



This is a postprint version of the published document at:

Martín-Pérez, J., Cominardi, L. Bernardos, C. J. y Mourad, A. (2019). 5GEN: A tool to generate 5G infrastructure graphs. In *IEEE Conference on Standards for Communications and Networking*.

DOI: <https://doi.org/10.1109/CSCN.2019.8931334>

© 2019 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

# 5GEN: A tool to generate 5G infrastructure graphs

Jorge Martín-Pérez<sup>\*†</sup>, Luca Cominardi<sup>†</sup>, Carlos J. Bernardos<sup>†</sup>, Alain Mourad<sup>‡</sup>

<sup>†</sup>Universidad Carlos III de Madrid, Spain

<sup>‡</sup>InterDigital Europe, UK

**Abstract**—Ongoing research on 5G is looking on software platforms to evaluate new developments on 5G networks. Some 5G hardware is now starting to be available, but it is scarce and very limited, which makes validation and performance evaluation of 5G quite challenging. Simulation is the tool of choice for most of the cases, but this requires creating large descriptor files representing a 5G network. This brings forward the need for tools that facilitate the generation of 5G networks' topologies. In this paper we present 5GEN, a tool that automatically creates graphs representing 5G networks. With 5GEN, a researcher can just define the number of resources, and 5GEN will generate the nodes and edges that interconnect them across the infrastructure. The tool has been successfully used to test several 5G network scenarios within the EU 5G-CORAL project.

**Index Terms**—5G, MEC, Fog, Edge, Tool, Network infrastructure

## I. INTRODUCTION

The next generation of 3GPP mobile networks, commonly referred to as 5G, will enable new communication scenarios and a network performance never seen before. Key features of 5G are *flexibility* and *automation*, making possible to create new network services in a way much faster and powerful than with current networks, allowing a new type of customers (the so-called vertical industries) to get into the picture. Among the multiple ingredients that need to be properly integrated to create a flexible and fully automated 5G system, network virtualization is a clear pillar. Network function virtualization (NFV) breaks the traditional strong bound between functionality and capacity of networking equipment: nowadays, a router is a physical box (i.e., hardware providing certain capacities) performing routing and traffic forwarding (i.e., implementing a certain functionality). With NFV, the capacity is provided by commodity high performance servers which provide virtualized resources on top of which the networking functionality is implemented as a set of interconnected virtual network functions (VNFs).

A very interesting and timely research problem is how to map VNFs to available resources and interconnect them. This is known as VNF placement problem, and it is critical in order to fully exploit the advantages of 5G. Most existing work, such as [1] and [2] tackle the placement problem using optimization techniques, which are then validated on small scenarios consisting of a couple of servers, or on top of outdated internet topologies like [3] which do not represent complex 5G infrastructures. Initial 5G research focusing on

network virtualization aspects, such as [4], and even implementing experimental systems that incorporate placement algorithms (such as [5]), are limited in terms of validation due to the use of scenarios that do not properly mimic how a 5G network deployment will actually look like. Another example of this is [6], which evaluates algorithms using a topology representing current network interconnection between cities of Germany.

There is a new computing paradigm that has recently appeared thanks to the fast development of NFV: *fog computing*. It has emerged driven by the Internet of Things (IoT), due to the need of handling the data generated from the end-user devices. The term fog is referred to any networked computational resource in the continuum between things and cloud. A fog node may therefore be an infrastructure network node such as an eNodeB or gNodeB, an edge server, a customer premises equipment (CPE), or even a user equipment (UE) terminal node such as a laptop, a smartphone, or a computing unit on-board a vehicle, robot or drone. In fog computing, virtual functions are hosted on resources that are inherently heterogeneous, volatile and mobile. This means that resources might appear and disappear, and the connectivity characteristics between these resources may also change dynamically. These VNFs compose service function chains (SFCs) providing more complex network services. Both the 5G-CORAL [7] and 5G-TRANSFORMER [8] projects are addressing this type of scenarios. The problem of VNF placement becomes even more complicated in fog computing scenarios, as there are more constraints that need to be considered.

In this short paper we aim at mitigating some of the pitfalls of the validation of VNF placement mechanisms, even in the presence of fog computing scenarios. We have developed 5GEN, an open source R package that generates realistic 5G graphs for testing and research purposes. Section II explains the topology of the generated graphs, and code snippets on how to generate an example. Then, Section III shows how 5GEN is used with some orchestration experiments within the context of the 5G-CORAL project. Finally, Section IV concludes the paper with some conclusions and future steps in the 5GEN development.

## II. GENERATED GRAPHS

5GEN generates graphs based on the 5G network characterization of [9]. The topology contains access rings of 6 M1 switches, each of them with 6 attached Active Antenna Units (AAUs) that provide connectivity to fog devices, e.g., motherboards, surveillance cameras, or robots. Access rings send traffic towards the core through the aggregation ring,

\*Corresponding author: jmartinp@it.uc3m.es

Work was partially funded by EU H2020 5G-TRANSFORMER Project (grant no. 761536) and EU H2020 5G-CORAL Project (grant no. 761586).

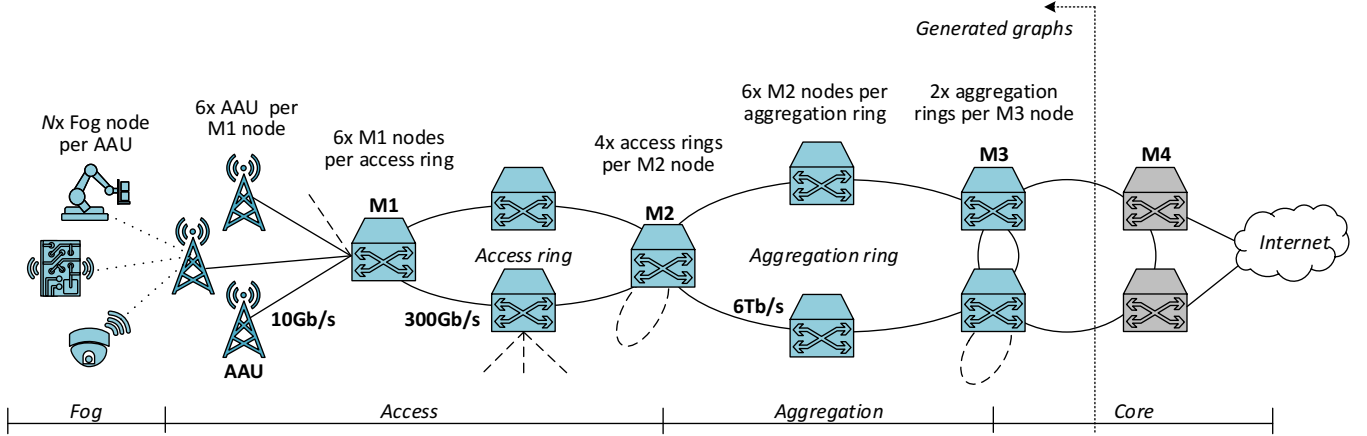


Fig. 1: Reference topologies generated by 5GEN

which has 6 M2 switches, each of them connected to 4 different access rings. Finally, 5GEN interconnects every pair of aggregation rings through two redundant M3 switches. Fig. 1 illustrates the 5G reference topology and the main entities present in the generated graphs.

5GEN adopts a clustering strategy to obtain the graphs. It creates clusters of 6 AAUs and connects each one to a M1 switch. Then, access and aggregation rings are created as clusters of M1 and M2 switches, respectively.

To generate a 5G infrastructure graph, the user invokes 5GEN with a set of AAUs' coordinates. Upon such input, the *build5GScenario* function (see Algorithm 1) executes a complete hierarchical clustering to build up a dendrogram<sup>1</sup> of AAUs. Then it cuts the dendrogram at a distance of 10km and creates groups of 6 AAUs with an M1 switch in the middle. This process is repeated to cluster M1 switches and create the access rings, and to cluster M2 switches connecting them in the aggregation rings. This is also done to group the aggregation rings and connect them to the M3 switches. Algorithm 1 shows the pseudo-code of the described procedure.

The next step is the attachment of resources to the generated infrastructure, by invoking *attachServers* and *attachFogNodes*. Our tool allows to specify where to attach resources in the graph (fog nodes are always attached to the nearest AAU), and how many CPU, memory and disk they have. Both *attachServers* and *attachFogNodes* collapse all the information in two R data frames that contain nodes and edges of the infrastructure graph. One data frame contains the switches, AAUs, servers and fog devices. The other contains edges representing fiber links among switches, links between them and the generated servers, and the wireless connectivity of AAUs and fog devices. 5GEN allows further customization, thanks to the *addNodeProps* and *addLinkProps* functions, which can be used to add or edit properties of the nodes and edges of the generated graphs.

To sum up, the generation of a infrastructure with 5GEN

**Data:** AAUs

**Result:** M1s, M2s, M3s

```

1 levels = [AAUs, access, aggregation];
2 distances = [10km, 20km, 40km];
3 foreach (l, d) in (levels, distance) do
4
5     N = {
6         AAUs,   if l = AAUs
7         M1,     if l = access
8         M2,     if l = aggregation
9     };
10    H = hclustering(N);
11    C = cut_dendrogram(H, d);
12    C = { {ci, ..., ci+5}i=0⌊C/6⌋ };
13    for c ∈ C do
14        c̄ = ∑ci ∈ c ci / |c|;
15        connect(c̄, {ci ∈ c}, l);
16        M1s ∪ {c},   if l = AAUs;
17        M2s ∪ {c},   if l = access;
18        M3s ∪ {c, c}, if l = aggregation;
19    end
20 end

```

Algorithm 1: build5GScenario

comprises the invocation of these functions<sup>2</sup>:

- 1) *build5GScenario*(AAUs);
- 2) *attachServers*(number, properties);
- 3) *attachFogNodes*(number, properties);
- 4) (*addNodeProps* or *addLinksProps*);

After that, the user generates a GML graph file calling *igraph::graph\_from\_data\_frame* [10] using 5GEN data frames of nodes and edges.

Fig. 2 shows a 5G graph generated invoking functions above over a set of AAUs generated with inhomogeneous Poisson Point Processes [11] in the industrial area of Cobo Calleja,

<sup>1</sup>*dendrogram*: tree diagram representing a hierarchical grouping of elements.

<sup>2</sup>For further details check code snippet examples at: <https://github.com/MartinPJorge/mec-generator/tree/f7e0aa3b7db2b24eb910e623e4ad2d8d9ada9718>

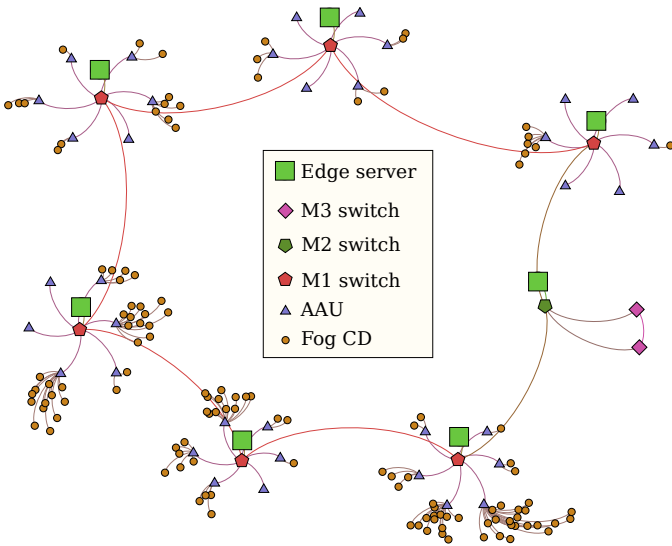


Fig. 2: 5G infrastructure graph with fog and MEC entities

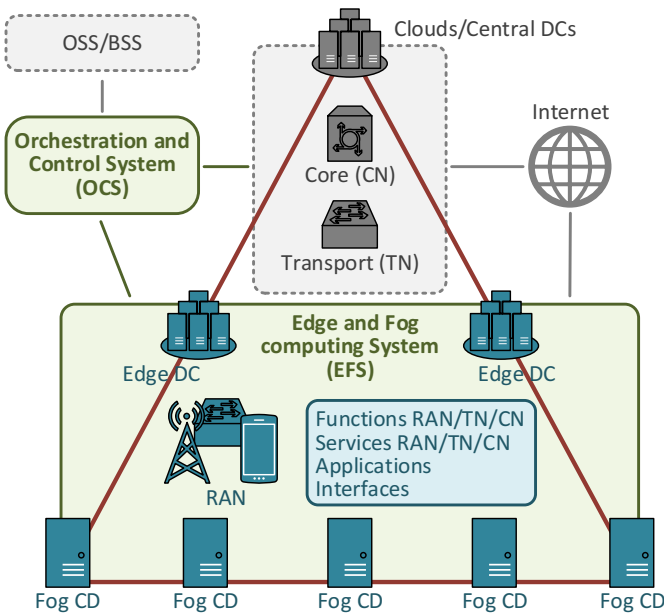


Fig. 3: 5G-CORAL hierarchical multi-tier computing infrastructure

Madrid. Such set of AAUs are included in the *cobo* dataset of 5GEN R package.

### III. EXAMPLE OF USE: A 5G-CORAL USE CASE

In this section we show an example of how 5GEN has been used to evaluate orchestration algorithms developed by the 5G-CORAL project.

Thanks to NFV, the functions belonging to different network domains (namely radio access, transport, and core) can be virtualized and instantiated at different locations. 5G-CORAL envisages the overall computing infrastructure as a hierarchical multi-tier system as illustrated in Fig. 3. Such architecture comprises clouds and central data centers (DCs) on top, edge data centers (Edge DCs) in the middle, and fog computing

devices (Fog CDs) that are available locally in the access area. Central DCs are large scale public/operator-owned data centers while edge DCs are small scale computing infrastructure deployed at the edge (e.g., fewer servers). Finally, Fog CDs comprise a variegated set of resources with limited computing capabilities like network nodes, end user devices, etc.

5G-CORAL deals with a large heterogeneity of computational resources in the project, from robots, sensors, raspberry Pis to edge servers and cloud servers. A realistic evaluation of orchestration mechanisms for 5G-CORAL should be conducted in scenarios conveying such heterogeneity. In 5G-CORAL, two greedy VNF placement algorithms have been developed:

- *CAPEX-g*: which greedily looks for the cheapest links and computational nodes to steer and deploy the Virtual Links' (VLs) traffic, and VNFs;
- *LongRun-g*: which deploys VLs and VNFs in those links and computational nodes that minimize the volatility of the deployed Network Service (NS);

We define volatility  $v \in [0, 1]$  as the probability of a 5G network component to stop working during the NS lifetime, i.e., the time the NS is running.

In the testing phase of 5G-CORAL, both algorithms perform the deployment of a NS composed of AWS instances: x1 t3.medium, x2 t3.micro and x2 t3.nano<sup>3</sup>, using an infrastructure generated by 5GEN. Such infrastructure, is composed by a single access and aggregation ring, with x6 Azure Data Boxes<sup>4</sup> collocated with each M1 switch, 128 Raspberry Pi 3 B+<sup>5</sup>, and x1 PowerEdge R840 Rack Server<sup>6</sup> with 10 times more memory and disk than an Azure Data Box.

Regarding our pricing model we just consider that the deployment of a t3 instance is  $\delta_e$  times more expensive in Azure Data Boxes, and  $\delta_f$  times more expensive in a Raspberry Pi 3 B+. The Azure Data Boxes and Raspberry Pis have an associated volatility  $v$ , which is generated using an exponential random variable with mean  $\mu_e, \mu_f$ , respectively. Volatilities are assign to the devices using 5GEN *addNodeProps* function.

5G-CORAL experiments focus on minimizing the cost, while being capable of running longer even under volatile infrastructures. We thus use the lifetime cost as evaluation metric, which is the result of dividing the deployment cost by the hours the service works before an infrastructure component fails. Highly volatile infrastructures lead to high lifetime cost.

Fig. 4 shows how *LongRun-g* outperforms *CAPEX-g* as we increase the infrastructure volatility, i.e., as  $\mu_e, \mu_f$  increase. This is because *CAPEX-g* aims to minimize the volatility of nodes used to run the service, and they execute for longer before an infrastructure failure occurs.

The results shown in Fig. 4 also show that the impact of  $\delta_f$  is negligible with respect to  $\delta_e$  in the lifetime cost of the NS. This is because in the t3 AWS instances, just t3.nano instances are deployed in Fog CDs, and they are the less demanding ones in terms of resources and cost, which translates into a minor

<sup>3</sup><https://aws.amazon.com/ec2/instance-types/>

<sup>4</sup><https://azure.microsoft.com/en-us/services/databox/>

<sup>5</sup><https://www.raspberrypi.org/products/raspberry-pi-3-model-b-plus/>

<sup>6</sup><https://www.dell.com/en-us/work/shop/povw/poweredge-r840>

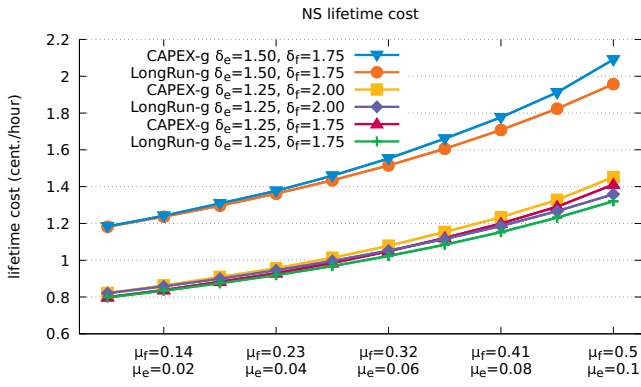


Fig. 4: 5G-CORAL heuristics over topology of Fig. 1

increase of the lifetime cost. That minor increase is depicted in curves ( $\delta_e = 1.25, \delta_f = 1.75$ ) and ( $\delta_e = 1.25, \delta_f = 2.0$ ) of Fig. 4.

#### IV. CONCLUSIONS

This paper presents 5GEN, an open source graph generator of 5G graphs. The developed R package exposes functions to create a topology as big as desired, and offers the possibility of customizing the properties of the generated infrastructure. The user can use 5GEN to create graphs with fog devices and edge servers, that are explicitly attached to switching nodes and AAUs of a reference 5G network. The tool is flexible enough to allow the edition of properties like the volatility of devices, which speeds up significantly testing different algorithms.

In order to validate our tool, we have successfully tested 2 reference 5G-CORAL orchestration algorithms against 5G network graphs created using 5GEN. While the first version of 5GEN focused on access and aggregation segments, future

work is expected to extend the tool by also integrating the core segment and inter-operator connectivity to support multi-domain scenarios.

#### REFERENCES

- [1] S. Agarwal, F. Malandrino, C. Chiasserini, and S. De, "Joint vnf placement and cpu allocation in 5g," in *IEEE INFOCOM 2018 - IEEE Conference on Computer Communications*, April 2018, pp. 1943–1951.
- [2] J. Li, W. Shi, Q. Ye, W. Zhuang, X. Shen, and X. Li, "Online joint vnf chain composition and embedding for 5g networks," in *2018 IEEE Global Communications Conference (GLOBECOM)*, Dec 2018, pp. 1–6.
- [3] E. W. Zegura, K. L. Calvert, and S. Bhattacharjee, "How to model an internetwork," in *Proceedings of IEEE INFOCOM '96. Conference on Computer Communications*, vol. 2, March 1996, pp. 594–602 vol.2.
- [4] 5GEx project, "D4.3; Experiment Description, Results, Analysis and Conclusions v3," Deliverable, July 2018. [Online]. Available: [http://www.5gex.eu/wp/?page\\_id=55](http://www.5gex.eu/wp/?page_id=55)
- [5] B. Németh, B. Sonkoly, M. Rost, and S. Schmid, "Efficient service graph embedding: A practical approach," in *2016 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN)*, Nov 2016, pp. 19–25.
- [6] S. Orłowski, R. Wessály, M. Pióro, and A. Tomaszewski, "Sndlib 1.0—survivable network design library," *Networks: An International Journal*, vol. 55, no. 3, pp. 276–286, 2010.
- [7] 5G-CORAL, "D3.1: Initial design of 5G-CORAL orchestration and control system," Deliverable, May 2018. [Online]. Available: <http://5g-coral.eu/wp-content/uploads/2018/06/D3.19802.pdf>
- [8] A. de la Oliva, X. Li, X. Costa-Perez, C. J. Bernardos, P. Bertin, P. Iovanna, T. Deiss, J. Mangués, A. Mourad, C. Casetti, J. E. Gonzalez, and A. Azcorra, "5g-transformer: Slicing and orchestrating transport networks for industry verticals," *IEEE Communications Magazine*, vol. 56, no. 8, pp. 78–84, August 2018.
- [9] L. Cominardi, L. M. Contreras, C. J. Bernardos, and I. Berberana, "Understanding QoS Applicability in 5G Transport Networks," in *2018 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB)*, June 2018, pp. 1–5.
- [10] G. Csardi and T. Nepusz, "The igraph software package for complex network research," *InterJournal*, vol. Complex Systems, p. 1695, 2006. [Online]. Available: <http://igraph.org>
- [11] A. Baddeley, C. internazionale matematico estivo, and W. Weil, *Stochastic Geometry: Lectures Given at the C.I.M.E. Summer School Held in Martina Franca, Italy, September 13-18, 2004*, ser. Lecture Notes in Mathematics / C.I.M.E. Foundation Subseries. Springer, 2007.