Research Article

Limiting Energy Consumption by Decreasing Packets Retransmissions in 5G Network

Łukasz Apiecionek

Institute of Technology, Kazimierz Wielki University, Bydgoszcz, Poland

Correspondence should be addressed to Łukasz Apiecionek; lukasz.apiecionek@ukw.edu.pl

This article presents the potential of using Multipath Transmission Control Protocol for limiting the energy consumption in 5G network. The number of errors occurring during packet transmissions and in effect the number of retransmissions affect the consumption of energy by the devices in the network. The paper analyzes the potential energy savings from implementing an algorithm for detecting problems and predicting the future retransmissions. Although this is the main object of the paper, it must be emphasized that the proposed method also allows increasing the speed of transmission and improving the security of the data and it is easy to implement in 5G networks.

1. Introduction

A general opinion on the 5G network is that this technology could not be used in the near future, mostly due to the fact that the business segment remains unprepared for its implementation. Lots of GSM/3G/LTE operators have invested much in their infrastructure and what they now focus on is the return on this investment.

In a technical aspect, 5G networks are defined by the following parameters [1, 2]:

(i) 1 millisecond end-to-end round trip delay (latency).

(ii) 1000x bandwidth per unit area.

(iii) 10–100x number of connected devices.

(iv) (Perception of) 99.999% availability.

(v) (Perception of) 100% coverage.

(vi) 90% reduction in network energy usage.

(vii) Up to ten-year battery life for low power, machine-type devices.

Only a network meeting these provisions can be called a 5G network. An additional parameter is a new radio interface [1, 3–5]. The ability to work with more than one radio transmitter/receiver, called MIMO (multiple-input, multiple-output) [1–3, 6, 7], is also very important. Moreover, it is often postulated that 5G network should be able to coexist with the old network. It should use Ethernet protocols and treat the old network as a backup [1]. It is essential for critical infrastructures, where the reliability of the network is a top priority [8].

One of the existing technologies which can be taken into account for cross-layer optimization for 5G network communications and which is able to use MIMO as well as the existing infrastructure, is MultiPath Transmission Control Protocol (MPTCP in short). MPTCP technology allows using all the existing links to provide one stable and fast connection between two points of communication. The fact of using more than one connection in 5G network is something which does not neglect the MPTCP technology but can lead to limiting the energy consumption, which can be achieved by reducing the amount of retransmissions.

No less important than the reliability of 5G network is its speed. In order to eliminate the problems with delays in MPTCP, fuzzy logic can be applied, especially Ordered Fuzzy Numbers (OFN in short, called in some papers Kosinski's Fuzzy Numbers), for predicting problems in the network [9, 10]. This will allow predicting errors in the network and deciding on using a different link.

Current State of Research. MPTCP is not widely used today. The best known user of this technology is Apple company, which has implemented it in its IPad and IPhone devices.

1	2	3	4
Source port		Destination port	
Sequence number (4 octets)			
Acknowledgment number (if ACK set)			

FIGURE 1: TCP header.

FIGURE 2: Three-way handshake.

FIGURE 3: N different TCP connections are represented as one logical data.

FIGURE 4: MPTCP in the stack.

Most research on this solution focuses on the ability to maintain a stable connection between two points by the use of different links [11, 12], while the issues concerning Ordered Fuzzy Numbers are mostly investigated by the coworkers of professor Kosínski [9, 10]. However, by far no research has been performed on the usability of OFN as a means for predicting the increase in bit error rate in TCP (Transmission Control Protocol) transmission, which could allow for decreasing the number of retransmissions of packets. This paper aims to analyze this new practical application of OFN solution combined with MPTCP technology for limiting the consumption of energy during the transmission of data.

The following section of the paper describes MPTCP technology in regard to its features and implementation status. Section 3 describes OFN and their applicability in 5G network along with the MPTCP technology, for the purpose of limiting the energy consumption. The recapitulation of the paper covers conclusions and final remarks.

2. Multipath TCP

Prior to presenting the MPTCP technology, it is necessary to first introduce the concept of TCP [13], which is used for transferring data between the processes running on different machines. TCP can send data in two directions between two hosts. A unique identifier of the TCP connection is two pairs of values (one for each side of the connection)—IP and port number. By using checksums and sequence numbers, TCP provides a complete and orderly data exchange for higher layer applications. The header presented in Figure 1 contains all necessary details to establish a connection.

Before the application starts transmitting the information, it is necessary to exchange the initialization data, as presented in Figure 2. Host A sends a segment with set SYN flag, and then host B confirms receiving the packet and sends back SYN and ACK flag. Finally, host A sends an empty segment, with only ACK flag [13].

TCP connections cannot move from one IP address to another. When a PC switches from Ethernet to Wi-Fi, it is assigned a different IP address. Thus, all the existing TCP connections must be shut down and reestablished.

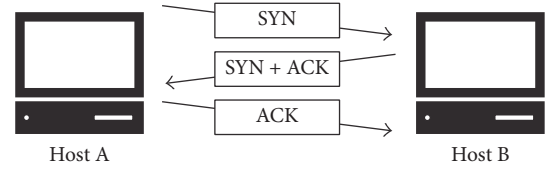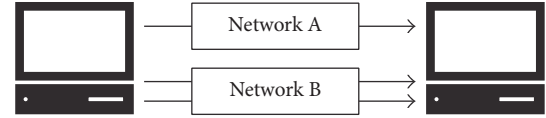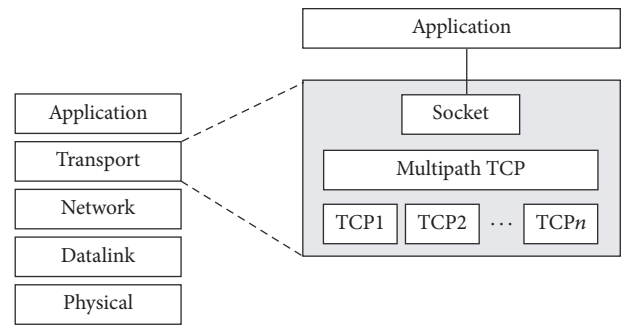MPTCP is a set of extensions to the specification of TCP which allows the client to establish multiple connections using different network cards with the same destination host. This way fault-tolerant and efficient data connections are maintained between the hosts that are compatible with the existing network infrastructures.

The main goal of this solution is to enable using multiple network paths for a single connection as presented in Figure 3. Its another advantage is that it increases the throughput of transport connections. This approach also allows controlling effectively the congestion between the network paths. Simultaneous use of MPTCP must not hinder the connectivity on a path where regular TCP works [14–16].

MPTCP is located at the transport layer and it aims to be transparent to both higher and lower layers, as presented in Figure 4. It is an additional function of higher layers of TCP standard.

New connection of MPTCP is established in the same way as a standard TCP, which is presented in Figure 5. However, the protocol is enhanced by a new feature. MP_CAPABLE option informs both hosts if the MPTCP connection can be established and if the data can be transmitted.

Nowadays, establishing a new connection is complicated by middle boxes (switches, routers). A pair (IP, port) of source and destination hosts is not sufficient for identifying the connection. Therefore, MPTCP extended its functionality with an additional option—MP_JOIN. Adding a new subflow is made in three steps presented in Figure 6.

First, MP_JOIN option provides a token which is generated with the key (truncated hash of the key), formed during
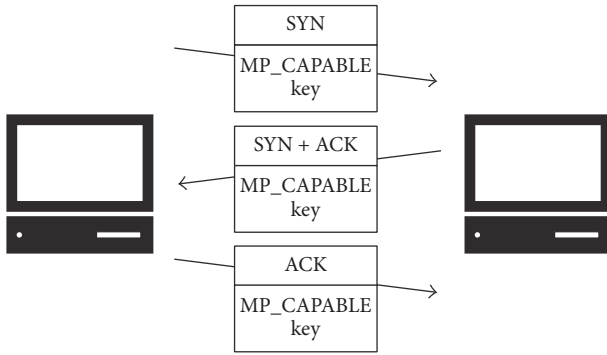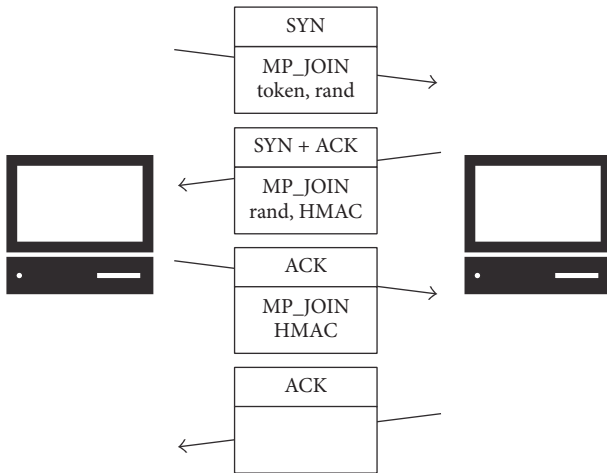
FIGURE 5: Establishing connection.

FIGURE 6: Adding a new subflow into MPTCP.

FIGURE 7: Error control in MPTCP.

Source port	Destination port
Data sequence number (8 octets)	
⋯	
Subflow sequence number (4 octets)	

FIGURE 8: MPTCP header (in short).

the initial connection. The exchange of HMAC (hash-based message authentication code) is the second step.

Now that the subflows have been established, MPTCP can use them to exchange data. Each host can send data over any of the established subflows. Furthermore, Figure 7 presents that the data transmitted over one subflow can be retransmitted on another to recover from losses.

Standard TCP "subflow sequence number" supports the reception of a single subflow and ensures detecting any loss of data. MPTCP uses "data sequence number" to sort the received data before passing them to the application [15]. MPTCP header is presented in Figure 8. When the source host wants to inform the destination host that it has no more data to send, it signals this fact with "Data FIN." It has the same semantics and behavior as a regular TCP FIN, but at the connection level.

In general, the users connect to the Internet on their smartphones via Wi-Fi or 3G, but not by both at the same time. If TCP's connection fails for some reason, it must be reestablished. MultiPath TCP avoids this situation by dynamically switching to the link, so the user does not waste time for reconnecting. It may also select the optimum speed.

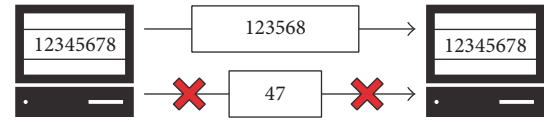The first mobile system that has been supporting MPTCP [16] is iOS 7. It ensures an uninterrupted transmission when one connection is defective or the connection is aborted. At the moment, MPTCP is used only for data transfers of Siri. Siri is an intelligent personal assistant which facilitates numerous actions of the users according to their wish. It allows them to send messages, schedule meetings, make phone calls, and perform many other tasks using voice commands. It is much likely that this option will be extended to the entire operating system.

MPTCP is able to increase the security level of the transmitted data by using many different links to reach the destination, in contrast to the present methodology, which is based only on network protection [17–19] and securing of possibility of connection into this network [20, 21]. The transmitted data are treated as raw binary data, which can be divided into blocks and passed to the transmission layer. With regard to protecting the data from being sniffed by a hacker, the author proposes an extended algorithm [22], which consists of the following steps:

Step 1: the data are divided into blocks,

Step 2: the data obtain a special sequence number, DSN, *Data Sequence Number*,

Step 3: the blocks are collected in a random sequence,

Step 4: the data are encoded,

Step 5: the blocks of data are passed to MPTCP socket, which will transmit them to their destination,

Step 6: the receiver side collects the blocks of data,

Step 7: the data are decrypted,

Step 8: the receiver side connects the blocks of data in the right order.

Table 1 presents transfer time for various file sizes, using two combinations of different links:

(i) Wi-Fi in three speed variants: 54, 22, and 11 Mbps, with 3G as a secondary link,

(ii) Wi-Fi in three speed variants: 54, 22, and 11 Mbps, with LTE in three different speed variants: 150, 50, and 5 Mbps, as a secondary link.

These results demonstrate the possibilities of transferring data using the proposed algorithm in many common situations, where mobile user does not get his full transfer speed.

TABLE 1: Data transfer results.

Data length [MB]	Transfer time [s] for different connection types					
	Wi-Fi 54 & 3G	Wi-Fi 22 & 3G	Wi-Fi 11 & 3G	Wi-Fi 54 & LTE 150	Wi-Fi 22 & LTE 50	Wi-Fi 11 & LTE 5
1	0.02	0.05	0.09	0.01	0.02	0.07
10	0.2	0.48	0.92	0.05	0.15	0.69
100	2	4.78	9.17	0.54	1.53	6.88
1000	20	47.83	91.67	5.39	15.28	68.75

TABLE 2: Packet transfer results.

Connection type	Number of packets transferred over Wi-Fi	Number of packets transferred over 3G/LTE	Transfer time [s]
Wi-Fi 54 & 3G	1080	20	0.02
Wi-Fi 22 & 3G	1052	48	0.05
Wi-Fi 11 & 3G	1008	92	0.09
Wi-Fi 54 & LTE 150	291	809	0.01
Wi-Fi 22 & LTE 50	336	764	0.02
Wi-Fi 11 & LTE 5	756	344	0.07

Such situation may occur in many places due to the specificity of building construction and network overload.

Table 2 presents statistical results of 1 MB data packet transfer over various connection types. As it can be noticed, the packet count transferred over different connection types depends on the connection speed and can be changed in an adaptive way.

3. OFN for 5G Network

In common sense, the term "fuzzy" refers to the situation where some precise data, like metric values or exact quantity of objects, cannot be given. Instead, words like "less," "a little," and "some" are used. Such verbal description of the reality is specific for the branch of artificial intelligence called fuzzy logic. The concept was developed by American professor of Columbia University in New York City and Berkeley University in California, Zadeh, who published a paper "Fuzzy Sets" in "Information and Control" journal in 1965 [23]. He defines the term of a fuzzy set, which allows describing imprecise data by means of values from the interval (0, 1). A number assigned to the set defines its degree of membership in this set. In his theory Zadeh used a three-valued logic, which was defined 45 years earlier by Polish researcher Łukasiewicz [24]. The next important step in the history of this branch of science was the introduction of L-R representation of fuzzy numbers, proposed by Dubois and Prade [25–27]. One of the less-known methods of generalization of fuzzy logic in a concept developed by Kosiński [28] and his coworkers, called Ordered Fuzzy Numbers, and it allows linking the change of trend to a fuzzy number.

The first attempt to describe the new operations on fuzzy sets was undertaken in early 1990s by Kosiński and Słysz [29]. Further studies of Kosiński, published in cooperation
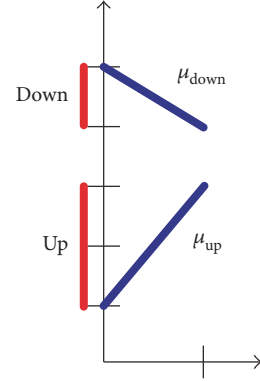

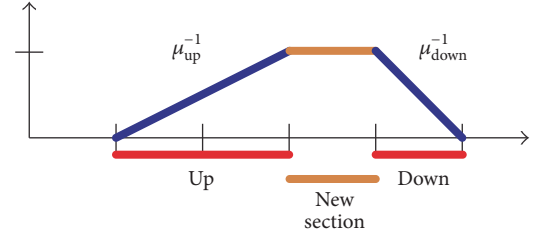FIGURE 9: Ordered Fuzzy Number—function up and down.

FIGURE 10: Ordered Fuzzy Number presented in a way referring to fuzzy numbers.

with Prokopowicz and Ślęzak, led to the introduction of the Ordered Fuzzy Numbers model, OFN [30–32], as follows.

Definition 1. An ordered fuzzy number A is an ordered pair of functions

$$A = \left(x_{\text{up}}, x_{\text{down}}\right), \tag{1}$$

where $x_{\text{up}}, x_{\text{down}}: [0, 1] \rightarrow R$ are continuous functions.

Respective parts of the functions are called part up and down and are presented in Figure 9. Figure 10 presents the OFN in a way referring to fuzzy numbers.

The continuity of the two parts called UP and DOWN shows that they are limited by a specific range. This range has been defined by the following values:

$$\begin{aligned} \text{UP} &= \left(l_A, 1_A^-\right), \\ \text{DOWN} &= \left(1_A^+, p_A\right). \end{aligned} \tag{2}$$

If both functions within the fuzzy number are continuous, then their inverse functions x_{up}^{-1} and x_{down}^{-1} are defined by their
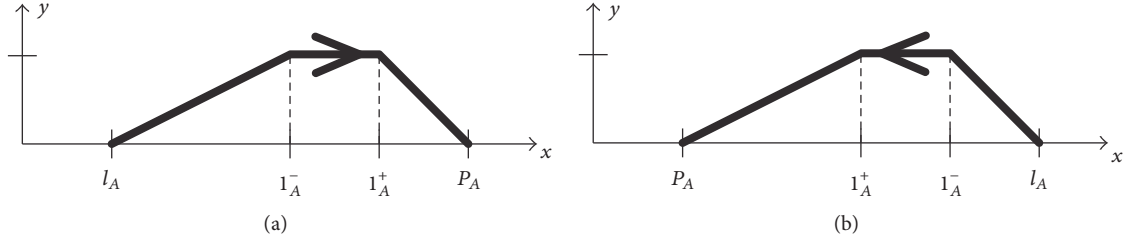
FIGURE 11: Fuzzy number in OFN notation where the order is (a) positive or (b) negative.

limits UP and DOWN. On this assumption the following equations are valid:

$$l_A := x_{up}(0),$$
$$1_A^- := x_{up}(1),$$
$$1_A^+ := x_{down}(1),$$
$$p_A := x_{down}(0). \tag{3}$$

If a constant function equal to 1 is added within the interval $[1_A^-, 1_A^+]$, then the result is functions UP and DOWN in one range (Figure 9, where $\mu_{down} = x_{down}$ and $\mu_{up} = x_{up}$), which may be treated as a carrier. Thereby, the function of membership $\mu_A(x)$ in the fuzzy set is defined on R set by the following formulas:

$$\mu_A(x) = 0 \quad \text{for } x \notin [l_A, p_A],$$
$$\mu_A(x) = x_{up}^{-1}(x) \quad \text{for } x \in \text{UP}, \tag{4}$$
$$\mu_A(x) = x_{down}^{-1}(x) \quad \text{for } x \in \text{DOWN}.$$

The fuzzy set defined in the aforementioned way acquires an additional parameter, order, whereas the following interval is the carrier:

$$\text{UP} \cup [1_A^+, 1_A^-] \cup \text{DOWN}. \tag{5}$$

The limit values for UP and DOWN functions are

$$\mu_A(l_A) = 0,$$
$$\mu_A(1_A^-) = 1,$$
$$\mu_A(1_A^+) = 1, \tag{6}$$
$$\mu_A(p_A) = 0.$$

In general, it can be assumed that Ordered Fuzzy Numbers take a form of a trapezoid. Each OFN can be defined by four real values:

$$A = (l_A \quad 1_A^- \quad 1_A^+ \quad p_A). \tag{7}$$

Figure 11 presents a sample of Ordered Fuzzy Numbers, including their characteristic points in a positive and negative order.

Functions f_A, g_A correspond to parts up_A and $down_A \subseteq R^2$, respectively, which gives

$$up_A = \{(f_A(y), y) : y \in [0, 1]\},$$
$$down_A = \{(g_A(y), y) : y \in [0, 1]\}. \tag{8}$$

Orientation corresponds to the order in graphs f_A and g_A.

Definition 2. A membership function of an ordered fuzzy number A is the function $\mu_A : R \to [0, 1]$ defined for $x \in R$ as follows:

$$\text{if } x \notin \sup p_A \Longrightarrow \mu_A(x) = 0$$
$$x \in (1_A^-, 1_A^+) \Longrightarrow \mu_A(x) = 1$$
$$x \in \sup p_A \wedge x \notin (1_A^-, 1_A^+) \Longrightarrow \tag{9}$$
$$\mu_A(x) = \max\left(f_A^{-1}(x), g_A^{-1}(x)\right).$$

Thus defined membership function can be applied for setting control rules in a similar way in classic fuzzy numbers. All quantities that can be found in fuzzy control describe a selected part of the reality. The process of determining this value is called fuzzy observation.

Definition 3. Reversal of the orientation of the ordered fuzzy number A is the replacement of the part up (function f_A) with part down (function g_A). This operation is described as follows:

$$B = A|^- \Longleftrightarrow g_B = f_A \wedge f_B = g_A, \tag{10}$$

where A means an ordered fuzzy number defined by the pair of functions (f_A, g_A), B is a result of the operation of reversal of the OFN orientation, and Sign "$|^-$" is a symbol of a reversed orientation of OFN. The number obtained in that way is called a reversed OFN or a reversed orientation number.

One of the most remarkable notations interpreting ordered fuzzy number is a set of key points:

$$[f(0), f(1), g(1), g(0)]. \tag{11}$$

Basic arithmetic operations, where ordered fuzzy number $A = (f, g)$ is a pair of affine functions and $B = (e, h)$ is another pair of affine functions, relate to the following formulas:

(i) addition: $A + B = (f + e, g + h) = C$,

$$C \longrightarrow [f(0) + e(0), f(1) + e(1), g(1) + h(1), g(0) \\ + h(0)], \tag{12}$$

(ii) scalar multiplication: $C = \lambda A = (\lambda f, \lambda g)$,

$$C \longrightarrow [\lambda f(0), \lambda f(1), \lambda g(1), \lambda g(0)], \tag{13}$$

(iii) subtraction: $A - B = (f - e, g - h) = C$,

$$C \longrightarrow [f(0) - e(0), f(1) - e(1), g(1) - h(1), g(0) \\ - h(0)], \tag{14}$$

(iv) multiplication: $A \times B = (f \times e, g \times h) = C$,

$$C \longrightarrow [f(0) \times e(0), f(1) \times e(1), g(1) \times h(1), g(0) \\ \times h(0)]. \tag{15}$$

As it was presented in Section 2, the MPTCP technology could be used for providing better network security in relation to such parameters as the destination reachability and network reliability. Of course, the transmission in all channels may generate some errors which require packet retransmission. This data retransmission will be made over the same channel in which the error occurred.

When the number of retransmissions rises, the energy consumption is increasing as well. The application of OFN could speed up the decision to change the data transmission path, which allows decreasing the number of transmission errors. OFN allow predicting data loss in the currently used channel and can also help make a decision to retransmit the packet faster and limit the number of retransmissions as well. Lowering the number of retransmissions allows limiting the energy consumption.

3.1. OFN for Errors Prediction. In this section the algorithm which uses OFN for detecting potential problems is described. The algorithm measures a TCP retransmission in all the used channels during the transmission and provides it in a percentage value of the transmitted packets. This measure is made both in a given period of time—a timeslot—and continuously. Four timeslots of the continuous measure could be defined in time:

$$t_i, t_{i-1}, t_{i-2}, t_{i-3}, \tag{16}$$

where t_i is a current timeslot.

These four results together give a fuzzy number in OFN notation, where

(i) $f_A(0)$ respond to t_{i-3},
(ii) $f_A(1)$ respond to t_{i-2},
(iii) $g_A(1)$ respond to t_{i-1},
(iv) $g_A(0)$ respond to t_i.

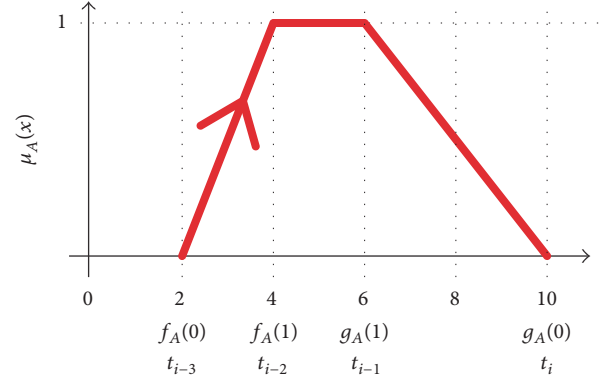This fuzzy number in OFN notation is presented in Figure 12.

Figure 12: Fuzzy number in OFN notation.

The definition of fuzzy observation of the connection used is as follows.

Definition 4. Fuzzy observation of C connection in time t_i is a set

$$\frac{C}{t_i} = \left\{ \frac{f_C(0)}{t_{i-3}}, \frac{f_C(1)}{t_{i-2}}, \frac{g_C(1)}{t_{i-1}}, \frac{g_C(0)}{t_i} \right\}, \tag{17}$$

where

$$t_i > t_{i-1} > t_{i-2} > t_{3-1}, \tag{18}$$

$|t_i - t_{i-1}| = |t_{i-1} - t_{i-2}| = |t_{i-2} - t_{i-3}| = t_n$, timeslot of the measurement

$$f_C(0) \le f_C(1) \le g_C(1) \le g_C(0). \tag{19}$$

This provides Lemma 5.

Lemma 5. *One has*

$$C_{positive} = \begin{cases} f_C(0) < f_C(1) < g_C(1) \\ \qquad\qquad or \\ f_C(1) < g_C(1) < g_C(0), \end{cases} \tag{20}$$

in another situation $C_{negative}$.

According to this definition, the counters of retransmissions in the links during the connection observation should give

(i) positive order of OFN when the packet retransmission count *increases*,
(ii) negative order of OFN when the packet retransmission count *decreases*.

The four measurements performed during the data transfer allow preparing fuzzy number in OFN notation. This gives a fuzzy observation of the MPTCP connections, defined as follows.

Definition 6. Fuzzy observation of the MPTCP connections is described by the following formula:

$$S_m = \sum_{i=1}^{n} \left\{ \frac{C_{positive}}{C_i * w_i} \middle| \frac{C_{negative}}{-C_i * w_i} \right\}, \tag{21}$$

where $w_i \in \{w_i, \ldots, w_n\}$ describes the impact on all connections.

According to Definition 2, it is possible to define the MPTCP scheduler using OFN.

3.2. The Algorithm of OFN Scheduler. The algorithm with application of OFN for predicting transmission errors in MPTCP transmission consists of four steps.

Step 1. The administrator declares the start value for w_i and L_i for all MPTCP links, where w_i describes the impact of errors on all connections and L_i describes the load of the whole data which should be sent by the i connections when the transmission starts. L_i is expressed in percentage rate.

Step 2. The amount of packets P_i which will be transferred over each connection during each timeslot is calculated using the following formula:

$$P_i = \frac{L_i}{\sum_{i=1}^n L_i} * \text{Data}. \tag{22}$$

Step 3. During the transmission the fuzzy observation of connection C_i is calculated for each connection according to data retransmissions and fuzzy observation of MPTCP S_i is calculated according to the given definition.

Step 4. When the calculated S_i is positive and it is higher than the acceptance level AL, it means that the number of errors is increasing on this connection. In this situation L_i for the given connection will be changed according to the following formula:

$$L_i = \frac{L_i}{\text{ErrorCorector}}. \tag{23}$$

When the calculated S_i is negative, the number of errors is decreasing on this connection. In this situation the L_i for given connection will be changed according to the following formula:

$$L_i = L_i * \text{ErrorCorector}. \tag{24}$$

The ErrorCorector is a value which describes how fast the system should stop using the connection in which the amount of errors increases. This value should be also declared by the network administrator.

3.3. Simulation Test Results. In order to check a MPTCP scheduler with OFN, a simulation was performed.

The system was designed with two connection links: Connections 1 and 2, labeled $C1$ and $C2$, with maximum speed 100 Gbit/s. The parameters of the algorithm were declared as follows:

(i) the corrector for the links ErrorCorector = 2;

(ii) acceptance level AL = 3;

(iii) load balance on start for connection $C1$ was $L_1 = 66$;

(iv) load balance on start for connection $C2$ was $L_2 = 34$;

(v) the timeslots used were 60 seconds.

TABLE 3: Fuzzy observation of the connections.

Time	% Error on connections		S [p: positive, n: negative]	
	C1	C2	S1	S2
1	1	3		
2	2	4		
3	2	6		
4	1	9	$[1, 2, 2, 1]$ p	$[3, 4, 6, 9]$ p
5	2	13	$[2, 2, 1, 2]$ n	$[4, 6, 9, 13]$ p
6	1	12	$[2, 1, 2, 1]$ p	$[6, 7, 12, 12]$ p
7	1	11	$[1, 2, 1, 1]$ p	$[9, 13, 12, 11]$ p
8	2	12	$[2, 1, 1, 2]$ p	$[13, 12, 11, 12]$ n
9	1	8	$[1, 1, 2, 1]$ p	$[12, 11, 12, 8]$ n
10	2	7	$[1, 2, 1, 2]$ p	$[11, 12, 8, 7]$ n
11	1	6	$[2, 1, 2, 1]$ p	$[12, 8, 7, 6]$ n
12	2	5	$[1, 2, 1, 2]$ p	$[8, 7, 6, 5]$ n
13	1	3	$[2, 1, 2, 1]$ p	$[7, 6, 5, 3]$ n
14	2	2	$[1, 2, 1, 2]$ p	$[6, 5, 3, 2]$ n
15	1	1	$[2, 1, 2, 1]$ p	$[5, 3, 2, 1]$ n

TABLE 4: Load balance for the connections.

Time	L: load balance	
	L1	L2
1	50	50
2	50	50
3	50	50
4	50	50
5	50	25
6	50	12,5
7	50	6,25
8	50	12,5
9	50	25
10	50	50
11	50	50
12	50	50
13	50	50
14	50	50
15	50	50

Table 3 presents the results of the used algorithm dependently on the number of errors during the data transfer. The fuzzy observation of the link gives OFN numbers $S1$ and $S2$.

As it can be observed, after the 4th measure the OFN numbers is generated and $S1$ exceeds the acceptance level, so OFN scheduler algorithm changes the load balance for the transmission, as it is presented in Table 4.

The total number of packets passed to the connections changes during the timeslots according to the load balance set at the beginning of the test. This affects the number of packets transferred through the connections and, depending on the

TABLE 5: Number of errors in packet transmission.

| Time | Errors count | | | | | |
| | MPTCP with OFN | | | MPTCP without OFN | | |
	L1	L2	L1 + L2	L1	L2	L1 + L2
1	400	1200	1600	400	1200	1600
2	800	1600	2400	800	1600	2400
3	800	2400	3200	800	2400	3200
4	400	3600	4000	400	3600	4000
5	1067	3467	4533	800	5200	6000
6	640	1920	2560	400	4800	5200
7	711	978	1689	400	4400	4800
8	1280	1920	3200	800	4800	5600
9	533	2133	2667	400	3200	3600
10	800	2800	3600	800	2800	3600
11	400	2400	2800	400	2400	2800
12	800	2000	2800	800	2000	2800
13	400	1200	1600	400	1200	1600
14	800	800	1600	800	800	1600
15	400	400	800	400	400	800

error level on the link, provides different number of errors. In the simulation there were 40000 packets passed to the link. Table 5 presents the numbers of errors in the connections with and without OFN scheduler in MPTCP. As it can be noticed, the total number of errors without OFN scheduler is higher than in the variant using this solution.

3.4. The Evidence for an Effect on Energy Consumption. The main goal of this paper is to analyze the effect of the proposed solution on the energy consumption. As it can be noticed, the system of transmission itself consumes much power. For instance, sending data packets over radio network in smartphones drains their batteries, while nowadays there are many processors available described as Low Energy Usage.

The presented algorithm does not require any complex calculations, so it does not cause any significant power dissipation. Rather than that, the problem is the energy consumption by the system of transmission. The process of sending data is power-consuming not only for the sender device, but also for the entire IT system. This system in the case of 5G networks comprises numerous devices: transmitters, switchers, and routers. Each of these devices on the packet's path uses energy. In a case where the packet is retransmitted the total energy consumption is even higher as

(i) the receiver has to realize that a problem with the received data occurred and, for example, recalculate the CRC checksum;

(ii) the receiver has to send the information to the sender about the problem with transmission;

(iii) the sender has to send the data again.

As it was described above, in the case of retransmission, the data is sent twice, and, moreover, additional data concerning the problem is transmitted over the network, which means that the system is used thrice.

The analysis of the proposed algorithm allows concluding that implementing simple calculations and making optimal decisions by the sender faster contributes to more efficient power management in the whole IT ecosystem.

4. Conclusions

In this article, a special algorithm for mobile connections, applicable in 5G network, was introduced. For this algorithm a MPTCP concept was used and the energy consumption as well as the security aspect were developed by a special usage of fuzzy observation of the errors in the transmissions.

The presented new concept of MPTCP scheduler combined with OFN was tested during a data transmission simulation. As it was presented in the previous section, the proposed algorithm allows reducing the number of retransmissions, as less packets are transferred over the connection link in which some problems are detected. Such application of MPTCP allows limiting the energy consumption by decreasing the amount of data which has to be transmitted. This is a possible usage of OFN for improving the existing solutions like MPTCP in a simple way, without any complicated algorithm requiring a high level processor performance. This is very important for the 5G network, in which the speed and power consumption is a given parameter.

As it was mentioned in the introduction, there are numerous configurations in which not only 5G, but also the existing infrastructure will be used. In this situation there will be more than one connection to the same destination. For this reason, the MPTCP with OFN could give a valuable solution in critical network connections. The presented algorithm is able to provide a better energy management and higher transmission security and reliability without generating higher costs in the implementation process [33–35].

Competing Interests

The author declares that there is no conflict of interests regarding the publication of this paper.

References

[1] Understanding 5G: Perspectives on future technological advancements in mobile, December 2014, https://www.gsmaintelligence.com/.

[2] 5G radio access, Ericsson White paper, Uen 284 23-3204 Rev C, April 2016.

[3] View on 5G Architecture, 5G PPP Architecture Working Group, https://5g-ppp.eu/white-papers/.

[4] 5G PPP Pre-Standardisation Working Group, Lisbon, Portugal, October 2015, https://5g-ppp.eu.

[5] M. Maternia and S. E. El Ayoubi, Eds., *5G PPP Use Cases and Performance Evaluation Models*, version 1, 2015, https://5g-ppp.eu/.

[6] 5G and Energy, 5G PPP Architecture Working Group, https://5g-ppp.eu/white-papers/.

[7] 5G Automotive Vision, "5G PPP Architecture Working Group," 2015, https://5g-ppp.eu/white-papers/.

[8] 5G and e-Health, 5G-Infrastructure-Association, September 2015, https://5g-ppp.eu/.

[9] J. Czerniak, "Evolutionary approach to data discretization for rough sets theory," *Fundamenta Informaticae*, vol. 92, no. 1-2, pp. 43–61, 2009.

[10] W. Kosínski, P. Prokopowicz, and A. Rosa, "Defuzzification functionals of ordered fuzzy numbers," *IEEE Transactions on Fuzzy Systems*, vol. 21, no. 6, pp. 1163–1169, 2013.

[11] C. Raiciu, C. Paasch, S. Barre et al., "How hard can it be? Designing and implementing a deployable multipath TCP," in *Proceedings of the Usenix Symposium on Networked Systems Design and Implementation*, San Jose, Calif, USA, April 2012.

[12] C. Paasch, G. Detal, F. Duchene, C. Raiciu, and O. Bonaventure, "Exploring mobile/WiFi handover with multipath TCP," in *Proceedings of the ACM SIGCOMM Workshop on Cellular Networks: Operations, Challenges, and Future Design (CellNet '12)*, pp. 31–36, Helsinki, Finland, August 2012.

[13] J. Postel, "Transmission control protocol," RFC 793, IETF, 1981.

[14] A. Ford, C. Raiciu, and M. Handley, "TCP extensions for multipath operation with multiple addresses," RFC 6824, 2013.

[15] U. Krieger, Evolution of Transport Protocols in high-Speed Networks, Lectures materials Multimedia-Kommunikation in Hochgeschwindigkeitsnetzen (KTR-MMK-M), 2014.

[16] K. M. Schneider, K. Mast, and U. R. Krieger, *Adapting Content-Centric Networking to the Characteristics of Multihomed Mobile Terminals*, 2014.

[17] Ł. Apiecionek and M. Romantowski, "Secure IP network model," *Computational Method in Science and Technology*, vol. 19, no. 4, pp. 209–213, 2013.

[18] L. Apiecionek and M. Romantowski, "Security solution for cloud computing," *Journal of Information, Control and Management Systems*, vol. 11, no. 3, 2013.

[19] L. Vokorokos, M. Ennert, M. Hartinger, and J. Radušovský, "A survey of parallel intrusion detection on graphical processors," in *Proceedings of the International Scientific Conference (INFORMATICS '13)*, Spišská Nová Ves, Slovakia, November 2013.

[20] W. R. Cheswick and S. M. Bellovin, *Firewalls and Internet Security: Repelling the Wily Hacker*, Addison-Wesley, 1994.

[21] B. Chapman and E. D. Zwicky, *Building Internet Firewalls*, O'Reilly & Associates, Inc., 1995.

[22] Ł. Apiecionek, M. Sobczak, W. Makowski, and T. Vince, "Multi Path Transmission Control Protocols as a security solution," in *Proceedings of the 13th IEEE International Scientific Conference on Informatics (INFORMATICS '15)*, pp. 27–31, Poprad, Slovakia, November 2015.

[23] L. A. Zadeh, "Fuzzy sets," *Information and Computation*, vol. 8, no. 3, pp. 338–353, 1965.

[24] J. Łukasiewicz, "O logice trójwartościowej," *Studia Filozoficzne*, vol. 270, no. 5, 1988.

[25] D. Dubois and H. Prade, "Operations on fuzzy numbers," *International Journal of Systems Science*, vol. 9, no. 6, pp. 613–626, 1978.

[26] D. Dubois and H. Prade, "Fuzzy elements in a fuzzy set," in *Proceedings of the 4th International Conference on Inertial Fusion Sciences (IFSA '05)*, vol. 5, pp. 55–60, September 2005.

[27] D. Dubois and H. Prade, "Gradual elements in a fuzzy set," *Soft Computing*, vol. 12, no. 2, pp. 165–175, 2008.

[28] W. Kosiński, "On fuzzy number calculus," *International Journal of Applied Mathematics and Computer Science*, vol. 16, no. 1, pp. 51–57, 2006.

[29] W. Kosiński and S. Słysz, "Fuzzy numbers and their quotient space with algebraic operations," *Bulletin of the Polish Academy of Sciences: Technical Sciences*, vol. 41, pp. 285–295, 1993.

[30] W. Kosiński, P. Prokopowicz, and D. Ślęzak, "On algebraic operations on fuzzy numbers," in *Intelligent Information Processing and Web Mining*, pp. 353–362, Springer, 2003.

[31] W. Kosiński, P. Prokopowicz, and D. Ślęzak, "Ordered fuzzy numbers," *Polish Academy of Sciences. Bulletin. Mathematics*, vol. 51, no. 3, pp. 329–341, 2003.

[32] W. Kosiński, P. Prokopowicz, and K. Frischmuth, *On Algebra of Ordered Fuzzy Numbers, Soft Computing Foundations and Theoretical Aspects*, Edited by J. K. Krassimir T. Atanassow, Olgierd Hryniewicz, EXIT, 2004.

[33] T. Andrysiak, Ł. Saganowski, M. Choraś, and R. Kozik, "Proposal and comparison of network anomaly detection based on long-memory statistical models," *Logic Journal of IGPL*, vol. 24, no. 6, pp. 944–956, 2016.

[34] Ł. Saganowski, T. Andrysiak, R. Kozik, and M. Choraś, "DWT-based anomaly detection method for cyber security of wireless sensor networks," *Security and Communication Networks*, vol. 9, no. 15, pp. 2911–2922, 2016.

[35] J. Byrski and W. Byrski, "A double window state observer for detection and isolation of abrupt changes in parameters," *International Journal of Applied Mathematics and Computer Science*, vol. 26, no. 3, 2016.