Attack vulnerability of Lisbon public transports

Group 46: João Tiago Aparício 97155 Daniel Castro 87644 Miguel Trinca 86490

Abstract—**In this study, we modelled and examined the Lisbon public transports as a multiplex network. This allows us to integrates different modalities and understand its characteristics and resilience performance. To accomplish this, we evaluate how the network decreases performance by removing nodes and edges. Comparing different attack strategies, we concluded that for each goal and type of removal (edge vs node), we should use slightly different strategies. Those who involve recalculating metrics are usually the most effective. However, we identified a specific context where, counter-intuitively, that is not the case. Proposing a novel normalized version of AUC, we were able to compare side by side the robustness of each modality layer, regardless of their size, as well as the whole network. Testing targeting strategies, we also observed that resilience tests needed to remove about half the nodes of the network to leave all the remaining nodes completely disconnected and node removals are more effective than removing edges in this topology. Lastly, we simulate cascading events such as the breakdown of an entire transportation line.**

Multiplex networks Network resilience Transportation systems.

I. Introduction

The complex network of Lisbon public transports has failures that affect the daily commute of its users. A way to guarantee and improve the robustness of this critical component of our public services is crucial. To tackle this problem, we aim to understand how resilient is the Lisbon public transport network. Like any other public transportation network, this one is also complex. It is essential to look at how different stations/stops may have similar properties, bridging communities, for instance. We ask ourselves, *What is the role of degree-degree correlations?* to better analyze this type of behaviour. Also, looking at *how do different attack strategies affect the connectedness of the network* will shed some light on which strategy is the best on dividing the largest strongly connected component.

Making stops along our path to our destination is inevitable, and, the more stops we do, the more uncomfortable we become. So naturally, *how do nodes and edges targeting, affect the average path length* is a question that we are interested in answering. Besides the effects on the APL, understanding *how many nodes do we have to delete to fragment the network into isolated components* it is a must to comprehend the impact of construction or disaster on a station/stop in creating isolated components. In addition, we also aim to infer the impact of removing **edges vs nodes**, if metrics should be **recalculated** after each attack and what is the impact of **cascading failures**.

II. Methods

A. Data Characterization

The network we are going to analyze is based on the information available about the Lisbon public transport network. We have several files describing the location of each station and another set of files describing the shapes of each line, i.e., what is the station following each station in each direction and its geographical location. An instance of the data available for the Carris modality is the following:

```
  shapes.txt
149554,38.742155,-9.102203,1, ...
  stops.txt
1_7601,,AEROPORTO,,38.767825,-9.128712,,,, ...
```

B. Data Extraction and pre-processing

To transform this data into a network we created a new file with the shapes and the stops. This describes each line, and the stops connected to one another via their order, within the line.

```
 shapeWithStop.txt
164|38.72011|-9.15486|1|RATO| ...
```

Given the structure of the data, we created a digraph for each transportation modality. We resorted to digraphs for each layer because transports do not always flow in both directions within the same path. This allows us to create each layer of our multilayer network. We apply a multilayer representation because the edges of different layers have different types that represented different realities. Modelling such characteristics was not possible with a single layer (or monolayer) network.

Now that we have each layer, we also have to represent the possible multimodality interactions, i.e., the possibility to change different means of transportation within a trip. These links are of extreme importance because they allow us to assess the connectivity of the transportation system as a whole. To accurately understand where these edges could be located, we created a script that extracted the Lisbon city map and calculated the walking distance between every two stations and combined it with a standard coordinate distance calculation to get faster calculations. After getting this result, we selected the pairs of stations that had less than a 50-meter walk from one another and connected them with an edge that represented an intermodal change. The distribution of the distances selected, which account for about 0,0040% of the distances calculated, exhibit an exciting feature as can be seen in Figure 1.

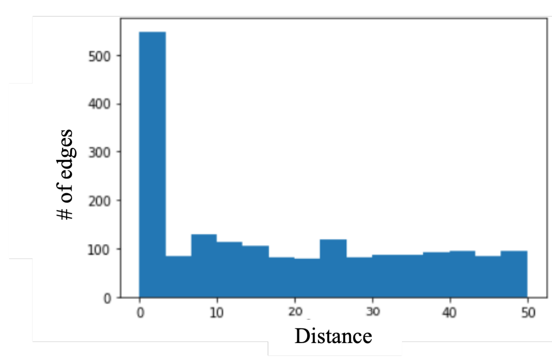

Fig. 1: The distribution of distances from stations selected (about 1878) have a very high cardinality of distances close to 0 meters. This represents the stations/stops that simultaneously belong to both layers, sharing a deeper connection than sheer multimodality.

We connected the eight modes of transportation in the Lisbon city using the described method. By programming a 3-dimensional method, for visualizing such network, using the geographical information available, we were able to see its structure, see Figure 2 for more detail. This network is composed of 8 layers, 7972 nodes and 11892 edges. The full network required the computation of 46399492 distances to join all the layers with the multimodal connections (requiring about 9.3 hours of computation). The set of layers has the following types: Carris, CP, Fertagus, Metro, Rodlisboa, Sulfertagus, Transtejo, TST.

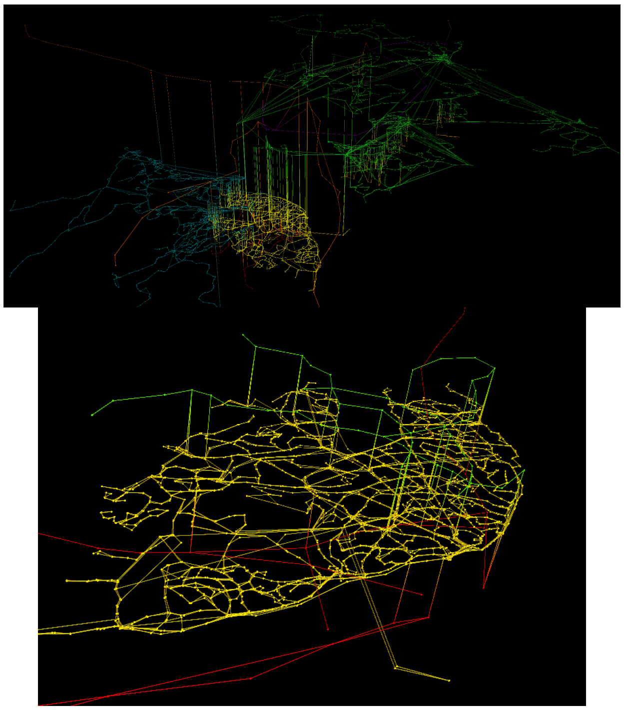

Fig. 2: Multilayer Lisbon transport network topology on a 3D representation. With all the layers (above) and a subsample of 3 layers (bellow) to convey a better understanding of the multilayer structure of the network, Metro in green, Carris in yellow and CP in red

This network is characterized by a quadruple [1]:

$$M = (V_M, E_M, V, \mathbf{L})$$

Where:

- V is the set of nodes in the network.
- V_M is the set of the node-layer combinations. These are the pairs of station-transportation modality.
- E_M is the set of intralayer and interlayer edges. These are the different paths that can be taken from a geographic place to another using the same transport modality and the possible modality changes with walkable distance.
- L is the set of layers in the network. In this particular case, each layer has only one aspect, since edges of different types can connect the same geographic places. Hence the usage of a **Multiplex network**.

C. Attack Strategies

To answer the questions above, we use different strategies. There are many approaches to remove nodes from a network. The simplest one is injecting **random** failures in the network, i.e., randomly removing nodes from the network. It is expected that this simple strategy may not maximize the damage on the network, but it can be a reasonably realistic approach since the Lisbon public transport network usually has failures in random stations due to unpredictable and unexpected events.

Other more precise strategies were defined by Holme et al. [2], and we based our attack strategies on them. We also attacked nodes and edges to be able to compare, which is more efficient. These more precise attacks utilize attributes from the network, such as **degree** and **centrality**. To attack a network with more precision, an effective strategy should be to attack the most important node. This notion of importance can vary depending on the attribute to which we give attention.

We could select nodes from descending order of degree. This attack is called **Initial Degree removal - ID removal** - because we base the attack on only the initial calculation of the degrees and we do not update it after removing nodes.

Another critical attribute is the centrality of a node. There are a few types of centrality, but the betweenness centrality is crucial since it represents the nodes that have edges that unify communities/groups inside the network. These nodes are an ideal location to increase the damage of the attack since it disconnects the communities of the network. This attack is called **Initial Betweenness removal - IB removal** - like ID removal, it only calculates the initial betweenness centrality of each node at the beginning and does not update it.

Both of these attacks only calculate the attributes, degree distribution and betweenness centrality, at the beginning of the algorithm. When we remove nodes from the network, it changes the degree of the neighbours of the removed node, and it modifies the betweenness centrality of all the nodes. Two new strategies appear called **Recalculate Degree removal - RD removal** - and **Recalculate Betweenness removal - RB removal**. These strategies are identical to the ID and IB ones except they update/recalculate the degree and the betweenness centrality after each node/edge removal. The RD and RB algorithms are considerably more time and computation expensive for each interaction, but we wanted to study if they are more efficient at destroying the network.

ID and RD removal are **local strategies** because when a node is removed, it only changes the degree of the neighbours of that node. Thus, only these local nodes need to be updated. On the other hand, IB and RB removal are **global strategies** since by removing a node, we need to recalculate the betweenness centrality globally, i.e., of all nodes.

Every calculation of the betweenness centrality of the network costs around 5 minutes. To apply the RB removal strategy, we would need at least 15 days. Because of that, we decided only to recalculate the betweenness centrality when batches of 5% of the nodes were removed. Being *N the number of Nodes* and *E the number of Edges*, the ID algorithm is **O(N+E)**, the RD is **O(N(N+E))**, the IB is **O(NE)** (using a faster algorithm for betweenness centrality created by Brandes [3]) and the RB is **O(N^{2}E)** .

The Lisbon public transportation network is a multilayer. Thus we also wanted to test if removing nodes that connect layers of the graph is critical, i.e., if it destroys the network faster. We call these nodes **multimodal hubs** and this attack strategy is **Multimodal Hubs removal**. For example, two multimodal hubs are *Alameda*, that connects Metro and Carris, and *Gare do Oriente*, that connects Carris, Metro, and CP. It is worth noting that the Multimodal Hubs removal attack strategy only makes sense when evaluating the multilayer network. When evaluating each modality, this strategy is invalid because there are no multimodal hubs.

Removing nodes from this network may correspond to a stop being destroyed or being in constructions, while removing edges may correspond to cutting roads or accidents preventing any vehicle from passing through. **Edges** can also fail. All our node attack strategies can be applied to edges with a few modifications. For example, how can we get the degree of an edge? The edge degree depends on the nodes that are connected by it. We used the following four different methods to calculate edge degree:

A $k_e = k_v * k_w$

B $k_e = k_v + k_w$
C $k_e = min(k_v, k_w)$
D $k_e = max(k_v, k_w)$

Where k_e is the edge degree and k_v and k_w are the degree of the nodes that are connect by the edge e. Since our network is directed, we also assume that v is the source and w is the destination node of the edge. However, there are other ways to calculate the degree of and edge such as $k_e = k_v$ or $k_e = k_w$.

Following the study of Holme et al. [2], where they concluded that method A. $k_e = k_v * k_w$ was the best fit to the majority of the network types that they studied, we based the decision of using method A. Method A showed the best correlation of the four methods between the edge degree and the edge betweenness centrality. It is expected that the node with a higher degree should also have a higher betweenness centrality. To calculate the betweenness centrality of an edge, we also used the Brandes [3] O(NE) algorithm.

We can expect that the more specific attacks should be more harmful than the random failures. Concerning the nodes vs edges, we believe that there should be a small difference between these attacks, but this is hard to predict. Degree based strategies should split the network into many subgraphs of vertices with low degrees. Betweenness centrality strategies should create clusters that are highly connected since they tend to destroy the bridges that connect communities first.

We understand that not only single nodes fail but also more catastrophic events can happen in the network, bringing the notion of cascading effect. For the **cascading effects**, we described and implemented two attack strategies.

The cascading effect **(1) Line Failure** has the goal of crashing an entire transportation line in the network. In our case, it could happen with a massive underground implosion, thus making the whole line stop and creating a Line Failure. We simulate this by removing all the nodes that have the attribute of that line. Some nodes may belong to more than one line, even from different modalities. For example, *Gare do Oriente* stop is a multimodal hub that belongs to Metro, Carris and CP.

The cascading effect **(2) Neighbors Failure** simulates the crashing of several layers of the neighbours of a failed node. One example of this in the Lisbon public transportation network could be a recursive failing of transports in the same area. When a station fails, the neighbour stations experience overflows and potentially fail as well. To simulate this type of cascading effects, we collect several layers of the node's neighbours, and then, remove all these nodes at the same time.

In our code, it is possible to check the algorithms we implemented to be able to utilize these strategies since they were not previously implemented nor available. In conclusion, we used the six following attack strategies for nodes and edges:

1) Random removal
2) Initial Degree removal (ID)
3) Initial Betweenness removal (IB)
4) Recalculate Degree removal (RD)
5) Recalculate Betweenness removal (RB)
6) Multimodal Hubs removal

We also used the 2 following attack strategies for cascading effects:

1) Line Failure
2) Neighbors Failure

D. *Evaluation of the attack strategies*

To assess the impact of each strategy, we will graph the evolution of different metrics for each removal. To understand how connectedness is affected we will measure the average path length (also known as average geodesic distance), average degree, number of isolated components, and the size of the giant strongly connected component. To understand how the average path length progresses, we only average out the shortest paths that still exist in the network. This approach allows us not to need to calculate the inverse path length, l^{-1}. As the study of this network is applied to a multimodal transportation network, we also measure the impact of the described attacks on the use of multimodality.

As standard multimodality usage metrics need the dynamic usage of the stations, we are unable to measure them. So, we propose a metric, the number of average modality changes within (every) 2 points for the shortest path, this tells us how many times we have to change transportation modalities for an average trip. Given that the focus of this work is to assess resilience, tests with this metric encompass future work.

III. Results & Discussion

A. *Network Analysis*

The distribution of nodes and edges respectively per each layer is the following: Nodes: 'carris': 2144, 'cp': 58, 'fertagus': 13, 'metro': 49, 'rodlisboa': 2210, 'sulfertagus': 110, 'transtejo': 8, 'tst': 3372 Edges: 'carris': 2894, 'cp': 125, 'fertagus': 25, 'metro': 105, 'multimodal': 1502, 'rodlisboa': 2752, 'sulfertagus': 187, 'transtejo': 12, 'tst': 4281. So, it is clear that the distribution of the station is not equitable in terms of layers. The average in and out-degree are approximately the same at 1.4917, which means that the majority of the edges are reciprocally directed.

As we are studying a is a directed network, its essential to assess the strongly connected components (SCC's). These are graph partitions, where the nodes are connected through a path. It is quite normal to have many single-node SCC's in unidirectional lines. This is precisely the case in this network. We observe 417 SCC's in the whole network. As some layers such as Metro and CP have only bidirectional relationships between nodes, what could contribute to a higher number of SCC's would be for instance an isolated CP line that is not connected to the rest of the network (which is not the case). However, this happens in the TST Sesimbra line and the Rodlisboa Vila nova line. The remaining small SCC's in the multilayer network are single stations where the flow is unidirectional. The giant strongly connected component has 7512 nodes which are about 94% of the total nodes of the network. The number of SCC's per each layer are: Carris: 67, CP: 3, Fertagus: 1, Metro: 1, Rodlisboa: 263, Soflusa: 1, Sulfertagus: 6, Transtejo: 3, TST: 177. The fact that the number of SCC's of the layers summed is higher than the SCC's in the multilayer network means that the multimodal edges are well placed on improving network connectivity.

1) Degree distribution: In this multilayered network, we can examine the kind of degree distribution we have, shown in Figure 3. We can also see some nodes with an out-degree of 0. These are start and end stations respectively that have no reciprocal edges on the opposite direction. The in-degree is always very close to the out-degree, with the information we have from the context, this tells us that the majority of the connections between stations flow both ways, forming what is known as a *chain*, even though there are apparent exceptions as we can see on the plot with a log scale.

2) Shortest path length: For the giant mutually connected component, the average path length (APL) is about 34.6878. This means that on average to get from a station/stop to any other on the network, we have to go through about 35 stations/stops before reaching the destination (this number includes the multimodal travels as well). Per layer, this value is usually smaller (Metro: 7.7176, CP: 10.2072, Fertagus: 5.0, Carris: 25.8734, Rodlisboa: 36.9525, Sulfertagus: 34.7543, Transtejo: 1.5, TST: 43.3558). However, each layer covers a smaller area than the composition of all the layers. In the case of TST, the APL is higher than in the composition of all the layers. This means that multimodality can be useful to avoid many stops.

3) Betweenness centrality: In the case of transportation networks, it is interesting to measure the betweenness centrality to understand what are the nodes that connect different communities of stations, i.e. sets of interconnected stations within a region.

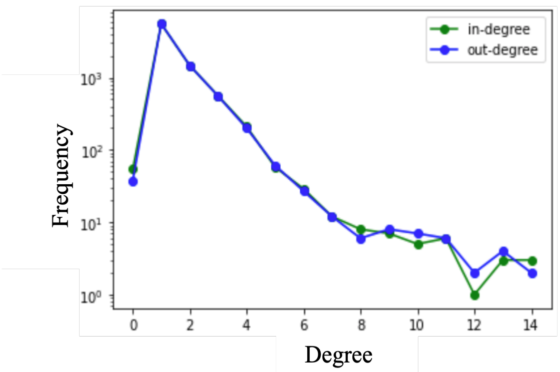

Fig. 3: Degree distribution of the network: Like all of the nodes belong to a specific line, we can see a high tendency of having both the in-degree and out-degree equal to one. There are also some intersections of lines, that is why we see a fairly high prominence of in and out-degree equal to two.

In the multilayer network, we identify some stations that have a very high centrality (See Fig. 4), these are mostly from TST, this may be a sign that TST is kind of a bridging layer in some zone. Some examples of such bridging stations are Lisboa Gare do Oriente (0,2529), Setúbal Ciprestes (0,1639), Lisboa Alcântra (0,1592) at TST. Metro also has three stations with exceptionally high betweenness centrality; these are Campo Grande (0,1813), Oriente (0,1504) and Cidade Unversitária (0,1480). Rod Lisboa also has a station with a very high betweenness centrality, Lisboa Campo Grande (0,1879). With this simple analysis, we can see that both Lisboa Campo Grande and Lisboa Oriente are multimodality hubs that bridge across different layers.

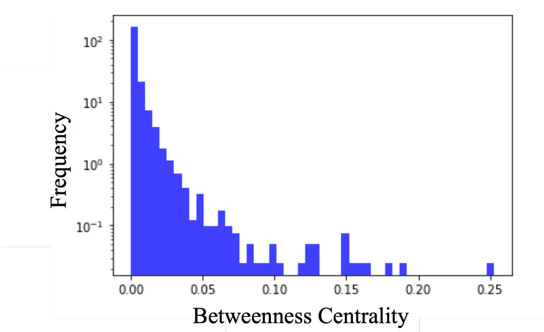

Fig. 4: Betweenness centrality distribution: In the graph we can clearly see a few nodes with very high centrality, these are the nodes that represent the stations mentioned above.

4) What is the role of degree-degree correlations?: To understand the role of degree-degree correlations, we look at degree assortativity. This measures the similarity of connected nodes concerning their degree. Arruda et al. [4] noted that in multilayer networks, degree-degree correlations should be measured system-wide. In that same study, he generalizes this concept for these types of networks and applies it to an airport transportation multilayer network. There he notes that the rich-club effect is, in fact, present in such networks, masked due to the high number of peripheral nodes that connect the hubs. However, intralayer, the networks tend to be disassortative as they focus on one specific region.

Studying this network, we expect to see similar behaviour given some geographical similarities of the reality being represented. We calculated the assortativity for the multilayer network and for each layer, and we observed the same results. In Figure 5, we see the same pattern described by Arruda et. al.. This is reasonably simple to understand since the assortativity is influenced by the high number of multimodal hubs that connect to one another. Analogous to the properties found in past case studies, we may find a kind of a rich-club effect, that may be harder to detect due to many peripheral nodes. Since this is not the focus of this work, it will be left for future studies.

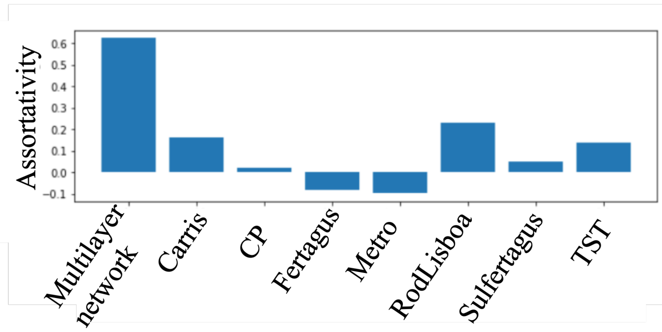

Fig. 5: Assortativity distribution of each layer and the multilayer network. We can see a much higher assortativity in the multilayer network than in any of the single layers.

5) How do different attack strategies affect the connectedness of the network?: In this section, we attempt to answer this question by analyzing the behaviour of the size of the largest strongly connected component - SCC - over time. By implementing the different attack strategies discussed in the previous subsections, we can accurately depict this behaviour. For each iteration, we compute the largest SCC and its size. This size will decrease over time, ergo the strategy which has a faster decrease has a more significant impact on the network. Also, networks that exhibit a higher decrease sooner, as the percentage of iteration steps, are less robust. In the same fashion, a high-performance attack strategy is one which the SSC descends faster.

To compare different strategies, we use the discrete Area Under Curve (AUC) and a normalized AUC to compare the resilience of the different layers and the multilayer network. The normalized $AUC \subset [0, 100]$ allows us to compare the values of networks with different sizes, and is calculated using the following formula:

$$AUC_{Normalized} = \sum_{i=0}^{\chi} \frac{\tau_i}{V} \cdot \frac{100}{\chi}$$

Where χ is the number of steps of the simulation, τ_i is the size of the largest strongly connected component at timestep i, and V is the number of nodes of the network. This measurement allows us to compare the different resilience side by side. It is important to note that this metric might have a higher variability (for both inflation and deflation) in smaller networks given the granularity. Observing table I, the most resilient layer is Fertagus, and the least resilient layer is RodLisboa.

Looking at the table I, we can see that the RD and RB strategies usually yield the best results across all layers. Nevertheless, the RB strategy tends to have a faster descent among the different layers as we can see in Figure 6. Moreover, the IB and Random strategies seem to have the least impact on the size of the largest SCC. There seems to be no particular reason why ID strategy has a better result than the RD. However, this is the case in SulFertagus, and it should be investigated further. We postulate that this happens because there may be fairly large components that have nodes with a high degree; however, removing nodes from these components does not affect the size of the largest component. So, this phenomena probably has more to do with the metric we are using than the strategy itself.

	Strategy				
Network	Random	ID	IB	RD	RB
Carris	9.8306	4.9715	8.1610	4.4104	**1.2501**
CP	10.7915	17.3442	14.5977	9.4074	**6.0337**
Fertagus	27.5555	30.2222	27.9999	24.8888	**22.2222**
Metro	17.9930	18.3006	16.9550	11.8800	**10.2652**
RodLisboa	7.0540	2.7484	6.7469	1.7580	**0.6569**
SulFertagus	2.4251	1.6707	12.4929	2.9098	**0.8832**
Transtejo	19.0	20.0	20.0	**15.0**	20.0
TST	5.7429	2.5852	7.3845	2.1839	**0.6288**
Multilayer	11.4000	6.6333	10.7513	**5.9189**	8.7007

TABLE I: Normalized AUC across all networks: RB is the most effective node removal strategy in all the networks, with exception of Transtejo and the multilayer network.

6) How do node and edge targeting affect the average path length?: To understand the evolution of path length when targeting nodes and edges, we calculated the APL only for existing paths along with the network. So, we expect the APL to reduce along each time step quickly. We ran the result for each layer and the whole network as well.

As we can see, for Carris on Figure 7, the RD still is the best strategy. The remaining strategies have about the same efficiency, with RB being slightly ahead towards the end of the simulation. This result is not unanimous, as we can see in Figure 8. In the multimodal network with all the layers, see Figure 9, we see that RB is a strategy that promotes the fast decrease the APL. This indicates a lower network resilience.

7) How many nodes or edges do we have to delete to fragment the network into isolated components?: An **isolated component** is when a node loses all its edges. Since our graph is directed, when we talk about all the edges, we are mentioning both the in and out edges. To be able to answer this question, we used the attack strategies proposed on the methods and evaluate the evolution of the network by showing the distribution of the isolated components. To accurately understand the attack strategies, we ran them across every layer in isolation and then in the complete network, except for multimodal hub removal on isolated layers because there are no multimodal hubs in isolated layers. Figure 10 shows the evolution of isolated components for each strategy in the Carris layer. We obtained about the same results in every layer and for multilayer network is also very similar.

In the graph, we notice that all attack strategies, except RD, increase the number of isolated components and then decrease. This means that after we reach the maximum, we are only removing isolated nodes. This means that the lower the maximum and the later its reached, the worst the strategy is. Intuitively, IC's continuously growing in RD makes sense since we want to increase the number of IC's, and IC's are nodes without edges when taking the degree hubs, it is normal to increase the number of IC's. Halfway through RB, we start to see rapid growth. This is because after removing the main bridges from communities, there is still are redundancy paths. Once these paths start to be destroyed, it is simpler to destroy each community individually. It should be noted that it is likely that a node with a high degree also has a high betweenness centrality since multimodal hubs are often points that connect communities. The IB attack is the worst because we are dividing the problem infinitely. Similarly, it is like wanting to go from A to B, then divide the path into 2, then divide into 2, recursively.

So, to answer the question in this section, we look at the number of iterations when RD ends the simulation. This is the number of nodes that have been removed when all the remaining nodes are isolated. The higher the percentage of nodes that need to be removed from the total of nodes in this network, the more resilient the network is, see Figure 11.

To understand the multimodal hub removal, we recurred to another graph due to scale and purpose. As the graph reaches a pique and

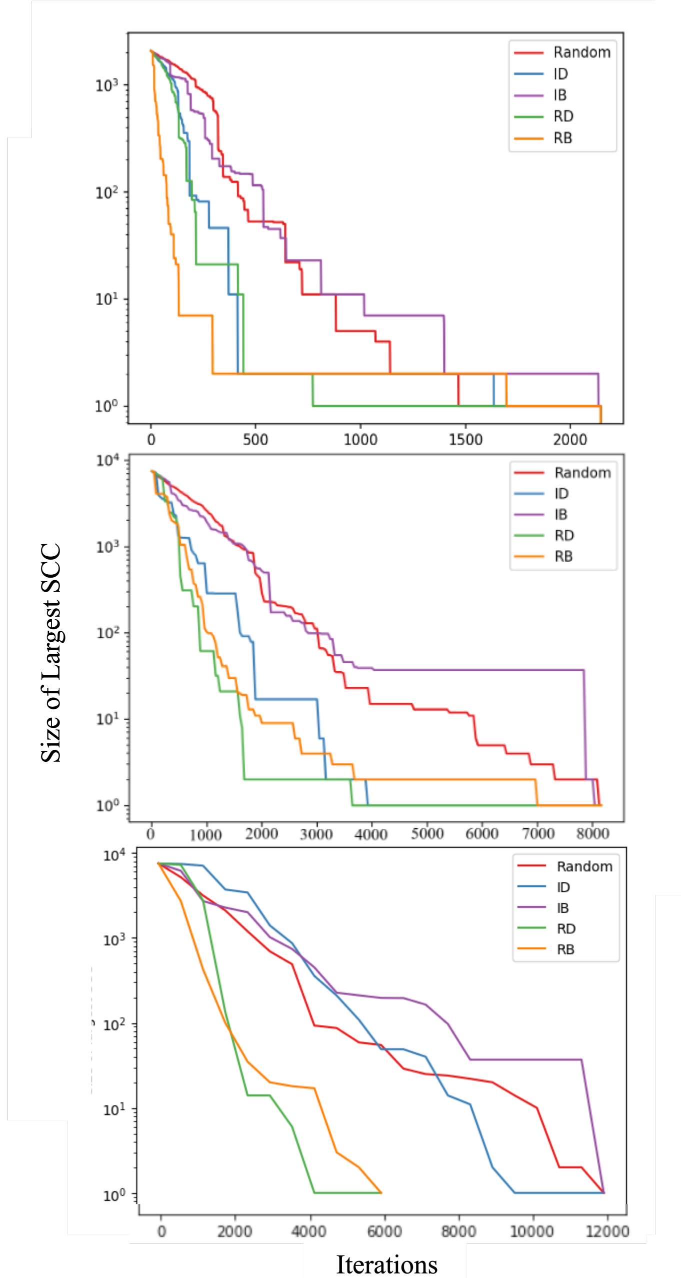

Fig. 6: Evolution of SCC size along node attack strategies for a single layer, Carris (above) all layers (middle) and edge removal in multilayer network (bellow). We observe the RD strategy yields a faster decrease, this checks out with the Normalized AUC with a lowest value. In the 3rd graph we observe that we still need many more edge removals to get the same result of size of largest SCC, as in node strategies.

then decreases, we can see that we do not need to remove all the multimodal nodes to get isolated layers. As we can see, there is a slight variation when some nodes are removed. This means that removing specific nodes has more impact on the interlayer connections, see Figure 12. This is not directly comparable with other strategies, as it only encompasses the removal of about 10% of the nodes. However, from what we can assess, in the beginning, the growth in the number of isolated components follows approximately a linear function, which is slower than RD.

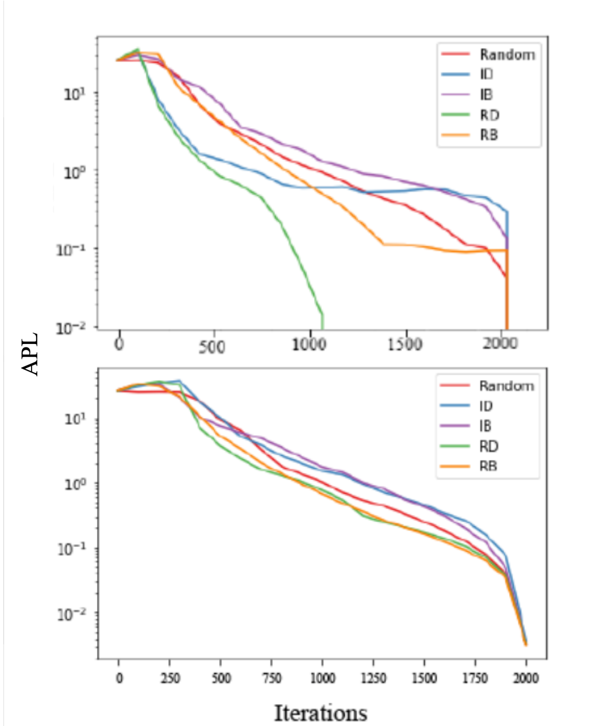

Fig. 7: Evolution of APL along node (above) and edge (bellow) attack strategies for Carris. We can see that the APL decreases much faster in the RD node strategy. Interestingly, we can also see a slight increase in the beginning of the simulation, this is due to redundancy of paths. In terms of node vs edge attacks, the performance of all strategies except ID and RB yield similar results.

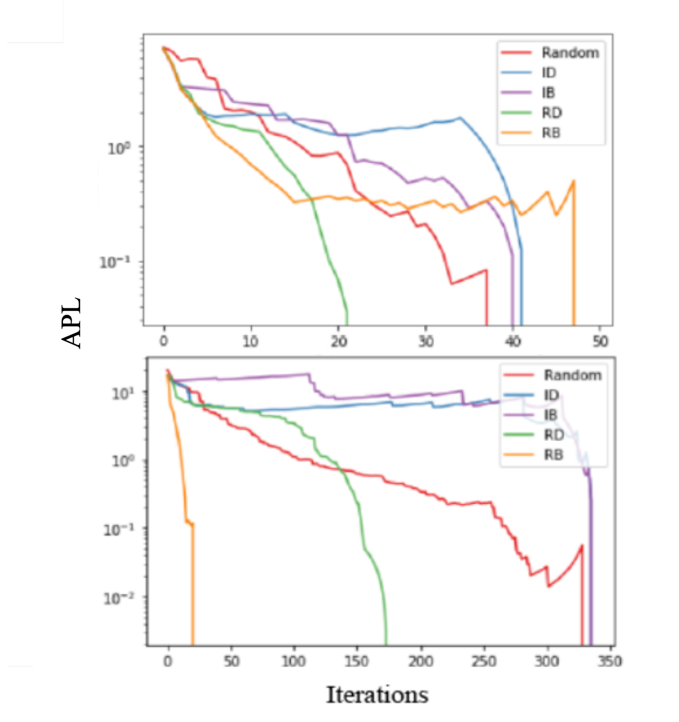

Fig. 8: Evolution of APL along node attack strategies for Metro (above) and SulFertagus (bellow). As we see, RB can either be the best or the worst strategy, depending on the network. This can be due to nodes with high betweenness centrality being part of paths that do not have alternatives. RB providing an efficient strategy indicates a lower network resilience.

On the other hand, there are edge attacks. The reasonably apparent result we are expecting, in this case, is that we have to remove all the edges to have all nodes be isolated, and that is exactly what we get, see Figure 13. The behaviour of attack per layer and in the multilayer network is identical as can be seen in both figure 13. Contrary to node targeting, the worst strategy is RD, and the best strategy is IB. We can conclude that the betweenness centrality is a better metric for edge targeting and degree is better for node targeting.

8) What is the impact of removing nodes vs edges?: We observed that nodes attacks are more efficient than attacking the edges, as we need more removals to get the same increase of IC's and decrease of APL and decrease of SCC. As the previous Figures show, it takes much more iterations to destroy a network by attacking the edges. This happens for various reasons. Trivially there are more edges than nodes, so more iterations are needed to create the same impact. Another important reason is that when we remove a node, we also are removing all the edges of that specific node while removing an edge, does not remove the nodes that are connected. This makes the node removal more efficient on harming the network.

On the SCC and the APL evaluation, we also see that edges attacks are very similar to each other and also less efficient at destroying the network than nodes removal. We can conclude that the network has various redundant paths. So removing edges, we can go to the same destination by going through more stations.

It is also important to note that in the IC evaluation, the ID and RD removal attacks on the edges are not as efficient as in the nodes and has a slow impact. With a slow impact, we mean that we need to remove more edges, to create the same damage as the node removal. This happens because, when we base the attack on the edge degree, we are attacking edges of hub nodes (nodes with high degree). This means that we are removing edges from nodes with lots of edges and so it does not show the effect at first.

9) Should we recalculate the degree and betweenness after each attack?: More often than not, the random strategy is better than the strategies which do not recalculate the new metrics of the new graph (IB, and ID). On the other hand, RB and RD removal strategies can be very effective, yet not so computationally efficient. They take much more computational power to compute metrics, especially the RB. As can be seen in previous sections, recalculating degree after each node removal has proven to be beneficial, but recalculating betweenness centrality has yielded relatively stable result for IC's and wildly variable results in APL evolution. Intuitively this could probably be explained by the lack of redundant paths in some networks, which is a fair way to assess its robustness. Both RB and RD strategies have the same similar results across layers and being the best ones to reduce the SSC.

For the edge removing strategies, in the case of the goal being generating more isolated components, recalculating the betweenness centrality yields worst results. These results are curious as they are quite different from the ones proposed by [2], "suggesting that the network structure changes as important vertices or edges are removed.".

Despite the RD and RB strategy overall having better results for nodes attacks, in massive graphs seems unfeasible to calculate the betweenness centrality per each iteration, maybe choosing a slightly worst strategy is not such an issue. Another possible strategy is, instead of calculating per each iteration, i.e., each node removal, we calculate per each percentage of nodes. This way, we are avoiding

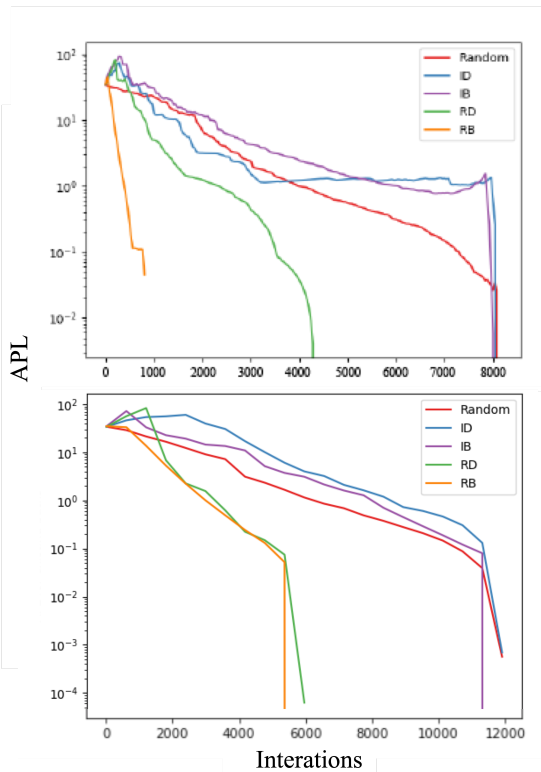

Fig. 9: Evolution of APL along node (above) and edge (bellow) attack strategies for the Multilayer network. We can see that the APL decreases much faster in the RB than in the RD. We can also see a slight increase in the beginning of the simulation (only for other attack strategies), this is due to redundancy of paths. The number of removals needed to get the same APL is much higher for edge strategies, meaning node removal is more effective for decreasing APL.

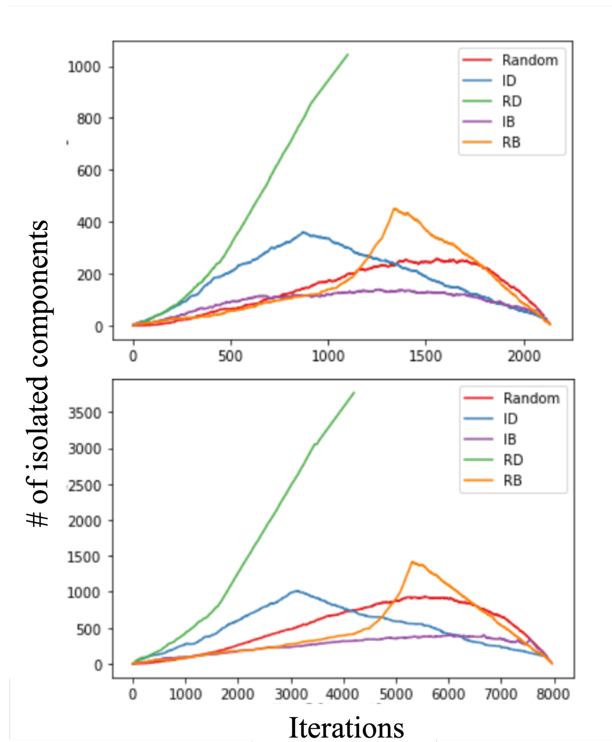

Fig. 10: Evolution of the Isolated components (IC) in the Carris layer (above) and Multi-modality network (bellow) for node removal. In this graphs we can clearly see the that RD had the best results, this is the only one that stopped before the end of the simulation, because there are only isolated components when it stops.

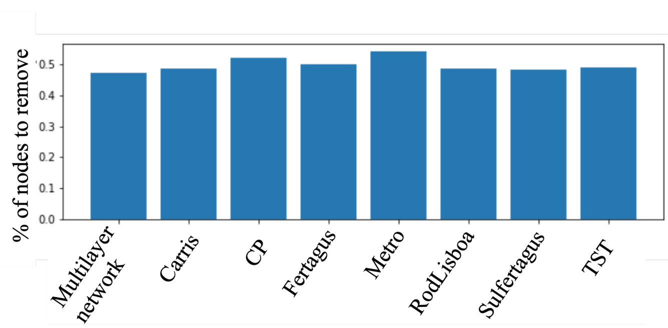

Fig. 11: Percentage of nodes that need to be removed to separate the network into only isolated components. The lower this value the less resilient is this network. We can see that the resilience across all networks is about the same.

the complexity, with the trade-off of disrupting a little of the result.

10) What is the impact of cascading failures?: The attack strategies that we implemented on cascading failures are the Line Failure, completely removing a transportation line, and removing the neighbours of a specific node. The cascading failures by itself, are a complex study, and we only saw the top of the iceberg. For the first attack, we simulated the crash of each line individually, i.e., we crashed one line, measure the number of Isolated Components on the network and then, after resetting the network to its initial state, repeated the process for each line. By doing this evaluation, we got the maximum number of 6 isolated components from one line. This line was from Rodlisboa modality and it connects *Pontinha* to *Campo Grande*. This is a known line that connects people from the suburbs of Lisbon to the centre of Lisbon.

Our second simulation used the neighbour's failure attack strategy. The simulation was similar to Line Failure. For each node, we calculated their neighbours, removed them and collect the metric of the isolated component. We decided to use three layers of neighbours of a node since it had the average number more closely to the average number of each line to be able to compare both. From this strategy, we found that Odivelas was the critical point having a maximum number of isolated components of 10. Again, we see that these nodes that connect the suburbs of Lisbon are crucial to the organization of the public transportation network of Lisbon. Nevertheless, this Neighbor Failure strategy presents better results than the Line Failure strategy, from almost double the damage on the network (6 vs 10), but this is from a small sample.

A future study could be done with this cascading events attack strategy. For example, fail more than one line at the time and also not crash lines randomly and use a specific metric to decide what lines to crash, analogous to what we have done with the degree and betweenness centrality. The second attack could be further explored by testing a different number of layers of neighbours, with more depth. The archetype of these could be joining these two methods, trying to fail lines that are near other lines by using the information of the neighbours of each node. This alone is an extensive study on the cascading effects area.

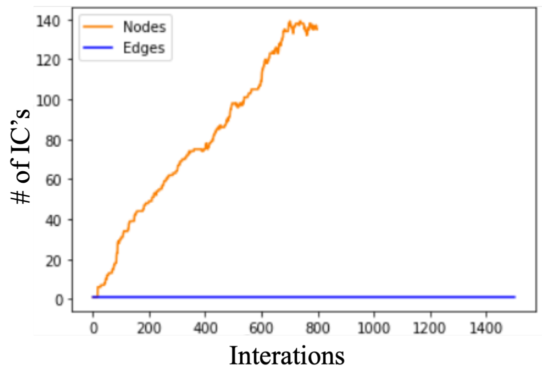

Fig. 12: Evolution of the Isolated components: IC in the Multimodal Hubs removal strategy, we can see that node removal is much better than edge removal, that makes sense since in edges we only separate in at most 8 IC's (single layers)

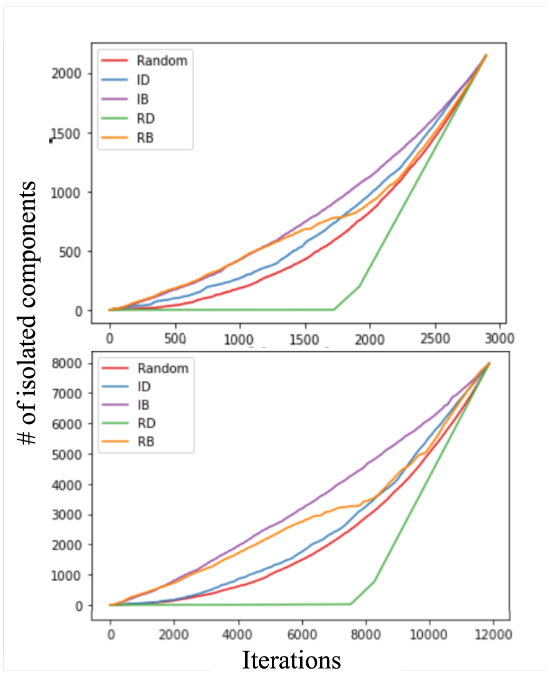

Fig. 13: Evolution of Isolated components along edge attack strategies for Carris (above) and for the multilayer network (bellow). The RD is always the worst strategy, this is the case because we are removing edges from nodes with high degree.

IV. Conclusions

In this work, we modelled and analyzed the structure of the Lisbon public transport network. We were able to understand the geographic information inherent to the multiplex network by programming a 3D plotting method. After calculating and interpreting some of the network metrics, we showed that multimodality can help diminish the number of stops in a path between two nodes. Then we identified the stations that work as bridges between different layers and communities. Then we analyzed different attack strategies based on previous work. This was an important step to guide the assessment, along with the evaluation metrics we chose. These include the size of the largest SCC and a proposed normalized AUC, APL and IC's. These allowed us to conclude that for each goal and type of removal (edge vs node), we should use slightly different strategies. We were able to empirically show that to get the same results, in this network topology, we needed many more edge removals to get the same result as some node removals.

The presented graphs also demonstrated that the strategies that depend on recalculating metrics are usually more effective, with the exception of a particular case of edge removal using betweenness centrality to maximize IC's, that counter-intuitively had better results for the IB strategy. Even though we were able to postulate on this phenomena, further research needs to be done to understand it. We also showed that the resilience tests needed to remove about half the nodes of the network to leave all the remaining nodes wholly disconnected. This is a phenomenon that happens in all layers and the multilayer network as well. We also verified the higher assortativity phenomena in multilayered networks, in contrast to single layers, and explained its plausibility.

Based on the resilience tests we were able to conclude that the most effective method for targeting nodes was RD, however, in some cases, RB yielded better results for multilayer APL decadence (in both nodes and edges strategies), although it showed higher variability. For decreasing the size of the larges SCC, RD yielded better results for the multilayer network, but for most of its individual layers, the best strategy was actually RB. With the normalized AUC, we were able to compare side by side the robustness of each network, regardless of their size. Carris, RodLisboa, SulFertagus ad TST were much less resilient than the other networks. The two most resilient networks were Fertagus and Transtejo. However, this result may be due to inflation on smaller sizes due to granularity. The multilayer network was more resilient than one-half of the layers but less than the other. We also performed a few cascading failures in our network to understand some of its impacts. However, since it is a very vast and exciting topic for itself, we only did a small experimented with two strategies, showing that neighbour failure is more effective than line failure in this particular network, even though they are moderately similar in the structure of the present network. Last but not least, we identified that further research is needed in the multimodality index metrics, rich club on peripheral nodes and cascading effects in the context of this network.

References

[1] M. Tomasini, "An introduction to multilayer networks," *BioComplex Laboratory, Florida Institute of Technology, Melbourne, USA*, pp. 1–14, 2015.

[2] P. Holme, B. J. Kim, C. N. Yoon, and S. K. Han, "Attack vulnerability of complex networks," *Physical review E*, vol. 65, no. 5, p. 056 109, 2002.

[3] U. Brandes, "A faster algorithm for betweenness centrality," *Journal of mathematical sociology*, vol. 25, no. 2, pp. 163–177, 2001.

[4] G. F. de Arruda, E. Cozzo, Y. Moreno, and F. A. Rodrigues, "On degree–degree correlations in multilayer networks," *Physica D: Nonlinear Phenomena*, vol. 323, pp. 5–11, 2016.