Massive Graphs: Algorithms, Techniques and Challenges

Ljupco Kocarev
Macedonian Academy of Sciences and Arts, Macedonia
Faculty of Computer Science and Engineering, UKIM, Macedonia
University of California San Diego, San Diego, USA

2016 SEE Data Science Forum
20-21 June 2016, Belgrade, Serbia

Linked (networked) world

Electric power grids, the Internet, the Web, and neural net-
works are all large complex systems that share an important
feature: they are networked.

Networks: many systems/phenomena can be represented
(approximated) as graphs: sets of (weakly) interacting entities.

Networks/graphs from data: what is a node? what is a link?

Social networks: person, group of people, county

Brain: single neuron, group of neurons, or region

Internet: single computer or AS (autonomous system)

Why networks?

- Support different processes: as many processes as networks
- Improve our understanding of systems/phenomena
- Suggest innovative solutions
  - Deep neural networks
  - Online social networks
  - LinkedIn: career paths of more than 60 000 graduates

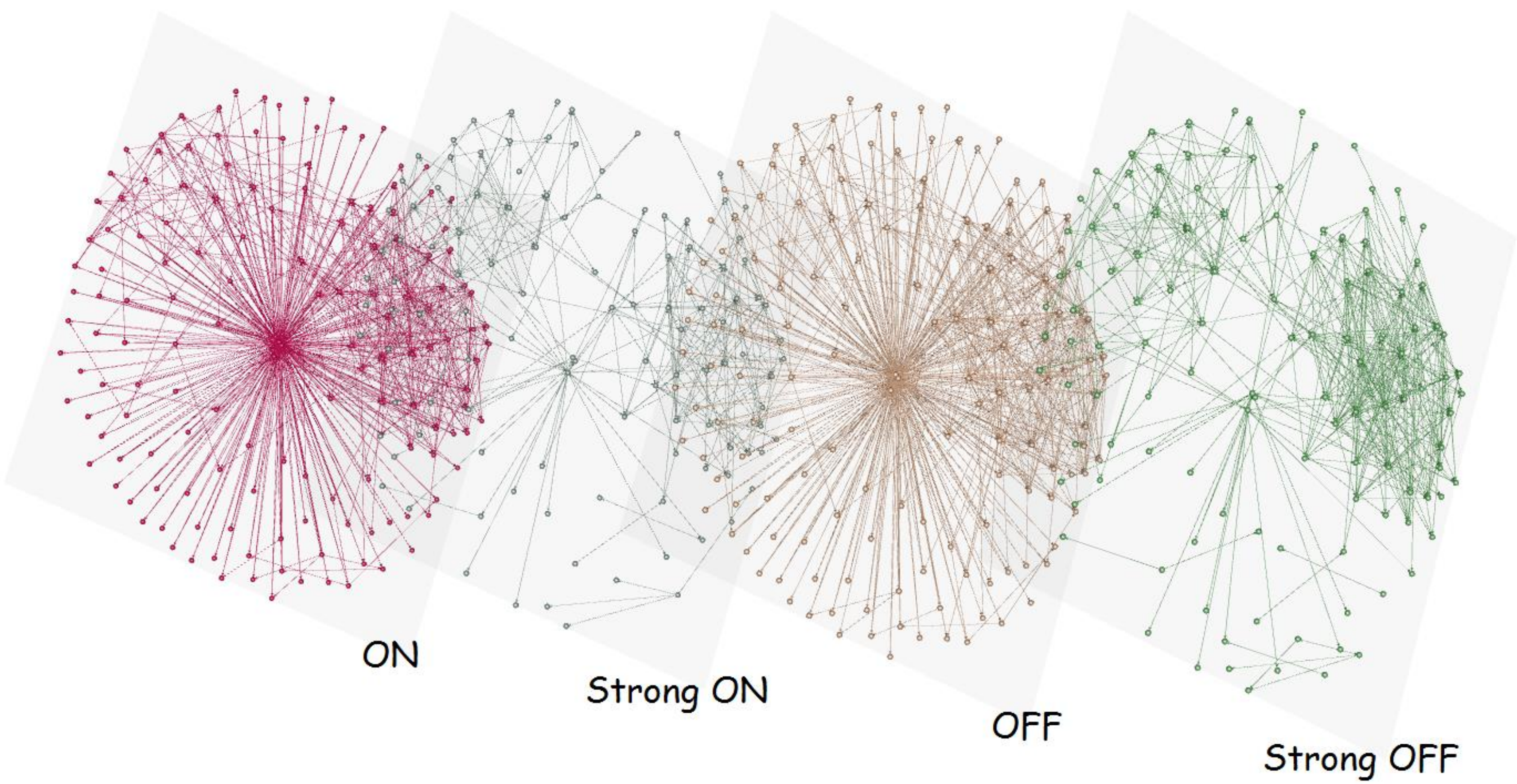
    MIT: Google, IBM, and Oracle

    Purdue: Lilly, Cummins, and Boeing

Social networks

The largest social network is the acquaintance graph of all living people: 7.4 billion (10^9) nodes.

Web 2.0/3.0 technologies have triggered user-generated content on social media making the Web and online social networks ***complex massive networks***.

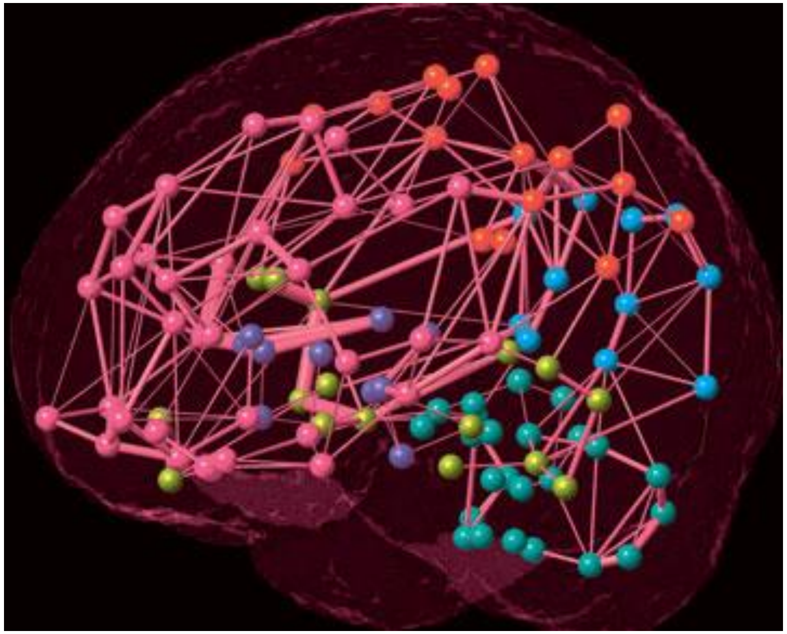


Brain connectome

Mapping of all neural connections

Massive network: the human brain has 10^{11} neurons and 10^{14} connections

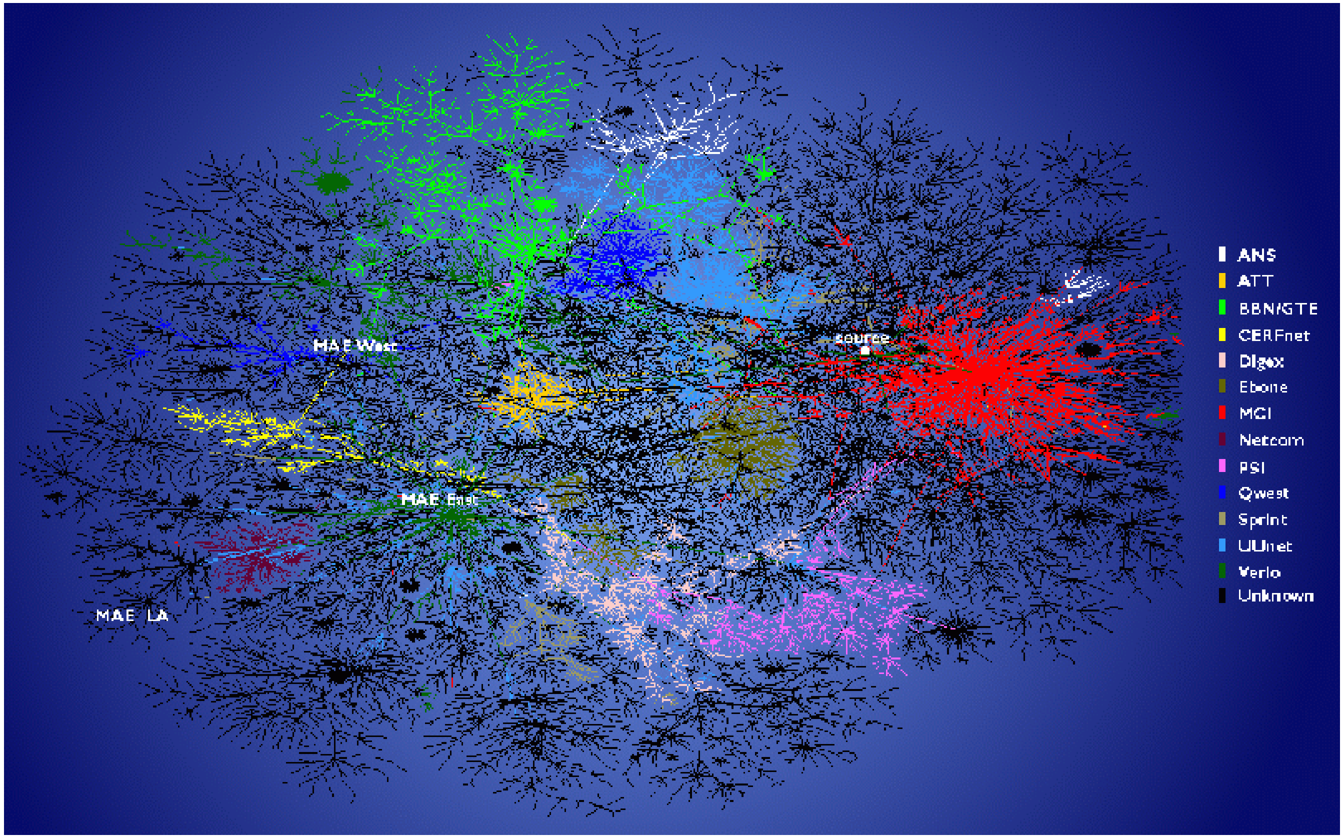
Region of interests: from correlations to connections (structural, functional, and effective brain networks)



Internet

From CAIDA: two-dimensional image depicting global connectivity among ISPs

IPv6 supports 2^{128} ($\sim 10^{38}$) possible addresses: with the Internet of Things ***massive networks*** are expected to become the cornerstone of our societies and modern business



Processing of massive graphs

Distributed graph processing problems:

- Pregel (designed by Google, 2010)
- Giraph (developed by Yahoo, 2012) utilizes Apache Hadoop's MapReduce
- Giraph is an open-source project used worldwide http://giraph.apache.org/
- August 2013 Giraph is used to analyze Facebook graphs with 10^{12} edges (200 machines)

Single machine-sized graph processing problems:

- X-Stream http://labos.epfl.ch/x-stream is capable of analytics on graphs with upwards of 64 billion edges on a ***single machine*** using only two attached 3TB magnetic disks
- M-Flash http://arxiv.org/abs/1506.01406 enables to easily implement essential graph algorithms, including the first ***single-machine billion-scale*** eigensolver
- Chaos (2015) is currently capable of working on graphs with over ***1 trillion (10^{12}) edges*** on n a single rack of commodity machines - a ***milestone*** that could only be reached before using HPC or large clusters consisting of hundreds or thousands of machines

Current big data technologies are inadequate for handling massive graphs, lacking the flexibility to allow **non-expert users** to set up complex analytic tasks, as well as the speed and scalability to support analysis of **massive graphs on commodity hardware**.

- Challenges:
  - Computational models for massive graphs
  - Sub-linear graph algorithms
  - Scalable graph algorithms
  - Local versus global metrics for massive graphs

Challenge 1: Computational models for massive graphs

Computation models for both parallel and stream computing

Classical models of parallel computation: PRAM and BSP (parallel random-access machine and bulk synchronous parallel model)

Stream models (graph stream models)

Computational classes (NC class)

Both Pregel and Giraph are inspired by the Bulk Synchronous Parallel (BSP) model of distributed computation.

The Bulk Synchronous Parallel (BSP) abstract computer is a ***bridging model*** for designing parallel algorithms.

A BSP computer: (1) components capable of processing and/or local memory transactions (i.e., processors), (2) a network that routes messages between pairs of such components, and (3) a hardware facility that allows for the synchronization of all or a subset of components.

Leslie G. Valiant, A bridging model for parallel computation, Communications of the ACM, Volume 33 Issue 8, Aug. 1990

In complexity theory, the class NC, for “Nick's Class“, (Pippenger), is the set of decision problems decidable in poly-logarithmic time on a parallel computer with a polynomial number of processors.

The class P – tractable problems (Cobham's thesis)
NC – problems that can be efficiently solved on a parallel computer

NC = P – most researchers suspect this to be false, meaning that there are probably some tractable problems that are “*inherently sequential*” and cannot significantly be sped up by using parallelism.

However, **rigorous algorithmic** analyses of both parallel and stream models have not been **completely** addressed yet.

Challenge 2: Sub-linear graph algorithms

Sub-linear graph algorithms with **guaranteed approximation**

Sub-linear time algorithms and sub-linear space algorithms

Randomized algorithms, Approximation algorithms

Property test: to distinguish graphs with a given property from graphs that are “far” from having the property

Example: approximating the Average Degree

Estimating the mean and moments of a sequence of *n* integers is a classic problem in statistics.

n numbers: almost all numbers in the input set are 1 and a few of them are $n - 1$.

The average degree of a graph with *n* vertices can be approximated using only $O(n^{0.5}/\varepsilon)$ vertices within a factor of $2 + \varepsilon$.

U. Feige "On sums of independent random variables with unbounded variance, and estimating the average degree in a graph" SICOMP 35(4): 964-984 (2006)

T. Eden, D. Ron, C. Seshadhri, "Sublinear Time Estimation of Degree Distribution Moments: The Arboricity Connection," http://arxiv.org/abs/1604.03661 (April, 2016)

Example: counting triangles

Triangle counting is a key operation in graph analysis for graph modeling, bioinformatics, social networks, and community analysis.

Tool for counting the number of triangles:

- Fast matrix multiplication
- Provable algorithms that employ sampling methods for approximate triangle counting
- Triangle counting in the streaming setting

Many of the algorithms for triangle counting read the ***entire*** graph.

Approximate triangle counting

- M. N Kolountzakis, G. L Miller, R. Peng, and C. E Tsourakakis. Efficient triangle counting in large graphs via degree-based vertex partitioning. Internet Mathematics, 8(1-2):161–185, 2012.
- S. Arifuzzaman, M. Khan, and M. Marathe. Patric: A parallel algorithm for counting triangles in massive networks. In Proceedings of the 22nd ACM international conference on Conference on information & knowledge management, pages 529–538. ACM, 2013.
- C. Seshadhri, A. Pinar, and T. G. Kolda. Fast triangle counting through wedge sampling. In Proceedings of the SIAM Conference on Data Mining, 2013.
- K. Tangwongsan, A. Pavan, and S. Tirthapura. Parallel triangle counting in massive streaming graphs. In ACM Conference on Information & Knowledge Management (CIKM), 2013.
- Y. Lim and U Kang, MASCOT: Memory-efficient and Accurate Sampling for Counting Local Triangles in Graph Streams, Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Pages 685-694, KDD '15, 2015.

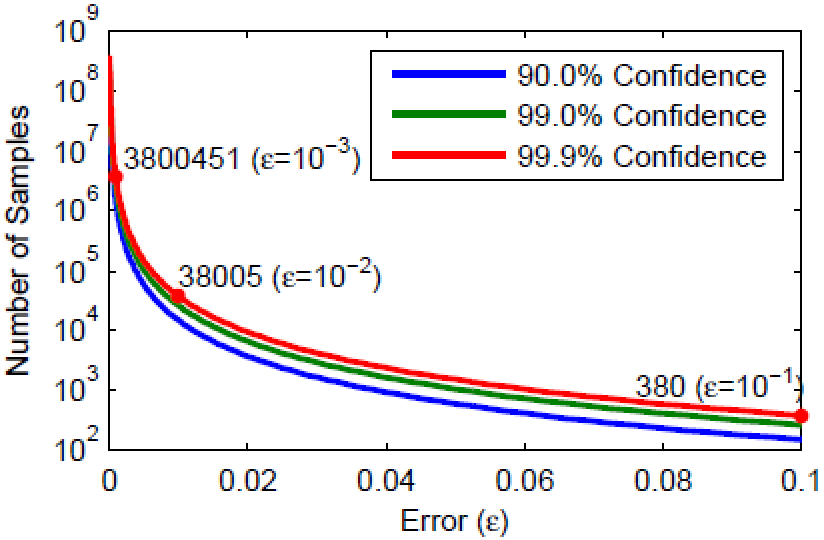
Triangle counting in the streaming setting

- K. J. Ahn, S. Guha, and A. McGregor. Graph sketches: sparsification, spanners, and subgraphs. In Principles of Database Systems, pages 5–14, 2012.
- D. M. Kane, K. Mehlhorn, T. Sauerwald, and H. Sun. Counting arbitrary subgraphs in data streams. In International Colloquium on Automata, Languages, and Programming (ICALP), pages 598–609, 2012.
- M. Jha, C. Seshadhri, and A. Pinar. A space efficient streaming algorithm for triangle counting using the birthday paradox. In Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining, KDD ’13, pages 589–597, New York, NY, USA, 2013. ACM.
- A. Pavan, K. Tangwongsan, S. Tirthapura, and K.-L. Wu. Counting and sampling triangles from a graph stream. In International Conference on Very Large Databases (VLDB), 2013.
- K. Tangwongsan, A. Pavan, and S. Tirthapura. Parallel triangle counting in massive streaming graphs. In ACM Conference on Information & Knowledge Management (CIKM), 2013.
- N. K. Ahmed, N. Duffield, J. Neville, and R. Kompella. Graph sample and hold: A framework for big graph analytics. In Conference on Knowledge Discovery and Data Mining (KDD), 2014.
- A. McGregor, S. Vorotnikova, and H. T. Vu, Better Algorithms for Counting Triangles in Data Streams, The Principles of Database Systems (PODS) PODS’16, June 26-July 01, 2016.

Example: wedge sampling for counting triangles

The term wedge refers to a path of length 2

The mathematical analysis of the method is a direct consequence of standard Chernoff-Hoeffding bounds



This estimate is **independent of the size of the graph**, though the preprocessing required by the method is linear in the number of edges (to obtain the degree distribution)

26492 wedges

amazon0312	401K	2350K	69M	3986K	3722K	0.16205	0.00200	0.41928	0.0020
amazon0505	410K	2439K	73M	3951K	4030K	0.16559	0.00360	0.42013	0.0070
amazon0601	403K	2443K	72M	3987K	4024K	0.16718	0.00100	0.42982	0.0002
as-skitter	1696K	11095K	16022M	28770K	29.4M	0.00551	0.00056	0.29706	0.0010
cit-Patients	3775K	16519K	336M	7515K	7643K	0.06828	0.00100	0.09587	0.0039
roadNet-CA	1965K	2767K	6M	121K	124K	0.06183	0.00180	0.05391	0.0011
web-BerkStan	685K	6649K	27983M	64691K	63.7M	0.00683	0.00017	0.63625	0.0020
web-Google	876K	4322K	727M	13392K	13.7M	0.05647	0.00150	0.61900	0.0050
web-Stanford	282K	1993K	3944M	11329K	11.9M	0.00906	0.00006	0.62888	0.0001
wiki-Talk	2394K	4660K	12594M	9204K	11.3M	0.00268	0.00068	0.20108	0.0001

Approximately Counting Triangles in Sublinear Time

Query access to the graph:

1) Degree queries, in which the algorithm can query the degree d_v of any vertex v.
2) Neighbor queries, in which the algorithm can query what vertex is the i-th neighbor of a vertex v, for any $i \le d_v$.
3) Vertex-pair queries, in which the algorithm can query for any pair of vertices v and u whether (u, v) is an edge.

T. Eden, A. Levi, D. Ron, and C. Seshadhri, Approximately Counting Triangles in Sublinear Time, 56th Annual IEEE Symposium on Foundations of Computer Science, FOCS, 2015, http://arxiv.org/pdf/1504.00954v3.pdf

Discussed algorithms (and others published in various papers) may suggest that many problems have sub-linear algorithms

However, it turns out that these algorithms are more like exceptions than a norm. Indeed, many problems have a trivial lower bound that exclude sub-linear algorithms.

The wiki **sublinear.info** collates open problems in the field of sub-linear algorithms: data stream algorithms, property testing, and communication complexity (sub-linear communication).

http://sublinear.info/index.php?title=Main_Page

Challenge 3: Scalable graph algorithms

How to address problems that are intractable when the available space is sub-linear in the number of nodes?

(1) To develop heuristic methods for design of sub-linear algorithms for which approximation will not be guaranteed and obtained solutions will not be optimal
(2) To consider linear-time algorithms, linearithmic-time algorithms ($O(n \log n)$) and/or sub-quadratic-time algorithms ($o(n^2)$)
(3) To combine heuristic approaches, graph sampling methods, and/or approximation for graph properties/characteristics

Example: shortest paths

The shortest-path query problem is different from the classical single-source and all-pairs shortest paths problems in that there are two stages: preprocessing and answering queries.

Scaling point-to-point path queries to large scale social networks is challenging for two reasons.

Latency: it is desirable to answer queries within milliseconds

Storing paths between each pair of users is infeasible due to memory limitations; even for a social network with 3 million users, this would require roughly 4.5×10^{12} entries.

Point-to-point approximate shortest-path query problem:

- preprocessing algorithm may compute certain information
- shortest-path queries (answered as fast as possible)

LiveJournal social network (5 million nodes, 69 million edges):

- Answer more than 99.9% of the queries by exploring less than 0.2% of the entire network
- Each query can be answered in roughly 365 microseconds

Christian Sommer, Shortest-path queries in static networks, ACM Computing Surveys (CSUR), Volume 46 Issue 4, April 2014

Example: Wedge sampling with approximation of degree distribution

amazon0312	0.16569	0.00569
amazon0505	0.15894	0.00306
amazon0601	0.15668	0.00932
as-skitter	0.00504	0.00004
cit-Patients	0.07109	0.00409
roadNet-CA	0.05228	0.00772
web-BerkStan	0.00654	0.00046
web-Google	0.05325	0.00175
web-Stanford	0.00826	0.00074
wiki-Talk	0.00218	0.00018

Challenge 4: Local versus global metrics for massive graphs

(1) algorithms for computing only those graph properties that can be estimated only locally from a given node
(2) distributed algorithms with guaranteed approximation where each node, from a subset of nodes, can access only given small portions of the input network, and
(3) heuristic methods for designing efficient algorithms with provably good practical performance.

Example: largest eigenvalue of the adjacency matrix

By using algebraic graph theory and convex optimization, Preciado and Jadbabaie proved that the largest eigenvalue of the adjacency matrix can be approximated in terms of the number of nodes, edges, and triangles.

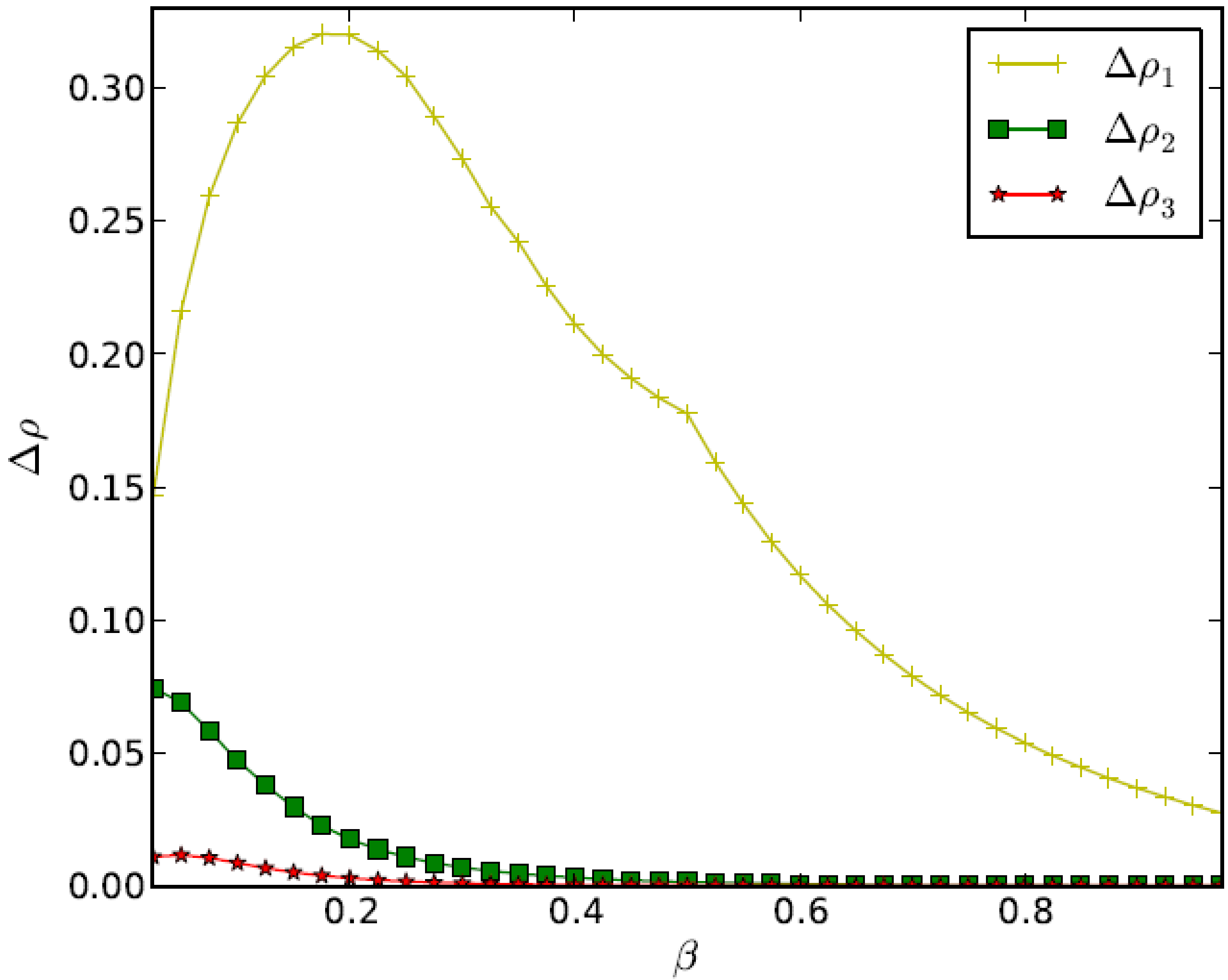
M. Preciado, and A. Jadbabaie, “Moment-Based Spectral Analysis of Large-Scale Networks Using Local Structural Information”, IEEE/ACM Transactions on Networking, Vol. 21, Issue 2, 373 - 382, 2013

Example: SIS epidemic spreading processes

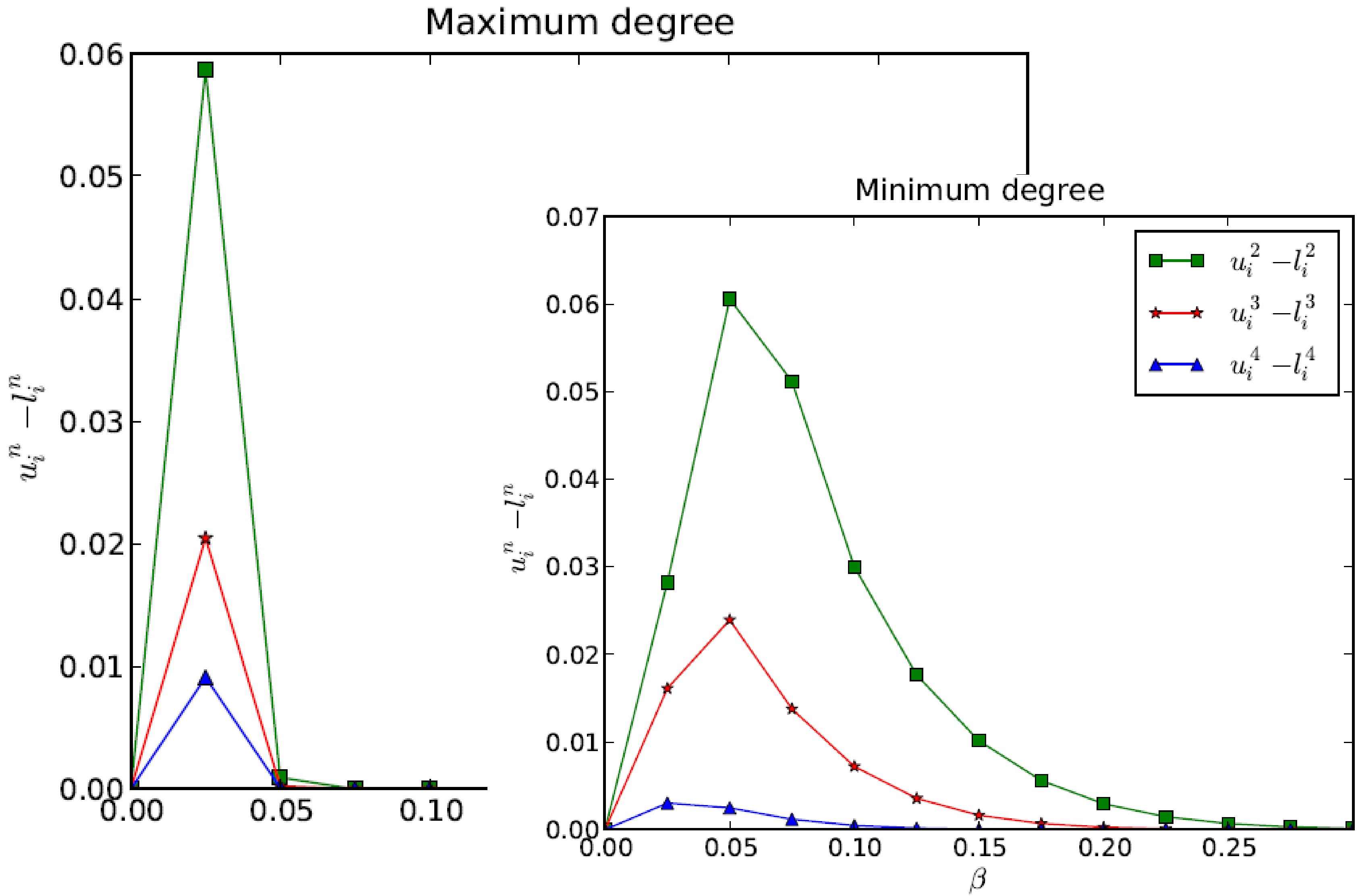
Upper and lower bounds on the probability that a node is infective can be estimated using only local information (considering only *n*-hop local topology, for small *n*), without knowing the whole network.

The results are valid to other ergodic models (such as SIRS, for example) and are related to all types of spreading (idea, failure, rumor).

D. Smilkov and L. Kocarev, “Influence of the network topology on epidemic spreading”, Physical Review E 85, 016114 (10 pages), 2012



$\Delta\rho_1$
$\Delta\rho_2$
$\Delta\rho_3$
0.30
0.25
0.20
0.15
0.10
0.05
0.00
$\Delta\rho$
0.2
0.4
0.6
0.8
β



Maximum degree
Minimum degree
$u_i^n - l_i^n$
$u_i^2 - l_i^2$
$u_i^3 - l_i^3$
$u_i^4 - l_i^4$
β
0.00
0.01
0.02
0.03
0.04
0.05
0.06
0.07
0.10
0.15
0.20
0.25

Conclusions

- Massive networks are expected to become the cornerstone of our societies and modern business

- Challenges:
  - Computational models for massive graphs
  - Sub-linear graph algorithms
  - Scalable graph algorithms
  - Local versus global metrics for massive graphs

In the future both the cyber world and our physical world will be integrated effectively blurring the gap between the two.

Humans are interconnected and beyond that all objects of our environment are becoming networked forming huge ***global village***.

“Time has ceased, space has vanished. We now live in a global village . . a ***simultaneous happening***.” McLuhan, Medium is the Massage, 1967.

However, there is a great ***challenge*** in front of us. The movie Star Trek: First Contact (1996) illustrates this challenge with particular force. In this film, mankind is threatened by the Borg Collective – civilization with a group mind.

The Borg exhibited no hierarchical command structure, instead using a structure similar in principle to the ***internet*** with no control center and distributed processing.

The ship's Captain, Jean Luc Picard, is "assimilated" by the Borg as is the rest of Picard's crew.

There is one character, however, who remains to be assimilated: the good android, ***Data***. For a while, it appears that Data is seduced by the pleasure of feeling truly alive. In the end, Data comes though, rejecting the Borg Queen and saving Captain Picard and his crew.

Data's choice shows him to be the most human of all, in spite of losing his chance to be, at least in part, corporeally human.