Designing for Mistrust

Eric Gossett, Department of Mathematics and Computer Science,
Bethel University, St. Paul, MN 5512, USA, gossett@bethel.edu

Abstract

The 2014 ACM North Central Region programming contest contained a problem about a group of v bandits who want to use multiple locks to seal their treasure and distribute keys in such a way that no group of less than m bandits can open all the locks. The problem asks for an algorithm that will determine the number of locks needed for any set of parameters (v, m).

I will present an analytic solution that produces a minimum number of locks, a recurrence relation solution, and a constructive algorithm that can print out a table showing the locks and which subset of bandits hold keys for each lock. Each table forms a balanced incomplete block design (BIBD). The parameters of the BIBD can be uniquely determined from v and m.

Locked Treasure

One of the problems from the 2014 ACM North Central Region programming contest was the following (with a change of variable names) [1].

Problem (ACM North Central Region)**.** *A group of v $(1 \leq v \leq 30)$ bandits hid their stolen treasure in a room. The treasure needs to be locked away until there is a need to retrieve it. Since the bandits do not trust each other, they wanted to ensure that at least m $(1 \leq m \leq v)$ of the bandits must agree in order to retrieve the treasure.*

They have decided to place multiple locks on the door such that the door can be opened if and only if all the locks are opened. Each lock may have up to v keys, distributed to a subset of the bandits. A group of bandits can open a particular lock if and only if someone in the group has a key to that lock.

Given v and m, how many locks are needed such that if the keys to the locks are distributed to the bandits properly, then every group of bandits of size at least m can open all the locks, and no smaller group of bandits can open all the locks?

For example, if $v = 3$ and $m = 2$, only 3 locks are needed — keys to lock 1 can be given to bandits 1 and 2, keys to lock 2 can be given to bandits 1 and 3, and keys to lock 3 can be given to bandits 2 and 3. No single bandit can open all the locks, but any group of 2 bandits can open all the locks.

Terminology

Assume that the values of v and m have been given.

Definition 1 (Conforming Solution)**.** *A set of locks and a distribution of keys for the locks is called a* conforming solution *if every set of m bandits can open every lock and no smaller set of bandits can open every lock.*

Definition 2 (Mistrust Lock; Mistrust Design)**.** *A lock is called a* mistrust lock *if there is a group of exactly* $m-1$ *bandits who* do not *have a key for that lock, and all of the other* $v-m+1$ *bandits* do *have a key for that lock.*

The unordered collection of all $\binom{v}{m-1}$ *distinct mistrust locks is called a* mistrust design*. This design will be denoted by* $M_{v,m}$*.*

Example 1 ($M_{5,3}$)**.** *In the chart below, the rows represent bandits, the columns represent the locks. Bandit* i *has a key for lock* L_j *if there is an X in* i*th row,* j*th column. The chart shows the mistrust design* $M_{5,3}$*.*

$M_{5,3}$	L_1	L_2	L_3	L_4	L_5	L_6	L_7	L_8	L_9	L_{10}
1	X			X		X	X		X	X
2		X			X	X		X	X	X
3			X	X	X		X	X		X
4	X	X	X				X	X	X	
5	X	X	X	X	X	X				

It will be convenient to mention two important theorems about the binomial coefficients $\binom{n}{r}$. Recall that for $n \geq 0$ and $0 \leq r \leq n$, $\binom{n}{r} = \frac{n!}{r!\cdot(n-r)!}$ and $\binom{n}{r} = 0$ if $0 \leq n < r$. The theorems are easy to prove algebraically and also by using simple combinatorial proofs.

Theorem 1. *For all nonnegative integers* n *and* r *with* $r \leq n$

$$\binom{n}{r} = \binom{n}{n-r}$$

Theorem 2 (Pascal's Theorem)**.** *For all positive integers* n *and* r *with* $r \leq n$

$$\binom{n}{r} = \binom{n-1}{r} + \binom{n-1}{r-1}$$

Proofs of both theorems can be found in [2].

An analytic solution

A solution can be found by considering subsets of bandits of size $m-1$. For each distinct subset, D_i, of size $m-1$, create a lock for which none of the bandits in that set have a key, but for which every other bandit *does* have a key. There are $\binom{v}{m-1}$ such subsets. Denote the lock associated with subset D_i as L_i. Thus, D_i consists of all the bandits who *do not* have a key to L_i.

Proposition 1. *At least* $v-m+1$ *bandits must each have a key for any lock which is part of a conforming solution.*

Proof. If fewer bandits each *have* a key, then at least $v-(v-m)=m$ bandits will *not* have a key. That means there is a set of m bandits who cannot open the lock, so the lock is not part of a conforming solution.

□

Definition 3 (Minimal Conforming Solution)**.** *A set of locks and a distribution of keys for the locks is called a* minimal conforming solution *if it is a conforming solution, no smaller set of locks is a conforming solution, and every lock has exactly $v - m + 1$ keys.*

Proposition 2. *A minimal conforming solution will not have any repeated locks (locks with keys distributed to the same set of bandits).*

Proof. Nothing is gained by repeating a lock.

□

Theorem 3 (Mistrust Designs)**.** *Suppose a set of v bandits wish to use the mistrust design $M_{v,m}$ to seal their treasure. Then*

- *Every subset of m of the bandits is able to open all the locks*
- *No subset of $m - 1$ or fewer bandits will be able to open all the locks.*

Proof. Notice that no lock has more than $m - 1$ bandits who do not have a key to that lock. This means that in any subset of m bandits and for each lock, at least one bandit will have a key to the lock.

Now consider any set of $m - 1$ bandits. That subset will be the associated subset D_i for some lock L_i. None of those bandits can open lock L_i. Clearly, no subset of fewer than $m - 1$ bandits will fare any better than a subset of size $m - 1$.

□

Theorem 4 (A Minimal Set Of Locks)**.** *Suppose a set of v bandits wish to use multiple locks to seal their treasure in such a way that every subset of m of the bandits is able to open all the locks, but no smaller subset can do so. Then*

- *It is possible to ensure this with a set of $\binom{v}{m-1}$ locks.*
- *No set of fewer than $\binom{v}{m-1}$ locks will accomplish this.*

Proof. The first claim can be achieved by using the mistrust design $M_{v,m}$ (see Theorem 3.)

Suppose now that a set, L, of fewer than $\binom{v}{m-1}$ locks will be used. Proposition 1 ensures that at least $v-m+1$ bandits must each have a key for any chosen lock. Thus, at most $m-1$ bandits will not have a key for any lock.

For each lock $L_i \in L$, denote by N_i the subset of bandits who *do not* have a key for that lock. So $|N_i| \leq m-1$ for all i. Since there are fewer than $\binom{v}{m-1}$ locks, there must be at least one subset of bandits having $m - 1$ members that does not equal N_i for any i. (If $|N_i| < m - 1$ it cannot equal any subset of $m - 1$ bandits. Even if $|N_i| = m - 1$ for all i, there are not enough locks to include every subset of $m - 1$ bandits.) Denote one such missing subset as T. The bandits in T can open every one of the locks in L, since at least one of the bandits in T is missing from N_i for all i. That is, for all i, at least one of the bandits in T has a key for L_i.

Since a subset of $m - 1$ bandits can open all the locks in L, L cannot be a conforming solution to the problem. The failure arose from the assumption that L has fewer than $\binom{v}{m-1}$ locks, so any conforming solution must contain at least $\binom{v}{m-1}$ locks.

□

Corollary 1. *For a given v and m, the only minimal conforming solution is a mistrust design.*

Proof. Any conforming solution must contain $\binom{v}{m-1}$ locks. If any of those locks have more than $v - m + 1$ keys, it will be more efficient to use a mistrust design, which has only $v - m + 1$ keys per lock.

□

A solution using a recurrence relation

It is possible to use a recurrence relation (and dynamic programming) to calculate the number of locks needed in a mistrust design. Since the number of locks in a mistrust design can be expressed with a binomial coefficient, it seems reasonable to try a recurrence relation patterned after Pascal's Theorem. The following definition and theorem show that this approach does work. As a bonus, the recurrence relation provides a constructive algorithm for actually printing the mistrust design as a table.

Definition 4 ($L(v, m)$)**.** *For $v, m \geq 1$, define $L(v, m)$ by*

$$\begin{aligned} L(v, m) &= 0 \text{ if } m > v \\ L(v, 1) &= 1 \\ L(v, v) &= v \\ L(v, m) &= L(v - 1, m) + L(v - 1, m - 1) \text{ for } v, m \geq 2 \end{aligned}$$

Theorem 5 ($L(v, m)$ is the number of locks in a mistrust design)**.** *The number of locks in the mistrust design $M_{v,m}$ can be calculated using the recurrence relation $L(v, m)$. That is, $L(v, m) = \binom{v}{m-1}$ for all $v, m \geq 1$.*

Proof. The three base conditions are easy to see. If more than v bandits are needed to open all the locks, then the task is not possible. If every bandit should be able to open all the locks, then only one lock and v keys are needed. If all v of the bandits are needed to open all the locks, then v locks, each having only 1 key will be needed. (See the first example after this proof.)

A two-way induction will be used to show that $L(v, m) = \binom{v}{m-1}$, the number of locks in $M_{v,m}$.
Notice that

$$L(v, 1) = 1 = \binom{v}{1 - 1} = \binom{v}{m - 1}$$

$$L(v, v) = v = \binom{v}{v - 1} = \binom{v}{m - 1}$$

Suppose for all w with $1 \leq w < v$ and any m with $1 \leq m \leq w$ that $L(w, m) = \binom{w}{m-1}$. Then

$$\begin{aligned} L(v, m) &= L(v - 1, m) + L(v - 1, m - 1) \\ &= \binom{v - 1}{m - 1} + \binom{v - 1}{m - 2} \\ &= \binom{v}{m - 1} \end{aligned}$$

by Pascal's Theorem. □

Example 2 ($M_{3,1}$ and $M_{4,4}$)**.** *The locks and the key distributions for $M_{3,1}$ and $M_{4,4}$ are shown here.*

$M_{3,1}$	L_1
1	X
2	X
3	X

$M_{4,4}$	L_1	L_2	L_3	L_4
1	X			
2		X		
3			X	
4				X

The recurrence relation provides a constructive algorithm for producing the table of key assignments for the locks. The table for $M_{v,1}$ is a column of v Xs. The table for $M_{v,v}$ consists of a v by v table with Xs on the main diagonal (and nowhere else). Otherwise, assume that the tables $M_{v-1,m}$ and $M_{v-1,m-1}$ can be constructed. Then the table for $M_{v,m}$ can be built in the following manner. Rename the locks for $M_{v-1,m-1}$ using the subscripts $\binom{v-1}{m-1}+1,\ldots,\binom{v}{m-1}$.

$M_{v,m}$	$L_1 \cdots L_{\binom{v-1}{m-1}}$	$L_{\binom{v-1}{m-1}+1} \cdots L_{\binom{v}{m-1}}$
1		
$\cdots$	$M_{v-1,m}$	$M_{v-1,m-1}$
$v-1$		
v	X X $\cdots$ X	

Notice that bandit v only holds keys to the first $\binom{v-1}{m-1}$ locks. If any $m-1$ of the first $v-1$ bandits try to unlock all the locks, by the design of $M_{v-1,m}$ and $M_{v-1,m-1}$, they will not have enough keys to unlock the lefthand subset of locks but they can open the righthand subset of locks. Adding bandit v to such a subset will allow all the locks to be opened.

Similarly, any subset of $m-1$ bandits that includes bandit v will be unable to open all the locks since bandit v does not have any keys to the second collection of locks and the other $m-2$ bandits do not have enough keys among them to unlock the collection. Adding one more bandit will enable all the locks to be opened.

Example 3 ($M_{6,3}$ from $M_{5,3}$ and $M_{5,2}$)**.** *The constructive algorithm can be illustrated by creating $M_{6,3}$ from $M_{5,3}$ and $M_{5,2}$. Each of those mistrust designs can be created from smaller mistrust designs. (It might be of interest to divide the chart even further to see this.)*

$M_{6,3}$	L_1	L_2	L_3	L_4	L_5	L_6	L_7	L_8	L_9	L_{10}	L_{11}	L_{12}	L_{13}	L_{14}	L_{15}
1	X			X		X	X		X	X	X		X	X	X
2		X			X	X		X	X	X		X	X	X	X
3			X	X	X		X	X		X	X	X		X	X
4	X	X	X				X	X	X		X	X	X		X
5	X	X	X	X	X	X					X	X	X	X	
6	X	X	X	X	X	X	X	X	X	X					

From Mistrust Designs to Balanced Incomplete Block Designs

Example 4 ($M_{5,3}$ in a different format)**.** *The following chart lists the locks for $M_{5,3}$. Under each lock is the subset of bandits that hold keys for that lock. The bandits are denoted as thief 1 through thief v: $t_1, \ldots, t_v$.*

L_1	L_2	L_3	L_4	L_5	L_6	L_7	L_8	L_9	L_{10}
t_1	t_2	t_3	t_1	t_2	t_1	t_1	t_2	t_1	t_1
t_4	t_4	t_4	t_3	t_3	t_2	t_3	t_3	t_2	t_2
t_5	t_5	t_5	t_5	t_5	t_5	t_4	t_4	t_4	t_3

This arrangement is an example of a combinatorial design called a balanced incomplete block design. In the formal definition below, the blocks correspond to the locks in Example 4 and the varieties correspond to the bandits.

Definition 5 (Balanced Incomplete Block Design)**.** *A* balanced incomplete block design*, abbreviated* BIBD*, is a combinatorial design consisting of a finite collection of finite sets (called* blocks*), each consisting of a finite number of elements (called* varieties*). The boundary conditions a BIBD must satisfy are expressed in terms of five parameters, commonly expressed as the 5-tuple of positive integers, (v, b, r, k, λ).*

The parameter v represents the number of distinct varieties; the parameter b represents the number of blocks. Every variety is required to be in exactly r blocks, and every block must contain exactly k varieties. Finally, every pair of distinct varieties must appear in exactly λ common blocks.

A combinatorial design which meets these conditions is often referred to as a (v, b, r, k, λ)-design.

A (v, b, r, k, λ)-design with $k = v$ and $r = b$ is called trivial*.*

Notes:

1. The first four parameters are positive: $v \geq 1, b \geq 1, r \geq 1, k \geq 1$.
   The parameter λ is nonnegative: $\lambda \geq 0$.

2. A trivial BIBD consists of b identical blocks,each containing every variety.

3. The design in Example 4 is a $(5, 10, 6, 3, 3)$-design.

Example 5 (A $(7, 14, 6, 3, 2)$-design)**.** *Here is a $(7, 14, 6, 3, 2)$-design using the set of varieties $\{0, 1, 2, 3, 4, 5, 6\}$. There are $\binom{7}{3} = 35$ variety subsets of size 3, but only 14 of them are represented as blocks.*

B_1	B_2	B_3	B_4	B_5	B_6	B_7	B_8	B_9	B_{10}	B_{11}	B_{12}	B_{13}	B_{14}
0	0	0	0	0	0	1	1	1	1	2	2	2	2
1	1	3	3	5	5	3	3	4	4	3	3	4	4
2	2	4	4	6	6	5	6	5	6	5	6	5	6

An alternative method for displaying a BIBD is to use an incidence matrix.

Definition 6 (The Incidence Matrix of a BIBD)**.** *Let D be a (v, b, r, k, λ)-design with varieties $\{t_1, t_2, \ldots, t_v\}$ and blocks $\{L_1, L_2, \ldots, L_b\}$.*

The incidence matrix *of D is the v by b matrix, M, where*

$$m_{ij} = \begin{cases} 1 & \text{if } t_i \in L_j \\ 0 & \text{otherwise} \end{cases}$$

Example 6 (The incidence matrix for $M_{5,3}$)**.** *If the varieties (bandits) in $M_{5,3}$ are denoted by $\{t_1, t_2, t_3, t_4, t_5\}$, then the incidence matrix for that mistrust design is:*

$$\begin{array}{c} \\ t_1 \\ t_2 \\ t_3 \\ t_4 \\ t_5 \end{array} \begin{array}{c} \begin{array}{cccccccccc} L_1 & L_2 & L_3 & L_4 & L_5 & L_6 & L_7 & L_8 & L_9 & L_{10} \end{array} \\ \left[\begin{array}{cccccccccc} 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \end{array}\right] \end{array}$$

Compare this to the previous representation as:

L_1	L_2	L_3	L_4	L_5	L_6	L_7	L_8	L_9	L_{10}
t_1	t_2	t_3	t_1	t_2	t_1	t_1	t_2	t_1	t_1
t_4	t_4	t_4	t_3	t_3	t_2	t_3	t_3	t_2	t_2
t_5	t_5	t_5	t_5	t_5	t_5	t_4	t_4	t_4	t_3

The incidence matrix makes it easy to see some very basic necessary conditions for the existence of a BIBD.

Theorem 6 (The Parameters of a BIBD)**.** *Let D be a balanced incomplete block design with parameters (v, b, r, k, λ). Then*

$$bk = vr \qquad \text{and} \qquad r(k-1) = \lambda(v-1)$$

Proof. Let M be the incidence matrix for the (v, b, r, k, λ)-design. Both equations will be proved using combinatorial proofs that count 1s in M.

The first equation is verified by counting all the 1s in M two different ways.

Since there are b blocks, each containing k varieties, each of the b columns of M will contain k 1s, for a total of bk 1s. On the other hand, each of the v varieties is in r blocks, so each of the v rows of M contains r 1s, for a total of vr 1s. Therefore, $bk = vr$.

The second equation is verified by counting the 1s in a submatrix of M. Start by choosing any variety, t. Delete the row of M that corresponds to t and delete every column that corresponds to a block that does not contain t. Now count the 1s in the matrix, M_t, that remains.

Since t is in r blocks, there will be r columns in M_t. Each of those columns will contain $k-1$ 1s (since the 1 in t's row has been removed). On the other hand, t is in λ common blocks with each of the $v-1$ other varieties. So each of those varieties contributes λ 1s to M_t. Consequently, $r(k-1) = \lambda(v-1)$.

□

Theorem 7 (Mistrust Designs as BIBDs)**.** *A mistrust design is a*

$$\left(v, \binom{v}{m-1}, \binom{v-1}{m-1}, v-m+1, \binom{v-2}{m-1}\right) \text{- design.}$$

Proof. v The number of varieties is v, the number of bandits.

b A mistrust design was defined by creating all $\binom{v}{m-1}$ subsets of $m-1$ bandits and for each subset associating a lock. Thus there are $b = \binom{v}{m-1}$ blocks.

k The subsets of $m-1$ bandits are the ones who do not hold a key to the associated lock. That means there must be $k = v-(m-1) = v-m+1$ who do have keys to a given lock. So there are $k = v-m+1$ varieties in every block. The design consists of *all* $\binom{v}{m-1} = \binom{v}{v-m+1} = \binom{v}{k}$ subsets of size k.

r How many blocks will contain each variety? Consider a variety t. Once t has been chosen, we still have $v-1$ varieties from which to chose the other $k-1 = (v-m+1)-1 = v-m$ varieties in the subset (block). So t will be in a subset of size $v-m+1$ exactly $\binom{v-1}{k-1} = \binom{v-1}{v-m} = \binom{v-1}{m-1}$ times (using Theorem 1). Therefore $r = \binom{v-1}{m-1}$.

λ If we choose a pair of varieties, there will be $k-2 = (v-m+1)-2$ other varieties with them in a block. There are $v-2$ other varieties from which to choose those $k-2$ varieties, so the original pair of varieties will be together in $\binom{v-2}{k-2} = \binom{v-2}{(v-m+1)-2} = \binom{v-2}{m-1}$ subsets of size $v-m+1$. Consequently, $\lambda = \binom{v-2}{m-1}$. □

Example 7 (Some BIBD parameters from Mistrust Designs)**.** *The following table lists the BIBD parameters for several mistrust designs. The parameters were calculated by using Theorem 7:* $(v, b, r, k, \lambda) = \left(v, \binom{v}{m-1}, \binom{v-1}{m-1}, v-m+1, \binom{v-2}{m-1}\right)$.

v	m	v	b	r	k	λ
6	*3*	*6*	*15*	*10*	*4*	*6*
6	*4*	*6*	*20*	*10*	*3*	*4*
8	*4*	*8*	*56*	*35*	*5*	*20*
10	*3*	*10*	*45*	*36*	*8*	*28*
10	*6*	*10*	*252*	*126*	*5*	*56*

References

[1] ACM International Collegiate Programming Competition North America North Central 2014 Regionals "Locked Treasure" problem (6873). `https://icpcarchive.ecs.baylor.edu/index.php?option=com\_onlinejudge\&Itemid=8\&category=663\&page=show\_problem\&problem=4885`. Accessed: 2015-07-22.

[2] E. Gossett. *Discrete Mathematics With Proof.* Wiley, 2 edition, 2009.