GENOMICS 34, 353–367 (1996)
ARTICLE NO. 0298

Evaluation of Gene Structure Prediction Programs

MOISÈS BURSET AND RODERIC GUIGÓ[1]

Departament d'Informàtica Mèdica, Institut Municipal d'Investigació Mèdica (IMIM), E-08003 Barcelona, Spain; and Departament D'Estadística, Facultat de Biologia, Universitat de Barcelona, Barcelona, Spain

Received September 18, 1995; accepted February 29, 1996

We evaluate a number of computer programs designed to predict the structure of protein coding genes in genomic DNA sequences. Computational gene identification is set to play an increasingly important role in the development of the genome projects, as emphasis turns from mapping to large-scale sequencing. The evaluation presented here serves both to assess the current status of the problem and to identify the most promising approaches to ensure further progress. The programs analyzed were uniformly tested on a large set of vertebrate sequences with simple gene structure, and several measures of predictive accuracy were computed at the nucleotide, exon, and protein product levels. The results indicated that the predictive accuracy of the programs analyzed was lower than originally found. The accuracy was even lower when considering only those sequences that had recently been entered and that did not show any similarity to previously entered sequences. This indicates that the programs are overly dependent on the particularities of the examples they learn from. For most of the programs, accuracy in this test set ranged from 0.60 to 0.70 as measured by the Correlation Coefficient (where 1.0 corresponds to a perfect prediction and 0.0 is the value expected for a random prediction), and the average percentage of exons exactly identified was less than 50%. Only those programs including protein sequence database searches showed substantially greater accuracy. The accuracy of the programs was severely affected by relatively high rates of sequence errors. Since the set on which the programs were tested included only relatively short sequences with simple gene structure, the accuracy of the programs is likely to be even lower when used for large uncharacterized genomic sequences with complex structure. While in such cases, programs currently available may still be of great use in pinpointing the regions likely to contain exons, they are far from being powerful enough to elucidate its genomic structure completely. © 1996 Academic Press, Inc.

1. INTRODUCTION

The problems of gene identification and gene structure prediction in higher eukaryotic genomic DNA sequences by computational analysis have received wide attention. The problems have unquestionable practical interest, as emphasis in the Human Genome Project turns from mapping to large-scale sequencing. Indeed, the availability of accurate computational methods to elucidate the genic structure of large genomic sequences will simplify the analysis of uncharacterized sequence data and may, thus, speed the pace at which genome projects are being carried out. The problems are also appealing from a conceptual standpoint. Indeed, deciphering the genetic code means, after all, deducing the protein products potentially encoded by genomic DNA sequences through a purely syntactical analysis. The extent to which such a deduction can successfully be carried through depends on the elucidation of the processes involved in the pathway leading from DNA to proteins—namely, transcription, splicing, and translation (in higher eukaryotes).

Although methods to predict potential protein coding regions in genomic DNA sequences (see Fickett and Tung, 1992, for a review) have existed since the 1980s, the first programs to assemble potential DNA coding regions into translatable mRNA sequences were not available until the early 1990s. The first such programs for eukaryotic organisms were probably gm (Fields and Soderlund, 1990), initially developed to predict genes in *Caenorhabditis elegans* DNA sequences, and the Gelfand method (Gelfand, 1990) for mammalian genes. Since then, a number of programs that either explicitly assemble genes or, at least, predict a set of spliceable exons have been developed; GeneID (Guigó *et al.*, 1992), SORFIND (Hutchinson and Hayden, 1992), GeneParser (Snyder and Stormo, 1993, 1995a), GREAT (Gelfand and Roytberg, 1993), GenViewer (Milanesi *et al.*, 1993), GenLang (Dong and Searls, 1994), GRAIL II and GAP (Xu *et al.*, 1994a,b), FGENEH (Solovyev *et al.*, 1994), and Xpound (Thomas and Skolnick, 1994) (see Fickett, 1995, and Gelfand, 1995, for reviews). Most of these programs have been made publicly available through Internet servers, and although not as widely used as programs such as TestCode (Fickett, 1982), GRAIL (Uberbacher and Mural, 1991), and other methods that simply identify coding regions in DNA sequences—and that can, thus, be used to determine the likelihood that short nonassembled DNA fragments from single shotgun sequencing gels occur in coding

[1] To whom correspondence should be addressed at Institut Municipal d'Investigació Mèdica (IMIM), C/Dr. Aiguader 80, E-08003 Barcelona, Spain. Telephone: (343) 221-1009. Fax: (343) 221-3237. E-mail: rguigo@indy.imim.es.

0888-7543/96 $18.00

regions or to determine if partial cDNA sequences from expressed mRNAs occur in the translated segment of the mRNA—they are increasingly being used by experimental researchers to help analyze large assembled genomic DNA sequences [just to mention a few random examples, see Guo *et al.*, 1993, for a GeneID prediction and Dodemont *et al.*, 1994, for a gm prediction. In prokaryotic organisms, the GenMark software (Borodovsky and McIninch, 1993) has recently been used to help in the identification of potential genes in the genomes of *Haemophilus influenzae* (Fleischmann *et al.*, 1995) and *Mycoplasma genitalium* (Fraser *et al.*, 1995)].

An issue that naturally arises is that of the reliability of the predictions obtained by such programs. The issue concerns both users and developers. It is particularly important for the users confronted with the need to identify those functional domains potentially encoded in large uncharacterized genomic regions. Indeed, experiments are often planned on the basis of such predictions, and these may involve a considerable amount of effort and resources. There are, however, no data set and performance metrics standards that software developers agree upon for the consistent testing of gene structure prediction computer programs. Although developers usually provide some kind of performance evaluation, different programs are tested on different data sets using different measures. In addition, the test sets used are usually very small—the largest consisting of a few dozen "well-behaved" sequences—the criteria by means of which the sequences have been selected are not always stated, and sometimes the division between training and test sets is unclear. Thus, it is not easy for users to assess the real value of the different programs, their strengths, their weaknesses, and the particular problem for which a given program may be particularly suited. On the other hand, it is not easy for developers to assess the actual status of the problem: if a general solution has already been found—as it might appear after some of the limited test results thus far obtained—or, in contrast, if current programs perform only moderately well and only in a very well-defined, restricted set of DNA sequences with simple genic structure.

A few attempts have recently been made toward the implementation of standard performance metrics and independent benchmarking of algorithms for coding region identification and gene structure prediction in genomic DNA sequences. Fickett and Tung (1992) carried out an exhaustive comparative assessment of protein coding measures, developing a standardized benchmark to evaluate such measures. Singh and Krawetz (1994) also compared a number of coding region identification methods, by introducing the concept of coding potential error to measure accuracy. In neither of these cases, however, were gene structure prediction programs considered. Lopez *et al.* (1994), on the other hand, used a set of very large DNA genomic sequences to provide an independent test of the widely used GRAIL software, including the GRAIL 2 program, which predicts spliceable exons, but no comparison of GRAIL 2's performance with that of other available gene structure prediction programs was made. A comparative evaluation, similar to the one presented here, can be found in Snyder and Stormo (1995b). These authors, however, analyzed only a few of the programs available, using a rather small data set.

Here, we present a comprehensive comparative analysis of a number of gene structure prediction programs. The programs were run on a large set of genomic DNA sequences of known gene structure, and a number of performance metrics were introduced to measure the accuracy of the predictions at three different levels: coding nucleotide, exon structure, and final protein product. Only programs assembling genes or predicting, at least, a set of spliceable exons were considered—programs that identify only putative coding regions, such as TestCode (Fickett, 1982) and GRAIL (Uberbacher and Mural, 1991), or splice sites, such as NetGene (Brunak *et al.*, 1991) and SITEVIDEO (Kel *et al.*, 1993), or programs aimed at identifying genes not encoding protein products, such as trna-scan (Fichant and Burks, 1991), developed to predict tRNA genes, were not considered. When possible, the programs were accessed through Internet (e-mail) servers; otherwise the latest available version was installed and run locally. All nonanomalous (see Materials and Methods) genomic vertebrate sequences from the GenBank database Release 85 that were shorter than 50,000 bp and that contained single complete genes were initially considered. However, in the final test set only the sequences that had been entered into the database after Release 74 were included, in an attempt to minimize the overlap between our test set and the training sets used during the development of the programs analyzed. The final test set consisted of 570 sequences.

Results obtained indicated that the accuracy of the programs on the above test set was systematically lower than that originally found. For all the programs analyzed, the accuracy was even lower when considering only those sequences that did not show any similarity to sequences included in the database releases used during the development of the programs analyzed. This suggests that programs learn too much from the particular examples they are trained on and too little from the general biological mechanisms involved in the pathway leading from DNA to protein sequences. For most of the programs, accuracy in this test set ranged from 0.60 to 0.70 as measured by the Correlation Coefficient (where 1.00 corresponds to a perfect prediction and 0.00 is the value expected for a random prediction), while the average percentage of exactly identified exons was less than 50%. Only those programs that include protein sequence database searches to score predicted exons showed substantially greater accuracy, clearly indicating that an integrated approach, which combines lookup (database searches) and template (search by signal and search by content) methods, constitutes, so far, the best strategy to predict gene structure in genomic DNA sequences.

Programs showed some dependency—usually only slight—on the phylogenetic group within vertebrates in which they had been trained and tended to perform worse on low G + C content sequences, as previously noted (Xu *et al.*, 1994a; Lopez *et al.*, 1994; Snyder and Stormo, 1995a,b), but this was not a general trend. Programs were also sensitive to relatively high rates of sequence errors, but their robustness varied widely.

Since the sequences included in the test set constitute only a selected subset of the database with standard behavior, and the database is still a very biased sample of the vertebrate genomes, the accuracy of the programs analyzed is likely to be lower when used on real data from sequencing projects—larger sequences, of lower coding density, containing several genes or incomplete genes, encoding alternative products, etc. Thus, although the current generation of programs may still be of great use in pinpointing some of the regions containing exons in large DNA genomic sequences, the programs are unlikely to be able, in most cases, to elucidate their genomic structure completely. That is, there is a long way to go before computational programs exist that are able to locate all exons encoded in the sequence, to identify functional splicing variants, if any, and to assemble them in the correct (and maybe incomplete) genes, producing, if pertinent, a catalogue of alternative products for each of those genes and indicating the genomic regions—and the concrete sequence motifs—involved in the regulation of their expression. This should, we feel, be the ultimate goal of computational gene identification projects—projects whose applicability may not be limited to the analysis of existing genomes, but may also be extended to the engineering of new ones. To achieve such a goal better knowledge is required of the biological processes involved in the pathway of mRNA formation and subsequent translation and the integration of a model of such processes in the predictive schema of the computational systems.

2. MATERIALS AND METHODS

2.1. The Sequence Test Set

The sequences that constituted the test set to benchmark the programs analyzed were obtained from the vertebrate divisions of GenBank Release 85.0 (October 15, 1994). The set of all genomic DNA sequences from these divisions encoding at least one complete protein coding gene was initially considered. In practice the sequences were extracted, annotated with the keyword “DNA” in the LOCUS field, and containing at least one “CDS Key” in the FEATURES table. From this initial set, the following sequences were discarded: the sequences encoding at least one incomplete protein product (the “CDS Location” indicated an incomplete protein product or the “CDS Key” was labeled with the Qualifier “/partial”), the sequences for which the exact location of the protein coding regions was not unambiguously determined (the “CDS Location” contained the Operator “one-of”), the sequences encoding protein coding genes in the complementary strand (the “CDS Location” contained the Operator “complement”), the sequences encoding protein coding genes defined in database entries other than the one corresponding to the sequence (the Operator “join” in the “CDS Location” contained the Accession Number of a database entry), the sequences encoding pseudogenes (the “CDS Key” was labeled with the Qualifier “/pseudo”), the sequences encoding more than one gene or alternatively spliced forms of the same gene (containing more than one “CDS Key” in the FEATURES table), and the sequences encoding protein coding genes without introns (the “CDS Location” described a single continuous coding region).

We obtained, in this way, the set of all vertebrate genomic DNA sequences in the database encoding one complete (and only one) spliceable functional protein product in the forward strand. There were 1410 sequences in this set. The integrity of the data set was further enforced by discarding the following sequences: the sequences whose protein coding segment did not start with the codon ATG (74 sequences), the sequences whose protein coding segment did not end with a stop codon (100 sequences), the sequences whose protein coding segment was not a multiple of three (15 sequences), the sequences with donor sites lacking the dinucleotide GT as the first dinucleotide after the termination of the exons and with acceptor sites lacking the dinucleotide AT as the first dinucleotide before the initiation of exons (133 sequences), and the sequences whose protein coding segment contained stop codons in-frame (1 sequence).

Finally, sequences corresponding to immunoglobulins and histocompatibility antigens were also discarded, as well as additional pseudogenes—as indicated in the sequence entry DEFINITION field; 1043 sequences remained. After discarding three sequences longer than 50,000 bp, the final test set was obtained by selecting only the sequences with “date of entry” after January 1993, essentially equivalent to selecting the sequences entered into GenBank or modified since Release 74. We attempted, in that way, to minimize the overlap between the test set constructed here and the training sets used during the development of the programs analyzed. Although it is not always possible to infer from the published papers the database release used to build the sequence set on which a given program has been trained, it appears that only Xpound and GenLang among the programs analyzed may have included sequences in their training sets entered into GenBank after Release 74. Total lack of overlap between the test set used here and the training sets of the programs analyzed, though, cannot be guaranteed for those programs accessed through Internet servers, which may have been updated since they were first published.

There were 570 sequences in the final test set, totaling 2,892,149 bp. There were 2649 coding exons, corresponding to 444,498 coding bp (which gives a coding density of about 15%). We will refer to this set as the ALLSEQ set. A subset of ALLSEQ was considered separately. It comprised those sequences in ALLSEQ that did not show a significant similarity to the vertebrate sequences entered into the database prior to January 1993. We will refer to this set as NEWSEQ. Our aim here was to build a test set of sequences that did not show similarity to the sequences used in the training sets of the programs analyzed. Performance of the programs in this set may provide the most objective estimation of the real performance of the programs, or at least of their ability to implement the general properties of the genic sequences and not only the particularities of the examples they learn from. In practice, to build the NEWSEQ set we compared the protein sequence encoded by each sequence in the ALLSEQ set with the set of protein sequences encoded by the vertebrate genomic DNA sequences among the 1043 sequences resulting from the above procedure that had been entered into the GenBank database prior to January 1993. To compare the protein sequences we used the BLASTP program (Altschul *et al.*, 1990). Only the sequences in ALLSEQ for which no HSP score greater than 100 was found when compared with the above set of sequences were included in the NEWSEQ set. There were 196 sequences in this set. Although sequences in the NEWSEQ data set are on average substantially longer (6838 bp) than sequences in the overall ALLSEQ data set (5073 bp), coding density and G + C content were essentially identical in both sets (coding density is 0.14 in NEWSEQ and 0.15 in ALLSEQ; G + C content is 0.50 in NEWSEQ and 0.49 in ALLSEQ). It does not therefore appear that genes encoded by the NEWSEQ sequences are intrinsically more difficult to predict than genes encoded by sequences in the overall ALLSEQ set.

2.1.1. The sequence test set with frameshift errors. We studied the relative robustness of gene structure prediction programs to sequencing errors. Indeed, artifactual nucleotide insertions and deletions do occur while sequencing DNA. Moreover, plans are underway to speed up the full-length sequencing of the human genome by allowing lower sequence accuracy (Marshall, 1995). It appears interesting, thus, to evaluate the performance of the programs on sequences with frameshift errors. This may provide a more realistic estimation of the accuracy of the programs when analyzing real data and of their real value when used in such large-scale sequencing projects.

We thus introduced random frameshift mutations in the sequences of our test set. Mutations were produced with a frequency of 1%, with insertions and deletions being equiprobable and with insertions leading to the insertion of any of the four nucleotides with equal probability. It may be argued that a rate of sequence errors of 1% is too high to be realistic (for instance, the overall accuracy planned for the aforementioned full-length sequencing projects is about one order of magnitude greater, 99.9%). However, the 1% error rate appears to be a reasonable estimation for single-pass sequencing, and thus it can be taken as the upper boundary error rate that will be tolerated for the sequences with which the gene structure prediction programs are to be confronted. In addition, by testing the programs under such extreme conditions, we were in a better position to assess the programs' differential behavior to sequencing errors.

All the data sets mentioned here can be accessed through the World Wide Web (WWW) at the URL "http://www.imim.es/GeneIdentification/Evaluation/Index.html."

2.2. Measures of Prediction Accuracy

We measured only the accuracy of the predictions, and no other components of the performance such as execution time or memory requirements of the programs were analyzed. We measured the accuracy of the predictions at three different levels: coding nucleotide sequence, exonic structure, and protein product.

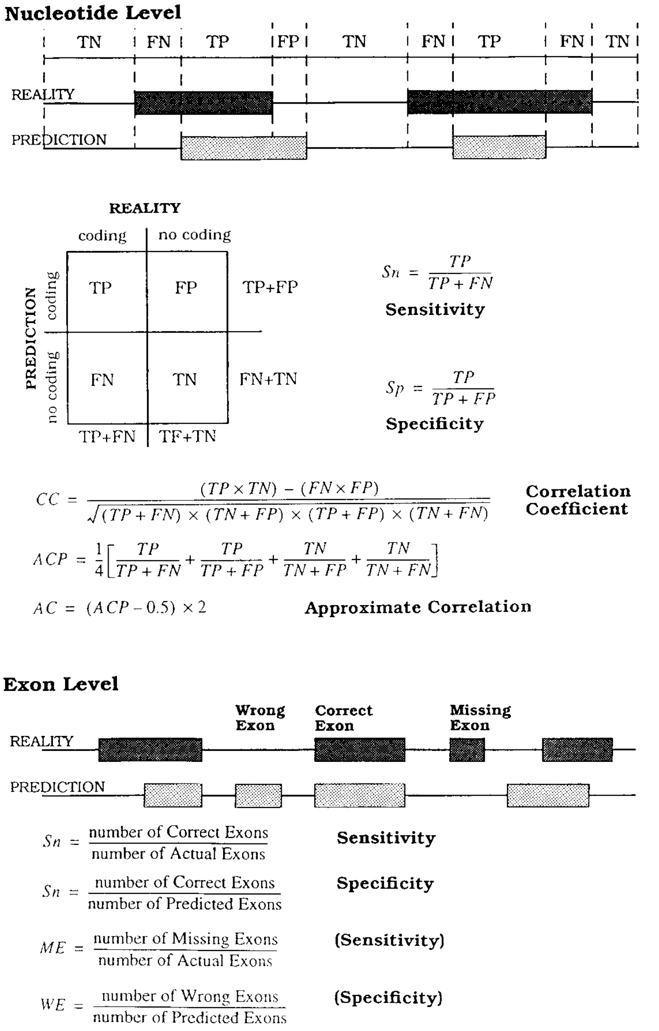

Nucleotide Level
TN
FN
TP
FP
TN
FN
TP
FN
TN
REALITY
PREDICTION
REALITY
coding
no coding
PREDICTION
coding
no coding
TP
FP
TP+FP
FN
TN
FN+TN
TP+FN
TF+TN
$Sn = \frac{TP}{TP+FN}$
Sensitivity
$Sp = \frac{TP}{TP+FP}$
Specificity
$CC = \frac{(TP \times TN) - (FN \times FP)}{\sqrt{(TP+FN) \times (TN+FP) \times (TP+FP) \times (TN+FN)}}$
Correlation Coefficient
$ACP = \frac{1}{4}\left[\frac{TP}{TP+FN} + \frac{TP}{TP+FP} + \frac{TN}{TN+FP} + \frac{TN}{TN+FN}\right]$
$AC = (ACP - 0.5) \times 2$
Approximate Correlation
Exon Level
Wrong Exon
Correct Exon
Missing Exon
REALITY
PREDICTION
$Sn = \frac{\text{number of Correct Exons}}{\text{number of Actual Exons}}$
Sensitivity
$Sn = \frac{\text{number of Correct Exons}}{\text{number of Predicted Exons}}$
Specificity
$ME = \frac{\text{number of Missing Exons}}{\text{number of Actual Exons}}$
(Sensitivity)
$WE = \frac{\text{number of Wrong Exons}}{\text{number of Predicted Exons}}$
(Specificity)

FIG. 1. Measures of prediction accuracy at the nucleotide and exon levels.

2.2.1. *Coding nucleotide.* At this level, we measured the accuracy of a prediction on a test sequence by comparing the predicted coding value (coding or noncoding) with the true coding value for each nucleotide along the test sequence. This has been one of the most widely used approaches in evaluating the accuracy of coding region identification and gene structure prediction methods. We can assume that both prediction and reality are binary variables whose values (coding or noncoding) have been observed along the nucleotides in the sequence. Most of the measures that have been used to evaluate the accuracy of the predictions can then be introduced as measures of association between these binary variables. As is customary in representing the joint distribution of two binary variables, we can use a 2 × 2 contingency table to represent the relationship between the actual and the predicted coding nucleotides on a test sequence (Fig. 1). In the left upper cell of the table we place the number of coding nucleotides that have been correctly predicted as coding (the true positives, TP), while in the right lower cell we place the number of noncoding nucleotides correctly predicted as noncoding (the true negatives, TN). The other two cells register the number of nucleotides in which prediction and reality disagree, that is, the number of coding nucleotides predicted as noncoding (the false negatives, FN) and the number of noncoding nucleotides predicted as coding (the false positives, FP). The 2 × 2 table contains the most information for measuring the association between prediction and reality (the accuracy of the prediction), and several measures have been proposed to capture such information in a single scalar metric (see Sneath and Sokal, 1973, and Anderberg, 1973, for a general discussion on measures of association between binary variables). Sensitivity and Specificity are probably the two most widely used measures among those that can be derived from a 2 × 2 table. Usually, sensitivity (Sn) and specificity (Sp) are defined as

$$\mathrm{Sn} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \quad \mathrm{Sp} = \frac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FP}}; \qquad [1]$$

that is, Sn is the proportion of coding nucleotides that have been correctly predicted as coding, and Sp is the proportion of noncoding nucleotides that have been correctly predicted as noncoding. However, since the frequency of noncoding nucleotides in genomic DNA sequences is much greater than the frequency of coding nucleotides (both in reality and in the predictions), TN tends to be much larger than FP, and thus Sp, as computed above, systematically produces very large noninformative values. Thus, in the gene structure prediction literature, specificity has traditionally been computed instead as

$$\mathrm{Sp} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}}; \qquad [2]$$

that is, in the context of gene structure prediction programs, Sp is the proportion of predicted coding nucleotides that are actually coding. This is the measure termed Sen2 in Guigó *et al.* (1992), and explicitly Specificity in Snyder and Stormo (1993) and Dong and Searls (1994), and it is the measure of specificity that we used here. Note that both sensitivity and specificity are conditional probabilities. Sn is the probability of a nucleotide being predicted as coding given that it is actually coding, and Sp is the probability of a nucleotide being actually coding given that it has been predicted as coding. Indeed, if x denotes the actual state of a given nucleotide (c for coding and n for noncoding), and $F(x)$ is the predicted state for such a nucleotide, then $\mathrm{Sn} = P(F(x) = c \mid x = c)$ and $\mathrm{Sp} = P(x = c \mid F(x) = c)$. Note that we can have very high sensitivity Sn with very low specificity Sp—predicting, for instance, every nucleotide as coding—and, conversely, high specificity with low sensitivity—predicting, for instance, only a few nucleotides as coding. Thus, neither Sp nor Sn alone constitute good measures of global accuracy, and it appears desirable to use a single scalar value summarizing both of them as a measure of global accuracy. In the gene structure prediction literature, the preferred measure has traditionally been the Correlation Coefficient CC. From the above 2 × 2 table, CC is defined as

$$\mathrm{CC} = \frac{(\mathrm{TP} \times \mathrm{TN}) - (\mathrm{FN} \times \mathrm{FP})}{\sqrt{(\mathrm{TP} + \mathrm{FN}) \times (\mathrm{TN} + \mathrm{FP}) \times (\mathrm{TP} + \mathrm{FP}) \times (\mathrm{TN} + \mathrm{FN})}}. \quad [3]$$

Although CC, as defined above, has received different names, Eq. [3] is only the special formula for the Pearson product-moment correlation coefficient in the particular case of two binary variables. CC depends not only on $P(F(x) = c \mid x = c)$ and $P(x = c \mid F(x) = c)$, but also on $P(F(x) = n \mid x = n)$ and $P(x = n \mid F(x) = n)$, respectively, the probability of a nucleotide being predicted as noncoding given that it is actually noncoding and the probability of a nucleotide being actually noncoding given that it has been predicted as noncoding. While the first two probabilities correspond to the sensitivity and specificity in predicting coding nucleotides, the second two probabilities can be seen as the sensitivity and specificity in predicting noncoding nucleotides. CC appears therefore to be particularly appropriate as a measure of overall prediction accuracy. It has, in addition, a statistical interpretation. It is easy to see that $\mathrm{CC}^2 = \chi^2/n$, where χ^2 is the statistic for the independence between the binary variables in the 2 × 2 table (reality and prediction) and where n is the sample size (the length of the sequence, $n = \mathrm{TP} + \mathrm{FN} + \mathrm{FP} + \mathrm{TN}$). To evaluate gene structure prediction programs, CC has been widely used (Guigó *et al.*, 1992; Snyder and Stormo, 1993, 1995a,b; Dong and Searls, 1994; Xu *et al.*, 1994b; Solovyev *et al.*, 1994). CC, however, has an undesirable property: it is not defined when TP + FN, FP + TN, TP + FP, or FN + TN is zero, that is, whenever either prediction or reality does not contain both coding and noncoding nucleotides, for instance when a program predicts no genes in a test sequence. Since this is often the case, and, on the other hand, it may be of interest to explore the behavior of gene structure prediction programs in DNA sequences lacking coding regions, it appears desirable to use a measure of accuracy that can be computed in any circumstance. One such measure is the simple matching coefficient, SMC, which is defined as

$$\mathrm{SMC} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{FN} + \mathrm{FP} + \mathrm{TN}}, \quad [4]$$

and is the probability of correct prediction (that is, of a nucleotide having the same coding value in reality and in the prediction). SMC has also been used to evaluate coding region and gene structure prediction methods (see, for instance, Uberbacher and Mural, 1991), and it is the measure to which the complement of Singh and Krawetz's normalized coding potential error (1994) reduces when the predicted coding potential is assumed to be 1 for predicted coding nucleotides and 0 for predicted noncoding nucleotides. Although the coding potential error may be very useful for measuring the performance of coding region prediction methods, when the prediction on each nucleotide is a continuous variable related to the nucleotide coding potential, SMC does not appear to be particularly appropriate for measuring the performance of the gene structure prediction methods, when the prediction on each nucleotide is a binary variable related to the nucleotide coding state. Indeed, it is easy to see that $\mathrm{SMC} = P(x = c)P(F(x) = c \mid x = c) + P(x = n)P(F(x) = n \mid x = n)$. Since $P(x = n)$ is usually much greater than $P(x = c)$, the "sensitivity" predicting noncoding nucleotides $P(F(x) = n \mid x = n)$—the specificity defined in [1], which in turn systematically produces very large values—carries much more weight than the sensitivity predicting coding nucleotides Sn. Thus, we can easily imagine situations in which a prediction missing all of the coding region gets the same SMC value as another prediction in which the exons were approximately located. Let us imagine, for example, a 1000-bp sequence containing a single 100-bp exon and two predictions, the first predicting a single 250-bp exon, totally covering the true exon and extending 75 bp at each side, and the second predicting a 50-bp exon within a noncoding region. In both cases, SMC would be 0.85; however, there would be general agreement that the first prediction is superior to the second one, since in the first case the true coding region had been fairly well identified, while in the second case an exon has been predicted within a noncoding region.

It appears convenient, thus, to have at our disposal a measure of prediction accuracy that, as the CC, appropriately summarizes the information from the 2 × 2 contingency table, but that, as the SMC, can be computed in any circumstance. Here, we use the following approach to compute one such measure. We compute above four conditional probabilities on which CC depends: $P(F(x) = c \mid x = c) = \mathrm{TP}/(\mathrm{TP} + \mathrm{FN})$, $P(x = c \mid F(x) = c) = \mathrm{TP}/(\mathrm{TP} + \mathrm{FP})$, $P(F(x) = n \mid x = n) = \mathrm{TN}/(\mathrm{TN} + \mathrm{FP})$, and $P(x = n \mid F(x) = n) = \mathrm{TN}/(\mathrm{TN} + \mathrm{FN})$ and average over those that are defined. Since at least two of these probabilities are always defined, such an average can always be computed, like the SMC. Unlike the SMC, however, it weights the accuracy on coding and noncoding regions equally, irrespective of their relative frequency in the test sequence, and it thus appears to be, like the CC, an appropriate measure of global prediction accuracy. We will refer to this measure as Average Conditional Probability, ACP. In the case in which the four probabilities are defined, the formula for ACP is

$$\mathrm{ACP} = \frac{1}{4}\left[\frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} + \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} + \frac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FP}} + \frac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FN}}\right]. \quad [5]$$

This formula appears in Anderberg (1973). In our case, it can be interpreted as the probability of a nucleotide being in a given state in either reality or prediction, since it is in a given state in the other. As ACP is a probability, it ranges from 0 to 1. The transformation

$$\mathrm{AC} = (\mathrm{ACP} - 0.5) \times 2 \quad [6]$$

ranges from −1 to 1 and can be compared to CC. We will refer to such a transformation as the Approximate Correlation, AC. We did not systematically study the behavior of the AC, but it appears to approximate that of CC. Although in all of the cases in which we were able to compute both of them (several thousand cases), we observed that $|\mathrm{AC}| \geqslant |\mathrm{CC}|$; in more than 90% of the cases AC was within 0.05 of the actual CC value. In consequence, AC seems to measure the association between prediction and reality appropriately and can thus be used as an alternative to the CC, both as a measure of gene structure prediction accuracy and as a measure to optimize when developing gene structure prediction programs. Unlike the CC, it has a probabilistic interpretation, and it can be computed in any circumstance.

2.2.2. Exonic structure. At this level, we measure the accuracy of the predictions by comparing predicted and true exons along the test sequence (Fig. 1). This has also been a widely used approach to evaluate gene structure prediction programs. There is not, however, a unique criterion to consider an exon as correctly predicted. The criterion generally used, and the one we follow here, is to consider an exon correctly predicted only when an exact match—with correct splicing boundaries—occurs between actual and predicted exons. But it could be legitimate as well to consider it when the overlap between predicted and actual exon is greater than some given threshold or when at least one of the splice sites defining the predicted exon has been correctly identified [termed partial matches in Hutchinson and Hayden (1992), 1-edge exons in Xu *et al.* (1994b), and half right exons in Dong and Searls (1994)].

To evaluate accuracy at the exonic structure level we measure both sensitivity and specificity. To compute sensitivity and specificity we use the equivalent to formulas [1] and [2]. Thus, Sensitivity is the proportion of actual exons in the test sequence that are correctly predicted, and Specificity is the proportion of predicted exons that are correctly predicted. As above, they are both conditional probabilities. Given the stringent criteria that we use to consider an exon as correctly predicted, we compute two additional measures of sensitivity and specificity. We compute first the proportion of true exons without overlap to predicted exons—the Missing Exons (ME)—and the proportion of predicted exons without overlap to actual exons—the Wrong Exons (WE).

Evaluation at the exonic structure level provides complementary information about the accuracy of the programs compared to that provided by evaluation at the coding nucleotide level. At the coding nucleotide level we are measuring how well the sequence coding regions are located—the search by content component of the gene structure prediction programs—while at the exonic structure level, we are measuring how well the sequence atomic signals (splice sites and start and stop codons) are identified—the search by signal component of the programs. High accuracy at the coding nucleotide level does not necessarily imply high accuracy at the exonic structure level. For instance, a program can have high sensitivity at the coding nucleotide level, because most of the coding nucleotides have been identified, but very low sensitivity at the exonic structure level, because most splicing signals have been incorrectly predicted.

2.2.3. Protein product. At this level, we measure the accuracy of the predictions by comparing the protein product encoded by the actual gene in the test sequence with the protein product encoded by the predicted gene. Again, measuring accuracy at this level provides additional information on the performance of the programs compared to that provided by the above approaches. Here, we are measuring not only how well the coding regions and splicing signals have been identified, but also how well the resulting predicted exons can be assembled into the correct final protein product. Note that it is possible for a program to have high accuracy at the coding nucleotide and exonic structure level, but very poor accuracy at the final protein product level, for instance, if a few mispredicted nucleotides change the reading frame along most of the predicted gene. The ability to predict correctly the protein sequence potentially encoded by the genomic fragment is essential to the experimenter, for whom the fact that the predicted protein sequence may show similarity to known protein sequences strengthens the evidence for the predicted gene. It is also essential to developing automatic methods for large genomic sequence analysis and annotation. In the gene structure prediction literature, only Fields and Soderlund (1990) provide an evaluation of the gm program at the final protein product level. They compute the percentage of the actual amino acid sequence predicted correctly and the absolute number of extra amino acids in the prediction. They appear to do this by directly inspecting the amino acid sequences. Such an approach, however, can be carried out only when there are few sequences to inspect (five in their case) and there is substantial similarity between predicted and actual protein sequences. We follow here a more generalizable approach. We obtain a global alignment between the predicted amino acid sequence and the actual sequence and compute the percentage of amino acid identity (over the total length of the alignment). In practice, we used the program ALIGN (Myers and Miller, 1988) from the FASTA program package (Pearson and Lipman, 1988; Pearson, 1990) to obtain the global alignments. The scoring matrix used was the PAM250 matrix (Dayhoff *et al.*, 1978), with penalties for the first residue in a gap and additional residues in a gap set to −12 and −4, respectively (the default FASTA values).

Evaluation at the protein product level was carried out only for those programs predicting complete gene structure or predicting individual exons with assignation of frame.

2.3. *The Programs Analyzed*

We analyzed the performance of most programs designed to predict gene structure (or at least a set of spliceable exons) in vertebrate or human genome sequences that are widely available. The following programs, however, are missing from our analysis: GeneModeler (Fields and Soderlund, 1990), GenMark (Borodovsky and McIninch, 1993), and GAP 3 (Xu *et al.*, 1994b). GeneModeler and GenMark were initially developed to analyze sequences from organisms other than vertebrates—*C. elegans* and *Escherichia coli,* respectively—and although they have been later extended to find exons in the sequences of other organisms, they have not been widely tested on such a case. Moreover, the users are allowed considerable freedom in specifying the input parameters. Although such flexibility may be beneficial in analyzing specific sequences, it hinders generic analysis of large sequence data sets. Indeed, we, like others (Snyder and Stormo, 1995b), have found it difficult to finetune input parameters so as to obtain general satisfactory results with such programs. GAP3 is the gene assembly program included in the GRAIL system. It assembles genes from the exons predicted by GRAIL 2. GAP3 may be the most accurate gene structure prediction program currently available; it has been reported to have higher accuracy than GRAIL 2 at the nucleotide level (Xu *et al.*, 1994b), although other analyses have reported contradictory results (Snyder and Stormo, 1995b). Unfortunately, when our evaluation was performed, GAP3 could be accessed only through XGRAIL, an X-based Internet client-server implementation of the GRAIL tools. In XGRAIL sequences have to be fetched individually, which, in practice, complicates enormously the type of automatic analysis of large data sets that we are presenting here. (The situation has, however, changed since we first submitted the paper, and we include in the Discussion the results obtained when testing a newly released version of GAP3.)

Other researchers within the field of computational gene identification have developed systems to predict gene structure in vertebrate DNA sequences. These include the GREAT program (Gelfand and Roytberg, 1993) and GenViewer (Milanesi *et al.*, 1993). These programs, however, are less generally available, and they have not been analyzed here.

We ennumerate the programs analyzed below, indicating if they were run locally or through Internet servers, the version used, the options with which they were run, and the parsing of the output we have carried out to derive a unique prediction for each test sequence. All analyses through Internet servers were performed between November 1994 and May 1995. Obviously, the practical utility of the programs analyzed here depends not only on prediction accuracy, but also on availability and type of access. Interested readers can find such information in Milanesi *et al.* (1994) and Fickett and Guigó (1996).

2.3.1. GeneID (Guigó et al., 1992) and GeneID+. The GeneID program was run through the e-mail server located at geneid@darwin.bu.edu. Sequences were sent with the option “-noexonblast” to disactivate amino acid database searches (see below). GeneID produced no prediction for two of the original sequences and for four of the mutated sequences. GeneID predicts up to 20 ranked complete gene models. The top ranking complete model was taken to be the predicted gene. The predicted protein product was derived by translating the predicted gene. GeneID+ (Guigó and Knudsen, manuscript in preparation) is a version of GeneID in which predicted exons showing similarity to known amino acid sequences are rescored. It is also accessed through the above e-mail server. Currently, it analyzes only sequences shorter than 8000 bp, and it is the default for sequences up to such a length. GeneID+ produced no prediction for one of the original sequences and for three of the mutated ones. The protein sequence databases installed at the server at the time it was accessed were those distributed with Entrez Release 8 (November 1993.) Note that there may be some overlap between the sequences in such databases and the protein sequences encoded by the DNA sequences in the test set.

2.3.2. SORFIND (Hutchinson and Hayden, 1992). Version 2.5 was installed and run locally under DOS on a PC 486DX2. The default options were used to analyze the sequences. SORFIND failed to analyze 9 of the original sequences, but only 6 of the mutated ones, and produced no prediction in 43 mutated sequences. SORFIND output consists of a set of nonoverlapping exons classified as first, internal, and terminal exons and the amino acid sequence corresponding to each exon. Used with the “-g” option, the program predicts only one first exon and one terminal exon, and it could be argued that this is the option that should have been used in our analysis. The program, however, does not attempt to assemble complete genes in any case, and its overall performance appears to be the same with or without the -g option (Gordon B. Hutchinson, B.C. Canada, pers. comm., 19 Dec. 1994). The set of predicted exons was taken to be the predicted gene, and the concatenation of the exon’s translations in a single amino acid sequence, the predicted protein product.

2.3.3. GeneParser2 and GeneParser3 (Snyder and Stormo, 1995). The GeneParser programs were installed and run locally on a SGI workstation running Irix 5.2. GeneParser2 failed to analyze eight of the original and mutated sequences. GeneParser2 predicts a set of nonoverlapping exons that may not necessarily be assembled in a single gene. Such a set of exons was taken to be the predicted gene, but no attempt was made to translate the prediction into a protein product. GeneParser3 is a version of GeneParser that uses the potential similarity between the query sequence and the known amino acid sequences as evidence in gene identification. Although the version installed locally does not have explicit limits in the length of the sequences that can be analyzed, to make the results comparable to the ones obtained by GeneID+—the only other program that uses information from protein sequence database searches—only sequences up to a length of 8000 bp were analyzed with GeneParser3. For the same reason, the subset from SWISSPROT and PIR, comprising those sequences with "date of entry" prior to November, 1993 have been used as the amino acid sequence database—which makes such an amino acid sequence database essentially identical to the one used at the GeneID+ Internet server site.

2.3.4. GRAIL 2 (Xu et al., 1994). The GRAIL 2 program was accessed through the GRAIL e-mail server at GRAIL@ornl.gov. Sequences were sent with the option "-2" to activate GRAIL 2 searches. GRAIL 2 produced no prediction in 23 of the original sequences and in 42 of the mutated ones. GRAIL 2's final prediction consists of a set of nonoverlapping exons in both strands with reading frame assignation and quality. No assembly of complete genes is attempted. Predictions in the reverse strand were ignored, and the predicted gene was taken to be the set of predicted exons in the forward strand irrespective of their quality. The predicted protein was obtained by translating the predicted exons in the assigned frame and concatenating the result in a single amino acid sequence.

2.3.5. GenLang (Dong and Searls, 1994). GenLang was accessed through the e-mail server at genlang@cbil.humgen.upenn.edu. Sequences were analyzed with the options "vertebrate" and "protein." GenLang predicted no gene in 30 of the original sequences and in 49 of the mutated ones. GenLang predicts a set of ranked assembled genes. The top ranking one was assumed to be the predicted gene, and its translation the predicted protein product.

2.3.6. FGENEH (Solovyev et al., 1994). FGENEH was accessed through the e-mail server at service@bchs.uh.edu, with the keyword "fgeneh" in the subject line of the messages. FGENEH failed to analyze one of the original sequences and predicted no gene in 22 of the original sequences and in 41 of the mutated ones. FGENEH predicts a single complete gene and the corresponding amino acid sequence, which were taken therefore as the predicted gene and the predicted protein product, respectively.

2.3.7. Xpound (Thomas and Skolnick, 1994). The Xpound program was installed and run locally on a Sun workstation with Solaris 5.3. Sequences were run through the Xpound and Xreport programs with default options. Xpound predicted no gene in 28 of the original sequences and in 51 of the mutated ones. With default cutoff coding probability, Xpound predicts a set of nonoverlapping coding regions with spliceable boundaries, but no assembly of exons is attempted. Such a set was taken to be the predicted gene. No predicted protein product was derived, since Xpound does not produce reading frame assignation.

3. RESULTS

Sequences in the test set were analyzed through the gene structure prediction programs. For each sequence the gene predicted by the programs was compared with the actual gene, as annotated in the "CDS Key" of the FEATURES table of the GenBank entry corresponding to the sequence. Table 1 summarizes the results obtained. It contains the average value over the set of sequences for the different measures of prediction accuracy considered. Averages are given both for the full set of sequences ALLSEQ and for the subset of sequences since GenBank Release 80 not showing similarity to sequences in previous GenBank releases, NEWSEQ. Measures are averaged only over those cases for which they are defined. This overestimates the accuracy of the programs that are designed to be conservative and that fail to predict genes in a substantial number of sequences from our test sets. The tables containing the exhaustive results on each of the test sequences can be accessed through the WWW at the URL "http://www.imim.es/GeneIdentification/Evaluation/Index.html."

There is some controversy as to the most appropriate way of summarizing prediction accuracy statistics: either averaging by sequence—as we have chosen to do here—or summing all results as in a single very large concatenated sequence and obtaining the statistics from these total numbers. The methods have been termed by gene and by base, respectively, in Dong and Searls (1994). Averaging by gene has been used in Guigó *et al.* (1992) and Dong and Searls (1994), while averaging by base has been preferred in Snyder and Stormo (1993, 1995a,b), Xu *et al.* (1994b), and Solovyev *et al.* (1994). Although calculating the statistics by base has the effect of weighting the sequences on the basis of length, it appears to have little justification from a statistical standpoint. Indeed, users are interested in the expected performance of the programs when analyzing a given anonymous DNA sequence, that is, in the expected value of the measures of prediction accuracy. The unbiased estimation of such an expectation is the average of the measures in a random sample of sequences. Moreover, some of the measures we use have little signification, or cannot even be computed, on a by base basis. This is the case, for instance, of the percentage similarity at the protein product level between the actual and the predicted gene.

Table 2 shows the results obtained for the set of mutated sequences. When comparing predictions and reality, the exon boundaries of the actual genes—as annotated in GenBank—were adjusted to take into account the frameshift mutations introduced in the sequence. The top of Table 2 is exactly the same as Table 1, while the bottom of the table summarizes the effect of the mutations on the accuracy of the programs, by providing the explicit differences between the values in the original and the mutated sets of sequences for some of the measures of accuracy considered. The measures for which the differences are explicitly provided are the Approximate Correlation (AC) at the nucleotide level, the Average Sensitivity and Specificity at the exon level, and the Percentage of Similarity at the protein level.

Most of the programs analyzed here have been developed specifically to analyze human sequences or have been used with parameters specific to human sequences. However, here they were tested on a set that included sequences from other vertebrate organisms. Since this may have affected their performance, we analyzed the accuracy of the programs on the human sequences and on the sequences from other vertebrate organisms separately. In addition, we analyzed the accuracy of the programs within each of the GenBank vertebrate divisions. The results appear in Table 3.

It has been noted that the accuracy of some of the

TABLE 1

Performance of the Programs in the Test Set of Original (Nonmutated) Sequences

	Sequences			Nucleotide				Exon					Protein
				Sn	Sp	Ac	(CC)	Sn	Sp	(Sn + Sp)/2	ME	WE	(% Sim)
FGENEH	All	569	22	0.77	0.88	0.78 ± 0.26	0.80	0.61	0.64	0.64 ± 0.33	0.15	0.12	71%
	New	195	11	0.70	0.83	0.70 ± 0.27	0.73	0.51	0.54	0.54 ± 0.30	0.22	0.18	62%
GeneID	All	570	2	0.63	0.81	0.67 ± 0.27	0.65	0.44	0.46	0.45 ± 0.34	0.28	0.24	57%
	New	196	1	0.58	0.78	0.62 ± 0.27	0.60	0.41	0.43	0.42 ± 0.30	0.34	0.27	52%
GeneParser2	All	562		0.66	0.79	0.67 ± 0.26	0.65	0.35	0.40	0.37 ± 0.30	0.29	0.17	
	New	188		0.63	0.76	0.63 ± 0.25	0.62	0.33	0.39	0.36 ± 0.27	0.32	0.20	
GenLang	All	570	30	0.72	0.79	0.69 ± 0.28	0.71	0.51	0.52	0.52 ± 0.33	0.21	0.22	62%
	New	196	13	0.63	0.73	0.60 ± 0.29	0.63	0.39	0.44	0.43 ± 0.29	0.29	0.25	53%
GRAIL 2	All	570	23	0.72	0.87	0.75 ± 0.21	0.76	0.36	0.43	0.40 ± 0.27	0.25	0.11	(64%)
	New	196	6	0.69	0.85	0.71 ± 0.22	0.72	0.34	0.41	0.38 ± 0.27	0.30	0.13	(61%)
SORFIND	All	561		0.71	0.85	0.73 ± 0.24	0.72	0.42	0.47	0.45 ± 0.28	0.24	0.14	(61%)
	New	189		0.65	0.79	0.66 ± 0.25	0.65	0.36	0.39	0.38 ± 0.27	0.29	0.19	(54%)
Xpound	All	570	28	0.61	0.87	0.68 ± 0.24	0.69	0.15	0.18	0.17 ± 0.23	0.33	0.13	
	New	196	7	0.58	0.83	0.64 ± 0.25	0.64	0.12	0.15	0.14 ± 0.20	0.36	0.16	
GeneID+	All	478	1	0.91	0.91	0.88 ± 0.16	0.88	0.73	0.70	0.71 ± 0.29	0.07	0.13	84%
	New	143		0.88	0.87	0.85 ± 0.18	0.84	0.68	0.64	0.66 ± 0.29	0.10	0.15	79%
	New2	50		0.85	0.85	0.82 ± 0.22	0.82	0.67	0.64	0.65 ± 0.30	0.13	0.16	75%
GeneParser3	All	478		0.86	0.91	0.86 ± 0.18	0.85	0.56	0.58	0.57 ± 0.30	0.14	0.09	
	New	143		0.83	0.89	0.82 ± 0.18	0.82	0.50	0.53	0.51 ± 0.33	0.17	0.09	
	New2	50		0.83	0.91	0.84 ± 0.15	0.84	0.51	0.56	0.53 ± 0.34	0.17	0.07	

Note. The measures of prediction accuracy discussed under Materials and Methods have been averaged over the set of sequences effectively analyzed. At the nucleotide level, Sensitivity (Sn), Specificity (Sp), and Approximate Correlation (AC) are given. For comparison purposes, the value of the Correlation Coefficient (CC) is also given. At the exon level, Sensitivity SN, Specificity Sp, their average (Sn + Sp)/2, Missing Exons (ME), and Wrong Exons (WE) are given. Finally at the protein level, the Percentage Similarity (% Sim) is given. % Sim is shown between parentheses for those programs that do not attempt to assemble genes, but that provide translational frame for the predicted exons. % Sim was assumed to be zero, when no prediction was made; all other measures were averaged only over those cases for which they are defined. Averages are given separately for all of the sequences in the test set, ALLSEQ, and for a subset unlikely to contain sequences similar to the sequences on which the programs were trained, NEWSEQ (results in boldface.) Standard deviations are given for AC at the nucleotide level and (Sn + Sp)/2 at the exon level. The table also gives (third column) the number of sequences effectively analyzed for each program (over a total of 570 sequences in ALLSEQ and 196 in NEWSEQ) and the number of sequences among those analyzed for which no gene was predicted (fourth column). Performance of the programs using amino acid similarity searches—GeneID+ and GeneParser3—is shown separately. In addition to performance in the ALLSEQ and NEWSEQ data sets, performance for such programs is also given in a third data set, NEWSEQ2, a test set in which the DNA sequences are unlikely to encode amino acid sequences included in the databases used by these programs.

programs analyzed here decreases on G + C-poor sequences (Xu *et al.*, 1994a; Snyder and Stormo, 1995a,b; Lopez *et al.*, 1994), and it has also been shown that some of the most widely used coding statistics are strongly dependent on G + C content (Guigó and Fickett, 1995). We analyzed the accuracy of the programs on the 15% G + C-poorest fraction of the sequence test set separately. The results appear in Table 4.

4. DISCUSSION

Computational gene identification will play an increasingly important role as the human genome project enters the large-scale sequencing phase, and a number of computational methods have recently been developed to predict gene structure in uncharacterized genomic DNA. The evaluation presented here attempts to assess the current status of the problem and to identify the most promising approaches on which future research could be based. We are interested in identifying the general trends—strengths and weaknesses—of computational gene identification methods, rather than in comparing the accuracy of the particular programs. In fact, computational gene identification is a rapidly evolving field, and systems are constantly being improved and developed; it is likely, thus, that by the time the present evaluation reaches press, new versions and programs will have been developed, and some of the values of accuracy reported here will be out of date (see below). However, the work presented here has a methodological interest, since the approach that we take in constructing sequence data test sets and the measures of gene structure prediction accuracy that we introduce might contribute to the standardization of data set and performance metrics within the field.

4.1. *Accuracy of the Programs*

The first conclusion that can be drawn from the results obtained here is that the accuracy of the currently available gene structure prediction programs appears to be substantially lower than that previously found in more limited test sets. As inferred from the results in the NEWSEQ test data set (Table 1)—a set of se-

TABLE 2

Performance of the Programs in the Test Set of Mutated Sequences

				Nucleotide				Exon					Protein
	Sequences			Sn	Sp	Ac	(CC)	Sn	Sp	$\frac{(Sn + Sp)}{2}$	ME	WE	(% Sim)
FGENEH	All	570	41	0.48	0.79	0.55 ± 0.25	0.56	0.24	0.29	0.28 ± 0.24	0.32	0.18	32%
	New	196	19	0.41	0.74	0.48 ± 0.24	0.49	0.21	0.28	0.26 ± 0.23	0.41	0.24	27%
GeneID	All	570	4	0.37	0.67	0.43 ± 0.25	0.40	0.20	0.23	0.22 ± 0.23	0.42	0.34	26%
	New	196	2	0.34	0.65	0.40 ± 0.26	0.37	0.21	0.24	0.23 ± 0.22	0.45	0.36	24%
GeneParser 2	All	562		0.46	0.75	0.53 ± 0.26	0.50	0.22	0.29	0.26 ± 0.26	0.41	0.20	
	New	188		0.44	0.71	0.49 ± 0.26	0.47	0.23	0.31	0.27 ± 0.26	0.43	0.22	
GenLang	All	570	49	0.48	0.65	0.47 ± 0.23	0.48	0.21	0.23	0.23 ± 0.22	0.34	0.33	32%
	New	196	14	0.44	0.63	0.44 ± 0.22	0.44	0.19	0.22	0.21 ± 0.21	0.41	0.35	29%
GRAIL 2	All	570	42	0.49	0.84	0.59 ± 0.22	0.59	0.17	0.24	0.21 ± 0.23	0.40	0.12	(39%)
	New	196	12	0.47	0.82	0.56 ± 0.22	0.56	0.16	0.22	0.19 ± 0.21	0.44	0.14	(37%)
SORFIND	All	564	43	0.43	0.80	0.52 ± 0.24	0.53	0.19	0.28	0.24 ± 0.24	0.41	0.17	(33%)
	New	190	20	0.38	0.73	0.46 ± 0.24	0.47	0.18	0.27	0.23 ± 0.23	0.49	0.24	(27%)
Xpound	All	570	51	0.42	0.83	0.54 ± 0.22	0.53	0.05	0.07	0.06 ± 0.15	0.43	0.14	
	New	196	16	0.40	0.79	0.51 ± 0.23	0.50	0.05	0.06	0.06 ± 0.13	0.45	0.16	
GeneID+	All	478	3	0.55	0.80	0.59 ± 0.23	0.58	0.29	0.28	0.28 ± 0.24	0.23	0.25	40%
	New	143	1	0.54	0.76	0.57 ± 0.24	0.55	0.29	0.27	0.28 ± 0.23	0.26	0.28	39%
	New2	50		0.49	0.74	0.52 ± 0.25	0.51	0.29	0.28	0.29 ± 0.25	0.29	0.28	35%
GeneParser3	All	478		0.61	0.87	0.68 ± 0.22	0.66	0.31	0.37	0.34 ± 0.28	0.26	0.11	
	New	143		0.56	0.83	0.63 ± 0.26	0.61	0.29	0.36	0.33 ± 0.29	0.31	0.13	
	New2	50		0.57	0.89	0.67 ± 0.22	0.65	0.32	0.39	0.35 ± 0.30	0.30	0.09	

	Nucleotide (AC)			Exon $\left(\frac{(Sn + Sp)}{2}\right)$			Protein (% Sim)		
	Original	Mutated	Diff	Original	Mutated	Diff	Original	Mutated	Diff
FGENEH	0.78	0.55	−0.23	0.64	0.28	−0.36	71	32	−39
GeneID	0.67	0.43	−0.24	0.45	0.22	−0.23	57	26	−31
GeneID+	0.88	0.59	−0.29	0.71	0.28	−0.43	84	40	−44
GeneParser2	0.67	0.53	−0.14	0.37	0.26	−0.11			
GeneParser3	0.86	0.68	−0.18	0.57	0.34	−0.23			
GenLang	0.69	0.47	−0.22	0.52	0.23	−0.29	62	32	−30
GRAIL 2	0.75	0.59	−0.16	0.40	0.21	−0.19	64	39	−25
SORFIND	0.73	0.52	−0.21	0.45	0.24	−0.21	61	33	−28
Xpound	0.68	0.54	−0.14	0.17	0.06	−0.11			

Note. Sequences have a 1% frameshift error. The upper section is as in Table 1. The lower section summarizes Table 1 and the upper section of Table 2. The Approximate Correlation (AC) at the nucleotide level, the Average Sensitivity and Specificity at the exon level, and the percentage Similarity (% Sim) at the protein levels are given for each program in the original and mutated set of sequences (ALLSEQ). The differences between both values are also given, indicating the absolute decrease in performance when the programs are confronted with sequences with errors.

quences not likely to be similar to sequences included in the “training” sets of the programs analyzed—and setting aside those programs relying on amino acid sequence database searches, the average accuracy of the current generation of programs at the coding nucleotide level ranged from 0.62 to 0.71 as measured by the approximate correlation, substantially below values of accuracy previously reported. At the exonic structure level, the average proportion of actual exons—with exact splice boundaries—correctly identified by the programs ranged in the main from 0.33 to 0.51, while the average proportion of predicted exons that were exactly correct ranged from 0.39 to 0.54. In addition, the average proportion of (absolutely) missing exons—actual exons not overlapped by any predicted exon—ranged from 0.22 to 0.36, while the average proportion of (absolutely) wrong exons—predicted exons not overlapping actual exons—ranged from 0.13 to 0.27. Finally, at the protein product level, programs predicted an amino acid sequence showing on average between 52 and 62% similarity with the true amino acid sequence. Since the sequence set on which the programs were tested is highly biased, including only relatively short DNA sequences with simple gene structure and high coding density, the accuracy of the programs is likely to be even lower when confronted with large DNA sequences from randomly sequenced genomic regions of less standard composition and more complex structure. Indeed, the independent evaluation of the GRAIL software by Lopez *et al.* (1994) suggests that this is indeed the case. While GRAIL 2 produces only an average of 13% of false exons in the NEWSEQ data set, Lopez *et al.* found that such a percentage rises to 30% when GRAIL 2 is tested in a set of very large genomic sequences, some

TABLE 3

Performance of the Programs in Different Vertebrate Groups

	Human	Other vertebrates	Developed for	gbpri	gbrod	gbmam	gbvrt
FGENEH	0.80	0.77	Human	0.79	0.75	0.75	0.83
GeneID	0.62	0.69	Vertebrates	0.64	0.68	0.67	0.70
GeneID+	0.86	0.89		0.87	0.89	0.89	0.90
GeneParser2	0.68	0.66	Human	0.67	0.67	0.69	0.64
GeneParser3	0.86	0.85		0.84	0.88	0.82	0.87
GenLang	0.68	0.69	Vertebrates	0.69	0.69	0.66	0.69
GRAIL 2	0.78	0.73	Human	0.79	0.72	0.70	0.72
SORFIND	0.72	0.73	Human	0.73	0.71	0.72	0.77
Xpound	0.73	0.66	Human	0.74	0.62	0.72	0.63

Note. Left: Performance of the programs in human (176 sequences) versus nonhuman vertebrate sequences (394 sequences). The Approximate Correlation (AC) is averaged separately for the subset of human sequences and for the subset of the remaining nonhuman vertebrate sequences in the ALLSEQ data set. Right: Performance of the programs in the different GenBank vertebrate divisions. The AC is averaged separately for the sequences in ALLSEQ belonging to the "gbpri" (primates, 232 sequences), "gbrod" (rodents, 191 sequences), "gbmam" (mammals other than primates and rodents, 67 sequences), and "gbvrt" (vertebrate other than mammals, 80 sequences) divisions.

of them containing several genes. Although there are no similar data available, it is likely that the same trend would be observed on the other programs considered here.

A general trend of the programs analyzed is that they tend to enforce specificity over sensitivity; that is, correction in the predicted exons is emphasized, even if this means missing many actual exons. In fact, programs often overemphasize specificity, and a number of them (FGENEH, GenLang, GRAIL 2, and Xpound) failed to predict genes in a substantial number of cases. Emphasizing specificity may be beneficial if predictions are going to be used for further experimental research, a situation where investigation of false positives may be unproductive. Note, however, that programs still predicted a relatively large fraction of exons in noncoding regions. Indeed, for the majority of the programs, between one-fifth and one-fourth of the predicted exons did not overlap with any actual known exon—that is, they are wrong exons (Table 1). Moreover, specificity is likely to suffer more than sensitivity when programs are confronted with large genomic sequences containing large intergenic regions, instead of short test sequences of anomalously high coding density. Note that while the coding density of the sequence test set used here is about 15%, the human genome coding density may be lower than 5%. Obviously, an anomalously high coding density may produce anomalously high values of coding region prediction specificity. Indeed, we observed that wrong exons tend to be predicted at the termini of the test sequences—that is, upstream from the first coding exon or downstream from the last coding one, even though the fraction of noncoding intronic DNA in such sequences is usually larger than the fraction of noncoding terminal DNA. Thus, for most of the programs—setting aside those using amino acid similarity searches—the proportion of wrong exons predicted at the sequence termini ranged between 0.50 and 0.65 (data for individual programs not shown, but GeneParser2 with 0.45 being the only exception), while the proportion of noncoding DNA at the sequence termini was about 0.45. Although the idea cannot be discarded that a few of these predicted WE may correspond to actual undetected exons from upstream or downstream genes, it appears a more realistic hypothesis that programs have a tendency to predict coding exons in otherwise noncoding intergenic DNA. Corroboration of such a hypothesis is provided by the aforementioned independent evaluation of the GRAIL software by Lopez *et al.* (1994), which found the proportion of wrong exons in large genomic sequences to be 30%, instead of only 13%, as we found in the NEWSEQ data set.

With respect to sensitivity, none of the programs—excluding those using amino acid similarity searches—predicted on average more than 50% of the actual exons in a given DNA sequence, and about one-third of the actual exons in the sequences went absolutely undetected (the missing exons). Most of the programs showed greater difficulty in identifying terminal exons than initial ones. The proportion of initial ME over the total of actual initial exons in the ALLSEQ data set ranged between 16 and 28%, while the proportion of terminal missing exons over the total terminal exons ranged from 20 to 43% (data for individual programs not shown). GenLang was the only exception, being more accurate at predicting terminal exons than initial

TABLE 4

Performance of the Programs in Sequences of Low G + C Content

	Low G + C (<40%)	Medium or high G + C
FGENEH	0.83	0.77
GeneID	0.64	0.67
GeneID+	0.93	0.88
GeneParser2	0.58	0.68
GeneParser3	0.87	0.85
GenLang	0.62	0.70
GRAIL 2	0.67	0.76
SORFIND	0.77	0.72
Xpound	0.56	0.70

Note. The Approximate Correlation (AC) is averaged separately for the 15% lowest G + C content fraction of sequences from the ALLSEQ data set and for the remaining sequences.

ones. On the other hand, sensitivity is not likely to be affected as much as specificity when the programs analyze large uncharacterized genomic DNA sequences. Again, the independent evaluation of Lopez *et al.* (1994) seems to corroborate such an hypothesis. Lopez *et al.* found the percentage of ME predicted by GRAIL 2 in a set of very large genomic sequences to be about 30%, essentially the same as we found in the NEWSEQ data set.

Differences apparently exist in the performance of the programs. Programs are ranked very differently, however, depending on the level at which accuracy is measured, indicating that the programs have different (and complementary) strengths. For instance (and setting aside those programs using amino acid similarity searches), while GeneID and GenLang were among those that performed worst at the nucleotide level, they were among those that performed best at the exact exon level. Conversely, GRAIL 2, which gave the best performance at the nucleotide level, did not perform so well at the exact exon structure level. FGENEH appeared to be the program missing fewer actual exons, but GRAIL 2 was the program that predicted fewer exons incorrectly. The program eventually chosen, therefore, depends on where the user places the emphasis. Here, our goal was not to establish a ranked comparison among the programs, and we warn against relying too much on a single data set to compare the accuracy of different programs. It has been previously noted (Snyder and Stormo, 1995b) and we have also noticed that the performance of the programs is very sensitive to different data sets.

On the other hand, computational gene identification is a rapidly evolving field in which programs are constantly being improved and developed, so that the values of accuracy reported here may be already out of date for some (or most) of the programs analyzed. This is the case, for instance, for GAP 3, the gene assembly program included in the GRAIL system. After this paper was submitted a new version of GAP 3 was made available. This version could be used in batch mode, and we tested it on the set of 570 original sequences. We cannot fairly compare GAP 3 accuracy with that of the other programs, since the assumption under which such programs were tested—that there was little overlap between our sequence test set and the set of sequences on which the programs had been trained—may not hold for the recent version of GAP 3; and for this reason we have chosen not to include GAP3 results in Table 1. GAP 3, however, appears to be significantly more accurate than GRAIL 2. Thus, accuracy in the NEWSEQ data set as measured by the Approximate Correlation was 0.75 (while it was 0.71 for GRAIL 2) and at the exon level, as measured by the exact exon prediction, was 0.49 (while it was 0.38 for GRAIL 2). The same values in the overall ALLSEQ data set were 0.79 and 0.56, respectively.

4.2. Increased Accuracy of the Programs Including Similarity Searches

Programs using potential similarity between regions in the genomic DNA sequences and known amino acid sequences for the detection of likely exons seem to show substantially greater accuracy (Table 1). We should, however, point out the intrinsic difficulty of evaluating the real accuracy of the programs that use similarity with amino acid sequences to help in the detection of exons. For instance, if the amino acid sequence database used by a program includes the amino acid sequences encoded by the DNA sequences in a given test set, the accuracy of the program measured in such a test set will not be particularly indicative of the real accuracy of the programs when analyzing newly obtained genomic sequences—the amino acid sequences potentially encoded by them not likely to be yet known. Given that the DNA sequence test set used here (ALLSEQ) comprises sequences with GenBank "date of entry" from January 1993 and the protein sequence database used by GeneID+ when accessed through the e-mail server and by GeneParser3 when run locally were essentially those in the release of ENTREZ distributed in November 1993, there may be some overlap between the set of amino acid sequences used by the programs evaluated here and the amino acid sequences encoded by the DNA sequences in the test set. Since this may have produced an overly optimistic estimation of these programs' performance, we reevaluated GeneID+ and GeneParser3 on a new test set, in which we tried to minimize the overlap between the set of amino acid sequences used by these programs and the amino acid sequences encoded by the DNA sequences in the test set. Thus, we extracted the subset of ALLSEQ made up of those sequences with GenBank "date of entry" after January 1994 and considered only those that did not show similarity to vertebrate genomic DNA sequences entered into GenBank prior to January 1994. When sequences longer than 8000 bp were removed, 46 sequences remained (NEWSEQ2). As can be seen from Table 1, GeneID+ and GeneParser3 performed essentially the same on these sequences as on the NEWSEQ sequences.

The above results indicate that inclusion of amino acid similarity searches in the design of gene structure prediction programs is highly beneficial. The rationale for including amino acid similarity searches is clear: significant similarity between (the translation of) a region of genomic DNA sequence and known amino acid sequences is strongly indicative of the existence of a coding exon in such a region. As the amino acid sequence database grows, and as an increasingly larger fraction of the genome of a number of model organisms is sequenced (*E. coli*, *Saccharomyces cerevisiae*, *C. elegans*, etc.), the likelihood also increases that the protein products potentially encoded by newly obtained genomic DNA sequences will have a remote homologous already sequenced. Similarity, however, will not always exist or will often be too weak and partial to allow for the derivation of the genic structure of the genomic DNA sequences. (In fact, even when the exact amino acid product encoded by a given genomic sequence is known, elucidating its genic structure may not be trivial.) It thus appears that hybrid systems capable of integrating heterogeneous information from a variety of sources (database searches, sequence statistics, and signal identification) constitute the most successful approach to the problem of computational gene structure prediction in large genomic regions. Programs like GeneID+, GeneParser3, and BLASTC—a version of BLASTX (Gish and States, 1993) that uses knowledge of biases in codon frequency to score the alignments (States and Gish, 1994)—may constitute the first steps in such a direction. The use of database similarity searches, however, has the drawback of substantially increasing the analysis time. Translating the six frames of very long genomic sequences and comparing them against increasingly larger amino acid sequence databases is extremely time consuming and may even become prohibitive. It may be necessary to develop specialized similarity search software and specialized coding region databases that are suited to the problem of finding regions similar to known coding exons in large genomic sequences. The recently developed FINEX system (Brown *et al.*, 1995) is an example of such software. FINEX uses “exon fingerprints” instead of exon sequences, in the efficient detection of possible weak sequence homologies apparent at the genic structure level.

4.3. *Combining the Programs*

It may be of interest to investigate the extent to which different programs produce correlated predictions, that is, to investigate if they tend to fail and succeed in the same set of genes. Users may be interested in such information to select a suite of programs for maximal “coverage,” while developers may learn which strategies appear to be complementary. We thus computed the Approximate Correlation at the nucleotide level and the Average Exact Accuracy at the exon level between pairs of program predictions. The Approximate Correlation and the Average Exon Accuracy were computed for each sequence in the original ALLSEQ data set, as described in Section 2.2, but now the gene predictions of two programs were compared, rather than comparing one program’s gene prediction with the true gene structure of the sequence. Only programs not using database similarity searches were considered. In addition, Xpound was excluded when correlating performances at the exon level, given the poor performance of such a program at this level. Approximate Correlation was averaged over all sequences in the ALLSEQ data set, while Average Exon Accuracy was averaged only over those sequences for which it is defined. The resulting “Correlation Matrices” are shown in Tables 5a and 5b. Dendrograms were derived from such matrices using the UPGMA clustering method. All programs are moderately correlated to each other, and none behaves markedly different than the others. GeneID and GeneParser, if any, are the two programs that appear to behave more differently, while FGENEH, GenLang, GRAIL 2, and SORFIND tend to cluster together.

If specificity is a critical issue, it may be beneficial to combine the output of several programs. Indeed, we obtained the set of predicted exons in the ALLSEQ data set common to all the programs analyzed—with the exception of Xpound and the programs using amino acid similarity searches. One hundred seventy-four exons were predicted by all these programs, 172 of which corresponded exactly to actual exons annotated in GenBank, while the remaining 2 have a large overlap with actual exons. It thus appears that coincidence of several programs will reinforce a given prediction. (In fact, given the results obtained here, it might be claimed that when all programs predict exactly the same exon, such an exon is almost certainly true.)

Given the relatively high proportion of actual exons missed by the individual programs, it may be of interest to investigate if there exists a class of exons with such peculiar characteristics that they are undetectable by current methods. We, therefore, identified all actual exons missed completely by all programs—ME common to all programs. However, of the 2649 actual exons in the ALLSEQ data set, only 33 (about 1%) were missed completely by all the programs—programs using amino acid similarity searches were not considered—suggesting that no anomalous class of exons exists. These exons, moreover, do not seem to exhibit a uniquely distinctive feature, but rather a combination of peculiarities: they tend to occur in genes with a larger than average number of exons, to be shorter than average, and to be slightly A + T rich.

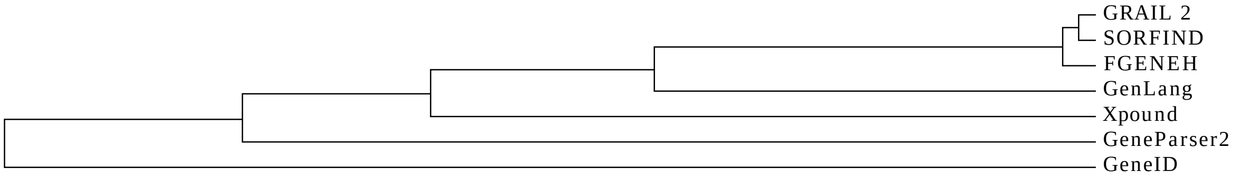
4.4. *Phylogenetic Specificity*

We investigated the phylogenetic specificity of the programs within vertebrates. Most of the programs studied here were designed to analyze human DNA sequences. Such programs did indeed show a decrease in accuracy when used to predict genes in sequences from vertebrate organisms other than human (Table 3, left). In general, however, the decrease in accuracy did not appear to be very important, and it could be mostly explained by the decrease in accuracy generally observed when programs are tested on data sets including sequences not closely related to those sequences in the data set on which the programs were trained (for instance, the decrease in accuracy in nonhuman sequences with respect to human sequences is not larger than that observed in the new sequences—sequences in NEWSEQ—with respect to all sequences—sequences in ALLSEQ). Moreover, no consistent pattern of variation in accuracy was observed as these programs analyzed sequences belonging to organisms increasingly divergent from human (Table 3, right); the variation in accuracy observed among the different GenBank divisions is likely to be mostly statistical. All these results suggest that sequence gene determinants remain essentially unchanged across the different vertebrate groups (at least across those represented in GenBank). Such a hypothesis is also supported by the absence of differences found by Dong and Searls (1994) between the mouse and the human gene grammars.

TABLE 5a

Pairwise Approximated Correlations between Program Predictions

	GeneID	GeneParser2	GenLang	GRAIL 2	SORFIND	Xpound
FGENEH	0.57	0.59	0.66	0.67	0.68	0.59
GeneID		0.51	0.54	0.53	0.54	0.49
GeneParser2			0.54	0.60	0.59	0.56
GenLang				0.61	0.62	0.53
GRAIL 2					0.68	0.67
SORFIND						0.61



Note. The Approximate Correlation at the nucleotide level has been computed between pairs of program predictions and averaged over all sequences in the ALLSEQ data set. "Correlation Matrices" are given for such measures. The dendogram was derived from the matrix using the UPGMA clustering method.

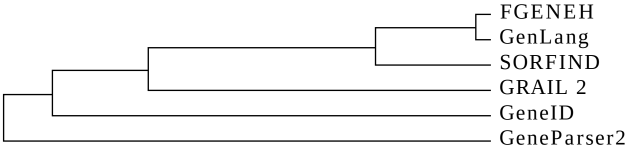
4.5. *Dependence on G + C content*

It has been previously reported that the accuracy of some gene prediction programs decreases on G + C-poor sequences (Xu *et al.*, 1994a; Snyder and Stormo, 1995a,b; Lopez *et al.*, 1994). We thus investigated the extent to which the program's performance was sensitive to G + C content. Table 4 shows the results of the separate analysis of program accuracy in the 15% G + C-poorest fraction of the ALLSEQ data set. As can be seen, a number of programs do appear to perform significantly worse in G + C-poor sequences—GeneParser2, GenLang, Grail, and Xpound, but this is not a general trend—thus GeneID performs only slightly better in G + C-rich sequences, while FGENEH and SORFIND appear to perform slightly better in G + C-poor sequences. The reason for such contradictory behavior remains unclear. It has recently been shown that coding statistics dependent on oligonucleotide frequencies are strongly correlated with G + C content (Guigó and Fickett, 1995). Since the coding regions known to date tend to be more G + C-rich than noncoding ones, the observed discriminative power of oligonucleotide frequency-derived measures may be partially a collateral consequence of their correlation with G + C content and not only a consequence of an intrinsic biological constraint. One would, therefore, expect that gene prediction programs that rely heavily on oligonucleotide frequency-derived measures are not as accurate in A + T-rich regions as in G + C-rich ones. Indeed, all the programs that perform worse in G + C-rich sequences do rely strongly on hexamer frequencies—a measure widely used because it appears to be highly discriminating between coding and noncoding regions (Claverie *et al.*, 1990; Fickett and Tung, 1992). However, FGENEH and SORFIND also rely on oligonucleotide frequency-derived measures (octanucleotides and codons, respectively), and their accuracy does not appear to depend on G + C content.

TABLE 5b

Pairwise Average Exon Accuracy between Program Predictions

	GeneID	GeneParser2	GenLang	GRAIL 2	SORFIND
FGENEH	0.34	0.32	0.47	0.34	0.43
GeneID		0.28	0.33	0.24	0.28
GeneParser2			0.30	0.24	0.27
GenLang				0.31	0.38
GRAIL 2					0.35



Note. The Average exact accuracy at the exon level has been computed between pairs of programs predictions and averaged over all sequences in the ALLSEQ data set. "Correlation Matrices" are given for such measures. The dendogram was derived from these matrices using the UPGMA clustering method.

Yet, the above results indicate that to explore regions of the genome of more extreme composition successfully, a new generation of coding statistics that are less sensitive to sequence properties other than codingness needs to be developed. Recent versions of GRAIL 2 and GeneParser have taken the first steps in such a direction, by calibrating the sequence statistics on which they rely at different values of G + C content. However, in accordance with the results obtained here, such steps appear still to be insufficient.

4.6. *Accuracy in Sequences with Errors*

The results discussed so far concern the accuracy of the programs in DNA sequences free of errors. Newly obtained sequences submitted to the programs, however, will often contain artifactual nucleotide insertions and deletions, since the most time- and cost-effective sequencing strategies tend to produce low-quality sequence data. Moreover, plans underway to accelerate the sequencing of the human genome are based on a relaxation of quality requirements of the sequences obtained. Sequencing of the human genome could initially proceed with an accuracy as low as 99.9 or even 99% (Marshall, 1995). To be of any practical use in the forthcoming large-scale sequencing phase of the genome projects, gene structure prediction programs should be robust to relatively high rates of DNA sequence errors. Table 2 summarizes the accuracy of the programs when 1% frameshift mutations were introduced in the sequences from the ALLSEQ dataset. Although this represented a very high rate of sequence errors, it is not unrealistically high for single-pass sequencing. As can be seen, the performance of all the programs suffered substantially with these data. Not surprisingly, those programs that assemble genes enforcing ORF compatibility between exons (such as FGENEH, GeneID, and GenLang) suffered more than programs predicting exons in isolation (such as GRAIL II and SORFIND), and these in turn suffered more than programs not using ORF information at all (such as GeneParser and Xpound). In addition, performance of the programs enforcing ORF constraints suffered proportionally more at the exon or protein level than at the nucleotide level. Yet, the results obtained indicate that strategies need to be developed to improve the robustness of most programs to sequence errors. One possibility is to drop ORF constraints, such as is done in GeneParser—which we found to be least sensitive to sequence errors. However, ORF compatibility between exons is intrinsic in the logic of most current programs that assemble exons into genes. A further possibility is to attempt to correct the sequences before submitting them to the gene identification programs. This is the approach taken in the most recent version of the GRAIL system (not available when performing the evaluation presented here). Indeed, Xu *et al.* (1995) developed an algorithm to detect and correct sequencing errors that occur in DNA coding regions and implemented it as a frontend subsystem of the GRAIL analysis programs. Fichant and Quentin (1995) have developed a similar system.

4.7. Dependence on Training Sets

The decrease in accuracy observed when the programs are confronted with sequences showing little similarity to their training sequences (Table 1) is perhaps one of the most interesting results obtained here. The decrease in accuracy is systematically suffered by all the programs, whatever the level of accuracy measured, and is often substantial. Such a decrease in accuracy indicates that the programs are overly dependent on the sequence particularities of the examples they learn from and not dependent enough on the universal sequence features involved in the determination of the genes. This may be the result of the programs relying more on sequence compositional statistics than on the sequence signals that are recognized and processed by the cellular machinery leading the pathway from DNA to protein sequences (promoter elements, splice sites, and start and stop codons, mainly). Indeed, the sequence functions used to compute coding statistics are often inferred to maximize the discrimination in known sets of coding and noncoding sequences. The fraction of sequences known so far, however, is unlikely to be representative of the genome as a whole, and the discriminating functions optimized in such a fraction may not generalize well when used on new unrelated sequence data. Sequence signal scoring functions, on the other hand, are less data dependent, and often they can even be inferred without resorting to known positive (and negative) examples. (For instance, a scoring function for splice sites could be derived from the sequence of the complementary RNA in the snRNPs binding to the sites.) Although gene identification methods relying on sequence signals would be more generalizable, our knowledge of the mechanisms by which such signals are recognized and processed is limited, and it has been implicitly assumed that attempting to elucidate the genic structure of genomic sequences by relying solely on sequence signals—without resorting to sequence statistics to prune the potential exons defined by the identified signals—would result in a computationally untreatable combinatorial explosion of potential products. Exhaustive exploration, nevertheless, is unlikely to be the cellular mechanism by which proteins are derived from DNA sequences, and the combinatorial explosion could be avoided by considering a more realistic model of this mechanism. An interesting line of research would, therefore, appear to be the exploration of the possibilities of predicting gene structure in genomic DNA sequences by relying only on the identification of sequence signals and by using some model—if only rudimentary—of how such signals are recognized and processed.

ACKNOWLEDGMENTS

This work was supported by a Beca d'Investigació Científica from the Generalitat de Catalunya BPOST93-20, by a Proyecto de la Dirección General de Investigación Científica y Técnica from the Ministerio de Educación y Ciencia (Spain) PB94-1278, and by Public Health Service Grant HG00981-01A1 from the National Center for Human Genome Research (USA). We thank Jim Fickett for discussions on the general problem of gene identification, Ferran Sanz for discussions on measures of prediction accuracy, and the anonymous referees for their helpful suggestions. We also thank Mark Borodovsky, Shan Dong, Misha Gelfand, Gordon Hutchinson, Steen Knudsen, Eric Snyder, Victor Solovyev, Alun Thomas, and Ying Xu for advice in using their programs.

REFERENCES

Altschul, S. F., Gish, W., Miller, W., Myers, E. W., and Lipman, D. J. (1990). Basic local alignment search tool. *J. Mol. Biol.* 215: 403–410.

Anderberg, M. R. (1973). "Cluster Analysis for Applications," Academic Press, New York.

Borodovsky, M. Y., and McIninch, J. D. (1993). GENMARK: Parallel gene recognition for both DNA strands. *Comput. Chem.* 17: 123–133.

Borodovsky, M. Y., Rudd, K. E., and Koonin, E. V. (1994). Intrinsic and extrinsic approaches for detecting genes in a bacterial genome. *Nucleic Acids Res.* 22: 4756–4767.

Brown, N. P., Whittaker, A. J., and Newell, W. R. (1995). Identification and analysis of multigene families by comparison of exon fingerprints. *J. Mol. Biol.* 249: 342–359.

Brunak, S., Engelbrecht, J., and Knudsen, S. (1991). Prediction of human mRNA donor and acceptor sites from the DNA sequence. *J. Mol. Biol.* 220: 49–65.

Claverie, J.-M., Sauvaget, I., and Bougueleret, L. (1990). k-tuple frequency analysis: From intron/exon discrimination to T-cell epitope mapping. *Methods Enzymol.* 183: 237–252.

Dayhoff, M. O. (1978). Survey of new data and computer methods of analysis. *In* "Atlas of Protein Sequence and Structure," Suppl. 5, Vol. 3, pp. 1–8, National Biomedical Research Foundation, Washington, DC.

Dodemont, H., Riemer, D., Ledger, N., and Weber, K. (1994). 8 genes and alternative RNA processing pathways generate an unexpectedly large diversity of cytoplasmatic intermediate filament proteins in the nematode *Caenorhabditis elegans*. *EMBO J.* 13: 2625–2638.

Dong, S., and Searls, D. B. (1994). Gene structure prediction by linguistic methods. *Genomics* 23: 540–551.

Fichant, G. A., and Burks, C. (1991). Identifying potential tRNA genes in genomic DNA sequences. *J. Mol. Biol.* 220: 659–671.

Fichant, G. A., and Quentin, Y. (1995). A frameshift error detection algorithm for DNA sequencing projects *Nucleic Acids Res.* 23: 2900–2908.

Fickett, J. W. (1982). Recognition of protein coding regions in DNA sequences. *Nucleic Acids Res.* 10: 5303–5318.

Fickett, J. W. (1996). The gene identification problem: An overview for developers. *Comput. Chem.* 20: 103–118.

Fickett, J. W., and Tung, C.-S. (1992). Assessment of protein coding measures. *Nucleic Acids Res.* 20: 6441–6450.

Fickett, J. W., and Guigó, R. (1996). Computational gene identification. *In* "Internet for the Molecular Biologist" (S. Swindell, R. Miller, and G. Myers, Eds.), Horizon Scientific Press, Oxford.

Fields, C. A., and Soderlund, C. A. (1990). gm: A practical tool for automating DNA sequence analysis. *Comput. Appl. Biosci.* 6: 263–270.

Fleischmann, R. D., Adams, M. D., White, O., Clayton, R. A., *et al.* (1995). Whole-genome random sequencing and assembly of Haemophilus Influenzae Rd. *Science* 269: 496–512.

Fraser, C. M., Gocayne, J. D., White, O., Adams, M. D., *et al.* (1995). The minimal gene complement of *Mycoplasma genitalium*. *Science* 270: 397–403.

Gelfand, M. S. (1990). Computer prediction of exon–intron structure of mammalian pre-mRNAs. *Nucleic Acids Res.* 18: 5865–5869.

Gelfand, M. S. (1995). Prediction of function in DNA sequence analysis. *J. Comput. Biol.* 1: 87–115.

Gelfand, M. S., and Roytberg, M. A. (1993). Prediction of the exon–intron structure by a dynamic programming approach. *Biosystems* 30: 173–182.

Gish, W., and States, D. J. (1993). Identification of protein coding regions by database similarity search. *Nature Genet.* 3: 266–272.

Guigó, R., Knudsen, S., Drake, N., and Smith, T. F. (1992). Prediction of gene structure. *J. Mol. Biol.* 226: 141–157.

Guigó, R., and Fickett, J. W. (1995). Distinctive sequence features in protein coding, genic noncoding, and intergenic human DNA. *J. Mol. Biol.* 253: 51–60.

Guo, W., Worley, K., Adams, V., Mason, J., *et al.* (1993). Genomic scanning for expressed sequences in Xp21 identifies the glycerol kinase gene. *Nature Genet.* 4: 367–372.

Hutchinson, G. B., and Hayden, M. R. (1992). The prediction of exons through an analysis of spliceable open reading frames. *Nucleic Acids Res.* 20: 3453–3462.

Kel, A. E., Ponomarenko, M. P., Likhachev, E. A., Orlov, Y. L., Ischenko, I. V., Milanesi, L., and Kolchanov, N. A. (1993). SITEVIDEO: A computer system for functional site analysis and recognition. Investigation of the human splice sites. *Comput. Appl. Biosci.* 9: 617–627.

Lopez, R. S., Larsen, F., and Prydz, H. (1994). Evaluation of the exons predictions of the GRAIL software. *Genomics* 24: 133–136.

Marshall, E. (1995). Emphasis turns from mapping to large-scale sequencing. *Science* 268: 1270–1271.

Milanesi, L., Kolchanov, N. A., Rogozin, I. B., Ischenko, I. V., Kel, A. E., Orlov, Y. L., Ponomarenko, M. P., and Vezzoni, P. (1993). GenViewer: A computing tool for protein-coding regions prediction in nucleotide sequences. *In* "Proceedings, 2nd Int. Conf. on Bioinformatics, Supercomputing and Complex Genome Analysis St. Petersburg, FL, July 1992" (H. A. Lim, J. W. Fickett, C. R. Cantor, and R. J. Robbins, Eds.), pp. 573–587, World Scientific, Singapore.

Milanesi, L., Kolchanov, N., Rogozin, I., Kel, A., and Titov, I. (1994). Sequence functional inference. *In* "Guide to Human Genome Computing." (M. J. Bishop, Ed.), pp. 249–312, Academic Press, London.

Myers, E. W., and Miller, W. (1988). Optimal alignments in linear space. *Comput. Appl. Biosci.* 4: 11–17.

Pearson, W. R. (1990). Rapid and sensitive sequence comparison with FASTP and FASTA. *Methods Enzymol.* 183: 63–98.

Pearson, W. R., and Lipman, D. J. (1988). Improved tools for biological sequence comparison. *Proc. Natl. Acad. Sci. USA* 85: 2444–2448.

Singh, G. B., and Krawetz, S. A. (1994). Computer based exon detection: An evaluation metric for comparison. *Int. J. Genome Res.* 1: 321–338.

Sneath, P. H. A., and Sokal, R. R. (1973). "Numerical Taxonomy," Freeman, San Francisco.

Snyder, E. E., and Stormo, G. D. (1993). Identification of coding regions in genomic DNA sequences: An application of dynamic programming and neural networks. *Nucleic Acids Res.* 21: 607–613.

Snyder, E. E., and Stormo, G. D. (1995a). Identification of protein coding regions in genomic DNA. *J. Mol. Biol.* 248: 1–18.

Snyder, E. E., and Stormo, G. D. (1995b). Identifying genes in genomic DNA sequences. *In* "Nucleic Acid and Protein Sequence Analysis: A Practical Approach." 2nd ed., IRL Press, Oxford.

Solovyev, V. V., Salamov, A. A., and Lawrence, C. B. (1994). Predicting internal exons by oligonucleotide composition and discriminant analysis of spliceable open reading frames. *Nucleic Acids Res.* 22: 5156–5163.

States, D. J., and Gish, W. (1994). Combined use of sequence similarity and codon bias for coding region identification. *J. Comput. Biol.* 1: 39–50.

Thomas, A., and Skolnick, M. H. (1994). A probabilistic model for detecting coding regions in DNA sequences. *IMA J. Math. Appl. Med. Biol.* 11: 149–160.

Uberbacher, E. C., and Mural, R. J. (1991). Locating protein-coding regions in human DNA sequences by a multiple sensor-neural network approach. *Proc. Natl. Acad. Sci. USA* 88: 11261–11265.

Xu, Y., Einstein, J. R., Mural, R. J., Shah, M., and Uberbacher, E. C. (1994a). An improved system for exon recognition and gene modeling in human DNA sequences. *In* "ISMB-94 Proceedings Second International Conference on Intelligent Systems for Molecular Biology" (R. Altman, D. Brutlag, P. Karp, R. Lathrop, and D. Searls, Eds.), pp. 376–384, AAAI Press, Menlo Park.

Xu, Y., Mural, R. J., and Uberbacher, E. C. (1994b). Constructing gene models from accurately predicted exons: An application of dynamic programming. *Comput. Appl. Biosci.* 10: 613–623.

Xu, Y., Mural, R. J., and Uberbacher, E. C. (1995). Correcting sequencing errors in DNA coding regions using a dynamic programming approach. *Comput. Appl. Biosci.* 11: 117–124.