A methodology to evaluate the evolution of networks using topological data analysis

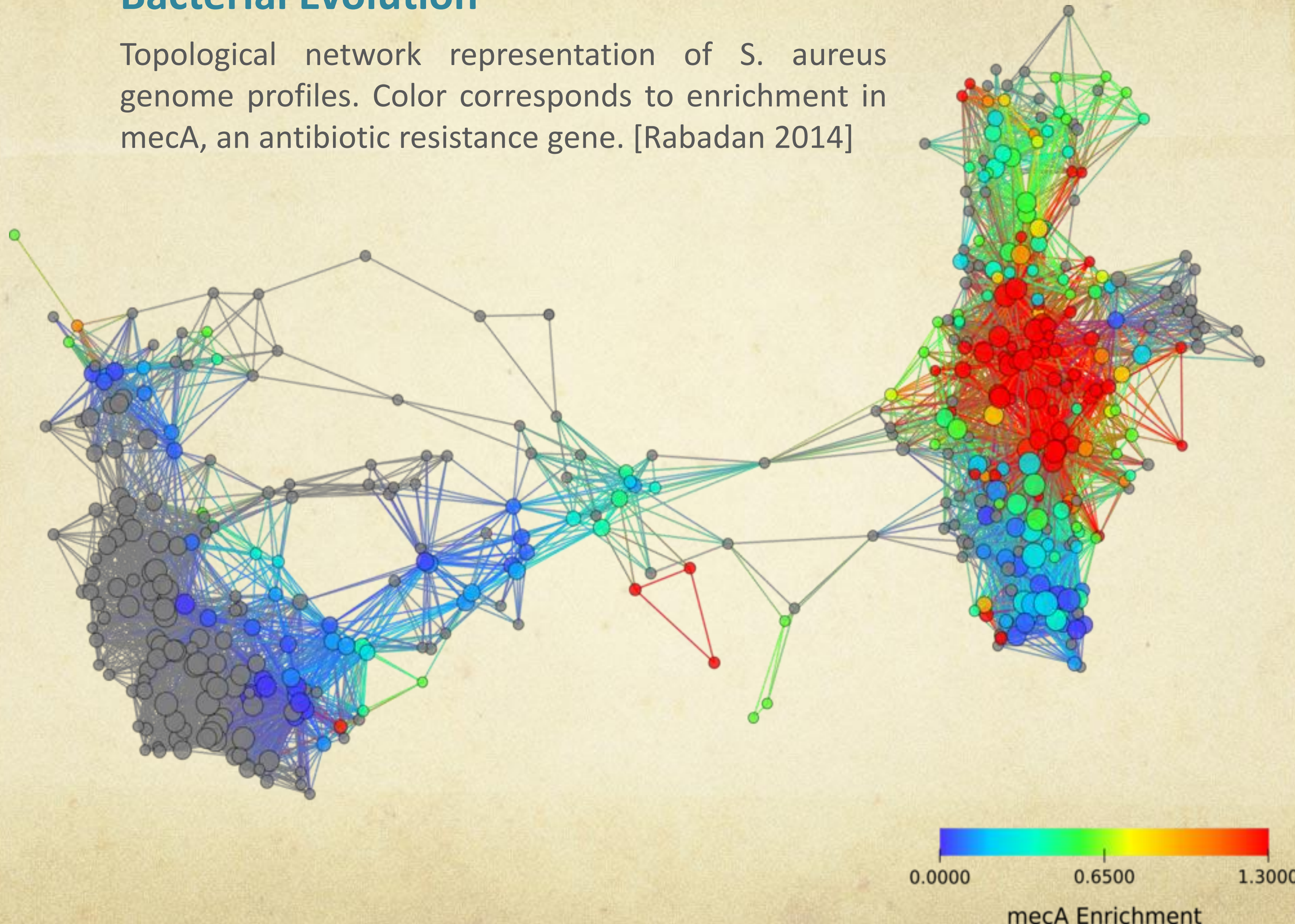
Joao Pita Costa and Tihana Galinac Grbac
University of Rijeka

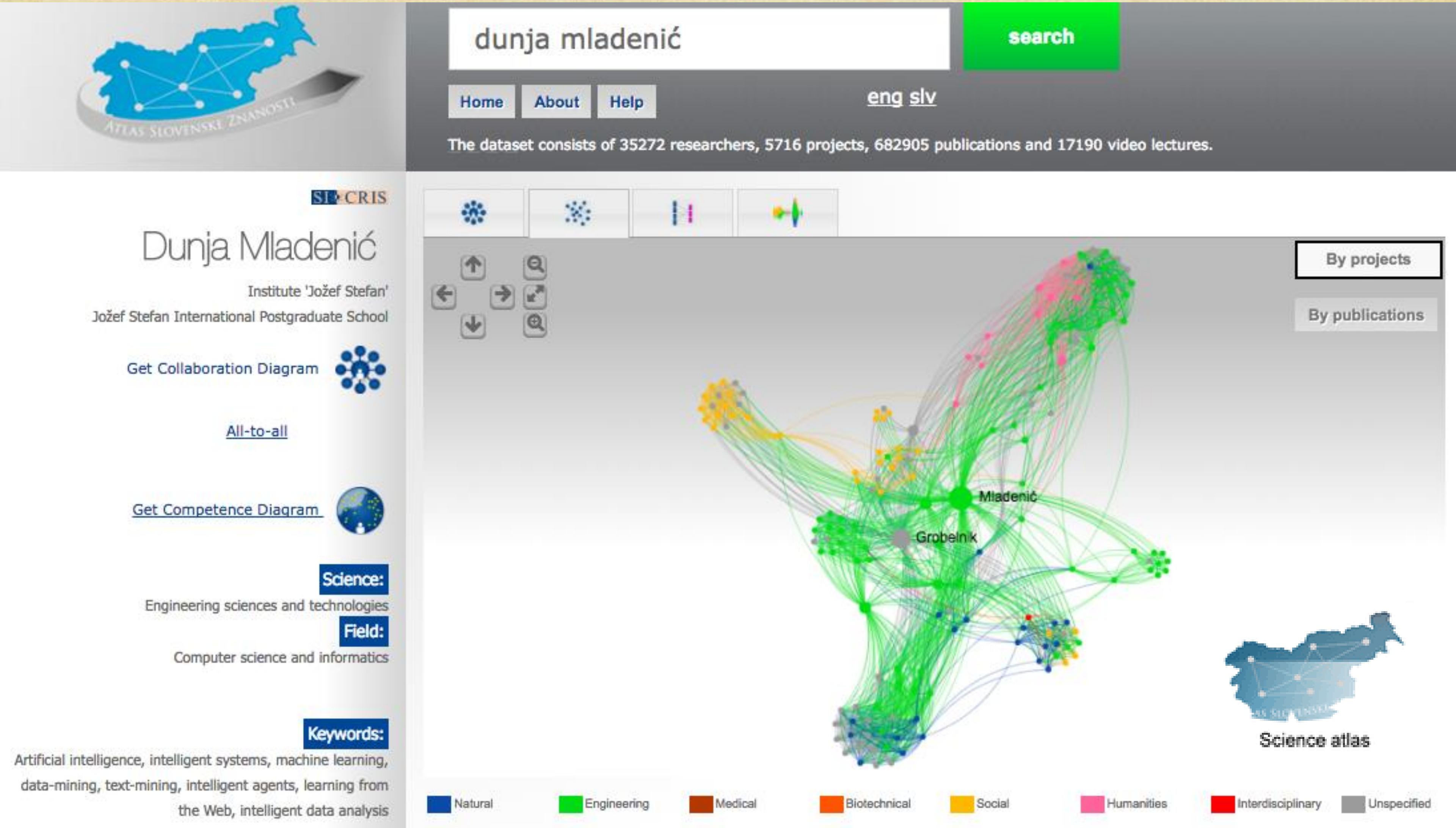
Ljubljana, October 9, SIKDD 2017

SiKDD 2017

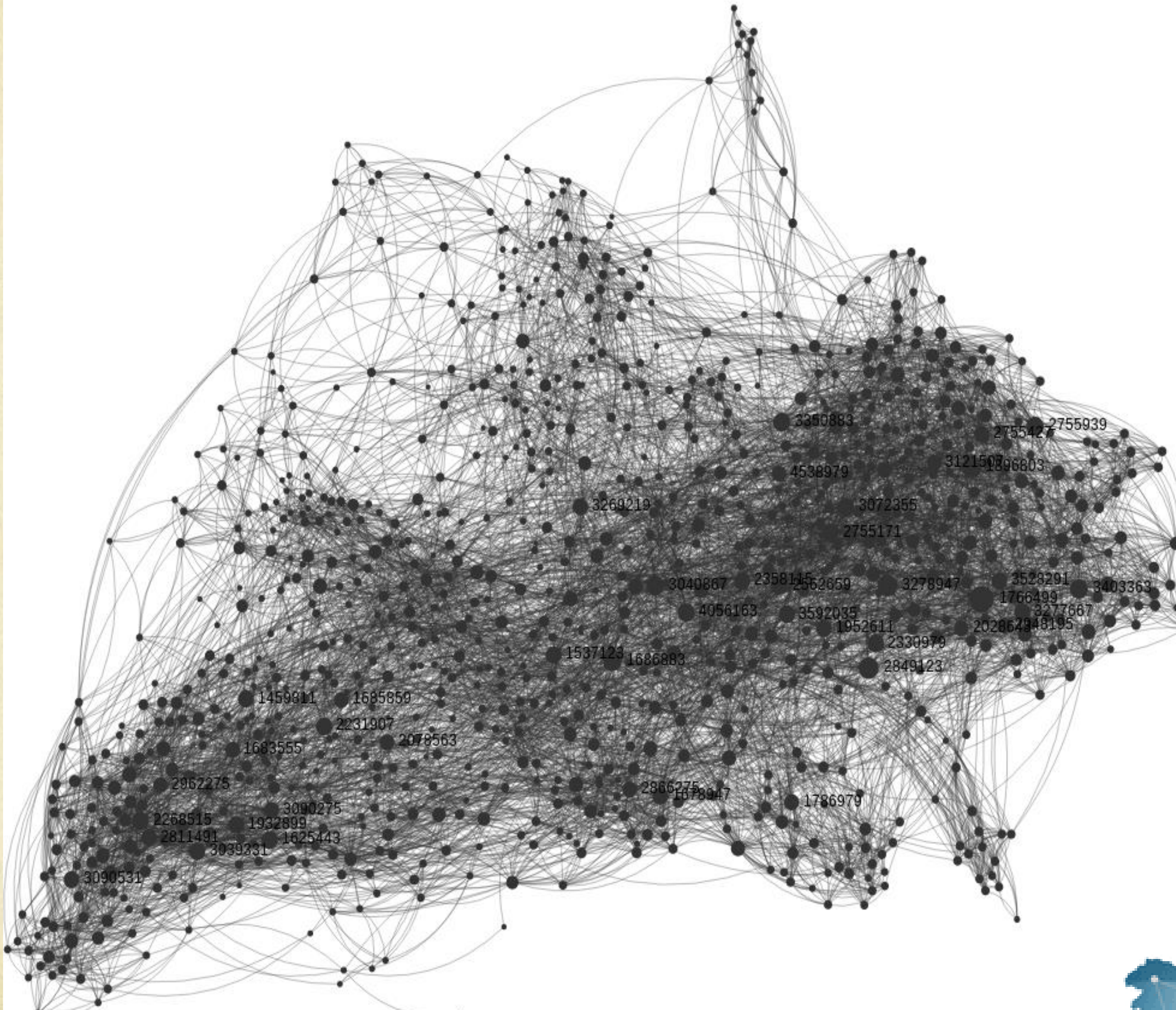
Bacterial Evolution

Topological network representation of S. aureus genome profiles. Color corresponds to enrichment in mecA, an antibiotic resistance gene. [Rabadan 2014]

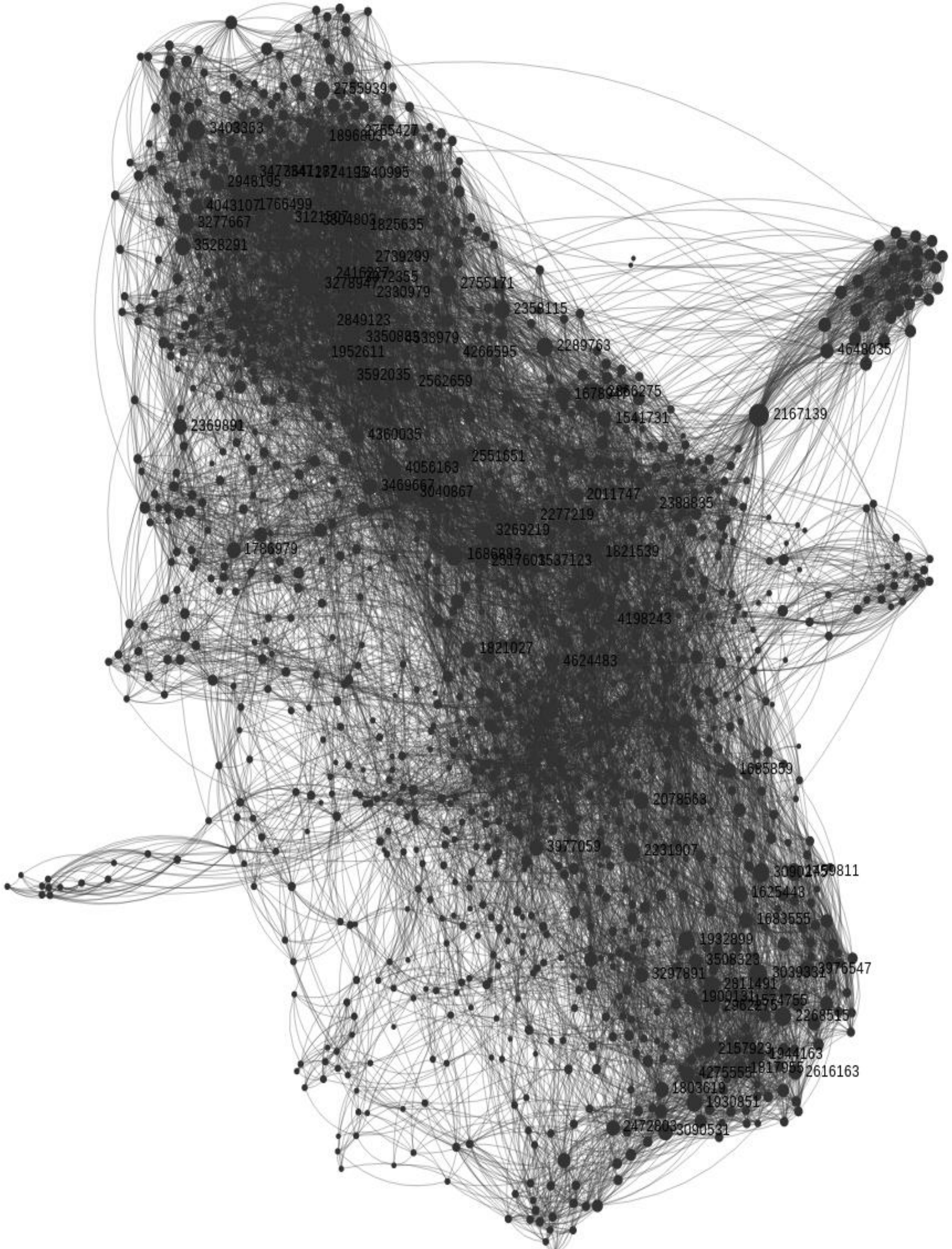




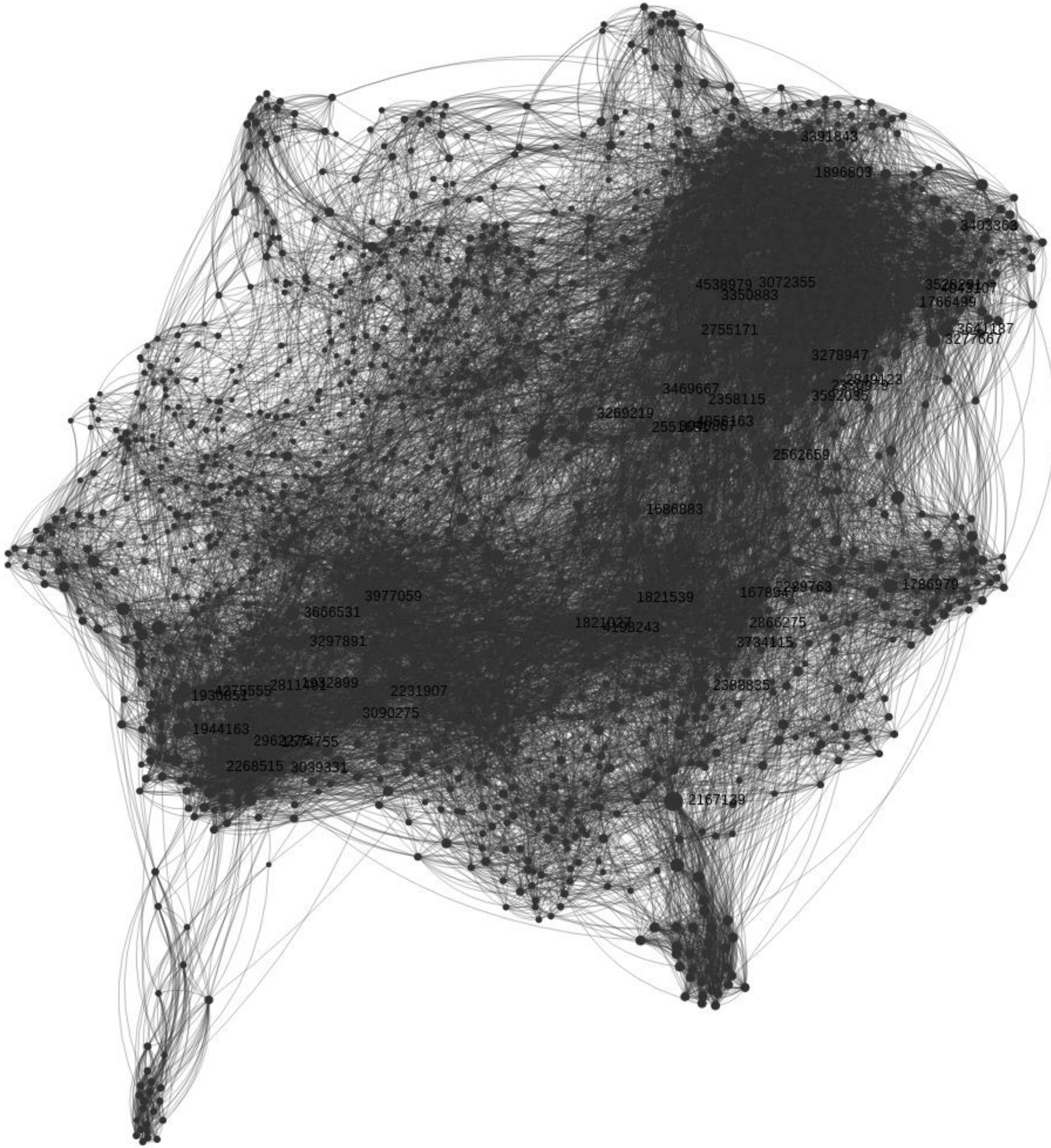
The portal **ScienceAtlas** permits the user to explore scientific collaborations between authors. This is based on the data from the Slovenian Research Agency ARRS, and exhibits a real example of evolving networks: the collaborative research networks of the researchers in the ARRS database.



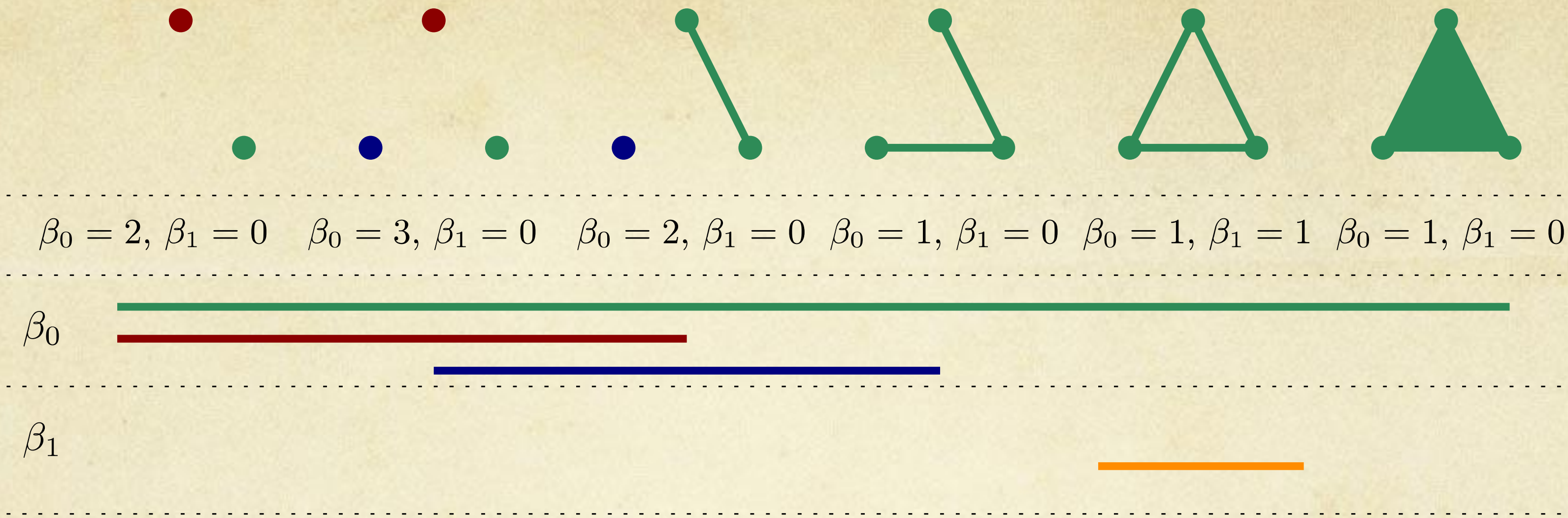
Science atlas



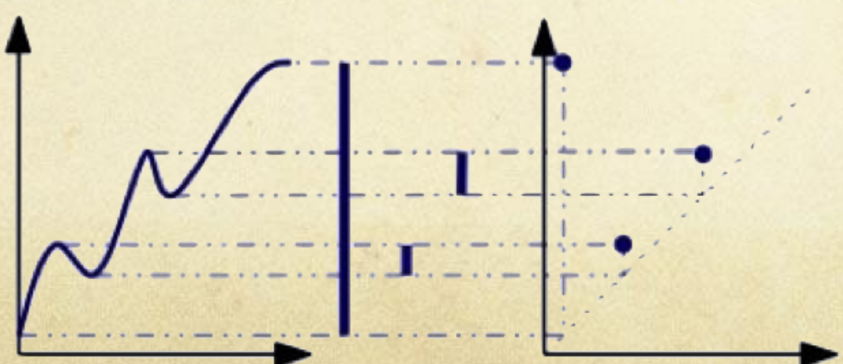
Science atlas

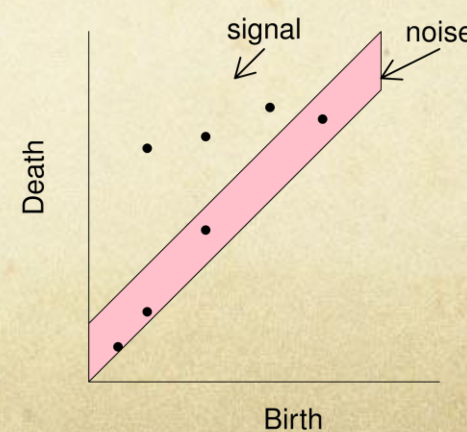


Science atlas

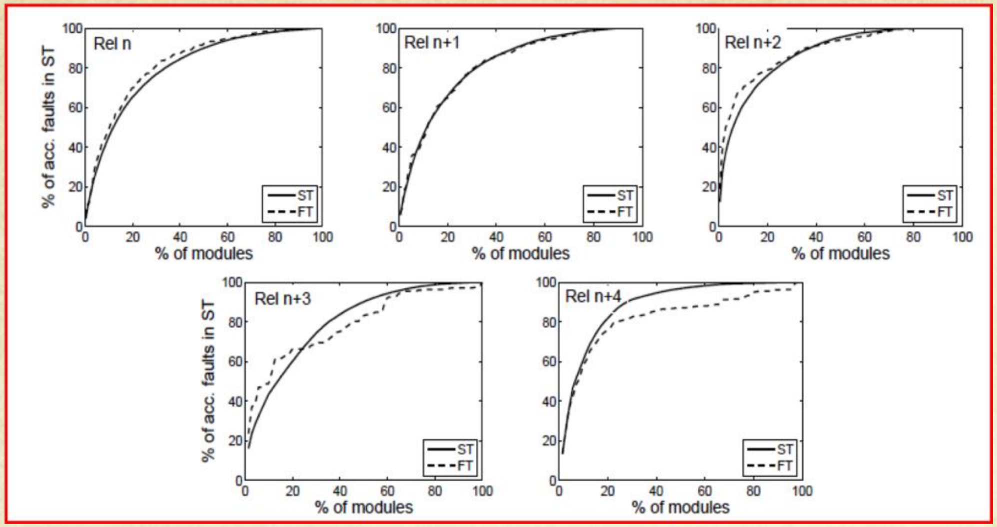


Topological data analysis - **TDA** - is interested in problems relating to **nonlinear systems**, **large scale data** and development of more **accurate models**. Persistent homology permits to consider the homology of the filtered simplicial complex at all times during the filtration. Persistent homology identifies a global structure by inferring **high-dimensional structure from low-dimensional representations** and studying **properties of a (often) continuous space by the analysis of a discrete sample of it.**

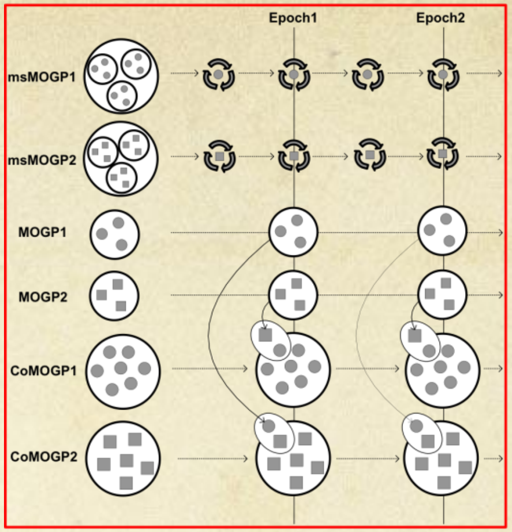




Reliability growth analysis:

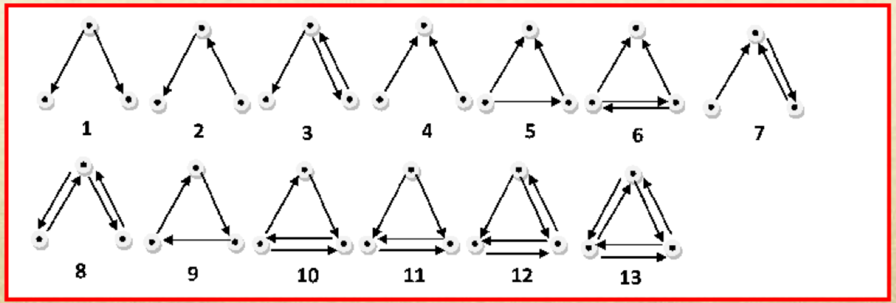


Co-evolutionary MOGP:



Evolving complex software systems - **EVOSOFT** - focus the influence of **abstract software structures** and **local system properties** in **fault distributions**, affecting mission critical system properties, among which availability and reliability and to develop innovative approaches for **smart management of their operation and evolution**. New findings will open opportunities in many fields, especially in **complex systems theory and its applications**, thus interacting with a wide spectrum of sciences, from natural sciences such as biomedicine to social sciences.

System complex structure motifs:



RITEH

SVEUČILIŠTE U RIJECI TEHNIČKI FAKULTET

hrzz
Hrvatska zaklada za znanost

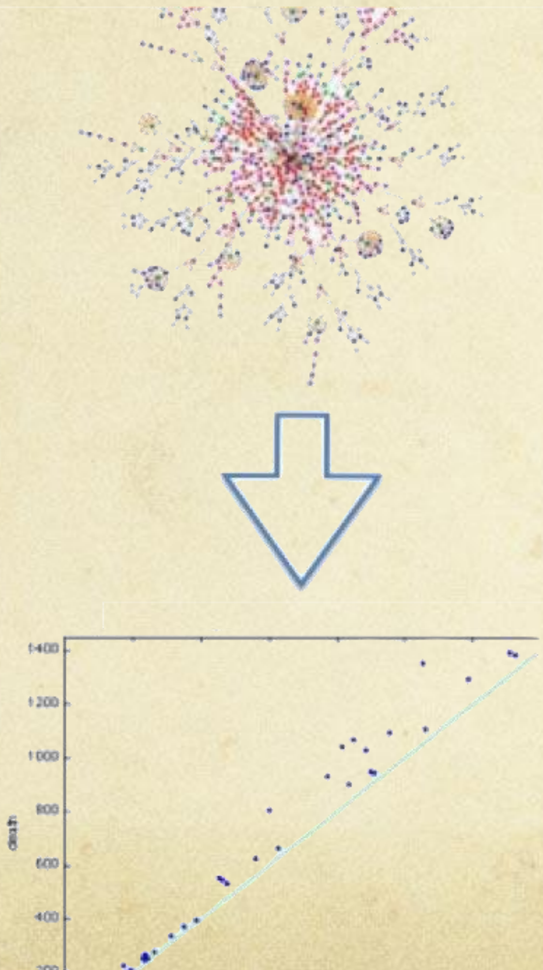
Network Data
Adjacency Matrix

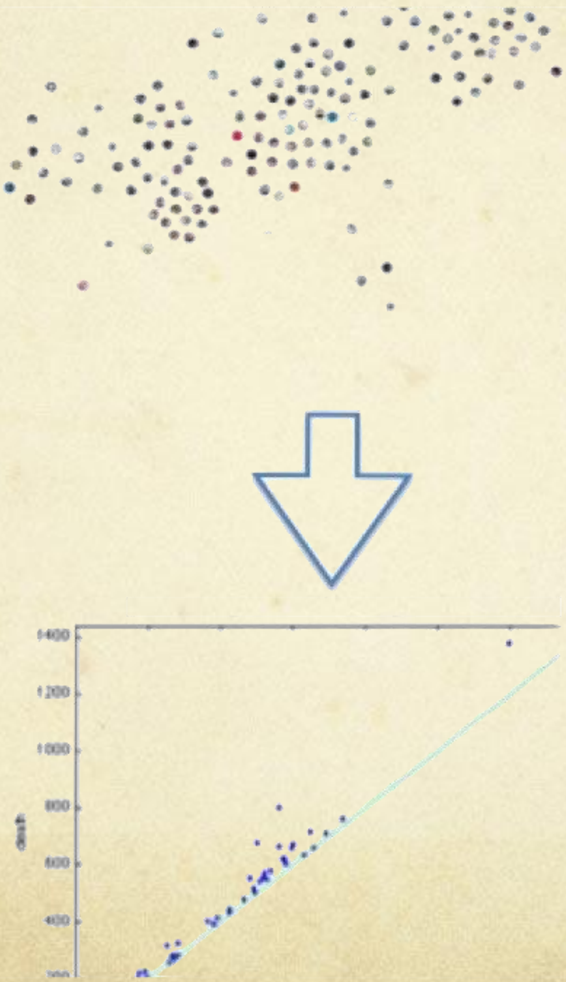
→ Persistent Homology Computation →

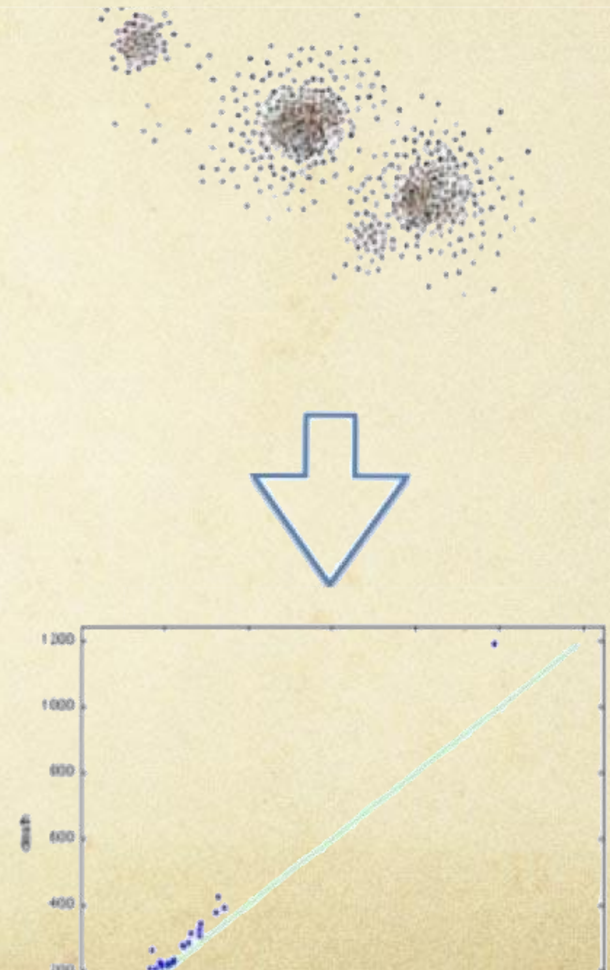
Encoded Topology
Persistence Diagram

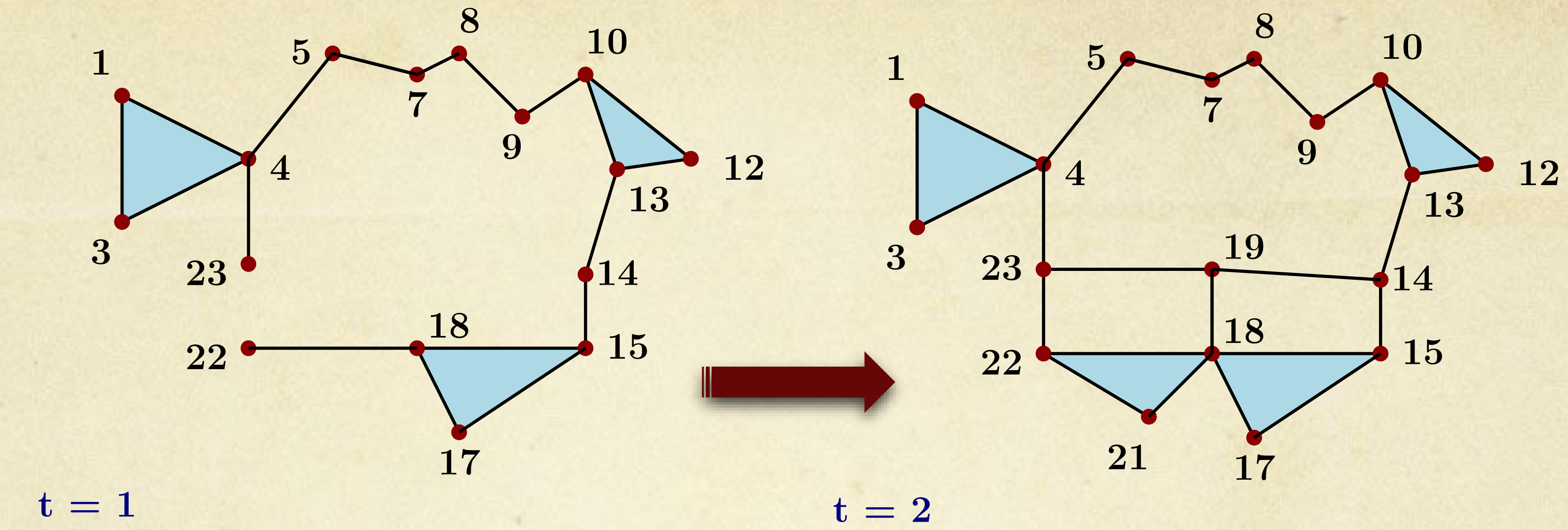
→ Bottleneck Distance Computation →

Absolute Distance
Bottleneck Distance

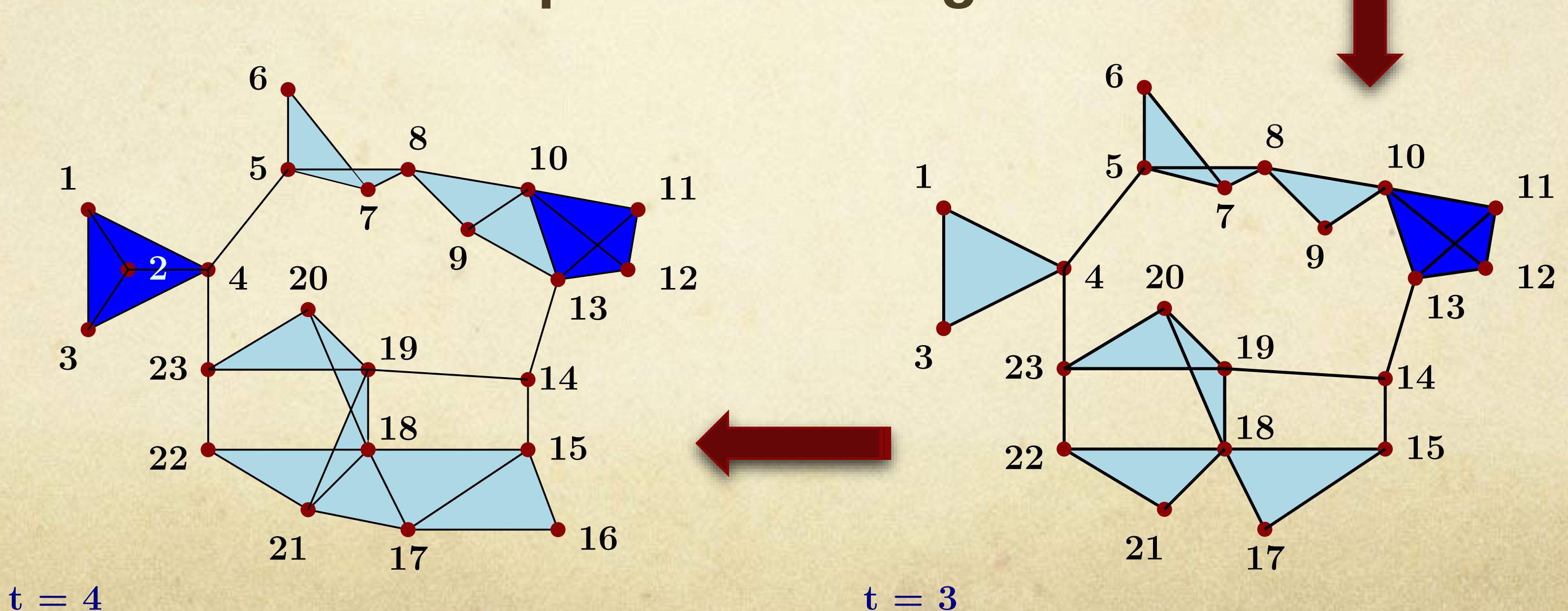


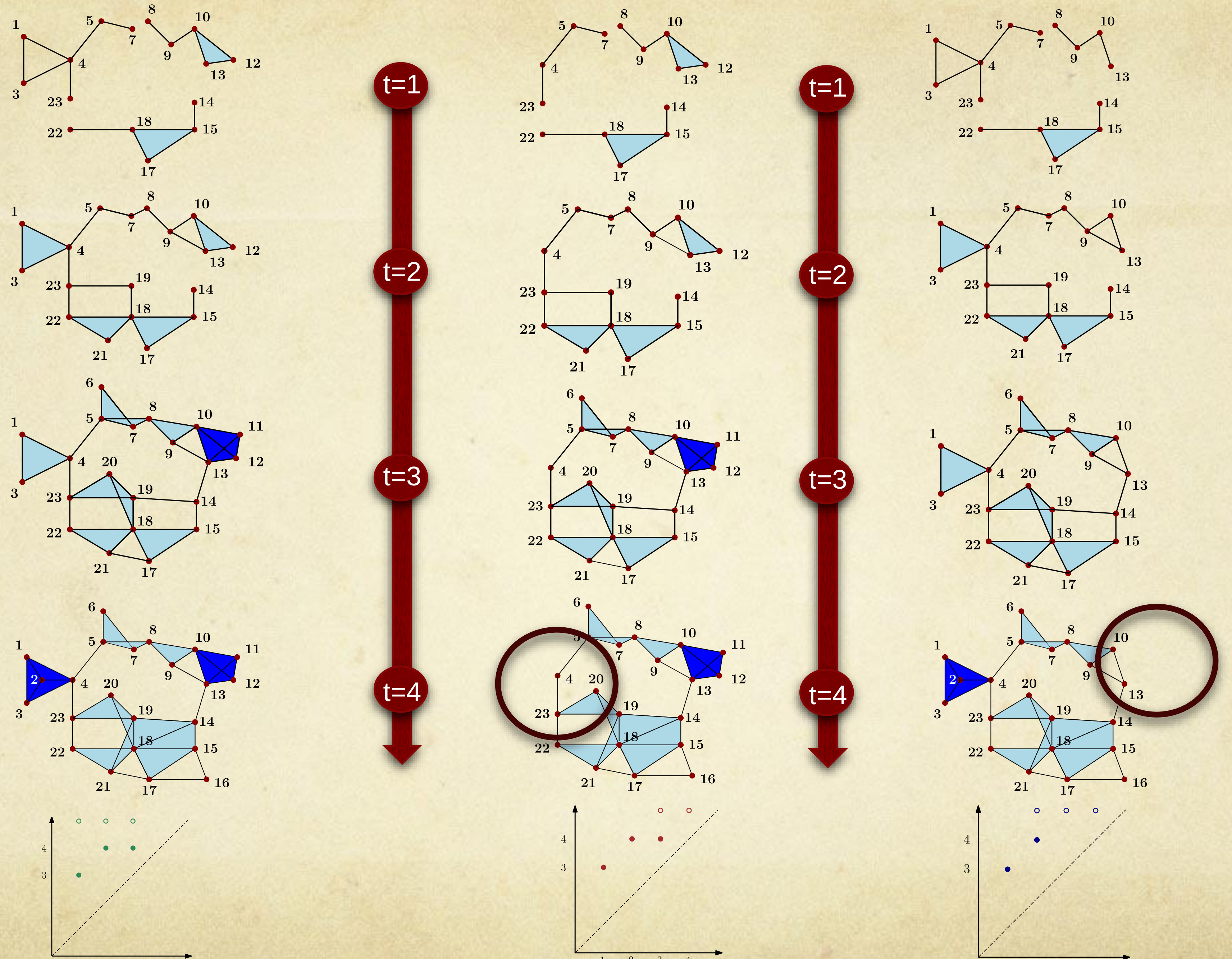






Artificial example of evolving network

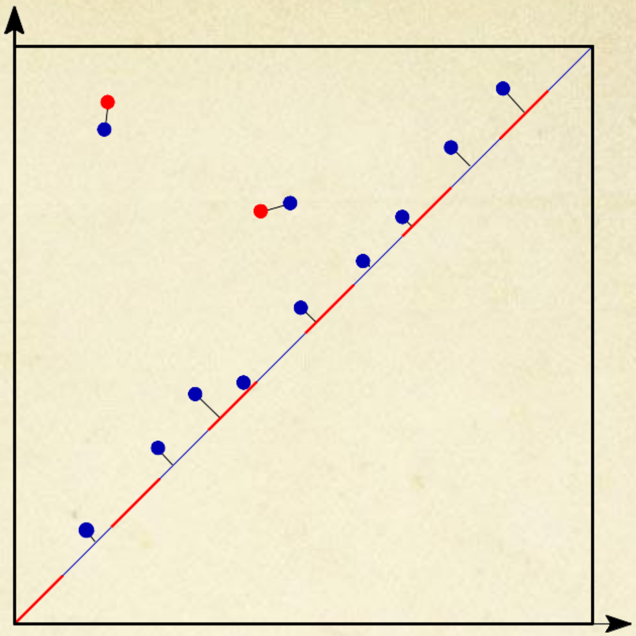




evolvnet A

evolvnet B

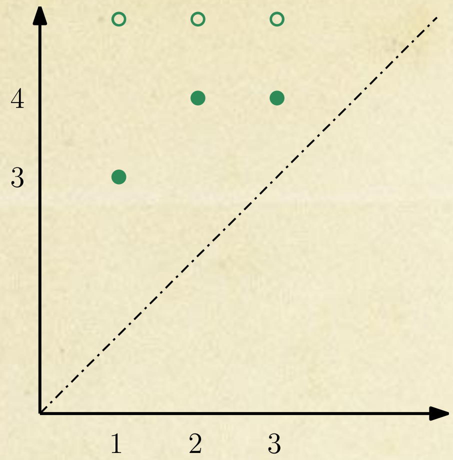
evolvnet C



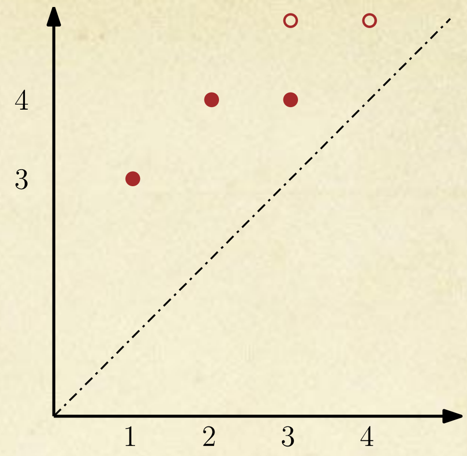
The **bottleneck distance** between X = Dgm(f) and Y = Dgm(g) is defined as

$$d_B(X,Y)=\inf_{\eta} \sup_{x\in X} \|x-\eta(x)\|_\infty$$

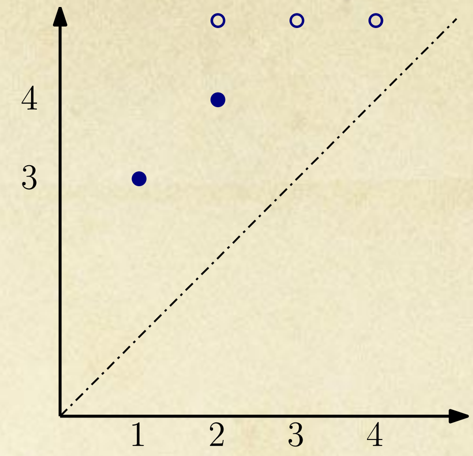
where the infimum is taken over all bijections \eta from X to Y. Each point with multiplicity k in a multiset is interpreted as k individual points, and the bijection is interpreted between the resulting sets.



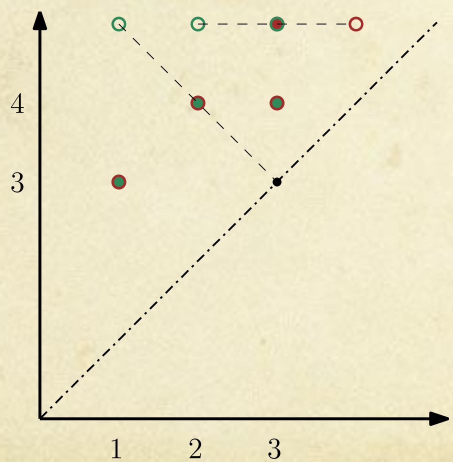
evolvnet A



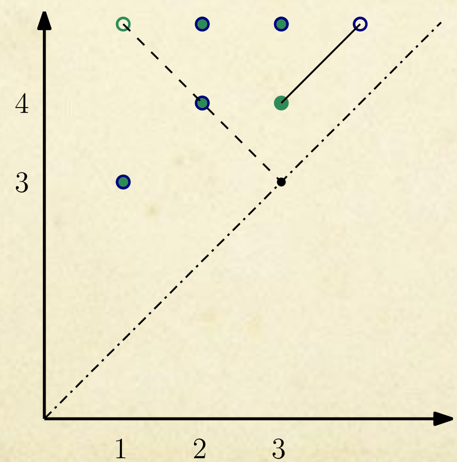
evolvnet B



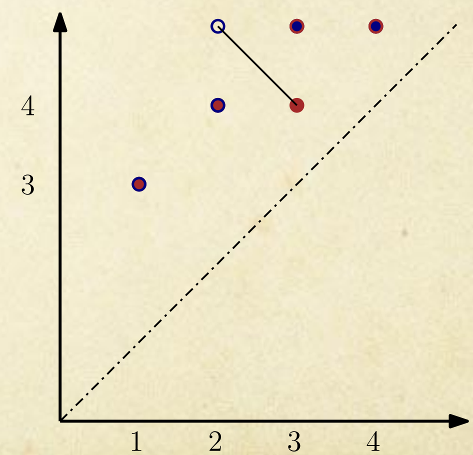
evolvnet C



d(A,B)=0

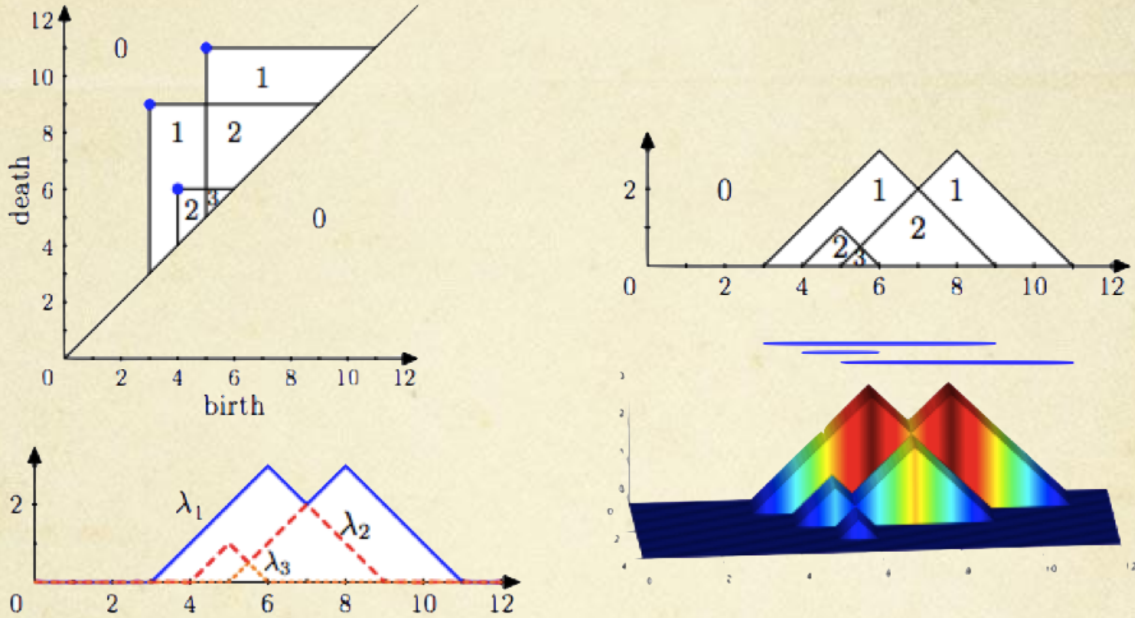


d(A,C)=1



d(B,C)=1

Persistence landscapes are techniques of TDA that permit us to measure the pairwise distance between persistence diagrams at several different levels.

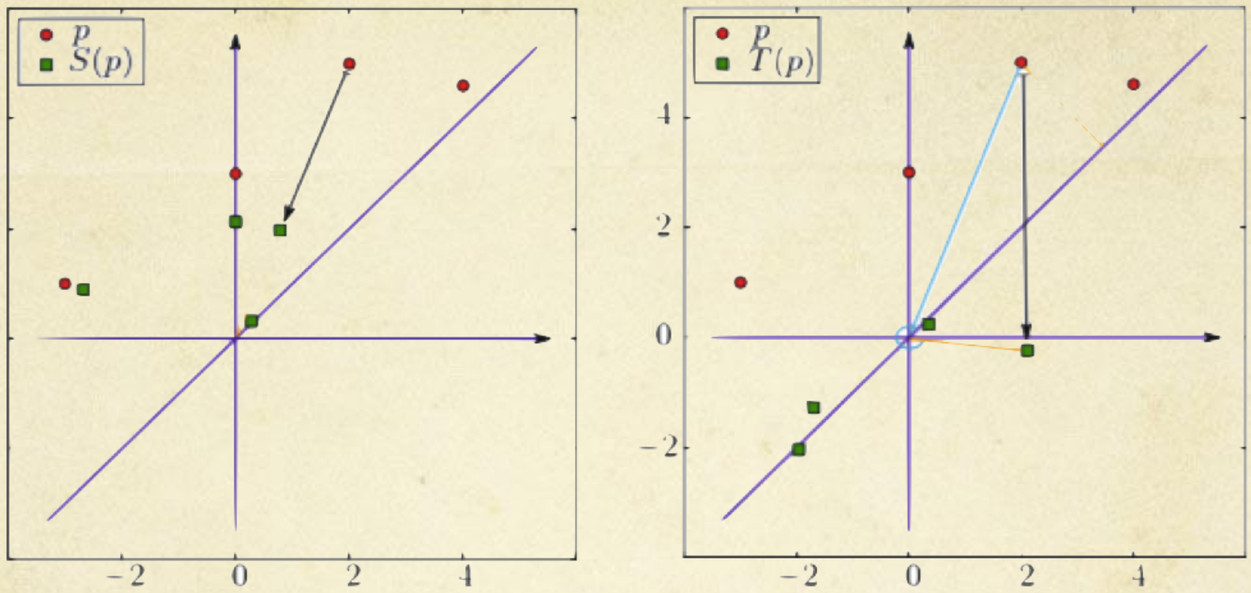


The **persistence landscape** is a function $\lambda : N \times R \to \underline{R}$, where $\underline{R} = [-\infty,\infty]$. Alternatively, it may be thought of as a sequence of functions $\lambda_k : R \to R$, where $\lambda_k(t) = \lambda(k, t)$. Define:

$$\lambda_k(t) = \sup(m \geq 0 \mid \beta^{t-m,t+m} \geq k),$$

with $\lambda : R^2 \to R$ given by:

$$\lambda(m, h) = \beta^{m-h,m+h} \text{ if } h \geq 0 \text{ and } \lambda(m, h) = 0 \text{ otherwise}$$



Given persistence diagram D described by its points $p_1 = (u_1, v_1), \dots , p_s = (u_s, v_s)$ with multiplicities $r_1,\dots,r_s$, the method **complex of vectors** considers complex numbers $z_1=u_1+iv_1,\dots, z_s = u_s+iv_s$ and associates to D the complex polynomial $f_D(t) = \prod_{j=1}^{s}(t - z_j)^{r_j}$ where r_j is the multiplicity of p_j

$p_A=(t-1-3i)(t-2-4i)(t-3-4i)=p_B$ and $p_C=(t-1-3i)(t-2-4i)^2$

Thank You : Hvala : Obrigado

jpitadacosta@riteh.hr — Joao Pita Costa

tihana.galinac@riteh.hr — Tihana Galinac Grbac