Coarse-graining dynamical networks and analysis of data collected in the form of graphs*

Karthikeyan Rajendran and Ioannis G. Kevrekidis

Department of Chemical and Biological Engineering

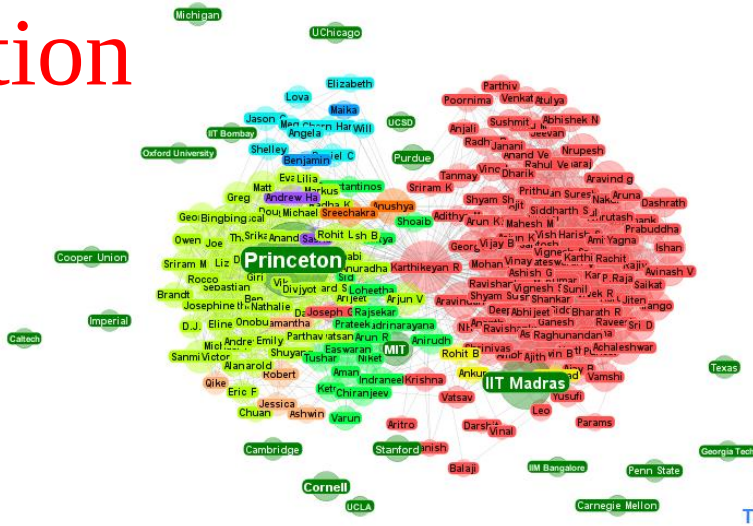
Princeton University, Princeton, NJ.

**** The work was supported by DTRA and the US DOE.***

Introduction

Complex networks

- Examples
  - Engineered (Internet)
  - Social (Facebook)
  - Biological (Metabolic networks)

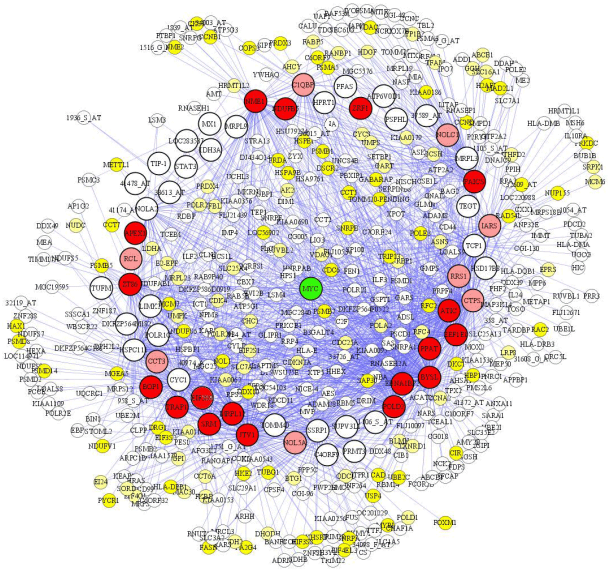


Social network (Facebook)

Motivation for coarse-graining

Dynamical models

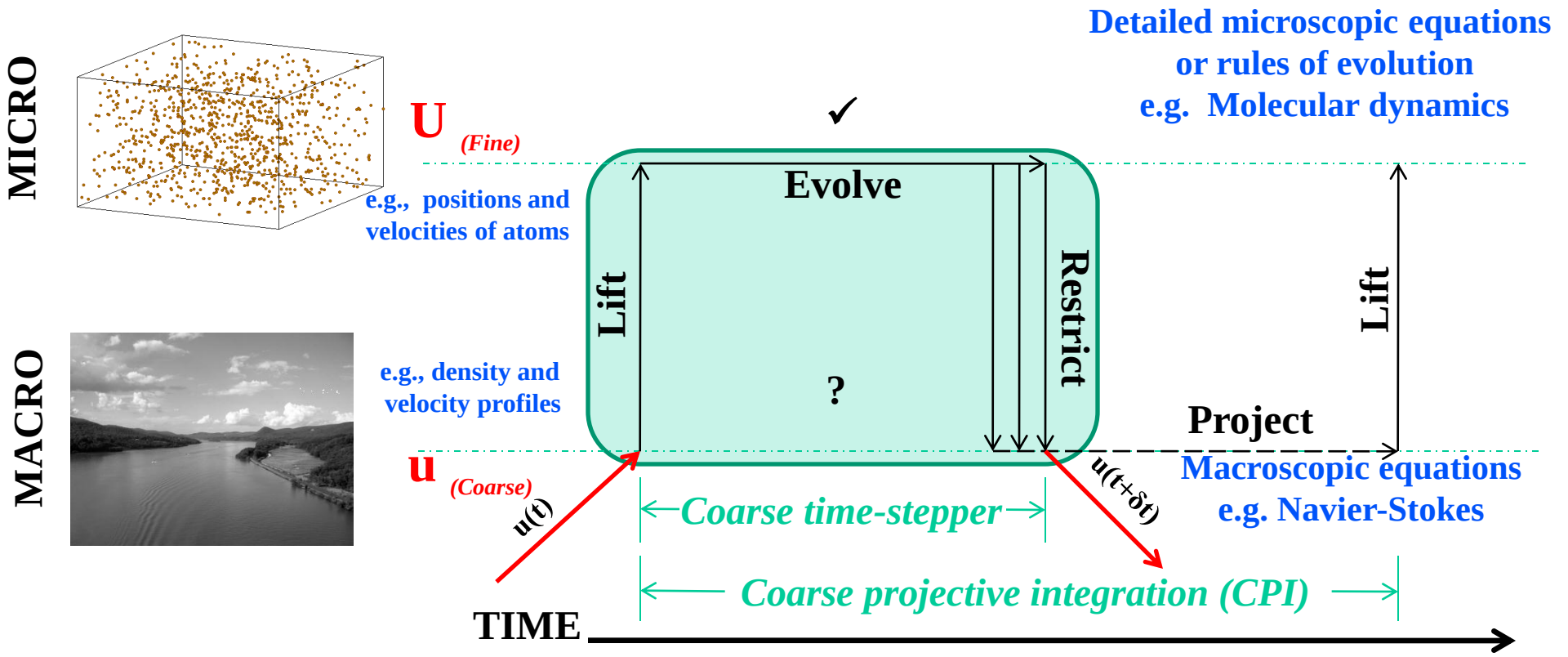
- Microscopic rules of evolution
- Use of coarse-grained models:
  - Identify the role of structure of the network in its dynamics



Metabolic network

Metabolic network (from http://www.di.unipi.it/~braccia/ToolCode/)

Equation-free (EF) approach



- Coarse time-stepper – black box code, substitute for macroscopic eqns.
  - Can be used in coarse projective integration (CPI), bifurcation etc.
- Choices for good coarse variables: Heuristic?

Reference: Kevrekidis, I. G., C.W. Gear, et al. (2004). "Equation-free:The computer-aided analysis of complex multiscale systems." Aiche Journal 50(7): 1346-1355.

Dynamics on networks

Problems where variables associated with nodes on a STATIC network evolve based on the specified network structure

Goal: Identify coarse variables to capture the effect of FEATURES of the network on features of the solutions

Illustrative example: Network of coupled oscillators

(Kuramoto model)

Reference: Kuramoto,Y., Chemical oscillations, waves, and turbulence, Berlin ; New York: Springer-Verlag, 1984.

Phase oscillators (synchronization)

- Phases, θ_i of oscillators
- Het. frequencies, ω_i

$$\dot{\theta}_i = \omega_i + F\left(\theta_{\forall j \neq i} - \theta_i\right).$$

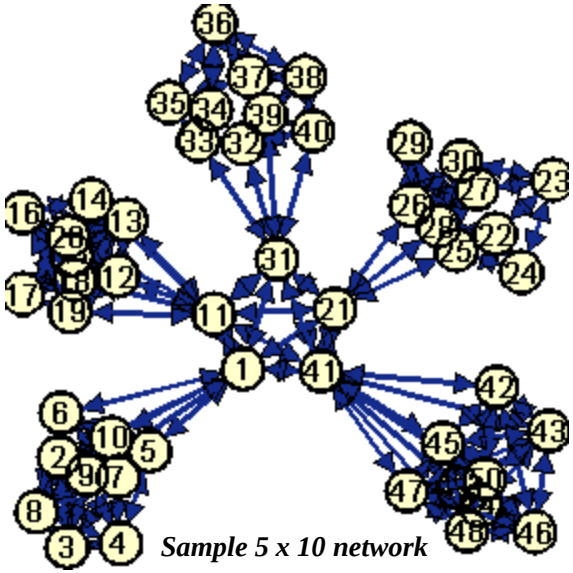
Kuramoto model[1] on a network:

$$\frac{d\theta_i}{dt} = \omega_i + K \sum_j A_{ij} \sin\left(\theta_j - \theta_i\right).$$

A – Adjacency matrix

- K is the coupling strength
- Networks constructed to facilitate separation of timescales

1- Y. Kuramoto, Chemical oscillations, waves, and turbulence, Berlin ; New York: Springer-Verlag, 1984.



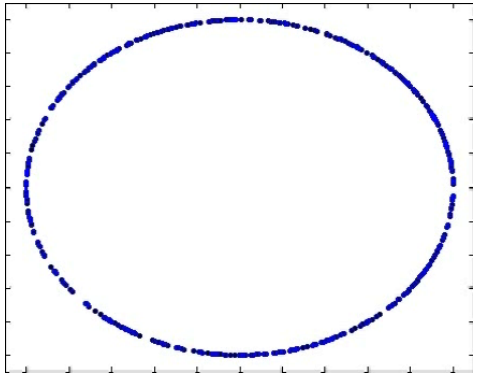
Sample 5 x 10 network

5 communities with 10 members each
Heterogeneous communities
Watts-Strogatz model
Varying average degrees
Varying rewiring probabilities
Leaders connected by a complete network

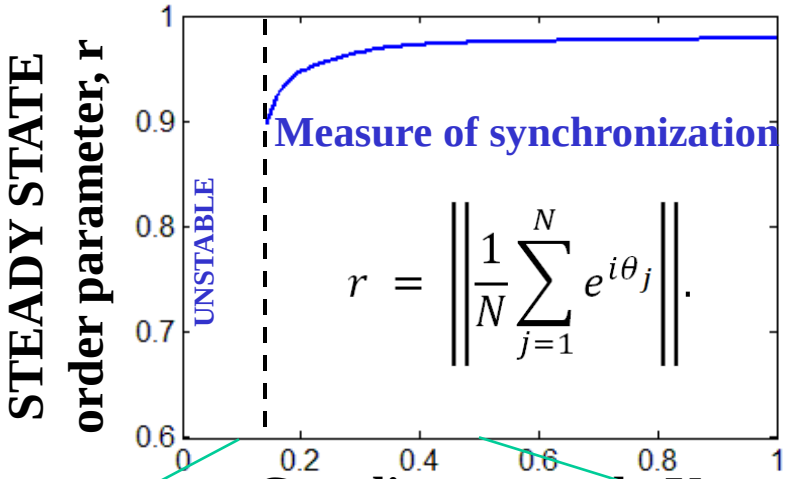
***Phys. Rev. E*, 84, 036708 (2011)**

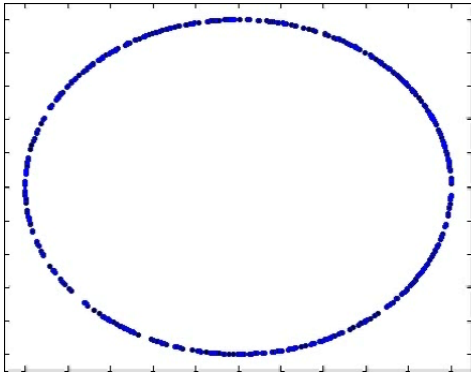
Dynamics at different coupling strengths

500 oscillators; 10 x 50 network; w ~ N(0,1/15)

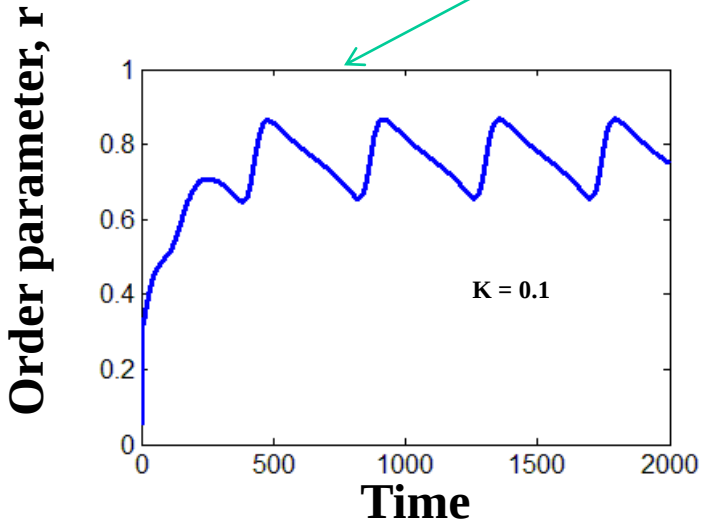


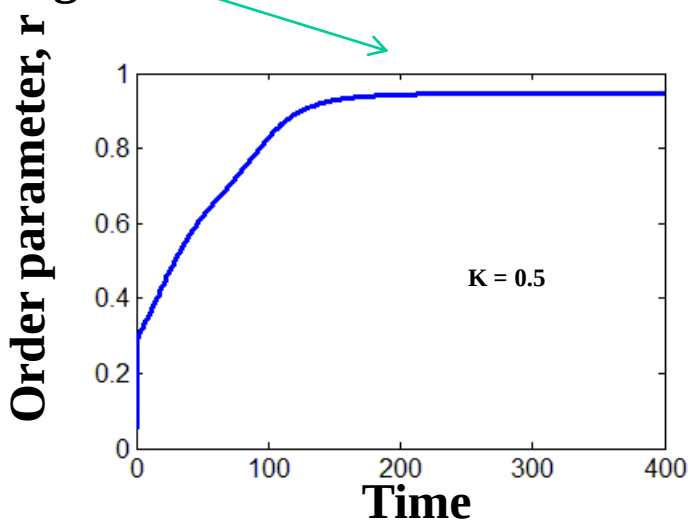
Animation of oscillators that NEVER completely synchronize





Animation of oscillators that synchronize





Evolution of order parameter at unstable (K=0.1) and stable (K=0.5) regimes

Basis functions for solutions on networks

- Our fine variables are functions on a network: phase angles, θ
- Functions in physical space are usually approximated using Fourier modes (sines and cosines – eigenfunctions of the Laplacian in space)
- By analogy, we examine the diffusion operator on a network, the **Graph Laplacian (L)***.
  - We use a FEW eigenvectors (v_j) of this matrix (L) as the basis vectors#.

$$\boldsymbol{\theta} \sim \sum_{j=1}^{k} z_j \boldsymbol{v}_j \qquad k \ll n$$

Projecting the phase angles onto the basis vectors The coefficients are z j

so that we reduce the number of ODEs from *n* to *k*.

$$\frac{d\boldsymbol{\theta}}{dt} = F(\theta_1, \theta_2, \ldots, \theta_n)$$

From *n* ODEs to *k* ODEs (k<<n)

$$\frac{d\boldsymbol{z}}{dt} = F(z_1, z_2, \ldots, z_k)$$

*Reference: Nadler, B., Lafon, S., Coifman, R. R. and Kevrekidis, I. G., Diffusion maps, spectral clustering and reaction coordinates of dynamical systems, Applied and Computational Harmonic Analysis, 21, 113 - 127 (2006)

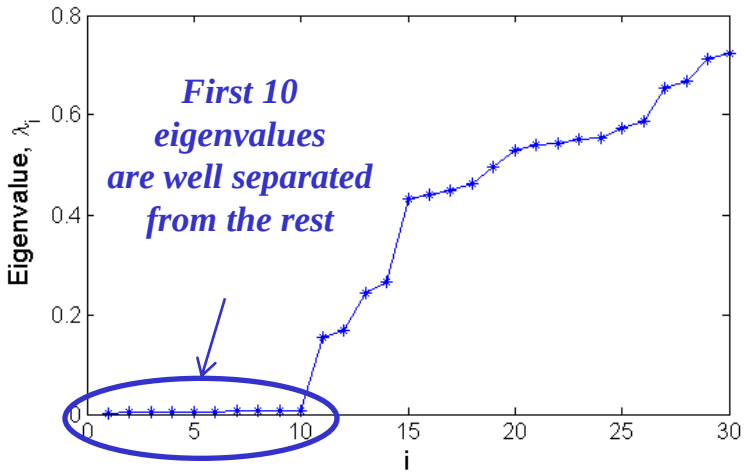
#Reference: McGraw, P. N. & Menzinger, M., Analysis of nonlinear synchronization dynamics of oscillator networks by Laplacian spectral methods, Phys Rev E Stat Nonlin Soft Matter Phys, 75, 027104 (2007)

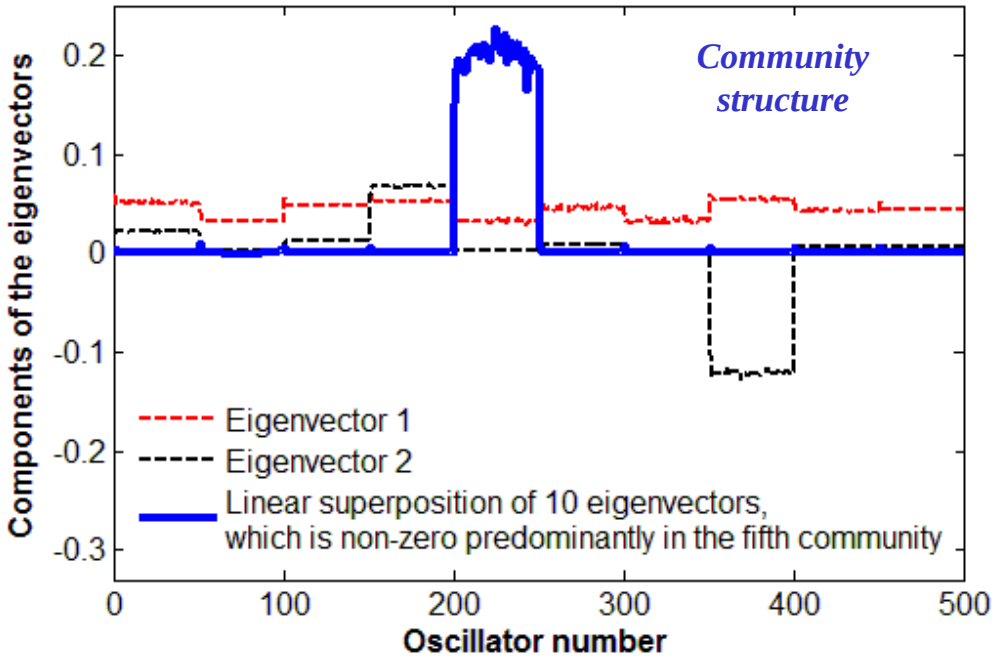
Graph Laplacian Eigenbasis

- Let *i*,*j* be the indices of nodes of a network and d_i be the degree of node *i*.
- Definition of a graph Laplacian (L) is given below:

$$L_{ij} := \begin{cases} 1 & \text{if } i = j \text{ and } d_i \neq 0, \\ -1/\sqrt{d_i d_j} & \text{if } i \neq j \text{ and } A_{ij} = 1, \\ 0 & \text{otherwise.} \end{cases}$$

Network: 10 communities x 50 members each

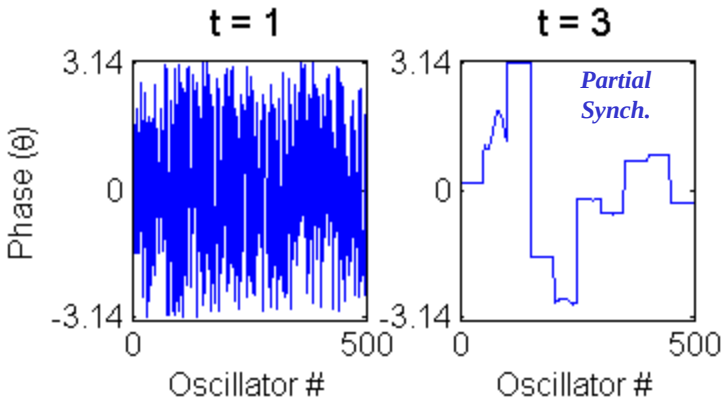




- **Thus, the first 10 Laplacian eigenvectors are the required basis vectors to project the phase angles of the oscillators!**

Submitted to PRE; arXiv:1105.4144

Coarse-graining



- For K=0.5, steady state is reached at t=60 , but ***partial synchronization*** inside communities is achieved before t=3 itself.
- Thus, representation using the lower-dimensional basis is a good approximation at all times

- **<u>A minor technical issue:</u>** Phase angles lie on a *circular* manifold, while the values of phase angles are represented on a *linear* scale (0 and 2π represent the same angle).
- Hence, sine and cosine of phase angles are used for representation instead of the angles themselves.

$$\Theta_j = e^{i\theta_j}$$ **Complex phase (sine and cosine of phase angles)**

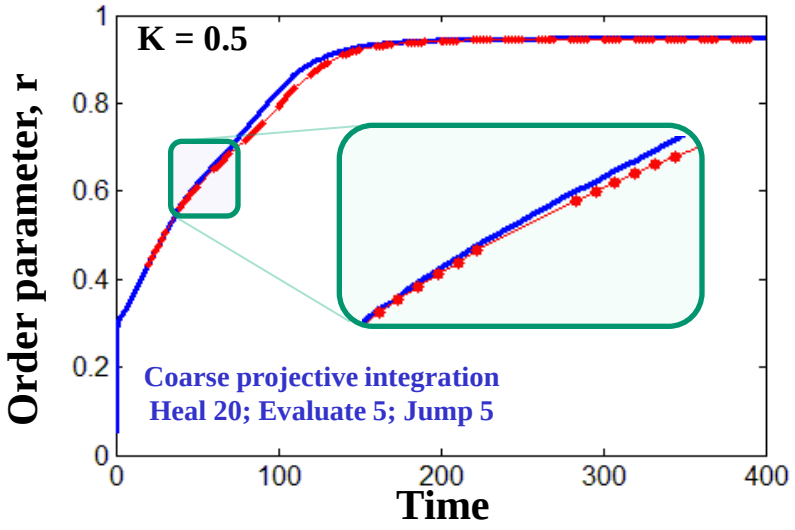
$$L\mathbf{v}_j = \lambda_j \mathbf{v}_j$$ **Laplacian eigenbasis, $\{v_j\}$**

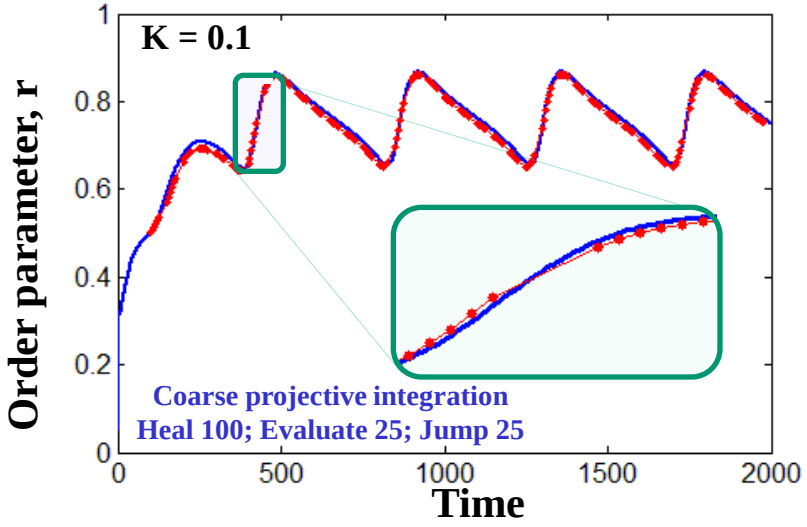
$$z_{j\in[1,m]} = \mathbf{v}_j^T \Theta$$ **Coarse variables, z_j**

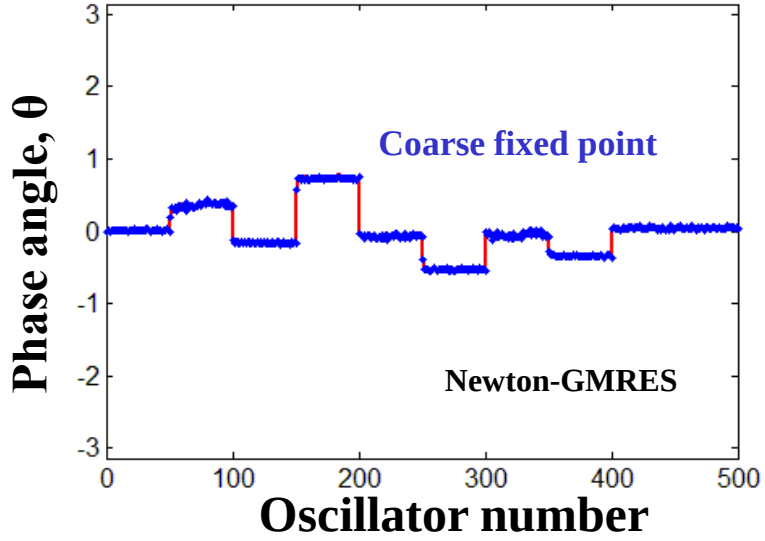
Coarse graining results

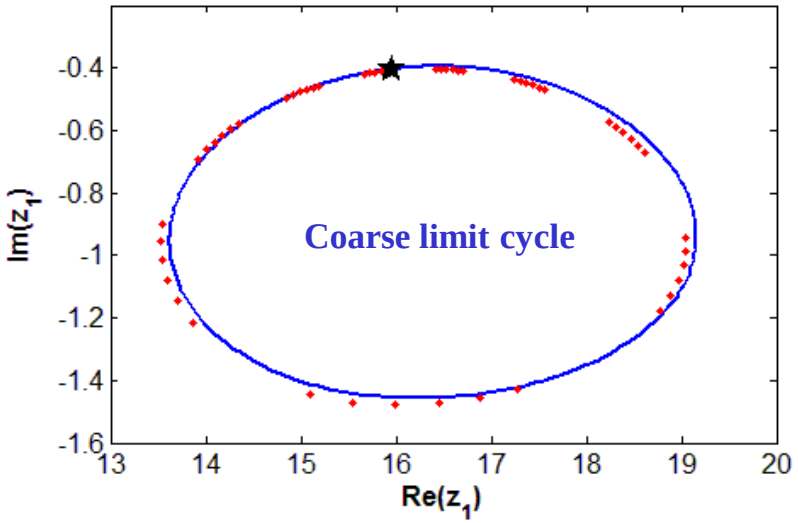
500 Phases → 10 Projection coefficients;
50 % Simulation, 50% Projection

Blue – From direct simulations; Red – From coarse model









Dynamics of networks

Problems where network structure/topology evolves according to microscopic rules

Goal: Identify coarse variables that capture the essential structure/topology of the networks evolving over time

14:40 - 15:20	**Ioannis Kevrekidis** (Princeton University) *Coarse-graining the dynamics of networks*

Selecting coarse variables

- Coarse variables selection
  - problem dependent
  - usually combinations of graph properties
    and they are chosen ***heuristically*.**
- Can we automate this coarse variable selection step?
- Assume we obtain a family of graphs by simulating the dynamical model.
- Is it possible to *automatically* find potential coarse variables (minimum crucial information) for representing the dynamical process at the macroscopic level?
- Need to use data mining techniques (like DMAPs).

Diffusion maps

(non linear dimensionality reduction technique)

P_i - Set of data points – *say vectors*

D_{ij} - distance/similarity metric – *like Euclidean distance*

From the matrix $D = \{d_{ij}\}$, we form

$W = \{w(d_{ij})\}$ - non-linear transformation of D

$w(d)$ is a non-negative function, $w(0) = 1$,

and $w(d)$ decreases as d increases such as

$$w(d) = \exp(-(d/\varepsilon)^2)$$ **(ε – a typical neighborhood distance)**

Each row of W is scaled by its row sum to get a Markov matrix K.

Reference: Nadler, B., Lafon, S., Coifman, R. R. and Kevrekidis, I. G., Diffusion maps, spectral clustering and reaction coordinates of dynamical systems, Applied and Computational Harmonic Analysis, 21, 113 - 127 (2006).

Diffusion maps (Intuition)

K is a Markov matrix.

Defines: Random walk process

States – Data points

Transition probabilities – proportional to "closeness" between data points.

Properties of K:

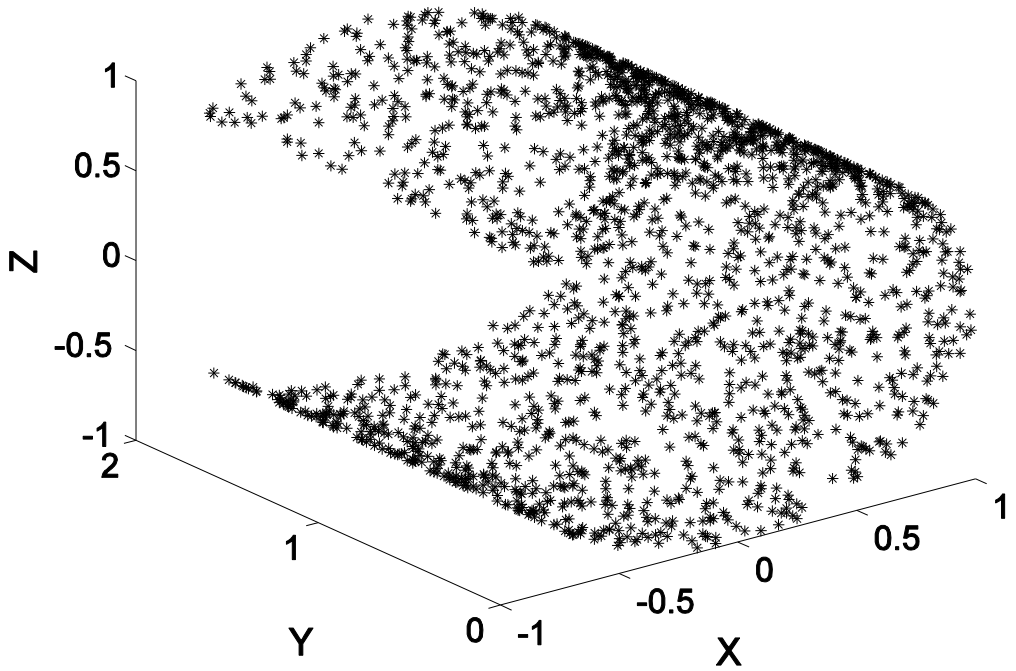
1. Largest eigenvalue is 1. (Trivial eigenvector)
2. Next few largest eigenvalues and their vectors determine the structure of the data.

Diffusion map example

EXAMPLE

2000 uniformly random points on a rectangle wrapped onto ¾ of a cylinder.

$\varepsilon = \max_i \min_k d_{ik}$ **(max nearest neighbor).**



Although there are three coordinates for every point,
we know that our data really lives on a two-dimensional surface!

Diffusion map example

If we run PCA:

3 important eigenvalues with their eigenvectors

corresponding to

Cartesian coordinates: x, y and z.

If we run DMAPs:

(we expect)

2 principal directions:

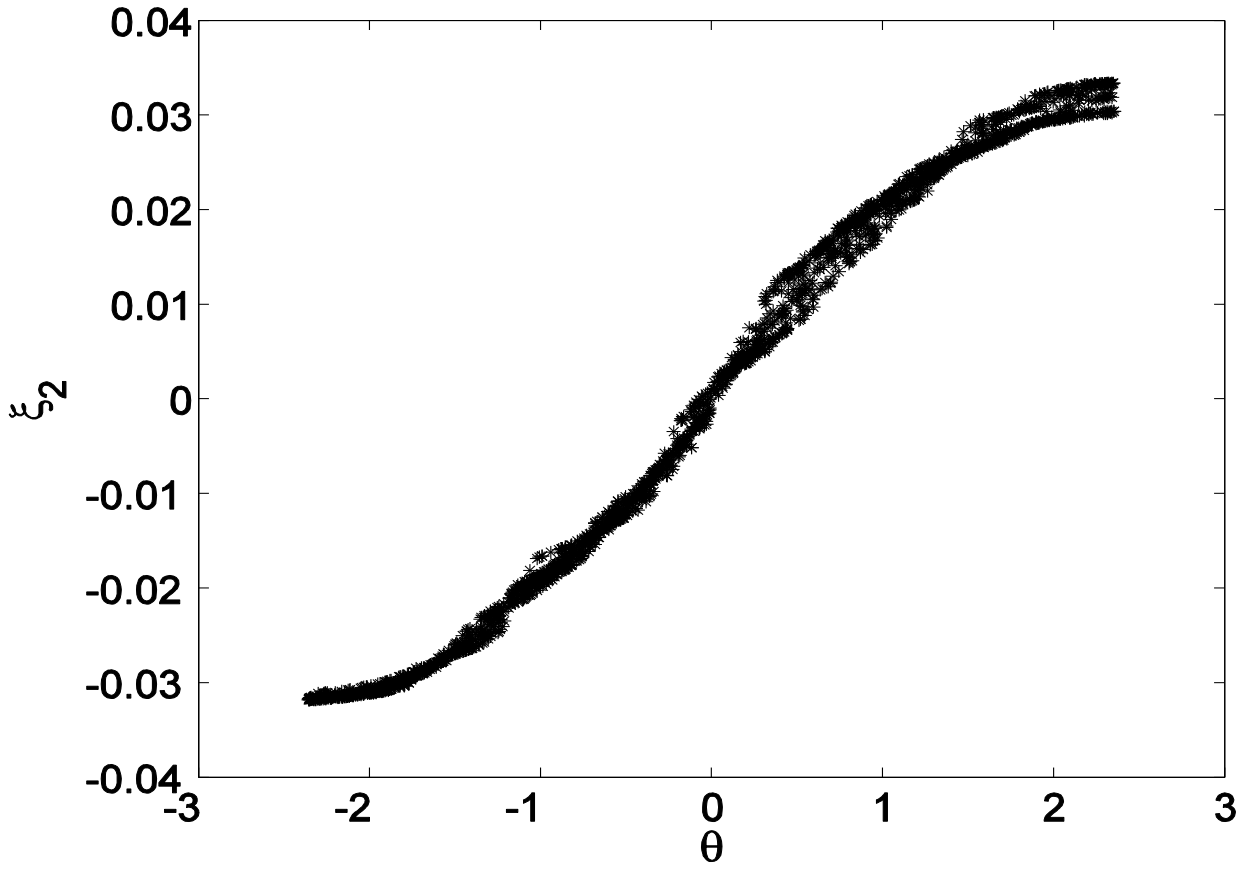
axial direction

AND

direction along the

curved surface of the cylinder.

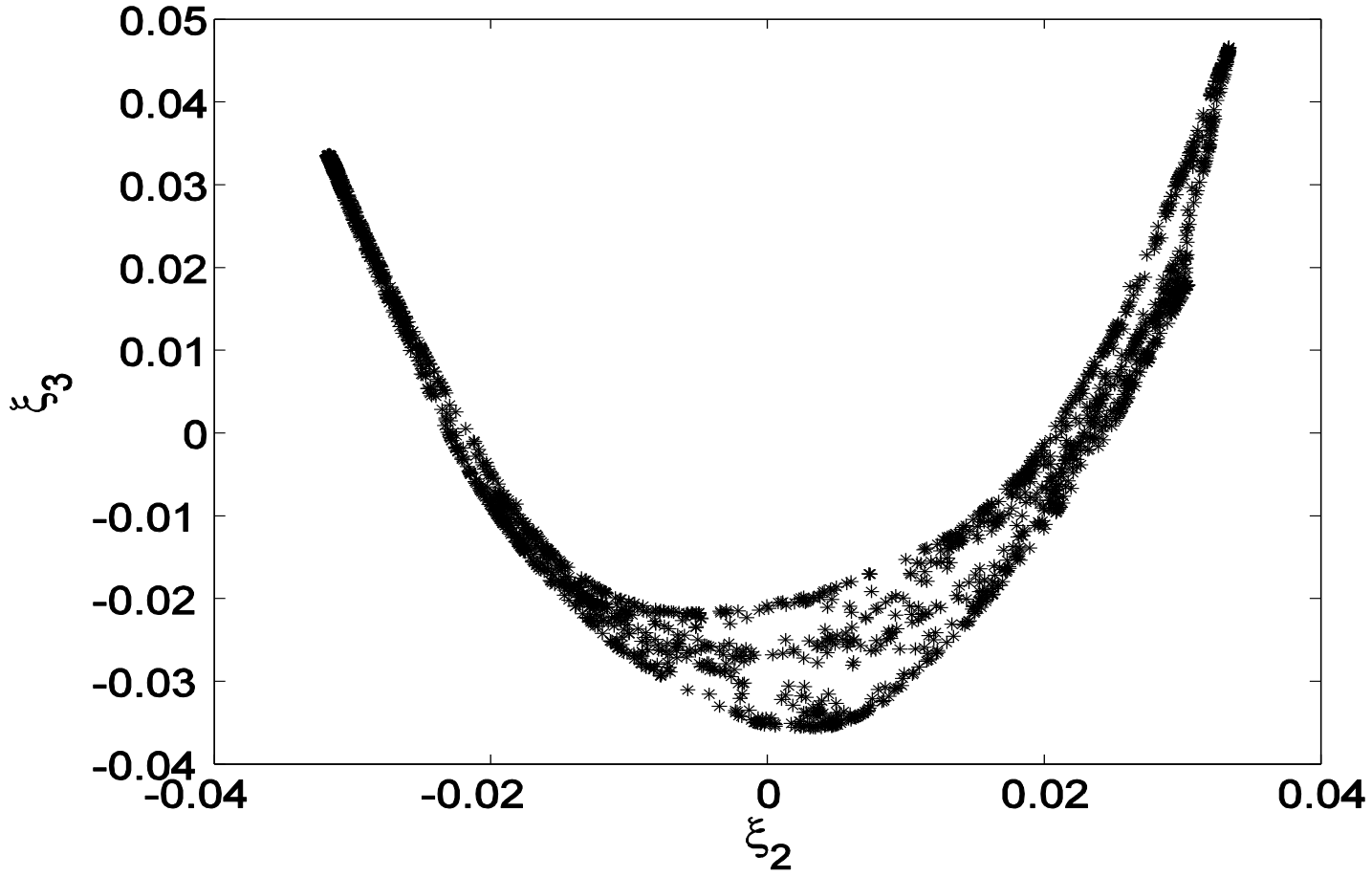
Diffusion map example



Components of second eigenvector versus angle around cylinder

- roughly parameterizes that coordinate

Diffusion map example

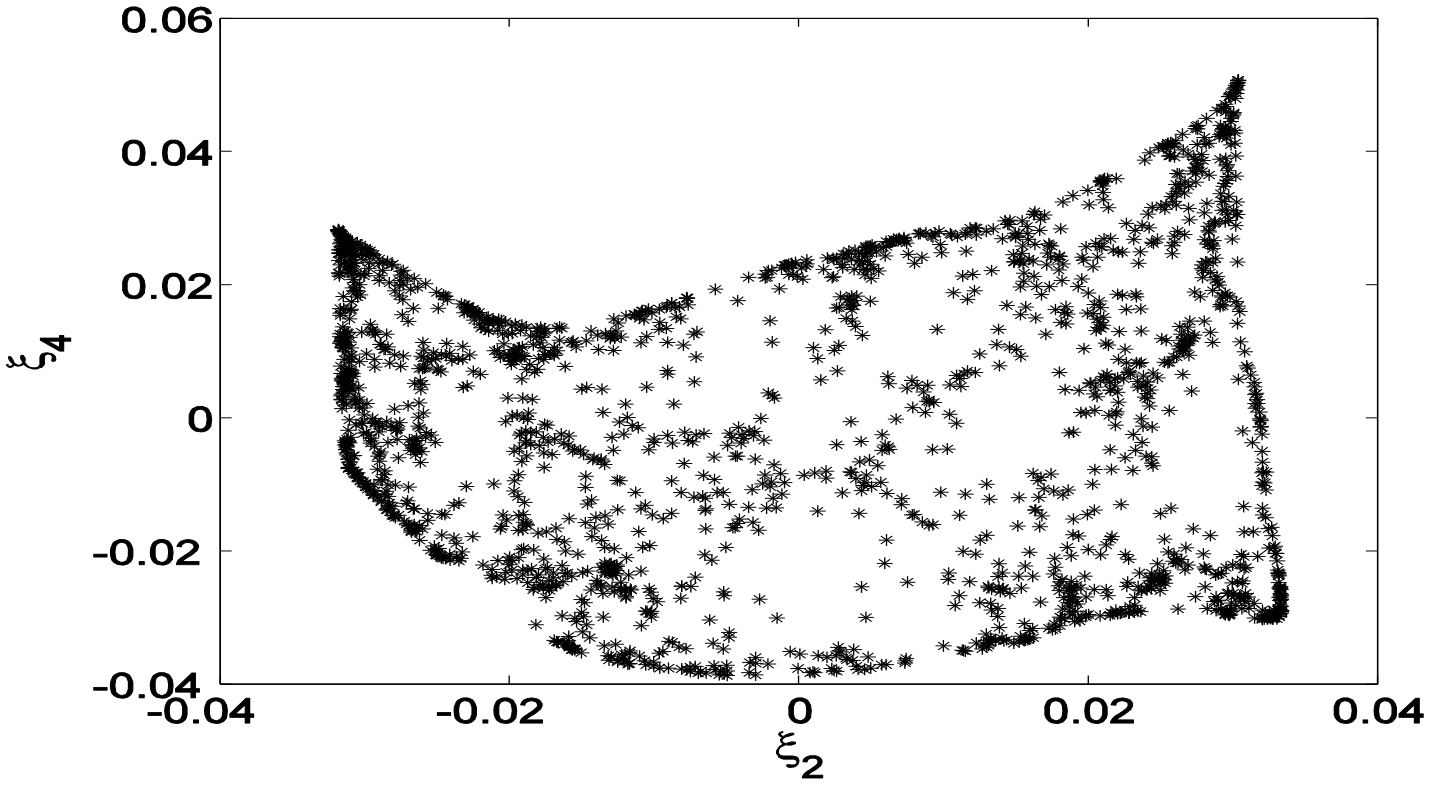


Components of second eigenvector versus components of third eigenvector

- they show a dependence

(third eigenvector is essentially in the same direction as the previous one)

Diffusion map example



Components of second eigenvector versus components of fourth eigenvector

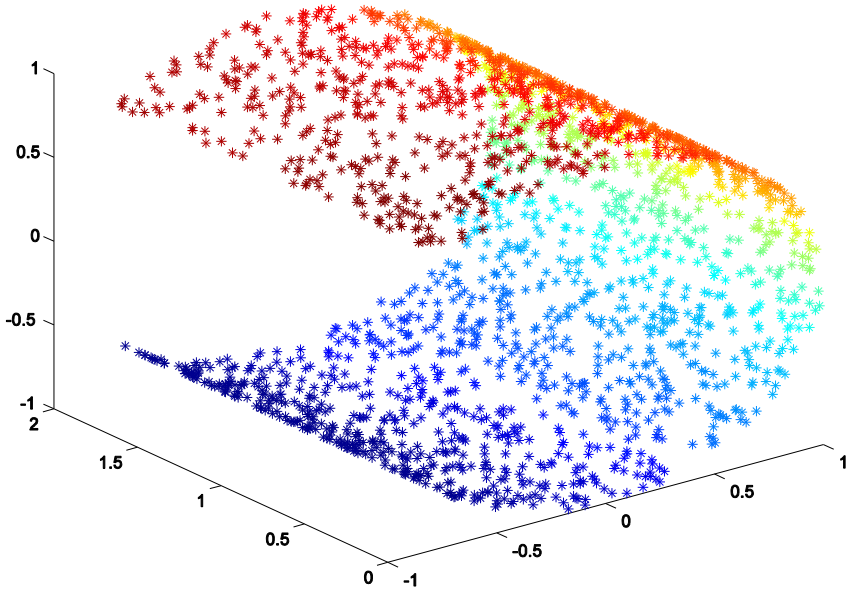
- they are not dependent

fourth eigenvector parameterizes another direction

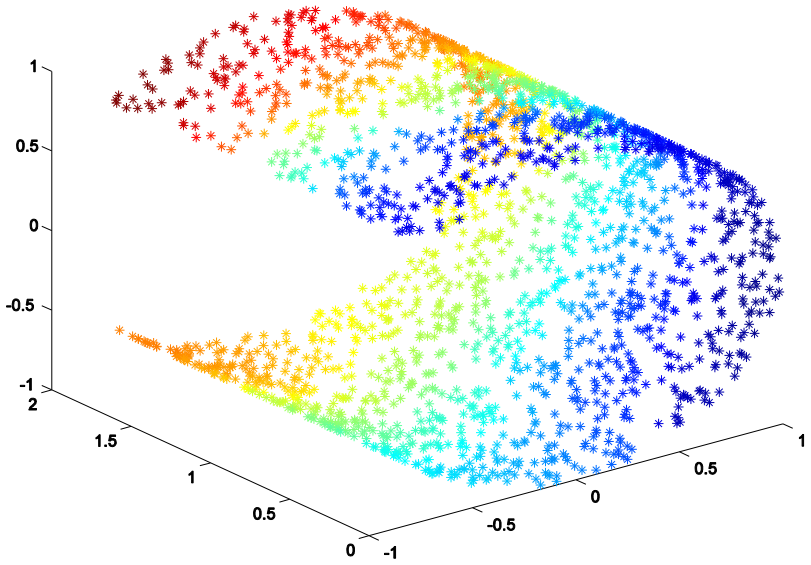
Diffusion map example

These are two sets of points colored by the size of the eigenvector entry for each point.

Colored by second eigenvector



Colored by fourth eigenvector



Extension to graph data

- Many data mining schemes (including DMAP) require definition of a similarity metric in the space of data points.
- However, defining a similarity metric is not trivial due to the problem of isomorphism.
- Challenge: Finding good ways to quantify the closeness (similarity) between pairs of graphs.

Two Options for similarity metrics

- Subgraph approach
  - *Structural information*
  - Choose a few representative subgraphs/motifs (for e.g. connected subgraphs of size less than 5) and compare densities (frequency of occurrence)

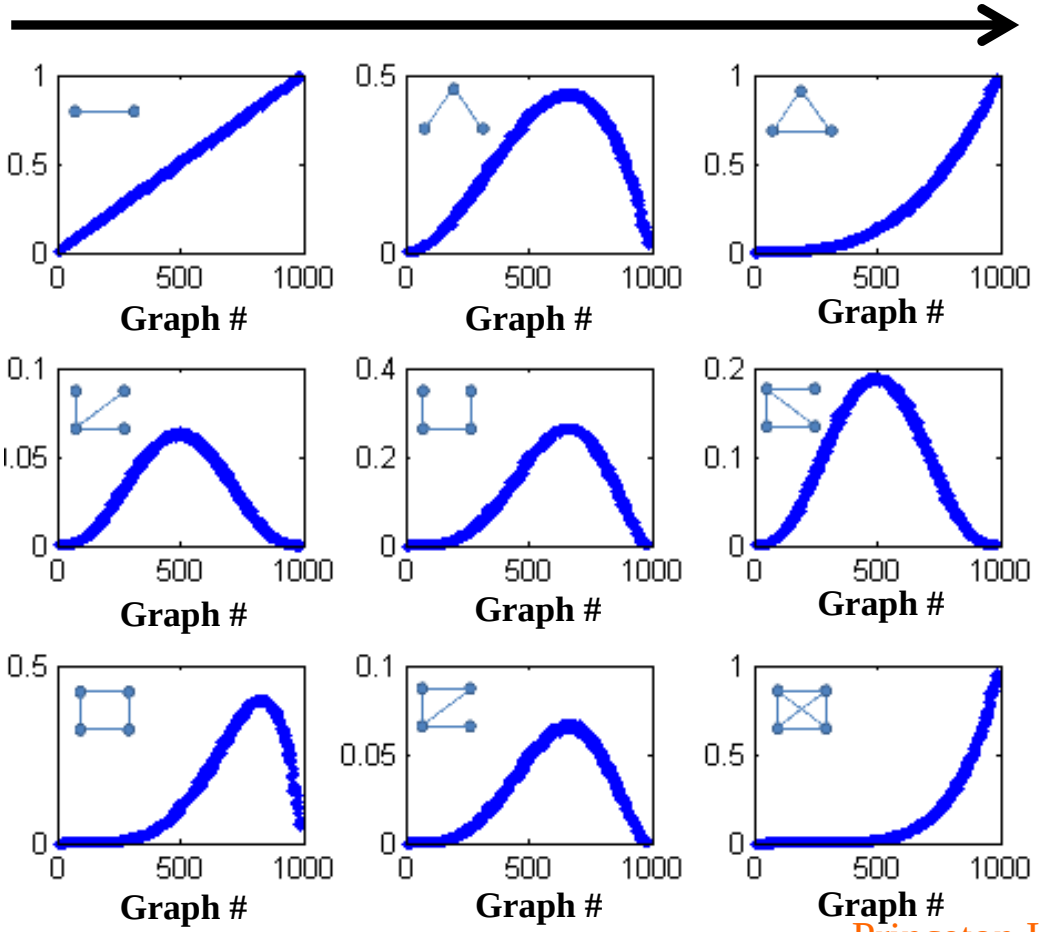
- Random walk approach [1]
  - Consider random walks with a finite stopping probability on the nodes on both graphs
  - Compare the number of k-length random walks
  - This can be evaluated efficiently using the spectral decomposition of the graphs.[1]

[1] **Ref:** S. V. N. Vishwanathan, K. M. Borgwardt, I. Risi Kondor, and N. N. Schraudolph. Graph Kernels. *ArXiv e-prints, July 2008.2*

Diffusion maps example

- Example: Consider a sequence of Erdős–Rényi graphs with increasing edge probability, p.

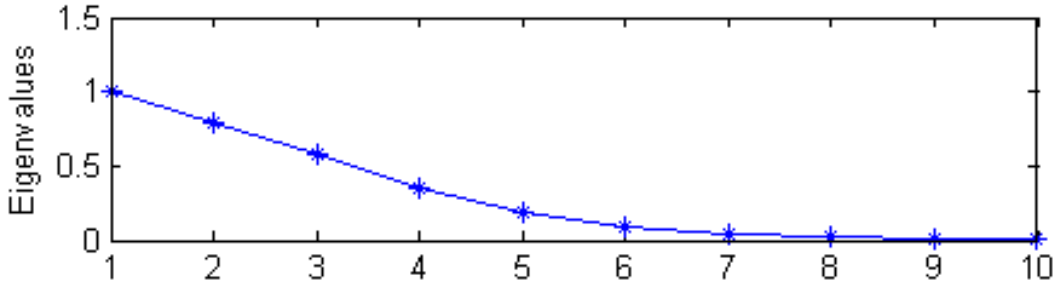
Graphs are arranged in the order of increasing p

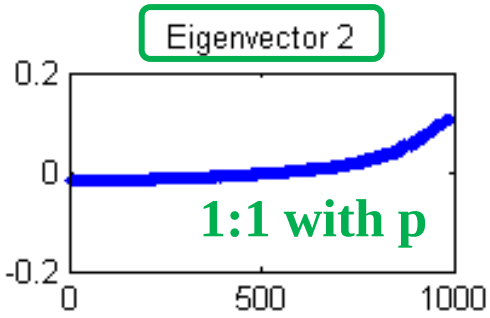


Densities of connected subgraphs[1] of size <= 4

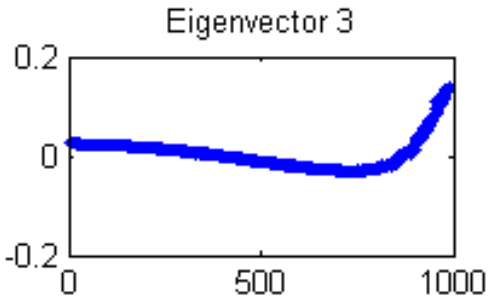
1 – Shown in insets

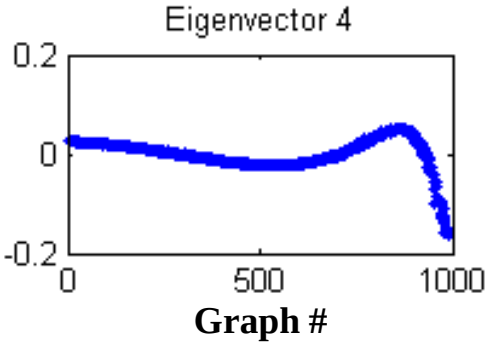
DMAP results: Subgraph approach

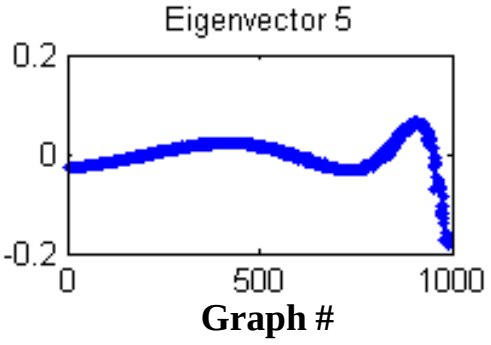




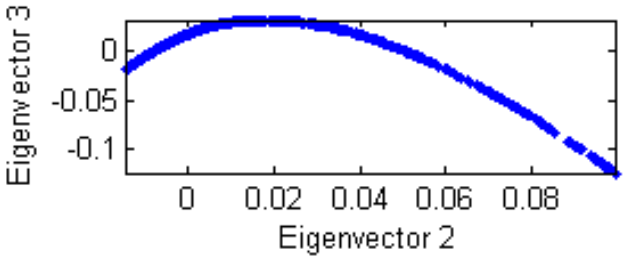
1:1 with p

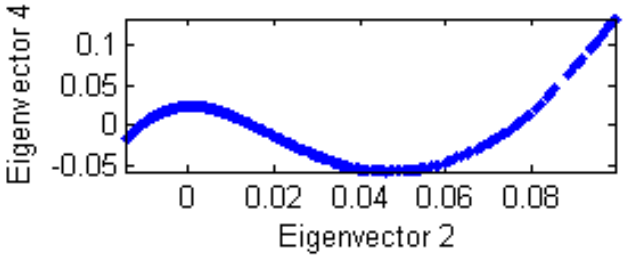


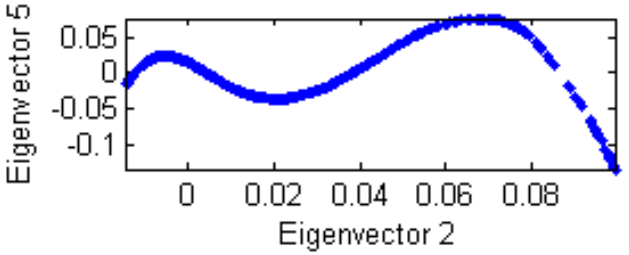




Other eigenvectors are just functions of eigenvector 2. No new direction!

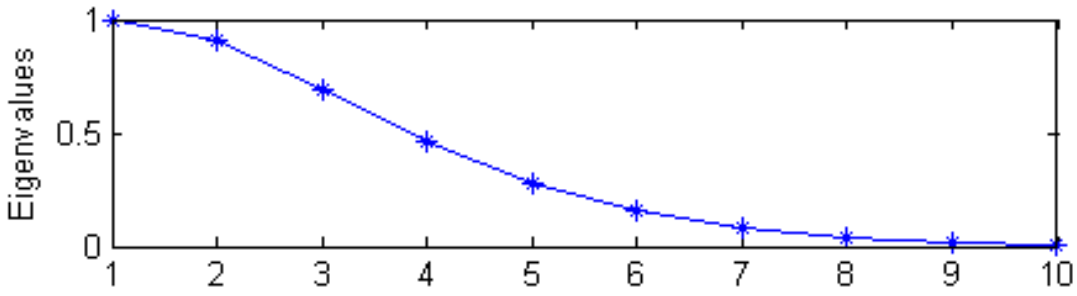


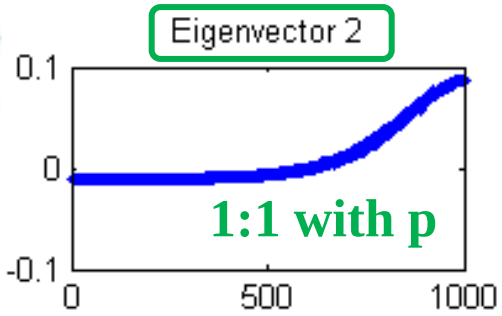




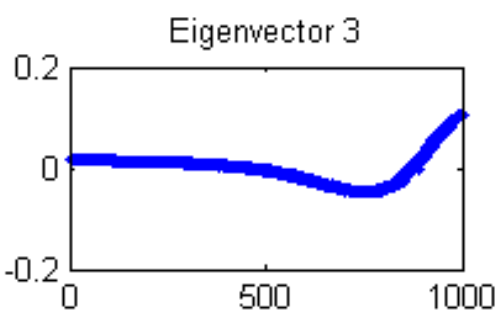
Graphs are arranged in the order of increasing p

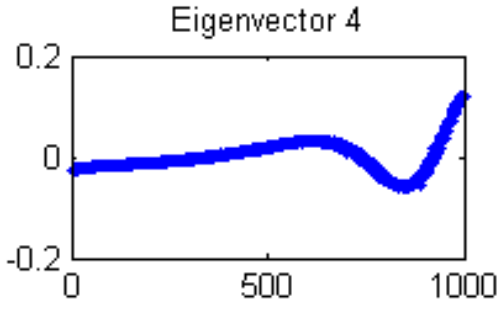
DMAP results: Random walk approach

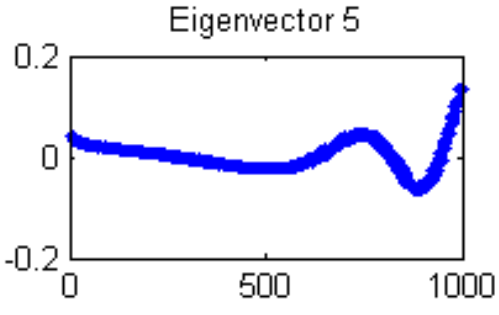




1:1 with p

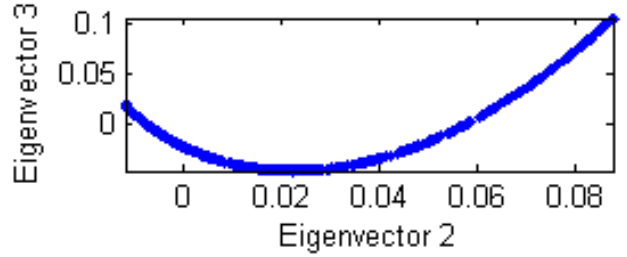


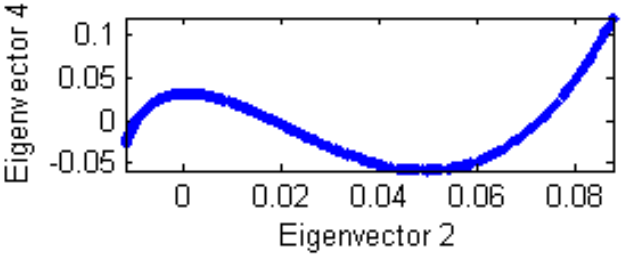


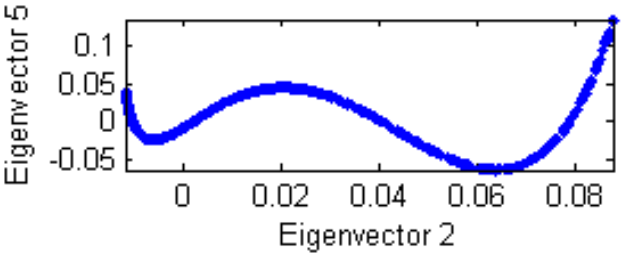


Graphs are arranged in the order of increasing p

Results similar to those for the previous approach







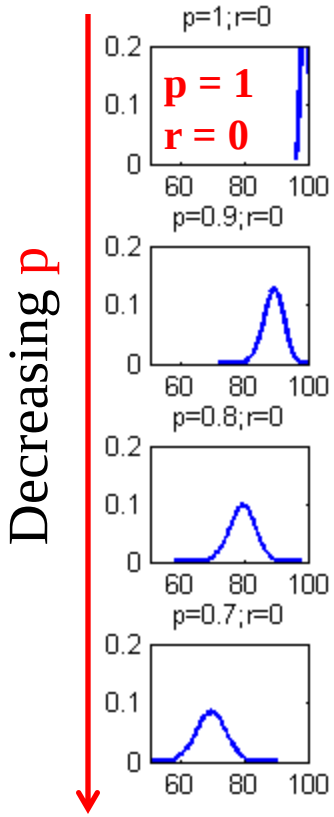
Note: Signs of eigenvectors are arbitrary.

2 parameter family (Chung-Lu based)

- Weights for each node, $w_i = Np(i/N)^r$, where i=1,2,… N
- Probability of existence of edge = $\min(w_i w_j / \Sigma w_k , 1)$

Increasing r

Decreasing p



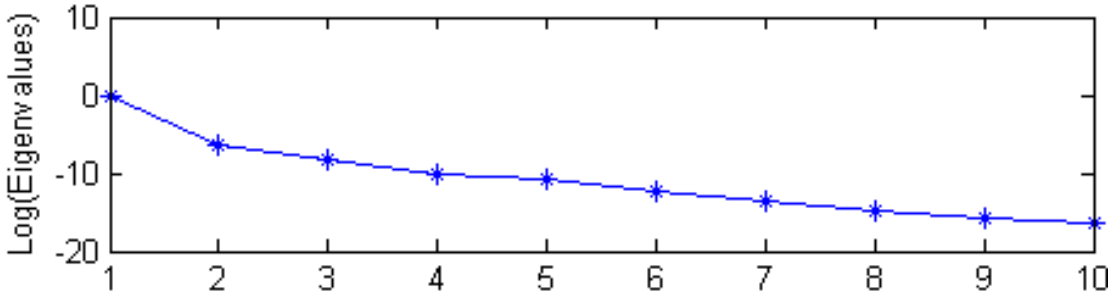
Degree distributions are plotted on the right.

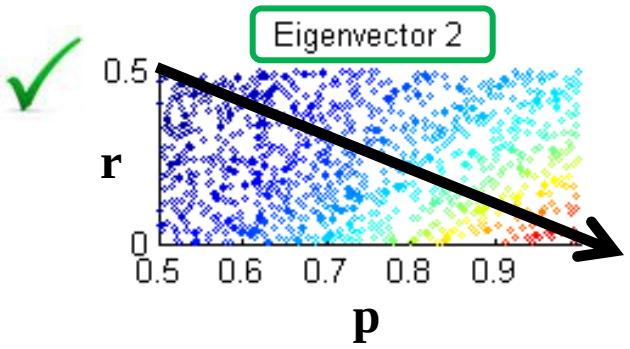
X- Axis: Degree
Y-Axis: Probability

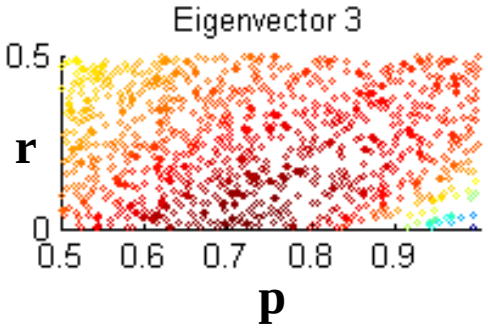
Approximately:

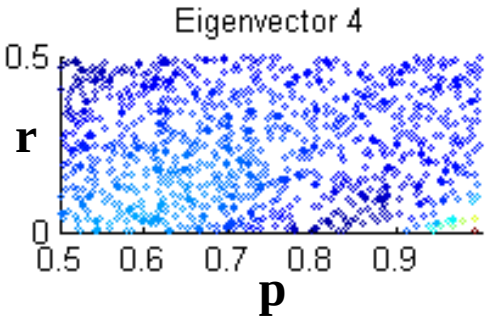
- p – *Density of edges*
- r – *Measure of skewness*

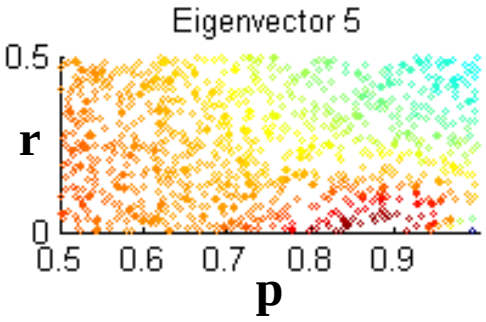
DMAP results: Subgraph approach

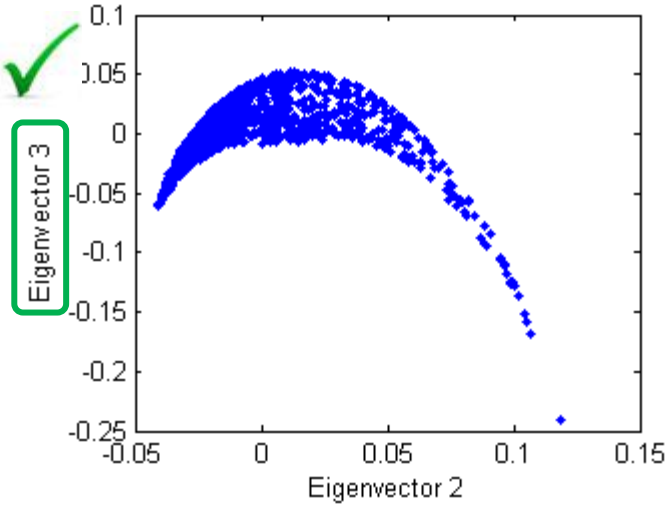








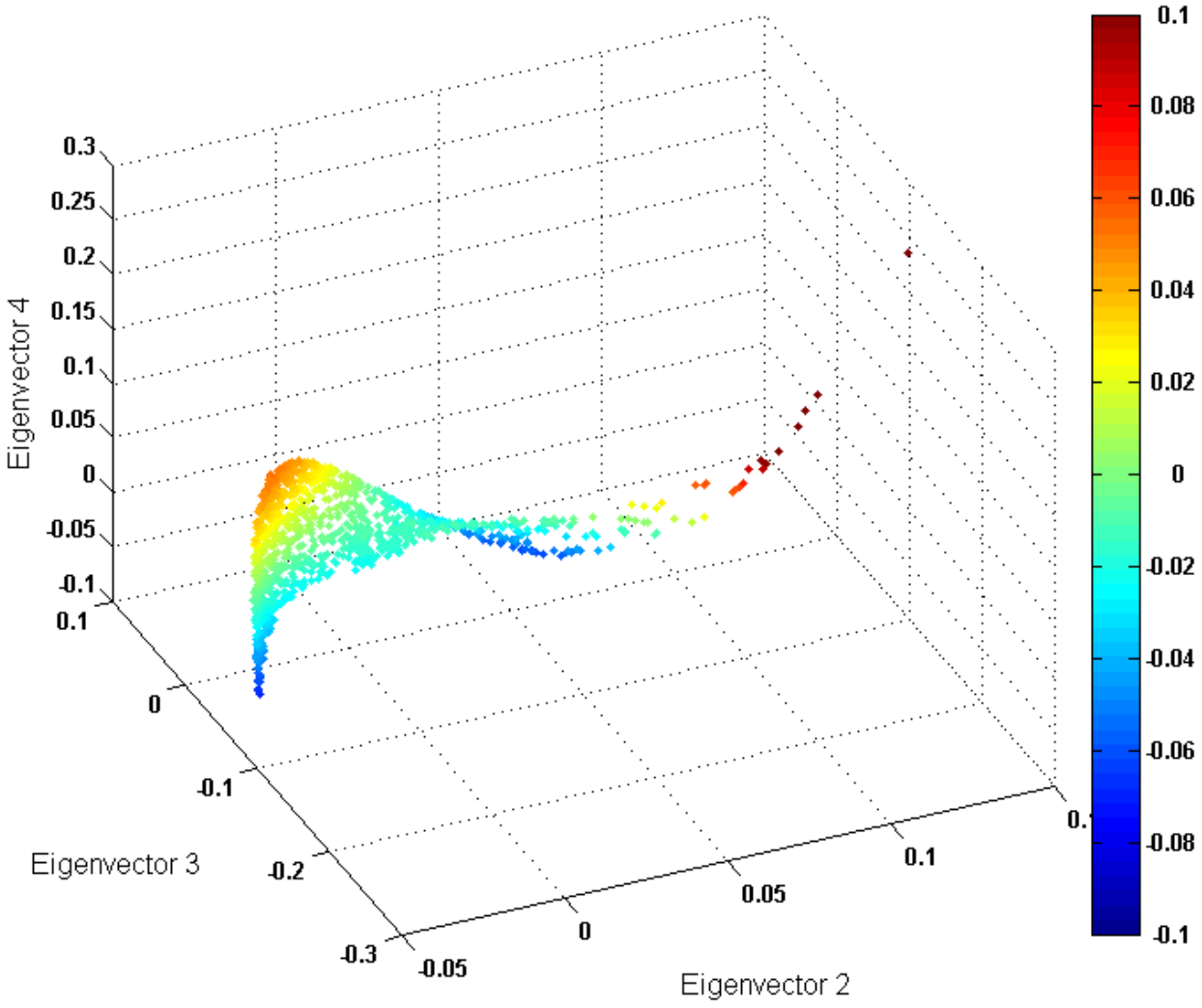




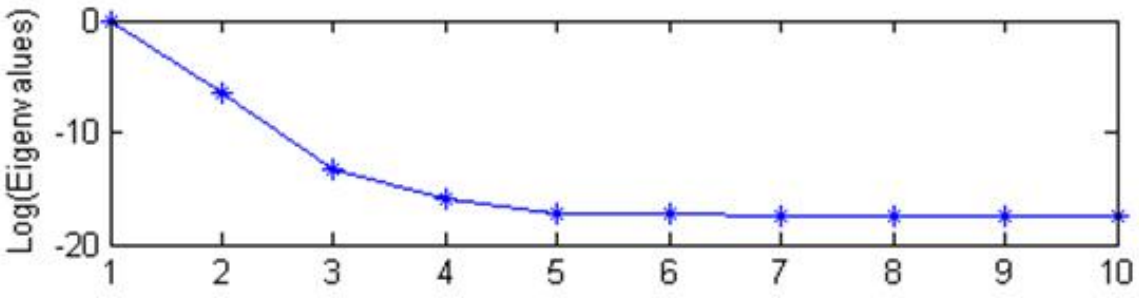
Colors are based on magnitude of eigenvectors

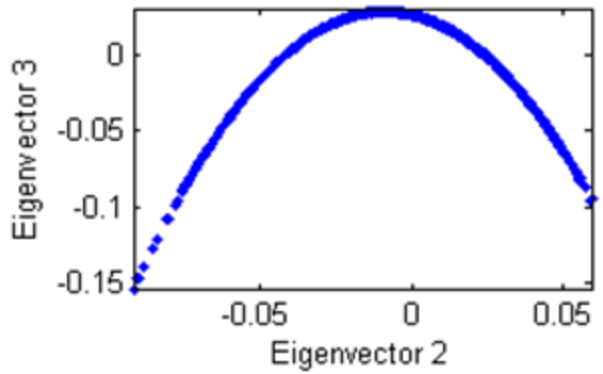
DMAP results: Subgraph approach

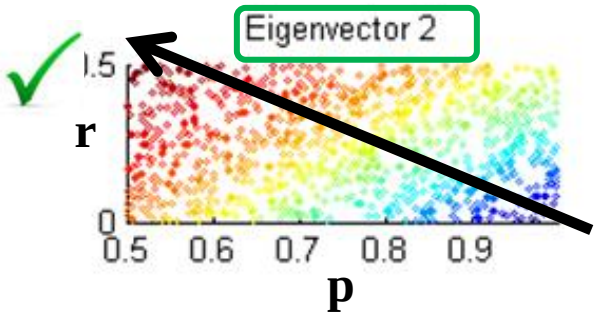
Eigenvector 4 is clearly a function of eigenvectors 2 and 3.

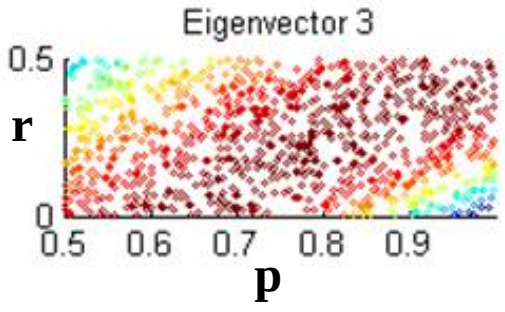


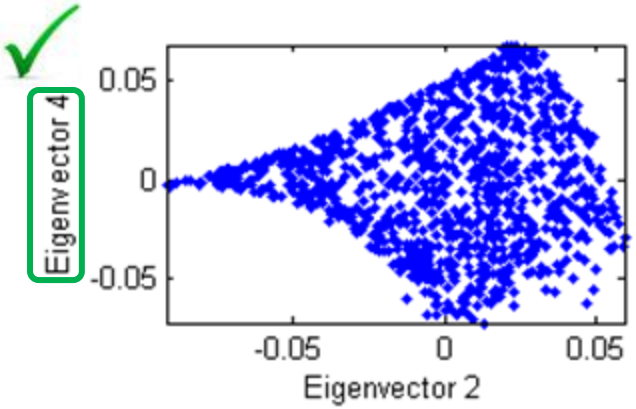
DMAP results: Random walk approach

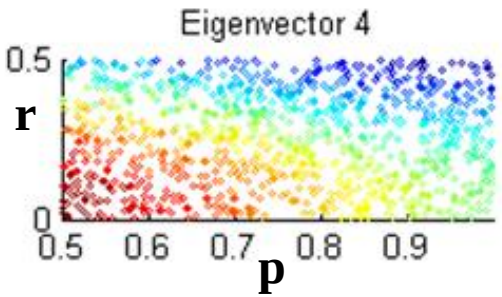


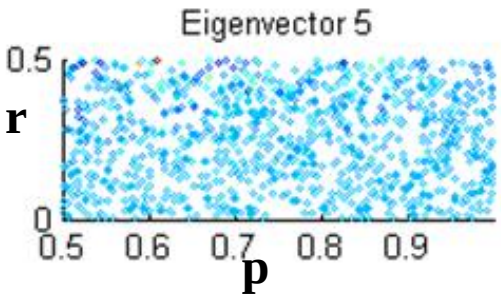












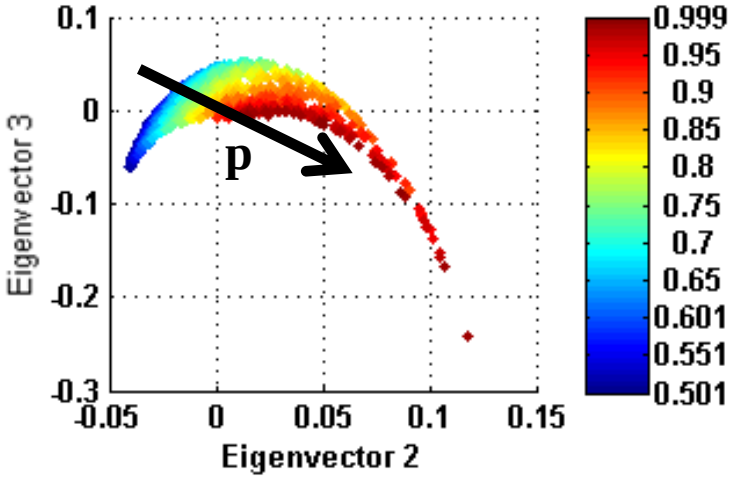
Colors are based on magnitude of eigenvectors

Quick recap

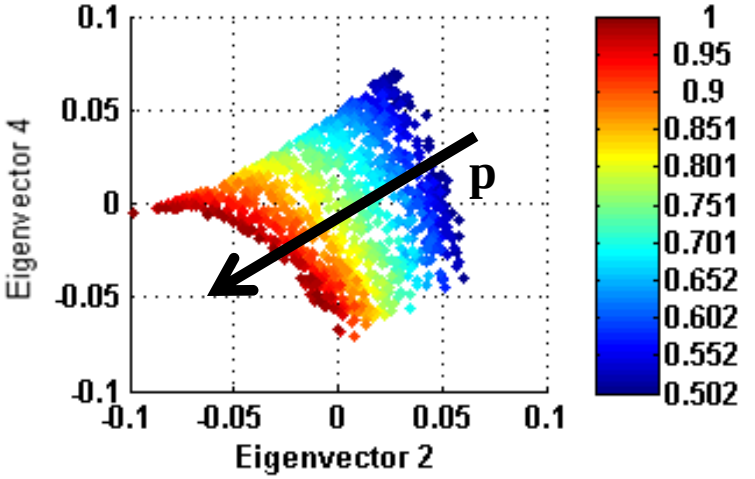
1. Create graphs using a 2D model, CL(p,r).
2. Forget the principal parameters, p and r.
3. Apply Diffusion MAPs.
4. Diffusion Map finds principal coordinates.
5. Check if we recover p and r!

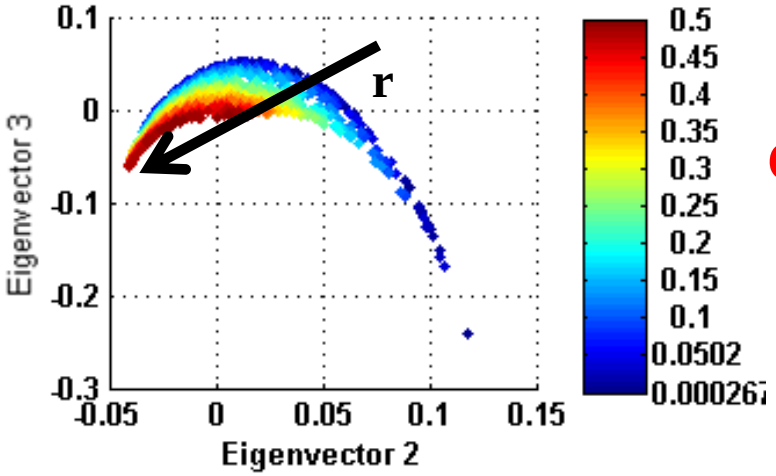
The two 2-D manifolds

Subgraph method

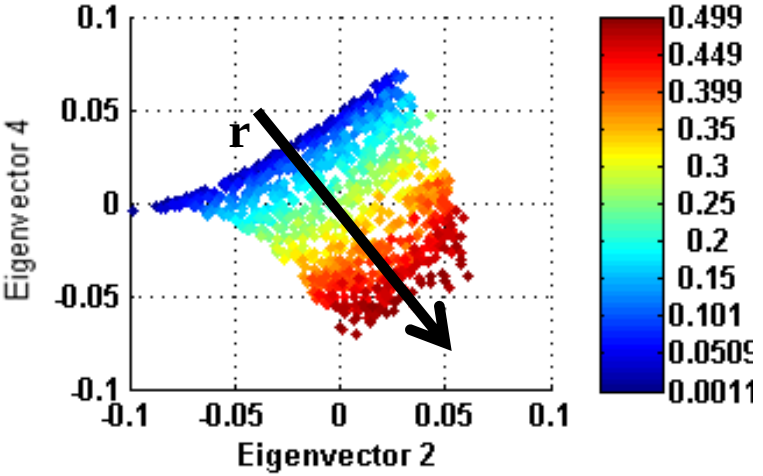


Colored by 'p'





Colored by 'r'



Random walk method

Conclusions

- Dynamics "on" networks
  - Coarse graining using ***observed features*** of solutions on networks
  - Specific example: Synchronization of networked oscillators

    The ***low-dimensional*** network structure imposes on the structure of solutions (oscillator phases) on the network. This structure is captured by ***eigenvectors of the graph Laplacian*** defined on the network
- Dynamics "of" networks
  - Data mining to find good coarse variables.
  - Defining ***similarity metrics*** between pairs of graphs.
    - "Subgraph method" and "random walk method"

      Both approaches require ***tuning***

      in terms of assigning weights.

      However, the random walk approach scales better

      in terms of computational effort.

Thank you!

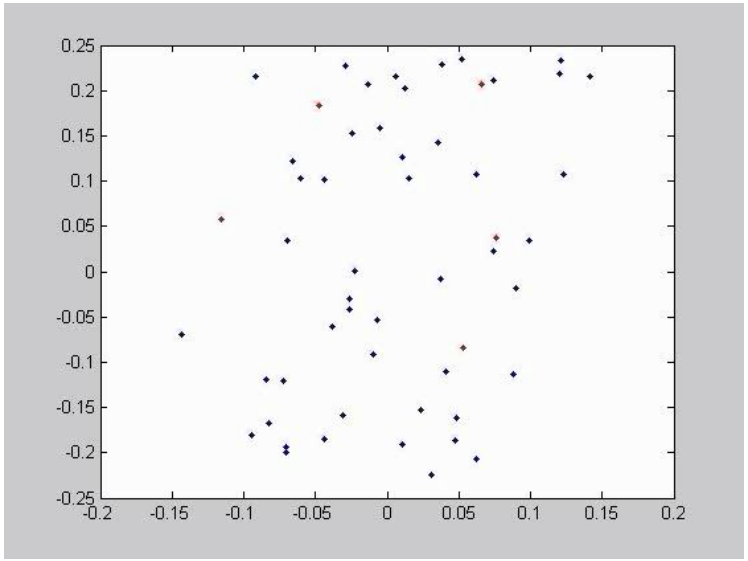
Effect of oscillator heterogeneity

- The portion of the phase vector NOT captured by the eigenbasis **(i.e., the excess over the projection or the residual)** is plotted against the oscillator frequencies.

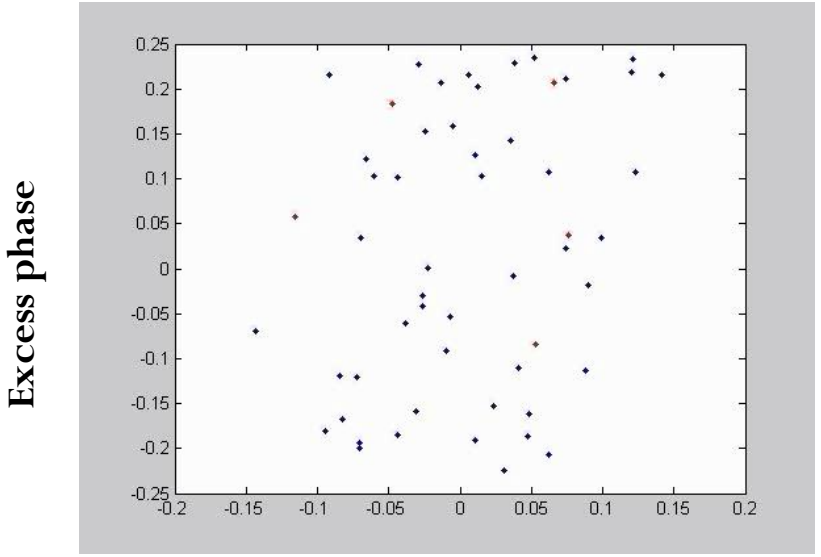
Excess phase $\Theta^{ex} = \Theta - \sum_{j=1}^{m} \mathbf{v}_j^T \Theta \mathbf{v}_j$

A correlation (*c*) develops *quickly* between this excess phase and the intrinsic oscillator frequency.

(Notice the red points – they belong to oscillators from one specific community)



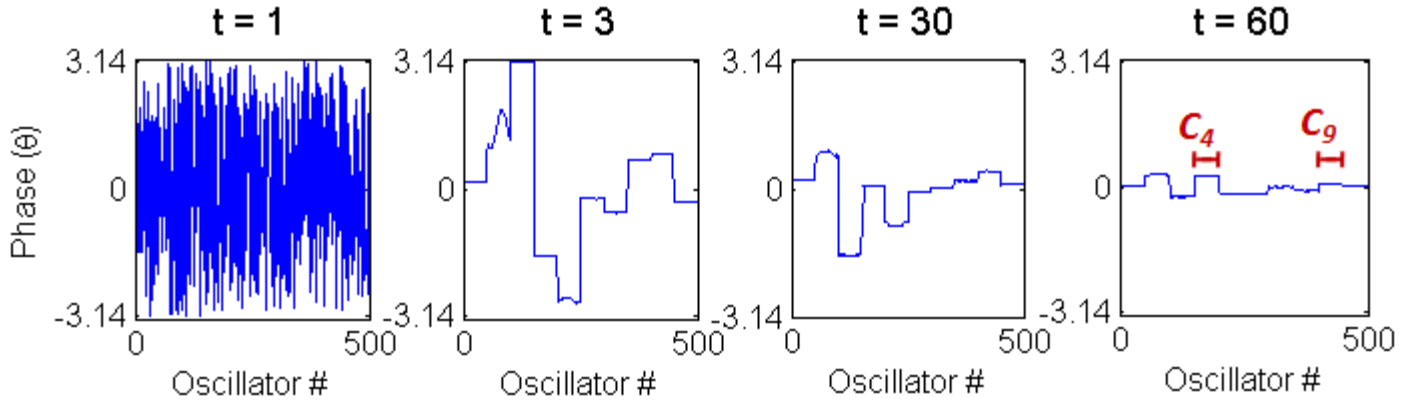
K = 0.1

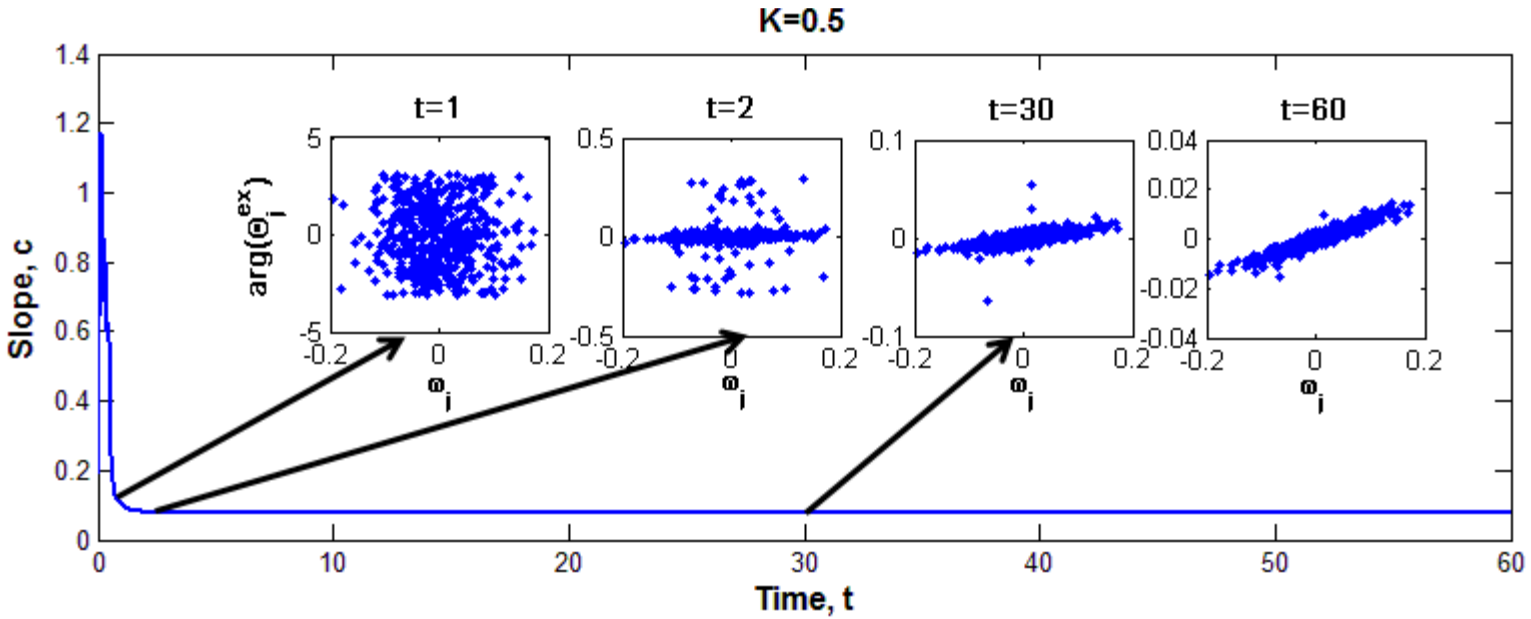


K = 0.5

Effect of oscillator heterogeneity

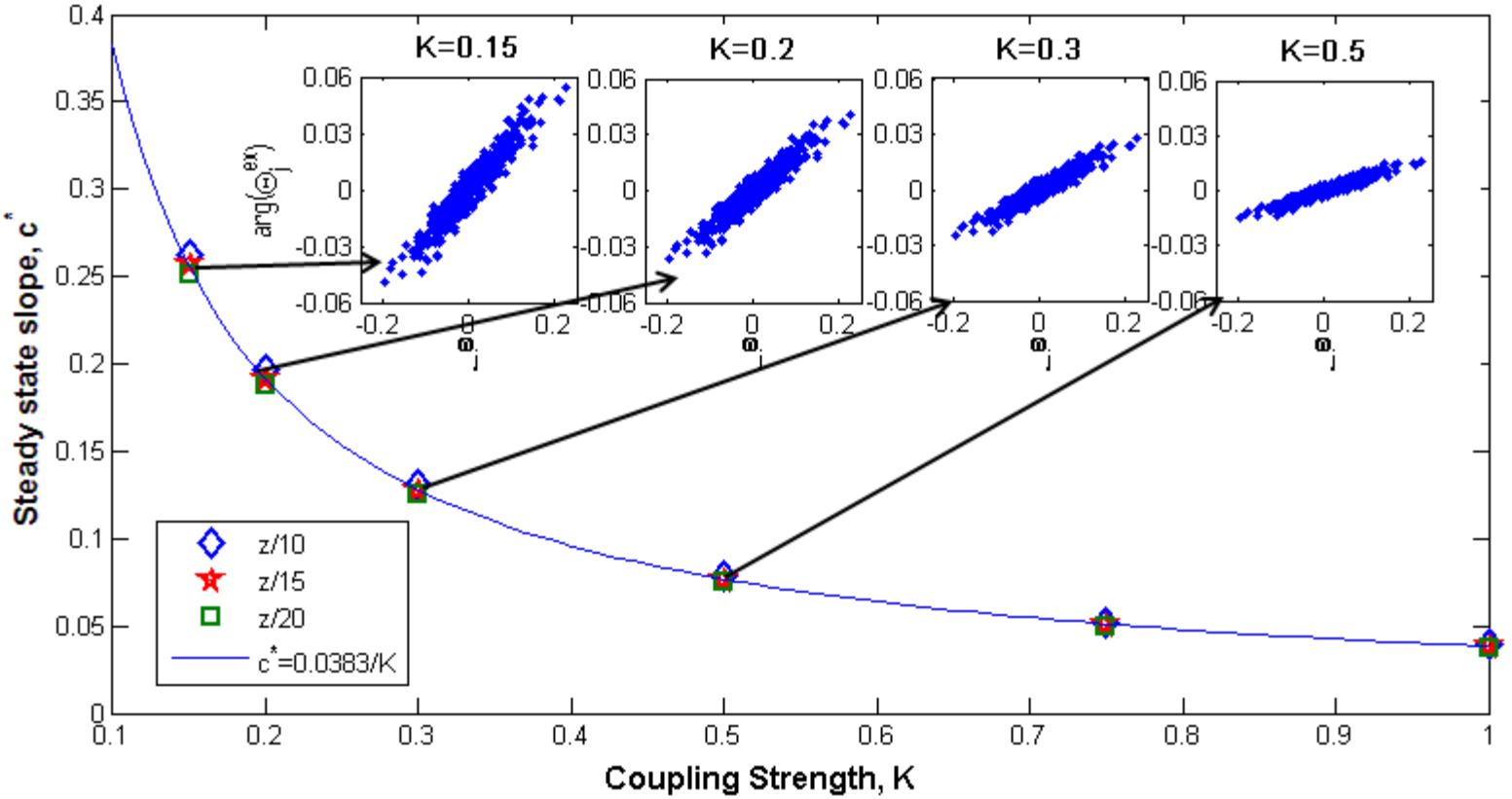
The correlation slope approaches its steady state value much faster than the time to reach the system steady state.





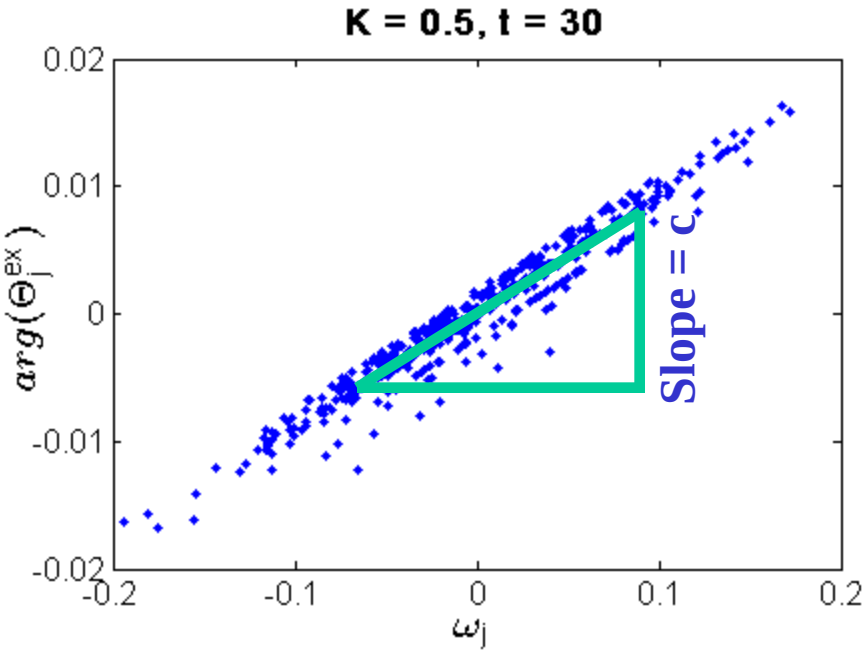
Effect of oscillator heterogeneity

The steady state value of correlation slope is observed to be inversely proportional to the coupling strength and independent of the variance in the intrinsic oscillator frequencies.



z = R.V. distributed according to Normal(0,1)

An improved coarse model



Oscillator frequencies

Additional coarse variable, c

$$\widetilde{\Theta}_j = e^{i(\theta_j - c\omega_j)}$$

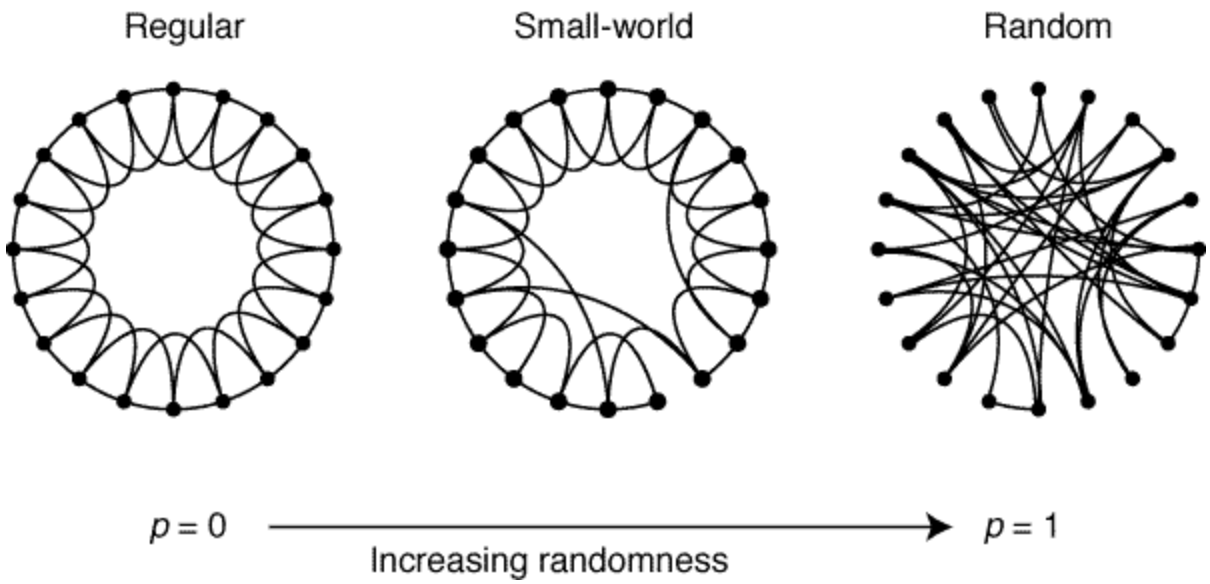
Projection coefficients, z_j

$$z_{j\in[1,m]} = \mathbf{v}_j^T \widetilde{\mathbf{\Theta}}$$

Projection of the "corrected" phase angles onto the graph Laplacian eigenbasis

Watts-Strogatz model

We start with a ring of *n* vertices, each connected to its *k* nearest neighbours by undirected edges. (*n* = 20 and *k* = 4 here). We choose a vertex and the edge that connects it to its nearest neighbour in a clockwise sense. With probability *p*, we reconnect this edge to a vertex chosen uniformly at random over the entire ring, with duplicate edges forbidden; otherwise we leave the edge in place. We repeat this process by moving clockwise around the ring, considering each vertex in turn until one lap is completed. Next, we consider the edges that connect vertices to their second-nearest neighbours clockwise and rewire as before. As there are *nk*/2 edges in the entire graph, the rewiring process stops after *k*/2 laps.



Courtesy: **Watts, D. J. and Strogatz, S. H., Collective dynamics of 'small-world' networks, *Nature, 393*, 440-442 (1998)**

Coarse fixed point performance

Case I : Average phase for each community

Case II: Laplacian eigenbasis (Structural)

Case III: Laplacian eigenbasis and oscillator frequency correction

Coarse variables	K=1	K=0.5	K=0.1
Case I	0.9974	0.9975	0.9976
Case II	0.9983	0.9983	0.9983
Case III	0.9994	0.9994	0.9995

Information added by including more coarse variables is meaningful!

Case II

$$\sum_{j=1}^{m} z_j \mathbf{v}_j \to \mathbf{\Theta}$$

Case III

$$\widetilde{\Theta}_j = e^{i(\theta_j - c\omega_j)}$$

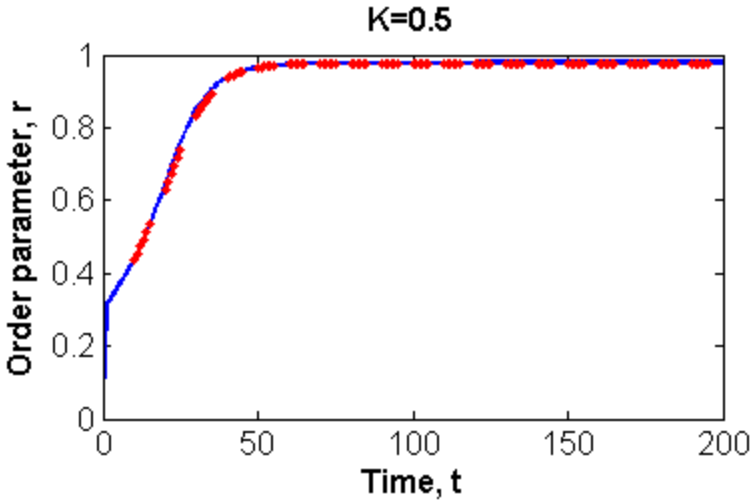
$$\sum_{j=1}^{m} z_j \mathbf{v}_j \to \widetilde{\mathbf{\Theta}}$$

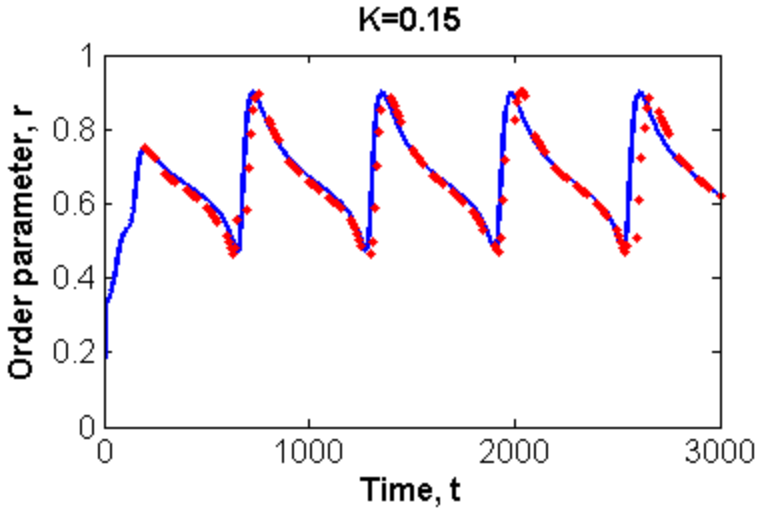
Oscillator frequencies from a Rayleigh distribution

- The oscillator frequencies were chosen by sampling 500 numbers from a Rayleigh distribution with parameter 0.1 and then subtracting the mean from these 500 samples.
- Coarse projective integration results similar to the case when frequencies were sampled from a Normal distribution

Blue – From direct simulations; Red – From coarse model

500 Phases → 11 Coarse variables; 50 % Simulation, 50% Projection

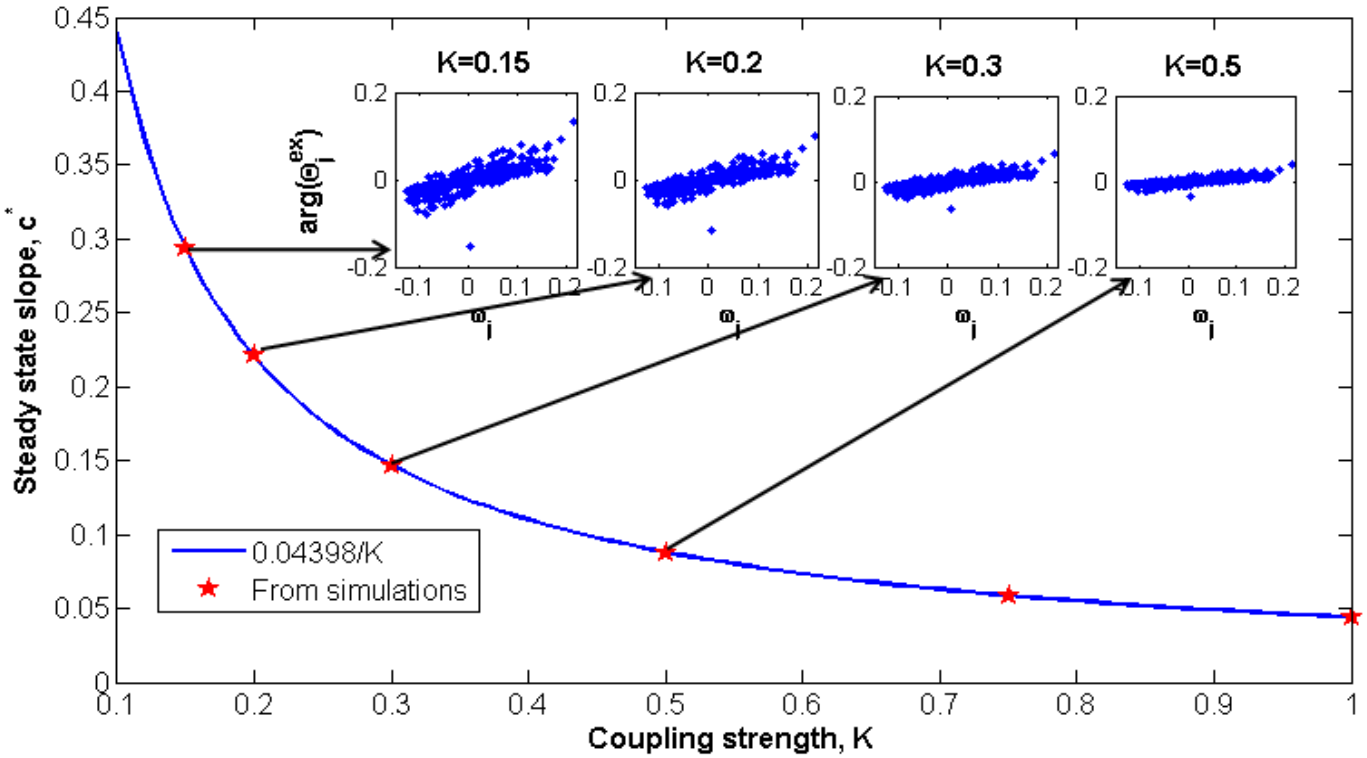




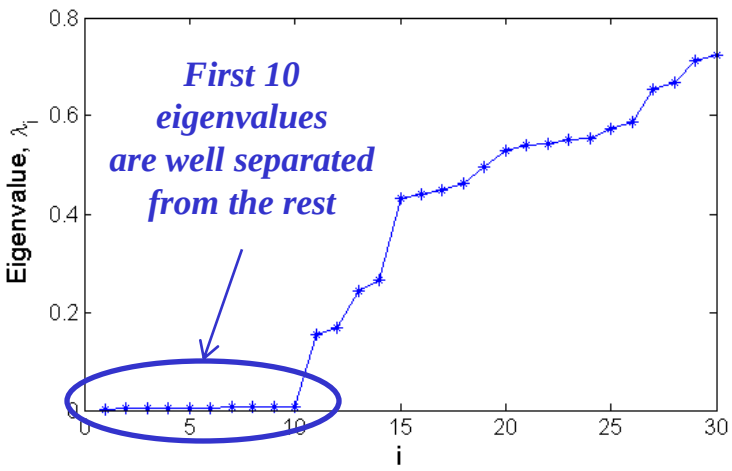
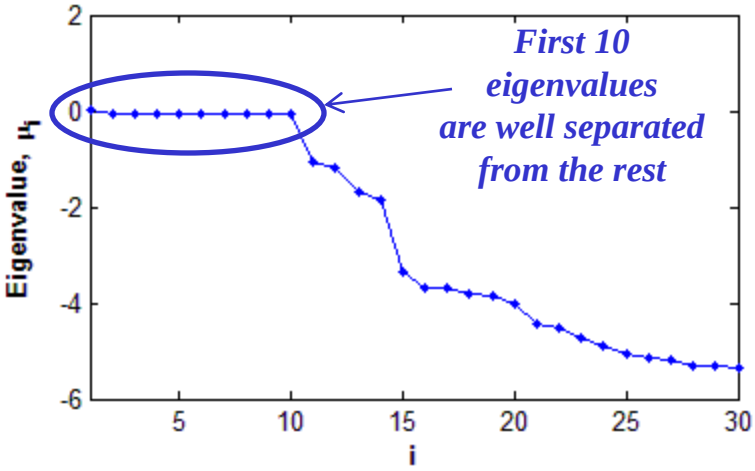
Effect of oscillator heterogeneity

- The inverse proportionality of the steady correlation slope with coupling strength holds for this case also.

  (intrinsic frequencies sampled from Rayleigh distribution)



Linearized Jacobian of the heterogeneous problem

Matrix	Linearized Jacobian	Graph Laplacian
Eigenvalues		
Eigenvectors	$\{w_i\}$	$\{v_i\}$

- Consider the matrix I, the inner product matrix of dimension 10 x 10, whose ij^{th} element is $w_i.v_j$ (dot product).
- The 10 eigenvectors of this matrix are listed here: 0.9993, 0.9993, 0.9993, 0.9993, 0.9995, 0.9992, 0.9992, 0.9995, 0.9995, 0.9991.
- Since all the 10 eigenvalues are close to 1, the space spanned by $\{w_i; i=1,2,\ldots10\}$ and $\{v_i; i=1,2,\ldots10\}$ are similar.
- Hence, one can use the first 10 graph Laplacian eigenvectors as basis vectors even for heterogeneous problems when the heterogeneity is small.