Algorithms for Molecular Biology **Fall Semester, 1998**

Lecture 7: January 17, 1999

Lecturer: Ron Shamir *Scribe: Benny Ben-Ami and Nadav Grossaug* [1]

7.1 Gene Finding

7.1.1 Motivation

The size of GenBank is growing rapidly. Currently there are approximately 200 millions bases of human DNA sequences available in it, and it's size is expected to triple over the next year. When the human genome project will be completed, it's size will exceed 3 billion base pairs. Roughly 90% of the human genome is *non-coding*, namely is not a template for a protein. Due to the size of the database, manual searching of genes, who do code for proteins, is not practical. Methods that help the biologists focusing their search are needed.

7.1.2 Biological Background

Gene expression is the biological process by which a DNA sequence generates a protein. It involves two steps: *transcription* and *translation*. Transcription produces a mRNA (messenger RNA) sequence using the DNA sequence as a template. The mRNA sequence produced is complementary to the DNA strand which was used as template. The subsequent process, called translation, synthesizes the protein from the mRNA. This process is performed by sub cellular elements called *ribosomes* (Figure 7.2).
The transcription is carried out from the 5' end to the 3' end of the DNA strand. This direction along the strand is called *downstream* while the opposite direction is called *upstream*. The enzyme preforming the transcription, RNA polymerase, starts transcription a few bases upstream of the start codon and terminates a few bases after the stop codon. The regions in both ends of the DNA coding region which are transcripted into a mRNA, but do not code the protein are called *untranslated regions (UTR)* (see figure 7.4 and 7.5). RNA polymerase molecules start transcription by recognizing and binding to *promoter regions* upstream of the desired transcription start sites. These promoter regions control the rate of gene expression.

Since there are 64 different possible codons, and only 20 amino acids, multiple codons represent the same amino acid. Besides those codons coding amino acids, there is one, called *start codon*, that indicates the beginning of translation (as well as code for the amino acid Metionine), and three, called *stop codons*, that indicate end of translation. The genetic code

[1] Based partially on notes by Manu Thambi [5] and Summet Sobti [4].

Figure 7.1: Steps in gene expression

is shown in figure 7.3. Because the codons are triplets of bases, any given DNA sequence can be interpreted in three possible ways, depending on where the coding starts. These three ways are called *reading frames*. An *open reading frame (ORF)* is a sequence of codons with no stop codon.

Prokaryotes

A *prokaryotic cell* is a cell which doesn't have a nucleus, while *eukaryotic* cells have nuclei. There are some differences between prokaryotic cells and eukaryotic cells which are relevant to gene recognition. In prokaryotic cells most of the DNA sequence is coding for protein. For example, almost 70% of the genome of the bacterium *H.influenzea* is coding. Also, each gene is one continuous stretch of bases. That is, there are no introns in the coding region.

7.1.3 Finding Long ORF's

One way to distinguish coding regions from non-coding regions, is to look at the frequencies of stop codons. Assuming a uniform random distribution, a stop codon is expected to be observed every $64/3 \approx 21$ codons (since there are 3 stop codons). Average proteins are much longer, being coded for by about 1000bp (base pairs). Each coding region has only one stop codon, which terminates the region. Therefore, one way to detect the coding regions, is to look for long sequences of codons, without any stop codon.

The algorithm that uses the above idea, scans the DNA sequence, looking for long ORF's in all three reading frames. After detecting a stop codon, the algorithm scans backward, searching for a start codon.

This algorithm will fail to detect very short genes, and also won't identify overlapping long ORFs on opposite strands.

7.1.4 Compositional Differences

A more informative method to determine coding regions, takes advantage of the frequencies in which the various codons occur in coding regions. For example, the amino acids Leucine, Alanine and Tryptophan are coded by 6,4 and 1 different codons respectively. In a translation of a uniformly random DNA sequence, these amino acids should occur in the ratio 6:4:1, but in a protein they occur in a different ratio - 6.9:6.5:1. Therefore coding DNA is not random.

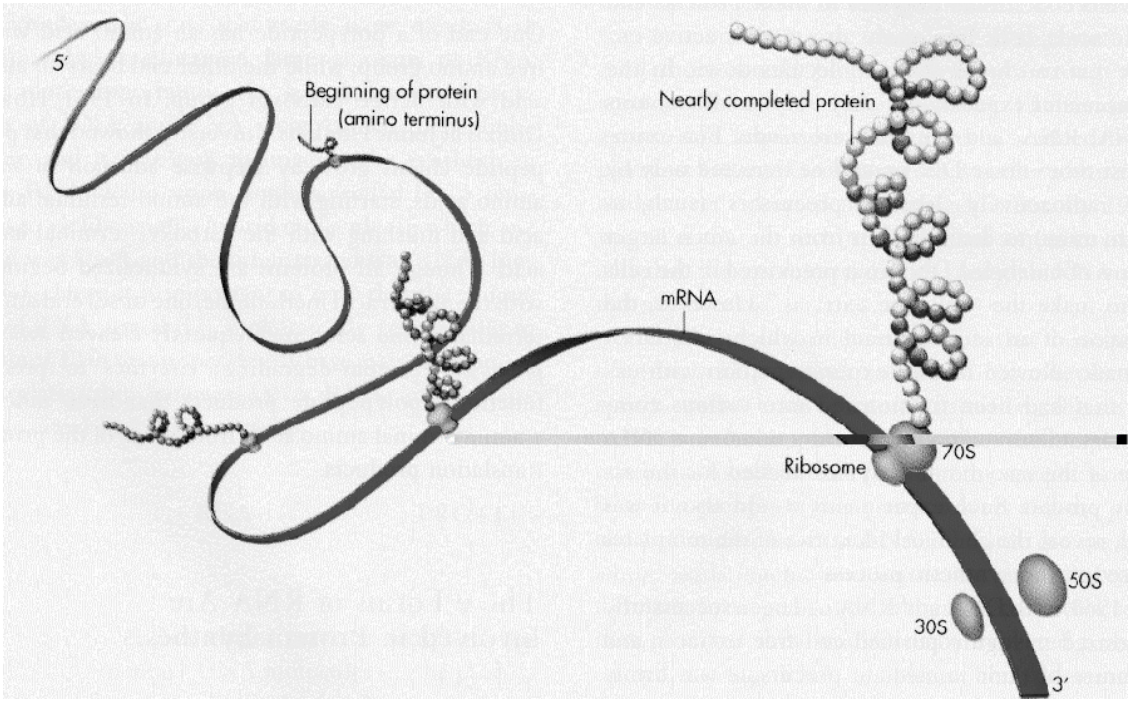


Figure 7.2: mRNA translation: The polypeptide chains are elongated as the ribosomes move along the mRNA molecules, with the 5' end of the mRNA being translated first.

Another example of the non-uniformness of coding DNA is the fact that A or T occurs in the 3^{rd} position of a codon 90% of the time, while G or C occur only 10% of the time (these statistics are different for different species).

Assume that $a_1, b_1, c_1, a_2, b_2, c_2, \ldots, a_{n+1}, b_{n+1}$ is a coding sequence of bases with an unknown reading frame. Let f_{abc} denote the frequency with which the codon abc appear in a coding region. Under the assumption that successive codons are chosen independently at random according to the frequencies f_{abc}, the probability of observing the sequence of n codons appearing in the reading frame starting with $a_1 b_1 c_1$ is

$$p_1 = f_{a_1 b_1 c_1} \times f_{a_2 b_2 c_2} \times \ldots \times f_{a_n b_n c_n}$$

Similarly, the probability of observing the n codons in the second and third coding frames are :

$$\begin{aligned} p_2 &= f_{b_1 c_1 a_2} \times f_{b_2 c_2 a_3} \times \ldots \times f_{b_n c_n a_{n+1}} \\ p_3 &= f_{c_1 a_2 b_2} \times f_{c_2 a_3 a_3} \times \ldots \times f_{c_n a_{n+1} b_{n+1}} \end{aligned}$$

The Genetic Code

	U	C	A	G	
U	Phe	Ser	Tyr	Cys	U
	Phe	Ser	Tyr	Cys	C
	Leu	Ser	STOP	STOP	A
	Leu	Ser	STOP	Trp	G
C	Leu	Pro	His	Arg	U
	Leu	Pro	His	Arg	C
	Leu	Pro	Gln	Arg	A
	Leu	Pro	Gln	Arg	G
A	Ile	Thr	Asn	Ser	U
	Ile	Thr	Asn	Ser	C
	Ile	Thr	Lys	Arg	A
	Met(start)	Thr	Lys	Arg	G
G	Val	Ala	Asp	Gly	U
	Val	Ala	Asp	Gly	C
	Val	Ala	Glu	Gly	A
	Val	Ala	Glu	Gly	G

Figure 7.3: The genetic code. AUG us the start codon, while UAA, UAG and UGA are the stop codons.

The probability P_i which is the probability of the ith reading frame being the translated one (assuming the region is coding) can be calculated as follows:

$$P_i = \frac{p_i}{p_1 + p_2 + p_3}$$

The above computation can be used in a search algorithm as follows: Slide a window of size n along the sequence, and compute P_i for each start position of the window. The *Codon Preference* program, which is part of the *GCG* library, implements this method.

Figure 7.6 shows the plot of $log(P/1-P)$, which is the log likelihood, for the three reading frames. It shows that the value of $log(P/1-P)$ is usually above 4 ($P > 99.99\%$) in regions where the DNA sequence codes for a protein in that reading frame.

This method depends on the accuracy of the codon frequency statistics of already found

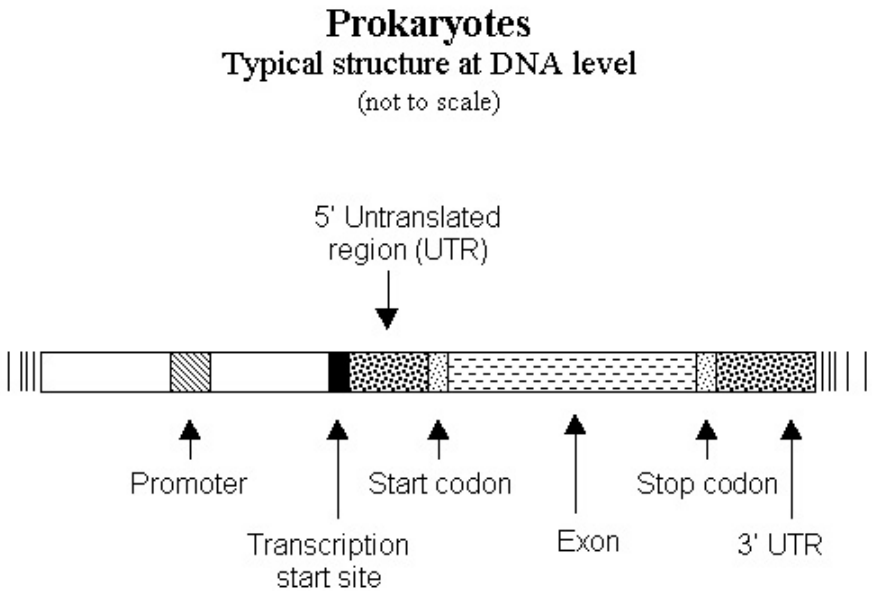


Figure 7.4: Typical prokaryotic gene structure at DNA level (not to scale)

genes. The algorithm will also have difficulty in detecting horizontal gene transfer and other causes of heterogeneity.

Various other methods have also been tried. One variation of the codon frequency method is to use 6-tuple frequencies, which are more informative than 3-tuple (codon) frequencies. Another approach (which is illustrated in figure 7.7) considers the periodicity of certain bases in protein coding regions, whose statistical behaviour distinguishes them from non coding regions.

7.1.5 Detecting Promoter Regions

Promoter regions in DNA sequences do not follow a strict pattern. This makes the identification of promoter regions more difficult. Although promoter regions vary, it is usually possible to find a DNA sequence (called the *consensus sequence*) to which all the of them are very similar. For example, the consensus in the bacterium *E.coli*, based on the study of 263 promoters, is TTGACA followed by 17 uncorrelated base pairs, followed by TATAAT, with the latter, called *TATA box*, located about 10 bases upstream of the transcription start site. None of the 263 promoter regions exactly match the above consensus sequence. Nevertheless, the consensus sequence is representative: nearly all of *E.coli*'s promoters terminate with 2 of the 3 specified letters of the sequence TAxyzT, 80-90% have all 3, and xyz is TAA in

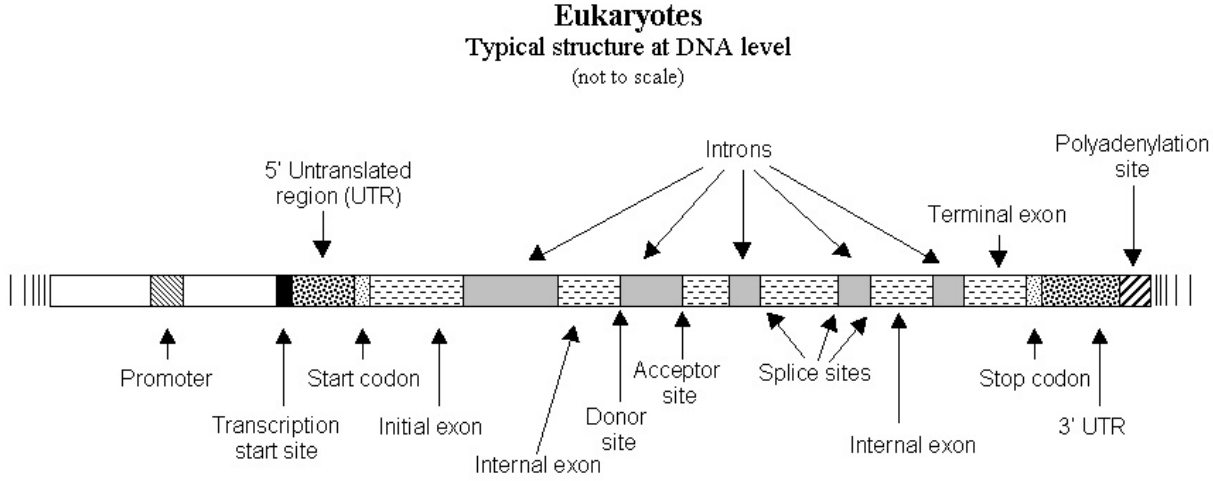


Figure 7.5: Typical eukaryotic gene structure at DNA level (not to scale)

approximately 50% of the promoter regions.

Due to the high variability, exact methods cannot be used for identifying promoter regions by the TATA box. Instead, we use a pattern search method based on frequencies. We construct a table of statistics, $f_{b,i}$, where $f_{b,i}$ is the frequency of the base b in position i of the known promoter region suffixes. We assume positions are independent.

Let f_b denote the expected frequency of the base b in the genome. We calculate the likelihood of a given sequence being a TATA-box. For a sequence $S = B_1 B_2 \ldots B_6$ the likelihood of it being a TATA-box is:

$$P(S|S \text{ is a TATA-box}) = \prod_{i=1}^{6} f_{B_i,i}$$

Similarly, the likelihood of observing it, given it is a "non-promoter" is:

$$P(S|S \text{ is not a TATA-box}) = \prod_{i=1}^{6} f_{B_i}$$

The log-likelihood ratio is therefore:

$$\log\left(\frac{P(S|\text{promoter})}{P(S|\text{non-promoter})}\right) = \log\left(\frac{\prod_{i=1}^{6} f_{B_i,i}}{\prod_{i=1}^{6} f_{B_i}}\right) = \sum_{i=1}^{6} \log\left(\frac{f_{B_i,i}}{f_{B_i}}\right)$$

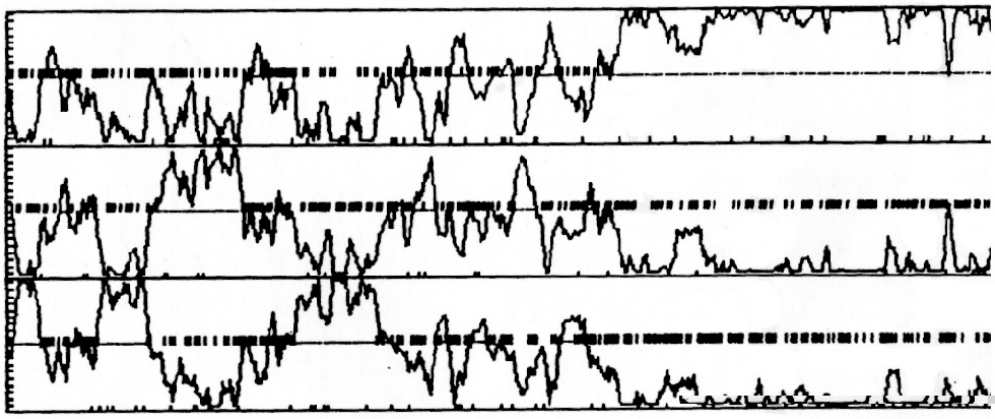


Figure 7.6: Application of codon frequency method to bases 1 to 10,000 of the liverwort *Marchantia* chloroplast genome. The horizontal scale marks every 100th base, and the bars above indicate the extent of known protein coding segments. The three boxes above contain plots of the probability that each of the three reading frames is coding for a protein. The short vertical lines that bisect the mid-height of each box mark the positions of the stop codons in the corresponding reading frames. The vertical scale within each box is of $log(P/1-P)$, so that, for example, 4 points up the scale from the mid-height corresponds to 99.99% probability of the current region being coding.

From the table $f_{B_i,i}$ we therefore construct a *scoring matrix*, with each entry $S_{b,i}$ denoting the score that a sequence should be given for having the base b in the i-th position. The score $s_{b,i}$ is computed by the following formula:

$$s_{b,i} = \log\left(\frac{f_{b,i}}{f_b}\right)$$

The algorithm simply scans through the DNA sequence and computes the likelihood ratio for every consecutive 6 bases and thus locates regions where the likelihood ratio is high. This model has the disadvantage that it doesn't exploit all of the known information (i.e. intron/exon statistic, dependencies between bases occurring in the promoter regions etc.)

Why aren't promoters precise like the stop codons etc.? A likely answer is that nature uses the variation in promoters to control expression levels of various genes. That is, the rate of the gene expression process depends on the conservation of the promoter region. This hypothesis is supported by results from chemistry. Experiments show that when a RNA polymerase molecule gets bounded to the prompter region in order to initial transcription, there is an 80% correlation between the weight matrix score of the region and the binding energy. This means that if the promoter region is very conserved, i.e., very similar to the

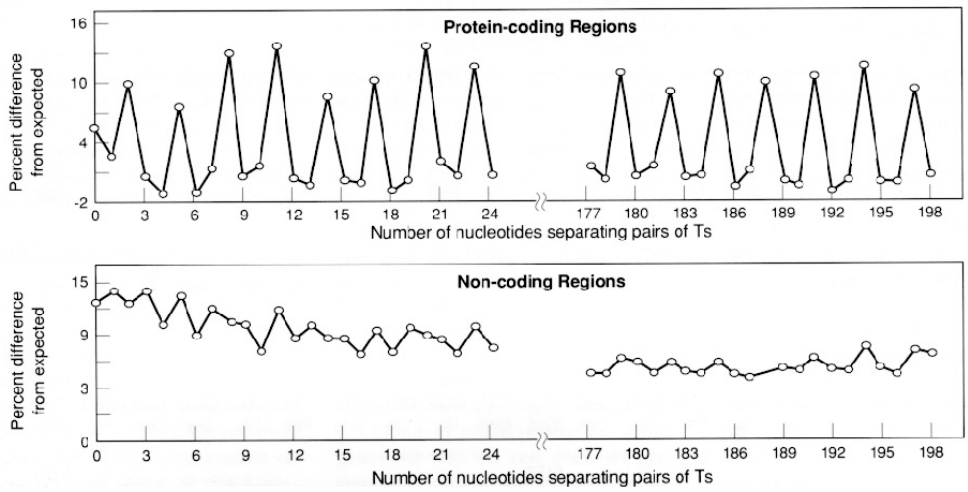


Figure 7.7: Periodicity of T: for each gap size n, the number of occurrences of the pattern T...T, with n bases between the two T's, was counted. Plotted is the present difference of that number from the number of such pair expected if the bases occurred at random. Since the T's occur preferentially at the second codon position, in coding regions, the present difference at n=2,5,8,... is noticeably large.

consensus sequence, then the binding energy barrier is low and thus the protein production rate is higher (because the RNA polymerase can easily bind to the protein coding region). When the difference from the consensus sequence is bigger, the energy barrier is higher, and the protein production is slower. The unfortunate consequence is that rarely expressed genes will be harder to find by this means.

7.1.6 Gene Structure in Eukaryotes

The gene structure and the gene expression mechanism in eukaryotes are far more complicated than in prokaryotes. In typical eukaryotes, the region of the DNA coding for a protein is usually not continuous. This region is composed of alternating stretches of *exons* and *introns*. During transcription, both exons and introns are transcribed onto the RNA, in their linear order. Thereafter, a process called *splicing* takes place, in which, the intron sequences are excised and discarded from the RNA sequence. The remaining RNA segments, the ones corresponding to the exons are ligated to form the mature RNA strand.

A typical multi-exon gene has the following structure (as illustrated in figure 7.5). It starts with the promoter region, which is followed by a transcribed but non-coding region called *5' untranslated region (5' UTR)*. Then follows the initial exon which contains the start

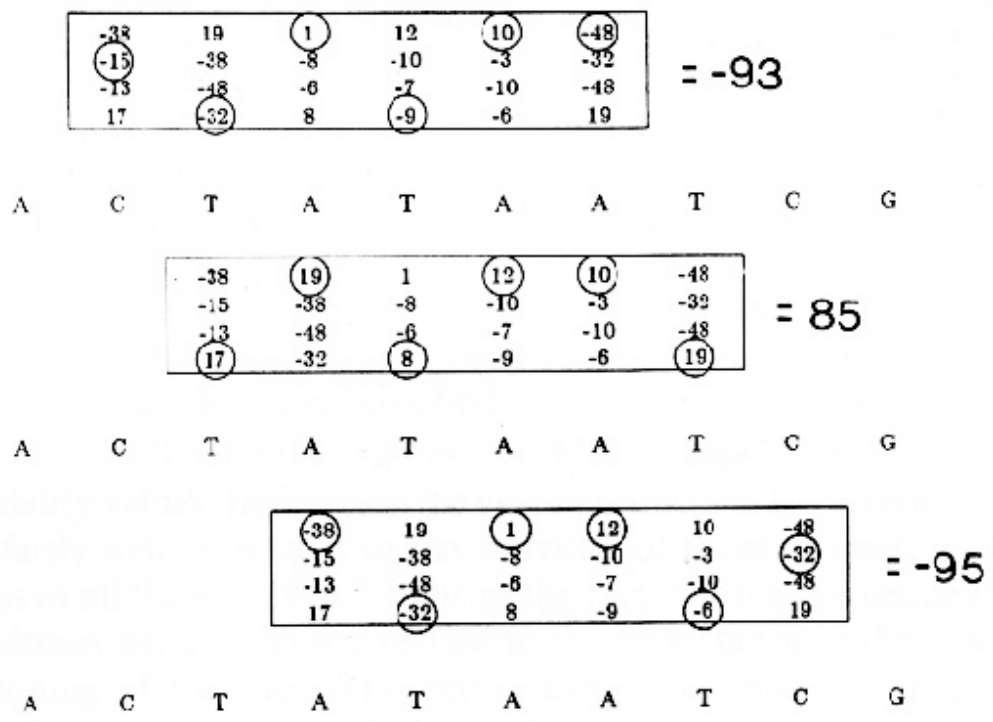


Figure 7.8: Promoter detection by matrix evaluations of the sequence for containing a TATA-box. The matrix contains an element for each possible matrix position. For each alignment of the matrix above the sequence, a score is computed based on the matrix elements corresponding to the sequence. The matrix rows correspond to the bases A,C,G,T from top to bottom. The sequence TATAAT scores highest.

codon. Following the initial exon, there is an alternating series of introns and internal exons, followed by the terminating exon, which contains the stop codon. It is followed by another non-coding region called the *3' UTR*. Ending the eukaryotic gene, there is a polyadenylation (polyA) signal: the nucleotide Adenine repeating several times. The exon-intron boundaries (i.e., the splice sites) are signalled by specific short (2bp long) sequences. The 5'(3') end of an intron (exon) is called the *donor* site, and the 3'(5') end of an intron (exon) is called the *acceptor* site.

The problem of gene identification is complicated in the case of eukaryotes by the vast variation that is found in gene structure. In order to be able to apprehend this, we shell consider some statistics from the available genomic data. On average, a vertebrate gene is around 30Kb long, out of which the coding region is only about 1Kb long. The average

coding region consists of 6 exons, each about 150bp long. Huge deviations from the average are observed. For example, the gene called *dystrophin* is 2.4MB long. *Blood coagulation-factor VIII* has 26 exons whose size varies from 69bp to 3106bp, with the total coding region reaching length around 186Kb and the introns lengths adding up to 32.4Kb. Intron number 22 produces 2 transcripts unrelated to this gene, one for each strand. An average 5' UTR is 750bp long, but it can be longer and span several exons (for example, in the MAGE family). On average, the 3' UTR is about 450bp long, but examples exist where its length exceeds 5Kb (e.g., the gene for Kallman's syndrome).

7.1.7 A Probabilistic Model of Gene Structure

As we have seen, a Hidden Markov Model (HMM) is a Markov chain in which the states are not directly observable. Instead, the output of the current state is observable. The output symbol for each state is randomly chosen from a finite output alphabet according to some probability distribution.

A *Generalized Hidden Markov Model* (GHMM) generalizes the HMM as follows: in a GHMM, the output of a state may not be a single symbol. Instead, the output may be a string of finite length. For a particular current state, the length of the output string as well as the output string it self might be randomly chosen according to some probability distribution. The probability distribution need not be the same for all states. For example, one state might use a weight matrix model for generating the output string, while another might use a HMM. Formally a GHMM is described by a set of four parameters:

- A finite set Q of states.
- Initial state probability distribution π.
- Transition probabilities $T_{i,j}$ for $i, j \in Q$.
- Length distributions f of the states (f_q is the length distribution for state q).
- Probabilistic models for each of the states, according to which output strings are generated upon visiting a state.

The probabilistic model for gene structure as suggested by Berge and Karlin [1], is based on GHMM (see figure 7.9). The states of the GHMM correspond to the different functional units on a gene, like promoter regions, exon, intron etc. The transition between the states ensure that the order in which the model visits various states is biologically consistent.

The states for an intron and an internal exon are subdivided according to *phase* offset to the codon frames. For $0 \leq i \leq 2$, the state I_i (respectively, E_i) corresponds to introns (exons) starting i positions after a codon starts. Note that the only transition from I_i to any internal exon state is to E_i. Also note that the model is divided into two symmetric halves.

The upper half of the figure (states with a "+" superscript) models a gene on the forward strand and the lower half models a gene on the backward strand of the genomic sequence. If the parameters (like π, $a_{i,j}$, etc.) are suitably determined, then the model can be used for gene structure prediction in the following manner.

Definition 7.1 *A* parse Φ *of sequence* S *is an ordered sequence of states* $(q_1, \ldots, q_t)$ *with an associated duration* d_i *to each state. The* length *of* Φ *is* $L = \sum_{i=1}^{t} d_i$

Suppose we are given a DNA sequence S and a parse Φ, both of length L. The conditional probability of the parse Φ given that the sequence generated is S, can be computed as:

$$P(\Phi_i|S) = \frac{P(\Phi_i, S)}{P(S)} = \frac{P(\Phi_i, S)}{\sum_{\Phi_j \text{ is a parse of length L}} P(\Phi_j, S)}$$

Let S_i be the segment of S produced by q_i, and let $P(S_i|q_i, d_i)$ be the probability of generating S_i by the sequence generation model of state q_i with length d_i.

$$P(\Phi_i|S) = \prod f_{q_i}(d_i) P(S_i|q_i, d_i) = \prod_{k=2}^{t} T_{q_{k-1}q_k} f_{q_k}(d_k) P(S_k|q_k, d_k)$$

The most probable parse, Φ_{opt}, can be computed by Viterbi like algorithm. $P(S)$ can be computed by a forward-like algorithm.

7.1.8 *GENSCAN*

GENSCAN, a computer program for gene identification, uses this model. In this section we shall consider how GENSCAN determines various parameters of the model to get meaningful results. The program uses a training set of 238 multi-exon genes and 142 single-exon genes. These are completely sequenced genes from GenBank. On whole, the training set consists of about 2.5 million base pairs .

Initial state probabilities

The initial probabilities of various states in the model should be proportional to the frequencies with which various functional units occur in the actual human genomic data. For example, if the estimated proportion of the non-coding intergenic region is 80%, then initial probability for the state N (see figure 7.9) must be around 0.8. But as a matter of fact, the relative bulk of the various functional units is found to vary considerably with the C+G content (*isochore*) of the genomic sequence (see figure 7.10). Thus, for training GENSCAN the training set is divided into four categories depending on the C+G content of the sequence. The categories are:

1. ($<$ 43% C+G)
2. (43 -51% C+G)
3. (51 - 57% C+G)
4. ($>$ 57% C+G)

For each of these categories, separate initial state probabilities are computed by estimating the relative frequencies of various functional units in these categories.

Transition probabilities

Transition probabilities are also known to vary quite a bit with the C+G content (although not as much as the initial probabilities). Thus, transition probabilities are also separately computed for each of the categories. Of course, while estimating these probabilities, it is ensured that the transitions are biologically permissible. For example, some transitions are obligatory (like $P^+ \rightarrow F^+$). Such transitions are assigned probability one.

State length distributions

Different functional units on a gene have vastly different lengths. For example, an average internal exon is about 150bp long, while introns of the order of 1Kbp length are not uncommon. Thus, in our probabilistic model of gene structure, different states need to have different length distributions. Intron lengths are known to vary dramatically with the C+G content. For example, the mean intron length for category I ($<$ 43% C+G) of the training set is 2069bp as opposed to only 518bp for category IV ($>$ 57 % C+G) (see figure 7.10). Thus, the program uses separate distributions for intron states in each category. The learning set shows quite different length distributions for initial exons, internal exons and terminal exons. Consequently, different distributions are used for them. It is important to note here is that the length of an internal exon has to be consistent with the phase of its adjacent introns. For example, if the preceding state is I_2 and the succeeding state is I_1 , then the generated internal exon length (for state E_2 in this case) must be $3n + 2$ for some n. n is therefore generated randomly according to the length distribution and then a string of length $3n + 2$ is generated according to the string generating model for that state. For the 5' UTR and 3' UTR states, geometric distributions with mean values of 769bp and 457bp are used.

Signal models

GENSCAN uses 3 kinds of signal models to model different functional units. It uses the weight matrix model (WMM) for modeling polyadenylation signals, translation initiation

signal, translation termination signal and promoters. A modified version of the weighted array model (WAM) is used for acceptor splice sites ([4] p. 61).

7.1.9 Performance of GENSCAN

Performance of GENSCAN has been compared to that of other computer programs written for gene finding. The Burset/Guigo set of 570 vertebrate multi-exon gene sequences [2] was used as the test set. The results (as reported in [2]) have been quite encouraging. Both at the nucleotide level as well as the exon-level, GENSCAN's accuracy has been significantly better that other programs.

Important features of GENSCAN include

- Identification of complete intron/exon structures of a gene in genomic DNA.
- Ability to predict multiple genes and to deal with partial as well as complete genes.
- Ability to predict consistent sets of genes occurring on either or both strands of the DNA.

Although the results are good they are still not good enough for massive gene finding. GENSCAN has 80% chance of detecting an exon. If a gene has more then one exon the probability of correctly detecting all of them declines rapidly.

7.1.10 Spliced Alignment

Given a genomic sequence and a set of candidate exons, the spliced alignment algorithm [3] explores all possible exon assemblies and finds a chain of exons which best fits a related target protein. The set of candidate exons is constructed by considering all blocks between candidate acceptor and donor sites (i.e., between AG dinucleotide at the intron-exon boundary and GU dinucleotide at exon-intron boundary) and further filtration of this set. To avoid losing true exons, the filtration procedure is designed to be very gentle, and the resulting set of blocks may contain a large number of false exons. Instead of trying to identify the correct exons by further pursuit of statistical methods, The algorithm considers all possible chains of candidate exons and finds a chain with the maximum global similarity to the target protein.

Spliced alignment problem

We start with the formal definitions and statement of the spliced alignment problem. Let $G = g_1 g_2 \dots g_n$ be a string of letters and let $B = g_i \dots g_j$ and $B' = g_{i'} \dots g_{j'}$ be substrings of G.

We write $B < B'$ if $j < i'$, i.e. if B ends before B' begins. A sequence $\Gamma = \{B_1, \ldots, B_k\}$ of substrings of G is a *chain* if $B_1 < B_2 < B_3 < \ldots < B_k$. We denote the concatenation of all the strings from Γ by Γ^*. Given two strings G and T, let $S(G,T)$ be the score of the optimal global alignment between them.

Problem 7.1 *Spliced alignment*
Instance: *A new genomic sequence $G = g_1 \ldots g_n$. A target sequence (related protein) $T = t_1 \ldots t_m$. A set $B = B_1, \ldots, B_k$ of blocks (substring of G).*
Question: *Find a chain Γ of blocks from B such that $S(\Gamma^*, T)$ is maximum.*

Spliced alignment algorithm

To solve the problem, we use dynamic programming.

We start with a few notations: Let $B_1 * B_2$ denote the concatenation of B_1 and B_2. A *j-prefix* of $A = a_i \ldots a_j \ldots a_l$ is $A(j) = a_i \ldots a_j$. For block $B = g_i \ldots g_j$ $\text{First}(B) = i$ and $\text{Last}(B) = j$. A Chain $\Gamma^* = B_1 * \ldots * B_k$ *ends* at $\text{Last}(B_k)$, and it ends *before* position i, if $\text{Last}(B_k) < i$. In addition, we define $P(i,j) = \max_{\substack{\text{all chains } \Gamma \\ \text{ending before } i}} S(\Gamma^*, T(j))$.

According to these definitions, the optimal spliced alignment score is $P(n+1, m)$. When $\text{First}(B_k) \le i \le \text{Last}(B_k)$ we can define the *i-prefix* of $\Gamma = B_1 * \cdots * B_k$ by the following expression: $\Gamma^*(i) = B_1 * \cdots * B_{k-1} * B_k(i)$. If $\text{First}(B_k) \le i \le \text{Last}(B_k)$ we define $BL(i,j,l) = \max_{\substack{\text{all chains } \Gamma \\ \text{ending at } l}} S(\Gamma^*(i), T(j))$. Now, we can calculate $P(i,j)$:

$P(1,j) = j \cdot \Delta(-, t_j)$ where $\Delta(x,y)$ is the score of aligning x and y.

$$P(i,j) = \max \begin{cases} P(i-1,j) \\ BL(i-1,j,i-1) & \text{if there exist chains ending at } i-1 \end{cases}$$

$BL(i,j,l)$ satisfies the following recurrence relation:

$$BL(i,j,l) = \max \begin{cases} BL(i-1,j-1,l) + \Delta(G_i, t_j) & \text{if } j \ge 1, \exists B\text{: } \text{First}(B) < i \le \text{Last}(B) = l \\ BL(i-1,j,l) + \Delta(g_i, -) & \text{if } \exists B\text{: } \text{First}(B) < i \le \text{Last}(B) = l \\ P(i,j-1) + \Delta(g_i, t_j) & \text{if } j \ge 1, \exists B\text{: } \text{First}(B) = i \le \text{Last}(B) = l \\ P(i,j) + \Delta(g_i, -) & \text{if } \exists B\text{: } \text{First}(B) = i \le \text{Last}(B) = l \\ BL(i,j-1,l) + \Delta(-, t_j) & \text{if } j \ge 1 \end{cases}$$

Complexity

We say that a block B is prime, if it contains all blocks ending at $\text{Last}(B)$. We define $C_p = \frac{1}{n} \sum_{\text{prime } B} \text{size}(B)$, and denote the number of blocks by b, then the complexity of the algorithm is $O(m \cdot n \cdot C_p + m \cdot b)$ both for time and space. The space complexity can be reduced using the Hirchberg method.

Results

The number of exon assemblies is huge; However, the spliced alignment algorithm is fast enough to process large genomic fragments (up to 180,000 nucleotides) containing multi-exon genes (more than 30 exons). After the highest-scoring exon assembly is found, the hope is that it represents the correct exon-intron structure. This is almost guaranteed if a protein sufficiently similar to the one encoded in the analyzed fragment is available (99% correlation between predicted and actual genes with mammalian targets). This method is good only if you have the right protein to compare it with. It can identify up to 53% of all genes.

7.1.11 Similarity Based Methods for Gene Finding

- Comparison to EST databases.
- Comparison of genomic sequence translated, to protein database.
- Spliced alignment.
- Comparison of translated genomic sequence to translated genome/cDNA sequence.
- Comparison of genomic sequence with homologous genomic sequence from close organisms.

All this methods can give a good clue for existence of a gene but they rely on the accuracy of the database they use. Similarity based methods can identify only 50% of human genes. They cannot identify genes unique to human, nor genes whose protein is unavailable.

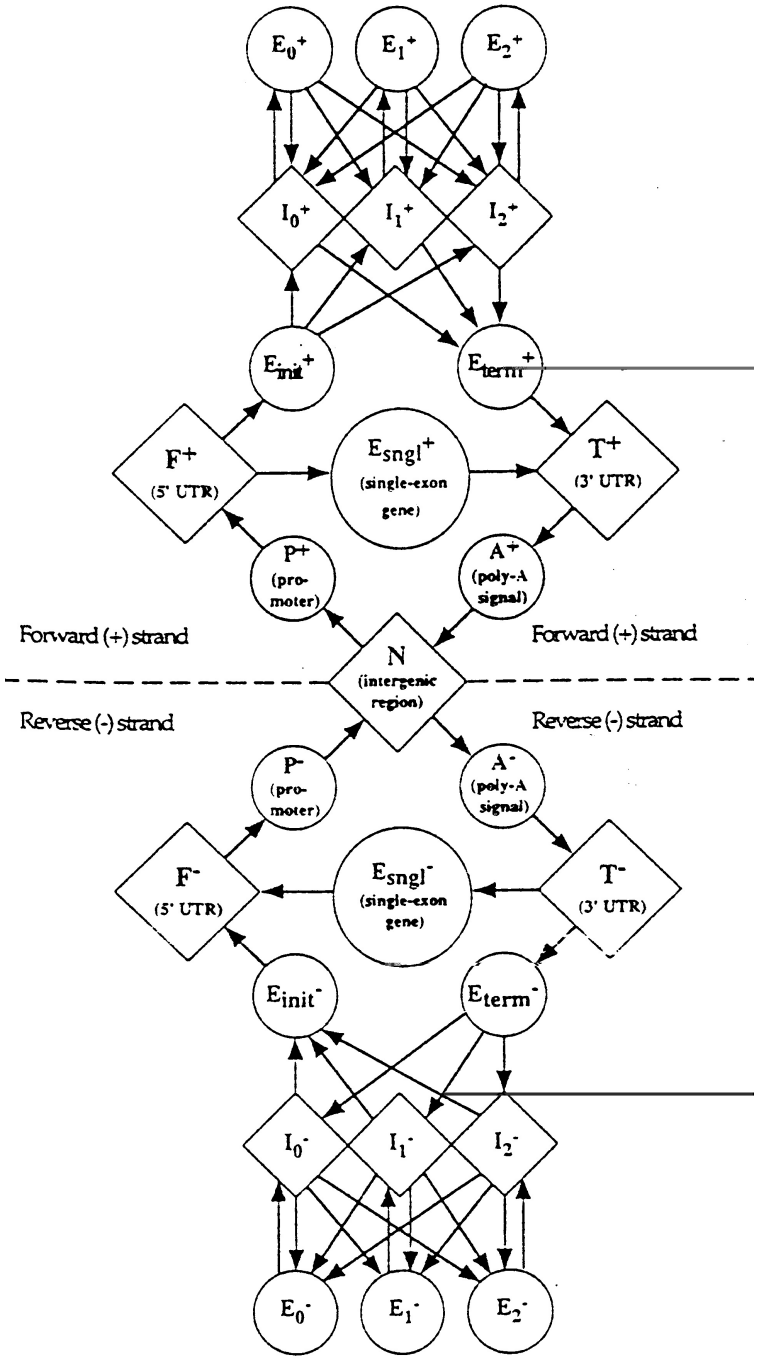


Figure 7.9: GHMM model describing the eukaryotic gene. E states correspond to exons, while I states correspond to introns.

Table 3. Gene density and structure as a function of C + G composition: derivation of initial and transition probabilities

Group	I	II	III	IV
C + G% range	<43	43-51	51-57	>57
Number of genes	65	115	99	101
Est. proportion single-exon genes	0.16	0.19	0.23	0.16
Codelen: single-exon genes (bp)	1130	1251	1304	1137
Codelen: multi-exon genes (bp)	902	908	1118	1165
Introns per multi-exon gene	5.1	4.9	5.5	5.6
Mean intron length (bp)	2069	1086	801	518
Est. mean transcript length (bp)	10866	6504	5781	4833
Isochore	L1 + L2	H1 + H2	H3	H3
DNA amount in genome (Mb)	2074	1054	102	68
Estimated gene number	22100	24700	9100	9100
Est. mean intergenic length	83000	36000	5400	2600
Initial probabilities:				
Intergenic (N)	0.892	0.867	0.540	0.418
Intron (I_0^+, I_1^+, I_2^+, I_0^-, I_1^-, I_2^-)	0.095	0.103	0.338	0.388
5′ Untranslated region (F^+, F^-)	0.008	0.010	0.077	0.122
3′ Untranslated region (T^+, T^-)	0.005	0.011	0.045	0.072

The top portion of the Table shows data from the learning set of 380 genes, partitioned into four groups according to the C + G% content of the GenBank sequence; the middle portion shows estimates of gene density from Duret *et al.* (1995) for isochore compartments corresponding to the four groups above; the bottom portion shows the initial probabilities used by GENSCAN for sequences of each C + G% compositional group, which are estimated using data from the top and middle portions of the Table. All of the values in the top portion are observed values, except the proportion of single-exon genes. Since single-exon genes are typically much shorter than multi-exon genes at the genomic level (due to the absence of introns) and hence easier to sequence completely, they are probably substantially over-represented in the learning set relative to their true genomic frequency; accordingly, the proportion of single-exon genes in each group was estimated (somewhat arbitrarily) to be one half of the observed fraction. Codelen refers to the total number of coding base-pairs per gene. Data for subsets III and IV are estimated from the Duret *et al.* (1995) data for isochore H3 assuming that one-half of the genes and 60% of the amount of DNA sequence in isochore H3 falls into the 51 to 57% C + G range. Mean transcript lengths were estimated assuming an average of 769 bp of 5′UTR and 457 bp of 3′UTR per gene (these values derived from comparison of the "prim_transcript" and "CDS" features of the GenBank annotation in the genes of the learning set). To simplify the model, the initial probabilities of the exon, polyadenylation signal and promoter states are set to zero. All other initial probabilities are estimated from the data shown above, assuming that all features are equally likely to occur on either DNA strand. The initial probability for all intron states was partitioned among the three intron phases according to the observed fraction of each phase in the learning set. Transition probabilities were estimated analogously.

Figure 7.10: Gene density and structure as a function of C+G composition.

Accuracy of GENSCAN for different signal and exon types.

(a) Prediction of individual splice sites and translational signals.

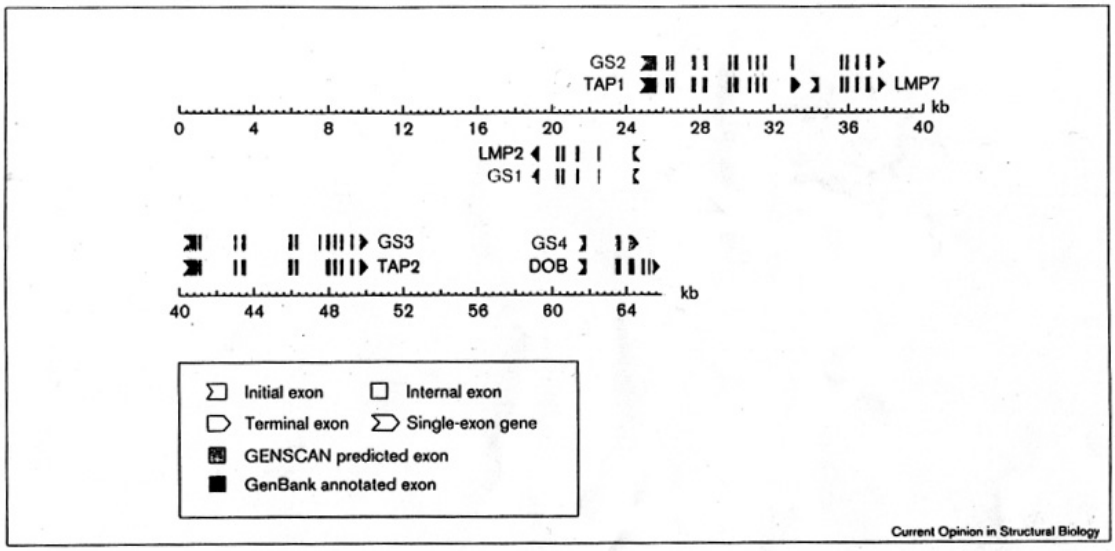
Type of signal	Type of exon	Annotated exons		Predicted exons	
		Number	% Correctly predicted	Number	% Correctly predicted
Initiation	Initial only	570	66	450	84
Termination	Terminal only	570	78	487	91
5′ splice site	Initial only	570	88	450	89
5′ splice site	Internal only	1510	93	1682	89
5′ splice site	Initial and internal	2080	91	2132	89
3′ splice site	Terminal only	570	81	487	92
3′ splice site	Internal only	1510	92	1682	83
3′ splice site	Internal and terminal	2080	89	2169	85

(b) Accuracy for initial, internal and terminal exons.

Exon type	Annotated exons				Predicted exons			
	Number	% Exactly	% Partially	% Missed	Number	% Exactly	% Partially	% Wrong
Initial	570	65	25	9	457	81	9	10
Internal	1510	90	5	4	1707	80	11	8
Terminal	570	76	8	15	509	84	6	8
All types	2650	81	10	8	2678	81	10	9

Accuracy statistics are shown for forward-strand exons predicted by the GENSCAN program [17••], as tested on the Burset and Guigo dataset of 570 vertebrate gene sequences [3••]. **(a)** Accuracy is shown for four types of signals: initiation codons, termination codons and 5′ and 3′ splice sites. For each signal type, the number of (true) sites according to the GenBank CDS (coding sequence) annotation and the percentage of sites predicted correctly by GENSCAN are shown in columns 3 and 4, respectively. Columns 5 and 6 show the number of sites predicted by GENSCAN and the percentage of predicted sites that were correct, respectively. For 5′ and 3′ splice sites, accuracy data are also shown separately for initial versus internal exons, and internal versus terminal exons, respectively. **(b)** Accuracy data at the exon level. The percentages of annotated exons that were predicted exactly (both endpoints correct), predicted partially (one endpoint correct) or missed (not overlapped by a predicted exon) are listed in columns 3, 4 and 5, respectively. Columns 7, 8 and 9 show the percentages of predicted exons that were exactly correct (both endpoints correct), partially correct (one endpoint correct), or wrong (not overlapping an annotated exon), respectively. In addition, five single-exon genes were predicted by GENSCAN in this set (not given a separate row, but included in the totals).

Figure 7.11: Accuracy of GENSCAN for different signal and exon types.



A schematic representation of predicted and annotated genes in a 66 kb portion of the human MHC II region (GenBank accession number X66401) is shown. Annotated genes are labeled according to the names given in the GenBank annotation; predicted genes are labeled GS1 through GS4 as they occur along the sequence. Genes coding on the forward and reverse DNA strands are shown above and below the sequence line, respectively. Predicted exons are shown in grey, annotated exons in black; the shape of the exon indicates its type, as shown in the key. Exon sizes and locations are drawn approximately to scale.

Figure 7.12: GENSCAN results.

Bibliography

[1] C.B. Burge and S. Karlin. Finding the genes in genomic DNA. *J. Mol. Bio*, 268:78–94, 1997.

[2] M. Burset and R. Guigo. Evaluation of gene structure prediction programs. *Genomics*, 34:353–367, 1996.

[3] M.S. Gelfand, A.A. Mironov, and P.A. Pevzner. Gene recognition via spliced sequence alignmnet. In *Proc. Natl Acad Sci USA*, volume 93, pages 9061–9066, 1996.

[4] Sumeet Sobti. Gene prediction, II - DRAFT. WWW: `http://www.cs.washington.edu/education/courses/590bi/98w/lect09.ps`, Feb 1998.

[5] Manu Thambi. Gene prediction, I - DRAFT. WWW: `http://www.cs.washington.edu/education/courses/590bi/98w/lect09.ps`, Jan 1998.