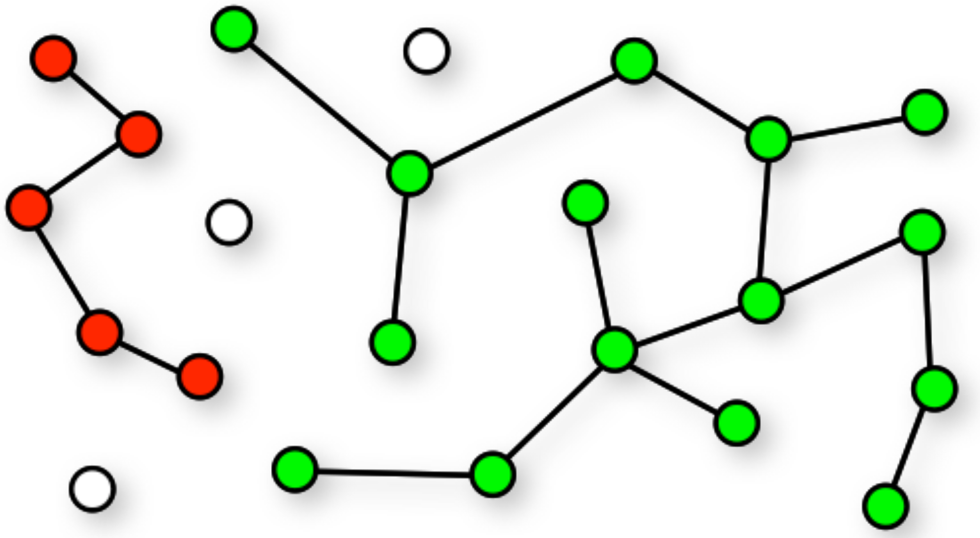
V2 Protein Networks and Complexes

Connected graph <=> there is a path between all pairs of nodes

In large (random) networks: complete $\{V\}$ is often not connected
→ identify connected subsets $\{V_i\}$ with $\{V\} = \cup \{V_i\}$
→ **connected components** (CC)



#CC = 5
N_{max} = 15
N_{min} = 1

Basic Types: (1) Random Network

Given: *N* vertices connected by *L* edges

where the edges are **randomly distributed** between the vertices

Maximal number of links between *N* vertices:

$$L_{max} = \frac{N(N-1)}{2}$$

=> **probability** *p* for an edge between two randomly selected nodes:

$$p = \frac{L}{L_{max}} = \frac{2L}{N(N-1)}$$

=> **average degree** λ

$$\lambda = \frac{2L}{N} = p(N-1)$$

path lengths in a random network grow with `ln`(*N*) => **"small world"**

Random Network: *P(k)*

Network with *N* vertices, *L* edges
=> probability for a random link:

$$p = \frac{2L}{N(N-1)}$$

Probability that random node has links to *k* other particular nodes:

$$W_k = p^k \, (1-p)^{N-k-1}$$

Probability that random node has links to any *k* other nodes:

$$P(k) = \binom{N-1}{k} W_k = \frac{(N-1)!}{(N-k-1)!\,k!} W_k$$

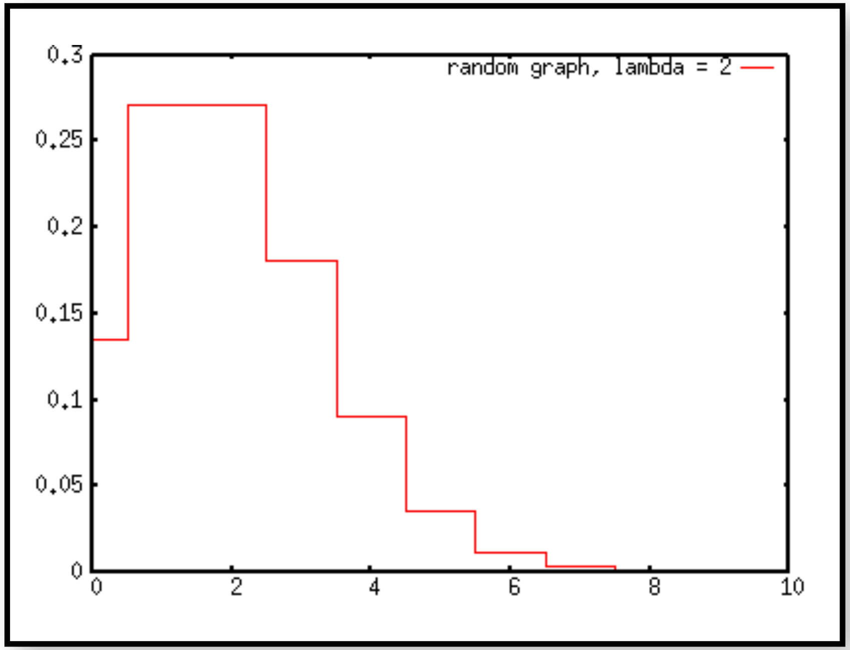
Limit of large graph: *N* → oo, *p* = λ / *N*

$$\begin{aligned}
\lim_{N\to\infty} P(k) &= \lim_{N\to\infty} \frac{N!}{(N-k)!\,k!} \, p^k \, (1-p)^{N-k} \\
&\underset{p\approx\frac{\lambda}{N}}{=} \lim_{N\to\infty} \left(\frac{N(N-1)\dots(N-k+1)}{N^k} \right) \frac{\lambda^k}{k!} \left(1-\frac{\lambda}{N}\right)^N \left(1-\frac{\lambda}{N}\right)^{-k} \\
&= \qquad 1 \qquad \frac{\lambda^k}{k!} \qquad \mathrm{e}^{-\lambda} \qquad 1 \\
&= \frac{\lambda^k}{k!} \, \mathrm{e}^{-\lambda}
\end{aligned}$$

Random Network: *P(k)*

Many independently placed edges => **Poisson statistics**

$$P(k) = \frac{\lambda^k}{k!}\, e^{-\lambda}$$



k	*P(k I λ = 2)*
0	0.14
1	0.27
2	0.27
3	0.18
4	0.090
5	0.036
6	0.012
7	0.0034
8	0.00086
9	0.00019
10	3.82e-05

=> Small probability for *k* >> λ

C(k) for a Random Network

Clustering coefficient when *m* edges exist between *k* neighbors

$$C(k,m) = \frac{2m}{k(k-1)}$$

Probability to have exactly *m* edges between the *k* neighbors

$$W(m) = \binom{k}{m} p^m \, (1-p)^{\frac{k(k-1)}{2}-m}$$

# possibilities of picking the *m* start nodes for the *m* edges from the *k* nodes.

Average *C(k)* for degree *k*:

$$C(k) = \frac{\sum_{m=0}^{\frac{k(k-1)}{2}} W(m)\, C(k,m)}{\sum_{m=0}^{\frac{k(k-1)}{2}} W(m)} = \ldots = p$$

→ *C(k)* is independent of *k*

<=> same local connectivity throughout the network

Basic Types: (2) Scale-Free

Growing network a la Barabasi and Albert (1999):

- start from a small "nucleus“ of m_0 connected nodes
- in each iteration step, add new node with *n* links
- connect new links to existing nodes with probability p_i proportional to degree k_i of each existing node (“preferential attachment”);

=> "the rich get richer"

$$p_i = \left(\frac{k_i}{\sum k_i} \right)^{\beta}$$

in BA-model β = 1

Properties:

- this leads to a power-law degree distribution:

$$P(k) \propto k^{-\gamma}$$

with γ = 3 for the BA model

- self-similar structure with highly connected hubs (no intrinsic length scale)

=> average path length grows with `ln` (N) / `ln`(`ln`(*N*))

=> this grows much slower than for random graphs

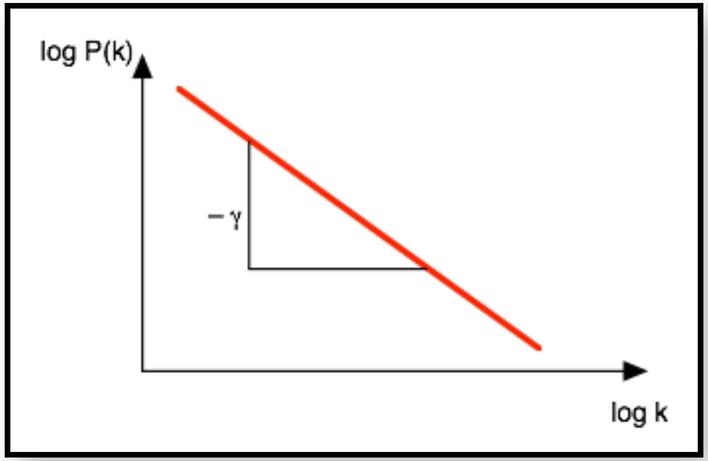
=> **“very small world”**

The Power-Law Signature

Power law $P(k) \propto k^{-\gamma}$

Take log on both sides:

$$\log(P(k)) = -\gamma \log(k)$$



Plot log(*P*) vs. log(*k*) => straight line

Note: for fitting γ against experimental data it is often better to use the integrated *P(k)*
=> integral smoothens the data

$$\int_{k_0}^{k} P(k)dk = \left[-\frac{k^{-(\gamma-1)}}{\gamma}\right]_{k_0}^{k}$$

Scale-Free: Examples

The World-Wide-Web:

=> growth via links to portal sites

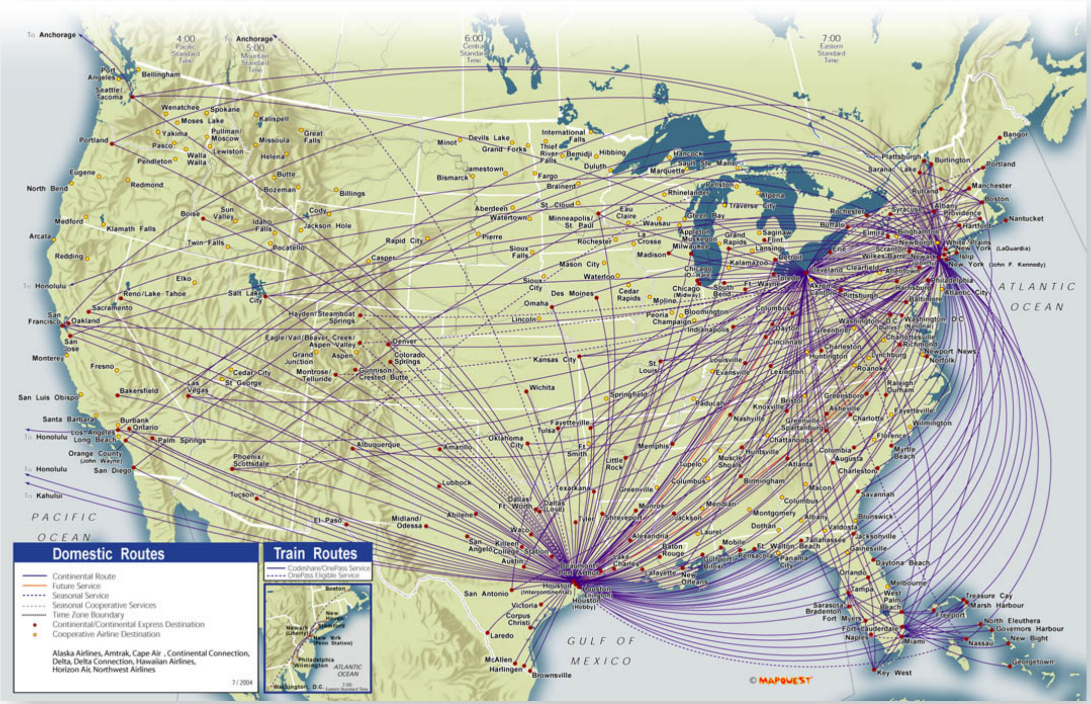
Flight connections between airports

=> large international hubs, small local airports

Protein interaction networks

=> some central,

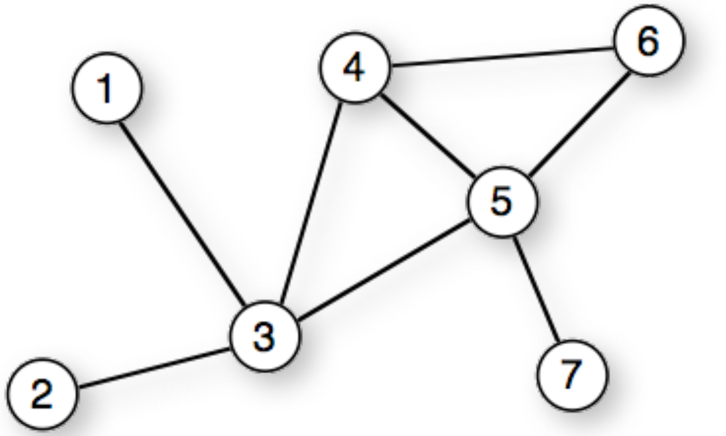
ubiquitous proteins


Domestic Routes
Continental Route
Future Service
Seasonal Service
Seasonal Cooperative Services
Time Zone Boundary
Continental/Continental Express Destination
Cooperative Airline Destination
Alaska Airlines, Amtrak, Cape Air , Continental Connection, Delta, Delta Connection, Hawaiian Airlines, Horizon Air, Northwest Airlines
7 / 2004
Train Routes
Codeshare/OnePass Service
OnePass Eligible Service
Boston
New Haven
Stamford
Newark (Liberty)
New York (Penn Station)
Philadelphia
Wilmington
ATLANTIC OCEAN
Washington, D.C.
PACIFIC OCEAN
ATLANTIC OCEAN
GULF OF MEXICO
To Anchorage
To Honolulu
To Kahului
4:00 Pacific Standard Time
5:00 Mountain Standard Time
6:00 Central Standard Time
7:00 Eastern Standard Time
Houston
Newark
Cleveland
Miami
©MAPQUEST

http://a.parsons.edu/~limam240/blogimages/16_full.jpg

Algorithms on Graphs

How to **represent** a graph in the **computer**?



1. Adjacency list

=> list of neighbors for each node

1:	(3)
2:	(3)
3:	(1, 2, 4, 5)
4:	(3, 5, 6)
5:	(3, 4, 6, 7)
6:	(4, 5)
7:	(5)

+ minimal memory requirement

+ vertices can easily be added or removed

– requires $O(\lambda)$ time to determine whether a certain edge exists

Note: for weighted graphs store pairs of (neighbor label, edge weight)

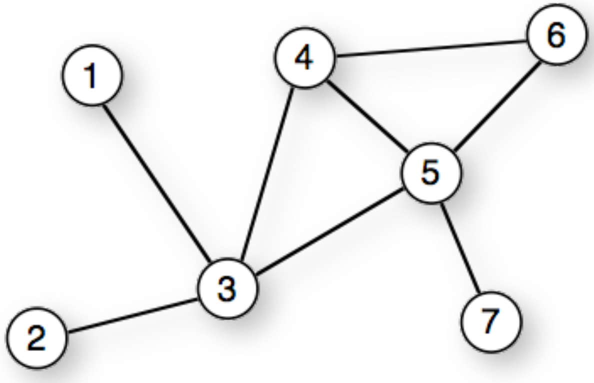
Graph Representation II

2. **Adjacency matrix** (see V1)

→ N x N matrix with entries M_{uv}

M_{uv} = weight when edge between u and v exists,
0 otherwise

→ symmetric for undirected graphs



+ fast $O(1)$ lookup of edges

– large memory requirements

– adding or removing nodes is expensive

	1	2	3	4	5	6	7
1	–	0	1	0	0	0	0
2	0	–	1	0	0	0	0
3	1	1	–	1	1	0	0
4	0	0	1	–	1	1	0
5	0	0	1	1	–	1	1
6	0	0	0	1	1	–	0
7	0	0	0	0	1	0	–

Note: very convenient in programming languages that support sparse multi-dimensional arrays
=> Perl

Graph Representation III

3. **Incidence matrix**

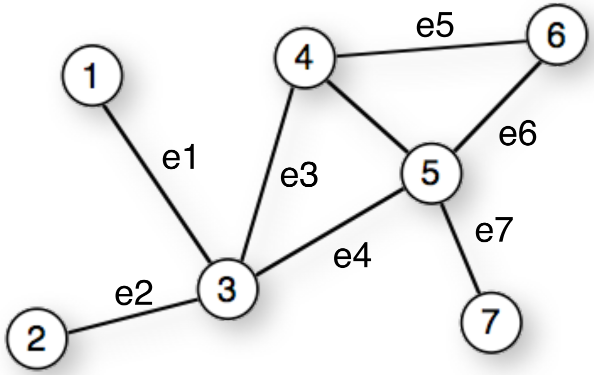
→ *N* x *M* matrix with entries M_{nm}

M_{nm} = weight when edge *m* ends at node *n*

0 otherwise

→ for a plain graph there are two entries per column

→ directed graph:
indicate direction via sign (in/out)



	e1	e2	e3	e4	e5	e6	e7
1	1						
2		1					
3	1	1	1	1			
4			1		1		
5				1		1	1
6					1	1	
7							1

The incidence matrix is a special form of the **stoichiometric matrix** of reaction networks.

V2(b): Structures of Protein Complexes and Subcellular Structures

(1) We normally assume that various enzymes of a biochemical pathway „swim“ in the cytosol and randomly meet the substrate molecules one after another.

Yet, sometimes **multiple enzymes** of a biochemical pathway associate into **large complexes** and „hand over“ the substrates from one active site to the next one.

Advantage: this avoids free diffusion, increases local substrate density.

(2) Membrane transporters and receptors often form **oligomers** in the **membrane**.

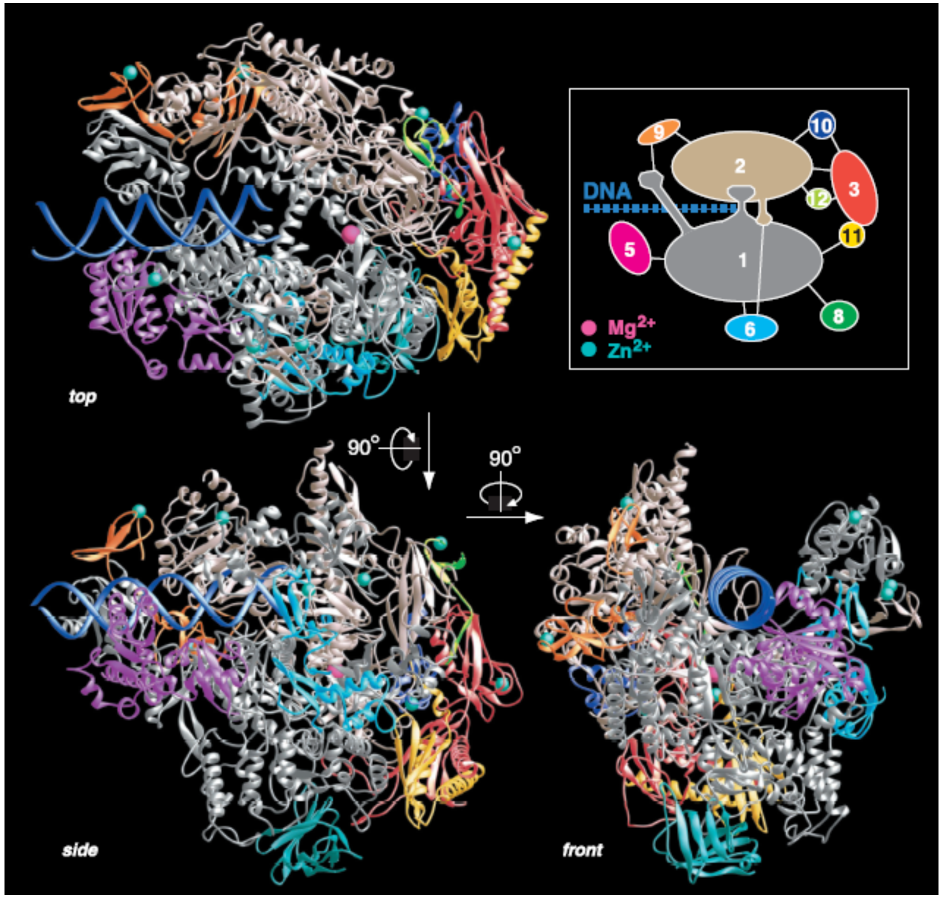
Advantage:

(i) large structures are built from small building blocks (simplicity)

(ii) Oligomer formation can be regulated separately from transcription.

(3) Also: complicated structural components of the cell (e.g. cytoskeleton) are built from many small components (e.g. actin)

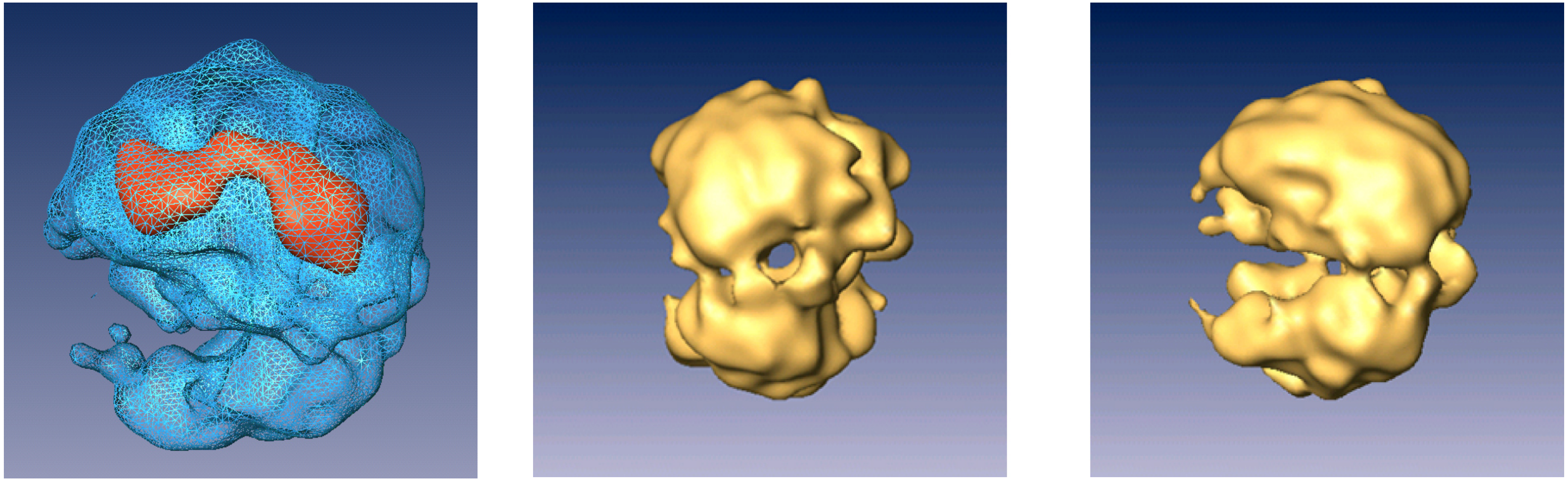
2.1 RNA Polymerase II



RNA polymerase II is the central enzyme of gene expression and synthesizes all messenger RNA in eukaryotes.

Cramer *et al.*, Science 288, 640 (2000)

2.1 RNA processing: splicesome



Structure of a **cellular editor** that "cuts and pastes" the first draft of RNA straight after it is formed from its DNA template.

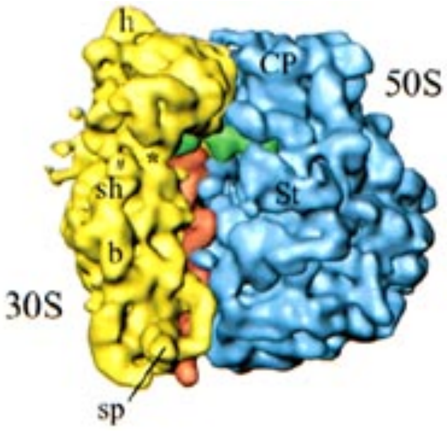
It has two distinct, unequal halves surrounding a tunnel.

Larger part: appears to contain proteins and the short segments of RNA, smaller half: is made up of proteins alone.

On one side, the tunnel opens up into a cavity, which is believed to function as a holding space for the fragile RNA waiting to be processed in the tunnel.

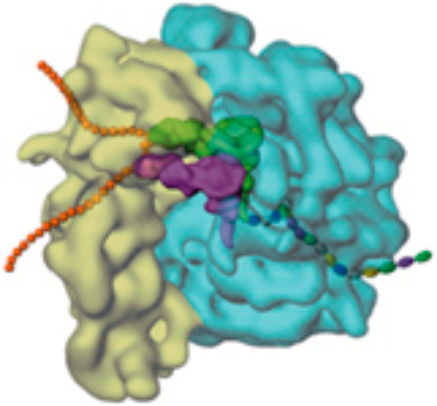
Profs. Ruth and Joseph Sperling, http://www.weizmann.ac.il/

2.1 Protein synthesis: ribosome

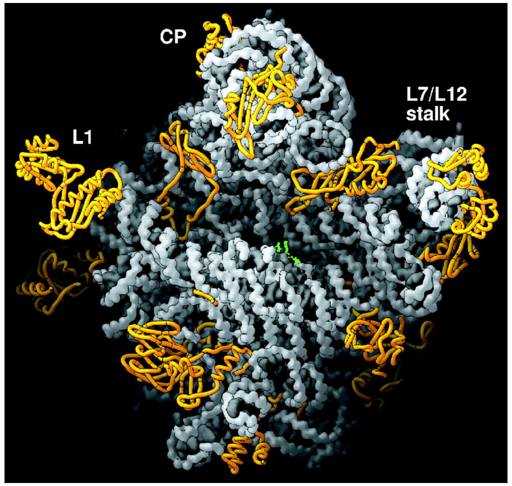


The ribosome is a complex subcellular particle composed of protein and RNA. It is the site of protein synthesis,

http://www.millerandlevine.com/chapter/12/cryo-em.html



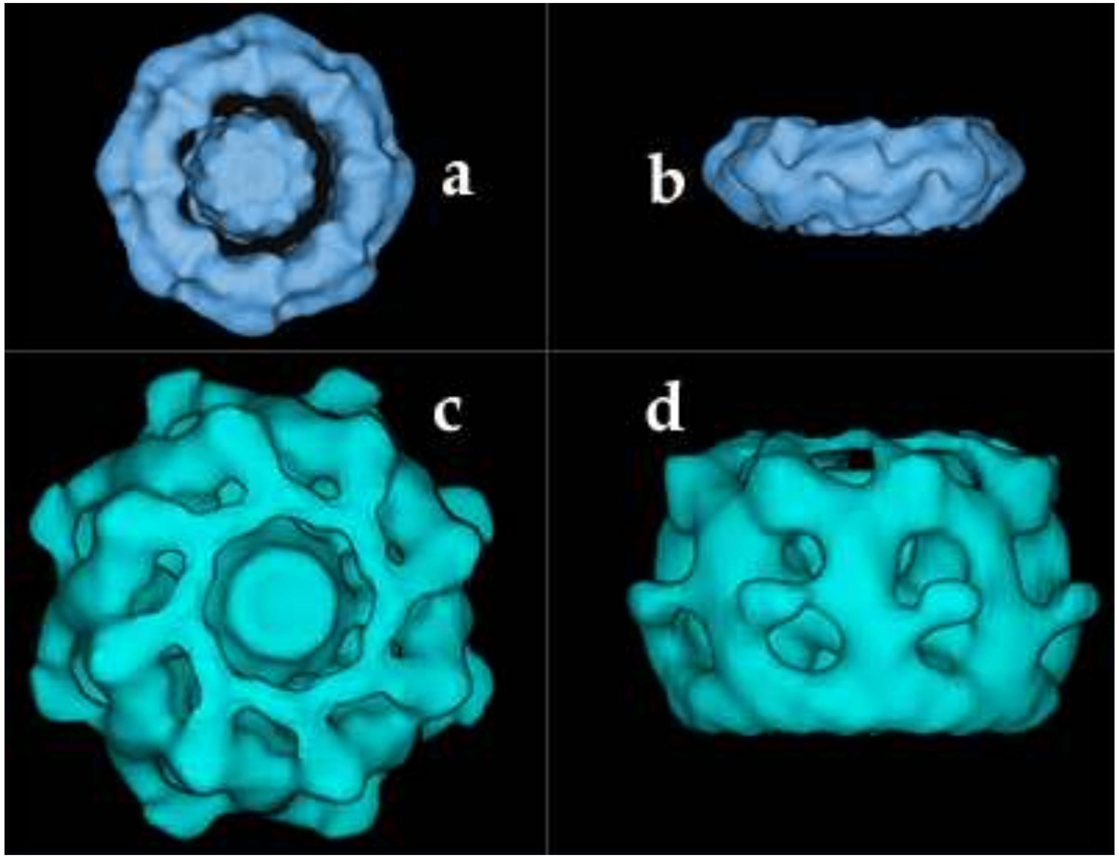
Model of a ribosome with a newly manufactured protein (multicolored beads) exiting on the right.



large ribosomal subunit from *Haloarcula marismortui*. RNA is shown in gray and the protein backbone in yellow.
Ban *et al.* (2000)

Components of ribosome assemble spontaneously in vitro: no helper proteins (assembly chaperones) needed

2.1 Nuclear Pore Complex (NPC)



Three-dimensional image of the NPC obtained by electron microscopy.

A-B The NPC in yeast.

Figure A shows the NPC seen from the cytoplasm while figure B displays a side view.

C-D The NPC in vertebrate (Xenopus).

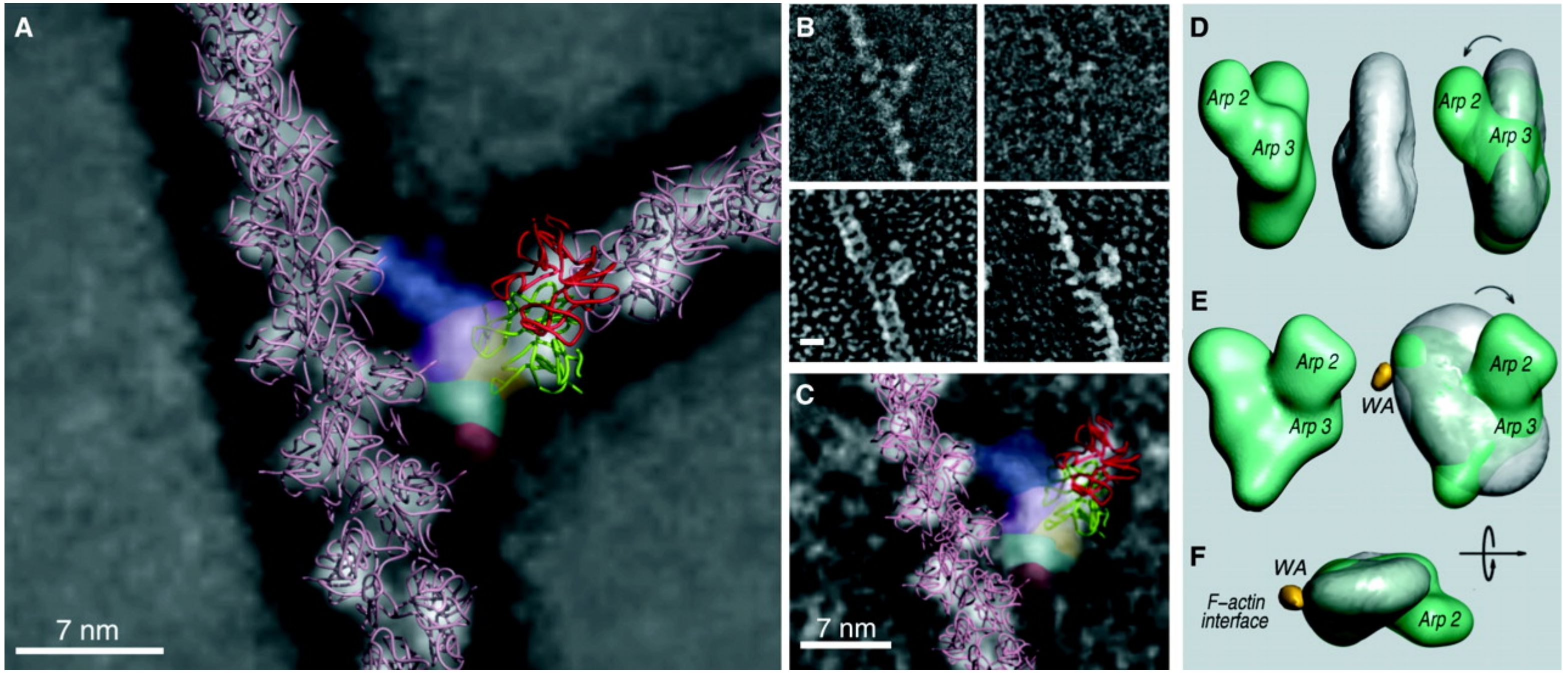
http://www.nobel.se/medicine/educational/dna/a/transport/ncp_em1.html

Three-Dimensional Architecture of the Isolated Yeast Nuclear Pore Complex: Functional and Evolutionary Implications, Qing Yang, Michael P. Rout and Christopher W. Akey. Molecular Cell, 1:223-234, 1998

NPC is a 50-100 MDa protein assembly that regulates and controls trafficking of macromolecules through the nuclear envelope.

Molecular structure:
lecture V20

2.1 Arp2/3 complex



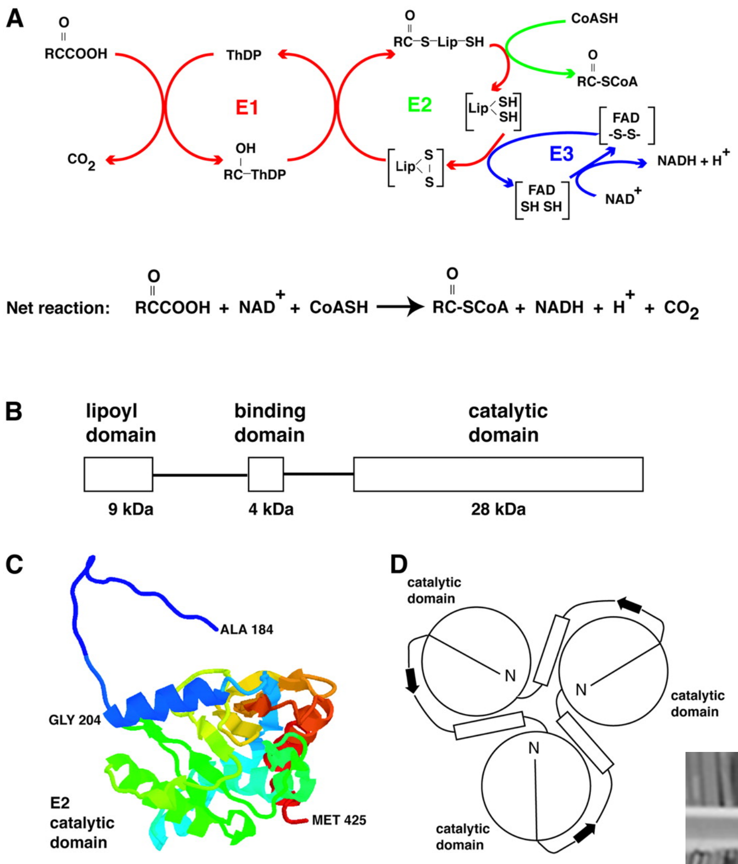
The seven-subunit Arp2/3 complex choreographs the formation of branched actin networks at the leading edge of migrating cells.

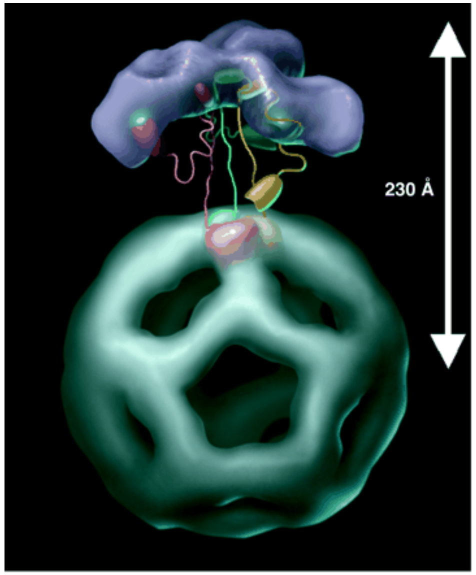
(A) Model of actin filament branches mediated by *Acanthamoeba* Arp2/3 complex.

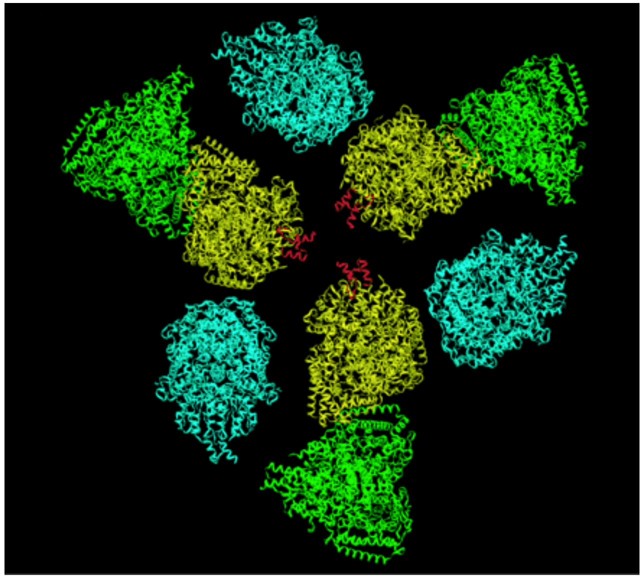
(D) Density representations of the models of actin-bound (green) and the free, WA-activated (as shown in Fig. 1D, gray) Arp2/3 complex.

Volkmann *et al.*, Science 293, 2456 (2001)

2.1 icosahedral pyruvate dehydrogenase complex: a multifunctional catalytic machine



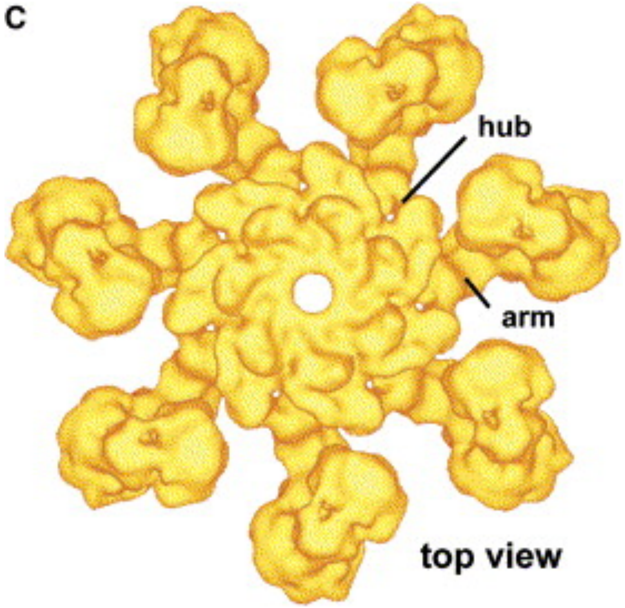


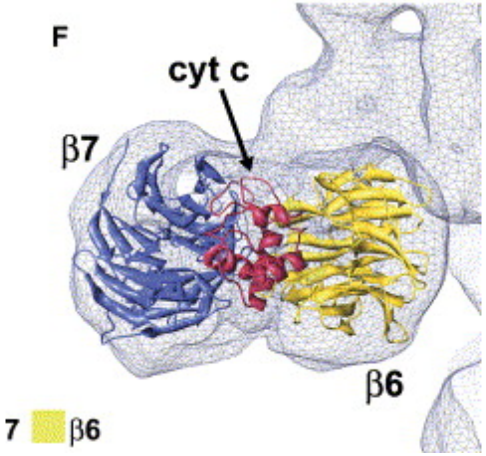


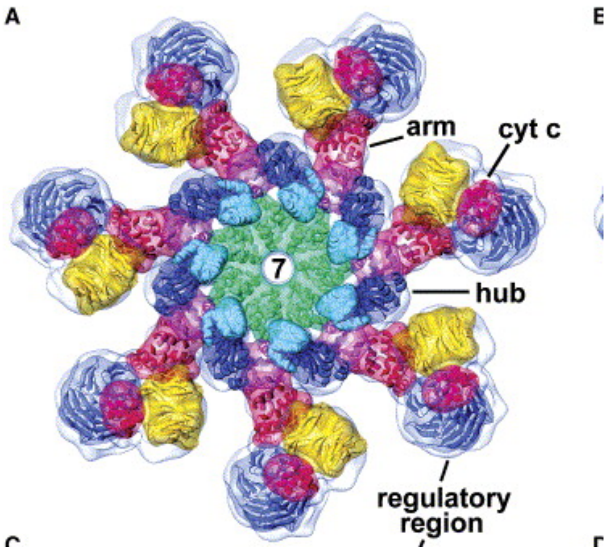
Milne *et al.,* EMBO J. 21, 5587 (2002)

Model for active-site coupling in the E1E2 complex. 3 E1 tetramers (purple) are shown located above the corresponding trimer of E2 catalytic domains in the icosahedral core. Three full-length E2 molecules are shown, colored red, green and yellow. The lipoyl domain of each E2 molecule shuttles between the active sites of E1 and those of E2. The lipoyl domain of the red E2 is shown attached to an E1 active site. The yellow and green lipoyl domains of the other E2 molecules are shown in intermediate positions in the annular region between the core and the outer E1 layer. Selected E1 and E2 active sites are shown as white ovals, although the lipoyl domain can reach additional sites in the complex.

2.1 Apoptosome







Apoptosis is the dominant form of programmed cell death during embryonic development and normal tissue turnover. In addition, apoptosis is upregulated in diseases such as AIDS, and neurodegenerative disorders, while it is downregulated in certain cancers. In apoptosis, death signals are transduced by biochemical pathways to activate caspases, a group of proteases that utilize cysteine at their active sites to cleave specific proteins at aspartate residues. The proteolysis of these critical proteins then initiates cellular events that include chromatin degradation into nucleosomes and organelle destruction. These steps prepare apoptotic cells for phagocytosis and result in the efficient recycling of biochemical resources.

In many cases, apoptotic signals are transmitted to mitochondria, which act as integrators of cell death because both effector and regulatory molecules converge at this organelle. Apoptosis mediated by mitochondria requires the **release** of **cytochrome c** into the cytosol through a process that may involve the formation of specific pores or rupture of the outer membrane. Cytochrome c binds to Apaf-1 and in the presence of dATP/ATP promotes assembly of the apoptosome. This large protein complex then binds and activates procaspase-9.

2.1.2 Categories of Protein Complexes

Complexes can be classified e.g. by function / size / involvement of other components (nucleic acids, carbohydrates, lipids).

Alternatively: mechanistic classification:

(1) transient vs. permanent

(2) obligate vs. non-obligate

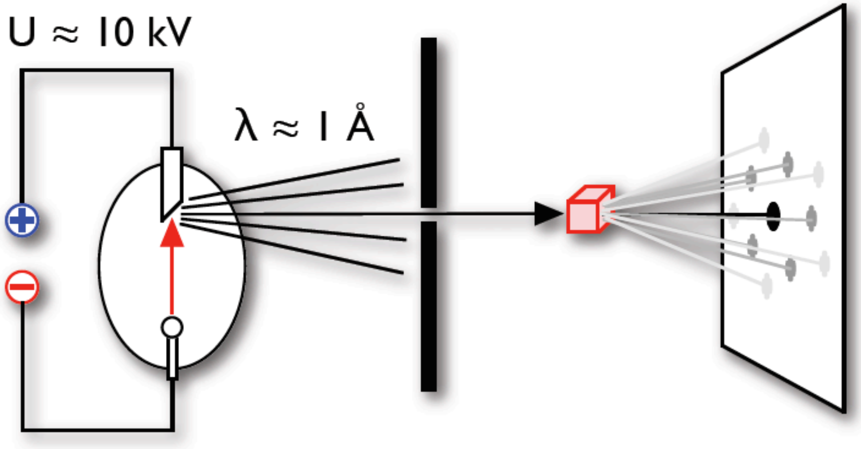
Obligate: components function only when in the bound state.

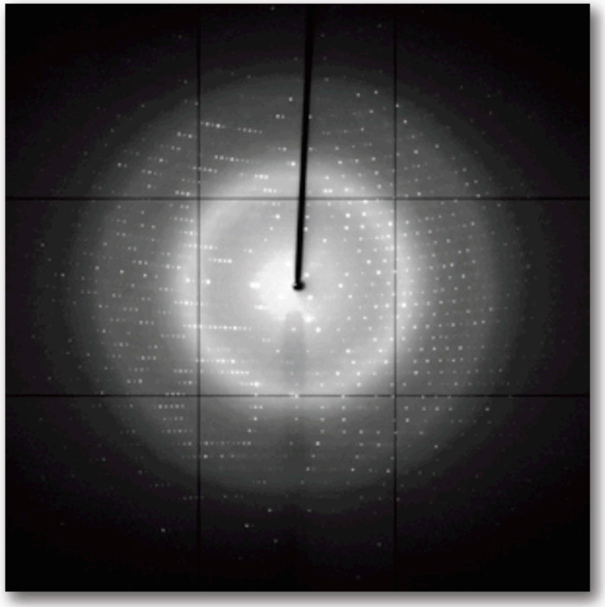
Non-obligate: unbound components can also exist as monomers.

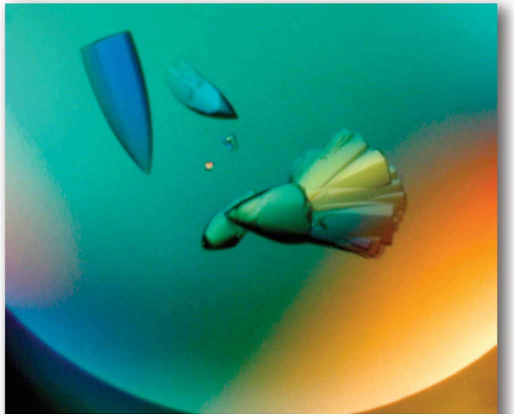
Examples of non-obligate complexes: antibodies, signalling complexes, complexes of RNA polymerase with different initiation and elongation factors.

2.3 Determining molecular 3D structures: X-ray crystallography

Beam of photons (no mass) with high energy, method needs relatively large samples







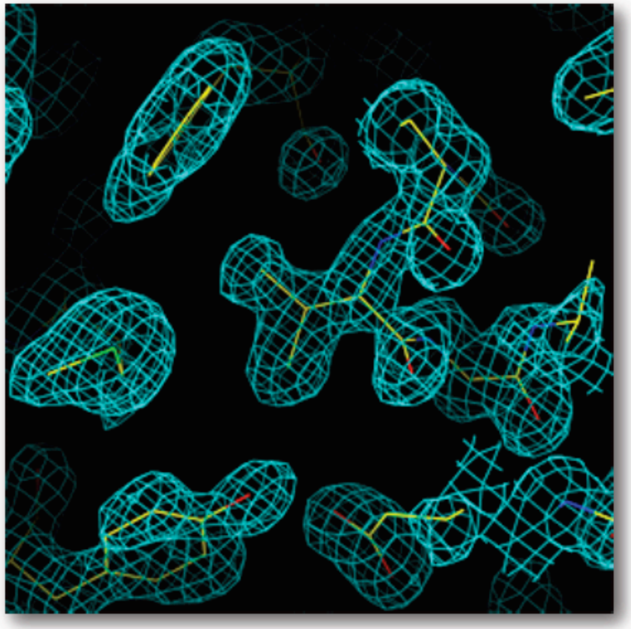
Crystallize proteins, record diffraction patterns of X-rays (scatter at the electron clouds) => reconstruction

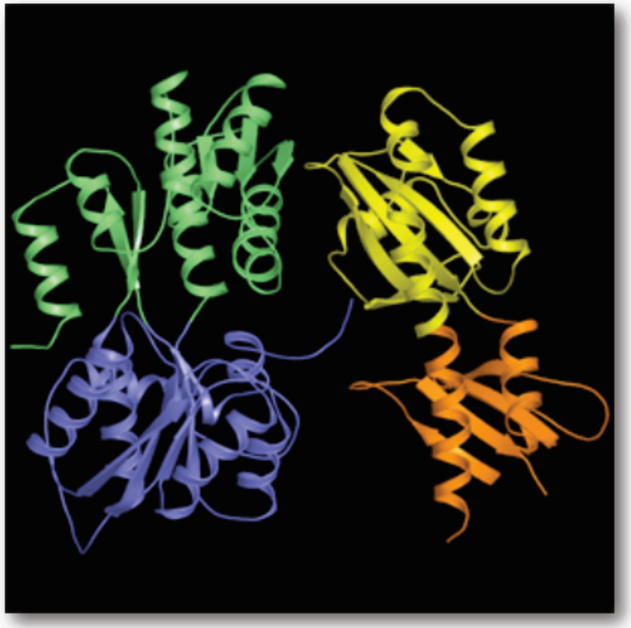
1 keV ≈ 10^5 kJ/mol

http://www.molbio1.princeton.edu/macro/about.html

V 20 – 14

X-ray reconstruction





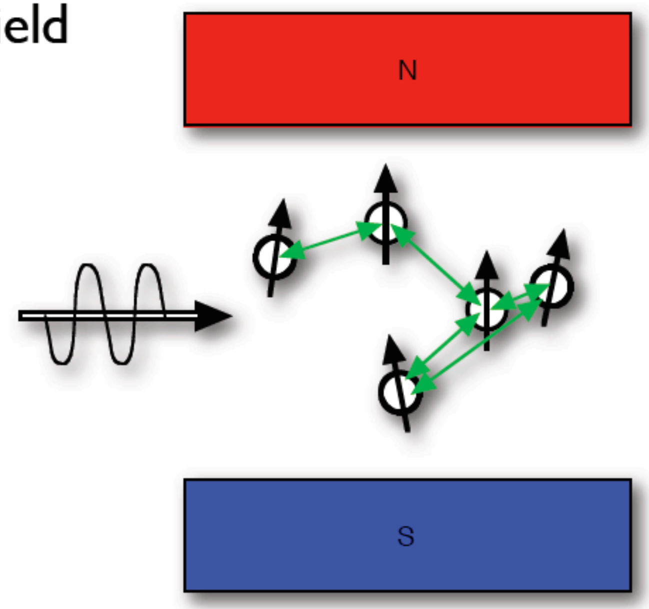
Reconstruct electron density maps from diffraction patterns at multiple wavelengths and orientations, refine structure computationally

Main problem: proteins do not like to crystallize (especially membrane proteins)

PDB:	experimental technique == X-Ray:	43138 results
	X-Ray && "in the membrane":	1232 results

http://www.molbio1.princeton.edu/macro/about.html

2.3.2 Nuclear magnetic resonance



Place sample with the proteins into strong magnetic field
=> nuclei with non-zero magnetic moment
(^{1}H, ^{13}C, ^{15}N) align

Apply electromagnetic RF field
=> resonances of the nuclear spins depending on

- atom type
- chemical environment

=> extract distances between close by ^{1}H atoms
(distance constraints)

+ no crystallization required, proteins in physiological environment

+ atomic resolution

– too much overlap for larger proteins (≥15 kDa)

2.3.3 Electron microscopy

Like X-ray crystallography, but with electrons (electrons have mass)

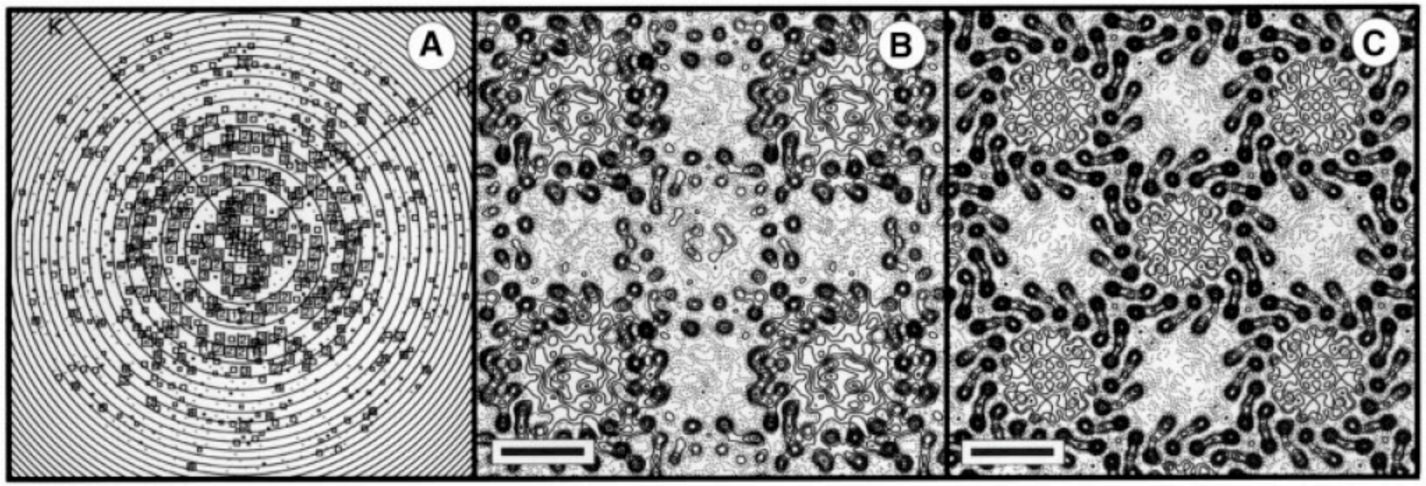
=> much stronger interaction with electron clouds

=> works already on 2D crystals (membrane proteins!) or even **single particles** (average over many of them)

=> strong radiation damage of the sample

=> resolution limited (keep electron energy low) (longer wavelength)

For 3D tomography: rotate sample

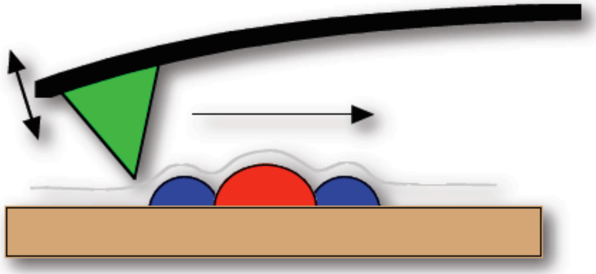


Cryo-EM images of the LH1-RC core complexes of *Rhodospirillum rubrum* at 8.5 Å resolution

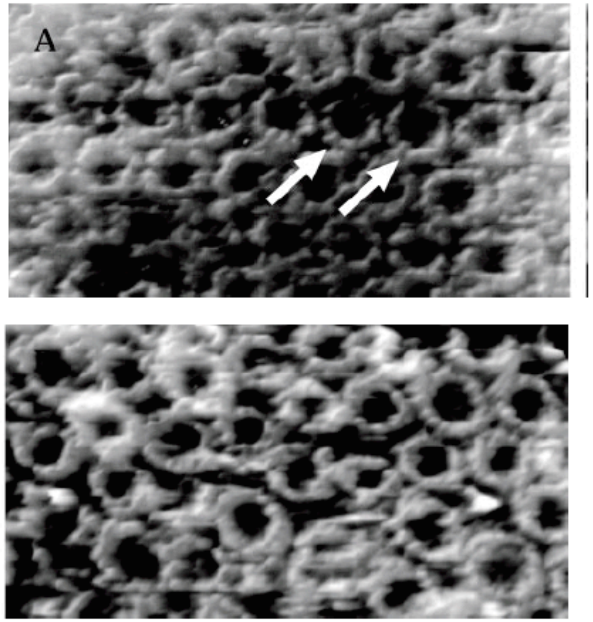
Fig. 3. (**A**) Representation of amplitudes of Fourier components calculated for one image of a glucose-embedded tetragonal crystal. Numbers and box sizes correspond to the spot IQ value, with spots of the highest signal-to-noise ratio having an IQ of 1 (Henderson *et al*., 1986). Spots are shown to a resolution of 1/6 Å^{-1}. (**B**) 8.5 Å resolution projection map calculated from five averaged images of glucose-embedded tetragonal crystals, assuming *p*1 symmetry. Contouring is at 0.5 r.m.s. density with density above mean (protein) represented by solid contours. Scale bar: 50 Å. (**C**) As (B), with *p*42$_1$2 symmetry applied.

Jamieson etal, *EMBO J* **21** (2003) 3927

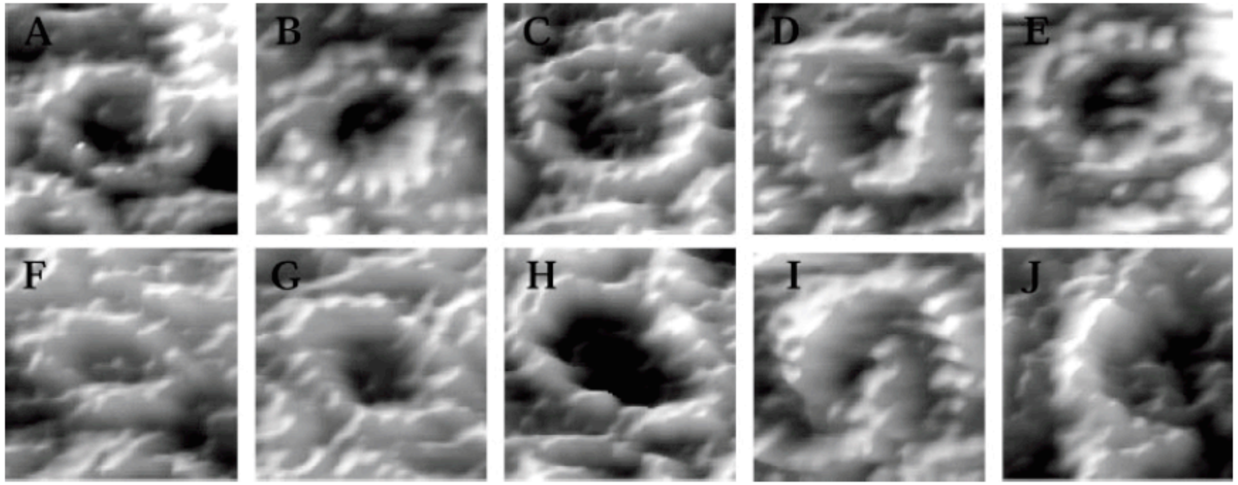
Atomic force microscopy



Scan membrane with proteins
(in physiological conditions)
=> protein arrangement (coarse view)
=> protein shape (high resolution)

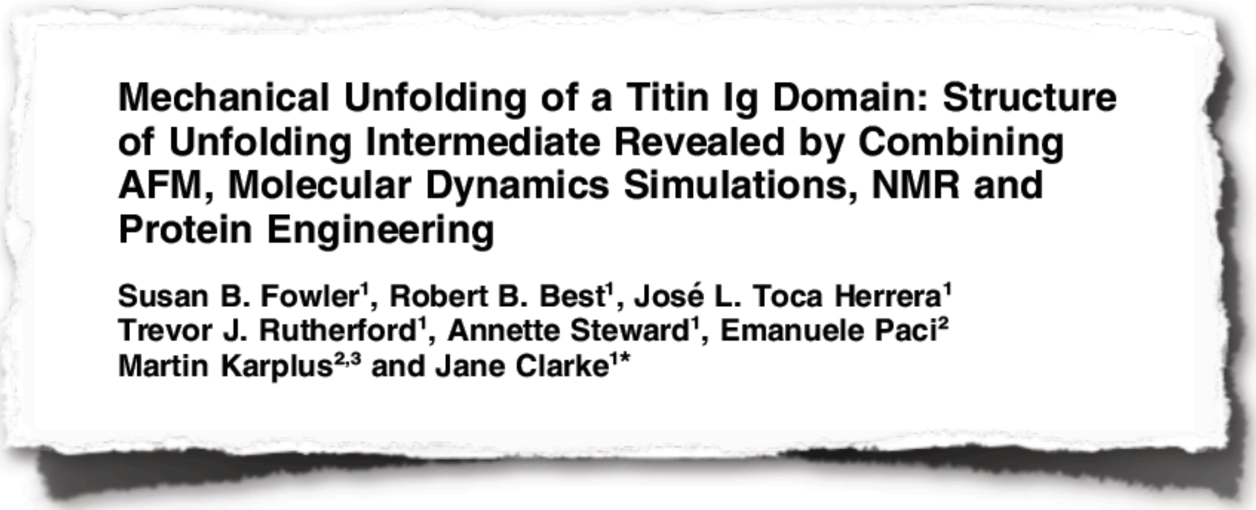


Shapes and sizes of monomeric
LH1 from purple bacteria



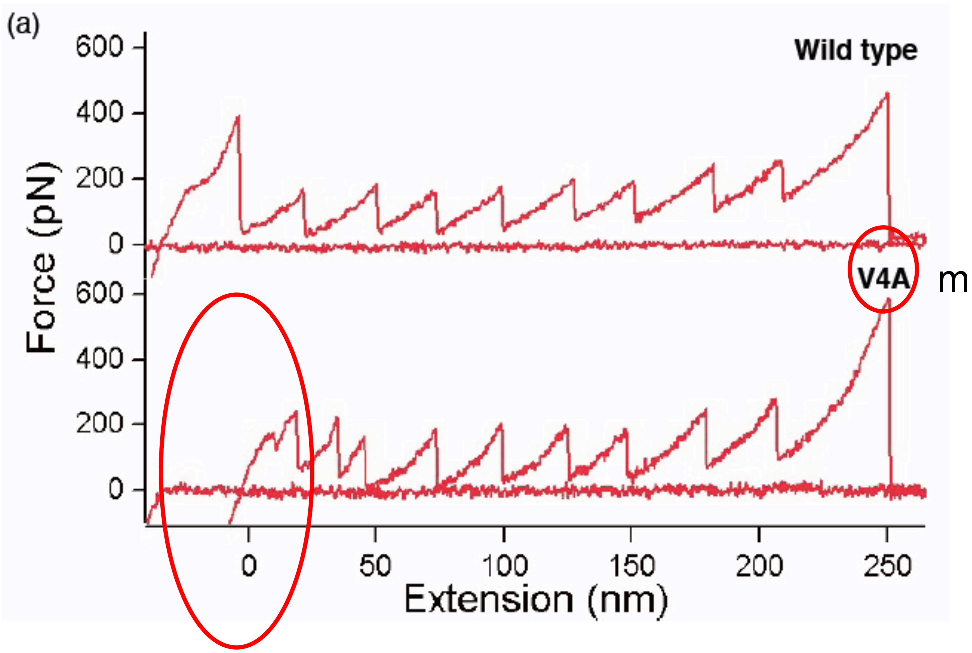
Bahatyrova etal, *J Biol Chem* **279** (2004) 21327

AFM pulling



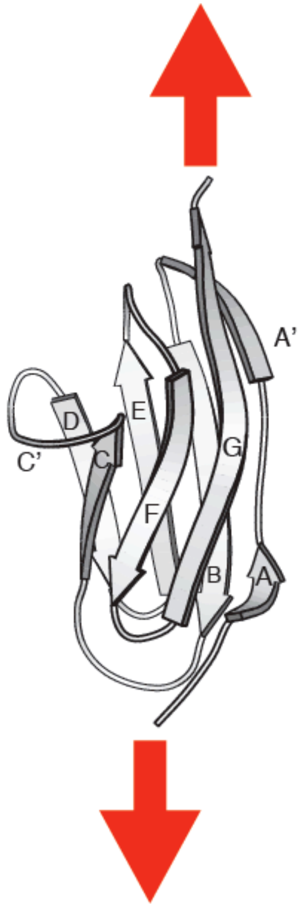
Mechanical Unfolding of a Titin Ig Domain: Structure of Unfolding Intermediate Revealed by Combining AFM, Molecular Dynamics Simulations, NMR and Protein Engineering

Susan B. Fowler[1], Robert B. Best[1], José L. Toca Herrera[1]
Trevor J. Rutherford[1], Annette Steward[1], Emanuele Paci[2]
Martin Karplus[2,3] and Jane Clarke[1*]



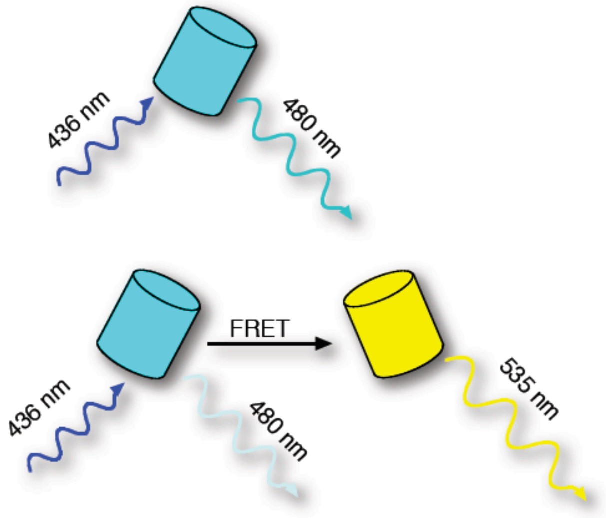
mutant

Can also be applied to protein complexes



J. Mol. Biol. **322** (2002) 841–849

2.3.6 Fluorescence energy transfer



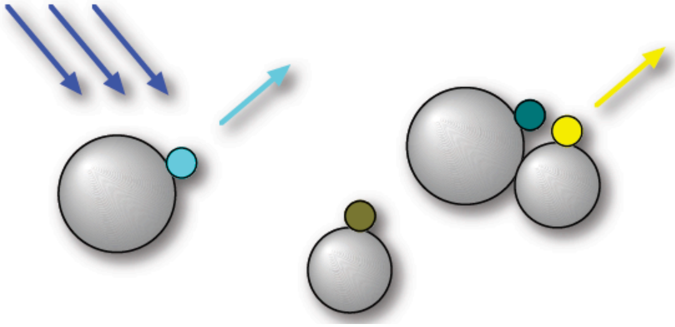
Chromophore of the cyan flourescent protein (CFP) absorbs at 436 nm and emits at 480 nm, YFP absorbs at 480 nm and emits at 535 nm.

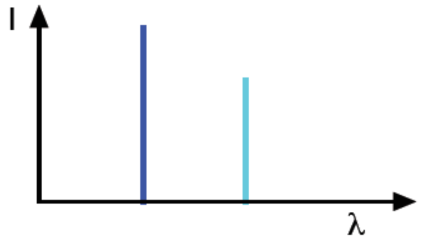
=> **resonant** (non-radiative) energy **transfer** from CFP onto YFP when both are close enough

Resonant Förster transfer ∝ d^{-6}

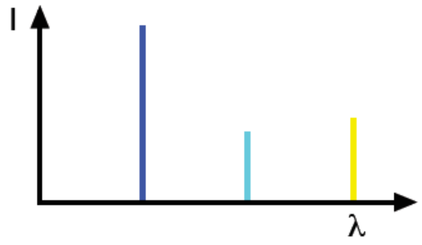
YFP: yellow fluorescent protein

Tag two potential **complex partners** with CFP and YFP and measure **flourescence spectrum**:



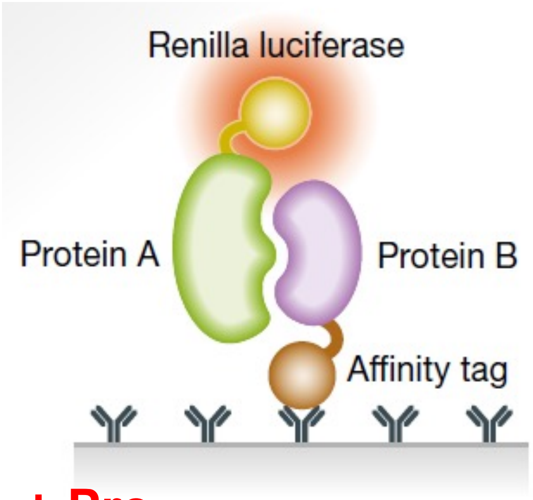


Observed when CFP and YFP are far away



Observed when CFP and YFP are close

Lumier-based mammalian interactome mapping



LUMIER assay is based on co-immunoprecipitation. Protein A is fused to Renilla luciferase, while Protein B is linked to an affinity tag. Tagged constructs are transfected into appropriate cell lines where they are overexpressed. Cells are then lysed and protein B is immunoprecipitated using an appropriate antibody against the affinity tag.

+ Pro

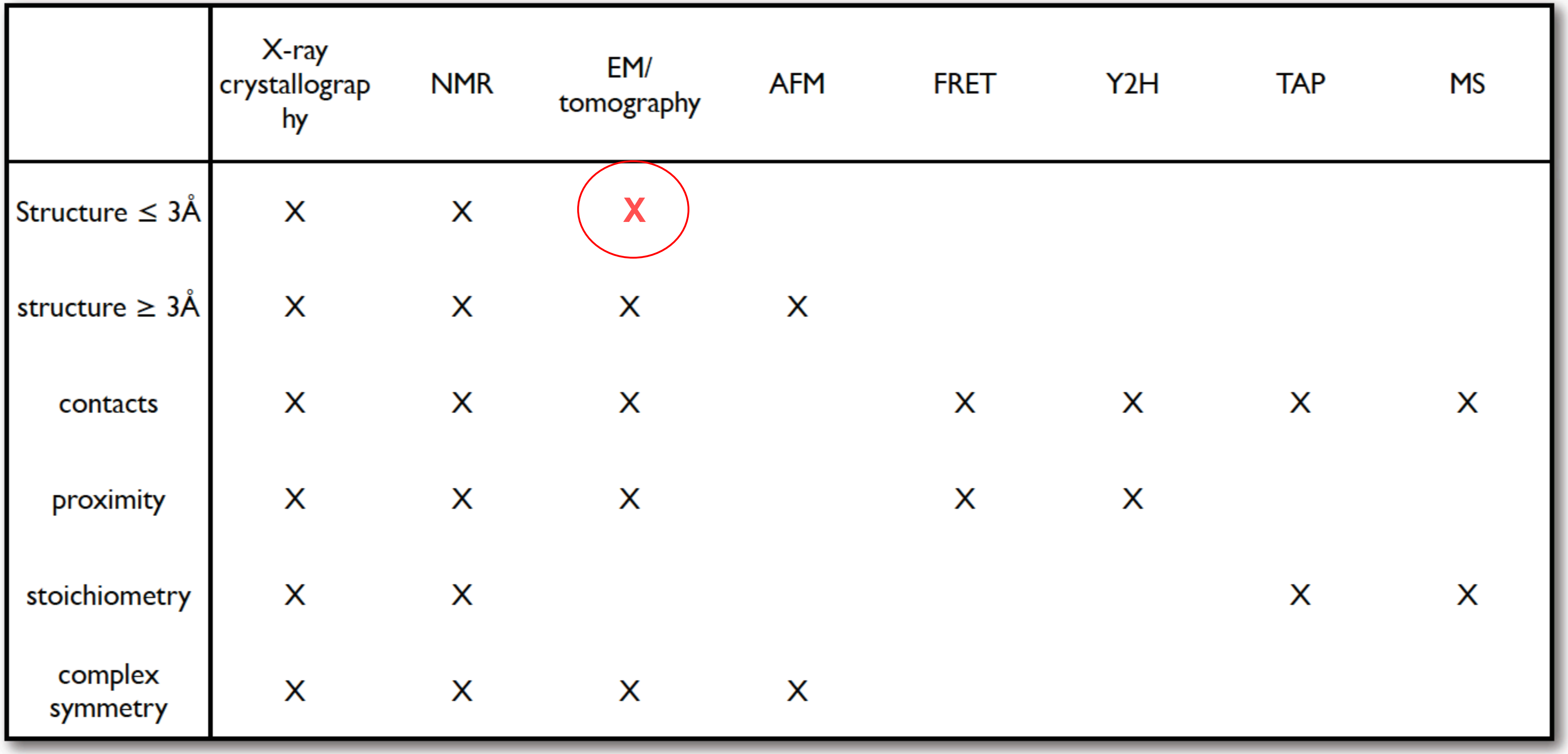
+ Easy to perform, can be used in a HT screening format.
+ Can be used in different cell lines.
+ Well suited for binary interactions, indirect interactions can also be detected

- Con

- **Cells need to be lysed** prior to immunoprecipitation. This can result in the disruption of weak and transient PPIs, as well as the introduction of potential artifacts (e.g., by bringing together proteins in the lysate, which might not normally interact with one another in the cell, destabilizing proteins and exposing previously concealed non-native binding surfaces).

Snider et al. Mol. Syst. Biol. 11, 848 (2015)

Structural techniques - overview

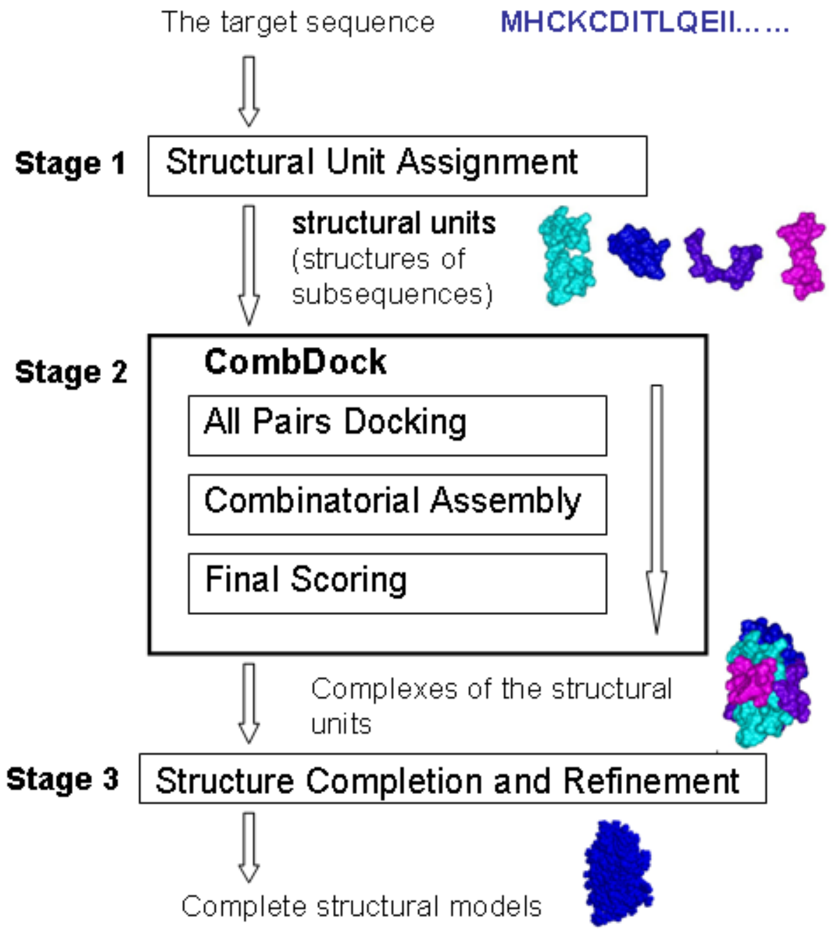


	X-ray crystallography	NMR	EM/ tomography	AFM	FRET	Y2H	TAP	MS
Structure ≤ 3Å	X	X	X					
structure ≥ 3Å	X	X	X	X				
contacts	X	X	X		X	X	X	X
proximity	X	X	X		X	X		
stoichiometry	X	X					X	X
complex symmetry	X	X	X	X				

Thanks to improvements
in EM detectors

Predicting Structures of Protein Complexes from Connectivities: CombDock

`CombDock:` automated approach for predicting 3D structure of heterogenous multimolecular assemblies.



The target sequence
MHCKCDITLQEII......
Stage 1
Structural Unit Assignment
structural units
(structures of subsequences)
Stage 2
CombDock
All Pairs Docking
Combinatorial Assembly
Final Scoring
Complexes of the structural units
Stage 3
Structure Completion and Refinement
Complete structural models

Input: structures of *N* individual proteins

Problem appears more difficult than the pairwise docking problem.

Idea: exploit additional geometric constraints that are part of the combinatorial problem.

Haim Wolfson
Tel Aviv University
http://www.cs.tau.ac.il/~wolfson/

Inbar et al., J. Mol. Biol. 349, 435 (2005)

Review: pairwise docking: Katchalski-Kazir algorithm

Discretize proteins A and B on a grid.

Every node is assigned a value

$$f_{A_{l,m,n}} = \begin{cases} 1 & : \quad \text{surface of molecule} \\ \rho & : \quad \text{core of molecule} \\ 0 & : \quad \text{open space} \end{cases}$$

and

$$f_{B_{l,m,n}} = \begin{cases} 1 & : \quad \text{inside molecule} \\ 0 & : \quad \text{open space} \end{cases}$$

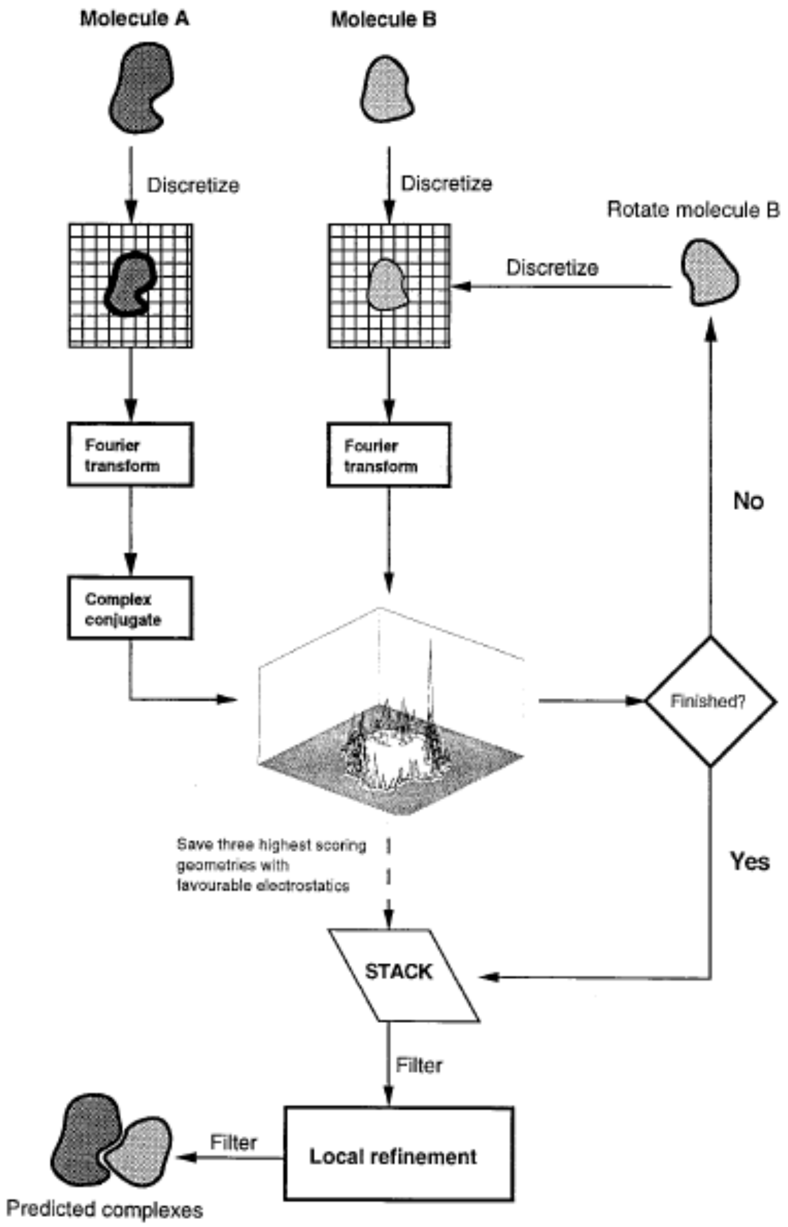
The correlation function of f_A and f_B is:

$$f_{C_{\alpha,\beta,\gamma}} = \sum_{l=1}^{N} \sum_{m=1}^{N} \sum_{n=1}^{N} f_{A_{l,m,n}} \times f_{B_{l+\alpha,m+\beta,n+\gamma}}$$

Use FFT to compute correlation efficiently (see V3).

Output: solutions with best surface complementarity.

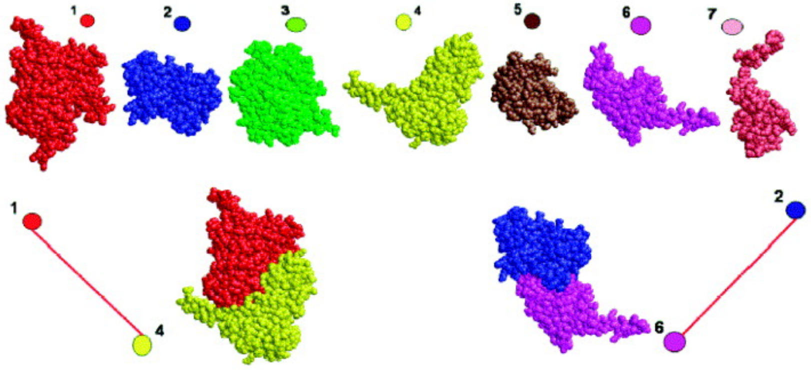
Gabb et al. J. Mol. Biol. (1997)



(1) All pairs docking module

Aim: predict putative pairwise interactions

Based on the *N* individual protein structures perform pairwise docking for each of the *N* (*N* - 1) / 2 pairs of proteins

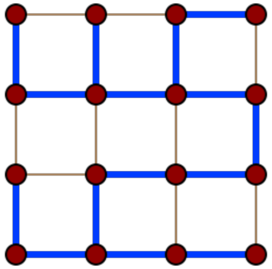


Since the correct scoring of pairwise-docking is difficult, the **correct solution** may be among the first few hundred solutions.

→ keep *K* best solutions for each pair of proteins.

Inbal *et al.* varied *K* from dozens to hundreds.

Spanning tree = a graph that connects all vertices and has no circles



Inbar et al., J. Mol. Biol. 349, 435 (2005)

Review: Spanning trees

Given a connected and undirected graph, a **spanning tree** of that graph is a subgraph that is a tree and connects all the vertices together.

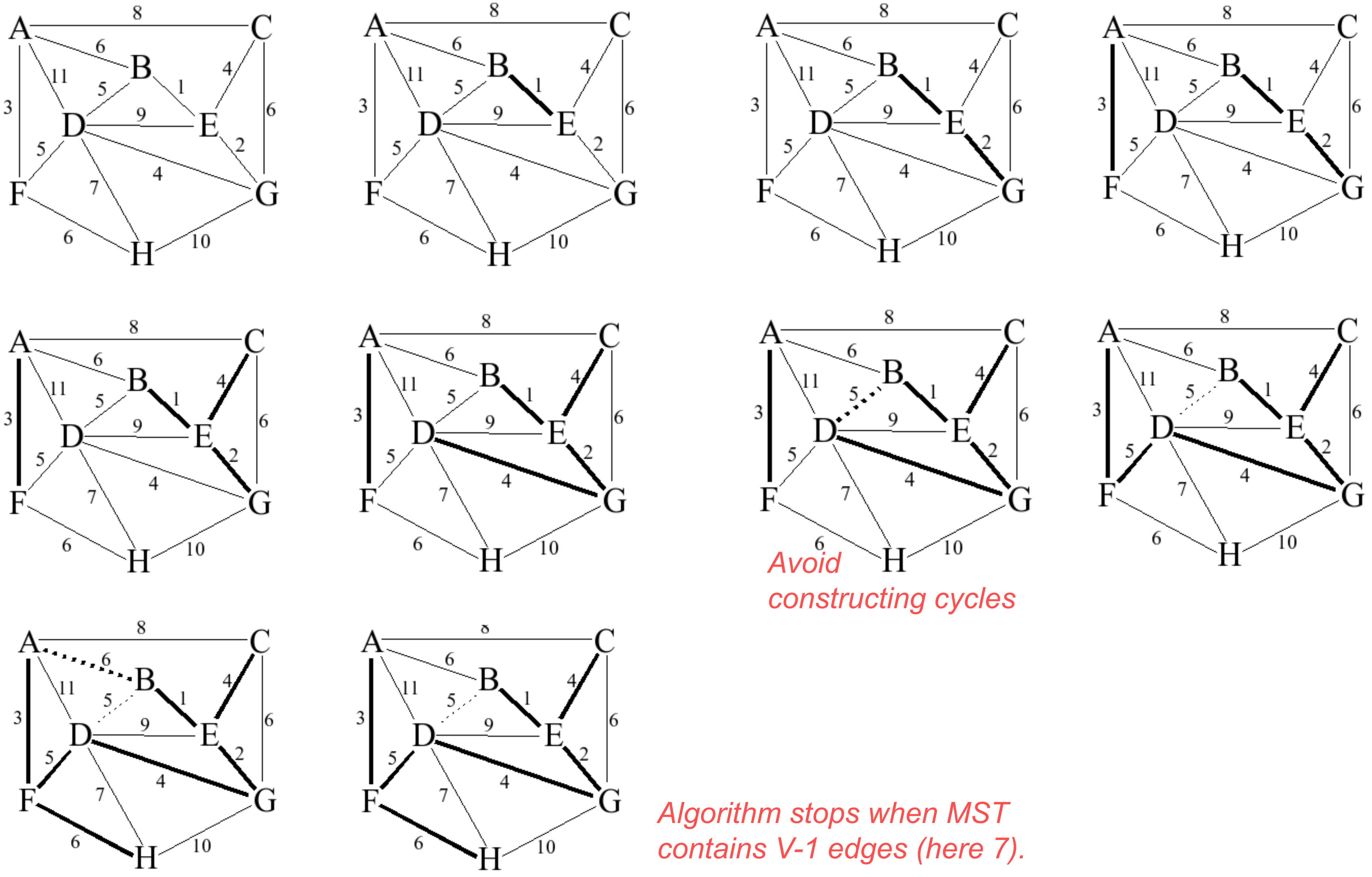
A single graph can have many different spanning trees.

A **minimum spanning tree (MST)** or minimum weight spanning tree for a weighted, connected and undirected graph is a spanning tree with weight less than or equal to the weight of every other spanning tree. The weight of a spanning tree is the sum of weights given to each edge of the spanning tree.

For a graph with V vertices, a minimum spanning tree has (V – 1) edges.

Kruskal's algorithm for finding a minimum spanning tree.
1. *Sort all the edges in non-decreasing order of their weight.*
2. *Pick the edge with smallest weight. Check if it forms a cycle with the spanning tree formed so far. If cycle is not formed, include this edge. Else, discard it.*
3. *Repeat step#2 until there are (V-1) edges in the spanning tree.*

Example: Spanning tree – algorithm of Kruskal



Avoid constructing cycles

Algorithm stops when MST contains V-1 edges (here 7).

(2) Combinatorial assembly module

Input: *N* subunits and *N* (*N* - 1) / 2 sets of *K* scored transformations.
These are the candidate interactions.

Reduction to a spanning tree

Build weighted graph representing the input:

- each protein structure	= vertex
- each transformation (docking orientation)	
	= edge connecting the corresponding vertices
- edge weight	= **docking score** of the transformation

→ Since the input contains *K* transformations for each pair of subunits, we get a complete graph with ***K* parallel edges** between each pair of vertices.

Inbar et al., J. Mol. Biol. 349, 435 (2005)
www.wikipedia.org

(2) Combinatorial assembly module

For 2 subunits, each candidate binary docking complex is represented by an **edge** and the 2 vertices.

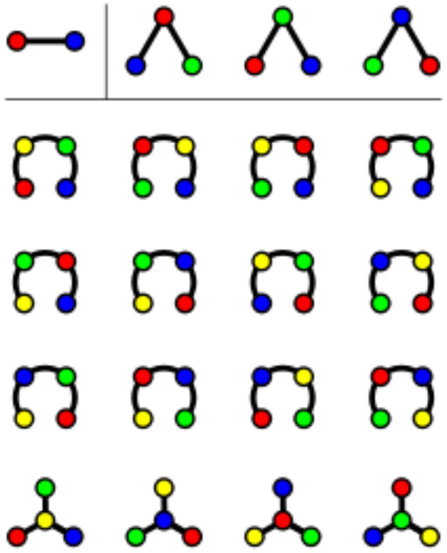
For the full complex, a candidate complex is represented by a **spanning tree**. Each spanning tree of the input graph represents a particular **3D structure** for the complex of all input structures.

→ Problem of finding 3D structures of complexes is equivalent to finding spanning trees.

The number of spanning trees in a complete graph with N nodes and **no parallel edges** is N^{N-2} (Cayley's formula).

Here, the input graph has K parallel edges between each pair of vertices. → the number of spanning trees is $N^{N-2} K^{N-1}$.

→ Exhaustive searches are infeasible!



Cayley's formula (the number of different trees on n vertices is n^{n-2}, graphically demonstrated for graphs with 2, 3 and 4 nodes.

Inbar et al., J. Mol. Biol. 349, 435 (2005)
www.wikipedia.org

(2) Combinatorial assembly module:algorithm

`CombDock` algorithm uses 2 basic principles:
(1) hierarchical construction of the spanning tree
(2) greedy selection of subtrees

→ 2 subtrees of smaller size (that were previously generated) are connected with an input edge to generate trees with *i* vertices

In this way, the common parts of different trees are generated only once.

When connecting subtrees, check whether there are severe **penetrations** between pairs of subunits that are represented by different subtrees.

Inbar et al., J. Mol. Biol. 349, 435 (2005)

(2) Combinatorial assembly module:algorithm

Stage 1: algorithm start with trees of size 1.
Each tree contains a single vertex that represents a subunit.

Stage *i*: the tree complexes that consist of exactly *i* vertices (subunits) are generated by connecting 2 trees generated at a lower stage with an input edge transformation.

Tree complexes that fulfil the penetration constraint are kept for the next stages.

Because it is impractical to search all valid spanning trees, the algorithm performs a greedy selection of subtrees.

For each subset of vertices, the algorithm keeps only the *D* best-scoring valid trees that connect them.

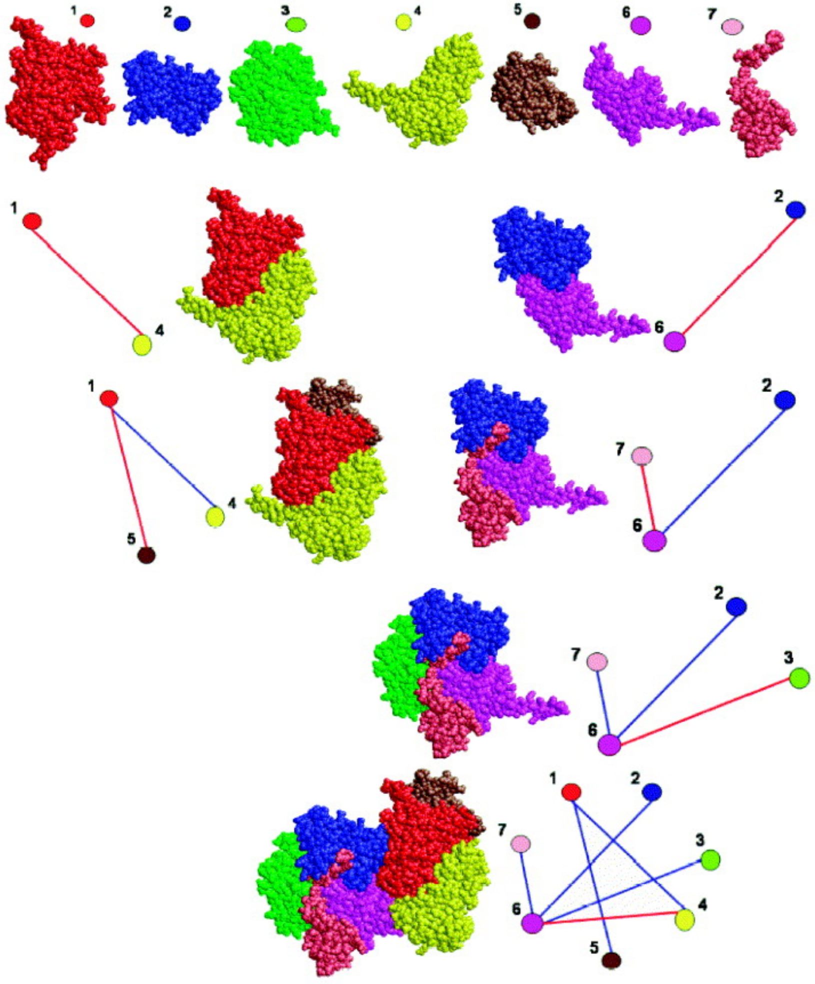
The **tree score** is the sum of its edge weights (pairwise docking scores).

Inbar et al., J. Mol. Biol. 349, 435 (2005)

Example: arp2/3 complex

The arp2/3 complex consists of 7 subunits (top).

Shown are only the complexes of the different stages that were relevant to the construction of the third-best scoring solution with RMSD 1.2 Å (bottom).



Red edge: transformation of the current stage,

Blue edges: transformations of previous stages.

Inbar et al., J. Mol. Biol. 349, 435 (2005)

Final scoring

A **geometric score** evaluates the shape complementarity between the subunits:

- check distances between surface points on adjacent subunits.

- close surface points increase score,

- penetrating surface points decrease score.

Physico-chemical component of the final score counts all surface points that belong to non-polar atoms = this gives an estimate of the hydrophobic effect.

Inbar et al., J. Mol. Biol. 349, 435 (2005)

Clustering of solutions

Clustering of solutions:

(1) compute **contact maps** between subunits: array of $N\,(\,N-1\,)$ bins.

If two subunits are in contact within the complex,
set the corresponding bit to 1, and to 0 otherwise.

(2) superimpose complexes that have the same contact map
and compute RMSD between C^{α} atoms.

If this distance is less than a threshold, consider complexes
as members of a **cluster**.

From each cluster, keep only the complex with the highest score.

Inbar et al., J. Mol. Biol. 349, 435 (2005)

Performance for known complexes

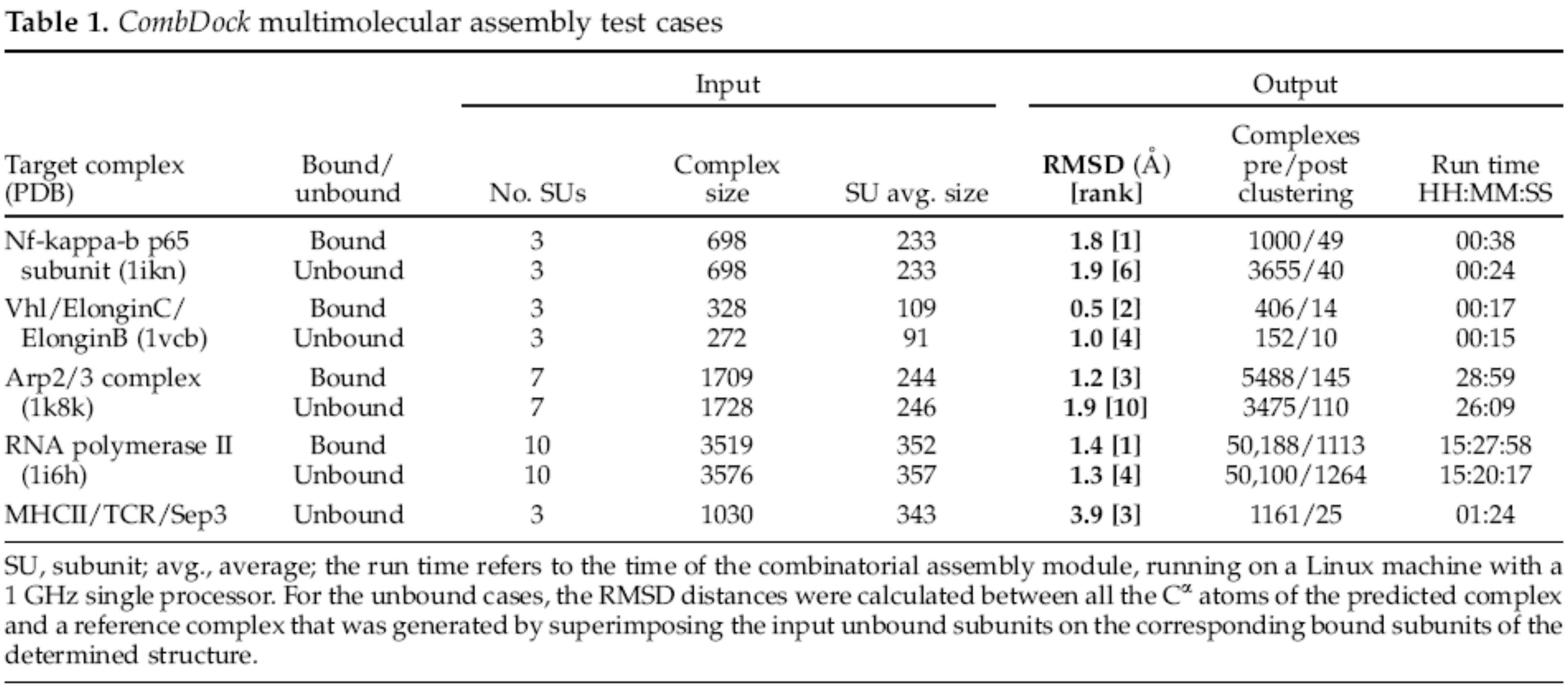


Table 1. *CombDock* multimolecular assembly test cases

		Input			Output		
Target complex (PDB)	Bound/ unbound	No. SUs	Complex size	SU avg. size	**RMSD** (Å) **[rank]**	Complexes pre/post clustering	Run time HH:MM:SS
Nf-kappa-b p65 subunit (1ikn)	Bound	3	698	233	**1.8 [1]**	1000/49	00:38
	Unbound	3	698	233	**1.9 [6]**	3655/40	00:24
Vhl/ElonginC/ ElonginB (1vcb)	Bound	3	328	109	**0.5 [2]**	406/14	00:17
	Unbound	3	272	91	**1.0 [4]**	152/10	00:15
Arp2/3 complex (1k8k)	Bound	7	1709	244	**1.2 [3]**	5488/145	28:59
	Unbound	7	1728	246	**1.9 [10]**	3475/110	26:09
RNA polymerase II (1i6h)	Bound	10	3519	352	**1.4 [1]**	50,188/1113	15:27:58
	Unbound	10	3576	357	**1.3 [4]**	50,100/1264	15:20:17
MHCII/TCR/Sep3	Unbound	3	1030	343	**3.9 [3]**	1161/25	01:24

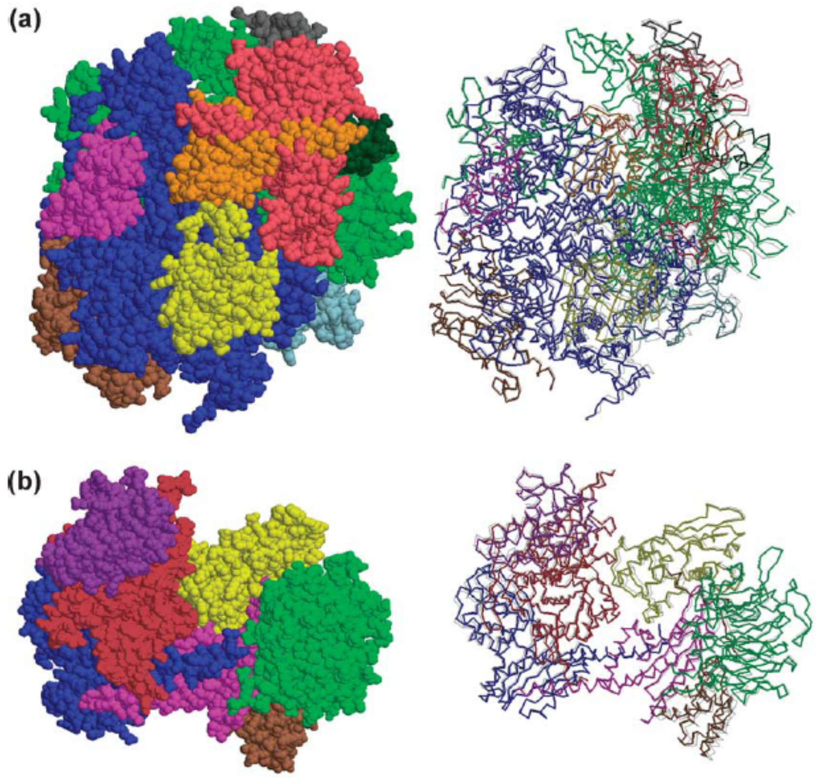
SU, subunit; avg., average; the run time refers to the time of the combinatorial assembly module, running on a Linux machine with a 1 GHz single processor. For the unbound cases, the RMSD distances were calculated between all the C^{α} atoms of the predicted complex and a reference complex that was generated by superimposing the input unbound subunits on the corresponding bound subunits of the determined structure.

Inbar et al., J. Mol. Biol. 349, 435 (2005)

Examples of large complexes

CombDock solution

solution superposed on the crystal structure (gray thiner lines)



(a) the bestranked complex of the 10 subunits of **RNA polymerase II**, RMSD 1.4 Å.

(b) the third-best scoring assembly of the 7 subunits of the **arp2/3 complex**, RMSD 1.2 Å.

CombDock is not as succesful for docking „unbound“ subunit structures that structurally differ from „bound“ conformations.

Inbar et al., J. Mol. Biol. 349, 435 (2005)

Summary

Today:

- Scale-free vs. random graphs
- Examples of important protein complexes
- Exp. methods to determine protein interactions
- Combinatorial assembly of protein complexes (CombDock)

Next lecture V3:

- Further computational methods to assemble higher-order protein complexes
- Docking into EM maps (FFT)