



A MULTIGRAPH CHARACTERIZATION OF PERMUTIPLE STRINGS

Benjamin V. Holt

Department of Mathematics, Southwestern Oregon Community College, Oregon
benjamin.holt@socc.edu

Received: 5/11/25, Accepted: 11/14/25, Published: 11/25/25

Abstract

A permutiple is a natural number whose representation in some base is an integer multiple of a number whose representation has the same collection of digits. A previous paper utilizes a finite-state-machine construction and its state graph to recognize permutiples and to generate new examples. Permutiples are associated with walks on the state graph which necessarily satisfy certain conditions. However, the above effort does not provide sufficient conditions for the existence of permutiples. In this paper, we provide such a condition which we will state using the language of multigraphs.

1. Introduction

Let $b > 1$ be an integer. A permutiple is a natural number which is an integer multiple of some permutation of its digits in base b [8]. Specific cases of digit-permutation problems include cyclic permutations of digits [4, 12], for example, $714285 = 5 \cdot 142857$, as well as digit reversals [5, 6, 7, 13, 16, 18, 19, 20], which include $87912 = 4 \cdot 21978$ and $98901 = 9 \cdot 10989$. Numbers which are multiples of cyclic permutations of their digits are known as *cyclic numbers* [4]. Numbers which are multiples of their reversals are known by several names, including *palintiples* [5, 6, 7], *reverse multiples* [13, 16, 19, 20], and *reverse divisors* [18].

Another specific case of permutiples worth mentioning is a paper by Qu and Curran [14] which considers base- b numbers which are multiples of $(b^{b-1}-1)/(b-1)^2$, whose representation is all consecutive base- b digits 1 through $b-1$. Two base-10 examples include $987654312 = 8 \cdot 123456789$ and $493827156 = 4 \cdot 123456789$.

Some works place no restrictions on the type of permutation which may arise in an arbitrary base and multiplier [8, 9, 10, 11]. In [8, 9], the author describes methods for finding new examples of permutiples from the digits of known examples. For instance, using these methods, we are able to find new examples $79128 = 4 \cdot 19782$

and $78912 = 4 \cdot 19728$ from the known example mentioned above, $87912 = 4 \cdot 21978$. Other works by the author [10, 11] utilize graph-theoretical and finite-state-machine methods to produce examples with any suitable number of digits. These methods are modifications of the work of Hoey [5] and Sloane [16], and both apply what are ultimately similar techniques to the digit-reversal problem. The former uses finite-state-machine methods, while the latter uses a graph-theoretical approach. Another example of applying finite-state-machine techniques to base-dependent integer sequences is a paper by Faber and Grantham [2] which finds pairs of integers whose sum is the reverse of their product. For example, 3 and 24 have this property since $3 + 24 = 27$ and $3 \cdot 24 = 72$.

The methods developed in [10] use a finite-state-machine construction, known as the *Hoey-Sloane machine*, and its state graph, known as the *Hoey-Sloane graph*, which describes a collection of possible base- b multiplications by a single-digit multiplier n . The states of the Hoey-Sloane machine are the possible carries which may occur when performing digit-preserving multiplication, and the input alphabet consists of ordered pairs representing directed edges from the *mother graph*, which catalogs how digits may be permuted in a single-digit multiplication. The initial state of the machine must be zero since the carry corresponding to the first digit in any single-digit multiplication is defined to be zero [8]. Similarly, the terminal digit of a multiplication requires the final (accepting) state to be zero, otherwise there would be digits beyond the terminal digit. It is shown in [10] that input strings which represent permutiples, known as *permutiple strings*, consist of ordered pairs which make up cycles on the mother graph. In this context, permutiples correspond to a sequence of state transitions (walks on the Hoey-Sloane graph) beginning and ending with the zero state, where each transition is induced by a collection of edges which is a multiset union of cycles of the mother graph.

In this paper, we continue the work described above to establish sufficient conditions for the existence of permutiple strings, yielding a multigraph characterization of permutiple strings. We also apply the results to several examples, which describe how to create novel permutiple examples of any suitable length.

2. Summary of Previous Work

What follows is a summary of several results and definitions from previous works [8, 10] that we will use in this article.

2.1. Basic Definitions and Results

The notation $(d_k, d_{k-1}, \dots, d_0)_b$ is used to represent $\sum_{j=0}^k d_j b^j$, where $0 \leq d_j < b$ for all $0 \leq j \leq k$. We may now define what it means to be a permutiple number.

Definition 1 ([8]). Let $1 < n < b$ be a natural number, and let σ be a permutation on $\{0, 1, 2, \dots, k\}$. We say that $(d_k, d_{k-1}, \dots, d_0)_b$ is an (n, b, σ) -permutiple provided

$$(d_k, d_{k-1}, \dots, d_1, d_0)_b = n(d_{\sigma(k)}, d_{\sigma(k-1)}, \dots, d_{\sigma(1)}, d_{\sigma(0)})_b.$$

When σ itself is not important to the discussion, we will refer to $(d_k, d_{k-1}, \dots, d_0)_b$ as simply an (n, b) -permutiple. The collection of all base- b permutiples having multiplier n will be referred to as (n, b) -permutiples.

The next result relates the digits and carries of a permutiple.

Theorem 1 ([8]). Let $(d_k, d_{k-1}, \dots, d_0)_b$ be an (n, b, σ) -permutiple, and let c_j be the j^{th} carry. Then,

$$bc_{j+1} - c_j = nd_{\sigma(j)} - d_j$$

for all $0 \leq j \leq k$.

The carries in any permutiple multiplication are always less than the multiplier.

Theorem 2 ([8]). Let $(d_k, d_{k-1}, \dots, d_0)_b$ be an (n, b, σ) -permutiple, and let c_j be the j^{th} carry. Then, $c_j \leq n - 1$ for all $0 \leq j \leq k$.

2.2. Permutiple Graphs and the Mother Graph

For any permutiple, we may define a directed graph which describes the most essential properties of the digit permutation.

Definition 2 ([10]). Let $p = (d_k, d_{k-1}, \dots, d_0)_b$ be an (n, b, σ) -permutiple. We define a directed graph, called the *graph of p* , denoted by G_p , to consist of the collection of base- b digits as vertices, and the collection of directed edges $E_p = \{(d_j, d_{\sigma(j)}) \mid 0 \leq j \leq k\}$. A graph G for which there is a permutiple p such that $G = G_p$ is called a *permutiple graph*.

For the remainder of this paper, we may drop the “directed” terminology with the understanding that all graphs and multigraphs considered here will have directed edges.

Table 1 gives a collection of permutiples with the same graph. Their common graph is shown in Figure 1.

$(4, 10, \sigma)$ -Permutiple	σ
$(8, 7, 9, 1, 2)_{10} = 4 \cdot (2, 1, 9, 7, 8)_{10}$	ρ
$(8, 7, 1, 9, 2)_{10} = 4 \cdot (2, 1, 7, 9, 8)_{10}$	$(1, 2)\rho(1, 2)$
$(7, 9, 1, 2, 8)_{10} = 4 \cdot (1, 9, 7, 8, 2)_{10}$	$\psi^{-4}\rho\psi^4$
$(7, 1, 9, 2, 8)_{10} = 4 \cdot (1, 7, 9, 8, 2)_{10}$	$\psi^{-4}(1, 2)\rho(1, 2)\psi^4$

Table 1: Permutiples with the same graph as $(8, 7, 9, 1, 2)_{10} = 4 \cdot (2, 1, 9, 7, 8)_{10}$, where $\psi = (0, 1, 2, 3, 4)$ and ρ is the reversal permutation.

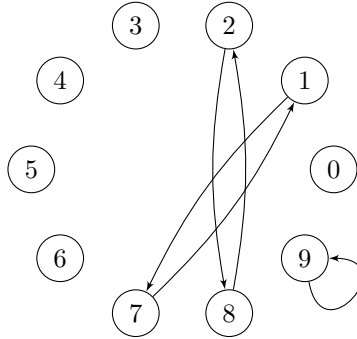


Figure 1: The directed graph which results from taking the collection of ordered pairs $\{(d_j, d_{\sigma(j)}) \mid 0 \leq j \leq 4\}$ from any example in Table 1 as edges.

Permutiple graphs provide a framework for classifying permutiples.

Definition 3 ([10]). Let p be an (n, b) -permutiple with graph G_p . We define the *class of p* to be the collection C of all (n, b) -permutiples q such that G_q is a subgraph of G_p . We also define the graph of the class to be G_p , which we will denote as G_C and will call the *graph of C* .

For an (n, b) -permutiple $(d_k, \dots, d_0)_b = n(d_{\sigma(k)}, \dots, d_{\sigma(0)})_b$ it is shown in [10] that $\lambda(d_j + (b - n)d_{\sigma(j)}) \leq n - 1$ for all $0 \leq j \leq k$, where λ gives the least non-negative residue modulo b . This condition puts a restriction on the possible edges of a permutiple graph, and we state it as a theorem.

Theorem 3 ([10]). Let $p = (d_k, d_{k-1}, \dots, d_0)_b$ be an (n, b, σ) -permutiple with graph G_p . Then, for any edge $(d_j, d_{\sigma(j)})$ of G_p , it must be that $\lambda(d_j + (b - n)d_{\sigma(j)}) \leq n - 1$ for all $0 \leq j \leq k$, where λ gives the least non-negative residue modulo b .

Theorem 3 enables us to gather all possible edges of a permutiple graph into a single graph to obtain the *mother graph*.

Definition 4 ([10]). The (n, b) -*mother graph*, denoted M , is the graph having all base- b digits as its vertices and the collection of edges (d_1, d_2) satisfying the inequality $\lambda(d_1 + (b - n)d_2) \leq n - 1$.

The next result underpins the methods presented in [10].

Theorem 4 ([10]). Let C be any (n, b) -permutiple class. Then, G_C is a union of cycles of M .

We now provide the reader with an example which brings together the concepts we have covered so far.

Example 1. The $(4, 10)$ -mother graph is displayed in Figure 2. Letting $p = (8, 7, 9, 1, 2)_{10} = 4 \cdot (2, 1, 9, 7, 8)_{10}$ from Table 1, and C be the $(4, 10)$ -permutiple class with graph $G_C = G_p$, Figure 2 features the graph of C in bold red.

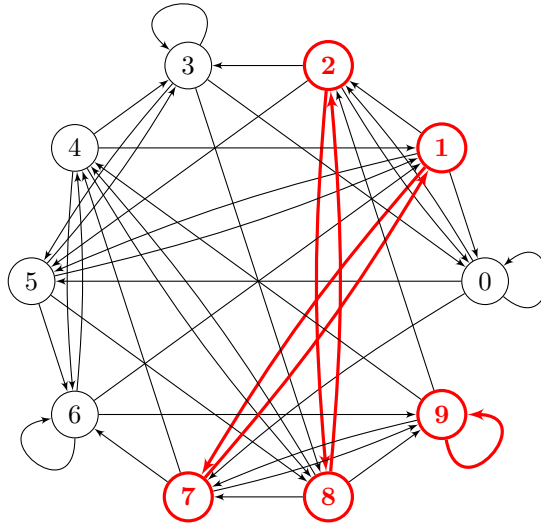


Figure 2: The $(4, 10)$ -mother graph with the graph of C featured in bold red.

2.3. Finite-State-Machine Description of the Permutiple Problem

Taking the carries of a permutiple as a collection of states, the above concepts can be placed within a finite-state-machine framework. Taking non-negative integers less than n as the collection of states, and the edges of M as the input alphabet, the equation

$$c_2 = [nd_2 - d_1 + c_1] \div b \tag{1}$$

defines a state-transition function from state c_1 to state c_2 with (d_1, d_2) serving as the input which induces the transition. This transition corresponds to a labeled edge on the state diagram as seen in Figure 3.

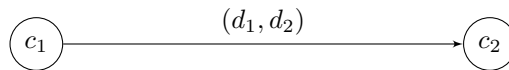


Figure 3: An edge on the state diagram.

The first carry c_0 of any single-digit multiplication is zero by definition [8]. This is to say that the initial state must be zero. Also, for an ℓ -digit (n, b) -permutiple, c_ℓ must also be zero, otherwise, the result would be an $(\ell + 1)$ -digit number. Thus,

the zero state is the only possible accepting state. The above is called the (n, b) -Hoey-Sloane machine, and its state diagram is called the (n, b) -Hoey-Sloane graph, which we denote as Γ .

Digit pairs (d_1, d_2) which solve Equation (1) for particular values of c_1 and c_2 are not unique. That is, there are generally multiple inputs that the machine will recognize for a transition to occur. Thus, a collection of inputs is assigned to each edge on Γ by the mapping $(c_1, c_2) \mapsto \{(d_1, d_2) \in E_M | c_2 = [nd_2 - d_1 + c_1] \div b\}$, where E_M is the collection of edges of M .

In the next section, we will define a labeled multigraph representation of Γ , where a unique multi-edge is assigned to each input.

The language of input strings accepted by the (n, b) -Hoey-Sloane machine is denoted as L . Thus, L may be described as finite sequences of edge-label inputs which define walks on Γ whose initial and final states are zero. Such walks are called L -walks. Members of L which produce permutiple numbers are called (n, b) -permutiple strings.

We may interpret Theorem 4 anew in this setting.

Corollary 1 ([10]). *Let $s = (d_0, \hat{d}_0)(d_1, \hat{d}_1) \cdots (d_k, \hat{d}_k)$ be a member of L . If s is a permutiple string, then the collection of ordered-pair inputs of s is a union of cycles of M .*

Corollary 1 tells us that any permutiple string must consist of a collection of mother-graph edge inputs which induce an L -walk on Γ and whose union is a collection of cycles of M . We note, however, that satisfying the above two conditions is not sufficient for a string to be a permutiple string. An example is provided by [10] of a member of L whose union is a collection of mother-graph cycles, yet is not a permutiple string. A more precise description of these ideas requires that we restate some definitions from [10].

Definition 5 ([10]). Let $\mathcal{C} = \{C_0, C_1, \dots, C_m\}$ be the cycles of M . For each element C_j of \mathcal{C} , define a subgraph Γ_j of Γ , where each edge of Γ_j is assigned the edge-label collection by the mapping $(c_1, c_2) \mapsto \{(d_1, d_2) \in C_j | c_2 = [nd_2 - d_1 + c_1] \div b\}$. Any edge (c_1, c_2) for which this collection is empty will not be included as an edge on Γ_j . With the above edges, any state for which both the indegree and outdegree are zero will not be included as a vertex. Each Γ_j will be referred to as the *image* of C_j , or simply as a *cycle image*.

In what follows, we will distinguish the usual set-theoretic union \cup from multiset unions by using the notation \uplus . For example, the multiset union of the multisets $\{1, 2, 2, 3\}$ and $\{2, 3, 4\}$ is denoted as $\{1, 2, 2, 3\} \uplus \{2, 3, 4\} = \{1, 2, 2, 2, 3, 3, 4\}$. We suppose that I is a multiset whose support is a subset J of $\{0, 1, \dots, m\}$. Then, if the cycle-image union $\Gamma_J = \bigcup_{j \in J} \Gamma_j$ (edge labels included) is a strongly-connected subgraph of Γ containing the zero state, then Γ_J describes a machine which recognizes members of L whose inputs form the union of cycles $\bigcup_{j \in J} C_j$. If the multiset

cycle union $C_I = \uplus_{j \in I} C_j$ can be ordered into a string s belonging to L , then s is a permutiple string. We note that every element of C_I must be used, including repeated elements, otherwise, the multisets of left and right components will not be equal, resulting in a multiplication which does not preserve the digits.

We now provide the reader with another example by considering the (4, 10)-Hoey-Sloane graph.

Example 2. The (4, 10)-Hoey-Sloane graph is shown in Figure 4. The graph depicting the union of the images of the cycles of G_C from Example 1 is shown in bold red. The cycles of G_C are $C_0 = \{(9, 9)\}$, $C_1 = \{(2, 8), (8, 2)\}$, and $C_2 = \{(1, 7), (7, 1)\}$, and the cycle-image union may be more precisely denoted as $\Gamma_0 \cup \Gamma_1 \cup \Gamma_2$.

Using the Hoey-Sloane graph, the multiset union $C_0 \uplus C_1 \uplus C_1 \uplus C_2 \uplus C_2$ may be ordered into a member of L , forming a permutiple string. There are multiple ways of accomplishing this, one of which is $(8, 2)(8, 2)(2, 8)(9, 9)(1, 7)(1, 7)(7, 1)(2, 8)(7, 1)$, yielding a new (4, 10)-permutiple $(7, 2, 7, 1, 1, 9, 2, 8, 8)_{10} = 4 \cdot (1, 8, 1, 7, 7, 9, 8, 2, 2)_{10}$. Another possible ordering is $(8, 2)(2, 8)(1, 7)(7, 1)(2, 8)(9, 9)(1, 7)(7, 1)(8, 2)$, which gives the new example $(8, 7, 1, 9, 2, 7, 1, 2, 8)_{10} = 4 \cdot (2, 1, 7, 9, 8, 1, 7, 8, 2)_{10}$.

As noted in [11], not every multiset union can be ordered into a permutiple string. As a trivial example, any of the above cycles individually are not sufficient to form an element of L . A less trivial example is $C_1 \uplus C_1 \uplus C_2$, which is also impossible to order into a member of L . We will address this specific case in a later example.

Generally speaking, we see that a strongly-connected union of cycle images containing the zero state is a necessary condition for ordering its corresponding multiset union of mother-graph cycles into permutiple strings [10], but it is not a sufficient one. Finding sufficient conditions is the purpose of this effort.

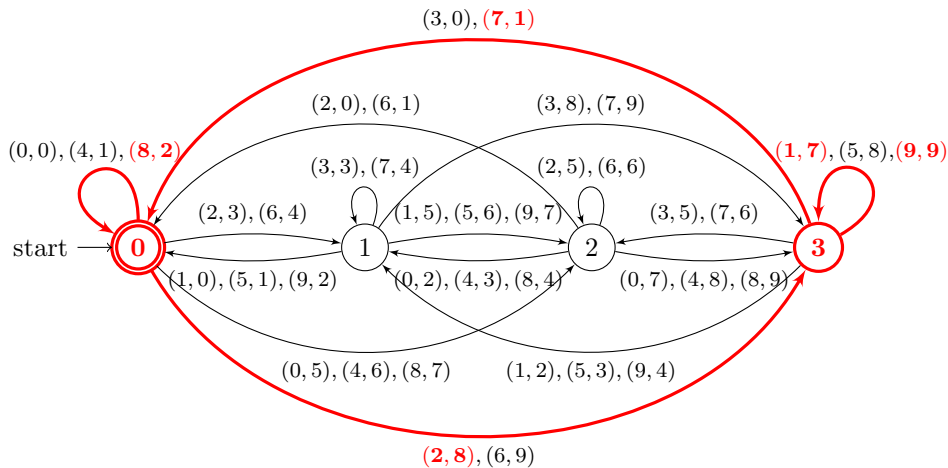


Figure 4: The (4, 10)-Hoey-Sloane graph with cycle images of G_C in bold red.

3. The Hoey-Sloane Multigraph

We begin by stating and proving a result which shows that an edge label (d_1, d_2) cannot appear on two or more distinct edges of Γ .

Theorem 5. *If c_1, c_2, \hat{c}_1 , and \hat{c}_2 are states on Γ , and (d_1, d_2) is an input associated with the transitions (c_1, c_2) and (\hat{c}_1, \hat{c}_2) , then $(c_1, c_2) = (\hat{c}_1, \hat{c}_2)$.*

Proof. By Equation (1), we have both $bc_2 - c_1 = nd_2 - d_1$ and $b\hat{c}_2 - \hat{c}_1 = nd_2 - d_1$. Reducing both equations modulo b , we have $c_1 \equiv d_1 - nd_2 \equiv \hat{c}_1 \pmod{b}$. Since c_1 and \hat{c}_1 are less than n by definition, it follows that $c_1 = \hat{c}_1$. A routine calculation then shows that $c_2 = \hat{c}_2$. \square

We now show that any edge (d_1, d_2) of M induces some transition (c_1, c_2) on Γ .

Theorem 6. *Let (d_1, d_2) be an edge of M . Then, there are integers $0 \leq c_1 \leq n - 1$ and $0 \leq c_2 \leq n - 1$ such that $bc_2 - c_1 = nd_2 - d_1$.*

Proof. From our assumption, we know that $\lambda(d_1 + (b - n)d_2) \leq n - 1$. Then, $d_1 + (b - n)d_2 \equiv d_1 - nd_2 \equiv c_1 \pmod{b}$, where $0 \leq c_1 \leq n - 1$. It follows that $nd_2 - d_1 \equiv -c_1 \pmod{b}$. We then have, for some integer c_2 , that $nd_2 - d_1 = bc_2 - c_1$. In another form, $bc_2 = nd_2 - d_1 + c_1$, from which we may say that $-b - 1 \leq bc_2 \leq n(b - 1) + n - 1 = nb - 1$. Thus, $0 \leq c_2 \leq n - 1$, and the proof is complete. \square

With Theorems 5 and 6, we may now define a directed, labeled multigraph which we will call the (n, b) -Hoey-Sloane multigraph.

Definition 6. Let N be the collection of non-negative integers less than n . We map each edge (d_1, d_2) of M to the multi-edge (c_1, c_2) in $N \times N$, which uniquely satisfies Equation (1), to form a labeled multi-edge, which we may visualize in the same fashion as Figure 3. Taking N as the collection of vertices, along with the collection of labeled multi-edges defined above, we will call this construction the (n, b) -Hoey-Sloane multigraph. To distinguish the multigraph from the usual (n, b) -Hoey-Sloane graph Γ , we will denote the (n, b) -Hoey-Sloane multigraph as Δ .

We note that Δ is simply an alternative representation of Γ , where each multi-edge has a unique element from M as a label, rather than a single edge with a collection of inputs from M . For this reason, when referencing Δ , we will retain the finite-state-machine terminology used when referencing Γ .

We now provide two simple examples of the above definition.

Example 3. The $(2, 4)$ -mother graph M is seen in Figure 5. Mapping each edge of M to its multi-edge uniquely determined by Equation (1), we obtain the $(2, 4)$ -Hoey-Sloane multigraph Δ in Figure 6.

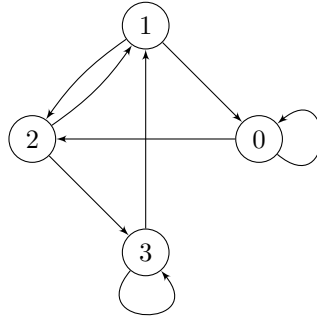


Figure 5: The $(2, 4)$ -mother graph.

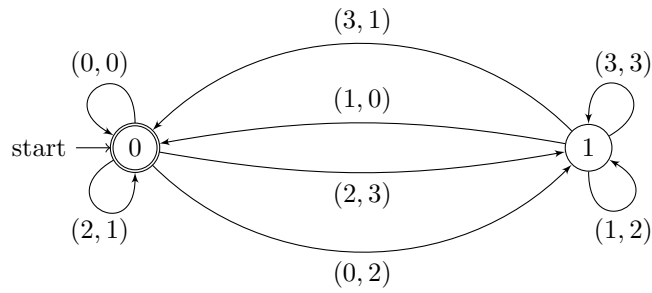


Figure 6: The $(2, 4)$ -Hoey-Sloane multigraph.

Example 4. The $(3, 4)$ -mother graph M is displayed in Figure 7. Mapping each edge of M to its multi-edge, we obtain the $(3, 4)$ -Hoey-Sloane multigraph Δ shown in Figure 8.

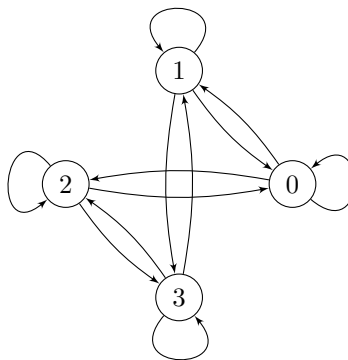


Figure 7: The $(3, 4)$ -mother graph.

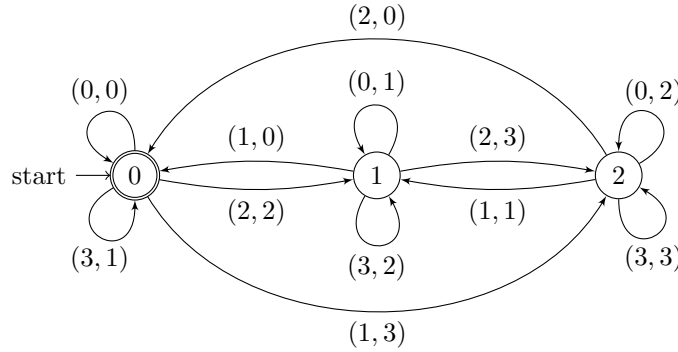


Figure 8: The (3, 4)-Hoey-Sloane multigraph.

4. A Multigraph Condition for Permutable Strings

We now define the multigraph analogue of the cycle images, which we encountered in Definition 5.

Definition 7. Let $\mathcal{C} = \{C_0, C_1, \dots, C_m\}$ be the cycles of M . For each element C_j of \mathcal{C} , define a multigraph Δ_j to consist of all edges (c_1, c_2) of the (n, b) -Hoey-Sloane multigraph Δ for which the inputs in C_j are an edge label of (c_1, c_2) . With these edges, any state for which both the indegree and outdegree are zero will not be included as a vertex of Δ_j . Each Δ_j will be referred to as the *multi-image* of C_j , or simply as a *cycle multi-image*.

We shall presently see that $C_I = \biguplus_{j \in I} C_j$ may be ordered into permutable strings precisely when the corresponding multigraph union $\Delta_I = \biguplus_{j \in I} \Delta_j$ contains the zero state, is strongly connected, and contains an Eulerian circuit in the multigraph sense [1, 3, 15]. That is, every multi-edge of Δ_I must be used once when transitioning between states.

Theorem 7. Let $\{C_0, C_1, \dots, C_m\}$ be the collection of cycles of M , and let Δ_j be the corresponding multi-image of C_j . Also, let I be a multiset whose support is a subset of $\{0, 1, \dots, m\}$. If a multigraph union of cycle multi-images $\Delta_I = \biguplus_{j \in I} \Delta_j$ contains an Eulerian circuit beginning and ending with the zero state, then the corresponding multiset union of mother-graph cycles $C_I = \biguplus_{j \in I} C_j$ may be ordered into a permutable string.

Proof. If Δ_I contains an Eulerian circuit whose initial and final state is zero, then every labeled multi-edge is used exactly once when traversing the circuit. It follows that every edge-label input of C_I is used to traverse the circuit. \square

Since a multigraph contains an Eulerian circuit if and only if it is strongly connected, and the indegree and outdegree are equal at each vertex [1, 3, 15], we may easily construct multigraphs which produce permutiple strings: form a multigraph union of cycle multi-images which 1) contains the zero state, 2) is strongly connected, and 3) the indegree is equal to the outdegree at each vertex. Any union of cycle multi-images satisfying all three of these conditions will be called *L-Eulerian*.

Corollary 2. *Let $\{C_0, C_1, \dots, C_m\}$ be the collection of cycles of M , and let Δ_j be the corresponding multi-image of C_j . Also, let I be a multiset whose support is a subset of $\{0, 1, \dots, m\}$. Then, the multiset union of mother-graph cycles $C_I = \biguplus_{j \in I} C_j$ may be ordered into a permutiple string if and only if the corresponding multigraph union of the cycle multi-images $\Delta_I = \biguplus_{j \in I} \Delta_j$ is *L-Eulerian*.*

Example 5. We continue where Example 3 leaves off. The cycles of the (2, 4)-mother graph and their corresponding multi-images are shown in Figure 6.





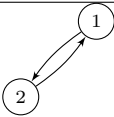

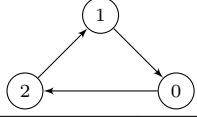
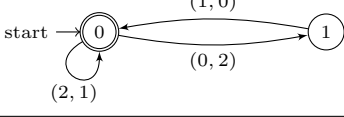
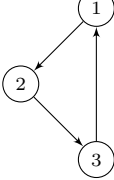
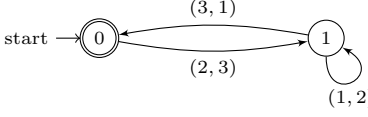
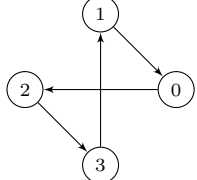
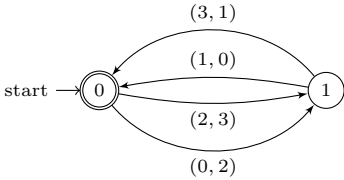
	Mother-Graph Cycle	Cycle Multi-Image	
C_0		$(0, 0)$ 	Δ_0
C_1		$(3, 3)$ 	Δ_1
C_2		$(2, 1)$ 	Δ_2
C_3		$(1, 0)$ $(0, 2)$ $(2, 1)$ 	Δ_3
C_4		$(3, 1)$ $(2, 3)$ $(1, 2)$ 	Δ_4
C_5		$(3, 1)$ $(1, 0)$ $(2, 3)$ $(0, 2)$ 	Δ_5

Table 2: Cycles of the (2, 4)-mother graph and corresponding cycle multi-images.

Since $\Delta_0, \Delta_3, \Delta_4,$ and Δ_5 are all L -Eulerian, we may form permutiple strings from these graphs individually. Moreover, since the indegrees and outdegrees are equal for all vertices for each Δ_j , we may say that any multigraph union of cycle multi-images which involves $\Delta_3, \Delta_4,$ and Δ_5 will enable us to form permutiple strings. Permutiples which may be formed from multi-images of individual cycles mentioned above are given in Table 3.

Cycle Multi-Image	Permutiple String	Permutiple Example
Δ_0	(0, 0)	$(0)_4 = 2 \cdot (0)_4$
Δ_3	(2, 1)(0, 2)(1, 0) (0, 2)(1, 0)(2, 1)	$(1, 0, 2)_4 = 2 \cdot (0, 2, 1)_4$ $(2, 1, 0)_4 = 2 \cdot (1, 0, 2)_4$
Δ_4	(2, 3)(1, 2)(3, 1)	$(3, 1, 2)_4 = 2 \cdot (1, 2, 3)_4$
Δ_5	(0, 2)(1, 0)(2, 3)(3, 1) (0, 2)(3, 1)(2, 3)(1, 0) (2, 3)(1, 0)(0, 2)(3, 1) (2, 3)(3, 1)(0, 2)(1, 0)	$(3, 2, 1, 0)_4 = 2 \cdot (1, 3, 0, 2)_4$ $(1, 2, 3, 0)_4 = 2 \cdot (0, 3, 1, 2)_4$ $(3, 0, 1, 2)_4 = 2 \cdot (1, 2, 0, 3)_4$ $(1, 0, 3, 2)_4 = 2 \cdot (0, 2, 1, 3)_4$

Table 3: Examples of (2, 4)-permutiples formed from $\Delta_0, \Delta_3, \Delta_4,$ and Δ_5 .

A less trivial instance is the multiset cycle union $C_2 \uplus C_3$, which gives a collection of inputs with which we may form permutiple strings since the corresponding multi-image union $\Delta_2 \uplus \Delta_3$ is L -Eulerian. This is verified by examining $\Delta_2 \uplus \Delta_3$ in Figure 9. We may apply Corollary 2 to deduce that we may order $C_2 \uplus C_3$ into permutiple strings using $\Delta_2 \uplus \Delta_3$.

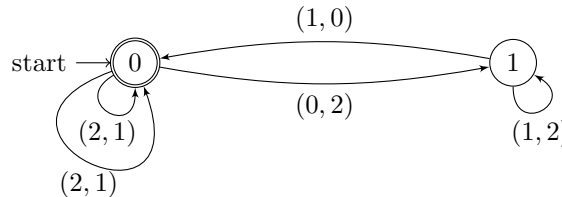


Figure 9: The multi-image union $\Delta_I = \Delta_2 \uplus \Delta_3$ corresponding to the multiset union $C_I = C_2 \uplus C_3$ of mother-graph cycles.

All numerically distinct, 5-digit examples obtained from Eulerian circuits on $\Delta_2 \uplus \Delta_3$, beginning and ending with the zero state, are included in Table 4.

Multi-Image Union	Permutiple String	Example
$\Delta_2 \uplus \Delta_3$	(2, 1)(0, 2)(1, 2)(1, 0)(2, 1) (2, 1)(2, 1)(0, 2)(1, 2)(1, 0) (0, 2)(1, 2)(1, 0)(2, 1)(2, 1)	$(2, 1, 1, 0, 2)_4 = 2 \cdot (1, 0, 2, 2, 1)_4$ $(1, 1, 0, 2, 2)_4 = 2 \cdot (0, 2, 2, 1, 1)_4$ $(2, 2, 1, 1, 0)_4 = 2 \cdot (1, 1, 0, 2, 2)_4$

Table 4: Numerically distinct, 5-digit (2, 4)-permutiples formed from the multigraph union $\Delta_2 \uplus \Delta_3$.

As mentioned earlier, as well as in [10], the multiset context allows for cycles to

appear more than once. We take the multiset union

$$C_3 \uplus C_3 = \{(0, 2), (2, 1), (1, 0), (0, 2), (2, 1), (1, 0)\}$$

as an example which gives us several six-digit permutiples. We first examine its corresponding multi-image union $\Delta_3 \uplus \Delta_3$ shown in Figure 10.

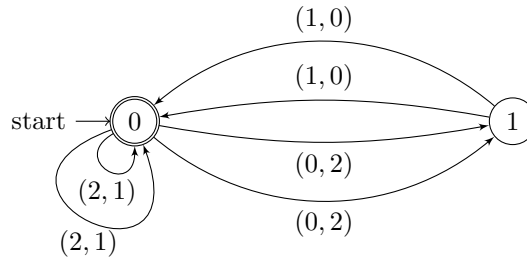


Figure 10: The multi-image union $\Delta_I = \Delta_3 \uplus \Delta_3$ corresponding to the multiset union $C_I = C_3 \uplus C_3$ of mother-graph cycles.

Again, we see that the multi-image union $\Delta_3 \uplus \Delta_3$ is L -Eulerian. The numerically distinct permutiple strings and permutiple examples formed from Eulerian circuits beginning and ending with the zero state are displayed in Table 5.

Multi-Image Union	Permutiple String	Example
$\Delta_3 \uplus \Delta_3$	$(2, 1)(0, 2)(1, 0)(2, 1)(0, 2)(1, 0)$	$(1, 0, 2, 1, 0, 2)_4 = 2 \cdot (0, 2, 1, 0, 2, 1)_4$
	$(0, 2)(1, 0)(2, 1)(0, 2)(1, 0)(2, 1)$	$(2, 1, 0, 2, 1, 0)_4 = 2 \cdot (1, 0, 2, 1, 0, 2)_4$
	$(2, 1)(0, 2)(1, 0)(0, 2)(1, 0)(2, 1)$	$(2, 1, 0, 1, 0, 2)_4 = 2 \cdot (1, 0, 2, 0, 2, 1)_4$
	$(2, 1)(2, 1)(0, 2)(1, 0)(0, 2)(1, 0)$	$(1, 0, 1, 0, 2, 2)_4 = 2 \cdot (0, 2, 0, 2, 1, 1)_4$
	$(0, 2)(1, 0)(0, 2)(1, 0)(2, 1)(2, 1)$	$(2, 2, 1, 0, 1, 0)_4 = 2 \cdot (1, 1, 0, 2, 0, 2)_4$
	$(0, 2)(1, 0)(2, 1)(2, 1)(0, 2)(1, 0)$	$(1, 0, 2, 2, 1, 0)_4 = 2 \cdot (0, 2, 1, 1, 0, 2)_4$

Table 5: Numerically distinct, 6-digit $(2, 4)$ -permutiples formed from the multigraph union $\Delta_3 \uplus \Delta_3$.

As already mentioned, since all of the cycle multi-images have equal indegree and outdegree at each vertex, any multigraph union of them will also share this property. It follows that all nontrivial $(2, 4)$ -permutiple strings can be formed from multigraph unions of cycle multi-images which contain at least one copy of Δ_3 , Δ_4 , or Δ_5 .

Example 6. We now continue with Example 4. The cycles of the $(3, 4)$ -mother graph and their corresponding cycle multi-images are seen in Table 6.


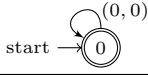

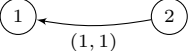

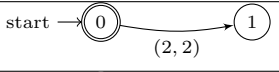

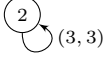
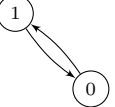
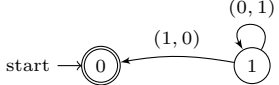

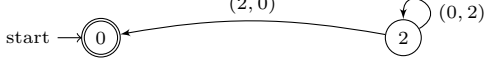

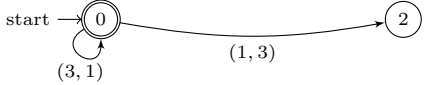
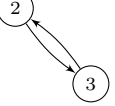
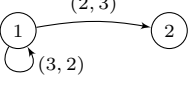
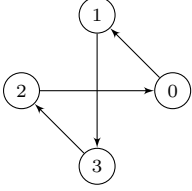
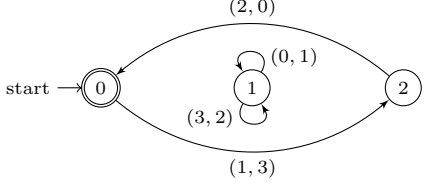
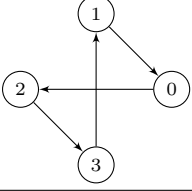
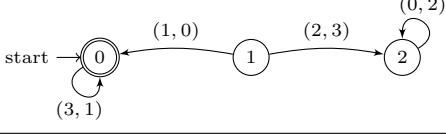
	Mother-Graph Cycle	Cycle Multi-Image	
C_0			Δ_0
C_1			Δ_1
C_2			Δ_2
C_3			Δ_3
C_4			Δ_4
C_5			Δ_5
C_6			Δ_6
C_7			Δ_7
C_8			Δ_8
C_9			Δ_9

Table 6: Cycles of the (3, 4)-mother graph and corresponding cycle multi-images.

We may now consider multiset unions of mother-graph cycles whose corresponding cycle multi-image union is L -Eulerian. In this way, Table 6 describes how to form any (3, 4)-permutiple string with a suitable number of digits. For instance, the multiset union of mother-graph cycles $C_I = C_1 \uplus C_2 \uplus C_5 \uplus C_6 \uplus C_9$ corresponds

to the multi-image union $\Delta_I = \Delta_1 \uplus \Delta_2 \uplus \Delta_5 \uplus \Delta_6 \uplus \Delta_9$, which is L -Eulerian. This multigraph is shown in Figure 11.

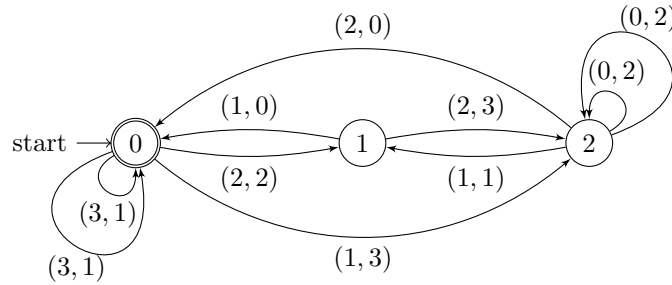


Figure 11: The multi-image union $\Delta_I = \Delta_1 \uplus \Delta_2 \uplus \Delta_5 \uplus \Delta_6 \uplus \Delta_9$ corresponding to the multiset union $C_I = C_1 \uplus C_2 \uplus C_5 \uplus C_6 \uplus C_9$ of mother-graph cycles.

We may use Figure 11 to form permutiple strings by traversing Eulerian circuits on Δ_I which begin and end with the zero state, such as

$$(1, 3)(0, 2)(1, 1)(1, 0)(3, 1)(2, 2)(2, 3)(0, 2)(2, 0)(3, 1),$$

which yields the permutiple $(3, 2, 0, 2, 2, 3, 1, 1, 0, 1)_4 = 3 \cdot (1, 0, 2, 3, 2, 1, 0, 1, 2, 3)_4$.

For our final example, we apply the above techniques to the $(4, 10)$ -permutiple class C whose graph G_C is featured in Figures 1 and 2.

Example 7. We begin by examining the multi-images of the cycles of G_C .

	Cycle of G_C	Cycle Multi-Image	
C_0			Δ_0
C_1			Δ_1
C_2			Δ_2

Table 7: Cycles of G_C and their corresponding cycle multi-images.

As shown in Example 2, the multiset union

$$C_0 \uplus C_1 \uplus C_1 \uplus C_2 \uplus C_2 = \{(9, 9), (8, 2), (8, 2), (2, 8), (2, 8), (7, 1), (7, 1), (1, 7), (1, 7)\}$$

may be ordered into permutiple strings. We verify this by examining the multi-image union $\Delta_0 \uplus \Delta_1 \uplus \Delta_1 \uplus \Delta_2 \uplus \Delta_2$ corresponding to the above multiset union of cycles, shown in Figure 12.

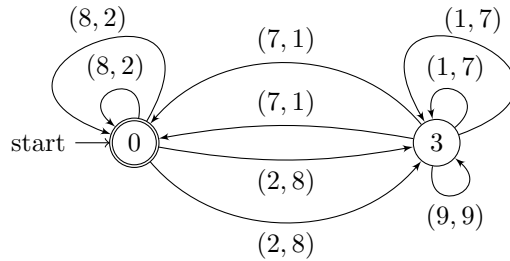


Figure 12: The multi-image union $\Delta_I = \Delta_0 \uplus \Delta_1 \uplus \Delta_1 \uplus \Delta_2 \uplus \Delta_2$ corresponding to the multiset union $C_I = C_0 \uplus C_1 \uplus C_1 \uplus C_2 \uplus C_2$ of mother-graph cycles.

Since the multi-image union is L -Eulerian, we see, by Corollary 2, that we may order the multiset union $C_0 \uplus C_1 \uplus C_1 \uplus C_2 \uplus C_2$ into permutiple strings by traversing Eulerian circuits beginning and ending with the zero state on $\Delta_0 \uplus \Delta_1 \uplus \Delta_1 \uplus \Delta_2 \uplus \Delta_2$. Two examples of these can be found in Example 2.

On the other hand, as claimed both in Example 2 and in [11], the multiset union $C_1 \uplus C_1 \uplus C_2$ cannot be ordered into permutiple strings. Again, we examine the corresponding multigraph union of multi-images $\Delta_1 \uplus \Delta_1 \uplus \Delta_2$.

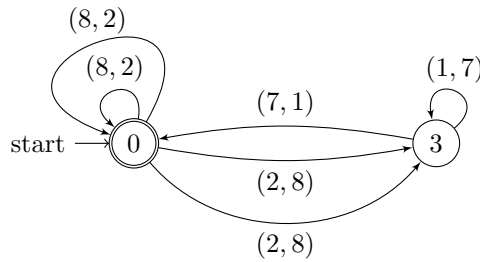


Figure 13: The multi-image union $\Delta_I = \Delta_1 \uplus \Delta_1 \uplus \Delta_2$ corresponding to the multiset union $C_I = C_1 \uplus C_1 \uplus C_2$ of mother-graph cycles.

Although $\Delta_1 \uplus \Delta_1 \uplus \Delta_2$ contains the zero state and is strongly connected, we see that the indegrees and outdegrees at both vertices are unequal. Thus, the multigraph

is not L -Eulerian, and the conditions of Corollary 2 are not met. Consequently, $C_1 \uplus C_1 \uplus C_2$ cannot be ordered into a permutiple string.

5. Summary, Conclusions, and Future Work

In [11], the question is raised regarding sufficient conditions which allow for the formation of permutiple strings. In this paper, we have provided such a condition, which leads to the following equivalence: a multiset union of mother-graph cycles can be ordered into a permutiple string if and only if the multigraph union of corresponding cycle multi-images is L -Eulerian (that is, if it contains the zero state, is strongly connected, and the indegree is equal to the outdegree at each vertex).

From the above considerations, we see that counting permutiples with a fixed base, multiplier, and length, and a fixed multiset of digits, is reduced to counting Eulerian circuits of a particular strongly-connected multi-image union containing the zero state having equal indegree and outdegree at each vertex. Counting Eulerian circuits of directed graphs may be accomplished in polynomial time using the BEST algorithm [17]. Multigraph variations also exist [3]. That said, if we lift the constraint of a fixed multiset of digits, then counting permutiples of a fixed base, multiplier, and length ℓ , becomes a problem of finding all multi-image unions which are L -Eulerian and have exactly ℓ multi-edges. We are uncertain of the difficulty of this latter problem, and it presents an area for further study.

Adding to the difficulties presented above, for larger bases and multipliers, the number of cycle multi-images increases rapidly. For example, as stated in [10], there are 986 cycles of the $(4, 10)$ -mother graph. Thus, considering $(4, 10)$ -permutiples of a fixed length ℓ , there are a very large number of possible cycle multi-image unions to consider. Filtering these possibilities down to those which are L -Eulerian appears to be a difficult task, and we leave it to future efforts.

As mentioned in [10], the methods presented here may be applied to the more specific *palintiple* problem where σ is the reversal permutation. Finding palintiple numbers means that we only need to examine the cycle multi-images of mother-graph 1- and 2-cycles. However, finding palintiple strings, which have the very particular form

$$(d_0, d_k)(d_1, d_{k-1}) \cdots (d_j, d_{k-j}) \cdots (d_{k-j}, d_j) \cdots (d_{k-1}, d_1)(d_k, d_0),$$

seems to be more difficult than one would expect; these are not obvious when examining the Hoey-Sloane graph or the individual cycle multi-images. This is yet another open question which could advance our understanding of palintiple numbers.

In contrast to the above, for the very specific case of palintiple numbers when $n+1$ divides b , Sloane [16] shows that the number of $k+1$ -digit (n, b) -palintiples, for $k \geq 3$,

is $F_{\lfloor \frac{k+1}{2} \rfloor - 1}$, where F_m is the m^{th} Fibonacci number. Restating this result in terms of the permutiple class C considered in Example 7, there are $F_{\lfloor \frac{k+1}{2} \rfloor - 1}$ palintiple strings of length $k + 1$. Other significant integer sequences which might arise when considering the general permutiple problem presents another open invitation for further investigation.

We conclude by presenting some questions relating to the “derived-permutiple” problem mentioned by [8, 9, 10]. A base- b permutiple $(d_k, \dots, d_0)_b$ is *derived* if its carries c_j , not including the trivial carry $c_0 = 0$, are the digits of a base- n permutiple $(c_k, c_{k-1}, \dots, c_1)_n$. This phenomenon is illustrated by the $(6, 12)$ -permutiple $(10, 3, 5, 1, 8, 6)_{12} = 6 \cdot (1, 8, 6, 10, 3, 5)_{12}$ whose non-trivial carries are the digits of the $(2, 6)$ -palintiple $(4, 3, 5, 1, 2)_6 = 2 \cdot (2, 1, 5, 3, 4)_6$. If we allow for leading digits which are zero, we have another pair of examples related to one another: considering the $(2, 3)$ -palintiple $(2, 1, 0, 1)_3 = 2 \cdot (1, 0, 1, 2)_3$, the $(3, 6)$ -permutiple $(2, 1, 0, 4, 3)_6 = 3 \cdot (0, 4, 2, 1, 3)_6$ has the carry vector $(2, 1, 0, 1, 0)$, and the $(6, 12)$ -permutiple $(8, 1, 6, 4, 3, 0)_{12} = 6 \cdot (1, 4, 3, 0, 8, 6)_{12}$ has the carry vector $(2, 1, 0, 4, 3, 0)$, whose non-trivial entries are the digits of the previous example. From the multi-graph perspective developed here, the derived-permutiple problem may be posed as the following: given a base- n permutiple $(c_k, c_{k-1}, \dots, c_1)_n$, find an (n, b) -Hoey-Sloane multigraph for which the circuit $(0, c_1, \dots, c_{k-1}, c_k, 0)$ may be traversed by using a collection of inputs which is a multiset union of (n, b) -mother-graph cycles. The above examples naturally lead to several questions: Given $(d_k, \dots, d_0)_b$, derived from $(c_k, c_{k-1}, \dots, c_1)_n$, under what conditions can we find a (b, \hat{b}) -permutiple derived from $(d_k, \dots, d_0)_b$? Are there cases for which this process may be continued indefinitely? Can entire permutiple classes (see Definition 3) be constructed from others? What larger patterns might exist between these classes?

Other questions regarding the general problem may be found in [8, 9].

Acknowledgement. We are grateful to the anonymous referee for their time, as well as their valued suggestions and corrections which greatly enhanced the quality of this work.

References

[1] J. Bang-Jensen and G. Z. Gutin, *Digraphs: Theory, Algorithms and Applications*, Springer, London, 2009.

[2] X. Faber and J. Grantham, On integers whose sum is the reverse of their product, *Fibonacci Quart.* **61**(1) (2023), 28-41.

[3] M. Farrell and L. Levine, Multi-Eulerian tours of directed graphs, *Electron. J. Combin.* **23**(2) (2016), P2.21.

[4] S. Guttman, On cyclic numbers, *Amer. Math. Monthly* **41**(3) (1934), 159-166.

- [5] D. J. Hoey, Untitled online article (not peer-reviewed). Available from OEIS Foundation Inc. (1992), *The On-Line Encyclopedia of Integer Sequences* at <https://oeis.org/A008919/a008919.txt>.
- [6] B. V. Holt, Some general results and open questions on palintiple numbers, *Integers* **14** (2014), #A42.
- [7] B. V. Holt, Derived palintiple families and their palinomials, *Integers* **16** (2016), #A27.
- [8] B. V. Holt, On permutiples having a fixed set of digits, *Integers* **17** (2017), #A20.
- [9] B. V. Holt, A method for finding all permutiples with a fixed set of digits from a single known example, *Integers* **24** (2024), #A88.
- [10] B. V. Holt, Finding permutiples of a known base and multiplier, *Integers* **25** (2025), #A13.
- [11] B. V. Holt, Utilizing the reflective symmetry of the mother graph in finding new permutiple classes from old, preprint, [arXiv:2504.17158v2](https://arxiv.org/abs/2504.17158v2).
- [12] D. Kalman, Fractions with cycling digit patterns, *College Math. J.* **27**(2) (1996), 109-115.
- [13] L. H. Kendrick, Young graphs: 1089 et al., *J. Integer Seq.* **18** (2015), Article 15.9.7.
- [14] B. Qu and S. J. Curran, An unexpected digit permutation from multiplying in any number base, in *Combinatorics, Graph Theory and Computing*, Springer, Cham, 2022, 13-31.
- [15] D. Silberger, Eulerian directed multigraphs, preprint, [arXiv:2408.12699](https://arxiv.org/abs/2408.12699).
- [16] N. J. A. Sloane, 2178 and all that, *Fibonacci Quart.* **52**(2) (2014), 99-120.
- [17] T. van Aardenne-Ehrenfest and N. G. de Bruijn, Circuits and trees in oriented linear graphs, *Bull. Belg. Math. Soc. Simon Stevin* **28** (1951), 203-217.
- [18] R. Webster and G. Williams, On the trail of reverse divisors: 1089 and all that follow, *Math. Spectrum* **45** (2012/2013): 96-102.
- [19] A. L. Young, k -reverse multiples, *Fibonacci Quart.* **30** (1992), 126-132.
- [20] A. L. Young, Trees for k -reverse multiples, *Fibonacci Quart.*, **30** (1992), 166-174.