A Parsimony Approach to Genome-Wide Ortholog Assignment

Zheng Fu[1], Xin Chen[2], Vladimir Vacic[1], Peng Nan[3], Yang Zhong[1], and Tao Jiang[1,4]

[1] Computer Science Department, University of California - Riverside
[2] School of Physical and Mathematical Sci., Nanyang Tech. Univ., Singapore
[3] Shanghai Center for Bioinformatics Technology, Shanghai, China
[4] Tsinghua University, Beijing, China

Abstract. The assignment of orthologous genes between a pair of genomes is a fundamental and challenging problem in comparative genomics, since many computational methods for solving various biological problems critically rely on *bona fide* orthologs as input. While it is usually done using sequence similarity search, we recently proposed a new combinatorial approach that combines sequence similarity and genome rearrangement. This paper continues the development of the approach and unites genome rearrangement events and (post-speciation) duplication events in a single framework under the parsimony principle. In this framework, orthologous genes are assumed to correspond to each other in the most parsimonious evolutionary scenario involving both genome rearrangement and (post-speciation) gene duplication. Besides several original algorithmic contributions, the enhanced method allows for the detection of inparalogs. Following this approach, we have implemented a high-throughput system for ortholog assignment on a genome scale, called MSOAR, and applied it to the genomes of human and mouse. As the result will show, MSOAR is able to find 99 more true orthologs than the INPARANOID program did. We have also compared MSOAR with the iterated exemplar algorithm on simulated data and found that MSOAR performed very well in terms of assignment accuracy. These test results indiate that our approach is very promising for genome-wide ortholog assignment.

1 Introduction

Orthologs and *paralogs*, originally defined in [6], refer to two fundamentally different types of homologous genes. They differ in the way that they arose: orthologs are genes that evolved by speciation, while paralogs are genes that evolved by duplication. To better describe the evolutionary process and functional diversification of genes, paralogs are further divided into two subtypes: *outparalogs*, which evolved via an ancient duplication preceding a given speciation event under consideration, and *inparalogs*, which evolved more recently, subsequent to the speciation event [16][10]. For a given set of inparalogs on a genome, there

A. Apostolico et al. (Eds.): RECOMB 2006, LNBI 3909, pp. 578–594, 2006.

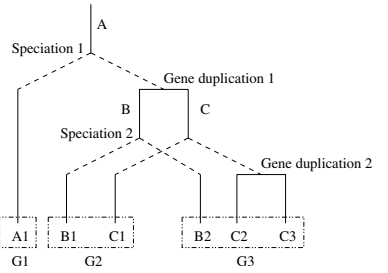

Fig. 1. An illustration of orthologous and paralogous relationships. After two speciation events and two gene duplications, three present genomes, $G_1 = (A_1)$, $G_2 = (B_1, C_1)$ and $G_3 = (B_2, C_2, C_3)$ are obtained. In this scenario, all genes in G_2 and G_3 are co-orthologous to gene A_1. Genes B_1 and C_1 are outparalogs w.r.t. G_3 (*i.e.*, the 2nd speciation), and are inparalogs w.r.t. G_1 (*i.e.*, the 1st speciation). Gene C_2 is the direct descendant (*i.e.*, true exemplar) of the ancestral gene C while C_3 is not, if C_3 is duplicated from C_2. C_1 and C_2 are said to form a pair of main orthologs.

commonly exists a gene that is the direct descendant of the ancestral gene of the set, namely the one that best reflects the original position of the ancestral gene in the ancestral genome. Sankoff [17] called such a gene the *true exemplar* of the inparalogous set. Given two genomes, two sets of inparalogous genes (one from each genome) are *co-orthologous* if they are descendants of the same ancestral gene at the time of speciation. These concepts are illustrated in Figure 1.

Clearly, orthologs are evolutionary and, typically, functional counterparts in different species. Therefore, many existing computational methods for solving various biological problems, *e.g.*, the inference of functions of new genes and the analysis of phylogenetic relationship between different species, use orthologs in a critical way. A major complication with the use of orthologs in these methods, however, is that orthology is not necessarily a one-to-one relationship because a single gene in one phylogenetic lineage may correspond to a whole family of inparalogs in another lineage. More caution should be taken while such one-to-many and many-to-many relationships are applied to the transfer of functional assignments because inparalogs could have acquired new functions during the course of evolution. As a consequence, the identification of orthologs and inparalogs, especially those one-to-one orthology relationships, is critical for evolutionary and functional genomics, and thus a fundamental problem in computational biology.

It follows from the definition of orthologs and paralogs that the best way to identify orthologs is to measure the divergence time between homologous genes in two different genomes. As the divergence time could be estimated by comparing the DNA or protein sequences of genes, most of the existing algorithms for ortholog assignment, such as the well-known COG system [21][23] and INPARANOID program [16], rely mainly on sequence similarity (usually measured via BLAST scores [1]). An implicit, but often questionable, assumption behind these methods is that the evolutionary rates of all genes in a homologous family are equal. Incorrect ortholog assignments might be obtained if the real rates of evolution vary significantly between paralogs. On the other hand, we observe

that molecular evolution proceeds in two different forms: local mutation and global rearrangement. Local mutations include base substitution, insertion and deletion, and global rearrangements include reversal (*i.e.* inversion), translocation, fusion, fission and so on. Apparently, the sequence similarity-based methods for ortholog assignment make use of local mutations only and neglect genome rearrangement events that might contain a lot of valuable information.

In our recent papers [4][5], we initiated the study of ortholog assignment via genome rearrangement and proposed an approach that takes advantage of evolutionary evidences from both local mutations and global rearrangements. It begins by identifying homologous gene families on each genome and the correspondence between families on both genomes using homology search. The homologs are then treated as copies of the same genes, and ortholog assignment is formulated as a natural optimization problem of rearranging one genome consisting of a sequence of (possibly duplicated) genes into the other with the minimum number of reversals, where the most parsimonious rearrangement process should suggest orthologous gene pairs in a straightforward way. A high-throughput system, called SOAR, was implemented based on this approach. Though our preliminary experiments on simulated data and real data (the X chromosomes of human and mouse) have demonstrated that SOAR is very promising as an ortholog assignment method, it has the drawback of ignoring the issue of inparalogs. In fact, it assumed that there were no gene duplications after the speciation event considered. As a consequence, it only outputs one-to-one orthology relationships and every gene is forced to form an orthologous pair. Moreover, it is only able to deal with unichromosomal genomes. In this paper, we present several improvements that are crucial for more accurate ortholog assignment. In particular, the method will be extended to deal with inparalogs explicitly by incorporating a more realistic evolutionary model that allows duplication events after the speciation event. In summary, our main contributions in this study include

- We introduce a subtype of orthologs, called *main orthologs*, to delineate sets of co-orthologous genes. For two inparalogous sets of co-orthologous genes, the main ortholog pair is simply defined as the two true exemplar genes of each set (see Figure 1 for an example).[1] Since a true exemplar is a gene that best reflects the original position of the ancestral gene in the ancestral genome, main orthologs are therefore the *positional counterpart* of orthologs in different species. By definition, main orthologs form a one-to-one correspondence, thus allowing for the possibility of direct transfer of functional assignments. We believe that, compared with other types of ortholog pairs, main orthologs are more likely to be functional counterparts in different species, since they are both evolutionary and positional counterparts.
- In our previous study, the evolutionary model assumes that there is no gene duplication subsequent to the given speciation event. Thus, no inparalogs are assumed to be present in the compared genomes, which is clearly inappropriate for nuclear genomes. In this paper, we propose a parsimony approach

[1] Note that, our definition of a main ortholog pair is different from the one in [16], where it refers to a mutually best hit in an orthologous group.

based on a more realistic evolutionary model that combines both rearrangement events (including reversal, translocation, gene fusion and gene fission) and gene duplication events. This will allow us to treat inparalogs explicitly. More specifically, in order to assign orthologs, we reconstruct an evolutionary scenario since the splitting of the two input genomes, by minimizing the (total) number of reversals, translocations, fusions, fissions and duplication events necessary to transform one genome into the other (*i.e.*, by computing the *rearrangement/duplication* distance between two genomes). Such a most parsimonious evolutionary scenario should reveal main ortholog pairs and inparalogs in a straightforward way.

– Computing the rearrangement/duplication distance between two genomes is known to be very hard. We have developed an efficient heuristic algorithm that works well on large multichromosomal genomes like human and mouse. We strengthen and extend some of the algorithmic techniques developed in [4][5], including (sub)optimal assignment rules, minimum common partition, and maximum graph decomposition, as well as a new post-processing step that removes "noise" gene pairs that are most likely to consist of inparalogs.
– Based on the above heuristic algorithm, we have implemented a high-through put system for automatic assignment of (main) orthologs and the detection of inparalogs on a genome scale, called MSOAR. By testing it on simulated data and human and mouse genomes, the MSOAR system is shown to be quite effective for ortholog assignment. For example, it is able to find 99 more true ortholog pairs between human and mouse than INPARANOID [16].

Related work. In the past decade, many computational methods for ortholog assignment have been proposed, most of which are based primarily on sequence similarity. These methods include the COG system [21][23], EGO (previously called TOGA)[11], INPARANOID [16], and OrthoMCL [12], just to name a few. Some of these methods combine sequence similarity and a parsimony principle, such as the reconciled tree method [25] and the bootstrap tree method [20], or make use of synteny information, such as OrthoParaMap [3] and the recent method proposed by Zheng *et al.* [26]. However, none of these papers use genome rearrangement. On the other hand, there have been a few papers in the literature that study rearrangement between genomes with duplicated genes, which is closely related to ortholog assignment. Sankoff [17] proposed an approach to identify the true exemplar gene of each gene family, by minimizing the breakpoint/reversal distance between two reduced genomes that consist of only true exemplar genes. El-Mabrouk [14] developed an approach to reconstruct an ancestor of a modern genome by minimizing the number of duplication transpositions and reversals. The work in [13][18] attempts to find a one-to-one gene correspondence between gene families based on conserved segments. Very recently, Swenson *et al.* [19] presented some algorithmic results on the cycle splitting problem in a combinatorial framework similar to the one introduced in [4][5].

The rest of the paper is organized as follows. We first discuss the parsimony principle employed in our ortholog assignment approach in Section 2. Section 3 describes the heuristic algorithm implemented in MSOAR. Section 4 will present

our experiments on simulated data and on the whole genome data of human and mouse. Finally, some concluding remarks are given in Section 5.

2 Assigning Orthologs Under Maximum Parsimony

The two genomes to be compared, denoted as Π and Γ, have typically undergone series of genome rearrangement and gene duplication events since they split from their last common ancestral genome. Clearly, we could easily identify main orthologs and inparalogs if given such an evolutionary scenario. Based on this observation, we propose an approach to reconstruct the evolutionary scenario on the basis of the parsimony principle, *i.e.*, we postulate the minimal possible number of rearrangement events and duplication events in the evolution of two genomes since their splitting so as to assign orthologs. Equivalently, it can be formulated as a problem of finding a most parsimonious transformation from one genome into the other by genome rearrangements and gene duplications, without explicitly inferring their ancestral genome. Let $R(\Pi, \Gamma)$ and $D(\Pi, \Gamma)$ denote the number of rearrangement events and the number of gene duplications in a most parsimonious transformation, respectively, and $RD(\Pi, \Gamma)$ denotes the *rearrangement/duplication (RD) distance* between Π and Γ satisfying $RD(\Pi, \Gamma) = R(\Pi, \Gamma) + D(\Pi, \Gamma)$. Most genome rearrangement events will be considered in this study, including reversal, translocation, fusion and fission.

In practice, we will impose two constraints on this optimization problem, based on some biological considerations. First, we require that at least one member of each family that appears in the other genome be assigned orthology, because each family should provide an essential function and the gene(s) retaining this function is more likely conserved during the evolution. Second, observe that the assignment of orthologs that leads to the minimum rearrangement/duplication distance is not necessarily unique. Therefore, among all assignments with the minimum rearrangement/duplication distance, we attempt to find one that also minimizes $R(\Pi, \Gamma)$, in order to avoid introducing unnecessary false orthologous pairs.

Figure 2 presents a simple example to illustrate the basic idea behind our parsimony approach. Consider two genomes, $\Pi = -b - a_1 + c + a_2 + d + a_4 + e + f + g$ and $\Gamma = +a_1 + b + c + a_2 + d + e + a_5 + f + a_3 + g$, sharing a gene family a with multiple copies. As shown in Figure 2, both genomes evolved from the same ancestral genome $+a + b + c + d + e + f + g$, Π by one inversion and one gene duplication and Γ by two gene duplications, respectively. By computing the rearrangement/duplication distance $RD(\Pi, \Gamma) = 4$, the true evolutionary scenario can be reconstructed, which then suggests that the two genes a_1 form a pair of main orthologs, as well as the two genes a_2. Meanwhile, a_3, a_4, and a_5 are inferred as inparalogs that were derived from duplications after the speciation event. It is interesting to see that here a_4 is not assigned orthology to a_3 or a_5 greedily. (Note that they are orthologs, but not main orthologs, by our definition.) This simple example illustrates that, by minimizing the reversal/duplication distance, our approach is able to pick correct main orthologs out of sets of inparalogs.

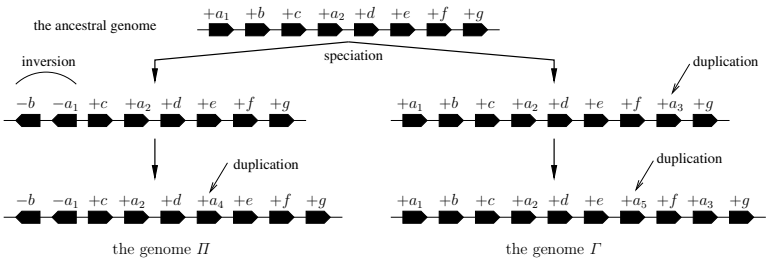

Fig. 2. An evolutionary history of two genomes Π and Γ. Π evolved from the ancestor by one inversion and one gene duplication, and Γ by two duplications.

Note that, although gene loss may occur in the course of evolution, it actually has no impact on the capability of assigning ortholog by our method. If an inparalog is lost, the gene loss event can be simply ignored and this will not affect ortholog assignment. If some gene of a main ortholog pair is lost, our approach attempts to identify the other gene as an inparalog rather than to assign it a wrong orthology, which also makes some sense especially when considering the transfer of functional assignment.

3 The MSOAR System

Following the parsimony principle discussed in the previous section, we have implemented a high-throughput system for ortholog assignment, called MSOAR. It employs a heuristic to calculate the rearrangement/duplication distance between two genomes, which can be used to reconstruct a most parsimonious evolutionary scenario. In this section, we discuss in detail the heuristic algorithm.

We represent a gene by a symbol of some finite alphabet $\mathcal{A}$, and its orientation by the sign + or −. A chromosome is a sequence of genes, while a genome is a set of chromosomes. Usually, a genome is represented as a set $\Pi = \{\pi(1), \cdots, \pi(N)\}$, where $\pi(i) = \langle \pi(i)_1 \cdots \pi(i)_{n_i} \rangle$ is a sequence of oriented genes in the ith chromosome. Recall the genome rearrangement problem between two genomes with distinct oriented genes. Hannenhalli and Pevzner developed algorithms for calculating genome rearrangement distance on both unichromosomal [7] and multichromosomal genomes [8] in polynomial time. The rearrangement distance between multichromosomal genomes is the minimum number of *reversals*, *translocations*, *fissions* and *fusions* that would transform one genome into the other. Given two multichromosomal genomes Π and Γ, Hannenhalli and Pevzner [8] gave a formula for calculating the genome rearrangement distance (called the HP formula in this paper). Tesler [22], and Ozery-Flato and Shamir [15] then suggested some corrections to the formula (called the revised HP formula):

$$d(\Pi,\Gamma) = b(\Pi,\Gamma) - c(\Pi,\Gamma) + p_{\Gamma\Gamma}(\Pi,\Gamma) + r(\Pi,\Gamma) + \left\lceil \frac{s'(\Pi,\Gamma) - gr'(\Pi,\Gamma) + fr'(\Pi,\Gamma)}{2} \right\rceil$$

where $b(\Pi,\Gamma)$ is the number of black edges in the breakpoint graph $G(\Pi,\Gamma)$, $c(\Pi,\Gamma)$ is the overall number of cycles and paths, $p_{\Gamma\Gamma}(\Pi,\Gamma)$ is the number of

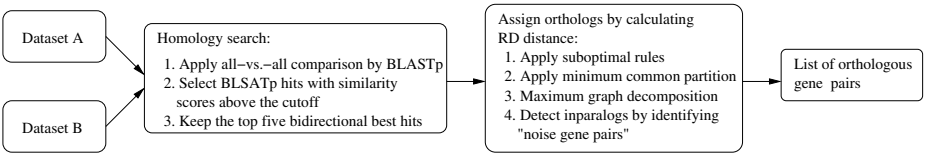

Fig. 3. An outline of MSOAR

the $\Gamma\Gamma$-paths, and r, s', gr' and fr' are some parameters in terms of real-knots [15]. In practice, the dominant parts of the formula are the first three terms.

When the genomes Π and Γ contain duplicated genes, however, the rearrangement/duplication distance problem (*i.e.* $RD(\Pi, \Gamma)$) cannot be directly solved by the revised HP formula. In fact, we can prove that computing the rearrangement/duplication distance is NP-hard by a reduction similar to the one employed in the proof of Theorem 4.2 of [5]. Note that, once the main ortholog pairs are assigned and the inparalogs are identified, the rearrangement/duplication distance can be easily computed as follows. The number of duplications is determined by the number of inparalogs. After removing all the inparalogs, the rearrangement distance between the reduced genomes, which now have equal gene content, can be computed using the above formula since every gene can be regarded as unique. An outline of MSOAR is illustrated in Figure 3.

3.1 Homology Search and Gene Family Construction

MSOAR starts by calculating the pairwise similarity scores between all gene sequences of the two input genomes. An all-*versus*-all gene sequence comparison by BLASTp is used to accomplish this. As in [16], two cutoffs are applied to each pair of BLASTp hits. Two genes are considered homologous if (1) the bit score is no less than 50 and (2) the matching segment spans above 50% of each gene in length. In order to eliminate potential false main ortholog pairs, we take the top five bidirectional best hits of each gene as its potential main orthologs if their logarithmic E-value is less than the 80% of the best logarithmic E-value. By clustering homologous genes using the standard single linkage method, we obtain gene families. A gene family is said to be *trivial* if it has cardinality exactly 2, *i.e.* with one occurrence in each genome. Otherwise it is said to be non-trivial. A gene belonging to a trivial (or non-trivial) family is said to be trivial (or non-trivial, resp.). We use a *hit graph* (denoted as $\mathcal{H}$) to describe the relationship between genes within each family. A hit graph is a bipartite graph illustrating the BLASTp hits between two genomes. Each vertex represents a gene and an edge connects two vertices from different genomes if they are potential main orthologs. Figure 4 gives an example of the hit graph. Adjacent genes in the hit graph are regarded as candidates for main ortholog pairs.

3.2 (Sub)Optimal Assignment Rules

We presented three assignment rules for identifying individual ortholog assignments that are (nearly) optimal in SOAR [4][5]. In MSOAR, we will add two

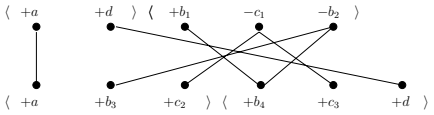

Fig. 4. A hit graph for genomes $\Pi = \{\langle +a, +d\rangle, \langle +b_1, -c_1, -b_2\rangle\}$ and $\Gamma = \{\langle +a, +b_3, +c_2\rangle, \langle +b_4, +c_3, +d\rangle\}$, each having two chromosomes

more assignment rules, which could make the system more efficient. The four rearrangement operations (reversal, translocation, fission and fusion) can be mimicked by reversals when we represent a multichromosomal genome by a *concatenate* [8][22]. This approach reduces the problem of computing rearrangement distance between two multichromosomal genomes to the problem of computing the reversal distance between two *optimal concatenates*. Since these two new rules are only concerned with segments of consecutive genes within a single chromosome, which also form gene segments in an optimal concatenate, the unichromosomal HP formula [7] can be used to prove their (sub)optimality. Let G and H be two chromosomes in genomes Π and Γ, respectively. A *chromosome segment* is defined as a substring of some chromosome (*i.e.* a consecutive sequence of genes). A chromosome segment $(g_{i_1} g_{i_2} \cdots g_{i_n})$ in G *matches* a chromosome segment $(h_{j_1} h_{j_2} \cdots h_{j_n})$ in H if g_{i_t} and h_{j_t} are connected by an edge in the hit graph and have the same orientations for all $1 \leq t \leq n$.

Theorem 1. *Assume that a chromosome segment $(g_{i_1} g_{i_2} \cdots g_{i_n})$ in G, matches a chromosome segment $(h_{j_1} h_{j_2} \cdots h_{j_n})$ in H or its reversal, where g_{i_1} ,g_{i_n}, h_{j_1} and h_{j_n} are trivial but the other genes are not. Define two new genomes Π' and Γ' by assigning orthology between g_{i_t} and h_{j_t} or g_{i_t} and $g_{j_{n+1-t}}$ (in the case of matching by a reversal), for all $1 \leq t \leq n$. Then, $RD(\Pi, \Gamma) \leq RD(\Pi', \Gamma') \leq RD(\Pi, \Gamma) + 2$.*

Theorem 2. *Assume that for a chromosome segment $(g_{i_1} g_{i_2} \cdots g_{i_n})$ in G and a chromosome segment $(h_{j_1} h_{j_2} \cdots h_{j_n})$ in H, g_{i_1} matches h_{j_1}, g_{i_n} matches h_{j_n}, and $g_{i_2} \cdots g_{i_{n-1}}$ matches the reversal of $h_{j_2} \cdots h_{j_{n-1}}$, where g_{i_1}, g_{i_n}, h_{j_1} and h_{j_n} are trivial but the other genes are not. Define two new genomes Π' and Γ' by assigning orthology between g_{i_t} and $g_{j_{n+1-t}}$, for all $1 < t < n$. Then, $RD(\Pi, \Gamma) \leq RD(\Pi', \Gamma') \leq RD(\Pi, \Gamma) + 2$.*

3.3 Minimum Common Partition

We extend the *minimum common partition* (MCP) problem, which was first introduced in [4][5] to reduce the number of duplicates of each gene in ortholog assignment, to multichromosomal genomes. Use $\overline{\pi(i)}_j$ to represent a chromosome segment or its reversal in chromosome i of genome Π. A *chromosome partition* is a list $\{\overline{\pi(i)}_1, \overline{\pi(i)}_2, \cdots, \overline{\pi(i)}_n\}$ of chromosome segments such that the concatenation of the segments (or their reversals) in some order results in the chromosome i. A *genome partition* is the union of some partitions of all the chromosomes. A list of chromosome segments is called a *common partition* of

two genomes Π and Γ if it is a partition of both Π and Γ. Furthermore, a *minimum common partition* is a partition with the minimum cardinality (denoted as $L(\Pi, \Gamma)$) over all possible common partitions of Π and Γ. The MCP problem is the problem of finding the minimum common partition between two given genomes. Two genomes have a common partition if and only if they have equal gene content (*i.e.* they have the same number of duplications for each gene).

We can further extend MCP to an arbitrary pair of genomes that might have unequal gene contents. A *gene matching* $\mathcal{M}$ between genomes Π and Γ is a matching between the genes of Π and Γ, which can be defined by a maximum matching in their hit graph $\mathcal{H}$. Given a gene mathing $\mathcal{M}$, two reduced genomes (denoted as $\tilde{\Pi}_{\mathcal{M}}$ and $\tilde{\Gamma}_{\mathcal{M}}$) with equal gene content can be obtained by removing all the unmatched genes. The minimum common partition of Π and Γ is defined as the minimum $L(\tilde{\Pi}_{\mathcal{M}}, \tilde{\Gamma}_{\mathcal{M}}))$ among all gene matchings $\mathcal{M}$.

Given two genomes Π and Γ, recall that $RD(\Pi, \Gamma)$ is the rearrangement/duplication distance between them. Let N_u be the number of unmatched genes introduced by a gene matching and N_c be the number of chromosomes. Based on the fact that inserting two genes into the two genomes under consideration, one for each genome, will increase the rearrangement distance by at most three, the following theorem can be obtained to establish the relationship between the minimum common partition and the rearrangement/duplication distance.

Theorem 3. *For any two genomes* Π *and* Γ, $(L(\Pi, \Gamma) - N_c - 2)/3 + N_u \leq RD(\Pi, \Gamma) \leq L(\Pi, \Gamma) + 2N_c + N_u + 1$.

An efficient heuristic algorithm for MCP on unichromosomal genomes was given in [4][5]. The algorithm constructs the so called "pair-match" graphs and then attempts to find a large independent set. We extend the method to multichromosomal genomes in a straightforward way.

3.4 Maximum Graph Decomposition

After minimum common partition, the genomes Π and Γ may still contain duplicates, although the number of duplicates is expected to be small. In order to match all the genes, we define another problem, called *maximum graph decomposition* (MGD). The MGD problem is: among all pairs of reduced genomes of Π and Γ obtained by all possible gene matchings, find one with the maximum value of $c(\Pi, \Gamma) - p_{\Gamma\Gamma}(\Pi, \Gamma)$.

Using the basic framework developed in [4][5], we design a greedy algorithm in MSOAR to solve MGD using a new graph, called the *complete-breakpoint graph*. The complete-breakpoint graph associated with Π and Γ is denoted as $\mathcal{G}$, which is adapted from the breakpoint graph of multichromosomal genomes of equal gene content consisting of only singletons [8]. The prefix "complete" is added here to differentiate from the partial graphs in [4][5]. If Π and Γ have different numbers of chromosomes, add null chromosomes to the genome with fewer chromosomes to make them both have N_c chromosomes. As defined in [8], a *cap* is used as a marker that serves as a chromosomal end delimiter when we convert a multichromosomal genome into a unichromosomal genome. A *capping*

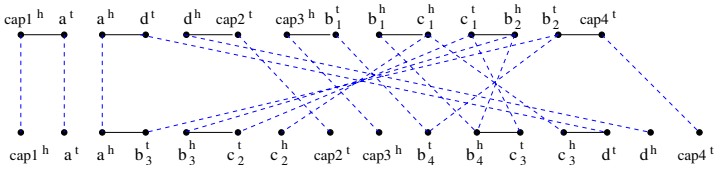

Fig. 5. The complete-breakpoint graph of two genomes with unequal gene contents $\Pi = \{\langle +a, +d\rangle, \langle +b_1, -c_1, -b_2\rangle\}$ and $\Gamma = \{\langle +a, +b_3, +c_2\rangle, \langle +b_4, +c_3, +d\rangle\}$. The hit graph of these two genomes is shown in Figure 4.

of a chromosome $\pi(i)$ is $\pi(i) = \langle \pi(i)_0 \pi(i)_1 \cdots \pi(i)_{n_i} \pi(i)_{n_i+1} \rangle$, where $\pi(i)_0$ is the left cap of $\pi(i)$, called *lcap*, and $\pi(i)_{n_i+1}$ is the right cap, called *rcap*. Choose any capping and an arbitrary concatenation to transform Π and Γ into unichromsomal genomes $\hat{\pi}$ and $\hat{\gamma}$. In the complete-breakpoint graph $\mathcal{G}$, every gene or cap g from each genome is represented by two ordered vertices α^t α^h if α is positive or α^h α^t if it is negative. Insert black edges between vertices that correspond to adjacent genes (or caps) in the same genome except the pairs of the form α_i^h and α_i^t from the same gene (or cap) α_i. Insert cross-genome grey edges ($\hat{\pi}_i^t$,$\hat{\gamma}_j^t$) and ($\hat{\pi}_i^h$, $\hat{\gamma}_j^h$) if gene $\hat{\pi}_i$ and gene $\hat{\gamma}_j$ are connected by an edge in the hit graph or they are the same caps. Next, we delete the left vertex of every lcap, the right vertex of every rcap and all the edges incident on them. The calculation of RD distance using the resulting graph no longer depends on the actual concatenations. Finally, we make the complete-breakpoint graph independent on the capping of Γ, by deleting the $2N_c$ black edges incident on the $\hat{\gamma}$ cap vertices. These cap vertices are called Π-caps. The vertex on the other end of a deleted black edge is called a Γ-tail unless the black edge arises from a null chromosome, in which case both of its ends are Π-caps. An example of the complete-breakpoint graph is shown in Figure 5. The complete-breakpoint graph contains both cycles and paths. Depending on whether the end points are both Π-caps, both Γ-tails or one of each, a path could be classified as a $\Pi\Pi$-path,$\Gamma\Gamma$-path or $\Pi\Gamma$-path.

After the complete-breakpoint graph is constructed, we try to find small cycles and short $\Pi\Gamma$-paths first, and then finish the decomposition by finding the rest of $\Pi\Pi$-paths and $\Gamma\Gamma$-paths. The decomposition has to satisfy the following three conditions: (1) every vertex belongs to at most one cycle or path (2) the two vertices representing each gene must be connected respectively to the two vertices of a single gene in the other genome by edges of the cycles or paths, otherwise both must be removed, *i.e.*, the connections satisfy a pairing condition; and (3) the edges within a genome and across genomes alternate in a cycle or a path. Intuitively, small cycles may lead to large cycle decompositions, although it is not always the case. Moreover, the more $\Pi\Gamma$-paths, the fewer $\Gamma\Gamma$-paths, because the number of Γ-tail vertices is fixed and each vertex can only belong to at most one path. Note that during the cycle decomposition, some gene vertices might have all of their cross-genome edges removed since Π and Γ may have unequal gene contents and these genes are regarded as inparalogs. If two gene vertices α_i^t and α_i^h have no cross genome edges incident on them during the cycle decomposition, they need to be removed from the complete-breakpoint graph right away and a

black edge need to be inserted between two endpoints of the deleted black edges arising from α_i^t and α_i^h.

Any feasible solution of the MCD problem gives a maximal matching between the genes of Π and the genes of Γ. The genes that have not been matched will be assigned as inparalogs of the matched ones in the same family. The matched genes suggest possible main ortholog pairs and a rearrangement scenario to transform Π to Γ by the operations reversal, translocation, fusion and fission.

3.5 "Noise" Gene Pairs Detection

The maximum graph decomposition of a complete-breakpoint graph $\mathcal{G}$ determines a one-to-one gene matching between two genomes. Unmatched genes are removed since either they are potential inparalogs or their orthology counterparts were lost during the evolution. However, some individual paralogs might be forced to be assigned as main ortholog pairs because the maximum graph decomposition always gives a maximal matching between all the genes. Therefore, it is necessary to remove these "noise" gene pairs so that the output main ortholog pairs are more reliable.

After removing the unmatched genes, we obtain two reduced genomes with equal gene content. Remove all the gene pairs whose deletion would decrease the rearrangement distance of reduced genomes by at least two. Note that, in this case, the rearrangement/duplication distance will never increase since the deletion of a gene pair may only increase the number of duplications required in an optimal scenario by two. As mentioned before, we require that at least one main ortholog pair of each gene family be kept during this post-processing.

MSOAR combines the suboptimal ortholog assignment rules, heuristic MCP algorithm, heuristic MGD algorithm, and "noise" gene pair detection step to find all the potential main ortholog pairs and detect inparalogs.

4 Experiments

In order to test the performance of MSOAR as a tool of assigning orthologs, we have applied it to both simulated and real genome sequence data, and compared its results with two well-known algorithms in the literature, namely, an iterated version of the exemplar algorithm and INPARANOID.

4.1 Simulated Data

In order to assess the validity of our parsimony principle as a means of distinguishing main orthologs from inparalogs, we conduct two simple experiments to estimate the probability of inparalogs that may incorrectly be assigned orthology by transforming one genome into another with the minimum number of rearrangement and duplication events. The first experiment is done as follows. First, we simulate a genome G with 100 distinct genes, and then randomly perform k reversals on G to obtain another genome H. The boundaries of these reversals are uniformly distributed within the genome. Next, make a copy of

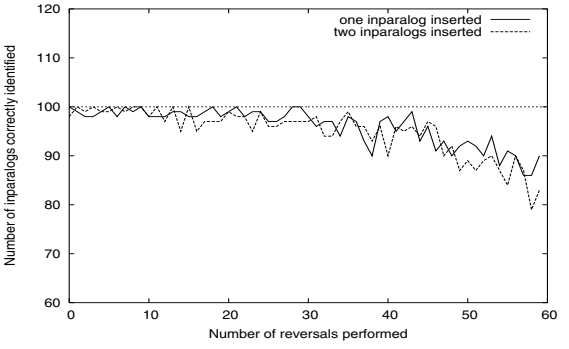

Fig. 6. Distribution of the number of inparalogs correctly identified by the parsimony principle

some gene that is randomly selected from H and insert it back into H as a duplicate. Clearly, the inserted gene is an inparalog, by definition. In this case, there are only two possible ortholog assignments between G and H. Therefore, we can easily calculate the rearrangement/duplicate distance between G and H, and know the ortholog assignment that will be made by the parsimony principle. We repeat the above procedure on 100 random instances for each k, $0 \leq k < 60$, and count the number of instances for which the inparalogs are correctly identified. The distribution of these inparalogs against the number k of reversals is plotted in Figure 6 (the curve marked "one inparalog inserted"). The result shows that with a very high probability ($> 90\%$, for each $k < 55$) the main ortholog and inparalog can be correctly identified. This suggests that an orthology assignment between two genes that are the positional counterparts between two genomes tends to result in the smaller rearrangement/duplication distance, compared to the distance given by an assignment involving inparalogs.

The second experiment is conducted to estimate the probability that two inparalogs, one from each genome, would be identified as a main ortholog pair, instead of as two individual inparalogs. Its data is generated similarly as the first experiment, except that a same copy of the gene is also inserted into genome G, resulting in a non-trivial gene family of size four in both genomes. As before, we count the number of instances for which two inparalogs are correctly identified, and plot its distribution in Figure 6 (the curve marked "two inparalogs inserted"). The result shows that it is very unlikely that two inparalogs from different genomes are assigned as a (main) ortholog pair. This and the above findings provide some basic support for the validity of using the parsimony approach to identify main orthologs.

We further use simulated data to assess the performance of our heuristic algorithm for ortholog assignment. In order to make a comparison test, we implemented the exemplar algorithm [17] and extended it into a tool for ortholog assignment as described in [5], called the iterated exemplar algorithm. The simulated data is generated as follows. Start from a genome G with n distinct symbols whose signs are generated randomly. Each symbol defines a single gene family.

Then randomly combine two gene families into a new family until r singletons are left in the genome G. Perform k reversals on G to obtain a genome H as in the previous experiments. Finally, randomly insert c inparalogs (each is a copy of some gene randomly selected) into the two genomes. Note that some singletons may be duplicated during this step so that more non-trivial gene families could be generated. The quadruple (n, r, k, c) specifies the parameters for generating two genomes as test data.

We run the iterated exemplar algorithm [17][5] and our heuristic algorithm on 20 random instances for each combination of parameters. The average performance of both algorithms is shown in Figure 7, in terms of the number of incorrectly assigned orthologs (*i.e.*, genes in a genome that are not assigned orthology to their positional counterparts in the other genome) and inparalogs. As we can see, our heuristic algorithm is quite reliable in assigning orthologs and identifying inparalogs. On average, the number of incorrect assignments generally increases as the number of reversals k increases. While both algorithms perform equally well for inparalogous gene identification, our heuristic algorithm produces fewer incorrect ortholog assignments than the iterated exemplar algorithm, especially for the instances generated using parameters $n = 100$, $r = 80$, and $c = 5$ (see Figure 7).

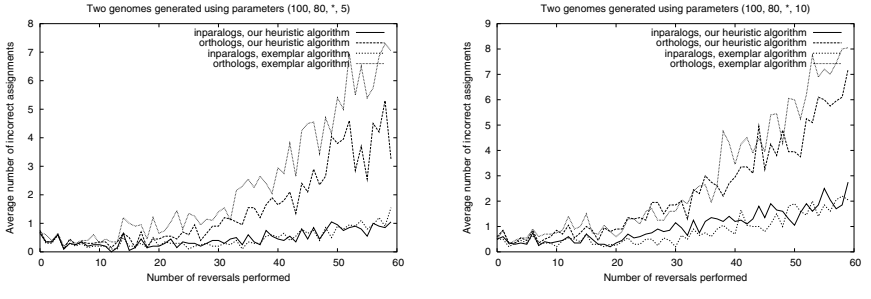

Fig. 7. Comparison of our heuristic and the exemplar algorithm on simulated data

4.2 Real Data

We consider two model genomes: Human (*Homo sapiens*) and Mouse (*Mus musculus*). Gene positions, transcripts and translations were downloaded from the UCSC Genome Browser [9] web site (http://genome.ucsc.edu). We used the canonical splice variants from the Build 35 human genome assembly (UCSC hg17, May 2004) and the Build 34 assembly of the mouse genome (UCSC mm6, March 2005). There are 20181 protein sequences in human genome assembly hg17 and 17858 sequences in mouse genome assembly mm6. Due to assembly errors and other reasons, 220 human and 114 mouse genes were mapped to more than one location in the respective genomes. For such a gene, we kept the first transcription start position which is closest to the $5'$ end as its start coordinate. A homology search was then performed and a hit graph between human and mouse built as described in Section 3.1.

As shown in Table 1, before removing "noise" gene pairs, MSOAR assigned 13395 main orthologs pairs between human and mouse. Then MSOAR removed 177 "noise"gene pairs and output 13218 main orthologs pairs. The distribution of the number of orthologs assigned by MSOAR between human chromosomes and mouse chromosomes is illustrated in Fig 8. It shows that the top 3 chromosome pairs between human and mouse with the largest numbers of orthologs are human chromosome 17 vs. mouse chromosome 11, human chromosome 1 vs. mouse chromosome 4, and human chromosome X vs. mouse chromosome X, which are consistent with the Mouse Human synteny alignments. (http://www.sanger.ac.uk/Projects/M_musculus/publications/fpcmap-2002/mouse-s.shtml).

We validate our assignments by using gene annotation, in particular, gene names. To obtain the most accurate list of gene names, we have cross-linked database tables from the UCSC Genome Browser with gene names extracted from UniProt [2] release 6.0 (September 2005). The official name of a gene is usually given to convey the character or function of the gene [24]. Genes with identical names are most likely to be an orthologous pair, although we should keep in mind that many names were given mostly based on sequence similarity and erroneous/inconsistent names are known to exist in the annotation. Some genes have names beginning with "LOC" or ending with "Rik" or even have no names, implying that these genes have not yet been assigned official names or their functions have not been validated. If a pair of genes output by MSOAR have completely identical gene symbol, we count them as a true positive pair; if they have different names without substring "LOC"or "Rik", it is a false positive pair; otherwise, it is counted as an unknown pair. We also calculate the total number of *assignable* pairs of orthologs, *i.e.* the total number of pairs of genes

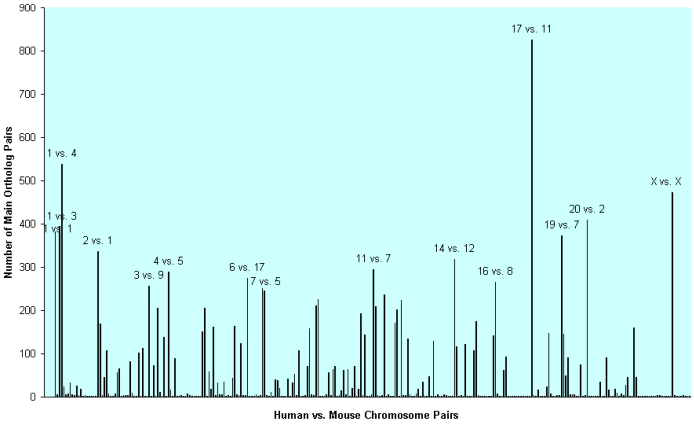

Fig. 8. Distribution of the number of ortholog pairs assigned by MSOAR across all pairs of the human and mouse chromosomes. The chromosome pairs with more than 250 main ortholog pairs are labeled. *E.g.*, the highest bar is human chromosome 17 vs. mouse chromosome 11, between which 825 main ortholog pairs were assigned.

with identical names. For example, there are 9891 assignable orthologous gene pairs between human and mouse. Before removing "noise" gene pairs, MSOAR predicted 13395 ortholog pairs, among which 9263 are true positives, 2171 are unknown pairs and 1961 are false positives, resulting in a sensitivity of 93.65% and a specificity of 81.01%. After removing "noise" gene pairs, MSOAR predicted 13218 ortholog pairs, among which 9214 are true positives, 2126 are unknown pairs and 1878 are false positives, resulting in a sensitivity of 93.16% and a specificity of 81.25%. It is interesting to note that the last step of MSOAR identified 177 "noise" gene pairs, among which 72.32% were false positives. This result shows that the identification of "noise" gene pairs effectively detects false positives and could provide more reliable ortholog assignment.

The comparison result between MSOAR and INPARANOID [16] is shown in Table 1. MSOAR was able to identify 99 more true ortholog pairs than INPARANOID, although it also reported more false positives.

Table 1. Comparison of ortholog assignments between MSOAR and INPARANOID

	assignable	assigned	true positive	unkown
MSOAR (before removing "noise" pairs)	9891	13395	9263	2171
MSOAR (after removing "noise" pairs)	9891	13218	9214	2126
INPARANOID	9891	12758	9115	2034

5 Concluding Remarks

Although we anticipate that the system MSOAR will be a very useful tool for ortholog assignment, more systematics tests will be needed to reveal the true potential of this parsimony approach. Our immediate future work includes the incorporation of transpositions into the system and consideration of weighing the evolutionary events.

Acknowledgment

This project is supported in part by NSF grant CCR-0309902, a DoE GtL subcontract, National Key Project for Basic Research (973) grant 2002CB512801, NSFC grant 60528001, and a fellowship from the Center for Advanced Study, Tsinghua University. Also, the authors wish it to be known that the first two authors should be regarded as joint First Authors of this paper. The contact email addresses are {zfu,jiang}@cs.ucr.edu

References

1. S. Altschul *et al.* Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Research*, vol. 25, no. 17, pp 3389-3402, 1997.
2. A. Bairoch *et al.* The Universal Protein Resource (UniProt). *Nuc. Acids Res.* 33:D154-D159, 2005.

3. S.B. Cannon and N.D. Young. OrthoParaMap: distinguishing orthologs from paralogs by integrating comparative genome data and gene phylogenies. *BMC Bioinformatics* 4(1):35, 2003.
4. X. Chen, J. Zheng, Z. Fu, P. Nan, Y. Zhong, S. Lonardi, and T. Jiang. Computing the assignment of orthologous genes via genome rearrangement. In *Proc. 3rd Asia Pacific Bioinformatics Conf. (APBC'05)*, pp. 363-378, 2005.
5. X. Chen, J. Zheng, Z. Fu, P. Nan, Y. Zhong, S. Lonardi, and T. Jiang The assignment of orthologous genes via genome rearrangement. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 2, no. 4, pp. 302-315, 2005.
6. W.M. Fitch. Distinguishing homologous from analogous proteins. *Syst. Zool.* 19: 99-113, 1970.
7. S. Hannenhalli and P. Pevzner. Transforming cabbage into turnip (polynomial algorithm for sorting signed permutations by reversals). *Proc. 27th Ann. ACM Symp. Theory of Comput.* (STOC'95), pp. 178-189, 1995.
8. S. Hannenhalli, P. Pevzner. Transforming men into mice (polynomial algorithm for genomic distance problem). *Proc. IEEE 36th Symp. Found. of Comp. Sci.*, 581-592, 1995.
9. D. Karolchik, K.M. Roskin, M. Schwartz, C.W. Sugnet, D.J. Thomas, R.J. Weber, D. Haussler and W.J. Kent. The UCSC Genome Browser Database. *Nucleic Acids Res.*, vol. 31, no. 1, pp. 51-54, 2003.
10. E. Koonin. Orthologs, paralogs, and evolutionary genomics. *Annu. Rev. Genet.*, 2005.
11. Y. Lee *et al.* Cross-referencing eukaryotic genomes: TIGR orthologous gene alignments (TOGA). *Genome Res.*, vol. 12, pp. 493-502, 2002.
12. L. Li, C. Stoeckert, D. Roos. OrthoMCL: identification of ortholog groups for eukaryotic genomes. *Genome Res.*, vol. 13, pp. 2178-2189, 2003.
13. M. Marron, K. Swenson, and B. Moret. Genomic distances under deletions and insertions. *Theoretic Computer Science*, vol. 325, no. 3, pp. 347-360, 2004.
14. N. El-Mabrouk. Reconstructing an ancestral genome using minimum segments duplications and reversals. *Journal of Computer and System Sciences*, vol. 65, pp. 442-464, 2002.
15. M. Ozery-Flato and Ron Shamir. Two notes on genome rearragnements. *Journal of Bioinformatics and Computational Biology*, Vol. 1, No. 1, pp. 71-94, 2003.
16. M. Remm, C. Storm, and E. Sonnhammer. Automatic clustering of orthologs and in-paralogs from pairwise species comparisons. *J. Mol. Biol.* 314, 1041-1052, 2001.
17. D. Sankoff. Genome rearrangement with gene families. *Bioinformatics* 15(11): 909-917, 1999.
18. K. Swenson, M. Marron, J. Earnest-DeYoung, and B. Moret. Approximating the true evolutionary distance between two genomes. *Proc. 7th SIA Workshop on Algorithm Engineering & Experiments*, pp. 121-125, 2005.
19. K. Swenson, N. Pattengale, and B. Moret. A framework for orthology assignment from gene rearrangement data. *Proc. 3rd RECOMB Workshop on Comparative Genomics*, Dublin, Ireland, LNCS 3678, pp. 153-166, 2005.
20. C. Storm and E. Sonnhammer. Automated ortholog inference from phylogenetic trees and calculation of orthology reliability. *Bioinformatics*, vol. 18, no. 1, 2002.
21. R.L. Tatusov, M.Y. Galperin, D.A. Natale, and E. Koonin. The COG database: A tool for genome-scale analysis of protein functions and evolution. *Nucleic Acids Res.* 28:33-36, 2000.
22. G. Tesler. Efficient algorithms for multichromosomal genome rearrangements. *Journal of Computer and System Sciences*, vol. 65, no. 3, pp. 587-609, 2002.

23. R.L. Tatusov, E. Koonin, and D.J. Lipman. A genomic perspective on protein families. *Science*, vol. 278, pp. 631-637, 1997.
24. H.M. Wain, E.A. Bruford, R.C. Lovering, M.J. Lush, M.W. Wright and S. Povey. Guidelines for human gene nomenclature. *Genomics* 79(4), 464-470, 2002.
25. Y.P. Yuan, O. Eulenstein, M. Vingron, and P. Bork. Towards detection of orthologues in sequence databases. *Bioinformatics*, vol. 14, no. 3, pp. 285-289, 1998.
26. X. Zheng *et al.* Using shared genomic synteny and shared protein functions to enhance the identification of orthologous gene pairs. *Bioinformatics* 21(6): 703-710, 2005.