Journal of Integer Sequences, Vol. 10 (2007),
Article 07.5.8

Enumeration of Factorizable Multi-Dimensional Permutations

Hao Zhang and Daniel Gildea
Department of Computer Science
University of Rochester
Rochester, NY 14627
USA
zhanghao@cs.rochester.edu
gildea@cs.rochester.edu

Abstract

A d-dimensional permutation is a sequence of $d+1$ permutations with the leading element being the identity permutation. It can be viewed as an alignment structure across $d+1$ sequences, or visualized as the result of permuting n hypercubes of $(d+1)$ dimensions. We study the hierarchical decomposition of d-dimensional permutations. We show that when $d \geq 2$, the ratio between non-decomposable or simple d-permutations and all d-permutations approaches 1. We also prove that the growth rate of the number of d-permutations that can be factorized into k-ary branching trees approaches $\left(\frac{k}{e}\right)^d$ as k grows.

1 Introduction

The study of multiple permutations has generated recent interest in both computational biology and computational linguistics. In biology, the original problem was the enumeration of all common intervals between genetic sequences aligned according to a specified permutation [12], which was then generalized to multiple permutations [5]. In linguistics, permutations play an essential role in the definition of Syntax Directed Translation Schemata [1], or equivalently Synchronous Context Free Grammars (SCFG) [9], which model language translation by generating strings in two languages in parallel. Melamed [7] generalized SCFG to "multitexts" of more than two languages, which also require multiple permutations. In both domains, factoring long permutations into compositions of smaller permutations has proved important. This technique is used as a representational tool for compactly representing

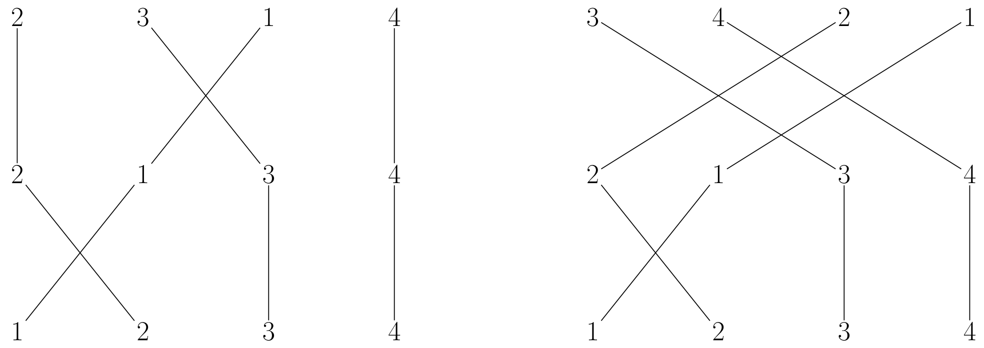


Figure 1: 2-dimensional permutations $((2,1,3,4),(2,3,1,4))$ (left) and $((2,1,3,4),(3,4,2,1))$ (right)

common intervals of gene sequences [6], and also as a method to model language translation using synchronous grammars with limited-length rules, which in turn admit efficient parsing algorithms [15].

Albert et al. [2] pioneered the combinatorial study of permutation factorization, studying the one-dimensional case. Albert et al. first reported the sequence of the number of simple permutations of n (sequence A111111 of [11]) and showed the fundamental role of simple permutations for generating all permutations. In this paper, we take a similar analytical approach to study the generalized problems of the factorization of d-dimensional permutations and the enumeration of d-dimensional simple permutations. We discover a new family of integer sequences $H_{d,n}$: the number of simple d-dimensional permutations of n. The first 8 elements of H_2 are

$$1, 4, 8, 172, 5204, 222716, 12509188, 889421564, \ldots$$

In Section 2, we formally define a d-dimensional permutation that induces the alignment view for both interval commonality in genomics and word translational equivalence in linguistics. We also define d-permutation trees as a structural analysis and description tool for d-dimensional permutations. In Section 3, we develop the generating function for k-ary d-permutation trees. In Section 4, we study the asymptotics of d-dimensional simple permutations. In Section 5, we study the growth rates of k-ary d-permutations.

2 Multi-dimensional Permutations

We begin with the definition of multi-dimensional permutations. An ordinary or one-dimensional *permutation* is a bijective function from the set $\{1, \ldots, n\}$ to the same set, conventionally denoted by π. A permutation is usually written compactly as the functional image sequence $\pi = (\pi(1), \ldots, \pi(n))$. A permutation can also be thought of as an alignment

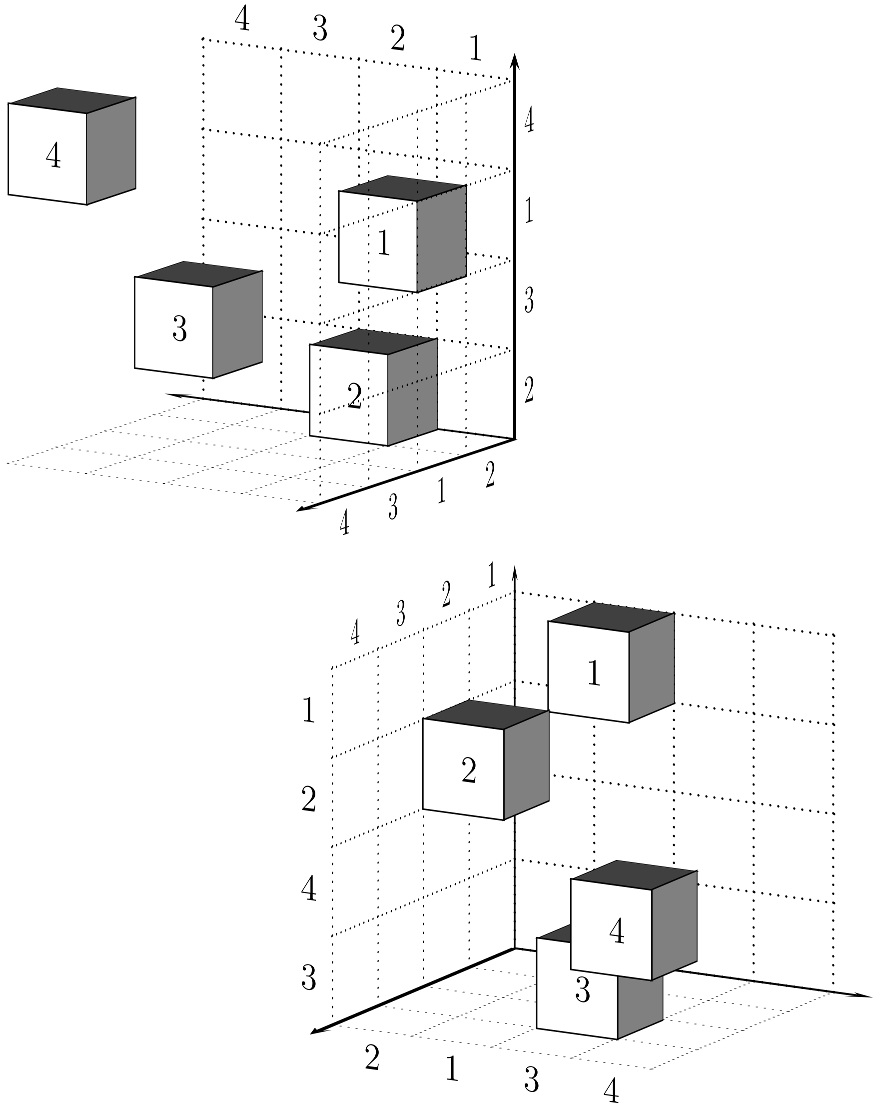


Figure 2: Three-dimensional view of the 2-dimensional permutations $((2,1,3,4),(2,3,1,4))$ (top) and $((2,1,3,4),(3,4,2,1))$ (bottom). If we project the cubes onto each axis, we can read off the permutation in each dimension.

between the two sequences $(1,\ldots,n)$ and $(\pi(1),\ldots,\pi(n))$ where identical numbers from both sides are linked together. A *d-dimensional permutation*, abbreviated as *d-permutation*, is a sequence of d one-dimensional permutations, represented as $\pi = (\pi_1,\ldots,\pi_d)$ where we use the notation $\pi_{d'}$ $(1 \le d' \le d)$ to denote the component function for the d'-th dimension. A multi-dimensional permutation can be viewed as a multilateral alignment across the identity permutation $id_n = (1,\ldots,n)$ and the remaining permutations $\pi_1,\ldots\pi_d$. Figure 1 shows two examples of 2-dimensional permutations. To visualize 2-dimensional permutations better, we can plot the numbers in the 3-dimensional cube of size n that has n^3 unit cubes of size 1. A 2-dimensional permutation is an arrangement of n numbers into n of the n^3 positions with the permutation constraints from all dimensions. Figure 2 demonstrates the 3-d view for the same two examples.

2.1 Multi-dimensional Permutation Trees

In this section, we introduce the hierarchical view of multi-dimensional permutations. We start with the hierarchical decomposition of one-dimensional permutations. We define a *permuted sequence* as a permutation of a sequence of consecutive natural numbers ranging from min to max. A standard permutation is the special case where $min = 1$ and $max = n$. We use the range notation $[min \dots max]$ to denote a permuted sequence when it is considered as a block that can be permuted as a unit with other blocks of numbers. A permutation can possibly be partitioned into permuted sequences which themselves can contain even shorter permuted sequences, thus specifying a hierarchical decomposition. Albert et al. [2] discuss the subset of permutations that do not have a decomposition which are called *simple permutations*. Zhang and Gildea [15] analyze the spectrum of decomposable permutations based on their minimum branching factors.

In a multi-dimensional permutation, a permuted sequence of numbers in one dimension may not form a permuted sequence in other dimensions. In order to produce a single factorization, we are interested in decompositions that guarantee the consistency of permuted sequences across *all* dimensions. Hence we have the notion of *d-dimensional permuted sequence*, which is a d-dimensional permutation of the consecutive range of numbers in $[min \dots max]$. Similarly, a d-dimensional permutation becomes the special case when $min = 1$ and $max = n$. We call a d-dimensional permuted sequence *k-ary ($k \geq 1$) parsable* if and only if one of the following conditions holds:

1. The permuted sequence only has one number, i.e., $min = max$.

2. It has more than one number and can be consistently segmented into k' ($2 \leq k' \leq k$) permuted sequences in all d dimensions. The permuted sequences sharing the same $[min \dots max]$ range are linked across the dimensions to form a d-dimensional permuted sequence. Each of the k' d-dimensional permuted sequences is also k-ary parsable.

The above definition is a recursive one, and implies a recursive structure that we call a *d-dimensional permutation tree*. We use the d-dimensional permutation applied on the children as the label for each parent node. The maximum number of children of any node in the tree is called the branching factor of the tree. A k-ary parsable d-dimensional permutation has a branching factor no larger than k. We are interested in minimizing the branching factor for a given multi-dimensional permutation. The two examples in Figure 1 can be decomposed into a ternary tree and a binary tree, respectively. The case $((2,1,3,4),(2,3,1,4))$ can not be binarized because $(2,1)$ is not a permuted sequence in the second dimension. However, the case $((2,1,3,4),(3,4,2,1))$ can be binarized as $(((2,1),(3,4)),((3,4),(2,1)))$. Figure 3 shows the tree structures of the two examples.

2.2 Factorization Algorithms

The particular recursive definition for permutation is natural in the context of Synchronous Context Free Grammars [9], [16]. Zhang and Gildea [15] generalize the definition in [16] from binary permutation trees to trees with an arbitrary maximum branching factor. In Section 2.1, we generalized the definition to d-permutations. We follow the same line of

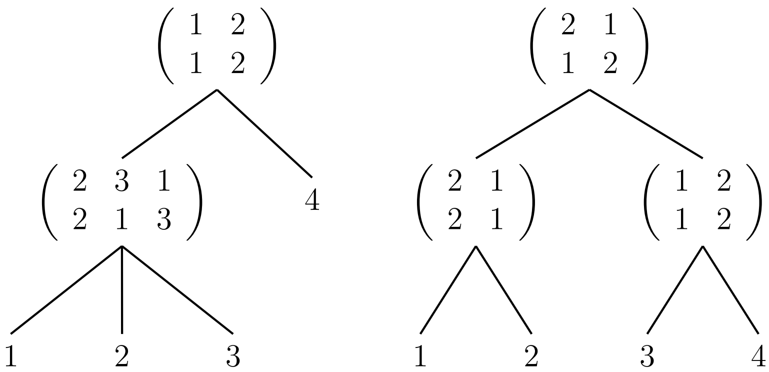


Figure 3: 2-dimensional permutation trees for $((2,1,3,4),(2,3,1,4))$ (left) and $((2,1,3,4),(3,4,2,1))$ (right). The labels for the intermediate nodes are 2-dimensional permutations indicating the reordering of the children in each dimension.

generalization for the factorization algorithm. The main idea of the algorithms is left-to-right scanning and greedy recursive reduction on permuted sequences.

The other line of development is in the context of finding all common intervals of permutations. Uno and Yagiura [12] is the breakthrough work that presents a clever algorithm optimal in time complexity: $O(n+K)$, where K is the number of common intervals between two permutations. The algorithm is then generalized to d permutations by Heber and Stoye [5], with an $O(d \cdot n + K)$ complexity. The main idea of these algorithms is right-to-left scanning and elimination of impossible right boundaries for reductions.

Landau et al. [6] and Bui-Xuan et al. [3] bridge the two lines of research by introducing PQ tree as a representational tool. PQ trees represent families of one-dimensional permutations that can be created by composing operations of scrambling subsequences according to any permutation (P nodes) and concatenating subsequences in order (Q nodes). Our definition of d-permutation tree can be thought of as a more specific version of a PQ tree, where the nodes are all labeled with a specific multidimensional permutation that is not decomposable.

Bui-Xuan et al. [3] modify the original Uno&Yagiura algorithm by introducing greedy recursive reductions, producing a tree-like representation from which all common intervals can be easily enumerated. Their algorithm solves the d-permutation factorization problem in $O(d \cdot n)$ time.

2.3 Uniqueness of Normalized Factorization

Roughly speaking, the minimum tree decomposition of a d-dimensional permutation is unique upon normalization. If a d-dimensional permutation of n has a k-ary tree, we can always produce a normalized tree that is minimized and left-heavy.

Albert et al. [2] present a theorem on the uniqueness of one-dimensional permutation decomposition. Zhang and Gildea [15] prove the uniqueness of permutation factorization from an algorithmic perspective. The main idea is that if there are two overlapping permuted sequences in a permutation, the overlapping portion itself must be a permuted sequence and the ambiguity can be succinctly captured as two possible binarization patterns: $((1,2),3)$

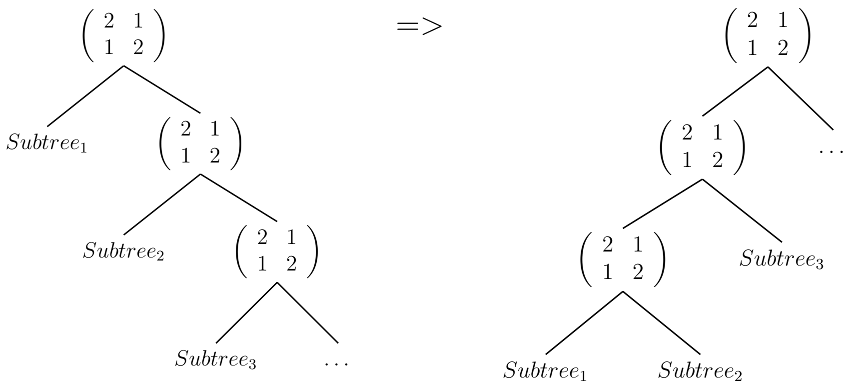


Figure 4: Spine Rotation

versus $(1,(2,3))$ or equivalently $(3,(2,1))$ versus $((3,2),1)$ for three consecutive blocks. The basic structure of ambiguity is the same for d-dimensional permutations: multiple bracketings are possible only in the case of groups of binary nodes labeled with the same internal permutation. However, there are now 2^d possible types of binary ambiguities, depending on the internal permutation, instead of only two types for one-dimensional permutations. We can perform two types of operations to make sure we prefer the left-heavy minimal trees.

The first type of normalization operation is *node expansion*. If a d-permutation being applied in the tree can be further parsed into a subtree with a smaller branching factor, we replace the original node with the tree fragment made up by these elementary permutations. We iteratively do the expansion until no node in the transformed tree can be expanded. This process ensures that we only use long d-permutations when absolutely necessary. Another way of stating this property is that no d-permutation in the tree has any strict subset of children that is consecutive in all $d+1$ dimensions.

The second type of normalization operation is *spine rotation*, and applies only to binary nodes. First, we label each internal node with its d-permutation. A spine is a sequence of nodes such that each node is the rightmost-child of the previous node. If we discover a section of consecutive identical binary d-permutation labels on a spine, we rotate the section to the left side as shown in Figure 4. We apply the rotation iteratively until no spine in the transformed tree has consecutive identical binary nodes.

The resultant parse tree yields the same d-permutation as before normalization, but with a minimized branching factor and left-most bracketing structure wherever there are multiple bracketings. The left-most preference for resolving binarization ambiguity is also used by both Shapiro and Stephens [10] and Wu [14] for the case of $k=2$ and $d=1$.

3 The number of k-ary parsable d-permutations

In this section, we develop the recurrence and the generating function for k-ary parsable d-permutations by counting the k-ary d-permutation trees. The counting technique is a generalization of that of Shapiro and Stephens [10], who studied the special case where $k=2$ and $d=1$.

In the previous section, we have stated the one-to-one correspondence between d-dimensional permutations and their normalized parses. We can count the normalized d-dimensional permutation trees with n leaves and a maximum of k children for any internal node. We focus on the cases where $d \geq 2$.

We let $S_{d,k}[n]$ ($d \geq 2$, $k \geq 2$, $n \geq 1$) be the number of k-ary parsable d-permutations of length n.

The standard trees have two defining properties according to the two-step normalization:

1. A k-ary d-permutation is used if and only if it is not $k-1$ parsable. We define $H_{d,k}$ to be the number of such simple or non-decomposable k-ary d-permutations applied in the trees:
$$H_{d,k} = (k!)^d - S_{d,k-1}[k]. \tag{1}$$
2. No two identical binary d-permutations can be applied consecutively on any spine.

From now on, the term *permutation tree* will refer to a normalized tree.

We use $A_d[k][i]$ to represent the number of ways of labeling a rightmost spine having i internal nodes in a binarized version of any k-ary permutation tree. To help derive the recurrence for A_d, we can divide spines into cases where the last child is one of the 2^d binary d-permutations ($A_{d,b}$ for $b = 1, \ldots, 2^d$) or one of the non-binary d-permutations ($A_{d,*}$)

$$\begin{aligned}
A_{d,b}[k][i] &= A_d[k][i-1] - A_{d,b}[k][i-1] \qquad (b = 1, \ldots, 2^d) \\
A_{d,*}[k][i] &= \sum_{k'=3}^{k} H_{d,k'} \cdot A_d[k][i-k'+1] \\
A_d[k][i] &= \sum_{b'=1}^{2^d} A_{d,b'}[k][i] + A_{d,*}[k][i].
\end{aligned} \tag{2}$$

Eq. 2 can be simplified into a recurrence on A itself

$$\begin{aligned}
A_d[k][i] &= \sum_{b'=1}^{2^d} (A_d[k][i-1] - A_{d,b'}[k][i-1]) + A_{d,*}[k][i] \\
&= 2^d A_d[k][i-1] - \sum_{b'=1}^{2^d} A_{d,b'}[k][i-1] + A_{d,*}[k][i] \\
&= 2^d A_d[k][i-1] - (A_d[k][i-1] - A_{d,*}[k][i-1]) + A_{d,*}[k][i] \\
&= (2^d - 1)A_d[k][i-1] + \sum_{k'=3}^{k} H_{d,k'} \cdot (A_d[k][i-k'] + A_d[k][i-k'+1]).
\end{aligned} \tag{3}$$

The base cases of the recurrence are $A_d[k][0] = 1$ and $A_d[k][1] = 2^d$.

Based on the numbers of spines, we count the k-ary permutation trees with n leaves as follows:

$$S_{d,k}[n] = \sum_{i=1}^{n-1} A_d[k][i] \cdot \sum_{\substack{l_1 \ldots l_i \\ \sum_j l_j = n-1}} \prod_{j=1}^{i} S_{d,k}[l_j] \tag{4}$$

with the base case $S_{d,k}[1] = 1$. The outermost summation is over spines of different lengths i. The inner summation is over all possible distributions of $n-1$ children into the i subtrees attached to the spine.

To simplify the equation, we enter the domain of generating functions. A sequence's generating function R is defined as a power series of a new variable, x, where each term x^n has as its coefficient the nth term of the series. We define the generating function for $S_{d,k}$ as

$$R_{d,k}(x) = \sum_{n=1}^{\infty} S_{d,k}[n] \cdot x^n.$$

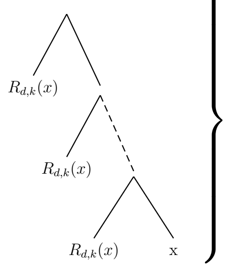
which can be interpreted as an infinite sum of all d-dimensional k-ary permutation trees. Starting with eq. 4, we can derive the following equation for $R_{d,k}(x)$:

$$R_{d,k}(x) \quad = \quad x \cdot \sum_{i=0}^{\infty} A_d[k][i] \cdot R^i_{d,k}(x) \tag{5}$$

where the initial factor of x shifts the series by one position, corresponding to the fact that we convolve terms summing to $n-1$ rather than n in eq. 4.

At a more intuitive level, each term in eq. 5 corresponds to d-permutation trees with a rightmost spine of length i, with a single leaf at the end of the spine (the x factor), and all possible combinations of trees attached at the i other points along the spine ($R^i_{d,k}(x)$ in the equation)

$R_{d,k}(x) = \sum_i$



$i.$

This simplified equation will help us eliminate the dependency on A in computing the series

S. We now subtract from eq. 5 its multiples

$$\begin{aligned} & R_{d,k}(x) - (2^d-1)R_{d,k}^2(x) - \sum_{k'=3}^{k} H_{d,k'} \left(R_{d,k}^{k'}(x) + R_{d,k}^{k'+1}(x)\right) \\ = \quad & x \cdot \sum_{i=0}^{\infty} \Bigg(A_d[k][i] - (2^d-1)A_d[k][i-1] - \\ & \qquad \sum_{k'=3}^{k} H_{d,k'} \cdot (A_d[k][i-k'] + A_d[k][i-k'+1]) \Bigg) R_{d,k}^i(x) \\ = \quad & x \cdot \left(1 + R_{d,k}(x) + \sum_{i=2}^{\infty} 0 \cdot R_{d,k}^i(x)\right) \\ = \quad & x + xR_{d,k}(x) \end{aligned}$$

where we have applied eq. 3 to show that all terms where $i \geq 2$ equal zero. Rearranging the above equation, we obtain a general form for R that does not require computing A

$$R_{d,k}(x) = x + xR_{d,k}(x) + (2^d-1)R_{d,k}^2(x) + \sum_{k'=3}^{k} H_{d,k'} \left(R_{d,k}^{k'}(x) + R_{d,k}^{k'+1}(x)\right). \tag{6}$$

Eq. 6 can be converted back into a recurrence on our original sequence $S_{d,k}$, by replacing powers of R with convolutions of S

$$\begin{aligned} S_{d,k}[n] \quad = \quad & S_{d,k}[n-1] \\ & + (2^d-1) \sum_{i_1+i_2=n} S_{d,k}[i_1] \cdot S_{d,k}[i_2] \\ & + \sum_{k'=3}^{k} H_{d,k'} \left(\sum_{\substack{l_1 \ldots l_{k'} \\ \sum_j l_j = n}} \prod_{j=1}^{k'} S_{d,k}[l_j] + \sum_{\substack{l_1 \ldots l_{k'+1} \\ \sum_j l_j = n}} \prod_{j=1}^{k'+1} S_{d,k}[l_j] \right) \end{aligned} \tag{7}$$

with the base case $S_{d,k}[1] = 1$. Compared to eq. 4, this equation does not depend on the numbers of spines. When $d = 1$ and $k = 2$, our sequence $S_{1,2}$ is the Large Schröder Number (sequence A006318 of [11]).

4 Asymptotics of Simple d-permutations

Table 1 shows the values of $H_{2,k}$ $(1 \leq k \leq 20)$. $H_{d,k'}$ $(1 \leq k' \leq k)$ plays an important role in the generation of the k-ary parsable d-permutations. When $d = 1$, the sequence of H is called the number of *simple* permutations by Albert et al. [2]. They show that $\frac{H_{1,n}}{n!} \to e^{-2}$ as n goes to infinity, demonstrating that H grows very rapidly. We show that when $d \geq 2$, $\frac{H_{d,n}}{(n!)^d} \to 1$.

k	$H_{2,k}$
1	1
2	4
3	8
4	172
5	5204
6	222716
7	12509188
8	889421564
9	78097622276
10	8312906703868
11	1056520142488580
12	158263730949406716
13	27626236450406776836
14	5563092167972597137404
15	1280742543230231763615748
16	334405228960123174787678204
17	98317121153947856929753989124
18	32339023133437156084762282819580
19	11831483864832785151824395066146820
20	4789379698138059405310741712024371196

Table 1: The numbers of two-dimensional simple permutations, i.e., 2-permutations of k that are *not* $k-1$-ary parsable.

Our proof technique is analogous to that used by Albert et al. [2]. To count the number of simple d-permutations, we subtract from $(n!)^d$ the numbers of permutations with component blocks of sizes ranging from 2 to $n-1$. More precisely, we let $p_{d,n,k}$ be the number of d-permutations of n whose non-trivial minimal non-decomposable d-dimensional permuted sequence is of size k. So we have the following equation:

$$H_{d,n} = (n!)^d - \sum_{k=2}^{n-1} p_{d,n,k}. \tag{8}$$

When $d = 1$, it has been shown that $\sum_{k=3}^{n-1} \frac{p_{1,n,k}}{n!} = O(n^{-1})$ [2]. However, the series $p_{1,n,2}$ grows rapidly as n increases. The counting of $p_{1,n,2}$ is a problem studied in an earlier context of runs of consecutive elements in permutations [13]. It turns out the numbers of consecutive runs follow a Poisson distribution of mean value 2. $\frac{p_{1,n,2}}{n!}$ is the probability mass contributed by the permutations containing a non-zero number of consecutive runs, which turns out to be $1-e^{-2}$. Thus, the ratio of simple permutations versus all permutations is asymptotically $1-(1-e^{-2}) - O(n^{-1}) = e^{-2} + O(n^{-1})$.

Now we look into the cases when $d \geq 2$. First of all, we generalize Lemma 7 of Albert et al. [2]:

Lemma 4.1. *For any fixed positive integer c:*

$$\sum_{k=c+2}^{n-c} \frac{p_{d,n,k}}{(n!)^d} = O(n^{-c\cdot d}).$$

Proof :
Following the same line of reasoning as in the original paper, we generalize the upper bound by raising an additional power of d. We observe that

$$p_{d,n,k} \leq H_{d,k}(n-k+1)((n-k+1)!)^d$$

since the right hand side overcounts d-permutations with more than one minimal block of size k. Using the fact that $H_{d,k} \leq (k!)^d$ and $(n-k+1) \leq (n-k+1)^d$, we arrive at the following inequality:

$$p_{d,n,k} \leq (k!(n-k+1)(n-k+1)!)^d$$

for which we can invoke the bounding results for

$$k!(n-k+1)(n-k+1)!$$

and we have the main result. ■

Based on Lemma 4.1, we know $\sum_{k=3}^{n-1} \frac{p_{d,n,k}}{(n!)^d} = O(n^{-d})$. We have the following lemma regarding $p_{d,n,2}$:

Lemma 4.2. *When $d \geq 2$,*

$$\frac{p_{d,n,2}}{(n!)^d} = O(n^{-(d-1)}).$$

Proof :
The idea is similar to that used for analyzing consecutive runs [13]. We imagine a stochastic process for generating a d-permutation in which we randomly pick a d-dimensional alignment link for i at the i-th step. Whenever we have a minimal block of size 2 in the d-permutation, we have two adjacent alignment links forming a binary permuted sequence. The probability of having one such binary block is bounded by $O((2/n)^d)$. So the mean of the number of d-dimensional consecutive runs is $O(2^d/n^{d-1})$. When $d \geq 2$, as n approaches infinity, the mean approaches zero. Thus, we can invoke the Markov inequality to bound the probability of observing a nonzero number of consecutive d-dimensional runs as $O(n^{-(d-1)})$. ■

Theorem 4.3. *When $d \geq 2$,*

$$\frac{H_{d,n}}{(n!)^d} = 1 + O(n^{-(d-1)}).$$

Proof :
Based on Lemma 4.1, Lemma 4.2, and eq. 8, the ratio is asymptotically $1 - O(n^{-(d-1)}) - O(n^{-d}) = 1 + O(n^{-(d-1)})$. ■

	$G_{2,k}$		$G_{2,k}$
2	13.93	12	57.60
3	17.91	13	63.22
4	22.08	14	69.17
5	26.03	15	75.43
6	29.97	16	82.01
7	34.00	17	88.90
8	38.19	18	96.10
9	42.61	19	103.60
10	47.31	20	111.41
11	52.30		

Table 2: The growth rate of $S_{2,k}$, which is the asymptotic ratio of $S_{2,k}[n]/S_{2,k}[n-1]$.

5 Growth Rates for Factorizable d-permutations

Since $S_{d,k}$ represents a monotonically increasing combinatorial sequence with an algebraic generating function (by Eq. 6), a standard argument [4] shows that the ratio between successive numbers in $S_{d,k}$ approaches a constant, which we call the growth rate $G_{d,k}$. A more careful analysis of Eq. 6 shows that it is a single-equation algebraically aperiodic irreducible polynomial system specified by a Context-free class with one combinatorial equation, which indicates that the underlying sequence $S_{d,k}$ satisfies $S_{d,k} \sim \frac{\gamma}{\sqrt{\pi n^3}} \omega^n$ [4]. We are interested in the asymptotic behavior of $\omega = G_{d,k}$ as k goes to infinity for a certain d.

Zhang and Gildea [15] show that the difference between successive growth rates $G_{1,k}$

$$G_{1,k} = \lim_{n \to \infty} \frac{S_{1,k}[n]}{S_{1,k}[n-1]}$$

approaches $1/e \simeq .37$

$$\lim_{k \to \infty} \{G_{1,k} - G_{1,k-1}\} = e^{-1}.$$

In this section, we study the generalized problem of the asymptotic behavior of $G_{d,k}$

$$G_{d,k} = \lim_{n \to \infty} \frac{S_{d,k}[n]}{S_{d,k}[n-1]}.$$

Table 2 shows the change of growth rate $G_{2,k}$ as k increases from 2 to 20. It seems the difference between successive rates is growing roughly linearly. This observation is confirmed by our main result in this section, the following theorem:

Theorem 5.1.

$$G_{d,k} = \left(\frac{k}{e}\right)^d + O(k^{d-1} \cdot \log k).$$

Proof :
We approximate $G_{d,k}$ by bounding $S_{d,k}$ using exponential functions from both directions. To get a lower bound of $S_{d,k}$, we under-count the k-ary trees by focusing only on the k-ary

d-permutation trees in which the maximum branching factor k is used throughout on the level immediately above the leaves, and these k-ary nodes are themselves ordered according to the identity permutation in each dimension. For sufficiently large k,

$$\begin{aligned} S_{d,k}[n] &\geq H_{d,k}^{n/k} \\ G_{d,k} &\geq H_{d,k}^{1/k} \\ &= \left((1-\epsilon)(k!)^d\right)^{1/k} \\ &= \left((1-\epsilon')\sqrt{2\pi}k^{k+\frac{1}{2}}e^{-k}\right)^{d/k} \\ &= \left(\frac{k}{e}\right)^d \left((1-\epsilon')\sqrt{2\pi}k^{\frac{1}{2}}\right)^{d/k} \\ &\geq \left(\frac{k}{e}\right)^d. \end{aligned} \tag{9}$$

To derive an upper bound on $G_{d,k}$, we return to the domain of generating functions. Because our generating function for any given k

$$R_{d,k}(x) = x + xR_{d,k}(x) + (2^d-1)R_{d,k}^2(x) + \sum_{k'=3}^{k} H_{d,k'}\left(R_{d,k}^{k'}(x) + R_{d,k}^{k'+1}(x)\right)$$

is analytic and all terms of the sequence S are positive, we can put a bound on the growth rate of S by finding the largest x for which the generating function converges [8].

We wish to understand the change in growth rate as k increases, but for each value of k, $R_{d,k}$ is a different (and increasingly complex) function. Thus we need a function of k, say $f_d(k)$, such that $R_{d,k}(f_d(k))$ is guaranteed to converge for all k as k increases. Because of the form of our generating function, it is easier to work with its inverse, choosing a value for the function $R_{d,k}(x)$ itself and then solving for x to obtain $f_d(k)$. Let $z = R_{d,k}(x)$ to simplify notation

$$x = \frac{z-(2^d-1)z^2}{1+z} - \sum_{k'} H_{d,k'}z^{k'} \tag{10}$$

we can choose a value of z as a function of k:

$$z = \left(\frac{e}{k}\left(\frac{1}{k^4}\right)^{\frac{1}{k}}\right)^d$$

and show that there always exists an x such that $z = R_{d,k}(x)$. The intuition behind this choice of z is that for x to remain positive, the first term must be larger than the sum over $H_{d,k'}$ terms. $H_{d,k'}$ grows very quickly, as $(k'!)^d$, and $z^{k'}$ must decrease faster than $H_{d,k'}$ grows. In the Appendix, we show that:

$$\sum_{k'} H_{d,k'}z^{k'} = O(k^{1-3d}).$$

For our growth rate, we need to analyze the behavior of $\frac{1}{x}$ as k increases, by substituting the result above into eq. 10

$$x = \frac{z - (2^d - 1)z^2}{1 + z} + O(k^{1-3d})$$

using $\frac{1}{1-y} = 1 + O(y)$ as $y \to 0$:

$$= z + O(z^2) + O(k^{1-3d})$$

using $z = O(k^{-d})$:

$$= z + O(k^{-2d}).$$

The upper bound on the growth rate is the inverse of x:

$$\frac{1}{x} = \frac{1}{z}\left(\frac{1}{1 + O(k^{-d})}\right)$$

again using $\frac{1}{1-y} = 1 + O(y)$:

$$\begin{aligned}
&= \frac{1}{z}\left(1 + O(k^{-d})\right) \\
&= \frac{1}{z} + O(z^{-1}k^{-d}) \\
&= \frac{1}{z} + O(1) \\
&= \left(\frac{k}{e}\left(k^4\right)^{\frac{1}{k}}\right)^d + O(1) \\
&= \left(\frac{k}{e} e^{\frac{4}{k}\log k}\right)^d + O(1)
\end{aligned}$$

using $e^y = 1 + O(y)$ as $y \to 0$:

$$\begin{aligned}
&= \left(\frac{k}{e}\left(1 + O\left(\frac{4}{k}\log k\right)\right)\right)^d + O(1) \\
&= \left(\frac{k}{e}\right)^d + O(k^{d-1} \cdot \log k).
\end{aligned}$$

This upper bound, together with the lower bound of eq. 9, show that the dominating term of the growth rate is $\left(\frac{k}{e}\right)^d$. ∎

6 Conclusion

We view a sequence of multiple permutations as a combinatorial object and study the recursive decomposition of such objects. With probability almost one a given d-dimensional ($d \geq 2$) permutation is simple. Although the number of k-ary parsable d-permutations grows very fast: the ratio between successive terms approaches $\left(\frac{k}{e}\right)^d$, the number of all d-permutations grows even faster: as $(n!)^d$. Previous work has studied the algorithmic aspect of the problem with the notion of PQ tree for representing common intervals of d permutations. Our d-permutation trees are PQ trees with detailed permutation specification at each node.

Given our finding that the probability of maintaining the consecutive property across all dimensions is extremely low, an interesting topic for future work would be the exploration of alternative definitions of d-permutation decomposition which allow for a sequence of consecutive numbers in one dimension to become several segments in another dimension.

7 Acknowledgments

We thank Daniel Štefankovič for helpful discussions. This work was supported by NSF grants IIS-0546554 and IIS-0428020.

Appendix

In this appendix we show that, given the choice of $z = \left(\frac{e}{k}\left(\frac{1}{k^4}\right)^{\frac{1}{k}}\right)^d$ in Section 5, $\sum_{k'} H_{d,k'} z^{k'} = O(k^{1-3d})$.

$$
\begin{aligned}
\sum_{k'=3}^{k} H_{d,k'} z^{k'} &= \sum_{k'=3}^{k} (k'!)^d \left(\frac{e}{k}\left(\frac{1}{k^4}\right)^{\frac{1}{k}}\right)^{d \cdot k'} \\
&= \sum_{k'} \left(\sqrt{2\pi k'}\left(\frac{k'}{e}\right)^{k'}\left(\frac{e}{k}\right)^{k'}\left(\frac{1}{k^4}\right)^{\frac{k'}{k}}\right)^d
\end{aligned}
$$

(The error of Stirling's approximation for $k'!$ can be bounded by a factor of $1 + O(k'^{-1})$, which we omit.)

$$\leq \sum_{k'} \left(\sqrt{2\pi} \left(\frac{k'}{k} \right)^{k'} k^{1/2} \left(\frac{1}{k^4} \right)^{\frac{k'}{k}} \right)^d$$
$$= \sum_{k'} \left(\sqrt{2\pi} \left(\frac{k'}{k} \right)^{k'} k^{1/2 - \frac{4k'}{k}} \right)^d$$
$$= \left(\frac{\sqrt{2\pi}}{k^3} \right)^d \sum_{k'} \left(\left(\frac{k'}{k} \right)^{k'} k^{3.5 - \frac{4k'}{k}} \right)^d$$

It can be shown that each term within the sum is less than one [15]

$$\left(\frac{k'}{k} \right)^{k'} k^{3.5 - \frac{4k'}{k}} < 1.$$

Therefore, for sufficiently large k,

$$\begin{aligned}
\sum_{k'} H_{d,k'} z^{k'} &\leq \left(\frac{\sqrt{2\pi}}{k^3} \right)^d \sum_{k'} \left(\left(\frac{k'}{k} \right)^{k'} k^{3.5 - \frac{4k'}{k}} \right)^d \\
&\leq \left(\frac{\sqrt{2\pi}}{k^3} \right)^d \sum_{k'} 1 \\
&\leq \left(\frac{\sqrt{2\pi}}{k^3} \right)^d k \\
&= O(k^{1-3d})
\end{aligned}$$

References

[1] Albert V. Aho and Jeffery D. Ullman. *The Theory of Parsing, Translation, and Compiling*, volume 1. Prentice-Hall, Englewood Cliffs, NJ, 1972.

[2] M. H. Albert, M. D. Atkinson, and M. Klazar. The enumeration of simple permutations. *J. Integer Sequences* **6** (2003), 03.6.??.

[3] Binh Minh Bui-Xuan, Michel Habib, and Christophe Paul. Revisiting T. Uno and M. Yagiura's algorithm. In *The 16th Annual International Symposium on Algorithms and Computation (ISAAC'05)*. 2005, pp. 146–155.

[4] Philippe Flajolet and Robert Sedgewick. Analytic combinatorics: Functional equations, rational and algebraic functions. Technical Report 4103, INRIA, 2001. 98 pages.

[5] Steffen Heber and Jens Stoye. Finding all common intervals of k permutations. In *Combinatorial Pattern Matching, 12th Annual Symposium*. Springer-Verlag, 2001, pp. 207–218.

[6] Gad M. Landau, Laxmi Parida, and Oren Weimann. Gene proximity analysis across whole genomes via PQ trees. *J. Computational Biology* **12** (2005), 1289–1306.

[7] I. Dan Melamed. Multitext grammars and synchronous parsers. In *Proceedings of the 2003 Meeting of the North American chapter of the Association for Computational Linguistics (NAACL-03)*. Edmonton, 2003, pp. 158–165.

[8] Andrew Odlyzko. Asymptotic enumeration methods. In R. L. Graham, M. Groetschel, and L. Lovasz, editors, *Handbook of Combinatorics vol. 2*, Elsevier. 1995, pp. 1063–1229.

[9] Giorgio Satta and Enoch Peserico. Some computational complexity results for synchronous context-free grammars. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing (HLT/EMNLP)*. Vancouver, Canada, 2005, pp. 803–810.

[10] L. Shapiro and A. B. Stephens. Bootstrap percolation, the Schröder numbers, and the n-kings problem. *SIAM J. Discrete Math.* **4** (1991), 275–280.

[11] N. J. A. Sloane. The On-Line Encyclopedia of Integer Sequences, 2006.

[12] Takeaki Uno and Mutsunori Yagiura. Fast algorithms to enumerate all common intervals of two permutations. *Algorithmica* **26** (2000), 290–309.

[13] J. Wolfowitz. Note on runs of consecutive elements. *Ann. of Math. Stat.* **15** (1944), 97–98.

[14] Dekai Wu. Stochastic inversion transduction grammars and bilingual parsing of parallel corpora. *Computational Linguistics* **23** (1997), 377–403.

[15] Hao Zhang and Daniel Gildea. Efficient factorization of synchronous context-free grammars. Technical Report 889, University of Rochester, 2006.

[16] Hao Zhang, Liang Huang, Daniel Gildea, and Kevin Knight. Synchronous binarization for machine translation. In *Proceedings of the Human Language Technology Conference/North American Chapter of the Association for Computational Linguistics (HLT/NAACL)*, 2006, pp. 256-263.

2000 *Mathematics Subject Classification*: Primary: 05A16; Secondary: 05A05, 05A15.
Keywords: asymptotic enumeration, permutation.

(Concerned with sequences A006318 and A111111.)

Received December 18 2006; revised version received June 5 2007. Published in *Journal of Integer Sequences*, June 5 2007.

Return to Journal of Integer Sequences home page.