

Research Article

Wearable Modular Telemetry System for the Integrated Rescue System Operational Use

**Radim Kliment,¹ Pavel Smrčka,¹ Karel Hána,^{1,2} Jakub Schlenker,¹
Vladimír Socha,³ Luboš Socha,⁴ and Patrik Kutílek¹**

¹*Faculty of Biomedical Engineering, Czech Technical University in Prague, Sitna Sq. 3105, 272 01 Kladno, Czech Republic*

²*UCEEB, Czech Technical University, Trinecka 1024, 273 43 Bustehrad, Czech Republic*

³*Faculty of Transportation Sciences, Czech Technical University in Prague, Horská 3, 128 03 Prague, Czech Republic*

⁴*Faculty of Aeronautics, Technical University of Kosice, Rampova 7, 041 21 Kosice, Slovakia*

Correspondence should be addressed to Jakub Schlenker; jakub.schlenker@gmail.com

Received 6 October 2016; Revised 8 March 2017; Accepted 22 March 2017; Published 10 April 2017

Academic Editor: Stepan Gorgutsa

Copyright © 2017 Radim Kliment et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The article summarizes the development of the FlexiGuard modular telemetry system designed for enhancing safety of the Integrated Rescue System team members in solving crisis situations and for improving training processes. Further framework solutions, which lead to the development of automatic modular telemetry system allowing for real time monitoring of physiological parameters, are provided as well. The system provides for the signalization of critical states such as exhaustion, mental stress, and overheating. It further provides differentiation between the nature and intensity of movement, including actual and overall energy output, monitoring environmental parameters, and analysis of an intervention or training. The system has been tested in laboratories as well as in the terrain under real circumstances, and the eventual end users participated in its optimization process. Following the theory of games, a model of a transmission system was also created which demonstrates higher transmission efficiency when using higher number of nodes.

1. Introduction

In the recent years, the field of wearable telemetry systems for physiological parameters monitoring is developing owing to the development of modern technologies. These are used mainly by sportsmen, who may choose from a range of the so-called sport testers designed to monitor physical activity [1], heart rate, sleep, and other parameters. Similarly doctors, who are now capable of more precise diagnosing [2] or prescribing medication and/or adjusting treatment for the patient, are also interested in the field [3]. Telemetric systems monitoring health are also expanding to fields such as car industry [4, 5] or housing. One may also come across with so-called smart homes, capable of monitoring health of their tenants [6, 7], or with commercial products designed for monitoring health of the elderly [8].

Besides generally known applications in sports and medicine, the above concept of remote monitoring of psychophysiological condition is starting to find its use also

with professions requiring mental or physical resistance. It is therefore currently possible to see the utilization of telemetry systems for monitoring physiological and environmental parameters [9] in professions by which monitoring of people performing highly demanding and responsible tasks appears well-grounded. The basic requirement for such applications is to limit the obstruction of carrying out the tasks to the highest possible extent. It is thus vital to opt for compact wearable telemetry systems. An example of such application would be monitoring of pilots [10–12], in situations when it is impossible to measure physiological signals using standard medical devices due to their dimensions (e.g., ECG) in a plane cockpit. The importance of wearable telemetry systems capable of measuring and evaluating physiological condition and environmental parameters of the close surroundings of the user may be clearly seen with members of IRS (Integrates Rescue System). Excessive stress and fatigue threatens their health and may have negative effects on their momentary ability to react promptly and appropriately to crisis situations.

As a result of this, consequences ranging from material to health damages may take place. One of the alternatives to avoid such scenarios is offered by the discussed systems. The primary interest is therefore in surveillance systems capable of providing the status on the condition and close surroundings of the members of IRS in real time [13].

Currently, there is a number of available systems and concepts which may find use with members of Integrated Rescue Systems and their localization in urban spaces, obtaining data from the team members, monitoring physiological signals, and so forth. Most of these systems, however, provide only a limited number of useful features (e.g., measures only a limited number of specialized parameters) and are not open to customization or further development.

LifeNET [14], designed for localization of firefighters inside complex buildings, is one of the already applied systems. The general idea of LifeNET evolves around a device worn by a firefighter, capable of deploying a certain number of lifeline beacons. These then serve as so-called access points which detect particular team members in their close proximity as well as their distance from and position towards the beacon using an ultrasound receiver. This allows for locating firefighters inside complex buildings. Miniature monitors can be attached to the device allowing for a firefighter to see his position as well as positions of other firefighters towards beacons. Localization of firefighters using beacons takes place in cooperation with a device which is mounted to their boots. This device also contains a temperature sensor and is furthermore compatible with other devices, for example, accelerometers via I2C interface. Among other applications of LifeNET [15] there is also monitoring of various physical activities (e.g., cycling, jogging, and walking). Besides the fire brigade, the system has also been implemented for use in military fields with soldiers required to perform mentally and physically demanding tasks in their surrounding environments. From among medical applications of the system, continuous monitoring of physiological parameters in patients with Parkinson's disease or epilepsy [15], helping doctors adjust medications would be worth mentioning.

Another system designed for IRS applications is the MiTag system (Medical information Tag) [16], designed to collect information on the condition of a number of impacted persons. The system is based on the MiTag platform, which includes two wireless interfaces. One interface provides for communication with sensors and creates so-called body area network (BAN) and a wide range network of MESH type allowing for communication with screener. A part of the system are also repeaters which may be deployed along the way between the screening unit and a patient should it be the scenario that there is no direct reach of the signal from the platform on the patient to the screening unit. The MESH network protocol then automatically redirects the data flow through the repeaters which offers virtually unlimited range that the data can be sent over. The platform can be extended by a considerable number of various sensors such as GPS, pulse oximeter, blood pressure sensor, and temperature sensor [16].

The architecture of the wireless network designed directly for the needs of broadcasting the sensor measured data in

the case of fire brigade interventions is represented by the FireNet system [17]. It can reconfigure itself automatically as needed and provide data flow to the desired location. Sensors of various types are connected to the network either on the fire fighters themselves or on some of their equipment, such as vehicles. A GPS receiver is attached to a vehicle enabling their localization, and then by using the network itself it makes it possible to partially locate relative distances between points. The obtained data are transmitted to screening unit of the intervention commander and the using internet sent further to the fire brigade headquarters [17].

Among fairly perspective systems, there is also Zephyr Bioharness [18] or ProeTex [19]. Zephyr Bioharness is a multifunctional chest harness able to monitor perspiration, temperature, body activities, and life functions of its user. It is primarily designed for training and exercise. The manufacturer claims the theoretical possibility of the use of this device's special configuration for purposes of military units and it is able to connect to the communication equipment of user [18]. ProeTex is a project running under the 6th EU framework program. This project focuses primarily on the development of "smart textiles" which are designed for the future production of protective wear and accessories for firefighters. Textile sensors developed within this project aim primarily at monitoring basic body functions, physiological parameters, and activities and identifying potential chemical threats (toxic fumes, etc.) and the issues of providing power sources for such devices [19].

Even though there is a considerable overall number of system concepts based on the idea of wearable telemetry systems applicable in IRS, these are in most cases focused on measuring a limited number of specialized parameter. Considering that commercial systems and devices do not allow for recording of the measured data in all cases, which would be useful for offline processing and evaluation of the measured data, the design of the modular telemetry system was also directed towards these kinds of utilization. The aim of this paper was to develop a modular telemetry system [20, 21], which would suit the needs of respective Integrated Rescue System bodies as well as other potential users according to their needs and requirements with the main emphasis on the sensor base, modularity, data transmission security, possibility of visualization of data online using various kinds of devices, battery life, and, last but not least, the user. Within the defined concept, among other issues, it was necessary to research and design a solution of the issue with centralized, or synchronized management of data flow of the designed biotelemetry system of the PAN type. Therefore, besides the overall description of technology, construction, topology, and modular aspects, the article focuses in more detail on the issue of managing data flow from several sensors or the entire sensory systems.

2. Materials and Methods

Over the course of telemetry system design phase, three firefighter units were contacted in order to obtain a portfolio of features desired by the user. The following rank among the major requirements: maximum user comfort (system must

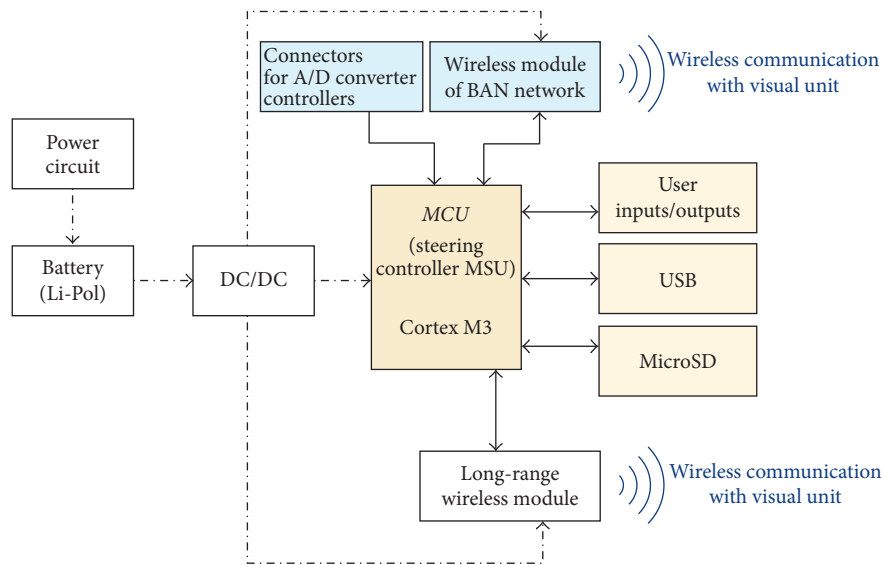


FIGURE 1: Hardware diagram.

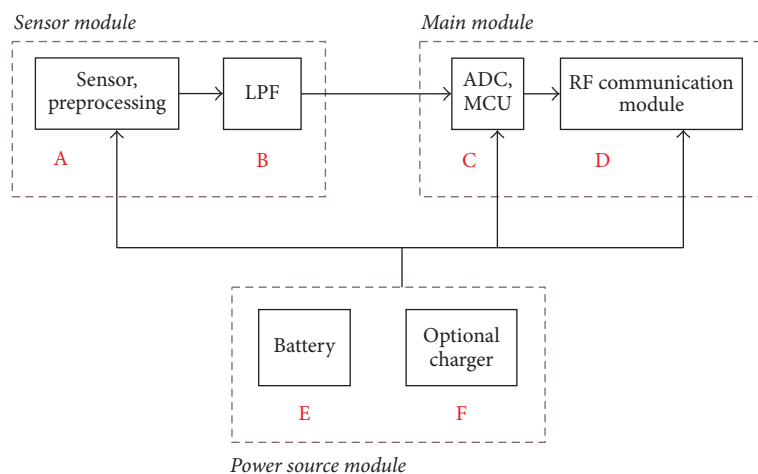


FIGURE 2: BAN unit designed for FlexiGuard.

not limit the user); minimum maintenance requirements; its use must not prolong the 2-minute time period for deployment and automatic use during intervention.

The results for our research and analyses are then primarily the requirements for high durability (i.e., low energy consumption), appropriate controlling unit (efficiency \times consumption ratio), wireless system (consumption \times range \times data flow), automatic use, simplicity, dimensions, and weight.

The development of the system further involved the design of the software as well as the three main hardware components of the device (sensor network, main module, and visualization unit).

2.1. Hardware Design Specifications. The hardware design was set in order to develop a modular telemetry system providing a simple connection between the sensory base, the receiver

(main unit), and visualization unit. Visualization unit may consist of a wide range of devices (laptop, smartphone, or tablet). The major components of the designed system however rely on the solution of sensory network and main unit. Figure 1 presents a chart of the hardware solution.

2.1.1. Sensor Network (WBAN System). Generally, the sensor network consists of several sensory WBAN units and main processing unit. Sensory units work independently; the main unit receives the measured data through simple tasks.

Sensory module comprises a sensor (Figure 2, part A) and supporting circuits such as low pass filter (Figure 2, part B) which processes the signal from sensor and converts analog data to digital if needed (Figure 2, part C). Microcontroller as well provides the data processing and communication with the wireless radio interface (Figure 2, part D). Power source

module serves to connect the battery (Figure 2, part E) or optional charger (Figure 2, part F).

In case of using an LiPol accumulator, it is possible to incorporate here the circuit providing the powering process or wireless powering module.

Based on the experience as well as feedback from the participating subjects, optimizations were made to the wireless measuring node. The typology of the system has proven itself useful and stays in use; changes primarily take place in the solutions for encasing this part of the system and further in the circuitry solutions for the measuring node which reflect the course of the current technology development.

The microcontroller type EFM32ZG (manufactured by SiliconLabs, ARM-Cortex0+ core) was used. Solution for the radio communication part of the sensory node represents a dedicated module. This eliminates several issues with the radio signal transmission. Furthermore, this solution is more feasible considering future certification (CE) as the dedicated module has already been certified by the manufacturer. This alternative applies current communication protocols and is therefore compatible both ways.

The result of the circuitry solution optimization process is also the decrease in the number of (passive) components on the printed circuit's board.

Following the experience from testing of the system's versions, space for connecting other peripherals was reduced and we used the universal board printed circuit.

Within the design optimization, respective sensor is placed directly on the main printed circuit of the sensory node; that is, each type of the sensory node (depending on the variable) has a slightly different printed circuit (board); nevertheless as for the modular design they all share a common platform of further evaluation and communication.

Encasing of the module and the powering unit is a major innovation proceeding from the experience from field testing. BAN node consists of a sensory unit smothered in polyurethane hermetical case and a powering unit, also hermetically isolated. Both parts are mounted on each other and held by the outer case. Due to requirements of the device use, emphasis was put on being humidity and sweat resistant, being simple to use, and having mechanical durability. The system of the sensory node consists of an individual measuring unit and the powering subsystem. Joining both of the parts and inserting them into the fixating case activate the device.

Sensitive electronics of the module are hermetically smothered using polyurethane resin. Powering inputs as well as service interface for setting communication parameters and firmware actualization are lined in the form of contact areas on the sides of the module. Depending on the measured variables, an own sensory element is either included (smothered) in the module (e.g., accelerometer module) or feasibly protruded from the module (e.g., contact temperature sensor).

The selected module concept enables the use of both the primary powering units in the case of nodes with low standards of power consumption (e.g., the widespread CR2032 3 V lithium battery type) as well as LiPol accumulators

(enhanced with a voltage stabilizer) for nodes with higher consumption.

The module design also includes a mechanical key for securing correct positioning of the powering and system module (contact areas have to face each other); the design calculates the positioning possibilities of the module around its vertical limit.

The wireless sensor network, which should consist of both digital and analog sensors placed on a subject's body, is connected through AD convertor. Each sensor connects to the main unit as soon as it is switched on within the range of the wireless network. Disconnection takes place by their deactivation or outside the reach of the network.

Basic wireless sensor network of proposed system includes thoracic harness used for heart activity detection, humidity sensor, temperature sensor, and accelerometer.

2.1.2. Main Unit. As already mentioned, the main unit receives the measured data and is able to control the sensory unit through simple commands (tasks).

Based on previous practical experience and measurement observations, the unit's hardware was adjusted for intervention monitoring. The adjustment affects both the printed circuit board upon which the main unit is constructed and its encasing. The adjustments reflect primarily the necessity of enhancing its mechanical and chemical resilience following the field testing experience pointing to rare cases of unit's deactivation caused by high amounts of humidity and sweat. Stability of the real time circuitry was enhanced as well, and upgrades also included the powering element of the main unit. Modular sensing unit used by each subject includes data acquisition, data processing, and two-way communication.

Main unit operates in two modes, the measuring and charging mode. Measuring mode activates by a hold switch located on the side of the device unless the device is connected to a PC or a charger through a USB cable. In this mode, the unit is in emergency status ready to connect to sensors. Frequency of saving and transmitting the data follows the device settings and the number of connected sensors.

Connecting the sensors (BAN nodes) is a virtually automatic process. After the sensor turns on within the reach of the wireless network, sensors are automatically connected to the main unit. This connection typically takes up to 30 seconds. Over the course of measuring, it is possible to connect or remove the sensors (BAN nodes) from the provided set. Disconnection of the sensors takes place by their deactivation or reaching outside the wireless range and reconnection then by their reactivation or returning to the area covered by the main unit without any restarts or manipulation with already connected sensors.

USB/charging mode activates once a unit is switched on by the hold switch and voltage is detected at the USB input (indicating the unit is connected to a PC or charger through a mini USB cable). This mode does not allow connection of sensors and proceeds with measuring. When a unit is connected through USB cable to the charger, only recharging takes place. As soon as a unit is connected to a PC, it is fully functional on every operation system as a Mass Storage Device besides being recharged. Before measuring it

is necessary to have *data.txt* file created in the root folder as this is where all the received data are stored. In case the file is empty, data is being stored here from the measuring initiation. Should the file already contain any data, new data will be stored at the end of the document to avoid their rewriting.

Main unit is built on a four-layer printed circuit with inner connective layers. Its shape was selected considering the used encasing (OKW box, Minitec series, size L). As a processing microcontroller, the LPC1769 chip by the NXP company is used. Sampling is provided by a crystal with the frequency of 12 MHz; the real time circuit is controlled by a 23.768 kHz crystal. The XB24CZ7UIS-004 module by Digi used for communication with the visualization unit connected to the UART interface as well as a line for controlling sleep mode. An AP281M4IB module provides wireless communication between sensors within the ANT network. It connects to the microcontroller similarly through UART interface together with several lines for controlling sleep mode, consumption, and baud setting. The printed circuit is further equipped with a holder for SD cards which are used as the storage device. Communication with an SD card is provided for by an SPI type interface. Powering branches of wireless modules of a card are supplied by MOS-FETs switched by the microcontroller. This enables total disconnection of powering from respective modules, thus reducing consumption to a minimum level if necessary.

A Li-Pol accumulator with its capacity of 1050 mAh together with a DC\DC LTC3530 convertor (set to 3,3 V) supplies the microcontroller with power. The accumulator is charged using the integrated LTD4054 circuit. As the input voltage, 5 V from the USB connector is used. The unit is furthermore ready for inductive charging of the Qi standard. This functionality has not been implemented into the training module but is to be used with the intervention module which represents one of the application requirements from end users. The status of accumulator is monitored by a simple circuit, once again switched by MOS-FET. This eliminates possible random loss of power in cases the unit has not been used for a longer time. An MCP111T module reduces risks of critical accumulator failures which would lead to its destruction. The module switches off the DC\DC convertor if low voltage is detected, therefore protecting the accumulator. Besides the above accumulator, the unit includes CRI216 coin button batteries holder for the purpose of powering the microcontroller's real time circuit during the absence or complete battery discharge. Modular sensing unit is also equipped by four LED indicators, button, USB connector, and a JTAG programming connector. The outer dimensions are $88 \times 50 \times 21$ mm. The main unit weights 57 grams.

2.2. Software Design Specifications. After the initial configuration has completed, the software functions independently without the necessity for operational interventions. The configuration is set in a way ensuring that, in cases of intervention module's and visualization unit's loss of connectivity, all data are stored on the internal memory of the intervention module and subsequently and automatically synchronized with the visualization unit. The system's functioning is therefore outage-free as for the quality of the measured data.

Current version is in accordance with the system concept fully modular and ready for work with multiple platforms of the controlling microprocessor. This solution will enable a low cost and technically simple porting of the entire operating firmware of the intervention module for potentially new or upgraded platforms based on the ARM Cortex A3 architecture. The created platform of embedded drivers remains the shared solution for firmware of the used sensory nodes for the used WBAN network and therefore applicable also to the ARM Cortex A0 platform. This solution offers flexible upgrades of the system as for new peripheries and functionality without the previous high cost and time consuming changes in the design of the whole system.

Firmware of the whole system is designed in the programming language C and optimized for the gnu ll standard. Firmware has been developed in the Eclipse environment and easy to implement.

The firmware configuration for a particular main unit runs through the configuration file. This is where the number of the main units is set as well as the desired number of up to eight sensors which would be used, time variables and so on. Complete setting possibilities are described in the file.

Once the controller is initialized and running, the main function is called. This initializes and sets up the entire unit, which is followed by the program entering an infinite loop which processes blocking and time consuming operations, especially the process of saving files to the storage device. The main loop is interrupted using operations of interruptions of individual peripheries such as timer, universal asynchronous serial port (UART), and analog-digital convertor.

The firmware uses the principle of double-buffering and saves the received data from sensors to either of the two magazines. As soon as a magazine is full, it switches to the other magazine and the full one is processed. The processing is initialized using the interruption of the timer (clocker) operation and proceeds further in the main loop where the content of the magazine can be converted in any combinations into ASCII code and can be saved on the SD card and sent to the visualization unit. The interruption also provides fully automatic connection process between sensors and the main unit. Data flow and processing diagram are shown in Figure 3.

Subsequently, a transmission system model of a BAN type (body area network [22]) was created based on the game theory (users behaving like game players). Considering, for example, a 3 s measuring cycle, during which all nodes are to send their respective measured data and the measuring is controlled deterministically, each node is assigned an exact time period of 0–3.000 ms during which it broadcasts (broadcast takes approximately 100 ms). An issue arises in case more nodes are connected that a single timeframe can contain, as simultaneous broadcast of two or more sensors causes collisions and data fail to transfer.

2.3. Design of the Transmission Network Model. The main idea of the discussed design was to avoid the issue arising with switching on the sensor unit, when the unit goes through a cold start and starts sending data to the main unit. As each unit switches on at a different time, this results in their

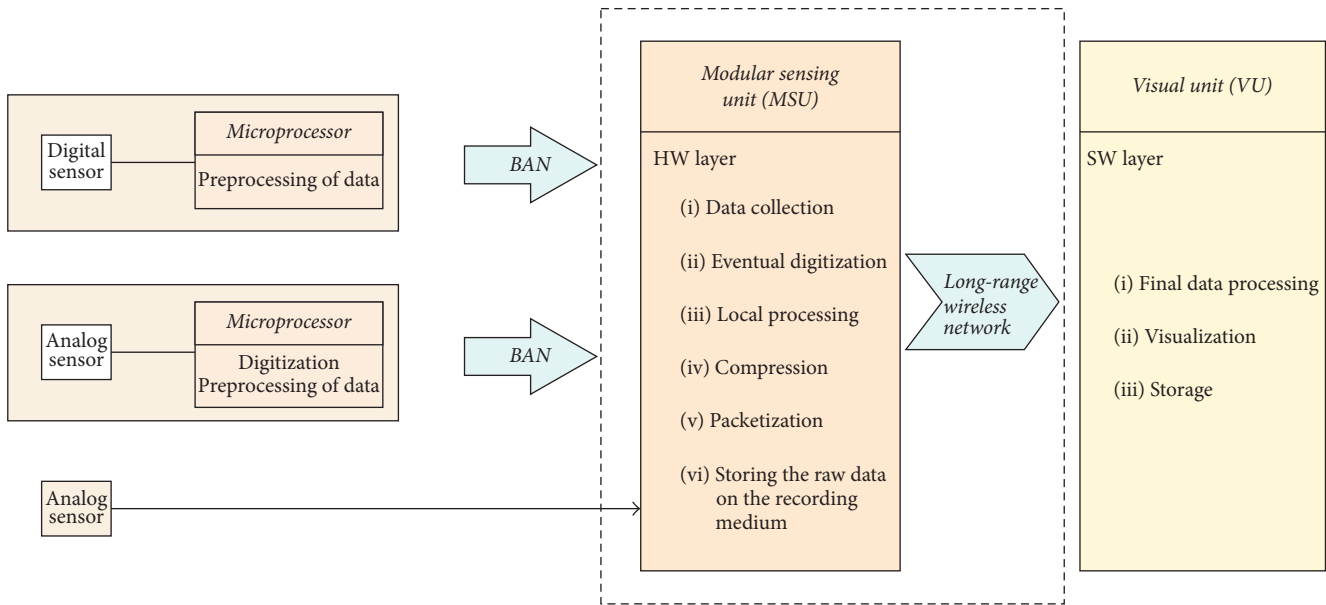


FIGURE 3: Data flow and processing.

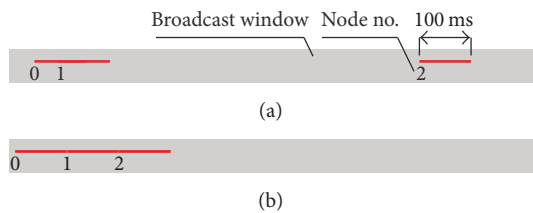


FIGURE 4: Example of possible units' colliding arrangement (a) and deterministic unit arrangement (b).

random arrangement in the broadband. Each unit broadcasts for a certain time period. As demonstrated on Figure 4(a), with three randomly arranged units, broadcasting times of units 0 and 1 collide and this results in no transmitted data. Unit 2 transmits all data. As per standard approach, the units receive a signal to stop transmitting. Then, each unit is addressed in order to obtain the total number of units. Once this information is obtained, the main unit assigns exact broadcasting times to the units and sends a signal to resume to data transfer. The resulting situation is demonstrated in Figure 4(b).

However, if the signal is not received when addressing the units, after the restart of the system the unit will then broadcast in the initial time and further collisions may occur. The signal is not received if the unit is outside the range or the batteries die. This is why the designed resolution method removes the main unit from control and leaves it to the units. Central unit only sends through the broadcast signal the amount of received data from respective units and successful broadcasting times. The units adjust their broadcasting time based on these data.

Model of the transmission network was realised using Javascript, HTML and CSS technologies. The ECMA 6 specification was used for Javascript and the code was subsequently transpiled using the `babeljs` compiler into ECMA 5 specification supported by current browsers. The entire development stack is controlled by `Gulp.js`, which represents technology for task management which supports previous technologies and individual tasks, such as compilation and minification of Javascript. The `node.js` technology was used for library management. At the same time, it is used as the environment for the compiler and `gulp.js`. The complete code runs in a browser on the user's end.

Basic construction of the model includes three elements simulating data transmission and timing, that is, central unit, clocker, and node (see Figure 5(a)). The entire model is based on a star typology in which nodes communicate only with the main unit and vice versa. This variation of typology does not allow nodes to communicate among each other.

Clocker is responsible for the functioning of the whole system (see Figure 5(b)). It works as an internal timer for all component systems. Upon activating the model, it launches under the `ServiceWorker` browser, which calls individual steps of nodes and the central unit during an infinite loop. `ServiceWorker` is a service (web thread) which runs in the browser on the background. Clocker has information on the broadcast window time, and depending on a given setting it is sent to broadcast units, once for every set of x broadcast rounds. `ServiceWorker` allows for switching between real time simulation and accelerated (faster) mode when clocker works at the maximum speed of the hardware.

Main unit is responsible for the count of the data sent successfully by the respective nodes (see Figure 5(c)). The inner implementation of the sent data count is based on the principle of associative array storing each unit that

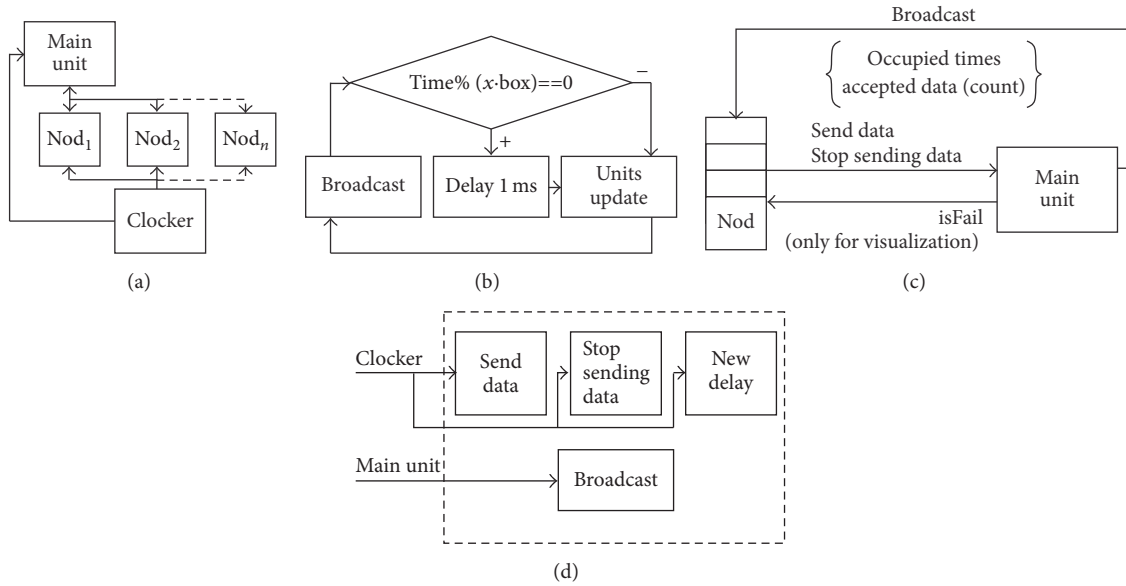


FIGURE 5: Flow chart of the basic model construction (a), inner implementation of clocker (b), main unit (c), and internal arrangement of a node (d).

starts broadcasting in a given time into the array of units broadcasting in the given time. If there is more than one unit in the array at a moment, the broadcast is considered as unsuccessful and the result is stored. The unit receives a message that it failed to transfer the data (due to visualization); however the unit disregards the message and does not further consider it. This is how the process described in the premises chapter is preserved, when the unit is not aware if it succeeded to transmit the data or not. The central unit also provides for sending the broadcast to all nodes active at the moment. Broadcast includes information on the amount of data sent by each unit and also information of occupied broadcast times.

Node is a single broadcasting unit in the system. Following the creation of a node type system, this node is included in the pole of nodes and using white noise it selects a random place in the broadcasted loop. From this place, it immediately starts to broadcast (transmit) with random latencies, modulated by red noise. Node incorporates the clock method which is called by the Clocker object and it is where the actual time in the window is stored. Once all the times have been generated, node waits until the method is called and, subsequently, depending on the generated times, proceeds with actions assigned to respective times (broadcast start, broadcast stop, and generating times for the next round). Once in x rounds the node receives the broadcast signal with the data on which times had the highest ratio of successful broadcasts and the number of successful broadcasts. It then adjusts its broadcast strategy based on the received information. Each node works as a state automat with two states, one when it broadcasts and the other when it is waiting for the next broadcast. Figure 5(d) demonstrates the internal arrangement of a node.

Each node furthermore includes the strategy module which is used by the unit to store the last six entries on success

percentage of broadcast. If the arithmetic mean of the total success percentage drops below the set threshold, the unit performs a so-called jump. It differs from the initial jump primarily in that the new position is selected from the pole of random numbers which are filtered using values generated from the memory heatmap. This heatmap is generated by each node before the jump.

Every broadcast carries information on the times of successful data broadcasts (transmissions) and uses this information to assign “scores” of respective times in its memory heatmap. To reduce the number of possible accidents, considering identical memory heatmaps, partially random forgetting of these heatmap prints takes place. This eliminates the possibility of an identical heatmap for two different units.

Before a jump, a unit assigns scores to times generated by the white noise. Assigning is processed as the unit considers the selected broadcast time from the heatmap and calculates the total score from its closest surroundings. All times with scores above the mean value remove the nod from the random selection.

During broadcast, unit selects a change in strategy if the conditions allow it. If 100% data transmission occurs over six consecutive broadcasts, the unit attempts a move towards zero times. This choice of strategy is desirable due to the fact that the units, considering the random nature of selecting broadcast times, tend to create considerable gaps between themselves which, however, are not big enough for another unit to fit into. This way they take place and a major part of bandwidth remains unused. Unit always attempts a random move by several tens of milliseconds. When its broadcast success decreases, it returns to its initial place before the move and the attempted move is no longer considered.

When the unit performs a big jump, the sensor for small moves is automatically reset to zero. It is possible to a certain

TABLE 1: Overview of the adjustable settings and standard model setting values.

Parameter	Description	M_{S8} setting value
Box (s)	Broadcast broadband capacity (for one round)	1
Jump threshold (%)	Percentage variable defining the level of threshold under which nodes opt for jump	20
x -broadcast	Variable defining number of rounds after which broadcast message launches	3
Move threshold (%)	Percentage variable defining success level required to attempt small move	100
Start lower threshold	Indicating whether a node decides to jump over or below the threshold	True
Experiment time (s)	Duration of the measuring	300
Start delay (s)	Initial broadcast time of a node	0.1
Rounds stay	Defining the difference between threshold and a unit's success ratio required for a unit to perform a jump	6
Fixed time	Switching to deterministic mode	False
Fast simulation	Switching from real time to fast simulation	True
Intelligence	Dismissing all strategies and switching to the random broadcast mode	True
Memory	Value which will appear in the memory map	3
Min shift (ms)	Minimum move when making a small move	1
Max shift (ms)	Maximum move when making a small move	10
Nodes	Total number of nodes	8

extent that unit stops moving due to fluctuations caused by red noise despite the available space around. Due to this fact, the sensor is reset with 1% probability also with unsuccessful jumps.

For the purposes of the model, a graphic interface was created allowing for adjustments in basic parameters of the entire system as well as monitoring its condition. Also, a visualization unit was created using the oCanvas library providing real time monitoring of the arrangement of units in broadcasting times and the duration of their broadcasts. The interface also includes a switch designed for switching the system from stochastic to deterministic, in which each unit calculates its broadcasting time based on its ID and the duration of the broadcasting window.

$$\text{Time} = \text{ID}_{\text{unit}} \cdot t, \quad (1)$$

where t is the actual time of the broadcast and ID_{unit} represents its position in the pole of units. In the deterministic state, unit does not oscillate around the broadcast time but transmits precisely in the calculated broadcast time. This implies limits of unit broadcasting in the deterministic mode, which, when exceeded, no other units fit into the broadcast bandwidth and thus produce a 100% data loss.

$$\text{limit} = \frac{\text{broadcast time window}}{t}. \quad (2)$$

The most vital adjustable parameters of the model were incorporated into the designed software. Table 1 provides characteristics of the adjustable parameters. The tab also includes settings of the model standard type (M_{S8}), which were compared to other adjustment settings. This type of settings proved the most stable and most successful during the experiments.

For further evaluation of the broadcasting success of the designed broadcasting strategy, strategic setting of the

standard type (see Table 1) was compared to the white noise broadcast that simulates cold starts of the units (i.e., the units performs cold start after each broadcast and its broadcasting time changes). Also, the designed broadcasting strategy was compared to the white noise broadcast in the case of full broadcasting range. The effects of memory, broadcast duration, jumps, and moves to the success of broadcasting were also examined. Fragmentation of time for strategic setting of M_{S8} was also examined within the scope of evaluation.

A set of 20 experiments was performed for each setting type of the model, and the duration of each experiments was 300 seconds. Having the broadcast set to once in three rounds, the success was evaluated every 3 seconds, which yields 100 values of success for each unit for one measurement. In the case of experiments with 8 broadcasting units it is 2,000 values of success for each unit, and therefore 16,000 values for the total of the 20 experiments.

3. Results and Discussion

For the purposes of evaluating the suitability of the designed strategy, success percentage of the broadcast was examined using standard setting for 8 nodes (M_{S8}), standard setting for fully occupied bandwidth, that is, for 10 nodes (M_{S10}), broadcasting using white noise (M_{wn8}) for 8 nodes, and broadcasting using white noise with fully occupied bandwidth (M_{wn10}). Success percentage was further measured when changing standard settings, particularly with the setting of so-called aggressive memory, which means that the standard memory was set to 10 and the forgetting parameter was set to 1.

Within the scope of results evaluation, normality of the measured data was considered using Shapiro-Wilk test of normality at the significance level of $p = 0.05$, when the normal distribution hypothesis was rejected for $p < 0.05$. Due to the fact that the normality test fails to confirm normal distribution of all measured data, Wilcoxon nonparametric

TABLE 2: Results of Wilcoxon test in the form of p values for the measured data distributions.

	M_{S8}	M_{wn8}	$M_{S8-Me-10}$	M_{wn10}	M_{S10}	$M_{S8-Sh-50}$	$M_{S8-Mo-20}$
M_{S8}	—	$<2.2e - 16$	0.1928	$<2.2e - 16$	$<2.2e - 16$	$2.795e - 08$	$2.428e - 13$
M_{wn8}	—	—	$<2.2e - 16$	$1.46e - 08$	$<2.2e - 16$	$<2.2e - 16$	$<2.2e - 16$
$M_{S8-Me-10}$	—	—	—	$<2.2e - 16$	$<2.2e - 16$	$7.434e - 08$	$1.108e - 15$
M_{wn10}	—	—	—	—	$<2.2e - 16$	$<2.2e - 16$	$<2.2e - 16$
M_{S10}	—	—	—	—	—	$<2.2e - 16$	$8.812e - 15$
$M_{S8-Sh-50}$	—	—	—	—	—	—	0.0003284
$M_{S8-Mo-20}$	—	—	—	—	—	—	—

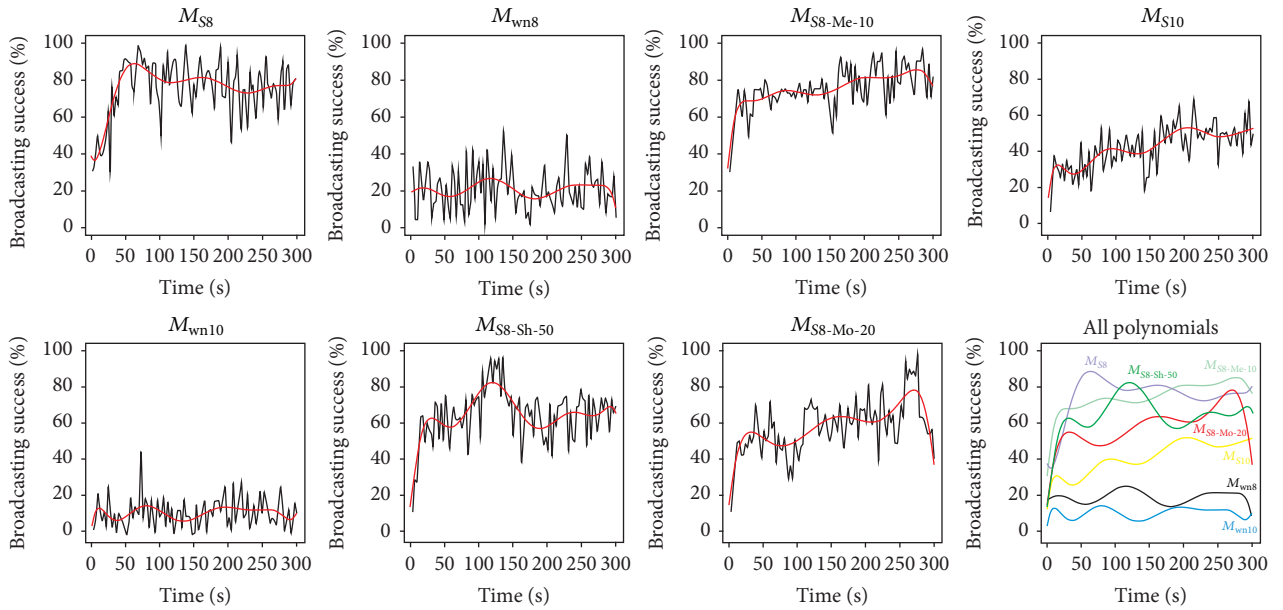


FIGURE 6: Mean broadcast success ratio depending on the selected broadcasting strategy.

test was used for further evaluation of differences in data distribution, testing the hypothesis that medians of two examined data vectors come from the same distribution. Significance level of $p = 0.05$ was used for the test, while hypothesis about intergroup differences on the level of medians was accepted for $p < 0.05$. The results of the Wilcoxon test show that statistically significant differences were found in all cases except when comparing M_{S8} compared to $M_{S8-Me-10}$; see Table 2.

Graphic representation of the summarized results of the mean success of units adjusting their behavior based on the prescribed strategy when using white noise and changing the above described parameters is demonstrated by Figure 6. Red curve crosses the data with polynomial of degree three.

When comparing the mean broadcast success ratio using M_{S8} to mean broadcast success ratio using M_{wn8} , it was found that the mean success ratio is 4 times greater with the designed broadcasting strategy (see Figure 6). In this case, comparing the broadcasting time of 1000 ms, the bandwidth is occupied at 70%. In the case of fully occupied bandwidth, the selected strategy M_{S10} succeeds better with every instance

compared to broadcasting using white noise M_{wn10} , approximately 4 times (see Figure 6).

When changing the memory setting to aggressive memory, no significant difference was found between $M_{S8-Me-10}$ and M_{S8} . The aggressive memory setting means that every broadcasting time leaves a greater trace in the memory print, which however had no influence on the success of broadcasting.

At high values of max shift (max shift > 50 ms), decrease in the mean success ratio of units was observed at high probability due to more frequent collisions during moves (see $M_{S8-Sh-50}$ on Figure 6). The system does not converge into a state which would allow for placing a unit in a freshly created free slot. Small shift describes the effort of a unit to defragment time. In other words, the value of max shift basically characterizes the speed of defragmentation; however increasing max shift values increases the probability of collision. With lower values, the free time defragments at slower speed. Statistically, a situation may take place in some cases when the units occupy broadcasting times inefficiently, and due to slow shift and slower defragmentation of free

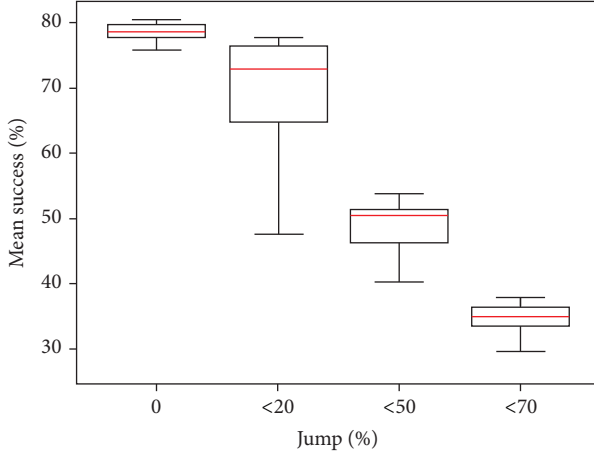


FIGURE 7: Effect of jump threshold on mean success.

time, some units would wait longer for a free slot. At lower values, as in the case of M_{S8} , with the max shift ≤ 20 ms, lesser collisions happen owing to the shifts and therefore also better defragmentation. Therefore, shift has no direct effect on the units' distribution; it is nevertheless vital for providing free broadcast range for next possible units.

The move parameter sets the success ratio at which the units attempt to defragment time. By standard setting, the value is set to 100%. In the diagram for $M_{S8-M0-20}$, this value is set to 20%; that is, units will attempt time defragmentation at low success ratios. Time defragmentation will therefore start earlier, which however causes more frequent collisions decreasing success, as illustrated.

Besides the above, the article aimed to identify the effect of jump threshold on the mean broadcast success ratio. Figure 7 demonstrates the results in the form of boxplots, with each boxplot describing the distribution of the mean broadcasting success ratios of nodes for respective jump threshold settings for M_{S8} . At the same time, each boxplot illustrates, starting from the bottom vertical line, the distribution minimum, first quartile, median, third quartile, and maximum. Graphic representations suggest that increasing the value of threshold for unit's big jump decreases the success of units. Units perform jump more frequently and come closer to the success of white noise. Performing jumps was experimentally proven to be statistically most accurate when reaching 0% success ratio. The influence of the jump threshold is defined by the jump threshold parameter (see Table 1), and it can be said that increasing the value of this parameter brings the system closer to the setting of the white noise.

Besides the presented results, the behavior of the model M_{S8} when changing x parameter was measured (see Table 1), that is, when changing the duration of the broadcast. Duration of the broadcast affects primarily the steepness of the success ratio curve, as presented in Figure 8. Increasing the broadcast duration slows down the feedback to the system and units therefore broadcast potentially longer in the collision spots. With lower values, the feedback speeds up which results in faster adaptation of the system; this however

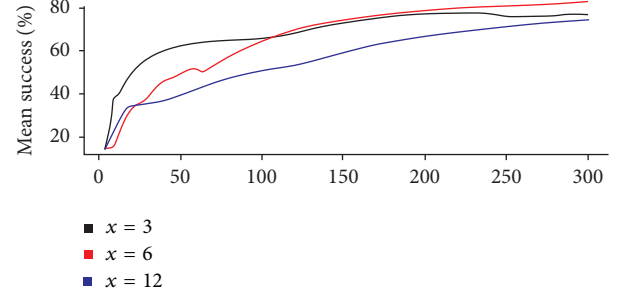


FIGURE 8: Mean broadcasting success depending on broadcast duration M_{S8} .

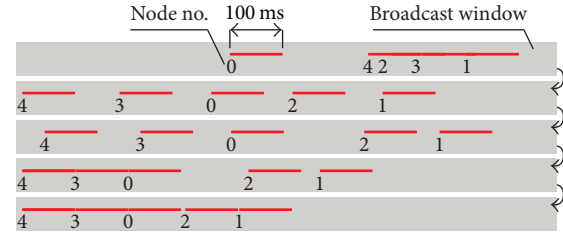


FIGURE 9: Example of gradual time defragmentation by the designed software.

causes increased energy consumption. Figure 8 demonstrates the decrease in success percentage caused by lower amount of broadcast messages received by units over the same experiment duration, which results in lower adaptation speed of the system.

The entire model and the source code were systematically tested aiming to eliminate implementation errors, enhancing accuracy of generating noise functions and functionality of the idea of decentralized operation. The designed operation system using competitive strategy is functional, as presented by the attached diagrams and confirmed by statistic tests. Figure 9 illustrates how basic arrangement of units into the broadcasting range is followed by applying the strategy of small moves of the units in the broadcast and their convergence to the left of the broadcasting range. The condition for move can be set by the *move threshold* parameter.

In the work of Gorman et al. [23], the following formula is used for fragmentation calculation:

$$\text{fraglevel} = \frac{\text{Total Free Pages} - \sum (2^i k_i)}{\text{Total Free Pages}} \times 100. \quad (3)$$

However, this formula does not consider possible collisions as it applies to a spot in the memory with no possibility of collisions. To describe time fragmentation by parameters, this formula was adjusted to consider use in an environment allowing for situations with a seemingly free slot. A seemingly free spot in the broadcast time represents a situation when units broadcast in collision, and these collision times appear

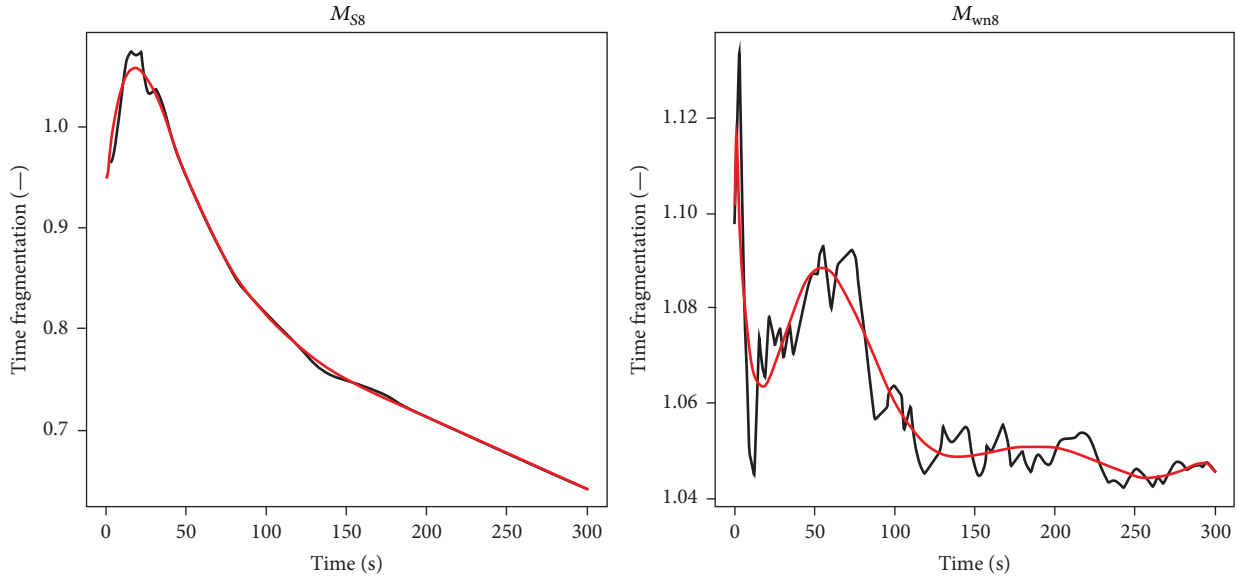


FIGURE 10: Defragmentation of time in standard and white noise setting.

to increase the free spot. The adjusted formula offers an idea of a free spot depending on collisions f_c .

$$f_c = \frac{\sum t_f - (\sum t_{f_{\max}} - \sum t_c)}{\sum t_f}, \quad (4)$$

where t_f represents free time, t_c represents collision time, and $t_{f_{\max}}$ is the total sum of the greatest block of free time. If the value of fragmentation exceeds 1, it means that collision time is greater than free time. The following generalized formula is used to determine the maximum value of f_c :

$$f_{c_{\max}} = \frac{(n-1)t_v}{t_{\max} - t_v}, \quad (5)$$

where t_{\max} represents the total broadcast duration, t_v is the duration of the broadcast time of a unit, and n represents the number of units.

Figure 10 presents the behavior of f_c parameter in the designed optimum model M_{S8} and M_{wn8} model. The parameter demonstrates the decrease of the value of time fragmentation, which indicates an increasing free spot in the broadcast time in M_{S8} . For comparison, we may observe that the value of the fragmentation parameter in M_{wn8} is always greater than 1; that is, the units experience more collisions than free time.

4. Conclusion and Future Work

FlexiGuard modular telemetry system was developed by the authors for the use by Integrated Rescue System bodies. Individual parts of the system were designed with the focus on minimum demands on the user, automatic use, and maximum modularity. Functionality of the system was verified by a series of practical laboratory tests as well as under real

conditions. Based on the performed testing and feedback from end users, the system was fine-tuned. Furthermore, a model of a broadcast network was created based on the theory of games. A new decentralized stochastic model of data flow management in a wireless biosensory network based on competitive strategy relying on the theory of games was also designed and tested. The model is operated by a delayed feedback proceeding from an approach which uses units with separated broadcasting and receiving components and are incapable of an immediate decision on successful or unsuccessful broadcast. If the units are switched on, they go through a cold start when broadcasting in white noise. This noise was modeled and compared to the approach of decentralized operation, in which the units receive delayed feedback on broadcasting times of other units together with the information on their own success. In the simulations, the designed competitive operation strategy, which evaluates each unit based on its success and changes the broadcasting strategy accordingly, has proven several times more efficient than leaving the units in their condition after the cold start. Each unit changes its strategy over the course of broadcast not only on the basis of success percentage. Several of the model parameters are controlled by noise functions; therefore the units do not behave identically. This is furthermore given by the fact that each unit creates its own memory heatmap of broadcast times modulated by a random selection.

The original design used the centralized operation which stops all units and then gradually calls them, which is followed by ascribing deterministic broadcast time to each unit in which they broadcast. This approach loses its effect, if does not fail completely, in situations with the number of sensory units unknown or changing over time, when broadcast times of units overlap and the central unit does not receive the signal. This leads the central unit to assume that the involved units are incapable of broadcast, switched

off, or out of the network range, and the deterministic control system fails.

We are currently working on implementing adjustments to the broadcast network in to the FlexiGuard units and testing larger numbers of units in cooperation with the army.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work has been supported by the Ministry of Education, Youth and Sports within National Sustainability Programme I, Project no. LO1605. The authors would also like to thank Andrej Madoran, B.A., for the translation of this work.

References

- [1] D. Giovanelli and E. Farella, "Force sensing resistor and evaluation of technology for wearable body pressure sensing," *Journal of Sensors*, vol. 2016, Article ID 9391850, 2016.
- [2] F. J. Martinez-Tabares, Y. J. Costa-Salas, D. Cuesta-Frau, and G. Castellanos-Dominguez, "Multiobjective design of wearable sensor systems for electrocardiogram monitoring," *Journal of Sensors*, vol. 2016, Article ID 2418065, 2016.
- [3] T. Liang and Y. Yuan, "Wearable medical monitoring systems based on wireless networks: a review," *IEEE Sensors Journal*, vol. 16, no. 23, pp. 8186–8199, 2016.
- [4] C. Zocchi, A. Rovetta, and F. Fanfulla, "Physiological parameters variation during driving simulations," in *Proceedings of the IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM '07)*, pp. 1–6, IEEE, Zurich, Switzerland, September 2007.
- [5] A. Sahayadhas, K. Sundaraj, and M. Murugappan, "Detecting driver drowsiness based on sensors: a review," *Sensors*, vol. 12, no. 12, pp. 16937–16953, 2012.
- [6] L. Liu, E. Stroulia, I. Nikolaidis, A. Miguel-Cruz, and A. Rios Rincon, "Smart homes and home health monitoring technologies for older adults: a systematic review," *International Journal of Medical Informatics*, vol. 91, pp. 44–59, 2016.
- [7] J.-V. Lee, Y.-D. Chuah, and K. T. H. Chieng, "Smart elderly home monitoring system with an android phone," *International Journal of Smart Home*, vol. 7, no. 3, pp. 17–32, 2013.
- [8] J.-V. Lee, Y.-D. Chuah, and K. T. H. Chieng, "Development of an elderly telecare system," *International Journal of Control and Automation*, vol. 6, no. 3, pp. 57–68, 2013.
- [9] L. Fu, P. Dallas, V. K. Sharma, and K. Zhang, "Sensors for environmental monitoring," *Journal of Sensors*, vol. 2016, Article ID 4108790, 2016.
- [10] M. Regula, V. Socha, P. Kutilek et al., "Study of heart rate as the main stress indicator in aircraft pilots," in *Proceedings of the 16th International Conference on Mechatronics, Mechatronika*, pp. 639–643, December 2014.
- [11] V. Socha, J. Schlenker, P. Kalavksý et al., "Effect of the change of flight, navigation and motor data visualization on psychophysiological state of pilots," in *Proceedings of the 13th IEEE International Symposium on Applied Machine Intelligence and Informatics (SAMI '15)*, pp. 339–344, January 2015.
- [12] V. Socha, S. Szabo, L. Socha, P. Kutilek, and V. Nemeč, "Evaluation of the variability of respiratory rate as a marker of stress changes," in *Proceedings of the International Conference on Transport Means*, vol. 2014, pp. 339–342, January 2014.
- [13] E. Ben Hamida, M. M. Alam, M. Maman, B. Denis, and R. D'Errico, "Wearable Body-to-Body networks for critical and rescue operations—The CROW2 project," in *Proceedings of the 25th IEEE Annual International Symposium on Personal, Indoor, and Mobile Radio Communication (IEEE PIMRC '14)*, pp. 2145–2149, September 2014.
- [14] M. Klann, "Tactical navigation support for fire-fighters: the LifeNet ad-hoc sensor-network and wearable system," in *Mobile Response 2008*, pp. 41–56, Springer, Berlin, Germany, 2009.
- [15] M. Sung, C. Marci, and A. Pentland, "Wearable feedback systems for rehabilitation," *Journal of NeuroEngineering and Rehabilitation*, vol. 2, article 17, 2005.
- [16] T. Gao, C. Pesto, L. Selavo et al., "Wireless medical sensor networks in emergency response: implementation and pilot results," in *Proceedings of the IEEE International Conference on Technologies for Homeland Security (HST '08)*, pp. 187–192, May 2008.
- [17] K. Sha, W. Shi, and O. Watkins, "Using wireless sensor networks for fire rescue applications: requirements and challenges," in *Proceedings of the IEEE International Conference on Electro/Information Technology*, pp. 239–244, IEEE, East Lansing, Mich, USA, May 2006.
- [18] E. Jovanov, M. Milosevic, and A. Milenkovic, "A mobile system for assessment of physiological response to posture transitions," in *Proceedings of the 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC '13)*, pp. 7205–7208, July 2013.
- [19] G. Magenes, D. Curone, L. Caldani, and E. L. Secco, "Fire fighters and rescuers monitoring through wearable sensors: The ProeTEX project," in *Proceedings of the 32nd Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC '10)*, pp. 3594–3597, September 2010.
- [20] J. Schlenker, V. Socha, P. Smrčka et al., "FlexiGuard: modular biotelemetry system for military applications," in *Proceedings of the 5th International Conference on Military Technologies (ICMT '15)*, May 2015.
- [21] Z. Hon, P. Smrcka, K. Hana et al., "A surveillance system for enhancing the safety of rescue teams," *Komunikacie*, vol. 17, no. 1, pp. 81–86, 2015.
- [22] R. Gravina, P. Alinia, H. Ghasemzadeh, and G. Fortino, "Multi-sensor fusion in body sensor networks: state-of-the-art and research challenges," *Information Fusion*, vol. 35, pp. 68–80, 2017.
- [23] M. Gorman, C. Damastown, and P. Healy, "Measuring the impact of the Linux memory management," in *Proceedings of the Libre Software Meeting*, Dijon, France, July 2005.

