WIP: Analysis of Feasible Topologies for Backhaul Mesh Networks

Gabriele Gemmi
University of Venice
gabriele.gemmi@unive.it

Renato Lo Cigno
University of Brescia
renato.locigno@unibs.it

Leonardo Maccari
University of Venice
leonardo.maccari@unive.it

***Abstract*—Mesh backhauls are getting attention for 5G networks, but not only. A backhaul mesh is attractive due to its multiple potential paths that grants redundancy and robustness. The real topology and its properties, however, is heavily influenced by the characteristics of the place where it is deployed, a fact that is rarely taken into account by scientific literature, mainly due to the lack of detailed topographic data. This WIP analyzes the impact of true topography on small backhaul meshes in nine different locations in Italy. Initial results stress how true data influence results and can help designing better networks and better services.**

I. Introduction and Data Sets

Due to the availability of high-speed point to point wireless links, wireless mesh backhauls are becoming a hot topic of research, and several recent works analyze them from different perspectives: reliability [1], energy efficiency [2], or cost [3]. The recent introduction of wireless backhaul in the 5G standardization documents (referred to with Integrated Access and Backhaul, IAB) have further increased the interest in this topic [4]–[6], because 5G (and beyond) requires a densification of base stations that can be achieved only with wireless backhauls, especially in rural areas. Out of the cellular network application, mesh networks have been used and are still used to provide connectivity where traditional networks cannot be deployed or are not profitable [7]–[9].

A mesh network, however, can be just as good (robust, resilient, with high capacity, ...) as the localization of the nodes allows: if the nodes form a line, then the topology will be a bus, and if a link breaks the network gets disconnected; if two nodes are very close one another, but a hill or a tall building blocks the Line-of-Sight (LoS) between them, then it will be impossible to set up a high capacity link between them, and so forth. The topographic data needed to design experiments on realistic topologies, however, is not easy to collect, albeit it is often available as Open Data, and it requires a thorough check and pre-processing to guarantee its quality, so very often scientific papers are based on abstract models or assumptions, the simplest of which is full visibility of all nodes, specially if they are placed on tall buildings in a small area. This paper leverages the analysis of real data extracted by the tri-dimensional building shapes of 9 Italian municipalities, reported in Tab. I: three are Urban, three Suburban, and three

	Urban	Suburban	Rural
1	Trento	Mezzolombardo	Predaia
2	Firenze	Pontremoli	Barberino
3	Napoli	Sorrento	Visciano

TABLE I: The nine municipalities in Italy selected for the study.

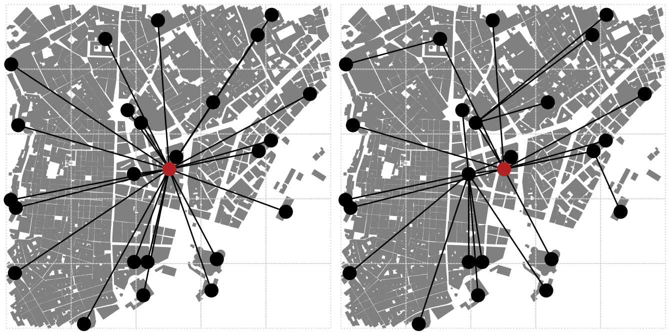
Rural, selected in North, Central and South Italy. We consider the altitude profiles of all the buildings to get what is called a *visibility graph*, i.e., a graph $G(\mathcal{N}, \mathcal{E})$in which $\mathcal{N}$ is the set of all the buildings of the area, and $\mathcal{E}$ is the set of all the edges that can be effectively realized between every couple of buildings. The simple criteria we adopt in this work to decide if a link is feasible is the presence or absence of line-of-sight. More details on the process of creation of the data-set can be found in [9], [10], which we refer to as `TrueNets`.

We focus on robustness metrics of the graphs realized with `TrueNets` and we compare them with the properties of a full-mesh graph realized on the same set of nodes. The motivation comes from recent works that deal with mesh backhaul assuming a full mesh between the network nodes [1], [11] and we want to test how this assumption influences the results in a more generic setting.

Given a geographical area, we choose a building in the area which serves as a gateway n_g. This building is selected according to heuristic considerations that define a good place for a gateway, and we initialize $\mathcal{N} = \{n_g\}$. We select a $1400\,m \times 1400\,m$ area centered on the gateway and we divide it in a 5×5 regular grid of squares with side 280 m. In every square we choose the tallest building and we place there a node of the graph, completing a set $\mathcal{N}$ with 26 nodes. In some cases there are no buildings in some squares, or the tallest building is not in line-of-sight with any other building, so not all the graphs are made of exactly 26 nodes.

Given $\mathcal{N}$ we compute two different graphs: i) the visibility graph $G^v(\mathcal{N}, \mathcal{E})$, where $\mathcal{E}$ contains only the edges that are feasible according to `TrueNets`; ii) the full mesh graph G^f obtained connecting every pair of nodes in $\mathcal{N}$.

Fig. 1 reports the spanning tree rooted in the gateway computed with Dijkstra algorithm when the cost of links is proportional to their length for an area in Urban 3 scenario (around 'Piazza del Municipio'); the thin dotted lines define the regular grid and one of the squares is in the gulf of Napoli and has no building. The left hand side refers to G^f, and it is obviously a star. The right hand side refers to G^v and the



(a) FullMesh (b) `TrueNets`

Fig. 1: G^f vs. G^v: minimum spanning tree to reach n_g in a Urban 3 area with full visibility or `TrueNets` model.

outcome is clearly very different. The goal of the figure is qualitative: to show how different the topology is when it is built considering effective topographic data from the one built on a simple visibility model, heuristically intuitive given the nodes are on the tallest buildings of each area.

The goal of this initial work is to study how G^v influences the feasible network topology given a set of nodes, and specifically, how different robustness metrics of networks built on G^v and G^f are.

II. Robustness and Performance Metrics

A. Effective Graph Resistance

The first metric we consider is related to the overall potential capacity of the network and its robustness, because it takes into account the presence of parallel (possibly disjoints) paths. The metric is called *Effective Graph Resistance* ξ and was studied deeply in [12], where we refer the reader for details. The effective graph resistance is the average of the resistance between any two nodes s, d in the network, computed as if the graph were an electrical resistive circuit where the links are resistances. The resistance of the links can be unitary, or proportional to the link length; we use unitary resistances for the sake of simplicity, but the extension to a graph weighted with the link length is trivial. The authors of [12] show that the effective graph resistance can be computed as

$$\xi = n \sum_{i=2}^{i=|\mathcal{N}|} \frac{1}{\lambda_i} \tag{1}$$

where λ_i is the value of the *i*-th eigenvalue (ordered by their value) of the Laplacian matrix of the graph. The Laplacian matrix L of a graph is defined as the difference $\Delta - A$ of the degree matrix Δ and the adjacency matrix A. A weighted Laplacian matrix can be defined if edges are weighted. We use this metric to give a high level estimate of the quality of a mesh build on a specific visibility graph G^v. ξ strictly decreases when an edge is added to the network, so the smaller ξ the more robust is the mesh, but also the overall capacity can be larger, as it can exploit more disjoint paths between nodes. To guarantee an easy interpretation of results independently from the number of nodes of the graph, we use the ratio between the effective resistance of G^f and that of G^v: $\xi_R = \frac{\xi_{G^f}}{\xi_{G^v}}$. Since the effective resistance of a full mesh is the minimal possible one, $\xi_R \leq 1$ for any visibility graph G^v, and the smaller it is the worst are the properties of meshes built on G^v.

B. k-edge Connectivity

Next, we consider $r[k]$, the fraction of nodes that are part of a k-edge connected graph embedded on a graph G and including n_g. Algorithm 1 describes how we compute $r[k]$.

Data: $G(\mathcal{N}, \mathcal{E})$, n_g, k_conn_max
Result: Number of nodes in the k-edges-connected subgraphs
k = 1;
r = dict() // a mapping int–>object ;
for *i in 1 … k_conn_max* **do**
  r[i] = [] // initialize to an empty list
end
newG = G;
while *k <= k_conn_max* **do**
  $S(\mathcal{N}_s, \mathcal{E}_s)$ = compute_min_spanning_tree(newG);
  for *e in* $\mathcal{E}_s$ **do**
    newG.remove_edge(e);
  end
  // connected component of newG including n_g;
  newG = connected_component(newG, n_g);
  sizeG = |newG| // number of nodes in newG;
  if *sizeG > 1* **then**
    r[k].append(sizeG);
  end
  k += 1;
end
return r;

Algorithm 1: The spanning-tree robustness metric

The algorithm starts from complete graph G and creates a minimum spanning tree St, then removes from G all the edges of St and checks the size $sizeG$ of the largest connected component including n_g. It iterates this process as long as n_g does not get disconnected from the rest of the graph ($sizeG = 1$). At each k-th iteration we save $r[k] = sizeG$. A simple interpretation of $r[k]$ is that when a node n is k-edge-connected with the gateway, then there are at least k independent spanning trees that connect n to n_g. This means that n can survive any pattern of $k-1$ failures, as in the worst case scenario one edge fails on a different St, but still there is at least one St on which n can communicate with n_g. The $r[k]$ metric tells what is the number of nodes of the graph that are k-edge-connected. Note that the spanning tree are generated with the classical Kruskal algorithm, and not centered on n_g as in Fig. 1, otherwise only one spanning tree would be possible around the gateway in a full mesh, as all its outgoing edges are removed after the first loop of Algorithm 1.

Given a municipality, we select 5 gateways and areas A around them with different sets of nodes $S_A = \{\mathcal{N}_0 \ldots \mathcal{N}_4\}$.

Then we compute 5 graphs G_i^v as a visibility graph, and for each one we obtain $r_i[k]$.

C. Weather Disruption-Tolerant Networks

The third metric is a state of the art algorithm for topology design [11]. This work tackles the problem of designing a robust backhaul topology using a realistic model for the probability of link availability under heavy rain or snow. It proposes two optimization algorithms. The first one, which assumes uncorrelated link failures, is formulated as an integer linear program and it is more scalable and easier to reproduce, the second introduces correlation among link failures and is formulated as a quadratic linear program. We chose to use the first one, as it is more scalable and useful to highlight our findings. This model, called in the original work TD_IF, takes as input all the paths from any $n \in \mathcal{N}$ to n_g and selects the least number of disjoint paths so that the probability that n is disconnected is below a certain threshold ϵ. As shown in algorithm 2, TD_IF returns a set of optimal paths for every n, which are joined into a single topology. We introduce two modifications on the original algorithm to make it run on our graphs, first, we limit the set of paths to the ones with less hops than $diam(G)+2$, where $diam(G)$ is the diameter of the graph G; second we relax the robustness requirement otherwise some of the `TrueNets` graphs do not support a solution: reliability ϵ is achieved for at least 90% of the nodes in the graph. The metric is the cost of the graph, which is proportional to the sum of the length of the links in all the paths. This is, like ξ_R, a synthetic number whose absolute value has no straightforward interpretation, but that can be used to compare two strategies.

Data: $G(\mathcal{N}, \mathcal{E})$, n_g
Result: G_t
$\xi_g = \emptyset$ // empty set of edges ;
for n *in* $\mathcal{N}$ **do**
  P_{all} = calc_simple_paths(G, n, n_g, $diam(G)$ + 2);
  P_{best} = TD_IF(P_{all}, ϵ);
  for e *in* P_{best} **do**
    $\xi_g = \xi_g \cup e$;
  end
end
return $G_t(\mathcal{N}, \xi_g)$;

Algorithm 2: Modified TD_IF algorithm.

III. Results

We can now analyze the nine different municipalities in Italy, verify the impact of the `TrueNets` modeling and check if there are regularities for Urban, Suburban, and Rural areas.

Fig. 2 reports the ratio ξ_R between the effective resistance of G^f and G^v for all the municipalities under analysis. It is clear that G^v differ significantly from a full mesh in all cases, even in densely populated areas as the Urban municipalities in Italy. There is no specific “trend” related to the density of building, while what seems to have more impact is the landscape itself (Suburban 1 is mostly flat, while Rural 1 is in the mountains) and possibly the presence of buildings that are indeed much higher than others.

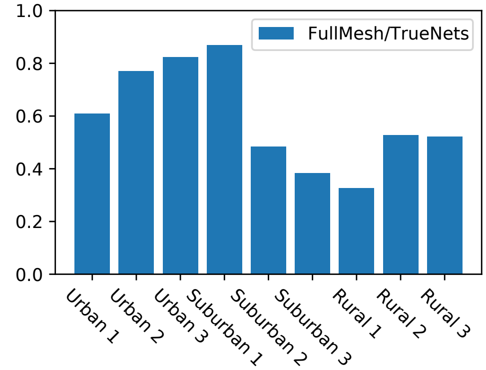


Fig. 2: The ratio ξ_R between the effective resistance of G^f and G^v.

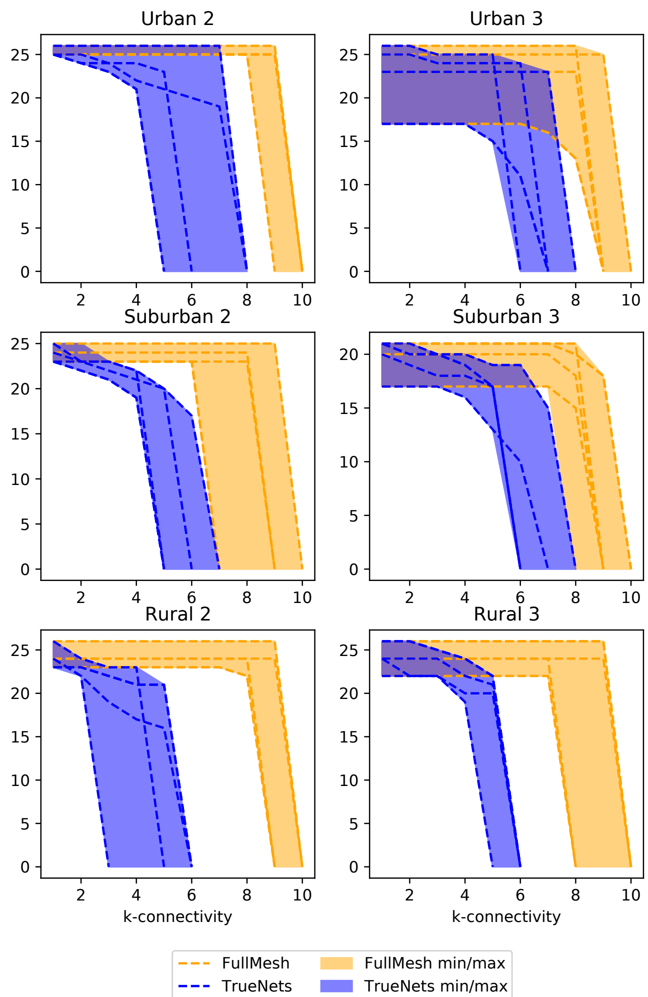


Fig. 3: The value of $r[k]$ computed on 5 instances of G^v and G^f in 6 out of 9 areas.

For space reasons Fig. 3 and 4 report the results for Central and South, and North and South municipalities respectively. The other results are similar and do not change the meaning

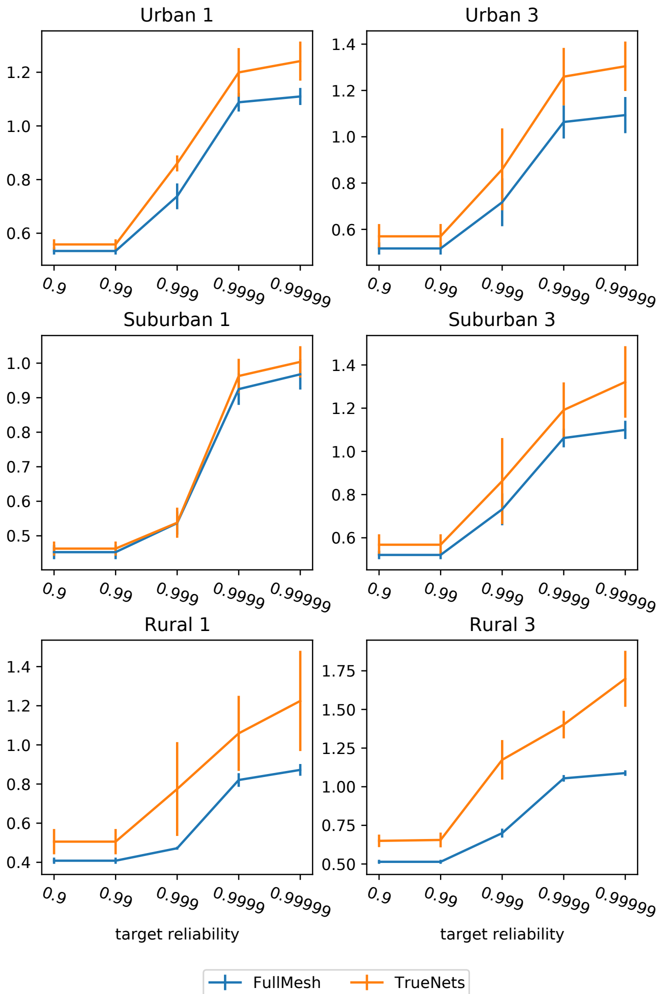


Fig. 4: Cost of the network produced by the TD_IF algorithm for on 5 instances of G^v and G^f in 6 out of 9 areas.

of the results.

Fig. 3 shows the value of the robustness $r[k]$ defined in Sect. II-B. Recall that $r[k] = |\mathcal{N}|$ means that there are k completely independent spanning trees that cover the entire network, while decreasing values means that some nodes are not k-edge-connected. It's clear that building a mesh on a graph that allows a large number of independent spanning trees makes it very robust and resilient, while if removing the links of the first spanning tree leaves some node disconnected the mesh is clearly fragile. Orange lines refer to FullMesh, while blue ones to `TrueNets` and we report results for the 5 different networks we generated in each area; shaded areas are the envelope of the five curves. The difference is evident with G^v that rarely allows the presence of more than two independent spanning trees without leaving some node disconnected; in some cases only one exists. Note that as explained in Sect. I not all the graph we generate have 26 nodes.

Fig. 4 finally describes the cost of the topologies generated with the TD_IF algorithm, again on 5 graphs per strategy. The x axis is the probability ϵ of letting a node disconnected, the y axis reports the cost of the corresponding graph generated using the TD_IF algorithm. Again due to lack of space we report only 6 out of 9 municipalities. In one area (Suburban 1) `TrueNets` generates topologies that are close to a FullMesh, and thus, the cost of the network is similar. This is consistent with Fig. 2 in which Suburban 1 has the closest score to the FullMesh. In all the other areas the cost is strongly affected by the underlying topology, especially with higher levels of robustness, with an more visible effect in rural areas.

IV. Future Work

This preliminary work shows how important it is to include in topological models of mesh networks details deriving from topographic characteristics. Once this need is assessed, a plethora of possibilities (and needs) open up for the research community: finding synthetic models that include these characteristics so that realistic studies can be carried out without the need of using specific or particular data; define strategies to select the best positions for mesh nodes given an area; study the impact of the mesh topologies on vertical applications or paradigms like Mobile Edge Computing (MEC) and many others.

References

[1] F. Yaghoubi, M. Furdek, A. Rostami, P. Öhlén, and L. Wosinska, "Consistency-Aware Weather Disruption-Tolerant Routing in SDN-Based Wireless Mesh Networks," *IEEE Transactions on Network and Service Management*, vol. 15, no. 2, pp. 582–595, Jun. 2018.

[2] O. Taghizadeh, P. Sirvi, S. Narasimha, J. A. L. Calvo, and R. Mathar, "Environment-Aware Minimum-Cost Wireless Backhaul Network Planning With Full-Duplex Links," *IEEE Systems Journal*, vol. 13, no. 3, pp. 2582–2593, Sep 2019.

[3] Y. Li, A. Cai, G. Qiao, L. Shi, S. K. Bose, and G. Shen, "Multi-Objective Topology Planning for Microwave-Based Wireless Backhaul Networks," *IEEE Access*, vol. 4, pp. 5742–5754, Jun. 2016.

[4] M. Polese, M. Giordani, T. Zugno, A. Roy, S. Goyal, D. Castor, and M. Zorzi, "Integrated Access and Backhaul in 5G mmWave Networks: Potential and Challenges," *IEEE Communications Magazine*, vol. 58, no. 3, 2020.

[5] B. Zhai, M. Yu, A. Tang, and X. Wang, "Mesh Architecture for Efficient Integrated Access and Backhaul Networking," in *IEEE Wireless Communications and Networking Conference (WCNC)*, 2020.

[6] M. Simsek, O. Orhan, M. Nassar, O. Elibol, and H. Nikopour, "IAB Topology Design: A Graph Embedding and Deep Reinforcement Learning Approach," *IEEE Communications Letters*, vol. 25, pp. 489–493, Feb. 2021.

[7] L. Maccari and R. Lo Cigno, "A week in the life of three large Wireless Community Networks," *Ad Hoc Networks, Elsevier*, vol. 24, Part B, p. 175–190, Jan. 2015.

[8] Y.-F. Chen and A. Arora, "Middle-mile network optimization in rural wireless meshes," in *IEEE 21st Int. Sym. on "A World of Wireless, Mobile and Multimedia Networks" (WoWMoM)*, aug 2020, pp. 224–233.

[9] L. Maccari, G. Gemmi, R. Lo Cigno, M. Karaliopoulos, and L. Navarro, "Towards Scalable Community Networks Topologies," *Ad Hoc Networks*, vol. 94, pp. 1570–8705, Nov. 2019.

[10] G. Gemmi, R. Lo Cigno, and L. Maccari, "`TrueNets`: a Topology Generator for Realistic Network Analysis," in *IFIP Networking*, Paris, France, June 22-25 2020.

[11] F. Yaghoubi, M. Furdek, A. Rostami, P. Oehlen, and L. Wosinska, "Reliable Topology Design of Wireless Networks under Correlated Failures," in *IEEE Int. Conference on Communications (ICC)*, 2018.

[12] W. Ellens, F. Spieksma, V. P., A. Jamakovic, and R. Kooij, "Effective graph resistance," *Linear Algebra and its Applications*, vol. 435, no. 10, pp. 2491–2506, Nov 2011.