OMG! Orthologs in Multiple Genomes – Competing Graph-Theoretical Formulations

Chunfang Zheng[1,2], Krister Swenson[1,2], Eric Lyons[3], and David Sankoff[1]

[1] Department of Mathematics and Statistics, University of Ottawa
[2] Département d'informatique et de recherche opérationnelle, Université de Montréal
[3] iPlant, Department of Plant Sciences, University of Arizona

Abstract. From the set of all pairwise homologies, weighted by sequence similarities, among a set of genomes, we seek disjoint orthology sets of genes, in which each element is *orthogonal* to all other genes (on a different genome) in the same set. In a graph-theoretical formulation, where genes are vertices and weighted edges represent homologies, we suggest three criteria, with three different biological motivations, for evaluating the partition of genes produced by deletion of a subset of edges: i) minimum weight edge removal, ii) minimum degree-zero vertex creation, and iii) maximum number of edges in the transitive closure of the graph after edge deletion. For each of the problems, all either proved or conjectured to be NP-hard, we suggest approximate and heuristic algorithms of finding orthology sets satisfying the criteria, and show how to incorporate genomes that have a whole genome duplication event in their immediate lineage. We apply this to ten flowering plant genomes, involving 160,000 different genes in given pairwise homologies. We evaluate the results in a number of ways and recommend criterion iii) as best suited to applications to multiple gene order alignment.

1 Introduction

Multiple alignment of the gene orders in sequenced genomes is an important and timely problem in comparative genomics [1,2,3,4]. A key aspect is the construction of disjoint orthology sets of genes, in which each element is orthologous to all other genes (on different genomes) in the same set. Approaches differ as to the nature and timing and relative importance of sequence alignment, synteny block construction, and paralogy resolution in constructing these sets. We argue that these considerations are best integrated in the construction of *pairwise* synteny blocks as a first step, followed by the conflation of the pairwise orthologies into larger sets. The advantages of this are the availability of finely tuned pairwise synteny block software (e.g., SynMap in the CoGe platform [5,6]), the possibility of dealing with paralogs dating from ancient whole genome duplication (WGD) in a natural way, and the opportunity to dispense with thresholds or other arbitrary settings during the construction of the orthology sets themselves. The task of discerning the orthology sets becomes a purely algorithmic problem on graphs.

T.M. Przytycka and M.-F. Sagot (Eds.): WABI 2011, LNBI 6833, pp. 364–375, 2011.
© Springer-Verlag Berlin Heidelberg 2011

Here we distinguish three variants of this Orthologs in Multiple Genomes (omg) problem, differentiated by their objective functions and their biological justifications. All are expected to be hard, from the algorithmic point of view, although some have available approximation algorithms. In Section 2, we first define the different formulations of the omg problem and discuss their biological interpretations. We then present and analyze the algorithms we design for each formulation, including variants of the omg problem incorporating paralogy data from genomes known to be descendants of wgd events.

Our approach, though widely applicable, was developed within the context of flowering plant genomics, an evolutionary domain characterized by recurrent wgd events. The massive data set we analyze is described in Section 3.

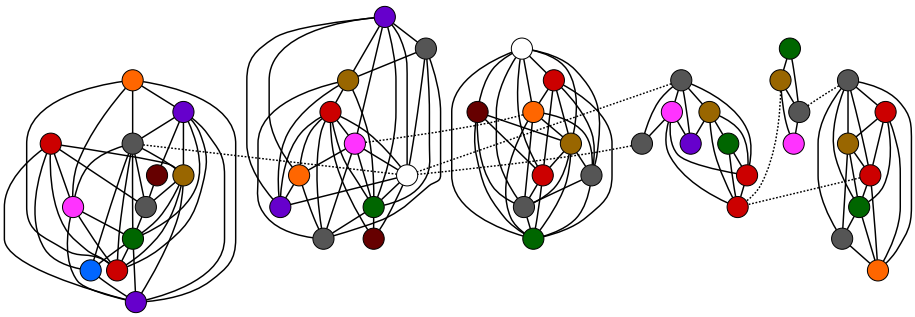

Fig. 1. Homology set, showing "erroneous" edges between orthology sets

In Section 4, we compare the results of applying the algorithms to a large data set of homologies among some 160,000 plant genes drawn from ten eudicotyledon genomes. We compare the results of the three methods when evaluated by the other two objectives, and by assessing the compatibility of the orthology sets with the phylogenetic tree that is assumed to have generated them. We find that some of the orthology sets recovered by each method are also recovered by the other two, but each method also finds many sets specific to that method.

Section 5 contains the conclusions, including reflections on restrictions on accessibility of genome data for the purposes of comparative genomics. We conclude that one of the three approaches produces somewhat bigger orthology sets, so that insofar as these sets are biologically validated, this method can be recommended for purposes of multiple gene order alignment.

2 The Competing Formulations and Their Algorithms

The pairwise homologies SynMap provides for all pairs of genomes constitute the set of edges E of the *homology graph* $H = (V, E)$, where V represents the set of genes in any of the genomes participating in at least one homology relation. In addition there is a weight $w(e) \in (0, 1]$ associated with each edge $e \in E$,

representing a protein similarity score. Let $H = H_1 \cup \cdots \cup H_s$ represent the decomposition of H into connected components. Since we expect SynMap to resolve all or most paralogies, ideally all the genes in each H_i should be orthologous. There should be *at most* one gene from each genome in such an orthology set, or at most two duplicate genes for genomes that descend from a WGD event. In practice, however, as in Fig. 1, there may be several genes from the same genome in an H_i, apparent paralogies, which we shall consider erroneous due to spurious homologies (edges) in the input. The problem we address, then, is how to convert H into a new graph $O = O_1 \cup \cdots O_t$ with the *orthogonality* property desired of each connected component O_j, namely that it contain no paralogs, except for duplicates in genomes descended from WGDs.

Definition 1. *A graph $O = (V, E)$ with vertices in c colour classes is an orthogonal partition if each of its components contains at most one vertex of any one colour.*
Given any graph $H = (V, E)$ with coloured vertices, it can be converted into an orthogonal partition $O = (V, E \setminus E')$ by deleting a subset E' of edges from E.

Problem 1. Given a criterion $\kappa = \kappa(V, E, E')$, where $(V, E \setminus E')$ is an orthogonal partition, the OMG problem is to find a subset $E' \subset E$ that optimizes κ.

Definition 2. *Let $c = c_1 + c_2$ where there are c_2 distinguished colours called* WGD *colours. The graph $O = (V, E)$ is a* WGD*-orthogonal partition if each of its components contains at most two vertices of any of the c_2 distinguished colours, and at most one vertex of any of the c_1 remaining colours.*

Problem 2. Given $c_2 \leq c$ distinguished colour classes and criterion $\kappa = \kappa(V, E, E')$, where $(V, E \setminus E')$ is an WGD-orthogonal partition, the OMG problem is to find a subset $E \subset E'$ that optimizes κ.

In Sections 2.1,2.2 and 2.3 we will discuss motivations for three different criteria κ and present algorithms for each one, for both the usual definition of orthogonality (Definition 1) and for the WGD version (Definition 2).

2.1 Minimum Weight Orthogonal Partition (MWOP)

Our first approach is simply to delete a set of edges E' of minimum weight. This definition of κ is motivated by the desire to conserve as many of the homology inferences in the input data as possible, and to discard as noise as few as possible. This is a NP-hard graph problem, Minimum Weight Orthogonal Partition (MWOP), for which He *et al.* have given an approximation algorithm [7].

The algorithm iteratively merges vertices into orthogonal sets using the maximum weight bipartite matching at each step; the result of a matching between two colours produces orthogonal sets which each can be considered as single vertices with two colours. Now a second matching can be found, and so on. We used an auction routine for maximum weight bipartite matching [11,12].

The output of the algorithm depends on the order in which the vertices are merged. In our version, this order is determined by phylogeny. Thus, part of the

input is a rooted, binary tree T, with each leaf corresponding to one of the given colours . The biological motivation is that a homology relation between genes in closely related genomes is less likely to be spurious than in distant relatives.

The case of no WGD descendants (Problem 1).

1. **for** each genome i with n_i genes, $n_m = \max_i n_i$, define sets

$$\mathbf{V^i} = \{V_1^{\{i\}} = \{v_1\}, \cdots, V_{n_i}^{\{i\}} = \{v_{n_i}\}, V_{n_i+1}^{\{i\}} = \emptyset, \cdots, V_{n_m}^{\{i\}} = \emptyset\}.$$

2. All the leaves of T are eligible to merge. All ancestral nodes are ineligible.
3. **while** there remain unmerged but eligible sister nodes (same immediate parent) in the tree T.
   (a) choose eligible sister nodes i and j.
   (b) construct a complete bipartite graph between $\mathbf{V^i}$ and $\mathbf{V^j}$.
   (c) if there is an edge uv, with $u \in V_a^{\{i\}}, v \in V_b^{\{j\}}$ set

$$w(V_a^{\{i\}}, V_b^{\{j\}}) = \sum_{u \in V_a^{\{i\}}, v \in V_b^{\{j\}}} w(uv),$$

   otherwise $w(V_a^{\{i\}}, V_b^{\{j\}}) = 0$.
   (d) find the maximum weight matching for the bipartite graph.
   (e) for each pair $V_a^{\{i\}} V_b^{\{j\}}$ in the matching, set

$$V_a^{\{i\}} \leftarrow V_a^{\{i\}} \cup V_b^{\{j\}}.$$

   (f) the new $\mathbf{V^i}$ is associated with the ancestral node of T that is the immediate parent of i and j, which now becomes eligible to merge; $V_b^{\{j\}}$ is now disregarded.
4. the remaining set V^i corresponds to O (i.e. $O_1 = V_1^{\{i\}}$, $O_2 = V_2^{\{i\}}$, etc.).

WGD descendants allowed. (Problem 2.) If genome j is a WGD descendant, then after Step 3 (e)

i. if a vertex $V_a^{\{i\}}$ is matched to a vertex $V_b^{\{j\}}$ in genome j, where $w(V_a^{\{i\}} V_b^{\{j\}}) > 0$, then set $V_b^{\{j\}} \leftarrow \emptyset$.
ii. construct a complete bipartite graph for $\mathbf{V^i}$ and the modified genome j.
iii. find the maximum weight matching for this graph.
iv. for any two positively weighted matching edges $e_1 = V_x^{\{i\}} V_y^{\{j\}}, e_2 = V_u^{\{i\}} V_v^{\{j\}}$ that share a gene z, remove the edge with less weight.
v. for all remaining edges e merge the vertex in genome j with the one in $\mathbf{V^i}$ if a positively matched edge ($w > 0$) exists between these two vertices, and remove that j genome gene from any other $\mathbf{V^i}$ vertex it may already be in (from the merge in step 3 (e)).

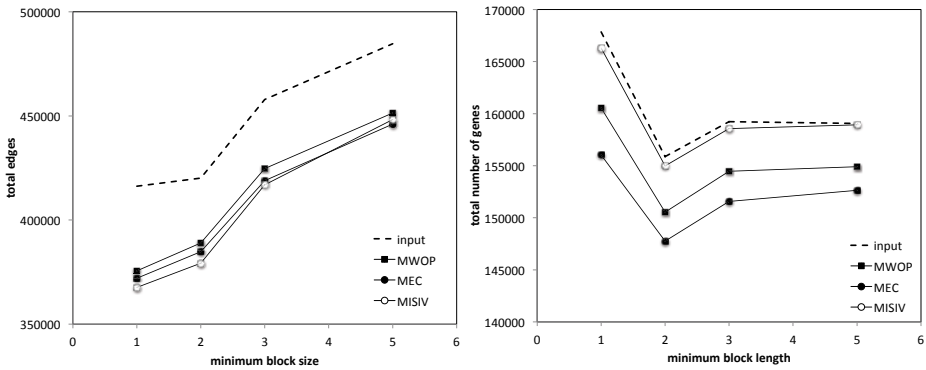

Fig. 2. Edges retained and degree-zero vertices created by the three OMG methods

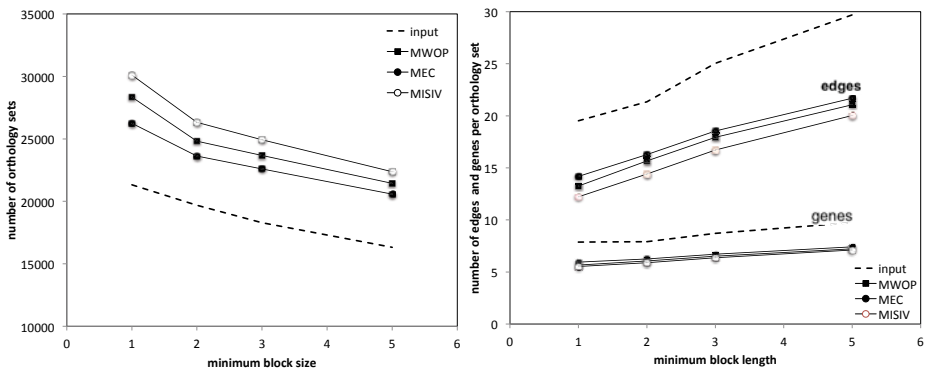

Fig. 3. Number and size of orthology sets produced by the three OMG methods

2.2 Minimize Singleton Vertices (MISIV)

Deleting edges in E' can create *singletons*, degree zero vertices. Each of these trivial components contains no orthology information, and is of little use in comparative genomic applications such as multiple gene order alignment. Our second criterion, then, is to minimize the number of vertices of degree zero created by edge removal. This definition of κ is designed to keep homology information on as many genes as possible in the data set. We seek an orthogonal partition $O = (V_1, E_1) \cup \cdots \cup (V_t, E_t)$ to minimize $|\{E_i | E_i = \emptyset\}|$. We are not aware of previous algorithms for this problem, and conjecture that it is NP-hard.

The case of no WGD descendants (Problem 1). We first initialize $E' = \emptyset$, subsets $V_v = \{v\}$ for each $v \in V$, and the relation c such that $V_v \,\mathsf{c}\, V_u$ if u and v have the same colour and $V_v \not\mathsf{c}\, V_u$ if not. We set $w(uv) = 0$ if $uv \notin E$. Without ambiguity, we extend w to be a weight on pairs of sets, and $w(\{u\}, \{v\}) = w(uv)$.

1. **while** there remains a subset V_v consisting of a single vertex of degree 1, and an edge $uv \in E \setminus E'$ for some vertex $u \in V_x$, where $V_v \not\subset V_x$, and $w(uv)$ is maximum over all V_v, do the following:
   (a) $V_x \leftarrow V_x \cup V_v$.
   (b) **if** $V_x \subset V_z$ or $V_v \subset V_z$ for any set V_z, then we set $V_x \subset V_z$.
   (c) **for** all V_w consisting of a single vertex of degree 1, where $zw \in E \setminus E'$, $z \in V_x$, $V_w \subset V_x$, delete edge zw, i.e. $E' \leftarrow E' \cup \{zw\}$
   (d) $V_v \leftarrow \emptyset$.
2. **while** there remains a subset V_v consisting of a single vertex v, and an edge $uv \in E \setminus E'$ for some vertex $u \in V_x$, where $V_v \not\subset V_x$,
   (a) Construct the subgraph of $(V, E \setminus E')$ induced by all v satisfying these conditions. Find the maximum weight matching of these subsets.
   (b) **for** each pair V_x and V_y in the matching,
   merge$[V_x, V_y]$:
      i. $V_x \leftarrow V_x \cup V_y$.
      ii. **if** $V_x \subset V_z$ or $V_y \subset V_z$ for any set V_z, then we set $V_x \subset V_z$ and $w(V_x, V_z) = 0$. If there is an edge e joining any vertex in V_x and any vertex in V_z, delete it, i.e., $E' \leftarrow E' \cup \{e\}$.
      iii. **for** all V_z, $V_z \not\subset V_x$, set $w(V_x, V_z) = \sum_{\{uv \mid u \in V_x, v \in V_z\}} w(u, v)$,
         if $w(V_x, V_z) > 0, E = E \cup \{V_x V_z\}$.
      iv. $V_y \leftarrow \emptyset$.
3. **while** there remain at least two subsets $V_x \not\subset V_y$, and vertices $u \in V_x, v \in V_y$, where $uv \in E \setminus E'$,
   (a) Find the maximum weight matching among all these subsets.
   (b) **for** each pair V_x and V_y in the matching,
   merge$[V_x, V_y]$ (Steps 2 (b) i.–iv. above)
4. relabel the remaining sets $O_1, \cdots, O_t$; these contain the vertices of the required components of O.

The strategy of this heuristic is to irreversibly enlarge the components by first adding the vertices most vulnerable to becoming degree zero through edge deletion, namely those of degree one. After as many of these as possible (or a combination of them having greatest weight) are thus "protected", we then try to protect as many others as possible through a series of maximum weight matches in the subgraph induced by unprotected vertices. Step 3 may merge some sets of vertices without any effect on the objective function (minimum number of degree zero vertices), but in a way that tends to improve our result with respect to other objective functions (fewer and larger components; fewer, or lesser combined weight of, edges deleted).

WGD descendants allowed. (Problem 2.) The algorithm is easily extended to genomes that have paralogs resulting from WGD. The color relation $\subset$ originally served to block any merger of two sets that would result in two paralogs in the same set. It needs only to be reinterpreted to block mergers resulting in three paralogs in the same set from a WGD descendant.

2.3 Maximum Edges in Transitive Closure (MEC)

The understanding of orthologous genes in two genomes as originating in a single gene in the most recent common ancestor of the two species leads logically to transitivity as a necessary property of the orthology relation. If gene x in genome X is orthologous both to gene y in genome Y and gene z in genome Z, then y and z must also be orthologous, even if SynMap does not detect this homology.

This motivates our third criterion for $O = O_1 \cup \cdots O_t$, namely that the weights of the edges in the transitive closure of O (or in all the cliques generated by the components O_i) be maximized. In other words, this definition of criterion κ maximizes the the sum of the weights over $\sum_1^t \binom{|O_i|}{2}$ edges, preferring to create a few large orthology sets rather than many smaller ones with the same total number of edges. Again, we are not aware of any previous algorithm for this problem, but conjecture it to be NP-hard.

Let $\bar{H} = (V, \bar{E})$ be the transitive closure of graph H. To obtain $\bar{H}$ we raise its adjacency matrix M_H (including 1's on the diagonal) to successively higher powers until convergence to some M_H^r. This could be accelerated using Warshall's algorithm [8]. (N.B. $r \leq \text{diameter}(H)$.) Without loss of generality we may assume H is connected, so $\bar{H}$ is a complete graph and all elements of M_H^r are non-zero. Elements of this matrix thus obtained should represent indirectly inferred orthologies as discussed above, but there may in fact be many paralogies. To remedy, this we first examine the star subgraph $s(v)$ of $\bar{H}$ containing $\nu(v)$ vertices, namely v, its $\nu(v) - 1$ neighbours, and the $\nu(v) - 1$ edges connecting the former to the latter.

Let $c(v) \geq 1$ be the number of distinct colours among the vertices in $s(v)$. Let $F(E) = \sum_{v \in V} c(v)$.

The case of no WGD descendants (Problem 1).

1. set $E' = \emptyset$.
2. **while** there are still some $v \in V$ where $\nu(v) > c(v)$,
   (a) find the edge $e \in E \setminus E'$ that maximizes
   $$F(E \setminus E'') = \sum_{v \in V} c(v), \text{where } E'' = E' \cup \{e\}$$
   (b) **if** there are several such e, find the one that minimizes
   $$F^+(E \setminus E'') = \sum_{(V, E \setminus E'')} \nu(v) - c(v).$$
   (c) **if** there are still several such e, find one with minimum $w(e)$.
   (d) $E' \leftarrow E' \cup \{e\}$
3. relabel as $O_1, \cdots, O_t$ the disjoint components created by deleting edges. These contain the vertices of the required components of O.

Implicit in each greedy step is an attempt to create large orthology sets. If the deleted edges create two partitioned components, i.e., each with no internal paralogy, then the increment in F will be proportional to the sum of the squares of the number of vertices in each one. This favours a decomposition into one large and one small component rather than two equal sized components.

WGD descendants allowed. (Problem 2.) To handle paralogs of WGD origin, the definition of $c(v)$ must be amended to take account an allowance of 2 vertices of the same colour in $s(v)$ if these are from the appropriate genomes. And the condition in Step 2 must require that at most two vertices be contained in $s(v)$ of any one colour, and only if these involve WGD descendants.

2.4 Issues of Accuracy, Deliberate Bias and Interpretation

In the next section, we focus more on the systematically divergent output generated by the different objective functions κ than on accuracy issues of the algorithms themselves. We know that the MWOP algorithm [7] is an approximation algorithm with a large approximation constant. Notwithstanding this uncertainty, it works well on small examples, as confirmed by our own testing. The other heuristics also satisfy the objective functions, or come very close, in small scale tests. In addition, they are efficient, an important consideration when there are many components H_i each containing hundreds of edges. Most importantly, however, the three algorithms will be seen to produce different inventories of orthology sets, each with a bias for meeting a particular biological motivation; MWOP to retain more edges, MISIV to avoid degree zero vertices and MEC to produce large orthology sets. This divergence was not contrived; indeed, while trying to satisfy one objective criterion, as a secondary policy we tried to satisfy the other criteria whenever there was a choice, e.g., step 3 of the MISIV algorithm.

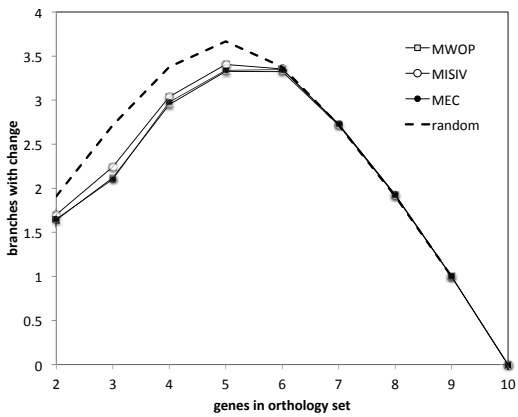

Fig. 4. Compatibility of orthology sets with phylogeny

3 Application to 10 Plant Genomes

We used data drawn from 10 core eudicotyledon genomes available in CoGe. While all genomes have been publicly available for at least one year, some lack a primary publication. To avoid infringing on the release conditions claimed by some of the sequencing groups, we will not identify the genomes we sampled. For the same reason, although we will make use of the phylogenetic relationships among these plants, which respects the current consensus [9], we will not present the phylogeny explicitly. However, for a list of sequenced eudicotyledon genomes in CoGe, consult `http://genomevolution.org/r/3119`; for a list of sequenced plant genomes, `http://genomevolution.org/r/3118`.

We used SynMap to produce sets of synteny blocks between all 45 pairs of genomes, and additionally within each of the five descendants of WGD events. Four different data sets were created using the QuotaAlign option [10], by varying the minimum block length parameter through the four values 1,2,3 and 5. We used the default options for all other settings. An average of more than 10,000 pairs of homologous genes were inferred in the runs for minimum block size 5, for example, involving slightly less than 10,000 distinct genes in each genome.

We extracted all the homology relations from all the synteny blocks in each pair, and put them all together, along with the paralogies within the five WGD descendants, to form the graph H. This had some 160,000 vertices and 485,000 edges, falling into 16,300 disjoint components.

3.1 Percolation, Tangles and Run Time

In order to pick up as many true orthologies as possible, SynMap will unavoidably have some low rate of spurious identification of homologs. This has little consequence for pairwise genomic correspondences for which SynMap was built, but when multiple sets of correspondences are merged to generate the set H, a kind of local percolation phenomenon manifests itself as a large tangle of genes, most of which are not closely related, but are contained in the same component. It is inherent in this non-zero rate of spurious homology, even though it is low, that the larger the tangle, the more likely it is to grow as the number of genomes increases. With our 10 genomes and for minimum block size 5, for example, 29,000 of the 485,000 edges, or 6 %, were involved in one large tangle, as shown in Table 1. With minimum block size 1, the tangle contained almost 90,000 edges.

Table 1. Distribution of component size, showing one large tangle

	frequencies of size of component in H					
edges in component	1	2-14	15-105	106-300	301-1000	29,134
frequency	2550	2776	9,350	523	26	1

Tuning SynMap to be more conservative might lead to smaller tangles, but the systematic experimentation that would be required is beyond the scope of the present study. Note that the ortholog sets we require contain at most 15 genes all linked by 105 edges, one gene from each genome or possibly two from each descendant of a WGD event. The current implementation of MEC requires excessive computing time when the number of edges exceeds a few hundred. To avoid this, we simply preprocess the large components in H, filtering out homologs with weights w below a threshold w^*. The value of $w*$ is raised until the tangle and other large components break up into pieces smaller than 300 edges. This step is unnecessary for MWOP and MISIV but for comparability we use the same reduced input graph for all the methods.

The worst case run time for the auction method we use in our MWOP implementation is $O(w'|V|^3)$, where w' is an integer representing the maximum weight on any edge in the current matching step - the original two-decimal input weights having been converted at the outset to integers. Then $O(Nw'|V|^3)$ is the run time for our entire algorithm, where N is the total number of genomes being matched. Since $|E| \leq 300$, the number of vertices $|V|$ is small and the algorithm runs in a few milliseconds.

Similarly, the small size of $|V|$ means that implementation of an efficient maximum weight matching, say $O(|V|^3)$, within the MISIV algorithm is unnecessary for the current comparison, even when the optional extra step 3 is executed. The algorithm generally runs in less than a second, but can occasionally take a few seconds.

The exhaustive greedy search at every step of the MEC algorithm is time consuming, with even the Warshall algorithm for finding the transitive closure requiring $O(|V|^3)$ time, or $O(|E|^2|V|^3)$ for finding all edges to remove, each time checking all $|E|$ of them. This works out to about 30 seconds per run.

Recall that each of the algorithms was applied to 15-20,000 homology graph components, for four different block length thresholds.

4 Results

Figure 2 shows that MWOP retains marginally more edges than the other two methods, but that MISIV creates far fewer degree-zero vertices. For minimum block length 5, the total number of deleted edges is about 8 % of all edges and total gene deletions about 3.5 %. The increasing number of edges as a function of minimum block length is an artifact of tangle size; in fact there may be marginally more edges with smaller minimum block lengths, but the tangle is much larger, so that when this is resolved by the method in Section 3.1, there are fewer edges input to our algorithms for smaller minimum block lengths.

Figure 3 shows that MEC packs the same number of edges into fewer, bigger (and therefore better for biological purposes) orthology sets.

To test to what extent the gene sets computed are compatible with the known phylogeny, we computed for each gene the minimum number of times it would have to be inserted and deleted on the tree (e.g. a gene set that includes a gene

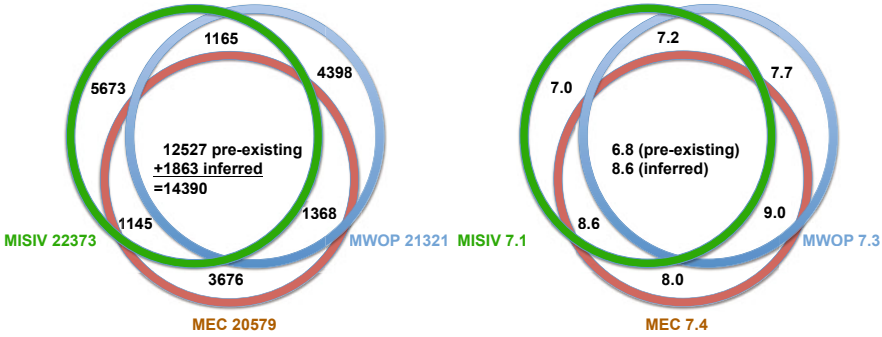

Fig. 5. Proportional Venn diagrams of the number (left) and mean size (right) of the orthology sets recovered by three methods

from all the genomes in the phylogeny requires zero insertions and deletions). The average of this value for each set size is shown in Figure 4. There is a strong phylogenetic signal appearing for small to moderate sized orthology sets, when compared to orthology sets with genes allocated at random to a given number of genomes, i.e., the inferred orthology sets implied a smaller number of insertions and deletion on the tree than random data.

Figure 5 depicts a proportional Venn diagram of the number of orthology sets shared among the results of the three methods. It can be seen that about 60% of the sets recovered by any of the methods were already orthogonal in the input data, and another 8 - 9% are also found by both of the other two methods. An additional 5 - 6.5 % of sets from one method are shared with only one of the other two. The MEC method can be seem as producing the smallest proportion of "idiosyncratic" sets, around 18% (or 45 % of the inferred, or non-pre-existing sets), and MISIV the most, more than 25% (or 57 % of the inferred, or non-pre-existing sets). MEC in this sense is somewhat of a compromise between the extreme "save homologies" goal of MWOP and the "save genes" aim of MISIV.

5 Conclusions

At least for the aims of gene-order alignment, the larger orthology sets produced by MEC lead to our recommendation of MEC as the method of choice. This must eventually be subjected, of course, to validation against curated orthology sets.

We can also recommend using minimum block length of at least 5 with SynMap, as Figures 2 and 3 show improved results according to several parameters with increasing block length. At some point, however, higher minimum block length will reduce the number of edges produced by SynMap.

At present, we consider only WGD events in the immediate lineage of single genomes in the data set; paralogs dating from ancient polyploidies shared by all the genomes present will already have been resolved in the output of SynMap. Future developments should allow several genomes to share a WGD event in their ancestry, and for multiple WGD events to occur in the lineage of a single genome.

The problem of tangles is a minor annoyance, quantitatively, in our work, but it could become much worse as increasing numbers of genomes are included in the analysis. Avenues available for attenuating this include tuning SynMap to be more conservative, increasing minimum block length, and more stringent criterion for WGD-origin paralogies in H.

The ultimate validation of the orthology sets produced by the different methods will involve the careful study of many of the sets, especially those that are not captured by all methods, and their comparison with biologically curated sets of homologs. This evaluation, as well as a comparison of the phylogenetic consequences of using different methods for OMG and the identification of genomic data most susceptible to tangles, is hindered by the severe interpretations of the Fort Lauderdale convention imposed by many genome projects. These constraints surpass those recognized by the broader community [13] and are not necessarily respected by journals [14].

References

1. Deniélou, Y.P., Sagot, M.-F., Boyer, F., Viari, A.: Bacterial syntenies: an exact approach with gene quorum. BMC Bioinformatics (in press, 2011)
2. Fostier, J., et al.: A greedy, graph-based algorithm for the alignment of multiple homologous gene lists. Bioinformatics 27, 749–756 (2011)
3. Shulaev, V., et al.: The genome of woodland strawberry (*Fragaria vesca*). Nat. Genetics 43, 109–116 (2011)
4. Zheng, C., Sankoff, D.: Gene order in Rosid phylogeny, inferred from pairwise syntenies among extant genomes. In: Chen, J., Wang, J., Zelikovsky, A. (eds.) ISBRA 2011. LNCS, vol. 6674, pp. 99–110. Springer, Heidelberg (2011)
5. Lyons, E., Freeling, M.: How to usefully compare homologous plant genes and chromosomes as DNA sequences. Plant J 53, 661–673 (2008)
6. Lyons, E., et al.: Finding and comparing syntenic regions among Arabidopsis and the outgroups papaya, poplar and grape: CoGe with rosids. Plant Phys. 148, 1772–1781 (2008)
7. He, G., Liu, J., Zhao, C.: Approximation algorithms for some graph partitioning problems. Journal of Graph Algorithms and Applications 4, 1–11 (2000)
8. Warshall, S.: A theorem on boolean matrices. Journal of the ACM 9, 11–12 (1962)
9. Angiosperm Phylogeny Group.: An update of the Angiosperm Phylogeny Group classification for the orders and families of flowering plants: APG III. Botanical Journal of the Linnean Society 161, 105–121 (2009)
10. Tang, H., et al.: Screening synteny blocks in pairwise genome comparisons through integer programming. BMC Bioinformatics 12, 102 (2011)
11. Mestre, J.: Maximum weight matching via auctions (2009), `http://www.mpi-inf.mpg.de/departments/d1/teaching/ws09_10/Opt2/handouts/lecture1.pdf`
12. Nisan, N.: Auction algorithm for bipartite matching. *Algorithmic Game Theory/Economics* (2009), `http://agtb.wordpress.com/2009/07/13/auction-algorithm-for-bipartite-matching/`
13. Birney, E., et al.: Prepublication data sharing. Nature 461, 168–170 (2009), Ehrlich SD
14. Nature Editors: Sacrifice for the greater good? Nature 421, 875 (2003)