Joint Base Station Selection and Adaptive Slicing in Virtualized Wireless Networks

Kory Teague, Mohammad J. Abdel-Rahman, and Allen B. MacKenzie
Department of Electrical and Computer Engineering, Virginia Tech, USA
{koryt, mo7ammad, mackenab}@vt.edu

Abstract

Wireless network virtualization is a promising avenue of research for next generation 5G cellular networks. Virtualization focuses on the concept of active resource sharing and the building of a network designed for specific demands, decreasing operational expenditures and improving demand satisfaction of the cellular network. This work investigates the problem of selecting base stations to construct a virtual network that meets the the specific demands of a service provider, and the adaptive slicing of the resources to the service provider's demand points. A two-stage stochastic optimization problem is introduced modeling this interaction, and present two approaches for determining a solution for the model. The first approach uses a sampling approach applied to the deterministic equivalent program of the stochastic model, providing a tight approximation with sufficient time. The second approach uses a genetic algorithm to approximate base station selection for the network, allowing for adaptive slicing to be determined by a simpler, single-stage optimization problem. Scenarios testing these approaches are generated using a log-normal model designed to emulate demand from real world cellular networks. The results from these simulations present promising results that can be expounded upon with further investigation.

I. Introduction

Resource infrastructure sharing has been a common practice between mobile network operators (MNOs) to fulfill two separate needs. First, MNOs needed to offer coverage for their users in regions where they had no infrastructure, leading to the creation of roaming agreements. Second, the cost to maintain the wireless infrastructure became high in some areas. Passive sharing of infrastructure, such as physical sites, tower masts, power, and air-conditioning, led to considerable cost savings [1]. These benefits motivated the use of more active sharing approaches, such as the reuse of backhaul and the sharing of radio access networks (RANs) in the form of base stations (BSs) and antenna systems. The technologies developed for MNO sharing have allowed for the rise of mobile virtual network operators (MVNOs), which provide MNO-like mobile services without directly owning a physical RAN.

Wireless virtualization is one of the most promising approaches for efficient sharing of radio resources in next-generation mobile networks [2]. In [3], [4], the opportunities for cost saving and additional flexibility are the motivations behind introducing virtualization schemes for LTE networks, focusing on resource sharing.

Doyle *et al.* [5] presented the Network without Borders (NwoB) paradigm which broadly explores the concepts of virtualization. The authors proposed removing the traditional constraints on spectrum and presented a virtualization-centric structure for mobile wireless networks. In this paradigm, a service provider (SP) is an MVNO that provides a specified service to its users, which could be in the form of traditional general data, voice, and messaging services or a more specific application (e.g. emergency services, internet of things, or video streaming). The virtual network builder (VNB) aggregates and selects resources from resource providers (RPs) to build virtual networks designed for the needs of the SPs. RPs, such as traditional MNOs, are the owners and maintainers of the physical resources and infrastructure that are provided for the virtualized wireless networks (VWNs) via contracts established with VNBs.

In this paper we consider the problem of BS resource allocation and the slicing of this joint resource selection in VWNs. First, we establish a stochastic optimization problem from the perspective of the VNB to determine the optimal joint selection of resources to be leased from the RPs and adaptively sliced and allocated to form VWNs that satisfy the demands of the SPs. We then develop two approaches that would be run in the VNB to reach a near-optimal solution of the stochastic optimization problem. We consider two optimality criteria: maximizing demand satisfaction of SP users and minimizing the cost of the BS resources to satisfy the demand. Then, we consider the efficacy of the approaches with a single SP with log-normal spatially-correlated demand modeled to mimic real cellular networks.

The rest of this paper is organized as follows. In Section II, we detail and define the system model assumed for our resource allocation methods. In Section III, we consider our stochastic optimization problem and resource selection and demand allocation approaches. In Section IV, we simulate the described approaches and evaluate their performance. Finally, in Section V, we discuss our conclusions and possible future extensions.

II. System Model

We consider a geographical area of width X (m) and length Y (m) that contains a set $\mathcal{S} = \{1, 2, \ldots, S\}$ of BSs available to be leased to the VNB by a set of N_{RP} RPs. The rate capacity of BS $s \in \mathcal{S}$ is denoted by r_s, its cost is denoted by c_s, and its broadcasting range is denoted by b_s.

A Service Provider (SP) seeking a Virtualized Wireless Network from the VNB is assumed to know the distribution of traffic demand within the region the VWN would cover. It has been shown that a log-normal distribution or a mixture of log-normal distributions can approximate traffic demand in real-world cellular networks [6], [7]. It has also been shown that traffic distribution is spatially correlated [7], [8]. We model the spatial traffic demand of a single SP using a similar, continuous form of the SSLT (Scalable, Spatially-correlated, and Log-normally distributed Traffic) model as proposed by Lee, Zhou, and Niu [9].

To generate this spatial distribution over the area of consideration, an initial Gaussian field, $\rho^G = \rho^G(x, y)\,,\ x \in [0, X]\,,\ y \in [0, Y]$, is generated by

$$\rho^G(x, y) = \frac{1}{L} \sum_{l=1}^{L} \cos(i_l x + \phi_l)\ \cos(j_l y + \psi_l) \tag{1}$$

where $\mathcal{L} \stackrel{\text{def}}{=} \{1, 2, \ldots, L\}$ is a set of the products of two cosines with angular frequencies $i_l, j_l \sim \mathcal{U}(0, \omega_{\max}), l \in \mathcal{L}$ and phases $\phi_l, \psi_l \sim \mathcal{U}(0, 2\pi), l \in \mathcal{L}$. As L increases, ρ^G approaches a Gaussian random field with a spatial autocorrelation dependent on $\omega_{\max}$ according to the central limit theorem.

The approximate Gaussian distribution ρ^G is then normalized to $\rho^S = \rho^S(x, y), x \in [0, X], y \in [0, Y]$, which has a standard normal distribution. The final log-normal distribution, $\rho = \rho(x, y), x \in [0, X], y \in [0, Y]$, is determined by assigning location and scale parameters μ and σ

$$\rho(x, y) = \exp\left(\sigma\, \rho^S(x, y) + \mu\right) \tag{2}$$

$\rho(x, y)$ can be sampled over the space into individual pixels as per Lee with each pixel's value indicating the number of homogeneous demand points within the pixel [9]. In contrast, we allow $\rho(x, y)$ to provide a continuous, spatially-correlated log-normal distribution depicting the demand density over the region for the SP.

Let $\mathcal{M} \stackrel{\text{def}}{=} \{1, 2, \ldots, M\}$ be the set of the SP's demand points seeking to connect to the VWN; the value of total traffic demand at each point is denoted by d_m. Further let $u_{ms} \in [0, 1]$, $m \in \mathcal{M}$, $s \in \mathcal{S}$, represent the normalized capacity (with respect to r_s) of BS s at point m, i.e., the normalized maximum rate that a user can receive at point m from BS s. $u_{ms} = 0$ when m is outside the coverage area of s and $u_{ms} = 1$ when m is within a small distance of s. The specific position of point m, and therefore the value of u_{ms}, is determined stochastically by the demand field ρ.

We assume that a BS $s \in \mathcal{S}$ can be allocated between multiple demand points, and $\delta_{ms} \in [0, r_s]$, $m \in \mathcal{M}$, $s \in \mathcal{S}$, represents the rate of BS s that is allocated to point m.

Throughout this paper, stochastic variables will be differentiated from deterministic variables with a tilde (~) placed above the symbol.

III. Solution Approach

In this section, we detail our approaches for selecting the subset of resources within $\mathcal{S}$ to create a virtual network with the minimum cost while allocating the selected resources to the demand points within the region such that it maximizes demand satisfaction. First, we formulate the problem of BS selection and their allocation to individual demand points as a two-stage stochastic optimization problem. Second, the optimization problem is converted to its deterministic equivalent program to remove the stochastic variables then approximated using a sampling method. Finally, a genetic algorithm is used as a separate approximation to the BS selection portion of the problem. In Section IV, we will consider the efficacy of these latter two approaches.

A. Problem Formulation

We formulate the presented problem as a two-stage stochastic optimization problem. We introduce $z_s, s \in \mathcal{S}$ as a binary decision variable defined as

$$z_s = \begin{cases} 1, & \text{if BS } s \text{ is selected for the created network,} \\ 0, & \text{otherwise} \end{cases}$$

To balance the interest of maximizing demand satisfaction against minimizing cost, we introduce the positive real number α as a weighting coefficient between the two stages.

Problem 1 (Two-Stage Stochastic Optimization Problem)

$$\underset{\{z_s,\, s \in \mathcal{S}\}}{\text{minimize}} \left\{ \sum_{s \in \mathcal{S}} c_s \; z_s + \alpha \mathbb{E}\left[h\left(\boldsymbol{z},\, \tilde{\boldsymbol{u}}\right)\right] \right\} \tag{3}$$

subject to:

$$z_s \in \{0, 1\}, \forall s \in \mathcal{S} \tag{4}$$

where $h(\boldsymbol{z}, \tilde{\boldsymbol{u}})$ is the optimal value of the second-stage problem, which is given by:

$$\underset{\{\delta_{ms}, m \in \mathcal{M}, s \in \mathcal{S}\}}{\text{minimize}} \left\{ - \sum_{m \in \mathcal{M}} \sum_{s \in \mathcal{S}} \delta_{ms} \; \tilde{u}_{ms} \right\} \tag{5}$$

subject to:

$$x_s = \mathbb{1}_{\left\{\sum_{m \in \mathcal{M}} \delta_{ms} > 0\right\}}, \forall s \in \mathcal{S} \tag{6}$$

$$\sum_{s \in \mathcal{S}} \delta_{ms} \; \tilde{u}_{ms} \leq d_m, \forall m \in \mathcal{M} \tag{7}$$

$$\sum_{m \in \mathcal{M}} \delta_{ms} \leq r_s, \forall s \in \mathcal{S} \tag{8}$$

The first stage objective function (**??**) minimizes the total cost of the selected network in context to that network's ability to satisfy the demand contained within the region, as determined by ρ. The second stage objective function (**??**) maximizes demand satisfaction by maximizing the total demand allocated to the resources comprising the network, as specified by δ_{ms} as the decision variable of the second stage. In this context, we define demand satisfaction as the ratio of the total demand allocated to the selected network to the total demand contained within the region.

Constraints (**??**), (**??**), and (**??**) implement the defined ranges and values of the decision variables z_s and δ_{ms}, with (**??**) ensuring that demand is allocated only to selected resources. For constraint (**??**), $\mathbb{1}_{\{*\}}$ is defined by

$$\mathbb{1}_{\{*\}} = \begin{cases} 1, & \text{if condition } * \text{ is true,} \\ 0, & \text{otherwise} \end{cases}$$

Constraint (**??**) ensures a demand point $m \in \mathcal{M}$ is not allocated more resource capacity than it demands.

B. Deterministic Equivalent Reformulation

One major obstacle with solving Problem 1 is that stochastic equations are impossible to solve with available linear programming tools. In order to be practical, Problem 1 must be reformulated such that it does not contain any stochastic variables.

Our approach for solving the proposed stochastic optimization formulation is to derive their deterministic equivalent programs (DEPs). The DEP is an equivalent reformulation of the original stochastic program, but only contains deterministic variables [10].

Let $\Omega \stackrel{\text{def}}{=} \{1, 2, \ldots, O\}$ be defined as the set of discrete scenarios, each of which contains a sampled version of the stochastic variables within Problem 1. As O approaches infinity, Ω contains the entire scope of the stochastic variables. The probability a given scenario $\omega \in \Omega$ occurs is denoted by $p^{\{\omega\}}$, $\omega \in \Omega$, where $\sum_{\omega\in\Omega} p^{\{\omega\}} = 1$. Variables that are dependent on Ω are shown with a superscript $\{\omega\}$ with the specific scenario it is dependent on indicated by ω.

Problem 2 (Deterministic Equivalent Program of Problem 1)

$$\underset{\substack{\left\{ \substack{z_s,\, \delta_{ms}^{\{\omega\}},\\ s\in\mathcal{S},\, m\in\mathcal{M},\\ \omega\in\Omega} \right\}}}{\text{minimize}} \sum_{s\in\mathcal{S}} c_s\, z_s - \alpha \sum_{\omega\in\Omega} p^{\{\omega\}} \left(\sum_{m\in\mathcal{M}} \sum_{s\in\mathcal{S}} \delta_{ms}^{\{\omega\}}\, u_{ms}^{\{\omega\}} \right) \tag{9}$$

subject to:

$$\sum_{s\in\mathcal{S}} \delta_{ms}^{\{\omega\}}\, u_{ms}^{\{\omega\}} \le d_m,\ \forall m \in \mathcal{M},\ \forall \omega \in \Omega \tag{10}$$

$$\sum_{m\in\mathcal{M}} \delta_{ms}^{\{\omega\}} \le r_s\, z_s,\ \forall s \in \mathcal{S},\ \forall \omega \in \Omega \tag{11}$$

$$z_s \in \{0, 1\},\ \forall s \in \mathcal{S} \tag{12}$$

The objective function (**??**) combines both objective functions (**??**) and (**??**) of the initial formulation into a deterministic form. Constraints (**??**) and (**??**) ensure demand is not overallocated and is only allocated to selected resources and within capacity for all scenarios.

While Problem 2 provides an equivalent deterministic form of Problem 1, it only remains so while Ω contains the entire scope of the stochastic variables (i.e., Ω contains every scenario that the stochastic variables can take with their associated probabilities occurring). For Problem 2 to be in a generally tractable form, Ω must be a finite set. We use a sampling approach, sampling Ω with a finite number of scenarios, O. With sufficiently large O, Ω approaches a tight approximation of the scope of the stochastic variables. Within each scenario $\omega \in \Omega$, the SSLT demand field ρ is sampled to provide a set of M discrete demand points. Each sampling of ρ is generated by creating a non-stationary 2D Poisson point process (PPP) with M points over the region using ρ as the spatial intensity function. To generate this non-stationary PPP, we use an acceptance-rejection method. That is, each point of a stationary PPP with an intensity of $\rho_{\max} =$

$\max_i \rho\left(x_i, y_i\right)$ is retained with probability $\frac{\rho\left(x_i, y_i\right)}{\rho_{\max}}$, where x_i and y_i are the x- and y-coordinates of the i^{th} point of the stationary PPP.

C. Adaptive Slicing within a Formed VWN

After the solution to the sampled DEP of Section III-B has been found, the VNB has determined the joint BS selection that forms the VWN and a proposed resource slicing of considered possible scenarios, Ω, that allocates the resources to the SP's demand points. Since O is not infinite, any given scenario present in the formed VWN is exceedingly unlikely to be an element of Ω. Further, as demand points move between BSs or enter or exit the VWN, a new scenario $\omega \notin \Omega$ is formed with a possibly new set $\mathcal{M}' \stackrel{\text{def}}{=} \{1, 2, \dots, M'\}$. The VWN must adapt its resource slicing to these new demand points to maintain maximal demand satisfaction. With the VWN built, the joint BS selection, z_s, becomes a constant of the network, simplifying Problem 2 to a single-stage optimization problem

Problem 3 (Deterministic Adaptive Slicing)

$$\underset{\left\{\substack{\delta_{ms}^{\{\omega\}},\, s\in\mathcal{S}, \\ m\in\mathcal{M}',\, \omega\in\Omega'}\right\}}{\text{minimize}} \; -\sum_{\omega\in\Omega'} p^{\{\omega\}} \left(\sum_{m\in\mathcal{M}'} \sum_{s\in\mathcal{S}} \delta_{ms}^{\{\omega\}}\, u_{ms}^{\{\omega\}}\right) \tag{13}$$

subject to:

$$\sum_{s\in\mathcal{S}} \delta_{ms}^{\{\omega\}}\, u_{ms}^{\{\omega\}} \le d_m,\ \forall m \in \mathcal{M}',\ \forall \omega \in \Omega' \tag{14}$$

$$\sum_{m\in\mathcal{M}'} \delta_{ms}^{\{\omega\}} \le r_s\, z_s,\ \forall s \in \mathcal{S},\ \forall \omega \in \Omega' \tag{15}$$

where $\Omega' \stackrel{\text{def}}{=} \{1, 2, \dots, O'\}$ is the set of O' scenarios independent of the original set Ω. In practice, $O' = 1$ with $p^{\{1\}} = 1$, as only the currently existing scenario of the network is of interest for slicing the resources at that moment. Higher values of O' are useful for simulating multiple scenarios with a homogeneous $\mathcal{M}'$. It is worth noting that Problem 3 is more tractable than Problem 2 as it only contains the single decision variable for resource slicing, simplifying the objective function (3) and constraint (5).

D. Genetic Algorithm Approach

The full sampled DEP formulation is notably intractable as its components increase in size. Most importantly, the accuracy of the sampled DEP is directly dependent on the size of Ω, directly causing a trade off between the accuracy of the sampled DEP and its computability in a reasonable amount of time. In this subsection, we reformulate the problem of joint BS selection for the VWN as a genetic algorithm, circumventing the need to discretize demand or to establish Ω, thereby simplifying the original problem into a more scalable form.

A genetic algorithm is an iterative metaheuristic in which an approximate solution to a given optimization problem is arrived at via a series of progressive generations. Each generation contains a number of candidate solutions, called individuals, each of which is defined by a chromosome. During a given generation, each individual is assessed a fitness heuristic based on its chromosome. Then individuals are selected at random, with more fit individuals being selected with greater probability. Pairs of selected individuals will crossover with probability p_{xov}, a process similar to genetic recombination in biology. The resulting chromosomes then have probability p_{mut} to mutate, altering the chromosome slightly. Once enough new individual chromosomes have been selected and possibly undergone crossover and mutation, this set of new individuals, called children, forms the next generation to repeat the process.

For the genetic algorithm, ρ is not sampled for discrete demand points. Instead, we assume that all demand over the region is allocated to the closest resource. The subset of $\mathcal{S}$, $\mathcal{S}'$, that is selected for a given possible VWN forms a Voronoi tessellation from the point locations of the selected resources. The total demand allocated to a selected resource $s \in \mathcal{S}' \subseteq \mathcal{S}$ is $\iint_{V_s} \rho\left(x,\, y\right)\, dx\, dy$, where V_s is the region bounded by the cell of resource s in the Voronoi tessellation. If the total demand allocated to s exceeds r_s, s is considered to be *overcapacity*. If V_s is not wholly contained within the coverage area of resource s, s is considered to be *overcoverage*.

Let $\mathcal{G} \stackrel{\text{def}}{=} \{1,\, 2,\, \dots,\, G\}$ be the set of generations used in the genetic algorithm and $\mathcal{I}_g \stackrel{\text{def}}{=} \{1,\, 2,\, \dots,\, I\}\,, g \in \mathcal{G}$ be the set of individuals within generation g. Each individual $i \in \mathcal{I}_{g\in\mathcal{G}}$ has a binary chromosome $z^{\{ig\}}$ of length S. $z_s^{\{ig\}}, s \in \mathcal{S}$, denoting each individual bit of the chromosome, is defined as follows:

$$z_s^{\{ig\}} = \begin{cases} 1, & \text{if BS } s \text{ is selected for the VWN for individual } i \text{ in generation } g, \\ 0, & \text{otherwise} \end{cases}$$

The fitness heuristic of each individual chromosome, $z^{\{ig\}}$, is assessed as the reciprocal of the chromosome's cost, which is defined as

$$\text{fitness}\left(z^{\{ig\}}\right) = \frac{1}{\text{cost}\left(z^{\{ig\}}\right)} \tag{16}$$

$$\text{cost}\left(z^{\{ig\}}\right) = \sum_{s\in\mathcal{S}} \left(c_s\, z_s^{\{ig\}} + c_{\mathrm{cov}}\, \mathbb{1}_{\{V_s \not\subseteq R_s\}} + \left(c_{\mathrm{cap}}^{g} - 1\right)\, \max\left(0,\, \iint_{R_s} \rho\left(x,\, y\right)\, dx\, dy - r_s\right)\right) \tag{17}$$

where R_s is the coverage area region of resource $s \in \mathcal{S}$.

The cost function (7) indicates cost increases not only based on the cost of the resources selected, but also with imperfection costs c_{cov} and c_{cap}, the costs of a selected resource being overcoverage or overcapacity, respectively. The overcapacity cost grows with each successive generation. For early generations, this allows for imperfect solutions to temporarily exist to seed later generations and improve diversity to increase the probability of finding a better final approximate solution.

Elitism is used, where the n most fit individuals of a given generation are automatically selected without crossover or mutation to be the first children of the next generation. Selection occurs via the roulette wheel selection method. Every individual i of a given generation g has a probability of being selected given by

$$\frac{\text{fitness}\left(z^{\{ig\}}\right)}{\sum_{i\in\mathcal{I}}\text{fitness}\left(z^{\{ig\}}\right)}$$

When crossover is performed on selected individuals, it is via the uniform crossover method with a mixing ratio of 0.5. That is, if two selected parent individuals crossover, each equivalent bit in the parents will swap with a probability of 50%. It has been suggested that uniform crossover is more exploratory than n-point crossover *(cite)*. Mutation occurs on a bit-by-bit level, with each bit mutating (i.e., flipping) with probability $\frac{1}{S}$. The uniqueness property is then enforced on the resulting children to ensure diversity; if a child chromosome is identical to another child chromosome in the next generation, the child is discarded and a new child generated, ensuring that each individual of any given generation is unique within that generation.

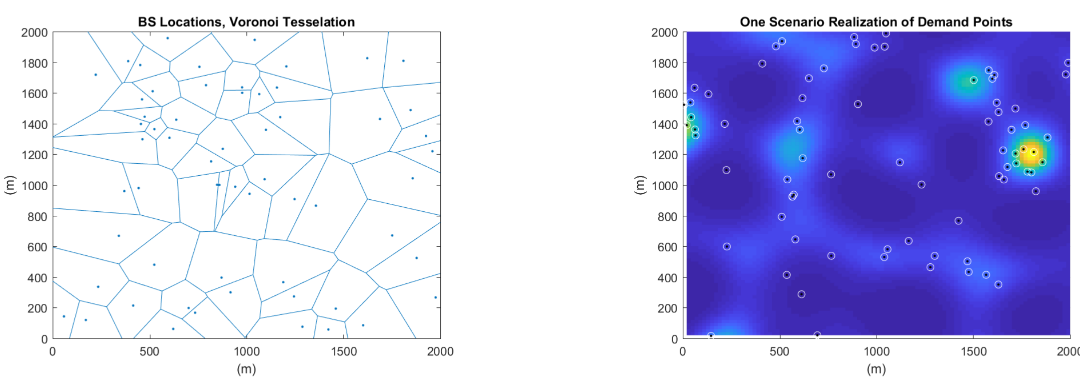
The genetic algorithm iterates for a number of generations G. If the genetic algorithm settles on a single individual for a number of continuous generations, G_{halt}, it will halt and present that individual's chromosome as the final approximate solution for z_s. Otherwise, the chromosome of the fittest individual of generation G determines z_s.

The genetic algorithm only determines an approximate solution to the BS selection forming the VWN, informing the VNB of which BSs to obtain from the RPs. With this selection, z_s, the SP's demand points can be dynamically allocated resource slices as described by Problem 3 in Section III-C.

IV. Simulation and Evaluation

In this section, we evaluate the sampled DEP and genetic algorithm approaches as approximations of Problem 1. We will compare the cost, demand satisfaction, and time to generate of the resultant networks.

A. Setup



(a) RP BS Locations in $\mathcal{S}$ with associated Voronoi tesselation

(b) SP SSLT demand density field with one scenario of SP demand points

Fig. 1: Visualization of network area. (1a) visualizes the resources available to VNB for creating the VWN, and (1b) visualizes the demand to be satisfied by the VWN.

TABLE I: Numerical Values of Relevant Parameters

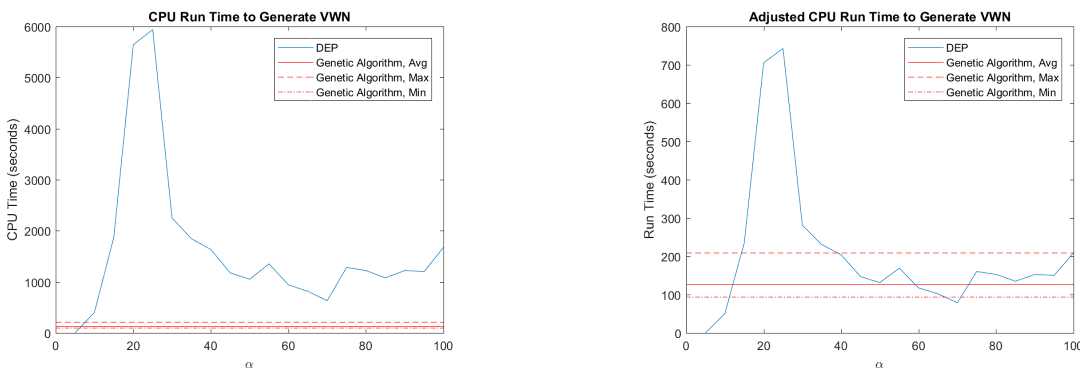
Parameter	**Value**
Width of Geographic Area (X)	2 km
Height of Geographic Area (Y)	2 km
Number of BSs (S)	60
Number of Demand Points (M)	75
Number of Scenarios (O)	25
BS cost ($c_s, \forall s \in \mathcal{S}$)	1
BS capacity ($r_s, \forall s \in \mathcal{S}$)	1.50 Mbps
BS range ($b_s, \forall s \in \mathcal{S}$)	500 m
Demand Point Demand ($d_m, \forall m \in \mathcal{M}$)	0.178 Mbps
Set of Two-Stage Model Weights (α)	{5, 10, ... , 100}
SSLT Approximation Depth (L)	50
SSLT Maximum Angular Frequency ($\omega_{\max}$)	$\frac{2\,\pi}{30}$
SSLT Location Parameter (σ)	0
SSLT Scale Parameter (μ)	1
Pixel Grid Size	100 x 100, 20 m side
Maximum Number of Generations (G)	3000
Minimum Number of Generations	300
Number of Unchanged Generations Before Halt (G_{halt})	150
Number of Individuals per Generation (I)	80
Number of Elite Individuals per Generation	4
Probability of Crossover (p_{xov})	0.7
Probability of Mutation per bit (p_{mut})	$\frac{1}{S} = 0.0167$
Overcoverage Cost (c_{cov})	3
Overcapacity Cost (c_{cap})	1.015

Unless stated otherwise, we use the default parameter values shown in Table I. BS locations are determined as a stationary PPP. Demand point locations are generated independently for each scenario as a non-stationary PPP using $\rho\,(x,\,y)\,,\,x\in[0,\,X],\,y\in[0,\,Y]$, as the spatial intensity function, as described in Section II. Fig. 1 provides a visualization of the simulation network area. (1a) shows the BS locations of $\mathcal{S}$ with the associated Voronoi Tessellation showing the coverage areas of the BSs when all are active with respect to the genetic algorithm. (1b) shows the SSLT demand density field with one example scenario of demand points, which acts as a single sample of the demand density field. To compute $u_{ms}^{\{\omega\}}$, it is assumed that there is perfect propagation between the demand points and BSs. Unlike as described in Section II, $u_{ms}^{\{\omega\}}=1$ if the distance between demand point $m\in\mathcal{M}$ of scenario $\omega\in\Omega$ and BS $s\in\mathcal{S}$ is less than b_s, and 0 otherwise. To compute the integral of the fitness function (7), $\rho\,(x,\,y)$ is discretized into a grid of congruent pixels, and the demands of all pixels within a Voronoi cell of interest are summed together.

We ran our simulations on an Intel Core i7-4790K 4.00 GHz 4 real/8 virtual core CPU with 16 GB of DDR3 RAM. We used CPLEX [11] to solve the sampled DEP optimization problems and we used MATLAB to simulate the genetic algorithm and to generate the demand field and stochastic data (i.e., $\rho\,(x,\,y)$ and

$u_{ms}^{\{\omega\}}$). During the simulations, extraneous processes were culled to allow maximal use of computer resources. Average values for the performance of the genetic algorithm are provided from 50 independent runs using the identical data set except for the set of initial individuals. The sampled DEP solutions were solved across multiple values of α as cost, time, and demand satisfaction are directly dependent on α.

B. Results



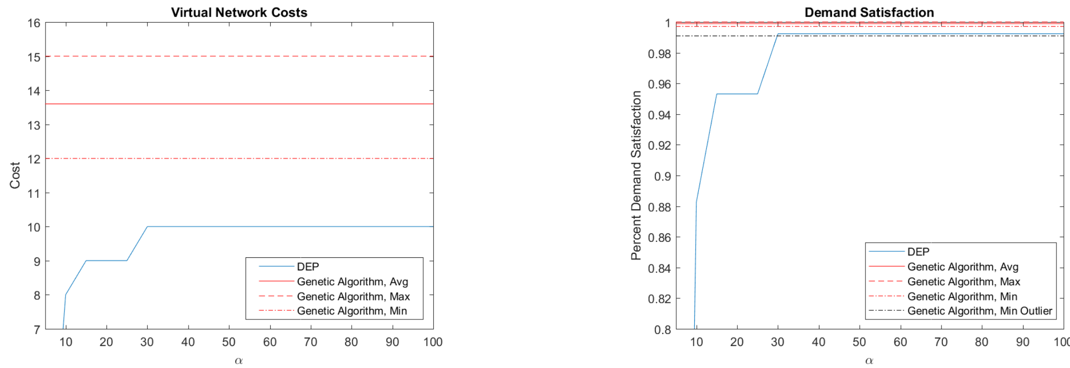
(a) True CPU Time

(b) Adjusted (Pseudo-Real) Run Time

Fig. 2: Run time comparison. The DEP run time is the solid blue line, while the minimum, average, and maximum genetic algorithm run times are provided as dot-dashed, solid, and dashed red reference lines, respectively.

In Fig. 2 is a comparison of the time to run. (2a) shows the absolute overall CPU time taken to converge to a solution in both the DEP and genetic algorithm. (2b) shows an adjusted form of the run time the DEP took to terminate to a single solution. CPLEX is capable of parallelizing across the 8 CPU cores, allowing for the real run time to be, at minimum, one-eighth the CPU run time. With this adjustment providing an improvement to the sampled DEP run time, the genetic algorithm converges in approximately the same time as the sampled DEP for higher levels of α, when a single solution dominates. Without the adjustment, the genetic algorithm outperforms the sampled DEP, converging to a solution in approximately 13% of the time for $\alpha \geq 30$ and in approximately 2% of the time for $\alpha \in \{20, 25\}$.

The trade off for the genetic algorithm's improved run time is that the solution provided is less optimal than the sampled DEP, as indicated by an increased cost for the VNB to build the VWN. Fig. 3a compares the increasing cost of the sampled DEP as α increases with the cost of the various genetic algorithm solutions. On average, the genetic algorithm incurs a 36% increased cost in selecting the BSs for the VWN. At minimum, the incurred cost is only 20% than the sampled DEP, which implies the genetic algorithm might be terminating early, and a tighter solution might be found by increasing G_{halt}. It should also be noted that one unit of cost is one additional BS being selected for the VWN, and the sampled DEP selections for $\alpha \geq 30$ have a cost of only 10 BSs. Any variance that incurs one additional BS for the genetic algorithm incurs 10% increased cost. Increasing the number of BSs required to comprise the VWN would introduce additional granularity



(a) VWN costs

(b) SP demand point satisfaction

Fig. 3: Comparison of VWN costs and SP demand point satisfaction. Note that cost and demand satisfaction show a direct correlation.

in $\mathcal{S}$ that might decrease the inefficiency of the genetic algorithm. This was not done as this data set was chosen specifically so the sampled DEP would terminate within 15 minutes (i.e., in a reasonable amount of time); increasing the number of BSs available to the VNB drastically increases the time it would take the sampled DEP to converge to a solution.

There is a direct correlation between the number of BS in the VWN and its capability for satisfying demand. As the cost increases of the sampled DEP solution, more BS are selected, and the demand satisfaction the sampled DEP solution similarly trends towards 100%, as shown in Fig. 3. Because of this, the genetic algorithm solutions have a very high demand satisfaction, averaging 99.9% demand satisfaction, even reaching 100% for some solutions. The most expensive 10-BS sampled DEP solutions reach 99.2% demand satisfaction when slicing the same set of resources that determined the sampled DEP BS selection.

When the set of demand points change to a scenario no longer in Ω, the sampled DEP performs very similarly. Fig. 4 shows the demand satisfaction for both the sampled DEP and genetic algorithm BS selections when sliced to a new set of scenarios. Here, the number of demand points increases to 200 points per scenario, each with 66.8 kbps rate demand, across 50 independent scenarios. The demand satisfaction trend of the sampled DEP BS selection follows very closely to the original set of scenarios but hits a maximum of 99.0% demand satisfaction with 10 BSs. In comparison, the SP demand point scenarios are far more beneficial to the genetic algorithm, which reaches greater than 99.99% demand satisfaction for all generated VWNs. This is expected as a side effect of the increased number of points and scenarios more accurately describing a sampling of the original SSLT demand density field of the SP, $\rho\,(x,\,y)$.

V. Conclusion

In this work, we studied the problem of determining the optimal joint BS selection for a virtual network and the adaptive slicing of the network to the SP demand points. We introduced a two-stage stochastic optimization problem which minimizes the cost of base station selection for a virtual network while

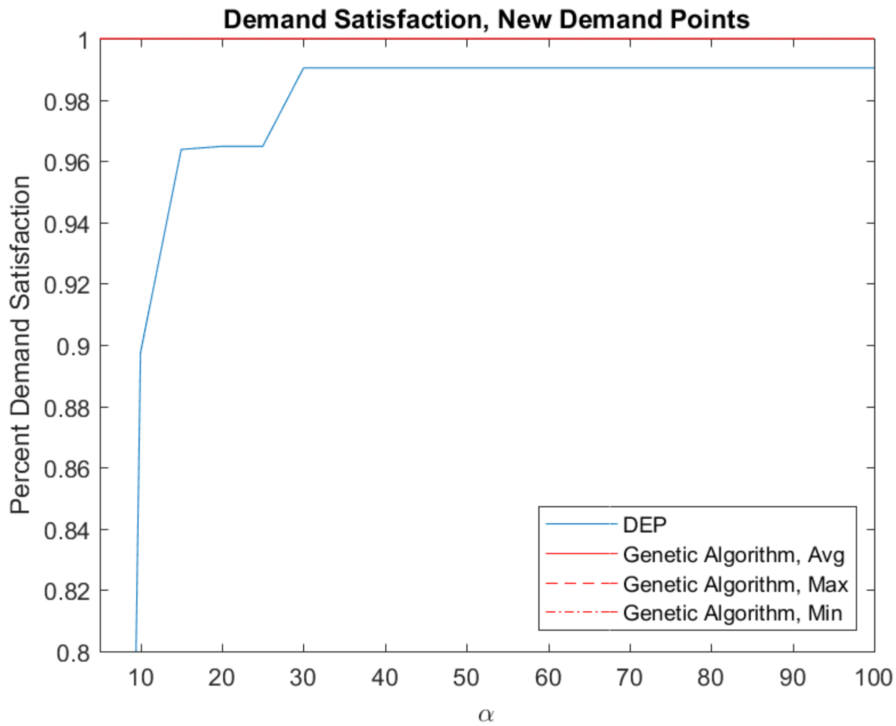


Fig. 4: Comparison of demand satisfaction with 50 scenarios of 200 SP demand points

maximizing demand satisfaction of the network's demand points. We then investigated two approaches that provide a solvable form for the stochastic model: a deterministic equivalent program of the stochastic problem which is simplified to a tractable form with a sampling method, and a genetic algorithm that determines base station selection so that slicing can be found via a single-stage optimization problem.

The sampled DEP is considered to be closer to the original problem, and will provide a tight approximate solution to the original DEP or stochastic model with a sufficiently large set of scenarios and sufficient time, but trends to the intractable as these are reached. The genetic algorithm is a hybrid approach intending to simplify the first stage of the model by using a metaheuristic. All demand, known from a continuous demand density model, within a resource's cell in a Voronoi tessellation is considered allocated to that resource, a simpler process than attempting to allocate individual demand points to the entire pool of resources. The advantage of this approach is that it simplifies the search space of the original two-stage model to a single-stage model which only handles slicing demand points to the selected resources. This provides a faster solution with the introduction of error.

The simulation results indicate that the genetic algorithm approach may be an adequate avenue for a solution. This approach trends to a solution in 2% of the time, incurring 20% increased error. While the error is significant, it could decrease with larger, more computationally-intensive data sets that would provide more resolution to both the resource and demand sets, allowing for less error to be incurred for small variations. It is important to note that increasing the size of the data set incurs a larger time cost on the sampled DEP approach than on the genetic algorithm approach. Analyzing these larger data sets, improving on the genetic algorithm formulation, or utilizing other metaheuristic algorithms could improve on these results and would be obvious avenues for further investigation.

References

[1] C. Beckman and G. Smith, "Shared networks: making wireless communication affordable," *IEEE Wireless Communications*, vol. 12, no. 2, pp. 78–85, April 2005.

[2] J. G. Andrews, S. Buzzi, W. Choi, S. V. Hanly, A. Lozano, A. C. K. Soong, and J. C. Zhang, "What will 5g be?" *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 6, pp. 1065–1082, June 2014.

[3] X. Costa-Perez, J. Swetina, T. Guo, R. Mahindra, and S. Rangarajan, "Radio access network virtualization for future mobile carrier networks," *IEEE Communications Magazine*, vol. 51, no. 7, pp. 27–35, July 2013.

[4] J. S. Panchal, R. D. Yates, and M. M. Buddhikot, "Mobile network resource sharing options: Performance comparisons," *IEEE Transactions on Wireless Communications*, vol. 12, no. 9, pp. 4470–4482, September 2013.

[5] L. Doyle, J. Kibiłda, T. K. Forde, and L. DaSilva, "Spectrum without bounds, networks without borders," *Proceedings of the IEEE*, vol. 102, no. 3, pp. 351–365, March 2014.

[6] U. Gotzner and R. Rathgeber, "Spatial traffic distribution in cellular networks," in *Vehicular Technology Conference, 1998. VTC 98. 48th IEEE*, vol. 3, May 1998, pp. 1994–1998 vol.3.

[7] M. Michalopoulou, J. Riihijärvi, and P. Mähönen, "Towards characterizing primary usage in cellular networks: A traffic-based study," in *2011 IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN)*, May 2011, pp. 652–655.

[8] J. Reades, F. Calabrese, and C. Ratti, "Eigenplaces: analysing cities using the space - time structure of the mobile phone network," *Environment and Planning B: Planning and Design*, vol. 36, pp. 824–836, 2009.

[9] D. Lee, S. Zhou, and Z. Niu, "Spatial modeling of scalable spatially-correlated log-normal distributed traffic inhomogeneity and energy-efficient network planning," in *2013 IEEE Wireless Communications and Networking Conference (WCNC)*, April 2013, pp. 1285–1290.

[10] P. Kall and S. W. Wallace, *Stochastic Programming*. John Wiley and Sons, 1994.

[11] IBM, "Optimization model development toolkit for mathematical and constraint programming (CPLEX)," http://www-03.ibm.com/software/products/en/ibmilogcpleoptistud, 2012.