Survey

Performance models for wireless channels

Dmitri Molchanov*

Department of Communication Engineering, Tampere University of Technology, P.O. Box 553, Tampere, Finland

ARTICLE INFO

Article history:
Received 13 December 2009
Received in revised form
15 April 2010
Accepted 15 April 2010

Keywords:
Wireless channels
Performance evaluation models

ABSTRACT

Performance modeling of wireless access technologies is useful to understand their limitations in various operational conditions and find a way to improve their performance. In the past two decades a number of models have been proposed. These models are often more complicated compared to those developed for wired networks. The reason is that in wireless networks performance degradation is caused by both incorrect reception of channel symbols at the physical layer and queuing at higher layers. Various error control mechanisms used to hide the effect of error-prone channel behavior complicate performance analysis and often require restrictive assumptions to retain analytical tractability. The aim of this paper is to review performance evaluation models proposed for wireless channels, highlighting their basic ideas, shortcomings, and advantages. We consider models developed for both centralized and distributed access technologies. Potential applications and extensions are also discussed. We believe that this study may provide a starting point for those looking for a suitable modeling framework and allow time to be saved in developing new performance models.

© 2010 Elsevier Inc. All rights reserved.

1. Introduction

Wireless access technologies are becoming widespread in our everyday life providing a fast and convenient way to access various Internet services “anytime and anywhere”. In response to the constantly growing need for more bandwidth at the air interface, vendors and standardization bodies continue to improve existing technologies and develop new ones. These new technologies include many parameters that are not necessarily optimized for specific environments. To evaluate the efficiency of new wireless access technologies and optimize the performance of existing ones, performance models are required.

The aim of this paper is to provide an in-depth review of analytical models proposed so far for performance evaluation of wireless access technologies. The motivation behind this work is manifold. First of all, performance modeling of wireless channels is an extensively studied research area nowadays. Although the literature on this topic has exploded over the past two decades, to the best of the author’s knowledge, there have been no studies trying to sort the performance evaluation models proposed to date and explain their basic approaches and ideas. Given the existing database of these models it is difficult for a newcomer to the field to choose a point to start from. Secondly, performance modeling of wireless systems brings a number of unique challenges that have never been an issue in wired networks. This includes the error-prone nature of wireless channels, channel access methods, various channel adaptation mechanisms, etc. As a result, performance models are often complex...
we review and discuss models used to capture wireless channel characteristics. We distinguish between signal strength models and PDU error models, discussing their applicability and shortcomings for various performance modeling studies. We also briefly address applications of performance evaluation models considered in this paper to cross-layer modeling studies. Next, in Section 4, we consider arrival traffic models for both real-time and non-real-time applications, highlighting that the TCP protocol alone poses many challenges to a performance analyst. We also discuss why batch arrival models are better suited for accurate performance modeling of real-time applications in a wireless environment. Performance models for centralized and distributed access mechanisms and their applications are reviewed in Sections 5 and 6, respectively. In both sections we distinguish between models developed for real-time and non-real-time applications. Conclusions are drawn in the last section.

2. Basic notes on performance models

2.1. Wireless channel specifics

There are a number of factors affecting the performance experienced by applications running over wireless channels. These are the traffic characteristics of a given application, time-varying characteristics of wireless channels, and protocols with a set of their parameters. It is known from teletraffic theory that each application is characterized by its own traffic characteristics that may significantly affect the performance provided by a particular networking technology. Environmental characteristics of a landscape and movement of a user are stochastic factors affecting the propagation characteristics of a wireless channel. Protocols and their parameters determine how given traffic is handled in the protocol stack. The performance that a given application achieves is then a complex function of the properties and interactions between these components.

To optimize the performance of applications in a wireless environment, state-of-the-art technologies incorporate a number of advanced channel adaptation mechanisms at different layers of the protocol stack. These are error correction techniques including both forward error correction (FEC) and automatic repeat request (ARQ), adaptive size of protocol data units (PDU) at different layers, automatic modulation and coding (AMC) schemes, multiple-in multiple-out (MIMO) antenna design, power control mechanisms, etc. At the application layer adaptive compression and coding (ACC) can be used to reduce the rate required from the network. These mechanisms are implemented at different layers of the protocol stack and affect the performance provided to applications differently [1]. Various channel adaptation mechanisms and their places in the protocol stack are marked by a grey color in Fig. 1, where nRT stands for non-real-time applications and RT refers to real-time applications. To effectively deal with the error-prone nature of wireless channels, a particular wireless access technology implements one or more of these mechanisms. To understand what performance level can be provided to applications under different wireless channel and traffic conditions, studies on the operation of these mechanisms are required.

---

![Channel adaptation mechanisms in the protocol stack.](image-url)
2.2. Performance metrics

Historically, the first performance models for wireless channels were those evaluating their theoretical capacity in the absence of communication protocols. At the beginning of 1960s Gilbert noticed that the noise in wireless channels is autocorrelated rather than independent [2]. He proposed using a two-state Markov model to estimate the theoretical capacity limits of wireless channels. His work was further extended by Elliot [3] and Fritchman [4]. Although those studies provided a starting point in performance characterization of wireless channels their impact is rather limited nowadays. The reason is that the authors completely neglected the effect of the protocols that have to be implemented to deal with error-prone nature of wireless channels. Indeed, high absolute values of the bit error rate (BER) impose specific limitations on the way traffic is handled in the protocol stack and require sophisticated error control mechanisms. These mechanisms may substantially affect the capacity of wireless channels.

The effect of protocol characteristics on IP layer performance metrics is illustrated in Fig. 2, where the packet loss probability and the mean transmission time of a packet are plotted as a function of FEC code and the number of retransmission attempts allowed for a single frame, r, at the data-link layer. These dependencies have been obtained analytically assuming a Type I non-persistent hybrid ARQ operating in selective repeat regime with the maximum number of retransmissions allowed for a single frame set to 3, 6, and 9 and (255, 131, 18), (255, 87, 26) BCH FEC codes (see [5] for specific details). The IP packet size was assumed to be 500 bytes including all headers. Note that first two figures correspond to the (255, 131, 18) BCH FEC code, the last two figures were made assuming r = 3. It is easy to notice that the choice of parameters of channel adaptation schemes at the data-link layer may significantly affect the performance provided to the IP layer.

Another class of models tries to represent the performance of wireless channels in terms of their spectral efficiency. Spectral efficiency is defined as the maximum throughput divided by the bandwidth in hertz (Hz) of a communication channel and measured in bits per second per hertz of the bandwidth. These models do take into account the properties of channel adaptation mechanisms implemented at lower layers. While they are useful in estimating the theoretical
throughput provided by a wireless channel, they still neglect the traffic properties of applications. Indeed, assuming a certain traffic stream wireless channel performance is determined by the stochastic interplay between the arrival and service processes and their properties are equally important, it was demonstrated by Li and Hwang in [6] that both the empirical distribution of the arrival process and the structure of its normalized autocorrelation function (NACF) may have a significant effect on the performance parameters provided by a servicing system. Hayek and He [7] confirmed the importance of an empirical distribution of the packet arrival process showing that the queuing response may vary for inputs with the same mean and NACF.

The effect of arrival traffic characteristics on performance metrics provided by a wireless channel at the IP layer is demonstrated in Fig. 3. These figures have been obtained assuming a Type I non-persistent hybrid ARQ operating in a selective repeat regime with the maximum number of retransmissions allowed for a single frame set to 6 and the (255, 87, 26) BCH FEC code. The IP packet size was assumed to be 500 bytes, including all headers, and the buffer size was set to 60 packets [5]. The packet arrival process was a covariance stationary Markov modulated one capable of capturing both an empirical distribution of the number of arrivals and the autocorrelation properties of real traffic patterns. Packet losses were allowed to occur as a result of buffer overflow at the IP layer and imperfect error correction (non-persistent ARQ) at the data-link layer. Note that the first two figures correspond to the offered traffic load, $\rho$, set to 0.8, the last two figures were made assuming that lag-1 NACF, $K(1)$, is 0.0. It is easy to observe that, depending on the characteristics of the packet arrival process, the difference between performance metrics at the IP layer can be substantial.

Wireless models with realistic channel and traffic characteristics are often more complicated to solve for conventional performance metrics of interest compared to...
similar models developed for wired networks. The reason is that packet losses in wireless networks can be caused by both buffer overflow at the IP layer and imperfect error correction strategies at the data-link and physical layers. When distributed medium access control (MAC) is used, packet collisions may also lead to packet losses. Furthermore, the service time distribution of a packet is affected by channel adaptation schemes implemented at the data-link and physical layers. The effect of these mechanisms is often complicated to take into account analytically. Finally, arrival traffic characteristics bring an additional degree of complexity.

2.3. Layer of interest

For any given configuration of the protocol stack performance parameters are usually computed for layers higher than physical, such as data-link, network, transport, or application layers. Throughput is often used to characterize performance ARQ protocols at the data-link layer. The probability distribution functions (and their moments) of frame/packet losses, delay, and delay jitter are common data-link and network layer performance measures. Throughput is an example of a transport layer performance parameter as it defines the actual rate obtained by the transport layer protocol such as TCP. The goodput is an application layer performance metric that determines the throughput obtained by an application excluding control information and possible retransmissions at underlying layers.

Nowadays, the performance of wireless channels is often estimated at the data-link layer. There are a number of reasons behind this. First of all, the data-link layer incorporates medium access procedures that are expected to affect the performance provided to higher layers. Secondly, channel adaptation mechanisms are mainly implemented at the data-link and physical layers. From this point of view, performance models at the data-link layer abstract the functionality of the underlying technology-specific protocols describing the performance provided to higher layers.

Recently, models describing the performance of applications at higher layers started to appear. The reason is that wireless technologies are nowadays extensively used to access Internet services and performance in IP-based networks is conventionally measured at IP or higher layers.

3. Wireless channel models

Modeling of wireless channel characteristics provides a starting point in any performance modeling framework. Wireless channel models are often represented in terms of the stochastic process as a function of propagation environment and transmission characteristics such as emitted power, modulation, noise, length of the codeword/frame/packet, FEC code, etc. Depending on how many of these properties are implicitly or explicitly taken into account we distinguish between (i) models representing the received signal strength or related process at the physical layer (ii) PDU error models at a certain layer of interest. There are a number of models proposed so far in both categories.

3.1. Signal strength and SNR models

Radio waves experience three propagation mechanisms, namely, reflection, diffraction, and scattering. When the propagating wave hits an object that is large compared to the wavelength, the wave is reflected. Reflection causes a phase shift of 180° between the initial and reflected rays. Diffraction occurs when the wave hits an impenetrable object. The dimensions of the object causing diffraction are comparable to the wavelength of the wave. This effect allows the wave to reach places behind the object that usually cannot be reached in line of sight (LOS) transmission. Scattering occurs when a wave travels through a medium which contains many objects whose dimensions are smaller compared to its wavelength. The wave is scattered into several weaker signals. Due to these phenomena the propagation path can be quite complex.

To represent the characteristics of a wireless channel in terms of the received signal strength, propagation models need to be used. We distinguish between two types of propagation models, namely, large-scale and small-scale propagation models. The former models focus on predicting the received local average signal strength (RLASS) over large travel distances. Propagation models characterizing rapid fluctuations of the received signal strength over short time durations or short travel distances are called small-scale propagation models. Combined with noise, large-scale and small-scale propagation models can be used to represent the signal-to-noise ratio (SNR) process.

When a mobile user moves away from the transmitter over large distances RLASS gradually decreases. The RLASS value is computed by averaging the received signal strength over movements of 1–10 m. Most analytical and empirical large-scale propagation models assume that the RLASS decays as a power law function of distance between the transmitter and a receiver. Large-scale propagation models are mainly used at the planning phase of a mobile system, where they provide rough estimates of the coverage area served by a single transmitter. Free-space and two-rays ground reflection models are two examples of large-scale propagation models.

When a mobile user moves over short distances the instantaneous received signal strength varies rapidly and may cause incorrect reception of channel symbols even when the average channel conditions are relatively good. The reason is that the received signal is a sum of many components coming from different directions due to reflection, diffraction and scattering. Since the phases, amplitudes and arrival times of components are random, the resulting signal rapidly fluctuates. Small-scale propagation models allow one to capture this phenomenon. Rayleigh and Rician fading models are two examples of small-scale propagation models.

Small-scale propagation models are classified into theoretical and empirical models. Theoretical models capture the fundamental principles of radio propagation. Unfortunately, most of those are relatively simple and cannot provide the required accuracy (see e.g. [8]). To provide a better description of the received signal strength measurement-based empirical models started to appear. In such models, all physical
propagation phenomena are implicitly taken into account regardless of whether they can be isolated and separately recognized. Their accuracy depends on similarities between the environment under consideration and the environment where measurements have been carried out.

As an example consider the measurement-based model proposed in [9], where the authors concentrated on modeling the SNR process of IEEE 802.11b wireless channel. They found SNR measurements to follow a normal distribution with a geometrically decaying NACF and proposed capturing these properties using an autoregressive process of order 1, AR(1), in the form $X(n + 1) = \phi_0 + \phi_1 X(n) + \epsilon(n + 1)$, $n = 0, 1, \ldots$ where $\phi_0$ and $\phi_1$ are some constants, $\epsilon(n)$, $n = 0, 1, \ldots$ are independently and identically distributed (iid) random variables having the same normal distribution with zero mean and variance $\sigma^2(\epsilon)$. The AR(1) process is fully characterized by a triplet $(\phi_0, \phi_1, \sigma^2(\epsilon))$. The authors demonstrated that these parameters can be found as follows

$$
\begin{align*}
\phi_1 &= K_Y(1) \\
\phi_0 &= \mu_Y(1 - \phi_1) \\
\sigma^2[x] &= \sigma^2[Y](1 - \phi_1^2),
\end{align*}
$$

where $K_Y(1)$, $\mu_Y$, and $\sigma^2[Y]$ are the lag-1 NACF, mean, and variance of SNR observations, respectively. Note that the proposed model is valid when the SNR is averaged over relatively long time intervals, e.g. 0.1–1 s.

### 3.2 Bit error models

PDU error models including symbolic, bit, and frame error models are represented by binary stochastic processes, where 0 stands for correct PDU reception and 1 refers to incorrectly received PDU. Bit error processes are conventionally represented using Markov models starting from the simple two-state Gilbert model [2] to quite complicated multi-state high-order Markov models [10]. In general, models of bit error observations can be divided into two broad categories. These are models based on partitioning of SNR observations and models based on direct fitting of parameters.

Recent measurement studies of the SNR and PDU error processes demonstrated that they are characterized by a strong degree of autocorrelation. To model these processes, hidden Markov models (HMM) are often used. HMMs, also known as Markov modulated processes, are probabilistic functions of Markov chains [11]. If the stochastic variable of interest is the state of the Markov process, we are given a conventional Markov model whose outputs are directly observable. A straightforward way is to associate each state of the “observable” Markov model with a certain random variable. Simulating this model we do not directly observe the state of the Markov chain, but a stochastic process over the Markov chain. It has been shown that HMMs can be effectively used to model PDU error processes resulting from different fading phenomena, including frequency-selective [12], flat [13], and fast fading [14]. See [15, 16] for a review and further discussion.

HMM, based on partitioning of the SNR was firstly proposed by Wang and Moayeri [13]. The authors assumed a Rayleigh fading channel and used a Markov process with a finite number of states as follows. Consider a discrete-time Markov chain $\{S(n), n = 0, 1, \ldots\}$, $S(n) \in \{1, 2, \ldots, M\}$, where each state is associated with the so-called binary symmetric channel (BSC). The BSC associated with state $i$, $i = 1, 2, \ldots, M$, determines how a channel symbol is transmitted while the channel is in the state $i$. The term symmetric stems from the fact that both 0 and 1 are assumed to be incorrectly received with the same probability. To parameterize such a model we have to determine the transition probability matrix $D$, and the so-called crossover probability vector $p = (p_2, p_3, \ldots, p_M)$, each element of which determines a certain BSC. The state transition diagram of the described model and the corresponding BSCs are shown in Fig. 4. In wireless channel modeling this model is often referred to as a finite state Markov chain (FSMC).

To parameterize this model we have to partition the received SNR into a finite number of intervals, $M$, where each interval represents the state of the Markov chain. Elements of the transition probability matrix, $D$, can be calculated by counting the number of transitions between SNR intervals, as shown in [13]. Taking into account a certain modulation scheme, HMM modeling the SNR process can be extended to the case of the symbol error process. Given a simple modulation scheme it is sometimes possible to analytically compute the probability of having a symbol in error as a function of the received SNR. When an analytical expression is not available field measurements need to be used. Taking into account a certain number of bits carried by a single symbol the bit error process can be obtained. There are a number of issues that are still open in parameterizations of this model, i.e. how best to choose the number of intervals to which SNR is partitioned, for which modulation techniques construction of this model is possible, etc. To date usage of this model with binary phase shift keying (BPSK) [13] and differential quadrature PSK (DQPSK) has been reported [17].

Note the described approach is essentially cross-layer in nature. It easy to notice that the described Extensions do not affect the overall structure of the original HMM modeling the SNR process. However, when a single channel symbol is used to represent more than one bit, the model becomes batch in nature, as each state is now associated with a certain number of correctly or incorrectly received bits. This property modifies the initial properties of the SNR and symbol error processes, adding additional memory to the already autocorrelated process.

Although bit error models can be obtained as an extension of SNR models, their major advantage is that they can be derived directly from the bit error traces. To date a number of such models has been proposed. The seminal work that mathematically described the bit error pattern observed on

---

**Fig. 4 ~ State transition diagram of the model and associated BSCs.**
a wireless channel is due to Gilbert [2]. This model has only two states, one of which is error-free while the other one is associated with a certain bit error probability, \( p \). Elliot [3] extended it allowing both states to have non-zero bit error probabilities. The next extension was due to Fritchman [4], who allowed a Markov chain to have more than one error-free state. It was shown that Fritchman’s model captures error-free intervals more precisely. In [18], the authors reviewed previous studies and compared the performance of different models. They demonstrated that Gilbert’s, Elliot’s, and Fritchman’s models may not capture bit error statistics accurately and proposed using HMMs instead.

A simple HMM of the bit error process, based on direct fitting to statistical data, is defined as follows. Assume that the bit error process is covariance stationary in nature with a certain BER and lag-1 NACF. Our aim here is to capture these two characteristics. Let \( \{W_E(n), n = 0, 1, \ldots\} \), \( W_E(n) \in [0, 1) \), denote the bit error model with modulating Markov chain \( \{S_E(n), n = 0, 1, \ldots\} \), \( S_E(n) \in \{0, 1\} \). The model is completely defined using two matrices \( D_E(k) \), \( k = 0, 1 \), containing transition probabilities from state 0 to state 1 with correct \( (k = 0) \) and incorrect \( (k = 1) \) reception of a single bit. The authors in [19] demonstrated that there is a HMM called switched Bernoulli process (SBP), matching the mean and lag-1 NACF value of covariance stationary bit error observations. This model is given by

\[
\begin{align*}
\alpha_E &= (1 - K_E(1))E[W_E] \\
\beta_E &= (1 - K_E(1))(1 - E[W_E]).
\end{align*}
\]

where \( \alpha_E \) and \( \beta_E \) are transition probabilities from state 0 to state 1 and from state 1 to state 0, respectively, \( K_E(1) \) is the lag-1 autocorrelation of bit error observations, \( E[W_E] \) is the mean of bit error observations, \( f_{0,E}(1) \) and \( f_{1,E}(1) \) are probabilities of incorrect reception of a bit in states 1 and 2, respectively. Values of \( E[W_E] \) and \( K_E(1) \) have to be estimated from empirical data. Although extension of this model to the case of an \( M \)-state HMM is rather straightforward, more advanced fitting procedures have to be used [16].

Historically, bit error models were used to evaluate the capacity of wireless channels in terms of their theoretical throughput or spectral efficiency in the absence of higher layers protocols [2,3]. During the last two decades they were rarely used in performance evaluation studies. The reason is that channel adaptation mechanisms implemented at the physical and data-link layers significantly complicate extensions of these models to the higher layers at which the performance of applications is to be evaluated. Instead, PDU error statistics at higher layers were directly measured and modeled.

Note that, recently, bit error models are starting to get more attention due to their suitability for cross-layer performance modeling studies, as we will briefly discuss further. However, when used in cross-layer studies bit error models do not allow one to evaluate the effect of modulation and reception techniques at the physical layer. The received signal strength of SNR models is better for this purpose.

### 3.3. Frame error models

Frame error models are defined at the data-link layer. They abstract the functionality of the physical and data-link layers’ channel adaptation mechanisms and are mostly based on direct fitting to observations of frame error processes. The reason for their widespread usage is that the operation of the data-link layer protocols of various wireless access technologies can be directly traced.

Implicitly assuming covariance stationarity of frame error observations, Zorzi and Rao, in [20–23], claimed that the two-state Markov model is sufficient to capture frame error statistics at the data-link layer. In [23] the accuracy of this model has been evaluated. The authors found that their model provides an adequate trade-off between accuracy of modeling and complexity of fitting to statistical data. Later, the focus of data-link layer wireless channel modeling was mainly on IEEE 802.11 wireless local area network (WLAN) technologies. The reason is the widespread availability of IEEE 802.11 equipment and the unwillingness of network operators to share their data. We specifically warn that those conclusions made for IEEE 802.11 channels may not be true for other wireless technologies.

In [24], the authors carried out a statistical analysis of IEEE 802.11b frame error traces and used a number of models, including HMM and first- and high-order Markov models. They demonstrated some inconsistency in using Markov models to represent frame error processes. Particularly, no models were found to provide satisfactory results for all available measurements. However, the authors did not suggest any particular reason for that. Statistical analysis of IEEE 802.11b frame error traces was also carried out in [25]. It was suggested that first-order Markov models with exceptionally large state spaces are capable of accurately capturing the autocorrelation properties of frame error traces. In particular, in [10], a 512-state Markov chain was introduced. However, the authors also noticed some uncertainty in the generality of their model. The number of states required for accurate modeling differs from trace to trace. Although the authors did not suggest any particular reason for this phenomenon, the property they revealed serves as an indication that some important statistics are still undetected or that Markov models are not best suited for wireless channel modeling.

### 3.4. Cross-layer channel modeling

The received signal strength, SNR, or bit error models, while were shown to be useful for design of transceivers, cannot be directly used in performance evaluation studies. To successfully use these models they must be properly extended to higher layers, providing convenient characterization of the dynamic nature of a wireless channel at the layer of interest. For example, based on the specific SNR model, modulation and detection technique, an FEC code frame error model can be obtained.

Cross-layer performance modeling is nowadays becoming a very popular technique to study the performance of wireless channels. The reason is that it explicitly takes into account the joint effect of various parameters of the protocol stack as well as the characteristics of propagation environment. As a result, cross-layer modeling is a tool allowing one to quantify numerically the effect of various parameters of channel adaptation mechanisms on the performance provided to various applications.
The general approach to cross-layer performance modeling consists of three basic steps (i) wireless channel modeling (ii) cross-layer extension of the wireless channel model to the layer of interest (iii) performance evaluation at the layer of interest. Most cross-layer models proposed so far closely follow these steps. The basic idea of cross-layer wireless channel modeling is illustrated in Fig. 5, where black rectangles denote incorrectly received PDUs and grey rectangles stand for correctly received PDUs. In this example we assume that FEC is capable of correcting at most one incorrectly received bit and ARQ is not used at the data-link layer. Since no error correction procedures are defined at the IP layer, even a single lost frame within a packet leads to the loss of the whole packet. Following the cross-layer approach, performance metrics at the IP of higher layers can now be obtained as a function of the underlying layers parameters. Note that cross-layer extensions may differ in detail and complexity, especially when ARQ, AMC, and MIMO functionalities are also incorporated.

We note that cross-layer modeling techniques are out of the scope of this work. However, it is worth mentioning that due to the modular structure of most cross-layer models, the conventional performance evaluation models considered in this paper can be used at the last step of the cross-layer modeling procedure. The interested reader may refer to up to date reviews of cross-layer performance modeling and optimization techniques provided in [26,27].

3.5. Non-stationarity of wireless channel statistics

The cross-layer extension of the wireless channel model to the layer of interest affects the autocorrelation properties that may exist in the original SNR or bit error processes. To illustrate this let \( X(n), n = 0, 1, \ldots \), \( X(n) \in \{0, 1\} \), be the bit error process. Assume that frames are consecutively transmitted over the wireless channel and each frame consists of exactly \( m \) bits. It is easy to show that the process \( Y(i), i = 0, 1, \ldots \), \( Y(i) \in \{0, 1, \ldots, m\} \), defined as \( Y(i) = \sum_{j=n+i}^{n+i+m-1} X(j) \), \( m = 0, 1, \ldots \), describing the number of incorrectly received bits in a frame is conventionally supposed to quickly lose its correlational properties as \( m \) increases. Indeed, the length of a frame is almost always large enough compared to the channel coherence time such that \( \{Y(i), i = 0, 1, \ldots\} \) is uncorrelated. In this case the process describing the correct reception of frames is uncorrelated too. This property is used in many cross-layer modeling studies to relax the computational complexity of the performance evaluation model at the layer of interest. Practically, it means that instead of a queuing system with autocorrelated service process, one with iid service times can be used. We also note that this property should be even stronger when there are guard or synchronization intervals between the transmission of frames or when distributed channel access is used.

To provide more insight into the time-varying behavior of wireless channel characteristics, consider the cumulative sum (CUSUM) statistics of the SNR process observed on a IEEE 802.11b wireless channel (Fig. 6). CUSUM statistics is defined as

\[
C(i) = \begin{cases} 
E[X], & i = 0 \\
C(i - 1) + (X(i) - E[X]), & i = 1, 2, \ldots 
\end{cases} 
\]

where \( C(i) \) is the CUSUM value at the time \( i \), \( X(i) \) is the value of observation at the time \( i \), and \( E[X] \) is the average value of observations.

CUSUM statistics demonstrate how the sample mean varies in time. If during a period of time most of the values are greater than the mean of the whole trace then the CUSUM statistics are increasing. Therefore, a segment of the CUSUM statistics with positive slope indicates a period where the values tend to be above the mean. Similarly, a segment with negative slope corresponds to a period of time where the values tend to be below the mean of the trace. A sudden change in direction of the CUSUM statistics indicates a sudden change in the mean value. A period of time where CUSUM statistics remain relatively the same refers to a segment where the average value did not change.

CUSUM statistics of IEEE 802.11b observations gathered (11Mbps mode) and shown in Fig. 6 demonstrate the very special behavior of SNR traces. First of all, there are exceptionally long periods of time when SNR stays relatively the same and this behavior cannot be attributed to the channel coherence time. Moreover, there are long periods of time when wireless channel statistics constantly change in time. Note that the original SNR observations were averaged over (relatively long) 0.5 s intervals and further represented
by a single value. This observation implies that either the memory of the process is exceptionally high or some statistical parameters change in time.

Note that PDU error observations at the upper layers may become iid (as a result of segmentation and reassembly) when the initial SNR or bit error process is both covariance stationary and short-range dependent. However, as we already highlighted previously, there were indications that PDU error processes may not retain these properties. Particularly, in [24], the authors studied the statistical characteristics of frame error traces collected on a IEEE 802.11b wireless channel and revealed that the frame error process still has exceptionally strong (long) memory. A number of other studies reported similar properties of PDU error processes at layers, higher than physical (see e.g. [10]). This indicates that either the underlying SNR, symbol, or bit error processes at the physical layer have exceptionally strong memory or they are not stationary at all.

Most wireless channel modeling studies performed so far either implicitly or explicitly assumed covariance stationarity for channel observations, whilst providing no particular reason for that. The first who questioned this assumption were Konrad et al. [28]. The authors considered GSM bit error traces and highlighted their possible non-stationary behavior. They also proposed an algorithm to extract covariance stationary parts and demonstrated that isolated parts can be sufficiently well described using conventional Markov modeling techniques. Unfortunately, no clear theoretical basis for their segmentation algorithm was proposed. Their work was continued in [29], where the authors observed similar piecewise covariance stationary behavior of SNR observations measured on a IEEE 802.11b channel. They further used an exponentially-weighted moving average (EWMA) change-point statistical test to distinguish between covariance stationary parts. Their test checks whether the current channel statistics conform to the original hypothesis about covariance stationarity. This procedure is implemented using upper and lower control limits, \(E[L_{Y'}(n)] \pm C(n)\), where

\[
C(n) = k\sigma[\gamma](\frac{\frac{\gamma}{2} - \gamma}{\frac{1}{2} + \phi(n)(1 - \frac{\gamma}{2})}) = \frac{1 + \phi(1 - \gamma)}{1 - \phi(1 - \gamma)}.
\]

where EWMA statistics \(L_{Y'}(n), n = 0, 1, \ldots\) is defined as

\[
L_{Y'}(n) = \gamma Y(n) + (1 - \gamma)L_{Y'}(n - 1).
\]

\(\gamma\) is the smoothing parameter of EWMA statistics, \(k\) is a design parameter, \(\sigma[\gamma]\) is the variance of original observations, \(\phi(n)\) is the parameter of the AR(1) model used to model covariance stationary parts. See [29] for more details. In [30] this approach was extended to the case of bit and frame error statistics observed on an IEEE 802.11b wireless link.

Unfortunately, there are no simple yet effective methods to be statistically strict concluding whether a limited set of observations is covariance stationary or not. The reason for that is the gap between theoretical definition of this phenomenon and practical limitations. Dealing with just a few realizations of a process, non-stationarity can be attributed to local changes in the observed process. We also note that another plausible explanation for the complex behavior of PDU error statistics at the data-link and higher layers is indeed long-term memory. However, to date no studies have reported long-range dependence for SNR, symbol, or bit error observations. Observe that even when the wireless channel characteristics are indeed covariance stationary with long-range dependence, it is still required to better understand how wireless channel modeling needs to be used in performance evaluation and optimization studies. Indeed, in order to provide the best possible performance of information transmission at any given instant of time we are only interested in the local behavior of wireless channel statistics.

4. Arrival models

Accurate modeling of PDU arrival processes is of paramount importance in the context of any performance evaluation study. Modeling packet arrival processes in wired networks is an extensively studied research area that has attracted a lot of attention during the past two decades. We refer to [31,32] for a detailed review of traffic models conventionally used in wired networks. Although all those models can be inherited for performance evaluation purposes of wireless systems, there are some specific details that need to be taken into account when applying them in the wireless environment.

4.1. Real-time applications

Recall that conventional arrival models describe the traffic characteristics of a source at the IP or higher layers. When the
performance of a wireless channel is estimated at the IP or lower layer in terms of PDU loss and delay metrics, classic arrival models have to be extended to that layer. To identify a class of models that is best suited for our purposes let us consider the process of data segmentation in the protocol stacks. To effectively conceal the effect of incorrect reception of channel symbols, PDUs at lower layers are usually short in size and limited to hundreds of bits. These portions of data are often coupled with correction bits to constitute codewords/frames. This means that each IP packet is usually segmented into a number of equal length frames. Due to the constant length of frames used at the data-link layer, traffic models need to be discrete in nature with time slot duration, $\Delta$, given by the time required to transmit a single frame over a wireless channel. Therefore, when IP packet arrival models are extended to the lower layers, arrival of a single packet can be approximated by a batch arrival of frames. In performance modeling studies these batches are usually assumed to arrive at the end of time slots, as illustrated in Fig. 7. As we will see in what follows, these properties also affect the choice of performance evaluation model. It is also important to note that, depending on the layer of interest, more than a single segmentation procedure often needs to be performed.

To represent the traffic arrival process from real-time applications, such as voice-over-IP or video streaming, Markov models are often used. The most general discrete-time Markov model with batch arrivals is the discrete-time batch Markovian arrival process (D-BMAP, [33]). This process is the discrete-time analogue of the batch Markovian arrival process (BMAP), introduced by Lucantoni in [34], and includes many stochastic processes as its special cases. Due to its simple matrix notation and versatile properties D-BMAP provides an attractive option for stochastic modeling of traffic arrival processes and usage in performance evaluation studies.

Assume a discrete-time environment, i.e. the time axis is slotted, the slot duration is constant and given by $\Delta = (t_{i+1} - t_i), i = 0, 1, \ldots$ Consider the discrete-time homogeneous ergodic Markov chain $\{S(n), n = 0, 1, \ldots\}$ defined at the state space $S(n) \in \{1, 2, \ldots, M\}$. Let $D$ be its transition probability matrix. Let then $\{W(n), n = 0, 1, \ldots\}$ be a D-BMAP whose underlying Markov chain is $\{S(n), n = 0, 1, \ldots\}$. The value of $\{W(n), n = 0, 1, \ldots\}$ is said to be modulated by the discrete-time Markov process $\{S(n), n = 0, 1, \ldots\}, S(n) \in \{1, 2, \ldots, M\}$. We define D-BMAP as a sequence of matrices $D(k), k = 0, 1, \ldots$, each of which contains probabilities of transition from state to state with $k$ arrivals. It is easy to see that for each pair of states $i, j \in \{1, 2, \ldots, M\}$ the following

$$d_{ij}(k) = Pr(W(n) = k; S(n) = j | S(n - 1) = i),$$

for $k = 0, 1, \ldots$ are conditional PFs of D-BMAP.

Let the vector $G = (G_1, G_2, \ldots, G_M)$ be the mean vector of D-BMAP, where $G_i = \sum_{j=1}^{\infty} \sum_{k=1}^{\infty} d_{ij}(k), i = 1, 2, \ldots, M$. The mean process of D-BMAP is denoted by $\{W_G(n), n = 0, 1, \ldots\}$ with $W_G(n) = G_i$, when the Markov chain is in the state $i$ in the time slot $n$. The ACF of the mean process of D-BMAP is [33]

$$R_G(i) = \sum_{l=1}^{\infty} \phi_l^i \lambda_{l-1}, i = 1, 2, \ldots,$$

where $\lambda_i$ is the $i$ eigenvalue of $D$, $\tilde{\gamma}_l$ and $\tilde{h}_l$ are left and right eigenvectors of $D$, respectively, and $\tilde{\phi}$ is the vector of ones, $\tilde{\phi}$ is the steady-state probability vector of the underlying Markov chain. Note that ACFs of D-BMAP and its mean process are generally different [33].

Observing (7) it is easy to notice that the number of terms composing the ACF of the mean process of D-BMAP depends on the number of eigenvalues. The number of eigenvalues is a function of the number of states of the modulating Markov chain. Thus, varying the number of states we change the number of terms composing the ACF. Recall that it is also allowed to have different probability functions for each different pair of states, as shown in (6). These properties have been used in many studies to derive models of various traffic sources with sophisticated distributional and autocorrelational properties (see [35–37] among others). It is also important to note that D-BMAP is closed with respect to the superposition property, i.e. superposition of two D-BMAPs is again D-BMAP.

The generic D-BMAP is often too complex to use in practical studies. However, there are a number of special

![Fig. 7 - Batch arrivals at the data-link and physical layers.](image-url)
cases of D-BMAP that are important from the practical point of view. These special processes usually limit the scope of one or more characteristics of D-BMAP such that the process can be handled easily in applied studies.

When at most a single arrival is allowed in a slot, D-BMAP reduces to the discrete-time Markovian arrival process (D-MAP). D-MAP is completely defined by two matrices $D(0)$ and $D(1)$. Note that it is still allowed for this process to have different probabilities of arrival for each different pair of states. Furthermore, if D-BMAP is allowed to have probability functions that depend on the current state only, it reduces to the Markov modulated batch process (MMBP). In this case, matrices $D(k), k = 0, 1, \ldots$ have the same elements on each row. It is important that the mean process of MMBP still has ACF distributed according to (7), while the marginal distribution is a weighted sum of $M$ distributions with weighting coefficients given by steady-state probabilities of the underlying Markov chain. Another important case is obtained when the modulating Markov chain of D-BMAP is allowed to have two states only. Such processes have only a single geometrical term contributing to the ACF of the their mean processes. They are often referred to as switched arrival processes.

When each state of a two-state MMBP is associated with a Poisson distributed number of arrivals in a single slot, it reduces to a switched Poisson process (SPP). The ACF of a SPP is given by [7]

$$R_W(i) = R_C(i) + E[W] \delta_i = \begin{cases} 1 & i = 0, \\ 0 & i = 1, 2, \ldots \end{cases}$$

$$E[W] = E[G] = \pi_1 G_1 + \pi_2 G_2,$$  \hspace{1cm} (8)

where $R_C$ is given in (7), $G_1$ and $G_2$ are the mean arrival rates in states 1 and 2, respectively. The advantage of using SPP is that the additional interference caused by variance of conditional PFs in each state of the modulating Markov chain is known ($E[W] \delta_i$ in (7)) and can be explicitly taken into account in modeling algorithms.

In [7] the authors proposed a fast and elegant modeling algorithm for SPP. In order to completely parameterize the mean process of SPP, we must provide four parameters ($G_1, G_2, \alpha, \beta$), where $\alpha$ and $\beta$ are transition probabilities of the Markov modulating process from state 1 to state 2 and from state 2 to state 1, respectively, $G_1$ and $G_2$ are mean arrival rates in states 1 and 2, respectively. If we choose $G_1$ as a free variable, with constraint $G_1 < E[W]$, to satisfy $0 < \lambda < 1$, we can determine $G_2, \alpha,$ and $\beta$ from the next set of equations

$$G_2 = \frac{D[X]}{E[X] - G_1} + G_1$$

$$\alpha = \frac{(1 - Kx(1))(E[X] - G_1)}{G_2 - G_1}$$

$$\beta = \frac{(1 - Kx(1))(G_2 - E[X])}{G_2 - G_1}$$

where $X(n), n = 0, 1, \ldots, N$ are observations of the covariance stationary arrival process, $E[X], D[X],$ and $Kx(1)$ are the mean, variance, and lag-1 NACF value of $(X(n), n = 0, 1, \ldots, N)$. Parameters $E[X], D[X],$ and $Kx(1)$ need to be estimated from empirical data.

When each state of a two-state MMBP is associated with Bernoulli distributed number of arrivals in a single slot, it reduces to a switched Bernoulli process (SBP). Recall that ACFs of a SBP and its mean process are generally different due to a difference in $R_W(0)$ and $R_C(0)$. The special case of SBP is obtained by setting the probability of arrival in one of the states to 0. This process is characterized by the following properties

$$E[W] = E[G], \hspace{0.5cm} R_C(i) = R_W(i), \hspace{0.5cm} i = 0, 1, \ldots \hspace{0.5cm} (10)$$

where the former property holds for any D-BMAP while the latter is true for this special SBP only. Note that SBP can be used to model the PDU arrival process with at most one arrival per slot and NACF decaying according to a single geometrical term.

4.2. Non-real-time applications

Non-real-time applications usually use TCP at the transport layer. Nowadays, TCP accounts for 80%–90% of all the traffic in the Internet and there is no indication that these numbers will decline in the future [38]. Evaluating performance of TCP in wireless environment is then crucial for better understanding, dimensioning and deployment of modern networks.

Modeling of TCP traffic is an extremely complicated task. The main reason is that TCP uses a feedback mechanism to decide upon traffic injection to the network. Assuming a loss-free environment, behavior of a TCP sender is completely deterministic and dictated by the receive window of TCP and network/server capacity limitations. Losses occurring randomly introduce stochastic factors into TCP traffic behavior. The TCP traffic pattern is also affected by the type of application generating packets. For example, arrival traffic patterns for http and ftp applications may differ substantially.

As an example consider transmission of a file of infinite size. After the slow start phase the TCP sender eventually enters the congestion avoidance state and remains in this state until the end of the transfer. The window evolution of a single TCP connection in congestion avoidance phase can be described by the following system of equations

$$\begin{cases} W_{i+1} = W_i + 1, & I_{L,i} = 0 \\ W_{i+1} = W_i / 2, & I_{L,i} = 1 \end{cases}$$

where $I_{L,i}$ is the indicator of the event that a single or multiple packets are lost in the round i. Here, round is defined as the time between two consecutive loss events. Note that the way TCP source handles the loss event differs in different versions of TCP.

In principle, the packet generation process from a single TCP source described by (11) can be modeled using a Markov chain with $M$ states, as was demonstrated in [39]. Each state in that model corresponds to a TCP congestion window of a certain size (measured in packets) with state 1 corresponding to 1 packet, state $M$ corresponding to $M$ packets. Following this approach and introducing additional states to capture

1 Here and in what follows, we use the term "packet" to refer to a single PDU of TCP protocol instead of the term "segment" used conventionally. The reason is that in most cases exactly one segment is contained within a single packet.
other phases of TCP behavior (e.g. slow start) we may come up with a very accurate TCP model. However, in order to determine transition probabilities between the states of this model we need to know the IP packet loss probability. This probability is usually unknown in advance and needs to be measured in operational networks, making our modeling efforts obsolete. Moreover, this model assumes that the TCP connection of interest does not affect the packet loss probability. This may not hold in practice when the number of TCP connections sharing resources of a wireless channel is not sufficiently large. As a result, the TCP sending process is a complex feedback-driven system whose behavior cannot be modeled in isolation. However, as we will see in what follows, there are a number of approaches to formulate joint network/source models of TCP traffic.

5. Performance models for centralized access

5.1. Description

In a centralized wireless access environment there is a controller, often called a base station, that is responsible for request arbitration between mobile stations competing for transmission resources. In such systems a channel or a transmission opportunity is exclusively assigned to a mobile station that can fully utilize it for information transmission. The term channel is used here in a broad sense and may refer to a TDMA time slot, (W)CDMA code, or FDMA frequency. Examples of wireless technologies using centralized channel access include GSM, UMTS, IEEE 802.16.

In centralized wireless access there is no contention for resources between mobile stations at the information transmission phase, and the major performance degradation is caused by the error-prone nature of wireless channels and buffer overflow at the IP layer. As a result, performance evaluation models proposed to date mainly concentrate on the effect of various channel adaptation mechanisms. It is important to note that most models are still limited to simple performance metrics at the layer of interest, such as throughput and mean values of delay and loss processes. There are a number of reasons behind that. First of all, the joint loss process caused by imperfect error correction and buffer overflows is often too complicated to characterize. Secondly, usage of error correction complicates application and interpretation of analytical models. Due to these reasons, to evaluate the performance of applications in a wireless environment simulation studies are still often used.

The reason for initial failure of analytical performance modeling of wireless channels is that queuing theory, which is widely used for performance evaluation of wired networks, was not initially considered as an appropriate tool in the wireless domain. Observe that the service process of a wireless channel as well as the arrival process of PDUs at the layer of interest can be autocorrelated simultaneously. The queuing theory does not provide computationally efficient solutions when both arrival and service processes are not renewal at the same time. Recently, it was recognized that queuing theory still offers a number of attractive features such as priorities, batch arrivals, vacations, etc., that can be efficiently utilized for analytical performance modeling even in such a complicated environment. When formulated and solved properly, analytical models allow the saving of time required for actual performance evaluation.

5.2. Models for real-time applications

5.2.1. Models at the data-link layer

Performance evaluation models at the data-link layer have been the main focus of many studies published in the last two decades. The reason is that, following the TCP/IP stack philosophy, most wireless access technologies standardize two lower layers of the protocol stack. As a result, performance models at the data-link layer abstract operation of channel adaptation mechanisms implemented by these technologies, allowing one to characterize performance parameters provided to the IP or higher layers.

Conventional error correction techniques, such as FEC and ARQ, are crucial for satisfactory performance of wireless channels. Since they are now an integral part of many modern wireless access technologies, most of the studies published so far evaluate performance of various ARQ implementations. ARQ eliminates the influence of bit errors by allowing the retransmission of incorrectly received frames. To notify the sender about the erroneously received frame, ARQ protocols require a feedback channel. We distinguish between Stop-and-Wait (SW), Go-Back-N (GBN), and Selective Repeat (SR) ARQ schemes. According to SW–ARQ the source transmits a frame and then waits for an acknowledgement frame from the receiver. GBN ARQ is a scheme where frames are consecutively transmitted. When a frame is incorrectly received the receiver asks to retransmit all the frames starting from the incorrectly received one. According to the SR–ARQ scheme only incorrectly received frames are retransmitted. When the channel conditions are ‘bad’ ARQ may introduce significant delays. Depending on whether the number of retransmission attempts is limited or not we distinguish between non-persistent and persistent ARQ schemes, respectively. In the former case the number of retransmissions is limited for a single frame or for an IP packet.

FEC procedures use a proactive approach to eliminate the influence of bit errors in advance introducing error correction redundancy. This redundancy is exploited at the receiver to recover from bit errors. The advantage of FEC schemes is that they do not introduce retransmission delays allowing some information to be lost. A system implementing both FEC and ARQ is called a hybrid ARQ (HARQ). We distinguish between Type I and Type II HARQ systems. In the former case the FEC code applied to a frame remains constant for all retransmissions performed by ARQ. Type II HARQ allows for an increase of the FEC capability with each subsequent retransmission attempt. An important advantage of Type II HARQ is that it adapts to (possibly) changing wireless channel conditions and still does not need additional feedback from the opposite site of the link as the channel status is implicitly returned using negative or positive acknowledgements. Type II HARQ schemes are also known as incremental redundancy (IR) ARQ. Nowadays, Type II HARQ is extensively used in various wireless technologies, e.g. IEEE 802.16, UMTS.
There are two approaches to implement incremental redundancy in Type II HARQ schemes. According to the first approach the data block is only sent in the first transmission attempt. If this frame is incorrectly received subsequent retransmissions carry redundant information only. This information is then combined at the receiver with the first frame and further used to decode it. As a result, higher coding gain is achieved as the number of retransmissions increases. According to the second approach a replica of data is always transmitted in all subsequent frames. The major difference compared to the former approach is that the subsequent frames are self-decodable. This algorithm was specifically designed for those situations when the first frame is so severely damaged that it cannot be decoded by combining it with redundant information only. Chase combining is one of the ways to implement such a scheme. According to this, the same data are encoded to create two codewords of the same length. If the first transmission attempt fails, the second version is transmitted. At the receiving end it is combined with the incorrectly received codeword to increase the coding gain. If the decoding attempt fails, the second version is decoded in isolation. Note that the performance of both Type II HARQ schemes heavily depends on the choice of FEC codes.

From the modeling point of view there are two fundamental differences between Type I and Type II HARQ systems. For Type I HARQ the probability of incorrect frame reception in a single transmission attempt is the same for all retransmissions performed by the sender. Assuming that the probability to receive a frame incorrectly in a single transmission attempt is \( p_F \) for the Type I HARQ scheme the probability of the frame loss due to excessive amount of retransmission attempts allowed for a single frame is \( p_{RT} = p_F^r \), where \( r \) is the maximum number of retransmissions allowed for a single frame. In the Type II HARQ scheme each subsequent retransmission attempt carries redundancy bits that are added to already received frames to increase the error correction capability. Let \( p_{RT} \) be the frame error probability after the \( i \)th retransmission attempt. The frame loss probability is then \( p_{RT} \). Another important difference between Type I and Type II HARQ systems is that the length of frames in Type I HARQ systems is always constant. This property allows a simplified delay analysis expressing the frame transmission time in terms of the number of time slots. In the Type II HARQ system the length of frames sent by the sender may vary.

Analytical models based on renewal theory were the first comprehensive performance evaluation studies of wireless channels with centralized access. Throughput analysis of GBN Type I HARQ with reliable and unreliable feedback has been carried out in [40] and [41], respectively. In [42] the authors extended previous results to the case of delay-constrained communication. Results have been summarized in [20]. In those studies the error-prone nature of a wireless channel at the data-link layer was represented using two-state Markov models similar to the Gilbert one. The following frame error models were used for forward (F) and backward (B) directions

\[
D_F = \begin{pmatrix} p & q \\ r & s \end{pmatrix}, \quad D_B = \begin{pmatrix} a & b \\ c & d \end{pmatrix}. \tag{12}
\]

Fig. 8 ~ A part of the Markov model used in [40,41].

The approach taken by authors consists in defining two Markov processes tracking states of the wireless channel and ARQ protocol, respectively. The first process represents the frame transmission time measured in the number of time slots. The second one models the result of the frame transmission. These two processes are combined together, resulting in a two-dimensional Markov chain jointly describing the state of the protocol and the channel state. A part of the combined model is shown in Fig. 8, where state 1 corresponds to incorrect frame reception and correct feedback, state 2 to correct transmission of both frames, state 3 to incorrect reception of a frame and erroneous feedback, and state 4 to correct frame transmission and incorrect feedback. The number of correctly received data frames is counted by the number of visits to states 2 and 4. Since two models are combined together, each state is associated with a certain frame transmission time (one for states 2 and 4, \( m \) for state 1, and \( t \) for state 3) and each state is accessible from any other state with probabilities expressed via the frame error models given in (12). Frame transmission times corresponding to each transition are inferred from operation of the GBN ARQ protocol. The authors further considered an arbitrary state of the Markov model as the reference one and used renewal theory to deduce the steady-state average frame transmission time and throughput of the protocol. Here, a cycle is defined as the amount of time between two consecutive visits to the reference state. From the theory of Markov chains, the mean recurrence time of the state \( i \), \( E[D_{ij}] \), is defined as the average time between two consecutive visits to the state \( i \), and given by

\[
E[D_{ij}] = \frac{\sum_{i=1}^{N} \pi_i D_i}{\pi_i}. \tag{13}
\]

where \( \pi_i \) is the steady-state probability that the Markov process is in the state \( i \) and \( D_i \) is the mean sojourn time in the state \( i \). The latter term can be found as

\[
D_i = \sum_{j=1}^{N} P_{ij} D_{ij} \tag{14}
\]

where \( D_{ij} \) is the delay associated with transition from state \( i \) to state \( j \), \( P_{ij} \) is the corresponding transition probability of the Markov model.
According to the renewal theory, (13) and (14) hold for the reward function too. Particularly, if $R_{ij}$ is the number of acknowledged transmissions associated with the transition from state i to state j, we have

$$E[R_{ij}] = \sum_{i=1}^{N} \frac{\pi_i R_i}{\pi_i} = \sum_{i=1}^{N} \frac{\pi_i R_i}{\pi_i},$$

where $E[R_{ij}]$ is the average number of correct frame receptions during a cycle.

Finally, applying the fundamental theorem of the renewal reward process we get the throughput of the system as

$$\lim_{r \to \infty} \frac{R(r)}{r} = \frac{E[R_{ij}]}{E[D_{ij}]} = \sum_{i=1}^{N} \frac{\pi_i R_i}{\pi_i} = \sum_{i=1}^{N} \pi_i D_i,$$

where $R(r)$ is the total number of correct frame receptions by time $r$. Note that the probabilities $\pi_i$ cancel out and the result becomes independent of the choice of the reference state. Among other conclusions, authors demonstrated that autocorrelation of the frame error process positively affects the throughput and delay characteristics of the Type I HARQ system. The described approach is general enough and can be used to analyze a wide variety of ARQ protocols including both SW–ARQ and SR–ARQ. Although minor modifications have been proposed by many authors the basic algorithm applied to renewal theory remains unchanged.

Although most of the data-link layer performance models concentrated on Type I HARQ schemes there are a number of studies that explicitly addressed Type II HARQ systems. A model with chase-combining, persistent and non-persistent SR operation of a Type II HARQ system, non-negligible constant round-trip time (RTT), and multiple segmentation and reassembly procedures in the protocol stack has been proposed in [43]. The FEC code was assumed to be of the maximum distance separable (MDS) type. The wireless channel was modeled using Gilbert’s model with good (G) and bad (B) states. Since the main emphasis of the work was put on taking into account an exceptionally strong memory of a wireless channel it was also assumed that the feedback is immediate and fully reliable. It is also important to note that frame sizes were assumed to be of the same length. Particularly, their model involves a Markov chain with $N$ macrostates, each of which corresponds to a certain retransmission attempt. A macrostate $i$, $i = 1, 2, \ldots, N$ consists of $(i - 1)^2$ states denoted as $(p_1, c_1, c_2, \ldots, c_i)$, $i = 1, 2, \ldots$. Each $c_i$ denotes the states of the channel model at the $i$th retransmission attempt. Observe that this model explicitly tracks the state of all retransmissions that are required to transmit a single frame assuming that the channel statistics at the data-link layer is autocorrelated. Nevertheless, the memory of the wireless channel model is still chopped at $N$. Transition probabilities between states of the model can be directly obtained by collecting wireless channel statistics of the data-link layer. However, in [43] the authors derived them using the cross-layer framework. In both cases this procedure is non-trivial when $N$ is sufficiently large. This model was further extended to the case of non-persistent operation in [44], where a frame is considered to be lost when the accumulated number of bit errors is greater than the error correction threshold. In [44], the authors also investigated the delay performance of a Type II HARQ scheme and the effect of correlation of the wireless channel characteristics.

The performance of the Type II HARQ has been studied by Baidia et al. in [45], where the authors assumed SR operation of the ARQ mechanism and Reed–Solomon erasure block FEC codes. The system time is slotted and the slot duration equals the amount of time to perform a single frame transmission attempt. The RTT is constant and equals $m$ time slots. The major contribution of the paper is that a Markov model is used to describe the number of bit errors occurring up to $i$th retransmission attempt. The state vector of the system is defined as $\vec{v} = (l_1, r_1), \ldots, (l_i, r_i), \ldots, (l_m, r_m)$. Here $C \in \{0, 1, \ldots, M - 1\}$ denotes the state of the $M$-state HMM modeling the frame error process. Each state of $C$ is associated with cumulative number of bit errors occurring up to $i$th retransmission attempt and can be computed using the bit error model. The rest of $\vec{v}$ is the array of size $m$ of ordered pairs $(l_i, r_i)$ describing the outcome of the last $m$ retransmissions, where $r_i$ is the number of HARQ retransmissions performed for a frame, $l_i$ is the accumulated error level up to the retransmission i. Since the model is Markov in nature, classic performance measures such as throughput and frame delay can be obtained using renewal theory arguments. However, it is easy to see that the computational efforts associated with solution of this model are significant. This technique has been recently used to obtain performance metrics of a Type I ARQ system operating in SR regime [46].

Note that the approaches proposed in [41,20] are examples of the few models where unreliable feedback as well as non-zero propagation time were assumed. In recent studies these properties are often neglected. Observe that assuming zero propagation time and fully reliable feedback operation then GBN-, SW-, and SR-ARQ systems become identical. Sometimes this simplified operation is referred to as “ideal” ARQ [47]. We discuss the validity of these assumptions at the end of this subsection. Assuming ideal operation delay and throughput analysis of Type I and Type II HARQ algorithms is much easier to perform. Since results for ideal operation of HARQ systems have been addressed in many studies we do not specifically cite all of them. Instead, we briefly summarize the findings of those studies below.

Consider a Type I HARQ system. Recalling that the length of all frames in a Type I HARQ system is constant the frame transmission time can be expressed in terms of time slots. Let $S, S \in \{1, 2, \ldots\}$ be a RV describing the frame transmission time and let $S_i = Pr(S = i), i = 1, 2, \ldots$ be its probability function (PF). If the number of retransmission attempts allowed for a single frame is unlimited and frame errors occur independently, the service time distribution of a frame is given by

$$s_i = p_i^{i-1}(1 - p_f), i = 1, 2, \ldots \quad (17)$$

where $p_f$ is the probability of incorrect reception of a frame in a single transmission attempt.

When the number of retransmissions is limited by $r$ a certain frame can be lost as a result an excessive number
of retransmission attempts. In this case the service time distribution, \( s_i \), \( i = 1, 2, \ldots, r \), is obtained as

\[
s_i = \begin{cases} 
  p_F^{i-1}(1 - p_F), & i = 1, 2, \ldots, r - 1 \\
  1 - \sum_{j=1}^{r-1} p_F^{i-1}(1 - p_F), & i = r. 
\end{cases}
\]  

(18)

In (18) the term corresponding to \( i = r \) takes into account the transmission time of the incorrectly received frames. When the service time distribution of successfully received frames is of interest (18) needs to be modified accordingly. Let \( L \) be the event of the frame loss and let \( l_i \) be the indicator of this event, i.e., \( l_i = 1 \) when a frame is lost and \( l_i = 0 \) otherwise. Let further \( s_c \) be a RV describing the frame transmission time of correctly received frames only and let \( s_{i,c} = Pr(s_c = i|l_i = 0) \) be its PF. Using conditional probabilities it can be found as follows

\[
s_{i,c} = \frac{p_F^{i-1}(1 - p_F)}{1 - \sum_{j=1}^{i-1} p_F^{j-1}(1 - p_F)}, \quad i = 1, 2, \ldots, r. 
\]  

(19)

When an ideal Type II HARQ is considered both the frame error probability and the time required to perform transmission attempt of a single frame are no longer constant. In this case the frame transmission time needs to be measured in seconds. Let \( T_j = \sum_{j=1}^{i} T_j \), \( j = 0, 1, \ldots, r \) be the time series representing all possible frame transmission times, where \( T_j, i = 1, 2, \ldots, r, \) is the time required to perform the \( i \)th retransmission attempt. Let further \( H \), \( H \in (D_1, D_2, \ldots, D_r) \) be a RV describing the frame transmission time and \( h_i = Pr(H = D_i), i = 1, 2, \ldots, r, \) be its probability function (PF). Assuming that successive frame error probabilities are independent of each other we have for Type II HARQ system

\[
h_i = \begin{cases} 
  p_{F,1} - p_{F,i-1}, & i = 1, 2, \ldots, r - 1 \\
  1 - \sum_{j=1}^{r-1} (p_{F,1} - p_{F,i-1}), & i = r. 
\end{cases}
\]  

(20)

where \( p_{F,1}, i = 1, 2, \ldots, r - 1, \) are probabilities of correct frame reception after the \( i \)th retransmission and \( p_{F,0} = 0 \) by default.

When the service time distribution of successfully received frames is of interest we need to modify (20). Letting \( l_i \) be the indicator of the frame loss we introduce RV \( H_c, H_c \in (D_1, D_2, \ldots, D_r) \) describing the frame transmission time of correctly received frames only and its PF \( h_{i,c} = Pr(H_c = D_i|l_i = 0), i = 1, 2, \ldots, r \). We have

\[
h_{i,c} = \frac{p_{F,1} - p_{F,i-1}}{1 - \sum_{j=1}^{r-1} (p_{F,1} - p_{F,i-1})}, \quad i = 1, 2, \ldots, r. 
\]  

(21)

When the frame error process is autocorrelated and represented using a Markov model the service time distribution can be expressed using matrix notation. When the number of retransmission attempts allowed for a single frame is unlimited the frame service time distribution for the Type I HARQ system is given by

\[
s_i = \tilde{\pi}_F D_F^{i-1}(1)D_F(0)\tilde{\epsilon}, \quad i = 1, 2, \ldots. 
\]  

(22)

where \( \tilde{\pi}_F \) is the steady-state distribution of the channel model, \( \tilde{\epsilon} \) is the vector of ones of appropriate size, \( D_F(0) \) and \( D_F(1) \) are transition probability matrices of the frame error process with correct and incorrect frame reception, respectively.

When the number of retransmission attempts is limited to \( r \) we have for a Type I HARQ system

\[
s_i = \begin{cases} 
  \tilde{\pi}_F D_F^{i-1}(1)D_F(0)\tilde{\epsilon}, & i = 1, 2, \ldots, r - 1 \\
  1 - \sum_{j=0}^{r-1} \tilde{\pi}_F D_F^{r-j}(1)D_F(0)\tilde{\epsilon}, & i = r. 
\end{cases}
\]  

(23)

Note that the service time distribution taking into account service times of correctly received frames only can be obtained similarly to (19) replacing scalars by matrix expressions. For Type II HARQ systems such simple expressions for the service time distribution of a single frame are not available. The underlying reason is the difficulty of channel modeling. Recall that classic Markov models of frame error processes assume that all frames are of the same size. Since in our case frame sizes are no longer constant and the wireless channel characteristics are autocorrelated in nature such models become obsolete. As a result, for Type II HARQ systems, models similar to that presented in [45] need to be used. However, observe that a rough approximation of the service time distribution can be obtained by assuming that all frames are of the same length. This approach was taken in [43].

Note that previously we assumed that frames are transmitted back-to-back over the wireless channel. When RTT is known and can be expressed (or, at least, approximated) in terms of the (integer) number of frame transmission intervals the above mentioned expressions for \( s_i \) and \( s_{i,c} \) need to be modified. Assume that non-zero RTT straightforward delay and throughput analysis is feasible for SW operation of both HARQ I and HARQ II systems. For the latter system we should also assume that all frames are of the same length. For example, assuming uncorrelated channel conditions the number of retransmission till successful frame reception, \( s_{i,1} = 1, 2, \ldots, \) for Type I HARQ system is geometrically distributed according to (17) and (18), (19) for unlimited and limited number of retransmission attempts allowed for a single frame, respectively. The only difference compared to ideal HARQ operation is that the amount of time between transmission of a frame and reception of the feedback now takes a certain (fixed) number of time slots. Note that for a Type II HARQ system the RTTs of frames are different and we need to use (20) and (21).

When wireless channel characteristics are correlated and represented using a Markov model, analysis is more complicated. The reason is that we should now take into account the time evolution of the channel model in between two transmission attempts. For example, if it takes exactly \( j \) frame transmission intervals to receive a feedback from the receiver then the frame service time distribution for SW operation of Type I persistent HARQ system with non-zero RTT can be found as

\[
s_i = \tilde{\pi}_F D_F(1)D_F(1)\tilde{\epsilon}, \quad i = 1, j + 2, 2j + 3, \ldots. 
\]  

(24)

where the number of \( D_F(1) \) terms needs to be exactly \( (i - 1) \) in order for \( s_i \) to take on \( i \) and \( D = D_F(0) + D_F(1) \).

Observe that with the increase of RTT the channel process loses its memory and subsequent frame retransmissions become uncorrelated. In this case we can approximate (24)
using (17). It is important to note that the described approach cannot be applied to study the throughput of GBN and SR operation of HARQ I system for both correlated and uncorrelated wireless channel conditions. In these cases, models similar to those introduced by Zorzi and Rao in [40,41] need to be used. When performance of a Type II HARQ system with non-zero RTT is of interest models similar to those proposed in [45,43] need to be used.

Finally, observe that throughout of Type I and Type II HARQ system is independent of the choice of the frame error model. For both types of system it is given by \( k/E(S_c) \), where \( k \) is the number of data bits in a frame and \( E(S_c) \) is the mean transmission time of successfully received frames.

Most approaches cited above use rather simple traffic models assuming either saturation conditions or renewal arrival processes. On the contrary, arrival processes from voice and video traffic sources are often characterized by complex distributional and autocorrelation properties that may significantly affect the performance provided by wireless channels. Furthermore, those models completely neglect the effect of buffering at the data-link layer. Indeed, under certain wireless channel and traffic conditions queuing delays and packet losses caused by buffer overflows may significantly contribute to the overall performance degradation. As a result, performance models based on the renewal theory may not always provide adequate representation of the frame transmission performance.

Nowadays, models based on the queuing theory are the most popular performance models for wireless channels with centralized access schemes. The straightforward way to represent the frame transmission process over a wireless channel is to use the \( G_A/G_S/1/K \) queuing system, where \( G_A \) is the frame arrival process, \( G_S \) is the service process of a wireless channel, and \( K \) is the capacity of the system. Here, the service process is defined as the times required to transmit successive frames over the wireless channel. Characteristics of this process are determined by the frame error process and error correction schemes of the data-link layer. Observe that this model explicitly captures the effect of buffering at the data-link layer. However, there are important obstacles that do not allow straightforward application of the \( G_A/G_S/1/K \) model. First of all, we need to obtain the service time distribution of frames at the wireless channel. In the case of a non-persistent ARQ system this time needs to account for the service time of those frames which are eventually lost as a result of an excessive number of retransmission attempts. Indeed, it does not matter for us whether a frame is successfully transmitted or dropped. In both cases it induces some waiting time for other frames waiting in a queue. Secondly, it is known that both the inter-arrival time of frames and transmission time of frames till successful reception may not be independent. If we allow both processes to be autocorrelated it would make the analysis of the \( G_A/G_S/1/K \) queuing system quite a complex task even when arrival and service processes can be accurately modeled by Markovian processes. Indeed, the theoretical background of queuing systems with autocorrelated arrival and service processes is not well-studied and only few contributions are available so far. Among a few others, one should mention the BMAP/S/1 queuing system and some modifications considered in [48–50]. Analysis of these systems is more computationally intensive compared to queuing systems with renewal service processes and often involves finding steady-state probabilities of Markov chains of high dimensions. However, as we already discussed, even when wireless channel characteristics are autocorrelated at the physical layer it does not necessarily imply that the frame error process is autocorrelated too.

The basic idea of the \( G_A/G_S/1/K \) model is shown in Fig. 9. Since frames are often of fixed length time in this system is slotted. The slot duration, \( \Delta_F \), equals the time to perform a single transmission attempt of a frame. Depending on the type of the wireless channel model and ARQ operation (persistent or non-persistent) the service time distribution is obtained using the frame error process, as discussed above. Given the service time distribution and a certain frame arrival process the \( G_A/G_S/1/K \) model can be further solved using conventional methods of queuing theory, e.g. the imbedded Markov chain approach. It is important to note that in the case of autocorrelated frame error the process distribution of the frame transmission time may not be sufficient to accurately describe the process of frame transmissions over a wireless channel. The reason is that successive frame transmission times may still be correlated. Also observe that the assumptions of immediate and fully reliable feedback are not required for successful application of the \( G_A/G_S/1/K \) model. Finally, note that the \( G_A/G_S/1/K \) model described above is not limited to centralized channel access. When the service time of a frame is available it can also be used for the distributed access environment as we will discuss in the next section.

The \( G_A/G_S/1/K \) model and its variants have been used in many studies. To model the process of frame transmissions the authors in [51] proposed to use a D-BMAP/PH/1 queuing
system. To capture the effect of persistent ARQ operation, they assumed that the service time of any frame is distributed according to a discrete phase-type (PH) distribution and service times of successive frames are independent. Recall that the PH distribution is dense in the class of realvalued distribution, i.e. can be used to model any service time distribution. However, fitting to statistical data requires additional efforts. Also observe that their model is only appropriate when the wireless channel conditions are relatively ‘good’ or the data-link layer is completely reliable. Finally, assumption of an infinite number of waiting positions may not be adequate, especially for a mobile station. In [52] the authors relaxed the latter assumption proposing to use a finite-capacity D-BMAP/G/1/K queuing system. To avoid a complex fitting procedure generally distributed uncorrelated service times were used. To the best of the author’s knowledge this was the first study where frame losses were allowed to occur due to both an excessive number of retransmission attempts allowed for a single frame and buffer overflow at the IP layer.

In [47], to study the delay performance of applications at the data-link layer in the presence of SR-ARQ, a fluid-flow queuing-analytic model was adopted. In [53] the authors extended their approach to the case of a Type I HARQ system. In those studies both traffic and error processes were allowed to be autocorrelated. The arrival process was modeled by an ON-OFF Markovian model. It should be stressed that the fluid-flow approximation usually provides good results for very high-speed links when the influence of single events (arrivals) has negligible impact on the overall performance of the system of interest. However, wireless channels do not always provide exceptionally high data rates. In this case single arrivals are of higher importance and point processes should be used instead.

Another approach to estimate performance parameters at the data-link layer is to treat the frame error process as another arrival process to the queuing system, as shown in Fig. 10. The queuing model can be represented as a non-preemptive priority $G_E + G_A/D/1/K$ queuing system, where $G_E$ is the “error arrival” process and $G_A$ is the frame arrival process. The “error arrival” process is independent of the arrival process and obtained as “one-to-one” mapping from the frame error process, i.e. whenever a frame is incorrectly received in the slot $n$, there is an arrival from the “error arrival” process. This process has absolute priority over the frame arrival process, meaning that whenever an error “arrives” it is always served first, “producing” a waiting time for normal arrivals. Indeed, since the frame at the head of the queue must wait till the error burst is completely served the presence of ARQ is implicitly taken into account. Note that sometimes this model results in less complicated analysis compared to the $G_A/G_S/I/K$ model. The reason behind this is the constant service times of unit duration. For example, some special systems of the $G_E + G_A/D/1/K$ type allow rather simple and straightforward algebraic solutions. Another inherent advantage of this model is that the frame error process is allowed to be correlated when needed and this does not substantially increase the complexity of analysis. Using the $G_E + G_A/D/1/K$ model the operation of an ideal Type I HARQ system can be approximated. Particularly, the described approach was used in [54,19] applying the $D$-MAP$_E + D$-BMAP$_A/D/1/K$ queuing system, where $D$-MAP$_E$ is the error arrival process and $D$-BMAP$_A$ is the frame arrival process. The solution of the queuing problem is based on the imbedded Markov chain approach, resulting in a two-dimensional Markov chain describing the queuing system at equilibrium [55,54]. In [56] the authors extended their model such that a wireless channel can now be modeled at the physical layer using SNR or bit error processes.

Let us now summarize and discuss those simplifications often taken in performance evaluation studies of ARQ protocols. They include (i) immediate feedback, (ii) a fully reliable feedback channel, and (iii) negligible probability of undetected error. The first assumption implies that the feedback from the receiver is returned immediately after the frame is sent. Observe that the coverage range of modern wireless access technologies making use of ARQ mechanisms is small and limited to a few kilometers only. Taking into account the relatively high raw rates of these technologies and neglecting the processing delay caused by decoding it is natural to expect that the feedback is almost instantaneous. A certain channel organization at the air interface may favor this assumption too, e.g. time division duplex (TDD). However, when scheduling is used at the lower layers (see e.g. UMTS, HSDPA) this assumption may not be justified. Fully reliable feedback is another assumption commonly taken in performance modeling of ARQ protocols. The reason to assume it is that the loss of positive or negative acknowledgements (ACK, NACK) has a very profound effect on ARQ protocols’ behavior invoking their timeout-based behavior and significantly complicating their analytical modeling. The rationale behind this assumption is that ARQ
(N)ACKs are small in size and well-protected by the FEC code. Unfortunately, this may not be true in practice. Next, it is common in ARQ modeling to assume that the probability of undetected error is negligible. Indeed, for most FEC codes the probability of undetected error exponentially decreases as the size of the codeword increases. While all above mentioned assumptions do look reliable for modern wireless access technologies, there are no studies explicitly evaluating how accurate they are. However, it is important to note that all these assumptions are not mandatory for analytical modeling and can be relaxed when needed.

There are also a number of assumptions about the systems authors have studied so far. First of all, most performance evaluation studies either explicitly or implicitly assume that the FEC code is applied to the basic operational unit of an ARQ protocol, i.e. frame. This may not always be true in practice. In some wireless access technologies the ARQ frame is often further divided into shorter data blocks to which the FEC code is applied to. Note that this assumption does not necessarily limit the scope of performance evaluation models and a particular model can often be modified to take this into account. Secondly, a single ARQ system is often assumed. In practice, multiple HARQ systems may co-exist at different layers of the protocol stack. In MIMO systems a number of ARQ instances may run in parallel, transmitting frames belonging to the same packet. These ARQ instances may interfere with each other affecting the performance of the whole system. Thus, additional care needs to be taken when making conclusions about the performance of a certain protocol. Very often numerical results presented by authors should only be interpreted as qualitative insights. Practically, each and every model needs to be modified to be suitable for analysis of real systems.

5.2.2. Models at the IP layer

As most modern wireless access technologies are expected to be an integral part of the IP-based Internet it is natural to define models at the IP layer, where performance metrics are measured and standardized. Although data-link layer models often give very valuable information about the performance provided by wireless technologies they are often insufficient to fully describe it. Indeed, the focus of most data-link layer studies is on throughput of HARQ schemes. While this is undoubtedly an important performance metric other parameters such as packet transmission time and packet loss probability (in the case of non-persistent ARQ) are of high practical importance. Note that these metrics affect the queuing process at the higher layers. Indeed, depending on the packet generation rate and packet transmission time, the buffer size must be designed such that the trade-off between queuing losses and delays is the best for a particular application. This can hardly be achieved using data-link layer performance models. Even when delay and loss characteristics are estimated at the data-link layer it is easy to observe that they do not take into account segmentation and reassembly procedures between the data-link and IP layers. Since no error correction is defined at the IP layer a single incorrectly received frame may lead to the loss of the whole packet this frame belong to. As a result, data-link layer performance parameters need to be appropriately extended to the IP layer, or IP layer performance models need to be directly solved. Another important shortcoming of data-link layer metrics is that they do not take into account the effect of buffering at the IP layer. Finally, it is also worth mentioning that most of the studies estimate throughput of HARQ schemes using simplified assumptions about the traffic arrival process at the station of interest.

Queuing theory is nowadays widely used to estimate performance of wireless channels at the IP layer. For this purpose the $G_A/G_e/1/K$ framework is often utilized. The input parameters for this model are the IP packet arrival process and the packet service times. The latter need to be described in terms of the amount of time required to deliver a single IP packet including all retransmission delays caused by incorrect reception of frames at the data-link layer. It can be obtained by directly observing packet transmission times or analytically using the cross-layer approach. Observe that in the case of imperfect error correction at the data-link layer the time spent transmitting those packets that are eventually lost needs to be taken into account when computing the packet service time distribution.

The authors in [52] used the D-BMAP/G/1/K queuing model to represent the packet transmission process over a wireless channel. It is important to note that the packet service time distribution is more complicated to derive compared to the service time distribution of a single frame. However, it is still possible under certain assumptions. Following [52], assume immediate and fully reliable feedback, $p_f$ as the probability of incorrect frame reception in a single transmission attempt, and consider the IP packet service time distribution when a persistent Type I HARQ system is used at the data-link layer. Let $g(k)$, $k = v, v+1, \ldots$ be the service time distribution of a single frame measured in frame transmission attempts, where $v$ is the number of frames a single IP packet consists of. Further assume that the frame transmission process is described using an $M$-state Markov model $\{W_F(n), n = 0, \ldots\}$ with $D_T(0)$ and $D_T(1)$ describing the transition probabilities with correct and incorrect frame reception, respectively.

Let $D_N(i, k)$ be the set of matrices containing transition probabilities from state $i$ to state $j$ of the Markov chain $\{S_F(n), n = 0, 1, \ldots\}$ with exactly $k$, $k = 0, 1, \ldots, i$, incorrectly received frames in a frame pattern of length $i$, $i = v, v+1, \ldots$. Setting $D_N(1, k) = D_T(k)$, $k = 0, 1$, we find $D_N(i, k)$ recursively using $D_T(k)$, $k = 0, 1$, as follows

$$D_N(i+1, k) = \begin{cases} D_N(i, k)D_T(0) + D_N(i, k-1)D_T(1), & k \neq 0, \\ D_N(i, k)D_T(0), & k = 0. \end{cases}$$

(25)

Observing that the successful packet transmission is only possible when the last frame is also successfully transmitted the result for $g(k)$ is given by

$$g(v) = \mathbb{P}_2 D_N(u, 0\epsilon),$$

$$g(k) = \mathbb{P}_2 D_N(k - 1, k - u\epsilon) D_T(0\epsilon). \quad k = v + 1, v + 2, \ldots$$

(26)

Assume now a non-persistent Type I HARQ system, where the number of frame retransmission attempts allowed for a single IP packet is limited to $r$. The packet transmission time $d(k)$, $k = v, v+1, \ldots, r$ can be obtained using $g(k)$, $k = v, v+1, \ldots$ derived previously in (26). Let $A$ be the event of successful packet transmission and $I_A$ be its indicator, i.e. $I_A = 1$ when
the packet is successfully transmitted and \( I_A = 0 \) otherwise. PF of the packet transmission time can be obtained as follows

\[
d(k) = c(k) + b(k), \quad k = u, v + 1, \ldots, r.
\]  

(27)

where \( c(k) = \Pr(D = k, I_A = 1) \) is the probability of delay on \( k \) slots induced by successful packet transmission, \( b(k) = \Pr(D = k, I_A = 0) \) is the probability of delay on \( k \) slots induced by unsuccessful packet transmission.

Observe that in the non-persistent case all those transmission times that are larger than \( r \) in (26) lead to the loss of a packet. Therefore, we have the following for the packet loss probability

\[
p_L = \sum_{i=1}^{r} g(i).
\]

(28)

Consider now the term \( b(k) \) in (27). As one may observe \( b(k) \) is only defined for \( k = r \). Indeed, transmission times of all those packets that are incorrectly received contribute to \( b(r) \) only. Packets that are correctly received, i.e. require less than \( (r+1) \) transmission slots to be completely received, contribute to \( c(k) \), \( k = u, v + 1, \ldots, r \). Therefore, using conditional probabilities PFs \( c(k) \) and \( b(k) \) can be found using (29) as follows

\[
c(k) = g(k), \quad k = u, v + 1, \ldots, r \quad b(r) = \frac{g(r)}{1 - \sum_{i=u}^{r} g(i)}
\]

Previously, we presented an algorithm that can be used to model the process of segmentation and reassembly with the persistent and non-persistent Type I HARQ system enabled. If multiple HARQ systems are defined for a particular technology this algorithm can be applied successively to estimate performance metrics provided to the IP layer. As an example, consider a situation when two HARQ systems are implemented at the data-link layer. Let us denote them using \( R_1 \) and \( R_2 \). According to this setup an IP packet is first segmented into a number of frames by the \( R_1 \) HARQ system. These frames have added correction bits and are then dispatched to the \( R_2 \) system one after another. \( R_2 \) also segments frames into smaller pieces of data (we call them codewords here), then channel coding FEC bits are added to each and every codeword. These codewords are further consecutively transmitted. Taking the bottom-up approach we may first analyze the performance provided by the \( R_2 \) HARQ system to the \( R_1 \) system using the algorithm described in this section. As a result, we get the frame error probability and PF of the frame transmission time. These parameters serve as an input to analysis of the \( R_1 \) HARQ system. Exploiting the same framework we get the IP packet loss probability and PF of the time required to transmit a single IP packet.

Instead of limiting the maximum transmission time for a single IP packet, HARQ systems sometimes limit the number of retransmissions allowed for each frame in a packet. According to it, once a frame is unsuccessfully received in \( r \) retransmission attempts the whole packet to which this frame belongs to is dropped. The rationale behind this operation is that the physical connection is likely to be lost whenever \( r \) successive retransmission attempts fail. Assuming that successive frame transmissions are independent, the frame loss probability is \( p_F, \) where \( p_F \) is the probability of incorrect frame reception in a single transmission attempt. The packet loss probability is then obtained as \( p_L = 1 - (1 - p_F^r)^r, \) where \( r \) is the number of frames in a packet. Finally, observe that the IP packet transmission time can be obtained using \( d(k) = c(k) + b(k), \) as in (27), where \( c(k) \) and \( d(k) \) are PFs of delay on \( k \) slots given that the packet is successfully or unsuccessfully received. However, the component \( c(k) \) is now conditioned on the event of successful frame receptions up to the \( k \)th frame. It can be found by estimating \( i \)-times convolutions of the frame transmission time. The easiest way is to do it recursively using the frame transmission time conditioned on successful frame reception. Component \( b(k) \) can be inferred from \( c(k) \) by observing that the probability to lose a packet as a result of incorrect reception of the \( i \)th frame is \( 1 - p_F^{i-1} p_F \) and incorrect reception of \( i \)th frame adds exactly \( r \) time slots to the delay caused by successful transmission of first \( (i-1) \) frames.

The proposed framework can also be extended to the case of the Type II HARQ system. Assume that in a Type II HARQ system retransmissions carry redundancy bits only which are added to the original codeword at the receiver, increasing the probability of successful frame reception. To proceed further we also need to assume that all frames are of the same size. Recall that the Type II HARQ system is non-persistent in nature, where the number of retransmission attempts allowed for a single frame is always limited. The only modification required to capture its operation is to recompute the frame and packet loss probabilities. Let \( p_{F,i} \) be the probability of incorrect frame reception after the \( i \)th retransmission attempt. The frame loss probability is then \( p_{F,r}, \) where \( r \) is the maximum number of retransmission attempts. Thus, the packet loss probability is \( p_L = 1 - (1 - p_{F,r})^r. \) Finally, observing that the packet loss probability caused by an excessive number of retransmission attempts made for \( i \)th frame is geometrically distributed with parameter \( p_{F,i}, \) the packet transmission time can be found using conditional PFs, as explained above for the Type I HARQ system.

The proposed framework can also be used when a single packet is transmitted over a number of channels using a multiple-in multiple-out (MIMO) transmission system. Assuming that channels are independent of each other they can be modeled using separate frame error models. Further, delay and loss analysis can be performed independently for each channel. Assume that a single packet is divided into \( a \) subpackets each of which contains the same number of frames, except for (perhaps) the last subpacket. These subpackets are further transmitted over individual channels. The packet loss probability can be obtained as

\[
p_L = 1 - \prod_{i=1}^{a} (1 - h_{L,i}),
\]

(30)

where \( h_{L,i}, i = 1, 2, \ldots, a \) are subpacket error probabilities estimated as explained previously. The PF of the time required to transmit a single subpacket over the \( i \)th channel, \( d_i(k), \) can also be obtained using those approaches described in this section. Finally, the time required to transmit a single IP packet over \( a \) channels is given by \( \max_{i \in \{1, 2, \ldots, a\}} d_i(k), \).
Note that other applications of MIMO antenna design, e.g. space–time block coding (STBC) can also be analyzed.

We also note that in most studies it is assumed that all IP packets are of the same size. This assumption is too restrictive for a broadband wireless channel where a number of connections having different packet sizes are multiplexed. In this case we may proceed as follows. Assume that IP packets of M bytes in length arrive with probability \( a_M \). Obviously, the packet loss probability and IP packet transmission time depends on the number of frames a single IP packet is segmented to. Using the approach discussed previously we firstly get the packet service time distributions \( d_M(k) \) and packet loss probabilities for each size \( M, p_{L,M} \). Finally, weighting these quantities with corresponding packet size probabilities we obtain the “averaged” metrics of interest as follows

\[
p_L = \sum_{VM} p_{L,M} a_M, \quad d_k = \sum_{VM} d_M(k) a_M, \quad k = 1, 2, \ldots \tag{31}
\]

Observe that the approach described above works well when inter-arrival times between successive IP packets are not correlated. When arrivals are correlated the analysis is more complicated and requires the exact knowledge of the packet arrival process.

To increase the throughput of HARQ systems operating in the SW regime a number of ARQ instances are sometimes implemented in parallel. This is especially beneficial for wireless channels with high or varying RTT, where waiting periods are filled with frame transmissions from other ARQ instances. Unfortunately, the author is unaware of studies investigating this implementation of the ARQ protocol. The rough approximation is obtained considering continuous (back-to-back) transmission of frames at the wireless channel. The reason is that this operation of the ARQ protocol is essentially a multiplexing process of frames. Intuitively, this approximation becomes better when the number of ARQ instances and the number of frames in a single packet get bigger.

Summarizing, the performance of real-time applications running over wireless channels with centralized access is relatively well studied at both data-link and IP layers. There are a number of models that are versatile enough to apply to different wireless access technologies. Moreover, these models can often be formulated in terms of well-studied queuing systems. As a result, performance metrics of interest can be readily obtained.

### 5.3 Models for non-real-time applications

TCP congestion control alone brings many challenges to a performance analyst. Taking into account the complexity of TCP window dynamics, exact models of TCP are often too complicated to be suitable for analytical analysis. Depending on the amount of details taken into account, more or less accurate models have been formulated so far. Usually, these models approximate throughput of a TCP connection, as it is the most important parameter for non-real-time applications. Researchers who investigated throughput of TCP in wired networks found that it depends on both RTT of packets and the packet loss probability. In spite of differences in the TCP modeling approaches proposed so far, all of them led to the special cases of the \( 1/\text{RTT} \) law stating that TCP throughput is inversely proportional to the average RTT and the square root of the packet loss probability (see e.g. [57,58]). This result holds for wireless channels too. However, more refined results have been recently obtained.

In wireless networks packet losses can happen due to both buffer overflow at the IP layer and non-persistent ARQ operation at the data-link layer. Note that the latter losses do not necessarily imply that the chosen path in the network cannot support the current sending rate of a TCP source. However, in both cases losses trigger congestion control algorithm of the protocol halving the sending rate and retransmitting lost packets. Additionally, the IP packet transmission time may vary substantially when error correction schemes are used. Such kind of variability is expected to cause a number of undesirable effects, including occasional delay spikes leading to TCP timeout expiration and a subsequent sharp decrease of the sending rate (see e.g. [59,60]). Finally, most TCP models proposed for wired networks assume that the packet loss probability is known in advance and not affected by the connection under investigation. This may not be true when only one or few connections use the resources of a wireless channel.

In [22] Zorzi and Rao presented their performance evaluation framework allowing the study of the throughput of various TCP versions. The IP packet loss process was assumed to follow a two-state Markov chain. The model is defined using a discrete-time three-dimensional stochastic process in the form \( (C(n), T(n), W(n), n = 1, 2, \ldots) \), where \( C(n) \in [0, 1] \) is the state of a wireless channel, \( T \in [0, 1, \ldots, T_{\text{max}}] \) is the state of the slow start threshold, and \( W(n) \in [1, 2, \ldots, W_{\text{max}}] \) is the current length of the TCP congestion window. However, this process is not Markov as the time evolution of its states also depends on the number of unacknowledged packets in the network. Taking it into account in the state description of the model will make the process Markovian. However, the resulting model will be too complicated to solve analytically. Instead, the authors in [22] suggested sampling this process at specific time instants such that the imbedded process is Markov in nature. One possible option is to sample the process just after timeout expirations or once the loss recovery phase of TCP is completed. To derive performance parameters of interest they further used the Markov renewal/reward approach originally proposed in [40,41] for the GBN ARQ protocol. Since the flow control techniques used in TCP and GBN ARQ are roughly similar no significant modifications to performance analysis were required. We also note that although the IP error model used in this study can be extended to the case of \( M \)-state HMM, this results in a significant increase in complexity of analysis as the state space will grow exponentially. Using the proposed model the authors qualitatively compared the performance of various TCP versions. Particularly, it was demonstrated that TCP Tahoe and TCP Reno perform similarly over slow fading channels characterized by strong memory. TCP NewReno does not provide significant advantages in that environment either.

In [21], using a similar model, the authors demonstrated that given the same packet loss rate (PLR), autocorrelation
in the packet loss process positively affects the throughput obtained by TCP Tahoe compared to completely random errors. Particularly, they demonstrated that for the same average PLR the transmission rate increases as the burstiness of losses gets higher. Similar conclusions have been stated in [61,62]. The reason behind this phenomenon is that autocorrelation in the packet loss process increases the probability that all packets in the TCP congestion window are received correctly. On the other hand, when packet losses occur completely randomly, more congestion windows are affected on average. Note that these results are in accordance with findings made for HARQ schemes and real-time applications in [56,19] and questions the need for memory removal techniques.

Another study that performed in-depth analytical investigation of TCP running over a wireless link is due to Kumar [63]. He considered a local networking scenario where propagation delays are negligible, implying that RTT of packets as seen by a TCP source is simply zero. Packet losses were assumed to occur according to the renewal Bernoulli process and caused by imperfect error correction and/or collisions. Although the proposed approach is similar to that one used by Zorzi and Rao in [22], Kumar observed that, given Bernoulli packet losses and negligible RTTs, the three-dimensional Markov model \( (C(n), T(n), W(n), n = 1, 2, \ldots) \) originally proposed in [22] can be replaced by a one-dimensional Markov model \( (U(n), n = 0, 1, \ldots) \) imbedded at the moments of packet losses, where \( W(n) = 0 \) is the TCP congestion window size just after those moments. The reason is that under these assumptions TCP timeouts never expire and there is no need to track the state of the wireless channel model. The TCP congestion window size and the slow start threshold just after a packet loss event are completely defined using a one-dimensional Markov process \( (U(n), n = 0, 1, \ldots) \). The values of \( \{U(n), n = 0, 1, \ldots\} \) can be found by analyzing the operation of a particular TCP version. For example, for TCP Tahoe we have the following relations

\[
W(t^+(n)) = 1, \quad T(t^+(n)) = \left\lceil \frac{U(n)}{2} \right\rceil.
\]

where \( T(t^+(n)) \in \{0, 1, \ldots, T_{\text{max}}\} \) is the state of the slow start threshold just after a packet loss, \( W(t^+(n)) \in \{1, 2, \ldots, W_{\text{max}}\} \) is the size of the congestion window just after loss of a packet. Note that the value of \( U(n) \) implicitly determines the values of \( W(t^+(n)) \) and \( T(t^+(n)) \) and, therefore, uniquely gives the value of \( U(n+1) \). When the time between two lost packets is considered as a cycle, renewal theory can be used to determine the throughput of the TCP.

Using their model the authors demonstrated that the old version of TCP Tahoe performs exceptionally poorly even when the packet loss probability is as low as 0.001. When the packet loss probability increases to 0.01 the TCP throughput drops to 15% of the link rate. Obviously, the reason is that the sending window of the old version of TCP Tahoe always decreases to 1 when the packet is lost. The fast retransmit feature of TCP Tahoe, Reno and NewReno improves the throughput in the loss probability range of 0.001–0.01. However, there is a still significant drop beyond this point. For example TCP Reno achieves only 50% of the link throughput when the packet loss probability is 0.01. A further increase of the packet loss probability leads to getting less than 10% of the actual link throughput for all TCP versions. It is important to note that it can be natural for wireless channels to experience a packet loss probability in the range of 0.001–0.01 and even greater than 0.01.

One of the most general models of TCP throughput proposed so far is due to Altman et al. [64,62]. The only requirement imposed on the IP packet loss process is its covariance stationarity. As a result, it includes many special cases of wireless channel models at the IP layer, e.g. iid loss process, two-state Markov chain [23], M-state HMM [65,66], etc. Given a covariance stationary packet loss process \( \{W(n), n = 0, 1, \ldots\} \) the approach taken by the authors consists in defining and solving a stochastic linear equation in the form

\[
X(n + 1) = \lambda X(n) + aS(n),
\]

where \( X(n) \) is the value of the TCP rate process just prior to a packet loss, \( S(n) = W(n + 1) - W(n) \) is the distance between two loss events, \( \nu \) is the decrease in the rate of TCP caused by the loss of a packet, \( \nu \) is the increase in the rate of TCP. For TCP congestion control \( \nu \) is set to one half and \( \alpha \) is given by \( 1/bRTT^2 \), where \( b \) is the number of packets confirmed by a single ACK.

The stationary solution of (33) is given by

\[
X^*(n) = \alpha \sum_{k=0}^{\infty} \nu^k S(n - 1 - k).
\]

The general solution of (33) in terms of the distribution of \( X^*(n) \) is difficult to obtain. Using stochastic arguments the authors in [64,62] estimated the first two moments of \( X^*(n) \). As expected, under specific loss patterns the throughput of TCP obtained by the authors converges to those given by the Mathis and PFTK models in [58] and [57], respectively. It is important to note that this model was not specifically developed for a wireless environment. As a result, their studies do not explicitly address the effect of error correction techniques implemented at the data-link layer. However, in order to derive the packet loss process and service time of a packet other models discussed previously can be used.

Observe that all the above mentioned models explicitly assume that the parameters of the packet loss process are known in advance and are not affected by the connection under investigation. This implies that such models can be used when a large number of TCP connections is multiplexed over high-speed links such as those provided by IEEE 802.16 point-to-point access technology. When a wireless channel is exclusively assigned to a single station, only few TCP connections are expected to use its resources simultaneously, and the TCP congestion window size of each of those connections may noticeably affect the packet loss probability. In this case a different class of model needs to be used. Additionally, most of the models considered above either explicitly or implicitly assumed that packet losses are caused only by imperfect error control techniques, e.g. non-persistent ARQ. Since the buffer space is always limited in practice packet losses may also occur as a result of insufficient buffer space.

Another approach to estimate the throughput of a TCP connection in a wireless environment is to use the
so-called fixed-point approximation (FPA) technique. This approach was originally proposed for wired networks, where it was used to obtain the throughput of a number of TCP connections sharing a bottleneck [67]. The distinguishing feature of FPA is that no a priori knowledge of the packet loss probability is required. This information is derived directly from the model of the buffering process on a link. At a glance the fixed-point approach is a combination of the TCP source model, providing TCP throughput as a function of delay and loss parameters, and the queuing system, describing delay and loss performance provided by the network for a given TCP sending rate(s). Such a combination allows one to determine the so-called stationary operational regime of a bottleneck describing the rate at which TCP sources transmit and performance parameters provided to them by the network. From the modeling point of view, the fixed-point approach brings a number of equations that bind together the congestion window dynamics of a TCP source and performance parameters provided by the bottleneck. Formally, it can be written as

\[
\begin{align*}
E[W] &= f(p, T_R) \\
(p, T_R) &= g(E[W]),
\end{align*}
\]

(35)

where \( W \) is the average size of the TCP congestion window, \( p \) is the packet loss probability, \( T_R \) is the RTT, \( f(\cdot) \) and \( g(\cdot) \) denote some functional relationships. In (35) the first equation provides the average TCP window size as a function of the packet loss probability and RTT. Examples of these models include the Mathis [58] and PFTK [57] models. The second equation describes the relationship between \( p, T_R \), and the average window size \( W \). This relationship can be obtained by solving an appropriate queuing system. The stationary fixed-point describing the network operating regime is found by solving (35) numerically. An excellent overview of fixed-point approaches with an extensive list of references can be found in [68], while the existence of a fixed-point solution has been formally proven in [69]. Note that in some special cases analytical solution is feasible.

To apply the FPA technique to the wireless environment the authors in [70] proposed to distinguish between two modes of operation: (i) packet losses are primarily caused by imperfect error correction at the data-link layer (ii) the packet loss process is dominated by buffer overflows. To obtain \( W = f(p, T_R) \) the Mathis model was used in the former case while PFTK was used otherwise. The reason is that the Mathis model assumes purely random losses while PFTK models are suitable when losses are strictly correlated within a single congestion window of packets sent by a TCP source. In the latter case a single lost packet in the sending window is assumed to lead to the loss of all subsequent packets in this window. This property was introduced to represent packet loss behavior in droptail routers (see [57] for more details). The authors used a queuing system of the D-BMAP/G/1/K type to obtain \( p(T_R) = g(E[W]) \). The approach originally suggested in [56] was used to obtain the service time distribution of a single IP packet.

Another attempt to use FPA in a wireless environment was taken by Le et al. [71]. The approach is essentially similar to that proposed in [70]. The only distinguishing feature is that the authors did not differentiate between different types of packet loss. Although the proposed framework is quite general, it does not incorporate some important features. First of all, packet losses are only assumed to occur as a result of an excessive number of retransmission attempts, meaning that, effectively, the amount of the buffer space is unlimited. This assumption is barely justified in practice. For example, given ideal wireless channel conditions and conventional values of the receive window size the TCP sender may eventually use the whole bottleneck capacity and start losing packets due to the buffer being full. Additionally, only one TCP connection was assumed to use the resources of a wireless channel. This assumption is inappropriate for modern wireless access technologies where a number of TCP connections may compete for transmission capacity. However, these assumptions are relatively easy to relax using those approaches developed to date for wired networks (see e.g. [72, 73, 67]).

The FPA approach was also used in [74] to study the throughput of a single TCP source with the SACK option enabled. The major advantage of that model compared to previous studies is that the solution of the fixed-point equations is obtained analytically. Similarly to [70] the authors distinguished between wireless dominated and buffer dominated regimes of operation. In the latter case a modified FPTK model was used. When losses are mostly induced by imperfect error correction the Mathis model was used. However, instead of a queuing system modeling the packet transmission process over a wireless channel, a geometrical interpretation of TCP rate was used to obtain \( (p, T_R) = g(E[W]) \). An interesting conclusion revealed by the authors is that the TCP timeout expiration probability is negligible when a persistent or non-persistent Type I HARQ system is used at the data-link layer. As a result, the evolution of the congestion window is mainly driven by in-order and duplicate acknowledgements. Another important observation is that the throughput of a TCP connection approaches \( 1/RTT \sqrt{p} \), which is a well-known result for TCP sources. The most restrictive feature of their model is that only a single TCP source is allowed to use the resources of a wireless channel. When more than one source is assumed the model becomes analytically intractable.

Note that the FPA approach can also be used to estimate TCP throughput when active queue management (AQM) is used, e.g. random early detection (RED) [75]. This can be done by representing \( (p, T_R) = g(E[W]) \) in (35) in terms of the RED dropping function in the following form

\[
p(i) = \begin{cases} 
0, & k < L_{\text{min}} \\
\frac{k - L_{\text{min}}}{L_{\text{max}} - L_{\text{min}}}, & L_{\text{min}} \leq k < L_{\text{max}} \\
1, & k \geq L_{\text{max}}.
\end{cases}
\]

(36)

where \( L_{\text{min}} \) is the threshold after which packets start to be dropped, \( L_{\text{max}} \) is the threshold after which all arriving packets start to be dropped, \( p_{\text{max}} \) is the packet drop probability corresponding to \( L_{\text{max}} \). These parameters are illustrated in Fig. 11.

While FPA is approximate in nature and, in most cases, can only be solved numerically, it is intuitively clear compared to renewal analysis. Additionally, it can be used even in those cases when only few connections share the bottleneck.
and the congestion window sizes of these connections affect the packet loss probability. However, there are still many open questions concerning the accuracy of the approach. The general belief is that it is mainly affected by the choice of the appropriate traffic model approximating the packet arrival process from single or multiple TCP sources [74]. Note that FPA does not impose any restrictions on the choice of the traffic model. The only limiting factor is the complexity of the queuing model that needs to be solved. Due to these advantages this technique has become popular recently.

6. Performance models for distributed access

6.1. Description

In a distributed wireless access environment there are no controllers responsible for request arbitration between mobile stations competing for transmission resources. Instead a medium is dynamically shared according to a certain MAC protocol. IEEE 802.11 WLAN is the perfect example of a wireless technology with distributed channel access.

In distributed wireless access there is contention for resources between mobile stations at the information transmission phase. Performance modeling of these systems is a more complicated task compared to those with centralized access. Indeed, in addition to performance degradation caused by the error-prone nature of wireless channels and buffering at the IP layer, we have to take into account frame collisions caused by simultaneous transmissions from multiple stations. Moreover, when the number of stations competing for resources is large, frame losses caused by collisions are the dominating factor responsible for performance degradation. As a result, performance evaluation models proposed to date mainly concentrate on the effect of various MAC schemes. Due to this reason we briefly review them here.

ALOHA was the first protocol proposed for a distributed access environment. According to it a station having a frame for transmission begins its transmission immediately. Obviously, when the number of mobile stations grows the performance of ALOHA quickly degrades. Slotted modification of ALOHA tries to reduce the collision probability between stations by dividing the time into fixed length units. In slotted ALOHA, frames start to be transmitted only at slot boundaries. Although its performance in terms of throughput was found to be twice that of the original protocol, it is still unacceptably low [76]. As a result, ALOHA protocols have never been used in commercial wireless systems.

The carrier sense multiple access (CSMA) family of protocols is the most popular approach for distributed channel access. There have been a number of CSMA versions proposed to date. According to $p$-persistent CSMA a mobile station having a frame for transmission, after checking the channel to be idle, begins sending at the next slot boundary with probability $p$. With complementary probability $(1-p)$ it postpones its transmission attempt for some number of time slots randomly chosen from a geometrical distribution with parameter $p$. If the channel is busy at the moment of sensing, a mobile station continues to listen to the medium till the channel becomes idle. $1$-persistent CSMA is a special case of $p$-persistent CSMA, where a station begins transmission at the next slot boundary with probability $p = 1$ after sensing the channel to be free. Another version is non-persistent CSMA. In this case a station, after sensing the channel to be busy, postpones its listening attempt for a certain number of slots randomly chosen from a uniform distribution. However, if the medium is sensed to be idle the transmission attempt is initiated at the next slot boundary with probability 1.

CSMA with binary exponential back-off (BEB), often referred to as CSMA with collision avoidance (CSMA/CA), is a modified version of the non-persistent CSMA algorithm specifically developed for WLANs. According to CSMA/CA, if the channel is sensed to be busy a station postpones its transmission attempt for a certain period of time. This period is chosen randomly from $(0, CW_i)$, where $CW_j$ is the length of the contention window (CW) at the transmission attempt $i$. With each unsuccessful transmission attempt the length of CW doubles. The number of transmission attempts is limited to $N$. If a frame is still incorrectly received after $N$ transmission attempts an error is returned to higher layers. Channel access procedures for various CSMA modifications are illustrated in Fig. 12, where $G(p)$ is a geometrical distribution with parameter $p$, $U(m)$ is a uniform discrete distribution between 0 and $m$.

The CSMA/CA scheme was accepted as a mandatory contention-based channel access method in the IEEE 802.11 family of standards, where it is called the distributed coordination function (DCF). The DCF of IEEE 802.11 defines two modes of operation, including basic access and 4-way handshake access. According to basic access a frame is transmitted using DATA-ACK frames exchange where positive acknowledgements are used to confirm correct reception of a frame. This scheme performs poorly when the number of competing stations is large. To operate well in such an environment the 4-way handshake was proposed. In this case request-to-send (RTS) and clear-to-send (CTS) frames are exchanged prior to the actual data transmission. The advantage of using a RTS-CTS handshake is that these frames are much smaller compared to the data frames and possible collisions caused by them last less. However, the choice of the mode of operation does not substantially affect
the performance models proposed for the CSMA/CA access scheme.

It is important to note that most performance evaluation models developed recently for wireless systems with distributed channel access concentrate on CSMA/CA protocol with BEB. This is mainly because of the standardization and successful deployment of IEEE 802.11 WLANs. As a result, they have a number of common properties. First of all, they all describe performance at the data-link layer. Due to the complicated back-off procedure implemented in IEEE 802.11, DCF performance metrics of interest are often limited to simple parameters such as throughput of a single station and the mean delay experienced by frames. Finally, the terms “frame” and “packet” are often used interchangeably in those models. The reason is that IEEE 802.11 defines the maximum frame size which is sufficient to contain a single IP packet of 1500 bytes. Such IP packets correspond to the message transfer unit (MTU) of Ethernet and dominate in the Internet. However, since all those models we consider in this section estimate performance at the data-link layer, we will refer to the basic PDU in distributed wireless systems as a frame.

6.2. Models for real-time applications

Performance models developed for CSMA with BEB should track the evolution of $CW$ at the mobile station of interest. Note that the current value of the back-off counter at the station of interest, and thus, the transmission time of a frame depends on the amount of stations competing for access and the frame arrival pattern at each of them. Secondly, usage of a uniform distribution for computing back-off intervals does not allow any benefit from the memoryless property of geometrical distribution as in the case of $p$-persistent CSMA. Finally, we also have to keep track of the protocol stack at the station of interest, including segmentation and reassembly between adjacent layers, buffering at the IP layer, frame losses due to imperfect error control, etc. Due to these reasons, the performance of wireless systems with distributed channel access was conventionally studied using either oversimplified analytical models or computer simulations. However, during the last decade wireless systems with distributed access have attracted a new degree of attention. Some authors have analytically studied the performance of CSMA modifications under relatively general wireless channel and arriving traffic conditions and obtained results which are close to those provided by simulation studies and measurements of operational systems.

Until the seminal work of Bianchi [77] published in 2000 most analytical studies of CSMA algorithms assumed simple versions of the back-off process. Performance of $p$-persistent CSMA/CA was firstly studied by Kleinrock and Tobagi in [78]. Thanks to the memoryless property of the geometrical distribution, results for throughput are readily available. Since then their model has been extended to evaluate other versions of CSMA protocol, including non-persistent, 1-persistent, and $p$-persistent ones. In [79] the authors assumed non-persistent CSMA and evaluated its throughput under saturation conditions, i.e. assuming that all the stations always have frames ready for transmission. Throughput of non-persistent CSMA has also been considered in [80,81], while a compromise between 1-persistent and $p$-persistent CSMA protocols has been studied in [82].

Although the above mentioned studies provided a starting point in the analysis of the CSMA mechanism they all failed to capture advanced features of the CSMA/CA algorithm where the number of back-off stages as well as back-off interval at each stage vary. Another common shortcoming is that frame losses caused by simultaneous transmission attempts were the only source of performance degradation taken into account. Finally, most of those studies considered the performance of the CSMA access scheme under saturation conditions. For these reasons we did not consider any details of those models.

The CSMA protocol with back-off mechanisms similar to BEB was studied by Cali et al. in [83,84]. The only difference compared to the BEB algorithm is that the back-off counter at each stage of the back-off process is assumed to be geometrically distributed. This assumption helps to benefit from the memoryless property by dispensing with maintaining an additional counting variable representing...
the number of back-off slots left till the next transmission attempt. According to the memoryless property it has the same distribution irrespective of the amount of time slots that the system has already spent in a certain back-off stage. One may expect that inaccuracies caused by this assumption can be significant. However, since mean performance measures, such as overall system throughput, were performance metrics of interest the authors argued that their model closely approximates the CSMA protocol with BEB. While this may indeed be true the proposed model may not be appropriate for other performance metrics such as delay distribution. In [84] the authors concentrated on the efficiency of the exponential back-off mechanism and determined criteria when the system consisting of N stations achieves its peak throughput. They demonstrated that in order for a system to achieve it on-line tuning of the exponential back-off mechanism is needed. Unfortunately, it requires knowledge of the environment a station operates in, e.g. the number of active stations and the length of frames they transmit. In [83] the authors continued their efforts demonstrating that these requirements can be relaxed and only the number of active stations is required to be known for optimized operation. Since all the stations share the same transmission medium this information can be inferred from ongoing transmissions.

The fundamental theoretical model proposed for the CSMA/CA access mechanism with the BEB algorithm is due to Bianchi [85]. He considered saturation conditions and evaluated the throughput obtained by a single station. He invoked when the medium is sensed to be busy. In order to obtain TCP throughput using the FPA technique. Recall that it consists in bringing together two equations binding two unknown variables of interest. In the case of CSMA/CA these variables are the frame collision probability, $p$, and the number of active stations, $n$. In [86,39] the authors improved the analysis of the back-off procedure such that it is only invoked when the medium is sensed to be busy. The model was validated against empirical measurements of the saturated IEEE 802.11b network and was found to provide fairly accurate results in some scenarios. However, this model still makes a number of unrealistic assumptions that may lead to biased results. First of all, it assumes that a frame can only be incorrectly received as a result of simultaneous transmissions from multiple stations, neglecting the effect of losses caused by incorrect reception of channel symbols. Next, the author assumes that the collision probability remains the same at all back-off stages, which may not be true in general. Recall that the number of slots from which the back-off counter is drawn doubles with each transmission attempt. As a result, the actual collision probability is different at different back-off stages. Finally, assumption of saturation condition was accepted for analysis. While saturation conditions address an important scenario of greedy applications, it still significantly limits application of the proposed approach. Notice that this assumption avoids another dimension in the Markov process that would be needed in models of this kind to represent the state of the arrival process. Nevertheless, the model has proven itself to be quite versatile in providing a building block for subsequent efforts in the modeling of CSMA/CA networks. Recently, performance models proposed for CSMA/CA networks tried to relax the restrictive assumptions of [85].

According to [85], the back-off procedure is always invoked even when the medium is sensed to be free in the first transmission attempt. In this context, extension of the work of Bianchi was performed in [87], where the authors improved the analysis of the back-off procedure such that it is only invoked when the medium is sensed to be busy. In order to perform this extension, only minor modifications were introduced to the original Markov model. The performance metrics of interest were the throughput obtained by a single station.

The system of two equations now reads as [85]

$$
\begin{align*}
\tau &= \frac{2(1-2p)}{(1-2p)(W+1) + pW(1-(2p)^m)}, \\
\tau &= 1 - (1-p)^{1/(n-1)}.
\end{align*}
$$

The system (39) is the fixed-point equation in the form $r(p) = p$, which can be solved numerically to obtain the so-called stationary operational regime of CSMA/CA mechanism $(r, p)$. Using the probability of transmission in an arbitrary slot one can obtain the throughput of the system and the mean delay experienced by an arbitrary frame. The model was validated against empirical measurements of the saturated IEEE 802.11b network and was found to provide fairly accurate results in some scenarios. However, this model still makes a number of unrealistic assumptions that may lead to biased results. First of all, it assumes that a frame can only be incorrectly received as a result of simultaneous transmissions from multiple stations, neglecting the effect of losses caused by incorrect reception of channel symbols. Next, the author assumes that the collision probability remains the same at all back-off stages, which may not be true in general. Recall that the number of slots from which the back-off counter is drawn doubles with each transmission attempt. As a result, the actual collision probability is different at different back-off stages. Finally, assumption of saturation condition was accepted for analysis. While saturation conditions address an important scenario of greedy applications, it still significantly limits application of the proposed approach. Notice that this assumption avoids another dimension in the Markov process that would be needed in models of this kind to represent the state of the arrival process. Nevertheless, the model has proven itself to be quite versatile in providing a building block for subsequent efforts in the modeling of CSMA/CA networks. Recently, performance models proposed for CSMA/CA networks tried to relax the restrictive assumptions of [85].

$$
\begin{align*}
\tau &= \frac{2(1-2p)}{(1-2p)(W+1) + pW(1-(2p)^m)}, \\
\tau &= 1 - (1-p)^{1/(n-1)}.
\end{align*}
$$

![Fig. 13 — Two-dimensional Markov chain modeling the CSMA/CA mechanism.](image)

Inverting (38) we obtain another independent expression for the probability of transmission in an arbitrary slot.
and the mean delay experienced by frames. Among other conclusions, it was shown that the behavior of a station during its first attempt to transmit a frame does not noticeably affect the delay and throughput performance metrics for both basic and 4-way handshake access schemes. On the other hand, the choice of the initial contention window and the number of back-off stages were shown to have a great impact on delay performance. It was also demonstrated that for large number of stations in the system the delay performance of the 4-way handshake access scheme is substantially better than that of the basic access method.

The authors in [88] were the first who considered unsaturated conditions allowing arbitrary traffic arrival models and explicitly took into account the evolution of the back-off counter caused by frame collisions. The modeling approach consists of two steps. At the first step the frame loss probability is found. The authors assumed that the frame arrival rate at each node is given by $\lambda$ and obtained the mean CW size in unsaturation condition as $E[W] = (1 - \pi_0)E[W_0]$, where $\pi_0$ is the probability that the node's buffer is empty when a new arrival occurs and $E[W_0]$ is the mean CW size in the saturated condition. The reasoning behind this expression is that with probability $(1 - \pi_0)$ any arrival frame is backlogged. Here, the authors explicitly assumed that those frames arriving to an idle node are immediately sent and no collisions occur. This is not always true and may significantly affect accuracy of the model. The authors further proposed to approximate $\pi_0$ using $(1 - \rho)$, $\rho = \lambda/\mu$, where $\mu$ is the mean service rate of frames. Note that this result is a rough approximation as it holds for the M/M/1 queuing system only. The average CW size for saturated conditions is obtained by observing that the $k$th unsuccessful transmission attempt increases the mean waiting time by $2^k/W_{\text{min}}$. This gives

$$E[W] = \frac{(1 - p)W_{\text{min}}}{2} + \frac{p(1 - p)2W_{\text{min}}}{2} + \cdots + \frac{p^{m+1}2^mW_{\text{min}}}{2} = \frac{1 - p - p(2p)^mW_{\text{min}}}{2(1 - 2p)} \quad (40)$$

Observing that only a fraction of stations with non-empty buffers actually contends for resources, the frame collision probability is obtained by substituting (40) in $E[W] = (1 - \pi_0)E[W_0]$ and solving for $\pi_0$. Numerical results provided by the authors demonstrate that the accuracy of the model is sometimes unsatisfactory. This is basically due to the number of approximations and assumptions taken by the authors. Another limiting feature of the proposed model is that buffers of mobile stations are allowed to be infinite. As a result, additional care must be taken using this model for specific performance parameters of interest such as buffer overflow probabilities. While this parameter can still be found using crossover probabilities $Pr(Q > q)$, where $q$ is the buffer size, the accuracy of this approximation is questionable. Finally, the model does not take into account frame losses occurring as a result of incorrect reception of channel symbols. Nevertheless, the proposed model was the first one that allowed the unsaturated condition to be studied in the CSMA/CA environment. The authors used this model in [89] to explore the effect of the IEEE 802.11 MAC protocol on traffic characteristics at the access point. The most interesting finding was that inter-arrival times are best characterized by a multimodal distribution. The plausible explanation is that the back-off interval doubles with each unsuccessful retransmission attempt leading to this specific behavior. The proposed model was further extended in [90], where the authors studied the impact of the network load on loss rates and mean delay of a frame. They also relaxed some of the assumptions used in their previous work, e.g. the M/M/1 approximation, constant collision probability, etc. However, their model still did not include frame losses caused by incorrect reception of channel symbols.

The approaches cited above assume infinite capacity models. To model CSMA/CA performance, Ozdemir and McDonald in [91,92] represented each station using a finite capacity queue of M/G/I/K type. Additionally, this is the only study where authors modeled the back-off mechanism of the CSMA/CA protocol such that frames are no longer assumed to collide with constant and equal probabilities. This was done by extending the basic model proposed in [85]. They started evaluating the conditional probability $\alpha_{ij,k}$ that the transmission from node 1 results in a collision given that node 1 is in stage $i$ and node 2 in stage $j$, node 1 selects slot $k$, and there are only two nodes in the system. If node 1 picks slot 1, for a collision to occur node 2 must pick slot 1 too. Thus, $\alpha_{00,1} = 1/CW_0$. If node 1 picks slot 2 we have two possible scenarios for a collision to occur (i) node 2 picks slot 2 and (ii) node 2 picks slot 1, successfully transmits a frame, immediately becomes busy again and picks slot 1. We have $\alpha_{00,2} = 1/CW_0 + b_0/2CW_0$, where $b_0$ is the probability that a station is busy, i.e. has a frame transmission. Continuing along these lines we get

$$\alpha_{00,k} = \frac{1}{2} \sum_{m=0}^{k-1} \left( \frac{b_0}{CW_0} \right)^m, \quad 0 \leq k \leq CW_0 - 1. \quad (41)$$

Let $P_{00,1}$ be the probability of collision between 1 nodes at stage 0. Using $a_k$ as the probability that node 1 chooses slot $k$, we have $P_{00,2} = \sum_{k=0}^{1} \alpha_{00,k} a_k$. Considering $P_{00,1}$ for $i = 1, 2, \ldots, m$, where $m$ is the maximum number of retransmission attempts allowed for a single frame, and taking into account that nodes may now choose time slots from uniform distributions corresponding to different back-off stages, the authors obtain the final expressions for $p_{ij,2}$ in the form

$$p_{ij,2} = \frac{W_0 P_{ij,2} \sum_{k=0}^{2i-1} b_k}{W_j} \quad (42)$$

Taking into account the steady-state occupancy probabilities $\eta_{ij} = 0, 1, \ldots, m$ for the $m$ back-off stages are known, conditioning on the second node being in state $j$ is removed, and the conditional probability of collision $p_{i,2}$ given that node 1 is in the stage 1 is given by

$$p_{i,2} = \sum_{j=0}^{m} p_{ij,2} \eta_j. \quad (43)$$

Finally, $p_{i,n}$ can be obtained by assuming that there are $n$ active nodes in the system and the probabilities of collision between different node pairs are independent. We have

$$p_{i,n} = 1 - (1 - p_{i,2})^{n-1}, \quad n = 2, 3, \ldots, N. \quad (44)$$

where $N$ is the overall number of stations.
Given the number of busy nodes the state-dependent frame collision probabilities $p_i$ can be computed using (44). As a result, the only modification to the Markov model originally proposed in [85] is that instead of the constant frame collision probability $p$ for all back-off stages the state-dependent collision probabilities $p_i$ are used. The closed form solution of the model is obtained in terms of the probability that a single node transmits in a randomly chosen slot as a function of probability that an arbitrary node has a frame for transmission, $b_0$. Note that the number of busy nodes is not known in advance, as nodes are not assumed to be saturated. However, given $b_0$ and assuming that all the stations are independent of each other, the probability that $k$ out of $N$ stations are active at an arbitrary instant of time is given by a binomial distribution with parameters ($b_0$, $k$, $N$). In order to determine $b_0$ the authors used the M/G/1/K queuing model, where the service time distribution is the distribution of the back-off window size found at the previous step plus the amount of time required to perform frame transmission attempts for a single frame. Particularly, this distribution describes the time between the first attempt to transmit a frame till its successful transmission or loss caused by excessive amount of retransmission attempts. Solving this queue the authors obtained the probability that there is more than 0 frames in the system which corresponds to $b_0$. It is important to note that the solution of the whole model is iterative in nature. Indeed, to derive the transition probabilities of the Markov model describing the back-off mechanism, the probability that the node is busy is required. However, this probability can only be obtained by solving the M/G/1/K queuing system which, in turn, needs the service time distribution. Numerical results presented by the authors demonstrated excellent agreement with simulation studies.

A similar model has been proposed by Zhai et al. [93]. The only difference compared to the work presented in [91,92] is in the computation of the service time distribution. Zhai et al. use the probability generating functions while Ozdemir and McDonald derive it algebraically. It is important to note that the authors in [93] also considered the error of replacing the M/G/1/K queuing system by the M/M/1/K one. Surprisingly, it was shown to be negligible for their parameters of choice. In [94] the authors further extended the work of Zhai et al., developing a more accurate and tractable approach to estimate the service time distribution. Nevertheless, it is still based on probability generating functions.

A step forward was taken in [95]. While the modeling approach used by the authors is similar to that one originally proposed in [91,92], to the best of the author’s knowledge this is the only study where an advanced arrival process was considered. In particular, the authors used the MMPP/G/1/K model to obtain the probability that an arbitrary node has no frames for transmission. The service time distribution in this model is obtained by solving a Markov model similar to that introduced in [85]. The proposed approach is the only one available to date for analysis of the CSMA/CA mechanism where autocorrelated properties of the arrival process are explicitly taken into account while the model still preserves analytical tractability. Unfortunately, the authors did not provide an in-depth look at the effect of memory of the frame arrival process on its performance characteristics in CSMA/CA environment. It is important to note that the solution of the model involves a three-dimensional Markov chain resulting in significant computational requirements. Another shortcoming of the model is that the wireless channel is still assumed to be completely reliable, i.e. frames cannot be lost as a result of incorrect reception of channel symbols. Finally, the model also assumes that frame collision probabilities at each stage of the back-off algorithm are the same.

Those models cited above considered frame collisions as the main source of performance degradations in CSMA/CA networks. The first paper that took into account losses of frames caused by incorrect reception of channel symbols is due to Giovanardi [96]. The authors considered the 1-persistent CSMA mechanism in a collision-free environment. The wireless channel is considered to be error-prone and modeled by the Gilbert model. Performance of a single station was then captured using a modification of the discrete M/G/1 queueing system, with a Poisson distributed number of arrivals in a slot. The underlying Markov chain was constructed taking into account incorrect reception of frames in certain slots. As a result, in terms of the queuing theory their model is actually non-preemptive M+$\text{IBP}/$G/1, where IBP stands for the interrupted Bernoulli process and models incorrect reception of frames. Performance parameters of interest were the mean number of frames in a queue and the mean transmission time of a frame. Unfortunately, due to the oversimplifying assumptions taken by the authors their model is far from realistic. Nevertheless, it served as a starting point and other studies taking into account the error-prone nature of wireless channels started to appear. A more accurate model was proposed by Hadzi-Velkov and Spasenovski [97], who extended Bianchi’s model to the case of error-prone wireless channel behavior. The effect of fading was taken into account using a single parameter, the probability of incorrect frame reception, $p_L$, resulting in the renewal loss process. As a result, (38) is modified and now given by $p = 1 - (1 - p_L)(1 - r)^{n-1}$. Inverting it we can construct a system of equations similar to (39) that can be solved numerically.

Performance of the frame transmission process from a single station in error-prone wireless channel conditions was also considered in [94]. The model is based on the M/G/1/K queuing system and represents unsaturated conditions. Another distinguishing feature is that this model explicitly takes into account frame losses caused by buffer overflows. However, error-prone channel conditions were taken into account using the frame error probability only resulting in the renewal loss process. In [98] the authors relaxed assumptions of the renewal frame error process. To the best of the author’s knowledge the work presented in [98] is the most general model of a CSMA/CA wireless environment as it includes most factors affecting the performance of CSMA/CA networks, e.g. BEB algorithm, incorrect reception of frames as a result of both incorrect reception of channel symbols and frame collisions, non-saturated traffic conditions, and limited buffer space of networking nodes. However, in order to solve the model an iterative procedure similar to that used by Ozdemir and McDonald [91,92] is required. Assumption of equal collision probabilities at different back-off stages is the only simplifying assumption.
A non-trivial approach has been taken by Foh and Zukerman in [99]. The authors considered a state-dependent M(0)/PH/1/K queuing system modeling the frame service process on a shared medium. The arrival process in this system represents the number of stations that become active, i.e. receive a frame from their applications and start contending for resources. For simplicity of analysis the station arrival process is chosen to be Poisson in nature. While this assumption may hold when the number of stations competing for resources is sufficiently high, i.e. more than 10, there is no particular reason to assume it for realistic conditions, especially, when buffering is used at each station. The statistical characteristics of the frame service time for i active stations in the system are obtained using the results of Bianchi [85]. Indeed, when i stations are active the service time provided by the wireless medium is similar to that of the saturation conditions for the i stations scenario. Since the service time distribution is not readily available in [85], the authors carried out simulation studies and found that it can be sufficiently well approximated by the PH distribution. Among other conclusions, the authors observed that the actual back-off time spent by a station in a certain back-off phase waiting for a retransmission is of secondary importance compared to the transmission time itself. The proposed approach has a further upside as it uses two models to represent a complex system. This allows the use of more sophisticated and accurate models for both parts of the system. Additionally, it effectively avoids an iterative approach to obtaining the steady-state performance metrics. Finally, it seems suitable for evaluating advanced performance metrics of individual stations in the CSMA/CA environment for non-homogenous traffic scenarios (which has never been done to date). Unfortunately, the authors did not further elaborate their model.

An interesting approach was introduced in [100], where the authors replaced the Markov model used in [85] by the closed queuing network model as shown in Fig. 14. Each back-off stage of BEB algorithm is represented by a separate queuing system. The number of queuing systems corresponds to the number of back-off stages. The service time at the queue \( i \), \( i = 0, 1, \ldots, m \) corresponds to the back-off interval selected uniformly from \( U(0, CW_i) \). With probability \( (1-p) \) the arrival leaves the queue \( i \). This situation is interpreted as successful transmission of a frame at the \( i \)th transmission attempt. With complementary probability \( p \) the frame enters the next queue and is delayed for some time randomly chosen from \( U(0, CW_{i+1}) \). Probability \( p \) is interpreted as the probability of collision and assumed to be the same for all back-off stages. It is easy to observe from the structure of the queuing network, that served customers immediately return back to the beginning of the system implying saturated conditions. Since the proposed structure was intended to represent the whole contention environment with a constant number of stations, the authors had to ensure that no frames are waiting in the queues before they start getting service. To fulfill this requirement they used G/G/\infty queuing systems with an infinite number of servers. Due to the chain-like structure of the queuing network, and observing that the probability of transmission attempt in an arbitrary slot is given by \( 1/N \), where \( N \) is the total rate of attempts and \( N \) is the number of stations, the analysis is rather straightforward and the solution for \( \tau \) can be obtained in closed form. Similarly to [85] the authors obtained an independent expression for \( \tau \) inverting \( p = 1 - (1 - r)^{n-1} \), and then solved the system of equations numerically to get the stationary operational regime of the contention environment \( (r, p) \). Using \( r \) and \( p \), the throughput and mean delay metrics have been obtained.

Summarizing, those models developed prior to Bianchi’s work [85] addressed performance of simple CSMA protocols including p-persistent, 1-persistent, and non-persistent CSMA. Nowadays, most analytical performance models of CSMA/CA mechanism are based on the seminal work of Bianchi and explicitly include evolution of the contention window in terms of the back-off stage and the back-off counter. The basic idea is to solve the fixed-point equation \( r(p) = p \) describing the stationary regime of a contention environment consisting of a certain number of active stations.

The major difference between the various approaches suggested to date is how wireless channel and frame arrivals are represented. In most studies frame losses caused by incorrect reception of channel symbols were assumed to have a secondary effect compared to collisions and were just neglected. When the number of stations competing for resources is large and the amount of traffic they generate is high then frame collisions are indeed a dominating source of performance degradation. Otherwise, the error-prone nature of a wireless channel may have a more profound effect. Moreover, as transmission rates in subsequent IEEE 802.11 standards are expected to increase the effect of error-prone wireless channel behavior will be more severe.

Another shortcoming of those models developed so far is that frame arrival process is often assumed to be very simple. Most studies consider saturation conditions assuming that all nodes always have frames ready for transmission. Although this assumption is useful to evaluate performance in the overload scenario it leads to many limitations of the proposed models. First of all, it becomes impossible to evaluate the performance of individual applications in 802.11 environment. For example, access delays inherent for wireless channels with distributed access schemes are expected to have a negative effect on the performance of real-time applications such as voice-over-IP. The degree of this effect is still an open question. Secondly, individual characteristics of various applications may produce an additional impact on the performance of the whole system. For example, correlation in the frame arrival process is known to produce a negative effect on buffering and loss processes. When unsaturated conditions are considered, the situation is also complex. The reason is that there are no universal frame
arrival models suitable for a wide range of traffic sources. As a result, upcoming performance models should be suitable for use with different arrival models.

We also note that most models developed to date are aimed at performance evaluation of the DCF mode of the IEEE 802.11 WLAN standard. Since no error correction is defined for IEEE 802.11, neither of them include FEC, ARQ or HARQ at the data-link layer. Practically, it means that possible gains that might be achieved using error correction techniques in future WLAN technologies can be evaluated using simulation studies only. Also there are no models that evaluate potential performance gains provided by channel adaptation techniques including AMC, MIMO, and power control. Attempting to make WLAN technology as simple as possible these mechanisms were not defined in pioneering WLAN standards. However, they now started to appear as an important part of future wireless systems with distributed channel access, see e.g. IEEE 802.11n.

6.3. Models for non-real-time applications

Modeling performance of TCP running over wireless channels with a distributed access scheme is an extremely complicated task. The major reason is the complexity of the system we want to model. Indeed, as we observed above, the distributed access environment is complicated to represent analytically even when arriving traffic is independent of network conditions. We also highlighted that TCP congestion and flow control procedures create many challenges for a performance analyst when a centralized access scheme is considered. Adding frame collisions as an additional source of performance degradation would result in new interdependencies that need to be taken into account. For example, the congestion window size of a TCP source, and therefore, the amount of frames that need to be transmitted depends on both RTT and the frame loss probability. In addition to those losses caused by buffer overflow and the error-prone nature of a wireless channel, the latter parameter should now take into account the probability of frame collisions which, in turn, depends on the congestion window size. The situation becomes more complex when TCP timeouts need to be taken into account. Indeed, as was demonstrated in [101] the frame delay distribution is multimodal in CSMA/CA networks. This property may lead to frequent TCP timeouts. As a result, little to no progress has been achieved so far in analytical modeling of TCP performance in a distributed access environment.

One of the promising ways to approach the problem of TCP modeling in a distributed access environment is to divide the whole system into a number of parts and further analyze each part separately. In order to do so a framework similar to that proposed by Foh and Zukerman for real-time applications can be used [99]. Firstly, one can model the number of active stations using a queuing system of the Geo(i)/G/1/K type with state-dependent arrivals. In this the queue arrival process represents those stations that become busy, i.e. receive a new frame for transmission. The service process reflects stations that become empty, i.e. transmitted all their frames and have no new frames for transmission. The state of the system is represented by the number of active stations. The service process in each state of the system can be obtained by using a set of queuing systems of the G/G/1/K type. In these, the queue arrival process and service times remain unchanged and correspond to i active stations in the system. The service time distribution can be obtained using the approach originally presented by Bianchi in [85] and then extended to the case of non-ideal channel conditions by Hadzi-Velkov and Spasenovski in [97]. The solution of G/G/1/K queuing systems gives steady-state probabilities \( p_i \) that there are \( i, i = 0, 1, \ldots, N \) active stations in the system. Weighting the service time distribution in each state of the system with the respective steady-state probabilities of these states we get the overall service time distribution for the Geo(i)/G/1/K queuing system. Finally, we can use the fixed-point approximation technique to determine the throughput of TCP sources in the CSMA/CA environment. Note that the model in this form can be computationally intensive as it requires solving \( i+1 \) queuing systems.

6.4. Models for special effects

To date there has been rather limited activity in modeling the special effects of a distributed channel access environment. This includes the effect of hidden terminals, channel capture, interference between adjacent stations, fairness of CSMA algorithms, etc. Most models implicitly or explicitly imply that the performance degradation caused by these effects is of secondary importance compared to incorrect frame reception caused by either collisions or insufficient received signal strength.

Fairness of the CSMA protocol has been considered in many studies (see [102] for up-to-date discussion on the topic). Almost all CSMA-based protocols were found to be unfair in terms of the throughput provided to individual stations. There are many reasons behind this, including the channel capture effect, the effect of hidden terminals, neighbor receivers problem, exposed node uncertainty, etc. Observe that most studies reviewed previously implicitly assume that CSMA protocols are inherently fair in providing a share of throughput to individual stations. If this is not true then to a great extent the accuracy of those studies needs to be questioned. To evaluate the scope of available models the degree of unfairness existing in current and/or upcoming CSMA-based protocols should be further addressed.

The effect of intercell interference and hidden terminals has been analyzed in [80], where the authors considered the non-persistent version of CSMA. It was demonstrated that under high traffic conditions the presence of RTS/CTS mechanisms allows a significant shortening of the time required to resolve collisions and results in a better throughput of the whole system. Interference was shown to have a dramatic impact on the system throughput. The only way to resolve this problem is to increase the distance between the same frequencies in a network. The effect of hidden terminals was also studied in [103], where the authors considered the performance of CSMA/CA in a number of simple topologies. Among other conclusions, the authors demonstrated that even in presence of the RTS/CTS mechanism some nodes in a network may saturate even when the offered traffic load is as low as 0.15. This implies
that the effect of hidden terminals can be a dominating factor in performance degradation and should be taken into account in performance modeling studies. In [104] the authors introduced a model similar to that used in [104] demonstrating that the channel capture effect may lead to significant improvement in terms of the peak throughput of the whole system when a basic access method is used. When RTS/CTS is used the effect is not that dramatic. However, note that this is achieved at the expense of fairness. In [94] the authors confirmed the results of [104] for a system with multiple frame reception capabilities operating according to the CSMA/CA mechanism in low traffic conditions.

7. Conclusions

As there are already many wireless access technologies and more are expected to appear in the future to fulfill the need of users for greater bandwidth at the air interface, there is a growing need for performance analysis of new technologies and optimization of the existing ones. Given the existing database of models and approaches proposed so far, it is extremely difficult for a newcomer to the field to choose a point to start from and get a basic picture of the work that has been done before. In this paper we tried to summarize the work in performance evaluation of wireless technologies with centralized and distributed channel access schemes. These two access schemes differ considerably and require different types of model to evaluate their performance. However, for each access type the models are basically similar and differ in minor details only. It is also important to note that different approaches are required to estimate the performance of real-time and non-real-time applications.

In performance evaluation of wireless access technologies special attention should be paid to channel and traffic modeling algorithms. These two components affect the performance provided to applications, and their accurate modeling is crucial for successful performance evaluation. We demonstrated that there are a number of unsolved issues in wireless channel modeling. Specifically, there are still no conclusions as to whether wireless channel observations can be considered as a realization of a covariance stationary stochastic process. We also reviewed traffic models and identified that discrete-time batch arrival processes are better suited for performance modeling of wireless channels.

Nowadays, state-of-the-art wireless access technologies differ in many aspects of their operation, including operating frequency, modulation and coding schemes, multiple access schemes, channel access algorithms, error mitigation principles, etc. Although the literature on performance evaluation of wireless technologies as exploded over the past decade there are no approaches that are versatile enough to apply to any technology of interest. The reason is that a wireless environment provides a very challenging task for developers requiring advanced channel access schemes and adaptation mechanisms. The choice of the candidate model for performance analysis depends on the specific purposes of an analyst, protocol configuration of the wireless channel, and environment the system operates in. Nevertheless, many performance evaluation models proposed to date already include certain combinations of these characteristics. Our study shows that for many future wireless access technologies there is already a huge set of performance modeling approaches to apply directly or to start from.

REFERENCES

[34] D. Lucantoni, New results on the single server queue with a batch Markovian arrival process, Stoch. Mod. 7 (1) (1991) 1–46.