Distributed Generation of Random Graphs Based on Social Network Models

Institute for System Programming of the Russian Academy of Sciences

Kyrylo Chykhradze et. al
chykhradze@ispras.ru

GraphHPC-2015

March 5th, 2015
Moscow, Russia

ISP RAS

Outline

- About Spark
- Task definition
- RGG: graphs without a community structure
- CKB: graphs with a community structure
- Testing
- Conclusions

ISP RAS

Outline

- About Spark
- Task definition
- RGG: graphs without a community structure
- CKB: graphs with a community structure
- Testing
- Conclusions

What is Spark?

- Independent fast platform for distributed computing that supports the data processing by MapReduce model, Pregel and Graphx
  - ✓ Storing data in memory for fast processing of interactive inquiries
  - ✓ Can be 100 times faster than Hadoop
- Compatible with the Hadoop storage system (HDFS, Hbase, SequenceFiles etc)

ISP RAS

Spark programming model

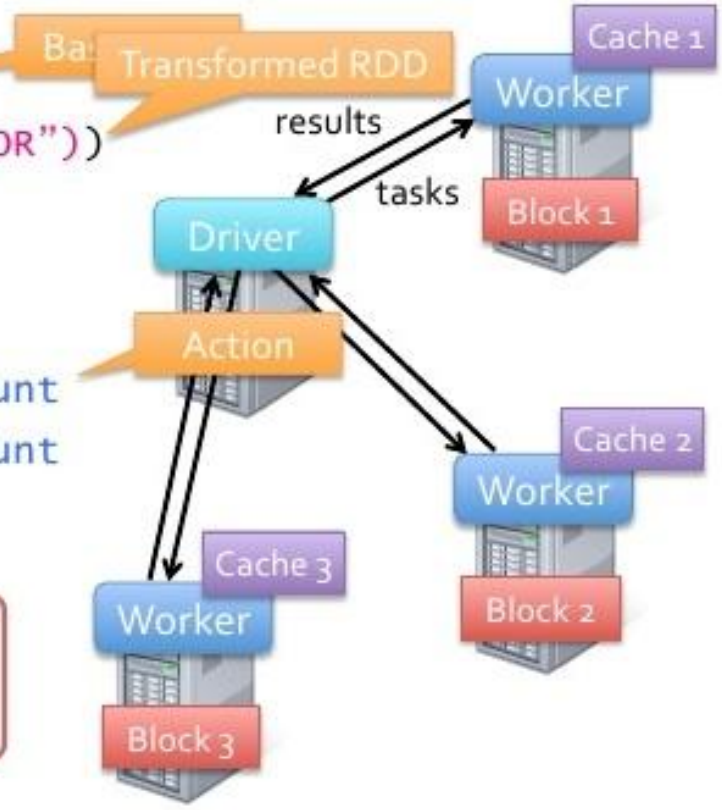
- The main idea: resilient distributed datasets (RDDs)
  - ✓ Distributed collection of objects that can be cached in the cluster nodes memory
  - ✓ One can manipulate using various parallel operations (such as map and reduce)
  - ✓ Automatically rebuild in case of failures
- Interface:
  - ✓ Elegant interface integrated into the Scala language
  - ✓ Can be used interactively from the Scala console

Example: logs analysis

1. Upload an error message in the memory
2. Interactive execute queries to them

```
lines = spark.textFile("hdfs://...")
errors = lines.filter(_.startsWith("ERROR"))
messages = errors.map(_.split('\t')(2))
cachedMsgs = messages.cache()

cachedMsgs.filter(_.contains("foo")).count
cachedMsgs.filter(_.contains("bar")).count
. . .
```



Result: scaled to 1 TB data in 5-7 sec
(vs 170 sec for on-disk data)

ISP RAS

Outline

- About Spark
- Task definition
- RGG: graphs without a community structure
- CKB: graphs with a community structure
- Testing
- Conclusions

Task definition

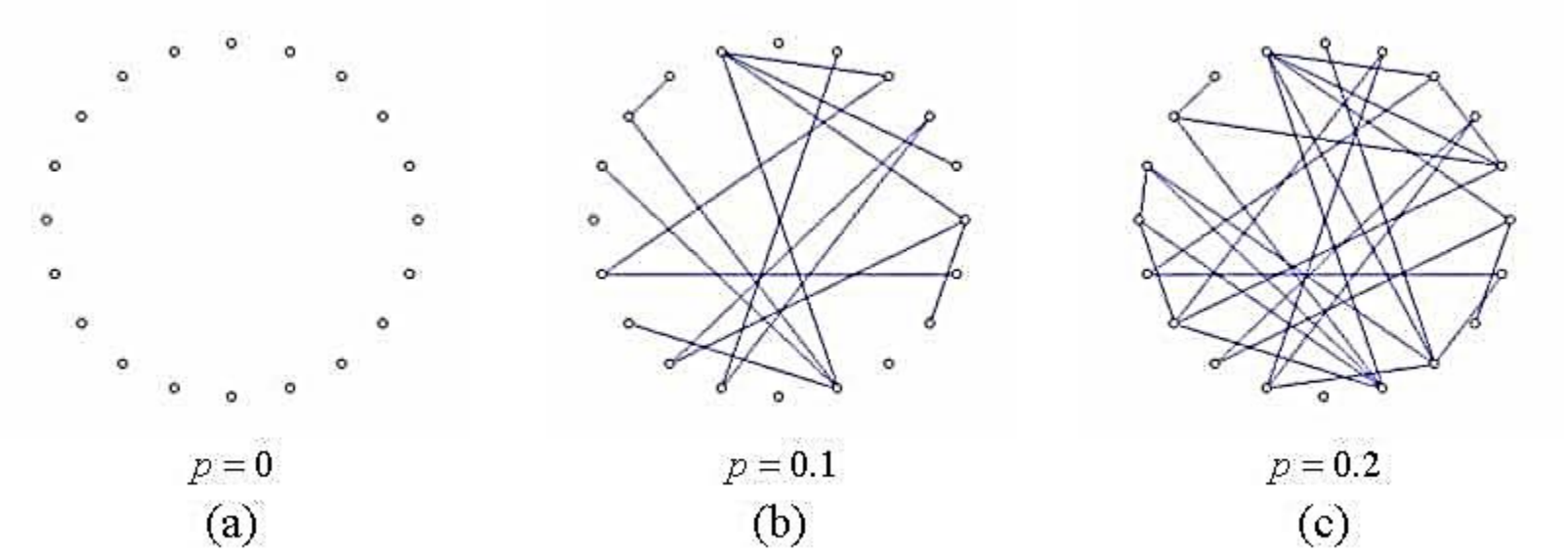
To generate random graph…

- …which will satisfy the basic properties of social networks;
- … in a reasonable time (even a billion vertices);

What is a random graph?

Erdős–Rényi graph

- N nodes
- Edge appears with a probability p



$p = 0$ (a)

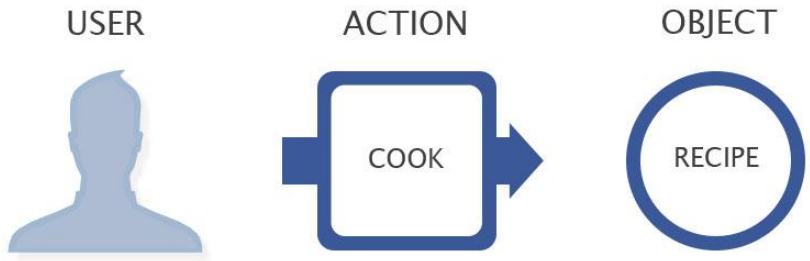
$p = 0.1$ (b)

$p = 0.2$ (c)

What is a social graph?

Type of nodes

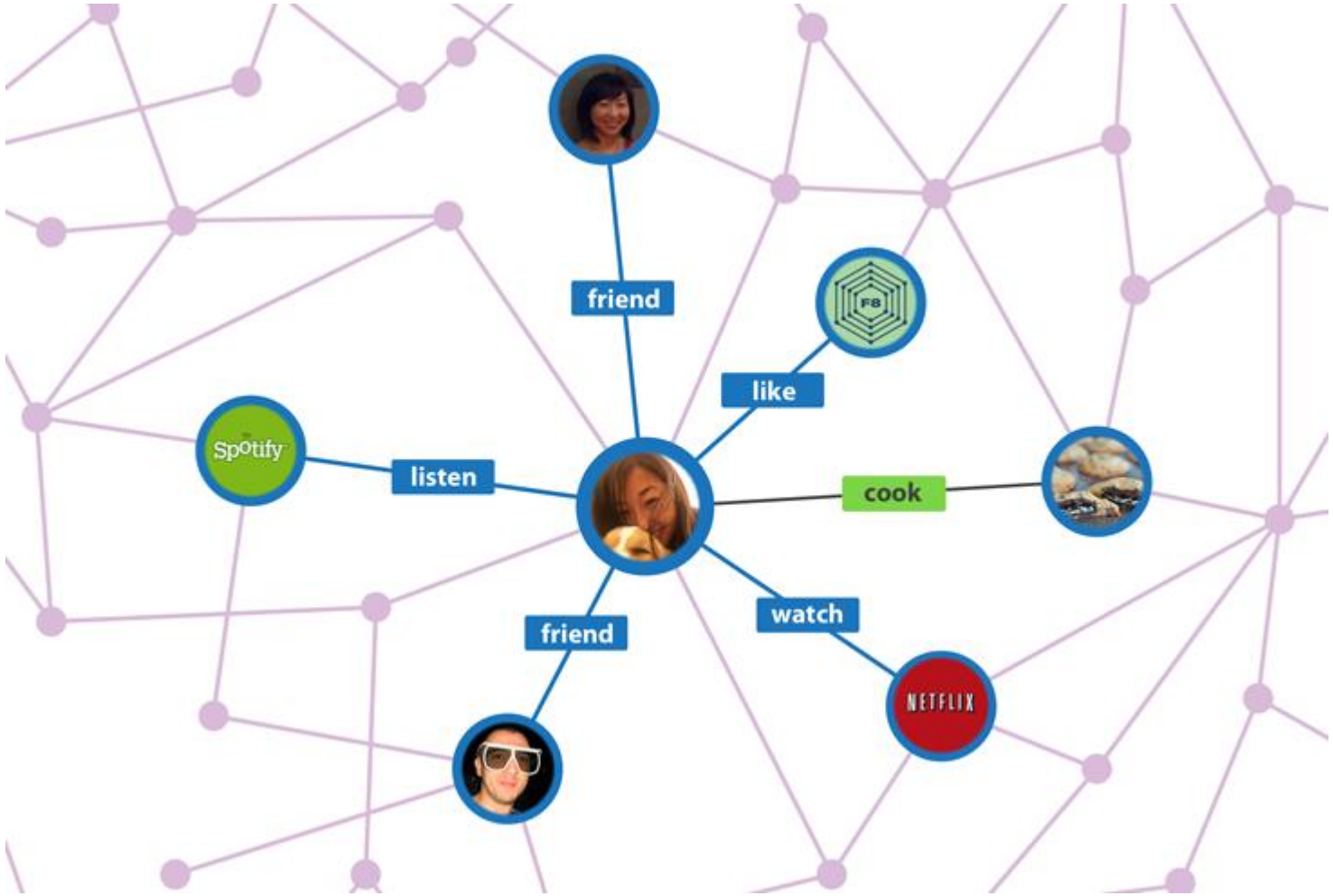
- Users: profiles field: attributes, interests, contacts
- Communities: lists, groups
- *Content*: messages, pictures, videos



Type of edges

- Social ties: friends, followers
- *Interacting with a content*: «likes», reposts, comments

What is a social graph?



ISP RAS

Social network properties

- Node degree distribution is a power law:

$$P(x) \propto x^{-\beta}$$

- Small effective diameter:

$$D \approx \frac{\ln(N)}{\ln(\ln(N))}$$

- Users are clustered in the overlapping communities

ISP RAS

Motivation

- The dimensions of modern social networks reach hundreds of millions vertices.
- It is required network analysis algorithms, whose effectiveness is proven on large graphs.
- Collecting real data is hindered due to the large time and resource costs.

ISP RAS

Outline

- About Spark
- Task definition
- RGG: graphs without a community structure
- CKB: graphs with a community structure
- Testing
- Conclusions

RGG: graphs without a community structure

- Node degree distribution is a power law:

$$P(x) \propto x^{-\beta}$$

- Small effective diameter:

$$D \approx \frac{\ln(N)}{\ln(\ln(N))}$$

- Users are clustered in the overlapping communities

RGG: input

- ***N*** – number of nodes
- ***d*** – mean degree
- **β** – degree distribution power law exponent

RGG: main steps

- Natural numbers (node degrees) are generated by power law distribution
- Computing the number of edges to generate
- Choosing the pair of numbers (edges) (i, j) proportional to their degrees

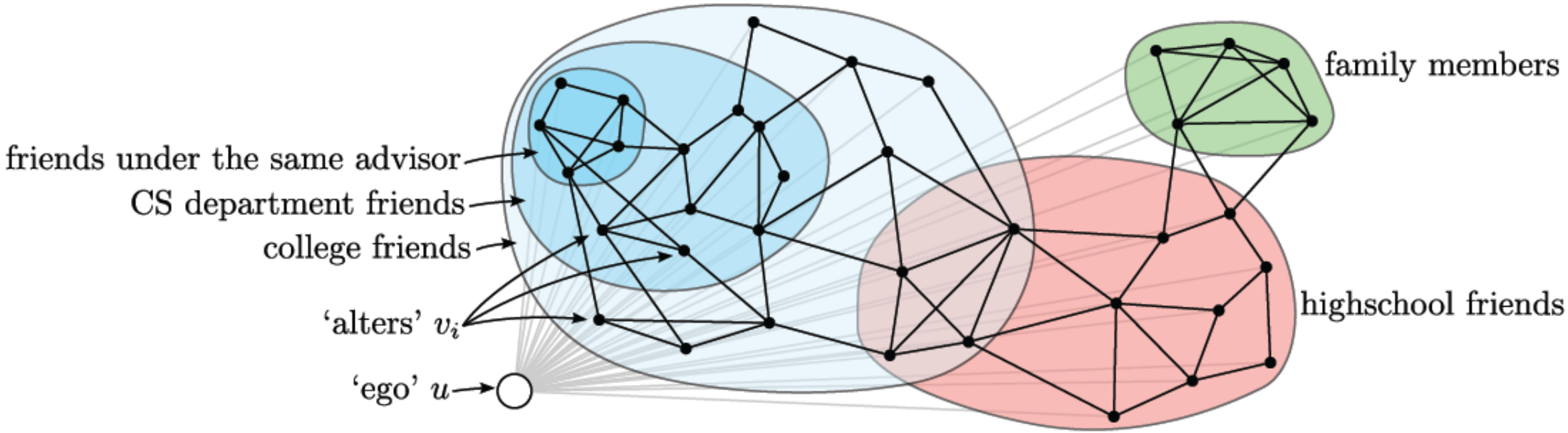
RGG: extensions

- Directed version
- Bigraph generation
- Attributes, texts, «likes»
- Communities

ISP RAS

What is a community?

What is a community?



ISP RAS

Outline

- About Spark
- Task definition
- RGG: graphs without a community structure
- CKB: graphs with a community structure
- Testing
- Conclusions

CKB: graphs with a community structure

- Node degree distribution is a power law:

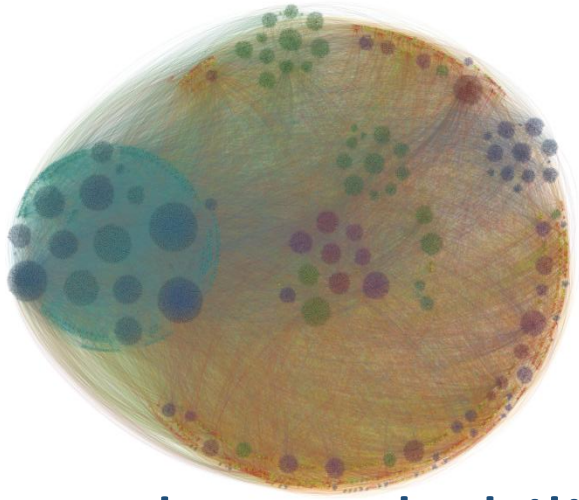
$$P(x) \propto x^{-\beta}$$

- Small effective diameter:

$$D \approx \frac{\ln(N)}{\ln(\ln(N))}$$

- Users are clustered in the overlapping communities

CKB: input



- N_1 – number of nodes
- d – mean degree
- Power law distribution parameters: max, min and exponent values
- α, γ – two constants that determine the edge probability
- ε – the edge probability between users regardless the community structure

CKB

Main steps:

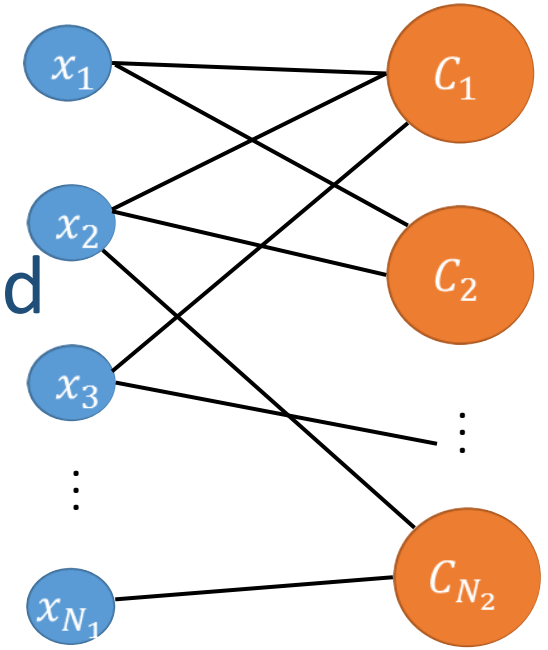
1. Bigraph node-community generation
2. Edges in communities are generated
3. Edges between users regardless the community structure are generated

CKB: node-community

1. Number of communities is computed from*

$$N_1 \cdot E[X_1] = N_2 \cdot E[X_2]$$

2. Memberships and community sizes are generated according to a power law with β_1 and β_2 exponents
3. Graph realization of these 2 degree sequences is created by random pairwise combinations of vertices from different parts



* $E[X_1]$ and $E[X_2]$ is the average values of membership and community size respectively

CKB: edges generation

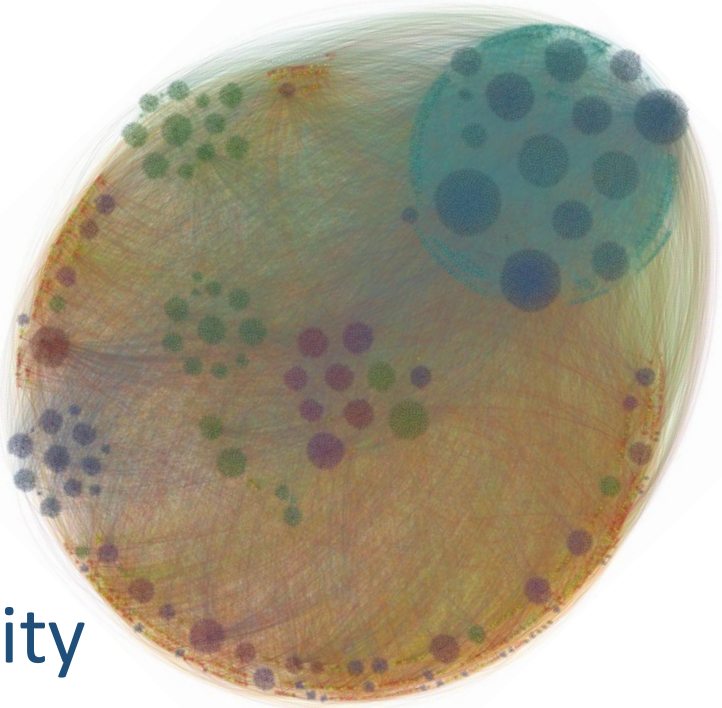
1. Edges in community are generated with probability*:

$$p = \frac{\alpha}{x_i^{\gamma}},$$

where x_i – size of i-th community

2. Edges between users regardless the community structure are generated with a probability:

$$p_{out} = \varepsilon$$



* Yang, J., and Leskovec, J. Structure and overlaps of communities in networks.
**Yang, J., and Leskovec, J. Community-affiliation graph model for overlapping network community detection.

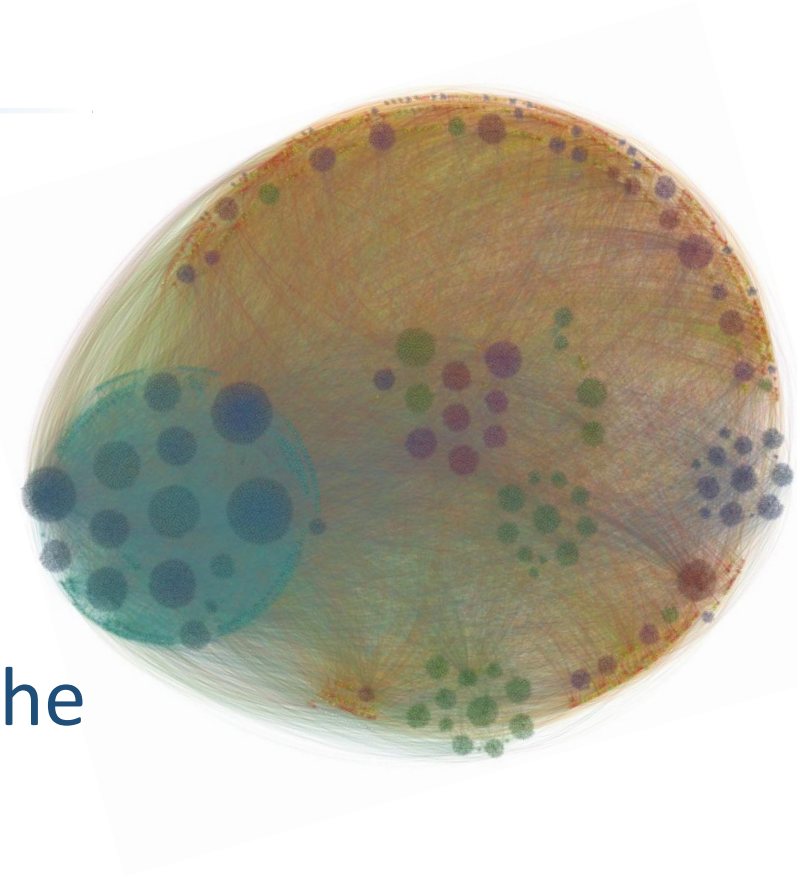
CKB: edges generation

1. Number of edges in a community is

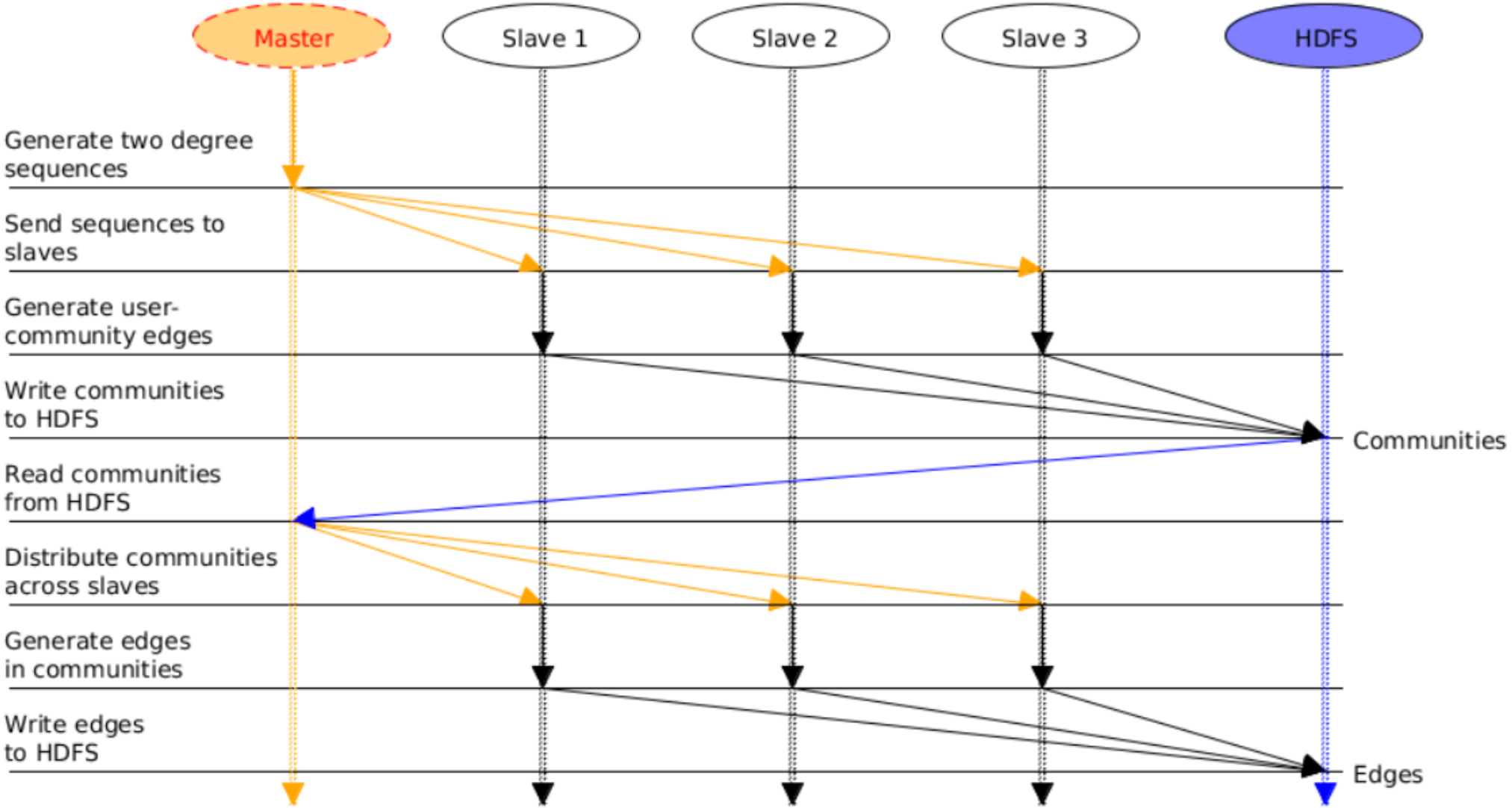
$$M_i \sim Bin\left(\frac{x_i(x_i-1)}{2}, \frac{\alpha}{x_i^{\gamma}}\right)$$

2. Number of edges between users regardless the community structure is:

$$M_i \sim Bin\left(\frac{N_1(N_1-1)}{2}, \varepsilon\right)$$



CKB: Apache Spark realization



ISP RAS

Outline

- About Spark
- Task definition
- RGG: graphs without a community structure
- CKB: graph with a community structure
- Testing
- Conclusions

Comparing with a real data

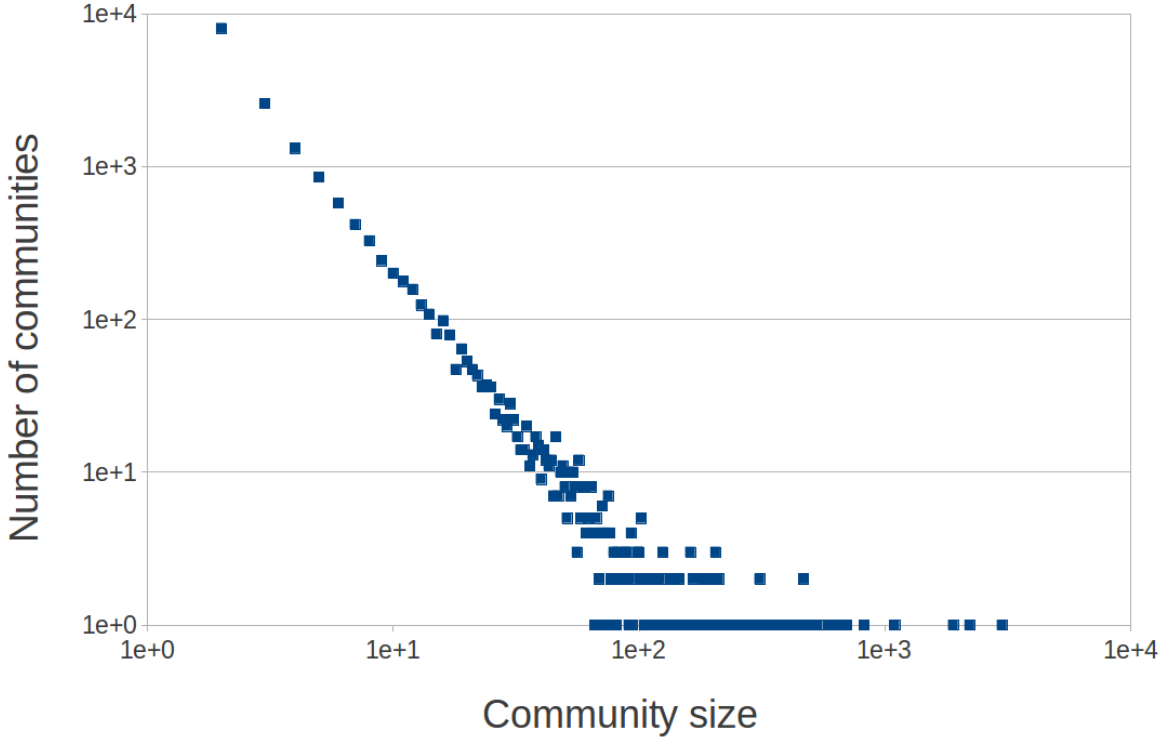
	LiveJournal	CKB
Number of nodes	$\approx 4 \cdot 10^6$	$\approx 4.2 \cdot 10^6$
Number of edges	$\approx 34.6 \cdot 10^6$	$\approx 38.2 \cdot 10^6$
Degree distribution exponent	2.14	2.15
Community size distribution exponent	2.22	2.26
Membership distribution exponent	2.15	2.15
Median of community size distribution	10	8
Median of membership distribution	2	2
Percentage of nodes with membership more than 1	63%	66%
Average clustering coefficient	0.3538	0.1034
Effective diameter	6.4	5.16

Comparing with a real data

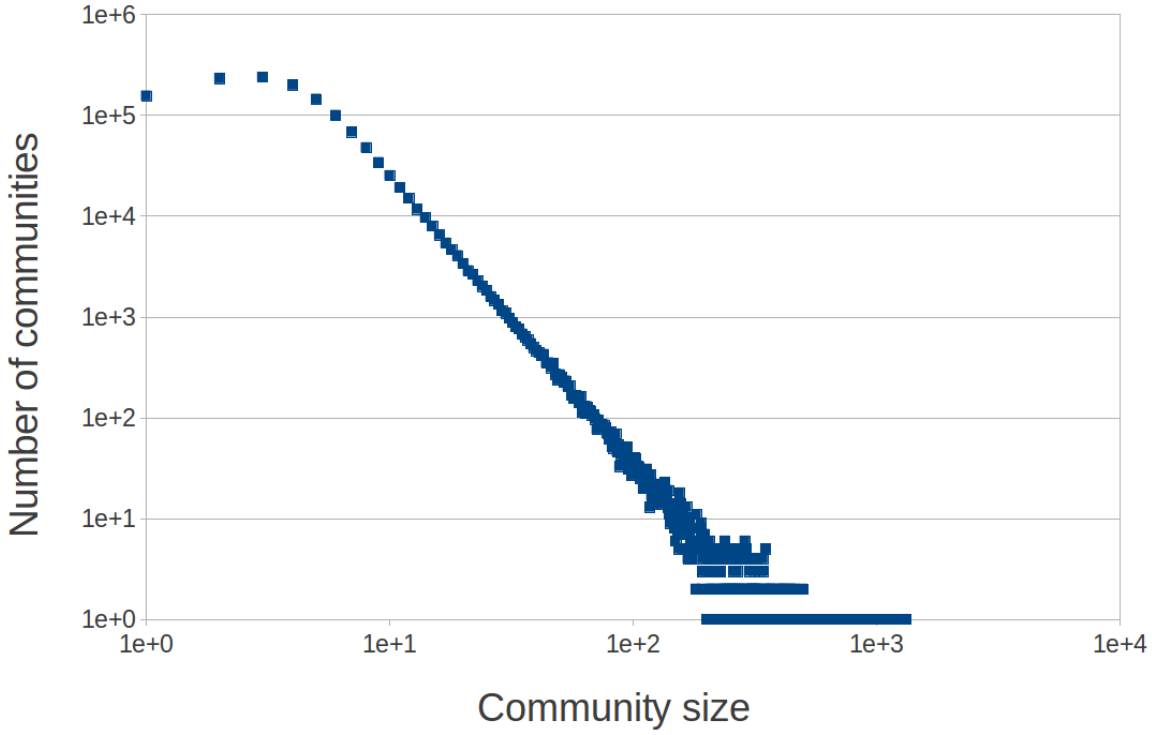
	YouTube	CKB
Number of nodes	$\approx 1.1 \cdot 10^6$	$\approx 1.1 \cdot 10^6$
Number of edges	$\approx 3 \cdot 10^6$	$\approx 3 \cdot 10^6$
Degree distribution exponent	2.36	2.41
Community size distribution exponent	2.83	2.95
Membership distribution exponent	2.53	2.45
Median of community size distribution	3	4
Median of membership distribution	2	2
Percentage of nodes with membership more than 1	38%	68%
Average clustering coefficient	0.1723	0.1066
Effective diameter	6.5	6.2

Comparing YouTube and CKB

YouTube

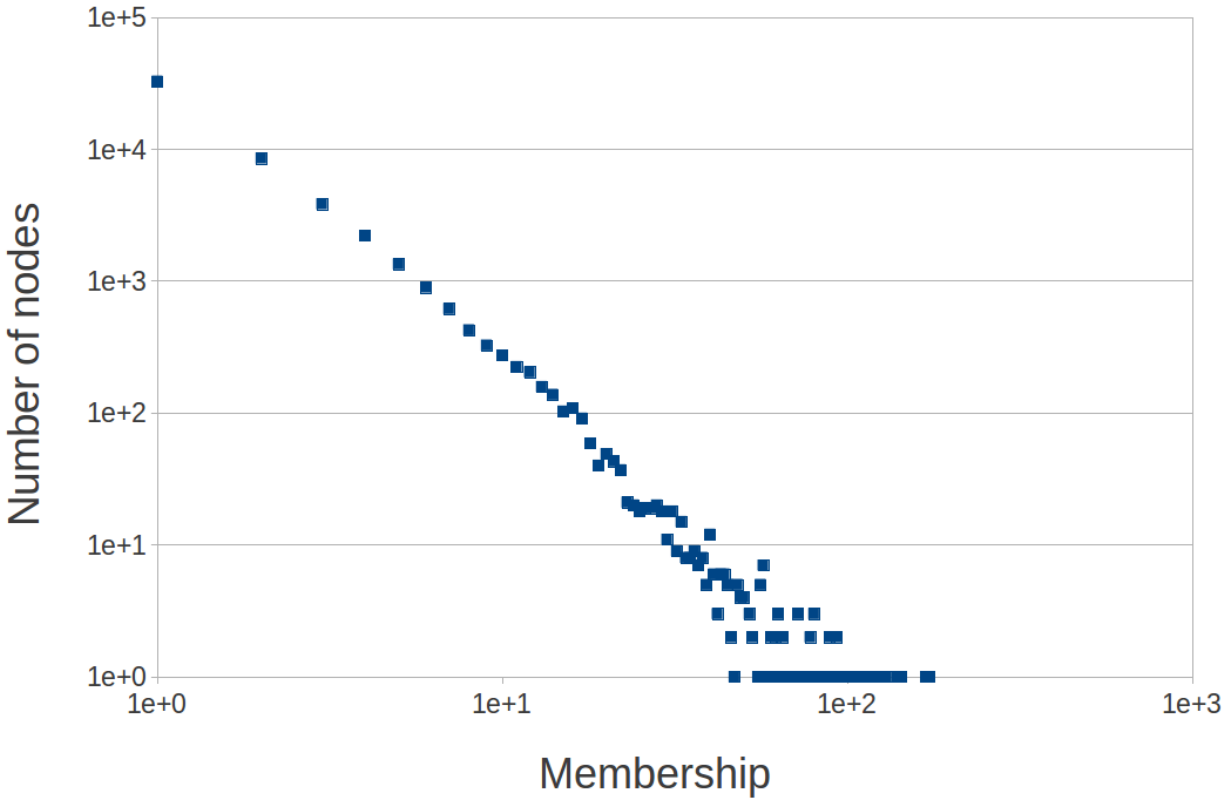


CKB

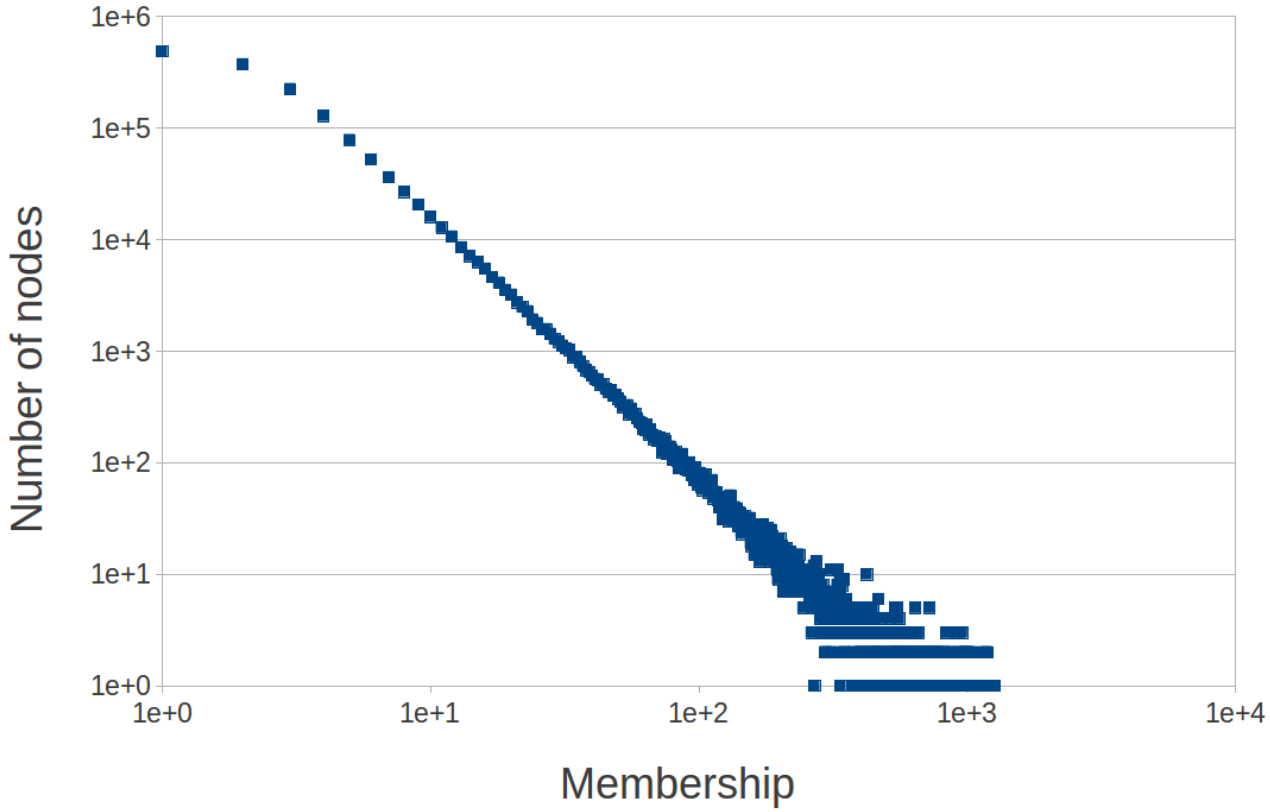


Comparing YouTube and CKB

YouTube

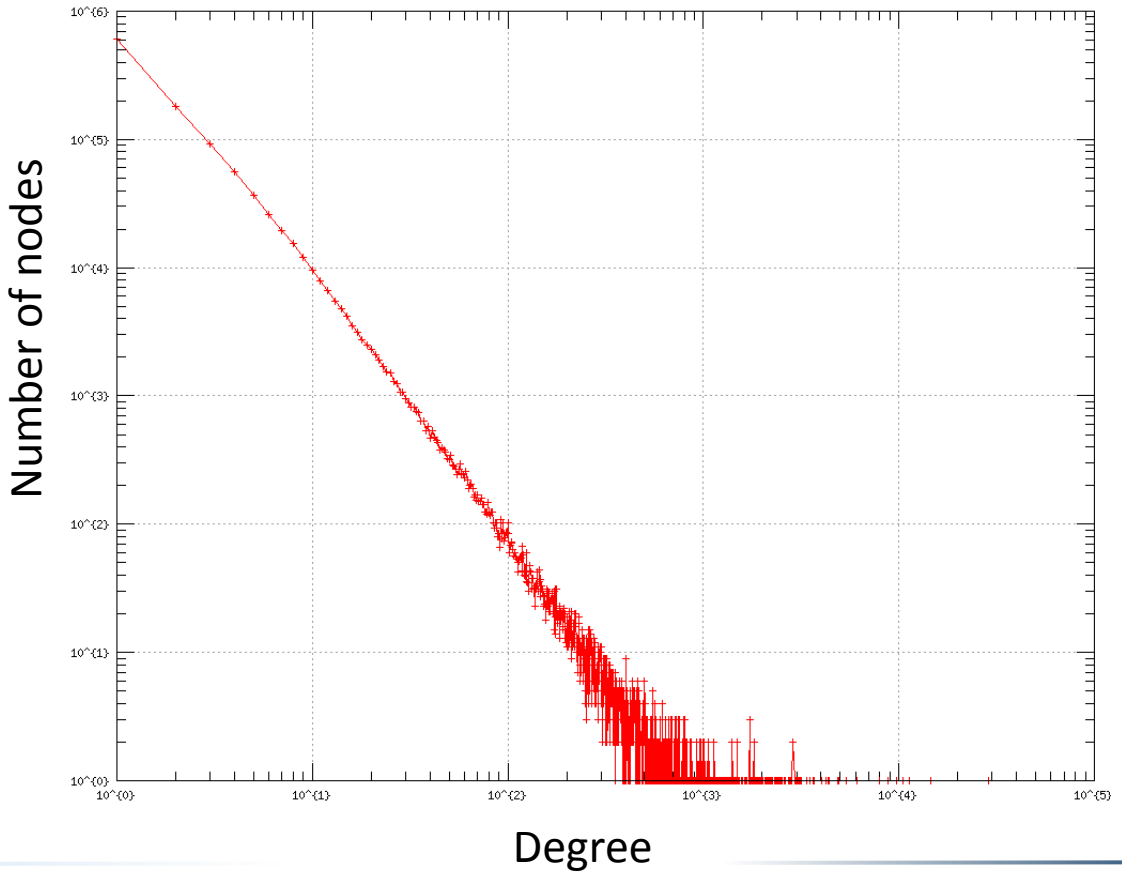


CKB

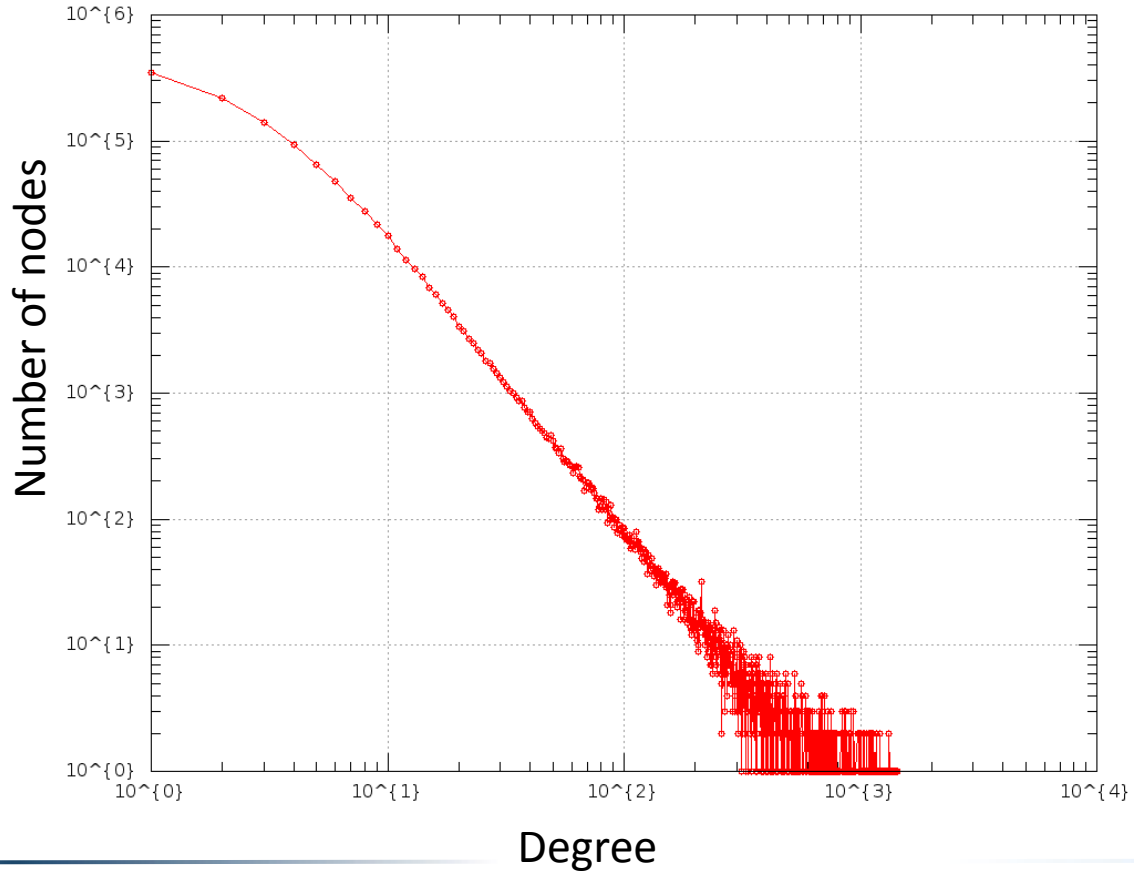


Comparing YouTube and CKB

YouTube

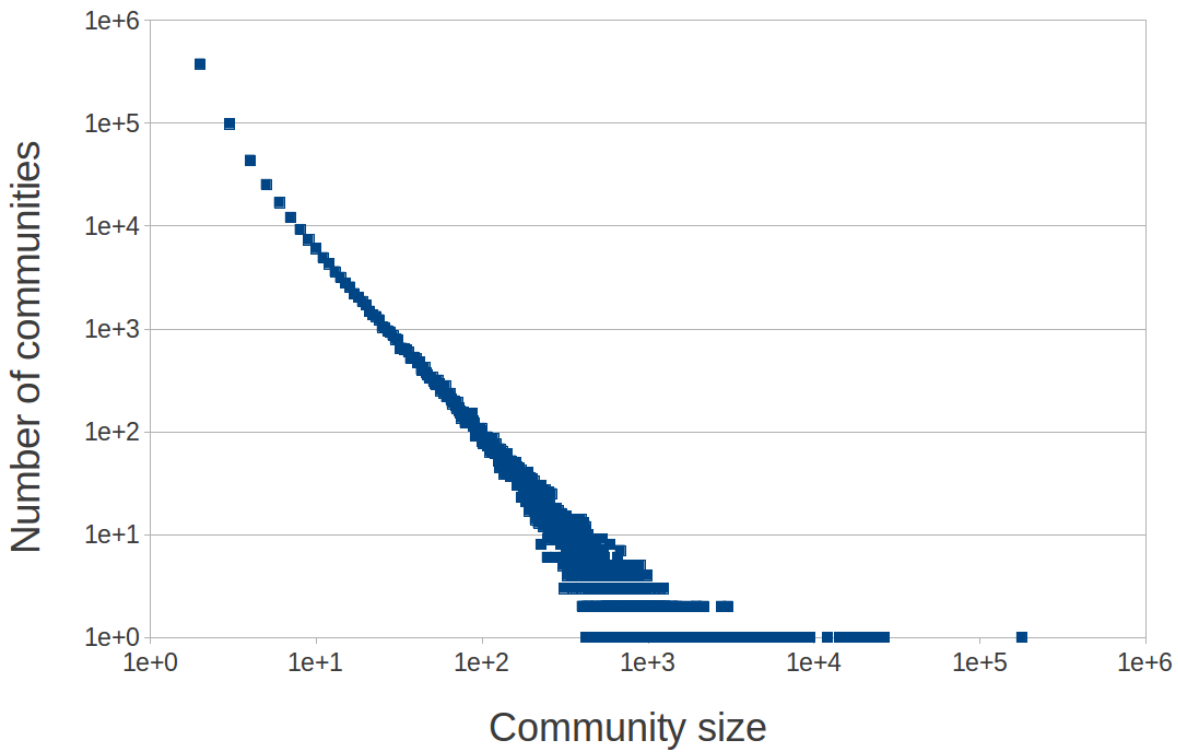


CKB

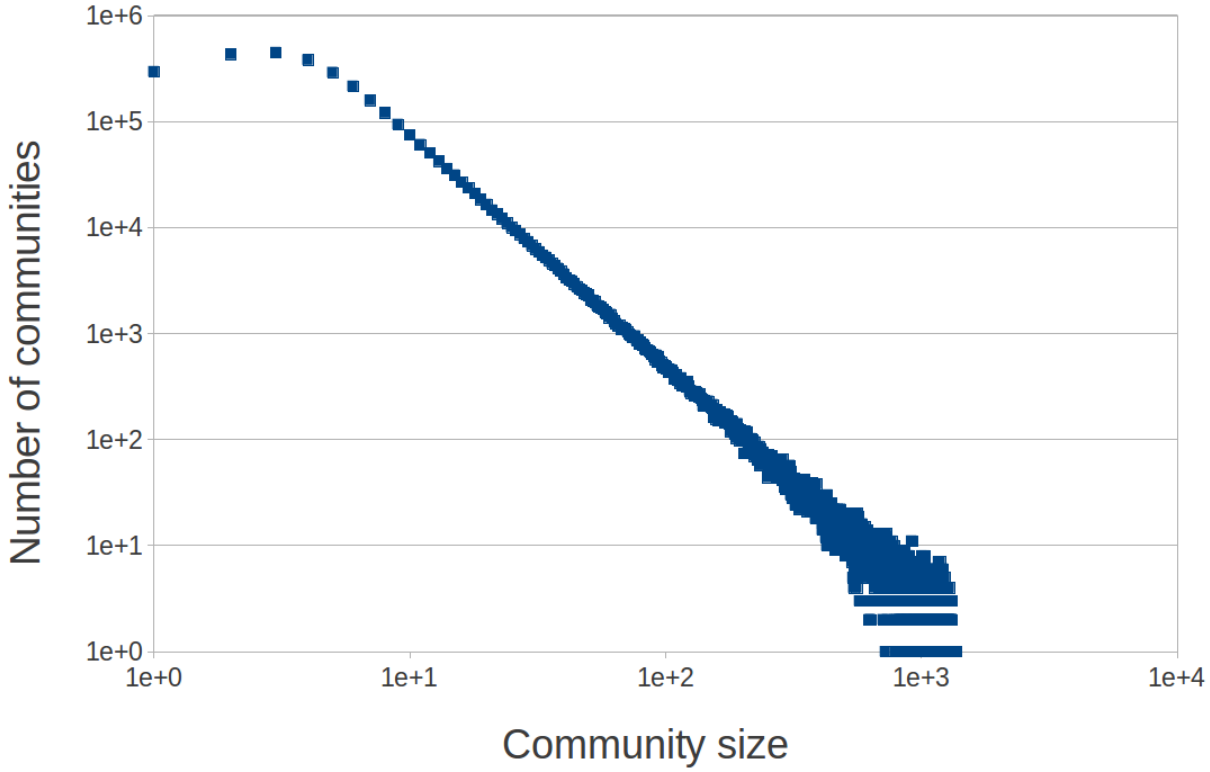


Comparing LiveJournal and CKB

LiveJournal

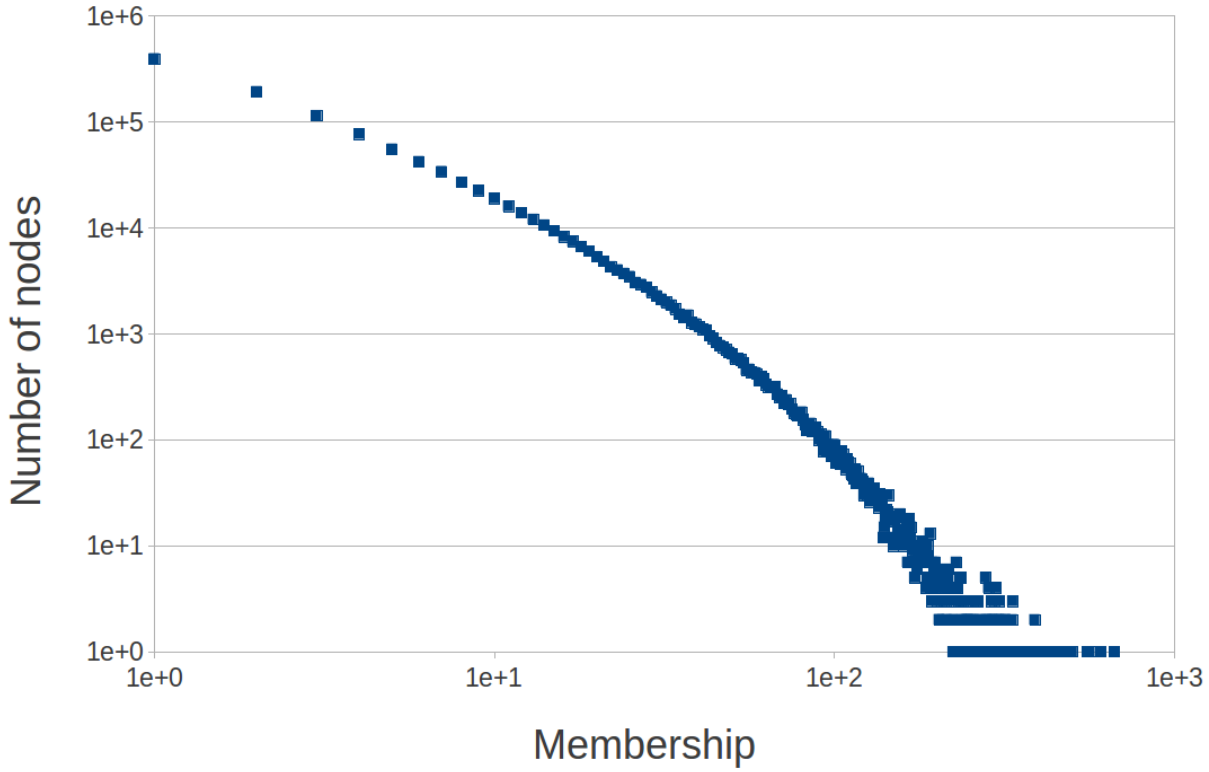


CKB

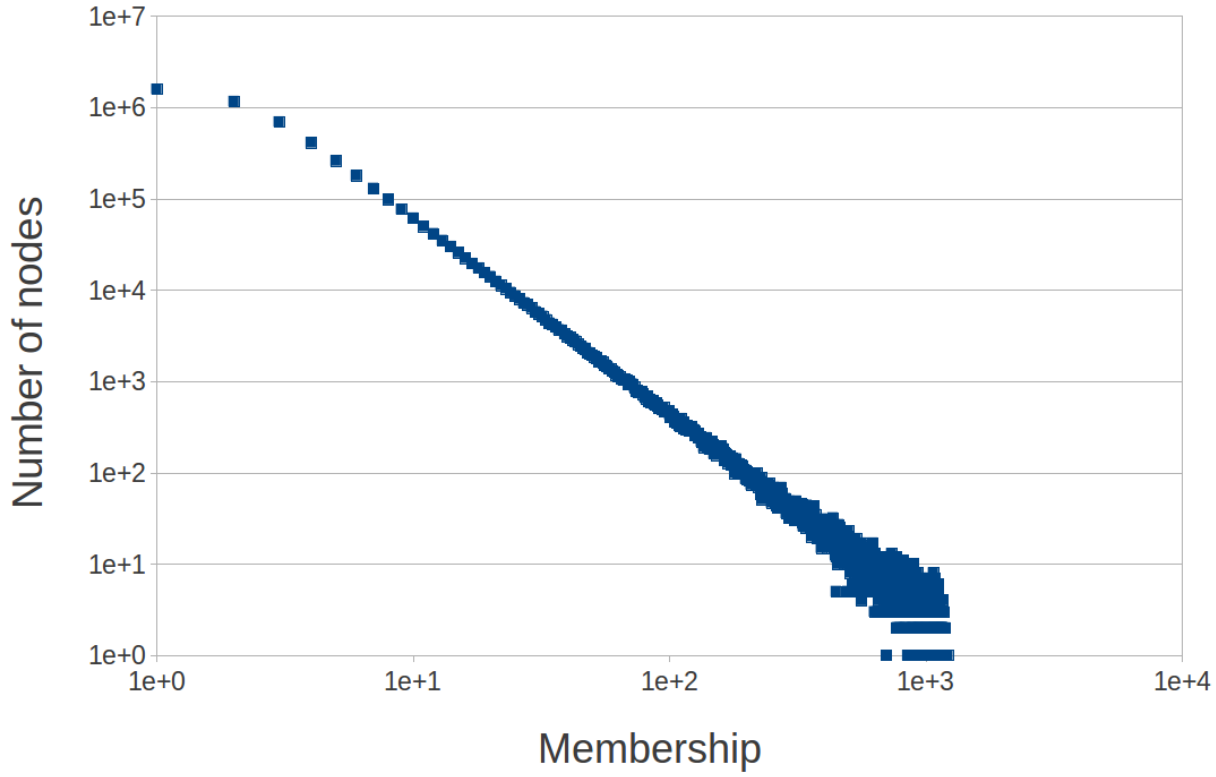


Comparing LiveJournal and CKB

LiveJournal

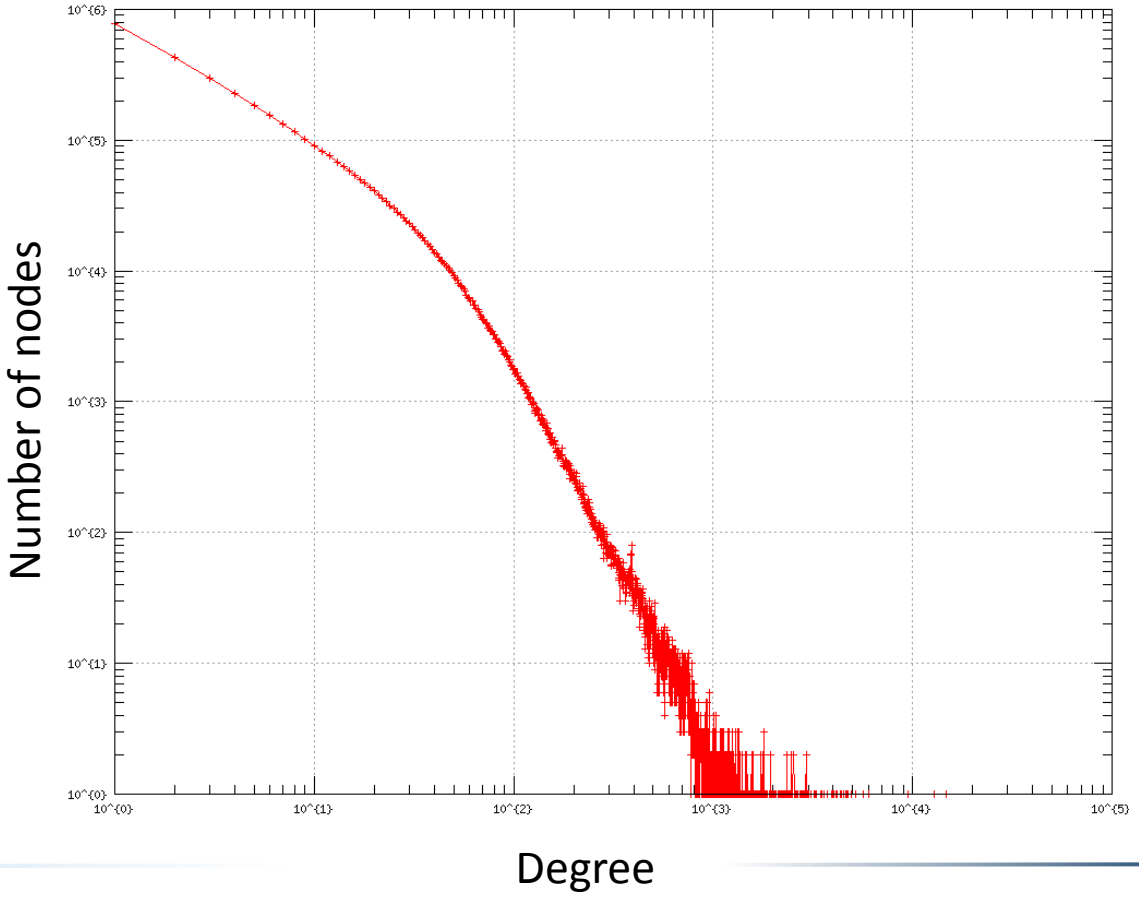


CKB

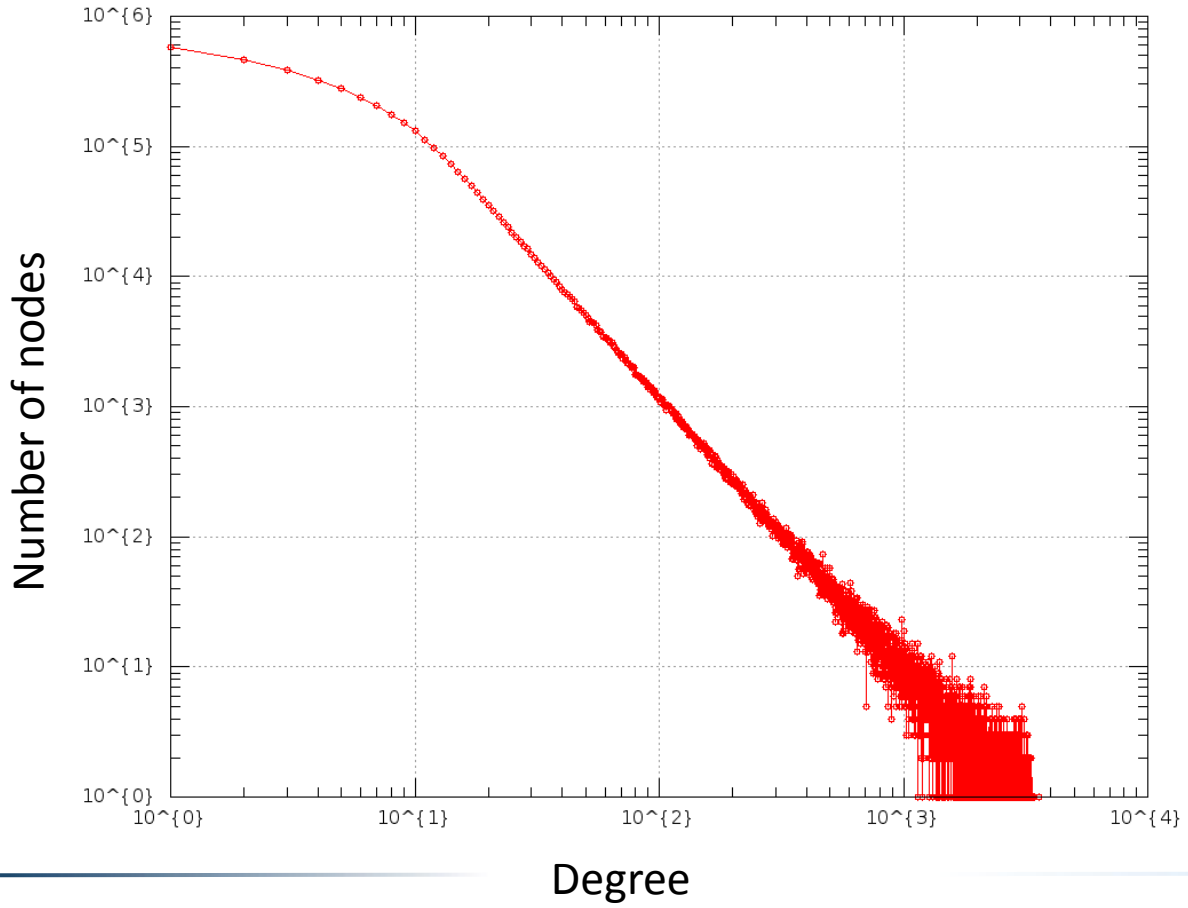


Comparing LiveJournal and CKB

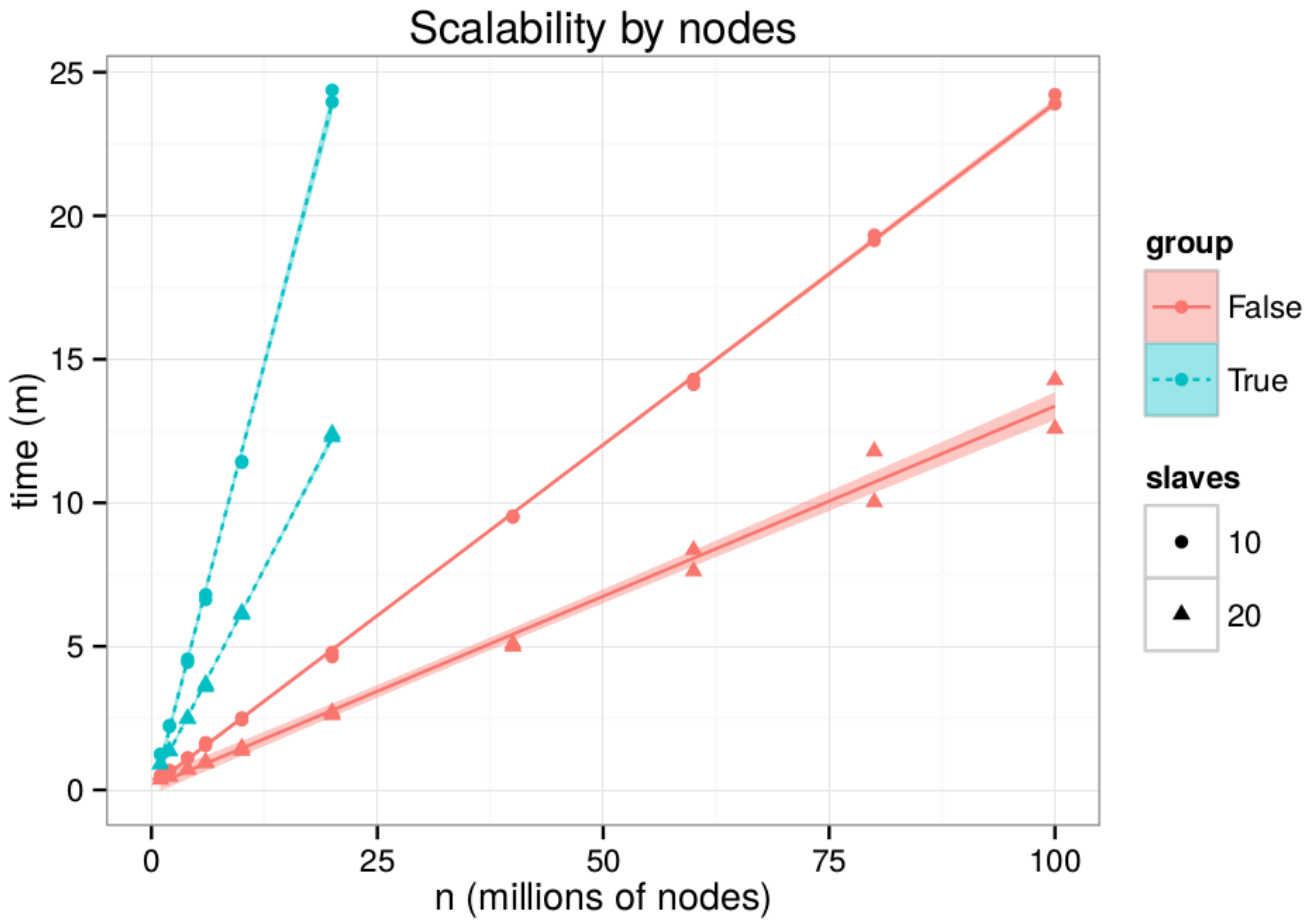
LiveJournal



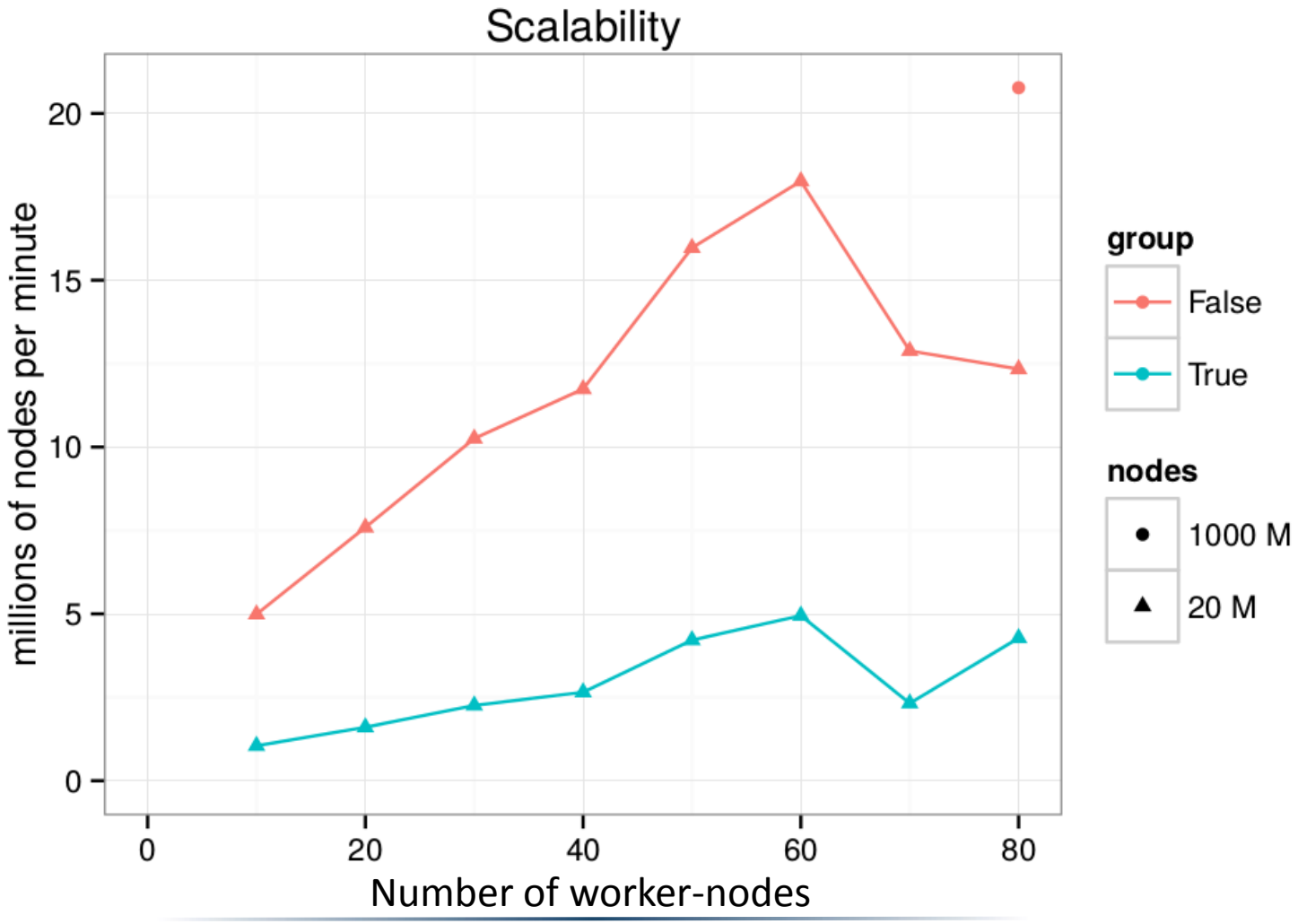
CKB



Scalability: RGG



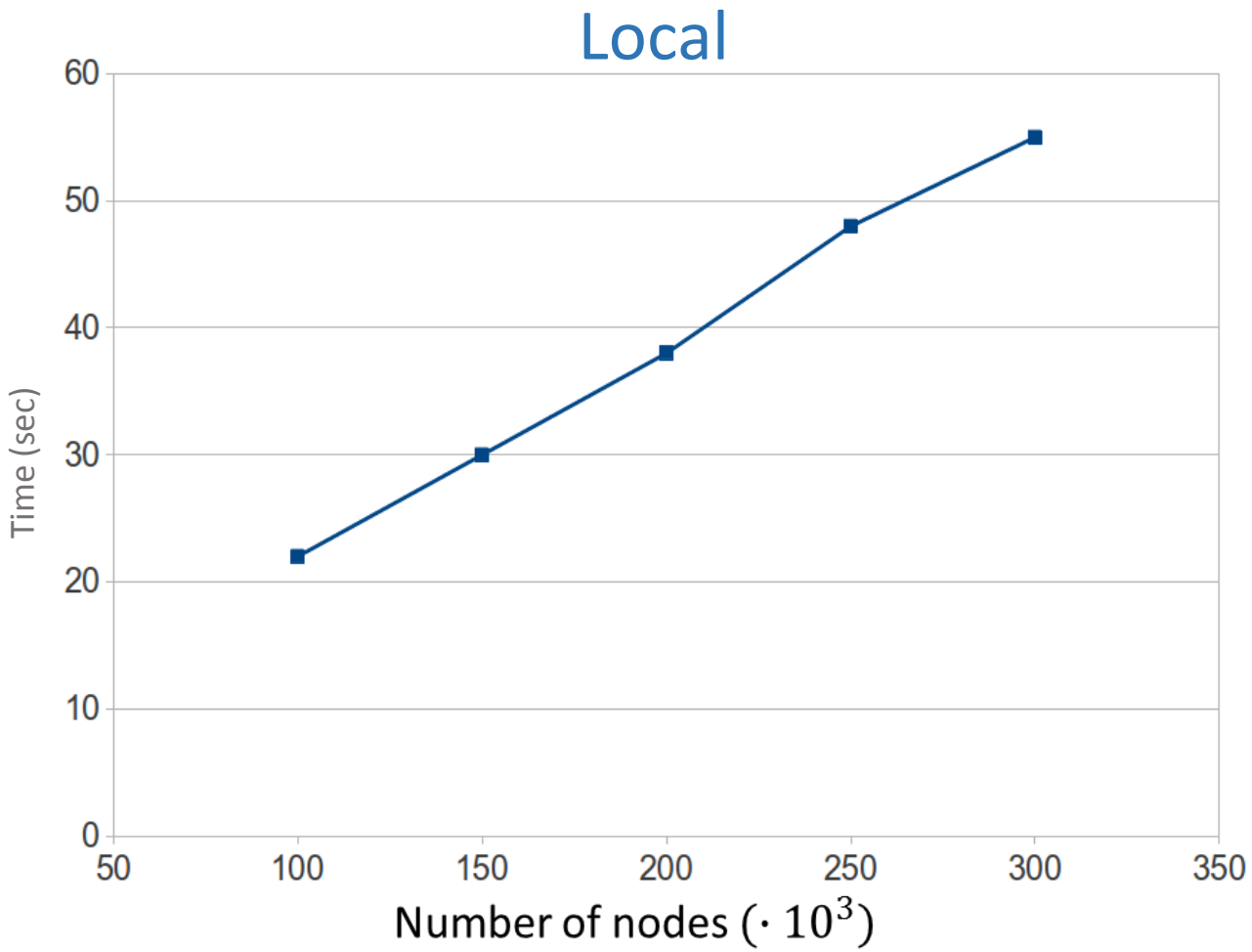
Scalability: RGG



Scalability: CKB

Parameters of generation for scalability testing:

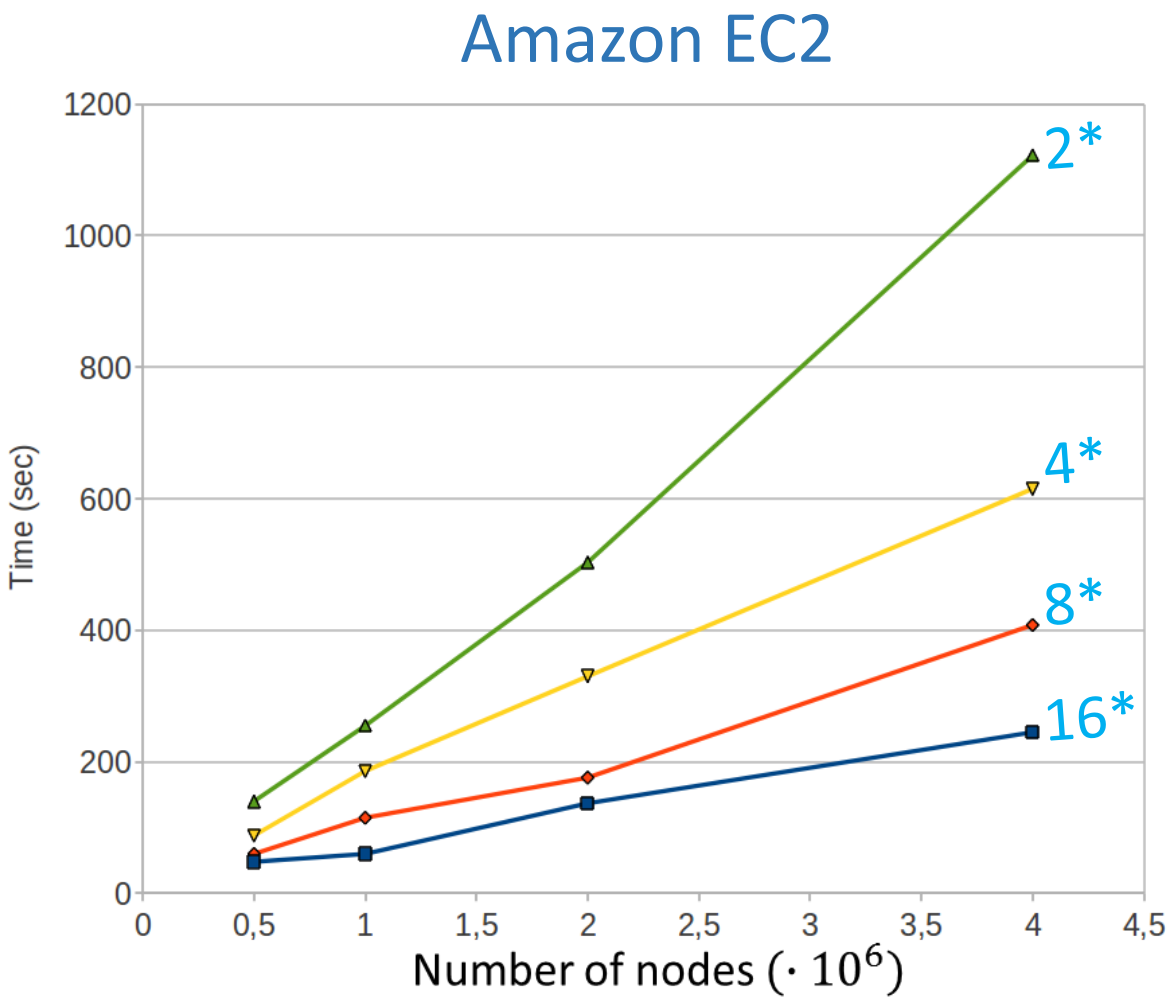
- $\beta_1=\beta_2=2.5$
- $\alpha=4$, $\gamma=0.5$
- $\min(m_i)=1$
- $\min(c_i)=2$
- $\max(m_i)=\max(c_i)=10{,}000$



Scalability: CKB

Parameters of generation for scalability testing:

- $\beta_1=\beta_2=2.5$
- $\alpha=4$, $\gamma=0.5$
- $\min(m_i)=1$
- $\min(c_i)=2$
- $\max(m_i)=\max(c_i)=10{,}000$



*The numbers at the end of lines indicate the number of machines m1.large** in cluster which was used for generation

**m1.large – type of the machine on Amazon EC2 cluster (2 vCPU, 7.5 GiB memory, 2x420GB instance storage).

ISP RAS

Outline

- About Spark
- Task definition
- RGG: graphs without a community structure
- CKB: graphs with a community structure
- Testing
- Conclusions

Conclusions

Described tools create real network and have a number of advantages:

- Graphs with social network properties
- Linear scalability
- A billion-node generation takes reasonable time (55 (150) minutes on 80 (150) machines for RGG (CKB))
- Flexibility of the models

ISP RAS

Open questions

- Testing of various community detection algorithms
- Support the variation of clustering coefficient
- Weighted and hierarchical versions
- Community generation based on attributes and content

The End

Questions?

chykhradze@ispras.ru