Closed-Form Solution to the Lights Out Puzzle

Sahand Rabbani

Lights Out is a family of hand-held electronic puzzles by Tiger Toys featuring a contiguous arrangement of red lighted buttons. Each puzzle presents some lighted pattern of these buttons, where the object of the game is to turn off all of the lights by pressing the correct sequence of buttons.

In most variations of the game, pressing a button will toggle the lighted state of that button as well as the state of the four buttons immediately adjacent to it. In some variations, buttons located on the edge of the grid may toggle the lights on the opposite side of the grid through a wrap-around effect, while, in other variations, the effect of the edge buttons is truncated.

The classic Lights Out game features a five-by-five grid without wrap-around. One variation called Lights Out Cube features a cube with a three-by-three grid of lights on each of its six faces, while another variation called Lighths Out Mini is a two-dimensional four-by-four grid with wrap-around effects.

Here, we will derive a closed-form solution to the family of Lights Out puzzles. In the discussion that follows, the terms "button" and "light" are used interchangeably. First, we will generalize the game to one of N ordered buttons, numbered the 1 through N. The starting configuration, which we deem the *problem*, is a lighted subset of these N buttons. We seek a pattern of buttons, a *solution*, that will extinguish all of the lights. Our generic game adheres to two underlying rules:

Rule 1. Pressing the button i will toggle the state of a deterministic subset S_i of the N buttons.

Rule 2. Every light j in the subset of lights S_i, when pressed, toggles the button i. Furthermore, *only* the lights j in the subset of lights S_i can toggle the button i.

We can now fully characterize any Lights Out game by enumerating the subsets S_i for every $i \in \{1, 2, \ldots, N\}$. A convenient way to summarize these subsets is in matrix form. Let us define a N-by-N matrix $\mathbf{M}$ whose j^{th} column contains the information in the subset S_j. Specifically, we express the i^{th} row of the j^{th} column of $\mathbf{M}$:

$$\mathbf{M}_{ij} = \begin{cases} 1 & , \quad \text{if } i \in S_j \\ 0 & , \quad \text{otherwise} \end{cases} \tag{1}$$

Note that because of Rule (2), $\mathbf{M} = \mathbf{M}^T$. We can formulate a problem as an N-point column vector, p, whose i^{th} element equals one if light i is initially on and equals zero if light i is initially off. Before we formulate the solution mathematically, we must first note two important features of the game:

1. Due to linearity, the order in which a set of buttons is pressed will not affect the final outcome.

2. Pressing a button twice is equivalent to not pressing that button at all.

The second observation implies that pressing a button an even number of times is equivalent to not pressing that button at all, and pressing a button an odd number of times is equivalent to pressing that button once. From these observations we realize that we can fully characterize a solution with a N-point vector, s, whose i^{th} element equals one if the solution entails pressing button i and equals zero otherwise.

Finally, we define the binary mathematical set in modulo two $\mathbb{Z}_2 = \{0, 1\}$, in which we will perform our analysis. It is important to note that $\mathbb{Z}_2$ forms a mathematical field, so that we may extend the operations of multiplication and addition from the set of real numbers. We note that in the field $\mathbb{Z}_2$, the matrix $\mathbf{M}$ maps a series of button presses, $x \in {\mathbb{Z}_2}^N$, to a light configuration $y \in {\mathbb{Z}_2}^N$. Given a game defined by $\mathbf{M} \in {\mathbb{Z}_2}^{(N,N)}$ and a problem defined by $p \in {\mathbb{Z}_2}^N$, we seek all of the solutions, $s \in {\mathbb{Z}_2}^N$, such that when these buttons are applied to the configuration p, the result is a completely dark board, a N-point vector whose elements are all zero. Formally, we seek all s such that,

$$p + \mathbf{M}s = 0 \tag{2}$$

We are ready for our first theorem.

Theorem 1. Given a game specified by the mapping matrix $\mathbf{M}$ and a problem specified by the vector p, a solution s will satisfy,

$$\mathbf{M}s = p \tag{3}$$

Proof. Adding p to both sides of Equation (2) gives,

$$p + \mathbf{M}s + p = p$$

The result follows from the commutative property of addition in $\mathbb{Z}_2$, and from the identity $a + a = 0$.

We can now solve for at least one s by Gauss-Jordan elimination in $\mathbb{Z}_2$. However, to find all values of s, we must first identify the rank, R, of the mapping matrix $\mathbf{M}$, which can be determined from the number of nonzero rows in its echelon form. Let us call the dimension of the null space, or the number of zero rows in the echelon form, U. It follows that $N = R + U$. A vector ν is in the null space of $\mathbf{M}$ if it satisfies,

$$\mathbf{M}\nu = 0 \tag{4}$$

The basis of the null space is described by U unique vectors, which, when combined linearly, can form 2^U unique vectors that satisfy Equation (4). Let us define a N-by-U matrix $\mathbf{N}$ whose column space is the null space of $\mathbf{M}$. We can now determine all of the solutions to a given problem.

Theorem 2. If s is a solution, then $s + \mathbf{N}b$ is also a solution, for any $b \in {\mathbb{Z}_2}^U$.

Proof. To be a solution, $s' = s + \mathbf{N}b$ must satisfy:

$$\mathbf{M}s' = p$$

Substituting for s' gives,

$$\mathbf{M}(s + \mathbf{N}b) = \mathbf{M}s + \mathbf{MN}b = p$$

Since $\mathbf{MN} = \mathbf{0}$ and $\mathbf{M}s = p$, the condition is satisfied.

We can take this theorem a step further:

Theorem 3. If s is a solution and s' is also a solution, then there is some $b \in {\mathbb{Z}_2}^U$ such that $s' = s + \mathbf{N}b$.

Proof. This is equivalent to stating that if both s and s' are solutions, then $s' + s = \mathbf{N}b$, which is to say that $t = s' + s$ is in the null space of $\mathbf{M}$. This is easily shown:

$$\begin{aligned} \mathbf{M}t &= \mathbf{M}(s' + s) \\ &= \mathbf{M}s' + \mathbf{M}s \\ &= p + p \\ &= 0 \end{aligned}$$

And the assertion is proved.

Theorems (2) and (3) taken together imply that if there is at least one solution, then there are exactly 2^U solutions. This raises the question of solvability: How do we determine whether there exists some s that satisfies Equation (3)?

Theorem 4. For a game specified by $\mathbf{M}$ and a problem specified by p, there exists a solution s if and only if $\mathbf{N}^T p = 0$; that is, if and only if p is orthogonal to the null space of $\mathbf{M}$.

Proof. Since s is a solution,

$$\begin{aligned} \mathbf{M}s &= p \\ \mathbf{N}^T\mathbf{M}s &= \mathbf{N}^T p \\ 0 &= \mathbf{N}^T p \end{aligned}$$

Thus, $(\mathbf{N}^T p = 0) \leftrightarrow (\exists s \mid \mathbf{M}s = p)$.

For choices of $\mathbf{M}$ and p that admit a solution, it is convenient to write s in closed form:

$$s = \mathbf{W}p \tag{5}$$

where $\mathbf{W}$ is the "pseudoinverse" of $\mathbf{M}$. We can solve for $\mathbf{W}$ by Gauss-Jordan Elimination. We first express the matrix $[\mathbf{M}\ \mathbf{I}]$, in "pseudoechelon" form, where $\mathbf{I}$ is the identity matrix. Pseudoechelon form is the same as echelon form except that the zero rows are swapped with nonzero rows so that in every nonzero row i, the leading term is in the i^{th} column. The matrix $\mathbf{W}$ is the right half of the pseudoechelon matrix of $[\mathbf{M}\ \mathbf{I}]$. We can summarize our closed-form solution to the Lights Out problem as follows:

For a Lights Out game described by the mapping matrix $\mathbf{M}$ and a problem described by the vector p:

1. The problem p is solvable if and only if $\mathbf{N}^T p = 0$.
2. If the problem p is solvable, the only solutions are $s = \mathbf{W}p + \mathbf{N}b$ for all $b \in {\mathbb{Z}_2}^U$.

The pseudoinverse matrix $\mathbf{W}$ is easily derived from the mapping matrix $\mathbf{M}$, as mentioned above. The basis for the null space $\mathbf{N}$ can be determined from the pseudoechelon form of $\mathbf{M}$, which we will refer to as $\mathbf{M}_e$. Specifically, we collect the i^{th} columns of $\mathbf{M}_e$ for all i such that the i^{th} row of $\mathbf{M}_e$ is zero. The i^{th} element of these column vectors will necessarily be zero. We replace each i^{th} element with one, and the resulting U vectors form the basis for the null space of $\mathbf{M}$, which we concatenate to form $\mathbf{N}$. We now have a closed-form solution to the Lights Out family of puzzles as defined by Rules (1) and (2).

A Mapping, Pseudoinverse, and Null Matrices for the Classic Lights Out Puzzle

Figure 1: Classic five-by-five square Lights Out game

Figure (1) depicts the classic five-by-five Lights Out puzzle. In this game, the nine central buttons each toggle five lights: itself and the four buttons north, south, east, and west of it. The nine buttons on the edge toggle only four lights instead of five, and four buttons on the corner toggle three. If we take the set of twenty-five buttons columnwise from the game grid, we can determine the matrices $\mathbf{M}$, $\mathbf{W}$, and $\mathbf{N}$ for this game.

The dimension of the null space for this game is $U = 2$, implying that every solvable problem has exactly $2^U = 4$ unique solutions and, furthermore, that only $2^{-U} = 1/4$ of the games are solvable.

$$\mathbf{M} = \begin{pmatrix}
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 \\
1 & 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 & 1 & 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 & 1 & 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 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 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 \\
0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 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 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 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 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 \\
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 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 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 & 0 & 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 & 0 & 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 & 0 & 0 & 0 & 1 & 1
\end{pmatrix} \tag{6}$$

$$
\mathbf{W} = \begin{pmatrix}
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 1 \\
0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\
0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 \\
0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 0 \\
0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \\
0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\
0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 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 & 1 \\
0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\
0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 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 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{pmatrix} \tag{7}
$$

$$
\mathbf{N} = \begin{pmatrix}
0 & 1 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 0 \\
1 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 1
\end{pmatrix}^T \tag{8}
$$

B Mapping, Pseudoinverse, and Null Matrices for the Lights Out Mini Game

Figure 2: Four-by-four Lights Out Mini game

Figure (2) depicts the four-by-four Lights Out Mini puzzle. In this game, there is wrap-around: the northernmost and southernmost rows are considered adjacent, as are the easternmost and westernmost columns. Thus, every button toggles five lights: itself and the four buttons in the adjacent cardinal directions. If we take the set of sixteen buttons columnwise from the game grid, we can determine the matrices **M** for this game:

$$\mathbf{M} = \begin{pmatrix}
1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1
\end{pmatrix} \tag{9}$$

It turns out that for this game, the dimension of the null space is $U = 0$, implying that every problem is solvable and has exactly one solution. Furthermore, it turns out that $\mathbf{M} = \mathbf{W}$, which admits a remarkable method for finding the solution to a problem: We need only remember the positions of the initially lighted buttons. After pressing each of them once, the resulting configuration betrays the unique solution to the puzzle. We remember these new lighted buttons and replay the game, pressing each one—and the problem is solved without the use of a calculator.

C Other Variations

There are other variations of the Lights Out game, including the aforementioned Lights Out Cube, with fifty-four lights, and the slightly more sophisticated Lights Out Deluxe, a thirty-six-light grid which in itself contains a few variations of the game. We can determine the mapping and pseudoinverse matrices corresponding to these games using the methods discussed earlier, but reproducing these large matrices here would not enlighten us further.

However, one variation worth noting is the Lights Out 2000 game. Each light in this five-by-five grid can assume three states, instead of two: off, red, and green. Pressing a button will cycle that button and its four cardinal neighbors through these three states in order. As in the other variations, the objective of the game is to turn off all of the lights.

This game does not fall under the category of the generic Lights Out puzzle that we have solved, but we may extend the tools of linear algebra and modulo arithmetic to set up the problem. We must consider the set in modulo three $\mathbb{Z}_3 = \{0, 1, 2\}$. Note that this set does not form a mathematical field because it fails the test of multiplicative inversion. This does not, however, hinder our ability to derive a pseudoinverse, since every element of the mapping matrix is still either one or zero.

We cannot directly use Equation (3), since the property $a + a = 0$ does not hold in modulo three. Therefore, we step back to Equation (2) and subtract the problem vector from both sides:

$$\mathbf{M}s = -p \tag{10}$$

Subtraction is modulo three admits $-0 = 0$, $-1 = 2$, and $-2 = 1$. From $\mathbf{M}$, we determine the pseudoinverse matrix $\mathbf{W} \in {\mathbb{Z}_2}^{(N,N)}$ as usual. The null matrix $\mathbf{N} \in {\mathbb{Z}_3}^{(N,U)}$ is determined, with a little bit of work, from the echelon form of $\mathbf{M}$. As in the previous case, the test for solvability is $\mathbf{N}^T p = 0$. The solutions, if they exist, are,

$$s = \mathbf{W}(-p) + \mathbf{N}b,\ \forall b \in {\mathbb{Z}_3}^U. \tag{11}$$

Unfortunately, we cannot realistically address the countless other variations, but the important realization is that since these games are linear, they can be solved with relative ease afforded by the powerful and well developed tools of linear algebra.