Benchmarking tools for the alignment of functional noncoding DNA.

Daniel A. Pollard (dpollard@socrates.berkeley.edu) [1], Casey M. Bergman (cbergman@gen.cam.ac.uk) [2,3,‡,*], Jens Stoye (stoye@techfak.uni-bielefeld.de) [4], Susan E. Celniker (celniker@fruitfly.org) [2,3], and Michael B. Eisen (mbeisen@lbl.gov) [2,5]

[1] Biophysics Graduate Group, University of California, Berkeley, CA 94720, USA

[2] Department of Genome Science, Life Science Division, Lawrence Orlando Berkeley National Laboratory, Berkeley, CA 94720, USA

[3] Berkeley *Drosophila* Genome Project, Lawrence Berkeley National Laboratory, Berkeley, CA 94720, USA

[4] Technische Fakultät, Universität Bielefeld, 33594 Bielefeld, Germany

[5] Department of Molecular and Cell Biology, University of California, Berkeley, CA 94720, USA

[‡] Current address: Department of Genetics, University of Cambridge, Cambridge, UK CB2 3EH

[*] corresponding author.

Abstract

Background

Numerous tools have been developed to align genomic sequences. However, their relative performance in specific applications remains poorly characterized. Alignments of protein-coding sequences typically have been benchmarked against "correct" alignments inferred from structural data. For noncoding sequences, where such independent validation is lacking, simulation provides an effective means to generate "correct" alignments with which to benchmark alignment tools.

Results

Using rates of noncoding sequence evolution estimated from the genus *Drosophila*, we simulated alignments over a range of divergence times under varying models incorporating point substitution, insertion/deletion events, and short blocks of constrained sequences such as those found in *cis*-regulatory regions. We then compared "correct" alignments generated by a modified version of the `ROSE` simulation platform to alignments of the simulated derived sequences produced by eight pairwise alignment tools (`Avid`, `BlastZ`, `Chaos`, `ClustalW`, `DiAlign`, `Lagan`, `Needle`, and `WABA`) to determine the off-the-shelf performance of each tool. As expected, the ability to align noncoding sequences accurately decreases with increasing divergence for all tools, and declines faster in the presence of insertion/deletion evolution. Global alignments tools (`Avid`, `ClustalW`, `Lagan`, and `Needle`) typically have higher sensitivity over entire noncoding sequences as well as in constrained sequences. Local tools (`BlastZ`, `Chaos`, and `WABA`) have lower overall sensitivity as a consequence of incomplete coverage, but have high specificity to detect constrained sequences as well as high sensitivity within the subset of sequences they align. Tools such as `DiAlign`, which generate both local and global outputs, produce alignments of constrained sequences with both high sensitivity and specificity for divergence distances in the range of 1.25-3.0 substitutions per site.

Conclusion

For species with genomic properties similar to *Drosophila*, we conclude that a single pair of optimally diverged species analyzed with a high performance alignment tool can yield accurate and specific alignments of functionally constrained noncoding sequences. Further algorithm development, optimization of alignment parameters, and benchmarking studies will be necessary to extract the maximal biological information from alignments of functional noncoding DNA.

Background

The increasing availability of genome sequences of related organisms offers myriad opportunities to address questions in gene function, genome organization and evolution, but also presents new challenges for sequence analysis. Many classical tools for sequence analysis are obsolete, and there has been active effort in recent years to develop tools that work efficiently with whole genome data. Aligning long genomic sequences – the first step in many analyses – is substantially more complex and computational taxing than aligning short sequences, and many methods have been developed in recent years to address this challenge (reviewed in [1, 2]). Nevertheless, comparative genomic researchers are still faced with the task of making decisions such as which alignment tools to use and which genomes to compare for their particular application. Benchmarking studies that address both the selection of alignment methods and the choice of species can provide the needed framework for informed application of genomic alignment tools and biological discovery in the field of comparative genomics.

Research in alignment benchmarking has focused on the alignment of protein-coding sequences [3, 4], where independent evidence (either the three-dimensional structure of a protein sequence [5, 6] or cDNA sequence [7, 8]) is available to use as a "gold standard" to assess the relative performance of different alignment tools. In contrast, little is known about the relative performance of tools to align noncoding sequences, which comprise the vast majority of metazoan genomes and contain many functional sequences including *cis*-regulatory elements that control gene regulation. For noncoding sequences, little external evidence is available to evaluate alignment tool performance. Benchmarking, however, can be achieved by simulating sequence divergence *in silico* where it is possible to generate sequences that are related by a known, “correct” alignment [9]. Simulation experiments have been used extensively to assess the performance of different methods for phylogenetic reconstruction [10]. Yet only a few studies to date have exploited simulated data to benchmark alignment tools [11-18], and currently none have done so explicitly for the purposes of functional noncoding sequence alignment.

Here we present results of a simulation-based benchmarking study designed to assess the performance of eight tools (`Avid`, `BlastZ`, `Chaos`, `ClustalW`, `DiAlign`, `Lagan`, `Needle`, and `WABA`) for the pairwise alignment of noncoding sequences. We have chosen to address the question of pairwise alignment since pairwise alignment methods often are used in the construction of multiple alignments, since the evaluation of pairwise alignment performance is more tractable than that of multiple alignment, and since pairwise alignment performance is an important part of a general assessment of noncoding alignment strategies. We have chosen to model noncoding sequence evolution in the genus *Drosophila* as a biological system for methodological evaluation, because of the high quality sequence and annotations available for *D. melanogaster* [19, 20], and the recent availability of the genome sequence for the related species, *D. pseudoobscura* [21]. In addition, because of the high rate of deletion as well as the relatively low density of repetitive DNA as compared with mammalian genomes [22-24], *Drosophila* noncoding regions are likely to be enriched for sequences under functional constraint. Previous results indicate that *Drosophila* noncoding regions contain an abundance of short blocks of highly conserved sequences, but that the detection of these sequences is dependent on the alignment method used [25]. Optimizing strategies for the accurate identification of functionally constrained noncoding sequences will play a critical role in the annotation of *cis*-regulatory elements and other important noncoding sequences in *Drosophila* as well as other metazoan genomes.

In this study, we use empirically-derived estimates to parameterize simulations of noncoding sequence evolution over a range of divergences that includes those between species commonly used in comparative genomics such as *H. sapiens-M. musculus* [26, 27], *C. elegans-C. briggsae* [28, 29] and *D. melanogaster-D. pseudoobscura* [30, 31]. Alignments of simulated descendent sequences produced by the tools under consideration were compared to correct alignments and various performance measures were calculated. In general, we find that global tools (`Avid`, `ClustalW`, `DiAlign-G`, `Lagan`, and `Needle`), which align the entirety of input sequences, tend to have the highest accuracy over entire sequences as well as within interspersed blocks of constrained sequences, but both measures were decreasing functions of divergence. Local tools (`BlastZ`, `Chaos`, `DiAlign-L`, and `WABA`), which align subsets of input sequences, tend to have the highest accuracy for the portion of the sequences they align, but the proportion of sequences included in their alignments decreased quickly with increasing divergence distance. For intermediate to high divergences, local tools also showed a high specificity for only aligning interspersed blocks of constrained sequences. Despite these general trends, we find that some tools can systematically out-perform others over a wide range of divergence distances. These results should prove useful for comparative genomics researchers and algorithm developers alike.

Results

Properties of noncoding DNA in *Drosophila*

To make our simulation results as biologically meaningful as possible, we estimated properties of noncoding regions in *D. melanogaster* using Release 3 euchromatic genome sequences and annotations [19, 20]. As described in the methods, we masked all annotated coding exons and known transposable elements to derive a data set of unique sequences representative of noncoding regions in the *D. melanogaster* genome. In total, we obtained 55,325 noncoding regions ranging in size from 1 to 156,299 bp with two modes at approximately 70 and 500 bp (Figure 1). Greater than 95% of noncoding sequences in the *D. melanogaster* genome are less than 10 Kb in length, thus 10 Kb was used as the sequence length for our simulations. Nucleotide frequencies derived from this set of noncoding regions were used to parameterize both our model of noncoding DNA as well as our substitution model used in our simulations.

Estimates of divergence between taxa used in comparative genomics

To link our simulations to species commonly used in comparative genomic analyses of noncoding DNA, we estimated silent site divergence (K_s) between *H. sapiens* vs. *M. musculus, C. elegans* vs. *C. briggsae*, and *D. melanogaster* vs. *D. pseudoobscura* (see methods). Since estimates of K_s are highly dependent on methodology, we sought to generate estimates between these three species pairs using a single method. We estimate the mean (and median) of K_s measured in expected number of substitutions per silent site, for these species pairs to be: *H. sapiens* vs. *M. musculus* 0.64 (0.56); *C. elegans* vs. *C. briggsae*, 1.39 (1.26); and *D. melanogaster* vs. *D. pseudoobscura*, 2.40 (2.24). We note that these divergence estimates do not underlie our simulation, but rather are intended to frame the interpretation of our simulation results in a biological context.

Simulating noncoding sequence evolution

Using a model of noncoding DNA, parameterized with *D. melanogaster* nucleotide frequencies (see Methods for details), we generated 10 Kb sequences which were used as "ancestral" inputs to the `ROSE` sequence evolution simulation program [9, 32] to create pairs of "derived" output sequences. It is important to note that `ROSE` provides both pairs of derived sequences and their correct alignment, and that the modifications to `ROSE` implemented here allow ancestral constraints to be mapped onto derived sequences. Sequence evolution in `ROSE` occurred under four simulation regimes: A) without insertion/deletion (indel) evolution and without constrained blocks; B) with indel evolution and without constrained blocks; C) without indel evolution and with constrained blocks; and D) with indel evolution and with constrained blocks. Regime D is the most realistic and relevant for the interpretation of real biological data. Other regimes were used to calibrate the outputs of our simulations and address the effects of different models of evolution on noncoding sequence alignment. Under each regime, 1,000 replicate pairs of sequences were evolved to each of eleven divergence distances ranging from 0.25 to 5.0 substitutions per site. Levels of constraint as well as relative evolutionary rates of constrained to unconstrained sites and of indels to point substitution were chosen based on previously reported estimates from the literature (see Table 1 and Methods).

Characterization of simulation outputs

To characterize simulation outputs, derived pairs of sequences in alignments provided by ROSE were analyzed for the following measures: estimated overall divergence, estimated divergence in constrained blocks, estimated divergence in unconstrained blocks, overall identity, identity in constrained blocks, identity in unconstrained blocks, fraction of ancestral sequence remaining, fraction of sequences constrained, and differences in length. These simulation statistics are summarized in Figure 2 and demonstrate that the expected outputs of our simulations are observed. In the absence of constrained blocks, estimated overall divergences correspond well with the input distance parameters up to 3.0-4.0 substitutions per site (Figure 2A and 2B, black boxes). In the presence of constrained blocks, estimated overall divergences (Figure 2C and 2D, black boxes) are less than the input distance parameters because these sequences are made up of both unconstrained sites evolving at the rate set by the input parameter (Figure 2C and 2D, brown triangles) as well as blocks of constrained sites evolving ten times more slowly (Figure 2C and 2D, grey circles). The more pronounced deviation of the estimated overall divergences from the input distance parameters in the regime with indel evolution (Figure 2C vs. 2D) is due to preferential deletion of sequence under no constraint which enriches for constrained sites and leads to a decrease in estimated divergences.

Overall identity between derived pairs in the regimes without constrained blocks decreases to the random background of 0.26 (the sum of the squares of the mononucleotide frequencies) by 5.0 substitutions per site with and without indel evolution (Figure 2A and 2B, red crosses). In the regimes with constrained blocks, unconstrained sites have the same level of identity as entire sequences in the regimes without constrained blocks (Figure 2C and 2D, green diamonds), whereas the identity in the constrained blocks is much greater (Figure 2C and 2D, yellow x’s). In the regimes with indel evolution, the fraction of the ancestral sequence remaining diminishes most quickly in the absence of constrained blocks (Figure 2B, green triangles). In regime C (with constrained blocks and without indel evolution), the fraction of constrained sites in derived sequences matches the input parameter of 0.2 (Figure 2C, blue checked-boxes). However, in regime D (with constrained blocks and indel evolution), the fraction of constrained sites in derived sequences decreases below the input parameter of 0.2 at large divergence distances (Figure 2D, blue checked-boxes). This is because the derived sequences are on average longer than ancestral sequences in regime D, differing by 300-400 bp at 1 substitution per site, 400-500 bp at 2 substitutions per site and 700-800 bp at 5 substitutions per site. In our simulation there are equal input rates of insertion and deletion, however deletions are unable to extend into constrained blocks and are omitted, creating a net excess of insertions to deletions. This phenomenon was recently proposed as a possible explanation for differences in observed insertion:deletion ratios in unconstrained dead-on-arrival retrotransposon pseudogenes versus noncoding sequences flanking genes [33].

Comparative analysis of genomic alignment tools

Unaligned pairs of derived sequences generated by ROSE were used as input to each of the eight genomic alignment tools (see Methods) and resulting alignments were compared to the simulated alignments produced by ROSE. Our objective was to test the off-the-shelf performance of these tools over a wide range of different divergences, so each tool was run using default parameter settings. In addition, BlastZ and Chaos were run using author suggested settings (BlastZ-A and Chaos-A), as described in the Methods. We note that the output of DiAlign can be treated as both a global alignment as well as a local alignment, so we analyzed both (DiAlign-G and DiAlign-L). Alignments produced by each tool were scored for the overall coverage and overall sensitivity for all regimes (A-D), and were also scored for constraint coverage, constraint sensitivity, constraint specificity, and local constraint sensitivity in the regimes with constrained blocks (C and D) (see Methods for details).

Coverage

Overall coverage was measured to understand the proportion of ungapped, orthologous pairs of sites in the simulated alignment that were aligned by local tools under various evolutionary scenarios. The coverage of each tool under the four simulation regimes is a decreasing function of divergence for local (but not global) tools (Figure 3). In the absence of constrained blocks, local tools tend to align most or all of the sequences for only small divergence distances (0.25-1.0 substitutions per site), but little or none of the sequences for intermediate to large divergence distances (Figure 3A and 3B). [For convenience, for the remainder of this report we shall refer to 0.25-1.0 substitutions per site as small distances, 1.25-3.0 substitutions per site as intermediate distances, and 4.0-5.0 substitutions per site as large distances.] One exception is Chaos, which has negligible coverage past 0.25 substitutions per site. In the presence of constrained blocks, the coverage of local tools improves substantially at all but the most extreme divergence distances. WABA, which was typical of local tools in the absence of constrained blocks, maintains high coverage out to more than twice the divergence distance of the rest of the local tools in the presence of constrained blocks. WABA also appears to be relatively unaffected by indel evolution, while the other local tools show a reduction in coverage of about 0.5 substitutions per site in regimes with indel evolution (Figure 3A vs. 3B, 3C vs. 3D).

Sensitivity

Overall sensitivity was measured to understand the accuracy of each tool to align all orthologous nucleotide sites under various evolutionary scenarios. The sensitivity of each tool under the four simulation regimes is a decreasing function of divergence for both local and global tools (Figure 4). It is important to note that the maximum sensitivity a tool can attain is limited by its coverage. Thus for most divergence distances, global tools (which by definition have complete coverage) have greater potential for high sensitivity relative to local tools, which have incomplete coverage (see above, Figure 3). Nevertheless, with the exception of WABA, the sensitivity of local tools tends to remain very close to the maximum set by their coverage. This implies that although local tools have diminishing coverage with divergence, the portion of the sequence they do align is aligned quite accurately (see below). Despite the trend of high sensitivity in aligned regions for local tools, the sensitivity of the top global tools tends to be as good as or better than the sensitivity for the top local tools (Figure 4). This is particularly true for intermediate to high divergence distances in the absence of indel evolution. In each of the four regimes, at least one global tool has a higher sensitivity than the next best local tool for intermediate to high divergence distances. In the most biologically relevant regime D, the sensitivity of the highest performing tools (such as `Lagan` and `DiAlign`) plateaus over the range of 1.25-3.0 substitutions per site at higher than 0.35, implying that sites other than those in constrained blocks are being accurately aligned (Figure 4D). In contrast, in the absence of constraint but with indels (regime B), the sensitivity of all alignment tools is practically nil for divergences greater than 1 substitution per site (Figure 4B).

Coverage and sensitivity in constrained sequences

Alignment coverage and sensitivity across all orthologous sites are informative for understanding the overall performance of a tool, but, for many applications (such as aligning characterized *cis*-regulatory elements), researchers may only be interested in accurately aligning functionally constrained sites. To assess the ability of each tool to align potentially functional portions of sequences we measured the coverage and sensitivity only for orthologous nucleotide sites within constrained blocks (Figure 5). Constraint coverage is better than overall coverage for local tools but the degree of improvement varies considerably (Figure 5A and 5B). `BlastZ`, `BlastZ-A` and `WABA` all have very similar overall and constraint coverage, suggesting little discrimination in attempting to align constrained versus unconstrained sites. In contrast, `DiAlign-L` and

Chaos-A have much improved constraint coverage compared with overall coverage, suggesting a preferential alignment of constrained sites. For example in the presence of indels, DiAlign-L accurately aligns 86% and 64% of constrained sequences at divergences between 1.25 and 3.0 substitutions per site.

Constraint sensitivity of all tools is much better than overall sensitivity but, as with constraint coverage, the degree of improvement varies considerably across tools (Figure 5C and 5D). Similar to overall sensitivity, global tools tend to maintain the highest sensitivity out to large divergence distances in the presence of constrained sites. It is of note that in the presence of indel evolution (Figure 5D), constraint sensitivity of the best performing global tools (as well as the local Dialign-L) closely parallels the decrease in identity of constrained sites (Figure 2D), suggesting that they are attaining near-maximal constraint sensitivity. Most tools show only moderate decreases in constraint sensitivity in the presence of indel evolution but a few, like ClustalW, Chaos-A, and BlastZ have dramatic decreases in constraint sensitivity in the presence of indel evolution.

Specificity to detect constrained sequences

Constraint coverage and constraint sensitivity reveal the ability of alignment tools to detect and align *all* orthologous nucleotides sites within constrained blocks, but for some purposes (like *cis*-regulatory element prediction) researchers may want to align *only* constrained nucleotide sites and nothing else, even at the expense of missing some functionally constrained sites. To evaluate the ability of each tool to provide high quality alignments of just potential functionally constrained sites, we measured their constraint specificity and local constraint sensitivity. As shown in Figure 6, constraint specificity is an increasing function of divergence for most tools because unconstrained sequences accumulate mismatches and indels more quickly than the constrained blocks and are thus more likely to be gapped or left out of local alignments. This is particularly true for local tools where decreasing coverage can increase constraint specificity, and less so for global tools for which it is gap parameters that predominantly affect constraint specificity at different divergence distances. Most tools have higher constraint specificity in the presence of indel evolution, although this trend is less pronounced in the highest specificity tools, `Chaos` and `DiAlign-L`. All local tools except `WABA` increase quickly until they reach a constraint specificity of 0.8-0.9 at which point their constraint specificity plateaus. In the presence of indel evolution, near-maximal constraint specificity is achieved between 1.25 and 3.0 substitutions per site.

Local constraint sensitivity (Figure 6) is equivalent to constraint sensitivity (Figure 5) for the global tools, but for the local tools it differs in that it is a measure of their constraint sensitivity just within the subsequences they align. For `BlastZ`, `BlastZ-A`, `Chaos`, and `DiAlign-L`, local constraint sensitivity is nearly maximal (1.0) with and without indel evolution across all divergences studied. For `Chaos-A` and `WABA`, local constraint sensitivity varies with divergence distance and is less than the other local tools. Thus local tools can produce nearly perfect alignments within constraint blocks while maintaining relatively high constraint specificity, though it is important to note that this may not be meaningful if the coverage of a tool is extremely low (e.g. `BlastZ`, `BlastZ-A`, `Chaos`).

Discussion

In this report we investigate the performance of eight pairwise genomic alignment tools to align functional noncoding DNA such as that found in metazoan *cis*-regulatory regions. To do so, we have used a biologically-informed simulation approach to determine off-the-shelf performance over a range of divergence distances. This study provides important information regarding the ability of genomic alignment tools to identify and align constrained sequences in noncoding regions, which would not otherwise be possible. We argue that a simulation study is necessary to achieve our goal since large datasets of functionally annotated noncoding sequences are not available to use as "gold standards" of alignment accuracy. Likewise, datasets of large orthologous genomic regions spanning a range of divergence distances are only recently becoming available [31, 34]. As is common in alignment benchmarking [4, 17, 35], we have studied performance of alignment tools using default parameters since fundamental differences in objective functions, scoring matrices, the type and values of parameters, and algorithmic design prevent a systematic exploration of parameter space.

We have attempted to construct a realistic simulation of noncoding sequence evolution and test alignment performance for species with genomic properties similar *Drosophila*. Noncoding alignment assessment for mammalian and other species with large, repeat-rich genomes would require modifications to our current simulation, such as the inclusion of ancestral repeats and lineage-specific transposition events. Moreover, as more becomes known about the substitution process in noncoding regions (especially those under weak primary sequence constraint), it will be important to implement more realistic models such as context-dependent substitution [36-38]. It would be also instructive to assess alignment performance based on a simulation that decouples suppression of indel rates from substitution rates, given the possibility that the spacing (but not the primary sequence) between conserved noncoding segments may be constrained [31]. In addition, though we have attempted to be systematic in our evaluation of tools, we unfortunately cannot have included all available pairwise alignment tools. As new pairwise alignment tools emerge and old tools are modified or brought to our attention, we will update our results periodically on the web using the same set of simulated alignments presented here [39]. Moreover, assessment of tools which take advantage of the phylogenetic information and higher signal-to-noise inherent in multiple alignments will be an essential extension to this work to provide a more general evaluation of strategies for noncoding alignment.

From the standpoint of the most biologically relevant simulation regime studied here (D, which includes indel evolution and interspersed blocks of constrained sequences), our results indicate that global alignment tools have the highest sensitivity in general to align orthologous sites accurately in noncoding sequences, as well as blocks of constrained sites (Figures 4D, 5D). We find that constraint sensitivity of the top global tools can be quite high (>75%) and limited only by sequence identity in constrained sites at intermediate divergence distances (1.25-3.0 substitutions per site), whereas overall sensitivity is relatively low beyond such intermediate divergence distances. The improved performance of global tools over local tools is largely a consequence of incomplete coverage of both constrained and unconstrained sites in alignments produced by local tools (Figure 3). The subset of sequences aligned by the highest performing local tools, however, is accurately aligned and specifically corresponds to constrained sites (Figure 6). In fact, most local tools can effectively discriminate between constrained and unconstrained sites to greater than 80% specificity at intermediate divergence distances while the constrained portions of their alignments are nearly perfectly aligned at large divergence distances. Finally, when compared with regime C (which excludes indel evolution but includes interspersed constrained blocks), it is clear that our model of indel evolution affects alignment coverage, sensitivity and specificity, but not enough to overturn these major trends.

These results have important implications for the analysis of functional noncoding sequences. First, if a researcher's goal is to align all constrained sites in a noncoding region, then a global tool like `Lagan` will reliably produce the best results, but will require post-processing to identify constrained sequences [40, 41]. Conversely, if one's goal is to align only constrained blocks in a noncoding region, then a local tool like `Chaos` will reliably produce the best results, provided that complete recovery of all constrained sequences is not required. The distinct virtues of both global and local tools are currently incorporated in the output of only one alignment tool, `DiAlign`. For this reason, use of the global parse of `DiAlign` (`DiAlign-G`) can provide high coverage and sensitivity across entire noncoding regions, while use of the local parse of `DiAlign`

(DiAlign-L) will specifically provide highly accurate alignments of blocks of constrained sites. In light of these results, we recommend the further development of global alignment tools that also output a local parse of high confidence local alignments contained within, which should be possible since local anchors are often used in the construction of the global alignment (e.g. [7, 8]).

Our results also indicate that for species with structural and evolutionary constraints on noncoding sequences such as those found in *Drosophila*, DiAlign can produce alignments with high coverage and sensitivity, as well as high specificity to detect constrained sites in the range of 1.25-3.0 substitutions per site. Since the divergence between *D. melanogaster* vs. *D. pseudoobscura* and between *C. elegans* vs. *C. briggsae* falls within this range, we suggest that the use of DiAlign for detecting functionally constrained noncoding sequences will prove successful in these taxa on a genomic scale. In contrast, our results also indicate that species pairs such as *H. sapiens* and. *M. musculus* may not be sufficiently diverged for a single pairwise comparison to provide the needed resolution to detect functionally constrained noncoding sequences, though differences in genome organization and evolution between flies and mammals require a more thorough evaluation of this claim. This conclusion, however, supports results based on Poisson modelling of point substitution that approximately 3 substitutions per site would be needed to detect functional constrained sites reliably in mammalian noncoding DNA [42].

Finally, the results presented here also imply that biological and technical conditions exist with which to study with accuracy the evolutionary events underlying the process of *cis*-regulatory evolution in flies and worms. Current evolutionary models of *cis*-regulatory sequence divergence posit the gain and loss of transcription factor binding sites, even under constant functional constraints [43, 44]. However, the absence of alignable binding sites in comparisons of divergent sequences may result from inaccuracies in alignment as well as the *bona fide* loss of transcription factor binding sites. We suggest that alignments of noncoding sequences using tools such as `DiAlign` in the range of 1.25-3.0 substitutions per site are of sufficient accuracy to measure binding site loss among divergent species pairs, such as the high levels recently reported in the genus *Drosophila* [45, 46].

Conclusions

Our study demonstrates that recently developed alignment tools have the potential to produce biologically meaningful alignments of functional noncoding DNA on a genome scale. Continued development of alignment algorithms in conjunction with parameter optimization and continued benchmarking will be necessary to provide the highest quality genomic alignments under the wide diversity of genomic and evolutionary scenarios to be studied.

Methods

Modelling input sequences for the simulation of *Drosophila* noncoding DNA.

To generate biologically relevant input sequences for our simulation, we estimated properties of noncoding sequences in the genome sequences of the fruitfly, *D. melanogaster*. First we extracted all noncoding regions from the Release 3 *D. melanogaster* genomic sequences based on annotations in the Gadfly database [19, 20, 47]. This was accomplished by masking all DNA corresponding to coding exons, producing inter-coding-exon intervals. Subsequent to extracting noncoding regions, transposable elements were masked using annotations in Gadfly to create "pre-integration" noncoding sequences. In our analysis, we chose to treat all noncoding sequences (intergenic, intronic, untranslated region) together since many noncoding sequences cannot be unambiguously categorized because of alternative splicing or alternative promoter usage. Moreover, previous results revealed that similar evolutionary constraints act on intergenic and intronic sequences in *Drosophila* [25]. Summary statistics of noncoding sequence lengths were calculated using the R statistical package (Figure 1) [48].

The probabilistic dependence of adjacent bases in *D. melanogaster* noncoding sequences was assessed by Markov chain analysis in order to create an accurate model of random noncoding sequences [49]. TE-masked noncoding sequences were concatenated, and n-mers of size 1 to 10 were counted. Counts of reverse complementing n-mers were averaged, and used to estimate frequencies of each n-mer [50]. Based on these counts and frequencies, we determined the likelihood of Markov chains of orders 1 through 9 describing *Drosophila* noncoding sequences, and evaluated the likelihood of each Markov chain using the Bayesian information criterion [49, 51]. This analysis revealed that *D. melanogaster* noncoding sequences are best modeled by a 7th-order Markov chain (data not shown). We therefore created the ancestral input sequences for our evolution simulations using a 7th-order Markov chain. We note that because our evolutionary simulation models bases independently (see below), the higher order structure of these ancestral input sequences was not maintained in the more divergent derived output sequences. Nevertheless, sequences generated by a 0th-order Markov chain gave qualitatively and quantitatively similar simulation and alignment results, with correlation among performance measures for the 0th-order and 7th-order generated sequences exceeding an r^2 of 0.97 (data not shown).

Divergence estimates in flies, worms and mammals.

Estimates of silent site divergence (K_s) between *H. sapiens* vs. *M. musculus*, *C. elegans* vs. *C. briggsae*, and *D. melanogaster* vs. *D. pseudoobscura* were obtained using the `yn00` method in `PAML` (version 3.13) [52, 53]. The mean and median of K_s were calculated for 29 fly, 193 worm, and 153 mammalian coding sequence alignments taken from references [31], [28] and [26], respectively.

Simulating noncoding sequence divergence.

Noncoding sequence evolution was simulated using a modified version of the sequence simulation program `ROSE` [9]. In general, in the absence of large datasets of noncoding sequences from closely related *Drosophila* species, we have taken estimates of noncoding evolution from previous results reported in the literature. Beginning with ancestral sequences, evolution occurred on two descendent branches of equal length under the HKY model of point substitution [54], with a transition/transversion bias of 2 to reflect the nucleotide and transition biases observed in *Drosophila* noncoding sequences [25, 55, 56]. The substitution rate was set to 0.01 such that a branch length unit was on average 0.01 substitutions per site. Total branch lengths spanned a range of divergence times from 0.25 to 5.0 substitutions per site. Insertion/deletion evolution was based on the length distribution of polymorphic indels estimated in [57], and occurred at a 10-fold lower rate than point substitution, approximating relative rates estimated in [22, 23].

To model the evolution of constrained blocks in noncoding sequences a modification of the `ROSE` sequence simulation program was developed to map constraints on ancestral sequences onto derived sequences (available for download as ROSE version 1.3 from [58]). Constraints on noncoding sequences were modelled as short blocks of highly conserved sequences typical of *cis*-regulatory sequences, and follow a lognormal distribution with parameters estimated in [25]. On average, interspersed blocks of constrained sites accounted for 20% of the sites in ancestral sequences, a conservative estimate of constraint in *Drosophila* noncoding DNA [25]. Parameters used in our simulations are summarized in Table 1.

Estimation of evolutionary distance for simulated alignments was performed using the F84 model of sequence evolution in the `DnaDist` program of the `PHYLIP` package [59] with a transition:transversion ratio of 1.0 (note that a transition:transversion ratio of 1.0 in PHYLIP is equivalent to a transition/transversion bias of 2 in ROSE, see discussion in [53]). Summary statistics for the simulations were calculated using the R statistical package (Figure 2) [48].

Tools for aligning noncoding DNA.

The alignment tools tested in this study were chosen based on the criteria that they are (1) publicly available, (2) run in batch mode from the command line and are able to produce (3) strictly co-linear, (4) error-free, pairwise genomic alignments of sequences (5) up to 10 Kb in length. Tools like BBA [60] (5), Bl2seq [61] (3), DBA [62] (4), MUMmer [63] (3), Owen [64] (2) and SSEARCH [65] (3) were not evaluated since they do not satisfy one of these criteria. We now briefly describe the tools that we tested.

Avid [7] is a pairwise global alignment tool whose general strategy for aligning two sequences is to anchor and align iteratively. A set of maximal (but not necessarily unique) matches between the sequences is constructed using a suffix tree. Dynamic programming is used to order and orient the longest matches, which are then fixed. For each subsequence remaining between the fixed matches, the process is repeated until every base is aligned. When sequences are short and the matches make up less than half of the total sequence, the program defaults to the Needleman-Wunsch algorithm [66].

The Chaos/Lagan [8] suite of tools consists of a pairwise local alignment tool, Chaos, and a global alignment tool, Lagan. Chaos starts by finding all words between the two sequences of a specified length and a specified maximum number of mismatches. These words are then chained together if they are close together in both sequences. These maximal chains are then scored and all chains that are above a specified threshold are returned. `Lagan` starts by running `Chaos` with conservative parameter settings and then finds the optimal path through the maximal chains using dynamic programming. `Lagan` then recursively calls `Chaos` with increasingly more permissive parameters on the regions between each maximal chain in the optimal path. When the recursion has created a dense map of maximal chains that have been ordered with dynamic programming, `Lagan` runs the Needleman-Wunsch algorithm on the whole length of both sequences but puts close bounds around the maximal chains to provide the final global alignment. `Chaos` was run on default parameters as well as using parameters suggested by the authors: word length = 7, number of degeneracies = 1, score cut-off = 20 and extension mode on.

`BlastZ` [67] is a pairwise local alignment tool that is based on the gapped `BLAST` algorithm that has been redesigned for the alignment of long genomic sequences. `BlastZ` first removes lineage-specific interspersed repeats from each sequence, then searches for short near-perfect matches between the two sequences. Each match is extended first using gap-free dynamic programming and if it scores above a specified threshold it will be extended using dynamic programming with gaps; extended matches that score above a specified threshold are then kept. Part of the unique implementation of `BlastZ` is that it can be forced to return alignments that are both unique within each sequence as well as collinear with respect to each other. To satisfy our strict collinear requirement, we ran BlastZ with both of these options. Blastz was also run using the author's suggestion of lowering the score cut-off (k) to 2000 (BlastZ-A).

DiAlign (v. 2.1) [68] is a segment-to-segment alignment algorithm. Like the BLAST algorithms, DiAlign looks for short ungapped segments that have a similarity that deviates from what would be expected by random chance, keeping segments with a score above a certain threshold. These high scoring segments are then aligned into a collinear global alignment using a dynamic programming algorithm. DiAlign produces a global alignment but distinguishes high confidence columns of an alignment from low confidence columns. We used DiAlign as both a global (DiAlign-G) and a local (DiAlign-L) alignment tool.

ClustalW (v. 1.8) [69] was used on default settings. ClustalW is a progressive multiple alignment tool that reduces to the Needleman-Wunsch algorithm in the pair-wise case with default parameters of a match score of 1.9, mismatch penalty of 0, a gap open penalty of 10 and a gap extension penalty of 0.1.

The second implementation of the Needleman-Wunsch algorithm used in this study is the needle program in the EMBOSS suite of tools [70]. needle was used with default parameter settings of a match score of 5, a mismatch penalty of 4, a gap open penalty of 10 and a gap extension penalty of 0.5.

The final tool tested, WABA [71], is a three-tier alignment algorithm. The first tier partitions the first sequence into overlapping windows of 2 Kb and then defines a synteny map of high scoring 2 Kb windows of the first sequence onto the second sequence. The second tier then carefully aligns syntenic regions using a seven-state, pair Hidden Markov Model that includes separate query and database insertion/deletion states, high and low noncoding conservation states, as well as three coding states (one for each position in a codon). The final tier then attempts to assemble individual alignments together into a more global alignment.

Alignment performance measures.

The performance of alignment tools was assessed using six basic measures: overall coverage, overall sensitivity, constraint coverage, constraint sensitivity, constraint specificity and local constraint sensitivity. Overall coverage and overall sensitivity were measured for all four evolutionary regimes (A-D) while the constraint measures were only measured in the two regimes that included constrained blocks (C, D). Alignments produced by each alignment tool were parsed to generate the statistics, which were then used to calculate each performance measure.

Each site in an alignment produced by a tool (a site being a base in one strand of a column of an alignment) can have two simulated alignment states, two constraint states, three tool alignment states, and two conditional tool alignment states. The two simulated alignment states are “homolog” (h), ungapped sites in the simulated alignments, and “no homolog” (nh), gapped sites in the simulated alignments. Simulations without indel evolution have only homolog sites since there are no gaps in the simulated alignments. The two constraint states are “constrained” (c), sites in constraint blocks, and “unconstrained” (u), sites not in constrained blocks. The three tool alignment states are “aligned” (a), sites aligned in the tool alignment, “gapped” (g), sites gapped in the tool alignment, and “not aligned” (na), sites not included in a local tool alignment. The two conditional tool alignment states are “aligned correctly” (ac), sites aligned to the same site in both the tool and simulated alignments, and “aligned incorrectly” (ai), sites aligned to different sites in the tool and simulated alignments. There are fourteen possible combinations of these states (e.g. homolog constrained aligned correctly, h_c_ac), giving us fourteen statistics to calculate for each estimated alignment. Counts for each statistic were used to calculate the following measures:

Overall coverage is the fraction of ungapped sites in a simulated alignment that are included in a tool alignment. Overall Coverage = (h_c_ac + h_c_ai + h_c_g + h_u_ac + h_u_ai + h_u_g) / (h_c_ac + h_c_ai + h_c_g + h_c_na + h_u_ac + h_u_ai + h_u_g + h_u_na)

Overall sensitivity is the fraction of ungapped sites in a simulated alignment that are aligned to the correct base in a tool alignment. Overall Sensitivity = (h_c_ac + h_u_ac) / (h_c_ac + h_c_ai + h_c_g + h_c_na + h_u_ac + h_u_ai + h_u_g + h_u_na)

Constraint coverage is the fraction of ungapped constrained sites in a simulated alignment that are included in a tool alignment. Constraint Coverage = (h_c_ac + h_c_ai + h_c_g) / (h_c_ac + h_c_ai + h_c_g + h_c_na)

Constraint sensitivity is the fraction of ungapped constrained sites in a simulated alignment that are aligned to the correct base in a tool alignment. Constraint Sensitivity = (h_c_ac) / (h_c_ac + h_c_ai + h_c_g + h_c_na)

Constraint specificity is the fraction of unconstrained sites in a simulated alignment that are gapped or not included in a tool alignment. Constraint Specificity = (h_u_g + h_u_na + nh_u_g + nh_u_na) / (h_u_ac + h_u_ai + h_u_g + h_u_na + nh_u_a + nh_u_g + nh_u_na)

Local constraint sensitivity is the fraction of sites that are both, contained in a tool alignment and are ungapped constrained sites in a simulated alignment, that are aligned to the correct base in the tool alignment. Local Constraint Sensitivity = (h_c_ac) / (h_c_ac + h_c_ai + h_c_g)

For each of these six measures, a mean and standard error of the mean were calculated for up to 1000 replicates (local tools do not always return an alignment and replicates which produced no alignment were not counted toward the mean) using R.

Authors' contributions

DAP conducted the sequence simulation, alignment accuracy experiments and analyses and drafted the manuscript. CMB conceived of the study, participated in its design and analyses, and drafted the manuscript. JS developed the simulation software. SEC and MBE provided computational infrastructure and participated in the coordination of the study. All authors read and approved the final manuscript.

Acknowledgements

We thank Mark Yandell and Chris Mungall for providing scripts to extract noncoding sequences, Erwin Frise for assistance with cluster computing, and Cristian Castillo-Davis and Anton Nekrutenko for kindly providing worm and mammalian coding sequence alignments. We thank Venky Nandagopal for discussions on alignment specificity, Hunter Fraser, Emily Hare, Alan Moses, and Monty Slatkin for critical reading of the manuscript, and Alex Kondrashov and one anonymous reviewer for helpful comments on the manuscript. This work was supported by NIH grant HG00750 to G. Rubin. DAP is supported by NIH training grant T32 HG00047 to D. Rokhsar and J. Rine. CMB is supported by NIH training grant T32 HL07279 to E. Rubin and by a Royal Society USA Research Fellowship. MBE is a Pew Scholar in the Biomedical Sciences. Research was conducted at the Lawrence Berkeley National Laboratory under Department of Energy contract DE-AC0376SF00098.

References

1. Miller W: **Comparison of genomic DNA sequences: solved and unsolved problems.** *Bioinformatics* 2001, **17**:391-7.

2. Frazer KA, Elnitski L, Church DM, Dubchak I, Hardison RC: **Cross-species sequence comparisons: a review of methods and available resources**. *Genome Res* 2003, **13**:1-12.

3. McClure MA, Vasi TK, Fitch WM: **Comparative analysis of multiple protein-sequence alignment methods**. *Mol Biol Evol* 1994, **11**:571-92.

4. Thompson JD, Plewniak F, Poch O: **A comprehensive comparison of multiple sequence alignment programs**. *Nucleic Acids Res* 1999, **27**:2682-90.

5. Sauder JM, Arthur JW, Dunbrack RL, Jr.: **Large-scale comparison of protein sequence alignment algorithms with structure alignments**. *Proteins* 2000, **40**:6-22.

6. Brenner SE, Chothia C, Hubbard TJ: **Assessing sequence comparison methods with reliable structurally identified distant evolutionary relationships**. *Proc Natl Acad Sci U S A* 1998, **95**:6073-8.

7. Bray N, Dubchak I, Pachter L: **AVID: A Global Alignment Program**. *Genome Res* 2003, **13**:97-102.

8. Brudno M, Do CB, Cooper GM, Kim MF, Davydov E, Program NC, Green ED, Sidow A, Batzoglou S: **LAGAN and Multi-LAGAN: Efficient Tools for Large-Scale Multiple Alignment of Genomic DNA**. *Genome Res* 2003.

9. Stoye J, Evers D, Meyer F: **Rose: generating sequence families**. *Bioinformatics* 1998, **14**:157-63.

10. Hillis DM, Huelsenbeck JP, Cunningham CW: **Application and accuracy of molecular phylogenies**. *Science* 1994, **264**:671-7.

11. Thorne JL, Kishino H, Felsenstein J: **An evolutionary model for maximum likelihood alignment of DNA sequences**. *J Mol Evol* 1991, **33**:114-24.

12. Thorne JL, Kishino H, Felsenstein J: **Inching toward reality: an improved likelihood model of sequence evolution**. *J Mol Evol* 1992, **34**:3-16.

13. Holmes I, Durbin R: **Dynamic programming alignment accuracy.** *J Comput Biol* 1998, **5**:493-504.

14. Stoye J: **Multiple sequence alignment with the Divide-and-Conquer method**. *Gene* 1998, **211**:GC45-56.

15. Hein J, Wiuf C, Knudsen B, Moller MB, Wibling G: **Statistical alignment: computational properties, homology testing and goodness-of-fit**. *J Mol Biol* 2000, **302**:265-79.

16. Katoh K, Misawa K, Kuma K, Miyata T: **MAFFT: a novel method for rapid multiple sequence alignment based on fast Fourier transform**. *Nucleic Acids Res* 2002, **30**:3059-66.

17. Lassmann T, Sonnhammer EL: **Quality assessment of multiple alignment programs**. *FEBS Lett* 2002, **529**:126-30.

18. Metzler D: **Statistical alignment based on fragment insertion and deletion models**. *Bioinformatics* 2003, **19**:490-9.

19. Celniker SE, Wheeler DA, Kronmiller B, Carlson JW, Halpern A, Patel S, Adams M, Champe M, Dugan SP, Frise E, et al: **Finishing a whole genome shotgun sequence assembly: Release 3 of the Drosophila euchromatic genome sequence.** *Genome Biology* 2002, **3**:research0079.1-research0079.14.

20. Misra S, Crosby MA, Mungall CJ, Matthews BB, Campbell K, Hradecky P, Huang Y, Kaminker JS, Millburn GH, Prochnik SE, et al: **Annotation of the Drosophila euchromatic genome: a systematic review.** *Genome Biology* 2002, **3**:research0083.1-research0083.22.

21. **Baylor College of Medicine Drosophila Genome Project** [http://www.hgsc.bcm.tmc.edu/projects/drosophila/]

22. Petrov DA, Lozovskaya ER, Hartl DL: **High intrinsic rate of DNA loss in Drosophila.** *Nature* 1996, **384**:346-349.

23. Petrov DA, Hartl DL: **High rate of DNA loss in the Drosophila melanogaster and Drosophila virilis species groups.** *Mol Biol Evol* 1998, **15**:293-302.

24. Kaminker JS, Bergman CM, Kronmiller B, Carlson J, Svirskas R, Patel S, Frise E, Wheeler DA, Lewis S, Rubin GM, et al: **The transposable elements of the Drosophila melanogaster euchromatin – a genomics perspective**. *Genome Biology* 2002, **3**:research0084.

25. Bergman CM, Kreitman M: **Analysis of conserved noncoding DNA in Drosophila reveals similar constraints in intergenic and intronic sequences.** *Genome Res* 2001, **11**:1335-45.

26. Nekrutenko A, Makova KD, Li WH: **The K(A)/K(S) ratio test for assessing the protein-coding potential of genomic regions: an empirical and simulation study.** *Genome Res* 2002, **12**:198-202.

27. Waterston RH, Lindblad-Toh K, Birney E, Rogers J, Abril JF, Agarwal P, Agarwala R, Ainscough R, Alexandersson M, An P, et al: **Initial sequencing and comparative analysis of the mouse genome**. *Nature* 2002, **420**:520-62.

28. Castillo-Davis CI, Hartl DL: **Genome evolution and developmental constraint in Caenorhabditis elegans**. *Mol Biol Evol* 2002, **19**:728-35.

29. Stein LD, Bao Z, Blasiar D, Blumenthal T, Brent MR, Chen N, Chinwalla A, Clarke L, Clee C, Coghlan A, et al: **The Genome Sequence of Caenorhabditis briggsae: A Platform for Comparative Genomics**. *PLoS Biol* 2003, **1**:E45.

30. Zeng LW, Comeron JM, Chen B, Kreitman M: **The molecular clock revisited: the rate of synonymous vs. replacement change in Drosophila.** *Genetica* 1998, **102-103**:369-82.

31. Bergman CM, Pfeiffer BD, Rincón-Limas DE, Hoskins RA, Gnirke A, Mungall CJ, Wang AM, Kronmiller B, Pacleb J, Park S, et al: **Assessing the impact of comparative genomic sequences data on the functional annotation of the Drosophila genome.** *Genome Biology* 2002, **3**:research0086.1-research0086.20.

32. Stoye J, Evers D, Meyer F: **Generating benchmarks for multiple sequence alignments and phylogenetic reconstructions**. *Proc Int Conf Intell Syst Mol Biol* 1997, **5**:303-6.

33. Ptak SE, Petrov DA: **How intron splicing affects the deletion and insertion profile in Drosophila melanogaster**. *Genetics* 2002, **162**:1233-44.

34. Thomas JW, Touchman JW, Blakesley RW, Bouffard GG, Beckstrom-Sternberg SM, Margulies EH, Blanchette M, Siepel AC, Thomas PJ, McDowell JC, et al: **Comparative analyses of multi-species sequences from targeted genomic regions**. *Nature* 2003, **424**:788-93.

35. Morgenstern B, Frech K, Dress A, Werner T: **DIALIGN: finding local similarities by multiple sequence alignment.** *Bioinformatics* 1998, **14**:290-4.

36. Averof M, Rokas A, Wolfe KH, Sharp PM: **Evidence for a high frequency of simultaneous double-nucleotide substitutions**. *Science* 2000, **287**:1283-6.

37. Arndt PF, Burge CB, Hwa T: **DNA sequence evolution with neighbor-dependent mutation**. *J Comput Biol* 2003, **10**:313-22.

38. Siepel A, Haussler D: **Phylogenetic Estimation of Context-Dependent Substitution Rates by Maximum Likelihood**. *Mol Biol Evol* 2003.

39. **AlignmentBenchmarking** [http://rana.lbl.gov/AlignmentBenchmarking]

40. Boffelli D, McAuliffe J, Ovcharenko D, Lewis KD, Ovcharenko I, Pachter L, Rubin EM: **Phylogenetic shadowing of primate sequences to find functional regions of the human genome**. *Science* 2003, **299**:1391-4.

41. Elnitski L, Hardison RC, Li J, Yang S, Kolbe D, Eswara P, O'Connor MJ, Schwartz S, Miller W, Chiaromonte F: **Distinguishing regulatory DNA from neutral sites**. *Genome Res* 2003, **13**:64-72.

42. Cooper GM, Brudno M, Green ED, Batzoglou S, Sidow A: **Quantitative estimates of sequence divergence for comparative analyses of mammalian genomes**. *Genome Res* 2003, **13**:813-20.

43. Ludwig MZ, Bergman C, Patel N, Kreitman M: **Evidence for stabilizing selection in a eukaryotic cis-regulatory element**. *Nature* 2000, **403**:564-567.

44. Cuadrado M, Sacristan M, Antequera F: **Species-specific organization of CpG island promoters at mammalian homologous genes.** *EMBO Rep* 2001, **2**:586-92.

45. Costas J, Casares F, Vieira J: **Turnover of binding sites for transcription factors involved in early Drosophila development**. *Gene* 2003, **310**:215-20.

46. Emberly E, Rajewsky N, Siggia ED: **Conservation of regulatory elements between two species of Drosophila**. *BMC Bioinformatics* 2003, **4**:57.

47. Mungall CJ, Misra S, Berman BP, Carlson J, Frise E, Harris N, Marshall B, Shu S, Kaminker JS, Prochnik SE, et al: **An integrated computational pipeline and database to support whole genome sequence annotation.** *Genome Biology* 2002, **3**:research0081.1-research0081.11.

48. **Comprehensive R Archive Network** [http://cran.r-project.org/]

49. Weir BS: **Genetic Data Analysis II.** Sunderland, MA: Sinauer Associates, Inc.; 1996.

50. Burge C, Campbell, A.M., Karlin, S.: **Over- and under-representation of short oligonucleotides in DNA sequences.** *Proc Natl Acad Sci U S A* 1992, **89**:1358-1362.

51. Katz RW: **On some criteria for estimating the order of a Markov chain.** *Technometrics* 1981, **23**:243-249.

52. Yang Z, Nielsen R: **Estimating synonymous and nonsynonymous substitution rates under realistic evolutionary models.** *Mol Biol Evol* 2000, **17**:32-43.

53. **PAML (version 3.13)** [http://abacus.gene.ucl.ac.uk/software/paml.html]

54. Hasegawa M, Kishino H, Yano T: **Dating of the human-ape splitting by a molecular clock of mitochondrial DNA.** *J Mol Evol* 1985, **22**:160-74.

55. Moriyama EN, Hartl DL: **Codon usage bias and base composition of nuclear genes in Drosophila.** *Genetics* 1993, **134**:847-858.

56. Moriyama EN, Powell JR: **Intraspecific nuclear DNA variation in Drosophila.** *Mol Biol Evol* 1996, **13**:261-277.

57. Comeron JM, Kreitman M: **The correlation between intron length and recombination in Drosophila. Dynamic equilibrium between mutational and selective forces.** *Genetics* 2000, **156**:1175-1190.

58. **ROSE (version 1.3)** [http://bibiserv.techfak.uni-bielefeld.de/rose/]

59. **PHYLIP (version 3.5c)** [http://evolution.genetics.washington.edu/phylip.html]

60. Zhu J, Liu JS, Lawrence CE: **Bayesian adaptive sequence alignment algorithms**. *Bioinformatics* 1998, **14**:25-39.

61. Tatusova TA, Madden TL: **BLAST 2 Sequences, a new tool for comparing protein and nucleotide sequences**. *FEMS Microbiol Lett* 1999, **174**:247-50.

62. Jareborg N, Birney E, Durbin R: **Comparative analysis of noncoding regions of 77 orthologous mouse and human gene pairs.** *Genome Res* 1999, **9**:815-24.

63. Delcher AL, Phillippy A, Carlton J, Salzberg SL: **Fast algorithms for large-scale genome alignment and comparison**. *Nucleic Acids Res* 2002, **30**:2478-83.

64. Ogurtsov AY, Roytberg MA, Shabalina SA, Kondrashov AS: **OWEN: aligning long collinear regions of genomes**. *Bioinformatics* 2002, **18**:1703-4.

65. Pearson WR, Lipman DJ: **Improved tools for biological sequence comparison**. *Proc Natl Acad Sci U S A* 1988, **85**:2444-8.

66. Needleman SB, Wunsch, C.D.: **A general method applicable to the search for similarities in the amino acid sequence of two proteins.** *J Mol Biol* 1970, **48**:443-53.

67. Schwartz S, Kent WJ, Smit A, Zhang Z, Baertsch R, Hardison RC, Haussler D, Miller W: **Human-mouse alignments with BLASTZ**. *Genome Res* 2003, **13**:103-7.

68. Morgenstern B: **DIALIGN 2: improvement of the segment-to-segment approach to multiple sequence alignment.** *Bioinformatics* 1999, **15**:211-8.

69. Thompson JD, Higgins DG, Gibson TJ: **CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence**

weighting, position-specific gap penalties and weight matrix choice. *Nucleic Acids Res* 1994, **22**:4673-80.

70. Rice P, Longden I, Bleasby A: **EMBOSS: the European Molecular Biology Open Software Suite**. *Trends Genet* 2000, **16**:276-7.

71. Kent WJ, Zahler AM: **Conservation, regulation, synteny, and introns in a large-scale C. briggsae-C. elegans genomic alignment.** *Genome Res* 2000, **10**:1115-25.

Figures

Figure 1 - Distribution of noncoding sequence lengths in the *D. melanogaster* Release 3 genome sequence.

Sequences between coding exons were extracted from the *D. melanogaster* Release 3 euchromatic genome sequence and annotations, and transposable element sequences were subsequently subtracted to produce the "pre-integration" distribution of noncoding sequence lengths (see Methods for details).

Figure 2 - Simulation statistics

Pairwise alignments were simulated for a range of divergence distances, using a modified version of the ROSE simulation platform under four different regimes: A) without indel evolution and without constrained blocks; B) with indel evolution and without constrained blocks; C) without indel evolution and with constrained blocks; and D) with indel evolution without constrained blocks. For each divergence distance, 1,000 replicates were used to calculate the mean and standard error for the following statistics: estimated overall divergence (black boxes), estimated divergence in constrained blocks of sites (grey circles), estimated divergence in unconstrained blocks of sites (brown triangles), identity (red crosses), identity in constrained blocks (yellow x's), identity in unconstrained blocks (green diamonds), fraction of ancestral sequence remaining in derived sequences (green triangle), and fraction of constraint (light blue checked boxes). Note that the divergence scale in this and following figures is discontinuous.

Figure 3 - Overall alignment coverage

For each divergence distance and each tool, 1,000 replicates were used to calculate the mean and standard error of overall alignment coverage, which was defined as the fraction of ungapped, orthologous pairs of sites in the simulated alignment that were included in an alignment produced by a tool (see Methods for details). A) overall coverage without constrained blocks and without insertion/deletion evolution; B) overall coverage without constrained blocks and with insertion/deletion evolution; C) overall coverage with constrained blocks and without insertion/deletion evolution; D) overall coverage with constrained blocks and with insertion/deletion evolution.

Figure 4 - Overall alignment sensitivity

For each divergence distance and each tool, 1,000 replicates were used to calculate the mean and standard error of overall alignment sensitivity, which was defined as the fraction of ungapped, orthologous pairs of sites in the simulated alignment that were aligned correctly in an alignment produced by a tool (see Methods for details). A) overall sensitivity without constrained blocks and without insertion/deletion evolution; B) overall sensitivity without constrained blocks and with insertion/deletion evolution; C) overall sensitivity with constrained blocks and without insertion/deletion evolution; D) overall sensitivity with constrained blocks and with insertion/deletion evolution.

Figure 5 - Constraint coverage and sensitivity

For each divergence distance and each tool, 1,000 replicates were used to calculate the mean and standard error of constraint coverage and constraint sensitivity, which were defined as the coverage and sensitivity within interspersed constrained blocks (see Methods for details). A) constraint coverage without insertion/deletion evolution; B) constraint coverage with insertion/deletion evolution; C) constraint sensitivity without insertion/deletion evolution; D) constraint sensitivity with insertion/deletion evolution.

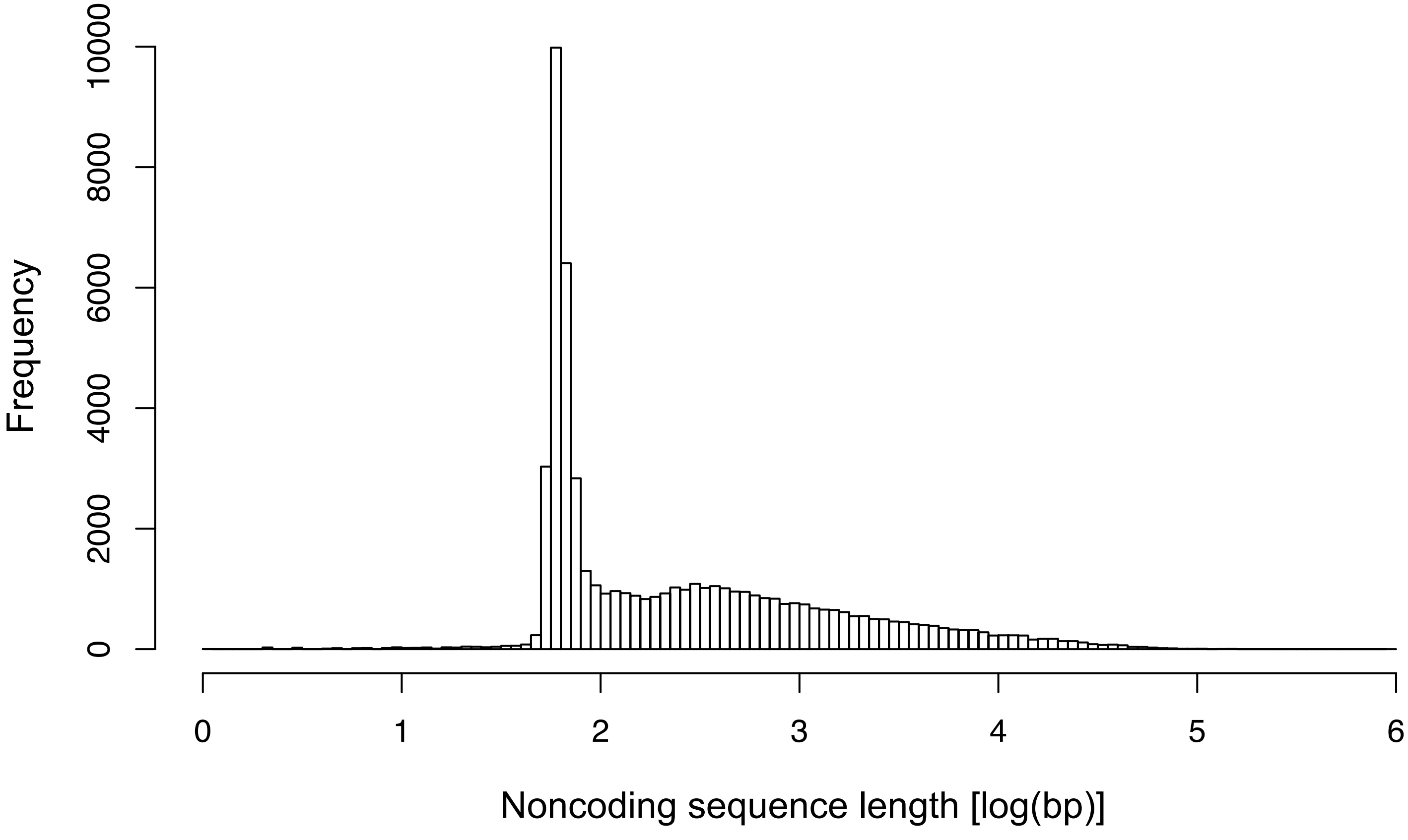
Figure 6 - Constraint specificity and local constraint sensitivity

For each divergence distance and each tool, 1,000 replicates were used to calculate a mean and standard error of constraint specificity and local constraint sensitivity. Constraint specificity was defined as the fraction of unconstrained sites in the simulated alignment that were unaligned or gapped in an alignment produced by a tool. Local constraint specificity was defined the constraint sensitivity for just the sites contained in an alignment produced by a tool (see Methods for details). A) constraint specificity without insertion/deletion evolution; B) constraint specificity with insertion/deletion evolution; C) local constraint sensitivity without insertion/deletion evolution; D) local constraint sensitivity with insertion/deletion evolution.

Tables

Table 1 – Summary of parameters used in simulations of noncoding sequence evolution.

Parameter	Value	Source	Refs.
Sequence length	10 Kb	*D. mel*	this work (Fig. 1)
AT : GC	60 : 40	*Drosophila spp.*	this work, [31, 55]
Transition / Transversion Bias	2	*Drosophila spp.*	[25, 56]
Substitution model	HKY85	-	[54]
Point substitutions : Indels	10 : 1	*Drosophila spp.*	[22, 23, 25]
Indel spectrum	-	*D.mel*	[57]
Median constrained block length	18 bp	*D.mel* vs. *D.vir*	[25]
Mean density of constrained blocks	0.2	*D.mel* vs. *D.vir*	[25]


Frequency
0
2000
4000
6000
8000
10000
0
1
2
3
4
5
6
Noncoding sequence length [log(bp)]

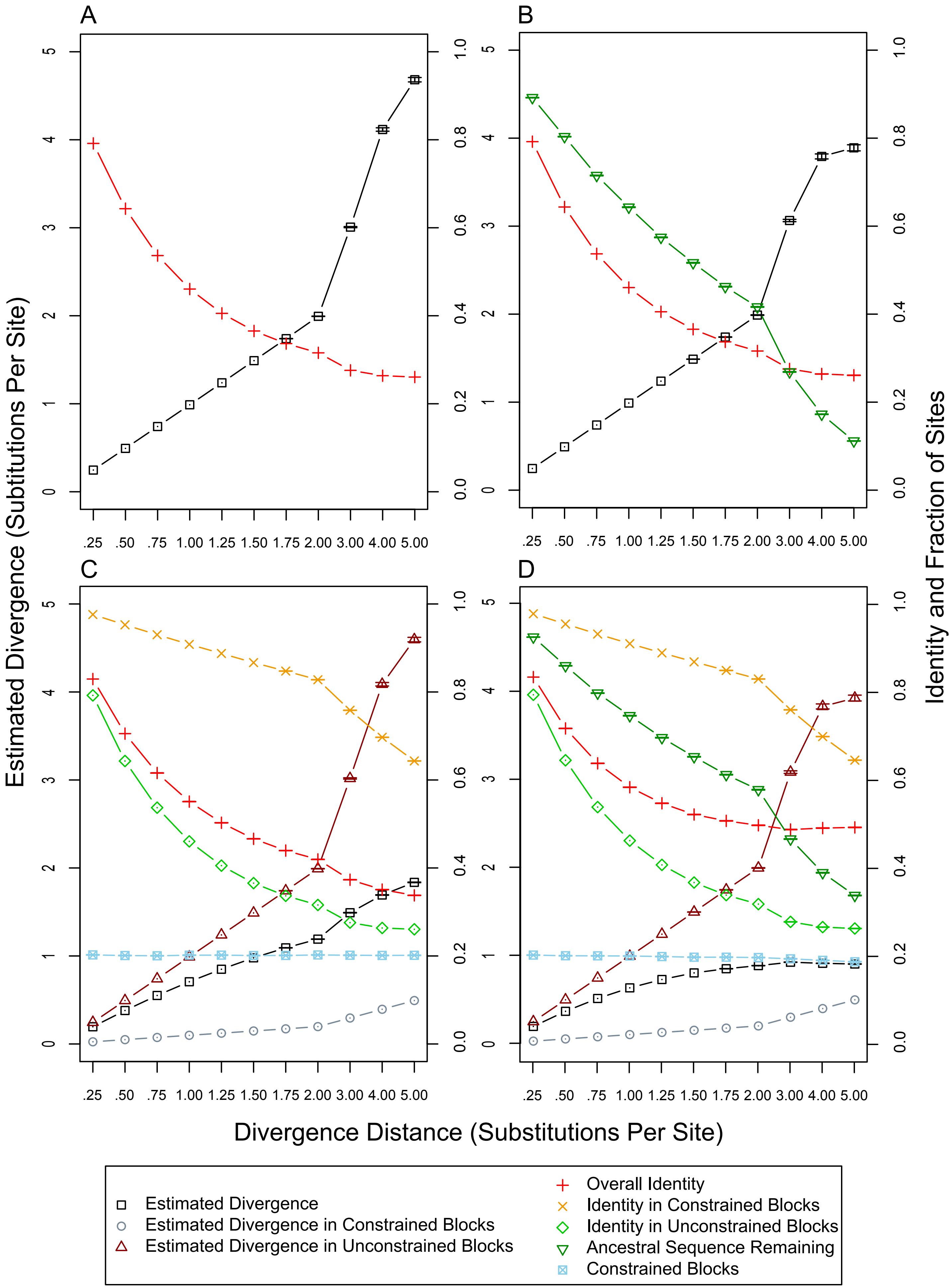

A
B
C
D
Estimated Divergence (Subtitutions Per Site)
Identity and Fraction of Sites
Divergence Distance (Substitutions Per Site)
0
1
2
3
4
5
0.0
0.2
0.4
0.6
0.8
1.0
.25
.50
.75
1.00
1.25
1.50
1.75
2.00
3.00
4.00
5.00
□ Estimated Divergence
○ Estimated Divergence in Constrained Blocks
△ Estimated Divergence in Unconstrained Blocks
+ Overall Identity
× Identity in Constrained Blocks
◇ Identity in Unconstrained Blocks
▽ Ancestral Sequence Remaining
⊠ Constrained Blocks

Figure 2

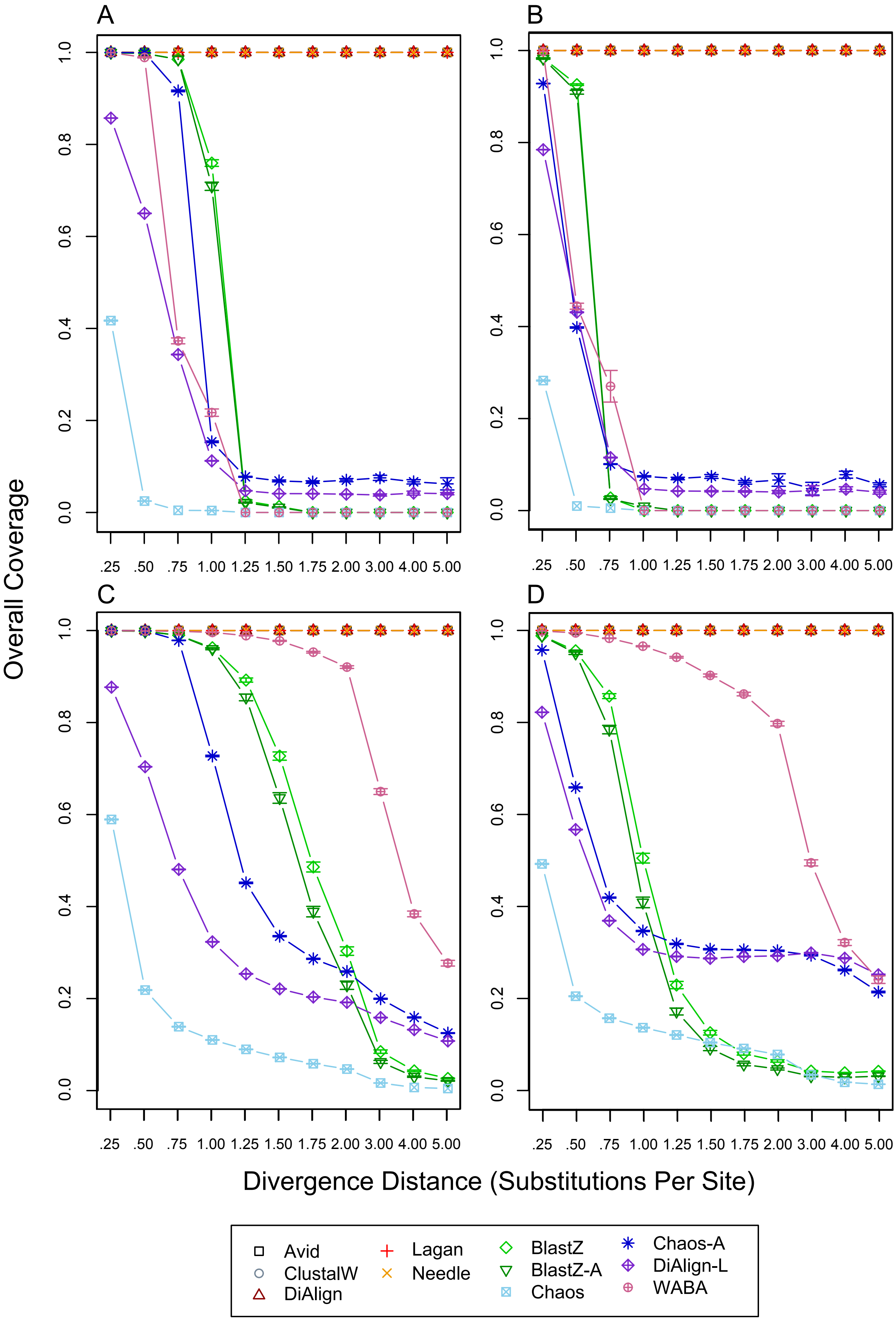

A
B
C
D
Overall Coverage
0.0
0.2
0.4
0.6
0.8
1.0
.25
.50
.75
1.00
1.25
1.50
1.75
2.00
3.00
4.00
5.00
Divergence Distance (Substitutions Per Site)
□ Avid
○ ClustalW
△ DiAlign
+ Lagan
× Needle
◇ BlastZ
▽ BlastZ-A
⊠ Chaos
✳ Chaos-A
◇ DiAlign-L
⊕ WABA

Figure 3

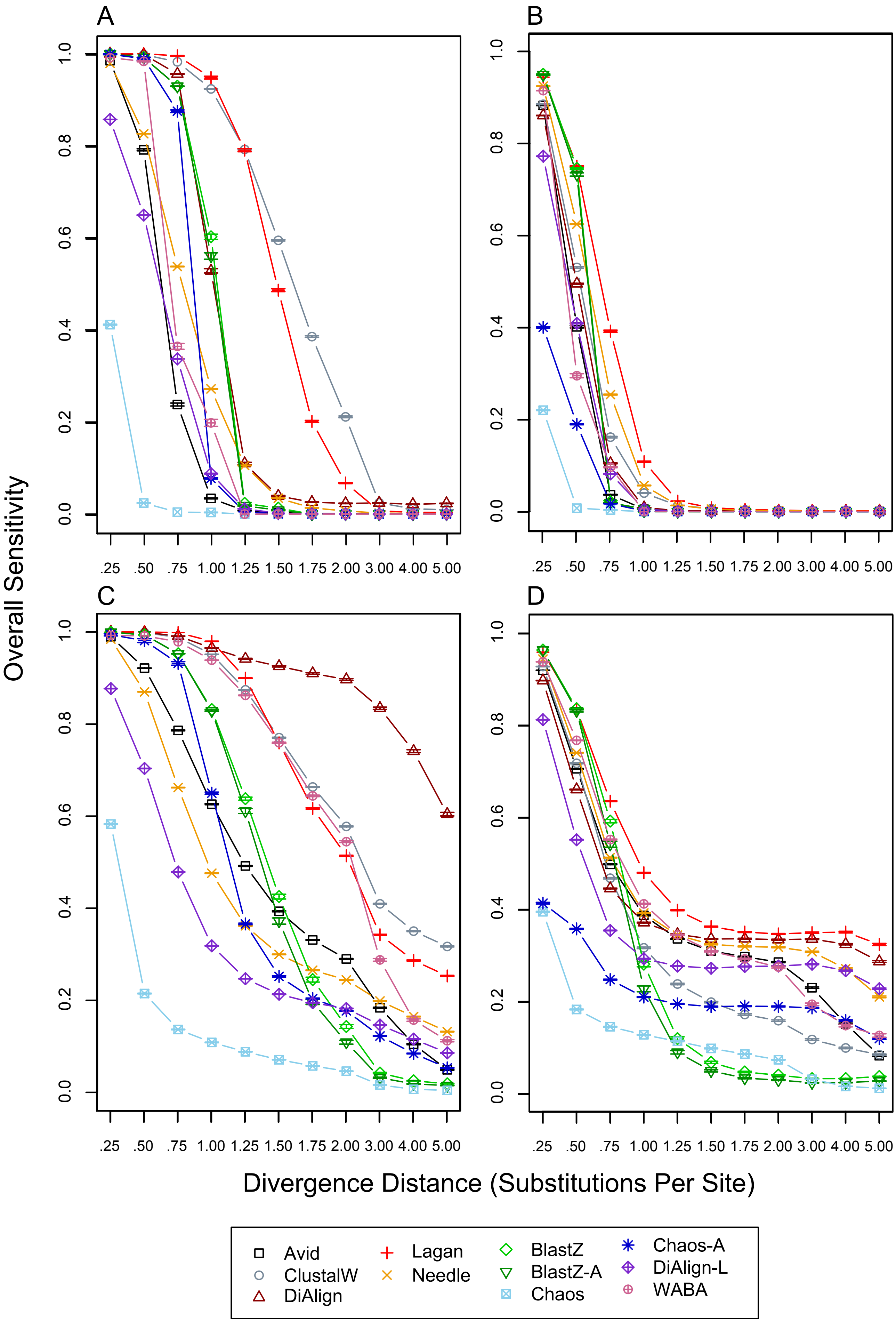

A
B
C
D
Overall Sensitivity
0.0
0.2
0.4
0.6
0.8
1.0
.25 .50 .75 1.00 1.25 1.50 1.75 2.00 3.00 4.00 5.00
Divergence Distance (Substitutions Per Site)
□ Avid
○ ClustalW
△ DiAlign
+ Lagan
× Needle
◇ BlastZ
▽ BlastZ-A
☒ Chaos
✳ Chaos-A
◇ DiAlign-L
⊕ WABA

Figure 4

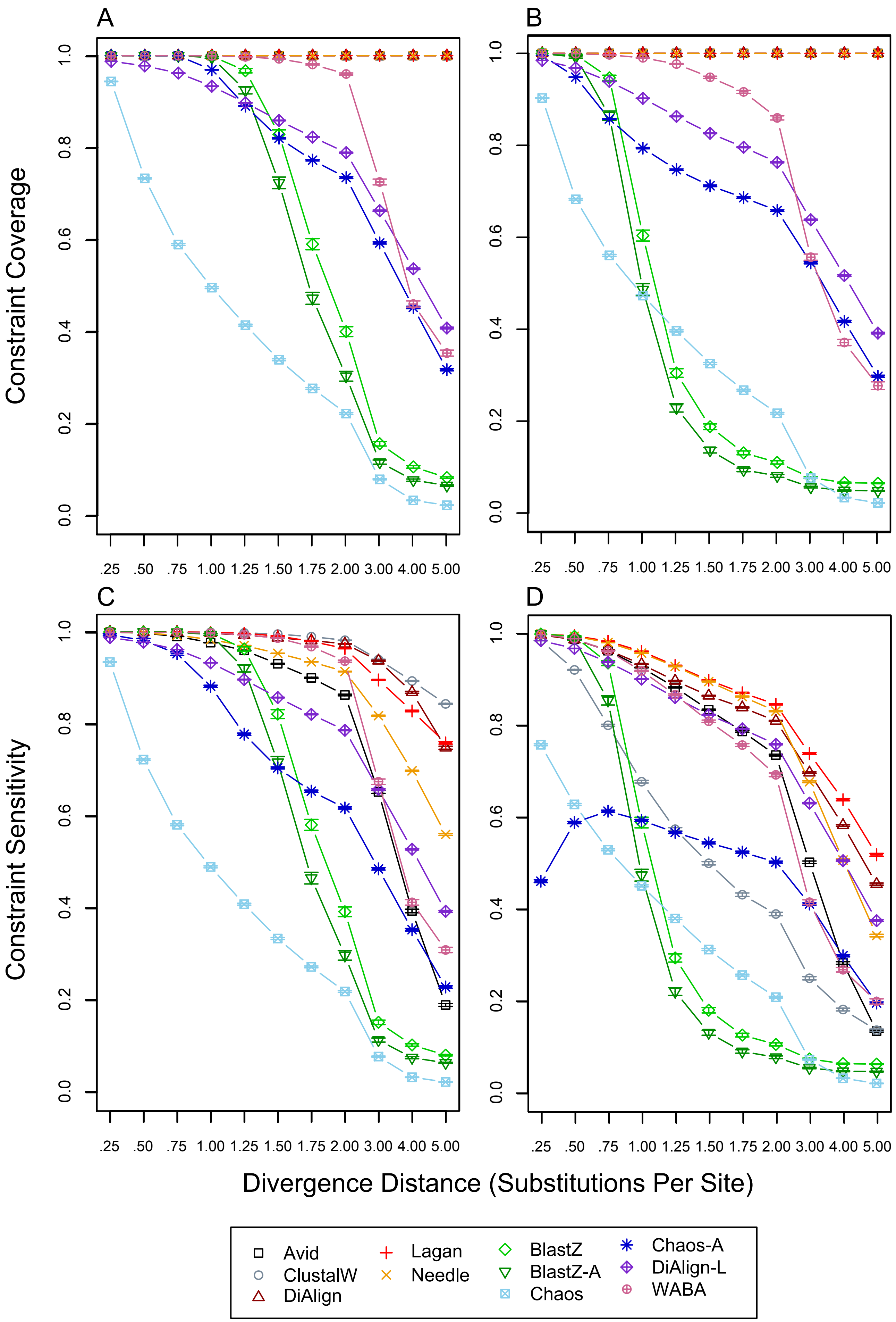

A
B
C
D
Constraint Coverage
Constraint Sensitivity
0.0
0.2
0.4
0.6
0.8
1.0
.25
.50
.75
1.00
1.25
1.50
1.75
2.00
3.00
4.00
5.00
Divergence Distance (Substitutions Per Site)
Avid
ClustalW
DiAlign
Lagan
Needle
BlastZ
BlastZ-A
Chaos
Chaos-A
DiAlign-L
WABA

Figure 5

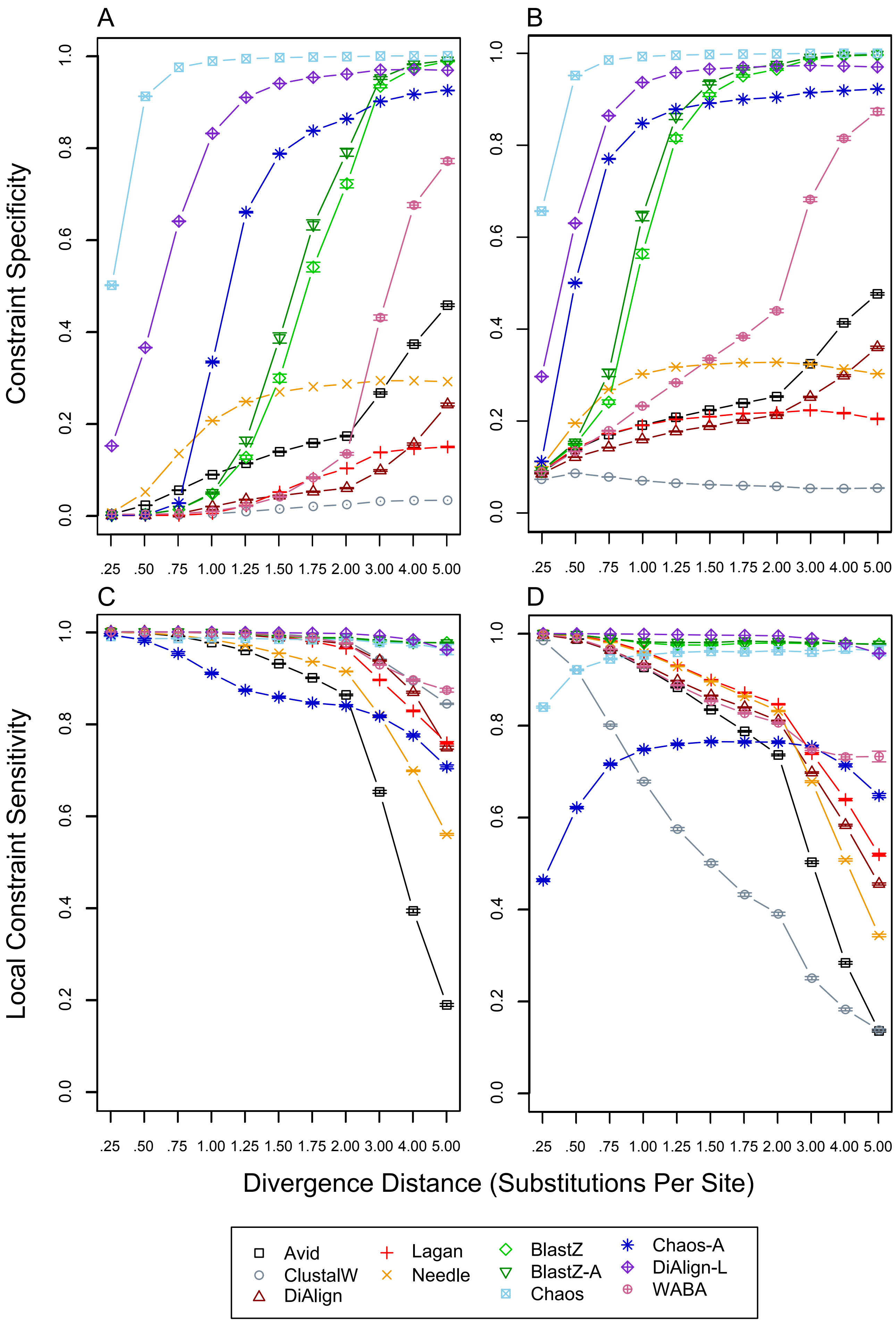

A
B
C
D
Constraint Specificity
Local Constraint Sensitivity
0.0
0.2
0.4
0.6
0.8
1.0
.25
.50
.75
1.00
1.25
1.50
1.75
2.00
3.00
4.00
5.00
Divergence Distance (Substitutions Per Site)
□ Avid
○ ClustalW
△ DiAlign
+ Lagan
× Needle
◇ BlastZ
▽ BlastZ-A
⊠ Chaos
✱ Chaos-A
◇ DiAlign-L
⊕ WABA

Figure 6