') - 1; e; T) \\ C_2 &= (h_2, \ldots, h_k, x, \max(h_1, y) - 1; e; T) \end{aligned}$$

     We have $h_1 > x$, so $\max(h_1, y') \geq \max(h_1, y) - 1 \geq x$. Hence by induction, C_2' is winning.

3. G team chooses P_i such that $|P_i| = 1$: Without loss of generality, let $i = 1$. Suppose $(x, y, y') = (2, 2, 1)$. Then by assumption either $e > 0$ or there is a $j \neq i$ such that $h_j = 1$. In either case the G adversary can match to get a resulting configuration that has a size 0 pile, and has at least two team piles, since C_0 has at least three team piles, a contradiction to Lemma 12. Hence $(x, y, y') \neq (2, 2, 1)$. Similarly, there can not be any singleton other than P_1 in C_0. This implies that P_1 is feasible in C_0', so by Lemma 14, C_0' is winning. ■

Lemma 19 says that for teams of size two, Theorem 1 is exact.

Lemma 19 *If $C = (h_1, h_2; e; T)$, then C is winning if and only if $h_1 + h_2 \geq 2 + e$.*

Proof: From Theorem 1 and Lemma 11 (Stick Game Lemma), we know that if $h_1 + h_2 \geq 2 + e$. To show the converse, suppose that $h_1 + h_2 < 2 + e$. In order for the adversary to force the team to lose, the adversary can never match with E, for this would immediately result in a win for the team. This can be done until $|E| = 0$. Every match with E results in exactly one stick being removed from the team piles, so after e matches with E, the total number of sticks in the team piles is $h_1 + h_2 - e$. Since $h_1 + h_2 < 2 + e$, we have $h_1 + h_2 - e < 2$. Hence after e matches with E, at least one team pile is of size 0. Since there are still two piles, Lemma 12 implies that the configuration is losing. ■

Thus SFP is optimal for configurations with only two team players, since SFP works whenever the configuration is winning. We can now show the optimality of the SFP strategy.

Theorem 20 *The SFP strategy is an optimal team strategy for the stick game.*

Proof: By Lemma 19, it remains only to show that SFP is optimal for configurations with at least 3 team piles. Suppose $C_0 = (h_1, \ldots, h_k; e; T)$ is winning, but is not yet a win, such that $k \geq 3$. We show that $C_1' = (h_1, \ldots, h_k; e; i)$, where i is chosen by SFP, is winning. Because C_0 is winning, there is some strategy OPT for the team such that the resulting configuration $C_1' = (h_1, \ldots, h_k; e; j)$, where j is chosen by OPT, is winning. Also, since $k \geq 3$, Lemma 12 implies $h_l \geq 1$ for every l. If $x_j = x_i$, then up to permutation $C_1' = C_1$, so C_1' is winning, and we are done. Hence we need only consider $x_j \neq x_i$. As before we distinguish the C_1 stick game and the C_1' stick game by calling them the G game and the G' game.

Let x_i and x_j be $|P_i|$ and $|P_j|$, respectively. Suppose $x_j = 1$. Then then P_j is not feasible, or it would have been chosen by SFP. Hence either $|E| \geq 1$ or there is another pile P_m such that $|P_m| = 1$. In either case, the G adversary can match to get a resulting configuration C_2 with at least two team piles, one of which is of size 0. Thus by Lemma 12, C_2 is losing, a contradiction. Hence $x_j > 1$.

Then because x_i is chosen according to the *smallest* feasible pile rule, it must be the case that $x_j > x_i$. We consider possible G' adversary moves from C_1', and show that for every possible G' adversary move from C_1', the resulting C_2' is winning, and hence C_1' is winning.

If the G' adversary matches with E, then the resulting configuration $C_2' = (h_1, \ldots, h_i - 1, \ldots, h_k; e - 1; T)$. Hence C_0 and C_2' satisfy the conditions of Lemma 15, so C_2' is winning. Otherwise, the L adversary matches with some P_a, where $a \neq i$. Let $y = |P_a|$. Without

loss of generality, say $i = 1$, $j = 2$ and $a = 3$. Then the resulting configuration $C_2' = (\max(x_i, y) - 1, x_j, h_4, \ldots, h_k; e; T)$.

We consider separately possibilities for y, and in each case consider a possible match for the G adversary. Each match is with either P_i or P_a, so all resulting configurations C_2 are of the form $(p, q, h_4, \ldots; e; T)$, where only p and q vary. C_2' is also of this form. We abbreviate this $(p, q, \ldots)$. For example $C_2' = (\max(x_i, y) - 1, x_j, \ldots)$. In each case we show C_2' is winning, so C_1' is winning, as desired.

- $x_j < y$: G adversary matches with P_a. Then $C_2 = (x_i, \max(x_j, y) - 1, \ldots) = (x_i, y - 1, \ldots)$. Hence $C_2' \geq C_2$ (flip first two piles). By domination, C_2' is winning.
- $x_j = y$: G adversary matches with P_i. Then $C_2 = (\max(x_i, x_j) - 1, y, \ldots) = (x_i - 1, x_j, \ldots) = C_2$, so C_2' is winning.
- $x_i < y < x_j$: G adversary matches with P_a. Then $C_2 = (x_i, y - 1, \ldots) \leq C_2'$ (flip first two piles), so C_2' is winning.
- $y < x_i$: G adversary matches with P_i. Then $C_2 = (x_i - 1, x_j, \ldots) = C_2'$, so C_2' is winning.
- $x_i = y$: Then $C_2' = (x_i - 1, x_j, \ldots)$ and $C_0 = (x_i, x_i, x_j, \ldots)$. If $x_i = 1$, then because there are two piles of size 1, P_i is not feasible. Since P_i was chosen by SFP, it must be the case that no pile is feasible, and hence all piles are of size 1. Thus $x_j = 1$, and so $x_j = x_i$, a case previously handled. Otherwise, if $x_i = 2$, then because P_i was chosen by SFP, C_0 has no feasible pile of size 1. Hence C_0 and C_2' satisfy Lemma 18, so C_2' is winning. Otherwise, $x_i > 2$, and by Lemma 18, C_2' is winning. ■

7.4 Optimality of the SFP Key Set Protocol

We are now able to show the optimality of the SFP proposing rule for the key set protocol.

Theorem 21 *The SFP key set protocol is optimal for the class of key set protocols.*

Proof: Suppose that the $\mathcal{R}$ key set protocol works for $\xi = (h_1, \ldots, h_k; e)$. Then by Lemma 11 (Stick Game Lemma), $C = (h_1, \ldots, h_k; e; T)$ is a winning configuration. Then by Theorem 20, if the team uses the SFP strategy, it wins the C stick game. Finally, by Lemma 11, the SFP key set protocol works for ξ. ■

Theorem 21 indicates that changing the proposing rule is not a sufficient modification to the key set protocol to close the gap described at the beginning of the section. However, there are other possible modifications to the key set protocol to consider. For example, it is possible to allow the players to communicate in order to choose the proposer. This also does not close the gap, for we can show that the SFP key set protocol is optimal for the larger class of protocols this gives rise to. However, the optimality may fail if the proposed key set is allowed to be chosen non-randomly.

In the key set protocols described here, every time a key set is found, one of the team players discards all the cards in her hand and drops out of the protocol, except to wait to hear the secret bit. We do this in order to avoid getting more than one key set between any

two players. It would be possible to consider key set protocols in which a team player only drops out when a team player in the same connected component of the key set graph is chosen to propose a key set. We suspect that this does not give the team additional power, and conjecture that Theorem 21 holds for this larger class of protocols.

Another possible modification to the key set protocol is to allow team players to discard only the key set cards and risk getting multiple key sets between two team players. It is an open question whether multiple key sets can be used (for example to "send" some of the cards in a player's hand to another player) to achieve 1-bit secret key exchange where no key set protocol of the class described in this paper succeeds.

8 Concluding Remarks

We have shown here some conditions on the signature of the deal that allow secret key exchange to take place and some conditions under which secret key exchange is not possible. However, there is a large gap. There are many signatures for which we can neither give a secret key exchange protocol nor demonstrate the nonexistence of such a protocol.

As a future direction for this work, we intend to look at the concept of shared secret information between a team. We would like to develop a theory of shared secret information which can be applied to arbitrary correlated random variables. Specifically, can we quantify how many bits of shared secret information a deal contains for the team? How can we use this information to develop better protocols and tighter lower bounds on the signatures for which secret key exchange is possible? More generally, what other mechanisms besides deals from a common deck of cards give correlated random variables that can be used for secret key exchange?

Deals of cards have a small amount of initial information. However, deals of cards appear somewhat inefficient for secret key exchange, in that the number of secret bits the team can obtain is small in comparison to the number of cards they are dealt. Michael Rabin [8] suggests a protocol that uses private correlated random variables to solve another classical security problem, authentication. His method requires random variables that appear to contain more initial information than a deal of cards, but also appear to contain more shared secret information. We would like to use the theory of shared secret information suggested above to quantify the ratio of initial information to shared secret information, and to investigate upper and lower bounds on this ratio for secret key exchange protocols.

9 Acknowledgements

We thank Michael Merritt for his contribution to the proof of Lemma 3. We thank Peter Winkler for many helpful comments. We thank Nick Reingold for countless discussions, and for suggesting a simpler proof of Lemma 7.

References

[1] D. Beaver, S. Haber, and P. Winkler. On the Isolation of a Common Secret, preprint, Bellcore, 1991.

[2] R. Beigel, 1991. (Private communication.)

[3] E. R. Berlekamp, J. H. Conway, and R. K. Guy. *Winning Ways*, Volume I, Academic Press Inc., London, 1982.

[4] W. Diffie and M. E. Hellman. New Directions in Cryptography. *IEEE Trans. Inform. Theory, IT-22, Vol. 6,* Nov. 1976, pp. 644–654.

[5] M. J. Fischer, M. S. Paterson, and C. Rackoff. Secret Bit Transmission Using a Random Deal of Cards, *Distributed Computing and Cryptography*, American Mathematical Society, 1991, pp. 173–181.

[6] M. J. Fischer, P. Winkler, and R. N. Wright. June, 1990. (Private communication.)

[7] J. Flint, Cheating by Degrees, *The Times Saturday Review* , May 9, 1981.

[8] M. Rabin. Cryptography Without Secrets. Presented at *DIMACS 1990 Workshop on Cryptography*, Princeton, NJ. October 1–4, 1990.

[9] P. Winkler, Cryptologic Techniques in Bidding and Defense, Parts I, II, III, and IV, *Bridge Magazine*, April – July, 1981.

[10] P. Winkler, My Night at the Cryppie Club, *Bridge Magazine*, August 1981, pp. 60–63.

[11] P. Winkler, The Advent of Cryptology in the Game of Bridge, *Cryptologia*, Vol. 7, No. 4, October 1983, pp. 327–332.