ISSN 2182-1976

Recreational Mathematics Magazine

A Very Mathematical Card Trick

Carlos Santos, Jorge Nuno Silva, Pedro Duarte
CELC, University of Lisbon, University of Lisbon
cmfsantos@fc.ul.pt, jnsilva@cal.berkeley.edu, pduarte@ptmat.fc.ul.pt

Number 2
September, 2014

Ludus

CIUHCT

MathMagic

A Very Mathematical Card Trick

Carlos Santos, Jorge Nuno Silva, Pedro Duarte
CELC, University of Lisbon, University of Lisbon
cmfsantos@fc.ul.pt, jnsilva@cal.berkeley.edu, pduarte@ptmat.fc.ul.pt

Abstract: *Andy Liu proposed a very interesting card trick whose explanation is based in the well-known Hamming codes. In this work, a performance for the same trick based in* NIM *mental calculation, is presented. A similar idea is also useful to analyze the game* TOE TACK TRICK *proposed by Colm Mulcahy.*

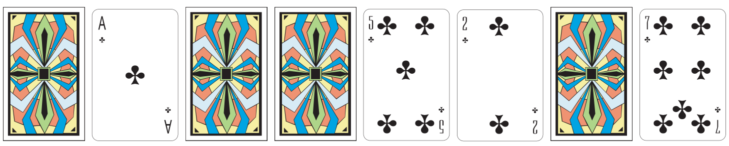
Key-words: mathematical card tricks, Hamming codes, NIM.

1 Introduction

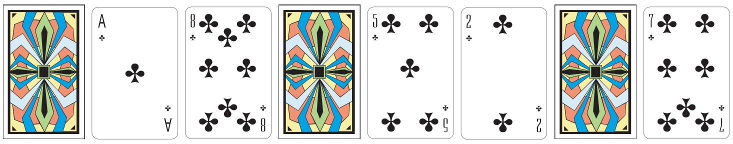
We start with the principal purpose of this paper: a magical card trick. It uses a small set of cards: ace through eight of clubs. The audience chooses one of them, giving the information about the choice to a magician's helper. Then, one volunteer of the audience shuffles the eight card deck, places the cards in a row, arbitrarily deciding which should be turned up. The helper does not do anything and the magician is not in the room.

Following, the helper turns **exactly** one card. After all this, the magician, who does not know what happened, enters the room and, looking at the cards, determines the card chosen by the audience.

Consider the following example. The audience chooses the deuce and leaves the following setup:



Following, the helper turns the third position leaving the following row:



To finish the trick, the magician enters in the room and shouts "deuce of clubs!".

2 Hamming Codes and the Liu's Card Trick

In [3], it is shown how the trick is conceived using Hamming codes. A Hamming code is a linear error-correcting code to detect single-bit errors. Exemplifying, consider a 8-bit word $a_0a_1a_2a_3a_4a_5a_6a_7$. The codification of the message includes 4 more digits (test-bits t_i). The t_i occupy the positions 1, 2, 4 and 8 (powers of 2). The values t_i are test-bits chosen by solving the following 4 equations:

$$t_0 + a_0 + a_1 + a_3 + a_4 + a_6 \equiv 0 \pmod 2 \quad (1)$$
$$t_1 + a_0 + a_2 + a_3 + a_5 + a_6 \equiv 0 \pmod 2 \quad (2)$$
$$t_2 + a_1 + a_2 + a_3 + a_7 \equiv 0 \pmod 2 \quad (3)$$
$$t_3 + a_4 + a_5 + a_6 + a_7 \equiv 0 \pmod 2 \quad (4)$$

It is usual to organize the following chart related to the encoding process:

Bit Position	1	2	3	4	5	6	7	8	9	10	11	12
Encoded Bits	t_0	t_1	a_0	t_2	a_1	a_2	a_3	t_3	a_4	a_5	a_6	a_7
Equation 1	×		×		×		×		×		×	
Equation 2		×	×			×	×			×	×	
Equation 3				×	×	×	×					×
Equation 4								×	×	×	×	×

To understand the concept behind the codification, we note that the 4 equations can fail by 15 distinct ways (15 of the 16 subsets of a set with 4 elements). For instance, the simultaneous failure of the 1st and the 4th is one of these possible failures. The codification is done in such way that every single-bit error is related to exactly one of the possible failures. From code theory, it is known that the rules for the equations can be listed like this:

Eq 1: *skip* 0, *check* 1, *skip* 1, *check* 1, *skip* 1, … → *positions* $1, 3, 5, 7, 9, 11, 13, 15, \ldots$

Eq 2: *skip* 1, *check* 2, *skip* 2, *check* 2, *skip* 2, … → *positions* $2, 3, 6, 7, 10, 11, 14, 15, \ldots$

Eq 3: *skip* 3, *check* 4, *skip* 4, *check* 4, *skip* 4, … → *positions* $4, 5, 6, 7, 12, 13, 14, 15, \ldots$

Eq 4: *skip* 7, *check* 8, *skip* 8, *check* 8, *skip* 8, … → *positions* 8–$15, 24$–$31, 40$–$47, \ldots$

(…)

Eq k: *skip* $2^k - 1$, *check* 2^k, *skip* 2^k, *check* 2^k, *skip* $2^k, \ldots$

There is a unique bit coverage. For example, the bit responsible for the failure of the equations 1 and 4 is the 9th bit (a_4). The receptor of the message just has to check the congruences (1), (2), (3), and (4) to determine the bits with error.

Consider the message 10010111. The equations (1), (2), (3), and (4) produce the Hamming code 101000110111. Imagine a single-bit error and the sent message 101001110111 (with an error in the 6th position). The receptor calculates the congruences (1), (2), (3), and (4) and sees that (1) and (4) hold while (2) and (3) fail. The error occurs in the 2nd and 3rd equations so, by table inspection, the error-bit is in 6th bit. The detection of the error position can be made by visual inspection of a table. There are 2^k subsets of a finite set with cardinality k, so, if the message length (k) is such that $2^j \leqslant k < 2^{j+1}$ then the encoded message needs $j+1$ test-bits.

There are several card tricks based in the Hamming codes (see the chapter "Hamming It Down" of [4]). Andy Liu's idea is *to prepare the magician's reception.* Card's backs act like 1s and card's faces act like 0s. In our example, the trick's victim leaves a configuration encoded by 10110010 and the helper wants to "construct an error" in the second position (to inform the magician about the chosen card, the deuce). First he should organize the following table:

Bit Position	1	2	3	4	5	6	7	8
Encoded Bits	1	0	1	1	0	0	1	0
Eq 1	×		×		×		×	
Eq 2		×	×			×	×	
Eq 3				×	×	×	×	
Eq 4								×

Then, checking the four congruences, the helper finds that only (1) fails. As the helper wants just (2) to fail, he has to adjust (1) and (2). This can be done flipping the third bit. With cards, the helper has to turn the third card. The magician arrives and makes the same congruence calculations and table inspection. In this communication scheme, t_3 acts like neutral element. If the magician, after inspection, sees that (2) and (4) fail, it is the same as if only (2) fails. If the magician, after inspection, sees that nothing fails, it is the same as if only (4) fails. This is not an easy process. We repeat,

THIS IS NOT NICE FOR THE MAGICIAN!

3 NIM Approach

The main goal of this paper is to show how NIM sum can help the assistant and the magician to execute the trick.

The classic game of NIM, first studied by C. Bouton [1], is played with piles of stones. On his turn, each player can remove any number of stones from any pile. Under Normal Play convention, the winner is the player who takes the last stone.

The *nim-sum* of two nonnegative integers is the exclusive or (XOR), written $\oplus$, of their binary representations. It can also be described as adding the numbers in binary without carrying.

If a position is a previous player win, we say it is a P-position. If a game is a next player win, we say it is a N-position. The set of P-positions is noted $\mathcal{P}$ and the set of N-positions is noted $\mathcal{N}$. Bouton proved that $\mathcal{P}$ is exactly the set of positions such that the nim-sum of the sizes of the piles is zero.

The structure $(\mathbb{N}_0, \oplus)$ is an infinite group. It is very easy to execute mental calculations with nim-sum. It is just needed to write the summands in binary notation, canceling repetitions in pairs and using standard addition for the remaining powers. Some examples:

$$5 \oplus 3 = (4+1) \oplus (2+1) = (4+\cancel{1}) + (2+\cancel{1}) = 6$$

$$\begin{aligned} 11 \oplus 22 \oplus 35 &= (8+2+1) \oplus (16+4+2) \oplus (32+2+1) = \\ &= (8+\cancel{2}+\cancel{1}) + (16+4+\cancel{2}) + (32+2+\cancel{1}) = 62 \end{aligned}$$

The structures $(\{0, \ldots, 2^k-1\}, \oplus)$ are finite groups with the property $x \oplus x = 0$. Following, the table for the case $(\{0, \ldots, 15\}, \oplus)$.

$\oplus$	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
0	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	1	0	3	2	5	4	7	6	9	8	11	10	13	12	15	14
2	2	3	0	1	6	7	4	5	10	11	8	9	14	15	12	13
3	3	2	1	0	7	6	5	4	11	10	9	8	15	14	13	12
4	4	5	6	7	0	1	2	3	12	13	14	15	8	9	10	11
5	5	4	7	6	1	0	3	2	13	12	15	14	9	8	11	10
6	6	7	4	5	2	3	0	1	14	15	12	13	10	11	8	9
7	7	6	5	4	3	2	1	0	15	14	13	12	11	10	9	8
8	8	9	10	11	12	13	14	15	0	1	2	3	4	5	6	7
9	9	8	11	10	13	12	15	14	1	0	3	2	5	4	7	6
10	10	11	8	9	14	15	12	13	2	3	0	1	6	7	4	5
11	11	10	9	8	15	14	13	12	3	2	1	0	7	6	5	4
12	12	13	14	15	8	9	10	11	4	5	6	7	0	1	2	3
13	13	12	15	14	9	8	11	10	5	4	7	6	1	0	3	2
14	14	15	12	13	10	11	8	9	6	7	4	5	2	3	0	1
15	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

For the “easy” implementation of the card trick it is important to prove the following theorem:

Theorem 1. *Let* $\{a_0, a_1 \ldots, a_j\} \subseteq \{0, \ldots, 2^k - 1\}$*. For all* $N \in \{0, \ldots, 2^k - 1\}$*, one of the following holds:*

1. $\exists\ i \in \{0, \ldots, j\} : a_0 \oplus a_1 \oplus \cdots \oplus a_{i-1} \oplus a_{i+1} \oplus \cdots \oplus a_j = N$*;*
2. $\exists\ b \in \{0, \ldots, 2^k - 1\} \setminus \{a_0, a_1, \ldots, a_j\} : a_0 \oplus a_1 \oplus \cdots \oplus a_j \oplus b = N$*.*

Proof. The proof is a direct consequence of the property $x \oplus x = 0$. We begin to consider the equation

$$a_0 \oplus a_1 \oplus \cdots \oplus a_j \oplus x = N.$$

As the inverse of a number is itself, the solution of the equation is

$$x = N \oplus a_0 \oplus a_1 \oplus \cdots \oplus a_j.$$

If $N \oplus a_0 \oplus a_1 \oplus \cdots \oplus a_j \notin \{a_0, a_1, \ldots, a_j\}$ then *2* holds and

$$b = N \oplus a_0 \oplus a_1 \oplus \cdots \oplus a_j.$$

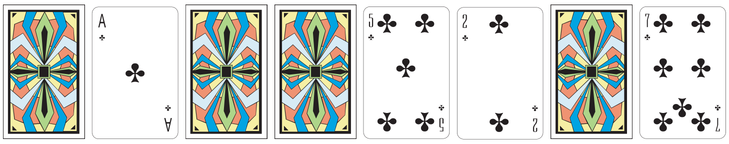
If $N \oplus a_0 \oplus a_1 \oplus \cdots \oplus a_j \in \{a_0, a_1, \ldots, a_j\}$ then *1* holds and

$$a_i = N \oplus a_0 \oplus a_1 \oplus \cdots \oplus a_j.$$

□

This theorem provides a very elegant communication between the helper and the magician. Consider again our first example.

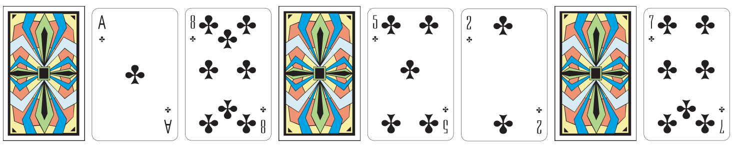
The audience chooses the deuce and leaves the following setup:



With the order $1, 2, \ldots, 6, 7, 0$ (0 corresponds to 8) and with the convention *On→Back* and *Off→Face*, the helper calculates

$$x = \underbrace{1 \oplus 3 \oplus 4 \oplus 7}_{a_i(Back)} \oplus \underbrace{2}_{N} = 3.$$

In this case, $x = 3$. Because the third card is backward, the situation corresponds to the first item of the Theorem 1. So, the helper turns the third card giving the following setup to the magician:



Now, the magician just calculates $N = 1 \oplus 4 \oplus 7 = 2$ and shouts "deuce of clubs!"

This is a MUCH EASIER execution of the trick.

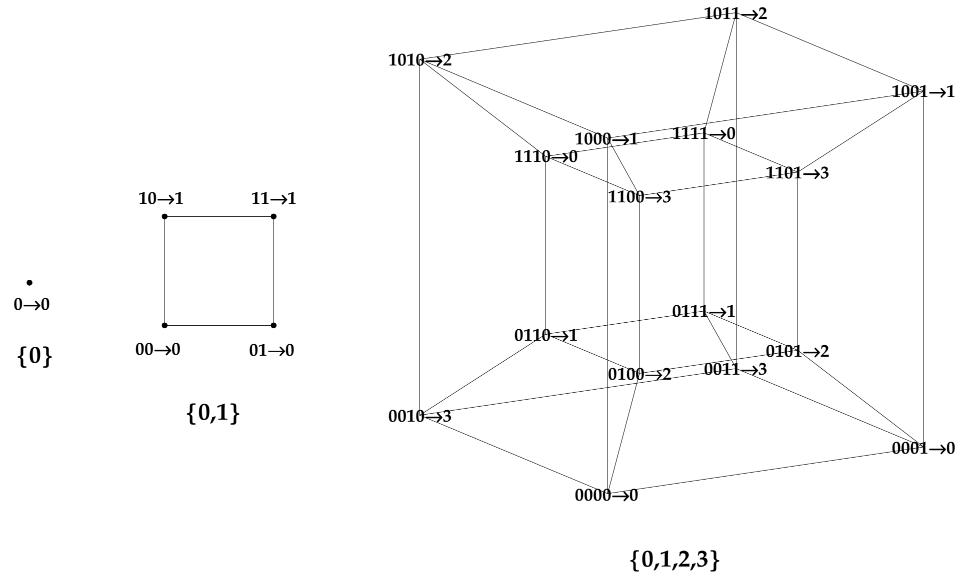
The trick has a nice geometric interpretation. In the first part of the card trick, the victim gives a configuration to the helper and the information about a chosen card $N \in \{0, \ldots, 2^k - 1\}$. We can associate each configuration to a graph's vertex. The helper's move is to choose an adjacent vertex of the given configuration. If we can define a function $f : V(G) \to \{0, \ldots, 2^k - 1\}$ over the set of vertices such that the helper can always choose a move giving $f(v) = N$, the trick is explained.

"Good" graphs are the hypercubes $I^k = \{0,1\}^{2^k}$ with 2^{2^k} vertices (the vertices are all the arrangements $\alpha_1\alpha_2 \ldots \alpha_{2^{k-1}}\alpha_{2^k}$ $(\alpha_i \in \{0,1\})$. In those hypercubes, each vertex has degree 2^k. A function satisfying our goal is

$$f : I^k \to \{0, 1, \ldots, 2^k - 1\} \text{ given by}$$

$$f(\alpha_1\alpha_2 \ldots \alpha_{2^{k-1}}\alpha_{2^k}) = \alpha_1 \oplus 2\alpha_2 \oplus 3\alpha_3 \oplus \cdots \oplus (k-1)\alpha_{2^{k-1}}$$

We can visualize the geometric idea:



If we perform the trick with just 4 cards, the action on the hypercube is easy to visualize. For instance, if the helper gets the configuration 1010 and wants to inform the chosen card 3, he must chose the vertex 0010 (he turns the first card).

4 Colm Mulcahy's TOE TACK TRICK

The knowledge about the structure $(\mathbb{N}_0, \oplus)$ gives a very practical way to deal with Hamming codes. Suppose the message 10010111 and the related scheme:

Bit Position	1	2	3	4	5	6	7	8	9	10	11	12
	t_0	t_1	a_0	t_2	a_1	a_2	a_3	t_3	a_4	a_5	a_6	a_7
Encoded Bits	?	?	1	?	0	0	1	?	0	1	1	1

The procedure starts NIM adding the *positions* of the $a_i = 1$ (N). In this example, $N = 3 \oplus 7 \oplus 10 \oplus 11 \oplus 12 = 9$. Because $(\{0, 1, \dots, 15\}, \oplus)$ is a finite group, it is mandatory that, for all possible messages with the considered length, N is an element of $\{0, 1, \dots, 15\}$. After, the t_i are chosen in such a way that the nim-sum of the positions of 1's is N. This is always possible because the t_i's positions are the powers of 2 (in this case, 1, 2, 4 and 8). In fact, the t_i are chosen in such way that $t_3t_2t_1t_0$ is the binary expansion of N. In the example, $t_3 = 1$, $t_2 = 0$, $t_1 = 0$, and $t_0 = 1$. The encoded message is 101000110111.

When the receptor receives the message, he calculates the nim-sum $\bigoplus p_i$ (nim-sum of the positions with digit 1). If the result is different than zero, an error occurred. Say that $\bigoplus p_i = x$. The question is: what is the position where the error occurred? That is, what is the value y such that $\bigoplus p_i \oplus y = 0$? It is possible to understand that x is the answer, revealing the position where the error occurred. If $x \notin \{p_i\}$ then the element in position x is 0 and must be replaced by 1. If $x \in \{p_i\}$ then the element in position x is 1 and must be replaced by 0. The receptor just has to compute $\bigoplus p_i$ to discover the position x. Suppose that the error transforms the encoded message 101000110111 into 101000110101. The receptor calculates $1 \oplus 3 \oplus 7 \oplus 8 \oplus 10 \oplus 12 = 11$ and immediately understands that the error was in the 11th position. The finite NIM groups provide an elegant explanation for the Hamming's idea.

At *Gathering for Gardner 9*, Colm Mulcahy showed his TOE TACK TRICK. The game starts with a an empty 3×3 grid as in the ordinary TIC-TAC-TOE, but in TOE TACK TRICK the grid is totally filled and the winner is the player who finishes with the smallest number of three-in-row. In this version, both players can use both symbols ("X" and "O"). However there is another important difference, in TIC-TAC-TOE players can place the symbols where they want, but, in TOE TACK TRICK, First can only play in the middle of each side and Second can only play in the corners.

2nd	1st	2nd
1st		1st
2nd	1st	2nd

During the first stage, First fills in **all** his four cells. **After**, during the second stage, Second fills in all his corners. During a third and last stage, the center is filled with "X" or "O", depending of a coin toss.

A three-in-row counts for Second if the configuration ends in the corners (a side or a diagonal). A three-in-row counts for First if the configuration ends in the middle of the sides.

Suppose the final configuration:

X	O	O
X	O	X
O	O	X

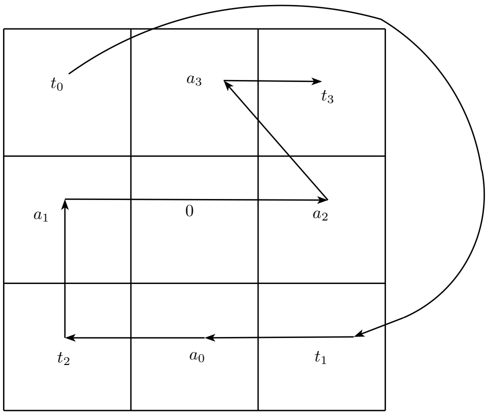
Each player has one three-in-row. The game results in a draw.

Mulcahy's proposal is related, not to the game itself, but to a very interesting communication situation. Suppose that the second player, fearing that his opponent could be a cheater, prepares a communication scheme with a good friend (Sherlock). Second knows that, after playing a game, First always cheats, *switching exactly one symbol.* In the previous example, suppose that First changes the final grid to the following one:

O	O	O
X	O	X
O	O	X

Sherlock, who has witnessed nothing, enters in the scene and comments "You, First, are a cheater. You changed that mark in the top-left corner, winning the game $(1-2)$. Before the switch, you each had one three-in-row and the game was a draw.”.

How is this possible? Sherlock saw and heard nothing! Again, instead of Hamming codes, we can explain everything with recourse to nim-sum. Consider the following communication scheme:



During the first stage of the game, Second saw the First’s choices:

	0 (7th)	
1 (5th)		1 (6th)
	0 (3rd)	

He calculated $5 \oplus 6 = 3$ and, because, in binary, $3 = 11$, he made the choice $t_3 t_2 t_1 t_0 = 0011$:

1	0	0
1		1
0	0	1

After the randomized central move and the "cheating switch", Sherlock entered in the room and observed the final configuration:

0	0	0
1	0	1
0	0	1

He calculated $2 \oplus 5 \oplus 6 = 1$ and discovered that the first position was changed. If the nim-sum was equal to 0, then the cheating switch should have been in the center.

For more relations between Error-Correcting Codes and NIM algebraic operations see [2]. This reference is concerned with various classes of *lexicographic* codes, that is, codes that are defined by a greedy algorithm: each successive codeword is selected as the first word not prohibitively near to earlier codewords (in the sense of Hamming distance, the number of positions at which the corresponding symbols are different). Among others, the authors proved a very interesting theorem: for a base $B = 2^{2^a}$, unrestricted lexicodes are closed under NIM addition and NIM multiplication.

References

[1] C. L. Bouton. "Nim, a game with a complete mathematical theory", *Ann. Math.* 3 (2), 35–39, 1902.

[2] J. Conway, N. Sloane. "Lexicographic codes: Error-correcting codes from game theory", *IEEE Transactions on Information Theory*, 32 (3), 1986.

[3] A. Liu. 'Two applications of a Hamming code", *The College Mathematics Journal*, 40 (1), 2009.

[4] C. Mulcahy. *Mathematical Card Magic: Fifty-Two New Effects*, A. K. Peters/CRC Press, 2013.