Andrzej Borys
Gdynia Maritime University

SOME REMARKS ON PRACTICAL ASPECTS OF THE EFFECTIVE SERVICE CURVE USE IN AD HOC NETWORKS

The so-called $\in$-effective service curve has been thought out for the use in a distributed call admission procedure for wireless ad hoc networks. Its practical usefulness relies upon the fact that it can be constructed on-line exploiting the measured data, and modified accordingly, when the intensity of the cross traffic changes, allowing the call admission to be matched to the system actual traffic load. In this paper, we demonstrate that this curve can be approximated by system parametric service curve for through traffic, depending upon the intensity of system cross traffic, too. We show also that an expression published in the literature that describes the $\in$-effective service must be corrected and its right form is given here. This form allows the correct interpretation of servicing the through traffic in absence of the cross traffic. Moreover, we demonstrate that the use of the so-called greedy pattern of probing packets can be interpreted approximately as applying the Dirac impulse to the system through traffic input.

Keywords: *Network calculus, ad hoc networks, effective and parametric service curves.*

INTRODUCTION

Using an approach called nowadays *Network Calculus* [7], we are able to solve many problems occurring in the area of networking, as, for example, that shown in [4] or [5]. In this method, the notion of network service curve [4, 7] plays a fundamental role. Its basic form is assumed to be deterministic and time-invariant, but there is also possibility to consider a more sophisticated stochastic variant. The latter is called the stochastic service curve [4]. Furthermore, in each of the aforementioned variants, the influence of the so-called cross traffic (when servicing the through traffic) can be taken into account. For purposes of this paper, we use such a variant of the conventional deterministic service curve [4] that depends upon the system cross traffic intensity. We call it a system parametric service curve for through traffic.

In general, such the parametric service curve will describe a non-stationary traffic system. However, we will assume in this study that it is quasi stationary (depending only upon some parameters of the cross traffic) in time intervals in which the traffic is analyzed and/or controlled. In other words, it will be assumed to be approximately time-invariant in these periods. At this point, we point out that the similar assumptions underlie also derivations of the $\in$-effective service curve [7]. So, because of this fact, it will be useful to look for some relationships between the aforementioned curves; we will do this in the next sections. Both of them try to determine, on their own way, how much bandwidth is actually left over in a traffic system in a given approximately stationary interval for servicing the through traffic.

This paper is organized as follows: After a short Introduction, we discuss in Section II the probing traffic pattern devised in [8] in the context of mechanisms of the carrier sense multiple access with collision avoidance (CSMA/CA) and the request-to-send (RTS) handshaking procedure used in ad hoc networks. We explain in more detail how these mechanisms work in the presence or absence of the cross traffic, pointing out the differences between these scenarios. Section III is devoted to a detailed explanation of the means of constructing the $\in$-effective service curve; it aims in making a needed correction of a basic expression provided for it in [8]. The next section presents a new interpretation of the evaluation of the $\in$-effective service curve as performing the measurement with the use of Dirac impulse. Finally, Section V presents the concluding remarks.

1. PROBING TRAFFIC FOR MEASURING THE EFFECTIVE SERVICE CURVE

The $\in$-effective service curve can be applied to different kinds of traffic. However, it was originally devised [8] for the use in ad hoc networks. More specifically, development of the effective service curve presented in [8] is related to wireless networks using the carrier sense multiple access with collision avoidance (CSMA/CA) mechanism of the medium access control (MAC) [3]. Moreover, the networks considered in [8] and in this paper use handshaking procedure. This means that the transmitter sends the request-to-send (RTS) packet to its neighbor, and the destination receiver, if ready to receive the data, responds sending out the clear-to-send (CTS) packet. After receiving the CTS packet, the transmitter sends the data packet. Receipt of this packet must be acknowledged by the destination receiver by sending the so-called acknowledgement (ACK) packet to the transmitter. This procedure is schematically depicted in Fig. 1, where the periods DIFS and SIFS mean the so-called distributed and short, respectively, inter frame spaces. Also, note that the propagation delays between the transmitter and the destination receiver are omitted in Fig. 1 for simplicity.

For more details regarding the CSMA/CA procedure, see, for example, [3].

In what follows, we refer to as exactly that modeling environment, shortly described above, which was assumed by Valaee and Li in [8]; for that environment, we calculate effective service curves using simulated data. Moreover, we use also in this study, for comparison purposes, the same notation as was assumed by Valaee and Li.

In the measurement method presented in [8], the probing traffic consisted of a sequence of packets named a greedy pattern of probing packets. This greedy pattern means that at the moment at which a given probing packet leaves the transmitter buffer, the next one is generated and put immediately into the queue of probing packets.

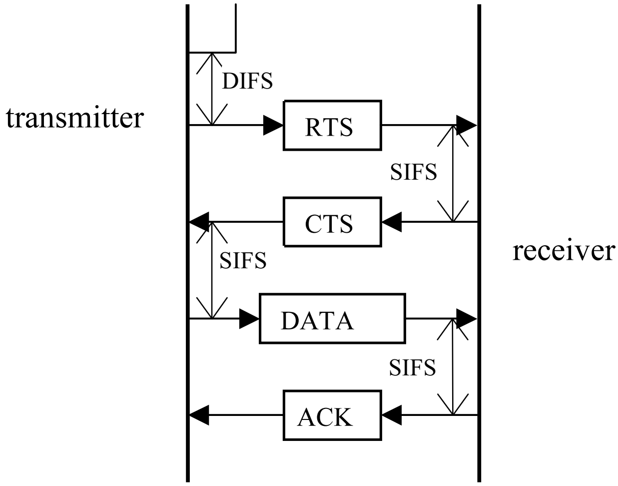

Fig. 1. Schematic illustration of the CSMA/CA procedure for sending the through traffic packet, with the use of RTS and CTS packets, and the acknowledgement

Let us denote a moment of the *i*th packet arrival at the destination receiver (including acknowledgement) by τ_i, $i = 1, 2, \ldots$. Using this, we can express the delays of the successive probing packets in the head of the probing queue as

$$\delta_i = \tau_i - \tau_{i-1} \quad \text{with} \quad i = 1, 2, \ldots \quad \text{and} \quad \tau_0 = 0 \, . \tag{1}$$

After [8], we assume here that the delays δ_i consist of two components. That is they can be expressed as

$$\delta_i = b + w_i \, , \tag{2}$$

where b stands for the total transmission time of the probing packet and is given by

$$b = T_{DIFS} + 3T_{SIFS} + T_{RTS} + T_{CTS} + T_{ACK} + T_{PRB} \tag{3}$$

for the scheme shown in Fig. 1. In (3), the times T_{DIFS} and T_{SIFS} stand for the duration times of the DIFS and SIFS periods, respectively. Moreover, the remaining ones in (3), i.e. T_{RTS}, T_{CTS}, T_{ACK} and T_{PRB}, are the transmission times of the RTS, CTS, ACK, and of the probing data packets, accordingly. Further, $T_{PRB} = L_{PRB} / C$, where L_{PRB} stands for the probing packet length in bits (packets are assumed to have the same length), and C is the channel transmission rate. Finally, w_i in (2) means the *i*th packet total waiting time. It is assumed that w_i depends upon the number of active nodes, upon the lengths of the transmitted packets of the cross traffic, and upon the size of the backoff window.

Consider now a case, which was also considered in [8], when no cross traffic is present in the system. Obviously in this case, apart from two nodes, other ones are not active, so no cross traffic packets are sent. Nevertheless, according to the IEEE 802.11 standard description given in [6] (see also, for example, [1] and [2]), the backoff window is then also used in transmitting each of the packets of a sequence, however, with exception of the first one (initial one). So, certainly for the initial packet, we shall have $w_0 = 0$. But all the other waiting times w_i, $i = 2, 3, \ldots$, will have random values different from zero.

Nevertheless, note that an occurrence of w_i, $i = 2, 3, \ldots$, having zero values cannot be fully excluded. This is so because the backoff time, being a random integer number of time slots, is taken from the interval $(0, CW-1)$, where CW (or more precisely, $CW-1$) denotes the length of the backoff window. Hence, occurrence of the backoff time equal to zero is possible. Of course, it will happen very rarely.

The use of the backoff window also in the absence of cross traffic (in transmitting the second, third, and next packets) follows from a rule embedded in the IEEE 802.11 standard. This rule enables the stations, which share the medium, a fair access to it. It says that a station that has just finished transmission of a packet and has the next one ready to transmit cannot send it out immediately (of course, after waiting first the usual time interval DIFS). In the next step, it always carries out the backoff procedure. And in the time of performing this procedure, other stations, having at a given moment a smaller value of the backoff time, get a chance to transmit their packets (if have any).

Obviously, the contention procedure using backoff window and described shortly above is superfluous in the case of lack of cross traffic. However, it does not matter because occurrence of such the situation is absolutely unusual in a real network.

After performing the initial transmission, the first backoff window length is set to $CW = CW_{\min}$, where $CW_{\min}$ means a minimal value of CW chosen for a given technology. For example, for the Direct Sequence Spread Spectrum (DSSS) technology, $CW_{\min} = 32$ slots, which gives 620 μs for the value of $CW-1$ expressed in μs [2]. So, in other words, the backoff time in this case can assume, randomly, one of the discrete values taken from the set {0 μs, 20 μs, …, 620 μs}. And consequently, if the backoff window is not doubled, w_i has just the same value as indicated above for the backoff time.

More complicated situation occurs when the channel impairments cause that the transmission fails and the backoff window is doubled. And such a means of extending the contention window size, as sketched above, can be continued [2, 6], if needed, up to achieving the assumed maximal value $CW_{\max}$ of CW. For example,

$CW_{\max} = 2024$ slots for the DSSS technology. Furthermore, in the variant of the IEEE 802.11 standard named IEEE 802.11e, packets at a node can be assigned to four different queues competing with each other. And this, obviously, can cause increase of the backoff window.

Comparison of the above description of the process of transmission in the absence of cross traffic with that in its presence (see, for example, [8] and [1, 2, 6]) shows that they are in principle identical, from the point of view of the contention window used. However, viewed as stochastic processes, they will differ more or less from each other. Generally, the representatives of the through traffic transmission processes in the presence of cross traffic will be far more involved than those taking place in its absence. Obviously, the first ones will depend hardly upon the statistical characteristics of the cross traffic.

Note that an example given in [8, Fig. 3, p. 1247] confirms the fact stated above. The random waiting times of the probing packets changing from about 4,25 ms to about 4,85 ms illustrated therein, which were registered in the absence of cross traffic in a simulated network, correspond to $b \cong 4{,}25$ ms and w_i's, i = 1,2, …., varying randomly in the range from 0 to about 0,6 ms. So, from the previous discussion of the influence of contention window lengths on the values of w_i's, i = 1, 2, …, it follows that doubling of the backoff window did not occur in simulations performed in [8]. And, in these simulations, the probabilistic characteristics of w_i's, $i = 1,2,\ldots$, were solely determined by the statistical characteristics of the backoff time.

Obviously, such a simple scenario regarding the waiting times w_i's, i = 1,2, …, will rather not take place in the case of cross traffic occurrence. Then, it will be more complicated.

2. BACKGROUND OF THE EFFECTIVE SERVICE CURVE FOR PRACTICAL EXPLOITATION

The method of estimating the $\in$-effective service curve devised in [8] uses a batch of probing packets that are sent out at the system input. This can take place in the presence as well as absence of the crossing traffic. The input packet sequence applied is generated in a greedy fashion, as explained in the previous section. The probing packets sent constitute the through traffic. Generally, they are not serviced immediately, but must wait some periods that depend upon the cross traffic intensity. These periods, that is the delays δ_i given by (2), are measured, and afterwards released from the constant delay b given by (3). As the result, we get the waiting times $w_i = \delta_i - b$. Next, on applying a batch of probing packets that comprises K consecutive packets, we express the successive sums of the waiting times as

$$\Delta^{(k)} = \sum_{i=1}^{k} w_i, \;\; k = 1,2,\ldots,K \tag{4}$$

where w_i means, as expressed by (2), the waiting time of the *i*th probing packet. (By the way, (4) here represents a little bit simplified version of an equivalent expression given in [8].) Further, assume that the times (variables) $w_i,\ i=1,2,...,K$, are random, then their sums $\Delta^{(k)},\ k=1,2,...,K$, are also random. Moreover, we assume that the sequence $\{w_i\}$ is an independent identically distributed one. And it is used to define the "most probable values of sums of delays" in such a way

$$T_k^{\in} = \inf\left\{\tau \middle| \Pr\left(\Delta^{(k)} > \tau\right) \le \in\right\},\ \ k=1,2,...,K, \tag{5}$$

where $\Pr\left(\Delta^{(k)} > \tau\right) \le \in$ means that the probability of occurrence of the event $\Delta^{(k)} > \tau$ is less than or equal to $\in$. Simply, (5) states first that all the values of τ that fulfil the above relation are found, and then the infimum operation is performed on the set obtained. (At this point, we remember that the infimum operation, denoted here by a symbol inf, means taking the greatest value of the lower bounds of a set considered.)

Using (4) and (5), the $\in$-effective service curve $S_{\in}(t)$ was defined in [8] as a piecewise linear function consisting of *K* segments of straight lines, described by

$$S_{\in}(t) = \frac{t - T_{k-1}^{\in}}{T_k^{\in} - T_{k-1}^{\in}} + k - 1,\ \ T_{k-1}^{\in} \le t \le T_k^{\in},\ \ k=1,2,...,K, \tag{6}$$

with the index *k* referring to the *k*th curve segment, and with the initial value of $T_0^{\in} = 0$.

The expression given by (6) is not, however, fully correct because of a "squeezed" time variable *t* used in it. That is the time variable *t* used therein does not describe a really elapsing time; it equals the really elapsed time minus the corresponding number of periods *b* (in which the packet transmission tooks place). Because of this fact, the effective service curve given by (6) is rather useless. Convolution of this service curve with an input traffic would not lead to receiving a correct result.

Taking into account the argument invoked above, it can be easily shown that the correct expression for $S_{\in}(t)$, instead of that in (6), is given by

$$S_{\in}(t) = \frac{t - \left(T_{k-1}^{\in} + (k-1)b\right)}{T_k^{\in} - T_{k-1}^{\in} + b} + k - 1,\ \ T_{k-1}^{\in} + (k-1)b \le t \le T_k^{\in} + kb,\ \ k=1,2,...,K \tag{7}$$

for a really elapsing time *t*.

Note that the traffic expressed with the use of $S_{\in}(t)$ given by (7) is counted in packets. To express it in bits, the service curve must be multiplied by the packet length (in bits). We remind that the latter was denoted above as L_{PRB}.

Obviously, in practical applications of the effective service curve, the probability $\Pr\left(\Delta^{(k)} > \tau\right)$ occurring in (5) is unknown. In these cases, however, we can estimate (approximate) the "probabilistic" sums of delays, $T_k^{\in}$, $k = 1,2,...,K$, in (7) by their actual values $\Delta^{(k)}$, $k = 1,2,...,K$. So, we write then

$$T_k^{\in} \cong T_k = \Delta^{(k)}, \;\; k = 1,2,...,K. \tag{8}$$

Such the simplification of (5) as given by (8) is in fact used in measurements of the knee points $T_k^{\in}$, $k = 1,2,...,K$, of the curve (7).

Consider now the case of sending probing packets in the absence of any cross traffic. Then, as we know from the analysis carried out in the previous section, we shall have $w_1 = 0$ (by making $\Delta^{(1)} = 0$); however, all the other waiting times w_i's, $i = 2, 3, \ldots$, will be greater than zero (with the probability approaching one). Hence, we shall also have all the next $\Delta^{(k)}$, $k = 2,3,...,K$, greater than zero (with the increasing probability of certainty, as they are the sums of w_i's, $i = 2, 3, \ldots$.)

Let us now apply $\Delta^{(1)} = 0$ in (5). Then, we get $\Pr\left(\Delta^{(1)} > \tau \geq 0\right) = 0$ satisfied for every $\tau \geq 0$. In consequence, the inequality $\Pr\left(\Delta^{(1)} > \tau\right) \leq \in$ occurring in (5) will be satisfied for every $\tau \geq 0$ and for every $\in \geq 0$. So, from this, we shall be able to conclude that $T_1^{\in} = 0,$ after performing the operation of taking infimum in (5).

By the way, note also that using (instead of (5)) the estimate of $\Delta^{(1)} = 0$ in (8), one gets exactly the same result. That is $T_1^{\in} = T_1 = \Delta^{(1)} = 0.$

Applying $T_1^{\in} = 0$ in (7) leads to

$$S_{\in}\left(t\right) = \frac{t}{b}, \;\; 0 \leq t \leq b. \tag{9}$$

Looking now at (9), observe that even in this specific case, when $T_1^{\in}$ equals zero, the effective service curve $S_{\in}\left(t\right)$ given by (7) assumes finite values (it does not equal infinity or gives an undefined symbol as for example 0/0). Evidently, this is not the case if we use expression (6) after [8]. Note that then, using $k = 1$ and $T_1^{\in} = 0$ in (6), we get $S_{\in}\left(t\right) = t/0 = \infty$ for $t > 0$ and $S_{\in}\left(t\right) = 0/0$ for $t = 0$. Both the two latter results are evidently not correct. So, really, formula (6) given in [8] cannot be assumed to be correct for the index $k = 1$; then, (7) should be used. Furthermore, the arguments used by derivation of (7) convince also that, for all the next indices $k = 2,3,...,K$ (with the corresponding delays $T_{k-1}^{\in} \leq T_k^{\in}$, $k = 2,3,...,K$), (7) represents an appropriate formula for the effective service curve (not (6)). As mentioned before, it is a piecewise linear function with finite slopes in its segments. And because $0 \le T_k^{\in} - T_{k-1}^{\in}$, $k = 2,3,...,K$, holds, the relation between the slopes in the corresponding segments of this function can be written as

$$\frac{1}{T_k^{\in} - T_{k-1}^{\in} + b} < \frac{1}{b}, \; k = 2,3,...,K. \tag{10}$$

Observe from (9) and (10) that the slope cannot be greater than $1/b$ (or L_{PRB}/b in bits/s) in any of the segments of the piecewise linear function given by (7). So $1/b$ is the maximal throughput for the through traffic in the absence of the cross traffic.

3. MEASURING SCHEME IN WHICH A DIRAC IMPULSE IS USED

Construction of the greedy pattern of probing packets used in the method of [8] was described in Section 1. Here, we recall that a basic mechanism in this pattern is the following: if a given probing packet of the pattern leaves the transmitter buffer, the next one is generated and put immediately into the queue of the probing packets.

Consider next a virtual situation (an artificial one) in which all the *K* probing packets appear jointly at the same time at the transmitter buffer at the starting time instant $t = 0$.

Compare then these two situations described above (real with artificial one). Observe that traffic servicing process of a system will not distinguish between them. The results of servicing will be identical in both the cases. The second description (that is that virtual one) will be, however, more convenient for us for analysis of the measurement process. So, in what follows, let us denote the input through traffic by $A^t(t)$ and describe it by the following function

$$A^t(t) = \begin{cases} K & \text{for } t \ge 0 \\ 0 & \text{for } t < 0 \end{cases}. \tag{11}$$

This function is illustrated in Fig. 2.

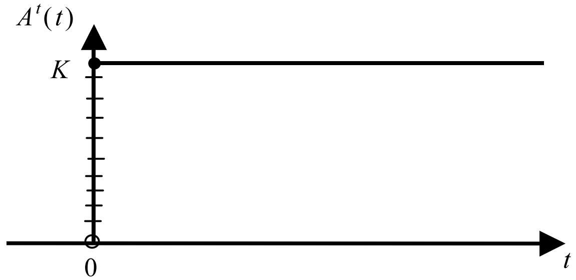

Fig. 2. Illustration of the function $A^t(t)$ given by (11)

Observe now that $A^t(t)$ given by (11) approaches the so-called Dirac function [7] (named also the burst impulse or the Dirac impulse [4]), when the number K of probing packets goes to infinity. In the network calculus, this special impulse (function) is defined as [4, 7]

$$\delta(t)=\begin{cases}\infty & \text{for } t\geq 0\\ 0 & \text{for } t<0\end{cases}. \tag{12}$$

We interpret the last result saying that the function $A^t(t)$ in an idealized form (that is when the parameter K goes to infinity) approaches the burst impulse (12). (By the way, note that then $A^t(t)$ is not a function anymore because $\delta(t)$, strictly saying, is not.).

Further, assume that a traffic system analyzed behaves approximately linearly. That is the following

$$D^t(t)\cong\inf_{0\leq\tau\leq t}\left\{A^t(\tau)+S^t(t-\tau)\right\} \tag{13}$$

holds, where $D^t(t)$ and $S^t(t)$ mean the cumulative through traffic at the system output and its service curve [4, 5, 7], respectively. In (13), the service curve $S^t(t)$ is assumed to be deterministic. However, because it depends upon the parameters of the cross traffic, we shall call it a parametric one.

Moreover, we use the following relation

$$D(t)=\inf_{0\leq\tau\leq t}\left\{\delta(\tau)+D(t-\tau)\right\}, \tag{14}$$

which was proved, for example, in [1], where $D(t)$ means any cumulative traffic function. We use also the approximation of (11) by (12). Altogether, this gives

$$D^t(t)\cong S^t(t). \tag{15}$$

To proceed further, we recall now, see Section 2, that construction of the effective service curve $S_\in(t)$ can be described as servicing the through traffic in the successive time intervals according to a rule of constant rate server(s). In other words, it can be viewed as a kind of service curve calculated for "a virtual constant rate server in the successive periods", and with "processing rate of this server varying from period to period". Obviously, its successive "period-constant" rates are given by

$$\frac{1}{T_k^\in-T_{k-1}^\in+b},\ k=1,2,\ldots,K \tag{16}$$

(see (10)). So, as such, the effective service curve $S_\in(t)$ can be assumed to be an estimate of the service curve $S^t(t)$ introduced above, and vice versa. Hence, taking also (15) into account, we write

$$S^t(t)\cong S_\in(t)\cong D^t(t) \tag{17}$$

where the symbol "$\cong$" means „estimate of".

In summary, interpretation of the final result in this section expressed by (17) is as follows: Applying the greedy pattern of probing packets, that is shaping the system input through traffic as described in Section 1, results in its output through traffic being approximately equal to the system service curve (for this traffic), when the system considered is linear or approximately linear. Furthermore, the system effective service curve $S_{\in}(t)$ can be estimated by the service curve for the through traffic $S^t(t)$, and vice versa. And finally, the approximates of both the service curves can be quickly obtained by measuring the system output through traffic.

CONCLUSIONS

In this paper, we have discussed some practical aspects of a technique devised in [8] for ad hoc networks that uses the so-called $\in$-effective service curve. Among other things, the basic expression describing the above curve has been corrected to the form given by (7); this form allows the correct interpretation of servicing the through traffic in the absence of cross traffic. Furthermore, it has been shown that the use of the greedy pattern of probing packets used in the method of [8] and described also here in Section 1 can be interpreted approximately as applying the Dirac impulse to the system through traffic input. It has been also shown that the system effective service curve $S_{\in}(t)$ can be estimated by the service curve for the through traffic $S^t(t)$, and vice versa. Finally, we have demonstrated that the approximates of both the aforementioned service curves can be quickly obtained by measuring the system output through traffic.

REFERENCES

1. Agrawal R., Cruz R.L., Okino C., Rajan R., *Performance bounds for flow control protocols*, IEEE/ACM Trans. Networking, June 1999, vol. 7, no. 3, s. 310–323.
2. Bianchi G., *Performance analysis of the IEEE 802.11 distributed coordination function*, IEEE J. Sel. Areas Commun., March 2000, vol. 18, no. 3, s. 535–547.
3. Brenner P., *A Technical Tutorial on the IEEE 802.11 Protocol*, BreezeCOM, 1997.
4. Fidler M., *A survey of deterministic and stochastic service curve models in the network calculus*, IEEE Communications Surveys & Tutorials, First Quarter 2010, vol. 12, no. 1, s. 59–86.
5. Firoiu V., Le Boudec J.-Y., Towsley D., Zhang Z.-L., *Theories and models for Internet quality of service*, Proc. IEEE, May 2002, vol. 90, no. 9, s. 1565–1591.
6. IEEE 802.11 WG, *Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification*, 1999.
7. Le Boudec J.-Y., Thiran P., *Network Calculus. A Theory of Deterministic Queuing Systems for the Internet*, Springer Verlag, Berlin, Germany, 2004.

8. Valaee S., Li B., *Distributed call admission control for ad hoc networks*, Proc. IEEE 56th VTC, September 2002, s. 1244–1248.
9. Xiao Y., Rosdahl J., *Throughput and delay limits of IEEE 802.11*, IEEE Commun. Lett., August 2002, vol. 6, no. 8, s. 355–357.

O PRAKTYCZNYCH ASPEKTACH ZASTOSOWANIA EFEKTYWNEJ KRZYWEJ SERWISOWEJ W SIECIACH TYPU AD HOC

Streszczenie

Tak zwana $\in$-efektywna krzywa serwisowa pojawiła się w literaturze przedmiotu przy rozpatrywaniu rozproszonej obsługi żądań dostępu w bezprzewodowych sieciach typu ad hoc. Jej praktyczna użyteczność polega na tym, że może ona być konstruowana w czasie rzeczywistym przy wykorzystaniu danych pomiarowych. Może ona być również na bieżąco modyfikowana w zależności od zmian intensywności tzw. ruchu krzyżowego w sieci, dopasowując obsługę żądań do aktualnego obciążenia sieci. W tym artykule pokazano, jak $\in$-efektywną krzywą serwisową można aproksymować za pomocą parametrycznej krzywej serwisowej sieci dla ruchu głównego, która w tym przypadku będzie zależeć od intensywności ruchu krzyżowego. Pokazano również, że podane w literaturze wyrażenie, opisujące $\in$-efektywną krzywą serwisową, nie do końca jest poprawne i musi być skorygowane. W tej pracy wyprowadzono wzór w pełni poprawny, który pozwala również na poprawny opis obsługi głównego ruchu przy braku w sieci ruchu krzyżowego. Ponadto pokazano, że użycie w pomiarach tzw. łapczywej (ang. greedy) próbkującej sekwencji bitów można zinterpretować w przybliżeniu jako użycie impulsu Diraca na wejściu do sieci dla ruchu głównego.

Słowa kluczowe: *rachunek sieciowy, sieci typu ad hoc, efektywne i parametryczne krzywe serwisowe.*