CS481: Bioinformatics Algorithms

Can Alkan
EA224
calkan@cs.bilkent.edu.tr

http://www.cs.bilkent.edu.tr/~calkan/teaching/cs481/

Outline

- Codons
- Discovery of Split Genes
- Exons and Introns
- Splicing
- Open Reading Frames
- Codon Usage
- Splicing Signals
- TestCode

Gene Prediction: Computational Challenge

- Gene: A sequence of nucleotides coding for protein

- Gene Prediction Problem: Determine the beginning and end positions of genes in a genome

Gene Prediction: Computational Challenge

aatgcatgcggctatgctaatgcatgcggctatgctaagctgggatccgatgacaatgcatgc
ggctatgctaatgcatgcggctatgcaagctgggatccgatgactatgctaagctgggatccg
atgacaatgcatgcggctatgctaatgaatggtcttgggatttaccttggaatgctaagctggga
tccgatgacaatgcatgcggctatgctaatgaatggtcttgggatttaccttggaatatgctaatg
catgcggctatgctaagctgggatccgatgacaatgcatgcggctatgctaatgcatgcggct
atgcaagctgggatccgatgactatgctaagctgcggctatgctaatgcatgcggctatgctaa
gctgggatccgatgacaatgcatgcggctatgctaatgcatgcggctatgcaagctgggatcc
tgcggctatgctaatgaatggtcttgggatttaccttggaatgctaagctgggatccgatgacaa
tgcatgcggctatgctaatgaatggtcttgggatttaccttggaatatgctaatgcatgcggctat
gctaagctgggaatgcatgcggctatgctaagctgggatccgatgacaatgcatgcggctat
gctaatgcatgcggctatgcaagctgggatccgatgactatgctaagctgcggctatgctaatg
catgcggctatgctaagctcatgcggctatgctaagctgggaatgcatgcggctatgctaagct
gggatccgatgacaatgcatgcggctatgctaatgcatgcggctatgcaagctgggatccgat
gactatgctaagctgcggctatgctaatgcatgcggctatgctaagctcggctatgctaatgaat
ggtcttgggatttaccttggaatgctaagctgggatccgatgacaatgcatgcggctatgctaat
gaatggtcttgggatttaccttggaatatgctaatgcatgcggctatgctaagctgggaatgcat
gcggctatgctaagctgggatccgatgacaatgcatgcggctatgctaatgcatgcggctatg
caagctgggatccgatgactatgctaagctgcggctatgctaatgcatgcggctatgctaagct
catgcgg

Gene Prediction: Computational Challenge

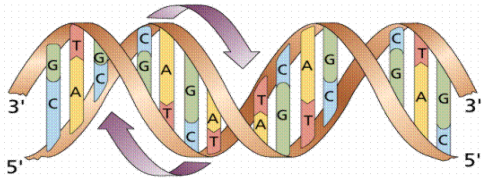
aatgcatgcggctatgctaatgcatgcggctatgctaagctgggatccgatgacaatgcatgc
ggctatgctaatgcatgcggctatgcaagctgggatccgatgactatgctaagctgggatccg
atgacaatgcatgcggctatgctaatgaatggtcttgggatttaccttggaatgctaagctggga
tccgatgacaatgcatgcggctatgctaatgaatggtcttgggatttaccttggaatatgctaatg
catgcggctatgctaagctgggatccgatgacaatgcatgcggctatgctaatgcatgcggct
atgcaagctgggatccgatgactatgctaagctgcggctatgctaatgcatgcggctatgctaa
gctgggatccgatgacaatgcatgcggctatgctaatgcatgcggctatgcaagctgggatcc
tgcggctatgctaatgaatggtcttgggatttaccttggaatgctaagctgggatccgatgacaa
tgcatgcggctatgctaatgaatggtcttgggatttaccttggaatatgctaatgcatgcggctat
gctaagctgggaatgcatgcggctatgctaagctgggatccgatgacaatgcatgcggctat
gctaatgcatgcggctatgcaagctgggatccgatgactatgctaagctgcggctatgctaatg
catgcggctatgctaagctcatgcggctatgctaagctgggaatgcatgcggctatgctaagct
gggatccgatgacaatgcatgcggctatgctaatgcatgcggctatgcaagctgggatccgat
gactatgctaagctgcggctatgctaatgcatgcggctatgctaagctcggctatgctaatgaat
ggtcttgggatttaccttggaatgctaagctgggatccgatgacaatgcatgcggctatgctaat
gaatggtcttgggatttaccttggaatatgctaatgcatgcggctatgctaagctgggaatgcat
gcggctatgctaagctgggatccgatgacaatgcatgcggctatgctaatgcatgcggctatg
caagctgggatccgatgactatgctaagctgcggctatgctaatgcatgcggctatgctaagct
catgcgg

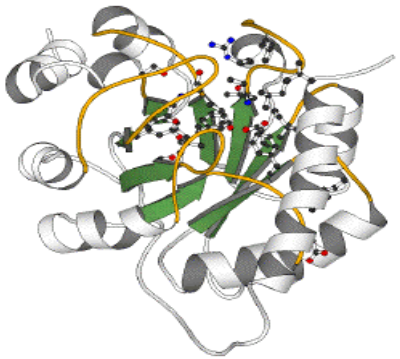
Gene Prediction: Computational Challenge

aatgcatgcggctatgctaatgcatgcggctatgctaagctgggatccgatgacaatgcatgc
ggctatgctaatgcatgcggctatgcaagctgggatccgatgactatgctaagctgggatccg
atgacaatgcatgcggctatgctaatgaatggtcttgggatttaccttggaatgctaagctggga
tccgatgacaatgcatgcggctatgctaatgaatggtcttgggatttaccttggaatatgctaatg
catgcggctatgctaagctgggatccgatgacaatgcatgcggctatgctaatgcatgcggct
atgcaagctgggatccgatgactatgctaagctgcggctatgctaatgcatgcggctatgctaa
gctgggatccgatgacaatgcatgcggctatgctaatgcatgcggctatgcaagctgggatcc
tgcggctatgctaatgaatggtcttgggatttaccttggaatgctaagctgggatccgatgacaa
tgcatgcggctatgctaatgaatggtcttgggatttaccttggaatatgctaatgcatgcggctat
gctaagctgggaatgcatgcggctatgctaagctgggatccgatgacaatgcatgcggctat
gctaatgcatgcggctatgcaagctgggatccgatgactatgctaagctgcggctatgctaatg
catgcggctatgctaagctcatgcggctatgctaagctgggaatgcatgcggctatgctaagct
gggatccgatgacaatgcatgcggctatgctaatgcatgcggctatgcaagctgggatccgat
gactatgctaagctgcggctatgctaatgcatgcggctatgctaagctcggctatgctaatgaat
ggtcttgggatttaccttggaatgctaagctgggatccgatgacaatgcatgcggctatgctaat
gaatggtcttgggatttaccttggaatatgctaatgcatgcggctatgctaagctgggaatgcat
gcggctatgctaagctgggatccgatgacaatgcatgcggctatgctaatgcatgcggctatg
caagctgggatccgatgactatgctaagctgcggctatgctaatgcatgcggctatgctaagct
catgcgg

Gene!

Central Dogma: DNA -> RNA -> Protein





DNA

transcription

RNA

translation

Protein

CCTGAGCCAACTATTGATGAA

CCUGAGCCAACUAUUGAUGAA

PEPTIDE

Codons

- In 1961 Sydney Brenner and Francis Crick discovered frameshift mutations
- Systematically deleted nucleotides from DNA
  - Single and double deletions dramatically altered protein product
  - Effects of triple deletions were minor
  - Conclusion: every triplet of nucleotides, each ***codon***, codes for exactly one amino acid in a protein

The Sly Fox

- In the following string

  THE SLY FOX AND THE SHY DOG

- Delete 1, 2, and 3 nucleotides after the first ‘S’:

  THE SYF OXA NDT HES HYD OG

  THE SFO XAN DTH ESH YDO G

  THE SOX AND THE SHY DOG

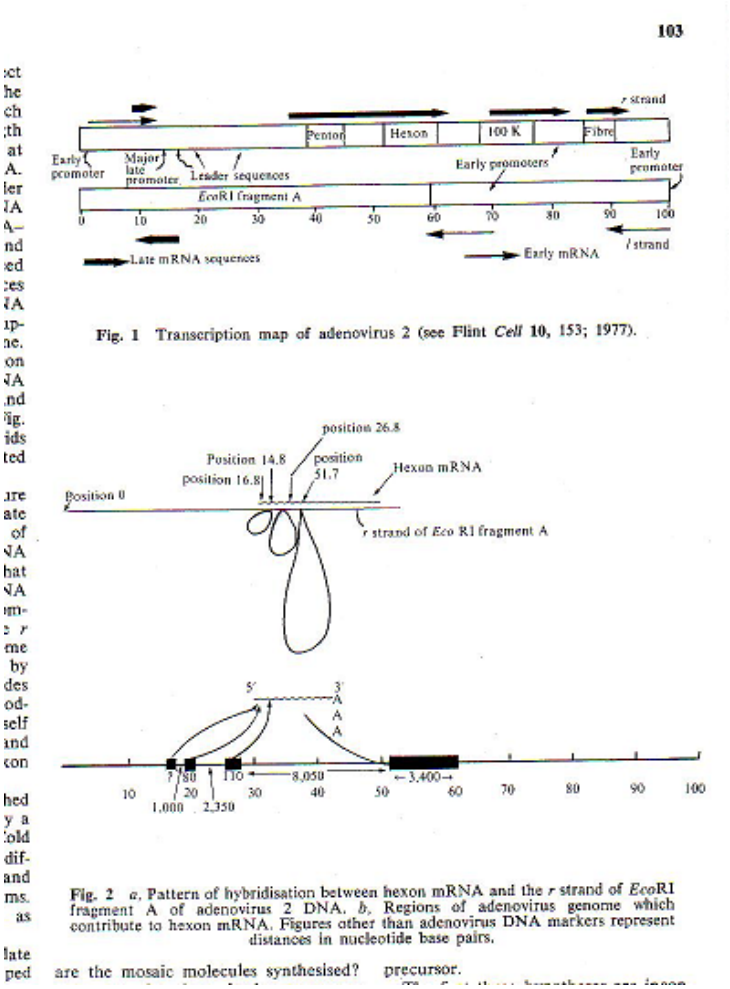
- Which of the above makes the most sense?

Translating Nucleotides into Amino Acids

- Codon: 3 consecutive nucleotides
- $4^3 = 64$ possible codons
- Genetic code is degenerative and redundant
  - Includes start and stop codons
  - An amino acid may be coded by more than one codon

Discovery of Split Genes

- "Adenovirus Amazes at Cold Spring Harbor" (1977, Nature 268) documented "mosaic molecules consisting of sequences complementary to several non-contiguous segments of the viral genome".
- In 1978 Walter Gilbert coined the term **intron** in the Nature paper "Why Genes in Pieces?"



103

r strand
Penton
Hexon
100 K
Fibre
Early promoter
Major late promoter
Leader sequences
Early promoters
Early promoter
EcoRI fragment A
0 10 20 30 40 50 60 70 80 90 100
l strand
Late mRNA sequences
Early mRNA

Fig. 1 Transcription map of adenovirus 2 (see Flint *Cell* **10**, 153; 1977).

position 26.8
Position 14.8
position 16.8
position 51.7
Hexon mRNA
Position 0
r strand of *Eco* RI fragment A
5′
3′
A
A
A
7
80
110
8,050
3,400
10 20 30 40 50 60 70 80 90 100
1,000
2,350

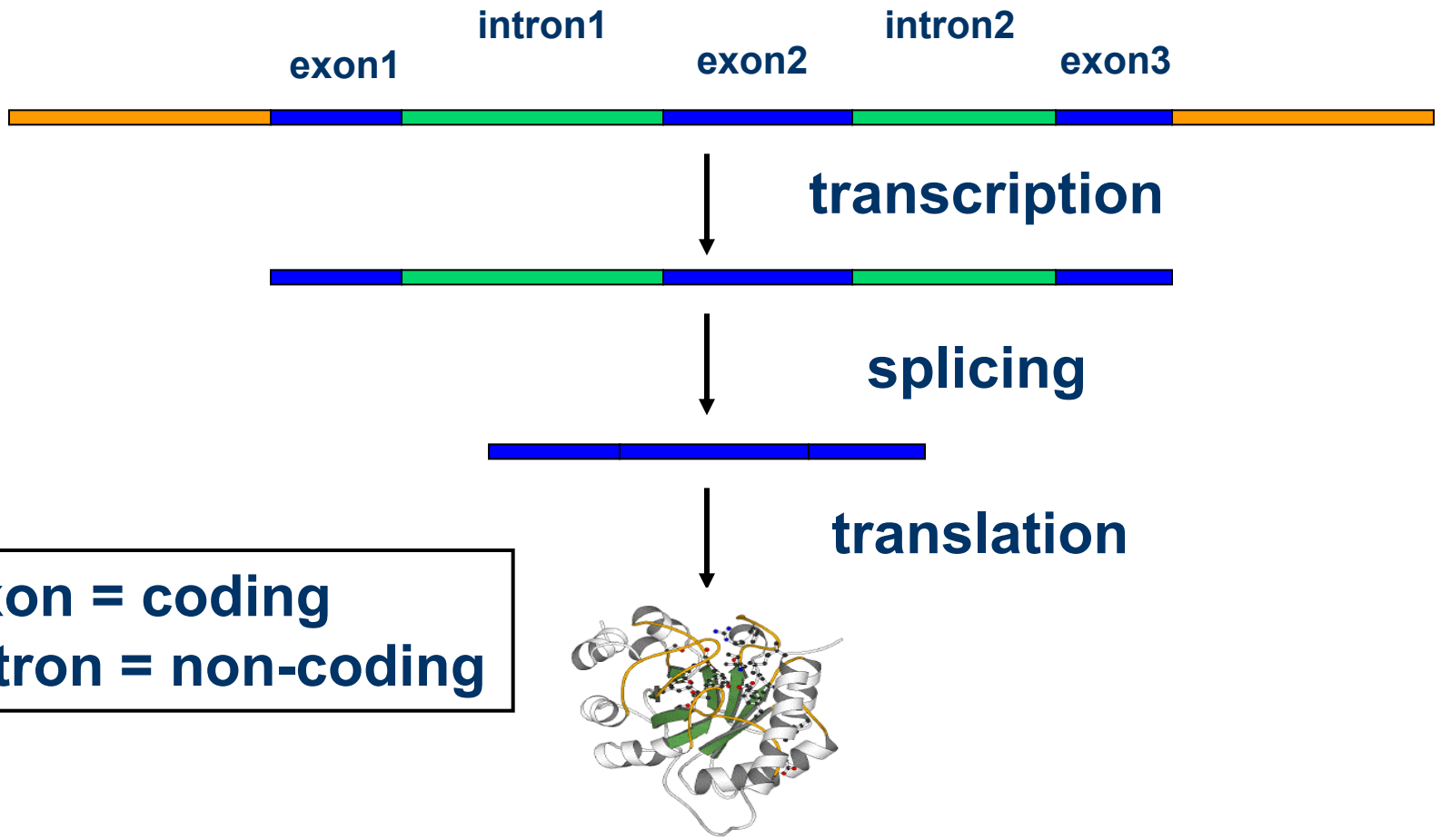
Fig. 2 *a*, Pattern of hybridisation between hexon mRNA and the *r* strand of *Eco*RI fragment A of adenovirus 2 DNA. *b*, Regions of adenovirus genome which contribute to hexon mRNA. Figures other than adenovirus DNA markers represent distances in nucleotide base pairs.

are the mosaic molecules synthesised? precursor.

Exons and introns

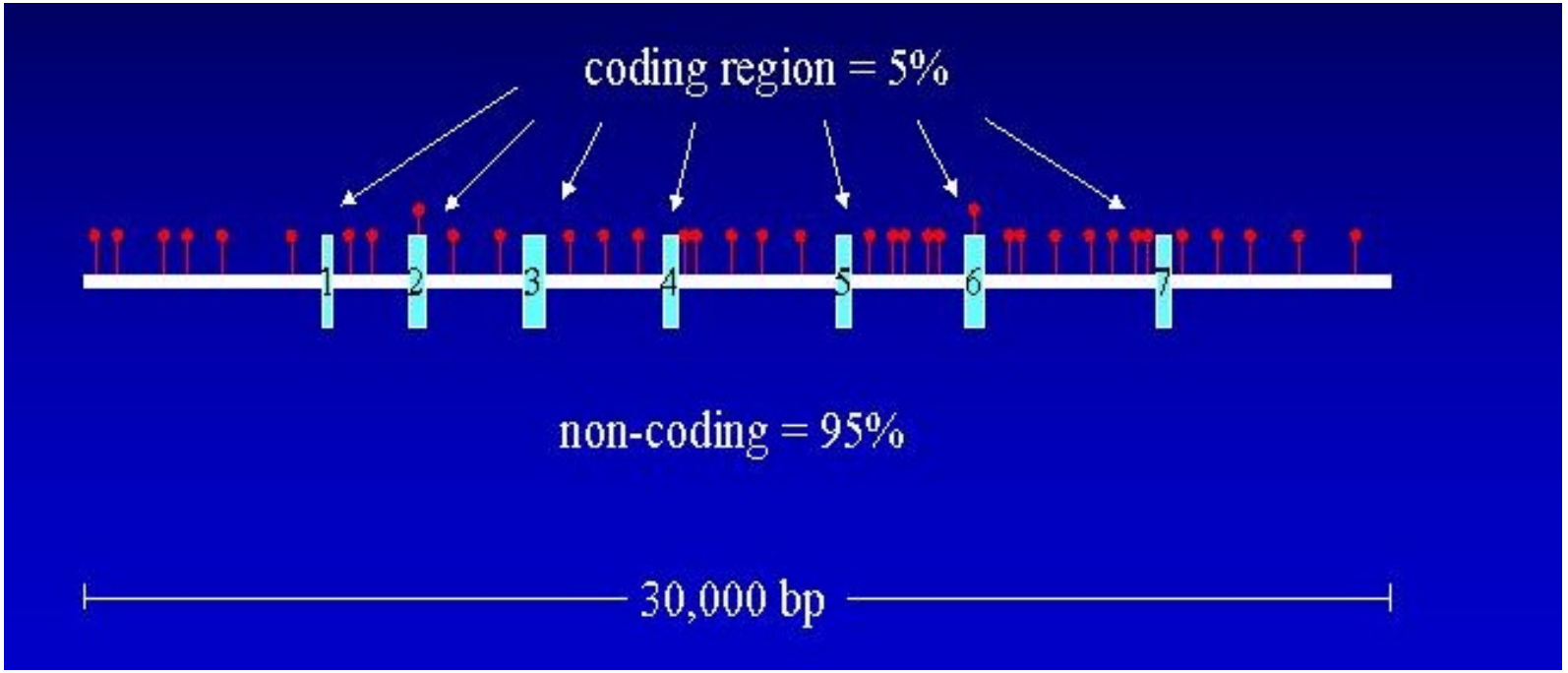
- In eukaryotes, the gene is a combination of coding segments (**exons**) that are interrupted by non-coding segments (**introns**)
- This makes computational gene prediction in eukaryotes even more difficult
- Prokaryotes don’t have introns - Genes in prokaryotes are continuous

Central Dogma and Splicing



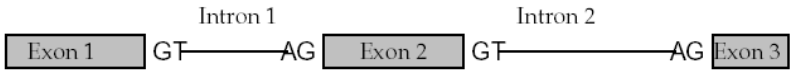
exon = coding
intron = non-coding

Gene Structure

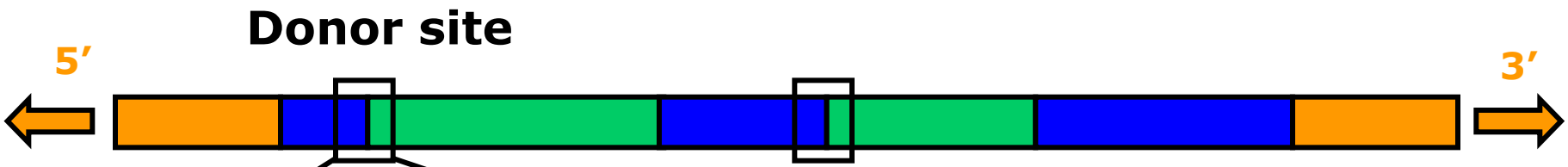


Splicing Signals

- Exons are interspersed with introns and typically flanked by GT and AG

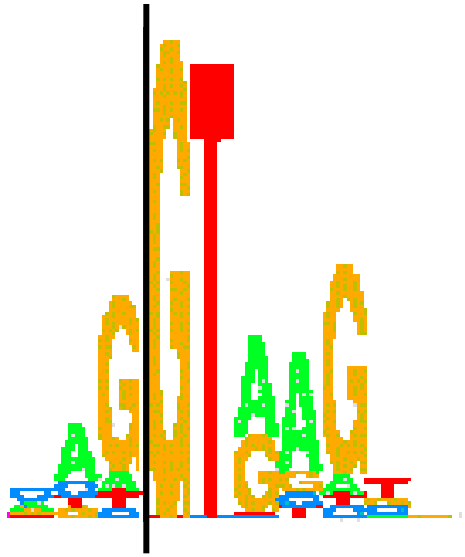


Splice site detection



Position

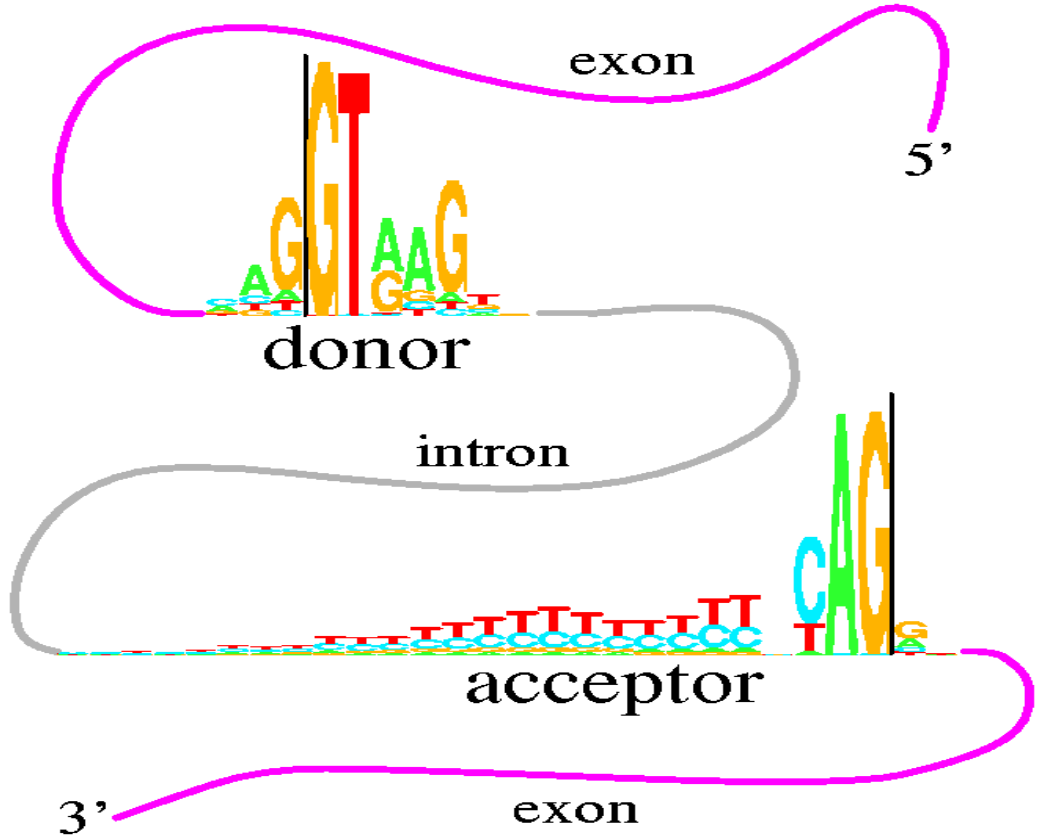
%	-8	…	-2	-1	0	1	2	…	17
A	26	…	60	9	0	1	54	…	21
C	26	…	15	5	0	1	2	…	27
G	25	…	12	78	99	0	41	…	27
T	23	…	13	8	1	98	3	…	25



From lectures by Serafim Batzoglou (Stanford)

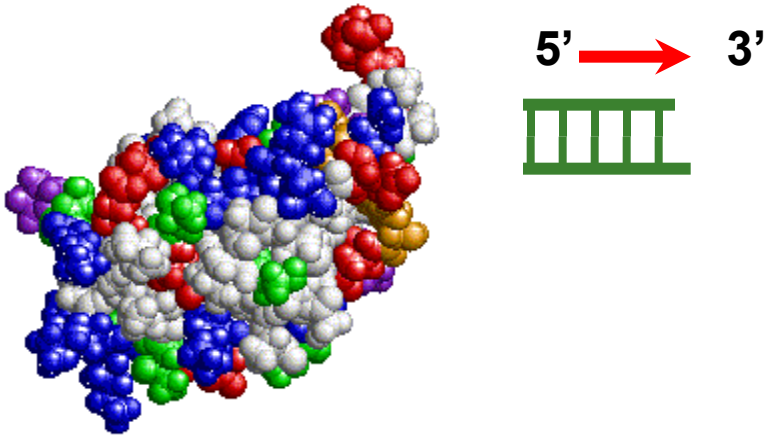
Consensus splice sites

This figure shows two "sequence logos" which represent sequence conservation at the 5' (donor) and 3' (acceptor) ends of human introns. The region between the black vertical bars is removed during mRNA splicing. The logos graphically demonstrate that most of the pattern for locating the intron ends resides on the intron. This allows more codon choices in the protein-coding exons. The logos also show a common pattern "CAG|GT", which suggests that the mechanisms that recognize the two ends of the intron had a common ancestor. See R. M. Stephens and T. D. Schneider, "Features of spliceosome evolution and function inferred from an analysis of the information at human splice sites", J. Mol. Biol., 228, 1124-1136, (1992)



Promoters

- Promoters are DNA segments upstream of transcripts that initiate transcription

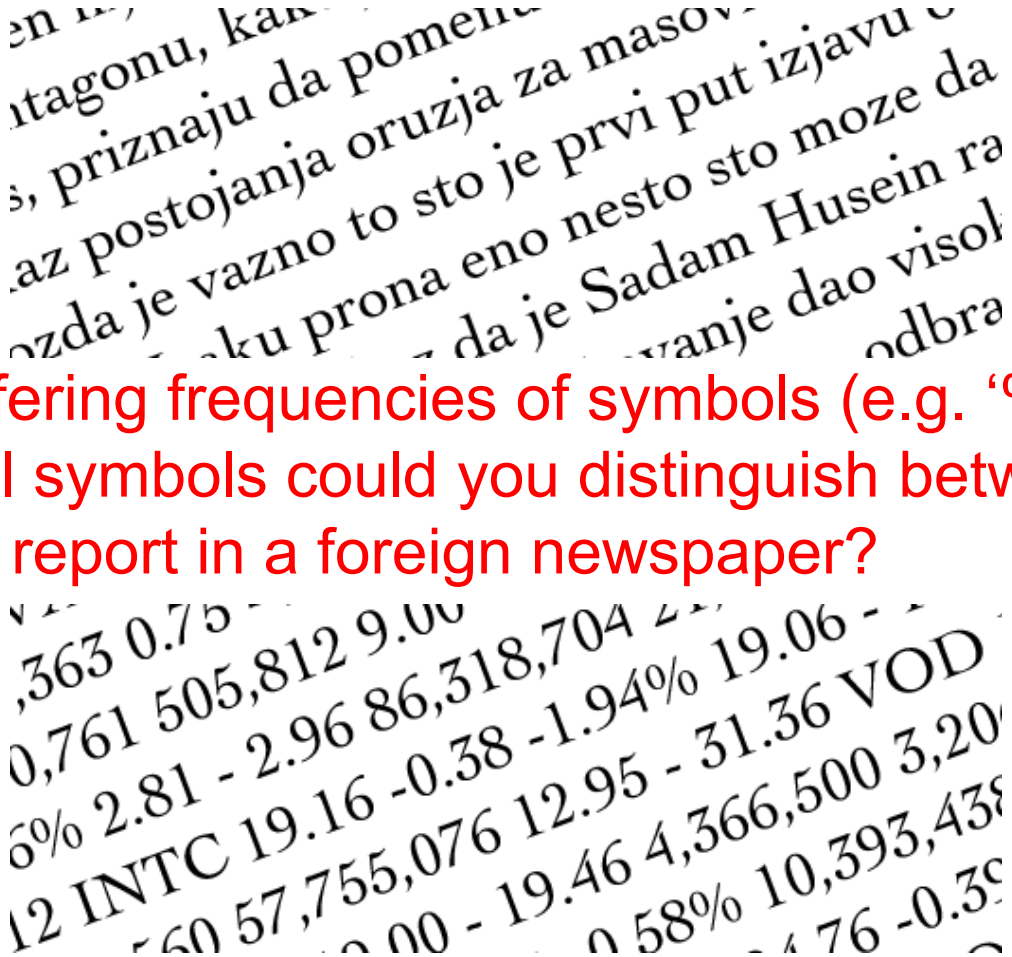


- Promoter *attracts* RNA Polymerase to the transcription start site

Gene Prediction Analogy

- Newspaper written in unknown language
  - Certain pages contain encoded message, say 99 letters on page 7, 30 on page 12 and 63 on page 15.
- How do you recognize the message? You could probably distinguish between the ads and the story (ads contain the “$” sign often)
- Statistics-based approach to Gene Prediction tries to make similar distinctions between exons and introns.

Statistical Approach: Metaphor in Unknown Language

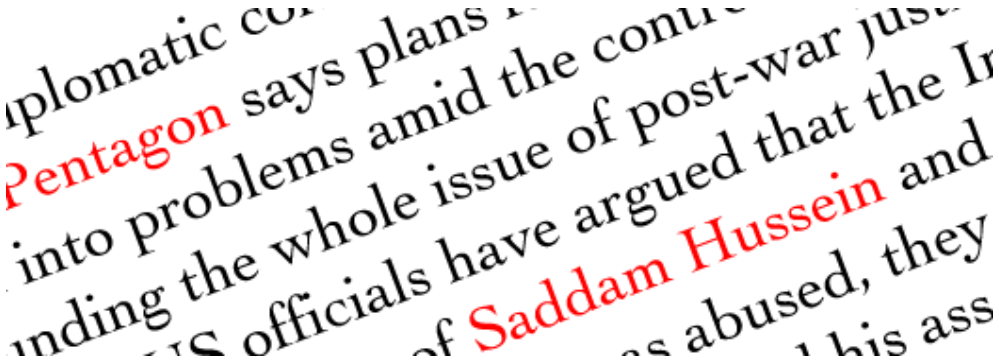


Noting the differing frequencies of symbols (e.g. ‘%’, ‘.’, ‘-’) and numerical symbols could you distinguish between a story and the stock report in a foreign newspaper?

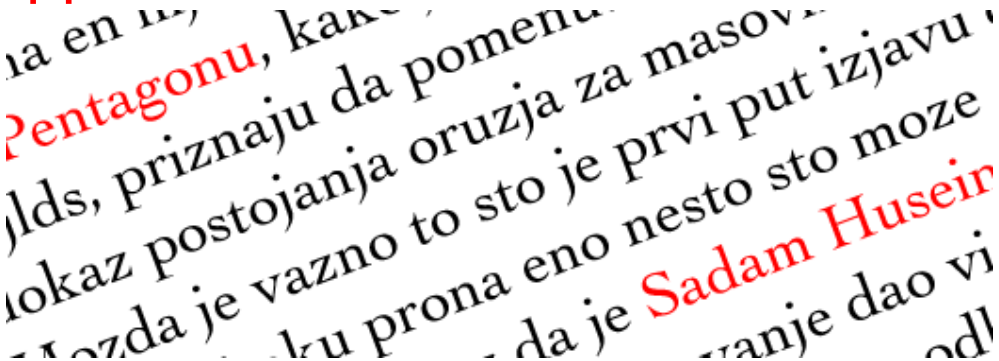
Two Approaches to Gene Prediction

- Statistical: coding segments (exons) have typical sequences on either end and use different subwords than non-coding segments (introns).
- Similarity-based: many human genes are similar to genes in mice, chicken, or even bacteria. Therefore, already known mouse, chicken, and bacterial genes may help to find human genes.

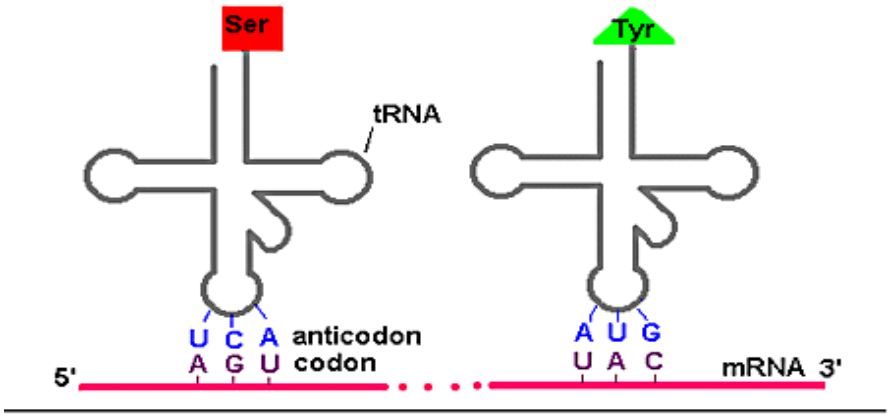
Similarity-Based Approach: Metaphor in Different Languages



If you could compare the day's news in English, side-by-side to the same news in a foreign language, some similarities may become apparent



Genetic Code and Stop Codons



2nd base in codon

1st base in codon	U	C	A	G	3rd base in codon
U	Phe Phe Leu Leu	Ser Ser Ser Ser	Tyr Tyr STOP STOP	Cys Cys STOP Trp	U C A G
C	Leu Leu Leu Leu	Pro Pro Pro Pro	His His Gln Gln	Arg Arg Arg Arg	U C A G
A	Ile Ile Ile Met	Thr Thr Thr Thr	Asn Asn Lys Lys	Ser Ser Arg Arg	U C A G
G	Val Val Val Val	Ala Ala Ala Ala	Asp Asp Glu Glu	Gly Gly Gly Gly	U C A G

The Genetic Code

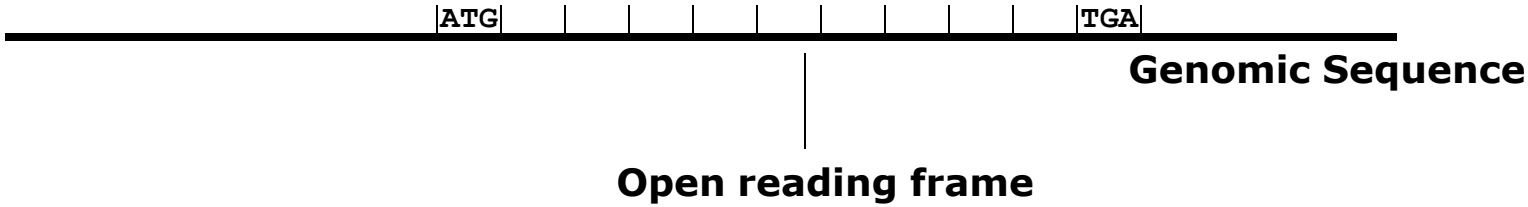
UAA, UAG and UGA correspond to 3 Stop codons that (together with Start codon ATG) delineate Open Reading Frames

Six Frames in a DNA Sequence

- stop codons – TAA, TAG, TGA
- start codons - ATG

Open Reading Frames (ORFs)

- Detect potential coding regions by looking at ORFs
  - A genome of length n is comprised of (n/3) codons
  - Stop codons break genome into segments between consecutive Stop codons
  - The subsegments of these that start from the Start codon (ATG) are ORFs
    - ORFs in different frames may overlap



Long vs. Short ORFs

- Long open reading frames may be a gene
  - At random, we should expect one stop codon every (64/3) ~= 21 codons
  - However, genes are usually much longer than this
- A basic approach is to scan for ORFs whose length exceeds certain threshold
  - This is naïve because some genes (e.g. some neural and immune system genes) are relatively short

Testing ORFs: Codon Usage

- Create a 64-element hash table and count the frequencies of codons in an ORF
- Amino acids typically have more than one codon, but in nature certain codons are more in use
- Uneven use of the codons may characterize a real gene
- This compensate for pitfalls of the ORF length test

Codon Usage in Human Genome

	U		C		A		G	
U	UUU Phe	57	UCU Ser	16	UAU Tyr	58	UGU Cys	45
	UUC Phe	43	UCC Ser	15	UAC Tyr	42	UGC Cys	55
	UUA Leu	13	UCA Ser	13	UAA Stp	62	UGA Stp	30
	UUG Leu	13	UCG Ser	15	UAG Stp	8	UGG Trp	100
C	CUU Leu	11	CCU Pro	17	CAU His	57	CGU Arg	37
	CUC Leu	10	CCC Pro	17	CAC His	43	CGC Arg	38
	CUA Leu	4	CCA Pro	20	CAA Gln	45	CGA Arg	7
	CUG Leu	49	CCG Pro	51	CAG Gln	66	CGG Arg	10
A	AUU Ile	50	ACU Thr	18	AAU Asn	46	AGU Ser	15
	AUC Ile	41	ACC Thr	42	AAC Asn	54	AGC Ser	26
	AUA Ile	9	ACA Thr	15	AAA Lys	75	AGA Arg	5
	AUG Met	100	ACG Thr	26	AAG Lys	25	AGG Arg	3
G	GUU Val	27	GCU Ala	17	GAU Asp	63	GGU Gly	34
	GUC Val	21	GCC Ala	27	GAC Asp	37	GGC Gly	39
	GUA Val	16	GCA Ala	22	GAA Glu	68	GGA Gly	12
	GUG Val	36	GCG Ala	34	GAG Glu	32	GGG Gly	15

Codon Usage in Mouse Genome

AA	codon	/1000	frac
Ser	TCG	4.31	0.05
Ser	TCA	11.44	0.14
Ser	TCT	15.70	0.19
Ser	TCC	17.92	0.22
Ser	AGT	12.25	0.15
Ser	AGC	19.54	0.24
Pro	CCG	6.33	0.11
Pro	CCA	17.10	0.28
Pro	CCT	18.31	0.30
Pro	CCC	18.42	0.31

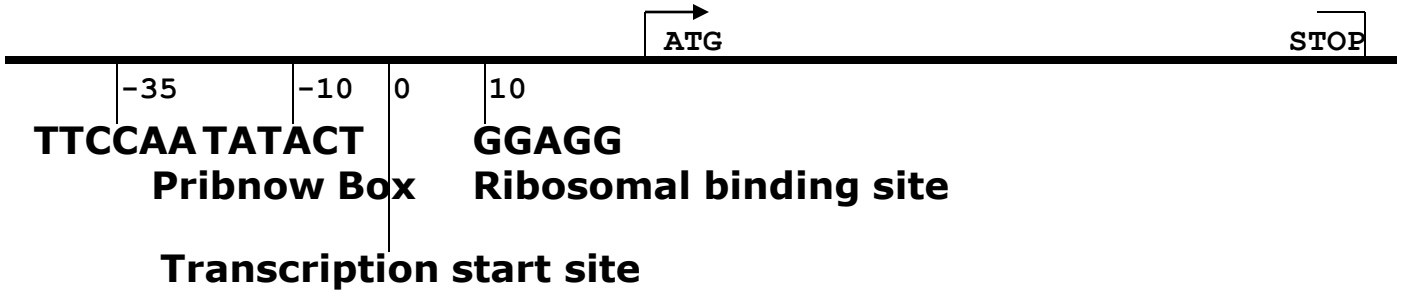
AA	codon	/1000	frac
Leu	CTG	39.95	0.40
Leu	CTA	7.89	0.08
Leu	CTT	12.97	0.13
Leu	CTC	20.04	0.20
Ala	GCG	6.72	0.10
Ala	GCA	15.80	0.23
Ala	GCT	20.12	0.29
Ala	GCC	26.51	0.38
Gln	CAG	34.18	0.75
Gln	CAA	11.51	0.25

Codon Usage and Likelihood Ratio

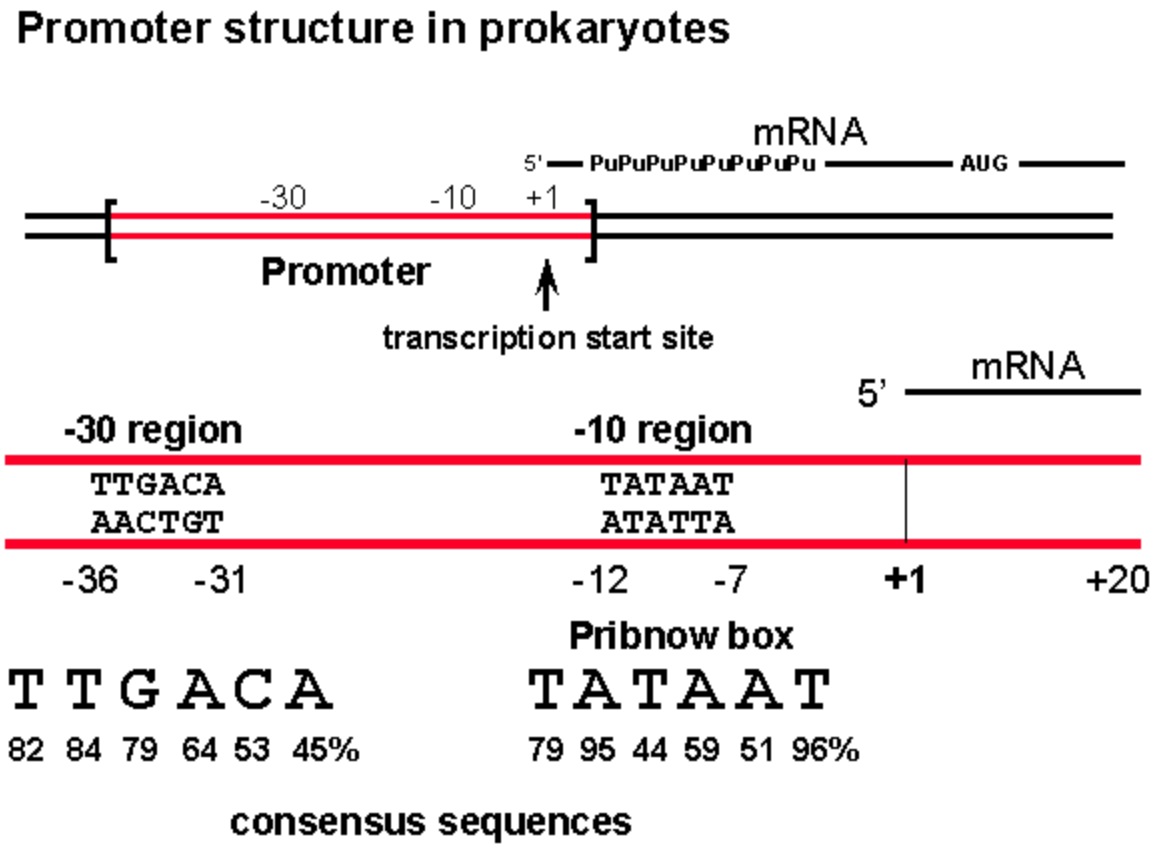
- An ORF is more “believable” than another if it has more “likely” codons
- Do sliding window calculations to find ORFs that have the “likely” codon usage
- Allows for higher precision in identifying true ORFs; much better than merely testing for length.
- However, average vertebrate exon length is 130 nucleotides, which is often too small to produce reliable peaks in the likelihood ratio
- Further improvement: in-frame hexamer count (frequencies of pairs of consecutive codons)

Gene Prediction and Motifs

- Upstream regions of genes often contain motifs that can be used for gene prediction



Promoter Structure in Prokaryotes (E.Coli)

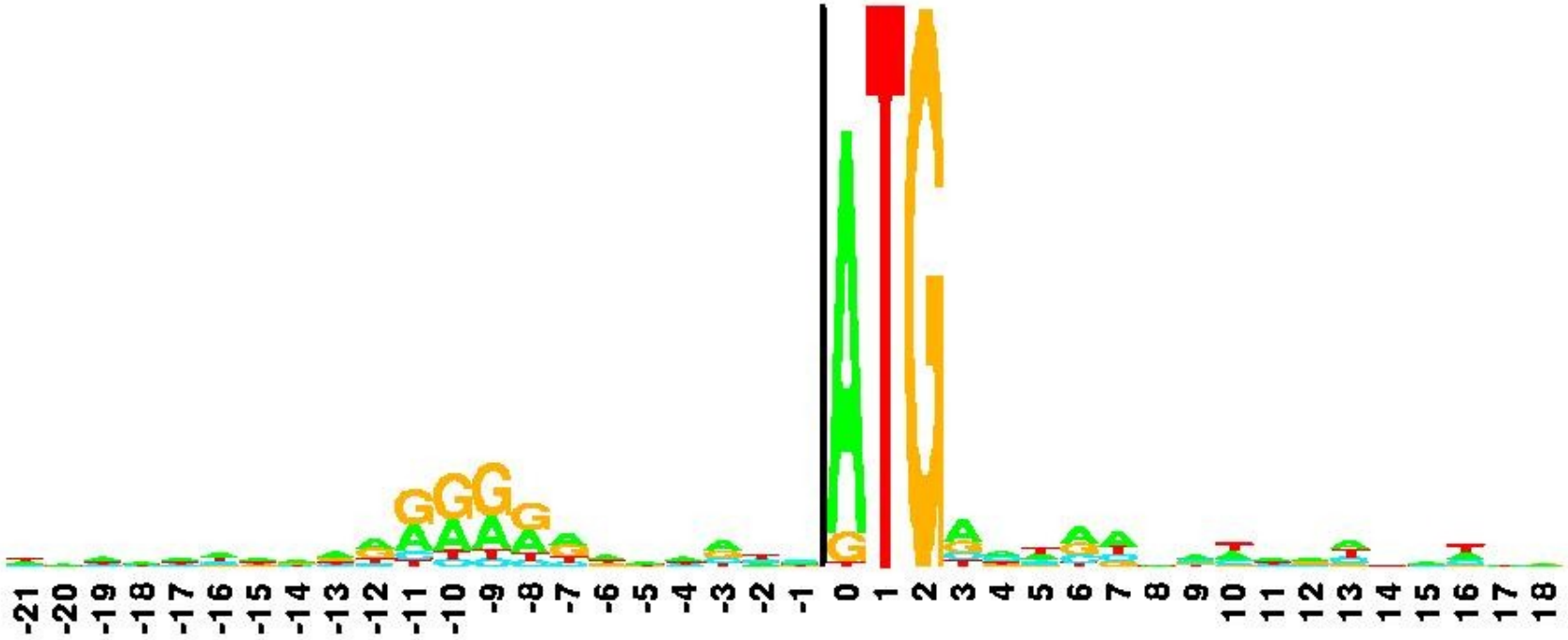


Transcription starts at offset 0.

- Pribnow Box (-10)
- Gilbert Box (-30)
- Ribosomal Binding Site (+10)

Ribosomal Binding Site

1055 E. coli Ribosome binding sites listed in the Miller book

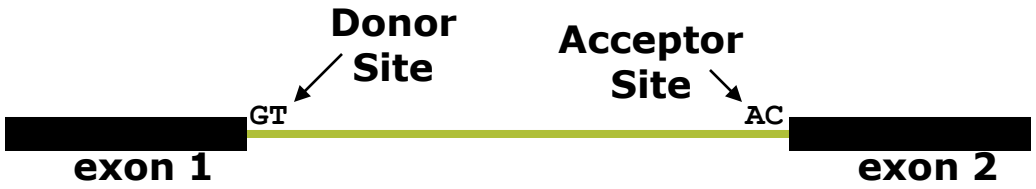


Splicing Signals

- Try to recognize location of splicing signals at exon-intron junctions
  - This has yielded a weakly conserved donor splice site and acceptor splice site
- Profiles for sites are still weak, and lends the problem to the Hidden Markov Model (HMM) approaches, which capture the statistical dependencies between sites

Donor and Acceptor Sites: GT and AG dinucleotides

- The beginning and end of exons are signaled by donor and acceptor sites that usually have GT and AC dinucleotides
- Detecting these sites is difficult, because GT and AC appear very often



TestCode

- Statistical test described by James Fickett in 1982: tendency for nucleotides in coding regions to be repeated with periodicity of 3
  - Judges randomness instead of codon frequency
  - Finds “putative” coding regions, not introns, exons, or splice sites
- TestCode finds ORFs based on compositional bias with a periodicity of three

TestCode Statistics

- Let
  - A1 = Number of A's in positions 1,4,7
  - A2 = Number of A's in positions 2,5,8
  - A3 = Number of A's in positions 3,6,9 …
  - Apos = MAX(A1,A2,A3) / MIN(A1,A2,A3)+1
- Define a window size no less than 200 bp, slide the window the sequence down 3 bases. In each window:
  - Calculate for each base {A, T, G, C} Apos, Cpos, Tpos, Gpos
  - Use these values to obtain a probability from a lookup table (which was a previously defined and determined experimentally with known coding and noncoding sequences)

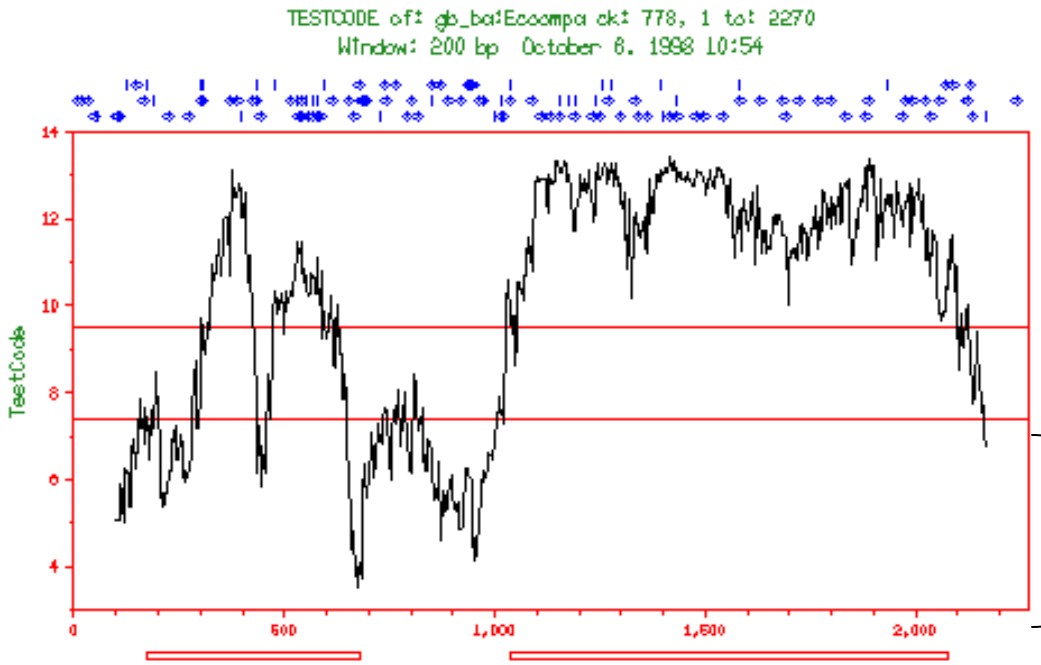
http://emboss.sourceforge.net/apps/release/6.4/emboss/apps/tcode.html

TestCode Statistics (cont'd)

- Probabilities can be classified as indicative of " coding" or "noncoding" regions, or "no opinion" when it is unclear what level of randomization tolerance a sequence carries

- The resulting sequence of probabilities can be plotted

TestCode Sample Output



Coding

No opinion

Non-coding

Popular Gene Prediction Algorithms

- GENSCAN: uses Hidden Markov Models (HMMs)
- TWINSCAN
  - Uses both HMM and similarity (e.g., between human and mouse genomes)

SIMILARITY BASED GENE PREDICTION

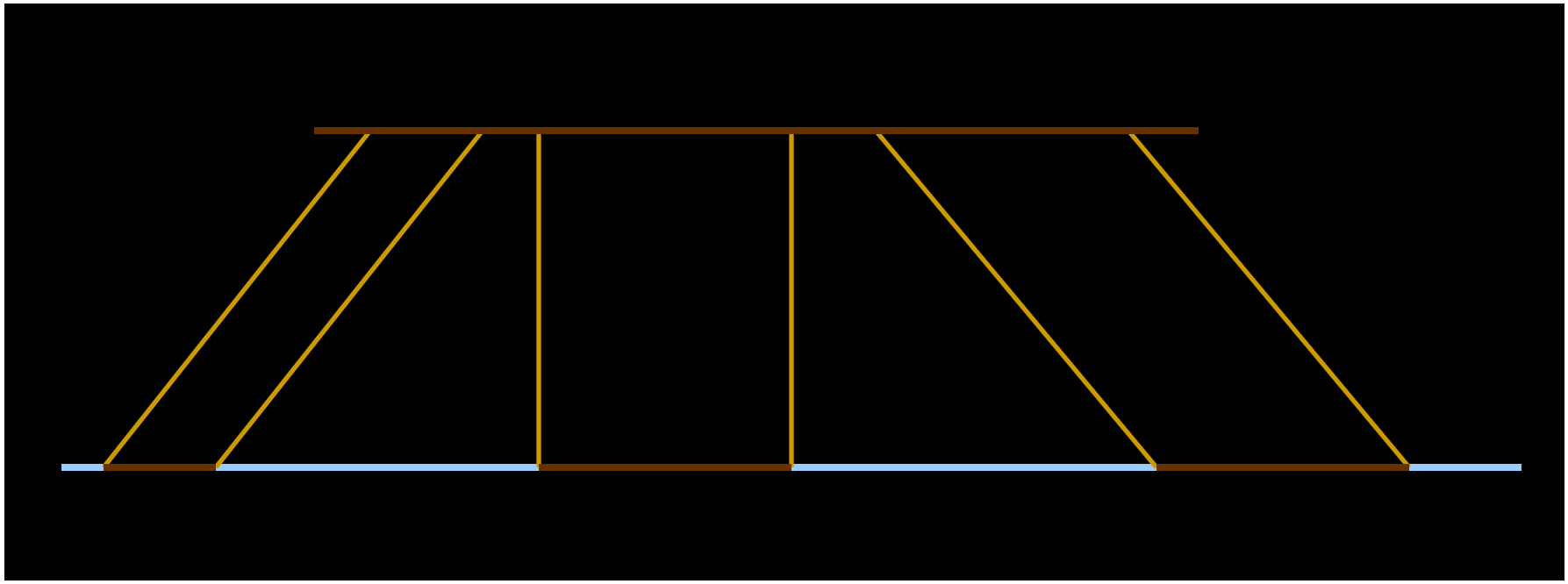
Using Known Genes to Predict New Genes

- Some genomes may be very well-studied, with many genes having been experimentally verified.
- Closely-related organisms may have similar genes
- Unknown genes in one species may be compared to genes in some closely-related species

Similarity-Based Approach to Gene Prediction

- Genes in different organisms are similar
- The similarity-based approach uses known genes in one genome to predict (unknown) genes in another genome
- **Problem:** Given a known gene and an unannotated genome sequence, find a set of substrings of the genomic sequence whose concatenation best fits the gene

Comparing Genes in Two Genomes



- Small islands of similarity corresponding to similarities between exons

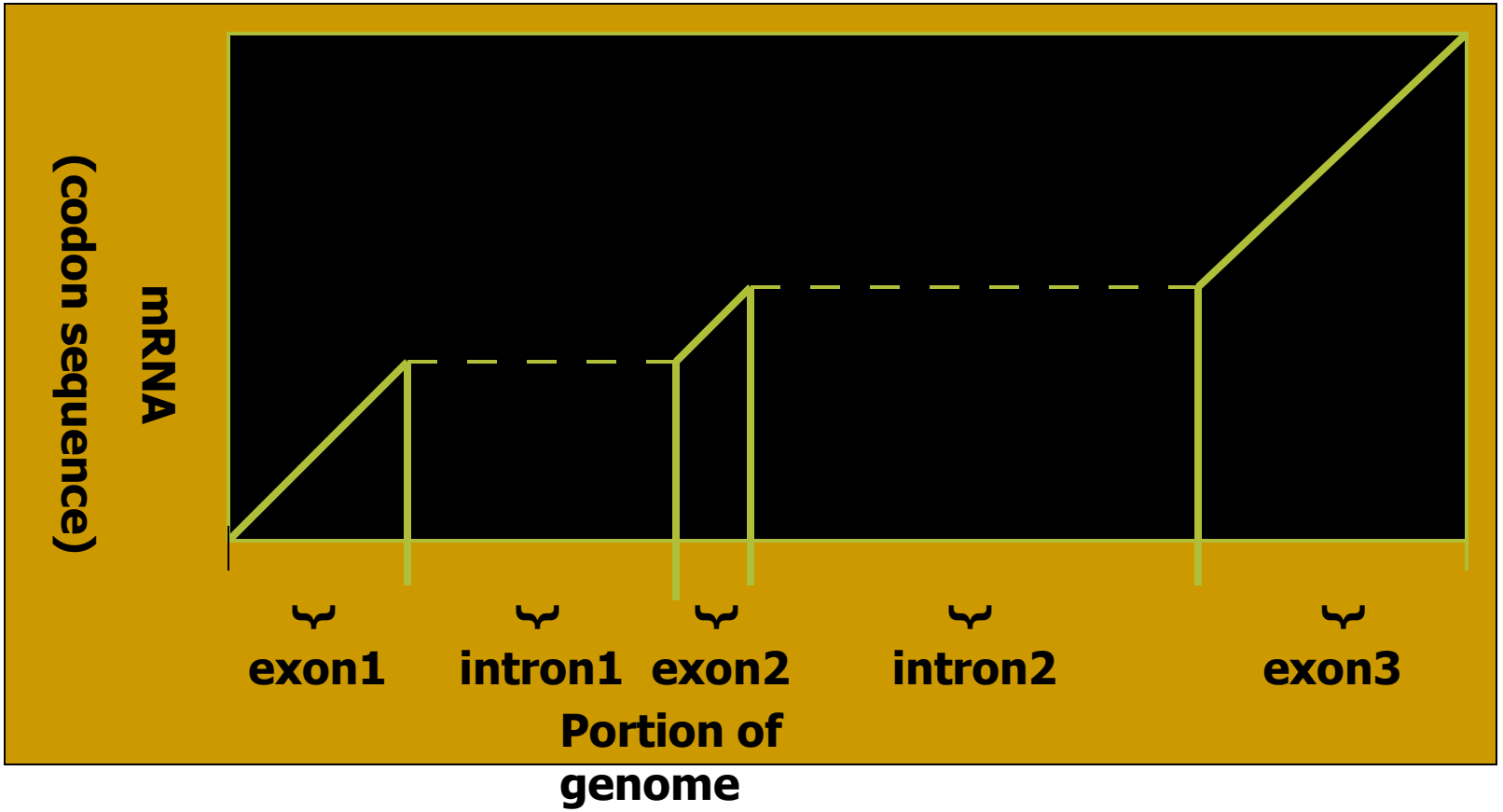
Reverse Translation

- Given a known protein, find a gene in the genome which codes for it
- One might infer the coding DNA of the given protein by reversing the translation process
  - Inexact: amino acids map to > 1 codon
  - This problem is essentially reduced to an alignment problem

Reverse Translation (cont'd)

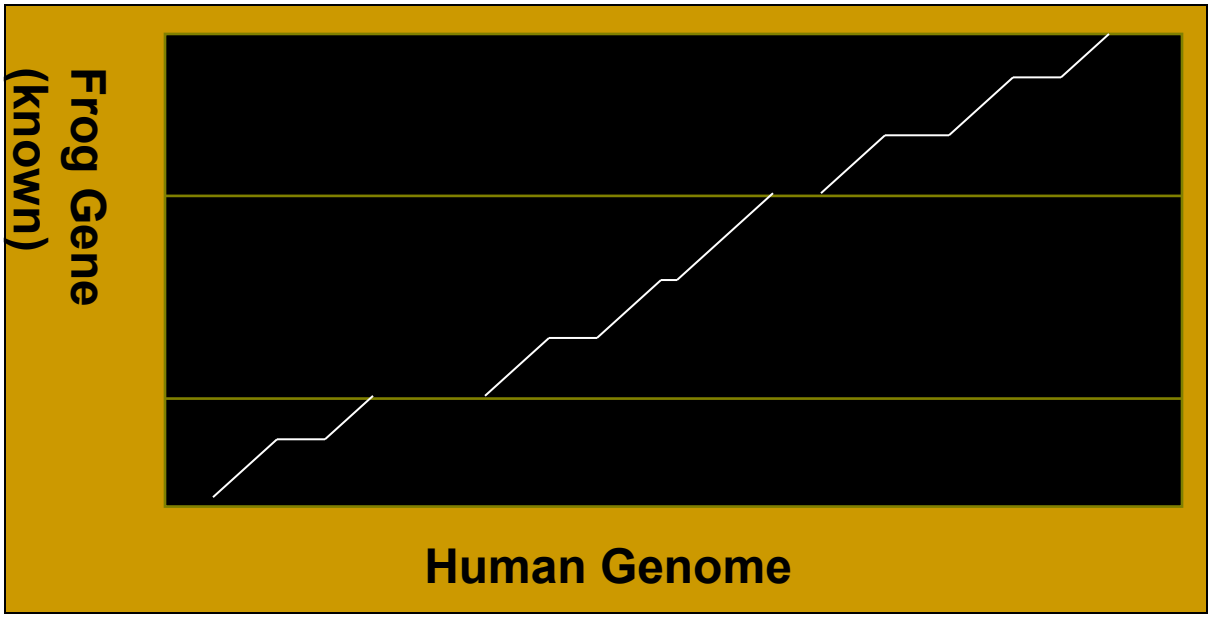
- This reverse translation problem can be modeled as traveling in Manhattan grid with free horizontal jumps
  - Complexity of Manhattan is n^3
- Every horizontal jump models an insertion of an intron
- Problem with this approach: it would match nucleotides pointwise and use horizontal jumps at every opportunity

Comparing Genomic DNA Against mRNA



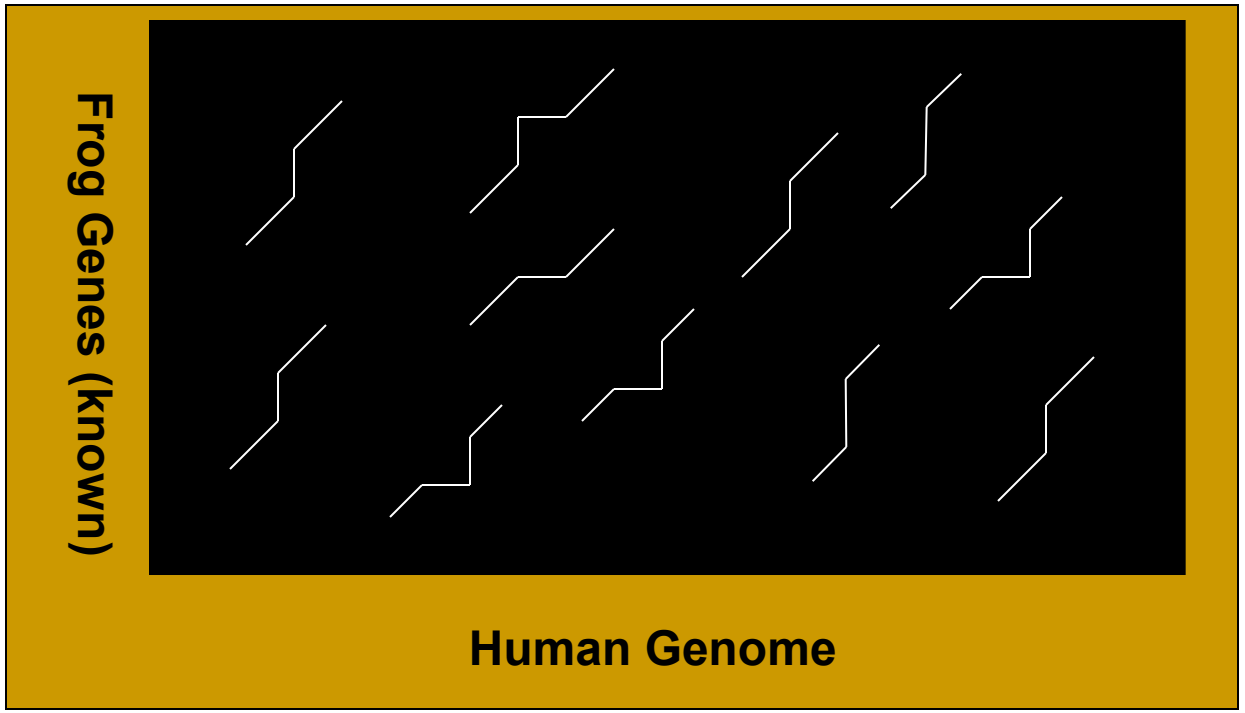
Using Similarities to Find the Exon Structure

- The known frog gene is aligned to different locations in the human genome
- Find the “best” path to reveal the exon structure of human gene



Finding Local Alignments

Use local alignments to find all islands of similarity



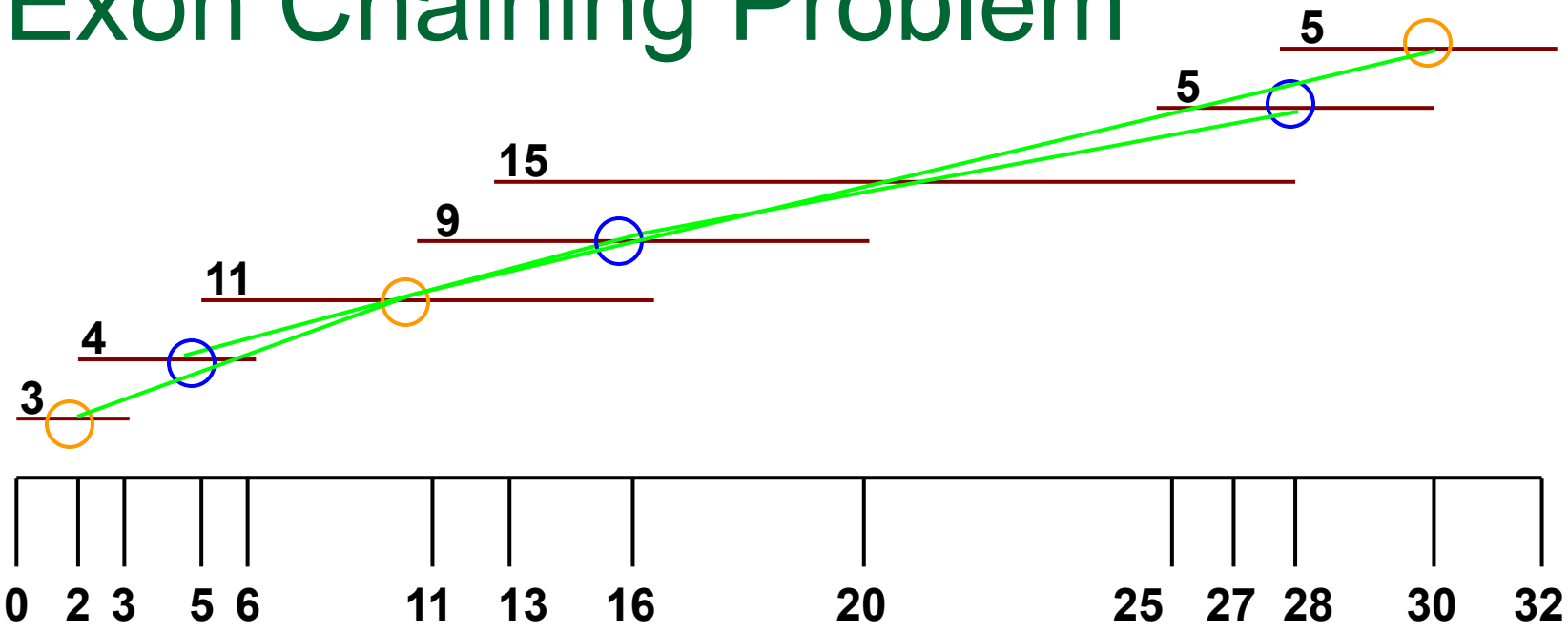
Chaining Local Alignments

- Find substrings that match a given gene sequence (candidate exons)
- Define a candidate exons as

  (l, r, w)

  (*left*, *right*, *weight* defined as score of local alignment)
- Look for a maximum chain of substrings
  - Chain: a set of non-overlapping nonadjacent intervals.

Exon Chaining Problem



- Locate the beginning and end of each interval (*2n* points)
- Find the “best” path

Exon Chaining Problem: Formulation

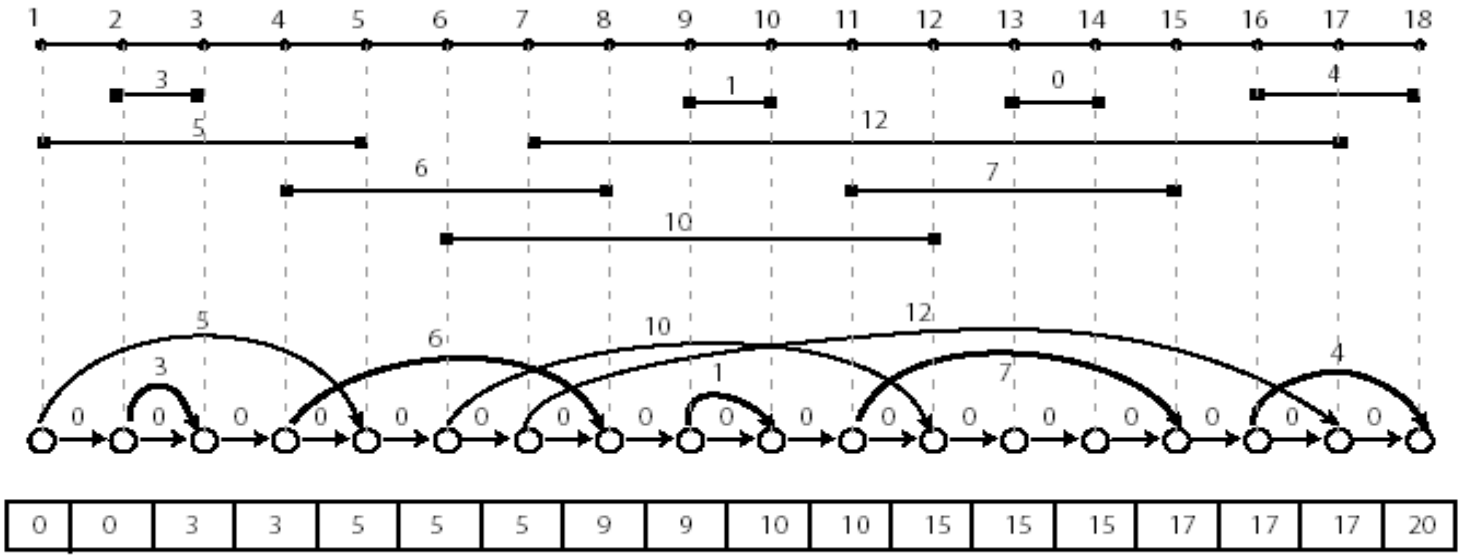
- **Exon Chaining Problem:** Given a set of putative exons, find a maximum set of non-overlapping putative exons
- **Input**: a set of weighted intervals (putative exons)
- **Output**: A maximum chain of intervals from this set

Exon Chaining Problem: Formulation

- **Exon Chaining Problem:** Given a set of putative exons, find a maximum set of non-overlapping putative exons
- **Input**: a set of weighted intervals (putative exons)
- **Output**: A maximum chain of intervals from this set

Would a greedy algorithm solve this problem?

Exon Chaining Problem: Graph Representation



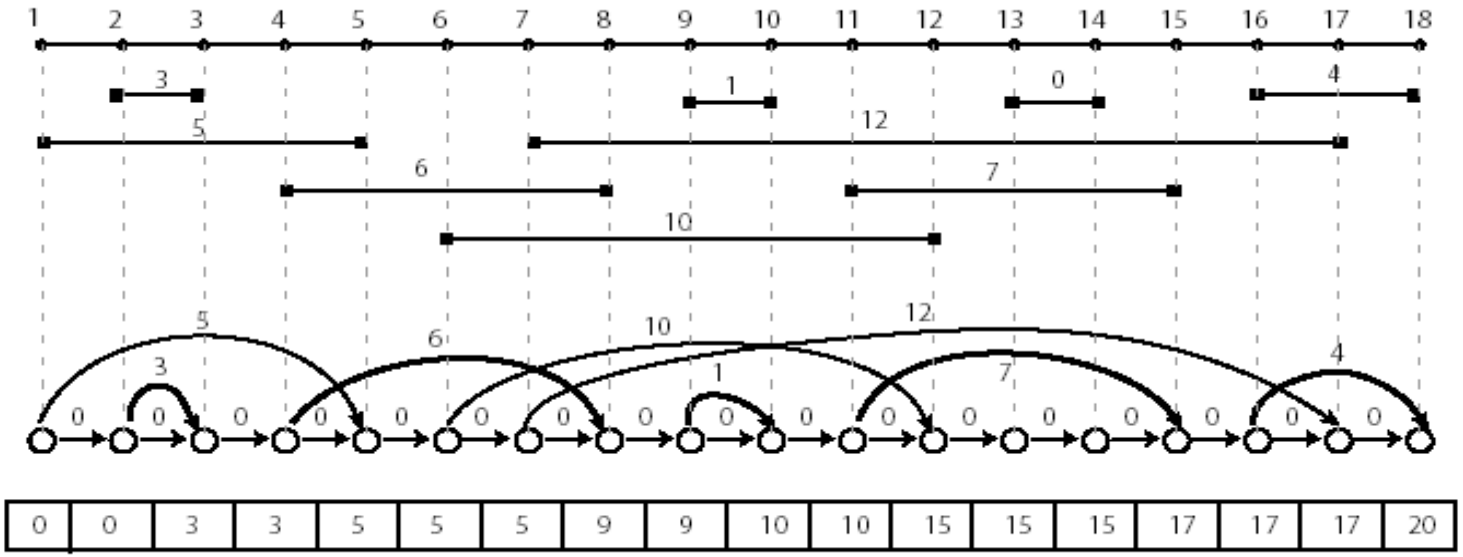
- This problem can be solved with dynamic programming in O(*n*) time.

Exon Chaining Algorithm

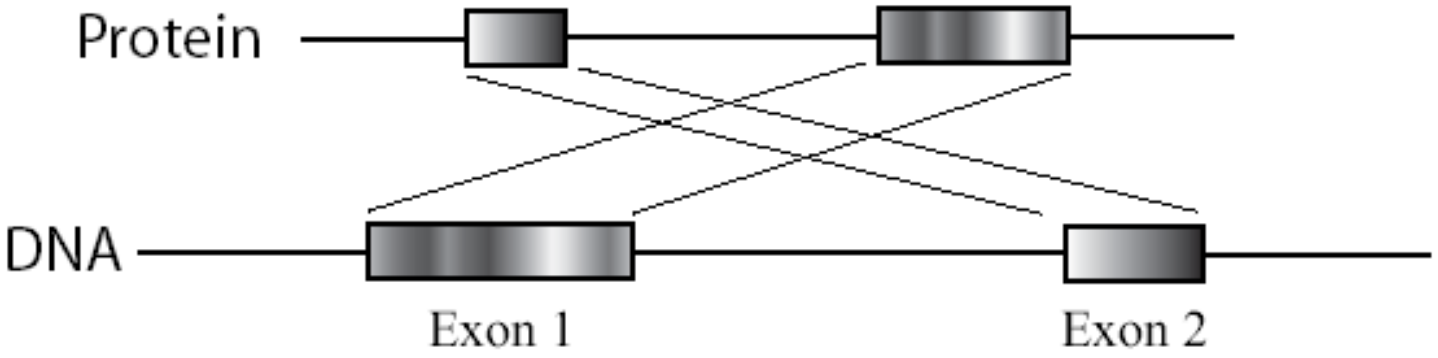
ExonChaining (G, n) //Graph, number of intervals

1 **for** $i \leftarrow$ to $2n$
2 $s_i \leftarrow 0$
3 **for** $i \leftarrow 1$ to $2n$
4 **if** vertex v_i in G corresponds to right end of the interval I
5 $j \leftarrow$ index of vertex for left end of the interval I
6 $w \leftarrow$ weight of the interval I
7 $s_j \leftarrow max\{s_j + w, s_{i-1}\}$
8 **else**
9 $s_i \leftarrow s_{i-1}$
10 **return** s_{2n}

Exon Chaining Problem: Graph Representation



Exon Chaining: Deficiencies



- Poor definition of the putative exon endpoints
- Optimal chain of intervals may not correspond to any valid alignment
  - First interval may correspond to a suffix, whereas second interval may correspond to a prefix
  - Combination of such intervals is not a valid alignment

Gene Prediction: Aligning Genome vs. Genome

- Align entire human and mouse genomes
- Predict genes in both sequences simultaneously as chains of aligned blocks (exons)
- This approach does not assume any annotation of either human or mouse genes.

Gene Prediction Tools

- GENSCAN/Genome Scan
- TwinScan
- Glimmer
- GenMark

The GENSCAN Algorithm

- Algorithm is based on probabilistic model of gene structure similar to *Hidden Markov Models (HMMs).*
- GENSCAN uses a training set in order to estimate the *HMM parameters*, then the algorithm returns the exon structure using maximum likelihood approach standard to many HMM algorithms (*Viterbi* algorithm).
  - Biological input: Codon bias in coding regions, gene structure (start and stop codons, typical exon and intron length, presence of promoters, presence of genes on both strands, etc)
  - Covers cases where input sequence contains no gene, partial gene, complete gene, multiple genes.

GENSCAN Limitations

- Does not use similarity search to predict genes.
- Does not address alternative splicing.
- Could combine two exons from consecutive genes together

GenomeScan

- Incorporates similarity information into GENSCAN: predicts gene structure which corresponds to maximum probability conditional on similarity information
- Algorithm is a combination of two sources of information
  - Probabilistic models of exons-introns
  - Sequence similarity information

TwinScan

- Aligns two sequences and marks each base as gap (-), mismatch (:), match (|), resulting in a new alphabet of 12 letters: Σ {A-, A:, A |, C-, C:, C |, G-, G:, G |, T-, T:, T|}.
- Run Viterbi algorithm using emissions $e_k(b)$ where $b \in$ {A-, A:, A|, …, T|}.

http://www.standford.edu/class/cs262/Spring2003/Notes/ln10.pdf

TwinScan (cont'd)

- The emission probabilities are estimated from from human/mouse gene pairs.
  - Ex. $e_I(x|) < e_E(x|)$ since matches are favored in exons, and $e_I(x-) > e_E(x-)$ since gaps (as well as mismatches) are favored in introns.
  - Compensates for dominant occurrence of poly-A region in introns

Glimmer

- **G**ene **L**ocator and **I**nterpolated **M**arkov **M**odel**ER**
- Finds genes in bacterial DNA
- Uses interpolated Markov Models

The Glimmer Algorithm

- Made of 2 programs
  - **BuildIMM**
    - Takes sequences as input and outputs the Interpolated Markov Models (IMMs)
  - **Glimmer**
    - Takes IMMs and outputs all candidate genes
    - Automatically resolves overlapping genes by choosing one, hence limited
    - Marks “suspected to truly overlap” genes for closer inspection by user

GenMark

- Based on *non-stationary* Markov chain models
- Results displayed graphically with coding vs. noncoding probability dependent on position in nucleotide sequence