BIOINFORMATICS

Vol. 17 Suppl. 1 2001
Pages S140–S148

Integrating genomic homology into gene structure prediction

Ian Korf[1], Paul Flicek[2], Daniel Duan[1] and Michael R. Brent[1]

[1]Department of Computer Science, Washington University, Campus Box 1045, St. Louis, MO, 63130, USA and [2]Department of Biomedical Engineering, Washington University, Campus Box 1097, St. Louis, MO, 63130, USA

Received on February 6, 2001; revised and accepted on April 2, 2001

ABSTRACT

Twinscan is a new gene-structure prediction system that directly extends the probability model of Genscan, allowing it to exploit homology between two related genomes. Separate probability models are used for conservation in exons, introns, splice sites, and UTRs, reflecting the differences among their patterns of evolutionary conservation. Twinscan is specifically designed for the analysis of high-throughput genomic sequences containing an unknown number of genes. In experiments on high-throughput mouse sequences, using homologous sequences from the human genome, Twinscan shows notable improvement over Genscan in exon sensitivity and specificity and dramatic improvement in exact gene sensitivity and specificity. This improvement can be attributed entirely to modeling the patterns of evolutionary conservation in genomic sequence.

Contact: {ikorf, pflicek, duan, brent}@cs.wustl.edu

INTRODUCTION

A complete mapping from genome to proteome would constitute a foundation for genome-based biology. However, determining the structures of protein coding genes in eukaryotic genomic DNA is a difficult problem for which there are no reliable experimental or computational tools. The large volume of genomic sequence now available necessitates automated analysis methods; this need will become even more pronounced as the sequences of additional genomes become available.

Computational gene prediction has been an active area of research for over 20 years. The algorithms that have been developed are traditionally categorized as either *ab initio* or alignment-based. *Ab initio* methods, such as Genscan (Burge, 1997; Burge & Karlin, 1997) and Genie (Reese *et al.*, 2000b), make predictions using only the DNA sequence to be annotated and a model of gene structure. Alignment-based methods, such as Procrustes (Gelfand *et al.*, 1996) and GeneWise (E. Birney, unplublished), attempt to align homologous proteins to genomic sequence. This dichotomy of methods has broken down in recent years with the invention of hybrid approaches that integrate EST or protein similarities into *ab initio* methods (Hooper *et al.*, 2000; Reese *et al.*, 2000a; Yeh *et al.*, 2001).

The performance of new algorithms is often bench marked using the dataset and methods of Burset & Guigo (1996). This dataset consists of 570 short genomic sequences (average length 5074 bp) containing one complete multi-exon gene without alternatively spliced forms. Burset and Guigo employ several measures of gene prediction accuracy, including exact *exon sensitivity* — the fraction of true exons whose boundaries are predicted precisely. Most *ab initio* methods correctly identify about 50% of the real exons (Burset & Guigo, 1996). One of the most accurate is Genscan which is capable of achieving a respectable 78% exon sensitivity on the Burset-Guigo set (Burge & Karlin, 1997). The accuracy of alignment-based methods depends upon their protein databases; genes with similarities in the database may be predicted with high accuracy, while genes that are not similar to those in the database will not be predicted at all. Although hybrid methods have great promise because they can use both intrinsic and extrinsic information, they have shown only a moderate improvement over purely *ab initio* methods so far.

A new class of gene-prediction algorithms has recently been reported that leverages the power of comparative genomics. Alignments between genomic sequences from related species, particularly mouse and human, have shown that sequence similarity is a powerful approach for identifying genes and regulatory elements (Hardison *et al.*, 1997; Oeltjen *et al.*, 1997; Ansari-Lari *et al.*, 1998; Jang *et al.*, 1999). The two reported comparative genomics algorithms are Rosetta (Batzoglou *et al.*, 2000) and CEM (Bafna & Huson, 2000). These programs cannot be bench marked with the Burset-Guigo set because they require pairs of sequences, so the authors created their own gene sets. Like the Burset-Guigo set, the Rosetta and CEM sets also consist of short sequences containing exactly one gene.

Single-gene sequences do not represent a typical application for gene prediction programs. A common scenario is the analysis of high-throughput genomic (HTG) sequences, which are generally 100-200 Kb in length and contain an unknown number of genes. The gene prediction problem is much harder in these sequences than in single-gene sequences because gene boundaries are unknown, genes may be incomplete, and genes may lie on either strand. The performance of ROSETTA, CEM, and many other gene-prediction programs has not been evaluated on HTG sequences either because their algorithms assume the input sequence contains exactly one gene or because the authors chose not to evaluate them in this way. GENSCAN was designed for HTG sequences, but its performance drops from 43% exact gene sensitivity on the Burset-Guigo set to 15% on the HTG sequences in the experiments reported below (see Results).

In this paper, we report on TWINSCAN, a novel comparative-genomics-based gene-prediction system that has been designed for the analysis of HTG sequences. TWINSCAN is based on GENSCAN++, our C++ reimplementation of GENSCAN. TWINSCAN integrates cross-species similarity of HTG sequences into the probability model underlying GENSCAN. In the experiments reported below, TWINSCAN shows notable improvement over GENSCAN in exon sensitivity and specificity and dramatic improvement in exact gene sensitivity and specificity. Because TWINSCAN is a direct extension of GENSCAN, we can attribute this performance difference directly to modeling evolutionary conservation.

METHODS

Datasets

In order to train and test our algorithm, we needed annotated genomic sequences and their homologs. Ideally, we would have liked to use a set of collinear orthologous human and mouse sequences with experimentally verified genes. In addition, because we are focusing on high-throughput analysis, the sequences should be large and contain a mixture of complete and incomplete genes, single- and multi-exon genes, and genes on both strands. Jareborg *et al.* (1999) curated a set of orthologous genomic sequences, but only a few were long, and a detailed inspection revealed some annotation errors. Because no appropriate dataset was available, we developed two of our own.

In describing our data sets, we make use of the following terms.

Target sequence A genomic sequence to be annotated by a gene prediction program.

Informant sequence A genomic sequence from a related organism that is similar to the target sequence.

Operational ortholog The sequence from the informant genome that matches a given target sequence best. Match quality is measured by the sum of alignment scores in a BLAST search.

Top homologs One or more sequences from the informant genome that match a given target sequence best. For our experiments, we chose the four best matches.

Finished sequence Contiguous genomic sequence with an error rate of less than 1 in 10,000 bp.

Draft sequence A collection of typically 10-40 genomic sequence fragments of various sizes, in unknown order and orientation, produced by automated assembly of shotgun reads from BAC clones (100-200 Kb).

High Scoring Pair (HSP) A local alignment reported by BLAST.

Our first data set (Set 1) contains 68 mouse sequences and their top homologs from the human genome. We chose mouse target sequences because the human genome is effectively complete and therefore offers many possible informant sequences, including likely orthologs. Our original plan was to use the operational orthologs, but this simplistic one-to-one mapping means that large regions of the target and informant sequences may be unaligned. This can result from offsets in the BAC clones or from gene rearrangements that disrupt conserved syntenies. Visual inspection within ACEDB (R. Durbin and J. Thierry-Mieg unpublished, `www.acedb.org`) showed that top homologs frequently fill in the ends and holes left after alignment with the operational ortholog.

Set 1 was constructed by first querying GenBank release 121 for all mouse sequences over 30 Kb in length with annotated coding sequences (CDS). The resulting 86 sequences were filtered to remove those sequences with unsupported genes or those that omit real genes. Unsupported genes were identified as CDS annotations without any protein or EST evidence for any part of the predicted coding sequence. The challenge of finding unsupported genes is that the original CDS annotation shows up as a protein match, falsely verifying the gene. We therefore required at least two protein similarities for every CDS or, alternatively, transcript similarities. Omitted genes were identified as strong protein similarities (P-value $\leq$ 1e-5 and percent identity $\geq$ 50%) without any corresponding CDS annotation. The challenge in finding omitted genes is that pseudogenes can look like unannotated CDSs because they have high quality BLASTX alignments. We were able to identify pseudogenes by looking for stop codons and frameshifts in the BLASTX reports. Eighteen sequences with unsupported or omitted genes were removed, which left 68 sequences with plausible annotation. The 68 mouse sequences total 7.6 Mb with mean length 112 Kb and median length 98 Kb.

To recover the top homologs, we processed each sequence with RepeatMasker (A.F.A. Smit and P. Green, unpublished) and then performed a WU-BLASTN (W. Gish, unpublished, `blast.wustl.edu`) search against a database containing all human genomic sequences in GenBank release 121 (default parameters were used). The top four matches were kept as informant sequences. The danger in keeping too many top homologs is that one may introduce noise from spurious matches that are not truly homolgous. We chose the top four because of the possible two complete genome duplications in the vertebrate lineage (Meyer & Schartl, 1999). Draft sequences accounted for 53% of the top homologs.

Our second data set (Set 2) is a subset of Set 1 containing eight pairs of finished *operational orthologs*. The coding sequences were annotated by one of the authors (IK) using a typical sequence analysis pipeline involving protein, transcript, and genome similarities. Importantly, the sequences were annotated *in parallel*. The advantage of this is that the gene structures were checked against each other, and this helped to avoid mistakes. The disadvantage is that the annotator is a biased individual predisposed to believe in the importance of genomic conservation. However, the annotator believes that the gene structures are of the highest possible quality given the current information. Where the gene structures are different from their counterparts reported in the GenBank entry, they were corrected for good reasons. In one sequence AP001917 (gi:8953895), a potential sequencing error was predicted because it changed the parallel gene structures. This was reported to the sequencing center, the error was promptly verified, and the sequence has been updated (gi:10945234).

To ensure that GENSCAN, GENSCAN++, and TWINSCAN were on equal footing, and only predicted genes within the limits of conservation, the target sequences were edited by masking all sequence outside the region of conservation. For Set 1 this was 100 bp beyond the 5′-most and 3′-most HSPs. For Set 2 this was determined manually.

Conservation Sequence

We chose to model sequence similarity by a representation we call a *conservation sequence.* A conservation sequence pairs one of three symbols with each nucleotide of the target sequence:

.	unaligned
|	matched
:	mismatched

Gaps in the informant sequence become mismatch symbols; gaps in the target sequence are ignored. For example, consider the sequence:

```
123456789 position
GAATTCCGT target sequence
```

and suppose that BLAST yields the following HSP:

```
 345 6789  target position
 ATT-CCGT  target alignment
 ||  || |  BLAST match symbols
 ATCACC-T  informant alignment
```

Note that positions 1 and 2 of the target sequence are not aligned to anything in the informant sequence. The conservation sequence derived from this HSP is:

```
123456789 position
GAATTCCGT target sequence
..||:||:| conservation sequence
```

To create the conservation sequence, we first masked the target sequence with RepeatMasker using the `xsmall` option to report repetitive regions in lowercase rather than converting them to N's. Next, we aligned the target sequence to the informant sequences with WU-BLASTN (parameters: `W=8 M=1 N=-1 Q=5 R=1 X=20 S=15 S2=15 gapS2=30 lcmask wordmask=dust wordmask=seg topcomboN=3`). The `lcmask` option prevents alignments from being seeded in lowercase regions but does not prevent externally-seeded alignments from extending into these regions. Lowercase masking, also known as *soft-masking* is important because it prevents many false-positive alignments without prohibiting alignments in low-complexity or repeat-like regions, which are sometimes present in coding sequences.

The conservation sequence was constructed by the following procedure, which makes use of the HSPs from the top four homologs (`HSPs`) and a target sequence (`Targ`):

```
MakeConservationSequence
ConSeq[1...n] = unaligned
Sort HSPs by alignment score
for i = 1 to Targ.Length
  for each HSP H from best to worst
    if H extends to position i
      if ConSeq[i] == unaligned
        ConSeq[i] = H[i]
```

Note that the conservation symbol assigned to the target nucleotide in position `i` is determined by the best individual HSP to overlap position `i`, regardless of which homologous sequence it comes from. Position `i` is classified as unaligned only if none of the HSPs overlap it.

```
         10        20        30
123456789|123456789|123456789|123456789
ATTTAGCCTACTGAAATGGACCGCTTCAGCATGGTATCC
||:|||.........|:|:|||||||||:||:|||::||
```

Fig. 1. An example DNA sequence together with the corresponding conservation sequence.

Genscan, Genscan++, and Twinscan

We began by reimplementing Genscan, as specified in Burge (1997). After incorporating most of the minor differences between the published version and the distributed executable, both the predictions and the accuracy of the our reimplementation are extremely close to those of the distributed GENSCAN executable (see Results). Since the new implementation is in C++, we refer to it as Genscan++.

Genscan assigns each nucleotide of an input sequence to one of seven general categories: promoter, 5′ UTR, exon, intron, 3′ UTR, poly-adenylation (poly-A) signal, and intergenic. Genscan chooses the most likely assignment of categories to nucleotides according to a probabilistic model of gene structure, called the *Genscan model* hereafter. Let us call any DNA sequence together with a categorization of all its nucleotides a *parsed sequence*. The Genscan model assigns a probability to every possible parsed sequence. The Genscan system consists of the model together with an *optimization algorithm*. Given a DNA sequence, the optimization algorithm evaluates parses of that sequence in order to find the one with greatest probability, according to the model.

We have developed a new model that assigns probability to any parsed DNA sequence together with a parallel conservation sequence. Under our model, the probability of a DNA sequence and the probability of the parallel conservation sequence are independent, given a parse. The probability of the DNA sequence, given the parse, is the same as under the Genscan model. The probability of the conservation sequence, given the parse, is computed according to the conservation model described below. For example, consider the pair consisting of the target sequence and conservation sequence shown in Figure 1. In particular, consider the probability of observing the target sequence and the conservation sequence extending from position 7 to position 33, given that there is an an internal exon extending from position 7 to position 33. This is simply the probability of the target sequence under the Genscan model times the probability of the conservation sequence under the conservation model, given an exon extending from 7 to 33:

$$\mathrm{Pr}(\mathrm{T}_{7,33}, \mathrm{C}_{7,33}|\mathrm{E}_{7,33}) = \mathrm{Pr}(\mathrm{T}_{7,33}|\mathrm{E}_{7,33})\,\mathrm{Pr}(\mathrm{C}_{7,33}|\mathrm{E}_{7,33})$$

where $\mathrm{T}_{7,33}$ is the target DNA sequence from position 7 to position 33, inclusive, $\mathrm{C}_{7,33}$ is the corresponding conservation sequence, and $\mathrm{E}_{7,33}$ is the hypothesis that an exon begins at position 7 and ends at position 33.

Twinscan consists of the new, joint probability model on DNA sequences and conservation sequences, together with the same optimization algorithm used by Genscan.

Probability models The Genscan model is based on an *explicit state duration Hidden Markov Model* (HMM). Each state of the HMM corresponds to one of the seven categories with which all nucleotides are ultimately labeled — promoter, 5′ UTR, exon, intron, etc. (see Fig. 2). The model can be divided into three components: the *transition model*, which specifies the probability of moving from any one state to any other state, the *duration model*, which specifies the probability of staying in a given state for a given number of nucleotides before changing to another state, and the *state-specific sequence models*, which specify the probability of any given nucleotide sequence being generated from any given state. For details of these models, see Burge (1997).

Twinscan augments the state-specific sequence models of Genscan with models of the probability of generating any given conservation sequence from any given state. Thus, Twinscan's state-specific models specify joint probability distributions on DNA sequence and conservation sequence. Coding, UTR, and intron/intergenic states all assign probability to stretches of conservation sequence using homogeneous 5th-order Markov chains. One set of parameters is estimated for the coding regions (excluding translation initiation and termination signals), one for the translation initiation and termination signals, one for the UTR states, and one for the intron and intergenic states. In Figure 1, for example, the probability of observing the conservation sequence from position 7 to position 33, given that an exon spans positions 7 to 33, is:

$$\begin{aligned}&\mathrm{Pr}_{\mathrm{C}}(C_{7,33} \mid E_{7,33}) = \\ &\qquad \mathrm{Pr}_{\mathrm{E}}(C_{7,7} \mid C_{2,6}) \cdot \ldots \cdot \mathrm{Pr}_{\mathrm{E}}(C_{33,33} \mid C_{28,32}),\end{aligned}$$

where $\mathrm{Pr}_{\mathrm{E}}(\mathrm{C}_{33,33} \mid \mathrm{C}_{28,32})$, for example, is the estimated probability of a | (match) following the five context symbols |:||: in the conservation sequence of an exon. (Note that the symbol | used in conservation sequence is not related to the use of the same symbol to indicate conditioning of a probability distribution.) On the other hand, the probability of the same conservation sequence given that positions 7 to 33 are part of a UTR is computed according to a different 5th-order Markov chain:

$$\begin{aligned}&\mathrm{Pr}_{\mathrm{C}}(\mathrm{C}_{7,33} \mid \mathrm{U}_{7,33}) = \\ &\qquad \mathrm{Pr}_{\mathrm{U}}(\mathrm{C}_{7,7} \mid \mathrm{C}_{2,6}) \cdot \ldots \cdot \mathrm{Pr}_{\mathrm{U}}(\mathrm{C}_{33,33} \mid \mathrm{C}_{28,32}),\end{aligned}$$

where $\mathrm{Pr}_{\mathrm{U}}(\mathrm{C}_{33,33} \mid \mathrm{C}_{28,32})$, for example, is the estimated probability of a | following the five symbols |:||: in the

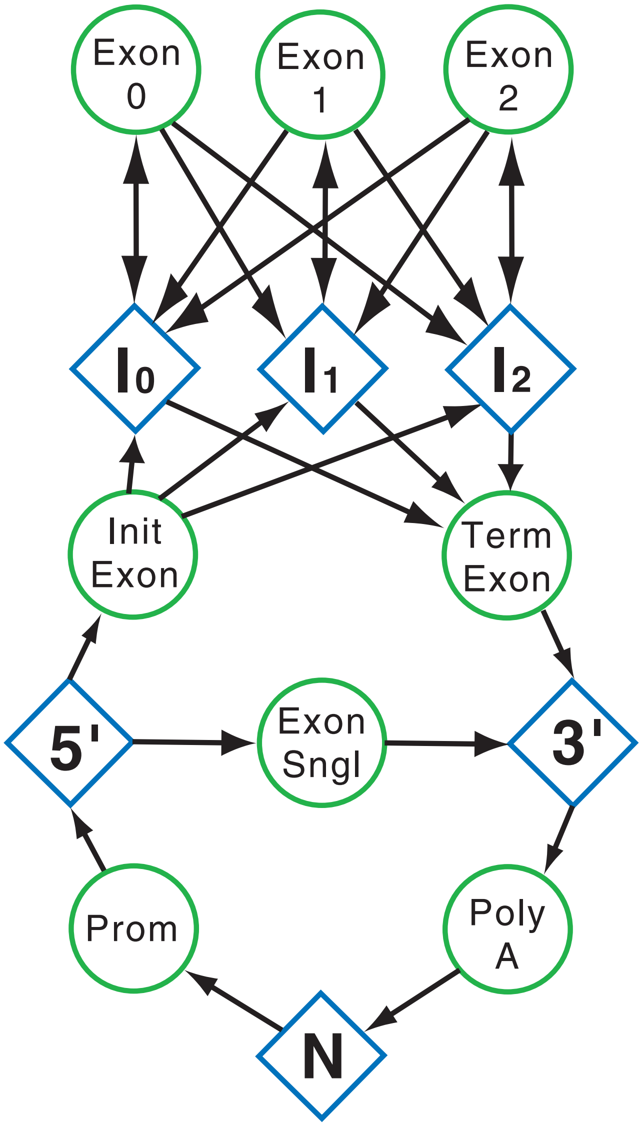

Fig. 2. The states of the GENSCAN model. Arrows indicate transitions with nonzero probability for genes on the forward strand. States Exon 0, Exon 1, and Exon 2 represent internal exons with different reading frames; states I0, I1, and I2 represent introns following exons with different reading frames; 5′ represents the region upstream of the first coding exon of a gene; 3′ represents the region downstream of the last coding exon of a gene; Prom represents the promoter region, and PolyA represents the polyadenylation signal. An analogous model is used for genes on the opposite strand.

conservation sequence of a UTR. Models of conservation sequence at splice donor and acceptor sites were based on 2nd-order Weight Array Matrices (WAMs: Salzberg, 1997; Zhang and Marr, 1993). These models consist of separate 2nd-order Markov chains for each position in a fixed-sized window. Following GENSCAN, the donor site window was fixed at 9 bp and the acceptor site at 43 bp.

The parameters of the conservation models must be estimated from annotated training sequences. The experiments we report were performed using an eight-fold cross-validation. Specifically, all sequences in each dataset were divided into eight groups by their accession numbers. Eight experiments were performed. In each experiment, a different 1/8 of the sequences was used for performance evaluation while the remaining 7/8 were used for parameter estimation. The results of these eight experiments were combined in the results reported below. Parameter estimates were smoothed by adding one to the counts for each symbol in each context. Because it is difficult to determine the ends of the UTRs, the parameters of the conservation model for UTRs were estimated from 100 nucleotides upstream of the initial ATG and 100 nucleotides downstream of the stop codon.

Optimization algorithm Given a particular HMM, the problem of figuring out the state sequence that is most likely to generate an observed output sequence is known as *decoding*. The Viterbi algorithm is an efficient (worst-case linear time) decoding algorithm for standard HMMs. However, the GENSCAN model is an explicit state-duration HMM, meaning that each state specifies the probability of staying in that state for d consecutive nucleotides. In a standard HMM, the duration distributions all decay exponentially. In general, the decoding problem for explicit state-duration HMMs takes time proportional to the cube of the input length, which is not acceptable for most genomic applications. However, all states in the GENSCAN model except the exon states do use either exponential duration distributions (like in a standard HMM) or fixed, constant durations. Burge was able to exploit this, along with other special properties of the GENSCAN model, to derive an algorithm whose running time grows as the square of the input length in the theoretical worst case. In practice, GENSCAN's running time, and hence TWINSCAN's, grows only linearly for genomic sequences longer than a few kilobases. The main reason is that the number of potential exons (spliceable open reading frames) grows only linearly in genomic sequences longer than a few kilobases.

Performance Evaluation

We used three categories for comparing exons: true positives (TP) are exons where the prediction matches the annotation exactly, false positives (FP) are predicted exons that do not match the annotation exactly, and false negatives (FN) are annotated exons that are not predicted exactly. We used the same categories for comparing predicted genes to annotated genes: true positives are predicted genes that exactly match annotated genes, false pos-

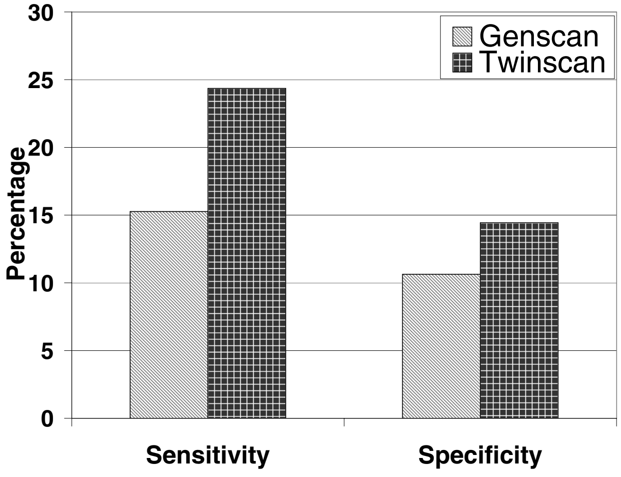

Fig. 3. Exact Gene Accuracy in Set 1.

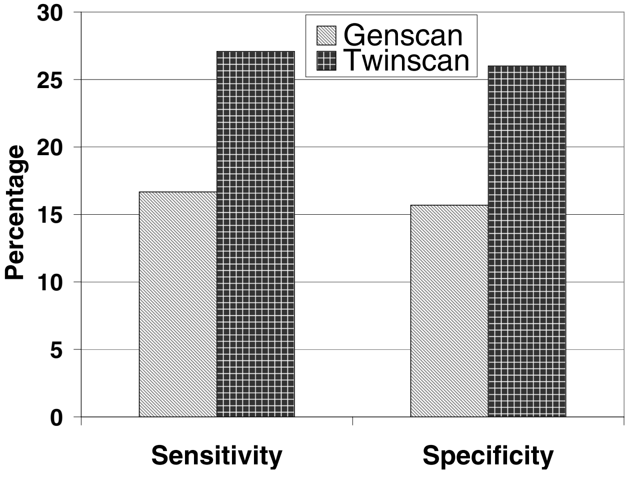

Fig. 4. Exact Gene Accuracy in Set 2.

itives are predicted genes that do not exactly match annotated genes, and false negatives are annotated genes that are not predicted exactly. We assessed the performance of GENSCAN, GENSCAN++, and TWINSCAN by measuring their sensitivity (SN) and specificity (SP) at the exon and gene levels. SN and SP are defined as follows:

$$\mathrm{SN} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}$$

$$\mathrm{SP} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}}$$

RESULTS

The results of our comparison of GENSCAN, GENSCAN++, and TWINSCAN are shown in Table 1 and Table 2. The tables show that the performance of GENSCAN++ is very similar to that of the original GENSCAN. GENSCAN++ tends to find more exons and more genes, but its sensitivity and specificity at both the exon and gene levels are nearly identical. From our manual inspection, it appears that GENSCAN++ finds additional exons in long introns. We suspect that this difference is due to an undocumented feature of GENSCAN that boosts the probabilities of long introns and intergenic regions.

Table 1 shows that TWINSCAN outperforms GENSCAN by all measures in Set 1. TWINSCAN achieves 68% sensitivity at the exon level compared to 63% for GENSCAN. The difference in exon specificity is even greater: 66% vs. 58%. TWINSCAN also performs better in comparisons at the nucleotide level (data not shown). Most importantly, TWINSCAN achieves 24% sensitivity in exact gene prediction, as compared to 15% for GENSCAN (Figure 3). For specificity in exact gene prediction TWINSCAN achieves 14.4%, as compared to 10.6% for GENSCAN. A chi-squared analysis reveals that TWINSCAN predicts a reliably greater fraction of the annotated genes exactly ($\chi^2(1) = 7.15$, $p = .007$). Conversely, a greater fraction of the genes predicted by TWINSCAN are exact matches to annotated genes, but this difference does not reach conventional levels of significance ($\chi^2(1) = 2.79$, $p = .095$).

The coding sequence annotations for the 68 sequences in Set 1 were taken to be correct for purposes of this experiment. However, it is likely that these annotations contain some errors. Compiling a set of reliable annotations for HTG sequences is a difficult problem, especially when pairs of sequences are required. However, such a set is essential to properly train and test gene prediction algorithms. To remedy this, we evaluated the performance of TWINSCAN on sequences with high quality annotation.

The performance of both TWINSCAN and GENSCAN on Set 2 is better than on Set 1. As in Set 1, TWINSCAN outperforms GENSCAN by all measures. Perhaps the most impressive result is the relative difference between TWINSCAN and GENSCAN at the gene level (See Figure 4). TWINSCAN outperforms GENSCAN by 62% for sensitivity and 66% for specificity. Overall, however, the results from Set 2 are consistent with the results from Set 1.

An example of how TWINSCAN predictions differ from GENSCAN predictions is depicted in Figure 5. The 5′-exon of the gene shown in this region is predicted correctly by TWINSCAN. GENSCAN misses the proper structure at the 5′ end, finding a different initial exon and an internal exon instead of the short initial exon in the annotated structure. TWINSCAN chooses the correct structure, in part, because the pattern of conservation in the alignments near the correct initial exon increases the score of that

Table 1. Gene Prediction Performance on Set 1.

Program	Exons	Exon SN	Exon SP	Genes	Gene SN	Gene SP
Annotation	2758			275		
GENSCAN	2997	0.631	0.581	395	0.153	0.106
GENSCAN++	3024	0.628	0.572	413	0.156	0.104
TWINSCAN	2854	0.683	0.660	464	0.244	0.144

Table 2. Gene Prediction Performance on Set 2.

Program	Exons	Exon SN	Exon SP	Genes	Gene SN	Gene SP
Annotation	610			48		
GENSCAN	731	0.798	0.666	51	0.167	0.157
GENSCAN++	734	0.795	0.661	53	0.167	0.151
TWINSCAN	684	0.843	0.752	50	0.271	0.260

exon. This decision is also influenced by the lack of conservation near the initial exon chosen by GENSCAN, and by the fact that the conservation sequence near the second exon chosen by GENSCAN contains many gaps and/or mismatches, pulling down its score. Moving to exon 3 of the annotated structure, GENSCAN choses an incorrect splice donor site (see magnified image at right of Figure 5). TWINSCAN chooses the correct splice site because the alignment ends before the end of the exon chosen by GENSCAN, pulling down the conservation score of the exon chosen by GENSCAN below that of the correct exon.

DISCUSSION

Our experiments demonstrate that integrating genomic similarity into the GENSCAN algorithm significantly improves its accuracy, especially in the important task of complete gene prediction. The relative improvement in exact gene sensitivity is about 60%. Although this is impressive, TWINSCAN only predicts about one quarter of the genes correctly. This underscores the fact that gene prediction is a difficult problem, especially in realistic data sets. To make TWINSCAN more accurate, we plan to use more expressive measures of conservation. We also plan to extend the system to make use of additional information sources, such as transcript and protein similarities.

Although our comparisons focused on GENSCAN, a number of other systems are under active development. Recently, the Genome Annotation Assessment Project (GASP) compared a number of gene prediction systems using a 2.9-Mb sequence contig from the *Adh* region of the fruit fly *Drosophila melanogaster* (Reese *et al.*, 2000a). The systems compared included FGENES (Salamov & Solovyev, 2000), GENEID (Parra *et al.*, 2000), GENIE (Kulp *et al.*, 1996; Reese *et al.*, 2000b), HMMGENE (Krogh, 1997, 2000), MAGPIE EXON (Gaasterland *et al.*, 2000), and GRAIL (Xu *et al.*, 1997). GENSCAN was not tested. Instead, it was used in defining one of the gold standards of correct annotation against which other systems were evaluated, suggesting that it is still widely considered to define the state of the art. The best programs were reported to have exact-gene sensitivity of about 40% and specificity of about 30%. However, these numbers cannot be compared to the GENSCAN and TWINSCAN results reported above. These two numbers were computed by comparison to two very different standards of correct annotation for the test sequence, one designed to yield an upper bound on the sensitivity and the other designed to yield an upper bound on the specificity. Our sensitivity and specificity, on the other hand, were computed against the same standard. In addition, the accuracy of gene-structure prediction systems on the *Drosophila* genome may differ from their accuracy on vertebrate genomes.

An important difference between TWINSCAN and the previously published comparative-genomics methods, ROSETTA and CEM, is that TWINSCAN does not attempt to globally align the exons of two orthologous sequences. Global alignment requires that the two sequences have the same exon-intron structure. Extending this global alignment strategy to multi-gene sequences would require the assumption that the two sequences have the same genes in the same order and orientation. We have observed several cases in our data sets where this is not true. Similarly, in a large-scale comparison between the nematodes *Caenorhabditis briggsae* and *Caenorhabditis elegans* Kent & Zahler (2000) found that rearrangements are common, occurring on average every 8.5 Kb.

Using a global alignment strategy requires informant

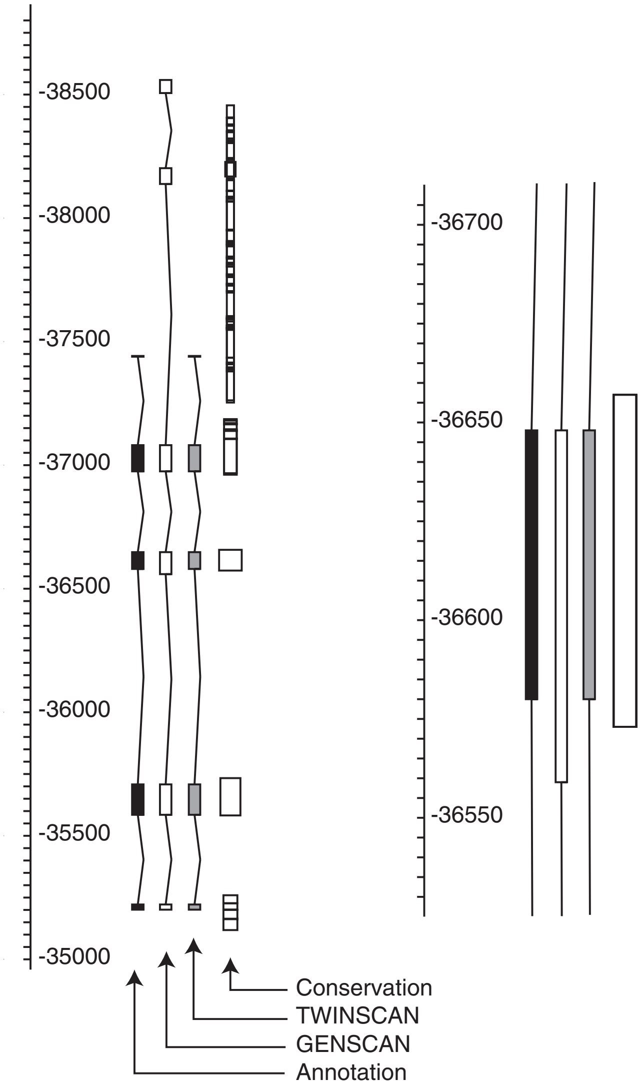

Fig. 5. Detailed view of the annotation, gene predictions and conservation at the L44L gene (AAB47245.1) from the *Mus musculus* Bruton's tyrosine kinase locus (U58105.1). The magnification at right shows the region around exon 3. The width of boxes representing BLAST alignments corresponds to the quality of the alignment. The image comes from ACEDB.

sequences to be finished. Our conservation sequence approach, which is based on the highest scoring local alignments, allows one to use draft and shotgun sequences. The conservation sequence effectively rearranges the alignments into the correct order and orientation. In addition, because the HSPs are sorted by score and conservation symbols are not overwritten, orthologies can be represented in favor of paralogies and spurious similarities. In our experiments, more than half of the top homologs were, in fact, draft sequences. Even whole-genome shotgun sequence can be used to create conservation sequence—one merely adjusts the parameters for defining the top homologs. Experiments to determine the proper parameters are currently underway.

TWINSCAN's main limitation is that the target sequence must have appropriate informant sequences. With TWINSCAN's ability to utilize unfinished informant sequences, and with the current rate of genome sequencing, this will become a minor restriction.

More information, including the datasets, is available online at `genes.cs.wustl.edu`.

ACKNOWLEDGMENTS

We wish to thank Chris Burge for answering email queries about GENSCAN and offering to let us see the source code (although we declined this generous offer). Thanks also to Matthew Snover for help in organizing and reimplementing part of our code base at critical moments. Thanks to Tamara Doering for suggesting the name TWINSCAN. This work was supported, in part, by NHGRI grant K22 HG00064-01 (IK), a Whitaker Foundation graduate fellowship in Biomedical Engineering (PF), National Science Foundation grant DBI-0091270 (MB), and by Washington University in St. Louis.

REFERENCES

Ansari-Lari, M. A., Schaffer, A. A., Zhang, J., Zhang, Z., Miller, W. & Lipman, D. J. (1998). Comparative sequence anlaysis of a gene-rich cluster at human chromosome 12p13 and its syntenic region in mouse chromosome 6. *Genome Research*, **8**, 29–40.

Bafna, V. & Huson, D. H. (2000). The conserved exon method for gene finding. In *Proceedings from the Eigth International Conference on Intelligent Systems for Molecular Biology*. AAAI Press, pp. 3–12.

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

Burge, C. (1997). Identification of genes in human genomic DNA. Ph.D. thesis, Stanford University.

Burge, C. & Karlin, S. (1997). Prediction of complete gene structures in genomic DNA. *Journal of Molecular Biology*, **268**, 78–94.

Burset, M. & Guigo, R. (1996). Evaluation of gene structure prediction programs. *Genomics*, **34**, 353–367.

Gaasterland, T., Sczyrba, A., Thomas, E., Aytekin-Kurban, G., Gordon, P. & Sensen, C. W. (2000). MAGPIE/EGRET annotation of the 2.9-mb *Drosophila melanogaster Adh* region. *Genome Research*, **10**, 502–510.

Gelfand, S., Mironov, A. A. & Pevzner, P. A. (1996). Gene recognition via spliced sequence alignment. *PNAS USA*, **93**, 9061–9066.

Hardison, R. C., Oeltjen, J. & Miller, W. (1997). Long human-mouse sequence alignments reveal novel regulatory elments: a reason to sequence the mouse genome. *Genome Research*, **7**, 959–966.

Hooper, P. M., Zhang, H. & Wishart, D. S. (2000). Prediction of genetic structure in eukaryotic DNA using reference point logistic regression and sequence alignment. *Bioinformatics*, **16**, 425–438.

Jang, W., Hua, A., Spilson, S. V., Miller, W., Roe, B. A. & Meisler, M. H. (1999). Comparative sequence of human and mouse BAC clones from the *mnd* region of chromosome 2p13. *Genome Research*, **9**, 53–61.

Jareborg, N., Birney, E. & Durbin, R. (1999). Comparative analysis of noncoding regions of 77 orthologous mouse and human gene pairs. *Genome Research*, **9**, 815–824.

Kent, W. J. & Zahler, A. M. (2000). Conservation, regulation, synteny, and introns in a large-scale *C. elegans–C. briggsae* genomic alignment. *Genome Research*, **10**, 1115–1125.

Krogh, A. (1997). Two methods for improving performance of an HMM and their application for gene finding. In *Proceedings of the Fifth International Conference on Intelligent Systems for Molecular Biology*. AAAI Press, pp. 179–186.

Krogh, A. (2000). Using database matches with HMMGene for automated gene detection in *Drosophila*. *Genome Research*, **10**, 523–528.

Kulp, D., Haussler, D., Reese, M. G. & Eeckman, F. H. (1996). A generalized hidden markov model for the recognition of human genes in DNA. In *Proceedings of the Fourth International Conference on Intelligent Systems for Molecular Biology*. AAAI Press, pp. 134–142.

Meyer, A. & Schartl, M. (1999). Gene and genome duplications in vetebrates: the one-to-four (-to-eight in fish) rule and the evolution of novel gene functions. *Current Opinions in Cell Biology*, **11**, 699–704.

Oeltjen, J. C., Malley, T. M., Muzny, D. M., Miller, W., Gibbs, R. A. & Belmont, J. W. (1997). Large-scale comparative sequence analysis of the human and murine Bruton's tyrosine kinase loci reveals conserved regulatory domains. *Genome Research*, **7**, 315–329.

Parra, G., Blanco, E. & Guigo, R. (2000). GeneID in *Drosphila*. *Genome Research*, **10**, 511–515.

Reese, M., Hartzell, G., Harris, N., Ohler, U., Abril, J. & Lewis, S. (2000a). Genome annotation assessment in *Drosphila melanogaster*. *Genome Research*, **10**, 483–501.

Reese, M., Kulp, D., Tammana, H. & Haussler, D. (2000b). Genie–gene finding in *Drosophila melanogaster*. *Genome Research*, **10**, 529–538.

Salamov, A. A. & Solovyev, V. V. (2000). Ab initio gene finding in *Drosophila* genomic DNA. *Genome Research*, **10**, 516–522.

Salzberg, S. (1997). A method for identifying splice sites and translational start sites in eukaryotic mRNA. *Computer Applications in the Biosciences*, **13**, 365–376.

Xu, Y., Mural, R. J. & Uberbacher, E. C. (1997). Inferring gene structures in genomic sequences using pattern recognition and expressed sequence tags. In *Proceedings of the Fifth International Conference on Intelligent Systems for Molecular Biology*. AAAI Press, pp. 344–353.

Yeh, R. F., Lim, L. P. & Burge, C. B. (2001). Computational inference of homologous gene structures in the human genome. *Genome Research*, **11**, 803–816.

Zhang, M. & Marr, T. (1993). A weight array method for splicing signal analysis. *Computer Applications in the Biosciences*, **9**, 499–509.