A variant of the tandem duplication - random loss model of genome rearrangement

Mathilde Bouvel and Dominique Rossin

1 The tandem duplication - random loss model

In the usual models of genome rearrangement, duplications and losses of genes are not taken into account. There were attempts to incorporate them to the classical models, but the consecutive combinatorial complexity of the models so obtained made their study quite difficult. Following [3], we focus on the duplication-loss problem by considering the *tandem duplication - random loss model* of genome rearrangement in which genomes are modified *only* by duplications and losses of genes.

One *step* of tandem duplication - random loss, or duplication-loss for short, consists in (1) the tandem duplication of a contiguous fragment of the genome, *i.e.,* the duplicated fragment is inserted immediately after the original fragment, and (2) the loss of one of the two copies of every duplicated gene. We assume that the loss occurs immediately after the duplication of genes, which is, on an evolutionary time-scale, a good approximation to reality. The *width* of a step is the number of duplicated genes. See Figure 1 for an example.

$$\begin{array}{rcl} 1\ 2\ \overbrace{3\ 4\ 5\ 6}\ 7 & \rightsquigarrow & 1\ 2\ \overbrace{3\ 4\ 5\ 6}\ \overbrace{3\ 4\ 5\ 6}\ 7 \\ & & \text{(tandem duplication)} \\ & \rightsquigarrow & 1\ 2\ \cancel{3}\ 4\ 5\ \cancel{6}\ 3\ \cancel{4}\ \cancel{5}\ 6\ 7 \\ & & \text{(random loss)} \\ & \rightsquigarrow & 1\ 2\ 4\ 5\ 3\ 6\ 7 \end{array}$$

Figure 1: Example of one step of tandem duplication - random loss of width 4

From a formal point of view, a genome consisting of n genes is modelled by a permutation $\pi \in S_n$ of the set of integers $\{1, 2, \ldots, n\}$. In [3], the authors define the cost of a duplication-loss step of width k to be α^k, $\alpha \geq 1$ being a constant parameter. They suggest that other cost functions can be considered, and in particular affine functions. In this paper, we consider a *piecewise constant* cost function: the cost of a step of width k is 1 if $k \leq K$ and is infinite for $k > K$, for some fixed parameter $K \in \mathbb{N} \cup \{\infty\}$. Obviously, for this model to be meaningful, we assume that $K \geq 2$. We also consider the possibility that $K = K(n)$ is dependent on the size n of the permutation on which the duplication-loss operations are performed. Both models are generalizations of the *whole genome duplication - random loss model*: it corresponds to the case $\alpha = 1$ in the model of [3], $K = \infty$ or $K = K(n) = n$ in our model.

Many models of evolution of permutations are inspired by computational biology issues: see [2], [4], [5], [6] for examples in the litterature.

Our model of evolution of permutations can be viewed in the framework of *permuting machines* defined in [1]. Such a machine takes a permutation in input, and transforms it into an output permutation, the transformation being subject to satisfy the two properties of independance with respect to the values and of stability with respect to pattern-involvement (see [1] for more details).

The important point is that the duplication-loss transformation satisfies these two properties. Thus, one duplication-loss step (in one of the models defined above) corresponds to running an adequate permuting machine once. When we will consider permutations obtained after a sequence of duplication-loss steps, it will correspond to permutations obtained in the output of a combination in series of identical permuting machines.

Though not appearing clearly for the moment, there exist strong links between the duplication-loss model and some pattern-avoiding classes of permutations. Hence, we need to recall a few definitions concerning those classes.

A permutation $\sigma \in S_n$ is a bijective map from $[1..n]$ to itself. The integer n is called the *size* of σ, denoted $|\sigma|$. We denote by σ_i the image of i under σ. A permutation can be seen as a word $\sigma_1\sigma_2\ldots\sigma_n$ containing exactly once each letter $i \in [1..n]$. For each entry σ_i of a permutation σ, we call i its *position* and σ_i its *value*.

Definition 1. *A permutation $\pi \in S_k$ is a* pattern *of a permutation $\sigma \in S_n$ if there is a subsequence of σ which is order-isomorphic to π; in other words, if there is a subsequence $\sigma_{i_1}\sigma_{i_2}\ldots\sigma_{i_k}$ of σ (with $1 \leq i_1 < i_2 < \ldots < i_k \leq n$) such that $\sigma_{i_\ell} < \sigma_{i_m}$ whenever $\pi_\ell < \pi_m$.*
We also say that π is involved *in σ and call $\sigma_{i_1}\sigma_{i_2}\ldots\sigma_{i_k}$ an* occurrence *of π in σ.*

We write $\pi \prec \sigma$ to denote that π is a pattern of σ.

A permutation σ that does not contain π as a pattern is said to *avoid* π. The class of all permutations avoiding the patterns $\pi_1, \pi_2 \ldots \pi_k$ is denoted $S(\pi_1, \pi_2, \ldots, \pi_k)$, and $S_n(\pi_1, \pi_2, \ldots, \pi_k)$ denotes the set of permutations of size n avoiding $\pi_1, \pi_2, \ldots, \pi_k$. We say that $S(\pi_1, \pi_2, \ldots, \pi_k)$ is a class of pattern-avoiding permutations of *basis* $\{\pi_1, \pi_2, \ldots, \pi_k\}$.

Example 1. *For example $\sigma = 142563$ contains the pattern $\pi = 1342$; and 1563, 1463, 2563 and 1453 are the occurrences of this pattern in σ. But $\sigma \in S(321)$: σ avoids the pattern 321 as no subsequence of size 3 of σ is isomorphic to 321, i.e., is decreasing.*

2 Our study of this model

2.1 Characterization with excluded patterns

In the tandem duplication - random loss model described above, we will focus on two kinds of problems. First, as hinted before, we will consider permutations obtained after a certain number of duplication-loss steps, that is to say permutations in output of a combination in series of a certain number of permuting machines. For this, we define the class $\mathcal{C}(K, p)$ as follows:

Definition 2. *The class $\mathcal{C}(K, p)$ denotes the class of all permutations obtained from $12\ldots n$ (for any n) after p duplication-loss steps of width at most K, for some constant parameters p and K.*

We do not consider the case $K = K(n)$ here.

Be careful that the duplication-loss steps are not reversible, as noticed in [3], and that consequently $\mathcal{C}(K, p)$ is *not* the class of permutations that can be *sorted to* $12\ldots n$ in p steps of duplication-loss of width at most K.

Like for the various classes of permutations obtained after a combination in series of permuting machines considered in [1], we obtained combinatorial properties of $\mathcal{C}(K, p)$ in terms of pattern-avoidance. Namely, we show that $\mathcal{C}(K, p)$ is a class of pattern-avoiding permutations, whose basis is finite. Indeed, since $\mathcal{C}(K, p)$ is stable for the pattern-involvement relation $\prec$, it is a simple consequence that $\mathcal{C}(K, p)$ is a class of pattern-avoiding permutations, and that its basis is composed of the minimal permutations (minimal being intended in the sense of $\prec$) that do not belong to $\mathcal{C}(K, p)$. Our goal is then to characterize those minimal permutations, and prove that there is only a finite number of them.

In the case $p = 1$, we give a precise description of the basis B of excluded patterns that describe $\mathcal{C}(K, 1)$:

Proposition 1. *$\mathcal{C}(K,1)$ is a class of pattern-avoiding permutations $S(B)$ whose basis B is finite. More precisely, $B = \{321, 3142, 2143\} \cup D$, D being the set of all permutations of S_{K+1} that do not start with 1 nor end with $K+1$, and containing exactly one descent (*i.e.* two consecutive elements in decreasing order).*

In particular, B is of cardinality $3 + 2^{K-1}$ and contains patterns of size at most $K+1$.

For the general case $\mathcal{C}(K,p)$, we cannot get such a precise result but only a bound on the size of the excluded patterns. This is achieved by defining the *value-position vectors* and *value-position domains* on permutations.

The *vector* from i to j in a permutation σ consists of all elements whose positions lie between the positions of i and j, i and j being included. The *size* of a vector is the number of elements in it. For example, the vector from 7 to 2 in the permutation 4123576 is $\overleftarrow{2357}$, and has size 4.

Definition 3. *Let σ be a permutation of S_n. The* value-position vector *associated with $i \in [1..n]$ is the vector of σ going from i to σ_i, if i is not a fixpoint of σ. In the case $i = \sigma_i$, the value-position vector associated with i is empty.*

Let σ be a permutation of S_n. The value-position domain *of σ is composed of all elements of σ appearing in at least one value-position vector.*

This definition is illustrated on Figure 2.

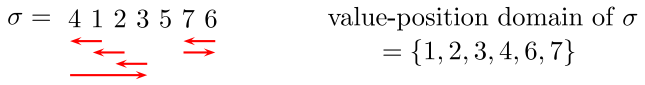

Figure 2: value-position vectors and value-position domain for $\sigma = 4123576$

With these tools, we can bound the size of the value-position domain of any minimal permutation (in the sense of $\prec$) that do not belong to $\mathcal{C}(K,p)$. Further technical considerations allow us to prove that the minimal permutations that do not belong to $\mathcal{C}(K,p)$ are of size at most $(Kp+2)^2 - 2$. This finally shows that:

Proposition 2. *$\mathcal{C}(K,p)$ is a class of pattern-avoiding permutations whose basis is finite and contains patterns of size at most $(Kp+2)^2 - 2$.*

2.2 Number of steps of width K to obtain any permutation of size n

A second point of view is to examine how many steps of a given width are necessary to obtain any permutation of S_n starting from $12 \ldots n$. Namely we fix a width K (constant, or $K = K(n)$) and a size n and search for the number p such that any permutation of S_n can be obtained from $12 \ldots n$ in at most p duplication-loss steps of width at most K.

Algorithm 1 An optimal whole genome duplication - random loss scenario from $12 \ldots K$ to $\sigma \in S_K$

```
$\pi = 12 \ldots K$
Partition $\sigma$ into maximal increasing substrings, from left to right
Each element of $[1..K]$ appearing in the $i^{th}$ maximal increasing substring gets as a label the binary representation of $i$
for $j = 1$ to $\lceil \log_2(\text{number of maximal increasing substrings of } \sigma) \rceil$ do
    Perform a duplication-loss step on $\pi$ that keeps in the first copy of $\pi$ exactly the elements whose label has a 0 in its $j^{th}$ least significant bit
end for
```

We describe an algorithm computing a possible scenario of duplications and losses for any $\pi \in S_n$, this scenario involving $\Theta(\frac{n}{K} \log K + \frac{n^2}{K^2})$ duplication-loss steps in the worst case and on average.

For this purpose, we use the algorithm described in [3] that computes a scenario whose number of steps is minimal in the *whole genome duplication* - random loss model. This procedure is described in Algorithm 1.

Note that the same algorithm can be used to compute an optimal whole genome duplication - random loss scenario from $i_1 i_2 \ldots i_k$, with $k \leq K$ and $i_1 < i_2 < \ldots < i_k$, to any permutation of $\{i_1, i_2, \ldots, i_k\}$.

Our procedure for computing a tandem duplication - random loss scenario for any permutation $\sigma \in S_n$ is given in Algorithm 2.

Algorithm 2 A duplication-loss scenario from $12 \ldots n$ to $\sigma \in S_n$

```
1: $\pi \leftarrow 12 \ldots n$
2: for $i = 1$ to $\lceil \frac{n-K}{\lfloor K/2 \rfloor} \rceil$ do
3:     Let $L^i = \{\sigma_j : n - i\lfloor K/2 \rfloor + 1 \leq j \leq n - (i-1)\lfloor K/2 \rfloor\}$
4:     Perform duplication-loss steps on $\pi$ to move from left to right the elements of $L^i$ to the
       positions $n - i\lfloor K/2 \rfloor + 1$ to $n - (i-1)\lfloor K/2 \rfloor$ of $\pi$, without changing their respective order
5: end for
6: for $i = 1$ to $\lceil \frac{n-K}{\lfloor K/2 \rfloor} \rceil$ do
7:     Perform Algorithm 1 on the window of $\pi$ between the indices $n - i\lfloor K/2 \rfloor + 1$ and $n - (i-1)\lfloor K/2 \rfloor$
8: end for
9: Perform Algorithm 1 on the window of $\pi$ between the indices 1 and $n - \lceil \frac{n-K}{\lfloor K/2 \rfloor} \rceil \lfloor K/2 \rfloor$
```

A few keys to understand Algorithm 2 are the following remarks.

The set L^i of values defined at line 3 represents the rightmost $\lfloor K/2 \rfloor$ elements of σ not yet examined. At the end of the first loop (line 5), π is decomposed into windows of width $\lfloor K/2 \rfloor$ (except the leftmost one which is of width at most K) ; and each of these windows is an increasing sequence containing exactly the same elements as the window of σ corresponding to the same indices. In the second loop, we consider these windows from right to left and since there are of width less than K, we can call Algorithm 1 (that implements whole genome duplication-random loss) on each window successively to transform π into σ.

We could analyse the number of steps of the scenario produced by Algorithm 2 and prove that:

Proposition 3. *The number of duplication-loss steps of a scenario produced by Algorithm 2 on a permutation of size n is at most $\Theta(\frac{n}{K} \log K + \frac{n^2}{K^2})$ asymptotically. The same holds for the average number of steps.*

Provided with this upper bound on the number of steps that are necessary to obtain a permutation from the indentity in the tandem duplication - random loss model, we now turn to the computation of lower bounds. Again, [3] provides us with a $\Omega(\log n)$ lower bound, refering to the whole genome duplication - random loss model which is more general than ours, so that this bound applies in our context. Simple considerations on the number on inversions that a duplication-loss step can create yield a $\Omega(\frac{n^2}{K^2})$ lower bound. Those results hold for the worst case and for the average case.

We finally obtain that:

Proposition 4. $\Omega(\log n + \frac{n^2}{K^2})$ *duplication-loss steps are necessary (in the worst case and on average) to obtain any permutation of S_n from $12 \ldots n$.*

We could not provide lower bounds that coincide with the upper bounds given above, but we claim that they are tight in many cases. Indeed, whenever $K = o(\frac{n}{\log n})$, we get that $\frac{n}{K} \log K = o(\frac{n^2}{K^2})$, and consequently the upper bound can be rewritten as $\Theta(\frac{n}{K} \log K + \frac{n^2}{K^2}) = \Theta(\frac{n^2}{K^2})$, which coincides up to a constant factor with the lower bound $\Omega(\log n + \frac{n^2}{K^2}) = \Omega(\frac{n^2}{K^2})$. For the case $K = \Theta(\frac{n}{\log n})$, the same argument holds, but the constant factor between the lower and the upper bound might be much greater. Finally, if $K = \Theta(n)$, then $\Theta(\frac{n}{K}\log K + \frac{n^2}{K^2}) = \Theta(\log n)$ and $\Omega(\log n + \frac{n^2}{K^2}) = \Omega(\log n)$, so that upper and lower bounds coincide again.

On the contrary, when $\frac{n}{\log n} \ll K \ll n$, the upper and lower bounds provided do not coincide. We leave as an open question the problem of finding an algorithm that computes a duplication-loss scenario whose number of steps is optimal (on average and in the worst case) up to a constant factor, when the width K of the duplicated windows satisfies $\frac{n}{\log n} \ll K \ll n$.

References

[1] M.H. Albert, R.E.L. Aldred, M.D. Atkinson, H.P. Van Ditmarsch, C.C. Handley, D.A. Hotlon, and D.J. McCaughan. Compositions of pattern restricted sets of permutations. Technical report, University of Otago, 2004. Technical report number OUCS-2004-12.

[2] S. Bérard, A. Bergeron, C. Chauve, and C. Paul. Perfect sorting by reversals is not always difficult. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 4(1), 2007.

[3] K. Chaudhuri, K. Chen, R. Mihaescu, and S. Rao. On the tandem duplication-random loss model of genome rearrangement. SODA, pages 564 – 570, 2006.

[4] M.C. Chen and R.C.T. Lee. Sorting by transpositions based on the first increasing substring concept. In *BIBE '04: Proceedings of the 4th IEEE Symposium on Bioinformatics and Bioengineering*, page 553, Washington, DC, USA, 2004. IEEE Computer Society.

[5] Anthony Labarre. A new tight upper bound on the transposition distance. In Rita Casadio and Gene Myers, editors, *WABI*, volume 3692 of *Lecture Notes in Computer Science*, pages 216–227. Springer, 2005.

[6] Anthony Labarre. New bounds and tractable instances for the transposition distance. *IEEE/ACM Trans. Comput. Biology Bioinform*, 3(4):380–394, 2006.