

Understanding Edge Connectivity in the Internet through Core Decomposition

J. Ignacio Alvarez-Hamelin , Mariano G. Beiró & Jorge R. Busch

To cite this article: J. Ignacio Alvarez-Hamelin , Mariano G. Beiró & Jorge R. Busch (2011) Understanding Edge Connectivity in the Internet through Core Decomposition, Internet Mathematics, 7:1, 45-66, DOI: [10.1080/15427951.2011.560786](https://doi.org/10.1080/15427951.2011.560786)

To link to this article: <http://dx.doi.org/10.1080/15427951.2011.560786>



Published online: 16 Mar 2011.



Submit your article to this journal [↗](#)



Article views: 134



View related articles [↗](#)



Citing articles: 1 View citing articles [↗](#)

Understanding Edge Connectivity in the Internet through Core Decomposition

J. Ignacio Alvarez-Hamelin, Mariano G. Beiró, and Jorge R. Busch

Abstract. The Internet is a complex network composed of several networks: the autonomous systems. Each of them is designed with the aim of transporting information efficiently. This information is carried over routes, which are discovered by routing protocols, such as the border gateway protocol (BGP). The protocols may find possible paths between nodes whenever they exist, or even find paths satisfying specific constraints, e.g., a certain quality of service (QoS). Here, we study connectivity as a network attribute related to both situations; we provide a formal lower bound to it based on core decomposition and low-complexity algorithms to find it. Then we apply these algorithms to analyze maps obtained from the prominent Internet mapping projects, and use the LaNet-vi open-source software for their visualization.

I. Introduction

Today, the 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 to them. This paper aims at the robustness problem and the study of connectivity in the Internet using the k -core decomposition of

the network. Connectivity is quite important, for it represents how tolerant the network is to link failures or what quality of service (QoS) it may offer (e.g., see the requests for comments (RFCs) of multiProtocol label switching [Rosen et al. 01, Andersson and Asati 09]). For example, forwarding packets to well-connected nodes may help in finding paths that fulfill a required QoS. Therefore, node-to-node connectivity information may speed up the convergence of routing protocols such as the border gateway protocol (BGP).

To classify our work, we resort to [Bollobás 04], in which the author splits the research on random graphs into five categories: the study of real-world networks and their properties, the proposition of new models, model simulation, heuristic analysis, and rigorous mathematical study of the models to provide theorems about their properties. Our work falls into the first category in this classification, i.e., the study of real-world network properties. Understanding graph properties and measuring them is the basis for modeling the Internet. By studying degree distribution, the authors of [Faloutsos et al. 99] pointed out a remarkable characteristic of the Internet topology: the node degrees have a heavy-tailed behavior. Until then, prominent models, such as those of [Waxman 88] and [Zegura et al. 97], were based on classical [Erdős and Rényi 59] random graphs, whose degrees obey a Poisson distribution. But from then on, a huge number of models arose with respect to inhomogeneous random graphs, of which we mention only some nodal ones, starting with the model of [Barabási and Albert 99] based on preferential attachment; the model of [Aiello et al. 00] with given expected degrees; the generalized random graph of [Newman et al. 01], whereby the degree distribution may be arbitrarily set; the more general model in [Bollobás et al. 07]; and the FKP model, which proposes a trade-off between a topological and a geometrical objective (see [Spatharis et al. 09] and the references therein).

Another example of attributes that provided a better understanding of the Internet is found in [Pastor-Satorras et al. 01]. Using the average neighbor degree, the authors showed that the Internet topology at the autonomous system (AS) level is different from the Internet topology at the inter-router (IR) level, leading to refinements in some models.

Our interest here is to study connectivity through k -cores, but we do not deal with modeling. In fact, several authors have studied the emergence of cores in random graphs [Riordan 08, Janson 09], and some of them have even given conditions under which the k -core is k -connected with high probability (whp). For real graphs, this topic has been addressed in [Carmi et al. 06].

We call graphs whose k -core is k -connected *k -core-connected*, and those graphs for which this holds for all k , *core-connected*. We present some algorithms to obtain core-connected subgraphs of a given graph, and a general lower bound for the connectivity of the k -cores. These algorithms are of low complexity and are

based on a generalization of Plesník's theorem [Plesník 75] on the connectivity of diameter-2 graphs.

The k -core decomposition is useful for identifying exploration sources through visualization [Alvarez-Hamelin et al. 06, Beiró et al. 08], to validate models [Serrano et al. 06], and to discover exploration biases in the Internet [Alvarez-Hamelin et al. 08]. These facts support the k -core analysis as one of the relevant tools for describing Internet topology maps.

The paper is organized as follows. Section 2 is devoted to developing some mathematical tools used later and to formalizing the relation between k -cores and k -connected subgraphs. In Section 3, we present our software LaNet-vi. This tool implements the aforementioned ideas and provides a visualization of the network according to its connectivity. We present some applications to Internet maps. Finally, we conclude our paper with some remarks and comments on future work.

2. Foundations

Let us introduce some general graph notions and notation. Let G be a simple graph (i.e., a graph with no loops and no multiple edges) with vertex set $V(G)$ and edge set $E(G)$ (we follow the notation in [West 01]). 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 $b \in B$. Since we consider edges to be undirected, we have $[A, B] = [B, A]$. Abusing 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 define $n(G) \doteq |V|$, $e(G) \doteq |E|$, $\delta(G) \doteq \min_{v \in V} d_G(v)$, $\Delta(G) \doteq \max_{v \in V} d_G(v)$. The *neighborhood* $N(v)$ of a vertex v is the set of vertices 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 length of some shortest 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 in this paper. An *edge cut* in G is a set of edges $[S, \bar{S}]$ such that $S \subset V(G)$ and $\bar{S} \doteq V(G) \setminus S$ are nonempty.

The *edge-connectivity* $k'(G)$ of G is the size of the smallest edge cut in G . We say that G is *k -edge-connected* if $k'(G) \geq k$.

From Menger's theorem it follows that if G is k -edge-connected, then given two vertices v, w in $V(G)$, there are at least k edge-disjoint paths joining v to w (see [West 01, pp. 153–169]).

Since we do not deal with vertex-disjoint paths in this paper, we shall hereinafter speak of k -connectivity, avoiding any reference to the edges.

2.1. An Expansion Theorem

Let G be a simple graph. Let $Q, C \subset V(G)$, and set $C' \doteq Q \cup C$, $G' \doteq G(C')$. From here on, we assume that Q and C are nonempty 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' .

Observe 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 $\rho_{C'/C}(x, z) = \rho_{C'/C}(z, y) = 1$. Note also that $\rho_{C'/C}$ is—whence the notation—the pseudodistance 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 notation

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

In this context, we also consider

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

Since Q and C will be fixed in this section, we shall freely omit the subscript 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 the following theorem.

Theorem 2.1. *If $\rho_{C'/C} \leq 2$, $[S, \bar{S}]$ is an edge cut in G' such that $C \subset S$, and we let $S_1 \doteq S \cap Q$, then we have the following:*

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 $\bar{s} \in \bar{S}$, we have $\rho(\bar{s}, S) = 1$. 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 $\rho(\bar{s}_j, S) = 1$, and thus we have k_2 new edges (G' is simple, because the vertices \bar{s}_j are distinct) $\bar{s}_j s'_j$, with $s'_j \in S$, whence

$$|[S, \bar{S}]| \geq k_1 + k_2 = |N'(\bar{s})|.$$

2. This assertion follows at once if we observe that in this case, for each $\bar{s} \in \bar{S}$, there is at least one $s \in S$ such that $\bar{s}s \in [S, \bar{S}]$.

3. Observe 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 $s \in S$, since $\rho(\bar{s}_0, s) = 2$, there exists s' such that $\rho(\bar{s}_0, s') = \rho(\bar{s}', s) = 1$. But again since $\rho(\bar{s}_0, S) = 2$, it follows that $s' \in \bar{S}$, whence $\rho(s, \bar{S}) = 1$.

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. For the last statement, we observe 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 $[S_1, \bar{S}]$. \square

Corollary 2.2. *Assume that in addition to the hypotheses of Theorem 2.1, 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. $|[\bar{S}, C]| \geq 1$.

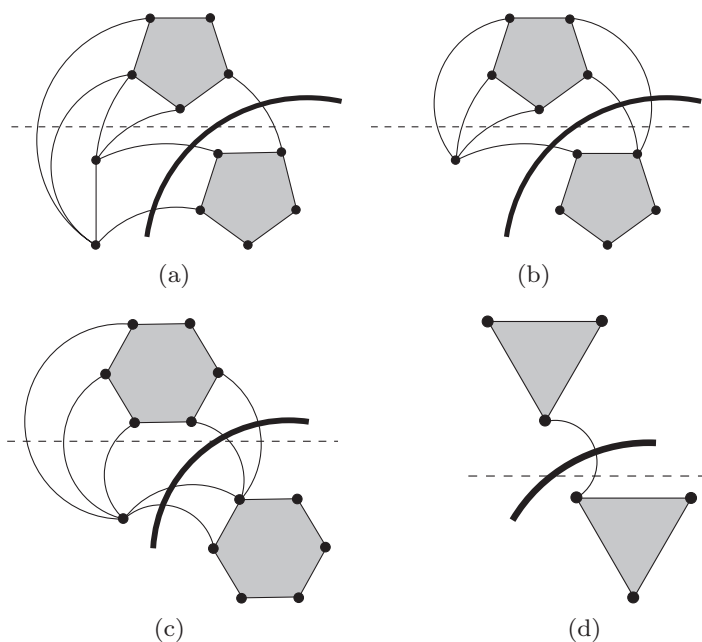


Figure 1. Conventions: 1. Filled polygons represent cliques, and curved arcs represent edges. 2. The dotted line separates C (the upper set of vertices) from Q . 3. The widest arc shows the cut $[S, \bar{S}]$. 4. $k = \min_{v \in Q} |N'(v)|$. Descriptions: (a) Here $||[S, \bar{S}]|| = 3 < k = 4$, $|S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

$$4. |S_1| < ||[S, \bar{S}]|| < \min_{v \in Q} |N'(v)| < |\bar{S}|.$$

$$5. S \cap Q \subset \partial^2 Q, \bar{S} \supset \bar{\partial}^2 Q.$$

$$6. \Phi \leq ||[S, \bar{S}]||.$$

(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 2.1. To prove point 3, observe 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 2.1, and point 3 in the present theorem. The second of these inequalities is assumed by our additional hypothesis, and the third one follows immediately from point 3 in Theorem 2.1.

From point 5 in Theorem 2.1 and $||[C, \bar{S}]|| \geq 1$ we obtain

$$|[S, \bar{S}]| > \max_{s \in \bar{S}_1} (|N'(s)| - |N'(s) \cap C|).$$

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

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

□

Corollary 2.3. *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 implies that G' is k -edge-connected:

1. $\Phi \geq k$;
2. $|\partial^1 Q| \geq k$;
3. $Q = \partial^1 Q$.

(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 since

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

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

Without loss of generality, let us we assume 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 hypotheses of Corollary 2.2.

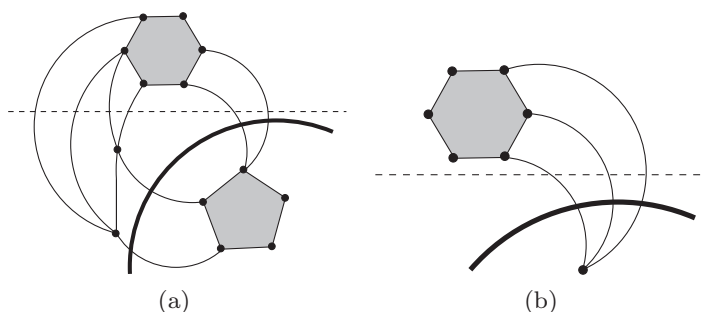


Figure 2. Conventions: 1. Filled polygons represent cliques, and curved arcs represent edges. 2. The dotted line separates C (the upper set of vertices) from Q . 3. The widest arc shows a minimal cut $[S, \bar{S}]$. Descriptions: (a) Here $||[S, \bar{S}]|| = k = 4$, $\Phi = 4$, $|\partial^1 Q| = 3$. (b) Here $||[S, \bar{S}]|| = k = 3$, $\Phi = 1$, $|\partial^1 Q| = 1$, $Q = \partial^1 Q$. This example shows that Corollary 2.3 includes an edge-connectivity version of the expansion lemma in [West 01, Lemma 4.2.3].

The first of our alternative hypotheses contradicts point 6 in the conclusions of Corollary 2.2.

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 hypotheses implies the first one.

To finish our proof, observe that if $Q = \partial^1 Q$, then since $\bar{S} \subset Q$, we have $\rho(\bar{s}, S) = 1$ for any $\bar{s} \in \bar{S}$, contradicting point 1 in the conclusions of Corollary 2.2. \square

Definition 2.4. Since this corollary is key to our later results, we shall set for future reference

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

so that now the three disjunctive hypotheses in Corollary 2.3 can be restated as $\Psi_{C'/C}(k, G) \geq 0$.

Remark 2.5. Corollary 2.3 is related to a well-known theorem of Plesník (see [Plesník 75, Theorem 6]), which states that in a simple graph of diameter 2, the edge connectivity is equal to the minimum degree.

2.2. Edge-Connectivity and Core Decomposition

Definition 2.6. [Seidman 83, Bollobás 84] 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 $\min_{v \in C} d_{G(C)}(v) \geq k$.

Let $k_{\max} \doteq \max\{k : C_k \neq \emptyset\}$, $S_k \doteq C_k \setminus C_{k+1}$, $0 \leq k \leq k_{\max} - 1$, $S_{k_{\max}} = C_{k_{\max}}$. We call S_k the k -shell of G , and if $v \in S_k$, we say that v has *shell-index* k , $k(v) = k$. The connected components of S_k will be called *clusters*.

A k -core of G can be obtained by recursively removing all the vertices of degree lower than k , together 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 03] 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 as $n(G)$.

Definition 2.7. Let $A \subset V(G)$.

1. We say that A is *k -connected* in the strict sense if $G(A)$ is k -connected, that is, if every cut in $G(A)$ has at least k edges, or equivalently, if for $u, v \in A$, there exist at least k edge-disjoint paths from u to v in $G(A)$.
2. We say that A is *k -connected in the broad 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, if given $u, v \in A$, there exist at least k edge-disjoint paths from u to v in G .

Of course, if A is k -connected in the strict sense, it is also k -connected in the broad sense. The reciprocal statement is valid only for $k = 1, 2$.

Let us introduce the following definition.

Definition 2.8. Consider a graph G , its k -cores $C_k : 0 \leq k \leq k_{\max}$ (see Definition 2.6), and $A \subset V(G)$.

1. We say that A is *k -core-connected in the strict sense* if $A \cap C_k$ is k -connected in the strict sense.
2. We say that A is *k -core-connected in the broad sense* if $A \cap C_k$ is k -connected in the broad sense in $G(C_k)$.

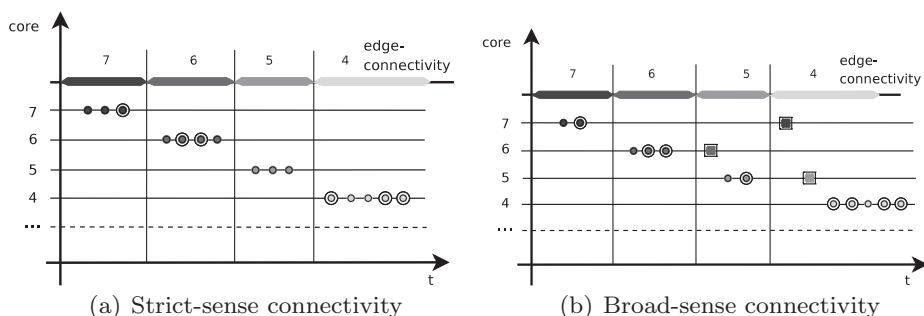


Figure 3. These two figures show the progress in time (t) of both algorithms. Dots represent clusters; dots in a circle, clusters in C ; dots in a square, clusters in D . Both algorithms start looking for a cluster of diameter 2, descending from the top shell. If they find one, this is the initial C . Then the first algorithm descends shell by shell looking for clusters to add to C and preserving core-connectivity in the strict sense. The second algorithm does the same, but each time it ends with the inspection of a shell, it searches among the omitted clusters from previous shells (saved in \mathfrak{Q}') in order to find clusters to add to D . These clusters provide more possibilities of connection (in the broad sense) to the next shell, thus producing a bigger set C .

3. We say that A is *core-connected in the strict sense* if $A \cap C_k$ is k -connected in the strict sense for all k such that $A \cap C_k \neq \emptyset$.
4. We say that A is *core-connected in the broad sense* if $A \cap C_k$ is k -connected in the broad sense in $G(C_k)$ for all k such that $A \cap C_k \neq \emptyset$.

Next, we shall describe two algorithms, Algorithms 1 and 2, that provide a mechanism to build (hopefully big) core-connected sets of vertices in both the strict and broad senses. These algorithms are illustrated in Figure 3.

Both algorithms proceed recursively, starting from the highest core and appending joinable clusters to the core-connected set. The difference between the algorithms lies in the meaning of “joinable cluster.”

When Q is a cluster, we denote by $k(Q)$ 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} .

Theorem 2.9. *Algorithm 1 always stops, and when it stops, C is core-connected in the strict sense.*

Algorithm 1. (Building core-connected sets of vertices in the strict sense.)

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

```

1.1 begin
1.2   while  $C = \emptyset$  and  $\mathfrak{Q} \neq \emptyset$  and  $k \geq 2$  do
1.3      $k \leftarrow k(\mathfrak{Q})$  ;
1.4     if there is some  $Q \in \mathfrak{Q}$  satisfying:  $\left\{ \begin{array}{l} k(Q) = k \\ \rho_Q \leq 2 \end{array} \right\}$  then
1.5        $C \leftarrow C \cup Q$  ;
1.6     end
1.7      $\mathfrak{Q} \leftarrow \mathfrak{Q} \setminus \{Q \in \mathfrak{Q} : k(Q) = k\}$  ;
1.8   end
1.9   while  $\mathfrak{Q} \neq \emptyset$  and  $k \geq 2$  do
1.10     $k \leftarrow k(\mathfrak{Q})$  ;
1.11    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
1.12       $C \leftarrow C \cup Q$  ;
1.13       $\mathfrak{Q} \leftarrow \mathfrak{Q} \setminus \{Q\}$  ;
1.14    end
1.15     $\mathfrak{Q} \leftarrow \mathfrak{Q} \setminus \{Q \in \mathfrak{Q} : k(Q) = k\}$  ;
1.16  end
1.17 end
    
```

Proof. For the first WHILE loop, step 1.3 computes the maximum k for the current family of clusters, while step 1.7 deletes from \mathfrak{Q} all clusters with shell index k . Hence, k is strictly decreasing, and when the algorithm reaches step 1.8, either \mathfrak{Q} is empty or C has a cluster satisfying the hypotheses of Corollary 2.3.

The second WHILE loop will also finish, because steps 1.10 and 1.15 ensure that k is strictly decreasing.

The nested WHILE loop will finish because \mathfrak{Q} is finite.

Assume that the current C is core-connected in the strict sense when we reach step 1.11. 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, these instances are obtained each time we come to step 1.16). The new C , let us call it C' for the moment, has the same intersections with C_j for $j > k$, and when $j \leq k$, the intersection is $C' = C \cup Q$, which is k -connected in the strict sense by Corollary 2.3, since the conditions for the selection of Q in step 1.11 match the hypotheses of Corollary 2.3. Thus, every instance of C throughout the algorithm is core-connected in the strict sense, whence the final C is core-connected in the strict sense. \square

Algorithm 2. (Building sets of core-connected vertices in the broad sense.)

Input: Ω , the family of all clusters of a graph G
Output: $C \subset V$, core-connected in the broad sense

Initialization: $C \leftarrow \emptyset$, $D \leftarrow \emptyset$, $\Omega' \leftarrow \emptyset$, $k \leftarrow k_{max}$

```

2.1 begin
2.2   while  $C = \emptyset$  and  $\Omega \neq \emptyset$  and  $k \geq 2$  do
2.3      $k \leftarrow k(\Omega)$ ;
2.4     if there is some  $Q \in \Omega$  satisfying:  $\begin{cases} k(Q) = k \\ \rho_Q \leq 2 \end{cases}$  then
2.5        $C \leftarrow C \cup Q$ ;
2.6        $\Omega \leftarrow \Omega \setminus \{Q\}$ ;
2.7     end
2.8      $\Omega' \leftarrow \Omega' \cup \{Q \in \Omega : k(Q) = k\}$ ;
2.9      $\Omega \leftarrow \Omega \setminus \{Q \in \Omega : k(Q) = k\}$ ;
2.10  end
2.11  while  $\Omega \neq \emptyset$  and  $k \geq 2$  do
2.12     $k \leftarrow k(\Omega)$ ;
2.13    while there is some  $Q' \in \Omega'$  satisfying:  $\begin{cases} k(Q') \geq k \\ \rho_{(C \cup D \cup Q')/(C \cup D)} \leq 2 \\ \Psi_{(C \cup D \cup Q')/(C \cup D)}(k, G) \geq 0 \end{cases}$  do
2.14       $D \leftarrow D \cup Q'$ ;
2.15       $\Omega' \leftarrow \Omega' \setminus \{Q'\}$ ;
2.16    end
2.17    while there is some  $Q \in \Omega$  satisfying:  $\begin{cases} k(Q) = k \\ \rho_{(C \cup D \cup Q)/(C \cup D)} \leq 2 \\ \Psi_{(C \cup D \cup Q)/(C \cup D)}(k, G) \geq 0 \end{cases}$  do
2.18       $C \leftarrow C \cup Q$ ;
2.19       $\Omega \leftarrow \Omega \setminus \{Q\}$ ;
2.20    end
2.21     $\Omega' \leftarrow \Omega' \cup \{Q \in \Omega : k(Q) = k\}$ ;
2.22     $\Omega \leftarrow \Omega \setminus \{Q \in \Omega : k(Q) = k\}$ ;
2.23  end
2.24 end

```

Theorem 2.10. *Algorithm 2 always stops, and when it stops, C is core-connected in the broad sense.*

Proof. For the first WHILE loop, step 2.3 computes the maximum k for the current family of clusters, while step 2.9 deletes from the current Ω all the clusters with shell index k . Hence k is strictly decreasing, and when the algorithm reaches step 2.10, either Ω is empty or C has a cluster satisfying the hypothesis of Corollary 2.3.

The second WHILE loop will also finish because steps 2.12 and 2.22 allow us to ensure that k is strictly decreasing.

The nested WHILE loops will finish because the families \mathfrak{Q}' and \mathfrak{Q} are finite.

Assume that the current C is core-connected in the broad sense when we reach step 2.17. 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, these instances are obtained each time we come to step 2.23). The new C , let us call it C' for the moment, has the same intersections with C_j for $j > k$, and when $j \leq k$, the intersection is $C' = C \cup Q$, which is k -connected in the broad sense by Corollary 2.3, since the conditions for the selection of Q in step 2.17 match the hypotheses of Corollary 2.3. Thus, all the instances of C during the algorithm are core-connected in the broad sense; hence the final C is core-connected in the broad sense. \square

Observe that in both algorithms, C includes nodes only from the shells with $k \geq 2$. The conditions from Corollary 2.3 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 the broad 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 on 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 01], for instance, uses a small number of BGP routers to peer with routers in other autonomous systems (AS) and thus obtain 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 98] developed *skitter* probes (which have evolved into the *Ark* infrastructure). Based on traceroute, these measuring nodes send ICMP requests to routers in order to discover paths. *Skitter* had around 24 monitors in 2003, 22 monitors in 2005, and only 11 in 2008. *Ark* project has 54 monitors in February 2011.
3. The DIMES Project [DIMES 04] is a distributed system composed of around one thousand voluntary nodes; anyone may subscribe, even those

with low CPU power or bandwidth. It explores the Internet with tools such as traceroute and ping to discover its topology.

4. Mercator [Govindan and Tangmunarunkit 00] uses hop-limited ICMP probes to discover the Internet map with *informed random* heuristics. The algorithm induces the path to traverse certain nodes using the IP source-routing option, which is no longer available.

At the autonomous systems (AS) interconnection level, data have 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 have implemented them in the LaNet-vi open source software [Alvarez-Hamelin et al. 05]. Based on the k -core decomposition, this tool computes and visualizes the strict-sense and wide-sense core-connected subgraphs. This bound is obtained by applying Corollary 2.3 for nodes in the core-connected subgraph and using the value of Φ for the others. In the LaNet-vi color visualization (black-and-white differences will be mentioned in parentheses), the node colors (node shades) show their core, while the border colors (border shades) suggest a lower bound for connectivity with other nodes. In fact, the absence of a 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) implies instead that the node is not in C but belongs to the D set in Algorithm 2, meaning that we ensure a certain level of edge-connectivity with internal clusters in C , though this bound is less than the node shell index. Finally, for white nodes (nodes in a square) we can ensure no edge-connectivity with others, and perhaps that the node is poorly connected.

We have 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

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

we studied the following quantities:

$$\begin{aligned}\alpha &= \frac{1}{k_{\max}} \sum_{k=1}^{k_{\max}} \rho_k, \\ \beta &= \frac{2}{k_{\max}(k_{\max} + 1)} \sum_{k=1}^{k_{\max}} \rho_k k, \\ \gamma &= \sum_{k=1}^{k_{\max}} \rho_k \frac{|S_k|}{|C|}.\end{aligned}$$

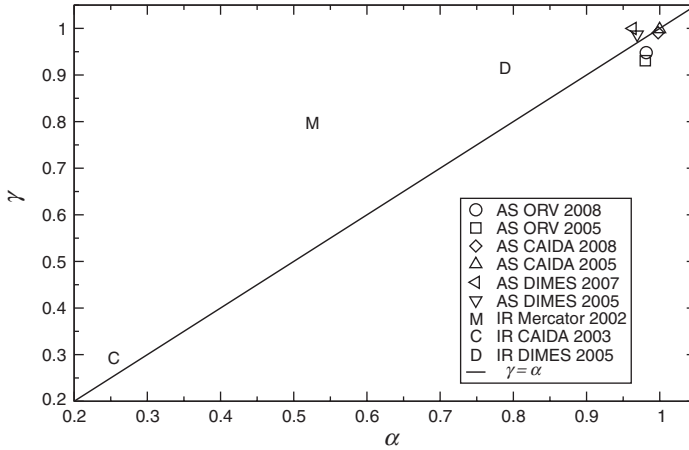


Figure 4. Comparison between γ and α . The line $\alpha = \gamma$ splits the space into two areas: the upper one for graphs with C concentrated on highly populated shells ($\gamma > \alpha$), and the bottom for C mainly in low-populated shells ($\gamma < \alpha$).

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 index; 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 the greater part of C in highly populated shells. We also present β as a function of α in Figure 5, where $\beta > \alpha$ means that a smaller part of C is found in lower-shell indices.

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. On the other hand, IR maps are worse than AS maps because the ratio of nodes in C is low. Each IR has a different behavior; the best is IR DIMES (90% of the nodes in C and in higher shells). The reason that 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 greater number of sources (thousands, according to [Dall’Asta et al. 06]). We think that a detailed picture of the Internet at the IR level will satisfy our hypotheses.

We carried out a different analysis at the shell level. In the first place, we counted pairs of nodes $\{u, v\}$ belonging to the wide-sense core-connected subgraph, i.e., having both ends in C . Figure 6 presents this information as a function of $\min(k(u), k(v))$ for the AS CAIDA 2008 map. It follows that only the first

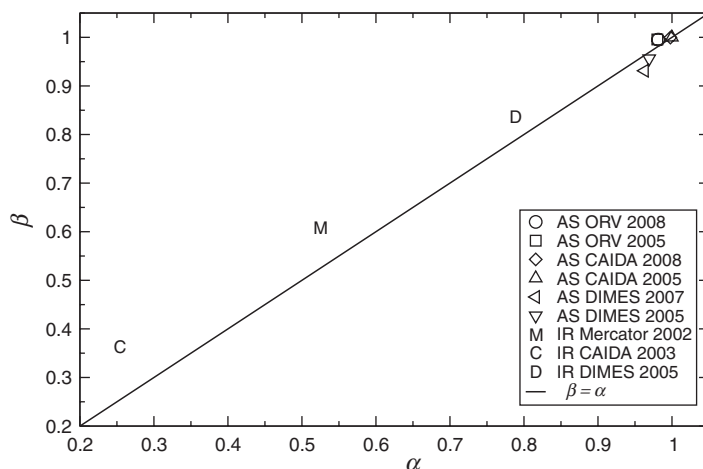


Figure 5. β vs. α for different maps.

three shells have nodes out of C , and that they are few (about 1% per shell). This behavior is similar in all maps except for IR CAIDA 2003, where some nodes in low, medium, and high shells do not belong to the core-connected subgraph.

In the second place, 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

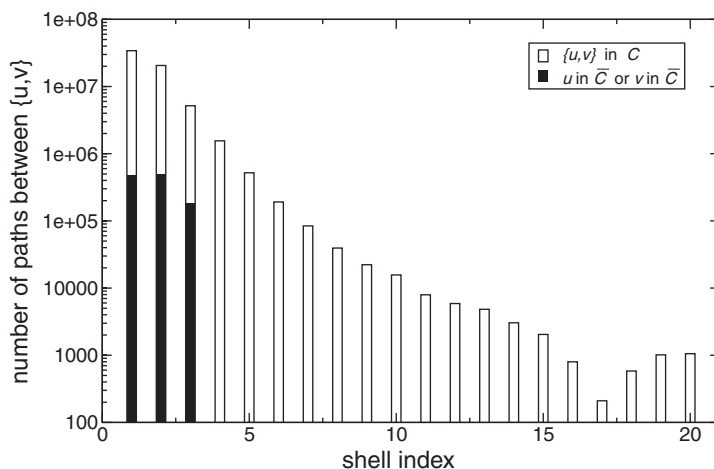


Figure 6. Histogram showing the number of paths from u to v in the AS CAIDA 2008 map. White is for $\{u, v\} \in C$, and black is for $u \in \bar{C}$ or $v \in \bar{C}$.

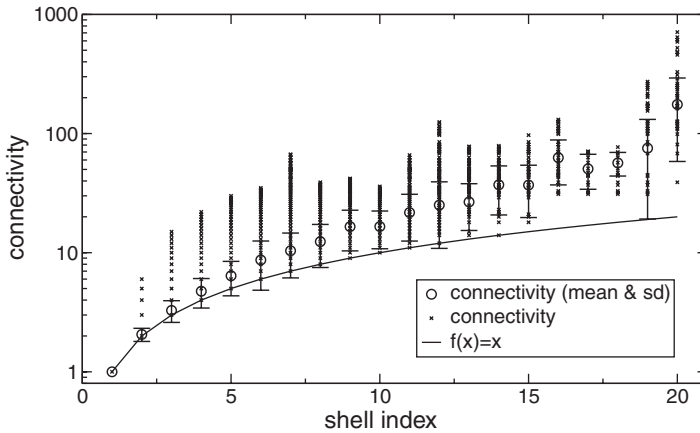


Figure 7. Connectivity for $\{u, v\}$ pairs vs. their minimum shell index. Circles stand for mean values per shell index, and error bars show the standard deviation; crosses show connectivity; the bounding line $connectivity = shell\ index$ is also displayed.

the Gomory–Hu algorithm (see [Ford and Fulkerson 62, Chapter 4]). In Figure 7 we display the connectivity of each pair $\{u, v\}$ as a function of the minimum shell index $\min(k(u), k(v))$ for the AS CAIDA 2008 map. This analysis is similar to that in [Carmi et al. 06]. We see that connectivity is higher than the bound

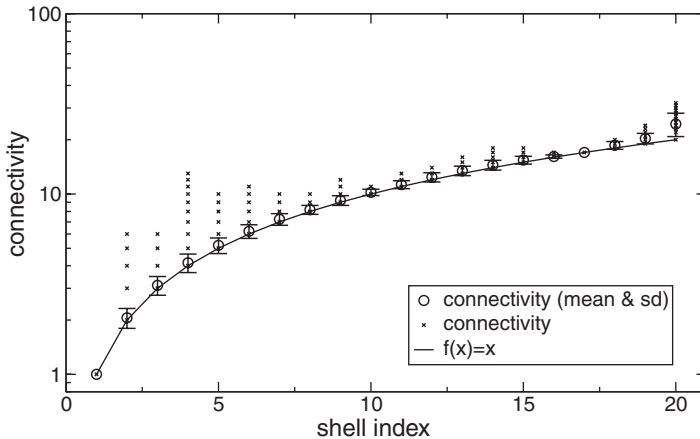


Figure 8. Connectivity through the cores for $\{u, v\}$ pairs vs. their minimum shell index. Circles stand for mean values, and error bars show the standard deviation; crosses show connectivity; the bounding line $connectivity = shell\ index$ is also displayed.

obtained through the core-connected graph. Nonetheless, the average values are relatively similar for low and medium shells (less than 100% up to shell 13). Since our bound is related to connectivity through cores (i.e., taking only the maximum k -core containing $\{u, v\}$ to find paths), we also show the connectivity through the k -core, where $k = \min(k(u), k(v))$ in Figure 8. Node-to-node connectivity is clearly larger than connectivity through the k -core, but the latter is closer to the minimum shell index. The other AS maps have analogous behavior.

We note that we were unable to compute connectivity for IR maps because it is expensive on RAM due to their size, which is greater than 100,000 nodes.

To conclude our analysis, in Figure 9 we present a gray-scale visualization for the AS CAIDA 2008 map, testing Algorithm 2.

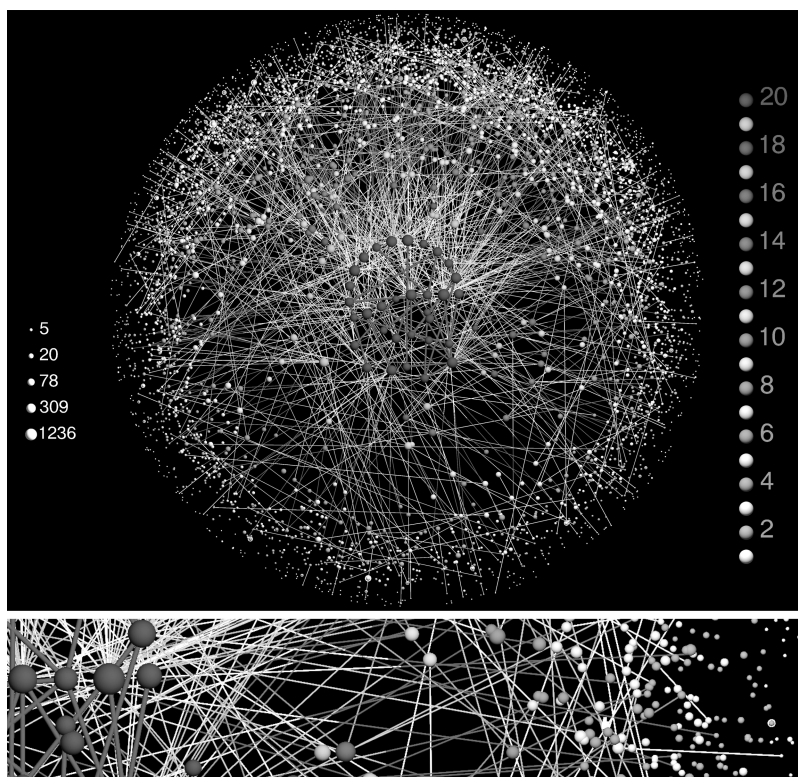


Figure 9. Gray-scale visualization of the AS CAIDA 2008 map displaying the wide-sense core-connected subgraph (Algorithm 2). Bottom: Detail showing some nodes out of the core-connected subgraph, i.e., some out of C and D appearing as small white-filled squares (on the start of the last quarter of the picture, from left to right); and one node belonging to D having a white border, on the right.

4. Conclusions

In this paper we have defined core-connected graphs, for which we find a lower bound of node-to-node connectivity between nodes: the minimum shell index of pairs of nodes in the graph k -core decomposition. In order to formalize this relation we have provided a theorem (Corollary 2.3) that gives sufficient conditions to ensure the aforementioned lower bound. This theorem is an extension of Plesník’s theorem (see [Plesník 75]):

Plesník’s theorem asserts that if a simple graph has diameter 2, then the connectivity is at least $\delta(G)$. The hypothesis of G being simple can be easily relaxed: for general multigraphs the connectivity is at least $\min_{v \in V} |N(v)|$ (which is $\delta(G)$ in the simple case). The combinatorial nature of connectivity, in which multiple bifurcations give rise to a great multiplicity of paths between two vertices, in the absence of bottlenecks, justifies the presumption that the condition of diameter 2 is rather artificial. In fact, the authors of [Carmi et al. 06, Carmi et al. 07] have noticed that the node-to-node connectivity $k'(u, v)$ is at least $\min(k(u), k(v))$ 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 either edges or 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 its next $(k - 1)$ -shell. Rather than an alternative procedure to find the connectivity, which can be quite efficiently found with the Gomory–Hu algorithm (see [Ford and Fulkerson 62, Chapter 4]), we hope that our results give some new insight into the local–global relations for connectivity, useful in real-life net graphs.

We also developed two algorithms to obtain core-connected subgraphs of a given graph G : one for strict-sense connectivity (whose complexity is $O(e)$) and the other for wide-sense connectivity ($O(e \times \sqrt{e})$). In the strict-sense algorithm each cluster is considered only once to determine whether it fulfills conditions about its diameter and Φ . With $e(Q)$ the number of edges of cluster Q counting the internal ones and those that connect it with inner cores (clusters do not share edges with other clusters in the same core), it can be determined in $O(e(Q))$ whether the cluster’s diameter is less than or equal to 2. The Φ condition can also be evaluated in $O(e(Q))$, since 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 in order to obtain a k -shell, $k + 1$ nodes are needed with at least 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 have 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 Internet maps.

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

Acknowledgments. This work was partially funded by UBACyT I413/2008 grant. M. G. Beiró acknowledges a Peruilh fellowship.

References

- [Aiello et al. 00] W. Aiello, F. Chung, and L. Lu. “A Random Graph Model for Power Law Graphs.” *Experimental Math.* 10 (2000), 53–66.
- [Alvarez-Hamelin et al. 05] J. I. Alvarez-Hamelin, M. G. Beiró, L. Dall’Asta, A. Barrat, and A. Vespignani. “Large Network Visualization Tool.” Available online (<http://sourceforge.net/projects/lanet-vi>), 2005.
- [Alvarez-Hamelin et al. 06] J. I. Alvarez-Hamelin, L. Dall’Asta, A. Barrat, and A. Vespignani. “Large Scale Networks Fingerprinting and Visualization Using the k -Core Decomposition.” In *Advances in Neural Information Processing Systems 18*, edited by Y. Weiss, B. Schölkopf, and J. Platt, pp. 41–50. Cambridge, MA: MIT Press, 2006.
- [Alvarez-Hamelin et al. 08] J. I. Alvarez-Hamelin, L. Dall’Asta, A. Barrat, and A. Vespignani. “ k -Core Decomposition of Internet Graphs: Hierarchies, Self-Similarity and Measurement Biases.” *Networks and Heterogeneous Media* 3:2 (2008), 371–393.
- [Andersson and Asati 09] L. Andersson and R. Asati. “RFC 5462: Multiprotocol Label Switching (MPLS) Label Stack Entry: ‘EXP’ Field Renamed to ‘Traffic Class’ Field.” Available online (<ftp://ftp.rfc-editor.org/in-notes/rfc5462.txt>), 2009.
- [Barabási and Albert 99] A. L. Barabási and R. Albert. “Emergence of Scaling in Random Networks.” *Science* 286 (1999), 509–512.
- [Batagelj and Zaversnik 03] V. Batagelj and M. Zaversnik. “An $O(m)$ Algorithm for Cores Decomposition of Networks.” *CoRR*, arXiv.org/cs.DS/0310049, 2003.
- [Beiró et al. 08] M. G. Beiró, J. I. Alvarez-Hamelin, and J. R. Busch. “A Low Complexity Visualization Tool That Helps to Perform Complex Systems Analysis.” *New J. Phys.* 10:12 (2008), 125003.
- [Bollobás 84] B. Bollobás. “The Evolution of Sparse Graphs.” *Graph Theory and Combinatorics: Proc. Cambridge Combinatorial Conf. in honors of Paul Erdős*, edited by B. Bollobás, pp. 35–57. New York: Academic Press, 1984.
- [Bollobás 04] B. Bollobás. *Handbook of Graphs and Networks*. Berlin: Wiley, 2004.
- [Bollobás et al. 07] B. Bollobás, S. Janson, and O. Riordan. “The Phase Transition in Inhomogeneous Random Graphs.” *Random Struct. Algorithms* 31:1 (2007), 3–122.

- [CAIDA 98] Cooperative Association for Internet Data Analysis (CAIDA). “Router-Level Topology Measurements.” Available online (<http://www.caida.org/tools/measurement/skitter/>), 1998.
- [Carmi et al. 06] S. Carmi, S. Havlin, S. Kirkpatrick, Y. Shavitt, and E. Shir. “MEDUSA—New Model of Internet Topology Using k -Shell Decomposition.” Technical Report 0601240v1, arxiv, 2006.
- [Carmi et al. 07] S. Carmi, S. Havlin, S. Kirkpatrick, Y. Shavitt, and E. Shir. “A Model of Internet Topology Using k -Shell Decomposition.” *PNAS* 104 (2007), 11150–11154.
- [Dall’Asta et al. 06] L. Dall’Asta, J. I. Alvarez-Hamelin, A. Barrat, A. Vázquez, and A. Vespignani. “Exploring Networks with Traceroute-like Probes: Theory and Simulations.” *Theor. Comput. Sci.* 355:1 (2006), 6–24.
- [DIMES 04] Distributed Internet Measurements and Simulations (DIMES). Available online (<http://www.netdimes.org/>), 2004.
- [Erdős and Rényi 59] P. Erdős, and A. Rényi. “On Random Graphs I.” *Publ. Math. (Debrecen)* 6 (1959), 290–297.
- [Faloutsos et al. 99] M. Faloutsos, P. Faloutsos, and C. Faloutsos. “On Power-Law Relationships of the Internet Topology.” In *SIGCOMM*, pp. 251–262, 1999.
- [Ford and Fulkerson 62] L. R. Ford and D. R. Fulkerson. *Flows in Networks*. Princeton: Princeton University Press, 1962.
- [Govindan and Tangmunarunkit 00] R. Govindan and H. Tangmunarunkit. “Heuristics for Internet Map Discovery.” In *IEEE INFOCOM 2000*, pp. 1371–1380. Tel Aviv: IEEE, 2000.
- [Janson 09] S. Janson. “On Percolation in Random Graphs with Given Vertex Degrees.” *Electron. J. Probab.* 14:5 (2009), 87–118.
- [Newman et al. 01] M. E. Newman, S. H. Strogatz, and D. J. Watts. “Random Graphs with Arbitrary Degree Distributions and Their Applications.” *Phys. Rev. E Stat. Nonlin. Soft Matter Phys.* 64:2 Pt. 2 (2001), 026118.
- [ORV 01] University of Oregon Route Views (ORV) Project. Available online (<http://www.routeviews.org/>), 2001.
- [Pastor-Satorras et al. 01] R. Pastor-Satorras, A. Vázquez, and A. Vespignani. “Dynamical and Correlation Properties of the Internet.” *Phys. Rev. Lett.* 87:25 (2001), 258701.
- [Plesník 75] J. Plesník. “Critical Graphs of a Given Diameter.” *Acta Fac. Rerum Natur. Univ. Comenian. Math.* 30 (1975), 71–93.
- [Riordan 08] O. Riordan. “The k -Core and Branching Processes.” *Comb. Probab. Comput.* 17:1 (2008), 111–136.
- [Rosen et al. 01] E. Rosen, A. Viswanathan, and R. Callon. “RFC 3031: Multiprotocol Label Switching Architecture.” Available online (<http://www.rfc-editor.org/rfc/rfc3031.txt>), 2001.
- [Seidman 83] S. B. Seidman. “Network Structure and Minimum Degree.” *Social Networks* 5 (1983), 269–287.

- [Serrano et al. 06] M. A. Serrano, M. Boguñá, and A. Díaz-Guilera. “Modeling the Internet.” *Eur. Phys. J. B* 50 (2006), 249–254.
- [Spatharis et al. 09] A. Spatharis, I. Foudalis, M. Sideri, and C. Papadimitriou. “Comparing Trade-Off Based Models of the Internet.” *Fundam. Inf.* 92:4 (2009), 363–372.
- [Waxman 88] B. M. Waxman. “Routing of Multipoint Connections.” *IEEE Journal on Selected Areas in Communications* 6:9 (1988), 1617–1622.
- [West 01] D. B. West. *Introduction to Graph Theory*. Englewood Cliffs, NJ: Prentice Hall, 2001.
- [Zegura et al. 97] E. W. Zegura, K. L. Calvert, and M. J. Donahoo. “A Quantitative Comparison of Graph-Based Models for Internet Topology.” *IEEE/ACM Transactions on Networking* 5:6 (1997), 770–783.

J. Ignacio Alvarez-Hamelin, INTECIN, Facultad de Ingeniería, Universidad de Buenos Aires, Paseo Colón 850, C1063ACV Buenos Aires, Argentina (ignacio.alvarez-hamelin@cnet.fi.uba.ar)

Mariano G. Beiró, Facultad de Ingeniería, Universidad de Buenos Aires, Paseo Colón 850, C1063ACV Buenos Aires, Argentina (mbeiro@fi.uba.ar)

Jorge R. Busch, Facultad de Ingeniería, Universidad de Buenos Aires, Paseo Colón 850, C1063ACV Buenos Aires, Argentina (jbusch@fi.uba.ar)

Received May 27, 2010; accepted June 26, 2010.