A Nomenclature Scheme for Permutations of the Set of Natural Numbers

Michael Daniel V. Samson
Division of Mathematical Sciences
School of Physical and Mathematical Sciences
College of Science
Nanyang Technical University
michael.daniel.samson@gmail.com

ABSTRACT

The set of nonnegative integers $\mathbb{W}$ can be mapped bijectively to the finitary permutation group $\mathbf{S}_\omega$ on the set of natural numbers, as shown using the factoradic expansion of an integer and, broadly interpreting Knuth, any exhaustive permutation-generation algorithm. One such mapping using minimal composition of transpositions is presented, and this mapping is used to define a nomenclature scheme. This scheme is used with regard to representation and application of permutations within the structure of common computer-scientific uses, while determining other computational uses. The mapping can also be extended to determine a nomenclature scheme for the entire permutation group $\mathbf{S}_\infty$ on the set of natural numbers.

1. OVERVIEW

This paper clarifies and continues work done in an earlier paper [9] concerning the `GBS` exhaustive permutation-generation algorithm (similar to an algorithm described in [8]) and the nomenclature scheme encouraged from factoradic expansion (cited as *Cantor expansion* previously, from [5], though the latter is synonymous with general multi-radix number systems) and by Knuth's general permutation-generation algorithm using Sims tables in [4]. Definitions and a summary of properties of `GBS` nomenclature are in Sections 2 and 3; some computer-scientific applications of `GBS` nomenclature are discussed in Section 4; finally, further applications of `GBS` nomenclature are given in Section 5.

2. DEFINITIONS

To avoid potential confusion, this paper uses the term *whole number* ($n \in \mathbb{W}$) to denote a nonnegative integer, and the term *natural number* ($n \in \mathbb{N}$) to denote a positive integer, i.e. $0 \in \mathbb{W}$, $0 \notin \mathbb{N}$.

A notation for permutation composition:

Convention 2.1.
$$\prod_{i=1}^{n} \sigma_i = \sigma_1\sigma_2\cdots\sigma_n \textit{ where } \sigma_i \in \mathbf{S}, \textit{ a permutation group,}$$
which for this text is evaluated from left-to-right.

The definition given in [5] for a factoradic expansion can be modified thus:

Definition 2.2. *The* nth-order factoradic set *is* $\mathbb{F}_n = \prod_{m=2}^{n} \mathbb{Z}_m$ *and the* (infinite) factoradic set *is* $\mathbb{F}_\infty = \mathbb{F} = \prod_{m=2}^{\infty} \mathbb{Z}_m$. *Their elements are integer sequences* $\{a_i\}$ *where* $0 \le a_i \le i$.

Note 2.3. $\mathbb{F}_m$ *is embedded in* $\mathbb{F}_n$, $m < n$ *as such:*
$$\begin{aligned} \{a_1,\ldots,a_{m-1}\} \in \mathbb{F}_m &\mapsto \{a_1,\ldots,a_{m-1},0,\ldots,0\} \in \mathbb{F}_n \\ &\mapsto \{a_1,\ldots,a_{m-1},0,\ldots\} \in \mathbb{F} \end{aligned}$$

Definition 2.4. *The* factoradic expansion of $x \in \mathbb{Z}_{n!}$ *is* $\Phi(x)$, *where*
$$\Phi : \mathbb{Z}_n \to \mathbb{F}_n \textit{ and } \Phi(x) = \{a_i\} \iff x = \sum_{i=1}^{n-1} a_i \cdot i!.$$

Knuth [3] shows that Φ is a bijection, and that the extension $\Phi : \mathbb{W} \to \mathbb{F}$ is an injection; $\Phi(\mathbb{W})$ contains only sequences with a finite number of nonzero terms.

According to Knuth [4], if `A` represents an exhaustive permutation-generation algorithm (usually applied to a string of length n), the use of the Sims table specific to A induces a mapping $\mathtt{A} : \mathbb{F}_n \to \mathbf{S}_n$. Without ambiguity in notation, pre-compose Φ to this mapping to define the mapping $\mathtt{A} : \mathbb{Z}_{n!} \to \mathbf{S}_n$, where $\mathtt{A}(x) = \mathtt{A}(\Phi(x))$:

Example 2.5. $\mathtt{A}(7) = \mathtt{A}(\Phi(7)) = \mathtt{A}(\{1, 0, 1\})$

The composition $\mathtt{A}(x)$, $x \in \mathbb{W}$ is the mapping in use everywhere in this text except in Section 5.

Using the Sims table defined in [9],

Definition 2.6. *The* GBS *algorithm induces the mapping*

$$\mathtt{GBS} : \mathbb{F}_n \to \mathbf{S}_n \textit{ where } \mathtt{GBS}(\{a_i\}) = \prod_{a_i > 0}^{1 \le i < n} (a_i \quad i+1).$$

Likewise, $\mathtt{GBS}(x)$ is defined for $x \in \mathbb{Z}_{n!}$.

Example 2.7. $\mathtt{GBS}(17) = (1 \ \ 2)(2 \ \ 3)(2 \ \ 4) = (1 \ \ 3 \ \ 4 \ \ 2)$.

Example 2.8. $\mathtt{GBS}(153) = \prod_{i=2}^{6} (1 \ \ i) = (1 \ \ 2 \ \ 3 \ \ 4 \ \ 5 \ \ 6)$.

A family of mappings $\varphi : \mathbb{Z}_{n!} \to \mathbf{S}_n$ that can be extended to $\varphi : \mathbb{W} \to \mathbf{S}_\infty$ can be described as having the following property:

Definition 2.9. *The mappings* $\varphi : \mathbb{Z}_{n!} \to \mathbf{S}_n$ telescope *if, for* $x \in \mathbb{W}$, *for all* n *such that* $x < n!$, $\varphi(x)$ *remains the same.*

Note 2.10. GBS *telescopes; Lehmer codes [6], which uses factoradic expansion as an inversion table, do not telescope, but this formulation does not follow Knuth's.*

3. GBS NOMENCLATURE FOR $\mathbf{S}_\omega$

The notation for finitary permutations over the set of natural numbers (in the parlance of [1]; in that text denoted as $FS(\mathbb{N})$) given by an exhaustive permutation-generation algorithm through the use of factoradic expansion—outlined in [4] and described for the GBS algorithm in [9]—allows for a compact reference to the action of permutation over a sequence of unique elements. Such alternative nomenclature has the computational advatages of compression and exhaustiveness over more commonly-used notation, such as two-line—that is, a factoradic expansion will always correspond to a permutation, while an array of numbers within a range must be verified to each have a unique value before it can be validated as a permutation.

As proven in [9]:

Theorem 3.1. *For* $n \in \mathbb{N}$, $\mathtt{GBS} : \mathbb{Z}_{n!} \to \mathbf{S}_n$ *is bijective.*

Lemma 3.2. *Let* $x, y \in \mathbb{W}$ *such that* $\mathtt{GBS}(x)$ *and* $\mathtt{GBS}(y)$ *are disjoint. Then the composition of* $\mathtt{GBS}(x)$ *and* $\mathtt{GBS}(y)$ *is* $\mathtt{GBS}(x + y)$.

Theorem 3.3. *For* $k \in \mathbb{W}$, *the minimum number of transpositions that can compose* $\mathtt{GBS}(k)$ *is the number of nonzero terms of* $\Phi(k)$.

Theorem 3.4. *The average number of minimum transpositions composing a permutation in* $\mathbf{S}_n$ *is* $n - H_n = n - \sum_{i=1}^{n} \frac{1}{i}$.

In the computational implementation found in [9], $\{C_i\} \in \mathbb{F}_n$ is stored in a numeric array $\mathtt{a[1}\ldots n\mathtt{]}$ such that $\mathtt{a[}i+1\mathtt{]}$ stores the value of C_i and $\mathtt{a[1]}$ is set to zero. This numeric array also determines an ordered tree as such:

Definition 3.5. *Let* $k \in \mathbb{Z}_{n!}$ *and* $\Phi(k) = \{C_i\}$ *stored in numeric array* $\mathtt{a[1}\ldots n\mathtt{]}$. *The* tree determined by k, $\mathtt{T(}k\mathtt{)}$, *has* $n + 1$ *nodes numbered* 0, 1, ..., n, *and each node numbered* j *has a link pointing to its parent node,* $\mathtt{a[}j\mathtt{]}$, *with all siblings ordered left-to-right in increasing order and* 0 *as the root.*

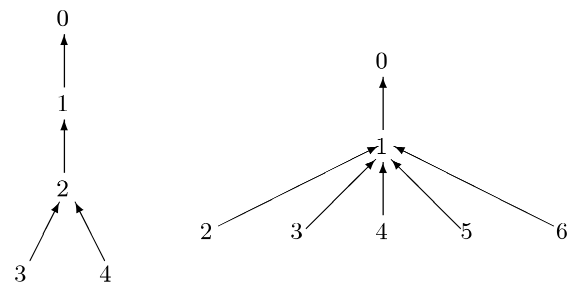

Figure 1: Trees determined by 17 and 153

Remark 3.6. *This directed graph can be treated as a forest of* n *nodes numbered* 1, 2, ..., n, *and if* $\mathtt{a[}j\mathtt{]}$ *stores the value 0, the node numbered* j *is the root of an ordered tree in that forest.*

As shown in [9]:

Theorem 3.7. *For* $k \in \mathbb{W}$, *the postorder traversals of each of the subtrees of* $\mathtt{T(}k\mathtt{)}$ *rooted in the children of* 0 *represent and indicate the order of the elements within the disjoint cycles of* $\mathtt{GBS}(k)$.

This bijection between such ordered trees and the disjoint-cycle notation of permutations indicates a direct method to generate one from the other; reversing the operation of Theorem 3.7 to determine the ordered tree (and `GBS` nomenclature) of a permutation in disjoint-cycle notation follows:

Proposition 3.8. *To determine the nth term of $\{C_i\} = \Phi(k)$ of a permutation $\sigma =$ `GBS`(k) given in disjoint-cycle notation, which determines the transposition $(C_n \quad n+1)$ that is part of the composition of σ, consider:*

If $n+1$ is the least element of its disjoint cycle, $C_n = 0$, otherwise the least element of the disjoint cycle containing $n+1$ is m, $m \leq n$. In the latter case, after arranging the disjoint cycle so that m is the rightmost element in the sequence, as the root of the tree formed in the forest described in Remark 3.6:

1. *Set a marker `X` at the rightmost element of the sequence.*
2. *Set a marker `L` at the leftmost element of the sequence.*
3. *Set a marker `R` at the smallest-numbered element of the sequence other than element marked by `X` that is at or to the right of the element marked by `L`.*
4. *If `R` marks $n+1$, then set `a[R]` to `X`—that is, C_n is equal to the rightmost element of this sequence—and this operation ends.*
5. *If `L` and `R` do not mark the same element, (recursively) apply this operation on the subsequence in between these markers, inclusive.*
6. *Mark with `L` the element to the right of the element marked by `R`; go to step 3.*

Applying Proposition 3.8 to all $n+1$ moved by σ determines $\Phi(k)$.

Theorem 3.9. *If, for all $n \in \mathbb{N}$, $\varphi : \mathbb{Z}_{n!} \to \mathbf{S}_n$ is a bijection, and φ is a telescoping mapping, the mapping $\varphi : \mathbb{W} \to \mathbf{S}_\omega$ is a bijection.*

Corollary 3.10. `GBS` $: \mathbb{W} \to \mathbf{S}_\omega$ *is a bijection.*

4. `GBS` NOMENCLATURE COMPUTATIONS

The following computer-scientific results have been computationally determined.

4.1 Group-Theoretic Applications

As an isomorphism, `GBS` $: \mathbb{W} \to \mathbf{S}_\omega$ induces a group table on $\mathbb{W}$ that mirrors that of $\mathbf{S}_\omega$ — that is, in the group table for $\mathbb{W}$, $T : \mathbb{W} \times \mathbb{W} \to \mathbb{W}$, $T(x, y) = z$ if and only if `GBS`(x)`GBS`$(y) =$ `GBS`(z). For computational purposes, it is ideal to find some analytic function $f : \mathbb{W} \times \mathbb{W} \to \mathbb{W}$ describing the group table on $\mathbb{W}$ that would perform permutation composition under the isomorphism: such a function would run at asymptotically constant time $\Theta(1)$, or as much as asymptotically linear time $\Theta(n)$ in the case a constant number of passes over the corresponding factorial expansions, while using negligible additional space, as much as a constant number of linear arrays $\Theta(n)$. The following theorem gives a restriction:

Theorem 4.1. *Let $\varphi : \mathbb{W} \to \mathbf{S}_\omega$ be a telescoping isomorphism and $f : \mathbb{W} \times \mathbb{W} \to \mathbb{W}$ be the group operation on $\mathbb{W}$ implied by φ, i.e. for $x, y \in \mathbb{W}$, $\varphi(f(x, y)) = \varphi(x)\varphi(y)$. Then f cannot be described by a rational function.*

With regard to `GBS` nomenclature, permutation composition can be staggered into a succession of transposition post-compositions, decomposing the post-composing permutation into its component transpositions.

An enumerative nomenclature scheme used to represent the permutations of $\mathbf{S}_n$ allows the group table itself to be used as a look-up table to speed up the processing of permutation computation into constant time $\Theta(1)$, at the cost of storing the table in memory as a $n! \times n!$ two-dimensional array. Both the speed of processing permutation composition and the amount of memory used by the process determine the viability of studying larger symmetric groups in the detail exhibited herein.

`GBS` nomenclature further allows the look-up table to only store results when the post-composing permutation is a transposition, at the cost of looking up as many times as the (minimal) number of transpositions in the decomposition of the post-composing permutation. The overall effect is changing a constant-time $\Theta(1)$ operation to a linear-time $\Theta(n)$ operation, but drastically reducing the size of the array from $n! \times n!$ to $n!$ times the number of transpositions in $\mathbf{S}_n$, which is $\frac{n(n-1)}{2}$, changing space usage from $\Theta(n! \cdot n!)$ to $\Theta(n^2 \cdot n!)$.

An enumerative nomenclature scheme used to represent the permutations of $\mathbf{S}_n$, such as `GBS` nomenclature, is useful in the implementation of the author's algorithm to generate the subgroups of $\mathbf{S}_n$.[1] This task was undertaken with the hope

[1] A general form of this algorithm was provided by the author as a solution to Problem D of the 2003 ACM ICPC Regional Finals (Manila), "Finding the Subgroups of a Finite Group".

that any family of subgroups of $\mathbf{S}_\omega$, or at least restricted to $\mathbf{S}_n$, can be determined analytically with the use of `GBS` nomenclature. Only one such analytic function has been determined:

Theorem 4.2. *The signature function* sgn *can be defined recursively:*

$$\operatorname{sgn}(\texttt{GBS}(k)) = \begin{cases} 1 & k = 0 \\ -\operatorname{sgn}(\texttt{GBS}(k \bmod n!)) & n! \leq k < (n+1)! \end{cases}$$

Remark 4.3. *$\sigma \in \mathbf{A}_n$, the alternating group, are determined by* $\operatorname{sgn}(\sigma) = 1$. *Mantaci and Rakotondrajao [8] have a different definition for $\sigma \in \mathbf{A}_n$.*

4.2 Permutation-Generation Algorithms as S_n-Walks

Current permutation-generating algorithms generate permutations of a string in sequence one permutation at a time—barring quantum computing—prescribed by the algorithm's mechanisms. When the set of permutations being generated is treated as a set of nodes, and consecutively-generated permutations indicate edges, the algorithm, through its sequence of generated permutations, can be treated as a walk over that set.

Knuth [4] notes that most efficient exhaustive permutation-generation algorithms start with the string and uses, at each step, only a single transposition to generate the next permutation. A naïve procedure undertaken to determine the uniqueness, or at least the smallness in number, of efficient exhaustive permutation-generation algorithms was devised: find all Hamiltonian paths over $\mathbf{S}_n$, starting from the identity, two permutations forming an edge connected by a post-composed transposition.

`GBS` nomenclature is not particularly efficient in generating the consequent permutations—operating in linear time instead of constant time—but, as a telescoping mapping, is efficient in determining whether or not a transposition composition still maintains the Hamiltonian path—the check can be performed in constant time over a (binary) flag array of size $n!$, instead of, say, an $n \times n!$ array, with quadratic-time checking, or a hash array of size $n!$ with linear-time checking.

Embarking on this method shows a fairly surprising result: for $\mathbf{S}_3$, there are 8 such distinct paths; and, for $\mathbf{S}_4$, there are more than 100,000 paths.

Question 4.4. *It can be concluded that there are fast-growing number of arbitrary Hamiltonian paths; is this also the case for restricted Hamiltonian paths?*

H Knuth [4] cites Heap's algorithm to be an example of the most efficient type of exhaustive permutation-generation algorithm. What is notable about Heap's algorithm is that it is recursive, like the algorithm generating a Gray code. The initial process was modified: to simulate a recursive algorithm, such as Heap's algorithm, we restrict the transpositions such that the ith transposition is $(a_i \;\; b_i)$, where $a_i < b_i$ and b_i is one more than the index of the first nonzero term of $\Phi(i)$. Additionally, the transpositions chosen were symmetric—the nth transposition used is the nth-from-the-last transposition used—as was the case with Heap's algorithm.

Under these conditions, the following results have been determined: for $\mathbf{S}_3$, there are two paths, with both even-numbered transpositions either $(1 \;\; 3)$, for Heap's algorithm, or $(2 \;\; 3)$; for $\mathbf{S}_4$, there are 96 such paths, shown in [9].

JT Levitin [7] cites the Johnson-Trotter algorithm (also with Steinhaus) to be an example of the most efficient type of exhaustive permutation-generation algorithm. What is notable about the Johnson-Trotter algorithm is that it is only swaps adjacent elements. The initial process was modified: only transpositions that involve adjacent elements can be used. Additionally, the transpositions chosen were symmetric, as was the case with the Johnson-Trotter algorithm, at least for the cases when the length of the permutations does not exceed four.

Under these conditions, the following results have been determined: for $\mathbf{S}_3$, there are two paths, alternating starting with $(2 \;\; 3)$, for the Johnson-Trotter algorithm, or its "mirror image"; for $\mathbf{S}_4$, there are six paths with their "mirror images", given in Table 5, with the path generated by the Johnson-Trotter algorithm denoted by an asterisk (*).

The process takes too long on contemporary household hardware for $\mathbf{S}_5$: there are approzimately 7.22×10^{12} possible paths for recursive algorithms, with 100,000 confirmed reasonably quickly; and, there are approzimately 5.09×10^{29} possible paths for adjacent-transposition algorithms, none of the confirmed paths generated reasonably quickly.

While it is possible that none of the other paths determined for $\mathbf{S}_4$ can be defined by an algorithm, it should be worthwhile to formulate efficient exhaustive permutation-generation algorithms that can generate any of the other 100 paths, and then test them for veracity.

5. GBS NOMENCLATURE FOR THE SYMMETRIC GROUP OF THE NATURALS $\mathbf{S}_\infty$

Terminology found in Dixon and Mortimer [2] are used to define some concepts.

Dixon and Mortimer introduce a caveat for $x \in \mathbf{S}_\infty$ (as $\mathbf{S}_X$ where X is infinite) in an exercise:

> . . . x may have infinite cycles and may also have infinitely many cycles. In the latter case, the product as disjoint cycles has to be interpreted suitably.

A demonstration of why caution is advised in defining disjoint cycles for the infinite case will be seen later; for now, a point of contention can be raised, whose resolution concerns the validity of Theorem 5.9:

It seems reasonable to assume that $\sigma \in \mathbf{S}_\infty$ has well-defined disjoint cycles, which partition $\mathbb{N}$ unambiguously—however, no reference has assured that this, in fact, is the case; Dixon and Mortimer comment that the product of disjoint cycles for $\mathbf{S}_X$ where X is infinite is "merely formal". Thus:

Conjecture 5.1. *The disjoint cycles of $\sigma \in \mathbf{S}_\infty$ are well-defined—it is always possible to assert whether or not $m, n \in \mathbb{N}$ are in the same disjoint cycle of σ.*

Consider the mapping $\mathtt{GBS} : \mathbb{F} \to \mathbf{S}_\infty$: evidently,

$$\mathtt{GBS}(\mathbb{F} \setminus \Phi(\mathbb{W})) \subseteq \mathbf{S}_\infty \setminus \mathbf{S}_\omega.$$

Example 5.2. *For $\alpha = \{a_i \mid a_i = 1\} \in \mathbb{F}$, $\mathtt{GBS}(\alpha) = (1\ \ 2)(1\ \ 3)(1\ \ 4)\cdots = \prod_{i=2}^{\infty}(1\ \ i) \notin \mathbf{S}_\omega$. Note that $\mathtt{GBS}(\alpha)$ does not have finite order.*

Example 5.3. *For $\beta = \{b_i \mid b_i = i(i \bmod 2)\} \in \mathbb{F}$, $\mathtt{GBS}(\beta) = (1\ \ 2)(3\ \ 4)(5\ \ 6)\cdots = \prod_{i=1}^{\infty}(2i-1\ \ 2i) \notin \mathbf{S}_\omega$. Note that $\mathtt{GBS}(\beta)$ has order 2.*

Here is the deferred demonstration illustrating Dixon and Mortimer's point of caution regarding disjoint cycles: disjoint cycles of $\sigma \in \mathbf{S}_\infty$ cannot be defined in the same way that disjoint cycles for finite permutations are defined—where a disjoint cycle containing m would contain $\sigma(m)$, $\sigma^2(m), \ldots$; the ellipses implicitly indicate that each other term is $\sigma^i(m)$ for some $i \in \mathbb{N}$. However:

Counterexample 5.4. *Let $\gamma = \{c_i \mid c_1 = 1$ else $c_i = i-1\}$. That is, $\mathtt{GBS}(\gamma) = (1\ \ 2)\prod_{i=3}^{\infty}(i-2\ \ i)$. It is evident that $\mathtt{GBS}(\gamma)$ is a cycle that moves every element of $\mathbb{N}$. However, there is no such $n \in \mathbb{N}$ for which $[\mathtt{GBS}(\gamma)]^n(1) = 2$.*

This leads to proposing a specialized definition for disjoint cycles:

Definition 5.5. *The* disjoint cycles *of $\sigma = \prod(a_{i-1}\ \ i) \in \mathtt{GBS}(\mathbb{F})$ partition $\mathbb{N}$ as such: $m, n \in \mathbb{N}$ are in the same disjoint cycle of σ if there exists $p \in \mathbb{N}$ and two (monotone decreasing) sequences $m = m_0, m_1, \ldots, m_j = p$ and $n = n_0, n_1, \ldots, n_k = p$ such that $m_{i+1} = a_{m_i-1}$ for $0 \le i < j$ and $n_{i+1} = a_{n_i-1}$ for $0 \le i < k$.*

Note 5.6. *The order of elements in a cycle is not affected by Definition 5.5: b comes right after a in a disjoint cycle of $\sigma \in \mathbf{S}_\infty$ if and only if $\sigma(a) = b$.*

It cannot be assumed that two permutations can be distinguished by their respective applications to the set they permute:

Counterexample 5.7. *Let $\delta = \{d_i \mid d_i = i-1\} = \{0, 1, 2, \ldots\}$. That is, $\mathtt{GBS}(\delta) = \prod_{i=3}^{\infty}(i-2\ \ i)$. When $\mathtt{GBS}(\gamma)$ from Counterexample 5.4 and $\mathtt{GBS}(\delta)$ are applied to $\mathbb{N}$ as a sequence, both produce sequences $3, 4, 5, 6, \ldots$ with the same n leading terms, for any $n \in \mathbb{N}$. However, $\mathtt{GBS}(\delta)$ has two disjoint cycles, whereas $\mathtt{GBS}(\gamma)$ is a cycle.*

Theorem 5.8. *The mapping $\mathtt{GBS} : \mathbb{F} \to \mathbf{S}_\infty$ is injective.*

Consider the construction outlined in the proof for Theorem 3.7: the tree determined for $k \in \mathbb{W}$ is constructed by adding nodes parallel to enumerating $\Phi(k)$. A tree can be determined from $\alpha \in \mathbb{F}$ through similar construction, although requiring a countably infinite number of added nodes. The proof allows Theorem 3.7 to hold for $\mathtt{T}(\alpha)$; Definition 5.5 is consistent with and implied by this construction.

Proposition 3.8 can determine the nth term of the $\alpha = \{a_i\} \in \mathbb{F}$, where $\sigma = \mathtt{GBS}(\alpha)$, in a finite number of steps, for any $n \in \mathbb{N}$, hinging on $\mathbb{N}$ being well-ordered. It follows that the algorithm implementing the reversal of Theorem 3.7 holds for $\sigma = \mathtt{GBS}(\alpha)$: even if α cannot be completely determined after a finite number of steps, for $\beta = \{b_i\} \in \mathbb{F}$ where $\alpha \ne \beta$—$a_m \ne b_m$ for some $m \in \mathbb{N}$—$\sigma \ne \mathtt{GBS}(\beta)$ can be verfied after a finite number of steps.

Theorem 5.9. *The mapping $\mathtt{GBS} : \mathbb{F} \to \mathbf{S}_\infty$ is a bijection.*

6. REFERENCES

[1] P. J. Cameron, *Permutation Groups*, London Mathematical Society Student Texts 45, Cambridge University Texts, 1999.

[2] J. D. Dixon and B. Mortimer, *Permutation Groups*, Graduate Texts in Mathematics 163, Springer-Verlag, New York, 1996.

[3] D. H. Knuth, *Seminumeral Algorithms*, The Art of Computer Programming Vol. 2, 3rd ed, Addison-Wesley, Massachusetts, 1997.

[4] D. H. Knuth, *Generating All Tuples and Permutations*, The Art of Computer Programming Vol. 4, Fascicle 2, Addison-Wesley, Massachusetts, 2005.

[5] E. L. Lady, *The Cantor Expansion of a Number*, Some Materials for Discrete Mathematics, http://www.math.hawaii.edu/˜lee/discrete/, 2004.

[6] D. H. Lehmer, *Teaching combinatorial tricks to a computer*, Proc. Sympos. Appl. Math. **10** (1960), 179–193.

[7] A. Levitin, *Introduction to the Design & Analysis of Algorithms*, Pearson Education, Inc. Addison-Wesley, 2004.

[8] R. Mantaci and F. Rakotondrajao, *A permutation representation that knows what "Eulerian" means*, Disc. Math. and Theo. Comp. Sci. Vol. 4 No. 2 (2001), 101–108.

[9] M. D. Samson, *Cantor Expansion and Permutation Generation Using Transpositions*, 2nd Sympos. on Math. Asp. of Comp. Sci. Preproc. (2004), 87–98.

Appendix: Selected Proofs

PROOF FOR LEMMA 3.2. Let $\Phi(x) = \{X_i\}$, $\Phi(y) = \{Y_i\}$ and $\Phi(x+y) = \{Z_i\}$, such that

$$\begin{aligned}\texttt{GBS}(x) &= \prod_{X_i>0}(X_i \;\; i+1),\\ \texttt{GBS}(y) &= \prod_{Y_j>0}(Y_j \;\; j+1),\\ \texttt{GBS}(x+y) &= \prod_{Z_k>0}(Z_k \;\; k+1).\end{aligned}$$

For every element $p \in \mathbb{N}$ fixed by $\texttt{GBS}(x)$, $X_{p-1} = 0$ and $X_i \neq p$ for all $i \geq p$; for every element $q \in \mathbb{N}$ moved by $\texttt{GBS}(y)$, $Y_{q-1} \neq 0$ or $Y_j = q$ for some $j \geq q$. Since $\texttt{GBS}(x)$ and $\texttt{GBS}(y)$ are disjoint, for every $k \in \mathbb{N}$, $X_k = 0$ or $Y_k = 0$ so $Z_k = X_k + Y_k$.

$\texttt{GBS}(x)\texttt{GBS}(y) = \prod(X_i \;\; i+1)\prod(Y_j \;\; j+1)$. Since $\texttt{GBS}(x)$ and $\texttt{GBS}(y)$ are disjoint, $(X_i \;\; i+1)$ and $(Y_j \;\; j+1)$ share no point, for $i, j \in \mathbb{N}$ where $X_i, Y_j > 0$; the transpositions commute, and since for any $k \in \mathbb{N}$, $X_k = 0$ or $Y_k = 0$, the transpositions can be rearranged such that $\texttt{GBS}(x)\texttt{GBS}(y) = \prod(Z'_k \;\; k+1)$, where $Z'_k = \max(X_k, Y_k)$. Since $Z'_k = Z_k$ for all $k \in \mathbb{N}$, $\texttt{GBS}(x+y) = \texttt{GBS}(x)\texttt{GBS}(y)$. □

PROOF FOR THEOREM 3.3. Let $\text{supp}_\Phi(k)$ be the number of nonzero terms in $\Phi(k)$; then, if $\texttt{GBS}(k)$ has m disjoint cycles, $\texttt{GBS}(k_1), \texttt{GBS}(k_2), \ldots, \texttt{GBS}(k_m)$, then, by Lemma 3.2, $k = \sum_{i=1}^m k_i$. Among $\Phi(k_i) = \{a_{ij}\}$, for any index j, at most one of the corresponding terms will be nonzero, so the nonzero terms of $\Phi(k)$ will be partitioned among the $\Phi(k_i)$—that is, $\text{supp}_\Phi(k) = \sum_{i=1}^m \text{supp}_\Phi(k_i)$.

Let $\{C_j\} = \Phi(k_i)$ and $\texttt{GBS}(k_i) = \prod_{C_j>0}(C_j \;\; j+1)$ be an n-cycle. The minimum number of transpositions needed to compose an n-cycle is $n-1$, so $\text{supp}_\Phi(k_i) \geq n-1$; since each post-composing transposition $(C_i \;\; i+1)$ introduces a new point to the permutation, $\text{supp}_\Phi(k_i) \leq n-1$. So $\text{supp}_\Phi(k_i)$ is precisely the minimum number of transpositions required to compose $\texttt{GBS}(k_i)$.

The $\texttt{GBS}(k_i)$ are disjoint, so the minimum number of transpositions required to compose $\texttt{GBS}(k)$ is the sum of the minimum number of transpositions required to compose each $\texttt{GBS}(k_i)$—otherwise, some j-cycle would be composed of less than $j-1$ transpositions—which is $\text{supp}_\Phi(k)$. □

PROOF FOR THEOREM 3.4. Let t_n be the average number of minimum transpositions composing a permutation in $\mathbf{S}_n$. $\mathbf{S}_1 = \{\texttt{GBS}(0)\}$, thus $t_1 = 0 = 1 - H_1$. $\mathbf{S}_2 = \{\texttt{GBS}(0), (1\ 2)\}$, thus $t_2 = \frac{1}{2} = 2 - H_2$.

Assume that $t_n = n - H_n$. $\mathbf{S}_{n+1}$ can be partitioned into $\mathbf{S}_n$ and its n cosets $\mathbf{S}_n(i \;\; n+1)$, for $i \in \{1, 2, \ldots, n\}$. For each of the cosets $\mathbf{S}_n(i \;\; n+1)$, the average number of minimal transpositions composing a permutation is increased by one, thus the total number of minimal transpositions composing permutations in $\mathbf{S}_{n+1}$ is:

$$\begin{aligned}(n+1)!t_{n+1} &= n!t_n + \sum_{i=1}^{n} n!(t_n+1) = n!t_n + n[n!t_n + n!]\\ &= (n+1)!t_n + (n+1)! - [(n+1)! - n \cdot n!]\\ &= (n+1)!\left[n - H_n + 1 - \left(1 - \frac{n}{n+1}\right)\right]\\ &= (n+1)!\left[(n+1) - \left(H_n + \frac{1}{n+1}\right)\right]\\ &= (n+1)![(n+1) - H_{n+1}].\end{aligned}$$

So $t_{n+1} = (n+1) - H_{n+1}$, and the induction is complete. □

PROOF FOR THEOREM 3.9. For $x \in \mathbb{W}$, $\varphi(x)$ is a finite permutation on $\mathbb{N}$, since $\varphi(x) \in \mathbf{S}_n$ whenever $x \in \mathbb{Z}_{n!}$. For any $y \in \mathbb{W}$, $x \neq y$, there is an n such that $x, y \in \mathbb{Z}_{n!}$, and since the mapping is bijective over $\mathbb{Z}_{n!}$, $\varphi(x) \neq \varphi(y)$.

Let $\sigma \in \mathbf{S}_\omega$. Then σ permutes a finite number of elements of $\mathbb{N}$. If the number of elements is permuted is zero, then σ is the identity element, and $\varphi(0) = \sigma \in \mathbf{S}_1$; otherwise, let n be the largest element not fixed by σ. Then $\sigma \in \mathbf{S}_n$, and since $\varphi : \mathbb{Z}_{n!} \to \mathbf{S}_n$ is bijective, there exists $x \in \mathbb{Z}_{n!}$ such that $\varphi(x) = \sigma$.

Thus $\varphi : \mathbb{W} \to \mathbf{S}_\omega$ is a bijection. $\square$

Proof for Theorem 4.1. If φ is a telescoping isomorphism, $\varphi(0)$ is the identity permutation. Let $f(x, y)$ be a rational function and consider $f(x, x) = F(x)$, which should also be a rational function. If $\varphi(k)$ is an involution, then $F(k) = 0$. However, the number of involutions in $\mathbf{S}_\omega$ is infinite. Thus $F(x)$ has an infinite number of zeros—a contradiction, since a rational function should have a finite number of zeros. $\square$

Proof for Theorem 5.8. Let $\alpha = \{a_i\}, \beta = \{b_i\} \in \mathbb{F}$, such that $\alpha \neq \beta$, i.e. $a_i \neq b_i$ for some $i \in \mathbb{N}$. Let $n \in \mathbb{N}$ such that $a_i = b_i$ for all $i < n$ and $a_n \neq b_n$. Consider if there is $a_i \neq 0$ for $i < n$; let

$$\sigma = \prod_{b_i > 0}^{1 \le i < n} (b_i \quad i+1).$$

Let $\alpha', \beta' \in \mathbb{F}$ where $\alpha' = \{a'_i\}$, $\beta' = \{b'_i\}$, $a'_i = b'_i = 0$ for $1 \le i < n$ and $a'_i = a_i$ and $b'_i = b_i$ for $i \ge n$. Then $\sigma\texttt{GBS}(\alpha') = \texttt{GBS}(\alpha)$ and $\sigma\texttt{GBS}(\beta') = \texttt{GBS}(\beta)$. Thus, it can be assumed that $a_i = b_i = 0$ whenever $i < n$, and the result will hold for both cases.

Let $b_n \neq 0$; there is a cycle in $\texttt{GBS}(\beta)$ that contains both b_n and $n+1$. The transpositions $(a_i \quad i+1)$ composing $\texttt{GBS}(\alpha)$ individually can move b_n or $n+1$ or neither, but not both, since $b_n < n+1 < i+1$; exactly one of the following is true:

- a_i and $n+1$ are in the same disjoint cycle of $\texttt{GBS}(\alpha)$;
- a_i and b_n are in the same disjoint cycle of $\texttt{GBS}(\alpha)$;
- a_i is in a disjoint cycle of $\texttt{GBS}(\alpha)$ that contains neither $n+1$ nor b_n.

Therefore, in $\texttt{GBS}(\alpha)$, b_n and $n+1$ are in disjoint cycles, so $\texttt{GBS}(\alpha) \neq \texttt{GBS}(\beta)$ and the mapping is injective. $\square$

Proof for Theorem 5.9. Let $\sigma \in \mathbf{S}_\infty$. The sequence $\{a_i\}$ can be determined as follows. By Conjecture 5.1, the disjoint cycles of σ are well-defined; since $\mathbb{N}$ is well-ordered, for every $i \in \mathbb{N}$, the root r_i of the disjoint cycle that contains $i+1$ can be determined: if $r_i = i+1$, $a_i = 0$; otherwise, use Proposition 3.8 on the cycle rooted in r_i to determine a_i. Then $\alpha = \{a_i\} \in \mathbb{F}$ and $\texttt{GBS}(\alpha) = \sigma$.

Therefore, $\texttt{GBS}$ is surjective; by Theorem 5.8, it is injective, so it is bijective. $\square$

Appendix: Adjacent-Transposition Hamiltonian Paths of $\mathtt{S}_4$ under GBS Nomenclature

0–1–5–2–3–4–22–21–14–15–20–23–13
–6–10–11–7–12–16–9–8–17–19–18–

0–1–5–23–20–2–3–4–22–21–14–15–10
–11–7–12–16–9–8–6–13–17–19–18–

0–1–5–23–13–6–8–17–19–18–16–9–7
–12–22–21–14–11–10–15–20–2–3–4–

0–1–5–23–13–17–19–18–16–9–8–6–10
–11–7–12–22–21–14–15–20–2–3–4–

0–4–3–2–5–1–19–18–16–12–22–21–14
–11–7–9–8–17–13–23–20–15–10–6–

0–4–3–21–22–12–16–18–19–1–5–2–20
–23–13–17–8–9–7–11–14–15–10–6–

0–4–22–21–3–2–5–1–19–18–16–12–7
–9–8–17–13–23–20–15–14–11–10–6–

0–4–22–12–16–18–19–1–5–2–3–21–14
–15–20–23–13–17–8–9–7–11–10–6–

0–18–16–9–8–17–19–1–5–23–13–6–10
–15–20–2–3–21–14–11–7–12–22–4–

0–18–16–9–8–6–13–17–19–1–5–23–20
–2–3–21–14–15–10–11–7–12–22–4–

0–18–16–9–7–12–22–4–3–21–14–11–10
–15–20–2–5–23–13–6–8–17–19–1–*

0–18–16–12–22–4–3–21–14–11–7–9–8
–6–10–15–20–2–5–23–13–17–19–1–

15 20 2 3 21 14 11 7 12 22 4
0 18 16 9 8 6 13 17 19 1 5 23 20
2 3 21 14 15 10 11 7 12 22 4
0 18 16 9 7 12 22 4 3 21 14 11 10
15 20

Portable Document Formatted using TeXnicCenter under MiKTeX on June 1, 2011.