Assistive Growth: Towards Scalable Community Networks Topologies☆

Leonardo Maccari[a], Gabriele Gemmi[a], Renato Lo Cigno[a], Merkourios Karaliopoulos[b], Leandro Navarro[c]

[a]*Department of Information Engineering and Computer Science, University of Trento, Italy*
[b]*Athens University of Economics and Business, Athens, Greece*
[c]*Department of Computer Architecture, Universitat Politécnica de Catalunya, Barcelona, Spain*

Abstract

Community Networks are grassroots, bottom-up initiatives to build local infrastructures, normally based on WiFi technology, that bring broadband networking in areas with inadequate offer or other reasons prevent the deployment of traditional infrastructures such as ADSL, FTTx or wide-band cellular (LTE, 5G). The growth of these networks is mostly unplanned, depending on the one hand on the willingness of people to participate, and on the other hand on the feasibility of the wireless links connecting the home of the potential participant to the infrastructure. Exploiting open source resources, such as Open Street Map and very detailed (less than 1 m resolution) LIDAR-based data on buildings, we present a methodology for the stochastic forecast the growth of a Community Network given the area where the "community" starts building it. This base methodology, implemented into an automated tool, takes into account the technical and economic feasibility of adding nodes to the network, as well as guaranteed limits on the per-node performance of the network in saturation predicting the topology of the network and its limit size given the above constraints. The methodology is coupled with simple economic incentive schemes to explore if proper incentives mechanisms can influence (and improve) the growth of the network. We selected four different scenarios: Urban, Suburban, Intermediate, and Rural targeting spots that expressed interest in developing a Community Network in Tuscany, Italy. Results in all three scenarios highlight the characteristics of the topology that spontaneously emerge from the natural growth of the network, and the advantages that properly crafted incentives bring to this process, improving the size, the performance, and the resilience of the network emerging from this spontaneous process.

Keywords: Community Networks, Stochastic Network Evolution, Economic Incentives

1. Introduction

Community Networks (CNs) are grassroots, bottom up networks usually built as 802.11-based Wireless Mesh Networks (VMNs). CNs are flourishing in Europe and beyond, they grow in many different environments, but the "preferred" ecosystem are areas where, whatever the reason, standard telecommunication infrastructures don't work properly. Often these are areas of "market failure", i.e., areas where commercial operators think it is not profitable to invest. In other cases they flourish in places where there is a fervent cultural life, where people have a strong sentiment of pertaining and invest into a local infrastructure that can bestow on the community much more than the standard Internet: In terms of digital divide reduction, in terms of rich and non-commercial services, in terms of local economy support, and so forth. It is now accepted that CNs, even if they sometimes fail, are part of the global Internet, and that they should be nurtured by the regulatory system and policy makers, because when they do not fail, they are a strong factor in the socio-economic development and well being of their region[1].

A CN is typically launched thanks to the initiative of a small group of people who may be driven by different motives ranging all the way from enthusiasm about technology and do-it-yourself practices to social activism and political causes. This group invests personal resources (effort, time, money) to set up a first small set of network nodes that ensure connectivity to the rest of the Internet or, when the CNs are conceived as a sort of alternative Internet, and not only an access network, the provision of local services. This initial burst of activity normally brings the network to have a few tens of nodes with a topology that is mainly decided by the location of the members' homes. In a second longer phase of the CN lifetime, the network grows thanks to the addition of nodes by people who join the network and become members of the Community. The growth of the network is a distributed process with a strong crowdsourcing characterization that is not subject to top-down planning as it is the case with conventional communication networks. Instead, the decision as to which existing network node(s) should be the point(s) of network attachment for a new node that submits a request to join the network is typically taken *locally* and *heuristically*, considering a few parameters and constraints such as the geo-location of the node, the network coverage in the area, the availability (and cost)

☆This work was financed partially by the European Commission, H2020-ICT-2015 Programme, Grant Number 688768 "netCommons" (Network Infrastructure as Commons).

Email addresses: `leonardo.maccari@unitn.it` (Leonardo Maccari), `gabriele.gemmi@stud.unitn.it` (Gabriele Gemmi), `renato.locigno@unitn.it` (Renato Lo Cigno), `renato.locigno@unitn.it` (Merkourios Karaliopoulos), `leandro@ac.upc.edu` (Leandro Navarro)

[1]For additional information on CNs, their diffusion in the world, their size and characteristics see the web page of the netCommons project at `https://netcommons.eu` with particular reference to Deliverable 1.2 "Report on the Existing CNs and their Organization," `https://netcommons.eu/?q=content/report-existing-cns-and-their-organization-v2` and Deliverable 1.4 "Report on the Governance Instruments and their Application to CNs," `https://netcommons.eu/?q=content/report-governance-instruments-and-their-application-cns-v2`.

of proper devices and so forth. As a first instance, the decision is "greedy," meaning that the cost of the node to be added is sustained by the new member, and he takes the decision on what and how to deploy the node based on his own convenience.

These *local* evolutionary decisions, however, shape the growth of the network, determine the main *global* properties of the resulting network graph, and strongly influence the network performance and the overall cost of the developed infrastructure. These properties are, for instance, the average length of the shortest paths to the Internet gateway(s), the robustness of the network to topology failures, as well as the distribution of its overall capacity and traffic load across its nodes and links.

Two key questions arise from the above discussion: i) Is it possible to understand how the local decisions affects the shape, performance, and sustainability of the network, and ii) is it possible to improve this growth process with appropriate advice and incentives that inform the local choice of the new users when they set up their own node?

The two main contributions of this paper are answers to these two questions. The first one takes the form of a tool that simulate the growth of a CN given a (very) detailed topological description of the area, a database of possible devices to be used to setup the node and algorithms that describe the local heuristic decision process. The second one is instead showing that carefully crafted, yet simple, incentives schemes to influence these local decisions can lead to drastic improvements of the network performance as it grows, making them scalable up to several hundreds of nodes yet maintaining good guarantees of a satisfactory capacity for all nodes.

The tool starts from an OpenStreetMap description of the area considered, then add a lidar-based buildings height description with a precision better than 1 m, and finally, using an appropriate propagation model and datasheets of real devices, implements the growth of the network selecting randomly a new potential user (a location on a building) and adding the links from this node, if proper connectivity is possible, following an algorithm describing the local decision heuristic.

The incentives scheme starts from modeling the problem of "optimal" network growth as a Multicommodity Flow (MCF). The MCF problem is obviously too complex to be solved for networks of interesting sizes, but this modeling allows deriving simple heuristics that can be added to the local decision process to improve the overall performance of the network. Clearly these heuristics may increase the cost of the new node, which calls for a proper scheme of incentives to avoid that the local choice is based solely on a greedy decision.

Results are obtained in four different scenarios: 1 urban, 1 suburban, 1 intermediate, and 1 rural targeting spots that expressed interest in developing a Community Network in Tuscany, Italy. The results show that it is indeed possible to obtain very useful insight on the growth process and also that proper incentives to modify the local choices of new users can modify the global properties of the network and let it grow to 2–3 times the size it would grow without them yet maintaining capacity guarantees for (almost) all nodes.

2. Background and system model

The development of CNs is a participatory and evolutionary process, so we need to have a dynamic model to represent it. The dynamic part of the model is represented by new nodes that are added to the existing network when a new user desires to join the community network. This section describe the main characteristics of this process and the challenge we tackle to represent it properly in a simulator.

2.1. CN deployment: an evolutionary participatory process

There are typically two main actors in wireless community networks (WCNs). The first one is the small group of people who lead the initiative and set up the first wireless nodes (routers). More often than not, they remain involved in the CN throughout its lifetime, undertaking a major role in its maintenance and management. The need to interact with other actors (e.g., municipalities, policy makers, regulating authorities) motivates their organization into various types of legal entities, varying from associations and cooperatives to non-profit (or, rarely, small for-profit) ISPs. Subsequently, we will be referring to them as the CN operator (CNO) entity.

The second main actor in WCNs are the end users who join the network by contributing their own equipment, money and effort. At the same time, through the nodes they add, they expand the geographic coverage of the CN, making it accessible to more people. This participatory network deployment process challenges their sustainability in several ways, whether this is approached from techno-economical only or a broader socio-political point of view [1]. In this paper, we are concerned with the first aspect. In particular, we focus on the challenge of maintaining a sustainable network topology as the CN scales up with the addition of new users. This is a non-trivial problem given that, contrary to conventional commercial ISP networks, the CNs are not built according to systematic top-down planning practices but rather grow bottom-up in response to community interest in them. This way, end users have a direct impact on the emerging CN topology and coverage. The process is evolutionary and may evolve over months and years, in fact, several studies in literature report pathological CN topologies exhibiting high dependence (in terms of connectivity and routing functionality) on a single or a few nodes and high asymmetries in routes and speeds connecting end users to the Internet [2, 3, 4].

One of the contribution of this paper is the introduction of a global forecast function that can provide assistance in this otherwise highly decentralized, almost random, network growth process. This proactive support action can be exerted by the CNO by assisting new users as to which CN node(s) to connect to, and with targeted interventions to address specific structural problems. The fact that the new node installation process is often in the hands of, or at least assisted by, the CNO team, facilitates this type of intervention.

2.2. System model

2.2.1. Set-up of the first CN nodes by the CNO

Generally, the CNO sets-up an initial set of wireless nodes N_0. The locations of these nodes, including an indoor and an outdoor router with antennas, coincides with the houses/residences of the CNO team members or friends of theirs. Their selection may be optimized to maximize the aggregate geographic coverage. Among these nodes there is one or more CN gateway(s) attached to the Internet with a broadband connection. The set-up of these nodes yields an original CN topology graph $G_0 = (V_0, E_0)$, with $|V_0| = N_0$. In this work we assume that $N_0 = 1$, so that no planning can be initially made to shape the evolution of the network. We also assume that the gateway has sufficient uplink capacity for the whole network, so in general this is not a standard ADSL connection but an appropriate gateway set up by the CNO with traditional operators or even directly in an Internet Exchange.

2.2.2. Evolutionary growth of the CN and CNO interventions

The second and main phase in the CN growth process is driven by the population of potential community members who submit requests to join the CN. In doing so, the users may undertake a cost c_u of setting up a node at their home. On the other hand, the set up of a new node n also implies a cost $C_{n,l}$ for setting up the peer point for the link l at the point of attachment to the CN. This cost is closely related to the hardware that will be chosen for the node (wireless outdoor router device and antenna), which in turns depends on the distance and the quality of link l. For example, a cheaper device with smaller wireless range might suffice for attaching to the closest node, whereas a more expensive device with higher range would be needed to reach a more distant one, who may lie closer to the Internet gateway. In general, in wireless multihop networks, distant links reduce the number of wireless hops but tend to induce higher interference, although directive point-to-point links are less affected than multi-point links in mesh networks.

Generally, different criteria may apply when choosing the point of attachment to the CN. If this choice is not subject to some form of control/regulation on the CNO side, a new user might end up connecting to the CN node that can be received most powerfully or a CN node contributed by a friend. However, in our model, we assume that the CNO intervenes in the bottom-up process of the CN formation by choosing itself the point of attachment to the CN among the available alternatives. Practically, this could be realized through managing the new node set up process, or, when this is not the case, through a recommendation to the new node owner.

At every time t when a new node is added to the CN, the CNO tries to ensure that, after the new addition, all CN nodes can obtain a minimum acceptable share of the network capacity. This is directly related to the way the topology $G_t = (V_t, E_t)$ grows over time and the routing policy followed by the CNO. The CN may block a new user joining the CN to add a specific node and link(s) if it figures out that its addition results in unacceptable network performance degradation. Section 2.3 discusses a feasible algorithm to be used as a stop criteria, which is not a trivial problem.

Once they entered the network users may pay a recurring fee which covers the uplink connection and the network maintenance (in economic terms, fees cover the Operating Expenditure). In this paper we are interested in how much such a network can grow, and how much it can cost, so we limit our analysis to the Capital Expenditure (i.e., the initial cost to create the infrastructure).

2.3. Problem formulation as a Multicommodity Flow problem

Consider a finite period of the CN lifetime, beginning at time t_0, when the first CN nodes are set up by the CNO. Without loss of generality, we could take $t_0 = 0$. We would like to track the evolution of the CN up to some time T which marks an horizon for the evolutionary growth of the CN topology.

Requests for the addition of new nodes to the CN come randomly at times $\{t_j\}, t_j \in [0,T], j \in \mathcal{N}$, with $\mathcal{N}$ the set of users who express requests to join the CN. Every time a new node n joins the CN, the following things happen:

- The node attaches to an existing CN node. A new link l of capacity b_l is added, depending on which node it attaches to and what kind of hardware is used for the node. A list of indicative devices used is given in Table 1.
- The new node brings in a new amount of local demand d_n to be served. At the same time, it becomes itself a point of attachment for more nodes that could join the network, implying that the new link could serve demand from additional, currently unknown, nodes.
- Depending on the antenna used (gain, directivity) the new node may create more than one link with existing nodes, thus interfering with existing CN links. It also increases the expected load on all links on its path to the Internet gateway.
- The CNO will check whether the preset performance targets can be preserved after the addition of the new node. If yes, it will simply add the new node. Else, the node is added but the network evolution stops.
- A node that was evaluated at time t but did not have line of sight with any of the existent nodes will be later on evaluated again when new nodes are added.

Ideally, to optimize the link creation, the CNO should solve instances of the Multicommodity Flow (MCF) problem [5]. At any time during the network growth process, as captured by the running digraph $G_t = (V_t, E_t)$, a subset $V_s \subset V_t$ serve as sources of demand $d_i, i \in V_s$ and another possibly overlapping subset $V_e \subset V_t$ serve as destinations for these demands. In our case, V_s is the set of all nodes, V_e contains the gateway, and d_i is the minimum guaranteed bandwidth every node should have. The question is if the network, after the addition of a new node, can satisfy all the demand.

More specifically, the demand distribution is captured into decision variables (*flows*), f_{ij}^l expressing what is the portion of demand appearing in each link l in the network. When single-path routing applies, $f_{ij}^l \in \{0,1\}\ \forall i, j \in V_t,\ l \in E_t$ and the problem is referred to as the Integer Multicommodity Flow (IMCF) problem [6].

The flow feasibility constraints primarily pertain to: (a) the individual link capacities, i.e., the flows through a link l should not sum beyond its capacity; (b) the interference constraints, typically expressed as sets of links that cannot be simultaneously active because they interfere with each other [7].

The optimization objective can then take various forms: maximize the total flow through the CN; maximize the (minimum) percentage of individual demands that can be served; minimize the number of individual demands, who cannot satisfy a target portion of their (elastic) demand. Alternatively, a cost can be defined for each link as a function of the flow it has to serve. The objective could then become to minimize the aggregate (MINSUM) or the maximum (MINMAX) of this cost throughout the CN.

The computational requirements of such an exhaustive optimization approach are prohibitive and diminish its practical value. Eventually, we are after algorithms that could be integrated in a lightweight open source tool that could assist communities building networks in a participatory manner. The "greedy" algorithm we present below is one such algorithm.

The algorithm first computes the minimum cost paths based on the negotiated bandwidth and number of links per device. It then derives estimates of the minimum bandwidth that is available to each node on its route to the gateway and uses them to inform its decision as to whether to accept or reject a user join request. The algorithm relies on information that is made available by state-of-the art routing protocols, namely:

- Each node is able to estimate the available link capacity. Modern routing protocols communicate with the wireless devices to obtain the negotiated bit-rate of the recently sent packets, and use this data as a link metric [8]. The Optimized Link State Protocol (OLSRv2) is one such protocol [9].
- Each node is also able to estimate the number of links that are directly interfering with the device on the receiving side. In the simplest case, this is the number of ingoing links to the destination device, which can be communicated by the neighbor node. There are protocols that use a similar approach to penalize the use of links that share the same channel with more than one neighbor [2].

The pseudocode in algorithm 1 outlines the algorithm. Section 3 explains the practical integration in our CN planning tool.

Algorithm 1 Greedy approach to CN topology control

Input: A graph made by only the gateway node $G_0 = (V_0 = \{g\}, E_0 = \{\})$, minimum acceptable throughput per node R_{thr}, acceptable percent of nodes below threshold p, gateway node g, list of potential nodes P.
Output: A feasible network graph $G = (V, E)$ and its total cost C

$C = 0$
$V_t = V_0$
while True **do**
  $V = P \setminus V_t$
  if $V == \emptyset$ **then**
    return(G_t=(V_t, E_t), C)
  end if
  while $V != \emptyset$ **do**
    V.randomize()
    n = V.pop()
    V_p = get_los(V_t, n)
    n',c = get_best_neigh(n,V_p,R_{thr},p,g,V_t,E_t)
    $C = C + c$
    $V_t \leftarrow V_t \cup n$, $E_t \leftarrow E_t \cup l = link(n, n'))$
    N_{fail}, _ = stop_condition(V_t,E_t,R_{thr},p,g)
    if $N_{fail}/|V_t| < p$ **then**
      return(G_t=(V_t, E_t), C)
    end if
  end while
end while

Function get_loss(V_t, n)
returns the set of nodes in V_t that are in line of sight with n

Function new_link(n,n',G)
computes link attributes: maximum negotiated bit rate and number of links shared on the same device on n'

3. A CN planning tool

We have developed a CN topology generator that implement different CN topology growth strategies and algorithms. These strategies take into account the terrain and radio propagation properties of the geographic area covered by the CN. The produced topologies are annotated with estimates of the available bandwidth in each link and the total amount of money invested in the CN deployment. Different metrics can then be used to assess their critical features such as the expected throughput they provide to end users and their resilience to failures.

The generator consists of three main components:

- A database of open data;
- A stochastic engine that implements the different topology growth strategies, i.e., it runs a continuous iteration that selects one potential new node to add to the CN based on a probability distribution that depends on the topology and seek for a possible connection point based on the growth strategy; and
- A library of metrics that assess graph properties.

Before we present them in detail, let us introduce the necessary terminology. In our simulator a "node" is an installation of a mesh node, which is made of an indoor part and an outdoor part. The former is a wireless router which is configured to redistribute Internet access in the user's house and to perform the IP routing. The latter is made of a metallic pole, a PoE (Power over Ethernet) switch and a number of outdoor wireless devices ("devices" for brevity). From the home router one Ethernet cable brings connectivity and power to the outdoor devices. The cost of a node is thus split into a fixed part for the router, switch and installation material, and a variable part for the outdoor wireless device which we will detail later on.

3.1. The open data database

The CN simulator analyzes data from three sources:

- Street maps including building shapes, taken from OpenStreetMap (OSM) and other public open data repositories. Whereas OSM is generally very precise in urban areas, open data sets from public administrations tend to be more precise, though less up-to-date, in rural areas. In some cases (e.g., in

[2] See for instance the Batman-adv multi-link optimization https://www.open-mesh.org/projects/batman-adv/wiki/Network-wide-multi-link-optimization

Name	Avg. Price (EUR)	Beamwidth Angle (H,V degrees ^{o})	Sensitivity (dBm)	Max TX Power (dBm)	Antenna Gain (dB)	Max distance (km)
ISO90	200	90,30	-65	21	14	1.34
ISO45	112	45,45	-65	21	14	1.34
LB	73	20,10	-65	21	23	3.79
NB	100	30,30	-65	20	19	2.39
NS	134	60,20	-65	21	16	1.69
NSL	49	50,40	-65	21	13	1.20
PB3	110	20,10	-65	21	22	3.38
PB4	129	20,10	-65	21	25	4.77
PB5	185	20,10	-65	18	27	6.00

Table 1: The technical specifications of the Ubiquiti 5G-ready devices included in the simulator, limited to the highest Modulation and Coding Scheme (MCS9).

France) the public open data sets have already been imported in OSM.

- Buildings altitude profiles obtained from LIDAR traces. Several public bodies have published open data from LIDAR survey campaigns, with various level of precision[3]. Public administrations publish much more precise data sets; the one we use reaches a precision of one point per squared meter.
- A database of technical specifications of real 5G-ready (802.11ac) devices, from the Ubiquiti equipment manufacturer. For each device, the database stores the maximum transmission power, the antenna gain and aperture, the sensitivity for each supported Modulation and Coding Scheme (MCS), and an average price, extracted from official stores (dated Sept. 2018). Table 1 reports the features of the devices we have used in the simulations of section 4.

With these sources of data we can verify if, placing devices on the roofs of any two buildings, we can expect to achieve line of sight (LoS) between them. If this is the case, we compute the theoretical path loss with an attenuation exponent set to 2, and then we refine it through a single knife edge approximation model, which takes into account the occupation of the Fresnel zone [10]. We apply the European regulatory limits that sets an upper bound of 30 dBm to the emitted power in the unlicensed band around 5 GHz, and, using the antenna gain and the sensitivity reported in the data sheet we estimate the negotiated bandwidth of the link. We assume each device is configured in ad-hoc mode (or equivalent). A typical problem in mesh networks is how to assign channels to links in order to avoid interference with neighboring nodes. The problem can be described as a variant of a graph coloring problem and it is NP [11], but if the number of devices per node is sufficiently smaller than the number of available channels, any local heuristic can succeed in avoiding interference. In order to reinforce this condition, we use 20MHz channels, which limits the available bandwidth per link but ensures the presence of at least 12 non-overlapping channels[4].

[3]The Shuttle Radar Topography Mission: https://www2.jpl.nasa.gov/srtm/ is a public repository of terrain elevation profiles, but its precision does not allow to estimate building height (roughly one point every $900m^2$).

[4]The 802.11ac standard defines 25 independent channels but only 12 of them are usable for outdoor applications in Europe

3.2. Topology growth strategies

Each node in the CN topology is mounted on a real building. The first node of the network is the gateway node. This node is chosen manually in order to avoid pathological conditions. We typically choose a building that has reasonable connectivity such as the City/Town Hall, a hospital, or a university and we assume the availability of a 10m trellis structure that can be used to mount up to eight wireless devices.

Then we generate a random sequence of requests from end users who want to join the CN. For each such request, mapped to a unique building across the city, we apply a local decision heuristic to determine to how many existing CN nodes and which ones, the new node will be connected to. We assume that every non-gateway node can bear a pole up to 2 meters high that can mount up to 4 devices. The appearance of requests to join, preference for attachment and numbers reflect our practical experience with real world CNs.

A particular topology growth strategy should then specify, at least, two elements: a) the *neighbor choice* algorithm, which determines to how many nodes, and which ones, a particular node is connected tol b) a *topology-growth stopping criterion*, which determines the point in which no more nodes can be added to the network.

3.2.1. Neighbor choice algorithm

Let $G_t = (V_t, E_t)$ be the CN graph at the time t that a new node n requests to join the network and g be the gateway node. As a baseline algorithm for the neighbor choice, we consider the local heuristic described formally in Alg. 2 and summarized below.

local heuristic: Given n, find the subset V_p of nodes in the graph that potentially have line of sight with n. For each $n' \in V_p$ compute the expected path loss and find the two devices that need to be added to n and n' to obtain the maximum negotiated bit rate. Pick the node $n' \in V_p$ that guarantees the highest bit rate. In case n' already has the right device pointing towards n and operating on channel k, then add a new device only on n and use channel k. Otherwise add a new device also to n' operating on a channel that is not already used by any device on n'.

The local strategy simply tries to maximize the available bandwidth on the link connecting the new node n to some existing node n'. This is a plausible strategy, as it tries to myopically create the seemingly best link among all the possible ones.

Algorithm 2 Neighbor choice strategy: local

Input: n,V_p (ignore other parameters)

Output: Neighbor that maximises the bandwidth of the new link

```
link_bw = dictionary()
link_cost = dictionary()
for n' in V_p do
    if n' has an antenna pointed towards n then
        link_bw[n] = compute the maximum possible bandwidth
        considering a new device for node n and an existing device
        for n'
        link_cost[n] = cost of the new device for n
    else
        link_bw[n] = compute the maximum possible bandwidth
        considering a new device for both n and n'
        link_cost[n] = cost of the new devices for n and n'

    end if
end for
best_neigh = node that provides the best negotiated bandwidth,
if there is a tie choose the cheapest solution
cost = the cost associated to best_neigh
return(best_neigh, cost + 200)
```

Algorithm 3 Neighbor choice strategy: network-aware

Input: n,V_p,E_p,R_{thr},p,g,V,E

Output: Neighbor that minimizes the network-wide performance drop due to the new node bandwidth requirements

```
B = dictionary()
for n' in V_p do
    V'_p = V_p ∪ n
    E'_p = E_p ∪ l = new_link(n,n')
    _, B_min = compute_stop_condition(V'_p, E'_p, R_thr, p, g)
    B[n'] = B_min
end for
return the node corresponding to maximum B_min
```

Algorithm 4 Topology growth stopping criterion

Input: V,E,R_{thr},p,g,$C = constant$

Output: Nodes below threshold, lowest guaranteed bandwidth

```
path_dict = dictionary()
paths_per_edge = dictionary()
for l in E do
    l.metric = C*l.sharing_factor/l.max_bandwidth
end for
for n in V do
    path = compute_dijkstra(n, g)
    path_dict[n] = path
    for l in path do
        paths_per_edge[l] += 1
    end for
end for
bottleneck = list()
nodes_below_threshold = 0
for n in V do
    b = min([l.max_bandwidth/paths_per_edge[l] for l in
    path_dict[n]])
    bottleneck.append(b)
    if b < R_thr then
        nodes_below_threshold += 1
    end if
end for
return(nodes_below_threshold, min(bottleneck))
```

In this work, we compare this local heuristic with an alternative approach that tries to factor the network-wide view in the otherwise local decision. The algorithm is presented as pseudocode in Alg. 3 and summarized below.

network-aware heuristic: Given n, find the subset V_p of nodes in the graph that potentially have line of sight with n. For each $n' \in V_p$ temporary add the link l from n to n' to the network graph and evaluate the stop condition (see 3.2.2). Call B_{min} the minimum bandwidth guaranteed to any existing node. Pick the node n' that produces the highest B_{min}. In case of a tie, pick the neighbour that produces the link with the maximum bandwidth.

The network-aware heuristic does not directly add the fastest new link, it evaluates all the potential nodes with respect to their impact on the performance of the whole network. Only if there is a tie, it picks the fastest link. Note that if every time we add a node together with a single link, the CN would become a tree. Actually, when we add a device to a node we always check if the new device can generate a connection with any node other than the intended one (on the same channel). We allow a maximum of 3 outgoing links per device, in order to limit the number of links that share a given device.

3.2.2. *Topology growth stop condition*

Irrespective of how the neighbors of a new CN node are chosen, at some point the total demand will exceed the aggregated network capacity, and the network should stop accepting new nodes. In our simulator, the stop condition relates to the minimum bandwidth that can be guaranteed to a CN node all the way to the CN gateway.

Topology growth_stopping_criterion(*min_bandwidth*, *percentile*) For each directional link $l = (n,m)$ in the CN, from node n to a node m, compute a link metric:

$$l.metric = \frac{C \cdot l.sharing\_factor}{l.max\_bitrate} \quad (1)$$

where *max_bitrate* is the maximum achievable bit rate on the link l, *sharing_factor* is the number of incoming links on m that share the same device, and C is a constant. The metric reflects cost and can be used with the Dijkstra algorithm.

For each node n compute the minimum-cost path towards the gateway g. For each link, count the number of paths that pass through it and divide its *max_bitrate* for that number. For each node n, the minimum available bandwidth is given by the minimum bandwidth on all the links in the path to g. If the fraction of nodes that have less than *min_bandwidth* is higher than a target *percentile*, the network stops to grow.

In reality, when the network performance degrade enough to trigger the stop condition, rhe community may also decide to collectively support the creation of another gateway to unload the existing one. We will take this option into consideration in our future work based on the simulator.

The network stops growing also in three more cases: i) all nodes have been added; ii) the unconnected nodes do not have line of sight with any connected one; iii) unconnected nodes have line of sight with some node that already saturated the number of devices, and these devices are pointing in another direction.

3.3. Graph metrics

Once a CN graph is generated, we evaluate two metrics:

The network size: The final number nodes with a path to the gateway when the stopping criterion is satisfied or no new nodes can be added.

The average cost of a node Every time a new node is added to the CN, the total cost C of the network is increased by a fixed cost (200€, generally enough for the Ethernet devices plus installation material) plus the cost of the new devices. In order to take into account the re-use of existing devices, we need to estimate if a node falls inside the cone created by an antenna attached to an existent device. To do this, from the devices' datasheets we extract the angle at which the device exhibits < 3dB loss, and we consider that as the real antenna aperture. When we add a link l from a new node n to a neighbor n', l may be attached to an existing device in n'. In this case the cost is increased by the price of one device only but the link capacity is split by the $sharing\_factor$ as explained before. Else, if no existing antenna can be re-used and n' features fewer than 4 devices, two devices are added and the cost of both nodes is increased.

These metrics are sufficient to compare the impact of strategies and configuration parameters on the required investment to set-up the network, which is the goal of this paper. More elaborated metrics can be used to estimate the operating expense of the network. Among them we mention metrics expressing the graph robustness, which are key to asses the self-healing properties of the network. For instance, the number of cut-points is a key indicator to show how much a network is resistant to node failures, and thus to estimate recurrent maintenance costs. In section 4.1 we give a qualitative evaluation of network robustness, which we will investigate in future work.

4. Experimental evaluation

In this work, we use measurement data from four areas in the Tuscany region, in Italy. There are three good reasons for this choice: first, topological data are available for these areas; second, some of the authors are original from this region easing the access and interpretation of data that are often in the local language and third, the familiarity with the area helps in the correct interpretation of results. To facilitate the extension of this analysis to areas outside Italy we used a territorial characterization that is standardized by the OECD (Organisation for Economic Co-operation and Development) and divides regions into "predominantly urban", "intermediate", and "predominantly rural" [12]. This classification is based on population density, percentage of country population in the area and the presence of large cities, and is intended to facilitate numeric comparisons between similar areas in different countries. In Italy, the OECD categories are further split to make the classification more precise. A subset of urban areas is defined as "rural with intensive agriculture" including prevalence of flat lands that are generally close to a city and have an advanced economy. Rural areas are separated into "rural intermediate", which include hilly and some mountainous areas far from cities, and "rural with development problems", which include mostly mountainous areas that are generally far from cities and face development problems. Table 2 summarizes the main features of the four scenarios we consider, which are implicitly re-labeled urban, suburban, intermediate, and rural.

Scenario	Buildings	km^2	Buildings/km^2
Urban	43853	102	429
Suburban	6663	45	148
Intermediate	2052	34	60
Rural	4414	182	24

Table 2: The density of buildings in the selected scenarios.

4.1. Simulation results

In this section, we describe a single run of our tool, in order to give a qualitative interpretation of the network evolution. We take as an example one network evaluated in the intermediate scenario. In the next subsection we will discuss aggregate results over sets of multiple runs.

We apply the stopping criterion formulated in section 3 with $min\_bandwidth = 1Mb/s$ and $percentile = 10$. The network stops growing when more than 10% of the nodes have less 1 Mb/s guaranteed bandwidth. We stress that this is an estimation of the minimum guaranteed bandwidth, which means that in normal condition (when not every single user is overloading the network) users will enjoy a much higher connection speed. ISPs generally use a 'contention ratio' of X:1, thus advertising speed that can be X times higher than the minimum guaranteed bandwidth per node. Typical contention ratios are in the order of 50:1, as recommended by the British telecom regulator authority in 2016[5]. Hence, a minimal guaranteed bandwidth of 1 Mb/s can be advertised as a 50 Mb/s connection, which is comparable to a typical FTTC connection.

In the intermediate scenario the simulator was able to generate a CN of 227 nodes, before the stop condition was triggered. The final CN graph has 907 (bi-directional) edges and the average CN node degree is roughly 8. Since the average number of devices per node is 3.67, we conclude that each device is connected to an average of 2.44 neighbors. Figure 1 reports the Empirical Cumulative Distribution Function (ECDF) of the node degree values in the CN. Roughly 80% of the nodes have fewer than 12 neighbors,

[5]The 2016 recommendation of the British Regulator considers 50:1 a good contention ratio for a standard broadband profile[13]

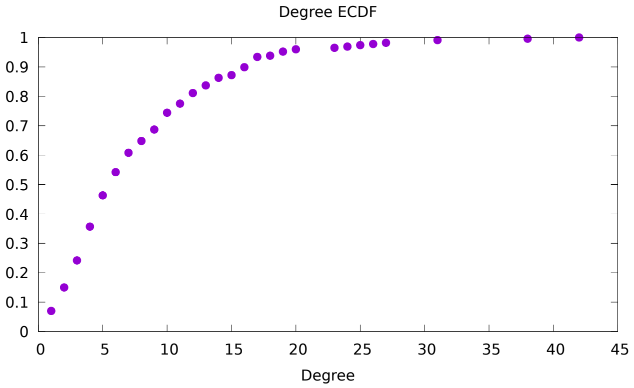

Figure 1: The Empirical Cumulative Density Function of the node degrees.

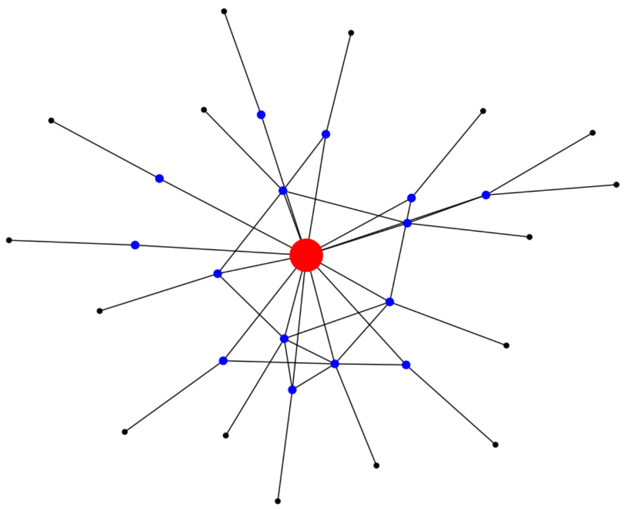

Figure 2: The block-cut tree representation of the network.

but there are a few outliers with to 42 neighbors. The reason for this is that, although we limit the number of outgoing links per device to three, we do not limit the number of incoming links. Since the scenario area is geographically small, some nodes in the center of the area are connected to many neighbors. Note that all these links are not necessarily used to route traffic to the gateway, but they offer redundancy in the case of failures.

Figure 2 shows the Block-Cut tree representation of the graph, in which every red node represents a biconnected component, every blue node is a cut-point and every black node is a leaf node[6]. The number of cut-points is 15, and every cut-point (but one) is connected only to a single leaf node. This means that only 15 nodes out of 227, if removed from the network, render the gateway unreachable for at most one (two) more node(s). The network is indeed quite robust.

Figure 3 reports the minimum bandwidth for all the CN nodes. The minimum bandwidth is quantized, as it is given by the bandwidth of the bottleneck link from any node to the gateway. The

[6]A cut-point is a node that, if removed from the network, disconnects the network in separate components. A biconnected component is a sub-graph without cut-points, which remains connected even if one of the nodes fails.

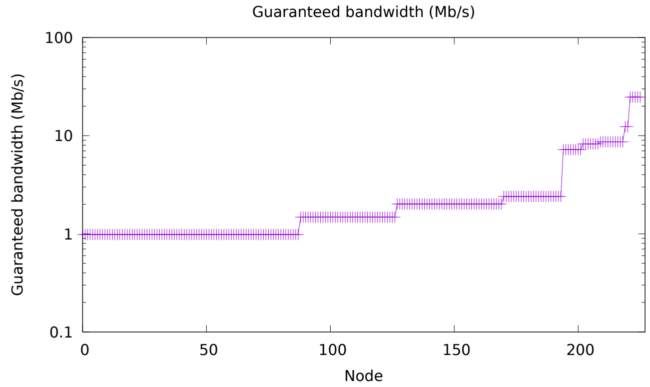

Figure 3: The minimum guaranteed bandwidth for all the nodes in the final network.

same link is a bottleneck for more than one node, so groups of nodes end up with the same minimum bandwidth. Note that there are 88 nodes ($\simeq$39% of the total) with a minimum guaranteed bandwidth of 0.98 Mb/s. This happens because before the last node was added to the network, one link with an estimated bandwidth of 87 Mb/s was used by 87 nodes. When the last node was added, this pushed the bandwidth of all the 88 nodes using that link below 1Mb/s and terminate the CN growth process.

Figure 3 shows that there is some unused capacity in the network. Imagine that l is a link that is the bottleneck for a sub-graph C of the network. The available bandwidth $b(l)$ must be divided among the $|C|$ nodes that use it to reach the gateway. If $\frac{b(l)}{|C|} \geq 1$ then hypothetically more nodes could be added to C, and the network could grow larger. In this case this did not happen because the stop condition was hit before. Figure 3 suggests that a strategy that tries to better distribute the load among the existing bottleneck links could allow the network to scale. This observation serves as the main motivation for the network-aware neighbor choice heuristic.

Figure 4 reports the 10th, 50th and 90th percentile of the distribution of the guaranteed bandwidth per node against the number of nodes in the network. In line with intuition, as the network grows all percentiles decrease, meaning that the overall available bandwidth decreases with the number of nodes. The 10th and 50th percentile are very close, as it emerges also from Figure 3 which corresponds to the last point of fig. 4, in which the large majority of the nodes (194 over 226) have less than 3Mb/s bandwidth.

Figure 5 reports the trend of the average cost per node in the network, together with the average number of devices per node. As said, for each node we assume a fixed cost (200€) to which we add a cost for each device, given by the average market price we found on on-line resellers of Ubiquiti devices. Prices range from 49€ to 200€, but the difference in the performance of the devices emerges only on very long links. For links below a few kilometers the mid-range devices perform as the high-end devices, so high-end devices are rarely used. For this reason the average cost of the nodes is directly correlated with the number of devices per node.

5. Results on Multiple Runs

This section documents the simulations under the four scenarios (urban, suburban intermediate, rural) when the minimum guaranteed

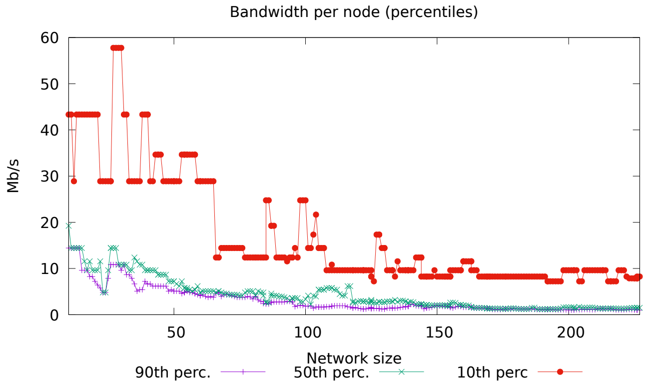

Figure 4: The 10th, 50th and 90th percentiles of guaranteed bandwidth for a growing network.

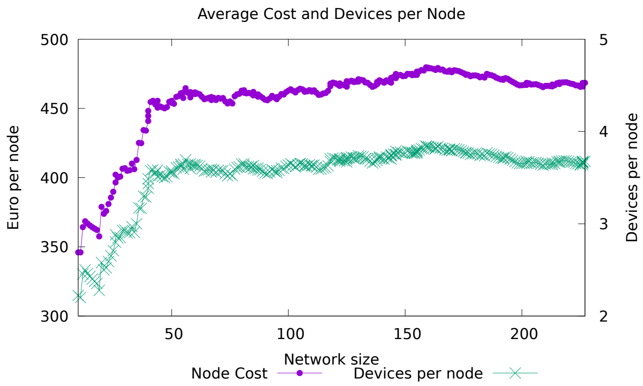

Figure 5: The average cost and number of devices per node.

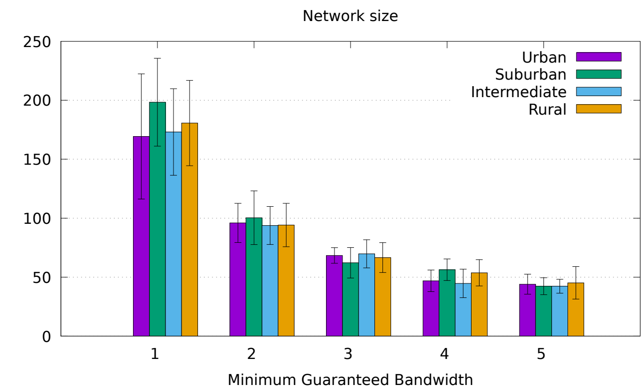

Figure 6: The number of nodes in the network.

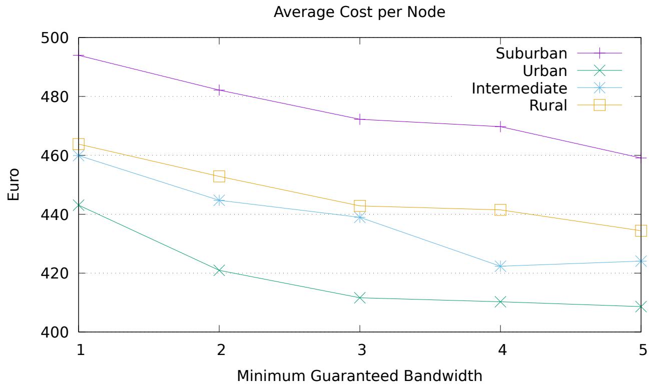

Figure 7: The average cost per node.

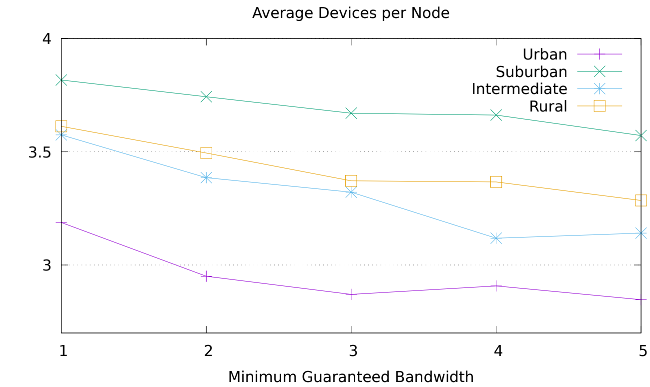

Figure 8: The average number of devices per node.

bandwidth per node ranges from 1 to 5 Mb/s and the stop condition is parameterized to $percentage = 10\%$. Each point in the graph is realized as the average of 10 runs and the error-bars represent the standard deviation. In each run we choose the same gateway and vary the random seed.

Figure 6 shows the average number of nodes in the network under the four scenarios. With a minimum guaranteed bandwidth of 1Mb/s the average size of the network reaches 200 nodes, without significant differences between the four scenarios. Moreover, the size of the network decreases roughly as the inverse of the guaranteed bandwidth. Hence, what primarily influences the maximum network size is the available bandwidth at the gateway, and not the scenario. In other words, using high-performance devices reduces the probability that the links that are far away from the gateway become bottlenecks. As we have observed in fig. 3, there are links to the gateway that are not fully exploited. Improving the utilization efficiency of these links benefits all scenarios.

Figures 7 and 8 reports the average cost and number of devices per node and clearly differentiates one scenario from another. Two factors dictate the number of devices per node: the choice of the neighbor when a new node is added and the terrain characteristics. We will discuss the former in the next subsection, while for the latter, it is intuitive to note that the building density plays a key role. When a new device is added, the antenna creates a cone in which future node could be connected. The higher the building density, the higher the chances that a new node could be connected to an existing device. On the other hand, buildings are themselves radio propagation obstacles, which constrain the LoS between two network nodes.

Apparently, in the urban scenario the density of buildings suffices to re-use many of the existing devices, while the worst trade-off is represented by the suburban area. Figure 7 shows that the impact of the scenario on the variable part of the node cost is pretty high (recall that we consider 200€per node as a fixed cost). It would be extremely interesting to characterize how the efficiency of the

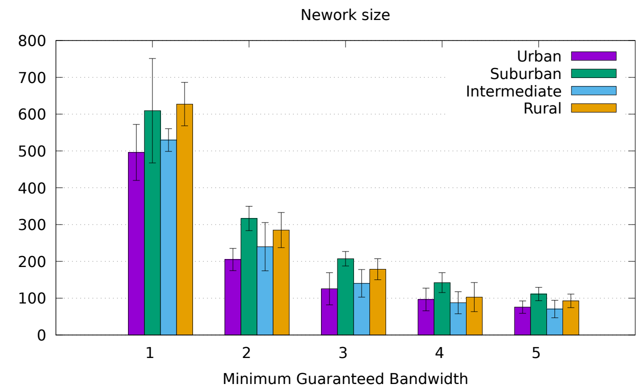

Figure 9: The number of nodes in the network.

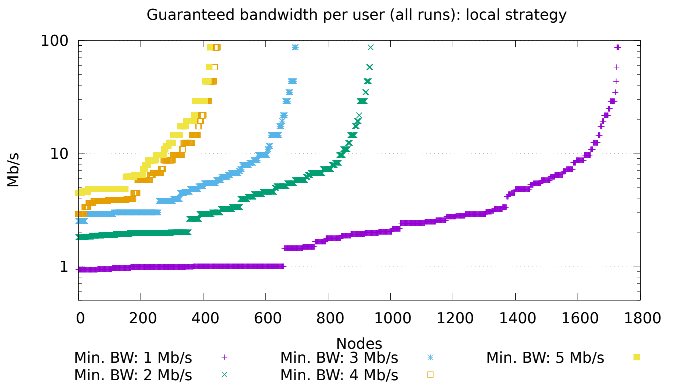

Figure 10: The guaranteed bandwidth to all nodes in all runs: intermediate scenario, local strategy.

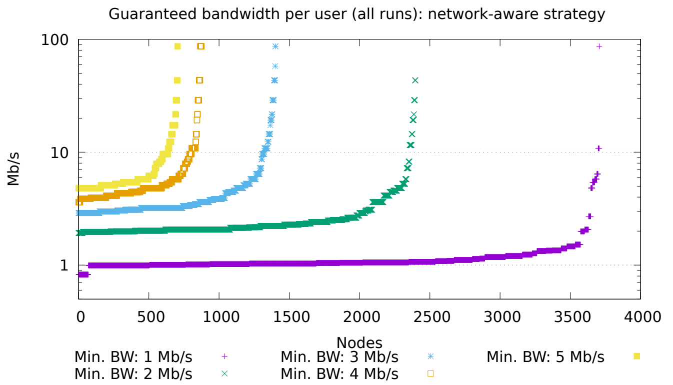

Figure 11: The guaranteed bandwidth to all nodes in all runs: intermediate scenario, network-aware strategy.

mesh network technology varies with the scenario and what are the design parameters we can modify (number of devices, type of devices, choice of neighbors) to make it perform better in each case. With only one data-set for each of the four area types, we cannot draw generic conclusions; but we will explore this correlation in our future works.

5.1. Comparison of different topology growth strategies

This section compares the local and the network-aware strategies regarding the neighbor choice of new nodes. Figure 9 shows the average size of the network when the network-aware connection strategy is used to direct the connection of new nodes. Compared to the data reported in fig. 6, the gain is considerable: we can more than double the number of nodes in all the scenarios for almost all the guaranteed bandwidth values.

As we commented in fig. 3 for a single run of the intermediate scenario, the local heuristic fails to evenly distribute the load on the links to the gateway. Figures 10 and 11 generalize the data shown in fig. 3 reporting the guaranteed bandwidth for all nodes and all runs, and the neighbor choice algorithms under the same intermediate scenario. The two plots suggest that, excluding a small number of outliers, the network-aware strategy produces a much fairer distribution as most of the nodes achieve a guaranteed bandwidth close to the minimum acceptable. In other words, the network-aware heuristic is able to structure the network in a more efficient way and better distribute the load across gateway links.

Another positive effect is the considerably lower cost per node. The local greedy algorithm myopically tries to provide the highest capacity to each new link. This often results in the addition of new devices on existing CN nodes since a new device always performs better than a shared device. However, the high-capacity links in the fringes of the network do not help much with increased capacity needs in the paths towards the gateway. The network-aware algorithm, instead, has no specific incentive to add new devices unless this produces a network-wide improvement. Figures 12 and 13, when put side-by-side with figs. 7 and 8, show that in fact, both the average number of devices per node and the node cost decrease when factoring the network-wide viewpoint in the local decisions.

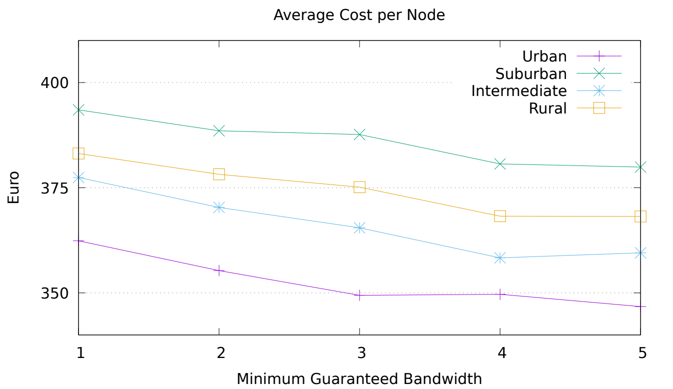

Figure 12: The average cost per node.

6. Related work

Our work combines elements from two, originally distinct, operations: *network planning* and *topology control*. The first one is a longer-term centralized process that is carried out top-down and concerns primarily static multihop networks. Topology control, on the other hand, is carried out over shorter time intervals and is more relevant to mobile ad hoc networks and sensor networks. It involves the distributed control of the transmit power of nodes in order to

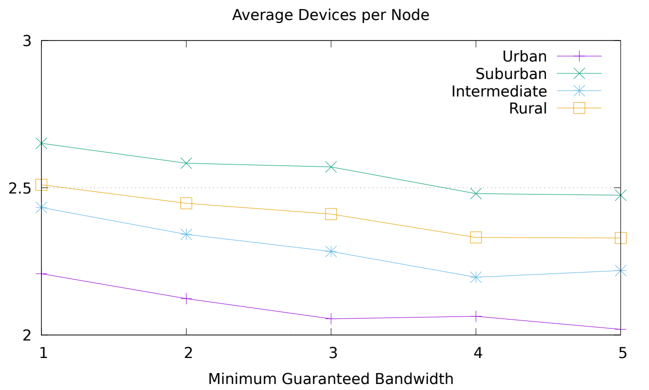

Figure 13: The average number of devices per node

achieve certain performance objectives such as energy savings or resilience to node and link failures.

Recalling two of the most classic contributions to the topic, the authors in [14] propose, analyze and optimize a cone-based distributed topology-control algorithm for minimizing the transmit power of ad hoc network nodes, while preserving their connectivity. At the core of the algorithm lies the finding that, when focusing the nodes' transmissions within cone areas, whose width does not exceed a critical threshold, there is some minimum power that can ensure their connectivity. On the other hand, Li and Hou in [15] are concerned with routing redundancy as a response to the increased risk of node failures or departures. Hence, they propose a fully localized algorithm that enforces k-vertex connectivity in the constructed topology while minimizing the maximum transmission power used in the network.

More relevant to our work is the thread on wireless mesh network planning, which has been primarily driven by the need to deploy wireless mesh networks in rural areas with limited or no network coverage. In [16], Chandra et al. consider the placement of Internet gateways (they call them Internet transit access points (ITAPs)) in a wireless neighborhood mesh network route traffic from residential nodes equipped with low-cost antennas to the Internet. They develop ITAP-placement algorithms that perform close-to-optimally over a number of scenarios addressing the neighborhood layout, end user demands, and the propagation environment.

A more holistic approach, including the selection of tower heights and antenna types, and aiming directly at minimizing the network infrastructure cost, is introduced in [17]. Motivated by projects addressing the digital divide in rural India, the authors formulate the planning problem and then decompose it, exploiting dependencies between the different design variables and heuristics. They report cost-effective topologies for real deployment cases, within about 2% of a lower bound they compute.

The cost of antenna towers, as a function of its height, is the main concern also in [18], which assumes that there is reasonable flexibility as to where to set up such towers. The authors seek to optimize the selection of links to establish such that all nodes are connected and the resulting cost of antenna tower construction is minimized. They propose a greedy algorithm that provides an $O(logn)$-approximation for they prove to be an NP-hard problem.

Last, the wireless mesh network deployment costs are subject to a budget constraint in [19], where the objective is set to maximize the coverage of the users while ensuring that the network is resilient to node failures. The authors propose an approximation algorithm called greedy selection rounding (GSR), which persistently generates topologies with coverage at least 95% of the optimal at a cost that does not exceed by more than 15% their budget.

Our main distinguishing feature from the research work in [16]-[19] is that the topology/network planning process is not centrally directed. In our case, the network grows bottom-up, in participatory manner, with users' locations determining the possible node and link additions. The CNO team tries to intervene *online* to the network growth process, to shape it in ways that facilitate their sustainable scaling.

Finally, on the topology generators' front, we have counted only a few attempts in the literature to build WMN planning software tools starting from the observation of real world mesh networks. Among them Cerdá-Alabern studied the topological features of the Guifi.net network [20] and derived a corresponding generator function, while Milic and Malek studied two networks of the Freifunk community and produced a geometric model that seems to be the most accurate model for mesh networks topology generation [21]. These models capture the macroscopic features of a network (like the degree distribution) but again, lack important details to characterize their behavior in the real world.

7. Conclusions

Community Networks are growing in many areas in Europe and beyond, offering a novel techno-social organization for Internet access and distributed digital services. Technically they are primarily Wireless Mesh Networks, though in some cases they can be hybrid networks or, more rarely purely optical fiber networks. WMNs have the huge advantage of low cost, easy bootstrapping and 'organic' growth, meaning that new users can join the network simply adding a node on their house and connecting it to existing nodes. This last advantage, however, contains a potential poisonous grain: As the network grows following basically a stochastic process constrained by the local topography and by the attitude of newcomers to connect either to a friend's node or to the simplest to reach, the topology of the network is unplanned, and often encounter scalability problems and stop its growth, e.g., because the aggregate bandwidth toward the gateway becomes a bottleneck, with a dimension much smaller than what it could have with a proper, centralized, a-priori planning. Such planning, however, is simply alien the the concept itself of Community Network.

This work has presented the community three key contributions. First of all, for the first time, the way Community Networks grow has been modeled as an (implicit) stochastic graph evolution. Based on this conceptual model, we have been able to build a tool that simulate the evolution of a Community Network given the topography and the building data-base of the area where the network is developing. Running the tool on urban, sub-urban and rural areas in Tuscany, Italy, where we know there is interest for founding CNs, we have shed light on the reasons these networks may fail, i.e., we were able to explain why a network reaches a saturation point where the available capacity becomes unsatisfactory for a large percentage of users, ending the 'interest' of new users to join the network. Thanks to these insights, it has been possible to formulate simple and local countermeasures that the Community Network operator itself can deploy to influence the way new nodes connect to the CN to improve the topology characteristics. These countermeasures, that are heuristics inspired by the problem formulation as a Multicommodity Flow, have been implemented in the simulation tool and they show that a network can grow to a size 2–3 times larger than a simple greedy growth would allow. Indeed, the simulation tool turns out to be also a powerful planning tool in the hand of a CN operator, as it allows estimating the impact of adding a node, optimizing the device this node should install to avoid creating bottlenecks, or becoming a bottleneck later on. This way the CN operator can arrange proper incentive schemes to steer the growth of the network for the benefit of the entire community and for the sustainability of the CN in the medium-long term.

References

[1] P. Micholia, M. Karaliopoulos, I. Koutsopoulos, L. Navarro, R. Baig, D. Boucas, M. Michalis, P. Antoniadis, Community networks and sustainability: a survey of perceptions, practices, and proposed solutions, IEEE Communications Surveys & Tutorials.

[2] D. Vega, L. Cerda-Alabern, L. Navarro, R. Meseguer, Topology patterns of a community network: Guifi. net, in: Wireless and Mobile Computing, Networking and Communications (WiMob), 2012 IEEE 8th International Conference on, IEEE, 2012.

[3] L. Maccari, R. Lo Cigno, A week in the life of three large wireless community networks, Ad Hoc Networks 24, Part B (2015) 175–190.

[4] L. Maccari, On the Technical and Social Structure of Community Networks, in: IFIP Internet of People Workshop,IoP, Vienna (Austria), 2016.

[5] R. K. Ahuja, T. L. Magnanti, J. B. Orlin, Network Flows: Theory, Algorithms, and Applications, Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1993.

[6] C. Barnhart, C. A. Hane, P. H. Vance, Integer multicommodity flow problems, in: P. M. Pardalos, D. W. Hearn, W. W. Hager (Eds.), Network Optimization, Springer Berlin Heidelberg, Berlin, Heidelberg, 1997, pp. 17–31.

[7] K. Jain, J. Padhye, V. N. Padmanabhan, L. Qiu, Impact of interference on multi-hop wireless network performance, in: Proc. 9th Annual International Conference on Mobile Computing and Networking, MobiCom '03, 2003, pp. 66–80.

[8] H. Rogge, E. Baccelli, Request for comments 7797: Directional airtime metric based on packet sequence numbers for optimized link state routing version 2 (olsrv2) (2016).
URL `https://tools.ietf.org/html/rfc7779`

[9] C. Barz, C. Fuchs, J. Kirchhoff, J. Niewiejska, H. Rogge, Olsrv2 for community networks: Using directional airtime metric with external radios, Computer Networks 93 (2015) 324 – 341, community Networks.

[10] C. Sommer, S. Joerer, M. Segata, O. K. Tonguz, R. Lo Cigno, F. Dressler, How Shadowing Hurts Vehicular Communications and How Dynamic Beaconing Can Help, IEEE Transactions on Mobile Computing 14 (7) (2015) 1411–1421. doi:10.1109/TMC.2014.2362752.

[11] A. Raniwala, K. Gopalan, T.-c. Chiueh, Centralized channel assignment and routing algorithms for multi-channel wireless mesh networks, SIGMOBILE Mobile Computing and Communications Review 8 (2) (2004) 50–65.

[12] O. for Economic Co-operation, Development, Defining and describing regions (2011). doi:https://doi.org/10.1787/reg_glance-2011-4-en.

[13] O. UK Office of Communications, Achieving decent broadband connectivity for everyone technical advice to uk government on broadband universal service (2016).
URL `https://www.ofcom.org.uk/__data/assets/pdf_file/0028/95581/final-report.pdf`

[14] L. Li, J. Y. Halpern, P. Bahl, Y.-M. Wang, R. Wattenhofer, A cone-based distributed topology-control algorithm for wireless multi-hop networks, IEEE/ACM Transactions on Networking 13 (1) (2005) 147–159.

[15] N. Li, J. C. Hou, Localized fault-tolerant topology control in wireless ad hoc networks, IEEE Transactions on Parallel and Distributed Systems 17 (4) (2006) 307–320.

[16] R. Chandra, L. Qiu, K. Jain, M. Mahdian, Optimizing the placement of internet taps in wireless neighborhood networks, in: Proc. 12th IEEE ICNP 2004, 2004, pp. 271–282.

[17] S. Sen, B. Raman, Long distance wireless mesh network planning: Problem formulation and solution, in: Proc. ACM WWW '07, 2007, pp. 893–902.

[18] D. Panigrahi, P. Dutta, S. Jaiswal, K. V. M. Naidu, R. Rastogi, Minimum cost topology construction for rural wireless mesh networks, in: Proc. IEEE INFOCOM 2008, 2008, pp. 771–779.

[19] C. Chen, C. Chekuri, D. Klabjan, Topology formation for wireless mesh network planning, in: IEEE INFOCOM 2009, 2009, pp. 2671–2675.

[20] L. Cerda-Alabern, On the topology characterization of Guifi.net, in: IEEE 8th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), 2012.

[21] B. Milic, M. Malek, Npart-node placement algorithm for realistic topologies in wireless multihop network simulation, in: Proceedings of the 2nd international conference on simulation tools and techniques, 2009.