 Human Molecular Genetics, 1997, Vol. 6, No. 10 Review 1735–1744

Computational methods for the identification of genes in vertebrate genomic sequences

Jean-Michel Claverie

Structural and Genetic Information Laboratory, CNRS-EP.91, 31 chemin Joseph-Aiguier, 13402 Marseille cedex 20, France

Received May 19, 1997

Research into new methods to identify genes in anonymous genomic sequences has been going on for more than 15 years. Over this period of time, the field has evolved from the designing of programs to identify protein coding regions in compact mitochondrial or bacterial genomes, to the challenge of predicting the detailed organization of multi-exon vertebrate genes. The best program currently available perfectly locates more than 80% of the internal coding exons, and only 5% of the predictions do not overlap a real exon. Given such accuracy, computational methods are indeed very useful; however, they do not alleviate the need for experimental validation. If the performances are satisfactory for the identification of the coding moiety of genes (internal coding exons), the determination of the full extent of the transcript (5′ and 3′ extremities of the gene) and the location of promoter regions are still unreliable. As the human and mouse genome sequencing projects enter a production mode, the fully automated annotation of megabase-long anonymous genomic sequences is the next big challenge in bioinformatics.

INTRODUCTION

Computational methods for identifying genes in genomic DNA sequences have been an active field of research for 15 years, enjoying the calm and obscurity of confidential bioinformatics circles. As the human and mouse genome projects enter a phase of systematic sequencing, reliable automated techniques for interpreting long anonymous genomic sequence (i.e., partitioning them into genes, promoters, regulatory elements, intergenic region, etc.) are suddenly needed. As a consequence, bioinformatics, and the problem of gene finding have been attracting a lot more attention in recent years (1). At the time of writing of this article, ~45 Megabases (Mb) of human genomic sequence are finished, a further 100–150 Mb should be sequenced in 1998, and then 300–500 Mb in each of the following years, until completion (3000 Mb). On the mouse front, systematic sequencing should start with 20 Mb in 1998, and progressively increase up to 500 Mb a year, provided adequate financing is found. If current experimental methods are adequate for characterizing sequences of a few hundred kilobases (kb) at loci of special interest (e.g., disease genes), it is clear that they cannot be systematically used to 'annotate' multi-megabase-long anonymous sequences. If the human genome sequence data is to be exploited, computational methods are the only alternative that can be used to provide a minimal amount of characterization, either in an automated or semi-automated way.

Among the large number of programs and methods currently available, surprisingly few are known in the molecular geneticist community. The first and main purpose of this article is to make non-specialists aware of the diversity of programs (listed in Table 1) that have been proposed to locate and analyze genes in vertebrate genomic sequences.

The second purpose of this article is to provide some background information about the principles on which the different categories of programs are based. A minimal grasp of these principles is necessary to understand which program will work best for a certain type of data (e.g., genome survey versus finished contig), or which programs can be usefully combined for improved predictions.

Before entering the subject, the two main concepts governing the measure of prediction accuracy have to be introduced. First, it should be clear that it is trivial to design a method capable of predicting 100% of all internal exons, whether coding or non-coding, in the human genome: retaining all the segments flanked by AG and GT, will do it. Of course, such a method is useless, as it produces many more false predictions (chance occurrence of splicing sites) than real ones. This method has 100% sensitivity, but near 0% specificity. Requiring the splice site to adhere to a stronger consensus and the candidate exon sequence to obey additional rules [e.g., to contain an open reading frame (ORF)] will certainly increase the specificity, but immediately decrease the sensitivity; for instance, non-coding internal exons will no longer be detected. The development of sequence analysis methods has always been a struggle to keep both sensitivity (Sn) and specificity (Sp) to an acceptable level. For this reason the accuracy of methods, including those predicting genes, is best expressed as the average of the two: (Sn + Sp)/2. In general, authors adjust their program parameters so as to obtain Sn ≈ Sp.

Of the most recent review articles published on the subject of gene identification one can cite an overview by Fickett (2), and two more technical articles by Fickett (3) and Gelfand (4). We must also cite the landmark comparative study by Burset and Guigo (5). Finally, Li (6) and Gelfand (7) are maintaining very useful bibliographies in electronic form.

Tel: +33 4 9116 4548; Fax: +33 4 9116 4549; Email: jmc@igs.cnrs-mrs.fr

Table 1. Contact addresses and availability of the programs cited in this article

Program (ref)	Electronic address	Type of access[a]
GeneID (57)	geneid@darwin.bu.edu	ES
	www.imim.es/GeneIdentification/Geneid/geneid_input.html	HP
GeneParser (63)	beagle.colorado.edu/~eesnyder/GeneParser.html	HP, EX
Genie (71)	www-hgc.lbl.gov/inf/genie.html	HP, WS, ES
GenLang (58)	www.cbil.upenn.edu/~sdong/genlang_home.html	HP, WS, SC
GENSCAN (72)	gnomic.stanford.edu/GENSCANW.html	HP, WS, ES
GENVIEW (65)	www.itba.mi.cnr.it/webgene	HP, WS
GRAIL (66)	avalon.epm.ornl.gov	HP, ES, CL
HEXON/FGENEH (59)	dot.imgen.bcm.tmc.edu:9331/gene-finder/gf.html	HP, WS, ES
MORGAN (–)	www.cs.jhu.edu/labs/compbio/morgan.html	HP, WS, EX
MZEF (52)	clio.cshl.org/genefinder	HP, WS, EX
ORFgene (75)	www.itba.mi.cnr.it/webgene	HP, WS
PROCRUSTES (74)	www-hto.usc.edu/software/procrustes/index.html	HP, WS, ES, EX
SorFind (24)	www.rabbithutch.com	HP, EX
VEIL (70)	www.cs.jhu.edu/labs/compbio/veil.html	HP, WS, ES, EX
Xpound (51)	ftp://igs-server.cnrs-mrs.fr/pub/Banbury/xpound	SC
Banbury Cross	igs-server.cnrs-mrs.fr	HP

Except for GeneID (E-mail server) and Xpound (ftp site), the addresses have to be invoked with the 'http://' prefix.
[a]HP, home page; ES, E-mail server (sequences are sent by E-mail); WS, web server (sequences are pasted in an interactive window); CL, client/server protocol (part of the program is run on the calling machine (Xgrail); EX, executable code is available; SC, source code is available (file transfer by ftp). The Banbury Cross home page (a gene identification software benchmark site) maintains an up-to-date list of gene structure prediction programs and sites.

GENE FINDING: A QUICK HISTORY OF THE METHODS AND CONCEPTS

Most of the early sequence data were obtained from mitochondrial or bacterial genomes. Accordingly, computer methods to identify genes were first developed in that context. Bacterial protein coding regions consist of contiguous open reading frames (ORF). On a pure statistical basis, one ORF (ATG to stop) longer than 300 bp (100 residue protein) is expected to randomly occur every 36 kb on a single strand of DNA (with %A = %C = %G =%T =25) (8). Real proteins correspond on average to a 1000 bp ORF (330 residues). A simple algorithm retaining the longest overlapping ORFs and applying a size threshold (for instance 300 bp) will already detect most real genes, with good specificity. Thus, more sophisticated methods are only needed to locate small genes, interpret partial sequences (e.g. genome survey data) containing incomplete ORFs, or overcome sequencing errors.

The codon usage statistics were first introduced for this purpose by Staden and McLachlan in 1982 (9). The method simply consisted of scanning the DNA sequence and measuring the strength of codon preference within successive windows. At the same time, Shepherd (10) and Fickett (11) proposed other methods that take advantage of the compositional bias between codon positions. As more genomic sequence data became available for higher eukaryotes and vertebrates, it was clear that reliably discriminating between exons and introns would require much more sophisticated methods. Vertebrate protein coding genes consist of six exons spanning ~30 kb on average. However, there is a wide variety in size and complexity, and 'atypical' genes are not rare. In the 'monster gallery' of genes, one can cite dystrophin, a gene spanning 2.4 Mb (reviewed in 12), and the gene for blood coagulation factor VIII, spanning 186 kb with 26 exons ranging in size from 69 to 3106 bp, and introns as large as 32.4 kb. A CpG island in one of them (intron 22) initiates two transcripts: a nested transcript in the same orientation, and another one from the opposite strand (reviewed in 13). Other genes have unusual extremities, such as those of the MAGE family (14), with a 5′ untranslated region (UTR) spanning several internal exons, or the Kallmann syndrome gene with a 4 kb 3′ UTR (15).

The mean internal coding exon size is 150 bp. This is a very short segment on which to base a detection procedure. By chance alone, ORFs (stop to stop) longer than 225 bp are expected to randomly occur every kb on a single strand of DNA (8). Thus, ORF size can no longer be a useful criteria for locating protein coding regions. The challenge of identifying the short and sparse vertebrate coding regions prompted the development of new statistical methods to estimate the coding potential of arbitrary genomic sub sequences.

A wide variety of protein coding measures (reviewed in 16) were proposed and applied to the analysis of genomic sequences. The amount of sequence data available led to the discovery that exons and introns exhibit a distinct usage of nucleotide 'words' (17,18). This global property probably results from the combination of codon preference with other characteristic periodicities (19,20). The contrast in the usage of six nucleotide words (hexamers) (17,21) was found to be the best single property to predict whether a window of vertebrate genomic sequence was coding or non-coding (16,22). The accuracy of the best coding measure was ~70% (i.e., 1/3 of the coding exons were

missed, and 1/3 of the ones predicted are not real) for coding windows of at least 50 nucleotides in length.

With little prospect of finding better coding measures, scientists in the field began to try various combinations of the existing methods, hoping to improve the overall accuracy of predictions. A straightforward, but effective, way of implementing this concept was through a visual interface, simultaneously displaying graphical representations of the selected coding measures as well as 'signal' information (such as start/stop codons and splice sites). This approach was pioneered by Staden (23). Legouis *et al.* (15) used a semi-automated protocol to successfully identify the gene for Kallmann syndrome from a 67 kb genomic contig containing only two internal exons (141 + 222 coding nucleotides). The protocol combined: (i) the selection of all ORFs larger than 50 bp and flanked by reasonable consensus acceptor and donor splice sites; (ii) ranking the candidate exons according to the hexamer coding measure; and (iii) scanning the candidate exons for similarity against protein sequence databases. A very similar protocol (ORFs flanked by AG/GT are ranked according to their coding potential and splice site strength) was formally integrated in the SorFind program (24). In an independent test (5) SorFind predicted 71% of the coding nucleotides, with a specificity of 85%. Similar performances (25,26) were reached by GRAIL I (25) using a neural network to combine multiple coding measures but disregarding splice site information. GRAIL I, the first exon prediction program readily accessible through an E-mail server, enjoyed a tremendous success within the community of molecular geneticists, and the program marked the entry into the modern era for gene identification software. Thanks to GRAIL, biologists became aware of computational prediction methods and began to trust them. It also prompted computer scientists to explore increasingly sophisticated ways of combining sequence analysis techniques, as well as to pay more attention to the ease of use and accessibility of their programs.

EXON FINDING BY SIMILARITY SEARCH

As the above developments in statistical gene finding methods were taking place, new sequence data accumulated exponentially in the GenBank/EMBL/DDBJ databases (27,28). It became increasingly likely that protein coding exons could be simply recognized by a similarity search against the whole translated database (29,30). About 50% (31,32) of all vertebrate genes have retained enough similarity with their pre-metazoan ancestors to exhibit a significant BLASTX (30) match in a database containing the whole yeast genome, several complete bacterial genomes, and most of *Caenorhabditis elegans*. The similarity search approach received a tremendous boost from the large scale sequencing of Expressed Sequence Tags (EST) (33–35).

Instead of having to detect exons through borderline similarities with distant homologues, we are now in a position to look for exact matches. More than 50% of all human genes might already have a cognate public EST (36–38). However, the direct comparison of large vertebrate genomic sequences and EST data is prone to artifacts and computationally intensive. A very large number of informative matches are due to the presence of ubiquitous SINES and LINES repeats in vertebrate genomic sequences [up to 36%; (39)]. It is thus imperative to carefully filter the genomic sequence query prior to using it to scan an EST database. A flexible protocol involves pre-scanning against a specific 'junk' database as well as a small database of simple (i.e., microsatellite) repeats with a standard similarity search program, and masking out the matching nucleotides in the query (40–43). Specialized programs are also available for this purpose (44,45).

Except in the rare cases where the complete sequence of a cDNA or a homologous protein is already in the database, similarity searches do not usually identify the entire gene. Due to the modular structure of vertebrate proteins and the conservation of functional motifs, protein databases similarities tend to only reveal a small subset of the coding exons. On the other hand, EST matches most often only identify the 3′ end (coding or non-coding) exons.

The positive results from similarity searches can also be used as accessory evidence to reinforce exon predictions made from signal-based or statistical methods. This was done manually prior to the development of integrated software. The GRAIL II/GENQUEST system was first to introduce an option to run *a posteriori* database searches on the predicted exons (46). The majority of current programs in use have the capacity to incorporate database similarity search information in their gene prediction scheme (see below).

FROM FINDING INDIVIDUAL EXONS TO PREDICTING COMPLETE GENE STRUCTURES

Besides the compositional bias imposed by the constraints of protein coding, vertebrate exons are also characterized by sequence 'signals'. Internal (coding and non-coding) exons are bracketed by acceptor and donor splice sites, 5′ exons must lie immediately downstream to a core promoter site (e.g., a TATA-box) and eventually contain a translation start site (e.g., ATG), and 3′ exons should contain a polyadenylation signal and eventually a stop codon. No exon prediction method can solely be based on detecting these signals, because of their very low information content (47) and/or their lack of statistical significance (48).

However, important progress in automated gene identification has come from the combining of statistical/compositional techniques with signal detection methods into a single framework. For instance, the prediction of individual internal coding exons significantly improves when measures of the coding potential are associated with the strength of the flanking sites such as in SorFind (24), HEXON (49,50), Xpound (51), GRAIL II (46), or the latest MZEF (52). According to its documentation, GRAIL II finds 91% of all coding nucleotides, with a performance independent of exon size, and a false positive rate of 8.6%. In a later independent testing using larger genomic sequences (16) those numbers became 71% and 30%, respectively. This illustrates a general trend; the performances estimated in independent benchmark studies tend to be lower than initially published. Most of the performances summarized in Table 2 are extracted from the work of Burset and Guigo (5).

The ultimate task of gene identification programs is to generate a complete gene model including the correct assembly of the individual internal exons and recognition of the 5′ and 3′ extremities of the transcript. Most programs to date have limited their goal to the detection and assembly of the whole protein-coding moiety: from the ATG of the first coding exon, through all internal coding exons, to the stop codon of the last coding exon. The performances cited in this article mostly concern the identification of internal coding exons.

Table 2. Estimated performances of the various programs

Program	Original ref.	Test ref.	Prediction type	Sensitivity (%nucl.)	Specificity (%nucl.)	Sensitivity (%exact exon)	Specificity (%exact exon)	Missed exons	Wrong exon
FGENEH	59	52	Gene structure	83	93	73	78	15	11
GeneID	57	5	Gene structure	69	77	42	46	28	24
GeneParser	63	5	Gene structure	66	79	35	40	29	17
Genie	71	71	Gene structure	87	88	69	70	10	15
GenLang	58	5	Gene structure	72	79	51	52	21	21
GENSCAN	72	72	Gene structure	**93**	**93**	**78**	**81**	**9**	**5**
GRAIL II	46	52	Internal exons	79	85	51	57	25	28
GRAIL II/GAP	66	63	Gene structure	83	87	–	52	25	10
HEXON	50	52	Internal exons	88	80	71	65	10	27
MORGAN	–	–	Gene structure	83	79	58	51	14	–
MZEF	52	52	Internal exons	**87**	**95**	**78**	**86**	**14**	**7**
SorFind	24	5	Internal exons	71	85	42	47	24	14
VEIL	70	70	Gene structure	83	72	53	49	19	–
Xpound	51	5	Internal exons	61	87	15	18	32	13

We listed: (i) the best performance cited in an independent study or, (ii) the worse performance cited by the authors about their own program. Test sets vary in size, complexity or (G+C) composition. Performances given here should be interpreted with caution. Any program can behave better or worse against a given sequence or a new data set.
Definition of the data columns:
Sensitivity (nucl.): % of the actually coding nucleotides been predicted as coding.
Sensitivity (exon): % of the actual coding exons been predicted exactly right (both splice junctions).
Specificity (nucl.): % of the nucleotide predicted as coding been actually coding.
Specificity (exon): % of the predicted exons perfectly matching an actual exon.
Missed exons: % of actual exons not overlapping any prediction.
Wrong exons: % of predicted exons not overlapping any actual exon.
The best overall performers: MZEF (individual exon finder) and GENSCAN (gene structure prediction) are shown in bold.

In 1990, Fields and Soderlund (53) and Gelfand (54) pioneered the field of whole gene structure prediction. The difference with the previous problem of detecting individual exons is that the predicted exons now have to fit and be assembled into a coherent gene model.

Years of research have now resulted in many different programs (Tables 1 and 2). Despite their diversity, most of them use the 'combinatorial approach'. They first generate a set of candidate exons using a combination of coding measures and splice site quality scores, or other specific signals for 5′ exons (TATA-box, initiator ATG, etc.) and 3′ exons (stop and poly-adenylation site). The resulting set of candidate exons are then assembled to construct candidate gene structures, the best of which is finally chosen as the most likely prediction.

A straightforward implementation of the combinatorial approach encounters both conceptual and computational problems. First, converting all the parameters associated with the various components of a given gene model (coding measures, signal strength, exon length, etc.) into a meaningful unique quality index is not trivial. Second, the number of different assemblies of exons and signals consistent with legitimate gene models of realistic sizes (30–250 kb) can be huge. Sophisticated algorithms have to be designed to solve this combinatorial problem and find the best gene model(s) in a reasonable amount of time. The diversity of the current gene structure prediction programs (listed in Table 1) attests that many ways have been tried to solve the above problems.

The significant differences between these programs reside in: (i) the methods used to combine the recognition of the individual components; (ii) the ways used to estimate the 'quality' of concurrent gene models; and (iii) the algorithms used to extract the optimal gene model(s) and deal with the combinatorial complexity. Some rules, like the one enforcing that protein translation must proceed through the chain of internal exons ('in frame assembly') may concern all three aspects; by dividing the large pool of candidate exons into compatible subsets, the added constraint strongly reduces the number of putative gene models, enhances their *a priori* quality, and can even improve the prediction of individual exons (55,56).

The detailed presentation of the various 'gene parsing' and gene scoring methods behind each program would be rather technical, and beyond the scope of the present review. In the section below, a few key concepts are simply mentioned and associated with various programs. The interested reader will find more details in the original sources. The relative sizes of the paragraphs are somewhat proportional to the impact of the concept in the field.

Rule-based systems

GeneID (57) starts by identifying first, internal and last exons on the basis of coding measures and signal strength, and uses a heuristic, rule-based system to assemble these into models of

ONE likely gene in each sequence. Typically, GeneID evaluates tens of thousand of gene models. In a comparative study (5) using a data set of 556 genes, GeneID predicted 44% of exons exactly, with a specificity of 45% (Table 2).

Linguistic methods

GenLang (58) does not use the combinatorial approach but interprets the usual coding measures and signal strengths in a linguistic context as 'leaf rules' associated to a cost. A formal grammar, optimized on a training set, is then used to generate a gene model as the parse that minimizes the total cost. GenLang performances are listed in Table 2.

Linear discriminant analysis (LDA)

LDA is a standard technique in multivariate analysis that can be used to linearly combine several measures in order to perform the best discrimination between two functional classes of sequences. It can also serve to identify the most significant measure for a given discrimination problem. Building on a suggestion by Fickett (16), Solovyev and collaborators used a linear discriminant function to combine information about significant preferences of oligonucleotides in DNA sequences of different function (5′, internal, 3′ exons). The approach is implemented in the HEXON and FEX programs (50). In FGENEH (59), they then apply dynamic programming (see below) to predict optimal gene models from the list of potential exons. HEXON and FGENEH performances are listed in Table 2.

Decision tree

MORGAN (60), an integrated system for finding genes, uses a variety of techniques, the most distinctive of which is a 'decision tree' algorithm. Well established machine learning techniques, decision tree classifiers have been introduced by Salzberg (61) for solving the simpler problem of discriminating coding and non-coding DNA. The internal nodes of a decision tree are property values that are tested for each sub sequence passed to the tree. Properties can be various coding measures (e.g., hexamer frequency) or signal strengths. The bottom nodes (leaves) of the tree contains class labels to be finally associated with the sub sequence. Once classified, the various components are assembled into an optimal gene model using a dynamic programming approach (see below). MORGAN performances (kindly communicated by Dr Salzberg prior to publication) are given in Table 2.

Dynamic programming

Briefly, the dynamic programming algorithm (reviewed in 62) is a well established recursive procedure for finding the optimal (e.g., minimal cost or top scoring) pathway among a series of weighted steps. GeneParser (63,64) uses coding measures and signal strengths to compute scores for all subintervals in the test sequence. A neural network is first used to combine the various measures into the log-likelihood ratio for each subinterval to exactly represent an intron or exon. A dynamic programming approach is then used to find the optimal combination of introns and exons. Ranked sub optimal solutions can also be generated by the program. The performances of GeneParser are listed in Table 2. Gelfand and Roytberg (65) have reviewed the use of dynamic programming in gene prediction, and suggested 'vector dynamic programming' to combine multiple exon quality indices without the time-consuming training of a neural network. Those ideas have been implemented in CASSANDRA, a program to predict protein-coding segments, and the experimental gene structure prediction program GREAT (4). The GENVIEW system (66) is again based on the prediction of spliceable ORFs ranked by the strength of their splice signal and their coding potential ('in phase' hexamer measure). The best gene structure is then constructed using dynamic programming to sift through the numerous possible exon assemblies. Finally, the gene assembly program GAP III also uses dynamic programming (as well as heuristics) to construct optimal gene models from the candidate exons predicted by GRAIL II (67). The performances of the GRAIL II/GAP system are listed in Table 2.

Markov models

Biological sequences can be modeled as the output of a stochastic process in which the probability for a given nucleotide to occur at position p depends on the nucleotide occupying the k previous positions. Such a representation is called a k-order Markov model. Different functional domains of a sequence (e.g., coding versus non-coding regions) exhibiting different statistical properties (e.g., dinucleotide frequency or 3-periodicity) will correspond to different Markov models. Parsing a natural biological sequence into non-coding versus coding region, thus simply consists in determining if a given region is more likely to be generated by the coding versus the non-coding Markov models (previously built using training sets). Such a procedure is the basis for GenMark (68), an efficient program for finding genes in bacterial genomes. Given the more complex structure of vertebrate genes, many Markov models are needed to capture the information within exons, introns, intergenic regions, splice junctions and other 5′ and 3′ signals. It then becomes more convenient to represent the sequence as the output of an abstract process that progresses through a series of discrete states some of which are 'hidden' from the observer. This is referred to as the Hidden Markov Model (HMM) approach. HMMs, and their use in computational biology, have already been reviewed (69). ECOPARSE (70), a gene finder for *Escherichia coli*, introduced the use of HMMs in gene recognition. The VEIL program (71) uses an HMM system for segmenting anonymous vertebrate sequences into exons, introns and intergenic regions. At a further level of abstraction, Generalized Hidden Markov Models (GHMMs) are HMMs where states are arbitrary sub models (e.g., neural networks, position weight matrices, etc.) outputting variable length sequences (i.e., 'states' can have variable durations). GENIE (72) introduced GHMMs in the context of gene structure prediction. More recently, Burge and Karlin (73) introduced a general probabilistic model of gene structure with a similar architecture, implemented in GENSCAN. In contrast with previous works, the authors of GENSCAN devoted a lot of attention to the optimization of the lower level modules performing the recognition of the basic signals (e.g., transcriptional, translational and splicing signals), and incorporated the influence of (C+G) content. GENSCAN explores possible gene models on both DNA strands simultaneously and is capable of parsing sequences containing multiple (eventually embedded) genes. The performances of VEIL, GENIE and GENSCAN are listed in Table 2. VEIL, GENIE and GENSCAN also use a variation (74)

of the dynamic programming algorithm to find the most likely gene structure by optimally aligning the sequence to their respective HMM systems.

'Spliced alignment'

The 'bed of Procrustes' or 'procrustean bed', proverbial for 'arbitrarily forcing someone or something to fit into an unnatural scheme or pattern', apparently inspired an original algorithm by Gelfand *et al.* (74). Their program PROCRUSTES provides an integrated procedure to use protein (and cDNA) similarity information to identify genes and predict gene structure. Given a genomic DNA sequence, the program first generates a set of candidate exons. These candidates consists of all sub sequences between candidate acceptor and donor splice sites, with very little filtration to minimize the risk of losing true exons. PROCRUSTES then considers all possible chains of candidate exons and finds a chain with the maximum global similarity to the target protein. Even though the number of exon assemblies is huge, the 'spliced alignment algorithm' is fast enough to process large genomic fragments (up to 180 000 nucleotides) containing multi-exon genes (>30 exons). If a protein sufficiently similar to the one encoded in the analyzed sequence is available, the highest-scoring exon assembly very often represents the correct exon–intron structure. According to the original study (75), the average correlation coefficients for non-primate mammalian, bird, plant and fungal targets are, respectively, 98, 96, 95 and 93%. For target proteins with similarity scores above 60% the average correlation coefficient is 99%. The basic idea of using protein homology as a guide to predict exon structure was also proposed by Rogozin *et al.* (76) and implemented in ORFgene (Table 1). This program lacks the very efficient spliced alignment algorithm that characterizes PROCRUSTES.

DISCUSSION

Most of the programs and methods that have been presented here share a number of limitations. They will be briefly discussed below.

Current methods only detect protein coding genes

The performances listed in Table 2 correspond to the prediction of protein coding regions, that is: (i) the coding moiety of the first exon (from ATG to the first donor splice site); (ii) the internal coding exons; and (iii) the coding moiety of the last coding exon (from the acceptor splice site to a stop codon. No reliable methods are presently available for predicting the non-coding part of genes, i.e., the 5′ and 3′ UTRs. As a consequence, non-coding RNA genes, such as XIST (77), H19 (78), IPW (79) and the newly discovered NTT (80) would have been totally transparent to the current gene prediction programs. In the absence of a method to identify them, it is impossible to estimate how many genes of this type are hidden in the human genome, although they might constitute an essential regulatory component of its expression. XIST, H19 and IPW are all known to play a key role in transcription inactivation and/or imprinting.

By most statistical measures used to date, non-coding parts of genes do not differ much from intron or intergenic sequences. With no statistical measure at hand, we are left to look for the sequence signals supposed to bracket transcription units: the core promoter region in 5′, and the polyadenylation site in 3′.

Only two core promoter elements are located at a fixed distance from the transcription start site: an (A+T)-rich sequence (the so-called TATA-box) positioned some 30 bp upstream (reviewed in 81–83), and the initiator element (Inr, reviewed in 84). Between 70 and 80% of promoters contain a TATA box. Given their variability and ubiquity, those signals do not contain enough information to specifically locate the 5′ end of genes. The difficulty of accurately predicting the location of vertebrate promoters has been well documented in recent reviews (2,85–87).

At the other extreme, the AATAAA polyadenylation signal is supposed to end transcription. This short signal is again ubiquitous and does not contain enough information by itself to specifically locate the 3′ extremity of genes. Moreover, we found it missing from 54% of the 3′ end of transcripts, as estimated from our survey (Audic, Gautheret, Seilhamer and Claverie, unpublished) of all Merck/Washington University 3′ EST sequences (36,37). The fraction is approximately the same in complete mRNA sequences in GenBank. In the absence of the canonical AATAAA signal, no variations over the consensus (with the possible exception of ATTAAA) stand out in a statistically significant manner. Thus, one can anticipate that ~50% of vertebrate genes will have a particularly difficult 3′ end to map with precision by lack of a clear signal.

In summary, without the help of a strong statistical bias as exhibited by coding regions, a pure 'signal' analysis of vertebrate genomic sequences is unable to identify non-protein coding genes, or the precise 5′ and 3′ extremities of protein coding genes. Non-coding RNA genes, or the 3′ UTRs of regular protein genes can only be located by similarity searches if they correspond to an EST. Cases where the current programs *do not predict* any exons upstream from a perfect match with an EST are, in fact, suggestive of non-coding RNA genes.

Finally, efficient programs to detect tRNA genes (88), or any family of RNA genes (89) with a specific sequence or secondary structure signature have been available for some time.

Most current methods only detect one typical gene

With the exception of the recent GENSCAN (and the interactive XGRAIL system), all gene structure prediction programs assume that the input genomic sequence contains a single complete gene. The programs enforce solutions including a gene 'beginning' and a gene 'end'. The predictions made on sequences containing a partial gene, or multiple genes, do not usually make sense. Single-exon genes are also not well predicted by most programs. GENSCAN incorporates the concept of partial genes, multiple genes and single-exon genes in its probabilistic model of gene structures. The model is also 'double-stranded', i.e., potential genes occurring on both DNA strands are analyzed simultaneously and have to be compatible. However, at the moment, different genes (on the same or opposite strand) must be separated by an 'intergenic region'. Thus, cases of overlapping transcription units such as nested genes (13), or a gene embedded in an intron of another gene (13), are not yet considered by the program. Fortunately, those situations are probably rare in vertebrate genomes. A much more common situation, alternatively spliced transcripts, is not yet adequately handled by any program.

Another serious limitation is that all programs have been trained (and their performance assessed) on a rather special subset of vertebrate genes, with relatively few exons spanning no more

than a few kilobases. In the reference test set of Burset and Guigo (5), the average gene length is 5.1 kb, and the average number of exons is 4.6. It is thus feared that the accuracy of the current programs (Table 2) will be considerably lower when analyzing contigs of hundreds of kilobases now currently generated in genome sequencing laboratories. An independent evaluation (26) of the GRAIL software on larger (15–101 kb) sequences confirmed a significant, but not alarming, decrease in performance. More recently, Ansari-Lari *et al.* (90) have systematically combined experimental tools (RT-PCR and cDNA sequencing), various gene prediction methods (GRAIL II, FGENEH and GeneID), and similarity searches to analyze a 223 kb genomic sequence in the CD4 region (chromosome 12p13). The most important parameter, the fraction of totally missed exons (ME in Table 2) was established at 22% for Grail II and FGENEH, and at 54% for GeneID. False positive rates were of the order of 17, 12 and 27% for Grail II, FGENEH and GeneID, respectively. Except for the large fraction of missed exons cited for GeneID, these numbers are not dramatically worse than those listed in Table 2. On an *a posteriori* analysis of a 117 kb fragment of the same genomic sequence (91), more than 60 of the known exons in the region overlapped GENSCAN predictions (100% exact for internal exons), and the rate of false positives was close to zero (73). At the level of accuracy now reached, the interpretation of 'false positives' (i.e., exons predicted but not experimentally validated) becomes difficult, as they may correspond to pseudo genes or true genes not yet experimentally identified.

Finally, it is important to notice that, out of 570 test genes, 243 (43%) were perfectly predicted at the coding level by GENSCAN (73). This is a very encouraging result, demonstrating that it is indeed possible to predict multi-exon gene structures using an entirely automated procedure, that is to automatically 'annotate' anonymous genomic sequences. Some successes are truly impressive such as the perfect assembly of the 22 coding exons of the human gastric ATPase (GenBank accession no. J05451). On the other hand, predictions of gene boundaries are still inaccurate. For instance, while GENSCAN predicted 100% of the internal exons in the 117 kb CD4 regions (91), none of the six genes found in the region were correctly mapped from beginning to end, and two were fused together (73). A specific web site (the 'Banbury Cross', named after a Banbury meeting where the concept originated) is now devoted to the study of the performance of gene finding algorithms in the context of very long genomic sequences (see Table 2).

All current methods are conservative

A more conceptual limitation of current gene/exon prediction methods is their implicit conservatism. Indeed, homology-based methods such as simple database similarity searches (29,30), or the more integrated PROCRUSTES program (75), are by construction unable to discover 'new' genes, i.e., with no significant resemblance to previously encountered ones. To a lesser extent, this is also true of all other gene prediction methods. All the programs listed in Tables 1 and 2 do use 'training' sets of 'typical genes' to optimize their signal or coding region detection modules, as well as to determine the weight associated with the assembly of the various features (e.g., explicit probabilities in HMM models). Thus, 'detectable' genes are more likely to have a TATA box, coding exons exhibiting the usual hexamer frequency, average intron and exon sizes, a single 5′ UTR and 3′ UTR exon, and a consensus polyadenylation signal. Unfortunately, many interesting genes do not have such an ideal architecture. Evolutionarily 'recent' vertebrate genes, i.e., coding for proteins with no detectable homologues in other phyla (i.e., invertebrates) may represent ~50% of all human genes (31,32). Many of these genes seem to evolve rapidly, eventually so fast that human and mouse homologous sequences do not cross-hybridize (see 15,40,92 and references therein). Although there has been no systematic study performed, genes that have undergone rapid evolution appear to be more difficult to predict than more conserved ones (40). Possibly fast evolving genes are too variable to acquire the characteristic word usage that exon/gene finder programs are trained to expect. Since these genes, by definition of their recent ancestry, cannot be found by similarity search against the complete genomes of bacteria, yeast or *C.elegans*, they are more likely to be missed altogether. The difficulty for current programs to locate genes in sequences with low (G+C) content might also be related to a higher rate of evolution in these genomic regions.

For these reasons, trying to improve the performances of the gene prediction programs by having them integrate similarity search information might not be so wise in the long run. Nevertheless, this trend has been followed by the authors of GeneID (5), GeneParser (64), FGENEHB (93), Genie (94) and GRAIL (95). The gain in accuracy on the standard benchmark set (5) is of the order of 20%, but this obviously depends on the fraction of the test set with homologues in the databases. For the sake of clarity, and to objectively evaluate our progress in the interpretation of genomic data, it would be preferable to keep the similarity information separate. There is, beyond any practical consideration, a fundamental interest in being able to decipher the genomic information on the basis of 'first principles', i.e., by detecting all the biologically significant signals hidden in the DNA sequence. Designing even more ambitious methods that do not involve a 'training set' is still worthwhile in this context, and significant progress has recently been made in this direction for the analysis of new bacterial genomes (Audic *et al.*, manuscript submitted). From a practical point of view, it is indeed advisable to always use a program like GENSCAN (or any program recognizing genes from their general properties) in combination with methods explicitly taking advantage of homology information such as PROCRUSTES, or plain database similarity searches against protein or EST databases.

The decision on which gene prediction program to use, cannot only be based on the performances listed in Table 2. Two other criteria, (i) the type and quality of sequence data at our disposal, and (ii) the type of access offered by the program, are important. Two main types of sequence data can be distinguished: well polished sequences of tens to hundreds of contiguous nucleotides, or genome survey data, i.e., resulting from a low coverage shotgun sequencing protocol (96,97). In the latter case, no more than a single exon is expected to be found in each piece of contiguous sequence. Simple exon finders, such as GRAIL I/II, HEXON, and MZEF are well suited for this type of data, as well as BLASTN (98) searches against the EST databases, and BLASTX (30) searches against the protein databases. In the special case of exon trapping data (99,100), a method not taking into account splice site information should be used, such as GRAIL I. At the other extreme, when large contigs are available, GENSCAN seems to be the best program currently available.

In case sequence data are too confidential to be transmitted over the Internet, a gene prediction program has to be installed on a local computer, and the analysis run in-house. Table 1 lists the sites where program source codes or executable files are available free of charge.

In all instances, an adequate filtering of the most abundant repeats [SINES, LINES, MAR, etc. (39)] should precede the search for exons or genes to maintain the signal/noise ratio to an acceptable level, especially if one wants to take advantage of EST similarity searches (42,43).

Future directions

Practically all coding nucleotides are now detected by the most advanced software such as MZEF and GENSCAN, according to their published performances. The *practical* problem of detecting exons in anonymous genomic sequence can thus be considered to be solved, in the sense that cognate cDNA clones are extremely likely to be detected by PCR primers or probes designed after the predictions. On the positive side, in more than 40% of (easy) cases, the protein products deduced from the predicted exon assemblies are entirely correct. In the best cases, if the amino-acid sequence reveals a specific motif, a putative function can even be assigned automatically to a newly identified gene. On the down side, 60% of the predicted proteins are wrong, and nearly 100% of the predicted gene structures have incorrect 5′ or 3′ boundaries. Thus, the full definition of the transcription unit (and of its variant through alternative processing) still requires some non-trivial experimental work. In that sense, the problem of automatically annotating genomic sequences to a level comparable to GenBank is far from being solved, even for typical protein coding genes. Given that most 3′ extremities of genes will eventually be mapped by ESTs, achieving a significant improvement on computer methods for the detection of vertebrate promoters, and thus the 5′ end of genes, is now the key to the development of the next generation of gene identification programs.

ACKNOWLEDGEMENTS

I wish to thank S.Audic, J.Fickett, M.Gelfand, D.Haussler, S.Karlin, S.Salzberg, V.Solovyev, G.Stormo and E.Uberbacher for sharing unpublished data or preprints, as well as C.Abergel, D.Gautheret and D.Robertson for their critical reading of the manuscript. J.Witkowski, R.Gibbs and P.Green were the co-organizers of the Banbury meeting on ‘Finding genes’ (Cold Spring Harbor Laboratory, 23–26 March 1997) where most information was gathered. M.Bidaud deserves a special mention for his work on the Banbury Cross web site. Research in the Structural and Genetic Information Laboratory is partly supported by Incyte Pharmaceuticals, Inc.

REFERENCES

1. Hunting through the ‘garbage’ for DNA. *Business Week*, September 2, 1996, p. 81.
2. Fickett,J.W. (1996) Finding genes by computer: the state of the art. *Trends Genet.* **12**, 316–320.
3. Fickett,J.W. (1995) The gene identification problem: an overview for developers. *Comput. Chem.* **20**, 103–118.
4. Gelfand,M.S. (1995) Prediction of function in DNA sequence analysis. *J. Comput. Biol.* **1**, 87–115.
5. Burset,M. and Guigo,R. (1996) Evaluation of gene structure prediction programs. *Genomics* **34**, 353–367.
6. Li,W. at http://linkage.rockefeller.edu/wli/gene/
7. Gelfand,M.S. (1995) FANS-REF: a bibliography on statistics and functional analysis of nucleotide sequences. *Comput. Appl. Biosci.* **11**, 541–541.
8. Claverie,J.-M., Poirot,O. and Lopez,F. (1997) The difficulty of identifying genes in anonymous vertebrate sequences. *Comput. Chem.* **21**, (in press).
9. Staden,R. and McLachlan,A.D. (1982) Codon preference and its use in identifying protein coding regions in long DNA sequences. *Nucleic Acids Res.* **10**, 141–156.
10. Shepherd,J.C. (1981) Method to determine the reading frame of a protein from the purine/pyrimidine genome sequence and its possible evolutionary justification. *Proc. Natl. Acad. Sci. USA* **78**, 1596–1600.
11. Fickett,J.W. (1982) Recognition of protein coding regions in DNA sequences. *Nucleic Acids Res.* **10**, 5303–5318.
12. Pearce,M., Blake,D.J., Tinsley,J.M., Byth,B.C., Campbell,L., Monaco,A.P. and Davies,K.E. (1993) The utrophin and dystrophin genes share similarities in genomic structure. *Hum. Mol. Genet.* **2**, 1765–1772.
13. Levinson,B., Kenwrick,S., Gamel,P., Fisher,K. and Gitschier,J. (1992) Evidence for a third transcript from the human factor VIII gene. *Genomics* **14**, 585–589.
14. De Backer,O., Verheyden,A.M., Martin,B., Godelaine,D., De Plaen,E., Brasseur,R., Avner,P. and Boon,T. (1995) Structure, chromosomal location, and expression pattern of three mouse genes homologous to the human MAGE genes. *Genomics* **28**, 74–83.
15. Legouis,R., Hardelin,J.P., Levilliers,J., Claverie,J.-M., Compain,S., Wunderle,V., Millasseau,P., Le Paslier,D., Cohen,D., Caterina,D. *et al.* (1991) The candidate gene for the X-linked Kallmann syndrome encodes a protein related to adhesion molecules. *Cell* **67**, 423–435.
16. Fickett,J.W. and Tung,C.S. (1992) Assessment of protein coding measures. *Nucleic Acids Res.* **20**, 6441–6450.
17. Claverie,J.-M. and Bougueleret,L. (1986) Heuristic informational analysis of sequences. *Nucleic Acids Res.* **14**, 179–196.
18. Beckmann,J.S., Brendel,V. and Trifonov,E.N. (1986) Intervening sequences exhibit distinct vocabulary. *J. Biomol. Struct. Dyn.* **4**, 391–400.
19. Arques,D.G. and Michel,C.J. (1987) Periodicities in introns. *Nucleic Acids Res.* **15**, 7581–7592.
20. Konopka,A.K., Smythers,G.W., Owens,J. and Maizel,J.V. Jr (1987) Distance analysis helps to establish characteristic motifs in intron sequences. *Gene Anal. Tech.* **4**, 63–74.
21. 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.
22. Farber,R., Lapedes,A. and Sirotkin,K. (1992) Determination of eukaryotic protein coding regions using neural networks and information theory. *J. Mol. Biol.* **226**, 471–479.
23. Staden,R. (1986) The current status and portability of our sequence handling software. *Nucleic Acids Res.* **14**, 217–237.
24. 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.
25. 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.
26. Lopez,R., Larsen,F. and Prydz,H. (1994) Evaluation of the exon predictions of the GRAIL software. *Genomics* **24**, 133–136.
27. Benson,D.A., Boguski,M., Lipman,D.J. and Ostell,J. (1997) GenBank. *Nucleic Acids Res.* **25**, 1–6.
28. Rice,C.M. and Cameron,G.N. (1994) Submission of nucleotide sequence data to EMBL/GenBank/DDBJ. *Methods Mol. Biol.* **25**, 413–424.
29. Claverie,J.-M. (1992) Identifying coding exons by similarity search: alu-derived and other potentially misleading protein sequences. *Genomics* **12**, 838–841.
30. Gish,W. and States,D.J. (1993) Identification of protein coding regions by database similarity search. *Nature Genet.* **3**, 266–272.
31. Claverie,J.-M. (1993) Database of ancient sequences. *Nature* **364**, 19–20.
32. Green,P., Lipman,D., Hillier,L., Waterston,R., States,D. and Claverie,J.-M. (1993) Ancient conserved regions in new gene sequences and the protein databases. *Science* **259**, 1711–1716.
33. Adams,M.D., Kelley,J.M., Gocayne,J.D., Dubnick,M., Polymeropoulos,M.H., Xiao,H., Merril,C.R., Wu,A., Olde,B., Moreno,R.F. *et al.* (1991) Complementary DNA sequencing: expressed sequence tags and human genome project. *Science* **252**, 1651–1656.

34. Okubo,K., Hori,N., Matoba,R., Niiyama,T., Fukushima,A., Kojima,Y. and Matsubara,K. (1992) Large scale cDNA sequencing for analysis of quantitative and qualitative aspects of gene expression. *Nature Genet.* **2**, 173–179.
35. McCombie,W.R., Adams,M.D., Kelley,J.M., FitzGerald,M.G., Utterback,T.R., Khan,M., Dubnick,M., Kerlavage,A.R., Venter,J.C. and Fields,C. (1992) Caenorhabditis elegans expressed sequence tags identify gene families and potential disease gene homologues. *Nature Genet.* **1**, 124–131.
36. Hillier,L.D., Lennon,G., Becker,M., Bonaldo,M.F, Chiapelli,B., Chissoe,S., Dietrich,N., DuBuque,T., Favello,A., Gish,W., Hawkins,M., Hultman,M., Kucaba,T., Lacy,M., Le,M., Le,N., Mardis,E., Moore,B., Morris,M., Parsons,J., Prange,C., Rifkin,L., Rohlfing,T., Schellenberg,K., Marra,M. *et al.* (1996) Generation and analysis of 280,000 human expressed sequence tags. *Genome Res.* **6**, 807–828.
37. Aaronson,J.S., Eckman,B., Blevins,R.A., Borkowski,J.A., Myerson,J., Imran,S. and Elliston,K.O. (1996) Toward the development of a gene index to the human genome: an assessment of the nature of high-throughput EST sequence data. *Genome Res.* **6**, 829–845.
38. Adams,M.D., Kerlavage,A.R., Fleischmann,R.D., Fuldner,R.A., Bult,C.J., Lee,N.H., Kirkness,E.F., Weinstock,K.G., Gocayne,J.D., White,O. *et al* (1995) Initial assessment of human gene diversity and expression patterns based upon 83 million nucleotides of cDNA sequence. *Nature* **377 (6547 Suppl)**, 3–174.
39. Smit,A.F.A. (1996) The origin of interspersed repeats in the human genome *Curr. Opin. Genet. Dev.* **6**, 743–748.
40. Claverie,J.-M. (1996) Progress in large scale sequence analysis. In Villar,H. (ed.) *Advances in Computational Biology* Vol. 2, pp. 161–208. JAI Press Inc., London.
41. Claverie,J.-M. and States,D.J. (1993) Information enhancement methods for large scale sequence analysis. *Comput. Chem.* **17**, 191–201.
42. Claverie,J.-M. (1996) Exon detection by similarity searches. *Methods Mol. Biol*. **68**, 283–313.
43. Claverie,J.-M. (1996) Effective large scale sequence similarity searches, in computer methods for macromolecular sequence analysis. *Methods Enzymol.* **266**, 212–227.
44. Jurka,J., Klonowski,P., Dagman,V. and Pelton,P. (1996) CENSOR—a program for identification and elimination of repetitive elements from DNA sequences. *Comput. Chem.* **20**, 119–121.
45. Smit,A.F.A. and Green,P. (1997) at http://ftp.genome.washington.edu
46. Xu,Y., Mural,R., Shah,M. and Uberbacher,E.C. (1994) Recognizing exons in genomic sequence using GRAIL II. *Genet. Eng. (NY)* **16**, 241–253.
47. 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.
48. Claverie,J.-M. and Audic,S. (1996) The statistical significance of nucleotide position-weight matrix matches. *Comput. Appl. Biosci.* **12**, 431–439.
49. Solovyev,V.V. and Lawrence,C.B. (1993) Identification of human gene functional regions based on oligonucleotide composition. In *Proc. First International Conference on Intelligent Systems for Molecular Biology*, pp. 371–379.
50. 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.
51. 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.
52. Zhang,M.Q. (1997) Identification of protein coding regions in the human genome by quadratic discriminant analysis. *Proc. Natl. Acad. Sci. USA* **94**, 565–568.
53. Fields,C.A. and Soderlund,C.A. (1990) gm: a practical tool for automating DNA sequence analysis. *Comput. Appl. Biosci.* **6**, 263–270.
54. Gelfand,M.S. (1990) Computer prediction of the exon-intron structure of mammalian pre-mRNAs. *Nucleic Acids Res.* **18**, 5865–5869.
55. Wu,T.D. (1996) A segment-based dynamic programming algorithm for predicting gene. *J. Comput. Biol.* **3**, 375–394.
56. Roytberg,M.A., Astahova,T.V. and Gelfand,M.S. (1997) Combinatorial approaches to gene recognition. *Comput. Chem.* **21**, (in press).
57. Guigo,R., Knudsen,S., Drake,N. and Smith,T. (1992) Prediction of gene structure. *J. Mol. Biol.* **226**, 141–157.
58. Dong,S. and Searls,D.B. (1994) Gene structure prediction by linguistic methods. *Genomics* **23**, 540–551.
59. Solovyev,V.V., Salamov,A.A. and Lawrence,C.B. (1995) Identification of human gene structure using linear discriminant functions and dynamic programming. In *Proc. Third International Conference on Intelligent Systems for Molecular Biology*, pp. 367–375.
60. Salzberg,S., Delcher,A., Fasman,K. and Henderson,J. (1997) A Decision Tree System for Finding Genes in DNA. Technical Report 1997-03, Department of Computer Science, Johns Hopkins University.
61. Salzberg,S. (1995) Locating protein coding in human DNA using a decision tree algorithm. *J. Comput. Biol.* **2**, 473–485.
62. Sankoff,D. and Kruskal,J.B. (1983) *Time Warps, String Edits, and Macromolecules: The Theory and Practice of Sequence Comparison*. pp: 23–29. Addison-Wesley, London.
63. 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.
64. Snyder,E.E. and Stormo,G.D. (1995) Identification of protein coding regions in genomic DNA. *J. Mol. Biol.* **248**, 1–18.
65. Gelfand,M.S. and Roytberg,M.A. (1993) Prediction of exon-intron structure by dynamic programming approach. *BioSystems* **30**, 173–182.
66. Milanesi,L., Kolchanov,N., Rogozin,I., Kel,A. and Titov,I. (1993) Sequence functional inference. In Bishop,M.J. (ed.) *Guide to Human Genome Computing*. Academic Press, Cambridge, UK, pp. 249–312.
67. Xu,Y., Mural,R.J. and Uberbacher,E.C. (1994) Constructing gene models from accurately predicted exons: an application of dynamic programming. *Comput. Appl. Biosci.* **10**, 613–623.
68. Borodovsky,M. and McIninch,J. (1993) GENMARK: Parallel gene recognition for both DNA strands. *Comput. Chem.* **17**, 123–133.
69. Krogh,A., Brown,M., Mian,I.S. Sjölander,K. and Haussler,D. (1994) Hidden Markov models in computational biology: application to protein modeling. *J. Mol. Biol.* **235**, 1501–1531.
70. Krogh,A., Mian,I.S. and Haussler,D. (1994) A hidden Markov model that finds genes in *E.coli* DNA. *Nucleic Acids Res.* **22**, 4768–4778.
71. Henderson,J., Salzberg,S. and Fasman,K.H. (1997) Finding genes in DNA with a hidden Markov model. *J. Comput. Biol.* **4**, 127–142.
72. Kulp,D., Haussler,D., Reese,M.G. and Eeckman,F.H. (1996) A generalized hidden Markov model for the recognition of human genes in DNA. In *Proc. Fourth International Conference on Intelligent Systems for Molecular Biology*, pp. 134–142.
73. Burge,C. and Karlin,S. (1997) Prediction of complete gene structure in human genomic DNA. *J. Mol. Biol.* **268**, 1–17.
74. Viterbi,A.J. (1967) Error bounds for convolutional codes and an asymptotically optimal decoding algorithm. *IEEE Trans. Informt. Theory* **IT-13**, 260–269.
75. Gelfand,M.S., Mironov,A.A. and Pevzner,P.A. (1996) Gene recognition via spliced sequence alignment. *Proc. Natl. Acad. Sci. USA* **93**, 9061–9066.
76. Rogozin,I.B., Milanesi,L. and Kolchanov,N.A. (1996) Gene structure prediction using information on homologous protein sequence. *Comput. Appl. Biosci*. **12**, 161–170.
77. Brockdorff,N., Ashworth,A., Kay,G.F., McCabe,V.M., Norris,D.P., Cooper,P.J., Swift,S. and Rastan,S. (1992) The product of the mouse Xist gene is a 15 kb inactive X-specific transcript containing no conserved ORF and located in the nucleus. *Cell* **71**, 515–526.
78. Pfeifer,K., Leighton,P.A. and Tilghman,S.M. (1996) The structural H19 gene is required for transgene imprinting. *Proc. Natl. Acad. Sci. USA* **93**, 13876–13883.
79. Wevrick,R. and Francke,U. (1997) An imprinted mouse transcript homologous to the human imprinted in Prader-Willi syndrome (IPW) gene. *Hum. Mol. Genet.* **6,** 325–332.
80. Liu,A.Y., Torchia,B.S., Migeon,B.R. and Siliciano,R.F. (1997) The human NTT gene: identification of a novel 17-kb noncoding nuclear RNA expressed in activated CD4+ T cells. *Genomics* **39**, 171–184.
81. Burley,S.K. and Roeder,R.G. (1996) Biochemistry and structural biology of transcription factor IID (TFIID). *Annu. Rev. Biochem.* **65**, 769–799.
82. Novina,C.D. and Roy,A.L. (1996) Core promoters and transcriptional control. *Trends Genet*. **12**, 351–355
83. Ptashne,M. and Gann,A. (1997) Transcriptional activation by recruitment. *Nature*. **386**, 569–577.
84. Kauffmann,J., Verrijzer,C.P., Shao,J. and Smale,S.L. (1996) Ras 1 signaling and transcriptional ompetence in the R7 cell of Drosophila. *Genes Dev.* **10**, 2167–2178.
85. Prestridge,D.S. (1995) Predicting Pol II promoter sequences using transcription factor binding sites. *J. Mol. Biol.* **249**, 923–932.
86. Bucher,P., Fickett,J.W. and Hatzigeorgiou,A. (1996) Computational analysis of eukaryotic transcriptional regulatory elements. *Comput. Appl. Biosci*. **12**, 361–446.
87. Fickett,J.W. and Hatzigeorgiou,A. (1997) Eukaryotic promoter recognition. *J. Comput. Biol*. (in press).
88. Fichant,G.A. and Burks,C. (1991) Identifying potential tRNA genes in genomic DNA sequences. *J. Mol. Biol.* **220**, 659–671.

89. Laferriere,A., Gautheret,D. and Cedergren,R. (1994) An RNA pattern matching program with enhanced performance and portability. *Comput. Appl. Biosci*. **10**, 211–212.
90. Ansari-Lari,M.A., Shen,Y., Muzny,D.M., Lee,W. and Gibbs,R.A. (1997) Large-scale sequencing in human chromosome 12p13: experimental and computational gene structure determination. *Genome Res.* **7**, 268–280.
91. Ansari-Lari,M.A., Muzny,D.M., Lu,J., Lu,F., Lilley,C.E., Spanos,S., Malley,T. and Gibbs,R.A (1996) A gene-rich cluster between the CD4 and triose-phosphate isomerase genes at human chromosome 12p13. *Genome Res.* **6**, 314–326.
92. Simmler,M.C., Cunningham,D., Clerc,P., Vermat,T., Cruaud,C., Pawlak,A., Szpirer,C., Weissenbach,J., Claverie,J.-M. and Avner,P. (1996) A 94 kb genomic sequence 3′ to the murine *Xist* gene reveals an AT-rich region containing a new testis specific gene *Tex. Hum. Mol. Genet.* **5**, 1713–1726.
93. Solovyev,V.V. and Salamov,A.A. (1997) The gene-finder computer tools for analysis of human and model organisms genome sequences. In *Proc. Fifth International Conference on Intelligent Systems for Molecular Biology*, pp. 294–302.
94. Kulp,D., Haussler,D., Reese,M.G. and Eeckman,F.H. (1997) Integrating database homology in a probabilistic gene structure model. *Proc. Pacific Symp. Biocomputing '97*.
95. Xu,Y. and Uberbacher,E.C. (1997) Automated gene identification in large scale genomic sequences. *J. Comp. Biol.* **4**, 325–338.
96. Claverie,J.-M. (1994) A streamlined random sequencing strategy for finding coding exons. *Genomics* **23**, 575–581.
97. Kamb,A., Wang,C., Thomas,A., DeHoff,B.S., Norris,F.H., Richardson,K., Rine,J., Skolnick,M.H. and Rosteck,P.R. Jr (1995) Software trapping: a strategy for finding genes in large genomic regions. *Comput. Biomed. Res.* **28**, 140–153.
98. 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.
99. Buckler,A.J., Chang,D.D., Graw,S.L., Brook,J.D., Haber,D.A., Sharp,P.A. and Housman,D.E. (1991) Exon amplification: a strategy to isolate mammalian genes based on RNA splicing. *Proc. Natl. Acad. Sci. USA* **88**, 4005–4009.
100. Datson,N.A., van de Vosse,E., Dauwerse,H.G., Bout,M., van Ommen,G.J. and den Dunnen,J.T. (1996) Scanning for genes in large genomic regions: cosmid-based exon trapping of multiple exons in a single product. *Nucleic Acids Res.* **24**, 1105–1111.