The Solutions and Generalization of the Lights Out Game

Team Member

Tang Haoran, Lu Jiejun, Huang Yifeng

School

Guangdong Experimental High School

Province

Guangdong Province

Teacher

Guo Weidong, Cheng Jianhua, Jiang Qiuming

Abstract：

Lights Out is a video game, which was released by Tiger Toys in 1995. Generally, there are 3×3, 4×4, 5×5, 10×10 grids of games. Under the normal circumstances, it's easy to deal with some special configurations manually. However, the possibility of solving the randomly-generated configuration by hand is small, for it's a bit difficult to find the law. Out of interest and enthusiasm about the game, our team reads some reference and gathered four mathematic methods to solve the game. What's more, we extend the methods to m×n games, the cycling game, the lit-only game and N-color game. We independently propose a method to classify all configurations in the game for 5×5, which can also be extended to general cases. We also put forward the issues about N-order hexagon games, and provide some feasible solutions.

Our independent research productions include:

1. Researching using the method of *row by row* to solve an m×n game when not always winnable.

2. Trying to put forward two possible methods of classifying all configurations. Based on THEOREM 1., we put forward THEOREM 4，and classify all configurations in the 5×5 game. This THEOREM can be applied to any m×n Lights Out game.

3. Using the method of *row by row* to solve the cycling game, lit-only game and 3-color game in another way.

4. Putting forward the issues about N-order Hexagon Games, and provide some feasible solutions.

5. Summarize advantages and disadvantages and correlation of four methods, and explore the best solution to a specific m×n game.

Key words:

Graph Theory, Linear Algebra, Lights Out

Catalogue

(Here is the note: for the demand of distinguishing, we type * before our independent research production.)

1. Lights Out Games Introduction

Lights Out is a video game, which was released by Tiger Toys in 1995. The game consists of 5 by 5 grids of lights. When the game starts, a set of these lights (random, or one of a set of stored puzzle patterns) are switched on. Pressing one of the lights will toggle it and the four lights adjacent to it on and off. Doing so changes the on/off state of the light on the button pushed, and of all its vertical and horizontal neighbors. Given an initial configuration of lights which are turned on, the object is to turn out all the lights.

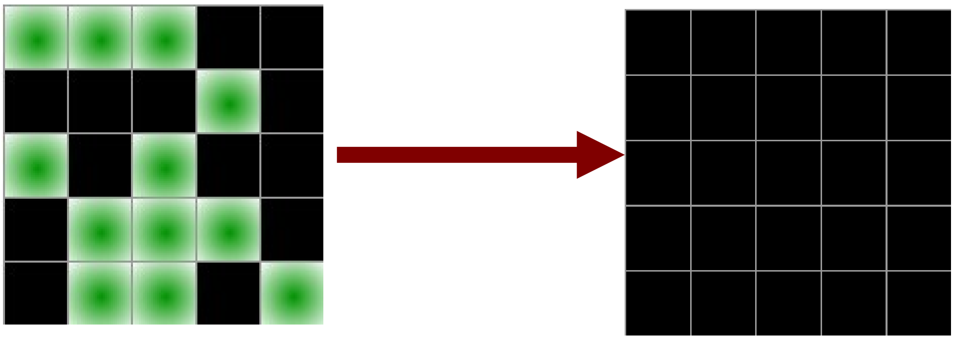


Figure 1.1 the object of lights out game: to turn out all the lights

First working on the simple cases, we choose the 3 by 3 grids of lights, which are easily understood.

Figure 1.2 denote these 9 lights from left to right and up to down respectively by $a_1, a_{2,}......a_9$

All the m×n grids games have these properties:

1. Pushing a button for even times is equivalent to not pushing it at all. Meanwhile, pushing it for odd times is equivalent to pushing it once.

2. The order in which the buttons are pushed is immaterial if they are fixed.

In order to note conveniently, we define as followed:

1. Denote state matrices by C_x (configuration), which is to represent a certain state of the game. Meanwhile, denote the configuration of all lights out by C_0 and all lights on by C_1

$$C_x = \begin{pmatrix} a_1 & a_2 & a_3 \\ a_4 & a_5 & a_6 \\ a_7 & a_8 & a_9 \end{pmatrix}.$$

2. Denote the entire matrix C_x by a column vector $\mathbf{b} = (a_1, a_2, a_3 \cdots\cdots, a_9)^T$

3. Denote these 9 lights by $a_1, a_2, a_3 \cdots\cdots a_9$ (figure 1.2). Sometimes we also represent them by $a_{11}, a_{12}, a_{13} \cdots\cdots a_{33}$.

4. Represent the strategy which solves a configuration by a column vector $\mathbf{x} = (x_1, x_2, x_3 \cdots\cdots x_9)^T$. $x_1, x_2, x_3 \cdots\cdots x_9$ for the location of the buttons mentioned respectively.

5. If C_x can be transformed into C_0 by pushing a set of buttons, we call C_x winnable, otherwise we call it unwinnable. If all C_x are winnable in the m×n game, we call this m×n game always winnable.

2. Lights Out Game Solving Methods

Up to now, we have sum up four possible solving methods to Lights Out game:

2.1 ±1 Exhaustion

Definition:

Define a_i =1, if light (i) is on; a_i =-1 if off. Define x_i =-1 as pushing the button (i); a_i =1 not pushing it. And a_i' is the final state of light(i).

Therefore we get:

$$a_1' = a_1 x_1 x_2 x_4$$
$$a_2' = a_2 x_1 x_2 x_3 x_5$$
$$a_3' = a_3 x_2 x_3 x_6$$
$$a_4' = a_4 x_1 x_4 x_5 x_7$$
$$a_5' = a_5 x_2 x_4 x_5 x_6 x_8$$
$$a_6' = a_6 x_3 x_5 x_6 x_9$$
$$a_7' = a_7 x_4 x_7 x_8$$
$$a_8' = a_8 x_5 x_7 x_8 x_9$$
$$a_9' = a_9 x_6 x_8 x_9$$

Turning all lights out, means $a_1' = a_2' = \cdots\cdots = a_9' = -1$, thus exhaustion will do the job. More generally, an m×n game has 2^{mn} cases.

What's more, given a final configuration C_y, exhaustion still works.

2.2 Exhaustion with Addition of Matrix [1]

[Please open "3×3 Exhaustion with addition of matrices.exe"]

After learning some basic knowledge about matrices, we found it easier using matrices to solve the problem.

Definition:

Use $a_{11}, a_{12}, a_{13} \cdots\cdots a_{33}$ in place of $a_1, a_2, a_3 \cdots\cdots a_9$. Define a_{ij} =0 when (i,j) is off; a_{ij} =1 when (i,j) is on. Use $\mathbf{x} = (x_{11}, x_{12}, x_{13} \cdots\cdots x_{33})$ instead of $\mathbf{x} = (x_1, x_2, x_3 \cdots\cdots x_9)$, 1 for pushing the button and 0 for not pushing it.

$$C_x = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix}$$

Note that pushing the button (1,1) in C_0 leads to A_{11}.Thus write down all the matrices.

$$A_{11}=\begin{pmatrix}1&1&0\\1&0&0\\0&0&0\end{pmatrix} A_{12}=\begin{pmatrix}1&1&1\\0&1&0\\0&0&0\end{pmatrix} A_{13}=\begin{pmatrix}0&1&1\\0&0&1\\0&0&0\end{pmatrix}$$

$$A_{21}=\begin{pmatrix}1&0&0\\1&1&0\\1&0&0\end{pmatrix} A_{22}=\begin{pmatrix}0&1&0\\1&1&1\\0&1&0\end{pmatrix} A_{23}=\begin{pmatrix}0&0&1\\0&1&1\\0&0&1\end{pmatrix}$$

$$A_{31}=\begin{pmatrix}0&0&0\\1&0&0\\1&1&0\end{pmatrix} A_{32}=\begin{pmatrix}0&0&0\\0&1&0\\1&1&1\end{pmatrix} A_{33}=\begin{pmatrix}0&0&0\\0&0&1\\0&1&1\end{pmatrix}$$

Turning all lights out means to find $\mathbf{x}=(x_{11},x_{12},x_{13}\cdots\cdots x_{33})$ as:

$$C_x+\sum_{1\le i,j\le 3} x_{ij}A_{ij}=C_0 \pmod 2$$

More generally, given a final configuration C_y, we just need to find $\mathbf{x}=(x_{11},x_{12},x_{13}\cdots\cdots x_{33})$ as:

$$C_x+\sum_{1\le i,j\le 3} x_{ij}A_{ij}=C_y \pmod 2$$

Also, exhaustion will do the job.

Example: Given $C_x=\begin{pmatrix}1&1&1\\1&1&1\\1&1&1\end{pmatrix}$, a computer works out $\mathbf{x}=(1,0,1,0,1,0,1,0,1)^T$, and just pushing the corresponding buttons will transform C_x into C_0.

2.3 Linear Algebra

2.3.1 when always winnable：[2]

Here we will give a brief introduction to Anderson and Feil's achievements.

Definition:

Denote matrix C_x by a 9×1 column vector $\mathbf{b}=(a_{11},a_{12},a_{13}\cdots\cdots,a_{33})^T$ .Also represent the strategy by a 9×1 column vector $\mathbf{x}=(x_{11},x_{12},x_{13}\cdots\cdots,x_{33})^T$

Note that buttons which affect a_{11} are x_{11},x_{12},x_{21} , represented by a row vector $\alpha_1=(1,1,0,1,0,0,0,0,0)$

Also write down $\alpha_2,\alpha_3\cdots\cdots\alpha_9$

Define a 9×9 matrix A where $\alpha_1,\alpha_2,\alpha_3\cdots\cdots\alpha_9$ are rows 1 to 9.

That means if a strategy $\mathbf{x}$ is performed on a configuration $\mathbf{b}$, $\mathbf{b}+A\mathbf{x}$ will be the final configuration.

To turn all lights out means finding out the solution $\mathbf{x}$ of equation $\mathbf{b}+A\mathbf{x}=\mathbf{0}$. Thus $\mathbf{x}$ can be found because: $\mathbf{x}=A^{-1}\mathbf{b} \quad (\mathrm{mod}\,2)$.

2.3.2 when not always winnable [2]:

Problems may emerge when A^{-1} does not exist. (Illustrated by the following A in the 5×5 game)

$$A=\begin{pmatrix} B & I & O & O & O \\ I & B & I & O & O \\ O & I & B & I & O \\ O & O & I & B & I \\ O & O & O & I & B \end{pmatrix} \qquad B=\begin{pmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 \end{pmatrix}$$

Thus perform a Gauss-Jordan elimination on A. That will yield $RA=E$, where E is the Gauss-Jordan echelon form, and R is the product of the elementary matrices which perform the reducing row operations.

Then find out the orthogonal basis of $A\mathbf{x}=\mathbf{0}$:

$$\mathbf{n_1}=(0,1,1,1,0,1,0,1,0,1,1,1,0,1,1,1,0,1,0,1,0,1,1,1,0)^T$$

$$\mathbf{n}_2=(1,0,1,0,1,1,0,1,0,1,0,0,0,0,0,1,0,1,0,1,1,0,1,0,1)^T$$

Then, because $A^T=A$,they got two theorems:

THEOREM1: A configuration is winnable if and only if $\mathbf{b}$ is perpendicular to the two vectors $\mathbf{n}_1$ and $\mathbf{n}_2$ (mod 2).

THEOREM2: Suppose $\mathbf{b}$ is a winnable configuration. Then the four winning strategies for $\mathbf{b}$ are $R\mathbf{b}$, $R\mathbf{b}+\mathbf{n}_1$, $R\mathbf{b}+\mathbf{n}_2$, $R\mathbf{b}+\mathbf{n}_1+\mathbf{n}_2$.

With the above theorems, it is possible to judge whether C_x is winnable, and to find out all its strategies. It can be proved that this method makes sense in any m×n game, and what's more, $A^T = A$ is true in any m×n game.

Below are results of n×n game given by Anderson and Feil.

n	Dimension of Null(An)	n	Dimension of Null(An)
2	0	12	0
3	0	13	0
4	4	14	4
5	2	15	0
6	0	16	8
7	0	17	2
8	0	18	0
9	8	19	16
10	0	20	0
11	6	21	0

2.4. Method of *Row by Row* [3]

2.4.1 When always winnable:

It is more interesting using this kind of method, comparatively easier to play a game by hand, which improves the charm and appeal of Lights out. (In order to show its advantages, we use 10×10 as an example. The game is named "10×10 Lights Out.swf" in the attachment.)

Definition:

Operation *P*: For any C_x in a game, do not push any button in the last row, and push the button immediately above each light which is on at the last row. That will turn the last row into all-off state. Repeat this process. Then all lights, except those in the first row, will be off (C'_x). Use row vector $\mathbf{s}=(a_1,a_2,\cdots\cdots,a_{10})$ to represent the state of first row.

For a given C_x, its operation *P* is unique, so does vector $\mathbf{s}$ (ignoring the order of buttons pushed). Take 10×10 as an example. Because if not push any buttons in last row, in order to turn all lights out in the last row, the operation in row 9 is unique, which means the operation in row 8 is unique too……thus, *P* is unique.

Therefore, performing *P* can transform all 2^{100} configurations into 2^{10} configurations represented by vector $\mathbf{s}$, thus the game is simplified.

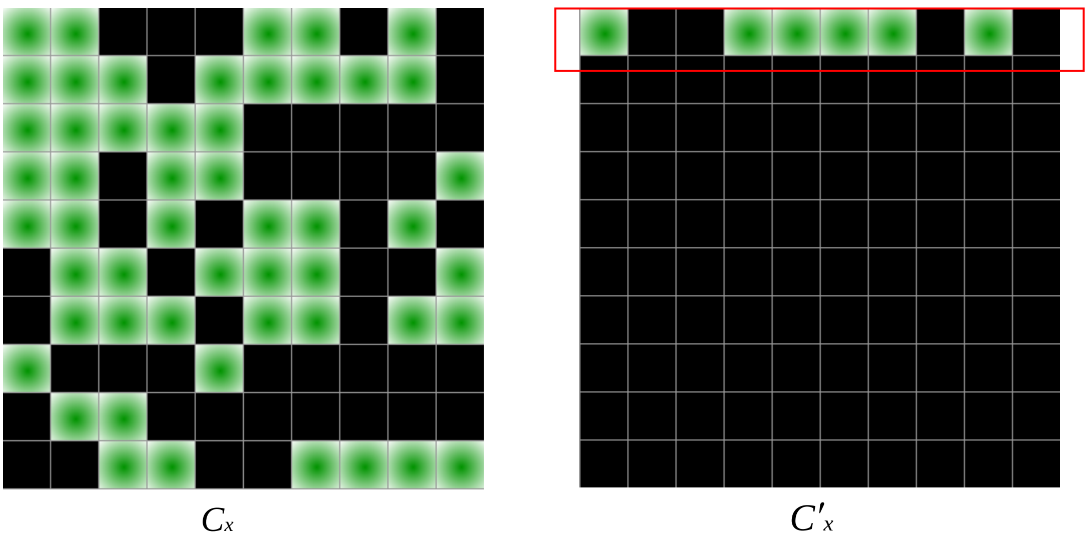
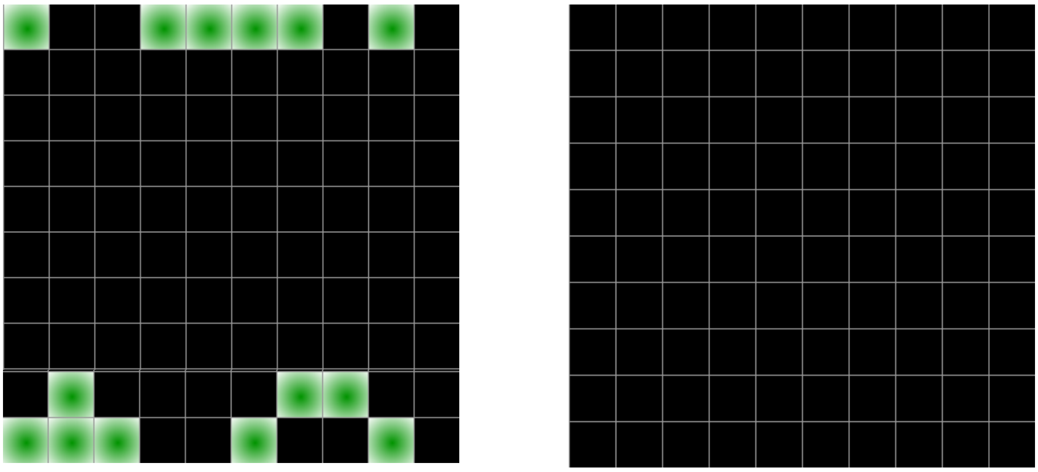


Figure 4.1 Perform *P* on C_x, and get C'_x

Note that not pushing any buttons in the last row, is the reason why we cannot get C_0 straightly. So we consider using $\mathbf{x}=(x_1,x_2,\cdots\cdots,x_{10})$ to operate the last row, and then get C''_x. Perform *P* again, and we get C_0.(here $\mathbf{x}=(0,1,0,0,0,0,1,1,0,0)$)



C''_x C_0

Figure 4.2 C''_x after operation *P* can be transformed into C_0.

Then we should find out the relations between $\mathbf{s}$ and $\mathbf{x}$.

Try pushing button x_1 in C_0, and then perform operation *P*, we find lights 1,3,7,9 in the first row is on, represented by $\mathbf{s}=(1,0,1,0,0,0,1,0,1,0)$. Similarly, we get:

$\mathbf{x}$=(1,0,0,0,0,0,0,0,0,0)， $\mathbf{s}$=(1,0,1,0,0,0,1,0,1,0)

$\mathbf{x}$=(0,1,0,0,0,0,0,0,0,0), $\mathbf{s}$=(0,0,0,1,0,1,0,0,0,1)

$\mathbf{x}$=(0,0,1,0,0,0,0,0,0,0), $\mathbf{s}$=(1,0,0,0,0,0,0,0,0,0)

$\mathbf{x}$=(0,0,0,1,0,0,0,0,0,0), $\mathbf{s}$=(0,1,0,1,0,1,0,0,0,1)

$\mathbf{x}$=(0,0,0,0,1,0,0,0,0,0), $\mathbf{s}$=(0,0,0,0,0,0,1,0,1,0)

Because the graph is symmetric, we have:

$\mathbf{x}$=(0,0,0,0,0,1,0,0,0,0), $\mathbf{s}$=(0,1,0,1,0,0,0,0,0,0)

$\mathbf{x}$=(0,0,0,0,0,0,1,0,0,0), $\mathbf{s}$=(1,0,0,0,1,0,1,0,1,0)

$\mathbf{x}$=(0,0,0,0,0,0,0,1,0,0), $\mathbf{s}$=(0,0,0,0,0,0,0,0,0,1)

$\mathbf{x}$=(0,0,0,0,0,0,0,0,1,0), $\mathbf{s}$=(1,0,0,0,1,0,1,0,0,0)

$\mathbf{x}$=(0,0,0,0,0,0,0,0,0,1), $\mathbf{s}$=(0,1,0,1,0,0,0,1,0,1)

That indicates if $\mathbf{s}=(a_1,a_2,\cdots\cdots,a_{10})$ represents the first row of a C'_x, then turning all lights out is equivalent to finding $\mathbf{x}=(x_1,x_2,\cdots\cdots,x_{10})$ as:

$$a_1 + x_1 + x_3 + x_7 + x_9 = 0$$
$$a_2 + x_4 + x_6 + x_{10} = 0$$
$$a_3 + x_1 = 0$$
$$a_4 + x_2 + x_4 +_{x6} + x_{10} = 0$$
$$a_5 + x_7 + x_9 = 0$$

Because the graph is symmetric, we have:

$$a_6 + x_2 + x_4 = 0$$
$$a_7 + x_1 + x_5 + x_7 + x_9 = 0$$
$$a_8 + x_{10} = 0$$
$$a_9 + x_1 + x_5 + x_7 = 0$$
$$a_{10} + x_2 + x_4 + x_8 + x_{10} = 0$$

We get:

$$x_1 = a_3$$
$$x_2 = a_2 + a_4$$
$$x_3 = a_1 + a_3 + a_5$$
$$x_4 = a_2 + a_4 + a_6$$
$$x_5 = a_3 + a_5 + a_7$$
$$x_6 = a_4 + a_6 + a_8$$
$$x_7 = a_5 + a_7 + a_9$$
$$x_8 = a_6 + a_8 + a_{10}$$
$$x_9 = a_7 + a_9$$
$$x_{10} = a_8 \quad (\operatorname{mod} 2)$$ **[3]**

Conclusion: The expressions given above help us to figure out which buttons should be pushed in the last row of C'_x , in order to get C''_x.That means if given a winnable C_x, use step 1(perform *P*), step 2(perform $\mathbf{x}$) and step 3(perform *P* again), and we will get C_0.

**2.4.2 ***When not always winnable:

In fact, when an m×n game is not always winnable, the system of equations might be insolvable.

For example, in the 5×5 game:

We get:

$$
\begin{aligned}
&a_1 + x_2 + x_3 + x_5 = 0 \qquad ① \\
&a_2 + x_1 + x_2 + x_3 = 0 \qquad ② \\
&a_3 + x_1 + x_2 + x_4 + x_5 = 0 \qquad ③ \\
&a_4 + x_3 + x_4 + x_5 = 0 \qquad ④ \\
&a_5 + x_1 + x_3 + x_4 = 0 \qquad ⑤ \quad \text{(mod 2)}
\end{aligned}
$$

②+④-③，we get： $a_3 = a_2 + a_4$ ⑥

①+⑤-③，we get： $a_3 = a_1 + a_5$ ⑦

However, ⑥⑦ are not always true. We call ⑥⑦ “the premise of winnable”.

When the “premise” is true, we get:

$$
\begin{aligned}
x_3 &= a_2 + x_1 + x_2 \\
x_4 &= a_1 + a_2 + a_3 + x_2 \\
x_5 &= a_1 + a_2 + x_1
\end{aligned}
$$

(Here x_1, x_2 are two free variables. When playing a game, we usually set them zeros)

Conclusion: A winnable configuration in 5×5 game can be transformed into C_0, with the expressions given above.

More generally, in an m×n game, we have:

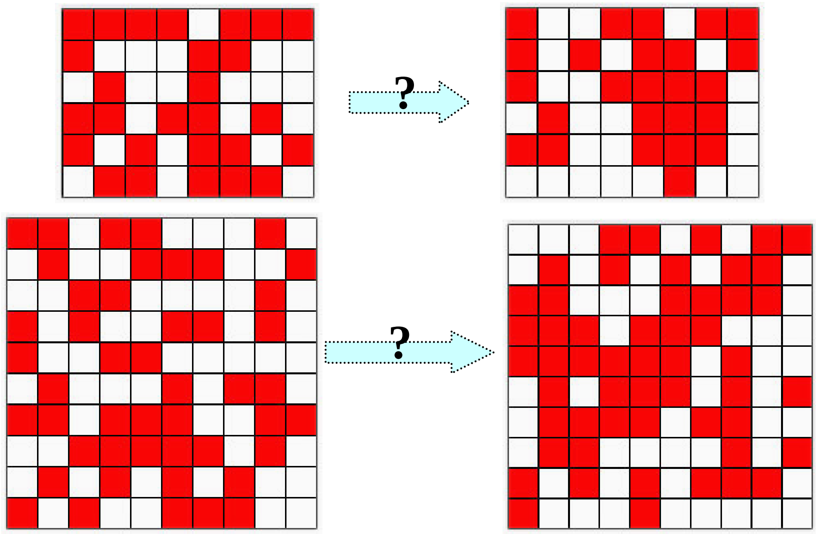
*THEOREM $1'$: C_x is winnable, if and only if its “premise of winnable” is true.

Thus computing a system of equations about a_i, x_i in a specific game, we will get the “premise”, which helps to judge whether a configuration C_x is winnable.

Special cases: We discovered an interesting conclusion: in 3×2, 7×2, 5×3, 4×4, 8×6 games, the “premises” are $\mathbf{s} = \mathbf{0}$. That means, C_x is winnable, if and only if it can be transformed into C_0 straightly by performing *P*. This will be simpler compared to using linear algebra and exhaustion.

3. *Classification of the Configurations

When a m×n game is not always winnable, we notice that for a given C_x, if we perform all possible $\mathbf{x}$ on C_x, we get many final configurations. But they can't cover all 2^{mn} cases. That means some configurations can't be transformed into one another by pushing a set of buttons. So we wonder if we can transform the following left configuration to the right one.



We've done some research on these questions and have the following results.

Definition:

Isomorphism: In an m×n game, if C_x can be transformed into C_y by pushing a set of buttons, then we call C_x and C_y are isomorphic.

1. Isomorphism has transitivity and symmetry.

2. C_x is winnable when it's isomorphic with C_0, otherwise it's unwinnable.

In an m×n game, if the isomorphic configurations are put into a same set, all the configurations can be divided into sets $S_1, S_2, \cdots\cdots, S_r\,(S_i \cap S_j = 0,\ \ i \neq j)$.According to the definition of isomorphism, if $C_x \in S_i\,(1 \leq i \leq r)$, then $C_x + A\mathbf{x} \in S_i$ (where $\mathbf{x}$ can be any column vector, mod2). $S_1, S_2, \cdots\cdots, S_r$ are called the classes of the game, and the number of the classes is represented by r. Denote the orthogonal basis of $A\mathbf{x} = 0$ by $\mathbf{n}_1, \mathbf{n}_2 \cdots\cdots \mathbf{n}_k$, and we can get:

*THEOREM 3: C_x and C_y are isomorphic, if and only if $\Delta\mathbf{b}$ is perpendicular to $\mathbf{n}_1,\mathbf{n}_2\cdots\cdots\mathbf{n}_k$. Where $\Delta\mathbf{b}$ is the column vector of $C_y - C_x$.

Proof: C_x and C_y are isomorphic $\Leftrightarrow$ there is a solution $\mathbf{x}$ of the equation $\mathbf{b}_x + A\mathbf{x} = \mathbf{b}_y$ $\Leftrightarrow$ $A\mathbf{x} = \mathbf{b}_y - \mathbf{b}_x = \Delta\mathbf{b}$ .It can be inferred according to THEOREM 1.

*THEOREM 4: We define $\mathbf{c} = (c_1, c_2, \cdots\cdots, c_k) = \mathbf{b}^T(\mathbf{n}_1, \mathbf{n}_2, \cdots\cdots, \mathbf{n}_k)$ as the bases of classification , then $S_1, S_2, \cdots\cdots, S_r$ can be and can only be divided according to the value of $\mathbf{c}$

Illustrated by the 5×5 game, THEOREM 4 can be represented as:

$$S_1 = \{C_x \mid \mathbf{c} = (0,0)\} \quad S_2 = \{C_x \mid \mathbf{c} = (1,0)\}$$
$$S_3 = \{C_x \mid \mathbf{c} = (0,1)\} \quad S_4 = \{C_x \mid \mathbf{c} = (1,1)\}$$

In order to explain there are specific C_x existing that belong to the sets mentioned respectively, we give four following examples. It can be proved that they belong to S_1, S_2, S_3, S_4 respectively.

$$C_{x1} = \begin{pmatrix} 0&0&0&0&0\\0&0&0&0&0\\0&0&0&0&0\\0&0&0&0&0\\0&0&0&0&0 \end{pmatrix} C_{x2} = \begin{pmatrix} 1&0&0&0&0\\0&0&0&0&0\\0&0&0&0&0\\0&0&0&0&0\\0&0&0&0&0 \end{pmatrix} C_{x3} = \begin{pmatrix} 0&1&0&0&0\\0&0&0&0&0\\0&0&0&0&0\\0&0&0&0&0\\0&0&0&0&0 \end{pmatrix} C_{x4} = \begin{pmatrix} 1&1&0&0&0\\0&0&0&0&0\\0&0&0&0&0\\0&0&0&0&0\\0&0&0&0&0 \end{pmatrix}$$

The proof of THEOREM 4 is in appendix.

We try using Method of *Row by Row* and discovered that THEOREM 4 has another form, which requires fewer calculations.

*THEOREM $3'$: C_x and C_y are isomorphic, if and only if the “premise of winnable “ is true for $C_y - C_x$

*THEOREM $4'$: let $(1)(2)\cdots\cdots\cdots(\mathrm{k})$ be the “premise of winnable”, then $S_1, S_2, \cdots\cdots, S_r$ can be and can only be divided according to the conditions of $(1)(2)\cdots\cdots\cdots(\mathrm{k})$.

In the 5×5 game, where k=2, this theorem is described as:

$$S_1 = \{C_x \mid a_3 = a_2 + a_4, a_3 = a_1 + a_5\}$$
$$S_2 = \{C_x \mid a_3 = a_2 + a_4, a_3 \neq a_1 + a_5\}$$
$$S_3 = \{C_x \mid a_3 \neq a_2 + a_4, a_3 = a_1 + a_5\}$$
$$S_4 = \{C_x \mid a_3 \neq a_2 + a_4, a_3 \neq a_1 + a_5\}$$

For an m×n game, according to THEOREM 4, we get:

*DEDUCTION 1: If the dimension of Null(A) is k, then $r = 2^k$

*DEDUCTION 2: the solution $\mathbf{x}$ of $A\mathbf{x} = \Delta\mathbf{b}$ is the operation that transforms C_x into C_y.

*DEDUCTION 3: $|S_1| = |S_2| = \cdots\cdots = |S_r|$, where the number of the elements in S_i is represented by $|S_i|$ (A proof is in the appendix.)

*DEDUCTION 4: The possibility of solving a randomly generated C_x is $\frac{1}{2^k}$.

4. Generalization of Lights Out Game

4.1 Generalization of Rules

4.1.1 Cycling Game (Please open the "5×5 cycling game.exe")

The effect of pushing a button is not restricted by the frame of a game. The effect can fly to another side of the graph. However, in the cycling game, the effect of pressing the angular points, edge points and interior points is alike.

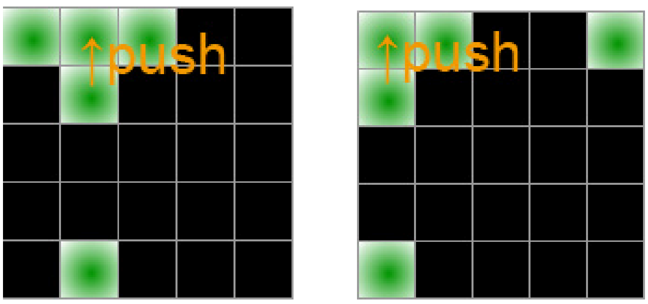


Figure 5.1 rules of Cycling Game

Solving Method: **1**.the **exhaustion** is similar.

2. Linear Algebra:

$$A=\begin{pmatrix} B & I & O & O & I \\ I & B & I & O & O \\ O & I & B & I & O \\ O & O & I & B & I \\ I & O & O & I & B \end{pmatrix}\text{,where } B=\begin{pmatrix} 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 \end{pmatrix}$$

Here $A^T = A$ .We know that A^{-1} does not exist, so we compute $RA = E$. The orthogonal basis of $A\mathbf{x} = \mathbf{0}$ is $\mathbf{n}_1, \mathbf{n}_2 \cdots\cdots \mathbf{n}_8$. Similarly, THEOREM 1,2,3,4 are still true. If $\mathbf{b}$ is winnable, it has 2^8=64 strategies.

3.* Method of *Row by Row*

We should pay attention that:

1. In the original game, C'_x has lights of the first row on. But in this game it has row 1 and 2 on. Because if we turn row 2's lights out, we should push row 1's buttons, which will lead to some lights of the last row's on.

2. If we want to operate until only row 1 and 2's lights on, the operation *P* is not unique. Because if we want to turn the lights in the last row off, not only can we push the buttons in the last but one row, but we can also push the buttons in the first row.

So we stipulate not to press the buttons in the first and last row in *P*, therefore *P* is unique to a given C_x .

"Premise of winnable":

$$a_{11} = a_{12} = a_{13} = a_{14} = a_{15}$$

$$a_{21} = a_{22} = a_{23} = a_{24} = a_{25}$$

Solutions of equations:

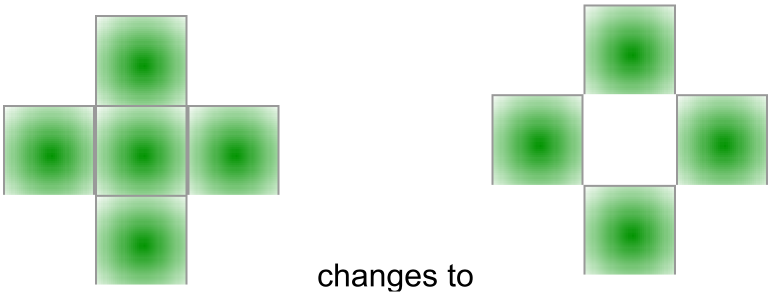
$$x_{11} = x_{12} + x_{13} + x_{14} + x_{15} + a_{11}$$

$$x_{51} = x_{52} + x_{53} + x_{54} + x_{55} + a_{21}$$

Free variables are $x_{12}, x_{13}, x_{14}, x_{15}, x_{52}, x_{53}, x_{54}, x_{55}$, thus in a winnable C_x, the number of solutions is 2^8=64. THEOREM 1',3',4' are still true.

4.1.2 Lit-only game [5] (please open the "5×5 lit-only game.exe")

We take the 5×5 game as an instance, in the original lights out game, pushing a button will change the on/off state of the light and its neighbors. But in the lit-only game, lights itself won't be changed. That is:



Firgure 6.2 the effect of pressing a button in the lit-only game

Solving methods: **1**. Also, the **exhaustion** is alike.

2. Linear Algebra

$$A = \begin{pmatrix} B & I & O & O & O \\ I & B & I & O & O \\ O & I & B & I & O \\ O & O & I & B & I \\ O & O & O & I & B \end{pmatrix} \text{,where } B = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

Here still $A^T = A$. We compute $RA = E$. Then we work out the orthogonal basis:

$\mathbf{n}_1$=(0,0,0,0,1,0,0,0,1,0,0,0,1,0,0,0,1,0,0,0,1,0,0,0,0)

$\mathbf{n}_2$=(0,0,0,1,0,0,0,1,0,1,0,1,0,1,0,1,0,1,0,0,0,1,0,0,0)

$\mathbf{n}_3$=(0,0,1,0,0,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,0,0,1,0,0)

$\mathbf{n}_4$=(0,1,0,0,0,1,0,1,0,0,0,1,0,1,0,0,0,1,0,1,0,0,0,1,0)

$\mathbf{n}_5$=(1,0,0,0,0,0,1,0,0,0,0,0,1,0,0,0,0,0,1,0,0,0,0,0,1)

Therefore, we can judge whether C_x is winnable, write down all the strategies, judge whether C_x and C_y are isomorphic and classify the 5×5 lit-only games, respectively, according to the THEOREM 1,2,3,4.

3. *Method of *Row by Row*

Here Method of *Row by Row* is easier. *P* is the same as the original game, which means not pushing any buttons in the last row, and pushing buttons row by row until only lights in the first row are on. Finally we find out that for any $\mathbf{x} = (x_{51}, x_{52}, x_{53}, x_{54}, x_{55})$, we can always get $\mathbf{s} = (0,0,0,0,0)$. So 5×5 lit-only game is the same as the 4×4 original game: C_x is winnable, if and on if C_x can be transformed into C_0 straightly by performing *P*.

Evidently, in some occasions, Method of *Row by Row* is better than linear algebra.

4.2 Generalization of N-Color Game

It has early been questioned whether the Lights Out game varies if each light has more than 2 states. This can be described as N-color game. Starting from white, once pushed, a light changes its color from white (color 0) to color 1, 2, …… N-1, then white again. One game of this kind has been released and named “Lights Out 2000”(3 colours,5×5 grids).

Solving methods:

1. $\pm$1 Exhaustion does not work, while using **addition of matrices** still makes sense.

2. Linear Algebra:

Matrix A is the same as that in the 2-colour game.

$$A = \begin{pmatrix} B & I & O & O & O \\ I & B & I & O & O \\ O & I & B & I & O \\ O & O & I & B & I \\ O & O & O & I & B \end{pmatrix} \quad B = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 \end{pmatrix}$$

Turning all lights out is equivalent to solving a formula: $A\mathbf{x} + \mathbf{b} = \mathbf{0}$, where each number can be 0, 1 and 2. Still define E as the Gauss-Jordan echelon form, and $RA = E$.

Find out the orthogonal basis of $A\mathbf{x} = \mathbf{0}$:

$$\mathbf{n}_1 = (0,2,1,1,1,1,0,2,0,1,2,1,0,2,1,2,0,1,0,2,2,2,2,1,0)^T$$

$$\mathbf{n}_2 = (2,0,1,0,2,1,0,2,0,1,0,0,0,0,0,2,0,1,0,2,1,0,2,0,1)^T$$

$$\mathbf{n}_3 = (1,1,2,2,1,1,2,1,1,0,2,1,0,2,1,2,1,2,2,0,1,0,1,0,0)^T$$

THEOREM 1: A configuration C_x is winnable if and only if $\mathbf{b}$ is perpendicular to $\mathbf{n}_1, \mathbf{n}_2, \mathbf{n}_3$.

THEOREM 2: Suppose $\mathbf{b}$ is a winnable configuration. Then its winning strategy is $\mathbf{x} = k_1\mathbf{n}_1 + k_2\mathbf{n}_2 + k_3\mathbf{n}_3$, where k_1, k_2, k_3 are integers 0,1,2.

THEOREM 3 (to judge isomorphism) does not change.

THEOREM 4 changes a little, where each number in $\mathbf{c}$ can be 0,1,2. Also THEOREM 4 works in any N-color game.

3. *Method of *Row by Row*:

Still define operation *P*, and *P* is still unique to a given C_x. Through *P* we can bring up all lights to the first row. Use $s = (a_1, a_2, a_3, a_4, a_5)$ to represent the state of first row now.

The "premise of winnable" is $a_3 = 2a_1 = 2a_5, \quad a_2 = a_4$

We get:

$$x_1 = 2a_1 + 2x_5 + x_3$$
$$x_2 = a_3 + 2a_2 + 2x_4$$

where x_3, x_4, x_5 are free variables.

Conclusion: Thus a winnable configuration in 5×5 3-color game can be turned into C_0, with the expressions given above.

4.3 *The Hexagon Game

We know that squares and hexagons are shapes of special properties, so we can see hexagonal ceramic tiles on the pavement. Inspired by it, we came up with an idea that hexagon can be the shape of every light in the lights out game. Next we want to talk about this game.

Definition:

Each light is hexagonal. Bring them all together to form a larger hexagon (like a honeycomb). If the number of lights on each side of honeycomb is N, we called this game an N-order Hexagon Game.

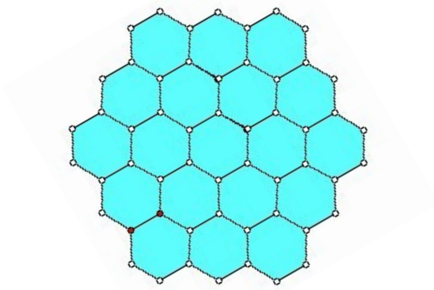


Figure7.1 N-order Hexagon Game

4.3.1 3-order Hexagon Game for example

Rules: Each light has a button. Pushing a button changes the state of itself and its neighbors.

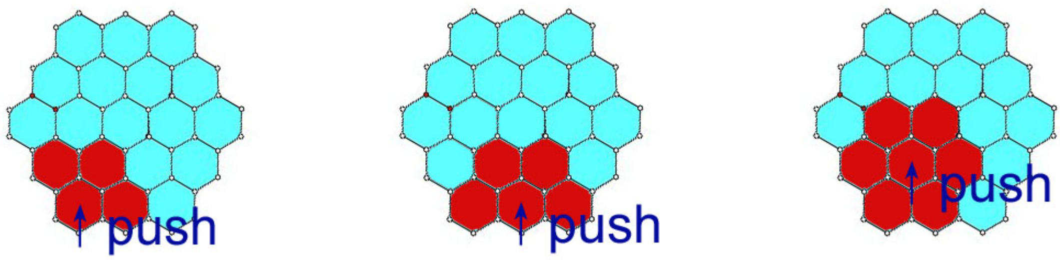


Figure 7.2 rules of 3-order Hexagon Game

To describe a configuration, we use a 5×5 matrix C_x.

$$C_x = \begin{pmatrix} a_{11} & a_{12} & a_{13} & 0 & 0 \\ a_{21} & a_{22} & a_{23} & a_{24} & 0 \\ a_{31} & a_{32} & a_{33} & a_{34} & a_{35} \\ a_{41} & a_{42} & a_{43} & a_{44} & 0 \\ a_{51} & a_{52} & a_{53} & 0 & 0 \end{pmatrix}$$

Solving methods: 1. Exhaustion:

According to the game's rules, we can write down:

$$A_{11}=\begin{pmatrix}1&1&0&0&0\\1&1&0&0&0\\0&0&0&0&0\\0&0&0&0&0\\0&0&0&0&0\end{pmatrix} A_{12}=\begin{pmatrix}1&1&1&0&0\\0&1&1&0&0\\0&0&0&0&0\\0&0&0&0&0\\0&0&0&0&0\end{pmatrix} A_{13}=\begin{pmatrix}0&1&1&0&0\\0&0&1&1&0\\0&0&0&0&0\\0&0&0&0&0\\0&0&0&0&0\end{pmatrix}$$

……

However, if (i,j) is just zero in C_x, then $A_{ij}=\mathbf{0}$

$$C_x+\sum_{1\le i,j\le 5} x_{ij}A_{ij}=C_0 \pmod 2$$

2. *Linear Algebra:

Write down A according to the rules.

$$A=\begin{pmatrix}
1&1&0&0&0&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
1&1&1&0&0&0&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&1&1&0&0&0&0&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
1&0&0&0&0&1&1&0&0&0&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0\\
1&1&0&0&0&1&1&1&0&0&0&1&1&0&0&0&0&0&0&0&0&0&0&0&0\\
0&1&1&0&0&0&1&1&1&0&0&0&1&1&0&0&0&0&0&0&0&0&0&0&0\\
0&0&1&0&0&0&0&1&1&0&0&0&0&1&1&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&1&0&0&0&0&1&1&0&0&0&1&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&1&1&0&0&0&1&1&1&0&0&1&1&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&1&1&0&0&0&1&1&1&0&0&1&1&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&1&1&0&0&0&1&1&1&0&0&1&1&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&1&0&0&0&0&1&1&0&0&0&1&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&1&1&0&0&0&1&1&0&0&0&1&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&1&1&0&0&1&1&1&0&0&1&1&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&1&1&0&0&1&1&1&0&0&1&1&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&1&1&0&0&1&1&0&0&0&1&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&1&0&0&0&1&1&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&1&0&0&1&1&1&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&1&0&0&1&1&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0
\end{pmatrix}$$

Perform Gauss-Jordan elimination on A. That will yield $RA=E$, where E is Gauss-Jordan echelon form, and R is the product of the product of the elementary matrices which perform the reducing row operations.

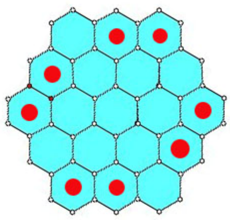
The orthogonal basis of $A\mathbf{x}=\mathbf{0}$:

$$\mathbf{n}_1=(0,1,1,N,N,1,0,0,0,N,1,0,0,0,1,0,0,0,1,N,1,1,0,N,N)^T$$

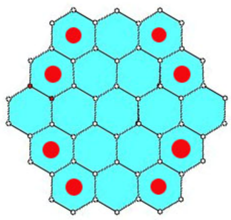
$$\mathbf{n}_2 = (1,0,1,N,N,1,0,0,1,N,0,0,0,0,0,1,0,0,1,N,1,0,1,N,N)^T$$

$$\mathbf{n_3} = (0,0,0,N,N,1,1,1,1,N,1,1,0,1,1,1,1,1,N,0,0,0,N,N)^T$$

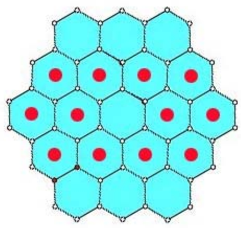
"N"s stands for nothing because there are no lights in the corresponding positions. We can mark $\mathbf{n}_1,\mathbf{n}_2,\mathbf{n}_3$ on the graph. Starting from C_0, and pushing the buttons which are marked a red point, we get C_0 again.



$\mathbf{n}_1$



$\mathbf{n}_2$



$\mathbf{n}_3$

Because $A^T = A$, THEOREM 1 is true as well.

THEOREM 1: A configuration C_x is winnable if and only if b is perpendicular to the two vectors $\mathbf{n}_1,\mathbf{n}_2,\mathbf{n}_3$

THEOREM 2 changes a little. We make $x' = (x_{11}, x_{12}, \cdots\cdots, x_{53}, 0,0,0,0,0,0)$ be another form of $\mathbf{x}$. To find out a special solution vector $\mathbf{x}_0$, we make $x_{5,2} = x_{5,3} = 0$, so it's possible to prove that:

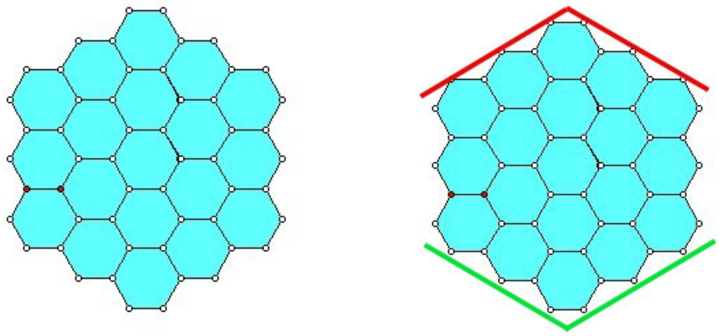
$$\mathbf{x}_0' = E\mathbf{x_0} = RA\mathbf{x}_0 = R\mathbf{b}$$

THEOREM2: Suppose $\mathbf{b}$ is a winnable configuration. Then the eight winning strategies for $\mathbf{b}$ are

$$R\mathbf{b}, R\mathbf{b}+\mathbf{n}_1', R\mathbf{b}+\mathbf{n}_2', R\mathbf{b}+\mathbf{n}_3', R\mathbf{b}+\mathbf{n}_1'+\mathbf{n}_2',$$
$$R\mathbf{b}+\mathbf{n}_1'+\mathbf{n}_3', R\mathbf{b}+\mathbf{n}_2'+\mathbf{n}_3', R\mathbf{b}+\mathbf{n}_1'+\mathbf{n}_2'+\mathbf{n}_3'$$

3. *Method of *Row by Row*

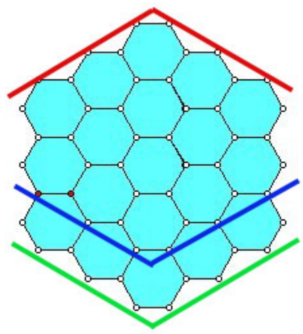
We rotate 3-order Hexagon Game into this state.



That gives us the inspiration of seeing the top five lights as the first row (we draw a red line to represent it) , and the bottom five lights as the last row (green line).

Definition:

By analogy, we define *P* as “do not push any button in the last row, and then push a set of buttons until the only lights which are on, are in the first row”.



Perform *P* on C_x , then we get C'_x . Use row vector $\mathbf{s} = (a_1, a_2, a_3, a_4, a_5)$ to represent the state of the first row from left to right.

We can prove the *P* is unique to a given C_x .

Proof:

We define the last row as row 5,the row above it as row 4 (blue line) and the lights as (5,1)(5,2)……(5,5) and (4,1)(4,2)……(4,5) from left to right. In a given C_x , pushing (4,3) is the only way to turn off (5,3). So when performing *P*, whether pushing (4,3) or not is determined by the state of (5,3). After (5,3) is off, there is no necessity to push (4,3) once more, which means whether pushing (4,2) and (4,4) or not is determined by the state of (5,2) and (5,4). So do (4,1) and (4,5).

All the above shows that given an original state of a specific row, there is only one way to turn all its lights off by pushing the buttons in the row above it.

Thus, row 4 is determined by row 5. Row 3 is determined by row 4. We can prove in a similar way that there is only one way now to bring all lights up to row 1.

Here come the steps of transforming a winnable C_x into C_0 . Step 1: perform *P*, step

2: use $\mathbf{x}=(x_1,x_2,x_3,x_4,x_5)$ to operate the last row, step 3: perform *P* again.

To find out the relation between $\mathbf{s}$ and $\mathbf{x}$, we get the following system of equations.

$$
\begin{aligned}
&a_1+x_3+x_4+x_5=0\\
&a_2+x_3+x_4+x_5=0\\
&a_3+x_1+x_2+x_4+x_5=0\\
&a_4+x_1+x_2+x_3=0\\
&a_5+x_1+x_2+x_3=0 \quad (\mathrm{mod}\,2)
\end{aligned}
$$

The "premise": $a_1=a_2, a_4=a_5, a_3=a_1+a_5$

$x_1=a_5+x_2+x_3$

$x_5=a_1+x_3+x_4$

x_2,x_3,x_4 are the free variables

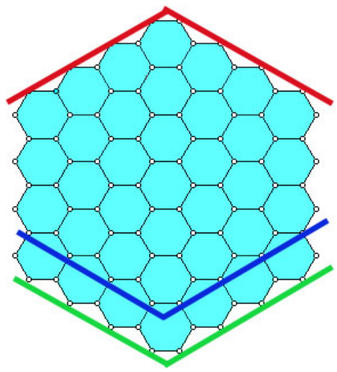
We usually set $x_2,x_3,x_4=0$, then

$x_1=a_5$

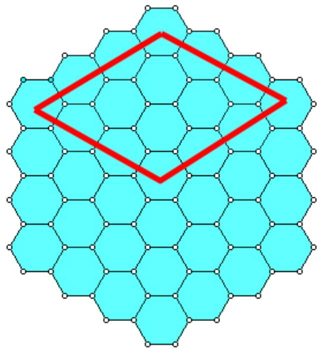
$x_5=a_1$

Conclusion: The premise helps us to judge whether C_x is winnable. If it is, perform the 3 steps with the expressions given above, and then will get C_0

4.3.2 *About N-order Hexagon Game:



We draw an N-order Hexagon Game. Using linear algebra to solve the game is just the same, while using method of *Row by Row* gives more joy. We define operation *P* as “do not push any button in the last row (drawn a green line), and then push a set of buttons until the only lights which are on, are in the first row(drawn a red line).



P is unique to a given C_x.

Proof:

It has been proved that given an original state of a specific row, there is only one way to turn all its lights off by pushing the buttons in the row immediately above it. That shows there is only one way to bring all lights up to a diamond-shaped part (shown in the second figure). It's rather easy to prove that in such a diamond-shaped part, only one way can bring all lights up to the first row. The proof is tediously long so we bypass it.

Therefore, find out the relations between $\mathbf{S}$ and $\mathbf{X}$, solve a system of equations , find out the "premise of winnable" and use the 3 steps, so an N-order Hexagon Game will be solved.

It's obvious that THEOREM 1',3',4' are true.

5. *Comparison of Four Solutions

5.1 Comparison of Advantages and Disadvantages

Solutions	Advantages	Disadvantages
±1Exhaustion	Easy to operate. It can be applied to games in any shapes or any regular rules.	Computationally intensive, inconvenient to solve higher order games and can't be applicable to games in more colors.
Exhaustion with Addition of Matrices	Easy to operate. It can be applied to games in any shapes, any regular rules or any colors.	Computationally intensive and inconvenient to solve higher order games.
Linear Algebra	Reveal the essences of the game and calculate readily. It can be applied to games in any shapes, any regular rules or any colors.	Complex to program and inconvenient to operate by hand.
Method of *Row by Row*	Fun to play. It can be applied to games in any	It should make a bit more adjustment when the games

	shapes, any regular rules or any colors.	are changed in shapes, rules and colors.

5.2 Correlation of Four Solving Methods:

1. Exhaustion with using ±1 and matrices are equivalent.

2. According to THEOREM 1,2,3,4, linear algebra and Method of *Row by Row* are equivalent. The differences are merely their ways of describing theorems. The former performs in higher algebra while the latter does in elementary algebra. This reflects the unity of higher algebra and elementary algebra.

5.3 Exploration of the Best Solutions to a Specific Game:

1. Exhaustion is suitable to games in lower order.

2. Linear algebra can reveal the essence of the game most. Moreover, it can avoid computationally intensive calculation when it's in higher order. Thus linear algebra is fit for researching on the higher order game and laws of games.

3. Method of *Row by Row* is much easier to understand and operate. Its theory is simple and practicable for normal players. Compared with computationally intensive linear algebra, Method of *Row by Row* needs only computing a system of equations. However when the games become higher orders, it will be complex to compute the equations. Method of *Row by Row* does not adapt to.

4. What's more, Method of *Row by Row* can reflect the fun of games itself. In addition, the changes of shapes (such as hexagons), Method of *Row by Row* will change correspondingly and challenge the wisdom of players.

5. In some specific cases (such as 3×2, 7×2, 5×3, 4×4, 8×6 original games and 5×5 lit-only game), C_x is winnable if and only if it can be transformed into C_0 straightly by performing *P*. We have tried researching on these specific cases but haven't found other games with this property.

6. Summary

This paper summarizes properties of lights out game and emphasizes on four solutions. We put forward two methods to classify the configurations of the game and generalize it to changes in rules, colors, shapes etc. What's more we design a new hexagon game and solve it.

Our independent research productions include:

(1) Researching on the games not always winnable by Method of *Row by Row*.

(2) Emphasize the classification of configurations and draw the conclusion: THEOREM 3 for judging the isomorphism configuration, THEOREM 4 for classify the configuration, DEDUCTION 1 for number of classification, DEDUCTION2 for strategy of isomorphism configuration.

(3) Alternate solution with Method of *Row by Row* in cycling game, lit- only game and 3-color game.

(4) Put forward the hexagon game and solve it with linear algebra and Method of *Row by Row*.

Appendices:

1. *Proof of THEOREM4:

Use $\mathbf{b}_x, \mathbf{b}_y$ to represent C_x and C_y,

$$\mathbf{c}_x = (c_{x1}, c_{x2}, \cdots\cdots, c_{xk}) = \mathbf{b}_x^T(\mathbf{n}_1, \mathbf{n}_2, \cdots\cdots, \mathbf{n}_k)$$

$$\mathbf{c}_y = (c_{y1}, c_{y2}, \cdots\cdots, c_{yk}) = \mathbf{b}_y^T(\mathbf{n}_1, \mathbf{n}_2, \cdots\cdots, \mathbf{n}_k)$$

$C_x, C_y \in Si(1 \le i \le k)$

$\Leftrightarrow C_x, C_y$ are isomorphic

$\Leftrightarrow \Delta\mathbf{b} = \mathbf{b}_y - \mathbf{b}_x, \Delta\mathbf{b}^T(\mathbf{n}_1, \mathbf{n}_2, \cdots\cdots, \mathbf{n}_k) = (0, 0, \cdots\cdots, 0)$

$\Leftrightarrow \mathbf{c}_y - \mathbf{c}_x = (0, 0, \cdots\cdots, 0)$

$\Leftrightarrow \mathbf{c}_x = \mathbf{c}_y$

$\therefore$ Theorem 4 is true.

2. *Proof of DEDUCTION 3

Deduction 3: $|S_1| = |S_2| = \cdots\cdots = |S_r|$

Proving: Set the orthogonal basis respectively:

$$\mathbf{n}_1 = (\cdots\cdots 1, 0, 0, \cdots 0)$$
$$\mathbf{n}_2 = (\cdots\cdots 0, 1, 0, \cdots 0)$$
$$\cdots\cdots$$
$$\mathbf{n}_k = (\cdots\cdots 0, 0, 0, \cdots 1)$$

Set $S_1 = \{C_x \mid \mathbf{b}^T(\mathbf{n}_1, \mathbf{n}_2 \cdots\cdots \mathbf{n}_k) = (1, 0, 0 \cdots\cdots 0)\}$

$S_2 = \{C_x \mid \mathbf{b}^T(\mathbf{n}_1, \mathbf{n}_2 \cdots\cdots \mathbf{n}_k) = (0, 0, 0 \cdots\cdots 0)\}$

For any $\mathbf{b}_x$, if C_x (represented by $\mathbf{b}_x$) $\in S_1$, then

set $\mathbf{b}'_x = \mathbf{b}_x + (\cdots\cdots 1, \underbrace{0, 0 \cdots\cdots 0}_{k-1 \quad zeros})$

then $\mathbf{b}'^T_x(\mathbf{n}_1, \mathbf{n}_2 \cdots\cdots \mathbf{n}_k) = (0, 0, 0 \cdots\cdots 0)$

It can be computed, C'_x (represented by $\mathbf{b}'_x$) $\in S_2$

$\therefore \quad |S_1| \le |S_2|$

For any $\mathbf{b}_y$, if C_y(represented by $\mathbf{b}_y$) $\in S_2$,

set $\mathbf{b}'_y = \mathbf{b}_y + (\cdots\cdots 1, \underbrace{0, 0 \cdots\cdots 0}_{k-1 \quad \text{zeros}})$

then $\mathbf{b}'^T_y(\mathbf{n}_1, \mathbf{n}_2 \cdots\cdots \mathbf{n}_k) = (1, 0, 0 \cdots\cdots 0)$

It can be computed, C'_y (represented by $\mathbf{b}'_y$) $\in S_1$

$\therefore |S_2| \le |S_1|$

$\therefore |S_1| = |S_2|$

In like manner, it can be proved that $|S_1| = |S_2| = \cdots\cdots = |S_r|$

3. *Proof of THEOREM $4'$

Represent C_x which is performed P by C_x', denote its state vector by s_x

then $C_x' = \begin{pmatrix} s_x \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{pmatrix}$

Set $P_x = C_x' - C_x \cdots\cdots ①$, it represents operation matrix P

Represent C_y which is performed P by C_y',

denote its state vector by s_y, Set $P_y = C_y' - C_y \cdots\cdots ②$

Represent $(C_y - C_x)$ which is performed P by C_{y-x}',

denote its state vector by s_{y-x}, Set $P_{y-x} = C_{y-x}' - (C_y - C_x) \cdots\cdots ③$

then ② − ①, we can get: $P_y - P_x = C_y' - C_x' - (C_y - C_x) \cdots\cdots ④$

$\because$ for $(C_y - C_x)$, P is unique

$\therefore$ According to ③④, $P_{y-x} = P_y - P_x$

$\therefore C_{y-x}' = C_y' - C_x'$

$s_{y-x} = s_y - s_x$

C_x and C_y are isomorphic $\Leftrightarrow$ the "premise of winnable" is true for s_{y-x}

$\Leftrightarrow$ the situations of equations(1)(2)$\cdots\cdots\cdots$(k) for C_x and C_y are the same

$\therefore$ theorem $4'$ is true.

Attachment Explanation:

"5×5　3-color game.exe", "3-order Hexagon Game(for linear algebra)", "3-order Hexagon Game(for row by row)"is provided by teammate Huang Yifeng.

"3×3 Exhaustion with addition of matrices.exe";

"5×5 cycling game.exe";

"5×5 lit-only game.exe";

All LightsOutPuzzles.exe is done by net friend 金眼睛.

10×10 Lights Out.swf is downloaded from the Internet.

Reference:

1. http://mathworld.wolfram.com/LightsOutPuzzle.html（July, 2009）

2. Marlow Anderson; Todd Feil, Mathematics Magazine, Turning Lights Out with Linear Algebra, Vol.71, No.4,pp, 300-303.(Oct.,1998)

3. 魔方吧·中国魔方俱乐部 » ★ 数学、算术趣题 ★ » 点灯游戏的解法讨论 20#　net friend "lulijie" put forward method of *row by row*.

http://bbs.mf8.com.cn/viewthread.php?tid=32759&extra=&page=3 （July 2009）

4. Jennie Missigman; Richard Weida, An Easy Solution to Mini Lights Out, Mathematics Magazine. Vol. 74, No.1, pp. 57-59. (Feb., 2001)

5. Does the lit-only restriction make any difference for the σ-game andσ+-game? Joint work with John Goldwasser and　王新茂