Rose-Hulman Undergraduate Mathematics Journal

A Non-geometric Switch Toggling Problem

Megan Duke [a]

Volume 14, no. 2, Fall 2013

Sponsored by

Rose-Hulman Institute of Technology
Department of Mathematics
Terre Haute, IN 47803
Email: mathjournal@rose-hulman.edu
http://www.rose-hulman.edu/mathjournal

[a]Muskingum University, 199 Stormont Street, New Concord, Ohio 43762, USA. email: mduke@muskingum.edu

Rose-Hulman Undergraduate Mathematics Journal
Volume 14, no. 2, Fall 2013

A Non-geometric Switch Toggling Problem

Megan Duke

Abstract. Switch toggling games such as Lights Out and the σ^+-game are widely studied in mathematics and have been applied to model a variety of situations such as genetic networks and cellular automata. This paper introduces a class of toggling games where at each iteration a fixed number of switches is chosen to be toggled where the only switches changed are the switches chosen to be toggled. The switches all operate independently of each other and do not depend on the proximity or the position relative to any other switch. This paper classifies the conditions necessary and the steps taken to transition from all switches in the on state to all switches in the off state. Further results include the conditions required of the parity between the number of switches in the system and the fixed number of switches toggled at each step in order to transition from a given initial state to a specified terminal state.

Acknowledgements: I would like to thank Dr. Richard Daquila for all of his help and support during this project as a part of the Muskie Fellowship program at Muskingum University. I would also like to thank Rose-Hulman Undergraduate Math Journal for giving me this opportunity as well as my parents for their continued love and support.

1 Introduction

Switch toggling games are widely studied in a variety of fields of mathematics. Some of the most famous examples of switch toggling games are Lights Out, the classic Toggling Light Switch game, and σ^+-game. The state of the switches that are toggled on or off in each of these games is determined by the position of each switch relative to one particular switch that is chosen to be toggled. The game considered in this paper fixes the number of switches, not necessarily one, to be toggled at each iteration. The location of the toggled switches relevant to the non-toggled switches is irrelevant in this game since toggling a switch only affects the switch itself. This paper classifies conditions when a system of switches in an inital state of all on can be transitioned to the state of all off by toggling a fixed number of switches at each iteration. When the number of switches chosen to be toggled at each iteration is odd, any system of switches can be transformed to the state where all switches are in the off state. When the number of switches chosen to be toggled at each iteration is even, only a system that consists of an even number of switches can be transitioned to the state where all switches are in the off state. The concluding results of this paper demonstrate when a system of switches can be transformed from any arbitrary initial state to a given terminal state.

The first version of the game discussed here focuses on a system of n switches all in the on state. At each iteration a fixed number, k, of switches is chosen to be toggled where the only switches whose states are changed are the k switches that were chosen. This means each switch chosen to be toggled only affects the switch itself and not its neighbors. At each iteration of this game, a switches will be turned off, and b switches will be turned on where $a + b = k$. The goal of this game is to toggle all the switches from the on state to the off state transitioning from state to state by not turning off more switches than are currently in the on state and not turning on more switches than are currently in the off state in the current state. The initial and terminal configurations of switches are exactly like the σ-game discussed by Sutner [4, p.25] except the switches in this game are not arranged in a specific geometrical configuration. In fact, the positioning of the switches is irrelevant to the game.

Lights Out is one of the most commonly known toggling switch games. It is played on a 5×5 grid of switches where choosing a single switch to be toggled will toggle not only that switch but the switches horizontal and vertical to the chosen switch. No more than 5 switches can be toggled at each iteration depending on the position of the toggled switches in the grid. The object of the game is to turn all of the lights out (i.e. all switches to the off state) through a sequence of toggles changing the grid from its original state to a state where all 25 lights are switched off. Various strategies for this game have been discussed by other mathematicians. For example, Anderson and Feil [1] give a strategy to solve this game taking a linear algebra approach. Torrence [5, p.366] introduces a variation of the classic Lights Out game using an unbounded board. In this version, choosing one switch to toggle will also toggle its 4 adjacent switches by wrapping around the board to the corresponding position when a switch along an edge is chosen. This version of Lights Out toggles exactly 5 switches at each iteration which is a specific example of a characteristic it shares with the game discussed in this paper.

The classic Toggling Light Switch game, described by Su, Francis et al. [3] starts with all switches in a row numbered 1 to 100 in the off state. In the first round, all the switches are toggled to the on state. Then every other switch is toggled in the second round leaving half of the switches in the on state and the other half in the off state. Then every third switch is toggled, every fourth switch, every fifth switch, and so on. This continues until every $100th$ switch is toggled. A parity argument of the number of divisors shows that only switches that are numbered with a perfect square remain in the on state. The classic Toggling Light Switch game is similar to the game discussed in this paper since characterizing the solution to both is based on parity.

The σ-game and σ^+-game described by Barua and Ramakrishnan [2] have a system of switches arranged in a network represented by a graph. Selecting a switch in the σ-game will cause all of the switches in the network that are a neighbor (i.e. adjacent vertex) to change without changing the selected switch, but in the case of the σ^+-game, the selected switch will also change. In this version, an arbitrary number of switches can be toggled at each iteration depending on the configuration of the network (i.e.graph). Sutner [4] examines the graph being arranged in the form of a grid. This means that all switches (i.e.vertices) have exactly 4 neighbors except for those on the boundary. In this configuration, the number of switches that can be toggled at each iteration is at least 3 but no more than 5. The game discussed here extends the restrictions on the number of switches that can be toggled at each iteration to a single fixed number, however, this restriction is imposed by the rules of the game not by the geometric arrangement of the switches as in the σ-game.

In Section 2, basic terminology describing states, transition rules, and toggling is formalized as well as giving fundamental results concerning toggling systems of switches. Section 3 provides a range of examples to demonstrate the various situations when a specific combination of toggling rules is applied in order to achieve the terminal state. The results in Section 4 establish the conditions on the number of switches in the system and the number of switches being toggled at each iteration in order to construct a sequence of transition rules that when applied transitions from all switches in the on state to all switches in the off state. A generalization of the results of Section 4 is presented in Section 5 to determine the conditions necessary to transition from an initial given state to a specified terminal state. Lastly, in Section 6, a summary of all results is given as well as suggestions for future research concerning related problems.

2 Preliminary Definitions and Results

This section establishes a formal mathematical construct to characterize a system of switches in which a fixed number can be turned on or off at each iteration. The representation of a system of switches with a particular ratio of on switches to off switches and the rules for how to transition to the same system of switches where the ratio of on switches to off switches has changed are described using familiar mathematical notation. Additionally, some basic results for transitioning from a state where all switches are in the on state to all switches in the off state are given by Theorem 2.4.

Definition 2.1 *For a positive integer n and an integer p with $0 \le p \le n$, a system of n switches can be in a state $(p, n-p)$ where exactly p switches are in the on state and $n-p$ switches are in the off state.*

The goal of this game is to go from the state $(n, 0)$ to $(0, n)$ where all n switches have been toggled from the on state to the off state. In order to change the state of the system, a transition rule must be applied to that state. However, it is not always possible to apply each transition rule to every state. A transition rule cannot be applied to a state that requires turning off more switches than there are switches already in the on state. The same is true for turning on switches. This idea is characterized by the following definition.

Definition 2.2 *For a positive integer k, a k-toggle transition rule is a function depending on a pair of integers $a, b \ge 0$ and $a + b = k$ where the function $\tau_{a,b}$ is defined by the rule $(x, y) \xrightarrow{\tau_{a,b}} (x - a + b, y + a - b)$. If $x \ge a$ and $y \ge b$ then we say that the transition rule $\tau_{a,b}$ is applicable to the state (x, y) and results in the state $(x - a + b, y + a - b)$.*

If a transition rule can be applied to a state (x, y), then it can always be applied to a state $(x + c, y + d)$ where $c, d \ge 0$ because this always results in two larger numbers in each position. If a transition rule $\tau_{a,b}$ can be applied to a state (x, y) resulting in a state (x_1, y_1), then $\tau_{a,b}$ applied to the state $(x + c, y + d)$ results in the state $(x_1 + c, y_1 + d)$.

Definition 2.3 *A system of n switches can be k-toggled if there exists a finite sequence of k-toggle transition rules $\tau_{a_1,b_1}, \tau_{a_2,b_2}, \tau_{a_3,b_3}, \ldots, \tau_{a_q,b_q}$ such that*

$$(n, 0) \xrightarrow{\tau_{a_1,b_1}} (x_1, y_1) \xrightarrow{\tau_{a_2,b_2}} (x_2, y_2) \xrightarrow{\tau_{a_3,b_3}} \ldots \xrightarrow{\tau_{a_q,b_q}} (0, n)$$

where at each iteration the k-toggle transition rule is applicable to the current state.

The following theorem establishes some fundamental results for k-toggling systems of n switches. The result is that if n switches can be k-toggled, then adding ks switches, s any positive integer, to the system gives a new system that can also be k-toggled. Additionally, it shows that a system of switches that is a multiple of k also k-toggles by turning off exactly k switches each time.

Theorem 2.4 *If a system of n switches can be k-toggled, then the following are true:*

i A system of $n + k$ switches can be k-toggled.

ii For a positive integer s, the system of sk switches can be k-toggled.

iii For a nonnegative integer s, the system $n + sk$ switches can be k-toggled.

Proof. To establish (i), assume n switches can be k-toggled. Then there exists an applicable finite sequence of k-toggle transition rules $\tau_{a_1,b_1}, \tau_{a_2,b_2}, \tau_{a_3,b_3}, \ldots, \tau_{a_q,b_q}$ with

$$(n,0) \xrightarrow{\tau_{a_1,b_1}} (x_1,y_1) \xrightarrow{\tau_{a_2,b_2}} (x_2,y_2) \xrightarrow{\tau_{a_3,b_3}} \ldots \xrightarrow{\tau_{a_q,b_q}} (0,n)$$

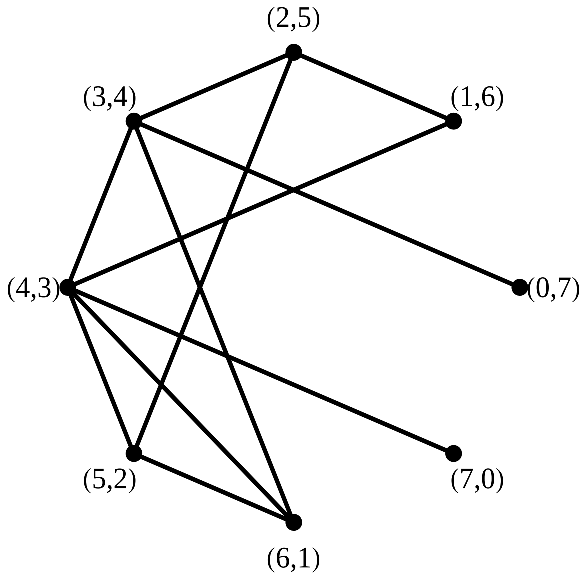
The transition rule $\tau_{k,0}$ is applicable to the state $(n+k,0)$ and results in (n,k). Now the finite sequence of transition rules $\tau_{a_1,b_1}, \tau_{a_2,b_2}, \tau_{a_3,b_3}, \ldots, \tau_{a_q,b_q}$ is applicable to (n,k). Since the composition of two applicable sequences of transition rules is also applicable, consider the sequence formed by applying $\tau_{k,0}$ first.

$$(n+k,0) \xrightarrow{\tau_{k,0}} (n,k) \xrightarrow{\tau_{a_1,b_1}} (x_1,y_1+k) \xrightarrow{\tau_{a_2,b_2}} (x_2,y_2+k) \xrightarrow{\tau_{a_3,b_3}} \ldots \xrightarrow{\tau_{a_q,b_q}} (0,n+k)$$

To prove(ii), we apply induction. In the case where $s=1$, we need to show that a system of k switches can be k-toggled. The transition rule $\tau_{k,0}$ is applicable to $(k,0)$ and results in $(0,k)$. Assume this is true for s so that a system of sk switches can be k-toggled. We need to prove that a system of $(s+1)k$ switches can be k-toggled. Rewriting $(s+1)k$ as $sk+k$ switches, (i) can be applied to conclude this system of switches can be k-toggled. A similar type of inductive argument establishes (iii). $\square$

3 Specific k-toggling Cases

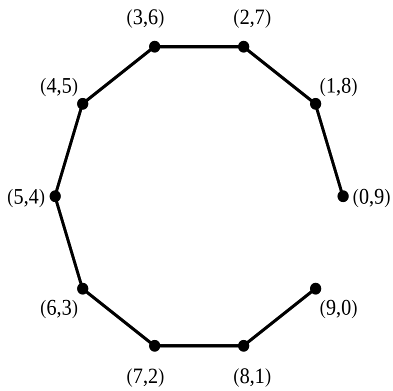
In this section, examples of systems of switches are given for specific values of n when $k \leq 4$. The examples that are chosen demonstrate the conditions when a system of switches can or cannot be k-toggled. Additionally, this section gives examples of sequences of transition rules to move from all switches in the on state to all switches in the off state. All of the $n+1$ states of a system of n switches and applicable transition rules can be represented as a graph. The vertices of the graph are each of the $n+1$ states, and the applicable transition rules form the edges by connecting each state to its result after applying the transition rule. If $(n,0)$ and $(0,n)$ are in the same connected component of the graph, then the system can be toggled. The graph below is an example of a system of 7 switches where the transition rules for a 3-toggle are being applied.



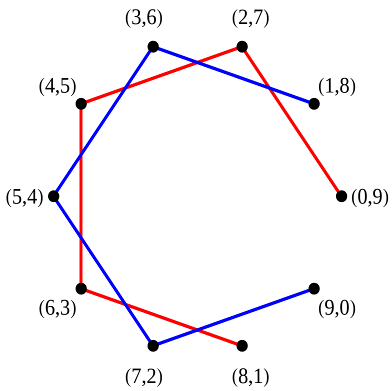
Since this graph only has one component, there is a sequence of 3-toggle transition rules that can be applied to go from one state to any other state. This demonstrates that a system of 7 switches can be 3-toggled.

Consider the case of n switches where a 1-toggle transition rule is applied. A 1-toggle transition rule either turns one switch on or one switch off at each step. So any system of n switches can be 1-toggled by turning each switch in the on state to the off state.

The case when $k = 2$ gives the first example of a system of n switches that could 2-toggle or not depending on the value of n. A system of n switches will only 2-toggle if n is even. The only applicable 2-toggle transition rules are $\tau_{2,0}$, $\tau_{1,1}$, and $\tau_{0,2}$. Each of these rules either increases or decreases the number of switches in the on state by 2 or leaves the number of switches in the on state unchanged. For this reason, only when the initial number of switches in the on state, which is n, is even will the system 2-toggle.



Graph of 9 switches with a 1-toggle transition rule



Graph of 9 switches with a 2-toggle transition rule

Look at the case of n switches where a 3-toggle transition rule is applied. Begin by looking at two specific instances where $n = 4$ and $n = 5$. In each instance, there is a finite sequence of transition rules which 3-toggle the system of switches as shown below.

$$\begin{array}{l} (4,0) \xrightarrow{\tau_{3,0}} (1,3) \xrightarrow{\tau_{1,2}} (2,2) \xrightarrow{\tau_{1,2}} (3,1) \xrightarrow{\tau_{3,0}} (0,4) \\ (5,0) \xrightarrow{\tau_{3,0}} (2,3) \xrightarrow{\tau_{1,2}} (3,2) \xrightarrow{\tau_{3,0}} (0,5) \end{array}$$

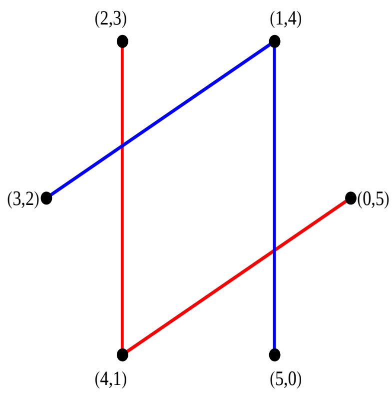
Consider any system of n switches that are being 3-toggled where $n \geq 3$. In the situation where $n \equiv 0 (\bmod 3)$, which implies that $n = 3s$ where $s \geq 1$, invoke Theorem 2.4.ii to establish that this system of n switches can be 3-toggled. Similarly in the situations where $n \equiv 1 (\bmod 3)$ or $n \equiv 2 (\bmod 3)$, which implies $n = 4 + 3s$ or $n = 5 + 3s$ where $s \geq 0$, invoke Theorem 2.4.iii with each of the rules above to establish that these systems of n switches can be 3-toggled. This shows any system of n switches ($n \geq 3$) can be 3-toggled.

In the next example, the case when $k = 4$ is investigated. Begin by looking at the instance where $n = 6$. A finite sequence of transition rules is shown below to demonstrate how 6 switches can be 4-toggled.

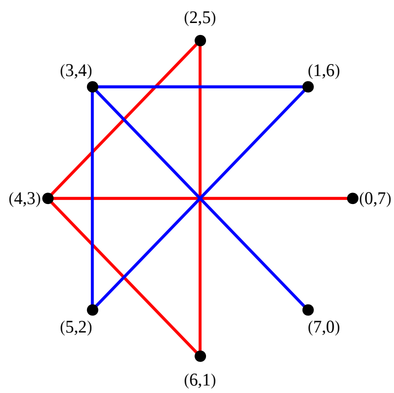
$$(6,0) \xrightarrow{\tau_{4,0}} (2,4) \xrightarrow{\tau_{1,3}} (4,2) \xrightarrow{\tau_{4,0}} (0,6)$$

Similarly, as in the case of the 3-toggle, invoke Theorem 2.4 to establish that if $n \equiv 0 (\bmod 4)$ or $n \equiv 2 (\bmod 4)$, then a system of n switches can be 4-toggled.

The graphs below show a system of 5 switches and a system of 7 switches with a 4-toggle applied to both. Since $(0,5)$ and $(5,0)$ as well as $(0,7)$ and $(7,0)$ are not in the same component of their respective graphs, it demonstrates that a system of 5 switches and a system of 7 switches cannot be 4-toggled.



Graph of 5 switches with a 4-toggle transition rule



Graph of 7 switches with a 4-toggle transition rule

Since drawing the graphs of every possible situation for $n \equiv 1 (\bmod 4)$ or $n \equiv 3 (\bmod 4)$ is impractical, another argument must be used. There are 5 possible transition rules for a 4-toggle; however, disregard the transition rule $\tau_{2,2}$ since this transition rule does not change the state. Consider the remaining 4 transition rules for a 4-toggle shown below.

$$(x,y) \xrightarrow{\tau_{4,0}} (x-4, y+4)$$
$$(x,y) \xrightarrow{\tau_{3,1}} (x-2, y+2)$$
$$(x,y) \xrightarrow{\tau_{1,3}} (x+2, y-2)$$
$$(x,y) \xrightarrow{\tau_{0,4}} (x+4, y-4)$$

Since each of the transition rules only changes the number of switches in the on state, denoted x, by either ± 2 or ± 4, if the number of lights on in the initial state is congruent to $1, 3 (\bmod 4)$, the resulting state will again be congruent to $1, 3 (\bmod 4)$. Therefore it is impossible to 4-toggle a system of n switches when $n \equiv 1, 3 (\bmod 4)$. For any $n \geq 4$, a system of n switches can be 4-toggled if n is even and cannot be 4-toggled when n is odd.

4 General k-toggling Cases

The theorems proven in this section classify all the possible situations for when a system of n switches can be k-toggled. The classification of systems of switches is determined by the relationship of the parity between n and k. The results of this section classify this situation by first examining the parity of k, the number of switches being toggled, then comparing that to the parity of n, the number of switches in the system. All of the results in this section are established by using a constructive technique of proof to show how to transition from all switches in the on state to all switches in the off state. In each situation, the sequence of transition rules to get from the state $(n, 0)$ (i.e. all switches on) to the state $(0, n)$ (i.e. all switches off) is explicitly described.

A notational convention used in this section to describe a transition rule $\tau_{a,b}$ being applied r times successively is given by $\tau_{a,b}^r$.

$$\tau_{a,b}^r = \underbrace{\tau_{a,b} \circ \tau_{a,b} \circ \cdots \circ \tau_{a,b}}_{r \text{ times}}$$

Theorem 4.1 considers the situation when k is odd and n is at least twice k. The technique used turns off k switches which then gives the ability to decrease the number of switches in the on state by one at each iteration where one more switch is toggled off than is toggled on. This is repeated until exactly k switches remain in the on state, then those k switches which still remain on can be turned off.

Theorem 4.1 *If k is odd and $n \geq 2k$, then a system of n switches can be k-toggled.*

Proof. Assume k is odd and $k = 2m + 1$ for some positive integer m, which implies $k > m$. Also since $n \geq 2k$, then $n = 2k + r$ for some $r \geq 0$. Beginning with the initial state $(n, 0)$, the only transition rule that can be applied is $\tau_{k,0}$. In the resulting state $(k + r, k)$, the transition rule $\tau_{m+1,m}$ can be applied r times since $k \geq m$ until the state $(k, k + r)$ is reached. Each time this transition rule is applied, it decreases the number of switches in the on state by 1.

Lastly to the state $(k, k+r)$, apply $\tau_{k,0}$ to result in $(0, 2k+r) = (0, n)$. This sequence of transition rules is demonstrated below to show how a system of n switches under the given conditions on n and k can be k-toggled.

$$\begin{array}{rcl} (n,0) & \xrightarrow{\tau_{k,0}} & (n-k,k) = (k+r,k) \\ (k+r,k) & \xrightarrow{\tau_{m+1,m}^{r}} & (k,k+r) \\ (k,k+r) & \xrightarrow{\tau_{k,0}} & (0,2k+r) = (0,n) \end{array}$$

□

The next theorem, again, considers the case when k is odd but differs from the previous theorem in that the values for n have to be marginally larger than k but not too much larger than k (i.e. the values of k must fit in a particular range). The method used in Theorem 4.2 is similar to that of Theorem 4.1. It begins by turning off k switches leaving less than k switches on, but there are enough switches on in order to apply a transition rule that increases the number of switches on by one each time by toggling one more on than off at each iteration. Once exactly k switches are on, all k switches can be toggled off.

Theorem 4.2 *If k is odd with $k = 2m+1$ for some natural number m, then for a number n with $m+k+1 \leq n \leq 2k-1$, a system of n switches will k-toggle.*

Proof. Assume k is odd where $k = 2m+1$ for some natural number m and n is in the given range. Starting in the state $(n, 0)$, there is only one transition rule $\tau_{k,0}$ that can be applied to result in the state $(n-k, k)$. The assumption on the value of n gives $n - k \geq m+1$ which, for all positive values of j, implies $n - k + j \geq m + 1$. From the same inequality it also follows that $k - (2k - n) \geq m + 1$. This means for all values j with $0 \leq j \leq 2k - n$, we get the inequality $k - j \geq m + 1$. Additionally, the transition rule $\tau_{m,m+1}$ is applicable at least $2k - n$ times to the state $(n - k, k)$. Applying the transition rule $\tau_{m,m+1}$ to this state $2k - n$ times results in the state $(k, n - k)$. Lastly, apply $\tau_{k,0}$ to get to the state $(0, n)$. This sequence of transition rules is demonstrated below to show how a system of n switches with n in the given range can be k-toggled.

$$\begin{array}{rcl} (n,0) & \xrightarrow{\tau_{k,0}} & (n-k,k) \\ (n-k,k) & \xrightarrow{\tau_{m,m+1}^{2k-n}} & (k,n-k) \\ (k,n-k) & \xrightarrow{\tau_{k,0}} & (0,n) \end{array}$$

□

Once again, considering the situation when k is odd, Theorem 4.3 addresses the instances when n is odd but not substantially larger than k. The technique used here optimizes the number steps required to toggle off all of the switches. In fact, only 3 steps are needed. Again, begin by turning off k switches. A specific transition rule that is applicable to the resulting state is constructed to reverse the number of switches in the on state and the number of switches in the off state. After reversing the number of switches in the on state with the number of switches in the off state, it leaves exactly k switches in the on state which can then be toggled off in a single step.

Theorem 4.3 *If k is odd, then for an odd number n with $k < n < 2k$ a system of n switches will k-toggle after applying 3 non-distinct transition rules.*

Proof. The only transition rule that can be applied to the initial state $(n, 0)$ is $\tau_{k,0}$ which results in the state $(n - k, k)$. Assuming n and k are both odd, the difference $n - k$ is even so let $n - k = 2a$ for some positive integer a. Further, assuming n is in the given range, then $2a$ satisfies the inequality $0 < 2a < k$. Dividing this inequality by 2 would yield $0 < a < \frac{k}{2}$. Since $2a < k$, it is clear that $a < k$. The transition rule $\tau_{a,k-a}$ can be applied to any state (x, y) where $x \geq a$ and $y \geq k - a$. In particular, $\tau_{a,k-a}$ can be applied to the state $(n-k, k) = (2a, k)$ resulting in the state $(k, 2a) = (k, n - k)$. Now apply $\tau_{k,0}$ to result in the state $(0, n)$. This sequence of 3 non-distinct transition rules is demonstrated below to show how a system of n switches under the given conditions can be k-toggled.

$$
\begin{aligned}
(n, 0) &\xrightarrow{\tau_{k,0}} (n - k, k) = (2a, k) \\
(2a, k) &\xrightarrow{\tau_{a,k-a}} (k, 2a) = (k, n - k) \\
(k, n - k) &\xrightarrow{\tau_{k,0}} (0, n)
\end{aligned}
$$

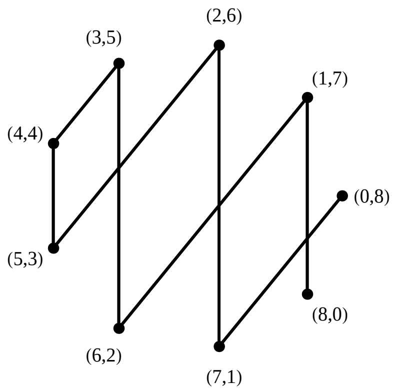
□

In the situation where k is odd and n is even but not substantially larger than k, no way was found to easily describe, in a constructive manner, the transition rules to get from all switches in the on state to all switches in the off state. However, a later theorem shows that a sequence of transition rules that manages to do this will always exist. The pattern that exists for this sequence of transition rules changes as n gets larger than k.

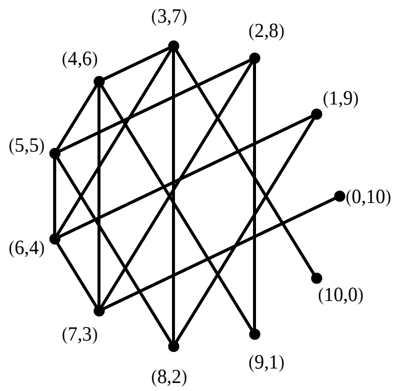
Theorem 4.4 *If k is odd and n is even with $k < n < 2k$, then a system of n switches k-toggles.*

The proof of the fact that a system of n switches with n in the given range will k-toggle for k being an odd number is given by applying Theorem 5.2 where the inital state is $(n, 0)$. The sequence of transition rules to go from the state $(n, 0)$ to the state $(0, n)$ does not follow a pattern that is easily described using the methods in this section.

Theorem 5.2 proves that a system of n switches where n is even and in the given range has a corresponding graph that is connected. In particular, the states $(n, 0)$ and $(0, n)$ are connected by some path which corresponds to a sequence of states and transition rules. The examples given below demonstrate that transitioning from the initial state $(n, 0)$ to the state $(0, n)$ follows a fundamentally different pattern of transition rules depending on how much larger n is than k.



Graph of 8 switches with a 7-toggle transition rule



Graph of 10 switches with a 7-toggle transition rule

The remaining results in this section consider the situation when k is an even number. Theorem 4.5 provides results that depend on the parity of n to determine if the system can be k-toggled. After turning off k switches, when n is even and considerably larger than k, this method uses a transition rule that decreases the number of switches in the on state by two at each iteration until exactly k switches remain in the on state. From this point, all k switches can be turned off in one step. In the event when n is odd, a system of switches can never be in the state where all of the switches are in the off state. By examining the effect transition rules have on the parity of the number of switches in the on state in the current state, the parity of the number of switches in the on state never changes from odd to even or visa versa. Thus, it is not possible to toggle all of the switches to the off state in such a situation.

Theorem 4.5 *Let k be even.*

i If n is even and $n \geq 2k$, then a system of n switches can be k-toggled.

ii If n is odd, then a system of n switches cannot be k-toggled.

Proof. To prove (i), the argument is similar to the case when k is odd. Assume k is even then $k = 2m$ for some positive integer m, which implies $k > m$. Also since $n \geq 2k$, then $n = 2k + 2r$ for some $r \geq 0$. Beginning with the initial state $(n, 0)$, the only transition rule that can be applied is $\tau_{k,0}$. In the resulting state $(k + 2r, k)$, the transition rule $\tau_{m+1,m-1}$ can be applied r times since $k > m$ until the state $(k, k + 2r)$ is reached. Each time this transition rule is applied, it decreases the number of switches in the on state by 2. Lastly to the state $(k, k + 2r)$, apply $\tau_{k,0}$ to result in $(0, 2k + 2r) = (0, n)$. This sequence of transition rules is demonstrated below to show how a system of n switches where n is even can be k-toggled.

$$\begin{aligned}
(n, 0) &\xrightarrow{\tau_{k,0}} (n - k, k) = (k + 2r, k) \\
(k + 2r, k) &\xrightarrow{\tau^{r}_{m+1,m-1}} (k, k + 2r) \\
(k, k + 2r) &\xrightarrow{\tau_{k,0}} (0, 2k + 2r) = (0, n)
\end{aligned}$$

To prove (ii), assume n is odd, then n is congruent to $1(\bmod 2)$. For an even number k, the set of k-toggle transition rules is given by $\tau_{k-w,w}$, where $0 \leq w \leq k$. Applying the transition rule $\tau_{k-w,w}$ to the state (x, y) results in the state $(x-k+2w, y+k-2w)$. Since k is even, $-k+2w$ must also be even. If x is congruent to $1(\bmod 2)$, then $x-k+2w$ is also congruent to $1(\bmod 2)$. This implies if n is congruent to $1(\bmod 2)$, then any sequence of k-toggle transition rules results in a state where the number of switches in the on state is congruent to $1(\bmod 2)$. Hence, a system of n switches where n is odd cannot k-toggle. □

Theorem 4.6 considers the situation when both k and n are even and n is not substantially larger than k. The method used in this theorem is the same as the one used in Theorem 4.3.

Theorem 4.6 *If k is even, then for an even number n with $k < n < 2k$ a system of n switches will k-toggle after applying 3 non-distinct transition rules.*

Proof. The only transition rule that can be applied to the initial state $(n, 0)$ is $\tau_{k,0}$ which results in the state $(n-k, k)$. Assuming n and k are both even, the difference $n-k$ is even so let $n-k = 2a$ for some positive integer a. Further, assuming n is in the given range, then $2a$ satisfies the inequality $0 < 2a < k$. Dividing this inequality by 2 would yield $0 < a < \frac{k}{2}$. Since $2a < k$, it is clear that $a < k$. The transition rule $\tau_{a,k-a}$ can be applied to any state (x, y) where $x \geq a$ and $y \geq k-a$. In particular, $\tau_{a,k-a}$ can be applied to the state $(n-k, k) = (2a, k)$ resulting in the state $(k, 2a) = (k, n-k)$. Now apply $\tau_{k,0}$ to result in the state $(0, n)$. This sequence of 3 non-distinct transition rules is demonstrated below to show how a system of n switches under the given conditions can be k-toggled.

$$\begin{aligned} (n,0) &\xrightarrow{\tau_{k,0}} (n-k,k) = (2a,k) \\ (2a,k) &\xrightarrow{\tau_{a,k-a}} (k,2a) = (k,n-k) \\ (k,n-k) &\xrightarrow{\tau_{k,0}} (0,n) \end{aligned}$$

□

Collectively, the results of this section establish not only when a system of n switches can be k-toggled but also give a constructive method for doing so. In the case when k is odd and n is greater than k, a system of n switches can always be k-toggled. To see this, Theorem 4.1 proves the case when $n \geq 2k$. When the number of switches is between k and $2k$, the result of Theorem 4.3 shows that all systems of an odd number of switches in this range can be k-toggled, and likewise, the result of Theorem 4.4 demonstrates that all systems of an even number of switches in this range can be k-toggled.

In the case when k is even and n is greater than k, a system of n switches can only be k-toggled when n is even. The results of Theorem 4.5 show that a system of n switches can be k-toggled only when n is an even number that is greater than $2k$. For an even number of switches between k and $2k$, the results of Theorem 4.6 show that such a system can be k-toggled.

5 An Arbitrary Initial State k-toggle Game

The theorems proven in this section classify when it is possible to transition from any initial state to any specified terminal state depending on the parity of k. This more closely models both the Lights Out game dicussed by Torrence [5] and the σ^+-game discussed by Barua and Ramakrishnan [2] where the goal in each of these games is to start from an arbitrary state and transition to a state where all of the switches are in the off state. Due to the nature of the arguments given in this section, a specific sequence of transition rules is not specified to show how to transition from the initial state to the terminal state but is described more generally using Graph Theory.

Theorem 5.1 considers the situation when k is odd and n is at least twice k. The strategy for the proof that is used begins by looking at a connected component of the graph that represents the system of switches in which the vertices of the graph have the number of switches in the on state in a particular range. The range of the number of switches in the on state depends on k and has a certain symmetry. Then a state is chosen where the number of switches in the on state is outside of that range, and one of two transition rules is applied to that state that will always result in a state that is a vertex in the connected component.

Theorem 5.1 *Let $k \geq 1$ be odd and $n \geq 2k$. The graph of the corresponding system of n switches has only one connected component containing all of the vertices (i.e. states) of the graph. In other words, given an initial state $(p, n-p)$, there exists a finite sequence of transition rules that can be applied to result in the state $(q, n-q)$.*

Proof. Since k is odd, let $k = 2m+1$ for some integer m. Starting in the state $(m, n-m)$, apply the transition rule $\tau_{m,m+1}$ to increase the number of switches in the on state by one with each application until the resulting state is $(n-m, m)$. This means all states $(p, n-p)$, where $m \leq p \leq n-m$, are in the same connected component of the graph. Call this set of states S.

Let d be an integer in the range $0 \leq d \leq m$ which implies that $k \leq d+k \leq m+k$. Use the definitions of $n, m,$ and k to get the following inequality.

$$d \leq m < k \leq d+k \leq 2k \leq n \tag{5.1}$$

It follows from (5.1) that $n-d \geq k$ which shows the transition rule $\tau_{0,k}$ can be applied to the state $(d, n-d)$ resulting in the state $(d+k, n-d-k)$. It also follows from (5.1) that $m \leq d+k$. Furthermore, the state $(d+k, n-d-k)$ is one of the states in the set S, since $d+k+m \leq 2m+k \leq n$. This inequality further shows that the transition rule $\tau_{k,0}$ can be applied to the state $(n-d, d)$ to result in the state $(n-d-k, d+k)$ which is again in one of the states of the set S. For all values of p in the range $0 \leq p \leq n$, the state $(p, n-p)$ is either in S, or there is a transition rule whose application results in a state in S. Therefore, all states are in the same connected component of the graph. □

This next theorem looks at the case when k is odd and n is no more than $2k$. The strategy used to prove Theorem 5.2 begins in the same fashion as the proof of Theorem 5.1 by defining a set of vertices all in the same connected component where the number of switches in the on state is in a specified range. For each state in this set of vertices, the difference between the number of switches in the on state and the number of switches in the off state is less than a certain number (Remark: The difference between the number of switches in the on state and the number of switches in the off state is implied to be the absolute value of the difference in the previous statement as well as in future similar statements). Starting with two states in which the number of switches in the on state and the number of switches in the off state is reversed that are not in the original set of vertices, a transition rule is applied to each state that results in one of two possibilities. Either both of the resulting states are in the set of vertices or neither of the resulting states are in the set of vertices. If both of the resulting states are in the set of vertices, then both of the original states are in the same component of the graph. If both of the resulting states are not in the set of vertices, then the difference between the number of switches in the on state and the number of switches in the off state strictly decreases. The corresponding transition rules are then iteratively applied to the resulting pair of states until a pair of states that is in the set of vertices is attained.

Theorem 5.2 *Let $k \geq 1$ be odd and $k < n < 2k$. The graph of the corresponding system of n switches has only one connected component containing all of the vertices (i.e. states) of the graph. In other words, given an initial state $(p, n-p)$, there exists a finite sequence of transition rules that can be applied to result in the state $(q, n-q)$.*

Proof. Since k is odd, let $k = 2m+1$ for some integer m. Again as was done in Theorem 5.1, start in the state $(m, n-m)$ and apply the transition rule $\tau_{m,m+1}$ to increase the number of switches in the on state by one with each application until the resulting state is $(n-m, m)$. This means all states $(p, n-p)$, where $m \leq p \leq n-m$, are in the same connected component of the graph. Call this set of states S. It is important to note that if a state $(q, n-q)$ is in the set S, then $q \geq m$ so that the state $(n-q, q)$ will also be in the set S.

Let d be an integer in the range $0 \leq d < m$ so that the states $(d, n-d)$ and $(n-d, d)$ are not in the set S. In the special case that $m = 0$ (i.e. $k = 1$), the set S consists of all states and no such d exists or needs to be considered since the graph is connected. Since $d < m < k$, the transition rule $\tau_{d,k-d}$ applied to the state $(d, n-d)$ results in the state $(k-d, n-(k-d))$. Likewise, apply the transition rule $\tau_{k-d,d}$ to the state $(n-d, d)$ to result in the state $(n-(k-d), k-d)$. If either of the resulting states are in the set S, then both of the resulting states are in the set S which implies both of the initial states $(d, n-d)$ and $(n-d, d)$ are in the connected component containing the vertices of S. If the resulting states are not in the set S, each of the resulting states again has the form $(d_1, n-d_1)$ and $(n-d_1, d_1)$ for $0 \leq d_1 < m$.

In the initial state the difference between the number of switches that are in the on state and the number that are in the off state is $n-2d$ in both of the initial states $(d, n-d)$ and $(n-d, d)$. It remains to be shown that in either situation the difference between the number of switches in the on state and the number of switches that are in the off state will strictly decrease from the initial state to the resulting state. Consider the following cases.

CASE 1: $k-d < n-(k-d)$

This implies that the difference between the number of switches in the on state and the number of switches in the off state is $n - 2(k - d)$. Since it was assumed that $d < m$, it follows that $2d < 2m < k$ or $4d < 2k$. This gives the following equivalent inequalities.

$$\begin{aligned} 4d &< 2k \\ -2k + 2d &< -2d \\ n - 2(k - d) &< n - 2d \end{aligned}$$

This shows that the difference of the number of switches in the on state and the number of switches in the off state in the resulting states $(k - d, n - (k - d))$ and $(n - (k - d), k - d)$ is smaller than in their corresponding initial states, $(d, n - d)$ and $(n - d, d)$.

CASE 2: $k - d > n - (k - d)$
This implies that the difference between the number of switches in the on state and the number of switches in the off state is $2(k - d) - n$. Since it was assumed that $k < n$, it follows that $2k < 2n$. This gives the following equivalent inequalities.

$$\begin{aligned} 2k &< 2n \\ 2k - n &< n \\ 2(k - d) - n &< n - 2d \end{aligned}$$

Again, this shows that the difference of the number of switches in the on state and the number of switches in the off state in the resulting states $(k - d, n - (k - d))$ and $(n - (k - d), k - d)$ is smaller than in their corresponding initial states, $(d, n - d)$ and $(n - d, d)$.

CASE 3: $k - d = n - (k - d)$
This implies that the difference between the number of switches in the on state and the number of switches in the off state is 0. Since it was assumed that $d < m$, it follows that $2d < 2m < k < n$. This gives the difference in the initial states $n - 2d > 0$, and again the number of switches in the on state and the number of switches in the off state in the resulting states $(k - d, n - (k - d))$ and $(n - (k - d), k - d)$ is smaller than in their corresponding initial states, $(d, n - d)$ and $(n - d, d)$.

If the resulting pair of states are not in the set S, iteratively apply the corresponding transition rules. At each iteration, either the resulting pair of states are in S, where it follows that the states $(d, n - d)$ and $(n - d, d)$ are in the same connected component as the set S, or the difference between the number of switches in the on state and the number of switches in the off state strictly decreases.

The difference in the number of switches in the on state and the number of switches in the off state is bounded below by either 0 if n is even, or 1 if n odd. If n is even then $n = 2w$, and the states $(d, n - d)$ and $(n - d, d)$ are in the same connected component as the state (w, w) which must be in S. On the other hand, if n is odd then $n = 2w + 1$ and the states $(w, w + 1)$ and $(w + 1, w)$ are in the same connected component of the graph which is demonstrated by applying $\tau_{m,m+1}$ to the state $(w, w+1)$ which results in the state $(w+1, w)$. The states $(w, w + 1)$ and $(w + 1, w)$ are the only states where the difference between the number of switches in the on state and the number of switches in the off state is 1. This implies that the states $(d, n - d)$ and $(n - d, d)$ are in the same connected component as the states $(w, w+1)$ and $(w+1, w)$, both of which must be in S. Thus all states are in the same connected component of the graph. □

The remaining two theorems examine even values for k and show that the graphs representing the systems of switches have exactly two connected components. Theorem 5.3 is concerned with values of n that are at least $2k$. Again, as in Theorem 5.1, a set of vertices is defined so that the number of switches in the on state all lie in a certain range, however, the vertices are in two separate components of the graph. As was done in Theorem 5.1, a state where the number of switches in the on state is outside the range is chosen, and one of two transition rules is applied. The resulting state is in the set of vertices and thus, in one of the two connected components of the graph.

Theorem 5.3 *Let $k \geq 2$ be even and $n \geq 2k$. The graph of the corresponding system of n switches has exactly two connected components, one component containing all of the vertices (i.e. states) of the graph that have an even number of switches in the on state, and the other component contains all of the vertices (i.e. states) where the number of switches in the on state is odd. In other words, given two states $(p, n - p)$ and $(q, n - q)$, there exists a finite sequence of transition rules starting from the state $(p, n - p)$ that can be applied to result in the state $(q, n - q)$ if and only if $p \equiv q \pmod 2$.*

Proof. Since k is even, let $k = 2m$ for some integer m. Starting in the state $(m, n - m)$, apply the transition rule $\tau_{m-1,m+1}$ to increase the number of switches on by two with each application until the resulting state is either $(n - m, m)$ or $(n - (m - 1), m - 1)$ when this transition rule can no longer be applied. This means all states $(p, n-p)$, where $m \leq p \leq n-m$ and $p \equiv m (\text{mod } 2)$, are in the same connected component of the graph. Call this set of states S_0. The set S_0 is never empty since $n \neq 2m = k$ the state $(m, n-m)$ is an element of the set S_0. Likewise, starting in the state $(m + 1, n - (m + 1))$, apply the transition rule $\tau_{m-1,m+1}$ to increase the number of switches on by two with each application until the resulting state is either $(n - m, m)$ or $(n - (m - 1), m - 1)$. This means all states $(p, n - p)$, where $m + 1 \leq p \leq n - m$ and $p \equiv m + 1 (\text{mod } 2)$, are in the same connected component of the graph. Call this set of states S_1. The set S_1 is never empty since the state $(m+1, n-(m+1))$ is distinct from the state $(n - m, m)$ since $m + 1 < n - m$ because $k + 1 < 2k \leq n$.

The sets S_0 and S_1 are disjoint since if $(p, n - p) \in S_0$ and $(q, n - q) \in S_1$ then $p \not\equiv q (\text{mod } 2)$. Each of the sets of states S_0 and S_1 are vertices in two distinct connected components of the graph. Define the set of states S as the disjoint union of the set of states S_0 and S_1 (i.e. $S = S_0 \cup S_1$). This means all states $(p, n - p)$, where $m \leq p \leq n - m$, are in S. Now let d be an integer in the range $0 \leq d \leq m$ which implies that $k \leq d + k \leq m + k$.

Use the definitions of $n, m,$ and k to get the following inequality.

$$d \leq m < k \leq d + k \leq 2k \leq n \tag{5.2}$$

It follows from (5.2) that $n - d \geq k$, which shows the transition rule $\tau_{0,k}$ can be applied to the state $(d, n - d)$ resulting in the state $(d + k, n - d - k)$. It also follows from (5.2) that $m \leq d + k$. Furthermore, the state $(d + k, n - d - k)$ is one of the states in the set S, since $d + k + m \leq 2m + k \leq n$ (i.e. $d + k \leq n - m$). This inequality further shows that the transition rule $\tau_{k,0}$ can be applied to the state $(n-d, d)$ to result in the state $(n-d-k, d+k)$ which is again in the set of states contained in S. For all values of p in the range $0 \leq p \leq n$, the state $(p, n - p)$ is either in S, or there is a transition rule whose application results in a state in S.

For an even number k, the set of k-toggle transition rules is given by $\tau_{k-q,q}$, where $0 \leq q \leq k$. Applying the transition rule $\tau_{k-q,q}$ to the state (x, y) results in the state $(x - k + 2q, y + k - 2q)$. Since k is even, $-k + 2q$ must also be even. If x is congruent to $0 (\bmod 2)$, then $x - k + 2q$ will also be congruent to $0 (\bmod 2)$. Similarly, if x is congruent to $1 (\bmod 2)$, then $x - k + 2q$ will also be congruent to $1 (\bmod 2)$. Therefore, the corresponding graph of the system of n switches consists of two connected components, one with all of the states that have an even number of switches in the on state and the other component with all of the states that have an odd number of switches in the on state. □

The final result, Theorem 5.4, considers the cases when n is in a specified range. The strategy for the proof of this theorem begins by defining the same set of vertices that was introduced in Theorem 5.3. As in Theorem 5.2, a pair of states with the number of switches in the on state and the number of switches in the off state reversed is chosen, and the corresponding transition rules are applied to each. Again, the resulting states are either both in the set of vertices, or the difference between the number of switches in the on state and the number of switches in the off state strictly decreases. If the latter is the case, then the corresponding transition rules are iteratively applied to the resulting pair of states until a pair of states in the set of vertices is attained. The pair of states do not necessarily lie in the same component as each other, but each of the states must lie in one of the two components.

Theorem 5.4 *Let $k \geq 2$ be even and $k < n < 2k$. The graph of the corresponding system of n switches has exactly two connected components, one component containing all of the vertices (i.e. states) of the graph that have an even number of switches in the on state, and the other component contains all of the vertices (i.e. states) where the number of switches in the on state is odd. In other words, given two states $(p, n - p)$ and $(q, n - q)$, there exists a finite sequence of transition rules starting from the state $(p, n - p)$ that can be applied to result in the state $(q, n - q)$ if and only if $p \equiv q \pmod 2$.*

Proof. Begin by constructing the set S as was done in the proof of Theorem 5.3 with the same corresponding S_0 and S_1. Recall the set S is the disjoint union of the sets S_0 and S_1 each of whose vertices are congruent $(\bmod 2)$. The set S consists of the states $(p, n - p)$ for $m \leq p \leq n - m$. For $0 \leq d < m$, apply the same corresponding sequence of transition rules iteratively to the states $(d, n - d)$ and $(n - d, d)$ as was done in the proof of Theorem 5.2. At each iteration, the resulting state will either be in S, or the difference between the number of switches in the on state and the number of switches in the off state will strictly decrease.

If the result at each iteration is in S, then the states are connected to either the component consisting of the states in S_0 or S_1. Each of the two initial states are not necessarily in the same connected component but are in one of the two components. If the difference between the number of switches in the on state and the number of switches in the off state in the resulting state is 0, then n is even with $n = 2w$, and both states are in the same connected component as the state (w, w). Since $2k \leq k + n \leq 2n$, it follows that $m \leq w \leq n - m$ and the state (w, w) is then in S. On the other hand, if the difference between the number of switches in the on state and the number of switches in the off state in the resulting state is 1, then n is odd with $n = 2w + 1$, and one of the initial states in is in the same connected component as the state $(w, w + 1)$ while the other initial state is in the same connected component as the state $(w + 1, w)$. It was assumed that $n > k$ from which it follows that the inequality

$$2k \leq n - 1 + k \leq n + 1 + k \leq 2n$$

is true. From this it follows that

$$m \leq w \leq w + 1 \leq n - m$$

is also true. This implies that both of the states $(w, w + 1)$ and $(w + 1, w)$ are in S. In either case, the two initial states $(m, n - m)$ and $(n - m, m)$ are each in one of the two connected components of the graph. Again, as was in Theorem 5.2, each of the states in the connected components of the graph must have the number of switches in the on state congruent (mod 2), and each transition rule results in a state where the number of switches in the on state is congruent (mod 2). Therefore, each connected component of the graph must have all the states where the number of switches in the on state is congruent (mod 2). □

Collectively, the results of this section establish when it is possible to transition from a given initial state to a specified terminal state. Theorem 5.1 and Theorem 5.2 address the cases when k is odd and n is greater than k to show it is always possible to transition among states. Furthermore, Theorem 5.3 and Theorem 5.4 consider the cases when k is even and n is greater than k to demonstrate that it is possible to transition among states if and only if the number of switches in the on state in the inital state and the terminal state have the same parity.

6 Conclusion

The goal of this paper is to classify when a system of n switches can be transitioned from a given initial state to a specified terminal state by toggling exactly k switches at each iteration. This paper only considers the cases when $n \geq k$. That is, the number of switches being toggled must be no more than the total number of switches in the system.

The overall results of Section 4 show the conditions necessary using constructive methods to transition from all switches in the on state to all switches in the off state depend on the parity of n and k. These results are that if k is odd then a system of switches can be transitioned from all switches in the on state to all switches in the off state regardless of the parity of n, and if k is even, then a system of switches can be transitioned from all switches in the on state to all switches in the off state only if n is also even.

The conditions necessary to be able to transition from an initial state to a terminal state are given as the results in Section 5. By examining the number of connected components of the corresponding graph, the parity of n and k determine if it is possible to make a specified transition. When k is odd, a system of switches can transition from any initial state to any terminal state regardless of the parity of n. However, when k is even, a system of switches can transition from an initial state to a terminal state only when the number of switches in the on state in the initial state and the number of switches in the on state in the terminal state have the same parity.

Further exploration of this topic could include the situation when a switch could be in more than the 2 states of on and off. That is, a switch could be toggled from on to blinking to off or any number of levels of gradation between on and off. Using the notation of this paper, this problem can be stated in the following way. A state can be described as an ordered t-tuple of n switches by $(x_1, x_2, \cdots, x_t)$ such that $x_1 + x_2 + \cdots + x_t = n$ where x_ν, for $1 \leq \nu \leq t$, represents the number of switches that are in the ν^{th} level of gradation. On the other hand, the toggling rules can take on different forms such as a switch needs to cycle through all of the levels of gradation in succession or go directly from one level of gradation to any other level of gradation. In other words, a switch goes from level i to level $i+1$ each time it is toggled until it arrives at level t and then cycles back to the first level. The other possibility is that a switch can go directly from the i^{th} level to the j^{th} level in a single iteration. This gives two different families of transition rules that can be applied when analyzing this situation.

References

[1] M. Anderson and T. Feil, *Turning Lights Out With Linear Algebra*, Math. Mag. **71**, (1998) 300-303.

[2] R. Barua and S. Ramakrishnan, *σ-game, σ^+-game and two-dimensional additive cellular automata*, Theoretical Computer Science **154**, (1996) 346-366.

[3] Su, Francis E., et al., "Toggling Light Switches." *Math Fun Facts*, <http://www.math.hmc.edu/funfacts>.

[4] Sutner, Klaus. *The σ-Game and Cellular Automata*, *Amer. Math. Mon.* **97** No. 1 (1990) 24-34.

[5] B. Torrence, *The Easiest Lights Out Games*, The College Mathematics Journal, **42**, No. 5 (2011) 361-372.