Adjacencies in Permutations

*[1,2]Bhadrachalam Chitturi [1]Krishnaveni K S

[1]Department of Computer Science, Amrita Vishwa Vidyapeetham, Amritapuri Campus, Kollam, Kerala 690525, India.
[2]Department of Computer Science, University of Texas at Dallas, Richardson, Texas 75083, USA.

Abstract: A permutation on an alphabet Σ, is a sequence where every element in Σ occurs precisely once. Given a permutation π= (π_1, π_2, π_3,....., π_n) over the alphabet Σ ={0, 1, ... , n−1 } the elements in two consecutive positions in π e.g. π_i and π_{i+1} are said to form an *adjacency* if π_{i+1} =π_i+1. The concept of adjacencies is widely used in computation. The set of permutations over Σ forms a symmetric group, that we call P_n. The identity permutation, $I_n \in P_n$ where I_n =(0,1,2,...,n−1) has exactly n−1 adjacencies. Likewise, the reverse order permutation $R_n(\in P_n)$=(n−1, n−2, n−3, n−4, ...,0) has no adjacencies. We denote the set of permutations in P_n with exactly k adjacencies with P_n(k). We study variations of adjacency. A transposition exchanges adjacent sublists; when one of the sublists is restricted to be a prefix (suffix) then one obtains a prefix (suffix) transposition. We call the operations: transpositions, prefix transpositions and suffix transpositions as block-moves. A particular type of adjacency and a particular block-move are closely related. In this article we compute the cardinalities of P_n(k) i.e. $\forall_k$ |P_n (k) | for each type of adjacency in $O(n^2)$ time. Given a particular adjacency and the corresponding block-move, we show that $\forall_k$ | $P_n(k)$ | and the expected number of moves to sort a permutation in P_n are closely related. Consequently, we propose a model to estimate the expected number of moves to sort a permutation in P_n with a block-move. We show the results for prefix transposition. Due to symmetry, these results are also applicable to suffix transposition.

Key words: Adjacency, permutations, recurrence relations, sorting, transpositions, prefix transpositions, expected number of moves.

1 Introduction

Sets and multisets are collections of objects. Given an object *o* and a set *S*, one can only enquire whether $o \in S$. If one imposes order on the objects within a set then one obtains sequences; e.g. vectors, strings, permutations etc.. In a sequence *T* if $x \in T$ then one can also query the position of *x*. A permutation on an alphabet Σ, is a sequence where every object in Σ occurs precisely once. In a string a symbol can repeat whereas in a permutation there is bijection from the positions to the symbols. Given a permutation π= (π_1, π_2, π_3,.....,π_n) over the alphabet Σ ={0, 1, ... , n−1 } π_i and π_{i+1} form an *adjacency* if π_{i+1} =π_i+1, we call this as *normal* adjacency or *type 1* adjacency. The concept of adjacencies is widely used computation. The set of permutations with n symbols is called a symmetric group that we denote with P_n. The identity permutation with n symbols denoted by I_n where I_n=(0,1,2,...,n−1) has exactly n−1 adjacencies. Likewise, the reverse order permutation denoted by R_n where R_n=(n−1, n−2, n−3, n−4, ...,0) has no adjacencies. We say that $\pi^a \in P_n(k)$ reduces to $\pi^b \in P_{n-k}(0)$ if π^b is obtained by eliminating all the adjacencies in π^a. For example, (4, 5, 2, 1, 3, 0) in P_6 reduces to (4, 2, 1, 3, 0) in P_5 where (4, 2, 1, 3, 0) is *irreducible*. The algorithm for reduction identifies and eliminates all maximal blocks of consecutive adjacencies (> 0). Let the first symbol of one such block B be f and the last one be l then B is replaced by f and the value of every symbol with value >l is decreased by l−f. This process is repeated until all adjacencies are eliminated.

* *Corr. author*

In this article, $\forall k$ we compute the cardinalities of P_n(k) that is, we compute $\forall_k$ $|P_n(k)|$ in $O(n^2)$ time. We call the classic adjacency as type 1 adjacency. We define three variations of it. The first variation is back-adjacency or simply b-adjacency or type 2 adjacency where in addition to the normal adjacencies if π_n =n−1 then it forms an adjacency with (an imagined) π_{n+1} =n. The second variation of adjacency is called front-adjacency or simply f-adjacency or type 3 adjacency where in addition to the normal adjacencies if π_1 =0 then it forms an adjacency with (imagined) π_0 =−1. The third variation of adjacency is called front-and-back-adjacency or simply bf-adjacency or type 4 adjacency where in addition to the normal adjacencies if π_n =n−1 then it forms an adjacency with (imagined) π_{n+1} =n and if π_1 =0 then it forms an adjacency with (imagined)$\pi_0 = -1$. P_n(k) denotes the set of permutations in P_n with exactly k adjacencies; the type of adjacency will be evident from the context. We compute $\forall k$ $|P_n(k)|$ in $O(n^2)$ time for any type of adjacency. When necessary we employ the notation $P_n(k, i)$ where i indicates the type of adjacency. We call two permutations π^a and π^b as *mirrors* of each other if they are corresponding permutations from two different alphabets. That is, (0, 2, 1) and (1, 3, 2) are mirrors of each other. Mirrors are equivalent; i.e. the numbers of adjacencies and the numbers of moves that are required to sort them are identical.

The concept of adjacencies is inherent in sorting by comparison algorithms. Quicksort seeks to reduce the inversions in a permutation by swapping two distant objects that form an inversion whereas bubble sort swaps adjacent objects that form an inversion and thus, it reduces exactly one inversion per swap. Adjacencies and inversions are inherently related. An inversion exists if and only if the total number of (type 1) adjacencies is less than n−1. All the algorithms terminate when n b-adjacencies are created [14]. Graham et al. study the related topics: ascents, cycles, left-to-right maxima and excedances in permutations [19]. Given π= (π_1, π_2, π_3,....., π_n) an ascent is defined as a position j where $\pi_j < \pi_{j+1}$ and the corresponding recurrence relation is given by α(n,k)= α(n−1,k)*(k+1) + α(n−1,k−1)*(n−k) [17, 19, 24]. $\alpha(n, k)$ denotes the number of permutations in P_n that have exactly k ascents. The numbers thus generated are known as Eulerian numbers. The cycles in permutations correspond to Stirling numbers of the first kind [26, 19]. The left to right maxima corresponds to π_j where $\forall_{i<j}$ $\pi_i < \pi_j$ [19]. A symbol π_j in π is an excedance if j< π_j [19].

Transforming permutations with transpositions, and prefix transpositions has been well studied. The symmetric *distance* between two permutations α and β with a symmetric operation τ i.e. $d_\tau(\alpha, \beta)$ is the minimum number of τ operations required to transform α into β or vice versa. So, the transposition distance between α and β, i.e. $d_T(\alpha, \beta)$ is the minimum number of transpositions required to transform α into β or vice versa. The concept of *breakpoint* was used in many articles, e.g. [2, 15] where a breakpoint denotes an absence of an adjacency. Bafna and Pevzner [2] studied sorting permutations in P_n with transpositions and showed a lower bound of $\lfloor n/2 \rfloor$+ 1 and an upper bound of $\frac{3n}{4}$. They also gave a 1.5 approximation algorithm for the same. Eriksson et al. improved the upper bound to $\frac{2n}{3}$ [16] and also showed that R_n, the reverse order permutation can be sorted in $\frac{n+1}{2}$ transpositions.

Dias and Meidanis [15]studied the prefix transposition distance over P_n and showed that: (a) n−1 is an upper bound, (b)$\frac{n}{2}$ is a lower bound, and (c) R_n, can be sorted in $\frac{3n}{4}$ prefix transpositions. They conjectured that R_n is the hardest permutation to sort. Recall the R_n has no adjacencies. Chitturi and Sudborough improved the lower bound to $\frac{3n}{4}$ [8] and the upper bound to n-$\log_{\frac{9}{2}} n$ [9]. Labarre [20] improved the lower bound of prefix transposition distance over P_n to $\frac{3n}{4}$. Recently, Chitturi [6] showed that an upper bound for the prefix transposition distance over P_n is $n - \log_{\frac{7}{2}} n$.

Sorting permutations with prefix reversals i.e. flips, also known as the pancake problem has been widely studied. The best known upper bound for this problem is $18n/11 + O(1)$ [7]. Cibulka showed that sorting a random stack of n pancakes can be done with at most $\frac{17n}{12}$+O(1) flips on average. The average number of flips of the optimal algorithm for sorting stacks of n burnt pancakes is shown to be between n+$\Omega(\frac{n}{logn})$ and $\frac{7n}{4}$+O(1) and the author conjectures that it is n+$\Theta(\frac{n}{logn})$ [13].

Bulteau et al. show that sorting permutations by transpositions is NP-hard [3]. Thus, it is desirable to estimate the expected number of moves to sort a permutation $\in P_n$ with transpositions. It is believed that sorting permutations by either prefix or suffix transpositions is intractable. So, the model that estimates the expected number of moves to sort permutations with various block-moves is sought.

The block moves are studied on strings as well. The earliest known articles on transforming strings with transpositions and prefix transpositions are [12] and [4, 9, 8] respectively. Adjacent transpositions, where adjacent elements swap their positions, on permutations has a proven exact upper bound [1, 5, 21] whereas an efficient algorithm to count the exact number of adjacent transpositions required to transform one string to another string is presented in [10]. Further, cyclic *short swaps* and adjacent transpositions are studied on permutations in [18].

The main contributions of this article are: (i) computing $\forall_k$ $|P_n$ (k)| for any type of adjacency in O(n^2) time, (ii) a theoretical framework that forms a basis for models that estimate the expected number of block-moves to sort a permutation in $P_n(0)$ (and thus, in P_n). We were made aware of OEIS and [28] by an anonymous referee which lead to [27]. An examination of some intger sequences in OEIS reveals that our article provides an alternative explanation for some of the known integer sequences. To our knowledge, the current type of exploration of adjacencies in permutations and their applications are novel.

2 Regular adjacencies

Symbols π_i and π_{i+1} form an adjacency if π_{i+1} =π_i+1 in regular i.e. type 1 adjacencies. The sorted permutation with n symbols, i.e. $\mathbf{I}_n$ where $\forall_k \pi_k$ = k, also called as the identity permutation, has the maximum number i.e. n−1 adjacencies. The reverse order permutation i.e. R_n where $\forall_k$ π_k = n−k, has zero adjacencies. A permutation is *reduced* or *irreducible* if it has no adjacencies [9]. Let π a member of P_7 be (4 6 3 1 2 0 5) then (3 5 2 1 0 4) is the reduced form of π where (1 2) is reduced to 1 and all the symbols with a value greater than two are decremented by one. This process is repeated until all the adjacencies are removed from π. Here, the resulting permutation is a member of P_6.

The following theorem establishes a recurrence relation to compute $|P_n(k)|$ for the first type of adjacencies.

Theorem 1 Let P_n(k) be the subset of P_n where any $\pi\in P_n$(k) has exactly k type 1 adjacencies. Let f(n,k) be the cardinality of P_n(k). Then f(n,k) = f(n−1,k−1)+ (n−1−k) * f(n−1,k) + (k+1)* f(n−1,k+1) where 0 ≤k<n.

The cardinalities of $P_0(n)$, $P_1(n)$ etc. occur in OEIS [25] with sequence numbers A000255, A000166 etc.. Tanny studied the cardinalities of the sets of permutations with n symbols and k successions (or type 1 adjacencies) [27]. He gave the expression for f(n,k) as follows where D_i is a derangement number for size i [27]. f(n,k)=$\binom{n-1}{k}$ (D_{n-k} +D_{n-1-k}).

Tanny also studied circular successions where π_i and $\pi_{\equiv(i+1)}$ form an adjacency if $\pi_{\equiv(i+1)} \equiv (1+\pi_i)$. Here $\equiv x$ denotes $x \mod n$. He showed that $\lim_{n\to\infty}$ ($Q^*(n,k)/n!$)=$e^{-1}/k!$ where $Q^*(n,k)$ denotes number of permutations with k circular successions [27].

Roselle determined the cardinality of P(n,r,s) as P(n,r,s)= $\binom{n-1}{s}$ P(n-s,r-s,0) where s denotes the number of type 1 adjacencies (that he calls successions) and r denotes the number of rises [23]. A rise in a permutation exists at a position i if $\pi_i < \pi_{i+1}$ [23].

3 Adjacency Variations

3.1 Type 2 adjacency

In type 2 adjacency or b-adjacency, in addition to the adjacencies in type 1 adjacency we imagine that π_{n+1}= n; i.e. if π_n= n−1 then π_n and π_{n+1} form an adjacency. The sorted permutation with n symbols, i.e. $\mathbf{I}_n$ where $\forall_k$ π_k =k, also called as the identity permutation, has the maximum number i.e. n adjacencies. The reverse order permutation i.e. $\mathbf{R}_n$ where $\forall_k$ π_k =n−k, has zero adjacencies. If π = (4 6 3 5 0 2 1 7) then (4 6 3 5 0 2 1) is the reduced form of it where π_n is deleted because π_n= n−1.

Theorem 2 Let f(i, j) denote the number of permutations in P_i with exactly j adjacencies. Then the recurrence relation for f(i, j) is:
f(i, j)= (f(i−1, j−1)−f(i−2, j−2))*2 +f(i−2, j−2) +
(f(i−1, j+1)−f(i−2, j))*(j+1)+f(i−2, j)*(i−j−1)+
(f(i−1, j)−f(i−2, j−1))*(i−j−2) +
f(i−2, j+1)*(j+1); 0≤ j≤i+1.

3.2 Type 3 adjacency

In type 3 adjacency or f-adjacency, in addition to the adjacencies in type 1 adjacency we imagine that π_0=−1. That is, if π_1= 0 then π_0 and π_1 form an adjacency. $\mathbf{I}_n$ has the maximum number i.e. n adjacencies and $\mathbf{R}_n$ has zero adjacencies. Type 2 and type 3 adjacencies are symmetrical.In addition to the adjacencies defined in type 1 adjacency, an adjacency is defined between π_n and π_{n+1} in b-adjacency whereas the same is defined between π_0 and π_1 in the f-adjacency. The recurrences governing $\forall k$ $|P_k(n)|$ and their base values (i.e. $n \leq 4$) are identical for type 2 and type 3 adjacencies. Thus, yielding identical values for $|P_k(n)|$ for f-adjacency and b-adjacency.

The cardinalities of $P_0(n)$, $P_1(n)$ etc. occur in OEIS [25] with sequence numbers A000166 denoting subfactorial or rencontres numbers, or derangements: number of permutations of n elements with no fixed points; A000240: rencontres numbers: number of permutations of [n] with exactly one fixed point etc..

3.3 Type 4 adjacency

Type 4 adjacency or bf-adjacency has additional adjacencies defined compared to type 1 adjacency. We imagine that π_{n+1}= n and $\pi_0 = -1$. That is if π_n= n−1 then π_n and π_{n+1} form an adjacency; likewise, if π_1=0 then π_0 and π_1 form an adjacency. $\mathbf{I}_n$ has the maximum number of adjacencies, i.e. n+1 and R_n has zero adjacencies. If π = (0 4 6 3 5 2 1 7) then (4 6 3 5 2 1) is the reduced form of it where π_n is deleted because π_n= n−1 and π_1 is deleted because π_1= 0. The following theorem establishes a recurrence relation to compute $|P_n(k)|$ for bf-adjacency.

The cardinalities of $P_0(n)$, $P_1(n)$ and $P_2(n)$ occur in OEIS [25] with the following sequence numbers. A000757: Number of cyclic permutations of n symbols with no [i] immediately followed by [i+1] where [i] denotes $i\%n$; A135799: second column (k=1) of triangle A134832 (circular succession numbers); A134515: third column (k=1) of triangle A134832, etc..

Theorem 3 Let f(i, j) denote the number of permutations in P_i with exactly j adjacencies. Then the recurrence relation for f(i, j) is:
f(i, j)= (f(i−1, j)−f(i−2, j−1))*(i−j−2) +
(f(i−1, j−1)−f(i−2, j−2))*2 +f(i−2, j−2) +
(f(i−1, j+1)−f(i−2, j))*(j+1)+f(i−2, j)*(i−j−1)+
f(i−2, j+1)*(j+1); 0≤ j≤i+1.

4 A General Model for Block Moves

Transpositions, prefix transpositions and suffix transpositions are called as block-moves in this article. In the sequel a move refers to any of the above operations and its meaning is clarified by the context. If we are referring to a particular operation then the context clarifies the same. We assume that the permutation that we are sorting is irreducible. A prefix transposition can either create or destroy zero, one or two adjacencies. Likewise, a transposition can either create or destroy zero, one, two or three adjacencies. The moves that create one, two or three adjacencies are called a *single*, a *double* and a *triple* respectively. A move can also break adjacencies, however, because the permutation that is being sorted is irreducible we do not consider such moves.
Bulteau et al. show that sorting permutations by transpositions is NP-hard [3]. Thus, it is desirable to estimate the expected number of moves to sort a permutation $\in P_n$ with transpositions. It is believed that sorting permutations by either prefix transpositions or suffix transpositions is also hard. So, a model that estimates the expected number of moves to sort permutations with various block-moves is sought.

The first type of adjacencies can be used for estimating the expected number of moves to sort a permutation with any block-move, e.g. transposition, prefix transposition, suffix transposition, prefix/suffix transposition. However, the other types of adjacencies are more apt depending on the operation. For example, f-adjacency is applicable to prefix transpositions and the b-adjacency is applicable to suffix transpositions. The bf-adjacency is applicable for transpositions.

Christie [11] showed that $d_t(\pi, I_n) = d_t(\pi^*, I_n)$ where $d_t(A, B)$ is the transposition distance between A and B and π^* is the reduced form of π. Similar to an optimal sequence of transpositions that sort a given permutation, an optimal sequence of prefix transpositions also need not break any adjacencies. It follows that $d_{pt}(\pi, I_n) = d_{pt}(\pi^*, I_n)$.

Theorem 4 Let S be the maximal set of permutations in $P_n(n-k)$ such that every permutation in S yields the same permutation in $P_k(0)$ upon reduction. If $|\ S\ |= \mu$ then for any given permutation $\pi^x \in P_k(0)$ there are exactly μ permutations in P_n that reduce to π^x.

Proof: Consider P_n(n−k), the set of all permutations $\in P_n$ whose reduced length is k. That is, for each $\pi \in P_n$(n−k) there is a corresponding permutation $\pi^* \in P_k(0)$. The numbers of the corresponding transposition based moves to sort π and π^* are identical. Thus, we treat them as equivalent. Under this equivalence, we seek to show that P_n(n−k) is a multiset composed exclusively of some $c \in Z^+$ copies of each $\pi \in P_k(0)$.

We first analyze type 1 adjacency. Consider P_n(n−k) where n=3 and k=3 it consists of three permutations: $\{(021), (102), (210)\}$ corresponding to $P_3(0)$. Consider a split of I_5 =(0 1 2 3 4) into three substrings s_1, s_2, s_3 such that $\forall_i$ $|s_i|>0$ yielding I_5^* where I_5^*= (0 1, 2 3, 4). Here we separate adjacent substrings with comma. Note that in its reduced form I_5^* equals I_3. Consider an alternate split of I_5 =(0 1 2 3 4) into three substrings t_1, t_2, t_3 yielding I_5^*=(0, 1, 2 3 4). Let A={(0 1 4 2 3), (2 3 0 1 4),(4 2 3 0 1) } and B={(0 2 3 4 1), (1 0 2 3 4),(2 3 4 1 0) }. A and B belong to I_5 where where both these sets in their reduced form equal {(0 2 1), (1 0 2),(2 1 0) } which is the same as $P_3(0)$. That is, for each distinct split of (0 1 2 3 4) one will have three permutations in P_5. The number of such splits equals the number of integer solutions to the equation x1 + x2 +x3=5 i.e. $\binom{5-1}{3-1}$. Extending this argument to a general k and a general n (n>k), there are $\binom{n-1}{k-1}$ copies of P_n(n−k) in P_n. That is, any member of $P_k(0)$ has exactly $c = \binom{n-1}{k-1}$ occurrences in P_n.

Consider type 2 adjacency. Here, $P_3(0)$ ={(0 2 1), (2 1 0) }. Consider a split of I_5 =(0 1 2 3 4) into three substrings s_1, s_2, s_3 such that $\forall_i$ $|s_i|>0$ to yield I_5^* where I_5^*= (0 1, 2 3, 4). Note that I_5^* is equivalent to I_3. Consider an alternate split of I_5 =(0 1 2 3 4) into t_1, t_2, t_3 yielding I_5^*=(0, 1, 2 3 4). Let A={(0 1 4 2 3), (4 2 3 0 1) } and B={(0 2 3 4 1), (2 3 4 1 0) }. A and B belong to I_5 where where both these sets in their reduced form equal C, where C= {(0 2 1), (2 1 0) } which is the same as $P_3(0)$. First we note that if $n-1$ is in the last position then if a permutation from $P_{n-1}(n-k-1)$ precedes it then effectively we have a permutation in $P_n(n-k)$. In the above example consider a split of (0 1 2 3) into three non empty substrings, say (0 1, 2, 3). This will yield (3 2 0 1 4) as a permutation in $I_5(2)$ where the trailing 4 remains in the last position. Thus, extending the above argument to a general k and a general n (n>k), there are $\binom{n-1}{k-1}$ $+b$ copies of P_n(n−k) in P_n where b is the number of copies of P_{n-1}(n−k-1) in P_{n-1}. If one expands the recurrence then one obtains the total number of copies of of P_n(n−k) in P_n as $c = \sum_{n=1}^{n-k+1} \binom{n-i}{k-1}$ That is, any member of $P_k(0)$ has exactly c occurrences in P_n. Likewise, the same can be shown from f-adjacency.

Consider type 4 adjacency. Here, $P_3(0)$ ={ (2 1 0) }. Consider splitting of I_5 =(0 1 2 3 4) into three substrings s_1, s_2, s_3 such that $\forall_i$ $|s_i|>0$ yielding I_5^* where I_5^*= (0 1, 2 3, 4). Note that I_5^* is equivalent to I_3. Consider an alternative split of I_5 =(0 1 2 3 4) into t_1, t_2, t_3 yielding I_5^*=(0, 1, 2 3 4). Note that the sets of permutations A and B belong to I_5 where A={(4 2 3 0

1)} and B={ (2 3 4 1 0) } where both these sets in their reduced form are {(2 1 0)} which is the same as $P_3(0)$.

Let $\pi \in P_n$ and let $\pi_n = n-1$. We note that if a permutation from $P_{n-1}(n-k-1)$ precedes π_n then effectively we have a permutation in $P_n(n-k)$. Likewise, if $\pi_1 = 0$ is in the first position then if a permutation from $P_{n-1}(n-k-1)$ succeeds it then effectively we have a permutation in $P_n(n-k)$. Note that here a permutation from $P_{n-1}(n-k-1)$ is a mirror permutation that is it is defined on the alphabet $(1, 2, \ldots n-1)$. However, the above cases count the permutations that begin with 0 and end with $n-1$ twice. The number of such permutations is $|$ P_{n-2}(n-k-2) $|$. So, the recurrence is $|P_n(k)| = 2* |P_{n-1}(n-k-1)| -$ $|$ P_{n-2}(n-k-2) $|$. This recurrence relation has coefficients that are positive integers and the base case for this recurrence relation is $|$ $P_k(0)$ $|$ where the base case corresponds to one copy of $P_k(0)$. Thus, we can conclude that there are integral number of copies of $P_k(0)$ in P_n(n−k). ■

We define the set of irreducible permutations in P_n as the *vector alphabet* of P_n and it is denoted by $\alpha(P_n)$. Note that $\alpha(P_n)$ is a set i.e. if any permutation $\pi^x \in P_n$ reduces to an irreducible permutation π then only π will be a member of $\alpha(P_n)$. Let the *offset* denoted by δ be the term that must be added to n to obtain the maximum number of adjacencies possible for each type of adjacency. That is, $\delta = -1$ for type 1 adjacency, $\delta = 0$ for type 2 and type 3 adjacencies, and $\delta = 1$ for type 4 adjacency. Furthermore, let $P_k(0)^{c_k}$ denote c_k copies of the set $P_k(0)$ where $c_k \in Z^+$. The corollaries given below follow.

Corollary 4.1 $\alpha(P_n) = \bigcup_{k=1}^{n+\delta} P_k(0)$. ■

Corollary 4.2 $P_n = \bigcup_{k=1}^{n+\delta} P_k(0)^{c_k}$. ■

Corollary 4.3 Let $\phi(P_n(k))$ denote the average number of moves to optimally sort all permutations in P_n(k) with block moves. If $\phi(P_k(0))=\mu$ then $\phi(P_n(n-k)) = \mu$ for n>k. ■

Model

Theorem 1 and Theorem 3 lead to the corresponding algorithms that compute $\forall_k$ $P_n(k)$. Theorem 4 and Corollary 4.3 show that the distribution of $P_k(0)$ is uniform in P_n for all n>k. That is, every $\pi \in P_k(0)$ has exactly some c ($\in Z^+$) permutations in P_n that reduce to it. For each of the operation we evaluate the probabilities of executing a single, a double, and a triple i.e. p_1, p_2, p_3 on a (uniformly) random permutation in $P_n(0)$. A prefix or a suffix transposition does not admit a triple. Based on these probabilities, we compute the expected number of adjacencies created per one move in P_n, ψ. We employ ψ, the limiting value of ψ and the expected/estimated number of moves to optimally sort a permutations in $P_2(0) \ldots P_{i-1}(0)$ to compute the estimated number of moves that are required to sort a $\pi \in P_i(0)$. For example it can be seen that the limiting value of ψ for prefix transpositions is 1.5 from Observation 5.

The goal of sorting a permutation is to obtain a permutation of size one (after reduction) by starting with a permutation in $P_n(0)$. Thus, our first measure for expected number of moves to sort a permutation in $P_n(0)$ is $(n-1)/\psi$. The second measure computes the weighted average of the estimates for $P_{n-x}(0)$ and $P_{n-x+1}(0)$ and adds one to it where $n-x < n-\psi < n-x+1$. The weighted average is based on the position of $n-\psi$ in $[n-x, n-x+1]$. In this measure, in one move the size of the permutation is presumed to be reduced by ψ and we add the expected number of moves to sort the permutation of the resultant size. Note that this is not an integer size, so, we compute the weighted average. The first measure mimics the future behavior of the expected number of moves and the second measure mimics the past behavior of the expected number of moves. We take the mean of the above two measures as the estimate for the expected number of moves to sort a permutation in $P_n(0)$.

The following algorithm *Move_Count* estimates the expected number of moves to sort a permutation $\pi \in P_n(0)$. The average number of moves to sort a permutation $\pi \in P_i(0)$ for i=(2..limit) is computed by a branch and bound program. These values are used as base cases for *Move_Count*.

Algorithm Move_Count(i)

Precomputation. Execute a branch and bound algorithm that computes the average number of moves to sort for all permutations $\pi \in P_n(0)$ for n=2,....,limit. Let base[2..limit] hold the respective averages.

Intilization: cnt=0

for (i=limit+1,....max) **do**

j ← i

j← j−$\psi(n)$ \\ Observation 5, $\psi(n)$: expected number adjacencies that a move in P_n creates

x ← 1 + (j−$\lfloor j \rfloor$)* base[$\lceil j \rceil$] +($\lceil j \rceil$−j)*base[$\lfloor j \rfloor$]

y ← $(n-1)/\psi$

base[i] ← $(x+y)/2$

end for

We let X be the random variable that denotes the expected number of moves to sort a $\pi \in P_n$. Due to Theorem 4, E(X)= $\forall_i \Sigma$ f_i E(X_i) where X_i is the random variable that denotes the expected number of moves required to sort a $\pi \in P_n(i)$ and $f_i = \frac{|P_n(i)|}{n!}$. The estimate for E(X) is evaluated by algorithm Expected_Value. Note that we use the appropriate definition for adjacency, and the corresponding algorithm Adjacency_Countx is used (for the type of adjacency x, where x ∈1, 2, 4).

Algorithm Expected_value

Intilization: for (j= 2 ...limit) set E(X_j) from the branch and bound program.

offset that determines the maximum possible adjacencies =−1 for x=1, =0 for x=2 or 3, =+1 for x=4.

for (i=limit+1,....n) **do** compute E(X_j)by executing Move_Count(j). \\Note that this order is important

end for

for (i=limit+1,....n) **do**

for (i=0...j+offset) **do** $f_i=\frac{|P_j(i)|}{j!}$ \\ $|P_j(i)|$ is read from output of the appropriate Adjacency_Countx

end for

Estimate of E(X) = Σ $f_i E(X_i)$.

end for

Algorithm Move_Count uses the already computed averages for the number of moves required to sort all permutations with zero adjacencies of a given size, up to size eight. These numbers are used as base cases to compute the same for larger values of n.

The following theorem establishes a lower bound for the fraction of permutations in P_n that have exactly one adjacency. Note that a permutation $\pi \in P_n$ (k) where $k \geq 1$ can be reduced to $\pi^* \in P_{n-k}$. Thus, if the results for optimally sorting all permutations for P_i (i<n) are known then one need only look up the results for π^*. So, the computation of optimal sequences is required only for irreducible permutations. From the following theorem it follows that for P_n, approximately $\frac{n!}{e}$ permutations require the computation of optimal moves.

Observation 1 Let f_i (0)= $\frac{|P_n(0)|}{n!}$ where i is the type of adjacency $\in$ {1,2,3,4 } and $|P_n(0)|$ be the corresponding magnitude of the set of irreducible permutations $\in P_n$. We have the following inequalities:(i) $f_1(0)>\frac{1}{e}$. (ii) $f_2(0)\leq\frac{1}{e}$. (iii) $f_3(0)\leq\frac{1}{e}$.(iv) $f_4(0)<\frac{1}{e}$.

Proof: Consider the first type of adjacency. Recall that Σ={0,1,2,..., n−1}. The probability that n−1 is not present at a given position is $\frac{n-1}{n}$. Given that that n−1 does not occur at position i, the probability that an adjacency exists between π_i and π_{i+1} is $\frac{1}{n-1}$, note that out of n−1 remaining symbols only π_{i+1} is favorable for position i+1. Thus, the probability that there is an adjacency between π_i and π_{i+1} is = $\frac{n-1}{n} * \frac{1}{n-1} = \frac{1}{n}$. Thus, the probability that there is no adjacency between π_i and π_{i+1} is 1− $\frac{1}{n}$. So, the probability that there is no adjacency between π_i and π_{i+1} for $1\leq i\leq n-1$, that we call $p_1(0)=(1-\frac{1}{n})^{\,n-1} = \frac{(1-\frac{1}{n})^n}{(1-\frac{1}{n})}$. For large values of n, $p_1(0) \approxeq \frac{1}{e*(1-\frac{1}{n})}$. Thus, $\frac{1}{e}$ is a strict lower bound.

Consider the second type of adjacency. Similar to the first type of adjacency, the probability that there is no adjacency between π_i and π_{i+1} for $1\leq i\leq n-1=(1-\frac{1}{n})^{n-1}$. Additionally , the probability that $\pi_n \neq$n−1 is $(1-\frac{1}{n})$. So, $p_2(0)$, the probability that no adjacency exists is $(1-\frac{1}{n})^n$. For large values of n, $p_2(0) \approxeq \frac{1}{e}$. Likewise, $p_3(0) \approxeq \frac{1}{e}$. Note that instead of $\pi_n \neq$n−1 here we require that $\pi_1 \neq 0$ with the identical probability. Thus, $p_2(0)\leq \frac{1}{e}$ and $p_3(0)\leq\frac{1}{e}$.

Consider the fourth type of adjacency. In addition to the restrictions of second type of adjacency we require that $\pi_1 \neq 0$. This yields $p_4(0)=(1-\frac{1}{n})^{(n+1)}\approxeq \frac{(1-\frac{1}{n})}{e}$. Thus, $p_4(0)< \frac{1}{e}$. ■

A similar result was given by Whitworth in [28]. Whitworth evaluated the number of permutations without any adjacencies(of type 2 or type 3) as $n!(e_n^{-1})$ [28]. Where e_n^{-1} denotes the summation of the first n+1 terms in the series expansion of e^{-1}.

The following observation directly follows from Observation 1. It states that the computation of the expected number of moves for $P_n(0)$ is the bottleneck in the computation of the same for P_n, where $|P_n(0)|$ is equal to or approximately equal to $|P_n|/e$ for large values of n.

Observation 2 Let the number of moves of a particular block-move to optimally sort any permutation in P_i $\forall$ $i \leq n-1$ be known. For computing the expected number of moves to sort P_n with the same operation, one needs to compute the optimum moves (instead of looking up the answer) for approximately $\frac{n!}{e}$ permutations. ■

Note that even for bf-adjacency $\forall_n \geq 20$ the lower bound on $\frac{P_n}{n!} \geq 0.34056$ and when $\forall_n \geq 50$ we have $\frac{P_n}{n!} \geq 0.35688$. For larger values of n this fraction approaches (but never equals), $\frac{1}{e} = 0.36787\ldots$.

5 Prefix Transpositions

A transposition exchanges adjacent sublists; when one of the sublists is restricted to be a prefix then one obtains a prefix transposition. Given a permutation π that must be sorted, if $\pi_n = n-1$ then in an optimal sorting sequence π_n need not be moved again. Likewise, if $\pi_i = i-1$ $\forall_{(i=n...k)}$ then the last n-k+1 elements need not be moved again. This follows from a result in sorting transpositions by Christie [11]. Thus, type 2 adjacencies capture the adjacencies created by prefix transpositions. Likewise, type 3 adjacencies capture the adjacencies created by suffix transpositions and type 4 adjacencies capture the adjacencies created by transpositions.

A prefix transposition can either create or destroy zero, one or two adjacencies. The moves that create one, two or three adjacencies are called a single, a double and a triple respectively. A move can also break adjacencies, however, because the permutation that is being sorted is reduced no adjacencies exist, so, we do not consider such moves. The following two observations are well known.

Observation 3 If $\pi \in P_n$ (n>1) is reduced then a single can always be executed.

Proof: Let π= π_1, π_2, π_3,....., π_n.
If $\pi_1 = 0$ then moving π_1 to just before 1 creates a new adjacency. If $\pi_1 = n-1$ then moving π_1 to just after n-2 creates a new adjacency. If $\pi_1 \neq 0$ and $\pi_1 \neq n-1$ then moving π_1 to just before $\pi_1 + 1$ or just after $\pi_1 - 1$ creates a new adjacency. Note that these moves are both transpositions and prefix transpositions. ■

Observation 4 Let π= π_1, ..., $\pi_i = \pi_1 - 1$, $\pi_{i+1} = a$,..., π_n. A double with prefix transposition is possible iff a-1$\in$ $[\pi_1 ... \pi_{i-1})$.

Proof : ($\rightarrow$) Clearly(π_0, π_1, ..., a-1),.....,$\pi_i = \pi_{1-1}, *\pi_{i+1} = a, ...\pi_{n-1}$is a double.
($\leftarrow$)To execute a double we must create an adjacency with the left end of the moved prefix $[\pi_0, ...x]$ i.e. π_0. Therefore the prefix is moved to a position just after $\pi_0 - 1$. In order to create an adjacency at the right end of the moved prefix, $x = \pi_{i+1} - 1$, i.e. $x = a - 1$. Further, x is to the left of π_i therefore $a - 1 \in (\pi_1 .... \pi_{i-1})$ ■
A permutation that has no adjacencies is said to be irreducible. Likewise, if π can be reduced

then the length of the resulting permutation π^* that cannot be reduced any further is called the reduced length of π. For n≥K, $P_n(k)$ denotes the subset of P_n where the reduced length of any $\pi \in P_n(k)$ is K.
The following theorem evaluates the expected probability that a $\pi \in P_n(n)$ admits a double. In a reduced permutation also a given symbol is equally likely in all positions. For n=3, i.e. for the set {(0 2 1), (1 0 2),(2 1 0)} this assumption holds. For n=4, i.e. for the set { (0 2 1 3), (0 2 3 1), (0 3 2 1), (1 0 3 2), (1 3 0 2), (1 3 2 0), (2 0 3 1), (2 1 0 3), (2 1 3 0), (3 0 2 1), (3 1 0 2), (3 2 1 0)} this holds.

Theorem 5 The expected probability of π= (π_1=f, π_2, ...π_i=f-1, $\pi_{i+1} = a$,..., π_n) $\in P_n(0)$ admitting a double is $\sigma = \frac{1}{2} - \frac{2}{n(n-1)}$.

Proof : General form of π is $f, \ldots, f-1, a \ldots$ where $f = \pi_1$ and a is the element succeeding $f-1$. Let p be the probability of a double. Any permutation where $f = \pi_1$ and $f-1 \in$ an arbitrary position in $[2, ..., n]$ occurs with a probability of $\frac{1}{n(n-1)}$. Here $f = \pi_1$ with a probability of $\frac{1}{n}$ and $f-1$ can be in any of the positions $[2, ..., n]$ with an equal probability of $\frac{1}{n-1}$. We analyze the cases where $f = 0, \ldots, n-1$. In each case the position of $f-1$ forms a sub-case. Here we denote the probability of a double where $f = k$ (or $f \in S$, a set of symbols) as $\sigma(k)$ (resp. σ(S)). Likewise, $\sigma(k, j)$ ($\sigma(S, j)$) is the probability of a double where $f = k$ (resp. $\in S$) and the position of $f-1$ is j.
Case(i): $f = 0$. $\sigma(0)$=0. Here the moved prefix cannot create a new adjacency at the left end. Thus, a double is not possible.
Case(ii): $f = n-1$. σ(n-1) $=\frac{(n-3)}{2n(n-1)} + \frac{1}{n(n-1)}$.
The following subcases are partitioned as per the index of n-2, i.e. π^{-1}_{n-2}. Note that π_{n-2} is the element at the position n-2. Thus, the total probability of $f, f-1$ and a being in their respective positions is $\mu = \frac{1}{n(n-1)(n-2)}$

- $\sigma(n-1, 2) = 0$. The moved prefix i.e. n-1 cannot create an adjacency at the right end. That is $(n-1)+1 \notin \Sigma$.

- $\sigma(n-1, 3) = \frac{\mu}{n-3}$. Here one can execute a double if $\pi_2 = a-1$. Here for $a = 1, \ldots, n-3$, $\pi_2 = a-1$ with a probability of $\frac{1}{n-3}$. Note that out of n-3 positions only one position is favorable.

- $\sigma(n-1, 4) = \frac{2\mu}{n-3}$. Here one can execute a double if $\pi_2 = a-1$ or $\pi_3 = a-1$. Here for $a = 1, \ldots, n-3$, $\pi_2 =$ a-1 with a probability of $\frac{2}{n-3}$. Thus out of n-3 positions only two position are favorable.

- $\sigma(n-1, n-1) = \frac{(n-3)\mu}{n-3}$. Here one can execute a double if $\pi_2 = a-1$ or $\pi_3 = a-1 \ldots \pi_{n-2} = a-1$. Here for $a = 1, \ldots, n-3$, one of π_2 ,...$\pi_{n-2} = a-1$ with a probability of $\frac{n-3}{n-3}$. Thus the total probability for a particular value of a is the summation of the above probabilities. That is, $0 + \frac{1}{n(n-1)(n-2)}$ ($\frac{1}{(n-3)}$ +$\frac{2}{(n-3)}$ +$\frac{3}{(n-3)}$ +.......+ $\frac{n-3}{(n-3)}$)= $\frac{1}{2n(n-1)}$

- Here a can take values from 1, 2, 3,......, n-3 (a cannot be f or f-1 or 0. a=0 is not possible since $a-1 \notin \Sigma$). Thus, $n-3$ values are possible for a. So, σ(n-1,[2 ... n-1]) $=\frac{(n-3)}{2n(n-1)}$.

- Type 2 adjacency includes an adjacency between $n-1$ in the last position and (imagined) $\pi_{n+1}=n$. Thus, if $f-1$ occurs in the last position then one can execute $[f,\ldots n-1]\ldots f-1*$ and create two new adjacencies, one between f-1 and f and another one between n-1 and imagined n. Thus, σ(n-1,n) $=\frac{1}{n(n-1)}$.

Case(iii): $f \neq$n-1 and $f \neq$0; i.e. $f \in S$ where $S=\{1,2,3,....,n-2\}$. $\sigma(S)=\frac{1}{2}-\frac{3}{2(n-1)}-\frac{1}{2(n-1)(n-2)}-\frac{1}{n(n-1)(n-2)}+\frac{n-2}{n(n-1)}$.

This case is partitioned into two sub cases. Case (iii-a) $f=1$. Case (iii-b) $f>1$. Similar to Case(ii) the probability of f and $f-1$ to be positioned in their respective positions equals $\frac{1}{n(n-1)}$. Also, a occurs in its position with $\frac{1}{(n-2)}$ probability.

Case (iii-a): Here $f=1$. $\sigma(1)=\frac{(n-3)}{2(n-1)(n-2)}-\frac{1}{n(n-1)(n-2)}+\frac{1}{n(n-1)}$.

Consider $f=1, f-1=0=\pi_2$ and $a=f+1=2$. Here one can move f in between $f-1$ and a. Further, the same move works if $f-1$ is any of the positions $3\ldots n-1$). Given these configurations the probability of a double is 1. So, the probability when $a=f+1$ is $\frac{(n-2)}{n(n-1)(n-2)}=\frac{1}{n(n-1)}$. (A)

If $0=\pi_3(\pi_k)$ then for a double $a-1=\pi_2(\pi_{<k})$. So, for a particular value of a the probability is $\frac{1}{n(n-1)(n-2)}$ ($\frac{1}{(n-3)}+\frac{2}{(n-3)}+\frac{3}{(n-3)}+.......+=\frac{n-3}{(n-3)}$)= $\frac{1}{2n(n-1)}$

Here $a \in [4,5,\ldots,n-1]$ ($a \notin \{0,1,f+1=2,f+2=3\}$; $f+1=2$ is analyzed above and $f+2=3$ is analyzed below. So for all the $n-4$ values the combined probability is $\frac{(n-4)}{2n(n-1)}$. (B)

When $a=f+2=3$ and $f-1=0=\pi_3$ a double is infeasible because $\pi=$ (1 2 0 3.....) is disallowed due to the presence of the adjacency **1 2**. Recall that we only consider the reduced permutations. When $f-1$ comes in the fourth position then for a double to be feasible $a-1=2$ can be in the third position. So, given f and $f-1$, the probability of a double is$\frac{1}{n-3}$. Similarly the probabilities for the other positions of $f-1$ are deduced. Recall that $a-1 \neq \pi_2$. So, the total probability for this subcase is

$\frac{1}{n(n-1)(n-2)}$ ($\frac{1}{(n-3)}+\frac{2}{(n-3)}+\frac{3}{(n-3)}+.......+\frac{n-4}{(n-3)}$)= $\frac{(n-4)}{2n(n-1)(n-2)}$ (C).

Thus, the total probability of case(iii-a) $\sigma(1)=A+B+C-K$ where K is the probability corresponding to $\pi_n=a=n-1$. Note that in Type 2 adjacency $\pi_n=n-1$ creates an adjacency and we assume that the permutation is reduced, so, this scenario is not possible. Here, $K=\frac{1}{n(n-1)(n-2)}$. The total probability of case(iii-a) equals $\sigma(1)=A+B+C-K=$

$$\frac{1}{n(n-1)}+\frac{(n-4)}{2n(n-1)}+\frac{(n-4)}{2n(n-1)(n-2)}-\frac{1}{n(n-1)(n-2)}=\frac{2(n-2)+(n-4)(n-2)+(n-4)}{2n(n-1)(n-2)}-\frac{1}{n(n-1)(n-2)}$$

$$=\frac{2n-4+n^2-6n+8+(n-4)}{2n(n-1)(n-2)}-\frac{1}{n(n-1)(n-2)}=\frac{(n-3)}{2(n-1)(n-2)}-\frac{1}{n(n-1)(n-2)}$$

- Type 2 adjacency includes an adjacency between $\pi_n=n-1$ and imagined $\pi_{n+1}=n$. Thus, if $f-1$ occurs in the last position then one can execute $[f,\ldots n-1]\ldots f-1*]$ and create two new adjacencies, one between $f-1$ and f and another one between $n-1$ and imagined n in position number $n+1$. Thus, σ(1,n) $=\frac{1}{n(n-1)}$. When this probability is also included then the total probability for Case (iii-a) is: $\frac{(n-3)}{2(n-1)(n-2)}$ - $\frac{1}{n(n-1)(n-2)}$ + $\frac{1}{n(n-1)}$

Case(iii-b): $f > 1$ i.e. $S = \{2, 3, ..., n-2\}$. σ(S) $=\frac{(n-3)(n^2-4n)}{2n(n-1)(n-2)} + \frac{n-3}{n(n-1)}$.

The probability that $f = \pi_1$ and $f-1 = \pi_j$ (for $j \neq 1$) is $\frac{1}{n(n-1)}$. The probability that $a = \pi_{j+1}$ is $\frac{1}{n-2}$. When $f-1$ immediately succeeds f then a must be $f+1$. When $a = f+1$, $f-1$ can be in any of the positions $[2 \ldots n-1]$. Given this configuration the probability of a double is 1. So the total probability corresponding to $f-1$ occupying all possible positions is $\frac{(n-2)}{n(n-1)(n-2)} = \frac{1}{n(n-1)}$. (P)

If $f-1 = \pi_3$ then for a double to be feasible $a-1 = \pi_2$. Given that $f-1 = \pi_3$, $a-1 = \pi_2$ with a probability of $\frac{1}{n-3}$. The denominator is $n-3$ because we exclude $f, f-1$ and a. Similarly, for $f-1 = (\pi_4 \ldots \pi_{n-1})$ we obtain the respective probabilities as ($\frac{2}{n-3}$,$\frac{3}{n-3}$, ...$\frac{(n-3)}{(n-3)}$). Thus, for particular values of f and a the probability of a double in this case is = $\frac{1}{n(n-1)(n-2)}$ ($\frac{1}{(n-3)}$ +$\frac{2}{(n-3)}$ +$\frac{3}{(n-3)}$ +.......+ $\frac{n-3}{(n-3)}$)= $\frac{1}{2n(n-1)}$.

Here a can take $n-5$ values which exclude $0, f, f-1, a = f+1$ and $a = f+2$ where $a = f+1$ is considered above and $a = f+2$ is analyzed below. Thus, the probability for all values of a is $\frac{(n-5)}{2n(n-1)}$. (Q)

When $a = f+2$ and $f-1 = \pi_3$ then double is infeasible because $\pi = (f, f+1, f-1a = f+2, \ldots)$ is not reduced. If $f-1$ is in the fourth position then for a double to occur $a-1$ can be in the third position only. So the probability of $a-1$ occurring between f and $f-1$ is $\frac{1}{(n-3)}$ (we exclude $f, f-1$ and a). Similarly, the probabilities of a double are derived when $f-1$ occurs in other positions. If $f-1$ is in the$(n-1)^{th}$, position then for a double to be feasible $a-1$ can reside in any of the $(n-4)$ positions. The probability of $a-1$ residing between f and $f-1$ is $\frac{(n-4)}{(n-3)}$. So, the total probability for this case = $\frac{1}{n(n-1)(n-2)}$ ($\frac{1}{(n-3)}$ +$\frac{2}{(n-3)}$ +$\frac{3}{(n-3)}$ +.......+ $\frac{n-4}{(n-3)}$) = $\frac{(n-4)}{2n(n-1)(n-2)}$ (R)

Recall that K is the probability corresponding to $\pi_n = a = n-1$. As the permutation is reduced this scenario is avoided. Here f can assume $n-3$ values that remain after excluding $0, n-1$, and 1. Total probability for a particular value of f in Case(iii-b) equals $P+Q+R-K =$

$\frac{1}{n(n-1)}+\frac{(n-5)}{2n(n-1)}+\frac{(n-4)}{2n(n-1)(n-2)} - K=$

$\frac{2(n-2)+(n-5)(n-2)+(n-4)}{2n(n-1)(n-2)} - \frac{1}{n(n-1)(n-2)}=$

$\frac{2n-4+n^2-7n+10+(n-4)}{2n(n-1)(n-2)} - \frac{1}{n(n-1)(n-2)}=\frac{n^2-4n+2-2}{2(n-1)(n-2)} = \frac{n^2-4n}{2(n-1)(n-2)}$.

- Type 2 adjacency includes an adjacency between $\pi_n = n-1$ and the imagined $\pi_{n+1} = n$.

Thus, if $f-1$ occurs in the last position then one can execute $[f,\ldots n-1]\ldots f-1*$ and create two new adjacencies, one between $f-1$ and f and another one between $n-1$ and imagined n. There are $n-3$ choices for f, so, $\sigma([2\ldots n-2],n) = \frac{n-3}{n(n-1)}$.

So, the total probability for all values of f for Case(iii-b) $=\frac{(n-3)(n^2-4n)}{2n(n-1)(n-2)}+ \frac{n-3}{n(n-1)}$. The total probability for Case(iii) that is partitioned into Case(iii-a) and Case(iii-b) is therefore

$$\frac{(n-3)}{2(n-1)(n-2)} - \frac{1}{n(n-1)(n-2)} + \frac{(n-3)(n^2-4n)}{2n(n-1)(n-2)} + \frac{n-3+1}{n(n-1)}$$

$$= \frac{(n-3)}{2n(n-1)(n-2)}(n^2-4n+n) - K + \frac{n-3+1}{n(n-1)}$$

$$=\frac{(n-3)}{2n(n-1)(n-2)}(n^2-3n) - K + \frac{n-3+1}{n(n-1)} =$$

$$\frac{(n-3)(n-3)}{2(n-1)(n-2)} - \frac{1}{n(n-1)(n-2)} + \frac{n-3+1}{n(n-1)} =$$

$\frac{1}{2} - \frac{3}{2(n-1)} + \frac{1}{2(n-2)(n-1)} - \frac{1}{n(n-1)(n-2)} + \frac{n-3+1}{n(n-1)}$. We add the probability of case(ii) to this expression to obtain the final probability as:
$\frac{1}{2}- \frac{2}{n(n-1)}$.

Thus, the total probability for all cases that is $\sigma(\Sigma) = \frac{1}{2}- \frac{2}{n(n-1)}$. ■

Observation 5 Let $\pi \in \mathrm{P}_n$ be reduced then the expected number of new adjacencies created per move is 1+σ.

Proof : A double can be executed with a probability of σ and it creates two new adjacencies and a single can be executed with a probability of (1-σ) and it creates one new adjacency. Thus, in a move the expected number of new adjacencies created= $\sigma(2)$ + (1-σ)(1)= 1+σ. Note that from Theorem 5, for prefix transpositions, this measure ≈ 1.5 for large values of n. ■

6 Results and Conclusions

The average number of moves to sort all permutations in $\mathrm{P}_n(0)$ for $n \leqslant 9$ and the average number of moves to sort all permutations in P_n for $1 < n \leqslant 9$ are computed and shown in Tables 3 and 4. When n=0, no moves are needed. In each of these table the computed values of $\mathrm{P}_j(0)$, $j \leqslant i$ are used as a basis to predict the rest of the values. The predicted values can be computed for large values of n because given the computed values of $\mathrm{P}_j(0)$, $j \leqslant i$ the model runs in $\mathrm{O}(n^2)$. The predicted values can be compared to the computed values for $n \leqslant 9$. Currently, for $n > 9$ the computed values are not applicable, denoted by: na. For n = $2\ldots i$ the values are computed thus the predicted values are shown as dashes. The model gets more accurate for larger values of n.

We compute $\forall k|P_n(k)|$ in O(n^2) time. We show that the number of irreducible permutations in P_n is $\Theta(n!)$. The computation of $\forall k|P_n(k)|$ leads to a framework for analysis of block-moves on permutations. Bulteau et al. show that sorting permutations by transpositions is NP-hard

[3]. Thus, it is desirable to compute the expected number of moves to sort a permutation $\in P_n$ with transpositions. It is believed that sorting permutations by either prefix transpositions or suffix transpositions or prefix/suffix transpositions is also NP-hard. So, the computation of say $E(P_n)$, the expected number of moves to sort permutations, with various block-moves is of interest. $E(P_n)$ for a particular operation indicates the expected number of moves a packet from some source node u to some destination node v must traverse in the corresponding Cayley graph.

The main contribution of this article is the theoretical framework for estimating the expected number of moves, i.e. $E(P_n)$ to sort permutations in P_n with various block-moves in $\mathrm{O}(n^2)$ time; given the computation of $E(P_i)$ for some $i < n$. We employ a model based on the proposed framework to estimate $E(P_n)$ in $\mathrm{O}(n^2)$ time for prefix transpositions. Due to symmetry, the corresponding results are applicable for suffix transpositions also. Based on this model, Pai and Kumarasamy worked on estimating $E(P_n)$ with transpositions [22]. Current work is focused on exploring models for estimation.

References

[1] S. B. Akers and B. Krishnamurthy. A group-theoretic model for symmetric interconnection networks. *IEEE Trans. Computers*, 38(4):555–566, 1989.

[2] V. Bafna and P. Pevzner. Sorting permutations by tanspositions. *SIAM Journal of Discrete Mathematics*, 11(2):224 – 240, 1998.

[3] L. Bulteau, G. Fertin, and I. Rusu. Sorting by transpositions is difficult. *SIAM Journal on Discrete Mathematics*, 26(3):1148–1180, 2012.

[4] B. Chitturi. *On Transforming Sequences.* The University of Texas at Dallas, 2007.

[5] B. Chitturi. Upper bounds for sorting permutations with a transposition tree. *Discrete Math., Alg. and Appl.*, 5(1), 2013.

[6] B. Chitturi. Tighter upper bound for sorting permutations with prefix transpositions. *Theor. Comput. Sci.*, 602:22–31, 2015.

[7] B. Chitturi, W. Fahle, Z. Meng, L. Morales, C. O. S. Jr., I. H. Sudborough, and W. Voit. An (18/11)n upper bound for sorting by prefix reversals. *Theor. Comput. Sci.*, 410(36):3372–3390, 2009.

[8] B. Chitturi and I. H. Sudborough. Bounding prefix transposition distance for strings and permutations. In *41st Hawaii International International Conference on Systems Science (HICSS-41 2008), Proceedings, 7-10 January 2008, Waikoloa, Big Island, HI, USA*, page 468, 2008.

[9] B. Chitturi and I. H. Sudborough. Bounding prefix transposition distance for strings and permutations. *Theoretical Computer Science*, 421(0):15 – 24, 2012.

[10] B. Chitturi, I. H. Sudborough, W. Voit, and X. Feng. Adjacent swaps on strings. In *Computing and Combinatorics, 14th Annual International Conference, COCOON 2008, Dalian, China, June 27-29, 2008, Proceedings*, pages 299–308, 2008.

[11] D. Christie. *Genome Rearrangement Problems.* University of Glasgow, 1998.

[12] D. A. Christie and R. W. Irving. Sorting strings by reversals and by transpositions. *SIAM J. Discrete Math.*, 14(2):193–206, 2001.

[13] J. Cibulka. On average and highest number of flips in pancake sorting. *Theoretical Computer Science*, 412(8-10):822 – 834, 2011.

[14] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to Algorithms*. PHI Learning, 2010.

[15] Z. Dias and J. Meidanis. Sorting by prefix transpositions. In A. Laender and A. Oliveira, editors, *String Processing and Information Retrieval*, volume 2476 of *Lecture Notes in Computer Science*, pages 65–76. Springer Berlin Heidelberg, 2002.

[16] H. Eriksson, K. Eriksson, J. Karlander, L. J. Svensson, and J. W. ästlund. Sorting a bridge hand. *Discrete Mathematics*, 241(1-3):289–300, 2001.

[17] L. Euler. *Methodus universalis series summandi ulterius promota.* Commentarii academiae scientiarum Petropolitanae, 2nd edition, (1736):147-158.

[18] X. Feng, B. Chitturi, and H. Sudborough. Sorting circular permutations by bounded transpositions. In H. R. Arabnia, editor, *Advances in Computational Biology*, volume 680 of *Advances in Experimental Medicine and Biology*, pages 725–736. Springer New York, 2010.

[19] R. L. Graham, D. E. Knuth, and O. Patashnik. *Concrete Mathematics: A Foundation for Computer Science*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2nd edition, 1994.

[20] A. Labarre. Lower bounding edit distances between permutations. *SIAM Journal on Discrete Mathematics*, 27(3):1410–1428, 2013.

[21] S. Lakshmivarahan, J. Jwo, and S. K. Dhall. Symmetry in interconnection networks based on cayley graphs of permutation groups: A survey. *Parallel Computing*, 19(4):361–407, 1993.

[22] S. Pai and S. Kumarasamy. *Towards estimating the expected number of transpositions required to sort a permutation in P_n*. Amrita Vishwa Vidyapeetham, Amritapuri, Kollam, India, 2015.

[23] D. P. Roselle. Permutations by number of rises and successions.

[24] L. Shan-Lan. *Sums of Piles Obtained Inductively,Classically Inspired Meditations in Mathematics.* 1867.

[25] N. J. A. Sloane. *https://oeis.org/.*

[26] J. Stirling. *The Differential Method, Original 1730, English translation.* 1749.

[27] S. M. Tanny. Permutations and successions. *Journal of Combinatorial Theory, Series A*, 21:196–202, 1976.

[28] W. A. Whitworth. *Choice and Chance.* Stechert, New York, 1943.

Table 0. Values of $| P_n(k) |$ for type 1 adjacency.

n\k	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	$\Sigma \mid P_n(k) \mid$
2	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	2
3	3	2	1	0	0	0	0	0	0	0	0	0	0	0	0	6
4	11	9	3	1	0	0	0	0	0	0	0	0	0	0	0	24
5	53	44	18	4	1	0	0	0	0	0	0	0	0	0	0	120
6	309	265	110	30	5	1	0	0	0	0	0	0	0	0	0	720
7	2119	1854	795	220	45	6	1	0	0	0	0	0	0	0	0	5040
8	16687	14833	6489	1855	385	63	7	1	0	0	0	0	0	0	0	40320
9	148329	133496	59332	17304	3710	616	84	8	1	0	0	0	0	0	0	362880
10	1468457	1334961	600732	177996	38934	6678	924	108	9	1	0	0	0	0	0	3628800
11	16019531	14684570	6674805	2002440	444990	77868	11130	1320	135	10	1	0	0	0	0	39916800
12	190899411	176214841	80765135	24474285	5506710	978978	142758	17490	1815	165	11	1	0	0	0	479001600
13	2467007773	2290792932	1057289046	323060540	73422855	13216104	1957956	244728	26235	2420	198	12	1	0	0	6227020800
14	34361893981	32071101049	14890154058	4581585866	1049946755	190899423	28634892	3636204	397683	37895	3146	234	13	1	0	87178291200

7 Acknowledgments

This article studies adjacencies in permutations as defined in [8]. It is based on the tech. report UTDCS-03-15, dept. of CS, Univ. of Texas at Dallas. The articles [23, 28] and the existence of integer sequences equivalent to the ones studied in the current article came to our attention due to an anonymous referee.

Table 1. Values of $\mid P_n(k) \mid$ for the b-adjacency. These numbers hold for f-adjacency as well.

n\k	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	$\Sigma \mid P_n(k) \mid$
2	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	2
3	2	3	0	1	0	0	0	0	0	0	0	0	0	0	0	6
4	9	8	6	0	1	0	0	0	0	0	0	0	0	0	0	24
5	44	45	20	10	0	1	0	0	0	0	0	0	0	0	0	120
6	265	264	135	40	15	0	1	0	0	0	0	0	0	0	0	720
7	1854	1855	924	315	70	21	0	1	0	0	0	0	0	0	0	5040
8	14833	14832	7420	2464	630	112	28	0	1	0	0	0	0	0	0	40320
9	133496	133497	66744	22260	5544	1134	168	36	0	1	0	0	0	0	0	362880
10	1334961	1334960	667485	222480	55650	11088	1890	240	45	0	1	0	0	0	0	3628800
11	14684570	14684571	7342280	2447445	611820	122430	20328	2970	330	55	0	1	0	0	0	39916800
12	176214841	176214840	88107426	29369120	7342335	1468368	244860	34848	4455	440	66	0	1	0	0	479001600
13	2290792932	2290792933	1145396460	381798846	95449640	19090071	3181464	454740	56628	6435	572	78	0	1	0	6227020800
14	32071101049	32071101048	16035550531	5345183480	1336295961	267258992	44543499	6362928	795795	88088	9009	728	91	0	1	87178291200

Table 2. Values of $\mid P_n(k) \mid$ for the bf-adjacency.

n\k	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	$\Sigma \mid P_n(k) \mid$
2	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	2
3	1	4	0	0	1	0	0	0	0	0	0	0	0	0	0	0	6
4	8	5	10	0	0	1	0	0	0	0	0	0	0	0	0	0	24
5	36	48	15	20	0	0	1	0	0	0	0	0	0	0	0	0	120
6	229	252	168	35	35	0	0	1	0	0	0	0	0	0	0	0	720
7	1625	1832	1008	448	70	56	0	0	1	0	0	0	0	0	0	0	5040
8	13208	14625	8244	3024	1008	126	84	0	0	1	0	0	0	0	0	0	40320
9	120288	132080	73125	27480	7560	2016	210	120	0	0	1	0	0	0	0	0	362880
10	1214673	1323168	726440	268125	75570	16632	3696	330	165	0	0	1	0	0	0	0	3628800
11	13469897	14576076	7939008	2905760	804375	181368	33264	6336	495	220	0	0	1	0	0	0	39916800
12	162744944	175108661	94744494	34402368	9443720	2091375	392964	61776	10296	715	286	0	0	1	0	0	479001600
13	2128047988	2278429216	1225760627	442140972	120408288	26442416	4879875	785928	108108	16016	1001	364	0	0	1	0	6227020800
14	29943053061	31920719820	17088219120	6128803135	1658028645	361224864	66106040	10456875	1473615	180180	24024	1365	455	0	0	1	87178291200

Table 3. Computed and Predicted values of $P_n(0)$. Initialization: $P_1(0)...P_i(0)$.

n	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Computed	1.0	2.0	2.33	3.09	3.68	4.29	4.91	5.50	na	na	na	na	na	na	na
Pred. i=6	-	-	-	-	-	4.21	4.81	5.43	6.07	6.71	7.37	8.02	8.69	9.35	10.01
Pred. i=7	-	-	-	-	-	-	4.83	5.46	6.08	6.72	7.37	8.03	8.69	9.35	10.01
Pred. i=8	-	-	-	-	-	-	-	5.47	6.10	6.73	7.38	8,03	8.69	9.35	10.01

Table 4. Computed and Predicted values of $E(X_n)$. Initialization: $P_1(0)...P_i(0)$

n	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Computed	0.5	1.16	1.79	2.42	3.06	3.68	4.29	4.90	na	na	na	na	na	na	na
Pred. i=6	-	-	-	-	-	3.65	4.23	4.82	5.44	6.07	6.72	7.37	8.03	8.69	9.35
Pred. i=7	-	-	-	-	-	-	4.26	4.86	5.46	6.09	6.73	7.38	8.03	8.69	9.35
Pred. i=8	-	-	-	-	-	-	-	4.89	5.50	6.11	6.74	7.38	8.03	8.69	9.35

8 Appendix

The proofs for Theorem 1-3 are deferred to the appendix. This is intended to improve the readability of the main text. The proof of Theorem 2 is similar to that of Theorem 3 and hence it is omitted.

Theorem 1 Let $P_n(k)$ be the subset of P_n where any $\pi \in P_n(k)$ has exactly k type 1 adjacencies. Let f(n,k) be the cardinality of $P_n(k)$. Then f(n,k) = f(n−1,k−1)+ (n−1−k) * f(n−1,k) + (k+1)* f(n−1,k+1) where $0 \leq k < n$.

Proof We denote the number of adjacencies in a permutation $\pi \in P_n$ with $\alpha(\pi)$ and the number of permutations $\in P_n$ having $\alpha(\pi)$ adjacencies with f(n, $\alpha(\pi)$). Recall that $\Sigma = \{0,....,n-1\}$. Thus, $\pi^* \in P_{n-1}$ is composed of $\{0,....,n-2\}$and a member of P_n additionally contains n−1. Let $\alpha(\pi^*)$=q for $\pi^* \in P_{n-1}$. When a $\pi \in P_n$ is formed from π^* by inserting n−1 we have the following three cases: (i) $\alpha(\pi)=\alpha(\pi^*)$, (ii) $\alpha(\pi)=\alpha(\pi^*)+1$ and (iii) $\alpha(\pi)=\alpha(\pi^*)-1$.

The element n−1 can create an adjacency only if it immediately succeeds n−2; however in such a case it cannot destroy an existing adjacency; i.e. $\alpha(\pi)=\alpha(\pi^*)+1$ (Case (ii)). Thus, it is not possible to insert n−1 in any position where it simultaneously creates one adjacency and destroys one adjacency. Similarly, if n−1 is inserted between a, a+1 for some a then $\alpha(\pi)=\alpha(\pi^*)-1$ (Case (iii)). Finally, if n−1 neither succeeds n−2 nor splits a, a+1 (for some a) then $\alpha(\pi)=\alpha(\pi^*)$ (Case (i)). We determine the number of permutations π of P_n with $\alpha(\pi)$=k that can be generated from some permutations in P_{n-1} corresponding to each of these cases.

Let (π^*)= k. For a given π^* we want to determine how many $\pi \in P_n$ exist such that $\alpha(\pi^*)=\alpha(\pi)$. In order to generate $\pi \in P_n$ from $\pi^* \in P_{n-1}$ one can insert n−1 in any of the n positions (n−2 internal and two extreme positions). However, in order to maintain the same number of adjacencies, k+1 positions are forbidden; where k positions correspond to existing adjacencies and one corresponds to the adjacency that can be created between n−1 and n. Thus, the contribution of f(n−1,k) to f(n,k) is (n−1−k) * f(n−1,k). Let $\alpha(\pi^*)$= k−1, we want to determine the contribution of f(n−1,k) to f(n,k). In order to create a $\pi \in P_n$ from π^* where $\alpha(\pi)$= k the only possibility is that n−1 is inserted to the immediate right of n−2. Thus, f(n−1,k) contributes to f(n,k) exactly f(n−1,k). Finally, we want to determine the contribution of f(n−1,k+1) to f(n,k). Here, any one of the k+1 adjacencies can be broken by inserting n+1 in between. Thus, the contribution of f(n−1,k+1) to f(n,k) is (k+1)* f(n−1,k+1). Note, that f(n,k) is restricted to the above cases. The corresponding algorithm, Adjacency_Count, is given below. The theorem follows. ■

```
Algorithm: Adjacency_Count
Initialization: f(2,0)=1;f(2,1)=1;∀_i f(i,i−1)=1
for (i=3,....n) do
    for (k=0,....i−2) do
        if (k==0)  thenf(i,k) = (i−1) * f(i−1,0) + f(i−1,1);
        else
            if (k==i−2) then f(i,k) = f(i−1,k−1) + f(i−1,k);
            elsef(i,k) = f(i−1,k−1)+ (i−k−1) * f(i−1,k) + (k+1)* f(i−1,k+1);
            end if
        end if
    end for
```

end for

Theorem 3 is restated here. The proof accompanies.

Theorem 3 Let f(i, j) denote the number of permutations in P_i with exactly j adjacencies. Then the recurrence relation for f(i, j) is:
f(i, j)= (f(i−1, j)−f(i−2, j−1))*(i−j−2) +
(f(i−1, j−1)−f(i−2, j−2))*2 +f(i−2, j−2) +
(f(i−1, j+1)−f(i−2, j))*(j+1)+f(i−2, j)*(i−j−1)+
f(i−2, j+1)*(j+1); 0≤ j≤i+1.

Proof We presume that π_{n+1} =n and $\pi_0 = -1$. So, if π_n =n−1 then π_n and π_{n+1} form an adjacency. Likewise, if $\pi_1 = 0$ then π_0 and π_1 form an adjacency. We denote the number of adjacencies in $\pi\in P_n$ with $\alpha(\pi)$ and the number of permutations$\in P_n$ having $\alpha(\pi)$ adjacencies with f(n, $\alpha(\pi)$). Recall that Σ ={0, ..., n−1}. Thus, $\pi^*\in P_{n-1}$ is composed of {0, ..., n−2} and $\pi\in P_n$ additionally contains n - 1. The formation of $\pi\in P_n$ from $\pi^* \in P_{n-1}$ by inserting n−1 can be partitioned into: **Case (i)** $\alpha(\pi)$=$\alpha(\pi^*)$, **Case (ii)** $\alpha(\pi)$=$\alpha(\pi^*)$+1, **Case (iii)** $\alpha(\pi)$=$\alpha(\pi^*)$−1 and **Case (iv)** $\alpha(\pi)$=$\alpha(\pi^*)$−2

The element n−1 can create an additional adjacency only if it immediately succeeds n−2 or π_n =n−1. However in such a case it cannot destroy an existing adjacency; i.e. $\alpha(\pi)$=$\alpha(\pi^*)$+1 **(Case (ii))**. Thus, it is not possible to insert n−1 in any position where it simultaneously creates one adjacency and destroys one adjacency. Similarly, if n−1 is inserted between x, x+1 for some x then $\alpha(\pi)$=$\alpha(\pi^*)$−1 **(Case (iii))**. If n−1 neither creates an adjacency nor splits any x, x+1 then $\alpha(\pi)$=$\alpha(\pi^*)$ **(Case (i))**. Consider π^* where π^*_{n-1}= n−2; here π^*_{n-1} and (imagined) π^*_n form an adjacency; thus, if n−1 is inserted into π^* in a position other than n then $\alpha(\pi)$=$\alpha(\pi^*)$−1. Furthermore, if n−1 breaks an existing adjacency in π^* then $\alpha(\pi)$=$\alpha(\pi^*)$−2 **(Case (iv))**.

Given $\pi^* \in P_{n-1}$ where $\alpha(\pi^*) = p$, we determine the number of permutations in P_n that are generated from π^* that have k adjacencies. From the above discussion $p \in$ {k−1, k, k+1, k+2}. First, we observe the following.
Observation: The number of permutations in P_{n-1} with k adjacencies where π^*_{n-1}= n−2 equals f(n−2, k−1). Justification: If we disregard the last element the remaining elements that belong to P_{n-2} need to form k−1 adjacencies. The observation follows.
Case(i): p=k. Here we determine the number of permutations in P_n that π^* generates such that $\alpha($ $\pi^*)$=$\alpha(\pi)$. In order to generate a permutation in P_n from a permutation in P_{n-1} one can insert n−1 in any of the n positions (n−2 internal and two extreme positions). **Case(i-a):** If π^*_{n-1}=n−2 then by inserting n−1 either we increase the number of adjacencies by one if π_n=n−1 or decrease the number of adjacencies by at least one. That is, if n−1 is not placed in the last position then the existing adjacency of the last element of π^*_{n-1} i.e. n−2 is automatically broken because after inserting n−1, in P_n, n− 2 is not the largest element. Further n−1 can break an existing adjacency; thus for this sub-case $\alpha(\pi)\in$ {k+1, k−1, k−2}; that is this sub-case is infeasible. **Case(i-b):** If $\pi^*_{n-1}\neq$ n−2 then if n−1 is inserted in a position where it does not create or break an adjacency then $\alpha(\pi^*)$=$\alpha(\pi)$. There are n−k−2 such, positions, where n denotes the number of positions where n−1 can be inserted, k of the excluded positions correspond to existing adjacencies in π^* and the remaining two excluded positions correspond to π_n and the position immediately following n−2. Note that, f(n−1, k)−f(n−2, k−1) denotes the number of permutations where $\pi^*_{n-1}\neq$ n−2 and $\alpha(\pi^*)$=k . Thus, the contribution of f(n−1, k) to f(n, k) is f(n−1, k)−f(n−2, k−1)*(n−k−2).
Case(ii): p= k−1. If π^*_{n-1} =n−2 then π_n=n−1 is the only possibility the corresponding con-

tribution is f(i−2, j−2). If $\pi^*_{n-1} \neq$ n−2 then π_n=n−1 or n−1 can be inserted immediately after n−2. Thus, the contribution of f(n−1, k) to f(n, k) is(f(n−1, k−1)−f(n−2, k−2))*2+f(n−2, k−2).

Case(iii): p = k+1. If $\pi^*_{n-1} \neq$n−2 any of the existing k+1 adjacencies can be broken. Otherwise, $\pi_n \neq$ n−1 and n−1 does not break any of the existing k adjacencies. Thus, the contribution of f(n-1, k+1) to f(n, k) is
(f(n−1, k+1)−f(n−2, k))*(k+1)+f(n−2, k)*(n−k−1).

Case(iv): p=k+2. The only feasibility is that π^*_{n-1} = n−2 and n−1 breaks one of the existing adjacencies. Thus, the contribution of f(n−1, k+1) to f(n, k) is f(i− 2, j+1)*(j+1). The theorem follows (the proof for Theorem 2 is similar). The corresponding algorithm, **Adjacency_Count2**, is given below. ■

```
Algorithm :Adjacency_Count2
Initialization: f[2][0]= 1; f[2][1]= 0; f[2][2]= 0; f[2][3]= 1; f[3][0]= 1; f[3][1]= 4; f[3][2]= 0; f[3][3]=0; f[3][4]=1;

for (i=4,....n) do
    for (j=0,....i) do
        if (j==0)  then f[i][j] ← (f[i−1][j]) *(i−2) + ( f[i−1][j+1]−f[i−2][j])* (j+1) +
                              f[i−2][j] *(i−1−j)+ f[i−2][j+1] *(j+1);
        else
            if (j==i)  then f[i][j] ←( (f[i−1] [j−1]) − f[i−2][j−2]) ∗ 2+ f[i−2][j−2]
            else
                if (j==1)  then f[i][j] ←( (f[i−1] [j−1])*2
                else  f[i][j] ←( (f[i−1] [j−1]) − f[i−2][j−2]) ∗ 2+ f[i−2][j−2]
                    f[i][j] ← f[i][j]+( (f[i−1] [j+1]) − f[i−2][j]) ∗ (j + 1)+ f[i−2][j ]*(i−j−1) ;
                    if i−j−2 >0 then f[i][j] ← f[i][j]+( (f[i−1] [j]) − f[i−2][j−1] )* (i−j−2)
                        f[i][j] ← f[i][j]+f[i−2] [j+1])*(j+1) \\Break one adjacency from f[i−2][j+1]
                    end if
                end if
            end if
        end if
    end for
end for
```