生命科学中的随机动力学模型三

姚 远

2011.03.08

It is important to understand what you CAN DO before you learn to measure how well you seem to have DONE it.

To learn about data analysis, it is right that each of us try many things that do not work.

---- John W. Tuckey

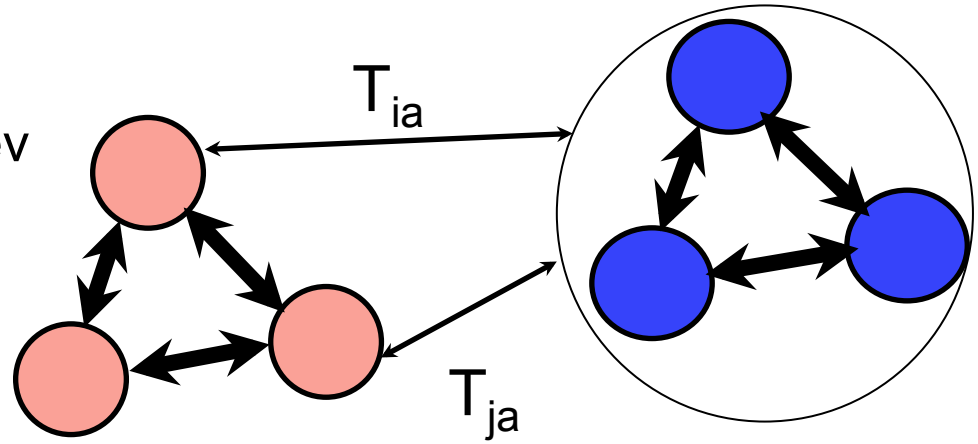
Spectral Clustering Theories

- In the last lecture, we mentioned two theories for spectral clustering
  - Lumpable Markov Chains
  - Cheeger's inequalities for graph min-cut

The two theories are different!

Spectral Clustering Theories

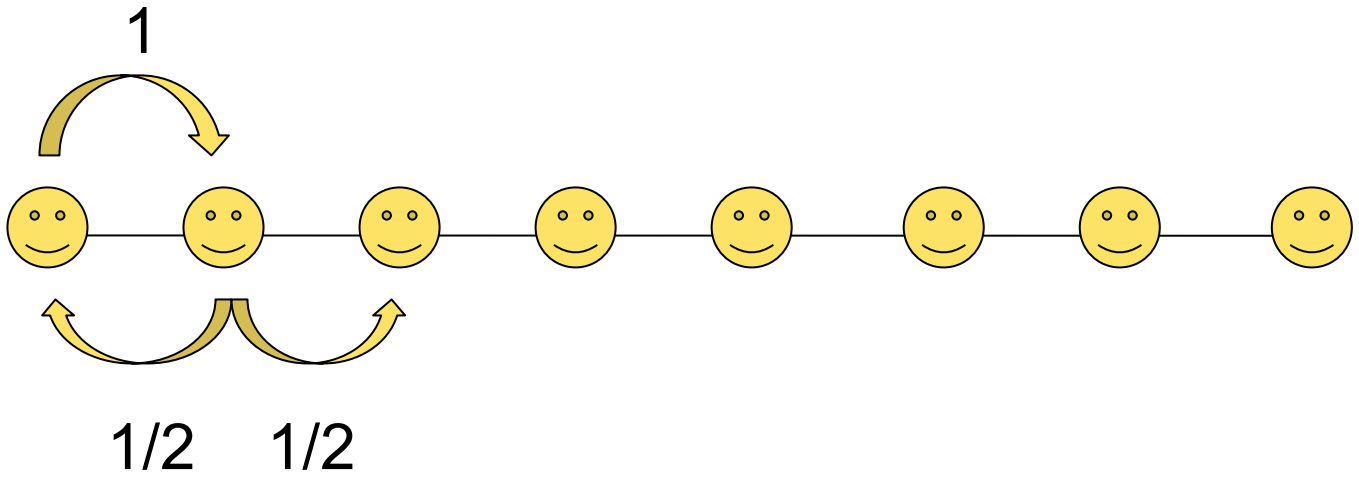
- 3 equivalent descriptions of Lumpability
  - Markovian
  - Spectral properties
    - Piecewise constant r.ev
    - Transition matrix
    - Mean-first-passage
  - Nonidentifiable



- Approximate Graph min-cut
  - Cheeger's inequalities

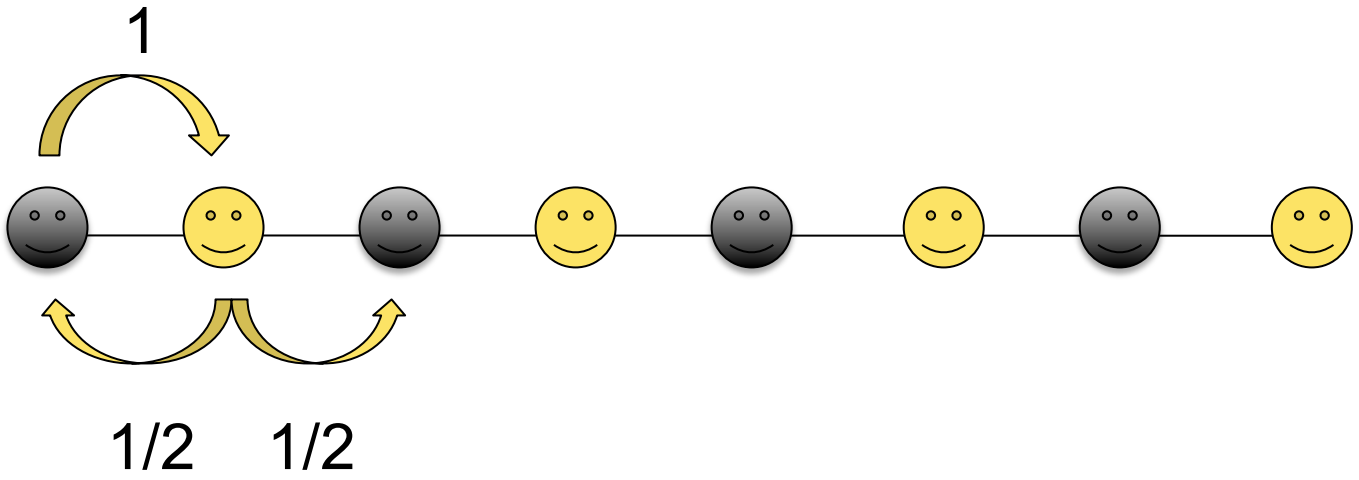
The two theories are different!

Example I



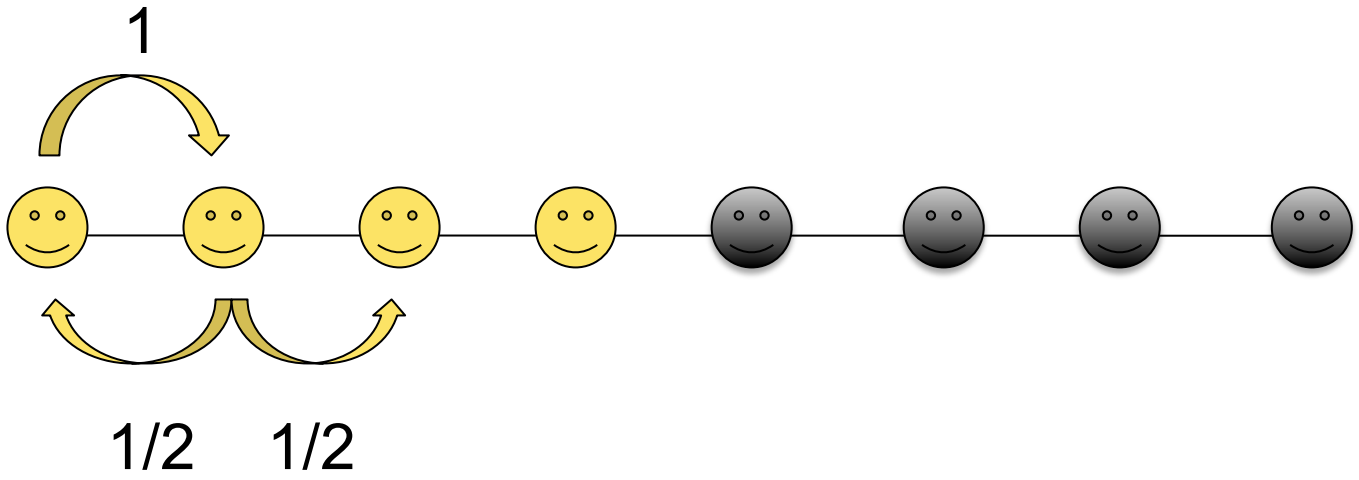
- Consider 2n nodes on a linear chain
- Markov Chain: a node will jump to its neighbors with equal probability
  - T(i, i-1) = T(i, i+1) = ½, for 2n>i>1
  - T(1,2) = T(2n,2n-1) = 1

Example I



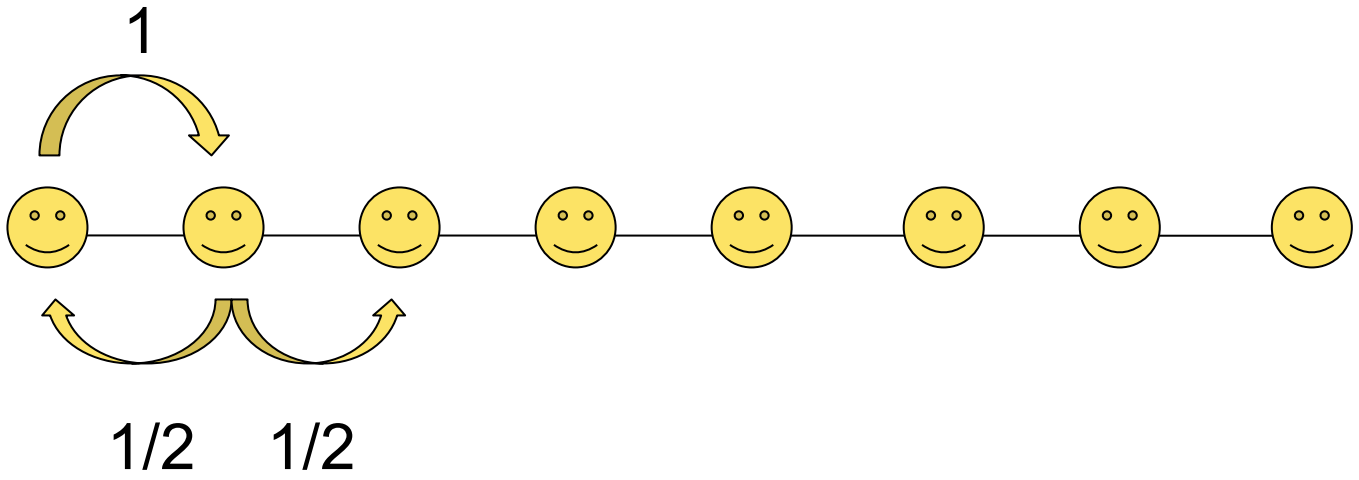
- T is lumpable w.r.t. $P^*=(S_{even},S_{odd})$
  - S_{even}: even nodes
  - S_{odd}: odd nodes
- P^* corresponds to eigenvector with eigenvalue -1

Example I



- One graph min-cut given by second largest right eigenvector of T
- n=8,
  - v_2=[0.4714 0.4247 0.2939 0.1049 -0.1049 -0.2939 -0.4247 -0.4714]
  - Eigenvalue is 0.9010

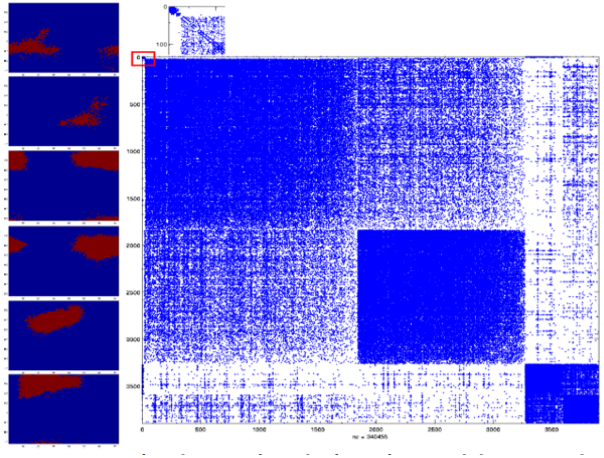
Homework



- Create a 2n-by-2n Transition matrix for the linear chain above
  - T=eye(2*n)-eye(2*n);
  - for i=2:2*n,T(i,i-1)=0.5; T(i-1,i)=0.5;end
  - T(1,2)=1;T(2*n,2*n-1)=1;
  - [v,d]=eigs(T)

When two theories meet?

- [Meila-Shi 2001, E-L-V 2008]
  - If the top k eigenvectors are piecewise constant functions w.r.t. partition $P=\{S_1,\ldots,S_k\}$
  - Or, T is nearly uncoupled Markov chain (nearly block diagonal)



Three algorithms

- Below 3 algorithms will be presented
  - Embedding + K-means
  - Bipartition clustering
  - PCCA (find inner simplex on embedded space)

Basic Algorithm I

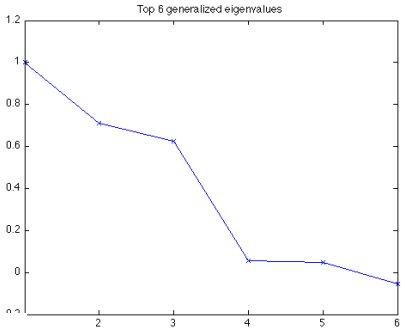
- T under the assumption above
  - Find top k right eigenvectors such that a spectral gap occurs at λ_k, $Tv_i=\lambda_i v_i$, i=1,…,k
  - Define Y = $[v_1, \ldots, v_k]$, an n-by-k matrix which maps n points into R^k
  - Kmeans on Y to cluster n points into k clusters

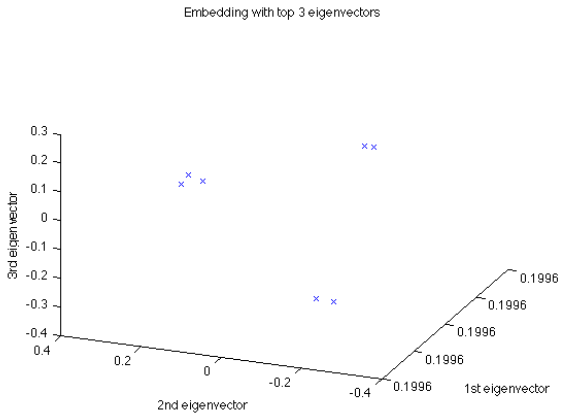
Equivalent to use top k eigenvectors of L=I-D^{-1} W.

Matlab Example

T_1	0	0
0	T_2	0
0	0	T_3

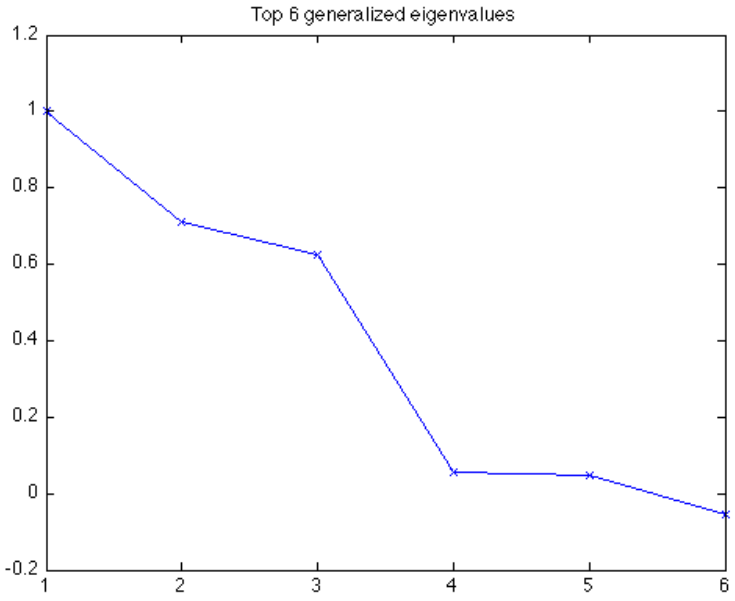
- Construct a nearly 3-block diagonal matrix A,
  - n1=3;
  - n2=2;
  - n3=2;
  - A = zeros(n1+n2+n3,n1+n2+n3);
  - A(1:n1,1:n1)=ones(n1);
  - A(n1+1:n1+n2,n1+1:n1+n2)=ones(n2);
  - A(n1+n2+1:n1+n2+n3,n1+n2+1:n1+n2+n3)=ones(n3);
  - A = A + 0.33*rand(n1+n2+n3,n1+n2+n3); % add 0.33 uniform noise
  - A = (A + A')/2; % make A symmetric
  - D = diag(sum(A,2)); % T=D^{-1} A
- Generalized eigenvector decomposition
  - [v, eeigs(A,D);
- Embedding and $K_{]=}$-means clustering
  - Y = v(:,1:3);
  - Kmeans(Y,3) % one may try kcenter
  - >> kmeans(Y,3)
  - ans =
  - 1
  - 1
  - 1
  - 3
  - 3
  - 2
  - 2

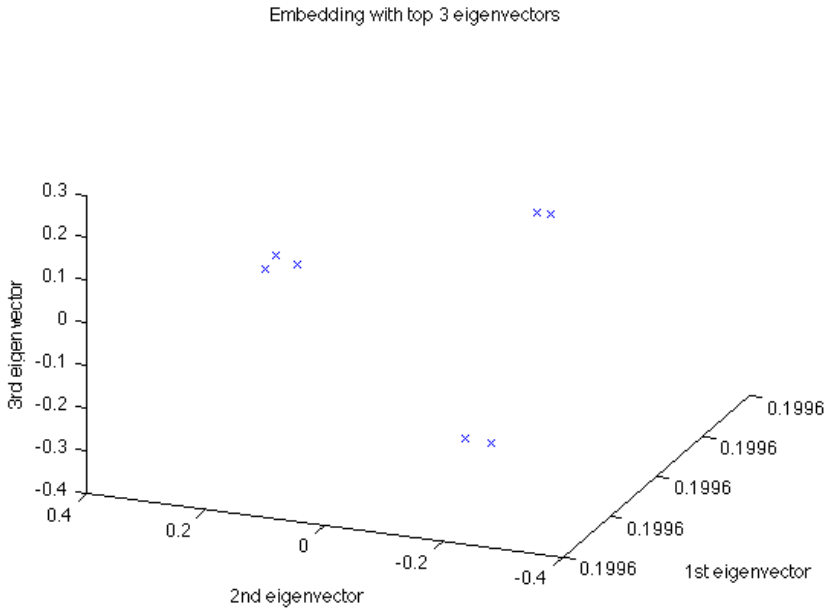




Matlab Example

- plot(diag(e),'-x'), title('Top 6 generalized eigenvalues')
- plot3(Y(:,1),Y(:,2),Y(:,3),'x'), xlabel('1st eigenvector'); ylabel('2nd eigenvector'), zlabel('3rd eigenvector'); title('Embedding with top 3 eigenvectors')





Note that Y maps data around a 2-simplex!

II Spectral Bi-partition as approximate Graph Min-cut

- T under the assumption above
  - Find top k right eigenvectors such that a spectral gap occurs at λk, $Tv_i=\lambda_i v_i$, i=1,…,k
  - fvec = v_2; %Fiedler vector
  - t=2;
  - While t<=k, S_2=S
    - Split current state space by minimal cut using fvec, A={i: fvec(i)<mean(fvec)}; B=S_t-A;
    - u = v(:,t+1); S_{t+1}=argmax$_{(A,B)}$ (var(u(A)),var(u(B))
    - Split S_{t+1} according to min-cut by T(S_{t+1}, S_{t+1})

Matlab Example

- >> help bipartClust
- Spectral Clustering via Bi-partition
- [ci,csize,spectrum,conduct,cluster_tree]=bipartClust(X,k,opt)
-
- Inputs
- X: transition count matrix, X(i,j) is the number of transitons between i and j;
- k: number of clusters,
- [0]|integer
- opt.disp: [0]|1|2, display the process of sparse eigenvalue solver, eigs()
- opt.clusterSizeThreshold: [0,1], threshold of cluster size as percentage of level size
- Outputs
- ci: n-by-1 cluster index vector, 0 for noise
- csize: max(ci)-by-1 cluster size
- spectrum: eigenvalue and eigenvectors of spectral clustering
- conduct: conductance along the spectral cuts
- cluster_tree: the cluster tree

- Try this --
- bipartClust(A,3)

III PCCA

- Perron-Cluster Clustering Analysis [Weber'03,04]: exploit the simplex structure
  - Find k extreme points of the (k-1)-simplex
    - Find two points a,b of Y in largest distance in Rk, S_1={a,b};
    - While t<k,
      - Find the next point,c, in largest distance of dist(v(:,t), span(v(:,S_t))
      - S_t = S_{t-1} + {c};
  - Find approximate membership map, chi=Y*Z, where Z =inv(v(S_k,1:k))
  - Decide the member by maximal association

Matlab Example

- >> help indsearch
- INDEX search for inner simplex representation
- [ind]=indsearch(Evs, NoOfClus)
-
-
- Reference:
- M. Weber, Clustering by using a simplex structure, 2004

- >> help almostinvar
- Find the almost invariant sets
- [Chi, Lambda]=almostinvar(matrix,NoOfClus)
- INPUTS:
- matrix - n-by-n matrix
- NoOfClus - number of clusters
- OUTPUTS:
- Chi - membership function
- Lambda - top NoOfClus eigenvalues
-
- Reference:
- M. Weber, Clustering by using a simplex structure, 2004

Try this –

```
>> indsearch(Y,3)

ans =

     7
     4
     2
```

```
>> chi=almostinvar(A,3)

chi =

    0.0136    0.1431    0.9216
    0.0000    0.0000    1.0000
   -0.0739    0.1062    0.9833
    0.0171    0.9316   -0.0383
         0    1.0000   -0.0000
    0.9239   -0.0512    0.0753
    1.0000   -0.0000   -0.0000
```

Optimal Lumpable Approximation

- Problem: Given a reversible Markov chain T on n-microstates, find an optimal approximation $\check{T}$ lumpable on a k-partition.
  - $\pi_i T_{ij} = \pi_j T_{ji}$, π is stationary distribution
  - microstates $\{1,\ldots,n\}$
  - $\check{T}$ is lumpable on $P_k = \{S_1,\ldots,S_k\}$
  - *What's the optimality condition*?

Metastability Maximization

- Find a k-partition of S, $P = \{S_1,\ldots,S_k\}$, such that the trace of lamped chain is maximized

$$\max_P \text{sum}_{a \text{ in } P} \check{T}_{aa}$$

  - where $\check{T}_{ab} = \text{sum}_{i \text{ in } a,\ j \text{ in } b}\ T_{ij}$
  - *It works when T is nearly uncoupled Markov chains*
  - But the linear chain example, it gives a bi-partition {1,…,n} {n+1,…,2n}!

A General Criterion

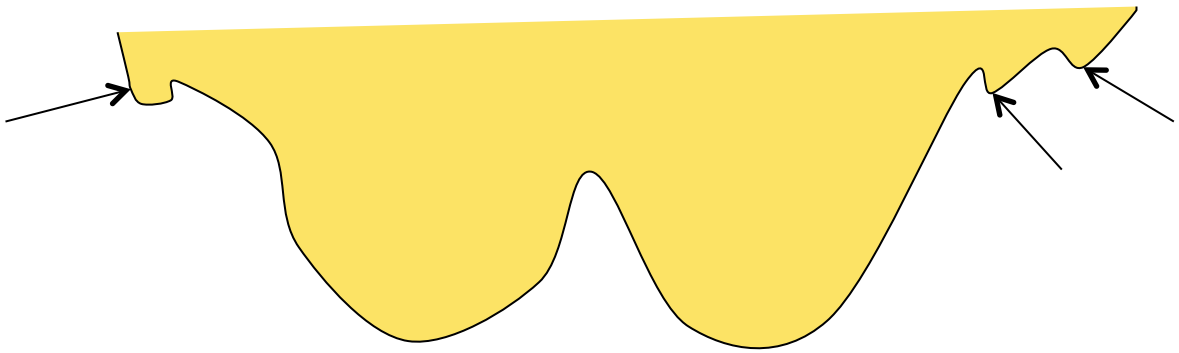
- [E-L-V'08] Find a k-partition of S, P = $\{S_1, \ldots, S_k\}$, such that the following is maximized

$$\max_P \text{sum}_{a,b \text{ in } P} (\check{T}_{ab})^2 \pi_a / \pi_b$$

  - where $\check{T}_{ab} = \text{sum}_{i \text{ in } a,\, j \text{ in } b} T_{ij,}$ $\pi_a = \text{sum}_{i \text{ in } a} \pi_i$
  - *It works when T is lumpable*
  - *When T is nearly uncoupled, the two criteria meet*

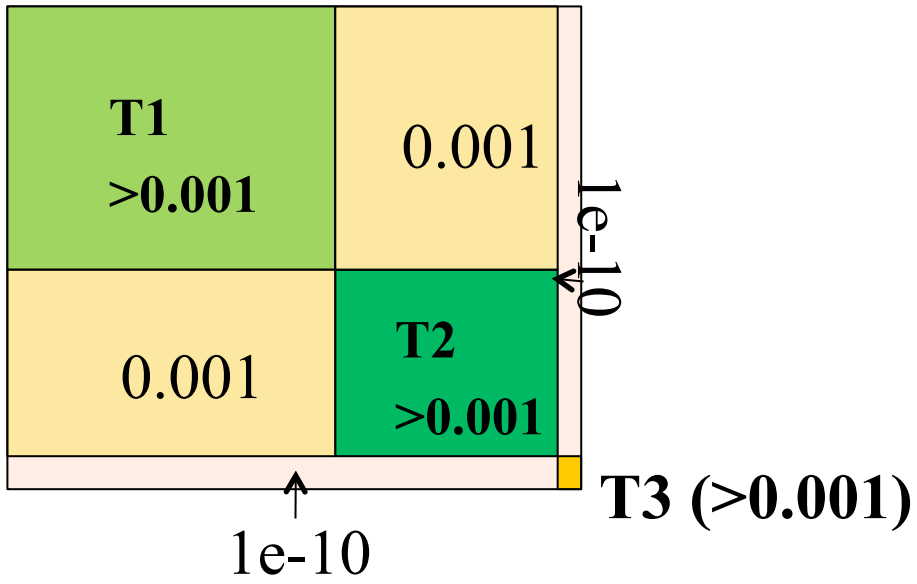
Issues in application

- But all above have issues in application:
  - microstates are given by kcenter algorithm
  - so microstates are metric R-net of samples
  - spurious lumpable states: many microstates on energy barrier are so rarely visited that regarded as approximate lumpable states



Spectral Clustering on Biomolecules

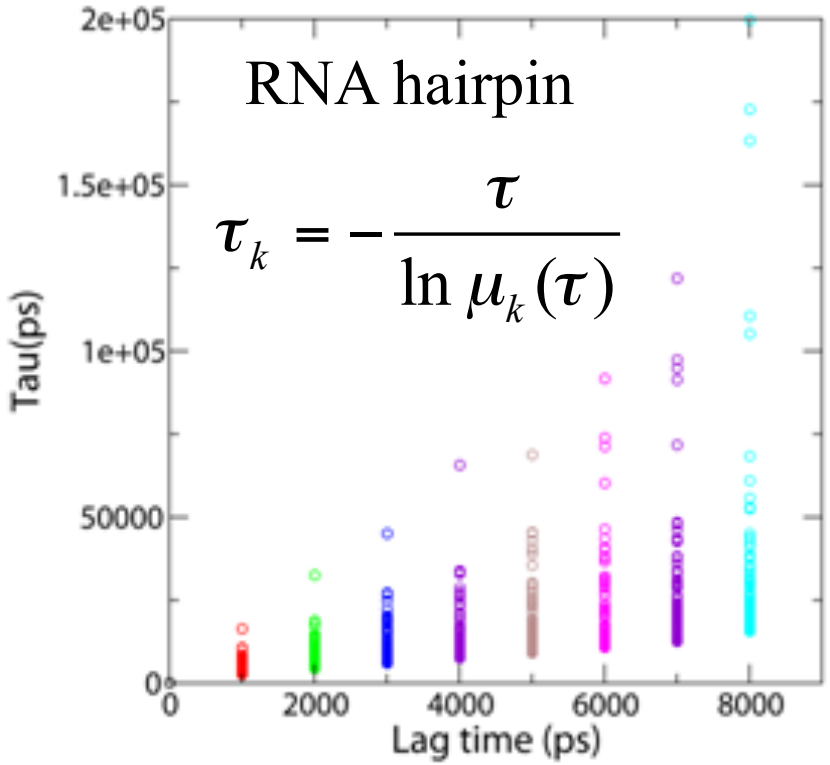
First separate the most disconnected blocks from the transition probability matrix.



Noe and Weber: Left out rarely visited or trapped states

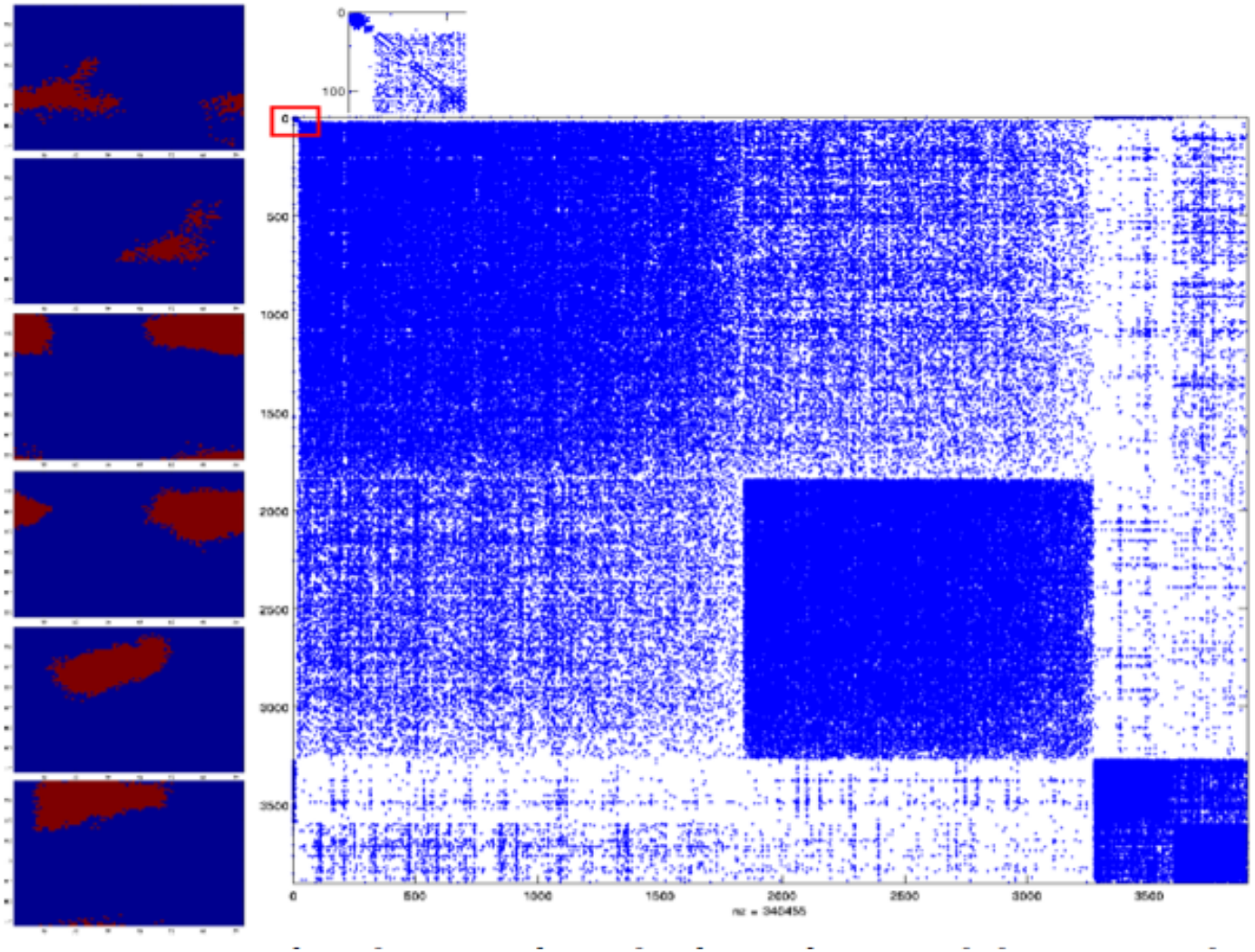
Bowman: Subsample the data

Difficult to determine number of macrostates!



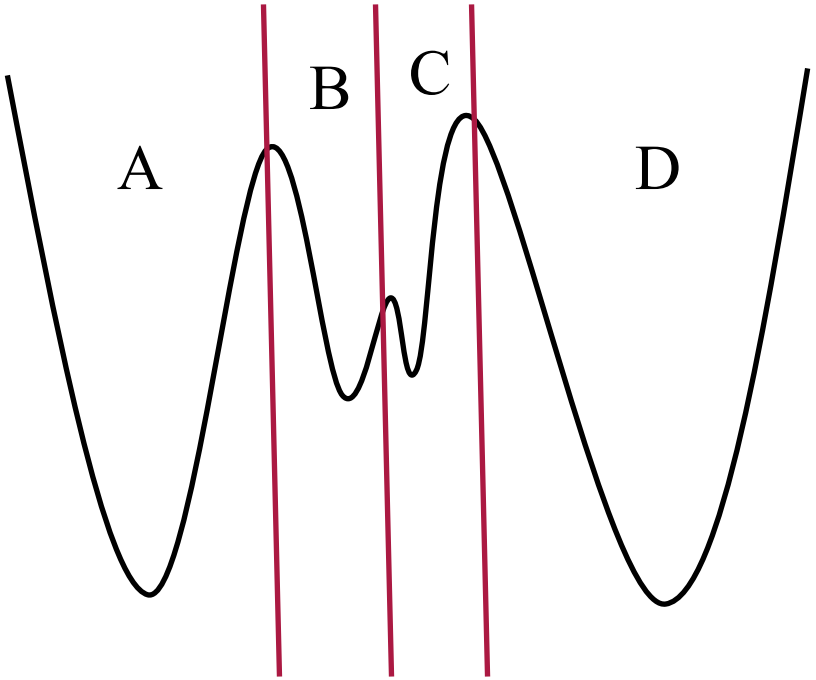
Huang, Bowman, Bacallado, and Pande. 2009

Block Structure is of Multiscale



MSMs are Multi-resolution in Nature

Number of states in an Markov State Model depends on the desired lag time.



MSM #1: A, B+C, D (3 states)

MSM #2: A, B, C, D (4 states)

- A short lag time results in a high resolution MSM having many metastable states, capturing numerous free energy minima separated by small barriers.
- A longer lag time results in a low resolution MSM with only a few states, each of which contains multiple local free energy minima

MSM #1 is a lower resolution model

Nystrom method for the 1st issue

- Let K be n-by-n transition count matrix:
  - K_{ij}, nonnegative integer counting number of transitions from i and j
  - $K_{ij} = K_{ji,}$ every physical system is reversible
  - D=diag(sum(K,2)), a diagonal matrix of row sum of K
  - Sort row/columns of K in a descend order of D_{ii}

Nystrom method for the 1st issue

- Assume K has block-partition

$$K = \begin{bmatrix} A & B \\ B^T & C \end{bmatrix}$$

- Nystrom Approximation, where A's SVD is $A=U \Lambda U^T$

$$\hat{K} = \begin{bmatrix} U \\ B^T U^T \Lambda^{-1} \end{bmatrix} \Lambda \begin{bmatrix} U^T & \Lambda^{-1} U B \end{bmatrix} = \begin{bmatrix} A & B \\ B^T & B^T A^{-1} B \end{bmatrix}$$

  - If Schur Complement $C-B^TA^{-1}B \approx 0$, then a good approximation
  - So we leave in C those microstates with rare visits
  - [U; B'U'inv(Λ)] gives a good approx. of K's top eigenvectors

Nystrom method literature

- Nystrom method is widely used to find approximate eigenvectors with large matrices
  - Image segmentation [Folkes et al. 2004]
  - Machine learning and massive data [Williams et al. 2001, Belabbas and Wolfe 2009]
  - and a lot more if you google…

Det(A) Det(Sc(A)) = Det(K)

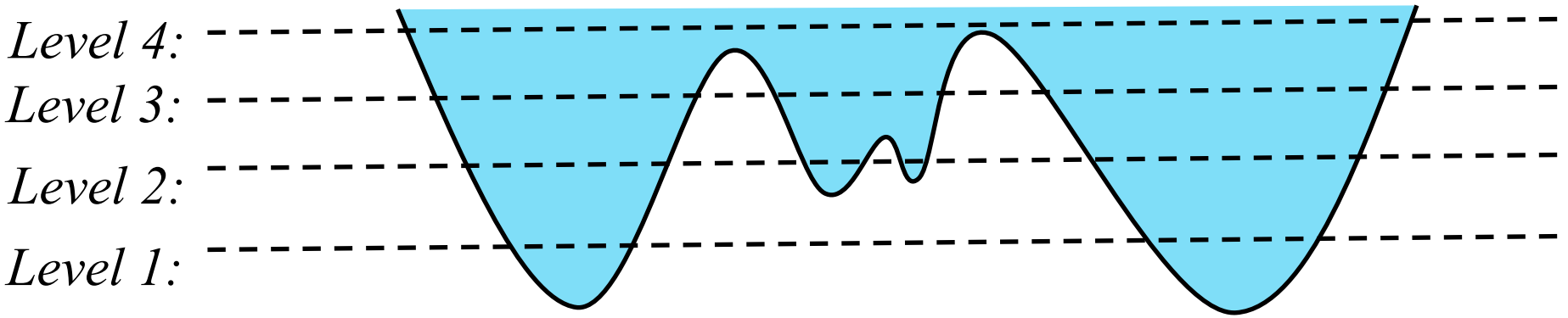
However …

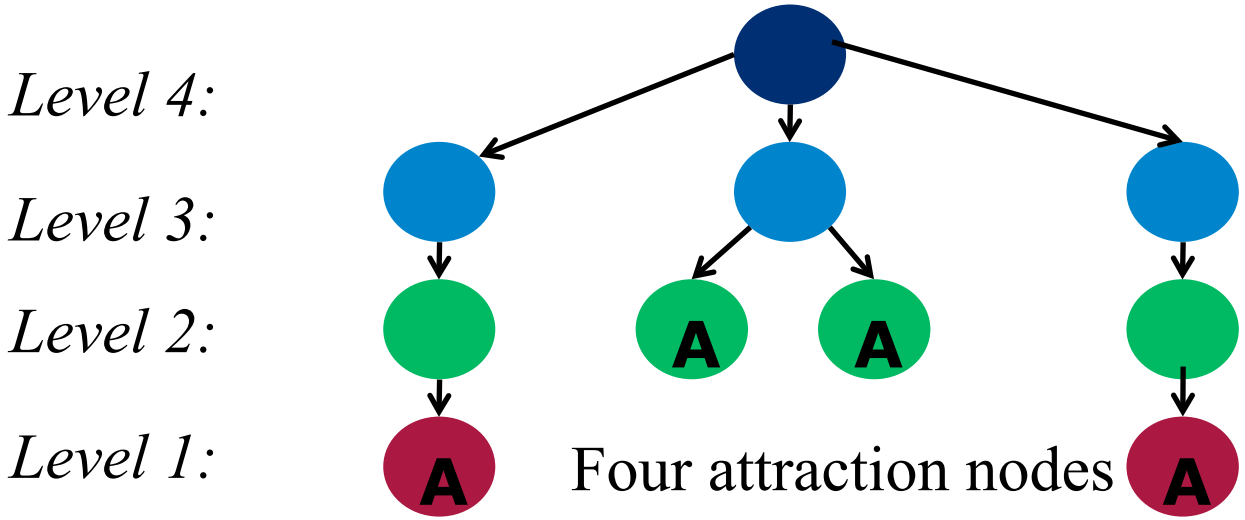
- We don’t know what’s the best cut, if any, between high populated A and low populated C;
- The energy landscape is of multiscale in nature, which prohibits any ‘best’ cut to be the ‘best’;

Answer: Hierarchical Nystrom Method!

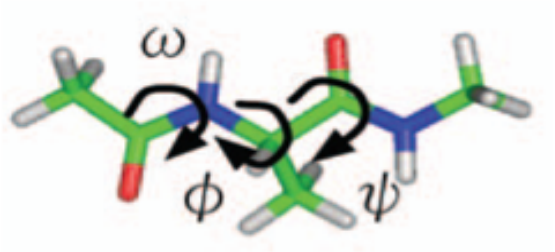
Super-level-set Hierarchical Clustering (SHC, Huang-Y-Sun'09)

Key insight: Cluster conformations hierarchically using super density level sets in a bottom-up fashion

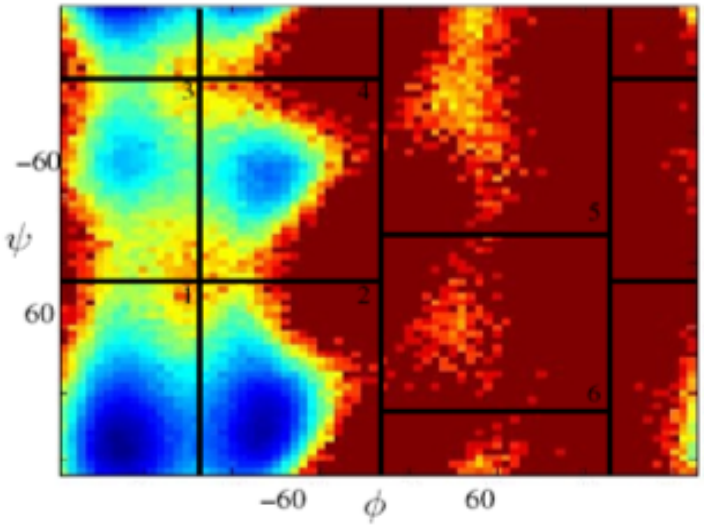




Example: Alanine-dipeptide



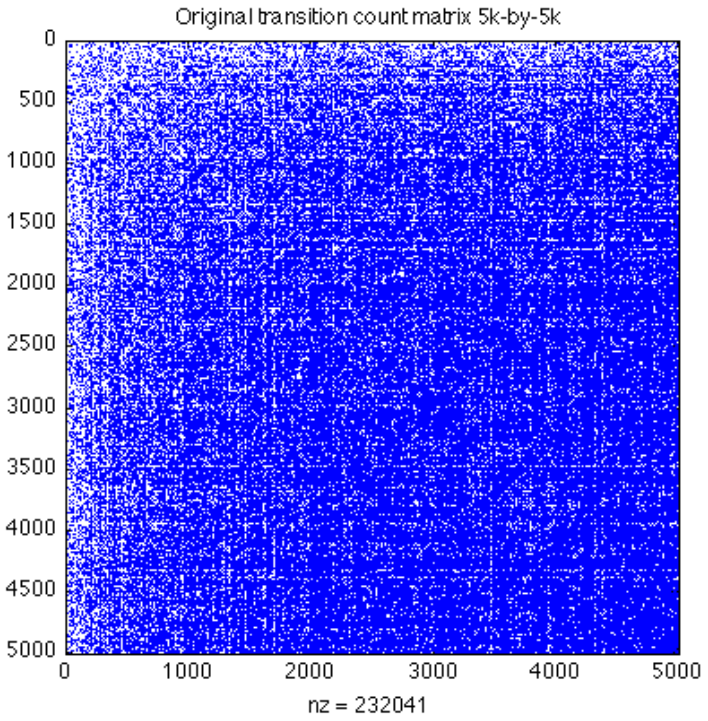
[Chodera et al. 2007]
975 trajectories
200 conformations per trajectory



density on $\phi - \psi$ plane

Example: Alanine-dipeptide

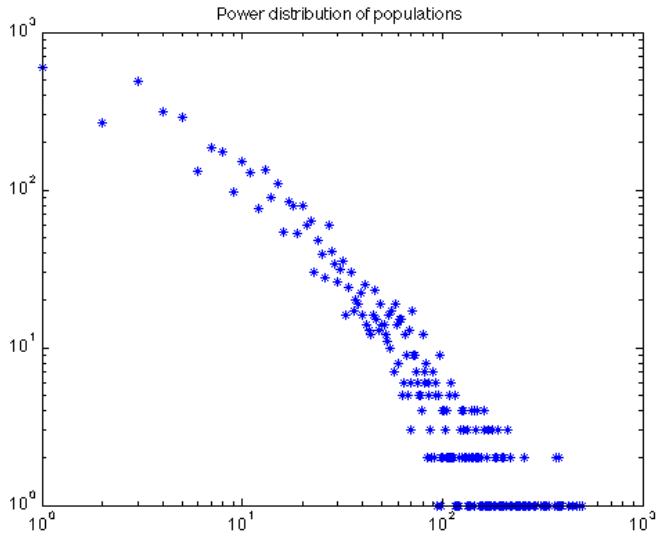
- Load ../data/T5000.mat X % 5000 microstates by kcenter
- spy(X), title('Original transition count matrix 5k-by-5k') % no block structure



Example: Alanine-dipeptide

- d = sum((X'+X)/2,2); % population filter
- filter = d;

- % Power-law distribution of populations
- loglog(hist(d,500),'*'),title('Power distribution of populations')



Example: Ala_nystrom.m

```
%% Choose 9 levels according to percentage of samples
plevels = [0.1:0.1:0.9], % 10%, 20%, 30%, ..., 90% of top populated data
fcdf = hcdf(filter);
for i=1:length(plevels), levels(i) = sum(fcdf<=plevels(i)); end

%% Form superlevel sets.
numLevels = length(levels);
% Compute approximate Betti_0 number
[b,g]=apBetti(X,filter,levels);
clusters = [];
for i=1:numLevels,
    alpha = 10/g{i}(1);
    choice_prob = exp(alpha*g{i}(1:min(3,end))-b{i}(1:min(3,end)));
    clusters(i) = b{i}(find(choice_prob>=max(choice_prob)-eps));
end
superLevelSet = superLvlSet(filter,levels,clusters);
```

Example: Ala_nystrom.m

- %% 1) Call nystrom to construct the Hierarchical Nystrom Extension graph
- opt.clusterSize = 0; % Keep those clusters larger than 0%
- opt.filter = filter;
- [adja, nodeInfo, levelIdx] = nystrom(X, superLevelSet,opt);

- %% 2) Analyze the gradient flow on the nystrom graph
- cpt = flowGrad(adja,levelIdx,nodeInfo);

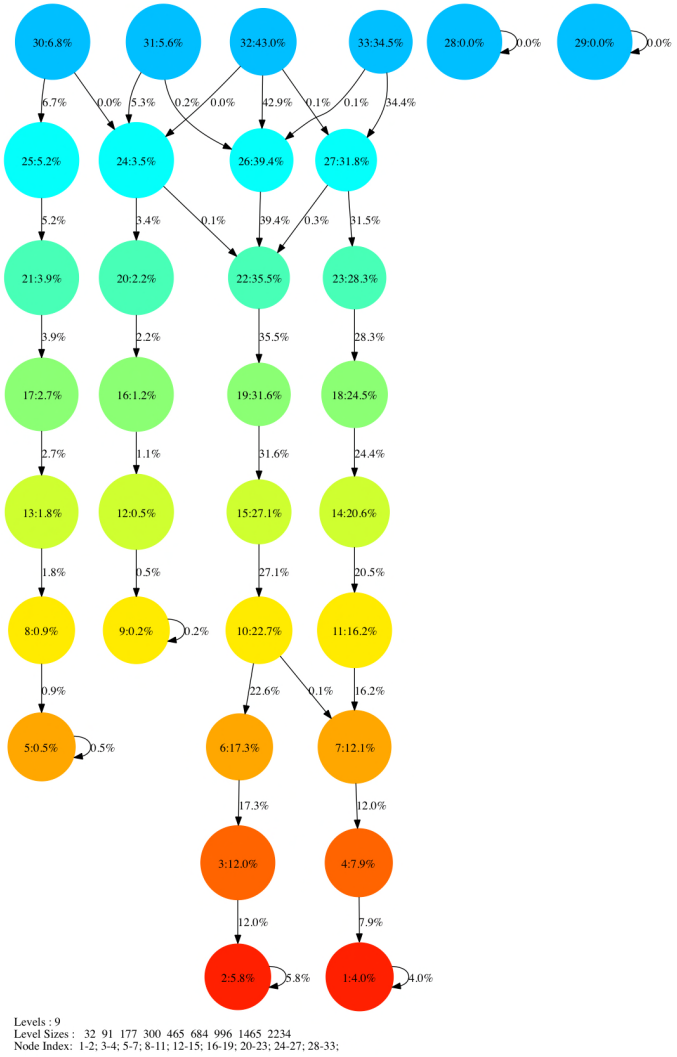
- %% 3) Get optimal assignment of microstates by Diffusion according to X, the transition count matrix
- [ci,csize,fassign,T,Qmax,id_fuzzy]=assign_optim (X,cpt.equilibrium,nodeInfo,opt);

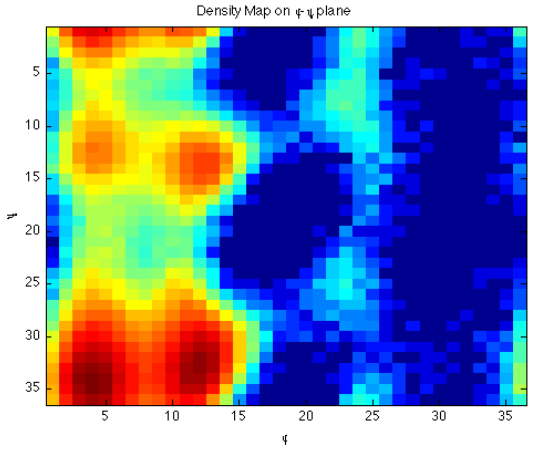
Example: Ala_nystrom.m

- % Write DOT file for Graphviz output
- DotFileName = sprintf('%s/%s_flow',figureDir,fname);
- writeFlowGraph_plus(sprintf('%s.dot',DotFileName), cpt.gradFlow, nodeInfo ,superLevelSet,levelIdx,filter);
- disp(sprintf('%s.dot File done.',DotFileName));

- % Write JPG file, need Graphviz
- % On my macbook air:
- system(sprintf('/usr/local/bin/dot %s.dot -Tjpg > %s.jpg', DotFileName, DotFileName));
- disp(sprintf('%s.jpg File done.',DotFileName));

Outputs of Ala_nystrom.m

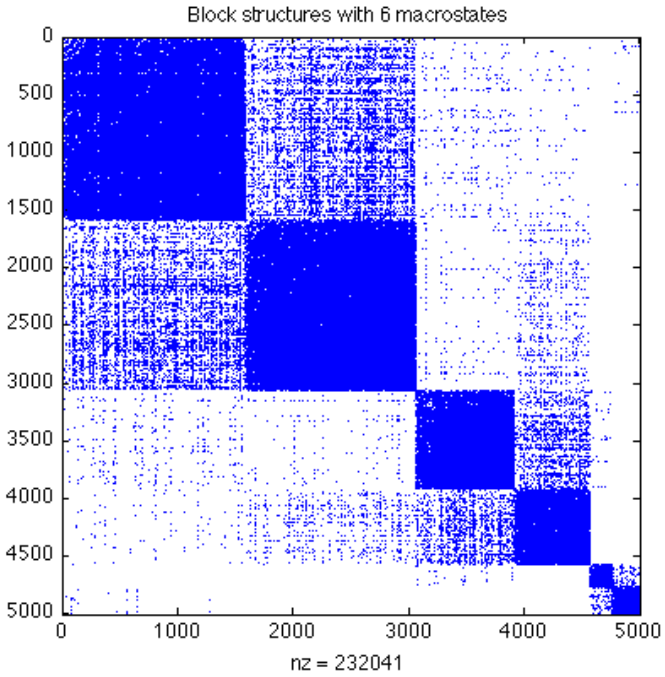


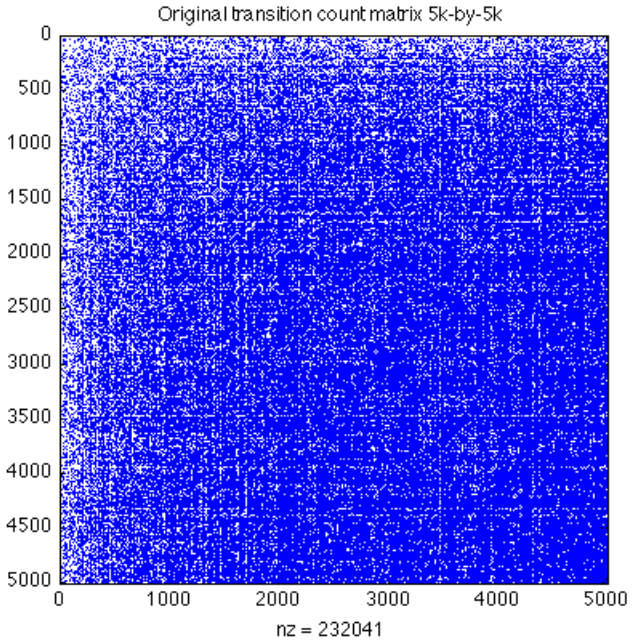


6 macrostates
Metastability=5.56 (92.6%)

Block structure unveiled

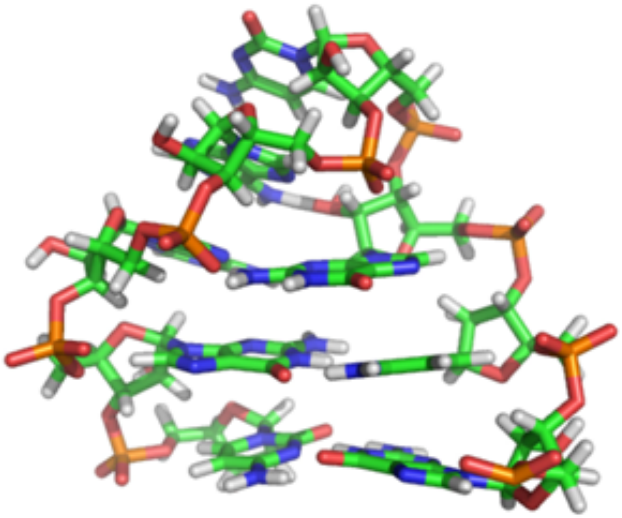
- [ig,id]=sort(ci,'ascend');
- spy(X(id,id)),title('Block structures with 6 macrostates')





Example: 8-RNA hairpin

An eight nucleotide RNA GCAA hairpin

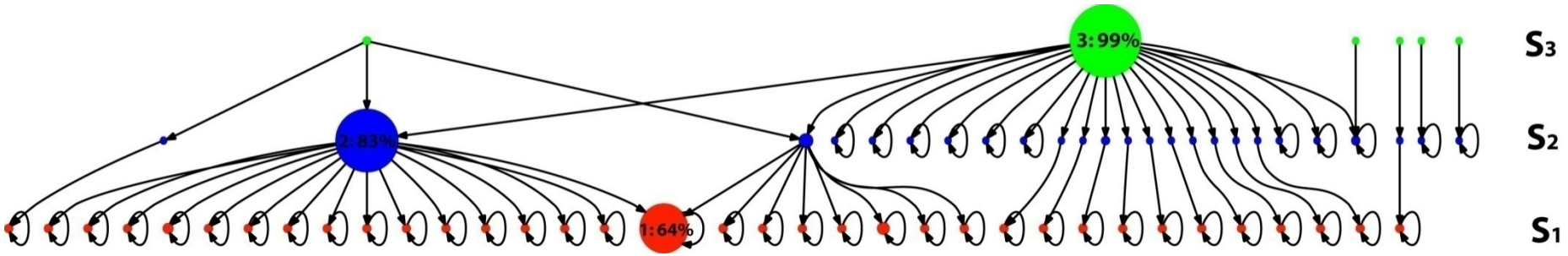


- 2,543 TIP3P waters and 7 Na^+ ions
- 9963 45ns simulations

> 2.3 million conformations in total

Assigning microstates not in attraction nodes

We assign microstates not in attraction nodes to metastable states.

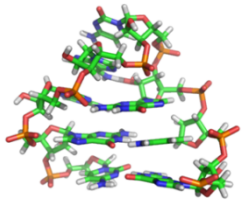


Assign each microstate to the metastable state it has the largest transition probability to.

Huang-Y.-Sun *et.al* *PSB* 2010

MSMs at Different Resolutions for RNA

L: # of super density levels, N: # of states, Q: metastability, $<T_{ii}>=Q/N$, Np: # of states with population > 1%.



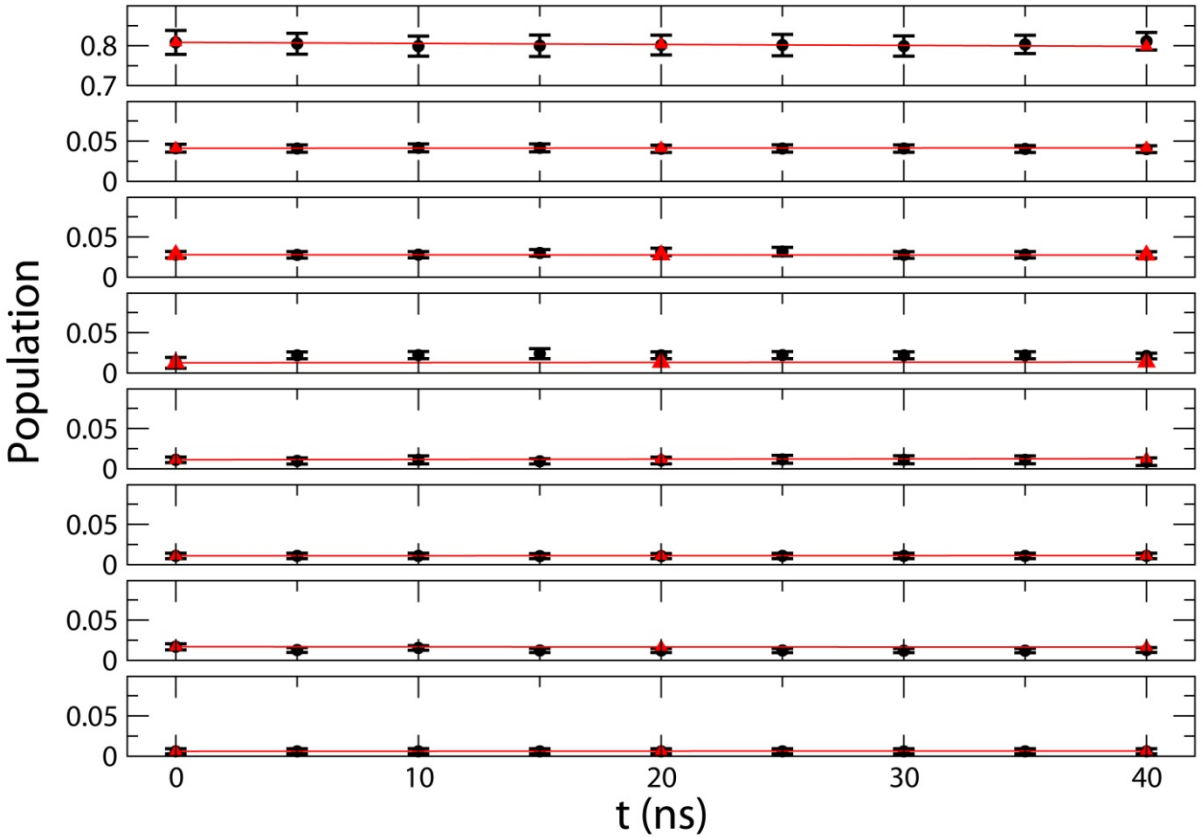
L	1	3	6	9	15
Name	L1MSM	L3MSM	L6MSM	L9MSM	L15MSM
N	6	46	57	63	68
Q	5.95	44.3	54.2	59.3	63.4
$<T_{ii}>$	99.1%	96.3%	95.1%	94.1%	93.2%
Np	2	8	15	12	10

SHC guarantees that highly populated metastable regions are identified before less populated ones.

Huang *et.al* *PSB* 2010, Submitted

Validating MSMs: Chapman-Kolmogorov Check

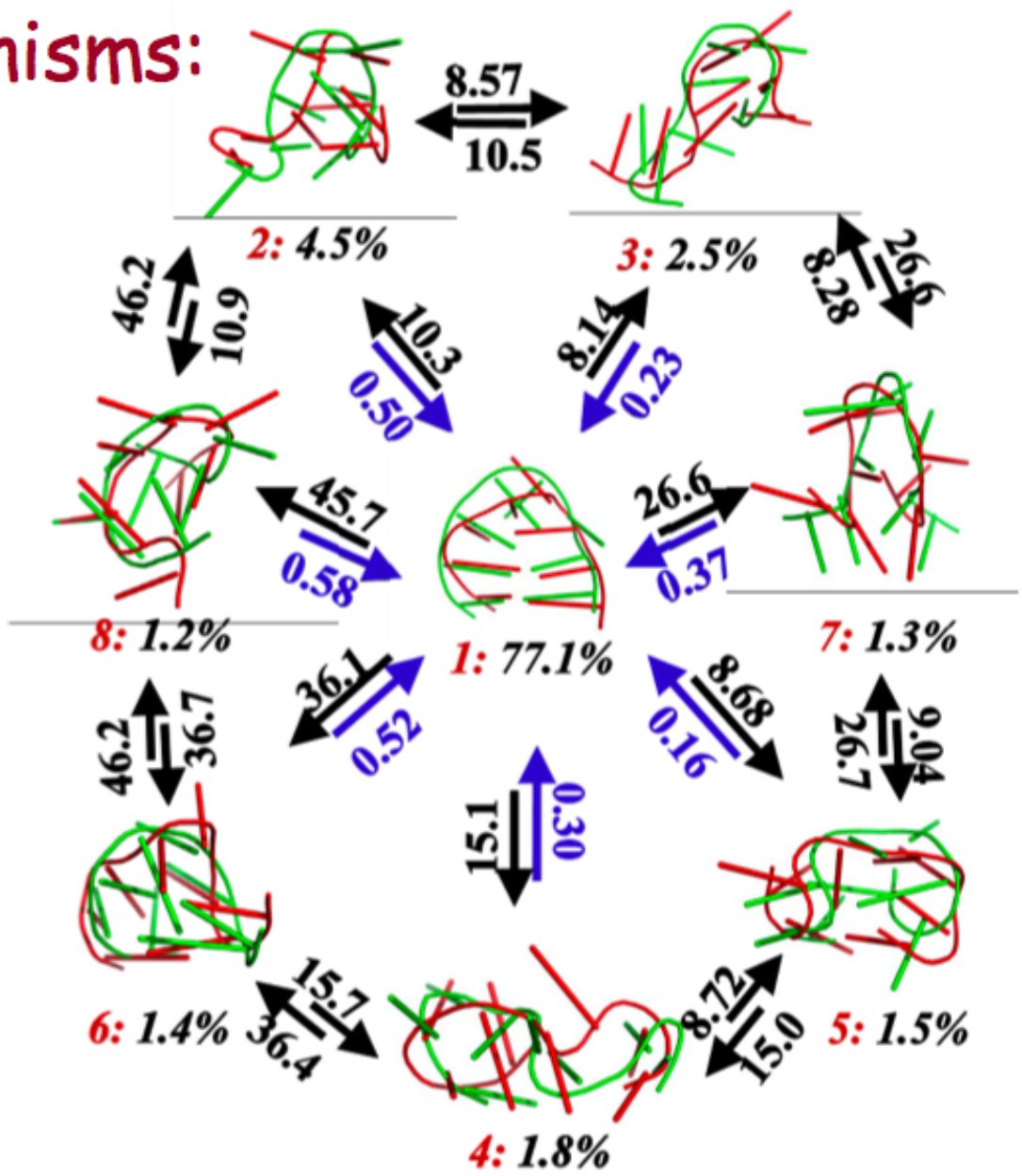
$$P(n\Delta t) = [T(\Delta t)]^n P(0)$$



Black curve: raw data, Red curve: from L3MSM

Folding mechanisms:

- No stable thermodynamic intermediate states
- Overall timescales agree with experiments



Mean First Passage Time in microseconds from L3MSM

Reference

- Shi, Belkin, and Yu, Data spectroscopy: Eigenspaces of convolution operators and clustering. Annals of Statistics, 37 (6B): 3960-3984. 2008.
- Chodera, J. D., Singhal, N., Pande V. S., Dill, K. A., and Swope W. C. (2007) *J. Chem. Phys., **126, 155101.***
- **E, Li, and** Vanden_Eijnden. Optimal partition and effective dynamics of complex networks. PNAS, 105 (23): 7907–7912, 2008.
- T.F. Gonzalez. Clustering to minimize the maximum intercluster distance. Theoretical Computer Science, 38:293-306, 1985.
- J.A. Hartigan. Consistency of single linkage for high-density clusters. Journal of the American Statistical Association, 76:388-394, 1981.
- Kemeny and Snell 1976. Finite Markov Chains. Springer-Verlag.
- Koutis, Miller, and Peng. Approaching Optimality For Solving SDD Linear Systems, arXiv: 1003.2958v2, 2010.
- Meila and Shi, A random walk view of spectral segmentation, AISTATS 2001.
- Andrew Y . Ng, Michael I. Jordan and Yair Weiss. On spectral clustering: Analysis and an algorithm. Advances in Neural Information Processing Systems (NIPS) 14, 2002.
- D. Pollard. Strong consistency of k-means clustering. Annals of Statistics, 9(1):135-140, 1981
- W. Stuetzle. Estimating the cluster tree of a density by analyzing the minimal spanning tree of a sample. Journal of Classification, 20(5):25-47, 2003.
- Spielman and Teng. Nearly-Linear Time Algorithms for Preconditioning and Solving Symmetric, Diagonally Dominant Linear Systems, **arXiv:cs/0607105v4, 2009**.

Reference

- Deuflhard, P. and M. Weber, Robust Perron Cluster Analysis in Conformation Dynamics, ZIB-Report 03-19, 2003.
- Weber, M. Clustering by using a simplex structure, ZIB-Report 04-03, 2004
- Ulrike von Luxburg, A Tutorial on Spectral Clustering, Max Planck Institute for Biological Cybernetics, TR-149, 2006
- Charless Fowlkes, Serge Belongie, Fan Chung, Jitendra Malik, Spectral Grouping Using the Nyström Method, IEEE Transactions on Pattern Analysis and Machine Intelligence, 26(2) 2004
- C. Williams and M. Seeger, “Using the Nystrom Method to SpeedUp Kernel Machines,” Advances in Neural Information Processing Systems (NIPS) 13: Proc. 2000 Conf., T.K. Leen, T.G. Dietterich, and V. Tresp, eds., pp. 682-688, 2001
- Mohamed-Ali Belabbas and Patrick J. Wolfe, Spectral methods in machine learning and new strategies for very large datasets, PNAS 2009.