ONJAG, network overlays supporting distributed graph processing

Andrea Esposito

Supervisors
Laura Ricci, Patrizio Dazzi

and1989@gmail.com

July 25, 2014

Outline

- **Big graphs**
- distributed graph processing
- towards a new abstraction $\Rightarrow$

- ✍ created *"Overlays Not Just a Graph"*
- ✍ adapted and ported well-known solutions in the P2P
- ✍ extensively studied an interesting recent proposal highly suitable to the abstraction
- ✍ proposed a novel solution improving a pre-existent algorithm exploiting overlays

Big Graphs & Distributed graph processing

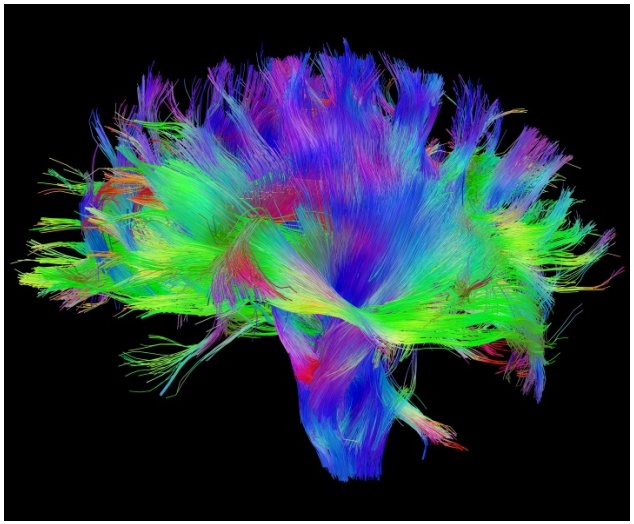
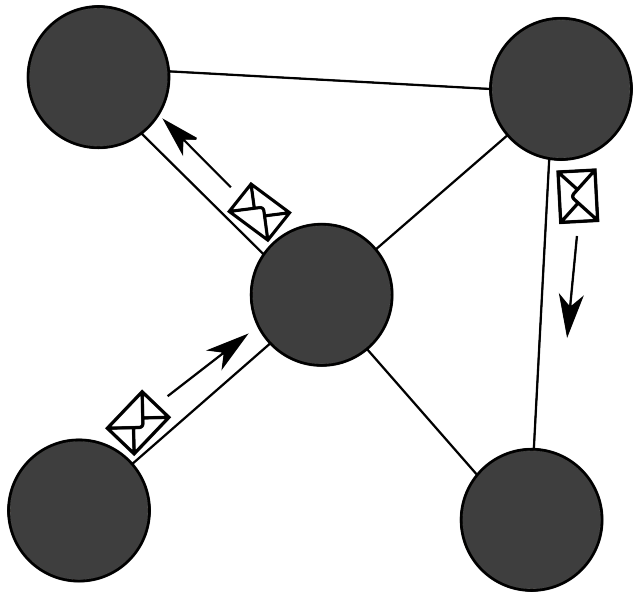


Figure: White Matters - Human Connectome Project

Figure: Yahoo! Cluster @ OSCON 2007

Distributed execution over several machines synchronized by a communication barrier.
Currently inspired frameworks: Hadoop and Pregel.

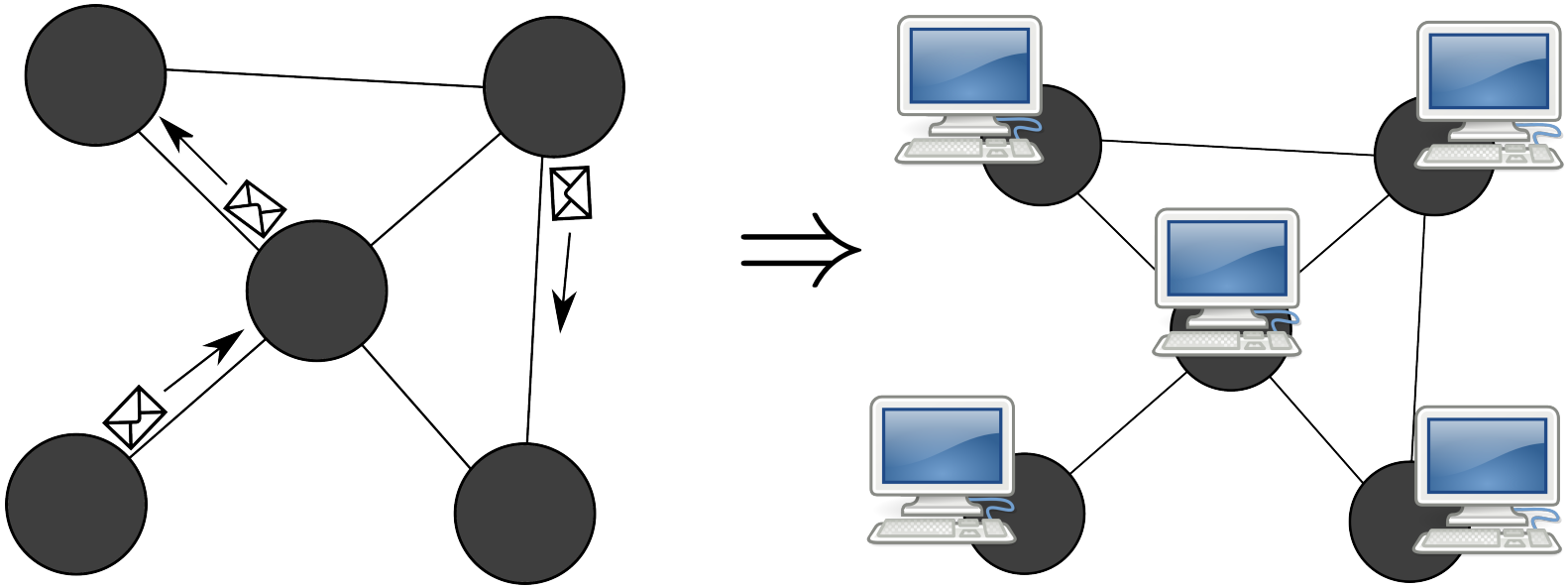
Pregel



..Think as a Vertex..

function vertexLogic:
$(curVertex, Set<InMessages>) \Rightarrow (updVertex, Set<OutMessages>)$

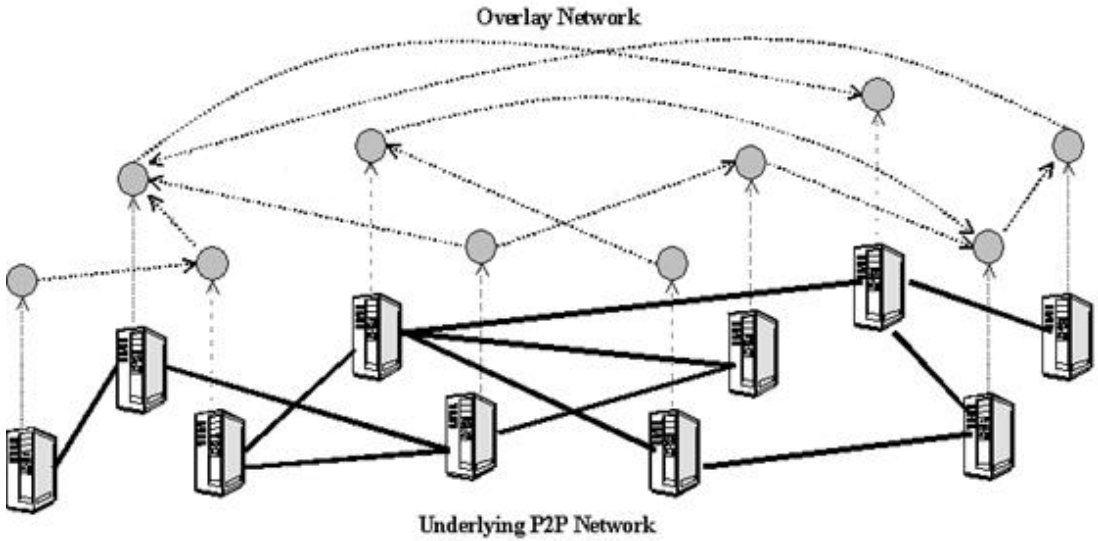
Graph as a Network



Network

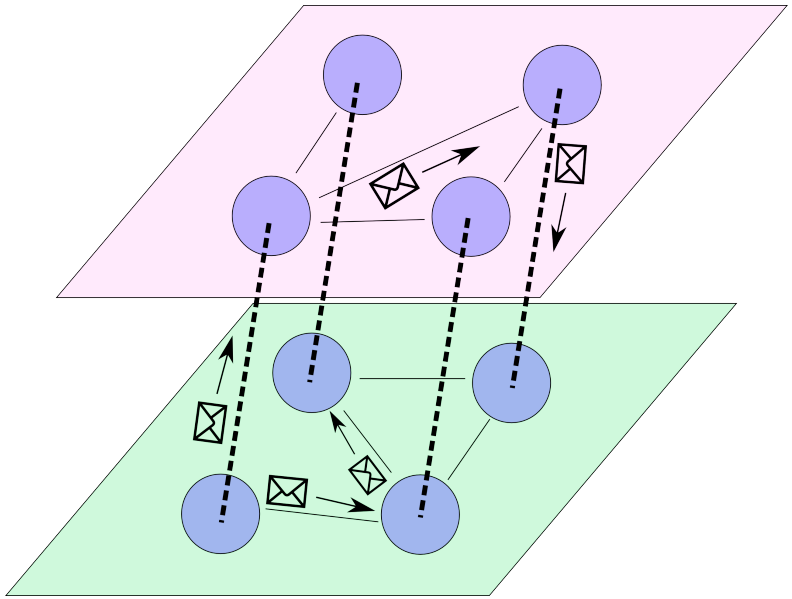
- effective and well-known environment
- modularity achieved by protocols
- availability of [backgrounded] network services
- nodes as computational unit
- overlay abstraction

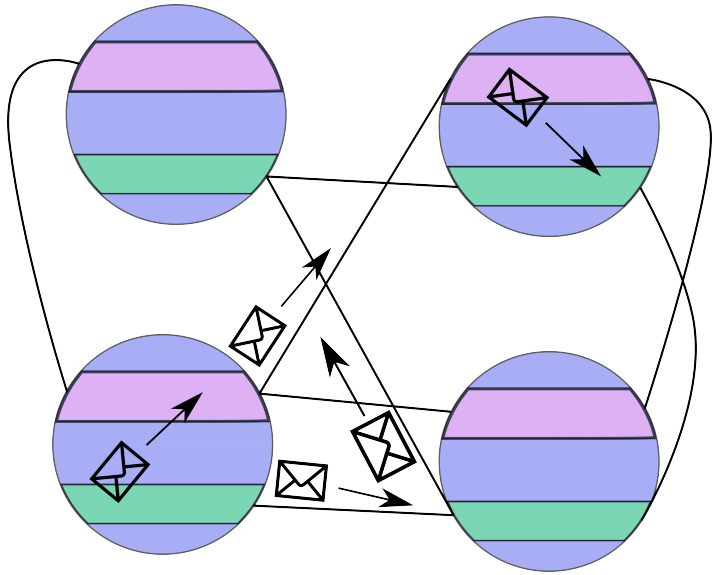
Framework motivations



- existing tools are limited
- improved re-usability of algorithms
- scale of distributed data processing environments
- topology overlay exploitation
- adopting P2P-like methodologies for graph processing

ONJAG





Overlays Not Just A Graph

A multi-layered graph abstraction

- Intra and inter protocol communications in order to orchestrate a complex but convenient computation
- Development of a communication protocol stack emphasizing the topology overlays

Facts

ONJAG

- Spark *Lightning-Fast Cluster Computing*
- about 7000 lines of code in Scala language (JVM based)
- licensed under the *Apache 2* license and publicly available at https://github.com/roy20021/ONJAG
- We developed a Protocols Toolbox, which includes:
  - Distributed *k*-core decomposition

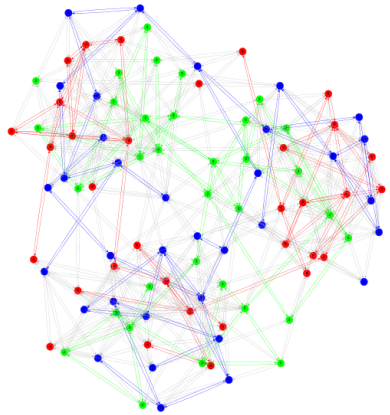
    Montresor A. and De Pellegrini F. and Miorandi D.. *"Distributed k-Core Decomposition"*. IEEE Transactions on Parallel and Distributed Systems, 2013.
  - Random Peer Sampling

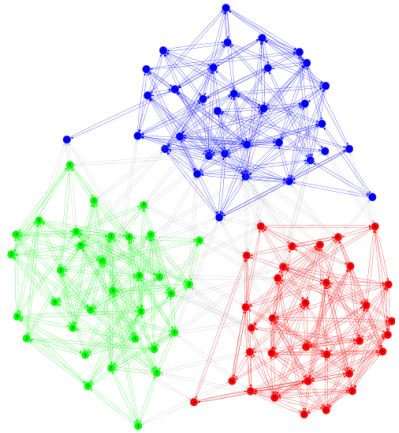
    Jelasity M. and Voulgaris S. and Guerraoui R. and Kermarrec A. and Van Steen M.. *"Gossip-based Peer Sampling"*. ACM Trans. Comput. Syst., 2007.
  - T-MAN, topology overlay manager

    Jelasity M. and Montresor A. and Babaoglu O.. *"T-Man: Gossip-based Fast Overlay Topology Construction"*. Comput. Netw., 2009.
  - JA-BE-JA, balanced minimum *k*-way graph partitioning

    Rahimian F. and Payberah A. H. and Girdzijauskas S. and Jelasity M. and Haridi S.. *"JA-BE-JA: A Distributed Algorithm for Balanced Graph Partitioning"*. SASO, IEEE. 2013. *Best Paper Award*.

JA-BE-JA (1)





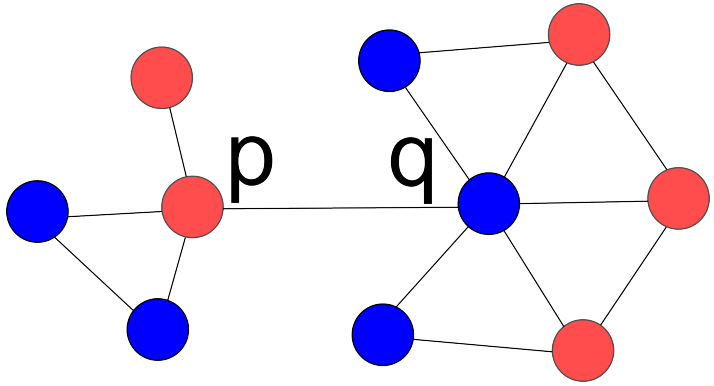
Balanced minimum k-way graph partitioning

Given an undirected graph $G = (V, E)$
partition into $k \in [2, \cdots, |V|]$
disjoint sets $S = \{P_1, P_2, \cdots, P_k\}$ minimizing the **edge-cut**, namely:

$$arg\ min_{P_1 \cdots P_k} \sum_{i=1}^{k-1} \sum_{j=i+1}^{k} \sum_{v_i \in P_i, v_j \in P_j} w(v_i, v_j)$$

where $w(a, b) = |\{(a, b)\} \cap E|$

JA-BE-JA (2)

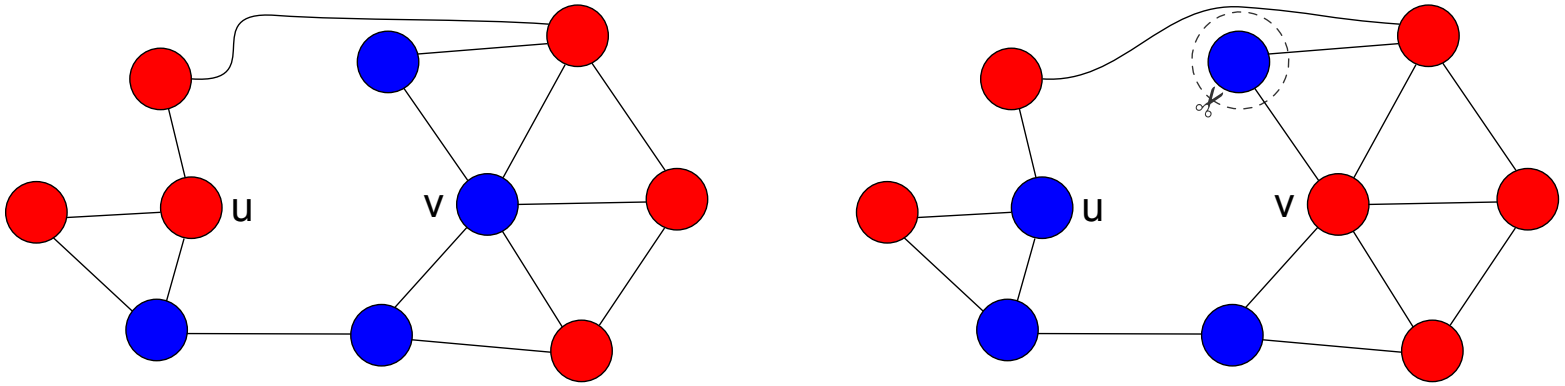


Local search optimization

- aims at extremely large graphs
- heuristic, simple, asynchronous vertex-centric algorithm
- Simulated Annealing against local minima
- update decision criterion:

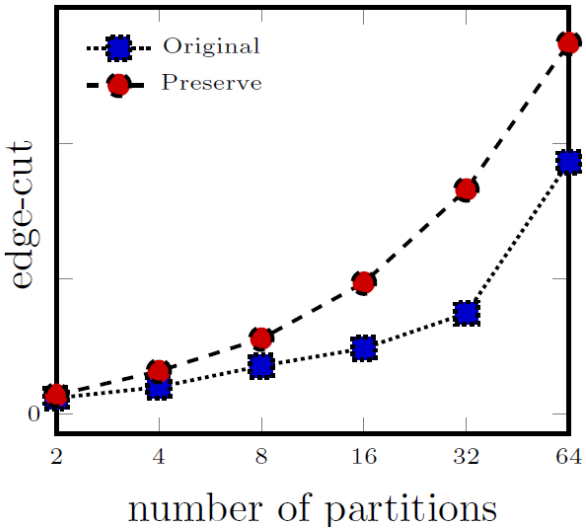
$$(d_p(\pi_q)^\alpha + d_q(\pi_p)^\alpha) \times T > d_p(\pi_p)^\alpha + d_q(\pi_q)^\alpha$$

JA-BE-JA Issues

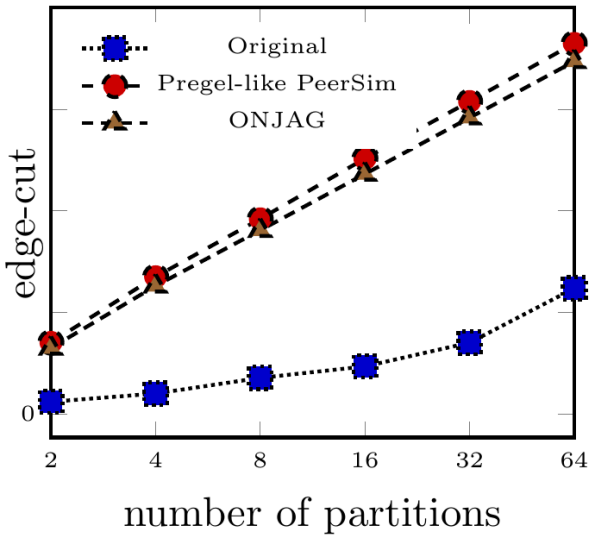


- does NOT preserve the *connected components* property
  ⇒ NO proper mincut!
  ✓ Fixed and re-evaluated in a P2P-like environment using PeerSim
- requires neighbours and neighbours neighbourhood's colors
  ⇒ NO easy portable to Pregel-like frameworks
  ✍ JA-BE-JA over a custom made Pregel-like PeerSim
  ✓ Implemented additional orchestrations to maintain original assumptions in the ONJAG platform
  ★ Implemented Approximated version on ONJAG

Experiments (1)



PeerSim

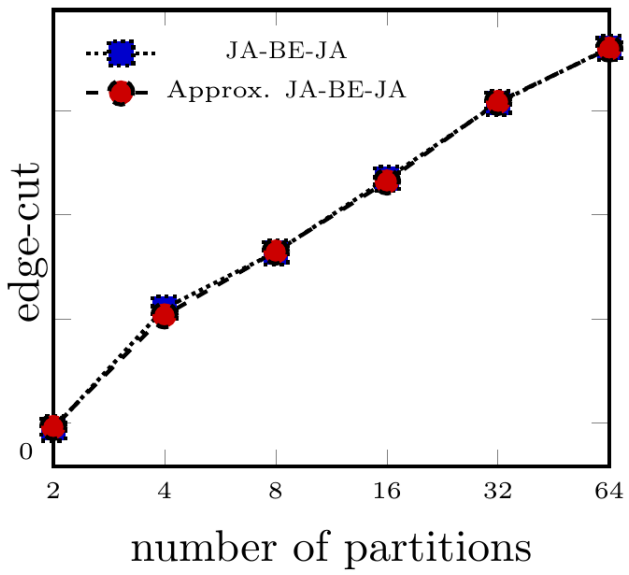


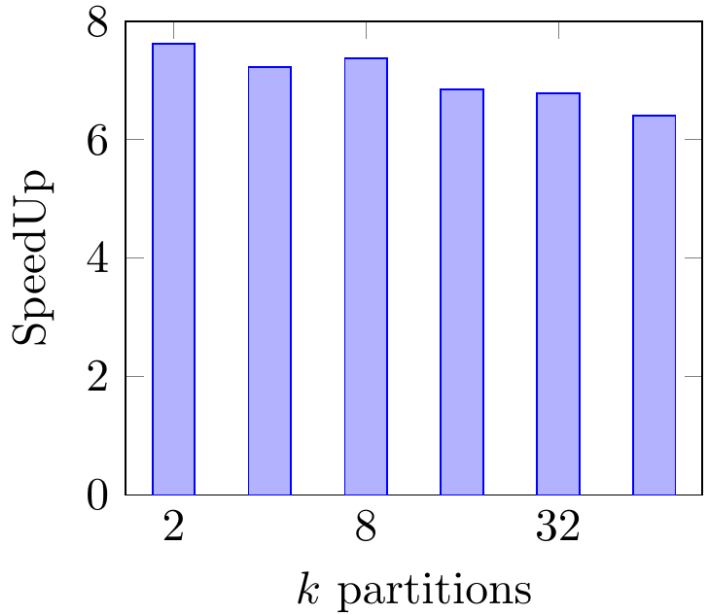
ONJAG

- The preservation of the connected components property affects the JA-BE-JA's effectiveness.
- The Pregel-like PeerSim environment obtains worse edge-cut results.
- The ONJAG platform is well simulated by the Pregel-like PeerSim environment.

Evaluation over 6 graphs: *3elt*, *4elt*, *vibrobox*, *Twitter* and *Facebook*.
Smallest: $|V| = 2395$ and $|E| = 7462$
Biggest: $|V| = 63731$ and $|E| = 817090$

Experiments (2)



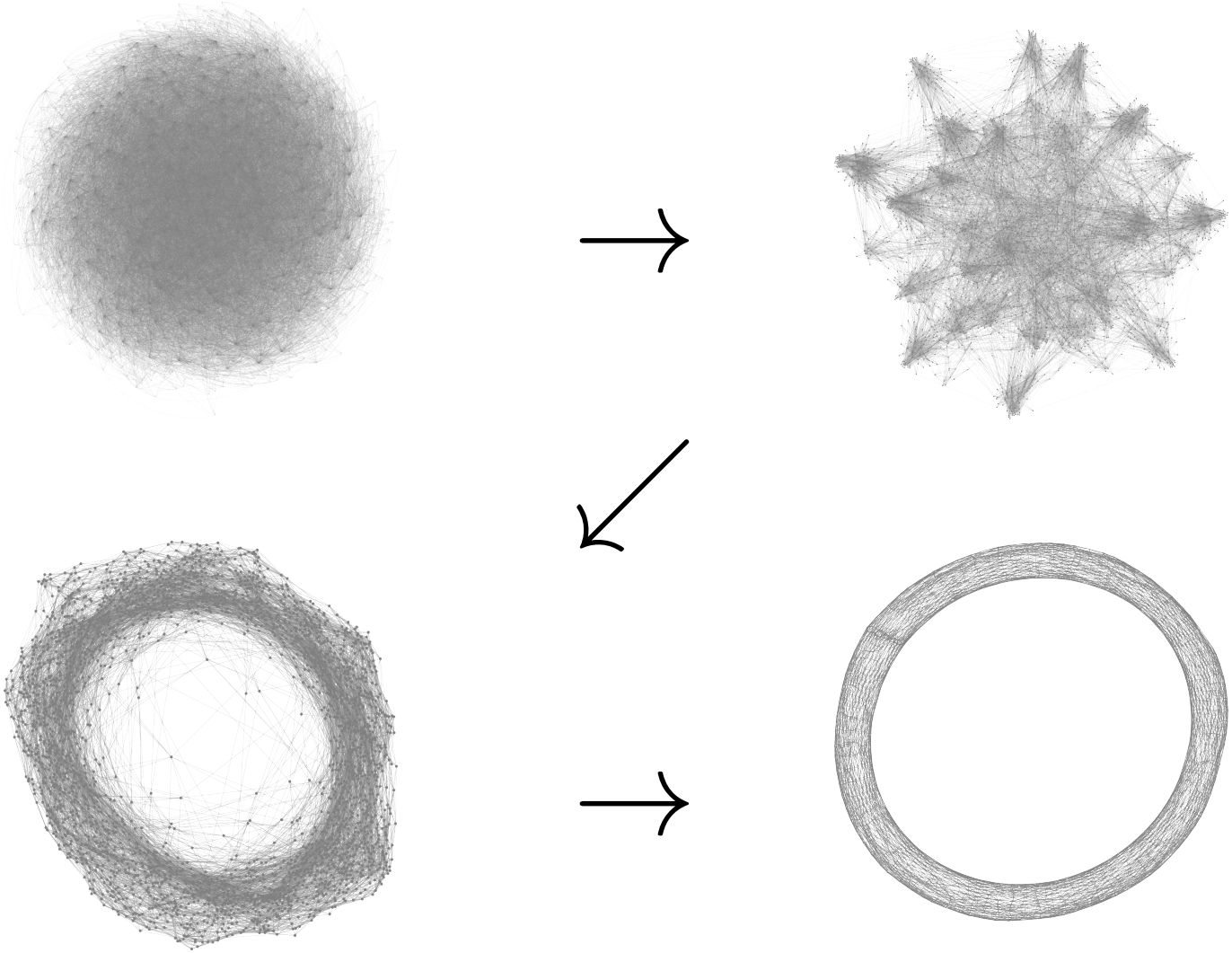


Approximated JA-BE-JA

✍ Achieved approximately same results requiring less computational efforts.

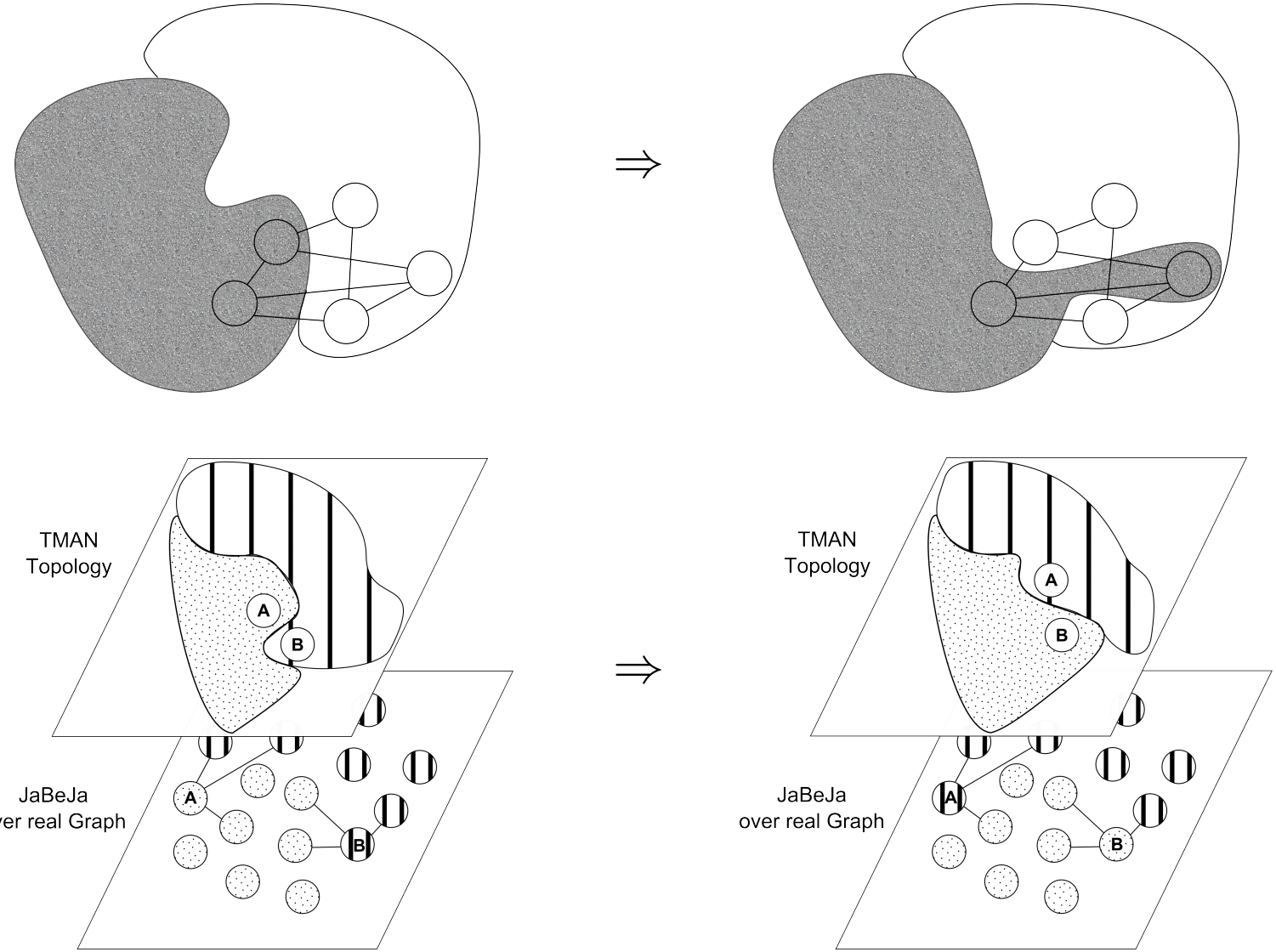
- simpler orchestration
- reduced number of messages
- faster computations

Torus overlay by T-MAN

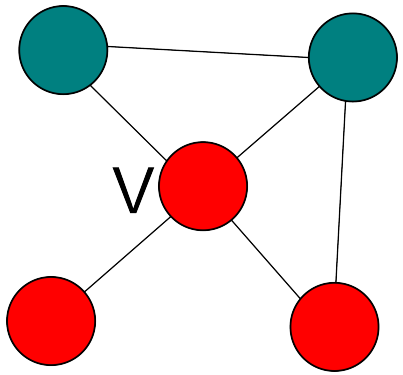


T-MAN uses a customizable *ranking function*

Stain metaphor



Exploiting T-MAN



- decisional equation evaluated over the T-MAN overlay
- swaps occur over the JA-BE-JA overlay
- "borderness" *ranking function*:

$$rankFunction(peerA, peerB) = \begin{cases} +\infty, & \text{if peerA.color} = \text{peerB.color} \\ |H_A - H_B|, & \text{otherwise} \end{cases}$$

$$\text{where } H_P = \frac{d_p(\pi_p)}{|N_p|}$$

Normalized Quasi cut (NQcut)

A qualitative measure for scattered partitions

- adopted for evaluating properly our proposal
- derived from the *Normalized cut* by Shi-Malik (extended to k partitions):

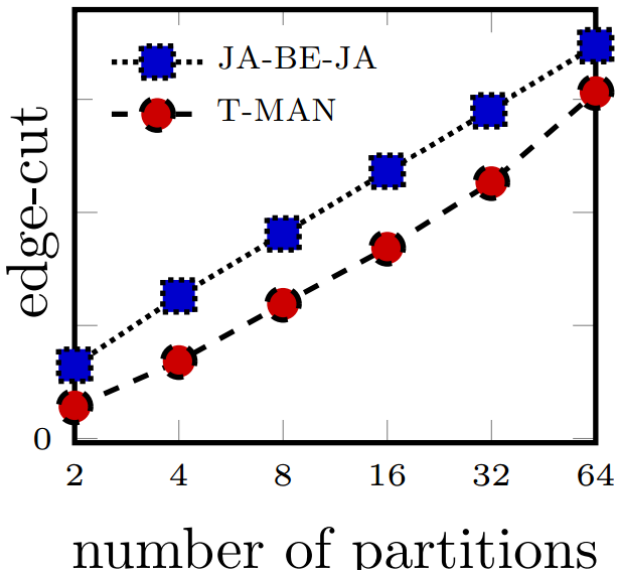
$$Nassocc(A,B) = \frac{assocc(A,A)}{assocc(A,V)} + \frac{assocc(B,B)}{assocc(B,V)} \qquad Ncut(A,B) = 2 - Nassocc(A,B)$$

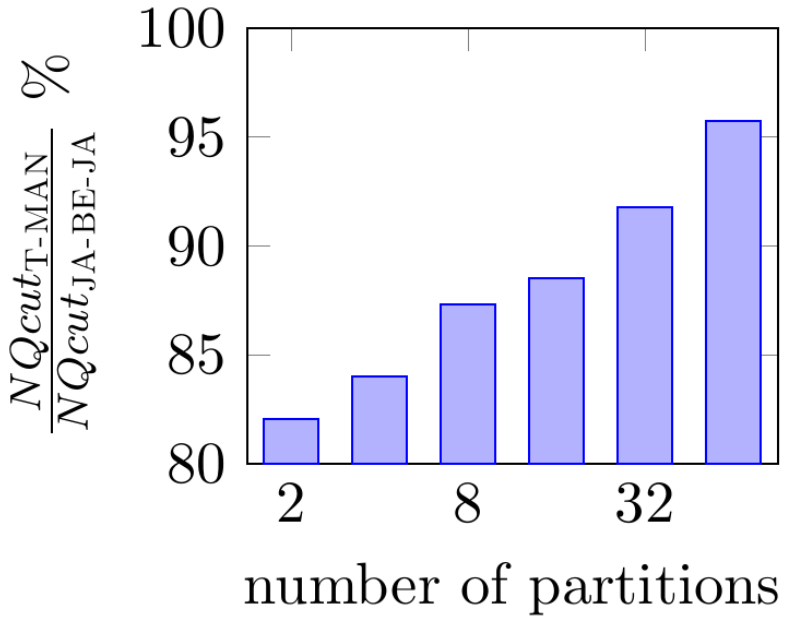
$$\text{where } assocc(X,Y) = \sum_{x \in X, y \in Y} w(x,y)$$

- $f(X_i)$ function that measures the fragmentation quality of a partition X_i

Shi J. and Malik J.. *"Normalized Cuts and Image Segmentation"*. IEEE Trans. Pattern Anal. Mach. Intell. 2000.

Experiments (1)

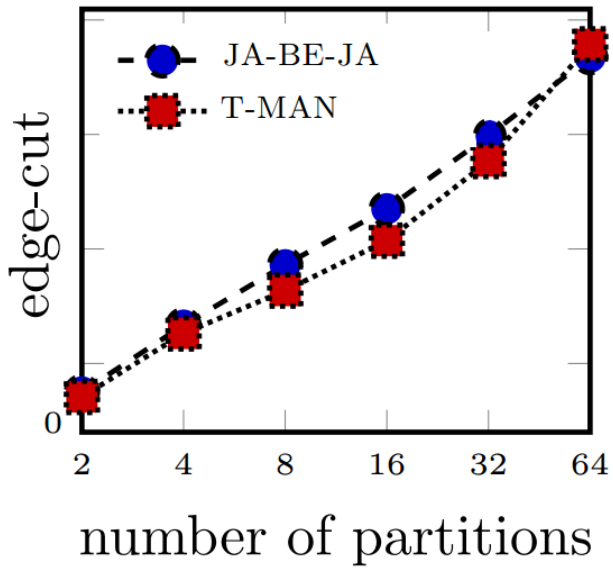


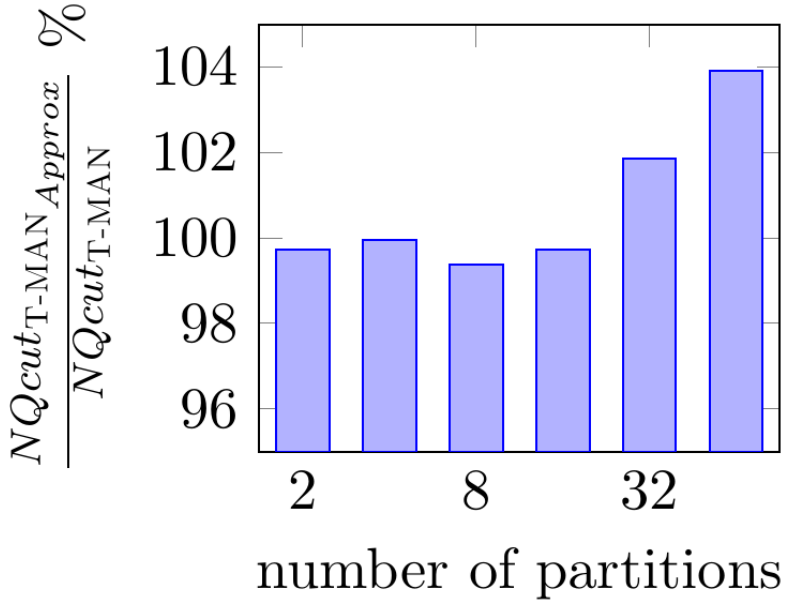


ORIGINAL JA-BE-JA vs JA-BE-JA + T-MAN

- ✍ The T-MAN add-on improves final partitioning with respect to *edge-cut* and *NQcut*.
- ✍ The T-MAN add-on does not scattered more than the original algorithm.

Experiments (2)





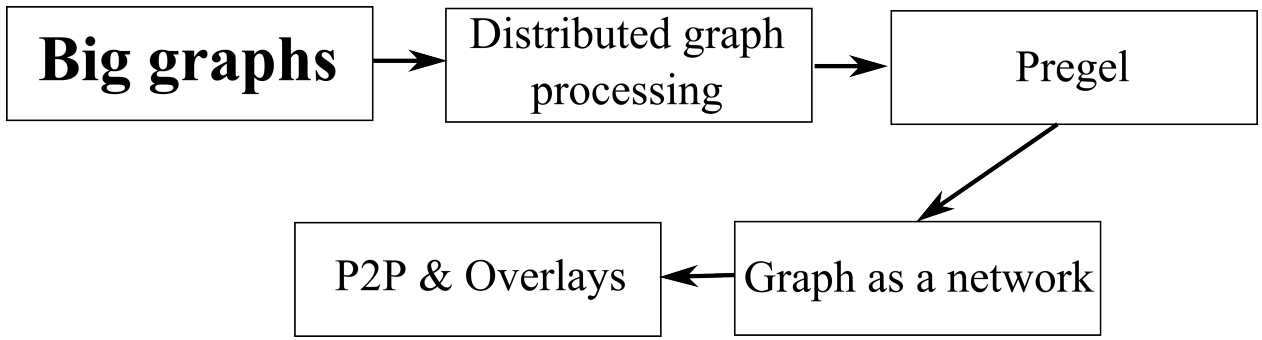
Approximated JA-BE-JA + T-MAN

vs

the NOT Approximated one

✍ Approximated version achieves same results!

Summary (1)



- conceived and designed **"Overlays Not Just a Graph"** abstraction and implemented on the Spark framework.
- adapted and ported well-known solutions in the P2P into the distributed graph processing.

Summary (2)

- ✍ extensively studied, analysed and enhanced JA-BE-JA
- ✍ re-evaluated considering the preservation of the connected components property in a P2P-like environment.
- ✍ re-arranged JA-BE-JA to run over ONJAG.
- ✍ approximated orchestration that performs similarly requiring less resources.
- ✍ a novel solution exploiting overlays and ONJAG.
- ✍ enhanced and improved results.

Current Achievements

- ONJAG is licensed under the *Apache 2* license and publicly available at https://github.com/roy20021/ONJAG
-  Carlini E., Dazzi P., Esposito A., Lulli A. and Ricci L.. *"Balanced Graph Partitioning with Apache Spark"*. BigDataClouds, Euro-Par. 2014.

Q&A