
This is an electronic reprint of the original article.
This reprint may differ from the original in pagination and typographic detail.

Laghrissi, Abdelquoddous; Taleb, Tarik; Bagaa, Miloud
Canonical domains for optimal network slice planning

Published in:
IEEE Wireless Communications and Networking Conference, WCNC 2018

DOI:
[10.1109/WCNC.2018.8377336](https://doi.org/10.1109/WCNC.2018.8377336)

Published: 01/01/2018

Document Version
Peer-reviewed accepted author manuscript, also known as Final accepted manuscript or Post-print

Please cite the original version:
Laghrissi, A., Taleb, T., & Bagaa, M. (2018). Canonical domains for optimal network slice planning. In *IEEE Wireless Communications and Networking Conference, WCNC 2018* (IEEE Wireless Communications and Networking Conference). IEEE. <https://doi.org/10.1109/WCNC.2018.8377336>

This material is protected by copyright and other intellectual property rights, and duplication or sale of all or part of any of the repository collections is not permitted, except that material may be duplicated by you for your research use or educational purposes in electronic or print form. You must obtain permission for any other use. Electronic or print copies may not be offered, whether for sale or otherwise to anyone who is not an authorised user.

Canonical domains for Optimal Network Slice Planning

Abdelquoddouss Laghrissi, Tarik Taleb and Miloud Bagaa

Communications and Networking Department, Aalto University, Finland.

Emails: abdelquoddouss.laghrissi@aalto.fi, talebtarik@ieee.org, miloud.bagaa@aalto.fi

Abstract—The existing conventional mobile networks are not flexible: if there is a new service, it unfortunately cannot be integrated automatically. Their traffic routing is not optimal and users’ traffic is forwarded to the core network without considering the optimal path. This causes high latency to access the desired service, and the use of resources is inefficient. This has motivated the evolution towards 5G. The 5G vision consists of managing highly dynamic network slices and provisions networks in an as-a-service fashion. In this vein, to answer to the elasticity and low-latency specifications of the upcoming 5G services, the optimal placement of Virtual Network Functions (VNFs) must overcome the non-uniform service demand and the irregular nature of the underlying network topologies. This paper addresses this issue by mapping the non-uniform signaling messages to a new uniform environment, namely, the canonical domain, whereby the placement of core functions is more feasible and efficient. This is carried out by using Schwartz-Christoffel conformal mappings. The conducted experimentation shows the efficiency of our approach, compared to some baseline approaches, in the virtual resource allocation (i.e. Virtual CPU, Virtual DISK) and that is in terms of reducing the overall cost, end-to-end delay and number of activated Virtual Machines (VMs; virtual resources in general).

I. INTRODUCTION

Network Function Virtualization (NFV) helps to achieve the virtualization of network functions and services provided by a dedicated hardware, and reduces the amount of virtual and hardware resources needed to launch and operate network services. Within the same data center (DC) or across multiple ones, different VNFs can be instantiated and inter-work, which results into a dynamic and flexible network that can be deployed, rapidly, in the Cloud. NFV enables the elastic scaling and rapid deployment of network functions, replacing the need to set up and maintain the corresponding hardware. The provision, deployment and management of such VNFs over virtualized infrastructures for service providers will definitely profit to IT companies.

New architectures that combine mobile networks and cloud computing took place to answer to the need of service customization, taking into account the several requirements of end-users, their mobility patterns, and their preferences in where, when, and how to consume services. This architectural work has motivated many research works [8]. In this vein, the Multi-access Edge Computing (MEC) concept has been coined to offer processing and storage capabilities nearby end-users. This has been achieved by placing Edge Cloud (EC) servers jointly with the mobile network base stations, which erases the need to move computation and storage of intensive

applications from a UE to a faraway centralized cloud server [3].

Also, network slicing made it possible to provide a flexible network and support different verticals within different architectures and with multiple requirements [1], as it divides the traditional single network into multiple instances; each instance is dedicated to meet a given requirement or specific service [19], [22]. In order to answer to the increasing service requirements of incoming requests, the network slices must join the cloud resources (e.g. bandwidth, CPU, etc) with the network architecture (e.g. network functions) [16]. Slices which basically consist of a set of VNFs can be dedicated to several verticals such as, virtual reality, and autonomous driving.

In this vein, cloud owners and service providers must determine what resources are required to operate and migrate in the cloud successfully and at the most convenient costs. By putting resources in the vicinity of cloud users, with the required amount of virtual resources to deploy, efficient Capital Expenditure (CAPEX) and Operational Expenditures (OPEX) can be achieved.

Our paper considers the provisioning of virtual resources and considers dealing with the deployment of virtual resources to run virtualized network functions, in an already set-up EC environment, and using the characteristic of “uniformity” offered by conformal maps (i.e. the preservation of angles). It also introduces many challenges, mainly, the conformal mapping spatial transformation and its corresponding inverse function, as well as, the theoretical and practical challenges for the analysis in canonical domains.

The remainder of the paper is structured as follows. Section II discusses related work. We define our problem in Section III. We present our solution, dubbed “Canonical Domain Framework for Slice Planning (CDSP-F)”, in Section IV. Experimentation and results of our framework, compared to other solutions, are given in Section V. Finally, the paper concludes in Section VI highlighting some future research work.

II. RELATED WORK

Several works addressed the optimal placement of specific VNFs with respect to the requirements of several 5G verticals. For video streaming, the ability to move transcoding resources could help to optimize the latency and bandwidth, thus, achieving better Quality of Experience (QoE) [10], [20].

It is also possible to migrate dynamically these resources to different locations, while the streaming is taking place thanks to coupling the open cloud computing architecture and NFV, and using heuristics. Game-theoretic models are used to evaluate the needed joint caching for content providers, transit networks and access networks [4]. The content caching in wireless networks can exploit the backhaul links for collaborative caching offering better pricing choices and better QoS [11]. Also, a Content Delivery Network As A Service (CDNaaS) platform was proposed to manage a high number of videos deployed on virtualized caches, transcoders, and streamers using a Gurobi Optimization tool with the objectives of maximizing the QoE of the streaming service and reducing the cost [21].

As key VNFs of the EPC architecture, many solutions addressed the optimal placement of Packet Data Network Gateways (P-GW) and Serving Gateways (S-GW). The work in [6], [7], [13] aimed to minimize the number of SGW relocations, shorten the path between UEs and the respective P-GWs, and reduce the associated overhead, respectively. In contrast to these works, our proposed solution takes into account the geographic distribution of variant ECs, as well as the spatio-temporal distribution of users' requests.

The Schwartz-Christoffel mapping was proposed in [12] to cope with the non-uniform distribution of nodes in cellular networks, by taking the analysis from the physical domain to the canonical domain, in order to make the best placement choices of access points. It was also used in [2] for routing in a mobile network and avoiding local minimums by encoding the map of the network domain to virtual coordinates of the canonical domain. To the best knowledge of the authors, our work is the first to apply canonical domains in EC environments. Our solution copes with the non-uniform signaling messages distribution (i.e., physical domain) which was not considered in the previous works. Connecting the physical domain to the canonical domain helped to deal with the irregularities in terms of the compatibility between capacity provision and service demand by defining a uniform distribution that makes it easier to dimension and plan the resources distribution, as the work in the canonical domain is easier and more efficient than the analysis of the irregular spatio-temporal model of users' requests.

III. PROBLEM FORMULATION

The objective of the paper is to place VMs in optimal positions for serving the variant users while QoE is ensured and the cost is minimized. We assume that the network consists of a set of ECs, whereby each one is supposedly positioned in an optimal place (i.e. at the edge of the network) in closer proximity to where the data is generated (i.e., as per the edge computing paradigm). Let us consider the usage of a vertical in the area translated by the generation of diverse signaling messages from UEs. We can summarize the problem through the following question: how to optimally put the VMs, running the vertical's VNFs, in the vicinity of places where the signaling messages are generated?

The deployment of VMs needed in the vicinity of the ECs in place should guarantee QoS, and reduce both the end-to-end latency and the energy consumption. These requirements are difficult to jointly meet as the service deployment has a complicated non-uniform distribution [17], [18]. This makes it difficult to directly decide where to position the VMs. To simplify the problem, the conformal mapping technique is used to obtain a uniform distribution of signaling messages that makes it easier to make such decisions. This will be detailed in Section IV.

A. Service usage

The deployment of services in our system is given as follows:

- Sessions lengths are highly variable, when users connect to each service. The distribution is fitted to a Zipf distribution of the form: $\beta e^{-\alpha}$
- The inter-arrival time of the i^{th} and $(i+1)^{th}$ sessions is given as time series $a(i)$ defined by $t(i+1) - t(i)$. $a(i)$ is fitted to a log-normal distribution.
- The service demand volume V_{SD} is modeled in terms of inter-arrival and session times. It denotes the average number of users in the system. V_{SD} is therefore expressed by:

$$V_{SD} = \frac{E(\text{sessiontime})}{E(\text{interarrival})} = \frac{\beta}{\beta-1} \frac{\max(x_t, x_0)}{e^{\frac{\mu+\sigma^2}{2}}} \quad (1)$$

With:

- μ and σ are, respectively, the mean and standard deviation of the variable natural logarithm.
- β is the scaling factor and α is a constant exponent.
- The mean for the log normal distribution is:

$$E(X) = e^{\frac{\mu+\sigma^2}{2}} \quad (2)$$

- $x_0 = \alpha^{\frac{1}{\beta}}$. It is also worth noting that the density function f_X , for the Zipf distribution, of X is such that for every x:

$$f_X(x) = \frac{\alpha\beta}{x^{\beta+1}} \quad (3)$$

And that:

$$E(X; X > x) = \int_x^\infty t f_x(t) dt \quad (4)$$

IV. CANONICAL DOMAIN FRAMEWORK

In this section, we will define the main functions used for our canonical domain framework, namely the mapping transformation functions (from the physical domain to the canonical domain and vice-versa), the service density distribution functions δ and δ' , in the physical domain and the canonical domain, respectively. In addition, we define the algorithm of the CDSP-F core function.

A. The spatial transformation functions of CDSP-F

A conformal map is the transformation of a complex valued function from one coordinate system to another, by applying a transformation function to the original complex function. The original complex function defines the physical domain A (non-uniform distribution) and the obtained new function (using the given transformation) defines the canonical domain R (uniform distribution). The objective is to translate our non-uniform service distribution using the transformation function into a Schwartz-Christoffel conformal mapping, to obtain a uniform distribution of service demand from the physical domain (x, y) to the canonical domain (u, v) , to deploy uniformly the VMs needed for core functions in the canonical domain (u, v) and then to obtain the physical position (x, y) of those VMs using the inverse function of the transformation function. The signaling messages are generated in a disk with a radius R while for the sake of efficiency as stated in [12], the targeted canonical domain is mapped as a rectangle divided into rectangular tiles.

To do so, the mapping function that maps every point (u, v) in the circular disc to a point (x, y) in the square region and the inverse function must be found. In other words, this function must have derivative equations such that $(u, v) = F(x, y)$ and $(x, y) = F^{-1}(u, v)$. Also, the equations will map each point (u, v) to point (x, y) . Similar in spirit to [9], we model the service distribution in the simulation area as a rectangular grid inside the Disk (i.e., the radius of the disk is equivalent to the radius of service deployment R):

$$F(x + iy) = u(x, y) + iv(x, y) \quad (5)$$

$$u = \operatorname{Re}\left(\frac{1-i}{\sqrt{2}} \operatorname{cn}\left(K_e \frac{1+i}{2}(x+iy) - K_e, \frac{1}{\sqrt{2}}\right)\right) \quad (6)$$

$$v = \operatorname{Im}\left(\frac{1-i}{\sqrt{2}} \operatorname{cn}\left(K_e \frac{1+i}{2}(x+iy) - K_e, \frac{1}{\sqrt{2}}\right)\right) \quad (7)$$

$$x = \operatorname{Re}\left(\frac{1-i}{-K_e} E\left(\cos^{-1}\left(\frac{1+i}{\sqrt{2}}(u+iv)\right), \frac{1}{\sqrt{2}}\right)\right) + 1 \quad (8)$$

$$y = \operatorname{Im}\left(\frac{1-i}{-K_e} E\left(\cos^{-1}\left(\frac{1+i}{\sqrt{2}}(u+iv)\right), \frac{1}{\sqrt{2}}\right)\right) - 1 \quad (9)$$

With:

- K_e is a constant which depends on the radius R . Its exact value is the complete Legendre elliptic integral of the 1st kind with modulus m . This constant arises from Schwarz and Christoffel's equations for the specific case when the desired polygonal shape is a square.
- cn is a Jacobi elliptic function
- E_f is the incomplete elliptic integral of the first kind:

$$E_f(z|m) = \int_0^z \frac{1}{\sqrt{1-m\sin^2(t)}} dt \quad (10)$$

The use of E_f is not only motivated by the fact that 1) $E_f(z|m)$ is an analytical function of z and m which are defined over \mathbb{C}^2 which allows its use for conformal mapping (i.e., it preserves angles throughout the whole mapping) and

2) through the inverse Jacobi elliptic functions we have an interesting property:

$$E_f(\cos^{-1}(z)|m) = \operatorname{cn}^{-1}(z|m) \quad (11)$$

with: $-1 < z < 1 \wedge m \in \mathbb{R}$. The choice of the analytic components u and v reflects the spatial service demand distribution, which is graphically fitted to a rectangular grid within a disk.

After the definition of the mapping function F , the question is how to link it to the service distribution density δ in order to be able to deduce the uniform service distribution density δ' in the canonical domain.

B. Service density distribution

In this subsection, we define δ and δ' the service density distributions in the physical domain and the canonical domain, respectively. Considering the physical concept of mass density, an object is dense if it has a large amount of mass over a relatively small area. Similarly, we can say that a region is dense if it has a large amount of signaling messages for it is relatively small. We can start by writing the following:

$$\ddot{\delta} = \frac{|S_m|}{\dot{V}} \quad (12)$$

$\ddot{\delta}$ is the density (i.e the ratio of the mean of signaling messages and spatial volume \dot{V}) and $|S_m|$ is the mean of signaling messages. We can go further and suppose that the region is divided into small areas and consider the density of each area. Let us think of it as a continuum, which is the case of the physical domain we are considering. We can take the process to the limit where we are finding the density at each point in the area. This will result in the following:

$$\ddot{\delta}(\vec{V}) = \lim_{\dot{V} \rightarrow 0} \left(\frac{|S_m|}{\dot{V}} \right) \quad (13)$$

The reverse process of calculating the mean of signaling messages of a given small area a is an integral over the spatial volume:

$$|S_m|_a = \iiint_V \ddot{\delta}(\vec{V}) d\vec{V} \quad (14)$$

Using probability density, we replace “mean of signaling messages” with “probability” and spatial volume \dot{V} with volume V in the parameter space considering that our signaling messages follow the log normal distribution. This will result in the following:

$$\begin{aligned} \delta &= \frac{d}{dx} \operatorname{Pr}(X \leq x) \\ &= \frac{d}{dx} \operatorname{Pr}(\ln(X) \leq \ln(x)) \\ &= \rho\left(\frac{\ln(x) - S_m}{V}\right) \frac{d}{dx} \left(\frac{\ln(x) - S_m}{V}\right) \\ &= \rho\left(\frac{\ln(x) - S_m}{V}\right) \frac{1}{S_m x} \\ &= \frac{1}{x} \frac{1}{S_m \sqrt{2\pi}} e^{-\frac{(\log(x)-V)^2}{2S_m^2}} \end{aligned} \quad (16)$$

Note that S_m and V depend on the inputs of the simulation (i.e. number of signaling messages, inter-arrival of sessions, and duration of a session). The analysis in the canonical domain aims at finding the number of VMs that is required to satisfy the uniformly distributed service demand with volume V . As stated before, we chose the rectangular tiling for the sake of efficiency. As the service demand is uniformly distributed in R , the uniform density can be written as follows [12]:

$$\delta' = \frac{L}{H.W} \quad (17)$$

W and H are the width and height of the rectangle, respectively, and L is the number of VM clusters. The latter can be obtained as follows:

$$L = \frac{2WH}{\left(\frac{W^2}{4} + \frac{H^2}{4}\right)3\sqrt{3}} \quad (18)$$

To be able to distribute the VMs uniformly within the rectangular tiles, taking into consideration the data usage generated in the canonical domain, we denote by L_{Argmin} the effective number of VMs needed as the following:

$$L_{Argmin}(x) = Argmin\left(\frac{x}{VMcapacity}\right) \quad (19)$$

Where x is the variable data usage of given signaling messages.

C. Algorithm of the core function of CDSP-F

As depicted in Algorithm 1, the planning of VM positioning is carried out using the conformal mapping technique. The first step consists of collecting the positions where the data are generated in the current snapshot of time using Equations (1) and (16). In the second step, using the mapping functions (6) and (7), we obtain the positions and occurrences of signaling messages in the canonical domain. In the third step, due to the uniform nature of the canonical domain, we divide the canonical domain in rectangle tiles using Equations (17) and (18). In addition to these equations, we use the argmin function (19) to know the needed number of VMs that could fulfill the requests generated in each tile of the canonical domain. In the fourth step, we apply Equations (8) and (9) to obtain the positions of generated VMs in the physical domain. In the last step, using Voronoi diagram, we allocate the VMs to the corresponding ECs in place.

V. EXPERIMENTATION AND RESULTS

In this section, we evaluate the performance of the proposed solution CDSP-F, and compare it to the existing base-line approaches, which are *i*) an adaptation of the Best Fit Algorithm (BFA) [5] and *ii*) the Random Fit Algorithm (RFA) [15]. The variant solutions have been evaluated in terms of end to end delay, cost defined in terms of number of activated VMs, and the execution time.

Algorithm 1 CDSP-F core function algorithm

Require:

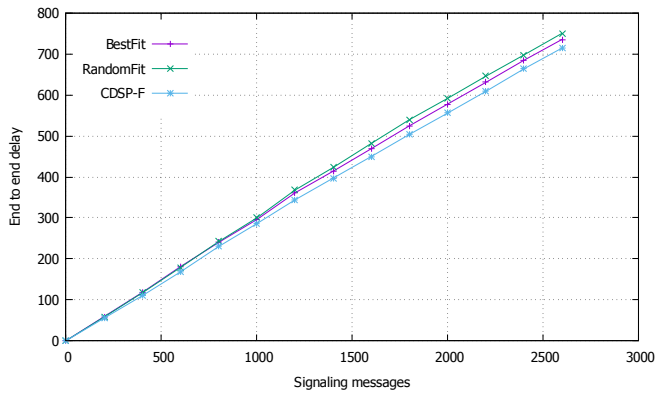
- Γ : Signaling messages.
- X : A set of x positions in the physical domain of signaling messages.
- Y : A set of y positions in the physical domain of signaling messages.
- X_e : A set of x positions in the physical domain of edge clouds.
- Y_e : A set of y positions in the physical domain of edge clouds.

Ensure:

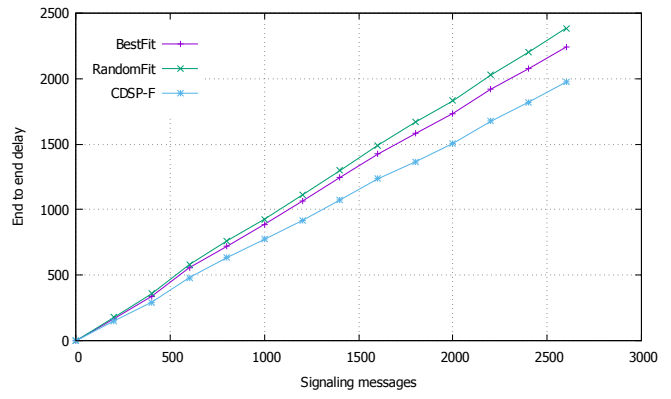
- U : The set of signaling messages' u positions in the canonical domain obtained by F
 - V : The set of signaling messages' v positions in the canonical domain obtained by F
 - X_{VM} : The set of VM' x positions in the physical domain obtained by F^{-1}
 - Y_{VM} : The set of VM' y positions in the physical domain obtained by F^{-1}
 - 1: **for all** $\Gamma_i \in \Gamma$ **do**
 - 2: $(U_i, V_i) = F(X_i, Y_i)$; {Calculate position(u, v) in the canonical domain}
 - 3: $GetNeededResources(\Gamma_i)$;
 - 4: **if** $NeededResources_{\Gamma_i} > VMCapacity(U_i, V_i)$ **then**
 - 5: $CreateNewVM(NeededResources_{\Gamma_i}, U_i, V_i)$;
 - 6: **else**
 - 7: $UpdateCurrentVM(NeededResources_{\Gamma_i})$;
 - 8: **end if**
 - 9: **end for**
 - 10: $(X_i, Y_i) = F^{-1}(U_i, V_i)$ {Calculate position(x, y) in the physical domain}
 - Cluster VMs of the Physical Domain using Voronoi diagram
 - 11: $\mathcal{AL} = VO(X_e, Y_e, X, Y)$;
 - 12: **return** $\mathcal{AL}s$;
-

The variant algorithms are evaluated by using our java-based simulator that defines a spatio-temporal model of mobile service usage over a particular geographical area [14]. All the algorithms have been executed in the same environment using Ubuntu 16.04 on an Intel Core i5 2 core CPU and 16 GB of RAM. The algorithms are evaluated by varying the radius of deployment area and the number of signaling messages. We conducted two sets of experiments: *i*) First, we fixed the radius of the deployed area to 20 *km*; *ii*) Second, we fixed the radius area to 40 *km*. In both experiments, we varied the number of signaling messages from 0 to 2600.

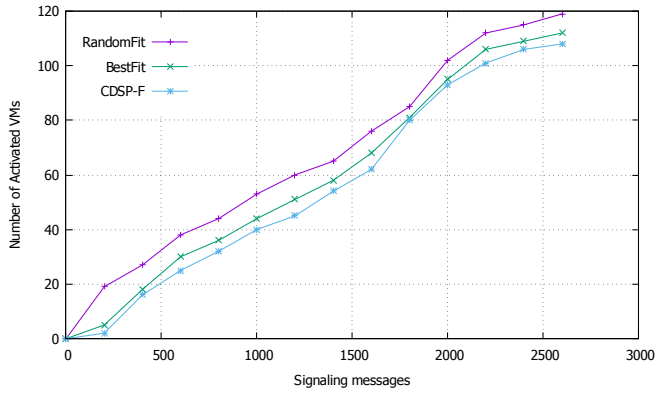
From Fig. 1(a) and Fig. 1(b), we can observe that when the number of signaling messages is low (inferior to 300), the gap in terms of end to end delay is relatively low for the three algorithms. As the number of signaling messages



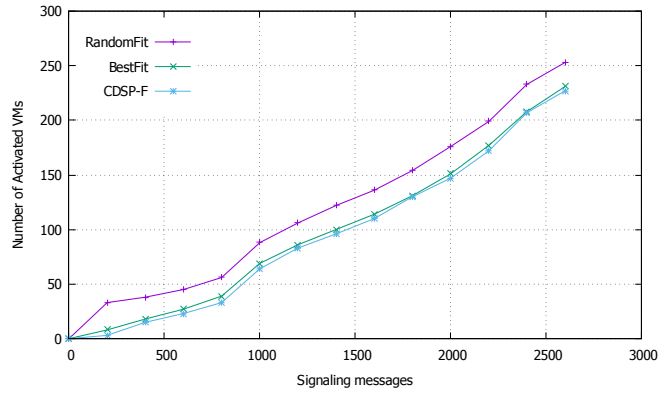
(a) End to end delay in 20 km radius



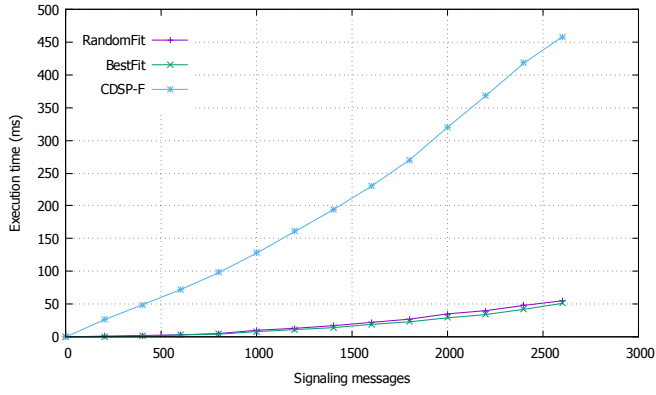
(b) End to end delay in 40 km radius



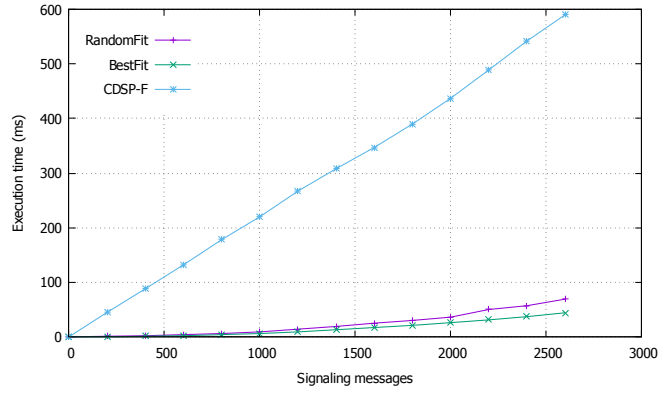
(c) Number of active VMs in 20 km radius



(d) Number of active VMs in 40 km radius



(e) Execution time in 20 km radius



(f) Execution time in 40 km radius

Fig. 1: The performances evaluation of variant solutions in terms of end to end delay, Cost and execution time

increases, the analysis in the canonical domain gives better positions of VMs in terms of distance to where the signaling messages are triggered, which encourages their usage by the neighboring UEs and impacts positively the total number of created resources without an impinge on the delay. Moreover, as CDSP-F continues to give satisfying results, the gap compared to BFA and RFA becomes more important when the number of signaling messages is high and the deployment area is bigger (i.e. from 20Km radius area to 40 Km radius area).

The number of signaling messages has a negative impact on the cost as depicted in Fig. 1(c) and Fig. 1(d), due to the

fact that more virtual resources are needed. The performance gap related to the number of activated VMs is very important when the radius of service deployment increases. This can be explained by the need to create resources in the proximity of where signaling messages are generated. Those resources are randomly generated by RFA, which is translated into the continuously high number of activated VMs, while BFA and CDSP-F have almost equal numbers of activated VMs, with better results in favor of CDSP-F, i.e., not exceeding 210 VMs as depicted in Fig. 1(d). On the one hand, BFA is known for seeking for VMs with the best available resources,