Multiple Whole Genome Alignments Without a Reference Organism

Inna Dubchak[1,2], Alexander Poliakov[1], Andrey Kislyuk[3], Michael Brudno[4*]

[1] Genome Sciences Division, Lawrence Berkeley National Laboratory, Berkeley, CA 94720 and [2] DOE Joint Genome Institutes, Walnut Creek, CA 94598 USA
[3] Department of Computer Science, Georgia Institute of Technology, Atlanta GA 30332 USA
[4] Department of Computer Science, Banting and Best Department of Medical Research, and Centre for Analysis of Genome Evolution and Function, University of Toronto, Toronto, ON M5R 3G4 Canada

[*]Corresponding author:
brudno@cs.toronto.edu
tel: 416-978-2589
fax: 416-978-1455

Running Title: Multiple Whole Genome Alignment
Keywords: Mutliple alignment, comparative genomics, ancestor reconstruction
Manuscript type: Methods

ABSTRACT

Multiple sequence alignments have become one of the most commonly used resources in genomics research. Most algorithms for multiple alignment of whole genomes rely either on a reference genome, against which all of the other sequences are laid out, or require a one-to-one mapping between the nucleotides of the genomes, preventing the alignment of recently duplicated regions. Both approaches have drawbacks for whole genome comparisons. In this paper we present a novel symmetric alignment algorithm. The resulting alignments not only represent all of the genomes equally well, but also include all relevant duplications that occurred since the divergence from the last common ancestor.

Our algorithm, implemented as a part of the VISTA Genome Pipeline (VGP), was used to align seven vertebrate and six drosophila genomes. The resulting whole genome alignments demonstrate a higher sensitivity and specificity than the pairwise alignments previously available through the VGP, and have higher exon alignment accuracy than comparable public whole genome alignments. Of the multiple alignment methods tested, ours performed the best at aligning genes from multi-gene families – perhaps the most challenging test for whole-genome alignments. Our whole-genome multiple alignments are available through the VISTA Browser at http://genome.lbl.gov/vista/index.shtml.

INTRODUCTION

Genome conservation is an essential guide for biologists and bioinformaticians attempting to locate functional elements and formulate biological hypotheses for testing in the laboratory. By searching for highly conserved sequences across multiple species scientists have identified critical functional elements (Bejerano et al. 2004; Pennacchio et al. 2006; Prabhakar et al. 2006). Sequence conservation is commonly used as input to programs that predict genes (Gross and Brent 2006)(Majoros et al. 2005) (Dewey et al. 2004), find transcription factor binding sites (Lenhard et al. 2003) (Moses et al. 2004), and other regulatory elements (Abbasi et al. 2007) (de la Calle-Mustienes et al. 2005). The conservation signal used by all of these applications is based on alignments between the input genomic sequences.

The first tools developed for alignment of longer genomic regions, such as GLASS (Batzoglou et al. 2000), AVID (Bray et al. 2003), and BLASTZ (Schwartz et al. 2003) could not align more than two DNA sequences. At the same time multiple alignment tools, such as CLUSTAL-W (Thompson et al. 1994) and DIALIGN (Morgenstern 2000, Morgenstern et al. 1998), could not handle more than a few kilobases of sequence. To address the need for multiple (three or more sequences) alignment of long genomic regions several tools have been developed, including LAGAN (Brudno et al. 2003a), MAVID (Bray and Pachter 2004), and TBA (Blanchette et al. 2004). Most recently several methods have been developed for probabilistic alignment of DNA sequences (Lunter, et al 2008, Paten et al 2008). These tools differ from previous approaches in that they can learn correct alignment parameters directly from the data, and use a probability-based score, instead of the heuristic Needleman-Wunsch penalties used by previous methods. All of these tools use a progressive alignment technique, which is based on the phylogenetic relationship between the sequences being aligned. First, the closest sequences are aligned to each other, and then the resulting alignment is aligned to the more distant sequences, following a phylogenetic tree. The progressive heuristic, because it closely mirrors the evolution of the organisms, has been found to be highly effective for alignment of both DNA (Blanchette et al. 2004; Bray and Pachter 2004; Brudno et al. 2003a; Paten et al 2008) and protein (Do et al. 2005; Thompson et al. 1994) sequences. In fact, it was shown that multiple DNA sequence alignment methods (as opposed to pairwise) are better at capturing functional signals from phylogenetically diverse vertebrates due to the use of intermediate sequences in multiple alignments (Margulies et al. 2006).

The problem of aligning whole genomes is more difficult than that of aligning individual, shorter, DNA segments because it is necessary to find the corresponding (orthologous) blocks in the genomes prior to the actual alignment. Perhaps the most straightforward approach to aligning two whole genomes is to perform local alignment between all of the chromosomes of both of the genomes. However classical local alignment methods do not consider whether a particular local alignment falls into a larger syntenic block (region without rearrangements). This leads to difficulties with unmasked repeats, and with paralogous copies of various genomic features: for example when both sequences have n paralogous genes, the classic local alignment methods would yield n^2 alignments between all pairs of these. Despite some of the disadvantages, local alignments were used for comparison of the human and mouse genomes (Schwartz et al. 2003) (Ma et al. 2002), and for the human/mouse/rat three way alignments, (Blanchette et al. 2004), because of their high sensitivity when aligning large mammalian genomes with complex rearrangements.

An alternate approach proposed for human/mouse comparison was the tandem local-global approach (Couronne et al. 2003). In this technique, one genome is split up into arbitrary sized pieces (the authors used 250k), and the potential orthologs for each contig are found in the second genome using a rapid, though less sensitive, alignment program, e.g. BLAT (Kent 2002). The sequence was extended around the

BLAT anchors, and aligned using a global alignment program. This procedure was later expanded to three way alignment of the human, mouse and rat genomes (Brudno et al. 2004). Although the tandem approach produces a map that is accurate within large syntenic blocks (regions of genomes without rearrangements), it has two main weaknesses: small syntenic blocks, resulting from rearrangements within a larger region, may be missed, and the initial arbitrary division of one genome into segments can split a syntenic region, making it difficult to map the region to its true ortholog.

Because of the shortcomings of these methods there has been increased effort in developing hybrid, "glocal" alignment methods. These methods attempt to combine the advantages of the local and global approaches by modeling the rearrangements (shuffling) that a genome undergoes during evolution. Some of the most common rearrangement events are inversions (a block of DNA changes direction, but not location in the genome), translocations (a piece of DNA moves to a new location in the genome), and duplications (two copies of a block of DNA appear where there was one previously). The more recent algorithms for whole genome alignments attempt to incorporate the likely evolutionary events as "operations" into their scoring schemes, including several tools that decide whether to accept or reject a local alignment based on other alignments near it. These include Shuffle-LAGAN (Brudno et al. 2003b), Chains and Nets on the UCSC Browser (Kent et al. 2003), Mercator (Dewey 2007), A-bruijn Block Aligner (Raphael et al 2004), and Mauve (Darling et al. 2004).

While most of the pairwise whole genome alignment algorithms described above have been generalized to multiple alignment, these approaches rely on a *reference* genome, against which all of the other sequences are laid out, or require a one-to-one mapping, where each nucleotide of one genome is constrained to align to at most one place in the other genome. Both of these approaches have drawbacks for whole genome comparisons: the first will not align segments conserved among some genomes, but missing in the reference, while the second will fail to align any element that has undergone a duplication. Most recently non-referenced genome alignment implementations have appeared, for example the ENREDO package (Paten, et al 2008), used by the ENSEMBL genome browser. ENREDO builds a genome alignment graph, akin to the A-Bruijn graph alignment of (Raphael et al 2004), and all of the genomes are aligned simultaneously. This approach has the disadvantage of not taking into account the phylogenetic information about the species, making it more difficult to align distant genomes.

In this work, we present a novel non-referenced multiple alignment algorithm. Our approach is based on the progressive technique for multiple alignment, and has several advantages over previous algorithms: 1) it does not utilize a reference genome, but creates a symmetric alignment equally valid for all genomes; 2) it allows for arbitrary duplications in all genomes, and does not require the nucleotides to have a one-to-one mapping; and 3) it is able to align short syntenic blocks based on their adjacency to high similarity areas, even in the presence of rearrangements. Our results demonstrate that our alignments have high exon alignment accuracy, and outperform other approaches, especially for alignment of genes from multi-gene families and distant species.

RESULTS

ALGORITHMS

Our algorithm is based on progressive alignment, with genomes aligned up the phylogenetic tree. After aligning two genomes, our algorithm joins together syntenic blocks based on the outgroups (those sequences that will be aligned at a later stage: for example if we have aligned mouse with rat, then human, dog, and chicken are all outgroups). By picking an order of the syntenic blocks which is closest to the outgroups we facilitate alignment of the more distant genomes.

In the sections below we start by describing SuperMap – a symmetric extension of the pairwise Shuffle-LAGAN algorithm capable of alignment of whole genomes. Secondly we describe a novel multiple whole genome alignment algorithm that uses SuperMap for pairwise genome alignment, and uses an algorithm based on the Maximum Weight Perfect Matching Problem to order the aligned areas of the two genomes to simplify the mapping in the next stages of the progressive algorithm.

SuperMap: Pairwise Alignment of Genomes

The SuperMap algorithm is based on the original Shuffle-LAGAN (S-LAGAN) chaining algorithm (Brudno et al. 2003b). The S-LAGAN alignment algorithm runs in three stages (Figure 1). During the first, all local alignments between the two input sequences are located. In the second stage, we select a subset of these alignments to represent a rearrangement map between the two sequences. Finally regions of conserved synteny (those without rearrangements) are re-aligned using the LAGAN global alignment algorithm.

The S-LAGAN chaining program takes as input a set of local alignments between the two sequences and returns the maximal scoring subset of these under certain gap criteria. In order to allow S-LAGAN to catch rearrangements, the collinearity assumption of global algorithms was relaxed to allow the map to be non-decreasing (monotonic) in only one sequence (the "base"), without putting any restrictions on the second sequence. This is called a 1-monotonic conservation map. Perhaps the main weakness of the Shuffle-LAGAN chaining algorithm is its asymmetry, since it depends on one genome being labeled as the "base", and duplications only in the base genome are aligned.

To address this issue we have built the SuperMap algorithm that solves the symmetry problem by adding a post-processing step. We run S-LAGAN twice, using each sequence as the base (see Figure 2). This gives us three pieces of data: the original local alignments, which were common to the two runs of S-LAGAN, and two chains of these alignments, each corresponding to the S-LAGAN 1-monotonic maps. We then classify all local alignments as belonging to both chains, and consequently orthologous (best bidirectional hits) or being in only one chain, and hence belonging to a duplication. Local alignments that do not fall into either chain are considered to be false positives, and are removed from consideration. We transform the two S-LAGAN chains into a graph as follows: Every alignment becomes a node. If the alignment A2 follows A1 on an S-LAGAN chain, we add an edge going from A1 to A2. Every node that has incoming edges from two different nodes is the beginning of a syntenic block, and every node with two outgoing edges to two different nodes is the end of such a region. Identifying all regions can easily be accomplished in linear time once the S-LAGAN chains are built.

This SuperMap algorithm has several advantages over regular S-LAGAN: 1. It is able to locate duplications in both sequences, overcoming a major weakness of the original algorithm; 2. In case of translocations, two of the pieces are no longer arbitrarily joined together; 3. This approach locates both regions of one-to-one similarity (those that were in both 1-monotonic chains) and likely duplications.

Multiple Alignment

We have generalized the SuperMap algorithm to alignment of more than two genomes through a progressive alignment framework. Our algorithm reorders, at each internal node of the phylogenetic tree, the alignments between its children genomes in order to simplify the alignment of these alignments to the next outgroup. We refer to this ordering as the "ancestral" ordering, as it most closely resembles the order of the same regions in the genomes of other, close genomes.

For every node of the tree, our algorithm starts by generating a set of local alignments between the two children genomes. SuperMap chaining is used to identify all rearrangements and define consistent subsegments among the local alignments. The resulting regions are aligned with LAGAN. Given the output of the SuperMap algorithm, for every syntenic block, we consider the two children genomes as the two possible next blocks in the best ordering of the alignments. To decide on the better ordering we use the most proximal outgroups to compute the support for each edge, and then select a subset of these edges such that each syntenic region is preceded by at most one region, and followed by at most one region.

In order to build this ancestral ordering, we first use Fitch's algorithm to build a consensus representation of all alignments. Fitch's algorithm recreates the character that should be used in the ancestral genome so as to minimize the number of mutations that take place in the alignment. We align these ancestral contigs to the most proximal outgroups (since we assume that the tree is binary, we follow one edge up the tree, and locate those genomes that are present in the other child of this node). For every breakpoint between syntenic blocks, we determine which of the two children is most likely to be the ancestral order by letting the outgroups "vote" on the proper ordering. Each outgroup is assigned a weight based on its proximity to the ancestral node. The outgroup's vote is distributed between the two children, with the child whose order of conserved elements is closest to the outgroup receiving the bigger fraction.

This problem can be formally written as the *Maximum Weight Path Cover* problem, in a similar manner to the reduction of the breakpoint median problem to the Traveling Salesman Problem (Sankoff and Blanchette 1998). Each path corresponds to an ordered segment of the ancestral genome. However this problem is known to be computationally intractable (NP-hard). Consequently, we solve the *Maximum Weight Perfect Matching* (MWPM) problem instead. We reduce the alignment problem to a graph in the same way as in the SuperMap algorithm (see Figure 3), though the new graph is built based on the syntenic regions that are produced by SuperMap. We define the weights for each edge in the graph based on how much it is supported by outlying genomic sequences. This procedure is explained in detail in the Methods section. The MWPM solution is a set of paths and cycles. We remove the smallest weight edge in each such cycle to break the circular path, and create a (possibly non-optimal) path cover. For each path we build an ancestral "contig" by filling the gaps between alignments with the genomic sequence that was closer to the ancestor. We use these ancestral contigs in higher levels of the tree.

It is important to note that our "ancestral genome order" and "ancestral contigs" should not be thought of as representing the genome of the ancestor of the organisms being aligned – in fact it is an ordering of the pieces that will make it easiest to align them to the next outgroup. This idea also appears in the context of progressive alignments of protein sequences, where alignment programs use the UPGMA guide tree to align the sequences, rather than the neighbor joining tree, even though the neighbor joining tree is a better approximation of the true phylogeny (Nelesen et al. 2008) (Edgar 2004).

EVALUATION

Our multiple genome alignment algorithm has been implemented as part of the VISTA Genome Pipeline (VGP), and has been used to align seven vertebrate genomes (human, rhesus, dog, horse, mouse, rat, chicken), six *Drosophila* genomes (*D. melanogaster, D. ananassae, D. erecta, D. pseudoobscura, D. simulans, and D. yakuba*). To evaluate the quality of our alignments we considered two metrics commonly used in alignment literature: the overall coverage of the genome and of important genomic features by high scoring alignments (Waterston et al 2002, Schwartz et al 2003), and the accuracy of alignment of annotated exons (Brudno et al 2003a, Bray and Pachter 2004). Another metric commonly used to evaluate alignments is the comparison of sequences that have undergone simulated evolution, and for which the true alignment is known. While this approach is useful for comparison of the alignments of regions without rearrangements (Blanchette et al 2004), where the only allowed evolutionary events are substitutions and insertions/deletion, it is not currently practical for whole genome alignment, as currently there are no tools for realistic simulation of evolution of a complete genome.

Genome Coverage

The first analysis we conducted was the comparison of the three-way human-mouse-rat alignment obtained using our progressive whole genome algorithm with the tandem local/global heuristic previously used by the VISTA Genome Pipeline (Brudno et al 2004, Couronne et al 2003). We evaluated the alignments based on the fraction of the gene coding regions and of the whole genome that are aligned above a certain threshold (coverage), and based on the total size of the alignments (specificity). The results (summarized in Table 1) show higher sensitivity and accuracy of the new method in aligning coding regions, while the overall length of the alignment was lower, indicating higher specificity. The increase in exon coverage is due to the fact that the new method is better able to align genes in regions with rearrangements. To illustrate this we demonstrate coverage statistics for chromosome 20, which has almost no rearrangements between the species, and the results of the two methods are very similar.

Exon Alignment Accuracy

Secondly, we compared the overall alignment accuracy of our progressive technique with the alignments produced by the Penn State/UCSC Alignment Pipeline and displayed by the UCSC Genome Browser for two clades: vertebrates and Drosophilas. We also compared our vertebrate alignments to the ENREDO/PECAN alignments displayed at the ENSEMBL Genome Browser. To measure the alignment quality we use the method that evaluates exon alignment (Brudno et al. 2003a, Bray & Pachter 2004). For both clades we have designated a reference organism (human and D. melanogaster respectively). We decompose the multiple alignments into pairwise alignments between the reference and all other species, and rank each exon of the non-reference genomes based on what percentage of its nucleotides are aligned within an exon in the reference genome. The results are summarized in Figure 4. For the mammalian genomes (4A&C) our alignment method consistently achieved exon alignment accuracies of above 90%, with the highest accuracy being for dog (94%). The difference between our alignments and those of the UCSC browser were small – we aligned anywhere between 1.6% (rhesus) and 4.8% (horse) more exons completely within a human exon than the UCSC pipeline, with a similar decrease in the number of exons not aligned at all (first column). However the differences grew when we considered more distant genomes: we were able to align 85% of the annotated chicken exons over their full length to human exons, while the UCSC pipeline aligned 15% less. We found the differences in exon alignment accuracy between the ENSEMBL and our alignments even greater (Figure 4D). As became evident from our analysis of multi-gene families (see below), these differences were mainly due to the inability of the ENSEMBL pipeline to properly deal with some duplication events (see below).

While the overall level of alignment accuracy in the Drosophila genomes was much lower (Figure 4E, from 83 to 60% of the exons aligned), the overall tendency of our alignment pipeline to perform better than the UCSC browser for alignment of more distant sequences is still evident (Figure 4F).

Pairwise versus Multiple Alignment

We wanted to test whether the ancestral multiple alignment method improves results compared with the pairwise one (using the SuperMap anchoring algorithm). Although the algorithm is generally the same, the use of intermediate sequences has been previously shown to improve the alignment of distant orthologs. For example, Margulies et al (Margulies et al. 2006) showed that multiple DNA sequence alignment methods (as opposed to pair-wise) are substantially better at aligning (or ‘capturing’) functional signals from phylogenetically diverse vertebrates. To test this hypothesis we compared the exon accuracy of pairwise alignments between the genomes present in our whole-genome alignment set with the multiple alignment results. The results, summarized in Figure 4B, confirm that using intermediate sequences improves alignment quality, especially when aligning more distant sequences, such as chicken.

Alignment of Inparanoid Gene Families

In order to test not only the sensitivity, but also the specificity of our method, we have compared the multiple alignments to the Inparanoid gene clusters for human and mouse genomes (O'Brien et al 2005). Inparanoid organizes human and mouse genes into groups, each containing one or more genes from each genome. All of the human and mouse genes within a group (cluster) are orthologues of each other, and putatatively evolve from a single gene in the genome of the human/mouse ancestor. The Inparanoid clusters are based on pairwise protein BLAST alignments between all of the genes; since this method is significantly different from whole genome multiple alignments, it provides an independent method for evaluating the accuracy of the alignments. Good genomic multiple alignments should align truly orthologous, rather than paralogous genes. We considered all of the mouse exons, and evaluated their alignments to human exons, labelling every alignment orthologous (aligned within a gene that is an Inparanoid ortholog) or paralogous (to an exon that is not an ortholog). For genes, we considered them ortholgous/paralogous if any exon in them was ortholgous/pralogous. As is illustrated in Table 2, the alignments generated by our method was the most sensitive (highest fraction of orthologous genes/exons aligned), while the UCSC-based alignments and those from ENSEMBL were more specific (fewer non-orthologous genes/exons).

Secondly, we evaluated the three methods on how well they can align genes that have undergone recent (since the divergence of human and mouse) duplications. To test this we considered only those genes and exons that were in clusters with multiple human and multiple mouse genes (many-many) and those with multiple human genes and a single mouse gene (one-many). For both of these groups, all of the alignment methods were able to align (even to a single ortholog) significantly fewer genes than in the genome as a whole. This trend was especially pronounced for ENSEMBL, that aligned only 20% of the many-many genes, and 44% of the one-many genes to even a single ortholog. Our alignments were the most sensitive for these clusters, aligning 70% and 79% of the genes, respectively. Furthermore, our alignments were the only ones that were able to align more then 3% of either the genes or the exons to *all* of the orthologs: only 46 of the 2500 exons in multi-gene clusters were aligned to all of the orthologs in ENSEMBL alignments (36 in the UCSC alignments), while 655 were in our alignments.

DISCUSSION

In this paper we describe the designed and implementation of a progressive alignment algorithm for whole genomes. Our method differs from other multiple alignment algorithms for whole genomes in that it does not assume a reference genome against which all of the other genomes are laid out. Instead we combine the "glocal" alignment framework that is widely used in whole genome alignment with a progressive approach, where at every progressive step we attempt to order the obtained alignments in such a way as to ease the comparison to the next outgroup. Thus, our approach takes advantage of highly conserved segments to align nearby less conserved ones, even in the cases where there has been a rearrangement at the locus in one of the species. We have implemented our method as part of the VISTA Genome Pipeline, and applied it to the alignment of 7 vertebrate and 6 fly genomes. We compared the resulting alignments to those available through the UCSC Genome Browser and ENSEMBL, and show that our approach is more accurate at aligning exons between the species, especially as the evolutionary distance between the organisms grows. All multiple alignments generated by our algorithm are available for browsing and analysis through the VISTA Browser at http://genome.lbl.gov/vista/index.shtml.

At the same time our approach to whole genome alignment has several weaknesses, which may prove to be fruitful grounds for future work. Our approach toward the reconstruction of “ancestral” sequences for further alignment in the progressive framework does not attempt to reconstruct the true genome of the ancestor species, but rather to construct the sequence that is easiest to align to the outgroup. A method that will attempt to reconstruct the true ancestor may be preferable for a wide range of evolutionary studies (Ma et al. 2006). Another potential area of improvement is our treatment of poorly assembled, draft genomes. For such genomes, our algorithm currently resorts to using the reference-based alignment approach, as using a draft genome in our ancestor reconstruction stage has lead to decreased alignment accuracy. Designing a progressive non-referenced framework for aligning both finished and draft genomes is an important future goal, as many genomes sequenced today are left in draft form.

Perhaps the most evident weakness of ours (and all of the other existing whole genome alignment algorithms) is their inability to deal with multi-gene families. While our alignment method was the most sensitive in capturing multi-gene inparanoid gene clusters (O’Brien et al 2005), aligning 70% of the genes and 64% of the exons in inparanoid clusters with multiple genes from both humans and mice to an ortholog, only 21% and 15% of these were aligned to *all* of the orthologs (see Table 2). This shows that there is still significant room to improve upon our methods for whole genome alignments.

METHODS

The sections below provide a more thorough description of the various components of our alignment pipeline. The Shuffle-LAGAN chaining algorithm and the original Multi-LAGAN alignment algorithms have been described earlier (Brudno et al. 2003a, 2003b).

Implementation & Availability

The whole genome pipeline algorithm has been implemented in a combination of Perl and C programs, using a MySQL relational database to store both input genomic sequences and generated alignments. All major stages of the pipeline - obtaining local hits with BLAT, SuperMap chaining, aligning syntenic regions with LAGAN, and computing ancestral contigs - make use of a Linux cluster. The pipeline software is publicly available at http://genome.lbl.gov/vista/downloads.shtml.

Local Alignments

The local alignments between all sequences can be computed using any alignment algorithm. We typically use BLAT, as it allows for rapid alignment. We run it in translated DNA mode, indexing non-overlapping 5-amino acid words, and requiring one word to trigger an alignment.

Global Alignments

Global alignments are done with PROLAGAN, which is a variation of the original Multi-LAGAN program that allows for the alignment of two alignment (profiles). The alignment of two profiles is a basic step in the Multi-LAGAN algorithm, and the PROLAGAN executable separates this functionality into a standalone program. The algorithm used is identical to the progressive step of the original LAGAN algorithm (Brudno et al 2003a), and is available as part of the LAGAN toolkit starting with version 2.0.

SuperMap

The SuperMap algorithm is implemented as a standalone Perl application, and is available as part of he LAGAN Toolkit. After running the S-LAGAN algorithm with both genomes as bases, the local hits that form both of the chains are sorted by their positions in the first genome. The two lists are traversed to identify local alignments that are in both chains, which are refered to as dual-monotonic (DM), and those which are in only one of the chains (labeled M1 and M2, depending on the chain). In this first pass we also group alignments which are labeled DM and M1 into segments of conserved synteny by unifying any alignment with the previous one if they are consistent (can be a part of the same global alignment) and have the same type (both M1 or both DM). The local alignments are then re-sorted based on the second genome, and the segments of the type M2 are formed.

This algorithm keeps all of the local alignments on disk, sorted using the Unix sort command. We use only a constant amount of memory, thus allowing for processing of extremely large sets of local alignments efficiently.

Extending the Alignments

One of the major weaknesses of fast, heuristic local alignment algorithms is that they often fail to discover weaker areas of similarity, and the borders of syntenic blocks based on these alignments may fail to include important conserved regions nearby because they failed to meet the local alignment criteria. Consequently, the Shuffle-LAGAN algorithm expanded the borders of every syntenic block to the subsequent syntenic block in the base sequence, or up to a constant, whichever was smaller. Expansion in the second sequence was based on a fixed multiplicative factor of the expansion in the first sequence. In SuperMap, we augment this approach by expanding each alignment to the nearest M1 or DM alignment in sequence 1, and either M2 or DM alignment in sequence 2. This approach limits the expansion of alignments to a minimum, while allowing for the addition of the border regions not included in the original set of local alignments.

Computing Ancestral Contigs

After an alignment between two segments is built, we compute the ancestral contigs as follows:

(1) Infer an ancestral sequence for all of the alignments using Fitch's algorithm (Fitch 1971). Gaps are treated as a fifth character.

(2) Build local alignment between the ancestral sequence and the genomes in the nearest outgroup (the nearest outgroup can have either one or two genomes).

(3) Convert every alignment to an edge that connects the two nodes corresponding to its two endpoints. We will refer to such edges as "alignment edges". Connect two endpoints if there is no third alignment that falls between them in the genome. This type of edge is referred to as a "connection edge". See Figure 3 (A&B) for an illustration.

(4) Compute the weight for every connection edge by running the S-LAGAN chaining algorithm on all of the local alignments built from every alignment edge, and also on pairs of alignments connected by a connection edge. Let *a* and *b* be two syntenic blocks joined by a connection edge. The weight for this edge is computed as follows: for both of the outgroup genomes X^1, X^2 we find all of the alignments between X^i and both *a* and *b*. We find the highest scoring consistent chain of local alignments between X^i and *a*, X^i and *b* and between X^i and ($a \cup b$). Let the cumulative scores of these three chains be called C_1, C_2 and U, respectively. Then we set $W^iab = (U\text{-}MIN(C_1,C_2))/ MAX(C_1,C_2)$. Note that W^iab ranges between 1 and 0, and is the support for the edge $E_L(a,b)$ from sequence X^i. We combine the supports to get the weight for the edge between a and b to be $Wab = \sum W^iab/n$. This is illustrated in Figure 3E.

(5) Remove the alignment edges from the graph and compute the maximum weight matching on the resulting graph. Remove the smallest edge from every cycle. For efficiency we split the graph into the connected components and perform the procedure on all connected components separately. The result of the maximum weight matching algorithm is shown in Figure 3C.

(6) Any edge in the matching joins together two alignments through a particular genome. Build an ancestral contig by resolving any overlap between the alignment if they overlap in the genome through which they were joined, or by inserting any in-between piece in the joining genome if they do not overlap. This is illustrated in Figure 3D.

Handling Low Quality Assemblies

When aligning a genome consisting of many short contigs to a high quality assembly, which usually consists of chromosomes, we modify our algorithm by replacing the ancestral genome ordering stage with one that orders all of the alignments based on their order in the better genome. This is done because a low quality genome assembly is likely to have regions that appear as duplications, but are in reality under-collapsed copies of the same genomic region. The copies are handled as duplications, and lead to inaccuracies in the ancestral reconstruction step. Instead, in such cases we create a "faux-ancestor" by ordering all of the M1 and DM alignments based on their order in the high quality genome.

Evaluation Based on Inparanoid Clusters

We have downloaded the database of human/mouse Inparanoid orthologous gene clusters (O'Brien et al 2005) from http://inparanoid.sbc.su.se and found the location of the orthologs in our genome assemblies using the tables at the UCSC Genome Browser. Inparanoid builds clusters of ortholgous genes based on their pairwise BLASTP scores. We removed from consideration all overlapping genes, as well as clusters where any of the genes had missing locations. The remaining set consisted of 13,780 genes with 141,244 exons. We counted two exons aligned if they overlapped by a single nucleotide in the multiple alignment. Two exons were considered orthologous if they were located on two genes that were member of a single Inparanoid cluster. The one-many and many-many clusters were those that had multiple genes from human and both human and mouse genomes, respectively.

ACKNOWLEDGMENTS

We would like to thank Rotem Sorek, Adrian Dalca and Nilgun Donmez for critical readings of this manuscript. We are also grateful to the anonymous reviewers for their feedback. Funding was provided by the US NIH grant R01-GM81080 and NSERC Discovery Grant 327669. This work was also supported by the Director, Office of Science, Office of Basic Energy Sciences, of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231

Figure & Table Legends

Figure 1. Overview of the Shuffle-LAGAN algorithm. S-LAGAN first locates all local areas of similarity between the two sequences using a local alignment algorithm. A subset of these is selected using the 1-monotonic chaining algorithm (Figure 2). Finally global alignments are built (using LAGAN) for consistent subsegments of the 1-monotonic chain (areas without rearrangements). The S-LAGAN algorithm is not symmetric, requiring two alignments to identify all duplications.

Figure 2: SuperMap Algorithm. The left side **(I)** is a dotplot demonstrating the local alignments between two hypothetical genomes. Local alignments A & B correspond to duplications in Organism 1 and Organism 2, respectively. Local alignment C corresponds to an inversion, and local alignments D are spurious false-positives. The Middle panel **(II)** shows (in blue) the result of running the regular S-LAGAN 1-monotonic chaining algorithm using Organism 1 as the base. On the right **(III)** we have built the 1-monotonic maps for Organism 1 (blue) and 2 (red). Whenever these chains merge, they are shown as purple. Similarly, local alignments are colored based on which chains they belong to: blue (M1), red (M2) or purple (both,DM). All points where the two chains split or join are borders of a region of conserved synteny.

Figure 3: A schematic representation of the reconstruction of ancestral orderings. **(A)** shows the result of running supermap on a set of local alignments. **(B)** shows the corresponding graph representation, with alignment edges colored black, and connection edges colored by the color of the genome in which these syntenic blocks are adjacent. The weight of all of the edges is computed as shown in part **(E)**. **(C)** is the output of running the maximum matching algorithm: each node is connected to only one connection edge, as well as the alignment edge. Note that by removing the alignment edges this graph is decomposed into two connected components, that can be solved separately. **(D)** shows the translation of the maximum matching output back to the alignments: the result of the algorithm is a chain of alignments, where the letters of the appropriate genome can be inserted between the sequences. These chains can then be used for alignment in higher nodes of the tree. **(E)** In this example we are recreating the ancestral order of the grey node in the phylogeny on the right. The top-right quadrant shows the output of the SuperMap algorithm applied to the blue & purple genomes. The top-left and bottom right quadrants show the local hits of the two genomes on the red outgroup. The selected regions on the left are used to compute the score for the blue edge marked S $(S = (U-MIN(C_1,C_2))/ MAX(C_1,C_2))$. All of the other edges will be scored the same way, and the MWPM problem is solved in the resulting graph. In this particular case the purple genome will have more support for being the ancestral order than the blue genome.

Figure 4: Exon alignment accuracy for vertebrate (A-D) and Drosophila (E&F) genomes. Each category on the X axis shows the exons for a particular species that are aligned to a reference genome exon over the given fraction of their length. The Y axis for plots **A&E** shows the overall fraction of exons in each category for our alignments, while the other plots show the difference of these fractions between our multiple alignments and those from the UCSC Genome Browser (ours minus UCSC, **C&F**), those from the ENSEMBL browser **(D)**, and our pairwise alignments **(B)**. Our algorithms aligns more exons perfectly (100% category) and fewer exons are not aligned at all (0—10 category) for all species. In the comparison between our multiple and our pairwise alignments, while the macaque alignments are identical, and the dog alignments are nearly identical (the two species are close) the human/mouse alignment is slightly improved, and nearly 10% of chicken exons were aligned in the multiple but not pairwise alignment. The 23-way ENSEMBL alignments that we used did had a different version of the Horse genome, preventing a direct comparison, and we did not generate a pairwise human/rat alignment (rat would be very similar to mouse), hence the missing columns in plots **B** and **D**.

Table 1: A comparison of the alignment quality for human Chromosome 20 and whole Human Genome to the mouse genome between the tandem local/global heuristic previously used in the VISTA Genome Pipeline (Couronne et al. 2003) (Brudno et al. 2004) and the new Ancestral Alignment technique. The numbers are the coverage (Schwartz et al. 2003) of the whole genome (Total) and the annotated coding exons of RefSeq genes (Exon). Size is the total size of the resulting alignments, and Time is the wall clock time for the alignment (20 dual node, 40 CPU cluster). This time excludes the running time for running pairwise local alignment (BLAT), which is approximately 3 days per pair of genomes.

Table 2: A comparison of the alignments at the UCSC Genome Browser, ENSEMBL and our alignments (VISTA) based on Inparanoid gene clusters. We considered two exons aligned if they overlapped even by a single nucleotide (see Methods). The results show that while the the VISTA Browser alignments have slightly higher sensitivity (1.8% on genes and 0.7% on exons), it also has a slightly higher rate of alignment to paralogs (3.2% on genes, 0.8% on exons). The bulk of this was due to genes that we aligned to both the true orthologs and to paralogs, with genes/exons aligned only to paralogs were less the 0.5% of the total. Simultaneously our methods showed significantly higher sensitivity at aligning genes in multi-gene clusters: ~10% higher for exons aligned to any ortholog, and 20-30% higher for genes aligned to all orthologs.

Figure 1

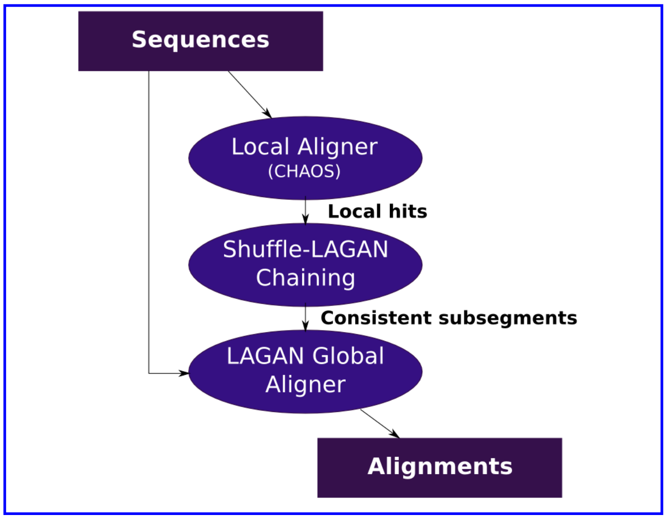
Sequences
Local Aligner
(CHAOS)
Local hits
Shuffle-LAGAN
Chaining
Consistent subsegments
LAGAN Global
Aligner
Alignments

Figure 2

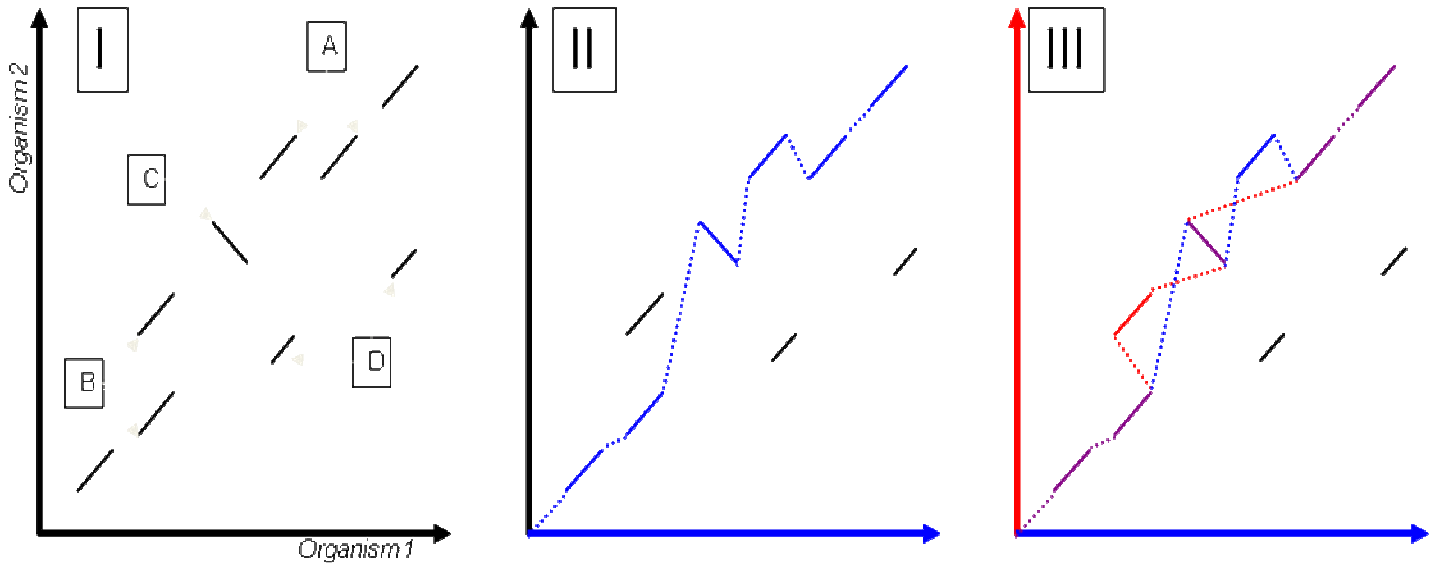

Figure 3

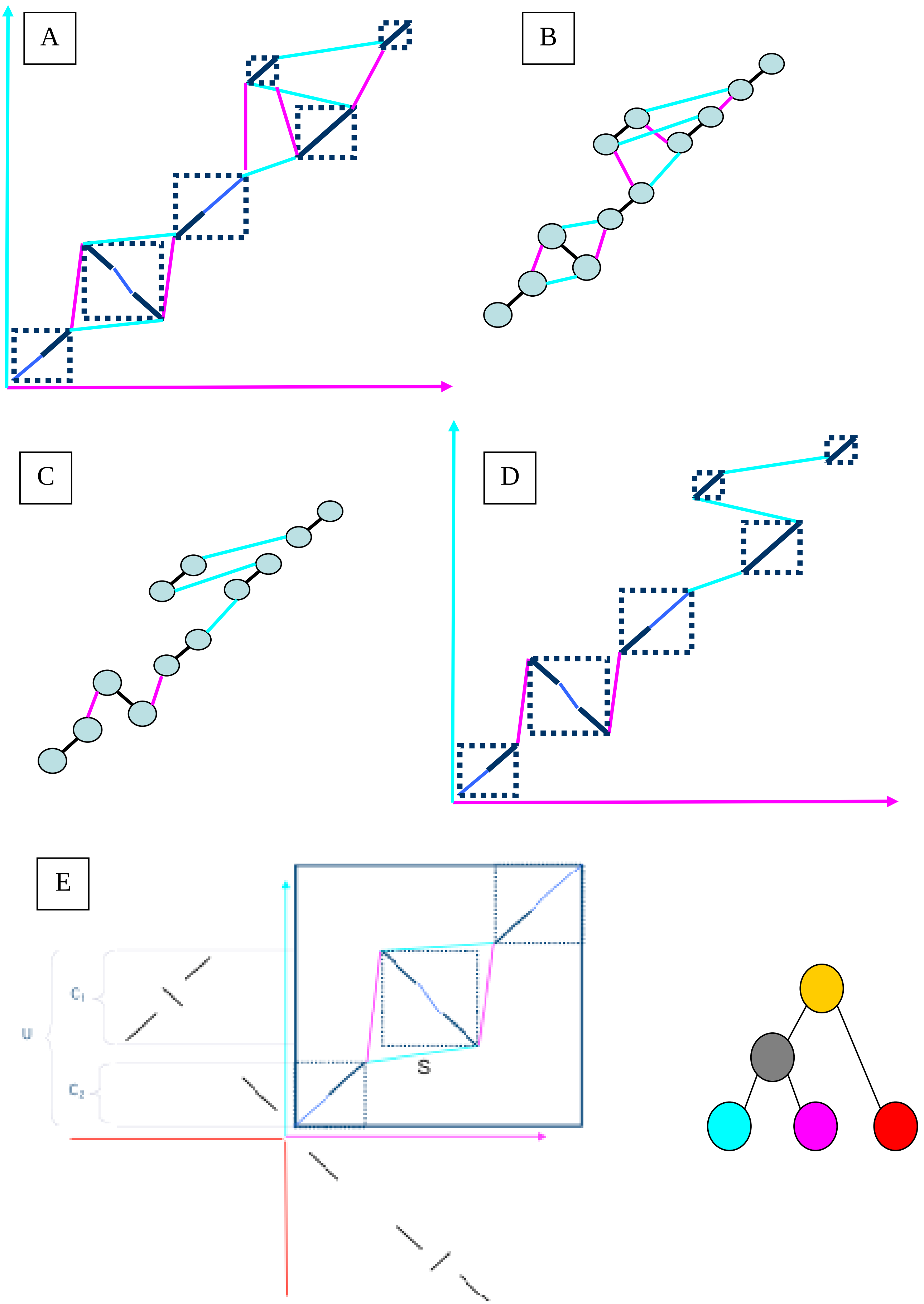

Figure 4

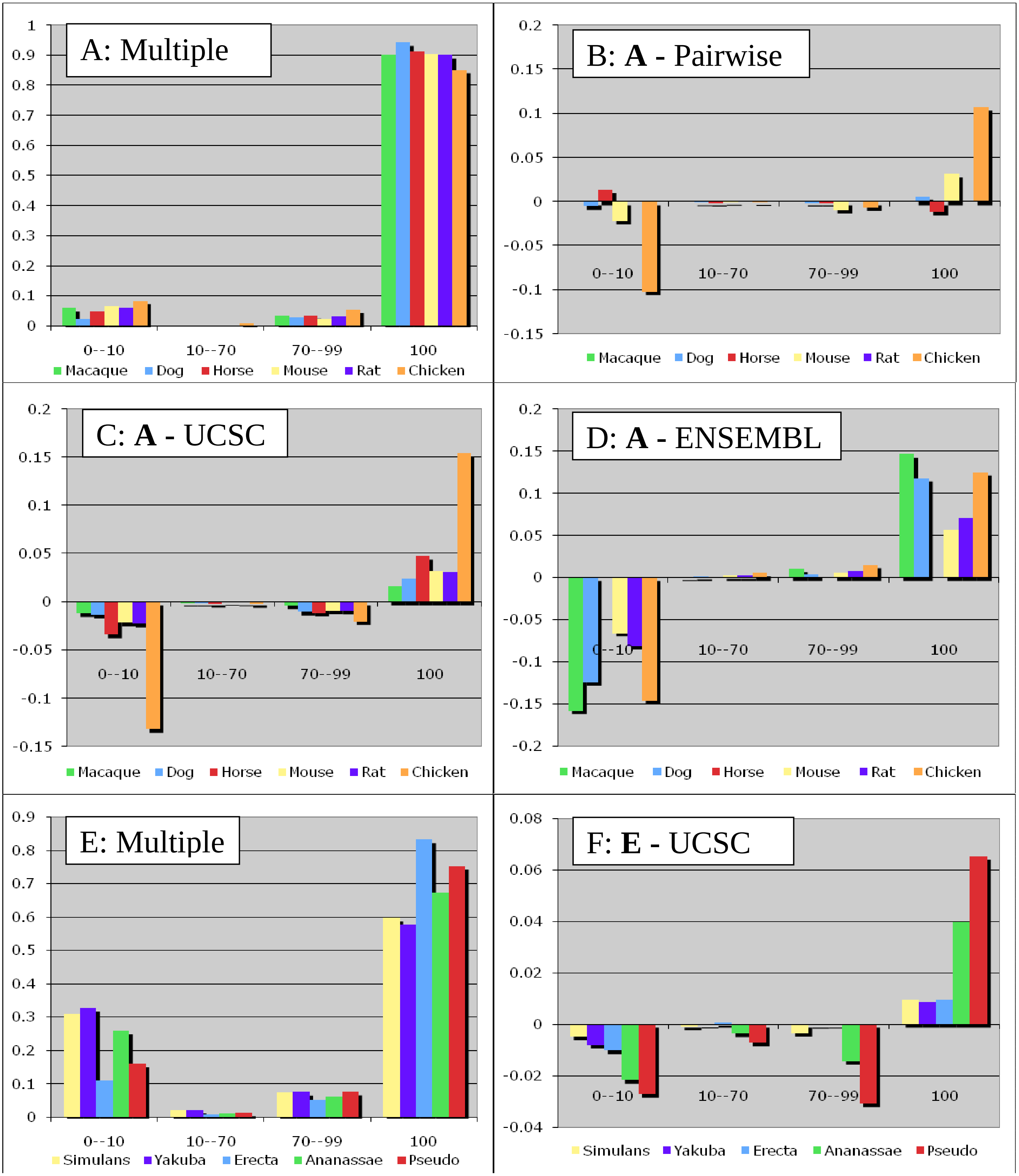
A: Multiple
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0--10
10--70
70--99
100
Macaque
Dog
Horse
Mouse
Rat
Chicken
B: A - Pairwise
0.2
0.15
0.1
0.05
0
-0.05
-0.1
-0.15
0--10
10--70
70--99
100
Macaque
Dog
Horse
Mouse
Rat
Chicken
C: A - UCSC
0.2
0.15
0.1
0.05
0
-0.05
-0.1
-0.15
0--10
10--70
70--99
100
Macaque
Dog
Horse
Mouse
Rat
Chicken
D: A - ENSEMBL
0.2
0.15
0.1
0.05
0
-0.05
-0.1
-0.15
-0.2
0--10
10--70
70--99
100
Macaque
Dog
Horse
Mouse
Rat
Chicken
E: Multiple
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0--10
10--70
70--99
100
Simulans
Yakuba
Erecta
Ananassae
Pseudo
F: E - UCSC
0.08
0.06
0.04
0.02
0
-0.02
-0.04
0--10
10--70
70--99
100
Simulans
Yakuba
Erecta
Ananassae
Pseudo

Table 1

	Chromosome 20		**Whole Genome**	
	Tandem	*Ancestral*	*Tandem*	*Ancestral*
Total	33.5 %	32.9 %	28.7 %	28.4 %
Exon	90.6 %	89.3 %	78.5 %	83.5 %
Size			13.7 Gb	11.0 Gb
Time			**15**	**10**

Table 2

	VISTA		UCSC/MultiZ		ENSEMBL/ENREDO	
	Genes	Exons	Genes	Exons	Genes	Exons
Aligned to gene/exon (of 13780 genes, 141244 exons)	**13444** **97.6%**	**134446** **95.2%**	13207 95.8%	133498 94.5%	11592 84.1%	113971 80.7%
Of these, aligned to orthologs only	12978 94.2%	133264 94.4%	**13170** **95.6%**	**133363** **94.4%**	11567 83.9%	113897 80.6%
Of these, aligned to orthologs and paralogs	**417** **3.0%**	**943** **0.7%**	19 0.1%	7 0%	11 0.1%	2 0%
Of these, aligned to paralogs only	49 0.4%	239 0.2%	18 0.1%	128 0.1%	**14** **0.1%**	**72** **0.1%**
Aligned to any ortholog, Many-many clusters (of 182 genes, 862 exons)	**128** **70.3%**	**549** **63.7%**	112 61.5%	475 55.1%	38 20.9%	162 18.8%
Of these, aligned to all orthologs	**39** **21.4%**	**126** **14.6%**	4 2.2%	2 0.2%	1 0.5%	1 0.1%
Aligned to any ortholog, One-many clusters (of 305 genes, 2500 exons)	**242** **79.3%**	**2131** **85.2%**	226 74.1%	1909 76.4%	133 43.6%	1153 46.1%
Of these, aligned to all orthologs	**97** **31.8%**	**655** **26.2%**	7 2.3%	36 1.4%	7 2.3%	46 1.8%

References

Abbasi, A.A., Z. Paparidis, S. Malik, D.K. Goode, H. Callaway, G. Elgar, and K.H. Grzeschik. 2007. Human GLI3 intragenic conserved non-coding sequences are tissue-specific enhancers. *PLoS ONE* **2:** e366.

Batzoglou, S., L. Pachter, J.P. Mesirov, B. Berger, and E.S. Lander. 2000. Human and mouse gene structure: comparative analysis and application to exon prediction. *Genome Res* **10:** 950-958.

Bejerano, G., M. Pheasant, I. Makunin, S. Stephen, W.J. Kent, J.S. Mattick, and D. Haussler. 2004. Ultraconserved elements in the human genome. *Science* **304:** 1321-1325.

Blanchette, M., W.J. Kent, C. Riemer, L. Elnitski, A.F. Smit, K.M. Roskin, R. Baertsch, K. Rosenbloom, H. Clawson, E.D. Green, D. Haussler, and W. Miller. 2004. Aligning multiple genomic sequences with the threaded blockset aligner. *Genome Res* **14:** 708-715.

Bray, N., I. Dubchak, and L. Pachter. 2003. AVID: A global alignment program. *Genome Res* **13:** 97-102.

Bray, N. and L. Pachter, 2004. MAVID: Constrained ancestral alignment of multiple sequences, *Genome Res* **14**:693-699

Brudno, M., C.B. Do, G.M. Cooper, M.F. Kim, E. Davydov, E.D. Green, A. Sidow, and S. Batzoglou. 2003a. LAGAN and Multi-LAGAN: efficient tools for large-scale multiple alignment of genomic DNA. *Genome Res* 13: 721-731.

Brudno, M., S. Malde, A. Poliakov, C.B. Do, O. Couronne, I. Dubchak, and S. Batzoglou. 2003b. Glocal alignment: finding rearrangements during alignment. *Bioinformatics* **19 Suppl 1:**i54-62.

Brudno, M., A. Poliakov, A. Salamov, G.M. Cooper, A. Sidow, E.M. Rubin, V. Solovyev, S. Batzoglou, and I. Dubchak. 2004. Automated whole-genome multiple alignment of rat, mouse, and human. *Genome Res* **14:** 685-692.

Couronne, O., A. Poliakov, N. Bray, T. Ishkhanov, D. Ryaboy, E. Rubin, L. Pachter, and I. Dubchak. 2003. Strategies and tools for whole-genome alignments. *Genome Res* **13:** 73-80.

Darling, A.C., B. Mau, F.R. Blattner, and N.T. Perna. 2004. Mauve: multiple alignment of conserved genomic sequence with rearrangements. *Genome Res* **14:** 1394-1403.

de la Calle-Mustienes, E., C.G. Feijoo, M. Manzanares, J.J. Tena, E. Rodriguez-Seguel, A. Letizia, M.L. Allende, and J.L. Gomez-Skarmeta. 2005. A functional survey of the enhancer activity of conserved non-coding sequences from vertebrate Iroquois cluster gene deserts. *Genome Res* **15:** 1061-1072.

Dewey, C., J.Q. Wu, S. Cawley, M. Alexandersson, R. Gibbs, and L. Pachter. 2004. Accurate identification of novel human genes through simultaneous gene prediction in human, mouse, and rat. *Genome Res* **14:** 661-664.

Dewey, C.N. 2007. Aligning Multiple Whole Genomes with Mercator and MAVID. *Methods Mol Biol* **395**: 221-236.

Do, C.B., M.S. Mahabhashyam, M. Brudno, and S. Batzoglou. 2005. ProbCons: Probabilistic consistency-based multiple sequence alignment. *Genome Res* **15:** 330-340.

Edgar, R.C. 2004. MUSCLE: a multiple sequence alignment method with reduced time and space complexity. *BMC Bioinformatics* **5:** 113.

Eppstein, D., Galil, R., Giancarlo, R., and Italiano, G.F. 1992. Sparse dynamic programming I: linear cost functions. J. ACM, **39**: 519-545.

Fitch, W.M. 1971. Toward Defining the Course of Evolution: Minimum Change for a Specific Tree Topology *Systematic Zoology* **20**: 406-416.

Gross, S.S. and M.R. Brent. 2006. Using multiple alignments to improve gene prediction. *J Comput Biol* **13:** 379-393.

Kent, W.J. 2002. BLAT--the BLAST-like alignment tool. *Genome Res* **12:** 656-664.

Kent, W.J., R. Baertsch, A. Hinrichs, W. Miller, and D. Haussler. 2003. Evolution's cauldron: duplication, deletion, and rearrangement in the mouse and human genomes. *Proc Natl Acad Sci U S A* **100:** 11484-11489.

Lenhard, B., A. Sandelin, L. Mendoza, P. Engstrom, N. Jareborg, and W.W. Wasserman. 2003. Identification of conserved regulatory elements by comparative genome analysis. *J Biol* **2:** 13.

Lopez, R., V. Silventoinen, S. Robinson, A. Kibria, and W. Gish. 2003. WU-Blast2 server at the European Bioinformatics Institute. *Nucleic Acids Res* **31:** 3795-3798.

Lunter, G. Rocco, A. Mimouni, N. Heger, A. Caldeira, A. Hein, J. 2008. Uncertainty in homology inferences: Assessing and improving genomic sequence alignment. Genomer research, **18**: 298-309

Ma, B., J. Tromp, and M. Li. 2002. PatternHunter: faster and more sensitive homology search. *Bioinformatics* **18:** 440-445.

Ma, J., L. Zhang, B.B. Suh, B.J. Raney, R.C. Burhans, W.J. Kent, M. Blanchette, D. Haussler, and W. Miller. 2006. Reconstructing contiguous regions of an ancestral genome. *Genome Res* **16:** 1557-1565.

Majoros, W.H., M. Pertea, and S.L. Salzberg. 2005. Efficient implementation of a generalized pair hidden Markov model for comparative gene finding. *Bioinformatics* **21:** 1782-1788.

Margulies, E.H., C.W. Chen, and E.D. Green. 2006. Differences between pair-wise and multi-sequence alignment methods affect vertebrate genome comparisons. *Trends Genet* **22:** 187-193.

Morgenstern, B. 2000. A space-efficient algorithm for aligning large genomic sequences. *Bioinformatics* **16:** 948-949.

Morgenstern, B., K. Frech, A. Dress, and T. Werner. 1998. DIALIGN: finding local similarities by multiple sequence alignment. *Bioinformatics* **14:** 290-294.

Moses, A.M., D.Y. Chiang, D.A. Pollard, V.N. Iyer, and M.B. Eisen. 2004. MONKEY: identifying conserved transcription-factor binding sites in multiple alignments using a binding site-specific evolutionary model. *Genome Biol* **5:** R98.

Nelesen, S., K. Liu, D. Zhao, C.R. Linder, and T. Warnow. 2008. The effect of the guide tree on multiple sequence alignments and subsequent phylogenetic analyses. *Pac Symp Biocomput*: 25-36.

O'Brien, K.P., Remm, M.m and Sonnhammer, E.L.L. 2005 Inparanoid: a comprehensive database of eukaryotic orthologs Nucleic Acids Res. **33**: D476

Paten B, Herrero J, Beal K, Fitzgerald S, Birney E. 2008. Enredo and Pecan: Genome-wide mammalian consistency-based multiple alignment with paralogs. Genome Research, in press.

Pennacchio, L.A., N. Ahituv, A.M. Moses, S. Prabhakar, M.A. Nobrega, M. Shoukry, S. Minovitsky, I. Dubchak, A. Holt, K.D. Lewis, I. Plajzer-Frick, J. Akiyama, S. De Val, V. Afzal, B.L. Black, O. Couronne, M.B. Eisen, A. Visel, and E.M. Rubin. 2006. In vivo enhancer analysis of human conserved non-coding sequences. *Nature* **444**: 499-502.

Prabhakar, S., F. Poulin, M. Shoukry, V. Afzal, E.M. Rubin, O. Couronne, and L.A. Pennacchio. 2006. Close sequence comparisons are sufficient to identify human cis-regulatory elements. *Genome Res* **16:** 855-863.

Raphael, B., Zhi, D., Tang, H., and Pevzner, P. 2004. A Novel Method for Multiple Alignment of Sequences with Repeated and Shuffled Elements. *Genome Research* **14**: 2336-2346.

Sankoff, D. and M. Blanchette. 1998. Multiple genome rearrangement and breakpoint phylogeny. *J Comput Biol* **5:** 555-570.

Schwartz, S., W.J. Kent, A. Smit, Z. Zhang, R. Baertsch, R.C. Hardison, D. Haussler, and W. Miller. 2003. Human-mouse alignments with BLASTZ. *Genome Res* **13:** 103-107.

Thompson, J.D., D.G. Higgins, and T.J. Gibson. 1994. CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucleic Acids Res* **22:** 4673-4680.