Finite Length Coding in Edge Computing Scenarios

Sebastian Schiessl, *Student Member, IEEE,* Hussein Al-Zubaidy, *Senior Member, IEEE,*
Mikael Skoglund, *Senior Member, IEEE,* and James Gross, *Senior Member, IEEE*

***Abstract*—Future cellular networks are expected to have resources for data processing at the edge of the network, providing the benefits of cloud computing without long routing delays. Such computing resources may for example be used in a control scenario, where wireless sensors upload data to a controller running at the edge of the mobile network. The controller computes an actuation command that is then sent back on the downlink to a wireless device. The delay in such a system not only depends on the processing time of the controller, but also on the uplink and downlink channels where fading and packet losses can result in a queuing delay. In this work, we present a probabilistic upper bound on the total delay of such systems when the channels are subject to Nakagami-m fading. Our method takes the effect of channel coding at finite blocklength in the uplink and downlink channels into account. Using several series expansions, the probabilistic delay bounds can be computed analytically, providing guidelines for resource allocation without the need for extensive simulations.**

***Index Terms*—Finite blocklength regime, rate adaptation, quasi-static fading, queueing analysis, edge computing, network calculus**

I. Introduction

Cloud computing is now employed by a wide variety of users, as cloud services can quickly provide large amounts of computational resources when necessary. Users thus pay only for the resources they need, instead of paying for private server farms that are not always fully utilized. Recently, a related concept called Edge Computing or Fog Computing has emerged, where the computational resources are located at the edge of the network [1], which eliminates delays due to the transmission of tasks over the Internet to remote cloud servers. The computational resources therefore become available to applications that require low latency, such as connected vehicles, the smart grid, and factory automation with wireless sensors and actuators. In such applications, data is sent wirelessly to a computing node, which processes the data and sends time-critical information, e.g. collision warnings or actuation commands, back to a wireless device [1]. Compared to traditional control logic chips located directly at the devices, such edge computing resources can provide high processing power right on demand, aggregate information from a large number of devices, and extract information for further analysis and optimization.

While the edge computing principle avoids delays due to routing in the network, delays can still occur due to the time-varying nature of wireless channels with fading and packet losses, which necessitate transmit buffers for the uplink and downlink. Furthermore, as the data rate in the uplink is time-varying, data may arrive in a bursty fashion at the edge computing node and may need to be buffered before it can be processed. An analysis of the end-to-end delay in edge computing scenarios must therefore examine the combined random queueing delay in the three buffers at the uplink, the edge computing node, and the downlink. As the end-to-end delay depends on the quality of the wireless links, the data rates and error probabilities in those links must be accurately modeled. Specifically, due to the typically small packet sizes and data rates in applications such as factory automation, channel models must take the effects of channel coding at finite blocklength [2] into account.

Previous works analyzed the probability distribution of queueing delays in wireless networks in different ways, most notably by applying the frameworks of effective capacity [3] and stochastic network calculus [4]. While the effective capacity approach offers an asymptotic evaluation for the tail of the delay distribution, i.e. for large delays, stochastic network calculus can provide non-asymptotic upper bounds on the distribution for any delay. Stochastic network calculus can also be extended to multi-hop systems with non-identical links [5]. The impact of channel coding at finite blocklength on the queueing delay was first investigated by Gursoy [6] using effective capacity. In our previous work [7], we developed closed-form approximations that allow fast computation of statistical delay bounds using stochastic network calculus. In that work, we considered a single-hop, single-antenna Rayleigh fading channel and provided closed-form approximations for computing probabilistic delay bounds. Recently, Li et al. [8] used effective capacity to analyze the delay in a two-hop wireless network at finite blocklength. However, their method requires that the first link has lower average data rate than the second link, which is not always the case. With regards to in-network processing, Al-Zubaidy et al. [10] studied a scenario with video processing in a computing node inside a wireless network. In that work, it was assumed that the processing node must wait for an entire video frame before processing can start, which can possibly lead to a large delay. However, finite blocklength effects in the wireless links were not considered.

In this work, we use stochastic network calculus in a transform domain [4] to analyze the queueing delay, which includes the queueing delay of the edge computing node. In contrast to our previous work [7], which only considered single-hop wireless links and was restricted to Rayleigh fading channels, we employ the Nakagami-m fading model, which can be used to model multi-antenna configurations. Moreover, we analyze systems with flow transformation, i.e. where the output data rate of the computing node is different from the input rate. We provide closed-form approximations of the

The authors are with the School of Electrical Engineering, KTH Royal Institute of Technology, Stockholm, Sweden (e-mail: schiessl@kth.se, hzubaidy@kth.se, skoglund@kth.se, james.gross@ee.kth.se).

relevant integrals, allowing fast computations of the delay bounds. An extensive simulation study was conducted to verify our computed bounds. The simulation results show that the analytically obtained bounds can accurately predict the optimal parameter regions for minimizing the delay. Specifically, the delay depends on the blocklengths of the channel codes in the uplink and downlink, and the choice of blocklengths which minimizes the analytical delay bounds also minimizes the actual delays that occur in the simulations.

II. System Model

We consider a scenario where data generated at a source is uploaded wirelessly to the cellular network. An edge computing node (ECN) processes the data and sends the result over the downlink to a different wireless node. For the wireless communication, we consider a time-slotted half-duplex system, where each time slot is split into an uplink and a downlink phase. In each time slot, the uplink and downlink can transmit codewords of length $n_{\mathtt{UL}}$ and $n_{\mathtt{DL}}$, respectively. The total number of symbols in each time slot is denoted by $N = n_{\mathtt{UL}} + n_{\mathtt{DL}}$. We use a block-fading model, where the signal-to-noise ratio (SNR) is assumed to be constant within one time slot but varies independently from slot to slot. Furthermore, we assume independence between the uplink and downlink channels. We employ the Nakagami-m fading model, where the SNR follows a gamma distribution with PDF [9, p. 849]

$$f(\gamma) = \frac{m^m}{\bar{\gamma}^m \Gamma(m)} \gamma^{m-1} e^{-\frac{m\gamma}{\bar{\gamma}}}, \tag{1}$$

where $\bar{\gamma}$ denotes the average SNR. The corresponding cumulative density function is denoted by $F(\gamma)$. For $m = 1$, Nakagami-m fading describes a single-antenna Rayleigh fading channel. Nakagami-m fading with $m > 1$ can also be used to describe single-input multiple-output (SIMO) Rayleigh fading channels with m receive antennas, average SNR $\bar{\gamma}/m$ and maximum-ratio combining at the receiver [9, p. 859].

The transmitters operate with fixed power. In each uplink or downlink phase, the corresponding transmitter has perfect channel state information (CSI), and adapts the rate of the channel code according to this channel state. The data packets and slot lengths in this scenario are assumed to be short. Therefore, modeling the communication as error-free and at a code rate equal to the channel capacity would be highly inaccurate. At finite blocklength of the channel code, transmission errors cannot be completely avoided. It was shown by Polyanskiy et al. [2] that for a given error probability ε, the achievable coding rate in bits per channel use can be approximated by:

$$r(n, \varepsilon, \gamma) \approx \log_2(1+\gamma) - \sqrt{\frac{V}{n}} Q^{-1}(\varepsilon) \log_2 e\,, \tag{2}$$

where the channel dispersion V is given as[1]

$$V = 1 - \frac{1}{(1+\gamma)^2}\,. \tag{3}$$

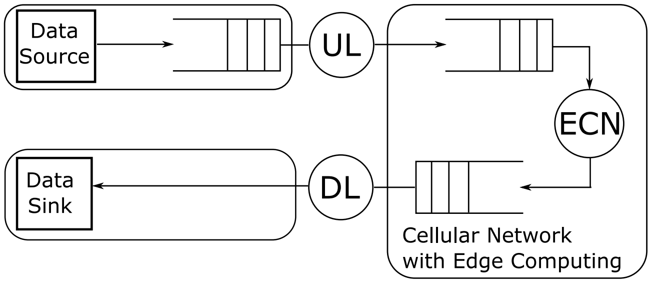

Fig. 1. The edge computing scenario from a queueing perspective.

We assume that (2) holds with equality, that decoding errors are always detected, and that the transmitter will get feedback whether the last packet was decoded successfully. We refer to this model as the finite blocklength (FBL) model. For comparison, we will also show results for a model where the effects of finite blocklength coding are ignored and, consequently, communication at rate $r = \log_2(1+\gamma)$ is assumed to be error-free. For brevity, we will refer to this as the Shannon capacity model.

Due to packet losses and due to time-varying data rates, the data generated at the source and at the ECN may not immediately reach the destination. Furthermore, the data arriving at the ECN cannot always be processed all at once due to limited processing power. Thus, data must be stored in buffers while waiting to be successfully transmitted or processed. This buffering or queueing leads to a random delay of the data. As illustrated by Fig. 1, the queueing system consists of a sequence of three *hops*, with queues at the uplink, the edge computing node, and the downlink. Queueing systems are characterized by random arrival, service, and departure processes. The arrival process $A(t)$ describes the number of bits entering the queue of the first link (the uplink) in time slot t. The service processes $S_l(t)$, $l \in \{\mathtt{UL}, \mathtt{ECN}, \mathtt{DL}\}$, describe how many bits can be potentially transmitted or processed in each time slot by uplink, ECN, or downlink, respectively, and only depend on the channel conditions or processing speeds. The departure process $D(t)$ describes the number of bits that are leaving the last queue (the downlink). In all three queues, the amount of data leaving queue l is upper-bounded by the amount of data waiting in that queue or by the service process $S_l(t)$, whichever is smaller. For the queueing analysis, define the cumulative arrival, service, and departure processes as

$$\mathbf{A}(\tau, t) \triangleq \sum_{i=\tau}^{t-1} A(i),\ \mathbf{S}(\tau, t) \triangleq \sum_{i=\tau}^{t-1} S(i),\ \mathbf{D}(\tau, t) \triangleq \sum_{i=\tau}^{t-1} D(i). \tag{4}$$

First, we consider the case where the ECN outputs c bits for every c bits that are processed as input. In that case, the queueing system is *flow-conserving*, and the total queueing delay of the multi-hop system at time t is defined as [4]:

$$W(t) = \inf\{u \geq 0 : \mathbf{A}(0,t) \leq \mathbf{D}(0, t+u)\} \tag{5}$$

In a time-critical system, long queueing delays must be avoided. However, due to the random nature of the service processes with fading and packet losses, it cannot be guaranteed that the delay never exceeds a certain target delay w. Instead, a probabilistic delay bound is considered. We define

[1] Unlike the authors in [2], we always measure the dispersion in natural units and place a seperate factor $\log_2 e$ in (2), which is measured in bits.

the delay violation probability for a target delay w at time t as

$$p_{\mathrm{v}}(w,t) \triangleq \mathbb{P}\left\{ W(t) > w \right\}. \tag{6}$$

Time-critical systems may still perform fine as long as the probability of exceeding the target delay w is very small, e.g. $p_{\mathrm{v}}(w,t) \leq 10^{-6}$. In addition to queueing delays, there will be further delays for the actual transmissions, for the encoding and decoding of the channel codes, and between the arrival of data packets and the beginning of the next time slot. However, such delays are generally smaller than one time slot and therefore negligible compared to the probabilistic queueing delays, which can occasionally be many time slots long.

For the edge computing node (ECN), we assume a constant processing rate and that processing starts immediately after data reception. When the output data rate of the ECN is different from the input rate, but scaled by a constant factor ϕ, i.e. that for any c input bits processed by the ECN only c/ϕ output bits are generated, then the system is not flow-conserving. Although queueing systems which are not flow-conserving are generally difficult to analyze, our system model with constant scaling of input and output rates can be transformed into an equivalent flow-conserving system model. In the equivalent model, the input and output data rates of the ECN are treated as equal, but the service of the following hop (the downlink) is multiplied by the scaling factor ϕ [10].

III. Queueing Analysis

An upper bound for the delay violation probability $p_{\mathrm{v}}(w,t)$ can be obtained through stochastic network calculus in a transform domain [4]. In this section, we review the key steps for obtaining this bound. We start with the single-hop case, where the description is based on our previous works [7], [11]. Then, we extend the analysis to multi-hop queueing systems by utilizing results from [5].

A. Stochastic Network Calculus in the SNR domain

The delay in (5) is defined in terms of the arrival and departure processes. However, it is difficult to obtain a statistical characterization of the departure process, as it depends on the arrival and service processes in previous time slots. The transmitters and the computing node are assumed to handle all incoming data without additional delay. Then, we can analyze the delay through characterizations of the random arrival and service processes. By taking the exponential of the arrival and service process, the authors in [4] characterized these processes in the exponential domain, also referred to as *SNR domain*. The main advantage of this approach is that it eliminates the logarithm in the channel capacity and allows analysis directly in terms of the channel gain. The arrival and service process in the SNR domain are denoted by calligraphic letters: $\mathcal{A}(t) \triangleq e^{A(t)}$ and $\mathcal{S}(t) \triangleq e^{S(t)}$. Furthermore, the corresponding cumulative processes are denoted as $\boldsymbol{\mathcal{A}}(\tau,t) = e^{\mathbf{A}(\tau,t)}$ and $\boldsymbol{\mathcal{S}}(\tau,t) = e^{\mathbf{S}(\tau,t)}$.

It was shown in [4] that an upper bound on the delay violation probability $p_{\mathrm{v}}(w,t)$ in (6) can be evaluated in terms of the Mellin transforms of $\boldsymbol{\mathcal{A}}(\tau,t)$ and $\boldsymbol{\mathcal{S}}(\tau,t)$. The Mellin transform $\mathcal{M}_{\mathcal{X}}(s)$ of a nonnegative random variable $\mathcal{X}$ is defined as [4]

$$\mathcal{M}_{\mathcal{X}}(s) \triangleq \mathbb{E}\left[\mathcal{X}^{s-1}\right] \tag{7}$$

for a parameter $s \in \mathbb{R}$. Thus, the Mellin transform of the service process in the SNR domain is essentially the same (except for the shift of -1) as the moment-generating function of the service process in the bit domain, which is also used in the delay analysis through effective capacity. The service processes $S_l(t)$ for the uplink and downlink depend on the statistics of the fading channels and are considered mutually independent and identically distributed (i.i.d.) between time slots due to the block-fading assumption. We assume throughout this paper that both the service process at the ECN $S_{\mathtt{ECN}}(t)$ and the arrival process $A(t)$ are constant, although an analysis with random arrivals and random service at the processor is generally possible. The Mellin transforms of the cumulative arrival and all service processes in the SNR domain $\boldsymbol{\mathcal{A}}(\tau,t)$ and $\boldsymbol{\mathcal{S}}_l(\tau,t)$ can then simply be written as $\mathcal{M}_{\mathcal{A}}(s)^{t-\tau}$ and $\mathcal{M}_{\mathcal{S}_l}(s)^{t-\tau}$, i.e. in terms of the Mellin transforms of the incremental arrival and service processes, where we dropped the subscript t because these processes are i.i.d. by assumption. In the single-hop case, the delay violation probability can be bounded as follows: start by choosing any $s > 0$ and first check whether the stability condition $\mathcal{M}_{\mathcal{A}}(1+s)\mathcal{M}_{\mathcal{S}}(1-s) < 1$ holds. If it holds, define the steady-state kernel [4], [7]

$$\begin{aligned}\mathcal{K}\left(s,w\right) &\triangleq \lim_{t\to\infty}\sum_{u=0}^{t}\mathcal{M}_{\mathcal{A}}(1+s)^{t-u}\mathcal{M}_{\mathcal{S}}(1-s)^{t+w-u} \quad (8)\\ &= \frac{\mathcal{M}_{\mathcal{S}}(1-s)^{w}}{1-\mathcal{M}_{\mathcal{A}}(1+s)\mathcal{M}_{\mathcal{S}}(1-s)}. \quad (9)\end{aligned}$$

For any $s > 0$, this kernel is an upper bound for the delay violation probability $p_{\mathrm{v}}(w)$ in steady state, i.e. in the limit $t \to \infty$. Optimizing over s yields the best possible bound for single-hop systems:

$$p_{\mathrm{v}}(w) \leq \inf_{s>0}\left\{\mathcal{K}\left(s,w\right)\right\}. \tag{10}$$

For a multi-hop queueing system with a path of independent, strictly non-identical links described as $\mathbb{L}$, where we label the first hop of the path as K and the last hop as L, a delay bound can be found by recursively computing the kernel [5]:

$$\begin{aligned}\mathcal{K}_{\mathbb{L}}\left(s,w\right) = &\frac{\mathcal{M}_{\mathcal{S}_K}(1-s)}{\mathcal{M}_{\mathcal{S}_K}(1-s)-\mathcal{M}_{\mathcal{S}_L}(1-s)}\mathcal{K}_{\mathbb{L}\backslash\{L\}}\left(s,w\right)\\ &+\frac{\mathcal{M}_{\mathcal{S}_L}(1-s)}{\mathcal{M}_{\mathcal{S}_L}(1-s)-\mathcal{M}_{\mathcal{S}_K}(1-s)}\mathcal{K}_{\mathbb{L}\backslash\{K\}}\left(s,w\right),\end{aligned} \tag{11}$$

where $\mathbb{L}\backslash\{L\}$ describes the path where the server L and queue L are removed. When the path consists only of a single link, the kernel is given by the single-hop kernel (9). In our model, the kernel for the path with the three links $\mathbb{L} = \{\mathtt{UL},\mathtt{ECN},\mathtt{DL}\}$ can thus be computed from the kernels for $\mathbb{L} = \{\mathtt{UL},\mathtt{ECN}\}$ and $\mathbb{L} = \{\mathtt{ECN},\mathtt{DL}\}$. These can in turn be computed from the kernels for $\mathbb{L} = \{\mathtt{UL}\}$, $\mathbb{L} = \{\mathtt{ECN}\}$ and $\mathbb{L} = \{\mathtt{DL}\}$, which are the single-link kernels given by (9).

IV. Delay Bounds at Finite Blocklength

An upper bound on the delay violation probability can be computed through the kernel (11), which requires computation of $\mathcal{M}_{\mathcal{A}}(s)$ and $\mathcal{M}_{\mathcal{S}_l}(s)$, i.e. of the Mellin transforms of the arrival process and the service processes in the SNR domain. For the wireless links ($l \in \{\texttt{UL}, \texttt{DL}\}$), we assume that the rate adaptation scheme operates at a fixed error probability ε at all values of the SNR γ. The service process is given as the number of successfully transmitted bits in each time slot, i.e. the service is either $n \cdot r(n, \varepsilon, \gamma)$, with probability $(1-\varepsilon)$, or equal to zero, with probability ε. In case the ECN transforms the flow by a factor ϕ, i.e. outputs only c/ϕ bits for every c input bits, we simply multiply the service in the downlink by ϕ in order to obtain an equivalent flow-conserving queueing model. For consistent notation, define $\tilde{\phi} = 1$ in the uplink and $\tilde{\phi} = \phi$ in the downlink and multiply the service by $\tilde{\phi}$. After converting to the SNR domain, the Mellin transform of $\mathcal{S}_l$, $l \in \{\texttt{UL}, \texttt{DL}\}$ is given as [7]:

$$\mathcal{M}_{\mathcal{S}_l}(s) = (1-\varepsilon)\mathbb{E}_\gamma\left[\left(e^{\tilde{\phi} n \cdot r(n,\varepsilon,\gamma)}\right)^{s-1}\right] + \varepsilon \tag{12}$$

$$= (1-\varepsilon)\mathbb{E}_\gamma\left[\left(e^{r(n,\varepsilon,\gamma)\ln 2}\right)^{\frac{\tilde{\phi} n(s-1)}{\ln 2}}\right] + \varepsilon \tag{13}$$

where the expected value is taken with respect to the fading distribution of the SNR γ. For fixed ε, the rate according to (2) would become smaller than zero when the SNR is below a certain threshold y_0, which we avoid by setting the rate equal to zero in this range. To shorten the notation, define $P \triangleq Q^{-1}(\epsilon)/\sqrt{n}$ and $\theta \triangleq \tilde{\phi} n(1-s)/\ln 2$. Thus:

$$\mathcal{M}_{\mathcal{S}}(s) = (1-\varepsilon)\left(F(y_0) + \int_{y_0}^{\infty}\left(\frac{1+\gamma}{e^{\sqrt{V}P}}\right)^{-\theta} f(\gamma)d\gamma\right) + \varepsilon. \tag{14}$$

In [7], we analyzed the integral

$$B(\theta) \triangleq \int_{y_0}^{\infty}\left(\frac{1+\gamma}{e^{\sqrt{V}P}}\right)^{-\theta} f(\gamma)d\gamma \tag{15}$$

for Rayleigh block-fading channels, where the SNR γ is exponentially distributed. At high average SNR, the dispersion V is very close to 1, and the integral can be solved in terms of the upper incomplete gamma function

$$\Gamma(s,x) = \int_x^{\infty} t^{s-1}e^{-t}dt. \tag{16}$$

For medium and low SNR, we applied the following series expansion for $-1 \leq x \leq 1$ to $\sqrt{V}$ [12, (1.110)]:

$$(1-x)^{\alpha} = \sum_{j=0}^{\infty}\binom{\alpha}{j}(-x)^j, \tag{17}$$

which results in

$$\sqrt{V} = \left(1 - \frac{1}{(1+\gamma_i)^2}\right)^{1/2} = 1 - \sum_{j=1}^{\infty}\frac{b_j}{(1+\gamma_i)^{2j}}, \tag{18}$$

where we used the fact that for $\alpha = 1/2$, the sign of the binomial coefficient is alternating in j for $j \geq 1$, and defined

$$b_j \triangleq \left|\binom{1/2}{j}\right| = \left|\frac{\left(\frac{1}{2}\right)\left(\frac{1}{2}-1\right)\dots\left(\frac{1}{2}-j+1\right)}{j!}\right|. \tag{19}$$

Applying this expansion to $B(\theta)$ yields [7]:

$$B(\theta) = \int_{y_0}^{\infty}\left(\frac{1+\gamma}{e^P}\right)^{-\theta}\prod_{j=1}^{\infty}e^{\frac{-b_jP\theta}{(1+\gamma)^{2j}}}f(\gamma)d\gamma. \tag{20}$$

We approximate the product by limiting it to J factors and expand each factor through its Taylor series [7]:

$$\prod_{j=1}^{J}e^{\frac{-b_jP\theta}{(1+\gamma)^{2j}}} = \prod_{j=1}^{J}\left(\sum_{k_j=0}^{\infty}\frac{1}{k_j!}\left(\frac{-b_jP\theta}{(1+\gamma)^{2j}}\right)^{k_j}\right) \tag{21}$$

$$= \sum_{k_1=0}^{\infty}\cdots\sum_{k_J=0}^{\infty}\frac{\prod_{j=1}^{J}\frac{1}{k_j!}\left(-b_jP\theta\right)^{k_j}}{(1+\gamma)^{\sum_{j=1}^{J}2jk_j}}. \tag{22}$$

Collecting and combining all terms where the exponent of $(1+\gamma)$ is equal to 2ν results in:

$$\prod_{j=1}^{J}e^{\frac{-b_jP\theta}{(1+\gamma)^{2j}}} = \sum_{\nu=0}^{\infty}\frac{C(\nu)}{(1+\gamma)^{2\nu}}, \tag{23}$$

with

$$C(\nu) \triangleq \sum_{k_1=0}^{\infty}\cdots\sum_{k_J=0}^{\infty}\prod_{j=1}^{J}\frac{1}{k_j!}\left(-b_jP\theta\right)^{k_j}\mathbb{1}_{\left\{\sum_{j=1}^{J}jk_j=\nu\right\}}. \tag{24}$$

Considering only the denominator, the terms can be expected to decrease exponentially in ν. Therefore, we proposed in [7] to specify a value $\hat{\nu}$ and consider only terms with $\nu \leq \hat{\nu}$. For an average SNR above 0 dB, the integral $B(\theta)$ was found to be well approximated with $2\nu \leq 20$ or $\hat{\nu} = 10$. In this case, even though each $C(\nu)$ consists of sums over $k_1, \dots, k_{10}$, the resulting complexity is surprisingly low, as there are at most 42 terms which satisfy $\sum_{j=1}^{10} jk_j = \nu$. Applying the approximation to $B(\theta)$ leads to:

$$B(\theta) \approx \sum_{\nu=0}^{\hat{\nu}}C(\nu)e^{P\theta}\int_{y_0}^{\infty}(1+\gamma)^{-\theta-2\nu}f(\gamma)d\gamma. \tag{25}$$

For Rayleigh fading channels, the PDF $f(\gamma)$ is exponential, and all factors involving the SNR γ can be brought into the form $(1+\gamma)/\bar{\gamma}$. Then, for each ν, the integral is given by the upper incomplete Gamma function (16), as we showed in [7].

For Nakagami-m fading channels, $f(\gamma)$ is given by (1), and the analysis is more involved because of the factor γ^{m-1}, which cannot be easily combined with the factor $(1+\gamma)^{-s}$. However, a solution can be obtained using the conversion

$$\gamma^{m-1} = (\gamma+1-1)^{m-1} = (1+\gamma)^{m-1}\left(1 - \frac{1}{1+\gamma}\right)^{m-1} \tag{26}$$

and then applying the series expansion (17):

$$\gamma^{m-1} = (1+\gamma)^{m-1}\sum_{\mu=0}^{\infty}\binom{m-1}{\mu}(-1)^{\mu}\left(\frac{1}{1+\gamma}\right)^{\mu}. \tag{27}$$

Then, the integral $B(\theta)$ is approximated as

$$B(\theta) \approx \sum_{\mu=0}^{\infty}\sum_{\nu=0}^{\hat{\nu}} \frac{m^m \binom{m-1}{\mu}(-1)^{\mu}}{\bar{\gamma}^m \Gamma(m)} C(\nu) e^{P\theta} \cdot \int_{y_0}^{\infty} (1+\gamma)^{-\theta-2\nu+m-\mu-1} e^{-\frac{m\gamma}{\bar{\gamma}}} d\gamma \tag{28}$$

$$= \sum_{\mu=0}^{\infty}\sum_{\nu=0}^{\hat{\nu}} \frac{\binom{m-1}{\mu}(-1)^{\mu}}{\Gamma(m)} C(\nu) \left(\frac{\bar{\gamma}}{m}\right)^{-s-2\nu-\mu} e^{P\theta} \cdot \int_{y_0}^{\infty} \left(\frac{m(1+\gamma)}{\bar{\gamma}}\right)^{-\theta-2\nu+m-\mu-1} \frac{m}{\bar{\gamma}} e^{-\frac{m\gamma}{\bar{\gamma}}} d\gamma \tag{29}$$

$$= \sum_{\mu=0}^{\infty}\sum_{\nu=0}^{\hat{\nu}} \frac{\binom{m-1}{\mu}(-1)^{\mu}}{\Gamma(m)} C(\nu) \left(\frac{\bar{\gamma}}{m}\right)^{-s-2\nu-\mu} e^{P\theta} \cdot e^{\frac{m}{\bar{\gamma}}} \Gamma\left(-\theta-2\nu-\mu+m, \frac{m(1+y_0)}{\bar{\gamma}}\right), \tag{30}$$

where in the last step, we performed a change of variables and applied (16). When m is not an integer, the sum over μ must be truncated at some point. However, when m is an integer, the sum over μ is non-zero only for $\mu \leq m-1$ and does not need to be truncated.

V. Numerical Analysis

In this section, we start with a validation of the analytical bounds on the delay and on the delay violation probability. Then, we study the effect of finite blocklength coding on the overall delay in different scenarios.

A. Validation

For validation of the analytical bounds on the delay violation probability $p_{\mathrm{v}}(w)$, we compare the analytical upper bounds obtained through the multi-hop kernel (11) with simulation results. For the simulations, we generate instances of the arrival and service processes according to the system model, and then obtain the empirical distribution of the random delay as defined in (5). The systems were simulated for at least 10^9 time slots. In Fig. 2, we show the probability $p_{\mathrm{v}}(w)$ and the corresponding analytic bound for a target delay of $w = 10$ time slots. For this plot, we assume that the total number N of symbols in each time slot is constant with $N = 500$, but we vary the number of symbols n_{UL} assigned to the uplink. The downlink will then use the remaining $n_{\mathrm{DL}} = N - n_{\mathrm{UL}}$ symbols. We start with symmetric channel conditions[2] in uplink and downlink with $\bar{\gamma}_{\mathrm{UL}} = \bar{\gamma}_{\mathrm{DL}} = 5$ dB, $m_{\mathrm{UL}} = m_{\mathrm{DL}} = 1$ (Rayleigh fading) and assume that the edge computing node is powerful and can process twice the amount of data arriving in every slot, i.e. $S_{\mathrm{ECN}} = 2A$. In order to choose the error probabilities ϵ_{UL} and ϵ_{DL}, we perform a simple line search and select the ϵ which minimizes the single-hop delay bound (9) for every choice of n_{UL} and n_{DL}.

We confirm in all cases that the empirical delay violation probability obtained from simulations is below the analytical bound. Although the difference between simulations and bound can be more than an order of magnitude, the overall shapes of the curves coincide, and the bounds correctly predict that the minimum delay violation probability is attained at $n_{\mathrm{UL}} = n_{\mathrm{DL}} = 250$.

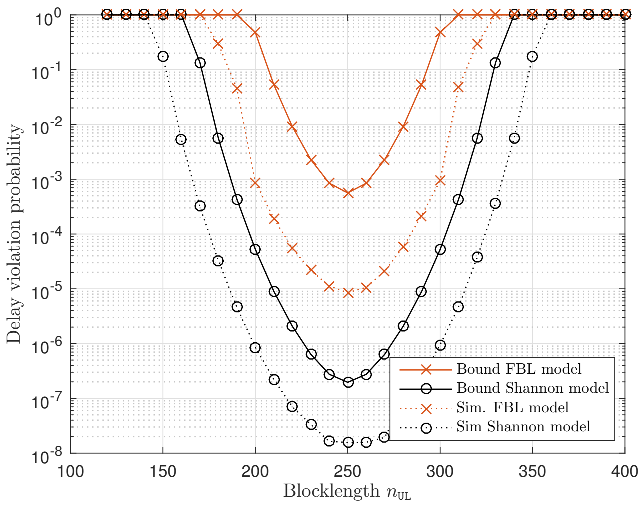

Fig. 2. Delay violation probability $p_{\mathrm{v}}(w)$ for target delay $w = 10$. $\bar{\gamma}_{\mathrm{UL}} = \bar{\gamma}_{\mathrm{DL}} = 5$ dB, $m_{\mathrm{UL}} = m_{\mathrm{DL}} = 1$, $A = 250$ bits, $S_{\mathrm{ECN}} = 2A$, $\phi = 1$.

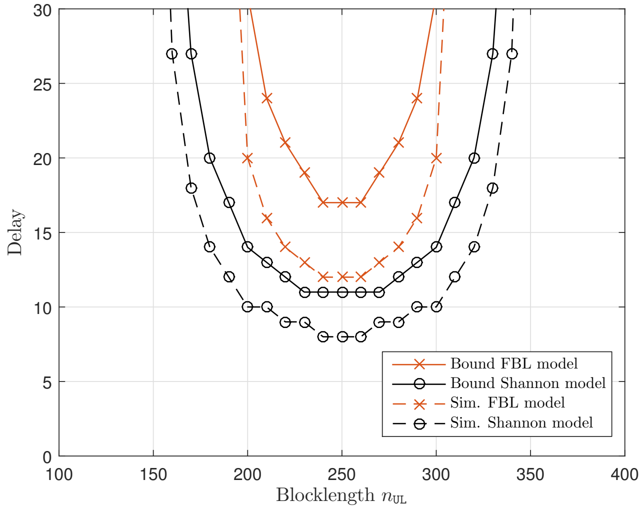

Fig. 3. Minimum delay w such that $p_{\mathrm{v}}(w) < 10^{-6}$. $\bar{\gamma}_{\mathrm{UL}} = \bar{\gamma}_{\mathrm{DL}} = 5$ dB, $m_{\mathrm{UL}} = m_{\mathrm{DL}} = 1$, $A = 250$ bits, $S_{\mathrm{ECN}} = 2A$, $\phi = 1$.

B. Delays for Different Parameters

Instead of using the delay violation probability for a given delay w as a metric, we can also look at the smallest value of the delay w such that the delay violation probability $p_{\mathrm{v}}(w)$ is below a given target value, e.g. 10^{-6}. This delay w is shown in Fig. 3 for the same parameters as in Fig. 2. We observe that the analytic delay bounds are still valid upper bounds for the actual delay as obtained from simulations, and that the bounds correctly predict the regions where the delay is lowest and where the delay increases. Furthermore, we see that the region of n_{UL} where the delay is short is significantly smaller when finite blocklength effects are considered.

In order to decrease the delay, we investigate the effect of adding a second antenna to the base station and applying

[2] Since the kernel (11) can only be computed for non-identical links, we set a slightly different value (5.001 dB) for the SNR in the downlink.

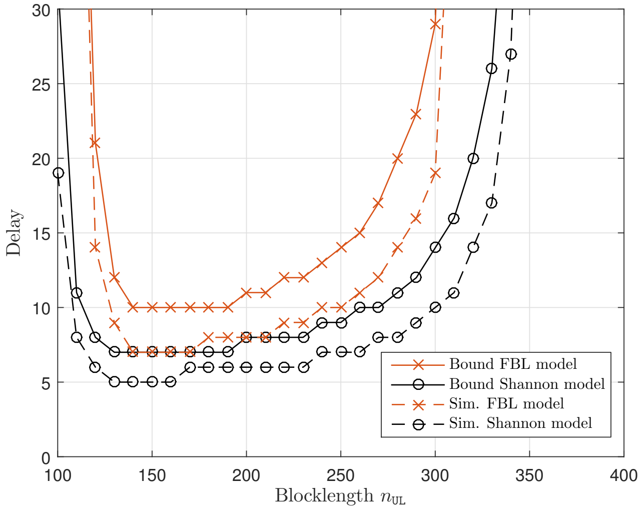

Fig. 4. Minimum delay w such that $p_{\text{v}}(w) < 10^{-6}$. $\bar{\gamma}_{\text{UL}} = 8$ dB, $\bar{\gamma}_{\text{DL}} = 5$ dB, $m_{\text{UL}} = 2$, $m_{\text{DL}} = 1$, $A = 250$ bits, $S_{\text{ECN}} = 2A$, $\phi = 1$.

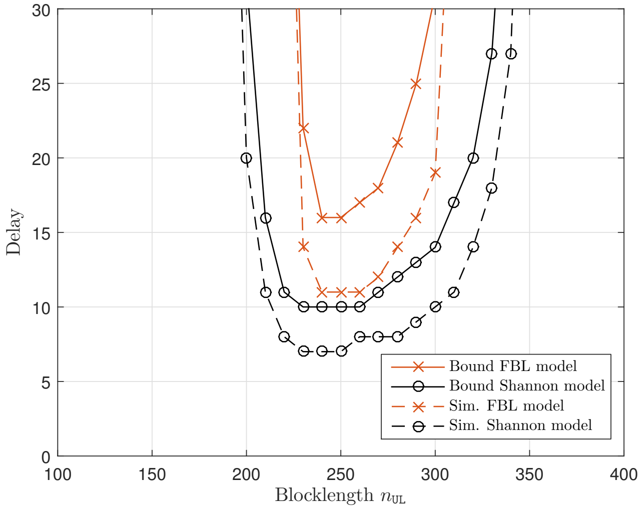

Fig. 5. Minimum delay w such that $p_{\text{v}}(w) < 10^{-6}$. $\bar{\gamma}_{\text{UL}} = 8$ dB, $\bar{\gamma}_{\text{DL}} = 5$ dB, $m_{\text{UL}} = 2$, $m_{\text{DL}} = 1$, $A = 500$ bits, $S_{\text{ECN}} = 2A$, $\phi = 2$.

maximum ratio combining in the uplink, which would result in a diversity and power gain that can be modeled in the Nakagami-m fading model with $m_{\text{UL}} = 2$ and with a 3dB gain in the average SNR $\bar{\gamma}$ of the uplink. The results[3] are shown in Fig. 4. As expected, the delays decrease significantly. We observe that even in this non-symmetric case, the shapes of the simulated and analytical delay curve match fairly well.

The increased uplink performance can be used to transmit more data in the uplink. For example, a wireless sensor network could transmit more accurate sensor readings. In Fig. 5, we double the arrival rate A to 500 bits. We assume that the ECN can still handle twice the amount of data generated in each slot, i.e. $S_{\text{ECN}} = 2A$. However, the average output data rate of the ECN stays the same, so the ECN produces only 250 output bits for 500 input bits. For the queueing analysis, we therefore set the service scaling factor in the downlink to $\phi = 2$. Compared to Fig. 4, the delay increases significantly when the uplink has limited resources ($n_{\text{UL}} < 240$), due to the increased load in the uplink. However, as soon as the uplink has sufficient resources ($n_{\text{UL}} > 240$), the delay is not much higher than seen in Fig. 4. We observe that the shape of the delay curve is asymmetrical. A small change towards shorter n_{UL} leads to an extreme increase in the delay. A possible explanation is that the Nakagami-m fading in the uplink has a more deterministic behavior than the Rayleigh fading channel in the downlink. A deterministic server can either handle all incoming deterministic arrivals immediately, or it cannot provide enough service, causing infinite delay. Similarly, we observe a sharp on-off transition when we decrease the blocklength of the uplink in Fig. 5.

VI. Conclusions and Future Work

We have demonstrated that the queueing delay in edge computing scenarios can be analyzed with stochastic network calculus. Our analytical results and simulations show that finite blocklength effects can have a significant impact on the delay in such systems. While we analyzed the delay for data from a single source, future work should examine the delay when the computing node is shared among different traffic flows with varying and independent arrival patterns.

References

[1] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, “Fog computing and its role in the internet of things,” in *Proc. First Ed. of the MCC Workshop on Mobile Cloud Computing*. ACM, 2012, pp. 13–16.

[2] Y. Polyanskiy, H. V. Poor, and S. Verdú, “Channel coding rate in the finite blocklength regime,” *IEEE Trans. Inf. Theory*, vol. 56, no. 5, pp. 2307–2359, May 2010.

[3] D. Wu and R. Negi, “Effective capacity: a wireless link model for support of quality of service,” *IEEE Trans. Wireless Commun.*, vol. 2, no. 4, pp. 630–643, Jul. 2003.

[4] H. Al-Zubaidy, J. Liebeherr, and A. Burchard, “Network-Layer Performance Analysis of Multihop Fading Channels,” *IEEE/ACM Trans. Netw.*, vol. 24, no. 1, pp. 204–217, Feb. 2016.

[5] N. Petreska, H. Al-Zubaidy, R. Knorr, and J. Gross, “On the recursive nature of end-to-end delay bound for heterogenous networks,” in *IEEE Int. Conf. on Communications (ICC)*, Jun. 2015.

[6] M. C. Gursoy, “Throughput analysis of buffer-constrained wireless systems in the finite blocklength regime,” *EURASIP Journal on Wireless Communications and Networking*, Dec. 2013.

[7] S. Schiessl, J. Gross, and H. Al-Zubaidy, “Delay analysis for wireless fading channels with finite blocklength channel coding,” in *Proc. 18th ACM Int. Conf. on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM)*. ACM, 2015, pp. 13–22.

[8] Y. Li, M. C. Gursoy, and S. Velipasalar, “Throughput of two-hop wireless channels with queueing constraints and finite blocklength codes,” in *Information Theory (ISIT), 2016 IEEE International Symposium on*. IEEE, 2016, pp. 2599–2603.

[9] J. Proakis and M. Salehi, *Digital Communications*, 5th ed. McGraw-Hill Education, 2007.

[10] H. M. Al-Zubaidy, G. Dán, and V. Fodor, “Performance of in-network processing for visual analysis in wireless sensor networks,” in *14th IFIP Networking Conference*, 2015.

[11] S. Schiessl, M. Skoglund, H. Al-Zubaidy, and J. Gross, “Analysis of wireless communications with finite blocklength and imperfect channel knowledge,” *arXiv preprint arXiv:1608.08445 [cs.IT]*, 2016.

[12] I. Gradshteyn and I. Ryzhik, *Table of Integrals, Series, and Products*, 7th ed. Elsevier, 2007.

[3] The results for $n_{\text{UL}} < 200$ should be treated with caution, as the approximation (2) becomes inaccurate in that range [2].