Bioinformatics III: Network View of Cell Biology

Molecular Systems Biology: "It's both + molecular interactions"

- genetic information ➞ molecular structure ➞ biochemical function ➞ phenotype

- highly connected network of various interactions, dependencies

=> study networks
V1 - Introduction

A cell is a crowded environment
=> many different proteins,
    metabolites, compartments, …

At the microscopic level
=> direct two-body interactions

At the macroscopic level
=> complex behavior

Can we understand the behavior
from the interactions?

=> Connectivity

Lecture – Overview

Protein complexes: **spatial** structure
  => experiments, spatial fitting, docking

Protein association:
  => interface properties, spatial simulations

Protein-Protein-Interaction Networks: **pairwise** connectivity
  => data from experiments, quality check

PPI: static network **structure**
  => network measures, clusters, modules, …

Gene regulation: cause and **response**
  => Boolean networks

Metabolic networks: steady state of **large networks**
  => FBA, extreme pathways

Metabolic networks / signaling networks: **dynamics**
  => ODEs, modules, stochastic effects
Lecture – Table of contents - Chapters

1. Introduction – Networks in Biological Cells
2. Structures of Protein Complexes and Subcellular Structures
3. Analysis of protein-protein binding
4. Algorithms on mathematical graphs
5. Protein-Protein Interaction Networks – Pairwise Connectivity
6. Protein-Protein Interaction Networks – Structural Hierarchies
7. Protein-DNA interactions
8. Gene Expression and Protein Synthesis
9. Gene Regulatory Networks
10. Regulatory Noncoding RNA
11. Computational Epigenetics
12. Metabolic Networks
13. Kinetic Modeling of Cellular Processes
14. Stochastic processes in biological cells
15. Integrated Cellular Networks
## Lecture – type of mathematics

<table>
<thead>
<tr>
<th>Mathematical concept</th>
<th>Object of Investigation</th>
<th>Analysis of Complexity</th>
<th>Time-dependent</th>
<th>Treated in Chapter #</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mathematical graphs</td>
<td>protein-protein networks; protein complexes; gene regulatory networks</td>
<td>Yes</td>
<td>no</td>
<td>5, 6, 9, 10</td>
</tr>
<tr>
<td>Stoichiometric analysis; matrix algebra</td>
<td>metabolic networks*</td>
<td>yes (count # of possible paths that connect two metabolites)</td>
<td>no</td>
<td>12</td>
</tr>
<tr>
<td>Differential Equations</td>
<td>signal transduction, energy transduction, gene regulatory networks</td>
<td>No</td>
<td>yes</td>
<td>9, 13</td>
</tr>
<tr>
<td>Equations of motion</td>
<td>individual proteins, protein complexes</td>
<td>yes</td>
<td>14, 15</td>
<td></td>
</tr>
<tr>
<td>Correlation functions, Fourier transformation</td>
<td>reconstruction of two- and three-dimensional structures of cellular structures and individual molecules</td>
<td>No</td>
<td>yes, when applied on time-dependent data</td>
<td>2</td>
</tr>
<tr>
<td>Statistical tests</td>
<td>Differential expression and methylation; enriched network motifs</td>
<td>No</td>
<td>yes, when applied on time-dependent data</td>
<td>8, 9, 10</td>
</tr>
<tr>
<td>Machine learning (linear regression, hidden Markov model)</td>
<td>Predict gene expression, classify chromatin states</td>
<td>No</td>
<td>no</td>
<td>8, 11</td>
</tr>
</tbody>
</table>
Appetizer: A whole-cell model for the life cycle of the human pathogen *Mycoplasma genitalium* (15.2)

**Theory**

**A Whole-Cell Computational Model Predicts Phenotype from Genotype**

Jonathan R. Karr,1,2* Jayodita C. Singhvi,1,3,4 Derek N. Macklin,2 Miriam V. Gutachow,2 Jared M. Jacobs,2 Benjamin Bolival, Jr.,2 Nancy Asad-Garcia,2 John I. Glass,2 and Markus W. Covert1,2,*

1Graduate Program in Biophysics
2Department of Bioengineering
Stanford University, Stanford, CA 94305, USA
3J. Craig Venter Institute, Rockville, MD 20850, USA
4These authors contributed equally to this work
*Correspondence: mo covert@stanford.edu

http://dx.doi.org/10.1016/j.cell.2012.05.044

**Cell 150, 389-401 (2012)**
Divide and conquer approach (Caesar): split whole-cell model into 28 independent submodels

28 submodels are built / parametrized / iterated independently
Cell variables

System state is described by 16 cell variables

Colored lines: cell variables affected by individual submodels

Mathematical tools:
- Differential equations
- Stochastic simulations
- Flux balance analysis
List S1. Primary sources of the *M. genitalium* reconstruction.

<table>
<thead>
<tr>
<th>Data source</th>
<th>Content</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bernstein et al., 2002</td>
<td>mRNA half-lives</td>
</tr>
<tr>
<td>BioCyc</td>
<td>Genome annotation, metabolic reactions</td>
</tr>
<tr>
<td>BREnda</td>
<td>Reaction kinetics</td>
</tr>
<tr>
<td>CMR</td>
<td>Genome annotation</td>
</tr>
<tr>
<td>Deuerling et al., 2003</td>
<td>Chaperone substrates</td>
</tr>
<tr>
<td>DrugBank</td>
<td>Antibiotics</td>
</tr>
<tr>
<td>Eisen et al., 1999</td>
<td>DNA repair</td>
</tr>
<tr>
<td>Endo et al., 2007</td>
<td>Chaperone substrates</td>
</tr>
<tr>
<td>Feist et al., 2007</td>
<td>Metabolic reactions</td>
</tr>
<tr>
<td>Glass et al., 2006</td>
<td>Gene essentiality</td>
</tr>
<tr>
<td>Güell et al., 2009</td>
<td>Transcription unit structure</td>
</tr>
<tr>
<td>Gupta et al., 2007</td>
<td>N-terminal methionine cleavage</td>
</tr>
<tr>
<td>KEGG</td>
<td>Genome annotation, orthology</td>
</tr>
<tr>
<td>Kerner et al., 2005</td>
<td>Chaperone substrates</td>
</tr>
<tr>
<td>Krause et al., 2004</td>
<td>Terminal organelle assembly</td>
</tr>
<tr>
<td>Lindahl et al., 2000</td>
<td>DNA damage</td>
</tr>
<tr>
<td>Morowitz et al., 1962</td>
<td>Cell chemical composition</td>
</tr>
<tr>
<td>NCBI Gene</td>
<td>Genome annotation</td>
</tr>
<tr>
<td>Neidhardt et al., 1990</td>
<td>Cell chemical composition</td>
</tr>
<tr>
<td>Pei, 2009</td>
<td>RNA modification</td>
</tr>
<tr>
<td>PubChem</td>
<td>Metabolite structures</td>
</tr>
<tr>
<td>SABIO-RK</td>
<td>Reaction kinetics</td>
</tr>
<tr>
<td>Solabia</td>
<td>Media chemical composition</td>
</tr>
<tr>
<td>Suthers et al., 2009</td>
<td>Metabolic reactions</td>
</tr>
<tr>
<td>UniProt</td>
<td>Genome annotation</td>
</tr>
<tr>
<td>Weiner et al., 2000</td>
<td>Promoters</td>
</tr>
<tr>
<td>Weiner et al., 2003</td>
<td>mRNA expression</td>
</tr>
</tbody>
</table>
Growth of virtual cell culture

The model calculations were consistent with the observed doubling time!

Growth of three cultures (dilutions indicated by shade of blue) and a blank control measured by OD550 of the pH indicator phenol red. The doubling time, \( t \), was calculated using the equation at the top left from the additional time required by more dilute cultures to reach the same OD550 (black lines).
DNA-binding and dissociation dynamics of the oriC DnaA complex (red) and of RNA (blue) and DNA (green) polymerases for one in silico cell. The oriC DnaA complex recruits DNA polymerase to the oriC to initiate replication, which in turn dissolves the oriC DnaA complex. RNA polymerase traces (blue line segments) indicate individual transcription events. The height, length, and slope of each trace represent the transcript length, transcription duration, and transcript elongation rate, respectively.

Inset: several predicted collisions between DNA and RNA polymerases that lead to the displacement of RNA polymerases and incomplete transcripts.
Predictions for cell-cycle regulation

Distributions of the duration of three cell-cycle phases, as well as that of the total cell-cycle length, across 128 simulations.

There was relatively more cell-to-cell variation in the durations of the replication initiation (64.3%) and replication (38.5%) stages than in cytokinesis (4.4%) or the overall cell cycle (9.4%).

This data raised two questions:

1. what is the source of duration variability in the initiation and replication phases; and

2. why is the overall cell-cycle duration less varied than either of these phases?
Single-gene knockouts: essential vs. non-essential genes

Each column depicts the temporal dynamics of one representative in silico cell of each essential disruption strain class.

Dynamics significantly different from wild-type are highlighted in red.

The identity of the representative cell and the number of disruption strains in each category are indicated in parenthesis.
Literature

Lecture slides — available before the lecture

Suggested reading

=> check our web page
http://gepard.bioinformatik.uni-saarland.de/teaching/…

Textbooks

=> check computer science library
How to pass this course

**Schein** = you need to qualify for the **final exam** and pass it

**Final exam:**
written test of 180 min length about selected parts of the lecture
(slides will be defined 2 weeks before exam) *AND* about selected assignments

requirements for participation in final exam:
• 50% of the points from the **assignments**
• one assignment task presented @ blackboard in tutorial

Final exam will take place at the end of the semester.
In case you are sick (final exam) you should submit a medical certificate
to take the written re-exam (then this will be counted as first exam).

**Re-exam:** will take place in first week of the winter term 2018/19.
Everybody can take the re-exam (first exam failed or passed).
Assignments

Tutors: Thorsten Will, Duy Nguyen, Daria Gaidar, Markus Hollander

Tutorial: Tue, 12:15–13:45, E2 1, room 007

10 assignments with 100 points each

Assignments are part of the course material (not everything is covered in lecture)

=> one solution for two students (or one)
=> hand-written or one printable PDF/PS file per email
=> content: data analysis + interpretation — think!
=> no 100% solutions required!!!
=> attach the source code of the programs for checking (no suppl. data)
=> present one task at the blackboard

Hand in at the following Fri electronically until 13:15 or
printed at the start of the lecture.
Some Graph Basics

Network <=> Graph

Formal definition:

A graph $G$ is an ordered pair $(V, E)$ of a set $V$ of vertices and a set $E$ of edges. 

$$G = (V, E)$$

If $E = V^{(2)}$ => fully connected graph
Graph Basics II

Subgraph:

\[ G' = (V', E') \] is a subset of \[ G = (V, E) \]

Weighted graph:

Weights assigned to the edges

Practical question: how to define useful subgraphs?

Note: no weights for vertices
Walk the Graph

**Path** = sequence of connected vertices
start vertex => internal vertices => end vertex

Two paths are **independent** (internally vertex-disjoint),
if they have no internal vertices in common.

Vertices $u$ and $v$ are **connected**, if there exists a path from $u$ to $v$.
otherwise: disconnected

**Trail** = path, in which all edges are distinct

**Length** of a path = number of vertices $\parallel$ sum of the edge weights

How many paths connect the green to the red vertex?

How long are the shortest paths?

Find the four trails from the green to the red vertex.

How many of them are independent?
Local Connectivity: Degree/Degree Distribution

**Degree** $k$ of a vertex = number of edges at this vertex

Directed graph => distinguish $k_{in}$ and $k_{out}$

**Degree distribution** $P(k) = \text{fraction of nodes with } k \text{ connections}$

$$P(k) = \frac{n_k}{N}$$

$k$ connections

<table>
<thead>
<tr>
<th>$k$</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P(k)$</td>
<td>0</td>
<td>3/7</td>
<td>1/7</td>
<td>1/7</td>
<td>2/7</td>
</tr>
</tbody>
</table>

Area 1

$k_{in}: k_{out} = 1; 0$

<table>
<thead>
<tr>
<th>$k$</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P(k_{in})$</td>
<td>1/7</td>
<td>5/7</td>
<td>0</td>
<td>1/7</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$k$</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P(k_{out})$</td>
<td>2/7</td>
<td>3/7</td>
<td>1/7</td>
<td>1/7</td>
</tr>
</tbody>
</table>
Graph Representation e.g. by adjacency matrix

**Adjacency matrix** is a $N \times N$ 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

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

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>–</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>–</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>1</td>
<td>–</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>–</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>–</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>6</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>–</td>
<td>0</td>
</tr>
<tr>
<td>7</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>–</td>
</tr>
</tbody>
</table>
Measures and Metrics

“Which are the most important or central vertices in a network?“

Examples of
A) Degree centrality,
B) Closeness centrality,
C) Betweenness centrality,
D) Eigenvector centrality,
E) Katz centrality,
F) Alpha centrality of the same graph.

- Chapter 7: measures and metrics
- Chapter 11: matrix algorithms and graph partitioning
Degree centrality

Perhaps the simplest centrality measure in a network is the *degree centrality* that is simply equal to the *degree* of each vertex.

E.g. in a social network, individuals that have many connections to others might have
- more *influence*,
- more *access to information*,
- or more *prestige* than those individuals who have fewer connections.
1.2 Biological Background

Central Paradigm of Molecular Biology

\[ \text{DNA} \rightarrow \text{RNA} \rightarrow \text{Protein} \rightarrow \text{Phenotype (Symptoms)} \]

Central Paradigm of Structural Biology

\[ \text{Genetic Information} \rightarrow \text{Molecular Structure} \rightarrow \text{Biochemical Function} \rightarrow \text{Phenotype (Symptoms)} \]

Central Paradigm of Molecular Systems Biology

\[ \text{Genetic Information} \rightarrow \text{Molecular Structure} \rightarrow \text{Biochemical Function} \rightarrow \text{Phenotype (Symptoms)} \]

However, there exist feedback loops (transcription factors, microRNAs)
1.2 Cellular components

We will mostly use 3 different levels of description:

Inventory lists and lists of processes:
- Proteins in particular compartments
- Proteins forming macromolecular complexes
- Biomolecular interactions
- Regulatory interactions
- Metabolic reactions

Structural descriptions:
- Structures of single proteins
- Topologies of protein complexes
- Subcellular compartments

Dynamic descriptions:
- Cellular processes ranging from nanosecond dynamics for the association of two biomolecules up to processes occurring in seconds and minutes such as the cell division of yeast cells.
1.2 Biomolecules

**Macromolecules**: proteins
nucleic acids
polysaccharides
lipids.

**Building blocks** of macromolecules:
- sugars as the precursors of poly-saccharides
- amino acids as the building blocks of proteins
- nucleotides as the precursors of DNA and RNA
- fatty acids which are incorporated into lipids.

Interestingly, in biological cells, only a small number of the theoretically synthesizable macromolecules exist at a given point in time.

At any moment during a normal cell cycle, many new macromolecules need to be synthesized from their building blocks and this is meticulously controlled by the complex **gene expression machinery**.

Even during a steady-state of the cell, there exists a constant **turn-over** of macromolecules.
1.2 Biomolecules

Metabolic intermediates (metabolites):
The molecules in a cell have complex chemical structures and must be synthesized step-by-step beginning with specific starting materials that may be taken up as energy source.

In the cell, connected chemical reactions are often grouped into metabolic pathways.

Molecules of miscellaneous function:
- vitamins
- steroid or amino acid hormones
- molecules involved in energy storage (e.g. ATP)
- regulatory molecules (e.g. cyclic AMP)
- metabolic waste products such as urea.
1.2 Compartments

Organization into various compartments greatly simplifies the temporal and spatial process flow in eukaryotic cells.

At each time point during a cell cycle only a small subfraction of all potential proteins are being synthesized (and not yet degraded).

Many proteins are only available in very small concentrations, possibly with only a few copies per cell.

However, due to localizing these proteins to particular spots in the cell, e.g. by attaching them to the cytoskeleton or by partitioning them into lipid rafts, their local concentrations may be much higher.
1.2 Compartments

Compartments of a typical animal cell:

1. nucleolus
2. nucleus
3. ribosome
4. vesicle
5. rough endoplasmic reticulum (ER)
6. Golgi apparatus
7. Cytoskeleton
8. smooth ER
9. mitochondria
10. vacuole
11. cytoplasm
12. lysosome
13. centrioles
## 1.2 Organisms

<table>
<thead>
<tr>
<th>Organism</th>
<th>Length of genome [Mb]</th>
<th>Number of protein-coding genes</th>
<th>Number of RNA genes</th>
<th>Number of transporter proteins</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Prokaryotes</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mycoplasma genitalium G37</td>
<td>0.6</td>
<td>476</td>
<td>43</td>
<td>53</td>
</tr>
<tr>
<td>Bacillus subtilis BSN5</td>
<td>4.2</td>
<td>4145</td>
<td>113</td>
<td>552</td>
</tr>
<tr>
<td>Escherichia coli (E. coli) APEC01</td>
<td>4.6</td>
<td>4890</td>
<td>93</td>
<td>665</td>
</tr>
<tr>
<td><strong>Eukaryotes</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Saccharomyces</td>
<td>1.3</td>
<td>6002</td>
<td>425</td>
<td>341</td>
</tr>
<tr>
<td>Cerevisae S288C</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Drosophila melanogaster</td>
<td>12</td>
<td>13929</td>
<td>3209</td>
<td>662</td>
</tr>
<tr>
<td>Caenorhabditis elegans</td>
<td>100.2</td>
<td>20093</td>
<td>24969</td>
<td>669</td>
</tr>
<tr>
<td>Homo sapiens</td>
<td>3150</td>
<td>20338</td>
<td>19201</td>
<td>1467</td>
</tr>
</tbody>
</table>
1.3 Cellular Pathways

After Babu (2004)
1.3 Major Metabolic Pathways

static connectivity <=> dynamic response to external conditions <=> different states during the cell cycle
1.3 Cellular Pathways

Glycolysis pathway of *E. coli*
KEGG database
1.4 Ontologies & Databases

- Gene Ontology (chapter 8)
- KEGG (http://www.genome.jp/kegg/)
- Reactome
- BRENDA (https://www.brenda-enzymes.org/)
- DAVID (https://david.ncifcrf.gov/)
- Protein Databank (www.rcsb.org)
Summary

What you learned today:

=> networks are everywhere

⇒ how to get the "Schein" for BI3

⇒ What is the lecture content

⇒ Basic biological background

Next lectures:

- Random graphs vs. scale-free networks (assignments 1 + 2)

- Structures of protein complexes
Additional slides (not used)
Towards Eigenvector Centrality

Let us start by defining the **centrality** of vertex $x_i$ as the sum of the centralities of all its neighbors:

$$x_i' = \sum_j A_{ij} x_j$$

where $A_{ij}$ is an element of the adjacency matrix.

(This equation system must be solved recursively until convergence.)

Remember the multiplication of a matrix with a vector below …

$$A = \begin{pmatrix} a & b & c \\ p & q & r \\ u & v & w \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} x \\ y \\ z \end{pmatrix},$$

$$AB = \begin{pmatrix} a & b & c \\ p & q & r \\ u & v & w \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} ax + by + cz \\ px + qy + rz \\ ux + vy + wz \end{pmatrix},$$
Towards Eigenvector Centrality

Let us start by defining the **centrality** of vertex $x_i$ as the sum of the centralities of all its neighbors:

$$
x_i' = \sum_j A_{ij} x_j
$$

where $A_{ij}$ is an element of the adjacency matrix.

We can also write this expression in matrix notation as

$$
x' = A \cdot x
$$

where $x$ is the vector with elements $x_i$.

Repeating this process to make better estimates gives after $t$ steps the following vector of centralities:

$$
x(t) = A^t \cdot x(0)
$$
Eigenvector Centrality

Now let us write $x(0)$ as a linear combination of the eigenvectors $v_i$ of the (quadratic) adjacency matrix\(^1\)

$$x(0) = \sum_i c_i v_i \quad \text{with suitable constants } c_i$$

Then

$$x(t) = A^t \sum_i c_i v_i$$

Because $v_i$ are eigenvectors of $A, A v_i = k_i v_i$ with the eigenvalue $k_i$.

Let $k_1$ be the largest eigenvector.

$$x(t) = A^t \sum_i c_i v_i = \sum_i c_i k_i^t v_i = k_1^t \sum_i c_i \left[\frac{k_i}{k_1}\right]^t v_i$$

Since $k_i / k_1 < 1$ for all $i \neq j$, all terms in the sum decay exponentially as $t$ becomes large, only the term with $i = j$ remains unchanged.

In the limit $t \to \infty$, we get for the centrality vector $x(t) = c_1 k_1^t v_1$

\(^1\) Remember from linear algebra that a quadratic matrix with full rank can be diagonalized.
Eigenvector Centrality

This limiting vector of the eigenvector centralities is simply proportional to the leading eigenvector of the adjacency matrix.

Equivalently, we could say that the centrality \( x \) satisfies

\[
A \ x = k_1 \ x
\]

This is the \textit{eigenvector centrality} first proposed by Bonacich (1987).

The centrality \( x_i \) of vertex \( i \) is proportional to the sum of the centralities of its neighbors:

\[
x_i = k_1^{-1} \sum_j A_{ij} x_j
\]

Divide above eq. by \( k_1 \)

This has the nice property that the centrality can be large either because a vertex has many neighbors or because it has important neighbors with high centralities (or both).
Problems of the Eigenvector Centrality

The eigenvector centrality works best for undirected networks.

For directed networks, certain complications can arise.

In the figure on the right, vertex A will have eigenvector centrality zero.

Hence, vertex B will also have centrality zero.

Figure 7.1: A portion of a directed network. Vertex A in this network has only outgoing edges and hence will have eigenvector centrality zero. Vertex B has outgoing edges and one incoming edge, but the incoming one originates at A, and hence vertex B will also have centrality zero.
Katz Centrality

One solution to the issues of the Eigenvector Centrality is the following:

We simply give each vertex a small amount of centrality “for free”, regardless of its position in the network or the centrality of its neighbors.

→ we define \( x_i = \alpha \sum_j A_{ij} x_j + \beta \) where \( \alpha \) and \( \beta \) are positive constants.

In matrix terms, this can be written as \( \mathbf{x} = \alpha \mathbf{A} \mathbf{x} + \beta \mathbf{1} \)

where \( \mathbf{1} \) is the vector \( (1,1,1,...)^T \). By rearranging for \( \mathbf{x} \) we find

\[
\begin{align*}
\mathbf{I} \mathbf{x} - \alpha \mathbf{A} \mathbf{x} &= \beta \mathbf{1} \\
(\mathbf{I} - \alpha \mathbf{A}) \mathbf{x} &= \beta \mathbf{1} \\
(\mathbf{I} - \alpha \mathbf{A})^{-1} (\mathbf{I} - \alpha \mathbf{A}) \mathbf{x} &= (\mathbf{I} - \alpha \mathbf{A})^{-1} \beta \mathbf{1} \\
\mathbf{x} &= \beta (\mathbf{I} - \alpha \mathbf{A})^{-1} \mathbf{1}
\end{align*}
\]

When setting \( \beta = 1 \), we get the **Katz centrality** (1953) \( \mathbf{x} = (\mathbf{I} - \alpha \mathbf{A})^{-1} \mathbf{1} \)
Computing the Katz Centrality

The Katz centrality differs from the ordinary eigenvector centrality by having a free parameter $\alpha$, which governs the balance between the eigenvector term and the constant term.

However, inverting a matrix on a computer has a complexity of $O(n^3)$ for a graph with $n$ vertices.

This becomes prohibitively expensive for networks with more than 1000 nodes or so.

It is more efficient to make an initial guess of $x$ and then repeat

$$x' = \alpha Ax + \beta I$$

many times. This will converge to a value close to the correct centrality.

A good test for convergence is to make two different initial guesses and run this until the resulting centrality vectors agree within some small threshold.
Towards PageRank

The Katz centrality also has one feature that can be undesirable.

If a vertex with high Katz centrality has edges pointing to many other vertices, then all those vertices also get high centrality.

E.g. if a Wikipedia page points to my webpage, my webpage will get a centrality comparable to Wikipedia!

But Wikipedia of course also points to many other websites, so that its contribution to my webpage “should” be relatively small because my page is only one of millions of others.

-> we will define a variation of the Katz centrality in which the centrality I derive from my network neighbors is proportional to their centrality divided by their out-degree.
PageRank

This centrality is defined by

\[ x_i = \alpha \sum_j A_{ij} \frac{x_j}{k_{j \text{out}}} + \beta \]

At first, this seems problematic if the network contains vertices with zero outdegree.

However, this can easily be fixed by setting \( k_{j \text{out}} = 1 \) for all such vertices.

In matrix terms, this equation becomes

\[ \mathbf{x} = \alpha \mathbf{A} \mathbf{D}^{-1} \mathbf{x} + \beta \mathbf{1} \]

where \( \mathbf{1} \) is the vector \((1, 1, 1, \ldots)\)^T and \( \mathbf{D} \) the diagonal matrix with \( D_{ij} = max(k_{j \text{out}}, 1) \)
PageRank

By rearranging we find that

\[ x = \beta \left( I - \alpha A D^{-1} \right)^{-1} I \]

Because \( \beta \) plays the same unimportant role as before, we will set \( \beta = 1 \).

Then we get

\[ x = \left( I - \alpha A D^{-1} \right)^{-1} I = D \left( D - \alpha A \right)^{-1} I \]

This centrality measure is commonly known as PageRank, using the term used by Google.

PageRank is one of the ingredients used by Google to determine the ranking of the answers to your queries.

\( \alpha \) is a free parameter and should be chosen less than 1. (Google uses 0.85).
Closeness centrality

An entirely different measure of centrality is provided by the **closeness centrality**.

Suppose $d_{ij}$ is the length of a geodesic path (i.e. the shortest path) from a vertex $i$ to another vertex $j$. Here, length means the number of edges along the path.

Then, the mean **geodesic distance** from $i$, averaged over all vertices $j$ in the network is

$$l_i = \frac{1}{n} \sum_j d_{ij}$$

The mean distance $l_i$ is not a centrality measure in the same sense as the other centrality measures.

It gives *low* values for more central vertices and *high* values for less central ones.
Closeness centrality

The inverse of $l_i$ is called the **closeness centrality** $C_i$

$$C_i = \frac{1}{l_i} = \frac{n}{\sum_j a_{ij}}$$

It has become popular in recent years to rank **film actors** according to their closeness centrality in the network of who has appeared in films with who else.

Using data from www.imdb.com the largest component of the network includes more than 98 % of about half a million actors.
Closeness centrality

The highest closeness centrality of any actor is 0.4143 for Christopher Lee.

The second highest centrality has Donald Pleasence (0.4138).

The lowest value has the Iranian actress Leia Zanganeh (0.1154).

→ the closeness centrality values are crammed in a very small interval [0,0.4143]

Other centrality measures including degree centrality and eigenvector centrality typically don‘t suffer from this problem. They have a wider dynamic range.

Pictures from wikipedia