CLIQUE PERCOLATION METHOD (CPM)

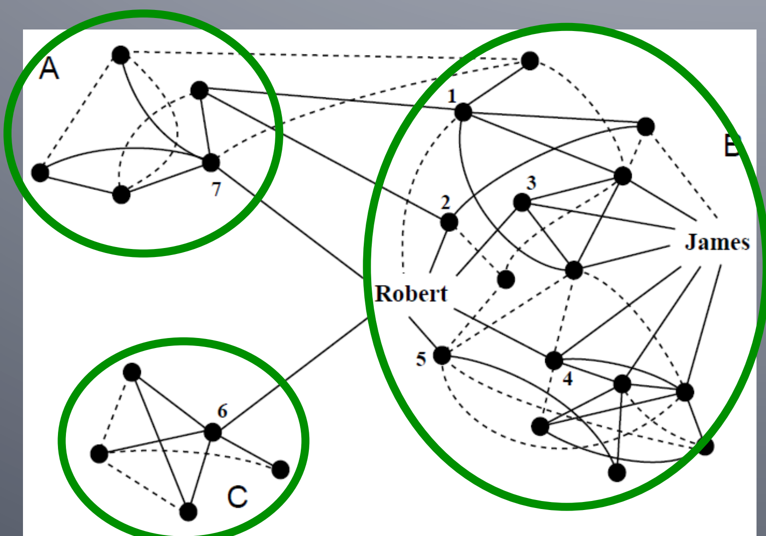
Slides mostly by

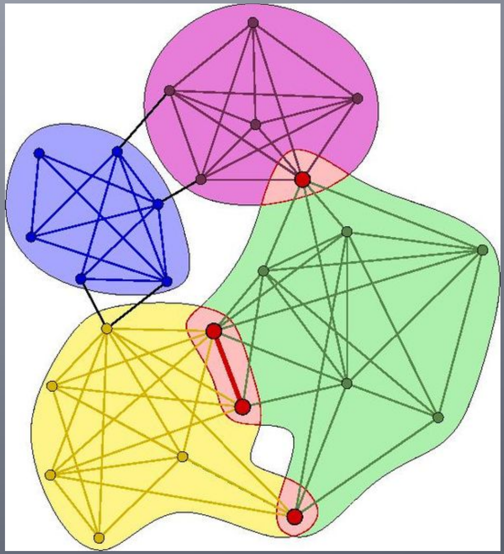
Eugene Lim

CONTENTS

Community Detection

How to find communities?





We will work with **undirected** (unweighted) networks

WHAT IS CPM?

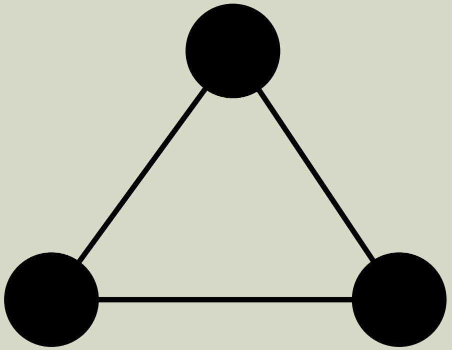
- Method to find <u>overlapping</u> communities
- Based on concept:
  - internal edges of community likely to form cliques
  - Intercommunity edges unlikely to form cliques

CLIQUE

- Clique: Complete graph
- k-clique: Complete graph with k vertices

CLIQUE

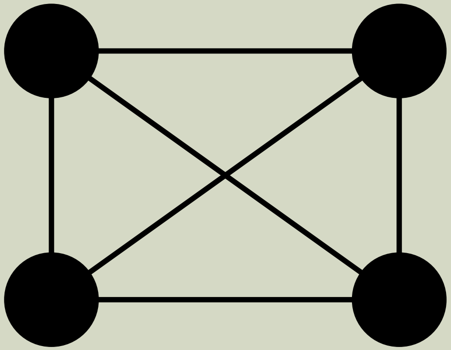
- Clique: Complete graph
- k-clique: Complete graph with k vertices



3-clique

CLIQUE

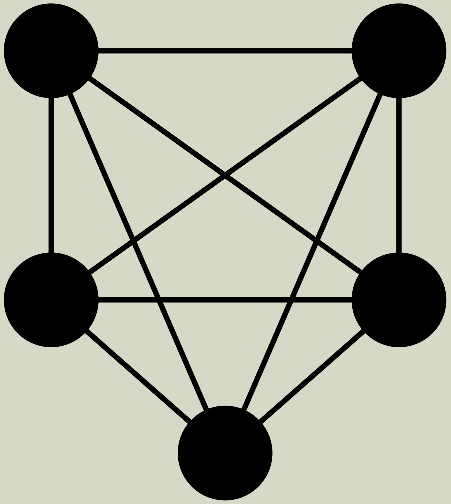
- Clique: Complete graph
- k-clique: Complete graph with k vertices



4-clique

CLIQUE

- Clique: Complete graph
- k-clique: Complete graph with k vertices



5-clique

K-CLIQUE COMMUNITIES

- **Adjacent k-cliques**

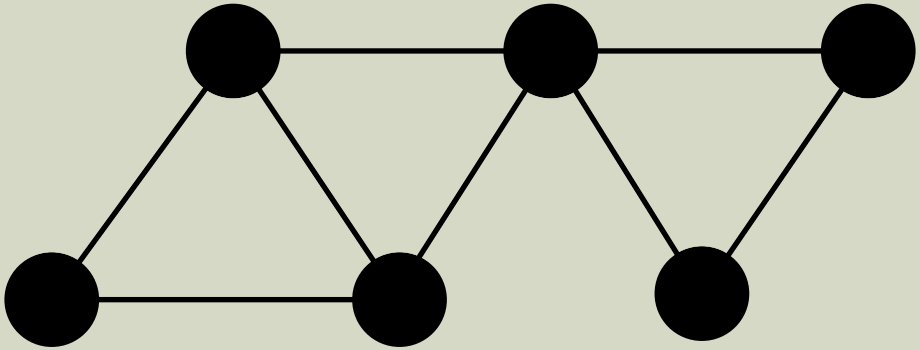
  Two k-cliques are adjacent when they share **k-1** nodes

K-CLIQUE COMMUNITIES

- **Adjacent k-cliques**

Two k-cliques are adjacent when they share **k-1** nodes

k = 3

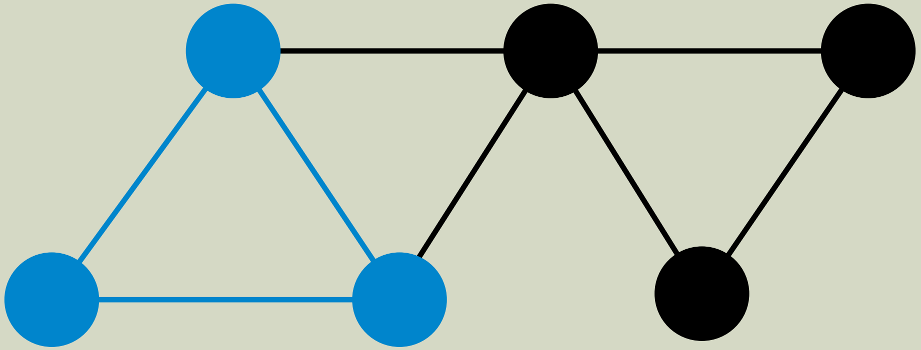


K-CLIQUE COMMUNITIES

- **Adjacent k-cliques**

  Two k-cliques are adjacent when they share **k-1** nodes

k = 3

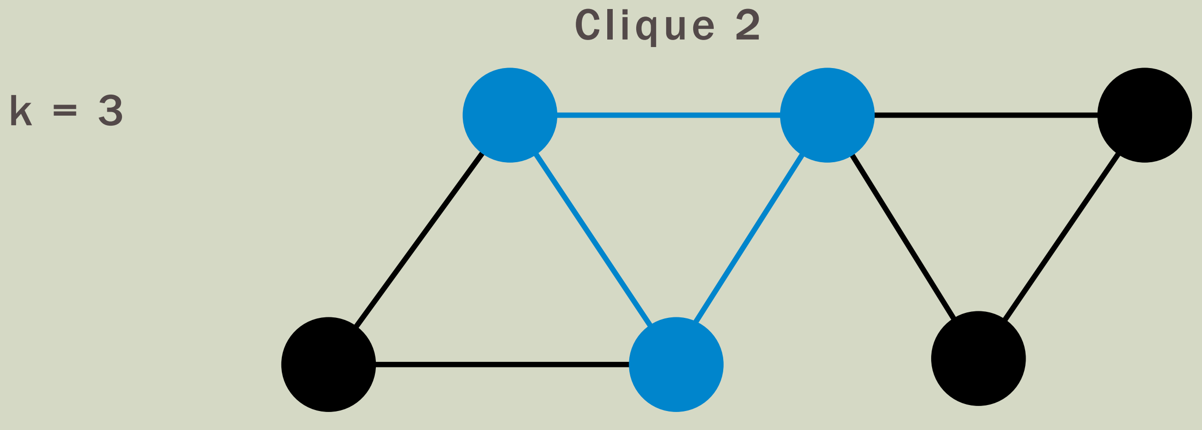


Clique 1

K-CLIQUE COMMUNITIES

- **Adjacent k-cliques**

  Two k-cliques are adjacent when they share **k-1** nodes

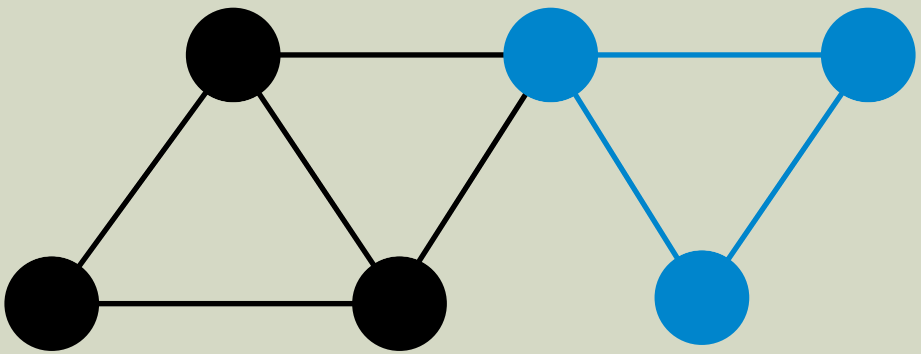


K-CLIQUE COMMUNITIES

- **Adjacent k-cliques**

  Two k-cliques are adjacent when they share **k-1** nodes

k = 3

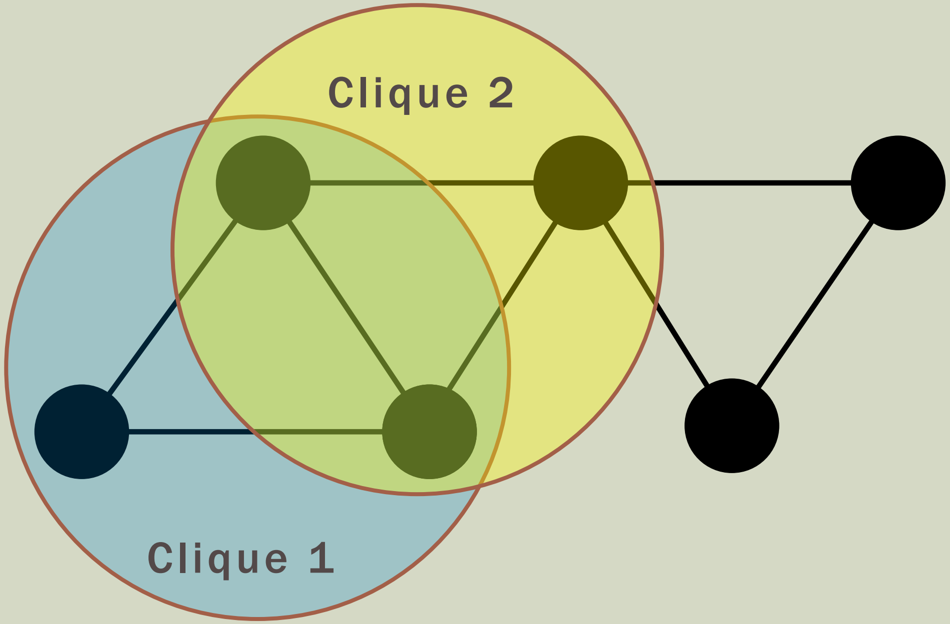


K-CLIQUE COMMUNITIES

- **Adjacent k-cliques**

  Two k-cliques are adjacent when they share **k-1** nodes

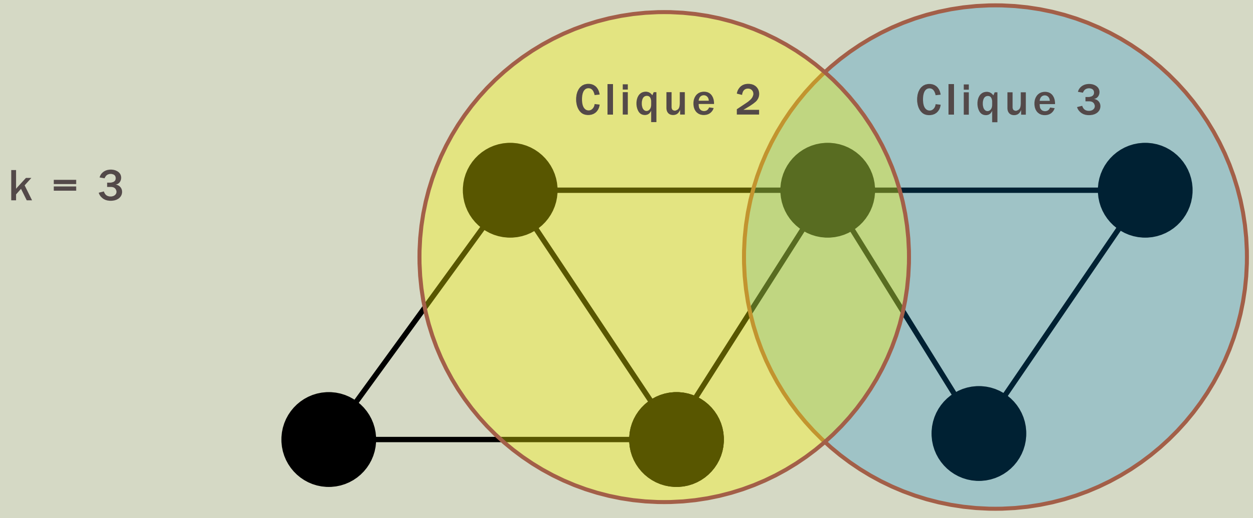
k = 3



K-CLIQUE COMMUNITIES

- **Adjacent k-cliques**

  Two k-cliques are adjacent when they share **k-1** nodes



K-CLIQUE COMMUNITIES

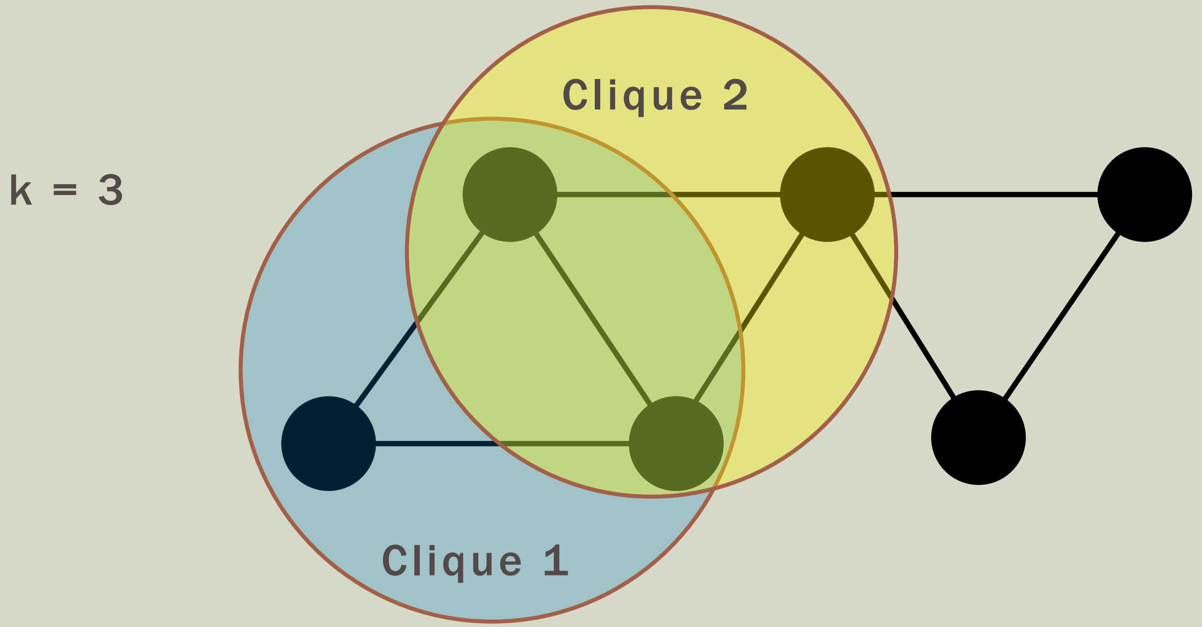
- **k-clique community**

  Union of all k-cliques that can be reached from each other through a series of adjacent k-cliques

K-CLIQUE COMMUNITIES

- **k-clique community**

  Union of all k-cliques that can be reached from each other through a series of adjacent k-cliques

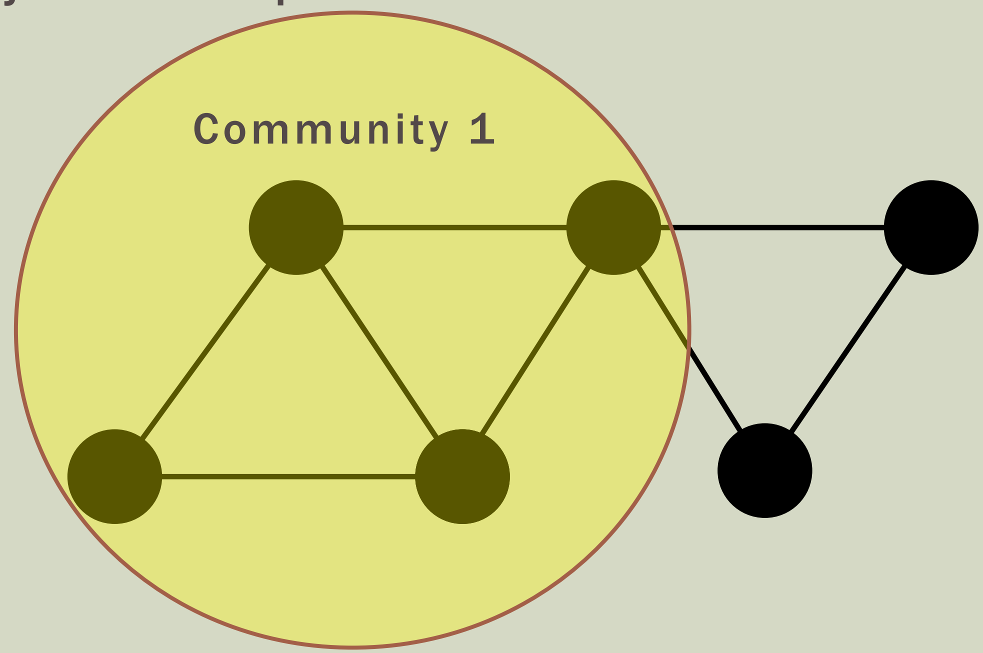


K-CLIQUE COMMUNITIES

- **k-clique community**

  Union of all k-cliques that can be reached from each other through a series of adjacent k-cliques

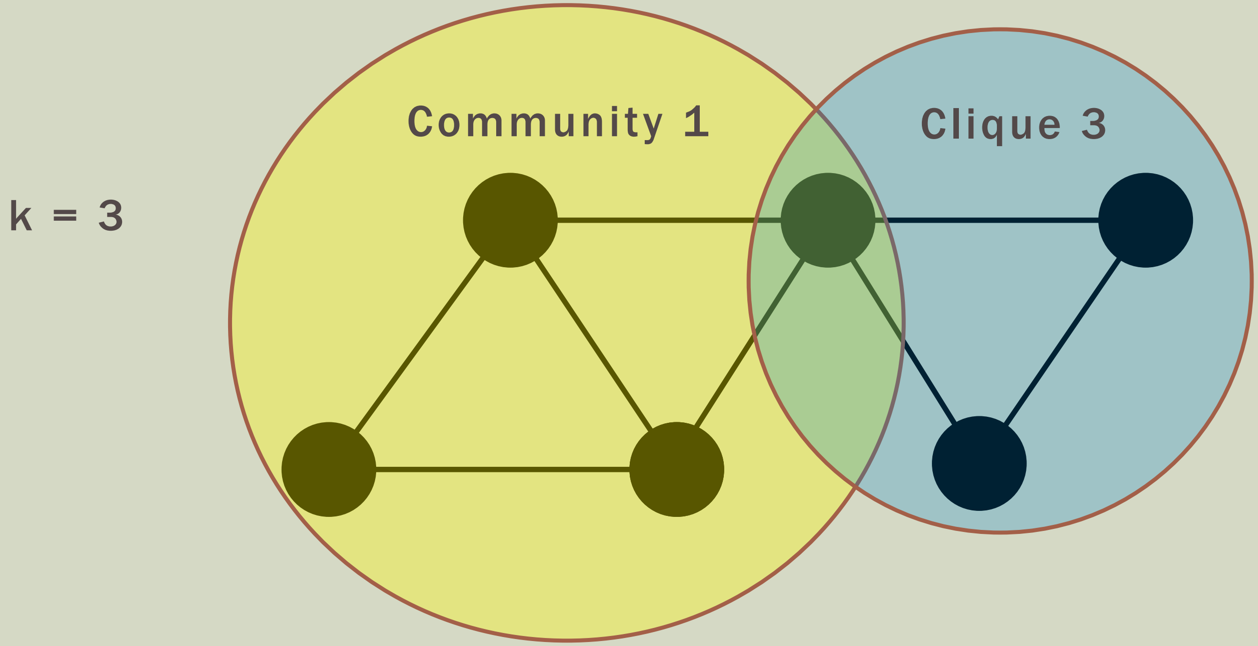
k = 3



K-CLIQUE COMMUNITIES

- **k-clique community**

  Union of all k-cliques that can be reached from each other through a series of adjacent k-cliques

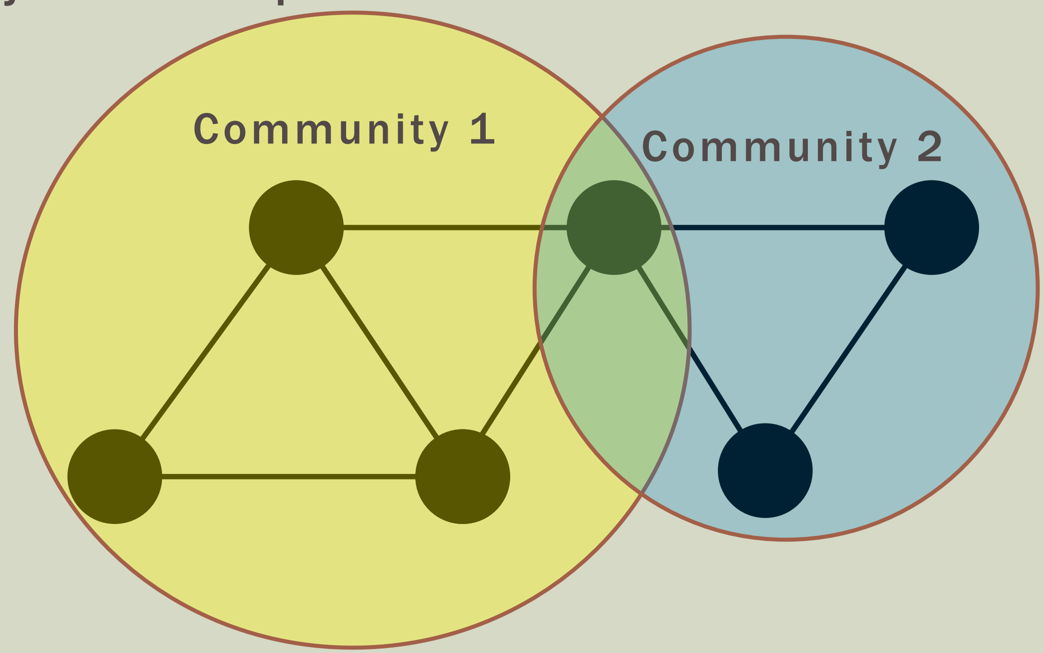


K-CLIQUE COMMUNITIES

- **k-clique community**

  Union of all k-cliques that can be reached from each other through a series of adjacent k-cliques

k = 3



ALGORITHM

- Locate maximal cliques
- Convert from cliques to k-clique communities

LOCATE MAXIMAL CLIQUES

- Largest possible clique size can be determined from degrees of vertices
- Starting from this size, find all cliques, then reduce size by 1 and repeat

LOCATE MAXIMAL CLIQUES

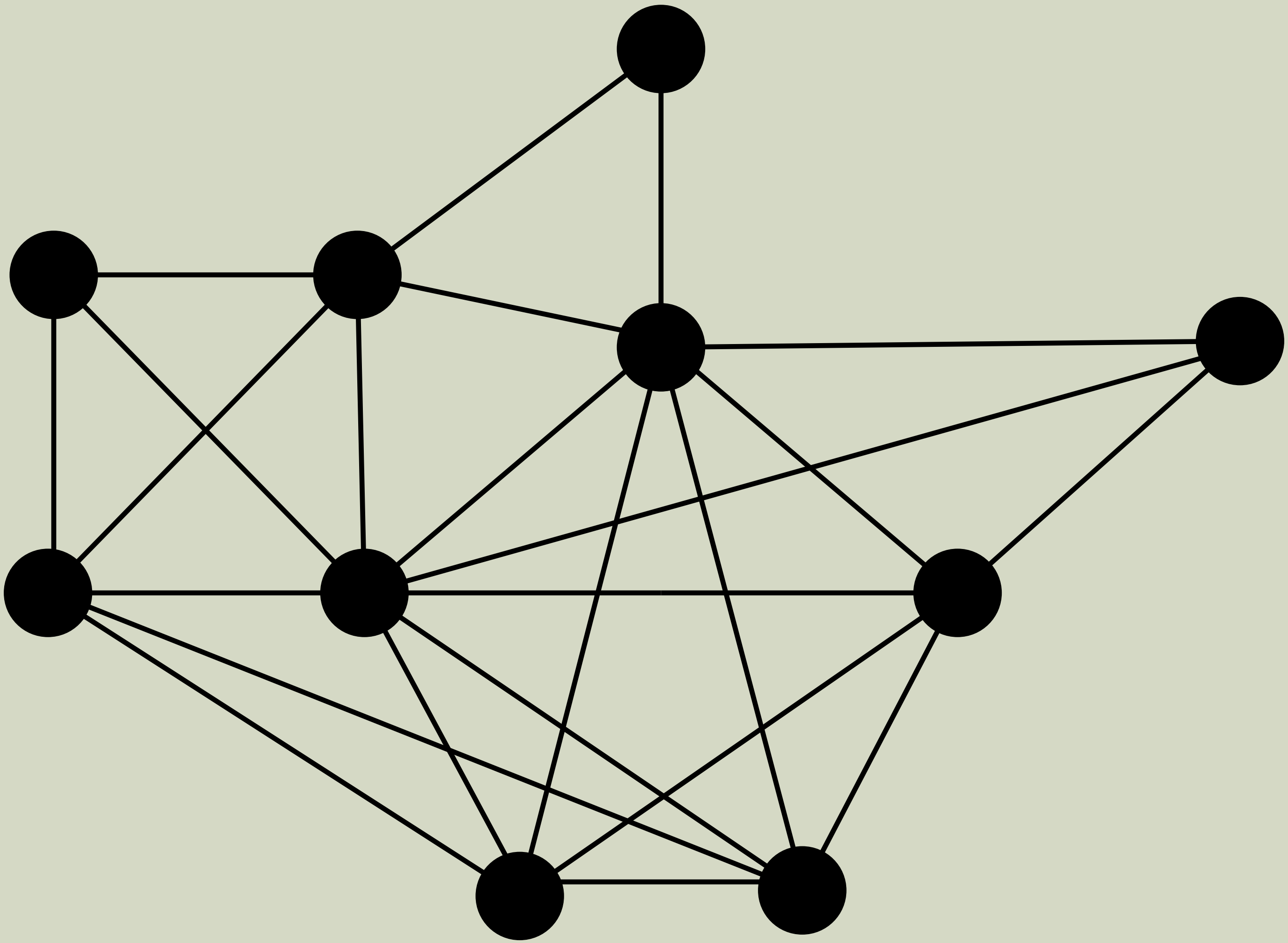
- Finding all cliques: brute-force

  1. Set A initially contains vertex v, Set B contains neighbours of v
  2. Transfer one vertex w from B to A
  3. Remove vertices that are not neighbours of w from B
  4. Repeat until A reaches desired size
  5. If fail, step back and try other possibilities

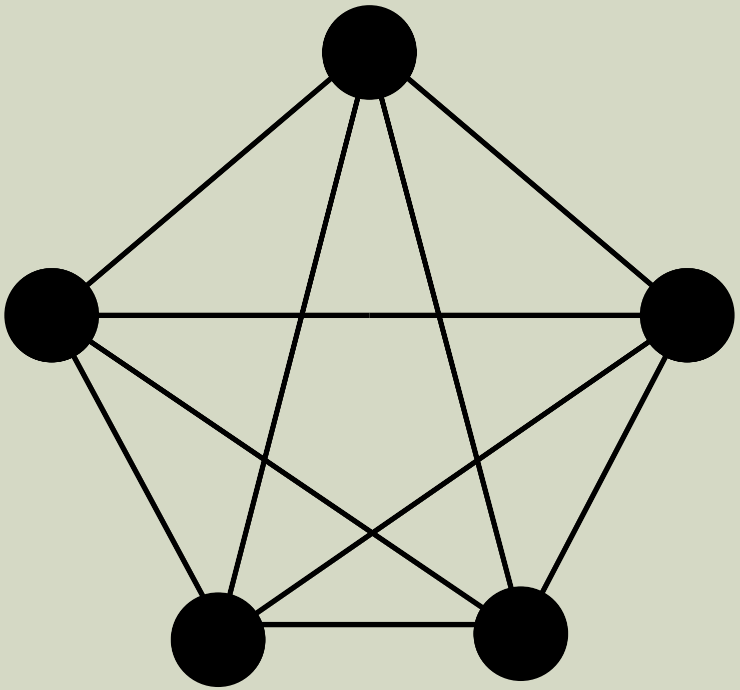
ALGORITHM

- Locate maximal cliques
- Convert from cliques to k-clique communities

CLIQUES TO K-CLIQUE COMMUNITIES

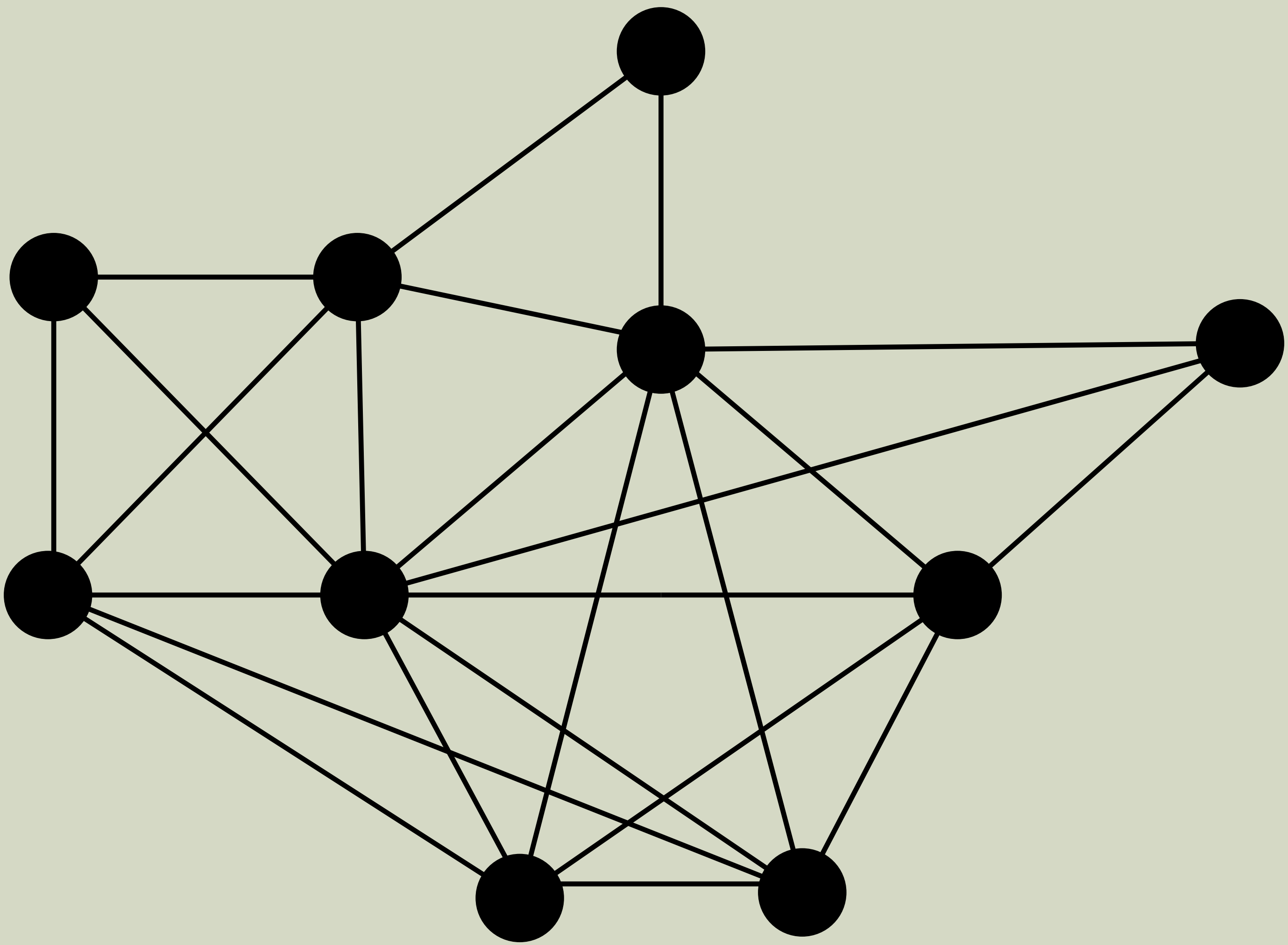


CLIQUES TO K-CLIQUE COMMUNITIES

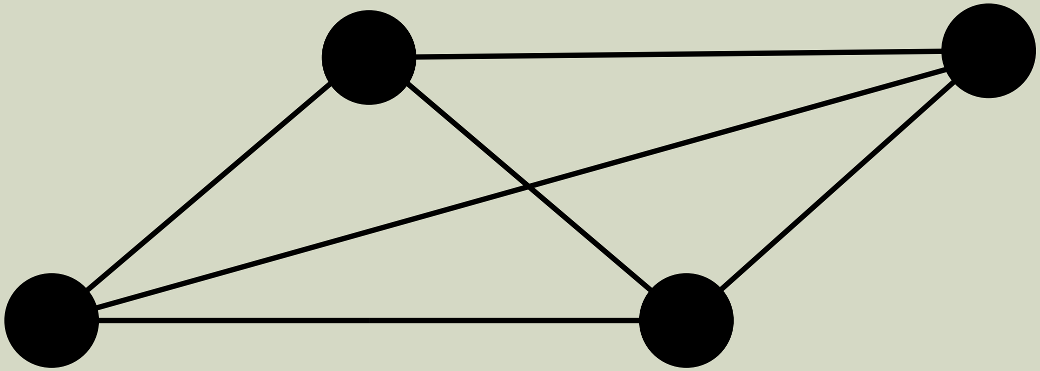


Clique 1: 5-clique

CLIQUES TO K-CLIQUE COMMUNITIES

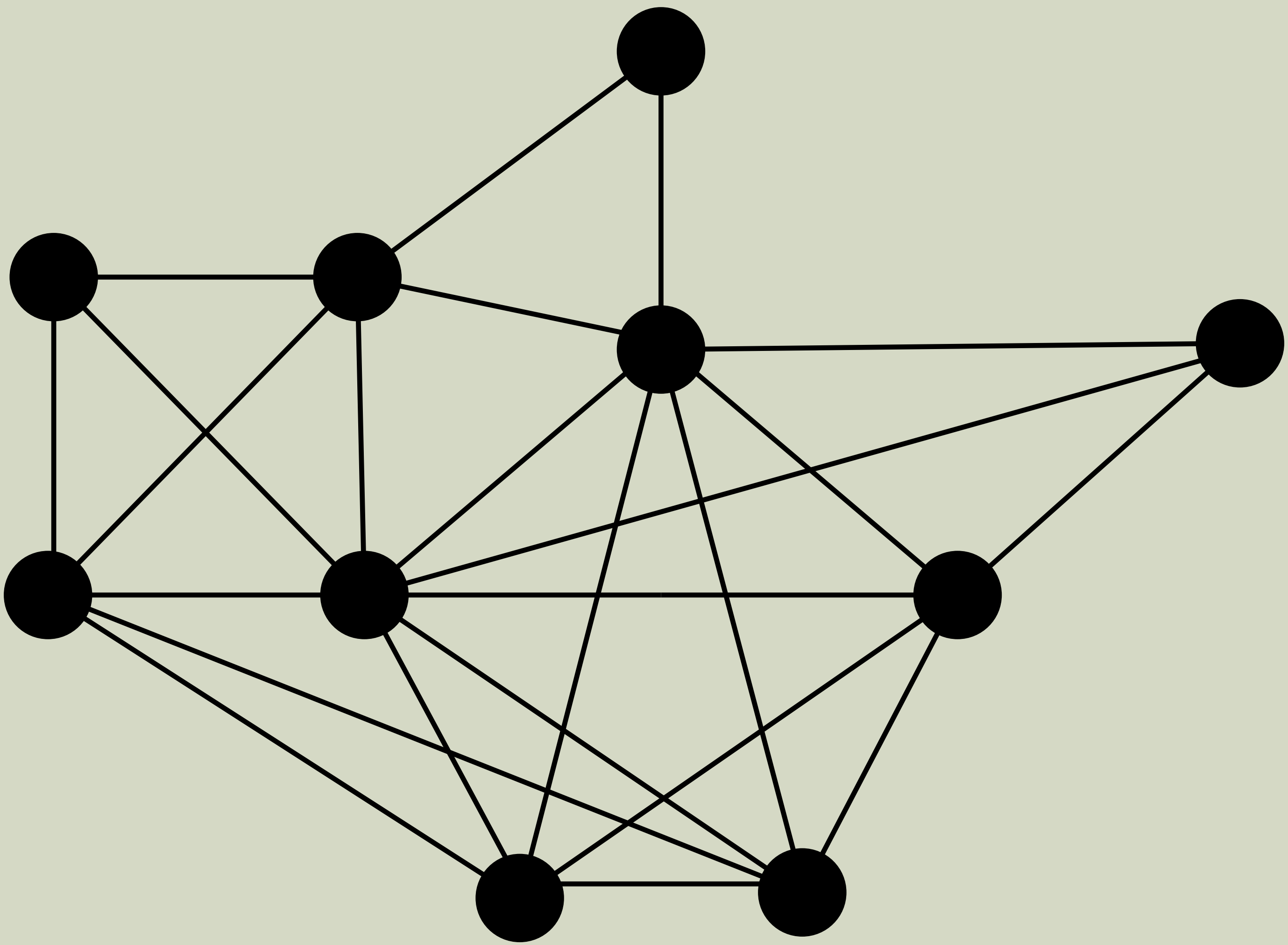


CLIQUES TO K-CLIQUE COMMUNITIES



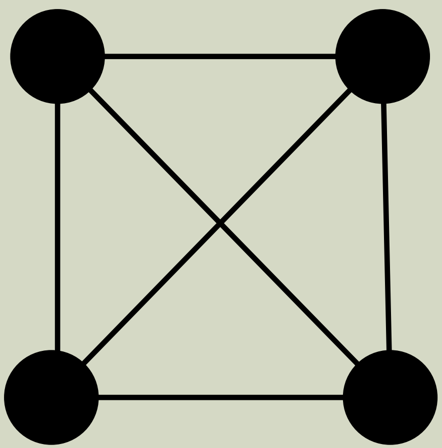
Clique 2: 4-clique

CLIQUES TO K-CLIQUE COMMUNITIES

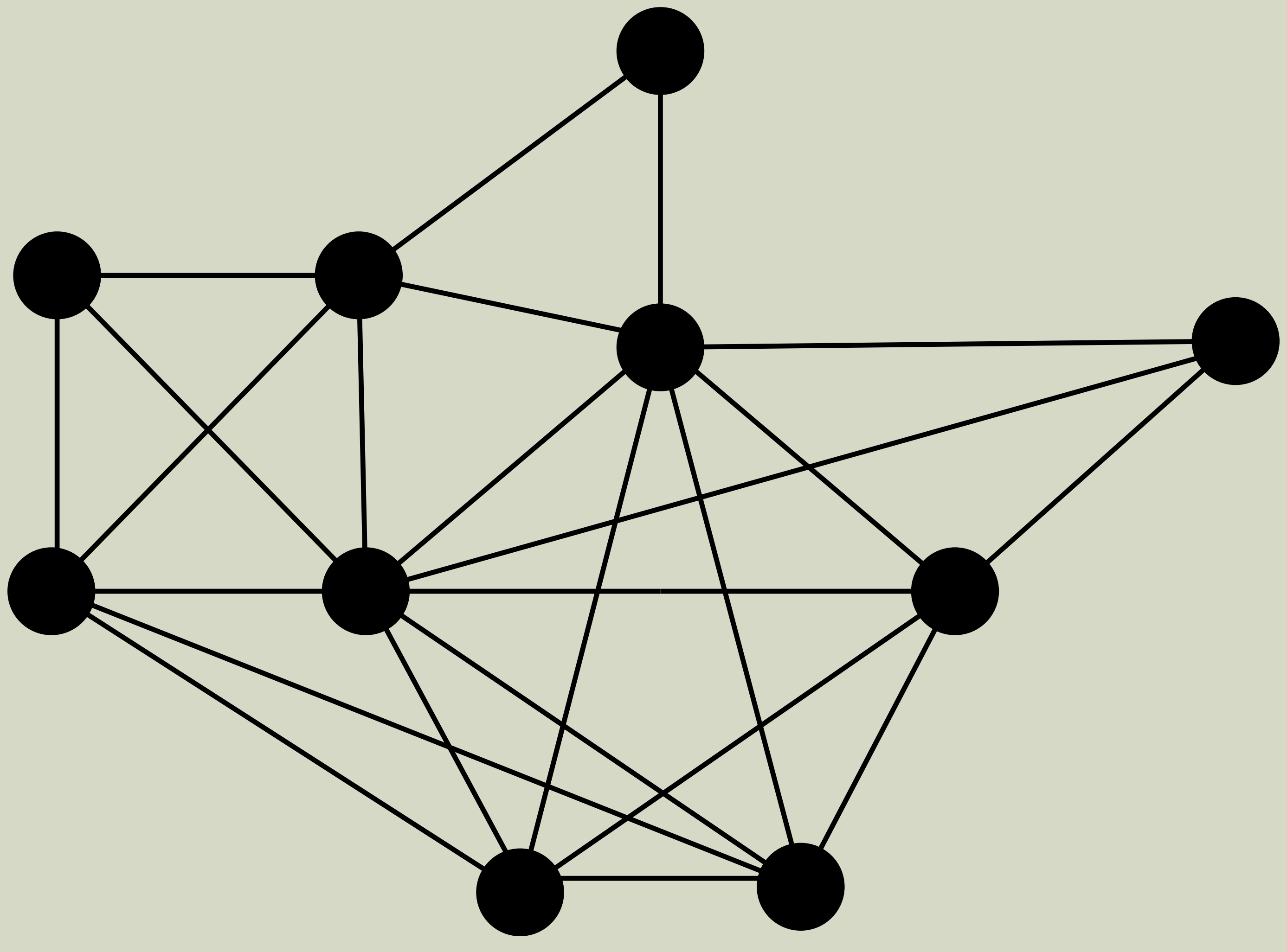


CLIQUES TO K-CLIQUE COMMUNITIES

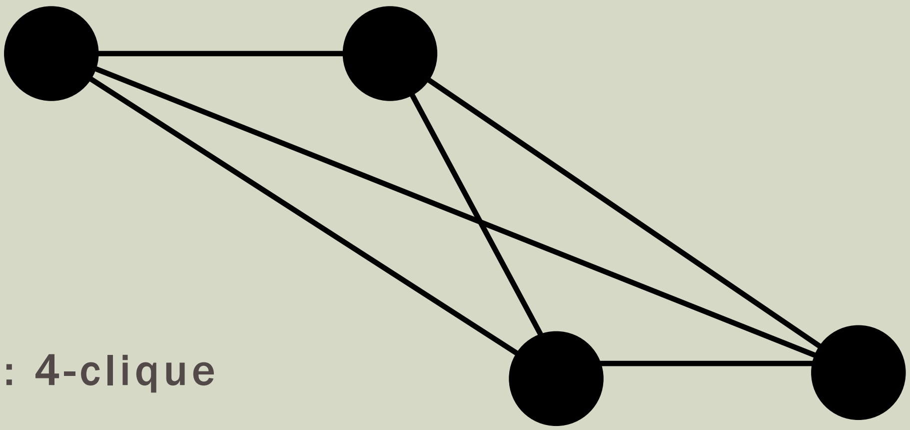
Clique 3: 4-clique



CLIQUES TO K-CLIQUE COMMUNITIES

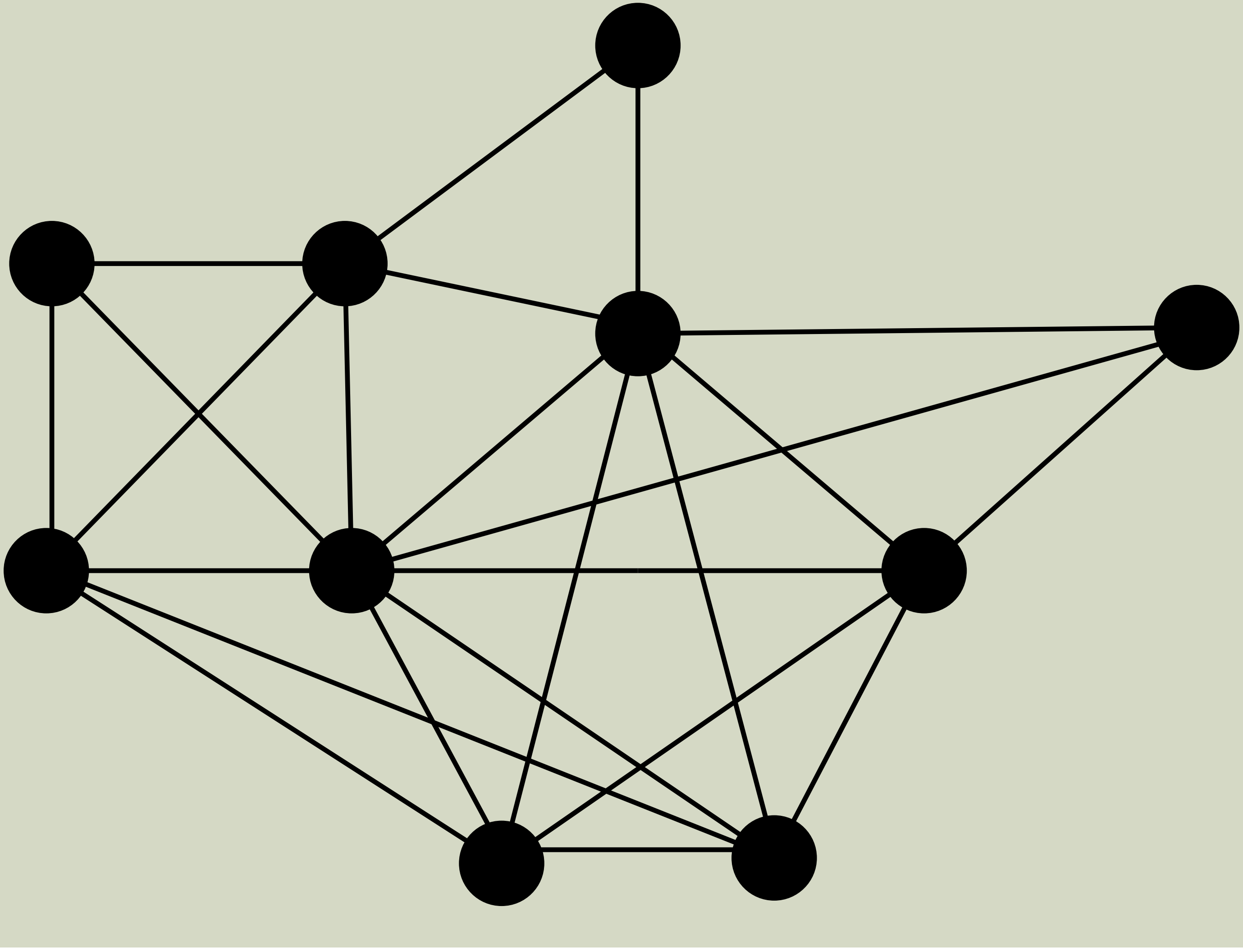


CLIQUES TO K-CLIQUE COMMUNITIES

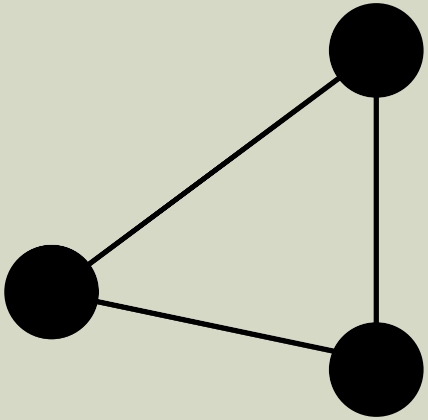


Clique 4: 4-clique

CLIQUES TO K-CLIQUE COMMUNITIES

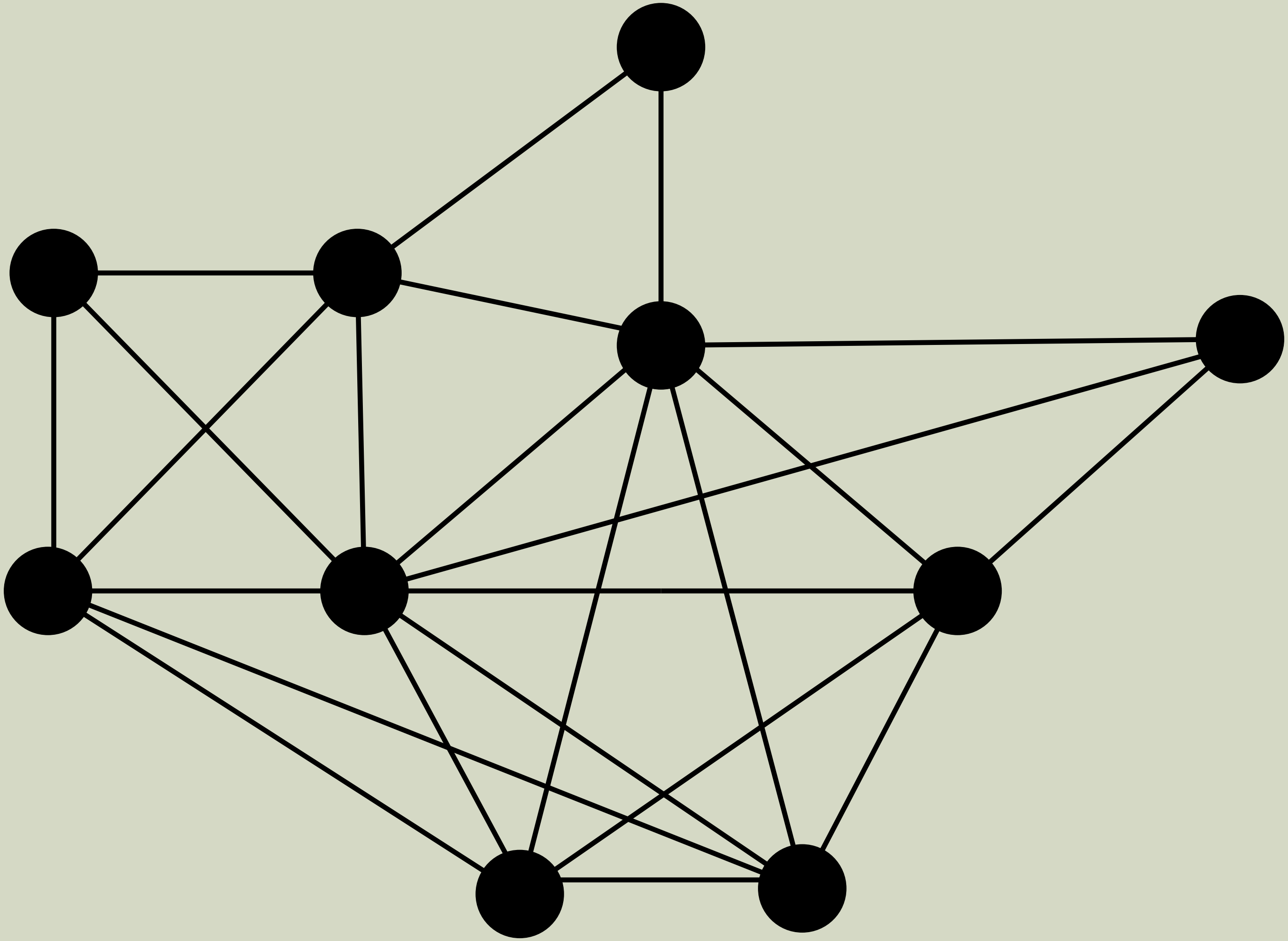


CLIQUES TO K-CLIQUE COMMUNITIES



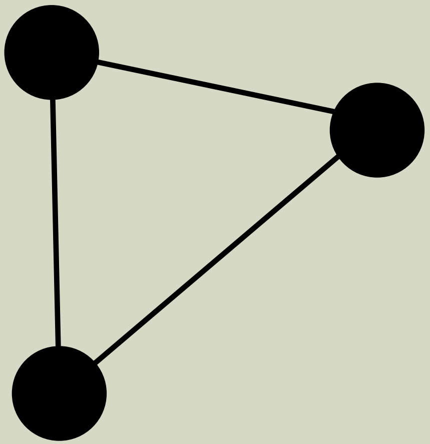
Clique 5: 3-clique

CLIQUES TO K-CLIQUE COMMUNITIES



CLIQUES TO K-CLIQUE COMMUNITIES

Clique 6: 3-clique



A fast algorithm to do form k-clique communities

CLIQUES TO K-CLIQUE COMMUNITIES

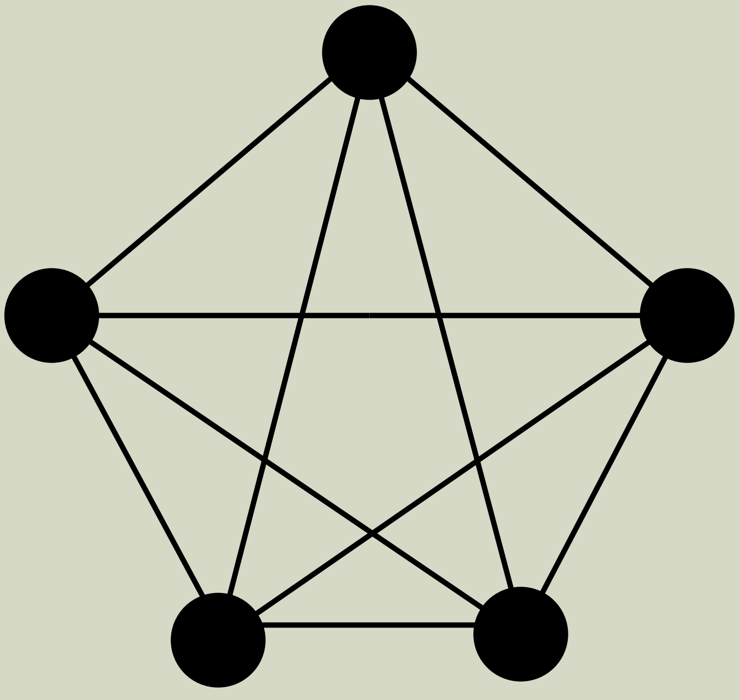
	1	2	3	4	5	6
1	5					
2		4				
3			4			
4				4		
5					3	
6						3

CLIQUES TO K-CLIQUE COMMUNITIES

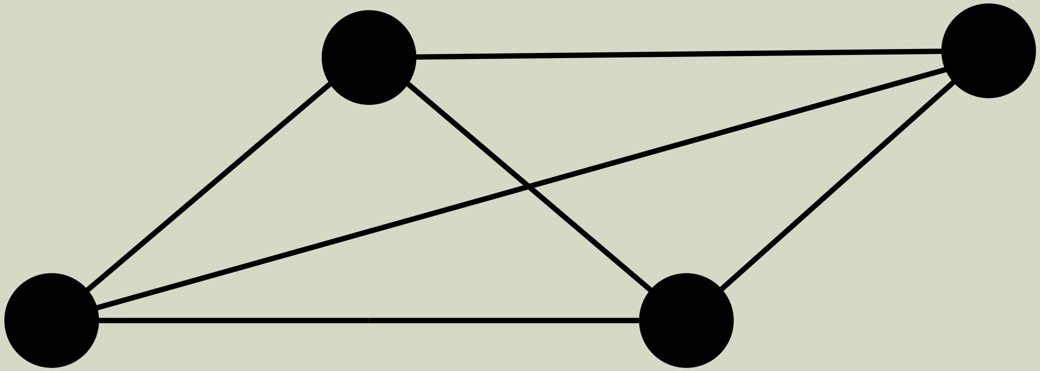
	1	2	3	4	5	6
1	5	3	1	3	1	2
2	3	4	1	1	1	2
3	1	1	4	2	1	2
4	3	1	2	4	0	1
5	1	1	1	0	3	2
6	2	2	2	1	2	3

CLIQUES TO K-CLIQUE COMMUNITIES

Clique 1: 5-clique



CLIQUES TO K-CLIQUE COMMUNITIES



Clique 2: 4-clique

CLIQUES TO K-CLIQUE COMMUNITIES

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

CLIQUES TO K-CLIQUE COMMUNITIES

k=4

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

CLIQUES TO K-CLIQUE COMMUNITIES

k=4

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

CLIQUES TO K-CLIQUE COMMUNITIES

k=4

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

Delete if less than k

CLIQUES TO K-CLIQUE COMMUNITIES

k=4

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

CLIQUES TO K-CLIQUE COMMUNITIES

k=4

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

CLIQUES TO K-CLIQUE COMMUNITIES

k=4

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

Delete if less than k-1

CLIQUES TO K-CLIQUE COMMUNITIES

k=4

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

CLIQUES TO K-CLIQUE COMMUNITIES

k=4

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

Change all non-zeros to 1

CLIQUES TO K-CLIQUE COMMUNITIES

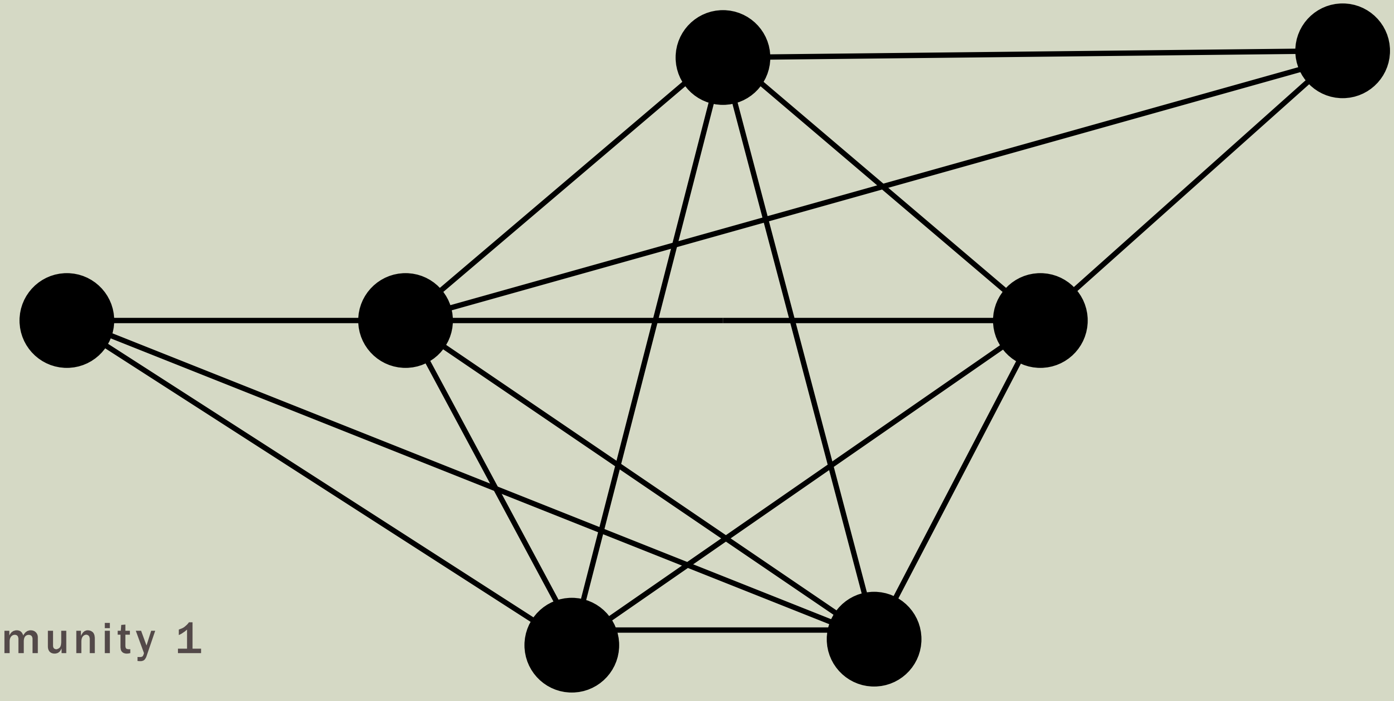
k=4

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

Clique-clique overlap matrix

CLIQUES TO K-CLIQUE COMMUNITIES

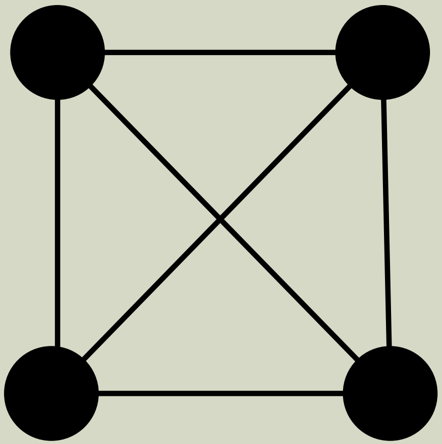
k=4



Community 1

CLIQUES TO K-CLIQUE COMMUNITIES

k=4



Community 2

ANALYSIS

- Believed to be non-polynomial
- No closed formula can be given
- However, claimed to be efficient on real systems

CONCLUSION

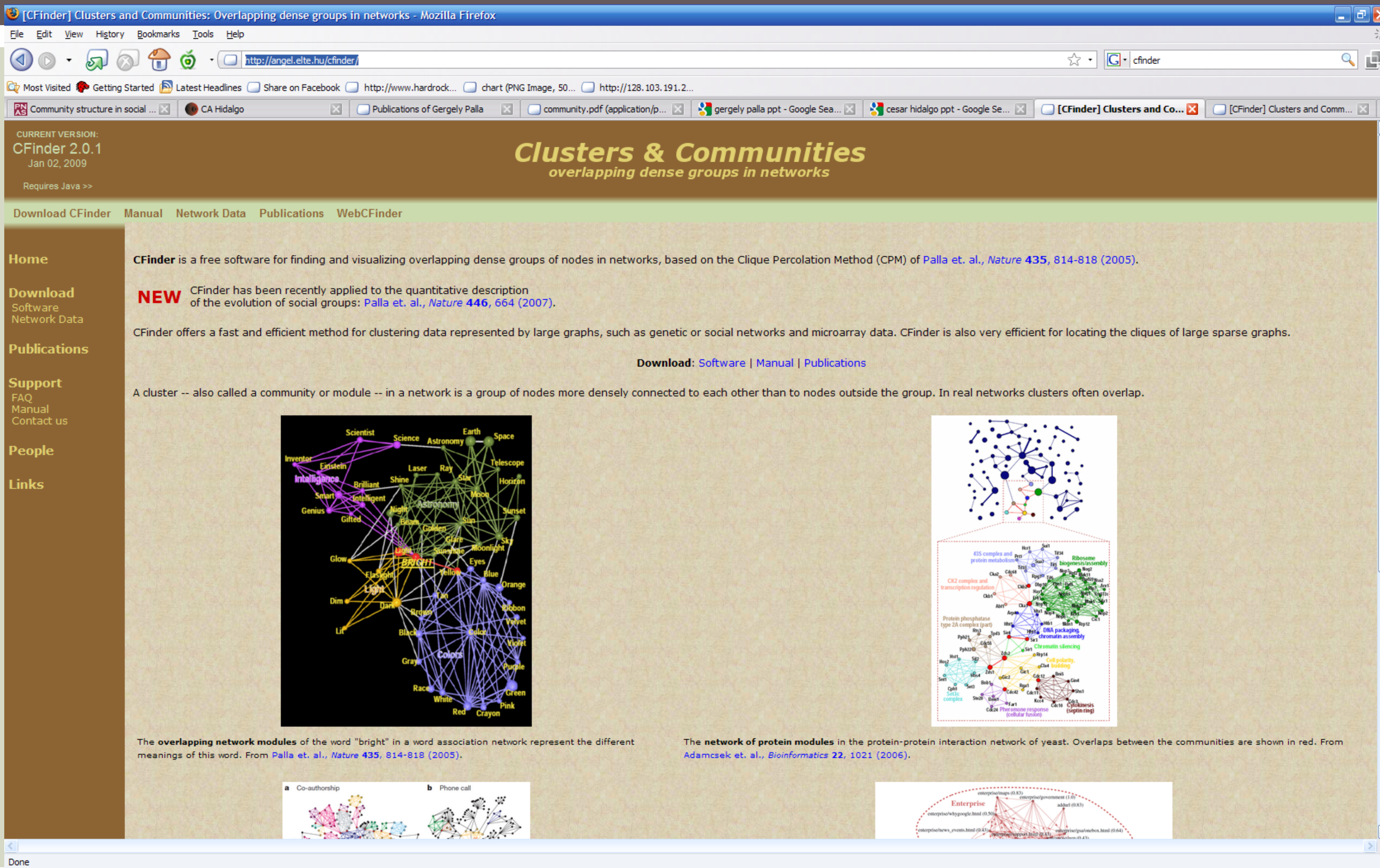
- Widely used algorithm for detecting overlapping communities
- However:
  - Fail to give meaningful covers for graph with few cliques
  - With too many cliques, might give a trivial community structure
  - Left out vertices?
  - Subgraphs containing many cliques == community?
  - What value of k to choose to give a meaningful structure?

REFERENCES

- Palla et al. – Uncovering the overlapping community structure of complex networks in nature and society
- Santo Fortunato - Community detection in graphs

Wanna use Clique Percolation Method?

Just google: “cfinder”



Thank you!