

Approaches to Joint Base Station Selection and Adaptive Slicing in Virtualized Wireless Networks

Kory A. Teague

Thesis submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Master of Science
in
Electrical Engineering

Allen B. MacKenzie, Chair
Luiz DaSilva
R. Michael Buehrer
Mohammad J. Abdel-Rahman

June 1, 2018 (TBD)
Blacksburg, Virginia

Keywords: TBD

Copyright 2018, Kory A. Teague

Approaches to Joint Base Station Selection and Adaptive Slicing in Virtualized Wireless Networks

Kory A. Teague

(ABSTRACT)

This work received support in part from the National Science Foundation via work involved with the Wireless @ Virginia Tech research group.

Contents

1	Introduction	1
1.1	Literature Review	1
2	VNB Model	3
2.1	Network Area Definitions	3
2.2	Stochastic Optimization	5
3	Approximation Approaches	8
3.1	Deterministic Equivalent Program	8
3.1.1	Sampling Approaches	10
3.1.2	Adaptive Slicing	11
3.2	Genetic Algorithm	12
4	Testing and Simulations	16
4.1	VWN Construction for a Single SP	16
4.1.1	Case I: Homogeneous Urban Cellular Network	16

4.1.2	Case II: Impact of Heterogeneous Resources	17
4.2	VWN Construction for Multiple SPs	17
4.2.1	Case III: Two Similar Urban Cellular Networks	17
4.2.2	Case IV: SPs with Specialized Demands	18
5	Conclusions	19

List of Figures

List of Tables

Chapter 1

Introduction

Placeholder to make bibTeX happy: [1]

Here I introduce the problem I have been working on (BS selection and dynamic slicing within a VWN). Start the paragraph talking about the increasing demands within the cellular network; IoT devices, trends toward massive demands. Mention how costs to maintain and increase the capacity of the network with current costs are rising. Lead into necessary needs for 5G as intended to be implemented over the next two years, and how virtualization is a potential solution to increase capacity while minimizing cost within 5G networks.

1.1 Literature Review

Discuss the previously existing literature that I have been reading. Discuss wireless cellular networks, virtualization and their use in wireless networks. Likely work discussing the network without borders paradigm that is at the core of the work. Discuss optimization and stochastic optimization. Discuss metaheuristic algorithms, especially genetic algorithms, and

perhaps how they have been used for and to simplify optimization.

Chapter 2

VNB Model

Begin breaking down the model used as the basis for the work. Start with a lead in, then start defining the model in the first subsection. Expound on some of the definitions and descriptions that I moved past in the conference paper (perhaps describing the SSLT model more fully). Make sure that the definitions and work are generalized for writing about a constructed VWN (or constructed VWNs, depending on the terminology I intend to use) for multiple RPs and SPs.

2.1 Network Area Definitions

Lay out the how the network and area is structured here. Define the SSLT demand field. This is effectively section 2 of my previous conference paper.

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

coverage radius 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 [2, 3]. It has also been shown that traffic distribution is spatially correlated [3, 4]. 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 [5].

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

$$\rho^G(x, y) = \frac{1}{L} \sum_{l=1}^L \cos(i_l x + \phi_l) \cos(j_l y + \psi_l) \quad (2.1)$$

where $\mathcal{L} \stackrel{\text{def}}{=} \{1, 2, \dots, L\}$ 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 ω_{\max} according to the central limit theorem.

The approximate Gaussian distribution ρ^G is then normalized to 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

$$\rho(x, y) = \exp\left(\frac{\sigma}{\sqrt{\text{Var}(\rho^G)}} \rho^G(x, y) + \mu\right) \quad (2.2)$$

where $\text{Var}(\rho^G)$ is the variance of ρ^G .

$\rho(x, y)$ 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 [5]. 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 $\mathcal{M} \stackrel{\text{def}}{=} \{1, 2, \dots, M\}$ 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 the points in \mathcal{M} , and therefore the values of u_{ms} , is determined via a non-stationary 2D Poisson point process (PPP) with M points using the demand field, ρ , as the spatial intensity function. To generate this non-stationary PPP, we use an acceptance-rejection method. Each point of a stationary PPP with an intensity of $\rho_{\max} = \max_i \rho(x_i, y_i)$ is retained with probability $\frac{\rho(x_i, y_i)}{\rho_{\max}}$, where x_i and y_i are the x- and y-coordinates of the i^{th} point of the stationary PPP.

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 (\sim) placed above the symbol.

2.2 Stochastic Optimization

Now that the model, area, and definitions are defined, proceed to use those terms and define the overall generalized stochastic optimization problem. This is effectively section 3 (a) of my conference paper, but with, perhaps, a little more focus.

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

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

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)

$$\underset{\{z_s, s \in \mathcal{S}\}}{\text{minimize}} \left\{ \sum_{s \in \mathcal{S}} c_s z_s + \alpha \mathbb{E} [h(z, u)] \right\} \quad (2.3)$$

subject to:

$$z_s \in \{0, 1\}, \forall s \in \mathcal{S} \quad (2.4)$$

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

$$\underset{\{\delta_{ms}, m \in \mathcal{M}, s \in \mathcal{S}\}}{\text{minimize}} \left\{ - \sum_{m \in \mathcal{M}} \sum_{s \in \mathcal{S}} \delta_{ms} \tilde{u}_{ms} \right\} \quad (2.5)$$

subject to:

$$z_s = \mathbb{1}_{\{\sum_{m \in \mathcal{M}} \delta_{ms} > 0\}}, \forall s \in \mathcal{S} \quad (2.6)$$

$$\sum_{s \in \mathcal{S}} \delta_{ms} \tilde{u}_{ms} \leq d_m, \forall m \in \mathcal{M} \quad (2.7)$$

$$\sum_{m \in \mathcal{M}} \delta_{ms} \leq r_s, \forall s \in \mathcal{S}. \quad (2.8)$$

The first stage objective function (2.3) minimizes the total cost of the selected network

with respect to that network's ability to satisfy the demand contained within the region. The second stage objective function (2.5) 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.

Constraints (2.4), (2.6), and (2.8) implement the defined ranges and values of the decision variables z_s and δ_{ms} , with (2.6) ensuring that demand is allocated only to selected resources. For constraint (2.6), $\mathbb{1}_{\{*\}}$ is defined by

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

Constraint (2.7) ensures a demand point $m \in \mathcal{M}$ is not allocated more resources than it demands.

Chapter 3

Approximation Approaches

In this chapter, I work on defining the approximation approaches used in my work. Lead in to discussing the need to approximate the stochastic optimization problem from section 2.2 to adequately solve my work, then introduce the two approaches I used to approximate the optimization problem: the DEP/its sampling/generalized post-selection slicing and the genetic algorithm as a selection method.

3.1 Deterministic Equivalent Program

Introduce the idea of a DEP as an approach for solving the original stochastic problem. Present the solved problem here in the form of the true deterministic equivalent program - as in, it is actually an equivalent to the original stochastic problem - with all the necessary expansions and additional variables. Focus on how this formulation no longer includes any stochastic variables and is purely deterministic. Mention that the trade off is that the deterministic variables are part of a infinitely large set of potential scenarios.

In order to solve the two-stage stochastic optimization formulation (Problem 1), we need to convert it to a deterministic equivalent program (DEP) that does not contain any stochastic variables (only deterministic variables) [6].

Let Ω be defined as the sample space, i.e., the set of all scenarios. Let $\hat{\Omega} \stackrel{\text{def}}{=} \{1, 2, \dots, O\}$ be a discrete set containing sampled scenarios. The probability a given scenario $\omega \in \hat{\Omega}$ occurs is denoted by $p^{(\omega)}$, $\omega \in \hat{\Omega}$, where $\sum_{\omega \in \hat{\Omega}} p^{(\omega)} = 1$. Variables that are dependent on the scenario are shown with a superscript (ω) with the specific scenario it is dependent on indicated by ω .

Problem 2 (Deterministic Equivalent Program of Problem 1)

$$\begin{aligned} & \underset{\substack{z_s, \delta_{ms}^{(\omega)} \\ s \in \mathcal{S}, m \in \mathcal{M}, \\ \omega \in \hat{\Omega}}}{\text{minimize}} \left\{ \sum_{s \in \mathcal{S}} c_s z_s - \right. \\ & \left. \alpha \sum_{\omega \in \hat{\Omega}} p^{(\omega)} \left(\sum_{m \in \mathcal{M}} \sum_{s \in \mathcal{S}} \delta_{ms}^{(\omega)} u_{ms}^{(\omega)} \right) \right\} \end{aligned} \quad (3.1)$$

subject to:

$$\sum_{s \in \mathcal{S}} \delta_{ms}^{(\omega)} u_{ms}^{(\omega)} \leq d_m, \forall m \in \mathcal{M}, \forall \omega \in \hat{\Omega} \quad (3.2)$$

$$\sum_{m \in \mathcal{M}} \delta_{ms}^{(\omega)} \leq r_s z_s, \forall s \in \mathcal{S}, \forall \omega \in \hat{\Omega} \quad (3.3)$$

$$z_s \in \{0, 1\}, \forall s \in \mathcal{S}. \quad (3.4)$$

The objective function (3.1) combines both objective functions (2.3) and (2.5) of the initial formulation into a deterministic form. Constraints (3.2) and (3.3) ensure demand is not overallocated and is only allocated to selected resources and within capacity for all scenarios.

Problem 2 provides an equivalent deterministic form of Problem 1 for the (finite) sampled

state space, $\hat{\Omega}$, containing O scenarios. With sufficiently large O , $\hat{\Omega}$ approaches a tight approximation of the original sample space. Within each scenario $\omega \in \hat{\Omega}$, 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 PPP with M points as described in Section ??.

3.1.1 Sampling Approaches

As the infinitely large set of scenarios renders the problem unable to be solved, it needs to be sampled into a finite set to be solved. Present the structure and nomenclature used to imply a sampled set of scenarios, and describe the structure of how the scenarios are sampled into a truncated set. Might be worth mentioning that there are other methods that might be better for sampling beyond the completely random sampling approach I am using. Worth consideration?

Sample Average Approximation

At what point is the sampling enough? As the set of scenarios considered within the sampled DEP increases, it more closely compares to the original DEP and the stochastic optimization problem, but it also becomes increasingly difficult to solve as the number of scenarios considered increases. So, it is beneficial to understand that a certain known number of scenarios provides a reasonably tight - what does reasonable mean? - solution to the original DEP to avoid being unnecessarily computationally expensive to solve. Finding this minimum necessary number of scenarios can be done via a sample average approximation (SAA) analysis, which should not be too complicated to do.

3.1.2 Adaptive Slicing

Now that we have a (close) approximation to the DEP and the original stochastic optimization problem, we have a method for deriving the minimum cost BS selection and adaptive slicing for the desired VWN. However, this selection is overly time consuming to constantly run, and the BSs selected for the VWN(s) by the VNB are fairly constant, so all that is needed is to dynamically (read: adaptively) slice the selected BSs to the various SPs. To do this, we simplify the sampled DEP such that it has only one scenario - ostensibly, the current scenario in time - and the BSs selected set to be a constant rather than a decision variable. The resulting problem is a single stage linear program that is much simpler to solve. This is used to adaptively slice resources to the demand.

After the solution to the sampled DEP of Section ?? has been found, the VNB has determined the joint BS selection that forms the VWN and a proposed resource slicing of considered possible scenarios, $\hat{\Omega}$, that allocates the resources to the SP's demand points. Since O is not infinite, any given scenario present in the formed VWN is unlikely to be an element of $\hat{\Omega}$. Further, as demand points move between BSs or enter or exit the VWN, a new scenario $\omega \notin \hat{\Omega}$ is formed. 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)

$$\underset{\{\delta_{ms}, s \in \mathcal{S}, m \in \mathcal{M}\}}{\text{maximize}} \left\{ \sum_{m \in \mathcal{M}} \sum_{s \in \mathcal{S}} \delta_{ms} u_{ms} \right\} \quad (3.5)$$

subject to:

$$\sum_{s \in \mathcal{S}} \delta_{ms} u_{ms} \leq d_m, \forall m \in \mathcal{M} \quad (3.6)$$

$$\sum_{m \in \mathcal{M}} \delta_{ms} \leq r_s z_s, \forall s \in \mathcal{S}. \quad (3.7)$$

It is worth noting that Problem 3 is more tractable than Problem 2 as it only contains the single continuous decision variable for resource slicing, simplifying the objective function (3.5) and constraint (3.7) from a mixed integer linear program to a linear programming problem.

3.2 Genetic Algorithm

Now that the first approach - DEP and its sampling - has been tackled, and the necessary tool to evaluate it has been derived from it - the simplified adaptive slicing program - move on to the genetic algorithm approach for approximating the BS selection process. Discuss the core algorithm of a genetic algorithm, then the various approaches that I used in its process (e.g., binary chromosomes, elitism, uniqueness, uniform crossover, bitwise mutation).

The Problem 2 formulation becomes intractable as O , S , or M increases. Most importantly, the accuracy of the sampled DEP is directly dependent on the size of $\hat{\Omega}$, O , 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 $\hat{\Omega}$, 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, a fitness heuristic is assessed for each individual based on its chromosome. Then individuals are selected at random, with more fit individuals being selected with higher probability. Pairs of selected individuals will crossover with probability p_{cov} , 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 $\iint_{V_s} \rho(x, y) dx dy$, 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:

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

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

$$\text{fitness} (z^{\{ig\}}) = \frac{1}{\text{cost} (z^{\{ig\}})} \quad (3.8)$$

$$\begin{aligned} \text{cost} (z^{\{ig\}}) = \sum_{s \in \mathcal{S}} & \left(c_s z_s^{\{ig\}} + c_{\text{cov}} \mathbb{1}_{\{V_s \not\subseteq R_s\}} + \right. \\ & \left. (c_{\text{cap}}^g - 1) \max \left(0, \iint_{R_s} \rho(x, y) dx dy - r_s \right) \right) \end{aligned} \quad (3.9)$$

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

The cost function (3.9) 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

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

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%. 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 3.1.2.

Chapter 4

Testing and Simulations

In this chapter I will be introducing four different cases to test the provided approximation approaches. The first will be the test case used in my conference paper (one SP, with homogeneous resources). The second will be an expansion of the test case used in my conference paper, but with heterogeneous resources. The third will extend to service multiple similar cellular SPs. The fourth will extend to a case with multiple SPs with various, specialized demands.

4.1 VWN Construction for a Single SP

Lead into the first two cases, which test the approaches while using a single SP.

4.1.1 Case I: Homogeneous Urban Cellular Network

Basically as presented in my conference paper. One SP, homogeneous resources within the RPs. Might need to use a new data set, though, with a larger data set.

4.1.2 Case II: Impact of Heterogeneous Resources

Same as Case I, but with heterogeneous resources within the RPs. Need to understand how this changes the approaches.

4.2 VWN Construction for Multiple SPs

Lead into the second two cases (should I have more?), which test using multiple SPs to satisfy from the same set of resources.

4.2.1 Case III: Two Similar Urban Cellular Networks

First consider a case with two SPs with similar demands. Overlapping cellular networks. Could see how the approaches behave while two SPs partially overlap.

Homogeneous Resources

If it appears that the difference between Case I and Case II (sections 4.1.1 and 4.1.2) is worth further consideration, then analyze here with homogeneous resources. Otherwise, a single comparison should be sufficient.

Heterogeneous Resources

As for the previous subsubsection (4.2.1), but consider with heterogeneous resources.

4.2.2 Case IV: SPs with Specialized Demands

This is the major case that is the extension of my work. Case I (4.1.1) analyzed what happens with a single SP, Case II (4.1.2) expanded that to heterogeneous resources, and Case III (4.2.1) added an additional similar SP, but Case IV considers when there are several SPs and with their own considerations and unique demands. Need to consider what these SPs look like. One would be a cellular network like in Case I (moderate to high number of users, moderate demand). Another could be a streaming service (few users, high individual demand). Another an emergency service (very low number of users and demand, but requiring virtually 100% demand satisfaction - see note below). What other SPs should I consider?

Note: *I need to consider how to accurately label demand satisfaction within the approaches. In effect, this would be controlled by α for the (sampled) DEP and controlled by β or some such for the genetic algorithm. I should investigate this at some point of the thesis, probably within their appropriate sections in chapter 3 (DEP: 3.1 and GA: 3.2).*

Homogeneous Resources

As for Case III (4.2.1), if a considerable difference was detected between Cases I and II (4.1.1 and 4.1.2), consider analyzing the case with homogeneous resources and

Heterogeneous Resources

also with heterogeneous resources.

Chapter 5

Conclusions

Consider conclusions of my work. I don't think this chapter would be long, but condense my findings into some coherent thoughts, and redirect to what they are. Also expound on some of the further work that my research could be expanded to (e.g., further use cases investigating my approaches, use of (meta)heuristics other than a genetic algorithm to approximate the optimization problem, improve the basic capacity function used in my optimization model).

Bibliography

- [1] C. Beckman and G. Smith, “Shared networks: making wireless communication affordable,” *IEEE Wireless Communications Magazine*, vol. 12, no. 2, pp. 78–85, April 2005.
- [2] U. Gotzner and R. Rathgeber, “Spatial traffic distribution in cellular networks,” in *Proceedings of the IEEE Vehicular Technology Conference (VTC)*, May 1998, pp. 1994–1998, vol. 3.
- [3] M. Michalopoulou, J. Riihijarvi, and P. Mhnen, “Towards characterizing primary usage in cellular networks: A traffic-based study,” in *Proceedings of the IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN)*, May 2011, pp. 652–655.
- [4] 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.
- [5] D. Lee, S. Zhou, and Z. Niu, “Spatial modeling of scalable spatially-correlated log-normal distributed traffic inhomogeneity and energy-efficient network planning,” in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC)*, April 2013, pp. 1285–1290.
- [6] P. Kall and S. W. Wallace, *Stochastic Programming*. John Wiley and Sons, 1994.