HAL
open science

Understanding edge-connectivity in the Internet through core-decomposition

José Ignacio Alvarez-Hamelin, Beiró Mariano Gastón, Jorge Rodolfo Busch

To cite this version:

José Ignacio Alvarez-Hamelin, Beiró Mariano Gastón, Jorge Rodolfo Busch. Understanding edge-connectivity in the Internet through core-decomposition. Internet Mathematics, 2011, 7 (1), pp.45-66. 10.1080/15427951.2011.560786 . inria-00439597

HAL Id: inria-00439597

https://inria.hal.science/inria-00439597

Submitted on 8 Dec 2009

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

UNDERSTANDING EDGE-CONNECTIVITY IN THE INTERNET THROUGH CORE-DECOMPOSITION

J. IGNACIO ALVAREZ-HAMELIN, MARIANO G. BEIRÓ, AND JORGE R. BUSCH

Abstract. Internet is a complex network composed by several networks: the Autonomous Systems, each one designed to transport information efficiently. Routing protocols aim to find paths between nodes whenever it is possible (i.e., the network is not partitioned), or to find paths verifying specific constraints (e.g., a certain QoS is required). As connectivity is a measure related to both of them (partitions and selected paths) this work provides a formal lower bound to it based on core-decomposition, under certain conditions, and low complexity algorithms to find it. We apply them to analyze maps obtained from the prominent Internet mapping projects, using the LaNet-vi open-source software for its visualization.

1. Introduction

Nowadays, Internet is a highly developed network connecting people around the world. Its continuous growth raises new problems, challenging us to find novel and creative solutions. Modeling the Internet is a problem as relevant as difficult: A tight model of the Internet would provide us with a powerful tool to analyze applications behavior or test routing algorithms during their design. In the beginnings of the Internet [Waxman, 1988] was the first to introduce an Internet model to test his multicast protocol. Then other authors introduced models trying to improve Waxman's one adding hierarchy and other sophistications (e.g., see [Zegura et al., 1997]). But the [Faloutsos et al., 1999] paper was the first to show a remarkable characteristic of the Internet's topology: the degree distribution of the nodes follows a heavy-tailed behavior, instead of a Poisson's one as was believed (i.e. this was the case of the forementioned models which are a variant of the Random Graph [Erdös and Rényi, 1959]). The unexpected consequence of this is that we have nodes with a large variety of degrees.

From this discovery, new simple models aroused which tried to reproduce heavy-tailed behavior, such as the [Barabási and Albert, 1999] model based on preferential attachment, and later the [Fabrikant et al., 2002] model proving that a multivariate optimization problem may lead to heavy-tailed degree distributions.

Later on, other properties were highlighted as important ones, like the average neighboring degree distribution by [Pastor-Satorras et al., 2001]. This work showed that the Internet topology at the Autonomous System level (AS) is different from the Internet topology at the inter-router level (IR). On this basis, we think that

Facultad de Ingeniería, Universidad de Buenos Aires, Paseo Colón 850, C1063ACV Buenos Aires – Argentina

E-mail address: ignacio.alvarez-hamelin@cnet.fi.uba.ar,{mbeiro,jbusch}@fi.uba.ar.

Key words and phrases. connectivity, core-decomposition, algorithm, visualization.

a better understanding and modeling of the Internet's nature may be achieved by discovering new significant properties.

In this paper we turn our attention to connectivity in graphs, i.e., the number of different independent paths between their nodes. It is quite relevant as it represents the robustness of the network: how tolerant it is to link failures, or what QoS it may offer when needed (e.g, see the RFCs of MultiProtocol Label Switching [Rosen et al., 2001, Andersson and Asati, 2009]). Also with regard to QoS, forwarding packets to more connected nodes helps to find paths that fulfill the needed QoS, therefore routing protocols like BGP [Rekhter et al., 2006] might take advantage of information about connectivity. Here we take different subgraphs (i.e., portions of the topology) and analyze each one's connectivity.

In particular, we introduce the concept of core-connected graphs. These graphs have the nice property that the minimum shell-index is a lower bound for the connectivity between two nodes. As graphs are not core-connected in general, we also present algorithms to find core-connected induced subgraphs. These algorithms are based on a generalization of Plesník's theorem [Plesník, 1975].

Finally, it has been shown previously that the k-core decomposition is capable of identifying networks sources by means of the visualization [Alvarez-Hamelin et al., 2006, Beiró et al., 2008], may be used to validate models [Serrano et al., 2006], and discover exploration biases on the Internet [Alvarez-Hamelin et al., 2008]. These facts support the k-core analysis as one of the relevant tools to describe the Internet topology maps.

The paper is organized as follows. Section 2 is devoted to formalize the relation between k-cores and k-connected subgraphs. Then in Section 3 we present a tool that implements these ideas giving a visualization of the network according to its connectivity, and also a data file containing the connectivity bound for every pair of nodes. We show some applications to Internet maps. Finally, we conclude with some important remarks and comments on future work.

2. Mathematics

Let us introduce some general graph notions and notation. Let G be a simple graph (*i.e.* a graph with no loops, no multiple edges) with vertex set $V(G)$ and edge set $E(G)$ (we follow in notation the book [West, 2001]). Given $A, B \subset V(G)$, $[A, B]$ is the set of edges of the form ab, joining a vertex $a \in A$ to a vertex $b \in B$. As we consider edges without orientation, $[A, B] = [B, A]$. Abusing of notation, for $v \in V(G), A \subset V(G)$, we write $[v, A]$ instead of $[\{v\}, A]$. The *degree* of a vertex $v \in V(G)$ is $d_G(v) \doteq |[v, V(G)]|$. We shall denote $n(G) \doteq |V|, e(G) \doteq |E|, \delta(G) \doteq \min_{v \in V} d_G(v), \Delta(G) \doteq \max_{v \in V} d_G(v)$.The *neighborhood* of a vertex v, $N(v)$, is the set of vertexes w such that $vw \in E(G)$. Given $A \subset V(G)$, $G(A)$ is the graph G' such that $V(G') = A$ and $E(G')$ is the set of edges in $E(G)$ having both endpoints in A. Given $v, w \in V(G)$, $\rho_G(v, w)$ is the distance in G from v to w, that is the minimum length of a path from v to w. If $v \in V(G), A \subset V(G)$ we set $\rho_G(v, A) \doteq \min_{w \in A} \rho_G(v, w)$. We shall also use the notation

$$\rho_A \doteq \max_{a,b \in A} \rho_{G(A)}(a, b)$$

for the diameter of $G(A)$.

We introduce some notions of connectivity used along the paper. An edge cut in G is a set of edges $[S, \bar{S}]$, where $S \subset V(G)$ and $\bar{S} \doteq V(G) \setminus S$ are non void.

The edge-connectivity of G, $k'(G)$, is the minimum cardinal of the cuts in G. We say that G is k-edge-connected if $k'(G) \geq k$.

Menger's theorem has as a consequence that, if G is k-edge-connected, given two vertices v, w in $V(G)$ there are at least k-edge-disjoint paths joining v to w (see [West, 2001], pp.153-169).

As we shall not deal in this work with vertex-disjoint paths, in the sequel we shall speak of k-connectivity, avoiding the reference to the edges.

2.1. **Cores decomposition.** Let, for $A \subset V$,

$$\psi(A) = \min_{v \in A} d_{G(A)}(v)$$

Notice that for $A_1, A_2 \subset V$,

$$\psi(A_1 \cup A_2) \geq \min(\psi(A_1), \psi(A_2))$$

and as a consequence for any k

$$C_k = \cup\{A : \psi(A) \geq k\}$$

satisfies $\psi(C_k) \geq k$, and of course C_k contains any other set that satisfies this property.

The preceding remark justifies the following [Batagelj and Zaversnik, 2002]

Definition 1. *A subgraph $H = G(C_k)$ induced by the set $C_k \subseteq V$ is the k-*core *(or the core of order k) in G if C_k is the maximal subset of V such that $\min_{v \in C} d_{G(C)}(v) \geq k$.*

Thus, if we let $k_{max} \doteq \max\{k : C_k \neq \emptyset\}$, we obtain the decomposition

$$V = \cup\{C_k : 0 \leq k \leq k_{\max}\}$$

where $C_{k+1} \subset C_k, 0 \leq k \leq k_{\max} - 1$.

A k-core of G can be obtained by recursively removing all the vertices of degree lower than k, with their incident edges, until all the vertices in the remaining graph have degree greater than or equal to k. This decomposition can be easily implemented: the algorithm by [Batagelj and Zaversnik, 2003] has a time complexity of order $O(n(G) + e(G))$ for a general simple graph G. This makes the algorithm very efficient for sparse graphs, where $e(G)$ is of the same order that $n(G)$.

Definition 2. *Let $S_k \doteq C_k \setminus C_{k+1}, 0 \leq k \leq k_{\max} - 1$, $S_{k_{\max}} = C_{k_{\max}}$. We call S_k the k-shell of G, and if $v \in S_k$ we say that v has* shell-index *k, $sh(v) = k$.*

Notice that C_k is thus the union of all shells S_s with $s \geq k$, and that the shells are pairwise disjoint.

Definition 3. *Every connected component of S_s will be called* cluster.

Each shell S_s is thus composed by clusters Q_s^m, such that $S_s = \cup\{Q_s^m : 1 \leq m \leq q_{\max}(s)\}$, where $q_{\max}(s)$ is the number of clusters in S_s.

In this paper we address the following expansion problem: given a k-edge-connected graph G_2, give conditions under which the result of adjoining to G_2 a graph G_1 will be also k edge-connected (see Corollary 2 below).

2.2. **k-connectivity.** We may analyze connectivity in a strict or wide sense.

Definition 4. *Let $A \subset V(G)$.*

(1) *We say that A is k-connected in strict sense if $G(A)$ is k-connected,* i.e *every cut in $G(A)$ has at least k edges. That is, given $u, v \in A$, there exist at least k edge disjoint paths form u to v in $G(A)$.*
(2) *We say that A is k-connected in wide sense if every cut $[X, \bar{X}]$ in G such that $X \cap A \neq \emptyset$ and $\bar{X} \cap A \neq \emptyset$ has at least k edges. That is, given $u, v \in A$, there exist at least k edge disjoint paths form u to v in G.*

Of course, if A is k-connected in strict sense, it is also k-connected in wide sense.

Lemma 1. *Let $A, B \subset V$ and $A \cap B \neq \emptyset$. Then*

(1) *If A and B are k-connected in strict sense, so is $A \cup B$.*
(2) *If A and B are k-connected in wide sense, so is $A \cup B$.*

Proof.

(1) Let $u, v \in A \cup B$. If $u, v \in A$ or $u, v \in B$, then there are k disjoint paths in $G(A)$ or in $G(B)$ from u to v, and in any case there are k disjoint paths in $G(A \cup B)$. Suppose then that $u \in A$, $v \in B$, and let $[X, \bar{X}]$ be a cut in $G(A \cup B)$ with $u \in X, v \in \bar{X}$. Suppose that $s \in A \cap B$ and, without loss of generality, assume that $s \in X$. Then there are k disjoint paths in $G(B)$ from s to v, whence $|[X, \bar{X}]| \geq k$.
(2) Let $u, v \in A \cup B$. If $u, v \in A$ or $u, v \in B$, then there are k disjoint paths in G from u to v. Suppose then that $u \in A$, $v \in B$, and let $[X, \bar{X}]$ be a cut with $u \in X, v \in \bar{X}$. Suppose that $s \in A \cap B$ and, without loss of generality, assume that $s \in X$. Then there are k disjoint paths in G from s to v, whence $|[X, \bar{X}]| \geq k$.

□

This lemma has as a consequence that given $v \in V$,

$$\cup\{A : A \text{ is } k\text{-connected in strict sense and } v \in A\}$$

is k connected in strict sense (and we call it the k-connected component of v in strict sense), and that

$$\cup\{A : A \text{ is } k\text{-connected in wide sense and } v \in A\}$$

is k connected in wide sense (and we call it the k-connected component of v in wide sense).

Lemma 2. *For $k = 1, 2$ the k-connected components in strict sense and in wide sense are the same. This is not true for $k \geq 3$.*

2.3. **An expansion theorem.** Let G be a simple graph. Let $Q, C \subset V(G)$, and set $C' \doteq Q \cup C$, $G' \doteq G(C')$. We assume in the sequel that Q and C are non void and that $Q \cap C = \emptyset$. We define, for $x, y \in Q$, the *contracted distance*

$$\rho_{C'/C}(x, y) \doteq \min\{\rho_{G(Q)}(x, y), \rho_{G'}(x, C) + \rho_{G'}(y, C)\}$$

and for $x \in C', y \in C$

$$\rho_{C'/C}(x, y) = \rho_{C'/C}(y, x) \doteq \rho_{G'}(x, C)$$

If $x \in C'$ and $A \subset C'$, we set $\rho_{C'/C}(x, A) \doteq \min_{a \in A} \rho_{C'/C}(x, a)$. We shall use also the notation

$$\rho_{C'/C} \doteq \max_{x,y \in C'} \rho_{C'/C}(x, y)$$

for the *contracted diameter* of C'.

Notice that with these definitions, if $\rho_{C'/C}(x, y) = 2$ for some $x, y \in C'$, then there exists $z \in C'$ such that $\rho_{C'/C}(x, z) = \rho_{C'/C}(z, y) = 1$. Notice also that $\rho_{C'/C}$ is, whence the notation, the pseudo-distance induced in C' by the distance in the quotient graph $G'/G(C)$, obtained from G' by contracting C to a vertex (this quotient graph is not a simple graph).

We shall also use the notations

$$\begin{aligned} \partial^j Q &\doteq \{x \in Q : |[x, C]| \geq j\} \\ \bar{\partial}^j Q &\doteq \{x \in Q : |[x, C]| < j\} = Q \setminus \partial^j Q \end{aligned}$$

Under these settings, we consider also

$$\Phi_{C'/C} \doteq \sum_{x \in Q} \min\{\max\{1, |[x, \bar{\partial}^2 Q]|\}, |[x, C]|\}$$

As Q and C will be fixed in the following of this Section, we shall freely omit the subindex C'/C when speaking of $\rho_{C'/C}$ and $\Phi_{C'/C}$. For $v \in C'$, $N'(v)$ denotes its neighborhood in G'.

In this general framework, we have

Theorem 1. *If $\rho_{C'/C} \leq 2$, $[S, \bar{S}]$ is an edge cut in G' such that $C \subset S$, and we let $S_1 \doteq S \cap Q$*

(1) *If $\max_{\bar{s} \in \bar{S}} \rho(\bar{s}, S) = 1$, then $|[S, \bar{S}]| \geq \max_{\bar{s} \in \bar{S}} |N'(\bar{s})|$.*
(2) *If $\max_{\bar{s} \in \bar{S}} \rho(\bar{s}, S) = 1$, then $|[S, \bar{S}] \geq |\bar{S}|$.*
(3) *If $\max_{\bar{s} \in \bar{S}} \rho(\bar{s}, S) = 2$, then $|\bar{S}| > \min_{\bar{s} \in \bar{S}} |N'(\bar{s})|$.*
(4) *If $\max_{\bar{s} \in \bar{S}} \rho(\bar{s}, S) = 2$, then $\max_{s \in S} \rho(s, \bar{S}) = 1$.*
(5) *If $\max_{s \in S_1} \rho(s, \bar{S}) = 1$, then $|[S_1, \bar{S}]| \geq max_{s \in S_1}(|N'(s)| - |N'(s) \cap C|)$.*
(6) *If $\max_{s \in S_1} \rho(s, \bar{S}) = 1$, then $|[S_1, \bar{S}]| \geq |S_1|$.*

Proof.

(1) Suppose that for any $\bar{s} \in \bar{S}$: $\rho(\bar{s}, S) = 1$. Let $\bar{s} \in \bar{S}$. Then we have k_1 edges $\bar{s}s_i, 1 \leq i \leq k_1$ with $s_i \in S$ and (eventually) k_2 edges $\bar{s}\bar{s}_j$, $\bar{s}_j \in \bar{S}$. But each $\bar{s}_j$ satisfies $\rho(\bar{s}_j, S) = 1$, thus we have k_2 new edges (here we used that G' is simple, because we assumed that the vertices $\bar{s}_j$ are different) $\bar{s}_j s'_j$, with $s'_j \in S$, whence
$$|[S, \bar{S}]| \geq k_1 + k_2 = |N'(\bar{s})|$$
(2) This follows at once if we notice that in this case for each $\bar{s} \in \bar{S}$ there is at least one $s \in S$ such that $\bar{s}s \in [\bar{S}, S]$.
(3) Notice that if $\rho(\bar{s}, S) = 2$, then $\{\bar{s}\} \cup N'(\bar{s}) \subset \bar{S}$.
(4) Let $\bar{s}_0 \in \bar{S}$ be such that $\rho(\bar{s}_0, S) = 2$. Then for each $s \in S$, as $\rho(\bar{s}_0, s) = 2$, there exists $\bar{s}'$ such that $\rho(\bar{s}_0, \bar{s}') = \rho(\bar{s}', s) = 1$. But, again, as $\rho(\bar{s}_0, S) = 2$, it follows that $\bar{s}' \in \bar{S}$, hence $\rho(s, \bar{S}) = 1$.
(5) For each $s \in S_1$ we have
$$N'(s) = (N'(s) \cap \bar{S}) \cup (N'(s) \cap S_1) \cup (N'(s) \cap C)$$
and, if $\max_{s \in S_1} \rho(s, \bar{S}) = 1$, then for each $s' \in N'(s) \cap S_1$ we have at least one edge in $[s', \bar{S}]$, thus $|N'(s)| \leq |[S_1, \bar{S}]| + |(N'(s) \cap C)|$.

(6) Our last statement follows noticing that $\max_{s\in S_1} \rho(s,\bar{S}) = 1$ means that for $s \in S_1$ $|[s,\bar{S}]| \geq 1$, and these sets are pairwise disjoint subsets of $[S_1,\bar{S}]$.

□

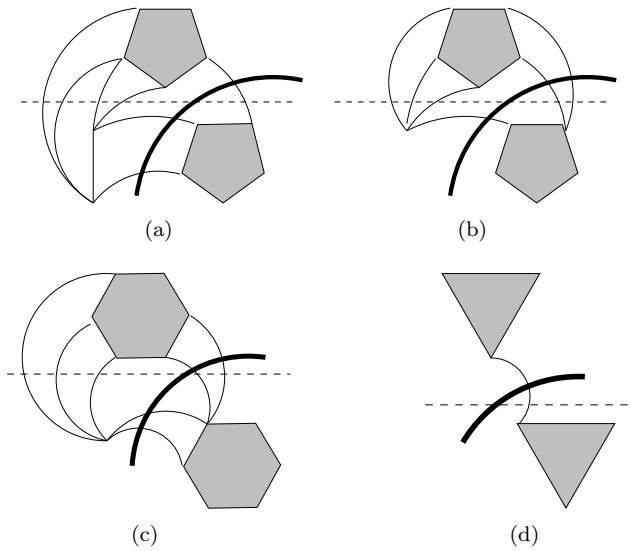


FIGURE 1. **Conventions:** 1.Filled polygons represent cliques, and curved arcs represent edges. 2.The dotted line separates C (the upper set of vertices) from Q . 3. The widest arc shows the cut $[S,\bar{S}]$. 4. $k = \min_{v\in Q} |N'(v)|$. **Descriptions:** (a) Here $|[S,\bar{S}]| = 3 < k = 4$, $|S\cap Q| = 2$, $|\bar{S}| = 5$, $\Phi = 3$, $S\cap Q = \partial^2 Q$. (b) Here $|[S,\bar{S}]| = 3 < k = 4$, $|S\cap Q| = 1$, $|\bar{S}| = 5$, $\Phi = 3$, $S\cap Q \neq \partial^2 Q$. (c) Here $|[S,\bar{S}]| = 4 < k = 5$, $|S\cap Q| = 1$, $|\bar{S}| = 6$, $\Phi = 3$, $S\cap Q \neq \partial^2 Q$. (d) Here $|[S,\bar{S}]| = 1 < k = 2$, $|S\cap Q| = 0$, $|\bar{S}| = 3$, $\Phi = 1$, $S\cap Q = \partial^2 Q = \emptyset$.

Corollary 1. *Assume that in addition to the hypotheses of Theorem 1, we have* $|[S,\bar{S}]| < \min_{v\in Q} |N'(v)|$. *Then*

(1) $\max_{\bar{s}\in\bar{S}} \rho(\bar{s},S) = 2$.
(2) $\max_{s\in S} \rho(s,\bar{S}) = 1$.
(3) $|[\bar{S},C]| \geq 1$.
(4) $|S_1| < |[S,\bar{S}]| < \min_{v\in Q} |N'(v)| < |\bar{S}|$.
(5) $S\cap Q \subset \partial^2 Q, \bar{S} \supset \bar{\partial}^2 Q$.
(6) $\Phi \leq |[S,\bar{S}]|$.

(See the examples in Figure 1.)

Proof. Points 1 and 2 are obvious consequences of our new hypothesis and points 1 and 4 in Theorem 1. To show point 3, notice that $\max_{s\in S} \rho(s,\bar{S}) = 1$. The first (from left to right) of the inequalities stated in point 4 follows from point 6 in Theorem 1, and point 3 in the present Theorem. The second of these inequalities

is assumed by our additional hypothesis, and the third follows immediately from point 3 in Theorem 1.

From point 5 in Theorem 1 and $|[C, \bar{S}]| \geq 1$ we obtain

$$|[S, \bar{S}]| > max_{s \in S_1}(|N'(s)| - |N'(s) \cap C|)$$

Thus for $s \in S_1$

$$N'(s) > |[S, \bar{S}]| > (|N'(s)| - |N'(s) \cap C|)$$

whence $|N'(s) \cap C| \geq 2$. Point 5 follows immediately from this.

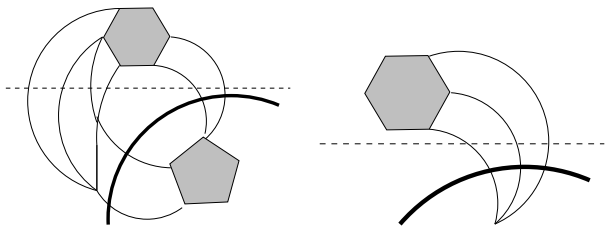
By our previous points, if $s \in S \cap Q$ then

$$|[s, \bar{S}]| \geq \max\{1, |[s, \bar{\partial}^2 Q]|\}$$

and of course for $\bar{s} \in \bar{S}$, $|[\bar{s}, S]| \geq |[\bar{s}, C]|$, thus

$$\begin{aligned} |[S, \bar{S}]| &= |[S \cap Q, \bar{S}]| + |[\bar{S}, C]| \\ &\geq \sum_{s \in S \cap Q} \max\{1, |[s, \bar{\partial}^2 Q]|\} + \sum_{\bar{s} \in \bar{S}} |[\bar{s}, C]| \\ &\geq \Phi \end{aligned}$$

□



(a) (b)

FIGURE 2. **Conventions:** 1.Filled polygons represent cliques, and curved arcs represent edges. 2.The dotted line separates C (the upper set of vertices) from Q. 3. The widest arc shows a minimal cut $[S, \bar{S}]$. **Descriptions:** (a) Here $|[S, \bar{S}]| = k = 4$, $\Phi = 4$, $|\partial^1 Q| = 3$. (b) Here $|[S, \bar{S}]| = k = 3$, $\Phi = 1$, $|\partial^1 Q| = 1$, $Q = \partial^1 Q$. This example shows that Corollary 2 includes an edge-connectivity version of the Expansion Lemma in [West, 2001], Lemma 4.2.3.

Corollary 2. *Let $k \leq \delta(G')$, and assume that*

(1) *$G(C)$ is $\delta(G')$-edge connected*
(2) *$\rho_{C'/C} \leq 2$*

Then any of the following

(1) *$\Phi \geq k$*
(2) *$|\partial^1 Q| \geq k$*
(3) *$Q = \partial^1 Q$*

implies that G' is k-edge-connected.

(See the examples in Figure 2.)

Proof. Let $[S, \bar{S}]$ be any cut in G'. We shall show that, under the listed hypotheses and any of the alternatives, $|[S, \bar{S}]| \geq k$.

If $S \cap C \neq \emptyset$ and $\bar{S} \cap C \neq \emptyset$, then, as

$$[S \cap C, \bar{S} \cap C] \subset [S, \bar{S}]$$

is a cut in $G(C)$, which we assumed to be k-edge connected, we obtain $|[S, \bar{S}]| \geq k$.

Without loss of generality, we assume in the sequel that $C \subset S$. We argue by contradiction assuming that there exists some S such that $|[S, \bar{S}]| < k$, so that we are under the hypothesis of Corollary 1.

The first of our alternative hypothesis contradicts point 6 in the conclusions of Corollary 1.

When $v \in \partial^1 Q$,

$$\min\{\max\{1, |[v, \bar{\partial}^2 Q]|\}, |[v, C]|\} \geq 1$$

so that we have $|\partial^1 Q| \leq \Phi$, *i.e.* the second of our alternative hypothesis implies the first one.

To finish our proof, notice that if $Q = \partial^1 Q$, as $\bar{S} \subset Q$, we have $\rho(\bar{s}, S) = 1$ for any $\bar{s} \in \bar{S}$, contradicting point 1 in the conclusions of Corollary 1. □

Definition 5. *As this Corollary will be a key for our later results, we shall set for future reference*

$$\Psi_{C'/C}(k, G) \doteq \max\{\Phi_{C'/C} - k, |\partial^1 Q| - k, |\partial^1 Q| - |Q|\}, \quad \textit{for } k \leq \delta(G')$$

so that the validity of some of the three last alternative hypotheses in Corollary 2 could be re-stated as $\Psi_{C'/C}(k, G) \geq 0$.

2.3.1. *Remarks.* Corollary 2 is related to a well known theorem of Plesník (see [Plesník, 1975], Theorem 6), which states that in a simple graph of diameter 2 the edge connectivity is equal to the minimum degree.

2.4. **Edge-connectivity and cores decomposition.** Let us introduce the following

Definition 6. *Consider a graph G, its cores decomposition (see Definition 1)*

$$V(G) = \cup\{C_k : 0 \leq k \leq k_{\max}\}$$

and $A \subset V(G)$.

1. *We say that A is k-core connected in strict sense if $A \cap C_k$ is k-connected in strict sense.*
2. *We say that A is k-core connected in wide sense if $A \cap C_k$ is k-connected in wide sense in $G(C_k)$.*
3. *We say that A is core connected in strict sense if $A \cap C_k$ is k-connected in strict sense for all k such that $A \cap C_k \neq \emptyset$.*
4. *We say that A is core connected in wide sense if $A \cap C_k$ is k-connected in wide sense in $G(C_k)$ for all k such that $A \cap C_k \neq \emptyset$.*
5. *We say that G is k-core connected when $V(G)$ is k-core connected.*
6. *We say that G is core-connected when $V(G)$ is core-connected.*

Next, we shall describe two algorithms that provide a mechanism to build (hopefully big) core-connected sets of vertices, in strict and wide sense respectively. Both algorithms proceed recursively, starting from the highest core, looking for a not-yet-joined cluster able to be joined, joining it, and restarting from this new set of

vertices. The difference between the algorithms lies in the meaning of "able to be joined".

When Q is a cluster, we denote $k(Q)$ its shell index. Let $\mathfrak{Q}$ be a family of clusters, and let $k(\mathfrak{Q}) \doteq \max\{k(Q) : Q \in \mathfrak{Q}\}$ denote the maximum shell index of the clusters in $\mathfrak{Q}$.

Algorithm 1: strict sense core-connected

Input: $\mathfrak{Q}$, the family of all clusters of the k-core decomposition of a graph G
Output: $C \subset V$, core-connected in strict sense
Initialization: $C \leftarrow \emptyset$, $k \leftarrow k_{max}$

1 **begin**
2   **while** $C = \emptyset$ **and** $\mathfrak{Q} \neq \emptyset$ **and** $k \geq 2$ **do**
3         $k \leftarrow k(\mathfrak{Q})$;
4   **if** *there is some* $Q \in \mathfrak{Q}$ *satisfying:* $\left\{ \begin{array}{l} k(Q) = k \\ \rho_Q \leq 2 \end{array} \right]$ **then**
5             $C \leftarrow C \cup Q$;
6   **end**
7         $\mathfrak{Q} \leftarrow \mathfrak{Q} \setminus \{Q \in \mathfrak{Q} : k(Q) = k\}$;
8   **end**
9   **while** $\mathfrak{Q} \neq \emptyset$ **and** $k \geq 2$ **do**
10         $k \leftarrow k(\mathfrak{Q})$;
11   **while** *there is some* $Q \in \mathfrak{Q}$ *satisfying:* $\left\{ \begin{array}{l} k(Q) = k \\ \rho_{C \cup Q/C} \leq 2 \\ \Psi_{C \cup Q/C}(k, G) \geq 0 \end{array} \right]$ **do**
12             $C \leftarrow C \cup Q$;
13             $\mathfrak{Q} \leftarrow \mathfrak{Q} \setminus \{Q\}$;
14   **end**
15         $\mathfrak{Q} \leftarrow \mathfrak{Q} \setminus \{Q \in \mathfrak{Q} : k(Q) = k\}$;
16   **end**
17 **end**

(see Figure 3 for illustration)

Theorem 2. *Algorithm 1 always stops, and when it stops the set C is core connected in strict sense.*

Proof.

- For the first **while** loop, step 1 computes the maximum k for the actual family of clusters, while step 1 deletes from $\mathfrak{Q}$ all clusters with shell index k. As a consequence, k is strictly decreasing, and when the algorithm arrives to step 1, either $\mathfrak{Q}$ is empty or C has a cluster verifying the hypothesis of Corollary 2.
- The second **while** loop will also finish because steps 1 and 1 assure that k is strictly decreasing.
- The nested **while** loop will finish because the family $\mathfrak{Q}$ is finite.
- Assume that the actual C is core connected in strict sense when we arrive to step 1. By construction, $C \cap C_j = C$ when $j \leq k$, and $C \cap C_j$ are previous instances of C when $j > k$ (in fact, this instances are obtained each time that we arrive to step 1). The new C, let us call it C' for a moment, has the same intersections with C_j for $j > k$, and when $j \leq k$ the

intersection is $C' = C \cup Q$, that is k connected in strict sense by Corollary 2, as the conditions for the selection of Q in step 1 match the hypothesis of Corollary 2. Thus, all the instances of C during the algorithm are core connected in strict sense, whence the final C is core connected in strict sense.

□

Algorithm 2: wide sense core-connected

Input: $\mathfrak{Q}$, the family of all clusters of a graph G
Output: $C \subset V$, core-connected in wide sense
Initialization: $C \leftarrow \emptyset$, $D \leftarrow \emptyset$, $\mathfrak{Q}' \leftarrow \emptyset$, $k \leftarrow k_{max}$

```
begin
    while C = ∅ and 𝔔 ≠ ∅ and k ≥ 2 do
        k ← k(𝔔) ;
        if there is some Q ∈ 𝔔 satisfying: { k(Q) = k ; ρ_Q ≤ 2 } then
            C ← C ∪ Q ;
            𝔔 ← 𝔔 \ {Q} ;
        end
        𝔔' ← 𝔔' ∪ {Q ∈ 𝔔 : k(Q) = k} ;
        𝔔 ← 𝔔 \ {Q ∈ 𝔔 : k(Q) = k} ;
    end
    while 𝔔 ≠ ∅ and k ≥ 2 do
        k ← k(𝔔) ;
        while there is some Q' ∈ 𝔔' satisfying:
        { k(Q') ≥ k ; ρ_{(C∪D∪Q')/(C∪D)} ≤ 2 ; Ψ_{(C∪D∪Q')/(C∪D)}(k, G) ≥ 0 } do
            D ← D ∪ Q' ;
            𝔔' ← 𝔔' \ {Q'} ;
        end
        while there is some Q ∈ 𝔔 satisfying:
        { k(Q) = k ; ρ_{(C∪D∪Q)/(C∪D)} ≤ 2 ; Ψ_{(C∪D∪Q)/(C∪D)}(k, G) ≥ 0 } do
            C ← C ∪ Q ;
            𝔔 ← 𝔔 \ {Q} ;
        end
        𝔔' ← 𝔔' ∪ {Q ∈ 𝔔 : k(Q) = k} ;
        𝔔 ← 𝔔 \ {Q ∈ 𝔔 : k(Q) = k};
    end
end
```

(see Figure 3 for illustration)

Theorem 3. *Algorithm 2 always stops, and when it stops the set C is core connected in wide sense.*

Proof.

- For the first **while** loop, step 2 computes the maximum k for the actual family of clusters, while step 2 deletes from the actual $\mathfrak{Q}$ all the clusters with shell index k. As a consequence, k is strictly decreasing, and when the algorithm arrives to step 2, either $\mathfrak{Q}$ is empty or C has a cluster verifying the hypothesis of Corollary 2.
- The second **while** loop will also finish because steps 2 and 2 assure that k is strictly decreasing.
- The nested **while** loops will finish because the families $\mathfrak{Q}'$ and $\mathfrak{Q}$ are finite.
- Assume that the actual C is core connected in wide sense when we arrive to step 2. By construction, $C \cap C_j = C$ when $j \leq k$, and $C \cap C_j$ are previous instances of C when $j > k$ (in fact, this instances are obtained each time that we arrive to step 2). The new C, let us call it C' for a moment, has the same intersections with C_j for $j > k$, and when $j \leq k$ the intersection is $C' = C \cup Q$, that is k connected in wide sense by Corollary 2, as the conditions for the selection of Q in step 2 match the hypothesis of Corollary 2. Thus, all the instances of C during the algorithm are core connected in wide sense, whence the final C is core connected in wide sense.

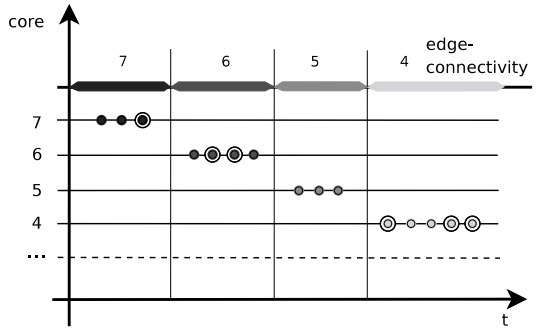
□

Notice that in both algorithms we have limited C to include nodes from the shells with $k \geq 2$. The conditions from Corollary 2 are not natural at level 1. It is obvious that if C is the strict sense core-connected set constructed by Algorithm 1, then after adjoining to C all the clusters from shell 1 that are connected to it, the result is also strict sense core-connected. Analogously, it is clear that if C and D are the sets constructed by Algorithm 2, if we adjoin to C all the clusters from shell 1 connected to C or D, then the new C is also core-connected in wide sense. We assume in the following Section that C has been extended according to these remarks.

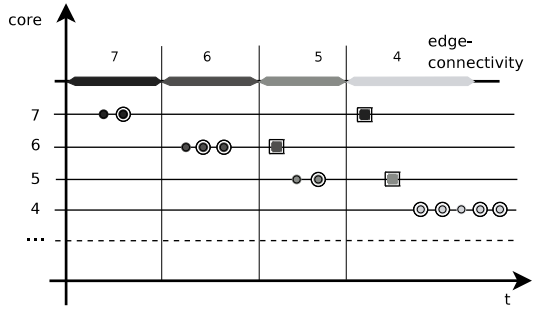
3. Applications

In this section we will test our algorithms in some Internet maps obtained from different sources. Each of them has its own biases and explores the Internet at a particular level:

(1) The Route Views Project [ORV, 2001], for instance, uses a short amount of BGP routers to peer with routers in other ASes and thus get routing tables. As a bias, this method does not detect hidden routes (not all inter-AS routes are public due to policies and agreements).
(2) The CAIDA Association [CAIDA, 1998] developed *skitter* probes (which now evolved into the *Ark* infrastructure). Based on traceroute, this measurement nodes send ICMP requests to routers in order to discover paths. *Ark* has 38 monitors at November 2009. As an advantage, these tool finds the real routes followed by packets, advertised or not.
(3) The DIMES Project [DIMES, 2004] is a distributed system composed of around one thousand voluntary nodes –anyone may subscribe even with a low CPU power or bandwidth–. It explores the Internet with tools such as traceroute and ping to discover the topology.



(a) Strict sense connectivity



(b) Wide sense connectivity

FIGURE 3. These two schemata show the progress in time (t) of both algorithms. Dots represent clusters, circled dots clusters in C, squared dots clusters in D. Both algorithms start looking for a cluster of diameter 2, descending from the top shell. If they find someone, this is the initial C. Then, the first algorithm looks, descending shell by shell, for clusters to add to C, preserving core-connectivity in strict sense. The second algorithm does the same, but each time that it ends with the inspection of a shell, it looks between the omitted clusters from previous shells (saved in $\mathfrak{Q}'$), for clusters to add to D. These clusters provide more possibilities of connection (in wide sense) to the next shell, thus producing a bigger set C.

(4) Mercator [Govindan and Tangmunarunkit, 2000] uses hop-limited ICMP probes to discover the Internet map with an *informed random* heuristic. The algorithm chooses some nodes in the path with the IP source-routing option, which is no longer available.

At the Autonomous Systems interconnection level (called AS) data has been obtained from the Oregon Route Views Project, the CAIDA Association and the DIMES Project. Router level maps (IR) come from CAIDA, DIMES and Mercator.

To test our algorithms 1 and 2 we implemented them into the LaNet-vi open source software [Alvarez-Hamelin et al., 2005]. Based on the k-core decomposition, this tool computes and visualizes the strict and wide core-connected components

giving also the lower bound of connectivity for every pair of nodes through a logging file. This bound is obtained applying Corollary 2 for nodes in the core-connected subgraph, and using the value of Φ for the others. In the LaNet-vi color visualization (B&W differences will be mentioned in brackets), the nodes color (nodes shade) determines their core, while the border color (border shade) suggests a lower bound for connectivity with other nodes. In fact, the absence of border points out that the node belongs to the core-connected subgraph and so its connectivity with nodes in inner shells from the core-connected subgraph is at least the node shell index. A colored border (shaded border), instead, implies that the node is not in C but it belongs to the D set in algorithm 2, meaning that we assure certain level of edge-connectivity with internal clusters in C though this bound is lesser than the node shell index. Finally for white nodes (squared nodes) we can assure no edge-connectivity with others, and maybe the node is poorly connected.

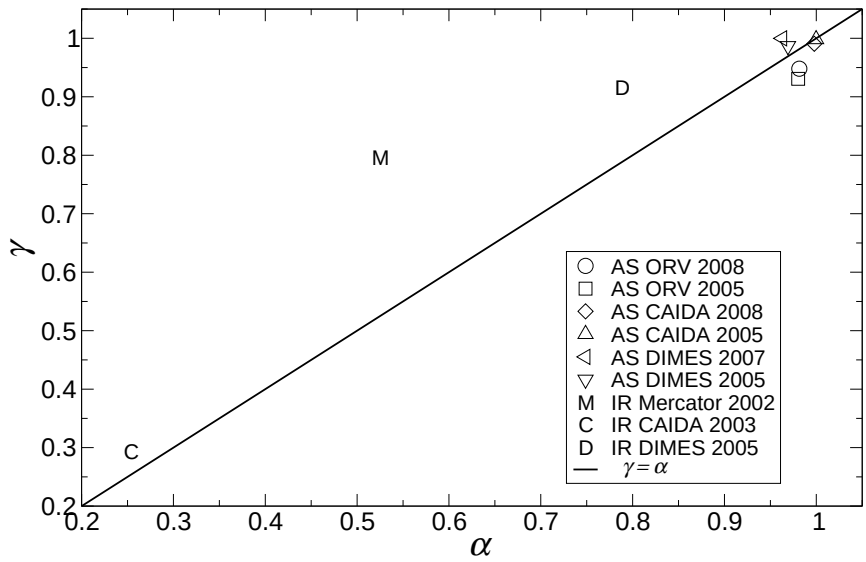


FIGURE 4. Comparison between γ and α. The line $\alpha = \gamma$ splits the space in two areas, the upper one for graphs with C concentrated on high populated shells ($\gamma > \alpha$), and the bottom for C mainly in low populated shells ($\gamma < \alpha$).

We performed different tests to show the effectiveness of our tool. Defining the fraction of nodes in C (the core-connected subgraph) for each shell k as:

$$\rho_k = \frac{|S_k \cap C|}{|S_k|} \quad ,$$

we studied the following quantities:

$$\alpha = \frac{1}{k_{\max}} \sum_{k=1}^{k_{\max}} \rho_k$$

$$\beta = \frac{2}{k_{\max}(k_{\max}+1)} \sum_{k=1}^{k_{\max}} \rho_k \, k$$

$$\gamma = \sum_{k=1}^{k_{\max}} \rho_k \, \frac{|S_k|}{|C|} \; .$$

In fact, α, β and γ may be interpreted as probabilities for a node to be in C under different models: α is the average of all ρ_k; β is a weighted average of ρ_k, where each shell is weighted according to its number; and γ is a weighted average of the fractions ρ_k with each k-shell weighted by its size.

Figure 4 shows γ as a function of α. Networks with $\gamma > \alpha$ have most part of C in high populated shells. We also present β as a function of α in figure 5, where $\beta > \alpha$ means that a lower part of C is found in lower shell indexes.

On the one hand, AS maps are close to point $(1, 1)$ in both figures, and this means that most of their nodes belong to C. AS DIMES maps have $\beta < \alpha$ because some of their higher shells are empty. But on the other hand, IR maps are worse core-connected than AS because the ratio of nodes in C is low. Each IR has a different behavior; the most preferable is IR DIMES (90% of nodes in C and in higher shells). The reason why the other IR maps give a small C is the presence of big clusters in which the $\rho \leq 2$ condition is not satisfied. This is probably a bias in the exploration: IR DIMES maps may be more accurate because they use a higher number of sources (thousands, according to [Dall'Asta et al., 2006]). We think that a detailed picture of the Internet at the IR level will verify our hypotheses.

A different analysis was done at shell level. At first we counted pairs of nodes $\{u, v\}$ belonging to the wide-sense core-connected graph, i.e., having both ends in C. Figure 6 presents this information as a function of $min(sh(u), sh(v))$ for the AS CAIDA 2008 map. It follows that only the first three shells have nodes out of C, and that they are few (about 1% per shell). This behavior is similar in all maps but IR CAIDA 2003, where some nodes in low, medium and high shells do not belong to the core-connected subgraph.

Secondly, we compared the minimum shell index of a pair of nodes $\{u, v\}$ in a core-connected subgraph with the connectivity, this last obtained with the Gomory-Hu algorithm (see chapter 4 in [Ford and Fulkerson, 1962]). In figure 7 we display the connectivity of each pair $\{u, v\}$ as a function of the minimum shell index $min(sh(u), sh(v))$, for the AS CAIDA 2008 map. We see that connectivity is higher than the bound obtained by the core-connected graph, still the average values are relatively close for low and medium shells (less than 100% up to shell 13). As our bound is related to connectivity through cores (i.e., taking only the maximum k-core containing $\{u, v\}$ to find paths) we show also the connectivity through the k-core, where $k = min(sh(u), sh(v))$ in Figure 8. Clearly, connectivity is larger than connectivity through the k-core, but this last is closer to the minimum shell index. The other AS maps have analogous behavior.

As a remark, we couldn't compute connectivity for IR maps because it is expensive on RAM memory for their size, which is greather than 100,000 nodes.

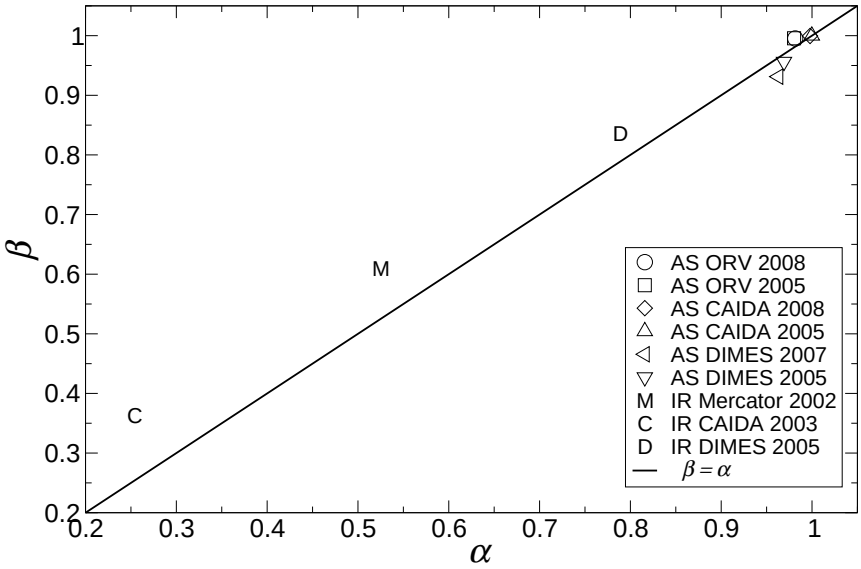


FIGURE 5. β vs. α for different maps.

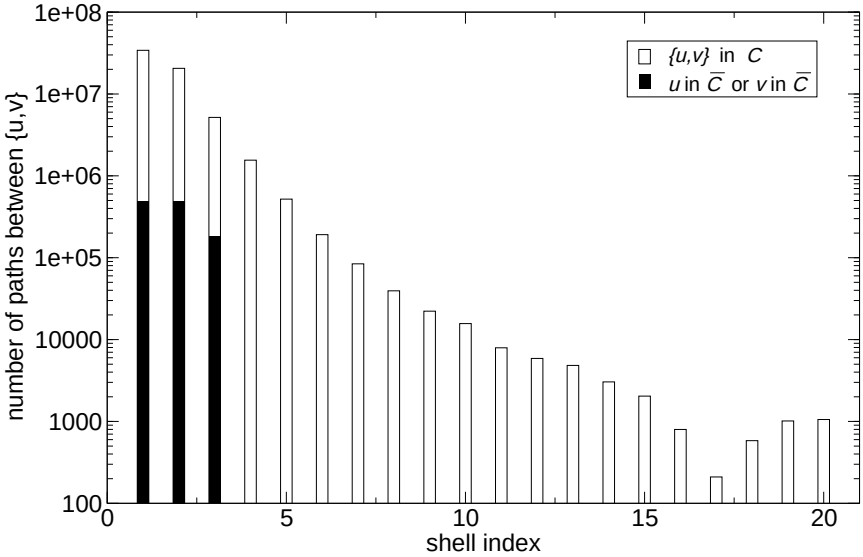


FIGURE 6. Histogram showing the number of paths from u to v in the AS CAIDA 2008 map. white is for $\{u, v\} \in C$ and black is for $u \in \bar{C}$ or $v \in \bar{C}$.

To conclude our analysis, we present in figure 9 a color visualization for AS CAIDA 2008 map (greyscale is shown in figure 10), testing the algorithm 2.

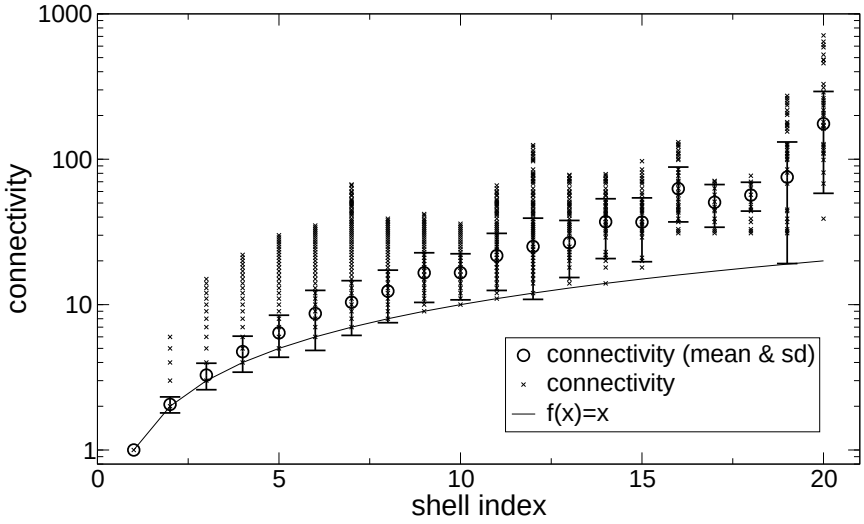


FIGURE 7. Connectivity for $\{u, v\}$ pairs vs. their minimum shell index. Circles stand for mean values per shell index and error bars show the standard deviation; crosses show connectivity; the bounding line *connectivity=shell index* is also displayed.

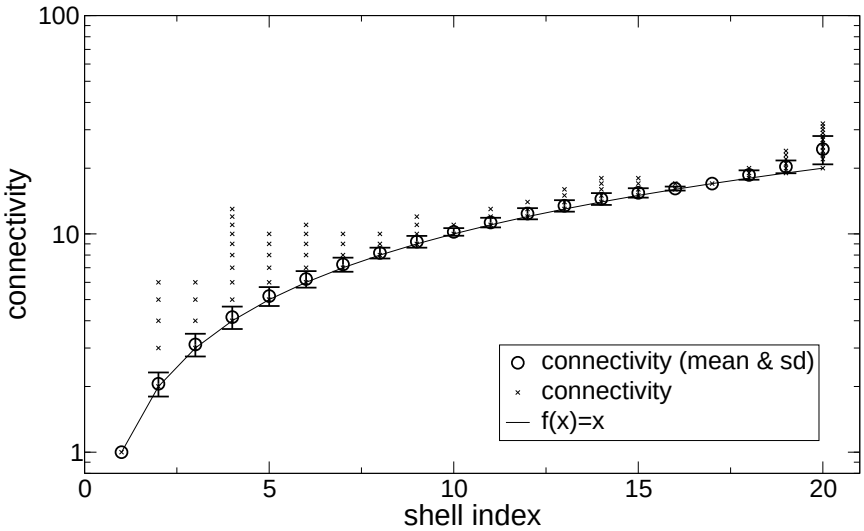


FIGURE 8. Connectivity through the cores for $\{u, v\}$ pairs vs. their minimum shell index. Circles stand for mean values and error bars show the standard deviation; crosses show connectivity; the bounding line *connectivity=shell index* is also displayed.

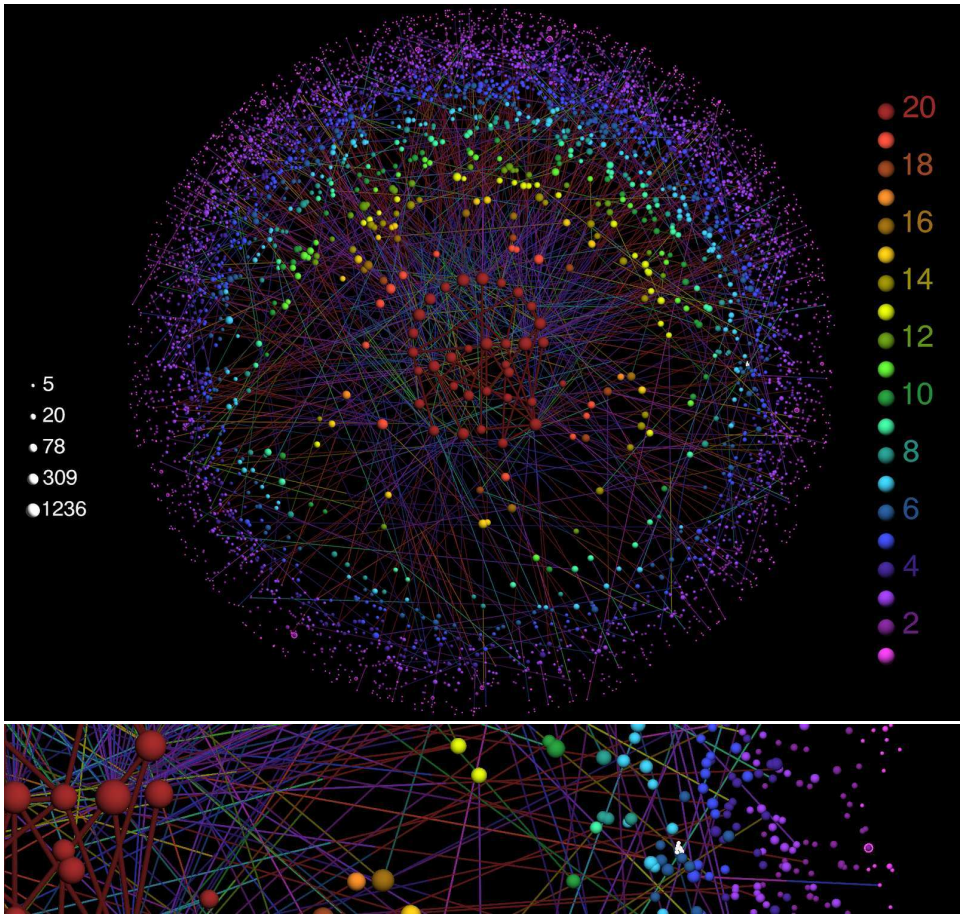


FIGURE 9. Color visualization of AS CAIDA 2008 map displaying the wide-sense core-connected subgraph (algorithm 2). Bottom: a detail showing some nodes out of the core-connected subgraph, and some in white (out of C and D); we can see one node belonging to D on the right.

4. CONCLUSIONS

In this work we have defined core-connected graphs, for which we obtain a lower bound of connectivity between nodes: the minimum shell index of them in the graph k-core decomposition. To formalize this relation we provided a theorem (Corollary 2) that gives sufficient conditions to assure the forementioned lower bound. This theorem is an extension of Plesnik's theorem (see [Plesník, 1975]):

Plesnik's theorem asserts that if a simple graph has diameter 2, then the connectivity is at least $\delta(G)$. It is not difficult to show that the hypotheses of G being simple can be easily relaxed, thus obtaining that the connectivity is at least $\min_{v \in V} |N(v)|$ (which is $\delta(G)$ in the simple case). The combinatorial nature of connectivity, where multiple bifurcations give rise to a great multiplicity of

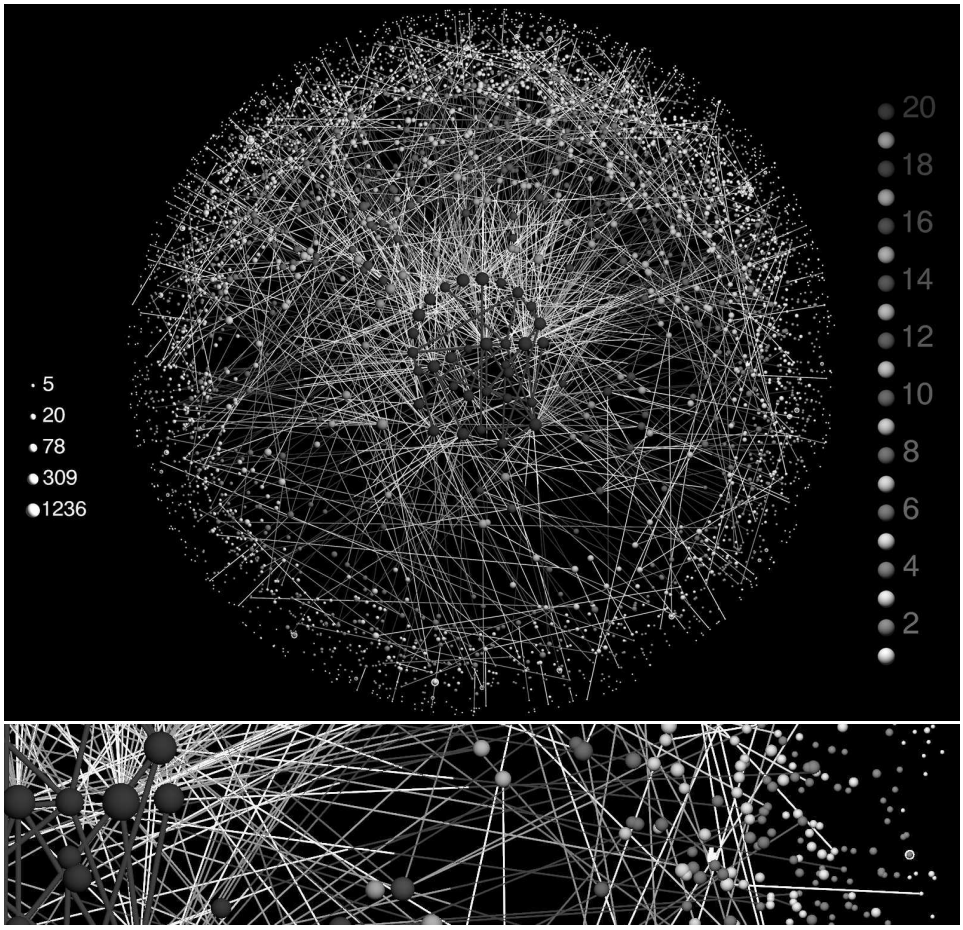


FIGURE 10. Greyscale visualization of AS CAIDA 2008 map displaying the wide-sense core-connected subgraph (algorithm 2). Bottom: a detail showing some nodes out of the core-connected subgraph, and some in white (out of C and D); we can see one node belonging to D on the right.

paths between two vertices, in the absence of bottle necks, justifies the presumption that the diameter 2 condition is rather artificial. In fact, [Carmi et al., 2006, Carmi et al., 2007] have noticed that the node to node connectivity $k'(u, v)$ is at least $\min(sh(u), sh(v))$ for all but very exceptional pairs u, v in many real life net graphs (indeed, the connectivity in this paper is less than k', because the authors count only disjoint paths, where disjoint means that they do not share neither edges nor vertices). Our results herein show some semilocal conditions under which this bound for the connectivity holds, where semilocal means here that the conditions involve, for each k, the relations between a k-core and his next $(k-1)$-shell. Rather that an alternative procedure to find the connectivity, which can be rather efficiently found with the Gomory-Hu algorithm (see chapter 4 of [Ford and Fulkerson, 1962]),

we hope that our results give some new insight on the local-global relations for connectivity, useful in real life net graphs.

We also developed two algorithms to get core-connected subgraphs of a given graph G: one for strict-sense connectivity (whose complexity is $O(e)$) and one for wide-sense connectivity ($O(e \times \sqrt{e})$). In the strict-sense algorithm each cluster is considered only once to determine if it fulfills conditions about its diameter and Φ. Being $e(Q)$ the amount of edges of cluster Q counting the internal ones and the ones that connect it with inner cores (clusters don't share edges with other clusters in the same core), then in $O(e(Q))$ it can be determined if the cluster's diameter is less or equal to 2. The Φ condition can also be evaluated in $O(e(Q))$ as it implies a BFS in the cluster. Consequently, covering all clusters, the algorithm runs in $O(e)$. In the wide-sense algorithm each cluster may be considered up to k_{max} times (once per each shell). But k_{max} is bounded above by $\sqrt{e}$ because to obtain a k-shell, $k+1$ nodes are needed at least with k connections each, so the graph must have $k \times (k-1)$ edges. Then we get a complexity of $O(e \times \sqrt{e})$.

Finally we included these algorithms in the open-source software LaNet-vi to visualize core-connected subgraphs and list nodes in them, showing that it works for the Internet maps.

We are working on a possible relaxation on the conditions involved with Φ and the treatment of clusters with $\rho > 2$. We will also look for explanations on the high connectivity found in the higher shells.

References

[Alvarez-Hamelin et al., 2005] Alvarez-Hamelin, J. I., Beiro, M. G., Dall'Asta, L., Barrat, A., and Vespignani, A. (2005). **La**rge **Net**work **vi**sualization tool, `http://sourceforge.net/projects/lanet-vi`.

[Alvarez-Hamelin et al., 2006] Alvarez-Hamelin, J. I., Dall'Asta, L., Barrat, A., and Vespignani, A. (2006). Large scale networks fingerprinting and visualization using the k-core decomposition. In Weiss, Y., Schölkopf, B., and Platt, J., editors, *Advances in Neural Information Processing Systems 18*, pages 41–50, Cambridge, MA. MIT Press.

[Alvarez-Hamelin et al., 2008] Alvarez-Hamelin, J. I., Dall'Asta, L., Barrat, A., and Vespignani, A. (2008). K-core decomposition of Internet graphs: hierarchies, self-similarity and measurement biases. *Networks and Heterogeneous Media*, 3(2):371–393.

[Andersson and Asati, 2009] Andersson, L. and Asati, R. (2009). RFC 5462: Multiprotocol Label Switching (MPLS) Label Stack Entry: "EXP" Field Renamed to "Traffic Class" Field . `ftp://ftp.rfc-editor.org/in-notes/rfc5462.txt`.

[Barabási and Albert, 1999] Barabási, A. L. and Albert, R. (1999). Emergence of scaling in random networks. *Science*, 286:509–512.

[Batagelj and Zaversnik, 2002] Batagelj, V. and Zaversnik, M. (2002). Generalized Cores. *CoRR*, arXiv.org/cs.DS/0202039.

[Batagelj and Zaversnik, 2003] Batagelj, V. and Zaversnik, M. (2003). An O(m) Algorithm for Cores Decomposition of Networks. *CoRR*, arXiv.org/cs.DS/0310049.

[Beiró et al., 2008] Beiró, M. G., Alvarez-Hamelin, J. I., and Busch, J. R. (2008). A low complexity visualization tool that helps to perform complex systems analysis. *New J. Phys*, 10(12):125003.

[CAIDA, 1998] CAIDA (1998). Cooperative Association for Internet Data Analysis , Router-Level Topology Measurements. `http://www.caida.org/tools/measurement/skitter/`.

[Carmi et al., 2006] Carmi, S., Havlin, S., Kirkpatrick, S., Shavitt, Y., and Shir, E. (2006). MEDUSA - New Model of Internet Topology Using k-shell Decomposition. Technical Report 0601240v1, arxiv.

[Carmi et al., 2007] Carmi, S., Havlin, S., Kirkpatrick, S., Shavitt, Y., and Shir, E. (2007). A model of internet topology using k-shell decomposition. *PNAS*, 104:11150–11154.

[Dall'Asta et al., 2006] Dall'Asta, L., Alvarez-Hamelin, J. I., Barrat, A., Vázquez, A., and Vespignani, A. (2006). Exploring networks with traceroute-like probes: Theory and simulations. *Theor. Comput. Sci.*, 355(1):6–24.

[DIMES, 2004] DIMES (2004). Distributed Internet MEasurements and Simulations. `http://www.netdimes.org`.

[Erdös and Rényi, 1959] Erdös, P. and Rényi, A. (1959). On random graphs I. *Publ. Math. (Debrecen)*, 6:290–297.

[Fabrikant et al., 2002] Fabrikant, A., Koutsoupias, E., and Papadimitriou, C. H. (2002). Heuristically Optimized Trade-Offs: A New Paradigm for Power Laws in the Internet. *LNCS*, 2380:110–.

[Faloutsos et al., 1999] Faloutsos, M., Faloutsos, P., and Faloutsos, C. (1999). On power-law relationships of the internet topology. In *SIGCOMM*, pages 251–262.

[Ford and Fulkerson, 1962] Ford, L. R. and Fulkerson, D. R. (1962). *Flows in Networks*. Princeton University Press.

[Govindan and Tangmunarunkit, 2000] Govindan, R. and Tangmunarunkit, H. (2000). Heuristics for Internet Map Discovery. In *IEEE INFOCOM 2000*, pages 1371–1380, Tel Aviv, Israel. IEEE.

[ORV, 2001] ORV (2001). University of Oregon Route Views Project. `http://www.routeviews.org/`.

[Pastor-Satorras et al., 2001] Pastor-Satorras, R., Vázquez, A., and Vespignani, A. (2001). Dynamical and correlation properties of the internet. *Phys. Rev. Lett.*, 87(25):258701.

[Plesník, 1975] Plesník, J. (1975). Critical graphs of a given diameter. *Acta Fac. Rerum Natur. Univ. Comenian. Math.*, 30:71–93.

[Rekhter et al., 2006] Rekhter, Y., Li, T., and Hares, S. (2006). RFC 4271: A Border Gateway Protocol 4. `http://www.rfc-archive.org/getrfc.php?rfc=4271`, `http://www.bgp4.as/`.

[Rosen et al., 2001] Rosen, E., Viswanathan, A., and Callon, R. (2001). RFC 3031: Multiprotocol Label Switching Architecture. `http://www.rfc-editor.org/rfc/rfc3031.txt`.

[Serrano et al., 2006] Serrano, M. A., Boguñá, M., and Díaz-Guilera, A. (2006). Modeling the Internet. *Eur. Phys. J. B*, 50:249–254.

[Waxman, 1988] Waxman, B. M. (1988). Routing of Multipoint Connections. *IEEE Journal on Selected Areas in Communications*, 6(9):1617–1622.

[West, 2001] West, D. B. (2001). *Introduction to Graph Theory*. Prentice Hall.

[Zegura et al., 1997] Zegura, E. W., Calvert, K. L., and Donahoo, M. J. (1997). A quantitative comparison of graph-based models for Internet topology. *IEEE/ACM Transactions on Networking*, 5(6):770–783.