1

Optimal Threshold Padlock Systems

Jannik Dreier [a,*], Jean-Guillaume Dumas [b], Pascal Lafourcade [c] and Léo Robert [c]

[a] *Univ. de Lorraine, CNRS, Inria, LORIA, F-54000 Nancy, France*
E-mail: Jannik.Dreier@loria.fr
[b] *Univ. Grenoble Alpes, Laboratoire Jean Kuntzmann, CNRS UMR 5224, 38058 Grenoble, France*
E-mail: Jean-Guillaume.Dumas@univ-grenoble-alpes.fr
[c] *Univ. Clermont Auvergne, LIMOS, CNRS UMR 6158, 63178 Aubière, France*
E-mails: Pascal.Lafourcade@uca.fr, Leo.Robert@uca.fr

Abstract. In 1968, Liu described the problem of securing documents in a shared secret project. In an example, at least six out of eleven participating scientists need to be present to open the lock securing the secret documents. Shamir proposed a mathematical solution to this physical problem in 1979, by designing an efficient *k*-out-of-*n* secret sharing scheme based on Lagrange's interpolation. Liu and Shamir also claimed that the minimal solution using physical locks is clearly impractical and exponential in the number of participants. In this paper we relax some implicit assumptions in their claim and propose an optimal physical solution to the problem of Liu that uses physical padlocks, but the number of padlocks is not greater than the number of participants. Then, we show that no device can do better for *k*-out-of-*n* threshold padlock systems as soon as $k \geqslant \sqrt{2n}$, which holds true in particular for Liu's example. More generally, we derive bounds required to implement any threshold system and prove a lower bound of $O(\log(n))$ padlocks for any threshold larger than 2. For instance we propose an optimal scheme reaching that bound for 2-out-of-*n* threshold systems and requiring less than $2\log_2(n)$ padlocks. We also discuss more complex access structures, a wrapping technique, and other sublinear realizations like an algorithm to generate 3-out-of-*n* systems with $2.5\sqrt{n}$ padlocks. Finally we give an algorithm building *k*-out-of-*n* threshold padlock systems with only $O\left(\log(n)^{k-1}\right)$ padlocks. Apart from the physical world, our results also show that it is possible to implement secret sharing over small fields.

Keywords: Threshold Cryptography, Physical Secret Sharing, Block Designs, Packings

1. Introduction

In 1979, in his paper on secret sharing [1], Shamir presented the following threshold problem introduced by Liu in [2, Example 1-11]: *Eleven scientists are working on a secret project. They wish to lock up the documents in a cabinet so that the cabinet can be opened if and only if six or more of the scientists are present. What is the smallest number of locks needed? What is the smallest number of keys to the locks each scientist must carry?* Liu and Shamir answered this *physical* problem using mathematics as follows: *It is not hard to show that the minimal solution uses 462 locks and 252 keys per scientist. These numbers are clearly impractical, and they become exponentially worse when the number of scientists increases.* This is why Shamir proposed to use polynomial and Lagrange's interpolation to solve Liu's question. His clever idea is to hide the secret in the constant term of a polynomial of degree $k-1$. Then he distributes one point of the chosen polynomial to each of the *n* participants. As soon as *k* participants share their points, they can recover the secret using Lagrange's interpolation and algorithms in $O(n \log^2 n)$ operations [3, 4]. A few years

---

*Corresponding author. E-mail: Jannik.Dreier@loria.fr.

later, verifiable secret sharing was introduced by Chor et al. in [5] and improved in [6]. The idea is to offer the possibility to verify if the points are valid.

We show that Liu's problem is solvable using far less locks. Liu and Shamir claim stems from the restriction that there should be a lock for each combination of 6 scientists, $462 = \binom{11}{6}$, and that every scientist needs the keys for every combination of scientists that includes him. This is $252 = \binom{10}{5}$ keys. Liu-Shamir's minimality result thus assumes that the only physical arrangements of locks that allow threshold systems are those where the opening of any lock opens the cabinet.

Contributions:

(1) As a warm up, we relax Liu and Shamir's assumption and design a *physical k-out-of-n threshold padlock system*. We have build a prototype of this physical device. Our system only requires one padlock and one key per participant, which is practical, when compared to the previous exponential solution.

(2) Then, we establish *lower bounds* on the number of padlocks necessary for any abstract threshold system.

Specifically, we show that for a 2-out-of-*n* configuration, less than $2\lceil \log_2(n) \rceil$ padlocks are sufficient, provided that keys can be duplicated. In fact, there is an optimal solution for this type of configuration, with $\mathcal{O}(\log(n))$ padlocks, and we also show that this optimum can be realized, using our physical system as one building block.

Differently, for *k*-out-of-*n* configurations with $k \geqslant 3$, it is more complicated to solve the problem with fewer than *n* padlocks. We first prove that this is impossible for $k \geqslant \sqrt{2n}$ and thus that our physical device is optimal in these cases. For instance, this answers Liu's question: the minimal number of padlocks for a 6-out-of-11 configuration is 11, as $6 > \sqrt{22}$. Our system with 11 padlocks and only 1 key per participant, is thus optimal in this case. We are then nonetheless able to give algorithms building systems for $k = 3$ with only about $2.5\sqrt{n}$ padlocks and each participant owns 3 keys. These realizations use more complex access structures and associated algorithmic building blocks, that we provide.

(3) We discuss more *complex access structures*, which include for instance ensuring that Alice and Bob can open the lock with any other third participant, but not together. Another possibility is for instance that Alice is highly ranked and can open the padlock by herself but that any others need to be at least two. For this we develop a tentative *padlock algebra* for logic gates and give associated algorithms. The idea is to combine threshold cryptography and secret sharing with the theory of block designs, packings and Sperner families.

(4) Finally, we propose a *recursive algorithm* to build larger systems, that requires only a logarithmic number of padlocks. Asymptotically, our algorithm requires only $O\big(\log(n)^{k-1}\big)$ padlocks to realize a *k*-out-of-*n* threshold padlock system.

(5) Lastly, we also show that our physical results do apply to the *numerical world* by linking the number of padlocks to the size of the finite field used for secret sharing.

Outline: In section 2, we review existing threshold mechanisms that use physical padlocks, or visual cryptography, which not perfectly answer Liu's problem. In section 3, we describe our novel physical *k*-out-of-*n* threshold padlock system device. In section 4, we derive generic bounds on the number of padlocks required to realize a given threshold configuration. We also show the optimality of our device for 2-out-of-*n* systems. Then, in section 5, we discuss more complex access structures. We provide for instance solutions on logic formulae in sections 5.1 and 5.2, and in section 5.3 we introduce the use of a sealed wire. Further lower bounds, including the optimality of our solution for sufficiently large *k*,

together with smaller realizations, with strictly less than *n* padlocks, are given in section 6. Our recursive construction is then given in section 7 and the link with the numerical aspects in section 8.

2. Related Work

Threshold cryptography in general received a lot of attention recently, since on March 1, 2019 the Computer Security Division (CSD) at the National Institute of Standards and Technology (NIST) published the final version of NISTIR 8214, "*Threshold Schemes for Cryptographic Primitives*" [7]. This reports explicitly also mentions physical threshold solutions (page 10, line 55): "*While we focus on secure implementations of cryptographic primitives, the actual threshold techniques may also include non-cryptographic techniques.*" We present existing physical solutions for threshold cryptography, while a survey of cryptographic threshold schemes by Y. Desmedt can be found in [8]. We distinguish two classes of solutions: the first one uses physical keys and padlocks; the second one uses visual cryptography, as introduced by M. Naor and A. Shamir in 1994 [9].

2.1. State of the art, using Padlocks

A 1-out-of-1 padlock is just one simple physical padlock. There are many systems for 1-out-of-*n* padlocks, both home made and commercial products. There also exist commercial solutions for *n*-out-of-*n* padlocks, which are used by for example by electricians to secure an electrical circuit as explained next.

2.1.1. 1-*out-of-n locks*

In fig. 1, left, a 1-out-of-2 padlocks is done simply with two physical padlocks. This approach can be generalized to 1-out-of-*n* as in fig. 1, right, and is called a *daisy chain*. We notice that the bottom left yellow padlock was badly placed, and it is useless. In this case the owner of this padlock cannot open the door. We call this the *daisy chain attack*. For example in fig. 1, if the owner of the bottom padlock opens it and then locks it upper in the chain, then he excludes all the owners of these padlocks, as they cannot open the door any more[1].

Fig. 1. Physical 1-out-of-*n* padlocks, forming a daisy chain. **Left:** The simplest daisy chain with two padlocks. **Right:** A longer daisy chain with one useless padlock.

---

[1]A deliberate attack adding an additional chain and padlock to the gate, or even welding padlocks together, is always possible, and out of scope here: we aim to protect against attacks that could be "excused" with a wrong use of the system.

In fig. 2, we can see two different mechanisms that perform 1-out-of-6 padlocks to open the gate of a field. The first one has six padlocks that block the trigger. As soon as one padlock is opened a latch is removed and then the door can be opened. It is the natural extension of the solution of fig. 1 that avoids the daisy chain attacks. Next, the second picture of fig. 2, shows a different solution also implementing a 1-out-of-6 padlock, and which is also resistant to the daisy chain attack. In this system, as soon as one padlock is removed, it is possible to turn the circle and then to pass the stick in the corresponding hole in order to open the door.

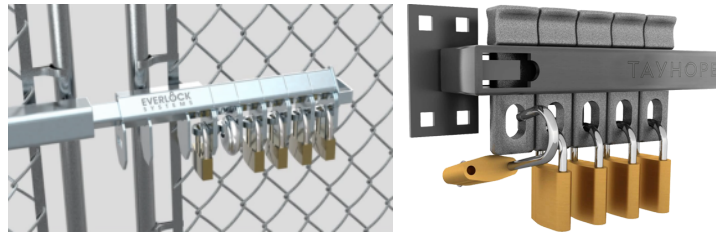Fig. 2. Two ad-hoc physical 1-out-of-6 padlocks.

Fig. 3. Physical 1-out-of-5 padlocks, first by Everlock System, then by Tayhope multi-locking system.

There are also commercial products for 1-out-of-n padlocks. The first picture of fig. 3 shows a commercial product designed by Everlock Systems: the model SLX2 [10]. The second picture of fig. 3 shows a commercial product sold by Tayhope Multi-Locking Systems [11]. Everlock Systems has multiple patents on their designs [12–15] and their solution is close to the mechanism proposed on the left side of fig. 2. Differently, Tayhope mechanism allows the owner of a padlock to remove the metallic stick which enables the opening of the door, by pushing all the padlocks on one side.

Now, if one is interested in reducing the number of padlocks, one can realize a 1-out-of-n threshold system with a single lock: duplicate the key of one padlock n times and distribute the key to all the participants. The obtained system has not all the physical properties of the daisy chain or the systems of fig. 3 (for instance the latter does not need a trusted third party to setup the chain or to duplicate the keys), but is probably more economical. Overall, we have the following possibilities for 1-out-of-n systems:

- A single padlock with n duplicated keys: probably most economical;

- A daisy chain: if keys cannot be duplicated;
- Systems like those of fig. 3: they do not require a trusted third party for the setup, as each participant can bring their own lock and key(s).

2.1.2. *n-out-of-n locks*

Finally, there are physical *n*-out-of-*n* mechanisms using padlocks that are used for example for operations on high-voltage circuits and transformers. Two examples of 6-out-of-6 padlocks are given in fig. 4. The idea is that nobody should be able to turn on the electricity while someone is still working on the high-voltage transformer. To achieve this, each technician places a padlock on the main switch before entering the danger zone. This ensures that all technicians have to leave the danger zone before electricity can be restored. The example can easily be extended to a *n*-out-of-*n* system.

Fig. 4. Physical 6-out-of-6 padlocks, by Seton (models SLECO and MANM8).

2.2. *Using Visual Cryptography*

In 1994, M. Naor and A. Shamir proposed the *visual cryptography* [9, 16] for black and white images. This was improved in [17] for gray images and in [18] for color images.

The idea is to split a secret into two images printed on transparent paper in a way such that their superposition makes the secret appear. An example is given in fig. 5.
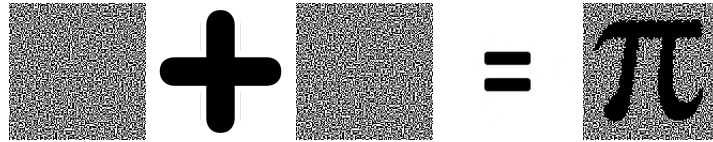

Fig. 5. Example of visual cryptography, superposing the images let the symbol π appear.

For color images, security cannot be perfectly achieved for more than 3 colors [19]. In [20], the authors proposed a generalization of the approach to *k*-out-of-*n* images. This can be used as a first physical answer to Liu's problem. This solution is not really practical since it needs a computer to compute the different images. Moreover in [21], the authors show that it is possible to cheat in visual cryptography by introducing fake shares that change the result. This clearly shows that this solution is not verifiable, which requires the ability to check that shares are valid.

3. A Novel Physical *k* out *n* Weighted Threshold Lock

As a natural extension of 1-out-of-*n* systems, we design a *k*-out-of-*n* physical threshold lock that uses *n* padlocks and works as follows. Each padlock secures one block, with a latch, attached to a sliding

bar, which is limited in its sliding movement by the blocks. If sufficiently many blocks are removed, the sliding bar can be moved far enough to open the barrier. In Algorithm 1, we describe our solution in a generic way.

We also have built a wooden prototype that can be configured for different cases, see fig. 6 for a 2-out-of-3 configuration.
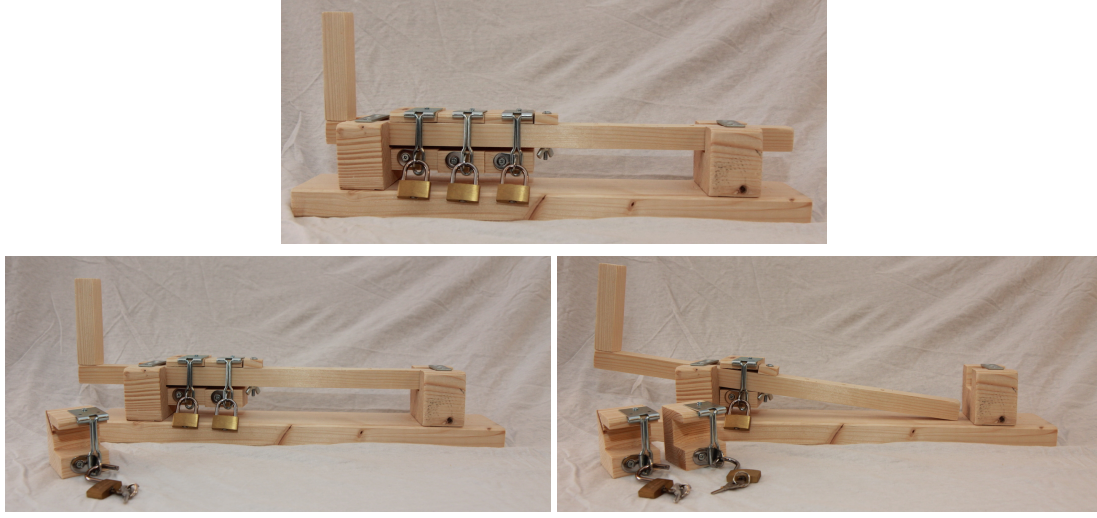

Fig. 6. Physical 2-out-of-3 lock. **Top:** all three blocks attached, padllock closed. **Left:** one block removed, the bar can be moved to the left, but not sufficiently far to open. **Right:** two padlocks are removed, the barrier is open.

In this example, on the top image, we have 3 padlocks attached to support of size l and the blocker is installed just after them on the initial configuration. The bar is installed in such a way that it over passes slightly more than the size of one padlock support on the right. In the left image, once one padlock and its support is removed then the bar can move to the left but not totally be removed. Finally, once two padlock supports are removed we can open the system.

Algorithm 1 k-out-of-n physical threshold lock

Input: $k \leqslant n$, n padlocks, n supports in wood of the same size l, a bar of wood of size at least $(n + k - 1) \times l$, a support of size l, and a blocker.

Output: A k-out-of-n physical threshold padlock.

 1: Distribute one individual key of one padlock to each of the n participants.
 2: Lock the n padlocks on the n latches attached to wooden supports on the long wooden bar.
 3: Install the blocker after the n padlocks.
 4: Install the bar in the system in order that it over passes by slightly more than $(k - 1) \times l$ (and not more than kl) the system general lock.

---

fig. 7, left, shows our prototype in a 2-out-of-4 configuration. The prototype can be configured for k-out-of-n systems for any $k \in \{1, 2, 3\}$ and $n \in \{3, 4, 5, 6\}$. By moving the wooden block attached to the moving bar (red circle in fig. 7, right) one can fix the number of blocks that can be attached, i.e., n. By moving the block on the right (blue circle in fig. 7, right) one can fix the number of blocks that need

to be removed before the bar can be opened, i.e., the threshold k: on the rightmost position, removing one block is sufficient to open the bar. When moving this block to the left one can increase the number of blocks that need to be removed before the bar opens. For convenience, in our prototype everything can be easily adjusted using screws, but obviously, in a real implementation, they need to be permanently fixed to ensure security.
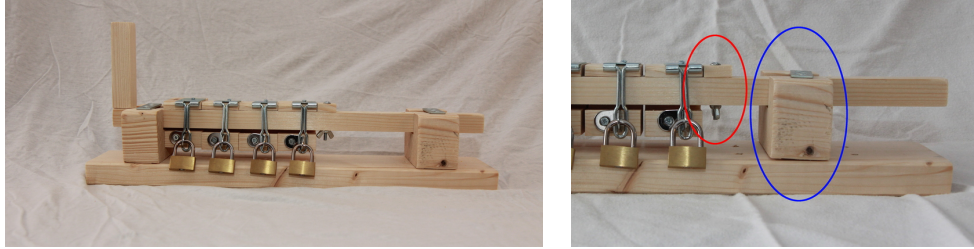

Fig. 7. **Left:** Physical 2-out-of-4 threshold lock system; **Right:** configuring our prototype: the block in the red circle fixes the number n (here: 4), the block in the blue circle fixes the threshold k (here: 3). One can also see the holes that allow the block to be fixed in other positions.

Our technique can also be used to implement *weights* by using blocks of different sizes. fig. 8 shows an example where either one "master" key (opening the padlock on a larger block) can be used to open the lock, or any two of the other keys (opening the padlocks on the smaller blocks). The same idea can also be used to implement a policy where, e.g., either Alice and one other participant, or any three other participants are required to open the lock. It suffices to give Alice the keys for the larger block, and use a configuration that requires the removal of three small blocks to open.

Our system is *ad-hoc* since once it is set up, each participant can install their own lock, which avoids having to trust the dealer as in existing cryptographic solutions. Note that to avoid problems during the
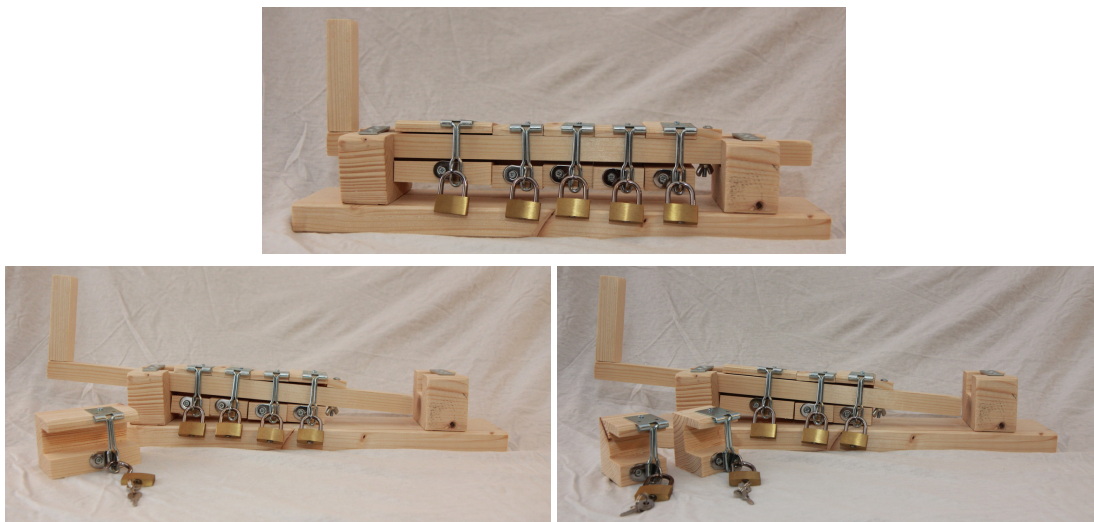

Fig. 8. Physical lock with weighted keys. **Left:** all blocks are attached and locked. **Right:** the "master" key was used to open the larger block, unlocking the barrier. **Bottom:** two "normal" keys opening normal blocks also allow to open the barrier.

setup phase, we assume that all participants install their locks at the same time, right after the lock has been set up.

Our solution is also *reusable* as it can be locked again, unlike for example a solution using cryptographic secret sharing to share a code for a combination lock, where the code would be revealed once and for all: such a lock thus cannot be effectively locked again without changing the code. Note that a system with a combination lock would also require a special procedure or a trusted third party to setup the combination initially. Moreover, our system also protects users against the *daisy chain attack* as only one padlock can be fitted to the latch of any block.

By construction our solution is *verifiable* since everyone can check if there is at least one padlock that can be opened with the secret key that he has received. Comparing to the mathematical solution proposed in [6] consisting in giving extra information to each participant to convince him that he received a valid point of the polynomial, our solution does not require any extra material, nor does it require any trusted third party. There are thus at least three direct applications of our physical threshold system:

(1) Our system can be used to construct a physical verifiable secret sharing protocol. As it can also easily be extended to deal with weights, we also have a physical equivalent to the cryptographic protocol given in [22, 23].
(2) Threshold cryptography has been applied to voting, e.g., in [24]. Our system can be used to secure physical pen and paper voting, by ensuring that the ballot box can only be opened if *k*-out-of-*n* trustees agree.
(3) As a user never has to reveal his physical key, our mechanism can also be used to design a *k*-out-of-*n* authentication mechanism.

4. Formalization and Generic Bounds on the Number of Padlocks

We now establish bounds on the number of padlocks required to realize a certain threshold. We assume that padlocks are more expensive than keys, i.e., we will try to implement threshold systems with fewer padlocks, even if this means duplicating some of the keys. We define a padlock system to be any arrangement of padlocks protecting something. For the sake of simplicity, in the following, we consider this to be the possibility to "*open a door*".

Definitions 1. *A* padlock *is a device requiring a single* key *to be opened (keys can be duplicated). A* padlock-system *is a device comprising an arrangement of* latches *that prevents a* door *to be opened when some padlocks are attached to some of the latches.*

Definition 1. *A k-threshold padlock system is a padlock system with an arrangement of padlocks and a distribution of keys that allows any group of k or more participants to open the door and prevents any group of strictly less than k participants to open it.*

Remark 1. *While directly applicable to physical padlock systems, this definition also applies to some cryptosystems. For instance consider any symmetric or asymmetric cryptosystem with a shared (duplicated) decryption key. Closing a padlock could just be ciphering with an encryption key; setting a padlock-system could just be multiple encryption (even if electronic threshold cryptosystems are more complicated) and opening the door is deciphering. For this example, the only difference with physical system is that the order of encryption must be taken into account for decryption.*

Now, most of the lower *bounds described in this section only suppose the existence of a threshold system satisfying the above definitions. Therefore those lower bounds also apply to electronic threshold cryptosystem satisfying definitions 1 and definition 1.*

Definition 2. *Let n be the number of players and $k \leqslant n$ be a threshold of players required to open the "door". Then $\ell_{k,n}$ is the minimal number of padlocks, in any arrangement, allowing a k-out-of-n threshold opening of the door. Also, we define the* rank *of an arrangement of padlocks and keys as the maximal number of keys owned by any player.*

For instance, we have that:

- $\ell_{1,n} = 1$: one padlock with everybody having a copy of the same key is sufficient.
- $\ell_{k,n} \leqslant n$: by our system described in section 3, see fig. 6.

4.1. Sperner Families

Using the fact that all subsets of size k of the n participants can open the door, and no subset of $k-1$ or less can do it, we have the following results. First, it is easy to see that with only $k-1$ or fewer different locks, one cannot ensure a threshold of at least k.

Lemma 1. $\forall k \geqslant 2, \ell_{k,n} \geqslant k$.

Proof. Suppose for $\ell_{k,n}$ we have an existing threshold system where a minimum of k people is required to open the door, and moreover any subset of k people can open it. Suppose $t = \ell_{k,n} \leqslant k-1$ and consider one group of k people able to open the door. For this, whatever the arrangement, they had to open some of the t padlocks, thus with at most t keys. This is less keys than the number of people, so there must exist a subgroup of at most t people owning these t keys and:

- Any of the k people must own at least one of the t keys, otherwise they are not required to open the cabinet and $k-1$ people are enough.
- By induction on a subgroup of size $1 \leqslant u < t$, an $(u+1)$-th person, among the remaining $k-u$, must own the key of one padlock not owned by the previous u, otherwise this person is not required and $k-1$ people are enough.

Now, this subgroup of size at most t is thus able to open the door by themselves. But $t \leqslant k-1 < k$, is below the threshold, a contradiction. $\quad\square$

Second, we see that if the set of keys of a participant is included in another participant's set of keys, intuitively the first participant is "useless" to achieve the threshold.

Lemma 2. *Let $k \geqslant 2$, and set up an arrangement of padlocks and a distribution of keys with a k-out-of-n threshold opening. No participant can own a set of keys that is a subset of another participant's set of keys.*

Proof. Let A have a set of keys included in that of B. As $k \geqslant 2$, A and B can be in a size k subset of participants that can open the door. But then the keys of A are useless since B has all of them. Therefore there would be a size $k-1$ subset of participants able to open the door, a contradiction. $\quad\square$

This shows for instance that each participant must have at least one key. Further, this means that the sets of keys must form a family of inclusion-free subsets. This is called a *Sperner family* or a *clutter* [25]. The padlocks can then be seen as the vertices of a hypergraph, where each participant is represented by a hyperedge, the set of its owned keys. The *rank* is then the maximal cardinality of a hyperedge. Then Sperner's Theorem combined with lemma 2, also gives the following lower bounds: $\forall n, t$ and $k \geqslant 2$, if $\ell_{k,n} = t$ then $\binom{t}{\lfloor t/2 \rfloor} \geqslant n$.

Proof. By Sperner's Theorem [25], the size of any Sperner family with t elements is upper bounded by $\binom{t}{\lfloor t/2 \rfloor}$. Distributing keys for t padlocks to n participants while satisfying lemma 2 thus requires $\binom{t}{\lfloor t/2 \rfloor} \geqslant n$. $\square$

$\forall n \geqslant 1$ and $\forall t \geqslant 2$ even, if $k \geqslant 3$ and $\ell_{k,n} = t$ then $\binom{t}{t/2} > n$.

Proof. By section 4.1, the only other possibility is $n = \binom{t}{\lfloor t/2 \rfloor} = \binom{t}{t/2}$. But then the unique available Sperner family is that of all subsets of equal size $t/2$. In this family there exist pairs of subsets with an empty intersection. The union of these two subsets is thus of exactly t keys and must be able to open the door. Therefore the threshold cannot be larger than 2. $\square$

Lemma 3. $\ell_{2,3} \geqslant 3$ *and* $\forall n \geqslant 4, \ell_{2,n} \geqslant 4$.

Proof. Suppose $\ell_{2,n} = 2$. Then, if a single person has both keys, she can open both padlocks. Hence, whatever the arrangement of padlocks, she can open the door alone and $k < 2$, a contradiction. Therefore nobody can have more than one key. As $k = 2$, then two persons are sufficient to open the door. They cannot have the same key by lemma 2. But with only 2 distinct keys and $n \geqslant 3$ people, at least two persons must have the same key, a contradiction again. Overall, 2 padlocks are thus not enough. For instance, $\ell_{2,3} \geqslant 3$. Finally, for $t = 3$, section 4.1, shows that $\binom{3}{1} = 3 \geqslant n$, thus $\forall n \geqslant 4, \ell_{2,n} \geqslant 4$. $\square$

We have thus now for instance the following results:

- $\ell_{n,n} = n$: use lemma 1 for the lower bound and our design for the upper bound.
- $\ell_{2,3} = 3$ and $\ell_{2,4} = 4$: use lemma 3 for the lower bound and our design for the upper bound.

4.2. Using $\mathcal{O}(\log(n))$ Padlocks for a Threshold of 2 with n Participants

Now we propose, in algorithm 2, an arrangement for a 2-out-of-n participants threshold system, using no more than n padlocks, and strictly less as soon as $n \geqslant 5$. Indeed if the threshold is only 2, then it is possible to reduce the number of padlocks using our design. The idea is that whenever two participants have a distinct set of keys then both of them have a strictly larger set of keys than any of them taken separately.

fig. 9 shows our prototype in a 3-out-of-4 configuration. Now, using algorithm 2, this configuration can also be used to implement a 2-out-of-5 threshold system with only four padlocks by copying keys and distributing them in such a way that each participant has a distinct subset of keys (as stated in theorem 1). In this example, any two participants together will have at least three different keys, which suffices to open the 3-out-of-4 device. This shows that $\ell_{2,5} = 4$.


Algorithm 2 Two-out-of-n threshold system with shared keys

Input: $n \geqslant 2$, and $1 \leqslant i \leqslant t \leqslant n$ such that $\binom{t}{i} \geqslant n$.

Output: A 2-out-of-n threshold padlock system with t padlocks.

  1: **if** $t < n$ **then**
  2:     Set up an $(i+1)$-out-of-t design of section 3;
  3:     Create a total of $i \cdot n$ keys by copying the original t keys, such that there are n distinct subsets of
    i keys;   ▷ Since $\binom{t}{i} \geqslant n$
  4:     Give each participant a distinct i-tuple of keys.
  5: **else**   ▷ If $t = n$, set $i = 1$ and use directly our device of section 3
  6:     Set up a 2-out-of-n design of section 3;
  7:     Give each participant one of the n keys.
  8: **end if**
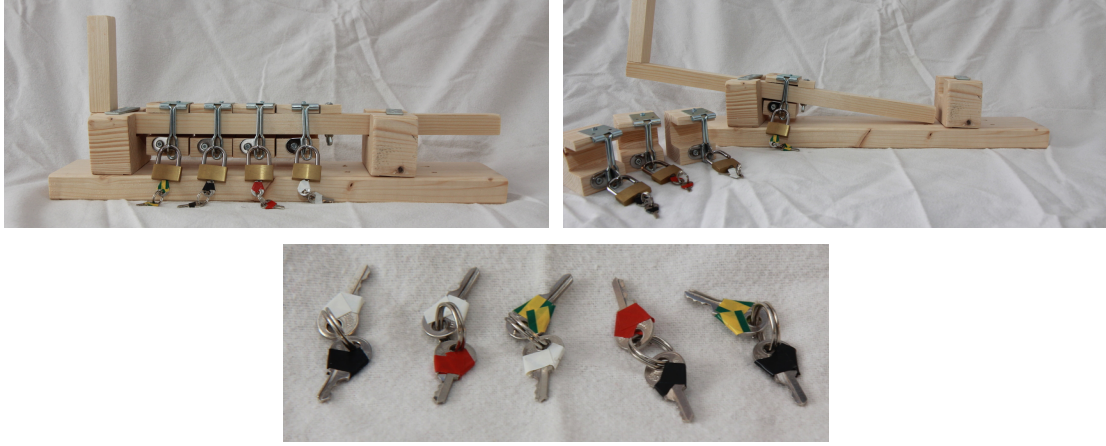
Fig. 9. Physical 3-out-of-4 threshold padlock system. **Left:** all locks closed. **Right:** device opened using 3 out of the 4 locks. **Bottom:** Example key distribution to achieve a 2-out-of-5 threshold system using only 4 padlocks and the 3-out-of-4 device.

The correctness and the optimality of this schemme are proven by the series of results in this section. First, this scheme settles the small cases:

Theorem 1. $\forall n \leqslant 5, \forall k \geqslant 2, \ell_{k,n} = n$, except $\ell_{2,5} = 4$. We also have $\ell_{2,6} = 4$ and $\forall n = 7..10, \ell_{2,n} = 5$.

Proof. For any $n \leqslant 3$ and $2 \leqslant k \leqslant n$ the results where already proven in lemmas 1 and 3. There remains $n = 4$ and $n = 5$, our design providing the upper bound. The proof is done by contradiction.

Let $t = \ell_{k,n}$ and suppose $t \leqslant n - 1$. Let s be the number of participants having a single key. These participants must have different single keys by lemma 2. The remaining $n - s$ participants must own at least 2 keys, but cannot own any of the first s keys, by lemma 2. Each set of keys of these remaining $n - s$ participants cannot be the full set of the remaining $t - s$ keys, again by lemma 2. Therefore, at least, the number $\delta_{s,t}$ of distinct subsets of size at least 2 and at most $t - s - 1$ must be larger than $n - s$ (the requirement is that the size of the clutter, must be larger than $n - s$, but in this clutter all the subsets

must at least be distinct). This is:

$$\delta_{s,t} = \sum_{i=2}^{t-s-1} \binom{t-s}{i} \geqslant n-s \tag{1}$$

But, if $t \leqslant n-1$ and $n \geqslant 4$, then:

$$\delta_{s,t} \leqslant \sum_{i=2}^{n-s-2} \binom{n-s-1}{i} = 2^{n-s-1} - 1 - (n-s-1) - (n-s-1) \tag{2}$$

Therefore, Equation (1) cannot be satisfied whenever Equation (2) is $< n-s$, that is:

$$2^{(n-s)-1} < 3(n-s) - 1 \tag{3}$$

But Equation (3) is true for $n-s \in \{1,2,3,4\}$. Yet $n-s > 1$, otherwise $n-1 \geqslant t \geqslant s$ implies at most $t = s = n-1$, but then there remains no available key for the n-th participant. Hence, we have $n-s \in \{2,3,4\}$.

For $n = 4$, if $s \in \{2,1,0\}$ then Equation (3) is satisfied thus we can dismiss those cases. Finally, there remains no value for s meaning that our hypothesis $t \leqslant n-1$ is false.

For $n = 5$, if $s \in \{3,2,1\}$ then Equation (3) is also satisfied so we can dismiss those cases. There remains the case $s = 0$ for $t = 4$ (the case $t = 3$ is excluded by the fact that $\binom{3}{2} = 3 < 5$). The 5 participants can thus only have 2 or 3 keys each (if one of them has the 4 keys he can open the door alone). If one of the 5 participants owns 3 keys K_1, K_2, K_3 then the other four must all own the fourth key K_4 (otherwise one of them will own only a subset of the first 3 keys, contradicting lemma 2). But then, excluding K_4, these four remaining participants must have distinct non-included subsets of size 1 or 2 of the 3 keys K_1, K_2, K_3, which is impossible. Therefore the rank of the arrangement is 2, that is, all 5 participants can only have 2 keys each. There are $\binom{4}{2} = 6$ possible pairs. W.l.o.g. suppose that only the pair K_3, K_4 is not among the participants pairs.

Then two participants owns (K_1, K_3) for one and (K_2, K_4) for the other, so the two of them can open all the padlocks. This means that $k \leqslant 2$. For $k > 2$, we have a contradiction since no value s can be taken, leading to refute the hypothesis $t \leqslant n-1$. Thus for $n = 5$ and $k \geqslant 3$ we have $\ell_{k,n} = n$.

The remaining case, $k = 2$ is thus actually 2-out-of-5 threshold with at least 4 padlocks where every player owns exactly 2 keys.

This is satisfiable as follows: use a 3-out-of-4 device with our design with 4 padlocks. Then provide the 5 users with distinct pairs of keys. Not a single user can open 3 padlocks. But with distinct pairs of keys all pairs of participants own at least 3 different keys.

Finally, algorithm 2 gives a solution as soon as t is such that $\binom{t}{2} \geqslant n$, while section 4.1 prevents any solution with $\binom{t}{\lfloor t/2 \rfloor} < n$. But with $t = 4$ and $t = 5$, $\binom{t}{2} = \binom{t}{\lfloor t/2 \rfloor}$. So the upper bound of algorithm 2 is also a lower bound. Now $\binom{4}{2} = 6$ and $\binom{5}{2} = 10$ give the maximal respective number of participants. $\square$

Second, we give an asymptotic estimate for larger cases: algorithm 2 makes it possible to implement a 2-out-of-n threshold padlock system with only $2\lceil \log_2(n) \rceil$ padlocks and $n\lceil \log_2(n) \rceil$ keys:

Proposition 1. *algorithm 2 correctly provides a 2-out-of-n threshold padlock system and for $n \geqslant 2$,*
$\ell_{2,n} \leqslant 2\lceil \log_2(n) \rceil$.

Proof. Consider an $(i + 1)$-out-of-t threshold system with t padlocks for $\binom{t}{i} \geqslant n$. Distribute i keys for each participant, such that all the hyperedges are distinct. This is possible as $\binom{t}{i} \geqslant n$. No single participant can open the device, but any two participants have different hyperedges of size i and thus have at least $i + 1$ distinct keys. This is enough to open the door and algorithm 2 is correct. Finally, $\binom{t}{i} \geqslant (t/i)^i$, so we can for instance set $i = \lceil \log_2(n) \rceil$, so that each participant gets that many keys, and setup $t = 2\lceil \log_2(n) \rceil$ padlocks, as $2^{\lceil \log_2(n) \rceil} \geqslant n$. $\square$

For instance, the first case where triples are better than couples in algorithm 2 is for $n = 16$. As $\binom{6}{2} = 15$ and $\binom{7}{2} = 21$, with pairs the Algorithm would use 7 padlocks, where 6 are enough: setup a 4-out-of-6 device and give distinct triples of copies of the 6 keys to each of the 16 participants. This is possible as $\binom{6}{3} = 20 \geqslant 16$. Then any pair of participants have at least $3 + 1 = 4$ different keys and they can open our device. Overall, we have that the minimal number of padlocks for a 2-out-of-n threshold system is $\mathcal{O}(\log(n))$ with $\mathcal{O}(n \log(n))$ keys. Indeed, the lower bound is given in section 4.1, and it is realizable by algorithm 2. We thus have proven section 4.2.

For $n \geqslant 2$, $\ell_{2,n} = \min \left\{ t, \text{ s.t. } \binom{t}{\lfloor t/2 \rfloor} \geqslant n \right\}$.

Proof. The lower bound is given by section 4.1. For the upper bound consider as in proposition 1 an $(\lfloor t/2 \rfloor + 1)$-out-of-$t$ threshold system with t padlocks and distribute $\lfloor t/2 \rfloor$ keys to each participant, such that all the hyperedges are distinct. $\square$

4.3. A Trick for 2-out-of-n Padlock System with Exactly $2\lceil \log_2(n) \rceil$ Padlocks

It is possible to directly obtain a 2-out-of-n padlock system with exactly $2\lceil \log_2(n) \rceil$ padlocks and no external device. To see this, one can mix 2-out-of-2 devices and a 1-out-of-$\log_2 n$. Then, remark first that daisy chains are 1-out-of-t devices and second that setting t padlocks on the same latch provides a t-out-of-t device. For the illustration purpose, we describe the alternative construction in two steps: first with devices, then without any device.

The first construction is as follows:

(1) Consider 2-out-of-2 devices, similar to those of fig. 4, and take $\lceil \log_2 n \rceil$ of them; On each one of these 2-out-of-2 devices set two padlocks, one black and one white. This is $2\lceil \log_2 n \rceil$ distinct padlocks;
(2) Attach these $\lceil \log_2 n \rceil$ devices to each latch of a 1-out-of-$\lceil \log_2 n \rceil$ device;
(3) Order the participants and give them a distinct number between 0 and $2^{\lceil \log_2 n \rceil}$. Then give each one of the participants $\lceil \log_2 n \rceil$ keys, following the binary digits of her number. On the one hand, if the i-th bit of her number is zero, then give her the key of the white padlock of the i-th 2-out-of-2 device; on the other hand, if the i-th bit of her number is one, then give her the key of the black padlock of the i-th 2-out-of-2 device.

This is a total of $2n\lceil \log_2 n \rceil$ keys. Now each participant alone cannot open any latch (she owns only one of the two keys required for that), therefore she cannot open the door. Differently, any two participants have at least one bit, i, of difference. For this bit, the two of them thus have both keys of the i-th 2-out-of-2 device. They can thus open it, thus open the 1-out-of-$\lceil \log_2 n \rceil$ device and open the door. This is overall a 2-out-of-n system.

Now the second construction mimics the first one given above, but without any particular device. It is shown in algorithm 3.

Algorithm 3 Two-out-of-n double daisy chain with only $2\lceil \log_2 n \rceil$ padlocks

---

 1: Setup a daisy chain with $\lceil \log_2 n \rceil$ white padlocks;
 2: For each white padlock, double it with a black padlock: that is form a daisy chain with double links;
 3: Similarly, each participant receives $\lceil \log_2 n \rceil$ keys, white or black, according to the binary decomposition of her number.

First, similarly, each participant alone cannot open the chain, as she cannot open any link, having only one of the two keys required to open one link. Second, similarly also, any two participants having different numbers have at least both keys of one double link and can open the chain and the door.

This is simple and does not require any additional device, apart from $2\lceil \log_2 n \rceil$ padlocks. Note that this usually uses more padlocks than algorithm 2. For instance, for a 2-out-of-8 system, the above method requires $2 * 3 = 6$ padlocks and for a 2-out-of-10 system, it requires $2 * 4 = 8$ padlocks, while section 4.2 uses only 5 padlocks for both cases as $\binom{5}{2} = 10$.

5. Access Structures

In a secret sharing scheme, a datum d is broken into shadows which are shared by a set of trustees. The family $\{G \subseteq P : G \text{ can reconstruct } d\}$ is called the *access structure* of the scheme. A k-out-of-n scheme is a secret sharing scheme having the access structure $\{G \subseteq P : |G| = k\}$ [26]. In this section we show how to physically implement access structures defined by logic gates. Numeric solutions with interpolation usually use one evaluation point for each literal and one polynomial per clause. Our physical solution uses instead only one padlock for each distinct variable and one device per whole normal form.

5.1. Towards a Padlock Algebra with One Device per Normal Form

A generalization of threshold schemes is to be able to implement any access scheme described by a logic formula. This is possible by implementing AND and OR gates, as shown in proposition 2 and algorithms 4 and 5, following [26, 27]. A first idea is to use chains so that opening a padlock actually frees a chain that can free several latches. Then a second idea is that 1-out-of-n systems are just like a disjunction while n-out-of-n systems are just like a conjunction.

algorithm 4 shows how to generate a padlock system openable by any satisfiable realization of a disjunction with t clauses and n distinct variables. For this, a single 1-out-of-t device is set. It will open if any of the t conjunctive clauses is true. Associate each latch of the device to one conjunctive clause. Then associate one padlock for each variable. To simulate the subjection of a clause to a variable, each padlock closes a chain passing through each latch corresponding to a clause containing that variable (and thus preventing the opening of those latches if that padlock is not open).

fig. 10 gives an example of algorithm 4 on the logic formula $(A \land B) \lor (A \land C) \lor (B \land D) \lor (E)$.

Now for conjunctions, we instead use a t-out-of-t master structure and several other 1-out-of-k systems, one for each conjunction in the CNF, as shown in algorithm 5.

fig. 11 gives an example of algorithm 5 on the logic formula $(A \lor B \lor C) \land (D) \land (C \lor E)$.

Both algorithms 4 and 5 thus provide a way to build systems with a number of padlocks equal to the number of distinct variables in the normal form. This is proposition 2.

Algorithm 4 Physical DNF with one padlock for each variable

Input: A disjunctive normal form with t clauses and n variables.

Output: A system with n padlocks, openable by any satisfiable realization of the normal form.

1: Set up a 1-out-of-t threshold system, with one latch for each clause;

2: For each variable present in the formula: pass a chain through the hole of each latch corresponding to a conjunction containing that variable; close that chain with one padlock.
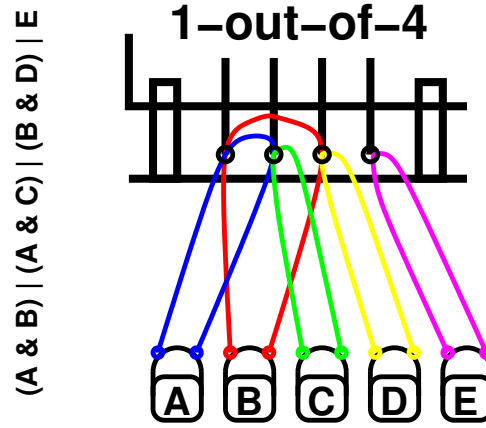

Fig. 10. Algorithm 4 on a disjunctive normal form: one padlock closing one chain per term and a 1-out-of-4 device using a single latch for each clause, to realize $(A \land B) \lor (A \land C) \lor (B \land D) \lor (E)$.

Algorithm 5 Physical CNF with one padlock for each variable

Input: A conjunctive normal form with t clauses and n variables.

Output: A system with n padlocks, openable by any satisfiable realization of the normal form.

1: Set up a t-out-of-t threshold system;

2: Set up one 1-out-of-k_i threshold system for each conjunctive clause with k_i variables. Attach each 1-out-of-k_i system to one of the t latches of the t-out-of-t system.

3: For each variable present in the formula: pass a chain through a free latch of each 1-out-of-k_i system corresponding to a clause containing that variable; close that chain with one padlock.

Proposition 2. *Any disjunctive or conjunctive normal form with t clauses, m distinct variables and no negation is realizable with m padlocks*

Proof. First for disjunctive clauses: they require one 1-out-of-t threshold system and m chains, as shown in algorithm 4. The "door" can be opened only by a satisfiable interpretation where TRUE means opening the padlock and FALSE means letting it closed.

Similarly one can create arrangements for conjunctive normal forms, also with as many padlocks as there are distinct variables as shown in algorithm 5. □

5.2. Further examples of logic formulae

In section 5, we show that any access scheme described by a logic formula without negation can be implemented using simple physical devices. In this section, we show some other constructions that can
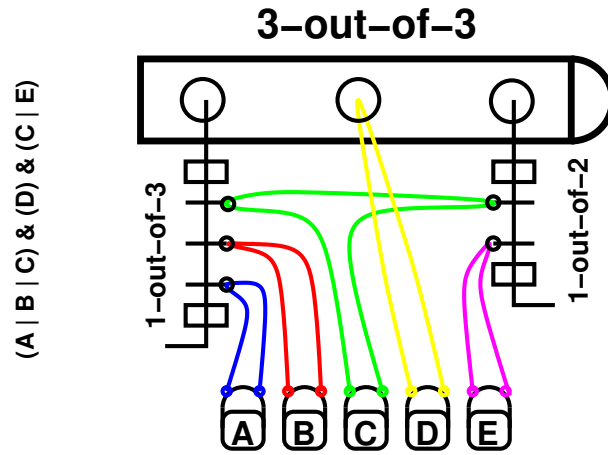
Fig. 11. Algorithm 5 on a conjunctive normal form: one padlock closing one chain per term and a 3-out-of-3 device using a single 1-out-of-k_i device for each clause (if $k_i > 1$), to realize $(A \vee B \vee C) \wedge (D) \wedge (C \vee E)$.

simplify the use of algorithms 4 and 5 for normal forms. We also show how our physical methods can implement some formulae that are proven impossible with a single secret sharing scheme.

First, to implement algorithm 5 we need a 1-out-of-k_i system for each clause. If this is simpler, one can always build such a system by composing small 1-out-of-2 systems. For instance fig. 12, left, shows how to create one tree for each disjunctive clause as used in algorithm 5: assemble U-shaped metal rods. It is also possible to create a daisy chain of 1-out-of-2 devices like the one in fig. 12, right.

Fig. 12. Left, a tree-like disjunctive clauses system by GateKeeper GM P6006 to combine 2, 3 or 4 locks; right, a physical 1-out-of-2 disjunction Model Cb2 by Sharelox.

Now, second, we give examples of usage of all our devices and construction. For this we offer physical solutions for two examples, proven unrealisable using a single scheme. Indeed, [27] shows that the two following cases cannot be solved if users must use the same system of shares:

(1) $(A \wedge B) \vee (C \wedge D)$
(2) $(A \wedge B) \vee (C \wedge D) \vee (B \wedge C)$

However, with a physical system, we can implement such access schemes with somewhat less devices as we have physical tools to combine conjunctions and disjunctions:

- Conjunctions can be implemented with n-out-of-n systems as in fig. 4;
- Disjunctions can be implemented with 1-out-of-n systems as in figs. 2 and 3 or in section 3.

For the first formula: $(A \wedge B) \vee (C \wedge D)$, we report no improvements. One can implement algorithm 4 on this formula, and provide a physical solution: use a daisy chain of two 2-out-of-2 classical equivalent

system, as in fig. 4, one for each conjunction. This is not different from the solution of [27] with two distinct secret sharing schemes.

Now, for the second formula, $(A \wedge B) \vee (C \wedge D) \vee (B \wedge C)$, a naive implementation (resp. [27] solution) would require six padlocks (resp. 6 shares for three systems of 2 shares, one for each clause). But it is possible to use only four padlocks, as shown in algorithm 6 and fig. 13.

Algorithm 6 Physical realization with only 4 padlocks of [27, Theorem 3]

1: Setup a 1-out-of-3 threshold system (thus with 3 latches);
2: Put a padlock A on the first latch and a padlock D on the third latch;
3: Pass a chain in the holes of the first two latches and close that chain with a padlock B;
4: Pass a chain in the holes of the last two latches and close that chain with a padlock C.
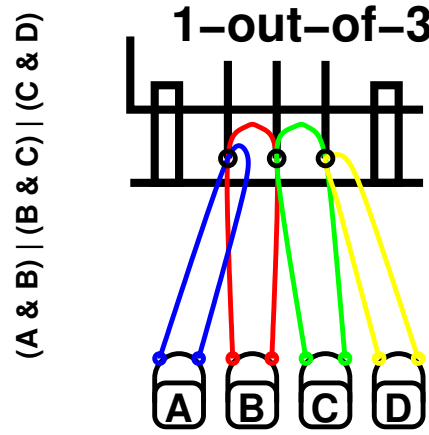

Fig. 13. Algorithm 6 on $(A \wedge B) \vee (C \wedge D) \vee (B \wedge C)$ with one padlock per variable (and not per literal in the formula).

Remark 2. *Finally, note that, with our novel design, proposition 2 is not optimal. Consider for instance the DNF with a single participant able to open the door or any two among five others: $A \vee (B \wedge C) \vee (B \wedge D) \vee (B \wedge E) \vee (B \wedge F) \vee (C \wedge D) \vee (C \wedge E) \vee (C \wedge F) \vee (D \wedge E) \vee (D \wedge F) \vee (E \wedge F)$. proposition 2 would require 6 padlocks and a 1-out-of-11 design. However, we can use theorem 1 and our design for a 3-out-of-4 lock with only 4 padlocks as described thereafter: set up a 3-out-of-4 design with 4 padlocks; give pairs of distinct keys to each participant B, C, D, E, F; give any three distinct keys to A.*

5.3. Knotted padlocks

A post on crypto.stackexchange.com by Ahle [28] hints that one could create k-out-of-n threshold padlock systems using n padlocks and some wire.

His idea is to have the wire securing the door and going around the rings of the padlocks in a certain configuration. If a padlock is opened then it frees his part of the wire and potentially more from other padlocks. The example given is for a 1-out-of-2 system: "*Say you have one wire to which the [door] is fastened and two padlocks. You want that if either of the locks are opened, the wire is completely freed. You do this*

by letting the wire go first clockwise around [the ring of] the first [padlock], then clockwise around the second, then anticlockwise around the first and finally anticlockwise around the second. It can be thought of as $aba^{-1}b^{-1}$. If you remove either, the other cancels out. It generalizes to any k out of n padlocks".

This is a neat idea, which however turns out to not generalize easily to any k out of n system, though.

First, associate a variable from a non-commutative group to each one of n padlocks. To simulate the opening of a padlock set this variable to 1, the neutral element of the group, seen multiplicatively. Suppose that this variable represents one clockwise wrapping of the wire and that the inverse of that variable represents the anticlockwise wrapping. Then, the sequential arrangement of the wrappings of the wire around the rings of the padlocks is a sequential multiplication of these variables and their inverses, just like a Knot group presentation.

For instance, if a clockwise wrapping is directly followed by an anticlockwise wrapping then this is useless and represented by $xx^{-1} = 1 = x^{-1}x$. Finally, if the door is opened when some padlocks are opened then it means that the multiplication of the variables is equal to 1 when the variables associated to the opened padlocks are set to 1. We will say that padlocks are *knotted* if there is a sequential wrapping of a wire around the rings of its padlocks.

On the one hand, we see that the equation $aba^{-1}b^{-1}$ represents a generic OR gate: for the two padlock case, if one variable is set to 1, then either $aba^{-1}b^{-1} = bb^{-1} = 1$ or $aba^{-1}b^{-1} = aa^{-1} = 1$. This also generalizes to creating the OR of any *independent* subsystems: if X and Y are two equations for two knotted systems with *distinct padlock sets*, then $XYX^{-1}Y^{-1}$ is the equation of the OR of these two systems.

If the equations are not independent, then some cancellations can occur. Consider for instance the formula $a \vee a$; then $aaa^{-1}a^{-1}$ is always 1 even if a is not open. Now, in order to prevent such cancellations, it is possible to surround a set of equations by an independent padlock and its inverse. Then no cancellation can happen. Even better, one can use a simple ring (this is a padlock that nobody can open) and wrap around it one way before the equation and the other way after the equation. For that simple ring, denote by $\odot$ one clockwise wrapping of the wire around it (and by $\odot^{-1}$ an anticlockwise wrapping). Note that the latch of the door, if any, could be used as this ring too. In any case, for instance, $X\odot Y\odot^{-1}X^{-1}\odot Y^{-1}\odot^{-1}$ then represents a generic OR gate where the subsystems need not be independent. Indeed, if, and only if, any of X or Y is 1 then everything collapses.

On the other hand, to represent an AND gate between two independent padlock systems, then simply multiplying both equations suffices, in any order and with any inverse (i.e., independence ensures that XY, YX, $X^{-1}Y^{-1}$, $Y^{-1}X^{-1}$, XY^{-1}, etc. all represent the conjunction). Similarly, one can enclose dependent subsets of padlocks with the simple ring $\odot$.

For instance some access structures of the previous subsections can also be realized this way with one padlock per literal:

- $(A \wedge B) \vee (C \wedge D)$ can be represented by $X = abcdb^{-1}a^{-1}d^{-1}c^{-1}$;
- $(A \wedge B) \vee (C \wedge D) \vee (B \wedge C)$ by $X\odot bc\odot^{-1}X^{-1}\odot c^{-1}d^{-1}\odot^{-1}$.

There is a nice linear setup, for $(n-1)$-out-of-n threshold systems, as shown in lemma 4.

Lemma 4. *The knotted padlock system, setup with with n padlocks and a wire, and wrapped $2n$ times, following the presentation $x_1 x_2 \ldots x_n x_1^{-1} x_2^{-1} \ldots x_n^{-1}$, is a $(n-1)$-out-of-n threshold padlock system.*

Proof. Set any subset of size $n-1$ of the variables to 1, there remains $x_j x_j^{-1}$. For any subset of size $n-2$ or less, there would remain at least $x_i x_j x_i^{-1} x_j^{-1}$ with $i \neq j$ and the door is not freed. $\square$

This setup actually is optimal as shown in lemma 5

Lemma 5. *Let $k \geqslant 1$ and $n \geqslant k+1$, A knotted padlock k-out-of-n threshold system, setup with with 1 padlock per participant and a wire, requires an even number of wrappings, and at least $2n$ of them.*

Proof. Let a_i be the number of wrappings around padlock i. Suppose that k participants not including i open their padlock. This is possible since $n > k$. Then the system must be freed. Therefore, any clockwise wrapping around i must be accompanied by an anticlockwise one. This shows that $a_i = 2\alpha_i$ and that the total number of wrappings is even. Now suppose that there are no wrappings around padlock i. Then the participant i is useless in opening the system. This contradicts the notion of a threshold system. Finally, we have that $a_i \neq 0$ and thus that $a_i \geqslant 2$. The total number of wrappings is thus larger than $2n$. $\quad\square$

By lemmas 4 and 5 we have an optimal linear knotted system for $(n-1)$-out-of-n threshold padlock systems. But, unfortunately, we have no simple candidate for other thresholds. Generic threshold system can be implemented with simple gates, but then they must use an exponential number of them [29]. So this method of knotting the padlocks might not be directly practical. For instance, an exhaustive search of the $\sum_{j=3}^{4} 6^{2j} = 1\,726\,272$ formulas with 3 variables and their 3 inverses (since the number of terms must be even and larger than $2*3$ by lemma 5), showed that no formula exists for a 1-out-of-3 threshold system with strictly less than 10 terms. The smallest one is thus a permutation of $abcb^{-1}c^{-1}a^{-1}cbc^{-1}b^{-1}$, with 10 wrappings.

For more generic thresholds we were only able to devise a solution with an exponential number of wrappings, as shown in algorithm 7, again loosing practicality for most of these knotted systems.

Algorithm 7 Knotted padlock threshold system

Input: $k \geqslant 1$, a wire and $n \geqslant k$ padlocks.
Output: A k-out-of-n threshold padlock system with n knotted padlocks.
 1: **if** $k == n$ **then**
 2:     **return** the wiring of all the padlocks together, with presentation $x_1 x_2 \ldots x_{n-1} \ldots x_n$.
 3: **end if**
 4: **if** $k == n-1$ **then**
 5:     **return** the wiring of all the padlocks together, with presentation $x_1 x_2 \ldots x_n x_1^{-1} x_2^{-1} \ldots x_n^{-1}$.
 6: **end if**
 7: **if** $k == 1$ **then**
 8:     Recursively compute a presentation X for a 1-out-of-$(n-1)$ system with the last $n-1$ padlocks;
 9:     **return** the wiring of all the padlocks together, with presentation $x_1 X x_1^{-1} X^{-1}$.
10: **end if**
11: Recursively compute a presentation X for a $(k-1)$-out-of-$(n-1)$ system with the last $n-1$ padlocks;
12: Recursively compute a presentation Y for a k-out-of-$(n-1)$ system with the last $n-1$ padlocks;
13: **return** the wiring of all the padlocks together, with presentation $x_1 X \odot Y \odot^{-1} X^{-1} x_1^{-1} \odot Y^{-1} \odot^{-1}$.

Theorem 2. *Let $k \geqslant 1$ and $n \geqslant 2$, algorithm 7 is correct and requires a number of wrappings $W(k,n)$ that satisfies:*

- $W(1,n) = \frac{3}{2} 2^n - 2$;
- $W(n-1,n) = 2n$;

- $W(n, n) = n$;
- *For $k \in [2..(n-2)]$, $W(k, n) \geqslant \frac{3}{2}2^n + 6$.*

Proof. For the correctness, we look at the cases. If $k = n$ then this is a n-out-of-n system. All padlocks are wired, one after the other a single time. Therefore no simplification can occur by opening padlocks. This means that all padlocks must be opened to free the system and that no strict subset of owners can open the system.

The $k = n - 1$ case is settled by lemma 4.

If $k = 1$. Then we proceed by induction. We have seen that a 1-out-of-2 system is indeed represented by a formula $aba^{-1}b^{-1}$. Now suppose that we have a presentation X valid for a 1-out-of-$(n-1)$ system with $n - 1$ padlocks. Then an additional participant, numbered 1, uses a new padlock and the overall presentation is $E = x_1 X x_1^{-1} X^{-1}$. If x_1 is opened then $E = XX^{-1}$ cancels out. If any of $x_2, \ldots, x_n$ is opened then X cancels and $E = x_1 x_1^{-1}$ also cancels out. Therefore any participant alone can open the system. Conversely, the system does not collapse: by induction, first, neither X nor X^{-1} is 1 if no padlock is opened. Second x_1 and X are using independent sets of padlocks so no cancellation can occur between $x_1 X$, $X x_1^{-1}$, nor $x_1^{-1} X^{-1}$.

Finally the generic case, with now $E = x_1 X \odot Y \odot^{-1} X^{-1} x_1^{-1} \odot Y^{-1} \odot^{-1}$, is also handled by induction. k participants are either x_1 and $k - 1$ others or k other than x_1. In the first case, if x_1 and, by induction, X cancel out, then $E = \odot Y \odot^{-1} \odot Y^{-1} \odot^{-1} = 1$. In the second case, also by induction, Y cancels out and $E = x_1 X \odot \odot^{-1} X^{-1} x_1^{-1} \odot \odot^{-1} = 1$. Therefore any k or more participants can open the system. Conversely, suppose only at most $k - 1$ padlocks are opened. Then by induction, X can vanish, but not Y. Further, X can vanish only with the opening of at least $k - 1$ padlocks. Therefore x_1 and X cannot vanish simultaneously. Thus either nothing vanishes or E has one of two forms, $E = X \odot Y \odot^{-1} X^{-1} \odot Y^{-1} \odot^{-1}$, with non-vanishing X and Y, or $E = x_1 \odot Y \odot^{-1} x_1^{-1} \odot Y^{-1} \odot^{-1}$, with non-vanishing x_1 and Y. In both cases, the system is not opened. We have proven that the system created by algorithm 7 is a k-out-of-n threshold system.

Now for the complexity bound with $n \geqslant 2$. Let $W(k, n)$ be the number of wrappings for a k-out-of-n system created by algorithm 7. We have that $W(2, 2) = 2$, the AND gate, and $W(1, 2) = 4$, the OR gate. Next, for $k = n$, by the construction of Line 2, we have that $W(n, n) = n$. For $k = n - 1$, by the construction of Line 5, we have that $W(n - 1, n) = 2n$. For $k = 1$, by the construction of Line 9, we have that $W(1, n) = 2(1 + W(1, n - 1))$. This is $W(1, n) = (\sum_{i=1}^{n-2} 2^i) + (2^{n-2} W(1, 2)) = \frac{3}{2}2^n - 2$.

Otherwise, we have that $n \geqslant 4$ and $k \in [2..(n-2)]$. There, we first show by induction that $W(k, n)$ satisfies:

$$W(k, n) \geqslant 2^{n-k}. \tag{4}$$

Indeed, this is true for both $W(2, 2) = 2 \geqslant 2^{2-2}$ and $W(1, 2) = 4 \geqslant 2^{2-1}$. Next, for $k = n$, $W(n, n) = n \geqslant 1 = 2^{n-n}$; for $k = n - 1$, $W(n - 1, n) = 2n \geqslant 2 = 2^{n-(n-1)}$ and for $k = 1$, $W(1, n) = \frac{3}{2}2^n - 2 \geqslant 2^{n-1}$. Otherwise, $W(k, n)$ satisfies:

$$W(k, n) = 2(3 + W(k - 1, n - 1) + W(k, n - 1)). \tag{5}$$

Thus by induction and eq. (5), we can lower out the two recursive calls, and have that $W(k, n) \geqslant 2(3 + 2^{n-1-(k-1)} + 2^{n-1-k}) \geqslant 2^{n-k}$ and eq. (4) is proven.

Then, second, we refine this analysis by lowering $W(k, n - 1)$ in eq. (5) with eq. (4). This gives $W(k, n) \geqslant 2(3 + W(k - 1, n - 1)) + 2 \cdot 2^{n-1-k}$. Then we recurse for $W(k - 1, n - 1)$, to obtain that:

$W(k, n) \geqslant 2(3 + 2(3 + W(k-2, n-2) + 2^{n-2-k})) + 2^{n-k} = (6 \sum_{i=0}^{k-2} 2^i) + 2^{k-1}W(1, n-k+1) + 2^{n-k}(k-1) = 6(2^{k-1} - 1) + 2^{k-1}(\frac{3}{2}2^{n-k+1} - 2) + 2^{n-k}(k-1) = \frac{3}{2} \cdot 2^n + 2^{k+1} - 6 + 2^{n-k}(k-1)$. This concludes the proof, using the fact that $(2^{k+1} + 2^{n-k}(k-1) - 6) \geqslant 6$ for $k \geqslant 2$ and $n - 2 \geqslant k$. $\quad\square$

To realize this solution in practice, one could for instance to use a high security cable seal: once fastened the wire cannot be taken out of the seizing device, see fig. 14, left. Before closing the seal, wrap it around the door latch clockwise; then install the *k*-out-of-*n* knotted padlock threshold system on the wire; finally wrap the wire around the door latch anticlockwise and then seal it. This is shown in fig. 14, right.
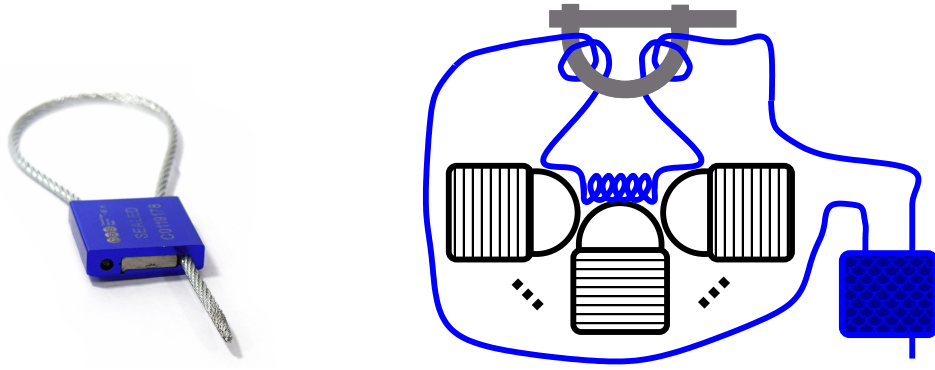

Fig. 14. Secure cable wire seal (left) and knotted padlock threshold system (right).

Now, if the latch is smaller than the seizing part and than the padlocks, the door cannot be opened unless all the locks are removed. This can happen if at least *k*-out-of-*n* participants open the padlocks: then the other ones are freed by construction.

Remark 3. *Unless $k = n$ from the start, algorithm 7 will never encounter the case that the threshold equals the number of remaining participants. All the other cases perform exactly as many clockwise and anticlockwise wrappings. Therefore it is not mandatory to use a sealable cable. Any wire loop with a large enough part will do, for instance an already sealed cable. The setup is more cumbersome, but it is sufficient to pass a curl inside the latch or the ring to simultaneously wrap clockwise and anticlockwise.*

Thus, we have another possibility for a *k*-out-of-*n* physical threshold system with exactly *n* padlocks. Unfortunately, we can make it work only with an exponential number of wrappings. For instance, algorithm 7 requires 279 038 initial wrappings for the 6-out-of-11 case. For now, the setup of this solution is therefore not really practical.

6. Square Root Bounds for Threshold Systems

We now have tools to deal with larger thresholds. First we give a necessary condition for systems using less than *n* padlocks. Knotted designs are not needed, but some small access structure arrangements can help. For instance, we can show that our physical device is optimal when k is larger than $\sqrt{2n}$. For instance, we fully answer Liu's question about the smallest number of locks needed to implement a

6-out-of-11 threshold system: this is 11 padlocks. Then, the necessary condition, together with block design theory and our padlock algebra of section 5, enables us to build padlock systems with strictly less than n padlocks: for instance systems with only about $2.5\sqrt{n}$ padlocks for 3-out-of-n thresholds.

6.1. A Necessary Condition and the Answer to Liu's Problem

We first begin with a necessary condition, analyzing the set difference cardinality of their sets of keys.

Proposition 3. $\forall n$ and $\forall k \geqslant 3$, if a k-out-of-n threshold system uses strictly less than n padlocks, then apart from participants owning the single key of a given padlock, the other participants must satisfy:

(1) *The cardinality of their 2 by 2 set difference is bigger or equal to $k-1$;*
(2) *Each of them owns at least k distinct keys.*

Proof. Apart from participants owning the single key of a given padlock, the others own only keys that are duplicated and owned by several users. In the following we say that these duplicated keys owned by several users are *shared*, and we identify any duplicated keys of the same padlock (we also thus say that shared keys are *reused* when we encounter the duplicate of an already used key).

We thus consider the subgroup of participants owning only shared keys. First, let A, B, C be the sets of keys of three users from this subgroup, such that $(A \setminus B) \subset C$; then $(A \cup B \cup C) = (B \cup C)$. More generally, suppose $|A \setminus B| = d$. Then these d keys are reused (as all the keys in A are shared keys). Thus there exist $d' \leqslant d$ other participants with sets of keys C_j such that $(A \setminus B) \subset \bigcup_{j=1}^{d'} C_j$. Let $\mathcal{C} = \bigcup_{j=1}^{d'} C_j$, then $A \cup B \cup \mathcal{C} = B \cup \mathcal{C}$. In other words, there is a group of $d' + 2$ users with the same keys as a group of $d' + 1$ users. Therefore $d' \geqslant k - 1$ (and $d \geqslant k - 1$, as $d \geqslant d'$): otherwise complete these $d' + 2$ participants with $k - d' - 2$ others. Those k participants can open the door, as well as the $k - 1$ participants obtained when removing A. This would contradict the fact that we have a k-threshold system. So we can restrict the analysis to groups of people having sets of shared keys, with minimal 2 by 2 set difference cardinality larger than $k - 1$.

Second, within such a group of participants owning only shared keys, suppose that one participant P owns a number i of distinct keys strictly lower than the threshold k. Then at least one of his keys cannot be reused. Otherwise there exists a group of $i' \leqslant i$ participants owning the same keys as these i' participants plus the initial one P. As $i' < k$, complete these $i' + 1$ participants with $k - i' - 1$ others. Those k participants can open the door, as well as the $k - 1$ participants obtained when removing P from the group. This would contradict the fact that we have a k-threshold system. We have proven: to build a threshold system with strictly less than n padlocks, apart from participants owning the single key of a given padlock, the other participants must satisfy that both, their 2 by 2 set difference has more than $k - 1$ keys and each of them owns at least k distinct keys. $\quad\square$

We can now fully answer Liu's question with theorem 3, $\ell_{6,11} = 11$. The theorem also shows that our system is optimal for all $k \geqslant \frac{\sqrt{8n+1}-1}{2}$.

Theorem 3. $\ell_{k,n} \geqslant \min\left\{n; \frac{k(k+1)}{2}\right\}$.

Proof. For $k = 2$, lemma 3 gives the result. Now, for $k \geqslant 3$, let G be the set of players only owning shared keys, let i be the number of these players and let t be the number of padlocks in a k-out-of-n threshold system. If $i = 0$, then n players own the single key of a padlock and $t \geqslant n$. If $i = 1$,

then that player has at least k new distinct and shared keys by proposition 3. Those shared keys are by definition not among the singly owned keys and thus $t = n - 1 + k \geqslant n$. More generally, if $i \leqslant k$, then $t \geqslant n - i + k \geqslant n$.

Now for $i > k$, one of the i players has at least k distinct, but shared, keys. Then the next participant has also all his shared keys not among the singly used, and at least $k - 1$ keys not shared with the previous player (otherwise their set difference is not larger than $k-1$). More generally, let $B_1, \ldots, B_h$ and A be the sets of keys of $h + 1$ distinct members of G, with $h \leqslant k$. Let $\mathcal{B} = \bigcup_{j=1}^{h} B_j$ and let $|A \setminus \mathcal{B}| = d$. Those d keys are shared (as all the keys in A) and, therefore, there exist $d' \leqslant d$ other participants with sets of keys C_j such that $(A \setminus \mathcal{B}) \subset \bigcup_{j=1}^{d'} C_j$. Let $\mathcal{C} = \bigcup_{j=1}^{d'} C_j$, then $A \cup \mathcal{B} \cup \mathcal{C} = \mathcal{B} \cup \mathcal{C}$. But then $1 + h + d' > k$: otherwise a group of k participants has the same set of keys as a group of $k - 1$. In other words, we have shown that $d' > k - 1 - h$, or more precisely that $d' \geqslant k - h$ (and thus $d \geqslant k - h$ as $d \geqslant d'$). Therefore, up to the k-th person in group G (a group of i participants with $i > k$), each person must have at least $k - h$ keys not in the sets of the h previous ones. Since $i > k$, this is at least $k + k - 1 + \sum_{h=2}^{k} (k - h) = k(k+1)/2$ keys. Then we have that the total number of keys satisfies $t \geqslant n - i + k(k+1)/2 \geqslant k(k+1)/2$. $\quad\square$

6.2. Packings, Johnson Bound and a 3-threshold Realization for up to 12 Participants with only 9 Padlocks

A sufficient condition to satisfy proposition 3 for a 3-threshold system with less than n padlocks is that a given pair of keys is never given to more than one person. Indeed, then, two persons never share a pair of keys and thus if they each own more than two keys, then their set difference is at least $2 = k - 1$.

This is thus sufficient for such a system to contain a $(2,1)-$packing, as defined thereafter:

Definition 3 (See e.g., [30])**.** *Let t, k, and p be integers with $t > k > p \geqslant 2$. Let λ be a positive integer. A $(p, \lambda)-$packing of order t, and blocksize k is a set V of t elements, and a collection B of k-element subsets (blocks) of V, so that every p-subset of V appears in at most λ blocks.*

With this, we have Johnson's bound [31], that states that a maximal packing has a number of blocks upper bounded by:

$$\left\lfloor \frac{t}{k} \left\lfloor \frac{t-1}{k-1} \right\rfloor \right\rfloor . \tag{6}$$

Equation (6) then suggests that systems with $t = \mathcal{O}(k\sqrt{n})$ padlocks might be possible.

Unfortunately, proposition 3 is probably not sufficient itself: it might be possible to fulfill its conditions while still having some set of players of size strictly lower than k having the same set of keys as some set of players of size k. However, we can at least prove that for $k = 3$ we can always use Steiner triad systems to build 3-threshold systems. A Steiner triad system is a $(2,1)$-packing with blocksize 3. In other words, it is a pair of sets such that every pair of elements of the first set appears together in a unique *triad* (or a triangle, or a triplet) of the second one [32]. As a consequence, it is possible to build a 3-threshold padlock system with only $\mathcal{O}(\sqrt{n})$ padlocks:

(1) in a Steiner triad system for a set of keys, no pair of keys is shared by two triads; therefore giving a triad of keys to each player will satisfy the necessary proposition 3;
(2) then, the following proposition 4 shows that for the particular case of $k = 3$ this is also sufficient;

(3) finally, with Johnson's bound, a Steiner triad system with $t = \mathcal{O}(\sqrt{n})$ will have sufficiently many triads to give one to each of the n players.

Proposition 4. *Any Steiner triad system gives rise to a* 3-*threshold system.*

Proof. By construction, a Steiner triad system satisfies the necessary condition of proposition 3. Second, in order to use it as a 3-threshold system, we need to differentiate triples of triads from pairs of triads (as a two participants should not be able to open the door, but three participants should). On the one hand, all triples that have 7, or more, distinct values all together, cannot be equated by pairs of triads. On the other hand, by the condition on pairs of elements being uniquely found in a single triad, triples of triads have at least 6 distinct values overall. So the only remaining case is to prove that the 6 distinct values of triples of triads with only 6 distinct values, in any construction, cannot be found in pairs of triads of the system.

To have only 6 distinct values, any two of the triple of triads must share one value, and the third one must share a value with each of the two others. W.l.o.g., this is triads $(a, b, c); (a, d, e); (b, d, f)$, with distinct values a, b, c, d, e, f. Now suppose that these 6 values are contained in a pair of triads. Then, among a, b, c, at least two of them must be in one of the pair. But by the unicity of triads containing a given pair this means that (a, b, c) *is* one of the pairs. The other pair must now be (d, e, f). But the triad (a, d, e) is in the system so the pair (d, e) is shared by two different blocks. This is a contradiction and no pair of triads can share the 6 distinct values of a triple. $\square$

Finally, by setting up a minimal Steiner system for any number of players, for instance using a Bose construction [32], we have the following algorithm 8 to setup a 3-out-of-n system. This provides an upper bound of $\mathcal{O}(\sqrt{n})$ for the number of padlocks for such a system. How to open such a system is then described in algorithm 9.

Theorem 4. *algorithm* 8 *is correct and*

$$\ell_{3,n} \leqslant \min\left\{6\left\lceil\frac{\sqrt{24n+1}-1}{12}\right\rceil + 1; 6\left\lceil\frac{\sqrt{24n+1}-5}{12}\right\rceil + 3\right\}.$$

Proof. Any construction of a Steiner triad block design works. For instance, the Bose construction [32] provides such a design for any $t = 6\nu + 3$. It is given in lines 3 to 16 of algorithm 8. proposition 4 proves that these constructions can be used as 3-threshold systems: use a 7-out-of-t design and a large DNF with all the possible groups of 6 distinct values never attained by pairs of participants. Further, the Bose construction attain the bound of Equation (6) for $t = 6\nu + 3$ and $k = 3$, that is $t/3(t-1)/2 = (2\nu+1)(3\nu+1)$. Thus for n players with $n \leqslant (2\nu+1)(3\nu+1)$ one can set up a Bose construction with $t = 6\nu + 3$ and discard the blocks between $n+1$ and $t(t-1)/6$. In other words, for a given n, use $\nu = \left\lceil\frac{\sqrt{24n+1}-5}{12}\right\rceil$ and only $t = 6\nu + 3$ padlocks. This proves that algorithm 8 is correct.

To achieve the sometimes slightly better bound of the theorem, one needs to use the Steiner triad system construction by Skolem [33, Lemma 2.5]. There use $t' = 6\mu+1$ and $t'/3(t'-1)/2 = (6\mu+1)\mu \geqslant n$, so that now $\mu = \left\lceil\frac{\sqrt{24n+1}-1}{12}\right\rceil$ and use $t' = 6\mu + 1$ padlocks. Without the ceilings, $t = t'$, for any n. But like this, t' and t are alternatively slightly better than the other (a difference of 2 or 4). $\square$

Finally, fig. 15 summarizes our current knowledge on the example of $n = 11$.

Algorithm 8 Bose Three-out-of-*n* threshold system with shared keys

---

Input: $n \geqslant 2$.

Output: A 3-out-of-*n* threshold system with $t = 6 \left\lceil \frac{\sqrt{24n+1}-5}{12} \right\rceil + 3$ padlocks.

1: Let $v = \left\lceil \dfrac{\sqrt{24n+1}-5}{12} \right\rceil$, $m = 2v + 1$ and $t = 6v + 3$;

2: Setup a 7-out-of-*t* threshold system;
  ▷ Bose construction [32]

3: **for** $x = 1..m$ **do**

4:    Give the triad of keys $< x, x + m, x + 2m$ to next player;

5: **end for**

6: **for** $x = 1..m$ **do**

7:    **for** $y = 1..m$ **do**

8:        **for** $k = 0..2$ **do**

9:            $a = x + km$;

10:           $b = y + km$;

11:           $c = ((x + y)2^{-1} - 1) \mod m$;

12:           $d = ((k + 1) \mod 3)$;

13:           Give the triad of keys $< a, b, 1 + c + dm >$ to next player;

14:       **end for**

15:    **end for**

16: **end for**
  ▷ CNF for the particular triads with only 6 distinct values

17: $h = 0$;

18: **for** each triple of triads of keys **do**

19:    **if** this triple of triads contains only 6 distinct keys **then**

20:        Setup a conjonctive clause with these 6 values;

21:        increment h;

22:    **end if**

23: **end for**
  ▷ Setup a DNF via algorithm 4

24: Setup a 1-out-of-$(h + 1)$ device   ▷ one latch for each conjonction, plus an additional one

25: **for** each of the *t* padlocks **do**

26:    Pass a chain through the hole of each latch corresponding to a conjunction containing that padlock;

27:    Pass the chain through the hole of one free latch of the 7-out-of-*t* device;

28:    Close the chain with that padlock.

29: **end for**

30: Attach the 7-out-of-*t* device to remaining latch of the 1-out-of-$(h + 1)$ device.

6.3. 3-threshold realizations with fewer than n padlocks

We give the smallest example realizing proposition 4: a 3-out-of-12 system (thus also a 3-out-of-11 system), with only 9 padlocks, 36 keys and 82 latches, and an example using normal forms to reduce the number of latches for a 3-out-of-13 system with only 11 padlocks, 36 keys and 33

Algorithm 9 Opening the device of algorithm 8

Input: $n \geqslant 2$;
Input: A 3-out-of-n device made by algorithm 8;
Input: The 3 sets of keys give to 3 participants by algorithm 8.
Output: The system is opened.
  1: **if** the 3 users own more than 7 distinct keys **then**
  2:     They open the 7-out-of-t device, which then opens the 1-out-of-$(h + 1)$ DNF.
  3: **else**   ▷ Thus they own one of the h groups of 6 distinct keys
  4:     They open their padlocks, this frees one clause of the DNF.
  5: **end if**
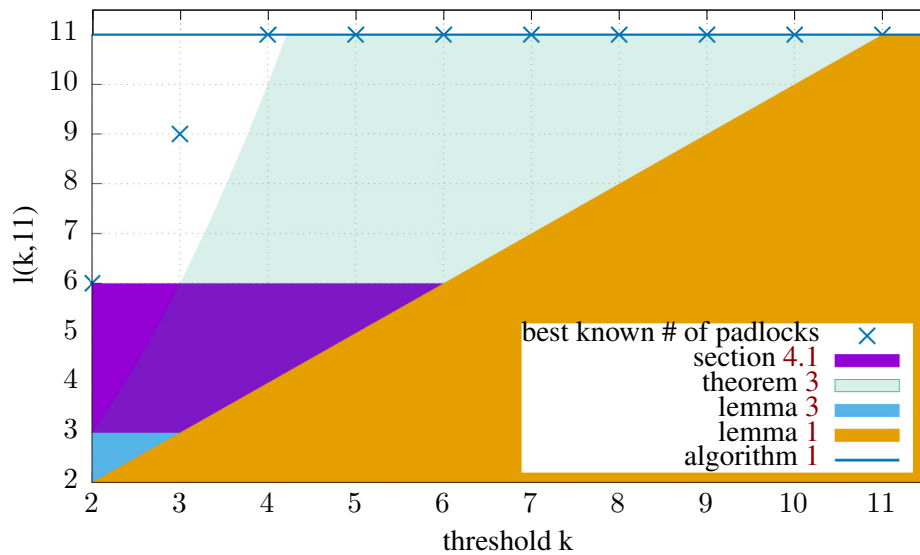
Fig. 15. n=11: number of padlocks for known k-out-of-11 algorithms (only the points above the regions are attainable).

latches. Indeed, consider the first terms of Equation (6) for $k = 3$ and $t = 0, 1, 2, \ldots$, this is: $0, 0, 0, 1, 1, 3, 4, 7, 8, 12, 13, 18, 20, 26, 28, \ldots$ [34].

6.3.1. An example realization of algorithm 8

The smallest t such that Equation (6) is strictly larger than t is for $t = 9$ with a bound of 12 subsets. Hence, packing with 9 padlocks is realizable, for instance with the Steiner triads of keys given in table 1.

Table 1

A maximal $(2, 1)-$ packing of order 9 and blocksize 3. It has 12 blocks.

1	2	3	1	4	3	1	4	2	2	5	1
4	5	6	2	5	7	3	6	7	3	6	8
7	8	9	6	9	8	5	8	9	4	7	9

By inspection, there are 72 triples of triads (so 3 participants owning each 3 keys) with only 6 distinct keys (for instance the triads $< 1, 2, 3 >$, $< 1, 4, 8 >$, $< 2, 4, 7 >$). The 148 other triples of triads have at

least 7 distinct keys (if a triple have a total of less than 5 distinct keys it would mean that at least two of them share a pair). But proposition 4 shows that none of the 72 sets of six keys obtained with three triads can be obtained with only a pair of triads (for instance the triple $(1, 2, 3); (1, 4, 8); (2, 4, 7)$ contains only the six distinct keys $1, 2, 3, 4, 7, 8$, but no pair of triples contain the same set of keys). The latter ensures that no subset of 2 participants can unlock the door. Further, all these 72 sets of 6 keys are distinct.

Therefore, it is possible to set up a 3-out-of-12 system using only 9 padlocks. The idea of algorithm 8 is that either a group owns 7 distinct keys or it owns one of the 72 sets of 6 keys not reachable by a pair of participants. Overall, that solution uses 9 padlocks, 9 chains, 36 keys, a 7-out-of-9 and a 1-out-of-73 design (that is $9 + 73 = 82$ latches). The following process gives the instance of algorithm 8 for this system:

(1) Set up 9 padlocks and make 4 copies of each key;
(2) Give 3 keys to each of the 12 participants following the packing of table 1;
(3) Set up a 1-out-of-73 design;
(4) Set up a 7-out-of-9 design and attach it to one the latches of the $1 - 73$ design;
(5) Use algorithm 4 to complete the 72 other latches: pass a chain through the hole of each latch corresponding to a disjunction containing that key; close that chain with the associated padlock.

6.3.2. *The bound of theorem 4 is not enough*

Next, we give a small example where there exists a shortcut to use less latches than with the latter construction. We use some results of section 5 to help for the construction. For 13 participants, theorem 4 would provide a system with either $15 = 6 \left\lceil \frac{\sqrt{24*13+1}-5}{12} \right\rceil + 3$ or $13 = 6 \left\lceil \frac{\sqrt{24*13+1}-1}{12} \right\rceil + 1$ padlocks. This is already not better than 13 padlocks, directly attainable with our 3-out-of-13 device.

But we even show next a 3-threshold system for 12 or 13 participants with only 11 padlocks, 36 or 39 keys and only 5 additional devices for a total of 33 latches. We give in table 2, afterwards, a realization of a packing with 3-subsets. Then we proceed by inspection of the triples and pairs of triads of keys. There are $\binom{13}{3} = 286$ triples of triads and among them 56 have only 6 distinct keys. All the other triples

Table 2

Distribution of 11 keys to 13 participants without any reused pair.

Player	1	2	3	4	5	6	7	8	9	10	11	12	13
	1	1	1	1	1	2	2	2	2	3	3	3	3
keys	2	4	6	8	10	4	5	8	9	4	5	8	9
	3	5	7	9	11	6	7	10	11	7	6	11	10

have at least 7 distinct keys. Also, there are $\binom{13}{2} = 78$ pairs of triads and among them 24 have exactly 6 distinct keys. All the other pairs have at most 5 distinct keys. Further, on the one hand, all those 24 pairs contain no more and no less than 2 keys among $8, 9, 10, 11$. On the other hand, among the 56 triples either they contain more than 3 keys among $8, 9, 10, 11$ or their 6 distinct keys are lower than 7. This is summarized by Equation (7).

$$(7 \text{ out-of } 1 \ldots 11) \quad \textbf{OR}$$
$$(\quad (6 \text{ out-of } 1 \ldots 11) \quad \textbf{AND} \quad ((3 \text{ out-of } 8 \ldots 11) \textbf{ OR } (5 \text{ out-of } 1 \ldots 7))) \quad (7)$$

So, by luck, the following construction realizes a 3-threshold system for 13 participants with 11 padlocks. We need a 7-out-of-11 device as well as a 6-out-of-11, a 5-out-of-7 and a 3-out-of-4 of our designs. Finally a classical 2-out-of-2 device is needed for the **AND** part. All of these are organized as follows, in order to realize the formula of Equation (7).
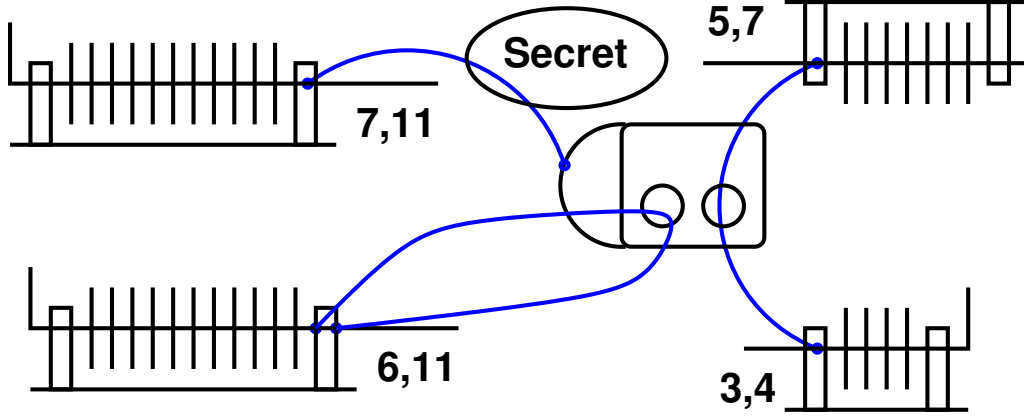

Fig. 16. A 3-threshold realization for 13 participants with 11 padlocks. Each participant owns 3 keys with the distribution of table 2. On the one hand, any 3 participants have either at least 7 distinct keys or if they have only 6 keys then they have at least 5 for padlocks numbered 1 to 7 or at least 3 for padlocks numbered 8 to 11. On the other hand, no pair of participants has a total of 6 distinct keys and either 5 of the first seven ones or 3 for the last four ones.

Each of the eleven padlocks is used once to close a chain as in algorithm 4. For each padlock its associated chain will go through the hole of each of up to the four devices (the devices $(7, 11)$ and $(6, 11)$ have each 11 latches so are linked to all the padlock; while device $(5, 7)$ is for the padlocks numbered 1 to 7 and device $(3, 4)$ is for the padlocks numbered 8 to 11). This will realize the disjunctions **OR** in Equation (7). Finally, the disjunction of the devices $(5, 7)$ and $(3, 4)$ is linked via a chain, and that together with the $(6, 11)$ device are associated via a 2-out-of-2 device, as in algorithm 5. The whole system is shown in fig. 16. Overall, it requires fewer padlocks, but quite a bunch of other devices.

The same system works also for a 3-threshold realization for 12 participants with 11 padlocks. Just use the 12 first triads of keys of table 2 with the same system. Yet this solution uses more padlocks than algorithm 8.

fig. 17 summarizes what we know for 3-out-of-n systems. We see that for a threshold of three the minimal number of padlocks is in between $\mathcal{O}(\log(n))$ and $\mathcal{O}(\sqrt{n})$.

7. A recursive asymptotic construction

For a larger number of participants, asymptotically, one can reduce the number of padlocks by making subgroups. For instance, consider building a 3-out-of-n system. Create two subgroups G and H of $n_0 = \left\lceil \frac{n}{2} \right\rceil$ and $n_1 = \left\lfloor \frac{n}{2} \right\rfloor$ participants. Setup a 3-out-of-n_0 system for the participants of the subgroup G. Then duplicate all the distributed keys and give them to the members of subgroup H, except potentially for the one supernumerary member of subgroup G. Then any 3 participants all in one of the two subgroups can open the system. Only some triads where one participant is in one subgroup, and the two others in the other subgroup, cannot open the system yet. But then, for these cases, we can build a conjunction of a 1-out-of-n_i system with a 2-out-of-n_{1-i} system. Any triad of participants are either in a single subgroup
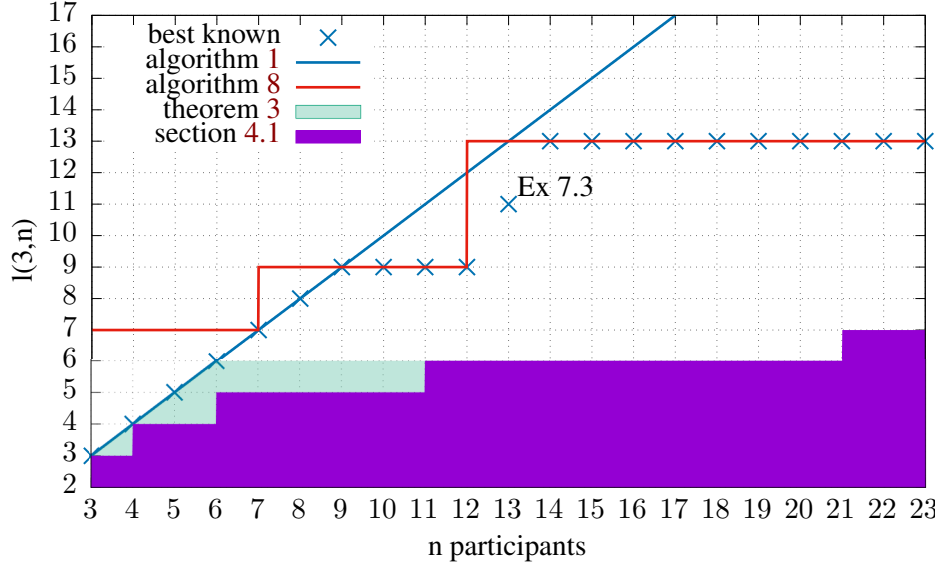
Fig. 17. k=3: number of padlocks for known 3-out-of-n algorithms (only the points below the algorithms lines and above the regions could use less padlocks).

or in a one and two configuration and can open the system. Now any single or pair of participants cannot open the 3-out-of-n_0 system, nor both of the 1-out-of-n_i and 2-out-of-n_{1-i} systems. Denote by $\mathcal{T}(3,n)$ the 3-out-of-*n* systems, we have thus proven that:

$$\forall n_0 \geqslant n_1 \geqslant 3, n_0 + n_1 = n, \mathrm{OR}(\ \mathcal{T}(3,n_0)\,;$$
$$\mathrm{AND}\,(\mathcal{T}(1,n_0)\,;\mathcal{T}(2,n_1))\,;$$
$$\mathrm{AND}\,(\mathcal{T}(2,n_0)\,;\mathcal{T}(1,n_1))$$
$$)\in\mathcal{T}(3,n) \tag{8}$$

To count the number of keys and padlocks, we first need Faulhaber's formula:

$$\sum_{j=1}^{s}j^k = \frac{1}{k+1}s^{k+1} + O(s^k) \tag{9}$$

Then we need the following formula:

$$\sum_{k=0}^{i}(-1)^{i-k}\binom{i}{k}\frac{1}{i-k+1} = \frac{1}{i+1}\sum_{k=0}^{i}(-1)^{i-k}\binom{i+1}{k}$$
$$= \frac{1}{i+1}\left(-(-1)^{-1}\binom{i+1}{i+1} + \sum_{k=0}^{i+1}(-1)^{i-k}\binom{i+1}{k}\right) \tag{10}$$
$$= \frac{1}{i+1}(1 + (1-1)^{i+1}) = \frac{1}{i+1}.$$

Finally, we need the following variant of the master theorem.

Lemma 6. *For $n \in \mathbb{N}$ and $i \in \mathbb{N}^*$, let $T(n)$ be a function defined by the recurrence $T(n) = T(\lceil n/2 \rceil) + c \left(\log_2 \left(\frac{n}{2} \right) \right)^i + o\left(\log(n)^i \right)$. Then $T(n) = \frac{c}{i+1} \log_2(n)^{i+1} + o\left(\log(n)^{i+1} \right)$.*

Proof. Expanding $log_2(n)$ times the recurrence, we obtain:

$$T(n) = T(1) + \sum_{j=1}^{\log_2 n} c \left(\log_2 \left(\frac{n}{2^{j+1}} \right) \right)^i + o\left(\left(\log \left(\frac{n}{2^j} \right) \right)^i \right)$$

$$= c \sum_{j=1}^{\log_2 n} \left(\sum_{k=0}^{i} (-1)^{i-k} \binom{i}{k} \log_2(n)^k j^{i-k} \right) + o\left(\log(n)^i \right) \qquad (11)$$

$$= c \sum_{k=0}^{i} \left((-1)^{i-k} \binom{i}{k} \log_2(n)^k \sum_{j=1}^{\log_2 n} j^{i-k} \right) + o\left(\log(n)^i \right)$$

Using Equation (9), with $s = \log_2(n)$, this is:

$$T(n) = c \sum_{k=0}^{i} \left((-1)^{i-k} \binom{i}{k} \log_2(n)^k \frac{1}{i-k+1} \log_2(n)^{i-k+1} \right) + O\left(\log(n)^i \right)$$

$$= c \log_2(n)^{i+1} \sum_{k=0}^{i} \left((-1)^{i-k} \binom{i}{k} \frac{1}{i-k+1} \right) + O\left(\log(n)^i \right) \qquad (12)$$

Finally, with Equation (10), we have that:

$$T(n) = \frac{c}{i+1} \log_2(n)^{i+1} + o\left(\log(n)^{i+1} \right) \qquad (13)$$

$\square$

With these, we can now count padlocks and keys for the strategy with two subgroups of Equation (8):

Lemma 7. *For $n \geqslant 6$,*

$$\ell_{3,n} \leqslant 2 \log_2(n)^2 + o\left(\log(n)^2 \right)$$

and the upper bound is attained with an average of $\frac{1}{2} \log_2(n)^2 + o\left(\log(n)^2 \right)$ keys per participant.

Proof. To realize Equation (8) we need 1 padlock for $\mathcal{T}(1, n_0)$ and another one for $\mathcal{T}(1, n_1)$. We also need less than $2\lceil \log_2(n_0) \rceil$ padlocks for $\mathcal{T}(2, n_0)$ and similarly $2\lceil \log_2(n_0) \rceil$ padlocks for $\mathcal{T}(2, n_1)$, using proposition 1. finally, $\mathcal{T}(3, n_0)$ is realized recursively. Therefore the number of padlocks for $\mathcal{T}(3, n)$ satisfies $P_3(n) \leqslant P_3(n_0) + 2 + 4 \log_2(n/2) + O(1)$. Lemma 6 then gives $P_3(n) = \frac{4}{2} \log_2(n)^2 + o\left(\log(n)^2 \right)$.

Similarly the participants of subgroup G get 1 key for $\mathcal{T}(1, n_0)$ and $\log_2(n_0)$ keys for $\mathcal{T}(2, n_0)$. The participants in the other subgroup get 1 key for $\mathcal{T}(1, n_1)$ and $\log_2(n_1)$ keys for $\mathcal{T}(2, n_1)$. Then they each get the keys needed for $\mathcal{T}(3, n_0)$. Thus the average number of keys per participant satisfies $K_3(n) \leqslant K_3(n/2) + 1 + \log_2(n/2) + O(1)$. Lemma 6 then gives $K_3(n) = \frac{1}{2}\log_2(n)^2 + o\big(\log(n)^2\big)$. $\quad\square$

Now, this scheme can be generalized for any threshold k as shown in algorithm 10.

Algorithm 10 Recursive k-out-of-n threshold system with shared keys

---

Input: $n \geqslant k \geqslant 2$.

Output: A recursively build k-out-of-n threshold padlock system.

1: **if** $k == 2$ **then**
2:     **return** algorithm 2.
3: **else if** $k(k+1)/2 \geqslant n$ **then**
4:     **return** a k-out-of-n system with n padlocks.   ▷ theorem 3
5: **else**
6:     Let $n_0 = \left\lceil \frac{n}{2} \right\rceil$ and $n_1 = \left\lfloor \frac{n}{2} \right\rfloor$;
7:     Separate the participants in two groups G and H with n_0 and n_1 members;
8:     Recursively setup a k-out-of-n_0 system;
9:     Duplicate the keys of this system and distribute one set to members of G and the other set to members of H;
10:     Setup an 1-out-of-k OR system without padlocks nor keys;
11:     Attach the k-out-of-n_0 system to one of the latches of the OR system;
12:     **for** $i = 1..k-1$ **do**
13:         Setup a 2-out-of-2 AND system and attach it to the OR system
14:         Setup a i-out-of-n_0 system for members of the subgroup G and attach it to this AND system;
15:         Setup a $(k-i)$-out-of-n_1 system for members of the subgroup H and attach it to this AND system;
16:     **end for**
17:     **return** the OR system openable either via the k-out-of-n_0 system or via one of the $(k-1)$ AND Systems.
18: **end if**

Theorem 5. *algorithm 10 is correct and asymptotically requires*

$$\begin{cases} \frac{2^{k-1}}{(k-1)!}\log_2(n)^{k-1} + o\big(\log(n)^{k-1}\big) \text{ padlocks} \\ \frac{1}{(k-1)!}\log_2(n)^{k-1} + o\big(\log(n)^{k-1}\big) \text{ keys per participants} \end{cases} \tag{14}$$

Proof. For the correctness, consider a group of at most $k-1$ participants. They cannot open the k-out-of-n_0 system. Then they are distributed with $j \in 0..k-1$ of them in group G and $k-1-j$ in group H. They can thus open any of the $\mathcal{T}(\alpha, n_0)$ for $\alpha = 0..j$, but none of the corresponding $\mathcal{T}(k-\alpha, n_1)$ since $k - \alpha > k - 1 - j$. They can also open any of the $\mathcal{T}(\beta, n_1)$ for $\beta = 0..k-1-j$, but none of the corresponding $\mathcal{T}(k-\beta, n_0)$ since $k - \beta > j$. So they can never open the system. On the contrary, consider a group of at least k participants. They are distributed with $j \in 0..k$ in group G and at least $k-j$

in group H. Thus they can either open the k-out-of-n_0 system or one of the $\mathcal{T}(j, n_0)$ AND $\mathcal{T}(k-j, n_1)$ group.

Now, for the complexity bound, we proceed by induction on $k \leqslant n$. The formulae are correct for $k = 3$ by lemma 7. Now suppose that the formulae are correct $\forall i \leqslant k$ and consider algorithm 10 at $k+1$. Then the number of padlocks used by the Algorithm is $P_{k+1}(n) = P_{k+1}(n_0) + \sum_{i=1}^{k} P_i(n_0) + P_{k+1-i}(n_1)$ and the average number of keys per participant is $K_{k+1}(n) = K_{k+1}(n_0) + \frac{1}{2}\left(\sum_{i=1}^{k} K_i(n_0) + K_{k+1-i}(n_1)\right)$. By the induction hypothesis, the number of padlocks thus satisfy:

$$
\begin{aligned}
P_{k+1}(n) \leqslant &P_{k+1}(n/2) + \sum_{i=1}^{k}\left(\frac{2^{i-1}}{(i-1)!}\log_2(n/2)^{i-1} + \frac{2^{k-i}}{(k-i)!}\log_2(n/2)^{k-i}\right) \\
&+ o\left(\log(n)^{k-1}\right) \\
= &P_{k+1}(n/2) + 2\frac{2^{k-1}}{(k-1)!}\log_2(n/2)^{k-1} + o\left(\log(n)^{k-1}\right);
\end{aligned}
\tag{15}
$$

and the number of keys satisfies:

$$
\begin{aligned}
K_{k+1}(n) \leqslant &K_{k+1}(n/2) + \frac{1}{2}\left(\sum_{i=1}^{k}\frac{1}{(i-1)!}\log_2(n/2)^{i-1}\right. \\
&\left. + \frac{1}{(k-i)!}\log_2(n/2)^{k-i}\right) + o\left(\log(n)^{k-1}\right) \\
= &K_{k+1}(n/2) + \frac{2}{2}\frac{1}{(k-1)!}\log_2(n/2)^{k-1} + o\left(\log(n)^{k-1}\right).
\end{aligned}
\tag{16}
$$

Finally, lemma 6, applied on both relations shows that:

$$
P_{k+1}(n) = \frac{2^k}{(k-1)!k}\log_2(n)^{k-1+1} + o\left(\log(n)^k\right)
\tag{17}
$$

$$
K_{k+1}(n) = \frac{1}{(k-1)!k}\log_2(n)^{k-1+1} + o\left(\log(n)^k\right)
\tag{18}
$$

These establish that the hypothesis is true for $k+1$. Therefore it is inductive and the theorem is proven. □

Note that algorithm 10 is useful only for a large number of participants. For instance with a threshold of three, $\lceil 2\log_2(n)^2\rceil$ is lower than theorem 4 only for $n \geqslant 33\,922$. This is overestimated, but, more precisely, the smallest case where algorithm 10 yields less padlocks than algorithm 8 is only at $n = 1248$. There, we have 63 padlocks for a 3-out-of-624 system with algorithm 8, then 12 padlocks for a 2-out-of-624 system via algorithm 2. This is a total of $63 + 2(1 + 12) = 89$ padlocks where algorithm 8 alone yields 91 padlocks. Now, for $k = 4$, and using theorem 4 when $i = 3$, the smallest case where algorithm 10 yields less than n padlocks is at $n = 114$. There, we have 57 padlocks for a 4-out-of-57 system, then 8 padlocks for a 2-out-of-57 system via algorithm 2 and 19 padlocks for a 3-out-of-57

system via algorithm 8. This is a total of $57 + 2(1 + 8 + 19) = 113$ padlocks for a 4-out-of-114 system via algorithm 10.

8. Secret sharing with reduced field size

Consider Shamir's secret sharing via interpolation over a finite field. For a secret value within a finite field $\mathbb{F}_q$, set it as the evaluation at zero of a degree $k-1$ polynomial whose other coefficient are randomly sampled. Then distribute an evaluation of the polynomial at distinct non-zero points to n participants. This is a k-out-of-n threshold system. It requires that there are enough evaluation points for all the participants and thus that $q > n$.

We in fact have shown that this is optimal in certain cases, but that one can use smaller fields in others: instead of a degree k polynomial, use a degree t polynomial, where t is the number of padlocks in one of our systems. This number of padlocks t is in fact the number of available evaluation points. Then the identical keys for a given padlock are the evaluations of the polynomial at the points. Thus participants have several evaluations instead of a single one. We have thus proposed a k-out-of-n secret sharing scheme where the field size is reduced. For instance, from theorem 5, if $n \geqslant 2k$, then it is sufficient to take the field size $q = O\big(\log(n)^{k-1}\big)$.

9. Conclusion

We designed a physical k-out-of-n threshold lock that can be used for various applications, including physical access control, voting or secret sharing. Our system only uses n padlocks, showing that previous exponential answers to Liu's problem were far too pessimistic. For $k = 2$, we were even able to identify an optimal solution using our device, which needs less than $2\lceil \log_2(n) \rceil$ padlocks, but requires duplicating keys. We also show that for $k \geqslant \sqrt{2n}$ the minimal number of padlocks is n (and our device also reaches this).

There are many open questions left, for example we have shown that reducing the number of padlocks is equivalent to reducing the size of the fields for interpolation-based secret sharing, but further exploration of the links with digital systems could be envisioned. Another future work is to find minimal solutions in terms of padlocks for small cases, in particular for k between 3 and $\sqrt{2n}$. For instance, when $k \geqslant 3$, Johnson's bound suggests that it might be possible to build systems with only $\mathcal{O}(k\sqrt{n})$ padlocks and we were able to prove this for $k = 3$.

We also devised algorithms that can implement more complex access policies beyond simple thresholds, expressed as disjunctive or conjunctive Boolean formulas. It is yet unclear for us whether there are general solutions using less locks than the number of variables.

We proposed one variant using sealed wire and wrappings to provide an alternative solution to our device with exactly n padlocks. The threshold systems we found with this approach unfortunately use an exponential number of wrappings. It is unclear to us if this could be improved.

Differently, on the asymptotic side, we have found an algorithm, recursively combining several of our devices, requiring only $O\big(log(n)^{k-1}\big)$ padlocks for k-out-of-n threshold systems but we have only a lower bound of $O(log(n))$.

Finally, if we do not only count the number of padlocks, but more generally the number of keys or of latches, then clearly a lower bound on the number of devices is n: each player must at least have something. Otherwise groups of k players with an empty player would have the same abilities of a group

of $k - 1$ players. With this model of complexity, our k-out-of-n designs are asymptotically optimal as they require just n padlocks, n latches and n keys.

References

[1] A. Shamir, How to Share a Secret, *Commun. ACM* **22**(11) (1979), 612–613.

[2] C.L. Liu, *Introduction to combinatorial mathematics*, McGraw-Hill New York, 1968, p. x, 393 p. :. ISBN 0070381240.

[3] A.V. Aho and J.E. Hopcroft, *The Design and Analysis of Computer Algorithms*, 1st edn, Addison-Wesley Longman Publishing Co., Inc., USA, 1974. ISBN 0201000296.

[4] D.E. Knuth, *The Art of Computer Programming, Volume 2 (3rd Ed.): Seminumerical Algorithms*, Addison-Wesley Longman Publishing Co., Inc., USA, 1997. ISBN 0201896842.

[5] B. Chor, S. Goldwasser, S. Micali and B. Awerbuch, Verifiable Secret Sharing and Achieving Simultaneity in the Presence of Faults, in: *Proceedings of the 26th Annual Symposium on Foundations of Computer Science*, SFCS '85, USA, 1985, pp. 383–395–. ISBN 0818608444.

[6] P. Feldman, A Practical Scheme for Non-Interactive Verifiable Secret Sharing, in: *Proceedings of the 28th Annual Symposium on Foundations of Computer Science*, SFCS '87, IEEE Computer Society, USA, 1987, pp. 427–438–. ISBN 0818608072.

[7] NIST, Threshold Schemes for Cryptographic Primitives: Challenges and Opportunities in Standardization and Validation of Threshold Cryptography, *Federal Register* **69**(8214) (2019), https://csrc.nist.gov/Projects/threshold-cryptography.

[8] Y. Desmedt, Threshold Cryptography, in: *Encyclopedia of Cryptography and Security*, H.C.A. van Tilborg and S. Jajodia, eds, Springer US, 2011, pp. 1288–1293. ISBN 978-1-4419-5906-5.

[9] M. Naor and A. Shamir, Visual Cryptography, in: *EUROCRYPT '94, Workshop on the Theory and Application of Cryptographic Techniques*, 1994, pp. 1–12.

[10] Everlock, System SLX2, 2019, https://vimeo.com/180052349.

[11] Tayhope, Multi-Locking Systems, 2020, http://sancy.univ-bpclermont.fr/~lafourcade/VIDEOS/secu/tayhope_voiceover.mp4.

[12] R. McNeil, Multiple padlock latch, 2003.

[13] R. McNeil, Multiple padlock lock system, 2008.

[14] R. McNeil, Multiple padlock locking device, 2014.

[15] R. McNeil, Multiple padlock locking system, 2015.

[16] M. Naor and A. Shamir, Visual Cryptography II: Improving the Contrast Via the Cover Base, in: *Security Protocols, International Workshop, Cambridge, United Kingdom, April 10-12, 1996, Proceedings*, 1996, pp. 197–202.

[17] C. Blundo, A.D. Santis and M. Naor, Visual cryptography for grey level images, *Inf. Process. Lett.* **75**(6) (2000), 255–259.

[18] Y.-C. Hou, Visual cryptography for color images, *Pattern Recognition* **36**(7) (2003), 1619–1629.

[19] B.W. Leung, F.Y. Ng and D.S. Wong, On the security of a visual cryptography scheme for color images, *Pattern Recognition* **42**(5) (2009), 929–940.

[20] E.R. Verheul and H.C.A. Van Tilborg, Constructions and Properties of k-out-of-N Visual Secret Sharing Schemes, *Des. Codes Cryptography* **11**(2) (1997), 179–196–.

[21] G. Horng, T. Chen and D.-S. Tsai, Cheating in Visual Cryptography, *Des. Codes Cryptography* **38**(2) (2006), 219–236–.

[22] A. Beimel, T. Tassa and E. Weinreb, Characterizing Ideal Weighted Threshold Secret Sharing, in: *Proceedings of the Second International Conference on Theory of Cryptography*, TCC'05, Springer-Verlag, Berlin, Heidelberg, 2005, pp. 600–619–. ISBN 3540245731.

[23] O. Farras and C. Padro, Ideal Hierarchical Secret Sharing Schemes, *IEEE Transactions on Information Theory* **58**(5) (2012), 3273–3286.

[24] B. Schoenmakers, A simple publicly verifiable secret sharing scheme and its application to electronic voting, in: *CRYPTO'99*, Springer-Verlag, 1999, pp. 148–164.

[25] E. Sperner, Ein Satz über die Untermengen einer endlichen Menge, *Mathematische Zeitschrift* **27** (1928), 544–548.

[26] M. Ito, A. Saito and T. Nishizeki, Multiple Assignment Scheme for Sharing Secret, *J. Cryptol.* **6**(1) (1993), 15–20. doi:10.1007/BF02620229.

[27] J. Benaloh and J. Leichter, Generalized Secret Sharing and Monotone Functions, in: *CRYPTO'88*, S. Goldwasser, ed., New York, NY, 1990, pp. 27–35. ISBN 978-0-387-34799-8.

[28] T.D. Ahle, K out of N encryption, 2012, accessed 08/01/2021. https://crypto.stackexchange.com/questions/2783/k-out-of-n-encryption.

[29] R. Smolensky, Algebraic Methods in the Theory of Lower Bounds for Boolean Circuit Complexity, in: *Proceedings of the Nineteenth Annual ACM Symposium on Theory of Computing*, STOC '87, Association for Computing Machinery, New York, NY, USA, 1987, pp. 77–82. ISBN 0897912217. doi:10.1145/28395.28404.

[30] Y.M. Chee, C.J. Colbourn, A.C.H. Ling and R.M. Wilson, Covering and packing for pairs, *Journal of Combinatorial Theory, Series A* **120**(7) (2013), 1440–1449.

[31] S.M. Johnson, A new upper bound for error-correcting codes, *IRE Transactions on Information Theory* **8**(3) (1962), 203–207.

[32] R.C. Bose, On the construction of balanced incomplete block designs, *Annals of Eugenics* **9**(4) (1939), 353–399.

[33] C.J. Colbourn, A. Rosa et al., *Triple systems*, Oxford University Press, 1999.

[34] OEIS, The On-Line Encyclopedia of Integer Sequences, 2020, https://oeis.org/A182079.