REGULAR ARTICLE

Router-level community structure of the Internet Autonomous Systems

Mariano G Beiró[1,2*] , Sebastián P Grynberg[2] and J Ignacio Alvarez-Hamelin[2,3,4]

*Correspondence:
mariano.beiro@isi.it
[1] ISI Foundation, Turin, Italy
Full list of author information is
available at the end of the article

Abstract

The Internet is composed of routing devices connected between them and organized into independent administrative entities: the Autonomous Systems. The existence of different types of Autonomous Systems (like large connectivity providers, Internet Service Providers or universities) together with geographical and economical constraints, turns the Internet into a complex modular and hierarchical network. This organization is reflected in many properties of the Internet topology, like its high degree of clustering and its robustness.

In this work we study the modular structure of the Internet router-level graph in order to assess to what extent the Autonomous Systems satisfy some of the known notions of community structure. We observe that most of the classical community detection methods fail to detect the Autonomous Systems as communities, mainly because the modular structure of the Internet (as that of many complex networks) is much richer than what can be captured by optimizing a global functional: Autonomous Systems have largely variable sizes, structures and functions. Classical methods are severely affected by resolution limits and by the heterogeneity of the communities; even when using multiresolution methods, there is no single resolution at which most of the communities can be captured.

However, we show that multiresolution methods do find the community structure of the Autonomous Systems, but each of them has to be observed at the correct resolution level. Then we develop a low-complexity multiresolution modularity optimization algorithm that finds communities at different resolution levels in a continuous scale, in one single run. Using this method, we show that with a scarce knowledge of the node affiliations, multiresolution methods can be adjusted to retrieve the Autonomous Systems, significantly improving the results of classical single-resolution methods. Finally, in the light of our results, we discuss recent work concerning the use of a priori information to find community structure in complex networks.

Keywords: Internet topology; community structure; autonomous systems; complex networks

1 Introduction

The Internet is a complex network composed of routing devices (*routers*) which are organized into administrative entities, the *Autonomous Systems (ASes)*. Each AS has its own routing policies and design criteria, which are based on technological, geographical and economical constraints and aim at maximizing the performance in terms of bandwidth,

delay and resilience. ASes are nowadays identified with large carriers, ISPs (Internet Service Providers), IXPs (Internet Exchange Points), CDNs (Content Delivery Networks) and universities, among others, and they are interconnected as the result of commercial agreements between them, either as peers or in provider-customer relationships.

ASes can also be classified into core ASes (once called Tier-1's), and peripheral ASes. The core ASes are typically the large carriers and some CDNs and are densely connected between them. Instead, most ISPs lay in the periphery and are connected to one or more core ASes. This core-periphery division provides the Internet with an interesting hierarchy. In fact, many large-scale properties as the resilience, scalability and small-worldness of the Internet arise largely as a consequence of its hierarchical structure and self-organization [1, 2]. Understanding them might help developing better models of the Internet and improving its performance, by optimizing routing or reducing congestion, for example.
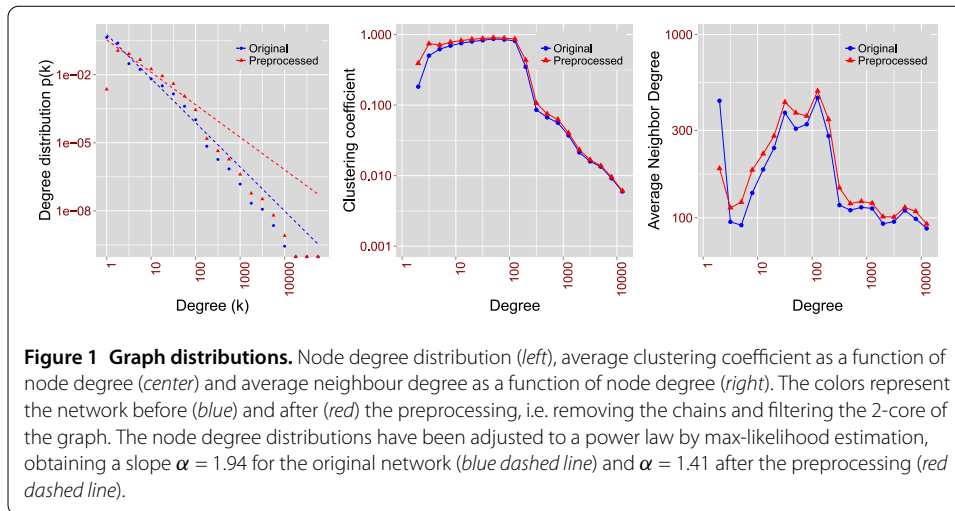
In this work we approach the modular structure of the Internet at the router level by trying to identify the ASes as communities of nodes. We explore different notions of community structure, and we shall see that: (a) The goodness of the communities is in some way related to the core-periphery hierarchy of the ASes. (b) Though most of the ASes satisfy some notion of community, their largely variable sizes represent a hard problem for many classical community detection algorithms. (c) Using multiresolution methods instead, the ASes can be retrieved if each of them is observed at the right resolution level. Finally, we shall propose a method for identifying ASes using samples of nodes.

Our work falls into the study of Internet topology. Some of the main results on this area can be found in [3]. In particular, the study of community structure of the Internet at the Autonomous Systems level has been previously approached in [4–7] and, to the best of our knowledge, the community structure at the router level has only been addressed in [8] in the context of traffic queuing analysis. For the community detection problem in general, we address the reader to the survey by Fortunato [9].

The paper is organized as follows: In Section 2 we describe the Internet exploration that we used. In Section 3 we apply several community detection algorithms to it, matching the obtained communities to the ASes; we also introduce a hierarchical partitioning algorithm based on multiresolution modularity maximization and we determine which is the best resolution for each AS; we also evaluate other multiresolution algorithms. In Section 4 we evaluate the results from the Internet topology perspective, explaining their variability: why some ASes satisfy our notion of community while others do not. We also discuss the methodological problems of community detection in heterogeneous networks and some possible solutions. Finally, in Section 5 we extract the main results of this work.

2 Dataset

The dataset used throughout this paper was provided by CAIDA [10]. The CAIDA association performs daily explorations of the Internet by sending periodic IP packets (*probes*) towards random destinations from a set of sources called *monitors* (which nowadays count around 100). These probes are sent with a `traceroute` tool which exploits the ICMP functionality: by means of the ICMP protocol, the intermediate routers in a datagram transmission can provide the source with some information on the traversed path. Using this information, the CAIDA infrastructure builds a map of the Internet at the router level. Several problems must be resolved through this process, such as IP aliasing (the association of several IP addresses with one same router) [11] and the existence of anonymous

Figure 1 Graph distributions. Node degree distribution (*left*), average clustering coefficient as a function of node degree (*center*) and average neighbour degree as a function of node degree (*right*). The colors represent the network before (*blue*) and after (*red*) the preprocessing, i.e. removing the chains and filtering the 2-core of the graph. The node degree distributions have been adjusted to a power law by max-likelihood estimation, obtaining a slope $\alpha = 1.94$ for the original network (*blue dashed line*) and $\alpha = 1.41$ after the preprocessing (*red dashed line*).

routers (routers which do not answer probes). Traceroute-based sampling is not a perfect sampling and has several biases [12]: some links are more frequently visited than others, and some links are probably not visited at all. This, together with the fact that the Internet is a fast evolving network, turns the map into a partial, approximate view of the Internet.

However, router-level maps of the Internet have several applications, like improving routing algorithms by taking into account the network topology, understanding information propagation, and studying the robustness and scale-free properties of networks [1, 13]. They can also be used for constructing maps of the Internet at the Autonomous Systems level [14, 15].

In this work we used the CAIDA router-level Internet map from October 19th, 2011 [16]. From the raw data we obtained an undirected graph consisting on 3,248,358 nodes (the routers) connected by 14,083,946 edges; 81.6% of the nodes also contain information about their Autonomous System (AS) affiliation (for a description on how this information was inferred, consult [17] and the CAIDA page [18]). Figure 1 plots the degree distribution and the average clustering coefficient and neighbour degree as a function of the node degree.

2.1 Graph preprocessing

The graph contains some low-degree structures which make it difficult to analyse its community structure without some preprocessing. Regarding the AS affiliation, for example, the dataset contains 21,108 different ASes, of which 8,023 have less than 5 nodes, and 865 ASes contain 1 single node of degree 1. We detected the presence of many *tendrils* (sequences of degree-2 nodes finishing with a degree-1 node) involving 1,393,846 nodes. These nodes do not take part in any triangle or clustered structure of the network, and this is probably due to a limitation on the exploration. However, they can be considered to belong to the community of the first node with higher degree they are attached to, so we removed these nodes by computing the 2-core of the graph [19].
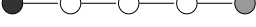
We also observed many sequences of degree-2 nodes finishing with higher degree nodes in their endpoints; we call these sequences *chains*. We found a negative correlation of −0.7 between the events *having degree 2* and *not having AS affiliation*, as shown in Table 1; in fact, 88.9% of nodes without AS affiliation belong to chains, while 97.2% of the remaining

Table 1 Correlation between having degree 2 and having AS affiliation

*degree*2	AS	
	True	False
True	0.12	0.16
False	0.70	0.02

The table shows the joint distribution of the indicator variables degree2:*having degree 2* and AS:*having AS affiliation*.

Table 2 Most common cases of chains of routers

Chain	Number of cases	Typical configuration
●—●—●	193,398	Internal AS connection
●—○—●	90,307	Internal AS tunnel
●—○—◐	86,775	Inter-AS tunnel
●—○—○—◐	57,988	Inter-AS tunnel
●—○—○—●	32,473	Internal AS tunnel
●—○—○—○—◐	28,774	Inter-AS tunnel

Black and gray colours represent some hypothetical ASes; white nodes do not have AS affiliation. Many of these chains are caused by tunnels (like MPLS tunnels); others are probably due to a tie in the AS affiliation algorithm. After removing these chains, 99.2% of the nodes have AS affiliation.

nodes do contain AS affiliation. Some of the chains are due to the presence of *MPLS tunnels* [20] or other types of tunnels: 97.5% of nodes involved in chains and without AS affiliation have private IP addresses (i.e. their interfaces did not answer the traceroute probes). We could classify 64.7% of the chains as internal to the ASes, and 33.8% of them as inter-AS chains.

Thus, we decided to clear the graph by restricting the analysis to the 2-core and replacing all the chains by an edge between their endpoints. We found 531,625 chains of different types, as resumed in Table 2. The resulting graph after this process has 1,119,672 nodes and 11,742,947 edges. 99.2% of the nodes have AS affiliation now, and just 15,288 different ASes are found. The graph distributions after this preprocessing are shown in red in Figure 1. We observe that low degree nodes tend to have a higher clustering now, while the removal of chains changed the average neighbor degree for nodes of low degree.

3 Analysis methods and results

In a first stage, we analysed the community structure of the Internet router-level graph by applying several community detection methods:

- Infomap, by Rosvall and Bergstrom, based in the description length [21].
- LPM, the Label Propagation Method by Raghavan et al. [22], which performs a diffusion process on the graph.
- Deltacom, an efficient greedy algorithm for modularity optimization introduced here.
- Louvain, a fast modularity optimization algorithm [23].
- CommUGP, a local community detection method [24].

In the following subsection we shall introduce Deltacom, an algorithm for hierarchical partitioning based on modularity maximization. Later on, we shall evaluate the results using a similarity metric.

3.1 A hierarchical partitioning method based on multiresolution modularity

Deltacom is based on the optimization of modularity. We recall that modularity was introduced by Newman in [25, 26] and, since then, several methods for its maximization have been proposed, like the one by Guimerà and Amaral based on simulated annealing [27], the extremal optimization method by Duch and Arenas [28], the fast greedy algorithm by Blondel et al. [23], or the multilevel algorithm by Noack and Rotta [29].

The following two results about modularity are fundamental to our present work: in [30] Fortunato and Barthélemy showed that modularity has a resolution limit, i.e., its maximization tends to put small communities together when they are connected among them; in [31] Reichardt and Bornholdt observed that modularity can be understood as the Hamiltonian of a ferromagnetic Potts model. Thus, maximizing the modularity implies finding the ground-state of this model, and the authors developed a simulated annealing based procedure for doing it.

Deltacom considers the optimization of the modularity Q as a particular case of the optimization of a more general functional Q_t, with a resolution parameter t, in which modularity corresponds to a normalized resolution, i.e., $t = 1$. The source of our idea can be traced back to the γ resolution parameter in [31], and has also been followed by Lancichinetti and Fortunato in [32]. In [31], the maximization is based on simulated annealing, it must run at one single resolution each time, and its computational complexity is very high. The advantage of our method is that the resolution evolves dynamically, so that all the partitions at different resolutions can be produced during one single run, and with a low computational complexity.

Deltacom is based on an agglomerative greedy algorithm; each step of the agglomerative process is a local maximum for Q_t, i.e., the generalized modularity, at some particular resolution t. In other words, for each t-value a community partition is found, which is locally t-optimal in some sense. As these communities are joined, the resolution shrinks and t gets smaller. For $t = 1$, the structure is a local maximum for the classical modularity. In other words, we observe the graph as by using a magnifying glass, obtaining partitions at every resolution level. The result is a hierarchical structure in which partitions at higher resolution levels are always refinements (in a mathematical sense) of those at lower ones.

Here we present a brief description of how the algorithm works. Further mathematical details on the properties of Q_t and of our maximization algorithm can be found in [33].

3.1.1 Newman's modularity

Given a graph $G = (V, E)$ and a partition of it, $\mathscr{C}$, the modularity of the partition is defined as [26]:

$$Q(\mathscr{C}) = \frac{1}{2m} \sum_{(i,j) \in V} \left[A_{ij} - \frac{k_i k_j}{2m} \right] \delta(i, j), \tag{1}$$

where m is the amount of edges, $A = (A_{ij})$, $i, j \in V$ is the adjacency matrix of the graph, k_i is the degree of the node i and $\delta(i, j)$ is the indicator function that points out if nodes i and j belong to the same community in the partition $\mathscr{C}$.

The expression on the right side of Eq. 1 compares the internal connections in the communities (represented by the $A_{ij} \cdot \delta(i,j)$ product) with the expected number of connections under a random graph with the same expected degrees in the nodes and the same communities. It can be restated as a sum throughout the communities, in the following way:

$$Q(\mathscr{C}) = \frac{1}{2m} \sum_{C \in \mathscr{C}} \left[e_C - \frac{k_C^2}{2m} \right], \tag{2}$$

where k_C is the sum of degrees of the nodes in each community C, and e_C is twice the number of internal edges of it.

Now, in many agglomerative clustering methods, communities are joined by pairs at each step, until finding a local maximum. Joining two communities C and C' during the process produces the following variation in the Q functional:

$$\Delta Q = \frac{1}{2m} \left(e(C,C') - \frac{k_C \cdot k_{C'}}{2m} \right), \tag{3}$$

where $e(C,C')$ is twice the number of edges between C and C'.

3.1.2 Introducing a resolution parameter

We introduce a resolution parameter t as

$$Q_t(\mathscr{C}) = \frac{1}{2m} \sum_{C \in \mathscr{C}} \left[e_C - t \cdot \frac{k_C^2}{2m} \right]. \tag{4}$$

Now the functional variation after joining two communities C and C' becomes:

$$\Delta Q_t = \frac{1}{2m} \left(e(C,C') - t \cdot \frac{k_C \cdot k_{C'}}{2m} \right). \tag{5}$$

It is clear that a positive ΔQ_t value for a particular t also implies a positive (and even higher) $\Delta Q_{t'}$ for any resolution $t' < t$. In other words, any agglomerative process which monotonically increases ΔQ_t also serves as a process monotonically increasing $\Delta Q_{t'}$ for any $t' < t$.

It is also immediate that a very large t value discourages any join, because ΔQ_t is negative for every pair of communities. It can also be shown that for t larger enough the global maximum for Q_t would have each node isolated in its own community.

Thus, what we propose is to start with a large enough t (so that the optimal initial partition $\mathscr{C}_t$ has as many communities as nodes) and then to decrease t as little as possible so that a join can be made, i.e., so that some ΔQ_t becomes positive. This t value is just

$$t(\mathscr{C}) = \max_{C,C' \in \mathscr{C}} \left\{ \frac{e(C,C') \cdot 2m}{k_C \cdot k_{C'}} \right\}. \tag{6}$$

When a local maximum is found for some t (i.e., no pair of communities can be joined without decreasing ΔQ_t), we decrease the resolution to t' such that $\Delta Q'_t$ becomes positive for some pair of communities, following the previous formula. The agglomerative process continues until obtaining as many communities as connected components of the graph,

---

Algorithm 1: Deltacom: Multiresolution modularity optimization

Input: A graph $G = (V, E)$

Output: A set of weakly optimal partitions $\mathscr{C}_t$ for every resolution value $t_{\min} \leq t \leq t_{\max}$, such that $\mathscr{C}_{t'} \prec \mathscr{C}_t$ whenever $t' < t$

1.1 **begin**

1.2 $\mathscr{D} = \{\{v\}, v \in V\}$ `#Initial partition`

1.3 $t_{\max} = \max_{C,C' \in \mathscr{D}} \{\frac{e(C,C') \cdot 2m}{k_C \cdot k_{C'}}\}$ `#Initial t`

1.4 $t = t_{\max}$

1.5 $\mathscr{C}_t = \mathscr{D}$

1.6   **while** $|\mathscr{D}| > 1$ **do**

1.7   **while** *there exists a pair* (C, C') *in* $\mathscr{D}$ *such that* $\Delta Q_t(C, C') = 0$ **do**

1.8 $\mathscr{D} = \mathscr{D} \setminus \{C, C'\} \cup \{C \cup C'\}$

1.9   **end**

1.10   **if** $|\mathscr{D}| > 1$ **then**

1.11 $t' = \max_{C,C' \in \mathscr{D}} \{\frac{e(C,C') \cdot 2m}{k_C \cdot k_{C'}}\}$ `#Decreasing t`

1.12 $\mathscr{C}_{(t',t]} = \mathscr{D}$ `#Output partition for the range` $(t', t]$ `is finished`

1.13   **else**

1.14 $t' = t$ `#In the last step we cannot decrease t`

1.15 $\mathscr{C}_t = \mathscr{D}$ `#Last partition; composed of one large community`

1.16   **end**

1.17 $t = t'$

1.18   **end**

1.19 $t_{\min} = t$ `#Minimum resolution; all the graph into one large community`

1.20 **end**

---

and the obtained result is a set of locally optimal partitions $\mathscr{C}_t$ for every resolution t, and such that the finer partitions are refinements of the coarser-grained ones. Even more, these partitions are what we call *weakly optimal* partitions, in the sense that not only the join of any pair of communities C, C' would decrease Q_t, but also any coarser partition (i.e. obtained by joining its communities in any way) would also decrease it.

The code for the Deltacom algorithm is freely available at SourceForge [34]; given a graph as a list of edges, it produces a set of community partitions for every resolution value.

The formalization of the process is contained in Algorithm 1. This algorithm produces weakly optimal partitions for the generalized modularity Q_t. In particular, $\mathscr{C}_1$ is a weakly optimal partition for the classical modularity Q. The notation $\mathscr{C}_{(a,b]}$ represents that the partition is weakly optimal for every resolution in the $(a, b]$ interval. For a complexity analysis, we shall consider keeping a set of lists L_C, one for each community, containing its neighbour communities C' and the edge cut with each of them. The lists shall be ordered by a community identifier. The initial construction of these structures has time complexity $\mathcal{O}(m \cdot \log(m))$ (as each list has size s_C, with $\sum_C s_C = m$, and $\sum_C s_C \cdot \log(s_C) \leq m \cdot \log(m)$). The `while` loop starting at line 1.6 has at most n cycles; each of these cycles has three

steps: (a) finding a pair of communities with $\Delta(Q_t) = 0$; (b) joining the communities; and (c) decreasing t. The first step implies traversing all the lists; each $\Delta(Q_t)$ computation is $\mathcal{O}(1)$ so this step's complexity can be bounded to $\mathcal{O}(m)$. Joining two communities implies merging their lists into one, and also updating the lists of their neighbour communities. As all the lists are ordered, this can be done by traversing those lists, which we can bound to $\mathcal{O}(m)$. Finally, decreasing t implies traversing all the lists and computing $\frac{e(C,C') \cdot 2m}{k_C \cdot k_{C'}}$ for each element. This is again $\mathcal{O}(m)$. Putting all together, we have at most n cycles with $\mathcal{O}(m)$. The final complexity is then $\mathcal{O}(nm)$.

While $\mathcal{O}(nm)$ is a theoretical upper bound for a general graph, the practical running time for sparse graphs can be greatly reduced with some considerations: step (c) gives all the information for the next step (a), because the community pair which maximized t' in line 1.11 is the same that will have $\Delta Q_t = 0$ in the next iteration; also, by keeping a search structure like an ordered tree with all the pair of connected communities C, C' ordered by $\frac{e(C,C') \cdot 2m}{k_C \cdot k_{C'}}$ decreasingly, there is no need to traverse all the lists in each iteration, but we must just update the modified values into this tree and choose the community pair at its head as the one maximizing t'. For sparse graphs, this process has an upper complexity bound of $\mathcal{O}(m \cdot \log(m) \cdot s_C^{\max})$, where $s_C^{\max}$ is the maximum number of neighbour communities that a community may have at any time of the process.

3.2 Finding ASes through similarity maximization

We applied the five forementioned algorithms (Infomap, LPM, Deltacom at $t = 1$, Louvain and CommUGP) to the CAIDA dataset. Each of these methods produces a partition $\mathcal{C}$ of the graph. In order to determine if the communities in these partitions are related to the ASes, we shall use this criterion: for each AS in the graph, and given a partition $\mathcal{C}$, we find the community $C \in \mathcal{C}$ which maximizes the *Jaccard similarity*:

$$J(C, AS) = \frac{|C \cap AS|}{|C \cup AS|}.$$

This most similar community shall be called $C_{AS}^{(1)}$:

$$C_{AS}^{(1)} = \underset{C \in \mathcal{C}}{\arg\max}\, J(C, AS).$$

And this maximum Jaccard similarity value will be called the *recall score* of the AS (according to [35]) and noted as $R_1(AS)$:

$$R_1(AS) = J\big(C_{AS}^{(1)}, AS\big). \tag{7}$$

In this way, we can evaluate each of the methods by observing the empirical cumulative distributions of the recall score of the ASes, $R_1(AS)$ (Eq. 7). Those methods with left-skewed cumulative distributions (i.e., recall values closer to 1.00) tend to better reproduce the Autonomous System structure. The cumulative distributions for the different methods are shown in Figure 2: *blue* for Infomap, *yellow* for LPM, *red* for Deltacom, *green* for CommUGP and *pink* for Louvain. At the top we plot the cumulative distribution considering all the ASes, and at the bottom we only consider ASes with at least 100 nodes (i.e., 639 ASes, which cover 90% of the network).
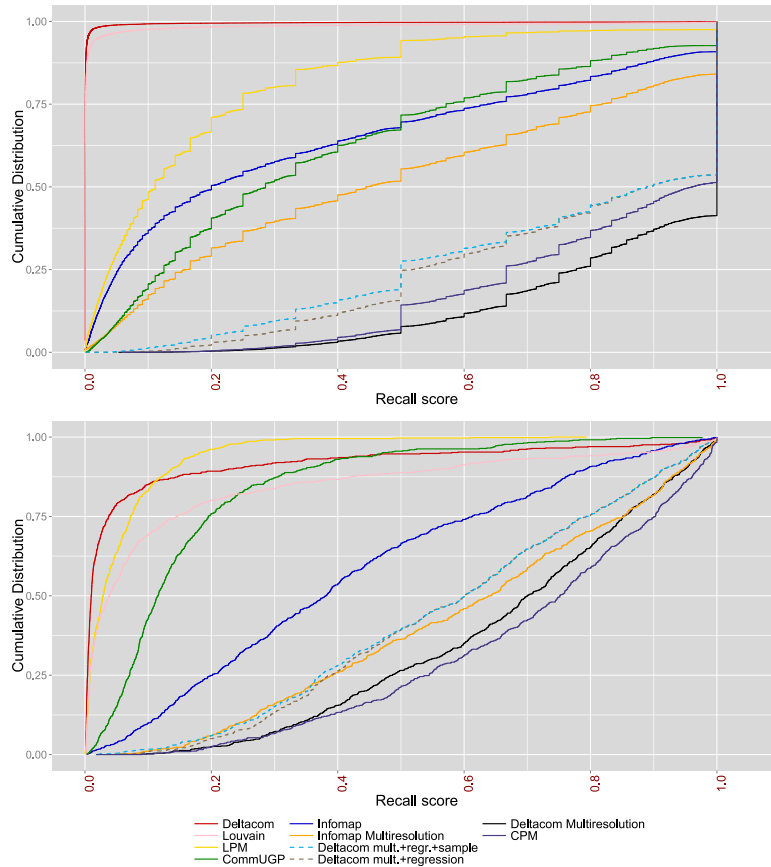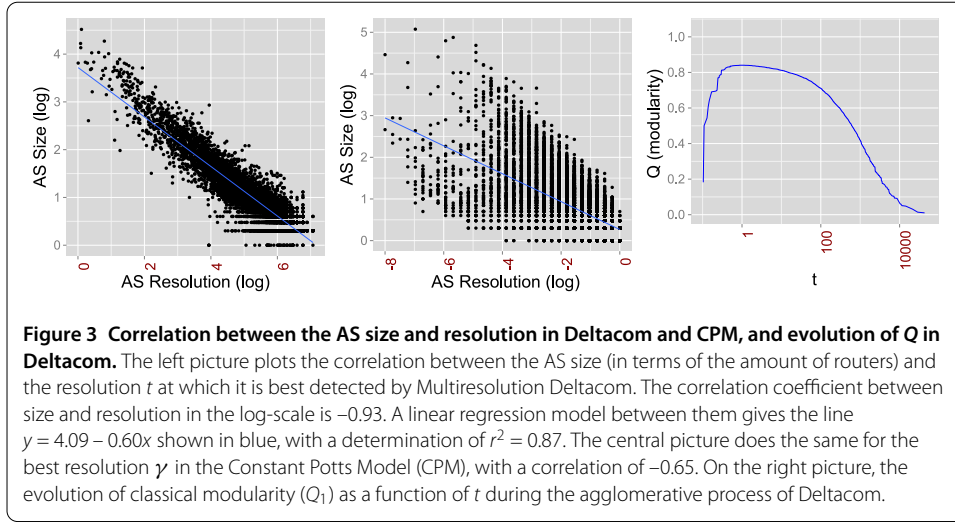
Figure 2 Cumulative distribution of the AS recall score. (*Upper*) The curves represent the cumulative distribution of the recall scores over all the ASes for different community discovery methods: $R_1(AS)$ is shown for Deltacom at $t = 1$ (i.e., modularity maximization) (*red*), LPM (*yellow*), Louvain modularity maximization (*pink*), CommUGP (*green*) and Infomap (*blue*). If we find the best resolution level for each AS using the multiresolution versions of Deltacom, Infomap and CPM, we get the *black*, *orange* and *purple* curves instead. The *brown* curve represents the Multiresolution Deltacom using the resolution value $\tilde{t}$ predicted by the regression line for each AS, and then computing $R_1(AS)$ in $\mathscr{C}_{\tilde{t}}$. Finally, if we use 15% of the AS routers when matching the partition at $\mathscr{C}_{\tilde{t}}$ for each AS, then we get the *light blue* curve. (*Lower*) The same methods as in the upper picture, but restricted to ASes containing at least 100 nodes.

We can clearly see that the five methods fail to reproduce the ASes structure. In the best performing one, Infomap, only 50% of the large ASes reach a recall of 0.4, which is however small. A premature conclusion might state that the Internet graph does not have communities at the router level, or at least that they are not related with the Autonomous Systems. But we will show that this is not the case, and that the failure of the methods is in part due to the largely variable sizes, internal structures and functions of the Internet ASes. However, the structure of the Internet graph at the router level can be better captured by using multiresolution methods.

In order to assess our hypothesis we shall exploit the resolution parameter t of Deltacom, which provides us with a new level of freedom for matching the ASes in the hierarchical community tree. For each t in the range $[t_{\min}, t_{\max}]$ we have a partition $\mathscr{C}_t$. Thus, we shall define a new recall score $R_2(AS)$ based on exploring a parameter space Θ with different partitions $\mathscr{C}_\theta$, $\theta \in \Theta$, associated to it such that the most similar community for

Figure 3 Correlation between the AS size and resolution in Deltacom and CPM, and evolution of *Q* in Deltacom. The left picture plots the correlation between the AS size (in terms of the amount of routers) and the resolution *t* at which it is best detected by Multiresolution Deltacom. The correlation coefficient between size and resolution in the log-scale is −0.93. A linear regression model between them gives the line $y = 4.09 - 0.60x$ shown in blue, with a determination of $r^2 = 0.87$. The central picture does the same for the best resolution γ in the Constant Potts Model (CPM), with a correlation of −0.65. On the right picture, the evolution of classical modularity (Q_1) as a function of *t* during the agglomerative process of Deltacom.

an Autonomous System, $C_{AS}^{(2)}$, is:

$$C_{AS}^{\theta} = \arg\max_{C \in \mathscr{C}_{\theta}} J(C, AS), \quad \theta \in \Theta,$$

$$C_{AS}^{(2)} = \arg\max_{\{C_{AS}^{\theta}: \theta \in \Theta\}} J(C_{AS}^{\theta}, AS),$$

and the recall score as:

$$R_2(AS) = J(C_{AS}^{(2)}, AS). \tag{8}$$

In the case of Deltacom, $\Theta = [t_{\min}, t_{\max}]$ and *t* plays the role of θ. In this way, the Jaccard similarity allows us to capture the moment in which each AS is formed. With this adjustments, the results clearly improve. The recall score $R_2(AS)$ (Eq. 8) over all the ASes is 0.87 (and 0.67 over the ASes of at least 100 nodes). The new cumulative distribution is presented as the black curve in Figure 2, and we shall call it *Deltacom Multiresolution*, as we are exploring all the communities at all possible resolutions.

The observed improvements tell us that each Autonomous System has a *natural* resolution at which it can be best observed. There is an important correlation between this resolution and the size of the AS, as the left picture in Figure 3 shows. The right picture in Figure 3 shows the evolution of classical modularity *Q* for the Internet graph as a function of the resolution *t*, i.e. as the algorithm evolves (from right to left). As one of the properties of our algorithm, Q_t has its maximum at *t* for every possible *t*, so that the classical *Q* is maximal at $t = 1$.

We stress that several multiresolution methods exist in literature, but the advantage of ours is that it can generate a hierarchical partition covering all the possible resolution levels, in one single run. However, we compared our results with those of two other multiresolution methods, i.e., Infomap multiresolution and the Constant Potts Model:

- The Infomap algorithm has a multiresolution variant [36], which generates a community tree with several levels $k \in \{1, \ldots, k_{\max}\}$. We matched each of the ASes into this structure in the same way as we did with Multiresolution Deltacom: i.e., we find the best level for each AS, and then we compute the recall score (now $\Theta = \{1, \ldots, k_{\max}\}$

and $\theta = k$). Here we found a clear improvement too, which is shown in the orange curve of Figure 2, pointing out that it is important to find the correct resolution for each AS.

- From the general schema described by Reichardt and Bornholdt in [31] several multiresolution approaches have been proposed. We used the Constant Potts Model by Traag et al. [37], which removes the null model in the Hamiltonian to cancel global dependences. For this algorithm we tested different values of $\gamma \in [10^{-8}, 1]$. We observed that this interval covers all the interesting situations, as for $\gamma = 10^{-8}$ we obtain a partition with 10 communities, and for $\gamma = 1$ all the nodes are in different communities. We sampled the interval by dividing it into 4, 8, 16 and 32 pieces, equidistant in the logarithmic scale. When dividing it into 2^k pieces we obtain $2^k + 1$ values of γ which we use as the parameter space $\Theta = \{\gamma_0, \gamma_1, \ldots, \gamma_{2^k}\}$ in order to match each AS against its most similar community (according to the recall score $R_2(AS)$) among the $2^k + 1$ different partitions. We chose to stop the exploration at $k = 5$ (33 values of γ) because the observed convergence suggested that refining the parameter space would not produce much better results. The curve for $k = 5$ representing CPM is shown in purple in Figure 2. Finally, the central picture in Figure 3 shows the best γ value (from among the 33 considered values) for each AS, as a function of its size. Compared to Deltacom, in CPM the correlation between resolution and size is not so strong.

From this analysis we conclude that there is no single resolution at which we can find most of the ASes. This had already been observed in [32] for modularity, and is partly due to the fact that modularity is affected by resolution limit issues [38]. Similar theoretical analysis have been provided for Infomap's map equation [39] and for some methods based on Hamiltonians of Potts models [40]. Despite these difficulties, here we show that the ASes do exist as communities at different levels of the hierarchical structure.

3.3 Detecting ASes from samples

The previous results show us that different ASes have different resolutions, and that we cannot expect to get all the ASes with classical community detection methods or at some particular resolution value. However, our results with the multiresolution methods are ideal, in the sense that we used the information of the AS structure in order to know at which resolution to stop for each AS. Now we wonder whether we might retrieve the ASes using a minimal amount of information: their size, and a small sample of their nodes. We observed that there is a strong dependence between the AS size and the AS resolution at which the AS is best identified by our algorithm, so that we shall use the linear regression in Figure 3 to approximate the best resolution for each AS, $\tilde{t}$. If we look for the most similar community to each AS restricted to the partition produced by Deltacom at that approximate resolution, $\mathscr{C}_{\tilde{t}}$, then we obtain the brown curve in dashed lines in Figure 2.

Lastly, we shall consider that we only know 15% of the nodes at random. Suppose that $AS_{\text{sample}} \subset AS$ is the known subset for each AS. We shall estimate the most similar community using the AS_{sample} and the partition at resolution $\tilde{t}$:

$$\hat{C}_{AS}^{(2)} = \arg\max_{C \in \mathscr{C}_{\tilde{t}}} J(C, AS_{\text{sample}}),$$

$$R_3(AS) = \hat{R}_2(AS) = J\left(\hat{C}_{AS}^{(2)}, AS\right), \tag{9}$$

i.e., we only consider the nodes of the AS which are part of our sample when choosing the most similar community. Instead, for computing the recall score $R_3(AS)$ (Eq. 9) we consider all the nodes of the AS, so that the method can be compared against the others. The results for $R_3(AS)$ are those of the light blue curve in dashed lines in Figure 2.

The closeness between the brown and light blue curves proves that we do not need to know the whole AS in order to identify it as a community at a certain step of the modularity optimization process: 15% of the nodes is enough in order to get as good a result as if we knew all the AS nodes. However, the distance between the brown and black curve tells us that the linear regression between size and resolution is just a rough approximation. We also point out that we tested sample sizes between 5% and 25% and the performance was similar (average recalls between 0.57 and 0.59), mainly because ideally the probability of misclassifying n random nodes of an AS decreases exponentially with n, so that even for a small sample of nodes the majority of them should be correctly classified if the AS exists as a community at that resolution (e.g., with a recall of at least 0.5). In conclusion, the sample-based method seems to be very successful, and is only limited by the error of the linear regression. We can thus identify many of the ASes just knowing their size and a small sample of their nodes. This method can also be extended to other networks, even if the relation between resolution and size is not known exactly. In that case, one should match the most similar community at each resolution, and then choose the resolution for which the most similar community has the closest size to the AS size, for example.

4 Discussion

The results in Figure 2 show a type of stochastical dominance among the methods. The lowest curves are the ones most left-skewed and thus represent better correspondence between the communities identified by maximizing the Jaccard similarity and the ASes. In particular, the area above each curve is an interesting measure of the goodness of the method, and it also represents the mean recall score over all the ASes. The list of areas is presented in Table 3, computed for the cumulative distributions of ASes with at least 100 nodes.

The difficulties of classical modularity maximization due to the resolution limit and the degeneracy of its maxima are manifested in the poor average values obtained by Deltacom

Table 3 Average recall scores of the methods

Method	avg $R(AS)$	
	ASes $\geq$ 100 nodes	All ASes
LPM	0.05	0.18
Deltacom ($t = 1$)	0.08	0.00
Louvain	0.13	0.01
CommUGP	0.16	0.37
Infomap	0.41	0.35
Deltacom Multiresolution + Regression line + 5% sample	0.57	0.75
Deltacom Multiresolution + Regression line	0.59	0.77
Infomap Multiresolution (Best Resolution)	0.61	0.49
Deltacom Multiresolution (Best Resolution)	0.67	0.87
CPM (Best Resolution from 33 values of γ)	0.70	0.83

Average recall score (or area above the cumulative distribution curve) for the different tested methods. The top-5 methods are single resolution methods which generate one partition. The bottom-5 methods are multiresolution methods in which we determine the best resolution for each AS by using some type of *a priori* information.
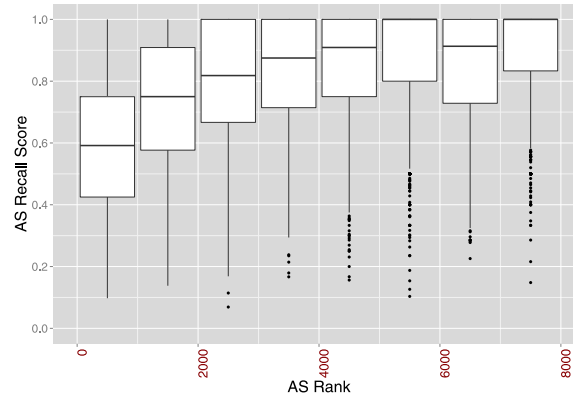
Figure 4 AS Recall score vs. AS Rank. Boxes in this figure represent Autonomous Systems grouped by *AS rank* according to CAIDA [10], while the *y*-axis represents the recall score obtained with the Multiresolution Deltacom algorithm, $R_2(AS)$ (Eq. 8). Each box contains information on the median recall score and its interquartile range (IQR). The whiskers represent extreme values within $1.5 \cdot IQR$, and the points are outliers. There is a clear trend towards higher recalls for low-ranked ASes.

at $t = 1$ (0.08) and by Louvain (0.13) for the recall score of ASes with at least 100 nodes. But the multiresolution algorithms have a good performance when we explore all the possible resolution levels, showing that most of the Autonomous Systems do have a community structure (80% of them have a recall over 0.70). Instead, most of the ASes which were not detected by Deltacom Multiresolution belong to the core of the Internet: in Figure 4 we plot the recall score of the ASes grouped according to their *AS rank*. The AS rank is a ranking of ASes designed by CAIDA and based on the customer cone size (the set of ASes that an AS can reach using customer links [41]). We observe that low ranked ASes have very good recall scores typically (above 0.7). Instead, most of the ASes found with a low recall have a high AS rank and belong to large carriers. Regarding the points with low AS Rank and low recall, many of them could be identified as IXPs and CDNs, which also belong to the Internet backbone but have a small customer cone.

In Figures 5, 6 and 7 we analyse some particular cases of (a) *well-detected, high-ranked ASes*, (b) *bad-detected, high-ranked ASes* and (c) *very well-detected, mean-ranked ASes*. Each row represents an AS, and the three columns represent (i) *the internal node degree distribution* (normalized by the node degree), (ii) *the internal node degree vs. node degree* and (iii) *the evolution of the best Jaccard similarity at different resolutions of the Multiresolution Deltacom algorithm* (its maximum corresponds to the AS recall score).

In the central pictures of Figure 5 we plot the number of internal edges as a function of node degree for some ASes which were not well-detected. The diagonal dashed line corresponds to k_{in}/k; the nodes under this line have more external connections than internal ones. The colour indicates if the node was classified inside the community which retrieved the best Jaccard similarity (*black*) or not (*red*). We observe that all the ASes in Figure 5 have high degree nodes with more external that internal connections; these nodes are not recognized as part of their ASes and thus induce many of the short-degree nodes into error. Into these group of ASes we found carrier networks and IXCs (e.g., AT&T and Sprint), IXPs (e.g., the London Internet Exchange) and some CDNs (e.g., Akamai and Amazon).
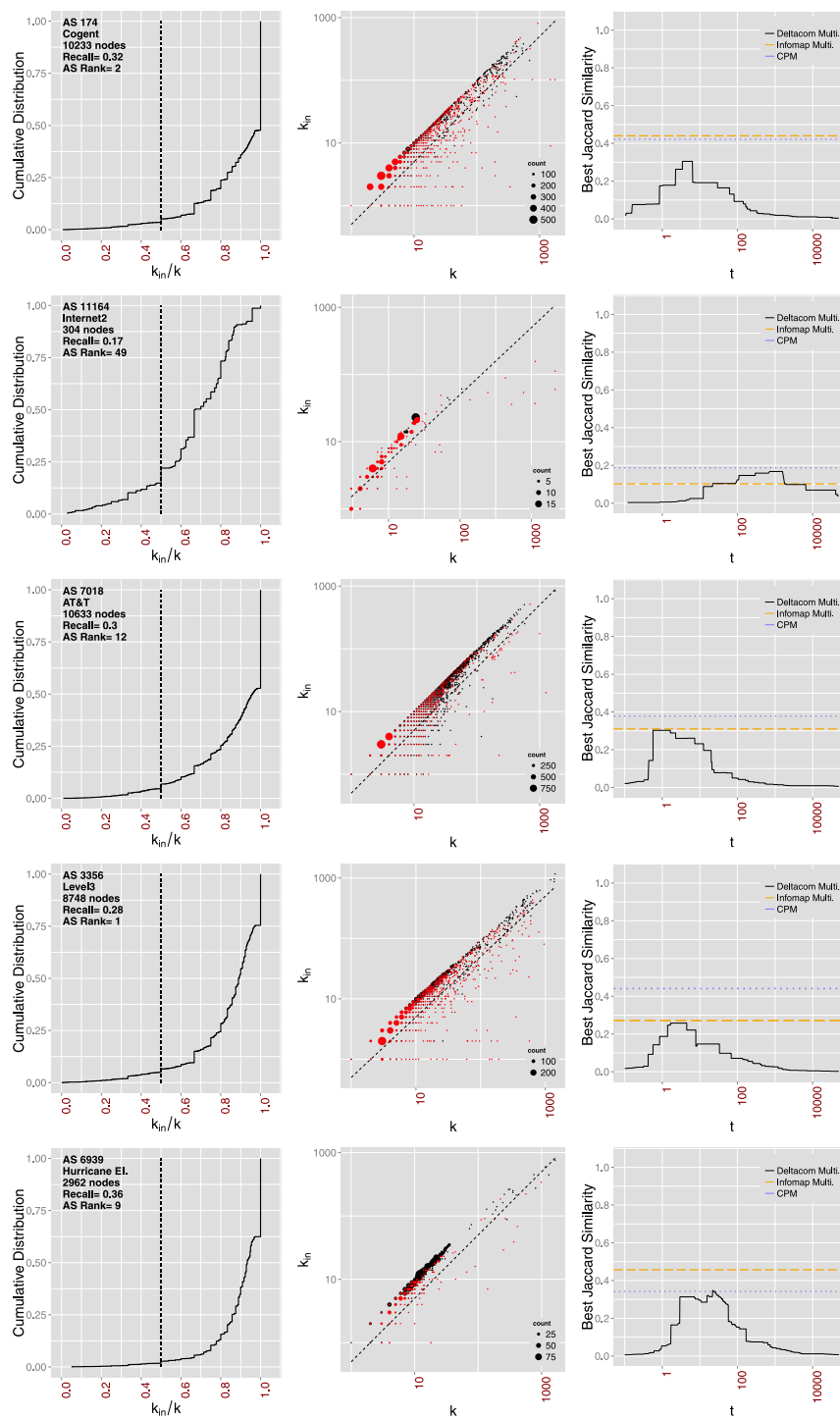
Figure 5 High-ranked ASes with bad recall score. Each row represents an AS, described in the upper-left labels. The left column plots the distribution of the internal node connections, k_{in}, normalized by the node degree k. The vertical dashed line at $k_{in}/k = 0.5$ represents the notion of strong community. The central column plots the internal node degree, k_{in}, vs. the node degree, k. The point size represents the number of nodes with that (k_{in}, k) pair; the diagonal line represents the condition $k_{in}/k = 0.5$. The node colour indicates if the node belongs to the most similar community retrieved by Multiresolution Deltacom (black) or not (red). The right column plots the best Jaccard similarity obtained with Multiresolution Deltacom at resolution t (its maximum being the recall). The horizontal lines represent the recall score for the other multiresolution methods: Infomap Multiresolution in orange (dashed line) and CPM in purple (dotted line). The ASes in these pictures have at least 100 nodes, high AS rank, and a bad recall score.
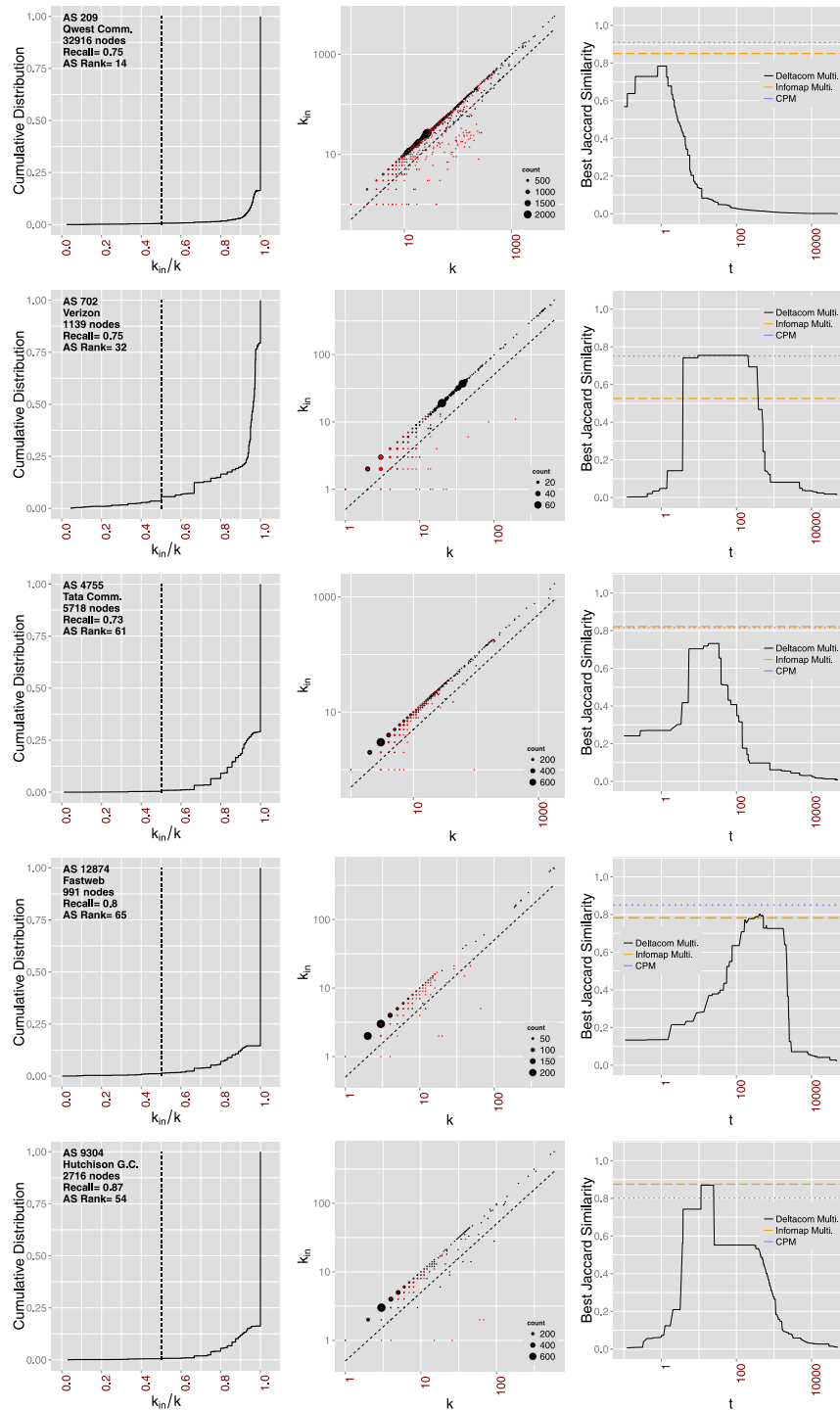
Figure 6 High-ranked ASes with good recall score. Normalized internal degree distribution (*left-column*), k_{in} vs. k (*middle-column*) and the best Jaccard similarity at resolution t (*right-column*) of the Multiresolution Deltacom algorithm for a set of 5 ASes with at least 100 nodes and high AS rank, retrieved with a good recall score. A more detailed description can be found in Figure 5.
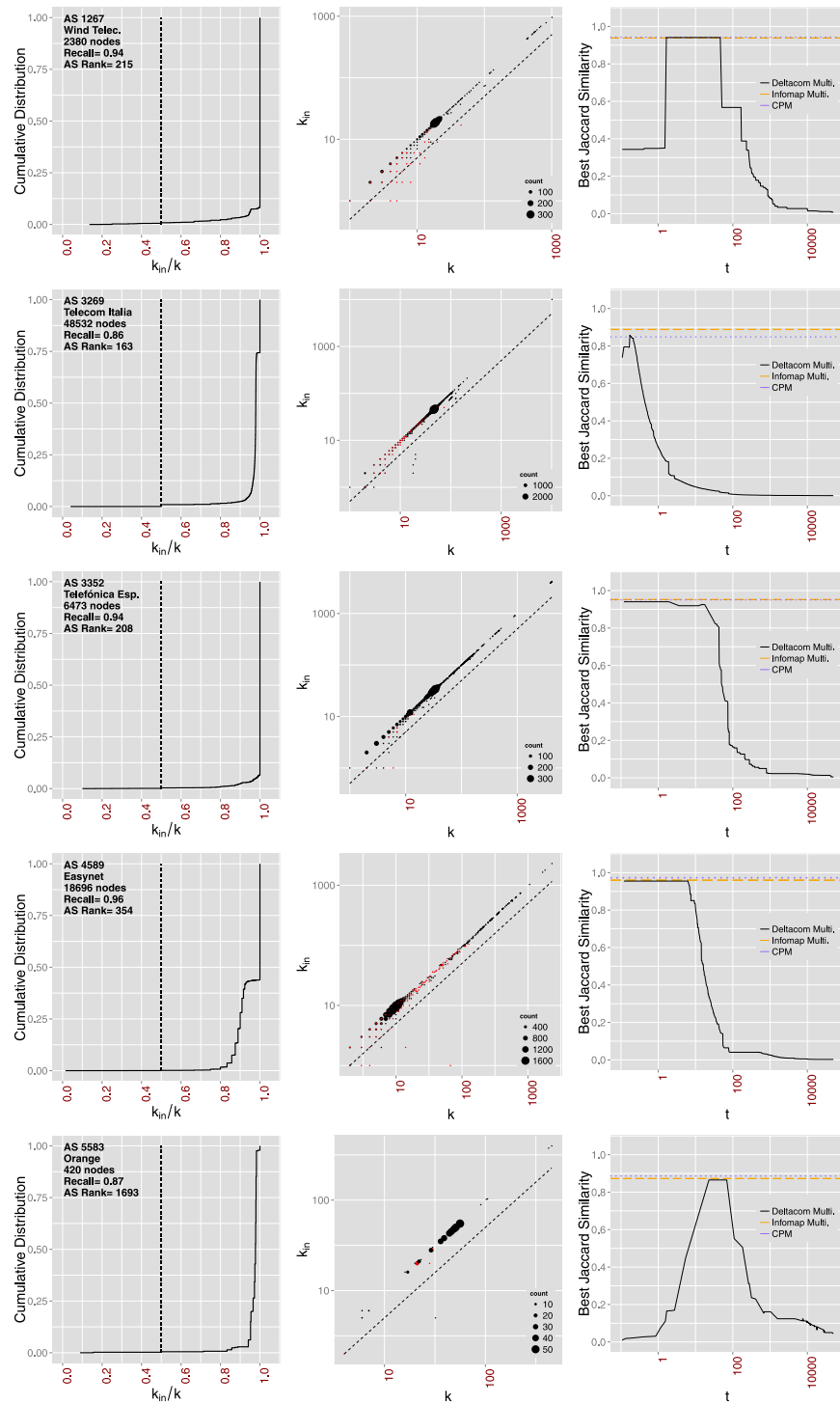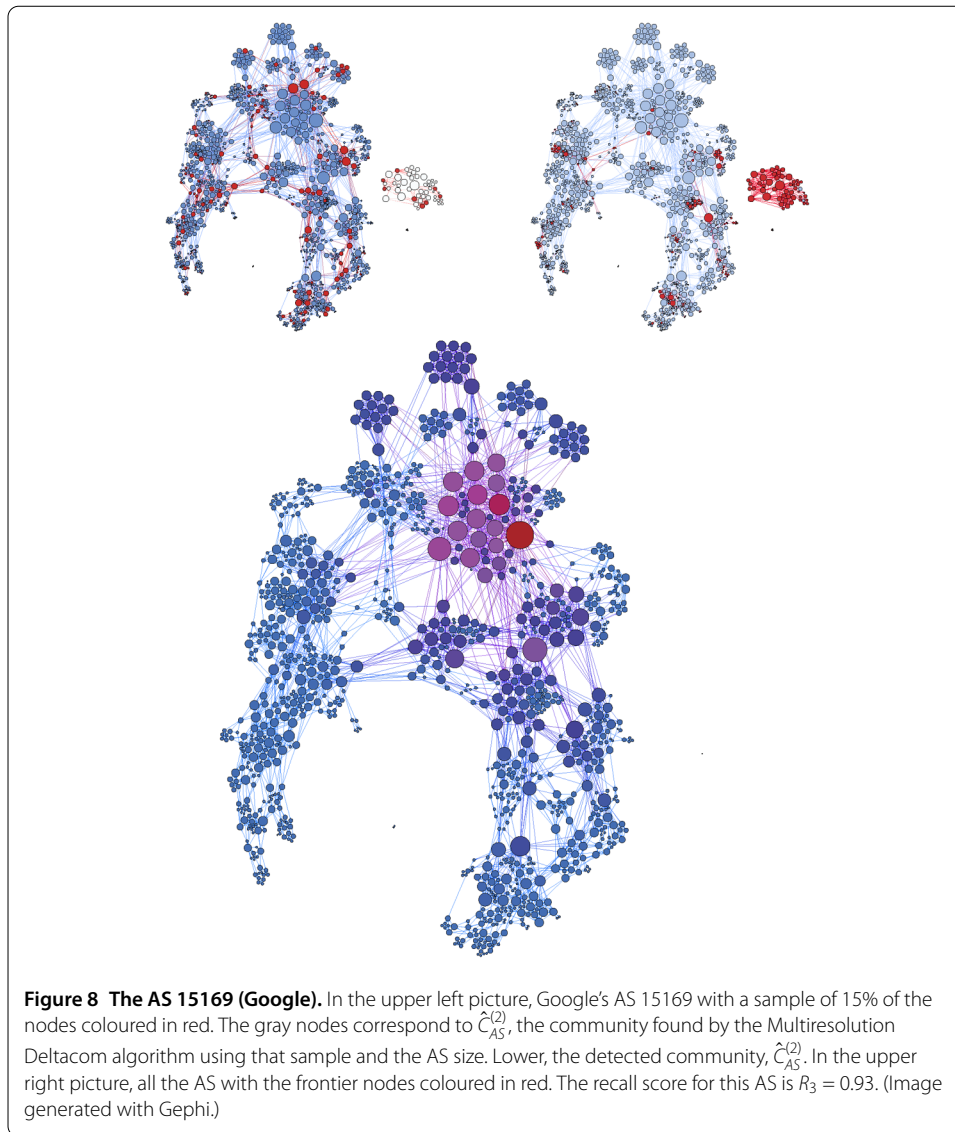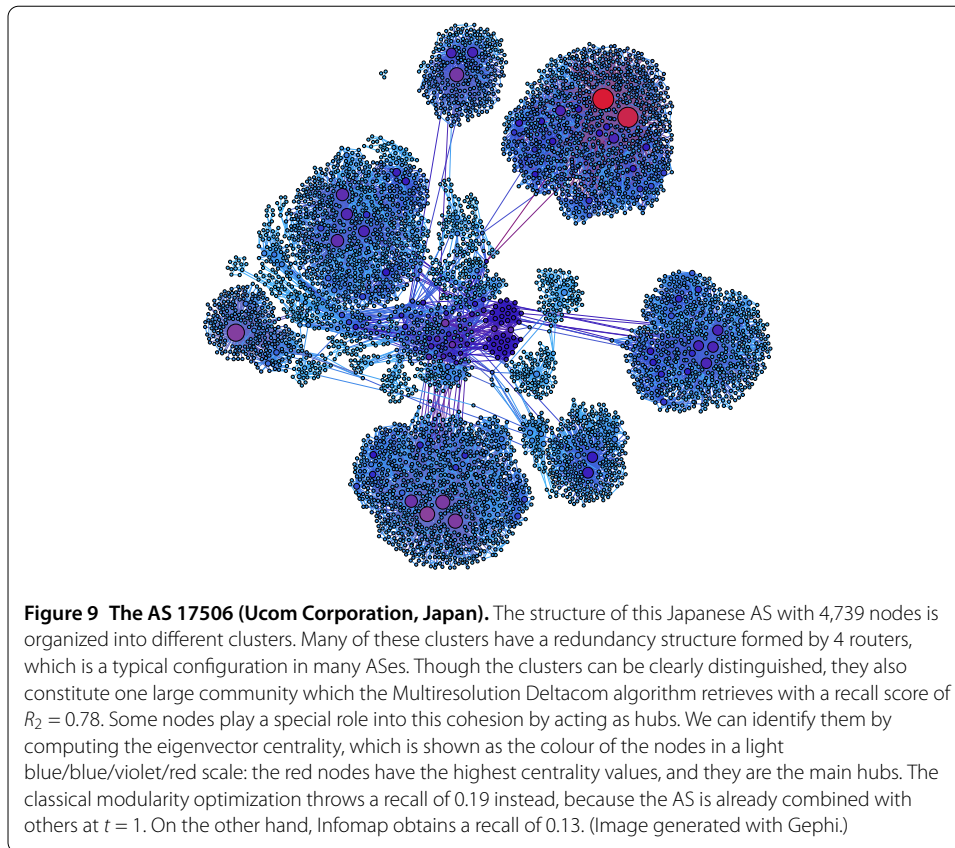
Figure 7 Average-ranked ASes with very good recall score. Normalized internal degree distribution (*left-column*), k_{in} vs. k (*middle-column*) and the best Jaccard similarity at resolution t (*right-column*) for a set of 5 ASes with at least 100 nodes and intermediate AS rank, retrieved with a very good recall score. A more detailed description can be found in Figure 5.

Figure 8 The AS 15169 (Google). In the upper left picture, Google's AS 15169 with a sample of 15% of the nodes coloured in red. The gray nodes correspond to $\hat{C}_{AS}^{(2)}$, the community found by the Multiresolution Deltacom algorithm using that sample and the AS size. Lower, the detected community, $\hat{C}_{AS}^{(2)}$. In the upper right picture, all the AS with the frontier nodes coloured in red. The recall score for this AS is $R_3 = 0.93$. (Image generated with Gephi.)

These types of ASes lay in the backbone of the Internet and they are densely connected among them, so that their nodes may not respect the classical notions of community. It is possible that we should think in an overlapping community structure for the core of the Internet, following the ideas in [42] regarding the core-periphery structure of networks and its relation with community structure. In this network, highly overlapping communities might be related with the presence of IXPs or nodes which do not satisfy the strong community definition, for example.

ASes with good recall score are shown in Figures 6 and 7. In them, all the high-degree nodes have a high proportion of internal connections. These nodes usually act as hubs for their ASes; into this category we find medium-sized and large-sized ISPs, university networks and other small ASes. An example of one of these networks is the Google AS, shown in Figure 8. In the lower part of the figure we plot it as detected by the Multiresolution Deltacom algorithm using the 15% of the nodes and the AS size; the recall score of the AS is 0.93. The colour of the nodes represents their eigenvector centrality inside the community with a light blue/blue/violet/red scale for increasing centrality values; the

Figure 9 The AS 17506 (Ucom Corporation, Japan). The structure of this Japanese AS with 4,739 nodes is organized into different clusters. Many of these clusters have a redundancy structure formed by 4 routers, which is a typical configuration in many ASes. Though the clusters can be clearly distinguished, they also constitute one large community which the Multiresolution Deltacom algorithm retrieves with a recall score of $R_2 = 0.78$. Some nodes play a special role into this cohesion by acting as hubs. We can identify them by computing the eigenvector centrality, which is shown as the colour of the nodes in a light blue/blue/violet/red scale: the red nodes have the highest centrality values, and they are the main hubs. The classical modularity optimization throws a recall of 0.19 instead, because the AS is already combined with others at $t = 1$. On the other hand, Infomap obtains a recall of 0.13. (Image generated with Gephi.)

radius represents their degree in the full graph. The upper left picture shows the 15% sample in red, and the upper right figure shows the nodes in the frontier of the AS, also in red. Those nodes from the AS which were not detected by our algorithm are shown in white in the upper left picture. We observe an interesting modular structure, with several small modules strongly connected among them, and with a tree-like structure. Some high-degree nodes act as hubs inside the AS; however, they are not connected to other ASes usually. The AS has also a small connected component, probably due to a bias in the exploration or to the limitations of the AS affiliation algorithm; this small component was not detected. However, the reconstruction of the main component was excellent; the only difference being some small degree nodes not belonging to the AS, but to carrier ASes like AT&T and Level3. Almost all the nodes in the frontier of the main component have small degree and small centrality values. The role of hubs is also visualized in Figure 9 for the UCom Corporation AS.

It is interesting to contrast the general results against some classical notions of community. Looking at the central pictures of Figures 5, 6 and 7 again, we can see that all of them have nodes under the dashed diagonal line which represents the limit of the notion of strong community by Radicchi et al. [43], i.e., having more internal connections than external ones. This notion is quite strict indeed, and neither the real ASes nor the communities found by modularity maximization follow it. In [44], Hu et al. introduced a relaxed version of this notion, in which each node has more internal connections than towards each of the other communities separately; we found that 4% of the nodes (spread through 70% of the ASes) do not follow this definition, either.

The limitations of our algorithm for correctly recovering some of the ASes might also be related to the detectability threshold of community detection [45–47] and to the community detection paradox [48]: even when there are more average internal connections than external ones, algorithms might not succeed to retrieve a community if the difference between these quantities is not above a certain threshold. This limitation not only affects modularity-based methods but is quite general.

Hric et al. [35] have recently used the Jaccard similarity for identifying communities with ground-truth groups as we did. They also observed that community detection algorithms perform very poorly according to this measure and they proposed some conjectures, as the need to change the notion of community or to introduce some non-topological information. In [49], for example, Darst et al. analyse how an a priori knowledge of the number of communities improves the performance by helping to set the resolution level. But, in many complex networks like the Internet there might not be one single resolution at which all the communities can be discovered. Here we have seen that knowing the community sizes and some of their nodes, multiresolution algorithms might increase their performance significantly.

The use of some additional information for the identification of communities may be a trade-off solution considering the complexity and limitations of the community detection problem. Our last experiment, in which we use a sample of nodes for matching the community in the hierarchical structure, is related to the seed set expansion problem (i.e., extracting a community by expanding a local seed set of nodes), which has been recently explored in [50–52].

Thus, our work may shed some light into the problem of improving the performance of community detection algorithms: in the Internet router-level graph, we have shown that the coexistence of communities with multiple resolutions is the main difficulty for the detection algorithms. So, when proposing new conceptions of community structure, we should take into account that very heterogeneous communities must coexist. Regarding the detection of ground-truth communities in real networks, the following question remains open: Can the ground-truth communities be retrieved without additional information, e.g., as significant communities at the right resolution level?

5 Conclusions

We analysed the internal community structure of the Internet Autonomous Systems by using Deltacom, a multiresolution algorithm. Deltacom performs hierarchical partitioning through modularity optimization at different resolution levels in one single run, and with a low computational cost. The algorithm was made available to the scientific community as an open-source software through SourceForge.

We applied Deltacom to the router-level graph of the Internet obtained from CAIDA, and we observed that most of the ASes which are not in the Internet backbone can be identified as communities if the correct resolution level is used in the community discovery process. However, we showed that several optimization algorithms fail to detect these Autonomous Systems due to their largely variable sizes and structures. With our results, we show that the use of proper additional information can significantly improve the performance of community identification.

Many of the Autonomous Systems in the Internet backbone (and usually identified by a very high AS rank) could not be identified as communities, and we observed that this was

due to the presence of high degree nodes with more external connections than internal ones, violating one of the main notions of community structure. Instead, stub ASes and ISP's have high-degree nodes which act as hubs and provide internal connectivity, while the external connectivity is usually provided by lower-degree nodes. These ASes were well detected as communities.

By correlating the resolution of each AS with its size and using a small fraction of the AS affiliation data, we showed that most of the ASes which do have a community structure can be retrieved. In this way we provide a method for identifying Autonomous Systems with scarce information: a short sample of their nodes and their size.

Finally, we discussed the limitations of the community detection methods for detecting ground-truth communities. These difficulties had been previously observed for many large networks and we also observed it here for the Internet Autonomous Systems. Based on our results, we consider that the use of some local information might arise as a trade-off solution for identifying communities in these cases.

We think that these results on the Internet structure at the router level can also be used for improving Internet topology models and the estimation of properties as node centrality or clustering. The work shall also lead to new discussions regarding the hierarchical structure of the Internet and the detection of significant communities in large heterogeneous networks.

Author details
[1]ISI Foundation, Turin, Italy. [2]Facultad de Ingeniería, Universidad de Buenos Aires, Buenos Aires, Argentina. [3]INTECIN (CONICET), Buenos Aires, Argentina. [4]ITBA, Buenos Aires, Argentina.

References
1. Caldarelli G, Vespignani A (2007) Large scale structure and dynamics of complex networks: from information technology to finance and natural science. World Scientific, River Edge
2. Ravasz E, Barabási A-L (2003) Hierarchical organization in complex networks. Phys Rev E 67(2):026112
3. Pastor-Satorras R, Vespignani A (2007) Evolution and structure of the Internet: a statistical physics approach. Cambridge University Press, Cambridge
4. Gregori E, Lenzini L, Orsini C (2013) k-Dense communities in the Internet AS-level topology graph. Comput Netw 57(1):213-227
5. Gregori E, Lenzini L, Orsini C (2011) k-Clique communities in the Internet AS-level topology graph. In: 31st international conference on distributed computing systems workshops (ICDCSW 2011), pp 134-139
6. Ge X, Wang H (2012) Community overlays upon real-world complex networks. Eur Phys J B 85(1):1-10
7. Rossi RA, Fahmy S, Talukder N (2013) A multi-level approach for evaluating Internet topology generators. In: IFIP networking conference, pp 1-9
8. Hirayama T, Arakawa S, Arai K, Murata M (2012) Modularity structure and traffic dynamics of ISP router-level topologies. In: The 2012 international symposium on nonlinear theory and its applications (NOLTA 2012), Palma, Mallorca, pp 856-859
9. Fortunato S (2010) Community detection in graphs. Phys Rep 486(3-5):75-174
10. CAIDA: the Cooperative Association for Internet Data Analysis. http://www.caida.org/
11. Keys K, Hyun Y, Luckie M, Claffy K (2013) Internet-scale IPv4 alias resolution with MIDAR. IEEE/ACM Trans Netw 21(2):383-399

12. Dall'Asta L, Alvarez-Hamelin JI, Barrat A, Vázquez A, Vespignani A (2006) Exploring networks with traceroute-like probes: theory and simulations. Theor Comput Sci 355(1):6-24

13. Pastor-Satorras R, Vázquez A, Vespignani A (2004) Topology, hierarchy, and correlations in Internet graphs. In: Complex networks. Lecture notes in physics, vol 650. Springer, Berlin, pp 425-440

14. Chang H, Jamin S, Willinger W (2001) Inferring AS-level Internet topology from router-level path traces. In: Proceedings of SPIE ITCom

15. Tangmunarunkit H, Govindan R, Shenker S, Estrin D (2001) The impact of routing policy on Internet paths. In: Proceedings of the 20th IEEE international conference on computer communications (INFOCOM), pp 736-742

16. The CAIDA UCSD IPv4 routed /24 topology dataset - 2011-10. http://www.caida.org/data/active/ipv4_routed_24_topology_dataset.xml

17. Huffaker B, Dhamdhere A, Fomenkov M, Claffy K (2010) Toward topology dualism: improving the accuracy of AS annotations for routers. In: 9th passive and active measurement conference (PAM 2010), Zurich, Switzerland

18. The CAIDA macroscopic Internet Topology Data Kit (ITDK) - 2011-10. http://www.caida.org/data/internet-topology-data-kit/release-2011-10.xml

19. Batagelj V, Zaveršnik M (2011) Fast algorithms for determining (generalized) core groups in social networks. Adv Data Anal Classif 5(2):129-145. doi:10.1007/s11634-010-0079-y

20. Donnet B, Luckie M, Mérindol P, Pansiot J-J (2012) Revealing MPLS tunnels obscured from traceroute. Comput Commun Rev 42(2):87-93

21. Rosvall M, Bergstrom CT (2007) An information-theoretic framework for resolving community structure in complex networks. Proc Natl Acad Sci USA 104(18):7327-7331

22. Raghavan UN, Albert R, Kumara S (2007) Near linear time algorithm to detect community structures in large-scale networks. Phys Rev E 76(3):036106

23. Blondel VD, Guillaume J-L, Lambiotte R, Lefebvre E (2008) Fast unfolding of communities in large networks. J Stat Mech Theory Exp 2008(10):P10008

24. Beiró MG, Busch JR, Grynberg SP, Alvarez-Hamelin JI (2013) Obtaining communities with a fitness growth process. Physica A 392(9):2278-2293

25. Newman MEJ, Girvan M (2004) Finding and evaluating community structure in networks. Phys Rev E 69(2):026113

26. Clauset A, Newman MEJ, Moore C (2004) Finding community structure in very large networks. Phys Rev E 70(6):066111

27. Guimerà R, Amaral LAN (2005) Cartography of complex networks: modules and universal roles. J Stat Mech Theory Exp 2:02001

28. Duch J, Arenas A (2005) Community detection in complex networks using extremal optimization. Phys Rev E 72(2):027104

29. Noack A, Rotta R (2009) Multi-level algorithms for modularity clustering. In: Proceedings of the 8th international symposium on experimental algorithms (SEA 2009). Springer, Berlin, pp 257-268

30. Fortunato S, Barthélemy M (2007) Resolution limit in community detection. Proc Natl Acad Sci USA 104(1):36-41

31. Reichardt J, Bornholdt S (2006) Statistical mechanics of community detection. Phys Rev E 74(1):016110

32. Lancichinetti A, Fortunato S (2011) Limits of modularity maximization in community detection. Phys Rev E 84(6):066122. http://pre.aps.org/abstract/PRE/v84/i6/e066122

33. Busch JR, Beiró MG, Alvarez-Hamelin JI (2010) On weakly optimal partitions in modular networks. arXiv:1008.3443

34. DeltaCom (2015) http://sourceforge.net/projects/deltacom/

35. Hric D, Darst RK, Fortunato S (2014) Community detection in networks: structural communities versus ground truth. Phys Rev E 90:062805

36. Rosvall M, Bergstrom CT (2011) Multilevel compression of random walks on networks reveals hierarchical organization in large integrated systems. PLoS ONE 6(4):e18209

37. Traag VA, Van Dooren P, Nesterov Y (2011) Narrow scope for resolution-limit-free community detection. Phys Rev E 84(1):016114

38. Kumpula JM, Saramäki J, Kaski K, Kertész J (2007) Limited resolution in complex network community detection with Potts model approach. Eur Phys J B 56(1):41-45

39. Kawamoto T, Rosvall M (2015) Estimating the resolution limit of the map equation in community detection. Phys Rev E 91(1):012809

40. Xiang J, Hu K (2012) Limitation of multi-resolution methods in community detection. Physica A 391(20):4995-5003

41. Luckie M, Huffaker B, Dhamdhere A, Giotsas V, Claffy K (2013) AS relationships, customer cones, and validation. In: Proceedings of the 2013 Internet measurement conference (IMC'13). ACM, New York, pp 243-256

42. Yang J, Leskovec J (2014) Overlapping communities explain core-periphery organization of networks. Proc IEEE 102(12):1892-1902

43. Radicchi F, Castellano C, Cecconi F, Loreto V, Parisi D (2004) Defining and identifying communities in networks. Proc Natl Acad Sci USA 101(9):2658

44. Hu Y, Chen H, Zhang P, Li M, Di Z, Fan Y (2008) Comparative definition of community and corresponding identifying algorithm. Phys Rev E 78(2):026121

45. Decelle A, Krzakala F, Moore C, Zdeborová L (2011) Inference and phase transitions in the detection of modules in sparse networks. Phys Rev Lett 107:065701

46. Nadakuditi RR, Newman MEJ (2012) Graph spectra and the detectability of community structure in networks. Phys Rev Lett 108:188701

47. Radicchi F (2013) Detectability of communities in heterogeneous networks. Phys Rev E 88:010801

48. Radicchi F (2014) A paradox in community detection. Europhys Lett 106(3):38001

49. Darst RK, Nussinov Z, Fortunato S (2014) Improving the performance of algorithms to find communities in networks. Phys Rev E 89:032809

50. Whang JJ, Gleich DF, Dhillon IS (2013) Overlapping community detection using seed set expansion. In: Proceedings of the 22nd ACM international conference on information & knowledge management. ACM, New York, pp 2099-2108

51. Kloumann IM, Kleinberg JM (2014) Community membership identification from small seed sets. In: Proceedings of the 20th ACM SIGKDD international conference on knowledge discovery and data mining. ACM, New York, pp 1366-1375
52. Li Y, He K, Bindel D, Hopcroft JE (2015) Uncovering the small community structure in large networks: a local spectral approach. In: Proceedings of the 24th international conference on world wide web, pp 658-668