MCL: Markov Clustering Algorithm

Slides are heavily modified from

Wang, Qichen
Chu Kochen Honors College
Zhejiang University
(08 Aug 2015)

Markov Clustering (MCL)

- **van Dongen 2000**
  (PhD thesis, University of Utrecht, 2000)
- **Highly scalable and fast, and popular**

Key idea:

Random walker "stuck" in dense regions

Introduction

- Random Walks
- Markov Chain
- Markov Clustering

  http://micans.org/mcl
- Discussion and Remarks

Random Walks

Key Idea:

If you choose a vertex and randomly walk in the graph, it is more likely for you to *stay within a cluster*, than for you to walk *between two clusters*.

So, by many doing random walks, it might be possible for us to *discover* the clusters.

Introduction

- ❑ Random Walks
- ❑ Markov Chain
- ❑ Markov Clustering

  http://micans.org/mcl
- ❑ Discussion and Remarks

Markov (Chain) Models

Modelling Random Walk: Markov Model

- A sequence of variables $X_1, X_2, X_3, \ldots$ (in our case, the probability matrices).
- Given the present state, the past and future states are independent.
- Probabilities for the next state only depend on transition probabilities.

Markov Chain

❑ A simple example:

	State1	State2
State1	0.6	0.5
State2	0.4	0.5

❑ Begin from State S1, after two steps,

	State1	State2
State1	0.6	0.5
State2	0.4	0.5

*

	State1	State2
State1	0.6	0.5
State2	0.4	0.5

Markov Chain

- The result:

	State1	State2
State1	0.56	0.55
State2	0.44	0.45

- How about after *n* steps? (for large *n*)

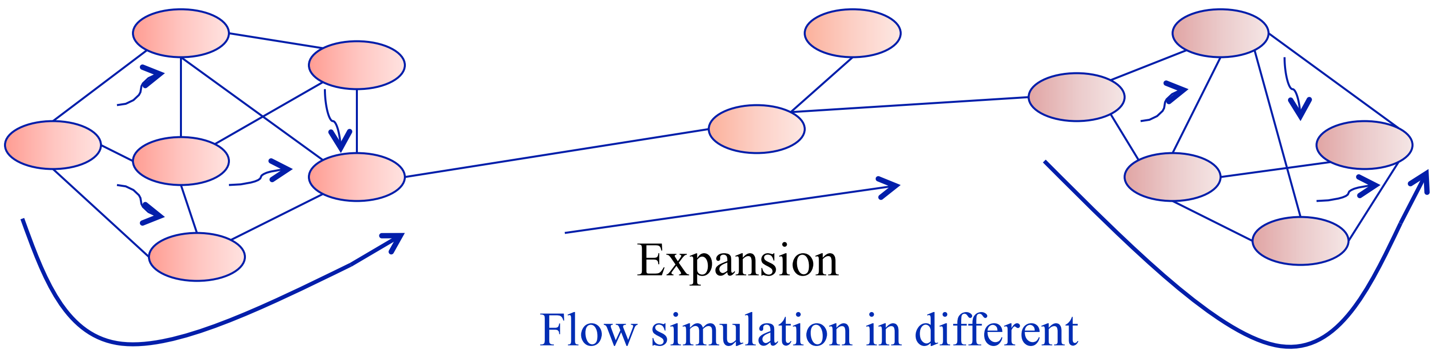
	State1	State2
State1	5/9	5/9
State2	4/9	4/9

Reached steady state

Introduction

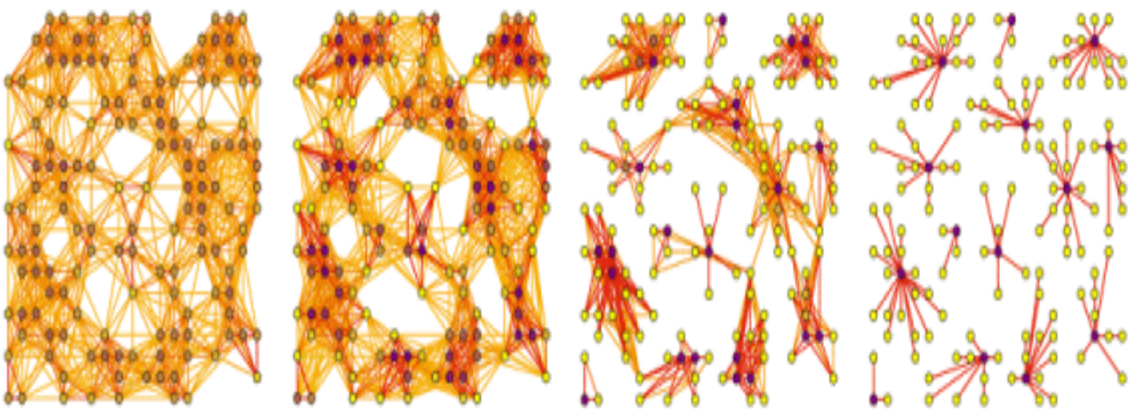
- Random Walks
- Markov Chain
- Markov Clustering
  http://micans.org/mcl
- Discussion and Remarks

MCL (van Dongen, 2000)



Inflation (I)

Flow simulation in different regions of the network



Repeated inflation and expansion separates the network into multiple dense regions

Dongen, PhD Thesis, CWI, Netherlands, 2000

Random walks with Markov Chain

Algorithm:

Given two parameter e $(e > 1)$ and r.

1. Normalize the adjacency matrix; get probability matrix M
2. **Expand** by taking the e^{th} power of the matrix. $M \leftarrow (M)^e$
3. **Inflate** the resulting matrix M with parameter r
4. Repeat 2 & 3 until the matrix M become stable.
5. Analyze the resulting matrix to discover clusters.

Expand & Inflate

Expand the matrix *M*: (take more "steps")

$$M = (M)^e$$

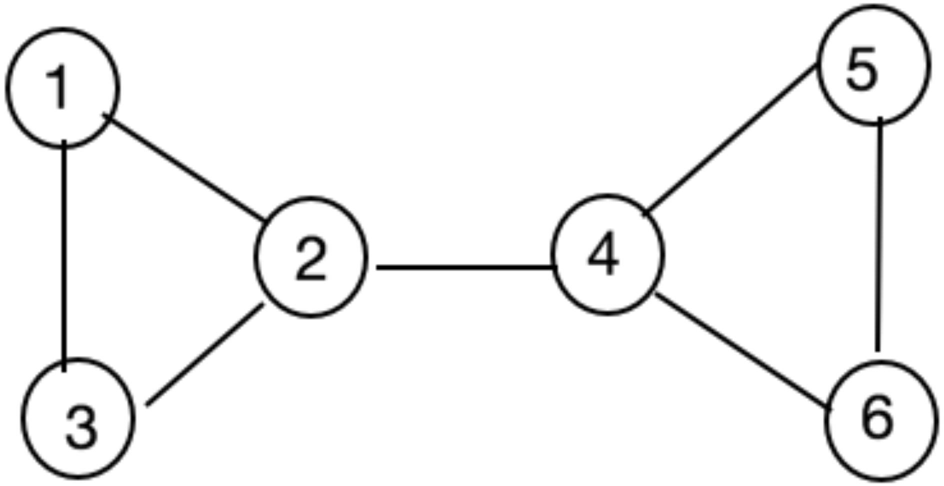
Inflate entries in *M*: (boost intra, reduce inter)

Inflate r means for all entries M_{ij}

$$M_{ij} = (M_{ij})^r$$

Re-normalize M.

MCL Algorithm: (adj matrix)



	1	2	3	4	5	6
1	0	1	1	0	0	0
2	1	0	1	1	0	0
3	1	1	0	0	0	0
4	0	1	0	0	1	1
5	0	0	0	1	0	1
6	0	0	0	1	1	0

MCL Algorithm: (normalize)

- First we normalize the matrix M:

	1	2	3	4	5	6
1	0	0.33	0.5	0	0	0
2	0.5	0	0.5	0.33	0	0
3	0.5	0.33	0	0	0	0
4	0	0.33	0	0	0.5	0.5
5	0	0	0	0.33	0	0.5
6	0	0	0	0.33	0.5	0

MCL Algorithm: (expand)

- Expand the matrix by e=2

	1	2	3	4	5	6
1	0.415	0.165	0.165	0.1089	0	0
2	0.25	0.4389	0.25	0	0.165	0.165
3	0.165	0.165	0.415	0.1089	0	0
4	0.165	0	0.165	0.4389	0.25	0.25
5	0	0.1089	0	0.165	0.415	0.165
6	0	0.1089	0	0.165	0.165	0.415

MCL Algorithm: (inflate, renorm)

- Inflate the matrix by r=2

	1
1	0.415
2	0.25
3	0.165
4	0.165
5	0
6	0

	1
1	0. 172225
2	0. 0625
3	0. 027225
4	0. 027225
5	0
6	0

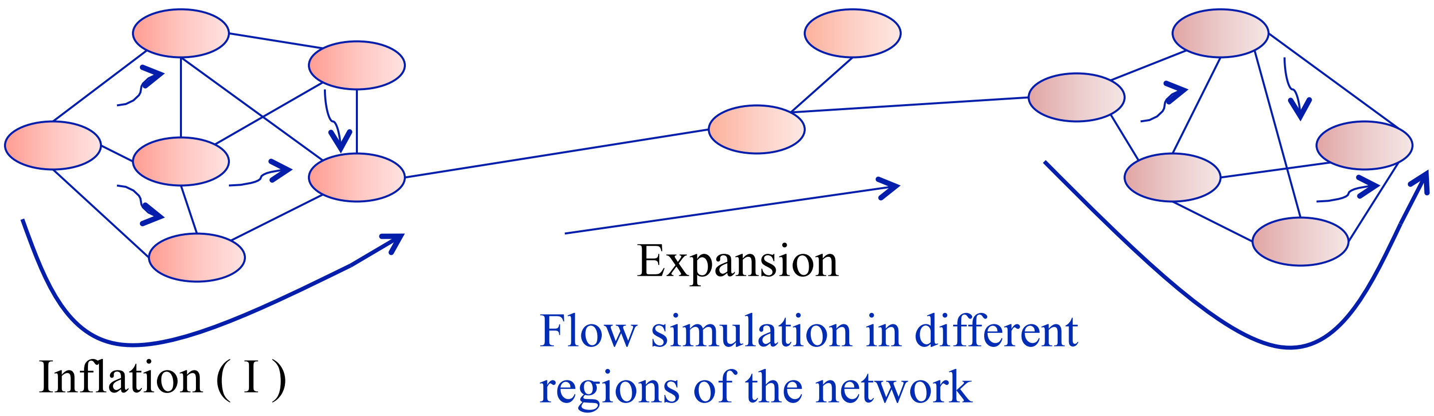
	1
1	0.60
2	0.22
3	0.09
4	0.09
5	0.00
6	0.00

MCL Algorithm: (in steady state)

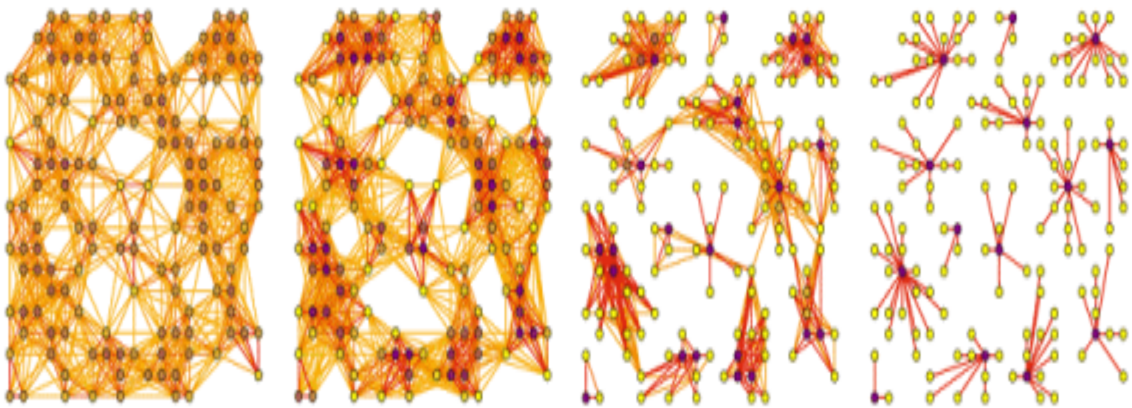
- Finally, after many interations…

	1	2	3	4	5	6
1	0	0	0	0	0	0
2	1	1	1	0	0	0
3	0	0	0	0	0	0
4	0	0	0	1	1	1
5	0	0	0	0	0	0
6	0	0	0	0	0	0

MCL (van Dongen, 2000)



Flow simulation in different regions of the network



Repeated inflation and expansion separates the network into multiple dense regions

Dongen, PhD Thesis, CWI, Netherlands, 2000

Introduction

- ❑ Random Walks
- ❑ Markov Chain
- ❑ Markov Clustering

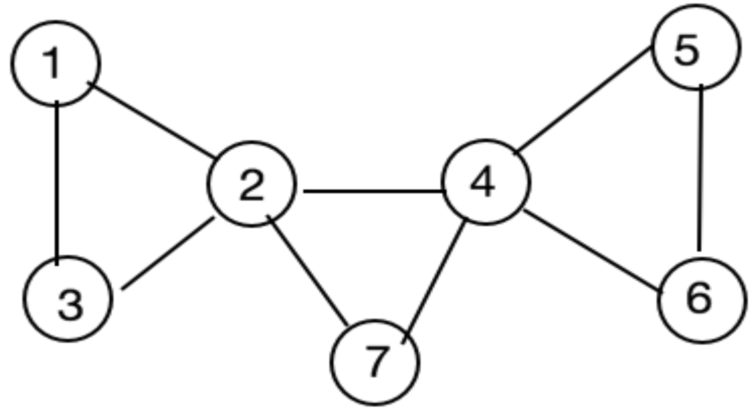
  http://micans.org/mcl
- ❑ Discussion and Remarks

Discussion & Areas for Further Work

- How to determine the r & e?
  - r & e should not be too large (Why?)
- What's the complexity of this algorithm?
  - $O(n^{2.x} + n^2)$
- How to improve the accuracy and efficiency?

Discussion & Areas for Further Work

- Can it work on a graph like this?



	1	2	3	4	5	6	7
1	0	0	0	0	0	0	0
2	1	1	1	0	0	0	0.5
3	0	0	0	0	0	0	0
4	0	0	0	1	1	1	0.5
5	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0

MCL (van Dongen, 2000)

- **Fast and scalable**
- **Robust to noise in datasets**
  - **Can tolerate random noise**
- **Reasonable precision and recall**
- **Produces *non-overlapping* clusters**
- **Tends to "lump up" small closely interacting clusters**

Thank you!