New Genome Similarity Measures based on Conserved Gene Adjacencies

Luis Antonio B. Kowada[1], Daniel Doerr[2], Simone Dantas[1], and Jens Stoye[1,3]

[1] Universidade Federal Fluminense, Niterói, Brazil
[2] École Polytechnique Fédérale de Lausanne, Switzerland
[3] Faculty of Technology and Center for Biotechnology, Bielefeld University, Germany

Abstract. Many important questions in molecular biology, evolution and biomedicine can be addressed by comparative genomics approaches. One of the basic tasks when comparing genomes is the definition of measures of similarity (or dissimilarity) between two genomes, for example to elucidate the phylogenetic relationships between species.
The power of different genome comparison methods varies with the underlying formal model of a genome. The simplest models impose the strong restriction that each genome under study must contain the same genes, each in exactly one copy. More realistic models allow several copies of a gene in a genome. One speaks of gene families, and comparative genomics methods that allow this kind of input are called *gene family-based*. The most powerful – but also most complex – models avoid this preprocessing of the input data and instead integrate the family assignment within the comparative analysis. Such methods are called *gene family-free*.
In this paper, we study an intermediate approach between family-based and family-free genomic similarity measures. The model, called *gene connections*, is on the one hand more flexible than the family-based model, on the other hand the resulting data structure is less complex than in the family-free approach. This intermediate status allows us to achieve results comparable to those for family-free methods, but at running times similar to those for the family-based approach.
Within the gene connection model, we define three variants of genomic similarity measures that have different expression power. We give polynomial-time algorithms for two of them, while we show NP-hardness of the third, most powerful one. We also generalize the measures and algorithms to make them more robust against recent local disruptions in gene order. Our theoretical findings are supported by experimental results, proving the applicability and performance of our newly defined similarity measures.

1 Introduction

Many important questions in molecular biology, evolution and biomedicine can be addressed by comparative genomics approaches. One of the

basic tasks in this area is the definition of measures of similarity between two genomes. Direct applications of such measures are the computation of phylogenetic trees or the reconstruction of ancestral genomes, but also more indirect tasks like the prediction of orthologous gene pairs (derived from the same ancestor gene through speciation) or the transfer of gene function across species profit immensely from accurate genome comparison methods.

Indeed, over the past forty-or-so years, many methods have been proposed to quantify the similarity of single genes, mostly based on pairwise or multiple sequence alignments. However, in many situations similarity measures based on whole genomes are more meaningful than gene-based measures, because they give a more representative picture and are more robust against side effects such as horizontal gene transfer. Therefore, in this paper we develop and analyze methods for whole genome comparison, based on the physical structure (gene order) of the genomes.

The most simple picture of a genome is one where in a set of genomes under study orthologous genes have been identified beforehand, and only groups of orthologous genes (also known as *gene families*) are considered that have exactly one member in each genome. In this model, a variety of genomic similarity (or distance) measures have been studied and are relatively easy to compute [1,2,3,4]. However, the singleton gene family is a great oversimplification compared to what we find in nature. Therefore, more general models have been devised where several genes from the same family can exist in one genome. The computation of genomic similarities in these cases is generally much more difficult, though. In fact, many problem variants are NP-hard [5,6,7,8,9].

Another biological inaccuracy arises from the fact that a gene family assignment is not always without dispute, because orthology is usually not known but just predicted, and most prediction methods require some arbitrary threshold, deciding when two genes belong to the same family and when not. Therefore *gene family-free* measures have recently been proposed, based on pairwise similarities between genes [10,11,12,13]. While the resulting similarity measures are very promising, their computation is usually not easier than for the family-based models and therefore NP-hard as well [10,13].

In this paper, we study an intermediate approach between family-based and family-free genomic similarity measures, *gene connections*. It requires some preprocessing of the genes contained in the genomes under study, but in a less stringent way than in the family-based approach. On the other hand, the resulting data structure is less complex than in the

family-free approach, where arbitrary (real-valued) similarities between genes are considered. This intermediate status allows us to achieve results comparable to those for family-free methods, but at time complexities similar to those for the family-based approach.

The paper is structured as follows. We first define three new genome similarity measures based on conserved gene adjacencies (Section 2), followed by some pointers to related literature (Section 3). Each of the three following sections is then devoted to one of the similarity measures. We show that the first problem can be computed in polynomial time, but is biologically quite simplistic. The second one, while avoiding some of the weaknesses of the first, is NP-hard to compute and can therefore not be applied for genomes of realistic size. The third measure, finally, provides a compromise between biological relevance and computational complexity. In Section 7 we compare the results obtained with our similarity measures experimentally, using a large data set of plant (rosid) genomes. The last section concludes the paper.

The implemented algorithms used in this work as well as the studied dataset are available for download from `http://bibiserv.cebitec.uni-bielefeld.de/newdist`.

2 Basic Definitions

An *alphabet* is a finite set of *characters*. A *string* over an alphabet $\mathcal{A}$ is a sequence of characters from $\mathcal{A}$. Given a string S, $S[i]$ refers to the ith character of S and $|S|$ is the *length* of S, i.e., the number of characters in S. In a *signed string* S, each character is labeled with a sign, denoted $sgn_S(i)$ for the character at index position i. A sign is either positive $(+)$ or negative $(-)$. In comparative genomics, for example, the signs may indicate the orientations of genes on their genomic sequences, which themselves are represented as strings. Therefore in this paper we use the term *gene* as a synonym for "signed character" and the term *genome* as a synonym for "signed string".

Definition 1 (gene connection graph). *Given two genomes S and T, a* gene connection graph *$G(S,T)$ of S and T is a bipartite graph with one vertex for each gene of S and one vertex for each gene of T. An edge between two vertices, one from S and one from T, indicates that there is some* connection *between the two genes represented by these vertices.*

The term *connection* in the above definition is not very specific. Depending on the data set and context, connections may be defined based on

gene homology, sequence similarity, functional relatedness, or any other similarity measure between genes.

For ease of notation, we let $S[i]$ denote both the ith gene of genome S, as well as the vertex of G representing this gene. Similar for $T[j]$. The set of edges of a graph G is denoted by $E(G)$. The size of a graph G is the number of its edges, $|G| = |E(G)|$. Further, we define a *connection function* t that returns for an index position i of S the list $t(i)$ of index positions in T that are connected to $S[i]$ by an edge in $G(S,T)$. That is, $t(i) = [j \mid (i,j) \in E(G(S,T))$ for $1 \leq j \leq |T|]$. The function $s(j)$ for an index position of T is defined analogously.

A pair of adjacent index positions (i,i') with $i' = i+1$ in a string is called an *adjacency*. Note that this definition of adjacency only considers direct neighborhood of genes ($i' = i+1$), while all our following uses of this term refer to an extended definition given by Zhu *et al.* [14], who introduced *generalized gene adjacencies* as follows:

Definition 2 (adjacency). *Given an integer $\theta \geq 1$, a pair of index positions (i,i') with $i' \leq i+\theta$ in a string is a* (θ-) adjacency.

In other words, two genes of the same genome form a θ-adjacency if the number of genes between them is less than θ. In the following we will frequently differentiate between *simple adjacencies* ($\theta = 1$) and *generalized adjacencies* ($\theta \geq 1$).

As mentioned in the Introduction, in this paper we are interested in defining measures of similarity to compare pairs of genomes. A simple approach is based on their number of *conserved adjacencies*. Although below we will study different variants of similarities, they all use the following basic notion of conserved adjacency:

Definition 3 (conserved adjacency). *Given two genomes S and T and a gene connection graph $G(S,T)$, a pair of adjacencies (i,i') in S and (j,j') in T is called a* conserved adjacency*, denoted $(i,i'||j,j')$, if one of the following two holds:*

(a) $(i,j) \in E(G(S,T))$, $(i',j') \in E(G(S,T))$, $sgn_S(i) = sgn_T(j)$ *and* $sgn_S(i') = sgn_T(j')$*; or*
(b) $(i,j') \in E(G(S,T))$, $(i',j) \in E(G(S,T))$, $sgn_S(i) \neq sgn_T(j')$ *and* $sgn_S(i') \neq sgn_T(j)$.

For an illustration of these definitions, see Figure 1.

We further denote two conserved adjacencies as *conflicting* if their intervals in either genome are overlapping:

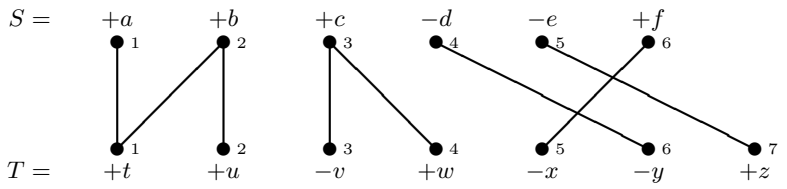

Fig. 1. Gene connection graph of two genomes $S = (+a, +b, +c, -d, -e, +f)$ (top row) and $T = (+t, +u, -v, +w, -x, -y, +z)$ (bottom row). Conserved 2-adjacencies are $(1, 2||1, 2)$, $(2, 3||2, 4)$, $(3, 4||4, 6)$ and $(5, 6||5, 7)$. Note that $(2, 3||1, 3)$, $(2, 3||2, 3)$, $(4, 5||6, 7)$ and $(4, 6||5, 6)$ are no conserved 2-adjacencies because the signs do not match the definition.

Definition 4 (conflicting conserved adjacencies). *Two conserved adjacencies* $(i, i'||j, j')$ *and* $(k, k'||l, l')$ *are* conflicting *if (1)* $(i, i'||j, j') \neq (k, k'||l, l')$ *and (2)* $[i, i' - 1] \cap [k, k' - 1] \neq \emptyset$ *or* $[j, j' - 1] \cap [l, l' - 1] \neq \emptyset$.

Subsequently a set of conserved adjacencies is denoted as *non-conflicting* if the above-defined property does not hold between any two of its members.

In the example of Figure 1, $(3, 4||4, 6)$ and $(5, 6||5, 7)$ are the only conflicting conserved adjacencies. All other pairs are non-conflicting.

The different similarity measures that we consider in this work are expressed by the following three problem statements:

Problem 1 (total adjacency model). Given two genomes S and T and a gene connection graph $G(S, T)$, count the number of pairs of index positions (i, i') in S and (j, j') in T that form a conserved adjacency. In other words, compute

$$adj(S, T) = |\{(i, i'||j, j') \mid 1 \leq i < i' \leq |S| \text{ and } 1 \leq j < j' \leq |T|\}|.$$

Because a gene connection graph $G(S, T)$ is not limited to one-to-one connections between genes of genomes S and T, solutions to Problem 1 may biologically not be very plausible. Therefore we define a second measure, motivated by the one used in [10,11], which asks for one-to-one correspondences between genes of S and T in its solutions:

Problem 2 (gene matching model). Given two genomes S and T, a gene connection graph $G(S, T)$ and a real-valued parameter $\alpha \in [0, 1]$, find a

bipartite matching M in $G(S,T)$ such that the induced sequences S^M and T^M maximize the measure

$$\mathcal{F}_\alpha(M) = \alpha \cdot adj(S^M, T^M) + (1-\alpha) \cdot edg(M),$$

where $edg(M) = |M|$ is the size of matching M. (The induced sequences S^M and T^M are the subsequences of S and T, respectively, that contain those characters incident to edges of M.)

As we will see later in this paper, solving Problem 2 is NP-hard even for simple adjacencies. Therefore we define a third, intermediate measure, which is more efficient to compute in practice, while producing one-to-one correspondences between gene extremities. It is defined as the size of the largest subset of non-conflicting conserved adjacencies found in a pair of genomes:

Problem 3 (adjacency matching model). Given two genomes S and T and a gene connection graph $G(S,T)$, let C be the set of conserved adjacencies between S and T. Compute the size $|C^\star|$ of a maximum cardinality set of non-conflicting conserved adjacencies $C^\star \subseteq C$.

3 Related Work

As mentioned above, the *gene connection graph* input format that we propose here is an intermediate between gene families and the family-free model. Indeed, we do not require the gene connection graph to be transitive, which is the main difference to the *gene family graph*, where vertices are assigned to genes and edges are drawn between genes from different genomes whenever they belong to the same family, thus forming bipartite cliques. (This graph has not been introduced under this name in the literature, but is implicitly mentioned already in [15] and later more explicitly in [10].) On the other end, the *gene similarity graph* [11] is a weighted version of the gene connection graph, increasing the expression power by its ability to represent different strengths of gene connections.

The only previous use of such an intermediate model in comparative genomics that we are aware of is in the form of *indeterminate strings* in [12].

Definition 5 (indeterminate string, signed indeterminate string). *Given an alphabet $\mathcal{A}$, a string S over the power set $\mathcal{P}(\mathcal{A}) \setminus \{\emptyset\}$ is called an* indeterminate string *over $\mathcal{A}$. In other words, for $1 \leq i \leq n$, $\emptyset \neq S[i] \subseteq \mathcal{A}$. In a* signed indeterminate string *S, any index position i has a sign $sgn_S(i)$, which therefore is the same for all characters at that position.*

Given two genomes S and T and a gene connection graph $G(S,T)$, it is easy to create a pair of signed indeterminate strings S' and T' over an alphabet $\mathcal{A}'$ that contain the same set of conserved adjacencies as S and T: For any edge $e = (S[i], T[j])$ of $G(S,T)$, create one symbol $e' \in \mathcal{A}'$ and let $e' \in S'[i]$ and $e' \in T'[j]$. The signs are just transferred from S and T to S' and T', respectively: $sgn_{S'}[i] = sgn_S[i]$ for all i, $1 \leq i \leq |S|$, and $sgn_{T'}[j] = sgn_T[j]$ for all j, $1 \leq j \leq |T|$.

Conversely, given two indeterminate strings S' and T', we can easily create sequences S and T and the corresponding gene connection graph with the same set of conserved adjacencies. In order to do this, let $\mathcal{A} = \{1, 2, \ldots, |S'|, 1', 2', \ldots, |T'|'\}$, set $S = sgn_{S'[1]}1, \ldots, sgn_{S'[|S'|]}|S'|$, $T = sgn_{T'[1]}1', \ldots, sgn_{T'[|T'|]}|T'|'$, and create in $G(S,T)$ an edge $e = (S[i], T[j])$ whenever $S'[i] \cap T'[j] \neq \emptyset$.

Clearly, all the information about conserved adjacencies between these two representations is identical, while sometimes the graph representation and sometimes the representation as signed indeterminate string is more concise.

Indeterminate strings in [12] were used to identify regions of common gene content (*gene clusters*) in two genomes, which is important in functional genomics. Here our focus is on conserved adjacencies (which can be seen as small clusters of just two genes) for defining whole-genome similarities. Similar measures are known for singleton gene families as the *breakpoint distance* [16,17], have been extended to gene families in [15,5,7] and were defined for the family-free model in [10].

4 An Optimal Solution for Problem 1

In order to solve Problem 1, we construct a list L of edges of $G(S,T)$ using connection function $t(i)$ for $1 \leq i \leq |S|$. In doing so, we assume that the elements of $t(i)$, $1 \leq i \leq |S|$, are sorted in increasing order. If this is not given as input, it can always be achieved by applying counting sort to all lists $t(i)$ in overall $O(|S| + |T| + |G(S,T)|)$ time, which is proportional to the input size.

We present with Algorithm 1 a solution to Problem 1 for simple adjacencies and subsequently extend this approach for the generalized case. Our algorithm is a simple, linear time procedure which uses three pointers e, e', e'' into list L. These pointers simultaneously traverse L while reporting any pair of adjacent parallel edges (e, e') or crossing edges (e, e'').

Correctness. Given a pair $(i,j) \in L$, there are overall four cases for the signs of index i in S and index j in T, each with two sub-cases for the

Algorithm 1

Input: genomes S and T, gene connection graph $G(S,T)$

Create a list L of all edges $(i,j) \in E(G(S,T))$ ordered by primary index i and secondary index j
Let $e' = (i',j')$ and $e'' = (i'',j'')$ point to the second element of L
for each element $e = (i,j)$ of L in sorted order **do**
  if $sgn_S(i) = sgn_T(j)$ **then**
    while $i' < i+1$ **or** $(i' = i+1$ **and** $j' < j+1)$ **do**
      advance $e' = (i',j')$ by one step in L
    end while
    if $(i',j') = (i+1, j+1)$ **and** $sgn_S(i') = sgn_T(j')$ **then**
      report the conserved adjacency $(i, i'||j, j')$
    end if
  else
    while $i'' < i+1$ **or** $(i'' = i+1$ **and** $j'' < j-1)$ **do**
      advance $e'' = (i'',j'')$ by one step in L
    end while
    if $(i'',j'') = (i+1, j-1)$ **and** $sgn_S(i'') \neq sgn_T(j'')$ **then**
      report the conserved adjacency $(i, i''||j'', j)$
    end if
  end if
end for

signs of index $i+1$ in S and index $j+1$ or index $j-1$ in T, listed in the following.

(1) If $sgn_S(i) = +$ and $sgn_T(j) = +$, then we have a conserved adjacency $(i, i{+}1||j, j{+}1)$ if and only if $(i{+}1, j{+}1) \in L$ and either $sgn_S(i{+}1) = +$ and $sgn_T(j+1) = +$ or $sgn_S(i+1) = -$ and $sgn_T(j+1) = -$.
(2) If $sgn_S(i) = +$ and $sgn_T(j) = -$, then we have a conserved adjacency $(i, i{+}1||j{-}1, j)$ if and only if $(i{+}1, j{-}1) \in L$ and either $sgn_S(i{+}1) = +$ and $sgn_T(j-1) = -$ or $sgn_S(i+1) = -$ and $sgn_T(j-1) = +$.
(3) If $sgn_S(i) = -$ and $sgn_T(j) = +$, then we have a conserved adjacency $(i, i{+}1||j{-}1, j)$ if and only if $(i{+}1, j{-}1) \in L$ and either $sgn_S(i{+}1) = -$ and $sgn_T(j-1) = +$ or $sgn_S(i+1) = +$ and $sgn_T(j-1) = -$.
(4) If $sgn_S(i) = -$ and $sgn_T(j) = -$, then we have a conserved adjacency $(i, i{+}1||j, j{+}1)$ if and only if $(i{+}1, j{+}1) \in L$ and either $sgn_S(i{+}1) = -$ and $sgn_T(j+1) = -$ or $sgn_S(i+1) = +$ and $sgn_T(j+1) = +$.

Clearly, cases 1 and 4 and cases 2 and 3 can be summarized to the two cases given in Algorithm 1.

Runtime analysis. The list L has length $|G(S,T)|$ and can be constructed and sorted in linear time $O(|S|+|T|+|G(S,T)|)$, as discussed above. Each of the three edge pointers e, e' and e'' traverses L once from the beginning

to the end, so that the **for** loop in lines 3–19 takes $O(|L|)$ time. Therefore the overall running time is $O(|S| + |T| + |G(S,T)|)$.

Space analysis. The algorithm needs space only for the two input strings S and T, the list L and some constant-space variables. Therefore the space usage is of order $O(|S| + |T| + |G(S,T)|)$.

Extension to generalized adjacencies. Algorithm 1' solves Problem 1 for generalized adjacencies. Following the same strategy as Algorithm 1, the extension requires next to the main pointer e additional 2θ pointers into list L that are denoted e'_t and e''_t, $1 \leq t \leq \theta$. While it traverses through each element (i,j) in the list using pointer e, each pointer e'_t, $1 \leq t \leq \theta$, is subsequently increased to point to the smallest element larger than or equal to $(i+t, j+1)$ in L. A copy $\hat{e}$ of pointer e'_t is then used to find candidates $(i+t, j+1), \ldots, (i+t, j+\theta)$. Likewise, pointers e''_t, $1 \leq t \leq \theta$, are incremented to the smallest element larger than or equal to $(i+t, j-\theta)$, whereupon copy $\hat{e}$ of e''_t is used to find candidates $(i+t, j-\theta), \ldots, (i+t, j-1)$.

All pointers e, e'_t, and e''_t, $1 \leq t \leq \theta$ are continuously increased, thus each traversing L once. Any instance of pointer $\hat{e}$ visits at most θ elements in each iteration, thus leading to an overall running time of $O(\theta^2|G(S,T)|)$. The running time is asymptotically optimal in the sense of worst case analysis, since there can be just as many θ-adjacencies in graph $G(S,T)$. Algorithm 1' requires $O(\theta + |S| + |T| + \theta^2|G(S,T)|)$ space.

5 Complexity of Problem 2

While one may hope that the intermediate status of the gene connection graph between the gene family graph and the gene similarity graph allows more efficient algorithms than for the more complex gene similarity graph, this is not the case for the gene matching model.

Only for $\alpha = 0$, we have $\mathcal{F}_\alpha(M) = edg(M) = |M|$ and therefore Problem 2 reduces to computing a maximum bipartite matching, which is possible in polynomial time [18]. However, this case is not very interesting because it completely ignores conserved adjacencies and just compares the gene content of the two genomes. All interesting cases are more difficult to solve, as the following theorem shows:[4]

Theorem 1. *Problem 2 is NP-hard for* $0 < \alpha \leq 1$.

[4] A weaker result, namely the NP-hardness of Problem 2 for values of α between 0 and 1/3, can be found in [19].

Algorithm 1'

Input: genomes S and T, gene connection graph $G(S,T)$, gap threshold θ

Create a list L of all edges $(i,j) \in E(G(S,T))$ ordered by primary index i and secondary index j
Let $e'_t = (i'_t, j'_t)$ and $e''_t = (i'_t, j'_t)$, $1 \le t \le \theta$, point to the second element of L
for each element $e = (i,j)$ of L in sorted order **do**
  if $sgn_S(i) = sgn_T(j)$ **then**
    for each $e'_t = (i'_t, j'_t)$, $1 \le t \le \theta$ **do**
      while $i'_t < i + t$ **or** $(i'_t = i + t$ **and** $j'_t < j + 1)$ **do**
        advance $e'_t = (i'_t, j'_t)$ by one step in L
      end while
      let $\hat{e} = (\hat{\imath}, \hat{\jmath}) \leftarrow e'_t$
      while $\hat{\imath} = i + t$ **and** $\hat{\jmath} \le j + \theta$ **do**
        if $sgn_S(\hat{\imath}) = sgn_T(\hat{\jmath})$ **then**
          report the conserved adjacency $(i, \hat{\imath} || j, \hat{\jmath})$
        end if
        advance $\hat{e} = (\hat{\imath}, \hat{\jmath})$ by one step in L
      end while
    end for
  else
    for each $e''_t = (i''_t, j''_t)$, $1 \le t \le \theta$ **do**
      while $i''_t < i + t$ **or** $(i''_t = i + t$ **and** $j''_t < j - \theta)$ **do**
        advance $e''_t = (i''_t, j''_t)$ by one step in L
      end while
      let $\hat{e} = (\hat{\imath}, \hat{\jmath}) \leftarrow e''_t$
      while $\hat{\imath} = i + t$ **and** $\hat{\jmath} < j - 1$ **do**
        if $sgn_S(\hat{\imath}) \neq sgn_T(\hat{\jmath})$ **then**
          report the conserved adjacency $(i, \hat{\imath} || \hat{\jmath}, j)$
        end if
        advance $\hat{e} = (\hat{\imath}, \hat{\jmath})$ by one step in L
      end while
    end for
  end if
end for

Proof. We will focus on simple adjacencies ($\theta = 1$), as this is sufficient to prove Theorem 1. Inspired by the proof of Bryant [5] for the family-based case, we provide a P-reduction from Vertex Cover: Given a graph $\mathcal{G} = (V, E)$ and an integer λ, does there exist a subset $V' \subseteq V$ such that $|V'| = \lambda$ and each edge in E is adjacent to at least one vertex in V'?

Our reduction transforms an instance of Vertex Cover into an instance of the decision version of Problem 2: Given strings S and T, a gene connection graph $G(S,T)$, a real value α, $0 < \alpha \leq 1$, and a real value $F \geq 0$, does there exist a bipartite matching M in $G(S,T)$ such that $\mathcal{F}_\alpha(M) \geq F$?

Let $\mathcal{G} = (V, E)$ and λ be an instance of Vertex Cover with $V = \{v_1, v_2, \ldots, v_n\}$ and $E = \{e_1, e_2, \ldots, e_m\}$. Then we construct an alphabet $\mathcal{A}$ of size $2n + 4m + 2$ given by

$$\mathcal{A} = V \cup \{v'_i \mid v_i \in V\} \cup E \cup \{e'_i \mid e_i \in E\} \cup \{x_i, x'_i \mid 1 \leq i \leq m+1\}.$$

The two genomes S and T are constructed as follows:

$$S = v_1 v'_1 v_2 v'_2 \ldots v_n v'_n x_1 x'_1 e_1 e'_1 x_2 x'_2 e_2 e'_2 x_3 x'_3 \ldots x_m x'_m e_m e'_m x_{m+1} x'_{m+1}$$

and

$$T = x_{m+1} x'_{m+1} x_m x'_m \ldots x_2 x'_2 x_1 x'_1 v_n \mathcal{E}_n v'_n v_{n-1} \mathcal{E}_{n-1} v'_{n-1} \ldots v_1 \mathcal{E}_1 v'_1$$

where $\mathcal{E}_i$ is a string of the symbol pairs $e_j e'_j$ for the edges e_j that are adjacent to v_i. The gene connection graph $G(S,T)$ has an edge for each pair of identical symbols $S[i]$ and $T[j]$. The parameter α may be chosen arbitrarily within the range $0 < \alpha \leq 1$.

First, we show that among the matchings maximizing the value $\mathcal{F}_\alpha$ for this problem, there is always at least one which is a maximal matching. Let M be a non-maximal matching in $G(S,T)$ maximizing $\mathcal{F}_\alpha$ and consider an edge $\ell \notin M$ that may be added to M, forming a new matching $M' = M \cup \{\ell\}$. Clearly, ℓ can dismiss at most two adjacencies of M in M', so $adj(M') \geq adj(M) - 2$. But in our construction, where the symbols of $\mathcal{A}$ (except the e_i and e'_i) are in reverse order in S related to T, and furthermore each e_i and each e'_i is between x_i and x_{i+1} in S, any new edge ℓ added to M can dismiss at most one adjacency: If ℓ is adjacent to a symbol a and the symbol a' is adjacent to another edge $\ell' \in M$ (or vice-versa) then $adj(M') = adj(M) + 1$. Moreover, if two partner edges $\ell, \ell' \notin M$ are added to M and thus $M' = M \cup \{\ell, \ell'\}$, then $adj(M') \geq adj(M)$ and $edg(M') = edg(M) + 2$. Therefore $\mathcal{F}_\alpha(M') > \mathcal{F}_\alpha(M)$ for $\alpha < 1$ and $\mathcal{F}_\alpha(M') \geq \mathcal{F}_\alpha(M)$ for $\alpha = 1$.

Next, we show that there is a vertex cover of size λ for a graph $\mathcal{G}$ if and only if Problem 2 has a solution with $F = \alpha(2m + 1 + (n - \lambda)) + (1 - \alpha)(2n + 4m + 2)$. Note that by construction of S, T and $G(S, T)$, conserved adjacencies in a maximal matching are only possible between pairs of the same symbol of $\mathcal{A}$, i.e. $v_i v_i'$, $e_i e_i'$ or $x_i x_i'$. Therefore we can simplify the notation and represent an adjacency $(i, i'||j, j')$ by the pair of elements in S, $S[i]S[i']$. Clearly, any maximal matching of $G(S, T)$ has $|S| = 2n + 4m + 2$ edges. Moreover, any maximal matching realizes at least the $2m + 1$ conserved adjacencies $e_i e_i'$ and $x_i x_i'$. The other possible adjacencies are the $v_i v_i'$. If there exists a solution with value $F = \alpha(2m + 1 + (n - \lambda)) + (1 - \alpha)|S|$, then there are at least $n - \lambda$ adjacencies involving $v_i v_i'$. These adjacencies are possible if the respective edges of $\mathcal{G}$ are covered by λ vertices. If we do not have a solution with value F, then $\mathcal{G}$ does not have a vertex cover of size λ. □

Solving Problem 2 for simple adjacencies, we make use of a method described in [19], that was originally developed for solving the gene family-free variant of Problem 2. In doing so, it constructs an integer linear program (ILP) similar to program `FFAdj-Int` described in [10]. It includes a preprocessing algorithm that identifies small components in gene similarity graphs which are part of an optimal solution. This approach enables the computation of optimal solutions for small and medium-sized gene similarity graphs. However, as the method is specifically tailored for gene family-free analysis, it does not perform very efficiently on gene connection graphs, as we will see in Section 7. We refer to this ILP and its preprocessing step as Algorithm 2.

We further believe it will be difficult to develop a practical algorithm solving Problem 2 for generalized adjacencies.

6 Computing Exact Solutions for Problem 3

We present a polynomial time algorithm solving Problem 3 for simple adjacencies which makes use of the following graph structure:

Definition 6 (conserved adjacencies graph). *Given two genomes S and T and a set $C = \{(i_1, i_1'||j_1, j_1'), \ldots, (i_n, i_n'||j_n, j_n')\}$ of conserved adjacencies between S and T, the* conserved adjacencies graph $A_C(S, T)$ *is a bipartite graph with one vertex for each gene adjacency (i, i') of S that occurs in C and one vertex for each gene adjacency (j, j') of T that occurs in C. The edges correspond to the conserved adjacencies in C.*

Pseudocode of our algorithm is shown in Algorithm 3. Clearly its running time is dominated by the time to compute a maximum matching in line 3, which in unweighted bipartite graphs with n vertices and m edges is possible in $O(m\sqrt{n})$ time [18]. In our case $n \leq |S| + |T| - 2$ and $m \leq n^2$, therefore Algorithm 3 takes overall $O((|S| + |T|)^{5/2})$ time.

Algorithm 3

Input: genomes S and T, gene connection graph $G(S, T)$

1: Let C be the set of conserved adjacencies reported by Algorithm 1 applied to S, T and $G(S, T)$
2: Construct the conserved adjacencies graph $A = A_C(S, T)$
3: Compute a maximum bipartite matching M on A
4: return $|M|$

Extension to generalized adjacencies. Other than for the first two problems, the properties of Problem 3 change drastically when generalized adjacencies are considered. Because a θ-adjacency corresponds to an interval of up to $\theta + 1$ consecutive genes, the intervals of two θ-adjacencies for $\theta \geq 2$ can overlap on more than two genes, or even be contained in one another. The complexity of Problem 3 for $\theta \geq 2$ remains an open question.

Solving Problem 3 for generalized adjacencies, we propose Algorithm 3' that follows the same strategy as its counterpart for simple adjacencies. However, while for the latter it was possible to find a maximum subset of non-conflicting θ-adjacencies using a maximum matching approach, here we propose an ILP, described in Figure 2. The ILP makes use of two types of binary variables, $\mathbf{a}(i, j)$ for each edge (i, j) in the gene connection graph $G(S, T)$, and $\mathbf{b}(i, i', j, j')$ for each θ-adjacency $(i, i'||j, j')$ in C_θ. We say that a binary variable is *saturated* if it is assigned value 1. While maximizing the number of saturated $\mathbf{b}(.)$ variables (which represents the output of the program), our ILP imposes matching constraints (`C.01`) for the set of edges in selected θ-adjacencies. Further constraints (`C.02`) ensure that for each θ-adjacency $(i, i'||j, j')$ (a) both edges between its corresponding genes are saturated and (b) no saturated edge is incident to a gene in interval $[i+1, i'-1]$ of genome S (i.e. a possibly empty interval corresponding to all genes between i and i') and interval $[j+1, j'-1]$ of genome T, respectively.

Algorithm 3'

Input: genomes S and T, gene connection graph $G(S,T)$, gap threshold θ

1: Let C_θ be the set of conserved adjacencies reported by Algorithm 1' applied to S, T and $G(S,T)$
2: Compute a maximum cardinality set of non-conflicting conserved θ-adjacencies $C_\theta^\star \subseteq C_\theta$ using the ILP given in Figure 2
3: return $|C_\theta^\star|$

ILP solving Step 2 in Algorithm 3'

Objective:

$$\texttt{maximize} \sum_{(i,i'||j,j'')\in C_\theta} \mathbf{b}(i,i',j,j')$$

Constraints:

(C.01) **for each** $i \leftarrow 1$ **to** $|S|$, $\sum_{j\in t(i)} \mathbf{a}(i,j) \leq 1$

for each $j \leftarrow 1$ **to** $|T|$, $\sum_{i\in s(j)} \mathbf{a}(i,j) \leq 1$

(C.02) **for each** $(i,i'||j,j') \in C_\theta$

if $sgn_S(i) = sgn_S(i')$ **then**
$2\cdot\mathbf{b}(i,i',j,j') - \mathbf{a}(i,j) - \mathbf{a}(i',j') \leq 0$
otherwise
$2\cdot\mathbf{b}(i,i',j,j') - \mathbf{a}(i,j') - \mathbf{a}(i',j) \leq 0$
end if

for each $\hat{\imath} \leftarrow [i+1, i'-1]$ **and each** $\hat{\jmath}$ **in** $t(\hat{\imath})$
$\mathbf{b}(i,i',j,j') + \mathbf{a}(\hat{\imath},\hat{\jmath}) \leq 1$
for each $\hat{\jmath} \leftarrow [j+1, j'-1]$ **and each** $\hat{\imath}$ **in** $s(\hat{\jmath})$
$\mathbf{b}(i,i',j,j') + \mathbf{a}(\hat{\imath},\hat{\jmath}) \leq 1$

end for

Domains:

(D.01) **for each** $(i,j) \in E(G(S,T))$, $\mathbf{a}(i,j) \in \{0,1\}$

(D.02) **for each** $(i,i'||j,j') \in C_\theta$, $\mathbf{b}(i,i',j,j') \in \{0,1\}$

Fig. 2. Integer linear program for finding a maximum subset of non-conflicting conserved adjacencies of a given set C_θ.

7 Experimental Results

Genomic dataset. We study genomes of 18 rosid species (see Table 1). Rosids are a prominent subclass of flowering plants to which also many agricultural crops belong. The genomic sequences of the studied species were obtained from *Phytozyme* [20][5], an online resource of the Joint Genome Institute providing databases and tools for comparative genomics analyses of plant genomes. Most of the studied plant genomes are partially assembled, comprising up to 5,000 scaffolds covering one or more annotated protein coding genes. While the smallest genome in our data set contains roughly 24,500 genes, the largest spans with 56,000 genes more than twice as many. Rosids, just like many other plants, met their evolutionary fate through multiple events of whole genome duplication, followed by periods of fractionation, in which many duplicated genes were lost again.

Construction of gene connection and gene family graphs. Next to the genomic sequences and gene annotations, Phytozyme also provides gene family information in form of co-orthologous clusters computed by InParanoid [21]. InParanoid follows a seed-based strategy by identifying pairs of orthologous genes (the "seeds") through reciprocal best BLASTP hits. These are subsequently used to recruit inparalogs, eventually forming groups of co-orthologous genes.

We ran BLASTP on all genes of our dataset using an e-value threshold of 10^{-5} and otherwise default parameter settings. We then constructed gene connection graphs for all 153 genome pairs by establishing edges between vertices whose corresponding genes share reciprocal BLASTP hits. We refer to these graphs as *BLASTP GC graphs*. Similarly, we constructed pairwise gene family graphs using InParanoid's homology assignment, which we refer to as *InParanoid GF graphs*.

Unsurprisingly, the BLASTP GC graphs are much larger in size than the InParanoid GF graphs. We observed average sizes of 150,000 edges for the former, whereas the latter graphs had on average only one fifth of this size. Moreover, only 4% of edges in InParanoid GF graphs were not contained in their BLASTP GC counterparts. Lacking ground truth of homologies in our dataset, we take a conservative stance by assuming that InParanoid's homology assignment can be considered true, or, in other words, that it contains only a negligible number of false positives. However, we conclude from a previous study [38], in which InParanoid

[5] The described experiments were performed on data sets of Phytozyme v10.3.

species	version	# genes	# scaffolds	reference
A. thaliana	TAIR10	27,416	7	[22]
B. rapa	FPSc v1.3	40,492	669	[20]
B. stricta	v1.2	27,416	854	[20]
C. clementina	v1.0	24,533	94	[23]
C. rubella	v1.0	26,521	123	[24]
E. grandis	v1.1	36,376	1,315	[25]
E. salsugineum	v1.0	26,351	61	[26]
F. vesca	v1.1	32,831	8	[27]
G. max	Wm82.a2	56,044	147	[28]
G. raimondii	v2.1	37,505	133	[29]
L. usitatissimum	v1.0	43,471	1,028	[30]
M. truncatula	Mt4.0v1	50,894	1,033	[31]
P. persica	v1.0	27,864	59	[32]
P. trichocarpa	v3.0	41,335	379	[33]
P. vulgaris	v1.0	27,197	91	[34]
R. communis	v0.1	31,221	4,962	[35]
T. cacao	v1.1	29,452	99	[36]
V. vinifera	Genoscope.12X	26,346	33	[37]

Table 1. The genomic dataset of 18 rosid species used in our experiments.

(as well as all other gene family prediction tools in that study) exhibited a poor recall, that the homology assignment may be incomplete. That being said, we regard the edges of BLASTP GC graphs with suspicion. In doing so, we assume many of them leading to false positive homology assignments. We perform subsequent analysis to outline a possible procedure of identifying additional potential homologies that are supported by conservation in gene order in BLASTP GC graphs.

Implementation. All computations were performed on a Linux machine using a single 2.3 GHz CPU. We implemented Algorithms 1, 1', 3, and 3' in Python. For Algorithm 2 we used the implementation of [19]. In Algorithm 3, the maximum cardinality matching was computed using an implementation of Hopcroft and Karp's algorithm [18] provided by the Python-based NetworkX[6] library. The ILPs of Algorithms 2 and 3' were run using CPLEX[7], a solver for various types of linear and quadratic programs.

Runtimes. The runtimes of Algorithms 1 and 3 are shown in Figure 3 (left). The runtime analysis was repeated 5 times and is visualized by whisker plots. For each of the 153 BLASTP GC graphs in our dataset,

[6] `http://networkx.github.io/`
[7] `http://www.ibm.com/software/integration/optimization/cplex-optimizer/`

the computation was finished in less than 50 CPU seconds. Moreover, our evaluation reveals that the enumeration of the set of conserved adjacencies in our dataset requires often more time than the subsequent computation of the maximum matching for Algorithm 3. The plot on the right side of Figure 3 shows that the runtimes of Algorithm 1' for $\theta = 2, 3, 4$ increase only moderately for higher values of θ.

Comparing our methods to the gene family-free approach, an implementation of a heuristic method described in [10] failed to return a result for the gene family free variant of Problem 2 on the BLASTP GC graph of *R. communis* and *V. vinifera* within 36 hours of computation. Surprisingly, running Algorithm 2 with $\alpha = 0.1$ just as long, we were able to obtain a suboptimal solution of which CPLEX reported an optimality gap of only 1.89%. Nevertheless, as a reference for comparison with our various models it would be even more informative to have optimal solutions of these problems. We leave it as an open problem whether it is possible to improve our ILPs in order to achieve this.

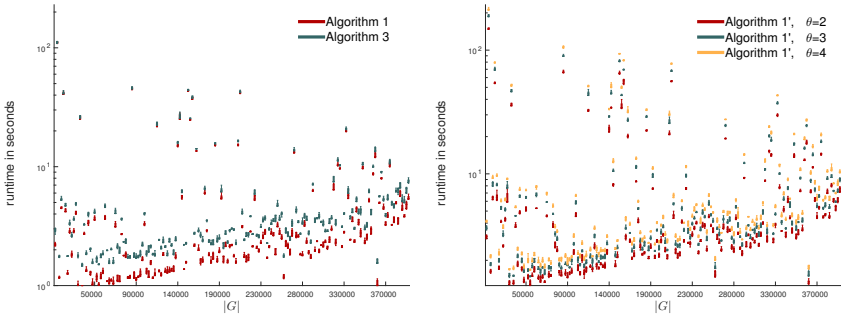

Fig. 3. Left: Runtimes of Algorithms 1 and 3 for all 153 BLASTP GC graphs of the studied dataset. Right: Runtimes of Algorithm 1' for $\theta = 2, 3, 4$.

Further, we were able to compute exact results for Problem 3 and $\theta = 2$ with Algorithm 3' for all 153 but 19 BLASTP GC graphs and all but 16 InParanoid GC graphs, limiting computation time to two hours per graph instance.

Gene connection vs. gene family graphs. The overlap between the set of conserved simple adjacencies identified in BLASTP GC graphs and in InParanoid GF graphs is visualized in the left plot of Figure 4. Overall, 70% of the conserved adjacencies of the InParanoid GF graphs were also

found in the BLASTP GC graphs whereas we find in the latter 90% more conserved adjacencies than in the former. Investigating the high number of InParanoid adjacencies that are missing in BLASTP GC graphs, we discovered that many generalized adjacencies of the former span genes that are connected (and therefore breaking the surrounding adjacency) in their BLASTP GC counterparts. However, the mean number of connected intervening genes was only 1.4. In fact, the overlap of 2-adjacencies in BLASTP GC graphs with 1-adjacencies of InParanoid GF graphs was at 83% of all adjacencies in the latter (Figure 4, right plot).

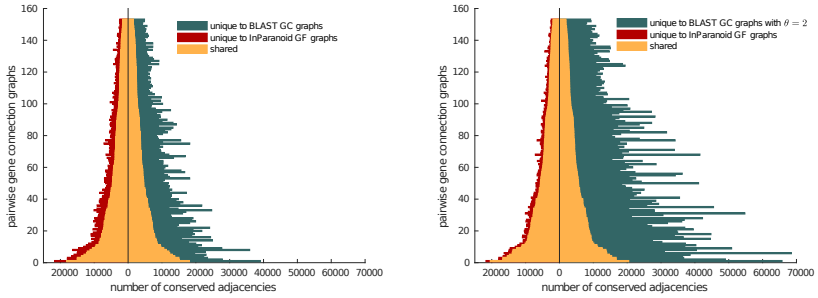

Fig. 4. Overlap of conserved adjacencies between BLASTP GC and InParanoid GF graphs

Lastly, Figure 5 visualizes the number of non-conflicting conserved adjacencies in BLASTP GC and InParanoid GF graphs computed for $\theta = 1$ using Algorithm 3 (left plot) and computed for $\theta = 2$ using Algorithm 3' (right plot). For the former we observed on average 42% more non-conflicting conserved adjacencies in BLASTP GC graphs when compared to their InParanoid GF counterparts, whereas for the latter, this number dropped to 32%. Nevertheless, from $\theta = 1$ to $\theta = 2$ the absolute number of non-conflicting conserved adjacencies increases on average by 27% for BLASTP GC graphs, respectively by 37% for InParanoid GF graphs.

8 Conclusion

We have presented new similarity measures for complete genomes, thereby defining gene connections as an intermediate model of genome similarity representations, between gene families and the gene family-free approach. Our theoretical results with some problem variants being polynomial and

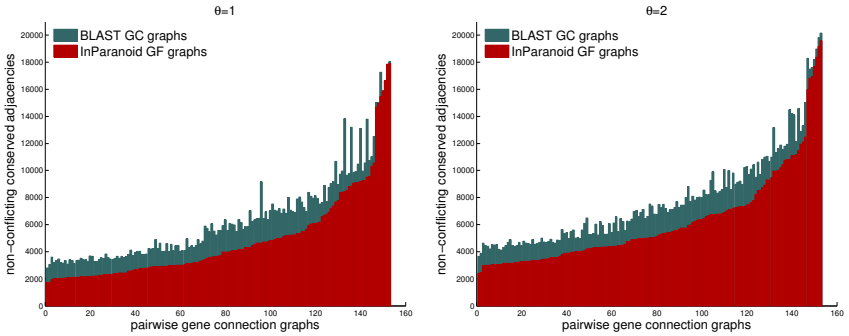

Fig. 5. Numbers of non-conflicting conserved adjacencies in BLASTP GC and InParanoid GF graphs for $\theta = 1$ (left) and $\theta = 2$ (right).

others being NP-hard show that we are very close to the hardness border of similarity computations between genomes with unrestricted gene content. On the practical side we could show that the computation of genomic similarities in the gene connection model gives meaningful results and is possible in reasonable time, if the measures and algorithms are designed carefully.

A few questions remain open, though. While Problem 3 is polynomial for $\theta = 1$, the complexity for larger values of θ is unknown. Moreover, it is always difficult to choose optimal values for parameters like the gap threshold θ. It will certainly be worthwhile to examine whether parameter estimation methods for generalized adjacencies as the ones developed in [39] can be adapted to the gene connection model.

Various model extensions can also be envisaged. The adjacency matching model (Problem 3) removes inconsistencies from the output of the total adjacencies model (Problem 1) by computing a maximum matching on it. It could be tested whether other criteria to remove genes from the genomes and thus derive consistent sets of conserved adjacencies yield even better results. Moreover, the resulting reduced genomes with conserved adjacencies could be used to predict orthologies between the involved genes, not only to compute genomic similarities.

An alternative objective function for our problem formulations, instead of counting (generalized) gene adjacencies, could be a variant of the *summed adjacency disruption number* [40] that also allows to distinguish between small and larger distortions in gene order.

Finally, Algorithm 3 can easily be generalized for weighted gene similarities (instead of gene connections). It remains to be evaluated if such

a more fine-grained measure in the spirit of a family-free analysis has advantages compared to the unit-cost measures studied in this paper.

Acknowledgements. The research of LABK and SD is partially supported by FAPERJ and CNPq. This work was performed while JS was on sabbatical as Special Visiting Researcher at UFF in Niterói, Brazil, funded by Ciência sem Fronteiras/CAPES.

References

1. D. Sankoff. Edit distance for genome comparison based on non-local operations. In A. Apostolico, M. Crochemore, Z. Galil, and U. Manber, editors, *Proc. of CPM 1992*, volume 644 of *LNCS*, pages 121–135, Berlin, 1992. Springer Verlag.
2. S. Hannenhalli and P. A. Pevzner. Transforming cabbage into turnip: Polynomial algorithm for sorting signed permutations by reversals. *J. ACM*, 46(1):1–27, 1999.
3. S. Yancopoulos, O. Attie, and R. Friedberg. Efficient sorting of genomic permutations by translocation, inversion and block interchange. *Bioinformatics*, 21(16):3340–3346, 2005.
4. A. Bergeron, J. Mixtacki, and J. Stoye. A new linear time algorithm to compute the genomic distance via the double cut and join distance. *Theor. Comput. Sci.*, 410(51):5300–5316, 2009.
5. D. Bryant. The complexity of calculating exemplar distances. In D. Sankoff and J. H. Nadeau, editors, *Comparative Genomics*, volume 1 of *Computational Biology Series*, pages 207–211. Kluwer Academic Publishers, London, 2000.
6. X. Chen, J. Zheng, Z. Fu, P. Nan, Y. Zhong, S. Lonardi, and T. Jiang. Assignment of orthologous genes via genome rearrangement. *IEEE/ACM Trans. Comput. Biol. Bioinf.*, 2(4):302–315, 2005.
7. S. Angibaud, G. Fertin, I. Rusu, A. Thevenin, and S. Vialette. Efficient tools for computing the number of breakpoints and the number of adjacencies between two genomes with duplicate genes. *J. Comp. Biol.*, 15(8):1093–1115, 2008.
8. L. Bulteau and M. Jiang. Inapproximability of (1,2)-exemplar distance. *IEEE/ACM Trans. Comput. Biol. Bioinf.*, 10(6):1384–1390, 2012.
9. M. Shao, Y. Lin, and B. M. E. Moret. An exact algorithm to compute the double-cut-and-join distance for genomes with duplicate genes. *J. Comp. Biol.*, 22(5):425–435, 2015.
10. D. Doerr, A. Thévenin, and J. Stoye. Gene family assignment-free comparative genomics. *BMC Bioinformatics*, 13(Suppl. 19):S3, 2012.
11. M. D. V. Braga, C. Chauve, D. Doerr, K. Jahn, J. Stoye, A. Thévenin, and R. Wittler. The potential of family-free genome comparison. In C. Chauve, N. El-Mabrouk, and E. Tannier, editors, *Models and Algorithms for Genome Evolution*, volume 19 of *Computational Biology Series*, pages 63–81. Springer Verlag, Berlin, 2013.
12. D. Doerr, J. Stoye, S. Böcker, and K. Jahn. Identifying gene clusters by discovering common intervals in indeterminate strings. *BMC Bioinformatics*, 15(Suppl. 6):S2, 2014.
13. F. V. Martinez, P. Feijão, M. D. V Braga, and J. Stoye. On the family-free DCJ distance and similarity. *Algorithms Mol. Biol.*, 10:13, 2015.

14. Q. Zhu, Z. Adam, V. Choi, and D. Sankoff. Generalized gene adjacencies, graph bandwidth, and clusters in yeast evolution. *IEEE/ACM Trans. Comput. Biol. Bioinf.*, 6(2):213–220, 2009.
15. D. Sankoff. Genome rearrangement with gene families. *Bioinformatics*, 15(11):909–917, 1999.
16. M. Blanchette, T. Kunisawa, and D. Sankoff. Gene order breakpoint evidence in animal mitochondrial phylogeny. *J. Mol. Evol.*, 49(2):193–203, 1999.
17. E. Tannier, C. Zheng, and D. Sankoff. Multichromosomal median and halving problems under different genomic distances. *BMC Bioinformatics*, 10:120, 2009.
18. J. E. Hopcroft and R. M. Karp. An $n^{5/2}$ algorithm for maximum matchings in bipartite graphs. *SIAM J. Computing*, 2(4):225–231, 1973.
19. D. Doerr. *Gene Family-free Genome Comparison*. Ph. D. thesis, Faculty of Technology, Bielefeld University, Germany, 2015.
20. D. M. Goodstein, S. Shu, R. Howson, R. Neupane, R. D. Hayes, J. Fazo, T. Mitros, W. Dirks, U. Hellsten, N. Putnam, and D. S. Rokhsar. Phytozome: a comparative platform for green plant genomics. *Nucleic Acids Res.*, 40(Database issue):D1178–D1186, 2012.
21. E L. L. Sonnhammer and G. Östlund. Inparanoid 8: orthology analysis between 273 proteomes, mostly eukaryotic. *Nucleic Acids Res.*, 43(Database issue):D234–D239, 2015.
22. P. Lamesch, T. Z. Berardini, D. Li, D. Swarbreck, C. Wilks, R. Sasidharan, R. Muller, K. Dreher, D. L. Alexander, M. Garcia-Hernandez, A. S. Karthikeyan, C. H. Lee, W. D. Nelson, L. Ploetz, S. Singh, A. Wensel, and E. Huala. The arabidopsis information resource (tair): improved gene annotation and new tools. *Nucleic Acids Res.*, 40(Database issue):D1202–D1210, 2011.
23. G. A. Wu, S. Prochnik, J. Jenkins, J. Salse, U. Hellsten, F. Murat, X. Perrier, M. Ruiz, S. Scalabrin, J. Terol, M. A. Takita, K. Labadie, J. Poulain, A. Couloux, K. Jabbari, F. Cattonaro, C. Del Fabbro, S. Pinosio, A. Zuccolo, J. Chapman, J. Grimwood, F. R. Tadeo, L. H. Estornell, J. V. Muñoz-Sanz, V. Ibanez, A. Herrero-Ortega, P. Aleza, J. Pérez-Pérez, D. Ramón, D. Brunel, F. Luro, C. Chen, W. G. Farmerie, B. Desany, C. Kodira, M. Mohiuddin, T. Harkins, K. Fredrikson, P. Burns, A. Lomsadze, Mark B., G. Reforgiato, J. Freitas-Astúa, F. Quetier, L. Navarro, M. Roose, P. Wincker, J. Schmutz, M. Morgante, M. A. Machado, M. Talón, O. Jaillon, P. Ollitrault, F. Gmitter, and D. Rokhsar. Sequencing of diverse mandarin, pummelo and orange genomes reveals complex history of admixture during citrus domestication. *Nat. Biotechnol.*, 32(7):656–662, 2014.
24. T. Slotte, K. M. Hazzouri, J. A. Ågren, D. Koenig, F. Maumus, Y.-L. Guo, K. Steige, A. E. Platts, J. S. Escobar, L. K. Newman, W. Wang, T. Mandáková, E. Vello, L. M. Smith, S. R. Henz, J. Steffen, S. Takuno, Y. Brandvain, G. Coop, P. Andolfatto, T. T. Hu, M. Blanchette, R. M. Clark, H. Quesneville, M. Nordborg, B. S. Gaut, M. A. Lysak, J. Jenkins, J. Grimwood, J. Chapman, S. Prochnik, S. Shu, D. Rokhsar, J. Schmutz, D. Weigel, and S. I. Wright. The *Capsella rubella* genome and the genomic consequences of rapid mating system evolution. *Nat. Genet.*, 45(7):831–835, 2013.
25. J. Bartholomé, E. Mandrou, A. Mabiala, J. Jenkins, I. Nabihoudine, C. Klopp, J. Schmutz, C. Plomion, and J.-M. Gion. High-resolution genetic maps of eucalyptus improve *Eucalyptus grandis* genome assembly. *New Phytol*, 206(4):1283–1296, 2015.
26. R. Yang, D. E. Jarvis, H. Chen, and M. A. Beilstein. The reference genome of the halophytic plant *Eutrema salsugineum*. *Frontiers in plant.*, 2013.

27. V. Shulaev, D. J. Sargent, R. N. Crowhurst, T. C. Mockler, O. Folkerts, A. L. Delcher, P. Jaiswal, K. Mockaitis, A. Liston, S. P. Mane, P. Burns, T. M. Davis, J. P. Slovin, N. Bassil, R. P. Hellens, C. Evans, T. Harkins, C. Kodira, B. Desany, O. R. Crasta, R. V. Jensen, A. C. Allan, T. P. Michael, J. C. Setubal, J.-M. Celton, D. J. G. Rees, K. P. Williams, S. H. Holt, J. J. R. Rojas, M. Chatterjee, B. Liu, H. Silva, L. Meisel, A. Adato, S. A. Filichkin, M. Troggio, R. Viola, T.-L. Ashman, H. Wang, P. Dharmawardhana, J. Elser, R. Raja, H. D. Priest, D. W. Bryant, S. E. Fox, S. A. Givan, L. J. Wilhelm, S. Naithani, A. Christoffels, D. Y. Salama, J. Carter, E. L. Girona, A. Zdepski, W. Wang, R. A. Kerstetter, W. Schwab, S. S. Korban, J. Davik, A. Monfort, B. Denoyes-Rothan, P. Arus, R. Mittler, B. Flinn, A. Aharoni, J. L Bennetzen, S. L. Salzberg, A. W. Dickerman, R. Velasco, M. Borodovsky, R. E. Veilleux, and K. M. Folta. The genome of woodland strawberry (*Fragaria vesca*). *Nat. Genet.*, 43(2):109–116, 2011.
28. J. Schmutz, S. B. Cannon, J. Schlueter, J. Ma, T. Mitros, W. Nelson, D. L. Hyten, Q. Song, J. J. Thelen, J. Cheng, D. Xu, U. Hellsten, G. D. May, Y. Yu, T. Sakurai, T. Umezawa, M. K. Bhattacharyya, D. Sandhu, B. Valliyodan, E. Lindquist, M. Peto, D. Grant, S. Shu, D. Goodstein, K. Barry, M. Futrell-Griggs, B. Abernathy, J. Du, Z. Tian, L. Zhu, N. Gill, T. Joshi, M. Libault, A. Sethuraman, X.-C. Zhang, K. Shinozaki, H. T. Nguyen, R. A. Wing, P. Cregan, J. Specht, J. Grimwood, D. Rokhsar, G. Stacey, R. C. Shoemaker, and S. A. Jackson. Genome sequence of the palaeopolyploid soybean. *Nature*, 463(7278):178–183, 2010.
29. A. H. Paterson, J. F. Wendel, H. Gundlach, H. Guo, J. Jenkins, D. Jin, D. Llewellyn, K. C. Showmaker, S. Shu, J. Udall, M.-J. Yoo, R. Byers, W. Chen, A. Doron-Faigenboim, M. V. Duke, L. Gong, J. Grimwood, C. Grover, K. Grupp, G. Hu, T.-H. Lee, J. Li, L. Lin, T. Liu, B. S. Marler, J. T. Page, A. W. Roberts, E. Romanel, W. S. Sanders, E. Szadkowski, X. Tan, H. Tang, C. Xu, J. Wang, Z. Wang, D. Zhang, L. Zhang, H. Ashrafi, F. Bedon, J. E. Bowers, C. L. Brubaker, P. W. Chee, S. Das, A. R. Gingle, C. H. Haigler, D. Harker, L. V. Hoffmann, R. Hovav, D. C. Jones, C. Lemke, S. Mansoor, M. U. Rahman, L. N. Rainville, A. Rambani, U. K. Reddy, J.-K. Rong, Y. Saranga, B. E. Scheffler, J. A. Scheffler, D. M. Stelly, B. A. Triplett, A. Van Deynze, M. F. S. Vaslin, V. N. Waghmare, S. A. Walford, R. J. Wright, E. A. Zaki, T. Zhang, E. S. Dennis, K. F. X. Mayer, D. G. Peterson, D. S. Rokhsar, X. Wang, and J. Schmutz. Repeated polyploidization of gossypium genomes and the evolution of spinnable cotton fibres. *Nature*, 492(7429):423–427, 2012.
30. Z. Wang, N. Hobson, L. Galindo, S. Zhu, D. Shi, J. McDill, L. Yang, S. Hawkins, G. Neutelings, R. Datla, G. Lambert, D. W. Galbraith, C. J. Grassa, A. Geraldes, Q. C. Cronk, C. Cullis, P. K. Dash, P. A. Kumar, S. Cloutier, A. G. Sharpe, G. K. S. Wong, J. Wang, and M. K. Deyholos. The genome of flax (*Linum usitatissimum*) assembled de novo from short shotgun sequence reads. *Plant J*, 72(3):461–473, 2012.
31. N. D. Young, F. Debellé, G. E. D. Oldroyd, R. Geurts, S. B. Cannon, M. K. Udvardi, V. A. Benedito, K. F. X. Mayer, J. Gouzy, H. Schoof, Y. Van de Peer, S. Proost, D. R. Cook, B. C. Meyers, M. Spannagl, F. Cheung, S. De Mita, V. Krishnakumar, H. Gundlach, S. Zhou, J. Mudge, A. K. Bharti, J. D. Murray, M. A. Naoumkina, B. Rosen, K. A. T. Silverstein, H. Tang, S. Rombauts, P. X. Zhao, P. Zhou, V. Barbe, P. Bardou, M. Bechner, A. Bellec, A. Berger, H. Bergès, S. Bidwell, T. Bisseling, N. Choisne, A. Couloux, R. Denny, S. Deshpande, X. Dai, J. J. Doyle, A.-M. Dudez, A. D. Farmer, S. Fouteau, C. Franken, C. Gibelin, J. Gish, S. Goldstein, A. J. González, P. J. Green, A. Hallab, M. Hartog, A. Hua, S. J.

Humphray, D.-H. Jeong, Y. Jing, A. Jöcker, S. M. Kenton, D.-J. Kim, K. Klee, H. Lai, C. Lang, S. Lin, S. L. Macmil, G. Magdelenat, L. Matthews, J. McCorrison, E. L. Monaghan, J.-H. Mun, F. Z. Najar, C. Nicholson, C. Noirot, M. O'Bleness, C. R. Paule, J. Poulain, F. Prion, B. Qin, C. Qu, E. F. Retzel, C. Riddle, E. Sallet, S. Samain, N. Samson, I. Sanders, O. Saurat, C. Scarpelli, T. Schiex, B. Segurens, A. J. Severin, D. J. Sherrier, R. Shi, S. Sims, S. R. Singer, S. Sinharoy, L. Sterck, A. Viollet, B.-B. Wang, K. Wang, M. Wang, X. Wang, J. Warfsmann, J. Weissenbach, D. D. White, J. D. White, G. B. Wiley, P. Wincker, Y. Xing, L. Yang, Z. Yao, F. Ying, J. Zhai, L. Zhou, A. Zuber, J. Dénarié, R. A. Dixon, G. D. May, D. C. Schwartz, J. Rogers, F. Quetier, C. D. Town, and B. A. Roe. The medicago genome provides insight into the evolution of rhizobial symbioses. *Nature*, 480(7378):520–524, 2011.

32. I. Verde, A. G. Abbott, S. Scalabrin, S. Jung, S. Shu, F. Marroni, T. Zhebentyayeva, M. T. Dettori, J. Grimwood, F. Cattonaro, A. Zuccolo, L. Rossini, J. Jenkins, E. Vendramin, L. A. Meisel, V. Decroocq, B. Sosinski, S. Prochnik, T. Mitros, A. Policriti, G. Cipriani, L. Dondini, S. Ficklin, D. M. Goodstein, P. Xuan, C. Del Fabbro, V. Aramini, D. Copetti, S. Gonzalez, D. S. Horner, R. Falchi, S. Lucas, E. Mica, J. Maldonado, B. Lazzari, D. Bielenberg, R. Pirona, M. Miculan, A. Barakat, R. Testolin, A. Stella, S. Tartarini, P. Tonutti, P. Arus, A. Orellana, C. Wells, D. Main, G. Vizzotto, H. Silva, F. Salamini, J. Schmutz, M. Morgante, and D. S. Rokhsar. The high-quality draft genome of peach (*Prunus persica*) identifies unique patterns of genetic diversity, domestication and genome evolution. *Nat. Genet.*, 45(5):487–494, 2013.
33. Q. Du, L. Wang, X. Yang, C. Gong, and D. Zhang. Populus endo-β-1,4-glucanases gene family: genomic organization, phylogenetic analysis, expression profiles and association mapping. *Planta*, 241(6):1417–1434, 2015.
34. J. Schmutz, P. E. McClean, S. Mamidi, G. A. Wu, S. B. Cannon, J. Grimwood, J. Jenkins, S. Shu, Q. Song, C. Chavarro, M. Torres-Torres, V. Geffroy, S. M. Moghaddam, D. Gao, B. Abernathy, K. Barry, M. Blair, M. A. Brick, M. Chovatia, P. Gepts, D. M. Goodstein, M. Gonzales, U. Hellsten, D. L. Hyten, G. Jia, J. D. Kelly, D. Kudrna, R. Lee, M. M. S. Richard, P. N. Miklas, J. M. Osorno, J. Rodrigues, V. Thareau, C. A. Urrea, M. Wang, Y. Yu, M. Zhang, R. A. Wing, P. B. Cregan, D. S. Rokhsar, and S. A. Jackson. A reference genome for common bean and genome-wide analysis of dual domestications. *Nat. Genet.*, 46(7):707–713, 2014.
35. A. P. Chan, J. Crabtree, Q. Zhao, H. Lorenzi, J. Orvis, D. Puiu, A. Melake-Berhan, K. M. Jones, J. Redman, G. Chen, E. B. Cahoon, M. Gedil, M. Stanke, B. J. Haas, J. R. Wortman, C. M. Fraser-Liggett, J. Ravel, and P. D. Rabinowicz. Draft genome sequence of the oilseed species *Ricinus communis*. *Nat. Biotechnol.*, 28(9):951–956, 2010.
36. J. C. Motamayor, K. Mockaitis, J. Schmutz, N. Haiminen, D. Livingstone, O. Cornejo, S. D. Findley, P. Zheng, F. Utro, S. Royaert, C. Saski, J. Jenkins, R. Podicheti, M. Zhao, B. E. Scheffler, J. C. Stack, F. A. Feltus, G. M. Mustiga, F. Amores, W. Phillips, J. P. Marelli, G. D. May, H. Shapiro, J. Ma, C. D. Bustamante, R. J. Schnell, D. Main, D. Gilbert, L. Parida, and D. N. Kuhn. The genome sequence of the most widely cultivated cacao type and its use to identify candidate genes regulating pod color. *Genome Biol*, 14(6):r53, 2012.
37. O. Jaillon, J.-M. Aury, B. Noel, A. Policriti, C. Clepet, A. Casagrande, N. Choisne, S. Aubourg, N. Vitulo, C. Jubin, A. Vezzi, F. Legeai, P. Hugueney, C. Dasilva, D. Horner, E. Mica, D. Jublot, J. Poulain, C. Bruyère, A. Billault, B. Segurens,

M. Gouyvenoux, E. Ugarte, F. Cattonaro, V. Anthouard, V. Vico, C. Del Fabbro, M. Alaux, G. Di Gaspero, V. Dumas, N. Felice, S. Paillard, I. Juman, M. Moroldo, S. Scalabrin, A. Canaguier, I. Le Clainche, G. Malacrida, E. Durand, G. Pesole, V. Laucou, P. Chatelet, D. Merdinoglu, M. Delledonne, M. Pezzotti, A. Lecharny, C. Scarpelli, F. Artiguenave, M. E. Pè, G. Valle, M. Morgante, M. Caboche, A.-F. Adam-Blondon, J. Weissenbach, F. Quetier, P. Wincker, and French-Italian Public Consortium for Grapevine Genome Characterization. The grapevine genome sequence suggests ancestral hexaploidization in major angiosperm phyla. *Nature*, 449(7161):463–467, 2007.

38. M. Lechner, M. Hernandez-Rosales, D. Doerr, N. Wieseke, A. Thévenin, J. Stoye, R. K. Hartmann, S. J. Prohaska, and P. F. Stadler. Orthology detection combining clustering and synteny for very large datasets. *PLoS ONE*, 9(8):e10515, 2014.
39. Z. Yang and D. Sankoff. Natural parameter values for generalized gene adjacencies. *J. Comp. Biol.*, 17(9):1113–1128, 2010.
40. J. Delgado, I. Lynce, and V. Manquinho. Computing the summed adjacency disruption number between two genomes with duplicate genes. *J. Comp. Biol.*, 17(9):1243–1265, 2010.