

Computational aspects of molecular structure

Lecture 1

Part 1: Introduction

Teresa Przytycka, Ph.D.

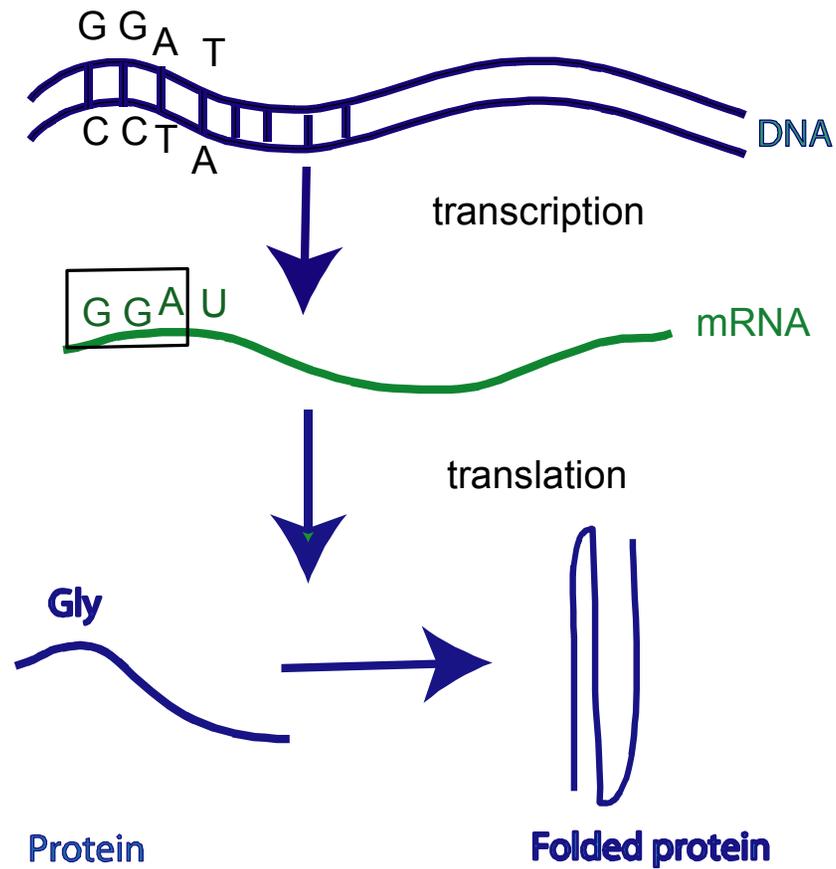
Why molecular structure?

- The function of a molecule is determined by its 3-D structure.

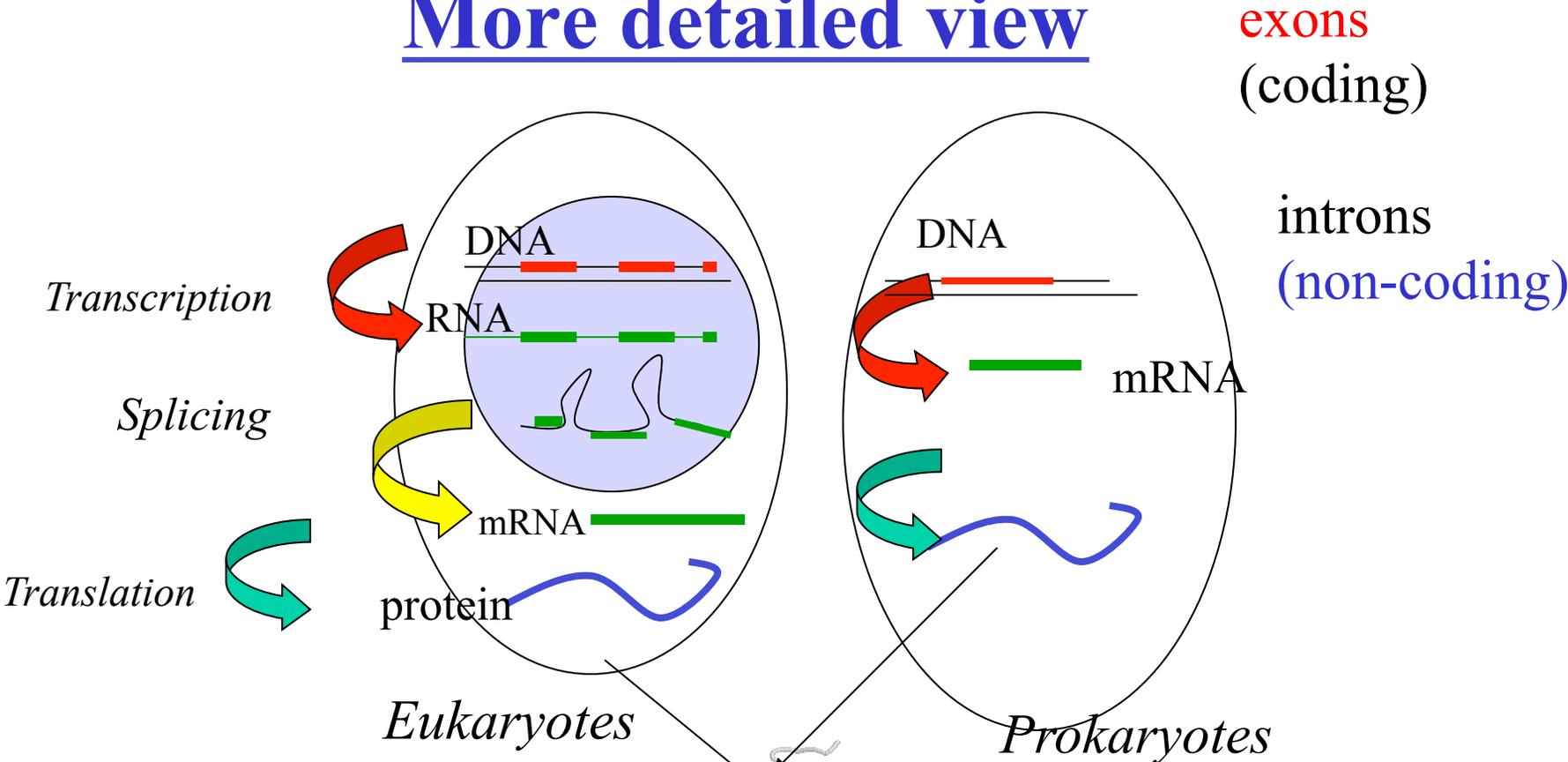
What type of computational aspects?

- Biophysical principles
- Sequence – structure relation
- Structure comparison
- Secondary and 3 D structure prediction (protein and RNA)
- Protein evolution from structural perspective
- Protein function
- Protein-protein interaction

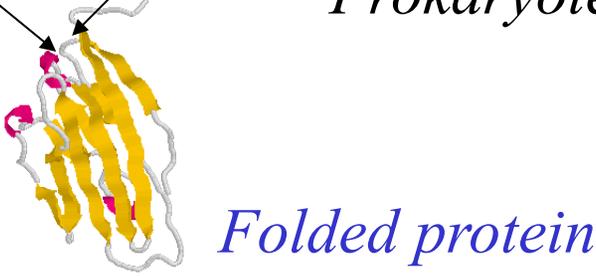
Organization of modern organisms



More detailed view



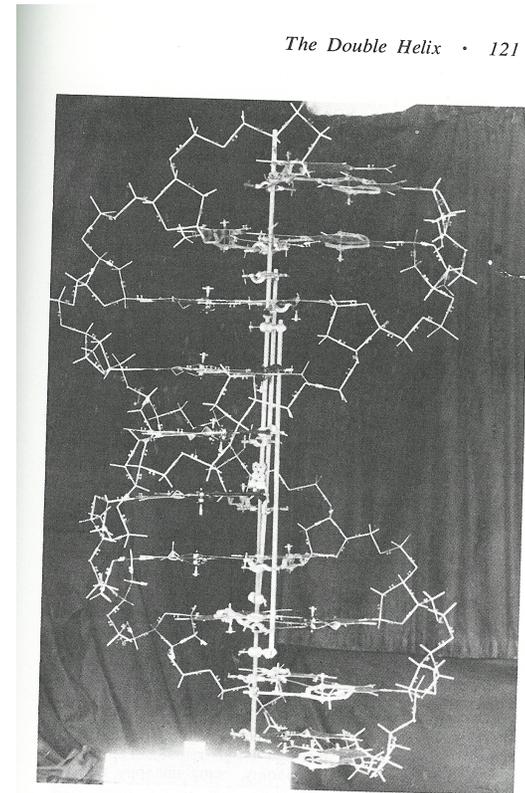
- DNA- sequence of nucleotides {A,T,G,C}
- RNA- sequence of {A,U,G,C}
- Protein- sequence of amino-acids; {A,V,L,I,G,P,....}



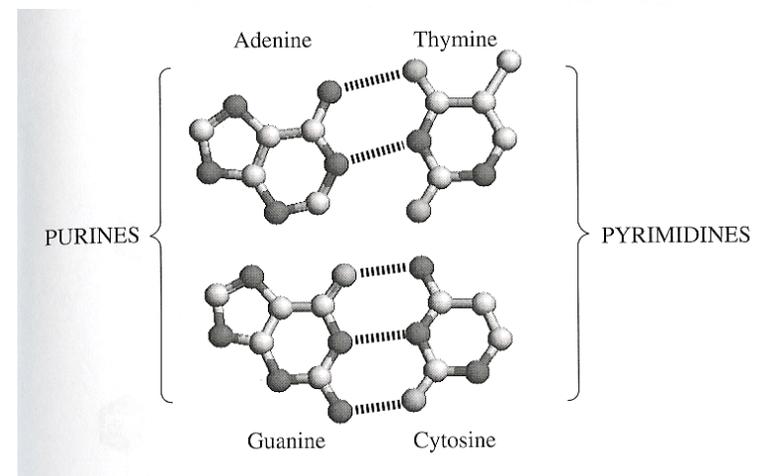
DNA

- A sequence **nucleotides**: adenine (A), cytosine (C), guanine (G) and thymine (T).
- It is **double stranded**: single strand of DNA in one direction is paired to a **complementary strand** forming in 3-dimension **double helix**
- Hydrogen bonding between **complementary base pairs** (the so called Watson-Creek base pairs) hold the two strands together. The base pairs are A-T, G-C
- DNA has **directionality** that corresponds to the direction of translation. The beginning is denoted by 5' and end by 3' .

5' AACTGC 3'
3' TTGACG 5'

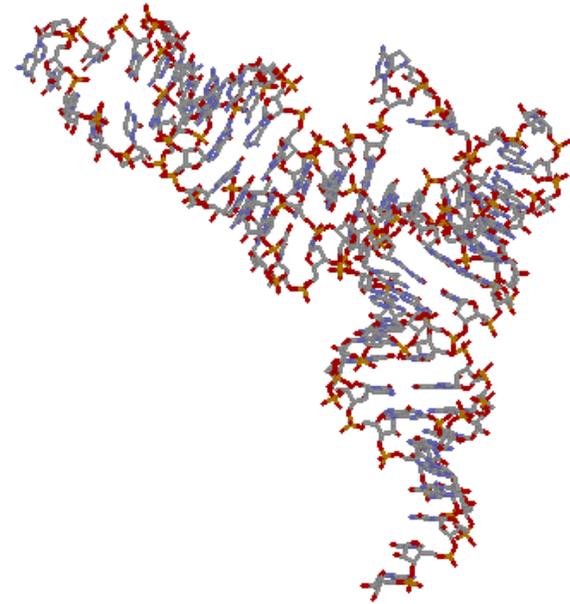


The original demonstration model of the double helix (the scale gives distances in Angstroms).



RNA

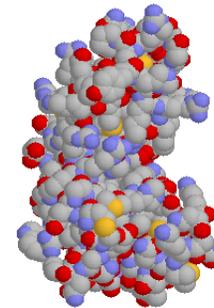
- **Single stranded**
- **Thymine (T) is replaced by uracyl (U).**
- **Base-pairs within the single strand.**



Proteins

- Sequence of amino-acids (word over 20-letter alphabet: A,L,V,...);
- No complementary pairing
- Each amino-acid has its distinct properties

VLSGTGLVLHV.....

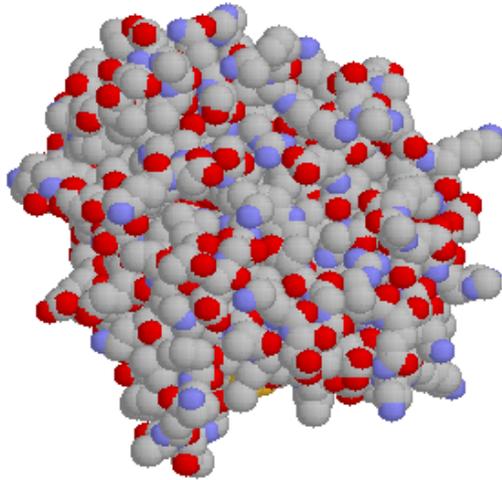


Assumption: *The native conformation is determined by the totality of interatomic interactions and hence the amino acid sequence in a given environment.* (Anfinsen 1960)

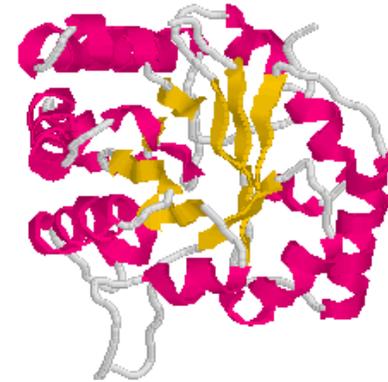
Hypothesis: *The native fold corresponds to the conformation with free energy minimum*

Wishful thinking: *If we understand forces driving the folding process and or we should be able to compute the structure from sequence.*

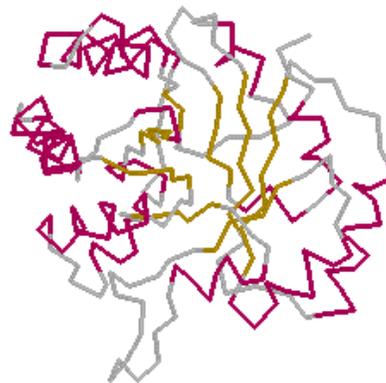
Representations Protein Structure



Space filling model



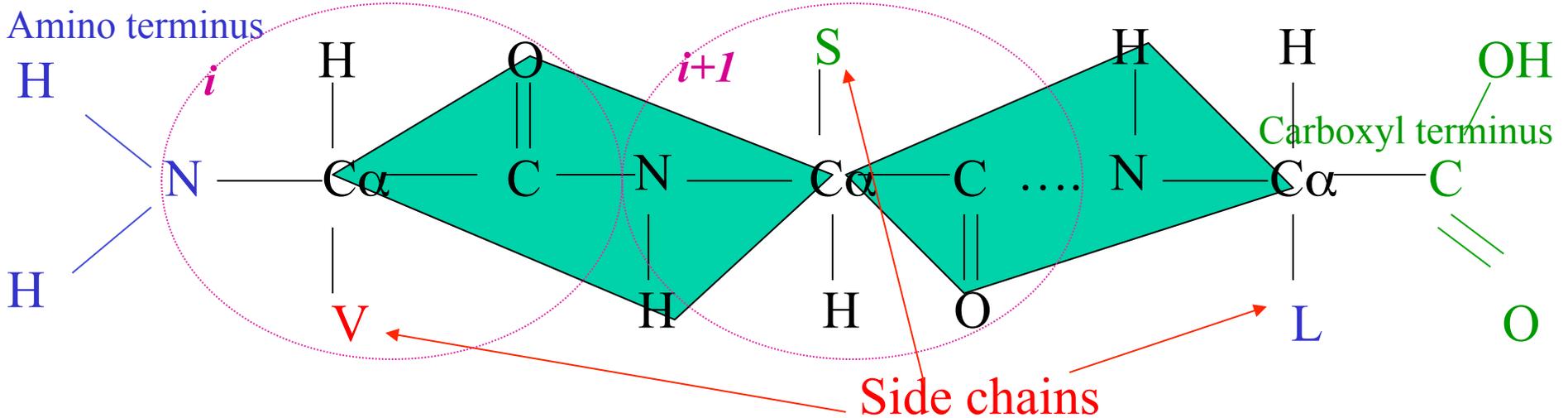
Ribbon diagram



Backbone diagram

Basis for backbone and ribbon representation

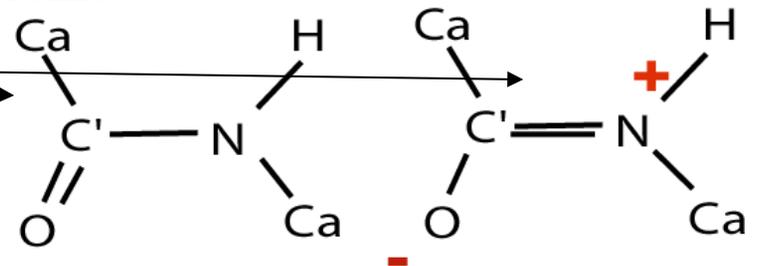
General structure of a polypeptide chain: Example VS...L



Based on crystal structure of molecules containing one or few peptide bonds Pauling discovered that $C' = O$ double bond was longer than expected from while $C' - N$ was shorter than expected

Pauling's explanation: resonance between two extreme structures:

Result: C_{α} , C_i' , O_i , N_{i+1} , H_{i+1} are coplanar



Translation Basics: Genetic Code

- Amino-acids are encoded by triplets of nucleotides called codons
- The genetic code is redundant: there are 64 possible codons and 20 amino-acids + special “termination” codon.

The genetic code

Second position

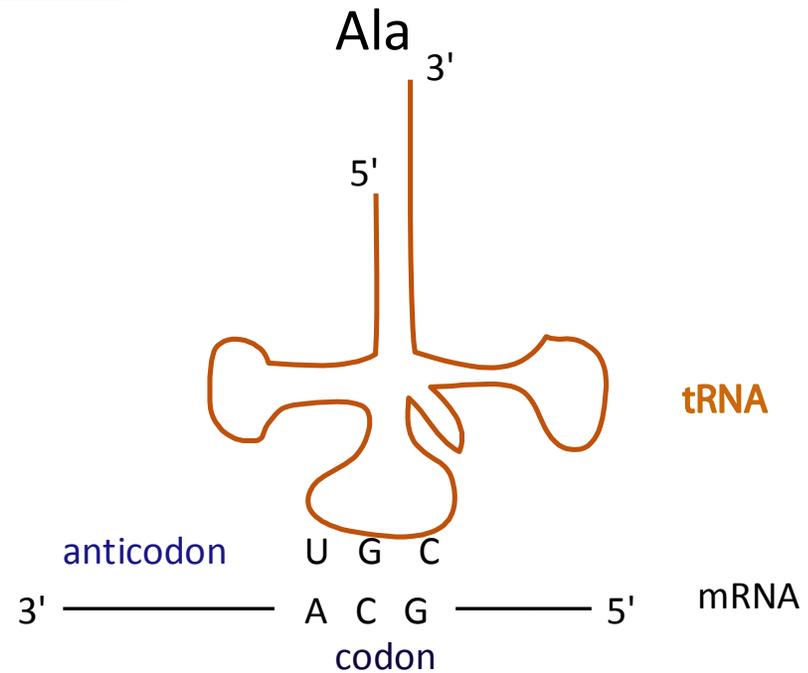
		Second position			
		U	C	A	G
First position	U	UUU phenyl UUC alanine UUA leucine UUG	UCU UCC serine UCA UCG	UAU tyrosine UAC UAA stop UAG	UGU cysteine UGC UGA stop UGG tryptophan
	C	CUU CUC leucine CUA CUG	CCU CCC proline CCA CCG	CAU histidine CAC CAA glutamine CAG	CGU CGC arginine CGA CGG
	A	AUU AUC isoleucine AUA AUG methionine	ACU ACC threonine ACA ACG	AAU asparagine AAC AAA lysine AAG	AGU serine AGC AGA arginine AGG
	G	GUU GUC valine GUA GUG	GCU GCC alanine GCA GCG	GAU aspartic GAC acid GAA glutamic GAG acid	GGU GGC glycine GGA GGG

Third position
U C A G
U C A G
U C A G
U C A G

RNA

t-RNA Fundamental in translation is the so called transfer RNA (tRNA) molecules – linked to a specific amino-acid on one side and containing the triple complementary to the codon (the anticodon) on the other side.

m-RNA (transcription)
“functional” RNA



**Basic Methods for sequence
Alignment and Similarity Search
(overview)**

**Computational aspects of molecular
structure**

Lecture 1, Part b

Teresa Przytycka, Ph.D.

Assumptions:

- Biological sequences evolved by evolution.
- Evolutionary related sequences are likely to have related functions.
- We assume that evolution of biological sequences proceeds by:
 - Substitutions
 - Insertions/Deletions
- Larger rearrangements of sequences are also possible but are modeled using different methods

Sequence alignment

- Write one sequence along the other so that to expose any similarity between the sequences. Each element of a sequence is either placed alongside of corresponding element in the other sequence or alongside a special “gap” character
- Example: TGKGI and AGKVGL can be aligned as
TGK - GI
AGKVGL
- Is there a better alignment? How can we compare the “goodness” of two alignments.
- We need to have:
 - A way of scoring an alignment
 - A way of computing maximum score alignment.

Percent Identity

Identity score – Exact matches receive score of 1 and non-exact matches score of 0

AVLILKQW

AVLI I LQ T

1 1 1 1 0 0 1 0 = 5 (Score of the alignment under “identity”)

Percent identity: $\text{identity_score} / \text{length_of_the_shorter_protein}$

Arguments for more sophisticated scoring

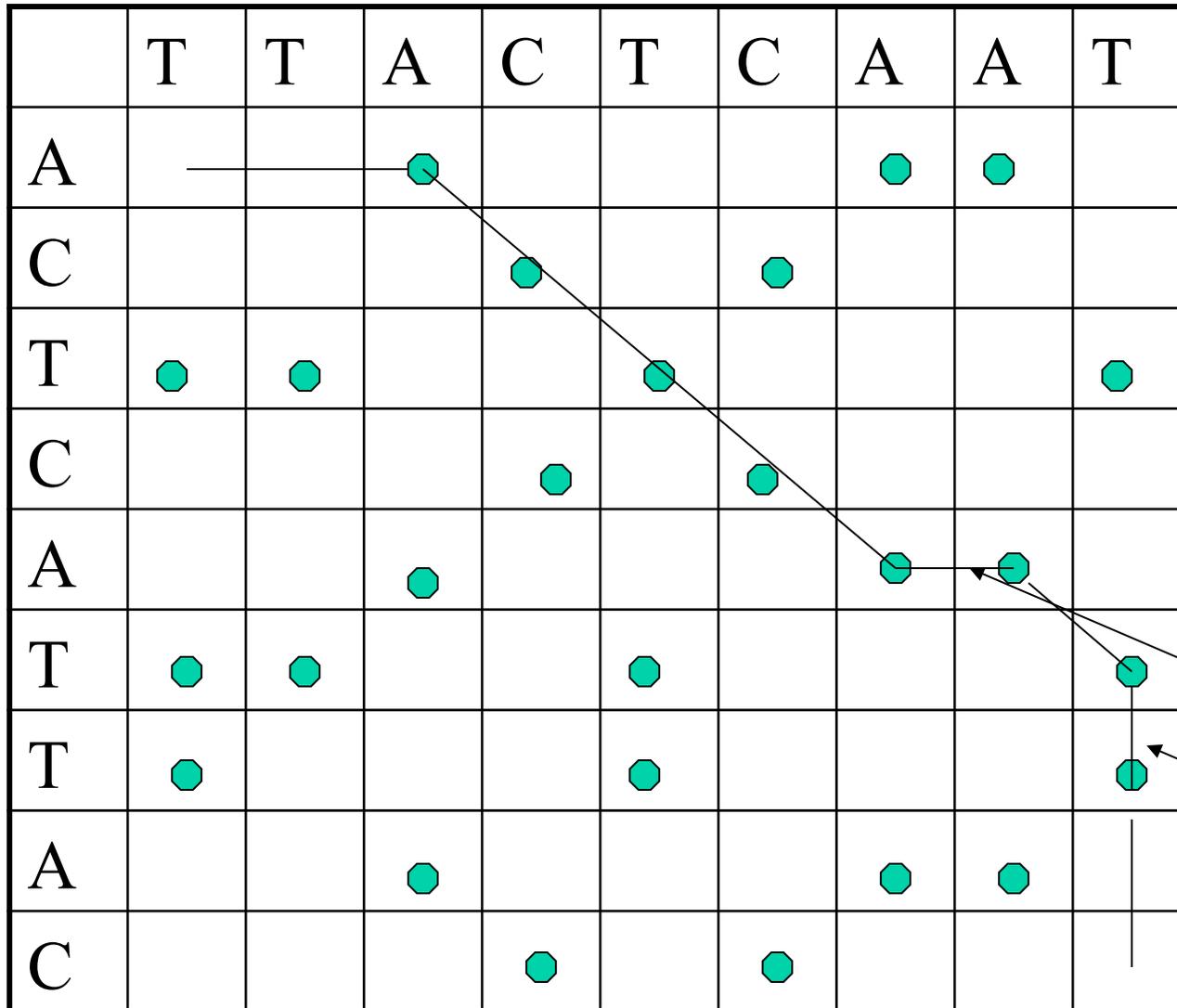
Some amino-acids are similar in their properties and size – such pairs mutate from one to other more frequently than pairs of very dissimilar amino-acids

Percent Identity and dot plots

- Given are two sequences of length n and m
- make $n \times m$ matrix D
- set $D(i,j) = 1$ if amino-acid (or nucleotide) position i in first sequence is the same as the amino-acid (nucleotide) at position j in the second sequence.
- Print graphically the matrix printing dot for 1 and space for 0

Dot plot illustration

Diagonals of dots:
regions that are
identical in both
sequences



The diagonals in the
perpendicular
direction correspond
to reverse matches
(interesting for DNA)

The alignment
corresponds to path
from upper left corner
to lower right corner
going through max. nr
of dots

Deletions

TTACTCAAT - - -
- - ACTCA-TTAC

Substitution Matrices – measure of “similarity” score of amino-acids

- Mutation data matrix: $M(i,j)$ is related to our an estimation of probability of mutating amino-acid i into j over some time period
- Percent Accepted Mutation (PAM) unit = evolutionary time corresponding to average of 1 mutation per 100 res.
- Two most popular classes of matrices:
 - PAM n : relates to mutation probabilities in evolutionary interval of n PAM units (PAM 120 is often used in practice)
 - BLOSUM x : relates to mutation probabilities observed between pairs of related proteins that diverged so above x % identity.

BLOSUM62 ~ PAM250

Gap penalties

Consider two pairs of alignments:

AT – C G and ATCG
AT T - G

They have the same score
but the right alignment is
more likely from
evolutionary perspective

AT - C - T A and ATC - - T A
AT T T T TA ATT T T TA

- First problem is corrected by introducing “gap penalty”: for each gap subtract gap penalty from the score
- Second problem is corrected by introducing additional penalty for opening a gap:

Affine gap penalty

$$w(k) = h + gk \quad ; h, g \text{ constants}$$

Interpretation: const of starting a gap: $h+g$, extending gap: $+g$

Computing best scoring alignment

- Use programming technique called **dynamic programming (Needleman-Wunsch algorithm)**
- Main idea:
 - Consider dot “dot plot” that in place of dots (0 or 1) has a value equal to a substitution matrix score for a given pair.
 - Find the best path from top left to bottom right through that matrix where the score of the path equals to the sum of scores of all cells on the path minus the gap penalty for every horizontal or vertical step.

Dynamic programming algorithm for computing for best scoring alignment

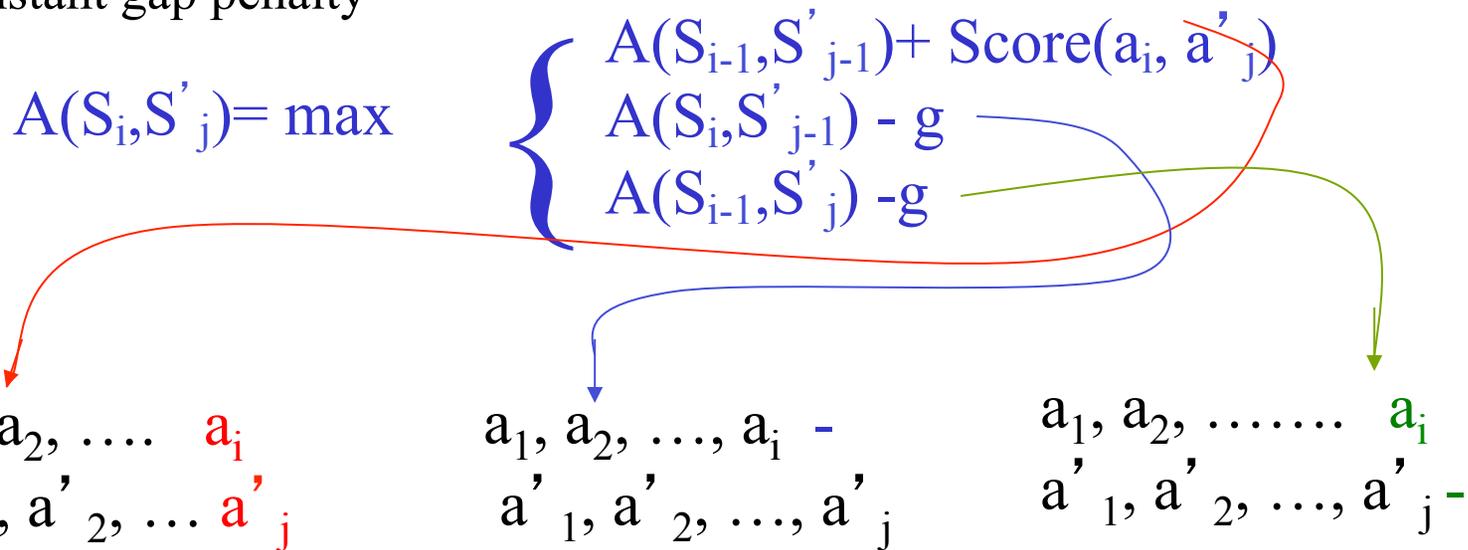
$S = a_1, a_2, \dots, a_n$, $S' = a'_1, a'_2, \dots, a'_m$ – sequences to be aligned

$S_i = a_1, a_2, \dots, a_i$; $S'_j = a'_1, a'_2, \dots, a'_j$

Score = 20 x 20 substitution scoring matrix (e.g.. PAM, BLOSUM)

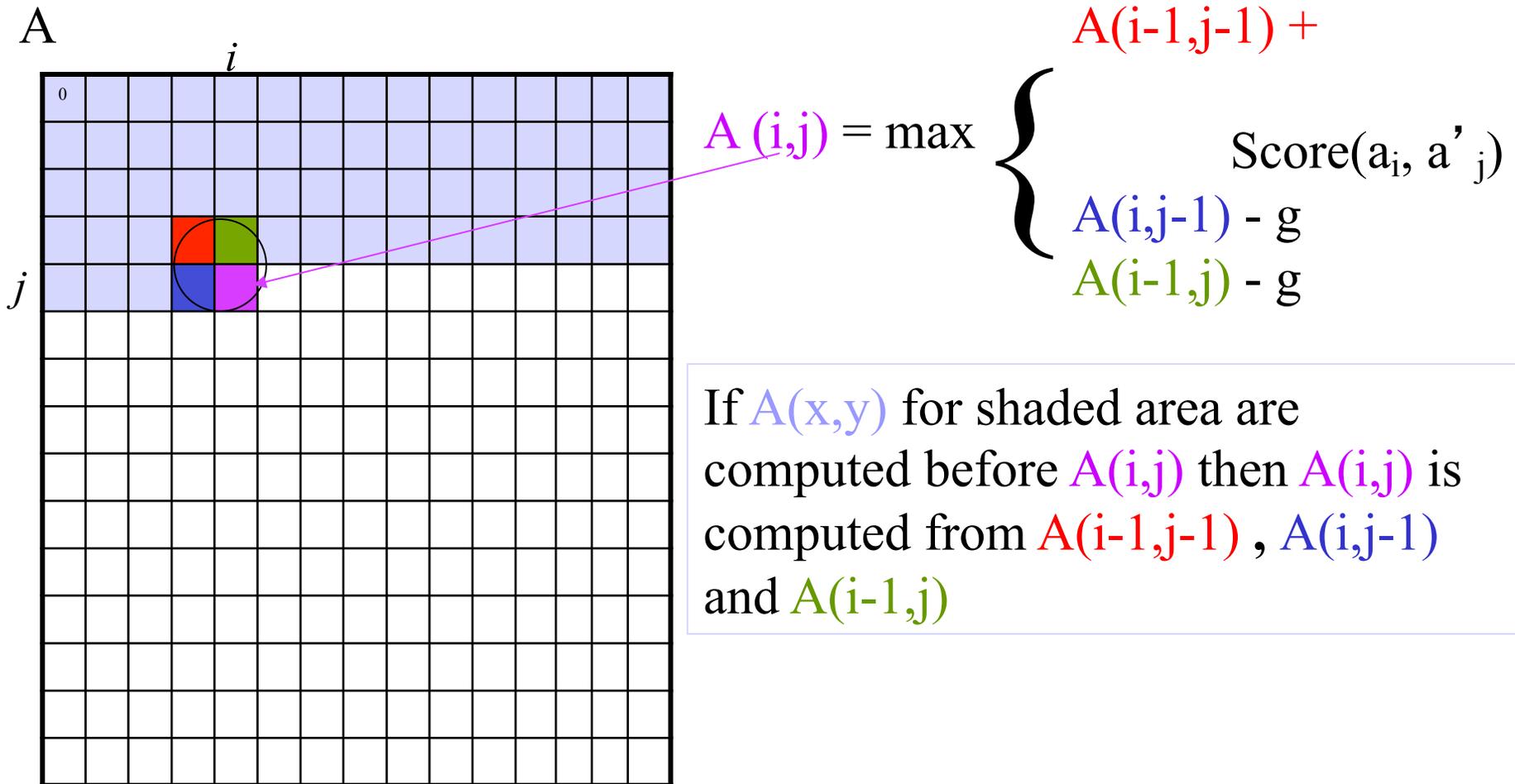
$A(S_i, S'_j)$ = score of the best scoring alignment for S_i, S'_j

g = constant gap penalty



Organizing the computation – dynamic programming table

Let $A(i,j) = A(S_i, S'_j)$. Compute all values of matrix A

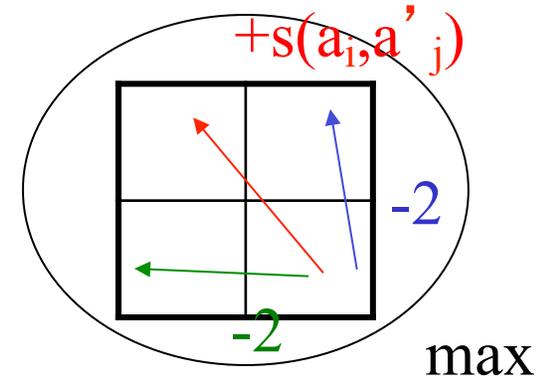


Example of DP computation

- $g = 2$; match = 2, mismatch = -1

	A	T	T	G	C	G	C	G	C	A	T
0	-2	-4	-6	-8	-10	-12	-14	-16	-18	-20	-22
A	-2	2	-3	-5							
T	-4	-3									
G	-6			6							
C	-8										
T	-10										
T	-12										
A	-14										
A	-16										
C	-18										
C	-20										
A	-22										

Initial score in column 0 and row 0 is the penalty of alignment of first i elements of our sequence with all gaps in the other sequence = $-ig$



The iterative algorithm

Initialization:

for $i \leftarrow 0$ to n

$A[i,0] = -ig$

for $j \leftarrow 0$ to m

$A[0,j] = -jg$

Main double loop:

for $i \leftarrow 1$ to n

 for $j \leftarrow 1$ to m

$A[i,j] = \max(A[i-1,j-1] + \text{Score}[s_i, s'_j],$

$A[i-1,j] - g,$

$A[i,j-1] - g)$

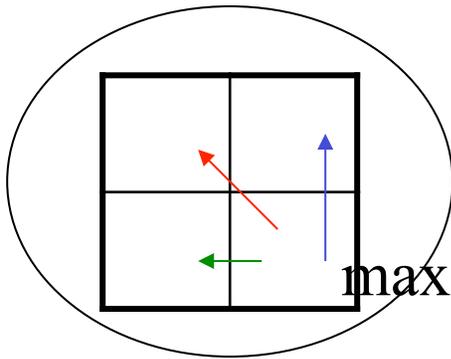
return($A[n,m]$)

From computing the score to commuting of the alignment

Desired output:

Sequence of substitutions/insertion/deletions leading to the optimal score.

ATTGCGTTATAT
AT- GCG- TATAT



Red direction = match

Blue direction = gap in horizontal sequence

Green direction = gap in vertical sequence

a_1, a_2, \dots, a_j
 a'_1, a'_2, \dots, a'_j

$a_1, a_2, \dots, a_j -$
 a'_1, a'_2, \dots, a'_j

a_1, a_2, \dots, a_j
 $a'_1, a'_2, \dots, a'_j -$

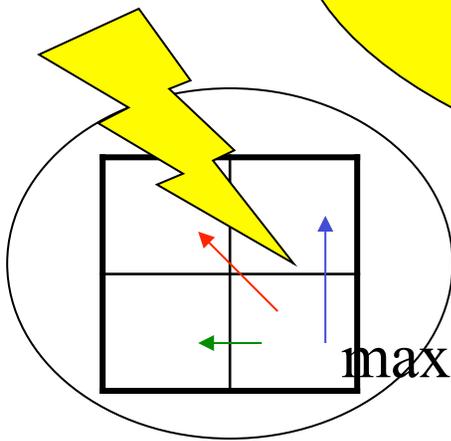
From computing the score to computing the alignment

Desired output

Sequence score.

This is not just max of the 3 number but the max of the 3 numbers after correcting for the gap penalty (horizontal and vertical arrows) or mach/mismatch (diagonal)

Optimal



Red direction = mach

Blue direction = gap in horizontal sequence

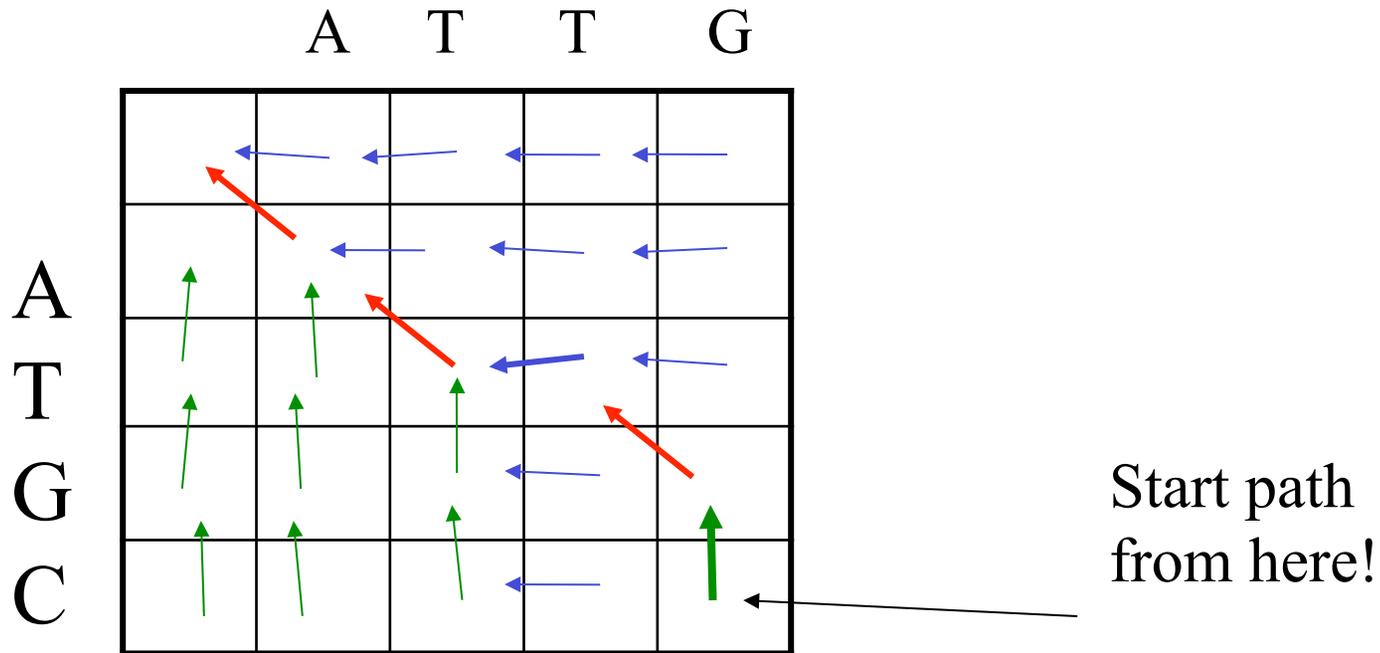
Green direction = gap in vertical sequence

a_1, a_2, \dots, a_j
 a'_1, a'_2, \dots, a'_j

$a_1, a_2, \dots, a_j -$
 a'_1, a'_2, \dots, a'_j

a_1, a_2, \dots, a_j
 $a'_1, a'_2, \dots, a'_j -$

Recovering the path



A T T G -

A T - G C

If at some position several choices lead to the same max value, the path need not be unique.

Global versus local alignment

So far we have been dealing with **global alignment**.

This is not good approach when we are interested finding similar fragments in two sequences that may be not share significant similarity outside these fragment.

	T	T	T	C	T	C	A	C	T
A							●		
C				●		●		●	
T	●	●	●		●				●
C				●		●		●	
A							●		
T	●	●	●		●				●
T	●	●	●		●				●
A							●		
C				●		●		●	

Any extension of this alignment will have significant gap penalties thus

- Global alignment may miss it
- Even when global alignment contains this local alignment the global score is likely to be low and without looking at the alignment we would have no chance of knowing that it contains two well alignment subsequences.
- **Smith Waterman algorithm** – dynamic programming algorithm for finding local alignment.

Local alignment (Smith, Waterman)

So far we have been dealing with **global alignment**.

Local alignment – alignment between substrings.

Main idea: If alignment becomes too bad – drop it.

Assumption: Scores are set in such a way that **alignment of random strings gives negative score**

$$A[i,j] = \max \left\{ \begin{array}{l} A[i-1,j-1] + \text{Score}(a_i, a_j) \\ A[i-1,j] - g \\ A[i,j-1] - g \\ 0 \end{array} \right.$$

Finding the alignment: find the highest scoring cell and trace it back

Example

2.3 Alignment algorithms

23

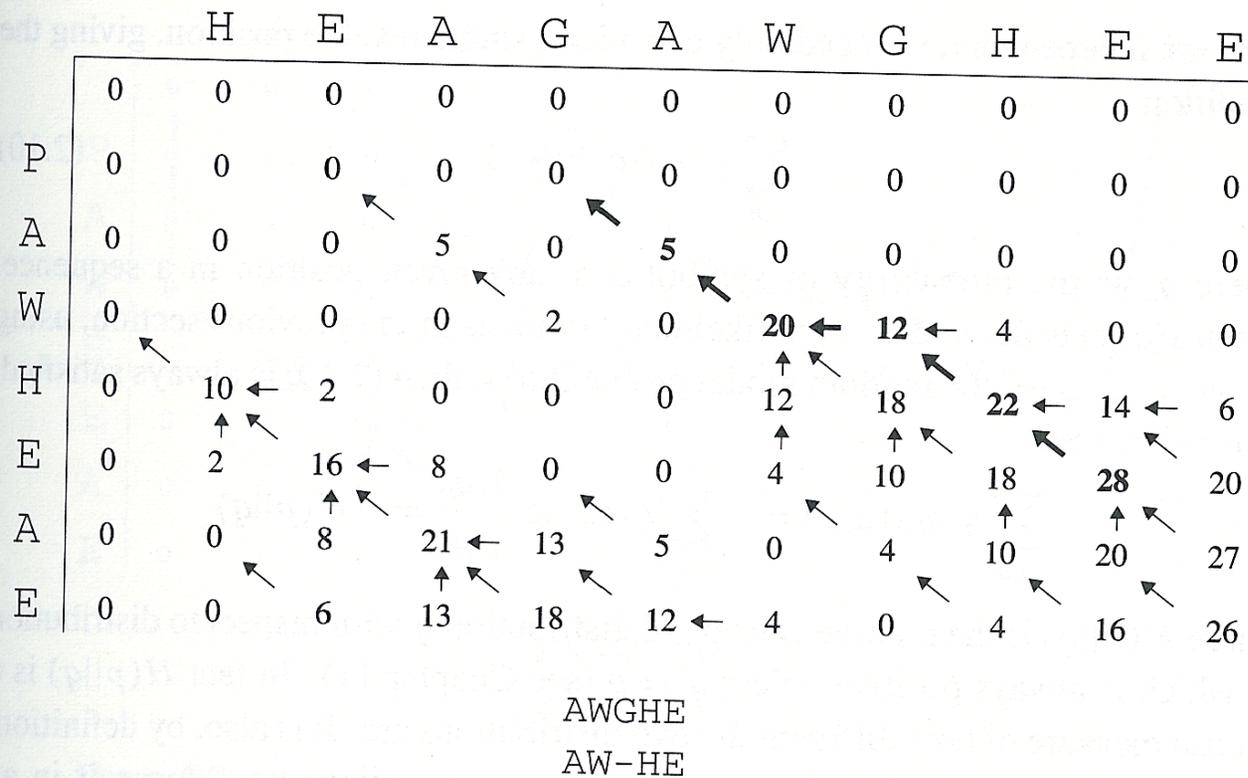


Figure 2.6 Above, the local dynamic programming matrix for the example sequences. Below, the optimal local alignment, with score 28.

Sequence Alignment Web-based exercise

- There is a number of data bases holding protein sequences organized in various ways:
 - **PIR** Protein Identification Resource
 - **SWISS-PROT** and **TrEMBL**
 - **Entrez**
- In this exercise we will download two sequences: one using Entrez and one using PIR and perform multiple alignment using [sequence alignment](#) tool available on USC website
- The two sequences are: 1eca, 1bob, (ids from Protein Data Base (PDB), other data base have their own id's but major databases cross-reference)
- Observe that pairs aligned identical residues are identified using ":". Pairs identified with "." are "similar" amino-acids

Implications sequence similarity

- Experimental observations:
 - If two protein sequences of length > 100 have more than 25% identity then they are homologous (evolutionary related).
 - For shorter sequences 25% identity may happen by chance thus a larger sequence identity (depending on length is required)
- Twilight zone – sequence similarity in 15%-25% range where it is hard to decide one way or the other.
- Modern usually provide a statistical assessment of significance of the alignment score (will be discussed later).

Data Base Searches

- **Goal:** Given a protein sequence and a protein data base find in the data base sequences that are homologous to a given sequence.
- **Why:** Homologous sequences often have same or related function thus if we fish out a sequence with known function this will provide a hint towards the function of the query protein.
- **Relevant issues:**
 - Speed (!)
 - Ability of assessment of relevance of the results returned by search
 - Specificity and Sensitivity of the search

BLAST

- Basic Local Alignment Search Tool – a family of most popular sequence search program including
- **Main idea:** Homologous sequences are likely to contain a short high scoring similarity region **a hit**.
 - Find two non-overlapping hits of length w (usually set to 3) of score at least T and distance at most d one from another
 - Invoke ungapped (cheep) extension.
 - If the HSP generated has score above certain threshold then start extension that allows gaps (expensive extension).
 - Report resulting alignment if it has sufficiently large statistical significance (defined using e-value – see next slide)
- **For BLAST tutorial visit** <http://www.ncbi.nlm.nih.gov/BLAST/>

Significance of results

P-value = given the length of query sequence and the size of the data base gives probability of finding an alignment with a certain score by chance.

E-value = expected number of “by chance” hits of given or higher score (also depends on data base size)

Normalized score = alignment score normalized so that the alignments obtained using different scoring functions can be directly compared.

Blast Web-exercise

- Open BLAST web server

BLAST: <http://www.ncbi.nlm.nih.gov/BLAST/>

- Get a protein sequence to serve as a query (1bob) from [Entrez](#)

Other Data Base Searching tools

- **FASTA** - basic principles are similar to BLAST but there are significant differences
- **PSI-Blast** – a tool for finding more distant homologues - will be discussed in a later classes.

Sensitivity /Specificity of a data base search

	Related	Unrelated
Retrieved by the search	TP True Positive	FP False Positive
Not retrieved by the search	FN False Negative	TN True Negative

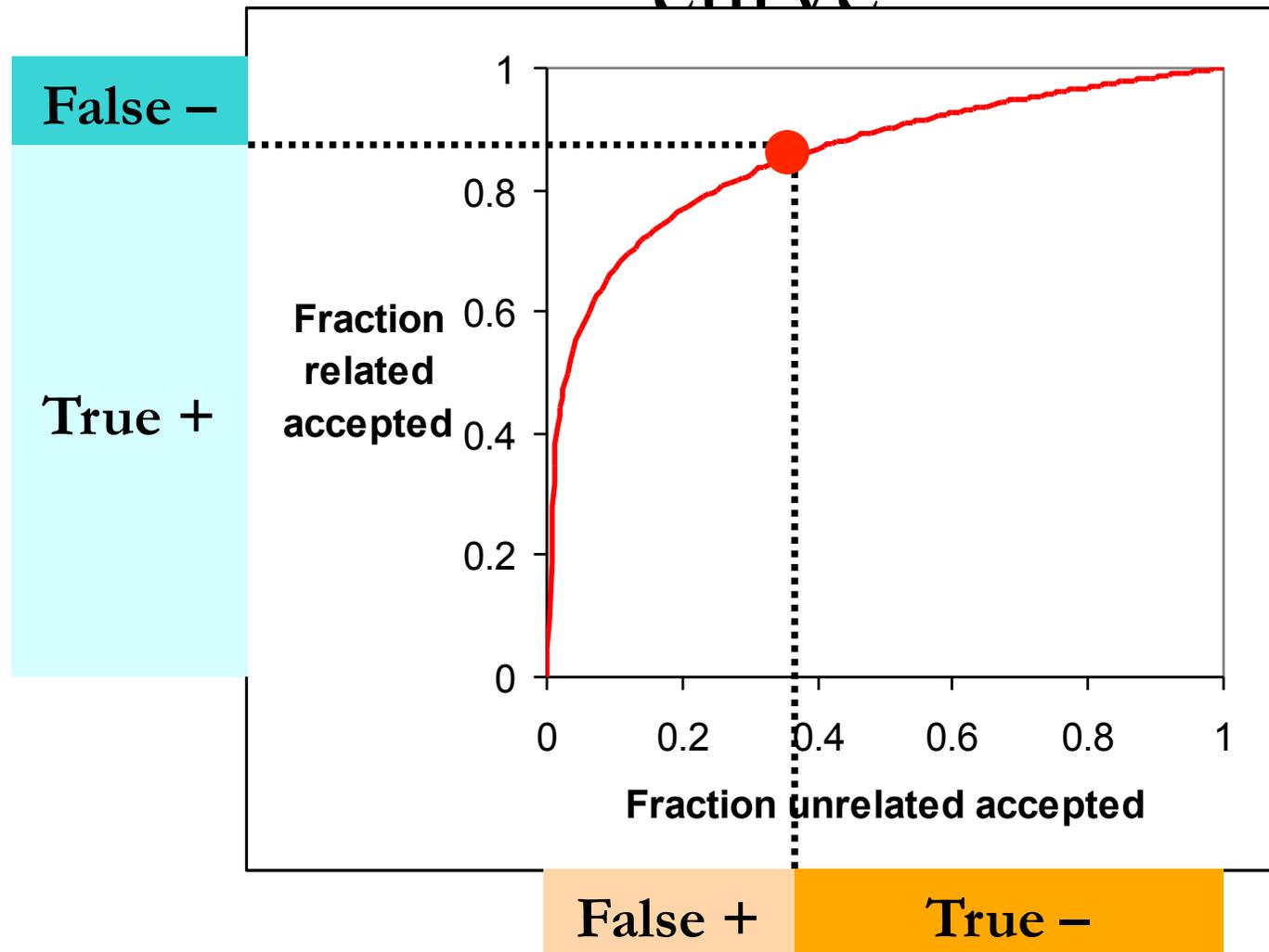
Sensitivity: $TP / (TP + FN)$

Specificity: $TN / (TN + FP)$

Positive Predictive Value:
 $TP / (TP + FP)$

Typically, increasing TP leads to increasing FP and decreasing FN thus as we change parameters to increase Sensitivity Specificity goes down. Need to take it into account in comparing various methods.

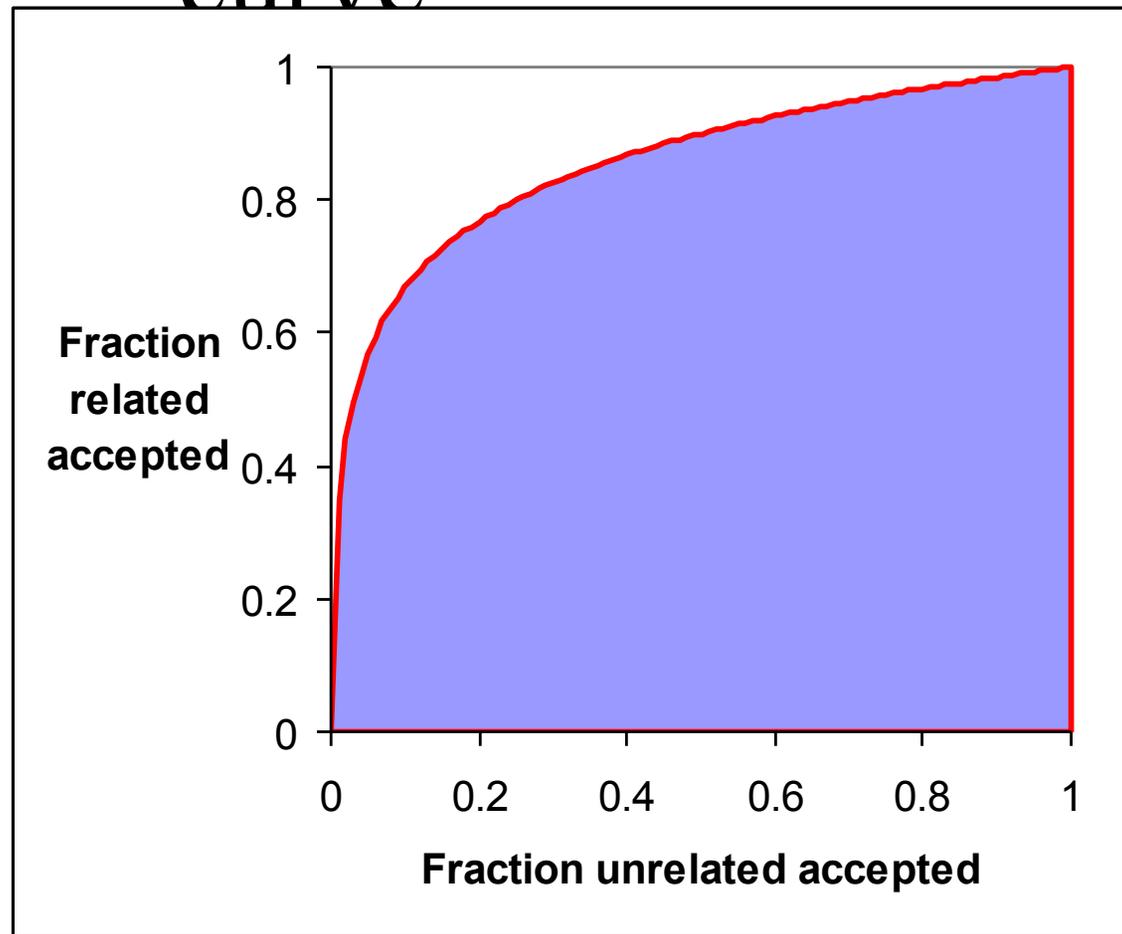
Receiver Operating Characteristic curve



This slide is by Stephen Altschul from talk: www.dimacs.rutgers.edu/workshops/proteindomains/dimacstalk1.ppt

ROC score: area under the *ROC* curve

Sensitivity of the search = $TP / (TP + FN)$
Specificity of the search = $TN / (FP + TN)$
So ROC plots are plots of Sensitivity vs. (1-Specificity)



Multiple alignment

S^1, S^2, \dots, S^k a set of sequences over the same alphabet

As for pair-wise alignment we would like to find alignment that maximizes some scoring function:

M Q P I L L L

M L R - L - L

M P V I L I L

How to score such multiple alignment?

Sum of pairs (SP) score

Example consider all pairs of letters in each column and add the scores:

$$\text{SP-score} \begin{pmatrix} A \\ V \\ V \\ - \end{pmatrix} = \text{score}(A,V) + \text{score}(V,V) + \text{score}(V,-) + \text{score}(A,-) + \text{score}(A,V)$$

k sequences gives $k(k-1)/2$ addends

Remark: $\text{Score}(-,-) = 0$

Entropy Score

$$-\sum (c_j/C) \log (c_j/C)$$

Entropy based score (minimum)

$$-\sum_j (c_j/C) \log (c_j/C)$$

c_j - number of occurrence of amino-acid j in the column

C – number of symbols in the column

A	A	A	A	A
A	A	A	A	I
A	A	A	A	K
A	A	A	I	L
A	A	I	I	S
A	I	I	I	W

0 -0.68 -0.9 -1 -2.58

Multiple sequence alignment algorithms

- MSA
 - Authors of the program and consecutive improvements: Carrillo , Lipman, Altschul, Shaffer, Gupta, Kecioglu,...
 - extension of dynamic programming approach to more sequences
 - accurate but expensive
- CLUSTAL W
 - Authors: Higgins & Sharp
 - Produces an evolutionary tree and progressively aligns partial alignments in the order guided by the tree – from leaves towards the root.
 - Fast but not perfect
- T-COFFE
 - Authors Noterdame, Higgins, Heringa, JMB 2000, 302 205-217
 - A hybrid approach, more accurate than CLASTAL W
 - Tries to negotiate best pair wise alignment based on several alignment and transitive closure and then uses progressive tree-based alignment
- MUSCLE newest algorithm, almost as accurate as T-COFFEE but fast.