Research Article

Model-Driven Approach to Fading-Aware Wireless Network Planning Leveraging Multiobjective Optimization and Deep Learning

Dragana Krsti´c1, Nenad Petrovi´c1, and Issam Al-Azzoni2

1University of Niˇs, Faculty of Electronic Engineering, Niˇs, Serbia
2Department of Software Engineering and Computer Science, Al Ain University, Al Ain, UAE

Correspondence should be addressed to Dragana Krsti´c; dragana.krstic@elfak.ni.ac.rs

Received 19 November 2021; Revised 6 February 2022; Accepted 3 March 2022; Published 8 April 2022

Copyright © 2022 Dragana Krsti´c et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Efficient resource planning is recognized as one of the key enablers making the large-scale deployment of next-generation wireless networks available for mass usage. Modelling, planning, and software simulation tools reduce both the time needed and costs of their tuning and realization. In this paper, we propose a model-driven framework for proactive network planning relying on synergy of deep learning and multiobjective optimization. The predictions about service demand and energy consumption are taken into account. Also, the impact of degradations resulting from fading and cochannel interference (CCI) effects is also considered. The optimization task is treated as a component allocation problem (CAP) aiming to find the best possible base station allocation for the considered smart city locations with respect to performance and service demand constraints. The goal is to maximize Quality of Service (QoS) while keeping the costs and energy consumption as low as possible. The adoption of a model-driven approach in combination with model-to-model transformations and automated code generation does not only reduce the complexity, making experimentation more rapid and convenient at the same time, but also increase the overall reusability and expandability of the planning tool. According to the obtained results, the proposed solution seems to be promising not only due to achieved benefits but also regarding the execution time, which is shorter than that achieved in our previous works, especially for larger distances. Further, we adopt model-based representation of handover strategies within the planning tool, enabling examination of the dynamic behavior of user-created plan, which is not exploited in other similar works. The main contributions of the paper are (1) wireless network planning (WNP) metamodel, a modelling notation for network plans; (2) model-to-model transformation for conversion of WNP to generalized CAP metamodel; (3) prediction problem (PP) metamodel, high-level abstraction for representation of prediction-related regression and classification problems; (4) code generator that creates PyTorch neural network from PP representation; (5) service demand and energy consumption prediction modules performing regression; (6) multiobjective optimization model for base station allocation; (7) Handover Strategy (HS) metamodel used for description of dynamic aspects and adaptability relevant to network planning.

1. Introduction

The next-generation wireless communication systems that will radically change society are beyond fifth-generation (B5G) networks or called the sixth-generation mobile communication (6G) networks [1]. Until now, many research efforts have been made on 5G and B5G networks, but still many areas are open for investigation. Since the desire to go towards 5G has been growing, a considerable number of papers have been made and extensive research activities are going on in global wise [1–3]. These papers tend to highlight use cases, requirements, and performance as well as some of the important enabling technologies of next-generation wireless networks.

Wireless communication services have become an integral part of modern life. As a result, the number of different smart devices in the network is constantly increasing. On the other hand, the range available for use by different
applications and users is limited. In addition, the development of the Internet of Things (IoT) service leads to the need for multiple connections between facilities and people. However, existing communication systems have strict limitations regarding the conditions that need to be realized.

Various wireless transmission techniques are constantly being developed to be integrated into a new generation of wireless communication systems. They should meet a large number of set requirements, such as high spectral efficiency, bandwidth, larger network capacity, higher energy efficiency, as little delay as possible, and also high data rates with high reliability, that is, less errors and cost as possible [1–3]. The constant growth of the need to serve an increasing number of users in synergy with the commercial needs of the market has led to the demand for permanent development of new and improvement of existing information and communication services and technologies, both civilian and military. The emergence of new generations of wireless communication systems is conditioned by the need to provide support for modern communication services. As a result, wireless radio frequency (RF) transmission moves to the microwave range. The goal is to achieve a higher bandwidth of the useful signal, higher transmission speed, and higher signal quality at reception [1–3]. Wireless optical transmission has also proven to be an effective means of achieving these goals. This way, the disadvantages of complicated and expensive implementation of transmission systems based on optical fibers are avoided, which represent a good choice for the realization of a communication link over longer distances. That is why it is necessary for wireless network deployments to execute a comprehensive analysis of wireless signal transmission characteristics of wireless network deployments in the presence of fading and cochannel interference (CCI) [4, 5]. The goal is to determine the optimal signal reception scenario when diversity techniques are applied, as well as determining the optimal values of the transmission parameters.

However, in order to provide service with such characteristics for a large number of users, huge efforts and investments are required due to the fact that next-generation wireless networks rely on various equipment and resources that have to be installed and set up for the desired scenarios of usage. Therefore, network resource planning together with simulation is of utmost importance for implementation and innovation in the context of next-generation wireless systems [6]. Such an approach gives the ability to analyze different aspects proactively and perform experiments with many alternative deployments, providing the freedom of parameter tuning, which potentially reduces the costs and required time [6]. On the other side, different aspects in wireless networks could be in conflict, such as Quality of Service (QoS) maximization against the cost and energy consumption minimization. Thus, in this paper, the multiobjective optimization approach to next-generation WNP is adopted in synergy with deep learning-based predictions. This way, it is enabled to make future network deployment-related plans proactively, based on acquired historical data, rather than waiting for present information to be collected, which is the key advantage of such an approach, which has been already approved in various domains [7, 8]. We leverage model-driven software engineering with the aim of covering dynamic aspects of network planning, automating certain steps, and increasing solutions’ expandability and reusability.

The rest of this paper is structured as follows. In the Background section, the underlying theory and concepts behind the proposed approach and its implementation are presented, fading in telecommunications together with the methods for its mitigation, system performance measures under fading influence, model-driven software engineering, deep learning, and multiobjective optimization. The second section after Introduction considers the relevant related works in this area, together with a discussion of the origin of the proposed approach. The next section provides a detailed overview of implementation from the perspective of the following aspects: component and workflow overview, WNP metamodel, model-to-model transformation from planning metamodel to component allocation in general, deep learning-based predictions, multiobjective optimization model for optimal base station allocation at given smart city locations, and model-drive handover strategy representation. After that, Experiments and Evaluation section gives an overview of various executed scenarios and the achieved results, considering both the execution time and value of network planning benefit (cost reduction and performance improvement). Discussion section is added to highlight the obtained results and advantages of the proposed model. In the end, the Conclusion summarizes the main contribution of the paper and proposes the possible future works and research directions on this topic.

2. Background

2.1. Description of the Impact of Fading. During wireless signal transmission, signal amplitude and phase values vary over time [4]. Phase variations, which occur as a result of the action of different interferences, in incoherent modulation schemes do not have any significant influence on the characteristics of the receiving signal. In this case, the signal phase information is not considered in the detection. However, in a coherent modulation scheme, the desired information is contained in the signal phase, so signal phase variations can seriously affect the reception quality.

On the other hand, multipath fading is caused due to the propagation of the signal through the wireless medium, the refraction of the signal in the ionosphere, and the refraction of the signal from various objects on the propagation path. Such reflected, scattered, and diffracted components of the same signal, with random delay times, are superimposed in a constructive or destructive manner in the resulting signal. The effect of multipath fading is manifested through short-term fluctuations in signal intensity, and its influence on the change in the envelope of the received signal can be statistically modelled using various models [4]. Precise characterization of this phenomenon is often impossible or extremely complicated, so the mathematical characterization of the system is a very complex task. However, significant efforts are being made to determine as simple,
accurate statistical models as possible to describe different types of propagation environments.

The paper [5] presents a simple but effective path loss channel model (fading model), appropriate for 5G millimeter-wave (mm-Wave) propagation in both indoor and outdoor environment. In [9], improvements for the determination of base station heights and multipath amplitude scaling are involved. The goal is to increase model fitting and efficiency in meeting the requirements of complex 5G mm-Wave. Simulated results showed that the large-scale and small-scale fading are consistent with measurements from available literature [5].

To estimate any real wireless communication system, it is necessary to have a fading channel model that contains and describes its main features. The most intensively used distributions for describing complex characteristics and random nature of fading envelope are Rayleigh, Rician or Nakagami-\(n\), Nakagami-\(m\), Nakagami-\(q\) (Hoyt), Weibull distribution, and so on [10, 11]. These well-known distributions are still the basis of the majority of theoretical researches of wireless technologies, even the new ones, such as mm-Wave communications.

The simple analytical form of Nakagami-\(m\) distribution allows obtaining wireless system performance measures in closed forms [12]. This fading model includes some other distributions by putting special values for parameter \(m\). Namely, Nakagami-\(m\) fading distribution can be transformed into Rician fading distribution, by setting the parameter \(m\) versus the Rician \(k\) factor, as \(m = (1 + k)2/(1 + 2k)\) [13]. Likewise, for \(m = 1\), the Rayleigh distribution is obtained. Rician distribution is used for multipath environments where a direct path or a line of sight (LOS) between the transmitter and receiver exists.

Besides, the wireless fading channels can be described by a significant number of general models, which are gaining in importance more and more. Their flexibility makes them suitable for situations in which none of the traditional distributions give a good fit [14–25]. In addition, their convenience is recognized in practical and realistic environments. Field measurements made in different propagation conditions have shown that these models mostly better adapt the statistical variations of the propagated information signal. Because of that, it is of the utmost importance to develop and improve the methods for simulating, describing, and computing the performance of general fading models and their special cases for every parameter value.

Recently, some general distributions describing the fading effects in wireless channels are increasingly investigated in the literature in order to better describe the conditions in wireless telecommunications channels and then determine their performance. Among them are \(\alpha-\mu\), \(\kappa-\mu\) [14], \(\eta-\mu\), \(\lambda-\mu\) [15], \(\alpha-\kappa-\mu\) [18, 19], \(\alpha-\eta-\mu\) [19], \(\alpha-\eta-\kappa-\mu\) [20], \(\alpha-\lambda-\mu\) [21], \(\eta-\lambda-\mu\) [22], \(\alpha-\eta-\kappa-F\) [23], \(\alpha-\eta-F\) [24], \(\alpha-\kappa-F\) [24], and \(\alpha-F\) [25] distributions and many others. Here, the fading parameters are as follows: \(\alpha\) represents nonlinearity of propagation environment, \(\mu\) represents the number of the multipath clusters in propagation environment, \(\kappa\) is the Rician factor, \(\eta\) accounts for the unequal power of the in-phase and quadrature components of the fading signal, \(\lambda\) indicates the correlation coefficient between the scattered-wave in-phase and quadrature components in the cluster, and so on.

One of the mentioned general distributions is \(\alpha-\kappa-\mu\) distribution [18]. The \(\alpha-\kappa-\mu\) nonlinear generalized multipath model is characterized by three parameters, which are able to provide good fitting to experimental data obtained from real communication environments. The generality of \(\alpha-\kappa-\mu\) fading model is visible from [18], Table 1. Other fading models, such as \(\alpha-\mu\), \(\kappa-\mu\), Nakagami-\(m\), and Rayleigh, can be obtained from it for different values of parameters \(\alpha\), \(\kappa\), and \(\mu\). By putting \(\alpha = 2\), the \(\alpha-\kappa-\mu\) distribution reduces to \(\kappa-\mu\) distribution; \(\alpha-\mu\) distribution can be obtained from \(\alpha-\kappa-\mu\) distribution if \(\kappa = 0\). By setting \(\alpha = 2\) and \(\kappa = 0\), the \(\alpha-\kappa-\mu\) distribution becomes Nakagami-\(m\) distribution.

Similar is valid for other general distributions, such as \(\alpha-\eta-\mu\), where different values of parameters give other known fading distributions ([19], Table 1). The \(\alpha-\eta-\mu\) distribution is the generalized multipath fading model that encompasses both the nonlinearity and the inhomogeneity of the propagation medium. The \(\alpha-\eta-\mu\) distribution includes, as special cases, other short-term fading distributions, such as Rayleigh, Nakagami-\(m\), Nakagami-\(q\) (Hoyt), \(\eta-\mu\), Weibull, and one-side Gaussian distribution. By putting the parameter \(\alpha\) to the value \(\alpha = 2\), \(\alpha-\eta-\mu\) distribution reduces to \(\eta-\mu\) distribution. Further, from the \(\eta-\mu\) fading distribution, the Nakagami-\(m\) distribution can be obtained in two cases: first, for \(\eta \longrightarrow 1\), the parameter \(m\) will be expressed as \(m = \mu/2\), and second, for \(\eta \longrightarrow 0\), the parameter \(m\) will be given as \(m = \mu\). It is known that the \(\eta-\mu\) distribution reduces to the Hoyt distribution, for the case when \(\mu\) is set to be \(\mu = 1\), with parameter \(b\) given as \(b = (1 - \eta)/(1 + \eta)\). If the in-phase and quadrature component variances are equaled, that is, by setting \(\eta = 1\), the Rayleigh distribution will be obtained from the Hoyt distribution. Next, the Weibull distribution could be derived as a special case of the \(\alpha-\eta-\mu\) distribution by setting corresponding values to the parameters \(\eta\) and \(\mu\), so that \(\eta = 1\) and \(\mu = 1\).

One more general small-scale fading model for wireless communications is the \(\alpha-\lambda-\mu\) distribution model introduced in [21]. This distribution includes the nonlinearity and the inhomogeneous nature of the propagation environment and is validated through field measurements. This distribution shows many conveniences, such as its generality and mathematical tractability, because of its expression in closed form. The \(\alpha-\lambda-\mu\) model can be also called \(\alpha-\eta-\mu\) format 2 model, arising from the \(\alpha-\eta-\mu\) model, format 1, by a rotation of the axes. This model unites known general distributions: \(\alpha-\mu\), \(\eta-\mu\), \(\lambda-\mu\) (in fact \(\eta-\mu\) format 2), and other which are included in them as their special cases.

Every analysis and performance derivation under the influence of fading with general distribution has the main contribution to the above-mentioned generality.

Besides, by short-term (multipath) fading, the wireless channels are affected by long-term fading (shadowing). The long-term shadowing is caused by obstacles such as buildings, trees, and hills. The consequence of shadowing is that the mean power of the short-term faded signal becomes
random. For describing shadowing, log-normal distribution was used in wireless systems [26]. Later, Gamma distribution started to be used to describe shadowing influence because it enables obtaining system performance in closed forms [27]. Similar, in addition to the log-normal and Gamma distributions, the inverse Gamma [28, 29] and inverse Gaussian [30–32] distributions can also be used to describe accurate characteristics of shadowing and offer analytical tractability.

According to that, Gamma distribution is widely used in the available literature to model shadowing in order to analyze the performance of composite fading channels because it better represents shadowing than log-normal distribution. But, since recently, a new tendency has been noticed, and in [32, 33], more recent findings have been published. Namely, it was shown in [31] that the inverse Gaussian (IG) distribution is more close to log-normal distribution than the Gamma distribution. After that, in order to prove this fact, some papers [34, 35] took IG to represent shadowing in composite fading models, and [32, 33] processed new applications with more general multipath fading distributions. In reality, multipath fading and shadowing occur simultaneously. Hence, composite distributions arise as more accurate statistical models of wireless communication channels. Older composite fading channels included Rayleigh-Lognormal (RL), Nakagami-Lognormal (NL), and Hoyt-Lognormal (HL) models [4], but nowadays, there are many new ones.

The performance of wireless communication systems in the presence of small- and large-scale fading and their composite fading has been thoroughly investigated in the past decade. So, the RL model, called the Suzuki model [36], consisting of the log-normal shadowing and the Rayleigh multipath fading, has been detailed and employed to investigate the performance of various communication schemes [4].

In the following, it is important to consider the influence and distribution of composite fading caused due to the simultaneous influence of fast (multipath) fading and slow fading (shadowing) [14, 37–41]. In [39], the K fading model is used to describe composite multipath/shadowing fading channels. This model is more accurate for real propagation environments. In [42], the fluctuating multiple-ray (FMR) and κ-µ shadowed models show better fit with measurements in outdoor environments at 28 GHz than previously introduced channel models. Therefore, it is shown again that generalized models are feasible alternatives for usage as a measure for evaluating communication performance in mm-Wave scenarios.

In [14], a model of κ-µ/α-µ for two independent non-identically distributed (i.i.d) random variables (RVs), which describes the effects of LOS short-term fading with κ-µ distribution and long-term fading with α-µ distribution, is given. This model contains special cases: Rice/α-µ, Rayleigh/α-µ, Nakagami-m/α-µ, and one-sided Gaussian/α-µ. Since α-µ distribution consists of Nakagami-m, Rayleigh, Weibull, one-sided Gaussian, negative exponential, and Gamma distribution as special cases, new special cases of composite fading will be obtained: Rice/Rayleigh, Rayleigh/Rayleigh, Rayleigh/one-sided Gaussian, and so on.

The α-κ-µ-shadowed (α-KMS) fading distribution is a generalization of the α-κ-µ and α-η-µ distributions. For this distribution in [43], the outage probability (Pout) and average channel capacity (CC) are discussed. The α-KMS fading distribution unites a wide set of fading distributions, as the α-κ-µ, α-η-µ, α-µ, Weibull, κ-µ shadowed, Rician shadowed, κ-µ, and η-µ distributions as special cases, together with classical models like Rice, Nakagami-m, Hoyt, Rayleigh, and one-sided Gaussian, as mentioned earlier. The connectivity between the α-KMS distribution and other popular fading models from the literature is summarized in Table 1 in [43].

To this effect, in [44], the impact of nonlinearity (α-µ) and shadowing (inverse Gamma) common acting over the wireless channel was investigated. An approximate closed-form expression for probability density function (PDF) was derived over α-µ/inverse Gamma fading channel. Further, the α-η-F and α-κ-F composite fading distributions were introduced in [24].

In the paper [45], α-κ-µ/Gamma distribution was derived. This is a composite distribution describing a physical fading model consisting of the α-κ-µ multipath fading and the Gamma shadowing. This model contains as special cases the other known composite fading models such as the κ-µ/Gamma model and the α-µ/Gamma model [27] and further widely used composite Rician, Weibull, Nakagami-m, and Rayleigh models as their special cases. First, analytic expressions for the PDF of these distributions are being performed in a suitable algebraic form. After, these expressions were used to derive different basic performance of wireless communications over composite shadowed multipath fading channels.

Accordingly, it is necessary to find expression for the PDF of the received signal power in a closed form, if possible, to describe the compound fading model of shadowed fading channels by dint of the next formula:

$$p(y) = \int_0^\infty p\left(\frac{y}{z}\right) p(z) dz,$$

where $p(y/z)$ is the PDF of the received signal power in short-term fading depending on average power and $p(z)$ is the PDF of the average power [11].

Except fading, the CCI also can disturb desire data in a wireless propagation environment. This impact of the CCI should also be taken into account and mitigated [46, 47] together with fading. There are several techniques adopted in order to reduce their impact, and they will be introduced in the following.

2.2. Mitigation of the Impact of Fading. Diversity is a set of techniques that reduces the effects of fading. At these techniques, several replicas of the same signal from the transmitter are sent to the receiver over independent fading channels. In such a scenario, the probability that all signals will fade simultaneously is considerably reduced. When all these independently fading signals are combined in the receiver, the technique is called diversity combining. A diversity system can be designed if fulfilled [48]:
(i) A copy of the same signal will be received from two or more different paths.

(ii) Each path fades in a different way.

(iii) It is possible to make some kind of diversity of the signals replicas received via these paths.

There are several manners of how the receiver can obtain $L$ independent fading replicas of the same information-bearing signal [48]: frequency diversity (the same information-bearing signal is transmitting by $L$ carriers), time diversity (the signal is transmitting in $L$ different time slots), and space diversity where multiple antennas are used and therefore known as antenna diversity. There is enough space between antennas, and it is considered that obtained signals fade independently. The space diversity technique is one of the most popular types of diversity used in wireless systems [13].

Several diversity combining schemes are used in the receiver to choose the best received signal and play a crucial role in mitigating the side effects of fading in wireless communications. The most common techniques for combining diversity signals are maximal ratio combining (MRC), equal gain combining (EGC), selection combining (SC), Switch and Stay Combining (SSC), and some others [4].

In the absence of interference, MRC is the optimal combining model but the most complex since it requires knowing all channel fading parameters. At MRC combining, the signals from all of the $L$ branches are weighted according to their signal-to-noise ratios (SNR) and then summed. Before summing, the individual signals have to be cophased. EGC combining is similar to MRC combining, except that all weights are set to unity. Although EGC is not optimal, but with slightly inferior performance to the MRC combining, it is a quite satisfactory solution since it does not require knowledge of the fading amplitudes and thus results in less complexity compared to the optimal MRC scheme. The first two introduced combining techniques (MRC and EGC) require all or some of the channel state information (CSI): amplitude, phase, and delay of fading, for all the received signals. This implies the necessity of an additional channel in the receiver for each diversity branch, which increases the overall receiver complexity.

On the contrary, SC combining is also a spatial diversity technique that receives signals from $L$ antennas (diversity branches), but SC combiner processes only one of them, actually the branch with the highest SNR, and sends it to the receiver. SC combining is relatively easy to implement but not optimal because it uses only one signal instead of all the received signals simultaneously. But, this fact gives another advantage: SC combiner can be used in conjunction with differentially coherent and noncoherent modulation techniques since it does not require knowledge of the signal phases on each branch. Contrary to that, the knowledge of the signal phases on each branch is necessary when using MRC or EGC receivers in coherent systems. So, SC is not with the best performance but is the least complicated and the cheapest and therefore often used in wireless communication systems.

For systems using continuous transmission, as frequency-division multiple-access systems, conventional SC is not practical since it requires continuous monitoring of all the diversity branches. Hence, SC is implemented in the form of switched diversity, in which it does not choose the best branch all the time, but the SSC receiver selects some branch until its SNR drops below a predetermined threshold. When this happens, the SSC receiver switches to another branch regardless of whether the SNR of that branch is higher or less than the predetermined threshold. SSC diversity is evidently the least complicated diversity scheme for implementation and also can be used together with coherent, noncoherent, and differentially coherent modulations [13].

Described diversity techniques are actually micro-diversity techniques, implemented at a base station, used to eliminate the effects of short-term (multipath) fading modeled using earlier described distributions.

In [49], the cumulative distribution function (CDF) of the sum $\alpha \sim \kappa, \kappa \sim \eta, \eta \sim \mu$ RVs was determined. The authors accurately estimated the Pout of multibranch maximum ratio combining and equal gain diversity receivers over $\alpha \sim \kappa, \kappa \sim \eta, \eta \sim \mu$ fading channels. Then, in [50], some first-order performances (Pout, bit error probability (BEP), and effective capacity (EC)) of a dual-branch SC system over $\alpha \sim \kappa, \kappa \sim \eta, \eta \sim \mu$ fading channels are derived.

Usually, wireless signals propagate in spaces containing large and small obstacles that affect the quality of propagation that cause a nonnegligible loss of the propagated power (i.e., shadowing). Since wireless systems that operate in environments simultaneously suffer from fading and shadowing, the microdiversity and macrodiversity techniques have to be implemented together. The macrodiversity technique uses the branches coming from multiple base stations to mitigate long-term fading (shadowing) described by log-normal, Gamma, or any other mentioned distributions [13].

As written above, many papers in available literature deal with different types of diversity combining techniques [18, 37, 38, 40, 41]. Thereby, some papers deal with microdiversity systems to mitigate multipath fading [18], and others analyze macrodiversity systems to fight against composite fading. Macrodiversity receiver consists of several microdiversity systems, which diminish fast fading, and two or more diversity combiner mitigating shadow effects [38, 40, 41].

In [38], the reduction of the signal quality degradation is achieved by introducing macrodiversity system. To mitigate fading and shadowing effects, a wireless system consisting of SC combiner at the macrolevel and two multibranch MRC combiners at the microlevel is introduced. The MRC combiners assume the presence of a single-base station disturbed by simultaneous impact of the multipath Nakagami-$m$ fading and Gamma shadowing, so the envelope was modelled by Generalized-K PDF. SC macrodiversity assumes the presence of two base stations, where the average power of the received signal is described using Gamma distribution.
Such a model is applied in [51] to mitigate both short-term fading and shadowing, respectively. It was shown that the use of a compound model of fading, even after the use of diversity combiners, MRC at the microlevel, and SC at the macrolevel, gives the expression for the PDF of the SNR in analytical form. Even in the case with correlated branches, PDF can still be expressed in an analytical form. That was a significant improvement over the approaches which use a log-normal PDF to describe the shadowing.

Our team also deals with mathematical modelling of the next-generation wireless systems and tries to find the best solutions for mitigation of the fading influence. In [52], the expression for $P_{out}$ of the SC macrodiversity system with two $L$-branch MRC microdiversity receivers in Gamma shadowed $k-\mu$ fading environment was derived. This expression was utilized to estimate QoS inside Graphics Processing Unit (GPU), enabling a software simulation environment for optimal planning of mobile networks in smart cities using deep learning for demand prediction and linear optimization.

A wireless mobile communication system with SC macrodiversity reception for the elimination of Gamma shadowing was analyzed in [53]. The macrodiversity system consists of three microdiversity SC receivers serving to mitigate $k-\mu$ short-term fading. The closed-form expression for average level crossing rate (LCR) of macrodiversity SC receiver output signal was performed.

There are many authors who deal with this subject and many papers in the available literature which consider the means for the suppression impact of fading and shadowing, but still new generalized fading distributions were not considered enough. This is a big area for further investigations. Next to fading and shadowing, it is necessary to include the interference effects in researching the performance of wireless systems.

Analysis and derivation of several performances of mobile radio systems working over composite fading/shadowing channels in the presence of colocated interference were given in [54]. Fading and CCI have Nakagami-$m$ distribution, and shadowing has Gamma distribution. In [55], the closed-form expressions for the secrecy performance of transmit antenna selection (TAS) and MRC systems, with and without the CCI over $\eta-\mu$ fading environments, were performed.

The paper [37] processes second-order statistics of macrodiversity system consisting of SC combining for radiofrequency (RF) vehicle-to-infrastructure (V2I) communications in an interference-limited fading channel. The macrodiversity receiver is used to reduce the impact of composite fading.

In [56], the wireless communication system in $\alpha-\eta-\mu$ fading environment subjected to the impact of CCI was investigated. The performance analysis of selection diversity used at the reception under simultaneous occurrence of multipath fading and shadowing was done, through derived novel composite Gamma long-time faded $\alpha-\eta-\mu$ fading distribution. Macrodiversity system with SC in the receiver and two microdiversity SC combiners, under the influence of Rayleigh short-term fading, Rician CCI, and Gamma shadowing, was considered in [57].

The multiple-input multiple-output (MIMO) is a very efficient antenna technology for wireless communication systems where multiple antennas are used at both sides, in transmitter and receiver [58]. The antennas at each end are combined in order to minimize errors, enlarge data rate, and improve the spectrum utilization and channel capacity. This technique allows radio signals to transmit over many different paths at the same time. This manner of transmission with multiple versions of the same signal provides possibilities for signals to arrive at the receiver without fading and CCI influence. This technique increases the signal-to-noise ratio and transmission quality, which creates more stable connections. The use of multiple antennas at transmitter and receiver is a cost-effective technology that offers many leverages [59].

A massive MIMO technology is considered an important technology in 5G. In this sense, the papers [60, 61] consider the standardization process of MIMO technology and the technical advantages of massive MIMO. In Figure 1 in [60], a basic model of massive MIMO is shown. The massive MIMO technologies at conventional sub-6 GHz frequencies are used for long distance outdoor communications, and mmWave technologies are deployed to provide a high data rate for indoor users [62].

In [63], a multiuser system with an arbitrary number of users communicating with a distributed receive array over independent Rayleigh fading channels was considered. A perfect CSI is assumed at the receiver. The receive array carries out minimum mean squared error (MMSE) or zero-forcing (ZF) combining. Some important dynamic characteristics of MIMO mobile fading channels (the level crossing rate and average fade durations (AFD)) were studied in [64].

An interference-limited MIMO cellular system was discussed in [65].

2.3. Network Performance Indicators. Many performance indicators are used to describe the quality of signal transmission in wireless systems. These statistical characteristics can be of the first and second order. Among the statistical characteristics of the first order are signal-to-noise ratio, signal-to-interference ratio (SIR) (used if interference is the dominant disturber, and noise can be ignored [66]), signal-to-interference-plus-noise ratio (SNR), outage probability, channel capacity, average symbol error probability (ASEP) or average bit error probability (ABER) or average bit error rate (ABER), moments, moment generating function (MGF), amount of fading (AoF), and others [67].

The instantaneous SNR per bit, $\gamma$, is a time-invariant RV with a PDF, $p_{\gamma}(\gamma)$, which depends on the type of fading [4]. The moment generating function $M_{\gamma}(s)$ of real RV is [4]

$$M_{\gamma}(s) = \int_{0}^{\infty} p_{\gamma}(\gamma) e^{sy} dy.$$  

(2)

The MGF of $\gamma$ is the Laplace transform of $p_{\gamma}(\gamma)$.

Taking the first derivative of previous expression (2), with respect to $s = 0$ [4],
\[
\mathcal{Y} = \frac{dM_\gamma(s)}{ds} \big|_{s = 0},
\]
the instantaneous SNR is obtained because it is valid:
\[
M_\gamma(s) = E[e^{s\gamma}] = \int_0^\infty e^{s\gamma} p_\gamma(\gamma)d\gamma,
\]
where \( s \) is a real RV and the \( n \)-th moment of \( \gamma \) will be
\[
m_n = E[\gamma^n] = \frac{d^n}{ds^n}M_\gamma(s) \big|_{s = 0} = \int_0^\infty \gamma^n p_\gamma(\gamma)d\gamma,
\]
where \( E \) is the statistical expectation operator.

In most cases, the PDF describes the RV fully, but sometimes, only its partial description is available provided by statistical average values. The statistical average values play an important role in the RV characterization and are called moments. They describe the shape of any distribution, where the most important and useful are the first few moments of RV as well as joint moments between any pair of RVs named the correlation and the covariance. The first moment is the expected value, that is, the mean value \([68]\). The second moment is also very important because it is actually the mean squared value or variance or the signal’s average power. The standard deviation is the positive square root of the variance. The following is the normalized third central moment, under the name the skewness. It presents a measure of limitedness of the distribution or the asymmetry of the PDF of a real RV about its mean value. The fourth central moment is the kurtosis. This measure indicates the thickness of the tail of the distribution compared to the normal distribution of the same variance.

Another important statistical characteristic of fading channels is the amount of fading. It is connected with the fading PDF as \([4]\)

\[
\text{AoF} = \frac{\text{var}(\alpha^2)}{\left(\frac{E[\alpha^2]}{\Omega^2}\right)} \cdot \frac{E\left[(\alpha^2 - \Omega)^2\right]}{\Omega^2} = \frac{E(\gamma^2) - (E[\gamma])^2}{(E[\gamma])^2}
\]

\( E[.] \) denotes statistical average, and \( \text{var}(.) \) denotes variance. This measure of the level of the fading is independent of the average fading power \( \Omega \). The AoF is introduced to quantify the severity of fading \([11]\). Based on \( (6) \), the AoF can also be defined as the ratio of the variance to the square of the expected value based on the PDF expression of the considered system. According to this, the moments of fading distribution are directly used for calculating the AoF. Well, it is a simple manner to quantify fading as presented by Equation \((9) \) \([69]\):

\[
\text{AoF} = \frac{m_2}{m_4 - 1}
\]

Many papers in accessible literature consider previously defined system performance. One of them is \([70]\), where a macrodiversity system with SC diversity receiver and three microdiversity MRC receivers in correlated Gamma shadowed Nakagami-\( m \) multipath fading channel was analyzed. First, MGF of microdiversity MRC receivers output signals is calculated, and based on these formulas, the closed-form expression for MGF macrodiversity SC receiver output signal envelope is determined. In addition, the BEP of the proposed macrodiversity system with different modulation schemes can be calculated.

It is important to mention that moments are also analyzed in the available literature. In \([44, 71, 72]\), multihop relay is introduced to enable data transmission between base station and mobile users in the risk environments. It is obtained by dividing the long distance into more segments to improve link quality in environments under deep fading and shadowing. The coverage is then significantly increased, and transmission is made more reliable and efficient, which is of great importance. In \([71]\), the wireless three-hop relay system with line of sight was analyzed, and different performance of the first and second order was derived in closed forms, including moments. A similar relay system scenario, but with two sections operating over Nakagami-\( m \) fading channel in the presence of CCI subjected to Rician fading, was observed in \([72]\). The closed-form expressions for moments of SIR were calculated.

In \([44]\), the authors investigated the simultaneous impact of nonlinearity, described by \( a-\mu \) distribution, and shadowing, described by inverse Gamma distribution, to the wireless channel. The expression for PDF was derived over \( a-\mu \) inverse Gamma fading channel utilizing Gauss–Laguerre quadrature polynomial. This formula is then used to evaluate various statistics such as CDF, MGF, and \( n \)-th moment. This work also uses a coding gain and diversity gain to analyze average symbol error probability (SEP) with MRC and EGC diversity combining.

In information theory, the outage probability of a communication channel is the probability that a given speed of information is not supported due to the variable capacity of the channel. The Pout is defined as the probability that the information rate is lower than the required threshold or the point at which the receiver power value is less than the predefined threshold. That power value refers to the minimum SNR or SIR within a cellular network, and it is said that the receiver is out of the range of base station (BS) in cellular communications. A predefined threshold is selected to provide a certain level of QoS. The Pout can be expressed as Equation \((2.23) \) \([73]\):
where $P_{\text{out}} = P_r[y < y_{th}]$, 
\[ \text{(8)} \]
where $y$ is SNR or SIR or SINR value and $y_{th}$ is the threshold value. Hence, $P_{\text{out}}$ is the probability that the value of $y$ falls below a given threshold $y_{th}$, defined as
\[ P_{\text{out}} = \int_{y_{th}}^{\infty} p_y(y)dy, \]
\[ \text{(9)} \]
where $p_y(y)$ is the PDF of the SNR or SIR and mathematically, $P_{\text{out}}$ is actually the CDF of $y$:
\[ F_Y(y) = \int_{y_{th}}^{\infty} p_y(y)dy. \]
\[ \text{(10)} \]

The next performance, the error probability, depends on the channel statistics and the applied modulation technique. The bit error probability (BEP) is the ratio of the number of incorrectly detected bits and the total number of sent bits. The ABEP is one of the most important performance measures of a digital communication system. For determining the ABEP, knowledge of the PDF is necessary. Well, ABEP is given by [54]
\[ P_{ae} = \int_{y_{th}}^{\infty} P_e(y)p_y(y)dy, \]
\[ \text{(11)} \]
where $P_e(y)$ is the conditional error probability (CEP) and has general expressions for different modulation formats. The average BER can be given by [74]
\[ P_r(e) = \int_{y_{th}}^{\infty} P_e\left(\frac{e}{y}\right)p_y(y)dy, \]
\[ \text{(12)} \]
where $p_y(y)$ is the PDF of the instantaneous SNR $y$.

Channel capacity is another important measure of system performance, which is useful for assessing the quality of various services during detection. The ergodic capacity defines the highest achievable value of the signal transmission rate for the case of the considered transmission in the presence of scintillation. The maximum data rate can be achieved after the channel has been expressed as follows, in the unit of bits per second [75, 76]:
\[ C = B \int_{y_{th}}^{\infty} \log_2(1 + y)p_y(y)dy, \]
\[ \text{(13)} \]
where $B$ is the channel bandwidth expressed in Hz.

In some cases, statistical characteristics of the first order are not sufficient, so statistical characteristics of the second order should be determined. In mobile communications with time-selective fading, the level crossing rate and average fade duration are two important second-order statistics allowing full understanding of the time-varying behavior of wireless communication channels [77]. Since the influence of long-time but rare envelope fades is completely different from the influence of short-time but frequent envelope fades [10], the LCR and AFD help to better understand and mitigate the disturbing effects of signal fades.

LCR is an important characteristic of the channel showing dynamic temporal behavior of envelope fluctuations. The LCR for envelope $Y$ at threshold $y$, represented as $N_Y(y)$, is defined as the rate at which the signal crosses level $y$ in the negative (or positive) direction [78], that is, the average number of times that a signal crosses a particular threshold in a positive or negative direction. To calculate LCR, it is necessary to determine the joint probability density function (JPDF) between $Y$ and $Y$, $f_{YY}(y, y)$, and then apply Rice’s formula given by Equation (2.106) [79]:
\[ N_Y(y) = \int_{-\infty}^{\infty} f_{YY}(y, y)dy. \]
\[ \text{(14)} \]

For example, the transition probabilities between different states of a Markov model for fading channels are obtained based on LCR at different levels [80]. The expressions for the LCR and average duration of fades (AFD) are derived in [37] for different RF V2I parameters combinations.

The average fade duration measures how long the signal stays below a given threshold. The AFD can be calculated from LCR. The AFD of $Y$ at predefined threshold $y$ is defined as the average time that a signal remains below the level $y$ after crossing that level in the downward (or upward) direction:
\[ T_Y(y) = \frac{F_Y(y)}{N_Y(y)}, \]
\[ \text{(15)} \]
where $F_Y(y)$ denotes the CDF of $y$ given by (10) and $N_Y(y)$ is given by dint of (14).

AFD has a large number of applications, such as choosing the frame length of coded packetized systems [81], interleaver optimization, and efficient mitigation of the burst of errors owing to long fades. The AFD information is necessary for adaptive modulation schemes, the average time where a particular constellation is continuously used. In multuser cellular systems, where interference from other users disrupts the system performance, AFD is also applied.

For example, the AFD specifies the average length of the error bursts in wireless fading channels. So, in fading channels with rather large AFD, long data blocks are more affected by the channel fades than short blocks. This can help to choose the frame length for coded packetized systems, designing interleaved or noninterleaved concatenated coding methods [82], optimizing the interleaver size, choosing the buffer depth for adaptive modulation schemes [83], throughput (efficiency) estimation of communication protocols [84], and so on. For all enumerated applications, experimentally verified formulas for the LCR and AFD of multipath fading models are needed.

Efficient energy use, called energy efficiency (EE), has a goal to reduce the quantity of energy required to provide products and services [85]. It is one of the major performance metrics for the 5G and 6G wireless communication systems, especially for the areas with excessively dense networks. The paper [86] presents a solution for an end-to-end (e2e) power consumption and considers the energy efficiency for a heterogeneous 5G cellular architecture that splits the indoor and outdoor communication scenarios in these networks.

In Table 1, the performance and formulas by which they are calculated are presented.


### Table 1: System performance indicators.

<table>
<thead>
<tr>
<th>Performance</th>
<th>Formula</th>
</tr>
</thead>
<tbody>
<tr>
<td>Moment generating function</td>
<td>$M_m(s) = \int_0^\infty \lambda_m(y)e^{\lambda_m}dy$</td>
</tr>
<tr>
<td>n-th moment</td>
<td>$m_n = E[y^n] = \int_0^\infty y^n\lambda_m(y)dy$</td>
</tr>
<tr>
<td>Amount of fading</td>
<td>$A_o = \frac{var(e)}{E(e)^2} = \frac{m_2-1}{m_2}$</td>
</tr>
<tr>
<td>Outage probability</td>
<td>$P_{out} = \int_0^\infty \lambda_{out} p_y(y)dy = E_y(y)$</td>
</tr>
<tr>
<td>Average bit error probability</td>
<td>$P_{aw} = \int_0^\infty \lambda_{aw} p_y(y)dy$</td>
</tr>
<tr>
<td>Channel capacity</td>
<td>$C = B \int_0^\infty \log_2(1+y)\lambda_m(y)dy$</td>
</tr>
<tr>
<td>Level crossing rate</td>
<td>$N_c(y) = \int_0^\infty \lambda_f(y, \tilde{y})dy$</td>
</tr>
<tr>
<td>Average fade duration</td>
<td>$T_f(y) = \frac{\int_0^\infty \lambda_f(y, \tilde{y})dy}{N_c(y)}$</td>
</tr>
</tbody>
</table>

2.4. Model-Driven Software Engineering Using Eclipse Modelling Framework and Ecore. The main idea of model-driven software engineering is to leverage system and other domain-related models when it comes to automation of certain software development phases and usage scenarios [87]. Therefore, models can be used in many different ways and for various purposes, ranging from generation of documentation to executable code to system verification and validation. Their role is either providing intuitive notations to end-users or reducing the time necessary for certain manual steps by leveraging model-to-model transformation, model analysis, and code generation. Commonly, while using model-driven tools, the user creates model instances with respect to the underlying metamodels. After that, the models created this way are further processed. Model-to-model transformations have a goal to translate model instance compliant with one metamodel (source) into a form corresponding to the desired metamodel (destination). This step is often included when integration of different tools is necessary or achieving interoperability. On the other side, the model analysis aims to examine whether some conditions are satisfied for a specific instance, usually defined in the form of rules. Finally, the goal of code generation is to create the desired target output (such as code in some programming language or human-readable text) based on the user-provided information. This way, the overall experience using model-driven tools becomes more convenient and user-friendly, even in the case of quite complex systems.

Ecore [88] within Eclipse Modelling Framework (EMF) [89] for Java programming language provides the capabilities for the construction of metamodels that can be further used as templates for user-specific instances, which are structured in a way that complies with the definition provided by the underlying metamodel. Moreover, it gives the ability to generate adapter classes that aid the manipulation of model instances, enabling parsing and traversal, which are crucial steps for retrieval of parameters and code generation. Finally, it offers an automatically generated Eclipse-alike tool with Graphical User Interface (GUI) for convenient model instance design with respect to the given metamodel.

In this paper, metamodels are used for the definition of representations regarding different aspects of network planning: static base station allocation, handover strategy, prediction problem, and component allocation problem in general. The first and the second metamodels provide intuitive notations describing static and dynamic properties of wireless network plan, respectively. After that, model-based representation of prediction problems is used for automated PyTorch deep learning code generation, enabling the automated expandability of the network planning tool when new predictive models are needed. The fourth metamodel is used as a means for achieving interoperability with a multiobjective optimization framework by performing the model-to-model transformation. In the end, the multiobjective optimization process is executed against the CAP representation of the network planning model.

2.5. Multiobjective Optimization Leveraging Pymoo and PyEcore. Multiobjective optimization problems include more than one objective functions, which should be minimized or maximized. In what follows, the generalized form of a multiobjective optimization problem [90] is given:

$$
\text{minimize } f_m(x), \quad \forall m \in 1 \ldots M
$$

subject to $g_j(x) \leq 0, \quad \forall j \in 1 \ldots J$

$$
\begin{align*}
& h_k(x) = 0, \quad \forall k \in 1 \ldots K \\
& x_i^{(L)} \leq x_i \leq x_i^{(U)}, \quad \forall i \in 1 \ldots N.
\end{align*}
$$

Here, $x \in \Omega$ refers to a vector in search of length $N$. $f_m(x)$ denotes the $m$-th objective function, $g_j(x)$ represents the $j$-th inequality constraint, and $h_k(x)$ is the $k$-th equality constraint. Additionally, for each dimension $i \in (1 \ldots N)$ of the variable vector $x$, each $x_i$ stands for the $i$-th variable that will be optimized, and there are box constraints for each variable, given by $x_i^{(L)} \leq x_i \leq x_i^{(U)}$. Here, $x_i^{(L)}$ and $x_i^{(U)}$ are lower and upper bound of the variable $x_i$. While minimizing the objective functions $f_m$, all the equality and inequality constraints have to be fulfilled. However, if we want to maximize one of the objective functions (maximize $f_j$ in the first line of (16) instead), the problem has to be redefined to the minimization of its negative value (given as minimize-$f_j$).

Multiobjective optimization tries to find a set of non-dominated solutions that are as close as possible to the Pareto front. Commonly, convergence and diversity of the obtained solution set are quantified in order to measure the performance.

When it comes to implementation, we rely on pymoo [91, 92], which is a well-known framework tackling multiobjective optimization directly. It offers implementation of state-of-the-art optimization algorithms in Python. The framework itself is modular, so the distinct tasks, such as decision making, visualization, or postprocessing procedures, can be easily accessed.

On the other side, we make use of PyEcore [93], which allows handling of EMF/Ecore metamodels and models in Python. It provides an Application Programming Interface (API) that is compatible with the original EMF implementation in Java. Using PyEcore, the network planning problems can be represented as Ecore model instances. Thereafter, the instance of a model can be verified and imported in the form of Python objects, which is suitable for...
2.6. Deep Learning in PyTorch. Deep learning represents an approach in the field of artificial intelligence relying on artificial neural networks that consist of multiple hidden layers between the input and output [95]. Their aim is to extract features from raw input data, which are as good as possible for making a specific type of prediction for the selected target (output) variable. When it comes to the adoption of GPU in the case of deep learning, it is also highly beneficial regarding the execution speed, as tensor operations responsible for the necessary calculations can be highly parallelized. Deep learning is identified among the key enablers of next-generation high-performance wireless and mobile networks, as it gives the ability to extract useful patterns and make predictions based on a huge amount of network-related data [96]. This knowledge is further leveraged for intelligent decisions leading to many benefits, such as improved performance and reliability. In this paper, deep learning using the PyTorch framework is adopted for traffic demand prediction at given smart city locations within the network planning and simulation environment.

PyTorch [97] is a deep learning framework for Python programming language, released by Facebook in 2016, which covers three main aspects enabling the implementation of predictive models [97, 98]: high-level neural network modelling, tensor-based operations, and dataset representation. In what follows, an overview of crucial classes and functions within PyTorch [97, 98] relevant for this paper is provided.

Module class is the highest-level abstraction when it comes to neural network representation in PyTorch, and any custom predictive model should inherit it. Moreover, it is necessary to provide the implementation for the following two methods: (1) __init__: a class constructor where the structure of a neural network is specified number of layers, number of nodes per layer, and their type (such as linear and convolutional). Various types of layers are available in torch.nn package provided by PyTorch; (2) forward: a function where we define how these layers are interconnected by describing the forwarding of input data through the neural network to the output layer. In this context, depending on layer type and role, different activation functions offered by PyTorch within package torch.nn.functional can be used, such as linear (for input), ReLU (commonly used in the case of hidden layers), sigmoid (output layer for binary classifier), and softmax (output layer of multiclass classifier). On the other side, PyTorch also includes a set of loss functions inside package torch.nn whose role is to estimate the distance between the predicted and expected values. Typical loss function examples are Binary Cross Entropy (torch.nn.BCELoss, in the case of binary classification) and Mean Squared Error (torch.nn.MSELoss, used for regression problems). When it comes to neural network training, PyTorch includes several widely used optimizers as part of torch.optim package, such as Stochastic Gradient Descent (SGD) and Adam that are commonly adopted in the case of supervised learning. The invocation of optimizer.step() in training iterations performs the update of neural network weights, considering the value of learning rate (α) that specifies how much the model will be adjusted as a response to the estimated value of loss function.

On the other hand, Dataset (from torch.utils.data package) is a class whose role is to encapsulate the dataset used for either training or test. However, the implementation of the following methods has to be provided: (1) __init__: it is a constructor where the data is first downloaded (optional) and then the corresponding file is opened. The corresponding data structures are initialized, relying on external libraries for Python, such as pandas or NumPy. (2) __len__: it returns the number of rows (samples) within the dataset. (3) __getitem__: it returns the i-th sample from the dataset. Furthermore, this package also includes DataLoader, which is a utility class aiding the iteration through the samples within the dataset. Additionally, it also gives the ability to shuffle the samples and split them into batches by setting the shuffle=True and batch_size parameters.

3. Related Work

Efficient network planning is a highly relevant issue in next-generation wireless and mobile networks and addressed by many recent scientific publications. For example, [99] considers 5G cellular network planning under service and Electromagnetic Fields (EMFs) constraints as multiobjective Mixed Integer Linear Programming (MILP) problem, aiming to minimize the installation costs and maximize the service coverage level. Similarly, in [100], a multiobjective data-driven approach to 5G mobile network planning leveraging propagation and service demand prediction was introduced, taking into account the economic objectives covering both the location-dependent and independent factors. Moreover, [101] makes use of game theory for optimal placement of base stations within a wireless network, aiming at cost minimization and maximization of capacity and coverage under service demand constraints. Another solution [102] adopts the Grouping Coral Reefs Optimization algorithm for solving mobile network deployment problem optimizing economical cost, coverage, electromagnetic pollution control, and capacity constraints.
On the other side, in [103], the approach to network planning based on the adoption of Integer Linear Programming (ILP) and deep reinforcement learning for multistep decision making and cost minimization was introduced. However, these solutions do not take advantage of using predictions.

The approach to network planning proposed in this paper is built upon our previous works in the area of smart grids [7] and tackling the COVID-19 pandemic crisis [8]. This approach leverages the synergy of deep learning and linear optimization for proactive planning. While deep learning is responsible for demand prediction, linear optimization performs the optimal resource allocation considering the constraints relying on these predictions. In the first case [7], the electricity demand forecast was used for cost-efficient energy trading between the smart grid consumers. On the other side, in the second case [8], the prediction of the new COVID-19 was leveraged for proactive planning of relevant resources, such as respiratory infection test devices, medical personnel, hospital places, and vehicles.

Previously, in [104, 105], we already adopted single-objective linear optimization using AMPL and CPLEX solver for optimal allocation of base stations at given smart city locations. However, in the current paper, the improvement regarding resource allocation is in extension, which adds multiobjective optimization approach, which aims to increase the overall benefits, even in the case of conflicting factors, such as the case of network performance maximization, energy consumption, and cost reduction at the same time. Furthermore, our goal is to ease the planning tool usage even in complex scenarios by relying on a model-driven approach and automated code generation in order to automatize the workflow and tool’s expandability as much as possible, which is not much exploited in other works on this topic. For example, a model-driven approach to software asset deployment leveraging automated code generation has been approved so far in many scenarios, such as fog computing infrastructures [106]. Finally, the handover strategy notation introduced in this paper is inspired by the adaptivity mechanisms previously applied in the area of smart grids [7] and fog computing infrastructure management [106] as well. Compared to other similar works, which cover only static aspects of network deployment planning, our solution also takes into account dynamic properties regarding the performance variation as well.

4. Implementation

4.1. Workflow Overview. Figure 1 depicts the steps which make the network planning workflow.

As the first step, the user designs a model representing a smart city wireless network model GUI-enabled modelling environment built on top of the Eclipse Modelling Framework (EMF) [106]. The following aspects are considered [104, 105]: (1) infrastructure of the service provider (represented as base stations); (2) environmental impact and communication channel properties; (3) service consumers (various types of devices and vehicles); and (4) areas of interest where base stations are about to be deployed within the smart city.

After the step of model instance creation, our framework calculates network performance measures leveraging GPU hardware for fading impact estimation [107]. For this purpose, we make use of NVIDIA CUDA [108] technology and program implementation in C. The crucial advantage of such an approach is in significant speed-up of processing compared against CPU due to loop-level parallelization, which is suitable for fading-related expressions [108]. Considering the fact that fading impact and cochannel interference-related expressions contain infinite sums, they have to be calculated approximately by maintaining only the fixed number of terms (up to 40 in our case) within loops. According to our previous works, the execution is up to 70 times faster than an equivalent program written in Mathematica and executed only on CPU in the case of a laptop equipped with Intel i7-7700HQ quad-core CPU, 16 GB DDR4 RAM, and NVIDIA GTX 1050 GPU with 2 GB VRAM [47, 52, 109–112].

Furthermore, based on the service usage history for the locations of interest, the number of potential users for a given day, taking into account the season, temperature, and COVID-19 cases number, is predicted. Additionally, the energy consumption prediction is performed for the candidate base stations if energy consumption history is available. For predictions, we rely on a deep neural network implemented in Python using PyTorch and making use of GPU for acceleration as well.

Once both the fading-related performance indicators, together with service demand and energy consumption predictions, are known, the user-created model is augmented with the calculated values, so optimal base station allocation to the desired locations of interest can be performed. For this purpose, we leverage pymoo framework for multiobjective optimization in Python. However, before executing the optimization procedure, the model has to be transformed to an appropriate form that can be processed. This task is handled by the model-to-model transformation, which converts WNP model instances to Component Allocation Planning models, which is later described. Finally, in this form, pymoo finds the optimal solutions (one or many), aiming to maximize QoS by keeping the negative impact of fading as low as possible while minimizing the overall costs necessary for network deployment on the other side.

4.2. Wireless Network Planning Metamodel (WNPM). The role of the wireless network planning metamodel (WNPM) is to define the structure of model instances used for planning wireless network deployments and configurations, covering the main aspects which are required for additional processing mechanisms, such as service demand prediction and optimal allocation. Figure 2 illustrates the previously described metamodel in the form of Unified Modelling Language (UML) class diagram.

Within the metamodel, the top-level concept is referred to as Plan. Furthermore, each Plan includes the following modelling elements: (1) Stations: service provider
infrastructure abstraction, embodied as a base station. Precisely, it refers to a set of candidates that are considered to be placed at desired places, depending on the achieved performance and overall costs. Each of the base stations has properties, such as frequency $f_s$, output power $P_s$, and SNR.

Another relevant aspect for the operator’s infrastructure is various types of cost, which include energy distribution ($E_{Cost}$) and maintenance and deployment ($M_{Cost}$) costs apart from the price of the base station itself ($S_{Cost}$). Finally, the property of the base station is also the number of users it can handle for a given location, denoted as $Capacity$.

(2) Locations: places that belong to the area of interest where the considered network deployment is planned with distinct $x$ and $y$ coordinates. Exactly one base station should be selected from the candidate set and installed on each of these places. (3) $X_{[station, location]}$: decision variable that represents the process of allocation. This allocation element is assigned to each pair of $(station, location)$ and takes value “1” if the considered station is about to be placed at the desired location but becomes “0” in the other case. The value of

Figure 1: Network planning workflow overview.

Figure 2: The wireless network planning (WNP) metamodel.
decision variables is determined as the outcome of the multiobjective optimization-based component allocation process. (4) **PerformanceMetric**: statistical indicators illustrating how well the current network deployment performs from the receiver’s perspective. The following commonly adopted measures are covered by the metamodel: outage probability, average bit error probability, level crossing rate, average fade duration, and channel capacity. These values are calculated as the output of the GPU-enabled fading calculation engine [104, 105] and assigned to each (station, location) pair.

On the other side, for each *Location*, the potential service consumers for the given area of interest can be represented as *ConsumerGroup* element, which is an abstraction of a specified number of network users with the selected *ReceiverType* and common characteristics. Among the supported types for receivers, we have included MIMO and the following diversity-based systems as either macro- or micro-diversity: SC combining, MRC combining, and EGC combining. The number of antennas (*L*) at the receiver’s side is also customizable within the modelling tool.

Moreover, the properties of the environment configuration at a given location are encapsulated within the *CommunicationChannel* element. It is expressed in the form of two aspects relevant for wireless communications: fading and cochannel interference, represented as corresponding elements. For these two, appropriate distributions have to be specified from the list of supported, including the commonly used ones (such as Rayleigh, Weibull, *k*-µ, *η*-µ, α-κ-µ, Nakagami-*m*, and Rician).

Additionally, for each of the available distributions, relevant parameters can be set using *ChannelParameter* element that includes name and value. For example, among these parameters are the Rician factor denoted as *K*_*r* in the case of Rician fading and severity parameter *m* in the case of Nakagami-*m* distribution, and many others.

Furthermore, the aspects of service demand history for a desired *Location* are included in the form of a *UsageHistory* element. It consists of service consumer numbers for a given day and season, taking into account the external factors that could affect how crowded the place would be, such as average daily temperature and the number of COVID-19 cases. The purpose of this element is to be leveraged by the service demand prediction module based on deep learning, which learns service demand patterns based on historical data. It treats day, season, and external factors (temperature, COVID-19 cases) as input variables, while the number of users is treated as output (target). On the other side, *ConsumptionHistory* element captures the aspects of base station average energy consumption depending on the number of users.

4.3. **Model-to-Model Transformation.** To solve the optimization problem, we developed a model transformation program that transforms mobile network planning models into new component allocation models that can be solved by the pyallocation framework. The model transformation program is in the Atlas Transformation Language (ATL) [113, 114]. Recently, ATL has become a mainstream model transformation language due to several of its features. These include its support of several metamodelling languages and its integration with Eclipse.

An ATL model transformation program is composed of a set of rules. A rule describes how target model elements should be generated from a given source model element. Therefore, a rule consists of an input pattern with an optional filter condition, which is matched on the source model. Also, a rule has an output pattern that specifies the target model elements that need to be generated for each match of the input pattern. The values of the target elements’ features are specified in the Object Constraint Language (OCL) [115].

In our case, the model transformation program is composed of 10 rules (shown in Table 2), where each rule is applied on one or more source model elements. The full program is available at https://github.com/penenadpi/network_planning_optimization/blob/main/TELCO2CAP/telco2cap.atl. Table 2 illustrates the mapping between source element(s) from the WNP metamodel and target element(s) in the CAP metamodel, where each row corresponds to a single model-to-model transformation rule. More details about the underlying transformation mechanisms we rely on in the context of model-driven CAP are presented in our previous work [94].

All rules in the program are matched rules, except *ResourceConsumption_DefaultUnit*, *ResourceConsumption_Capacity*, *ResourceConsumption_UnitUse*, and *TradeOffWeight*, which are called rules. In ATL, called rules are used to explicitly generate target model elements from within imperative code in other rules. Called rules must be explicitly called from an ATL imperative code block.

The transformation maps the concepts in the network planning domain to their matching ones in the domain of component allocation. In a transformation, the generated component allocation model represents an optimization problem that is solved using the pyallocation framework [94]. The transformation is designed so that the solutions to the optimization problem corresponding to the CAP are the same as the ones for the network planning model. Optimizing the component allocation models has been validated on several case studies, and hence it is expected to also be able to find the optimal solutions for the network planning problem.

Figure 3 shows the metamodel for the component allocation problem. The models generated by the model transformation program conform to this metamodel. An *AllocationProblem* is composed of *Components*, *Resources*, and *Units*. Each component consumes a certain amount from each resource when allocated to a unit. *ResourceConsumption* elements are used to define such amounts. A unit provides different amounts of resources that are captured by *ResourceAvailability* elements. There are *AllocationConstraints* and *AntiAllocationConstraints* that represent the allocation and antiallocation constraints, if there are any. The *TradeOffWeight* elements are used to specify the weights for the different resources. These
weights are necessary in the case of single-objective optimization. For multiobjective optimization, these weights are not necessary; they are ignored by the pyallocation framework.

Figure 4 shows a graphical representation of a mobile network planning problem model. It represents a small problem with three stations and two locations only. The performance metrics to the problem (i.e., the CC, LCR, and Pout values) are shown as values in their corresponding instances in the model. For example, the CC value assigned to the pair of station \( s_1 \) and location \( l_1 \) is 99.

4.4. Predictive Models: Service Demand and Energy Consumption. In this paper, we leverage the results of two regression-based models: service demand and energy consumption prediction.

Prediction of service demand expressed as a number of potential users \( Dem \) for given location \( p \) is performed by making use of deep learning approach and relying on GPU hardware for execution. The following features as input: (1) location id, identifier of the considered area of interest in smart city; (2) day, cardinal number of days within a week; (3) season, cardinal number of seasons; (4) temperature, the average temperature that day for the desired location; (5) coronavirus cases, daily number of persons infected by COVID-19 disease for the city where the considered location belongs. For this purpose, we created a neural network with two hidden layers that consist of 35 nodes each and executed the ReLU activation function. When it comes to the input layer, it must contain 5 nodes, as this value should match with a number of considered input features. From the other side, the output layer contains only one node performing linear activation. Furthermore, for the training phase, we make use of the Adam optimizer and select the Mean Squared Error loss function with value of learning rate \( \alpha = 0.01 \). Regarding the prediction performance, the best achieved mean relative error value was circa 8% on synthetically generated example data.

When it comes to base station energy consumption prediction, we treat the number of daily users as input, while the output represents the average daily energy consumption.
consumption in kW. Similar neural network was used in this case, as the problem is also treated as regression. However, the learning rate $\alpha$ was 0.03. The achieved mean relative error value was around 6%. Furthermore, the predicted value is multiplied by the unit cost of electricity for that location, so the overall energy cost of base station $bs$ is determined, which is leveraged within the optimization model for optimal allocation.

Additionally, to leverage the model-driven approach for automated deep learning code generation, we introduce the prediction problem (PP) metamodel (depicted in Figure 5). This way, users can create framework-independent descriptions of prediction problems that can be used to generate neural networks aiming at various target platforms and programming languages. Specifically, in this paper, the output is PyTorch code for the Python programming language. Two types of predictions are supported within the metamodel: regression (used in this paper) and classification. Each Prediction refers to a Dataset (which can be either used for testing or training) and contains some path for its file and a set of Features. Each Feature can be either an input (independent variable) or a target (the predicted value), which is important for training code generation. Moreover, for each Prediction, we can have one or more Candidate-Models with different parameters: batch size, learning rate, optimizer type, and loss function. In the end, once the code for candidate models is generated, the models are evaluated using the selected metrics (Mean Relative Error (MRE) and Mean Absolute Error (MAE) for regression and Accuracy for classification), while the one with the highest performance metric value is returned as output.

When it comes to the implementation of the previously mentioned predictive models, they are generated by populating the PyTorch code template from Listing 1. All the parameters retrieved from model instance are bold and bounded by “[ ]”, while the derived parameters (calculated based on instance) are written inside “<>.” As output, Python code responsible for definition and training of the

**Figure 4**: A model instance of WNP metamodel.
previously described neural networks relying on PyTorch classes and functions is generated.

4.5. Multiobjective Optimization Model. The underlying optimization model for WNP aims the assignment of best from the available base stations in set $S$ to the desired locations within smart city from set denoted as $L$, keeping the network performance as high as possible while reducing the costs of energy and deployment on the other side. Regarding the costs, several factors relevant to base station $bs \in S$ are considered within the optimization model: $SC_{bs}$: costs related to the base station equipment itself; $MC_{bs}$: equipment maintenance cost for given base station type; $EC_{bs}$: energy consumed on daily basis. Additionally, the cost of energy distribution to the place of interest $p \in L$, referred to as $DC_p$, is also taken into account.

When it comes to objective functions, the first one aims to minimize the costs, which can be expressed as

$$\text{minimize} \sum_{bs \in S, p \in L} X[p, bs] (SC[bs] + EC[bs] + MC[bs] + DC[p]).$$

Regarding the performance, the following indicators are covered: ABEP, AFD, Pout, CC, and LCR. Their values are determined relying on fading expressions, which depend on the selected distribution and cochannel interference, which is not within the scope of this paper. While CC and LCR are maximized, as their increase denotes the improvement of performance, the values of Pout, ABEP, and AFD have to be minimized, as a greater number means worse performance in their case. For that reason, the two additional objective functions take the following form:

$$\text{minimize} \sum_{bs \in S, p \in L} X[p, bs] \left( Pout[p, bs] + ABEP[p, bs] + AFD[p, bs] \right),$$

$$\text{maximize} \sum_{bs \in S, p \in L} X[p, bs] \left( LCR[p, bs] + CC[p, bs] \right).$$

In our case, it is assumed that all the optimization goals are equally weighted in order to make the performance and results comparable to our previous works considering single-objective allocation in the context of network planning [104, 105]. Fine-tuning of both the objective function and performance indicator impact coefficients automatically is outside the scope of this paper and considered as possible future work.

However, there are several constraints that have to be satisfied while performing the optimal allocation. First, for each base station $bs$, its capacity expressed as the maximum number of users which can be served, denoted as $Cap_{bs}$, must be greater than or equal to the predicted number of users $Dem_p$ at given location $p$:

$$\sum_{bs \in S} X[p, bs] \cdot Cap[bs] \geq Dem[p], p \in L.$$

The second group of constraints is related to the threshold values of performance measures, which have to be satisfied for each location $p$: $T_{\text{Pout}}$: maximum outage probability, $T_{\text{LCR}}$: minimal level crossing rate, and $T_{\text{CC}}$: minimal channel capacity and similar for the other two. There are five distinct constraints for each of these measures, following the form as given for Pout:

$$\sum_{bs \in S} X[s, l] \cdot \text{Pout}[s, l] \leq T_{\text{Pout}}[p], p \in L.$$
4.6. Handover Strategy Metamodel (HSM). Now we introduce Handover Strategy Metamodel (HSM) to enable the expression of adaptive and dynamic aspects within the planning tool. UML class diagram of this metamodel is depicted in Figure 6. The highest-level entity is Handover-Strategy, which contains an ordered sequence of steps (step1, bs[1];...; stepn, bs[n]), executed as a Response to some Trigger event given as stepi, bs[j] if (trigger) then responsei. (21)

Each step refers to the specific base station bs, identified by station id (sid). Step consists of two main parts: trigger and response. Trigger covers different types of events, such as a low number of users connected, drop of performance, base station failure, or too many users trying to connect (greater than the base station capacity). When it comes to PerformanceDrop, it represents relational expression (where relop is relational operator) given as performance Metric relop thres hold. (22)

In a similar way, if the number of users is lower than minUsers, a trigger event will be generated in that case.

On the other side, Response represents specific action taken when the trigger event is detected, such as transfer to destination station given by dstSid, switching to energy-saving mode or scaling up by tuning the parameters or turning additional equipment on. Considering the following rules for a base station, the effect would be to transfer all the users to base station s5 when there are less than 3 of them and switch it to energy-saving mode if there are no users:

\[
\text{if (num Users < 3) then Transfer (s5),}
\]
\[
\text{if (num Users < 1) then Standby.}
\]

5. Experiments and Evaluation

In this section, the proposed model-driven framework for cost-efficient and performance-aware network planning is evaluated, taking into account several relevant aspects: (1) execution time; (2) benefits: cost reduction and performance improvement; (3) reduction of time needed for the development of additional prediction modules. Furthermore, the obtained results are compared to our previous works relying on AMPL and CPLEX with single-objective linear optimization. The experiments were performed for varying sizes (number of places and base stations) of model instances and types of fading considered. For the purpose of evaluation, a laptop with the following characteristics was used: i7 7700HQ quad-core CPU, 16 GB of DDR4 RAM, 256 GB M2 SSD, and NVIDIA GTX 1050 GPU with 2 GB of VRAM. All the execution times are given in seconds, based on the average value calculated after 100 runs.

The summary of the achieved results is provided in Table 3. The first column shows how large the model
instance was, expressed as a pair of base station and location set sizes. The second and the third column give the information about the type of fading and cochannel interference distribution considered within the experiment. After that, the next three columns show the processing time necessary for distinct steps: network performance determination under selected fading and cochannel interference type, deep learning-based predictions, and allocation leveraging multiobjective optimization. Furthermore, the next two columns show the achieved benefits compared to the single-objective approach, expressed as a percentage of cost reduction and performance improvement [104]. In this case, the improvement is calculated with respect to QoS value expressed as the sum of fading-related performance metrics with positive impact containing plus sign, while those that reduce the performance are taken with minus. Finally, the last column shows the speed-up regarding the processing time, achieved compared to our previous work [105].

As it is noticeable, the processing time increases with model size for all the steps, as expected. In turn, the benefits vary from case to case, as they are dependent on the model instance itself and the assigned values. However, the speed-up of pyallocation-based optimization is more significant for larger model instances. Additionally, it can be seen that even in cases of the same model size, the processing speed might increase due to more complex fading and cochannel interference distributions considered. Despite the fact that additional time is needed for model-to-model transformation (which is also included for the allocation step during the comparison), this step is still obviously faster compared to our previous works in single-objective mode [104, 105]. When it comes to the execution time of multiobjective mode, it is in line with the results achieved by our previous works relying on single-objective linear optimization using AMPL.

Some of the benefits coming from the adoption of a model-driven approach are presented in Table 4. Here, we compared the time needed for manual creation of two deep neural networks using PyTorch from scratch in the case of one candidate model with the proposed model-driven method (considering the time needed for model creation and code generation). When it comes to PyTorch code writing time estimation, it was based on results obtained and achieved by a professional in this area. The implementation for all the necessary assets was included, predictive model, dataset adapter, training, and evaluation loops.
Finally, the code generation for handover plans with varying numbers of steps (from 1 to 10) was evaluated. Based on the obtained values, the processing time is quite fast (from 0.0002 to 0.0007 s) but increases with the number of steps and is acceptable even in the case of real-time usage scenarios.

6. Discussion

As it is noticeable in Table 2, the processing time increases with model size for all the steps, as expected. On the other side, the benefits vary from case to case, as they are dependent on the model instance itself and the assigned values. However, the speed-up of pyallocation-based optimization is more significant for larger model instances. Additionally, it can be seen that, even in cases of the same model size, the processing speed might increase due to more complex fading and cointerference distributions considered. Despite the fact that additional time is needed for model-to-model transformation (which is also included for the allocation step during the comparison), this step is still obviously faster compared to our previous works in single-objective mode [57, 58]. When it comes to the execution time of multi-objective mode, it is in line with the results achieved by our previous works relying on single-objective linear optimization using AMPL.

It can be concluded from Table 3 that such an approach not only increases the overall expandability of a network planning tool with the ability to include new predictive modules conveniently but also gives the ability to generate them in a shorter time. Based on the obtained values, it is observed that the speed-up increases for more complex neural network architectures. Furthermore, when it comes to multiple model candidate instances, the speed-up is even more significant, as additional time is needed for manual parameter tuning and experimentation. In the case of the 5 candidate models, the maximum achieved speed-up was around 25 times.

Considering the applicability of the proposed solution in practice, there are many benefits. First, the adoption of a model-driven approach reduces the learning curve for experimentation, making the software environment more user-friendly. This way, the experimenters can put more focus on the model design itself rather than spending time on mastering complex tools, which is of utmost importance for efficient large-scale network deployment plans. Thereafter, the automated code generation based on high-level model representations significantly reduces the needed time for many manual steps, such as integration of different tools and creation of predictive modules for different purposes, without going into details of neural networks and corresponding frameworks for their implementation. Additionally, the proposed software environment also covers the planning of performance-aware runtime adaptive behavior using simple notation, apart from static configuration. Therefore, our framework provides the necessary tools that enable rapid prototyping of large-scale next-generation wireless network deployments, aiming at telco operators and other researchers in this area, taking into account cost-efficiency, performance, and dynamic behavior under given conditions, at the same time. From an end-user perspective, the experimentation costs in both time and money for network deployments are reduced significantly, which is highly beneficial. Moreover, the architecture of the presented network planning is future-proof, as each backend component (fading calculation, prediction, and optimization) can be swapped with the newer one independently, providing the appropriate code generators and/or model-to-model transformations. Finally, our implementation leverages GPU for fading-related calculations and deep learning-based predictions, which obviously speeds-up processing compared to traditional CPU-only programs.

7. Conclusion and Future Works

In this paper, a model-driven approach to next-generation wireless network planning was proposed. It relies on deep learning-based predictions in order to introduce proactivity into planning. For base station assignment to given smart city locations of interest, the task is treated as a component allocation problem and uses multiobjective optimization. The approach is experimentally evaluated for various model sizes. According to the obtained results, the achieved plans showed significant benefits in terms of cost reduction and performance improvement. On the other side, the execution time is reduced compared to previous works, while the overhead of model-to-model transformation is negligible in this case. The benefits of the model-driven approach are also observable when it comes to the framework’s ease of usage and expandability. Intuitively, model-based representation of both the static and dynamic aspects of network planning reduces the cognitive load of the experimenters, even in complex scenarios. Except that, using a model-driven approach gives the ability to include additional predictive models much faster than writing code manually. Finally, the proposed model-driven framework is also applicable to other domains, and deep learning-based technologies in case appropriate model-to-model transformations and code generation algorithms are provided.

In future work, we plan to extend the presented work, considering the aspects of security in next-generation wireless systems. Adoption of deep learning in the context of anomaly and attack detection will be explored. Moreover, the elements of adaptability and reconfigurability in real-time would be covered in-depth, such as handover strategies execution leveraging software-defined radio (SDR) command generation as a response to the events detected using deep learning. At last, the aspects related to the selection of the most suitable solution among the several outputs of a multiobjective optimization algorithm for a given planning scenario would be included.

Data Availability

The source code and data that support the findings of this study are openly available at the GitHub repository: https://github.com/penenadpi/network_planning_optimization/.
Conflicts of Interest
The authors declare that they have no conflicts of interest.

Acknowledgments
This work was supported by the Ministry of Education, Science and Technological Development of the Republic of Serbia. Special thanks are due to Julian Blank (Department of Computer Science and Engineering, Michigan State University, East Lansing, USA) for help and useful discussion about the adoption of pynmo and multiobjective optimization.

References
[14] H. Huang and C. Yuan, "Product of $\kappa$-$\mu$ and $\alpha$-$\mu$ distributions and their composite fading distributions," in Proceedings of the IEEE/CIC International Conference on Communications in China (ICCC), Beijing, China, August 2018.
[27] H. Al-Hmood and H. S. Al-Raweshidy, "Unified modeling of composite $\kappa$-$\mu$ and $\eta$-$\mu$ Gamma, $\kappa$-$\mu$ $\eta$-$\mu$ Gamma, and $\kappa$-$\eta$-$\mu$ Gamma fading channels using a mixture gamma distribution with applications to energy detection," IEEE Antennas and Wireless Propagation Letters, vol. 16, pp. 104–108, 2017.


