Generating Graphs with Predefined k-Core Structure

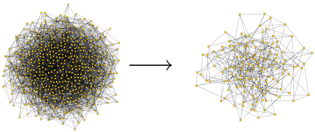
Michael Baur, Marco Gaertler, Robert Görke,
Marcus Krug, and Dorothea Wagner

Fakultät für Informatik
Universität Karlsruhe (TH)

European Conference on Complex Systems

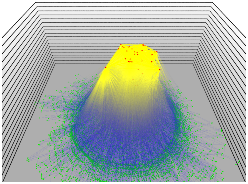
Motivation

cores are often used in network analysis / visualization



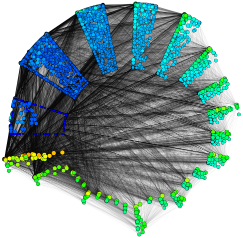
as reduction

Gkantsidis et al Infocom03



drawing Internet

Baur et al. GD04



visual analysis

Görke et al. GD07

Curiosity

The core structure of many (real) networks is well understood and used for analysis.

Yet:

No generators for graphs with predefined core structure exist.

Overview

Structure

Basics

concept of *cores*:

- introduced by Seidmann (1983)
- generalized by Batagelj et al (2002)

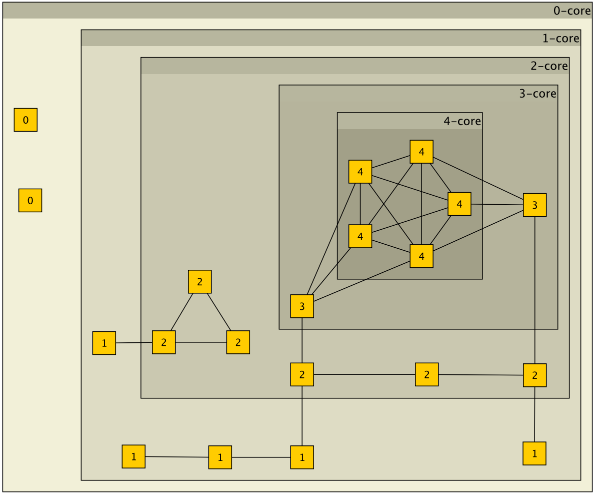
Definition

Given a graph $G = (V, E)$, the k-*core* is the maximum subgraph H such that each node has at least degree k in H.

Algorithmic Description

Given a graph $G = (V, E)$, the k-*core* is obtained by iteratively removing all nodes of degree less than k.

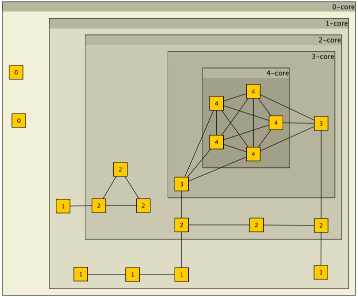
Example



Notation

- $G_k =: (V_k, E_k)$ denotes k-core of G
- core-number of G is maximum k such that the k-core is not empty
- coreness of a node v is index i such that
  $v \in V_i$ and $v \notin V_{i+1}$
- k-shell V'_k is the set of all nodes of coreness k

Basic Properties



- degree is an upper bound for coreness
- cores are nested, i. e., $V_{i+1} \subseteq V_i$
- k-core has at least $k+1$ nodes
- k-core/-shell needs not be connected

Properties: Size

Let G be a graph with core-number k,
$n_i := |V_i \setminus V_{i+1}|$ number of nodes with coreness i and
$m_i := |E_i \setminus E_{i+1}|$ number of edges induced by the i-shell. Then:

$$0 \leq n_i \leq |V|$$

$$\left.\begin{matrix} \left\lceil \frac{i \cdot n_i}{2} \right\rceil & , \text{if } n_i > i \\ \binom{n_i}{2} + n_i \cdot (i - n_i + 1) & , \text{if } n_i \leq i \end{matrix}\right\} \leq m_i \leq \begin{cases} i \cdot n_i & , \text{if } i < k \\ i \cdot n_i - \frac{i^2 + i}{2} & , \text{if } i = k \end{cases}$$

Properties: Size

Let G be a graph with core-number k,
$n_i := |V_i \setminus V_{i+1}|$ number of nodes with coreness i and
$m_i := |E_i \setminus E_{i+1}|$ number of edges induced by the i-shell. Then:

$$0 \leq n_i \leq |V|$$

$$\left.\begin{array}{ll} \left\lceil \frac{i \cdot n_i}{2} \right\rceil & \text{, if } n_i > i \\ \binom{n_i}{2} + n_i \cdot (i - n_i + 1) & \text{, if } n_i \leq i \end{array}\right\} \leq m_i \leq \begin{cases} i \cdot n_i & \text{, if } i < k \\ i \cdot n_i - \frac{i^2 + i}{2} & \text{, if } i = k \end{cases}$$

take-away message: tight bounds exists

Short Summary

- mathematical and algorithmic description
- basic properties such as:
  - (density) hierarchy
  - connectivity
  - tight bound on sizes

Structure

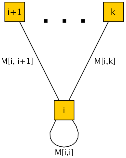
Input Parameter

Question: How to specify a core structure?

Answer: Sizes of shells/cores

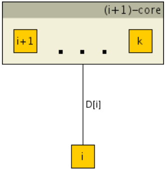
- vector of number of nodes for each shell
- distribution of edges via:

shell-connectivity matrix



$$\leftrightsquigarrow \begin{pmatrix} 3 & 1 & 0 & 0 \\ 1 & 2 & 2 & 0 \\ 0 & 2 & 0 & 6 \\ 0 & 0 & 6 & 10 \end{pmatrix}$$

shell-density vector



$$\leftrightsquigarrow \begin{pmatrix} 4 & 4 & 6 & 10 \end{pmatrix}$$

Advantages/Disadvantages

shell-connectivity matrix	shell-density vector
+ fixes many degrees of freedom + simple test for consistency – quadratic number of input data	+ easy to specify (in absolute or relative terms) + simple test for consistency + linear number of input data

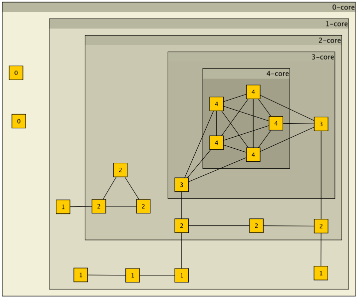
+ our generation technique can handle both types

Observation

Lemma

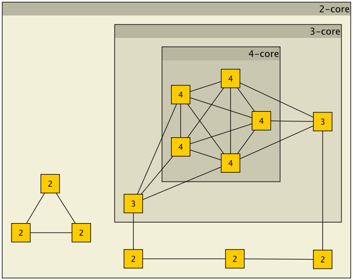
The structure of the *i*-core is independent of lower shells.

Example:



removing

0-, 1-shell



Founding Idea

incremental construction:

- starting with the core
- adding lower shells one by one

correctness:

- via algorithmic description
- needed invariant: added nodes have to have given coreness

Realization

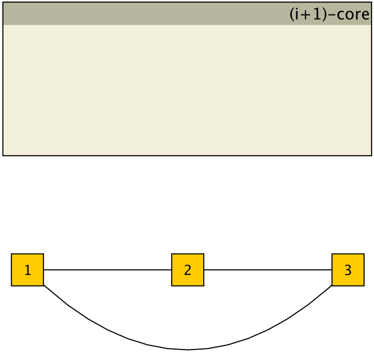
Given $G_{k+1} = (V_{k+1}, E_{k+1})$ a $(k+1)$-core, n_k number of nodes in the k-shell and $M[k, \cdot]$ number of edges starting in the k-shell and going to nodes with coreness greater or equal k.

- create n_k new nodes and store them in V'_k
- $E'_k \leftarrow \emptyset$
- for $\ell \leftarrow k, \ldots$ do
  - while $M[k, \ell] > 0$ do
    - select a node v in V'_k with degree less than k
    - if $\ell = k$, then select a node w in V'_k,
      otherwise select a node w in V_{k+1} with coreness ℓ
    - if $\{v, w\} \notin E'_k$ then
      $E'_k \leftarrow E'_k \cup \{\{v, w\}\}$ and $M[k, \ell] \leftarrow M[k, \ell] - 1$

Problems?

- Is it always possible to select a node of degree less than k while $M[k,\ell] > 0$?

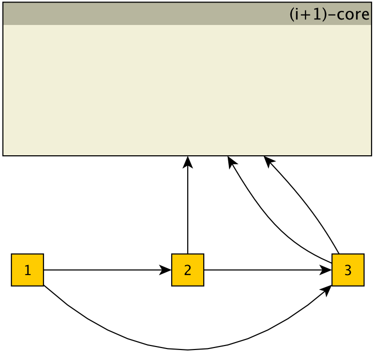
Problems?



M[2,2]=3, M[2, i+1]=3

- Is it always possible to select a node of degree less than k while $M[k,\ell] > 0$?
  No!

Problems?



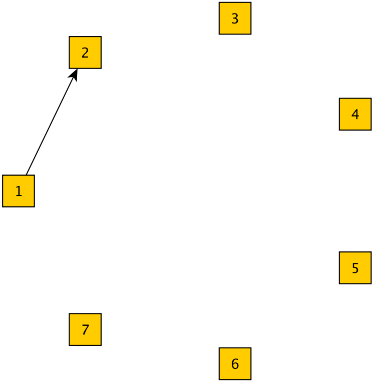
M[2,2]=3, M[2, i+1]=3

- Is it always possible to select a node of degree less than k while $M[k, \ell] > 0$?
  No!

Problems?

- Is it always possible to select a node of degree less than k while $M[k,\ell] > 0$?
  No!
- Has every node $v \in V_k$ coreness k, if $M[k,\ell] = 0$ holds for every $\ell \geq k$?

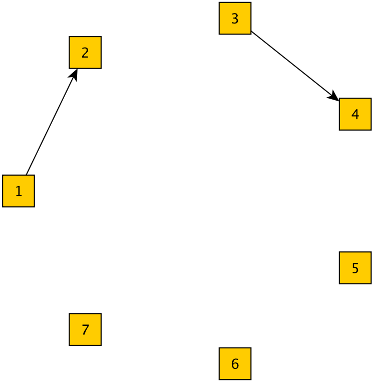
Problems?



M[3,3]=11

- Is it always possible to select a node of degree less than k while $M[k, \ell] > 0$?
  No!
- Has every node $v \in V_k$ coreness k, if $M[k, \ell] = 0$ holds for every $\ell \geq k$?
  No!

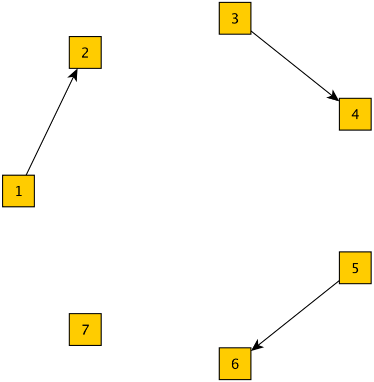
Problems?



M[3,3]=11

- Is it always possible to select a node of degree less than k while $M[k, \ell] > 0$?
  No!
- Has every node $v \in V_k$ coreness k, if $M[k, \ell] = 0$ holds for every $\ell \geq k$?
  No!

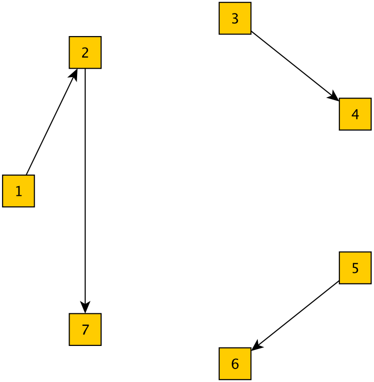
Problems?



M[3,3]=11

- Is it always possible to select a node of degree less than k while $M[k, \ell] > 0$?
  No!
- Has every node $v \in V_k$ coreness k, if $M[k, \ell] = 0$ holds for every $\ell \geq k$?
  No!

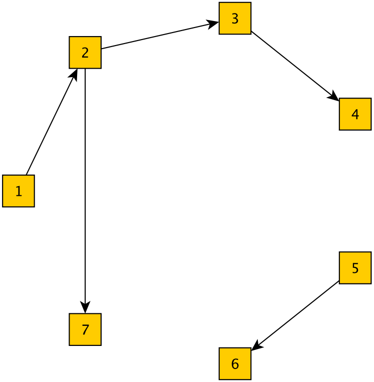
Problems?



M[3,3]=11

- Is it always possible to select a node of degree less than k while $M[k,\ell] > 0$?
  No!
- Has every node $v \in V_k$ coreness k, if $M[k,\ell] = 0$ holds for every $\ell \geq k$?
  No!

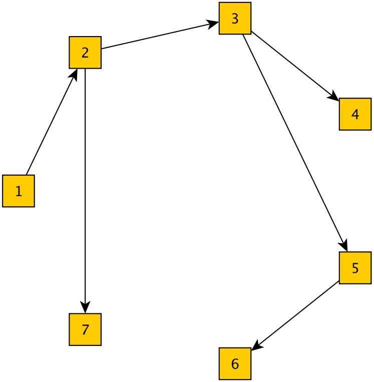
Problems?



M[3,3]=11

- Is it always possible to select a node of degree less than k while $M[k,\ell] > 0$?
  No!
- Has every node $v \in V_k$ coreness k, if $M[k,\ell] = 0$ holds for every $\ell \geq k$?
  No!

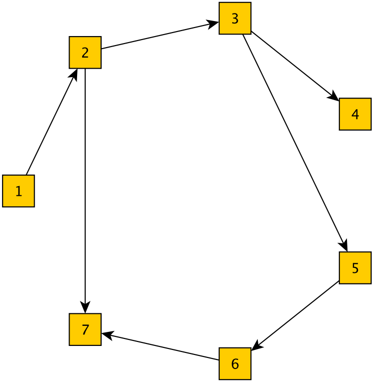
Problems?



M[3,3]=11

- Is it always possible to select a node of degree less than k while $M[k,\ell] > 0$?
  No!
- Has every node $v \in V_k$ coreness k, if $M[k,\ell] = 0$ holds for every $\ell \geq k$?
  No!

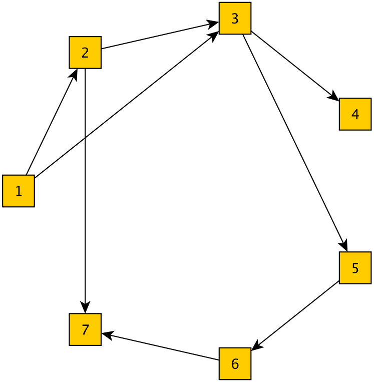
Problems?



M[3,3]=11

- Is it always possible to select a node of degree less than k while $M[k,\ell] > 0$?
  No!
- Has every node $v \in V_k$ coreness k, if $M[k,\ell] = 0$ holds for every $\ell \geq k$?
  No!

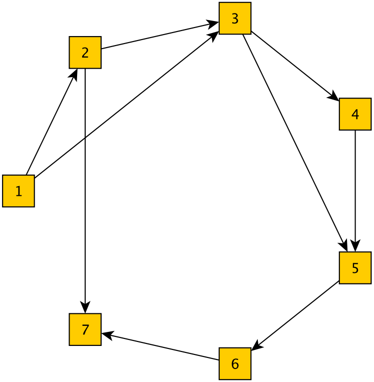
Problems?



M[3,3]=11

- Is it always possible to select a node of degree less than k while $M[k,\ell] > 0$?
  No!
- Has every node $v \in V_k$ coreness k, if $M[k,\ell] = 0$ holds for every $\ell \geq k$?
  No!

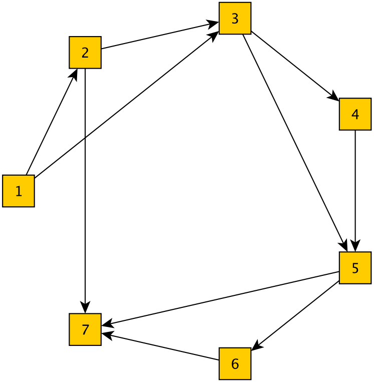
Problems?



M[3,3]=11

- Is it always possible to select a node of degree less than k while $M[k,\ell] > 0$?
  No!
- Has every node $v \in V_k$ coreness k, if $M[k,\ell] = 0$ holds for every $\ell \geq k$?
  No!

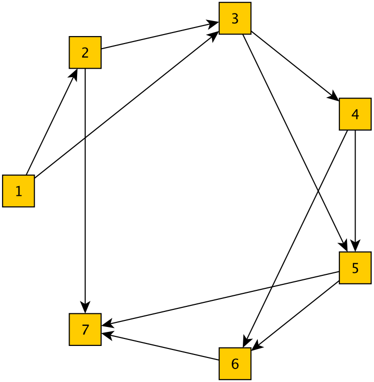
Problems?



M[3,3]=11

- Is it always possible to select a node of degree less than k while $M[k,\ell] > 0$?
  No!
- Has every node $v \in V_k$ coreness k, if $M[k,\ell] = 0$ holds for every $\ell \geq k$?
  No!

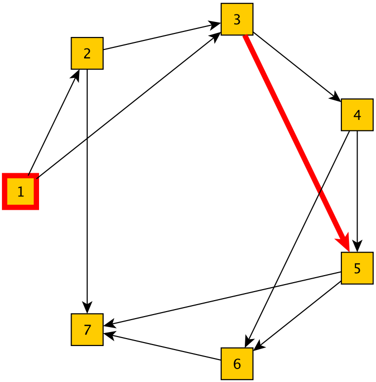
Problems?



M[3,3]=11

- Is it always possible to select a node of degree less than k while $M[k,\ell] > 0$?
  No!
- Has every node $v \in V_k$ coreness k, if $M[k,\ell] = 0$ holds for every $\ell \geq k$?
  No!

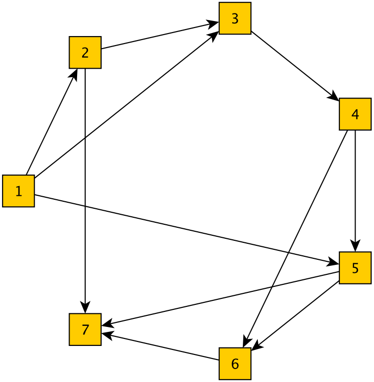
Problems?



M[3,3]=11

- Is it always possible to select a node of degree less than k while $M[k, \ell] > 0$?
  No!
- Has every node $v \in V_k$ coreness k, if $M[k, \ell] = 0$ holds for every $\ell \geq k$?
  No!

Problems?



M[3,3]=11

- Is it always possible to select a node of degree less than k while $M[k,\ell] > 0$?
  No!
- Has every node $v \in V_k$ coreness k, if $M[k,\ell] = 0$ holds for every $\ell \geq k$?
  No!

Solutions:

- Selection of nodes: introduction of removal order invariants:
  - intra-shell edges always start a lower order nodes
  - intra-shell edges always end a higher order nodes
  - every node (with coreness k) has at most k out-going edges
- coreness k: rewiring of edges after all are inserted

Short Summary

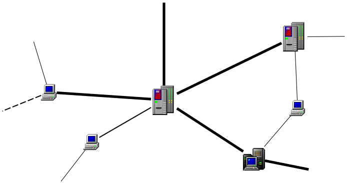
- possible input parameters for k-core structure
- general description of our generator
- problems with naive implementation
- solutions to these problems

Structure

Definition of AS Network

Autonomous System (AS)

collection of routers under same administrative authority presenting homogenous (routing-)policies to the Internet



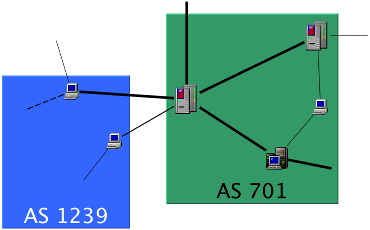
notation:

- node = AS
- edge = traffic exchange agreements

Definition of AS Network

Autonomous System (AS)

collection of routers under same administrative authority presenting homogenous (routing-)policies to the Internet



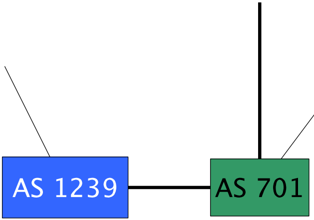
notation:

- node = AS
- edge = traffic exchange agreements

Definition of AS Network

Autonomous System (AS)

collection of routers under same administrative authority presenting homogenous (routing-)policies to the Internet

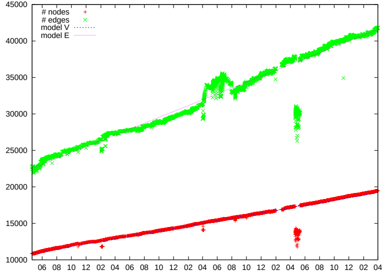


notation:

- node = AS
- edge = traffic exchange agreements

Basic Information

- many power-laws (degree vs. rank, . . .)
- significant *k*-core hierarchy
- snapshots are provided by (Oregon) Routeviews project



	# nodes	# edges
Jan 2002	12,485	25,980
Jan 2006	21,419	45,638
Jul 2007	25,787	53,014

Available Generators

Inet 3.0

- *“random networks with characteristics similar to those of the Internet from Nov 1997 to Feb 2002, and beyond”*
- fits degree distribution to power-law of Faloutsos
- optimizations for various specific conditions

BRITE

- *Boston university Representative Internet Topology gEnerator*
- geometric node placement
- fixed out-degree, connect by locality or preferential attachment

Test Settings

tests

- timestamps 2002, 2006, and 2007
- real AS and generators Inet, Brite, and Core
- evaluation based on selection of characteristics

Inet

- number of nodes, random seed, defaults

BRITE

- number of nodes, incremental growth
- out-degree 2, preferential attachment

Important Characteristics

general: # edges, minimum and maximum degree

cores: core number, core fingerprint

clustering coefficient:
fraction of connected pairs of neighbors

path length: average total length, average eccentricity

frequency vs. degree:
frequency of nodes with degree $\geq d$

k-neighboorhood size:
neighborhood sizes within distance k

Characteristics

	AS 2007-07	Core	BRITE	Inet
# nodes	25,787	25,787	25,787	25,787

Characteristics

	AS 2007-07	Core	BRITE	Inet
# nodes	25,787	25,787	25,787	25,787
# edges	53,014	53,014	51,571	76,467

Characteristics

	AS 2007-07	Core	BRITE	Inet
# nodes	25,787	25,787	25,787	25,787
# edges	53,014	53,014	51,571	76,467
min. degree	1	1	2	1
max. degree	2,391	838	393	5,168
core number	22	22	2	26

Characteristics

	AS 2007-07	Core	BRITE	Inet
# nodes	25,787	25,787	25,787	25,787
# edges	53,014	53,014	51,571	76,467
min. degree	1	1	2	1
max. degree	2,391	838	393	5,168
core number	22	22	2	26
# triples	13,889,150	6,759,443	757,653	56,514,215
# triangles	39,646	29,612	174	162,889
transitivity	0.009	0.013	0.001	0.009
clustering coeff.	0.33	0.15	0.00	0.65

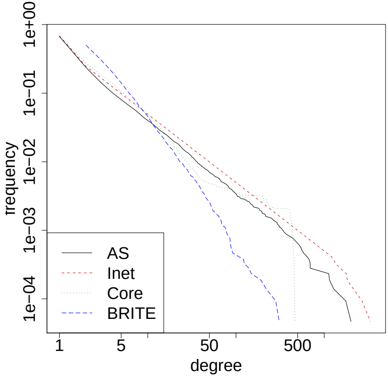
Characteristics

	AS 2007-07	Core	BRITE	Inet
# nodes	25,787	25,787	25,787	25,787
# edges	53,014	53,014	51,571	76,467
min. degree	1	1	2	1
max. degree	2,391	838	393	5,168
core number	22	22	2	26
# triples	13,889,150	6,759,443	757,653	56,514,215
# triangles	39,646	29,612	174	162,889
transitivity	0.009	0.013	0.001	0.009
clustering coeff.	0.33	0.15	0.00	0.65
avg. path length	3.89	3.92	5.39	2.99
avg. eccentricity	10.24	10.64	8.72	6.52

Characteristics – Summary

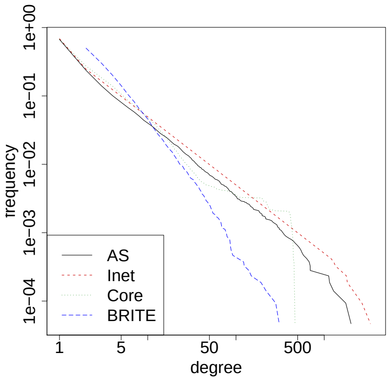
- # nodes, # edges well matches
- variations in degree / coreness indicate structural differences
- local density is fairly diverse
- path lengths are quite similar
- $\Longrightarrow$ simple preferential attachment is not sufficient

Distance Fingerprint



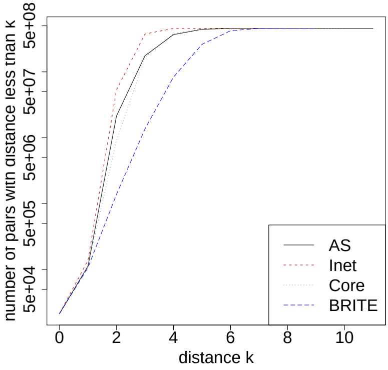
- degree distribution

Distance Fingerprint



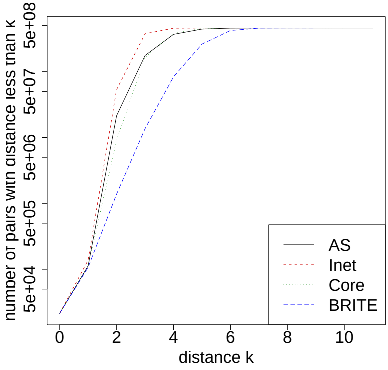
- degree distribution
  - INET fairly well matches
  - Core oscillates (note: no PA)

Distance Fingerprint



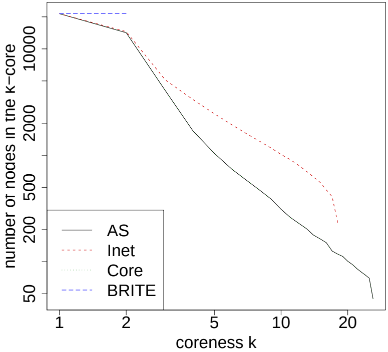
- degree distribution
  - INET fairly well matches
  - Core oscillates (note: no PA)
- hop distribution

Distance Fingerprint



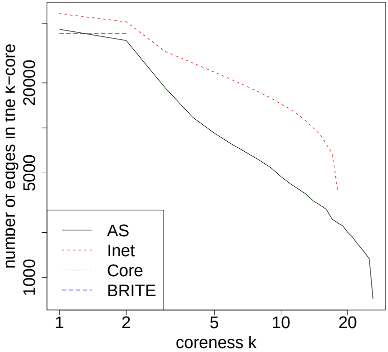
- degree distribution
  - INET fairly well matches
  - Core oscillates (note: no PA)
- hop distribution
  - INET/Core match
  - BRITE generally underestimates

Core Fingerprint



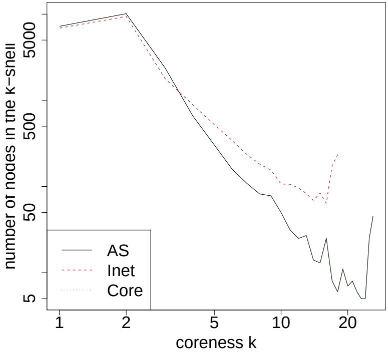
- *k*-core distr. (# nodes)

Core Fingerprint



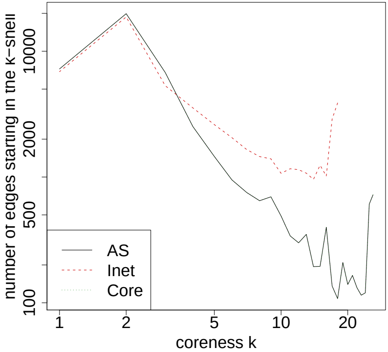
- k-core distr. (# nodes)
- k-core distr. (# edges)

Core Fingerprint



- *k*-core distr. (# nodes)
- *k*-core distr. (# edges)
- *k*-shell distr. (# nodes)

Core Fingerprint



- *k*-core distr. (# nodes)
- *k*-core distr. (# edges)
- *k*-shell distr. (# nodes)
- *k*-shell distr. (# edges)

Short Summary

- macroscopic view: all generated graphs are good
- microscopic view: mismatches in details
  examples: local density
- pure preferential attachment mechanism is not enough
- core structure seems to fix more degree of freedoms than degree distribution (for AS Network)

Conclusion

Summary:

- basic introduction to *k*-cores
- a flexible generator for given core structures
- experimental evaluation based on AS Networks

Future Work:

- further evaluation with different networks
- adaption to include other features, such as power-law in degrees

Conclusion

Summary:

- basic introduction to *k*-cores
- a flexible generator for given core structures
- experimental evaluation based on AS Networks

Future Work:

- further evaluation with different networks
- adaption to include other features, such as power-law in degrees

Thanks for your attention!