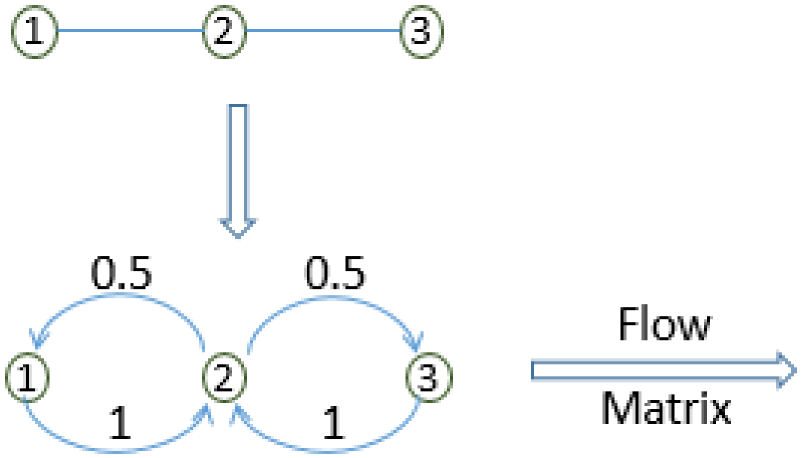
L26_SNA

Terminology

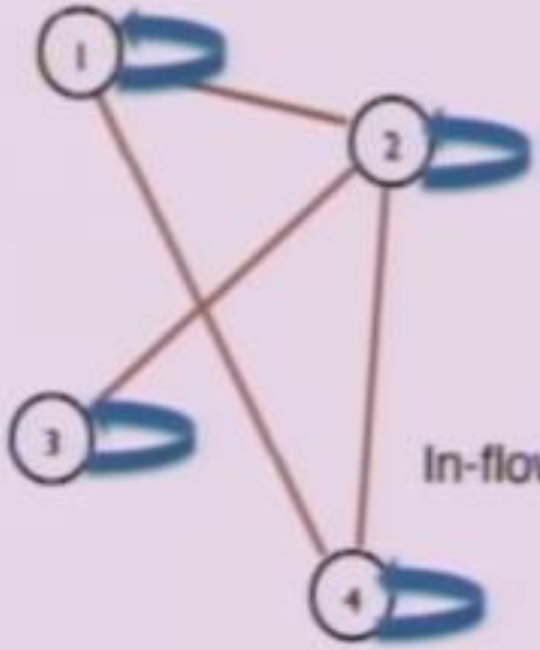
- **Flow**: M(j, i) represents transition probability from node i to node j.
- **Flow matrix**: Matrix with the flows among all nodes; i^{th} column represents flows out of i^{th} node. If each column sums to 1, we call it as **canonical transition matrix.**



	1	2	3
1	0	0.5	0
2	1.0	0	1.0
3	0	0.5	0

Column Stochastic Matrix: **A matrix where each column sums to 1.**

Stochastic Flow: **An entry in a column stochastic matrix, interpreted as the "flow" or "transition probability".**

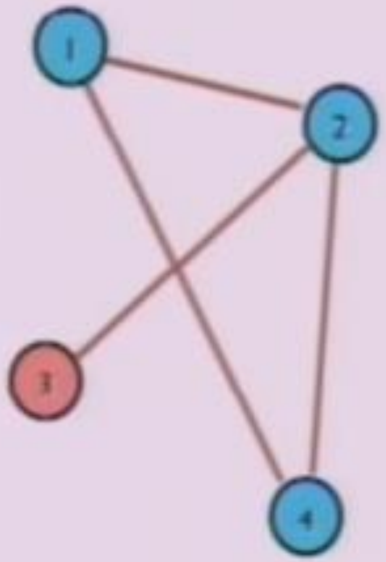


In-flows of 2

	1	2	3	4
1	0.33	0.25		0.33
2	0.33	0.25	0.5	0.33
3		0.25	0.5	
4	0.33	0.25		0.33

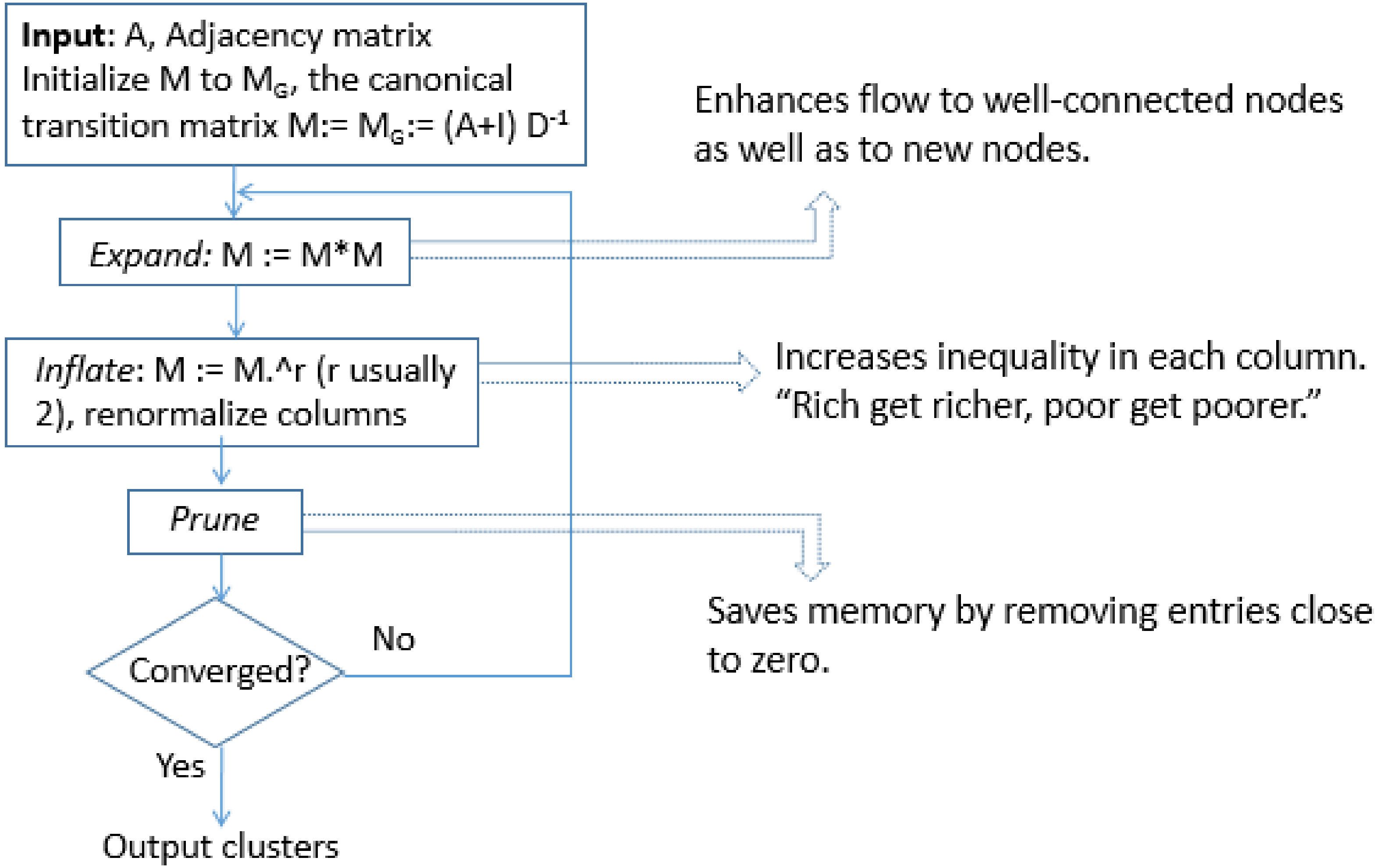
Out-flows of 2

Repeatedly apply certain operations to the flow matrix until the matrix converges and can be interpreted as a clustering.



	1	2	3	4
1				
2	1.0	1.0		1.0
3			1.0	
4				

The MCL algorithm

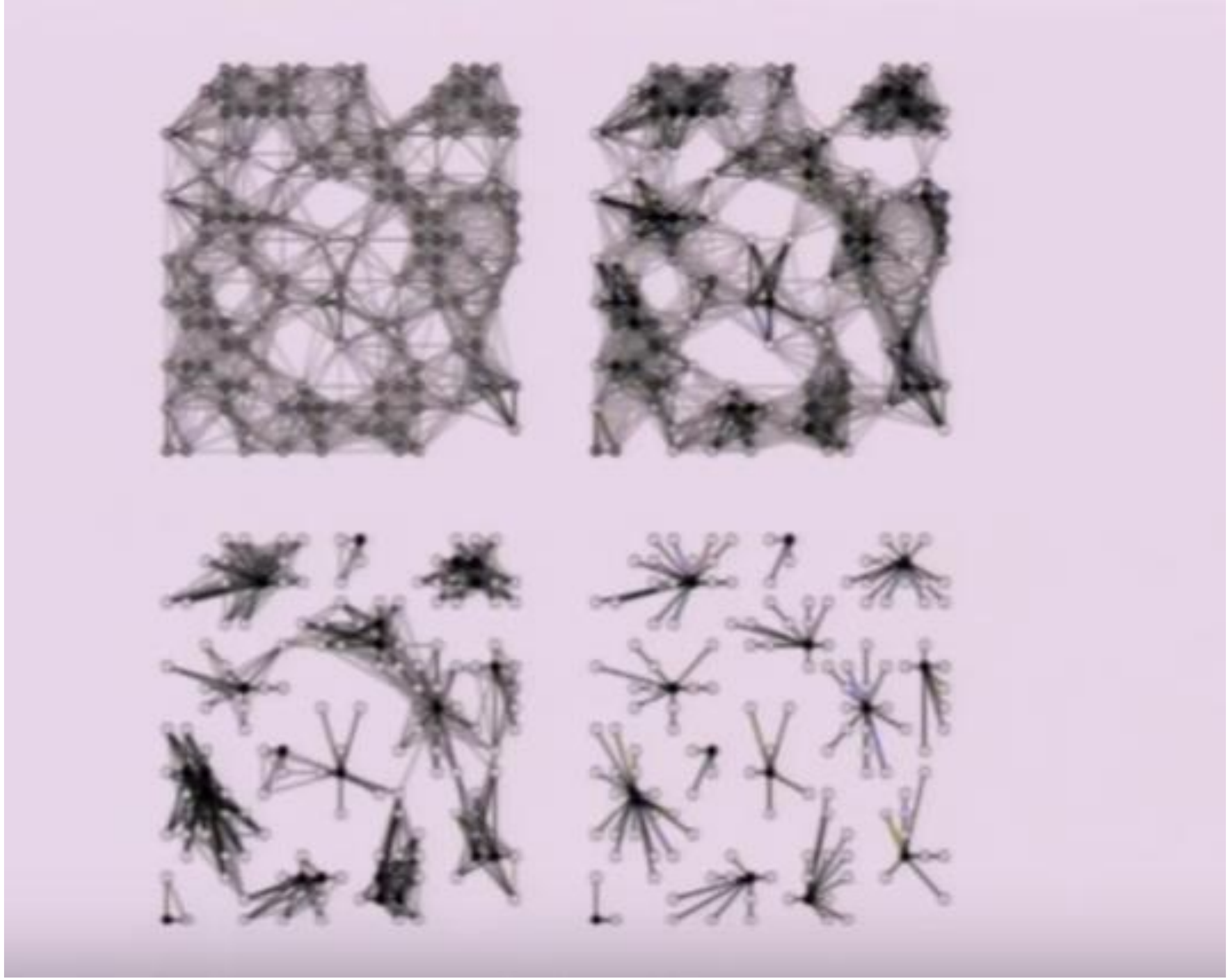


Markov Clustering (MCL) [van Dongen '00]

The original algorithm for clustering graphs using stochastic flows.

Advantages:

- Simple and elegant. No cluster number required.
- Widely used in Bioinformatics because of its noise tolerance and effectiveness.



MCL Flaws

1. Outputs many small clusters.

   Fix I: Regularized MCL

2. Does not scale well.

   Fix II: Multi-Level Regularized MCL
   Fix III: Localized Graph Sparsification

Key Idea I: The *Regularize* operator

Why does MCL output many clusters?
Due to overfitting; it does not penalize divergence of flows between neighbors.

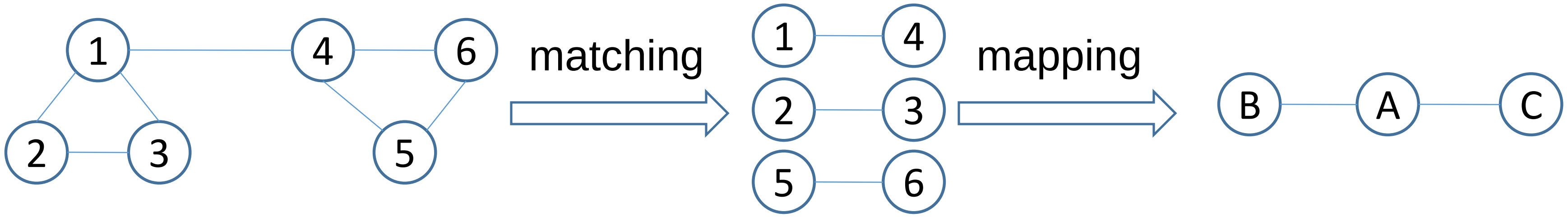
Remedy: Penalize divergence in flows between neighbors. Use KL Divergence (a well known measure for comparing probability distributions).

Turns out to have a nice closed form solution:

$$Regularize(M) := M*(A+I)D^{-1} = M*M_G$$

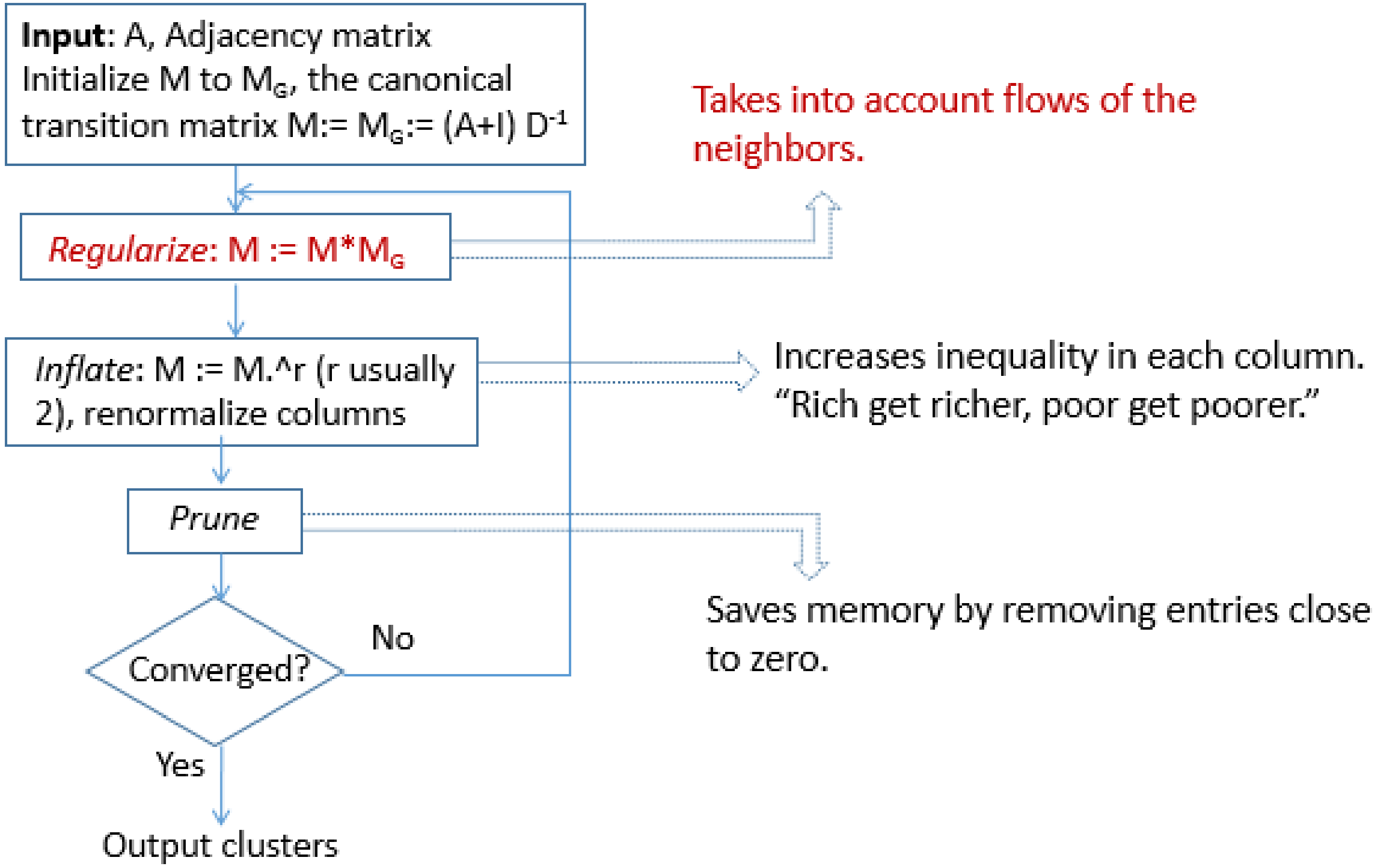
Coarsening operation

- Construct a matching: defined as a set of edges, no vertex is shared among these edges.
- Each edge is mapped into a super-node in the coarsened graph, and the new edges are the union of the original ones.
- Two maps used to keep the track of the process



	A	B	C
Map1:	1,	2,	5
Map2:	4,	3,	6

The Regularized-MCL algorithm



Multi-level Regularized MCL