



HAL
open science

The Combinatorics of Tandem Duplication Trees

Olivier Gascuel, Michael D. Hendy, Alain Jean-Marie, Robert Mclachlan

► **To cite this version:**

Olivier Gascuel, Michael D. Hendy, Alain Jean-Marie, Robert Mclachlan. The Combinatorics of Tandem Duplication Trees. *Systematic Biology*, 2003, 52, pp.110-118. 10.1080/10635150390132821 . lirmm-00192006

HAL Id: lirmm-00192006

<https://hal-lirmm.ccsd.cnrs.fr/lirmm-00192006>

Submitted on 26 Nov 2007

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

The Combinatorics of Tandem Duplication Trees

OLIVIER GASCUEL,¹ MICHAEL D. HENDY,^{2,3} ALAIN JEAN-MARIE,¹ AND ROBERT MCLACHLAN³

¹Département d'Informatique Fondamentale et Applications, LIRMM, 161 Rue Ada, 34392 Montpellier, France

²Allan Wilson Centre for Molecular Ecology and Evolution, Massey University, Palmerston North, New Zealand

³Institute of Fundamental Sciences, Massey University, Palmerston North, New Zealand

Abstract.— We developed a recurrence relation that counts the number of tandem duplication trees (either rooted or unrooted) that are consistent with a set of n tandemly repeated sequences generated under the standard unequal recombination (or crossover) model of tandem duplications. The number of rooted duplication trees is exactly twice the number of unrooted trees, which means that on average only two positions for a root on a duplication tree are possible. Using the recurrence, we tabulated these numbers for small values of n . We also developed an asymptotic formula that for large n provides estimates for these numbers. These numbers give a priori probabilities for phylogenies of the repeated sequences to be duplication trees. This work extends earlier studies where exhaustive counts of the numbers for small n were obtained. One application showed the significance of finding that most maximum-parsimony trees constructed from repeat sequences from human immunoglobins and T-cell receptors were tandem duplication trees. Those findings provided strong support to the proposed mechanisms of tandem gene duplication. The recurrence relation also suggests efficient algorithms to recognize duplication trees and to generate random duplication trees for simulation. We present a linear-time recognition algorithm. [Asymptotic enumeration; random generation; recognition; recursion; tandem duplication trees.]

Duplicated sequences in DNA are very common. Baltimore (2001) stated that over half of the (human) DNA consists of repeated sequences. Among these repeated sequences, we can distinguish so-called tandem repeats, in which the copies are adjacent along the genome. Tandem repeats can be made of mini- or microsatellites that are relatively short and are usually considered parasitic and potentially deleterious. But these repeats can also be larger and can contain genes. The duplication (in tandem or not) of genes is one of the most important evolutionary mechanisms for producing genes with novel functions.

The process of unequal recombination (or crossover) is widely viewed as responsible for the production of large tandemly repeated sequences, possibly containing genes. At each step, a segment (containing one or several copies) is duplicated into two adjacent and identical segments that are then free to diverge through mutations. As the process is repeated, a linearly ordered set of paralogous segments results. The evolutionary history of these segments shares many common points with that of orthologous sequences, as used in phylogenetic studies. However, because of the linear order of the segments and the nature of the process, tandem duplication histories are much more constrained than speciation histories, as was first pointed out by Fitch (1977) and recently and independently reiterated by Benson and Dong (1999), Tang et al. (2001), and Elemento et al. (2001, 2002). This latest study provided a duplication history for the human TRGV locus that was constructed from a single genome but allowed for the most striking polymorphism (absence of two adjacent genes) in the human population to be predicted. This fact lent strong support to the assumptions concerning the duplication mechanism and indicated that reliable tandem duplication histories can be constructed, at least in some cases.

However, because of the lack of a molecular clock, neither the location of the root nor the temporal order of the

duplication events can always be determined. Thus, for any given tandem duplication history, Fitch (1977) and Elemento et al. (2002) defined a corresponding rooted duplication tree (RDT) in which nonnested duplication events are not temporally related and an (unrooted) duplication tree (DT) in which the root location is also lost. Unrooted and eventually rooted duplication trees can be inferred from the contemporary sequences, but not from the full duplication history, just as is the case with speciation histories, where the temporal information is partly lost and we are only able to infer (eventually rooted) phylogenies.

Several algorithms to infer duplication trees were proposed by Fitch (1997), Benson and Dong (1999), Tang et al. (2002), and Elemento et al. (2002). A simple algorithm involves inferring one or several possible phylogenies for the sequences being studied and then checking whether these phylogenies are compatible with the tandem duplication model. For example, with the seven repeated segments of human apolipoprotein A-I, Fitch (1977) found four most-parsimonious phylogenies, only one of which was a duplication tree, whereas for the nine genes of the human TRGV locus, Elemento et al. (2002) found one most-parsimonious phylogeny, which was a duplication tree. Two problems thus arise. The first is related to the significance of such results. When their prior probability is low, we can be quite confident in the duplication model. For example, Elemento et al. (2002) showed that the probability for a phylogeny of nine sequences to be a duplication tree is only about 0.0385, which provided evidential support for the unequal recombination hypothesis. The second problem concerns computational efficiency. For large data sets, this simple approach requires a fast algorithm to recognize among the possibly very numerous near-optimal phylogenies those that are duplication trees. However, a specialized algorithm that considered duplication trees only could provide a more efficient search procedure.

This article deals with the combinatorics of tandem duplication trees. Fitch (1977) determined the numbers of rooted and unrooted duplication trees for a number of segments $n \leq 7$, whereas Elemento et al. (2002) computed these numbers exactly for $n \leq 10$ and estimated them by sampling for $10 < n \leq 14$. They also noted that for $n \leq 10$, the number of rooted duplication trees, denoted as $RDT(n)$, is twice the number of (unrooted) duplication trees, denoted by $DT(n)$. We developed a recurrence relation for both $RDT(n)$ and $DT(n)$, proved that $RDT(n) = 2DT(n)$ for $n \geq 3$, and determined their asymptotic behaviors as $n \rightarrow \infty$. These results provided the tool to address the problem of significance by making it possible to estimate the a priori probability of a phylogeny being a duplication tree. Moreover, our recurrence relations suggested efficient algorithms that could be used to recognize the phylogenies that are duplication trees, to search the space of duplication trees, and to uniformly randomly generate duplication trees for simulations. Here, we introduce the notation and define the tandem duplication model, demonstrate the recurrence relations and analyze their asymptotic behavior, and provide algorithms for recognition and random generation of duplication trees.

DUPLICATION EVENTS

An initial gene segment, σ , is duplicated to produce the string $\sigma\sigma$ of two concatenated segments. Over time, mutations accumulate and the copies become distinguishable. We distinguish these segments so the string is labeled $\sigma_1\sigma_2$. Subsequent duplications of one or more adjacent segment copies can occur, to produce a string $\sigma_1\sigma_2 \dots \sigma_n$ of n concatenated (and mutated) copies of σ .

When a duplication involves copying r adjacent segments, with k segments remaining to the right of these copies, then we refer to that as a (k, r) duplication event. When a string of n segments results, we have $1 \leq r \leq n/2$ and $0 \leq k \leq n - 2r$. Figure 1 provides an example of a $(k, r) = (3, 2)$ duplication event resulting in a string of $n = 8$ segments. A (k, r) duplication event involves copying the substring of segments $\sigma_a \dots \sigma_b$ from $\sigma_1 \dots \sigma_m$, where $m = n - r$, $a = m - r - k + 1$, and $b = m - k$, to

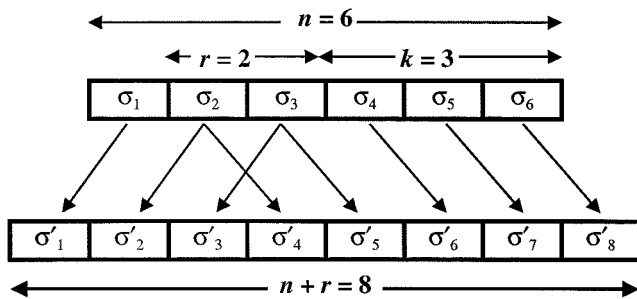


FIGURE 1. An example of a $(k, r) = (3, 2)$ duplication event on a string of $n = 6$ duplicated segments that produces a string of $n + r = 8$ segments. σ_1 is relabeled as σ'_1 , and the $r = 2$ segments σ_2 and σ_3 are duplicated, becoming σ'_2, σ'_4 and σ'_3, σ'_5 , respectively. The remaining $k = 3$ segments, $\sigma_4, \sigma_5, \sigma_6$, are relabeled as $\sigma'_6, \sigma'_7, \sigma'_8$.

produce a new string $\sigma'_1 \dots \sigma'_n$ of $n = m + r$ segments. The pair $\sigma'_{a+j}, \sigma'_{a+r+j}$, $0 \leq j < r$ comprising two copies of σ_{a+j} , is referred to as twins. As the copies accumulate mutations, the descendants of the twins diverge, and the duplication history is reflected in the phylogeny of the duplicated segments.

The sequence of duplication events (linearly ordered by time) giving rise to a set of n segment copies is called a duplication history (Fig. 2a). Let $\mathcal{DH}(n)$ be the set of all duplication histories that produce strings of n duplicate segments, and let $DH(n) = \#\mathcal{DH}(n)$ be the number of such histories. Given any $H \in \mathcal{DH}(m)$, any (k, r) duplication event transforms H to a duplication history $H' \in \mathcal{DH}(n)$, where $n = m + r$. The number of

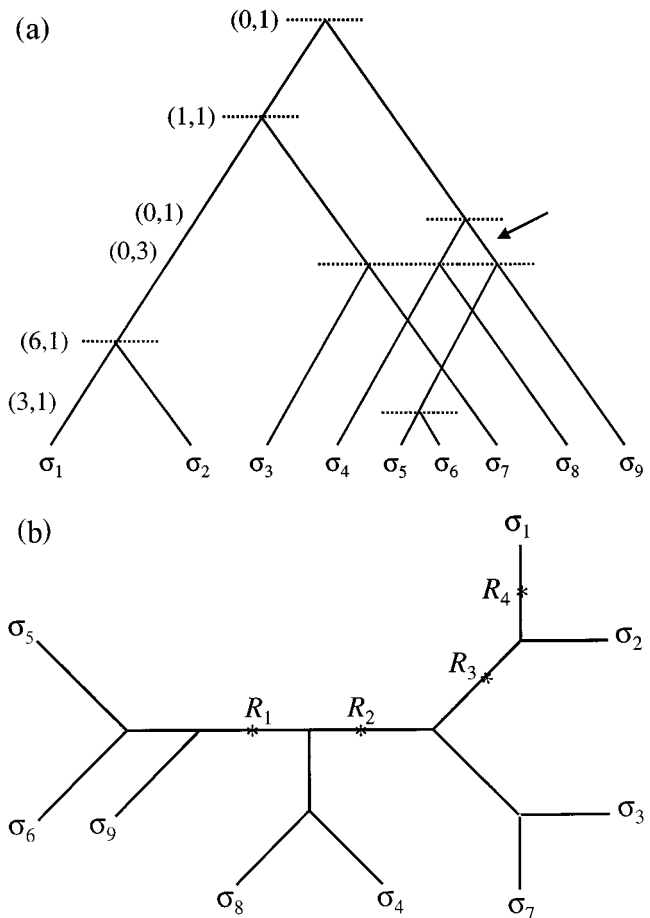


FIGURE 2. (a) An example of a duplication history $H \in \mathcal{DH}(9)$ resulting from the sequence of duplication events $(k, r) = (0, 1), (1, 1), (0, 1), (0, 3), (6, 1),$ and $(3, 1)$. The parameter k is the number of segments of remaining to the right of the duplication, and r is the number of segments duplicated in the event. The arrow indicates the location of the root of the canonical representative of H . (b) The corresponding duplication tree $T = T(H) \in DT(9)$. Only the two duplications, $(6, 1)$ and $(3, 1)$, of H are visible. There are four possible locations for a root (identified by $R_i, i = 1, 2, 3, 4$) on the path connecting σ_1 and σ_9 . R_1 is the closest to σ_9 (it cannot be closer because the next duplication $(0, 3)$ is a multiple duplication), so rooting at R_1 would give the canonical tree. R_2 is the location of the root for the history H of tree a .

duplication histories, $DH(n)$, that can give rise to n copies was shown (Elemento et al., 2002) to satisfy the recurrence

$$\begin{aligned} DH(n) &= \sum_{r=1}^{n/2} \sum_{k=0}^{n-2r} DH(n-r) \\ &= \sum_{r=1}^{n/2} (n-2r+1)DH(n-r), \quad n > 1, \end{aligned} \quad (1)$$

with $DH(1) = 1$.

For a duplication history H , we define the (k, r) duplication event in H to be visible if none of the $2r$ copied segments of that event have been duplicated subsequently. A phylogenetic tree of the n segments $\sigma_1, \dots, \sigma_n$ of $H \in \mathcal{DH}(n)$ does not distinguish the temporal order of the visible duplication events nor identify the position of the root of H . We define the RDT $R = R(H)$ to be the rooted tree derived from H (ignoring the temporal order of duplication events recorded), and the (unrooted) DT $T = T(H)$ is defined to be the unrooted tree derived from R . A duplication tree can be derived from more than one distinct duplication history, i.e., these maps are not one to one.

Let $\mathcal{RDT}(n)$ be the set of rooted duplication trees obtained from the histories in $\mathcal{DH}(n)$, and let $\mathcal{DT}(n)$ be the set of (unrooted) duplication trees obtained from the histories in $\mathcal{DH}(n)$.

RECURSIONS

For $n \geq 2$ and $0 < k \leq n - 2$, let $\mathcal{P}(n, k)$ be the set of all trees in $\mathcal{RDT}(n)$ whose leftmost visible duplication event is (k, r) , for $1 \leq r \leq (n - k)/2$. Thus for a tree in $\mathcal{P}(n, k)$ and given $1 \leq r \leq (n - k)/2$, the $2r$ duplicated segments form the substring $\sigma_{n-k-2r+1} \dots \sigma_{n-k}$, and there is no visible duplication among $\sigma_1 \dots \sigma_{n-k-2r}$. We find a bijection (a one to one and onto map) that shows that the number of trees in $\mathcal{P}(n, k)$ is the same as the number of trees in the union of the sets $\mathcal{P}(n - 1, 0), \dots, \mathcal{P}(n - 1, k + 1)$.

Theorem 1

For $n > 2$, $0 \leq k \leq n - 2$, there is a bijection between $\mathcal{P}(n, k)$ and $\cup_{j=0}^{k+1} \mathcal{P}(n - 1, j)$. For $k \geq n - 1$, $\mathcal{P}(n, k) = \emptyset$.

Proof.—Let T be a tree in $\mathcal{P}(n, k)$ for $n > 2$, hence the first (leftmost) visible duplication in T is a (k, r) duplication event for some $r \geq 1, k + 2r \leq n$. Thus $\mathcal{P}(n, k) = \emptyset$ if $k > n - 2$.

For $k \leq n - 2$, we can map T to $T' \in \mathcal{RDT}(n - 1)$ by deleting the segment σ_{n-k-r} (the left twin of σ_{n-k} , the rightmost segment of the first visible duplication).

When $r > 1$, the first visible duplication event of T' will be a $(k + 1, r - 1)$ event, so $T' \in \mathcal{P}(n - 1, k + 1)$.

When $r = 1$, the next visible duplication event ends at or to the right of $\sigma_{(n-1)-k}$, so $T' \in \mathcal{P}(n - 1, j)$ for some $j, 0 \leq j \leq k$.

This map is invertible because with any $T' \in \mathcal{P}(n - 1, j)$ for $0 \leq j \leq k$, we can duplicate $\sigma_{(n-1)-k}$ as adjacent

twins to obtain $T \in \mathcal{P}(n, k)$, with the leftmost duplication event being $(k, 1)$;

Given any $T' \in \mathcal{P}(n - 1, k + 1)$, with the first visible duplication being a $(k + 1, s)$ event with $s \geq 1$, we can duplicate $\sigma_{(n-1)-k}$ and insert the left twin between $\sigma_{(n-1)-k-s-1}$ and $\sigma_{(n-1)-k-s}$. This extends the event to a $(k, s + 1)$ duplication event and creates $T \in \mathcal{P}(n, k)$, with the leftmost duplication event being (k, r) with $r = s + 1 > 1$.

Hence the sets are bijective.

QED

Let $p(n, k) = \#\mathcal{P}(n, k)$ and $RDT(n) = \#\mathcal{RDT}(n)$.

Theorem 2

The number of rooted duplication trees $RDT(n) = \sum_{k=0}^{n-2} p(n, k)$ for $n > 2$ segments, can be determined by the two parameter recursion

$$p(n, 0) = p(n - 1, 0) + p(n - 1, 1) \quad (2)$$

and

$$\begin{aligned} p(n, k) &= p(n - 1, k + 1) + p(n, k - 1), \text{ for} \\ &k = 1, \dots, n - 2, \end{aligned} \quad (3)$$

and

$$p(n, n - 4) = p(n, n - 3) = p(n, n - 2) = RDT(n - 1), \quad (4)$$

where $p(n, k) = 0$ for $k < 0$ and $k \geq n - 1$; and with initial value $p(2, 0) = 1$.

Proof.—Because the first duplication event creates at least two segment copies, $\mathcal{P}(n, k) = \emptyset$ for $k > n - 2$ and $\mathcal{P}(2, 0)$ contains a single history.

For $n > 2$, the sets $\mathcal{P}(n - 1, j)$ are disjoint for distinct values of $j = 0, \dots, n - 3$, so from Theorem 1 the cardinality of $\mathcal{P}(n, k)$ is the sum of the cardinalities of $\mathcal{P}(n - 1, j)$ for $j = 0, \dots, k + 1$, hence

$$p(n, k) = \sum_{j=0}^{k+1} p(n - 1, j). \quad (5)$$

Thus, for $k = 0$, Equation 2 follows. For $k > 0$, it follows from Equation 5 that

$$p(n, k) - p(n, k - 1) = p(n - 1, k + 1),$$

so Equation 3 follows.

As $p(n - 1, j) = 0$ for $j \geq n - 2$,

$$\begin{aligned} p(n, n - 4) &= p(n, n - 3) = p(n, n - 2) = \sum_{j=0}^{n-3} p(n - 1, j) \\ &= RDT(n - 1), \end{aligned}$$

which implies Equation 4.

QED

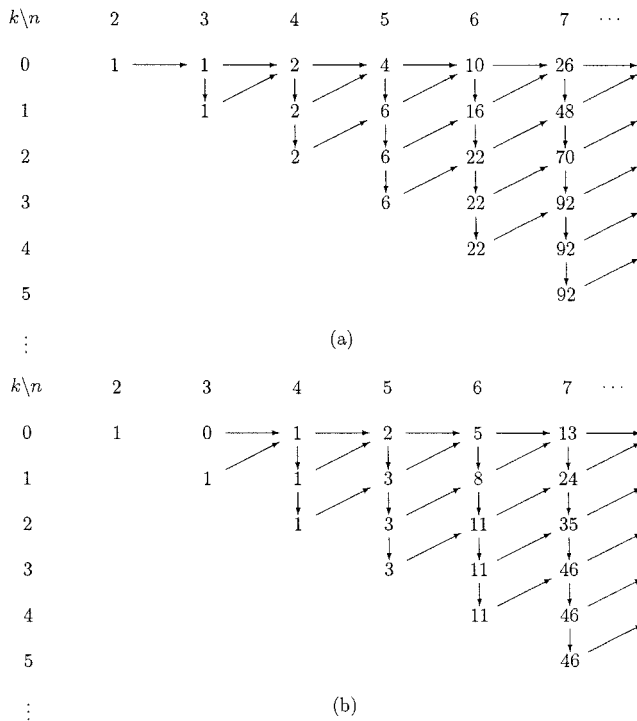


FIGURE 3. The recursion calculations. (a) The numbers $p(n, k)$, from which $RDT(n) = \sum_{k=0}^{n-2} p(n, k) = p(n+1, n-1)$, is the number of rooted duplication trees on n duplicated segments. (b) The numbers $a(n, k)$, from which $DT(n) = \sum_{k=0}^{n-2} a(n, k) = a(n+1, n-1)$, is the number of duplication trees on n duplicated segments. Each entry is the sum of the values of the two (or one) incoming arrows, except for the initial values at the left. For $k = 0, 1, 2$, $p(4, k) = 2a(4, k)$, and hence, by the recursion, $p(n, k) = 2a(n, k)$, $\forall n \geq 4, 0 \leq k \leq n-2$. Hence, $RDT(n) = 2DT(n)$, $\forall n \geq 3$.

The calculation of the values of $p(n, k)$, for $n = 2, \dots, 7$ and $0 \leq k \leq n-2$, are derived in Figure 3, using the recurrences of Theorem 2.

COUNTING DUPLICATION TREES

Starting with a single segment σ , the first duplication event of any history $H \in \mathcal{DH}(n)$ must duplicate σ , so it is a $(0, 1)$ event. This event separates σ_1 and σ_n , so the root of H must always lie on the path connecting σ_1 to σ_n . Thus, in any rooted duplication tree $T \in \mathcal{RDT}(n)$, the root must lie on the path connecting the leaves labeled by σ_1 and σ_n .

There are three possibilities for the second duplication event, either a single segment: the left (a $(1, 1)$ event) or the right (a $(0, 1)$ event) is duplicated, or both segments (a $(0, 2)$ event) are duplicated. Both single segment duplication histories $((0, 1), (1, 1))$ and $((0, 1), (0, 1))$ give rise to the same unrooted duplication tree, the unique binary tree on three leaves, where a root could be placed either on the edge to σ_1 or on the edge to σ_3 . The ambiguity of root placement continues with each further single duplication of either the first or last segment, but the root cannot occur beyond any multiple duplication. These events

mark the limits of the possible root locations on the path connecting σ_1 and σ_n in any unrooted duplication tree. In the case of the initial history $((0, 1), (0, 2))$, the root is “trapped” by the double duplication and only one root position is valid. Although the number of potential root placements on an unrooted duplication tree can vary, on average, as shown below, the number of possible root locations on duplication trees of $n > 2$ segments is exactly 2.

Let $u : \mathcal{RDT}(n) \rightarrow \mathcal{DT}(n)$ be the map that removes the root of any $T' \in \mathcal{RDT}(n)$. Then for $\bar{T} \in \mathcal{DT}(n)$, $u^{-1}(\bar{T})$ is the equivalence class of all trees $T' \in \mathcal{RDT}(n)$ that map to \bar{T} under u . We select as the canonical representative of this class the tree T' with root closest to σ_n . For $n > 2$, $T' \in \mathcal{RDT}(n)$ is canonical if one of the following holds: (1) the root of T' is the parent (direct ancestor) of σ_n or (2) the root of T' is the parent of a segment that is both an ancestor of σ_n and involved in a multiple duplication. Thus, T' is not canonical if its root is the parent of an ancestor of σ_n that is involved in a single duplication.

We count the number $DT(n) = \#\mathcal{DT}(n)$ of (unrooted) duplication trees by counting the number of canonical rooted duplication trees. Let $a(n, k)$ be the number of rooted trees in $\mathcal{P}(n, k)$ that are canonical. The history $((0, 1), (0, 1))$ does not give a canonical tree, but $((0, 1), (1, 1))$ and $((0, 1), (0, 2))$ do. Hence, $a(2, 0) = 1$, $a(3, 0) = 0$, and $a(3, 1) = 1$, so $DT(2) = DT(3) = 1$.

We now prove the following lemma that indicates the strong relationship between this canonical representation of duplication trees and the mapping of Theorem 1

Lemma 1

Let $T \in \mathcal{P}(n, k)$ for $n > 2$ and $0 \leq k \leq n-2$ and let $T' \in \cup_{j=0}^{k+1} \mathcal{P}(n-1, j)$ be the corresponding tree (under the bijection of Theorem 1). Then, T' is canonical iff T is canonical.

Proof.—The bijection of Theorem 1 maps $T \in \mathcal{P}(n, k)$ to a tree $T' \in \mathcal{P}(n-1, j)$, for some $j : 0 \leq j \leq k+1$, by deleting σ_i ($i = n-k-r$), the left twin of σ_{n-k} . Thus, the segments $\sigma_l, l > i$ in T are relabeled σ_{l-1} in T' , so in particular σ_n becomes σ_{n-1} . Let the root R of T be mapped to R' , the root of T' . R is on the edge to σ_n in T iff R' is on the edge to σ_{n-1} in T' . If R is above the first multiple duplication separating R from σ_n in T , then unless this duplication is visible with $k = 0$ and $r = 2$, the image of this duplication will remain, separating R' from σ_{n-1} in T' . Thus, in both cases T is canonical iff T' is canonical. Finally, if the first duplication separating R from σ_n in T is a $(0, 2)$ and visible, then in T' the image of this duplication will not be multiple, and R' will be on the edge to σ_{n-1} iff R is immediately above the duplication in T . Thus, in this case also T is canonical iff T' is canonical.

QED

Hence, apart from the special cases when $n = 2, 3$, the construction of Theorem 1 holds. Thus, following the proof of Theorem 2 we obtain the same recurrences but with different initial values.

Theorem 3

The number $DT(n) = \sum_{k=0}^{n-2} a(n, k)$, of duplication trees for $n \geq 4$ segments can be determined by the two parameter recursion: where for $1 \leq k \leq n - 3$,

$$a(n, 0) = a(n - 1, 0) + a(n - 1, 1), \tag{6}$$

$$a(n, k) = a(n - 1, k + 1) + a(n, k - 1), \tag{7}$$

$$a(n, n - 2) = a(n, n - 3) = a(n, n - 4) = DT(n - 1), \tag{8}$$

with initial values

$$a(2, 0) = 1, \quad a(3, 0) = 0, \quad a(3, 1) = 1,$$

(hence $DT(2) = DT(3) = 1$).

The calculation of the values of $a(n, k)$, for $n = 2, \dots, 7$ and $0 \leq k \leq n - 2$, are derived in Figure 3 using the recurrences of Theorem 3.

Corollary 1

For $n \geq 3$,

$$RDT(n) = 2DT(n),$$

i. e., the number of rooted duplication trees is twice the number of duplication trees.

Proof.—In Figure 3, the equation holds for $n = 3$ and $n = 4$, and $p(4, k) = 2a(4, k)$ for $k = 0, 1, 2$. By their common recursions Equations 3 and 7, $p(n, k) = 2a(n, k)$ for all $n > 4$ and $0 \leq k \leq n - 2$, and the result follows. QED

In Appendix 1, we use generating functions to find an asymptotic expression for $DT(n)$, which grows like 6.75^n .

Corollary 2

As $n \rightarrow \infty$

$$DT(n) \sim d \left(\frac{27}{4}\right)^n n^{-3/2} \tag{9}$$

where $d \simeq 0.00168809016$.

Exact and approximate values for $DT(n)$ are displayed in Table 1 for $n \leq 20$. The asymptotic Approximation 9 is sufficient for application purposes even with relatively small n (e.g., $n = 20$). Moreover, we recover the result from Elemento et al. (2002) that the a priori probability for a phylogeny of nine segments to be a duplication tree is only $\simeq 0.0385$, whereas the same probability with seven segments is $\simeq 0.222$. This latter relatively high value indicates that finding one duplication tree among four seven-segment phylogenies can be due to chance, as suggested by Fitch (1977: note added in proof).

UNIFORM RANDOM GENERATION OF DUPLICATION TREES

The recursions of Theorems 2 and 3 provide a means to generate sample duplication trees with uniform probability, both for the rooted and unrooted trees. These trees can be generated by starting from the unique two-segment history (0, 1) and iteratively adding segments, using the reverse mapping of Theorem 1, until the desired number of segments f is obtained. Let T' be the current tree in $\mathcal{P}(n, k)$, for $0 \leq k \leq n - 2$. Then T , the mapping of T' , is obtained by either (1) creating a twin of σ_j , for some j , where $1 \leq j \leq n - k$ or (2) when $k > 0$, extending the first visible (k, s) duplication to a $(k - 1, s + 1)$ duplication. Then T belongs to $\mathcal{P}(n + 1, n - j)$ or $\mathcal{P}(n + 1, k - 1)$, respectively. To obtain

TABLE 1. Numbers of duplication histories and trees calculated using the recurrences of Theorems 1 and 2. $DH(n)$ is the number of duplication histories (Eq. 1). $DT(n)$ is the number of (unrooted) duplication trees, and $f(n)$ is the value of the asymptotic approximation to $DT(n)$ given by Equation (9). $BT(n)$ is the number of (unrooted) binary trees with n labeled leaves, which is determined by the recursion $BT(n) = (2n - 5)BT(n - 1)$; for $n > 3$, $BT(3) = 1$ (Cavalli-Sforza and Edwards, 1967). The ratio $DT(n)/BT(n)$ is the probability that an arbitrary binary tree phylogeny represents a duplication tree.

| n | $DH(n)$ | $DT(n)$ | $f(n)$ | $BT(n)$ | $DT(n)/BT(n)$ |
|-----|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|
| 3 | 2 | 1 | 0.0999 | 1 | 1 |
| 4 | 7 | 3 | 0.4380 | 3 | 1 |
| 5 | 32 | 11 | 2.1157 | 15 | 7.33×10^{-1} |
| 6 | 182 | 46 | 10.864 | 105 | 4.38×10^{-1} |
| 7 | 1,224 | 210 | 58.194 | 945 | 2.22×10^{-1} |
| 8 | 9,500 | 1,021 | 321.51 | 1.04×10^4 | 9.82×10^{-2} |
| 9 | 8.35×10^4 | 5,202 | 1818.7 | 1.35×10^5 | 3.85×10^{-2} |
| 10 | 8.19×10^5 | 2.75×10^4 | 1.05×10^4 | 2.03×10^6 | 1.36×10^{-2} |
| 11 | 8.86×10^6 | 1.49×10^5 | 6.13×10^4 | 3.45×10^7 | 4.33×10^{-3} |
| 12 | 1.05×10^8 | 8.30×10^5 | 3.63×10^5 | 6.55×10^8 | 1.27×10^{-3} |
| 13 | 1.35×10^9 | 4.71×10^6 | 2.17×10^6 | 1.37×10^{10} | 3.42×10^{-4} |
| 14 | 1.87×10^{10} | 2.71×10^7 | 1.31×10^7 | 3.16×10^{11} | 8.57×10^{-5} |
| 15 | 2.78×10^{11} | 1.58×10^8 | 7.99×10^7 | 7.91×10^{12} | 2.00×10^{-5} |
| 16 | 4.41×10^{12} | 9.32×10^8 | 4.90×10^8 | 2.13×10^{14} | 4.37×10^{-6} |
| 17 | 7.45×10^{13} | 5.56×10^9 | 3.02×10^9 | 6.19×10^{15} | 8.98×10^{-7} |
| 18 | 1.33×10^{15} | 3.34×10^{10} | 1.87×10^{10} | 1.92×10^{17} | 1.74×10^{-7} |
| 19 | 2.52×10^{16} | 2.02×10^{11} | 1.16×10^{11} | 6.33×10^{18} | 3.19×10^{-8} |
| 20 | 5.01×10^{17} | 1.23×10^{12} | 7.28×10^{11} | 2.22×10^{20} | 5.56×10^{-9} |

a uniform distribution, we draw from among these possible moves with a probability that is proportional to the number of trees with f segments that can be generated from these moves. To generate unrooted trees, we generate canonical rooted trees only. Starting from the $(0, 1)$ history, the unique possible first move (the $(1, 1)$ duplication) is taken, and further steps are as for the rooted trees.

Let $m(f, n, k)$, for $3 \leq n \leq f$ and $0 \leq k \leq n - 2$, be the number of rooted trees of f segments that can be generated from a single tree in $\mathcal{P}(n, k)$. Clearly $m(f, f, k) = 1$ and from the above remarks,

$$m(f, n, k) = \sum_{j=\max(0, k-1)}^{n-1} m(f, n+1, j),$$

which for $0 \leq k \leq n - 2$ gives

$$\begin{aligned} m(f, n, 0) &= m(f, n, 1), \\ m(f, n, k) &= m(f, n+1, k-1) + m(f, n, k+1), \\ m(f, n, n) &= m(f, n+1, n-1). \end{aligned}$$

To generate uniformly random trees with f segments, we must first compute by dynamic programming all $m(f, n, k)$ values for $3 \leq n \leq f$ and $0 \leq k \leq n - 2$. Then these values are used at each step to compute the probability of each possible move.

The number of rooted duplication trees on f segments is $m(f, 2, 0)$, and the number of canonical rooted duplication trees and hence the number of unrooted duplication trees on f segments is $m(f, 3, 1)$, thus

$$\text{RDT}(f) = m(f, 2, 0), \quad \text{DT}(f) = m(f, 3, 1).$$

Further, for $f \geq 3$, $m(f, 3, 0) = m(f, 3, 1)$, so

$$\begin{aligned} m(f, 2, 0) &= m(f, 2, 1) = m(f, 3, 0) + m(f, 2, 2) \\ &= m(f, 3, 0) + m(f, 3, 1) = 2m(f, 3, 1), \end{aligned}$$

which gives another proof of

$$\text{RDT}(f) = 2\text{DT}(f).$$

LINEAR-TIME RECOGNITION ALGORITHM

Tang et al. (2002) and Elemento et al. (2001, 2002) provided an $\mathcal{O}(n^2)$ algorithm for testing whether a given rooted phylogeny on n ordered leaves is a duplication tree. Recently, Zhang et al. (2002) proposed a linear $\mathcal{O}(n)$ algorithm to solve the same problem. To recognize unrooted trees, which is the real problem because inferred trees are unrooted, potential roots were considered at each edge in the path from σ_1 to σ_n , providing an $\mathcal{O}(n^2)$ algorithm. This time complexity can be a serious disadvantage (at least for large n) when using the inference

procedure that consists of selecting from among the (possibly numerous) near-optimal phylogenies those that are duplication trees.

The recursion of Theorem 3 suggests an $\mathcal{O}(n)$ recognition algorithm for unrooted trees. Starting with the ordered segments $\sigma_1, \dots, \sigma_n$ at the leaves of a phylogeny, it searches for the leftmost visible duplication. If no visible duplication is found, then the answer is “no,” otherwise the duplication is reduced. If the reduction returns a tree of three segments, then the answer is “yes,” otherwise it returns to the previous search with the reduced tree.

For the algorithm to remain $\mathcal{O}(n)$, the search for the leftmost visible duplication uses a pointer that scans across the segments in order from σ_1 , never moving to useless points and storing the location of twins. A block, or partially visible duplication, is a nested sequence of twins, $(\sigma_i, \sigma_{i+r}), (\sigma_{i+1}, \sigma_{i+1+r}), \dots, (\sigma_{i+g}, \sigma_{i+g+r})$, for $g < r$. The algorithm remembers the endpoints of already encountered blocks, so that after the removal of a visible duplication, the pointer can continue the investigation of an uncovered duplication, without returning to its leftmost segment. In this way, the pointer always moves from left to right, unless a visible duplication is reduced, in which case it jumps to its starting point. Then the number of steps is $\mathcal{O}(n)$, as is the time complexity of the whole algorithm.

The same algorithm applies to the rooted case, returning “yes” when a tree is reduced to its root. More details and a pseudocode description are given in Appendix 2.

ACKNOWLEDGMENTS

We thank Mike Steel, Mike Hallett, and an anonymous reviewer for helpful discussions and comments on earlier drafts of the manuscript. The project was partially supported from funding from the Allan Wilson Centre, Massey University.

REFERENCES

- BALTIMORE, D. 2001. Our genome unveiled. *Nature* 409:814–816.
- BENSON, G., AND L. DONG. 1999. Reconstructing the duplication history of a tandem repeat. *Proc. Intelligent Syst. Mol. Biol.* 1999:44–53.
- CAVALLI-SFORZA, L., AND A. EDWARDS. 1967. Phylogenetic analysis: Models and estimation procedure. *Evolution* 21:550–570.
- ELEMENTO, O., O. GASCUEL, AND M.-P. LEFRANC. 2001. Reconstruction de l'histoire de duplication de gènes répétés en tandem. *Act. J. Ouv. Biol. Inform. Math.* 2001:9–11.
- ELEMENTO, O., O. GASCUEL, AND M.-P. LEFRANC. 2002. Reconstructing the duplication history of tandemly repeated genes. *Mol. Biol. Evol.* 19:278–288.
- FITCH, W. M. 1977. Phylogenies constrained by the crossover process as illustrated by human hemoglobins and a thirteen-cycle, eleven-amino-acid repeat in human apolipoprotein A-I. *Genetics* 86:623–644.
- ODLYZKO, A. M. 1995. Asymptotic enumerative methods. Pages 1063–1224 in *A handbook of combinatorics, Volume II* (R. L. Graham, M. Grötschel, and L. Lovász, eds.). Elsevier, Amsterdam.
- TANG, M., M. WATERMAN, AND S. YOOSEPH. 2002. Zinc finger gene clusters and tandem gene duplication. *J. Comp. Biol.* 9:429–446.
- ZHANG, L., B. MA, AND L. WANG. 2002. Efficient methods for inferring tandem duplication history. *Lect. Notes Comput. Sci.* 2542:97–111.

First submitted 7 July 2002; reviews returned 23 September 2002;

final acceptance 21 October 2002

Associate Editor: Mike Steel

APPENDIX 1
PROOF OF COROLLARY 2

Consider the family of numbers $\{a(n, k); 0 \leq k \leq n - 2; n \geq 2\}$, defined in Theorem 3, where for $n > 3$ and $1 \leq k \leq n - 2$

$$\begin{aligned} a(n, 0) &= a(n - 1, 0) + a(n - 1, 1); \\ a(n, k) &= a(n - 1, k + 1) + a(n, k - 1), \end{aligned} \tag{10}$$

with initial values

$$a(2, 0) = a(3, 1) = 1, \quad a(3, 0) = 0 \tag{11}$$

and $a(n, k) = 0$ elsewhere. Of particular interest are the values $\{b(n); n \geq 2\}$ for $b(n) := DT(n)$, defined as

$$b(n) = \sum_{k=0}^{n-2} a(n, k) \quad (= a(n + 1, n - 1) \text{ for } n \geq 3). \tag{12}$$

It is necessary to have bounds on the values of $a(n, k)$. Here, we prove the following property: for any $\beta > 1$, for $n \geq 3$ and $0 \leq k \leq n - 2$

$$a(n, k) \leq \left(\frac{\beta^2}{\beta - 1}\right)^{(n-3)} \beta^k.$$

This is true for $n = 3$, because $a(3, 0) = 0, a(3, 1) = 1$. If the property holds for $n - 1 \geq 3$, then from Equation 10,

$$\begin{aligned} a(n, 0) &= a(n - 1, 0) + a(n - 1, 1) \\ &\leq (1 + \beta) \left(\frac{\beta^2}{\beta - 1}\right)^{(n-4)} \leq \left(\frac{\beta^2}{\beta - 1}\right)^{(n-3)}, \end{aligned}$$

and if the equation holds for $n > 3, k - 1 \geq 0$, then by Equation 7

$$\begin{aligned} a(n, k) &= a(n - 1, k + 1) + a(n, k - 1) \\ &\leq \left(\frac{\beta^2}{\beta - 1}\right)^{(n-4)} \left[\beta^{k+1} + \frac{\beta^2}{\beta - 1} \beta^{k-1}\right] \\ &= \left(\frac{\beta^2}{\beta - 1}\right)^{(n-3)} \beta^k. \end{aligned}$$

Hence, for all $n \geq 3$ and $\beta > 1$

$$b(n) = a(n + 1, n - 2) \leq \left(\frac{\beta^3}{\beta - 1}\right)^{(n-2)}, \tag{13}$$

and in particular, as $\frac{\beta^3}{\beta - 1}$ is minimized by $\beta = 3/2$,

$$b(n) \leq \left(\frac{27}{4}\right)^{n-2}, \quad \forall n \geq 3.$$

We define generating functions for $a(n, k)$ and $b(n)$ as (omitting initial terms)

$$A(x, y) = \sum_{n \geq 4} \left(\sum_{k=0}^{n-2} a(n, k) y^k \right) x^n, \quad B(x) = \sum_{n \geq 4} b(n) x^n = A(x, 1). \tag{14}$$

Hence from these recurrences,

$$\begin{aligned} A(x, y) &= x^4 + \sum_{n \geq 4} a(n - 1, 0) x^n + \sum_{n \geq 4} \sum_{k=0}^{n-4} a(n - 1, k + 1) x^n y^k \\ &\quad + \sum_{n \geq 4} \sum_{k=1}^{n-2} a(n, k - 1) x^n y^k \\ &= x^4 + x A(x, 0) + \frac{x}{y} (A(x, y) - A(x, 0)) \\ &\quad + y A(x, y) - x (A(x, y) + x^3 y^3). \end{aligned}$$

Now grouping the terms involving $A(x, y)$, we obtain the functional equation

$$A(x, y)(y - x - y^2) = x^4 y(1 - y^3) + x(y - 1)A(x, 0) - xyA(x, y). \tag{15}$$

We now seek a solution to Equation 15 by applying the ‘kernel’ method (Odlyzko, 1995). The kernel of Equation 15 is the polynomial

$$K(x, y) = y - x - y^2.$$

If x and y are such that $K(x, y)$ vanishes while $A(x, y)$ is analytic, then the right side of Equation 15 must also vanish.

The roots of $K(x, y) = 0$ are $y = r_{1,2}(x)$, where

$$r_1(x) = \frac{1 + \sqrt{1 - 4x}}{2}$$

and

$$r_2(x) = \frac{1 - \sqrt{1 - 4x}}{2}.$$

Both paths $(x, r_{1,2}(x))$ enter the domain of Equation 14. Noting that $r_1 + r_2 = 1$ and $r_1 r_2 = x$, and writing $B(x) := A(x, 1)$,

$$x^3(1 - xr_i) + x(r_i - 1)A(x, 0) - xr_i B(xr_i) = 0$$

for $i = 1, 2$. Eliminating $A(x, 0)$ gives

$$r_1^2 B(xr_1) - r_2^2 B(xr_2) = (x^2 - x^3)\sqrt{1 - 4x},$$

which implies

$$B(xr_1) = r_2^2(1 - x)\sqrt{1 - 4x} + \left(\frac{r_2}{r_1}\right)^2 B(xr_2),$$

a functional equation to be solved for $B(x)$.

Rearranging,

$$B \circ R_1 = g + f \cdot B \circ R_2$$

where

$$\begin{aligned} R_1(x) &= xr_1(x), \quad R_2(x) = xr_2(x), \\ g(x) &= r_1^2(1 - x)\sqrt{1 - 4x}, \quad f(x) = \left(\frac{r_2}{r_1}\right)^2. \end{aligned}$$

That is,

$$B = G + F \cdot B \circ R$$

where $G = g \circ R_1^{-1}, F = f \circ R_1^{-1}$, and $R = R_2 \circ R_1^{-1}$.

Iterating the functional equation gives

$$B = \sum_{n=0}^{\infty} B_n,$$

where $B_0 = G$ and $B_n = F \cdot B_{n-1} \circ R$. Inspecting the graphs of R_1 and R_2 shows that for $0 < x < 4/27$, $\lim_{n \rightarrow \infty} R^{(n)}(x) = 0$ and hence one can check that $\lim_{n \rightarrow \infty} B_n(x) = 0$ also. In fact, $F(x) = \mathcal{O}[x^3]$ and $R(x) = \mathcal{O}[x^2]$ as $x \rightarrow 0$, so $B_n(x) \rightarrow 0$ quadratically.

The singularity of $R_1^{-1}(x)$ closest to the origin is the unique zero of $R_1(x)$, and this occurs at $x = 2/9$, $R_1 = 4/27$. Near there,

$$R_1(x) \sim \frac{4}{27} - 9\left(x - \frac{2}{9}\right)^2,$$

so

$$R_1^{-1}(x) = \frac{2}{9} - \frac{1}{3}\epsilon + \mathcal{O}(\epsilon^2),$$

where $\epsilon = (\frac{4}{27} - x)^{1/2}$. Thus each term $B_n(x)$ has a square root singularity at $x = 4/27$. Expanding for small ϵ , we find

$$R = \frac{2}{27} - \frac{1}{3}\epsilon + \mathcal{O}(\epsilon^2)$$

$$F = \frac{1}{4} - \frac{9}{4}\epsilon + \mathcal{O}(\epsilon^2)$$

$$G = \frac{7}{243} + \frac{1}{81}\epsilon + \mathcal{O}(\epsilon^2).$$

Thus, for small ϵ , B_n is evaluated near $x = 2/27$, and

$$\begin{aligned} B_n &= F \cdot B_{n-1} \circ R \\ &= \left(\frac{1}{4} - \frac{9}{4}\epsilon\right) \left(B_{n-1}\left(\frac{2}{27}\right) - \frac{1}{3}B'_{n-1}\left(\frac{2}{27}\right)\epsilon\right) + \mathcal{O}(\epsilon^2) \\ &= \frac{1}{4}B_{n-1}\left(\frac{2}{27}\right) - \left(\frac{9}{4}B_{n-1}\left(\frac{2}{27}\right) + \frac{1}{12}B'_{n-1}\left(\frac{2}{27}\right)\right)\epsilon + \mathcal{O}(\epsilon^2), \end{aligned}$$

where

$$B'_n = F \cdot R' \cdot B'_{n-1} \circ R + F' \cdot B_{n-1} \circ R.$$

From this, the coefficients c_n of ϵ in B_n are computed to be

$$c_0 = \frac{1}{81} \simeq 0.01234567901,$$

$$c_1 \simeq -0.02788743302,$$

$$c_2 \simeq -5.455702743 \times 10^{-6},$$

$$c_3 \simeq -3.328515425 \times 10^{-14}.$$

That is,

$$\begin{aligned} B(x) &= \text{const.} + c\left(\frac{4}{27} - x\right)^{1/2} + \mathcal{O}\left(\frac{4}{27} - x\right) \\ &= \text{const.} + c\sqrt{\frac{4}{27}}\sqrt{1 - \frac{27x}{4}} + \mathcal{O}\left(\frac{4}{27} - x\right), \end{aligned}$$

where $c = \sum_{n=0}^{\infty} c_n$. Recalling that the coefficient of x^n in the Taylor series of $(1-x)^{1/2}$ is asymptotic to $-n^{-3/2}/(2\sqrt{\pi})$, the coefficient of x^n in $B(x)$ is asymptotic to

$$d\left(\frac{27}{4}\right)^n n^{-3/2},$$

where

$$d = -\frac{c}{\sqrt{27\pi}} \simeq 0.001688090160.$$

Comparison with actual values of $b(n)$ supports this asymptotic behavior. A least-squares best fit of a quadratic polynomial P in $\frac{1}{n}$ to $b(n)\left(\frac{27}{4}\right)^{-n}n^{1.5}$ gives

$$P \simeq 0.001688090156 \left(1 + \frac{11.6456}{n} + \frac{43.1939}{n^2}\right),$$

which is consistent with the above asymptotic behavior.

APPENDIX 2

LINEAR-TIME RECOGNITION ALGORITHM

The algorithm inputs are the tree T and the ordered set of segments, denoted as O . The leaves of T are bijectively associated with the segments of O . Initial segments are labeled by their rank: $O = (1, 2, \dots, n)$ where n is the number of segments. During the course of the algorithm segments-leaves are removed from T and from O , but the ordering of two segments can still be computed in $\mathcal{O}(1)$ (i.e., constant time) using their numerical values. Moreover, O is double chained, so that the predecessor and successor of any given segment is also computed in $\mathcal{O}(1)$ time. For the sake of simplicity, $i-1$ and $i+1$ are the predecessor and successor of i , and "last" denotes the last element of O .

A cherry is a pair of twin segments, i.e., only one node separates these segments-leaves in T . $\text{twin}(i)$ is the twin of i (when it exists). Two cherries (i, j) and (k, l) are adjacent when $k = i + 1$ and $l = j + 1$, or $k = i - 1$ and $l = j - 1$.

A block, or partially visible duplication, is a sequence of adjacent cherries $(i, i+r), (i+1, i+1+r), \dots, (i+g, i+g+r)$, for $g < r$; i is the block start and $i+g$ is the block end. The block end might not be the last segment involved in the block, which is $i+g+r$ (see Fig. 4 for an illustration) A block is a (fully) visible duplication when $g = r - 1$. A block is memorized by a link, L , between its start and end segments, i.e., $L(i) = i+g$ and $L(i+g) = i$. So a block starting from i is a visible duplication when $\text{twin}(i) = L(i) + 1$; this enables one to test in $\mathcal{O}(1)$ whether a visible duplication has been found. Block links are detected

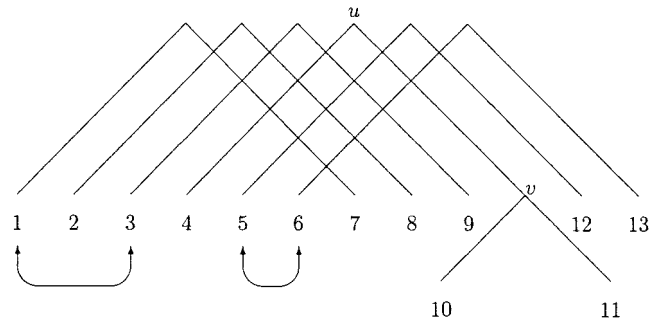


FIGURE 4. In this partial tree representation, there are six cherries: $(1, 7)$, $(2, 8)$, $(3, 9)$, $(5, 12)$, $(6, 13)$, and $(10, 11)$. The last is a simple (and visible) duplication, whereas the others form two blocks; the first block has 1 and 3 as start and end, respectively, and 5 and 6 define the second block.

and updated all along the course of the algorithm, using the Block procedure. The initial default value of $L(i)$ is i .

The basic principle of the algorithm is to iteratively reduce the leftmost visible duplication until either only three segments remains (returning "yes") or no more visible duplication can be found (returning "no"). Let $(i, i+1, i+2, \dots, i+2r-1)$ be the leftmost visible duplication; using the Reduce procedure, we remove from O and from T the r segments-leaves $(i+r, i+r+1, \dots, i+2r-1)$. In T , the common parent of $(i+j, i+r+j)$ is also removed; e.g., in Figure 4, to reduce the cherry (10, 11), 11 and v are removed and 10 is directly connected to u . Then, i cannot be on the right of the new leftmost visible duplication, but it can belong to this duplication, or be on its left. We then restart from i , first checking whether i belongs to a visible duplication.

The algorithm uses a pointer p that moves from left to right unless a visible duplication is found; then, as explained above, this duplication is reduced and p becomes equal to its leftmost remaining segment. When no reduction occurs, we continue with $p+1$, unless p is last in the order, which returns "no". The main algorithm (Dupli) and the Reduce and Block procedures are summarized below.

Dupli (p, O, T)

```

If only three segments remain, then return YES;
Else if  $p$  is equal to last+1, then return NO;
Else if  $p$  has no twin,
  then return Dupli( $p+1, O, T$ );
Else if  $p$  is one of  $(i, i+1)$  that is a cherry,
  then Reduce( $i, i, O, T$ )
  and return Dupli( $i, O, T$ );
Else let  $b = \text{Block}(p)$ ,
  If  $b$  is equal to (start, end),
    then Reduce(start, end,  $O, T$ )
    and return Dupli(start,  $O, T$ ),
  Else return Dupli( $p+1, O, T$ ).

```

Reduce (s, e, O, T)

Remove the segments $e+1, e+2, \dots, 2e-s+1$ from T and from O , and the block link between s and e : $L(s) = s$ and $L(e) = e$.

Block (p)

```

 $t = \text{twin}(p)$ ;
If  $t < p$ , then return Block( $t$ );

```

```

If  $p-1$  has a twin such that
  ( $p-1, \text{twin}(p-1)$ ) is adjacent to  $(p, t)$ ,
  then start = min( $p-1, L(p), L(p-1)$ ),
  Else start =  $p$ ;
If  $p+1$  has a twin such that
  ( $p+1, \text{twin}(p+1)$ ) is adjacent to  $(p, t)$ ,
  then end = max( $p+1, L(p), L(p+1)$ ),
  Else end =  $p$ ;
 $L(\text{start}) = \text{end}$  and  $L(\text{end}) = \text{start}$ ;
Unless  $p$  is equal to start or end,  $L(p) = p$ ;
Unless  $p-1$  is equal to start,  $L(p-1) = p-1$ ;
Unless  $p+1$  is equal to end,
   $L(p+1) = p+1$ ;
If twin(start) is equal to end+1,
  then return (start, end),
  Else return nil.

```

Illustration

We illustrate the way this algorithm proceeds using Figure 4. Dupli starts with $p = 1$ and builds a block between 1 and 2. Then $p = 2$, and this block is updated and links 1 and 3. With $p = 4$, nothing changes because 4 has no twin. Continuing, the block between 5 and 6 is built, whereas with $p = 7, 8$, and 9 nothing changes, and the block 1, 3 remains identical. With $p = 10$, a visible simple duplication is found; it is reduced by removing 11 and Dupli is called again with $p = 10$. Now, p belongs to a cherry and $\text{twin}(10) = 4$, so $\text{Block}(10)$ calls $\text{Block}(4)$; we then have $\text{start} = 1$ and $\text{end} = 6$, which corresponds to a visible multiple duplication. This duplication is reduced by removing segments 7, 8, 9, 10, 12, and 13, and Dupli is called again with $p = 1$, but only segments 1, 2, 3, 4, 5, and 6 remain.

Time Complexity

The time complexity of this algorithm with respect to n , the number of segments, is as follows.

1. Reduce requires $\mathcal{O}(r)$ to process a duplication composed of r cherries, i.e., an " r -duplication." So the total amount required by Reduce is at most $\mathcal{O}(n)$.
2. Block only uses tests and operations that are performed in $\mathcal{O}(1)$ and then requires constant times.
3. Each call of Dupli also requires constant time, so the total complexity depends on the number of recursive calls. Initially, p moves from left to right, but it returns to the left when a reduction occurs. Assume that when $p = p_1$ we find an r_1 -duplication. Then the number of calls is $\mathcal{O}(p_1)$, and $p = p_1 - r_1 + 1$. When now moving to p_2 and finding an r_2 -duplication, there are $\mathcal{O}(p_1 + p_2 - (p_1 - r_1 + 1) + 1) = \mathcal{O}(p_2 + r_1)$ calls, and $p = p_2 - 2r_2 + 1$ in the worst case. Continuing, there are: $\mathcal{O}(p_j + r_1 + 2r_2 + 2r_3 + \dots + 2r_j)$ calls at step j , which is $\mathcal{O}(n)$ for any j , because $p_j \leq n$ and $r_1 + 2r_2 + 2r_3 + \dots + 2r_j < 2n$.