

Approximation models for the evaluation of TCP/AQM networks

Dariusz MAREK¹, Adam DOMAŃSKI¹, Joanna DOMAŃSKA², Jakub SZYGUŁA¹*,
Tadeusz CZACHÓRSKI², Jerzy KLAMKA², and Katarzyna FILUS²

¹ Faculty of Automatic Control, Electronics and Computer Science, Department of Distributed Systems and Informatic Devices, Silesian University of Technology, Akademicka 16, 44-100 Gliwice, Poland

² Institute of Theoretical and Applied Informatics, Polish Academy of Sciences, Bałtycka 5, 44-100 Gliwice, Poland

Abstract. The article proposes a model in which Diffusion Approximation is used to analyse the TCP/AQM transmission mechanism in a multi-node computer network. In order to prevent traffic congestion, routers implement AQM (Active Queue Management) algorithms. We investigate the influence of using RED-based AQM mechanisms and the fractional controller PI^γ on the transport layer. Additionally, we examine the cases in which the TCP and the UDP flows occur and analyse their mutual influence. Both transport protocols used are independent and work simultaneously. We compare our solution with the Fluid Flow approximation, demonstrating the advantages of Diffusion Approximation.

Key words: Diffusion Approximation; Active Queue Management; congestion control; dropping packets; Fluid Flow Approximation; non-integer order PI^γ controller; G/G/1/N queueing model.

1. INTRODUCTION

The purpose of communication protocols used in computer networks is to ensure efficient and correct transmission between devices. It can be done in many ways. This article focuses on a specific group of mechanisms increasing transmission efficiency by minimising possible packet losses due to buffer overflows at transmission nodes. The Internet Engineering Task Force (IETF) suggests using Active Queue Management (AQM) algorithms for this purpose. These algorithms allow to preemptively drop packets, even if the buffer is not full, but there is a risk of overflowing it soon. A significant increase in Internet traffic and poor congestion control can result in partial or complete degradation of network performance [1]. Therefore, numerous works discuss Active Queue Management [1,2] as a remedy. Malicious actors can cause the increase in traffic intensity via Distributed Denial of Service, which becomes an increasingly frequent problem of the modern Internet [3]. Therefore, the analysis and modelling of TCP/AQM are also interesting for cybersecurity.

Since the creation of the primary AQM mechanism (Random Early Detection, RED), many modifications have been proposed, e.g. BLUE algorithm [4], hyperbola RED (HRED) [5], and Yellow algorithm [6]. The alternatives to RED algorithms can be, e.g. CoDel and PIE [7]. However, although many years have passed since the RED mechanism was introduced, this mechanism is still the most popular AQM scheme [8,9]; therefore it can serve as a baseline AQM.

The mechanism of RED algorithms is based on a packet loss probability function. The loss probability generally depends on the queue moving average of the length and affects the number of packets to be dropped. In most cases, increasing queue occupancy increases the probability of incoming packet drop. Different probability functions can be used. In the case of the RED algorithm, [10] it is a linear function. For NLRED algorithm [11, 12] it can be a quadratic or polynomial function. For more sophisticated cases, e.g. [13], the probability of packet drop can be calculated based on the PI controller response.

One of the advantages of AQM algorithms is that they cooperate with the transport layer protocol – TCP, which has its built-in congestion window control mechanisms. When possible, these mechanisms increase the transmission rate and decrease it when a data segment is lost. There are many congestion window control algorithms [14]. The TCP NewReno [15] algorithm is the most widely used in Internet transmissions. Random packet deletion provided by AQM allows avoiding so-called global synchronisation effect, which involves slowing down transmission by multiple TCP transmitters at the same time in case of losses caused by exceeding maximum queue size in nodes. The combination of these both mechanisms: AQM determining packet losses in IP routers and TCP congestion window reacting on the loss, allows us to eliminate congestion effectively in the network [16].

A dynamic TCP model is necessary to reflect the time-dependent behaviour of flows and queues. The most commonly used and the simplest mechanism to model TCP protocol behavior in cooperation with AQM is the Fluid Flow approximation [17–20]. As an alternative, simulation models can also be used. However, they are very time-consuming in the case of

*e-mail: jakub.szygula@polsl.pl

Manuscript submitted 2022-03-18, revised 2022-05-30, initially accepted for publication 2022-06-26, published in August 2022.

transient-state analysis, crucial in the evaluation of Internet behaviour [21].

In this paper, we present and use a Diffusion Approximation network model to evaluate network transmission efficiency based on the two mechanisms presented: a TCP NewReno-based transmission rate control mechanism and a preemptive packet deletion mechanism. The model presented in this paper is a multi-node extension of our earlier models presented in [22].

The Diffusion Approximation is a second-order approximation (it takes into account two first moments of interarrival and service times distributions), and it provides the distributions of queue lengths and waiting times; therefore it is more accurate than Fluid Flow Approximation, based only on mean values.

To create a diffusion TCP/AQM network model, we used the generic network models proposed in [23–25] having no control mechanisms. Three AQM algorithms were selected for the study: the basic RED algorithm and two of its modifications: NLRED and PI^λ .

The rest of the paper is organised as follows. In Section 2 one can find the literature overview regarding AQM techniques based on the control theory methods and an embedding of our solution in the area of diffusion modelling. Section 3 gives a brief description of the AQM techniques used in this paper. Section 4 describes our diffusion model of the non-integer PI^γ controller with TCP/UDP streams. In Section 5 numerical results are presented. Section 6 concludes the article.

2. BACKGROUND AND RELATED WORK

Congestion management is one of the critical issues in the domain of computer networks [16]. The most effective solutions are based on the TCP congestion window (CWND) management mechanism [26]. For UDP-based transmissions, mechanisms based on choke packets can be used [27]. These mechanisms are adequate for bandwidth control; however, they introduce transmission delays. Despite their disadvantages, they are still used, especially in wireless sensor networks (WSNs), due to their limited computational resources, storage, energy, and communication bandwidth requirements [27, 28]. The use of choke packets in TCP networks may be ineffective due to incompatibility with the backpressure queue size based routing [26]. The possible cooperation of TCP and backpressure mechanisms would require changes in the management mechanism of CWND in the TCP protocol [26].

AQM mechanisms are well adapted to work with the TCP protocol. Their cooperation can be modelled as a closed-loop control algorithm with AQM as a controller. Control theory traditional methods may be applied to investigate and improve proposed solutions stability, efficiency, and performance.

Paper [29] introduces a Fluid Flow dynamic model of TCP/RED based on stochastic differential equations. The presented model allowed modelling of the queue behaviour and the CWND mechanism for the TCP protocol. The traffic having both the TCP and the UDP streams has been considered in [30]. One of the properties of this model was the continuous merging of CBR/UDP traffic with the TCP stream. The paper [19] intro-

duced a Fluid Flow model that allows TCP and UDP streams to be treated independently. The streams for this model are also time-constrained. In the paper [31] a Fluid Flow model considering multiple AQM nodes was presented; [32] extends it to a vast network with topology mapped from the real Internet.

The control theory has resulted in the emergence of entirely new AQM algorithms based on the response from a PID controller or one of its variants. The use of a Proportional-Integral (PI) controller for low-frequency dynamics was proposed in the paper [17]. The article [33] makes a comparison of adaptive Proportional (P) and Proportional-Integral (PI) controllers; the PI controller was found to be more adaptive to large fluctuations in network traffic. A new algorithm from the RED family was proposed in [34]. The proposed Proportional-Derivative-RED (PD-RED) mechanism demonstrated better adaptability to network traffic than the previously known Adaptive RED algorithm. The acceleration of the controller's response to changing network conditions was achieved using the Proportional-Integral-Differential (PID) algorithm [35]. Generally, the advantage of PID controllers is their computation, and implementation simplicity [36]. An attempt to maintain this simplicity in more complex mechanisms has been described in the article [37]. The authors present a self-tuning compensated PID controller.

All the above articles are based on traditional integer-order differentiation and integration. As shown in [38], the behaviour of many real-world dynamic systems can be modelled more accurately using a fractional dynamic model. Such models provide better performance than the conventional integer-order ones. The first attempt to use the fractional-order PI controller in an AQM strategy was proposed in the article [39]. However, the author only focused on determining the range where the PI^γ controller ensures a given modulus margin (inverse of the H_∞ norm of the sensitivity function). A fully working AQM model based on PI^γ controller was proposed in [40], where Fluid Flow approximation and simulation models were used to demonstrate the correctness of the proposed mechanism. A method of correct selection of $PI^\alpha D^\beta$ controller parameters with the use of simulations was described in [41]. The article [22] presents a new model of the PI^γ controller based on a Diffusion Approximation. The article proves that the diffusion model allows us to obtain more detailed information on transmission delays and more accurately reflects real network traffic than the Fluid Flow model.

The tutorial [42] shows how diffusion is used in the analysis of traffic control mechanisms implemented in ATM networks. The use of the model in wireless networks has been described in [43]. TCP/RED mechanism has been first modelled in the article [44]. The CWND TCP mechanism and its ability to interoperate with various AQMs were described in [22]. A combined diffusion and simulation model for the analysis of TCP/AQM mechanisms has been proposed in [45]. A network traffic model based on independent TCP and UDP streams has been proposed in [46].

The models presented above can be used to analyse how Internet traffic (understood as a collection of TCP and UDP streams) affects the behaviour of the AQM queue in a sin-

gle network node. In this paper, we present a TCP/UDP/AQM model that reflects the behaviour of a more significant number of routers. The flexibility of the proposed model allows us to create network models with various and arbitrarily complex topologies and obtain more accurate results than in the case of Fluid models. To the best of our knowledge, no diffusion model proposed in the literature can be used for the TCP/UDP/AQM networks analysis.

3. AQM MECHANISMS BASED ON PACKET DROPPING PROBABILITY FUNCTION

The rate control mechanisms built into the TCP protocol are based on a packet loss mechanism. In case of loss, the transmitter decreases the transmission rate by reducing the size of the congestion window. If valid acknowledgments of the transmission are received, the transmission rate is increased. In static queue management, packet loss is a consequence of buffer overflow. In AQM mechanisms, packets can be dropped earlier, even when the queue is not yet overflowing. This drop is random and usually depends on the queue length. In our paper, we use three different mechanisms that differ in the calculation of packet loss probability: RED, NLRED, and PI^γ .

The RED family algorithms (which include RED and NLRED) use the weighted moving average avg computed at packets' arrival; for packet i :

$$\text{avg}_i = (1 - w)\text{avg}_{i-1} + wq_i,$$

where avg_{i-1} is used to denote the moving average determined at the arrival of the previous packet, whereas q_i denotes the queue length.

As avg value increases, the probability grows linearly between two thresholds Min_{th} and Max_{th} , from 0 to p_{\max} .

$$P_{\text{RED}}(\text{avg}) = p_{\max} \frac{\text{avg} - \text{Min}_{th}}{\text{Max}_{th} - \text{Min}_{th}}. \quad (1)$$

Finally, it becomes $p_{\text{RED}} = 1$ for $\text{avg} > \text{Max}_{th}$.

In case of NLRED (non-linear RED) algorithm, the linear packet drop function is replaced by a polynomial function [11]. The packet dropping probability function is based on the third-degree polynomials:

$$p(\text{avg}, a_1, a_2, p_{\max}) = \begin{cases} 0 & \text{for } \text{avg} < \text{Min}_{th}, \\ \varphi_0(\text{avg}) + a_1 \varphi_1(\text{avg}) + a_2 \varphi_2(\text{avg}) & \text{for } \text{Min}_{th} \leq \text{avg} \leq \text{Max}_{th}, \\ 1 & \text{for } \text{avg} > \text{Max}_{th}, \end{cases} \quad (2)$$

where a_1, a_2 are NLRED coefficients and:

$$\varphi_0(\text{avg}) = p_{\max} \frac{\text{avg} - \text{Min}_{th}}{\text{Max}_{th} - \text{Min}_{th}}, \quad (3)$$

$$\varphi_1(\text{avg}) = (\text{avg} - \text{Min}_{th})(\text{Max}_{th} - \text{avg}), \quad (4)$$

$$\varphi_2(\text{avg}) = (\text{avg} - \text{Min}_{th})^2(\text{Max}_{th} - \text{avg}). \quad (5)$$

In the case of the third solution PI^γ probability p is determined based on a comparison of the current queue q_i and the queue q_0 we want to maintain. This difference, called "error" in the control theory, affects the controller response, treated as packet dropping probability. In our case, we use the fractional-order proportional-integral PI^γ controller [13].

In this case the loss probability p_i of a packet i equals:

$$p_i = \max \{0, -(K_P e_i + K_I \Delta^\gamma e_i)\}. \quad (6)$$

This probability depends on K_P, K_I (the proportional and integral term respectively), the order of integration γ and the error $e_i = q_i - q$, where: e_i is an error in current slot, q_i is actual queue length and q (setpoint) is desired queue length. Articles [40, 41, 47, 48] discuss the selection and impact of these variables on controller effectiveness.

The method of non-integer order discrete differ-integral calculation is analogous to the generalization employed in the Grünwald-Letnikov (GrLET) formula [49, 50].

For a given sequence $f_0, f_1, \dots, f_j, \dots, f_k$

$$\Delta^\gamma f_k = \sum_{j=0}^k (-1)^j \binom{\gamma}{j} f_{k-j}, \quad (7)$$

where $\gamma \in \mathbb{R}$ is usually a non-integer fractional order, f_k is a differentiated discrete function, and $\binom{\gamma}{j}$ is a generalised Newton symbol which can be defined as follows:

$$\binom{\gamma}{j} = \begin{cases} 1 & \text{for } j = 0, \\ \frac{\gamma(\gamma-1)(\gamma-2)\dots(\gamma-j+1)}{j!} & \text{for } j = 1, 2, \dots \end{cases} \quad (8)$$

4. THE APPROXIMATION MODELS OF TCP/AQM BEHAVIOUR

This section describes the basics of the Fluid Flow and Diffusion Approximation models and how they can model TCP/AQM mechanisms. Network extensions to both models will also be described to evaluate AQM queues over multiple transmission nodes.

4.1. Fluid Flow model

The Fluid Flow model was first presented in [29]. This model demonstrates TCP protocol dynamics in a simplified way (the model ignores the TCP timeout mechanisms). It is based on the following nonlinear differential equation describing the evolution of the congestion window size [51]:

$$\frac{dW_i^{TCP}(t)}{dt} = \frac{1}{R_i(t)} - \frac{W_i^{TCP}(t) W_i^{TCP}(t - R_i(t))}{2 R_i(t - R_i(t))} p(t - R_i(t)) \quad (9)$$

together with the balance equation for the congested router:

$$\frac{dq(t)}{dt} = \sum_{i=1}^N \frac{W_i^{TCP}(t)}{R_i(t)} - C, \quad (10)$$

where:

W_i – expected size of the TCP congestion window (in packets) for a flow i , it defines the number of packets that may be sent without waiting for the acknowledgements of previous packets reception,

R_i – round-trip time, can be calculated as $R_i = q/C + T_p$, whereas the sum $\sum W_i/R_i$ denotes the total input flow to the examined congested router,

q – queue length (in packets),

C – link capacity (packets/time unit), the constant output flow of the router,

T_p – propagation delay,

N – number of TCP sessions passing through the router,

p – packet dropping probability.

Model of UDP streams illustrates a CBR stream with assumed number of packet being sent per time unit. We refer to such a traffic because it is the most demanding traffic for the data link. The sending rate of the i -th UDP stream is approximated by the following equation:

$$W_i^{UDP}(t) = U. \quad (11)$$

The maximum values of q and W are dependent on the buffer capacity and maximum size of the window. Such a Fluid Flow model may be extended to any network topology with any number of TCP flows [52]. This method allows us to evaluate the behaviour of TCP/AQM mechanisms for large networks. The article [32] presents a numerical example for a network topology with 134 023 nodes and 50 000 flows.

4.2. Diffusion Approximations

Diffusion Approximation can be applied in the queueing theory (e.g. [24, 53–55]) to determine a queue distribution. In this approximation method the discrete process representing a queue length is replaced by a continuous diffusion process $X(t)$. The probability density function (pdf) of this process $f(x, t; x_0)$ [54]:

$$f(x, t; x_0) dx = P[x \leq X(t) < x + dx | X(0) = x_0] \quad (12)$$

can be defined as follows [54]:

$$\frac{\partial f(x, t; x_0)}{\partial t} = \frac{\alpha}{2} \frac{\partial^2 f(x, t; x_0)}{\partial x^2} - \beta \frac{\partial f(x, t; x_0)}{\partial x}. \quad (13)$$

Equation (13) is used to evaluate the queue distribution in the G/G/1 queueing system, i.e. the one with general distributions of interarrival and service times. It is also used to analyse G/G/1/L systems, in which the queue is limited to a finite number L positions. As the number of packets is in the range $[0, L]$, the examined diffusion process is limited by the barriers at $x = 0$ and $x = L$. The diffusion equation for this case supplemented by

balance equations for the barriers [54]:

$$\begin{aligned} \frac{\partial f(x, t; x_0)}{\partial t} &= \frac{\alpha}{2} \frac{\partial^2 f(x, t; x_0)}{\partial x^2} - \beta \frac{\partial f(x, t; x_0)}{\partial x} \\ &\quad + \lambda p_0(t) \delta(x-1) + \lambda p_L(t) \delta(x-L+1), \\ \frac{dp_0(t)}{dt} &= \lim_{x \rightarrow 0} \left[\frac{\alpha}{2} \frac{\partial f(x, t; x_0)}{\partial x} - \beta f(x, t; x_0) \right] - \lambda p_0(t), \\ \frac{dp_L(t)}{dt} &= \lim_{x \rightarrow L} \left[\frac{\alpha}{2} \frac{\partial f(x, t; x_0)}{\partial x} - \beta f(x, t; x_0) \right] - \mu p_L(t). \end{aligned} \quad (14)$$

In the above equations, $p_0(t)$ and $p_L(t)$ denote probabilities that the process is in one of the barriers, the intensities of jumps from barriers (from $x = 0$ to $x = 1$ – an arrival of a packet to the empty queue – and from $x = L$ to $x = L - 1$ – a departure of a packet from the full queue) are denoted as λ and μ respectively. Parameters β and α are chosen as $\beta = \lambda - \mu$, $\alpha = \lambda^3 \sigma_A^2 + \mu^3 \sigma_B^2$, where $1/\lambda$ and $1/\mu$ are the first moments of interarrival and service time distributions, and σ_A^2 , σ_B^2 are their variances. This way the diffusion process and the queueing have the same mean and variance. The steady-state solution of equation (14) can be found in work [54] and the transient case is considered in [23]. The diffusion approximation model considering multiple independent data streams, including independent TCP and UDP streams, is described in [46].

The TCP/UDP/AQM diffusion model below allows us to obtain the mean value of queue length. This value is used to calculate the probability of packet loss and the new value of the congestion window. It is assumed that the flow λ , the delay q/μ of the router, and the congestion window size W are linked in the following way:

$$\lambda = \frac{W\mu}{q}, \quad (15)$$

where $1/\mu$ is the mean service time.

A queue distribution is obtained at time t_i (at the end of the i -th interval). We use it to calculate the average queue length $E[q_i]$ and the packet rejection probability p_i which is then used to define new value of λ : $\lambda_{i+1} = \lambda_i + \Delta\lambda_i$ due to the AQM algorithm, where

$$\Delta\lambda_i = \frac{\mu_i}{E[q_i]} - \frac{\lambda_i^2}{2} \frac{E[q_i]}{\mu} p_i. \quad (16)$$

The calculations are repeated over time ($t_{i+1} = t_i + 1/\lambda_i$) for new values of λ . We assume that the i -th flow can start or end a transmission at any moment. The change in the source intensity $\Delta\lambda$ at the time t affects the dispatching time and the queue length.

In a network model, the sum of arrivals at station j during time Δt has normal distribution with the average

$$\lambda_j \Delta t = \left[\sum_{i=1}^M \lambda_i r_{ij} + \lambda_{0j} \right] \Delta t, \quad (17)$$

where r_{ij} is the routing probability between station i and j , and M is the number of interconnected stations,

and variance

$$\sigma_{A_j}^2 \Delta t = \left[\sum_{i=1}^M C_{Di}^2 \lambda_i r_{ij} + C_{0j}^2 \lambda_{0j} \right] \Delta t. \quad (18)$$

where C_{Di}^2 is the squared coefficient of variation of interdeparture times at station i .

These equations, written for every station i , make a system that allows us to determine λ_i , and the squared coefficient of variation of the interarrival, arrival times at any station $j = 1, \dots, M$

$$C_{Aj}^2 = \frac{1}{\lambda_j} \sum_{i=1}^M r_{ij} \lambda_i [(C_{Di}^2 - 1)r_{ij} + 1] + \frac{C_{0j}^2 \lambda_{0j}}{\lambda_j}, \quad (19)$$

hence the parameters for the diffusion model of any station. Parameters λ_{0j} , C_{0j}^2 refer to the streams originating at the station j .

5. NUMERICAL RESULTS

This section is divided into two parts. The first one (Section 5.1) compares results of Fluid Flow approximation and the proposed Diffusion Approximation model in the case of a simple network. In the second part (Section 5.2), a more complex network is considered.

5.1. Comparison of the performance of the proposed model with the Fluid Flow approximation in a multi-device setup

The comparison of the developed Diffusion model with the Fluid Flow approximation was done for a network model consisting of 1, 2 or 3 routers and one transmitting and receiving station. We used three different AQM controllers: NLRED, RED and PI^γ . The following NLRED parameters were taken: $a_1 = 0.00042$, $a_2 = -0.0000038$, $p_{\max} = 0.01$, $w = 0.08$. The RED parameters were: $p_{\max} = 0.01$, $w = 0.08$. We considered $\text{Min}_{th} = 10$, $\text{Max}_{th} = 20$ and $= 30$ packets RED and NLRED AQM buffers. The impact of RED parameters on the network traffic was described in [56]. Higher values of these parameters result in an increase in the network traffic fluctuations and their proper choice makes the control more transparent in the case of TCP/UDP flows, (see [22]). The values of parameters obtained in such a manner slightly differ from the ones proposed in the literature [57]. In the case of the NLRED algorithm, we refer to the values proposed in [11] as the ones offering the best transmission performance. We also used the following values for PI^γ controller: setpoint = 10, $P = 0.0001$, $I = 0.0004$, buffer size (measured in packets) = 30, and $\gamma = -0.1$. Generally, it is difficult to choose the AQM/PI controller parameters. They strongly affect the packet dropping function, as integral order γ strengthens and accelerates the response of the controller.

Figures 1–31 present the queue behaviour for various AQM mechanisms, types and numbers of network flows present the queue (i.e. single TCP stream, single TCP and single UDP streams or one TCP and two UDP streams). Changes in the

queue are determined by two mathematical models: Diffusion Approximation and Fluid Flow Approximation.

Figures 1–3 display the influence of NewReno TCP algorithm on one queue length in a network consisting of the single Router R1. The queue is controlled by three types of AQM mechanisms: NLRED (Fig. 1) RED (Fig. 2), and PI^γ (Fig. 3). As can be seen, the results obtained with both methods are similar. Average queue lengths are almost the same. However, the Diffusion model better reflects the transient queue behaviour. In steady-state, both models stabilise queue occupancy at the same level. Obtained average queue lengths depend on the AQM mechanism and take the following values: NLRED 13.0, RED 16.8 and PI^γ 18.0.

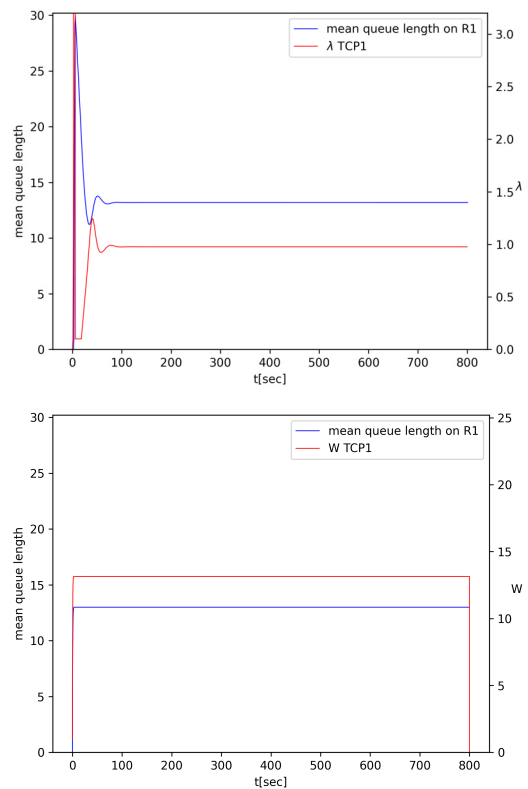


Fig. 1. The router mean queue length (blue) and throughput (red), 1 TCP stream and 1 router between sender and receiver for NLRED controller, Diffusion Approximation (top), Fluid Flow approximation (bottom)

In the next stage of the experiments, we increase the number of considered routers. Figures 4–6 present results for a single TCP stream and network consisting of three routers: R1, R2 and R3. Similarly to the previous experiment, the figures present queues controlled by the following AQMs: NLRED (Fig. 4), RED (Fig. 5) and PI^γ (Fig. 6). The results show the advantage of diffusion in the case of transients and the consistency of results obtained by both methods. The following average queue lengths have been obtained: 10.2 for NLRED, 10.6 for RED and 10.8 for PI^γ . These values actually came close to the Min_{th} (RED and NLRED) and setpoint (PI^γ) pa-

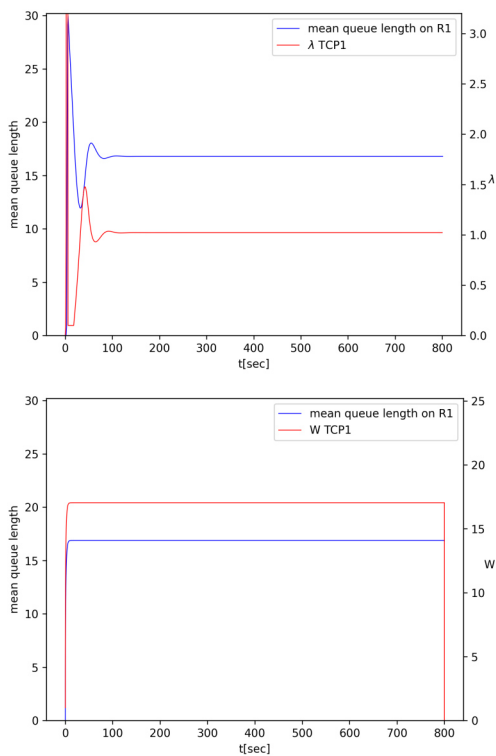


Fig. 2. The router mean queue length (blue) and throughput (red), 1 TCP stream and 1 router between sender and receiver for RED controller, Diffusion Approximation (top), Fluid Flow approximation (bottom)

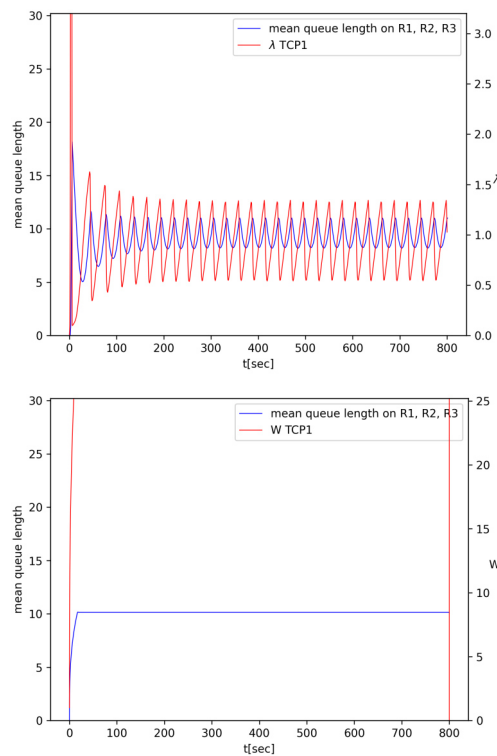


Fig. 4. The router mean queue length (blue) and throughput (red), 1 TCP stream and 3 routers between sender and receiver for NLRED controller, Diffusion Approximation (top), Fluid Flow approximation (bottom)

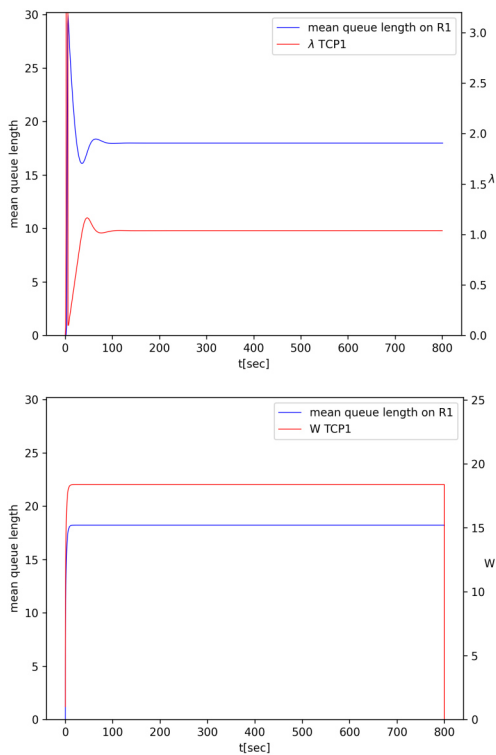


Fig. 3. The router mean queue length (blue) and throughput (red), 1 TCP stream and 1 router between sender and receiver for PI^γ controller, Diffusion Approximation (top), Fluid Flow approximation (bottom)

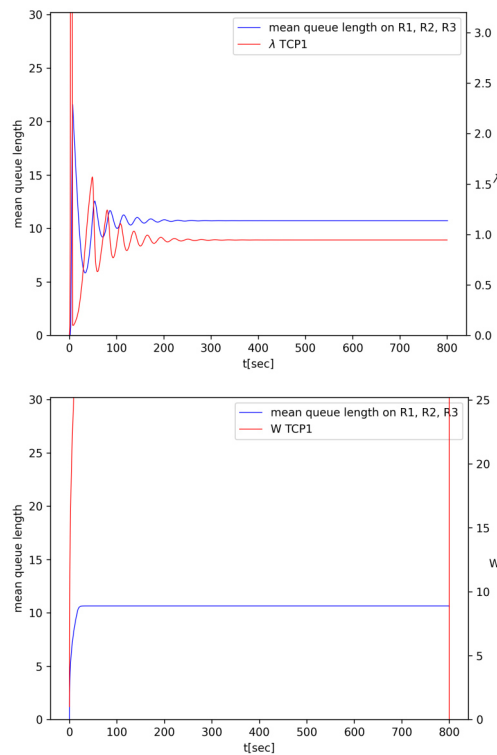


Fig. 5. The router mean queue length (blue) and throughput (red), 1 TCP stream and 3 routers between sender and receiver for RED controller, Diffusion Approximation (top), Fluid Flow approximation (bottom)

Approximation models for the evaluation of TCP/AQM networks

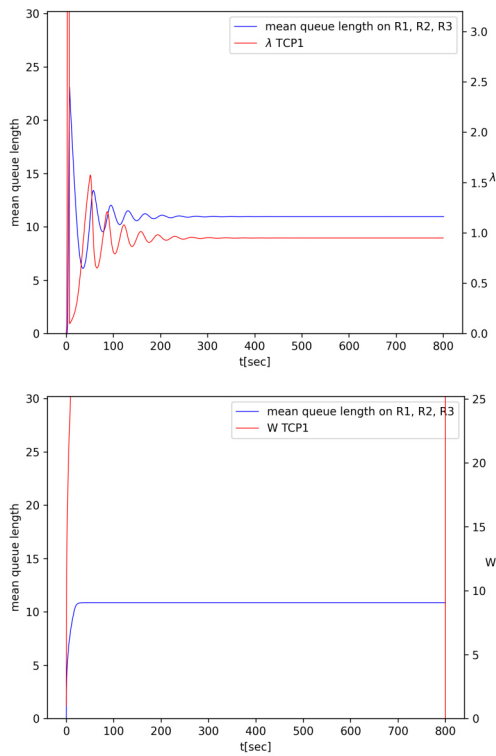


Fig. 6. The router mean queue length (blue) and throughput (red), 1 TCP stream and 3 routers between sender and receiver for PI^γ controller, Diffusion Approximation (top), Fluid Flow approximation (bottom)

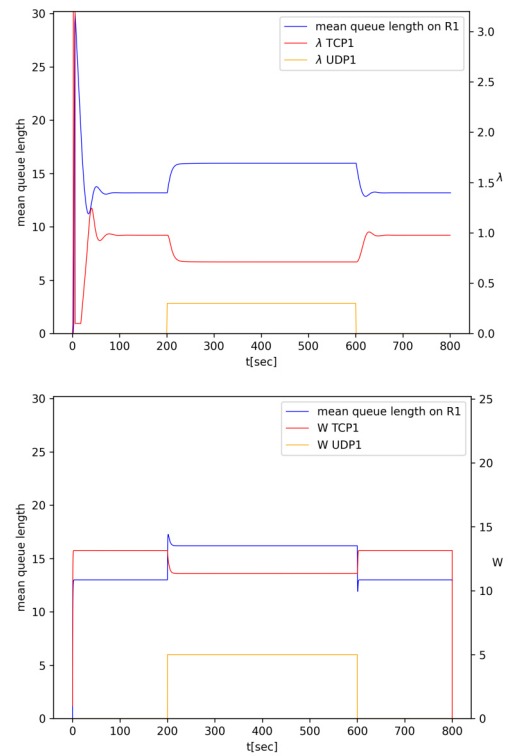


Fig. 7. The router mean queue length (blue) and throughput (red, yellow), 1 TCP and 1 UDP stream, 1 router between sender and receiver for NLRED controller, Diffusion Approximation (top), Fluid Flow approximation (bottom)

rameters. The decrease in queue occupancy is due to the reduction of the network load (a more significant number of nodes handles incoming traffic). An additional increase in the number of routers causes a further reduction in the length of the queues.

Figures 7–9 show the behaviour of the network with one router R1 in the case of one TCP and one UDP stream. The examined TCP stream starts at time $t = 0$. The UDP stream starts at time $t = 200$ and finishes at $t = 600$. We have used a constant intensity of UDP stream. Its intensity is set to the level $\lambda_{UDP1} = 0.3$ for the Diffusion Approximation model and $W_{UDP1} = 5.0$ for the Fluid Flow model.

The results show that between $t = 0$ and $t = 200$, the queue evolves similarly to the case of one TCP stream. The UDP stream (see $t = 200$) decreases the TCP intensity and increases the queue occupancy. The queue length grows and stabilizes at the level 16.0 for NLRED (see Fig. 7), 19.3 for RED (Fig. 8) and 20.7 for PI^γ (Fig. 9). The experiment was then repeated with the number of routers increased to 3 (Figs. 10–12). The same as in the previous experiment, for 3 routers and one TCP source, the queue occupancy fluctuates with the minimum value of Min_{th} for RED and NLRED and $setpoint = 10$ for PI^γ . For all AQMs, the Diffusion Approximation better reflects the queue transients. This can be seen in the case of the NLRED queue. The captured oscillations are imperceptible when using the fluid flow method.

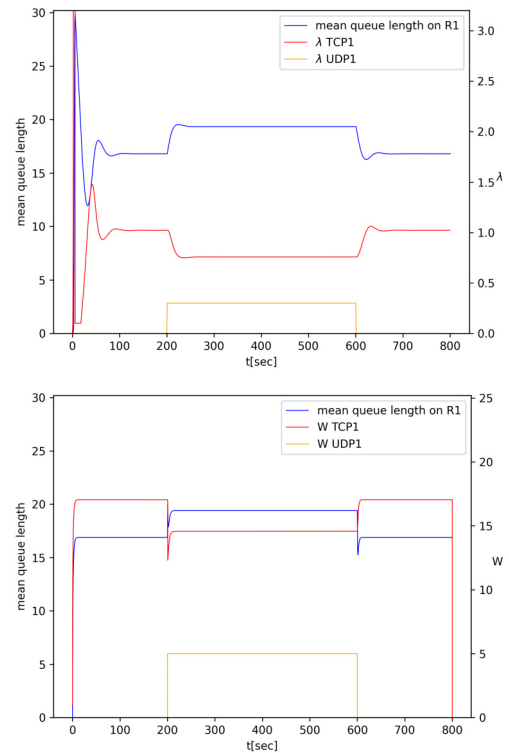


Fig. 8. The router mean queue length (blue) and throughput (red, yellow), 1 TCP and 1 UDP stream, 1 router between sender and receiver for RED controller, Diffusion Approximation (top), Fluid Flow approximation (bottom)

D. Marek, A. Domański, J. Domańska, J. Szyguła, T. Czachórski, J. Klamka, and K. Filus

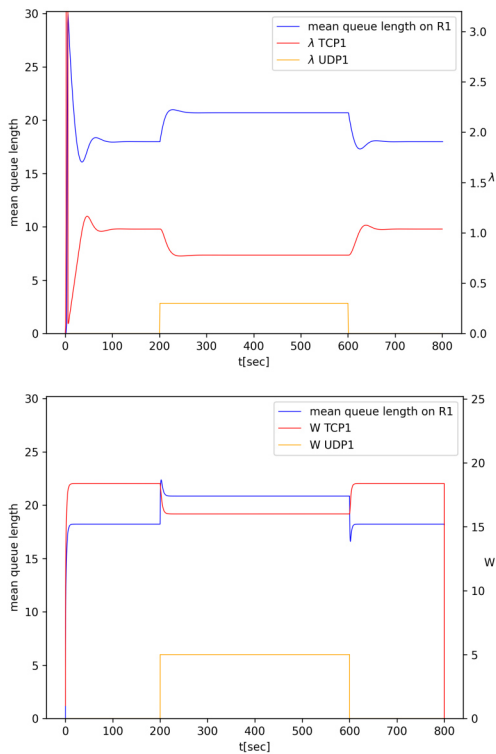


Fig. 9. The router mean queue length (blue) and throughput (red, yellow), 1 TCP and 1 UDP stream, 1 router between sender and receiver for PI^γ controller, Diffusion Approximation (top), Fluid Flow approximation (bottom)

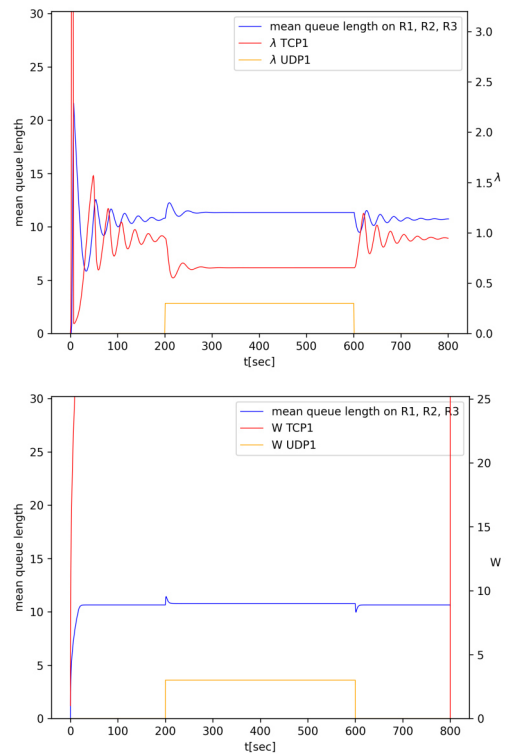


Fig. 11. The router mean queue length (blue) and throughput (red, yellow), 1 TCP and 1 UDP stream, 3 routers between sender and receiver for RED controller, Diffusion Approximation (top), Fluid Flow approximation (bottom)

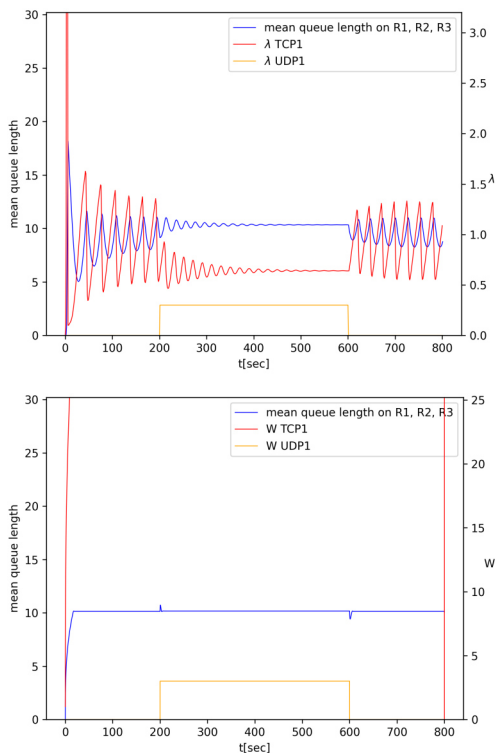


Fig. 10. The router mean queue length (blue) and throughput (red, yellow), 1 TCP and 1 UDP stream, 3 routers between sender and receiver for NLRED controller, Diffusion Approximation (top), Fluid Flow approximation (bottom)

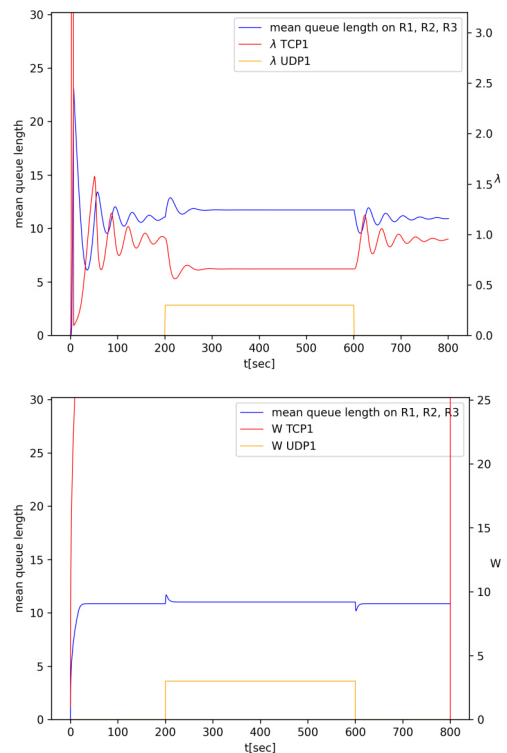


Fig. 12. The router mean queue length (blue) and throughput (red, yellow), 1 TCP and 1 UDP stream, 3 routers between sender and receiver for PI^γ controller, Diffusion Approximation (top), Fluid Flow approximation (bottom)

Approximation models for the evaluation of TCP/AQM networks

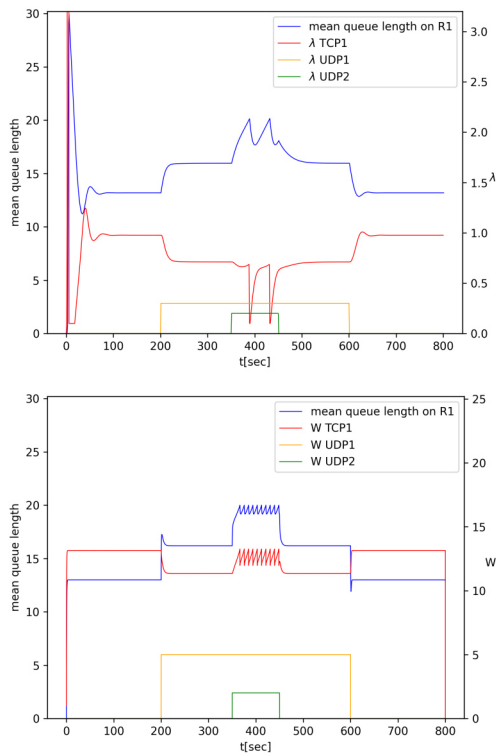


Fig. 13. The router mean queue length (blue) and throughput (red, yellow, green), 1 TCP and 2 UDP streams, 1 router between sender and receiver for NLRED controller, Diffusion Approximation (top), Fluid Flow approximation (bottom)

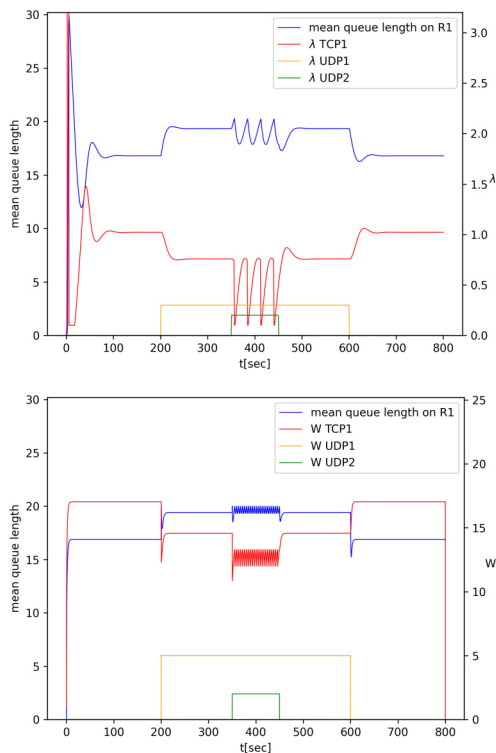


Fig. 14. The router mean queue length (blue) and throughput (red, yellow, green), 1 TCP and 2 UDP streams, 1 router between sender and receiver for RED controller, Diffusion Approximation (top), Fluid Flow approximation (bottom)

The influence of one TCP stream and two UDP streams on the queue occupancy for one router is presented in Figs. 13–15. They have used the following sequence of the experiments: at time $t = 0$ TCP sender starts transmission, first UDP stream starts transmission at time $t = 200$, and the second one - at time $t = 350$, and finishes at $t = 450$; at $t = 600$ first UDP stream ends data sending. Both UDP streams reflect CBR (Constant Bit Rate) flows. The intensity of the streams was set to $\lambda_{UDP1} = 0.3$ for Diffusion Approximation and $W_{UDP1} = 5.0$ for Fluid Flow, $\lambda_{UDP2} = 0.2$ for Diffusion Approximation and $W_{UDP2} = 3.5$ for Fluid Flow. Each start or stop of the UDP stream affects the queue behaviour and the evolution of the congestion window of the TCP sender. The Diffusion model better captures these changes.

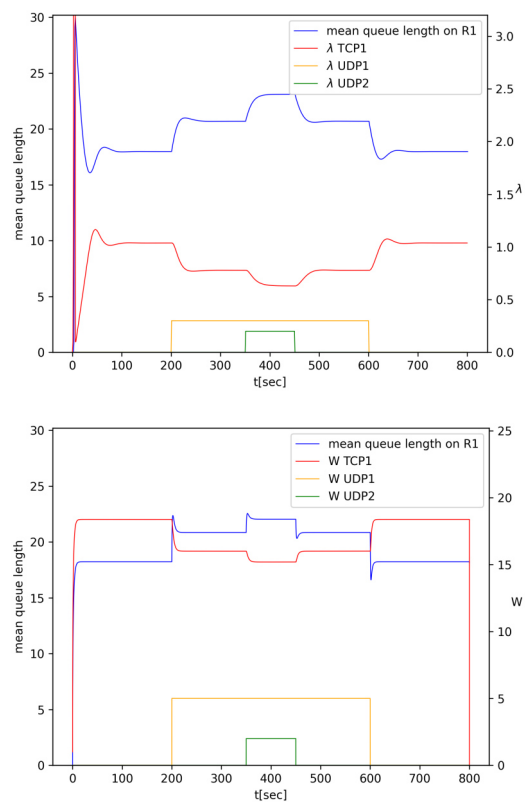


Fig. 15. The router mean queue length (blue) and throughput (red, yellow, green), 1 TCP and 2 UDP streams, 1 router between sender and receiver for PI^γ controller, Diffusion Approximation (top), Fluid Flow approximation (bottom)

In the case of 3 routers (Figs. 16–18) (similar to previous experiments) the queue length oscillates around the assumed thresholds.

Figures 19–21 present the aggregate results of the average queue size for all experiments conducted with 1, 2, 3, and 4 routers. As it could be observed in previous experiments, within a single experiment, the queueing behaviour is identical for all routers. The figures also show that each additional router decreases the average occupancy of queues.

D. Marek, A. Domański, J. Domańska, J. Szyguła, T. Czachórski, J. Klamka, and K. Filus

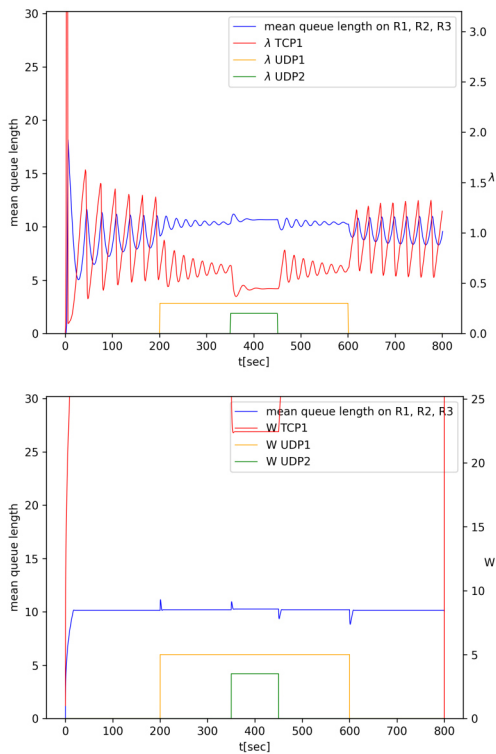


Fig. 16. The router mean queue length (blue) and throughput (red, yellow, green), 1 TCP and 2 UDP streams, 3 routers between sender and receiver for NLRED controller, Diffusion Approximation (top), Fluid Flow approximation (bottom)

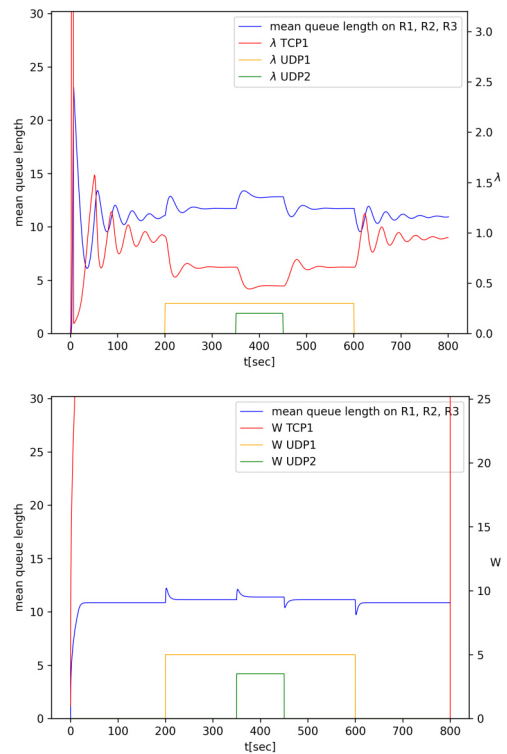


Fig. 18. The router mean queue length (blue) and throughput (red, yellow, green), 1 TCP and 2 UDP streams, 3 routers between sender and receiver for PI' controller, Diffusion Approximation (top), Fluid Flow approximation (bottom)

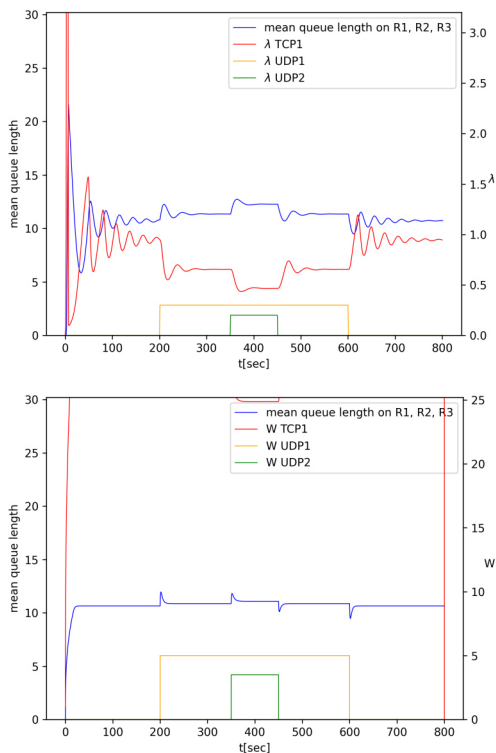


Fig. 17. The router mean queue length, 1 TCP and 2 UDP streams (blue) and throughput (red, yellow, green), 3 routers between sender and receiver for RED controller, Diffusion Approximation (top), Fluid Flow approximation (bottom)

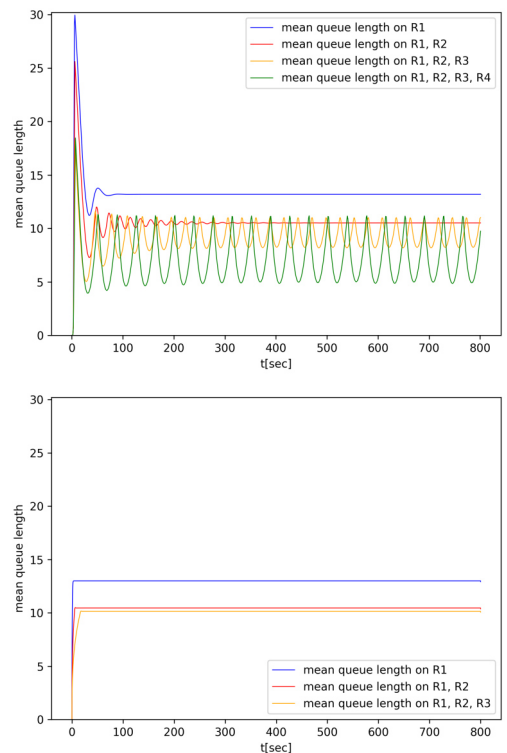


Fig. 19. The router mean queue length, 1 TCP stream for NLRED controller, Diffusion Approximation (top), Fluid Flow approximation (bottom)

Approximation models for the evaluation of TCP/AQM networks

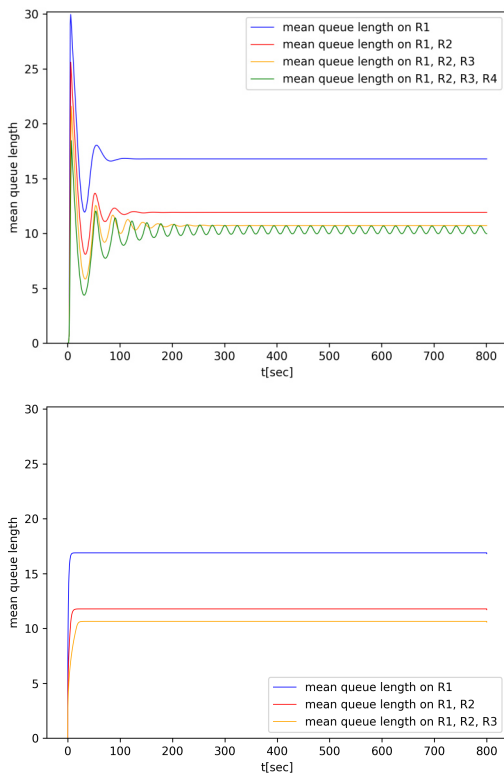


Fig. 20. The router mean queue length, 1 TCP stream for RED controller, Diffusion Approximation (top), Fluid Flow approximation (bottom)

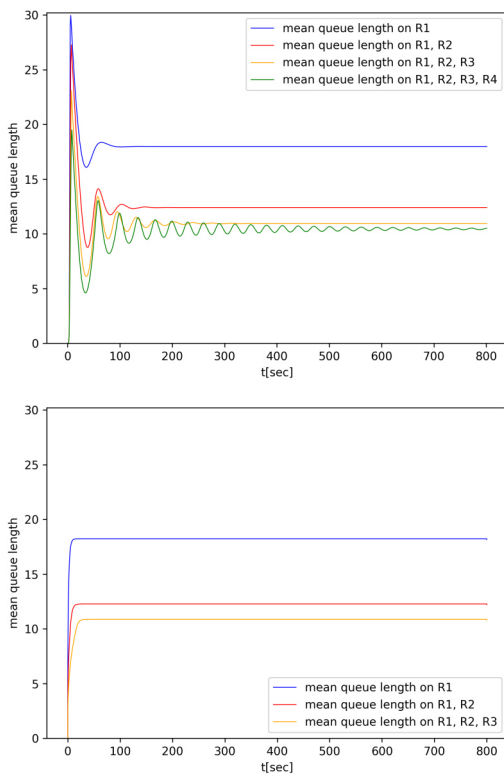


Fig. 21. The router mean queue length, 1 TCP stream for PI controller, Diffusion Approximation (top), Fluid Flow approximation (bottom)

5.2. Diffusion Approximation for complex network topology

The study of the Diffusion model was conducted based on a network model consisting of 6 routers, three transmitting stations and one receiving station, Fig. 22.

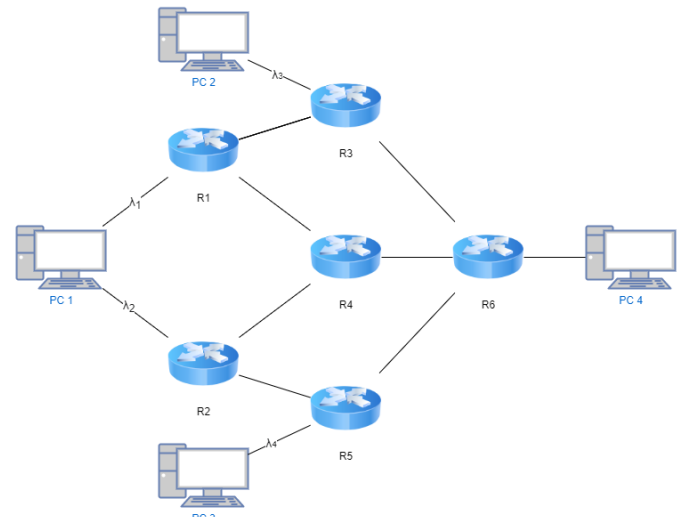


Fig. 22. Diagram of a complex network model

Router Details: R1: $\mu = 1$, $C_B^2 = 1$, R2: $\mu = 1$, $C_B^2 = 1$, R3: $\mu = 1.2$, $C_B^2 = 1$, R4: $\mu = 1.5$, $C_B^2 = 1$, R5: $\mu = 1.2$, $C_B^2 = 1$, R6: $\mu = 3$, $C_B^2 = 1$.

The network structure is no longer linear. In addition, the Diffusion Approximation model allows us (in the case of multiple outputs from a single node) to set the percentage of traffic that will be redirected to a specified node.

Verification of the correctness of the obtained results using simulation:

To verify the correctness of the diffusion approximation results, tests were carried out using simulation methods. A simulation model was created using SimPy and Python [58]. To show the realistic behavior of queues, the simulation was repeated 100,000 times. In Figs. 26–28, we present averaged results.

Figures 23–25 present the results obtained for one TCP stream and two UDP streams.

Description of the experiment:

- The UDP3 stream enters router R5 (λ_4) and begins at $t = 200$ and ends at $t = 500$ with a fixed intensity of 0.2.
- The UDP4 stream enters router R3 (λ_3) and begins at $t = 400$ and ends at $t = 650$ with a constant intensity of 0.3.
- The TCP stream propagates to routers R1 (λ_1) and R2 (λ_2).

It can be seen that the incorporation of UDP streams hardly affects the behaviour of TCP streams. Therefore, the differences between different AQM mechanisms are barely noticeable. The obtained queue sizes were below the set thresholds. When a UDP3 stream (with constant intensity $\lambda_{UDP3} = 0.3$) starts transmission at $t = 200$ and ends at $t = 500$, the effect can be observed in the average occupancy of queue R5. In the case of

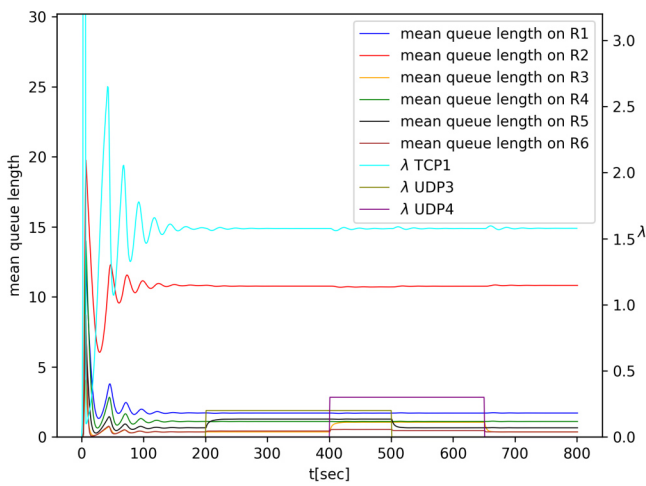


Fig. 23. The router mean queue length, 1 TCP and 2 UDP streams for NLRED controller

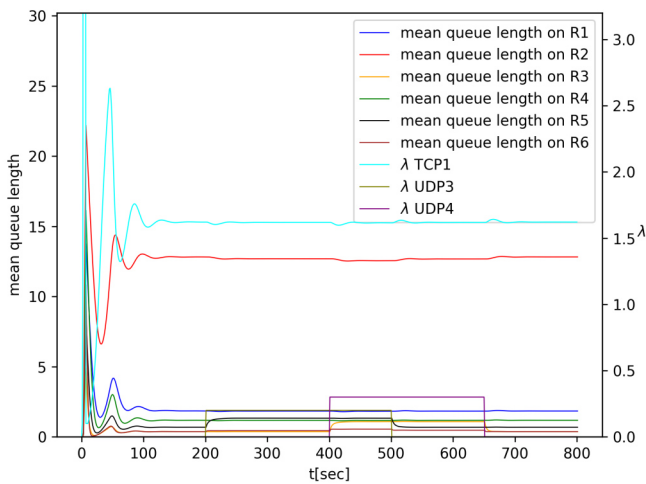


Fig. 24. The router mean queue length, 1 TCP and 2 UDP streams for RED controller

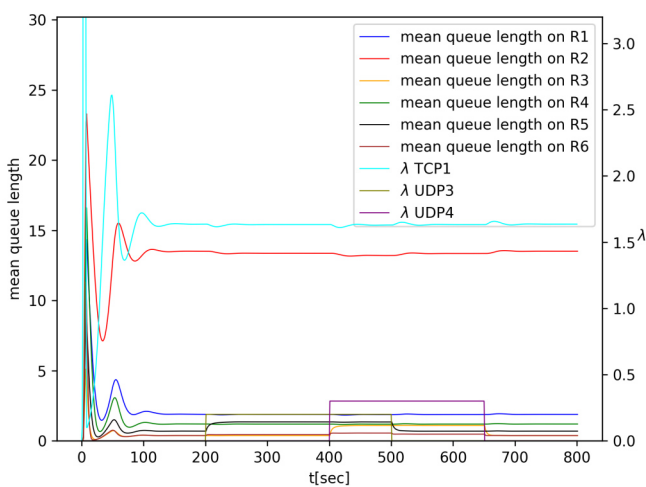


Fig. 25. The router mean queue length, 1 TCP and 2 UDP streams for PI^γ controller

UDP4 stream (with constant intensity $\lambda_{UDP4} = 0.3$), the average occupancy of queue R3 increases. Among all the controllers selected for the study, the highest average queue occupancy was observed for the PI^γ controller (see Fig. 25) and the lowest for the NLRED controller (see Fig. 23).

Figures 26–28 present the results for one TCP stream and three UDP streams.

Description of the experiment:

- The UDP3 begins at $t = 200$ and ends at $t = 500$; it enters router R5 with fixed intensity $\lambda_4 = 0.2$.
- The UDP4 begins at $t = 400$ and ends at $t = 650$; it enters router R3 with fixed intensity $\lambda_3 = 0.3$.
- The UDP1 stream propagates through routers R1 (λ_1) and R2 (λ_2), begins at $t = 200$ and ends at $t = 600$ with a constant intensity of 0.3.
- The TCP stream propagates through routers R1 (λ_1) and R2 (λ_2).

Such an experiment allowed us to show the influence of UDP streams on the TCP stream. However, it is not significant. It is because streams come from different sources and spread in different parts of the network. At time $t = 200$ an additional UDP1

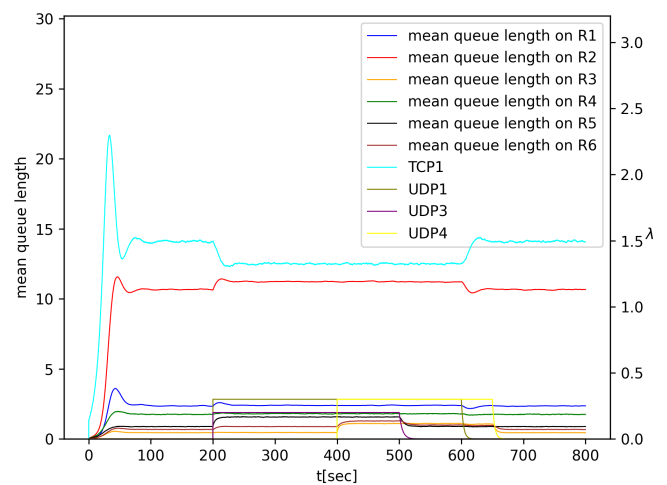
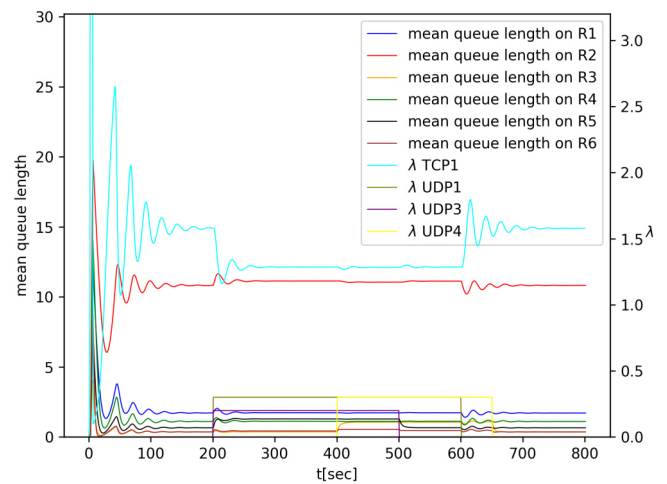


Fig. 26. The router mean queue length, 1 TCP and 3 UDP streams for NLRED controller, Diffusion Approximation (top), Simulation (bottom)

Approximation models for the evaluation of TCP/AQM networks

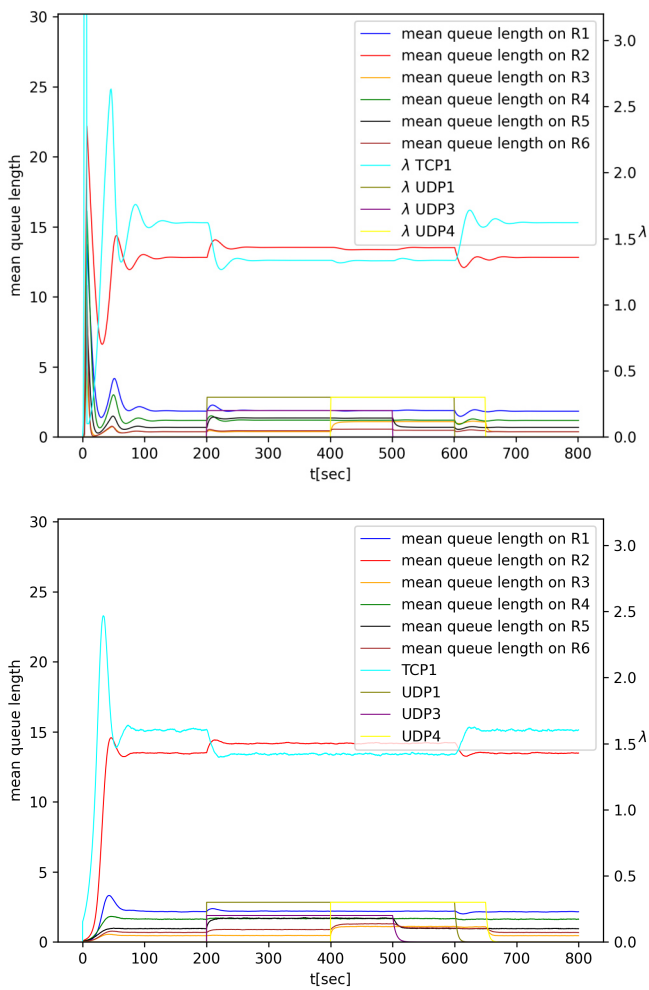


Fig. 27. The router mean queue length, 1 TCP and 3 UDP streams for RED controller, Diffusion Approximation (top), Simulation (bottom)

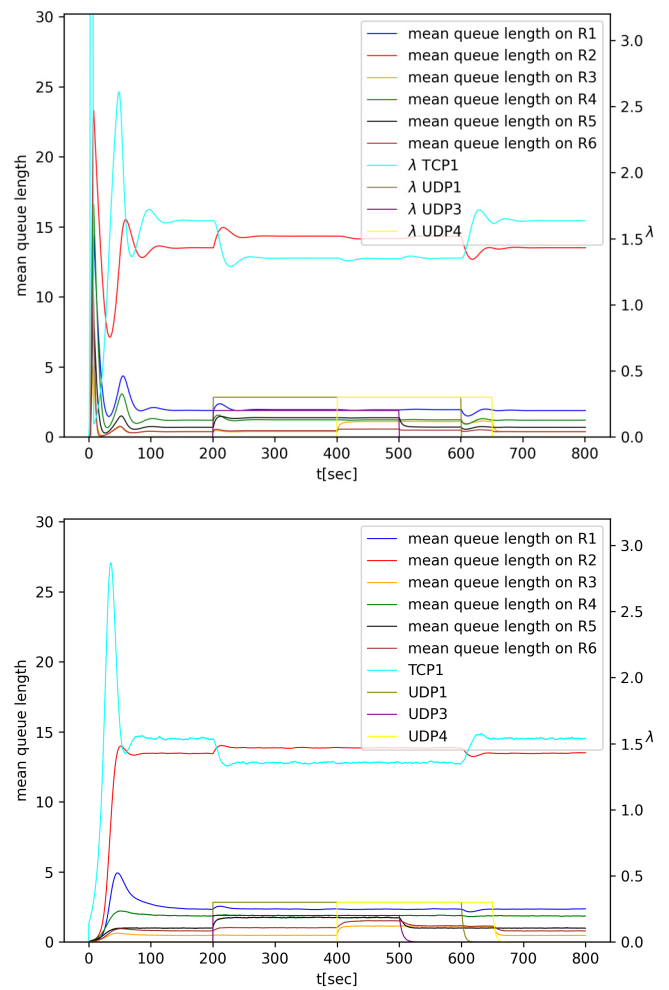


Fig. 28. The router mean queue length, 1 TCP and 3 UDP streams for PI^γ controller, Diffusion Approximation (top), Simulation (bottom)

stream (of constant intensity $\lambda_{UDP1} = 0.3$) begins the transmission. Until the end of the transmission at time $t = 600$, the average occupancy of queue R2 increases. The largest oscillation in queue occupancy during the start and stop of the additional stream transmission was observed for the NLRED controller (see Fig. 26). The most stable queue behavior and the highest average buffer occupancy values were observed for the PI^γ controller (see Fig. 28).

By comparing the values obtained with diffusion approximation (top part of Figs. 26–28) and simulation (see bottom part of Figs. 26–28), it should be noted that in the second case slightly higher average queue occupancy values are obtained. However, when the characteristics of the graphs are compared, the queuing behavior is similar.

Figures 29–31 present the obtained results for two TCP streams.

Description of the experiment:

- The TCP2 stream enters router R5 (λ_4); it begins at $t = 100$ and ends at $t = 700$.
- The TCP1 stream propagates to routers R1 (λ_1) and R2 (λ_2). Both streams seek to maximise the link utilisation. It affects

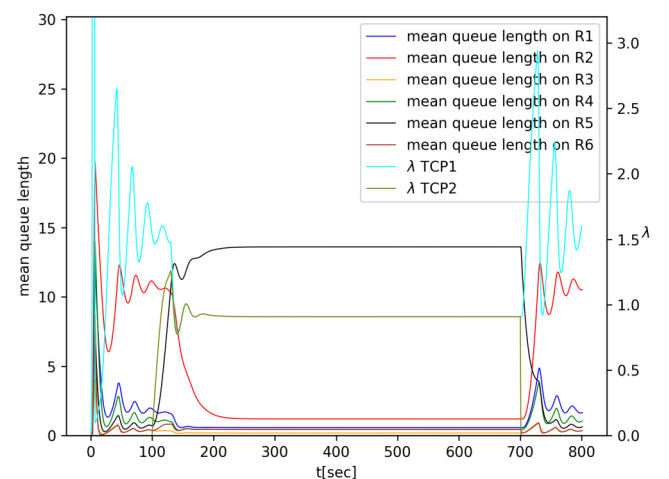


Fig. 29. The router mean queue length, 2 TCP streams for NLRED controller

mostly the nodes connected to the routing stations. The occupancy of the nodes which are far from the traffic sources is low.

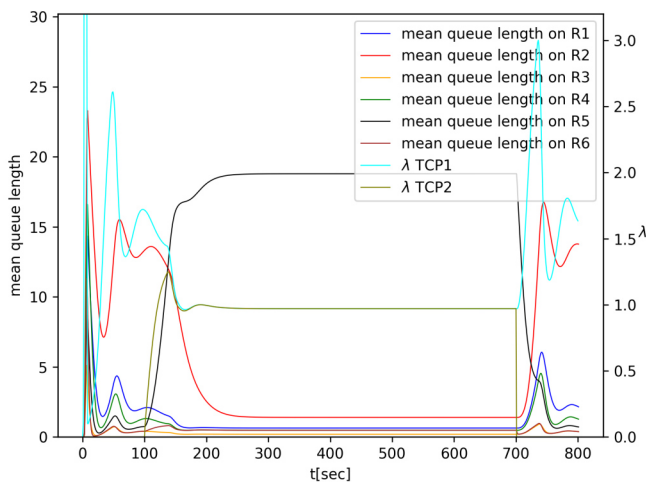


Fig. 30. The router mean queue length, 2 TCP streams for RED controller

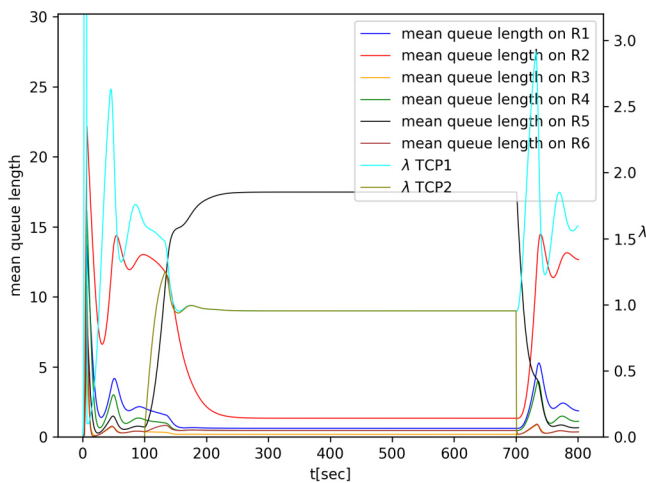


Fig. 31. The router mean queue length, 2 TCP streams for PI^γ controller

6. CONCLUSIONS

A good analysis of a computer network transient states, resulting from the changes of transmitted flows and the action of congestion control mechanisms, is essential in performance evaluation. In the article, we present a computer network model based on Diffusion Approximation, and we use it to evaluate the performance of TCP congestion window mechanisms cooperating with IP AQM mechanisms. The model allows multiple, existing simultaneously, TCP and UDP streams. We considered different types of AQM mechanisms (RED, NLRED, PI^γ), determining packet loss probability in IP routers.

The proposed model was verified using the Fluid Flow approximation, which is traditionally used in such problems. Fluid Flow approximation operates on time-dependent mean values, and it is valid for Poisson traffic and exponential service times. Our experiments were limited to these assumptions and displayed that the results given by both methods are consistent,

which is a kind of validation of the diffusion model. However, the diffusion model allows us to assume any distribution of packets' interarrival times and service times; therefore, it is more general. Furthermore, diffusion models, based on means and variations of traffic fluctuations, allow us to capture queues dynamics (transient states) with more precision. It is also adapted to include routing probabilities, hence allowing us a better description of traffic distribution along the network and is closer to the actual operation of a computer network.

ACKNOWLEDGEMENTS

This research was financed by National Science Center project no. 2017/27/B/ST6/00145.

REFERENCES

- [1] J.M. Amigó, G. Duran, Á. Giménez, J. Valero, and O.M. Bonastre, "Modeling a new aqm model for internet chaotic behavior using petri nets," *Appl. Sci.*, vol. 11, no. 13, p. 5877, 2021.
- [2] N. El Fezazi, Y. Elfakir, F.A. Bender, and S. Idrissi, "Aqm congestion controller for tcp/ip networks: Multiclass traffic," *J. Control Autom. Electr. Syst.*, vol. 31, no. 4, pp. 948–958, 2020.
- [3] L. Tan, K. Huang, G. Peng, and G. Chen, "Stability of tcp/aqm networks under ddos attacks with design," *IEEE Trans. Network Sci. Eng.*, vol. 7, no. 4, pp. 3042–3056, 2020.
- [4] Wu-chang Feng, K.G. Shin, D.D. Kandlur, and D. Saha, "The BLUE Active Queue Management algorithms," *IEEE/ACM Trans. Networking*, vol. 10, no. 4, pp. 513–528, 2002.
- [5] Liujia Hu and A.D. Kshemkalyani, "HRED: a simple and efficient Active Queue Management algorithm," in *Proceedings. 13th International Conference on Computer Communications and Networks*, 2004, pp. 387–393.
- [6] C. Long, B. Zhao, X. Guan, and J. Yang, "The Yellow Active Queue Management algorithm," *Computer Networks*, vol. 47, no. 4, pp. 525–550, 2005.
- [7] N. Khademi, D. Ros, and M. Welzl, "The new aqm kids on the block: An experimental evaluation of codel and pie," in *2014 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, 2014, pp. 85–90.
- [8] A. Adamu, V. Shorgin, S. Melnikov, and Y. Gaidamaka, "Flexible random early detection algorithm for queue management in routers," in *International Conference on Distributed Computer and Communication Networks*, 2020, pp. 196–208.
- [9] S.K. Bisoy and P.K. Pattnaik, "A neuron-based active queue management scheme for internet congestion control," *Int. J. Reasoning-based Intell. Syst.*, vol. 12, no. 4, pp. 238–247, 2020.
- [10] S. Floyd and V. Jacobson, "Random Early Detection gateways for congestion avoidance," *IEEE/ACM Trans. Networking*, vol. 1, no. 4, pp. 397–413, 1993, doi: [10.1109/90.251892](https://doi.org/10.1109/90.251892).
- [11] J. Domańska, D. Augustyn, and A. Domański, "The choice of optimal 3-rd order polynomial packet dropping function for NLRED in the presence of self-similar traffic," *Bull. Pol. Acad. Sci. Tech. Sci.*, vol. 60, no. 4, pp. 779–786, 2012, doi: [10.2478/v10175-012-0090-x](https://doi.org/10.2478/v10175-012-0090-x).
- [12] K. Zhou, K. Yeung, and V. Li, "Nonlinear RED: A simple yet efficient Active Queue Management scheme," *Comput Networks, Int. J. Comput. Telecommun. Networking*, vol. 50, no. 18, pp. 3784–3794, Dec. 2006, doi: [10.1016/j.comnet.2006.04.007](https://doi.org/10.1016/j.comnet.2006.04.007).

- [13] I. Podlubny, "Fractional order systems and $PI^\lambda D^\mu$ controllers," *IEEE Trans. Autom. Control*, vol. 44, no. 1, pp. 208–214, 1999.
- [14] M. Tarmizi, A. Albagul, O. Khalifa, and Wahyudi, "QoS Evaluation of Different TCPs Congestion Control Algorithm using NS2," in *2nd International Conference on Information Communication Technologies*, vol. 2, 2006, pp. 3222–3227.
- [15] S. Floyd, T. Henderson, and A. Gurtov, "The NewReno Modification to TCP's Fast Recovery Algorithm," *RFC 3782*, 2004.
- [16] C.A. Grazia, N. Patriciello, M. Klapez, and M. Casoni, "A cross-comparison between TCP and AQM algorithms: Which is the best couple for congestion control?" *14th International Conference on Telecommunications (ConTEL)*, pp. 75–82, 2017.
- [17] C.V. Hollot, V. Misra, and D. Towsley, "Analysis and design of controllers for AQM routers supporting TCP flows," *IEEE Transaction on Automatic Control*, vol. 47, no. 6, 2002.
- [18] P. Shah, S. Yasmin, S. Asghar, A. Qayyum, and H. Hasbullah, "A Fluid Flow Model for SCTP Traffic over the Internet," in *Proceedings of the International Conference on Emerging Technologies (ICET)*, 2012, pp. 1–6, doi: [10.1109/ICET.2012.6375481](https://doi.org/10.1109/ICET.2012.6375481).
- [19] J. Domańska, A. Domański, T. Czachórski, and J. Klamka, "Fluid flow approximation of time-limited TCP/UDP/XCP streams," *Bull. Pol. Acad. Sci. Tech. Sci.*, vol. 62, no. 2, pp. 217–225, 2014, doi: [10.2478/bpasts-2014-0021](https://doi.org/10.2478/bpasts-2014-0021).
- [20] M. Bonaventura and R. Castro, "Fluid-flow and packet-level models of data networks unified under a modular/hierarchical framework: speedups and simplicity, combined," in *2018 Winter Simulation Conference (WSC)*, 2018, pp. 3825–3836.
- [21] O.J. Kravets, I.V. Atlasov, I.A. Aksenov, A.S. Molchan, O.Y. Frantsisko, and P.A. Rahman, "Increasing efficiency of routing in transient modes of computer network operation," *Int. J. Intell. Technol. Secur.*, vol. 13, no. 2, pp. 3–14, 2021.
- [22] A. Domański, J. Domańska, T. Czachórski, J. Klamka, J. Szygula, and D. Marek, "Diffusion Approximation Model of TCP NewReno Congestion Control Mechanism," *SN Comput. Sci.*, vol. 1, p. 43, 2020, doi: [10.1007/s42979-019-0032-x](https://doi.org/10.1007/s42979-019-0032-x).
- [23] T. Czachórski, "A method to solve Diffusion Equation with Instantaneous return Processes Acting as Boundary Conditions," *Bull. Pol. Acad. Sci. Tech. Sci.*, vol. 41, no. 4, pp. 417–451, 1993.
- [24] E. Gelenbe and G. Pujolle, "The Behaviour of a Single-Queue in a General Queueing Network," *Acta Informatica*, vol. 7, no. 2, pp. 123–136, 1976, doi: [10.1007/BF00265766](https://doi.org/10.1007/BF00265766).
- [25] A. Duda, "Diffusion approximations for time-dependent queueing systems," *IEEE J. Sel. Areas Commun.*, vol. 4, pp. 905–918, 1986.
- [26] H. Seferoglu and E. Modiano, "TCP-aware backpressure routing and scheduling," in *Information Theory and Applications Workshop (ITA)*, 2014, pp. 1–9.
- [27] A. Bohloulzadeh and R. Mehri, "A Survey on Congestion Control Protocols in Wireless Sensor Networks," *Int. J. Wireless Inf. Networks*, vol. 27, pp. 365–384, 2020.
- [28] M.A. Jan, S. Jan, M. Alam, A. Akhuzada, and I. Rahman, "A Comprehensive Analysis of Congestion Control Protocols in Wireless Sensor Networks," *Mobile Networks Appl.*, vol. 23, pp. 456–468, 2018.
- [29] V. Misra, W. Gong, and D. Towsley, "Fluid-based analysis of network of AQM routers supporting TCP flows with an application to RED," *Computer Communication Review*, vol. 30, no. 4, pp. 151–160, 2000, doi: [10.1145/347059.347421](https://doi.org/10.1145/347059.347421).
- [30] W. Li, L. Zeng-zhi, C. Yan-ping, and X. Ke, "Fluid-based stability analysis of mixed TCP and UDP traffic under RED," in *10th IEEE International Conference on Engineering of Complex Computer Systems (ICECCS'05)*, 2005, pp. 341–348.
- [31] V. Misra, W.-B. Gong, and D. Towsley, "Fluid-Based Analysis of a Network of AQM Routers Supporting TCP Flows with an Application to RED," *Assoc. Comput. Mach.*, vol. 30, no. 4, p. 151–160, 2000, doi: [10.1145/347057.347421](https://doi.org/10.1145/347057.347421).
- [32] M. Nycz, T. Nycz, and T. Czachórski, "Modelling dynamics of TCP flows in very large network topologies," in *Information Sciences and Systems 2015*, 2016, pp. 251–259, doi: [10.1007/978-3-319-22635-4_23](https://doi.org/10.1007/978-3-319-22635-4_23).
- [33] Y. Hong and O.W.W. Yang, "Adaptive AQM controllers for IP routers with a heuristic monitor on TCP flows," *Int. J. Commun. Syst.*, vol. 19, no. 1, pp. 17–38, 2006.
- [34] Jinsheng Sun, K. Ko, Guanrong Chen, S. Chan, and M. Zukerman, "PD-RED: to improve the performance of RED," *IEEE Commun. Lett.*, vol. 7, no. 8, pp. 406–408, 2003.
- [35] F. Yanfie, R. Fengyuan, and L. Chuang, "Design a PID controller for Active Queue Management," in *Proc. of the Eighth IEEE Symposium on Computers and Communications. ISCC 2003*, vol. 2, 2003, pp. 985–990.
- [36] H. Unal, D. Melchor-Aguilar, D. Ustebay, S.-I. Niculescu, and H. Ozbay, "Comparison of PI controllers designed for the delay model of TCP/AQM," *Comput. Commun.*, vol. 36, no. 10, pp. 1225–1234, 2013.
- [37] G. Kahe and A.H. Jahangir, "A self-tuning controller for queuing delay regulation in TCP/AQM networks," *Telecommunication Systems*, vol. 71, pp. 215–229, 2019.
- [38] Y. Chen, I. Petras, and D. Xue, "Fractional order control – A tutorial," in *American Control Conference*, 2009, pp. 1397–1411.
- [39] W. Krajewski and U. Viaro, "On robust fractional order PI controller for TCP packet flow," in *BOS Conference: Systems and Operational Research*, Warsaw, Poland, Sep. 2014.
- [40] J. Domańska, A. Domański, T. Czachórski, and J. Klamka, "The use of a non-integer order PI controller with an Active Queue Management Mechanism," *Int. J. Appl. Math. Comput. Sci.*, vol. 26, pp. 777–789, 2016, doi: [10.1515/amcs-2016-0055](https://doi.org/10.1515/amcs-2016-0055).
- [41] J. Domańska, A. Domański, T. Czachórski, J. Klamka, and J. Szygula, "The AQM Dropping Packet Probability Function Based on Non-integer Order $PI^\alpha D^\beta$ Controller," in *Lecture Notes in Electrical Engineering. Non-Integer Order Calculus and its Applications*, vol. 496, 2019, pp. 36–48, doi: [10.1007/978-3-319-78458-8_4](https://doi.org/10.1007/978-3-319-78458-8_4).
- [42] T. Czachórski and F. Pekergin, "Diffusion Approximation as a Modelling Tool," *Network Performance Engineering. A Handbook on Convergent Multi-Service Networks and Next Generation Internet. LNCS*, vol. 5233, pp. 447–476, 2011, doi: [10.1007/978-3-642-02742-0_20](https://doi.org/10.1007/978-3-642-02742-0_20).
- [43] T. Czachórski, K. Grochla, T. Nycz, and F. Pekergin, "A diffusion approximation model for wireless networks based on IEEE 802.11 standard," *Comput. Commun.*, vol. 33, pp. 86–92, 2010.
- [44] T. Nycz, M. Nycz, and T. Czachórski, "A Numerical Comparison of Diffusion and Fluid-Flow Approximations Used in Modelling Transient States of TCP/IP Networks," *Commun. Comput. Inf. Sci.*, vol. 431, pp. 213–222, 2014.
- [45] D. Marek, A. Domański, J. Domańska, T. Czachórski, J. Klamka, and J. Szygula, "Combined diffusion approximation – simulation model of AQM's transient behavior," *Comput. Commun.*, vol. 166, pp. 40–48, 2020, doi: [10.1016/j.comcom.2020.11.014](https://doi.org/10.1016/j.comcom.2020.11.014).
- [46] D. Marek, A. Domański, J. Domańska, J. Szygula, T. Czachórski, and J. Klamka, "Diffusion Model of a Non-Integer Order PI^γ Controller with TCP/UDP Streams," *Entropy*, vol. 23, no. 5, 2021, doi: [10.3390/e23050619](https://doi.org/10.3390/e23050619).

- [47] J. Domańska, A. Domański, T. Czachórski, and J. Klamka, "Self-similarity Traffic and AQM Mechanism Based on Non-integer Order $PI^\alpha D^\beta$ Controller," in *Computer Networks: Communications in Computer and Information Science*, vol. 718, 2017, pp. 336–350, doi: [10.1007/978-3-319-59767-6_27](https://doi.org/10.1007/978-3-319-59767-6_27).
- [48] A. Domański, J. Domańska, T. Czachórski, J. Klamka, D. Marek, and J. Szyguła, "GPU Accelerated Non-integer Order $PI^\alpha D^\beta$ Controller Used as AQM Mechanism." in *Computer Networks: Communications in Computer and Information Science*, vol. 860, 2018, pp. 286–299, doi: [10.1007/978-3-319-92459-5_23](https://doi.org/10.1007/978-3-319-92459-5_23).
- [49] I. Podlubny, *Fractional Differential Equations*. Academic Press, San Diego, USA, 1999, vol. 198.
- [50] M. Ciesielski and J. Leszczynski, "A Numerical Method for Solution of Ordinary Differential Equations of Fractional Order," in *Parallel Processing and Applied Mathematics*, vol. 2328, 2002, pp. 695–702, doi: [10.1007/3-540-48086-2_77](https://doi.org/10.1007/3-540-48086-2_77).
- [51] C. Hollot, V. Misra, and D. Towsley, "A control theoretic analysis of RED," in *Proc. IEEE INFOCOM 2001*, 2001, pp. 1510–1519.
- [52] D. Towsley, W. Gong, K. Hollot, Y. Liu, and V. Misra, "Fluid Methods for Modeling Large, Heterogeneous Networks," *NTIS*, 2005.
- [53] G.F. Newell, "Queues with time-dependent arrival rates. I – The transition through saturation," *J. Appl. Probab.*, vol. 2, no. 2, pp. 436–451, 1968, doi: [10.2307/3212264](https://doi.org/10.2307/3212264).
- [54] E. Gelenbe, "On Approximate Computer Systems Models," *J. ACM*, vol. 22, no. 2, pp. 261–269, 1975, doi: [10.1145/321879.321888](https://doi.org/10.1145/321879.321888).
- [55] E. Gelenbe, "A Diffusion Model for Packet Travel Time in a Random Multi-hop Medium," *ACM Trans. Sens. Netw.*, vol. 3, no. 2, p. 10, 2007, doi: [10.1145/1240226.1240230](https://doi.org/10.1145/1240226.1240230).
- [56] T. Bonald, M. May, and J. Bolot, "Analytic evaluation of RED performance," in *Proc. of INFOCOM*, 2000.
- [57] B. Zheng and M. Atiquzzaman, "A Framework to Determine the Optimal Weight Parameter of RED in Next-Generation Internet Routers," *Int. J. Commun. Syst.*, vol. 21, no. 9, p. 987–1008, 2008, doi: [10.5555/1405579.1405584](https://doi.org/10.5555/1405579.1405584).
- [58] "Simpfy documentation," [Accessed: 2022-05-27]. [Online]. Available: <https://simpy.readthedocs.io/en/latest/>