CS 466
Introduction to Bioinformatics
Lecture 7

Mohammed El-Kebir

September 24, 2018

Course Announcements

Instructor:

- Mohammed El-Kebir (melkebir)
- Office hours: Mondays, 3:15-4:15pm

TA:

- Anusri Pampari (pampari2)
- Office hours: Thursdays, 11:00-11:59am in SC 4105

Midterm on Oct. 10, 7-9pm, 1310 DCL

Outline

- Recap: RNA Secondary Structure Prediction
- Protein Contact Map Overlap

Reading:

- Lecture notes
- Caprara, A., Carr, R., Istrail, S., Lancia, G., & Walenz, B. (2004). 1001 Optimal PDB Structure Alignments: Integer Programming Methods for Finding the Maximum Contact Map Overlap. *Journal of Computational Biology*, *11*(1), 27–52. http://doi.org/10.1089/1066527047734168 76

Nussinov Algorithm

RNA can fold into structures due to nucleotide complementarity:
A <--> U and G <--> C

Secondary structure is determined by a set of non-overlapping complimentary base pairs

SIAM J. APPL, MATH.
Vol. 35, No. 1, July 1978

0036–1399/78/3501–0006 $01.00/0

ALGORITHMS FOR LOOP MATCHINGS*

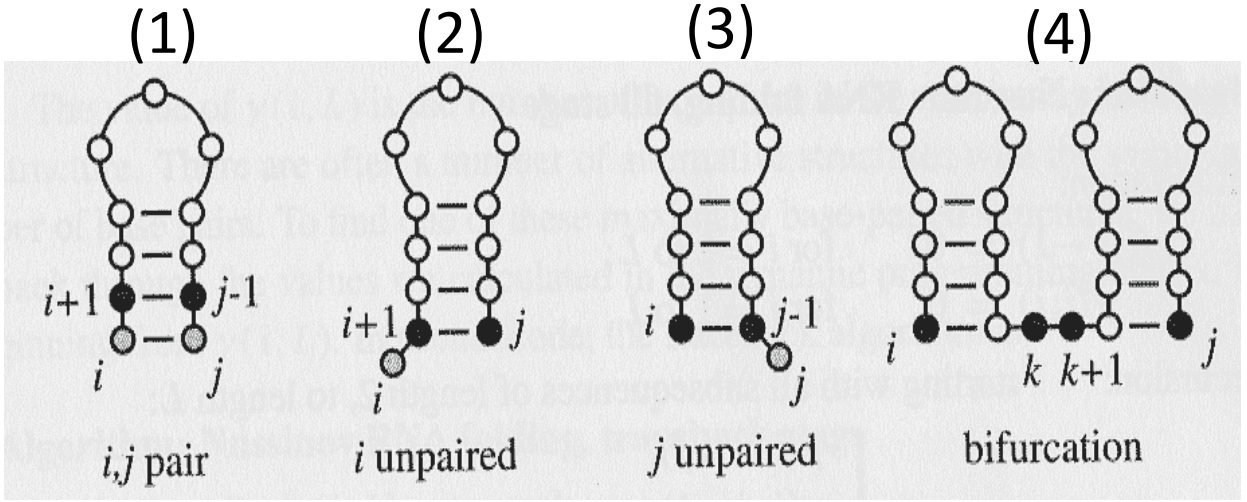
RUTH NUSSINOV,† GEORGE PIECZENIK,‡ JERROLD R. GRIGGS¶
AND DANIEL J. KLEITMAN§

Problem: Given RNA sequence $\mathbf{v} \in \{\mathrm{A}, \mathrm{U}, \mathrm{C}, \mathrm{G}\}^n$, find a *pseudoknot-free secondary structure* with the maximum number of complementary base pairings

Nussinov Algorithm – Dynamic Programming

Problem: Given RNA sequence $\mathbf{v} \in \{\mathrm{A}, \mathrm{U}, \mathrm{C}, \mathrm{G}\}^n$, find a *pseudoknot-free secondary structure* with the maximum number of complementary base pairings

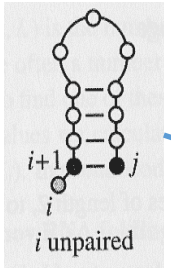
Let $s[i,j]$ denote the maximum number of pseudoknot-free complementary base pairings in subsequence $v_i, \ldots, v_j$

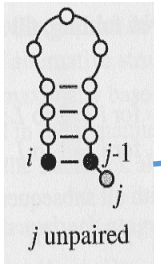


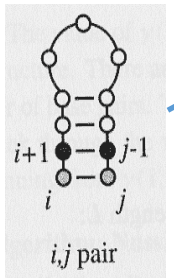
$$s[i,j] = \max \begin{cases} 0, & \text{if } i \geq j, \\ s[i+1, j-1] + 1, & \text{if } i < j \text{ and } (v_i, v_j) \in \Gamma, \quad (1) \\ s[i+1, j-1], & \text{if } i < j \text{ and } (v_i, v_j) \notin \Gamma, \quad (1^*) \\ s[i+1, j], & \text{if } i < j, \quad (2) \\ s[i, j-1], & \text{if } i < j, \quad (3) \\ \max_{i<k<j}\{s[i,k] + s[k+1, j]\}, & \text{if } i < j, \quad (4) \end{cases}$$

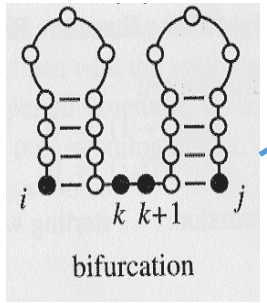
Question: Which case is redundant?

Nussinov Algorithm – Traceback Step



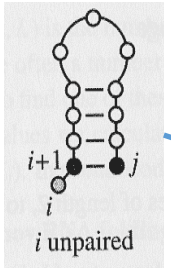


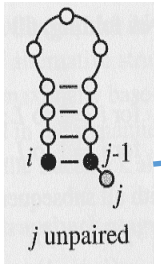


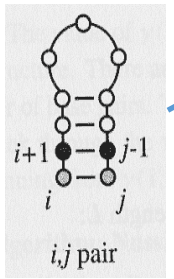


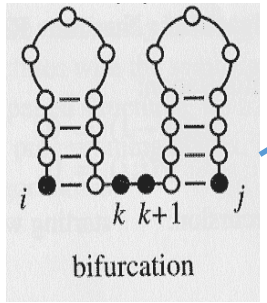
Push (1, *n*) onto stack
Repeat until stack is empty:
pop (*i*,*j*)
if *i* ≥ *j* continue
else if s[*i*+1,*j*] = s[*i*,*j*]
push (*i*+1,*j*)
else if s[*i*,*j*-1] = S[*i*,*j*]
push (i,j-1)
else if s[i+1,j-1] + 1 = s[i,j]
record (i,j) base pair
push (i+1,j-1)
else for k = i+1 to j-1
if s[i,k]+s[k+1,j] = s[i,j]
push (k+1,j)
push (i,k)
break (for loop)

Nussinov Algorithm – Traceback Step









Push (1, *n*) onto stack
Repeat until stack is empty:
pop (*i*,*j*)
if *i* ≥ *j* continue
else if s[*i*+1,*j*] = s[*i*,*j*]
push (*i*+1,*j*)
else if s[*i*,*j*-1] = S[*i*,*j*]
push (i,j-1)
else if s[i+1,j-1] + 1 = s[i,j]
record (i,j) base pair
push (i+1,j-1)
else for k = i+1 to j-1
if s[i,k]+s[k+1,j] = s[i,j]
push (k+1,j)
push (i,k)
break (for loop)

BackTrack(*i*, *j*)
if *i* < *j*
if s[*i*+1, *j*] = s[*i*, *j*]
BackTrack(*i*+1, *j*)
else if s[*i*, *j*-1] = S[*i*, *j*]
BackTrack(*i*, *j*-1)
else if s[*i*+1,*j*-1] + 1 = s[*i*, *j*]
Output (*i*, *j*)
BackTrack(*i*+1, *j*-1)
else for *k* = *i*+1 **to** *j*-1
if s[*i*, *k*]+s[*k*+1, *j*] = s[*i*, *j*]
BackTrack(*k*+1, *j*)
BackTrack(*i*, *k*)
break (for loop)

Outline

- Recap: RNA Secondary Structure Prediction
- Protein Contact Map Overlap

Reading:

- Lecture notes
- Caprara, A., Carr, R., Istrail, S., Lancia, G., & Walenz, B. (2004). 1001 Optimal PDB Structure Alignments: Integer Programming Methods for Finding the Maximum Contact Map Overlap. *Journal of Computational Biology*, *11*(1), 27–52. http://doi.org/10.1089/10665270477341687 6

Central Dogma of Molecular Biology

Three fundamental molecules:

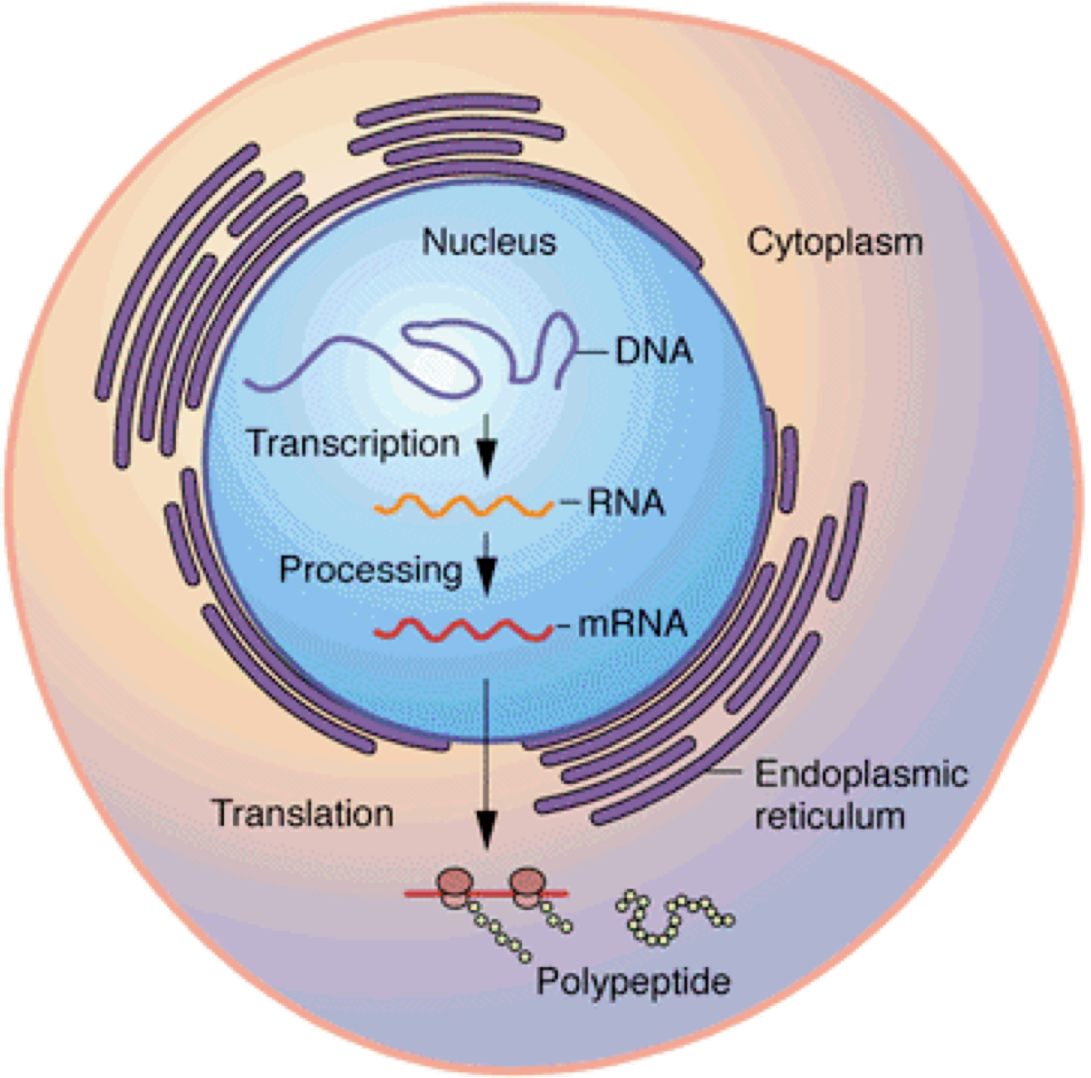
1. **DNA**
   Information storage.

2. **RNA**
   Old view: Mostly a "messenger".
   New view: Performs many important functions, through **3-D structure**!

3. **Protein**
   Perform most cellular functions (biochemistry, signaling, control, etc.)

DNA → RNA → Protein

First proposed by Francis Crick in 1956.



A GUIDE TO THE TWENTY COMMON AMINO ACIDS

AMINO ACIDS ARE THE BUILDING BLOCKS OF PROTEINS IN LIVING ORGANISMS. THERE ARE OVER 500 AMINO ACIDS FOUND IN NATURE - HOWEVER, THE HUMAN GENETIC CODE ONLY DIRECTLY ENCODES 20. 'ESSENTIAL' AMINO ACIDS MUST BE OBTAINED FROM THE DIET, WHILST NON-ESSENTIAL AMINO ACIDS CAN BE SYNTHESISED IN THE BODY.

Chart Key: ALIPHATIC · AROMATIC · ACIDIC · BASIC · HYDROXYLIC · SULFUR-CONTAINING · AMIDIC · NON-ESSENTIAL · ESSENTIAL

Chemical Structure

single letter code

NAME (A)
three letter code
DNA codons

ALANINE (A)
Ala
GCT, GCC, GCA, GCG

GLYCINE (G)
Gly
GGT, GGC, GGA, GGG

ISOLEUCINE (I)
Ile
ATT, ATC, ATA

LEUCINE (L)
Leu
CTT, CTC, CTA, CTG, TTA, TTG

PROLINE (P)
Pro
CCT, CCC, CCA, CCG

VALINE (V)
Val
GTT, GTC, GTA, GTG

PHENYLALANINE (F)
Phe
TTT, TTC

TRYPTOPHAN (W)
Trp
TGG

TYROSINE (Y)
Tyr
TAT, TAC

ASPARTIC ACID (D)
Asp
GAT, GAC

GLUTAMIC ACID (E)
Glu
GAA, GAG

ARGININE (R)
Arg
CGT, CGC, CGA, CGG, AGA, AGG

HISTIDINE (H)
His
CAT, CAC

LYSINE (K)
Lys
AAA, AAG

SERINE (S)
Ser
TCT, TCC, TCA, TCG, AGT, AGC

THREONINE (T)
Thr
ACT, ACC, ACA, ACG

CYSTEINE (C)
Cys
TGT, TGC

METHIONINE (M)
Met
ATG

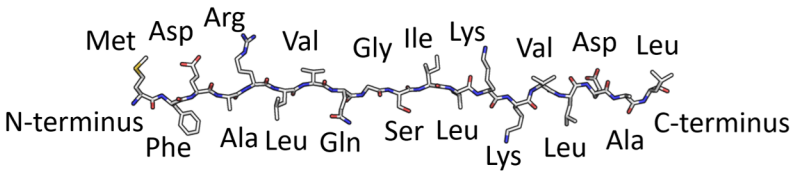
ASPARAGINE (N)
Asn
AAT, AAC

GLUTAMINE (Q)
Gln
CAA, CAG

Note: This chart only shows those amino acids for which the human genetic code directly codes for. Selenocysteine is often referred to as the 21st amino acid, but is encoded in a special manner. In some cases, distinguishing between asparagine/aspartic acid and glutamine/glutamic acid is difficult. In these cases, the codes asx (B) and glx (Z) are respectively used.

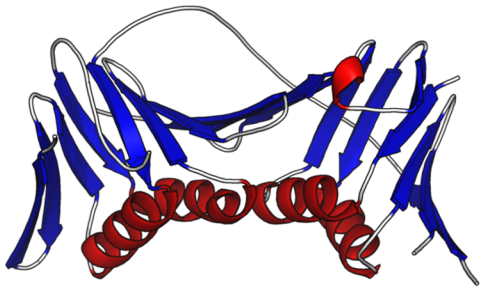
Protein Structure Prediction

Primary

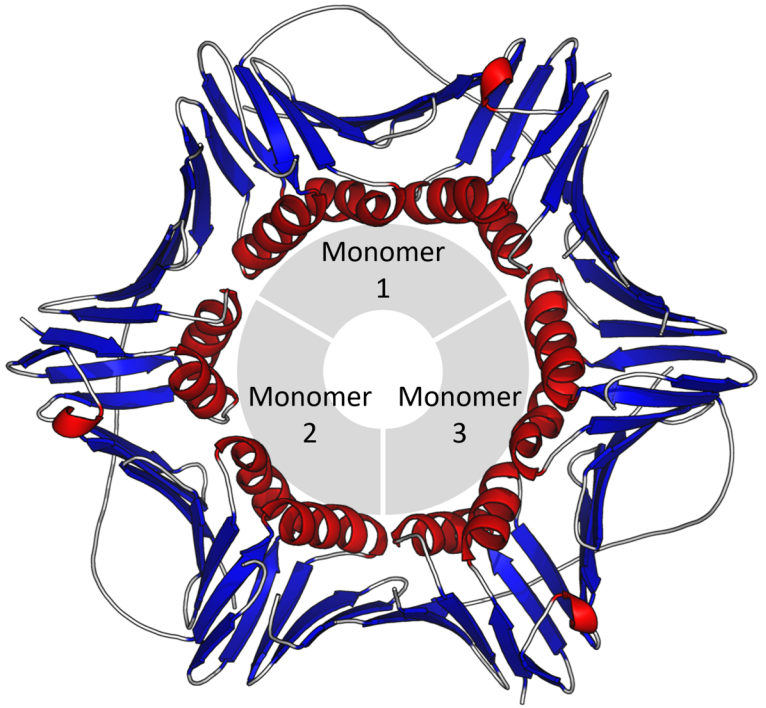


Secondary

Tertiary



Quaternary

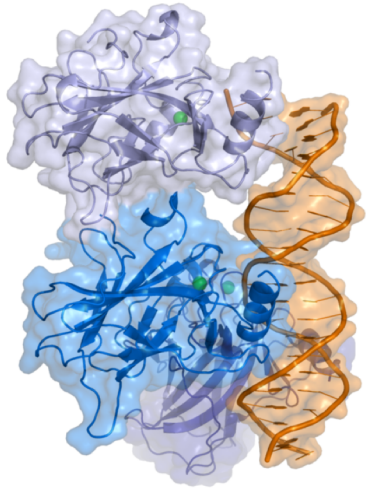


Example

- http://pdb101.rcsb.org/motm/218

On Sequence, Structure and Function: p53

```
        10         20         30         40         50
MEEPQSDPSV EPPLSQETFS DLWKLLPENN VLSPLPSQAM DDLMLSPDDI
        60         70         80         90        100
EQWFTEDPGP DEAPRMPEAA PPVAPAPAAP TPAAPAPAPS WPLSSSVPSQ
       110        120        130        140        150
KTYQGSYGFR LGFLHSGTAK SVTCTYSPAL NKMFCQLAKT CPVQLWVDST
       160        170        180        190        200
PPPGTRVRAM AIYKQSQHMT EVVRRCPHHE RCSDSDGLAP PQHLIRVEGN
       210        220        230        240        250
LRVEYLDDRN TFRHSVVVPY EPPEVGSDCT TIHYNYMCNS SCMGGMNRRP
       260        270        280        290        300
ILTIITLEDS SGNLLGRNSF EVRVCACPGR DRRTEEENLR KKGEPHHELP
       310        320        330        340        350
PGSTKRALPN NTSSSPQPKK KPLDGEYFTL QIRGRERFEM FRELNEALEL
       360        370        380        390
KDAQAGKEPG GSRAHSSHLK SKKGQSTSRH KKLMFKTEGP DSD
```

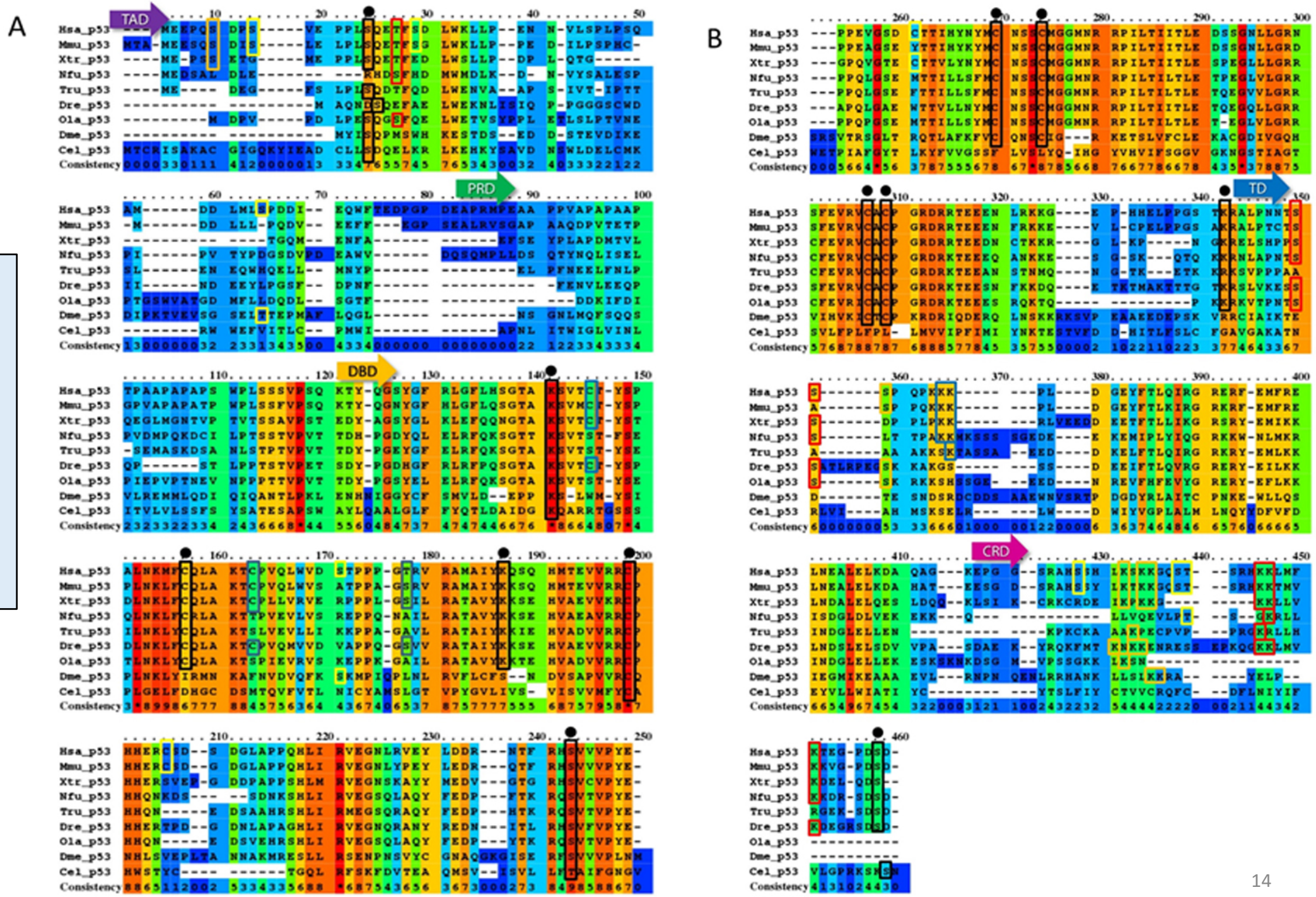


- It can activate DNA repair proteins when DNA has sustained damage
- It can arrest growth by holding the cell cycle at the G1/S regulation point on DNA damage
- It can initiate apoptosis (i.e., programmed cell death) if DNA damage proves to be irreparable.
- It is essential for the senescence response to short telomeres.

Sequence → Structure → Function

What is functionally important is conserved throughout evolution

What is functionally important is conserved throughout evolution



How to Compare Two Protein Sequences?

TP53 (Human)

```
  1 meepqsdpsv epplsqetfs dlwkllpenn vlsplpsqam ddlmlspddi eqwftedpgp
 61 deaprmpeaa ppvapapaap tpaapapaps wplsssvpsq ktyqgsygfr lgflhsgtak
121 svtctyspal nkmfcqlakt cpvqlwvdst pppgtrvram aiykqsqhmt evvrrcphhe
181 rcsdsdglap pqhlirvegn lrveylddrn tfrhsvvvpy eppevgsdct tihynymcns
241 scmggmnrrp iltiitleds sgnllgrnsf evrvcacpgr drrteeenlr kkgephhelp
301 pgstkralpn ntssspqpkk kpldgeyftl qirgrerfem frelnealel kdaqagkepg
361 gsrahsshlk skkgqstsrh kklmfktegp dsd
```

p53 (Mouse)

```
  1 mtameesqsd islelplsqe tfsglwkllp pedilpsphc mddlllpqdv eeffegpsea
 61 lrvsgapaaq dpvtetpgpv apapatpwpl ssfvpsqkty qgnygfhlgf lqsgtaksvm
121 ctyspplnkl fcqlaktcpv qlwvsatppa gsrvramaiy kksqhmtevv rrcphhercs
181 dgdglappqh rirvegnlyp eyledrqtfr hsvvvpyepp eagseyttih ykymcnsscm
241 ggmnrrpilt iitledssgn llgrdsfevr vcacpgrdrr teeenfrkke vlcpelppgs
301 akralptcts asppqkkkpl dgeyftlkir grkrfemfre lnealelkda hateesgdsr
361 ahssylktkk gqstsrhkkt mvkkvgpdsd
```

How to Compare Two Protein Sequences?

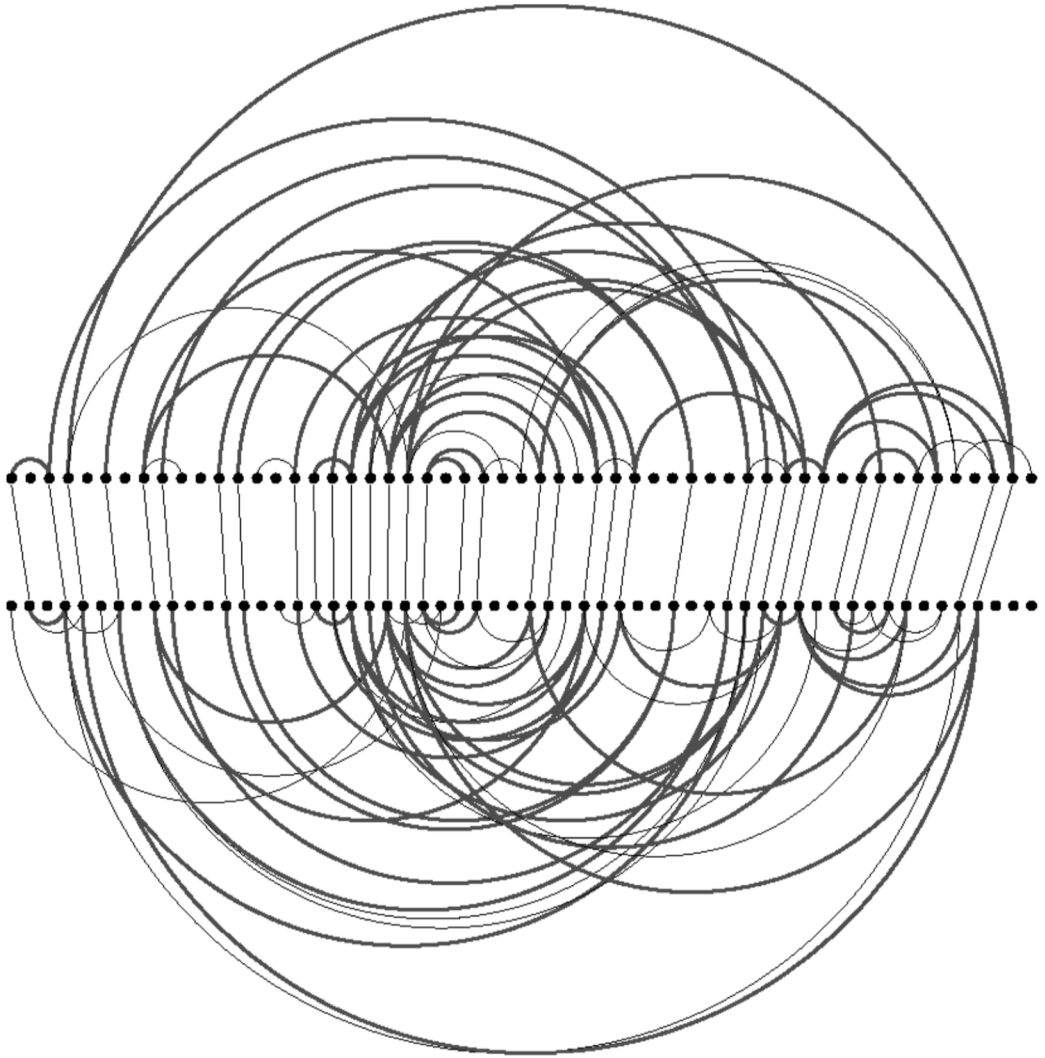
Global Alignment problem: Given strings $\mathbf{v} \in \Sigma^m$ and $\mathbf{w} \in \Sigma^n$ and scoring function δ, find alignment of $\mathbf{v}$ and $\mathbf{w}$ with maximum score

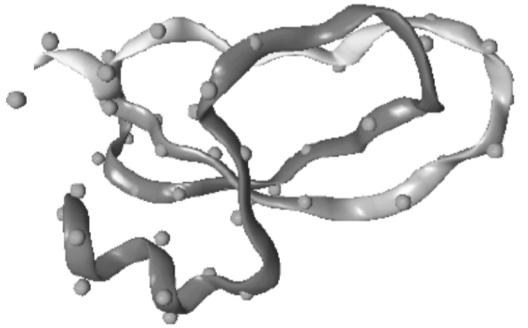
[Needleman-Wunsch algorithm]

Local Alignment problem: Given strings $\mathbf{v} \in \Sigma^m$ and $\mathbf{w} \in \Sigma^n$ and scoring function δ, find a substring of $\mathbf{v}$ and a substring of $\mathbf{w}$ whose alignment has maximum global alignment score s^* among *all* global alignments of *all* substrings of $\mathbf{v}$ and $\mathbf{w}$

[Smith-Waterman algorithm]

How to Compare Two Protein Structures?





1knt-4.0.cm: 55 residues with 43 contacts.

31 shared contacts.

1bpi-4.0.cm: 58 residues with 53 contacts.

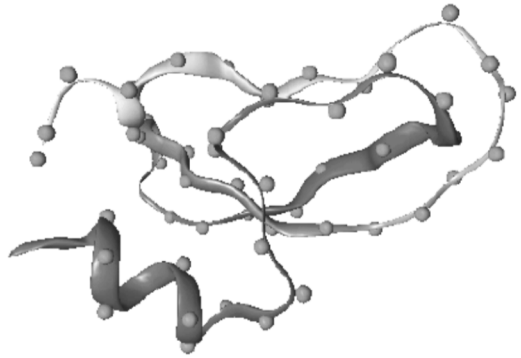


FIG. 1. An optimal alignment of two 4Å threshold contact maps of proteins `1bpi` and `1knt`.