

Molecular Interaction Networks

Abdullah Kahraman, Ph.D.

**Head of Clinical Bioinformatics
Institute for Pathology and Molecular Pathology
University Hospital Zurich, Switzerland**



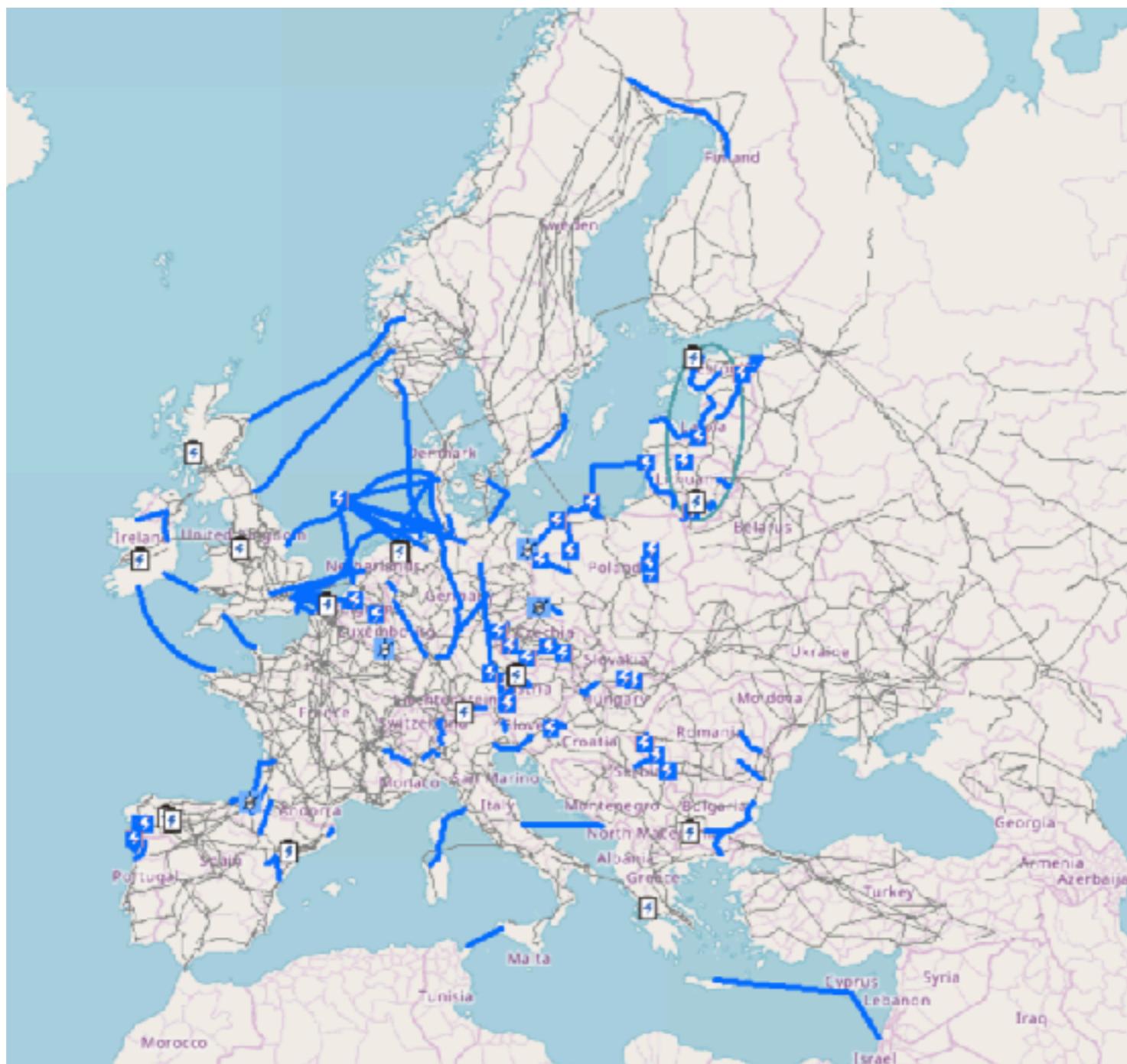
Outlook

- **Networks and Graphs**
- **Molecular Interaction Networks in Biology**
- **High-Throughput Methods to Probe Protein Interaction Networks**
- **Predicting Protein Interaction Networks**
- **Network Properties**
- **Shortest-Distance Algorithm**
- **Visualising Molecular Interaction Networks**

Networks and Graphs

Networks from our daily life

- Power grid



- Electricity
 - Electricity storage
 - Substation
 - Phase-shift transformer
 - High-voltage line
 - Electricity synchronisation
- Existing power grid

https://ec.europa.eu/energy/infrastructure/transparency_platform/map-viewer/main.html

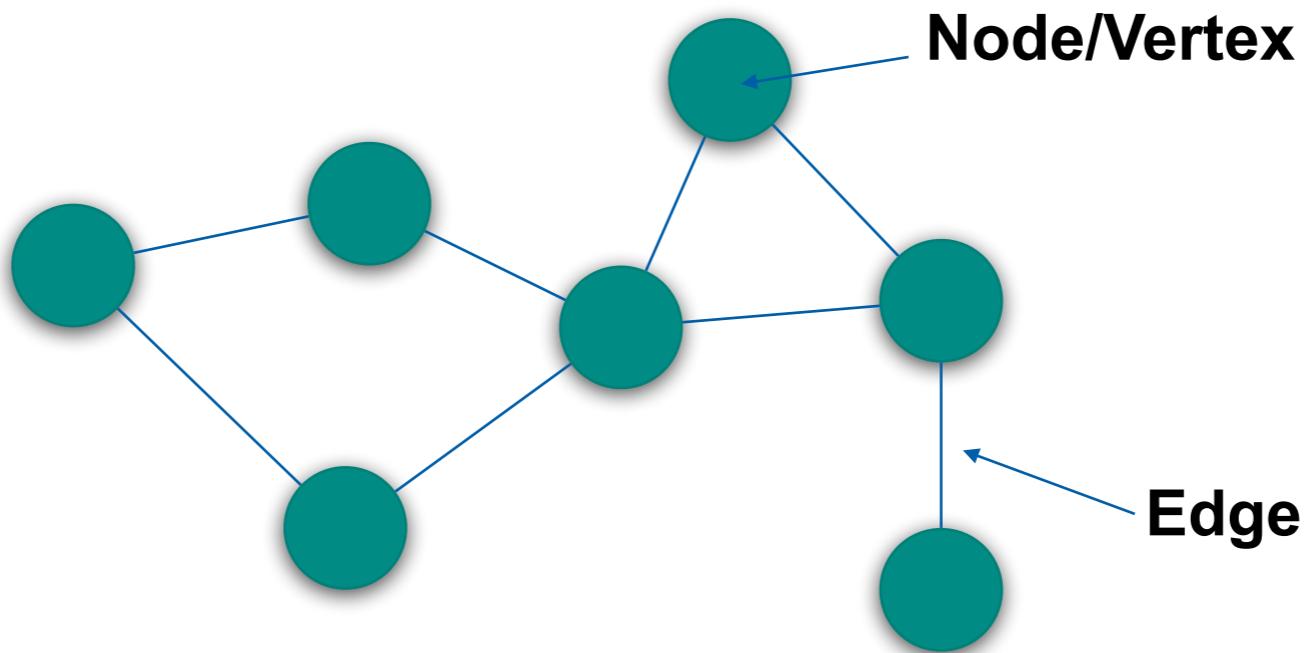
Networks from our daily life

- Facebook global connection map of 10 million users



<https://i.redd.it/5suvt9dpbtk11.png>

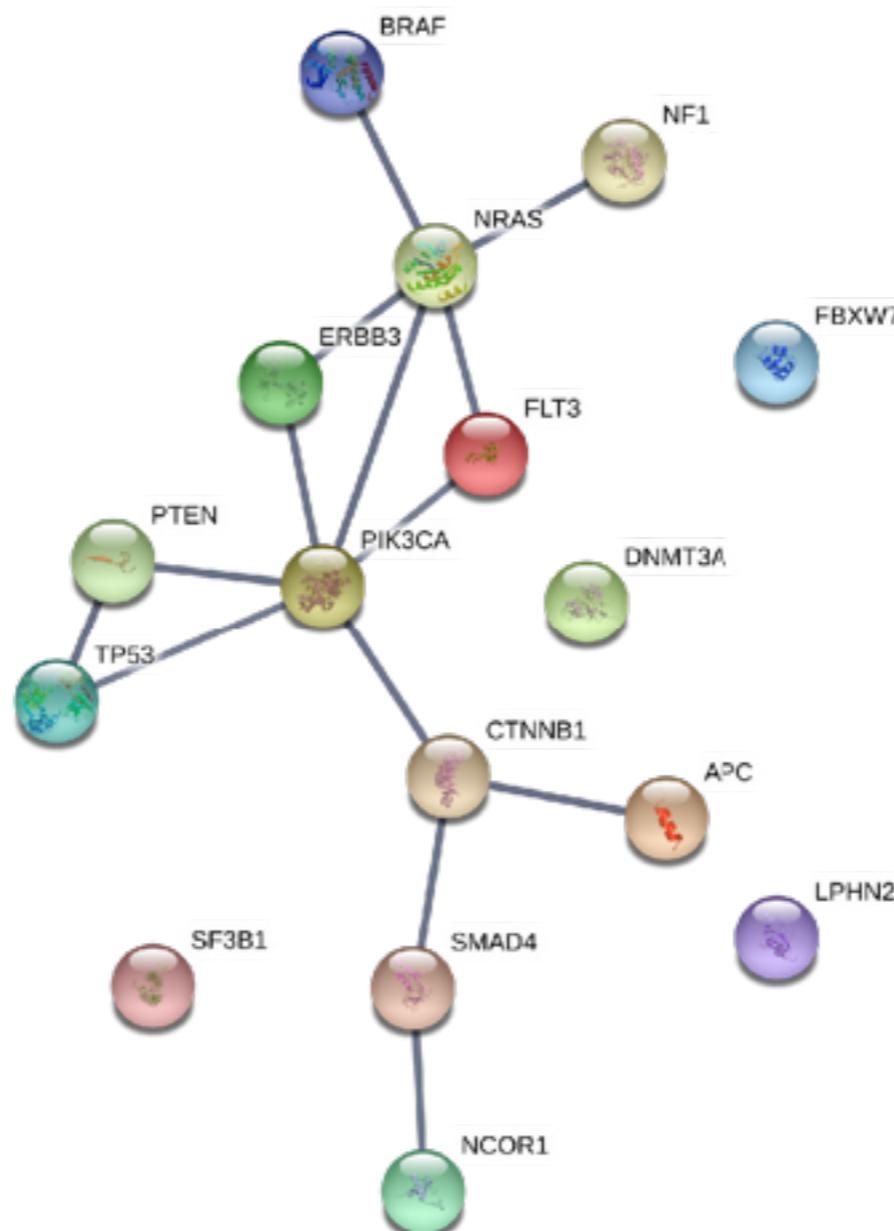
Network = Graphs



- A **graph** $G = (V, E)$ consists of a set V of nodes and a set E of edges
 - $V = \{V_1, \dots, V_n\}$
 - $E = \{(V_i, V_j), \dots, (V_k, V_l)\}$
- **Graph classes:** undirected vs directed vs weighted

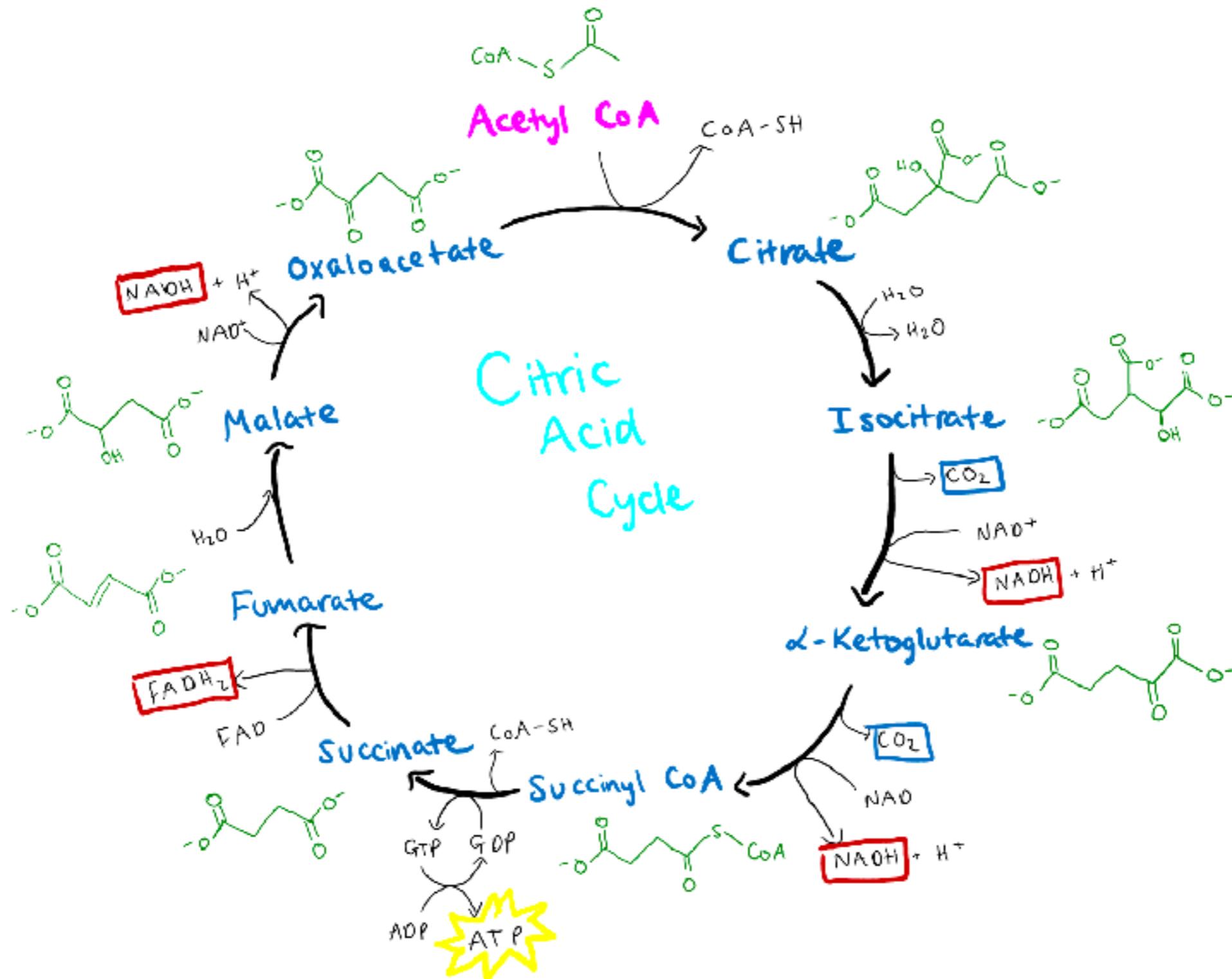
Undirected Graphs

- Protein interaction networks are undirected graphs



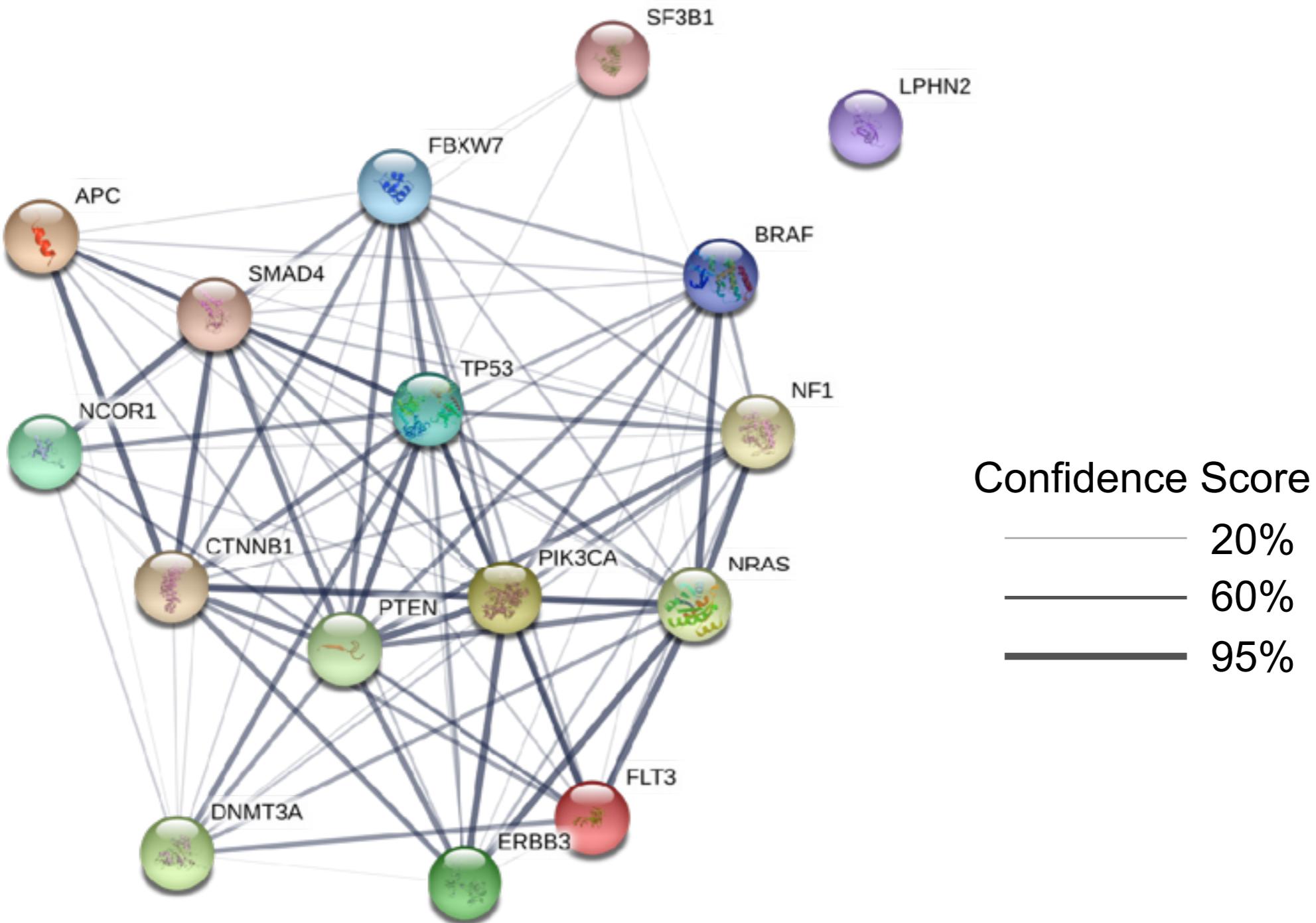
Directed Graphs

- Metabolic pathways are directed graphs of metabolites



Weighted Graphs

- Weighted graphs have a numerical value attached to each edge
- e.g. interaction confidence score, similarity between nodes, ...

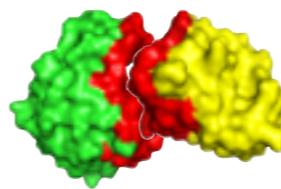


Molecular Interaction Networks in Biology

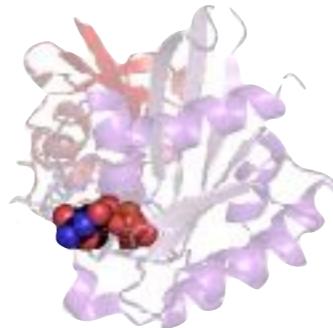
Molecules in Networks

- Protein interaction networks

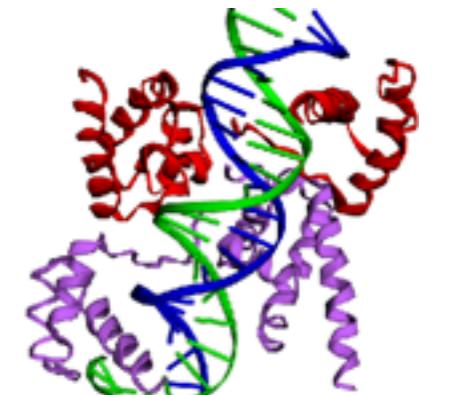
- Protein-Protein interaction networks



- Protein-Small molecule interaction network



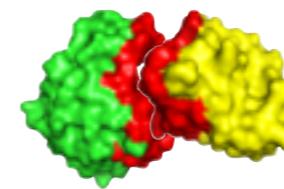
- Protein-DNA interaction network (Gene regulatory networks)



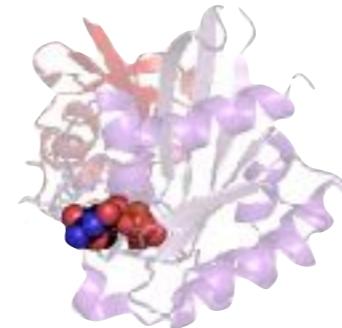
https://www.ebi.ac.uk/thornton-srv/databases/cgi-bin/enzymes/GetPage.pl?ec_number=2.7.11.1
https://en.wikipedia.org/wiki/Transcription_factor
<https://en.wikipedia.org/wiki/MicroRNA>
https://en.wikipedia.org/wiki/Synthetic_lethality

Molecules in Networks

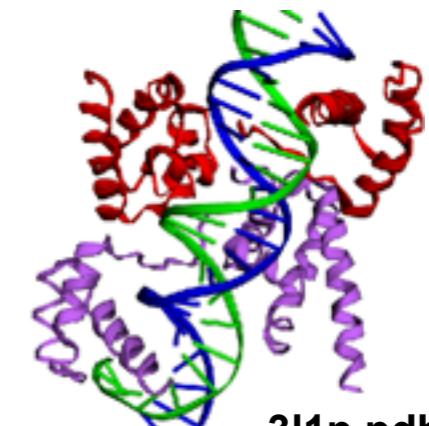
- Protein interaction networks



- Protein-Protein interaction networks



- Protein-Small molecule interaction network



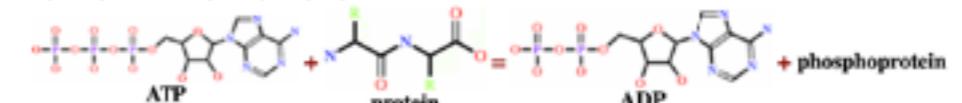
3I1p.pdb

- Metabolic networks/pathways

- Enzyme - Metabolite reactions

V = metabolites, E = enzyme reactions

Reaction: $ATP + \text{a protein} \rightleftharpoons ADP + \text{a phosphoprotein}$.

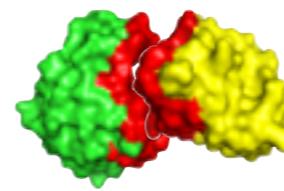


Molecule diagrams generated from .mol files obtained from the KEGG database.

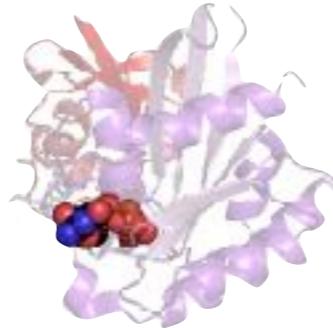
https://www.ebi.ac.uk/thornton-srv/databases/cgi-bin/enzymes/GetPage.pl?ec_number=2.7.11.1
https://en.wikipedia.org/wiki/Transcription_factor
<https://en.wikipedia.org/wiki/MicroRNA>
https://en.wikipedia.org/wiki/Synthetic_lethality

Molecules in Networks

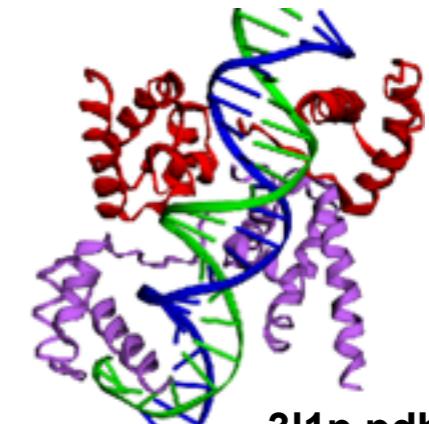
- Protein interaction networks



- Protein-Protein interaction networks



- Protein-Small molecule interaction network



3I1p.pdb

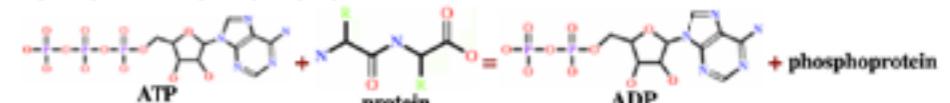
- Protein-DNA interaction network (Gene regulatory networks)

- Metabolic networks/pathways

- Enzyme - Metabolite reactions

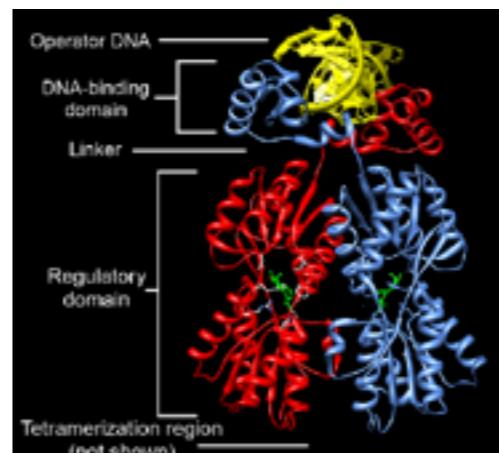
V = metabolites, E = enzyme reactions

Reaction: $ATP + \text{a protein} \rightarrow ADP + \text{a phosphoprotein}$.



- Gene regulatory networks

- Transcription factor - DNA interaction



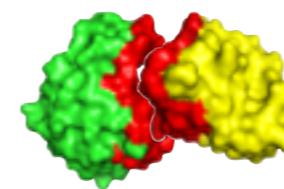
- miRNA - mRNA interactions



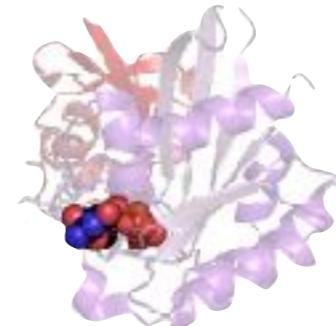
https://www.ebi.ac.uk/thornton-srv/databases/cgi-bin/enzymes/GetPage.pl?ec_number=2.7.11.1
https://en.wikipedia.org/wiki/Transcription_factor
<https://en.wikipedia.org/wiki/MicroRNA>
https://en.wikipedia.org/wiki/Synthetic_lethality

Molecules in Networks

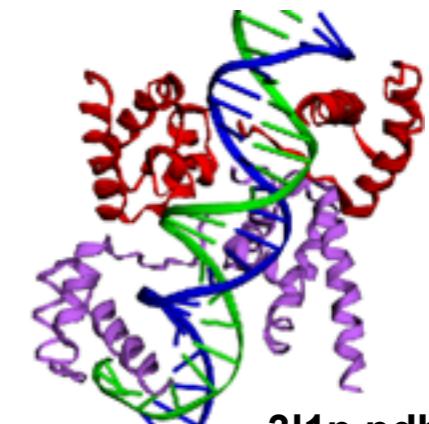
- Protein interaction networks



- Protein-Protein interaction networks



- Protein-Small molecule interaction network



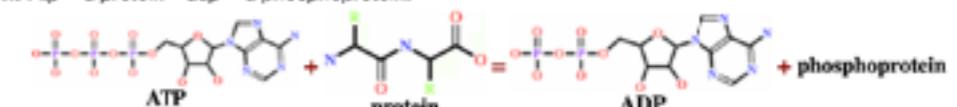
- Protein-DNA interaction network (Gene regulatory networks)

- Metabolic networks/pathways

- Enzyme - Metabolite reactions

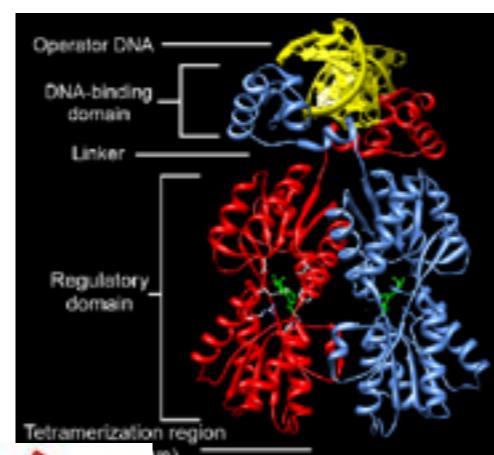
V = metabolites, E = enzyme reactions

Reaction: $\text{ATP} + \text{a protein} \rightarrow \text{ADP} + \text{a phosphoprotein}$.



Molecule diagrams generated from .mol files obtained from the KEGG database.

- Gene regulatory networks

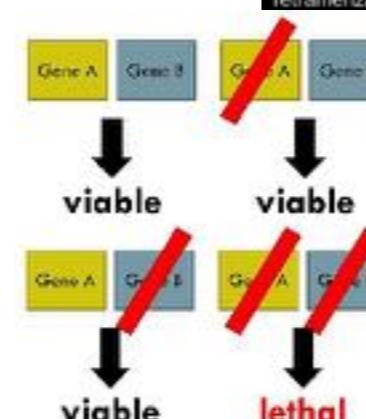


- Transcription factor - DNA interaction

- miRNA - mRNA interactions



- Genetic interaction networks

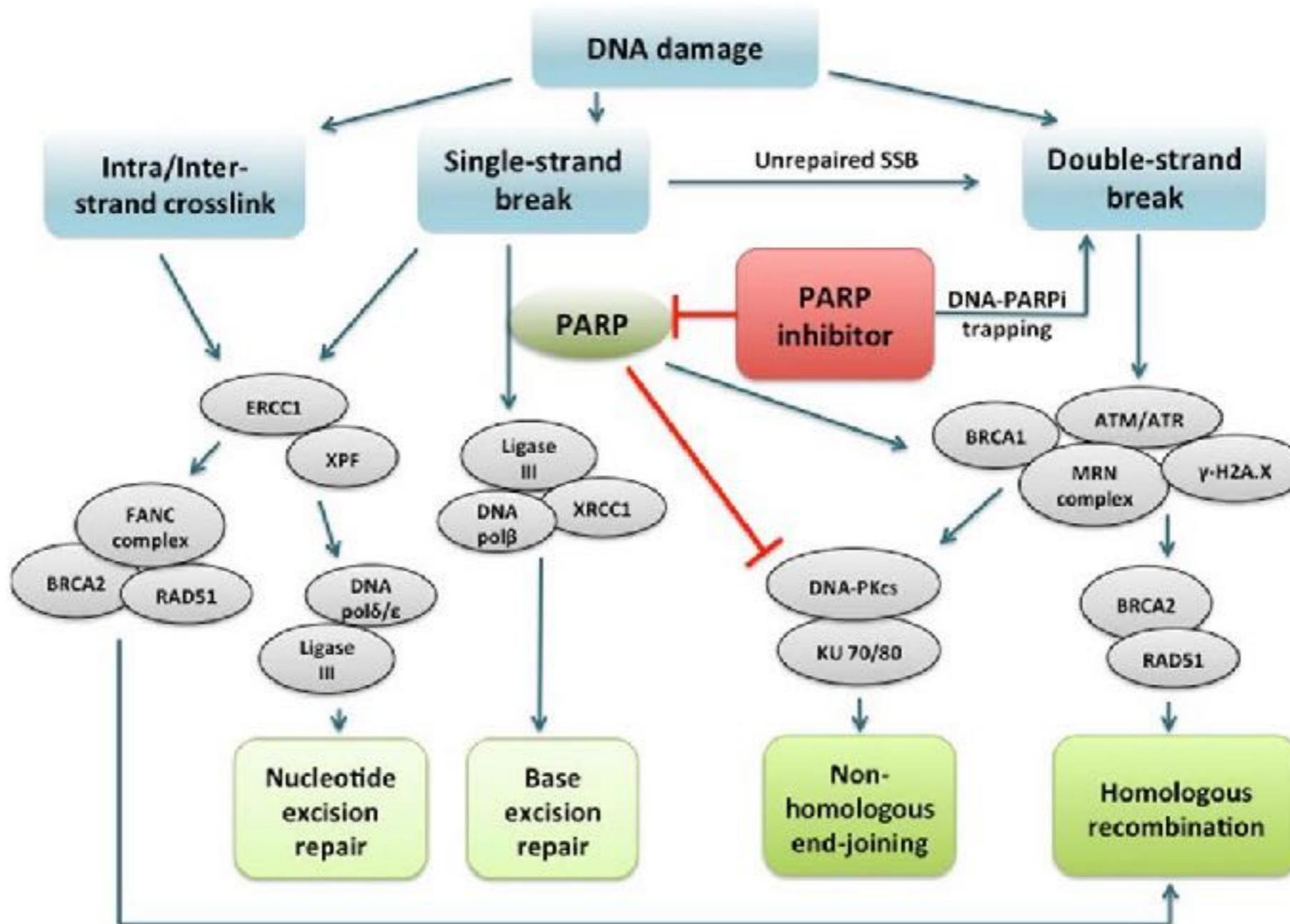


- Gene - Gene interactions

https://www.ebi.ac.uk/thornton-srv/databases/cgi-bin/enzymes/GetPage.pl?ec_number=2.7.11.1
https://en.wikipedia.org/wiki/Transcription_factor
<https://en.wikipedia.org/wiki/MicroRNA>
https://en.wikipedia.org/wiki/Synthetic_lethality

Molecules in Networks

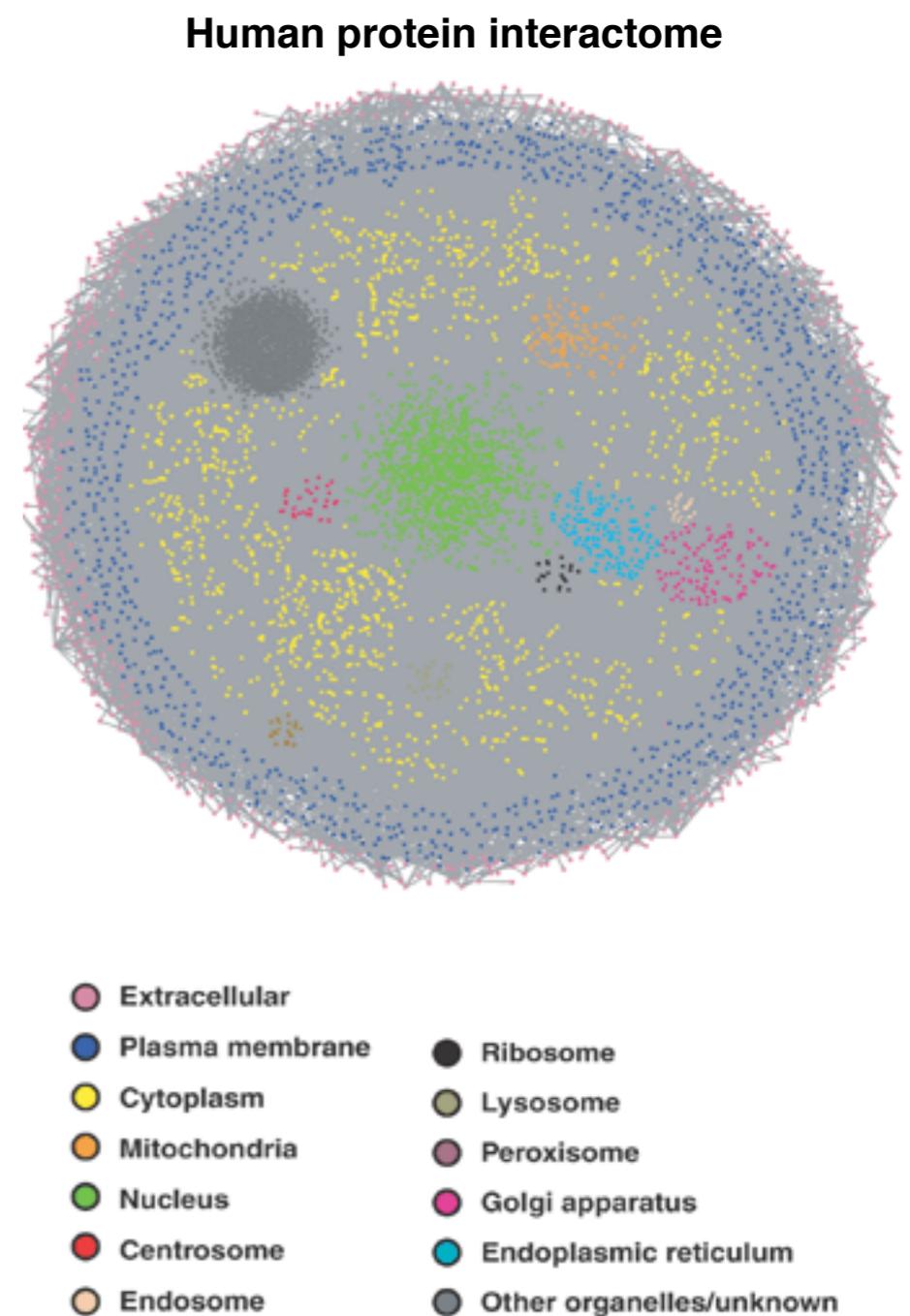
- PARP inhibitors are a new class of drugs for BRCA mutated cancer



O'Sullivan, C. C. et al (2014). *Frontiers in Oncology*, 4, 42

Network Scale

- OMICs networks via high-throughput technologies
 - Proteomics, Genomics, Transcriptomics
 - Mass-spectrometry, Next-Generation Sequencing, Microarray



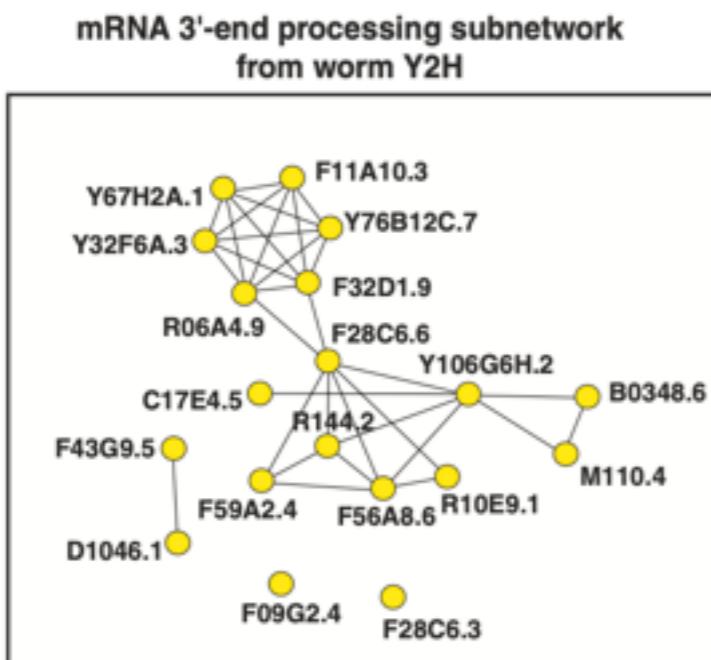
Network Scale

- OMICs networks via high-throughput technologies

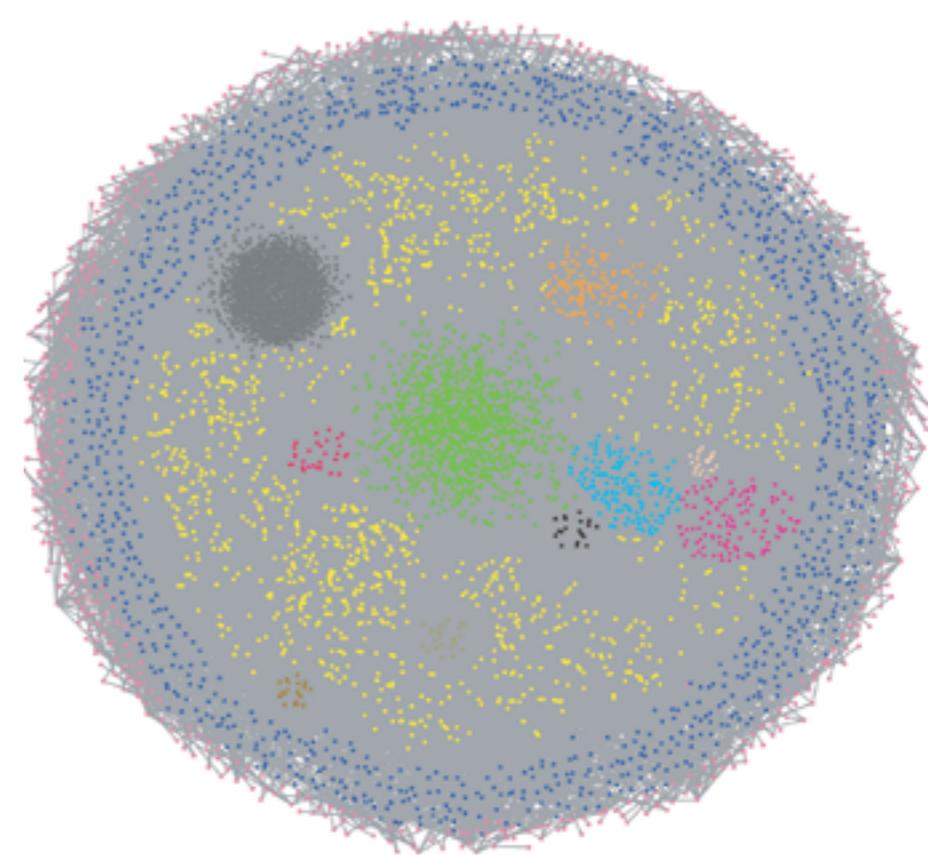
- Proteomics, Genomics, Transcriptomics

- Mass-spectrometry, Next-Generation Sequencing, Microarray

- Small-scale networks



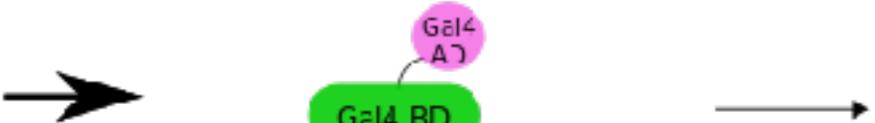
Human protein interactome



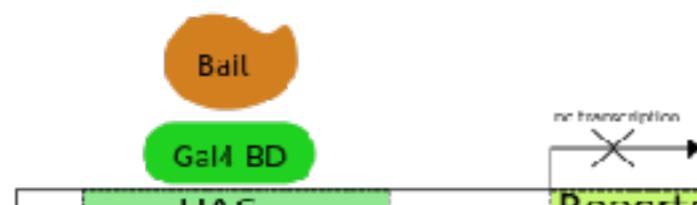
●	Extracellular	●	Ribosome
●	Plasma membrane	●	Lysosome
●	Cytoplasm	●	Peroxisome
●	Mitochondria	●	Golgi apparatus
●	Nucleus	●	Endoplasmic reticulum
●	Centrosome	●	Endosome
●	Endosome	●	Other organelles/unknown

High-Throughput Methods to Probe Interaction Networks

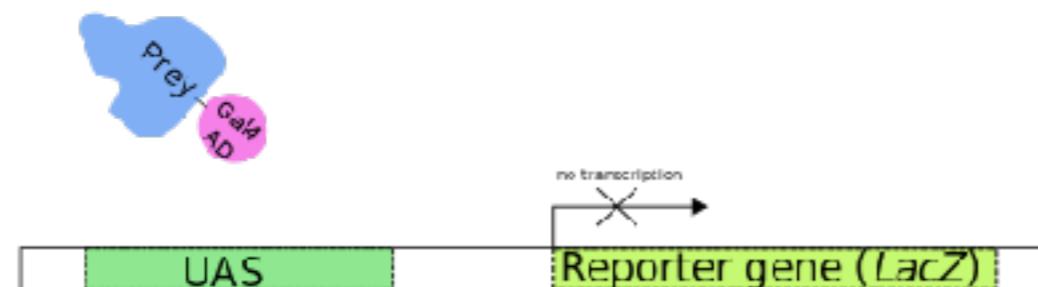
Binary interactions via yeast-two-hybrid screens (Y2H)



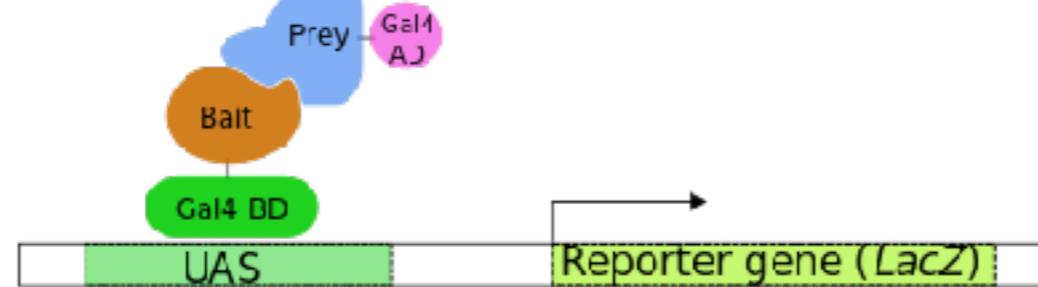
A. Regular transcription of the reporter gene



B. One fusion protein only (Gal4-BD + Bait) - no transcription



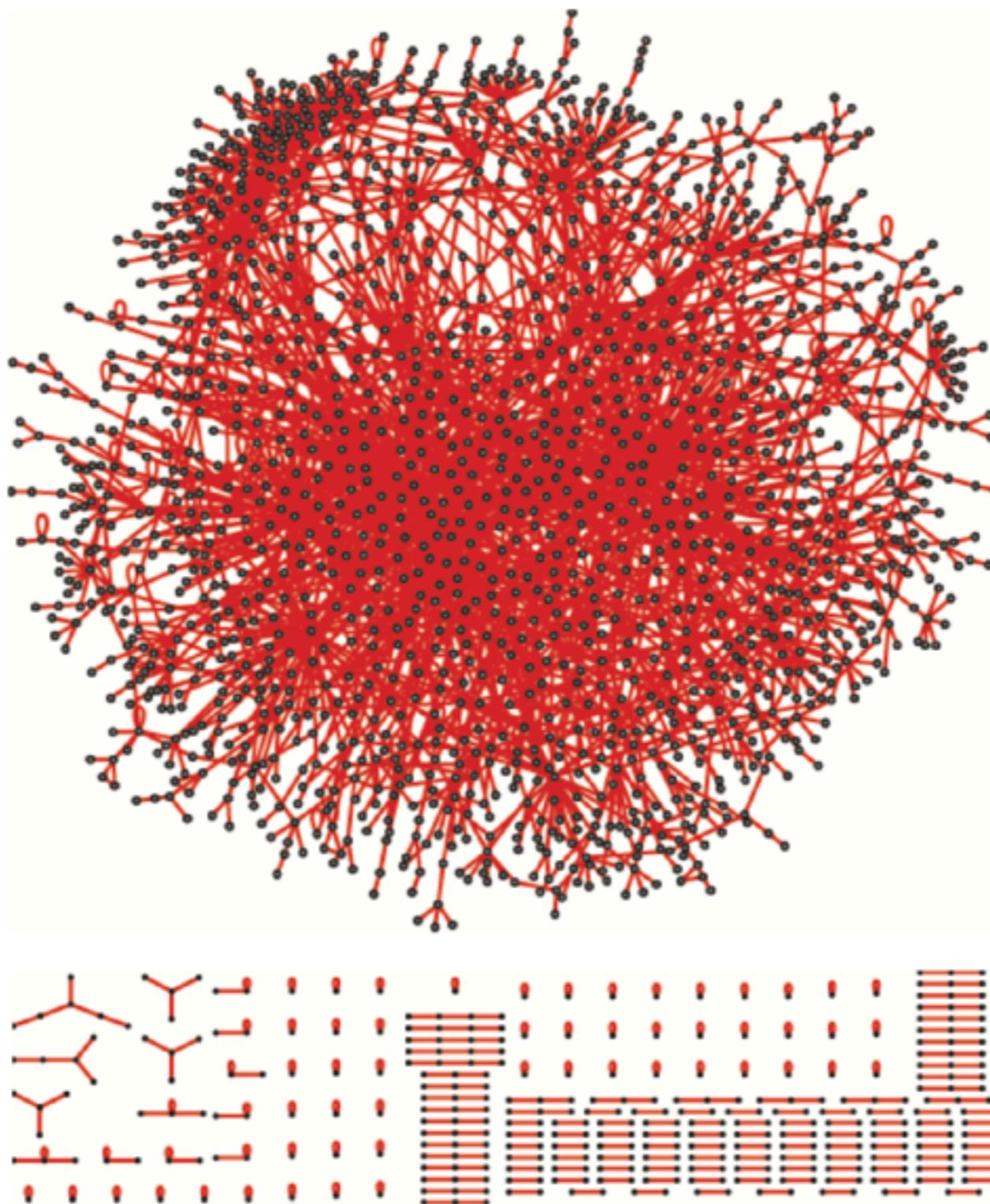
C. One fusion protein only (Gal4-AD + Prey) - no transcription



D. Two fusion proteins with interacting Bait and Prey

GAL4 = Galactose transcription factor
AD = Activation domain
BD = DNA-binding domain
UAS = Upstream Activating Sequence
LacZ = Lactose Operon

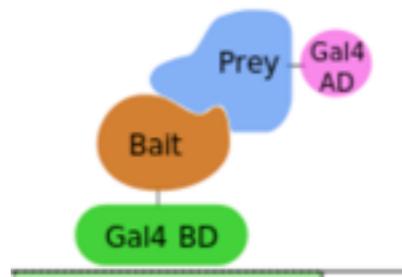
Yeast-two-hybrid screen in yeast



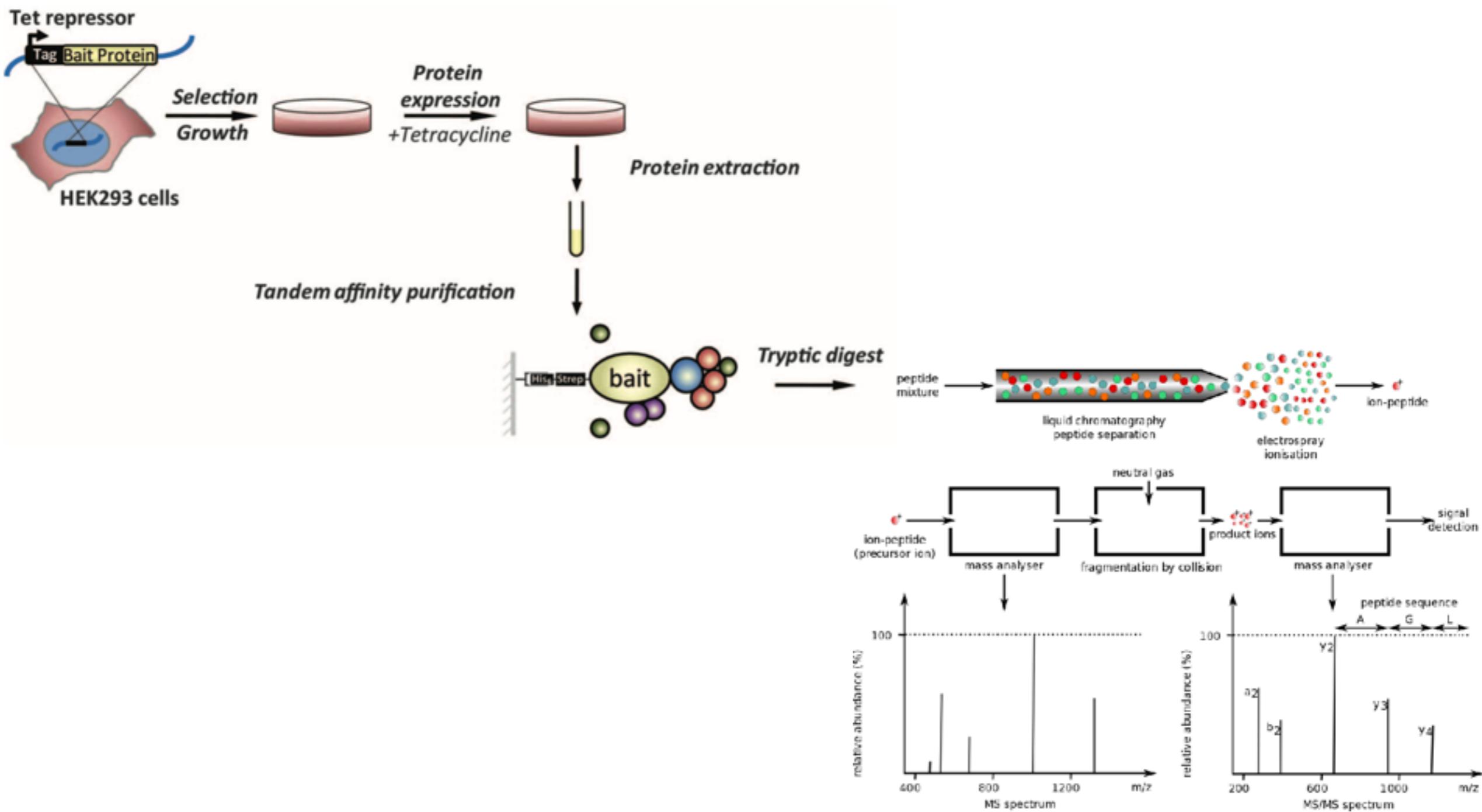
Uetz, P., Giot, L., Cagney, G., Mansfield, T. A., Judson, R. S., Knight, J. R., et al. (2000). A comprehensive analysis of protein-protein interactions in *Saccharomyces cerevisiae*. *Nature* Yu, H., Braun, P., Yildirim, M. A., Lemmens, I., Venkatesan, K., Sahalie, J., et al. (2008). High-quality binary protein interaction map of the yeast interactome network. *Science*

Limitations of Yeast-two-hybrid screen

- Noisy data i.e. many False-Positive (FP)
- Takes place in nucleus! Membrane proteins?
- Interactions driven by N-terminus of test protein complex ?
- Proteins lack chaperones in yeast for folding
- Transient interactions: enzymatic or signalling interactions?
- Completeness?

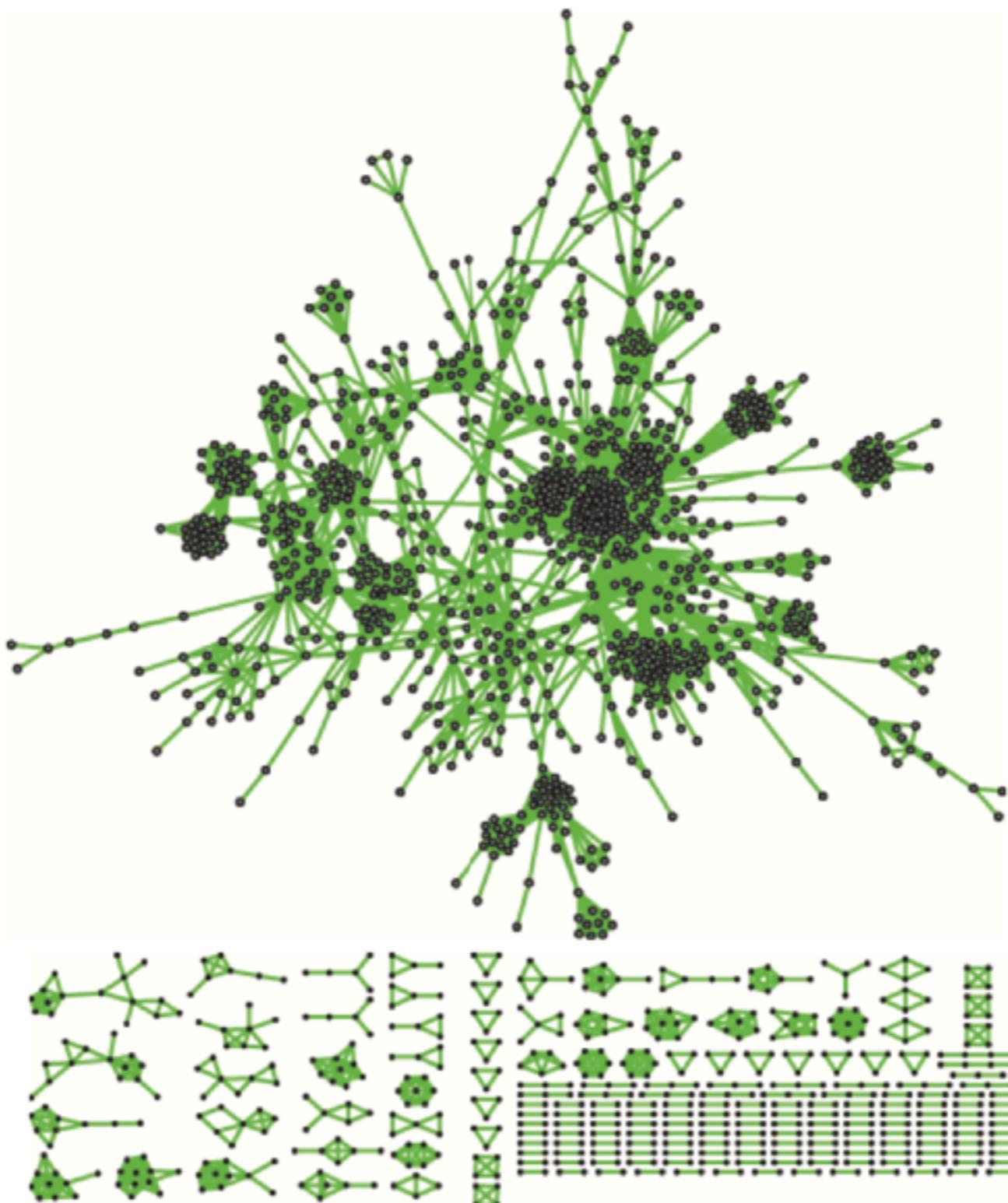


Co-membership in Protein-Complexes via Affinity-Purification coupled to Mass-Spectrometry (APMS)



Herzog, F., Kahraman, A., Boehringer, D., Mak, R., Bracher, A., Walzhoeni, T., et al. (2012). *Science* https://en.wikipedia.org/wiki/Protein_mass_spectrometry

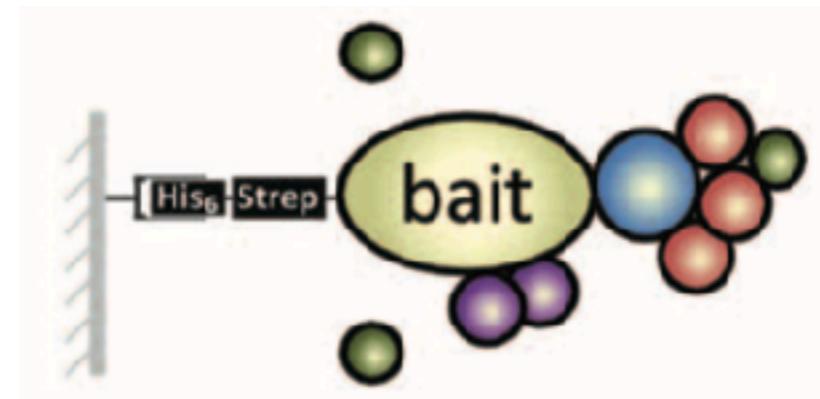
APMS in yeast



Gavin, A.-C., Aloy, P., Grandi, P., Krause, R., Boesche, M., Marzioch, M., et al. (2006). Proteome survey reveals modularity of the yeast cell machinery. *Nature*.
Yu, H., Braun, P., Yildirim, M. A., Lemmens, I., Venkatesan, K., Sahalie, J., et al. (2008). High-quality binary protein interaction map of the yeast interactome network. *Science*.

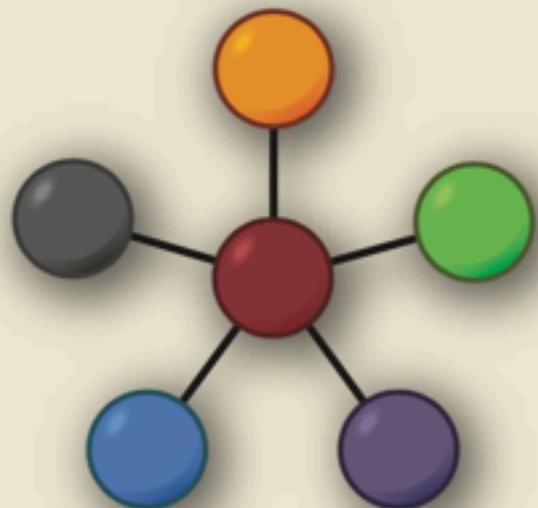
Limitations of APMS

- Highly abundant proteins hard to wash away
- Transient/weak interactors can be washed away
- Proteins nonspecifically interacting with tag protein or beads
- Components of the protein synthesis machinery binding to immature forms of prey proteins
- Physical interactions?
- Completeness?

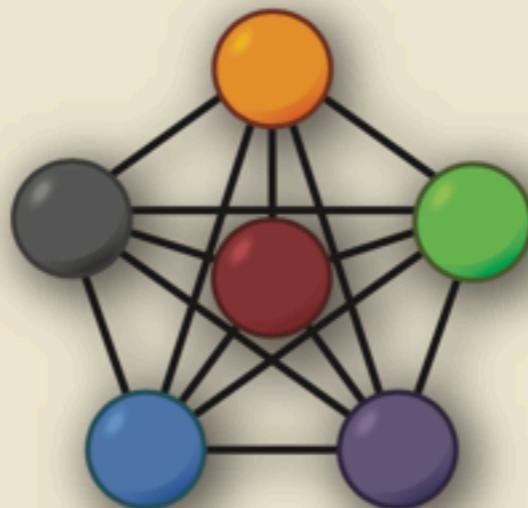


Graph Classes for Modelling APMS Complexes

Spoke model



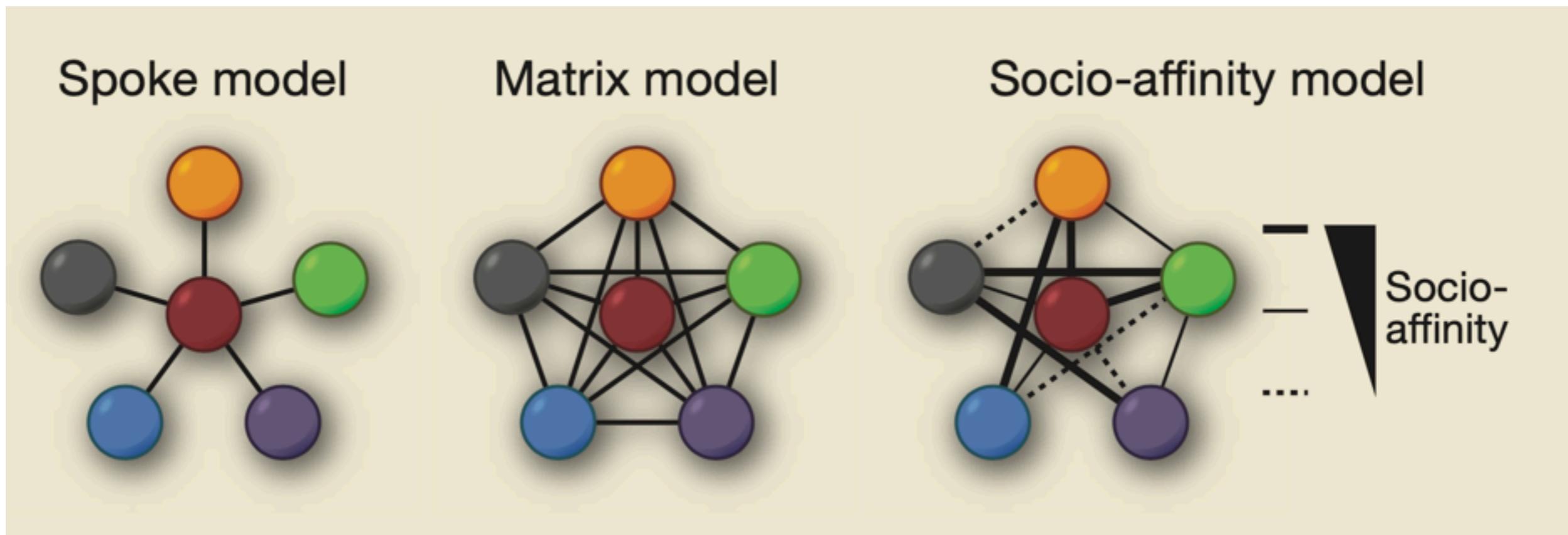
Matrix model



Socio-affinity model

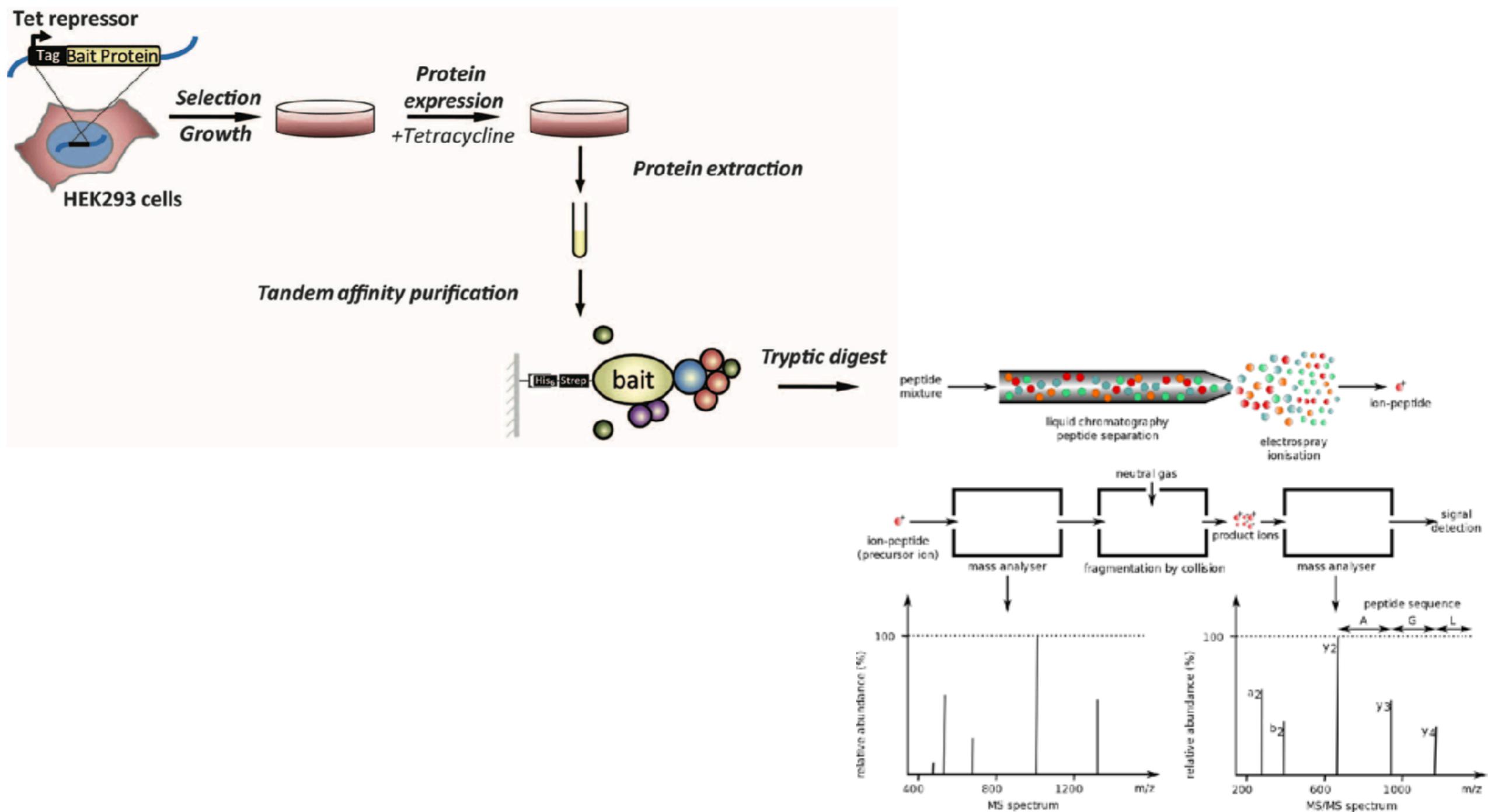


Graph Classes for Modelling APMS Complexes



- **Socio-affinity model**
- **Mixture of Spoke and Matrix model**
 - Frequency of two proteins observed together as bait-prey and prey-prey, relative to what would be expected from their frequency in the data set.

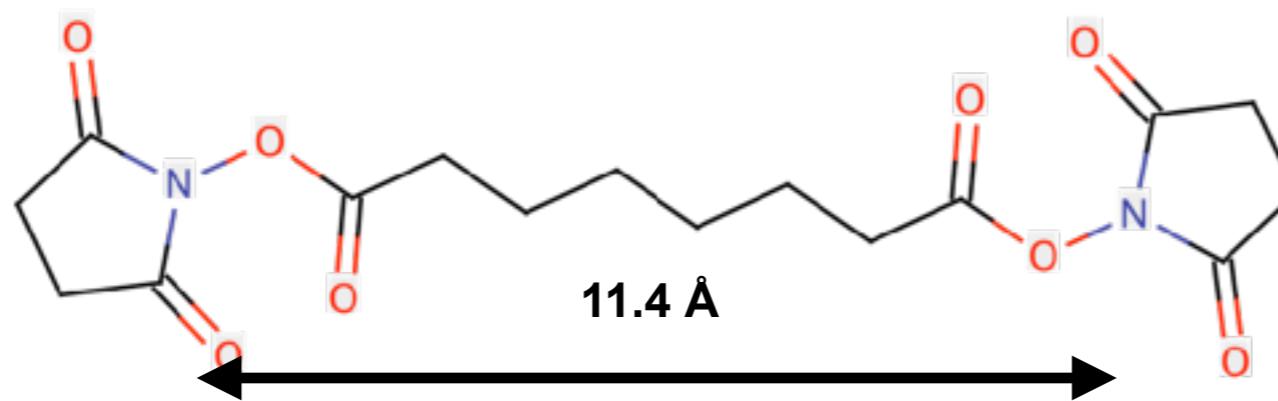
Affinity-Purification coupled to Mass-Spectrometry (APMS)



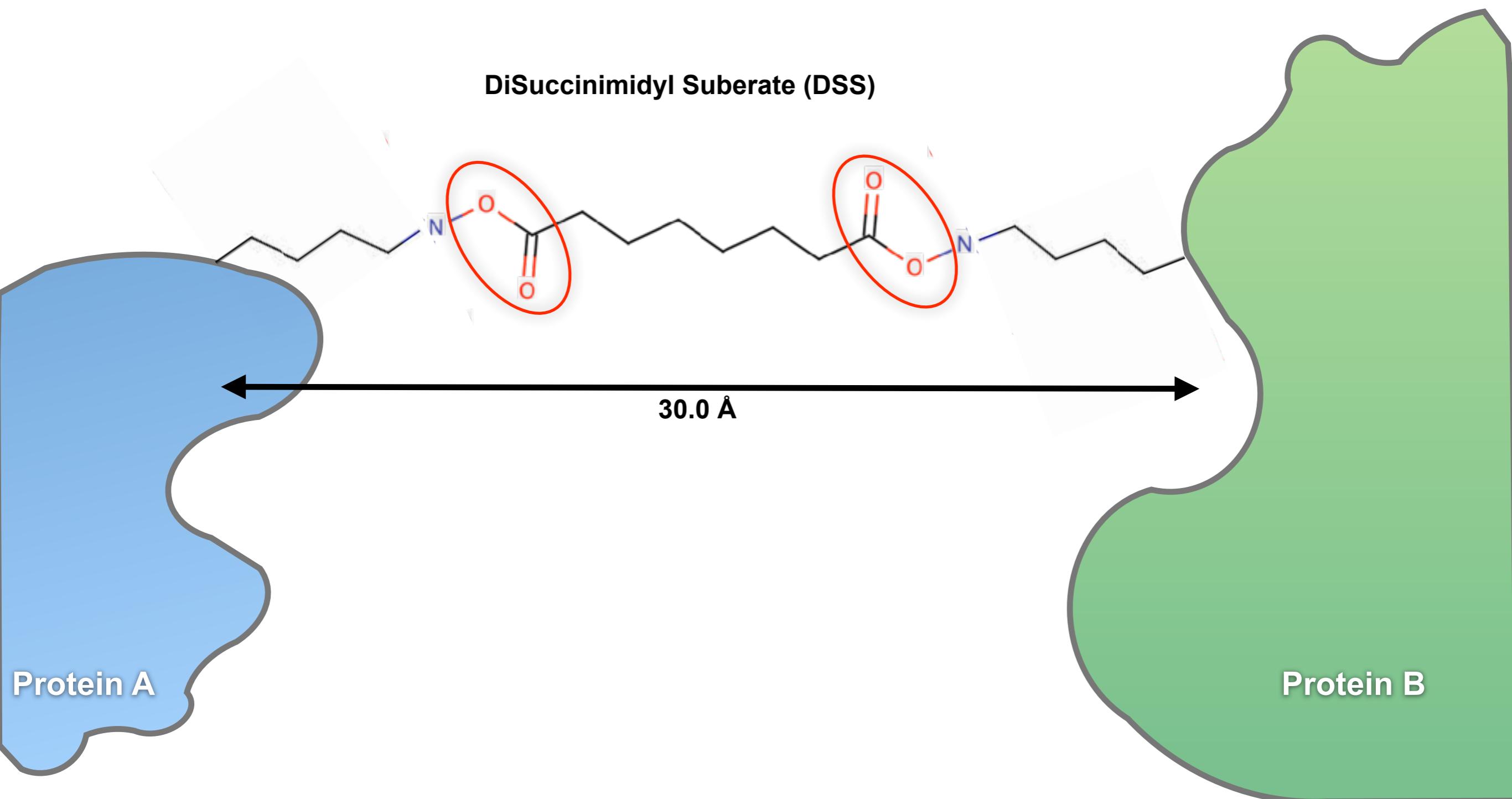
Herzog, F., Kahraman, A., Boehringer, D., Mak, R., Bracher, A., Walzthoeni, T., et al. (2012). *Science* https://en.wikipedia.org/wiki/Protein_mass_spectrometry

Chemical Cross-Linking

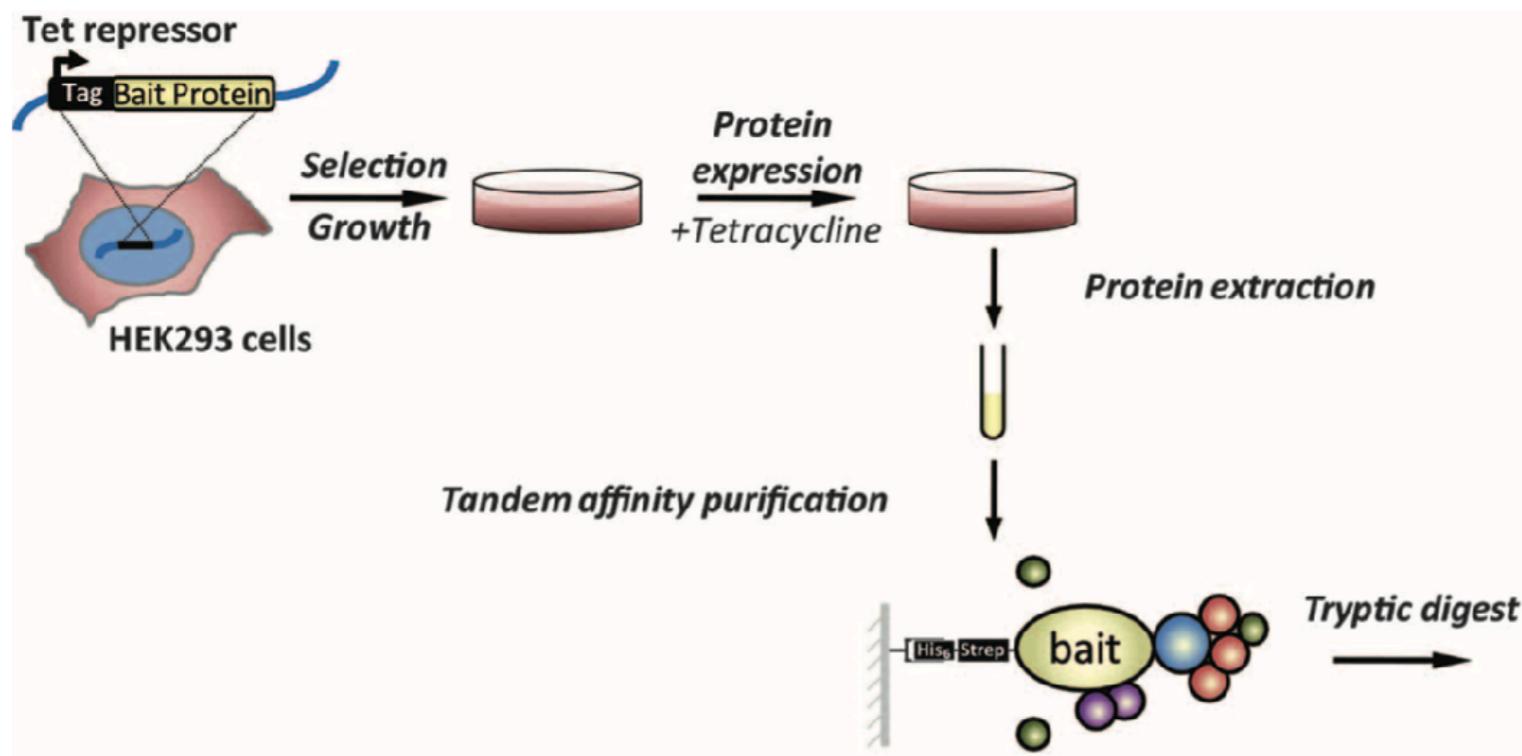
DiSuccinimidyl Suberate (DSS)



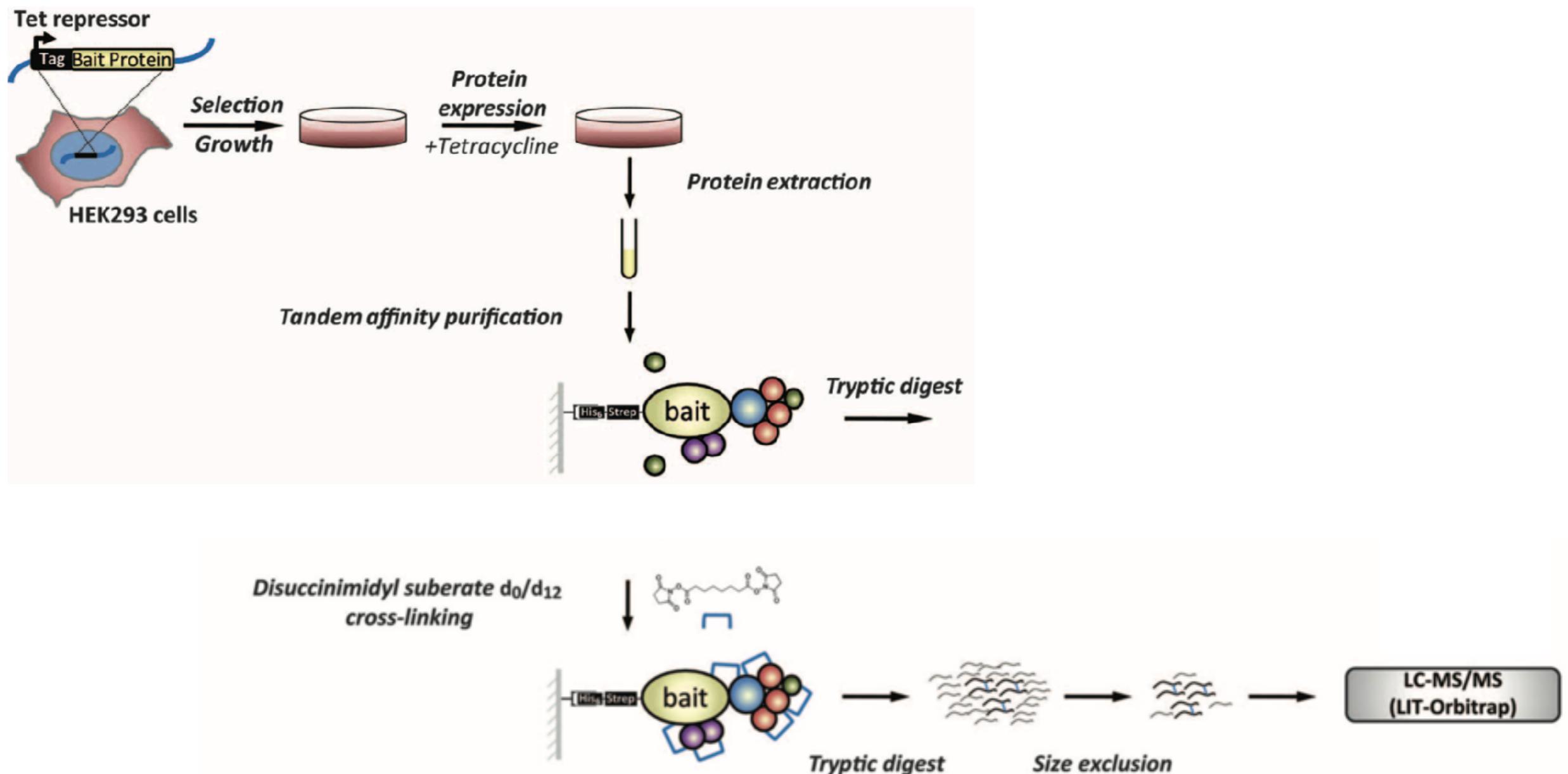
Chemical Cross-Linking



Affinity-Purification coupled to Chemical cross-linking-MS (CX-MS)

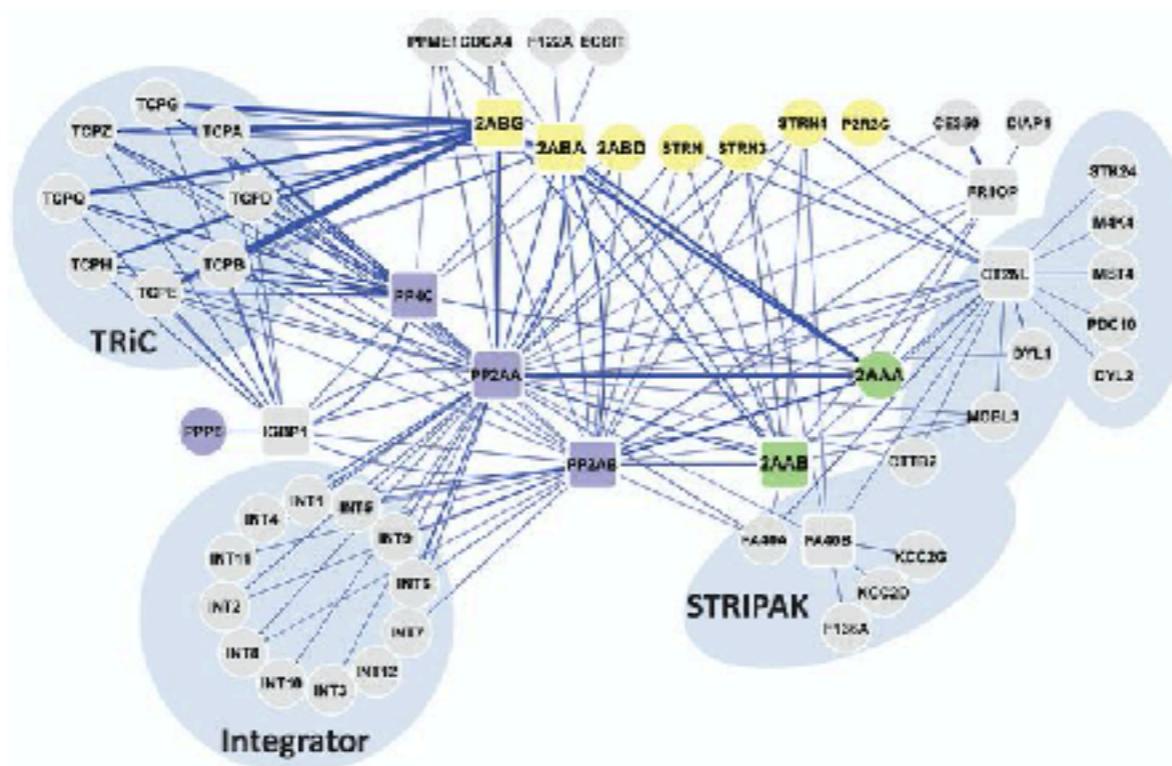


Affinity-Purification coupled to Chemical cross-linking-MS (CX-MS)

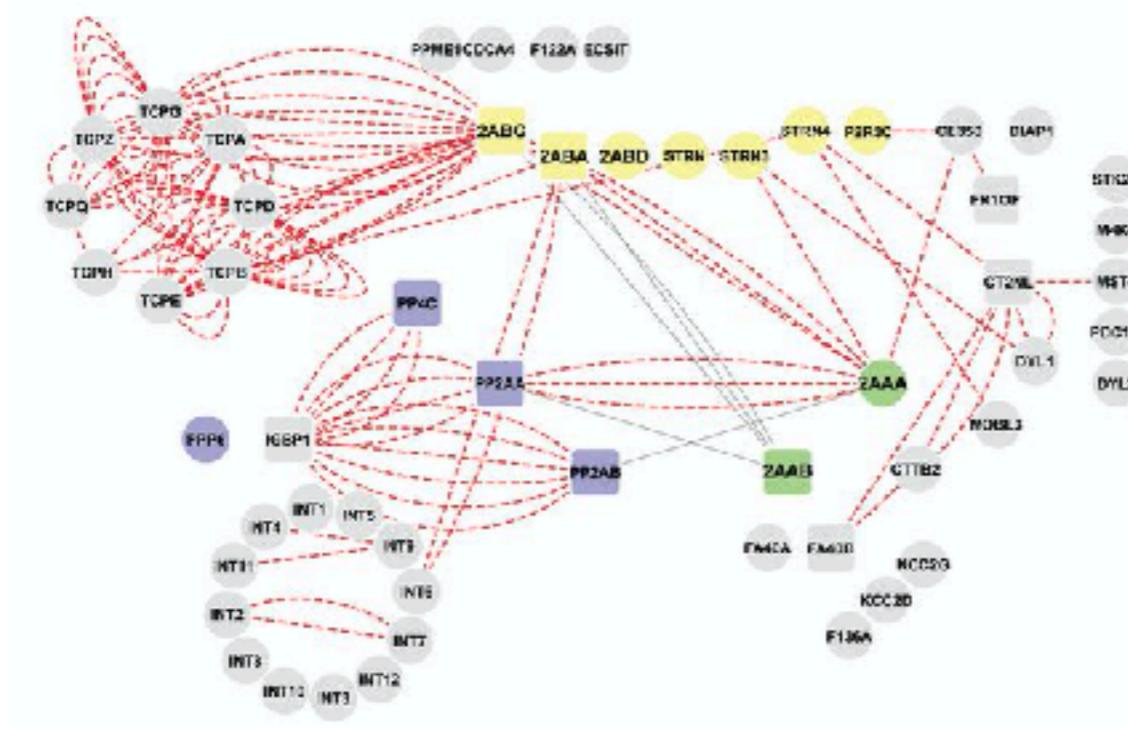


Herzog, F., Kahraman, A., Boehringer, D., Mak, R., Bracher, A., Walzthoeni, T., et al. (2012). *Science*

AP-CX-MS of Protein Phosphatase 2A Complex



APMS based interaction network of PP2A



AP-CX-MS based interaction network of PP2A

Predicting Protein Interaction Networks

Bioinformatic prediction of protein-protein interaction networks

string-db.org

Version: 11.0

LOGIN | REGISTER



Search Download Help My Data

Welcome to STRING

Protein-Protein Interaction Networks

Functional Enrichment Analysis

ORGANISMS | PROTEINS | INTERACTIONS
5090 | 24.6 mio | >2000 mio

SEARCH

© STRING CONSORTIUM 2019



SIB - Swiss Institute of Bioinformatics



CPR - NNF Center for Protein Research



EMBL - European Molecular Biology Laboratory

ABOUT

Content

References

Contributors

Statistics

INFO

Scores

Use scenarios

FAQs

Cookies/Privacy

ACCESS

Versions

APIs

Licensing

Usage

CREDITS

Funding

Data sources

Partners

Software



STRING: Protein Interaction Database

Version: 10.5

LOGIN | REGISTER



Search Download Help My Data

Protein by name >

Protein by sequence >

Multiple proteins >

Multiple sequences >

Organisms >

Protein families ("COGs") >

Examples >

Random entry >

SEARCH

Single Protein by Name / Identifier

Protein Name: (examples: #1 #2 #3)

Organism:

SEARCH

© STRING CONSORTIUM 2018



SIB - Swiss Institute of Bioinformatics



CPR - NNF Center for Protein Research



EMBL - European Molecular Biology Laboratory

ABOUT

Content

References

Contributors

Statistics

INFO

Scores

Use scenarios

FAQs

Cookies/Privacy

ACCESS

Versions

APIs

Licensing

Usage

CREDITS

Funding

Datasources

Partners

Software



STRING is part of the ELIXIR infrastructure: it is one of ELIXIR's Core Data Resources. [Learn more >](#)

STRING: Protein Interaction Database

Version: 10.5

LOGIN | REGISTER



Search

Download

Help

My Data

Protein by name



Protein by sequence



Multiple proteins



Multiple sequences



Organisms



Protein families ("COGs")



Examples



Random entry



SEARCH

Single Protein by Name / Identifier

Protein Name:

(examples: #1 #2 #3)

Organism:

auto-detect

SEARCH

© STRING CONSORTIUM 2018



SIB - Swiss Institute of Bioinformatics



CPR - NNF Center for Protein Research



EMBL - European Molecular Biology Laboratory

ABOUT

Content

References

Contributors

Statistics

INFO

Scores

Use scenarios

FAQs

Cookies/Privacy

ACCESS

Versions

APIs

Licensing

Usage

CREDITS

Funding

Datasources

Partners

Software



STRING is part of the ELIXIR infrastructure: it is one of ELIXIR's Core Data Resources. [Learn more >](#)

string-db.org/cgi/network.pl?taskId=dtyE9zvA7xKy

BRCA1 protein (Homo sapiens) - STRING network view

Version: 10.0

LOGIN | REGISTER

STRING

Search Download Help My Data

← Your query returned:
"BRCA1" - breast cancer 1, early onset in
Homo sapiens

Legend Data Settings View Settings Tables / Exports Evidence Analysis

Nodes:

- Network nodes represent proteins

splice isoforms or post-translational modifications are collapsed, i.e. each node represents all the proteins produced by a single, protein-coding gene locus.
- Node Size
 - small nodes: protein of unknown 3D structure
 - large nodes: some 3D structure is known or predicted
- Node Color
 - colored nodes: query proteins and first shell of interactors
 - white nodes: second shell of interactors

Edges:

- Edges represent protein-protein associations

associations are meant to be specific and meaningful, i.e. proteins jointly contribute to a shared function; this does not necessarily mean they are physically binding each other.
- Known Interactions
 - from curated databases
 - experimentally determined
- Predicted Interactions
 - gene neighborhood
 - gene fusions
 - gene co-occurrence
- Others
 - textmining
 - co-expression
 - protein homology

Your Input:

BRCA1: breast cancer 1, early onset; E3 ubiquitin-protein ligase that specifically mediates the formation of 'Lys-6'-linked polyubiquitin chains and plays a central role in DNA repair by facilitating cellular responses to DNA damage. It is unclear whether it also mediates the formation of other types of polyubiquitin chains. The E3 ubiquitin-protein ligase activity is required for its tumor suppressor function. The BRCA1-BARD1 heterodimer coordinates a diverse range of cellular pathways such as DNA damage repair, ubiquitination and transcriptional regulation to maintain genomic stability. *Reg.* 1 (1994) 91.

Version: 10.0

STRING

Actions

- re-center network on this node
- add this node to input nodes

Information

tumor protein p53; Acts as a tumor suppressor in many tumor types; induces growth arrest or apoptosis depending on the physiological circumstances and cell type. Involved in cell cycle regulation as a trans-activator that acts to negatively regulate cell division by controlling a set of genes required for this process. One of the activated genes is an inhibitor of cyclin-dependent kinase. Apoptosis induction seems to be mediated either by stimulation of BAX and FAS antigen expression, or by repression of Bcl-2 expression (By similarity)

Identifier: ENSP00000269305

UniProt

- show protein sequence
- homologs among STRING organisms

1 of 4 homology model (3q06A) identity: 78.4%

Legend

Nodes:

Network nodes represent proteins
splice isoforms or post-translational modifications are collapsed, i.e. each node represents all the proteins produced by a single, protein-coding gene locus.

Node Size

- small nodes: protein of unknown 3D structure
- large nodes: some 3D structure is known or predicted

Node Color

- colored nodes: query proteins and first shell of interactors
- white nodes: second shell of interactors

Edges:

Edges represent protein-protein associations
associations are meant to be specific and meaningful, i.e. proteins jointly contribute to a shared function; this does not necessarily mean they are physically binding each other.

Known Interactions

- from curated databases
- experimentally determined

Predicted Interactions

- gene neighborhood
- gene fusions
- gene co-occurrence

Others

- textmining
- co-expression
- protein homology

Your Input:

BRCA1

breast cancer 1, early onset; E3 ubiquitin-protein ligase that specifically mediates the formation of 'Lys-6'-linked polyubiquitin chains and plays a central role in DNA repair by facilitating cellular responses to DNA damage. It is unclear whether it also mediates the formation of other types of polyubiquitin chains. The E3 ubiquitin-protein ligase activity is required for its tumor suppressor function. The BRCA1-BARD1 heterodimer coordinates a diverse range of cellular pathways such as DNA damage repair, ubiquitination and transcriptional regulation to maintain genomic stability. Reg. 1 (1994 aa).

hood fusion nents sens sing logy?

string-db.org/cgi/network.pl?taskId=dtyE9zvA7xKy

BRCA1 protein (Homo sapiens) - STRING network view

Version: 10.0

LOGIN | REGISTER

STRING

Search Download Help My Data

— Your query returned:
"BRCA1" - breast cancer 1, early onset in
Homo sapiens

Legend > Data Settings > View Settings > Tables / Exports > Evidence > Analysis >

Experiments
Co-purification, co-crystallization, Yeast2Hybrid, Genetic Interactions, etc ... as imported from primary sources.

Databases
Known metabolic pathways, protein complexes, signal transduction pathways, etc ... from curated databases.

Textmining
Automated, unsupervised textmining - searching for proteins that are frequently mentioned together.

Coexpression
Proteins whose genes are observed to be correlated in expression, across a large number of experiments.

Cooccurrence
Gene families whose occurrence patterns across genomes show similarities.

Neighborhood
Groups of genes that are frequently observed in each other's genomic neighborhood.

Fusion
Genes that are sometimes fused into single open reading frames.

STRING allows inspection of the interaction evidence for any given network. Choose any of the viewers above (disabled if not applicable in your network).

bioRxiv preprint doi: https://doi.org/10.1101/2024.01.16.598442; this version posted January 16, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

string-db.org/cgi/network.pl?taskId=dtyE9zvA7xKy

BRCA1 protein (Homo sapiens) - STRING network view

Version: 10.0

LOGIN | REGISTER

STRING

Search Download Help My Data

Your query returned:
"BRCA1" - breast cancer 1, early onset in
Homo sapiens

Legend > Data Settings > View Settings > Tables / Exports > Evidence > Analysis >

Experiments
Co-purification, co-crystallization, Yeast2Hybrid, Genetic Interactions, etc ... as imported from primary sources.

Databases
Known metabolic pathways, protein complexes, signal transduction pathways, etc ... from curated databases.

Textmining
Automated, unsupervised textmining - searching for proteins that are frequently mentioned together.

Coexpression
Proteins whose genes are observed to be correlated in expression, across a large number of experiments.

Cooccurrence
Gene families whose occurrence patterns across genomes show similarities.

Neighborhood
Groups of genes that are frequently observed in each other's genomic neighborhood.

Fusion
Genes that are sometimes fused into single open reading frames.

STRING allows inspection of the interaction evidence for any given network. Choose any of the viewers above (disabled if not applicable in your network).

bioRxiv preprint doi: https://doi.org/10.1101/2023.09.21.552352; this version posted September 21, 2023. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

sources of evidence

STRING: Sources of Evidence



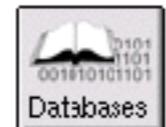
Genomic Neighborhood



Genes/Species Co-occurrence



Gene Fusions



Database Imports



Exp. Interaction Data

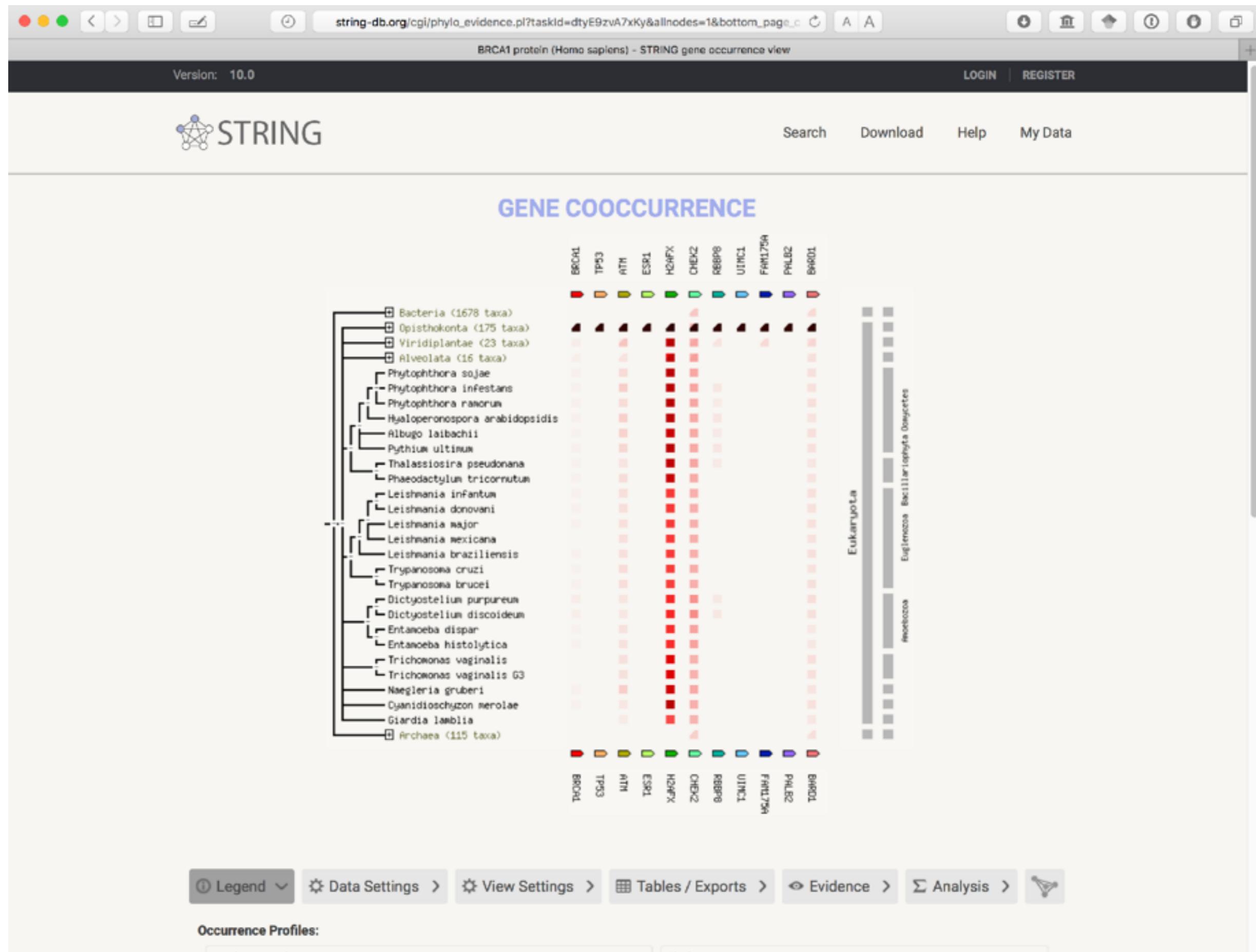


Co-expression

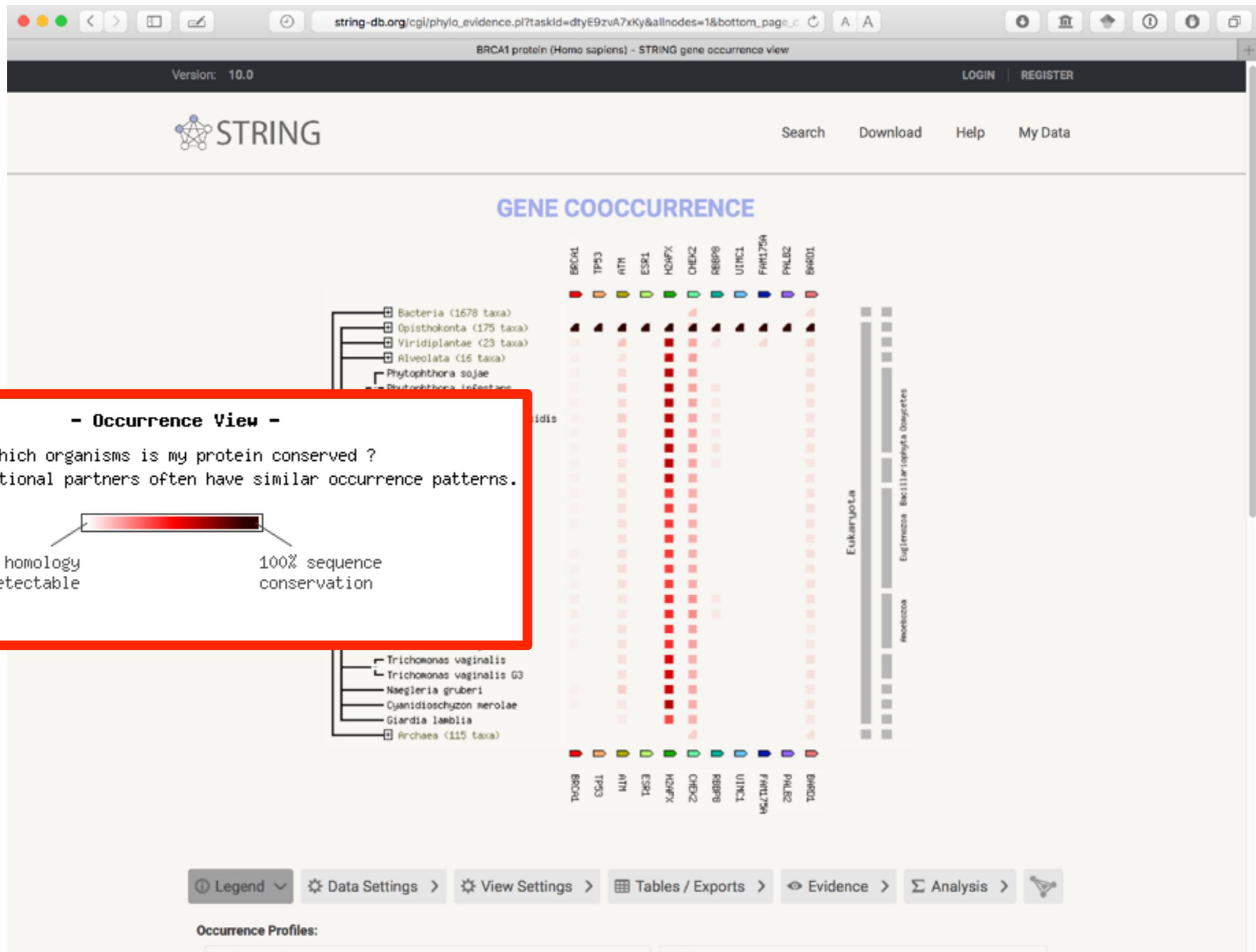


Literature co-occurrence

STRING: Interaction prediction from genome information



STRING: Interaction prediction from genome information



STRING: Interaction prediction from genome information

string-db.org/cgi/neighborhood.pl?taskId=DXJGtAxwaldo&node1=_unassigned&node2=

trpB protein (Escherichia coli K12 MG1655) - STRING chromosome neighborhood view

Version: 10.0

LOGIN | REGISTER

STRING

Search Download Help My Data

GENE NEIGHBORHOOD

Bacteria (1678 taxa)

Eukaryota (238 taxa)

Methanomicrobiales (6 taxa)

Methanococcaceae (8 taxa)

Methanocaldococcaceae (6 taxa)

Thermococcaceae (12 taxa)

Methanothermobacter (2 taxa)

Methanobrevibacter smithii

Methanobrevibacter ruminantium

Methanobacterium (2 taxa)

Methanospaera stadtmanae

Methanothermus fervidus

Archaeoglobaceae (4 taxa)

Thermoplasmatales (4 taxa)

unclassified Euryarchaeota (2 taxa)

Methanopyrus kandleri

Methanocella arvoryzae

Thermoprotei (30 taxa)

Nitrosopumilaceae (3 taxa)

halophilic archaeon DL31

Korarchaeum cryptofilum

Nanoarchaeum equitans

Conserved Neighbourhood

- operons in bacteria
- similar regulation

Legend Data Settings View Settings Tables / Exports Evidence Analysis

Gene Arrangements:

Consecutive Gene Runs

genes are shown in consecutive runs, if they are located in close vicinity on the same chromosome. They are oriented either in the same direction, or in a head-to-head orientation (likely sharing promoters).

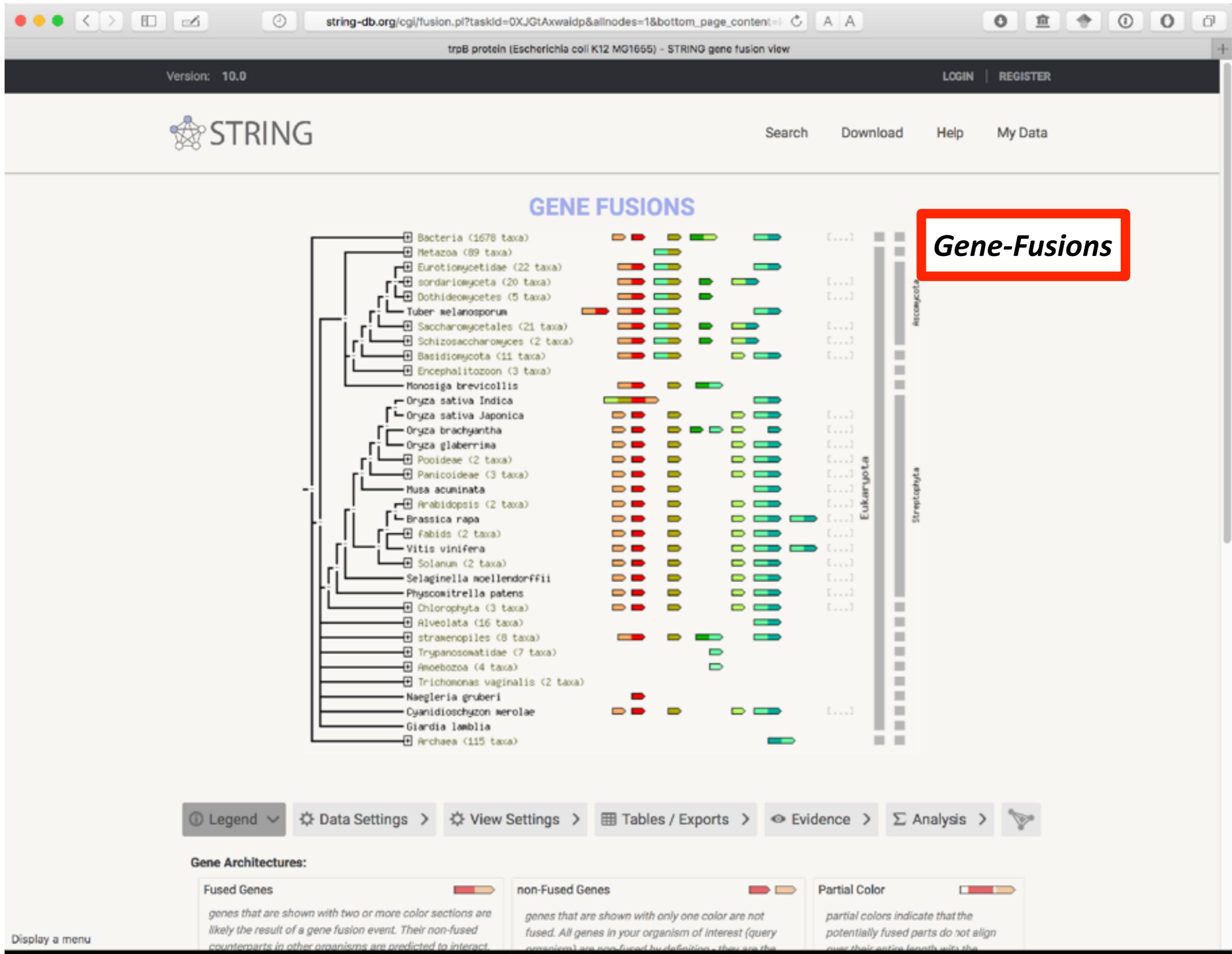
Horizontal Sections

horizontal sections indicate that the orthology relations of the gene are complex. This is either due to gene duplication events (paralogy), or due to technical problems when assigning orthology.

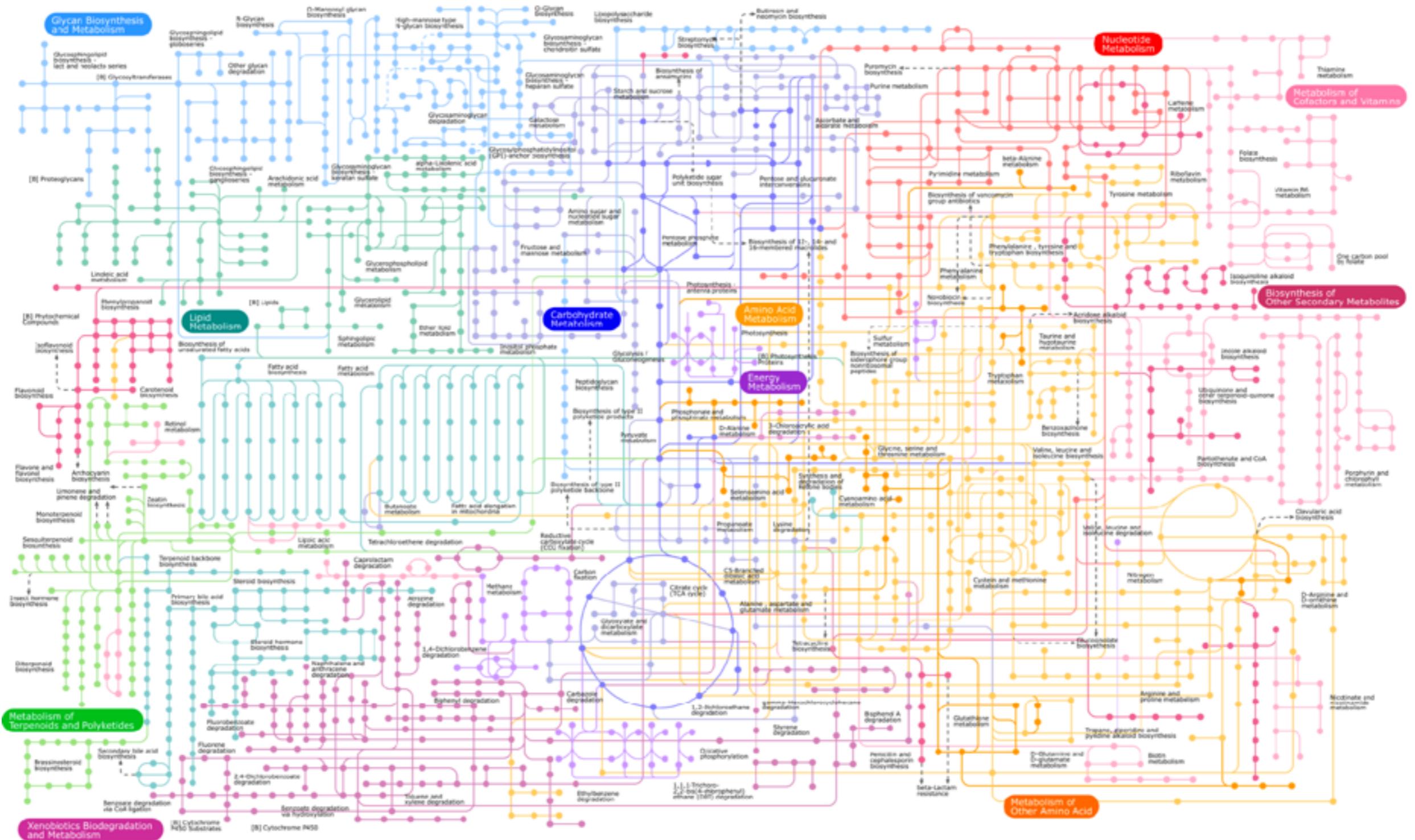
Vertical Sections

vertical sections indicate that different parts of the gene may have different evolutionary histories. This can be a consequence of gene-fusions, or gene-fissions.

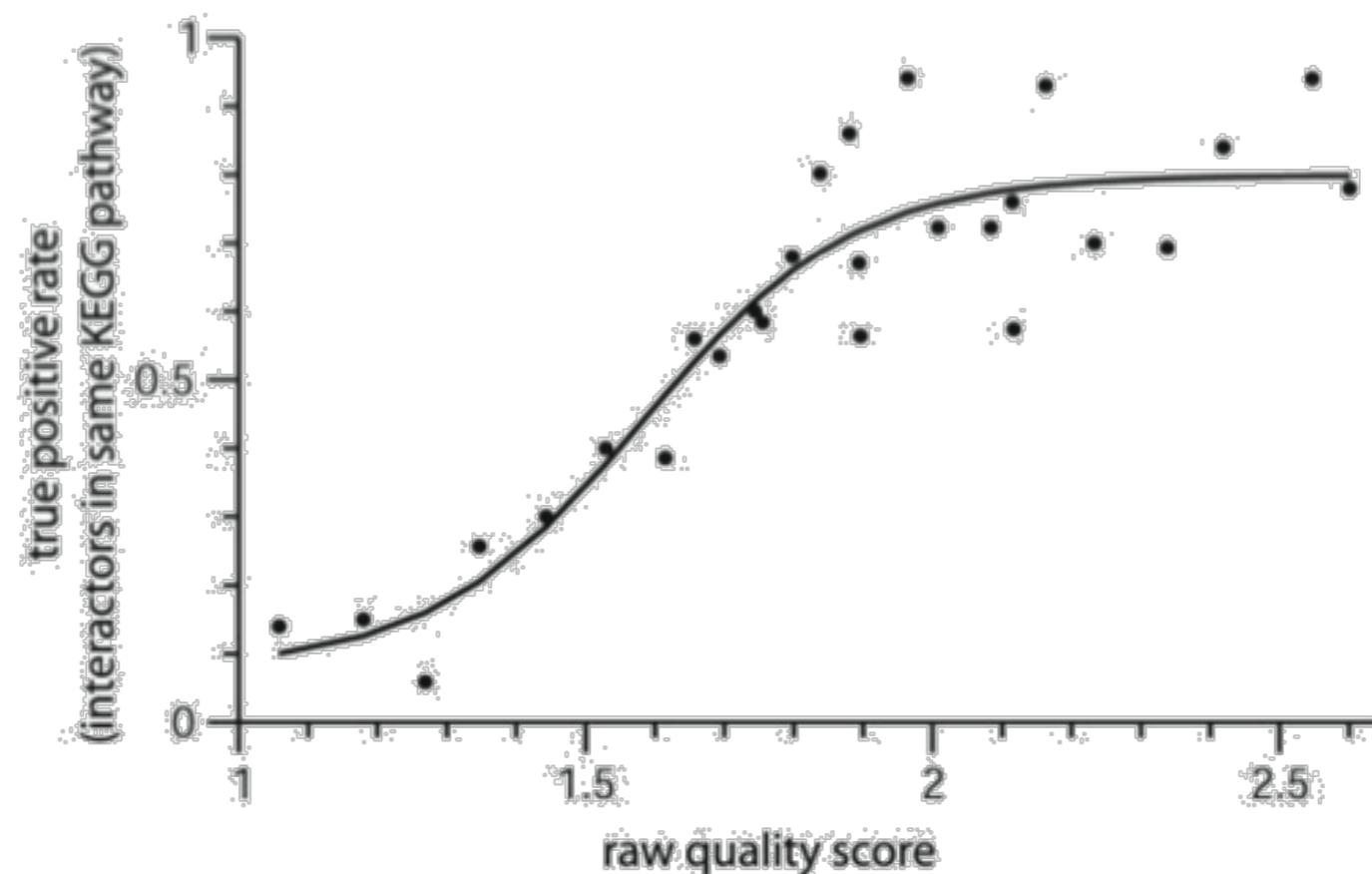
STRING: Interaction prediction from genome information



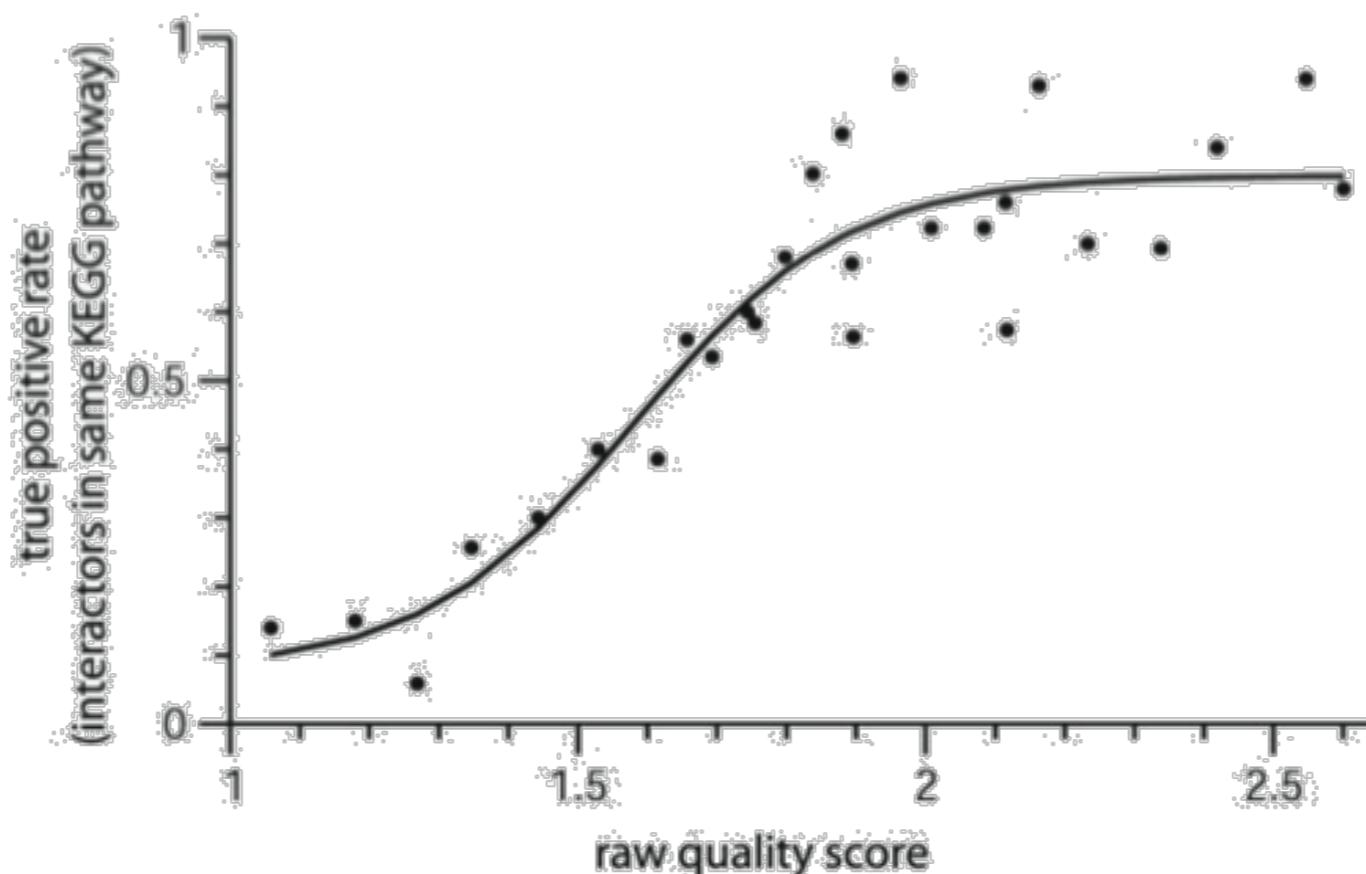
STRING: Scoring Interactions Predictions using KEGG pathways



STRING: Benchmarking



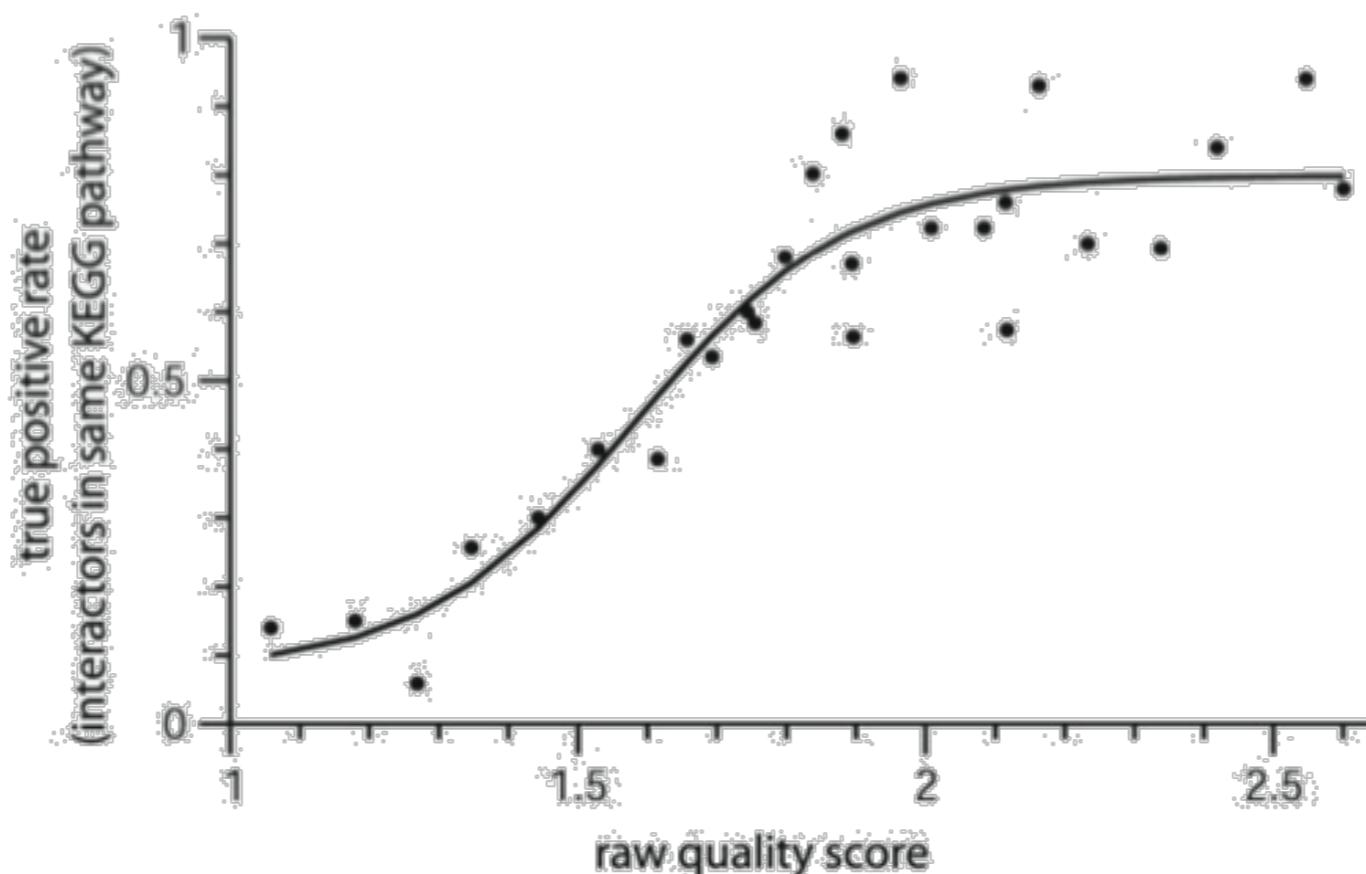
STRING: Benchmarking



combined score = $1 - (1 - n_{score}) * (1 - f_{score}) * (1 - p_{score}) * (1 - c_{score}) * (1 - e_{score}) * (1 - t_{score})$

neighborhood *fusion* *cooccurrence* *coexpression* *experimental* *textmining*

STRING: Benchmarking



$$\text{combined score} = 1 - (1 - n_{\text{score}}) * (1 - f_{\text{score}}) * (1 - p_{\text{score}}) * (1 - c_{\text{score}}) * (1 - e_{\text{score}}) * (1 - t_{\text{score}})$$

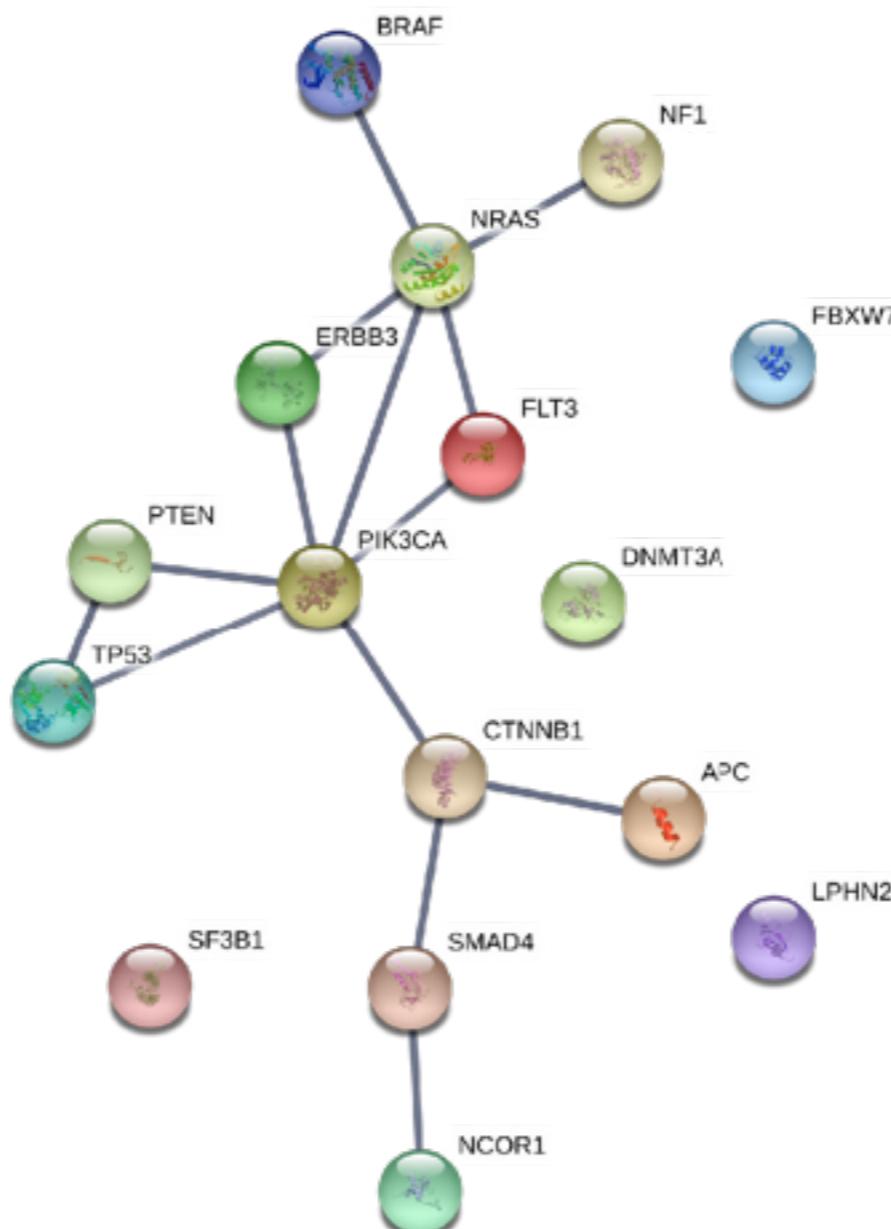
neighborhood *fusion* *cooccurrence* *coexpression* *experimental* *textmining*

combined score_{proteinA-proteinB} = 0.856

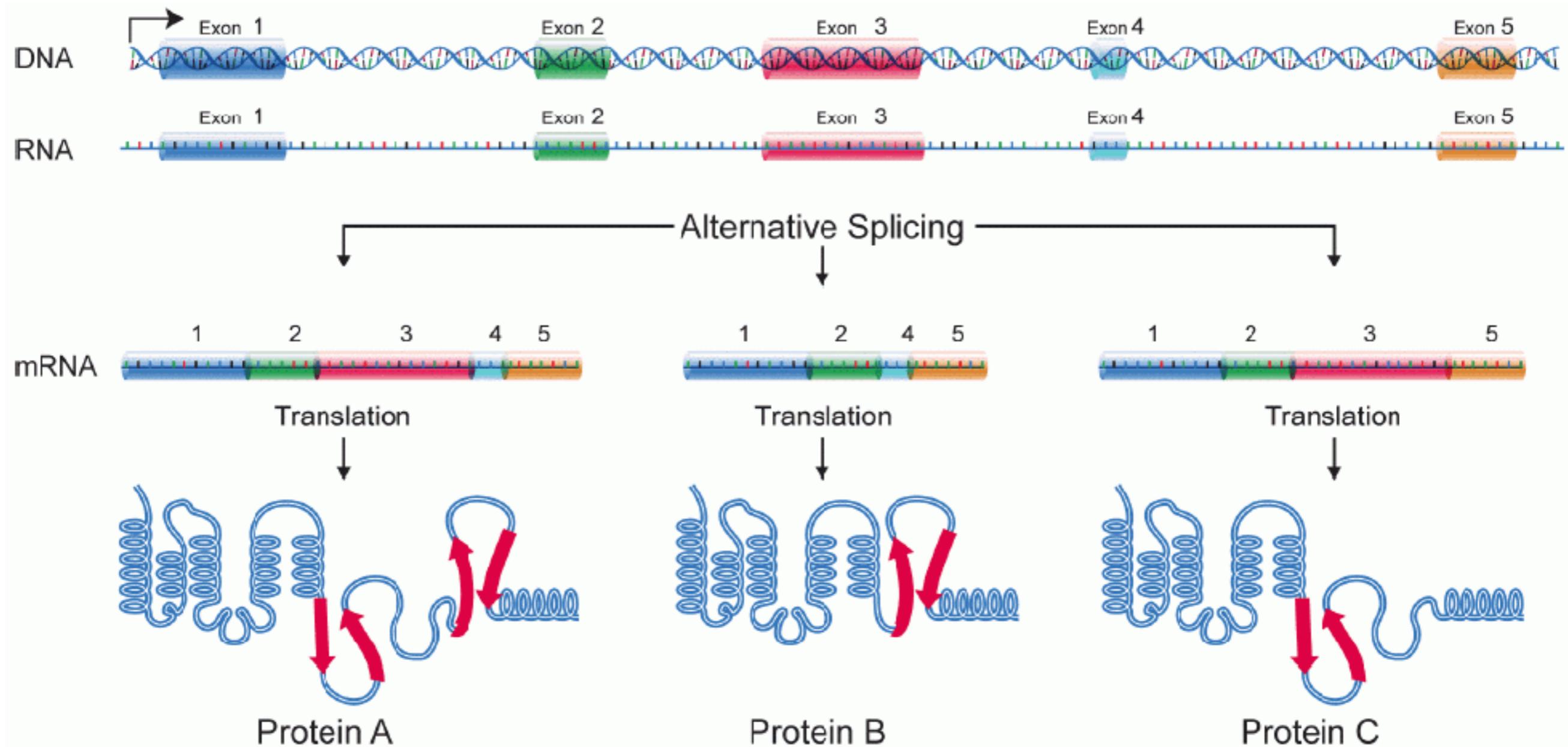
between 0 and 1, pseudoprobability,
“likelihood of functional association”

Isoform-specific interaction networks

- Protein interaction networks consider only canonical isoforms

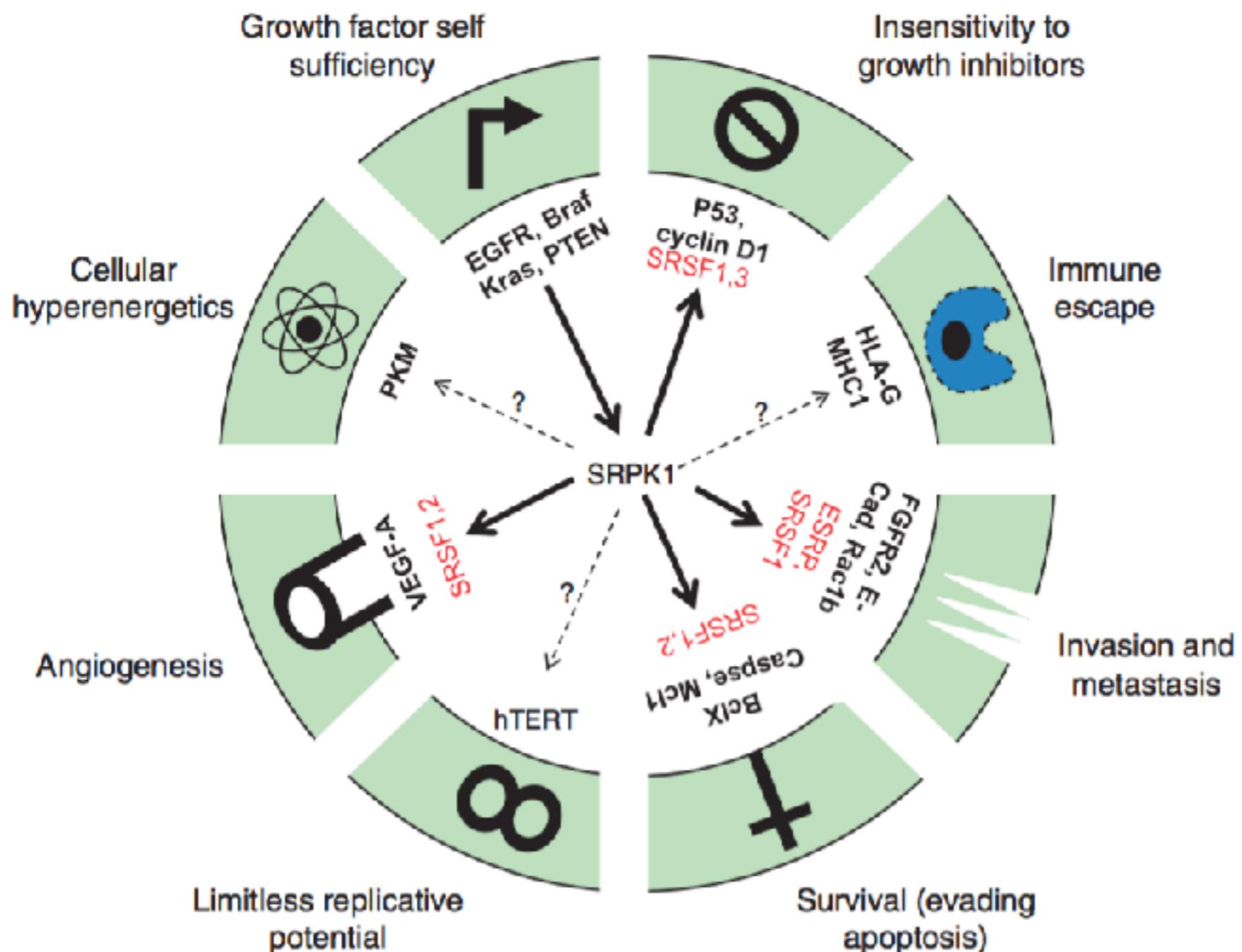


Alternative Splicing (AS)

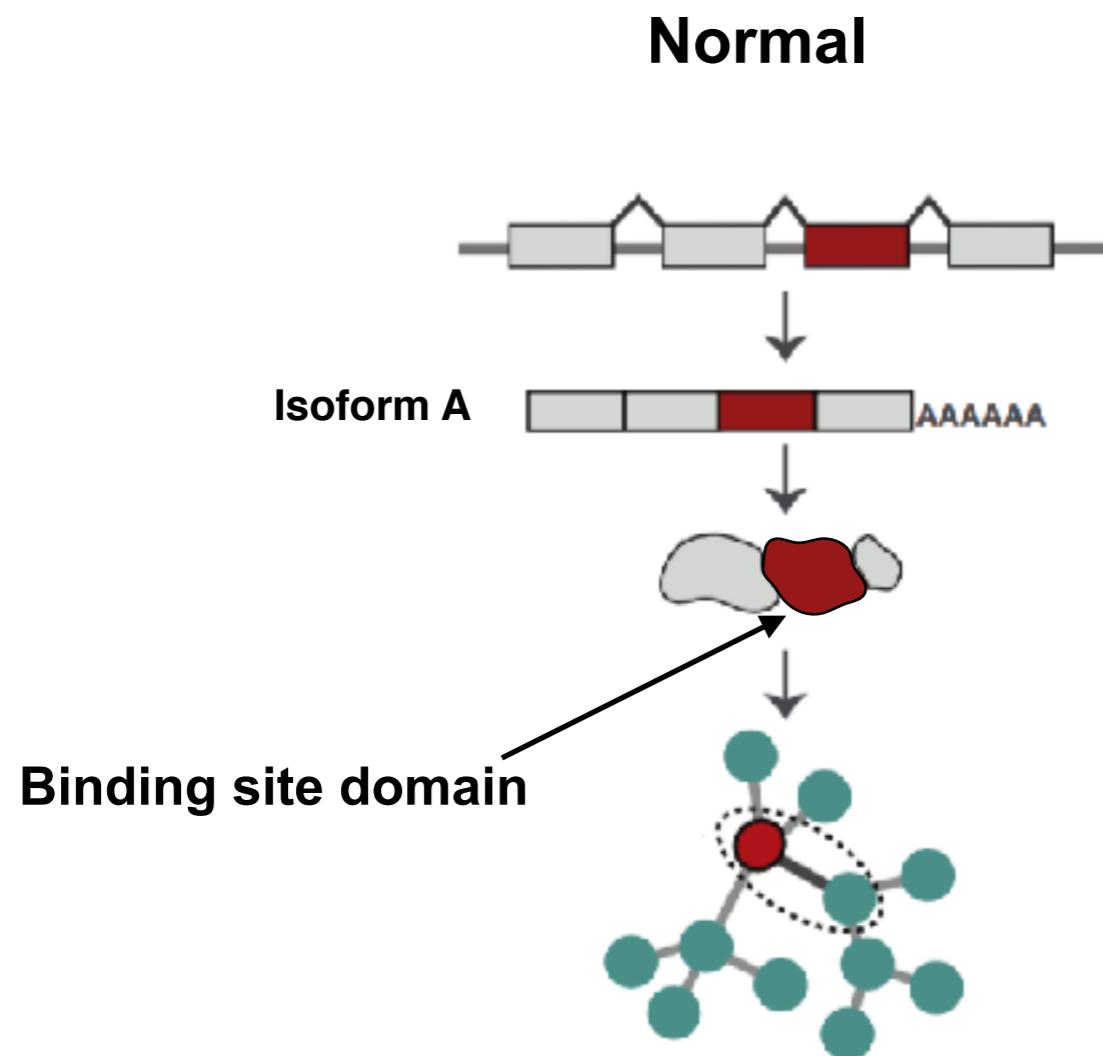


[wikipedia.org/wiki/Alternative_splicing](https://en.wikipedia.org/wiki/Alternative_splicing)

Hallmarks of Alternative Splicing in Cancer

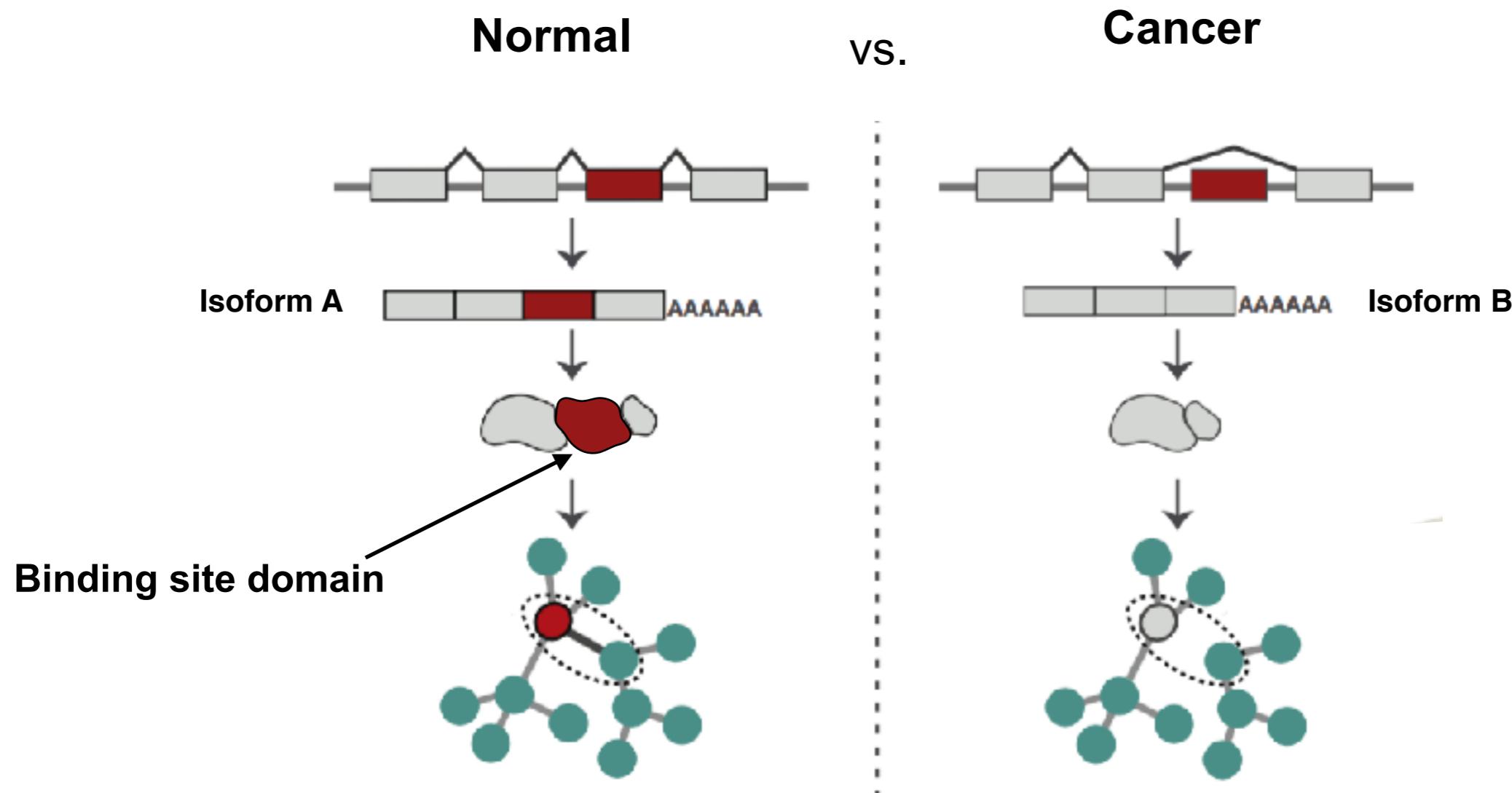


Functional Impact of AS Using Isoform-specific Interaction Networks



Kahraman, A., & Mering, von, C. (2019). Pathogenic impact of isoform switches in 1209 cancer samples covering 27 cancer types using an isoform-specific interaction network. bioRxiv, 742379
Buljan, M. et al. (2012) Tissue-specific splicing of disordered segments that embed binding motifs rewires protein interaction networks. Mol Cell 46, 871–883

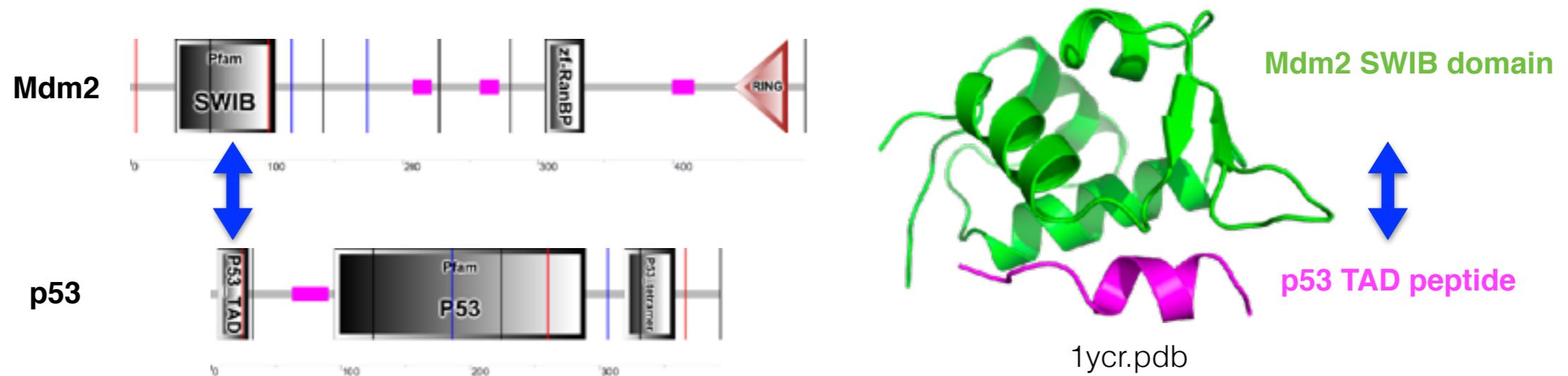
Functional Impact of AS Using Isoform-specific Interaction Networks



Kahraman, A., & Mering, von, C. (2019). Pathogenic impact of isoform switches in 1209 cancer samples covering 27 cancer types using an isoform-specific interaction network. bioRxiv, 742379
Buljan, M. et al. (2012) Tissue-specific splicing of disordered segments that embed binding motifs rewires protein interaction networks. Mol Cell 46, 871–883

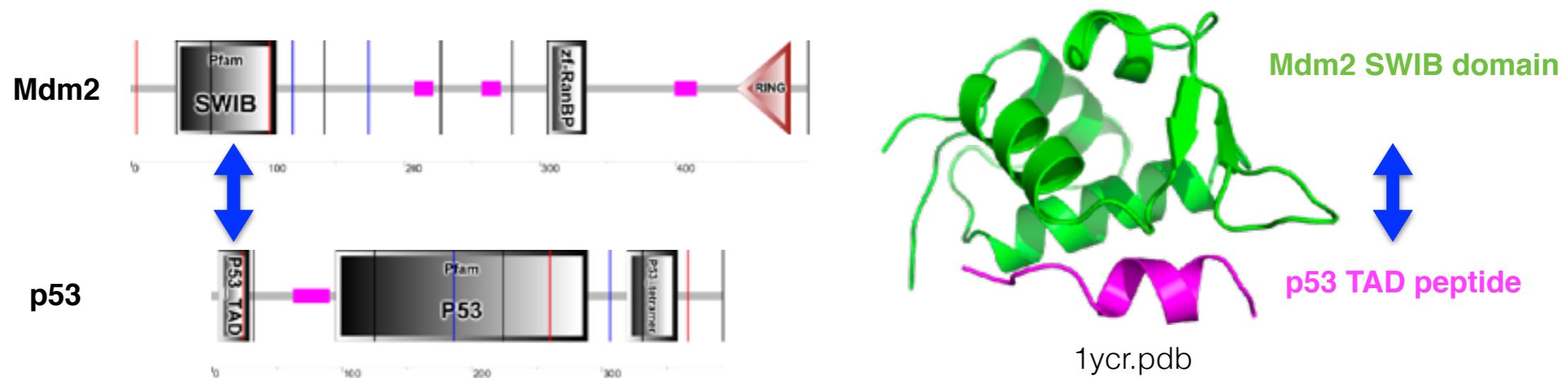
Mdm2-p53 Interactions disrupted due to cancer specific AS

- Interaction between Mdm2 and p53 via SWIB and TAD domain



Mdm2-p53 Interactions disrupted due to cancer specific AS

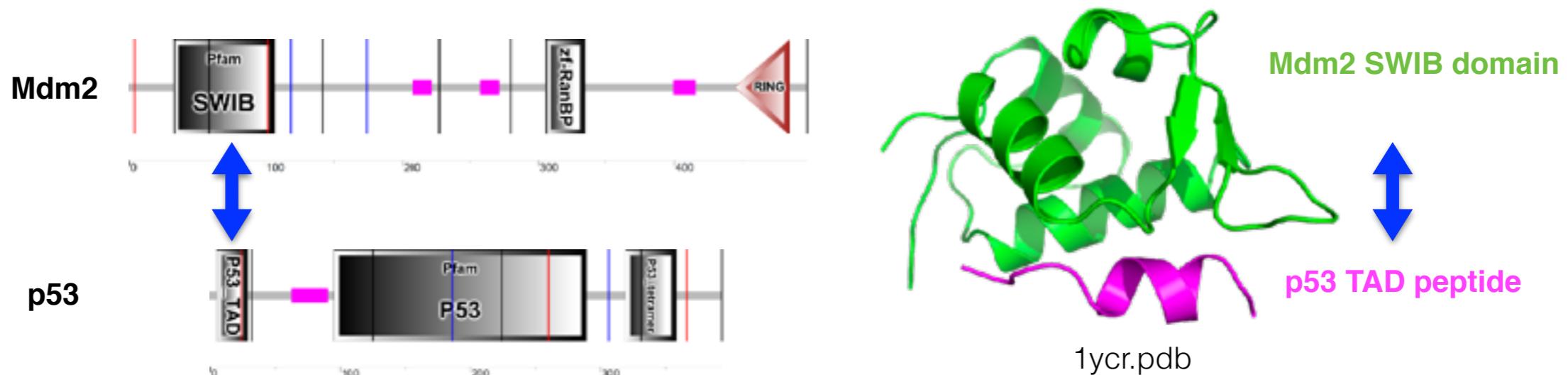
- Interaction between Mdm2 and p53 via SWIB and TAD domain



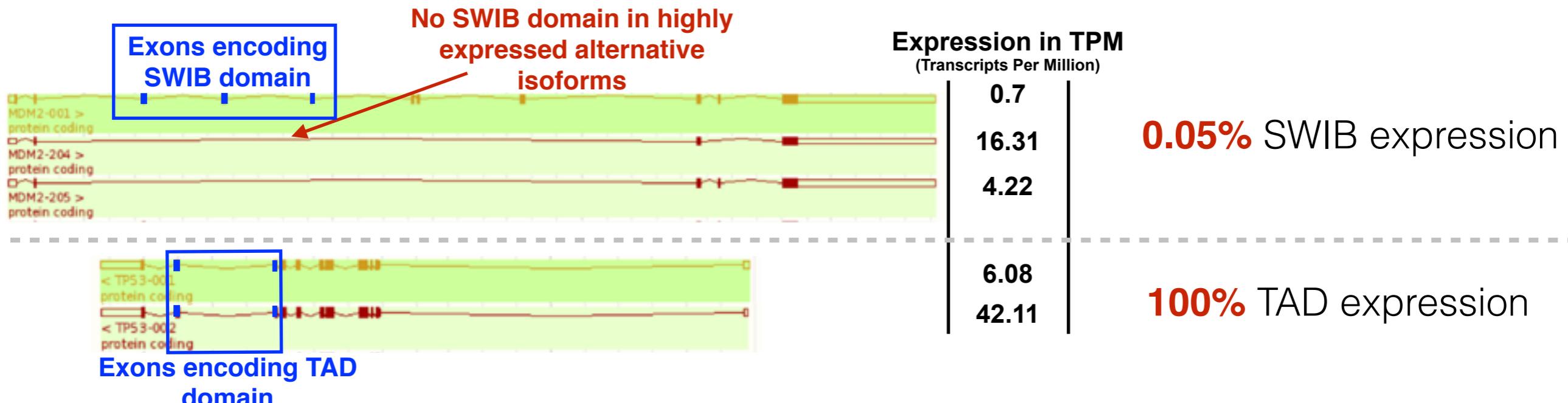
- Mdm2 SWIB domain is little expressed

Mdm2-p53 Interactions disrupted due to cancer specific AS

- Interaction between Mdm2 and p53 via SWIB and TAD domain

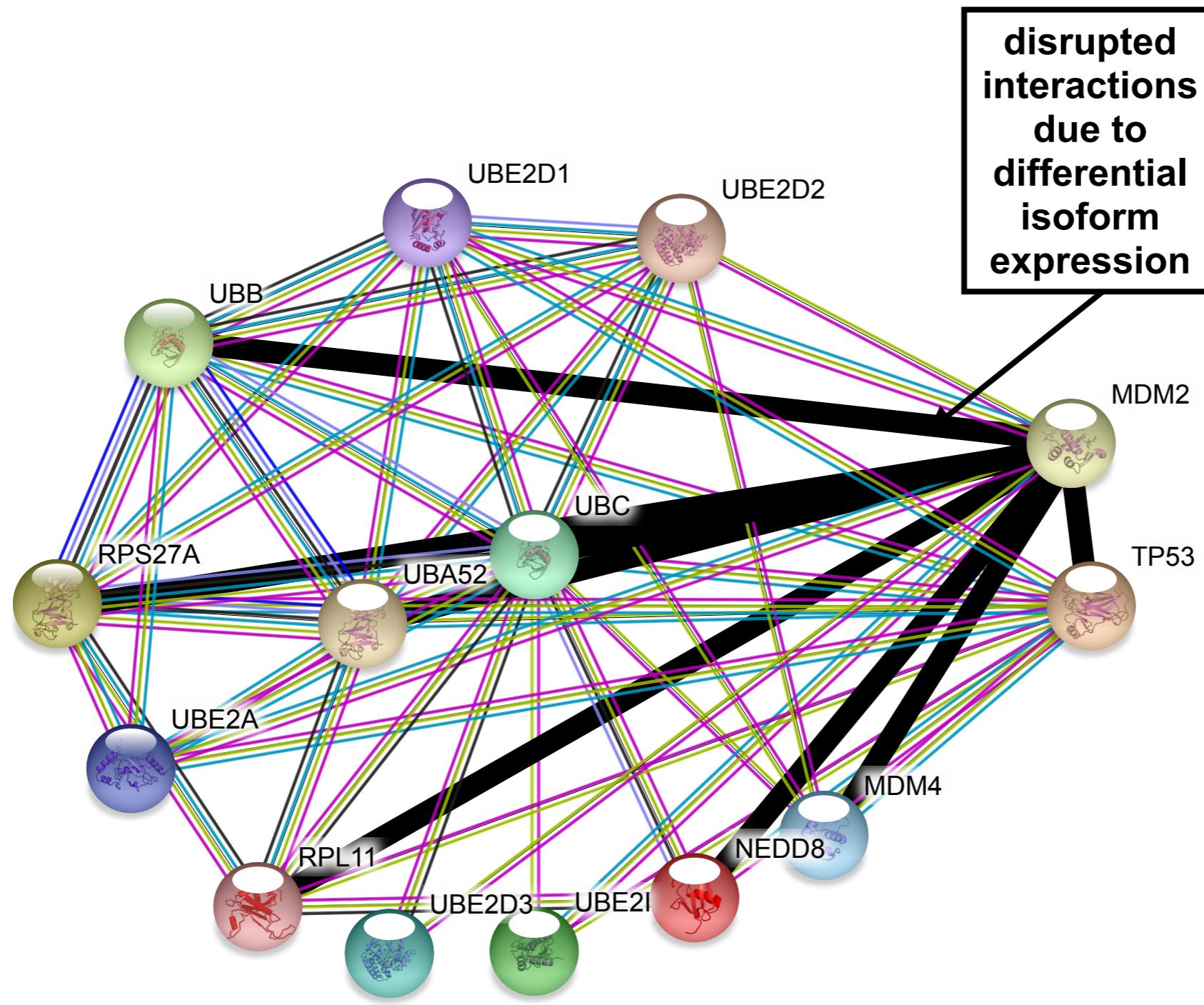


- Mdm2 SWIB domain is little expressed



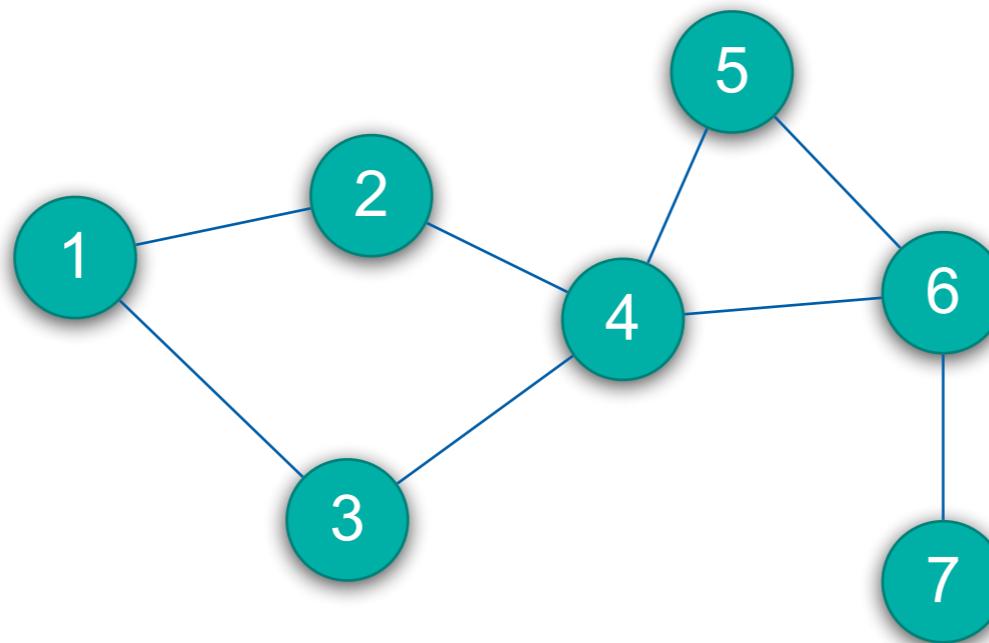
Functional Impact of AS Using Isoform-specific Interaction Networks

Functional Impact of AS Using Isoform-specific Interaction Networks



Network Properties

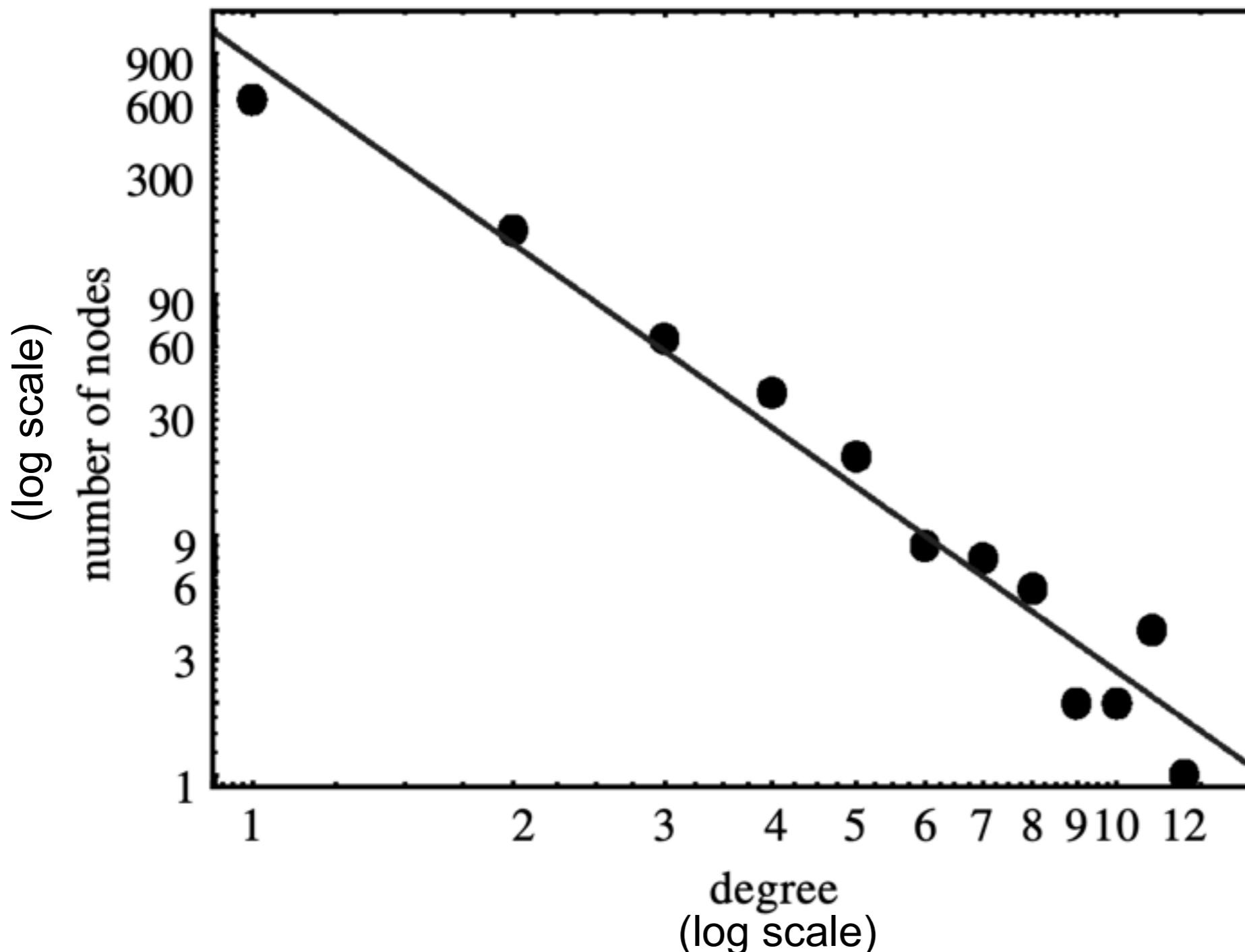
Graph Properties



- **Degree/connectivity** k_i = number of edges E of node *i*
- $$k_i = \sum_j E_{ij}$$
- e.g. $k_3 = 2$, $k_4 = 4$, $k_7 = 1$, ...

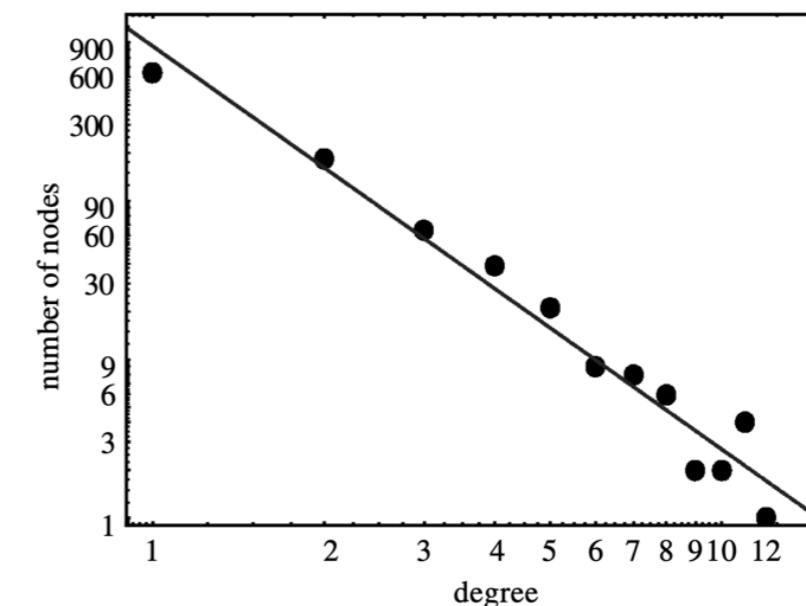
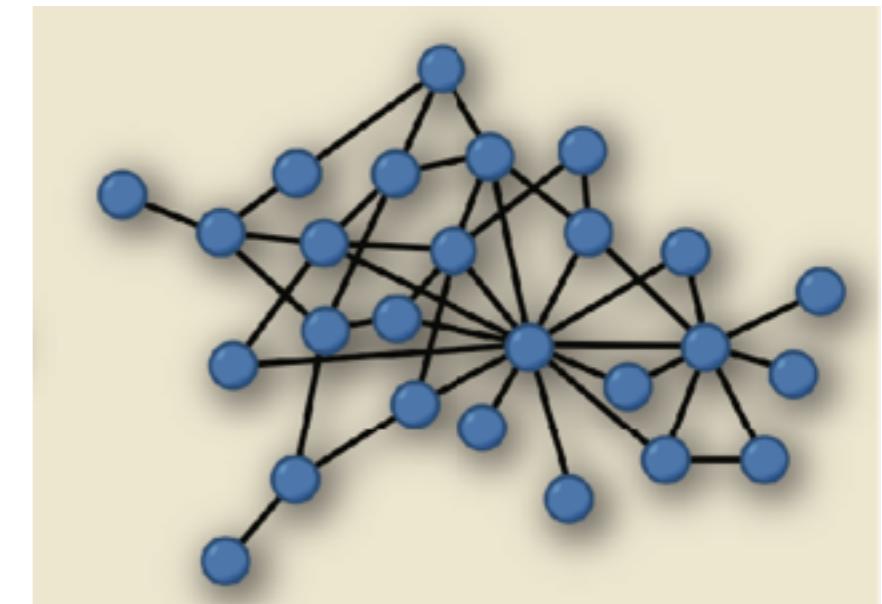
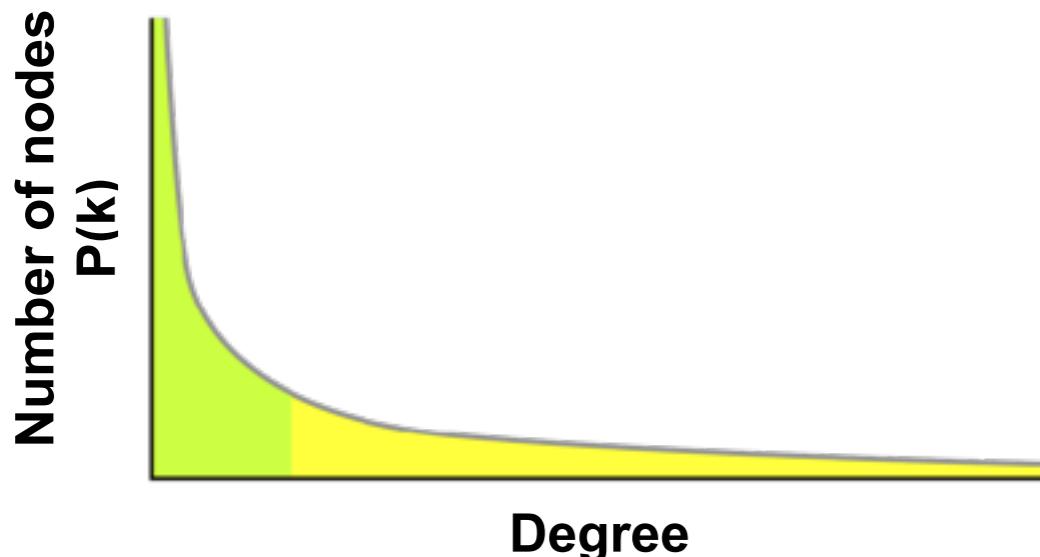
Degree distribution

- Networks can be classified according to their **degree distribution $P(k)$** , i.e. frequency of nodes having degree k .



Scale-free network

- Protein interaction networks (and many other biological networks) are almost **scale-free** as they follow a power-law degree distribution
- $P(k) \sim k^{-\gamma}$, typically with $2 \leq \gamma \leq 3$
- “**scale free**” as *functional form remains the same after rescaling:*
 - $P(ak) \sim a^{-\gamma}P(k)$
- **Robust against random failure**
- **Vulnerable to targeted attacks**



$$P(k) \sim k^{-2.55 \pm 0.35}$$

Hub Proteins

- **Highly connected nodes, i.e. hubs tend to**
 - **correspond to essential genes**
 - **be older and have evolved more slowly**
 - **have a tendency to be more abundant**
 - **have a larger diversity of phenotypic outcomes resulting from their deletion compared to the deletion of less connected proteins**
 - **be cancer associated genes.**

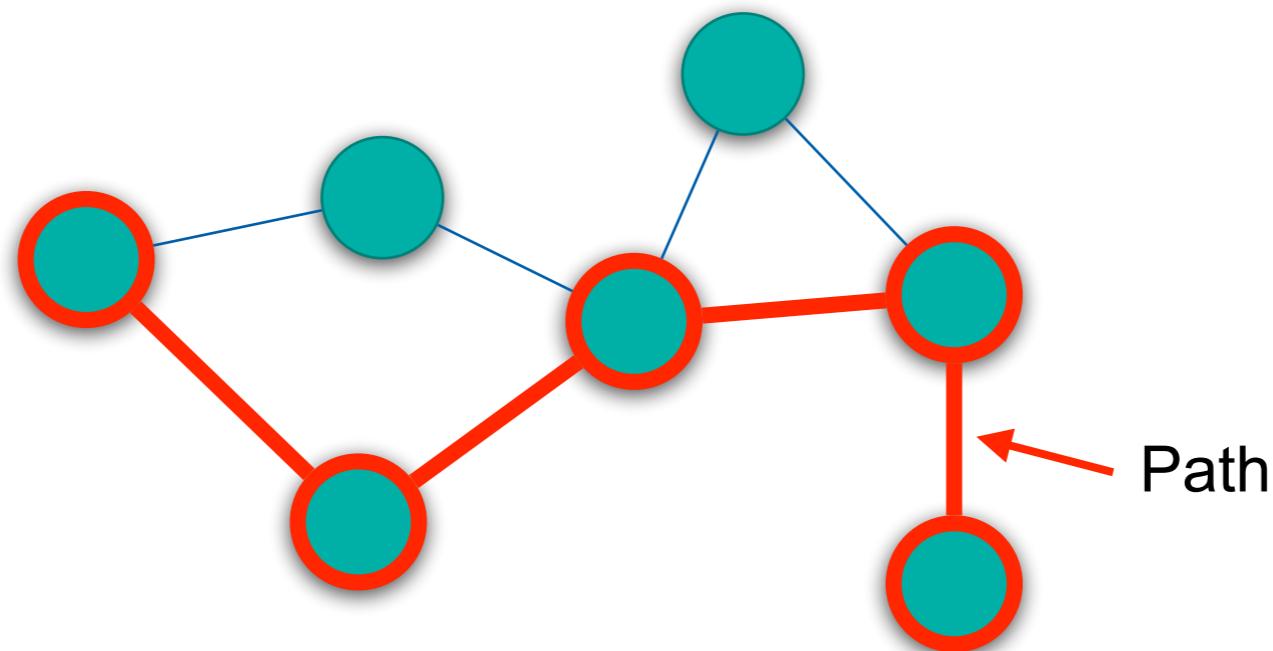
Small-world

- Six degrees of separation between any individual in the world



https://en.wikipedia.org/wiki/Small-world_experiment

Paths

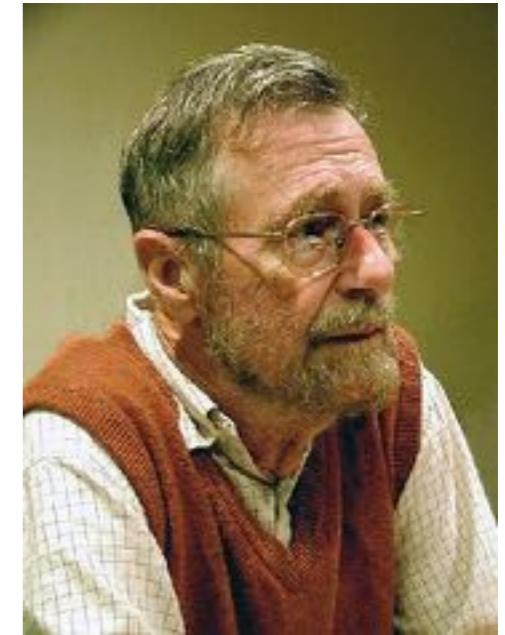


- A **path** is a sequence of alternating nodes and edges in which no node is visited more than once

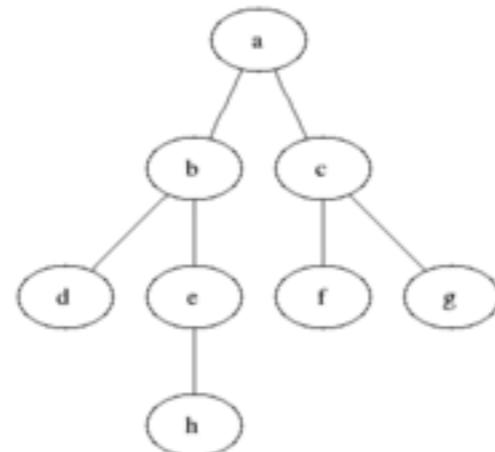
Shortest-Distance Algorithm

Shortest-Paths

- Dijkstra algorithm (1959)
- if edges have equal weights \Rightarrow Breadth-First Search



Edsger Wybe Dijkstra
*11 May 1930, Netherlands
https://en.wikipedia.org/wiki/Edsger_W._Dijkstra

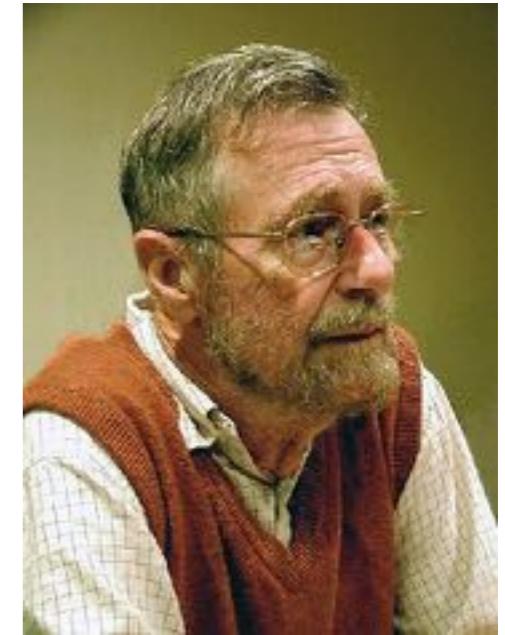


https://en.wikipedia.org/wiki/Breadth-first_search

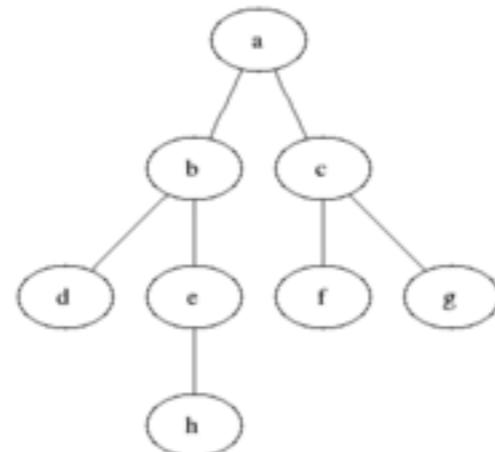
Dijkstra, E. W. (1959). "A note on two problems in connexion with graphs". *Numerische Mathematik*.

Shortest-Paths

- Dijkstra algorithm (1959)
- if edges have equal weights \Rightarrow Breadth-First Search



Edsger Wybe Dijkstra
*11 May 1930, Netherlands
https://en.wikipedia.org/wiki/Edsger_W._Dijkstra



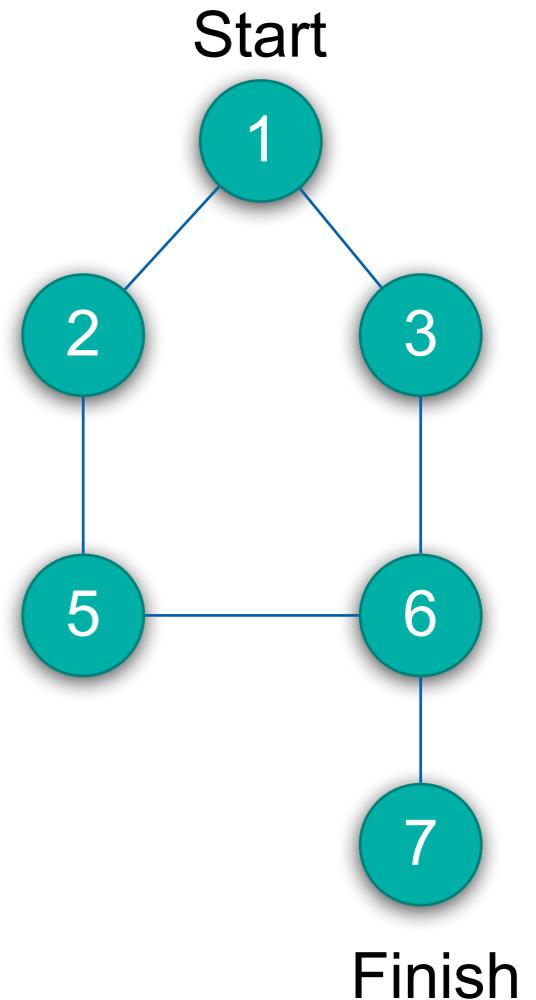
https://en.wikipedia.org/wiki/Breadth-first_search

Dijkstra, E. W. (1959). "A note on two problems in connexion with graphs". *Numerische Mathematik*.

Shortest-Paths

- Dijkstra algorithm
- if edges have equal weights \Rightarrow Breadth-First Search
- Pseudocode with reciprocal function BFS
- Initial code

```
function BFS(nodes):
    let queue = []
    foreach node in nodes:
        node.visited = true
        foreach neighbour in node.getNeighbours():
            let distance = node.distance + distance(node, neighbour)
            neighbour.distance = distance
            queue.add(neighbour)
    BFS(queue)
```

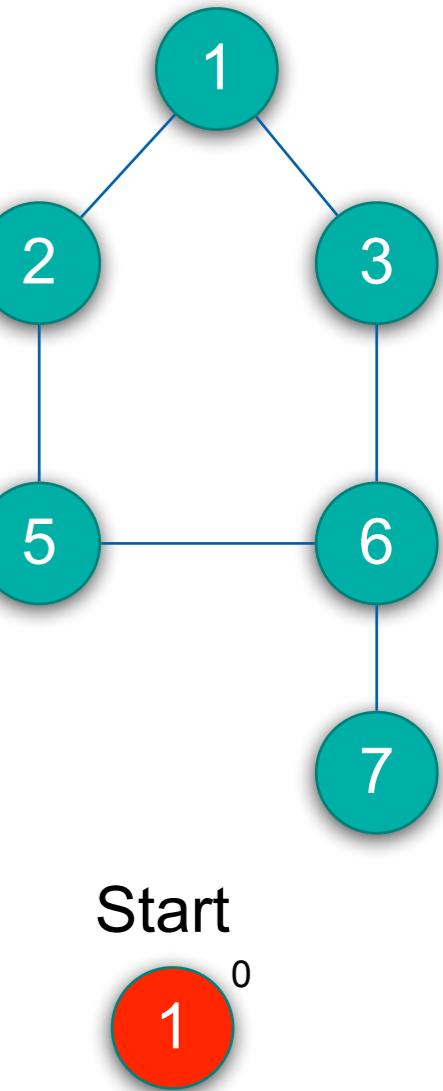


Finish

Shortest-Paths

- Dijkstra algorithm
- if edges have equal weights \Rightarrow Breadth-First Search
- Pseudocode with reciprocal function BFS
- Initial code

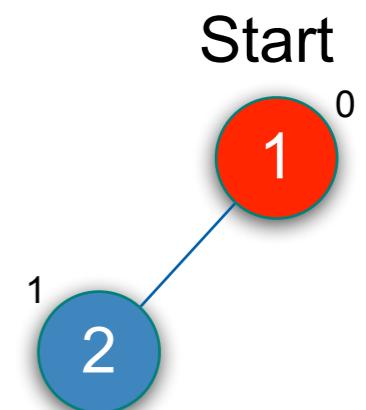
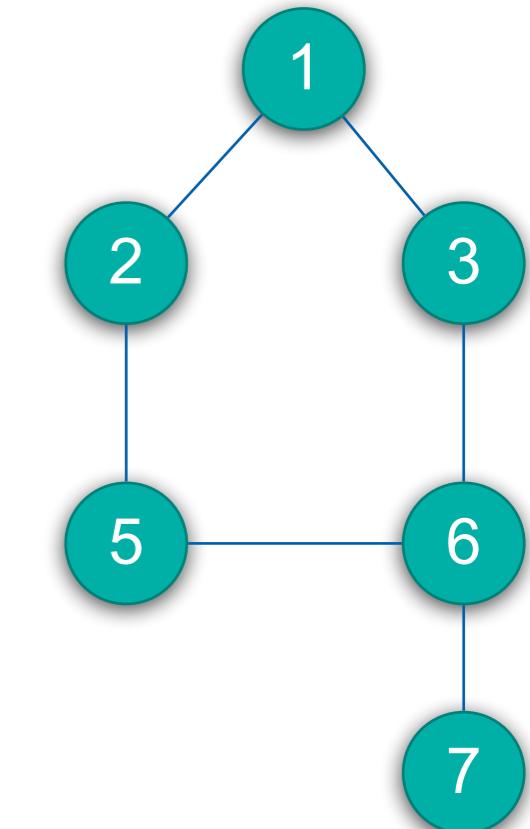
```
function BFS(nodes):
    let queue = []
    foreach node in nodes:
        node.visited = true
        foreach neighbour in node.getNeighbours():
            let distance = node.distance + distance(node, neighbour)
            neighbour.distance = distance
            queue.add(neighbour)
    BFS(queue)
```



Shortest-Paths

- Dijkstra algorithm
- if edges have equal weights \Rightarrow Breadth-First Search
- Pseudocode with reciprocal function BFS
- Initial code

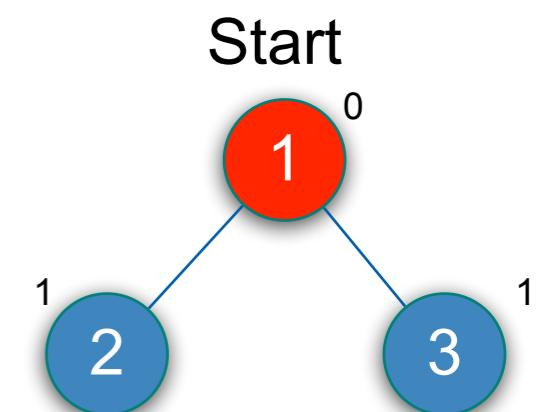
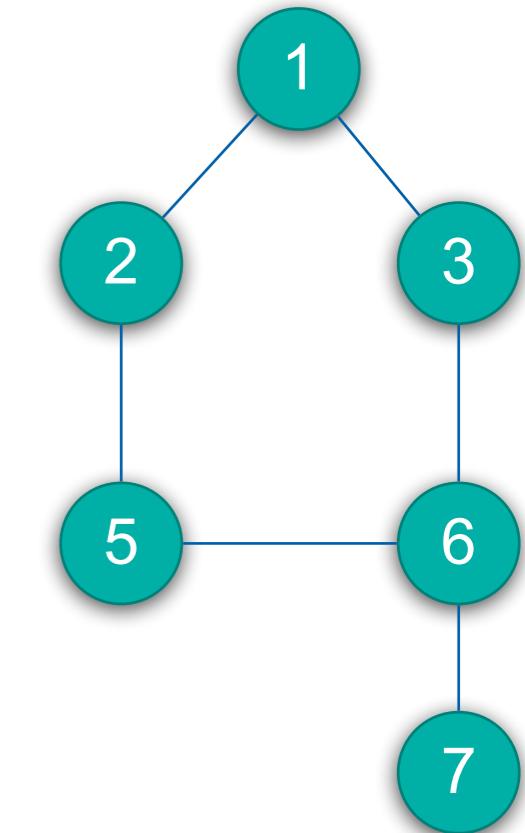
```
function BFS(nodes):
    let queue = []
    foreach node in nodes:
        node.visited = true
        foreach neighbour in node.getNeighbours():
            let distance = node.distance + distance(node, neighbour)
            neighbour.distance = distance
            queue.add(neighbour)
    BFS(queue)
```



Shortest-Paths

- Dijkstra algorithm
- if edges have equal weights \Rightarrow Breadth-First Search
- Pseudocode with reciprocal function BFS
- Initial code

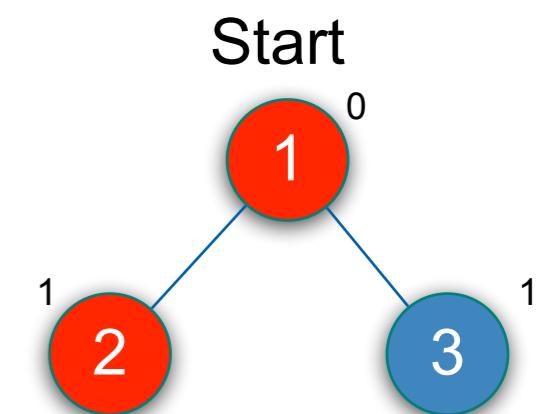
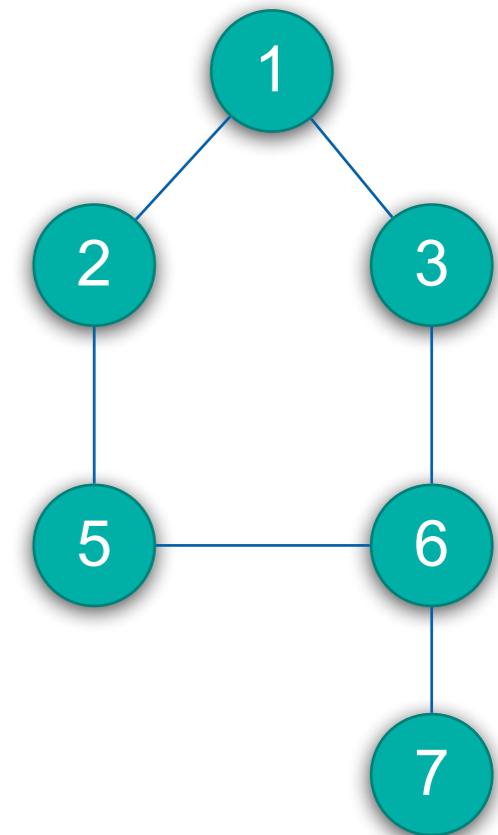
```
function BFS(nodes):
    let queue = []
    foreach node in nodes:
        node.visited = true
        foreach neighbour in node.getNeighbours():
            let distance = node.distance + distance(node, neighbour)
            neighbour.distance = distance
            queue.add(neighbour)
    BFS(queue)
```



Shortest-Paths

- Dijkstra algorithm
- if edges have equal weights \Rightarrow Breadth-First Search
- Pseudocode with reciprocal function BFS
- Initial code

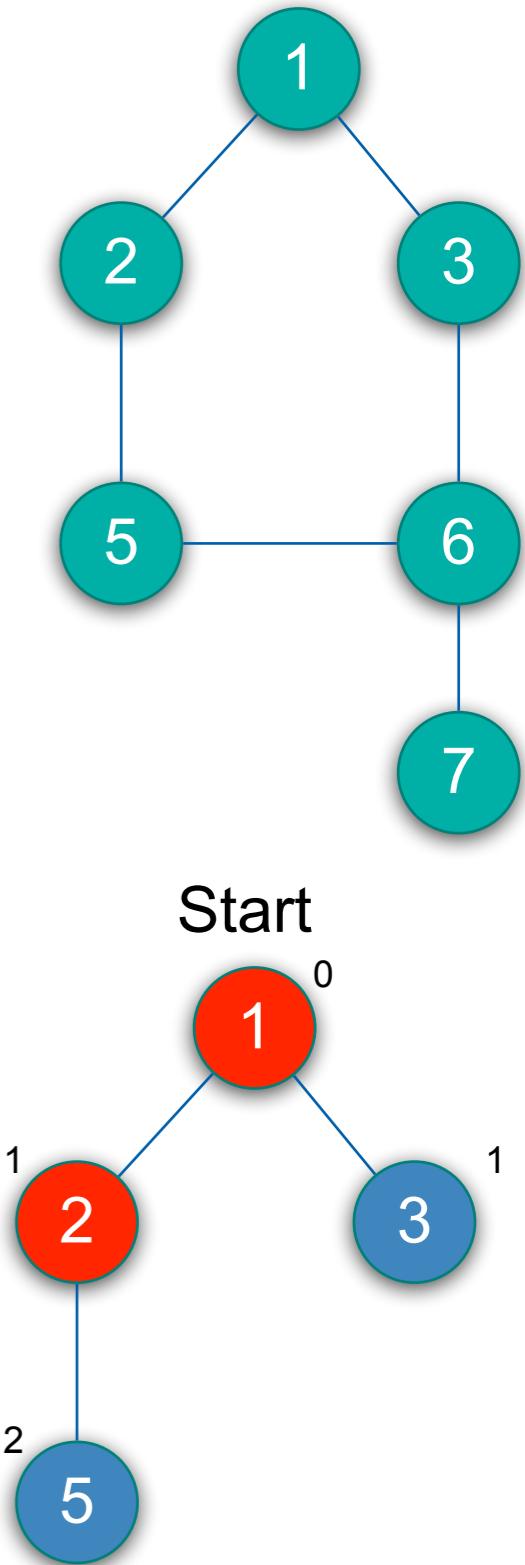
```
function BFS(nodes):
    let queue = []
    foreach node in nodes:
        node.visited = true
        foreach neighbour in node.getNeighbours():
            let distance = node.distance + distance(node, neighbour)
            neighbour.distance = distance
            queue.add(neighbour)
    BFS(queue)
```



Shortest-Paths

- Dijkstra algorithm
- if edges have equal weights \Rightarrow Breadth-First Search
- Pseudocode with reciprocal function BFS
- Initial code

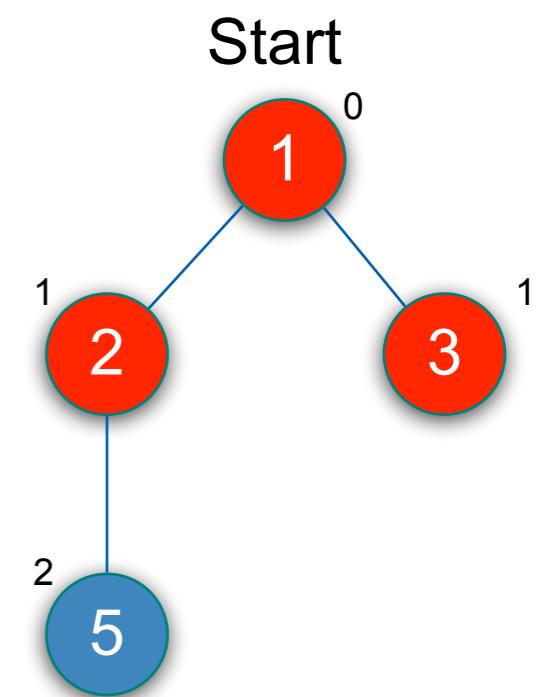
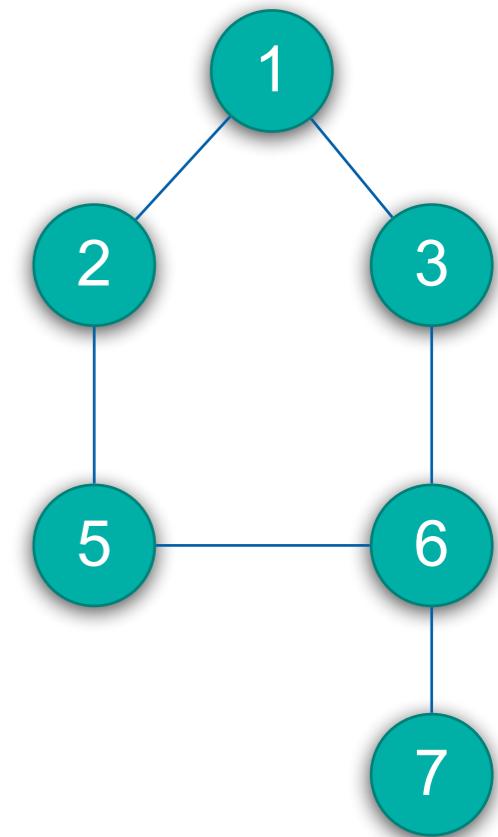
```
function BFS(nodes):
    let queue = []
    foreach node in nodes:
        node.visited = true
        foreach neighbour in node.getNeighbours():
            let distance = node.distance + distance(node, neighbour)
            neighbour.distance = distance
            queue.add(neighbour)
    BFS(queue)
```



Shortest-Paths

- Dijkstra algorithm
- if edges have equal weights \Rightarrow Breadth-First Search
- Pseudocode with reciprocal function BFS
- Initial code

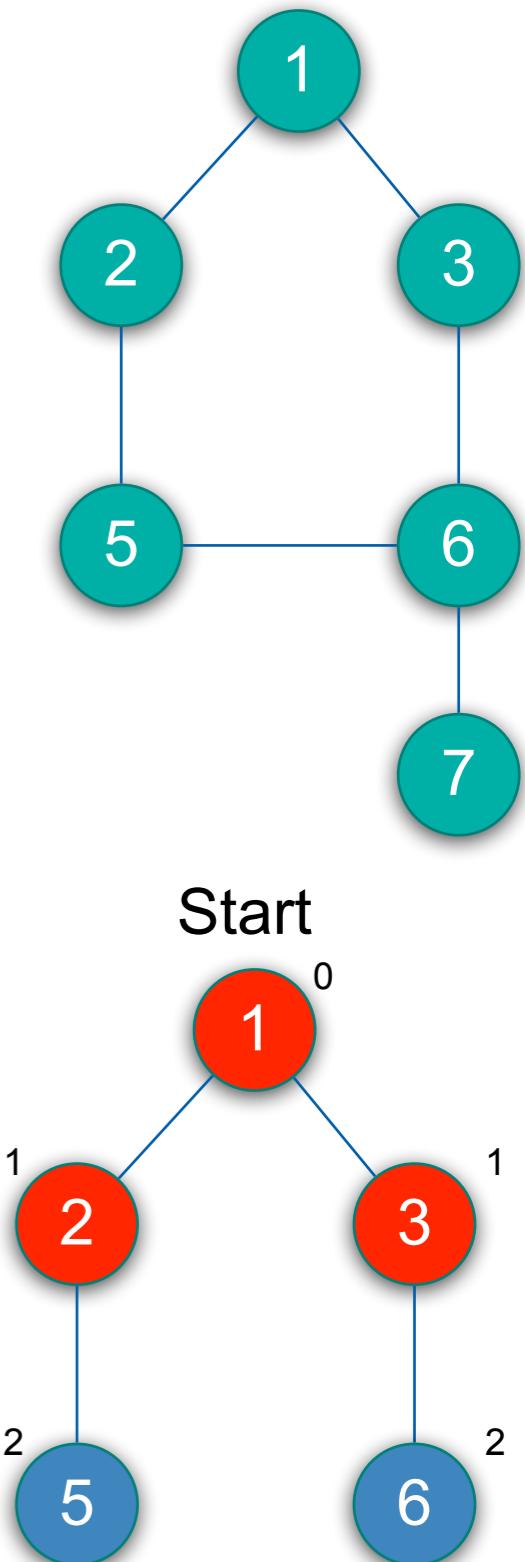
```
function BFS(nodes):
    let queue = []
    foreach node in nodes:
        node.visited = true
        foreach neighbour in node.getNeighbours():
            let distance = node.distance + distance(node, neighbour)
            neighbour.distance = distance
            queue.add(neighbour)
    BFS(queue)
```



Shortest-Paths

- Dijkstra algorithm
- if edges have equal weights \Rightarrow Breadth-First Search
- Pseudocode with reciprocal function BFS
- Initial code

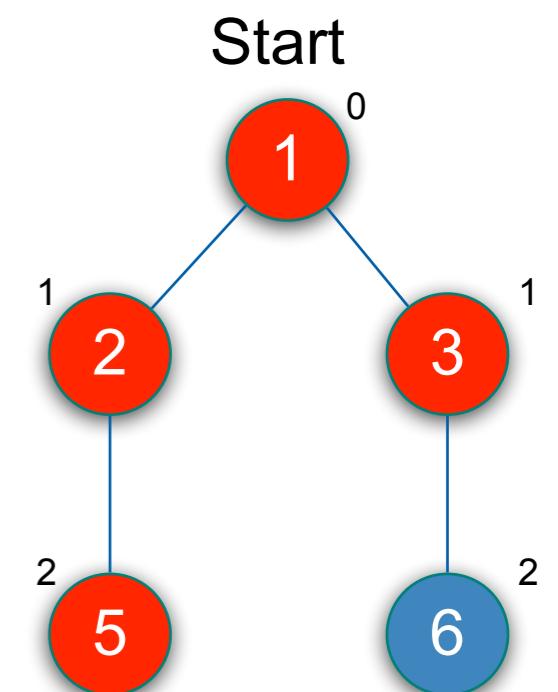
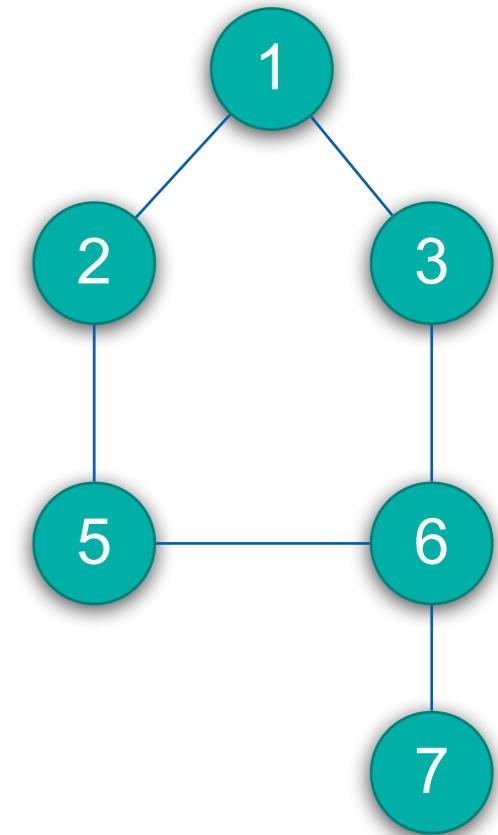
```
function BFS(nodes):
    let queue = []
    foreach node in nodes:
        node.visited = true
        foreach neighbour in node.getNeighbours():
            let distance = node.distance + distance(node, neighbour)
            neighbour.distance = distance
            queue.add(neighbour)
    BFS(queue)
```



Shortest-Paths

- Dijkstra algorithm
- if edges have equal weights \Rightarrow Breadth-First Search
- Pseudocode with reciprocal function BFS
- Initial code

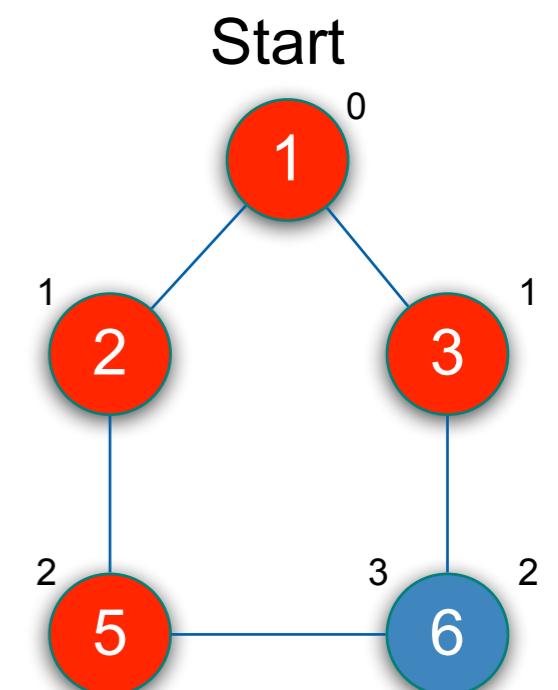
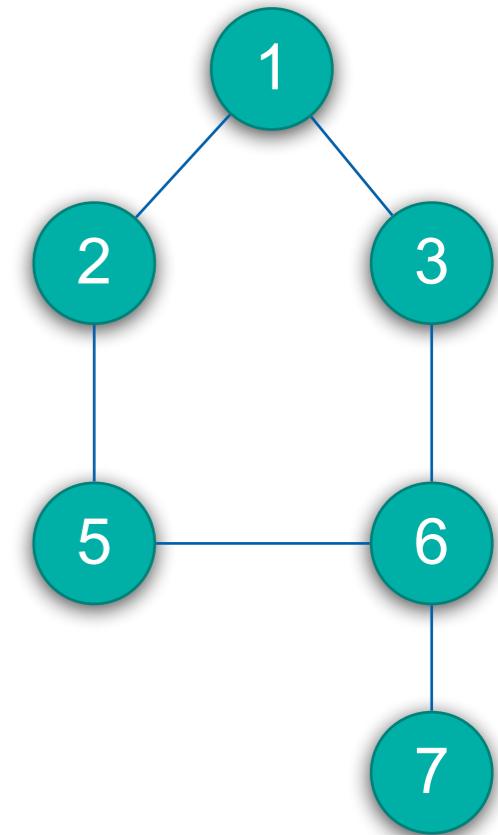
```
function BFS(nodes):
    let queue = []
    foreach node in nodes:
        node.visited = true
        foreach neighbour in node.getNeighbours():
            let distance = node.distance + distance(node, neighbour)
            neighbour.distance = distance
            queue.add(neighbour)
BFS(queue)
```



Shortest-Paths

- Dijkstra algorithm
- if edges have equal weights \Rightarrow Breadth-First Search
- Pseudocode with reciprocal function BFS
- Initial code

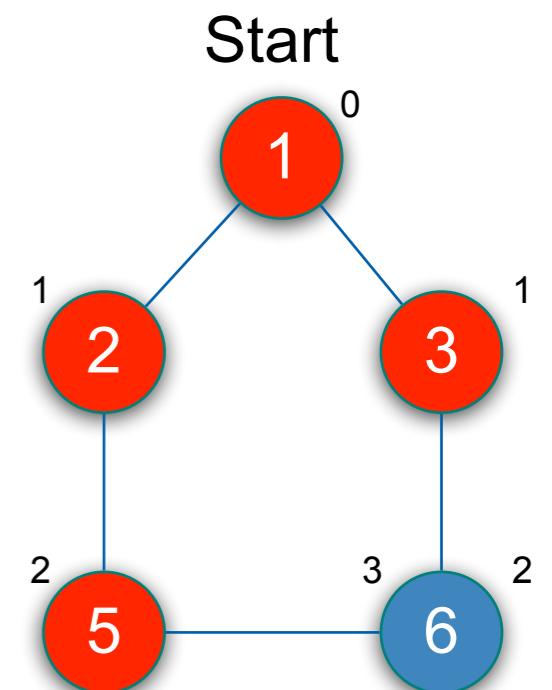
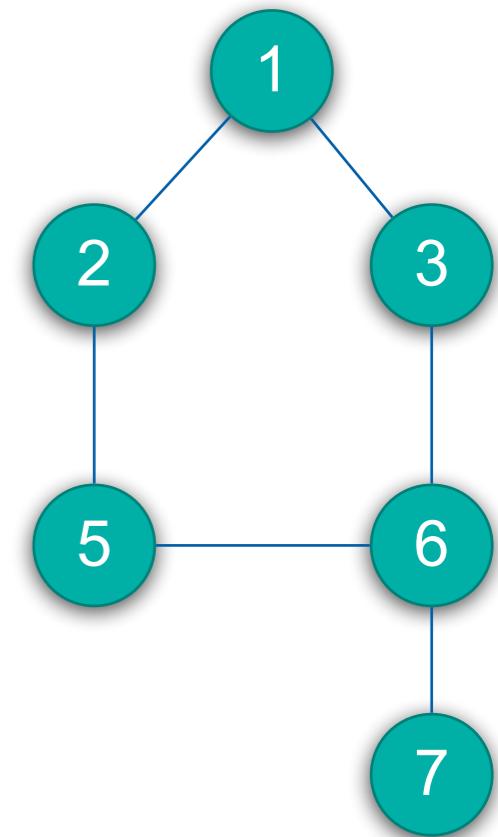
```
function BFS(nodes):
    let queue = []
    foreach node in nodes:
        node.visited = true
        foreach neighbour in node.getNeighbours():
            let distance = node.distance + distance(node, neighbour)
            neighbour.distance = distance
            queue.add(neighbour)
    BFS(queue)
```



Shortest-Paths

- Dijkstra algorithm
- if edges have equal weights \Rightarrow Breadth-First Search
- Pseudocode with reciprocal function BFS
- Initial code + prevent back traveling

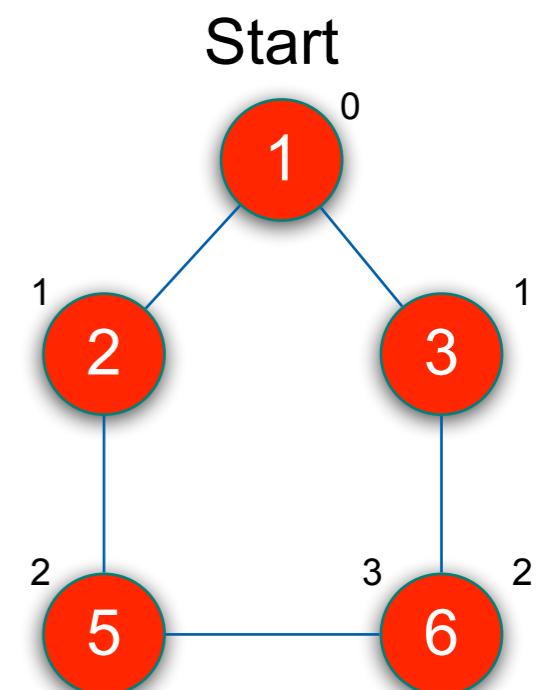
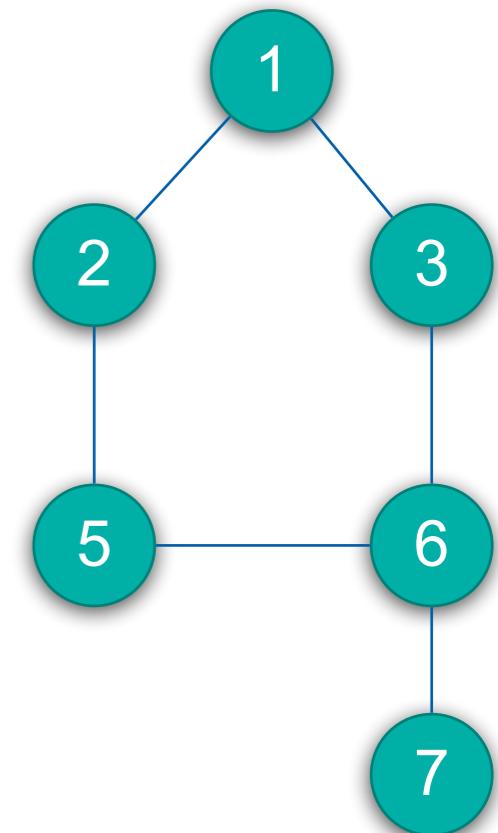
```
function BFS(nodes):
    let queue = []
    foreach node in nodes:
        node.visited = true
    foreach neighbour in node.getNeighbours():
        if not neighbour.visited:
            let distance = node.distance + distance(node, neighbour)
            neighbour.distance = distance
            queue.add(neighbour)
BFS(queue)
```



Shortest-Paths

- Dijkstra algorithm
- if edges have equal weights \Rightarrow Breadth-First Search
- Pseudocode with reciprocal function BFS
- Initial code + prevent back traveling

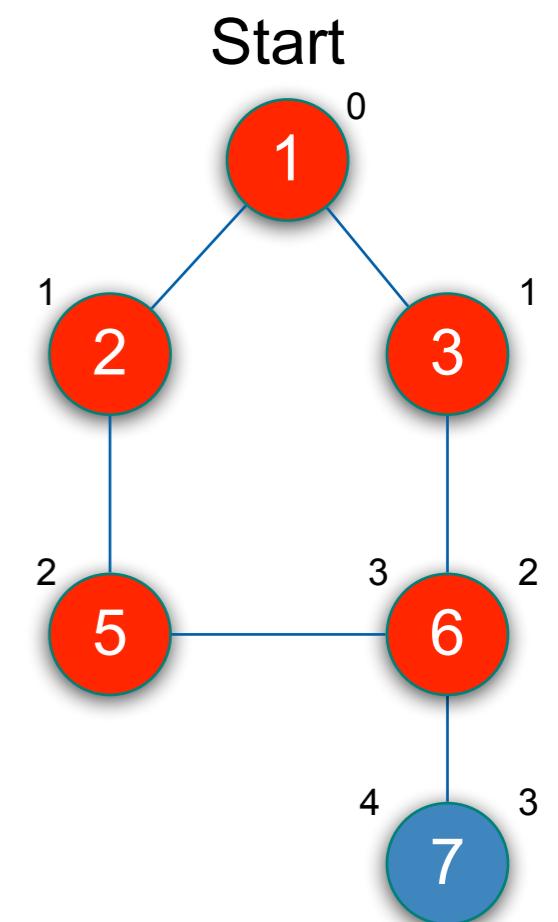
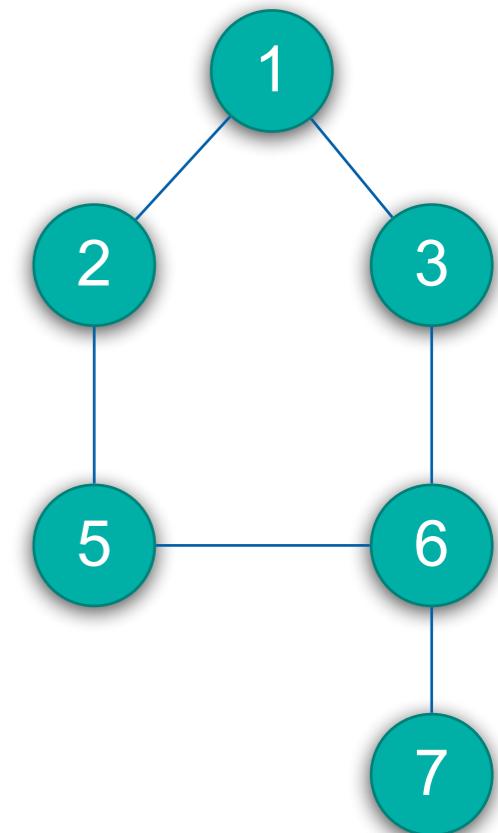
```
function BFS(nodes):
    let queue = []
    foreach node in nodes:
        node.visited = true
    foreach neighbour in node.getNeighbours():
        if not neighbour.visited:
            let distance = node.distance + distance(node, neighbour)
            neighbour.distance = distance
            queue.add(neighbour)
BFS(queue)
```



Shortest-Paths

- Dijkstra algorithm
- if edges have equal weights \Rightarrow Breadth-First Search
- Pseudocode with reciprocal function BFS
- Initial code + prevent back traveling

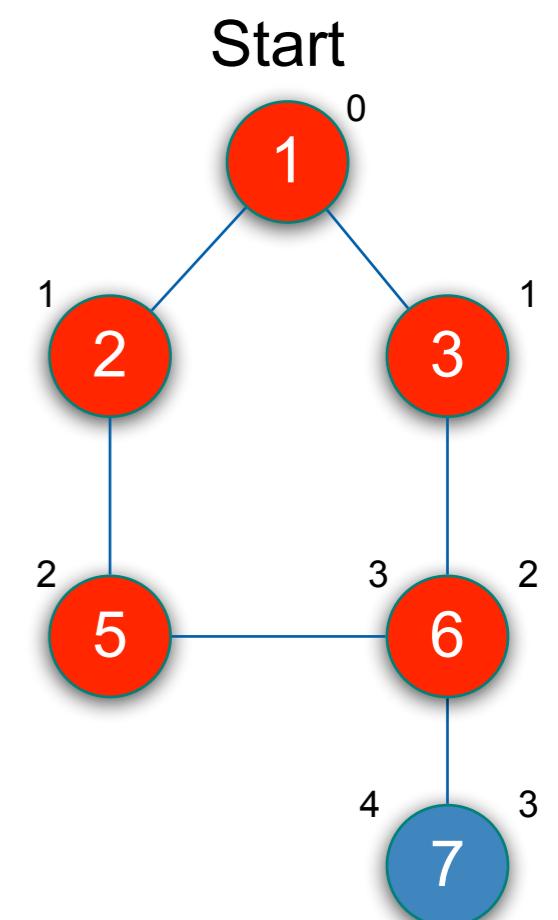
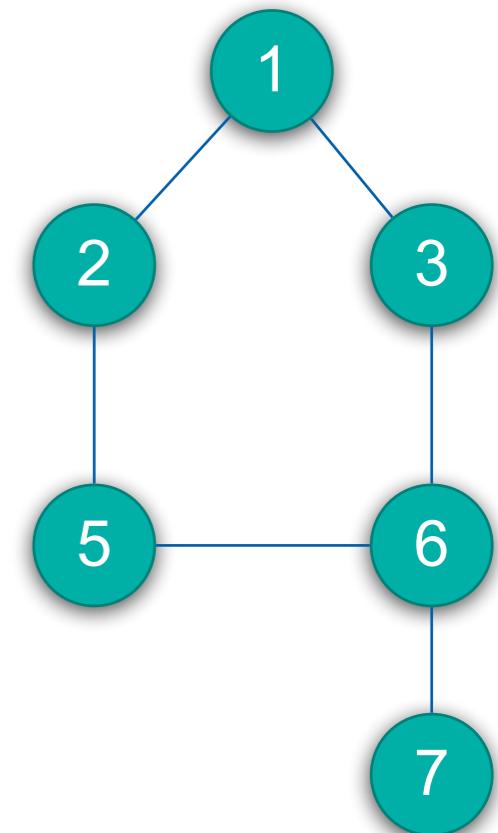
```
function BFS(nodes):
    let queue = []
    foreach node in nodes:
        node.visited = true
    foreach neighbour in node.getNeighbours():
        if not neighbour.visited:
            let distance = node.distance + distance(node, neighbour)
            neighbour.distance = distance
            queue.add(neighbour)
BFS(queue)
```



Shortest-Paths

- Dijkstra algorithm
- if edges have equal weights \Rightarrow Breadth-First Search
- Pseudocode with reciprocal function BFS
- Initial code + prevent back traveling + add stop condition

```
function BFS(nodes):
    let queue = []
    foreach node in nodes:
        node.visited = true
    foreach neighbour in node.getNeighbours():
        if not neighbour.visited:
            let distance = node.distance + distance(node, neighbour)
            neighbour.distance = distance
            queue.add(neighbour)
            if neighbour == goal:
                return
BFS(queue)
```



Visualising Molecular Interaction Networks

File Format for Molecular Interaction Data

File Format for Molecular Interaction Data

- **Simple interaction file (SIF or .sif format) (tab/whitespace delimited)**

- nodeA <relationship type> nodeB
nodeC <relationship type> nodeA
nodeD <relationship type> nodeE nodeF nodeB

- <relationship type>

- pp protein - protein interaction
pd protein -> DNA
pr protein -> reaction
rc reaction -> compound
cr compound -> reaction
gl genetic lethal relationship
pm protein-metabolite interaction
mp metabolite-protein interaction

File Format for Molecular Interaction Data

- **Simple interaction file (SIF or .sif format) (tab/whitespace delimited)**

- nodeA <relationship type> nodeB
nodeC <relationship type> nodeA
nodeD <relationship type> nodeE nodeF nodeB

- <relationship type>

- pp protein - protein interaction
pd protein -> DNA
pr protein -> reaction
rc reaction -> compound
cr compound -> reaction
gl genetic lethal relationship
pm protein-metabolite interaction
mp metabolite-protein interaction

- **PSI-MI Format (Proteomics Standards Initiative - Molecular Interaction)**

- XML-Format (current version 3.0)

File Format for Molecular Interaction Data

- **Simple interaction file (SIF or .sif format) (tab/whitespace delimited)**

• nodeA	<relationship type>	nodeB
nodeC	<relationship type>	nodeA
nodeD	<relationship type>	nodeE nodeF nodeB

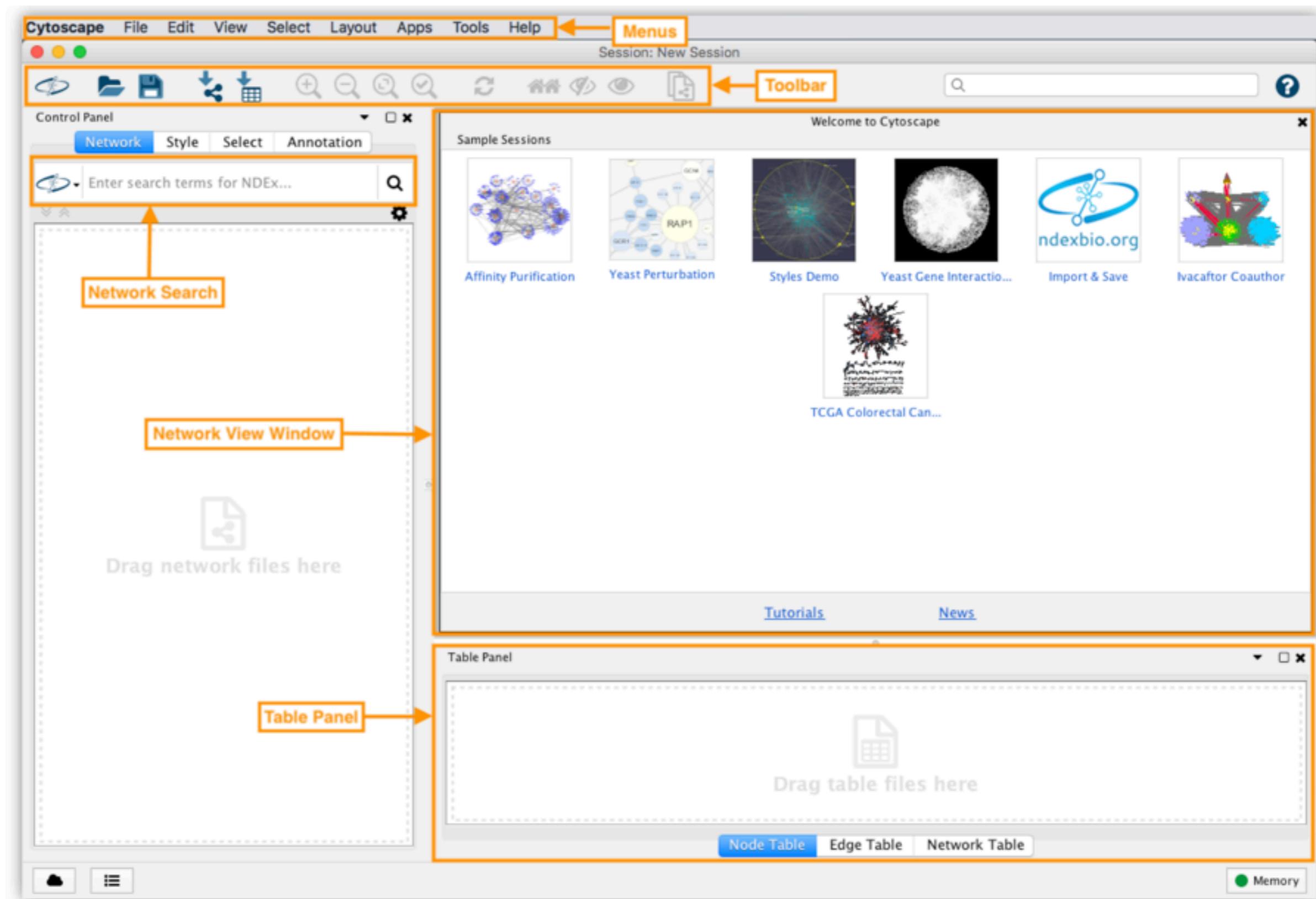
- <relationship type>

- pp protein - protein interaction
- pd protein -> DNA
- pr protein -> reaction
- rc reaction -> compound
- cr compound -> reaction
- gl genetic lethal relationship
- pm protein-metabolite interaction
- mp metabolite-protein interaction

- **PSI-MI Format (Proteomics Standards Initiative)**

- **XML-Format (current version 3.0)**

```
<interaction id="2606553">
  <names>
    <shortLabel>2606553</shortLabel>
  </names>
  <availabilityRef ref="1" />
  <experimentList>
    <experimentRef ref="BIOTGRID-PUBLICATION-218410" />
  </experimentList>
  <participantList>
    <proteinParticipant>
      <proteinInteractorRef ref="BIOTGRID-118061" />
      <role>bait</role>
    </proteinParticipant>
    <proteinParticipant>
      <proteinInteractorRef ref="BIOTGRID-123527" />
      <role>prey</role>
    </proteinParticipant>
  </participantList>
  <interactionType>
    <names>
      <shortLabel>Affinity Capture-MS</shortLabel>
    </names>
    <xref>
      <primaryRef db="" id="" secondary="" version="" />
    </xref>
  </interactionType>
  <xref>
    <primaryRef db="pubmed" id="29961565" secondary="" vers
  </xref>
</interaction>
```



Cytoscape - StringApp

Control Panel

Network Style Select Annotation ClueGO

colon cancer

1 of 1 Network selected

String Network – colon cancer 1

String Network – colon cancer 50 361

Results Panel

STRING

Glass ball effect Structure images

STRING style labels String colors

Singletons Highlight first neighbors

Functional enrichment Enriched publications

Select query

Tissue filters Compartment filters Selected nodes

Nodes Edges

Table Panel

String Network – colon cancer

shared name name stringdb canonical name display name stringdb full name stringdb database identifier stringdb description @id stringdb namespace stringdb node type

shared name	name	stringdb canonical name	display name	stringdb full name	stringdb database identifier	stringdb description	@id	stringdb namespace	stringdb node type
9606.ENSP0000034...	9606.ENS...	P43166	CA7		9606.ENSP00000345659	Carbonate dehydrogenase 9...	stringdb:9...	stringdb	protein
9606.ENSP0000038...	9606.ENS...	P05231	IL6		9606.ENSP00000385675	B-cell stimulatory factor 3...	stringdb:9...	stringdb	protein
9606.ENSP0000023...	9606.ENS...	P43246	MSH2		9606.ENSP00000233146	DNA mismatch repair protein MSH2	stringdb:9...	stringdb	protein
9606.ENSP0000022...	9606.ENS...	P60568	IL2		9606.ENSP00000226730	T-cell growth factor beta 1	stringdb:9...	stringdb	protein
9606.ENSP0000045...	9606.ENS...	P31749	AKT1		9606.ENSP00000451828	V-akt murine thymoma viral oncogene homolog 1	stringdb:9...	stringdb	protein

Node Table Edge Table Network Table

Memory

Cytoscape - StringApp

The screenshot shows the Cytoscape interface with the StringApp plugin loaded. The Control Panel on the left has a 'Network' tab selected, and a circled icon in the toolbar is highlighted.

The main workspace displays a network graph titled 'String Network - colon cancer' with 50 nodes and 361 edges. Nodes are colored by their STRING style labels, such as green for proteins and blue for metabolites.

The Results Panel on the right shows configuration options for the STRING visualization, including checkboxes for 'Glass ball effect', 'STRING style labels', 'Singletons', 'Structure images', 'String colors', and 'Highlight first neighbors'. It also includes tabs for 'Functional enrichment' and 'Enriched publications', and sections for 'Tissue filters', 'Compartment filters', and 'Selected nodes'.

The Table Panel at the bottom contains a table with columns for shared name, name, canonical name, display name, full name, database identifier, description, @id, namespace, and node type. The table lists several entries related to the colon cancer network.

shared name	name	canonical name	display name	full name	database identifier	description	@id	namespace	node type
9606.ENSP0000034...	9606.ENS...	P43166	CA7		9606.ENSP00000345659	Carbonate dehydrogenase 7	stringdb:9...	stringdb	protein
9606.ENSP0000038...	9606.ENS...	P05231	IL6		9606.ENSP00000385675	B-cell stimulatory factor 2	stringdb:9...	stringdb	protein
9606.ENSP0000023...	9606.ENS...	P43246	MSH2		9606.ENSP00000233146	DNA mismatch repair protein MSH2	stringdb:9...	stringdb	protein
9606.ENSP0000022...	9606.ENS...	P60568	IL2		9606.ENSP00000226730	T-cell growth factor beta 1	stringdb:9...	stringdb	protein
9606.ENSP0000045...	9606.ENS...	P31749	AKT1		9606.ENSP00000451828	V-akt murine thymoma viral oncogene homolog 1	stringdb:9...	stringdb	protein

Cytoscape - Style>Select

Control Panel

Network Style Select Annotation ClueGO

STRING style v1.5 – colon cancer

Properties

Def. Map. Byp.

Border Paint

Border Width

Fill Color

Column: disease score

Mapping Type: Continuous Mapping

Current Mapping

Height

Image/Chart 1

Image/Chart 2

Image/Chart 3

Image/Chart Position 3

Label

Label Color

Label Font Size: 12

Label Position

Shape

Size

Transparency

Width

Lock node width and height

Node Edge Network

Results Panel

STRING

Glass ball effect

STRING style labels

Singletons

Structure images

String colors

Highlight first neighbors

Functional enrichment

Enriched publications

Select query

Tissue filters

Compartment filters

Selected nodes

String Network – colon cancer

Nodes Edges

Table Panel

Node Table Edge Table Network Table

Memory

shared name	name	stringdb canonical name	display name	stringdb full name	stringdb database identifier	stringdb description	@id	stringdb namespace	stringdb node type
9606.ENSP0000034...	9606.ENS...	P43166	CA7		9606.ENSP00000345659	Carbonate dehydrogenase 7	stringdb:9...	stringdb	protein
9606.ENSP0000038...	9606.ENS...	P05231	IL6		9606.ENSP00000385675	B-cell stimulatory factor 2	stringdb:9...	stringdb	protein
9606.ENSP0000023...	9606.ENS...	P43246	MSH2		9606.ENSP00000233146	DNA mismatch repair protein MSH2	stringdb:9...	stringdb	protein
9606.ENSP0000022...	9606.ENS...	P60568	IL2		9606.ENSP00000226730	T-cell growth factor beta 1	stringdb:9...	stringdb	protein
9606.ENSP0000045...	9606.ENS...	P31749	AKT1		9606.ENSP00000451828	V-akt murine thymoma viral oncogene homolog 1	stringdb:9...	stringdb	protein

Cytoscape - Style>Select

The screenshot shows the Cytoscape interface with the "Style" tab selected in the top navigation bar. A large red circle highlights the "Style" panel on the left, which contains settings for node properties like Fill Color, Column, Mapping Type, and Label.

The main workspace displays a network graph titled "String Network - colon cancer". Nodes are represented by colored circles, and edges by lines connecting them. A small inset graph in the bottom right corner shows a different view of the same network.

The "Results Panel" on the right includes a "STRING" section with various visualization and analysis options, such as "Glass ball effect", "Structure images", and "Singletons". It also features sections for "Functional enrichment", "Enriched publications", "Tissue filters", "Compartment filters", and "Selected nodes".

The "Table Panel" at the bottom provides a detailed view of the network data, showing columns for shared name, name, stringdb canonical name, display name, full name, database identifier, description, @id, namespace, and node type. The "Node Table" is currently selected.

shared name	name	stringdb canonical name	display name	full name	database identifier	description	@id	namespace	node type
9606.ENSP0000034...	9606.ENS...	P43166	CA7		9606.ENSP00000345659	Carbonate dehydrogenase 7	stringdb:9...	stringdb	protein
9606.ENSP0000038...	9606.ENS...	P05231	IL6		9606.ENSP00000385675	B-cell stimulatory factor 6	stringdb:9...	stringdb	protein
9606.ENSP0000023...	9606.ENS...	P43246	MSH2		9606.ENSP00000233146	DNA mismatch repair protein MSH2	stringdb:9...	stringdb	protein
9606.ENSP0000022...	9606.ENS...	P60568	IL2		9606.ENSP00000226730	T-cell growth factor beta 2	stringdb:9...	stringdb	protein
9606.ENSP0000045...	9606.ENS...	P31749	AKT1		9606.ENSP00000451828	V-akt murine thymoma viral oncogene homolog 1	stringdb:9...	stringdb	protein

Summary

- Molecular Interaction Networks exist in various flavours
 - Different molecules
 - Different interaction types
 - Different scales
- High-throughput technologies like NGS, microarrays, mass-spectrometry allow to probe the interaction network of “entire” genomes, proteomes and transcriptomes
- Interaction networks can be predicted with sophisticated bioinformatics methods
- Breadth-first search algorithm to compute shortest path
- Biological networks tend to be “scale-free”

Literature

Literature

- Yu, H., Braun, P., Yildirim, M. A., Lemmens, I., Venkatesan, K., Sahalie, J., et al. (2008). High-quality binary protein interaction map of the yeast interactome network. *Science*.
- Gavin, A.-C., Aloy, P., Grandi, P., Krause, R., Boesche, M., Marzioch, M., et al. (2006). Proteome survey reveals modularity of the yeast cell machinery. *Nature*.
- Mering, von, C., Jensen, L. J., Snel, B., Hooper, S. D., Krupp, M., Foglierini, M., et al. (2005). STRING: known and predicted protein-protein associations, integrated and transferred across organisms. *Nucleic Acids Research*, 33(Database issue), D433–7.
- Vidal, M., Cusick, M. E., & Barabási, A.-L. (2011). Interactome networks and human disease. *Cell*, 144(6), 986–998.
- Herzog, F., Kahraman, A., Boehringer, D., Mak, R., Bracher, A., Walzthoeni, T., et al. (2012). Structural probing of a protein phosphatase 2A network by chemical cross-linking and mass spectrometry. *Science*, 337(6100), 1348–1352.
- Helms, V. (2018). *Principles of Computational Cell Biology*. John Wiley & Sons.
- Seebacher, J., & Gavin, A.-C. (2011). SnapShot: Protein-Protein Interaction Networks. *Cell*, 144(6).